



管理ガイド

Amazon AppStream 2.0



Amazon AppStream 2.0: 管理ガイド

Copyright © 2025 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、それぞれの所有者の所有物であり、Amazon と提携、接続、または後援されている場合とされていない場合があります。

Table of Contents

Amazon AppStream 2.0 とは何ですか？	1
機能	1
主なコンセプト	3
開始方法	6
アクセス	6
セットアップ	8
にサインアップする AWS アカウント	8
管理アクセスを持つユーザーを作成する	8
使用開始: サンプルアプリケーションのセットアップ	11
ステップ 1: サンプルスタックをセットアップし、イメージを選択して、フリートを設定する	11
ステップ 2: ユーザーへのアクセスを提供する	20
リソース	21
ネットワークとアクセス	23
インターネットアクセス	23
VPC の要件	25
VPC セットアップの推奨事項	25
プライベートサブネットの VPC および NAT ゲートウェイを設定する	28
パブリックサブネットを持つ VPC の設定	37
デフォルトの VPC とパブリックサブネットを使用する	41
Amazon S3 VPC エンドポイント	44
VPC への接続	45
ネットワークインターフェイス	45
管理ネットワークインターフェイス IP アドレス範囲とポート	46
カスタマーネットワークインターフェイスポート	47
AppStream 2.0 へのユーザー接続	48
推奨帯域幅	48
IP アドレスとポートの要件	50
許可されたドメイン	51
Image Builder	54
Image Builder を起動する	55
Image Builder に接続する	58
コンソール (ウェブ接続)	59
ストリーミング URL (クライアントまたはウェブ接続)	60

Image Builder のアクション	62
Image Builder のインスタンスメタデータ	64
Graphics Design インスタンスへの AMD ドライバーのインストール	64
ベースイメージおよびマネージド型イメージの更新版リリースノート	66
イメージ	89
デフォルト設定とアプリケーション起動パフォーマンス	90
デフォルトのアプリケーションおよび Windows 設定を作成する	90
アプリケーションの起動パフォーマンスの最適化	91
エージェントのバージョン管理	92
最新バージョンのエージェントを使用するイメージの作成	92
特定バージョンの エージェントを使用するイメージの作成	93
より新しいバージョンの エージェントを使用するイメージの作成	95
エージェントのリリースノート	96
チュートリアル: コンソールを使用したカスタムイメージの作成	128
ステップ 1: Image Builder でアプリケーションをインストールする	129
手順 2: AppStream 2.0 アプリケーションカタログを作成する	130
ステップ 3: デフォルトのアプリケーションと Windows の設定を作成する	131
ステップ 4: アプリケーションをテストする	133
ステップ 5: アプリケーションを最適化する	134
ステップ 6: イメージの作成を完了する	135
ステップ 7 (オプション): イメージにタグを付け、コピーする	136
ステップ 8: クリーンアップ	138
イメージの管理	138
プライベートイメージの削除	139
自分が所有しているイメージを別の リージョンにコピーする	140
自分が所有しているイメージを 別の アカウントと共有する	140
自己所有イメージの共有の停止	142
イメージを最新の状態に保つ	143
Windows Update とウイルス対策ソフトウェア	147
プログラムによる新しいイメージの作成	150
プログラムを使用した イメージの作成	150
デフォルトのアプリケーションおよび Windows 設定	151
アプリケーションの起動パフォーマンス	152
プロセスの概要	153
Image Assistant CLI オペレーション	154
Linux ベースのイメージを作成する	164

ユーザー用のデフォルトアプリケーション設定の作成	164
Linux ユーザー用のデフォルト環境変数の作成	168
Linux アプリケーションの起動パフォーマンスの最適化	169
セッションスクリプトの作成	170
Linux 用の Image Assistant CLI ツールの使用	171
ウェブカメラのサポートの有効化と無効化	171
ホームフォルダのヘビーファイル同期モードの有効化と無効化	172
チュートリアル: Linux ベースのカスタムイメージを作成する	173
チュートリアル: 日本語サポートを有効にする	181
ユーザーのストリーミングエクスペリエンスを管理するセッションスクリプト	186
ストリーミングセッションの開始前にスクリプトを実行する	186
ストリーミングセッションの終了後にスクリプトを実行する	191
セッションスクリプトを作成および指定する	193
セッションスクリプト設定ファイル	195
Windows PowerShell ファイルの使用	198
セッションスクリプト出力のログ記録	198
セッションスクリプトでストレージコネクタを使用する	199
セッションスクリプトログに対して Amazon S3 バケットストレージを有効にする	200
マルチセッションフリートでセッションスクリプトを使用する	203
アプリケーションマネージャー	206
App Block	206
カスタム App Blocks	207
AppStream 2.0 App Block	217
サポートされていないアプリケーション	229
App Block Builder	231
App Block Builder を作成する	232
App Block Builder に接続する	234
App Block Builder のアクション	236
アプリケーション	237
S3 バケットにアプリケーションアイコン、セットアップスクリプト、セッションスクリプト、および VHD を保存する	239
Amazon S3 バケット許可	240
アプリケーションを Elastic フリートに関連付ける	242
その他のリソース	242
フリートとスタック	244
セッションコンテキスト	244

セッションコンテキストを使用してストリーミングアプリケーションにパラメータを渡す	245
フリートタイプ	247
常時オンおよびオンデマンドフリート	248
Elastic フリート	249
インスタンスファミリー	249
フリートとスタックを作成する	253
フリートを作成する	253
スタックの作成	260
ユーザーへのアクセスを提供する	266
リソースのクリーンアップする	266
フリートのカスタマイズ	267
環境変数の永続化	268
デフォルトのファイルの関連付けを設定する	272
Internet Explorer セキュリティ強化の構成を無効にする	275
Internet Explorer のデフォルトホームページを変更する	277
ユーザーおよびインスタンスメタデータ	282
フリートを更新する	283
新しいイメージを使用してフリートを更新する	284
Elastic フリートに関連付けられたアプリケーションを管理する	285
Fleet Auto Scaling	286
スケーリングの概念	287
コンソールを使用したフリートスケーリングの管理	290
CLI を使用したフリートスケーリングの管理	292
その他のリソース	300
マルチセッションに関する推奨事項	300
ユーザー認証	304
ユーザープール	304
ユーザープールのエンドユーザーエクスペリエンス	305
パスワードを忘れた場合のリセット	305
ユーザープールの管理	306
SAML 2.0 の統合	311
認証ワークフローの例	311
SAML のセットアップ	313
AppStream 2.0 と SAML 2.0 の統合	328
アクティブディレクトリの使用	332

アクティブディレクトリドメイン	333
開始する前に	335
アクティブディレクトリドメイン環境	335
ドメイン参加済みの AppStream 2.0 ストリーミングインスタンス	336
グループポリシー設定	337
スマートカード認証	338
チュートリアル: セットアップ	338
ステップ 1: Directory Config オブジェクトを作成する	339
ステップ 2: ドメイン結合 Image Builder を使用してイメージを作成する	340
ステップ 3: ドメイン結合フリートを作成する	341
ステップ 4: SAML 2.0 を設定する	342
証明書ベースの認証	343
前提条件	344
証明書ベースの認証	348
証明書ベースの認証の管理	349
クロスアカウント PCA 共有を有効にする	350
管理	351
アクティブディレクトリコンピュータオブジェクトを作成および管理するための許可の付与	351
組織単位の識別子名を検索する	353
Image Builder のローカル管理者権限を付与する	354
ドメインの参加に使用するサービスアカウントの更新	356
ユーザーがアイドル状態の場合にストリーミングセッションをロックする	357
ディレクトリ設定を編集する	359
ディレクトリ設定を削除する	360
ドメインの信頼度を使用するように AppStream 2.0 を構成する	360
アクティブディレクトリでの AppStream 2.0 コンピュータオブジェクトの管理	361
詳細情報	362
カスタムブランディングを追加する	363
カスタムブランドのオプション	363
カスタムブランディングの追加	366
カスタムのリダイレクト URL とフィードバック URL	368
カスタムブランディングの変更のプレビュー	368
カラーテーマパレット	369
赤	369
水色	370

ブルー	371
ピンク	372
埋め込み ストリーミングセッション	374
前提条件	374
推奨事項と使用上の考慮事項	375
ステップ 1: ホストドメインを指定する	376
ステップ 2: ストリーミング URL を作成する	377
カスタムドメインを使用するための設定要件	377
ステップ 3: 埋め込み ファイルをダウンロードする	379
ステップ 4. 統合するためにウェブサイトを設定する	380
appstream-embed JavaScript ファイルをインポートする	380
AppStream.Embed インターフェイスオブジェクトを初期化して設定する	380
AppStream 2.0 ユーザーインタフェイスで項目を非表示にする例	382
定数、関数、イベント	383
「HIDDEN_ELEMENTS」の使用	384
AppStream.Embed オブジェクトの関数	385
埋め込み AppStream 2.0 ストリーミングセッションのイベント	386
イベントリスナーの追加例と埋め込み AppStream 2.0 ストリーミングセッションの終了 例	389
永続的ストレージを管理する	391
ホームフォルダを管理する	391
計算集約型アプリケーションに関連するファイルとディレクトリ	393
AppStream 2.0 ユーザーのホームフォルダを有効にする	393
ホームフォルダを管理する	395
Google Drive を管理する	401
AppStream 2.0 ユーザーの Google ドライブを有効にする	402
AppStream 2.0 ユーザーの Google ドライブを無効にする	406
OneDrive for Business を管理する	407
AppStream 2.0 ユーザーの OneDrive を有効にする	408
AppStream 2.0 ユーザーの OneDrive を無効にする	410
カスタム共有フォルダの管理 (SMB ネットワークドライブ)	411
マップサーバーメッセージブロック (SMB) ネットワークドライブ	412
ユーザーのアプリケーション設定の永続化を有効にする	418
アプリケーション設定の永続化の仕組み	418
アプリケーション設定の永続化の有効化	421
アプリケーション設定の永続化を有効にするための前提条件	421

アプリケーション設定の永続化のベストプラクティス	421
アプリケーション設定の永続化を有効にする方法	422
ユーザーのアプリケーション設定 VHD の管理	423
Amazon S3 バケットのストレージ	423
ユーザーのアプリケーション設定のリセット	425
Amazon S3 オブジェクトのバージョニングを有効にしてユーザーのアプリケーション設定 を元に戻す	426
アプリケーション設定 VHD のサイズ拡大	427
ユーザー用地域の設定を有効にする	430
ユーザーのためのデフォルトのリージョン設定を構成する	430
デフォルトのタイムゾーンを指定する	431
デフォルトの表示言語を指定する	435
デフォルトのシステムロケールを指定する	437
デフォルトのユーザーロケールを指定する	439
デフォルトの入力メソッドを指定する	441
アプリケーション設定の永続化に関する特別な考慮事項	443
日本語の言語設定に関する特別な考慮事項	443
ユーザーが自分のリージョンを構成できるようにする	444
サポート対象ロケール	445
AppStream 2.0 ユーザー用のリージョン設定を有効にする	446
アプリケーションの使用権限の管理	449
属性ベースのアプリケーションの使用権限	449
アプリケーションの使用権限の作成	449
SAML 2.0 マルチスタックアプリケーションカタログ	452
動的アプリケーションフレームワーク	453
API オペレーションのワークフロー例	454
動的アプリケーションフレームワークを使用する	455
動的アプリケーションプロバイダーの有効化	462
動的アプリケーションプロバイダーのテスト	463
その他のリソース	465
ユーザーにアクセスを許可する	466
サポートされている機能	466
ウェブブラウザを介したアクセスを許可する	474
要求事項と特徴	475
ユーザー接続方法の設定	479
クライアントを介したアクセスを許可する	480

要求事項と特徴	481
クライアントのインストールおよび設定	495
リソースのタグ付け	529
タグ付けの基本	529
タグの制限	530
リソース作成中のタグの追加	531
タグの追加、編集、削除	531
API、SDK、または CLI の使用	532
モニタリングとレポート	535
リソースのモニタリング	535
コンソールを使用してフリートの使用状況を表示する	535
コンソールを使用してインスタンスおよびセッションのパフォーマンスメトリクスを表示する	536
AppStream 2.0 のメトリクスとディメンション	537
使用状況レポート	545
使用状況レポートを有効にする	545
使用状況レポートのフィールド	548
カスタムレポートを作成する	555
AppStream 2.0 API 呼び出しのログ記録	563
CloudTrail での AppStream 2.0 情報	564
例: AppStream 2.0 ログファイルのエントリ	566
セキュリティ	569
データ保護	570
保管時の暗号化	571
転送時の暗号化	571
管理者のコントロール	572
アプリケーションアクセス	573
Identity and Access Management	575
ネットワークアクセス	576
AppStream 2.0 リソースへのアクセス	576
Application Auto Scaling へのアクセス	592
ホームフォルダおよびアプリケーション設定の永続化用の S3 バケットの削除	593
ストリーミングインスタンスでのアプリケーションとスクリプトへのアクセス	596
SELinux	601
Cookie ベースの認証	601
ログ記録とモニタリング	602

コンプライアンス検証	604
耐障害性	605
インフラストラクチャセキュリティ	605
ネットワークの隔離	606
物理ホストでの分離	607
ネットワークトラフィックの制御	607
インターフェイス VPC エンドポイント	608
FIPS エンドポイント	615
セキュリティグループ	616
更新管理	618
混乱した代理の防止	618
例: AppStream 2.0 サービスロールにおけるサービス間の混乱した代理の防止	620
例: AppStream 2.0 フリートマシンロールにおけるサービス間の混乱した代理の防止	621
例: AppStream 2.0 Elastic Fleets セッションスクリプトの Amazon S3 バケットポリシーに おけるサービス間の混乱した代理の防止	622
例: AppStream 2.0 Application の Amazon S3 バケットポリシーにおけるサービス間の混乱 した代理の防止	624
セキュリティのベストプラクティス	627
永続データの保護	628
エンドポイントのセキュリティとウイルス対策	629
ネットワーク除外	632
AppStream 2.0 セッションの保護	634
ファイアウォールとルーティング	634
データ損失の防止	635
Egress トラフィックの制御	636
AWS サービスの使用	636
トラブルシューティング	639
一般的なトラブルシューティング	639
SAML フェデレーションが機能していません。ユーザーに AppStream 2.0 アプリケーショ ンを表示する権限がありません。	640
ADFS ポータルからのフェデレーション後に、ストリーミングセッションが開始されませ ん。「接続が切断されました」というエラーが表示されます。	640
無効なリダイレクト URI エラーが発生します。	640
自分のイメージビルダーとフリートが [実行中] 状態になることはありません。自分の DNS サーバーは Simple AD ディレクトリにあります。	641

ユーザーのためにアプリケーション設定の永続化を有効にしましたが、永続的なアプリケーション設定が保存またはロードされません。	641
ユーザーに対してアプリケーション設定の永続化を有効にしましたが、特定のストリーミングアプリケーションでは、ユーザーのパスワードがセッション間で永続化されません。	642
Google Chrome データが、ユーザーの永続的なアプリケーション設定を保持する VHD ファイルに保存されます。このため、ユーザーの設定が保持されません。Chrome プロファイルを管理する方法を教えてください。	642
埋め込み AppStream 2.0 ストリーミングセッション用にカスタムドメインを設定しましたが、AppStream 2.0 ストリーミング URL はカスタムドメインにリダイレクトされません。	643
スマートカード対応の AppStream 2.0 フリートでアプリを起動しましたが、認証用に使用できる証明書数が限られています (またはありません)。	643
自分のスマートカード対応の AppStream 2.0 フリートで、Certificate Propagation サービスが開始されません。	645
SAML 認証後に Active Directory のユーザー名またはパスワードでログインできない。	646
Image Builder のトラブルシューティング	646
Image Builder からインターネットに接続できない。	647
アプリケーションをインストールするときに、「このオペレーティングシステムのバージョンはサポートされていません」というエラーが表示されます。	648
自分のアプリケーションを開くために Windows PowerShell スクリプトを使いたいのですが。	648
ユーザーが ClickOnce アプリケーションを利用できるようにしたいのですが。	648
Image Builder に接続するときに、Ctrl+Alt+Delete を入力してログインするように求めるログイン画面が表示されます。しかし、キーストロークがローカルマシンにインターセプトされます。	649
管理者モードおよびテストモードに切り替えるときに、パスワードが要求されます。パスワードを取得する方法がわかりません。	650
インストールされているアプリケーションを追加するとエラーが発生します。	650
Image Builder のバックグラウンドサービスを誤って終了して、切断されました。Image Builder に接続できなくなりました。	650
アプリケーションのテストモードでの起動ができなくなります。	650
VPC で、アプリケーションがネットワークリソースに接続できませんでした。	651
Image Builder デスクトップをカスタマイズしましたが、作成したイメージからフリートを起動してセッションに接続した後、変更内容が使用できません。	651
アプリケーションを起動するときにコマンドラインパラメータがなくなります。	651

ウイルス対策アプリケーションをインストールした後、フリートでイメージを使用できません。	651
イメージ作成に失敗しました。	652
PrewarmManifest.txt へのアクセスが拒否されたというエラーメッセージが表示され、Image Assistant create-image オペレーションが失敗しました	652
フリートのトラブルシューティング	652
フリートの容量を増やそうとしましたが、更新が有効になりません。	653
デフォルト設定の Internet Explorer を使用しないと、アプリケーションが動作しない。Internet Explorer のデフォルト設定を復元する方法を教えてください。	654
フリートインスタンス間で環境変数を永続化する必要がある。	656
ユーザーの Internet Explorer のデフォルトホームページを変更したい。	656
ユーザーがストリーミングセッションを終了してから新しいセッションを開始すると、ストリーミングリソースが利用できないというメッセージが表示される。	656
Active Directory のトラブルシューティング	657
Image Builder とフリートインスタンスが「PENDING」状態でスタックする	657
ユーザーが SAML アプリケーションを使用してログインできない	658
フリートインスタンスが 1 人のユーザーに対しては機能するが、正しくサイクルしない ...	658
ユーザーのグループポリシーオブジェクトが正常に適用されていない	658
AppStream 2.0 ストリーミングインスタンスが Active Directory ドメインに参加していない。	659
ドメインに参加しているストリーミングセッションでユーザーログインが完了するまで時間がかかる	660
ユーザーは、ドメイン参加済みのストリーミングセッションではドメインリソースにアクセスできないが、ドメイン参加済みの Image Builder からはリソースにアクセスできる	660
ユーザーに「証明書ベースの認証が利用できません」というエラーが表示され、ドメインパスワードの入力を求められる。または、証明書ベースの認証が有効になっているセッションを開始すると、「セッションから切断されました」というエラーが表示される	661
Active Directory (AD) サービスアカウントを変更した後、ドメイン結合が失敗しています。	662
AppStream 2.0 ユーザーの問題のトラブルシューティング	663
高度なログ記録を有効にする	663
ユーザーによる AppStream 2.0 クライアントのインストールが失敗し、.NET Framework 4.6 が必要であるというメッセージが表示されます。	665
ユーザーが AppStream 2.0 クライアントをインストールするときに、USB ドライバーのインストールが失敗し、AppStream 2.0 で USB デバイスを使用できなくなります。	666

私の AppStream 2.0 クライアントユーザーは、60 分ごとに AppStream 2.0 セッションから接続が解除されます。	666
ユーザーがローカルデバイスとストリーミングセッションの間でコピーと貼り付けができません。	668
一部のキーボードショートカットは、ストリーミングセッション中に動作しません。	669
ユーザーのドローイングタブレットは、自分がデプロイしたストリーミングアプリケーションで動作していません。	670
ユーザーのストリーミングセッション中に日本語入力方法が機能しない	671
ユーザーがアプリケーションカタログからアプリケーションを起動しようとすると、ストリーミングセッションの最大数到達に関するエラーが表示されます。	673
ユーザーに黒い画面またはデスクトップが表示され、ユーザーのアプリケーションが Elastic フリートで起動されません。エラーは表示されません。	673
永続ストレージ問題のトラブルシューティング	674
スタックのホームフォルダが正常に機能していません。	674
ユーザーがいずれかのアプリケーションからホームフォルダディレクトリにアクセスできない。	675
ユーザーがアプリケーションの 1 つからホームフォルダにアクセスすると、「Device is not ready」というエラーメッセージが表示されます。	675
Amazon S3 ユーザーのホームフォルダにあるファイルを削除したり置き換えたりしましたが、ストリーミングセッション中にフリートインスタンスのホームフォルダの変更がユーザーに表示されません。	676
永続ストレージが予期したとおりに動作しません。ユーザーのファイルは、永続ストレージに保存するのに予想以上に時間がかかっています。	677
私のユーザーは、ファイルを使用していないときにファイルがすでに使用されているというエラーを受け取っています。	678
フォルダに何千ものファイルが含まれている場合、AppStream 2.0 でファイルの一覧を表示するときに時間がかかることがあります。	678
通知コードのトラブルシューティング	678
Active Directory 内部サービス	679
Active Directory ドメイン参加	679
イメージの内部サービス	682
セッションプロビジョニング	683
クォータ	685
AppStream 2.0 ユーザー向けのガイダンス	691
アクセス方法とクライアント	691
ウェブブラウザアクセス	691

Windows 用クライアント	703
macOS 用クライアント	748
ファイルストレージのオプション	760
ホームフォルダを使用する	760
Google ドライブを使用する	762
OneDrive for Business を使用する	765
カスタム共有ネットワークフォルダの使用	769
地域の設定	769
拡張機能 SDK デベロッパーガイド	771
前提条件	771
サードパーティーベンダー拡張機能	772
ドキュメント履歴	773
以前の更新	798
.....	dccci

Amazon AppStream 2.0 とは何ですか？

Amazon AppStream 2.0 は、デスクトップアプリケーションに即座にアクセスできるようにする、完全マネージド型のアプリケーションストリーミングサービスです。AppStream 2.0 は、アプリケーションのホストと実行に必要な AWS リソースを管理し、自動的にスケーリングし、オンデマンドでユーザーにアクセスを提供します。AppStream 2.0 を使用すると、ユーザーは選択したデバイスで必要なアプリケーションにアクセスできます。これは、ネイティブにインストールされたアプリケーションと区別がつかず、素早く反応するユーザーエクスペリエンスを提供します。

AppStream 2.0 を使用すると、既存のデスクトップアプリケーションを簡単に追加し、ユーザーがすぐにストリーミングできるようになります。Windows ユーザーは AppStream 2.0 クライアントまたは HTML5 対応ウェブブラウザをアプリケーションのストリーミングに使用できます。アプリケーションごとに 1 つのバージョンを維持すればよいと、簡単にアプリケーションを管理できます。ユーザーはいつでも最新バージョンのアプリケーションにアクセスできます。アプリケーションは AWS コンピューティングリソースで実行され、データはユーザーのデバイスに保存されることはありません。つまり、ユーザーは常に高性能で安全なエクスペリエンスを得られます。

デスクトップアプリケーションストリーミングの従来のオンプレミスソリューションとは違い、AppStream 2.0 は従量課金制です。そのため、先行投資やインフラストラクチャの維持が不要です。すぐに、グローバルにスケーリングできるため、いつでも最高の体験をユーザーに提供できます。

詳細については、[AppStream 2.0](#) をご覧ください。

トピック

- [Amazon AppStream 2.0 の機能](#)
- [Amazon AppStream 2.0 の主な概念](#)
- [Amazon AppStream 2.0 の開始方法](#)
- [Amazon AppStream 2.0 へのアクセス](#)

Amazon AppStream 2.0 の機能

Amazon AppStream 2.0 を使用すると、次の利点があります。

サポートされているデバイスからデスクトップアプリケーションに安全にアクセス

デスクトップアプリケーションには、Windows および Linux PC、Mac、Chromebook、iPad、Android タブレットから HTML5 対応のウェブブラウザを介して安全にアクセスできます。または、サポートされているバージョンの Windows では、AppStream 2.0 クライアントをアプリケーションのストリーミングに使用できます。

アプリケーションとデータの保護

アプリケーションとデータはオンのまま AWS です。暗号化されたピクセルのみがユーザーにストリーミングされます。アプリケーションは、各ユーザー専用の AppStream 2.0 インスタンスで実行されるため、コンピューティングリソースは共有されません。アプリケーションは、お客様独自の Virtual Private Cloud (VPC) 内で動作し、Amazon VPC のセキュリティ機能を使用してアクセスを制御できます。これによって、アプリケーションを分離し、安全に実行できます。

一貫したスケーラブルなパフォーマンス

AppStream 2.0 は AWS、ローカルデバイスで利用できないコンピューティング機能にアクセスできるで実行されます。つまり、アプリケーションは一貫して高いパフォーマンスで実行されます。ローカルおよびグローバルに即座にスケールでき、ユーザーのために常に低レイテンシーのエクスペリエンスを確保できます。オンプレミスソリューションとは異なり、ユーザーに最も近い AWS リージョンにアプリケーションをすばやくデプロイし、増分設備投資なしでストリーミングを開始できます。

IT 環境との統合

既存の AWS サービスおよびオンプレミス環境と統合します。VPC 内でアプリケーションを実行することで、ユーザーが AWS で保持しているデータや他のリソースにアクセスできるようになります。これにより、AWS とお使いの環境間におけるデータの移動が少なくなり、より高速なユーザーエクスペリエンスが実現されます。

既存の Microsoft Active Directory 環境との統合。これにより、ストリーミングアプリケーションで既存の Active Directory ガバナンス、ユーザーエクスペリエンス、およびセキュリティポリシーを使用できるようになります。

ID フェデレーションを構成します。これにより、ユーザーは社内認証情報を使用してアプリケーションにアクセスできます。または、AppStream 2.0 で実行しているアプリケーションから IT リソースへの認証済みのアクセスを許可することもできます。

お客様のニーズに合うフリートの種類を選択してください。

フリートには 3 つのタイプがあります。

- 常時オン – ストリーミングインスタンスは、アプリケーションとデスクトップをストリーミングしているユーザーがいなくても、常に実行されます。ストリーミングインスタンスは、ユーザーがストリーミングする前にプロビジョニングされる必要があります。プロビジョニングされるストリーミングインスタンスの数は、オートスケーリングルールを使用して管理されます。詳細については、「[the section called “Fleet Auto Scaling”](#)」を参照してください。

ユーザーがアプリケーションまたはデスクトップを選択すると、それらのストリーミングが直ちに開始されます。ストリーミングしているユーザーがいない場合でも、すべてのストリーミングインスタンスに対して実行中のインスタンス料金が請求されます。

- オンデマンド – ストリーミングインスタンスは、ユーザーがアプリケーションとデスクトップをストリーミングしている場合にのみ実行されます。まだユーザーに割り当てられていないストリーミングインスタンスは、停止状態になります。ストリーミングインスタンスは、ユーザーがストリーミングする前にプロビジョニングされる必要があります。プロビジョニングされるストリーミングインスタンスの数は、オートスケーリングルールを使用して管理されます。詳細については、「[the section called “Fleet Auto Scaling”](#)」を参照してください。

ユーザーがアプリケーションまたはデスクトップを選択すると、1~2 分の待機時間後にそれらのストリーミングが開始されます。まだユーザーに割り当てられていないストリーミングインスタンスにはコストが低い停止中のインスタンス料金が請求され、ユーザーに割り当てられているストリーミングインスタンスには実行中のインスタンス料金が請求されます。

- Elastic – このストリーミングインスタンスのプールは、AppStream 2.0 によって管理されます。ユーザーが起動するアプリケーションまたはデスクトップを選択すると、App Block がダウンロードされ、ストリーミングインスタンスにマウントされた後で、これらのストリーミングが開始されます。

Elastic フリートストリーミングインスタンスに対する実行中のインスタンス料金は、ストリーミングセッションの持続時間分のみが秒単位で請求されます。

詳細については、[Amazon AppStream 2.0 の料金](#)を参照してください。

Amazon AppStream 2.0 の主な概念

AppStream 2.0 を最大限に活用するには、以下のコンセプトを理解しておく必要があります。

アプリケーション

アプリケーションには、ユーザーにストリーミングするアプリケーションを起動するために必要な情報が含まれています。アプリケーションは、App Block やイメージなど、アプリケーションを起動するために必要なファイルが含まれるリソースに関連付けられています。

App Block

App Block には、ユーザーにストリーミングするアプリケーションファイルと、アプリケーションを設定するために必要な詳細が含まれています。

App Block Builder

App Block Builder は、App Block の作成に使用される仮想マシンです App Block Builder を起動して接続するには、AppStream 2.0 コンソールを使用します。App Block Builder に接続すると、アプリケーションをインストールできます。App Block Builder は、アプリケーションのコンテンツをパッケージ化し、Amazon S3 バケットにアップロードして、App Block の作成を完了します。

Image Builder

Image Builder は、イメージの作成に使用する仮想マシンです。Image Builder を起動して接続するには、AppStream 2.0 コンソールを使用します。Image Builder に接続すると、アプリケーションをインストール、追加、テストできます。さらに Image Builder を使用してイメージを作成できます。自己所有のプライベートイメージを使用して新しい Image Builder を起動できます。

画像

イメージには、ユーザーにストリーミングできるアプリケーションと、ユーザーがアプリケーションをすばやく使い始めるためのデフォルトのシステムおよびアプリケーション設定が含まれています。は、Image Builder を作成してから独自のアプリケーションを含むイメージを作成するために使用できるベースイメージ AWS を提供します。イメージの作成後にイメージを変更することはできません。他のアプリケーションの追加、既存のアプリケーションの更新、またはイメージ設定の変更を行うには、新しいイメージを作成する必要があります。イメージを他の AWS リージョンにコピーしたり、同じ リージョン内の他の AWS アカウントと共有したりできます。ユーザー、デフォルトのシステム設定とアプリケーション設定により、ユーザーはアプリケーションをすばやく開始できます。

フリート

フリートは、ユーザー指定のアプリケーションとデスクトップが実行されるフリートインスタンス (ストリーミングインスタンスとも呼ばれます) で構成されています。

スタック

スタックは、関連付けられたフリート、ユーザーアクセスポリシー、ストレージ設定で構成されます。ユーザーに対してストリーミングアプリケーションを開始するためにスタックを設定します。

ストリーミングインスタンス

ストリーミングインスタンス (フリートインスタンスとも呼ばれる) は、アプリケーションストリーミングにユーザー 1 人あたりが使用できる EC2 インスタンスです。ユーザーのセッションが完了すると、インスタンスは EC2 によって終了します。

ユーザープール

ユーザープールを使用して、ユーザーと割り当てられたスタックを管理します。

オートスケーリングルール

オートスケーリングルールは、スケジュールベースおよび使用量ベースのポリシーで、常時オンまたはオンデマンドフリートに適用して、ユーザーがストリーミングに利用できるストリーミングインスタンスの数を自動的に管理することができます。

マルチセッション

マルチセッションフリートでは、1 つのフリートインスタンスで複数のユーザーセッションをプロビジョニングできます。基盤となるインフラストラクチャリソースは、すべてのユーザーセッションで共有されます。

Note

マルチセッションは、Windows オペレーティングシステムを搭載した常時オンおよびオンデマンドフリートでのみ利用できます。マルチセッションは Elastic フリートや Linux オペレーティングシステムでは使用できません。

マルチセッションフリートには必ず最新の AppStream 2.0 イメージを使用してください。イメージを最新状態に維持するには、「[the section called “イメージを最新の状態に保つ”](#)」を参照してください。マルチセッションでサポートされるイメージと AppStream 2.0 エージェントバージョンの詳細については、「[the section called “ベースイメージおよびマネージド型イメージの更新版リリースノート”](#)」を参照してください。

Amazon AppStream 2.0 の開始方法

AppStream 2.0 を初めて使用する場合には、Try it Now (今すぐ試す) 機能を使用するか、[Amazon AppStream 2.0 の使用を開始する: サンプルアプリケーションを使用してセットアップする](#) チュートリアルに従って使用できます (両方とも AppStream 2.0 コンソールで使用できます)。

- Try It Now では、デスクトップブラウザから簡単にデスクトップアプリケーションを起動できる無料トライアルの機能が提供されています。
- 入門チュートリアルを使用すると、サンプルアプリケーションまたはユーザー独自のアプリケーションを使用してアプリケーションストリーミングをセットアップできます。サンプルアプリケーションを使用してチュートリアルを開始した場合は、後でユーザー独自のアプリケーションをいつでも追加できます。

これら 2 つのオプションの詳細については、[Amazon AppStream 2.0 FAQs](#) を参照してください。

初めてサービスを使用するときは、AppStream 2.0 が、ユーザーに代わって AppStream 2.0 リソースを作成して管理する [AWS Identity and Access Management \(IAM\)](#) ロールを作成します。

「今すぐ試す」機能を使用するには

1. AppStream 2.0 コンソール (<https://console.aws.amazon.com/appstream2>) を開きます。
2. [今すぐ試す] を選択します。
3. リクエストされた場合は、AWS アカountの認証情報を使用してサインインします。
4. 諸条件を読み、[Agree and Continue] を選択します。
5. 表示されるアプリケーションのリストから、試したいアプリケーションを 1 つ選択します。

入門チュートリアルを実行するには

1. AppStream 2.0 コンソール (<https://console.aws.amazon.com/appstream2>) を開きます。
2. [今すぐ始める] を選択します。
3. AppStream 2.0 リソースの詳細については、オプションを選択します。

Amazon AppStream 2.0 へのアクセス

次のいずれかのインターフェイスを使用して AppStream 2.0 を操作できます。

AWS Management Console

コンソールは AppStream 2.0 リソースを管理するためのブラウザーベースのインターフェイスです。詳細については、「[Amazon AppStream 2.0 の使用を開始する: サンプルアプリケーションを使用してセットアップする](#)」を参照してください。

AWS コマンドラインツール

AWS には、[AWS Command Line Interface](#) (AWS CLI) と の 2 セットのコマンドラインツールが用意されています。[AWS Tools for Windows PowerShell](#)。を使用して AppStream 2.0 コマンドを実行する AWS CLI には、[Amazon AppStream 2.0 コマンドラインリファレンス](#)」を参照してください。

AWS SDKs

さまざまなプログラミング言語から AppStream 2.0 にアクセスできます。SDK は、自動的に次のようなタスクを処理します。

- AppStream 2.0 スタックまたはフリートの設定
- スタックにアプリケーションストリーミング URL を取得する
- リソースについて説明する

詳細については、[Tools for Amazon Web Services](#) を参照してください。

Amazon AppStream 2.0 のセットアップ

Amazon AppStream 2.0 用のセットアップを行うには、以下のタスクを実行します。

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部では、電話またはテキストメッセージを受信し、電話のキーパッドに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 Identity Center、 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 Management Console](#)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 Identity Center ユーザーガイド」の「[AWS IAM Identity Center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 Identity Center 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 Identity Center 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 Identity Center ユーザーガイド」の「[グループの結合](#)」を参照してください。

Amazon AppStream 2.0 の使用を開始する: サンプルアプリケーションを使用してセットアップする

アプリケーションをストリーミングするには、スタックと関連付けられたフリートを含む環境と、1つ以上のアプリケーションイメージが Amazon AppStream 2.0 に必要です。このチュートリアルでは、アプリケーションのストリーミング用にサンプルの AppStream 2.0 環境を設定し、そのストリームへのアクセス権をユーザーに付与する方法について説明します。

Note

AppStream 2.0 の使用方法を学ぶための追加ガイダンスについては、[Amazon AppStream 2.0 入門ガイド](#) を参照してください。このガイドでは、2 つのアプリケーションをインストールして設定し、AppStream 2.0 コンソールを使用して基本的な管理タスクを実行し、提供された AWS CloudFormation テンプレートを使用して Amazon Virtual Private Cloud をプロビジョニングする方法について説明します。

タスク

- [ステップ 1: サンプルスタックをセットアップし、イメージを選択して、フリートを設定する](#)
- [ステップ 2: ユーザーへのアクセスを提供する](#)
- [リソース](#)

ステップ 1: サンプルスタックをセットアップし、イメージを選択して、フリートを設定する

アプリケーションをストリーミングする前に、スタックを設定し、アプリケーションがインストールされているイメージを選択して、フリートを設定します。このステップでは、テンプレートを使用してこれらのタスクを簡素化します。

サンプルスタックをセットアップし、イメージを選択して、フリートを設定するには

1. AppStream 2.0 コンソール (<https://console.aws.amazon.com/appstream2>) を開きます。
2. コンソールに初めてアクセスする場合は、[Get Started] を選択するか、または、左側のナビゲーションメニューから [Quick Links] を選択します。[Set up with sample apps] を選択します。

3. [Step 1: Stack Details] で、スタック名をデフォルトのままにするか、独自の名前を入力します。オプションで、以下の操作を行うことができます。
- Display name (表示名) — スタックの表示名を入力します (最大 100 文字)。
 - Description (説明) — デフォルトの説明をそのまま使用するか、説明文を入力してください (最大 256 文字)。
 - Redirect URL (リダイレクト URL) — エンドユーザーがストリーミングセッションの終了後にリダイレクトされる URL を指定します。
 - フィードバック URL — ユーザーがアプリケーションストリーミングのエクスペリエンスについてフィードバックを送信する [フィードバックの送信] リンクをクリックした後、リダイレクトされる URL を指定します。URL を指定しない場合、このリンクは表示されません。
 - ストリーミングプロトコル環境設定 — スタックで優先するストリーミングプロトコル、UDP または TCP を指定します。現在、UDP は Windows ネイティブクライアントでのみサポートされます。詳細については、「[システム要件と機能のサポート \(AppStream 2.0 クライアント\)](#)」を参照してください。
 - Tags (タグ) — [Add Tag (タグの追加)] を選択し、タグのキーと値を入力します。タグをさらに追加するには、必要に応じてこの手順を繰り返します。詳細については、「[Amazon AppStream 2.0 リソースにタグを付ける](#)」を参照してください。
 - VPC Endpoints (Advanced) (VPC エンドポイント (アドバンスド)) — Virtual Private Cloud (VPC) に、[インターフェイス VPC エンドポイント](#) (インターフェイスエンドポイント) であるプライベートリンクを作成できます。インターフェイスエンドポイントの作成を開始するには、[Create VPC Endpoint (VPC エンドポイントの作成)] を選択します。このリンクを選択すると、VPC コンソールが開きます。エンドポイントの作成を完了するには、[チュートリアル: インターフェイス VPC エンドポイントからの作成とストリーミング](#) の To create an interface endpoint のステップ 3~6 に従います。
- インターフェイスエンドポイントを作成したら、VPC 内でストリーミングトラフィックを維持するために使用できます。
- Embed AppStream 2.0 (AppStream 2.0 の埋め込み) (オプション) — ウェブページに AppStream 2.0 ストリーミングセッションを埋め込むには、埋め込みストリーミングセッションをホストするドメインを指定します。埋め込みストリーミングセッションは、HTTPS (TCP ポート 443) 経由でのみサポートされます。

Note

埋め込み AppStream 2.0 ストリーミングセッションを設定するには、前提条件を満たし、追加のステップを実行する必要があります。詳細については、「[埋め込み Amazon AppStream 2.0 ストリーミングセッション](#)」を参照してください。

4. [Next (次へ)] を選択します。
5. [ステップ 2: イメージを選択] では、すでにサンプルイメージが選択されています。イメージには、評価目的のオープンソースのアプリケーションがプリインストールされています。[Next (次へ)] を選択します。
6. [ステップ 3: フリートを設定] では、指定されたデフォルト値をそのままにします。これらの値のほとんどは、フリートの作成後に変更することができます。
 - Choose instance type (インスタンスタイプの選択) — アプリケーションのパフォーマンス要件に一致するインスタンスタイプを選択します。フリート内のすべてのストリーミングインスタンスが選択したインスタンスタイプで起動します。詳細については、「[AppStream 2.0 インスタンスファミリー](#)」を参照してください。
 - Fleet type (フリートタイプ) — お客様のニーズに合うフリートの種類を選択します。フリートタイプごとに、その即時利用可能性と、支払い方法が異なります。
 - Maximum session duration in minutes (最大セッション期間設定 (分単位)) — ストリーミングセッションがアクティブのままとなることができる最大時間を選択します。この制限に達する 5 分前にユーザーがまだストリーミングインスタンスに接続されている場合は、切断される前に、開いているドキュメントを保存するように求められます。この時間が経過すると、インスタンスが終了され、新しいインスタンスに置き換えられます。
 - Disconnect timeout in minutes (切断タイムアウト (分単位)) — ユーザーが切断した後にストリーミングセッションをアクティブのままにする時間を選択します。切断後、またはこの時間間隔内のネットワークの中断の後、ユーザーがストリーミングインスタンスに再接続を試みる場合、前のセッションに接続されます。それ以外の場合は、新しいインスタンスで新しいセッションに接続されます。スタックをリダイレクト URL が指定されたフリートに関連付けると、ユーザーのストリーミングセッションの終了後に、ユーザーはその URL にリダイレクトされます。

ユーザーがストリーミングセッションツールバーで [End Session] を選択してセッションを終了した場合、切断タイムアウトは適用されません。代わりに、開いているドキュメントを保存するかどうかの確認がユーザーに求められ、その後すぐにストリーミングインスタンスから切断されます。

- Idle disconnect timeout in minutes (アイドル切断タイムアウト (分単位)) — ユーザーがストリーミングセッションから切断されるまでにアイドル状態 (非アクティブ) であることができる時間と、[Disconnect timeout in minutes (切断タイムアウト (分単位))] 期間の開始時刻を選択します。ユーザーは、アイドル状態が原因で切断される前に通知されます。ユーザーが [Disconnect timeout in minutes (切断タイムアウト (分単位))] で指定した期間が経過する前にストリーミングセッションへの再接続を試みると、前のセッションに接続されます。それ以外の場合は、新しいストリーミングインスタンスで新しいセッションに接続されます。この値を 0 に設定すると無効になります。この値を無効にした場合、ユーザーはアイドル状態が原因で切断されることはありません。

 Note

ユーザーがストリーミングセッション中にキーボードまたはマウスの入力を停止した場合、アイドル状態であると見なされます。ファイルのアップロードとダウンロード、オーディオ入力、オーディオ出力、およびピクセルの変更は、ユーザーアクティビティとはなりません。[Idle disconnect timeout in minutes (アイドル切断タイムアウト (分単位))] の期間が経過した後も引き続きアイドル状態である場合、ユーザーは切断されます。

- 複数のユーザーセッション — 1 つのインスタンスで複数のユーザーセッションをプロビジョニングする場合は、このオプションを選択します。デフォルトでは、すべての一意のユーザーセッションはインスタンスによって処理されます (シングルセッション)。

 Note

マルチセッションは、Windows オペレーティングシステムを搭載した常時オンおよびオンデマンドフリートでのみ利用できます。マルチセッションは Elastic フリートや Linux オペレーティングシステムでは使用できません。

マルチセッションフリートには必ず最新の AppStream 2.0 イメージを使用してください。イメージを最新状態に維持するには、「[the section called “イメージを最新の状態に保つ”](#)」を参照してください。マルチセッションでサポートされるイメージと AppStream 2.0 エージェントバージョンの詳細については、「[the section called “ベースイメージおよびマネージド型イメージの最新版リリースノート”](#)」を参照してください。

- インスタンスあたりの最大セッション数 — 1 つのインスタンスあたりのユーザーセッションの最大数。この値は、エンドユーザーのアプリケーションパフォーマンスのニーズに基づいて

選択する必要があります。フリートのインスタンスあたりの最大セッション数は、プロビジョニング後に調整することもできます。その場合、既存のユーザーセッションとインスタンスには影響しませんが、フリートはインスタンスあたりの最大セッション数の新しい値と一致するようになります。2~50 の範囲の値にする必要があります。この値をフリートに設定する前に、「[the section called “マルチセッションに関する推奨事項”](#)」を参照してください。

- **Minimum Capacity (最小容量)** — 予想される同時ユーザーの最小数に基づいて、フリートのインスタンスの最小数を選択します。一意のユーザーセッションはすべてインスタンスによって処理されます。たとえば、需要の少ない時にスタックが 100 人のユーザーを同時にサポートするには、最低容量に 100 を指定します。これにより、100 人以下のユーザーしかいない場合でも、100 のインスタンスが必ず実行されることになります。
- **Maximum Capacity (最大容量)** — 予想される同時ユーザーの最大数に基づいて、フリートのインスタンスの最大数を選択します。一意のユーザーセッションはすべてインスタンスによって処理されます。たとえば、需要の多い時にスタックが 500 人のユーザーを同時にサポートするには、最大容量に 500 を指定します。これにより、最大 500 のインスタンスをオンデマンドで作成できます。
- **フリートの最小ユーザーセッション数** — 予想される最小の同時ユーザー数に基づいて、フリートの最小ユーザーセッション数を選択します。たとえば、需要の少ない時にスタックが 100 人のユーザーを同時にサポートするには、最低容量に 100 を指定します。これにより、ユーザー数が 100 未満でも、100 のユーザーセッションを確保できます。
- **フリートの最大ユーザーセッション数** — 予想される最大の同時ユーザー数に基づいて、フリートの最大ユーザーセッション数を選択します。たとえば、需要の多い時にスタックが 500 人のユーザーを同時にサポートするには、最大容量に 500 を指定します。これにより、最大 500 のユーザーセッションをオンデマンドでプロビジョニングできます。

 Note

シングルセッションフリートの場合、ユーザーセッションごとに 1 つのインスタンスが起動されます。ただし、マルチセッションの場合、実行中のインスタンスの数はインスタンスあたりの最大セッション数によって異なります。容量はユーザーセッション数で指定する必要があります。サービスは、フリートタイプ (マルチセッションまたはシングルセッション) とインスタンスあたりの最大セッション数に基づいて、必要なインスタンス数を決定します。

- **Scaling details (スケーリングの詳細)** — で、特定の条件下で AppStream 2.0 が使用してフリートの容量を調整することができるデフォルトのスケーリングポリシーを指定します。フリート

のサイズは、指定した最小および最大容量によって制限されることに注意してください。詳細については、「[Amazon AppStream 2.0 向け Fleet Auto Scaling](#)」を参照してください。

- IAM ロール (アドバンスド) — アカウントから AppStream 2.0 フリートインスタンスに IAM ロールを適用すると、AWS 認証情報を手動で管理することなく、フリートインスタンスから AWS API リクエストを行うことができます。IAM ロールを適用するには、次のいずれかを行います。
 - Amazon Web Services アカウントで既存の IAM ロールを使用するには、[IAM ロール] の一覧表示から使用するロールを選択します。ロールは、フリートインスタンスからアクセス可能である必要があります。詳細については、「[AppStream 2.0 ストリーミングインスタンスで使用するための既存の IAM ロールの設定](#)」を参照してください。
 - 新しい IAM ロールを作成するには、[Create new IAM role (新しい IAM ロールの作成)] を選択し、[AppStream 2.0 ストリーミングインスタンスで使用する IAM ロールを作成する方法](#)のステップを実行します。

7. [Next (次へ)] を選択します。

8. ステップ 4: ネットワークの設定では、デフォルトの VPC が提供されます。この VPC には、各アベイラビリティゾーンのデフォルトのパブリックサブネットと、VPC にアタッチされたインターネットゲートウェイが含まれます。VPC にはデフォルトのセキュリティグループも含まれます。デフォルトの VPC 設定を使用するには、次の手順を実行します。

- [Default Internet Access (既定のインターネットアクセス)] チェックボックスはオンのままにします。

[Default Internet Access (デフォルトのインターネットアクセス)] を有効にすると、最大 100 のフリートインスタンスがサポートされます。デプロイで 100 を超える同時ユーザーをサポートする必要がある場合は、代わりに[NAT ゲートウェイ設定](#)を使用します。

- VPC の場合は、AWS リージョンで選択したデフォルトの VPC を保持します。

デフォルトの VPC 名では、vpc-*vpc-id* (No_default_value_Name) という形式が使用されます。

- [サブネット 1] および [サブネット 2] では、デフォルトのパブリックサブネットを選択したままにします。

デフォルトのサブネット名は、subnet-*subnet-id* | (*IPv4 CIDR #####*) | Default in *availability-zone* の形式を使用します。

- [セキュリティグループ] では、デフォルトのセキュリティグループを選択したままにします。

デフォルトのセキュリティグループ名は、sg-*security-group-id*-default という形式を使用します。

9. [Step 5: Enable Storage (ステップ 5: ストレージの有効化)] で、以下のうち、1 つ以上を選択し、[Next (次へ)] を選択します。

 Note

Google Drive オプションと OneDrive オプションは現在、マルチセッションフリートでは利用できません。

- Enable Home Folders (ホームフォルダの有効化) — デフォルトでは、この設定は有効になっています。デフォルト設定を維持します。ホームフォルダの有効化要件の詳細については、「[AppStream 2.0 ユーザーのホームフォルダを有効にする](#)」を参照してください。
 - Enable Google Drive (Google Drive の有効化) — 必要に応じて、ユーザーが Google Drive for G Suite のアカウントを AppStream 2.0 にリンクできるように設定することもできます。Google ドライブを有効にできるのは、G Suite ドメイン内のアカウントに対してのみで、個人の Gmail アカウントに対してはできません。Google ドライブの有効化要件の詳細については、「[AppStream 2.0 ユーザーの Google ドライブを有効にする](#)」を参照してください。
 - Enable OneDrive (OneDrive の有効化) — 必要に応じて、ユーザーが OneDrive for Business のアカウントを AppStream 2.0 にリンクできるように設定することもできます。OneDrive は、OneDrive ドメインのアカウントに対してのみ有効にすることができます。個人アカウントに対して設定することはできません。OneDrive の有効化に関する要件の詳細については、[AppStream 2.0 ユーザーの OneDrive を有効にする](#) を参照してください。
10. Step 6: User Settings で、次の設定を構成します。完了したら、[Review (確認)] を選択します。

クリップボード、ファイル転送、ローカルデバイスへのプリント、認証のアクセス許可オプション:

 Note

[ローカルデバイスに出力] と [Active Directory へのスマートカードサインイン] は現在、マルチセッションフリートでは利用できません。

- Clipboard (クリップボード) — デフォルトでは、ユーザーは、ローカルデバイスとストリーミングアプリケーション間でデータをコピーアンドペーストすることができます。クリップボードオプションを制限することで、ユーザーによるデータの貼り付け先をリモートストリーミングセッションに限定したり、データのコピー先をローカルデバイスに限定したりできます。クリップボードオプションを全面的に無効化することもできます。ユーザーはストリーミングセッション時にアプリケーション間でコピーアンドペーストできます。[ローカルデバイス文字数制限にコピー] または [リモートセッション文字数制限に貼り付け] を選択するか、またはその両方を選択して、AppStream 2.0 ストリーミングセッションの内外でクリップボードを使用する際にユーザーがコピーまたは貼り付けることができるデータの量を制限できます。値は 1 ~ 20,971,520 (20 MB) で、指定しない場合はデフォルト値は最大値になります。
- File transfer (ファイル転送) — デフォルトでは、ユーザーは、ローカルデバイスとストリーミングセッション間でファイルをアップロードおよびダウンロードすることができます。ファイル転送オプションを制限することで、ユーザーによるファイルのアップロード先をストリーミングセッションに限定したり、ファイルのダウンロード先をローカルデバイスに限定したりできます。ファイル転送を全面的に無効化することもできます。

 Important

ユーザーがストリーミングセッション中にローカルドライブとフォルダにアクセスするために AppStream 2.0 ファイルシステムリダイレクトが必要な場合は、ファイルのアップロードとダウンロードの両方を有効にする必要があります。ファイルシステムリダイレクトを使用するには、ユーザーには、AppStream 2.0 クライアントバージョン 1.0.480 以降がインストールされている必要があります。詳細については、「[AppStream 2.0 ユーザーのファイルシステムリダイレクトを有効にする](#)」を参照してください。

- Print to local device (ローカルデバイスへの出力) — デフォルトでは、ユーザーは、ストリーミングアプリケーションからローカルデバイスに出力することができます。アプリケーションの [Print (出力)] を選択すると、.pdf ファイルをダウンロードして、ローカルプリンタに出力することができます。ユーザーがローカルデバイスに出力できないようにするには、このオプションを無効にします。
- Active Directory のパスワードサインイン — ユーザーは、Active Directory ドメインに参加している AppStream 2.0 ストリーミングインスタンスにサインインするために、Active Directory ドメインパスワードを入力できます。

Active Directory のスマートカードサインインを有効にすることもできます。少なくとも 1 つの認証方法を有効にしてください。

- Smart card sign in for Active Directory (Active Directory へのスマートカードサインイン) — ユーザーはローカルコンピュータに接続されたスマートカードリーダーとスマートカードを使用して、Active Directory ドメインに参加している AppStream 2.0 ストリーミングインスタンスにサインインできます。

[Password sign in for Active Directory (Active Directory のパスワードサインイン)] を有効にすることもできます。少なくとも 1 つの認証方法を有効にしてください。

 Note

クリップボード、ファイル転送、およびローカルデバイスへのプリント — これらの設定は、ユーザーが AppStream 2.0 のデータ転送機能を使用できるかどうかのみを制御します。イメージを使用してブラウザ、ネットワークプリンタ、またはその他のリモートリソースにアクセスできる場合、ユーザーは別の方法を使用してストリーミングセッション間でデータを転送することができます。

認証設定 — これらの設定は、Windows の AppStream 2.0 ストリーミングインスタンス (フリートまたは Image Builder) へのサインインに使用できる認証方法のみを制御します。ユーザーは、ストリーミングインスタンスにサインインした後、セッション内認証に使用できる認証方法を制御しません。Windows サインインおよびセッション中の認証にスマートカードを使用するための構成要件については、[スマートカード](#) を参照してください。

アプリケーション設定の永続化オプション:

- Enable Application Settings Persistence (アプリケーション設定の永続化を有効にする) — ユーザーによるアプリケーションのカスタマイズや Windows 設定は各ストリーミングセッション後に自動的に保存され、次のセッションで適用されます。これらの設定は、アプリケーション設定の永続化が有効になっている AWS リージョン内のアカウントの Amazon Simple Storage Service (Amazon S3) バケットに保存されます。
- Settings Group (設定グループ) — 設定グループは、このスタックに保存されているどのアプリケーション設定をストリーミングセッションで使用するかを決定します。同じ設定グループを別のスタックに適用すると、両方のスタックで同じアプリケーション設定が使用されます。デフォルトでは、設定グループ値はスタックの名前です。

Note

アプリケーション設定の永続化を有効にして管理するための要件については、
「[AppStream 2.0 ユーザーのアプリケーション設定の永続化を有効にする](#)」を参照してください。

11. [Step 7: Review (ステップ 7: 確認)] で、スタックの詳細を確認します。任意のセクションの設定を変更するには、[Edit] を選択し、必要な変更を加えます。設定内容を確認したら、[作成] を選択します。
12. 料金確認ダイアログボックスで、確認チェックボックスをオンにし、[作成] を選択します。
13. サービスがリソースをセットアップした後、[Stacks] ページが表示されます。新しいスタックが使用できる状態になったら、ステータスに [Active] と表示されます。

ステップ 2: ユーザーへのアクセスを提供する

関連付けられたフリートを使用してスタックを作成した後は、AppStream 2.0 ユーザープールの SAML 2.0 [single sign-on (SSO)]、または AppStream 2.0 API を通じてユーザーにアクセス権を付与できます。詳細については、[Amazon AppStream 2.0 でのユーザープール管理](#)および[Amazon AppStream 2.0 と SAML 2.0 の統合](#)を参照してください。

Note

AppStream 2.0 ユーザープール内のユーザーは、Active Directory ドメインに参加しているフリートを持つスタックに割り当ててはできません。

この入門演習では、AppStream 2.0 ユーザープールを使用できます。このアクセスメソッドにより、永続的なログインポータル URL を使用してユーザーを作成および管理できます。ユーザーをセットアップせずにすばやくアプリケーションのストリーミングをテストするには、次のステップを実行して、ストリーミング URL の別名でも知られている、一時的な URL を作成します。

一時的な URL によりユーザーにアクセス権を付与するには

1. ナビゲーションペインの [Fleets] を選択します。
2. フリートのリストで、ストリーミング URL を作成するスタックに関連付けられているフリートを選択します。フリートのステータスが [Running (実行中)] であることを確認します。

3. ナビゲーションペインで、[Stacks] を選択します。スタックを選択してから、[アクション]、[Create Streaming URL (ストリーミング URL の作成)] を選択します。
4. [User id] にユーザー ID を入力します。有効期限を選択し、生成された URL が有効な期間の長さを指定します。
5. ユーザー ID と URL を表示するには、[Get URL] を選択します。
6. リンクをクリップボードにコピーするには、[Copy Link] を選択します。

AppStream 2.0 へのアクセスをユーザーに提供すると、ユーザーは AppStream 2.0 ストリーミングセッションを開始できます。AppStream 2.0 ユーザープールを介してアクセスを提供する場合、ストリーミングセッションにウェブブラウザを使用する必要があります。

SAML 2.0 [single sign-on (SSO)] または AppStream 2.0 API を使用してユーザーにアクセスを提供する予定の場合は、AppStream 2.0 クライアントを使用できるようにすることができます。AppStream 2.0 クライアントは、AppStream 2.0 ストリーミングセッション中に追加機能を必要とするユーザー用に設計されたネイティブアプリケーションです。詳細については、「[AppStream 2.0 クライアント経由でアクセスを提供する](#)」を参照してください。

リソース

詳細については次を参照してください:

- AppStream 2.0 Image Builder を使用して独自のアプリケーションを追加し、ユーザーにストリーミングできるイメージを作成する方法について説明します。詳細については、「[チュートリアル: AppStream 2.0 コンソールを使用してカスタム AppStream 2.0 イメージを作成する](#)」を参照してください。
- AppStream 2.0 ホームフォルダ、Google ドライブ、および OneDriveを使用して、セッションユーザーに永続的ストレージを提供します。詳細については、「[AppStream 2.0 ユーザーに対して永続的ストレージを有効にして管理する](#)」を参照してください。
- AppStream 2.0 ストリーミングリソースを Microsoft Active Directory 環境に統合します。詳細については、「[AppStream 2.0 でのアクティブディレクトリの使用](#)」を参照してください。
- AppStream 2.0 ストリーミングインスタンスにだれがアクセスできるかを制御します。詳細については、「[Amazon AppStream 2.0 向け Identity and Access Management](#)」、「[Amazon AppStream 2.0 ユーザープール](#)」、および「[Amazon AppStream 2.0 と SAML 2.0 の統合](#)」を参照してください。
- Amazon CloudWatch を使用して AppStream 2.0 リソースを監視します。詳細については、「[AppStream 2.0 のメトリクスとディメンション](#)」を参照してください。

- AppStream 2.0 ストリーミングエクスペリエンスのトラブルシューティングを行います。詳細については、「[トラブルシューティング](#)」を参照してください。

Amazon AppStream 2.0 のネットワーキングとアクセス

以下のトピックでは、ユーザーの AppStream 2.0 ストリーミングインスタンス (フリートインスタンス) への接続を有効にし、AppStream 2.0 フリート、Image Builder、および App Block Builder がネットワークリソースインターネットにアクセスするための情報を提供します。

内容

- [インターネットアクセス](#)
- [AppStream 2.0 用の VPC の設定](#)
- [AppStream 2.0 機能に対する Amazon S3 VPC エンドポイントの使用](#)
- [VPC への Amazon AppStream 2.0 の接続](#)
- [Amazon AppStream 2.0 へのユーザー接続](#)

インターネットアクセス

フリート、App Block Builder および Image Builder でインターネットアクセスが必要な場合は、いくつかの方法を使ってデフォルトのインターネットアクセス機能を有効にすることができます。インターネットアクセスを有効にする方法を選択するときは、デプロイでサポートする必要があるユーザーの数とデプロイの目標を考慮してください。次に例を示します:

- デプロイで 100 を超える同時実行ユーザーをサポートする必要がある場合は、[プライベートサブネットと NAT ゲートウェイを使用して VPC を設定](#)します。
- デプロイでサポートされる同時実行ユーザー数が 100 未満の場合は、[パブリックサブネットを使用して新規または既存の VPC を設定](#)できます。
- デプロイでサポートされる同時実行ユーザー数は 100 未満で、AppStream 2.0 を初めて使用し、サービスの使用を開始する場合は、[デフォルトの VPC、パブリックサブネット、セキュリティグループを使用](#)できます。

以下のセクションでは、これらの各デプロイオプションについて詳しく説明します。

- [プライベートサブネットの VPC および NAT ゲートウェイを設定する](#) (推奨) — この設定では、プライベートサブネットでフリート、App Block Builder、および Image Builder を起動し、VPC のパブリックサブネットで NAT ゲートウェイを設定します。ストリーミングインスタンスには、インターネットから直接アクセスできないプライベート IP アドレスが割り当てられます。

また、インターネットアクセスを有効にするために、[Default Internet Access (デフォルトのインターネットアクセス)] オプションを使用する設定とは異なり、NAT 設定は 100 フリートインスタンスに制限されません。デプロイで 100 を超える同時ユーザーをサポートする必要がある場合は、この設定を使用します。

NAT ゲートウェイで使用する新しい VPC を作成して設定することも、既存の VPC に NAT ゲートウェイを追加することもできます。

- [パブリックサブネットを使用して新しい VPC または既存の VPC を設定する](#) — この設定では、フリート、App Block Builder、および Image Builder をパブリックサブネットで起動し、[デフォルトのインターネットアクセス] 有効にします。このオプションを有効にすると、AppStream 2.0 は Amazon VPC パブリックサブネットのインターネットゲートウェイを使用してインターネット接続を提供します。ストリーミングインスタンスには、インターネットから直接アクセスできるパブリック IP アドレスが割り当てられます。この目的のために、新しい VPC を作成するか、既存の VPC を設定できます。

 Note

[Default Internet Access (デフォルトのインターネットアクセス)] を有効にすると、最大 100 のフリートインスタンスがサポートされます。デプロイで 100 を超える同時ユーザーをサポートする必要がある場合は、代わりに[NAT ゲートウェイ設定](#)を使用します。

- [デフォルト VPC、パブリックサブネット、およびセキュリティグループの使用](#) — AppStream 2.0 を初めて使用し、サービスの使用を開始する場合は、フリート、App Block Builder、および Image Builder をデフォルトのパブリックサブネットで起動して、[既定のインターネットアクセス] を有効にすることができます。このオプションを有効にすると、AppStream 2.0 は Amazon VPC パブリックサブネットのインターネットゲートウェイを使用してインターネット接続を提供します。ストリーミングインスタンスには、インターネットから直接アクセスできるパブリック IP アドレスが割り当てられます。

デフォルトの VPC は、2013 年 12 月 4 日以降に作成された Amazon Web Services アカウントで使用できます。

デフォルト VPC には、各アベイラビリティゾーンのデフォルトのパブリックサブネットと、VPC にアタッチされたインターネットゲートウェイが含まれます。VPC にはデフォルトのセキュリティグループも含まれます。

Note

[Default Internet Access (デフォルトのインターネットアクセス)] を有効にすると、最大 100 のフリートインスタンスがサポートされます。デプロイで 100 を超える同時ユーザーをサポートする必要がある場合は、代わりに [NAT ゲートウェイ設定](#) を使用します。

AppStream 2.0 用の VPC の設定

AppStream 2.0 を設定するときに、フリートインスタンスと Image Builder を起動する Virtual Private Cloud (VPC) と、少なくとも 1 つのサブネットを指定する必要があります。VPC は、Amazon Web Services クラウド内の論理的に分離された領域にある仮想ネットワークです。サブネットは、VPC の IP アドレスの範囲です。

VPC を AppStream 2.0 に設定する場合、パブリックサブネットまたはプライベートサブネットのいずれか、または両方のタイプのサブネットを組み合わせて指定できます。パブリックサブネットは、インターネットゲートウェイを介してインターネットに直接アクセスできます。インターネットゲートウェイへのルートを持たないプライベートサブネットには、インターネットへのアクセスを提供するためにネットワークアドレス変換 (NAT) ゲートウェイまたは NAT インスタンスが必要です。

目次

- [VPC セットアップの推奨事項](#)
- [プライベートサブネットの VPC および NAT ゲートウェイを設定する](#)
- [パブリックサブネットを使用して新しい VPC または既存の VPC を設定する](#)
- [デフォルト VPC、パブリックサブネット、およびセキュリティグループの使用](#)

VPC セットアップの推奨事項

フリートを作成する場合、Image Builder または App Block Builder を起動する場合は、VPC と使用する 1 つ以上のサブネットを指定します。セキュリティグループを指定することで、VPC に対する追加のアクセスコントロールを提供できます。

以下の推奨事項は、VPC をより効果的かつ安全に設定するのに役立ちます。また、効果的なフリートスケーリングをサポートする環境の設定にも役立ちます。効果的なフリートスケーリングにより、不必要なリソースの使用と関連コストを回避しながら、現在および予想される AppStream 2.0 ユーザーの需要に対応できます。

VPC 全体の設定

- VPC 設定が、フリートスケーリングのニーズをサポートできることを確認します。

フリートスケーリングの計画を作成する際には、1 人のユーザーがフリートインスタンスを 1 つ必要とすることに注意してください。したがって、フリートのサイズによって、同時にストリーミングできるユーザーの数が決まります。このため、使用する [インスタンスタイプ](#) ごとに、VPC がサポートできるフリートインスタンスの数が、同じインスタンスタイプで予想される同時ユーザー数よりも多いことを確認します。

- AppStream 2.0 アカウントのクォータ (制限とも呼ばれる) が、予想される需要に対応するのに十分であることを確認します。クォータの引き上げをリクエストするには、<https://console.aws.amazon.com/servicequotas/> の [Service Quotas] コンソールを使用します。デフォルトの AppStream 2.0 のクォータについては、「[Amazon AppStream 2.0 Service Quotas](#)」を参照してください。
- ストリーミングインスタンス (フリートインスタンス、App Block Builder、または Image Builder) にインターネットへのアクセスを提供する場合は、ストリーミングインスタンス用の 2 つのプライベートサブネットと、パブリックサブネットの NAT ゲートウェイを持つ VPC を設定することをお勧めします。

NAT ゲートウェイを使用すると、プライベートサブネット内のストリーミングインスタンスをインターネットやその他の AWS サービスに接続できます。ただし、インターネットはこれらのインスタンスとの接続を開始できません。また、インターネットアクセスを有効にするために [Default Internet Access (デフォルトのインターネットアクセス)] オプションを使用する設定とは異なり、NAT 設定では 100 以上のフリートインスタンスがサポートされます。詳細については、「[プライベートサブネットの VPC および NAT ゲートウェイを設定する](#)」を参照してください。

弾性ネットワークインターフェイス

- AppStream 2.0 はフリートの希望する容量で最大数の [Elastic Network Interface](#) (ネットワークインターフェイス) を作成します。デフォルトでは、リージョンごとのネットワークインターフェイスの制限は 5000 です。

何千ものストリーミングインスタンスなど、非常に大規模なデプロイの容量を計画する場合は、同じリージョンでも使用される EC2 インスタンスの数を考慮してください。

サブネット

- VPC に複数のプライベートサブネットを設定する場合は、それぞれを異なるアベイラビリティゾーンで設定します。これにより、耐障害性が向上し、容量不足エラーを防ぐことができます。同じ AZ で 2 つのサブネットを使用する場合、AppStream 2.0 は 2 つ目のサブネットを使用しないため、IP アドレスが不足する可能性があります。
- アプリケーションに必要なネットワークリソースが、両方のプライベートサブネットを通じてアクセスできることを確認します。
- 各プライベートサブネットに、予想される同時ユーザーの最大数を考慮するのに十分な数のクライアント IP アドレスを許可するサブネットマスクを設定します。また、予想される増加に対応するために、追加の IP アドレスを許可します。詳細については、[VPC and Subnet Sizing for IPv4](#) を参照してください。
- NAT で VPC を使用している場合は、インターネットアクセス用の NAT ゲートウェイを持つパブリックサブネットを少なくとも 1 つ、できれば 2 つ設定します。プライベートサブネットが存在する同じアベイラビリティゾーンにパブリックサブネットを設定します。

大規模な AppStream 2.0 フリートデプロイで耐障害性を強化し、容量不足エラーの可能性を減らすために、VPC 設定を 3 番目のアベイラビリティゾーンに拡張することを検討してください。この追加のアベイラビリティゾーンに、プライベートサブネット、パブリックサブネット、および NAT ゲートウェイを含めます。

セキュリティグループ

- セキュリティグループを使用して、VPC への追加のアクセスコントロールを提供します。
- VPC に属するセキュリティグループを使用すると、AppStream 2.0 ストリーミングインスタンスとアプリケーションに必要なネットワークリソース間のネットワークトラフィックを制御できます。これらのリソースには、Amazon RDS や Amazon FSx、ライセンスサーバー、データベースサーバー、ファイルサーバー、アプリケーションサーバーなどの他の AWS サービスが含まれる場合があります。
- アプリケーションに必要なネットワークリソースへのアクセスが、セキュリティグループで許可されていることを確認してください。

AppStream 2.0 セキュリティグループの詳しい設定については、[Amazon AppStream 2.0 のセキュリティグループ](#) を参照してください。セキュリティグループの一般的な情報については、Amazon VPC ユーザーガイドの [Security Groups for Your VPC](#) を参照してください。

プライベートサブネットの VPC および NAT ゲートウェイを設定する

ストリーミングインスタンス (フリートインスタンス、App Block Builder、および Image Builder) にインターネットへのアクセスを提供する場合は、ストリーミングインスタンス用の 2 つのプライベートサブネットと、パブリックサブネットの NAT ゲートウェイを持つ VPC を設定することをお勧めします。NAT ゲートウェイで使用する新しい VPC を作成して設定することも、既存の VPC に NAT ゲートウェイを追加することもできます。VPC 設定のその他の推奨事項については、[VPC セットアップの推奨事項](#) を参照してください。

NAT ゲートウェイは、プライベートサブネット内のストリーミングインスタンスがインターネットまたは他の AWS サービスに接続できるようにしますが、インターネットがそれらのインスタンスとの接続を開始できないようにします。また、AppStream 2.0 ストリーミングインスタンスのインターネットアクセスを有効にするために [Default Internet Access (デフォルトのインターネットアクセス)] オプションを使用する設定とは異なり、この設定は 100 フリートインスタンスに制限されません。

NAT ゲートウェイと本設定の使用については、Amazon VPC ユーザーガイドの [NAT Gateways](#) と [VPC with Public and Private Subnets \(NAT\)](#) を参照してください。

目次

- [新しい VPC の作成と設定](#)
- [既存の VPC に NAT ゲートウェイを追加する](#)
- [Amazon AppStream 2.0 で、フリート、Image Builder、または App Block Builder のインターネットアクセスを有効にする](#)

新しい VPC の作成と設定

このトピックでは、VPC ウィザードを使用して、パブリックサブネットと 1 つのプライベートサブネットを持つ VPC を作成する方法について説明します。このプロセスの一環として、ウィザードはインターネットゲートウェイと NAT ゲートウェイを作成します。また、パブリックサブネットに関連付けられたカスタムルートテーブルを作成し、プライベートサブネットに関連付けられたメインルートテーブルを更新します。NAT ゲートウェイは、VPC のパブリックサブネットで自動的に作成されます。

ウィザードを使用して初期 VPC 設定を作成したら、2 つ目のプライベートサブネットを追加します。この設定の詳細については、Amazon VPC ユーザーガイドの [VPC with Public and Private Subnets \(NAT\)](#) を参照してください。

Note

すでに VPC がある場合は、代わりに、[既存の VPC に NAT ゲートウェイを追加する](#) のステップを実行します。

目次

- [ステップ 1: Elastic IP アドレスの割り当て](#)
- [ステップ 2: 新しい VPC を作成する](#)
- [ステップ 3: 2 番目のプライベートサブネットの追加](#)
- [ステップ 4: サブネットルートテーブルの検証と名前付け](#)

ステップ 1: Elastic IP アドレスの割り当て

VPC を作成する前に、AppStream 2.0 リージョンに Elastic IP アドレスを割り当てる必要があります。最初に VPC で使用する Elastic IP アドレスを割り当てて、NAT ゲートウェイに関連付ける必要があります。詳細については、Amazon VPC ユーザーガイドの [Elastic IP Addresses](#) を参照してください。

Note

使用する Elastic IP アドレスには料金が適用される場合があります。詳しい情報については、Amazon EC2 の料金ページの [Elastic IP Addresses](#) を参照してください。

Elastic IP アドレスをまだ持っていない場合は、以下のステップを実行します。既存の Elastic IP アドレスを使用する場合は、そのアドレスが別のインスタンスやネットワークインターフェイスに現在関連付けられていないことを確認します。

Elastic IP アドレスを割り当てるには

1. Amazon EC2 コンソールの <https://console.aws.amazon.com/ec2/> を開いてください。
2. ナビゲーションペインの [Network & Security] で、[Elastic IPs] を選択します。
3. [Allocate New Address (新しいアドレスの割り当て)] を選択し、続いて [Allocate (割り当て)] を選択します。
4. Elastic IP アドレスを書き留めます。

5. [Elastic IP] ペインの右上にある [X] アイコンをクリックしてペインを閉じます。

ステップ 2: 新しい VPC を作成する

パブリックサブネットと 1 つのプライベートサブネットを持つ新しい VPC を作成するには、次のステップを実行します。

新しい VPC を作成するには

1. Amazon VPC コンソールの <https://console.aws.amazon.com/vpc/> を開いてください。
2. ナビゲーションペインで、[VPC ダッシュボード] を選択します。
3. Launch VPC Wizard (VPC ウィザードの起動) を選択します。
4. [Step 1: Select a VPC Configuration (ステップ 1: VPC 設定を選択する)] ページで [VPC with Public and Private Subnets (パブリックサブネットとプライベートサブネットを持つ VPC)] を選択し、[Select (選択)] を選択します。
5. [Step 2: VPC with Public and Private Subnets (ステップ 2: パブリックサブネットとプライベートサブネットを持つ VPC)] で、VPC を次のように設定します。
 - [IPv4 CIDR block (IPv4 CIDR ブロック)] では、VPC 用の IPv4 CIDR ブロックを指定します。
 - [IPv6 CIDR ブロック] は、デフォルト値の、[No IPv6 CIDR Block (IPv6 CIDR ブロックなし)] のままにしておきます。
 - [VPC name (VPC 名)] にキーの一意の名前を入力します。
6. パブリックサブネットを次のように設定します。
 - [Public subnet's IPv4 CIDR (パブリックサブネットの IPv4 CIDR)] に、サブネットの CIDR ブロックを指定します。
 - [Availability Zone (アベイラビリティーゾーン)] では、デフォルト値の、[No Preference (指定なし)] のままにしておきます。
 - [Public subnet name (パブリックサブネット名)] に、サブネットの名前を入力します (例: AppStream2 Public Subnet)。
7. 最初のプライベートサブネットを次のように設定します。
 - [Private subnet's IPv4 CIDR (プライベートサブネットの IPv4 CIDR)] に、サブネットの CIDR ブロックを入力します。指定した値を書き留めておきます。
 - [Availability Zone (アベイラビリティーゾーン)] で、特定のゾーンを選択し、選択したゾーンを書き留めます。

- [Private subnet name (プライベートサブネット名)] に、サブネットの名前を入力します (例: AppStream2 Private Subnet1)。
 - 残りのフィールドについては、該当する場合は、デフォルト値をそのまま使用します。
8. [Elastic IP Allocation ID (Elastic IP 割り当て ID)] で、テキストボックスをクリックし、作成した Elastic IP アドレスに対応する値を選択します。このアドレスは NAT ゲートウェイに割り当てられます。Elastic IP アドレスがない場合は、<https://console.aws.amazon.com/vpc/> の Amazon VPC コンソールを使用して作成します。
 9. [Service endpoints (サービスエンドポイント)] で、環境に Amazon S3 エンドポイントが必要な場合は、エンドポイントを指定します。S3 エンドポイントは、ユーザーに[ホームフォルダ](#)へのアクセスを提供したり、プライベートネットワークのユーザーに対して[アプリケーション設定の永続性](#)を有効にしたりするために必要です。

Amazon S3 エンドポイントを指定するには、次の手順を実行します。

- a. [Add Endpoint (エンドポイントの追加)] を選択します。
 - b. [Service (サービス)] で、末尾が「s3」 (VPC が作成されるリージョンに対応する `com.amazonaws.region.s3` エントリ) で終わるエントリをリストから選択します。
 - c. [Subnet (サブネット)] で、[Private subnet (プライベートサブネット)] を選択します。
 - d. [Policy (ポリシー)] では、既定値の [Full Access (フルアクセス)] のままにします。
10. [Enable DNS hostnames (DNS ホスト名を有効にする)] では、デフォルト値の [Yes (はい)] のままにします。
 11. [Hardware tenancy (ハードウェアテナンシー)] では、デフォルト値の [Default (デフォルト)] のままにします。
 12. [Create VPC] を選択します。
 13. VPC の設定には数分かかることに注意してください。VPC が作成されたら、[OK] を選択します。

ステップ 3: 2 番目のプライベートサブネットの追加

前のステップ ([ステップ 2: 新しい VPC を作成する](#)) で、1 つのパブリックサブネットと 1 つのプライベートサブネットを持つ VPC を作成しました。2 つ目のプライベートサブネットを追加するには、以下のステップを実行します。1 つ目のプライベートサブネットとは異なるアベイラビリティゾーンに 2 つ目のプライベートサブネットを追加することをお勧めします。

1. ナビゲーションペインで、[サブネット] を選択してください。

2. 前のステップで作成した最初のプライベートサブネットを選択します。サブネットのリストの下にある [Description (説明)] タブで、このサブネットのアベイラビリティゾーンを書き留めます。
3. サブネットペインの左上にある [Create Subnet (サブネットの作成)] を選択します。
4. [Name tag (名前タグ)] に、プライベートサブネットの名前を入力します (例: AppStream2 Private Subnet2)。
5. [VPC] では、前のステップで作成した VPC を選択します。
6. [Availability Zone (アベイラビリティゾーン)] で、最初のプライベートサブネットに使用しているアベイラビリティゾーン以外のアベイラビリティゾーンを選択します。別のアベイラビリティゾーンを選択すると、耐障害性が向上し、容量不足エラーを防ぐのに役立ちます。
7. [IPv4 CIDR block (IPv4 CIDR ブロック)] の場合は、新しいサブネットの一意的 CIDR ブロック範囲を指定します。たとえば、最初のプライベートサブネットの IPv4 CIDR ブロック範囲が 10.0.1.0/24 である場合、新しいプライベートサブネットに 10.0.2.0/24 CIDR ブロック範囲を指定できます。
8. [作成] を選択します。
9. サブネットが作成されたら、[Close (閉じる)] を選択します。

ステップ 4: サブネットルートテーブルの検証と名前付け

VPC を作成して設定したら、以下のステップを実行してルートテーブルの名前を指定し、そのことを確認します。

- NAT ゲートウェイが存在するサブネットに関連付けられたルートテーブルには、インターネットゲートウェイへのインターネットトラフィックを指すルートが含まれます。これにより、NAT ゲートウェイがインターネットにアクセスできるようになります。
 - プライベートサブネットに関連付けられたルートテーブルは、インターネットトラフィックを NAT ゲートウェイに向けるように設定されます。これにより、プライベートサブネットのストリーミングインスタンスがインターネットと通信できるようになります。
1. ナビゲーションペインで [Subnets (サブネット)] を選択し、作成したパブリックサブネットを選択します (例: AppStream 2.0 Public Subnet)。
 - a. [Route Table (ルートテーブル)] タブで、ルートテーブルの ID を選択します (たとえば、rtb-12345678)。

- b. ルートテーブルを選択します。[名前] の下で編集アイコン (鉛筆) を選択し、名前 (例: appstream2-public-routetable) を入力してから、チェックマークを選択して名前を保存します。
- c. パブリックルートテーブルを選択したまま、[ルート] タブで、ローカルトラフィック用に 1 つのルートが存在し、他のすべてのトラフィックをインターネットゲートウェイに送信する VPC 用の別のルートがあることを確認します。以下のテーブルでは、これらの 2 つのルートについて説明しています。

送信先	ターゲット	説明
パブリックサブネット IPv4 CIDR ブロック (10.0.0/20 など)	ローカル	パブリックサブネット IPv4 CIDR ブロック内の IPv4 アドレス宛てのリソースからのトラフィックはすべて、VPC 内でローカルにルーティングされます。
その他のすべての IPv4 アドレス宛てのトラフィック (0.0.0.0/0 など)	アウトバウンド (igw- ID)	その他すべての IPv4 アドレス宛てのトラフィックは、VPC ウィザードで作成されたインターネットゲートウェイ (igw- ID で識別) にルーティングされます。

2. ナビゲーションペインで [サブネット] を選択し、作成した最初のプライベートサブネットを選択します (例: AppStream2 Private Subnet1)。
 - a. [ルートテーブル] タブで、ルートテーブルの ID を選択します。
 - b. ルートテーブルを選択します。[名前] の下で編集アイコン (鉛筆) を選択し、名前 (例: appstream2-private-routetable) を入力してから、チェックマークを選択して名前を保存します。
 - c. [Routes (ルート)] タブで、ルートテーブルに次のルートが含まれていることを確認します。

送信先	ターゲット	説明
パブリックサブネット IPv4 CIDR ブロック (10.0.0/20 など)	ローカル	パブリックサブネット IPv4 CIDR ブロック内の IPv4 アドレス宛てのリソースからのトラフィックはすべて

送信先	ターゲット	説明
		て、VPC 内でローカルにルーティングされます。
その他のすべての IPv4 アドレス宛てのトラフィック (0.0.0.0/0 など)	アウトバウンド (nat- ID)	その他すべての IPv4 アドレス宛てのトラフィックは、NAT ゲートウェイ (nat- ID で識別) にルーティングされます。
S3 バケット宛てのトラフィック (S3 エンドポイントを指定した場合に適用)	ストレージ (vpce- ID)	S3 バケット宛てのトラフィックは、S3 エンドポイント (vpce- ID で識別) にルーティングされます。
[p1- ID (com.amazons. #####.s3)]		

- ナビゲーションペインで [サブネット] を選択し、作成した 2 番目のプライベートサブネットを選択します (例: AppStream2 Private Subnet2)。
- [ルートテーブル] タブで、ルートテーブルがプライベートルートテーブルであることを確認します (例: appstream2-private-routetable)。ルートテーブルが異なる場合は、[編集] を選択してこのルートテーブルを選択します。

次のステップ

フリートインスタンス、App Block Builder、および Image Builder がインターネットにアクセスできるようにするには、[Amazon AppStream 2.0 で、フリート、Image Builder、または App Block Builder のインターネットアクセスを有効にする](#)のステップを完了します。

既存の VPC に NAT ゲートウェイを追加する

すでに VPC を設定している場合は、次のステップを実行して NAT ゲートウェイを VPC に追加します。新しい VPC を作成する必要がある場合は、「[新しい VPC の作成と設定](#)」を参照してください。

既存の VPC に NAT ゲートウェイを追加するには

1. NAT ゲートウェイを作成するには、Amazon VPC ユーザーガイドの [Creating a NAT Gateway](#) の手順を完了します。
2. VPC に少なくとも 1 つのプライベートサブネットがあることを確認します。高可用性と耐障害性のために異なるアベイラビリティゾーンから 2 つのプライベートサブネットを指定することをお勧めします。2 番目のプライベートサブネットを作成する方法については、「[ステップ 3: 2 番目のプライベートサブネットの追加](#)」を参照してください。
3. 1 つ以上のプライベートサブネットに関連付けられたルートテーブルを更新して、インターネットバウンドトラフィックを NAT ゲートウェイに向かわせます。これにより、プライベートサブネットのストリーミングインスタンスがインターネットと通信できるようになります。そのためには、「[ルートテーブルの設定](#)」のステップを完了します。

次のステップ

フリートインスタンス、App Block Builder、および Image Builder がインターネットにアクセスできるようにするには、[Amazon AppStream 2.0 で、フリート、Image Builder、または App Block Builder のインターネットアクセスを有効にする](#)のステップを完了します。

Amazon AppStream 2.0 で、フリート、Image Builder、または App Block Builder のインターネットアクセスを有効にする

NAT ゲートウェイが VPC で利用可能になったら、フリート、Image Builder、および App Block Builder でインターネットアクセスを有効にすることができます。

トピック

- [Amazon AppStream 2.0 でフリートのインターネットアクセスを有効にする](#)
- [Amazon AppStream 2.0 で Image Builder のインターネットアクセスを有効にする](#)
- [Amazon AppStream 2.0 で App Block Builder のインターネットアクセスを有効にする](#)

Amazon AppStream 2.0 でフリートのインターネットアクセスを有効にする

インターネットアクセスは、フリートの作成時または後で有効にすることができます。

フリートの作成時にインターネットアクセスを有効にするには

1. [Amazon AppStream 2.0 でフリートを作成する](#) の [Step 4: Configure Network] までのステップを完了します。

2. NAT ゲートウェイのある VPC を選択します。
3. サブネットフィールドが空の場合は、[Subnet 1 (サブネット 1)] でプライベートサブネットを選択し、必要に応じて、[Subnet 2 (サブネット 2)] に別のプライベートサブネットを指定します。VPC にプライベートサブネットがない場合は、2 つ目のプライベートサブネットを作成する必要があります。
4. [Amazon AppStream 2.0 でフリートを作成する](#) のステップを続行します。

NAT ゲートウェイを使用してフリートの作成後にインターネットアクセスを有効にするには

1. ナビゲーションペインの [Fleets] を選択します。
2. フリートを選択し、状態が [Stopped (停止)] であることを確認します。
3. [Fleet Details]、[Edit] の順に選択し、NAT ゲートウェイのある VPC を選択します。
4. [Subnet 1 (サブネット 1)] にはプライベートサブネットを選択し、必要に応じて、[Subnet 2 (サブネット 2)] には別のプライベートサブネットを選択します。VPC にプライベートサブネットがない場合は、[2 つ目のプライベートサブネットを作成する必要があります](#)。
5. [更新] を選択します。

フリートを開始し、ストリーミングインスタンスに接続してインターネット参照を行うことによって、インターネット接続をテストできます。

Amazon AppStream 2.0 で Image Builder のインターネットアクセスを有効にする

Image Builder でインターネットアクセスを有効にする場合は、Image Builder の作成時にインターネットアクセスを有効にする必要があります。

Image Builder のインターネットアクセスを有効にするには

1. [Image Builder を起動し、ストリーミングアプリケーションをインストールして設定する](#) の Step 3: Configure Network までのステップを完了します。
2. NAT ゲートウェイのある VPC を選択します。
3. [Subnet] が空であれば、サブネットを選択します。
4. [Image Builder を起動し、ストリーミングアプリケーションをインストールして設定する](#) のステップを続行します。

Amazon AppStream 2.0 で App Block Builder のインターネットアクセスを有効にする

App Block Builder でインターネットアクセスを有効にする場合は、App Block の作成時にインターネットアクセスを有効にする必要があります。

App Block Builder のインターネットアクセスを有効にするには

1. [the section called “App Block Builder を作成する”](#) の [ステップ 2: ネットワークの設定] までのステップを完了します。
2. NAT ゲートウェイのある VPC を選択します。
3. [Subnet] が空であれば、サブネットを選択します。
4. [the section called “App Block Builder を作成する”](#) のステップを続行します。

パブリックサブネットを使用して新しい VPC または既存の VPC を設定する

2013-12-04 以降に Amazon Web Services アカウントを作成した場合、各 AWS リージョンにデフォルトのパブリックサブネットを含むデフォルトの [VPC](#) があります。ただし、独自のデフォルト以外の VPC を作成するか、AppStream 2.0 と共に使用する既存の VPC を設定することもできます。このトピックでは、AppStream 2.0 で使用するデフォルト以外の VPC とパブリックサブネットを設定する方法について説明します。

VPC とパブリックサブネットを設定したら、[Default Internet Access (既定のインターネットアクセス)] オプションを有効にすることで、ストリーミングインスタンス (フリートインスタンスと Image Builder) にインターネットへのアクセスを提供できます。AppStream 2.0 では、ストリーミングインスタンスから VPC パブリックサブネットにアタッチされたネットワークインターフェイスに [Elastic IP アドレス](#) を関連付けることにより、インターネット接続が有効になります。Elastic IP アドレスは、インターネットからアクセス可能なパブリック IPv4 アドレスです。このため、AppStream 2.0 インスタンスへのインターネットアクセスを提供するために NAT ゲートウェイを使用することをお勧めします。また、[Default Internet Access (既定のインターネットアクセス)] が有効になっている場合、最大 100 のフリートインスタンスがサポートされます。デプロイで 100 を超える同時ユーザーをサポートする必要がある場合は、代わりに [NAT ゲートウェイ設定](#) を使用します。

詳細については、[プライベートサブネットの VPC および NAT ゲートウェイを設定する](#) のステップを参照してください。VPC 設定のその他の推奨事項については、[VPC セットアップの推奨事項](#) を参照してください。

目次

- [ステップ 1: パブリックサブネットに VPC を設定する](#)
- [ステップ 2: フリート、Image Builder、または App Block Builder で既定のインターネットアクセスを有効にする](#)

ステップ 1: パブリックサブネットに VPC を設定する

以下のいずれかの方法を使用して、パブリックサブネットに既定以外の独自の VPC を設定できます。

- [1 つのパブリックサブネットを持つ VPC を作成する](#)
- [既存の VPC を設定する](#)

1 つのパブリックサブネットを持つ VPC を作成する

VPC ウィザードを使用して新しい VPC を作成すると、ウィザードによってインターネットゲートウェイとパブリックサブネットに関連付けられたカスタムルートテーブルが作成されます。ルートテーブルは、VPC の外部のアドレスを宛先とするすべてのトラフィックをインターネットゲートウェイにルーティングします。この設定の詳細については、Amazon VPC ユーザーガイドの [VPC with a Single Public Subnet](#) を参照してください。

1. Amazon VPC ユーザーガイドの [Step 1: Create the VPC](#) のステップを実行して、VPC を作成します。
2. フリートインスタンスと Image Builder がインターネットにアクセスできるようにするには、[ステップ 2: フリート、Image Builder、または App Block Builder で既定のインターネットアクセスを有効にする](#) のステップを完了します。

既存の VPC を設定する

パブリックサブネットが設定されていない既存の VPC を使用する場合は、新しいパブリックサブネットを追加します。パブリックサブネットに加えて、VPC にインターネットゲートウェイをアタッチし、VPC 外部のアドレス宛てのすべてのトラフィックをインターネットゲートウェイにルーティングするルートテーブルも必要です。これらのコンポーネントを設定するには、次のステップを実行します。

1. パブリックサブネットを追加するには、[Creating a Subnet in Your VPC](#) のステップを実行します。AppStream 2.0 で使用する予定の既存の VPC を使用します。

- VPC が IPv6 アドレス指定をサポートするように設定されている場合、[IPv6 CIDR block (IPv6 CIDR ブロック)] リストが表示されます。[Don't assign Ipv6 (Ipv6 を割り当てない)] を選択します。
2. インターネットゲートウェイを作成して VPC にアタッチするには、[Creating and Attaching an Internet Gateway](#) のステップを実行します。
 3. インターネットトラフィックがインターネットゲートウェイを介してルーティングされるようにサブネットを設定するには、[Creating a Custom Route Table](#) に記載されているステップに従います。ステップ 5 では、[Destination (宛先)] に IPv4 形式 (0.0.0.0/0) を使用します。
 4. フリートインスタンスと Image Builder がインターネットにアクセスできるようにするには、[ステップ 2: フリート、Image Builder、または App Block Builder で既定のインターネットアクセスを有効にする](#) のステップを完了します。

ステップ 2: フリート、Image Builder、または App Block Builder で既定のインターネットアクセスを有効にする

パブリックサブネットを持つ VPC を設定したら、フリートおよび Image Builder の [Default Internet Access (既定のインターネットアクセス)] オプションを有効にできます。

フリートの既定のインターネットアクセスを有効にする

フリートを作成するとき、または後で [Default Internet Access (既定のインターネットアクセス)] オプションを有効にできます。

Note

[Default Internet Access (既定のインターネットアクセス)] オプションを有効にしているフリートインスタンスの場合、制限は 100 です。

フリートの作成時にインターネットアクセスを有効にするには

1. [Amazon AppStream 2.0 でフリートを作成する](#) の [Step 4: Configure Network] までのステップを完了します。
2. [Default Internet Access (既定のインターネットアクセス)] チェックボックスをオンにします。
3. サブネットフィールドが空の場合は、[Subnet 1 (サブネット 1)] でサブネットを選択し、必要に応じて [Subnet 2 (サブネット 2)] も指定します。

4. [Amazon AppStream 2.0 でフリートを作成する](#) のステップを続行します。

フリートの作成後にインターネットアクセスを有効にするには

1. ナビゲーションペインの [Fleets] を選択します。
2. フリートを選択し、その状態が [Stopped (停止)] であることを確認します。
3. [Fleet Details (フリートの詳細)]、[Edit (編集)] の順に選択し、[Default Internet Access (既定のインターネットアクセス)] チェックボックスをオンにします。
4. [Subnet 1 (サブネット 1)] のサブネットを選択し、必要に応じて、[Subnet 2 (サブネット 2)] を選択します。[更新] を選択します。

フリートの開始とスタックの作成を行い、フリートをスタックに関連付けて、スタックのストリーミングセッションでインターネット参照を行うことによってインターネット接続をテストできます。詳細については、「[Amazon AppStream 2.0 フリートとスタックを作成する](#)」を参照してください。

Image Builder の既定のインターネットアクセスを有効にする

パブリックサブネットを持つ VPC を設定したら、Image Builder の [Default Internet Access (既定のインターネットアクセス)] オプションを有効にできます。これは、Image Builder を作成するときに行うことができます。

Image Builder のインターネットアクセスを有効にするには

1. [Image Builder を起動し、ストリーミングアプリケーションをインストールして設定する](#) の Step 3: Configure Network までのステップを完了します。
2. [Default Internet Access (既定のインターネットアクセス)] チェックボックスをオンにします。
3. [Subnet 1 (サブネット 1)] が空であれば、サブネットを選択します。
4. [Image Builder を起動し、ストリーミングアプリケーションをインストールして設定する](#) のステップを続行します。

App Block Builder の既定のインターネットアクセスを有効にする

パブリックサブネットを持つ VPC を設定したら、App Block Builder の [既定のインターネットアクセス] オプションを有効にできます。これは、App Block Builder を作成するときに行うことができます。

App Block Builder のインターネットアクセスを有効にするには

1. [the section called “App Block Builder を作成する”](#) の [ステップ 2: ネットワークの設定] までのステップに従います。
2. [Default Internet Access (既定のインターネットアクセス)] チェックボックスをオンにします。
3. [Subnet] が空であれば、サブネットを選択します。
4. [the section called “App Block Builder を作成する”](#) のステップを続行します。

デフォルト VPC、パブリックサブネット、およびセキュリティグループの使用

Amazon Web Services アカウントが 2013-12-04 以降に作成された場合、各 AWS リージョンにデフォルトの VPC があります。デフォルト VPC には、各アベイラビリティゾーンのデフォルトのパブリックサブネットと、VPC にアタッチされたインターネットゲートウェイが含まれます。VPC にはデフォルトのセキュリティグループも含まれます。AppStream 2.0 を初めて使用し、サービスの使用を開始する場合は、フリートの作成時、App Block Builder の作成時、Image Builder の起動時に、デフォルトの VPC とセキュリティグループを選択したままにしておくことができます。次に、少なくとも 1 つのデフォルトサブネットを選択できます。

Note

Amazon Web Services アカウントが 2013 年 12 月 4 日より前に作成されている場合は、新しい VPC を作成するか、AppStream 2.0 と共に使用する既存の VPC を設定する必要があります。フリート、App Block Builder、および Image Builder 用の 2 つのプライベートサブネットと、パブリックサブネットの NAT ゲートウェイを持つ VPC を手動で設定することをお勧めします。詳細については、「[プライベートサブネットの VPC および NAT ゲートウェイを設定する](#)」を参照してください。または、パブリックサブネットでデフォルト以外の VPC を設定することもできます。詳細については、「[パブリックサブネットを使用して新しい VPC または既存の VPC を設定する](#)」を参照してください。

フリートにデフォルトの VPC、サブネット、およびセキュリティグループを使用するには

1. [Amazon AppStream 2.0 でフリートを作成する](#) の [Step 4: Configure Network] までのステップを完了します。
2. [Step 4: Configure Network (ステップ 4: ネットワークの設定)] で、以下の操作を行います。

- フリートインスタンスがインターネットにアクセスできるようにするには、[Default Internet Access (既定のインターネットアクセス)] チェックボックスをオンにします。

 Note

[Default Internet Access (既定のインターネットアクセス)] オプションを有効にしているフリートインスタンスの場合、制限は 100 です。

- VPC の場合は、AWS リージョンのデフォルト VPC を選択します。

デフォルトの VPC 名では、vpc-*vpc-id* (No_default_value_Name) という形式が使用されます。

- [Subnet 1 (サブネット 1)] で、デフォルトのパブリックサブネットを選択し、アベイラビリティゾーンを書き留めます。

デフォルトのサブネット名は、subnet-*subnet-id* | (*IPv4 CIDR ####*) | Default in *availability-zone* の形式を使用します。

- 必要に応じて、[Subnet 2 (サブネット 2)] で、別のアベイラビリティゾーンのデフォルトサブネットを選択します。
- [Security groups (セキュリティグループ)] で、デフォルトのセキュリティグループを選択します。

デフォルトのセキュリティグループ名は、次の形式を使用します。sg-*security-group-id*-default

3. [Amazon AppStream 2.0 でフリートを作成する](#) のステップを続行します。

Image Builder のデフォルトの VPC、サブネット、およびセキュリティグループを使用するには、以下のステップを実行します。

Image Builder にデフォルトの VPC、サブネット、およびセキュリティグループを使用するには

1. [Image Builder を起動し、ストリーミングアプリケーションをインストールして設定する](#) の Step 3: Configure Network までのステップに従います。
2. [Step 4: Configure Network (ステップ 4: ネットワークの設定)] で、以下の操作を行います。
 - Image Builder でインターネットにアクセスできるようにするには、[Default Internet Access (既定のインターネットアクセス)] チェックボックスをオンにします。

- VPC の場合は、AWS リージョンのデフォルト VPC を選択します。

デフォルトの VPC 名では、vpc-*vpc-id* (No_default_value_Name) という形式が使用されます。

- [Subnet 1 (サブネット 1)] で、デフォルトのパブリックサブネットを選択します。

デフォルトのサブネット名は、subnet-*subnet-id* | (*IPv4 CIDR ####*) | Default in *availability-zone* の形式を使用します。

- [Security groups (セキュリティグループ)] で、デフォルトのセキュリティグループを選択します。

デフォルトのセキュリティグループ名は、次の形式を使用します。sg-*security-group-id*-default

3. [Image Builder を起動し、ストリーミングアプリケーションをインストールして設定する](#) のステップを続行します。

App Block Builder のデフォルトの VPC、サブネット、およびセキュリティグループを使用するには、以下のステップを実行します。

App Block Builder にデフォルトの VPC、サブネット、およびセキュリティグループを使用するには

1. [the section called “App Block Builder を作成する”](#) の [ステップ 2: ネットワークの設定] までのステップに従います。
2. [ステップ 2: ネットワークの設定] で、以下の操作を行います。
 - Image Builder でインターネットにアクセスできるようにするには、[Default Internet Access (既定のインターネットアクセス)] チェックボックスをオンにします。
 - VPC の場合は、AWS リージョンのデフォルト VPC を選択します。

デフォルトの VPC 名では、vpc-*vpc-id* (No_default_value_Name) という形式が使用されます。

- [Subnet 1 (サブネット 1)] で、デフォルトのパブリックサブネットを選択します。

デフォルトのサブネット名は、subnet-*subnet-id* | (*IPv4 CIDR ####*) | Default in *availability-zone* の形式を使用します。

- [Security groups (セキュリティグループ)] で、デフォルトのセキュリティグループを選択します。

デフォルトのセキュリティグループ名は、次の形式を使用します。sg-*security-group-id*-default

3. [the section called “App Block Builder を作成する”](#) のステップを続行します。

AppStream 2.0 機能に対する Amazon S3 VPC エンドポイントの使用

スタックでアプリケーション設定の永続化、またはホームフォルダを有効にすると、AppStream 2.0 はフリート用に指定された VPC を使用して Amazon Simple Storage Service (Amazon S3) バケットへのアクセスを提供します。Elastic フリートの場合、AppStream 2.0 は、フリートの App Block に割り当てられたアプリケーションが含まれる Amazon S3 バケットへのアクセスに VPC を使用します。プライベート S3 エンドポイントへの AppStream 2.0 アクセスを有効にするには、以下のカスタムポリシーを Amazon S3 の VPC エンドポイントにアタッチします。プライベート Amazon S3 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 Endpoints](#) および [Endpoints for Amazon S3](#) を参照してください。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "Allow-AppStream-to-access-S3-buckets",
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:sts::account-id-without-hyphens:assumed-role/
AmazonAppStreamServiceAccess/AppStream2.0"
      },
      "Action": [
        "s3:ListBucket",
        "s3:GetObject",
        "s3:PutObject",
        "s3:DeleteObject",
        "s3:GetObjectVersion",
        "s3:DeleteObjectVersion"
      ],
      "Resource": [
        "arn:aws:s3:::appstream2-36fb080bb8-*",
        "arn:aws:s3:::appstream-app-settings-*",
        "arn:aws:s3:::appstream-logs-*"
      ]
    }
  ]
}
```

```
    },
    {
      "Sid": "Allow-AppStream-ElasticFleetstoRetrieveObjects",
      "Effect": "Allow",
      "Principal": "*",
      "Action": "s3:GetObject",
      "Resource": "arn:aws:s3:::bucket-with-application-or-app-block-objects/*",
      "Condition": {
        "StringEquals": {
          "aws:PrincipalServiceName": "appstream.amazonaws.com"
        }
      }
    }
  ]
}
```

VPC への Amazon AppStream 2.0 の接続

ネットワークリソースとインターネットへの AppStream 2.0 接続を有効にするには、次のようにストリーミングインスタンスを設定します。

トピック

- [Amazon AppStream 2.0 のネットワークインターフェイス](#)
- [Amazon AppStream 2.0 での管理ネットワークインターフェイス IP アドレス範囲とポート](#)
- [Amazon AppStream 2.0 のカスタマーネットワークインターフェイスポート](#)

Amazon AppStream 2.0 のネットワークインターフェイス

各 AppStream 2.0 ストリーミングインスタンスには、次のネットワークインターフェイスがあります。

- カスタマーネットワークインターフェイスは、VPC 内だけでなくインターネットでのリソースへの接続を提供し、ストリーミングインスタンスをディレクトリに結合するために使用されます。
- 管理ネットワークインターフェイスは、セキュアな AppStream 2.0 管理ネットワークに接続します。これは、ストリーミングインスタンスがユーザーのデバイスと通信するために使用され、AppStream 2.0 がストリーミングインスタンスを管理できるようにします。

AppStream 2.0 は、管理ネットワークインターフェイス用の IP アドレスをプライベート IP アドレス範囲 (198.19.0.0/16) から選択します。この範囲を VPC CIDR に使用すること、あるいは VPC を他の VPC にこの範囲でピア接続することは避けてください。競合が生じて、ストリーミングインスタンスに接続できなくなることがあります。また、ストリーミングインスタンスに添付しているどのネットワークインターフェイスも編集あるいは削除しないでください。これも、ストリーミングインスタンスの未接続を引き起こす場合があります。

Amazon AppStream 2.0 での管理ネットワークインターフェイス IP アドレス範囲とポート

管理ネットワークインターフェイス IP アドレス範囲は、198.19.0.0/16 です。次のポートは、すべてのストリーミングインスタンスの管理ネットワークインターフェイスで開いている必要があります。

- ポート 8300 のインバウンド TCP。これはストリーミング接続の確立に使用されます。
- ポート 8000 と 8443 のインバウンド TCP。これらは、AppStream 2.0 によるストリーミングインスタンスの管理に使用されます。
- ポート 8300 のインバウンド UDP。これは UDP でのストリーミング接続の確立に使用されます。

管理ネットワークインターフェイスでインバウンドの範囲 198.19.0.0/16 に制限します。

通常の状況では、AppStream 2.0 はストリーミングインスタンスに対してこれらのポートを正しく設定します。これらのいずれかのポートをブロックするセキュリティソフトウェアまたはファイアウォールソフトウェアがストリーミングインスタンスにインストールされている場合、ストリーミングインスタンスは適切に機能することもあるが、アクセスできないこともあります。

IPv6 を無効にしないでください。IPv6 を無効にすると、AppStream 2.0 は正しく機能しません。Windows 用の IPv6 の設定については、「[上級ユーザー向けに Windows で IPv6 を構成するためのガイド](#)」を参照してください。

Note

AppStream 2.0 は VPC 内の DNS サーバーに依存して、存在しないローカルドメイン名に対して存在しないドメイン (NXDOMAIN) レスポンスを返します。これにより、AppStream 2.0 マネージド型のネットワークインターフェイスは管理サーバーとやり取りできます。

Simple AD でディレクトリを作成すると、はユーザーに代わって DNS サーバーとしても機能する 2 つのドメインコントローラー AWS Directory Service を作成します。これらのドメ

インコントローラは NXDOMAIN レスポンスを返さないため、AppStream 2.0 で使用することはできません。

Amazon AppStream 2.0 のカスタマーネットワークインターフェイスポート

カスタマーネットワークインターフェイスポートについては、以下のガイダンスに従ってください。

- インターネット接続の場合、すべての接続先に対して次のポートが開いている必要があります。変更された、またはカスタムセキュリティグループを使用している場合、手動で必須ルールを追加する必要があります。詳細については、Amazon VPC ユーザーガイドの [Security Group Rules](#) を参照してください。
 - TCP 80 (HTTP)
 - TCP 443 (HTTPS)
 - UDP 8433
- ディレクトリにストリーミングインスタンスを結合させる場合、AppStream 2.0 VPC とディレクトリコントローラの間で次のポートが開かれている必要があります。
 - TCP/UDP 53 - DNS
 - TCP/UDP 88 - Kerberos 認証
 - UDP 123 – NTP
 - TCP 135 - RPC
 - UDP 137-138 - Netlogon
 - TCP 139 - Netlogon
 - TCP/UDP 389 - LDAP
 - TCP/UDP 445 - SMB
 - TCP 1024-65535 - RPC 用ダイナミックポート

ポートの完全なリストについては、Microsoft ドキュメンテーション の「[Active Directory および Active Directory ドメインサービスのポート要件](#)」を参照してください。

- すべてのストリーミングインスタンスでは、EC2 メタデータサービスへのアクセスができるようにポート 80 (HTTP) が IP アドレス 169.254.169.254 に開放されている必要があります。IP アドレス範囲 169.254.0.0/16 は、管理トラフィックの AppStream 2.0 サービスの使用のため

に予約されています。この範囲を除外しないと、ストリーミングの問題が発生する可能性があります。

Amazon AppStream 2.0 へのユーザー接続

ユーザーは、デフォルトのパブリックインターネットエンドポイントを介して、または Virtual Private Cloud (VPC) に作成するインターフェイス VPC エンドポイント (インターフェイスエンドポイント) を使用し、AppStream 2.0 ストリーミングインスタンスに接続できます。詳細については、「[チュートリアル: インターフェイス VPC エンドポイントからの作成とストリーミング](#)」を参照してください。

デフォルトで AppStream 2.0 は、パブリックインターネットを介してストリーミング接続をルーティングするように設定されています。ユーザーを認証し、AppStream 2.0 が機能するために必要なウェブアセットを配信するためには、インターネットに接続できることが必須です。このトラフィックを許可するには、「[許可されたドメイン](#)」に示されたドメインを許可する必要があります。

Note

ユーザー認証では、AppStream 2.0 は、ユーザープール、Security Assertion Markup Language 2.0 (SAML 2.0)、および [CreateStreamingURL](#) API アクションをサポートします。詳細については、「[ユーザー認証](#)」を参照してください。

次のトピックでは、AppStream 2.0 へのユーザー接続を有効にする方法について説明します。

目次

- [推奨帯域幅](#)
- [AppStream 2.0 ユーザーデバイスの IP アドレスとポートの要件](#)
- [許可されたドメイン](#)

推奨帯域幅

AppStream 2.0 のパフォーマンスを最適化するには、ネットワーク帯域幅とレイテンシーがユーザーのニーズを満たせることを確認します。

AppStream 2.0 は、NICE Desktop Cloud Visualization (DCV) を使用して、さまざまなネットワーク条件でユーザーが安全にアプリケーションにアクセスしてストリーミングできるようにします。

帯域幅の使用量を減らすために、NICE DCV では H.264 ベースのビデオ圧縮とエンコードが使用されます。ストリーミングセッション中、アプリケーションの視覚的な出力は圧縮され、HTTPS で AES-256 暗号化ピクセルストリームとしてユーザーにストリーミングされます。ストリームを受信すると、復号されてユーザーのローカル画面に出力されます。ユーザーが自分のストリーミングアプリケーションを操作するときは、NICE DCV プロトコルでユーザーの入力が取得され、HTTPS でユーザーのストリーミングアプリケーションに返送されます。

この処理の間、ネットワーク状況は常に測定され、AppStream 2.0 に情報が送信されます。AppStream 2.0 は、リアルタイムでビデオとオーディオのエンコードを変更することで、変化するネットワーク状況に動的に対応して、さまざまなアプリケーションとネットワーク状況に合わせた高品質のストリームを生成します。

AppStream 2.0 ストリーミングセッションの推奨帯域幅とレイテンシーはワークロードによって異なります。たとえば、グラフィックを多用するアプリケーションを使用してコンピュータ支援設計タスクを実行するユーザーは、ビジネス生産性アプリケーションを使用してドキュメントを作成するユーザーよりも多くの帯域幅と短いレイテンシーを必要とします。

以下の表では、一般的なワークロード別に AppStream 2.0 ストリーミングセッションの推奨ネットワーク帯域幅およびレイテンシーに関するガイダンスを示しています。

各ワークロードでの推奨帯域幅は、個々のユーザーが特定の時点で何が必要になる可能性があるかに基づいています。これらの推奨事項には、持続的なスループットに必要な帯域幅は反映されていません。ストリーミングセッション中に画面上での変化がわずか数ピクセルである場合、持続的なスループットはさらに低くなります。使用可能な帯域幅が少ないユーザーでもアプリケーションをストリーミングできますが、最適なフレームレートや画質を得られない可能性があります。

ワークロード	説明	ユーザーあたりの推奨帯域幅	推奨最大ラウンドトリップレイテンシー
基幹業務アプリケーション	ドキュメント作成 アプリケーション、データベ ース分析ユーティリ ティ	2 Mbps	150 ミリ秒未満
グラフィックスアプリケー ション	コンピュータ支援 設計およびモデ	5 Mbps	100 ミリ秒未満

ワークロード	説明	ユーザーあたりの推奨帯域幅	推奨最大ラウンドトリップレイテンシー
	リングアプリケーション、写真およびビデオ編集		
高忠実度	マルチモニター対応の忠実度の高いデータセットやマップ	10 Mbps	50 ミリ秒未満

AppStream 2.0 ユーザーデバイスの IP アドレスとポートの要件

インターネットエンドポイントを使用している場合、AppStream 2.0 ユーザーのデバイスにはポート 443 (TCP) およびポート 8433 (UDP) での発信アクセスが必要となります。ドメイン名解決に DNS サーバーを使用している場合は、ポート 53 (UDP) での発信アクセスが必要です。

- インターネットエンドポイントを使用している場合、ポート 443 は、AppStream 2.0 ユーザーのデバイスとストリーミングインスタンスとの HTTPS 通信に使用します。通常の場合、ストリーミングセッション中にエンドユーザーがウェブを閲覧すると、ウェブブラウザはストリーミングトラフィックに広範囲のソースポートをランダムに選択します。このポートへのリターントラフィックが許可されていることを確認する必要があります。

Note

AppStream 2.0 は、ポート 443 で WebSockets を使用します。

- インターネットエンドポイントを使用している場合、ポート 8433 は、AppStream 2.0 ユーザーのデバイスとストリーミングインスタンスの間での UDP HTTPS 通信に使用します。現在、これは Windows ネイティブクライアントでのみサポートされます。VPC エンドポイントを使用している場合、UDP はサポートされません。

Note

インターフェイス VPC エンドポイントを介したストリーミングには、追加のポートが必要です。詳細については、「[チュートリアル: インターフェイス VPC エンドポイントからの作成とストリーミング](#)」を参照してください。

- ポート 53 は、AppStream 2.0 ユーザーのデバイスと DNS サービス間の通信に使用されます。パブリックドメイン名を解決できるように、このポートは DNS サーバーの IP アドレスに対して開いている必要があります。ドメイン名の解決のために DNS サーバーを使用していない場合、このポートはオプションです。

許可されたドメイン

AppStream 2.0 ユーザーがストリーミングインスタンスにアクセスできるようにするには、ユーザーがストリーミングインスタンスへのアクセスを開始するネットワークで、以下のドメインを許可する必要があります。

- ストリーミングゲートウェイ: *.amazonappstream.com

Note

ワイルドカードを使用してすべてのストリーミングゲートウェイを許可リストに登録する代わりに、VPC エンドポイントを作成し、その特定のエンドポイントのみを許可リストに登録できます。詳細については、「[the section called “インターフェイス VPC エンドポイント”](#)」を参照してください。

ユーザー認証を有効にするには、以下のドメインのうち 1 つ以上を許可する必要があります。AppStream 2.0 がデプロイされているリージョンに対応するドメインとサブドメインを許可する必要があります。

リージョン	分野
米国東部 (バージニア北部)	*.appstream2.us-east-1.aws.amazon.com
米国東部 (オハイオ)	*.appstream2.us-east-2.aws.amazon.com

リージョン	分野
米国西部 (オレゴン)	*.appstream2.us-west-2.aws.amazon.com
アジアパシフィック (ムンバイ)	*.appstream2.ap-south-1.aws.amazon.com
アジアパシフィック (ソウル)	*.appstream2.ap-northeast-2.aws.amazon.com
アジアパシフィック (シンガポール)	*.appstream2.ap-southeast-1.aws.amazon.com
アジアパシフィック (シドニー)	*.appstream2.ap-southeast-2.aws.amazon.com
アジアパシフィック (東京)	*.appstream2.ap-northeast-1.aws.amazon.com
カナダ (中部)	*.appstream2.ca-central-1.aws.amazon.com
欧州 (フランクフルト)	*.appstream2.eu-central-1.aws.amazon.com
欧州 (ロンドン)	*.appstream2.eu-west-2.aws.amazon.com
欧州 (アイルランド)	*.appstream2.eu-west-1.aws.amazon.com
欧州 (パリ)	*.appstream2.eu-west-3.aws.amazon.com
AWS GovCloud (米国東部)	*.appstream2.us-gov-east-1.amazonaws-us-gov.com
AWS GovCloud (米国西部)	*.appstream2.us-gov-west-1.amazonaws-us-gov.com
南米 (サンパウロ)	*.appstream2.sa-east-1.aws.amazon.com

Note

ユーザーがネットワークプロキシを使用してストリーミングインスタンスにアクセスする場合は、一覧にあるユーザー認証ドメインとセッションゲートウェイ *.amazonappstream.com のプロキシキャッシュを無効にします。

AWS は、Session Gateway ドメインと CloudFront ドメインが解決できる範囲を含む現在の IP アドレス範囲を JSON 形式で発行します。json ファイルをダウンロードして現在の範囲を表示する方法についての詳細は、Amazon Web Services 全般のリファレンスの「[AWS IP アドレスの範囲](#)」

を参照してください。または、AWS Tools for Windows PowerShell を使用している場合は、Get-AWSPublicIpAddressRange コマンドレットを使用して同じ情報にアクセスできます。Application Auto Scaling ユーザーガイド詳細については、「[AWSに対するパブリック IP アドレス範囲のクエリの実行](#)」を参照してください。

Elastic フリートにアクセスしている AppStream 2.0 ユーザーには、アプリケーションアイコンが含まれる Amazon Simple Storage Service (S3) バケットのドメインへのアクセスを許可する必要があります。

 Note

S3 バケットの名前に「.」文字が含まれている場合、使用されるドメインは `https://s3.<AWS リージョン>.amazonaws.com` になります。S3 バケットの名前に「.」文字が含まれていない場合、使用されるドメインは `https://<bucket name>.s3.<AWS #####>.amazonaws.com` になります。

Image Builder

Amazon AppStream 2.0 では、EC2 インスタンスを使用してアプリケーションをストリーミングします。インスタンスはベースイメージから起動します。ベースイメージは Image Builder と呼ばれ、AppStream 2.0 から提供されます。独自のカスタムイメージを作成するには、Image Builder インスタンスに接続し、ストリーミングするアプリケーションをインストールおよび設定します。次に、Image Builder インスタンスのスナップショットを作成してイメージを作成します。

Image Builder を起動する場合は、以下を選択します。

- インスタンスタイプ — AppStream 2.0 は、コンピューティング、メモリ、グラフィックスの設定がさまざまに異なるインスタンスタイプを提供します。インスタンスタイプは必要なインスタンスファミリーと合致する必要があります。詳細については、「[AppStream 2.0 インスタンスファミリー](#)」を参照してください。
- オペレーティングシステム — AppStream 2.0 は、以下の オペレーティングシステムを提供します。
 - Windows Server 2016 Base
 - Windows Server 2019 ベース
 - Windows Server 2022 Base
 - Amazon Linux 2
 - Red Hat Enterprise Linux 8
 - Rocky Linux 8 ([CIQ の Rocky Linux](#))
- 使用するサブネットとセキュリティグループ — アプリケーションに必要なネットワークリソースへのアクセスが、サブネットとセキュリティグループで許可されていることを確認してください。アプリケーションに必要な一般的なネットワークリソースとしては、ライセンスサーバー、データベースサーバー、ファイルサーバー、アプリケーションサーバーなどがあります。

目次

- [Image Builder を起動し、ストリーミングアプリケーションをインストールして設定する](#)
- [Amazon AppStream 2.0 で Image Builder に接続する](#)
- [Image Builder のアクション](#)
- [AppStream 2.0 Image Builder のインスタンスメタデータ](#)
- [Graphics Design インスタンスへの AMD ドライバーのインストール](#)

- [AppStream 2.0 ベースイメージとマネージド型イメージの更新に関するリリースノート](#)

Image Builder を起動し、ストリーミングアプリケーションをインストールして設定する

ユーザーにストリーミングするアプリケーションをインストールして設定するには、まず、次の手順に示すように Image Builder インスタンスを起動します。

Important

Image Builder を起動して実行すると、アカウントに少額の課金が発生する場合があります。詳細については、[AppStream 2.0 の料金](#)をご覧ください。

Image Builder を起動するには

1. AppStream 2.0 コンソールを <https://console.aws.amazon.com/appstream2> で開きます。
2. Image Builder は以下の方法で起動できます。
 - ウェルカム画面が表示され、2 つのオプション ([Try it now] と [Get started]) が表示された場合は、[Get started]、[Custom set up] の順に選択します。
 - これら 2 つのオプションの詳細については、[Amazon AppStream 2.0 のよくある質問](#) を参照してください。
 - ウェルカム画面が表示されない場合は、左のナビゲーションペインの [Quick links] を選択して、[Custom set up] を選択します。
 - または、左側のナビゲーションペインで [Images] を選択し、さらに [Image Builder] タブ、[Launch Image Builder] の順に選択します。
3. [Step 1: Choose Image] で、ベースイメージを選択します。Image Builder を初めて起動する場合は、によってリリースされた最新のベースイメージのいずれかを使用できます AWS (デフォルトで選択)。によってリリースされたベースイメージの最新バージョンのリストについては AWS、「」を参照してください [AppStream 2.0 ベースイメージとマネージド型イメージの更新に関するリリースノート](#)。イメージを作成済み、または既存のイメージでアプリケーションを更新する場合、既存のイメージのいずれか 1 つを選択できます。必要なインスタンスファミリーに合ったイメージを選択してください。詳細については、「[AppStream 2.0 インスタンスファミリー](#)」を参照してください。

[次へ] を選択します。

4. [Step 2: Configure Image Builder (ステップ 2: Image Builder の設定)] で、以下を実行して Image Builder を設定します。

- [名前]: Image Builder の固有の名前を入力します。
- [表示名 (オプション)]: イメージビルダーに表示する名前を入力します (最大 100 文字)。
- [タグ (省略可能)]: [タグの追加] を選択し、タグのキーと値を入力します。さらにタグを追加するには、この手順を繰り返します。詳細については、「[Amazon AppStream 2.0 リソースにタグを付ける](#)」を参照してください。
- [インスタンスタイプ]: Image Builder のインスタンスタイプを選択します。インストールする予定のアプリケーションのパフォーマンス要件と一致するタイプを選択します。詳細については、「[AppStream 2.0 インスタンスファミリー](#)」を参照してください。
- VPC エンドポイント (アドバンスト): 仮想プライベートインターフェイス (VPC) に、[インターフェイス VPC エンドポイント](#) (インターフェイスエンドポイント) を作成できます。インターフェイスエンドポイントの作成を開始するには、[Create VPC Endpoint (VPC エンドポイントの作成)] を選択します。このリンクを選択すると、VPC コンソールが開きます。エンドポイントの作成を完了するには、[チュートリアル: インターフェイス VPC エンドポイントからの作成とストリーミング](#) の To create an interface endpoint のステップ 3~6 に従います。

インターフェイスエンドポイントを作成したら、VPC 内でストリーミングトラフィックを維持するために使用できます。

- AppStream 2.0 エージェント: このセクションは、の最新のベースイメージ、AWS または最新バージョンのエージェントを使用するカスタムイメージを使用していない場合にのみ表示されます。

AppStream 2.0 エージェントソフトウェアは、ストリーミングインスタンスで実行され、ユーザーがアプリケーションに接続してストリーミングできるようにします。2017 年 12 月 7 日から、ストリーミングインスタンスは最新の AppStream 2.0 エージェントソフトウェアで自動的に更新できます。この機能は、Image Builder に利用可能な最新の機能、パフォーマンスの向上、セキュリティ更新プログラムが含まれていることを確認するのに役立ちます AWS。

AppStream 2.0 エージェントの自動更新を有効にするには、2017 年 12 月 7 日以降 AWS によって発行された任意のベースイメージから新しいイメージを作成します。Image Builder の起動元のイメージで最新バージョンの AppStream 2.0 エージェントが使用されていない場合は、最新のエージェントを使用して Image Builder を起動するためのオプションを選択してください。

- IAM ロール (アドバンスド) : アカウントから AppStream 2.0 Image Builder に IAM ロールを適用すると、AWS 認証情報を手動で管理することなく、Image Builder インスタンスから AWS API リクエストを行うことができます。Image Builder に IAM ロールを適用するには、次のいずれかを行います。
- Amazon Web Services アカウントで既存の IAM ロールを使用するには、[IAM ロール] の一覧表示から使用するロールを選択します。ロールは、イメージビルダーからアクセス可能である必要があります。詳細については、「[AppStream 2.0 ストリーミングインスタンスで使用するための既存の IAM ロールの設定](#)」を参照してください。
- 新しい IAM ロールを作成するには、[Create new IAM role (新しい IAM ロールの作成)] を選択し、[AppStream 2.0 ストリーミングインスタンスで使用する IAM ロールを作成する方法](#) のステップを実行します。

5. [次へ] を選択します。

6. [Step 3: Configure Network (ステップ 3: ネットワークの設定)] で、以下の操作を行います。

- パブリックサブネットを持つ VPC にフリーインスタンスのインターネットアクセスを追加するには、[Default Internet Access (デフォルトのインターネットアクセス)] を選択します。NAT ゲートウェイを使用してインターネットアクセスを提供する場合は、[Default Internet Access (デフォルトのインターネットアクセス)] を未選択のままにします。詳細については、「[インターネットアクセス](#)」を参照してください。
- [VPC] と [サブネット 1] で、VPC と少なくとも 1 つのサブネットを選択します。耐障害性を高めるために、異なるアベイラビリティゾーンで 2 つのサブネットを選択することをお勧めします。詳細については、「[プライベートサブネットの VPC および NAT ゲートウェイを設定する](#)」を参照してください。

独自の VPC とサブネットがない場合は、[デフォルトの VPC](#) を使用するか、独自の VPC を作成できます。独自の VPC とサブネットを作成するには、[Create a new VPC (新しい VPC を作成する)] リンクおよび [Create new subnet (新しいサブネットを作成する)] リンクを選択して作成します。これらのリンクを選択すると、Amazon VPC コンソールが開きます。VPC とサブネットを作成したら、AppStream 2.0 コンソールに戻り、[Create a new VPC (新しい VPC を作成する)] リンクと [Create new subnet (新しいサブネットを作成する)] リンクの左側にある更新アイコンを選択してリストに表示します。詳細については、「[AppStream 2.0 用の VPC の設定](#)」を参照してください。

- [Security group(s) (セキュリティグループ)] で、この Image Builder に関連付ける最大 5 つのセキュリティグループを選択します。独自のセキュリティグループを持たず、デフォルトのセキュリティグループを使用しない場合は、[Create new security group (新しいセキュリティグループを作成する)] リンクを選択して、セキュリティグループを作成します。Amazon

VPC コンソールでサブネットを作成したら、AppStream 2.0 コンソールに戻り、[Create new security group (新しいセキュリティグループを作成する)] リンクの左側にある更新アイコンを選択してリストに表示します。詳細については、「[Amazon AppStream 2.0 のセキュリティグループ](#)」を参照してください。

7. [Active Directory Domain (オプション)] で、このセクションを展開してストリーミングインスタンスコンピュータオブジェクトを配置する Active Directory 設定と組織単位を選択します。選択されたネットワークアクセス設定で DNS による解決が有効であり、ディレクトリと通信できることを確認します。詳細については、「[AppStream 2.0 でのアクティブディレクトリの使用](#)」を参照してください。
8. [Review] を選択し、Image Builder の詳細を確認します。任意のセクションの設定を変更するには、[Edit] を選択し、必要な変更を加えます。
9. 設定内容を確認した後、[Launch] を選択します。

Note

Image Builder を作成するのに十分な制限 (クォータ) がないというエラーメッセージが表示された場合は、<https://console.aws.amazon.com/servicequotas/> の Service Quotas コンソールから制限引き上げリクエストを送信してください。詳細については「Service Quotas ユーザーガイド」の「[クォータの引き上げのリクエスト](#)」を参照してください。

10. Image Builder の作成プロセスで、AppStream 2.0 が必要なリソースを準備するまでは、Image Builder のステータスが [Pending (保留中)] と表示されます。Image Builder のステータスを更新するには、[更新] アイコンを定期的にクリックします。ステータスが [Running (実行中)] に変わると、Image Builder の準備完了となり、カスタムイメージを作成できます。

次のステップ

次に、ストリーミングするアプリケーションをインストールして設定し、Image Builder インスタンスのスナップショットを作成してイメージを作成します。詳細については、「[チュートリアル: AppStream 2.0 コンソールを使用してカスタム AppStream 2.0 イメージを作成する](#)」を参照してください。

Amazon AppStream 2.0 で Image Builder に接続する

次のいずれかの操作を行って、Image Builder に接続できます。

- AppStream 2.0 コンソールの使用 (ウェブ接続のみ)

- ストリーミング URL の作成 (ウェブまたは AppStream 2.0 クライアント接続用)

Note

接続先の Image Builder が Active Directory ドメインに参加していて、組織でスマートカードサインインが必要な場合は、[ストリーミング URL を作成](#)し、接続に AppStream 2.0 クライアントを使用する必要があります。

目次

- [AppStream 2.0 コンソール \(ウェブ接続\)](#)
- [ストリーミング URL \(AppStream 2.0 クライアントまたはウェブ接続\)](#)

AppStream 2.0 コンソール (ウェブ接続)

AppStream 2.0 コンソールを使用してウェブブラウザ経由で Image Builder に接続するには、次の手順を実行します。

1. AppStream 2.0 コンソールを <https://console.aws.amazon.com/appstream2> で開きます。
2. 左側のナビゲーションペインで、[Images]、[Image Builder] の順に選択します。
3. Image Builder 一覧で、接続する Image Builder を選択します。Image Builder のステータスが [Running (実行中)] であることを確認し、[Connect (接続)] を選択します。

この手順を実行するには、<https://stream.<aws-region>.amazonappstream.com/> からのポップアップを許可するようにブラウザを設定する必要があるかもしれません。

4. 次のいずれかを実行して、Image Builder にログインします。
 - Image Builder が Windows、Red Hat Enterprise Linux、または Rocky Linux を利用している場合は、ローカルユーザータブで次のいずれかを選択します。
 - Administrator (管理者) — Image Builder にアプリケーションをインストールしてイメージを作成するか、ローカル管理者権限を必要とするその他のタスクを実行するには、[Administrator (管理者)] を選択します。
 - [テンプレートユーザー (Windows のみ)] — デフォルトのアプリケーションおよび Windows 設定を作成するには、[テンプレートユーザー] を選択します。

- Test User (ユーザーのテスト) — [Test User (ユーザーのテスト)] を選択して、アプリケーションを開き、設定を確認します。
- Image Builder が Windows、Red Hat Enterprise Linux、または Rocky Linux を使用していて、Active Directory ドメインに結合されており、アプリケーションをインストールするために Active Directory によって管理されるリソースにアクセスする必要がある場合は、ディレクトリユーザータブを選択します。次に、Image Builder に対するローカル管理者権限を持つドメインアカウントの認証情報を入力します。

 Note

スマートカードサインインは、ウェブブラウザを介した接続ではサポートされていません。代わりに、ストリーミング URL を作成し、AppStream 2.0 クライアントを使用する必要があります。スマートカードサインインの詳細については、「[スマートカード](#)」をご参照ください。

- Image Builder が Amazon Linux 2 で動作している場合は、以下のユーザーとして自動的にログインされます。

Amazon Linux GNOME デスクトップの ImageBuilderAdmin ユーザーで、ルート管理者権限を持ってる。

ストリーミング URL (AppStream 2.0 クライアントまたはウェブ接続)

ストリーミング URL を作成して、ウェブブラウザまたは AppStream 2.0 クライアントを介して Image Builder に接続できます。フリーインスタンスへのユーザーアクセスを有効にするように作成するストリーミング URL とは異なり、デフォルトでは、Image Builder にアクセスするために作成するストリーミング URL は、1 時間後に失効します。別の有効期限を設定するには、[CreateStreamingURL](#) API アクションを使用してストリーミング URL を生成する必要があります。

接続先の Image Builder が Active Directory ドメインに参加していて、組織でスマートカードサインインが必要な場合は、ストリーミング URL を作成し、接続に AppStream 2.0 クライアントを使用する必要があります。

 Note

ネイティブアプリケーションモードは、Image Builder への AppStream 2.0 クライアント接続ではサポートされていません。AppStream 2.0 クライアントを使用して Image Builder

に接続し、[Start in native application mode (ネイティブアプリケーションモードで起動)] チェックボックスがオンになっている場合、セッションがクラシックモードに切り替えられたことを示す AppStream 2.0 エラー通知が表示されます。

ストリーミング URL は、次のいずれかの方法で作成できます。

- AppStream 2.0 コンソール
- [CreateStreamingURL](#) API アクション
- [create-streaming-url](#) AWS CLI コマンド

AppStream 2.0 コンソールを使用してストリーミング URL を作成し、Image Builder に接続するには、以下の手順を実行します。

AppStream 2.0 コンソールを使用してストリーミング URL を作成し、Image Builder に接続するには

1. AppStream 2.0 コンソールを <https://console.aws.amazon.com/appstream2> で開きます。
2. ナビゲーションペインで、[Images (イメージ)]、[Image Builder (イメージビルダー)] の順に選択します。
3. Image Builder 一覧で、接続する Image Builder を選択します。Image Builder のステータスが [Running (実行中)] であることを確認します。
4. [Actions (アクション)]、[Create streaming URL (ストリーミング URL の作成)] の順に選択します。
5. 次のいずれかを行ってください。
 - ストリーミング URL を保存した後に Image Builder に接続するには、[Copy Link (リンクをコピー)] を選択して URL をコピーし、アクセス可能な場所に保存します。
 - AppStream 2.0 クライアント経由で Image Builder に接続するには、[Launch in Client (クライアントで起動)] を選択します。このオプションを選択すると、AppStream 2.0 クライアントのサインインページに事前にストリーミング URL が入力されます。
 - ウェブブラウザを使用して Image Builder に接続するには、[Launch in Browser (ブラウザで起動)] を選択します。このオプションを選択すると、ウェブブラウザが開き、ストリーミング URL が事前に入力されたアドレスバーが表示されます。
6. ストリーミング URL を作成して Image Builder に接続したら、次のいずれかの方法で Image Builder にログインします。

- Image Builder が Windows ベースであり、Active Directory ドメインに参加していない場合は、[ローカルユーザー] タブで、以下のいずれかを選択します。
 - Administrator (管理者) — Image Builder にアプリケーションをインストールしてイメージを作成するか、ローカル管理者権限を必要とするその他のタスクを実行するには、[Administrator (管理者)] を選択します。
 - Template User (テンプレートユーザー) — デフォルトのアプリケーションおよび Windows 設定を作成するには、[Template User (テンプレートユーザー)] を選択します。
 - Test User (ユーザーのテスト) — [Test User (ユーザーのテスト)] を選択して、アプリケーションを開き、設定を確認します。
- Image Builder が Windows ベースであり、Active Directory ドメインに参加しており、かつアプリケーションをインストールするために Active Directory によって管理されているリソースにアクセスする必要がある場合は、[ディレクトリユーザー] タブを選択し、Image Builder に対するローカル管理者権限を持つドメインアカウントの認証情報を入力します。

 Note

AppStream 2.0 クライアントを使用している場合は、Active Directory ドメインのパスワードを入力して [Password sign in (パスワードサインイン)] を選択するか、[Choose a smart card (スマートカードを選択)] を選択してプロンプトが表示されたら、スマートカードの PIN を入力します。

ウェブブラウザを使用している場合は、Active Directory ドメインパスワードを入力する必要があります。スマートカードサインインは、ストリーミングインスタンスへの AppStream 2.0 クライアント接続でのみサポートされます。

- Image Builder が Linux ベースの場合は、以下のユーザーとして自動的にログインされます。

Amazon Linux GNOME デスクトップの ImageBuilderAdmin ユーザーで、ルート管理者権限を持ってる。

Image Builder のアクション

Image Builder インスタンスの現在の状態 (ステータス) に応じて、Image Builder で以下のアクションを実行できます。

削除

Image Builder を完全に削除します。

インスタンスは [Stopped] 状態である必要があります。

接続

実行中の Image Builder に接続します。このアクションは、Image Builder を使用してデスクトップストリーミングセッションを開始して、イメージにアプリケーションをインストールして追加し、イメージを作成します。

インスタンスは [Running] 状態である必要があります。

開始

停止されている Image Builder を開始します。実行中のインスタンスは、お客様のアカウントに請求されます。

インスタンスは [Stopped] 状態である必要があります。

停止

実行中の Image Builder を停止します。停止されたインスタンスは、お客様のアカウントに請求されません。

インスタンスは [Running] 状態である必要があります。

次の中間の状態のインスタンスでは、これらのアクションは実行されません。

- [保留中]
- [Snapshotting]
- 停止中
- スタート
- 削除
- [更新中]
- [認定待ち]

AppStream 2.0 Image Builder のインスタンスメタデータ

AppStream 2.0 Image Builder には、Windows 環境変数を介して利用可能なインスタンスのメタデータがあります。アプリケーションおよびスクリプトで次の環境変数を使用して、Image Builder インスタンスの詳細に基づいて環境を変更できます。

環境変数	Context	説明
AppStream_Image_Arn	マシン	ストリーミングインスタンスを作成するために使用されたイメージの ARN。
AppStream_Instance_Type	マシン	ストリーミングインスタンスのインスタンスタイプ。例えば、 stream.standard.medium と指定します。
AppStream_Resource_Type	マシン	AppStream 2.0 リソースのタイプ。この値は fleet または imagebuilder のどちらかです。
AppStream_Resource_Name	マシン	Image Builder の名前。

Linux Image Builder では、`/etc/profile.d/appstream_system_vars.sh` にあるスクリプトを通じて環境変数がエクスポートされます。環境変数にアクセスするには、アプリケーションでこのファイルを明示的にソースすることができます。

Graphics Design インスタンスへの AMD ドライバーのインストール

Graphics Design インスタンスを使用している Windows Image Builder で AMD ドライバーを更新する必要がある場合は、最新の AppStream 2.0 Graphics Design ベースイメージを使用するか、AMD ドライバーをダウンロードして Image Builder にインストールすることができます。Graphics Design インスタンスファミリーの既存のイメージで、AMD ドライバを更新する必要がある場合は、AppStream 2.0 のマネージド型イメージ更新機能を使用します。詳細については、「[the section called “AppStream 2.0 のマネージド型イメージアップデートを使用してイメージを更新する”](#)」を参照してください。

AMD ドライバーのダウンロードは、AWS のお客様のみが利用できます。ダウンロードすることで、AMD FirePro S7150x2 Server GPU ハードウェアで動作する AppStream 2.0 Graphics Design インスタンスで使用するイメージの構築にのみ、このダウンロードしたソフトウェアを使用することに同意したものと見なされます。このソフトウェアをインストールした場合には、[AMD Software End User License Agreement](#) の規約の遵守に同意したものと見なされます。

Graphics Design インスタンス用の最新の AMD ドライバのバージョンは、24.20.13028.5012 です。

開始する前に、以下の前提条件を満たしていることを確認してください。

- Windows インスタンスで AWS Tools for Windows PowerShell のデフォルトの認証情報を設定します。詳細については、「[AWS Tools for Windows PowerShellの開始方法](#)」を参照してください。
- IAM ユーザーは、AmazonS3ReadOnlyAccess ポリシーによって付与されたアクセス許可を持っている必要があります。

Image Builder に AMD ドライバをインストールするには

1. Windows Image Builder インスタンスに接続し、管理者として PowerShell ウィンドウを開きます。
2. 次の PowerShell コマンドを使用して、Amazon S3 からデスクトップにドライバをダウンロードします。

```
$Bucket = "appstream2-driver-patches"
$LocalPath = "$home\Desktop\AMD"
$Objects = Get-S3Object -BucketName $Bucket -Region us-east-1
foreach ($Object in $Objects) {
    $LocalFileName = $Object.Key
    if ($LocalFileName -ne '' -and $Object.Size -ne 0) {
        $LocalFilePath = Join-Path $LocalPath $LocalFileName
        Copy-S3Object -BucketName $Bucket -Key $Object.Key -LocalFile $LocalFilePath -
Region us-east-1
    }
}
```

3. ダウンロードしたドライバファイルを .zip 解凍し、次の PowerShell コマンドによりインストールを実行します。

```
Expand-Archive $LocalFilePath -DestinationPath $home\Desktop -Verbose
$Driverdir = Get-ChildItem $home\Desktop\ -Directory -Filter "*210426a-366782C*"
Write-Host $Driverdir
pnputil /add-driver $home\Desktop\$Driverdir\Packages\Drivers\Display\WT6A_INF
\*inf /install
```

4. ドライバーをインストールする手順にしたがい、必要に応じてインスタンスを再起動します。
5. GPU が正しく動作していることを確認するには、デバイスマネージャをチェックします。ディスプレイアダプタとして AMD MxGPU が、最新のドライババージョンとともに表示されることを確認します。

AppStream 2.0 ベースイメージとマネージド型イメージの更新に関するリリースノート

Amazon AppStream 2.0 は、ユーザー独自のアプリケーションを含むイメージの作成を支援するために、ベースイメージを提供しています。ベースイメージは、オペレーティングシステムに固有のソフトウェア設定が含まれる Amazon マシンイメージ (AMI) です。AppStream 2.0 では、AppStream 2.0 エージェントと、以下のオペレーティングシステムのいずれかの最新バージョンが各ベースイメージに含まれています。

Important

ベンダーによってサポートされなくなったオペレーティングシステムのバージョンは、動作が保証されず、AWS サポートによってサポートされません。

- Windows Server 2016 Base — Base、Graphics Design、Graphics G4dn、Graphics Pro、Graphics G5 の各イメージタイプで使用可能
- Windows Server 2019 Base — Base、Graphics Design、Graphics G4dn、Graphics Pro、Graphics G5 の各イメージタイプで使用可能
- Windows Server 2022 Base — Base、Graphics G4dn、Graphics G5 の各イメージタイプで使用可能
- Amazon Linux 2 – Base、Graphics G4dn、Graphics Pro、Graphics G5 のイメージタイプで使用可能
- Red Hat Enterprise Linux 8 – Base、Graphics G4dn、Graphics G5 の各イメージタイプで使用可能

- Rocky Linux 8 – Base、Graphics G4dn、Graphics G5 のイメージタイプで利用可能

ユーザーが独自のアプリケーションが含まれる独自のイメージを作成する場合、オペレーティングシステム向けの更新プログラム、アプリケーション、およびそれらの依存関係のインストールと維持はユーザーの責任になります。AppStream 2.0 には、マネージド型のイメージ更新機能を使用する、イメージの自動的な更新手段が用意されています。マネージド型イメージ更新を使用するユーザーは、更新するイメージを選択します。AppStream 2.0 は、同じ AWS アカウントとリージョンに Image Builder を作成し、更新をインストールして新しいイメージを作成します。新しいイメージの作成完了後は、そのイメージを、実稼働前のフリートで更新する前にテストしたり、他の AWS アカウントと共有したりできます。詳細については、[Amazon AppStream 2.0 画像を管理する](#)の「Keep Your AppStream 2.0 Image Up-to-Date」を参照してください。

最新リリースの AppStream 2.0 エージェントの詳細については、「[AppStream 2.0 エージェントのリリースノート](#)」を参照してください。

最新のリリースされたイメージを次の表に示します。

Note

Graphics Pro インスタンスのパブリックベースイメージは、Graphics Pro インスタンスタイプをサポートするハードウェアのサポート終了により、10/31/2025 以降 AWS は使用できなくなります。

Graphics Design インスタンスのパブリックベースイメージは、Graphics Design インスタンスタイプをサポートするハードウェアのサポート終了により、12/31/2025 AWS 以降は使用できなくなります。

Image type (イメージタイプ)	イメージ名
基本	• AppStream-WinServer2016-05-30-2025 • AppStream-WinServer2019-05-30-2025 • AppStream-WinServer2022-05-30-2025 • AppStream-AmazonLinux2-02-11-2025 • AppStream-RHEL8-05-30-2025 • AppStream-RockyLinux8-05-30-2025

Image type (イメージタイプ)	イメージ名
グラフィックスデザイン	- AppStream-Graphics-Design-WinServer2016-10-22-2024 - AppStream-Graphics-Design-WinServer2019-10-22-2024
Graphics G4dn	- AppStream-Graphics-G4dn-WinServer2016-05-30-2025 - AppStream-Graphics-G4dn-WinServer2019-05-30-2025 - AppStream-Graphics-G4dn-WinServer2022-05-30-2025 - AppStream-Graphics-G4dn-AmazonLinux2-02-11-2025 - AppStream-Graphics-G4dn-RHEL8-05-30-2025 - AppStream-Graphics-G4dn-RockyLinux8-05-30-2025
Graphics G5	- AppStream-Graphics-G5-WinServer2016-05-30-2025 - AppStream-Graphics-G5-WinServer2019-05-30-2025 - AppStream-Graphics-G5-WinServer2022-05-30-2025 - AppStream-Graphics-G5-AmazonLinux2-02-11-2025 - AppStream-Graphics-G5-RHEL8-05-30-2025 - AppStream-Graphics-G5-RockyLinux8-05-30-2025
Graphics Pro	- AppStream-Graphics-Pro-WinServer2016-10-22-2024 - AppStream-Graphics-Pro-WinServer2019-10-22-2024 - AppStream-Graphics-Pro-AmazonLinux2-10-22-2024
サンプルアプリ	Amazon-AppStream2-Sample-Image-06-17-2024 このベースイメージにアクセスする方法については、 「Amazon AppStream 2.0 の使用を開始する: サンプルアプリケーションを使用してセットアップする」を参照してください。

次の表に、リリースされた最新ベースイメージ用のソフトウェアコンポーネント、ならびに、マネージド型のイメージ更新機能を使用してイメージを更新する場合に使用可能な各コンポーネントを示します。バージョンに「Latest」と記されている場合、ベンダーから入手可能な、現行の安定ソフトウェアコンポーネントがインストールされます。バージョンに「Not included」と記されているコンポーネントの場合は、マネージド型イメージ更新では管理されないため、イメージを更新してもバージョンは変更されません。

次の表に、最新のリリースされた Windows、Amazon Linux、Rocky Linux、Red Hat Enterprise Linux ベースイメージと Managed AppStream 2.0 イメージ更新のソフトウェアコンポーネントを示します。

Windows

ソフトウェアコンポーネント	最新のベースイメージ (2025 年 5 月 30 日)	マネージド AppStream 2.0 イメージの更新 (2025 年 5 月 30 日)
Amazon AWS (AvsCamera) ドライバー	1.0.0.6	1.0.0.6
Amazon CloudWatch Agent	1.300055.0b1095	1.300055.0b1095
SSM Agent	3.3.2299.0	3.3.2299.0
NICE DCV 仮想ディスプレイ	2024 年 0 月 ~ 19143 年	2024 年 0 月 ~ 19143 年
AMD Driver for Graphics Design instances	24.20.13028.7002	24.20.13028.7002
AppStream 2.0 エージェント	最新 (05-14-2025)	--
AWS コマンドラインインターフェイス (AWS CLI)	1.40.24 (Windows Server 2016/2019) 2.27.24.0 (Windows Server 2022)	含まれない
Firefox	138.0.3 (Windows Server 2016/2019)	含まれない

ソフトウェアコンポーネント	最新のベースイメージ (2025 年 5 月 30 日)	マネージド AppStream 2.0 イメージの更新 (2025 年 5 月 30 日)
Microsoft Message Queuing (MSMQ)	Windows Server にインストール済み	Windows Server にインストール済み
NVIDIA Graphics Driver for Graphics Pro、G4dn、および G5 インスタンス	572.83 (Windows Server 2022) 539.19 (Windows Server 2019) 512.78 (Windows Server 2,016)	572.83 (Windows Server 2022) 539.19 (Windows Server 2019) 512.78 (Windows Server 2,016)
プロセスモニタ	4.01	Latest
Windows の高品質な音声/動画エクスペリエンス (qWAVE)	Windows Server にインストール済み	Windows Server にインストール済み
Visual C++ 再頒布可能パッケージ	Microsoft Visual C++ 2013 Redistributable (x64) - 12.0.40664.0 Microsoft Visual C++ 2015-2022 再頒布可能 (x64) - 14.42.34438	Microsoft Visual C++ 2013 Redistributable (x64) - 12.0.40664.0 Microsoft Visual C++ 2015-2022 再頒布可能 (x64) - 14.42.34438
Windows Server 更新プログラム	2025 年 5 月現在のベースイメージの更新	Latest
WinSCARD フィルタードライバ	1.0.19.0	1.0.19.0
準仮想化 (PV) ドライバー	8.5.0	8.5.0
ENA ドライバー	2.9.0	2.9.0

ソフトウェアコンポーネント	最新のベースイメージ (2025 年 5 月 30 日)	マネージド AppStream 2.0 イメージの更新 (2025 年 5 月 30 日)
AWS NVMe ドライバー	1.6.0.35	1.6.0

Amazon Linux

ソフトウェアコンポーネント	最新のベースイメージ (2024 年 10 月 22 日)	マネージド AppStream 2.0 イメージの更新 (2024 年 10 月 22 日)
AWS コマンドラインインターフェイス (AWS CLI)	1.18.147-1	含まれない
Amazon CloudWatch Agent	1.300041.1-1	1.300041.1-1
SSM Agent	3.3.1611.0-1	3.3.1611.0-1
NICE DCV Server AppStream	2024.0.18380-1	2024.0.18380-1
Cloud-init	19.3 ~ 46	含まれない
AL2 カーネル	4.14.355-275.570	含まれない
NVIDIA Graphics Driver for G4dn、および G5 インスタンス	550.127.05	550.127.05
NVIDIA Graphics Driver for Graphics Pro instances	535.216.01	535.216.01
CUDA バージョン	12.4	含まれない

Rocky Linux

ソフトウェアコンポーネント	最新の Rocky ベースイメージ (2025 年 5 月 30 日)
AWS コマンドラインインターフェイス (AWS CLI)	2.27.24
Amazon CloudWatch Agent	1.300055.1b1106-1
SSM Agent	3.3.2471.0-1
NICE DCV Server AppStream	2024.0.17598-1
Cloud-init	23.4 ~ 78
カーネル	4.18.0-553.54.1.el8_10.x86_64
NVIDIA Graphics Driver for G4dn、および G5 インスタンス	570.133.20
CUDA バージョン	12.8

Red Hat Enterprise Linux

ソフトウェアコンポーネント	最新のベースイメージ (2025 年 5 月 30 日)
AWS コマンドラインインターフェイス (AWS CLI)	2.27.24
Amazon CloudWatch Agent	1.300055.1b1106-1
SSM Agent	3.3.2471.0-1
NICE DCV Server AppStream	2024.0.17598-1
Cloud-init	23.4 ~ 78
カーネル	4.18.0-553.54.1.el8_10.x86_64

ソフトウェアコンポーネント	最新のベースイメージ (2025 年 5 月 30 日)
NVIDIA Graphics Driver for G4dn、および G5 インスタンス	570.133.20
CUDA バージョン	12.8

Important

次のパブリックイメージは AWS から廃止されたため、使用できなくなります。

- 2024 年 6 月 17 日より前にリリースされた 2016/2019/2022 Windows イメージ
- 2024 年 2 月以前にリリースされた Amazon Linux 2 イメージ
- Graphics Desktop インスタンスファミリー用イメージ

イメージをマルチセッションフリートで使用する場合、イメージは以下の条件を満たす必要があります。

- イメージは、2023 年 6 月 12 日以降にリリースされたベースイメージから作成する必要があります。または、イメージは、2023 年 9 月 6 日以降にリリースされたマネージド AppStream 2.0 イメージ更新を使用してイメージを更新する必要があります。詳細については、「[the section called “AppStream 2.0 のマネージド型イメージアップデートを使用してイメージを更新する”](#)」を参照してください。
- AppStream 2.0 エージェントのリリースバージョンは 09-06-2023 以降である必要があります。詳細については、「[the section called “ エージェントのバージョン管理”](#)」を参照してください。
- マネージド AppStream 2.0 イメージ更新を使用してイメージを更新した場合、AppStream 2.0 エージェントリリースバージョンは適用されません。イメージは 2023 年 9 月 6 日以降にリリースされたマネージドイメージ更新を使用して更新する必要があります。詳細については、「[the section called “AppStream 2.0 のマネージド型イメージアップデートを使用してイメージを更新する”](#)」を参照してください。
- マルチセッションフリートは、Microsoft Server 2019 および 2022 でのみサポートされています。

次の表で、リリースされたすべてのベースイメージを説明します。

リリース	プラットフォーム	イメージ	変更
05-30-2025	Windows	- 基本 - Graphics G4dn - Graphics G5	- 最新の NVIDIA ドライバーに更新 - Amazon DCV がバージョン 2024.0-19143 に更新されました - 新しい CloudWatch エージェント 1.4.37911 を含む - 新しい SSM エージェント 3.3.2299.0 を含める
05-30-2025	Red Hat Enterprise Linux	- 基本 - Graphics G4dn - Graphics G5	- 最新の NVIDIA ドライバーに更新 - Amazon DCV サーバーがバージョン 2024.0.17598-1 に更新されました - 新しい CloudWatch エージェント 1.300055.1-1 を含む - 新しい SSM エージェント 3.3.2471.0-1 を含める
05-30-2025	Rocky Linux	- 基本 - Graphics G4dn - Graphics G5	- 最新の NVIDIA ドライバーに更新 - Amazon DCV サーバーがバージョン 2024.0.17598-1 に更新されました - 新しい CloudWatch エージェント 1.300055.1-1 を含む - 新しい SSM エージェント 3.3.2471.0-1 を含める
02-11-2025	Amazon Linux 2	- 基本 - Graphics G4dn - Graphics G5	- 最新の CloudWatch エージェント 1.300041.1-1 を含む - 新しい SSM エージェント 3.3.1611.0-1 を含む
02-11-2025	Windows	- 基本 - Graphics G4dn - Graphics G5	- 新しい ENA および NVMe ドライバーを含む - 更新された PV ドライバーを含む

リリース	プラットフォーム	イメージ	変更
			新しいバージョンが利用可能なドライバーとソフトウェアの更新
12-19-2024	Rocky Linux	- 基本 - Graphics G4dn - Graphics G5	Rocky Linux 8 のサポート
10-22-2024	Amazon Linux 2	- 基本 - Graphics G4dn - Graphics Pro - Graphics G5	- Linux 用の最新の NVIDIA ドライバーが含まれています。 - Linux をバージョン 2.0.20241014 に更新しました。詳細については、「Amazon Linux 2 バージョン 2.0.20241014 リリースノート」を参照してください。
10-22-2024	Windows	- 基本 - グラフィックスデザイン - Graphics G4dn - Graphics Pro - Graphics G5	- Windows 用の最新の NVIDIA ドライバーが含まれています - Microsoft Visual C++ 2015-2022 再頒布可能 (x64) - 14.40.33816 を含む 2024 年 10 月 9 日までの Microsoft セキュリティ更新プログラムを含む
09-12-2024	Amazon Linux 2	- 基本 - Graphics G4dn - Graphics G5	- Android、iOS、および Surface Pro のタッチ機能に関連するバグ修正 - Amazon Linux 2 の G4dn および G5 インスタンス用の最新の NVIDIA Graphics ドライバー (550.90.07) が含まれています。 - Linux をバージョン 2.0.20240709.1 に更新。詳細については、「2.0.20240709.1」を参照してください。

リリース	プラットフォーム	イメージ	変更
07-30-2024	Red Hat Enterprise Linux	- 基本 - Graphics G4dn - Graphics G5	Red Hat Enterprise Linux 8 のサポートが含まれます。
06-17-2024	Windows	- 基本 - グラフィックスデザイン - Graphics G4dn - Graphics Pro - Graphics G5	2024 年 6 月 13 日までの Microsoft セキュリティ更新が含まれます。 - CloudWatch エージェント 1.4.37896 が含まれます。 - SSM Agent 3.3.484.0 が含まれます。 - AWS コマンドラインインターフェイス (AWS CLI) (WinServer 2016/2019) 1.33.9 が含まれます。 - AWS コマンドラインインターフェイス (AWS CLI) (WinServer 2022) 2.16.9.0 が含まれます。
05-08-2024	Windows	- 基本 - グラフィックスデザイン - Graphics G4dn - Graphics Pro - Graphics G5	2024 年 5 月までの Microsoft セキュリティ更新が含まれます。 - Windows Server 2016 および Windows Server 2019 用 Graphics Pro および G4dn インスタンス用の最新の NVIDIA Graphics ドライバー (552.08) が含まれます。 - CloudWatch エージェント 1.4.37891 が含まれます。 - SSM Agent 3.3.131.0-1 が含まれます。 - AWS コマンドラインインターフェイス (AWS CLI) 1.32.89 が含まれます。 - AWSVirtualSmartCardReader 1.0.0.59 が含まれます。

リリース	プラットフォーム	イメージ	変更
05-08-2024	リナックス	- 基本 - Graphics G4dn - Graphics Pro - Graphics G5	Linux をバージョン 2.0.20240412.0 に更新。詳細については、「Amazon Linux 2.0.20240412.0 リリースノート」を参照してください。
03-24-2024	Windows	- 基本 - グラフィックスデザイン - Graphics G4dn - Graphics Pro - Graphics G5	2024 年 3 月までの Microsoft セキュリティ更新が含まれます。 - Windows Server 2016 および Windows Server 2019 用 Graphics Pro および G4dn インスタンス用の最新の NVIDIA Graphics ドライバー (551.61) が含まれます。 - CloudWatch エージェント 1.3.50742 が含まれます。 - SSM Agent 3.2.2303.0 が含まれます。 - AWS コマンドラインインターフェイス (AWS CLI) 2.15.33.0 が含まれます。 - AWSVirtualSmartCardReader 1.0.0.59 が含まれます。
03-24-2024	リナックス	- 基本 - Graphics G4dn - Graphics Pro - Graphics G5	Linux をバージョン 2.0.20240318.0 に更新。詳細については、「Amazon Linux 2 version 2.0.20240318.0 release notes」を参照してください。
01-26-2024	Windows	- 基本 - グラフィックスデザイン - Graphics G4dn - Graphics Pro - Graphics G5	2024 年 1 月までの Microsoft セキュリティ更新プログラムが含まれます。

リリース	プラットフォーム	イメージ	変更
12-11-2023	Windows	- 基本 - Graphics G4dn - Graphics G5	Windows Server 2022 のサポートを追加
11-13-2023	Windows	- 基本 - グラフィックスデザイン - Graphics G4dn - Graphics Pro - Graphics G5	2023 年 11 月までの Microsoft セキュリティ更新プログラムを含む
11-13-2023	Amazon Linux 2	- 基本 - Graphics G4dn - Graphics Pro - Graphics G5	Linux をバージョン 2.0.20231101.0 に更新。詳細については、「Amazon Linux 2.0.20231101.0 リリースノート」を参照してください。
06-12-2023	Windows	- 基本 - グラフィックスデザイン - Graphics G4dn - Graphics Pro	2023 年 6 月までの Microsoft セキュリティ更新が含まれます
06-11-2023	Amazon Linux 2	- 基本 - Graphics G4dn - Graphics Pro	Linux をバージョン 2.0.20230530.0 に更新しました。詳細については、「Amazon Linux 2 2.0.20230530.0 リリースノート」を参照してください。
03-29-2023	Windows	- 基本 - グラフィックスデザイン - Graphics G4dn - Graphics Pro	2023 年 2 月までの Microsoft セキュリティ更新が含まれます

リリース	プラットフォーム	イメージ	変更
03-15-2023	Amazon Linux 2	- 基本 - Graphics G4dn - Graphics Pro	- Linux をバージョン 2.0.20220805.0 に更新しました。詳細については、「Amazon Linux 2 2.0.20230221.0 リリースノート」を参照してください。 - ウェブカメラの体験を向上しました。 - システム暗号化が FIPS 準拠のアルゴリズムを使用するように設定されている場合に AppStream 2.0 フリートインスタンスがプロビジョニングできない問題を解決します。
10-05-2022	Windows	- 基本 - グラフィックスデザイン - Graphics G4dn - Graphics Pro	2021 年 9 月 13 日までの Microsoft セキュリティ更新プログラムを含む
09-21-2022	Amazon Linux 2	- 基本 - Graphics G4dn - Graphics Pro	- Linux をバージョン 2.0.20220805.0 に更新しました。詳細については、「Amazon Linux 2.0.20220805.0 リリースノート」を参照してください。 - Image Assistant GUI が含まれます - ウェブカメラのサポートが含まれます
09-14-2022	Amazon Linux 2	- Graphics G4dn - Graphics Pro	NVIDIA Graphics ドライバー (510.85.02) を含む
09-01-2022	Windows	- Graphics G4dn - Graphics Pro	- Windows Server 2012 R2 用の NVIDIA Graphics ドライバー (473.47) を含む - Windows Server 2016 および Windows Server 2019 用の NVIDIA Graphics ドライバー (512.78) を含む

リリース	プラットフォーム	イメージ	変更
07-12-2022	Windows	- 基本 - グラフィックスデザイン - Graphics G4dn - Graphics Pro	2022 年 6 月 14 日までの Microsoft セキュリティ更新が含まれます - Windows Server 2016 および Windows Server 2019 の Graphics Design インスタンス用の最新の AMD ドライバー (24.20.13028.7002) を含む - Graphics Pro 用の最新の NVIDIA Graphics ドライバー (472.98) および Windows Server 2012R2 用の G4dn インスタンスを含む - Windows Server 2016 および Windows Server 2019 用 Graphics Pro および G4dn インスタンス用の最新の NVIDIA Graphics ドライバー (511.65) を含む - CloudWatch Agent 1.3.50742 を含む - SSM Agent 3.1.1575.0 を含む - AWS コマンドラインインターフェイス (AWS CLI) 1.23.11 を含む
06-20-2022	Amazon Linux 2	- 基本 - Graphics G4dn - Graphics Pro	Linux をバージョン 2.0.20220426.0 に更新しました。詳細については、「Amazon Linux 2.0.20220426.0 リリースノート」を参照してください。
03-03-2022	Windows	- 基本 - グラフィックスデザイン - Graphics G4dn - Graphics Pro	2022 年 1 月 11 日までの Microsoft セキュリティ更新プログラムが含まれます

リリース	プラットフォーム	イメージ	変更
02-18-2022	Amazon Linux 2	- 基本 - Graphics G4dn - Graphics Pro	- Linuxをバージョン 2.0.20211223.0 に更新しました。詳細については、「Amazon Linux 2.0.20211223.0 リリースノート」を参照してください。 - 最新の Linux ベースイメージ
11-19-2021	Amazon Linux 2	- 基本 - Graphics G4dn - Graphics Pro	最新の Linux ベースイメージ (スモールインスタンスタイプにおける空白画面の修正を含む)
11-15-2021	Amazon Linux 2	- 基本 - Graphics G4dn - Graphics Pro	Linux ベースイメージ
10-08-2021	Windows	- 基本 - グラフィックスデザイン - Graphics G4dn - Graphics Pro - サンプルアプリ	2021 年 9 月 15 日までの Microsoft セキュリティ更新プログラムを含む - AWS Tools for PowerShell をバージョン 3.15.1398 に更新
07-19-2021	Windows	- 基本 - グラフィックスデザイン - Graphics G4dn - Graphics Pro	2021 年 7 月 13 日までの Microsoft Windows 更新プログラムを含む
06-01-2021	Windows	- 基本 - グラフィックスデザイン - Graphics G4dn - Graphics Pro	2021 年 4 月 14 日までの Microsoft Windows 更新プログラムを含む - Graphics Design インスタンス用の AMD ドライババージョン 24.20.13028.5012 を含む

リリース	プラットフォーム	イメージ	変更
2020 年 12 月 28 日	Windows	- 基本 - グラフィックスデザイン - Graphics G4dn - Graphics Pro	- スマートカードの使用をサポートするドライバを含む スマートカードは、Windows のサインイン、Active Directory に参加しているストリーミングインスタンス、およびストリーミングアプリケーションのセッション内認証などに使用できます 2020 年 12 月 9 日までの Microsoft Windows 更新プログラム込み - CLI AWS バージョン 1.18.138 を含む - Graphics Pro および Graphics G4dn インスタンス用の NVIDIA Graphics Driver バージョン 451.48 を含む
2020 年 7 月 16 日	Windows	- 基本 - グラフィックスデザイン - Graphics G4dn - Graphics Pro	2020 年 6 月 9 日までの Microsoft Windows 更新プログラム込み - CLI AWS バージョン 1.18.86 を含む - NVIDIA Graphics Driver version 441.66 for Graphics Pro instances を含む
04-22-2020	Windows	- Base (Windows Server 2019) - Graphics Design (Windows Server 2019) - Graphics G4dn (Windows Server 2019) - Graphics Pro (Windows Server 2019)	2020 年 3 月 10 日までの Microsoft Windows 更新プログラムが含まれます。 - CLI AWS バージョン 1.18.21 を含む - NVIDIA Graphics Driver version 441.66 for Graphics Pro instances を含む

リリース	プラットフォーム	イメージ	変更
03-18-2020	Windows	- 基本 - グラフィックスデザイン - Graphics Pro	2020 年 2 月 11 日までの Microsoft Windows 更新プログラムを含む - CLI AWS バージョン 1.17.5 を含む - Graphics Pro インスタンス用の NVIDIA Graphics Driver バージョン 412.16 を含む
03-16-2020	Windows	Graphics G4dn	- EC2 G4dn ファミリーに基づく Graphics G4dn インスタンスのサポートを追加 (Windows Server 2012 R2) 2020 年 2 月 11 日までの Microsoft Windows 更新プログラムを含む - CLI AWS バージョン 1.17.5 を含む
03-05-2020	Windows	Graphics G4dn	- EC2 G4dn ファミリーに基づく Graphics G4dn インスタンスのサポートを追加 (Windows Server 2016 および Windows Server 2019) 2020 年 2 月 11 日までの Microsoft Windows 更新プログラムを含む - CLI AWS バージョン 1.17.5 を含む
01-13-2020	Windows	グラフィックスデザイン	Windows Server 2019 のサポートを追加。Microsoft Windows の 2019 年 11 月 12 日までの更新プログラムを含む

リリース	プラットフォーム	イメージ	変更
12-12-2019	Windows	- 基本 - グラフィックスデザイン - Graphics Pro	2019 年 11 月 12 日までの Microsoft Windows 更新プログラムを含む - CLI AWS バージョン 1.16.284 を含む - SSM Agent の新しいバージョン (v2.3.760.0) が含まれています。これにより、ストリーミングインスタンスのプロビジョニングが妨げられる問題が解決されます。
09-18-2019	Windows	- 基本 - グラフィックスデザイン - Graphics Pro	- すべての Base インスタンスと Graphics Pro インスタンス、および Graphics Design Windows Server 2012 R2 用の、2019 年 8 月 13 日までの Microsoft Windows 更新プログラムが含まれています。Graphics Design Windows Server 2016 インスタンスには、すでにこのバージョンが含まれています。 - すべての Base および Graphics Pro インスタンスと Graphics Design Windows Server 2012 R2 用の AWS CLI バージョン 1.16.222 が含まれています。R2 Graphics Design Windows Server 2016 インスタンスには、すでにこのバージョンが含まれています。 - Windows Server 2016 および Windows Server 2019 イメージビルダーインスタンスで Windows Defender がデフォルトで有効にならないようにする修正が含まれています。詳細については、「Amazon AppStream 2.0 での Windows Update とウイルス対策ソフトウェア」を参照してください。

リリース	プラットフォーム	イメージ	変更
2019-09-05	Windows	グラフィックスデザイン	- Windows Server 2016 のサポートの追加 2019 年 8 月 13 日までの Microsoft Windows 更新プログラムが含まれます。 - CLI AWS バージョン 1.16.222 を含む - Graphics Design インスタンス用の AMD Driver バージョン 24.20.13028.3002 を含む (Windows Server 2016 と互換)
06-24-2019	Windows	- 基本 - Graphics Pro	Windows Server 2016 および Windows Server 2019 のサポートを追加しました
05-28-2019	Windows	- 基本 - グラフィックスデザイン - Graphics Pro	2019 年 5 月 14 日までの Microsoft Windows 更新プログラムが含まれます。
04-29-2019	Windows	- 基本 - グラフィックスデザイン - Graphics Pro	2019 年 4 月 20 日までの Microsoft Windows 更新プログラムが含まれます。 - CLI AWS バージョン 1.16.126 を含む - Graphics Pro インスタンス用 NVIDIA グラフィックドライバ 412.16 が含まれます。
01-22-2019	Windows	- 基本 - グラフィックスデザイン - Graphics Pro	2018 年 12 月 10 日までの Microsoft Windows 更新プログラムが含まれます。 - CLI AWS バージョン 1.16.84 を含む - Graphics Pro インスタンス用の NVIDIA Graphics Driver バージョン 391.58 を含む

リリース	プラットフォーム	イメージ	変更
06-12-2018	Windows	- 基本 - グラフィックスデザイン - Graphics Desktop - Graphics Pro	2018 年 5 月 9 日までの Microsoft Windows 更新プログラムが含まれます。 - Windows PowerShell 5.1 が含まれます。
05-02-2018	Windows	- 基本 - グラフィックスデザイン - Graphics Desktop - Graphics Pro	2018 年 4 月 10 日までの Microsoft Windows 更新プログラムが含まれます。 - 次の言語パックが追加されました: 日本語、韓国語、ポルトガル語 (ブラジル)、タイ語、中国語 (簡体字)、中国語 (繁体字)
03-19-2018	Windows	- 基本 - グラフィックスデザイン - Graphics Desktop - Graphics Pro	2018 年 2 月 23 日までの Microsoft Windows 更新プログラムが含まれます。 - 次の言語パックが含まれています。ドイツ語、フランス語、イタリア語、スペイン語、オランダ語 - ストリーミングセッション中に Microsoft Visio と Microsoft Project アプリケーションを使用した際の断続的な問題を解決
01-24-2018	Windows	- 基本 - グラフィックスデザイン - Graphics Desktop - Graphics Pro	2018 年 1 月 5 日までの Microsoft Windows 更新プログラムが含まれます - Spectre と Meltdown の脆弱性に関する Microsoft Windows の更新が含まれています - Image Builder でデフォルトのプロファイルを作成し、ストリーミングセッション中にコマンドラインインターフェイス (CLI) AWS に使用できるようにします。

リリース	プラットフォーム	イメージ	変更
01-01-2018	Windows	- 基本 - グラフィックスデザイン - Graphics Desktop - Graphics Pro	AppStream 2.0 インスタンスへの接続の問題を解決します
12-07-2017	Windows	- 基本 - グラフィックスデザイン - Graphics Desktop - Graphics Pro	2017 年 11 月 19 日までの Microsoft Windows 更新プログラムが含まれます。 - 管理対象の AppStream 2.0 エージェントアップデートのサポートを追加
11-13-2017	Windows	基本	- ストリーミングセッション中に機能しない Microsoft Office 365 アプリケーションに関する問題を解決します 2017 年 10 月 11 日までの Microsoft Windows 更新プログラムが含まれます。
09-05-2017	Windows	- 基本 - グラフィックスデザイン - Graphics Desktop - Graphics Pro	- 新しいグラフィックデザインインスタンスファミリー - オンデマンドフリートのサポート - セッションコンテキストのアプローチの更新 2017 年 8 月 9 日までの Microsoft Windows 更新プログラムが含まれます。 - フォアグラウンドに現れないアプリケーションの断続的な問題を解決します。 - タイトルビューに表示されないアプリケーションの断続的な問題を解決します。

リリース	プラットフォーム	イメージ	変更
07-25-2017	Windows	- Graphics Desktop - Graphics Pro	- 新しい Graphics Desktop および Graphics Pro インスタンスファミリー 2K 解像度のサポートを追加
07-24-2017	Windows	基本	2017 年 7 月 13 日までの Microsoft Windows 更新プログラムが含まれます。 - Microsoft Active Directory ドメインのサポートが追加されています。
06-20-2017	Windows	- 基本 - サンプルアプリ	- アプリケーションの起動パフォーマンスの最適化 - タイトルビューに表示されないアプリケーションの問題を解決 - タイトルビューにのみ表示されるアプリケーションの問題を解決 - タイトルビューに複数回表示されるアプリケーションの問題を解決 - 最近起動されたアプリケーションウィンドウがフォアグラウンドに表示されない問題を解決 - 印刷時のページ余白の問題を解決
05-18-2017	Windows	- 基本 - サンプルアプリ	- AppStream 2.0 ホームフォルダのサポート追加 2017 年 5 月 16 日までの Microsoft Windows 更新プログラムが含まれます。 - ストリーミングインスタンスからのインターネット接続に影響を与える、断続的なネットワークの問題を解決 - アプリケーションタイルが正しく機能しない問題を解決

イメージ

ユーザーにストリーミングできるアプリケーションと、ユーザーがこれらのアプリケーションの使用をすばやく開始できるようにするデフォルトのシステム設定とアプリケーション設定が含まれる Amazon AppStream 2.0 イメージを作成できます。ただし、作成後のイメージは変更できません。他のアプリケーションの追加、既存のアプリケーションの更新、またはイメージ設定の変更を行うには、イメージの作成に使用した Image Builder を起動して再接続する必要があります。この Image Builder が削除済みである場合は、イメージに基づく新しい Image Builder を起動します。次に、変更を反映して新しいイメージを作成します。詳細については、[Image Builder を起動し、ストリーミングアプリケーションをインストールして設定する](#)および[チュートリアル: AppStream 2.0 コンソールを使用してカスタム AppStream 2.0 イメージを作成する](#)を参照してください。

利用可能なイメージは、AppStream 2.0 コンソールの [Image Registry (イメージレジストリ)] セクションに表示されます。これらは、パブリック、プライベート、共有として分類されています。これらの各イメージタイプを使用して Image Builder を起動し、AppStream 2.0 フリートをセットアップできます。共有イメージは、他の Amazon Web Services アカウントによって所有され、ユーザーと共有されるイメージです。自分と共有されているイメージに設定されているアクセス許可により、これらのイメージに対する操作が制限される場合があります。詳細については、「[Amazon AppStream 2.0 画像を管理する](#)」を参照してください。

内容

- [Amazon AppStream 2.0 でのデフォルトのアプリケーションおよび Windows 設定とアプリケーション起動パフォーマンス](#)
- [AppStream 2.0 エージェントのバージョンを管理する](#)
- [AppStream 2.0 エージェントのリリースノート](#)
- [チュートリアル: AppStream 2.0 コンソールを使用してカスタム AppStream 2.0 イメージを作成する](#)
- [Amazon AppStream 2.0 画像を管理する](#)
- [Image Assistant CLI オペレーションを使用してプログラムで Amazon AppStream 2.0 イメージを作成する](#)
- [Linux ベースのイメージを作成する](#)
- [セッションスクリプトを使用して Amazon AppStream 2.0 ユーザーのストリーミングエクスペリエンスを管理する](#)

Amazon AppStream 2.0 でのデフォルトのアプリケーションおよび Windows 設定とアプリケーション起動パフォーマンス

デフォルトのアプリケーションおよび Windows 設定を作成すると、ユーザーはすぐにアプリケーションの使用を開始できます。ユーザー自身が設定を作成または構成する必要はありません。

AppStream 2.0 は、ユーザーのストリーミングセッションで、アプリケーションの起動パフォーマンスを最適化します。このプロセスにすべての必須ファイルを含めるには、最適化のマニフェストに従って特定のファイルやフォルダを手動で追加する必要があります。

内容

- [AppStream 2.0 ユーザー向けのデフォルトのアプリケーションおよび Windows 設定を作成する](#)
- [Amazon AppStream 2.0 でのアプリケーションの起動パフォーマンスの最適化](#)

AppStream 2.0 ユーザー向けのデフォルトのアプリケーションおよび Windows 設定を作成する

Windows ユーザープロファイルフォルダやユーザーレジストリハブに保存されているアプリケーションのカスタマイズおよび Windows 設定をデフォルトとして設定できます。Image Assistant の [Template User (テンプレートユーザー)] を使用してデフォルト設定を保存すると、AppStream 2.0 で Windows のデフォルトユーザープロファイルが、保存した設定のプロファイルに置き換えられます。次に、この Windows デフォルトユーザープロファイルを使用して、フリートインスタンスにユーザーの初期設定が作成されます。設定したアプリケーションまたは Windows 設定がフリートで機能しない場合は、設定の保存先が Windows ユーザープロファイルになっていることを確認してください。詳細については、[チュートリアル: AppStream 2.0 コンソールを使用してカスタム AppStream 2.0 イメージを作成する](#) の Step 3: Create Default Application and Windows Settings を参照してください。

デフォルト設定として作成および設定できる項目は、以下のとおりです。

- アプリケーション設定 (ブラウザのホームページ、ツールバーのカスタマイズ、セキュリティの設定など)。
- アプリケーションデータ設定 (ブラウザブックマーク、接続プロファイルなど)。
- Windows エクスペリエンスの設定 (ファイル拡張子の表示や非表示のフォルダなど)。

さらに、Internet Explorer のセキュリティ設定 (ESC: セキュリティ強化の構成など) を変更または無効化できます。詳細については、「[Amazon AppStream 2.0 で Internet Explorer 拡張セキュリティ設定を無効にする](#)」を参照してください。

Amazon AppStream 2.0 でのアプリケーションの起動パフォーマンスの最適化

イメージを作成すると、ユーザーのストリーミングセッションでアプリケーションの起動パフォーマンスを最適化することを AppStream 2.0 から要求されます。このプロセス中にアプリケーションが開いている場合は、ユーザーに必要な初期コンポーネントがアプリケーションで使用されていることを確認します。これにより、これらのコンポーネントは最適化プロセスでキャプチャされます。場合によっては、最適化に必要なファイルの一部が検出されないことがあります。これらのファイルの例として、Image Builder で開いていないプラグインやコンポーネントなどが挙げられます。アプリケーションに必要なすべてのファイルを確実にキャプチャするには、これらを最適化のマニフェストに含めることができます。ファイルを最適化のマニフェストに追加すると、フリートインスタンスを作成してユーザーに公開するまでの所要時間が長くなる場合があります。ただし、こうすることで、アプリケーションがフリートインスタンスで最初に起動されるまでにかかる時間が短縮されます。

フォルダ内のすべてのファイルを最適化するには、PowerShell を開いて次の PowerShell コマンドを使用します。

```
dir -path "C:\Path\To\Folder\To\Optimize" -Recurse -ErrorAction SilentlyContinue |  
%{$_.FullName} | Out-File "C:\ProgramData\Amazon\Photon\Prewarm\PrewarmManifest.txt" -  
encoding UTF8 -append
```

デフォルトでは、Image Assistant の最適化ステップを実行するたびに、アプリケーション最適化マニフェストが置き換えられます。フォルダ内のすべてのファイルを最適化するには、PowerShell コマンドを実行する必要があります。

- 最適化ステップが実行されるたびに。
- [Disconnect and create image (接続を解除してイメージを作成する)] を選択する前に、Image Assistant [Review (確認)] ページでイメージを作成します。

または、Image Assistant のコマンドラインインターフェイス (CLI) オペレーションを使用して、アプリケーションごとに最適化マニフェストを指定できます。Image Assistant の CLI 操作を使用して最適化マニフェストを指定すると、AppStream 2.0 は、指定されたアプリケーション最適化マニフェストを Image Assistant の最適化ステップで識別されたファイルにマージします。詳細については、

「[Image Assistant CLI オペレーションを使用してプログラムで Amazon AppStream 2.0 イメージを作成する](#)」を参照してください。

AppStream 2.0 エージェントのバージョンを管理する

AppStream 2.0 エージェントは、ストリーミングインスタンスで実行され、ユーザーによるアプリケーションのストリーミングを可能にするソフトウェアです。新しいイメージを作成すると、[Always use latest agent version (常に最新バージョンのエージェントを使用)] オプションがデフォルトで選択されます。このオプションが選択されていると、イメージから起動される新しい Image Builder またはフリートインスタンスでは常に最新バージョンの AppStream 2.0 エージェントが使用されます。エンドユーザーにデプロイする前に、エージェントの更新を制御して、ソフトウェアとの互換性を確保したり、更新された環境を認定したりできます。

以下の手順では、AppStream 2.0 エージェントのバージョンを管理する方法を示します。

内容

- [常に最新バージョンの AppStream 2.0 エージェントを使用するイメージを作成する](#)
- [特定バージョンの AppStream 2.0 エージェントを使用するイメージを作成する](#)
- [より新しいバージョンの AppStream 2.0 エージェントを使用するイメージを作成する](#)

常に最新バージョンの AppStream 2.0 エージェントを使用するイメージを作成する

イメージが常に最新の AppStream 2.0 エージェントバージョンを使用するように設定されている場合、ストリーミングインスタンスは、新しいエージェントバージョンがリリースされた AWS ときから利用可能な最新の機能、パフォーマンスの向上、およびセキュリティ更新プログラムで自動的に更新されます。

Note

場合によっては、新しいバージョンの AppStream 2.0 エージェントがご使用のソフトウェアと衝突することがあります。新しいバージョンの AppStream 2.0 エージェントを稼働フリートにデプロイする前に、そのバージョンを認定することをお勧めします。

常に最新バージョンの AppStream 2.0 エージェントを使用するイメージを作成するには

1. AppStream 2.0 コンソール (<https://console.aws.amazon.com/appstream2>) を開きます。
2. 次のいずれかを実行します。
 - 既存の Image Builder を使用してイメージを作成する場合は、この Image Builder を起動して接続します。Image Builder で最新バージョンの AppStream 2.0 エージェントが実行されていない場合は、最新のエージェントで Image Builder を起動するかどうかを確認するメッセージが表示されます。このオプションを必ず選択し、[Start (起動)] を選択して、Image Builder に接続します。
 - イメージの作成に使用できる Image Builder がない場合は、新しい Image Builder を起動します。ステップ 1: イメージ で、AWS ベースイメージまたはカスタムイメージを選択します。[Step 2: Configure Image Builder] では、選択したイメージで最新バージョンの AppStream 2.0 エージェントが実行されていない場合、[AppStream 2.0] セクションが表示されます。[Agent version (エージェントのバージョン)] リストで、エージェントの最新バージョンを選択します。残りのステップを完了して Image Builder を作成し、これに接続します。詳細については、「[Image Builder を起動し、ストリーミングアプリケーションをインストールして設定する](#)」を参照してください。
3. Image Builder のデスクトップで、Image Assistant を開き、手順に従って新しいイメージを作成します。[Configure Image (イメージを設定する)] ステップで、[Always use the latest agent version (常に最新バージョンのエージェントを使用)] が選択されていることを確認します。詳細については、「[チュートリアル: AppStream 2.0 コンソールを使用してカスタム AppStream 2.0 イメージを作成する](#)」を参照してください。

常に最新バージョンの AppStream 2.0 エージェントを使用する指定を後で取り消す場合は、新しいイメージを作成して、このオプションをオフにする必要があります。
4. 新しいフリートを作成するか、既存のフリートを変更します。フリートを設定するときに、作成した新しいイメージを選択します。詳細については、「[Amazon AppStream 2.0 フリートとスタックを作成する](#)」を参照してください。
5. 新しいスタックを作成するか、既存のスタックを変更して、フリートと関連付けます。

特定バージョンの AppStream 2.0 エージェントを使用するイメージを作成する

互換性テストを最初に実行する場合は、常に最新バージョンを使用する代わりに、AppStream 2.0 エージェントの更新を管理できます。使用している AppStream 2.0 エージェントのバージョンがス

ストリーミングアプリケーションと互換性があることを確認するには、特定バージョンのエージェントソフトウェアを使用するイメージを作成できます。次に、適格性確認テストを別個のフリートで実行し、その後で本番稼働用フリートにデプロイします。

新しいイメージを作成するときに、[Always use latest agent version (常に最新バージョンのエージェントを使用)] オプションが選択されていないことを確認します。これにより、常に最新バージョンを使用する代わりに、Image Builder の起動時に選択したバージョンの AppStream 2.0 エージェントがイメージで使用されます。適格性確認テストが完了したら、本番稼働用フリートを更新してイメージを反映できます。

特定バージョンの AppStream 2.0 エージェントを使用するイメージを作成するには

1. AppStream 2.0 コンソール (<https://console.aws.amazon.com/appstream2>) を開きます。
 2. 次のいずれかを実行します。
 - 既存の Image Builder を使用してイメージを作成する場合は、この Image Builder を起動して接続します。
 - イメージの作成に使用できる Image Builder がない場合は、新しい Image Builder を起動します。ステップ 1: イメージ で、AWS ベースイメージまたはカスタムイメージを選択します。[Step 2: Configure Image Builder] では、選択したイメージで最新バージョンの AppStream 2.0 エージェントが実行されていない場合、[AppStream 2.0] セクションが表示されます。[Agent version (エージェントのバージョン)] リストで、エージェントの最新バージョンを選択しません。残りのステップを完了して Image Builder を作成し、これに接続します。詳細については、「[Image Builder を起動し、ストリーミングアプリケーションをインストールして設定する](#)」を参照してください。
 3. Image Builder のデスクトップで、Image Assistant を開き、手順に従って新しいイメージを作成します。Image Assistant の [Configure Image (イメージを設定する)] ステップで、[Always use the latest agent version (常に最新バージョンのエージェントを使用)] が選択されていないことを確認します。詳細については、「[チュートリアル: AppStream 2.0 コンソールを使用してカスタム AppStream 2.0 イメージを作成する](#)」を参照してください。
- 常に最新バージョンの AppStream 2.0 エージェントを使用するように後で指定する場合は、新しいイメージを作成し、このオプションを選択する必要があります。
4. 新しいフリートを作成するか、既存のフリートを変更します。フリートを設定するときに、作成した新しいイメージを選択します。詳細については、「[Amazon AppStream 2.0 フリートとスタックを作成する](#)」を参照してください。
 5. 新しいスタックを作成するか、既存のスタックを変更して、フリートと関連付けます。

6. フリートに接続して、アプリケーションの互換性をテストします。

より新しいバージョンの AppStream 2.0 エージェントを使用するイメージを作成する

特定バージョンの AppStream 2.0 エージェントを使用するようにイメージを設定した場合、より新しいバージョンに更新するには、新しいイメージを作成する必要があります。これにより、エージェントの各更新の互換性をテストしてから、フリートの増分更新を行うことができます。

新しいイメージを作成するときに、[Always use latest agent version (常に最新バージョンのエージェントを使用)] オプションが選択されていないことを確認します。イメージを作成したら、適格性確認テストを別個のフリートで実行し、その後で本番稼働用フリートにデプロイします。適格性確認テストが完了したら、本番稼働用フリートを更新してイメージを反映できます。

より新しいバージョンの AppStream 2.0 エージェントを使用するイメージを作成するには

1. AppStream 2.0 コンソール (<https://console.aws.amazon.com/appstream2>) を開きます。
2. 次のいずれかを実行します。
 - 既存の Image Builder を使用してイメージを作成する場合は、この Image Builder を起動して接続します。Image Builder で最新バージョンの AppStream 2.0 エージェントが実行されていない場合は、最新のエージェントで Image Builder を起動するかどうかを確認するメッセージが表示されます。このオプションを必ず選択し、[Start (起動)] を選択して、Image Builder に接続します。
 - イメージの作成に使用できる Image Builder がない場合は、新しい Image Builder を起動します。ステップ 1: イメージ で、AWS ベースイメージまたはカスタムイメージを選択します。[Step 2: Configure Image Builder] では、選択したイメージで最新バージョンの AppStream 2.0 エージェントが実行されていない場合、[AppStream 2.0] セクションが表示されます。[Agent version (エージェントのバージョン)] リストで、エージェントの最新バージョンを選択します。残りのステップを完了して Image Builder を作成し、これに接続します。詳細については、「[Image Builder を起動し、ストリーミングアプリケーションをインストールして設定する](#)」を参照してください。
3. Image Builder のデスクトップで、Image Assistant を開き、手順に従って新しいイメージを作成します。Image Assistant の [Configure Image (イメージを設定する)] ステップで、[Always use the latest agent version (常に最新バージョンのエージェントを使用)] が選択されていないことを確認します。詳細については、「[チュートリアル: AppStream 2.0 コンソールを使用してカスタム AppStream 2.0 イメージを作成する](#)」を参照してください。

常に最新バージョンの AppStream 2.0 エージェントを使用するように後で指定する場合は、新しいイメージを作成し、このオプションを選択する必要があります。

4. 新しいフリートを作成するか、既存のフリートを変更します。フリートを設定するときに、作成した新しいイメージを選択します。詳細については、「[Amazon AppStream 2.0 フリートとスタックを作成する](#)」を参照してください。
5. 新しいスタックを作成するか、既存のスタックを変更して、フリートと関連付けます。
6. フリートに接続して、アプリケーションの互換性をテストします。

AppStream 2.0 エージェントのリリースノート

Amazon AppStream 2.0 エージェントソフトウェアは、ストリーミングインスタンスで実行され、エンドユーザーがストリーミングアプリケーションに接続して起動できるようにします。2017 年 12 月 7 日以降、ストリーミングインスタンスは、利用可能な最新の機能、パフォーマンスの向上、およびセキュリティ更新プログラムで自動的に更新できます AWS。2017 年 12 月 7 日より前は、新しいベースイメージリリースにエージェントの更新プログラムが含まれていました。

最新の AppStream 2.0 エージェントソフトウェアを使用するには、2017 年 12 月 7 日以降 AWS によって公開された新しいベースイメージを使用してイメージを再構築する必要があります。この操作を行う場合、エージェントの自動更新を有効にするオプションが Image Assistant でデフォルトで選択されます。このオプションを選択したままにし、新しい Image Builder またはイメージから起動される新しいフリートインスタンスで、常に最新バージョンのエージェントが表示されるようにすることをお勧めします。詳細については、「[チュートリアル: AppStream 2.0 コンソールを使用してカスタム AppStream 2.0 イメージを作成する](#)」を参照してください。

以下の表には、Windows インスタンス、Amazon Linux インスタンス、および Red Hat Enterprise Linux インスタンス向け AppStream 2.0 エージェントのリリース済みバージョンで利用できる最新更新の説明が記載されています。

Windows

Amazon AppStream 2.0 エージェントのバージョン	変更
05-29-2025	マルチセッションフリートのホームフォルダ/一時フォルダへのクイックリンクアクセスの問題を解決します。

Amazon AppStream 2.0 エージェントのバージョン	変更
	- マルチセッション AppStream 2.0 フリートでの証明書ベースの認証の証明書検証処理の問題を解決しました。 - ディスプレイ解像度の品質が向上しました。 - 一般的なバグ修正と機能強化
03-05-2025	- Windows FIPS でのインスタンス起動の問題を解決 - プライベート Amazon S3 インターフェイスエンドポイントでのアプリケーション設定の永続化に関する問題を解決 - 一般的なバグ修正と機能強化
02-07-2025	- マルチセッション証明書ベースの認証のサポートを追加 - マルチセッションフリートのパフォーマンス強化を追加 - User Account Control (UAC) Virtualization を使用する特定のレガシーアプリケーションが、アプリケーション設定の永続化の使用時にアクセス許可の問題が発生する問題を修正
10-31-2024	マルチセッションフリートのバグ修正と改善
10-21-2024	Microsoft Visual C++ Redistributable (14.38.33135.0 より新しいバージョン) で使用するとインスタンスの起動に失敗する AppStream 2.0 エージェントバージョン 09-18-2024 の問題を解決

Amazon AppStream 2.0 エージェントのバージョン	変更
09-18-2024	- リージョン設定は、マルチセッションフリートをサポートします。エンドユーザーは、ストリーミングセッションのタイムゾーン、ロケール、入力方法をカスタマイズできます。 - マルチセッションフリートのプリンターリダイレクトサポート。エンドユーザーは、ユーザーがマッピングしたネットワークプリンターを含め、各自のローカルコンピュータに接続されているプリンターに、ストリーミングアプリケーションからプリントジョブをリダイレクトできます。 - Windows のアプリケーションおよびデスクトップストリーミングセッションの自動タイムゾーンリダイレクト (単一セッションまたはマルチセッション)。AppStream 2.0 管理者は、エンドユーザーのストリーミングセッションのタイムゾーンリダイレクトを有効または無効にできます。有効にすると、エンドユーザーはローカルデバイスと AppStream 2.0 セッションで同じタイムゾーン設定を確認できます。 - ユーザープロファイル VHD の新しいデフォルトの最大サイズは、常時オンフリートとオンデマンドフリートでは 5 GB です - Desktop View による常時およびオンデマンドフリートの MSIX アプリケーションサポート - 一般的なバグ修正と機能強化
05-21-2024	- マルチセッションフリートでのオーディオのサポート - アプリケーションビューフリートの安定性の向上 - AD 参加マルチセッションフリートでのアクティブディレクトリの信頼関係のサポート - 一般的なバグ修正と機能強化

Amazon AppStream 2.0 エージェントのバージョン	変更
04-15-2024	• アプリケーション設定の永続化が有効になっている場合のストリーミングの耐障害性を改善 • マルチセッションフリートのシームレスモード/ネイティブアプリケーションモードのサポートを追加 • マルチセッションストリーミングでのマウスカーソルのエンドユーザーエクスペリエンスを改善
01-17-2024	• マルチセッションフリートでのオーディオ出力のサポートを追加 • マルチセッションフリートでのセッションスクリプトのサポートを追加 • マルチセッションフリートのプロビジョニングレジリエンスを改善
12-07-2023	• Windows Server 2022 のサポートを追加 • Windows Server 2019 のストリーミングパフォーマンスを改善 • AWS CLI v2 サポートを追加 • アプリケーション間を切り替えるためのキーボードサポートを追加 • Windows セッションがロックされている場合の証明書ベースの認証に関する問題を解決 • 注: Windows Server 2012 R2 は、2023 年 10 月 10 日にサポートが終了しました。ストリーミングエクスペリエンスのサポートを強化するには、Windows Server 2016、Windows Server 2019、または Windows Server 2022 にアップグレードしてください。

Amazon AppStream 2.0 エージェントのバージョン	変更
09-06-2023	- マルチセッションフリートのサポートを追加 - インスタンスとセッションのプロビジョニングを改善しました - コピー/貼り付け機能に関する問題を解決しました - 以下のソフトウェアコンポーネントが必要です。 - Microsoft .NET Framework Runtime — 4.7.2
05-30-2023	インスタンスのプロビジョニングの耐障害性が向上
05-08-2023	Windows 2016 および Windows 2012 R2 のフリートインスタンスでシャットダウン警告が表示される問題を解決します
04-13-2023	ストリーミングセッションが接続状態でスタックする問題を解決しました
03-21-2023	- アプリケーションがフリーズする問題を解決しました。 - 物理スマートカードの認証エラーの問題を解決しました。 - Windows で FIPS が有効になっていると、ホームフォルダーが機能しないという問題を解決しました - インスタンスのプロビジョニングの耐障害性が向上 - Windows Server 2019 の物理スマートカードによるログオン時間のパフォーマンスを向上しました
10-13-2022	- エージェントのパフォーマンスを改善 - DCV 物理スマートカードの問題を解決しました
06-20-2022	- 古いイメージの USB 文字列フィルターファイルの場所に対する下位互換性を追加 - インスタンスのプロビジョニングの耐障害性が向上 - セッション接続の信頼性が向上

Amazon AppStream 2.0 エージェントのバージョン	変更
03-14-2022	リージョンの設定が更新されない問題を解決しました
02-21-2022	- Microsoft OneDrive が大きなファイルをコピーする際の問題を解決します - 小さなインスタンスタイプでのエージェントの堅牢性が向上 - 以下のソフトウェアコンポーネントで動作します。詳細については、「the section called “ベースイメージおよびマネージド型イメージの更新版リリースノート”」を参照してください。 - Amazon SSM Agent — 3.0.1295.0 - Amazon WDDM Hook ドライバー — 1.0.0.56 (Windows Server 2012 R2) - NICE DCV 仮想ディスプレイ — 1.0.34.0 (Windows Server 2016/2019) - EC2Config サービス (Windows Server 2012 R2 のみ) – 4.9.4500

Amazon AppStream 2.0 エージェントのバージョン	変更
12-20-2021	• ネイティブクライアントの使用時にマウスが消える問題を解決します • セッション終了時のストレージのアンマウント時間の問題を解決します • Windows Server 2016 を実行しているグラフィックスインスタンスでのシステムクラッシュに関する問題を解決します • システム暗号化グループポリシーが有効になっている場合の Windows Server インスタンスのサポートが追加されました。詳細については、「システム暗号化」を参照してください。 • ファイルシステムのキャッシュの切り替え機能を追加しました。 • 以下のソフトウェアコンポーネントで動作します。詳細については、「the section called “ベースイメージおよびマネージド型イメージの更新版リリースノート”」を参照してください。 • Amazon SSM Agent — 3.0.1295.0 • Amazon WDDM Hook ドライバー — 1.0.0.56 (Windows Server 2012 R2) • NICE DCV 仮想ディスプレイ — 1.0.34.0 (Windows Server 2016/2019) • EC2Config サービス (Windows Server 2012 R2 のみ) – 4.9.4500

Amazon AppStream 2.0 エージェントのバージョン	変更
10-19-2021	• Microsoft Windows プリンターサービスが無効になっているときにユーザーがストリーミングできない問題を解決 • 言語パックのインストールが正常に完了しない問題を解決 • フォルダとファイルがすべて大文字に変更される S3 ホームフォルダの問題を解決 • 以下のソフトウェアコンポーネントで動作します。詳細については、「the section called “ベースイメージおよびマネージド型イメージの更新版リリースノート”」を参照してください。 • Amazon SSM Agent — 3.0.1295.0 • Amazon WDDM Hook ドライバー — 1.0.0.56 (Windows Server 2012 R2) • NICE DCV 仮想ディスプレイ — 1.0.34.0 (Windows Server 2016/2019) • EC2Config サービス (Windows Server 2012 R2 のみ) – 4.9.4500

Amazon AppStream 2.0 エージェントのバージョン	変更
08-02-2021	• 重要な修正を含む USB ドライバの更新 • お客様のローカルマシンの caps lock 状態が、リモートマシンの caps lock 状態と同期しなくなる問題を解決しました。 • 以下のソフトウェアコンポーネントで動作します。詳細については、「the section called “ベースイメージおよびマネージド型イメージの最新版リリースノート”」を参照してください。 • Amazon SSM Agent — 3.0.1295.0 • Amazon WDDM Hook ドライバー — 1.0.0.56 (Windows Server 2012 R2) • NICE DCV 仮想ディスプレイ — 1.0.34.0 (Windows Server 2016/2019) • EC2Config サービス (Windows Server 2012 R2 のみ) — 4.9.4419.0
07-01-2021	• マネージド型イメージ更新用の増分エージェントをリリース。詳細については、「AppStream 2.0 のマネージド型イメージアップデートを使用してイメージを更新する」を参照してください。 • 2021 年 6 月 25 日リリースのエージェントからの変更が含まれます。

Amazon AppStream 2.0 エージェントのバージョン	変更
06-25-2021	• ネットワークに関する複数の問題を解決しました。 • ローカルグループポリシーが上書きされる問題を解決しました。 • OneDrive および Google クラウドストレージからフェッチしようとしたファイルが含まれる親ディレクトリが、ターゲット側に存在しなかった場合にファイル作成が失敗する問題を解決しました。 • セッション終了時のセッションスクリプトが実行されない問題を解決しました • ウェブクライアントでのウェブカメラのリダイレクトを新たにサポート
05-17-2021	• リアルタイムの音声/動画 (AV) 機能をデフォルトで有効にしました。 • イメージアシスタントの CLI コマンドの出力を有効な JSON になるように修正しました。 • 内部タイムアウトが原因で、インスタンスのプロビジョニングが失敗する問題を修正しました。 • Amazon SSM Agent、Amazon WDDM Hook Driver、および EC2Config サービスのバージョンは、以前のリリースのエージェントバージョンから変わりません。

Amazon AppStream 2.0 エージェントのバージョン	変更
2021 年 3 月 4 日	• 接続エラーの原因となるスマートカード認証に関する問題を解決します。接続エラーは、ユーザーが複数回ストリーミングセッションを閉じて再度開くときに発生します • Microsoft Office アプリケーションで右クリックメニューの項目を使用できない問題を解決します • OneDrive と Google ドライブの複数のストレージコネクタプロセスがタスクマネージャーに表示される問題を解決します • 2 GB を超えるファイルを Google ドライブからダウンロードできない問題を解決します • Microsoft Active Directory ドメインに参加している AppStream 2.0 フリートインスタンスのプロビジョニングの遅延が発生する断続的な問題を解決します • 以下のソフトウェアコンポーネントと連携します。 • Amazon SSM Agent – 3.0.431.0 • Amazon WDDM Hook Driver — 1.0.0.56 • EC2Config サービス (Windows Server 2012 R2 のみ) – 4.9.4279.0

Amazon AppStream 2.0 エージェントのバージョン	変更
2020 年 12 月 17 日	• アプリケーション設定の永続性 VHD ファイルが AppStream 2.0 フリートストリーミングインスタンスにダウンロードされない問題を解決します。 • AppStream 2.0 ストリーミングセッション中にローカルプリンターリダイレクトが機能しなくなる問題を解決します。この問題は、Microsoft KB4571694 が AppStream 2.0 Image Builder またはフリートストリーミングインスタンスにインストールされている場合に発生することがあります。 • ローカルの Microsoft Windows ユーザーをデフォルトのユーザープロファイルのソースとして参照しようとする、Image Assistant update-default-profile コマンドラインインターフェイス (CLI) 操作によってエラーが返される問題を解決します。 • システム暗号化が FIPS 準拠のアルゴリズムを使用するように設定されている場合に AppStream 2.0 フリートインスタンスがプロビジョニングできない問題を解決します。 • ネイティブアプリケーションモードでストリーミングセッション中に、ユーザーのローカルコンピュータのタスクバーにアイコンが表示されない問題を解決します。 • Microsoft SharePoint で共有されるファイルのサポートを OneDrive for Business 永続ストレージコネクタに追加します。 • 以下のソフトウェアコンポーネントと連携します。 • Amazon SSM Agent — 2.3.1319.0 • Amazon WDDM Hook Driver — 1.0.0.56 • EC2Config サービス (Windows Server 2012 R2 のみ) — 4.9.4222.0

Amazon AppStream 2.0 エージェントのバージョン	変更
2021 年 1 月 4 日	• Active Directory に参加しているストリーミングインスタンスに Windows でサインインするためのスマートカードの使用と、ストリーミングアプリケーションのセッション内認証のサポートを追加します。 • 以下のソフトウェアコンポーネントと連携します。 • Amazon SSM Agent — 2.3.1319.0 • Amazon WDDM Hook Driver — 1.0.0.56 • EC2Config サービス (Windows Server 2012 R2 のみ) — 4.9.4222.0

Amazon AppStream 2.0 エージェントのバージョン	変更
2020 年 12 月 17 日	• アプリケーション設定の永続性 VHD ファイルが AppStream 2.0 フリートストリーミングインスタンスにダウンロードされない問題を解決します。 • AppStream 2.0 ストリーミングセッション中にローカルプリンターリダイレクトが機能しなくなる問題を解決します。この問題は、Microsoft KB4571694 が AppStream 2.0 Image Builder またはフリートストリーミングインスタンスにインストールされている場合に発生することがあります。 • ローカルの Microsoft Windows ユーザーをデフォルトのユーザープロファイルのソースとして参照しようとする、Image Assistant update-default-profile コマンドラインインターフェイス (CLI) 操作によってエラーが返される問題を解決します。 • システム暗号化が FIPS 準拠のアルゴリズムを使用するように設定されている場合に AppStream 2.0 フリートインスタンスがプロビジョニングできない問題を解決します。 • ネイティブアプリケーションモードでストリーミングセッション中に、ユーザーのローカルコンピュータのタスクバーにアイコンが表示されない問題を解決します。 • Microsoft SharePoint で共有されるファイルのサポートを OneDrive for Business 永続ストレージコネクタに追加します。 • 以下のソフトウェアコンポーネントと連携します。 • Amazon SSM Agent — 2.3.1319.0 • Amazon WDDM Hook Driver — 1.0.0.56 • EC2Config サービス (Windows Server 2012 R2 のみ) — 4.9.4222.0

Amazon AppStream 2.0 エージェントのバージョン	変更
2020 年 10 月 8 日	• ユーザーが AppStream 2.0 ストリーミングセッションに接続したときに内部エラー通知を受信する問題を解決します。 • AppStream 2.0 ストリーミングセッション中に断続的なコピーアンドペーストエラーが発生する問題を解決します。 • ネイティブアプリケーションモードで AppStream 2.0 ストリーミングセッション中に、アプリケーションアイコンがタスクバーに表示されない問題を解決します。 • アイドル状態の接続解除後にユーザーが AppStream 2.0 に再接続すると、アプリケーションカタログが空に表示される問題を解決します。 • AppStream 2.0 ホームフォルダと AppStream 2.0 フリートインスタンス間のダウンロード速度が向上 • 以下のソフトウェアコンポーネントと連携します。 • Amazon SSM Agent — 3.0.161.0 • Amazon WDDM Hook Driver — 1.0.0.56 • EC2Config サービス (Windows Server 2012R2 のみ) — 4.9.4222.0

Amazon AppStream 2.0 エージェントのバージョン	変更
2020 年 9 月 1 日	• グラフィックデザインのインスタンスで正しい解像度が表示されない問題を解決します。 • ネイティブアプリケーションモードで AppStream 2.0 クライアントを使用して Microsoft リモートデスクトップをストリームすると、白い画面が表示される問題を解決します。 • 最小化時にストリーミングアプリケーションがフリーズする問題を解決します。この問題は、App Stream 2.0 クライアントをネイティブアプリケーションモードで使用するときに発生します。 • 以下のソフトウェアコンポーネントと連携します。 • Amazon SSM Agent — 2.3.1319.0 • Amazon WDDM Hook Driver — 1.0.0.56 • EC2Config サービス (Windows Server 2012R2 のみ) — 4.9.4222.0
2020 年 7 月 30 日	• Windows 用 AppStream 2.0 クライアントへのプリンターリダイレクトのサポートを追加します。 • 5 GB を超えるファイルのダウンロードが停止し、失敗する問題を解決します。 • 2016 年の Microsoft Office プラグインを使用するときのクリップボードのパフォーマンスが向上します • 以下のソフトウェアコンポーネントと連携します。 • Amazon SSM Agent — 2.3.1319.0 • Amazon WDDM Hook Driver — 1.0.0.56 • EC2Config サービス (Windows Server 2012R2 のみ) — 4.9.4222.0

Amazon AppStream 2.0 エージェントのバージョン	変更
05-27-2020	• Windows 用の AppStream 2.0 クライアントを使用してネイティブアプリケーションモードでストリーミングすると、一部のアプリケーションのサイズ変更、移動、最大化のいずれかに失敗する問題を解決します。 • ユーティリティソフトウェアをダウンロードする際の断続的な問題を解決します。この問題により、イメージビルダーとフリートインスタンスがプロビジョニングされない場合があります。 • イメージビルダーとフリートインスタンスのプロビジョニングエラーを起こす可能性がある、特定の言語設定に関する断続的な問題を解決します。 • 以下のソフトウェアコンポーネントと連携します。 • Amazon SSM Agent — 2.3.701.0 • Amazon WDDM Hook Driver — 1.0.0.56 • EC2Config サービス (Windows Server 2012R2 のみ) — 4.9.3519.0
2020 年 4 月 20 日	• セッションスクリプトの実行時にストリーミングセッションが失敗する問題を解決 • IAM ロールの使用時のパフォーマンスが向上 • 以下のソフトウェアコンポーネントと連携します。 • Amazon SSM Agent — 2.3.701.0 • Amazon WDDM Hook Driver — 1.0.0.56 • EC2Config サービス (Windows Server 2012R2 のみ) — 4.9.3519.0

Amazon AppStream 2.0 エージェントのバージョン	変更
02-19-2020	• ネイティブアプリケーションモードのサポートを追加しました。詳細については、「ネイティブアプリケーションモード」を参照してください。 • [Desktop (デスクトップ)] ストリームビューのサポートを追加 • AppStream 2.0 コンポーネント間のプロセス間通信を改良しました。 • ストリーミングインスタンスのプロビジョニングに失敗する原因となっていた問題を解決しました。 • 以下のソフトウェアコンポーネントと連携します。 • Amazon SSM Agent — 2.3.701.0 • Amazon WDDM Hook Driver — 1.0.0.56 • EC2Config サービス (Windows Server 2012R2 のみ) — 4.9.3519.0
01-13-2020	• G Suite 用 Google ドライブを使用した永続的ストレージで、チームドライブが共有ドライブに変更されました • ユーザーの数が多い Active Directory 環境でストリーミングインスタンスのプロビジョニングが遅くなる問題を解決しました • フリートユーザーが管理者である場合に、アプリケーションスイッチャーからアプリケーションにアクセスする際の問題を解決しました • 以下のソフトウェアコンポーネントと連携します。 • Amazon SSM Agent — 2.3.701.0 • Amazon WDDM Hook Driver — 1.0.0.56 • EC2Config サービス (Windows Server 2012R2 のみ) — 4.9.3519.0

Amazon AppStream 2.0 エージェントのバージョン	変更
11-13-2019	• AppStream 2.0 アセンブリは現在、実行可能ファイルとインストーラーパッケージを含め、署名されています • 以下のソフトウェアコンポーネントと連携します。 • Amazon SSM Agent — 2.3.701.0 • Amazon WDDM Hook Driver — 1.0.0.56 • EC2Config サービス — 4.9.3519.0
10-08-2019	• システムプロキシサーバーをバイパスしないように AppStream 2.0 ストレージコネクタを変更 • 以下のソフトウェアコンポーネントと連携します。 • Amazon SSM Agent — 2.3.701.0 • Amazon WDDM Hook Driver — 1.0.0.56 • EC2Config サービス — 4.9.3519.0
09-23-2019	• 子プロセスを開始するアプリケーションを起動するときに発生する問題を解決しました。 • ディレクトリトラバーサルの問題を解決しました。 • AppStream 2.0 エージェントが機能停止し、アプリケーションとのやり取りが妨げられる問題を解決しました。 • 以下のソフトウェアコンポーネントと連携します。 • Amazon SSM Agent — 2.3.701.0 • Amazon WDDM Hook Driver — 1.0.0.56 • EC2Config サービス — 4.9.3519.0

Amazon AppStream 2.0 エージェントのバージョン	変更
2019-09-03	• AppStream 2.0 ストリーミングインスタンスに IAM ロールを適用するためのサポートを追加します。詳細については、「IAM ロールを使用して、AppStream 2.0 ストリーミングインスタンスで実行されるアプリケーションとスクリプトにアクセス許可を付与する」を参照してください。 • コマンドラインインターフェイスを使用してプログラムで AppStream 2.0 イメージを作成するときにタグを指定するためのサポートを追加 • ストレージのマウント時にシステムプロキシサーバーをバイパスするように AppStream 2.0 ストレージコネクタを変更します。 • Image Assistant で .lnk ファイルが指定されない問題を解決します。 • 以下のソフトウェアコンポーネントと連携します。 • Amazon SSM Agent — 2.3.612.0 • Amazon WDDM Hook Driver — 1.0.0.56 • EC2Config サービス — 4.9.3429
2019-08-08	• AppStream 2.0 ファイルシステムリダイレクトのサポートを追加します。詳細については、「AppStream 2.0 ユーザーのファイルシステムリダイレクトを有効にする」を参照してください。 • 英語 - 英国 (en-GB)、英語 - カナダ (en-CA)、英語 - オーストラリア (en-AU) の 3 つの新しいロケールのサポートを追加 • 以下のソフトウェアコンポーネントと連携します。 • Amazon SSM Agent — 2.3.612.0 • Amazon WDDM Hook Driver — 1.0.0.56 • EC2Config サービス — 4.9.3429

Amazon AppStream 2.0 エージェントのバージョン	変更
07-26-2019	• コマンドラインインターフェイスを使用して AppStream 2.0 イメージをプログラムで作成および管理するサポートを追加します。詳細については、「Image Assistant CLI オペレーションを使用してプログラムで Amazon AppStream 2.0 イメージを作成する」を参照してください。 • イメージビルダーで Windows 自動更新が有効になっている場合に、イメージの作成がブロックされなくなりました。ただし、この場合、フリートで Windows の自動更新が無効になることが管理者に通知されます (つまり、フリートインスタンスでは Windows の自動更新は有効になりません)。 • フリートインスタンスの起動時に Windows 更新プログラムを無効にします。 • Image Builder インスタンスの起動時に、Administrators グループのユーザーが無効化されなくなりました。 • Image Builder インスタンスの起動時に、Administrators グループのユーザーが削除されるのではなく、無効化されるようになりました。 • ネットワーク接続が変更された場合にストリーミング解像度のサイズを変更できない問題を解決しました。 • アプリケーション設定の永続性が有効な場合、ストリーミング解像度のサイズを変更できない競合状態を解決しました。 • 以下のソフトウェアコンポーネントと連携します。 • Amazon SSM Agent — 2.3.612.0 • Amazon WDDM Hook Driver — 1.0.0.56 • EC2Config サービス — 4.9.3429

Amazon AppStream 2.0 エージェントのバージョン	変更
06-19-2019	• Windows Server 2016 と Windows Server 2019 ベースイメージのサポートを追加しました。 • AppStream 2.0 セッションスクリプトは、設定されたタイムアウトが超過した後に終了されるようになりました。 • ロケールが変更されるとストリーミングインスタンスがプロビジョニングされないことがある問題を解決しました。 • Image Builder で Windows 自動更新が有効になっているときに、イメージの作成をブロックする変更が含まれました。 • ストレージコネクタのマウントが失敗するとストリーミングインスタンスの停止に時間がかかることがある問題を解決しました。 • 以下のソフトウェアコンポーネントと連携します。 • Amazon SSM Agent — 2.3.612.0 • Amazon WDDM Hook Driver — 1.0.0.56 • EC2Config サービス — 4.9.3429

Amazon AppStream 2.0 エージェントのバージョン	変更
05-07-2019	• AppStream 2.0 使用状況レポートをサブスクライブするためのサポートを追加しました。詳細については、「AppStream 2.0 使用状況レポート」を参照してください。 • ユーザーがストリーミングセッションから切断されるまでにアイドル状態 (非アクティブ) となることのできる時間を設定するためのサポートを追加しました。詳細については、Amazon AppStream 2.0 フリートとスタックを作成するの「Create a Fleet」を参照してください。 • Amazon S3 仮想プライベートゲートウェイを使用して、ホームフォルダ設定とアプリケーション設定の永続性に関する Amazon S3 バケットの使用の問題を解決します。 • Image Builder で Windows 自動更新が有効になっているときに、イメージの作成をブロックする変更が含まれました。 • 永続的ストレージドライブ (ホームフォルダ、OneDrive、Google Drive) が断続的に [My Files] ダイアログボックスに表示されなくなる問題を解決しました。 • 以下のソフトウェアコンポーネントと連携します。 • Amazon SSM Agent — 2.3.542.0 • Amazon WDDM Hook Driver — 1.0.0.56 • EC2Config サービス — 4.9.3289

Amazon AppStream 2.0 エージェントのバージョン	変更
04-02-2019	• セッションスクリプトやストレージコネクタのマウントに関する問題を解決します • インスタンスのプロビジョニングに関する小規模な問題を解決します • 以下のソフトウェアコンポーネントと連携します。 • Amazon SSM Agent — 2.3.344.0 • Amazon WDDM Hook Driver — 1.0.0.56 • EC2Config サービス — 4.9.3067
03-07-2019	• タッチ対応の iPad、Android タブレット、Windows デバイスでのジェスチャのサポートを追加しました。 • イメージビルダーインスタンスでユーザーを切り替える際の問題を解決しました。 • インスタンス予約に関する断続的な問題を解決しました。 • 以下のソフトウェアコンポーネントと連携します。 • Amazon SSM Agent — 2.3.344.0 • Amazon WDDM Hook Driver — 1.0.0.56 • EC2Config サービス — 4.9.3067

Amazon AppStream 2.0 エージェントのバージョン	変更
01-22-2019	• ユーザーのストリーミングセッションで特定のイベントが発生したときに、インスタンスセッションスクリプトを使用して独自のカスタムスクリプトを実行できるようになりました。 • リソース作成中に AppStream 2.0 リソースタイプ (Image Builder、イメージ、フリート、スタック) にタグを追加できるようになりました。 • アプリケーション設定の Virtual Hard Disk (VHD) ファイルからストレージコネクタのログファイルを削除するように修正しました。 • 表示言語が英語から変更され、コマンドラインインターフェイス (AWS CLI) AWS バージョンが 1.16.36 より前の場合のイメージの作成を防止します。詳細については、AppStream 2.0 ユーザーのためのデフォルトのリージョン設定を構成するの「Special Considerations for Japanese Language Settings」を参照してください。 • 以下のソフトウェアコンポーネントと連携します。 • Amazon SSM Agent — 2.3.344.0 • Amazon WDDM Hook Driver — 1.0.0.56 • EC2Config サービス — 4.9.3067
01-08-2019	• 日付が 2019 年 1 月 8 日のベースイメージのインスタンスプロビジョニング時間を短縮しました。 • 以下のソフトウェアコンポーネントと連携します。 • Amazon SSM Agent — 2.3.344.0 • Amazon WDDM Hook Driver — 1.0.0.56 • EC2Config サービス — 4.9.3067

Amazon AppStream 2.0 エージェントのバージョン	変更
12-19-2018	- 動的カタログがアプリケーションカタログに追加されない問題を解決 - 以下のソフトウェアコンポーネントと連携します。 - Amazon SSM Agent — 2.2.619.0 - Amazon WDDM Hook Driver — 1.0.0.56 - EC2Config サービス — 4.9.2644
12-17-2018	- AppStream 2.0 クライアントが、グラフィックデザインインスタンスタイプを使用するストリーミングインスタンスのマルチモニタエクスペリエンスをサポート。 - グラフィックデスクトップまたはメモリ最適化インスタンスタイプを使用するフリートインスタンスで一時ドライブが表示される問題を解決。 - 以下のソフトウェアコンポーネントと連携します。 - Amazon SSM Agent — 2.2.619.0 - Amazon WDDM Hook Driver — 1.0.0.56 - EC2Config サービス — 4.9.2644

Amazon AppStream 2.0 エージェントのバージョン	変更
12-04-2018	• Windows を実行するウェブクライアントで日本語キーボードを使用するためのサポートも追加されています。 • AppStream 2.0 動的アプリケーションフレームワーク API を使用して動的アプリケーションプロバイダーを構築するためのサポートを追加しました。 • 複数のタブまたはブラウザで同じセッションを同時にストリーミングする問題を解決しました。 • マウントが完了するまで、ホームフォルダ、Google ドライブ、OneDrive を読み取り専用にするための修正が含まれています。 • Amazon S3 VPC エンドポイントに接続されたフリートインスタンスに格納されているホームフォルダのマウント時間を改善します。 • 以下のソフトウェアコンポーネントと連携します。 • Amazon SSM Agent — 2.2.619.0 • Amazon WDDM Hook Driver — 1.0.0.56 • EC2Config サービス — 4.9.2644
11-14-2018	• AppStream 2.0 Windows クライアントを使用してストリーミングセッションを開始するためのサポートを追加します。 • フリートユーザー名に環境変数を使用するアプリケーションを開く際の問題を解決します。 • 以下のソフトウェアコンポーネントと連携します。 • Amazon SSM Agent — 2.2.619.0 • Amazon WDDM Hook Driver — 1.0.0.56 • EC2Config サービス — 4.9.2644

Amazon AppStream 2.0 エージェントのバージョン	変更
10-30-2018	• アプリケーション設定の永続性が有効になっているときに 1 GB を超えるホームフォルダをマウントする問題を解決します。 • IPv6 が無効になっているときのイメージ作成に関する問題を解決します。 • セッション情報は、ストリーミングインスタンス内の環境変数として提供されるようになりました。 • 以下のソフトウェアコンポーネントと連携します。 • Amazon SSM Agent — 2.2.619.0 • Amazon WDDM Hook Driver — 1.0.0.56 • EC2Config サービス — 4.9.2644
10-24-2018	• Amazon S3 ホームフォルダディレクトリに 1,000 を超えるファイルを表示する修正が含まれています。 • 以下のソフトウェアコンポーネントと連携します。 • Amazon SSM Agent — 2.2.619.0 • Amazon WDDM Hook Driver — 1.0.0.56 • EC2Config サービス — 4.9.2644

Amazon AppStream 2.0 エージェントのバージョン	変更
2018 年 10 月 1 日	- 永続性アプリケーション設定のパフォーマンスを改善します - フリートインスタンスのすべてのドライブを非表示にする修正が含まれます。ただし、インスタンスから起動されるユーザーストリーミングセッション中のドライブ C およびドライブ D を除きます - アプリケーションスイッチャーから最小化されたアプリケーションのサブ画面にアクセスする際の問題を解決します。 - 以下のソフトウェアコンポーネントと連携します。 - Amazon SSM Agent — 2.2.619.0 - Amazon WDDM Hook Driver — 1.0.0.56 - EC2Config サービス — 4.9.2644
08-29-2018	- アプリケーション設定の永続化に関するサポートを追加します。 - AppStream 2.0 ストリーミングセッション内で大量のデータをアプリケーション間でコピーして貼り付ける際の問題を解決します。 - アプリケーションスイッチャーから応答しないアプリケーションにアクセスする際の問題を解決します。 - 以下のソフトウェアコンポーネントと連携します。 - Amazon SSM Agent — 2.2.619.0 - Amazon WDDM Hook Driver — 1.0.0.56 - EC2Config サービス — 4.9.2644

Amazon AppStream 2.0 エージェントのバージョン	変更
07-26-2018	• OneDrive の永続的ストレージのサポートを追加 • ホームフォルダおよび Google Drive への Visio ファイルの保存に関する問題を解決します。 • 以下のソフトウェアコンポーネントと連携します。 • Amazon SSM Agent — 2.2.619.0 • Amazon WDDM Hook Driver — 1.0.0.56 • EC2Config サービス — 4.9.2644
06-19-2018	• アプリケーションの起動用イメージの最適化に関する問題を解決します。 • 以下のソフトウェアコンポーネントと連携します。 • Amazon SSM Agent — 2.2.619.0 • Amazon WDDM Hook Driver — 1.0.0.56 • EC2Config サービス — 4.9.2644
06-06-2018	• 地域の設定とデフォルトのアプリケーションと Windows 設定のサポートを追加します。 • 以下のソフトウェアコンポーネントと連携します。 • Amazon SSM Agent — 2.2.619.0 • Amazon WDDM Hook Driver — 1.0.0.56 • EC2Config サービス — 4.9.2644
05-31-2018	• Google ドライブの永続的ストレージのサポートを追加 • 以下のソフトウェアコンポーネントと連携します。 • Amazon SSM Agent — 2.2.392.0 • Amazon WDDM Hook Driver — 1.0.0.56 • EC2Config サービス — 4.9.2586

Amazon AppStream 2.0 エージェントのバージョン	変更
05-21-2018	• データ転送の管理コントロールのサポートを追加 • macOS X での Safari ブラウザのサポートを追加します。 • 以下のソフトウェアコンポーネントと連携します。 • Amazon SSM Agent — 2.2.392.0 • Amazon WDDM Hook Driver — 1.0.0.56 • EC2Config サービス — 4.9.2586
03-19-2018	• 一部の環境におけるアプリケーションウィンドウの最小化の問題を解決 • 以下のソフトウェアコンポーネントと連携します。 • Amazon SSM Agent — 2.2.160.0 • Amazon WDDM Hook Driver — 1.0.0.56 • EC2Config サービス — 4.9.2400.0
01-24-2018	• 一部のキーボードレイアウトで Alt グラフキーが機能しない問題を解決 • 以下のソフトウェアコンポーネントと連携します。 • Amazon SSM Agent — 2.2.93.0 • Amazon WDDM Hook Driver — 1.0.0.50 • EC2Config サービス — 4.9.2262.0

Amazon AppStream 2.0 エージェントのバージョン	変更
12-07-2017	- Alt キーの組み合わせの使用に関する問題を解決します - ローカルコンピュータからストリーミングセッションへのファイルのアップロードに関する問題を解決します - 以下のソフトウェアコンポーネントと連携します。 - Amazon SSM Agent — 2.2.93.0 - Amazon WDDM Hook Driver — 1.0.0.21 - EC2Config サービス — 4.9.2218.0

Amazon Linux

Amazon AppStream 2.0 エージェントのバージョン	変更
03-24-2024	環境変数 DISPLAY が正しく設定されていない場合にブラックスクリーンの問題を引き起こすバグを修正しました。
11-13-2023	Linux をバージョン 2.0.20231020.1 に更新。詳細については、「Amazon Linux 2.0.20231020.1 リリースノート」を参照してください。
06-11-2023	エージェントのアップデートなし
03-15-2023	- ウェブカメラのサポートを向上しました。 - システム暗号化が FIPS 準拠のアルゴリズムを使用するように設定されている場合に AppStream 2.0 フリートインスタンスがプロビジョニングできない問題を解決します。
09-21-2022	ウェブカメラをサポート

Amazon AppStream 2.0 エージェントのバージョン	変更
	• Image Assistant GUI をサポート
11-19-2021	• スモールインスタンスタイプにおける空白画面の問題を解決
11-15-2021	• Linux インスタンスをサポート

Rocky Linux

Amazon AppStream 2.0 エージェントのバージョン	変更
12-19-2024	• Rocky Linux 8 をサポート

Red Hat Enterprise Linux

Amazon AppStream 2.0 エージェントのバージョン	変更
07-30-2024	• Red Hat Enterprise Linux 8 のサポート

チュートリアル: AppStream 2.0 コンソールを使用してカスタム AppStream 2.0 イメージを作成する

このチュートリアルでは、Microsoft Windows Server オペレーティングシステムに基づく AppStream 2.0 イメージを作成する方法について説明します。Amazon Linux 2、Rocky Linux、または Red Hat Enterprise Linux オペレーティングシステムに基づくカスタムイメージを作成する場合は、「」を参照してください[the section called “チュートリアル: Linux ベースのカスタムイメージを作成する”](#)。

このチュートリアルでは、ユーザーにストリーミングできるアプリケーションと、ユーザーがそのアプリケーションの使用をすばやく開始できるようにするためのデフォルトのアプリケーション設定

と Windows 設定が含まれたカスタム Amazon AppStream 2.0 イメージを作成する方法を学びます。このチュートリアルを完了するには、事前に Image Builder が必要です。Image Builder がない場合は、「[Image Builder を起動し、ストリーミングアプリケーションをインストールして設定する](#)」を参照してください。

⚠ Important

このチュートリアルに含まれている情報は、最新リリースのベースイメージに該当します。詳細については、「[AppStream 2.0 ベースイメージとマネージド型イメージの更新に関するリリースノート](#)」を参照してください。

内容

- [ステップ 1: Image Builder でアプリケーションをインストールする](#)
- [手順 2: AppStream 2.0 アプリケーションカタログを作成する](#)
- [ステップ 3: デフォルトのアプリケーションと Windows の設定を作成する](#)
- [ステップ 4: アプリケーションをテストする](#)
- [ステップ 5: アプリケーションを最適化する](#)
- [ステップ 6: イメージの作成を完了する](#)
- [ステップ 7 \(オプション\): イメージにタグを付け、コピーする](#)
- [ステップ 8: クリーンアップ](#)

ステップ 1: Image Builder でアプリケーションをインストールする

このステップでは、Image Builder に接続し、Image Builder にアプリケーションをインストールします。

⚠ Important

このステップを完了するには、Image Builder に、ローカル Administrator アカウントまたはローカル管理者権限を持つドメインアカウントでログインする必要があります。組み込まれたローカル Administrator アカウントの名前を変更したり、このアカウントを削除したりしないでください。このアカウントの名前変更や削除を行うと、Image Builder は起動せず、イメージの作成は失敗します。

Image Builder にアプリケーションをインストールする

1. 以下のいずれかを実行して、Image Builder に接続します。

- [AppStream 2.0 コンソールを使用する](#) (ウェブ接続のみ)
- [ストリーミング URL を作成する](#) (ウェブまたは AppStream 2.0 クライアント接続用)

Note

接続先の Image Builder が Active Directory ドメインに参加していて、組織でスマートカードサインインが必要な場合は、ストリーミング URL を作成し、接続に AppStream 2.0 クライアントを使用する必要があります。スマートカードサインインの詳細については、「[スマートカード](#)」をご参照ください。

2. アプリケーションウェブサイトまたは他のダウンロードソースから取得したアプリケーションをインストールします。アプリケーションをインストールし、次のステップに進みます。

Note

アプリケーションのダウンロードおよびインストールは、信頼済みサイトからのみ行います。

アプリケーションで Windows オペレーティングシステムの再起動が必要な場合は、再起動します。オペレーティングシステムが再起動する前に、Image Builder から切断されます。再起動が完了したら、Image Builder に再接続し、アプリケーションのインストールを完了します。

手順 2: AppStream 2.0 アプリケーションカタログを作成する

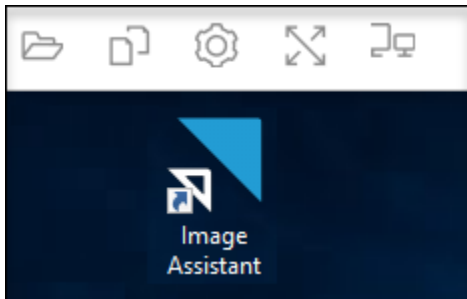
このステップでは、イメージに対してアプリケーション (.exe)、バッチスクリプト (.bat)、アプリケーションのショートカット (.lnk) を指定することで、AppStream 2.0 アプリケーションカタログを作成します。ストリーミングを計画するアプリケーションごとに、名前、表示名、起動する実行可能ファイルと表示するアイコンを指定できます。アプリケーションショートカットを選択した場合は、これらの値が設定されます。

⚠ Important

このステップを完了するには、ローカル Administrator アカウントまたはローカル管理者権限を持つドメインアカウントで Image Builder にログインする必要があります。

AppStream 2.0 アプリケーションカタログを作成する

1. Image Builder デスクトップから、Image Assistant を開きます。Image Assistant でイメージ作成プロセスを順番に実行します。



2. [1. Add Apps] (1. アプリケーションの追加) で [+ Add App] (アプリケーションの追加) を選択し、追加するアプリケーション、スクリプト、またはショートカットの場所に移動します。開く をクリックします。
3. [App Launch Settings] ダイアログボックスで、[Name]、[Display Name] および [Icon Path] のデフォルト設定をそのまま使用するか、変更します。必要に応じて、アプリケーションの起動パラメータ (起動したときにアプリケーションに渡される追加の引数) と作業ディレクトリを指定することができます。完了したら、[Save] を選択します。

[Display Name] と [Icon Path] 設定で、アプリケーションカタログに、どのようにアプリケーション名とアイコンが表示されるのかを決定します。カタログは、ユーザーが AppStream 2.0 ストリーミングセッションにサインインするときに表示されます。

4. Image Assistant で各アプリケーションに対してステップ 2 および 3 を繰り返し、アプリケーションが [Add Apps] タブに表示されていることを確認します。終了したら、[Next] を選択し、Image Assistant を使用してイメージの作成を続けます。

ステップ 3: デフォルトのアプリケーションと Windows の設定を作成する

このステップでは、AppStream 2.0 ユーザーに対してデフォルトのアプリケーションと Windows の設定を作成します。これにより、ユーザーがこれらの設定を作成または構成する必要がなくなるため、AppStream 2.0 ストリーミングセッション中にアプリケーションを迅速に開始できます。

⚠ Important

このステップを完了するには、ローカル Template User アカウントまたはローカル管理者権限がないドメインユーザーアカウントで Image Builder にログインする必要があります。

ユーザーに対してデフォルトのアプリケーションと Windows の設定を作成するには

1. Image Assistant の [2. Configure Apps] (2. アプリケーションの設定) で、[Switch user] (ユーザーを切り替える) を選択します。これにより現在のセッションから切断され、ログインメニューが表示されます。
2. 次のいずれかを実行します。
 - Image Builder が Active Directory ドメインに参加していない場合、[Local User] タブで、[Template User] を選択します。このアカウントでは、ユーザーがデフォルトのアプリケーションと Windows の設定を作成できます。
 - Image Builder が Active Directory ドメインに参加している場合は、[Directory User] を選択し、ローカル管理者権限がないドメインユーザーとしてログインします。
3. Image Builder デスクトップから、Image Assistant を開きます。アプリケーションカタログを作成したときに追加したアプリケーションが表示されます。
4. デフォルトのアプリケーション設定を作成するアプリケーションを選択します。
5. アプリケーションが開いた後、必要に応じて、これらの設定を作成します。
6. 終了したら、アプリケーションを閉じ、Image Assistant に戻ります。
7. Image Assistant で複数のアプリケーションを指定した場合は、必要に応じてアプリケーションごとにステップ 4~6 を繰り返します。
8. デフォルトの Windows 設定が必要な場合は、ここで作成します。終了したら、Image Assistant に戻ります。
9. [Switch user] を選択し、アプリケーションカタログを作成した際に使用した、同じアカウント (ローカル管理者権限を持つアカウント) でログインします。
10. Image Assistant の [2. Configure Apps] (2. アプリケーションの設定) で、以下のいずれかを実行します。
 - Image Builder が Active Directory ドメインに参加していない場合は、[Save settings] を選択します。

- Image Builder が Active Directory ドメインに参加している場合は、[Choose which user settings to copy] リストで、デフォルトのアプリケーションおよび Windows 設定を作成したときに Image Builder にログインするために使用したのと同じアカウントを選択し、[Save settings] を選択します。

[コピーする設定を選択] リストに、Image Builder に設定が現在保存されているアカウントが表示されます。

11. 終了したら、[Next] を選択し、イメージの作成を続行します。

ステップ 4: アプリケーションをテストする

このステップでは、追加したアプリケーションが、正しく開き、予想どおり動作することを確認します。確認するには、対象ユーザーと同じアクセス権限を持つユーザーとして新規に Windows セッションを開始します。

Important

このステップを完了するには、[Test User] アカウントまたはローカル管理者権限がないドメインアカウントで Image Builder にログインする必要があります。

アプリケーションをテストするには

1. Image Assistant の [3. Test] (3. テスト) で、以下のいずれかを行います。
 - Image Builder が Active Directory ドメインに参加していない場合は、[Switch user] を選択します。
 - Image Builder が Active Directory ドメインに参加しており、アプリケーションをテストするためにドメインアカウントが必要であり、かつユーザーの設定が Image Builder に既にある場合は、そのユーザーのアプリケーション設定をリセットする必要があります。これを行うには、[User to reset] リストからユーザーを選択し、[Reset] を選択します。完了したら、[Switch user] を選択します。

Note

Image Builder が新規作成され、Image Builder に設定を持っているユーザーがない場合は、リストにユーザーが表示されません。

2. テストに使用するユーザーを選択し、次のいずれかを実行します。
 - Image Builder が Active Directory ドメインに参加していない場合、[Test User] を選択します。このアカウントにより、対象ユーザーと同じポリシーおよびアクセス権限を使用してアプリケーションをテストすることができます。
 - Image Builder が Active Directory ドメインに参加している場合は、[ディレクトリユーザー] を選択し、ローカル管理者権限がないドメインアカウントの認証情報を指定して、[ログイン] を選択します。
3. Image Builder デスクトップから、Image Assistant を開きます。アプリケーションカタログを作成したときに指定したアプリケーションが表示されます。
4. テストするアプリケーションを選択し、正しく開くこと、作成したデフォルトのアプリケーション設定が適用されていることを確認します。
5. アプリケーションが開いたら、必要に応じてテストします。終了したら、アプリケーションを閉じ、Image Assistant に戻ります。
6. Image Assistant で複数のアプリケーションを指定した場合は、必要に応じてテストするアプリケーションごとにステップ 4~5 を繰り返します。
7. 終了したら、[Switch user] を選択し、次のいずれかの操作を行います。
 - Image Builder が Active Directory ドメインに参加していない場合、[Local User] タブで、[Administrator] を選択します。
 - Image Builder が Active Directory ドメインに参加しており、Image Assistant でアプリケーションを指定するためローカル管理者権限を持つドメインユーザーとしてログインした場合、そのユーザーとしてログインします。
8. [Next] を選択し、イメージの作成を続行します。

ステップ 5: アプリケーションを最適化する

このステップでは、Image Assistant でアプリケーションを 1 つずつ開き、起動の依存関係を識別し、アプリケーションがすばやく起動するように最適化を実行します。これらは、リスト内のすべてのアプリケーションで実行する必要があるステップです。

アプリケーションを最適化するには

1. Image Assistant の [4. Optimize] (4. 最適化) で [Launch] (起動) を選択します。
2. AppStream 2.0 は、リスト内の最初のアプリケーションを自動的に起動します。アプリケーションの起動が完了したら、アプリケーションの初回実行エクスペリエンスを実行するために必要な

情報を入力します。たとえば、ウェブブラウザが完全に起動して実行される前に、設定をインポートするように求められることがあります。

3. 初回実行エクスペリエンスが完了し、予想どおりにアプリケーションが動作することを確認してから、[Continue] を選択します。複数のアプリケーションをイメージに追加した場合は、各アプリケーションが自動的に開きます。必要に応じてアプリケーションごとに、このステップを繰り返し、すべてのアプリケーションを実行状態にします。
4. 終了したら、Image Assistant の次のタブ、[5. Configure Image] (5. イメージの設定) が自動的に表示されます。

ステップ 6: イメージの作成を完了する

このステップでは、イメージの名前を選択してイメージの作成を完了します。

イメージを作成する

1. 一意のイメージ名、およびオプションのイメージ表示名と説明を入力します。イメージ名は「Amazon」、「AWS」、または「AppStream」で始めることはできません。

イメージに 1 つ以上のタグを追加することもできます。そのためには、[タグの追加] を選択し、タグのキーと値を入力します。さらにタグを追加するには、この手順を繰り返します。詳細については、「[Amazon AppStream 2.0 リソースにタグを付ける](#)」を参照してください。終了したら、[次へ] を選択します。

Note

2017 年 12 月 7 日以降 AWS に よって公開されるベースイメージを選択した場合、デフォルトで選択されている最新のエージェントバージョンを常に使用するオプションが表示されます。このオプションを選択したままにし、イメージから起動されるストリーミングインスタンスで、常に最新バージョンのエージェントを使用するようにします。このオプションを無効にすると、イメージの作成完了後に再度有効にすることはできません。最新リリースの AppStream 2.0 エージェントの詳細については、[AppStream 2.0 エージェントのリリースノート](#) を参照してください。

2. [6. Review] (6. 確認) で、イメージの詳細を確認します。設定を変更するには、[Previous] を選択し、適切な Image Assistant タブに移動します。変更を加えた後、必要に応じて Image Assistant の各手順を実行します。
3. イメージの詳細の確認が完了した後、[Disconnect and Create Image] を選択します。

- 数秒以内に、リモートセッションが切断されます。[Lost Connectivity] メッセージが表示されたら、ブラウザタブを閉じます。イメージが作成されると、Image Builder のステータスが [Snapshotting] と表示されます。このプロセスが終了するまで Image Builder に接続することはできません。
- コンソールに戻り、[Images]、[Image Registry] の順に移動します。新しいイメージがリストに表示されていることを確認します。

イメージが作成中であるときは、コンソールのイメージレジストリでイメージのステータスが [Pending] と表示され、そのイメージに接続することはできません。

- ステータスを更新するには、定期的に [Refresh] アイコンを選択します。イメージが作成されたら、イメージのステータスは [Available] に変わり、Image Builder が自動的に停止されます。

イメージの作成を続行するには、Image Builder を開始してコンソールから接続するか、新しい Image Builder を作成することができます。

Note

イメージの作成後、Windows オペレーティングシステムの更新プログラムの維持はユーザーの責任になります。これを行うには、マネージド AppStream 2.0 イメージ更新を使用できます。ユーザーは、アプリケーションとその依存関係の更新を維持する責任も担います。詳細については、「[Amazon AppStream 2.0 イメージを最新の状態に保つ](#)」を参照してください。

他のアプリケーションの追加、既存のアプリケーションの更新、またはイメージ設定の変更を行うには、イメージの作成に使用した Image Builder を起動して再接続する必要があります。または、その Image Builder を削除したという場合は、イメージに基づいた新しい Image Builder を起動します。次に、変更を反映して新しいイメージを作成します。

ステップ 7 (オプション): イメージにタグを付け、コピーする

イメージの作成中または作成後に 1 つ以上のタグをイメージに追加できます。イメージは、同じリージョン内でコピーする、または同じ Amazon Web Services アカウント内の新しいリージョンにコピーすることもできます。ただし、ソースイメージをコピーすると、送信先イメージは同一でも区別されます。AWS はユーザー定義タグをコピーしません。また、コピーできるのは独自に作成したカスタムイメージのみであり、AWS が提供するベースイメージはコピーできません。

Note

同時に 2 つのイメージをコピー先にコピーできます。イメージのコピー先がイメージの制限に達している場合は、エラーが発生します。この場合にイメージをコピーするには、最初にコピー先からイメージを削除する必要があります。コピー先がイメージのクォータ（制限とも呼ばれます）を下回ったら、ソースリージョンからイメージのコピーを開始します。詳細については、「[Amazon AppStream 2.0 Service Quotas](#)」を参照してください。

既存のイメージにタグを追加するには

1. ナビゲーションペインで、[イメージ]、[イメージレジストリ] の順に選択します。
2. イメージのリストで、タグを追加するイメージを選択します。
3. [Tags]、[Add/Edit Tags]、[Add Tag] の順に選択し、タグのキーと値を指定して、[Save] を選択します。

詳細については、「[Amazon AppStream 2.0 リソースにタグを付ける](#)」を参照してください。

イメージをコピーするには

地理的に異なるリージョン間でイメージをコピーすると、同じイメージに基づいて複数のリージョンからアプリケーションをストリーミングできます。ユーザーに近い場所でアプリケーションをストリーミングすることによって、AppStream 2.0 を使用したアプリケーションのストリーミングに関するユーザーのエクスペリエンスを向上させることができます。

1. ナビゲーションペインで、[イメージ]、[イメージレジストリ] の順に選択します。
2. イメージのリストで、コピーするイメージを選択します。
3. アクション、コピーの順に選択します。
4. [Copy Image (イメージのコピー)] ダイアログボックスで、以下の情報を指定し、[Copy Image (イメージのコピー)] を選択します。
 - [送信先リージョン] で、新しいイメージのコピー先となるリージョンを選択します。
 - [名前] に、イメージが送信先にコピーされるときの名前を指定します。
 - [説明] (オプション) に、イメージが送信先にコピーされときの説明を指定します。

5. コピー操作の進行状況を確認するには、コンソールに戻り、[イメージ]、[イメージレジストリ]に移動します。ナビゲーションバーを使用して、コピー先のリージョンに切り替え (該当する場合)、新しいイメージがイメージのリストに表示されることを確認します。

新しいイメージは、コンソールのイメージレジストリで、最初のステータスがコピー中として表示されます。イメージが正常に作成されると、イメージのステータスが [Available] に変わります。つまり、イメージを使用してスタックを起動し、アプリケーションをストリーミングすることが可能です。

ステップ 8: クリーンアップ

最後に、実行中の Image Builder を停止してリソースを解放し、アカウントに意図しない料金が発生するのを避けます。使用していない、実行中の Image Builder を停止することをお勧めします。詳細については、[AppStream 2.0 の料金](#)をご覧ください。

実行中の Image Builder を停止するには

1. ナビゲーションペインで、[Images]、[Image Builders] の順に選択し、実行中の Image Builder インスタンスを選択します。
2. [Actions]、[Stop] の順に選択します。

Amazon AppStream 2.0 画像を管理する

使用可能なイメージは、AppStream 2.0 コンソールの [Image Registry (イメージレジストリ)] にあり、以下のように可視性に基づいて分類されています。

- **パブリック** — によって所有され、利用可能になったベースイメージ AWS。ベースイメージには、最新の Windows オペレーティングシステムと AppStream 2.0 エージェントソフトウェアが含まれます。これらのベースイメージを使用して、ユーザー独自のアプリケーションを含む新しいイメージを作成できます。によってリリースされるベースイメージの詳細については AWS、「」を参照してください[AppStream 2.0 ベースイメージとマネージド型イメージの更新に関するリリースノート](#)。
- **プライベート** — 作成して所有し、他の AWS アカウントと共有していないイメージ。
- **他のユーザーと共有** — 作成して所有し、同じ AWS リージョン内の 1 つ以上の AWS アカウントと共有したイメージ。イメージを別の AWS アカウントと共有する場合、イメージを Image Builder (新しいイメージを作成)、フリート、またはその両方に使用できるかどうかを指定できます。

- 自分と共有 — 同じ AWS リージョン内の別の AWS アカウントによって作成および所有され、自分の AWS アカウントと共有されているイメージ。自分のアカウントでイメージを共有するときに所有者が指定したアクセス許可に応じて、このイメージは Image Builder、フリート、またはその両方で使用できます。

内容

- [Amazon AppStream 2.0 でプライベートイメージを削除する](#)
- [Amazon AppStream 2.0 の別の AWS リージョンに所有しているイメージをコピーする](#)
- [Amazon AppStream 2.0 で自分が所有しているイメージを 別の AWS アカウントと共有する](#)
- [Amazon AppStream 2.0 で所有しているイメージの共有を停止する](#)
- [Amazon AppStream 2.0 イメージを最新の状態に保つ](#)
- [Amazon AppStream 2.0 での Windows Update とウイルス対策ソフトウェア](#)
- [Amazon AppStream 2.0 でプログラムで新しいイメージを作成する](#)

Amazon AppStream 2.0 でプライベートイメージを削除する

不要になったプライベートイメージは削除できます。フリートで使用されているイメージや他の AWS アカウントと共有されているイメージは削除できません。フリートで使用されているイメージや共有されているイメージを削除するには、最初に、フリートからのイメージの削除とすべてのイメージ共有アクセス権限の削除を行う必要があります。削除したイメージは回復できません。

プライベートイメージを削除するには

1. AppStream 2.0 コンソール (<https://console.aws.amazon.com/appstream2>) を開きます。
2. ナビゲーションペインで、[イメージ]、[イメージレジストリ] の順に選択します。
3. イメージのリストで、削除するプライベートイメージを選択します。
4. [Actions (アクション)]、[Delete (削除)] の順に選択し、[Delete (削除)] を再度選択します。

イメージはイメージレジストリから除去されて削除されます。

Amazon AppStream 2.0 の別の AWS リージョンに所有しているイメージをコピーする

所有しているイメージを別の AWS リージョンにコピーできます。異なる AWS リージョンで同じイメージを使用すると、AppStream 2.0 でのアプリケーションのグローバルデプロイを簡素化できます。ユーザーに最も地理的に近い AWS リージョンにアプリケーションをデプロイすることで、ユーザーにより応答性の高いエクスペリエンスを提供できます。

所有しているイメージを別の AWS リージョンにコピーするには

1. AppStream 2.0 コンソール (<https://console.aws.amazon.com/appstream2>) を開きます。
2. ナビゲーションペインで、[イメージ]、[イメージレジストリ] の順に選択します。
3. イメージリストで、別の AWS リージョンにコピーするイメージを選択します。
4. アクション、コピーの順に選択します。
5. イメージのコピーダイアログボックスの宛先リージョンで、イメージのコピー先の AWS リージョンを選択します。
6. イメージの一意の名前と説明 (オプション) を [Destination region (コピー先リージョン)] に入力します。
7. [Copy Image (イメージのコピー)] を選択します。

Amazon AppStream 2.0 で自分が所有しているイメージを 別の AWS アカウントと共有する

AppStream 2.0 イメージはリージョンリソースであるため、所有しているイメージを同じ AWS リージョン内の他の AWS アカウントと共有できます。イメージの共有は、さまざまなシナリオで役立つことがあります。例えば、異なる AWS アカウントを使用して開発リソースと本番稼働用リソースを分離する場合、開発アカウントを使用してイメージを作成できます。次に、本番稼働用アカウントでイメージを共有できます。所属組織が独立系ソフトウェアベンダー (ISV) である場合は、最適化したイメージを顧客と共有できます。最適化したイメージには必要なアプリケーションがインストールおよび設定済みであるため、顧客はアプリケーションを自分でインストールして設定する必要がなく、アプリケーションをすぐに使い始めることができます。

イメージを別の AWS アカウントと共有する場合、コピー先アカウントがフリート内のイメージを使用できるか、Image Builder を作成して新しいイメージを作成できるかを指定します。共有イメージの所有権は自分に帰属します。これにより、共有イメージのアクセス権限の追加、変更、または削除を必要に応じて行うことができます。

イメージを別のアカウントと共有して、このアカウントにフリートへのアクセス権限を付与すると、共有イメージを使用して当該アカウントのフリートを作成または更新できます。これらのアクセス権限を後で削除すると、アカウントではイメージを使用できなくなります。共有イメージを使用するアカウントのフリートの場合、希望する容量を 0 に設定すると、新しいフリートインスタンスは作成されません。ストリーミングセッションが終了するまで既存のセッションが続行されます。新規作成するフリートインスタンスの場合は、そのアカウントのフリートを有効なイメージで更新する必要があります。

イメージを別のアカウントと共有して、このアカウントに Image Builder へのアクセス権限を付与すると、共有イメージを使用して当該アカウントの Image Builder およびイメージを作成できます。これらのアクセス権限を後で削除しても、自分のイメージから作成した Image Builder やイメージは影響を受けません。

 Important

イメージを別のアカウントと共有した後で、このアカウントでこのイメージから作成した Image Builder やイメージをコントロールすることはできません。このため、アカウントに対して Image Builder へのアクセス権限を付与するのは、イメージのコピーを作成することをアカウントに許可する場合に限ります。また、イメージの共有を停止した後のコピーへのアクセス権は保持します。

所有しているイメージを別の AWS アカウントと共有するには

1. AppStream 2.0 コンソール (<https://console.aws.amazon.com/appstream2>) を開きます。
2. ナビゲーションペインで、[イメージ]、[イメージレジストリ] の順に選択します。
3. イメージのリストで、共有するイメージを選択します。
4. [Actions (アクション)]、[Edit (編集)] の順に選択します。
5. [Share image (イメージの共有)] ダイアログボックスで、[Add account (アカウントの追加)] を選択します。
6. イメージを共有するアカウントの 12 桁の AWS アカウント ID を入力し、アカウントが次のいずれかまたは両方を実行できるかどうかを選択します。
 - イメージを使用して Image Builder を起動する (新しいイメージを作成する場合)。
 - イメージをフリートで使用する。

イメージを共有する先のアカウントのリストからアカウントを削除するには、削除するアカウントの行で [Use for fleet (フリートで使用)] オプションの右側にある [X] アイコンを選択します。

- より多くの AWS アカウントとイメージを共有するには、イメージを共有するアカウントごとにステップ 6 を繰り返します。
- [Share Image (イメージの共有)] を選択します。

所有しているイメージに対してイメージ共有アクセス権限を追加または更新するには

- AppStream 2.0 コンソール (<https://console.aws.amazon.com/appstream2>) を開きます。
- ナビゲーションペインで、[イメージ]、[イメージレジストリ] の順に選択します。
- イメージのリストで、アクセス権限を変更するイメージを選択します。
- イメージのリストの下で、選択したイメージの [Permissions (アクセス権限)] タブを選択し、[Edit (編集)] を選択します。
- 「イメージのアクセス許可の編集」ダイアログボックスで、1 つ以上の AWS アカウントが必要に応じて、次のイメージ共有オプションの 1 つまたは両方を選択またはクリアします。アカウントで両方のオプションを選択解除すると、そのアカウントではイメージが共有されなくなります。
 - イメージを使用して Image Builder を起動する (新しいイメージを作成する場合)。
 - イメージをフリートで使用する。

イメージを共有する先のアカウントのリストからアカウントを削除するには、削除するアカウントの行で [Use for fleet (フリートで使用)] オプションの右側にある [X] アイコンを選択します。

- さらに多くの AWS アカウントのイメージ共有アクセス許可を編集するには、アクセス許可を更新するアカウントごとにステップ 5 を繰り返します。
- [Update image sharing permissions (イメージ共有アクセス権限の更新)] を選択します。

Amazon AppStream 2.0 で所有しているイメージの共有を停止する

所有しているイメージを他の AWS アカウントと共有しないようにするには、次の手順に従います。

所有しているイメージを他の AWS アカウントとの共有を停止するには

- AppStream 2.0 コンソール (<https://console.aws.amazon.com/appstream2>) を開きます。

2. ナビゲーションペインで、[イメージ]、[イメージレジストリ] の順に選択します。
3. イメージのリストで、アクセス権限を変更するイメージを選択します。
4. イメージのリストの下で、選択したイメージの [Permissions (アクセス権限)] タブを選択し、[Edit (編集)] を選択します。
5. イメージのアクセス許可の編集ダイアログボックスで、イメージが共有されているすべての AWS アカウントの行で、フリートに使用するオプションの右側にある X アイコンを選択します。
6. [Update image sharing permissions (イメージ共有アクセス権限の更新)] を選択します。

Amazon AppStream 2.0 イメージを最新の状態に保つ

AppStream 2.0 イメージを最新の状態に保つには、次のいずれかの操作を行います。

- [AppStream 2.0 のマネージド型イメージアップデートを使用してイメージを更新する](#) — この更新機能では、最新の Windows オペレーティングシステムの更新とドライバーの更新、および最新の AppStream 2.0 エージェントソフトウェアが提供されます。
- [マネージド型 AppStream 2.0 エージェントバージョンを使用して AppStream 2.0 エージェントソフトウェアを更新する](#) — この更新方法では、最新の AppStream 2.0 エージェントソフトウェアが提供されます。

AppStream 2.0 のマネージド型イメージアップデートを使用してイメージを更新する

AppStream 2.0 には、最新の Windows オペレーティングシステムの更新、ドライバーの更新、および AppStream 2.0 エージェントソフトウェアとともにイメージを自動的に更新する方法が用意されています。マネージド型 AppStream 2.0 イメージの更新では、ユーザーは更新するイメージを選択します。AppStream 2.0 は、同じ AWS アカウントとリージョンに Image Builder を作成し、更新をインストールして新しいイメージを作成します。新しいイメージが作成されたら、本番稼働用フリートを更新する前、または他の AWS アカウントとイメージを共有する前に、本番稼働前フリートでテストできます。

Note

新しいイメージの作成後、Windows オペレーティングシステムの更新プログラムの維持はユーザーの責任になります。これを行うには、マネージド型 AppStream 2.0 イメージの更新を継続的に使用できます。

Amazon EC2 Windows Paravirtual (PV) ドライバー、ENA ドライバー、AWS NVMe ドライバーの更新を維持する責任があります。ドライバーの更新方法の詳細については、[EC2 インスタンスのデバイスドライバーを管理する](#)」を参照してください。

ユーザーは、アプリケーションとその依存関係を維持する責任も負います。他のアプリケーションの追加、既存のアプリケーションの更新、またはイメージ設定の変更を行うには、イメージの作成に使用した Image Builder を起動して再接続する必要があります。または、その Image Builder を削除したという場合は、イメージに基づいた新しい Image Builder を起動します。次に、変更を反映して新しいイメージを作成します。

前提条件

マネージド型イメージの更新を使用する際の前提条件と考慮事項を以下に示します。

- AppStream 2.0 アカウントのクォータ (制限とも呼ばれる) が、新しい Image Builder と新しいイメージの作成に対応するのに十分であることを確認します。クォータの引き上げをリクエストするには、<https://console.aws.amazon.com/servicequotas/> の [Service Quotas] コンソールを使用します。デフォルトの AppStream 2.0 のクォータについては、「[Amazon AppStream 2.0 Service Quotas](#)」を参照してください。
- 更新するイメージは自分が所有している必要があります。自分が共有先になっているイメージを更新することはできません。
- AppStream 2.0 が Image Builder を作成して、最新の Windows オペレーティングシステムの更新、ドライバーの更新、および AppStream 2.0 エージェントソフトウェアをインストールし、新しいイメージを作成するときには、更新中の Image Builder インスタンスの料金が請求されます。
- サポートされるイメージは、2017-07-24T00:00:00Z 以降にリリースされたベースイメージから作成する必要があります。
- サポートされる表示言語は、英語と日本語です。詳細については、「[デフォルトの表示言語を指定する](#)」を参照してください。
- SSM Agent の最新バージョンを使用してください。バージョンについては、「[the section called “ベースイメージおよびマネージド型イメージの更新版リリースノート”](#)」を参照してください。インストールの詳細については、「[Windows Server の EC2 インスタンスに SSM Agent を手動でインストールする](#)」を参照してください。

AppStream 2.0 のマネージド型イメージ更新機能を使用してイメージを更新する方法

AppStream 2.0 イメージを最新のパッチ、ドライバーの更新、および AppStream 2.0 エージェントソフトウェアで更新するには、次の手順を実行します。

1. AppStream 2.0 コンソール (<https://console.aws.amazon.com/appstream2>) を開きます。
2. ナビゲーションペインで、[イメージ]、[イメージレジストリ] の順に選択します。
3. イメージのリストで、更新するイメージを選択します。イメージのステータスが Available になっていることを確認します。
4. [アクション]、[更新] の順に選択します。
5. [イメージの更新] ダイアログボックスで、次の操作を行います。
 - 新しいイメージ名に、AWS アカウントとリージョン内で一意のイメージ名を入力します。イメージ名を「Amazon」、「AWS」、または「AppStream」で始めることはできません。
 - [新しいイメージ表示名] には、オプションでこのイメージについて表示する名前を入力できます。
 - [新しいイメージの説明] には、オプションでこのイメージの説明を入力できます。
 - [タグ] では、[タグの追加] を選択し、タグのキーと値を入力します。さらにタグを追加するには、この手順を繰り返します。詳細については、「[Amazon AppStream 2.0 リソースにタグを付ける](#)」を参照してください。
6. [イメージの更新] を選択します。

現在のイメージが既に最新である場合は、メッセージで通知されます。

7. ナビゲーションペインで、[イメージ]、[Image Builder] の順に選択します。
8. Image Builder 一覧で、新しい Image Builder が [更新中] 状態になっていることを確認します。Image Builder の名前には 10 桁のランダムなサフィックスが含まれます。

Image Builder は、ステップ 5 で新しいイメージ用に選択したインスタンスファミリーの中で最小のサイズです。Image Builder は仮想プライベートクラウド (VPC) に接続していないため、サブネットは指定されていません。

9. [イメージレジストリ] を選択し、新しいイメージがリストに表示されていることを確認します。

イメージの作成中、イメージのステータスは、コンソールのイメージレジストリに [作成中] と表示されます。

10. イメージが作成されると、AppStream 2.0 は認定プロセスを実行して、イメージが期待どおりに機能することを確認します。

この間、このプロセスにも使用される Image Builder が、[Image Builder] リストに [認定待ち] のステータスで表示されます。

11. 認定プロセスが正常に完了すると、コンソールの上部に [成功] メッセージが表示され、イメージレジストリにイメージステータスが [使用可能] と表示されます。

さらに、AppStream 2.0 が作成した Image Builder が自動的に削除されます。

Note

Windows オペレーティングシステムの更新の量によっては、イメージの更新が完了するまでに数時間かかる場合があります。問題によりイメージを更新できない場合、イメージ名の横に感嘆符の付いた赤いアイコンが表示され、イメージレジストリのイメージステータスが [失敗] と表示されます。このような場合は、イメージを選択し、[通知] タブを選択して、エラー通知を確認してください。詳細については、通知コードのトラブルシューティングに関するドキュメントの [イメージの内部サービス](#) セクションの情報を参照してください。

認定プロセスが成功しなかった場合でも、AppStream 2.0 が作成した Image Builder は自動的に削除されます。

12. AppStream 2.0 が新しいイメージを作成したら、そのイメージを運用前のフリートでテストします。アプリケーションが想定どおりに動作することを確認したら、新しいイメージで本番稼働用フリートを更新します。

マネージド型 AppStream 2.0 エージェントバージョンを使用して AppStream 2.0 エージェントソフトウェアを更新する

AppStream 2.0 には、Image Builder をより新しい AppStream 2.0 エージェントソフトウェアで自動的に更新する方法が用意されています。この方法では、新しいバージョンのエージェントがリリースされるたびに新しいイメージを作成できます。その後、本番稼働用フリートを更新する前にイメージをテストできます。AppStream 2.0 エージェントソフトウェアの管理方法の詳細については、[AppStream 2.0 エージェントのバージョンを管理する](#) を参照してください。

Note

Windows オペレーティングシステムの更新、アプリケーション、それらの依存関係のインストールとメンテナンスは、お客様の責任で行います。

AppStream 2.0 イメージを Windows オペレーティングシステムの更新プログラムで最新の状態に保つには、以下のいずれかを実行します。

- 新しいイメージがリリースされるたびに、最新のベースイメージにアプリケーションをインストールする。
- Windows オペレーティングシステムの更新プログラム、お客様のアプリケーション、それらの依存関係を既存のイメージビルダーにインストールする。
- Windows オペレーティングシステムの更新プログラム、お客様のアプリケーション、それらの依存関係を既存のイメージから新しいイメージビルダーにインストールする。

最新の Windows オペレーティングシステム、お客様のアプリケーション、それらの依存関係、AppStream 2.0 エージェントソフトウェアを使用して新しいイメージを作成したら、開発フリートでそのイメージをテストします。アプリケーションが想定どおりに動作することを確認したら、新しいイメージで本番稼働用フリートを更新します。

Amazon AppStream 2.0 での Windows Update とウイルス対策ソフトウェア

AppStream 2.0 ストリーミングのインスタンスは永続的ではありません。ユーザーストリーミングセッションが終了すると、AppStream 2.0 はセッションで使用されたインスタンスを終了し、スケリングポリシーに応じて、新しいインスタンスをプロビジョニングしてフリート内のインスタンスに置き換えます。すべてのフリートインスタンスは、同じイメージからプロビジョニングされます。イメージは一度作成すると変更できないため、ユーザーストリーミングセッションで 사용되는すべてのフリートインスタンスには、イメージの作成時に基になるイメージにインストールされていた Windows とアプリケーションの更新のみが含まれます。さらに、ストリーミングセッションに使用されるフリートインスタンスはセッションの終了時に終了するため、ストリーミングセッション中にインスタンス上の Windows またはアプリケーションに対して行われた更新は、同じユーザーや他のユーザーによる以降のセッションでは保持されません。

Note

スタックでアプリケーション設定の永続性を有効にした場合、AppStream 2.0 は、ユーザーが行った Windows およびアプリケーション設定の変更を、同じユーザーの今後のセッションに保持します (これらの設定の変更がユーザーの Windows プロファイルに保存されている場合)。ただし、アプリケーション設定の永続化機能は、Windows およびアプリケーション

構成設定のみを保持します。ストリーミングインスタンス上の Windows またはアプリケーションに対するソフトウェア更新は保持されません。

このような理由から、AppStream 2.0 は AppStream 2.0 インスタンスで Windows Update およびウイルス対策ソフトウェアに対して次のアプローチを取ります。

Windows 更新

Windows Update は、AppStream 2.0 ベースイメージではデフォルトで有効になっていません。イメージビルダーで Windows Update を有効にしてからイメージを作成しようとすると、Image Assistant によって警告が表示され、イメージ作成プロセス中は Windows Update が無効になります。フリートインスタンスに最新の Windows 更新プログラムが確実にインストールされるようにするには、イメージビルダーに Windows 更新プログラムをインストールし、新しいイメージを作成して、その新しいイメージでフリートを定期的に更新することをお勧めします。

ウイルス対策ソフトウェア

イメージにウイルス対策ソフトウェアをインストールする場合は、ウイルス対策ソフトウェアの自動更新を有効にしないことをお勧めします。そうしないと、ウイルス対策ソフトウェアがユーザーセッション中に最新の定義ファイルまたはその他の更新でソフトウェア自体を更新しようとします。これは、パフォーマンスに影響を与える可能性があります。さらに、ウイルス対策ソフトウェアに加えられた更新は、現在のユーザーセッションを超えて保持されません。フリートインスタンスに常に最新のウイルス対策ソフトウェアの更新を適用するには、次のいずれかを実行することをお勧めします。

- イメージビルダーを更新し、定期的に新しいイメージを作成します (たとえば [Image Assistant CLI オペレーション](#) を使用)。
- スキャンやその他のオペレーションを常に最新の状態にある外部サーバーに委任するウイルス対策アプリケーションを使用します。

Note

ウイルス対策ソフトウェアの自動更新を有効にしていなくても、ウイルス対策ソフトウェアがユーザーセッション中にハードドライブスキャンやその他の操作を実行し、フリートインスタンスのパフォーマンスに影響を与えることがあります。

AppStream 2.0 Windows Server 2012 R2 ベースイメージには、ウイルス対策ソフトウェアは含まれていません。2019 年 9 月 10 日以降に公開された AppStream 2.0 Windows Server 2016 および Windows Server 2019 ベースイメージでは、Windows Defender はデフォルトでは有効になっていません。2019 年 6 月 24 日に発行された AppStream 2.0 Windows Server 2016 および Windows Server 2019 ベースイメージでは、Windows Defender はデフォルトで有効になっています。

Windows Defender を手動で有効にするには

ベースイメージで Windows Defender が有効になっていない場合は、手動で有効にすることができます。これには、以下のステップを実行します。

1. AppStream 2.0 コンソールを <https://console.aws.amazon.com/appstream2> で開きます。
2. 左側のナビゲーションペインで、[Images]、[Image Builder] の順に選択します。
3. Windows Defender を有効にするイメージビルダーを選択し、そのイメージが [実行中] 状態であることを確認して、[接続] を選択します。
4. Image Builder に、ローカル Administrator アカウントまたはローカル管理者権限を持つドメインアカウントを使用してログインします。
5. レジストリエディタを開きます。
6. レジストリ内の [HKLM\SOFTWARE\Policies\Microsoft\Windows Defender\DisableAntiSpyware] に移動します。
7. このレジストリキーを編集するには、そのレジストリキーをダブルクリックするか、レジストリキーを右クリックして [変更] を選択します。
8. [Edit DWORD (32-bit) Value (DWORD (32 ビット) 値の編集)] ダイアログボックスの [値のデータ] で [1] を [0] に変更します。
9. [OK] を選択します。
10. レジストリエディタを閉じます。
11. Microsoft 管理コンソール (MMC) サービススナップイン (services.msc) を開きます。
12. サービスのリストで、次のいずれかを実行します。

Microsoft Windows Server 2022 を使用している場合は、次のいずれかを実行します。

- Microsoft Defender Antivirus Service を右クリックし、開始を選択します。
- Microsoft Defender Antivirus Service をダブルクリックし、プロパティダイアログボックスで開始を選択し、OK を選択します。

Microsoft Windows Server 2019 または 2016 を使用している場合は、次のいずれかを実行します。

- [Windows Defender ウィルス対策サービス] を右クリックし、[スタート] を選択します。
- [Windows Defender ウィルス対策サービス] をダブルクリックし、プロパティダイアログボックスで [スタート] を選択し、[OK] を選択します。

13. サービススナップインを閉じます。

Amazon AppStream 2.0 でプログラムで新しいイメージを作成する

AppStream 2.0 イメージをプログラムで作成するには、Image Builder に接続し、Image Assistant のコマンドラインインターフェイス (CLI) オペレーションを使用します。詳細については、「[Image Assistant CLI オペレーションを使用してプログラムで Amazon AppStream 2.0 イメージを作成する](#)」を参照してください。

Image Assistant CLI オペレーションを使用してプログラムで Amazon AppStream 2.0 イメージを作成する

Amazon AppStream 2.0 イメージを作成するには、Image Builder に接続し、Image Assistant のグラフィカルユーザーインターフェイス (GUI) またはコマンドラインインターフェイス (CLI) のオペレーションを使用します。Image Assistant CLI オペレーションは、Image Assistant GUI と同様の機能を提供します。これらのオペレーションでは、次の操作をプログラムで実行できます。

- イメージに含まれているアプリケーションを管理する。
- デフォルトのアプリケーション設定を保存、更新、リセットする。
- AppStream 2.0 動的アプリケーションフレームワークを有効または無効にします。
- タグを指定する。
- イメージを作成する。

これらのオペレーションを使用して、AppStream 2.0 イメージ作成を継続的な統合またはデプロイソフトウェア開発プロセスに統合できます。

Image Assistant CLI オペレーションを使用するには、任意のコマンドラインシェルをイメージビルダーで使用します。たとえば、Windows コマンドプロンプトや PowerShell を使用できます。

Note

Image Builder は、2019 年 7 月 26 日以降にリリースされたバージョンの AppStream 2.0 エージェントを使用する必要があります。イメージビルダーがない場合は、作成する必要があります。詳細については、「[Image Builder を起動し、ストリーミングアプリケーションをインストールして設定する](#)」を参照してください。

目次

- [Image Assistant CLI 操作を使用したデフォルトのアプリケーション設定と Windows 設定の作成](#)
- [Image Assistant CLI オペレーションによるアプリケーションの起動パフォーマンスの最適化](#)
- [Amazon AppStream 2.0 イメージをプログラムで作成するプロセスの概要](#)
- [Amazon AppStream 2.0 イメージを作成および管理するための Image Assistant CLI オペレーション](#)

Image Assistant CLI 操作を使用したデフォルトのアプリケーション設定と Windows 設定の作成

ユーザーがアプリケーションをすぐに使い始められるように、デフォルトのアプリケーションおよび Windows 設定を作成できます。これらの設定を作成すると、AppStream 2.0 により、Windows のデフォルトユーザープロファイルが構成したプロファイルに置き換えられます。次に、この Windows デフォルトユーザープロファイルを使用して、フリートインスタンスにユーザーの初期設定が作成されます。Image Assistant CLI オペレーション、アプリケーションインストーラ、またはオートメーションを使用してこれらの設定を作成する場合は、Windows デフォルトユーザープロファイルを直接変更する必要があります。

Windows のデフォルトユーザープロファイルを別の Windows ユーザーのプロファイルで上書きするには、Image Assistant update-default-profile CLI オペレーションを使用することもできます。

デフォルトのアプリケーションおよび Windows の設定を設定する方法の詳細については、[Amazon AppStream 2.0 でのデフォルトのアプリケーションおよび Windows 設定とアプリケーション起動パフォーマンス](#) で Creating Default Application and Windows Settings for Your AppStream 2.0 Users を参照してください。

Image Assistant CLI オペレーションによるアプリケーションの起動パフォーマンスの最適化

AppStream 2.0 では、ユーザーのストリーミングセッションに合わせてアプリケーションの起動パフォーマンスを最適化できます。Image Assistant CLI オペレーションを使用してこれを行う場合、アプリケーションの起動に最適化するファイルを指定できます。アプリケーション最適化マニフェストにファイルを追加すると、新しいフリートインスタンスでアプリケーションを初めて起動するのにかかる時間が短縮されます。ただし、これにより、ユーザーがフリートインスタンスを利用できるようになるまでの時間が長くなります。最適化マニフェストは、アプリケーションごとの行区切りテキストファイルです。

Note

Image Assistant CLI オペレーションと Image Assistant GUI の両方を使用してアプリケーション最適化マニフェストをオンボードすると、マニフェストが結合されます。

以下は、アプリケーション最適化マニフェストファイルの例です。

```
C:\Program Files (x86)\Notepad++\autoCompletion
C:\Program Files (x86)\Notepad++\localization
C:\Program Files (x86)\Notepad++\plugins
C:\Program Files (x86)\Notepad++\themes
C:\Program Files (x86)\Notepad++\updater
C:\Program Files (x86)\Notepad++\userDefineLangs
C:\Program Files (x86)\Notepad++\change.log
C:\Program Files (x86)\Notepad++\config.xml
C:\Program Files (x86)\Notepad++\contextMenu.xml
C:\Program Files (x86)\Notepad++\doLocalConf.xml
C:\Program Files (x86)\Notepad++\functionList.xml
C:\Program Files (x86)\Notepad++\langs.model.xml
C:\Program Files (x86)\Notepad++\license.txt
C:\Program Files (x86)\Notepad++\notepad++.exe
C:\Program Files (x86)\Notepad++\readme.txt
C:\Program Files (x86)\Notepad++\SciLexer.dll
C:\Program Files (x86)\Notepad++\shortcuts.xml
C:\Program Files (x86)\Notepad++\stylers.model.xml
```


詳細については、[Amazon AppStream 2.0 でのデフォルトのアプリケーションおよび Windows 設定とアプリケーション起動パフォーマンス](#)で「アプリケーションの起動パフォーマンスの最適化」を参照してください。

Amazon AppStream 2.0 イメージをプログラムで作成するプロセスの概要

Image Assistant CLI オペレーションをアプリケーションのインストール自動化でを使用して、完全にプログラムによる AppStream 2.0 イメージ作成ワークフローを作成できます。アプリケーションのインストールの自動化が完了した後、イメージが作成される前に、Image Assistant CLI オペレーションを使用して以下を指定します。

- ユーザーが起動できる実行可能ファイル
- アプリケーション用の最適化マニフェスト
- その他の AppStream 2.0 イメージメタデータ

次の概要では、プログラムで AppStream 2.0 イメージを作成するプロセスについて説明します。

1. アプリケーションインストールの自動化を使用して、イメージビルダーに必要なアプリケーションをインストールします。このインストールには、ユーザーが起動するアプリケーション、依存関係、およびバックグラウンドアプリケーションが含まれる場合があります。
2. 最適化するファイルとフォルダーを決定します。
3. 該当する場合は、Image Assistant add-application CLI オペレーションを使用して、AppStream 2.0 イメージのアプリケーションメタデータと最適化マニフェストを指定します。
4. AppStream 2.0 イメージに追加のアプリケーションを指定するには、必要に応じてアプリケーションごとに手順 1～3 を繰り返します。
5. 該当する場合は、Image Assistant update-default-profile CLI オペレーションを使用して、デフォルトの Windows プロファイルを上書きし、ユーザーのデフォルトのアプリケーションと Windows 設定を作成します。
6. Image Assistant create-image CLI オペレーションを使用してイメージを作成します。

Amazon AppStream 2.0 イメージを作成および管理するための Image Assistant CLI オペレーション

このセクションでは、AppStream 2.0 イメージの作成と管理に使用できる Image Assistant CLI オペレーションについて説明します。

Windows Image Builder では、C:\Program Files\Amazon\Photon\ConsoleImageBuilder\Image-Assistant.exe にコマンドラインインターフェイスが含まれる実行可能ファイルがあります。便宜上、この実行可能ファイルは Windows PATH 変数に含まれています。これにより、実行可能ファイルへの絶対パスを指定せずに Image Assistant CLI オペレーションを呼び出すことができます。これらのオペレーションを呼び出すには、image-assistant.exe コマンドを入力します。

Linux Image Builder では、/usr/local/appstream/image-assistant/AppStreamImageAssistant に Image Assistant ツールがあり、シンボリックリンクは /bin/appstreamImageAssistant にあります。

help オペレーション

すべての Image Assistant CLI オペレーションのリストを取得します。リスト内の各オペレーションについて、説明と使用構文が提供されます。特定のオペレーションのヘルプを表示するには、オペレーションの名前を入力し、--help パラメータを指定します。次に例を示します。

```
add-application --help
```

概要

```
help
```

出力

利用可能なオペレーションのリストとその機能の説明を標準出力します。

add-application オペレーション

AppStream 2.0 ユーザーのアプリケーションリストにアプリケーションを追加します。このリストのアプリケーションは、アプリケーションカタログに含まれています。アプリケーションカタログは、ユーザーが AppStream 2.0 ストリーミングセッションにサインインするときに表示されます。

Note

アプリケーションの設定を変更する必要がある場合は、そのアプリケーションを削除した上で、新たな設定のアプリケーションを追加します。

概要

```
add-application
--name <value>
--absolute-app-path <value>
[--display-name <value>]
[--absolute-icon-path <value>]
[--working-directory <value>]
[--launch-parameters <"-escaped value">]
[--absolute-manifest-path <value>]
```

オプション

--name (文字列)

アプリケーションの一意の名前。最大長は 256 文字です。最大 50 個のアプリケーションを追加できます。空白文字を使用することはできません。

--absolute-app-path (文字列)

アプリケーションの実行可能ファイル、バッチファイル、またはスクリプトへの絶対パス。有効なファイルのパスを指定する必要があります。

--display-name (文字列)

アプリケーションカタログに表示されるアプリケーションの名前。表示名を指定しない場合、AppStream 2.0 は実行可能ファイル名から派生した名前を作成します。名前は、ファイル拡張子なしで、スペースの代わりにアンダースコアを使用して作成されます。最大長は 256 文字です。

--absolute-icon-path (文字列)

アプリケーションのアイコンへの絶対パスです。パスは、.jpg、.png、または .bmp のいずれかのタイプの有効なアイコンファイルを指している必要があります。最大サイズは、256 px x 256 px です。パスを指定しない場合、実行可能ファイルのデフォルトアイコンが使用可能であれば使

用されます。実行可能ファイル用のデフォルトのアイコンがない場合、デフォルトの AppStream 2.0 アプリケーションアイコンが使用されます。

--working-directory (文字列)

アプリケーション起動時のアプリケーションの最初の作業ディレクトリ。

--absolute-manifest-path (文字列)

改行で区切られた新しいテキストファイルへのパス。このファイルは、ユーザーがフリートインスタンスをストリーミングできるようにする前に最適化するファイルの絶対パスを指定します。有効なテキストファイルのパスを指定する必要があります。

メッセージ出力

終了コード	標準出力に出力されるメッセージ	説明
0	{"status": 0, "message": "Success"}	アプリケーションが正常に追加されました。
1	{"status": 1, "message": "Name is not unique"}	オペレーションを完了するには、管理者権限が必要です。
1	{"status": 1, "message": "Unable to add more than 50 apps to the catalog."}	AppStream 2.0 アプリケーションカタログに追加できるアプリケーションの最大数は 50 であるため、アプリケーションを追加できませんでした。
1	{"status": 1, "message": "Name is not unique"}	その名前のアプリケーションは AppStream 2.0 アプリケーションカタログに既に存在しています。
1	{"status": 1, "message": "File not found (absolute-app-path)"}	<code>absolute-app-path</code> で指定されたファイルが見つかりませんでした。
1	{"status": 1, "message": "Unsupported file extension"}	<code>Absolute-app-path</code> パラメーターは、 <code>.exe</code> および <code>.bat</code> のファイルタイプのみをサポートしています。

終了コード	標準出力に出力されるメッセージ	説明
1	{"status": 1, "message": "Directory not found (working-directory)"}	working-directory に指定されたディレクトリが見つかりませんでした。
1	{"status": 1, "message": "Optimization-manifest not found: <filename>"}	optimization-manifest で指定されたファイルが見つかりませんでした。
1	{"status": 1, "message": "File not found: <filename>"}	最適化マニフェスト内で指定されたファイルが見つかりませんでした。
255	{"status": 255, "message": "<error message>"}	予期しないエラーが発生しました。リクエストを再試行してください。エラーが解決しない場合は、AWS サポート にお問い合わせください。詳細については、「 AWS サポート センター 」を参照してください。

remove-application オペレーション

AppStream 2.0 イメージのアプリケーションリストからアプリケーションを削除します。アプリケーションはアンインストールまたは変更されませんが、ユーザーはこのアプリケーションを AppStream 2.0 アプリケーションカタログから起動できなくなります。

概要

```
remove-application
--name <value>
```

オプション

--name (文字列)

削除するアプリケーションの一意の識別子。

メッセージ出力

終了コード	標準出力に出力されるメッセージ	説明
0	<code>{"status": 0, "message": "Success"}</code>	アプリケーションが正常に削除されました。
1	<code>{"status": 1, "message": "Name is not unique"}</code>	オペレーションを完了するには、管理者権限が必要です。
1	<code>{"status": 0, "message": "Success"}</code>	指定されたアプリケーションが AppStream 2.0 アプリケーションカタログで見つかりませんでした。
255	<code>{"status": 255, "message": <error message>}</code>	予期しないエラーが発生しました。リクエストを再試行してください。エラーが解決しない場合は、AWS サポート にお問い合わせください。詳細については、「 AWS サポート センター 」を参照してください。

list-applications オペレーション

アプリケーションカタログで指定されているすべてのアプリケーションを一覧表示します。

概要

```
list-applications
```

メッセージ出力

終了コード	標準出力に出力されるメッセージ	説明
0	<code>{"status": 0, "message": "Success", "applications": [{ ..app1.. }, { ..app2.. }]}</code>	AppStream 2.0 アプリケーションカタログ内のアプリケーションのリスト。

終了コード	標準出力に出力されるメッセージ	説明
255	{"status": 255, "message": <error message>}	予期しないエラーが発生しました。リクエストを再試行してください。エラーが解決しない場合は、AWS サポート にお問い合わせください。詳細については、「 AWS サポート センター 」を参照してください。

update-default-profile オペレーション

指定された Windows ユーザーのプロファイルを Windows のデフォルトのユーザープロファイルにコピーします。ストリーミングする新しいユーザーは、この指定されたプロファイルの設定を継承します。

Note

Linux Image Assistant CLI ツールでは、この操作はサポートされていません。

概要

```
update-default-profile  
[--profile <value>]
```

オプション

--profile (文字列)

Windows プロファイルが Windows のデフォルトのユーザープロファイルにコピーされるユーザーの名前。名前には次の形式を使用します。

"<domain>\<username>"

イメージビルダーが Microsoft Active Directory ドメインに結合されていない場合は、ドメインの代わりにピリオド「.」を入力します。ユーザーを指定しない場合、AppStream 2.0 テンプレートのユーザーアカウントが使用されます。

メッセージ出力

終了コード	標準出力に出力されるメッセージ	説明
0	{"status": 0, "message": "Success"}	ユーザー設定はデフォルトの Windows プロファイルに正常にコピーされました。
1	{"status": 1, "message": "Name is not unique"}	オペレーションを完了するには、管理者権限が必要です。
1	{"status": 1, "message": "Unable to copy file or folder: <path>. <reason>"}	ファイルまたはフォルダーが使用できないため、ユーザー設定をコピーできませんでした。
1	{"status": 1, "message": "Cannot copy a domain user when not joined to a domain"}	Microsoft Active Directory ドメインユーザーは指定されましたが、イメージビルダーが Active Directory ドメインに結合されていません。
255	{"status": 255, "message": "<error message>"}	予期しないエラーが発生しました。リクエストを再試行してください。エラーが解決しない場合は、AWS サポート にお問い合わせください。詳細については、「 AWS サポート センター 」を参照してください。

reset-user-profile オペレーション

指定されたユーザーの Windows ユーザープロファイルを削除します。

Note

Linux Image Assistant CLI ツールでは、この操作はサポートされていません。

概要

```
reset-user-profile
[--profile <value>]
```

オプション

--profile (文字列)

Windows プロファイルが削除される Windows ユーザーの名前。名前には次の形式を使用します。

```
"<domain>\<username>"
```

イメージビルダーが Microsoft Active Directory ドメインに結合されていない場合は、ドメインの代わりにピリオド「.」を入力します。

メッセージ出力

終了コード	標準出力に出力されるメッセージ	説明
0	{"status": 0, "message": "Success"}	指定されたユーザー設定の削除が完了しました。
1	{"status": 1, "message": "Name is not unique"}	オペレーションを完了するには、管理者権限が必要です。
1	{"status": 1, "message": "Unable to copy file or folder: <path>.\n<reason>"}	ファイルまたはフォルダーが使用できないため、ユーザー設定をリセットできませんでした。
1	{"status": 1, "message": "Cannot copy a domain user when not joined to a domain"}	Microsoft Active Directory ドメインユーザーは指定されましたが、イメージビルダーが Active Directory ドメインに結合されていません。
255	{"status": 255, "message": "<error message>"}	予期しないエラーが発生しました。リクエストを再試行してください。エラーが解決しない場合は、AWS サポート にお問い合わせください。詳細については、「 AWS サポート センター 」を参照してください。

create-image オペレーション

イメージ作成ワークフローを開始し、AppStream 2.0 フリートに使用できる AppStream 2.0 イメージを作成します。

概要

```
create-image
--name <value>
[--description <value>]
[--display-name <value>]
[--enable-dynamic-app-catalog] | [--no-enable-dynamic-app-catalog]
[--use-latest-agent-version] | [--no-use-latest-agent-version]
[--tags <value>]
[--dry-run]
```

オプション

--name (文字列)

AppStream 2.0 イメージの名前。名前は、Amazon Web Services アカウントと AWS リージョン内で一意である必要があります。最大長は 100 文字です。使用できる文字は次のとおりです。

a~z、A~Z、0~9、アンダースコア (_)、ハイフン (-)、ピリオド (.)

イメージ名は、「aws」、「appstream」、「amazon」のいずれのプレフィックスでも始めることはできません。これらのプレフィックスは AWS 用に予約されています。

--description (文字列)

イメージの説明を表示します。最大長は 256 文字です。

--display-name (文字列)

イメージの名前を表示します。最大長は 256 文字です。

--enable-dynamic-app-catalog | --no-enable-dynamic-app-catalog

AppStream 2.0 動的アプリケーションフレームワークのサポートを有効または無効にします。パラメータを指定しない場合、動的アプリケーションフレームワークのサポートは有効化されません。

動的アプリケーションフレームワークは、動的アプリケーションプロバイダーの構築に使用できる AppStream 2.0 ストリーミングインスタンス内のオペレーションを提供します。動的

アプリケーションプロバイダーは、提供されているオペレーションを使用して、ユーザーがリアルタイムでアクセスできるアプリケーションのカatalogを変更できます。詳細については、「[AppStream 2.0 動的アプリケーションフレームワークを使用して動的アプリケーションプロバイダーを構築する](#)」を参照してください。

--use-latest-agent-version | --no-use-latest-agent-version

現在インストールされている AppStream 2.0 エージェントのバージョンにイメージを固定するか、常に最新のエージェントバージョンを使用するかを指定します。どちらのパラメーターも指定しない場合、イメージは現在インストールされている AppStream 2.0 エージェントのバージョンに固定されます。詳細については、「[AppStream 2.0 エージェントのバージョンを管理する](#)」を参照してください。

--tags (文字列)

イメージに関連付けるタグ。タグはキーと値のペアで構成されます。次の形式を使用します。

```
--tags "mykey" "myval" "mykey2" "myval2"
```

タグの詳細については、「[Amazon AppStream 2.0 リソースにタグを付ける](#)」を参照してください。

--dry-run (文字列)

イメージを作成せずに検証を実行します。このコマンドを使用して、作成する前にイメージに問題があるかどうかを確認します。

メッセージ出力

終了コード	標準出力に出力されるメッセージ	説明
0	{"status": 0, "message": "Success"}	イメージを作成するワークフローが正常に開始されました。
1	{"status": 1, "message": "Name is not unique"}	オペレーションを完了するには、管理者権限が必要です。
1	{"status": 1, "message": "An image with the given name already exists"}	指定された名前を使用しているイメージは、Amazon Web Services アカウントに既に存在しています。

終了コード	標準出力に出力されるメッセージ	説明
1	{"status": 1, "message": "Invalid value (tags)"}	指定されたタグは有効ではありません。
255	{"status": 255, "message": <error message>}	予期しないエラーが発生しました。リクエストを再試行してください。エラーが解決しない場合は、AWS サポート にお問い合わせください。詳細については、「 AWS サポート サポートセンター 」を参照してください。

Linux ベースのイメージを作成する

Linux ベースの Amazon AppStream 2.0 イメージは、Linux Image Builder への接続、必要なアプリケーションのインストール、デフォルトのアプリケーション設定と環境変数の作成、およびコマンドラインインターフェイス (CLI) ツールまたは Image Assistant (GUI) ツールを使用したこれらのアプリケーションのアプリケーションカタログへの追加を行うことによって作成できます。GUI ツールを開くには、アプリケーションのリストで [Image Assistant] を見つけます。

内容

- [ユーザー用のデフォルトアプリケーション設定の作成](#)
- [Linux ユーザー用のデフォルト環境変数の作成](#)
- [Linux アプリケーションの起動パフォーマンスの最適化](#)
- [セッションスクリプトの作成](#)
- [Linux 用の Image Assistant CLI ツールの使用](#)
- [ウェブカメラのサポートの有効化と無効化](#)
- [ホームフォルダのヘビーファイル同期モードの有効化と無効化](#)
- [チュートリアル: Linux ベースのカスタム AppStream 2.0 イメージを作成する](#)
- [Linux イメージの日本語サポートを有効にする](#)

ユーザー用のデフォルトアプリケーション設定の作成

ユーザーに対してデフォルトのアプリケーション設定を作成するには、以下の手順に従います。

内容

- [ステップ 1: Image Builder に Linux アプリケーションをインストールする](#)
- [ステップ 2: TemplateUser アカウントを作成する](#)
- [ステップ 3: デフォルトのアプリケーション設定を作成する](#)
- [ステップ 4: デフォルトのアプリケーション設定を保存する](#)
- [ステップ 5: デフォルトのアプリケーション設定をテストする \(オプション\)](#)
- [ステップ 6: クリーンアップ](#)

ステップ 1: Image Builder に Linux アプリケーションをインストールする

このステップでは、Linux Image Builder を接続して、Image Builder にアプリケーションをインストールします。

Image Builder にアプリケーションをインストールする

1. 以下のいずれかを実行して、Image Builder に接続します。
 - [AppStream 2.0 コンソールを使用する](#) (ウェブ接続のみ)
 - [ストリーミング URL を作成する](#) (ウェブまたは AppStream 2.0 クライアント接続用)

Note

Amazon Linux GNOME デスクトップには ImageBuilderAdmin ユーザーとしてログインされ、ルート管理者権限が付与されます。

2. 必要なアプリケーションをインストールします。例えば、パブリック yum リポジトリから Chromium ブラウザをインストールするには、まず Terminal アプリケーションを開いてから、以下のコマンドを実行します。

```
[ImageBuilderAdmin]$ sudo yum update && sudo yum install chromium.x86_64
```

ステップ 2: TemplateUser アカウントを作成する

このステップでは、TemplateUser アカウントを作成します。これにより、ストリーミングユーザー用のデフォルトのアプリケーション設定が作成されます。

TemplateUser アカウントを作成するには

1. root 権限がない TemplateUser アカウントを作成します。例えば、[Terminal] (ターミナル) ウィンドウで以下のコマンドを実行して、Image Builder に TemplateUser を作成します。

```
[ImageBuilderAdmin]$ sudo useradd -m TemplateUser
```

```
[ImageBuilderAdmin]$ echo -e '<password>\n<password>\n' | sudo passwd TemplateUser
```

2. TemplateUser アカウントに切り替えます。

```
[ImageBuilderAdmin]$ su - TemplateUser
```

ステップ 3: デフォルトのアプリケーション設定を作成する

このステップでは、AppStream 2.0 ユーザーに対してデフォルトのアプリケーション設定を作成します。これにより、ユーザーがこれらの設定を作成または構成する必要がなくなるため、AppStream 2.0 ストリーミングセッション中にアプリケーションを迅速に開始できます。

ユーザーに対してデフォルトのアプリケーション設定を作成するには

1. デフォルト設定を作成するアプリケーションを起動します。例えば、Terminal ウィンドウで以下のコマンドを実行して、Chromium ブラウザを起動します。

```
[TemplateUser]$ chromium-browser
```

2. アプリケーションの設定を行う 例えば、Chromium ブラウザのホームページを **https://aws.amazon.com** に設定します。
3. アプリケーションを閉じます。
4. ログアウト:

```
[TemplateUser]$ logout
```

ステップ 4: デフォルトのアプリケーション設定を保存する

このステップでは、/etc/skel/ ディレクトリに追加したデフォルトのアプリケーション設定をコピーし、ストリーミングユーザーが利用できるようにします。

デフォルトのアプリケーション設定を保存するには

- ターミナルウィンドウで次のコマンドを実行して、ストリーミングユーザー用のデフォルトのアプリケーション設定をコピーします。

```
[ImageBuilderAdmin]$ sudo cp -r -f /home/TemplateUser/. /etc/skel
```

ステップ 5: デフォルトのアプリケーション設定をテストする (オプション)

このステップでは、追加したアプリケーションが正しく実行され、デフォルトのアプリケーション設定が期待どおりに動作することを確認します。

Image Builder でアプリケーションとデフォルト設定をテストする

- root 権限がないテストユーザーを作成します。例えば、[Terminal] (ターミナル) ウィンドウで以下のコマンドを実行して、Image Builder に test-user を作成します。

```
[ImageBuilderAdmin]$ sudo useradd -m test-user
```

```
[ImageBuilderAdmin]$ echo -e 'password>\n<password>\n' | sudo passwd test-user
```

- テストユーザーに切り替えます。

```
[ImageBuilderAdmin]$ su - test-user
```

- アプリケーション (Chromium など) をテストユーザーとして起動します。

```
[test-user]$ /usr/bin/chromium-browser
```

- テストユーザーに対してデフォルト設定が利用可能であることを確認します (Chromium ホームページが <https://aws.amazon.com/> であるなど)。

- ログアウト:

```
[test-user]$ logout
```

ステップ 6: クリーンアップ

最後のステップは、クリーンアップです。

次をクリーンアップするには :

- TemplateUser を削除する:

```
[ImageBuilderAdmin]$ sudo killall -u TemplateUser
```

```
[ImageBuilderAdmin]$ sudo userdel -r TemplateUser
```

2. test-user を削除する (ステップ 5 をスキップした場合は不要):

```
[ImageBuilderAdmin]$ sudo killall -u test-user
```

```
ImageBuilderAdmin]$ sudo userdel -r test-user
```

Linux ユーザー用のデフォルト環境変数の作成

Linux Image Builder インスタンスに環境変数を作成することができます。環境変数を作成すると、そのイメージから作成されたストリーミングインスタンスでそれらを使用できるようになります。

Note

Linux フリートインスタンスで、Image Assistant (GUI) ツールを使用して設定された環境変数とデフォルトのシステム環境変数は、`/etc/profile.d/appstream_system_vars.sh` スクリプトを通じてエクスポートされます。これらの環境変数にアクセスするには、アプリケーションで `/etc/profile.d/appstream_system_vars.sh` スクリプトを明示的にソースにする必要があります。

ユーザー用の環境変数を作成する

1. `/etc/profile` フォルダが存在しない場合は、以下のコマンドを実行してフォルダを作成します。

```
[ImageBuilderAdmin]$ sudo mkdir -p /etc/profile.d
```

2. このフォルダ内に新しいシェルスクリプトファイル (`my-environment.sh` など) を作成するには、以下のコマンドを実行します。

```
[ImageBuilderAdmin]$ vim my-environment.sh
```

3. スクリプトファイルの最初の行に、以下のコンテンツを追加します。

```
#!/bin/sh
```

4. 後続の各行に、イメージの環境変数を設定するための `export` コマンドを追加します。以下の例は、`PATH` 変数に `$HOME/bin` を追加します。

```
export PATH="$HOME/bin:$PATH"
```

5. Esc キーを押して vim のコマンドモードに戻ってから、以下のコマンドを実行してスクリプトを保存し、vim を終了します。

```
:x
```

6. 以下のコマンドを実行して、プログラムとしてのスクリプトの実行を許可します。

```
[ImageBuilderAdmin]$ chmod +x my-environment.sh
```

Linux アプリケーションの起動パフォーマンスの最適化

Image Assistant GUI ツールを使用している場合、アプリケーションの起動パフォーマンスが自動的に最適化されます。

Image Assistant CLI を使用している場合は、以下のステップに従って起動パフォーマンスを手動で最適化します。アプリケーション最適化マニフェストを作成してファイルを追加すると、新しいフリートインスタンスでの初回起動時におけるアプリケーションの起動がより迅速になります。ただし、これにより、ユーザーがフリートインスタンスを利用できるようになるまでの時間が長くなります。最適化マニフェストは、アプリケーションごとに 1 つある行区切りのテキストファイルです。

マニフェストファイル (<*your-app*>-manifest.txt など) は、手動で作成する、または以下の手順に従って作成することができます。

マニフェストファイルを作成する

1. 最適化しようとしているアプリケーションが起動済みで実行中であることを確認します。
2. Linux Image Builder のターミナルから、以下のコマンドを実行します。

```
ps -ef | grep <application-process-name>
```

3. 最後のステップの出力で最小の PID 番号を検索します。これは、アプリケーションのルート親プロセスの PID です。
4. アプリケーションが実行されている状態を維持して、ユーザーが必要とする初期コンポーネントを使用するようにしてください。そうすることで、これらのコンポーネントが最適化プロセスによってキャプチャされることを確実にします。
5. 以下のコンテンツを使用して、スクリプトファイル (~/.getfilestool.sh など) を作成します。


```
#!/bin/bash
## usage getfilestool.sh $pid
lsof -p $(pstree -p $1 | grep -o '([0-9]\+)' | grep -o '[0-9]\+' | tr '\012' ,)|
grep REG | sed -n '1!p' | awk '{print $9}'|awk 'NF'
```

6. 以下のコマンドを使用して、ファイルが実行可能であることを確認します。

```
[ImageBuilderAdmin]$ chmod u+x ~/getfilestool.sh
```

7. 以下のコマンドを実行して、ステップ 3 で検索したルート親プロセスからの実行中ファイルのすべてをキャプチャし、一時的なマニフェストファイルに保存します。

```
[ImageBuilderAdmin]$ sudo ~/getfilestool.sh <root-parent-pid> > /tmp/<your-app>-manifest.txt
```

8. 最適化マニフェストの内容を確認します。最適化マニフェストは、アプリケーションごとの行区切りテキストファイルです。

最適化マニフェストは、Image Assistant コマンドラインインターフェイス (CLI) ツールを使用して、アプリケーションごとに指定することができます。詳細については、「[the section called “Linux 用の Image Assistant CLI ツールの使用”](#)」を参照してください。

セッションスクリプトの作成

AppStream 2.0 は、Windows ベースおよび Linux ベースのストリーミングインスタンスの両方でオンラインインスタンスセッションスクリプトを提供します。セッションスクリプトの詳細については、[the section called “ユーザーのストリーミングエクスペリエンスを管理するセッションスクリプト”](#)を参照してください。

セッションスクリプトは AppStream 2.0 イメージ内で指定されます。Linux インスタンス上のセッションスクリプト設定ファイルを見つけるには、`/opt/appstream/SessionScripts/config.json` に移動します。以下のコードは、「test-session-start」という名前のセッション開始スクリプトと、「test-session-stop」という名前のセッション終了スクリプトを、それらのランタイムパラメータと共に指定するサンプル config.json ファイルです。config.json で参照されているスクリプトに実行許可があり、コマンドインタプリタが定義されていることを確認します (例: `#!/bin/bash`)。

```
{
  "SessionStart": {
    "Executables": [
```

```
{
    {
        "Context": "system",
        "Filename": "/opt/appstream/SessionScripts/test-session-start",
        "Arguments": "arg1",
        "S3LogEnabled": true
    }
},
    "WaitingTime": 30
},
    "SessionTermination": {
        "Executables": [
            {
                "Context": "system",
                "Filename": "/opt/appstream/SessionScripts/test-session-stop",
                "Arguments": "arg2",
                "S3LogEnabled": true
            }
        ],
        "WaitingTime": 30
    }
}
```

Linux 用の Image Assistant CLI ツールの使用

Linux ベースの Image Builder では、Image Assistant CLI ツールである AppStreamImageAssistant を使用して、AppStream 2.0 イメージを作成し、管理することができます。このツールは `/usr/local/appstream/image-assistant/AppStreamImageAssistant` にあり、シンボリックリンクは `/bin/AppStreamImageAssistant` にあります。Linux 用のこの CLI ツールは、Windows 用の Image Assistant CLI ツールと同じ操作を多数サポートしています。これらの操作の詳細については、[the section called “Image Assistant CLI オペレーション”](#) を参照してください。

ウェブカメラのサポートの有効化と無効化

AppStream 2.0 は、ローカルウェブカメラ動画入力を AppStream 2.0 ストリーミングセッションにリダイレクトすることにより、リアルタイム音声動画 (AV) をサポートしています。この機能により、ユーザーは AppStream 2.0 ストリーミングセッション内でビデオ会議や音声会議にローカルウェブカメラを使用できます。リアルタイム AV とリアルタイムオーディオのサポートにより、ユーザーは AppStream 2.0 ストリーミングセッションを離れることなく、使い慣れたビデオおよび音声会議アプリケーションを使用して共同作業を行うことができます。

この機能を使用するには、2022 年 9 月 21 日以降にリリースされた Linux AppStream 2.0 エージェントを利用する Linux AppStream 2.0 イメージを使用する必要があります。

 Note

リアルタイム AV は、Rocky Linux または Red Hat Enterprise Linux を搭載した stream.standard.small インスタンスではサポートされていません。クライアントツールバーにカメラアイコンとマイクアイコンが表示されません。

リアルタイム AV 機能は、Linux ストリーミングセッションではデフォルトで有効になっています。Linux Image Builder でユーザーのウェブカメラのアクセス許可を設定するには、`/etc/appstream/appstream.conf` を作成し、次のコンテンツを追加します。

 Note

ウェブカメラを有効にする場合は **1** を、無効にする場合は **0** を指定します。

```
[webcam]
permission = 1
```

ホームフォルダのヘビーファイル同期モードの有効化と無効化

組織の Amazon Simple Storage Service ホームフォルダオプションを有効にできます。AppStream 2.0 スタックで Amazon S3 ホームフォルダを有効にすると、アプリケーションストリーミングセッション中に、スタックのユーザーは永続的ストレージのフォルダにアクセスできます。ユーザーがホームフォルダにアクセスするために必要な設定はありません。ユーザーがホームフォルダに保存したデータは、の Amazon S3 バケットに自動的にバックアップされ AWS アカウント、以降のセッションでそれらのユーザーが利用できるようになります。詳細については、「[the section called “ホームフォルダを管理する”](#)」を参照してください。

ユーザーがストリーミングインスタンスからホームフォルダに大きなテキストファイルを保存する際に、スムーズな利用を保証し、一貫性のないファイル同期が発生する可能性がある既存の制限に対処するため、AppStream 2.0 の管理者は、Amazon S3 への大きなファイルのアップロードが AppStream 2.0 の使用中に一般的なユーザー・シナリオである場合、`[heavy_sync]` 設定オプションをオンにすることができます。このオプションをオンにすると、ホームフォルダのファイル同期プロセス

スにレイテンシーが追加される可能性があります、Amazon S3 へのすべての同期の完全性が保証されます。

この機能は、2024 年 9 月 12 日以降にリリースされたすべての Red Hat Enterprise Linux イメージ、および Linux AppStream 2.0 エージェントを使用する Linux AppStream 2.0 イメージで使用できます。

Red Hat Enterprise Linux および Amazon Linux ストリーミングセッションでは、重同期機能はデフォルトで無効になっています。Red Hat Enterprise Linux または Amazon Linux Image Builder でユーザーの重同期のアクセス許可を設定するには、`/etc/appstream/appstream.conf` を作成し、次のコンテンツを追加します。

 Note

1 を指定して重同期を有効にするか、**0** を指定して重同期を無効にします。

```
[storage]
heavy_sync = 1
```

チュートリアル: Linux ベースのカスタム AppStream 2.0 イメージを作成する

このチュートリアルでは、ユーザーにストリーミングできるアプリケーションが含まれた、Linux ベースのカスタム Amazon AppStream 2.0 イメージを作成する方法を説明します。

 Important

Image Builder に「as2-streaming-user」という名前のユーザーを作成しないでください。これはフリートの予約済みユーザー名です。AppStream 2.0 ワークフローの外部でこのユーザー名を作成すると、フリートでストリーミングの問題が発生する可能性があります。

内容

- [ステップ 1: Image Builder に Linux アプリケーションをインストールする](#)
- [ステップ 2: アプリケーション最適化マニフェストファイルを生成する](#)
- [ステップ 3: AppStream 2.0 アプリケーションカタログを作成する](#)

- [ステップ 4: デフォルトのアプリケーション設定と環境変数を作成する](#)
- [ステップ 5: アプリケーションと設定をテストする](#)
- [ステップ 6: イメージの作成を完了する](#)
- [ステップ 7 \(オプション\): イメージにタグを付け、コピーする](#)
- [ステップ 8: クリーンアップ](#)

ステップ 1: Image Builder に Linux アプリケーションをインストールする

このステップでは、Linux Image Builder を接続して、Image Builder にアプリケーションをインストールします。

Image Builder にアプリケーションをインストールする

1. 以下のいずれかを実行して、Image Builder に接続します。
 - [AppStream 2.0 コンソールを使用する](#) (ウェブ接続のみ)
 - [ストリーミング URL を作成する](#) (ウェブまたは AppStream 2.0 クライアント接続用)

Note

Amazon Linux GNOME デスクトップには ImageBuilderAdmin ユーザーとしてログインされ、ルート管理者権限が付与されます。

2. 必要なアプリケーションをインストールします。例えば、パブリック yum リポジトリから Chromium ブラウザをインストールするには、まず Terminal アプリケーションを開いてから、以下のコマンドを実行します。

```
[ImageBuilderAdmin]$ sudo yum update && sudo yum install chromium.x86_64
```

Note

アプリケーションのダウンロードおよびインストールは、信頼済みサイトからのみ行います。

ステップ 2: アプリケーション最適化マニフェストファイルを生成する

このステップでは、手順 1 でインストールしたアプリケーションごとにマニフェストファイルを生成します。

アプリケーションの起動パフォーマンスを最適化するためのマニフェストファイルを生成する

1. 最適化しようとしているアプリケーション (Chromium など) が起動済みで実行中であることを確認します。
2. Terminal ウィンドウで以下のコマンドを実行して、アプリケーションに関連するプロセスをリストします。

```
[ImageBuilderAdmin]$ ps -ef | grep chromium
```

3. 上記のコマンドの出力から、ルート親 PID を見つけます。以下はサンプル出力で、ルート親 PID は 16712 です。

Example

```
[ImageBuilderAdmin]$ ps -ef | grep chromium
```

```
ImageBu+ 16712 4128 0 Aug26 ? 00:00:44 /usr/lib64/chromium- browser/chromium-  
browser --enable-plugins --enable-extensions -- enable-user- scripts --enable-  
printing --enable-gpu-rasterization -- enable-sync --auto-ssl- client-auth
```

```
ImageBu+ 16726 16712 0 Aug26 ? 00:00:00 /usr/lib64/chromium- browser/chromium-  
browser --type=zygote --no-zygote-sandbox ImageBu+ 16727 16712 0 Aug26 ? 00:00:00 /  
usr/lib64/chromium- browser/chromium- browser --type=zygote
```

```
ImageBu+ 16731 16727 0 Aug26 ? 00:00:00 /usr/lib64/chromium- browser/chromium-  
browser --type=zygot
```

4. アプリケーションが実行されている状態を維持して、ユーザーが必要とする初期コンポーネントを使用するようにしてください。そうすることで、これらのコンポーネントが最適化プロセスによってキャプチャされることを確実にします。
5. 以下のコンテンツを使用して、スクリプトファイル (~/getfilestool.sh など) を作成します。

```
#!/bin/bash  
## usage getfilestool.sh $pid
```

```
ls -l | grep -o '([0-9]\+)' | grep -o '[0-9]\+' | tr '\012' , |  
grep REG | sed -n '1!p' | awk '{print $9}' | awk 'NF'
```

6. 以下のコマンドを実行して、ファイルが実行可能であることを検証します。

```
[ImageBuilderAdmin]$ chmod u+x ~/getfilestool.sh
```

7. 以下のコマンドを実行して、上記のステップ 3 で見つかったルート親プロセスからの実行中ファイルをすべてキャプチャし、一時的なマニフェストファイルに保存します。

```
[ImageBuilderAdmin]$ sudo ~/getfilestool.sh 16712 > /tmp/chromium-manifest.txt
```

8. 最適化マニフェストの内容を確認します。最適化マニフェストは、アプリケーションごとの行区切りテキストファイルです。

ステップ 3: AppStream 2.0 アプリケーションカタログを作成する

このステップでは、Image Builder で CLI ツール AppStreamImageAssistant を使用して、イメージ用のアプリケーションを指定することで AppStream 2.0 アプリケーションカタログを作成します。ストリーミングする予定のアプリケーションごとに、名前、表示名、起動する実行可能ファイル、および表示するアイコンを指定できます。

AppStream 2.0 アプリケーションカタログを作成する

1. Image Builder デスクトップで、サイドパネルから、またはアプリケーショングリッドを開いて [Terminal] (ターミナル) を開きます。
2. AppStreamImageAssistant --help を実行して、使用可能なコマンドのリストを表示します。これらのコマンドを使用して、アプリケーションを追加し、イメージを作成します。
3. 以下のコマンドを実行して、インストール済みのアプリケーション (Chromium など) を AppStream 2.0 ユーザーのアプリケーションリストに追加します。

```
AppStreamImageAssistant add-application \  
--name Chromium \  
--absolute-app-path /usr/lib64/chromium-browser/chromium-browser \  
--display-name Chromium \  
--absolute-icon-path /usr/share/icons/hicolor/256x256/apps/chromium-browser.png \  
--absolute-manifest-path /tmp/chromium-manifest.txt
```

または、以下のコマンドを実行します。

```
AppStreamImageAssistant add-application \  

```

```
--name="Chromium" \  
--absolute-app-path="/usr/lib64/chromium-browser/chromium-browser" \  
--display-name="Chromium" \  
--absolute-icon-path="/usr/share/icons/hicolor/256x256/apps/chromium-browser.png" \  
\  
--absolute-manifest-path="/tmp/chromium-manifest.txt"
```

4. 他のアプリケーションを追加するには、追加のアプリケーションごとにステップ 3 を繰り返します。
5. カタログに追加されたアプリケーションのリストを、アイコンパスや起動パラメータといったメタデータと共に表示するには、以下のコマンドを実行します。

```
AppStreamImageAssistant list-applications
```

6. アプリケーションをカタログから削除するには、以下のコマンドを実行します。

```
AppStreamImageAssistant remove-application --name application_name
```

ステップ 4: デフォルトのアプリケーション設定と環境変数を作成する

このステップでは、AppStream 2.0 ユーザー用のデフォルトのアプリケーション設定と環境変数を作成します。これにより、ユーザーがこれらの設定を作成または構成する必要がなくなるため、AppStream 2.0 ストリーミングセッション中にアプリケーションを迅速に開始できます。

ユーザー用のデフォルトのアプリケーション変数と環境変数を作成する

1. デフォルト設定を作成するアプリケーションを起動します。例えば、Terminal ウィンドウで以下のコマンドを実行して、Chromium ブラウザを起動します。

```
[ImageBuilderAdmin]$ chromium-browser
```

2. アプリケーションの設定を行う 例えば、Chromium ブラウザのホームページを **https://aws.amazon.com** に設定します。
3. Chromium アプリケーションが閉じられていることを確認し、次のコマンドを実行して Chromium の設定を /etc/skel にコピーします。

```
[ImageBuilderAdmin]$ sudo mkdir /etc/skel/.config
```

```
[ImageBuilderAdmin]$ sudo cp -R ~/.config/chromium /etc/skel/.config
```

4. 環境変数を設定して、スクリプトファイルに追加します。例えば、以下のコマンドを実行します。


```
[ImageBuilderAdmin]$ echo "export FOO=BAR" | sudo tee -a /etc/profile.d/myenvvars.sh
```

```
[ImageBuilderAdmin]$ sudo chmod +x /etc/profile.d/myenvvars.sh
```

ステップ 5: アプリケーションと設定をテストする

このステップでは、追加したアプリケーションが正しく実行され、デフォルトのアプリケーション設定と環境変数が期待どおりに動作することを確認します。

Image Builder でアプリケーションとデフォルト設定をテストする

1. root 権限がないテストユーザーを作成します。例えば、[Terminal] (ターミナル) ウィンドウで以下のコマンドを実行して、Image Builder に test-user を作成します。

```
[ImageBuilderAdmin]$ sudo useradd -m test-user
```

```
[ImageBuilderAdmin]$ echo -e 'Pa55w0rdas2!!!\nPa55w0rdas2!!!\n' | sudo passwd test-user
```

2. テストユーザーに切り替えます。

```
[ImageBuilderAdmin]$ su - test-user
```

3. アプリケーション (Chromium など) をテストユーザーとして起動します。

```
[test-user]$ /usr/bin/chromium-browser
```

4. テストユーザーに対してデフォルト設定が利用可能であることを確認します (Chromium ホームページが <https://aws.amazon.com/> であるなど)。
5. テストユーザーに対して環境変数が利用可能であることを確認します。例えば、以下のコマンドを使用します。

```
[test-user]$ echo $FOO
```

このコマンドは、Terminal に出力 **BAR** を表示します。

6. この Image Builder からイメージを作成する前に、以下のコマンドを実行してテストユーザーを削除します。

```
# logout test user
```

```
[test-user]$ logout
```

```
# kill test user's running processes
```

```
[ImageBuilderAdmin]$ sudo killall -u test-user
```

```
# delete user
```

```
[ImageBuilderAdmin]$ sudo userdel -r test-user
```

ステップ 6: イメージの作成を完了する

このステップでは、イメージの名前を選択してイメージの作成を完了します。

イメージを作成する

1. [Terminal] (ターミナル) ウィンドウで AppStreamImageAssistant create-image を実行して、Image Builder からイメージを作成します。このイメージには、インストール済みかつ登録済みのアプリケーションに加えて、設定したセッションスクリプトとデフォルトのアプリケーション設定が含まれています。

利用可能なオプションのリストを表示するには、AppStreamImageAssistant create-image --help を実行します。詳細については、[the section called “プログラムを使用した イメージの作成”](#) で create-image 操作を参照してください。

2. 数秒後にリモートセッションが切断されます。[Lost Connectivity] メッセージが表示されたら、ブラウザタブを閉じます。イメージが作成されると、Image Builder のステータスが [Snapshotting] と表示されます。このプロセスが終了するまで Image Builder に接続することはできません。
3. コンソールに戻り、[Images]、[Image Registry] の順に移動します。新しいイメージがリストに表示されていることを確認します。

イメージの作成中、イメージのステータスは、コンソールのイメージレジストリに [Pending (保留中)] と表示されます。[Pending] (保留中) ステータスのイメージに接続することはできません。status。

4. ステータスを更新するには、[Refresh] (更新) アイコンをクリックします。イメージが作成されたら、イメージのステータスは [Available] に変わり、Image Builder が自動的に停止されます。

イメージの作成を続行するには、Image Builder を開始してコンソールから接続するか、新しい Image Builder を作成することができます。

ステップ 7 (オプション): イメージにタグを付け、コピーする

イメージの作成中または作成後に 1 つ以上のタグをイメージに追加できます。イメージは、同じリージョン内でコピーする、または同じ Amazon Web Services アカウント内の新しいリージョンにコピーすることもできます。ただし、ソースイメージをコピーすると、送信先イメージは同一ですが区別されます。AWS はユーザー定義タグをコピーしません。また、コピーできるのは独自に作成したカスタムイメージのみであり、AWS が提供するベースイメージはコピーできません。

Note

同時に 2 つのイメージをコピー先にコピーできます。イメージのコピー先がイメージの制限に達している場合は、エラーが発生します。この場合にイメージをコピーするには、最初にコピー先からイメージを削除する必要があります。コピー先がイメージのクォータ (制限とも呼ばれます) を下回ったら、ソースリージョンからイメージのコピーを開始します。詳細については、「[Amazon AppStream 2.0 Service Quotas](#)」を参照してください。

既存のイメージにタグを追加するには

1. ナビゲーションペインで、[イメージ]、[イメージレジストリ] の順に選択します。
2. イメージのリストで、タグを追加するイメージを選択します。
3. [Tags] (タグ)、[Add/Edit Tags] (タグの追加/編集) の順に選択してから、[Add Tag] (タグを追加する) をクリックします。タグのキーと値を指定してから、[Save] (保存) をクリックします。

詳細については、「[Amazon AppStream 2.0 リソースにタグを付ける](#)」を参照してください。

イメージをコピーするには

地理的に異なるリージョン間でイメージをコピーすると、同じイメージに基づいて複数のリージョンからアプリケーションをストリーミングできます。ユーザーに近い場所でアプリケーションをストリーミングすることによって、AppStream 2.0 を使用したアプリケーションのストリーミングに関するユーザーのエクスペリエンスを向上させることができます。

1. ナビゲーションペインで、[イメージ]、[イメージレジストリ] の順に選択します。
2. イメージのリストで、コピーするイメージを選択します。
3. アクション、コピーの順に選択します。
4. [Copy Image (イメージのコピー)] ダイアログボックスで、以下の情報を指定し、[Copy Image (イメージのコピー)] を選択します。

- [送信先リージョン] で、新しいイメージのコピー先となるリージョンを選択します。
 - [名前] に、イメージが送信先にコピーされるときの名前を指定します。
 - [説明] (オプション) に、イメージが送信先にコピーされるときの説明を指定します。
5. コピー操作の進行状況を確認するには、コンソールに戻り、[イメージ]、[イメージレジストリ] に移動します。ナビゲーションバーを使用して、コピー先のリージョンに切り替え (該当する場合)、新しいイメージがイメージのリストに表示されることを確認します。

新しいイメージは、コンソールのイメージレジストリで、最初のステータスがコピー中として表示されます。イメージが正常に作成されると、イメージのステータスが [Available] に変わります。つまり、イメージを使用してスタックを起動し、アプリケーションをストリーミングすることができます。

ステップ 8: クリーンアップ

最後に、実行中の Image Builder を停止してリソースを解放し、アカウントに意図しない料金が発生しないようにすることができます。使用していない、実行中の Image Builder を停止することをお勧めします。詳細については、[AppStream 2.0 の料金](#)をご覧ください。

実行中の Image Builder を停止するには

1. ナビゲーションペインで、[Images]、[Image Builders] の順に選択し、実行中の Image Builder インスタンスを選択します。
2. [Actions]、[Stop] の順に選択します。

Linux イメージの日本語サポートを有効にする

このチュートリアルでは、Linux イメージの日本語サポートを有効にする方法について説明します。これにより、イメージ上のアプリケーションは日本語の文字を表示でき、ストリーミングユーザーはイメージからのストリーミングセッションで日本語入力方法を使用できるようになります。

内容

- [ステップ 1: 日本語フォントと入力方法をインストールする](#)
- [ステップ 2: システムタイムゾーンを設定する](#)
- [ステップ 3: システムロケールと表示言語を設定する](#)
- [ステップ 4: 入力方法を設定する](#)

- [ステップ 5: キーボードレイアウトを設定する](#)
- [ステップ 6: Image Builder で検証を行う](#)
- [ステップ 7: イメージを作成する](#)

ステップ 1: 日本語フォントと入力方法をインストールする

このステップでは、Linux Image Builder に接続し、任意のフォントパッケージと入力方法パッケージをインストールします。

日本語フォントと入力方法をインストールするには

1. 以下のいずれかを実行して、Image Builder に接続します。
 - [AppStream 2.0 コンソールを使用する](#) (ウェブ接続のみ)
 - [ストリーミング URL を作成する](#) (ウェブまたは AppStream 2.0 クライアント接続用)

Note

Amazon Linux GNOME デスクトップには ImageBuilderAdmin ユーザーとしてログインされ、ルート管理者権限が付与されます。

2. 必要なフォントと入力方法をインストールします。これを行うには、ターミナルアプリケーションを開き、次のコマンドを実行します。

```
sudo yum install vlgothic-p-fonts.noarch
```

```
sudo yum install ibus-kkc.x86_64
```

3. 上記のコマンドに加えて、Rocky Linux および Red Hat Enterprise Linux の場合は、次のコマンドを実行します。

```
sudo yum install glibc-langpack-ja
```

ステップ 2: システムタイムゾーンを設定する

システムタイムゾーンを設定するには、次のコマンドを実行します。

```
sudo timedatectl set-timezone "Asia/Tokyo"
```

ステップ 3: システムロケールと表示言語を設定する

システムロケールと表示言語を設定するには、次のコマンドを実行します。

システムロケールと表示言語を設定するには

1. `sudo vim /etc/cloud/cloud.cfg` コマンドを実行して `cloud-init config` ファイルを更新し、`[ロケール]` を `[locale: ja_jp.utf8]` に変更し、ファイルを保存して閉じます。
2. `sudo localectl set-locale LANG=ja_JP.utf8` を実行してシステム設定を更新します。
3. `sudo gsettings set org.gnome.system.locale region "ja_JP.utf8"` を実行して Gnome シェル設定を更新します。

ステップ 4: 入力方法を設定する

イメージに追加するアプリケーションの入力方法を設定します。アプリケーションのインストール方法、マニフェストファイルの生成方法、およびデフォルト設定の作成方法の詳細については、「[チュートリアル: Linux ベースのカスタム AppStream 2.0 イメージを作成する](#)」を参照してください。このステップでは、`/usr/local/firefox/firefox` にある Firefox アプリケーションが既にインストールされていることを前提としています。

入力方法を設定するには

1. `sudo vim /usr/local/bin/update-input-method.sh` コマンドを実行してスクリプトを作成します。スクリプトに次のコンテンツを追加します。

```
#!/bin/bash

function start_process()
{
    command=$1
    process_name=$2

    process_count=$(pgrep $process_name -c)
    echo "$(date) current $process_name count: $process_count"
    while [ $process_count -lt 1 ]
    do
        echo "$(date) starting $process_name"
        eval $command
        sleep 1
        process_count=$(pgrep $process_name -c)
    done
}
```

```
done
echo "$(date) $process_name started"
}

start_process "ibus-daemon --xim &" "ibus-daemon"
start_process "/usr/libexec/ibus-engine-kkc --ibus &" "ibus-engine-kkc"

gsettings set org.gnome.desktop.input-sources sources "[('ibus','kkc'), ('xkb',
'us')]"
gsettings set org.gnome.desktop.wm.keybindings switch-input-source
"['<Control>space']"
gsettings set org.gnome.desktop.wm.keybindings switch-input-source-backward
"['<Shift><Control>space']"

echo "$(date) updated input source and switch shortcut"
```

上記のスクリプトでは、最初の入力ソース ('ibus', 'kkc') がデフォルトの入力方法です。デフォルトの入力方法を変更するには、入力ソースの順序を変更します。また、入力方法を切り替えるためのショートカットキーの組み合わせとして、Control + Space と Shift + Control + Space が指定されています。ユーザーがストリーミングセッション中に入力方法を切り替えるために使用できる、独自のキーの組み合わせを指定できます。

2. イメージに追加するアプリケーション (Firefox) を起動するためのスクリプトを作成します。これを行うには、`sudo vim /usr/local/bin/firefox-jp.sh` コマンドを実行し、スクリプトに以下のコンテンツを追加します。

```
#!/bin/bash

/usr/local/bin/update-input-method.sh > /var/tmp/update-input-method.log 2>&1 &

/usr/local/firefox/firefox &
```

3. 次のコマンドを実行して、両方のスクリプトに実行許可を追加します。

```
sudo chmod +x /usr/local/bin/update-input-method.sh
```

```
sudo chmod +x /usr/local/bin/firefox-jp.sh
```

4. アプリケーションの最適化マニフェストファイルを既に作成している場合は、次のコマンドを実行して、アプリケーション起動スクリプトをアプリケーションカタログに追加します。

```
sudo AppStreamImageAssistant add-application \  
--name firefox \  
--absolute-app-path /usr/local/bin/firefox-jp.sh \  
--display-name firefox \  
--absolute-icon-path /usr/local/firefox/browser/chrome/icons/default/default128.png \  
\  
--absolute-manifest-path /tmp/firefox-manifest.txt
```

または、update-input-method.sh スクリプトをイメージのアプリケーションカタログに別のアプリケーションとして追加して、入力方法を設定することもできます。ストリーミングセッション中、ユーザーはこのアプリケーションを起動して日本語入力を有効にし、同じセッション内で指定したショートカットキーを使用して入力方法を切り替えることができます。

ステップ 5: キーボードレイアウトを設定する

ストリーミングセッション中にユーザーが使用するキーボードに合わせてキーボードレイアウトを設定します。localectl list-keymaps コマンドを使用して、使用可能なすべてのキーマップを一覧表示し、sudo localectl set-keymap jp106 コマンドを使用して、106 キーなどの日本語キーボードにキーマップを設定できます。

ステップ 6: Image Builder で検証を行う

Image Builder で検証を行うには、最初に sudo shutdown -r now コマンドを実行して Image Builder を再起動します。再起動後、Image Builder に再度接続し、タイムゾーン、ロケール、言語、入力方法などのすべてが想定どおりに機能することを確認します。

ステップ 7: イメージを作成する

Image Builder にイメージを作成します。詳細については、「[チュートリアル: Linux ベースのカスタム AppStream 2.0 イメージを作成する](#)」を参照してください。必ず、先ほど指定したリージョン設定を含む、デフォルトのアプリケーション設定を作成します。詳細については、[Linux ベースのイメージを作成する](#)の「ユーザー用のデフォルトアプリケーション設定の作成」を参照してください。

このイメージから作成されたすべての Linux フリートインスタンスは、イメージに対して設定したデフォルトのタイムゾーン、ロケール、言語、および入力方法の設定と同じになります。

セッションスクリプトを使用して Amazon AppStream 2.0 ユーザーのストリーミングエクスペリエンスを管理する

AppStream 2.0 には、インスタンスセッションスクリプトが用意されています。ユーザーのストリーミングセッションで特定のイベントが発生したときに、これらのスクリプトを使用して独自のカスタムスクリプトを実行できます。たとえば、ユーザーのストリーミングセッションが開始される前に、カスタムスクリプトを使用して AppStream 2.0 環境を準備できます。ユーザーがストリーミングセッションを完了した後に、カスタムスクリプトを使用してストリーミングインスタンスをクリーンアップすることもできます。

セッションスクリプトは AppStream 2.0 イメージ内で指定されます。これらのスクリプトはユーザーコンテキストまたはシステムコンテキスト内で実行されます。セッションスクリプトが情報、エラー、またはデバッグメッセージの書き込みに標準出力を使用する場合は、オプションで、それらを Amazon Web Services アカウント内の Amazon S3 バケットに保存することができます。

内容

- [ストリーミングセッションの開始前にスクリプトを実行する](#)
- [ストリーミングセッションの終了後にスクリプトを実行する](#)
- [セッションスクリプトを作成および指定する](#)
- [セッションスクリプト設定ファイル](#)
- [Windows PowerShell ファイルの使用](#)
- [セッションスクリプト出力のログ記録](#)
- [セッションスクリプトでストレージコネクタを使用する](#)
- [セッションスクリプトログに対して Amazon S3 バケットストレージを有効にする](#)
- [マルチセッションフリートでセッションスクリプトを使用する](#)

ストリーミングセッションの開始前にスクリプトを実行する

ユーザーのアプリケーションが起動されてストリーミングセッションが開始されるまでに最大 60 秒間実行されるようにスクリプトを設定できます。それにより、ユーザーがアプリケーションのストリーミングを開始する前に AppStream 2.0 環境をカスタマイズできます。セッションスクリプトが実行されると、読み込みスピナーがユーザーに表示されます。スクリプトが正常に完了するか、最大待機時間が経過すると、ユーザーのストリーミングセッションが開始されます。スクリプトが正常に完了しなかった場合は、エラーメッセージがユーザーに表示されます。ただし、ユーザーはストリーミングセッションの使用を禁止されません。

Windows インスタンスでファイル名を指定するときは、ダブルバックスラッシュを使用する必要があります。例えば、次のようになります。

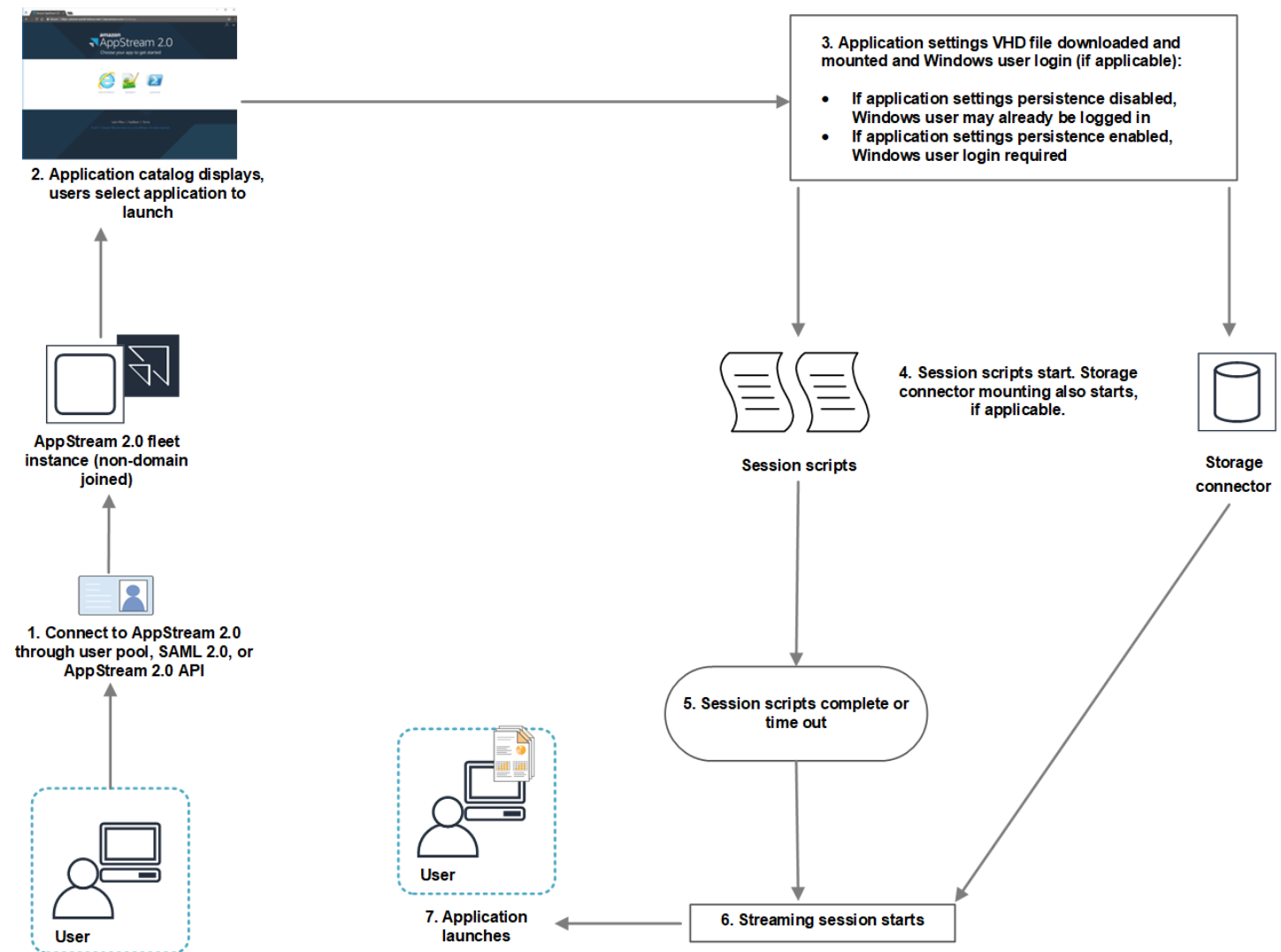
```
C:\\Scripts\\Myscript.bat
```

二重のバックスラッシュを使用しないと、.json ファイル形式が正しくないことを示すエラーが表示されます。

 Note

スクリプトは正常に完了したら、値 0 を返します。スクリプトが 0 以外の値を返した場合、AppStream 2.0 はユーザーにエラーメッセージを表示します。

ストリーミングセッションの開始前にスクリプトを実行し、AppStream 2.0 の動的アプリケーションフレームワークが有効になっていないと、以下のプロセスが発生します。



1. ユーザーはドメインに参加していない AppStream 2.0 フリートインスタンスに接続します。この接続には、以下のいずれかのアクセス方法を使用します。

- AppStream 2.0 ユーザープール
- SAML 2.0
- AppStream 2.0 API

2. アプリケーションカタログが AppStream 2.0 ポータルに表示され、ユーザーは起動するアプリケーションを選択します。

3. 以下のいずれかのプロセスが発生します。

- ユーザーに対してアプリケーション設定の永続化が有効になっている場合は、ユーザーのカスタマイズ内容と Windows の設定内容を保存しているアプリケーション設定の Virtual Hard Disk

(VHD) ファイルがダウンロードされてマウントされます。この場合は、Windows ユーザーのログインが必要です。

アプリケーション設定の永続化については、[AppStream 2.0 ユーザーのアプリケーション設定の永続化を有効にする](#) を参照してください。

- アプリケーション設定の永続化が有効になっていない場合、Windows ユーザーはすでにログインしています。
4. セッションスクリプトが起動されます。ユーザーに対して永続的ストレージが有効になっている場合は、ストレージコネクタのマウントも開始されます。永続的ストレージについては、[AppStream 2.0 ユーザーに対して永続的ストレージを有効にして管理する](#) を参照してください。

 Note

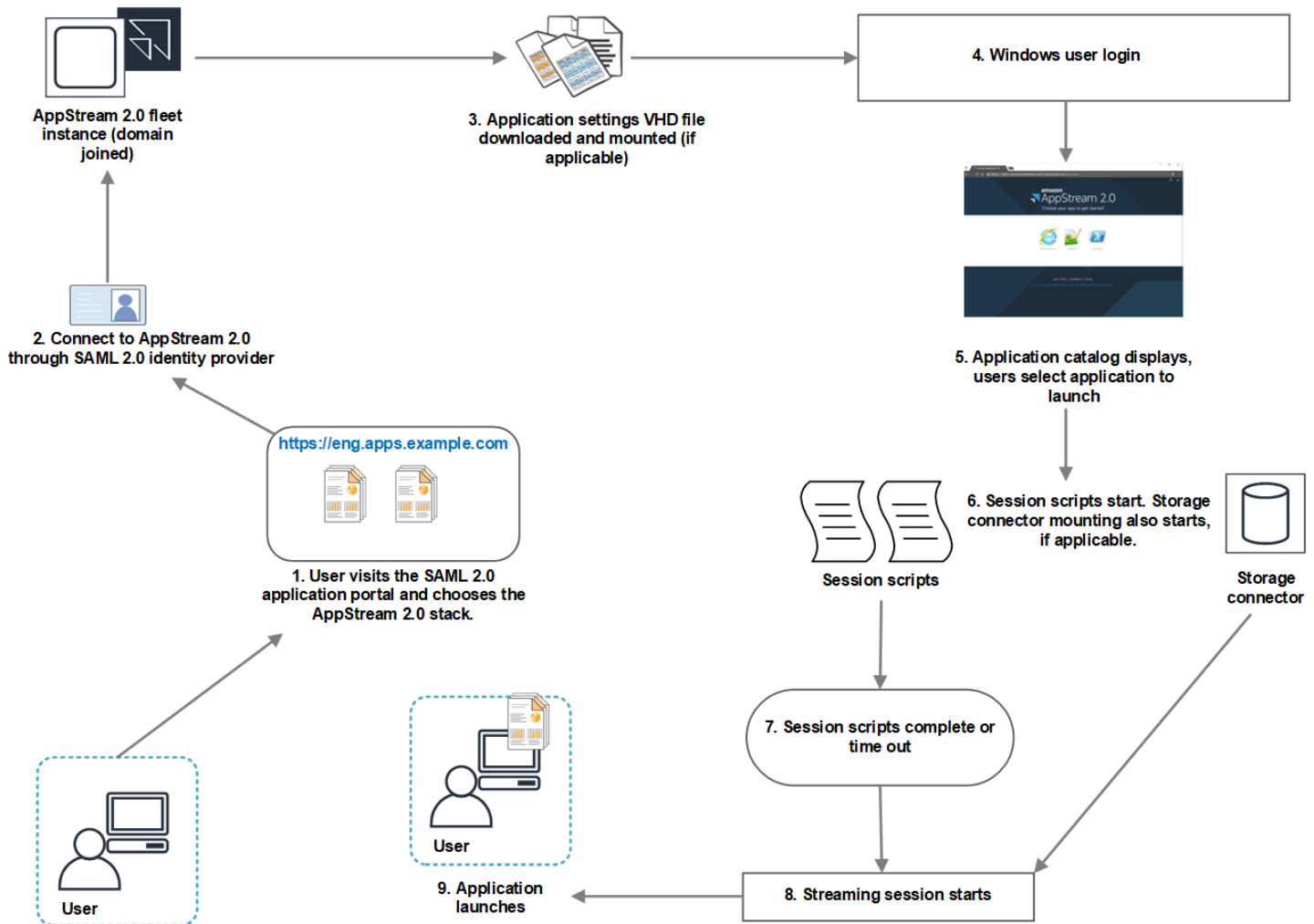
ストリーミングセッションを開始するためにストレージコネクタのマウントを完了する必要はありません。セッションスクリプトが完了したとき、まだストレージコネクタのマウントが完了していなくても、ストリーミングセッションは開始されます。

ストレージコネクタのマウント状況のモニタリングについては、[セッションスクリプトでストレージコネクタを使用する](#) を参照してください。

5. セッションスクリプトは完了するかタイムアウトします。
6. ユーザーのストリーミングセッションが開始されます。
7. ユーザーが選択したアプリケーションが起動されます。

AppStream 2.0 の動的アプリケーションフレームワークについては、[AppStream 2.0 動的アプリケーションフレームワークを使用して動的アプリケーションプロバイダーを構築する](#) を参照してください。

ストリーミングセッションの開始前にスクリプトを実行し、AppStream 2.0 の動的アプリケーションフレームワークが有効になっていると、以下のプロセスが発生します。



1. ユーザーは組織の SAML 2.0 アプリケーションポータルにアクセスし、AppStream 2.0 スタックを選択します。
2. その後、ドメインに参加している AppStream 2.0 フリートインスタンスに接続します。
3. ユーザーに対してアプリケーション設定の永続化が有効になっている場合は、ユーザーのカスタマイズ内容と Windows の設定内容を保存しているアプリケーション設定の VHD ファイルがダウンロードされてマウントされます。
4. Windows ユーザーのログオンが発生します。
5. アプリケーションカタログが AppStream 2.0 ポータルに表示され、ユーザーは起動するアプリケーションを選択します。
6. セッションスクリプトが起動されます。ユーザーに対して永続的ストレージが有効になっている場合は、ストレージコネクタのマウントも開始されます。

 Note

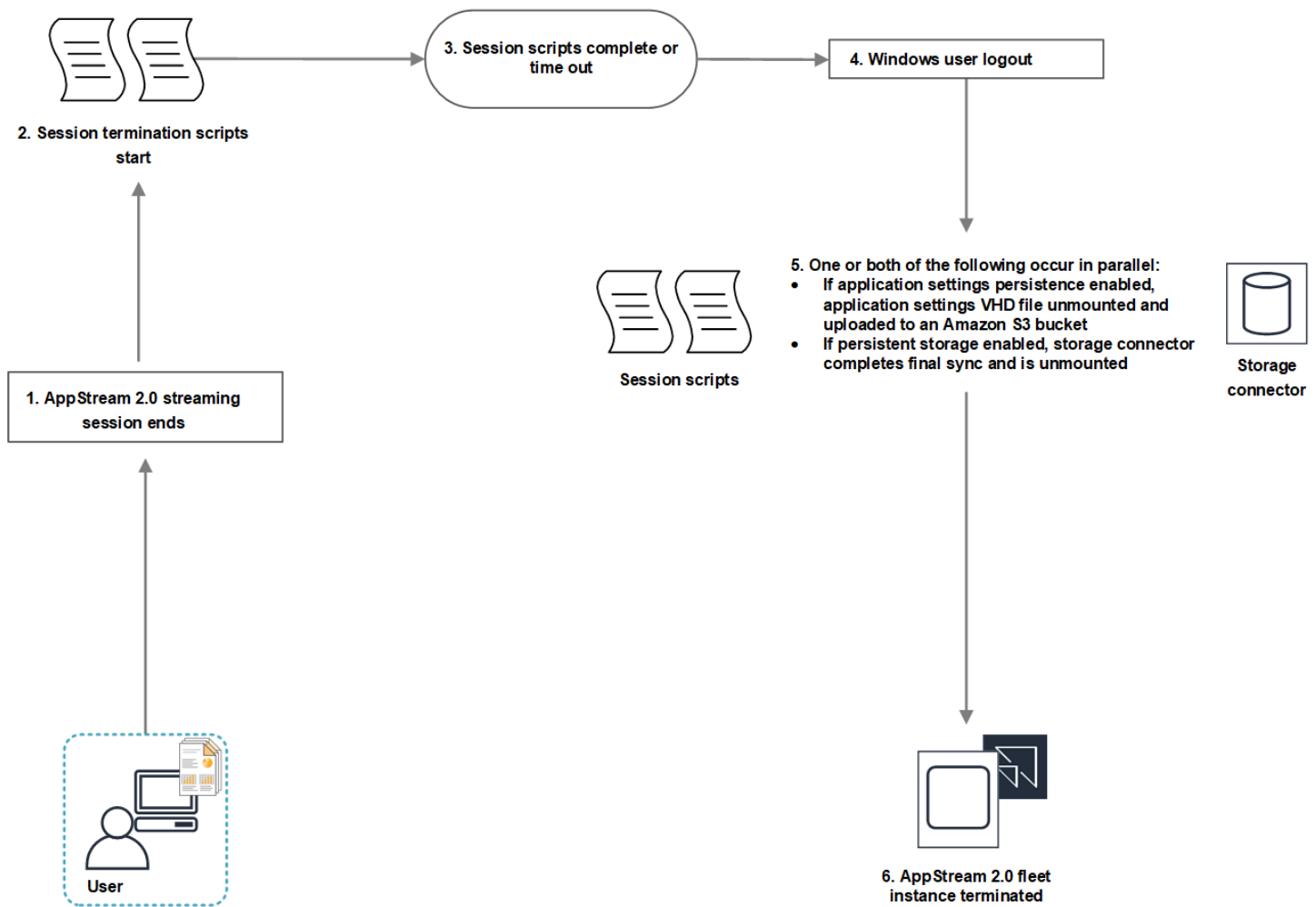
ストリーミングセッションを開始するためにストレージコネクタのマウントを完了する必要はありません。セッションスクリプトが完了したとき、まだストレージコネクタのマウントが完了していなくても、ストリーミングセッションは開始されます。

ストレージコネクタのマウント状況のモニタリングについては、[セッションスクリプトでストレージコネクタを使用する](#) を参照してください。

7. セッションスクリプトは完了するかタイムアウトします。
8. ユーザーのストリーミングセッションが開始されます。
9. ユーザーが選択したアプリケーションが起動されます。

ストリーミングセッションの終了後にスクリプトを実行する

ユーザーのストリーミングセッションの終了後にスクリプトを実行するように設定することもできます。たとえば、ユーザーが AppStream 2.0 ツールバーから [End Session (セッションの終了)] を選択したとき、またはユーザーがセッションの最大許容期間に達したときに、スクリプトを実行できます。これらのセッションスクリプトを使用して、ストリーミングインスタンスが削除される前に AppStream 2.0 環境をクリーンアップすることもできます。たとえば、スクリプトを使用してファイルロックを解除したり、ログファイルをアップロードしたりできます。ストリーミングセッションの終了後にスクリプトを実行すると、以下のプロセスが発生します。



1. ユーザーの AppStream 2.0 ストリーミングセッションが終了します。
2. セッション終了スクリプトが起動されます。
3. セッション終了スクリプトが完了またはタイムアウトします。
4. Windows ユーザーのログアウトが発生します。
5. 以下のうち該当する一方が実行されるか、両方が同時に実行されます。
 - ユーザーに対してアプリケーション設定の永続化が有効になっている場合、ユーザーのカスタマイズ内容と Windows 設定内容を保存しているアプリケーション設定の VHD ファイルがマウント解除され、アカウントの Amazon S3 バケットにアップロードされます。
 - ユーザーに対して永続的ストレージが有効になっている場合、ストレージコネクタは最後の同期を完了し、マウント解除されます。
6. フリートインスタンスは削除されます。

セッションスクリプトを作成および指定する

常時オン、オンデマンド、および Elastic フリートのセッションスクリプトを設定および指定できます。

常時オンおよびオンデマンドフリートのセッションスクリプトを設定および指定するには

1. AppStream 2.0 コンソールを <https://console.aws.amazon.com/appstream2> で開きます。
2. ナビゲーションペインで、[Images (イメージ)]、[Image Builder (イメージビルダー)] の順に選択します。
3. [実行中] 状態のイメージビルダーを選択してから、[接続] を選択します。
4. プロンプトが表示されたら、[管理者] を選択します。
5. C:\AppStream\SessionScripts に移動し、config.json 設定ファイルを開きます。

セッションスクリプトパラメータについては、[セッションスクリプト設定ファイル](#) を参照してください。

6. 変更が終了したら、config.json ファイルを保存して閉じます。
7. Image Builder デスクトップから、Image Assistant を開きます。
8. (オプション) イメージに含める他のアプリケーションを指定します。
9. Image Assistant で、必要な手順に従って、イメージの作成を完了します。

セッションスクリプトの設定の検証でエラーになる場合 (.json ファイルの形式が正しくない場合など)、[Disconnect and create image (イメージの接続解除と作成)] を選択してイメージを作成すると、通知されます。

Note

Linux ベースの Image Builder 向けのセッションスクリプト設定ファイルを見つけるには、/opt/appstream/SessionScripts/config.json に移動します。

Elastic フリートのセッションスクリプトを設定および指定するには

1. セッションスクリプトと config.json ファイルを含む zip ファイルを作成します。スクリプトファイルは、次の場所にコピーされます。config.json には、これらの場所を使用する必要があります。

- Windows の場合は C:\AppStream\SessionScripts*SessionScript* を使用します。
- Linux の場合は /opt/appstream/SessionScripts/*SessionScript* を使用します。

Note

セッションスクリプトファイルを実行するには、.zip ファイルに、含まれているフォルダではなくセッションスクリプトと config.json ファイルのみが含まれていることを確認します。詳細については、「[セッションスクリプト設定ファイル](#)」を参照してください。

2. ZIP ファイルを、アカウントの Amazon S3 バケットにアップロードします。

Note

VPC は Amazon S3 バケットに対するアクセス権を提供する必要があります。詳細については、「[AppStream 2.0 機能に対する Amazon S3 VPC エンドポイントの使用](#)」を参照してください。

S3 バケットと AppStream 2.0 フリートが同じ AWS リージョンにある必要があります。

Amazon S3 バケット内のセッションスクリプトオブジェクトで S3:GetObject アクションを実行するための IAM 許可が必要です。Amazon S3 バケットでのセッションスクリプトの保存に関する詳細については、「[S3 バケットにアプリケーションアイコン、セットアップスクリプト、セッションスクリプト、および VHD を保存する](#)」を参照してください。

3. AppStream 2.0 コンソールを <https://console.aws.amazon.com/appstream2> で開きます。
4. ナビゲーションペインの [Fleets] を選択します。
5. 更新する Elastic フリートを選択し、[View Details] (詳細を表示) を選択します。
6. [Session scripts settings] (セッションスクリプトの設定) タブで、[編集] を選択します。
7. [Session scripts object in S3] (S3 のセッションスクリプトオブジェクト) で、セットアップスクリプトオブジェクトを表す S3 URI を入力するか、[Browse S3] (S3 を参照する) を選択して S3 バケットに移動し、セットアップスクリプトオブジェクトを見つけます。
8. 変更が完了したら、[Save Changes] (変更を保存) を選択します。
9. この時点で、セッションスクリプトは起動されたすべてのフリートインスタンスで使用できます。

Note

新しい Elastic フリートを作成するときに、セッションスクリプトを設定することもできます。

セッションスクリプト設定ファイル

Windows インスタンスのセッションスクリプト設定ファイルを見つけるには、C:\AppStream\sessionScripts\config.json に移動してください。Linux インスタンスでは、/opt/appStream/sessionScripts/config.json に移動してください。ファイル形式は次のとおりです。

Note

設定ファイルは .json 形式です。このファイルに入力したテキストが有効な .json 形式であることを確認します。

```
{
  "SessionStart": {
    "executables": [
      {
        "context": "system",
        "filename": "",
        "arguments": "",
        "s3LogEnabled": true
      },
      {
        "context": "user",
        "filename": "",
        "arguments": "",
        "s3LogEnabled": true
      }
    ],
    "waitingTime": 30
  },
  "SessionTermination": {
    "executables": [
      {
        "context": "system",
```

```
    "filename": "",
    "arguments": "",
    "s3LogEnabled": true
  },
  {
    "context": "user",
    "filename": "",
    "arguments": "",
    "s3LogEnabled": true
  }
],
"waitingTime": 30
}
```

セッションスクリプト設定ファイルでは、以下のパラメータを使用できます。

SessionStart/SessionTermination

オブジェクトの名前に基づいて該当するセッションイベントで実行するセッションスクリプト。

型: 文字列

必須: いいえ

使用できる値: **SessionStart**、**SessionTermination**

WaitingTime

セッションスクリプトの最大期間 (秒単位)。

タイプ: 整数

必須: いいえ

制約: 最大期間は 60 秒です。セッションスクリプトは、この期間内に完了しない場合、停止されます。スクリプトを引き続き実行する必要がある場合は、別のプロセスとして起動してください。

Executables

実行するセッションスクリプトの詳細。

型: 文字列

必須: はい

制約: セッションイベントごとに実行できるスクリプトの最大数は 2 です (1 つはユーザーコンテキスト用、もう 1 つはシステムコンテキスト用)。

Context

セッションスクリプトを実行するコンテキスト。

型: 文字列

必須: はい

使用できる値: **user**、**system**

Filename

実行するセッションスクリプトへの完全パス。このパラメータを指定しない場合、セッションスクリプトは実行されません。

型: 文字列

必須: いいえ

制約: ファイル名と完全パスの最大長は 1,000 文字です。

使用できる値: **.bat**、**.exe**、**.sh**

Note

Windows PowerShell ファイルを使用することもできます。詳細については、「[Windows PowerShell ファイルの使用](#)」を参照してください。

##

セッションスクリプトまたは実行可能ファイルの引数。

型: 文字列

必須: いいえ

長さの制限: 最大長は 1,000 文字です。

S3LogEnabled

このパラメータの値が **True** に設定されていると、セッションスクリプトによって作成されたログを保存するための S3 バケットが Amazon Web Services アカウント内に作成されます。デフォ

ルトではこの値は **True** に設定されます。詳細については、このトピックの後半の「セッションスクリプト出力のログ記録」セクションを参照してください。

タイプ: ブール

必須: いいえ

使用できる値: **True**、**False**

Windows PowerShell ファイルの使用

Windows PowerShell ファイルを使用するには、**filename** パラメータに PowerShell ファイルへの絶対パスを指定します。

```
"filename":  
"C:\\Windows\\System32\\WindowsPowerShell\\v1.0\\powershell.exe",
```

次に、**arguments** パラメータにセッションスクリプトを指定します。

```
"arguments": "-File \"C:\\path\\to\\session\\script.ps1\"",
```

最後に、PowerShell 実行ポリシーで PowerShell ファイルの実行が許可されていることを確認します。

セッションスクリプト出力のログ記録

設定ファイルでこのオプションが有効になっていると、AppStream 2.0 によってセッションスクリプトから標準出力に書き込まれた出力が自動的に収集されます。この出力はアカウントの Amazon S3 バケットにアップロードされます。トラブルシューティングやデバッグの目的でログファイルを確認できます。

Note

ログファイルは、セッションスクリプトが値を返したときか、**WaitingTime** に設定された時間を経過したときの、どちらか早いほうでアップロードされます。

セッションスクリプトでストレージコネクタを使用する

AppStream 2.0 ストレージコネクタが有効になっている場合、セッション開始スクリプトの実行時にマウントが開始されます。スクリプトがマウントされているストレージコネクタに依存している場合は、コネクタが使用可能になるまで待つことができます。AppStream 2.0 は、以下のキーで、Windows インスタンスの Windows レジストリ内にあるストレージコネクタのマウントステータスを維持します。

HKEY_LOCAL_MACHINE\SOFTWARE\Amazon\AppStream\Storage\<provided user name>
\<Storage connector>

レジストリキーの値は以下のとおりです。

- 提供されたユーザー名 — アクセスモードで提供されたユーザー ID。アクセスモードと各モードの値は以下のとおりです。
 - ユーザープール — ユーザーの E メールアドレス。
 - ストリーミング URL — ユーザー ID。
 - SAML — NameID。ユーザー名にスラッシュが含まれる場合 (ドメインユーザーの SAMAccountName が含まれる場合など)、スラッシュは「-」文字に置き換えられます。
- ストレージコネクタ — ユーザーに対して有効になっている永続的ストレージオプションに対応するコネクタ。ストレージコネクタの値は以下のとおりです。
 - HomeFolder
 - GoogleDrive
 - OneDrive

ストレージコネクタの各レジストリキーには MountStatus DWORD 値が含まれています。次の表は、MountStatus に指定できる値の一覧です。

Note

これらのレジストリキーを表示するには、イメージに Microsoft .NET Framework バージョン 4.7.2 以降がインストールされている必要があります。

値	説明
0	ストレージコネクタはこのユーザーに対して有効になっていない
1	ストレージコネクタのマウントが進行中
2	ストレージコネクタのマウントに成功した
3	ストレージコネクタのマウントに失敗した
4	ストレージコネクタのマウントは有効ですが、まだマウントされていません

Linux インスタンスでは、file ~/.config/appstream-home-folder/appstream-home-folder-mount-status にある appstream_home_folder_mount_status の値を調べることによって、ホームフォルダのマウントステータスを確認できます。

値	説明
真	ホームフォルダが正常にマウントされています
False	ホームフォルダがまだマウントされていません

セッションスクリプトログに対して Amazon S3 バケットストレージを有効にする

セッションスクリプト設定で Amazon S3 のログ記録を有効にすると、AppStream 2.0 によってセッションスクリプトから標準出力に書き込まれた出力が収集されます。出力は、Amazon Web Services アカウント内の S3 バケットに定期的にアップロードされます。AppStream 2.0 は、すべての AWS リージョンについて、アカウントとリージョンに固有のバケットをアカウントに作成します。

これらの S3 バケットを管理するための設定タスクを実行する必要はありません。これらは AppStream 2.0 サービスによって完全に管理されています。各バケットに保存されているログファイルは、転送時には Amazon S3 の SSL エンドポイントを使用して暗号化され、保管時には Amazon

S3 管理の暗号化キーを使用して暗号化されます。バケットは、以下にあるような特定の形式で命名されます。

```
appstream-logs-region-code-account-id-without-hyphens-random-identifier
```

region-code

これは、セッションスクリプトログに対して Amazon S3 バケットストレージを有効にしてスタックを作成する AWS リージョンコードです。

account-id-without-hyphens

ご自身の Amazon Web Services アカウント ID ランダムな ID は、そのリージョン内の他のバケットとの競合が発生しないことを確実にします。バケット名の最初の部分 appstream-logs は、複数のアカウントやリージョンにまたがる場合でも変更されません。

たとえば、アカウント番号 123456789012 の米国西部 (オレゴン) リージョン (us-west-2) のイメージでセッションスクリプトを指定した場合、AppStream 2.0 では、示された名前でそのリージョンのアカウント内に Amazon S3 バケットが作成されます。適切なアクセス許可を持つ管理者のみが、このバケットを削除できます。

```
appstream-logs-us-west-2-1234567890123-abcdefg
```

セッションスクリプトを無効にしても、S3 バケットに保存されているログファイルは削除されません。ログファイルを完全に削除するには、Amazon S3 コンソールまたは API を使用して、ユーザーまたは適切なアクセス許可を持つ別の管理者が削除する必要があります。AppStream 2.0 は、バケットの誤った削除を防止するバケットポリシーを追加します。詳細については、[Amazon AppStream 2.0 向け Identity and Access Management](#) の「アプリケーション設定の永続化用の IAM ポリシーと Amazon S3 バケット」を参照してください。

セッションスクリプトを有効にすると、開始されるストリーミングセッションごとに固有のフォルダが作成されます。

アカウントの S3 バケットでログファイルが保存されているフォルダへのパスは、以下の構造になります。

```
bucket-name/stack-name/fleet-name/access-mode/user-id-SHA-256-hash/session-id/  
SessionScriptsLogs/session-event
```


bucket-name

セッションスクリプトが保存されている S3 バケットの名前。名前の形式については、このセクションで先ほど説明しました。

stack-name

セッションが発生したスタックの名前。

fleet-name

セッションスクリプトが実行されているフリートの名前。

access-mode

ユーザーの ID メソッド: AppStream 2.0 API または CLI の場合は custom、SAML の場合は federated、ユーザープールのユーザーの場合は userpool。

user-id-SHA-256-hash

ユーザー固有のフォルダ名。この名前は、ユーザー識別子から生成された小文字の SHA-256 ハッシュ 16 進数文字列を使用して作成されます。

session-id

ユーザーのストリーミングセッションの識別子。ユーザーの各ストリーミングセッションでは一意の ID が生成されます。

session-event

セッションスクリプトログを生成したイベント。イベント値は SessionStart と SessionTermination です。

以下のフォルダ構造の例は、test-stack と test-fleet から始まるストリーミングセッションに当てはまります。セッションでは testuser@mydomain.com、AWS アカウントの ID からユーザー ID の API と 123456789012、test-stack 米国西部 (オレゴン) リージョン (us-west-2) の設定グループを使用します。

```
appstream-logs-us-west-2-1234567890123-abcdefg/test-stack/test-fleet/custom/
a0bcb1da11f480d9b5b3e90f91243143eac04cfccfbdc777e740fab628a1cd13/05yd1391-4805-3da6-
f498-76f5x6746016/SessionScriptsLogs/SessionStart/
```

このフォルダ構造の例には、ユーザーコンテキストセッション開始スクリプト用の 1 つのログファイルと、必要に応じてシステムコンテキストセッション開始スクリプト用の 1 つのログファイルが含まれています。

マルチセッションフリートでセッションスクリプトを使用する

マルチセッションフリートでセッションスクリプトを使用する場合、最適なパフォーマンスとセキュリティを確保するための追加の要件と考慮事項があります。

要件

単一セッションフリートでは、特定のインスタンスに対して、[SessionStart] フックと [SessionTermination] フックは 1 回のみ実行されることが保証されます。これは、セッションとインスタンスの 1:1 マッピングがあるためです。マルチセッションフリートを使用する場合、セッションとインスタンスの N:M マッピングがあり、各セッションは独自の [SessionStart] フックと [SessionTermination] フックを実行します。つまり、[SessionStart] フックと [SessionTermination] フックは、特定のインスタンスでは、さまざまな順序で複数回実行できます。最良のエクスペリエンスを得るには、マルチセッションフリートで使用する場合、セッションスクリプトについて以下が当てはまる必要があります。

- スクリプトがべき等である。

アクションがすでに実行されている場合、スクリプトは同じインスタンス上で複数の実行を正常に処理する必要があります。

- スクリプトが独立している。

スクリプトはセッションごとに実行されるため、あるセッションが [SessionTermination] を実行している間に別のセッションが [SessionStart] を実行していたとしても、それらは互いに、あるいは他のセッションのエクスペリエンスに干渉することはありません。

- スクリプトが実行可能である。

マルチセッションインスタンスでは、複数のセッションを同時にプロビジョニングできます。つまり、セッションスクリプトを複数回同時に実行できます。スクリプトは効率的で、過剰なリソースを消費せず、インスタンスでの他のユーザーの経験やセッションの安定性に影響を与えないようにする必要があります。

これらの要件の多くは、セッションスクリプトロジックをスクリプトが実行されている特定のユーザーセッションに集中させることで満たすことができます。

セキュリティに関する考慮事項

AppStream 2.0 イメージは、ユーザーがセッションスクリプトファイルへの書き込みアクセス許可を付与するように設定しないでください。これにより、悪意のあるユーザーにとって重要な攻撃ベクトル

ルが導入され、スクリプトファイルを変更できるようになります。これらのファイルは、設定に応じて SYSTEM または別のユーザーとして実行できるようになります。

 Important

AppStream 2.0 イメージが安全に設定されていることを確認するのはユーザーの責任です。これは、複数のユーザーが同じインスタンスを使用しているマルチセッションインスタンスで特に重要です。イメージが安全に設定されていない場合、そのインスタンスのすべてのユーザーにセキュリティ上のリスクがあります。

イメージとセッションスクリプトファイルは、次の条件を満たしている必要があります。

- ユーザーには、セッションスクリプトファイルを変更するアクセス許可がありません。
- ユーザーには、セッションスクリプト config.json を変更するアクセス許可がありません。イメージのデフォルトの動作により、管理者へのアクセスが制限されます。

セッションスクリプトの実行ファイルは、実行時に変更されない安全な場所に保存する必要があります。

サービスがセッションスクリプトの実行ファイルが変更されたことを検出すると、そのインスタンスでそのフックの後続の実行が失敗し、ログファイルが Amazon S3 にアップロードされ (Amazon S3 ログ記録が有効になっている場合)、次のメッセージが表示されます。

インスタンスのプロビジョニング後に実行可能ファイルが変更されたため、セッションスクリプトは実行されませんでした。セキュリティ上の理由から実行がスキップされました。

ユースケースで実行時にセッションスクリプト実行可能ファイルを変更する必要がある場合 (例えば、実行時に自動更新プロセスによって変更される EXE ファイルを指す場合)、上記のチェックは失敗します。この場合、スクリプトを使用して、実行を変更した実行可能ファイルにリダイレクトします。サービスがセキュリティチェックを実行する場合、スクリプトは実行時に変更しないでください。

セッションスクリプトファイルが過度に大きい (100 MB を超える) 場合、インスタンスとセッションのプロビジョニングが遅れる可能性があり、セキュリティチェックにはさらに時間がかかります (インスタンスタイプと使用可能なリソースによって異なります)。ユースケースで大規模なセッションスクリプトが必要な場合は、実行をリダイレクトするために小さなスクリプトを使用することを検討してください。これにより、インスタンスとセッションのプロビジョニングエクスペリエンスが向上します。

サービスはセッションスクリプト config.json で定義された実行可能ファイルのみをチェックしており、これはフォールバック/ベストエフォートメカニズムにすぎないことに注意してください。セッションスクリプト実行可能ファイル内のすべてのコードパスが安全であり、エンドユーザーが変更できないことを確認するのはユーザーの責任です。

アプリケーションマネージャー

Elastic フリートをを使用するときは、App Block とアプリケーションを作成することができます。App Block は、ユーザーが使用することになるアプリケーションの起動に必要なアプリケーションファイルとバイナリが含まれる、アカウント内の Amazon S3 バケットに格納された仮想ハードディスク (VHD) を表します。アプリケーションには、VHD のマウント後にアプリケーションを起動するために必要な詳細情報が含まれています。以下のセクションでは、これらのリソースを作成して管理する方法を説明します。

内容

- [App Block](#)
- [App Block Builder](#)
- [アプリケーション](#)
- [S3 バケットにアプリケーションアイコン、セットアップスクリプト、セッションスクリプト、および VHD を保存する](#)
- [アプリケーションを Elastic フリートに関連付ける](#)
- [その他のリソース](#)

App Block

App Block は、ユーザーが使用することになるアプリケーションの起動に必要なアプリケーションファイルとバイナリが含まれる、アカウント内の Amazon S3 バケットに格納された仮想ハードディスク (VHD) を表します。App Block には、VHD ファイルの処理方法をオペレーティングシステムに伝えるセットアップスクリプトも含まれています。

App Block では、次の 2 種類のパッケージをサポートしています。

- カスタム - アプリケーションパッケージ (VHD) を手動で作成するには、このオプションを選択します。詳細については、「[the section called “カスタム App Blocks”](#)」を参照してください。
- AppStream 2.0 - App Block Builder を使用してアプリケーションパッケージを作成するには、この推奨オプションを選択します。詳細については、「[the section called “AppStream 2.0 App Block”](#)」を参照してください。

内容

- [カスタム App Blocks](#)
- [AppStream 2.0 App Block](#)
- [サポートされていないアプリケーション](#)

カスタム App Blocks

Elastic フリートストリーミングインスタンスは、アカウントの Amazon S3 バケットに保存されている仮想ハードディスク (VHD) ファイルにインストールされたアプリケーションを利用します。カスタムパッケージを使用した App Block では、独自の VHD ファイルを自由に作成でき、その App Block をアカウント内の Amazon S3 バケットにアップロードできます。

内容

- [VHD を作成する](#)
- [Amazon AppStream 2.0 で VHD のセットアップスクリプトを作成する](#)
- [カスタム App Block を作成する](#)
- [App Block、VHD、およびセットアップスクリプトを更新する](#)

VHD を作成する

VHD は、オペレーティングシステムにマウントされると、ハードディスクのように扱われる単一のファイルです。VHD は、ドライブ文字としてマウントする、フォルダパスにマウントする、またはそれらの両方にマウントできます。VHD がマウントされると、他のハードディスクと同様に扱うことが可能になり、これには VHD へのユーザーに必要となるアプリケーションのインストールやファイルのコピーが含まれます。

App Block を作成するには、VHD を作成し、それにアプリケーションをインストールしてから、デタッチする必要があります。デタッチされたら、別の PC、EC2 インスタンス、または AppStream 2.0 Image Builder で VHD をテストして、アプリケーションが期待どおりに動作することを確認できます。それが完了したら、アカウント内の Amazon S3 バケットにアップロードして、App Block を作成します。

Note

このページでは、VHD を使用してアプリケーションを配信する方法について説明します。ただし、AppStream 2.0 ストリーミングインスタンスは Amazon S3 からオブジェクトをダウンロードします。Amazon S3 に保存するオブジェクトは、zip ファイル、アプリケーション

インストーラー、またはアプリケーションの実行可能ファイル自体でもかまいません。セットアップスクリプトを使用して、ユーザーがアプリケーションを起動する前に、ストリーミングインスタンスで正しく設定することができます。

AppStream 2.0 ストリーミングインスタンスは、セットアップスクリプトが実行される前に VHD のダウンロードが完了するまで最大 120 秒間待機します。VHD がこの期間内にダウンロードを完了しない場合、ダウンロードは停止し、セットアップスクリプトは実行されません。

VHD の最大サイズは 1.5 ギガバイトにすることをお勧めします。圧縮することで、VHD のサイズを小さくできる可能性があります。このファイルをマウントし、アプリケーションを起動する前に、ファイルを Amazon S3 から完全にダウンロードしておく必要があるため、セットアップスクリプトを使用して、このファイルを解凍しておく必要があります。VHD のサイズが大きいほど、アプリケーションの起動とストリーミングセッションの開始にかかる時間が長くなります。

Microsoft Windows 用の VHD を作成する

1. Windows PC または Windows Amazon Elastic Compute Cloud (Amazon EC2) インスタンスから、管理者権限を使用してコマンドプロンプトを開きます。
2. 以下のコマンドを入力して、Microsoft diskpart ユーティリティを起動します。

```
diskpart
```

3. 以下のコマンドを入力して、フォーマット化と初期化が行われていない VHD ファイルを作成します。ここでの *<maximum file size>* は、VHD ファイルのサイズ (MB 単位) です。

```
create vdisk file=C:\path\to\new\file.vhdx maximum=<maximum file size> type=expandable
```

4. 以下のコマンドを入力して、新しく作成された VHD を選択します。

```
select vdisk file=C:\path\to\new\file.vhdx
```

5. 以下のコマンドを入力して、新しく作成された VHD をアタッチします。

```
attach vdisk
```

6. 以下のコマンドを入力して、新しく作成された VHD を初期化します。

```
convert mbr
```

7. 以下のコマンドを入力して、VHD 全体にまたがるプライマリパーティションを作成します。

```
create partition primary
```

- 以下のコマンドを入力して、新しく作成されたパーティションをフォーマットします。

```
format fs=ntfs quick
```

- 新しく作成された VHD は、未使用のドライブ文字、ルートボリューム上のフォルダパス、またはそれらの両方にマウントできます。

ドライブ文字をマウントするには、assign letter=<unused drive letter> を入力します。

フォルダをマウントするには、assign mount=**C:\path\to\empty\folder\to\mount** を入力します。

 Note

フォルダパスにマウントするには、フォルダがすでに存在し、空である必要があります。

- これで、ステップ 9 で選択したドライブ文字またはフォルダマウントパスを使用して、アプリケーションを VHD にインストールできるようになりました。

VHD へのアプリケーションのインストールが完了したら、Amazon S3 バケットに安全にアップロードする前に、VHD をデタッチする必要があります。

Microsoft Windows 用の VHD をデタッチする

- 以下のコマンドを入力して、Microsoft diskpart ユーティリティを起動します。

```
diskpart
```

- 以下のコマンドを入力して、VHD を選択します。

```
select vdisk file=C:\path\to\new\file.vhdx
```

- 以下のコマンドを入力して、VHD をデタッチします。

```
detach vdisk
```

- これで VHD がデタッチされたので、別の Windows PC、Amazon EC2 インスタンス、または AppStream 2.0 Image Builder で VHD をテストすることができます。

Linux 用の VHD を作成する

1. Amazon Linux 2 EC2 インスタンス、Amazon Linux 2 AppStream 2.0 Image Builder、または Amazon Linux 2 WorkSpaces から、ターミナルセッションを開きます。
2. フォーマット化および初期化が行われていない VHD ファイルを作成します。

```
dd if=/dev/zero of=<name of file> bs=<size of VHD> count=1
```

3. 以下のコマンドを入力して、作成された VHD にファイルシステムを追加します。

```
sudo mkfs -t ext4 <name of file>
```

Note

ファイルがブロックスペシャルデバイスではないというメッセージが表示される場合があります。そのまま続行することを選択できます。

4. 以下のコマンドを入力して、マウントポイント用に使用する空のフォルダを作成します。

```
sudo mkdir /path/to/mount/point
```

5. 以下のコマンドを実行して、新しく作成された VHD をファイルシステムパスにマウントします。

```
sudo mount -t auto -o loop <name of file> /path/to/mount/point
```

6. これで、ステップ 4 で選択したフォルダマウントパスを使用して、アプリケーションを VHD にインストールできるようになりました。

Note

VHD で作成されたファイルとフォルダに対するデフォルトの許可は、管理者以外のユーザーによるアプリケーションの起動やファイルの読み取りを防ぐことができます。許可を検証し、必要に応じて変更します。

VHD へのアプリケーションのインストールが完了したら、Amazon S3 バケットに安全にアップロードする前に、VHD をデタッチする必要があります。

Linux 用の VHD をデタッチする

1. ターミナルセッションを開き、以下のコマンドを入力します。

```
sudo umount /path/to/mount/point
```

2. これで VHD がデタッチされたので、別の Amazon Linux 2 Amazon EC2 インスタンス、Amazon Linux 2 AppStream 2.0 Image Builder、または Amazon Linux 2 WorkSpaces で VHD をテストすることができます。

Amazon AppStream 2.0 で VHD のセットアップスクリプトを作成する

AppStream 2.0 は、ユーザー提供のセットアップスクリプトを使用して、アプリケーションを起動する前に VHD をマウントします。セットアップスクリプトは、アプリケーションを動作させるために必要なその他のタスクを完了するために使用することもできます。例えば、レジストリキーの設定、DLL の登録、前提条件の管理、またはセットアップスクリプトからのユーザープロファイルの変更を行うことができます。AppStream 2.0 は、VHD のマウントに使用できるスクリプトサンプルを提供します。これらのスクリプトは、お使いの VHD とアプリケーションのニーズに合わせて変更する必要があります。

Note

AppStream 2.0 パッケージの App Block には、セットアップスクリプトは必要ありません。ただし、オプションでセットアップ後のスクリプトを使用することで、アプリケーションのインストールをカスタマイズできます。

サンプルスクリプトをダウンロードするには、次のリンクを使用します。

- [Amazon Linux 2 bash スクリプト](#)
- [Microsoft Windows PowerShell スクリプト](#)

Note

AppStream 2.0 および Microsoft Windows オペレーティングシステムは、ドライブ文字 A ~ E を予約します。これらのドライブ文字に VHD またはネットワーク共有をマウントしないでください。

AppStream 2.0 は、セットアップスクリプトと VHD をフリーストリーミングインスタンス上のディレクトリにダウンロードしてから、セットアップスクリプトを実行します。セットアップスクリプトは、完全な管理者権限を用いてオペレーティングシステム上で実行されます。セットアップスクリプト

リプトは、Microsoft Windows の SYSTEM コンテキスト内で実行、および Amazon Linux 2 の root ユーザーとして実行されます。

VHD とセットアップスクリプトのファイルシステムの場所は以下のとおりです。

- Amazon Linux 2:

```
/opt/appstream/AppBlocks/appblock-name/  
appblock-name
```

VHD とセットアップスクリプトに対応する App Block の名前です。

- Microsoft Windows:

```
C:\AppStream\AppBlocks\appblock-name\  
appblock-name
```

VHD とセットアップスクリプトに対応する App Block の名前です。

AppStream 2.0 は、ファイル名をオブジェクト上にあるものとして維持します。例えば、App Block の名前が MyApps で、apps.vhd という名前の VHD と mount-apps.ps1 という名前のセットアップスクリプトがある場合、Windows ストリーミングインスタンス上のフルパスは以下のようになります。

- VHD

```
C:\AppStream\AppBlocks\MyApps\apps.vhd
```

- セットアップスクリプト

```
C:\AppStream\AppBlocks\MyApps\mount-apps.ps1
```

AppStream 2.0 は、セットアップスクリプトがフリートストリーミングインスタンスで実行されるときにセットアップスクリプトからの標準エラーと標準出力をキャプチャして、出力をアカウント内の Amazon S3 バケットにアップロードします。これらのログを使用して、セットアップスクリプトで発生する可能性のある問題を特定し、解決することができます。バケットは、以下にあるような特定の形式で命名されます。

```
appstream-logs-region-code-account-id-without-hyphens-random-identifier
```

region-code

これは、Elastic フリートが作成される AWS リージョンコードです。

account-id-without-hyphens

AWS アカウント識別子。ランダムな ID は、そのリージョン内のその他バケットとの競合が発生しないことを確実にします。バケット名の最初の部分である appstream-logs は、アカウントおよびリージョンに応じて変化するものではありません。

例えば、アカウント番号 123456789012 で米国西部 (オレゴン) リージョン (us-west-2) に Elastic フリートを作成する場合、AppStream 2.0 は、このリージョン内のアカウントに、以下に記載されている名前の Amazon S3 バケットを作成します。適切なアクセス許可を持つ管理者のみが、このバケットを削除できます。

```
appstream-logs-us-west-2-1234567890123-abcdefg
```

アカウントの S3 バケットでログファイルが保存されているフォルダへのパスは、以下の構造になります。

```
bucket-name/fleet-name/instance-id/appblock-name/
```

bucket-name

セッションスクリプトログが保存されている Amazon S3 バケットの名前です。名前の形式については、このセクションで先ほど説明しました。

Instance-id

セットアップスクリプトが実行されたストリーミングインスタンスの一意的識別子です。

appblock-name

セットアップスクリプトに対応する App Block の名前です。

以下のフォルダ構造の例は、test-fleet で始まるストリーミングセッションに該当します。セッションは 123456789012 の AWS アカウント ID からのもので、appblock 名は米国西部 (オレゴン) リージョン (us-west-2) の testappblock です。

```
appstream-logs-us-west-2-1234567890123-abcdefg/test-fleet/  
i-084427ab4a1cff7f5/testappblock/
```

このフォルダ構造の例には、標準出力用のログファイルが 1 つ、および標準エラー用のログファイルが 1 つ含まれています。

トピック

- [Amazon AppStream 2.0 での App Block 設定スクリプトの実行](#)

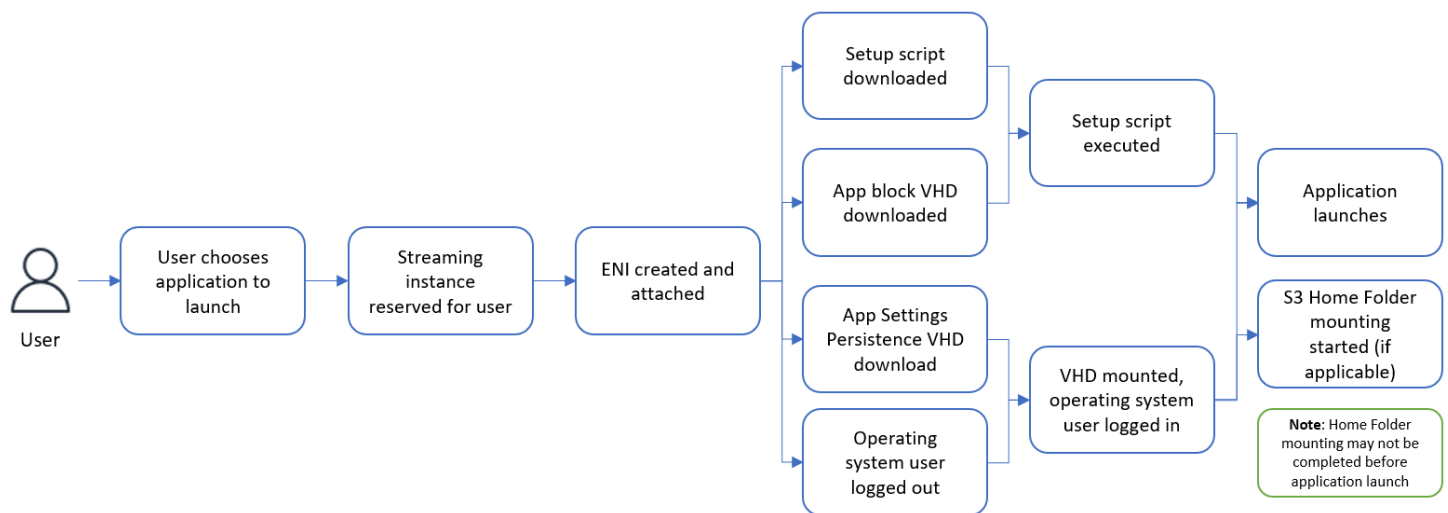
Amazon AppStream 2.0 での App Block 設定スクリプトの実行

以下の図は、セットアップスクリプトがプロセスのどこで実行されるかを示しています。実行順序は、Elastic フリートに関連付けられたスタックでアプリケーション設定の永続化が有効になっているかどうかに応じて異なります。

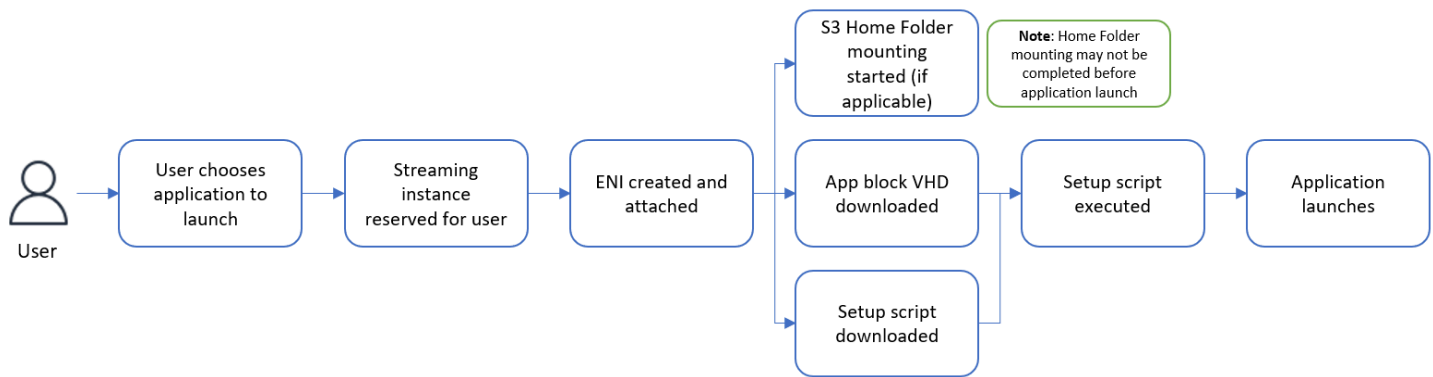
Note

AppStream 2.0 は、Amazon S3 バケットからの VHD とセットアップスクリプトのダウンロードに VPC の詳細情報を使用します。VPC は Amazon S3 バケットに対するアクセス権を提供する必要があります。詳細については、「[AppStream 2.0 機能に対する Amazon S3 VPC エンドポイントの使用](#)」を参照してください。

アプリケーション設定の永続化が有効:



アプリケーション設定の永続化が無効:



カスタム App Block を作成する

VHD とセットアップスクリプトを作成して、それらを AWS アカウント内の S3 バケットにアップロードしたら、AppStream 2.0 コンソールを使用して App Block リソースを作成できます。Amazon S3 バケットでの VHD とセットアップスクリプトの保存に関する詳細については、[the section called “S3 バケットにアプリケーションアイコン、セットアップスクリプト、セッションスクリプト、および VHD を保存する”](#) を参照してください。

Note

App Block リソースを作成するには、Amazon S3 バケット内の VHD オブジェクトとセットアップスクリプトオブジェクトで S3:GetObject アクションを実行するための IAM 許可が必要です。

App Block リソースを作成する

1. AppStream 2.0 コンソール (<https://console.aws.amazon.com/appstream2>) を開きます。
2. 左側のナビゲーションメニューから、[Applications] (アプリケーション)、[App Block]、[Create app block] (App Block を作成する) の順にクリックします。
3. App Block パッケージの場合は、[カスタム] を選択します。
4. [App block details] (App Blockの詳細情報) に、App Block 用の一意の名前識別子を入力します。オプションで、以下を指定することもできます。
 - [Display name] (表示名) – App Block のフレンドリー名です。
 - [Description] (説明) – App Block の説明です。

5. [Script settings] (スクリプト設定) にある [Virtual hard disk object in S3] (S3 内の仮想ハードディスクオブジェクト) では、VHD オブジェクトを表す S3 URI を入力するか、[Browse S3] (S3 を参照する) をクリックして S3 バケットに移動し、VHD オブジェクトを見つけます。
6. [Script settings] (スクリプト設定) にある [Setup script object in S3] (S3 内のセットアップスクリプトオブジェクト) では、セットアップスクリプトオブジェクトを表す S3 URI を入力するか、[Browse S3] (S3 を参照する) をクリックして S3 バケットに移動し、セットアップスクリプトオブジェクトを見つけます。
7. [Script settings] (スクリプト設定) にある [Setup script executable] (セットアップスクリプト実行可能ファイル) には、セットアップスクリプトに必要な実行可能ファイルを入力します。

 Note

セットアップスクリプトが直接実行できる場合は、セットアップスクリプトのファイル名を入力します。セットアップスクリプトが、実行において別の実行可能ファイル (Microsoft PowerShell など) に依存する場合は、その実行可能ファイルへのパスを入力します。

Microsoft Windows 上の Microsoft PowerShell へのパス

`C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe`

8. オプションで、[Script settings] (スクリプト設定) にある [Setup script executable arguments] (セットアップスクリプト実行可能ファイルの引数) に、セットアップスクリプトを実行するためにセットアップスクリプト実行可能ファイルに提供する必要がある引数を入力します。

 Note

Microsoft PowerShell スクリプトを使用している場合は、「-File」パラメータには、セットアップスクリプトの名前を実行可能な引数として指定する必要があります。また、実行ポリシーがスクリプトの実行を許可することを確認してください。詳細については、[about_Execution_Policies](#) と [PowerShell とは](#)を参照してください。

9. [Script settings] (スクリプト設定) にある [Execution duration in seconds] (実行時間 (秒単位)) には、セットアップスクリプトのタイムアウト時間を入力します。

 Note

秒単位の実行時間は、AppStream 2.0 が続行する前にセットアップスクリプトの実行を待機する時間です。セットアップスクリプトがこの時間内に完了しない場合は、ユー

ザーにエラーが表示され、アプリケーションの起動が試行されます。実行時間経過後、セットアップスクリプトを終了します。

10. (オプション) [Tags] (タグ) で、App Block リソース用のタグを作成します。
11. 入力した情報を確認して、[Create] (作成) をクリックします。
12. App Block が正常に作成された場合は、コンソールの上部に成功メッセージが表示されます。エラーが発生した場合は、説明的なエラーメッセージが表示されるので、App Block を再度作成する必要があります。

App Block、VHD、およびセットアップスクリプトを更新する

App Block リソースはイミュータブルで、作成後に変更することはできません。VHD またはセットアップスクリプトに対して下位互換性のある更新を行う必要がある場合は、ファイルの新しいバージョンを Amazon S3 バケットにアップロードして、現在のバージョンを上書きすることをお勧めします。新しい Elastic フリートのストリーミングセッションは、オブジェクトの最新バージョンをダウンロードして、それらを使用します。

VHD またはセットアップスクリプトに対して下位互換性のない更新を行う必要がある場合は、それらを新しいオブジェクトとして Amazon S3 バケットにアップロードして、新しい App Block とアプリケーションリソースを作成することをお勧めします。そうすることで、ユーザーへのデプロイを変更時間枠またはその他停止の一環として管理することができます。

AppStream 2.0 App Block

Elastic フリートストリーミングインスタンスは、アカウントの Amazon S3 バケットに保存されている仮想ハードディスク (VHD) ファイルにインストールされたアプリケーションを利用します。カスタムパッケージの App Block については、独自の VHD ファイルを作成して、それをアカウント内の Amazon S3 バケットにアップロードするという柔軟性を備えています。また、AppStream 2.0 パッケージを使用する App Block の場合は、App Block Builder を使用して、アプリケーションのパッケージ化の処理を行い、VHD ファイルを作成して Amazon S3 バケットにアップロードできます。

AppStream 2.0 をパッケージ化した App Block を使用すると、VHD ファイルを手動で作成する必要がなくなるだけでなく、セットアップスクリプトも不要になります。これにより、Elastic フリートとのアプリケーションの互換性が広がり、App Block の作成に必要な手動による管理手順も削減されます。AppStream 2.0 は、AppStream 2.0 パッケージを使用して、セットアップスクリプトなしで App Block のセットアップを自動的に処理します。ただし、オプションでセットアップ後のスクリプトを使用することで、必要に応じてインストール内容をカスタマイズできます。

内容

- [概要](#)
- [サポートされていないアプリケーション](#)
- [AppStream 2.0 App Block を作成する](#)
- [App Block をアクティブにする](#)
- [既存のアプリケーションパッケージを使用して App Block を作成する](#)
- [App Block をテストする](#)
- [Amazon AppStream 2.0 で App Block を関連付ける](#)
- [Amazon AppStream 2.0 で App Block の関連付けを解除する](#)

概要

AppStream 2.0 パッケージを使用して App Block を作成するには、App Block Builder でストリーミングセッションを開始する必要があります。セッションが開始されたら、アプリケーションインストーラをダウンロードして、記録オプションを有効にできます。これ以降 AppStream 2.0 では、アプリケーションリダイレクトテクノロジーを使用して、App Block Builder で行われたファイルシステムとレジストリの変更を記録します。

アプリケーションリダイレクトは、Windows フィルタードライバーストックを使用し、ファイルシステムとレジストリの変更をインターセプトし、リダイレクトします。このリダイレクトは、インストール中のアプリケーションに対してシームレスに行われます。アプリケーションでは、C: ドライブの元のファイルの場所とのやり取りを継続します。例えば、「TestApplication」のインストーラをアプリケーションリダイレクトが設定されたマシンで実行すると、デフォルトで C:\Program Files\TestApplication にインストールされます。ただし、バックグラウンドでは、すべてのファイルとフォルダがマウントされた仮想ハードディスク (VHD) にリダイレクトされ、元のファイルの場所から実際のファイルの場所へのリンクが作成されます。マシン上では、TestApplication はまだ C:\Program Files\TestApplication にインストールされているように見えます。

インストールの変更がすべて記録されると、VHD ファイルがアカウントの Amazon S3 バケットにアップロードされます。

ユーザーが Elastic フリートを使用してセッションをリクエストすると、AppStream 2.0 は VHD ファイルをダウンロードして、アプリケーションをセットアップし、インストール後のセットアップスクリプト (オプション) を実行して、アプリケーションストリーミングを開始します。

Note

アプリケーションリダイレクトテクノロジーでは、%APPDATA% ディレクトリと %LOCALAPPDATA% ディレクトリの下に作成された新しいディレクトリを除き、%USERPROFILE% の下のファイルシステムの変更は記録されません。アプリケーションリダイレクトテクノロジーでは、現在のユーザーである HKEY_CURRENT_USER (HKCU) のレジストリの変更は記録されません。

サポートされていないアプリケーション

次のシナリオでは、アプリケーションのインストールまたは実行時にエラーが発生する可能性があります。

- インストール中にロケーションベースのチェックが行われるアプリケーション: アプリケーションのインストールプロセスで、インストールされているファイルの実際の場所が検証されると、エラーが発生する可能性があります。AppStream 2.0 では、ファイルを App Block の VHD にリダイレクトするため、実際のファイルへのリンクのみが元の場所に保持されます。

アプリケーションがこれらのカテゴリのいずれに該当するか不明な場合は、AppStream 2.0 パッケージを使用して App Block を作成できます。このプロセスでは、アプリケーションを App Block Builder インスタンスにインストールします。アプリケーションを App Block Builder インスタンスにインストールできなかった場合は、次のアクションを実行できます。

- ログの確認 App Block Builder インスタンスのエラーログファイルは C:\AppStream\AppBlocks\errorLog にあります。このログには、レジストリキーやファイル操作処理を含む、すべてのインストールエラーが記録されます。errorLog に次のいずれかのログが表示された場合は、現在、AppStream 2.0 App Block Builder でアプリケーションパッケージがサポートされていないことを示しています。
 - 「シンボリックリンクを作成できません」
 - 「サービスではファイル名の変更をサポートしていません」

errorLog ファイルがない場合、またはこのファイルが空の場合は、アプリケーションのインストールログを確認してエラーの原因を特定してください。

- 問題の報告 App Block Builder のアプリケーションビルダーアシスタントにある [問題の報告] ボタンを選択します。このオプションを選択すると、App Block Builder インスタンスからすべての AppStream 2.0 ログを収集し、サポートのために AppStream 2.0 チームに送信されます。

- カスタムパッケージを使用した App Block 作成: App Block Builder を使用してアプリケーションをパッケージ化できない場合は、カスタムパッケージの方法を使用して App Block の作成を試行できます。詳細については、「[the section called “カスタム App Blocks”](#)」を参照してください。
- さらにサポートが必要な場合は、AWS サポートにお問い合わせください。詳細については、[AWS サポート センター](#)を参照してください。

アプリケーションに AppStream 2.0 パッケージを使用する場合は、これらに制限がある可能性を考慮し、それに応じて計画を立てることが重要です。

AppStream 2.0 App Block を作成する

次のステップに従って、AppStream 2.0 パッケージタイプの App Block を作成します。

ステップ 1: App Block を設定する

App Block を設定するには

1. AppStream 2.0 コンソール (<https://console.aws.amazon.com/appstream2>) を開きます。
2. 左側のナビゲーションメニューから、[アプリケーションマネージャー]、[App Block]、[App Block の作成] の順にクリックします。
3. App Block パッケージには、[AppStream 2.0] を選択します。
4. [App block details] (App Blockの詳細情報) に、App Block 用の一意の名前識別子を入力します。オプションで、以下を指定することもできます。
 - [名前] – App Block 用に一意の名前を入力します。
 - [表示名] (オプション) – App Block のフレンドリー名です。
 - [説明] (オプション) – App Block の説明です。
5. (オプション) AppStream 2.0 パッケージを使用した App Block には、セットアップスクリプトは必要ありません。オプションで、インストール後のステップを、次の [詳細オプション] で指定できます。
 - [S3 のセットアップ後のスクリプトオブジェクト] では、セットアップ後のスクリプトオブジェクトを表す Amazon S3 URI を入力するか、[S3 を参照する] を選択して S3 バケットに移動し、セットアップスクリプトオブジェクトを検索します。
 - [セットアップ後のスクリプト実行可能ファイル] では、セットアップ後のスクリプトに必要な実行可能ファイルを入力します。

Note

セットアップスクリプトが直接実行できる場合は、セットアップスクリプトのファイル名を入力します。セットアップスクリプトが、実行において別の実行可能ファイル (Microsoft PowerShell など) に依存する場合は、その実行可能ファイルへのパスを入力します。

Microsoft Windows 上の Microsoft PowerShell へのパス

C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe

オプションで、[Script settings] (スクリプト設定) にある [Setup script executable arguments] (セットアップスクリプト実行可能ファイルの引数) に、セットアップスクリプトを実行するためにセットアップスクリプト実行可能ファイルに提供する必要がある引数を入力します。

Note

Microsoft PowerShell スクリプトを使用している場合は、「-File」パラメータには、セットアップスクリプトの名前を実行可能な引数として指定する必要があります。また、実行ポリシーがスクリプトの実行を許可することを確認してください。詳細については、[about_Execution_Policies](#) と [PowerShell とは](#) を参照してください。

[Script settings] (スクリプト設定) にある [Execution duration in seconds] (実行時間 (秒単位)) には、セットアップスクリプトのタイムアウト時間を入力します。

Note

秒単位の実行時間は、AppStream 2.0 が続行する前にセットアップスクリプトの実行を待機する時間です。セットアップスクリプトがこの時間内に完了しない場合は、ユーザーにエラーが表示され、アプリケーションの起動が試行されます。実行時間経過後、セットアップスクリプトを終了します。

6. [設定をインポート] で、[App Block アプリケーションファイルの新規作成] を選択します。[設定をインポート] の [S3 の場所] に、バケットを表す Amazon S3 URI を入力するか、[S3 を参照する] を選択して Amazon S3 バケットに移動し、適切なバケットを選択します。Amazon S3 バケットのリストはグローバルで、すべてのリージョンですべてのバケットがリスト表示されま

す。App Block を作成するリージョンのバケットを必ず選択してください。バケットのアクセス権限の設定に関する詳細については、「[the section called “S3 バケットにアプリケーションアイコン、セットアップスクリプト、セッションスクリプト、および VHD を保存する”](#)」を参照してください。

7. App Block Builder を選択します。他の App Block に関連付けられていない App Block Builder のみを使用できます。リストが空の場合は、新しい App Block Builder を作成するか、既存の App Block Builder の関連付けを解除して使用してください。App Block Builder は、アプリケーションパッケージの作成に使用できる再利用可能なリソースです。

 Note

ここで App Block Builder を選択しない場合でも、[非アクティブ] 状態で App Block を作成し、後で App Block をアクティブにできます。詳細については、「[the section called “App Block をアクティブにする”](#)」を参照してください。

8. (オプション) [タグ] で、App Block リソース用のタグを作成します。
9. [Next (次へ)] を選択します。
10. 入力した情報を確認し、次のオプションのいずれかを選択します。
 - ステップ 7 で App Block Builder を選択しなかった場合は、[App Block の作成] を選択します。
 - ステップ 7 で App Block Builder を選択した場合は、[App Block Builder の起動] を選択します。次に、ステップ 2 に進み、App Block Builder ストリーミングセッションを使用して、アプリケーションパッケージを作成します。

この時点で、App Block リソースが作成されますが [非アクティブ] であり、Elastic フリートには使用できません。

ステップ 2: アプリケーションパッケージを作成する

App Block Builder ストリーミングインスタンスを使用してアプリケーションをパッケージ化し、App Block をアクティブにします。App Block Builder を使用して作成された App Block には AppStream 2.0 パッケージが含まれており、アプリケーションパッケージが AWS アカウント の Amazon S3 バケットにアップロードされます。

アプリケーションパッケージを作成するには

1. ストリーミングセッションがオンになると、アプリケーションビルダーアシスタントが自動的に起動します。起動しない場合は、デスクトップアイコンを使用して手動で起動します。
2. 最初の画面には、アプリケーションのパッケージ化プロセスの説明が表示されます。
3. 次のいずれかのオプションを使用して、アプリケーションインストーラを App Block Builder ストリーミングセッションに移行します。
 - ウェブからアプリケーションインストーラをダウンロードします。
 - ストリーミングセッションファイルインターフェイスを使用してください。
 - マシンロールを使用して、別の AWS サービスからアプリケーションインストーラをダウンロードします。
4. 必要なアプリケーションインストーラをすべて用意したら、インスタンスで実行されている他のアプリケーションをすべて停止し、[記録の開始] を選択します。App Block Builder によってシステム変更の記録が開始されると、画面に [記録中] と表示されます。
5. アプリケーションを 1 つずつインストールし始めます。
6. アプリケーションのインストールが完了したら、[記録の停止] を選択すると、システムは変更の記録を停止します。アプリケーションの追加や、既にインストールされているアプリケーションの削除など、アプリケーションパッケージにさらに変更を加える場合は、[記録の開始] を選択し、システムが [記録中] モードになっていることを確認します。

Note

アプリケーションのインストールが失敗した場合は、[問題の報告] を選択してインスタンスから AppStream 2.0 関連のログを収集し、その問題を AppStream 2.0 チームに報告してください。完了したら、App Block Builder のストリーミングセッションを終了します。新しい App Block Builder インスタンスを使用して、App Block の作成プロセスの再開を試行できます。問題が解決しない場合は、カスタムパッケージを使用して App Block を作成してみてください。

7. すべてのアプリケーションのインストールが完了したら、[記録の停止] を選択します。[スタート] メニューを使用するか、File Explorer を使用してアプリケーションを参照することで、アプリケーションをテストできます。
8. [次へ] を選択して、App Block の詳細を確認します。

Note

Elastic フリートのアプリケーションパッケージ (VHD) ファイルの推奨サイズは 1.5 GB 未満です。VHD のファイルサイズが 1.5 GB を超える場合は、1 つの App Block にパッケージされるアプリケーションの数を減らします。

アプリケーションをアンインストールしても、アプリケーションパッケージ (VHD) のファイルサイズは縮小しません。新しい App Block ストリーミングセッションを使用してアプリケーションのパッケージ化プロセスを再開し、インストールするアプリケーションの数を減らしてみてください。

9. [App Block の作成を終了して接続解除する] を選択し、アプリケーションパッケージを作成して Amazon S3 バケットにアップロードします。正常にアップロードされると、ストリーミングセッションは自動的に切断され、App Block は [アクティブ] 状態になります。

Note

アプリケーションのインストールが失敗した場合は、[問題の報告] を選択してインスタンスから AppStream 2.0 関連のログを収集し、その問題を AppStream 2.0 チームに報告してください。完了したら、App Block Builder のストリーミングセッションを終了します。新しい App Block Builder インスタンスを使用して、App Block の作成プロセスの再開を試行できます。問題が解決しない場合は、カスタムパッケージを使用して App Block を作成してみてください。

App Block をアクティブにする

AppStream 2.0 パッケージを使用した App Block が作成されても、そのアプリケーションパッケージ (VHD) がアタッチされていない場合、App Block は非アクティブ状態になり、アプリケーションを Elastic フリートに関連付けることはできません。App Block をアクティブにするには、アプリケーションパッケージ (VHD) を App Block に関連付ける必要があります。

アプリケーションパッケージを作成するには

1. AppStream 2.0 コンソール (<https://console.aws.amazon.com/appstream2>) を開きます。
2. 左側のナビゲーションメニューから、[アプリケーションマネージャー]、[App Block] の順にクリックします。

3. アクティブにする [非アクティブ] の App Block を選択し、[アクション] メニューから [アクティブ] を選択します。
4. App Block Builder を選択し、[App Block Builder の起動] を選択します。
 - リストが空の場合は、App Block Builder がないか、すべての App Block Builder が他の App Block に関連付けられています。新しい App Block Builder を作成するか、既存の App Block Builder の関連付けを解除してテストしてください。
 - App Block Builder が既に App Block に関連付けられている場合は、その App Block をそのまま使用して App Block をアクティブにできます。
 - 選択した App Block Builder が App Block Builder に関連付けられていない場合は、選択した App Block Builder に関連付けられ、ストリーミングセッションが開始されます。App Block Builder は、セッション終了後もこの App Block に関連付けられたままになります。
5. App Block Builder のストリーミングセッションが開始されたら、[「the section called “ステップ 2: アプリケーションパッケージを作成する”」](#) のステップに従ってアプリケーションパッケージ (VHD) を作成し、App Block をアクティブにします。

既存のアプリケーションパッケージを使用して App Block を作成する

AppStream 2.0 パッケージで、既存のアプリケーションパッケージ (VHD) を使用して AppStream 2.0 App Block を作成できます。作成するには、ソースの Amazon S3 バケットから別のターゲットの Amazon S3 バケットにアプリケーションパッケージ (VHD) ファイルをコピーします。ターゲットバケットが別のリージョンにある場合があります。

既存のアプリケーションパッケージを使用して App Block を作成するには

1. AppStream 2.0 コンソール (<https://console.aws.amazon.com/appstream2>) を開きます。
2. 左側のナビゲーションメニューから、[アプリケーションマネージャー]、[App Block]、[App Block の作成] の順にクリックします。
3. App Block パッケージには、[AppStream 2.0] を選択します。
4. [App block details] (App Blockの詳細情報) に、App Block 用の一意の名前識別子を入力します。オプションで、以下を指定することもできます。
 - [名前] – App Block 用に一意の名前を入力します。
 - [表示名] (オプション) – App Block のフレンドリー名です。
 - [説明] (オプション) – App Block の説明です。

5. (オプション) AppStream 2.0 パッケージを使用した App Block には、セットアップスクリプトは必要ありません。オプションで、インストール後のステップを、次の [詳細オプション] で指定できます。
- [S3 のセットアップ後のスクリプトオブジェクト] では、セットアップ後のスクリプトオブジェクトを表す Amazon S3 URI を入力するか、[S3 を参照する] を選択して S3 バケットに移動し、セットアップスクリプトオブジェクトを検索します。
 - [セットアップ後のスクリプト実行可能ファイル] では、セットアップ後のスクリプトに必要な実行可能ファイルを入力します。

 Note

セットアップ後のスクリプトが直接実行できる場合は、セットアップ後のスクリプトのファイル名を入力します。セットアップ後のスクリプトが、実行時に別の実行可能ファイル (Microsoft PowerShell など) に依存する場合は、その実行可能ファイルへのパスを入力します。

Microsoft Windows 上の Microsoft PowerShell へのパス

C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe

オプションで、[セットアップ後のスクリプトの実行可能ファイルの引数] に、セットアップスクリプトを実行するためにセットアップスクリプト実行可能ファイルに必要な引数を入力します。

 Note

Microsoft PowerShell スクリプトを使用している場合は、「-File」パラメータには、セットアップ後のスクリプトの名前を実行可能な引数として指定する必要があります。また、実行ポリシーがスクリプトの実行を許可することを確認してください。詳細については、[about_Execution_Policies](#) と [PowerShell とは](#) を参照してください。

[Script settings] (スクリプト設定) にある [Execution duration in seconds] (実行時間 (秒単位)) には、セットアップスクリプトのタイムアウト時間を入力します。

Note

秒単位の実行時間は、AppStream 2.0 が続行する前にセットアップ後のスクリプトの実行を待機する時間です。セットアップ後のスクリプトがこの時間内に完了しない場合は、ユーザーにエラーが表示され、アプリケーションの起動が試行されます。実行時間経過後、セットアップスクリプトを終了します。

6. [設定をインポート] で、[既存の App Block アプリケーションファイルを使用] を選択します。[S3 の場所] には、アプリケーションパッケージ (VHD) を表す Amazon S3 バケット内のオブジェクトの Amazon S3 URI を入力できます。または、[S3 を参照する] を選択して Amazon S3 バケットに移動し、Amazon S3 バケット内のオブジェクトを選択することもできます。Amazon S3 バケットのリストはグローバルで、すべてのリージョンですべてのバケットがリスト表示されます。App Block を作成するリージョンのバケットを必ず選択してください。
7. [Next (次へ)] を選択します。
8. 入力した情報を確認して、[App Block の作成] をクリックします。

この時点で、App Block リソースが作成され、[アクティブ] 状態になります。

App Block をテストする

App Block Builder を使用して App Block をテストし、アプリケーションの機能を検証できます。このオプションでは、Elastic フリートを起動する必要はありません。また、インスタンスタイプやサイズが異なる複数の App Block Builder を作成し、さまざまなコンピューティングオプションでアプリケーションのパフォーマンスをテストできます。

Note

App Block のテストオプションは、AppStream 2.0 パッケージの App Block でのみサポートされます。

App Block をテストするには

1. AppStream 2.0 コンソール (<https://console.aws.amazon.com/appstream2>) を開きます。
2. 左側のナビゲーションメニューから、[アプリケーションマネージャー]、[App Block] の順にクリックします。

3. テストする App Block を選択し、[アクション] メニューから [テスト] を選択します。
4. App Block Builder を選択し、[App Block の起動とテスト] を選択します。
 - リストが空の場合は、App Block Builder がないか、すべての App Block Builder が他の App Block に関連付けられています。新しい App Block Builder を作成するか、既存の App Block Builder の関連付けを解除してテストしてください。
 - App Block Builder が既に App Block に関連付けられている場合は、その App Block をそのまま使用して App Block をアクティブにできます。
 - 選択した App Block Builder が App Block Builder に関連付けられていない場合は、選択した App Block Builder に関連付けられ、ストリーミングセッションが開始されます。App Block Builder は、セッション終了後もこの App Block に関連付けられたままになります。
5. App Block Builder は、デスクトップストリーミングモードにおいて、別のブラウザウィンドウで起動します。このサービスによって Amazon S3 バケットから App Block をダウンロードし、App Block Builder インスタンスにインストールします。
6. これで、アプリケーションをストリーミングし、テストできます。アプリケーションを開くには、File Explorer を使用してアプリケーションを参照するか、[スタート] メニューを使用します。
7. テストが終了したら、ストリーミングセッションを終了します。

Amazon AppStream 2.0 で App Block を関連付ける

AppStream 2.0 パッケージを使用して App Block を作成、テスト、アクティブ化するには、App Block Builder に関連付ける必要があります。1 つの App Block Builder は 1 つの App Block にのみ関連付けることができ、その逆も同様です。

Note

App Block の関連付けと関連付けの解除は、AppStream 2.0 パッケージを使用した App Block でのみサポートされます。

Amazon AppStream 2.0 で App Block を App Block Builder に関連付ける

1. AppStream 2.0 コンソール (<https://console.aws.amazon.com/appstream2>) を開きます。
2. 左側のナビゲーションメニューから、[アプリケーションマネージャー]、[App Block] の順にクリックします。

3. App Block を選択し、[アクション] メニューから [関連付け] を選択します。
4. App Block Builder を選択し、[App Block Builder の関連付け] を選択します。

リストが空の場合は、App Block Builder がないか、すべての App Block Builder が他の App Block に関連付けられています。新しい App Block Builder を作成するか、既存の App Block Builder の関連付けを解除してから関連付けしてください。

Amazon AppStream 2.0 で App Block の関連付けを解除する

すべての App Block Builder が他の App Block に関連付けられていて、別の App Block をテスト、作成、アクティブ化する場合は、新しい App Block Builder を作成するか、既存の App Block Builder を App Block から切り離すことで、新しい App Block で使用できます。

Note

App Block の関連付けと関連付けの解除は、AppStream 2.0 パッケージを使用した App Block でのみサポートされます。

関連付けの解除は、App Block Builder が [停止済み] 状態の場合のみ許可されます。

App Block と App Block Builder の関連付けを解除する

1. AppStream 2.0 コンソール (<https://console.aws.amazon.com/appstream2>) を開きます。
2. 左側のナビゲーションメニューから、[アプリケーションマネージャー]、[App Block] の順にクリックします。
3. App Block を選択し、[アクション] メニューから [関連付けの解除] を選択します。
4. 既に関連付けられた App Block Builder を選択し、[App Block Builder の関連付けの解除] を選択します。

サポートされていないアプリケーション

次のシナリオでは、アプリケーションのインストールまたは実行時にエラーが発生する可能性があります。

- インストール後に再起動が必要なアプリケーション: アプリケーションがインストール後に再起動が必要な追加の変更や設定を行う必要がある場合、エラーになる可能性があります。現在、App

Block Builder は再起動をサポートしていないため、アプリケーションがインストール後に必要なステップを完了できない可能性があります。

- ユーザー固有の詳細に依存するアプリケーション: App Block Builder に現在ログインしているユーザーのみにインストールすることを目的としたアプリケーション、またはインストール中のセキュリティ識別子 (SID) など、App Block Builder にログインしているユーザーの詳細に依存するアプリケーションは、Elastic フリートで正しく機能しない場合があります。これは、Elastic フリート環境内でのログインユーザーの変更によるものです。また、アプリケーションのリダイレクトでは、%USERPROFILE% の下のすべてのディレクトリが記録されるわけではありません。ただし、環境に基づいてアプリケーション構成を動的に変更するように、セットアップ後のスクリプトを設定することもできます。
- マシン固有の詳細に依存するアプリケーション: ネットワークアダプタの GUID など、インストール時に App Block Builder のマシン固有の詳細に依存するアプリケーションでは、Elastic フリートで問題が発生する可能性があります。これは、ネットワークアダプタの GUID を含むマシンの詳細が、Elastic フリート環境内で変更される可能性があるためです。これに対処するため、これらのマシン固有の詳細の設定を処理するようにセットアップ後のスクリプトを構成できます。

アプリケーションがこれらのカテゴリのいずれに該当するか不明な場合は、AppStream 2.0 パッケージを使用して App Block を作成できます。このプロセスでは、アプリケーションを App Block Builder インスタンスにインストールします。アプリケーションを App Block Builder インスタンスにインストールできなかった場合は、次のアクションを実行できます。

- ログの確認 App Block Builder インスタンスのエラーログファイルは C:\AppStream\AppBlocks\errorLog にあります。このログには、レジストリキーやファイル操作処理を含む、すべてのインストールエラーが記録されます。errorLog に次のいずれかのログが表示された場合は、現在、AppStream 2.0 App Block Builder でアプリケーションパッケージがサポートされていないことを示しています。
 - 「シンボリックリンクを作成できません」
 - 「サービスではファイル名の変更をサポートしていません」

errorLog ファイルがない場合、またはこのファイルが空の場合は、アプリケーションのインストールログを確認してエラーの原因を特定してください。

- 問題の報告 App Block Builder のアプリケーションビルダーアシスタントにある [問題の報告] ボタンを選択します。このオプションを選択すると、App Block Builder インスタンスからすべての AppStream 2.0 ログを収集し、AppStream 2.0 チームに送信されます。

- カスタムパッケージを使用した App Block 作成: App Block Builder を使用してアプリケーションをパッケージ化できない場合は、カスタムパッケージの方法を使用して App Block の作成を試行できます。詳細については、「[the section called “カスタム App Blocks”](#)」を参照してください。
- さらにサポートが必要な場合は、AWS サポートにお問い合わせください。詳細については、[AWS サポート センター](#)を参照してください。

アプリケーションに AppStream 2.0 パッケージを使用する場合は、これらに制限がある可能性を考慮し、それに応じて計画を立てることが重要です。

App Block Builder

App Block Builder は、アプリケーション (または App Block) のパッケージ化に使用できる再利用可能なリソースです。また、アプリケーションを Elastic フリートに関連付ける前に、アプリケーションパッケージをテストするためにも使用できます。1 つの App Block Builder を使用して、複数の App Block を 1 つずつ作成し、テストできます。App Block を作成またはテストするために App Block Builder のストリーミングセッションを作成するたびに、新しいインスタンスが作成され、使用されます。App Block Builder インスタンスが終了すると、インスタンスの状態は永続化されません。

AppStream 2.0 Elastic フリートでは、Amazon EC2 インスタンスを使用してアプリケーションをストリーミングします。アプリケーションパッケージを用意して、それをフリートに関連付ける必要があります。独自のカスタムアプリケーションパッケージを作成するには、App Block Builder インスタンスに接続し、アプリケーションをインストールして、ストリーミング用に設定します。App Block Builder によってアプリケーションのパッケージを作成し、それを AWS アカウントの Amazon S3 バケットにアップロードします。

App Block Builder を作成する場合は、以下のものを選択します。

- インスタンスタイプ — AppStream 2.0 は、さまざまな CPU とメモリ構成で、異なるインスタンスサイズを提供します。インスタンスタイプは必要なインスタンスファミリーと合致する必要があります。
- 使用する VPC、サブネット、セキュリティグループ — アプリケーションに必要なネットワークリソースへのアクセスが、サブネットとセキュリティグループで許可されていることを確認してください。アプリケーションに必要な一般的なネットワークリソースとしては、ライセンスサーバー、データベースサーバー、ファイルサーバー、アプリケーションサーバーなどがあります。App Block Builder によってアプリケーションパッケージを作成し、それを AWS アカウントの Amazon S3 バケットにアップロードします。フリート用に選択する VPC は、Amazon S3

バケットに対する十分なネットワークアクセス権を提供する必要があります。詳細については、[「the section called “S3 バケットにアプリケーションアイコン、セットアップスクリプト、セッションスクリプト、および VHD を保存する”」](#)を参照してください。

内容

- [App Block Builder を作成する](#)
- [Amazon AppStream 2.0 で App Block Builder に接続する](#)
- [App Block Builder のアクション](#)

App Block Builder を作成する

App Block Builder インスタンスを使用して、AppStream 2.0 Elastic フリート用のアプリケーションパッケージを作成できます。

App Block Builder を作成するには

1. AppStream 2.0 コンソール (<https://console.aws.amazon.com/appstream2>) を開きます。
2. 左側のナビゲーションペインで [アプリケーションマネージャー] を選択し、[App Block Builder] タブと [App Block Builder の作成] を選択します。
3. [ステップ 1: App Block Builder の設定]では、次の詳細を入力して App Block Builder を設定します。
 - [名前]: App Block Builder の固有の名前を入力します。
 - [表示名] (オプション): App Block Builder に表示する名前を入力します (最大 100 文字)。
 - [オペレーティングシステム]: アプリケーションのオペレーティングシステムを選択します。これは、エンドユーザーがアプリケーションをストリーミングするために使用する Elastic フリート用に選択されるオペレーティングシステムと一致している必要があります。
 - IAM ロール (オプション) : アカウントから AppStream 2.0 App Block Builder に IAM ロールを適用すると、AWS 認証情報を手動で管理することなく、App Block Builder インスタンスから AWS API リクエストを行うことができます。App Block Builder に IAM ロールを適用するには、次のいずれかを行います。
 - Amazon Web Services アカウントで既存の IAM ロールを使用するには、[IAM ロール] の一覧表示から使用するロールを選択します。ロールは、イメージビルダーからアクセス可能である必要があります。詳細については、[「AppStream 2.0 ストリーミングインスタンスで使用するための既存の IAM ロールの設定」](#)を参照してください。

- 新しい IAM ロールを作成するには、[Create new IAM role (新しい IAM ロールの作成)] を選択し、[AppStream 2.0 ストリーミングインスタンスで使用する IAM ロールを作成する方法](#) のステップを実行します。
 - [インスタンスタイプ]: App Block Builder のインスタンスタイプを選択します。インストールする予定のアプリケーションのパフォーマンス要件と一致するタイプを選択します。
 - [タグ (省略可能)]: [タグの追加] を選択し、タグのキーと値を入力します。さらにタグを追加するには、この手順を繰り返します。詳細については、「[Amazon AppStream 2.0 リソースにタグを付ける](#)」を参照してください。
4. [Next (次へ)] を選択します。
5. [ステップ 2: ネットワークの設定] では、以下の操作を行います。
- パブリックサブネットを持つ VPC に App Block Builder のインターネットアクセスを追加するには、[デフォルトのインターネットアクセス] を選択します。NAT ゲートウェイを使用してインターネットアクセスを提供する場合は、[Default Internet Access (デフォルトのインターネットアクセス)] を未選択のままにします。詳細については、「[インターネットアクセス](#)」を参照してください。
 - [VPC] と [サブネット 1] では、VPC と少なくとも 2 つのサブネットを選択します。耐障害性を高めるために、異なるアベイラビリティーゾーンで 3 つのサブネットを選択することをお勧めします。詳細については、「[プライベートサブネットの VPC および NAT ゲートウェイを設定する](#)」を参照してください。
- 独自の VPC とサブネットがない場合は、[デフォルトの VPC](#) を使用するか、独自の VPC を作成できます。独自の VPC とサブネットを作成するには、[Create a new VPC (新しい VPC を作成する)] リンクおよび [Create new subnet (新しいサブネットを作成する)] リンクを選択して作成します。これらのリンクを選択すると、Amazon VPC コンソールが開きます。VPC とサブネットを作成したら、AppStream 2.0 コンソールに戻り、[Create a new VPC (新しい VPC を作成する)] リンクと [Create new subnet (新しいサブネットを作成する)] リンクの左側にある更新アイコンを選択してリストに表示します。詳細については、「[AppStream 2.0 用の VPC の設定](#)」を参照してください。
- [Security group(s) (セキュリティグループ)] で、この Image Builder に関連付ける最大 5 つのセキュリティグループを選択します。独自のセキュリティグループを持たず、デフォルトのセキュリティグループを使用しない場合は、[Create new security group (新しいセキュリティグループを作成する)] リンクを選択して、セキュリティグループを作成します。Amazon VPC コンソールでサブネットを作成したら、AppStream 2.0 コンソールに戻り、[Create new security group (新しいセキュリティグループを作成する)] リンクの左側にある更新アイコンを

選択してリストに表示します。詳細については、「[Amazon AppStream 2.0 のセキュリティグループ](#)」を参照してください。

- [VPC エンドポイント] (オプション): 仮想プライベートクラウド (VPC) に、インターフェイス VPC エンドポイント (インターフェイスエンドポイント) を作成できます。インターフェイスエンドポイントを作成するには、[VPC エンドポイントの作成] を選択します。このリンクを選択すると、VPC コンソールが開きます。エンドポイントの作成を完了するには、「[the section called “チュートリアル: インターフェイス VPC エンドポイントからの作成とストリーミング”](#)」のステップ 3~6 に従います。インターフェイスエンドポイントを作成したら、VPC 内でストリーミングトラフィックを維持するために使用できます。
6. [Next (次へ)] を選択します。
 7. [レビュー] を選択し、App Block Builder の詳細を確認します。任意のセクションの設定を変更するには、[Edit] を選択し、必要な変更を加えます。
 8. 設定内容を確認したら、[App Block Builder の作成] を選択します。

 Note

Image Builder を作成するのに十分な制限 (クォータ) がないというエラーメッセージが表示された場合は、<https://console.aws.amazon.com/servicequotas/> の Service Quotas コンソールから制限引き上げリクエストを送信してください。詳細については「Service Quotas ユーザーガイド」の「[クォータの引き上げのリクエスト](#)」を参照してください。

Amazon AppStream 2.0 で App Block Builder に接続する

App Block Builder に接続するには、次のいずれかの操作を行います。

- AppStream 2.0 コンソールの使用 (ブラウザ接続のみ)
- ストリーミング URL の作成 (ブラウザまたは AppStream 2.0 クライアント接続用)

 Note

App Block Builder は Active Directory ドメイン参加をサポートしていません。

内容

- [Amazon AppStream 2.0 コンソール \(ブラウザ接続\)](#)

- [ストリーミング URL \(Amazon AppStream 2.0 クライアントまたはブラウザ接続\)](#)

Amazon AppStream 2.0 コンソール (ブラウザ接続)

AppStream 2.0 コンソールを使用してブラウザ経由で App Block Builder に接続するには、次の手順を実行します。

1. AppStream 2.0 コンソール (<https://console.aws.amazon.com/appstream2>) を開きます。
2. 左側のナビゲーションペインで [アプリケーションマネージャー] を選択し、次に [App Block Builder] を選択します。
3. App Block Builder 一覧で、接続する App Block Builder を選択します。App Block Builder のステータスが [実行中] であることを確認し、[接続] を選択します。

この手順を実行するには、<https://stream.<aws-region>.amazonappstream.com/> からのポップアップを許可するようにブラウザを設定する必要があるかもしれません。

4. App Block Builder のストリーミングを開始します。

ストリーミング URL (Amazon AppStream 2.0 クライアントまたはブラウザ接続)

ストリーミング URL を作成して、ブラウザまたは AppStream 2.0 クライアントを介して App Block Builder に接続できます。フリーインスタンスへのユーザーアクセスを有効にするように作成するストリーミング URL とは異なり、デフォルトでは、Image Builder にアクセスするために作成するストリーミング URL は、1 時間後に失効します。別の有効期限を設定するには、[CreateAppBlockBuilderStreamingURL](#) API アクションを使用してストリーミング URL を生成する必要があります。

Note

App Block Builder に接続するための URL のストリーミングは、macOS クライアントではサポートされていません。

ストリーミング URL は、次のいずれかの方法で作成できます。

- AppStream 2.0 コンソール
- [CreateAppBlockBuilderStreamingURL](#) API アクション
- [create-app-block-builder-streaming-url](#) AWS CLI コマンド

AppStream 2.0 コンソールを使用してストリーミング URL を作成し、App Block Builder に接続するには、以下の手順を実行します。

AppStream 2.0 コンソールを使用してストリーミング URL を作成し、App Block Builder に接続するには

1. AppStream 2.0 コンソール (<https://console.aws.amazon.com/appstream2>) を開きます。
2. ナビゲーションペインで [アプリケーションマネージャー] を選択し、次に [App Block Builder] を選択します。
3. App Block Builder 一覧で、接続する App Block Builder を選択します。App Block Builder のステータスが [実行中] であることを確認します。
4. [Actions (アクション)]、[Create streaming URL (ストリーミング URL の作成)] の順に選択します。
5. 次のいずれかを行います：
 - ストリーミング URL を保存した後に App Block Builder に接続するには、[リンクをコピー] を選択して URL をコピーし、アクセス可能な場所に保存します。
 - AppStream 2.0 クライアント経由で App Block Builder に接続するには、[クライアントで起動] を選択します。このオプションを選択すると、AppStream 2.0 クライアントのサインインページに事前にストリーミング URL が入力されます。
 - ブラウザを使用して App Block Builder に接続するには、[ブラウザで起動] を選択します。このオプションを選択すると、ブラウザが開き、ストリーミング URL が事前に入力されたアドレスバーが表示されます。
6. ストリーミング URL を作成して App Block Builder に接続したら、App Block Builder のストリーミングを開始します。

App Block Builder のアクション

App Block Builder インスタンスの現在の状態 (ステータス) に応じて、App Block Builder で以下のアクションを実行できます。

削除

App Block Builder を完全に削除します。

インスタンスは [Stopped] 状態である必要があります。

接続

実行中の App Block Builder に接続します。このアクションは、App Block Builder を使用してデスクトップストリーミングセッションを開始して、アプリケーションのインストール、追加、App Block の作成を行います。

インスタンスは [Running] 状態である必要があります。

開始

停止した App Block Builder を起動します。実行中のインスタンスは、お客様のアカウントに請求されます。

インスタンスは [停止] 状態で、App Block に関連付けられている必要があります。

停止

実行中の App Block Builder を停止します。停止されたインスタンスは、お客様のアカウントに請求されません。

インスタンスは [Running] 状態である必要があります。

更新

名前を除き、App Block Builder のプロパティをすべて更新します。

インスタンスは [Stopped] 状態である必要があります。

次の中間の状態のインスタンスでは、これらのアクションは実行されません。

- [保留中]
- 停止中
- スタート
- 削除

アプリケーション

アプリケーションには、VHD のマウント後にアプリケーションを起動するために必要な詳細情報が含まれています。アプリケーションには、アプリケーションカタログでユーザーに表示される名前とアイコンも含まれています。アプリケーションは、そのアプリケーション用のファイルとバイナリが含まれる App Block リソースに関連付けられています。

アプリケーションアイコンを Amazon S3 バケットにアップロードし、アプリケーションを起動するために必要なファイルとフォルダが含まれる App Block を作成したら、AppStream 2.0 コンソールを使用してアプリケーションリソースを作成できます。Amazon S3 バケットへのアプリケーションアイコンのアップロードに関する詳細については、[the section called “S3 バケットにアプリケーションアイコン、セットアップスクリプト、セッションスクリプト、および VHD を保存する”](#) を参照してください。

 Note

アプリケーションリソースを作成するには、S3 バケット内のアプリケーションアイコンオブジェクトで S3:GetObject アクションを実行するための IAM 許可が必要です。

アプリケーションリソースを作成する

1. AppStream 2.0 コンソール (<https://console.aws.amazon.com/appstream2>) を開きます。
2. 左側のナビゲーションメニューから、[Applications] (アプリケーション)、[Create application] (アプリケーションを作成する) の順にクリックします。
3. [Application details] (アプリケーションの詳細情報) にある [Name] (名前) に、アプリケーション用の一意の名前識別子を入力します。
4. (オプション) [Application details] (アプリケーションの詳細情報) にある [Display name] (表示名) に、アプリケーションカタログでユーザーに表示されるフレンドリー名を入力します。
5. (オプション) [Application details] (アプリケーションの詳細情報) にある [Description] (説明) に、アプリケーションの説明を入力します。
6. [Application details] (アプリケーションの詳細情報) にある [Application icon object in S3] (S3 内のアプリケーションアイコンオブジェクト) では、VHD オブジェクトを表す S3 URI を入力するか、[Browse S3] (S3 を参照する) をクリックして S3 バケットに移動し、アプリケーションアイコンオブジェクトを見つけます。
7. [Application settings] (アプリケーション設定) にある [Application executable launch path] (アプリケーション実行可能ファイルの起動パス) には、アプリケーション実行可能ファイルへのストリーミングインスタンス上のパスを入力します。
8. (オプション) [Application settings] (アプリケーション設定) セクションにある [Application working directory] (アプリケーションの作業ディレクトリ) に、アプリケーションの作業ディレクトリ用に使用するストリーミングインスタンス上のディレクトリを入力します。

9. (オプション) [Application settings] (アプリケーション設定) セクションにある [Application launch parameters] (アプリケーション起動パラメータ) に、アプリケーションの起動時にアプリケーション実行可能ファイルに提供するパラメータを入力します。
10. [Application settings] (アプリケーション設定) セクションにある [Supported operating systems (OS)] (サポートされるオペレーティングシステム (OS)) で、このアプリケーションを起動できるオペレーティングシステムを選択します。
11. [Application settings] (アプリケーション設定) セクションにある [Supported instance families] (サポートされるインスタンスファミリー) で、このアプリケーションを起動できるインスタンスファミリーを選択します。
12. [Application settings] (アプリケーション設定) セクションにある [App Block] で、このアプリケーションに必要なファイルとフォルダが含まれている App Block を選択します。
13. (オプション) [Tags] (タグ) セクションで、App Block リソース用のタグを作成します。
14. 入力した情報を確認してから、[Create] (作成) をクリックします。
15. アプリケーションが正常に作成された場合は、コンソールの上部に成功メッセージが表示されます。エラーが発生した場合は、説明的なエラーメッセージが表示されるので、アプリケーションを再度作成する必要があります。

S3 バケットにアプリケーションアイコン、セットアップスクリプト、セッションスクリプト、および VHD を保存する

アプリケーションと App Block に使用するアプリケーションアイコン、セットアップスクリプト、セッションスクリプト、および VHD は、AWS アカウント内の Amazon Simple Storage Service (Amazon S3) バケットに保存する必要があります。AppStream 2.0 Elastic フリートは、ユーザーがストリーミングセッションを開始するときに、S3 バケットからアプリケーションアイコン、セットアップスクリプト、および VHD をダウンロードします。S3 バケットは、AppStream 2.0 Elastic フリートを作成する予定の AWS リージョンに存在する必要があります。

Elastic フリートで使用する予定のアプリケーションアイコン、セットアップスクリプト、セッションスクリプト、および VHD の保存専用を使用される新しい S3 バケットを作成することが推奨されます。また、S3 バケットでバージョンングを有効にすることもお勧めします。そうすることで、必要に応じて以前のオブジェクトバージョンに戻すことが可能になります。新しい S3 バケットを作成する方法の詳細については、[バケットの作成](#)を参照してください。オブジェクトのバージョンングを管理する方法の詳細については、[S3 バケットでのバージョンングの使用](#)を参照してください。

Note

AppStream 2.0 は、選択された S3 バケットへのアクセスに VPC を使用します。フリート用
に選択する VPC は、S3 バケットに対する十分なネットワークアクセス権を提供する必要が
あります。

S3 バケットコンテンツが、管理するキー (カスタマーマネージドキー) を使用して暗号化さ
れていないことを確認します。

現在、カスタマー提供の暗号化キー (SSE-C) でサーバー側の暗号化を使用するように設定さ
れた S3 バケットは、Elastic フリートではサポートされていません。S3 オブジェクトの保存
時に暗号化が必要な場合は、Amazon S3 で管理された暗号化キー (SSE-S3) によるサーバー
側の暗号化が Elastic フリートで機能するオプションです。

トピック

- [Amazon S3 バケット許可](#)

Amazon S3 バケット許可

選択する Amazon S3 バケットには、Amazon S3 バケットにアクセスして、そのバケットからオブ
ジェクトをダウンロードするために十分なアクセス権を AppStream 2.0 サービスプリンシパルに
提供するバケットポリシーが必要です。次のバケットポリシーは、変更を加えてから、アプリケー
ションアイコン、セットアップスクリプト、および VHD 用に使用する予定の Amazon S3 バケット
に適用する必要があります。Amazon S3 バケットにポリシーを適用する方法の詳細については、
「[Amazon S3 コンソールを使用したバケットポリシーの追加](#)」を参照してください。

Amazon S3 バケットのアクセスコントロールリスト (ACL) が無効になっていることを確認してくだ
さい。詳細については、「[すべての新しいバケットの ACL を無効にし、オブジェクト所有権を執行
する](#)」を参照してください。

このセクションでは、バケットポリシーの一般的なユースケース例を紹介します。これらのサンプル
ポリシーでは、####をリソース値として使用します。これらのポリシーをテストするには、#####
#####をお客様の情報 (バケット名など) と置き換えます。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AllowAppStream2.0ToRetrieveObjects",
```



```

    "Effect": "Allow",
    "Principal": {
      "Service": ["appstream.amazonaws.com"]
    },
    "Action": ["s3:GetObject"],
    "Resource": [
      "arn:aws:s3:::bucket/VHD object",
      "arn:aws:s3:::bucket/Setup script object",
      "arn:aws:s3:::bucket/Application icon object",
      "arn:aws:s3:::bucket/Session scripts zip file object"
    ]
  }
}

```

Note

このバケットポリシーの例は、AppStream 2.0 がアクセスできる S3 バケット内の特定のオブジェクトを定義します。プレフィックスとワイルドカードを使用して、App Block の増加に伴うポリシーの管理を簡素化することもできます。バケットポリシーの詳細については、[バケットポリシーの使用](#)を参照してください。一般的なバケット例の詳細については、[バケットポリシーの例](#)を参照してください。

AppStream 2.0 App Block を使用している場合、アプリケーションパッケージを適切な Amazon S3 バケットにアップロードするため、AppStream 2.0 に追加の権限が必要です。AppStream 2.0 App Block の詳細については、「[the section called “AppStream 2.0 App Block”](#)」を参照してください。

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AllowAppStream2.0ToPutAndRetrieveObjects",
      "Effect": "Allow",
      "Principal": {
        "Service": [
          "appstream.amazonaws.com"
        ]
      },
      "Action": [
        "s3:GetObject",

```



```

    "s3:ListBucket",
    "s3:PutObject",
    "s3:GetBucketOwnershipControls"
  ],
  "Resource": [
    "arn:aws:s3:::bucket",
    "arn:aws:s3:::bucket/AppStream2/*",
    "arn:aws:s3:::bucket/Setup script object",
    "arn:aws:s3:::bucket/Application icon object",
    "arn:aws:s3:::bucket/Session scripts zip file object"
  ]
}
]
}

```

アプリケーションを Elastic フリートに関連付ける

アプリケーションは、起動されるアプリケーションのカatalog内でユーザーに表示される前に、Elastic フリートに関連付けておく必要があります。アプリケーションの関連付けは、AppStream 2.0 コンソールを使用して管理できます。Elastic フリーの作成時にアプリケーションを関連付ける方法、および既存のフリーでアプリケーションの関連付けを管理する方法については、[Amazon AppStream 2.0 フリーを更新する](#) を参照してください。

その他のリソース

次のリンクは、Elastic フリーを使用してアプリケーションをパッケージ化して配信するのに役立つ情報およびその他のリソースを提供します。

ソリューション	説明
AWS	• Amazon AppStream 2.0 で安価で安全な要塞ホストに Elastic フリーと Linux を使用する — 一般的な要塞ホストアプリケーションをパッケージ化し、Elastic フリーで Linux インスタンスを使用してそれらを配信する方法について説明します。 • を使用して AppStream 2.0 Elastic フリーアプリケーションの更新を自動化 AWS Systems Managerする — で EC2 インスタンスを使用して VHDs と App Block の

ソリューション	説明
	作成を自動化する方法について説明します AWS Systems Manager。 • Amazon AppStream 2.0 Elastic フリートと Linux の互換性により、アプリケーションを低コストでストリーミングする — アプリケーションを VHD にパッケージ化し、Elastic フリートで Linux インスタンスを使用して配信する方法について説明します。
Liquidware FlexApp	AppStream 2.0 Elastic フリートと Liquidware FlexApp との統合 - Windowsベースのアプリケーションを FlexApp ワンフォーマットにパッケージ化し、AppStream 2.0 Elastic フリートで配信するためのステップバイステップの手順が含まれています。
Numecent クラウドページング	AppStream 2.0 Elastic フリースの概要: クラウドページングでアプリケーションの互換性と移植性を最大化する方法 - Numecent クラウドページングは、アプリケーションの互換性の問題を排除し、最も複雑なレガシーアプリケーションや高度にカスタマイズされたアプリケーションでも、その依存関係とともにパッケージ化され、1日以内にデプロイできるようになります。AppStream 2.0 ユーザーは、IT 部門がさまざまな Windows オペレーティングシステムやデバイス用に再パッケージ化する必要なく、オンデマンドでストリーミングできます。
Turbo.Net	Amazon AppStream 2.0 Elastic フリースの Turbo サポートの導入 — Turbo は、主要なプラットフォームやデバイスでアプリケーションを即座に提供します。Turbo 独自のコンテナテクノロジーにより、インストールを排除し、依存関係、コンフリクト、使用権限を管理します。AppStream 2.0 ユーザーは、Turbo Hub で任意のアプリケーションをすぐに起動できます。

Amazon AppStream 2.0 のフリートタイプのフリートとスタック

Amazon AppStream 2.0 では、アプリケーションのストリーミングプロセスの一環としてフリートインスタンスとスタックを作成します。フリートは、指定したイメージを実行するストリーミングインスタンスで構成されています。スタックは、関連付けられたフリート、ユーザーアクセスポリシー、ストレージ構成で構成されます。

内容

- [Amazon AppStream 2.0 のセッションコンテキスト](#)
- [AppStream 2.0 のフリートタイプ](#)
- [AppStream 2.0 インスタンスファミリー](#)
- [Amazon AppStream 2.0 フリートとスタックを作成する](#)
- [ユーザーのアプリケーションストリーミング体験を最適化するための Amazon AppStream 2.0 フリートのカスタマイズ](#)
- [Amazon AppStream 2.0 フリートを更新する](#)
- [Amazon AppStream 2.0 向け Fleet Auto Scaling](#)
- [マルチセッションに関する推奨事項](#)

Amazon AppStream 2.0 のセッションコンテキスト

次のメソッドのいずれかを実行することによって、ストリーミングアプリケーションにパラメータを渡すことができます。

- CreateStreamingURL AppStream 2.0 API オペレーションでセッションコンテンツを指定します。詳細については、「[CreateStreamingURL](#)」を参照してください。
- sts:TagSession アクセス許可を IAM ロールの信頼ポリシーに追加し、セッションコンテキストを SAML 2.0 ID プロバイダーの認証レスポンスで SAML アサーションとして指定します。詳細については、[ステップ 5: SAML 認証レスポンスのアサーションを作成する](#) および [the section called “ステップ 5: SAML 認証レスポンスのアサーションを作成する”](#)を参照してください。

イメージが 2018 年 10 月 30 日以降にリリースされた AppStream 2.0 エージェントバージョンを使用している場合、セッションコンテキストは Windows または Linux の環境変数としてイメージ内に

保存されます。特定の環境変数については、[ユーザーのアプリケーションストリーミング体験を最適化するための Amazon AppStream 2.0 フリートのカスタマイズ](#) の「User and Instance Metadata for AppStream 2.0 Fleets」を参照してください。

Note

セッションコンテキストパラメータは、AppStream 2.0 ストリーミング URL でユーザーに表示されます。セッションコンテキストパラメータに機密情報や機微情報を入れないことを強くお勧めします。ユーザーがストリーミング URL を変更できるので、セッションコンテキストがエンドユーザーに対して有効であることを確認するために、追加の検証を実行することをお勧めします。たとえば、AppStream 2.0 フリートのユーザーやインスタンスのメタデータなど、セッションコンテキストを他のセッション情報と比較できます。AppStream 2.0 は、セッションコンテキストパラメータに対して検証を実行しません。

セッションコンテキストを使用してストリーミングアプリケーションにパラメータを渡す

次の手順では、セッションコンテキストを使用してウェブブラウザをスタートし、特定のウェブサイトを自動的に開きます。Windows を実行しているインスタンスでは、Firefox を使用します。Linux を実行しているインスタンスでは Chromium を使用します。

セッションコンテキストを使用してウェブサイトを起動するには

1. 左側のナビゲーションペインで、[Images] (イメージ)、[Image Builder] の順に選択します。
2. 使用する Image Builder を選択し、[Running] (実行中) 状態であることを確認してから、[Connect] (接続) を選択します。
3. [Local User] (ローカルユーザー) タブで [管理者] を選択して、Image Builder にログインします。
4. 子フォルダ C:\ を作成します。この例では C:\Scripts を使用します。
5. 新しいフォルダに Windows バッチファイルを作成します。この例では、C:\Scripts \session-context-test.bat を作成し、セッションコンテキストからの URL で Firefox を起動するスクリプトを追加します。

次のスクリプトを実行します。

```
CD "C:\Program Files (x86)\Mozilla Firefox"
```

```
Start firefox.exe %APPSTREAM_SESSION_CONTEXT%
```

6. Image Assistant で session-context-test.bat を追加し、名前を **Firefox** に変更します。

Firefox を追加する必要はありません。この手順では、バッチファイルのみを追加する必要があります。

7. イメージ、フリート、およびスタックを作成します。この例では、**session-context-test-fleet** のフリート名と **session-context-test-stack** のスタック名を使用します。
8. フリートが実行された後で、この例にあるように、session-context パラメータで [create-streaming-url](#) を呼び出すことができます。

```
aws appstream create-streaming-url --stack-name session-context-test-stack \  
--fleet-name session-context-test-fleet \  
--user-id username --validity 10000 \  
--application-id firefox --session-context "www.amazon.com"
```

9. ブラウザでストリーミング URL を開きます。スクリプトファイルが Firefox を起動し、http://www.amazon.com をロードします。

同様に、以下の手順を実行して Linux ストリーミングアプリケーションにパラメータを渡すことができます。

Linux ストリーミングアプリケーションにパラメータを渡す

1. 左側のナビゲーションペインで、[Images] (イメージ)、[Image Builder] の順に選択します。
2. 使用する Image Builder を選択し、[Running] (実行中) 状態であることを確認してから、[Connect] (接続) を選択します。
3. デフォルトで、ImageBuilderAdmin として Image Builder にログインします。
4. 以下のコマンドを実行して、スクリプトファイル (launch-chromium.sh など) を作成します。

```
sudo vim /usr/bin/launch-chromium.sh
```

5. 以下の例にあるように、スクリプトを記述して、実行可能な許可を設定します。

Note

スクリプトでは、常に `#!/bin/bash` と `source /etc/profile` が必要です。

```
#!/bin/bash
source /etc/profile
/usr/bin/chromium-browser $APPSTREAM_SESSION_CONTEXT
```

6. Image Assistant CLI を使用して、`launch-chromium.sh` を追加します。

```
sudo AppStreamImageAssistant add-application \
--name chromium \
--absolute-app-path /usr/bin/launch-chromium.sh
```

7. イメージ、フリート、およびスタックを作成します。この例では、**session-context-test-fleet** のフリート名と **session-context-test-stack** のスタック名を使用します。
8. フリートが実行された後で、この例にあるように、`session-context` パラメータで [create-streaming-url](#) を呼び出すことができます。

```
aws appstream create-streaming-url --stack-name session-context-test-stack \
--fleet-name session-context-test-fleet \
--user-id username \
--application-id chromium --session-context "www.amazon.com"
```

9. ブラウザでストリーミング URL を開きます。バッチファイルが Chromium を起動し、`http://www.amazon.com` をロードします。

AppStream 2.0 のフリートタイプ

フリートタイプによって、インスタンスの実行タイミングと支払い方法が決まります。フリートを作成するときにフリートタイプを指定できます。一度フリートを作成したら、そのフリートタイプを変更することはできません。

可能なフリートタイプは以下のとおりです。

常時オン

ストリーミングインスタンスは、アプリケーションとデスクトップをストリーミングしているユーザーがいなくても、常に実行されます。ストリーミングインスタンスは、ユーザーがストリーミングする前にプロビジョニングされる必要があります。プロビジョニングされるストリーミングインスタンスの数は、オートスケーリングルールを使用して管理されます。

ユーザーがアプリケーションまたはデスクトップを選択すると、それらのストリーミングが直ちに開始されます。ストリーミングしているユーザーがいない場合でも、すべてのストリーミングインスタンスに対して実行中のインスタンス料金が請求されます。

オンデマンド

ストリーミングインスタンスは、ユーザーがアプリケーションとデスクトップをストリーミングしている場合にのみ実行されます。まだユーザーに割り当てられていないストリーミングインスタンスは停止状態になります。ストリーミングインスタンスは、ユーザーがストリーミングする前にプロビジョニングされる必要があります。プロビジョニングされるストリーミングインスタンスの数は、オートスケーリングルールを使用して管理されます。

ユーザーがアプリケーションまたはデスクトップを選択すると、1~2 分の待機時間後にそれらのストリーミングが開始されます。まだユーザーに割り当てられていないストリーミングインスタンスにはコストが低い停止中のインスタンス料金が請求され、ユーザーに割り当てられているストリーミングインスタンスには実行中のインスタンス料金が請求されます。

Elastic

このストリーミングインスタンスのプールは、AppStream 2.0 によって管理されます。ユーザーが起動するアプリケーションまたはデスクトップを選択すると、App Block がダウンロードされ、ストリーミングインスタンスにマウントされた後で、これらのストリーミングが開始されます。Elastic フリート用の App Block の作成に関する詳細については、[App Block](#) を参照してください。

Elastic フリートストリーミングインスタンスに対する実行中のインスタンス料金は、ストリーミングセッションの持続時間分 (最低 15 分) のみが秒単位で請求されます。

フリートタイプ別の請求方法の詳細については、[Amazon AppStream 2.0 の料金](#)を参照してください。

常時オンおよびオンデマンドフリート

常時オンフリートとオンデマンドフリートは、オートスケーリングポリシーを使用して容量を管理するストリーミングインスタンスのプールを表します。ユーザーは、ストリーミングインスタンスを使用してアプリケーションとデスクトップをストリーミングします。常時オンフリートでは、ユーザーのアプリケーションがほぼ瞬時に起動され、ユーザーがストリーミングしていないときでも、インスタンスごとに実行中のインスタンス料金を支払います。オンデマンドフリートでは、ストリーミングインスタンスが起動される 1~2 分の待機時間後にユーザーのアプリケーションが起動され、使用さ

れていないインスタンスにはコストが低い停止中のインスタンス料金を支払い、使用されているインスタンスには実行中のインスタンス料金を支払います。

常時オンおよびオンデマンドフリートインスタンスのアプリケーションは、Image Builder によって作成された AppStream 2.0 イメージ経由で配信されます。Image Builder の作成方法、アプリケーションのインストール方法、およびイメージの作成方法の詳細については、[イメージ](#) をお読みください。

常時オンおよびオンデマンドフリートのストリーミングインスタンスは、ユーザーがストリーミングする前にプロビジョニングし、既存のユーザーへの割り当てを解除する必要があります。固定または動的なフリートオートスケーリングポリシーを使用してフリート内のインスタンス数を管理することで、コストをコントロールしながらユーザーのニーズを満たすために十分な容量を確保することができます。フリートのスケーリングに関する詳細については、[the section called “Fleet Auto Scaling”](#) をお読みください。

Elastic フリート

Elastic フリートは、AppStream 2.0 が管理するストリーミングインスタンスのプールを表します。ユーザーによるアプリケーションとデスクトップのストリーミングのために同時実行数を予測したり、オートスケーリングポリシーを作成して管理したりする必要はありません。ユーザーがストリーミングインスタンスをリクエストすると、プールからストリーミングインスタンスが割り当てられ、設定完了後に利用可能になります。

Elastic フリートは、App Block に格納されているアプリケーションに依存します。ユーザーがカタログからアプリケーションを選択すると、App Block がインスタンスにダウンロードされ、マウントされてから、アプリケーションが起動します。

AWS は、Elastic フリートによるストリーミングインスタンスのプロビジョニングと可用性を管理します。フリートを作成および更新するときは期待される最大同時実行数を設定して、ユーザーの需要を満たすために十分なストリーミングインスタンス制限が設定されていることを確認する必要があります。

Elastic フリート用の App Block の作成に関する詳細については、[App Block](#) を参照してください。

AppStream 2.0 インスタンスファミリー

Amazon AppStream 2.0 ユーザーは、作成したスタックからアプリケーションをストリーミングします。各スタックはフリートに関連付けられています。フリートを作成するときは、指定したインスタ

インスタンスタイプによって、フリートに使用するホストコンピュータのハードウェアが決まります。インスタンスタイプごとに異なるコンピューティング、メモリ、GPU が提供されます。インスタンスタイプは、これらの機能に基づいてインスタンスファミリーにグループ分けされます。ハードウェアの仕様と料金情報については、[AppStream 2.0 の料金](#) を参照してください。

フリートまたは Image Builder を作成するときは、そのフリートを実行するインスタンスファミリーと互換性のあるイメージを選択する必要があります。

- 新しい Image Builder を起動すると、イメージレジストリのイメージの一覧が表示されます。適切なベースイメージを選択します。
- フリートを起動するときは、選択しているプライベートイメージが適切なベースイメージから作成されたものであることを確認してください。

次の表は、使用可能なインスタンスファミリーの要約と、それぞれのベースイメージ命名形式を提供します。フリートでストリーミングするアプリケーションの要件に基づいて、インスタンスファミリーからインスタンスタイプを選択し、次の表に従ってベースイメージと一致させます。

Note

ユースケースにリアルタイムオーディオ動画 (AV)、または高いフレームレートが必要で、表示パフォーマンスが想定どおりではない場合は、より大きなインスタンスサイズにスケールアップすることをご検討ください。

Graphics Pro インスタンスは、Graphics Pro インスタンスタイプをサポートするハードウェアのサポートが終了するため、10/31/2025 AWS 以降は使用できなくなります。

Graphics Design インスタンスは、Graphics Design インスタンスタイプをサポートするハードウェアのサポートが終了するため、12/31/2025 AWS 以降は使用できなくなります。

インスタンスファミリー	説明	ベースイメージ名
汎用	ウェブブラウザとほとんどのビジネスアプリケーションを実行するための基本的なコンピューティングリソース。	AppStream-WinServer- <i>OperatingSystemVersion</i> - <i>MM-DD-YYYY</i> AppStream-AmazonLinux2- <i>MM-DD-YYYY</i>

インスタンスファミリー	説明	ベースイメージ名
		AppStream-RockyLinux8- <i>MM-DD-YYYY</i> AppStream-RHEL8- <i>MM-DD-YYYY</i>
コンピューティング最適化	高パフォーマンスプロセッサの恩恵を受けるコンピューティング関連アプリケーション向けに最適化します。	AppStream-WinServer- <i>OperatingSystemVersion-MM-DD-YYYY</i> AppStream-AmaonLinux2- <i>MM-DD-YYYY</i> AppStream-RockyLinux8- <i>MM-DD-YYYY</i> AppStream-RHEL8- <i>MM-DD-YYYY</i>
メモリを最適化	大量のデータを処理するメモリ負荷の高いアプリケーション向けに最適化します。  Note AppStream 2.0 z1d ベースのインスタンスを使用する場合は、2018 年 6 月 12 日以降に公開された AppStream 2.0 ベースのイメージから作成されたイメージからプロビジョニングする必要があります。	AppStream-WinServer- <i>OperatingSystemVersion-MM-DD-YYYY</i> AppStream-AmaonLinux2- <i>MM-DD-YYYY</i> AppStream-RockyLinux8- <i>MM-DD-YYYY</i> AppStream-RHEL8- <i>MM-DD-YYYY</i>

インスタンスファミリー	説明	ベースイメージ名
Graphics Pro	NVIDIA Tesla M60 GPU を使用し、DirectX、OpenGL、OpenCL、または CUDA を使用してグラフィックアプリケーション向けに高パフォーマンスのワークステーションに似た使い勝手を提供します。	AppStream-Graphics-Pro- <i>OperatingSystemVersion</i> -MM-DD-YYYY
グラフィックスデザイン	AMD FirePro S7150x2 サーバー GPU と AMD Multiuser GPU テクノロジーを使用して、DirectX、OpenGL、または OpenCL を使用するグラフィックスアプリケーションをサポートしています。	AppStream-Graphics-Design- <i>OperatingSystemVersion</i> -MM-DD-YYYY
Graphics G4dn	NVIDIA T4 GPU は、グラフィックスを多用するアプリケーションをサポートするために使用します。	AppStream-Graphics-G4dn- <i>OperatingSystemVersion</i> -MM-DD-YYYY AppStream-Graphics-G4dn-RockyLinux8-MM-DD-YYYY AppStream-Graphics-G4dn-RHEL8-MM-DD-YYYY
Graphics G5	NVIDIA A10G GPU を使用して、リモートワークステーション、ビデオレンダリング、ゲームなど、グラフィックスを多用するアプリケーションをサポートし、リアルタイムで忠実度の高いグラフィックスを生成します。	AppStream-Graphics-G5- <i>OperatingSystemVersion</i> -07-17-2023 AppStream-Graphics-G5-RockyLinux8-MM-DD-YYYY AppStream-Graphics-G5-RHEL8-MM-DD-YYYY

AppStream 2.0 インスタンスには 200 GB の固定サイズのボリュームが 1 つあり、C ドライブに使用されます。AppStream 2.0 は非永続的であるため、各インスタンスのボリュームは、各ユーザーセッションの終了後に直ちに削除されます。

詳細については次を参照してください:

- [AppStream 2.0 ベースイメージとマネージド型イメージの更新に関するリリースノート](#)
- [Amazon AppStream 2.0 Service Quotas](#)
- [AppStream 2.0 の料金](#)

Amazon AppStream 2.0 フリートとスタックを作成する

Amazon AppStream 2.0 では、アプリケーションをストリーミングするために、スタックに関連付けられているフリートが含まれる環境に加えて、少なくとも 1 つのアプリケーションイメージが必要になります。このチュートリアルでは、フリートおよびスタックのセットアップ手順と、スタックへのアクセス権をユーザーに付与する方法を示します。まだ行っていない場合は、最初に「[Amazon AppStream 2.0 の使用を開始する: サンプルアプリケーションを使用してセットアップする](#)」の手順を確認することをお勧めします。

使用するイメージを作成する場合は、「[チュートリアル: AppStream 2.0 コンソールを使用してカスタム AppStream 2.0 イメージを作成する](#)」を参照してください。

フリートを Active Directory ドメインに結合する場合は、Active Directory ドメインを設定してから、以下のステップを行ってください。詳細については、「[AppStream 2.0 でのアクティブディレクトリの使用](#)」を参照してください。

タスク

- [Amazon AppStream 2.0 でフリートを作成する](#)
- [Amazon AppStream 2.0 でスタックを作成する](#)
- [ユーザーに Amazon AppStream 2.0 へのアクセスを提供する](#)
- [Amazon AppStream 2.0 リソースのクリーンアップ](#)

Amazon AppStream 2.0 でフリートを作成する

ユーザーアプリケーションを起動し、ストリーミングするフリートをセットアップおよび作成します。

Note

常時オンまたはオンデマンドフリートを作成するには、ユーザーがストリーミングソースとして使用できる常時オンまたはオンデマンドフリートを作成するためのアプリケーションがインストールされたイメージが必要です。イメージを作成するには、[the section called “チュートリアル: コンソールを使用したカスタムイメージの作成”](#) を参照してください。Elastic フリートを作成するには、App Block に関連付けられたアプリケーションが必要です。Elastic フリート用のアプリケーションと App Block を作成するには、[アプリケーションマネージャー](#) を参照してください。

フリートをセットアップして設定する

1. AppStream 2.0 コンソール (<https://console.aws.amazon.com/appstream2>) を開きます。
2. コンソールに初めてアクセスする場合は、[Get Started] を選択するか、または、左側のナビゲーションメニューから [Fleets] を選択します。[Create Fleet] を選択します。
3. [Step 1: Select fleet type] (ステップ 1: フリートタイプの選択) では、フリートタイプの詳細を確認し、ユースケースに基づいて作成するフリートのタイプを選択して、[Next] (次へ) をクリックします。

Note

フリートタイプごとに、その即時利用可能性と、支払い方法が異なります。詳細については、「[AppStream 2.0 のフリートタイプ](#)」を参照してください。

4. [Step 2: Configure fleet] (ステップ 2: フリートの設定) では、以下の [details] (詳細情報) を入力します:
 - [Name] (名前) には、フリート用の一意の名前識別子を入力します。特殊文字は使用できません。
 - [Display Name] (表示名) には、フリート用に表示する名前を入力します (最大 100 文字)。特殊文字は使用できません。
 - [Description] (説明) には、フリートの説明を入力します (最大 256 文字)。
 - [Choose instance type] で、アプリケーションのパフォーマンス要件に合うインスタンスタイプを選択します。フリート内のすべてのストリーミングインスタンスが選択したインスタンスタイプで起動します。詳細については、「[AppStream 2.0 インスタンスファミリー](#)」を参照してください。

- Elastic フリートの場合は、[Choose platform type] (プラットフォームタイプの選択) で、ユーザーのアプリケーションの要件に一致するオペレーティングシステムを選択します。
- [Maximum session duration in minutes] (セッションの最大継続時間 (分単位)) には、ストリーミングセッションがアクティブな状態を維持できる最大時間を選択します。この制限に達する 5 分前にユーザーがまだストリーミングインスタンスに接続されている場合は、切断される前に、開いているドキュメントを保存するように求められます。この時間が経過すると、インスタンスが終了され、新しいインスタンスに置き換えられます。AppStream 2.0 コンソールで設定できる最大セッション時間は 5,760 分 (96 時間) です。AppStream 2.0 API と CLI を使用して設定できる最大セッション時間は 432000 秒 (120 時間) です。
- [Disconnect timeout in minutes (切断タイムアウト (分単位))] では、ユーザーが切断した後にストリーミングセッションをアクティブのままにする時間を選択します。切断、またはこの時間間隔内のネットワークの中断の後、ユーザーが再接続を試みる場合、前のセッションに接続されます。それ以外の場合は、新しいストリーミングインスタンスで新しいセッションに接続されます。スタックをリダイレクト URL が指定されたフリートに関連付けると、ユーザーのストリーミングセッションの終了後に、ユーザーはその URL にリダイレクトされます。

ユーザーが AppStream 2.0 ツールバーで [End Session (セッション終了)] や [Logout (ログアウト)] を選択してセッションを終了した場合、切断タイムアウトは適用されません。代わりに、開いているドキュメントを保存するかどうかの確認がユーザーに求められ、その後すぐにストリーミングインスタンスから切断されます。ユーザーが使用しているインスタンスは終了されます。

- [Idle disconnect timeout in minutes (アイドル切断タイムアウト (分単位))] では、ユーザーがストリーミングセッションから切断されるまでにアイドル状態 (非アクティブ) であることができる時間と、[Disconnect timeout in minutes (切断タイムアウト (分単位))] 期間の開始時刻を選択します。ユーザーは、アイドル状態が原因で切断される前に通知されます。ユーザーが [Disconnect timeout in minutes (切断タイムアウト (分単位))] で指定した期間が経過する前にストリーミングセッションへの再接続を試みると、前のセッションに接続されます。それ以外の場合は、新しいストリーミングインスタンスで新しいセッションに接続されます。この値を 0 に設定すると無効になります。この値を無効にした場合、ユーザーはアイドル状態が原因で切断されることはありません。

Note

ユーザーがストリーミングセッション中にキーボードまたはマウスの入力を停止した場合、アイドル状態であると見なされます。ドメインに参加しているフリートの場合、アイドル切断タイムアウトのカウントダウンは、ユーザーが Active Directory ド

メインパスワードまたはスマートカードを使用してログインするまで開始されません。ファイルのアップロードとダウンロード、オーディオ入力、オーディオ出力、およびピクセルの変更は、ユーザーアクティビティとはなりません。[Idle disconnect timeout in minutes (アイドル切断タイムアウト (分単位))] の期間が経過した後も引き続きアイドル状態である場合、ユーザーは切断されます。

- Elastic フリートの場合は、[Max concurrent sessions] (最大同時セッション数) で、このフリートに必要な同時セッションの最大数を指定します。

Note

「アカウントの同時セッションの最大数を超過しました」というエラーメッセージが表示された場合は、Service Quotas コンソール (<https://console.aws.amazon.com/servicequotas/>) で、制限の引き上げを送信できます。詳細については「Service Quotas ユーザーガイド」の「[クォータの引き上げのリクエスト](#)」を参照してください。

- 複数のユーザーセッション — 1 つのインスタンスで複数のユーザーセッションをプロビジョニングする場合は、このオプションを選択します。デフォルトでは、すべての一意のユーザーセッションはインスタンスによって処理されます (シングルセッション)。

Note

マルチセッションは、Windows オペレーティングシステムを搭載した常時オンおよびオンデマンドフリートでのみ利用できます。マルチセッションは Elastic フリートや Linux オペレーティングシステムでは使用できません。

2023 年 5 月 15 日以降にリリースされたベースイメージとマネージドイメージ更新でのみ、マルチセッションフリートをサポートしています。詳細については、[the section called “ベースイメージおよびマネージド型イメージの更新版リリースノート”](#)を参照してください。

- インスタンスあたりの最大セッション数 — 1 つのインスタンスあたりのユーザーセッションの最大数。この値は、エンドユーザーのアプリケーションパフォーマンスのニーズに基づいて選択する必要があります。フリートのインスタンスあたりの最大セッション数は、プロビジョニング後に調整することもできます。その場合、既存のユーザーセッションとインスタンスには影響しませんが、フリートはインスタンスあたりの最大セッション数の新しい値と一致す

るようになります。2~50 の範囲の値にする必要があります。この値をフリートに設定する前に、「[the section called “マルチセッションに関する推奨事項”](#)」を参照してください。

- 常時オンおよびオンデマンドフリートの場合は、[最小容量] で、予想される同時ユーザーの最小数に基づいてフリートの最小インスタンス数 (シングルセッションフリートの場合) または最小ユーザーセッション数 (マルチセッションフリートの場合) を選択します。
- 常時オンおよびオンデマンドフリートの場合は、[最大容量] で、予想される同時ユーザーの最大数に基づいてフリートの最大インスタンス数 (シングルセッションフリートの場合) または最大ユーザーセッション数 (マルチセッションフリートの場合) を選択します。

 Note

マルチセッションの場合は、ユーザーセッション数に基づいて容量を指定する必要があります。このサービスは、フリート設定とインスタンスあたりの最大セッション数に基づいて、起動する必要があるインスタンスの数を計算します。

- [Stream view (ストリームビュー)] で、ストリーミングセッション中にユーザーに表示される AppStream 2.0 ビューを選択します。[Application (アプリケーション)] を選択すると、ユーザーが開いたアプリケーションのウィンドウのみが表示されます。[Desktop (デスクトップ)] を選択すると、オペレーティングシステムが提供する標準デスクトップが表示されます。

 Note

デフォルトでは、AppStream 2.0 は、ストリーミングセッション中にユーザーが開いたアプリケーションのウィンドウのみを表示します。ユーザーの [Desktop (デスクトップ)] ビューを有効にするには、2020 年 2 月 19 日以降にリリースされた AppStream 2.0 エージェントのバージョンを使用する AppStream 2.0 イメージを使用するようにフリートを構成します。

- [Scaling details (Advanced) (スケーリングの詳細 (アドバンスド))] で、特定の条件下で AppStream 2.0 が使用してフリートの容量を調整することができるデフォルトのスケーリングポリシーを指定します。フリートのサイズは、指定した最小および最大容量によって制限されることに注意してください。詳細については、「[Amazon AppStream 2.0 向け Fleet Auto Scaling](#)」を参照してください。
- [IAM role (Advanced)] (IAM ロール (アドバンスド)) では、アカウントからの IAM ロールを AppStream 2.0 フリートインスタンスに適用するときに、AWS 認証情報を手動で管理することなく、フリートインスタンスから AWS API リクエストを実行することができます。IAM ロールを適用するには、次のいずれかを行います。

- AWS アカウントで既存の IAM ロールを使用するには、IAM ロールリストから使用するロールを選択します。ロールは、フリートインスタンスからアクセス可能である必要があります。詳細については、「[AppStream 2.0 ストリーミングインスタンスで使用するための既存の IAM ロールの設定](#)」を参照してください。
 - 新しい IAM ロールを作成するには、[Create new IAM role (新しい IAM ロールの作成)] を選択し、[AppStream 2.0 ストリーミングインスタンスで使用する IAM ロールを作成する方法](#)のステップを実行します。
 - Elastic フリートの場合は、[USB Redirection (advanced)] (USB リダイレクト (アドバンスド)) で、Windows ネイティブのクライアントを使用するときにストリーミングセッションにリダイレクトできる、ローカルデバイスにアタッチされた USB デバイスのタイプを指定する文字列を最大 10 個指定できます。詳細については、「[the section called “ストリーミングアプリケーションで使用する USB デバイスを認証する”](#)」を参照してください。
5. [Next (次へ)] を選択します。
 6. 常時オンまたはオンデマンドフリートの作成を選択する場合は、[Step 3: Choose an Image] (ステップ 3: イメージの選択) で、ニーズに即したイメージを選択してから、[Next] (次へ) を選択します。
 7. Elastic フリートの作成を選択する場合は、[Step 3: Assign applications] (ステップ 3: アプリケーションの割り当て) で、ユーザーがこのフリートから起動できるアプリケーションを選択します。
 8. [Step 4: Configure Network] で、以下の操作を行います。
 - パブリックサブネットを持つ VPC にフリートインスタンスのインターネットアクセスを追加するには、[Default Internet Access] を選択します。NAT ゲートウェイを使用してインターネットアクセスを提供する場合は、[Default Internet Access (デフォルトのインターネットアクセス)] を未選択のままにします。詳細については、「[インターネットアクセス](#)」を参照してください。

 Note

S3 バケットへの保存に依存する機能を有効にする場合は、VPC が Amazon Simple Storage Service (S3) へのアクセスを提供する必要があります。詳細については、「[the section called “Amazon S3 VPC エンドポイント”](#)」を参照してください。

- [VPC] と [Subnet 1 (サブネット 1)] では、VPC とアプリケーションが必要とするネットワークリソースにアクセスできるサブネットを 1 つ以上選択します。耐障害性を高めるために、

異なるアベイラビリティーゾーンで 2 つのサブネットを選択することをお勧めします。詳細については、「[プライベートサブネットの VPC および NAT ゲートウェイを設定する](#)」を参照してください。

 Note

Elastic フリートでは、異なるアベイラビリティーゾーン内にあるサブネットを少なくとも 2 つ指定する必要があります。

独自の VPC とサブネットがない場合は、[デフォルトの VPC](#) を使用するか、独自の VPC を作成できます。独自の VPC とサブネットを作成するには、[Create a new VPC (新しい VPC を作成する)] リンクおよび [Create new subnet (新しいサブネットを作成する)] リンクを選択して作成します。これらのリンクを選択すると、Amazon VPC コンソールが開きます。VPC とサブネットを作成したら、AppStream 2.0 コンソールに戻り、[Create a new VPC (新しい VPC を作成する)] リンクと [Create new subnet (新しいサブネットを作成する)] リンクの左側にある更新アイコンを選択してリストに表示します。詳細については、「[AppStream 2.0 用の VPC の設定](#)」を参照してください。

- [Security group(s) (セキュリティグループ)] で、このフリートに関連付ける最大 5 つのセキュリティグループを選択します。独自のセキュリティグループを持たず、デフォルトのセキュリティグループを使用しない場合は、[Create new security group (新しいセキュリティグループを作成する)] リンクを選択して、セキュリティグループを作成します。Amazon VPC コンソールでサブネットを作成したら、AppStream 2.0 コンソールに戻り、[Create new security group (新しいセキュリティグループを作成する)] リンクの左側にある更新アイコンを選択してリストに表示します。詳細については、「[Amazon AppStream 2.0 のセキュリティグループ](#)」を参照してください。
- 常時オンおよびオンデマンドフリートの場合は、[Active Directory Domain (Optional)] (アクティブディレクトリドメイン (オプション)) で、アクティブディレクトリと、ストリーミングインスタンスのコンピュータオブジェクト用の組織単位 (OU) を選択します。選択されたネットワークアクセス設定で DNS による解決が有効であり、ディレクトリと通信できることを確認します。詳細については、「[AppStream 2.0 でのアクティブディレクトリの使用](#)」を参照してください。

9. [Next (次へ)] を選択します。

10. [Step 5: Review (ステップ 5: 確認)] で、フリートの詳細を確認します。任意のセクションの設定を変更するには、[Edit] を選択し、必要な変更を加えます。設定内容を確認したら、[作成] を選択します。

11. 料金確認ダイアログボックスで、確認チェックボックスをオンにし、[作成] を選択します。

 Note

フリートを作成するのに十分な制限 (クォータ) がないというエラーメッセージが表示された場合は、<https://console.aws.amazon.com/servicequotas/> の Service Quotas コンソールから制限引き上げリクエストを送信してください。詳細については「Service Quotas ユーザーガイド」の「[クォータの引き上げのリクエスト](#)」を参照してください。

12. フリートが作成されている間は、[Fleets] (フリート) リストでフリートのステータスが [Starting] (起動中) として表示されます。ステータスが [実行中] になるまで、フリートのステータスを更新するには、定期的に [更新] アイコンを選択します。フリートをスタックに関連付けることはできません。また、フリートのステータスが [Running] になるまでストリーミングセッションにフリートを使用することはできません。

Amazon AppStream 2.0 でスタックを作成する

フリートへのアクセスを制御するために、スタックをセットアップして作成します。

 Note

Google ドライブ、OneDrive、およびアプリケーション設定の永続化は、Windows フリートに関連付けられたスタックのみで有効化できます。既存のスタックを Linux フリートに関連付ける前に、これらの設定が無効化されていることを確認してください。

スタックをセットアップして作成するには

1. 左側のナビゲーションペインで、[Stacks] を選択し、それから [Create Stack] を選択します。
2. [Step 1: Stack Details] (ステップ 1: スタックの詳細情報) の [Stack details] (スタックの詳細情報) に、スタック用の一意の名前識別子を入力します。オプションで、以下の操作を行うことができます。
 - Display name (表示名) — スタックの表示名を入力します (最大 100 文字)。
 - Description (説明) — スタックの説明を入力します (最大 256 文字)。
 - Redirect URL (リダイレクト URL) — エンドユーザーがストリーミングセッションの終了後にリダイレクトされる URL を指定します。

- フィードバック URL — ユーザーがアプリケーションストリーミングのエクスペリエンスについてフィードバックを送信する [フィードバックの送信] リンクをクリックした後、リダイレクトされる URL を指定します。URL を指定しない場合、このリンクは表示されません。
- Fleet (フリート) — 既存のフリートを選択するか、新しいフリートを作成してスタックに関連付けます。
- ストリーミングプロトコル環境設定 — スタックで優先するストリーミングプロトコル、UDP または TCP を指定します。現在、UDP は Windows ネイティブクライアントでのみサポートされます。詳細については、「[システム要件と機能のサポート \(AppStream 2.0 クライアント\)](#)」を参照してください。
- Tags (タグ) — [Add Tag (タグの追加)] を選択し、タグのキーと値を入力します。さらにタグを追加するには、この手順を繰り返します。詳細については、「[Amazon AppStream 2.0 リソースにタグを付ける](#)」を参照してください。
- VPC Endpoints (Advanced) (VPC エンドポイント (アドバンスド)) — Virtual Private Cloud (VPC) に、[インターフェイス VPC エンドポイント](#) (インターフェイスエンドポイント) であるプライベートリンクを作成できます。インターフェイスエンドポイントの作成を開始するには、[Create VPC Endpoint (VPC エンドポイントの作成)] を選択します。このリンクを選択すると、VPC コンソールが開きます。エンドポイントの作成を完了するには、[チュートリアル: インターフェイス VPC エンドポイントからの作成とストリーミング](#) の To create an interface endpoint のステップ 3~6 に従います。

インターフェイスエンドポイントを作成したら、VPC 内でストリーミングトラフィックを維持するために使用できます。

- Embed AppStream 2.0 (AppStream 2.0 の埋め込み) (オプション) — ウェブページに AppStream 2.0 ストリーミングセッションを埋め込むには、埋め込みストリーミングセッションをホストするドメインを指定します。埋め込みストリーミングセッションは、HTTPS (TCP ポート 443) 経由でのみサポートされます。

 Note

埋め込み AppStream 2.0 ストリーミングセッションを設定するには、前提条件を満たし、追加のステップを実行する必要があります。詳細については、「[埋め込み Amazon AppStream 2.0 ストリーミングセッション](#)」を参照してください。

3. [Next (次へ)] を選択します。
4. [Step 2: Enable Storage] で、以下のうち、1 つ以上を選択して、ユーザーに永続的ストレージを提供できます。

- Home Folders (ホームフォルダ) — アプリケーションストリーミングセッション中に、ユーザーはホームフォルダにファイルを保存し、ホームフォルダ内の既存のファイルにアクセスできます。ホームフォルダの有効化要件の詳細については、「[AppStream 2.0 ユーザーのホームフォルダを有効にする](#)」を参照してください。
- Google Drive for Google — ユーザーは、必要に応じて、Google Drive for Google Workspace のアカウントを AppStream 2.0 にリンクすることもできます。アプリケーションストリーミングセッション中、Google ドライブアカウントへのサインイン、Google ドライブへのファイル保存、Google ドライブの既存のファイルへのアクセスを行うことができます。Google ドライブを有効にできるのは、Google Workspace ドメイン内のアカウントに対してのみで、個人の Gmail アカウントに対してはできません。

 Note

Google ドライブの有効化は、Linux ベースのスタック、またはマルチセッションフリートに関連付けられたスタックではサポートされていません。

 Note

[Google Drive を有効化] を選択したら、Google Workspace アカウントに関連付けられている組織ドメインの名前を 1 つ以上入力します。アプリケーションストリーミングセッション中の Google ドライブへのアクセスは、指定するドメインにあるユーザーに制限されます。最大 10 個のドメインを指定できます。Google ドライブの有効化要件の詳細については、「[AppStream 2.0 ユーザーの Google ドライブを有効にする](#)」を参照してください。

- OneDrive for Business — ユーザーは OneDrive for Business のアカウントを AppStream 2.0 にリンクすることができます。アプリケーションストリーミングセッション中、OneDrive アカウントへのサインイン、OneDrive へのファイル保存、OneDrive の既存のファイルへのアクセスを行うことができます。OneDrive は、OneDrive ドメインのアカウントに対してのみ有効にすることができます。個人アカウントに対して設定することはできません。

 Note

OneDrive の有効化は、Linux ベースのスタック、またはマルチセッションフリートに関連付けられたスタックではサポートされていません。

Note

[Enable OneDrive] を選択したら、OneDrive アカウントに関連付けられている組織ドメインの名前を少なくとも 1 つ入力します。アプリケーションストリーミングセッション中の OneDrive へのアクセスは、指定するドメインにあるユーザーに制限されます。最大 10 個のドメインを指定できます。OneDrive の有効化に関する要件の詳細については、[AppStream 2.0 ユーザーの OneDrive を有効にする](#) を参照してください。

5. [Next (次へ)] を選択します。
6. Step 3: User Settings で、次の設定を構成します。完了したら、[Review (確認)] を選択します。

クリップボード、ファイル転送、ローカルデバイスへのプリント、認証のアクセス許可:

Note

[Active Directory へのスマートカードサインイン] は現在、マルチセッションフリートでは利用できません。

- Clipboard (クリップボード) — デフォルトでは、ユーザーは、ローカルデバイスとストリーミングアプリケーション間でデータをコピーアンドペーストすることができます。クリップボードオプションを制限することで、ユーザーによるデータの貼り付け先をリモートストリーミングセッションに限定したり、データのコピー先をローカルデバイスに限定したりできます。クリップボードオプションを全面的に無効化することもできます。ユーザーはストリーミングセッション時にアプリケーション間でコピーアンドペーストできます。[ローカルデバイス文字数制限にコピー] または [リモートセッション文字数制限に貼り付け] を選択するか、またはその両方を選択して、AppStream 2.0 ストリーミングセッションの内外でクリップボードを使用する際にユーザーがコピーまたは貼り付けることができるデータの量を制限できます。値は 1 ~ 20,971,520 (20 MB) で、指定しない場合はデフォルト値は最大値になります。
- File transfer (ファイル転送) — デフォルトでは、ユーザーは、ローカルデバイスとストリーミングセッション間でファイルをアップロードおよびダウンロードすることができます。ファイル転送オプションを制限することで、ユーザーによるファイルのアップロード先をストリーミングセッションに限定したり、ファイルのダウンロード先をローカルデバイスに限定したりできます。ファイル転送を全面的に無効化することもできます。

⚠ Important

ユーザーがストリーミングセッション中にローカルドライブとフォルダにアクセスするために AppStream 2.0 ファイルシステムリダイレクトが必要な場合は、ファイルのアップロードとダウンロードの両方を有効にする必要があります。ファイルシステムリダイレクトを使用するには、ユーザーには、AppStream 2.0 クライアントバージョン 1.0.480 以降がインストールされている必要があります。詳細については、「[AppStream 2.0 ユーザーのファイルシステムリダイレクトを有効にする](#)」を参照してください。

- Print to local device (ローカルデバイスへの出力) — デフォルトでは、ユーザーは、ストリーミングアプリケーションからローカルデバイスに出力することができます。アプリケーションの [Print (出力)] を選択すると、.pdf ファイルをダウンロードして、ローカルプリンタに出力することができます。ユーザーがローカルデバイスに出力できないようにするには、このオプションを無効にします。
- Active Directory のパスワードサインイン — ユーザーは、Active Directory ドメインに参加している AppStream 2.0 ストリーミングインスタンスにサインインするために、Active Directory ドメインパスワードを入力できます。

Active Directory のスマートカードサインインを有効にすることもできます。少なくとも 1 つの認証を有効にする必要があります。

- Smart card sign in for Active Directory (Active Directory へのスマートカードサインイン) — ユーザーはローカルコンピュータに接続されたスマートカードリーダーとスマートカードを使用して、Active Directory ドメインに参加している AppStream 2.0 ストリーミングインスタンスにサインインできます。

[Password sign in for Active Directory (Active Directory のパスワードサインイン)] を有効にすることもできます。少なくとも 1 つの認証方法を有効にしてください。

i Note

クリップボード、ファイル転送、およびローカルデバイスへのプリント — これらの設定は、ユーザーが AppStream 2.0 のデータ転送機能を使用できるかどうかのみを制御します。イメージを使用してブラウザ、ネットワークプリンタ、またはその他のリモートリソースにアクセスできる場合、ユーザーは別の方法を使用してストリーミングセッション間でデータを転送することができます。

認証設定 — これらの設定は、Windows の AppStream 2.0 ストリーミングインスタンス (フリートまたは Image Builder) へのサインインに使用できる認証方法のみを制御します。ユーザーは、ストリーミングインスタンスにサインインした後、セッション内認証に使用できる認証方法を制御しません。Windows サインインおよびセッション中の認証にスマートカードを使用するための構成要件については、[スマートカード](#) を参照してください。これらの設定は、Linux ベースのスタック向けにサポートされていません。

タイムゾーン:

- [リモートセッションのタイムゾーンを自動的に設定] — この設定は、ストリーミングに使用されるタイムゾーンを同期して、ユーザーのデバイスに設定されているタイムゾーンと一致させます。ユーザーはこれを上書きし、独自の優先タイムゾーンを設定できます。

アプリケーション設定の永続化:

- Enable Application Settings Persistence (アプリケーション設定の永続化を有効にする) — ユーザーによるアプリケーションのカスタマイズや Windows 設定は各ストリーミングセッション後に自動的に保存され、次のセッションで適用されます。これらの設定は、アプリケーション設定の永続化が有効になっている AWS リージョン内のアカウントの Amazon Simple Storage Service (Amazon S3) バケットに保存されます。
- Settings Group (設定グループ) — 設定グループは、このスタックに保存されているどのアプリケーション設定をストリーミングセッションで使用するかを決定します。同じ設定グループを別のスタックに適用すると、両方のスタックで同じアプリケーション設定が使用されます。デフォルトでは、設定グループ値はスタックの名前です。

Note

アプリケーション設定の永続化を有効にして管理するための要件については、「[AppStream 2.0 ユーザーのアプリケーション設定の永続化を有効にする](#)」を参照してください。

7. [Step 4: Review (ステップ 4: 確認)] で、スタックの詳細を確認します。任意のセクションの設定を変更するには、[Edit] を選択し、必要な変更を加えます。設定内容を確認したら、[作成] を選択します。

サービスがリソースをセットアップした後、[Stacks] ページが表示されます。新しいスタックが使用できる状態になったら、ステータスに [Active] と表示されます。

ユーザーに Amazon AppStream 2.0 へのアクセスを提供する

関連付けられたフリートを使用してスタックを作成した後は、AppStream 2.0 ユーザープールの SAML 2.0 [single sign-on (SSO)]、または AppStream 2.0 API を通じてユーザーにアクセス権を付与できます。詳細については、[Amazon AppStream 2.0 でのユーザープール管理](#)および[Amazon AppStream 2.0 と SAML 2.0 の統合](#)を参照してください。

Note

AppStream 2.0 ユーザープール内のユーザーは、Active Directory ドメインに参加しているフリートを持つスタックに割り当ててすることはできません。

ユーザーに AppStream 2.0 へのアクセスを提供したら、ユーザーは、ウェブブラウザを使用するか、AppStream 2.0 クライアントアプリケーションを使用して、AppStream 2.0 ストリーミングセッションを開始することができます。AppStream 2.0 ユーザープールを介してユーザーにアクセスを提供する場合、ストリーミングセッションにウェブブラウザを使用する必要があります。SAML 2.0 または AppStream 2.0 API を使用する場合は、AppStream 2.0 クライアントをそれらのユーザーが使用できるようにすることができます。AppStream 2.0 クライアントは、AppStream 2.0 ストリーミングセッション中に追加機能を必要とするユーザー用に設計されたネイティブアプリケーションです。詳細については、「[AppStream 2.0 クライアント経由でアクセスを提供する](#)」を参照してください。

Amazon AppStream 2.0 リソースのクリーンアップ

実行中のフリートを停止してアクティブなスタックを削除することで、リソースを開放し、アカウントに対する意図しない料金が発生することを回避できます。未使用で実行中のフリートを停止することをお勧めします。

関連付けられているフリートと一緒にスタックを削除することはできません。

リソースをクリーンアップするには

1. ナビゲーションペインで、[Stacks] を選択します。
2. スタックを選択し、[Actions] を選択して、[Disassociate Address] を選択します。確認ダイアログボックスで、[関連付け解除] を選択します。

3. ナビゲーションペインの [Fleets] を選択します。
4. 停止するフリートを選択し、[アクション]、[停止] の順に選択します。フリートを停止するには約 5 分かかります。
5. フリートのステータスが [Stopped] のときは、[Actions]、[Delete] の順に選択します。
6. ナビゲーションペインで、[Stacks] を選択します。
7. スタックを選択し、[Actions]、[Delete] を選択します。

ユーザーのアプリケーションストリーミング体験を最適化するための Amazon AppStream 2.0 フリートのカスタマイズ

AppStream 2.0 フリートインスタンスをカスタマイズして、ユーザーのアプリケーションストリーミング体験を最適化するには、AppStream 2.0 環境の特定の側面を定義します。たとえば、環境変数を永続化して、アプリケーション間で設定を動的に渡し、適用されているデフォルトファイルの関連付けをすべてのユーザーに設定できます。概要として、フリートインスタンスのカスタマイズには次のタスクが含まれます。

- Image Builder に接続し、必要に応じてカスタマイズする。
- Image Builder で Image Assistant を使用して、カスタマイズ内容を含む新しいイメージを作成する。
- 新しいフリートインスタンスを作成するか、既存のフリートインスタンスを変更する。フリートインスタンスを設定する際、作成およびカスタマイズされた新しいイメージを選択します。
- 新しいスタックを作成するか、既存のスタックを変更して、フリートインスタンスと関連付ける。

Note

特定のフリートをカスタマイズする場合は、Active Directory 環境で、グループポリシーマネジメントコンソール (GPMC) を使用して、ドメインに結合されたコンピュータでグループポリシーオブジェクト (GPO) 設定を更新する必要がある場合があります。

内容

- [Amazon AppStream 2.0 で環境変数を保持する](#)
- [Amazon AppStream 2.0 でユーザー向けにデフォルトファイルの関連付けを設定する](#)

- [Amazon AppStream 2.0 で Internet Explorer 拡張セキュリティ設定を無効にする](#)
- [Amazon AppStream 2.0 でユーザーのストリーミングセッション用に Internet Explorer のデフォルトホームページを変更する](#)
- [Amazon AppStream 2.0 フリートのユーザーおよびインスタンスのメタデータ](#)

Amazon AppStream 2.0 で環境変数を保持する

環境変数を使用すると、アプリケーション間で設定を動的に渡すことができます。たとえば、多くの工学アプリケーションでは、環境変数を使用して、設置するライセンスサーバーの IP アドレスまたはホスト名を指定し、そのサーバーからライセンスをチェックアウトします。

フリートインスタンス間で環境変数を使用するには、この手順のステップを行います。

Note

次の手順は Windows フリートのみに適用されます。

内容

- [システム環境変数の変更](#)
- [ユーザー環境変数の変更](#)
- [範囲を制限して環境変数を作成する](#)

Note

AppStream 2.0 で Active Directory およびグループポリシーを使用している場合は、環境変数でグループポリシーが使用できるように、ストリーミングインスタンスが Active Directory ドメインに結合されていることを確認します。グループポリシーの [環境変数] の設定項目の設定については、Microsoft のドキュメントの「[環境変数の項目を設定する](#)」を参照してください。

システム環境変数の変更

フリートインスタンス全体でシステム環境変数を変更するには、以下の手順を実行します。

Image Builder のシステム環境変数を変更するには

この手順は、システム環境変数にのみ適用されます。ユーザー環境変数には適用されません。フリートインスタンス間で永続化するユーザー環境変数を変更するには、次の手順のステップを実行します。

1. システム環境変数を変更する Image Builder に接続し、ローカル管理者権限を持つアカウントでサインインします。そのためには、次のいずれかを実行します。
 - [AppStream 2.0 コンソールを使用する](#) (ウェブ接続のみ)
 - [ストリーミング URL を作成する](#) (ウェブまたは AppStream 2.0 クライアント接続用)

Note

接続先の Image Builder が Active Directory ドメインに参加していて、組織でスマートカードサインインが必要な場合は、ストリーミング URL を作成し、接続に AppStream 2.0 クライアントを使用する必要があります。スマートカードサインインの詳細については、「[スマートカード](#)」をご参照ください。

2. Windows の [スタート] ボタンを選択して、[コンピュータ] のコンテキスト (右クリック) メニューを開き、[プロパティ] を選択します。
3. ナビゲーションペインで [Advanced system settings] (システムの詳細設定) を選択します。
4. [システム変数] で、フリートインスタンス間で永続化する環境変数を変更し、[OK] を選択します。
5. Image Builder デスクトップから、Image Assistant を開きます。
6. Image Assistant で、必要な手順に従って、イメージの作成を完了します。詳細については、「[チュートリアル: AppStream 2.0 コンソールを使用してカスタム AppStream 2.0 イメージを作成する](#)」を参照してください。

システム環境変数の変更は、フリートインスタンス間で永続化され、該当するインスタンスから起動されるストリーミングセッションに適用されます。

Note

AWS CLI 認証情報をシステム環境変数として設定すると、AppStream 2.0 がイメージを作成できなくなる可能性があります。

ユーザー環境変数の変更

フリートインスタンス全体でユーザー環境変数を変更するには、以下の手順を実行します。

ユーザー環境変数を変更するには

1. システム環境変数を変更する Image Builder に接続し、[Template User (テンプレートユーザー)] としてサインインします。そのためには、次のいずれかを実行します。
 - [AppStream 2.0 コンソールを使用する](#) (ウェブ接続のみ)
 - [ストリーミング URL を作成する](#) (ウェブまたは AppStream 2.0 クライアント接続用)

Note

接続先の Image Builder が Active Directory ドメインに参加していて、組織でスマートカードサインインが必要な場合は、ストリーミング URL を作成し、接続に AppStream 2.0 クライアントを使用する必要があります。スマートカードサインインの詳細については、「[スマートカード](#)」をご参照ください。

[Template User (テンプレートユーザー)] では、ユーザー向けのデフォルトアプリケーションと Windows の設定を作成できます。詳細については、[Amazon AppStream 2.0 でのデフォルトのアプリケーションおよび Windows 設定とアプリケーション起動パフォーマンス](#)の「Creating Default Application and Windows Settings for Your AppStream 2.0 Users」を参照してください。

2. Image Builder で、Windows の [スタート] ボタンをクリックし、[コントロールパネル]、[ユーザー アカウント] の順に選択します。
3. [ユーザー アカウント] を再度選択します。左のナビゲーションペインで、[環境変数の変更] を選択します。
4. [DefaultProfileUser] の [ユーザー環境変数] で、必要に応じてユーザー環境変数を設定または作成し、[OK] を選択します。
5. これにより現在のセッションが切断され、ログインメニューが開きます。次のいずれかを実行して、Image Builder にログインします。
 - Image Builder が Active Directory ドメインに参加していない場合、[Local User] タブで、[Administrator] を選択します。

- Image Builder が Active Directory ドメインに参加している場合、[Directory User] タブを選択し、Image Builder でローカル監理者権限があるドメインユーザーとしてログインします。
6. Image Builder デスクトップから、Image Assistant を開きます。
 7. Image Assistant で、必要な手順に従って、イメージの作成を完了します。詳細については、「[チュートリアル: AppStream 2.0 コンソールを使用してカスタム AppStream 2.0 イメージを作成する](#)」を参照してください。

範囲を制限して環境変数を作成する

スクリプトから離れてスポンするプロセスに範囲が制限されている環境変数を作成するには、次のステップに従います。このアプローチは、同じ環境変数をアプリケーションごとに異なる値で使用する必要がある場合に便利です。たとえば、2 つの異なるアプリケーションでいずれも環境変数「LIC_SERVER」を使用しているが、使用する「LIC_SERVER」の値は異なる場合です。

範囲を制限して環境変数を作成するには

1. 範囲が制限された環境変数を作成する Image Builder に接続し、ローカル管理者権限を持つアカウントでサインインします。そのためには、次のいずれかを実行します。
 - [AppStream 2.0 コンソールを使用する](#) (ウェブ接続のみ)
 - [ストリーミング URL を作成する](#) (ウェブまたは AppStream 2.0 クライアント接続用)

Note

接続先の Image Builder が Active Directory ドメインに参加していて、組織でスマートカードサインインが必要な場合は、ストリーミング URL を作成し、接続に AppStream 2.0 クライアントを使用する必要があります。スマートカードサインインの詳細については、「[スマートカード](#)」をご参照ください。

2. スクリプト用に子フォルダを C:\ ドライブに作成します (例: C:\Scripts)。
3. メモ帳を開き、新しいスクリプトを作成して、次の行を入力します。

```
set ##=#
```

```
start " " "C:\path\to\application.exe"
```

コードの説明は以下のとおりです。

##は、使用する変数名

#は、一定の変数名の値

Note

アプリケーションのパスにスペースを含む場合は、文字列全体を引用符でカプセル化する必要があります。以下はその例です。

```
start " " "C:\Program Files\application.exe"
```

4. [File] (ファイル)、[Save] (保存) の順に選択します。ファイルに名前を付け、拡張子を .bat にして C:\Scripts に保存します。たとえば、ファイルの名前を LaunchApp.bat とします。
5. 必要に応じて、ステップ 4 および 5 を繰り返し、独自の環境変数および値が必要な追加アプリケーションごとにスクリプトを作成します。
6. Image Builder デスクトップから、Image Assistant を起動します。
7. [Add App] を選択して、C:\Scripts を参照し、ステップ 5 で作成したスクリプトのいずれかを選択します。開く をクリックします。
8. [App Launch Settings] ダイアログボックスで、必要に応じて設定をそのまま使用するか、変更します。完了したら、[Save] を選択します。
9. 複数のスクリプトを作成した場合は、スクリプトごとにステップ 8 と 9 を繰り返します。
10. Image Assistant で、必要な手順に従って、イメージの作成を完了します。詳細については、[「チュートリアル: AppStream 2.0 コンソールを使用してカスタム AppStream 2.0 イメージを作成する」](#)を参照してください。

スクリプトから実行されるプロセスで、環境変数と特定の値を使用できるようになりました。その他のプロセスで、この変数および値にアクセスすることはできません。

Amazon AppStream 2.0 でユーザー向けにデフォルトファイルの関連付けを設定する

アプリケーションファイルの拡張子の関連付けは、ユーザー単位で設定されるため、AppStream 2.0 ストリーミングセッションを起動するすべてのユーザーに自動的に適用されることはありません。たとえば、Image Builder の .pdf ファイルを開くデフォルトのアプリケーションとして Adobe Reader を設定しても、この変更はユーザーには適用されません。

Note

次の手順は Windows フリートにのみ適用されます。

Note

以下の手順は、Active Directory ドメインに参加している Image Builder で実行する必要があります。さらに、フリートは Active Directory ドメインに参加している必要があります。それ以外の場合、設定したデフォルトのファイルの関連付けは適用されません。

ユーザー向けにデフォルトファイルの関連付けを設定するには

1. デフォルトファイルの関連付けを設定する Image Builder に接続し、Image Builder に対するローカル管理者権限を持つドメインアカウントでサインインします。そのためには、次のいずれかを実行します。
 - [AppStream 2.0 コンソールを使用する](#) (ウェブ接続のみ)
 - [ストリーミング URL を作成する](#) (ウェブまたは AppStream 2.0 クライアント接続用)

Note

組織でスマートカードサインインが必要な場合は、ストリーミング URL を作成し、接続に AppStream 2.0 クライアントを使用する必要があります。スマートカードサインインの詳細については、「[スマートカード](#)」をご参照ください。

2. 必要に応じて、デフォルトのファイルの関連付けを設定します。
3. 管理者として Windows コマンドプロンプトを開きます。
4. コマンドプロンプトで次のコマンドを入力して、Image Builder のファイル関連付けを XML ファイルでエクスポートし、ENTER キーを押します。

```
dism.exe /online /export-DefaultAppAssociations:c:\  
\default_associations.xml
```

32 ビットバージョンの DISM で、実行中の 64 ビットのオペレーティングシステムを使用できないというエラーメッセージが表示される場合は、コマンドプロンプトウィンドウを閉じま

す。File Explorer を開いて、C:\Windows\System32 を参照し、cmd.exe を右クリックしたら、[管理者として実行] を選択し、再度コマンドを実行します。

- ローカルグループポリシーエディタまたは GPMC を使用して、デフォルトの関連付け設定ファイルを設定できます。

- ローカルグループポリシーエディタ:

Image Builder で、管理者としてコマンドプロンプトを開き、gpedit.msc と入力し、ENTER キーを押します。

コンソールツリーで、[Computer Configuration] (コンピュータの構成) を [Administrative Templates] (管理用テンプレート)、[Windows Components] (Windows コンポーネント) の順に展開し、[File Explorer] を選択します。

- GPMC:

ディレクトリまたはドメインコントローラーで、管理者としてコマンドプロンプトを開き、gpmc.msc と入力し、ENTER キーを押します。

左のコンソールツリーで、新しい GPO を作成する OU、または既存の GPO を使用する OU を選択して、以下のいずれかの操作を行います。

- コンテキスト (右クリック) メニューを開き、[Create a GPO in this domain, Link it here] を選択して、新しい GPO を作成します。[Name] に、この GPO のわかりやすい名前を入力します。

- 既存の GPO を選択します。

GPO のコンテキストメニューを開き、[編集] を選択します。

[User Configuration (ユーザーの構成)] を [ポリシー]、[Administrative Templates (管理用テンプレート)]、[Windows Components (Windows コンポーネント)] の順に展開し、[File Explorer (ファイルエクスプローラー)] を選択します。

- [デフォルトの関連付け設定ファイルを設定] をダブルクリックします。
- [Set a default associations configuration file properties (デフォルトの関連付け設定ファイルプロパティを設定)] ダイアログボックスの [有効] を選択し、次のいずれかを実行します。
 - ローカルグループポリシーエディタを使用している場合は、パスを入力します: c:\default_associations.xml。
 - GPMC を使用している場合は、ネットワークパスを入力します。例えば、\\networkshare\default_associations.xml と指定します。

8. [適用]、[OK] の順に選択します。
9. ローカルグループポリシーエディタまたは GPMC を終了します。
10. Image Builder デスクトップから、Image Assistant を開きます。
11. Image Assistant で、必要な手順に従って、イメージの作成を完了します。詳細については、[「チュートリアル: AppStream 2.0 コンソールを使用してカスタム AppStream 2.0 イメージを作成する」](#)を参照してください。

設定したファイルの関連付けは、フリートインスタンスと、そのインスタンスから起動されるユーザーストリーミングセッションに適用されます。

Amazon AppStream 2.0 で Internet Explorer 拡張セキュリティ設定を無効にする

Internet Explorer セキュリティ強化の構成 (ESC) では、サーバーと Internet Explorer のインターネットへの公開が制限されるように設定されます。ただし、この設定は、AppStream 2.0 エンドユーザーエクスペリエンスに影響する可能性があります。AppStream 2.0 ストリーミングセッションに接続されているユーザーでは、以下の場合にウェブサイトが想定どおりに表示されない、または実行されないことがあります。

- ユーザーのストリーミングセッションを起動したフリートインスタンスで Internet Explorer ESC が有効になっている
- ユーザーが、ストリーミングセッションの間に Internet Explorer を実行
- アプリケーションが Internet Explorer を使用してデータをロードする

Note

次の手順は Windows フリートにのみ適用されます。

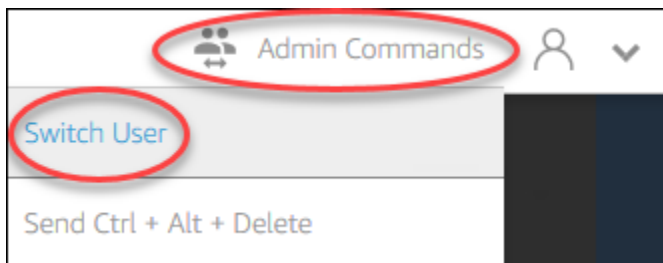
Internet Explorer セキュリティ強化の構成を無効にするには

1. Internet Explorer ESC を無効にする Image Builder に接続し、ローカル管理者権限を持つアカウントでサインインします。そのためには、次のいずれかを実行します。
 - [AppStream 2.0 コンソールを使用する](#) (ウェブ接続のみ)
 - [ストリーミング URL を作成する](#) (ウェブまたは AppStream 2.0 クライアント接続用)

Note

接続先の Image Builder が Active Directory ドメインに参加していて、組織でスマートカードサインインが必要な場合は、ストリーミング URL を作成し、接続に AppStream 2.0 クライアントを使用する必要があります。スマートカードサインインの詳細については、「[スマートカード](#)」をご参照ください。

2. Image Builder で Internet Explorer ESC を無効にするには、次の操作を行います。
 - a. サーバーマネージャーを開きます。Windows の [スタート] ボタンをクリックし、[サーバーマネージャー] を選択します。
 - b. 左のナビゲーションペインの [ローカル サーバー] を選択します。
 - c. 右側のプロパティペインで、[IE Enhanced Security Configuration] (IE セキュリティ強化の構成) の横にある [On] (有効) リンクを選択します。
 - d. [Internet Explorer セキュリティ強化の構成] ダイアログボックスで、[Administrators] と [Users] の [オフ] オプションを選択し、[OK] を選択します。
3. Image Builder デスクトップの右上領域で、[Admin Commands]、[Switch User] の順に選択します。



4. これにより現在のセッションが切断され、ログインメニューが開きます。次のいずれかを実行して、Image Builder にログインします。
 - Image Builder が Active Directory ドメインに参加していない場合、[Local User] タブで、[Template User] を選択します。
 - Image Builder が Active Directory ドメインに参加している場合は、[Directory User] タブを選択し、Image Builder でローカル管理者権限がないドメインユーザーとしてログインします。
5. Internet Explorer を開いて次の手順を実行して設定をリセットします。
 - a. Internet Explorer ブラウザウィンドウの右上で、ツールアイコンを選択し、[インターネット オプション] を選択します。

- b. [詳細設定] タブを選択し、[リセット] を選択します。
 - c. 選択の確認を求められたら、[リセット] を選択します。
 - d. [Internet Explorer の設定をリセット] メッセージが表示されたら、[閉じる] を選択します。
6. Image Builder を再起動します。
 7. [Admin Commands]、[Switch User] を選択し、次のいずれかの操作を行います。
 - Image Builder が Active Directory ドメインに参加していない場合、[Local User] タブで、[Administrator] を選択します。
 - Image Builder が Active Directory ドメインに参加している場合、[ディレクトリユーザー] タブを選択し、ステップ 4 で使用したのと同じドメインアカウントでログインします。
 8. Image Builder デスクトップから、Image Assistant を開きます。
 9. [Step 2. Configure Apps] (ステップ 2。アプリを設定する) で、[Save settings] (設定の保存) を選択します。
 10. Image Assistant で、必要な手順に従って、イメージの作成を完了します。詳細については、[「チュートリアル: AppStream 2.0 コンソールを使用してカスタム AppStream 2.0 イメージを作成する」](#)を参照してください。

Amazon AppStream 2.0 でユーザーのストリーミングセッション用に Internet Explorer のデフォルトホームページを変更する

グループポリシーを使用して、ユーザーのストリーミングセッションに対する Internet Explorer のデフォルトホームページを変更できます。また、使用環境にグループポリシーがないか、グループポリシーを使用したくない場合、代わりに AppStream 2.0 Template User アカウントを使用できます。

Note

次の手順は Windows フリートにのみ適用されます。

内容

- [グループポリシーを使用して Internet Explorer のデフォルトホームページを変更する](#)
- [AppStream 2.0 Template User アカウントを使用して、Internet Explorer のデフォルトホームページを変更する](#)

グループポリシーを使用して Internet Explorer のデフォルトホームページを変更する

Active Directory 環境で、グループ ポリシー管理 (GPMC) MMC スナップインを使用して、デフォルトホームページをユーザーが変更できないように設定します。Active Directory が使用環境に存在しない場合は、ローカルグループポリシーエディタを使用してこのタスクを実行します。ユーザーが変更できるホームページを設定するには、GPMC を使用する必要があります。

GPMC を使用するには、最初に次のことを行います。

- ドメインに参加済みのコンピュータまたは EC2 インスタンスにアクセスします。
- GPMC をインストールします。詳細については、Microsoft ドキュメントの「[Installing or Removing Remote Server Administration Tools for Windows 7](#)」を参照してください。
- アクセス許可を持つドメインとしてログインし、GPO を作成します。GPO を適切な組織ユニット (OU) にリンクします。

グループポリシーの管理用テンプレートを使用して Internet Explorer のデフォルトホームページを変更するには

ユーザーが変更できないデフォルトホームページを設定するには、グループポリシーの管理用テンプレートを使用できます。管理用テンプレートの詳細については、Microsoft のドキュメントの「[管理用テンプレートのポリシー設定の編集](#)」を参照してください。

1. AppStream 2.0 コンソール (<https://console.aws.amazon.com/appstream2>) を開きます。
2. お客様の環境で Active Directory を使用していない場合は、ローカルグループポリシーエディタを開きます。Active Directory を使用している場合は、GPMC を開きます。[スクリプト (ログオン/ログオフ)] ポリシー設定を見つけます。

- ローカルグループポリシーエディタ:

Image Builder で、管理者としてコマンドプロンプトを開き、gpedit.msc と入力し、ENTER キーを押します。

[User Configuration] (ユーザーの構成) を [Administrative Templates] (管理用テンプレート)、[Windows Components] (Windows コンポーネント) の順に展開し、[Internet Explorer] を選択します。

- GPMC:

ディレクトリまたはドメインコントローラーで、管理者としてコマンドプロンプトを開き、gpmc.msc と入力し、ENTER キーを押します。

左のコンソールツリーで、新しい GPO を作成する OU、または既存の GPO を使用する OU を選択して、以下のいずれかの操作を行います。

- コンテキスト (右クリック) メニューを開き、[Create a GPO in this domain, Link it here] を選択して、新しい GPO を作成します。[Name] に、この GPO のわかりやすい名前を入力します。
- 既存の GPO を選択します。

GPO のコンテキストメニューを開き、[編集] を選択します。

[User Configuration] (ユーザーの構成) を [ポリシー]、[Administrative Templates] (管理用テンプレート)、[Windows Components] (Windows コンポーネント) の順に展開し、[Internet Explorer] を選択します。

3. [ホームページ設定の変更の無効化] をダブルクリックして [有効] を選択し、[ホームページ] に URL を入力します。
4. [適用]、[OK] の順に選択します。
5. ローカルグループポリシーエディタまたは GPMC を終了します。

グループポリシー設定を使用して Internet Explorer のデフォルトホームページを変更するには

ユーザーが変更できないデフォルトホームページを設定するには、グループポリシー設定を使用します。グループポリシーの設定の詳細については、Microsoft のドキュメントの「[レジストリ項目の設定](#)」および「[グループポリシー設定の入門ガイド](#)」を参照してください。

1. ディレクトリまたはドメインコントローラーで、管理者としてコマンドプロンプトを開き、`gpmc.msc` と入力し、ENTER キーを押します。
2. 左のコンソールツリーで、新しい GPO を作成する OU、または既存の GPO を使用する OU を選択して、以下のいずれかの操作を行います。
 - コンテキスト (右クリック) メニューを開き、[Create a GPO in this domain, Link it here] を選択して、新しい GPO を作成します。[Name] に、この GPO のわかりやすい名前を入力します。
 - 既存の GPO を選択します。
3. GPO のコンテキストメニューを開き、[編集] を選択します。
4. [User Configuration] (ユーザーの構成) の [設定] を展開し、[Windows Settings] (Windows 設定) を選択します。

5. [レジストリ] のコンテキスト (右クリック) メニューを開き、[新規]、[レジストリ項目] の順に選択します。
6. [新しいレジストリのプロパティ] ダイアログボックスで、グループポリシーの次のレジストリ設定を指定します。
 - [Action] で、[Update] を選択します。
 - [Hive] で、[HKEY_CURRENT_USER] を選択します。
 - [キーパス] で、HKEY_CURRENT_USER\SOFTWARE\Microsoft\Internet Explorer\Main を参照して選択します。
 - [値の名前] に、[Start Page] (開始ページ) を入力します。
 - [Value Data] (値データ) に、ホームページの URL を入力します。
7. [一般] タブで、[Apply Once] (一度のみ適用)、[Do not Re-Apply] (再適用しない) の順で選択します。

 Note

Internet Explorer のブラウザ設定で、ユーザーが [デフォルトを使用] ボタンを選択して、デフォルトホームページを企業ホームページにリセットできるように、Default_Page_URL の値を設定することもできます。[Apply Once] (一度のみ適用) や [Do not Re-Apply] (再適用しない) を選択する必要はありません。

8. [OK] を選択して、GPMC を閉じます。

AppStream 2.0 Template User アカウントを使用して、Internet Explorer のデフォルトホームページを変更する

Template User アカウントを使用して、Internet Explorer のデフォルトホームページを変更するには、以下の手順を実行します。

Template User アカウントを使用して、Internet Explorer のデフォルトホームページを変更するには

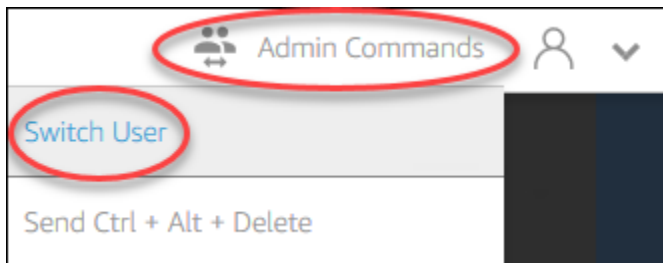
1. デフォルトの Internet Explorer ホームページを変更する Image Builder に接続し、[Template User (テンプレートユーザー)] アカウントでサインインします。そのためには、次のいずれかを実行します。
 - [AppStream 2.0 コンソールを使用する](#) (ウェブ接続のみ)
 - [ストリーミング URL を作成する](#) (ウェブまたは AppStream 2.0 クライアント接続用)

Note

接続先の Image Builder が Active Directory ドメインに参加していて、組織でスマートカードサインインが必要な場合は、ストリーミング URL を作成し、接続に AppStream 2.0 クライアントを使用する必要があります。スマートカードサインインの詳細については、「[スマートカード](#)」をご参照ください。

[Template User (テンプレートユーザー)] では、ユーザー向けのデフォルトアプリケーションと Windows の設定を作成できます。詳細については、[Amazon AppStream 2.0 でのデフォルトのアプリケーションおよび Windows 設定とアプリケーション起動パフォーマンス](#)の「Creating Default Application and Windows Settings for Your AppStream 2.0 Users」を参照してください。

2. Internet Explorer を開き、デフォルトのホームページを変更するために必要な手順を完了します。
3. Image Builder デスクトップの右上領域で、[Admin Commands]、[Switch User] の順に選択します。



4. これにより現在のセッションが切断され、ログインメニューが開きます。次のいずれかを実行して、Image Builder にログインします。
 - Image Builder が Active Directory ドメインに参加していない場合、[Local User] タブで、[Administrator] を選択します。
 - Image Builder が Active Directory ドメインに参加している場合、[Directory User] タブを選択し、Image Builder でローカル監理者権限があるドメインユーザーとしてログインします。
5. Image Builder デスクトップから、Image Assistant を開きます。
6. Image Assistant で、必要な手順に従って、イメージの作成を完了します。詳細については、「[チュートリアル: AppStream 2.0 コンソールを使用してカスタム AppStream 2.0 イメージを作成する](#)」を参照してください。

Amazon AppStream 2.0 フリートのユーザーおよびインスタンスのメタデータ

AppStream 2.0 フリートインスタンスには、Windows 環境変数を介して利用可能なユーザーおよびインスタンスのメタデータがあります。アプリケーションおよびスクリプトで次の環境変数を使用して、フリートインスタンスの詳細に基づいて環境を変更できます。

環境変数	Context	説明
AppStream_Stack_Name	ユーザー	ストリーミングセッションが開始されたスタックの名前。
AppStream_User_Access_Mode	ユーザー	ストリームへのユーザーアクセスを管理するために使用されるアクセスモード。指定できる値は custom 、 userpool 、または saml です。
AppStream_Session_Reservation_DateTime	ユーザー	ユーザーのストリーミングセッションが開始された日時。
AppStream_User_Name	ユーザー	ユーザーに関連付けられたユーザーの名前。
AppStream_Session_ID	ユーザー	ユーザーのストリーミングセッションのセッション識別子。
APPSTREAM_SESSION_CONTEXT	マシン	セッションの開始時にストリーミングアプリケーションに渡されるパラメータが含まれます。詳細については、「 Amazon AppStream 2.0 のセッションコンテキスト 」を参照してください。  Note この環境変数は、最初のアプリケーションの起動後にのみ使用できます。

環境変数	Context	説明
AppStream_Image_Arn	マシン	ストリーミングインスタンスを作成するために使用されたイメージの ARN。
AppStream_Instance_Type	マシン	ストリーミングインスタンスのタイプ。例えば、 stream.standard.medium と指定します。
AppStream_Resource_Type	マシン	AppStream 2.0 リソースのタイプ。この値は fleet または image-builder のどちらかです。
AppStream_Resource_Name	マシン	フリートの名前。

Linux フリートインスタンスでは、以下の profile.d スクリプトを通じてこれらの環境変数がエクスポートされます。

- /etc/profile.d/appstream_user_vars.sh 内のユーザー環境変数
- /etc/profile.d/appstream_system_vars.sh 内のシステム環境変数

環境変数にアクセスするには、アプリケーションでこれらのファイルを明示的にソースする必要があります。

Amazon AppStream 2.0 フリートを更新する

既存の AppStream 2.0 フリートを更新することが可能です。

新しい Amazon AppStream 2.0 イメージを作成するときは、常時オンおよびオンデマンドフリートを更新して、ユーザーが新しいイメージのアプリケーションとデータを利用できるようにする必要があります。小規模な更新がある場合 (たとえば、アプリケーションやオペレーティングシステムのパッチ適用)、実行中のフリートを更新することができます。新しいストリーミングインスタンスを作成すると、それらは更新されたイメージから作成されます。実行中のフリートでイメージを変更しても、ストリーミングセッションのアクティブなユーザーは中断されません。使用されていないストリーミングインスタンスを定期的に置き換えられ、ユーザーが接続しているストリーミングインスタンスは、ストリーミングセッションを終了した後で終了されます。

フリートが実行中または停止状態のとき、同じオペレーティングシステムを実行する新しいイメージでフリートを更新できます。ただし、別のオペレーティングシステムで実行される新しいイメージでフリートを更新できるのは、そのフリートが停止状態にある場合のみです。

 Note

AppStream 2.0 がユーザーに表示するアプリケーションカタログは、フリートに関連付けられている現在のイメージに基づいています。更新されたイメージに古いイメージで指定されていないアプリケーションが含まれている場合、ユーザーが古いイメージに基づいているインスタンスからストリーミングすると、アプリケーションが起動しないことがあります。

トピック

- [新しいイメージで Amazon AppStream 2.0 フリートを更新する](#)
- [Amazon AppStream 2.0 のElastic フリートに関連付けられたアプリケーションを管理する](#)

新しいイメージで Amazon AppStream 2.0 フリートを更新する

オペレーティングシステムの更新を適用、または新しいアプリケーションをユーザーが利用できるようにするには、これらの変更を含む新しいイメージを作成します。次に、新しいイメージでフリートを更新します。

AppStream 2.0 フリートを新しいイメージで更新するには

1. 使用する Image Builder に接続し、Image Builder に対するローカル管理者権限を持つアカウントでサインインします。そのためには、次のいずれかを実行します。
 - [AppStream 2.0 コンソールを使用する](#) (ウェブ接続のみ)
 - [ストリーミング URL を作成する](#) (ウェブまたは AppStream 2.0 クライアント接続用)

 Note

組織でスマートカードサインインが必要な場合は、ストリーミング URL を作成し、接続に AppStream 2.0 クライアントを使用する必要があります。スマートカードサインインの詳細については、「[スマートカード](#)」をご参照ください。

2. 必要に応じて、次のいずれかまたは両方を実行します。

- オペレーティングシステムの更新をインストールします。
- アプリケーションをインストールします。

アプリケーションで Windows オペレーティングシステムの再起動が必要な場合は、再起動します。オペレーティングシステムが再起動する前に、Image Builder から切断されます。再起動が完了したら、Image Builder に再接続し、アプリケーションのインストールを完了します。

3. Image Builder デスクトップから、Image Assistant を開きます。
4. Image Assistant で、必要な手順に従って、イメージの作成を完了します。詳細については、「[チュートリアル: AppStream 2.0 コンソールを使用してカスタム AppStream 2.0 イメージを作成する](#)」を参照してください。

イメージのステータスが [使用可能] に変わったら、新しいイメージでフリートを更新できます。

5. 左のナビゲーションペインの [フリート] を選択します。
6. 新しいイメージで更新するフリートを選択します。
7. [Fleet Details] タブで、[Edit] を選択します。
8. [フリートの編集] ダイアログボックスで、使用可能なイメージのリストが [名前] リストに表示されます。リストから新しいイメージを選択します。
9. [Update] (更新) を選択します。

Amazon AppStream 2.0 のElastic フリートに関連付けられたアプリケーションを管理する

Elastic フリートに対するアプリケーションの関連付けと関連付けの解除は、いつでも実行することができます。Elastic フリートに関連付けられたアプリケーションへの変更は、フリートから現在ストリーミングしているユーザーに表示されますが、有効にはならない場合があります。例えば、フリートからアプリケーションの関連付けを解除すると、そのアプリケーションがアプリケーションカタログから削除されますが、仮想ハードディスクは既存のストリーミングセッションにマウントされたままになります。

Elastic フリートに関連付けられたアプリケーションを管理する

1. [AppStream 2.0 コンソール](#)を開きます。

2. 左のナビゲーションペインで [フリート] を選択してから、フリートの名前を選択するか、フリートのラジオボタンをオンにして、[詳細を表示する] をクリックします。
3. フリートに新しいアプリケーションを関連付けるには、[割り当てられたアプリケーション] で [関連付ける] を選択し、関連付けられるアプリケーションを選択してから、[関連付ける] を選択します。
4. フリートから既存のアプリケーションの関連付けを解除するには、関連付けを解除するアプリケーションを選択し、[関連付けを解除する] を選択してから、[関連付けを解除する] を選択することで、選択したアプリケーションの関連付けを解除することを確認します。

Amazon AppStream 2.0 向け Fleet Auto Scaling

Fleet Auto Scaling では、AppStream 2.0 の常時オンまたはオンデマンドフリートのサイズを自動的に変更して、利用可能なインスタンスの提供をユーザーの需要に適合させることができます。フリートのサイズによって、同時にストリーミングできるユーザーの数が決まります。マルチセッションフリートでは、複数のユーザーが単一のインスタンスを使用できます。非マルチセッションフリートでは、ユーザーセッションごとに 1 つのインスタンスが必要です。フリートの容量は、インスタンス数 (シングルセッションフリートの場合) とユーザーセッション数 (マルチセッションフリートの場合) で指定できます。フリート設定と自動スケーリングポリシーに基づいて、必要な数のインスタンスが利用可能になります。さまざまな使用状況メトリクスに基づいてフリートのサイズを自動的に調整するスケーリングポリシーを定義し、使用可能なインスタンスの数を最適化してユーザーの需要に合わせることができます。自動スケーリングを無効にして、固定されたフリートサイズで実行することもできます。

Note

Elastic フリートの容量は AppStream 2.0 によって自動的に管理されます。Elastic フリート用に利用できるフリートストリーミングインスタンスの数を管理するためにオートスケーリングルールを作成する必要はありません。

Note

AppStream 2.0 フリートスケーリングの計画を策定する際には、ネットワーク設定が要件を満たしていることを確認してください。

Fleet Auto Scaling を使用する前に、Application Auto Scaling で Amazon CloudWatch アラームと AppStream 2.0 フリートにアクセスするためのアクセス許可が必要です。詳細については、[AWS 管理ポリシーとリンクされたロールを使用して AppStream 2.0 リソースへの管理者アクセスを管理する](#)および[IAM ポリシーを使用して Application Auto Scaling への管理者アクセスを管理する](#)を参照してください。

 Note

スケーリングを使用する場合は、Application Auto Scaling API を使用します。AppStream 2.0 で Fleet Auto Scaling が正しく機能するためには、Application Auto Scaling は AppStream 2.0 フリートの説明と更新、および Amazon CloudWatch アラームを記述するアクセス許可、およびお客様に代わってフリートの容量を変更するアクセス許可を必要とします。詳細については、[AppStream 2.0、Application Auto Scaling、および AWS Certificate Manager Private CA に必要なロール](#)および[IAM ポリシーを使用して Application Auto Scaling への管理者アクセスを管理する](#)を参照してください。

以下のトピックでは、AppStream 2.0 Fleet Auto Scaling の理解と使用に役立つ情報を示します。

内容

- [Amazon AppStream 2.0 のスケーリングの概念](#)
- [Amazon AppStream 2.0 コンソールを使用したフリートスケーリングの管理](#)
- [Amazon AppStream 2.0 の AWS CLI を使用したフリートスケーリングの管理](#)
- [自動スケーリング Amazon AppStream 2.0 の追加リソース](#)

Amazon AppStream 2.0 のスケーリングの概念

AppStream 2.0 スケーリングは、Application Auto Scaling によって提供されます。詳細については、[Application Auto Scaling API リファレンス](#)を参照してください。

AppStream 2.0 Fleet Auto Scaling の操作に関するステップバイステップガイドンスについては、AWS コンピューティングブログ [Scaling Your Desktop Application Streams with Amazon AppStream 2.0](#) を参照してください。

Fleet Auto Scaling を効果的に使用するには、以下の用語と概念を理解する必要があります。

マルチセッションとシングルセッション

シングルセッションのシナリオでは、ユーザーセッションごとに専有インスタンスがあります。マルチセッションモードでは、1つのインスタンスで複数のユーザーセッションをプロビジョニングできます。フリート容量と自動スケーリングポリシーは、ユーザーセッション数で設定する必要があります。これにより、サービスは必要な数のインスタンスを計算して起動します。

フリートの最小容量/最小ユーザーセッション数

インスタンス (シングルセッションフリートの場合) またはユーザーセッション (マルチセッションフリートの場合) の最小数。インスタンス数 (シングルセッションフリートの場合) またはユーザーセッション数 (マルチセッションフリートの場合) がこの値を下回ることはできません。また、スケーリングポリシーがこの値未満にフリートをスケールすることはありません。例えば、シングルセッションシナリオの場合、フリートの最小容量を 2 に設定した場合、フリートのインスタンス数が 2 を下回ることはありません。同様に、マルチセッションシナリオの場合、インスタンスの最大セッション数を 5 に設定し、フリートの最小容量を 12 に設定すると、フリートのインスタンス数が切り上げ数 $(12/5) = 3$ 未満になることはありません。

フリートの最大容量/最大ユーザーセッション数

インスタンス (シングルセッションフリートの場合) またはユーザーセッション (マルチセッションフリートの場合) の最大数。インスタンス数 (シングルセッションフリートの場合) またはユーザーセッション数 (マルチセッションフリートの場合) がこの値を上回ることはありません。また、スケーリングポリシーがこの値を超えてフリートをスケールすることはありません。例えば、シングルセッションシナリオの場合、フリートの最大容量を 10 に設定すると、フリートのインスタンス数が 10 を上回ることはありません。同様に、マルチセッションシナリオの場合、インスタンスの最大セッション数を 5 に設定し、フリートの最大容量を 52 に設定すると、フリートのインスタンス数が切り上げ数 $(52/5) = 11$ を超えることはありません。

希望する容量

実行中または保留中のインスタンス (シングルセッションフリートの場合) またはユーザーセッション (マルチセッションフリートの場合) の合計数。この値は、フリートが一定の状態をサポートできる同時ストリーミングセッションの合計数を表します。[希望する容量] の値を設定するには、[フリートの詳細] を編集します。[Scaling Policies] (スケーリングポリシー) を使用する場合は、[Desired Capacity] (希望する容量) の値を手動で変更することはお勧めしません。

[Desired Capacity] (希望する容量) の値が [Minimum Capacity] (最低容量) 未満の値に設定されている状態でスケールアウトアクティビティがトリガーされた場合、Application Auto Scaling は [Desired Capacity] (希望する容量) を [Minimum Capacity] (最低容量) の値までスケールアップしてから、スケーリングポリシーに基づいたスケールアウトを必要に応じて続行します。ただし、

この場合は、すでに [最低容量] の値を下回っているため、スケールインアクティビティは [希望する容量] を調整しません。

[Desired Capacity] (希望する容量) が [Maximum Capacity] (最大容量) を超える値に設定されている状態でスケールインアクティビティがトリガーされた場合、Application Auto Scaling は [Desired Capacity] (希望する容量) を [Maximum Capacity] (最大容量) の値までスケールダウンしてから、スケーリングポリシーに基づいたスケールインを必要に応じて続行します。ただし、この場合は、すでに [最大容量] の値を上回っているため、スケールアウトアクティビティは [希望する容量] を調整しません。

スケーリングポリシーのアクション

[Scaling Policy Condition] が満たされると、フリートでスケーリングポリシーがアクションを実行します。[% 容量] または [インスタンス数] (シングルセッションフリートの場合) と [ユーザーセッション数] (マルチセッションフリートの場合) のいずれかに基づいてアクションを選択できます。例えば、[現在の容量] が 4 の場合に [スケーリングポリシーアクション] が「容量を 25% 追加」に設定されていると、[スケーリングポリシー条件] が満たされるときに [希望する容量] は 5 に設定されます。

スケーリングポリシーの条件

[Scaling Policy Action] で設定されたアクションをトリガーする条件。この条件は、スケーリングポリシーのメトリクス、比較演算子、しきい値を含みます。たとえば、フリートの使用率が 50% 以上のときにフリートをスケーリングする場合は、スケーリングポリシーの条件を「Capacity Utilization > 50%」にする必要があります。

スケーリングポリシーのメトリクス

お客様のスケーリングポリシーはこのメトリクスに基づいています。スケーリングポリシーには、次のメトリクスを使用できます。

容量使用率

フリートで使用されているインスタンスの割合。このメトリクスを使用すると、フリートの使用量に基づいてフリートをスケールできます。たとえば、[Scaling Policy Condition]: 「容量使用率 < 25%」の場合、[Scaling Policy Action]: 「25% の容量を削除」を実行します。

使用可能な容量

ユーザーが利用できるフリート内のインスタンス数 (シングルセッションフリートの場合) またはユーザーセッション数 (マルチセッションフリートの場合)。このメトリクスを使用して、ユーザーがストリーミングセッションを開始するための、使用可能なメモリ容量のバッファを

維持できます。例として、[スケーリングポリシー条件] が「使用可能な容量が 5 未満の場合」では、[スケーリングポリシーアクション]として「インスタンス (シングルセッションフリートの場合) またはユーザーセッション (マルチセッションフリートの場合) を 5 つ追加」を実行します。

容量不足エラー

容量不足により拒否されたセッションリクエストの数。このメトリクスを使用して、容量不足のためにストリーミングセッションを開始できないユーザーの新しいインスタンスをプロビジョニングできます。例えば、[スケーリングポリシー条件] が「容量不足エラー > 0 である場合」では、[スケーリングポリシーアクション]として「インスタンス (シングルセッションフリートの場合) またはユーザーセッション (マルチセッションフリートの場合) を 1 つ追加」を実行します。

Amazon AppStream 2.0 コンソールを使用したフリートスケーリングの管理

AppStream 2.0 コンソールを使用してフリートスケーリングを設定および管理できます。この操作は、フリートの作成中、またはいつでも [Fleets (フリート)] タブを使用して行うことができます。起動後に新しく作成されたフリートには 2 つのデフォルトのスケーリングポリシーが関連付けられます。これらのポリシーは、AppStream 2.0 コンソールの [Scaling Policies (スケーリングポリシー)] タブで編集できます。詳細については、「[Amazon AppStream 2.0 でフリートを作成する](#)」を参照してください。

ユーザーの環境はさまざまに異なるため、需要に応じてスケーリングを制御するようにスケーリングポリシーを定義します。一定数のユーザーが予想される場合やスケーリングを無効にする他の理由がある場合には、一定数のインスタンスやユーザーセッションを使用してフリートを設定できます。

コンソールを使用してフリートのスケーリングポリシーを設定するには

1. AppStream 2.0 コンソールを <https://console.aws.amazon.com/appstream2> で開きます。
2. ナビゲーションペインの [Fleets] を選択します。
3. スケーリングポリシーを設定するフリートを選択して、[Scaling Policies] を選択します。
4. 各値の横にある編集アイコンを選択して既存のポリシーを編集します。編集フィールドで目的の値を設定して、[Update] を選択します。ポリシーの変更は数分以内で有効になります。
5. [Add Policy] リンクを使用して新しいポリシーを追加 (作成) します。編集フィールドで目的の値を設定して、[Create] を選択します。数分以内に新しいポリシーが適用されます。

[Fleet Usage] タブを使用して、スケーリングポリシーの変更の影響をモニタリングできます。次の例は、5 人のユーザーがフリートに接続して切断する場合のスケーリングアクティビティの使用状況グラフです。この例では、フリートは次のスケーリングポリシーを使用します。

- 最小容量 = 10
- 最大容量 = 50
- スケールアウト = 容量使用率が 75% を超える場合は、5 つのインスタンス (シングルセッションフリートの場合) またはユーザーセッション (マルチセッションフリートの場合) を追加
- スケールイン = 容量使用率が 25% 未満の場合は、6 つのインスタンス (シングルセッションフリートの場合) またはユーザーセッション (マルチセッションフリートの場合) を削除

Note

上記のポリシーは、シングルセッションシナリオとマルチセッションシナリオの両方に適用されます。シングルセッションシナリオでは、スケールアウトイベント中に 5 つの新しいインスタンスが起動され、スケールダウンイベント中に 4 つのインスタンスが再利用されます。マルチセッションシナリオの場合、インスタンスあたりの最大セッション数が 4 では、スケールアウトイベントによって切り上げ数 (ユーザーセッションの追加数 5/インスタンスあたりの最大セッション数 4) = 2 のインスタンスの起動がトリガーされます。スケールインイベント中、サービスは切り上げ数 (ユーザーセッションの削除数 6/インスタンスあたりの最大セッション数 4) = 2 のインスタンスを再利用します。ユーザーセッションが実行中であるインスタンスは再利用されません。実行中のユーザーセッションがないインスタンスのみが再利用されます。

コンソールを使用してフリートを固定容量に設定するには

1. AppStream 2.0 コンソールを <https://console.aws.amazon.com/appstream2> で開きます。
2. ナビゲーションペインの [Fleets] を選択します。
3. フリートを選択します。
4. [Scaling Policies] で、フリートに関連付けられているすべてのポリシーを削除します。
5. [Fleet Details] で、[Desired Capacity] を設定するフリートを編集します。

固定されたフリートは [Desired Capacity] で指定した値に基づく一定の容量となります。固定されたフリートで常に希望する数のインスタンスが利用可能で、および、そのフリートのコストの課金を停止するには、フリートを停止する必要があることに注意してください。

Amazon AppStream 2.0 の AWS CLI を使用したフリートスケーリングの管理

AWS Command Line Interface (AWS CLI) を使用してフリートスケーリングを設定および管理できます。複数のスケーリングポリシーの設定やスケールインおよびスケールアウトのクールダウン時間の設定など、より高度な機能については、CLI AWS を使用します。スケーリングポリシーコマンドを実行する前に、フリートをスケーラブルなターゲットとして登録する必要があります。これを行うには、以下の [register-scalable-target](#) コマンドを使用します。

```
aws application-autoscaling register-scalable-target
  --service-namespace appstream \
  --resource-id fleet/fleetname \
  --scalable-dimension appstream:fleet:DesiredCapacity \
  --min-capacity 1 --max-capacity 5
```

例

- [例 1: 容量使用率に基づくスケーリングポリシーの適用](#)
- [例 2: 容量不足エラーに基づくスケーリングポリシーの適用](#)
- [例 3: 低容量使用率に基づくスケーリングポリシーの適用](#)
- [例 4: スケジュールに基づくフリートの容量の変更](#)
- [例 5: ターゲット追跡スケーリングポリシーの適用](#)

例 1: 容量使用率に基づくスケーリングポリシーの適用

この AWS CLI の例では、使用率 $\geq 75\%$ の場合にフリートを 25% スケールアウトするスケーリングポリシーを設定します。

次の [put-scaling-policy](#) コマンドは使用率ベースのスケーリングポリシーを定義します。

```
aws application-autoscaling put-scaling-policy --cli-input-json file://scale-out-utilization.json
```

scale-out-utilization.json ファイルの内容は以下のようになります。

```
{
  "PolicyName": "pollicyname",
  "ServiceNamespace": "appstream",
```

```

"ResourceId": "fleet/fleetname",
"ScalableDimension": "appstream:fleet:DesiredCapacity",
"PolicyType": "StepScaling",
"StepScalingPolicyConfiguration": {
  "AdjustmentType": "PercentChangeInCapacity",
  "StepAdjustments": [
    {
      "MetricIntervalLowerBound": 0,
      "ScalingAdjustment": 25
    }
  ],
  "Cooldown": 120
}
}

```

コマンドが成功した場合、一部の詳細はアカウントおよびリージョンで固有ですが、出力は次のようになります。この例では、ポリシー識別子は e3425d21-16f0-d701-89fb-12f98dac64af です。

```

{"PolicyARN": "arn:aws:autoscaling:us-west-2:123456789012:scalingPolicy:e3425d21-16f0-d701-89fb-12f98dac64af:resource/appstream/fleet/SampleFleetName:policyName/scale-out-utilization-policy"}

```

ここで、このポリシーの CloudWatch アラームを設定します。該当する名前、リージョン、アカウント番号、およびポリシー識別子を使用します。前のコマンドで返されたポリシー ARN を `--alarm-actions` パラメータに使用できます。

```

aws cloudwatch put-metric-alarm
--alarm-name alarmname \
--alarm-description "Alarm when Capacity Utilization exceeds 75 percent" \
--metric-name CapacityUtilization \
--namespace AWS/AppStream \
--statistic Average \
--period 300 \
--threshold 75 \
--comparison-operator GreaterThanOrEqualToThreshold \
--dimensions "Name=Fleet,Value=fleetname" \
--evaluation-periods 1 --unit Percent \
--alarm-actions "arn:aws:autoscaling:your-region-code:account-number-without-hyphens:scalingPolicy:policyid:resource/appstream/fleet/fleetname:policyName/polycyname"

```

例 2: 容量不足エラーに基づくスケーリングポリシーの適用

この AWS CLI の例では、フリートが `InsufficientCapacityError` エラーを返した場合、フリートを 1 ずつスケールアウトするスケーリングポリシーを設定します。

次のコマンドは、容量不足に基づくスケーリングポリシーを定義します。

```
aws application-autoscaling put-scaling-policy --cli-input-json file://scale-out-capacity.json
```

`scale-out-capacity.json` ファイルの内容は以下のようになります。

```
{
  "PolicyName": "polycyname",
  "ServiceNamespace": "appstream",
  "ResourceId": "fleet/fleetname",
  "ScalableDimension": "appstream:fleet:DesiredCapacity",
  "PolicyType": "StepScaling",
  "StepScalingPolicyConfiguration": {
    "AdjustmentType": "ChangeInCapacity",
    "StepAdjustments": [
      {
        "MetricIntervalLowerBound": 0,
        "ScalingAdjustment": 1
      }
    ],
    "Cooldown": 120
  }
}
```

コマンドが成功した場合、一部の詳細はアカウントおよびリージョンで固有ですが、出力は次のようになります。この例では、ポリシー識別子は `f4495f21-0650-470c-88e6-0f393adb64fc` です。

```
{"PolicyARN": "arn:aws:autoscaling:us-west-2:123456789012:scalingPolicy:f4495f21-0650-470c-88e6-0f393adb64fc:resource/appstream/fleet/SampleFleetName:policyName/scale-out-insufficient-capacity-policy"}
```

ここで、このポリシーの CloudWatch アラームを設定します。該当する名前、リージョン、アカウント番号、およびポリシー識別子を使用します。前のコマンドで返されたポリシー ARN を `--alarm-actions` パラメータに使用できます。

```
aws cloudwatch put-metric-alarm
--alarm-name alarmname \
--alarm-description "Alarm when out of capacity is > 0" \
--metric-name InsufficientCapacityError \
--namespace AWS/AppStream \
--statistic Maximum \
--period 300 \
--threshold 0 \
--comparison-operator GreaterThanThreshold \
--dimensions "Name=Fleet,Value=fleetname" \
--evaluation-periods 1 --unit Count \
--alarm-actions "arn:aws:autoscaling:your-region-code:account-
number-without-hyphens:scalingPolicy:policyid:resource/appstream/
```

```
fleet/fleetname:policyName/polycyname"
```

例 3: 低容量使用率に基づくスケーリングポリシーの適用

AWS CLI この例では、CapacityUtilization が低い場合に実際の容量を減らすためにフリートをスケールインするスケーリングポリシーを設定します。

以下のコマンドは、容量超過に基づくスケーリングポリシーを定義します。

```
aws application-autoscaling put-scaling-policy --cli-input-json file://scale-in-
capacity.json
```

scale-in-capacity.json ファイルの内容は以下のようになります。

```
{
  "PolicyName": "polycyname",
  "ServiceNamespace": "appstream",
  "ResourceId": "fleet/fleetname",
  "ScalableDimension": "appstream:fleet:DesiredCapacity",
  "PolicyType": "StepScaling",
  "StepScalingPolicyConfiguration": {
    "AdjustmentType": "PercentChangeInCapacity",
    "StepAdjustments": [
      {
        "MetricIntervalUpperBound": 0,
        "ScalingAdjustment": -25
      }
    ],
    "Cooldown": 360
  }
}
```

```
}  
}
```

コマンドが成功した場合、一部の詳細はアカウントおよびリージョンで固有ですが、出力は次のようになります。この例では、ポリシー識別子は 12ab3c4d-56789-0ef1-2345-6ghi7jk8lm90 です。

```
{"PolicyARN": "arn:aws:autoscaling:us-west-2:123456789012:scalingPolicy:12ab3c4d-56789-0ef1-2345-6ghi7jk8lm90:resource/appstream/fleet/SampleFleetName:policyName/scale-in-utilization-policy"}
```

ここで、このポリシーの CloudWatch アラームを設定します。該当する名前、リージョン、アカウント番号、およびポリシー識別子を使用します。前のコマンドで返されたポリシー ARN を `--alarm-actions` パラメータに使用できます。

```
aws cloudwatch put-metric-alarm  
--alarm-name alarmname \  
--alarm-description "Alarm when Capacity Utilization is less than or equal to 25 percent" \  
--metric-name CapacityUtilization \  
--namespace AWS/AppStream \  
--statistic Average \  
--period 120 \  
--threshold 25 \  
--comparison-operator LessThanOrEqualToThreshold \  
--dimensions "Name=Fleet,Value=fleetname" \  
--evaluation-periods 10 --unit Percent \  
--alarm-actions "arn:aws:autoscaling:your-region-code:account-number-without-hyphens:scalingPolicy:policyid:resource/appstream/fleet/fleetname:policyName/polycyname"
```

例 4: スケジュールに基づくフリートの容量の変更

スケジュールに基づいてフリートの容量を変更すると、需要の予測可能な変更に応じてフリートの容量をスケールすることができます。たとえば、稼働日の最初に、特定の数のユーザーが同時にストリーミング接続をリクエストすることが予想されます。スケジュールに基づいてフリート容量を変更するには、Application Auto Scaling [PutScheduledAction](#) API アクションまたは [put-scheduled-action](#) AWS CLI コマンドを使用できます。

フリート容量を変更する前に、AppStream 2.0 [describe-fleets](#) AWS CLI コマンドを使用して現在のフリート容量を一覧表示できます。

```
aws appstream describe-fleets --name fleetname
```

現在のフリートの容量は、次の出力のように表示されます (JSON 形式で表示)。

```
{
  {
    "ComputeCapacityStatus": {
      "Available": 1,
      "Desired": 1,
      "Running": 1,
      "InUse": 0
    },
  }
}
```

次に、put-scheduled-action コマンドを使用してフリートの容量を変更するスケジュールされたアクションを作成します。たとえば、次のコマンドでは、毎日午前 9:00 時 (UTC) に最小容量を 3 に変更し、最大容量を 5 に変更します。

 Note

cron 式の場合は、アクションを実行するタイミングを UTC で指定します。詳細については、「[Cron 式](#)」を参照してください。

```
aws application-autoscaling put-scheduled-action --service-namespace appstream \
--resource-id fleet/fleetname \
--schedule="cron(0 9 * * ? *)" \
--scalable-target-action MinCapacity=3,MaxCapacity=5 \
--scheduled-action-name ExampleScheduledAction \
--scalable-dimension appstream:fleet:DesiredCapacity
```

フリートの容量を変更するスケジュールされたアクションが正しく作成されたことを確認するには、[describe-scheduled-actions](#) コマンドを実行します。

```
aws application-autoscaling describe-scheduled-actions --service-namespace appstream --
resource-id fleet/fleetname
```

スケジュールされたアクションが正常に作成された場合、出力は次のようになります。


```
{
  "ScheduledActions": [
    {
      "ScalableDimension": "appstream:fleet:DesiredCapacity",
      "Schedule": "cron(0 9 * * ? *)",
      "ResourceId": "fleet/ExampleFleet",
      "CreationTime": 1518651232.886,
      "ScheduledActionARN": "<arn>",
      "ScalableTargetAction": {
        "MinCapacity": 3,
        "MaxCapacity": 5
      },
      "ScheduledActionName": "ExampleScheduledAction",
      "ServiceNamespace": "appstream"
    }
  ]
}
```

詳細については、「Application Auto Scaling ユーザーガイド」の「[スケジュールされたスケーリング](#)」を参照してください。

例 5: ターゲット追跡スケーリングポリシーの適用

ターゲット追跡スケーリングでは、フリートの容量使用率レベルを指定できます。

ターゲット追跡スケーリングポリシーを作成すると、Application Auto Scaling は、スケーリングポリシーをトリガーする CloudWatch アラームを自動的に作成および管理します。スケーリングポリシーは、指定されたターゲット値、またはそれに近い値に容量使用率を維持するため、必要に応じて容量を追加または削除します。アプリケーションの可用性を高めるために、フリートのスケールアウトはメトリクスに比例して可能な限り高速に行われますが、スケールインはより緩やかです。

次の [put-scaling-policy](#) コマンドは、AppStream 2.0 フリートに 75% の容量使用率を維持しようとするターゲット追跡スケーリングポリシーを定義します。

```
aws application-autoscaling put-scaling-policy --cli-input-json file:///config.json
```

config.json ファイルの内容は以下のようになります。

```
{
  "PolicyName": "target-tracking-scaling-policy",
```

```

"ServiceNamespace":"appstream",
"ResourceId":"fleet/fleetname",
"ScalableDimension":"appstream:fleet:DesiredCapacity",
"PolicyType":"TargetTrackingScaling",
"TargetTrackingScalingPolicyConfiguration":{
  "TargetValue":75.0,
  "PredefinedMetricSpecification":{
    "PredefinedMetricType":"AppStreamAverageCapacityUtilization"
  },
  "ScaleOutCooldown":300,
  "ScaleInCooldown":300
}
}

```

コマンドが成功した場合、一部の詳細はアカウントおよびリージョンで固有ですが、出力は次のようになります。この例では、ポリシー識別子は 6d8972f3-efc8-437c-92d1-6270f29a66e7 です。

```

{
  "PolicyARN": "arn:aws:autoscaling:us-west-2:123456789012:scalingPolicy:6d8972f3-efc8-437c-92d1-6270f29a66e7:resource/appstream/fleet/fleetname:policyName/target-tracking-scaling-policy",
  "Alarms": [
    {
      "AlarmARN": "arn:aws:cloudwatch:us-west-2:123456789012:alarm:TargetTracking-fleet/fleetname-AlarmHigh-d4f0770c-b46e-434a-a60f-3b36d653feca",
      "AlarmName": "TargetTracking-fleet/fleetname-AlarmHigh-d4f0770c-b46e-434a-a60f-3b36d653feca"
    },
    {
      "AlarmARN": "arn:aws:cloudwatch:us-west-2:123456789012:alarm:TargetTracking-fleet/fleetname-AlarmLow-1b437334-d19b-4a63-a812-6c67aaf2910d",
      "AlarmName": "TargetTracking-fleet/fleetname-AlarmLow-1b437334-d19b-4a63-a812-6c67aaf2910d"
    }
  ]
}

```

詳細については、Application Auto Scaling ユーザーガイドの「[ターゲット追跡スケーリングポリシー](#)」を参照してください。

自動スケーリング Amazon AppStream 2.0 の追加リソース

AppStream 2.0 Fleet Auto Scaling の操作に関するステップバイステップガイドンスについては、AWS コンピューティングブログ [Scaling Your Desktop Application Streams with Amazon AppStream 2.0](#) を参照してください。

Application Auto Scaling AWS CLI コマンドまたは API アクションの使用の詳細については、以下のリソースを参照してください。

- AWS CLI コマンドリファレンスの [application-autoscaling](#) セクション
- Application Auto Scaling API リファレンス <https://docs.aws.amazon.com/autoscaling/application/APIReference/>
- アプリケーション Auto Scaling ユーザーガイド <https://docs.aws.amazon.com/autoscaling/application/userguide/>

マルチセッションに関する推奨事項

マルチセッション環境でインスタンスの最大ユーザーセッション数を決定する場合、最適なパフォーマンスとストリーミングエクスペリエンスを確保するために、いくつかの要因を考慮する必要があります。インスタンスの最適なユーザーセッション数を決定するための推奨事項は以下のとおりです。

- リソース要件を評価する: セッション内で使用するアプリケーションのリソース要件を理解します。CPU、メモリ、ディスク I/O、ネットワーク帯域幅などの要因を考慮します。この評価は、各ユーザーセッションに通常必要なリソースの量を判断するのに役立ちます。
- インスタンスの仕様を検討する: CPU 数、使用可能なメモリ、GPU の仕様など、インスタンスの仕様を考慮します。インスタンスの仕様が低いほど、より多くのユーザーセッションを処理できます。AppStream 2.0 でサポートされるさまざまなインスタンスタイプと料金の詳細については、「[Amazon AppStream 2.0 の料金](#)」を参照してください。
- パフォーマンステスト: ユーザーセッション内で実行する予定のアプリケーションとワークロードのパフォーマンステストを実施します。リソース使用率、応答時間、システム全体のパフォーマンスを測定します。このデータを使用して、同時ユーザーセッションの数がパフォーマンスに与える影響を評価し、セッションとインスタンスの最適な比率を決定します。このような評価を AppStream 2.0 が提供するさまざまなインスタンスタイプにわたって実行することで、エンドユーザーに最適なインスタンスタイプやサイズを見つけることができます。AppStream 2.0 が提供するさまざまなインスタンスタイプの詳細については、「[the section called “インスタンスファミリー”](#)」を参照してください。

- リソース使用率をモニタリングする: 通常の使用中は、インスタンスのリソース使用率を継続的にモニタリングします。CPU、メモリ、ディスク使用率を監視します。パフォーマンスの低下を避けるため、リソース使用率が許容範囲内であることを確認してください。マルチセッション環境では、これらのメトリクスを AppStream 2.0 コンソールと CloudWatch コンソールで確認できます。詳細については、「[the section called “リソースのモニタリング”](#)」を参照してください。
- ユーザーの行動パターンを考慮する: ユーザーの行動パターンを分析し、ピーク使用時と同時使用の可能性について理解します。使用パターンが断続的または散発的なユーザーもいれば、1 日を通して使用状況が一貫しているユーザーもいます。ピーク時にリソースの競合が発生しないように、これらのパターンを考慮して最大ユーザーセッション数を決定します。

AppStream 2.0 では、選択したインスタンスタイプやサイズに関係なく、インスタンスごとに最大 50 のユーザーセッションを設定できます。ただし、これは上限であり、推奨される制限ではありません。次の表は、マルチセッションフリート内のインスタンスの最大ユーザーセッション数を決定する際に役立つ例を示しています。この表で推奨している最大ユーザー数は、一般的なガイドラインと仮定に基づいています。実際のパフォーマンスは、ワークロードの個々の特性、アプリケーションのリソース要件、ユーザーの行動によって異なる場合があるため、実際のワークロードでテストすることが重要です。

ワークロードタイプに基づく推奨事項

エンドユーザーカテゴリ	ワークロードタイプ	ユーザー例	ユースケース	推奨設定
1 つのタスクを実行し、最小限のアプリケーション数を使用するエンドユーザー	[Light] (ライト)	タスクワーカー、フロントデスクユーザー	データ入力アプリケーション、テキスト編集、踏み台ホスト	Stream.standard.xlarge/2xlarge または Stream.compute.xlarge+ または Stream.memory.xlarge+ の vCPU ごとに 4 ユーザー
1 つのタスクを実行し、最小限のアプリケーション数を使用	軽度 ~ 中程度	タスクワーカー、フロントデスクユーザー、コンタクト	データ入力アプリケーション、テキスト編集、踏み台ホスト	Stream.standard.xlarge/2xlarge または Stream.co

エンドユーザー カテゴリ	ワークロードタ イプ	ユーザー例	ユースケース	推奨設定
するエンドユー ザー		センターの従業 員	、チャット、E メール、メッ セージングアプ リ	mpute.xlarge+ ま たは Stream.me mory.xlarge+ の vCPU ごとに 2 ユーザー
複雑なスプレッ ドシート、プ レゼンテーション、大きなドキ ュメントを作成 するエンドユー ザー	中程度	タスクワー カー、コンタク トセンターの従 業員、ビジネス アナリスト	データ入力ア プリケーション、チャット、E メール、メッ セージングアプ リ、生産性向上 アプリ	Stream.me mory.xlarge+ ま たは Stream.co mpute.xlarge+ の vCPU ごとに 2 ユーザー
高パフォーマンスのワークロー ドを持つエンド ユーザー	中程度～重度	ナレッジワー カー、ソフト ウェアデベロッ パー、ビジネス インテリジェン スアナリスト	ソフトウェアス クリプト作成	Stream.me mory.xlarge+ ま たは Stream.co mpute.xlarge+ の vCPU ごとに 1 ユーザー
高パフォーマンスのワークロー ドを持つエンド ユーザー	重度	ナレッジワー カー、ソフト ウェアデベロッ パー、データサ イエンティスト	画面共有、デー タ分析、音声会 議	Stream.me mory.xlarge+ ま たは Stream.co mpute.xlarge+ の 2 vCPU ごとに 1 ユーザー

エンドユーザー カテゴリ	ワークロードタ イプ	ユーザー例	ユースケース	推奨設定
グラフィック と大量のコン ピューティン グ/メモリリソ ースを必要とする ワークロードを 持つエンドユー ザー	重度 ~ 高速化	グラフィック ス/アーキテ クチャデザイ ナー、CAD/CAM ユーザー	オーディオ会 議、グラフィッ クスを多用する アプリケーション (リモートグラ フィックスワー クステーション など)	2 vCPU Graphics.g4dn.* ごとに 1 ユー ザー
グラフィックス と大量のコン ピューティング/ メモリリソー ースを必要とする ワークロードを 持つエンドユー ザー	高速化	ビデオエディ タ、ゲーマーと ゲームデベロッ パー、データ マイナー、GIS データエンジニア、AI サイエ ンティスト	オーディオ会 議、ビデオトラ ンスコーディン グと 3D レンダ リング、フォト リアリスティッ クデザイン、 グラフィックス ワークステー ション、ML モデ ルトレーニング 、ML 推論	2 vCPU Graphics.G5.* ご とに 1 ユーザー

ユーザー認証

次のトピックでは、Amazon AppStream 2.0 ユーザーの認証と認可について説明します。

内容

- [Amazon AppStream 2.0 ユーザープール](#)
- [Amazon AppStream 2.0 と SAML 2.0 の統合](#)

Amazon AppStream 2.0 ユーザープール

AppStream 2.0 ユーザープールは、各 AWS リージョンの永続ポータルを通じてユーザーのアプリケーションへのアクセスを管理する簡単な方法を提供します。この機能は、[Active Directory](#) と [SAML 2.0 フェデレーション](#) を介したユーザー管理に代わる組み込みの機能です。スタックが Active Directory ドメインに参加しているフリートに関連付けられている場合、スタックをユーザープール内のユーザーに割り当てることはできません。

AppStream 2.0 ユーザープールには次の主要機能があります。

- ユーザーは、自分が選択できる E メールアドレスとパスワードを使用して、永続的な URL とログイン認証情報を介してアプリケーションスタックにアクセスできます。
- ユーザーの E メールアドレスでは、大文字と小文字は区別されます。ログイン時に、ユーザープールアカウントを作成したときに指定した E メールアドレスと同じ大文字小文字を使用しない E メールアドレスを指定した場合、「ユーザーが存在しません」というエラーメッセージが表示されます。
- 複数のスタックをユーザーに割り当てることができます。これにより、AppStream 2.0 ではログイン時に複数のアプリケーションカタログがユーザーに表示されます。
- 新しいユーザーを作成すると、ようこそ E メールが自動的に送信されます。E メールには、説明、ログインポータルのリンク、およびログインポータルに接続するための一時パスワードが含まれています。
- ユーザーを作成した後は、ユーザーを特に無効にしない限り、ユーザーは有効になります。
- 管理者はどのユーザーがどのアプリケーションスタックにアクセスできるかを制御したり、完全にアクセスを無効にしたりできます。

トピック

- [Amazon AppStream 2.0 のユーザープールエンドユーザーエクスペリエンス](#)
- [Amazon AppStream 2.0 でパスワードを忘れた場合のリセット](#)
- [Amazon AppStream 2.0 でのユーザープール管理](#)

Amazon AppStream 2.0 のユーザープールエンドユーザーエクスペリエンス

次のステップは、ユーザープール内のユーザーの初期接続エクスペリエンスを要約したものです。

1. E メールアドレスを指定して、リージョンの新しいユーザーを作成します。
2. AppStream 2.0 がようこそ E メールをユーザーに送信します。
3. 1 つまたは複数のスタックをユーザーに割り当てます。
4. AppStream 2.0 はオプションの通知 E メールを送信します。この E メールには、新しく割り当てられたスタックへのアクセス方法に関する情報が含まれています。
5. ユーザーはようこそ E メールに含まれる情報を入力してログインポータルに接続し、永久パスワードを設定します。ログインポータルリンクの有効期限が切れることはなく、いつでも使用できます。
6. ユーザーは自分の E メールアドレスと永久パスワードを入力して AppStream 2.0 にサインインします。
7. サインインした後、ユーザーは自分のアプリケーションカタログを表示できます。

招待 E メールに含まれているログインポータルリンクは、将来の使用のために保存します。このリンクは変更されることはなく、ユーザープールのすべてのユーザーに対して有効です。ログインポータルの URL とユーザープール内のユーザーは、リージョンごとに管理されます。

Amazon AppStream 2.0 でパスワードを忘れた場合のリセット

ユーザーが自分のパスワードを忘れた場合は、次のステップに従ってログインポータルリンク (ようこそ E メールに記載) に接続し、新しいパスワードを選択してください。

新しいパスワードを選択するには

1. ようこそ E メールに記載されているログインリンクを使用して、AppStream 2.0 ログインポータルを開きます。
2. [Forgot Password?] を選択します。

3. ユーザーを作成するために使用した E メールアドレスをユーザープールに入力し、[次へ] を選択します。

E メールアドレスでは大文字と小文字が区別されます。ログイン時に、ユーザープールアカウントを作成したときに指定した E メールアドレスと同じ大文字小文字を使用しない E メールアドレスを使用した場合、「ユーザーが存在しません」というエラーメッセージが表示されます。

4. パスワードのリセットリクエストメッセージの E メールを確認します。E メールが見つからない場合は、スパム E メールフォルダを確認します。E メールにある検証コードを [Verification Code] に入力します。

 Note

検証コードは 24 時間有効です。この時間内に新しいパスワードが選択されない場合は、新しい検証コードをリクエストします。

5. 以下に示すパスワードルールに従って、新しいパスワードを入力して確定します。[Reset Password] をクリックします。

Amazon AppStream 2.0 でのユーザープール管理

ユーザープールでユーザーを作成および管理するには、目的の AWS リージョンの AppStream 2.0 コンソールにサインインし、左側のナビゲーションペインで [ユーザープール] を選択します。[User Pool] ダッシュボードは、一部のアクションに対してユーザーのリストで一括オペレーションをサポートします。管理者は、[アクション] リストから同じアクションを実行する複数のユーザーを選択できます。ユーザープール内のユーザーは、リージョンごとに作成および管理されます。

AppStream 2.0 では、一括ユーザー作成または無効化はサポートされていません。ただし、[CreateStreamingUrl](#) API アクションで Amazon Cognito を使用すると、複数のユーザーのアクセスを効率的に管理できます。Amazon Cognito ユーザープールを使用すると、ユーザーを登録してサインインするための独自のディレクトリをすばやく作成できます。さらに、Amazon Cognito ユーザープールを使用してユーザープロフィールを保存できます。AppStream 2.0 を Cognito ユーザープールと統合する方法については、[Create a SaaS Portal with Amazon AppStream 2.0](#) プロジェクトを参照してください。

 Note

AppStream 2.0 は、ユーザーが作った新規ユーザーを作成するか、ユーザーをスタックに割り当てると、ユーザーに代わって E メールを送信します。E メールを確実に配信するには、

許可リスト `no-reply@accounts.aws-region-code.amazonappstream.com` に追加します。ここで、*aws-region-code* は作業中の有効な AWS リージョンコードです。ユーザーが E メールを見つけられない場合は、「スパム」E メールフォルダを確認するようユーザーに求めます。

タスク

- [Amazon AppStream 2.0 でのユーザーの作成](#)
- [Amazon AppStream 2.0 でのユーザーの削除](#)
- [Amazon AppStream 2.0 でのユーザーへのスタックの割り当て](#)
- [Amazon AppStream 2.0 でのユーザーからのスタックの割り当て解除](#)
- [Amazon AppStream 2.0 でのユーザーの無効化](#)
- [Amazon AppStream 2.0 でのユーザーの有効化](#)
- [Amazon AppStream 2.0 での招待 Eメールの再送信](#)

Amazon AppStream 2.0 でのユーザーの作成

リージョン内の新しいユーザーごとに、有効な一意の E メールアドレスを入力する必要があります。ただし、新しいユーザー用の E メールアドレスは別のリージョンの新しいユーザーに対して再利用できます。

新しいユーザーを作成する際は、以下の点に注意してください。

- すでに作成したユーザーの E メールアドレス、姓名は変更できません。このユーザー情報を変更するには、ユーザーを無効にします。そして、ユーザーを (新しいユーザーとして) 作成し、必要に応じて更新した情報を入力します。
- ユーザーの E メールアドレスでは、大文字と小文字は区別されます。ログイン時に、ユーザープールアカウントを作成したときに指定した E メールアドレスと同じ大文字小文字を使用しない E メールアドレスを指定した場合、「ユーザーが存在しません」というエラーメッセージが表示されます。
- ユーザーの作成後、1 つ以上のスタックをユーザーに割り当てることができます。

新しいユーザーを作成するには

1. AppStream 2.0 コンソールを <https://console.aws.amazon.com/appstream2> で開きます。

2. 左側のナビゲーションペインで、[ユーザープール]、[Create User (ユーザーの作成)] の順に選択します。
3. [E メール] に、このユーザーの一意の E メールアドレスを入力します。
4. 対応するフィールドにユーザーの名と姓を入力します。これらのフィールドは一意である必要はありません。
5. [Create User] を選択します。

ユーザーが作成されたら、AppStream 2.0 がよろこ E メールを送信します。この E メールには、ログインポータルリンク、使用するログイン E メールアドレス、および仮パスワードが含まれています。ユーザーは、ログインポータルに移動し、仮パスワードを入力して、アプリケーションにアクセスするための永続的なパスワードを設定できます。

デフォルトでは、新しいユーザーのステータスは [有効] になります。この場合、1 つまたは複数のスタックをユーザーに割り当て、他の管理アクションを実行できます。

Amazon AppStream 2.0 でのユーザーの削除

ユーザーを有効または無効にすることはできますが、AppStream 2.0 コンソールを使用してユーザーを削除することはできません。ユーザーを削除するには、[DeleteUser](#) API アクションを使用する必要があります。

Amazon AppStream 2.0 でのユーザーへのスタックの割り当て

ユーザーは 1 つ以上のスタックをユーザープールの 1 人以上のユーザーに割り当てることができます。それらが少なくとも 1 つのスタックに割り当てられたら、ユーザーは AppStream 2.0 にログインしてアプリケーションを起動できます。ユーザーに複数のスタックが割り当てられている場合、アプリケーションを起動する前に選択できるカタログとしてスタックのリストが表示されます。

Note

スタックが Active Directory ドメインに参加しているフリートに関連付けられている場合、スタックをユーザーに割り当ててはできません。

スタックをユーザーに割り当てるには

1. AppStream 2.0 コンソールを <https://console.aws.amazon.com/appstream2> で開きます。

2. 左側のナビゲーションペインで、[ユーザープール] を選択し、希望するユーザーを選択します。
3. [Actions]、[Assign stack] の順に選択します。詳細については、[AppStream 2.0 でのアクティブディレクトリの使用](#) を参照してください。
4. 一覧を確認して、正しいユーザーが指定されたことを確認します。[スタック] で、割り当てるスタックを選択します。
5. デフォルトでは、[Send email notification to user] は有効になっています。この時点でユーザーに通知 E メール通知を送信しない場合は、このオプションをクリアします。
6. [Assign stack] を選択します。

Amazon AppStream 2.0 でのユーザーからのスタックの割り当て解除

ユーザーは、ユーザープールの 1 人以上のユーザーのスタックの割り当てを解除することができます。スタックを割り当て解除すると、ユーザーはそのスタックからアプリケーションを起動できなくなります。スタックを割り当て解除したときにユーザーが接続されていると、ユーザーのセッション Cookie が期限切れ (約 1 時間) になるまで、セッションはアクティブなままになります。

ユーザーからスタックの割り当てを解除するには

1. AppStream 2.0 コンソールを <https://console.aws.amazon.com/appstream2> で開きます。
2. 左側のナビゲーションペインで、[ユーザープール] を選択し、希望するユーザーを選択します。
3. [Actions]、[Unassign stack] の順に選択します。
4. 一覧を確認して、正しいユーザーが指定されたことを確認します。[スタック] で、割り当てを解除するスタックを選択します。このリストには、割り当て済みまたは未割り当てのすべてのスタックが含まれます。
5. [Unassign stack] を選択します。

Amazon AppStream 2.0 でのユーザーの無効化

ユーザープール内の 1 人以上のユーザーを一度に 1 人ずつ無効にできます。無効化されると、ユーザーは再び有効化されるまで AppStream 2.0 にログインできなくなります。このアクションではユーザーは削除されません。無効にしたときにユーザーが現在接続されている場合、セッション Cookie が期限切れになるまで (約 1 時間)、ユーザーのセッションは有効のままになります。ユーザーのスタック割り当ては保持されます。ユーザーが再び有効にされると、そのスタック割り当てが再びアクティブになります。

ユーザーを無効にするには

1. AppStream 2.0 コンソールを <https://console.aws.amazon.com/appstream2> で開きます。
2. 左側のナビゲーションペインで、[ユーザープール] を選択し、希望するユーザーを選択します。
3. [Actions]、[Disable user] の順に選択します。
4. 正しいユーザーが指定されていることを確認したら、[Disable User (ユーザーを無効にする)] を選択します。

Amazon AppStream 2.0 でのユーザーの有効化

ユーザープール内の 1 人以上のユーザーを一度に 1 人ずつ有効にできます。有効にすると、ユーザーは AppStream 2.0 にログインして、割り当てられているスタックからアプリケーションを起動できます。ユーザーが無効になった場合でも、これらの割り当ては保持されます。

ユーザーを有効にするには

1. AppStream 2.0 コンソールを <https://console.aws.amazon.com/appstream2> で開きます。
2. 左側のナビゲーションペインで、[ユーザープール] を選択し、希望するユーザーを選択します。
3. [Actions]、[Enable user] を選択します。
4. 正しいユーザーが指定されていることを確認したら、[Enable User (ユーザーを有効にする)] を選択します。

Amazon AppStream 2.0 での招待 E メールの再送信

接続手順が含まれた招待 E メールをユーザープールのユーザーに再送信できます。未使用のパスワードは 7 日後に有効期限が切れます。新しい仮パスワードを提供するには、よろこ E メールを再送信する必要があります。このオプションを使用できるのは、ユーザーが永続的なパスワードを設定するまでの期間のみです。既にパスワードを設定して忘れた場合は、新しいパスワードを設定できます。詳細については、[Amazon AppStream 2.0 でパスワードを忘れた場合のリセット](#) を参照してください。

ユーザーの招待 E メールを再送信するには

1. AppStream 2.0 コンソールを <https://console.aws.amazon.com/appstream2> で開きます。
2. 左側のナビゲーションペインで、[ユーザープール] を選択し、希望するユーザーを選択します。
3. [User Details] で、[Resend welcome email] を選択します。

4. ユーザープールダッシュボードの上部にある成功メッセージを確認します。

Amazon AppStream 2.0 と SAML 2.0 の統合

Amazon AppStream 2.0 では、Security Assertion Markup Language 2.0 (SAML 2.0) を通した AppStream 2.0 スタックへの ID フェデレーションをサポートしています。Windows Server の Active Directory Federation Service (AD FS)、Ping One Federation Server、Okta などの SAML 2.0 をサポートする ID プロバイダ (IdP) を使用して、AppStream 2.0 ユーザーにオンボーディングフローを提供できます。

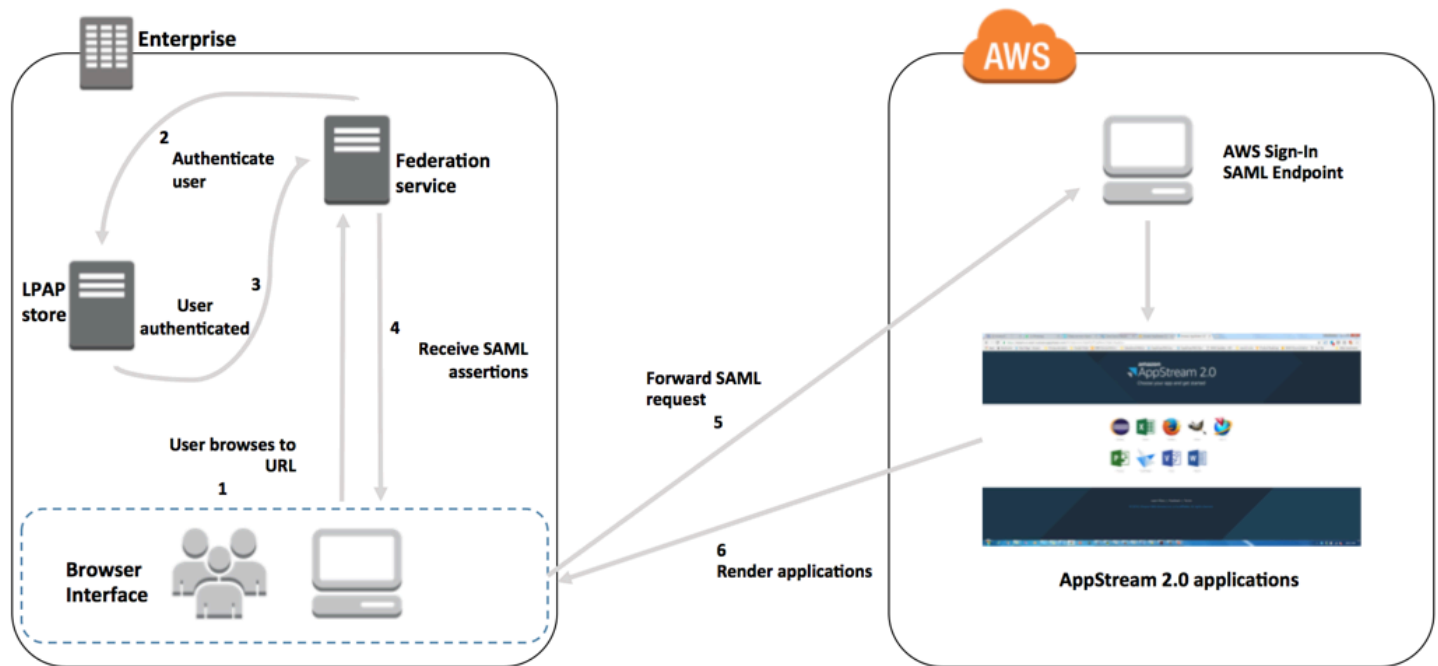
この機能により、ユーザーは既存の ID 認証情報を使用して AppStream 2.0 アプリケーションヘワンクリックでアクセスができ、便利です。また、IdP による ID 認証によるセキュリティ上の利点もあります。IdP を使用して、どのユーザーが特定の AppStream 2.0 スタックへのアクセス権限を持つかを制御することができます。

トピック

- [認証ワークフローの例](#)
- [SAML のセットアップ](#)
- [AppStream 2.0 と SAML 2.0 の統合](#)

認証ワークフローの例

次の図は、AppStream 2.0 とサードパーティーの ID プロバイダー (IdP) との間の認証フローを示しています。この例では、管理者が AppStream 2.0 へアクセスするためのサインインページ (applications.exampleco.com) をセットアップしています。ウェブページでは、SAML 2.0 準拠のフェデレーションサービスを使用してサインオンリクエストをトリガーしています。また、管理者は AppStream 2.0 へのアクセスを許可するユーザーをセットアップしています。



1. ユーザーは `https://applications.exampleco.com` をブラウジングします。サインインページがユーザーの認証をリクエストします。
2. フェデレーションサービスが組織の ID ストアからの認証をリクエストします。
3. ID ストアはユーザーを認証し、フェデレーションサービスに認証レスポンスを返します。
4. 認証が成功すると、フェデレーションサービスはユーザーのブラウザに SAML アサーションを送信します。
5. ユーザーのブラウザは、SAML アサーションを AWS サインイン SAML エンドポイント (`https://signin.aws.amazon.com/saml`) に投稿します。AWS サインインは SAML リクエストを受け取り、リクエストを処理し、ユーザーを認証し、認証トークンを AppStream 2.0 に転送します。

AWS GovCloud (US) リージョンでの SAML の使用については、AWS GovCloud (US) 「ユーザーガイド」の[AWS 「Identity and Access Management」](#)を参照してください。

6. AppStream 2.0 は AWS、からの認証トークンを使用してユーザーを認可し、アプリケーションをブラウザに提示します。

ユーザーの立場では、この処理を意識することはありません ユーザーは組織の内部ポータルから開始し、AWS 認証情報を入力することなく AppStream 2.0 アプリケーションポータルに自動的にリダイレクトされます。

SAML のセットアップ

既存の認証情報を使用してユーザーが AppStream 2.0 にサインインしたり、アプリケーションのストリーミングを開始したりできるように、SAML 2.0 を使用して ID フェデレーションを設定します。これを行うには、IAM ロールとリリースステート URL を使用して SAML 2.0 準拠の ID プロバイダー (IdP) を設定し、を有効に AWS してフェデレーティッドユーザーが AppStream 2.0 スタックにアクセスできるようにします。IAM ロールは、スタックへのアクセス許可をユーザーに付与します。リリースステートは、AWS による認証に成功した後にユーザーが転送されるスタックポータルです。

内容

- [前提条件](#)
- [ステップ 1: IAM で SAML ID AWS プロバイダーを作成する](#)
- [ステップ 2: SAML 2.0 フェデレーション IAM ロールを作成する](#)
- [ステップ 3: IAM ロールにインラインポリシーを埋め込む](#)
- [ステップ 4: SAML ベースの IdP を設定する](#)
- [ステップ 5: SAML 認証レスポンスのアサーションを作成する](#)
- [ステップ 6: フェデレーションのリリースステートを設定する](#)

前提条件

SAML 2.0 接続を設定する前に、以下の前提条件を満たしていることを確認してください。

1. AWSとの信頼関係を確立するように SAML ベースの IdP を設定します。
 - 組織のネットワーク内で、ID ストアを SAML ベースの IdP で機能するように設定します。設定のリソースについては、[AppStream 2.0 と SAML 2.0 の統合](#) を参照してください。
 - SAML ベースの IdP を使用して、組織を IdP として定義するフェデレーションメタデータドキュメントを生成し、ダウンロードします。署名されたこの XML ドキュメントは、証明書利用者の信頼を確立するために使用されます。後で IAM コンソールからアクセスできる場所にこのファイルを保存します。
2. AppStream 2.0 管理コンソールを使用して AppStream 2.0 スタックを作成します。この章でこれから説明するように、IAM ポリシーを作成し、AppStream 2.0 で IdP 統合を設定するには、このスタック名が必要です。

AppStream 2.0 スタックを作成するには、AppStream 2.0 マネジメントコンソール、AWS CLI、または AppStream 2.0 API を使用します。詳細については、「[Amazon AppStream 2.0 フリートとスタックを作成する](#)」を参照してください。

ステップ 1: IAM で SAML ID AWS プロバイダーを作成する

まず、IAM で SAML IdP AWS を作成します。この IdP は、組織内の IdP ソフトウェアによって生成されたメタデータドキュメントを使用して、組織の IdP と AWS 信頼の関係を定義します。詳細については、IAM ユーザーガイドの [Creating and Managing a SAML Identity Provider \(Amazon Web Services Management Console\)](#) を参照してください。AWS GovCloud (US) リージョンでの SAML IdPs の使用については、AWS GovCloud (US) 「ユーザーガイド」の [AWS 「Identity and Access Management」](#) を参照してください。

ステップ 2: SAML 2.0 フェデレーション IAM ロールを作成する

次に、SAML 2.0 フェデレーション IAM ロールを作成します。この手順では、IAM と組織の IdP 間に、IdP をフェデレーションの信頼されるエンティティと識別する信頼関係を確立します。

SAML IdP への IAM ロールを作成するには

1. IAM コンソール (<https://console.aws.amazon.com/iam/>) を開きます。
2. ナビゲーションペインで [ロール]、[ロールの作成] の順に選択します。
3. [ロールタイプ] で [SAML 2.0 フェデレーション] を選択します。
4. [SAML プロバイダー] で、作成した SAML IdP を選択します。

Important

2 つの SAML 2.0 アクセスメソッド ([プログラムによるアクセスのみを許可する] または [プログラムによるアクセスと Amazon Web Services マネジメントコンソールによるアクセスを許可する]) のいずれも選択しないでください。

5. [属性] で、[SAML:sub_type] を選択します。
6. [値] に「<https://signin.aws.amazon.com/saml>」と入力します。この値は、persistent の値の SAML サブジェクトタイプアサーションを含む SAML ユーザーストリーミングリクエストへのロールアクセスを制限します。SAML:sub_type が persistent の場合、IdP は特定のユーザーからのすべての SAML リクエストで同一意の値を NameID 要素に送信します。SAML:sub_type アサーションの詳細については、「[API アクセスに AWS SAML ベースの](#)

フェデレーションを使用する」の「SAML ベースのフェデレーションでユーザーを一意に識別する」セクションを参照してください。

Note

<https://signin.aws.amazon.com/saml> エンドポイントは高可用性ですが、の us-east-1 リージョンでのみホストされます AWS。リージョンエンドポイントの可用性の 1 つが影響を受けるような万一のサービスの中断を防ぐには、リージョンエンドポイントを使用し、フェイルオーバー用に追加の SAML サインインエンドポイントを設定します。詳細については、「[フェイルオーバーにリージョン SAML エンドポイントを使用する方法](#)」を参照してください。

7. 正しい信頼されたエンティティおよび条件を確認して SAML 2.0 の信頼情報をお確かめたら、[Next: Permissions] (次: アクセス許可) を選択します。
8. [アクセス権限ポリシーをアタッチする] ページで、[Next: Tags] を選択します。
9. (オプション) 追加する各タグのキーと値を入力します。詳細については、[Tagging IAM Users and Roles](#) を参照してください。
10. 終了したら、[Next: Review] を選択します。後でこのロールにインラインポリシーを作成して埋め込みます。
11. [Role name] (ロール名) に、このロールの目的を識別できる名前を入力します。なぜなら複数エンティティがロールを参照している可能性があります。ロールが作成された後のロールの名前の編集はできません。
12. (オプション) [ロールの説明] に、新しいロールの説明を入力します。
13. ロールの詳細を確認し、[ロールの作成] を選択します。
14. (オプション) サードパーティー SAML 2.0 アイデンティティプロバイダーまたは証明書ベースの認証を使用してセッションコンテキストまたは属性ベースのアプリケーションの使用権限を使用する場合は、新しい IAM ロールの信頼ポリシーに sts:TagSession 権限を追加する必要があります。詳細については、「[サードパーティーの SAML 2.0 アイデンティティプロバイダーを使用した属性ベースのアプリケーションの使用権限](#)」および「[AWS STSでのセッションタグの受け渡し](#)」を参照してください。

新しい IAM ロールの詳細ページで、[Trust relationships] (信頼関係) タブを選択してから、[Edit trust relationship] (信頼関係の編集) を選択します。[信頼関係の編集] ポリシーエディタが起動します。sts:TagSession 許可は、次のように追加します。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Federated": "arn:aws:iam::ACCOUNT-ID-WITHOUT-HYPHENS:saml-
provider/IDENTITY-PROVIDER"
      },
      "Action": [
        "sts:AssumeRoleWithSAML",
        "sts:TagSession"
      ],
      "Condition": {
        "StringEquals": {
          "SAML:sub_type": "persistent"
        }
      }
    }
  ]
}
```

[**IDENTITY-PROVIDER**] を、ステップ 1 で作成した SAML IdP の名前と置き換えます。次に、[Update Trust Policy] (信頼ポリシーの更新) を選択します。

ステップ 3: IAM ロールにインラインポリシーを埋め込む

次に、作成したロールにインライン IAM ポリシーを埋め込みます。インラインポリシーを埋め込むと、ポリシーのアクセス許可が、間違ったプリンシパルエンティティにアタッチされることを回避できます。インラインポリシーは、フェデレーティッドユーザーに作成した AppStream 2.0 スタックへのアクセスを提供します。

1. 作成した IAM ロールの詳細で、[Permissions (アクセス許可)] タブを選択し、[Add inline policy (インラインポリシーの追加)] を選択します。ポリシーの作成ウィザードが起動します。
2. [ポリシーの作成] で、[JSON] タブを選択します。
3. 次の JSON ポリシーを JSON ウィンドウにコピーして貼り付けます。次に、AWS リージョンコード、アカウント ID、スタック名を入力してリソースを変更します。次のポリシー、"Action": "appstream:Stream" は、作成したスタックのストリーミングセッションに接続するアクセス許可を AppStream 2.0 ユーザーに提供するアクションです。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "appstream:Stream",
      "Resource": "arn:aws:appstream:REGION-CODE:ACCOUNT-ID-WITHOUT-
HYPHENS:stack/STACK-NAME",
      "Condition": {
        "StringEquals": {
          "appstream:userId": "${saml:sub}"
        }
      }
    }
  ]
}
```

REGION-CODE を AppStream 2.0 スタックが存在する AWS リージョンに置き換えます。**STACK-NAME** をこのスタック名に置き換えます。**[STACK-NAME]** の大文字と小文字は区別されます。また、AppStream2.0 マネジメントコンソールのスタックダッシュボードに表示されるスタック名と完全に一致する必要があります。

AWS GovCloud (US) リージョンのリソースの場合は、ARN に次の形式を使用します。

arn:aws-us-gov:appstream:REGION-CODE:ACCOUNT-ID-WITHOUT-HYPHENS:stack/STACK-NAME

4. (オプション) SAML 2.0 マルチスタックアプリケーションカタログのサードパーティー SAML 2.0 アイデンティティプロバイダーを使用して、属性ベースのアプリケーションの使用権限を使用する予定の場合は、IAM ロールのインラインポリシー内のリソースは **"Resource": "arn:aws:appstream:REGION-CODE:ACCOUNT-ID-WITHOUT-HYPHENS:stack/*"** であり、アプリケーション使用権限がスタックへのストリーミングアクセスを制御します。スタックリソースに追加の保護を適用するには、ポリシーに明示的拒否を追加します。詳細については、「[サードパーティーの SAML 2.0 アイデンティティプロバイダーを使用した属性ベースのアプリケーションの使用権限](#)」、および「[ポリシーの評価論理](#)」を参照してください。
5. 完了したら、[ポリシーの確認] をクリックします。構文エラーがある場合は、[Policy Validator \(ポリシー検証\)](#) によってレポートされます。

ステップ 4: SAML ベースの IdP を設定する

次に、SAML ベースの IdP によっては、<https://signin.aws.amazon.com/static/saml-metadata.xml> の saml-metadata.xml ファイルを IdP にアップロードして、サービスプロバイダー AWS として信頼するように IdP を手動で更新する必要がある場合があります。このステップは、IdP のメタデータを更新します。一部の IdP では、すでに更新が設定されています。この場合は、次のステップに進みます。

IdP でこの更新がまだ設定されていない場合には、IdP から提供されるドキュメンテーションでメタデータを更新する方法に関する情報を確認します。プロバイダーによっては、URL を入力し、また IdP によってファイルを取得してインストールするオプションが提供されます。また、URL からファイルをダウンロードし、ローカルファイルとして指定する必要があるプロバイダーもあります。

ステップ 5: SAML 認証レスポンスのアサーションを作成する

次に、IdP が認証レスポンスで SAML 属性 AWS として 送信する情報を設定する必要がある場合があります。IdP によっては、この情報はすでに設定されています。この場合、この手順をスキップしてステップ 6 に進みます。

この情報がまだ設定されていない場合は、IdP で次を実行します。

- SAML Subject NameID (SAML サブジェクト名 ID) – 署名するユーザーの一意の識別子。

Note

ドメイン結合フリードを使用したスタックの場合、ユーザーの NameID 値は、sAMAccountName を使用した「*domain*\username」または userPrincipalName を使用した「username@domain.com」の形式で指定する必要があります。sAMAccountName 形式を使用している場合、NetBIOS 名または完全修飾ドメイン名 (FQDN) を使用して *domain* を指定できます。Active Directory の一方向信頼シナリオには、sAMAccountName 形式が必要です。詳細については、「[AppStream 2.0 でのアクティブディレクトリの使用](#)」を参照してください。

- SAML Subject Type (SAML サブジェクトタイプ) (値を persistent に設定) – 値を persistent に設定すると、特定のユーザーからのすべての SAML リクエストの NameID 要素に同じ一意の値を IdP が送信することを確保できます。persistent で説明するように、SAML sub_type が [the section called “ステップ 2: SAML 2.0 フェデレーション IAM ロールを作成する”](#) に設定されている SAML リクエストのみを許可する条件が IAM ポリシーに含まれていることを確認します。

- **Attribute** 要素 (**Name** 属性が <https://aws.amazon.com/SAML/Attributes/Role> に設定) – この要素には、IdP によってマッピングされたユーザーの IAM ロールと SAML IdP を一覧表示する 1 つ以上の **AttributeValue** 要素が含まれます。このロールと IdP は、カンマ区切りの ARN のペアとして指定されます。
- **Attribute Name** 属性が <https://aws.amazon.com/SAML/Attributes/RoleSessionName> に設定された 要素 – この要素には、SSO 用に発行された AWS 一時的な認証情報の識別子を提供する **AttributeValue** 要素が 1 つ含まれています。**AttributeValue** 要素の値は 2~64 文字で、英数字、アンダースコア、および + (プラス記号)、= (等号)、, (カンマ)、. (ピリオド)、@ (アットマーク)、- (ハイフン) のみで構成する必要があります。スペースを含めることはできません。通常、この値はユーザー ID (bobsmith) またはメールアドレス (bobsmith:bobsmith@example.com) になります。ユーザーの表示名 (Bob Smith) のように、スペースを含む値とすることはできません。
- **Attribute** 要素との **Name** 属性を <https://aws.amazon.com/SAML/Attributes/PrincipalTag:SessionContext> に設定する (オプション) – この要素には、セッションコンテキストパラメータをストリーミングアプリケーションに渡すために使用できるパラメータを提供する 1 つの **AttributeValue** 要素が含まれています。詳細については、「[Amazon AppStream 2.0 のセッションコンテキスト](#)」を参照してください。
- **Name** 属性が <https://aws.amazon.com/SAML/Attributes/PrincipalTag:ObjectSid> (オプション) に設定された **Attribute** 要素 – この要素には、サインインしているユーザーの Active Directory セキュリティ識別子 (SID) を提供する **AttributeValue** 要素が 1 つ含まれています。このパラメータを証明書ベースの認証で使用する、Active Directory ユーザーへの強力なマッピングが可能になります。
- **Name** 属性が <https://aws.amazon.com/SAML/Attributes/PrincipalTag:Domain> (オプション) に設定された **Attribute** 要素 – この要素には、サインインしているユーザーの Active Directory DNS 完全修飾ドメイン名 (FQDN) を提供する **AttributeValue** 要素が 1 つ含まれています。このパラメータは、ユーザーの Active Directory `userPrincipalName` に代替サフィックスが含まれている場合に、証明書ベースの認証で使用されます。値は、サブドメインを含め、**domain.com** の形式で指定する必要があります。
- **Attribute** 要素 (**SessionDuration** 属性が <https://aws.amazon.com/SAML/Attributes/SessionDuration> に設定) (オプション) – この要素には、再認証が必要となる前にユーザーがアクティブでいられるフェデレーティッドストリーミングセッションの最大時間を特定する 1 つの **AttributeValue** 要素が含まれています。デフォルト値は 60 分 (3,600 秒) です。詳細については、[Configuring SAML Assertions for the Authentication Response](#) で An optional Attribute element with the SessionDuration attribute set to <https://aws.amazon.com/SAML/Attributes/SessionDuration> セクション を参照してください。

Note

SessionDuration はオプションの属性ですが、これを SAML レスポンスに含めることをお勧めします。この属性を指定しない場合、セッション時間はデフォルト値の 60 分に設定されます。

ユーザーが AppStream 2.0 ネイティブクライアントを使用するか、新しいエクスペリエンスでウェブブラウザを使用して AppStream 2.0 でストリーミングアプリケーションにアクセスしている場合、ユーザーのセッション時間が期限切れになると、セッションは切断されます。ユーザーが古い/クラシックエクスペリエンスでウェブブラウザを使用して AppStream 2.0 のストリーミングアプリケーションにアクセスしている場合、ユーザーのセッション時間が期限切れになってユーザーがブラウザページを更新すると、セッションは切断されます。

これらの要素を設定する方法については、IAM ユーザーガイドの [Configuring SAML Assertions for the Authentication Response](#) を参照してください。IdP に特定の設定要件に関する詳細は、IdP のドキュメンテーションを参照してください。

ステップ 6: フェデレーションのリリース状態を設定する

最後に、IdP を使用して AppStream 2.0 スタックのリリース状態の URL を指すようにフェデレーションのリリース状態を設定します。による認証が成功すると AWS、ユーザーは AppStream 2.0 スタックポータルに移動します。このポータルは、SAML 認証レスポンスのリレー状態として定義されます。

リリース状態の URL の形式は次のようになります。

```
https://relay-state-region-endpoint?stack=stackname&accountId=aws-account-id-without-hyphens
```

Amazon Web Services アカウント ID、スタック名、およびスタックが位置するリージョンと関連付けられたリリース状態のエンドポイントを基に、リリース状態の URL を構成します。

オプションで、自動的に起動するアプリケーションの名前を指定できます。アプリケーション名を見つけるには、AppStream 2.0 コンソールでイメージを選択し、[Applications (アプリケーション)] タブを選択して、[Application Name (アプリケーション名)] 列に表示される名前を確認します。または、イメージをまだ作成していない場合は、アプリケーションをインストールした Image Builder に接続し、Image Assistant を開きます。アプリケーション名は [Add Apps] タブに表示されます。

[Desktop (デスクトップ)] ストリームビューに対してフリートが有効になっている場合は、オペレーティングシステムのデスクトップに直接起動することもできます。これを行うには、リレー状態 URL の最後の **Desktop** の末尾に、**&app=** を指定します。

ID プロバイダー (IdP) が開始するフローでは、システムのデフォルトブラウザで、ユーザーが IdP にサインインし、IdP ユーザーポータルから AppStream 2.0 アプリケーションを選択すると、システムのデフォルトブラウザの AppStream 2.0 サインインページにリダイレクトされ、次のオプションが表示されます。

- ブラウザで続行
- AppStream 2.0 クライアントを開く

そのページで、ユーザーはセッションをブラウザで開始するか、AppStream 2.0 クライアントアプリケーションで開始するかを選択できます。必要に応じて、SAML 2.0 フェデレーションに使用するクライアントを指定することもできます。これを行うには、リレーステート URL の最後の **&client=** の末尾に、**native** または **web** を指定します。パラメータがリレーステート URL に存在する場合、対応するセッションは、ユーザーが選択することなく、指定されたクライアントで自動的に開始されます。

Note

この機能は、新しいリレーステートリージョンエンドポイント (以下の表 1) を使用してリレーステート URL を構築し、AppStream 2.0 クライアントバージョン 1.1.1300 以降を使用する場合にのみ使用できます。また、ユーザーは常にシステムのデフォルトブラウザを使用して IdP にサインインする必要があります。この機能は、デフォルト以外のブラウザを使用している場合は動作しません。

サードパーティー SAML 2.0 アイデンティティプロバイダーを使用する属性ベースのアプリケーションの使用権限を使用すると、単一のリレー状態 URL から複数のスタックへのアクセスを有効にできます。次のように、リレー状態 URL からスタックと アプリケーション (存在する場合) パラメータを削除します。

```
https://relay-state-region-endpoint?accountId=aws-account-id-without-hyphens
```

ユーザーが AppStream 2.0 アプリケーションカタログにフェデレートすると、スタックが配置されているリージョンに関連付けられたアカウント ID およびリレー状態エンドポイントについて、アプ

リケーションの使用権限が 1 つ以上のアプリケーションをユーザーに一致させたすべてのスタックが表示されます。ユーザーがカタログを選択すると、アプリケーションの使用権限には、そのユーザーが資格を持つアプリケーションのみが表示されます。

 Note

ユーザーは複数のスタックから同時にストリーミングすることはできません。

詳細については、「[サードパーティーの SAML 2.0 アイデンティティプロバイダーを使用した属性ベースのアプリケーションの使用権限](#)」を参照してください。

以下の表 1 は、AppStream 2.0 を利用できるリージョンのリリースステートエンドポイントを示しています。表 1 のリリースステートエンドポイントは、[the section called “AppStream 2.0 Web ブラウザアクセス \(バージョン 2\)”](#) および Windows クライアントアプリケーションバージョン 1.1.1300 以降と互換性があります。Windows クライアントの古いバージョンを使用している場合は、表 2 に記載されている古いリリースステートエンドポイントを使用して SAML 2.0 フェデレーションを設定する必要があります。ユーザーに FIPS 準拠の接続を使用してストリーミングするように求める場合は、FIPS 準拠のエンドポイントを使用する必要があります。FIPS エンドポイントの詳細については、[the section called “FIPS エンドポイント”](#) を参照してください。

表 1: AppStream 2.0 リリースステートリージョンエンドポイント (推奨)

リージョン	リリースステートのエンドポイント
米国東部 (バージニア北部)	<code>https://appstream2.euc-sso.us-east-1.aws.amazon.com/saml</code> (FIPS) <code>https://appstream2.euc-sso-fips.us-east-1.aws.amazon.com/saml</code>
米国東部 (オハイオ)	<code>https://appstream2.euc-sso.us-east-2.aws.amazon.com/saml</code>
米国西部 (オレゴン)	<code>https://appstream2.euc-sso.us-west-2.aws.amazon.com/saml</code> (FIPS) <code>https://appstream2.euc-sso-fips.us-west-2.aws.amazon.com/saml</code>

リージョン	リリースステートのエンドポイント
アジアパシフィック (ムンバイ)	<code>https://appstream2.euc-sso.ap-south-1.aws.amazon.com/saml</code>
アジアパシフィック (ソウル)	<code>https://appstream2.euc-sso.ap-northeast-2.aws.amazon.com/saml</code>
アジアパシフィック (シンガポール)	<code>https://appstream2.euc-sso.ap-southeast-1.aws.amazon.com/saml</code>
アジアパシフィック (シドニー)	<code>https://appstream2.euc-sso.ap-southeast-2.aws.amazon.com/saml</code>
アジアパシフィック (東京)	<code>https://appstream2.euc-sso.ap-northeast-1.aws.amazon.com/saml</code>
カナダ (中部)	<code>https://appstream2.euc-sso.ca-central-1.aws.amazon.com/saml</code>
欧州 (フランクフルト)	<code>https://appstream2.euc-sso.eu-central-1.aws.amazon.com/saml</code>
欧州 (アイルランド)	<code>https://appstream2.euc-sso.eu-west-1.aws.amazon.com/saml</code>
欧州 (ロンドン)	<code>https://appstream2.euc-sso.eu-west-2.aws.amazon.com/saml</code>
欧州 (パリ)	<code>https://appstream2.euc-sso.eu-west-3.aws.amazon.com/saml</code>

リージョン	リリーステートのエンドポイント
AWS GovCloud (米国東部)	<code>https://appstream2.euc-sso.us-gov-east-1.amazonaws-us-gov.com/saml</code> (FIPS) <code>https://appstream2.euc-sso-fips.us-gov-east-1.amazonaws-us-gov.com/saml</code>  Note AWS GovCloud (US) リージョンで AppStream 2.0 を使用する詳しい方法については、AWS GovCloud (US) ユーザーガイドの「Amazon AppStream 2.0」を参照してください。
AWS GovCloud (米国西部)	<code>https://appstream2.euc-sso.us-gov-west-1.amazonaws-us-gov.com/saml</code> (FIPS) <code>https://appstream2.euc-sso-fips.us-gov-west-1.amazonaws-us-gov.com/saml</code>  Note AWS GovCloud (US) リージョンで AppStream 2.0 を使用する詳しい方法については、AWS GovCloud (US) ユーザーガイドの「Amazon AppStream 2.0」を参照してください。
南米 (サンパウロ)	<code>https://appstream2.euc-sso.sa-east-1.amazonaws.com/saml</code>

次の表 2 は、現在も利用可能な古いリリーステートエンドポイントを示しています。ただし、表 1 にリストされている新しいリリーステートエンドポイントを使用して SAML 2.0 フェデレーションを設定することをお勧めします。特に、新しいリリーステートエンドポイントを使用すると、IdP 開始ストリーミングセッションから AppStream 2.0 クライアントアプリケーション (バージョン 1.1.1300 以降) を起動できます。表 1 の新しいリリーステートエンドポイントでは、進行中の AppStream 2.0

ストリーミングセッションに影響を与えることなく、同じウェブブラウザの異なるタブで他の AWS アプリケーションにサインインすることもできます。これは、表 2 の古いリレーステートエンドポイントではサポートされていません。詳細については、[the section called “私の AppStream 2.0 クライアントユーザーは、60 分ごとに AppStream 2.0 セッションから接続が解除されます。”](#)を参照してください。

表 2: 古い AppStream 2.0 リレーステートリージョンエンドポイント (非推奨)

リージョン	リレーステートのエンドポイント
米国東部 (バージニア北部)	https://appstream2.us-east-1.aws.amazon.com/saml (FIPS) https://appstream2-fips.us-east-1.aws.amazon.com/saml
米国東部 (オハイオ)	https://appstream2.us-east-2.aws.amazon.com/saml
米国西部 (オレゴン)	https://appstream2.us-west-2.aws.amazon.com/saml (FIPS) https://appstream2-fips.us-west-2.aws.amazon.com/saml
アジアパシフィック (ムンバイ)	https://appstream2.ap-south-1.aws.amazon.com/saml
アジアパシフィック (ソウル)	https://appstream2.ap-northeast-2.amazonaws.com/saml
アジアパシフィック (シンガポール)	https://appstream2.ap-southeast-1.amazonaws.com/saml
アジアパシフィック (シドニー)	https://appstream2.ap-southeast-2.amazonaws.com/saml
アジアパシフィック (東京)	https://appstream2.ap-northeast-1.amazonaws.com/saml

リージョン	リリースステートのエンドポイント
カナダ (中部)	<code>https://appstream2.ca-central-1.aws.amazon.com/saml</code>
欧州 (フランクフルト)	<code>https://appstream2.eu-central-1.aws.amazon.com/saml</code>
欧州 (アイルランド)	<code>https://appstream2.eu-west-1.aws.amazon.com/saml</code>
欧州 (ロンドン)	<code>https://appstream2.eu-west-2.aws.amazon.com/saml</code>
AWS GovCloud (米国東部)	<code>https://appstream2.us-gov-east-1.amazonaws-us-gov.com/saml</code> (FIPS) <code>https://appstream2-fips.us-gov-east-1.amazonaws-us-gov.com/saml</code>

 **Note**
AWS GovCloud (US) リージョンで AppStream 2.0 を使用する詳しい方法については、AWS GovCloud (US) ユーザーガイドの「[Amazon AppStream 2.0](#)」を参照してください。

リージョン	リリーステートのエンドポイント
AWS GovCloud (米国西部)	<code>https://appstream2.us-gov-west-1.amazonaws-us-gov.com/saml</code> (FIPS) <code>https://appstream2-fips.us-gov-west-1.amazonaws-us-gov.com/saml</code>  Note AWS GovCloud (US) リージョンで AppStream 2.0 を使用する詳しい方法については、AWS GovCloud (US) ユーザーガイドの「Amazon AppStream 2.0」を参照してください。
南米 (サンパウロ)	<code>https://appstream2.sa-east-1.aws.amazon.com/saml</code>

以下の表 3 に、リリーステート URL の構築に使用できるすべてのパラメータを示します。

表 3: リリーステート URL パラメータ

[Parameter] (パラメータ)	必須	形式	サポート
accountId	必須	12 文字の AWS アカウント ID	表 1 および 2 の新しいエンドポイントと古いエンドポイント
スタック	オプションです。	スタック名	表 1 および 2 の新しいエンドポイントと古いエンドポイント
アプリケーション	オプションです。	アプリケーション名または「デスクトップ」	表 1 および 2 の新しいエンドポイントと古いエンドポイント

[Parameter] (パラメータ)	必須	形式	サポート
クライアント	オプションです。	「ネイティブ」または「ウェブ」	表 1 のみに示されている新しいエンドポイント

AppStream 2.0 と SAML 2.0 の統合

以下のリンクは、AppStream 2.0 で動作するように、サードパーティーの SAML 2.0 ID プロバイダーソリューションを設定するために役立ちます。

IdP ソリューション	詳細情報
AWS IAM Identity Center	IAM Identity Center および Amazon AppStream 2.0 を使用してフェデレーションを有効にする — IAM Identity Center で既存のエンタープライズ認証情報を使用して、AppStream 2.0 アプリケーションへのユーザーアクセスをフェデレートする方法について説明します。
Windows Server 用 Active Directory フェデレーションサービス (AD FS)	GG4L ウェブサイトの AppStream — 既存のエンタープライズ認証情報を使用して、ユーザーに AppStream 2.0 への SSO アクセスを提供する方法について説明します。AD FS 3.0 を使用して、AppStream 2.0 のフェデレーテッドアイデンティティを設定できます。
Azure Active Directory (Azure AD)	Azure AD シングルサインオンと Amazon AppStream 2.0 でのフェデレーションの有効化 — エンタープライズアプリケーションで Azure AD SSO を使用して、Amazon AppStream 2.0 向けにフェデレーテッドユーザーのアクセスを設定する方法を説明します。
GG4L School Passport™	GG4L の School Passport™ と Amazon AppStream 2.0 を使用した ID フェデレーションの有効化 — GG4L の School Passport™ を設定して Amazon AppStream 2.0 へのログインをフェデレートする方法について説明します。

IdP ソリューション	詳細情報
Google	Amazon AppStream 2.0 を使用した G Suite SAML 2.0 フェデレーションのセットアップ — G Suite 管理コンソールを使用して、G Suite ドメインのユーザー向けに SAML フェデレーションを AppStream 2.0 に設定します。
Okta	Amazon AppStream 2.0 用に SAML 2.0 を設定する方法 — Okta を使用して AppStream 2.0 に SAML フェデレーションを設定する方法を説明します。ドメインに結合されているスタックについては、[Application username format] を [AD user principal name] に設定する必要があります。
Ping Identity	Amazon AppStream 2.0 に SSO 接続を設定する — AppStream 2.0 にシングルサインオン (SSO) を設定する方法を説明しています。

IdP ソリューション	詳細情報
Shibboleth	シングルサインオン: 統合 AWS、OpenLDAP、および Shibboleth — Shibboleth IdP と 間の初期フェデレーションを設定する方法について説明します AWS Management Console。AppStream 2.0 にフェデレーションを有効化するには、次の追加ステップを完了する必要があります。 AWS セキュリティホワイトペーパーのステップ 4 では、フェデレーティッドユーザーの AWS Management Console に対するアクセス許可を定義するために、IAM ロールを作成する方法を説明しています。これらのロールを作成して、ホワイトペーパーに示されるようにインラインポリシーを埋め込んだ後、フェデレーティッドユーザーが AppStream 2.0 スタックのみにアクセスできる許可が付与されるようにこのポリシーを編集します。これを実行するには、既存のポリシーを SAML のセットアップ の Step 3: Embed an Inline Policy for the IAM Role でメモしたポリシーに置き換えます。 SAML のセットアップ の Step 6: Configure the Relay State of Your Federation で説明するようにスタックリリースステート URL を追加するとき、ターゲットのリクエスト属性としてリリースステートパラメータをフェデレーション URL に追加します。URL はエンコードする必要があります。リリースステートのパラメータの設定に関する詳細は、Shibboleth ドキュメンテーションの「SAML 2.0」を参照してください。 詳細については、「Enabling Identity Federation with Shibboleth and Amazon AppStream 2.0」を参照してください。

IdP ソリューション	詳細情報
VMware Workspace ONE	VMware Workspace ONE から Amazon AppStream 2.0 へのアクセスのフェデレーション — VMware Workspace ONE プラットフォームを使用して、AppStream 2.0 アプリケーションへのユーザーアクセスのフェデレーションを行う方法について説明します。
SimpleSAMLphp	Enabling Federation with SimpleSAMLphp and Amazon AppStream 2.0 – SimpleSamlPHP を使用して AppStream 2.0 の SAML 2.0 フェデレーションを設定する方法を説明します。
OneLogin Single Sign-On (SSO)	OneLogin SSO with Amazon AppStream 2.0 – OneLogin SSO を使用して AppStream 2.0 のフェデレーテッドユーザーアクセスを設定する方法を説明します。
JumpCloud Single Sign-On (SSO)	Enable federation with JumpCloud SSO and Amazon AppStream 2.0 – JumpCloud SSO を使用して AppStream 2.0 のフェデレーテッドユーザーアクセスを設定する方法を説明します。
BIO-key PortalGuard	BIO-key PortalGuard および Amazon AppStream 2.0 でフェデレーションを有効にする — AppStream 2.0 へのフェデレーションログイン用に BIO-key PortalGuard を構成する方法について説明します。

発生する可能性がある一般的な問題の解決策については、「[トラブルシューティング](#)」を参照してください。

追加のサポートされている SAML プロバイダーの詳細については、IAM ユーザーガイドの「[サードパーティの SAML ソリューションプロバイダーと AWS の統合](#)」を参照してください。

AppStream 2.0 でのアクティブディレクトリの使用

Amazon AppStream 2.0 の常時オンおよびオンデマンド Windows フリートと Image Builder を Microsoft アクティブディレクトリのドメインに参加させ、既存のアクティブディレクトリドメイン (クラウドベースまたはオンプレミス) を使用して、ドメイン参加済みのストリーミングインスタンスを起動することができます。Managed AWS Microsoft AD とも呼ばれる を使用して AWS Directory Service for Microsoft Active Directory Active Directory ドメインを作成し、それを使用して AppStream 2.0 リソースをサポートすることもできます。AWS Managed Microsoft AD の使用に関する詳細については、AWS Directory Service 管理ガイドで [Microsoft アクティブディレクトリ](#) について参照してください。

Note

Amazon Linux 2 フリート、Image Builder、Elastic フリート、App Block Builder は現在、ドメイン参加をサポートしていません。

Amazon AppStream 2.0 を アクティブディレクトリドメインに参加させることによって、以下を実行することができます。

- ストリーミングセッションからプリンターやファイル共有などのアクティブディレクトリリソースにアクセスすることをユーザーとアプリケーションに許可する。
- グループポリシーマネジメントコンソール (GPMC) で使用できるグループポリシー設定を使用して、エンドユーザーエクスペリエンスを定義する。
- アクティブディレクトリログイン認証情報を使用した認証をユーザーに義務付けるアプリケーションをストリーミングする。
- エンタープライズコンプライアンスとセキュリティポリシーを Amazon AppStream 2.0 ストリーミングインスタンスに適用する。

内容

- [アクティブディレクトリドメインの概要](#)
- [Amazon AppStream 2.0 でアクティブディレクトリの使用を開始する前に](#)
- [チュートリアル: アクティブディレクトリのセットアップ](#)
- [証明書ベースの認証](#)

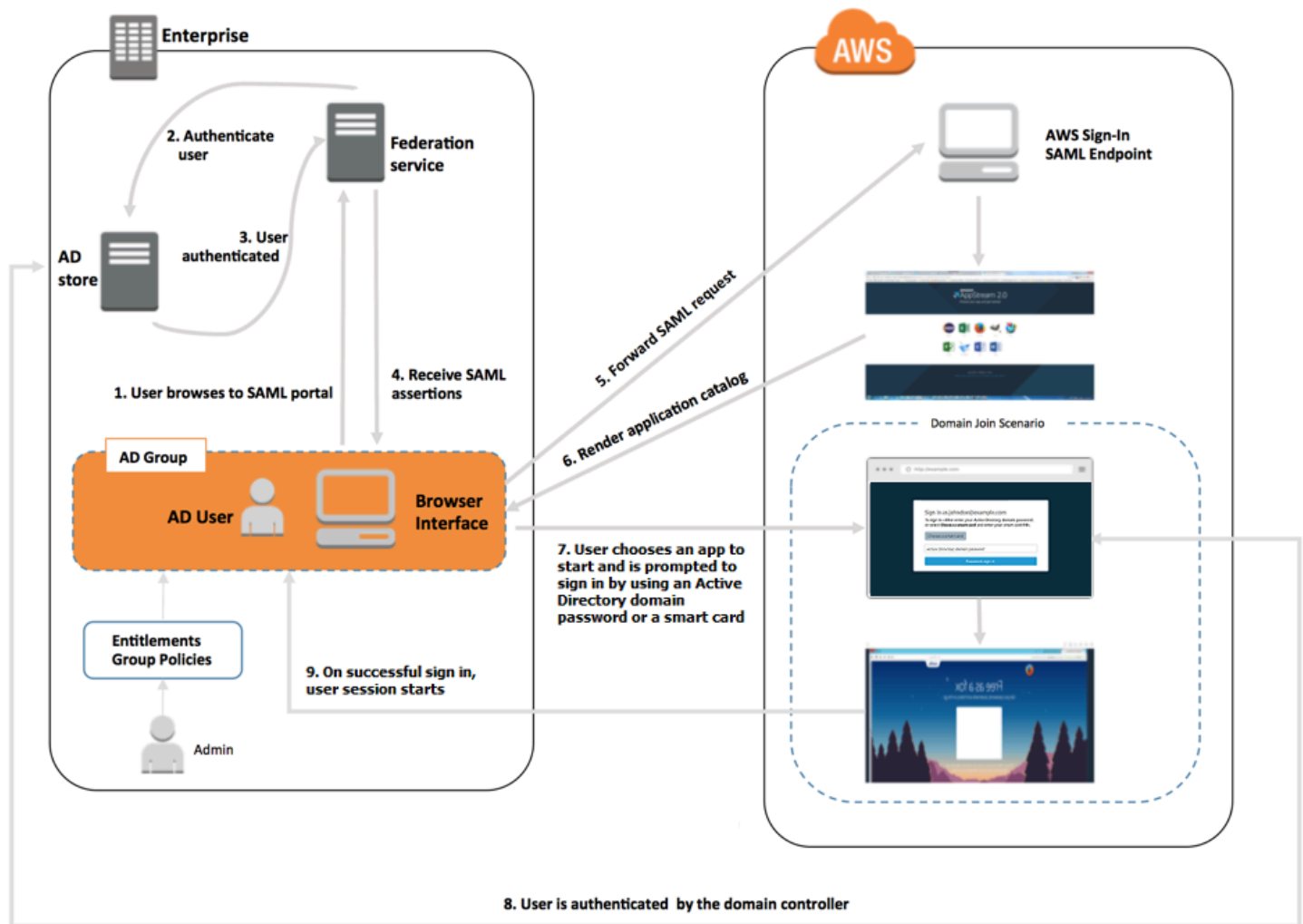
- [AppStream 2.0 のアクティブディレクトリ管理](#)
- [詳細情報](#)

アクティブディレクトリドメインの概要

Amazon AppStream 2.0 でアクティブディレクトリドメインを使用するには、それらの仕組みと、完了する必要がある設定タスクを理解する必要があります。次のタスクを実行する必要があります。

1. 必要に応じて、アプリケーションのエンドユーザーエクスペリエンスとセキュリティ要件を定義できるように、グループポリシーを設定する。
2. Amazon AppStream 2.0 でドメイン参加済みのアプリケーションスタックを作成する。
3. SAML 2.0 ID プロバイダーで AppStream 2.0 アプリケーションを作成し、直接またはアクティブディレクトリグループを使用してエンドユーザーに割り当てる。

ドメインに対してユーザーを認証する場合は、これらのユーザーが AppStream 2.0 ストリーミングセッションを開始する際に複数のステップを行う必要があります。以下の図は、SAML 経由での最初のブラウザリクエストからアクティブディレクトリ認証までのエンドツーエンドのユーザー認証フローを示しています。



ユーザー認証フロー

1. ユーザーが `https://applications.exampleco.com` を参照します。サインインページがユーザーの認証をリクエストします。
2. フェデレーションサービスが組織の ID ストアからの認証をリクエストします。
3. ID ストアはユーザーを認証し、フェデレーションサービスに認証レスポンスを返します。
4. 認証が成功すると、フェデレーションサービスはユーザーのブラウザに SAML アサーションを送信します。
5. ユーザーのブラウザは、SAML アサーションを AWS サインイン SAML エンドポイント (`https://signin.aws.amazon.com/saml`) に投稿します。AWS サインインは SAML リクエストを受け取り、リクエストを処理し、ユーザーを認証し、認証トークンを AppStream 2.0 サービスに転送します。

6. からの認証トークンを使用して AWS、AppStream 2.0 はユーザーを認可し、ブラウザにアプリケーションを表示します。
7. ユーザーはアプリケーションを選択し、AppStream 2.0 スタックで有効になっている Windows ログイン認証方法に応じて、アクティブディレクトリドメインパスワードを入力するか、スマートカードを選択するよう求められます。両方の認証方法が有効になっている場合、ユーザーはドメインパスワードを入力するか、スマートカードを使用するかを選択できます。証明書ベースの認証は、プロンプトを省略してユーザーの認証にも使用できます。
8. ドメインコントローラーに接続してユーザーを認証します。
9. ドメインで認証された後、ユーザーのセッションがドメインに接続できる状態で開始されます。

ユーザーの視点から見ると、このプロセスは透過的です。ユーザーはまず組織の内部ポータルに移動し、AWS 認証情報を入力することなく AppStream 2.0 アプリケーションポータルにリダイレクトされます。アクティブディレクトリドメインのパスワードまたはスマートカードの認証情報のみが必要です。

ユーザーがこのプロセスを開始する前に、必要な資格およびグループポリシーを使用してアクティブディレクトリを設定し、ドメイン参加済みのアプリケーションスタックを作成する必要があります。

Amazon AppStream 2.0 でアクティブディレクトリの使用を開始する前に

AppStream 2.0 で Microsoft アクティブディレクトリドメインを使用する前に、次の要件と考慮事項に注意してください。

内容

- [アクティブディレクトリドメイン環境](#)
- [ドメイン参加済みの AppStream 2.0 ストリーミングインスタンス](#)
- [グループポリシー設定](#)
- [スマートカード認証](#)

アクティブディレクトリドメイン環境

Active Directory ドメイン環境は、以下の要件を満たしている必要があります。

- ストリーミングインスタンスに参加させる Microsoft アクティブディレクトリドメインが必要です。Active Directory ドメインがない場合、またはオンプレミスの Active Directory 環境を使用する場合は、「[AWS パートナーソリューションデプロイガイド](#)」の「[Active Directory ドメインサービス](#)」を参照してください。
- AppStream 2.0 で使用するドメインでコンピュータオブジェクトを作成および管理するためのアクセス許可を持つドメインサービスアカウントが必要です。詳細については、Microsoft ドキュメントで [How to Create a Domain Account in Active Directory](#) を参照してください。

このアクティブディレクトリドメインを AppStream 2.0 に関連付けるときは、サービスアカウント名とパスワードを入力します。AppStream 2.0 はこのアカウントを使用して、ディレクトリ内にコンピュータオブジェクトを作成して管理します。詳細については、「[アクティブディレクトリコンピュータオブジェクトを作成および管理するための許可の付与](#)」を参照してください。

- AppStream 2.0 にアクティブディレクトリドメインを登録するときは、組織単位 (OU) 識別名を提供する必要があります。この目的のために OU を作成します。デフォルトのコンピュータコンテナは OU ではなく、AppStream 2.0 で使用することはできません。詳細については、「[組織単位の識別子名を検索する](#)」を参照してください。
- AppStream 2.0 で使用予定のディレクトリは、完全修飾ドメイン名 (FQDN) を使用して、ストリーミングインスタンスを起動する Virtual Private Cloud (VPC) 経由でアクセスできる必要があります。詳細については、Microsoft ドキュメントの [Active Directory and Active Directory Domain Services Port Requirements](#) を参照してください。

ドメイン参加済みの AppStream 2.0 ストリーミングインスタンス

ドメイン参加済みの常時オンおよびオンデマンドフリートからのアプリケーションのストリーミングには SAML 2.0 ベースのユーザーフェデレーションが必要です。[CreateStreamingURL](#) または AppStream 2.0 ユーザープールを使用して、ドメイン参加済みインスタンスに対してセッションを起動することはできません。

また、Image Builder とフリートのアクティブディレクトリドメインへの参加をサポートするイメージを使用する必要があります。2017 年 7 月 24 日以降に公開されたすべてのパブリックイメージはアクティブディレクトリドメインへの参加をサポートします。詳細については、[AppStream 2.0 ベースイメージとマネージド型イメージの更新に関するリリースノート](#)および[チュートリアル: アクティブディレクトリのセットアップ](#)を参照してください。

 Note

Active Directory ドメインに結合できるのは、Windows の常時オンおよびオンデマンドフリーストリートのストリーミングインスタンスのみです。

グループポリシー設定

次のグループポリシー設定の内容を確認します。AppStream 2.0 がドメインユーザーの認証とログインをブロックしないように、必要に応じて、次のセクションで説明するように設定を更新します。更新しないと、ユーザーが AppStream 2.0 にログインしようとしたときに、ログインに失敗する場合があります。「不明なエラーが発生しました」というエラーメッセージが表示される場合があります。

- [Computer Configuration] (コンピュータの構成) > [Administrative Templates] (管理用テンプレート) > [Windows Components] (Windows コンポーネント) > [Windows Logon Options] (Windows ログオンオプション) > [Disable or Enable software Secure Attention Sequence] (ソフトウェアの Secure Attention Sequence を無効または有効にする) から、[Services] (サービス) に対して [Enabled] (有効) に設定します。
- コンピュータ設定 > 管理テンプレート > システム > ログオン > 認証情報プロバイダーを除外 — 次の CLSID が表示されていないことを確認します。e7c1bab5-4b49-4e64-a966-8d99686f8c7c f148bAed-5f7f-40c9-8D48-51e24e571825
- [Computer Configuration (コンピュータの構成)] > [Policies (ポリシー)] > [Windows Settings (Windows 設定)] > [Security Settings (セキュリティ設定)] > [Local Policies (ローカルポリシー)] > [Security Options (セキュリティオプション)] > [Interactive Logon (対話型ログオン)] > [Interactive Logon (対話型ログオン)]: ログオンしようとしているユーザーへのメッセージテキストから、この値を [Not defined (未定義)] に設定します。
- [Computer Configuration (コンピュータの構成)] > [Policies (ポリシー)] > [Windows Settings (Windows 設定)] > [Security Settings (セキュリティ設定)] > [Local Policies (ローカルポリシー)] > [Security Options (セキュリティオプション)] > [Interactive Logon (対話型ログオン)] > [Interactive Logon (対話型ログオン)]: ログオンしようとしているユーザーへのメッセージタイトルから、この値を [Not defined (未定義)] に設定します。
- コンピュータ設定 > ポリシー > Windows 設定 > セキュリティ設定 > ローカルポリシー > ユーザー権限の割り当て > ローカルでログオンを許可する — これを未定義に設定するか、ドメインユーザー/グループをこのリストに追加します。

- コンピュータ設定 > ポリシー > Windows 設定 > セキュリティ設定 > ローカルポリシー > ユーザー権限の割り当て > ローカルでログオンを拒否 — これを未定義に設定するか、ドメインユーザー/グループがリストに含まれていないことを確認します。

マルチセッションフリートを使用している場合は、上記で指定した設定に加えて、次のグループポリシー設定も必要です。

- コンピュータ設定 > ポリシー > Windows 設定 > セキュリティ設定 > ローカルポリシー > ユーザー権限の割り当て > リモートデスクトップサービスを介したログオンを許可する — これを未定義に設定するか、ドメインユーザー/グループをこのリストに追加します。
- コンピュータ設定 > ポリシー > Windows 設定 > セキュリティ設定 > ローカルポリシー > ユーザー権限の割り当て > リモートデスクトップサービスを介したログオンの拒否 — これを未定義に設定するか、ドメインユーザー/グループがリストに含まれていないことを確認します。

スマートカード認証

AppStream 2.0 では、アクティブディレクトリドメインパスワード、または AppStream 2.0 ストリーミングインスタンスへの Windows サインイン用の[共通アクセスカード \(CAC\)](#) や[個人識別検証 \(PIV\)](#) スマートカードなどのスマートカードの使用をサポートしています。サードパーティーの証明機関 (CA) を使用してスマートカードサインインを有効にするようにアクティブディレクトリ環境を構成する詳細方法については、Microsoft ドキュメントの[Guidelines for enabling smart card logon with third-party certification authorities](#) を参照してください。

Note

AppStream 2.0 は、ユーザーがストリーミングインスタンスにサインインした後でのセッション内認証用のスマートカードの使用もサポートしています。この機能がサポートされるのは、Windows バージョン 1.1.257 以降向けの AppStream 2.0 クライアントをインストールしたユーザーのみです。その他の要件については、[スマートカード](#) を参照してください。

チュートリアル: アクティブディレクトリのセットアップ

AppStream 2.0 でアクティブディレクトリを使用するには、まず AppStream 2.0 にディレクトリ Config オブジェクトを作成して、ディレクトリ設定を登録する必要があります。このオブジェクトには、ストリーミングインスタンスをアクティブディレクトリドメインに参加させるために必要な情報が含まれています。Directory Config オブジェクトを作成するには、AppStream 2.0 マネジメント

コンソール、AWS SDK、または [AWS CLI](#) を使用します。その後、ディレクトリ設定を使用して、ドメイン参加済みの常時オンおよびオンデマンドフリートと Image Builder を起動できます。

Note

アクティブディレクトリドメインに参加させることができるのは、常時オンおよびオンデマンドフリートのストリーミングインスタンスのみです。

タスク

- [ステップ 1: Directory Config オブジェクトを作成する](#)
- [ステップ 2: ドメイン結合 Image Builder を使用してイメージを作成する](#)
- [ステップ 3: ドメイン結合フリートを作成する](#)
- [ステップ 4: SAML 2.0 を設定する](#)

ステップ 1: Directory Config オブジェクトを作成する

AppStream 2.0 で作成した Directory Config オブジェクトは、後のステップで使用されます。

AWS SDK を使用している場合は、[CreateDirectoryConfig](#) オペレーションを使用できます。を使用している場合は AWS CLI、[create-directory-config](#) コマンドを使用できます。

AppStream 2.0 コンソールを使用して Directory Config オブジェクトを作成するには

1. AppStream 2.0 コンソール (<https://console.aws.amazon.com/appstream2>) を開きます。
2. ナビゲーションペインで、[Directory Configs]、[Create Directory Config] (Directory Config の作成) の順に選択します。
3. [ディレクトリ名] に、アクティブディレクトリドメインの完全修飾ドメイン名 (FQDN) (例: corp.example.com) を入力します。各リージョンは、特定のディレクトリ名を持つ [Directory Config] 値を 1 つのみ持つことができます。
4. [Service Account Name] (サービスアカウント名) に、コンピュータオブジェクトを作成でき、ドメインを結合するアクセス許可を持つアカウント名を入力します。詳細については、「[アクティブディレクトリコンピュータオブジェクトを作成および管理するための許可の付与](#)」を参照してください。アカウント名は「DOMAIN\username」の形式である必要があります。
5. [パスワード] と [パスワードの確認] に、指定されたアカウントのディレクトリパスワードを入力します。

6. [Organizational Unit (OU)] に、少なくとも 1 つのストリーミングインスタンスコンピュータオブジェクトの OU 識別名を入力します。

 Note

OU 名にスペースを含めることはできません。スペースを含む OU 名を指定した場合、フリートまたは Image Builder がアクティブディレクトリドメインに再参加しようすると、AppStream 2.0 はコンピュータオブジェクトを正しく循環できず、ドメインの再参加は成功しません。この問題のトラブルシューティング方法については、[Active Directory ドメイン参加](#) の「アカウントが既に存在します」というメッセージで DOMAIN_JOIN_INTERNAL_SERVICE_ERROR トピックを参照してください。さらに、デフォルトのコンピュータコンテナは OU ではなく、AppStream 2.0 で使用することはできません。詳細については、「[組織単位の識別子名を検索する](#)」を参照してください。

7. 複数の OU を追加するには、[Organizational Unit (OU)] (部門名 (UI)) の横にあるプラス記号 (+) を選択します。OU を削除するには、[x] アイコンを選択します。
8. [Next (次へ)] を選択します。
9. 設定情報を確認して、[Create] を選択します。

ステップ 2: ドメイン結合 Image Builder を使用してイメージを作成する

次に、AppStream 2.0 Image Builder を使用して、アクティブディレクトリドメイン参加機能を持つ新しいイメージを作成します。フリートとイメージは、異なるドメインのメンバーにすることができます。Image Builder をドメインに結合してドメイン結合を有効にし、アプリケーションをインストールします。フリートドメイン結合については、次のセクションで説明します。

ドメイン結合フリートを起動するためのイメージを作成するには

1. 「[チュートリアル: AppStream 2.0 コンソールを使用してカスタム AppStream 2.0 イメージを作成する](#)」の手順に従います。
2. ベースイメージの選択ステップでは、2017 年 7 月 24 日以降にリリースされたベース AWS イメージを使用します。リリースされた AWS イメージの最新リストについては、「」を参照してください。[AppStream 2.0 ベースイメージとマネージド型イメージの更新に関するリリースノート](#)。

3. [Step 3: Configure Network] で、アクティブディレクトリ環境へのネットワーク接続がある VPC およびサブネットを選択します。VPC サブネットを使用してディレクトリにアクセスできるようにセットアップされたセキュリティグループを選択します。
4. また、[Step 3: Configure Network] (ステップ 3: ネットワークの設定) で、[Active Directory Domain (Optional)] (アクティブディレクトリドメイン (オプション)) セクションを展開し、Image Builder を参加させる [Directory Name] (ディレクトリ名) と [Directory OU] (ディレクトリ OU) の値を選択します。
5. Image Builder 設定を確認して、[Create] を選択します。
6. 新しい Image Builder が [Running] 状態になるまで待機してから、[Connect] を選択します。
7. 管理者モード、またはローカル管理者権限を持つディレクトリユーザーとして Image Builder にログインします。詳細については、「[Image Builder のローカル管理者権限を付与する](#)」を参照してください。
8. アプリケーションをインストールし、新しいイメージを作成するには、「[チュートリアル: AppStream 2.0 コンソールを使用してカスタム AppStream 2.0 イメージを作成する](#)」で説明されているステップを行います。

ステップ 3: ドメイン結合フリートを作成する

前のステップで作成したプライベートイメージを使用して、アプリケーションをストリーミングするためのアクティブディレクトリドメイン参加済みの常時オンおよびオンデマンドフリートを作成します。ドメインは Image Builder でイメージを作成するために使用されたものと別のものにできます。

ドメイン結合の常時オンまたはオンデマンドフリートを作成する

1. 「[Amazon AppStream 2.0 でフリートを作成する](#)」の手順に従います。
2. イメージ選択ステップで、前のステップ ([ステップ 2: ドメイン結合 Image Builder を使用してイメージを作成する](#)) で作成したイメージを使用します。
3. Step 4: Configure Network] で、アクティブディレクトリ環境へのネットワーク接続がある VPC およびサブネットを選択します。ドメインと通信できるようにセットアップされたセキュリティグループを選択します。
4. また、[Step 4: Configure Network] (ステップ 4: ネットワークの設定) で、[Active Directory Domain (Optional)] (アクティブディレクトリドメイン (オプション)) セクションを展開し、フリートを参加させる [Directory Name] (ディレクトリ名) と [Directory OU] (ディレクトリ OU) の値を選択します。
5. フリート設定を確認して、[Create] を選択します。

6. フリートをスタックに関連付けて実行するには、「[Amazon AppStream 2.0 フリートとスタックを作成する](#)」の残りのステップを実行します。

ステップ 4: SAML 2.0 を設定する

ユーザーは、SAML 2.0 ベースの ID フェデレーション環境を使用して、ドメイン参加済みフリートからストリーミングセッションを起動する必要があります。

シングルサインオンアクセス用の SAML 2.0 を設定するには

1. 「[SAML のセットアップ](#)」の手順に従います。
2. AppStream 2.0 では、ログインしているユーザーの SAML_Subject NameID 値を次のいずれかの形式で指定する必要があります。

- *domain*\username (sAMAccountName を使用)
- username@domain.com (userPrincipalName を使用)

sAMAccountName 形式を使用している場合、NetBIOS 名または完全修飾ドメイン名 (FQDN) を使用して *domain* を指定できます。

3. アクティブディレクトリユーザーまたはグループへのアクセスを提供して、AppStream 2.0 スタックに ID プロバイダアプリケーションポータルからアクセスできるようにします。
4. 「[SAML のセットアップ](#)」の残りのステップを行います。

SAML 2.0 を使用してユーザーにログインさせるには

1. SAML 2.0 プロバイダーのアプリケーションカタログにログインし、前の手順で作成した AppStream 2.0 SAML アプリケーションを開きます。
2. AppStream 2.0 アプリケーションカタログが表示されたら、起動するアプリケーションを選択します。
3. ロード中アイコンが表示されたら、パスワードの入力を求められます。SAML 2.0 ID プロバイダーから提供されたドメインユーザー名がパスワードフィールドの上に表示されます。パスワードを入力して、[ログイン] を選択します。

ストリーミングインスタンスで Windows ログイン手順を実行すると、選択したアプリケーションが開きます。

証明書ベースの認証

Microsoft Active Directory に結合している AppStream 2.0 フリートでは、証明書ベースの認証を使用できます。これにより、ユーザーがログインするときに Active Directory ドメインパスワードの入力を求めるユーザープロンプトが省略されます。Active Directory ドメインで証明書ベースの認証を使用すると、以下のことを行うことができます。

- SAML 2.0 ID プロバイダーに依頼してユーザーを認証し、Active Directory 内のユーザーと一致する SAML アサーションを提供する。
- ユーザープロンプトの回数を減らして、シングルサインオンでログオンできるようにする。
- SAML 2.0 ID プロバイダーを使用して、パスワードなしの認証フローを有効にする。

証明書ベースの認証では、AWS で Private Certificate Authority (AWS Private CA) リソースを使用します AWS アカウント。AWS プライベート CA を使用すると、ルート CA と下位 CAs を含むプライベート認証機関 (CA) 階層を作成できます。独自の CA 階層を作成し、そこから内部ユーザーを認証する証明書を発行することもできます。詳細については、[「とは AWS Private CA」](#)を参照してください。

証明書ベースの認証に AWS プライベート CA を使用すると、AppStream 2.0 は、各 AppStream 2.0 フリートインスタンスのセッション予約時にユーザーの証明書を自動的にリクエストします。証明書でプロビジョニングされた仮想スマートカードを使用して、ユーザーを Active Directory に対して認証します。

証明書ベースの認証 (CBA) は、Windows インスタンスを実行する AppStream 2.0 ドメイン参加フリート (シングルセッションフリートとマルチセッションフリートの両方) でサポートされています。マルチセッションフリートで CBA を有効にするには、02-07-2025 以降にリリースされた AppStream 2.0 エージェントを使用する AppStream 2.0 イメージを使用する必要があります。または、以降にリリースされたマネージド AppStream 2.0 イメージ更新をイメージで使用する必要があります02-11-2025。

内容

- [前提条件](#)
- [証明書ベースの認証](#)
- [証明書ベースの認証の管理](#)
- [クロスアカウント PCA 共有を有効にする](#)

前提条件

証明書ベースの認証を使用する前に、以下のステップを完了します。

1. ドメイン結合フリートを設定し、SAML 2.0 を設定する。必ず SAML_Subject NameID 用の `username@domain.com userPrincipalName` 形式を使用してください。詳細については、「[the section called “ステップ 5: SAML 認証レスポンスのアサーションを作成する”](#)」を参照してください。

Note

証明書ベースの認証を使用する場合は、スタック内の Active Directory のスマートカードサインインを有効にしないでください。詳細については、「[the section called “スマートカード”](#)」を参照してください。

2. イメージでは AppStream 2.0 エージェントのバージョン 10-13-2022 以降を使用してください。詳細については、「[the section called “イメージを最新の状態に保つ”](#)」を参照してください。
3. SAML アサーションの ObjectSid 属性を設定します。この属性を使用して、Active Directory ユーザーとの強力なマッピングを実行できます。ObjectSid 属性が SAML_Subject NameID で指定されたユーザーの Active Directory セキュリティ識別子 (SID) と一致しない場合、証明書ベースの認証は失敗します。詳細については、「[the section called “ステップ 5: SAML 認証レスポンスのアサーションを作成する”](#)」を参照してください。ObjectSid は、2025 年 9 月 10 日以降の証明書ベースの認証に必須です。詳細については、[KB5014754: Windows ドメインコントローラーでの証明書ベースの認証の変更](#)」を参照してください。
4. SAML 2.0 設定で使用する IAM ロールの信頼ポリシーに `sts:TagSession` アクセス権を追加します。詳細については、「[AWS STSでのセッションタグの受け渡し](#)」を参照してください。この権限は、証明書ベースの認証を使用する場合に必要です。詳細については、「[the section called “ステップ 2: SAML 2.0 フェデレーション IAM ロールを作成する”](#)」を参照してください。
5. プライベート CA を使用してプライベート認証機関 (CA) AWS を作成します。Active Directory で設定されていない場合 AWS は、証明書ベースの認証を使用する必要があります。詳細については、「[AWS Private CA デプロイの計画](#)」を参照してください。証明書ベースの認証の多くのユースケースでは、次の AWS プライベート CA 設定が一般的です。
 - CA タイプオプション
 - 使用期間が短い証明書 CA 使用モード – CA が証明書ベースの認証のためにエンドユーザー証明書のみを発行する場合に推奨されます。
 - ルート CA を含む単一レベルの階層 – 下位 CA を選択して既存の CA 階層と統合します。

- 主要なアルゴリズムオプション – RSA 2048
- サブジェクト識別名オプション – 最も適切なオプションを使用して、Active Directory の信頼されたルート証明機関ストアでこの CA を識別します。
- 証明書失効オプション – CRL のディストリビューション

 Note

証明書ベースの認証には、AppStream 2.0 フリートインスタンスとドメインコントローラーの両方からアクセスできるオンライン CRL ディストリビューションポイントが必要です。これには、AWS プライベート CA CRL エントリ用に設定された Amazon S3 バケットへの認証されていないアクセスが必要です。パブリックアクセスをブロックする場合は Amazon S3 バケットにアクセスできる CloudFront ディストリビューションが必要です。これらのオプションの詳細については、「[証明書失効リスト \(CRL\) の計画](#)」を参照してください。

6. プライベート CA に、AppStream 2.0 の証明書ベースの認証で使用する CA を指定するために、`euc-private-ca` というタイトルを持つキーをタグ付けします。このキーには値は必要ありません。詳細については、「[プライベート CA のタグ管理](#)」を参照してください。AppStream 2.0 でのリソースにアクセス許可を付与するために使用する AWS 管理ポリシーの詳細については AWS アカウント、「」を参照してください [the section called “AWS AppStream 2.0 リソースにアクセスするために必要な管理ポリシー”](#)。
7. 証明書ベースの認証では、仮想スマートカードを使用してログオンします。詳細については、「[サードパーティーの証明機関でスマートカードオンを有効にするためのガイドライン](#)」を参照してください。以下の手順に従ってください。
 - a. スマートカードユーザーを認証するには、ドメインコントローラー証明書を使用してドメインコントローラーを設定します。Active Directory 証明書サービスのエンタープライズ CA が Active Directory に設定されている場合、スマートカードによるログオンを可能にするドメインコントローラーが証明書に自動的に登録されます。Active Directory 証明書サービスがない場合は、「[サードパーティー CA からのドメインコントローラー証明書の要件](#)」を参照してください。は、Active Directory エンタープライズ認証機関がドメインコントローラー証明書の登録を自動的に管理することを AWS 推奨しています。

 Note

AWS Managed Microsoft AD を使用する場合は、ドメインコントローラー証明書の要件を満たす Amazon EC2 インスタンスで証明書サービスを設定できます。 [Active](#)

[Directory Certificate Services で設定された Managed Microsoft AD のデプロイ例については、「新しい Amazon Virtual Private Cloud に Active Directory をデプロイする」](#)を参照してください。AWS

AWS Managed Microsoft AD と Active Directory Certificate Services では、コントローラーの VPC セキュリティグループから Certificate Services を実行する Amazon EC2 インスタンスへのアウトバウンドルールも作成する必要があります。証明書の自動登録を有効にするには、セキュリティグループに TCP ポート 135 とポート 49152 ~ 65535 へのアクセスを提供する必要があります。Amazon EC2 インスタンスは、ドメインコントローラーを含むドメインインスタンスからの同じポートへのインバウンドアクセスも許可する必要があります。AWS Managed Microsoft AD のセキュリティグループを見つける方法の詳細については、「VPC [サブネットとセキュリティグループを設定する](#)」を参照してください。

- b. AWS プライベート CA コンソール、または SDK または CLI で、プライベート CA 証明書をエクスポートします。詳細については、「[プライベート証明書のエクスポート](#)」を参照してください。
- c. プライベート CA を Active Directory に公開します。ドメインコントローラーまたはドメイン結合マシンにログオンします。プライベート CA 証明書を任意の `<path>\<file>` にコピーし、ドメイン管理者として次のコマンドを実行します。また、グループポリシーと Microsoft PKI Health ツール (PKIView) を使用して CA を公開することもできます。詳細については、「[設定手順](#)」を参照してください。

```
certutil -dspublish -f <path>\<file> RootCA
```

```
certutil -dspublish -f <path>\<file> NTAAuthCA
```

コマンドが正常に完了したことを確認してから、プライベート CA 証明書ファイルを削除します。Active Directory のレプリケーション設定によっては、CA がドメインコントローラーと AppStream 2.0 フリートインスタンスに公開するまでに数分かかる場合があります。

 Note

Active Directory は、AppStream 2.0 フリートインスタンスがドメインに結合したときに、信頼されたルート認証機関とエンタープライズ NTAAuth ストアに CA を自動的に配布する必要があります。

Windows オペレーティングシステムの場合、CA (認証局) のディストリビューションは自動的に行われます。ただし、Rocky Linux および Red Hat Enterprise Linux の場合は、AppStream 2.0 Directory Config が使用する CA からルート CA 証明書 (複数可) をダウンロードする必要があります。KDC ルート CA 証明書が異なる場合、それらもダウンロードする必要があります。証明書ベースの認証を使用する前に、これらの証明書をイメージまたはスナップショットにインポートする必要があります。

イメージには、`/etc/sss/pki/sss_auth_ca_db.pem` という名前のファイルが必要です。次のようになります。

```
-----BEGIN CERTIFICATE-----
Base64-encoded certificate chain from ACM Private CA
-----END CERTIFICATE-----
-----BEGIN CERTIFICATE-----
Base64-encoded certificate body from ACM private CA
-----END CERTIFICATE-----
-----BEGIN CERTIFICATE-----
Base64-encoded root CA KDC certificate chain
-----END CERTIFICATE-----
```

Note

リージョンやアカウント間でイメージをコピーする場合、またはイメージを新しい Active Directory に再関連付けする場合は、このファイルを Image Builder の関連する証明書で再設定し、使用前に再度スナップショットを作成する必要があります。

ルート CA 証明書をダウンロードする手順は次のとおりです。

1. Image Builder で、`/etc/sss/pki/sss_auth_ca_db.pem` という名前のファイルを作成します。
2. [AWS Private CA コンソール](#) を開きます。
3. AppStream 2.0 Directory Config で使用されるプライベート証明書を選択します。
4. [CA 証明書] タブを選択します。
5. 証明書チェーンと証明書本文を Image Builder の `/etc/sss/pki/sss_auth_ca_db.pem` にコピーします。

KDC が使用するルート CA 証明書が AppStream 2.0 Directory Config が使用するルート CA 証明書と異なる場合は、以下のサンプルステップに従ってダウンロードします。

1. Image Builder と同じドメインに参加している Windows インスタンスに接続します。
2. certlm.msc を開きます。
3. 左側のペインで、[信頼できるルート証明機関] を選択し、[証明書] を選択します。
4. ルート CA 証明書ごとに、コンテキスト (右クリック) メニューを開きます。
5. [すべてのタスク] を選択し、[エクスポート] を選択して証明書エクスポートウィザードを開き、[次へ] を選択します。
6. [Base64 でエンコードされた X.509 (.CER)] を選択し、[次へ] を選択します。
7. [参照] を選択し、ファイル名を入力し、[次へ] を選択します。
8. [Finish] を選択してください。
9. エクスポートされた証明書をテキストエディタで開きます。
10. ファイルの内容を Image Builder の `/etc/sss/pki/sss_auth_ca_db.pem` にコピーします。

証明書ベースの認証

証明書ベースの認証を使用する前に、以下の手順を完了します。

証明書ベースの認証

1. AppStream 2.0 コンソール (<https://console.aws.amazon.com/appstream2>) を開きます。
2. ナビゲーションペインで、[Directory Configs] (ディレクトリの設定) を選択します。設定するディレクトリ設定を選択し、[編集] を選択します。
3. [証明書ベースの認証を有効にする] を選択します。
4. プライベート CA の ARN がリストで関連付けられていることを確認します。リストに表示するには、プライベート CA を同じ AWS アカウント とに保存する必要があります AWS リージョン。また、プライベート CA には `euc-private-ca` という名前のキーをタグ付けする必要があります。
5. フォールバックのディレクトリログを設定します。フォールバックを使用すると、証明書ベースの認証に失敗した場合でも、ユーザーは AD ドメインのパスワードでログインできます。これは、ユーザーがドメインパスワードを知っている場合にのみ推奨されます。フォールバックがオフになっていると、ロック画面や Windows のログオフが発生した場合に、セッションによって

ユーザーの接続が切断される可能性があります。フォールバックがオンになっている場合、セッションはユーザーに AD ドメインパスワードの入力を求めます。

6. [Save Changes] を選択します。
7. 証明書ベースの認証が有効になりました。ユーザーが SAML 2.0 を使用して AppStream 2.0 ウェブクライアントまたは Windows 用クライアント (バージョン 1.1.1099 以降) のドメイン結合フリートを使用して AppStream 2.0 スタックに対して認証を行うと、ドメインパスワードの入力を求めるプロンプトが表示されなくなります。証明書ベースの認証を有効にしたセッションに接続するときに、「証明書ベースの認証で接続中...」というメッセージがユーザーに表示されます。

証明書ベースの認証の管理

証明書ベースの認証を有効にしたら、次のタスクを確認します。

トピック

- [プライベート CA 証明書](#)
- [エンドユーザー証明書](#)
- [監査レポート](#)
- [ログ記録とモニタリング](#)

プライベート CA 証明書

一般的な設定では、プライベート CA 証明書の有効期間は 10 年です。証明書の有効期限が切れたプライベート CA を置き換える方法、またはプライベート CA を新しい有効期間で再発行する方法の詳細については、「[プライベート CA ライフサイクルの管理](#)」を参照してください。

エンドユーザー証明書

AppStream 2.0 証明書ベースの認証用に AWS Private CA によって発行されたエンドユーザー証明書は、更新や取り消しを必要としません。これらの証明書の有効期間は短くなっています。AppStream 2.0 は、新しいセッションごとに、または期間の長いセッションの場合は 24 時間ごとに新しい証明書を自動的に発行します。AppStream 2.0 セッションは、これらのエンドユーザー証明書の使用を管理します。セッションを終了すると、AppStream 2.0 はその証明書の使用を停止します。これらのエンドユーザー証明書の有効期間は、一般的な AWS プライベート CA CRL ディストリビューションよりも短くなります。そのため、エンドユーザー証明書を取り消さなくても、CRL に表示されなくなります。

監査レポート

プライベート CA が発行または取り消したすべての証明書を一覧表示する監査報告書を作成できます。詳細については、「[プライベート CA での監査レポートの使用](#)」を参照してください。

ログ記録とモニタリング

CloudTrail を使用して、AppStream 2.0 によるプライベート CA への API コールを記録できます。詳細については、「[AWS CloudTrail とは](#)」および「[CloudTrail の使用](#)」を参照してください。CloudTrail イベント履歴では、AppStream 2.0EcmAssumeRoleSession ユーザー名で作成された acm-pca.amazonaws.com イベントソースの GetCertificate および IssueCertificate のイベント名を表示できます。これらのイベントは、AppStream 2.0 の証明書ベースの認証リクエストごとに記録されます。詳細については、「[CloudTrail イベント履歴でのイベントの表示](#)」を参照してください。

クロスアカウント PCA 共有を有効にする

プライベート CA (PCA) のクロスアカウント共有を使用すると、他のアカウントに一元的な CA を使用するアクセス許可を付与できます。CA は、[AWS Resource Access Manager](#) (RAM) を使用して証明書を生成および発行し、アクセス許可を管理できます。これにより、アカウントごとのプライベート CA は不要になります。プライベート CA のクロスアカウント共有は、同じ AWS リージョン内で生じる AppStream 2.0 の証明書ベースの認証 (CBA) で使用できます。

AppStream 2.0 の CBA でプライベート CA の共有リソースを使用するには、次の手順を実行します。

1. CBA のプライベート CA を一元化された で設定します AWS アカウント。詳細については、「[the section called “証明書ベースの認証”](#)」を参照してください。
2. プライベート CA を AppStream 2.0 リソース AWS アカウント が CBA を利用するリソースと共有します。これを行うには、「[How to use AWS RAM to share your ACM Private CA cross-account](#)」の手順に従います。証明書を作成するために、ステップ 3 を完了する必要はありません。Private CA は、個人と共有することも AWS アカウント、を通じて共有することもできます AWS Organizations。個々のアカウントと共有する場合は、AWS Resource Access Manager コンソールまたは APIs を使用して、リソースアカウントで共有プライベート CA を受け入れる必要があります。

共有を設定するときは、AWS Resource Access Manager リソースアカウントのプライベート CA のリソース共有が

AWSRAMBlankEndEntityCertificateAPICSRPassthroughIssuanceCertificateAuthorityマネージドアクセス許可テンプレートを使用していることを確認します。このテンプレート

は、CBA 証明書の発行時に AppStream 2.0 サービスロールが使用する PCA テンプレートと一致します。

- 共有が成功したら、リソースアカウントのプライベート CA コンソールを使用して、共有プライベート CA を表示します。
- API または CLI を使用して、AppStream 2.0 Directory Config の CBA にプライベート CA ARN を関連付けます。現時点では、AppStream 2.0 コンソールは共有プライベート CA ARN の選択をサポートしていません。CLI コマンドの例を次に示します。

```
aws appstream update-directory-config --directory-  
name <value> --certificate-based-auth-properties  
Status=<value>,CertificateAuthorityArn=<value>
```

AppStream 2.0 でのアクティブディレクトリ管理

AppStream 2.0 でアクティブディレクトリをセットアップして使用するには、次の管理タスクを行う必要があります。

タスク

- [アクティブディレクトリコンピュータオブジェクトを作成および管理するための許可の付与](#)
- [組織単位の識別子名を検索する](#)
- [Image Builder のローカル管理者権限を付与する](#)
- [ドメインの参加に使用するサービスアカウントの更新](#)
- [ユーザーがアイドル状態の場合にストリーミングセッションをロックする](#)
- [ディレクトリ設定を編集する](#)
- [ディレクトリ設定を削除する](#)
- [ドメインの信頼度を使用するように AppStream 2.0 を構成する](#)
- [アクティブディレクトリでの AppStream 2.0 コンピュータオブジェクトの管理](#)

アクティブディレクトリコンピュータオブジェクトを作成および管理するための許可の付与

AppStream 2.0 にアクティブディレクトリコンピュータオブジェクト操作の実行を許可するには、十分なアクセス許可を持つアカウントが必要です。ベストプラクティスとして、必要最小限のアクセス

許可のみを持つアカウントを使用します。最小限のアクティブディレクトリ組織単位 (OU) 許可は以下のとおりです。

- コンピュータオブジェクトの作成
- パスワードの変更
- [Reset Password] (パスワードのリセット)
- 説明の書き込み

アクセス許可をセットアップする前に、まず以下の操作を行う必要があります。

- ドメインに参加済みのコンピュータまたは EC2 インスタンスにアクセスします。
- Active Directory User and Computers MMC スナップインをインストールします。詳細については、Microsoft ドキュメントの「[Installing or Removing Remote Server Administration Tools for Windows 7](#)」を参照してください。
- 適切なアクセス権限を持つドメインユーザーとしてログインし、OU のセキュリティ設定を変更します。
- アクセス権限を委任するユーザーアカウント、サービスアカウント、またはグループを作成または指定します。

最小限のアクセス権限をセットアップするには

1. ドメインまたはドメインコントローラーで [Active Directory Users and Computers] (アクティブディレクトリユーザーとコンピュータ) を開きます。
2. 左のナビゲーションペインで、ドメイン参加権限を提供する最初の OU を選択して、コンテキスト (右クリック) メニューを開き、[制御の委任] を選択します。
3. [Delegation of Control Wizard] ページで、[Next]、[Add] の順に選択します。
4. [ユーザー、コンピュータ、グループの選択] で、事前に作成したユーザーアカウント、サービスアカウント、またはグループを選択し、[OK] を選択します。
5. [Tasks to Delegate] (委任するタスク) ページで、[Create a custom task to delegate] (委任するカスタムタスクの作成) を選択し、[Next (次へ)] を選択します。
6. [Only the following objects in the folder]、[Computer objects] を選択します。
7. [Create selected objects in this folder]、[Next] を選択します。
8. [Permissions] で、[Read]、[Write]、[Change Password]、[Reset Password]、[Next] の順に選択します。

9. [Completing the Delegation of Control Wizard] ページで情報を確認し、[Finish] を選択します。
10. これらのアクセス権限を必要とする追加の OU に対して、ステップ 2 ~ 9 を繰り返します。

グループにアクセス権限を委任した場合は、強力なパスワードを持つユーザーアカウントまたはサービスアカウントを作成し、そのアカウントをグループに追加します。こうすることで、ディレクトリにストリーミングインスタンスを接続するための十分な権限がこのアカウントに与えられます。AppStream 2.0 ディレクトリ設定を作成するときはこのアカウントを使用します。

組織単位の識別子名を検索する

AppStream 2.0 にアクティブディレクトリドメインを登録するときは、組織単位 (OU) 識別名を提供する必要があります。この目的のために OU を作成します。デフォルトのコンピュータコンテナは OU ではなく、AppStream 2.0 で使用することはできません。以下に、この名前を取得する手順を示します。

Note

識別子名は、**OU=** で始まる必要があります。また、その名前をコンピュータオブジェクトに使用することはできません。

この手順を完了するには、まず以下の操作を行う必要があります。

- ドメインに参加済みのコンピュータまたは EC2 インスタンスにアクセスします。
- Active Directory User and Computers MMC スナップインをインストールします。詳細については、Microsoft ドキュメントの「[Installing or Removing Remote Server Administration Tools for Windows 7](#)」を参照してください。
- 適切なアクセス権限を持つドメインユーザーとしてログインし、OU のセキュリティプロパティを読み取ります。

OU 識別子名を確認するには

1. ドメインまたはドメインコントローラーで [Active Directory Users and Computers] (アクティブディレクトリユーザーとコンピュータ) を開きます。
2. [View] で、[Advanced Features] が有効になっていることを確認します。

3. 左のナビゲーションペインで、AppStream 2.0 ストリーミングインスタンスのコンピュータオブジェクトに使用する最初の OU を選択し、コンテキスト (右クリック) メニューを開き、[プロパティ] を選択します。
4. [Attribute Editor] を選択します。
5. [Attributes] の下の [distinguishedName] で、[View] を選択します。
6. [値] で識別子名を選択し、コンテキストメニューを開き、[コピー] を選択します。

Image Builder のローカル管理者権限を付与する

デフォルトで、アクティブディレクトリドメインユーザーに Image Builder インスタンスのローカル管理者権限はありません。このアクセス権限を付与するには、ディレクトリのグループポリシーの設定を使用するか、手動で Image Builder のローカル管理者アカウントを使用します。ローカル管理者権限をドメインユーザーに付与すると、それらのユーザーは、AppStream 2.0 の Image Builder にアプリケーションをインストールしたり、画像を作成したりできます。

内容

- [グループポリシー設定を使用する](#)
- [Image Builder でローカル管理者グループを使用する](#)

グループポリシー設定を使用する

ローカル管理者権限をアクティブディレクトリのユーザーやグループに付与したり、または指定された OU のすべてのコンピュータオブジェクトに付与したりするには、グループポリシー設定を使用します。ローカル管理者のアクセス許可を付与するアクティブディレクトリユーザーまたはグループが既に存在している必要があります。グループポリシー設定を使用するには、まず、以下の操作を行う必要があります。

- ドメインに参加済みのコンピュータまたは EC2 インスタンスにアクセスします。
- グループポリシーマネジメントコンソール (GPMC) の MMC スナップインをインストールします。詳細については、Microsoft ドキュメントの「[Installing or Removing Remote Server Administration Tools for Windows 7](#)」を参照してください。
- アクセス許可を持つドメインユーザーとしてログインし、グループポリシーオブジェクト (GPO) を作成します。GPO を適切な OU にリンクします。

グループポリシー設定を使用して、ローカル管理者のアクセス許可を付与するには

1. ディレクトリまたはドメインコントローラーで、管理者としてコマンドプロンプトを開き、`gpmmc.msc` と入力し、ENTER キーを押します。
2. 左のコンソールツリーで、新しい GPO を作成する OU、または既存の GPO を使用する OU を選択して、以下のいずれかの操作を行います。
 - コンテキスト (右クリック) メニューを開き、[Create a GPO in this domain, Link it here] を選択して、新しい GPO を作成します。[Name] に、この GPO のわかりやすい名前を入力します。
 - 既存の GPO を選択します。
3. GPO のコンテキストメニューを開き、[編集] を選択します。
4. コンソールツリーで、[Computer Configuration] (コンピュータの構成)、[設定]、[Windows Settings] (Windows 設定)、[Control Panel Settings] (コントロールパネル設定)、[Local Users and Groups] (ローカルユーザーおよびグループ) の順に選択します。
5. [Local Users and Groups] (ローカルユーザーおよびグループ) を選択して、コンテキストメニューを開き、[新規]、[Local Group] (ローカルグループ) の順に選択します。
6. [Action] で、[Update] を選択します。
7. [Group name] で、[Administrators(built-in)] を選択します。
8. [メンバー] で、[追加] を選択して、ストリーミングインスタンスに対するローカル管理者権限を割り当てるアクティブディレクトリユーザーアカウントまたはグループを指定します。[Action] で、[Add to this group] を選択し、[OK] を選択します。
9. この GPO を他の OU に適用するには、追加の OU を選択して、コンテキストメニューを開き、[Link an Existing GPO] (既存の GPO のリンク) を選択します。
10. ステップ 2 で指定した新規または既存の GPO 名を使用して、GPO までスクロールし、[OK] を選択します。
11. この設定が必要な追加の OU に対して、ステップ 9 および 10 を繰り返します。
12. 再度 [OK] を選択して、[New Local Group Properties] (新規のローカルグループプロパティ) ダイアログボックスを閉じます。
13. 再度 [OK] を選択し、GPMC を閉じます。

新しい設定を GPO に適用するには、実行中の Image Builder またはフロートを停止して再起動する必要があります。GPO がリンクされている OU の Image Builder およびフリースのローカル管理者

権限が、ステップ 8 で指定したアクティブディレクトリユーザーおよびグループに自動的に付与されます。

Image Builder でローカル管理者グループを使用する

Image Builder でアクティブディレクトリユーザーまたはグループのローカル管理者権限を付与するには、これらのユーザーまたはグループを Image Builder のローカル管理者グループに手動で追加します。これらの権限を使用してイメージから作成された Image Builder で、同様の権限を管理します。

ローカル管理者権限を付与するアクティブディレクトリユーザーまたはグループが既に存在している必要があります。

アクティブディレクトリユーザーまたはグループを Image Builder のローカル管理者グループに追加するには

1. AppStream 2.0 コンソール (<https://console.aws.amazon.com/appstream2>) を開きます。
2. 管理者モードで Image Builder に接続します。Image Builder は実行中でありドメイン参加済みである必要があります。詳細については、「[チュートリアル: アクティブディレクトリのセットアップ](#)」を参照してください。
3. [開始]、[管理ツール] の順に選択し、[コンピュータの管理] をダブルクリックします。
4. 左のナビゲーションペインで、[Local Users and Groups] を選択して [Groups] フォルダを開きます。
5. [Administrators] グループを開いて [Add...] を選択します。
6. ローカル管理者権限を割り当てるアクティブディレクトリユーザーまたはグループをすべて選択して、[OK] を選択します。再度 [OK] を選択して、[管理者プロパティ] ダイアログボックスを開きます。
7. コンピュータの管理を閉じます。
8. アクティブディレクトリユーザーとしてログインし、テストユーザーに Image Builder のローカル管理者権限があることを確認するには、[Admin Commands] (管理者コマンド)、[Switch user] (ユーザーの切り替え) の順に選択し、該当するユーザーの認証情報を入力します。

ドメインの参加に使用するサービスアカウントの更新

ドメインの参加向けに AppStream 2.0 が使用するサービスアカウントを更新するには、2 つのサービスアカウントを別々に使用して、Image Builder およびフリートをアクティブディレクトリドメイ

ンに参加させることをお勧めします。2 つの異なるサービスアカウントを使用することで、サービスアカウントを更新する必要がある場合 (例: パスワードの失効時) にサービスを中断しなくて済みます。

サービスアカウントを更新するには

1. アクティブディレクトリグループを作成し、グループに適切な許可を委任します。
2. サービスアカウントを新しいアクティブディレクトリグループに追加します。
3. 必要に応じて、AppStream 2.0 Directory Config オブジェクトを編集するには、新しいサービスアカウントのサインイン認証情報を入力します。

新しいサービスアカウントを使用してアクティブディレクトリグループをセットアップすると、新しいストリーミングインスタンス操作では新しいサービスアカウントが使用されるのに対し、処理中のストリーミングインスタンス操作では、中断されることなく古いアカウントが引き続き使用されます。

処理中のストリーミングインスタンスオペレーションが完了するまでのサービスアカウントの重複時間は短時間 (1 日以内) です。重複期間が必要なのは、重複期間中に古いサービスアカウントのパスワードを削除または変更できないためです。これを行うと既存のオペレーションが失敗することがあります。

ユーザーがアイドル状態の場合にストリーミングセッションをロックする

AppStream 2.0 では、GPMC の設定が使用されるため、一定の時間が経過してユーザーがアイドル状態になるとストリーミングセッションはロックされます。GPMC を使用するには、まず、以下の操作を行う必要があります。

- ドメインに参加済みのコンピュータまたは EC2 インスタンスにアクセスします。
- GPMC をインストールします。詳細については、Microsoft ドキュメントの「[Installing or Removing Remote Server Administration Tools for Windows 7](#)」を参照してください。
- アクセス許可を持つドメインとしてログインし、GPO を作成します。GPO を適切な OU にリンクします。

ユーザーがアイドル状態のときに自動的にストリーミングインスタンスをロックするには

1. ディレクトリまたはドメインコントローラーで、管理者としてコマンドプロンプトを開き、`gpmmc.msc` と入力し、ENTER キーを押します。

2. 左のコンソールツリーで、新しい GPO を作成する OU、または既存の GPO を使用する OU を選択して、以下のいずれかの操作を行います。
 - コンテキスト (右クリック) メニューを開き、[Create a GPO in this domain, Link it here] を選択して、新しい GPO を作成します。[Name] に、この GPO のわかりやすい名前を入力します。
 - 既存の GPO を選択します。
3. GPO のコンテキストメニューを開き、[編集] を選択します。
4. [User Configuration] (ユーザーの構成) を [ポリシー]、[Administrative Templates] (管理用テンプレート)、[コントロールパネル] の順に展開し、[Personalization] (パーソナライズ) を選択します。
5. [スクリーンセーバーの有効化] をダブルクリックします。
6. [Enable screen saver] (スクリーンセーバーの有効化) ポリシー設定で、[有効] を選択します。
7. [適用]、[OK] の順に選択します。
8. [スクリーンセーバーの指定] をダブルクリックします。
9. [Force specific screen saver] (スクリーンセーバーの指定) ポリシー設定で、[有効] を選択します。
10. [Screen saver executable name (スクリーンセーバーの実行ファイル名)] に **scrnsave.scr** と入力します。この設定が有効になると、システムによってユーザーのデスクトップに黒いスクリーンセーバーが表示されます。
11. [適用]、[OK] の順に選択します。
12. [スクリーンセーバーのパスワード保護] をダブルクリックします。
13. [Password protect the screen saver] (スクリーンセーバーのパスワード保護) ポリシー設定で、[有効] を選択します。
14. [適用]、[OK] の順に選択します。
15. [スクリーンセーバーのタイムアウト] をダブルクリックします。
16. [Screen saver timeout] (スクリーンセーバーのタイムアウト) ポリシー設定で、[有効] を選択します。
17. [Seconds] (秒) に、スクリーンセーバーが適用されるまでのユーザーのアイドル時間の長さを指定します。アイドル時間を 10 分に設定するには、600 秒を指定します。
18. [適用]、[OK] の順に選択します。

19. コンソールツリーの [User Configuration] (ユーザーの構成) を、[ポリシー]、[Administrative Templates] (管理用テンプレート)、[システム] の順に展開し、[Ctrl+Alt+Del Options] を選択します。
20. [コンピュータのロック解除] をダブルクリックします。
21. [Remove Lock Computer] (コンピュータのロック解除) ポリシー設定で、[無効] を選択します。
22. [適用]、[OK] の順に選択します。

ディレクトリ設定を編集する

AppStream 2.0 ディレクトリ設定が作成された後、これを編集して組織単位を追加、削除、変更したり、サービスアカウントのユーザー名を更新したり、サービスアカウントのパスワードを更新したりできます。

ディレクトリ設定を更新するには

1. AppStream 2.0 コンソール (<https://console.aws.amazon.com/appstream2>) を開きます。
2. 左のナビゲーションペインで、[Directory Configs] を選択し編集するディレクトリ設定を選択します。
3. [Actions]、[Edit] の順に選択します。
4. 変更するフィールドを更新します。追加の OU を追加するには、最上位の OU フィールドの横にあるプラス記号 (+) を選択します。OU フィールドを削除するには、フィールドの横にある [x] を選択します。

Note

少なくとも 1 つの OU が必要です。現在使用されている OU を削除することはできません。

5. 変更を保存するには、[Update Directory Config] を選択します。
6. [Details] タブの情報を、変更が反映されるように更新する必要があります。

サービスアカウントのサインイン認証情報の変更は、処理中のストリーミングインスタンスオペレーションに影響しません。新しいストリーミングインスタンスオペレーションでは更新された認証情報が使用されます。詳細については、「[ドメインの参加に使用するサービスアカウントの更新](#)」を参照してください。

ディレクトリ設定を削除する

不要な AppStream 2.0 ディレクトリ設定は削除できます。Image Builder またはフリートに関連付けられているディレクトリ設定は削除できません。

ディレクトリ設定を削除するには

1. AppStream 2.0 コンソール (<https://console.aws.amazon.com/appstream2>) を開きます。
2. 左のナビゲーションペインで、[Directory Configs] を選択し削除するディレクトリ設定を選択します。
3. [アクション]、[削除] の順に選択します。
4. ポップアップメッセージの名前を確認し、[Delete] を選択します。
5. [Update Directory Config] を選択します。

ドメインの信頼度を使用するように AppStream 2.0 を構成する

AppStream 2.0 は、あるドメインにファイルサーバー、アプリケーション、コンピュータオブジェクトなどのネットワークリソースが存在し、別のドメインにユーザーオブジェクトが存在するアクティブディレクトリドメイン環境をサポートします。コンピュータオブジェクトオペレーションに使用するドメインサービスアカウントが、AppStream 2.0 コンピュータオブジェクトと同じドメインにある必要はありません。

ディレクトリ設定を作成する際、適切なアクセス許可を持つサービスアカウントを指定して、サーバー、アプリケーション、コンピュータオブジェクト、その他のネットワークリソースが存在するアクティブディレクトリドメインのコンピュータオブジェクトを管理します。

エンドユーザーアクティブディレクトリアカウントには、以下に対して「Allowed to Authenticate」許可が必要です。

- AppStream 2.0 コンピュータオブジェクト
- ドメインのドメインコントローラー

詳細については、「[アクティブディレクトリコンピュータオブジェクトを作成および管理するための許可の付与](#)」を参照してください。

アクティブディレクトリでの AppStream 2.0 コンピュータオブジェクトの管理

AppStream 2.0 によって、コンピュータオブジェクトがアクティブディレクトリから削除されることはありません。これらのコンピュータオブジェクトはディレクトリで簡単に特定できます。ディレクトリ内の各コンピュータオブジェクトは、Description 属性で作成されます。この属性では、フリートまたは Image Builder インスタンスとその名前を指定します。

コンピュータオブジェクトの Description 例

タイプ	名前	Description 属性
フリート	ExampleFleet	AppStream 2.0 - fleet:ExampleFleet
Image Builder	ExampleImageBuilder	AppStream 2.0 - image-builder:ExampleImageBuilder

AppStream 2.0 によって作成された非アクティブなコンピュータオブジェクトを識別して削除するには、次の dsquery computer および dsrm コマンドを使用します。詳細については、Microsoft ドキュメントの [Dsquery computer](#) と [Dsrm](#) を参照してください。

dsquery コマンドは、一定期間非アクティブなコンピュータオブジェクトを識別します。また、次の形式が使用されます。さらに、dsquery コマンドは、AppStream 2.0 オブジェクトのみが表示されるように、-desc "AppStream 2.0*" パラメータを使用して実行されます。

```
dsquery computer "OU-distinguished-name" -desc "AppStream 2.0*" -inactive number-of-weeks-since-last-login
```

- **OU-distinguished-name** は組織単位の識別名です。詳細については、「[組織単位の識別子名を検索する](#)」を参照してください。**OU-distinguished-name** パラメータを指定しない場合、ディレクトリ全体が検索されます。
- **number-of-weeks-since-last-login** は、アイドル状態の定義方法に基づく任意の値です。

たとえば、次のコマンドでは、過去 2 週間以内にログインされていない

OU=ExampleOU,DC=EXAMPLECO,DC=COM 組織単位内のすべてのコンピュータオブジェクトが表示されます。


```
dsquery computer OU=ExampleOU,DC=EXAMPLECO,DC=COM -desc "AppStream 2.0*" -inactive 2
```

一致が検出された場合、結果は 1 つまたは複数のオブジェクト名です。dsrm コマンドは指定したオブジェクトを削除し、次の形式を使用します。

```
dsrm objectname
```

ここで、*objectname* は dsquery コマンドの出力で取得された完全なオブジェクト名です。たとえば、上記の dsquery コマンドの結果が「ExampleComputer」という名前のコンピュータオブジェクトであるとする、これを削除する dsrm コマンドは次のようになります。

```
dsrm "CN=ExampleComputer,OU=ExampleOU,DC=EXAMPLECO,DC=COM"
```

これらのコマンドは、パイプ (|) 演算子を使用してチェーン処理することができます。たとえば、それぞれに確認のプロンプトを表示しながらすべての AppStream 2.0 コンピュータオブジェクトを削除するには、次の形式を使用します。確認を無効にするには、dsrm に -noprompt パラメータを追加します。

```
dsquery computer OU-distinguished-name -desc "AppStream 2.0*" -inactive number-of-weeks-since-last-log-in | dsrm
```

詳細情報

このトピックに関連する詳細情報については、以下のリソースを参照してください。

- [通知コードのトラブルシューティング](#) — 通知コードエラーの解決策です。
- [Active Directory のトラブルシューティング](#) — 一般的な問題のヘルプです。
- [Microsoft Active Directory](#) — の使用に関する情報 AWS Directory Service。

カスタムブランディングを Amazon AppStream 2.0 に追加する

ユーザーがアプリケーションをストリーミングする際の使い慣れたエクスペリエンスを作成するには、独自のブランドイメージ、テキスト、ウェブサイトのリンクを使用して AppStream 2.0 の外観をカスタマイズできます。また、複数のカラーパレットから 1 つを選択することもできます。AppStream 2.0 をカスタマイズすると、アプリケーションのストリーミングセッション中にデフォルトの AppStream 2.0 ブランドではなく独自のブランドがユーザーに表示されます。

トピック

- [カスタムブランディングオプションを Amazon AppStream 2.0 に追加する](#)
- [カスタムブランディングを Amazon AppStream 2.0 に追加する](#)
- [Amazon AppStream 2.0 のカスタムのリダイレクト URL とフィードバック URL の指定](#)
- [Amazon AppStream 2.0 のカスタムブランディングの変更のプレビュー](#)
- [Amazon AppStream 2.0 のカラーテーマパレット](#)

カスタムブランディングオプションを Amazon AppStream 2.0 に追加する

次のブランドオプションを使用して、ストリーミングアプリケーションカタログページの外観をカスタマイズできます。

Note

カスタムブランドは、ユーザープールサインインポータルまたは AppStream 2.0 がユーザープールのユーザーに送信する E メール通知には使用できません。

ブランド要素	説明	要件と推奨事項
組織のロゴ	ユーザーが慣れ親しんでいるイメージを表示できます。イメージはストリーミングア	ファイルタイプ: .png、.jpg、.jpeg、または .gif 最大サイズ: 1000 px x 500 px

ブランド要素	説明	要件と推奨事項
	アプリケーションカタログページのヘッダーに表示されます。これはユーザーが AppStream 2.0 にサインインした後に表示されます。	最大ファイルサイズ: 300 KB
組織のウェブサイトのリンク	組織の IT サポートや製品のマーケティングサイトなど、ユーザーに便利なリソースへのリンクを表示できます。リンクはストリーミングアプリケーションカタログページのフッターに表示されます。	最大リンク数: 3 形式 (URL): https://example.com または http://example.com 最大長 (表示名): 100 文字 (スペースおよび数字含む) 使用できる特殊文字 (表示名): @ . / # & + \$
カラーテーマ	ウェブサイトのリンク、テキスト、ボタンに適用されます。これらの色は、ストリーミングアプリケーションカタログページの背景のアクセントとしても適用されます。	選択可能な事前定義済みテーマ: 4 各カラーテーマについては、このトピックの「Amazon AppStream 2.0 のカラーテーマパレット」を参照してください。
ページタイトル	ユーザーのアプリケーションストリーミングセッション中にブラウザタブの上部に表示されます。	最大長: 200 文字 (スペースおよび数字含む) 使用できる特殊文字: @ . / # & + \$

ブランド要素	説明	要件と推奨事項
ファビコン	ユーザーがブラウザの多数のタブやブックマークからアプリケーションストリーミングサイトを識別できます。ファビコンは、ユーザーのストリーミングセッション中にアプリケーションストリーミングサイトのブラウザタブの上部に表示されます。	ファイルタイプ: .png、.jpg、.jpeg、.gif、または .ico 最大サイズ: 128 px x 128 px 最大ファイルサイズ: 50 KB
リダイレクト URL	ストリーミングセッション終了時ユーザーがリダイレクトされる URL を指定できます。	形式: https://example.com または http://example.com この URL は、[Branding] (ブランド) ページではなく、スタックの作成時または編集時にスタックの [Details] (詳細) ページで設定します。

ブランド要素	説明	要件と推奨事項
フィードバック URL	[フィードバックを送信] リンクの URL を指定できます。こうすることでユーザーが組織にフィードバックを送信できます。URL を指定しない場合、[Send Feedback] (フィードバックの送信) リンクは表示されません。ユーザーは、送信された新しいポータルフィードバックの提供を選択して、新しいポータルフィードバックを送信できます AWS。	形式: https://example.com または http://example.com この URL は、[Branding] (ブランド) ページではなく、スタックの作成時または編集時にスタックの [Details] (詳細) ページで設定します。

カスタムブランディングを Amazon AppStream 2.0 に追加する

AppStream 2.0 を組織のブランドでカスタマイズするには、AppStream 2.0 コンソールを使用してカスタマイズするスタックを選択し、ブランドを追加します。

Amazon S3 バケットから組織ロゴまたはファビコンを選択する場合は、Amazon S3 バケットのコンテンツが、管理するキー (カスタマーマネージドキー) を使用して暗号化されていないことを確認してください。カスタマー提供の暗号化キー (SSE-C) でサーバー側の暗号化を使用するように設定された Amazon S3 バケットは、組織ロゴとファビコンではサポートされていません。Amazon S3 オブジェクトの保存時に暗号化が必要な場合は、Amazon S3-で管理された暗号化キー (SSE-S3) によるサーバー側の暗号化が組織ロゴとファビコンのオプションです。

カスタムブランドを AppStream 2.0 に追加するには

1. AppStream 2.0 コンソール (<https://console.aws.amazon.com/appstream2>) を開きます。
2. 左ペインで、[Stacks] (スタック) を選択します。
3. スタックリストで、ブランドでカスタマイズするスタックを選択します。

4. [Branding] (ブランド)、[Custom] (カスタム) の順に選択します。
5. [Application catalog page (アプリケーションカタログページ)] で、ユーザーが AppStream 2.0 にサインインすると表示されるストリーミングアプリケーションカタログページの外観をカスタマイズします。
 - a. [Organization logo] (組織のロゴ) では、以下のいずれかを実行します。
 - 組織ロゴを表す Amazon S3 URI を入力するか、[S3 を参照] を選択して Amazon S3 バケットに移動し、組織ロゴを見つけます。
 - 既にアップロードした組織ロゴを表示する場合は、[表示] を選択します。組織ロゴを表す別の Amazon S3 URI を入力するか、[S3 を参照] を選択して別の組織ロゴを検索することで変更できます。
 - b. [Organization website links] (組織のウェブサイトのリンク) では、ページのフッターに表示されるウェブサイトのリンクを 3 つまで指定します。各リンクで [Add Link] (リンクを追加) ボタンを選択し、表示名と URL を入力します。リンクを追加するには、追加するリンクごとにこれらのステップを繰り返します。リンクを削除するには、リンク URL の下にある [Remove] (削除) ボタンを選択します。
 - c. [Color theme] (カラーテーマ) では、ウェブサイトのリンク、本文テキスト、ボタン、およびページ背景のアクセントとして使用する色を選択します。各カラーテーマについては、このトピックの「[Amazon AppStream 2.0 のカラーテーマパレット](#)」を参照してください。
6. [Browser tab] (ブラウザタブ) では、ストリーミングセッション中にブラウザタブの上部でユーザーに表示するページタイトルとアイコンをカスタマイズします。
 - a. [Page title] (ページタイトル) には、ブラウザタブの上部に表示するタイトルを入力します。
 - b. [Favicon] (ファビコン) では、次のいずれかを実行します。
 - ファビコンを表す Amazon S3 URI を入力するか、[S3 を参照] を選択して Amazon S3 バケットに移動し、ファビコンを検索します。
 - 既にアップロードしたファビコンを表示する場合は、[表示] を選択します。または、ファビコンを表す別の Amazon S3 URI を入力するか、[S3 を参照] を選択して別のファビコンを検索することで変更できます。
7. 次のいずれかを実行します。
 - ブランドの変更を適用するには [Save] (保存) を選択します。スタックで起動された新しいストリーミングセッションにユーザーが接続すると、ブランドの変更が表示されます。

 Note

AppStream 2.0 によって、保存したカスタムブランドの変更が保持されます。カスタムブランドの変更を保存した後、AppStream 2.0 のデフォルトブランドを復元する場合、カスタムブランドの変更は保存され後で使用できます。AppStream 2.0 のデフォルトブランドを復元した後でカスタムブランドを再適用する場合は、[Custom (カスタム)]、[Save (保存)] の順に選択します。この場合、最後に保存されたカスタムブランドがユーザーに表示されます。

- ブランドの変更を破棄するには [Cancel] (キャンセル) を選択します。選択の確認を求められたら、[Confirm] (確認) を選択します。変更をキャンセルした場合、最後に保存されたブランドがユーザーに表示されます。

Amazon AppStream 2.0 でのカスタムのリダイレクト URL とフィードバック URL の指定

ユーザーがストリーミングセッションを終了するとリダイレクトされる URL を指定できます。また、ユーザーがフィードバックを送信できる URL も指定できます。デフォルトでは、AppStream 2.0 は、ユーザーがアプリケーションストリーミングセッションの品質に関するフィードバックを送信できるようにするフィードバック送信リンクを表示します。AWS ユーザーが指定のサイトにフィードバックを送信できるようにするには、カスタムのフィードバック URL を指定できます。リダイレクト URL とフィードバック URL は、新しいスタックの作成時または既存のスタックの詳細を編集する際に指定できます。詳細については、「[Amazon AppStream 2.0 でスタックを作成する](#)」を参照してください。

Amazon AppStream 2.0 でのカスタムブランディングの変更のプレビュー

ブランドの変更を本稼働スタックに適用する前にテストスタックに適用し、そのテストスタックのストリーミング URL を作成することで、ユーザーからの見え方をプレビューできます。ブランドの変更を検証した後、本稼働用スタックにデプロイすることができます。詳細については、Getting Started with Amazon AppStream 2.0 の [ステップ 2: ユーザーへのアクセスを提供する](#) を参照してください。

Amazon AppStream 2.0 のカラーテーマパレット

カラーテーマを選択すると、そのテーマの色がストリーミングアプリケーションカタログページのウェブサイトのリンク、テキスト、ボタンに適用されます。この色は、ストリーミングアプリケーションカタログページの背景のアクセントとしても適用されます。カラーテーマパレットの各色について、16 進値も記載します。

カラーテーマ

- [赤](#)
- [水色](#)
- [ブルー](#)
- [ピンク](#)

赤

赤のカラーテーマを選択した場合、次の色が適用されます。



赤 (#d51900) – ボタンとウェブサイトリンクに使用されます。

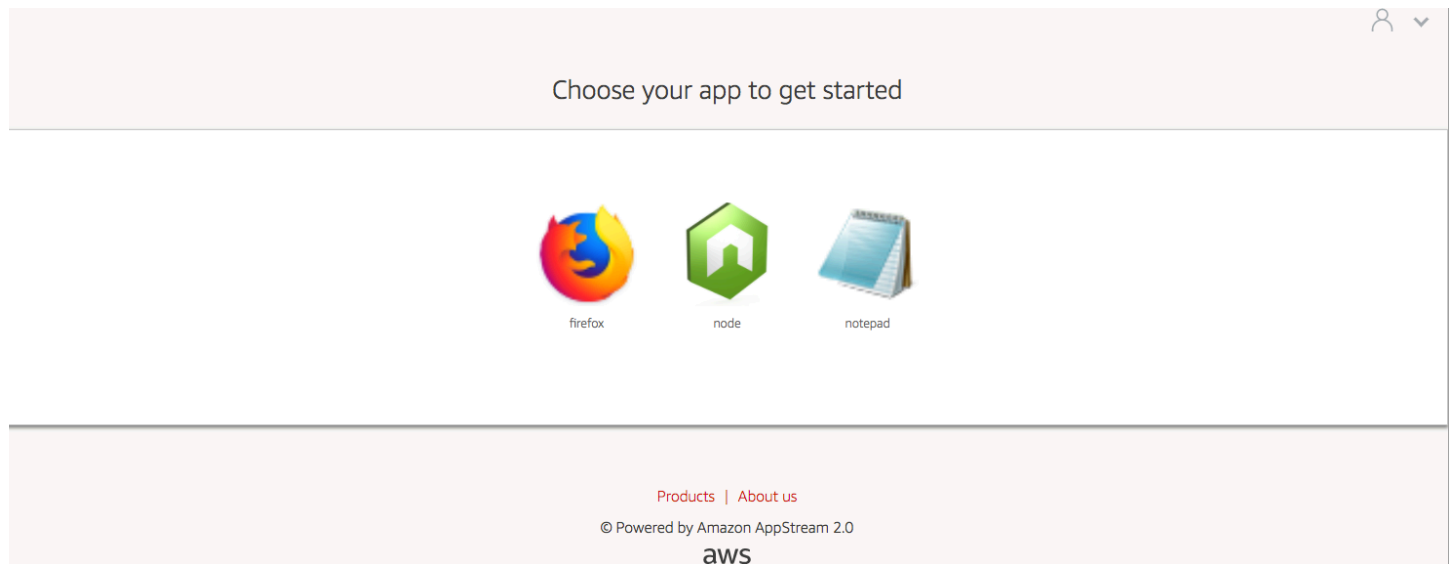


白 (#faf9f7) – 背景のアクセントとして使用されます。



ダークグレー (#404040) – 本文と進行状況を表すスピナーに使用されます。

赤のカラーテーマを選択すると、ストリーミングアプリケーションカタログページのウェブサイトのリンク、本文、背景アクセントは次のように表示されます。



水色

水色のカラーテーマを選択した場合、次の色が適用されます。



水色 (#1d83c2) – ボタンとウェブサイトリンクに使用されます。

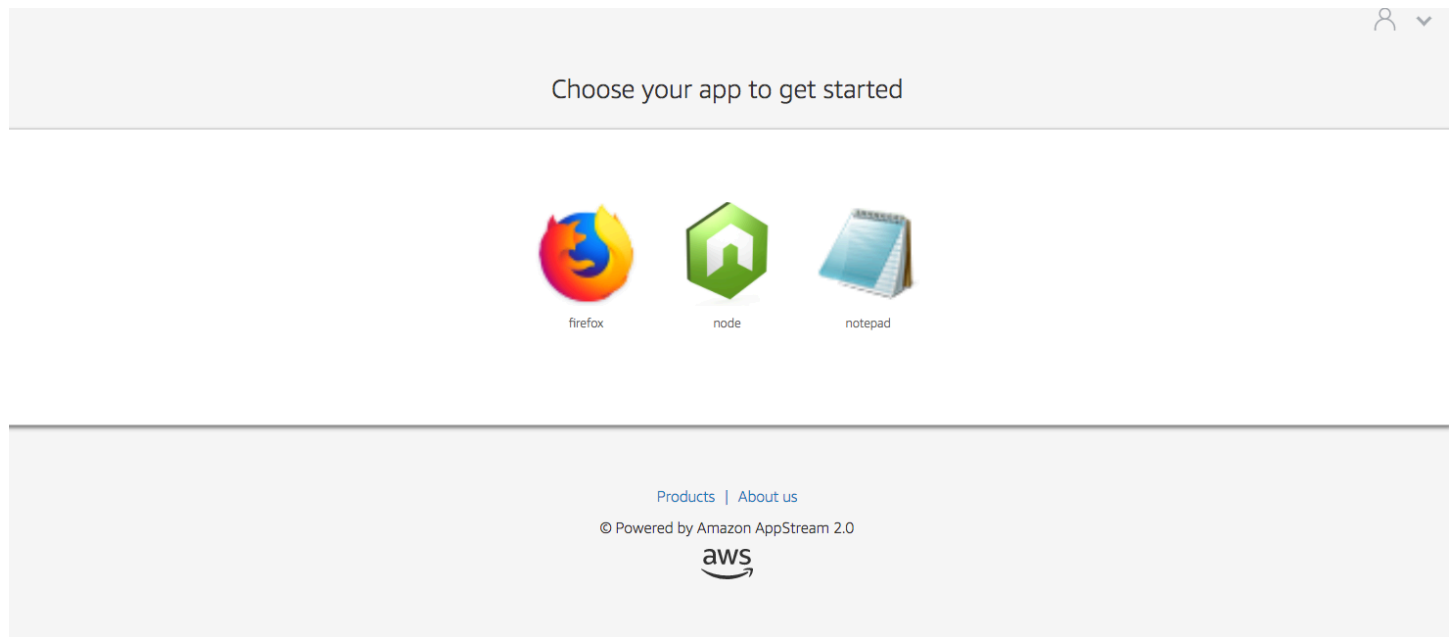


白 (#f6f6f6) – 背景のアクセントとして使用されます。



ダークグレー (#333333) – 本文と進行状況を表すスピナーに使用されます。

水色のカラーテーマを選択すると、ストリーミングアプリケーションカタログページのウェブサイトのリンク、本文、背景アクセントは次のように表示されます。



ブルー

青のカラーテーマを選択した場合、次の色が適用されます。



青 (#0070ba) – ウェブサイトリンクに使用されます。



白 (#ffffff) – 背景のアクセントとして使用されます。



薄緑 (#8ac53e) – ボタンに使用されます。



グレー (#666666) – 本文と進行状況を表すスピナーに使用されます。

青のカラーテーマを選択すると、ストリーミングアプリケーションカタログページのウェブサイトのリンク、本文、背景アクセントは次のように表示されます。



Choose your app to get started



[Products](#) | [About us](#)

© Powered by Amazon AppStream 2.0



ピンク

ピンクのカラーテーマを選択した場合、次の色が適用されます。



ピンク (#ec0069) – ウェブサイトリンクに使用されます。



白 (#ffffff) – 背景のアクセントとして使用されます。



青 (#3159a2) – ボタンに使用されます。



ダークグレー (#333333) – 本文と進行状況を表すスピナーに使用されます。

ピンクのカラーテーマを選択すると、ストリーミングアプリケーションカタログページのウェブサイトのリンク、本文、背景アクセントは次のように表示されます。



Choose your app to get started



firefox



node



notepad

[Products](#) | [About us](#)

© Powered by Amazon AppStream 2.0



埋め込み Amazon AppStream 2.0 ストリーミングセッション

ウェブサイト内に AppStream 2.0 ストリーミングセッションを埋め込むことで、ユーザー向けにカスタマイズした動的でインタラクティブなエクスペリエンスを作成できます。埋め込み AppStream 2.0 ストリーミングセッションにより、ユーザーはウェブサイトから直接、3D モデル、マップ、データセットを操作できます。たとえば、ユーザーはトレーニング手順や教材を表示しながら、AppStream 2.0 ストリーミングセッションを表示できます。

内容

- [Amazon AppStream 2.0 ストリーミングセッションを埋め込むための前提条件](#)
- [Amazon AppStream 2.0 ストリーミングセッションを埋め込む際の推奨事項と使用上の考慮事項](#)
- [ステップ 1: 埋め込み Amazon AppStream 2.0 ストリーミングセッションのホストドメインを指定する](#)
- [ステップ 2: ユーザー認証用のストリーミング URL を作成する](#)
- [手順 3: 埋め込み Amazon AppStream 2.0 ファイルをダウンロードする](#)
- [ステップ 4: Amazon AppStream 2.0 と統合するためにウェブサイトを設定する](#)
- [埋め込み Amazon AppStream 2.0 ストリーミングセッションの定数、関数、イベント](#)

Amazon AppStream 2.0 ストリーミングセッションを埋め込むための前提条件

ウェブサイトに AppStream 2.0 ストリーミングセッションを埋め込むには、以下のものがが必要です。

- AppStream 2.0 イメージ、フリート、およびスタックを含む、構成済みの AppStream 2.0 環境。リソースの作成方法の詳細については、AppStream 2.0 管理者ガイドの以下のトピックを参照してください。
 - [チュートリアル: AppStream 2.0 コンソールを使用してカスタム AppStream 2.0 イメージを作成する](#)、または [Image Assistant CLI オペレーションを使用してプログラムで Amazon AppStream 2.0 イメージを作成する](#)
 - [Amazon AppStream 2.0 でフリートを作成する](#)
 - [Amazon AppStream 2.0 でスタックを作成する](#)

- ユーザー認証用のストリーミング URL。現在、SAML 2.0 および AppStream 2.0 ユーザープールは、埋め込み AppStream 2.0 ストリーミングセッションの認証方法としてサポートされていません。
- オプションで、カスタムドメインを埋め込み AppStream 2.0 ストリーミングセッションに使用できます。カスタムドメインを使用すると、AppStream 2.0 の URL ではなく自社の URL をユーザーに表示できます。ユーザーがサードパーティー Cookie をブロックするウェブブラウザを使用している場合は、カスタムドメインが必要です。

Note

Amazon CloudFront を使用してカスタムドメインを設定できます。詳細については、[Using Custom Domains with AppStream 2.0](#) を参照してください。

カスタムドメインを使用する場合は、以下の操作を行う必要があります。

- 同じドメインを使用するストリーミング URL を作成する。
- 埋め込み AppStream 2.0 ストリーミングセッションをホストするウェブページのヘッダーに **appstream-custom-url-domain** を追加します。ヘッダー値には、リバースプロキシによってユーザーに表示されるドメインを使用します。詳細については、「[カスタムドメインを使用するための設定要件](#)」を参照してください。

Amazon AppStream 2.0 ストリーミングセッションを埋め込む際の推奨事項と使用上の考慮事項

埋め込み AppStream 2.0 ストリーミングセッションについて、以下の推奨事項と考慮事項があります。

- 埋め込み AppStream 2.0 ストリーミングのユーザーエクスペリエンスを最大限に制御するには、有効期間が短い (5 秒ほどの) ストリーミング URL を設定することをお勧めします。いずれのユーザーもウェブページのコンテンツを調べて、そのソースを表示できます。ソースには、ドキュメントオブジェクトモデル (DOM) と iframe の src (ソース) URL が含まれています。ユーザーがその URL をコピーし、別のブラウザタブに貼り付けることができれば、埋め込みオプションを使用することなく、標準の AppStream 2.0 ポータルユーザーインターフェイスを使用して、セッションをストリーミングできます。

- 埋め込み AppStream 2.0 ストリーミングセッションにカスタムドメインが使用されている場合、同時セッションはサポートされません。同時セッションは、ユーザーが同じウェブページまたは 2 つの異なるブラウザタブで 2 つの埋め込み AppStream 2.0 ストリーミングセッションを開始したときに発生します。1 人のユーザーが同時セッションを持つことはできませんが、複数のユーザーを持つことはできます。例えば、ユーザーがアプリケーションにログインし、アプリケーションが顧客に提供するストリーミング URL を生成し (請求では一意のユーザーとしてカウントされます)、顧客がストリーミング URL をロードすると、その顧客は指定されたプール内の appstream インスタンスに割り当てられます。

ステップ 1: 埋め込み Amazon AppStream 2.0 ストリーミングセッションのホストドメインを指定する

ウェブページに AppStream 2.0 ストリーミングセッションを埋め込むには、まずスタックを更新して、埋め込みストリーミングセッションをホストするドメインを指定します。これは、承認されたウェブサイトドメインのみが AppStream 2.0 ストリーミングセッションを埋め込むことができるようにするためのセキュリティ対策です。AppStream 2.0 は、指定したドメインを **Content-Security-Policy** (CSP) ヘッダーに追加します。詳細については、[Mozilla MDN Web Docs](https://developer.mozilla.org/ja/docs/Web/Security/CSP) ドキュメントの「[コンテンツセキュリティポリシー \(CSP\)](https://developer.mozilla.org/ja/docs/Web/Security/CSP)」を参照してください。

スタックを更新して埋め込みストリーミングセッションをホストするドメインを指定するには、次のいずれかの方法を使用します。

- AppStream 2.0 コンソール
- EmbedHostDomains API アクション
- embed-host-domains AWS コマンドラインインターフェイス (AWS CLI) コマンド

AppStream 2.0 コンソールを使用してホストドメインを指定するには、以下の手順を実行します。

1. AppStream 2.0 コンソール (<https://console.aws.amazon.com/appstream2>) を開きます。
2. 左側のナビゲーションペインで [スタック] を選択し、必要なスタックを選択します。
3. [編集] を選択します。
4. [Embed AppStream 2.0 (埋め込み AppStream 2.0) (オプション)] を展開します。
5. [Host Domains (ホストドメイン)] で、有効なドメインを指定します。例:
training.example.com。

 Note

埋め込みストリーミングセッションは、HTTPS (TCP ポート 443) 経由でのみサポートされます。

6. [Update] (更新) を選択します。

ステップ 2: ユーザー認証用のストリーミング URL を作成する

埋め込み AppStream 2.0 ストリーミングセッションのユーザーを認証するためのストリーミング URL を作成する必要があります。現在、SAML 2.0 とユーザープールは埋め込みストリーミングセッションではサポートされていません。ストリーミング URL を作成するには、以下のいずれかの方法を使用します。

- AppStream 2.0 コンソール
- [CreateStreamingURL](#) API アクション
- [create-streaming-url](#) AWS CLI コマンド

カスタムドメインを使用するための設定要件

カスタムドメインを使用する目的が、会社のブランディングを適用することであっても、サードパーティー Cookie をブロックするブラウザで埋め込み AppStream 2.0 ストリーミングセッションが動作するようにすることであっても、設定要件は同じです。

サードパーティー Cookie をブロックするウェブブラウザの場合は、カスタムドメインが必要です。AppStream 2.0 は、ブラウザの Cookie を使用してストリーミングセッションを認証し、ユーザーは毎回サインイン認証情報を入力しなくても、アクティブなセッションに再接続できます。デフォルトでは、AppStream 2.0 ストリーミング URL にはドメインとして **appstream.com** が含まれます。ウェブサイトにストリーミングセッションを埋め込むと、**appstream.com** はサードパーティードメインとして扱われます。その結果、サードパーティー Cookie をデフォルトでブロックする最新のブラウザを使用していると、ストリーミングセッションがブロックされる場合があります。

このシナリオで埋め込み AppStream 2.0 ストリーミングセッションがブロックされないようにするには、以下の手順に従います。

1. 埋め込み AppStream 2.0 ストリーミングセッションをホストするカスタムドメインを指定します。

カスタムドメインを設定するときは、そのドメインが、AppStream 2.0 を埋め込む予定のウェブページのサブドメインであることを確認します。たとえば、スタックを更新してホストドメインとして **training.example.com** を指定する場合、埋め込みストリーミングセッション用に **content.training.example.com** というサブドメインを作成できます。

2. この同じカスタムサブドメインを使用する埋め込み AppStream 2.0 ストリーミングセッション用のストリーミング URL を作成します。ストリーミング URL を作成するには、[CreateStreamingURL](#) API アクションまたは [create-streaming-url](#) AWS CLI コマンドを使用します。このシナリオでは、AppStream 2.0 コンソールを使用してストリーミング URL を作成することはできません。

埋め込み AppStream 2.0 ストリーミングセッション用のストリーミング URL を作成するには、URL で **appstream2.*region*.aws.amazon.com** を独自のドメインに置き換えます。

デフォルトでは、AppStream 2.0 ストリーミング URL は以下の形式になっています。

```
https://appstream2.region.aws.amazon.com/authenticate?parameters=authenticationcode
```

サブドメインが **content.training.example.com** の場合、新しいストリーミング URL は以下の形式になります。

```
https://content.training.example.com/authenticate?parameters=authenticationcode
```

Note

カスタムドメインを作成するときは、そのドメインが設定された AWS リージョンでのみ、埋め込み AppStream 2.0 ストリーミングセッションにドメインを使用できます。複数のリージョンでカスタムドメインをサポートする予定の場合は、該当するリージョンごとにカスタムドメインを作成します。また、埋め込みストリーミングセッションは HTTPS (TCP ポート 443) 経由でのみサポートされます。

3. 埋め込みストリーミングセッションをホストするウェブページのヘッダーに **appstream-custom-url-domain** を追加します。ヘッダー値には、リバースプロキシによってユーザーに表示されるドメインを使用します。例:

```
Header name: appstream-custom-url-domain  
Header value: training.example.com
```

カスタムドメインを設定し、その同じドメインを指定するストリーミング URL を作成すると、その Cookie をファーストパーティー Cookie として保存できるようになります。Amazon CloudFront を使用してカスタムドメインを設定する方法については、[Using Custom Domains with AppStream 2.0](#) を参照してください。

埋め込み AppStream 2.0 ストリーミングセッションのカスタムドメインを設定した後、ストリーミング URL がカスタムドメインにリダイレクトされない場合、またはカスタムドメインがユーザーに対して正しく表示されない場合は、次のトラブルシューティングのトピックをご参照ください。

- [埋め込み AppStream 2.0 ストリーミングセッション用にカスタムドメインを設定しましたが、AppStream 2.0 ストリーミング URL はカスタムドメインにリダイレクトされません。](#)

手順 3: 埋め込み Amazon AppStream 2.0 ファイルをダウンロードする

埋め込み AppStream 2.0 ストリーミングセッションをホストするには、提供されている AppStream 2.0 API JavaScript ファイルをダウンロードして設定する必要があります。

1. [\[Embedding AppStream 2.0 in Your Website \(ウェブサイトへの AppStream 2.0 の埋め込み\)\]](#) ウェブページで、ステップ 1 のリンクを選択して、AppStream 2.0 Embed Kit の .zip ファイル `appstream_embed_<version>.zip` をダウンロードします。
2. .zip ファイルをダウンロードした場所に移動し、ファイルの内容を抽出します。
3. ファイルの抽出された内容は `appstream-embed` という 1 つのフォルダで構成されます。COPYRIGHT.txt および THIRD_PARTY_NOTICES.txt ファイルに加えて、このフォルダには以下の 2 つのファイルが含まれています。
 - `appstream-embed.js` — 組み込み AppStream 2.0 API を提供します。この JavaScript ファイルには、埋め込み AppStream 2.0 ストリーミングセッションを設定および制御するための関数と API アクションが含まれています。
 - `embed-sample.html` — 埋め込み AppStream 2.0 API を使用して、ストリーミングセッションを初期化し、関数を呼び出し、イベントをリッスンする方法について説明されています。このサンプルファイルには、このトピックの情報を応用した開発者向けのユースケースの例が示されています。

ステップ 4. Amazon AppStream 2.0 と統合するためにウェブサイトを設定する

以下のセクションでは、埋め込み AppStream 2.0 ストリーミングセッションをホストするようにウェブページを設定する方法に関する情報を提供します。

内容

- [appstream-embed JavaScript ファイルをインポートする](#)
- [AppStream.Embed インターフェイスオブジェクトを初期化して設定する](#)
- [AppStream 2.0 ユーザーインターフェイスで項目を非表示にする例](#)

appstream-embed JavaScript ファイルをインポートする

1. AppStream 2.0 ストリーミングセッションを埋め込む予定のウェブページで、以下コードを追加して、appstream-embed.js ファイルをウェブページにインポートします。

```
<script type="text/javascript" src="./appstream_embed.js"> </script>
```

2. 次に、空のコンテナ div を作成します。設定した div の ID は AppStream 2.0 組み込みコンストラクタに渡されます。その後、ストリーミングセッションの iframe を注入するために使用されます。div を作成するには、以下のコードを追加します。

```
<div id="appstream-container"> </div>
```

AppStream.Embed インターフェイスオブジェクトを初期化して設定する

JavaScript で AppStream.Embed インターフェイスオブジェクトを初期化するには、ストリーミング URL およびユーザーインターフェイス設定のオプションを使用して AppStream.Embed オブジェクトを作成するコードを追加する必要があります。これらのオプション、および作成した div ID は、appstreamOptions というオブジェクトに保存されます。

以下のサンプルコードでは、AppStream.Embed インターフェイスオブジェクトを初期化する方法を示しています。

```
var appstreamOptions = {
```

```
sessionURL: 'https://appstream2.region.aws.amazon.com/authenticate?
parameters=authenticationcode...',
userInterfaceConfig:[AppStream.Embed.Options.HIDDEN_ELEMENTS]:
[AppStream.Embed.Elements.TOOLBAR]}
};
appstreamEmbed = new AppStream.Embed("appstream-container", appstreamOptions);
```

コードで、*sessionURL* と *userInterfaceConfig* を独自の値に置き換えます。

Note

userInterfaceConfig で指定された値により、AppStream 2.0 ツールバー全体が非表示になります。この値は例として含まれており、オプションです。

sessionUrl

AppStream 2.0 コンソール、[CreateStreamingURL](#) API アクション、または [create-streaming-url](#) AWS CLI コマンドを使用して作成したストリーミング URL。このパラメータは大文字と小文字が区別されます。

型: 文字列

必須: はい

userInterfaceConfig

ユーザーインターフェイス要素の初期状態を生成する設定。この設定はキーと値のペアです。

キー `AppStream.Embed.Options.HIDDEN_ELEMENTS` は、埋め込み AppStream 2.0 ストリーミングセッションの初期化時に最初に非表示になるユーザーインターフェイスオブジェクトを指定します。後で、`getInterfaceState` パラメータを使用して、非表示オブジェクトと表示オブジェクトの両方を返すことができます。

値は、定数 (ツールバーボタン) の配列です。使用できる定数のリストについては、「[「HIDDEN_ELEMENTS」の使用](#)」を参照してください。

タイプ: マップ (*key:Value*)

必須: いいえ

AppStream 2.0 ユーザーインターフェイスで項目を非表示にする例

このセクションの例では、埋め込み AppStream 2.0 ストリーミングセッション中に AppStream 2.0 ユーザーインターフェイスの項目をユーザーに対して非表示にする方法を示します。

例

- [例 1: AppStream 2.0 ツールバー全体を非表示にする](#)
- [例 2: AppStream 2.0 ツールバーの特定のボタンを非表示にする](#)
- [例 3: AppStream 2.0 ツールバーの複数のボタンを非表示にする](#)

例 1: AppStream 2.0 ツールバー全体を非表示にする

埋め込みストリーミングセッション中にユーザーが AppStream 2.0 ツールバーのボタンにアクセスできないようにするには、`AppStream.Embed.Elements.TOOLBAR` 定数を使用します。この定数を使用すると、すべての AppStream 2.0 ツールバーボタンを非表示にすることができます。

```
var appstreamOptions = {  
    sessionURL: 'https://appstream2.region.aws.amazon.com/authenticate?  
parameters=authenticationcode...',  
    userInterfaceConfig:[AppStream.Embed.Options.HIDDEN_ELEMENTS]:  
    [AppStream.Embed.Elements.TOOLBAR]}  
};
```

例 2: AppStream 2.0 ツールバーの特定のボタンを非表示にする

埋め込みストリーミングセッション中にユーザーが AppStream 2.0 ツールバーを表示できるが、特定のツールバーボタンにアクセスできないようにします。そのためには、非表示にするボタンの定数を指定します。以下のコードは、`AppStream.Embed.Elements.FILES_BUTTON` 定数を使用して [My Files] ボタンを非表示にします。これにより、ユーザーは埋め込みストリーミングセッション中に永続的ストレージオプションにアクセスできなくなります。

```
var appstreamOptions = {  
    sessionURL: 'https://appstream2.region.aws.amazon.com/authenticate?  
parameters=authenticationcode...',  
    userInterfaceConfig:[AppStream.Embed.Options.HIDDEN_ELEMENTS]:  
    [AppStream.Embed.Elements.FILES_BUTTON]}  
};
```

例 3: AppStream 2.0 ツールバーの複数のボタンを非表示にする

埋め込みストリーミングセッション中にユーザーが AppStream 2.0 ツールバーを表示できるが、複数のツールバーボタンにアクセスできないようにします。そのためには、非表示にするボタンの定数を指定します。以下のコードは、AppStream.Embed.Elements.END_SESSION_BUTTON および AppStream.Embed.Elements.FULLSCREEN_BUTTON 定数を使用して、[End Session] および [Fullscreen] ボタンを非表示にします。

Note

各定数は、前後にスペースを入れずにカンマで区切ります。

```
var appstreamOptions = {
    sessionURL: 'https://appstream2.region.aws.amazon.com/authenticate?
parameters=authenticationcode... (https://appstream2.region.aws.amazon.com/#/)',
    userInterfaceConfig:[AppStream.Embed.Options.HIDDEN_ELEMENTS]:
[AppStream.Embed.Elements.END_SESSION_BUTTON,AppStream.Embed.Elements.FULLSCREEN_BUTTON]}
};
```

埋め込み Amazon AppStream 2.0 ストリーミングセッションの定数、関数、イベント

以下のトピックでは、埋め込み AppStream 2.0 ストリーミングセッションの設定に使用できる定数、関数、イベントのリファレンス情報を提供します。

内容

- [「HIDDEN_ELEMENTS」の使用](#)
- [AppStream.Embed オブジェクトの関数](#)
- [埋め込み AppStream 2.0 ストリーミングセッションのイベント](#)
- [イベントリスナーの追加例と埋め込み AppStream 2.0 ストリーミングセッションの終了例](#)

以下の AppStream 2.0 ユーザーインターフェイス要素は、埋め込み AppStream 2.0 ストリーミングセッションの初期化時に HIDDEN_ELEMENTS 設定オプションに渡すことができます。

「HIDDEN_ELEMENTS」の使用

以下の AppStream 2.0 ユーザーインターフェイス要素は、埋め込み AppStream 2.0 ストリーミングセッションの初期化時に HIDDEN_ELEMENTS 設定オプションに定数として渡すことができます。

```
AppStream.Embed.Elements.TOOLBAR
AppStream.Embed.Elements.FULLSCREEN_BUTTON
AppStream.Embed.Elements.END_SESSION_BUTTON
AppStream.Embed.Elements.TOOLBAR
AppStream.Embed.Elements.CATALOG_BUTTON
AppStream.Embed.Elements.WINDOW_SWITCHER_BUTTON
AppStream.Embed.Elements.FILES_BUTTON
AppStream.Embed.Elements.CLIPBOARD_BUTTON
AppStream.Embed.Elements.COPY_LOCAL_BUTTON
AppStream.Embed.Elements.PASTE_REMOTE_BUTTON
AppStream.Embed.Elements.SETTINGS_BUTTON
AppStream.Embed.Elements.STREAMING_MODE_BUTTON
AppStream.Embed.Elements.SCREEN_RESOLUTION_BUTTON
AppStream.Embed.Elements.REGIONAL_SETTINGS_BUTTON
AppStream.Embed.Elements.FULLSCREEN_BUTTON
AppStream.Embed.Elements.END_SESSION_BUTTON
```

次の 3 つの要素は、定数としてではなく、文字列として HIDDEN_ELEMENTS に渡すことができます。

String	説明
'adminCommandsButton'	AppStream 2.0 Image Builder に接続すると、AppStream 2.0 ツールバーの右上隅に [Admin Commands (管理者コマンド)] ボタンが表示されます。この文字列を HIDDEN_ELEMENTS に渡すと、[Admin Commands] ボタンが非表示になります。
'softKeyboardButton'	ユーザーは、タッチ対応デバイスでの AppStream 2.0 ストリーミングセッション中に、AppStream 2.0 ツールバーのキーボードアイコンをタップしてオンスクリーンキーボードを表示できます。この文字列を HIDDEN_ELEMENTS に渡すと、キーボードアイコンが非表示になります。

String	説明
'keyboardShortcuts Button'	タッチ対応デバイスの AppStream 2.0 ストリーミングセッション中、ユーザーは AppStream 2.0 ツールバーの [Fn] アイコンをタップしてキーボードショートカットを表示できます。この文字列を <code>HIDDEN_ELEMENTS</code> に渡すと、Fn アイコンが非表示になります。

AppStream.Embed オブジェクトの関数

以下の表では、AppStream.Embed オブジェクトで実行できる関数を示しています。

関数	説明
<code>AppStream.Embed(containerId:string, options:object)</code>	AppStream.Embed オブジェクトコンストラクタ。このコンストラクタは AppStream.Embed オブジェクトを初期化し、このオブジェクトとやり取りします。また、div コンテナ ID を使用します。ID は iframe を注入するために使用されます。さらに、appstreamOptions の設定オプション (<i>sessionURL</i> および <code>HIDDEN_ELEMENTS</code>) を含むオブジェクトを注入します。
<code>endSession()</code>	この関数は、ストリーミングセッションを終了しますが、iframe は破棄しません。リダイレクト URL を指定している場合、iframe では URL のロードが試みられます。ページの CORS ヘッダーによっては、URL がロードされない場合があります。
<code>launchApp(appId:string)</code>	この関数は、イメージの作成中に指定されたアプリケーション ID に対応するアプリケーションをプログラムで起動します。
<code>launchAppSwitcher()</code>	この関数は、AppSwitcher コマンドを AppStream 2.0 ポータルに送信します。これにより、インスタンスに対して AppSwitcher コマンドがトリガーされます。

関数	説明
<code>getSessionState()</code>	この関数は、 <code>sessionStatus</code> のオブジェクトを返します。詳細については、「 埋め込み AppStream 2.0 ストリーミングセッションのイベント 」を参照してください。
<code>getUserInterfaceState()</code>	この関数は、<code>UserInterfaceState</code> のオブジェクトを返します。オブジェクトには、以下のキーと値のペアが含まれます。 <code>sessionStatus</code> : 状態の列挙 <code>sessionTerminationReason</code> : 文字列 <code>sessionDisconnectionReason</code> : 文字列 詳細については、「埋め込み AppStream 2.0 ストリーミングセッションのイベント」を参照してください。
<code>addEventListener(name, callback)</code>	この関数は、指定されたイベントのトリガー時に呼び出すコールバック関数を追加します。トリガーできるイベントのリストについては、「 埋め込み AppStream 2.0 ストリーミングセッションのイベント 」を参照してください。
<code>removeEventListener(name, callback)</code>	この関数は、指定されたイベントのコールバックを削除します。
<code>destroy()</code>	この関数は、 <code>iframe</code> を削除し、リソースをクリーンアップします。この関数は、進行中のストリーミングセッションには影響しません。

埋め込み AppStream 2.0 ストリーミングセッションのイベント

以下の表では、埋め込み AppStream 2.0 ストリーミングセッション中にトリガーできるイベントを示しています。

イベント	[データ]	説明
AppStream.Embed.Events.SESSION_STATE_CHANGE	sessionStatus : State enumeration sessionTerminationReason : 文字列 sessionDisconnectionReason : 文字列	このイベントは、セッション状態の変更が生じたときにトリガーされます。このイベントには、変更された状態のマップが含まれます。完全なセッション状態を取得するには、getSessionState() 関数を使用します。 セッション状態は以下のとおりです。 AppStream.Embed.SessionStatus.Unknown — セッションは開始されておらず、予約されていません AppStream.Embed.Se

イベント	[データ]	説明
		<code>SessionStatus.Reserved</code> — セッションは予約されていますが、開始されていません。
		<code>AppStream.Embed.SessionStatus.Started</code> — ユーザーがセッションに接続し、ストリーミングを開始しました。
		<code>AppStream.Embed.SessionStatus.Disconnected</code> — ユーザーがセッションから切断されました。
		<code>AppStream.Embed.SessionStatus.Ended</code> — セッションが終了また

イベント	[データ]	説明
		は期限切れとしてマークされました。
AppStream.Embed.Events.SESSION_INTERFACE_STATE_CHANGE	hiddenElements : 文字列の配列 isFullscreen : ブール値 isSoftKeyboardVisible : ブール値	このイベントは、セッション状態の変更が生じたときにトリガーされます。このイベントには、変更された状態のマップが含まれます。完全なセッション状態を取得するには、getSessionState() 関数を使用します。
AppStream.Embed.Events.SESSION_ERROR	errorCode : 番号 errorMessage : 文字列	このイベントは、セッション中にエラーが発生するとトリガーされます。

イベントリスナーの追加例と埋め込み AppStream 2.0 ストリーミングセッションの終了例

このセクションの例では、次の操作を行う方法を示します。

- 埋め込み AppStream 2.0 ストリーミングセッションのイベントリスナーを追加する。
- 埋め込み AppStream 2.0 ストリーミングセッションをプログラムで終了する。

例 1: 埋め込み AppStream 2.0 ストリーミングセッションのイベントリスナーを追加する

埋め込みストリーミングセッション中のセッションの状態の変更、セッションインターフェイスの状態の変更、セッションエラーに対するイベントリスナーを追加するには、以下のコードを使用します。

```
appstreamEmbed.addEventListener(AppStream.Embed.Events.SESSION_STATE_CHANGE,
    updateSessionStateCallback);

appstreamEmbed.addEventListener(AppStream.Embed.Events.SESSION_INTERFACE_STATE_CHANGE,
    updateUserInterfaceStateCallback);

appstreamEmbed.addEventListener(AppStream.Embed.Events.SESSION_ERROR, errorCallback);
```

この例で

は、AppStream.Embed.Events.SESSION_STATE_CHANGE、AppStream.Embed.Events.SESSION_I はイベント名です。

updateSessionStateCallback、updateUserInterfaceStateCallback、errorCallback は実装する関数です。これらの関数は addEventListener 関数に渡され、イベントがトリガーされると呼び出されます。

例 2: 埋め込み AppStream 2.0 ストリーミングセッションをプログラムで終了する

埋め込み AppStream 2.0 ストリーミングセッションを終了するには、以下の関数を使用します。

```
appstreamEmbed.endSession();
```

AppStream 2.0 ユーザーに対して永続的ストレージを有効にして管理する

Amazon AppStream 2.0 では、組織のユーザーに対して次の永続的ストレージをサポートしています。

- ホームフォルダ
- Google Drive for Google Workspace
- OneDrive for Business
- カスタム共有フォルダ (サーバーメッセージブロック (SMB) ネットワークドライブ)

組織に対して、1 つ以上のオプションを有効にすることができます。AppStream 2.0 管理者として、ユーザーに対して永続的ストレージを有効にして管理するため、次のタスクを実行する方法を理解する必要があります。

目次

- [AppStream 2.0 ユーザーに対してホームフォルダを有効にして管理する](#)
- [AppStream 2.0 ユーザーに対して Google ドライブを有効にして管理する](#)
- [AppStream 2.0 ユーザーに対して OneDrive for Business を有効にして管理する](#)
- [AppStream 2.0 ユーザーのカスタム共有フォルダ \(サーバーメッセージブロック \(SMB\) ネットワークドライブ\) の有効化と管理](#)

トラブルシューティング情報については、「[永続ストレージ問題のトラブルシューティング](#)」を参照してください。

AppStream 2.0 ユーザーに対してホームフォルダを有効にして管理する

AppStream 2.0 では、組織のユーザーに対して次の永続的ストレージをサポートしています。

- ホームフォルダ
- Google Drive for Google Workspace
- OneDrive for Business

- カスタム共有フォルダ (サーバーメッセージブロック (SMB) ネットワークドライブ)

組織に対して、1 つ以上のオプションを有効にすることができます。AppStream 2.0 スタックでホームフォルダを有効にすると、アプリケーションストリーミングセッション中に、スタックのユーザーは永続的ストレージのフォルダにアクセスできます。ユーザーがホームフォルダにアクセスするために必要な設定はありません。ユーザーが自分のホームフォルダに保存したデータは、Amazon Web Services アカウントの Amazon Simple Storage Service バケットに自動的にバックアップされ、そのユーザーの後のセッションで使用できるようになります。

転送中のファイルやフォルダは Amazon S3 の SSL エンドポイントを使用して暗号化されます。保管中のファイルやフォルダは Amazon S3 で管理される暗号化キーを使用して暗号化されます。

ホームフォルダは、以下のデフォルトの場所にあるフリートインスタンスに保存されます。

- シングルセッションで、ドメインに参加していない Windows インスタンス: C:\Users\PhotonUser\My Files\Home Folder
- マルチセッションで、ドメインに参加していない Windows インスタンス: C:\Users\as2-xxxxxxx\My Files\Home Folder (as2-xxxxxxx は各ユーザーセッションに割り当てられたランダムなユーザー名) ローカルユーザー名は、環境変数 \$USERNAME を使用して決定できます。
- ドメインに参加済みの Windows インスタンス: C:\Users\%username%\My Files\Home Folder
- Linux インスタンス: ~/MyFiles/HomeFolder

ホームフォルダを保存先とするようにアプリケーションを設定する場合は、該当パスを管理者として使用します。ユーザーがホームフォルダを見つけられない場合があります。アプリケーションによっては、File Explorer の最上位フォルダとしてホームフォルダを表示する、リダイレクトを認識しないためです。このような場合は、File Explorer 内の同じディレクトリを参照することで、ユーザーがホームフォルダにアクセスにできます。

目次

- [計算集約型アプリケーションに関連するファイルとディレクトリ](#)
- [AppStream 2.0 ユーザーのホームフォルダを有効にする](#)
- [ホームフォルダを管理する](#)

計算集約型アプリケーションに関連するファイルとディレクトリ

AppStream 2.0 ストリーミングセッションでは、処計算集約型アプリケーションに関連付けられた大きなファイルとディレクトリを永続ストレージに保存すると、基本的な生産性アプリケーションに必要なファイルとディレクトリを保存するよりも時間がかかる場合があります。たとえば、アプリケーションが大量のデータを保存したり、同じファイルを頻繁に変更したりする場合は、1 回の書き込み操作を実行するアプリケーションによって作成されたファイルを保存する場合よりも時間がかかる場合があります。また、多くの小さなファイルを保存するのに時間がかかる場合があります。

コンピューティング集約型アプリケーションに関連付けられたファイルとディレクトリをユーザーが保存し、AppStream 2.0 永続ストレージオプションが期待どおりに動作しない場合は、Amazon FSx for Windows File Server や AWS Storage Gateway ファイルゲートウェイなどのサーバーメッセージブロック (SMB) ソリューションを使用することをお勧めします。以下は、これらの SMB ソリューションでの使用に適した、計算集約型アプリケーションに関連するファイルとディレクトリの例です。

- 統合開発環境 (IDE) 用の Workspace フォルダ
- ローカルデータベースファイル
- グラフィックシミュレーションアプリケーションによって作成されたスクラッチスペースフォルダ

詳細については、以下を参照してください。

- [Amazon FSx for Windows File Server Windows ユーザーガイド](#)
- [Using Amazon FSx with Amazon AppStream 2.0](#)
- AWS Storage Gateway ユーザーガイドの[ファイルゲートウェイ](#)

AppStream 2.0 ユーザーのホームフォルダを有効にする

ホームフォルダを有効にする前に、以下を実行する必要があります。

- Amazon S3 アクションの正しい AWS Identity and Access Management (IAM) アクセス許可があることを確認します。詳細については、「[IAM ポリシーを使用したホームフォルダおよびアプリケーション設定の永続化用の Amazon S3 バケットへの管理者アクセスの管理](#)」を参照してください。
- 2017 年 5 月 18 日以降にリリースされた AWS ベースイメージから作成されたイメージを使用します。リリースされた AWS イメージの現在のリストについては、「」を参照してください。[AppStream 2.0 ベースイメージとマネージド型イメージの更新に関するリリースノート](#)。

- インターネットアクセスまたは Amazon S3 の VPC エンドポイントを設定して、Virtual Private Cloud (VPC) から Amazon S3 へのネットワーク接続を有効にします。詳細については、「[Amazon AppStream 2.0 のネットワーキングとアクセス](#)」および「[AppStream 2.0 機能に対する Amazon S3 VPC エンドポイントの使用](#)」を参照してください。

スタックの作成中 (「」を参照[Amazon AppStream 2.0 でスタックを作成する](#))、または AppStream 2.0、AWS SDK、または AWS Management Console のを使用してスタックの作成後に、ホームフォルダを有効または無効にできます AWS CLI。各 AWS リージョンについて、ホームフォルダは Amazon S3 バケットによってバックアップされます。

AWS リージョンで AppStream 2.0 スタックのホームフォルダを初めて有効にすると、サービスは同じリージョンのアカウントに Amazon S3 バケットを作成します。同じバケットを使用して、そのリージョンのすべてのユーザーおよびすべてのスタックのホームフォルダのコンテンツが保存されます。詳細については、[Amazon S3 バケットのストレージ](#) を参照してください。

 Note

AppStream 2.0 ストリーミングセッション中のホームフォルダの使用を開始するのに役立つようにユーザーに提供できるガイダンスについては、[ホームフォルダを使用する](#) を参照してください。

スタックの作成時にホームフォルダを有効にするには

- 「[Amazon AppStream 2.0 でスタックを作成する](#)」の手順に従い、[Enable Home Folders (ホームフォルダを有効にする)] が選択されていることを確認します。

既存のスタックのホームフォルダを有効にするには

- AppStream 2.0 コンソールを <https://console.aws.amazon.com/appstream2> で開きます。
- 左側のナビゲーションペインで、[Stacks (スタック)] を選択し、ホームフォルダを有効にするスタックを選択します。
- スタックリストの下で、[Storage] をクリックし、[Enable Home Folders] を選択します。
- [Enable Home Folders] ダイアログボックスで、[Enable] を選択します。

ホームフォルダを管理する

ホームフォルダを管理する方法については、次のトピックを参照してください。

内容

- [ホームフォルダを無効にする](#)
- [Amazon S3 バケットのストレージ](#)
- [ホームフォルダコンテンツの同期](#)
- [ホームフォルダの形式](#)
- [AWS Command Line Interface または AWS SDKsの使用](#)
- [その他のリソース](#)

ホームフォルダを無効にする

既にホームフォルダに保存されているユーザーコンテンツを失うことなく、スタックに対してホームフォルダを無効にできます。スタック用のホームフォルダを無効にすると、次のようになります。

- スタックのアクティブなストリーミングセッションに接続されているユーザーはエラーメッセージを受け取ります。ホームフォルダにコンテンツを保存できなくなることが通知されます。
- ホームフォルダが無効になったスタックを使用する新しいセッションでは、ホームフォルダは表示されません。
- 1つのスタックのホームフォルダを無効にしても、他のスタックに対して無効になりません。
- すべてのスタックに対してホームフォルダが無効になっている場合でも、AppStream 2.0 はユーザーコンテンツを削除しません。

スタックのホームフォルダへのアクセスを復元するには、このトピックで前に説明したステップに従って、ホームフォルダをもう一度有効にします。

スタックの作成時にホームフォルダを無効にするには

- 「[Amazon AppStream 2.0 でスタックを作成する](#)」の手順に従い、[Enable Home Folders (ホームフォルダを有効にする)] オプションが選択解除されていることを確認します。

既存のスタックのホームフォルダを無効にするには

1. AppStream 2.0 コンソールを <https://console.aws.amazon.com/appstream2> で開きます。

2. 左ナビゲーションペインで [Stacks (スタック)] を選択し、スタックを選択します。
3. スタックリストの下、[Storage] をクリックし、[Enable Home Folders] をクリアします。
4. [Disable Home Folders] ダイアログボックスで、CONFIRM (大文字と小文字は区別されます) と入力し選択を確認します。次に [Disable] を選択します。

Amazon S3 バケットのストレージ

AppStream 2.0 は、アカウントで作成された Amazon S3 バケットを使用して、ホームフォルダに保存されているユーザーコンテンツを管理します。AppStream 2.0 は、AWS リージョンごとにアカウントにバケットを作成します。そのリージョン内のスタックのストリーミングセッションから生成されたすべてのユーザーコンテンツはそのバケットに保存されます。このバケットは、管理者が入力または設定することなく、サービスによって完全に管理されます。

新しい拡張バケットには、次のように特定の形式 (バージョン 2) で名前が付けられます。

```
appstream2-36fb080bb8-region-code-account-id-without-hyphens-random-identifier
```

ここで *region-code*、はスタックが作成される AWS リージョンコードで、*account-id-without-hyphens* は Amazon Web Services アカウント ID です。バケット名の最初の部分 appstream2-36fb080bb8- は、複数のアカウントやリージョンにまたがる場合でも変更されません。

たとえば、アカウント番号 123456789012 で米国西部 (オレゴン) リージョン (us-west-2) のスタックのホームフォルダを有効にした場合、サービスにより、表示された名前でもリージョンに Amazon S3 バケットが作成されます。適切なアクセス許可を持つ管理者のみが、このバケットを削除できます。

```
appstream2-36fb080bb8-us-west-2-123456789012-abcdefg
```

古いバージョンのバケットの名前は次のとおりです。AppStream 2.0 が新しい拡張バケット命名 (バージョン 2) を導入する前に作成されたアカウントは、古い命名形式に従います。

```
appstream2-36fb080bb8-region-code-account-id-without-hyphens
```

ここで *region-code*、はスタックが作成される AWS リージョンコードで、*account-id-without-hyphens* は Amazon Web Services アカウント ID です。バケット名の最初の部分 appstream2-36fb080bb8- は、複数のアカウントやリージョンにまたがる場合でも変更されません。

たとえば、アカウント番号 123456789012 で米国西部 (オレゴン) リージョン (us-west-2) のスタックのホームフォルダを有効にした場合、サービスにより、表示された名前でリージョンに Amazon S3 バケットが作成されます。適切なアクセス許可を持つ管理者のみが、このバケットを削除できます。

```
appstream2-36fb080bb8-us-west-2-123456789012
```

前述のとおり、スタックに対してホームフォルダを無効にしても、Amazon S3 バケットに保存されたユーザーコンテンツは削除されません。ユーザーコンテンツを完全に削除するには、適切なアクセス権限を持った管理者が、Amazon S3 コンソールから行う必要があります。AppStream 2.0 は、バケットの誤った削除を防止するバケットポリシーを追加します。詳細については、[IAM ポリシーを使用したホームフォルダおよびアプリケーション設定の永続化用の Amazon S3 バケットへの管理者アクセスの管理](#) を参照してください。

ホームフォルダコンテンツの同期

ホームフォルダが有効な場合、AppStream 2.0 は、コンテンツを保存するユーザーごとに一意のフォルダを作成します。このフォルダは、Amazon Web Services アカウント (リージョン) にある S3 バケット内のユーザー名のハッシュを使用する、一意の Amazon S3 プレフィックスとして作成されます。AppStream 2.0 は Amazon S3 にホームフォルダを作成した後、そのフォルダ内のアクセスされたコンテンツを S3 バケットからフリートインスタンスにコピーします。これにより、ユーザーはストリーミングセッション中に、フリートインスタンスからホームフォルダのコンテンツにすばやくアクセスすることができます。S3 バケット内のユーザーのホームフォルダコンテンツに加えた変更と、フリートインスタンス上のホームフォルダコンテンツに対してユーザーが加えた変更は、Amazon S3 と AppStream 2.0 間で次のように同期されます。

1. ユーザーの AppStream 2.0 ストリーミングセッションが開始される際に、AppStream 2.0 は、そのユーザー用として Amazon S3 バケットに保存されているホームフォルダファイルを、Amazon Web Services アカウントとリージョンのためにカタログ化します。
2. ユーザーのホームフォルダコンテンツは、ストリーミング元の AppStream 2.0 フリートインスタンスにも保存されます。ユーザーが AppStream 2.0 フリートインスタンス上のホームフォルダにアクセスすると、カタログ化されたファイルの一覧が表示されます。
3. AppStream 2.0 は、ユーザーがストリーミングアプリケーションを使用してストリーミングセッション中にファイルを開いた後にのみ、S3 バケットからフリートインスタンスにファイルをダウンロードします。
4. AppStream 2.0 がフリートインスタンスにファイルをダウンロードした後、ファイルにアクセスした後に同期が行われます

5. ユーザーがストリーミングセッション中にファイルを変更した場合、AppStream 2.0 は定期的に、またはストリーミングセッションの最後に、新しいバージョンのファイルをフリートインスタンスから S3 バケットにアップロードします。ただし、ストリーミングセッション中にファイルは S3 バケットから再度ダウンロードされません。

以下のセクションでは、Amazon S3 でユーザーのホームフォルダファイルを追加し、置き換え、削除するときの同期動作について説明します。

目次

- [Amazon S3 ユーザーのホームフォルダに追加したファイルの同期](#)
- [Amazon S3 ユーザーのホームフォルダで置き換えたファイルの同期](#)
- [Amazon S3 ユーザーのホームフォルダから削除したファイルの同期](#)

Amazon S3 ユーザーのホームフォルダに追加したファイルの同期

S3 バケット内のユーザーのホームフォルダに新しいファイルを追加すると、AppStream 2.0 はファイルをカタログ化して、数分以内にユーザーのホームフォルダ内にあるファイルの一覧に表示します。ただし、ストリーミングセッション中にユーザーがアプリケーションでファイルを開くまで、ファイルは S3 バケットからフリートインスタンスにダウンロードされません。

Amazon S3 ユーザーのホームフォルダで置き換えたファイルの同期

ユーザーがストリーミングセッション中にフリートインスタンスのホームフォルダ内のファイルを開き、そのユーザーのアクティブなストリーミングセッション中に S3 バケットのホームフォルダにある同じファイルを新しいバージョンに置き換えた場合、新しいバージョンのファイルはフリートインスタンスにすぐにダウンロードされません。新しいバージョンは、ユーザーが新しいストリーミングセッションを開始してファイルを再度開いた後にのみ、S3 バケットからフリートインスタンスにダウンロードされます。

Amazon S3 ユーザーのホームフォルダから削除したファイルの同期

ユーザーがストリーミングセッション中にフリートインスタンスのホームフォルダ内のファイルを開き、そのユーザーのアクティブなストリーミングセッション中に S3 バケットのホームフォルダからファイルを削除すると、ユーザーが次のいずれかの操作を行った後、そのファイルはフリートインスタンスから削除されます。

- ホームフォルダを再度開く
- ホームフォルダを更新する

ホームフォルダの形式

ユーザーフォルダの階層は、次のセクションで説明するように、ユーザーがストリーミングセッションを起動する方法によって異なります。

AWS SDKsと AWS CLI

CreateStreamingURL または create-streaming-url を使用して起動されたセッションでは、ユーザーフォルダ構造は次のようになります。

```
bucket-name/user/custom/user-id-SHA-256-hash/
```

ここで、*bucket-name* は [Amazon S3 バケットのストレージ](#) に示す形式で、*user-id-SHA-256-hash* は CreateStreamingURL API オペレーションまたは UserId コマンドに渡された create-streaming-url の値から生成された小文字の SHA-256 ハッシュ 16 進数文字列を使用して作成された、ユーザー固有のフォルダ名です。詳細については、Amazon AppStream 2.0 API リファレンスの「[CreateStreamingURL](#)」、および AWS CLI コマンドリファレンスの「[create-streaming-url](#)」を参照してください。

次のフォルダ構造例は、米国西部 (オレゴン) リージョン (us-west-2) の API または AWS CLI UserId testuser@mydomain.com、アカウント ID 123456789012 を使用したセッションアクセスに適用されます。

```
appstream2-36fb080bb8-us-west-2-123456789012/user/custom/  
a0bcb1da11f480d9b5b3e90f91243143eac04cfccfbdc777e740fab628a1cd13/
```

ウェブサイトを使用するか、オンラインで入手できるオープンソースコーディングライブラリを使用して、小文字の UserId の SHA-256 ハッシュ値を生成してユーザーのフォルダを識別できます。

SAML 2.0

SAML フェデレーションを使用して作成されたセッションでは、ユーザーフォルダ構造は次のようになります。

```
bucket-name/user/federated/user-id-SHA-256-hash/
```

この場合、*user-id-SHA-256-hash* は、SAML フェデレーションリクエストに渡された NameID SAML 属性値から生成された、小文字の SHA-256 ハッシュ 16 進数文字列を使用して作成されたフォルダ名です。2 つの異なるドメインに属する同じ名前のユーザーを区別するには、NameID 形式

で `domainname\username` を含む SAML リクエストを送信します。詳細については、[Amazon AppStream 2.0 と SAML 2.0 の統合](#) を参照してください。

次の例のフォルダ構造は、米国西部 (オレゴン) リージョン の NameID `SAMPLEDOMAIN\testuser`、アカウント ID `123456789012` と SAML フェデレーションを使用したセッションアクセスに適用されます。

```
appstream2-36fb080bb8-us-west-2-123456789012/user/  
federated/8dd9a642f511609454d344d53cb861a71190e44fed2B8aF9fde0C507012a9901
```

NameID 文字列の一部またはすべてが大文字の場合 (例ではドメイン名 ***SAMPLEDOMAIN***)、AppStream 2.0 では、文字列を大文字化したものに基づいてハッシュ値が生成されます。この例では、`SAMPLEDOMAIN\testuser` のハッシュ値は `8DD9A642F511609454D344D53CB861A71190E44FED2B8AF9FDE0C507012A9901` です。そのユーザーのフォルダで、この値は、`8dd9a642f511609454d344d53cb861a71190e44fed2B8aF9fde0C507012a9901` のように小文字で表示されます。

ウェブサイトを使用するか、オンラインで入手できるオープンソースコーディングライブラリを使用して、NameID の SHA-256 ハッシュ値を生成してユーザーのフォルダを識別できます。

AWS Command Line Interface または AWS SDKsの使用

または AWS SDKs を使用して、スタックのホームフォルダを有効 AWS CLI または無効にできます。

次の [create-stack](#) コマンドを使用して、新しいスタックを作成中にホームフォルダを有効にできます。

```
aws appstream create-stack --name ExampleStack --storage-connectors  
ConnectorType=HOMEFOLDERS
```

次の [update-stack](#) コマンドを使用して、既存のスタックに対するホームフォルダを有効にできます。

```
aws appstream update-stack --name ExistingStack --storage-connectors  
ConnectorType=HOMEFOLDERS
```

次のコマンドを使用して、既存のスタックに対するホームフォルダを無効にできます。このコマンドによりユーザーデータは削除されません。


```
aws appstream update-stack --name ExistingStack --delete-storage-connectors
```

その他のリソース

Amazon S3 バケットの管理とベストプラクティスの詳細については、Amazon Simple Storage Service ユーザーガイドにある次のトピックを参照してください。

- Amazon S3 ポリシーにより、ユーザーにユーザーデータへのオフラインアクセスを提供できます。詳細については、IAM ユーザーガイドの [Amazon S3: Allows IAM Users Access to Their S3 Home Directory, Programmatically and In the Console](#) を参照してください。
- AppStream 2.0 によって使用される Amazon S3 バケットに保存されたコンテンツに対して、ファイルバージョニングを有効にすることができます。詳細については、「[バージョニングの使用](#)」を参照してください。

AppStream 2.0 ユーザーに対して Google ドライブを有効にして管理する

Note

Google API から受け取った情報の Amazon AppStream 2.0 による 使用および他のアプリケーションへの転送には、限定使用要件を含む [Google API サービスのユーザーデータポリシー](#) が適用されます。

Amazon AppStream 2.0 では、組織のユーザーに対して次の永続的ストレージをサポートしています。

- Google Drive for Google Workspace
- OneDrive for Business
- ホームフォルダ

組織に対して、1 つ以上のオプションを有効にすることができます。AppStream 2.0 スタックに対して Google Drive for Google Workspace を有効にすると、スタックのユーザーは Google Drive for Google Workspace アカウントを AppStream 2.0 にリンクできます。その後、ユーザーは、Google Drive for Google Workspace のアカウントにサインインして、アプリケーションストリーミングセッ

セッション中に Google ドライブフォルダにアクセスできます。このセッション中に、Google ドライブでユーザーがファイルやフォルダに行う変更は、自動的にバックアップ、同期されます。結果として、ストリーミングセッション外部から利用できるようになります。

Important

Google Drive for Google Workspace を有効にできるのは、Google Workspace ドメイン内のアカウントに対してのみで、個人の Gmail アカウントに対してはできません。

Note

Google ドライブは、Windows スタックに対しては有効にすることができますが、Linux スタックやマルチセッションに関連付けられたスタックに対して有効にすることはできません。

内容

- [AppStream 2.0 ユーザーの Google ドライブを有効にする](#)
- [AppStream 2.0 ユーザーの Google ドライブを無効にする](#)

AppStream 2.0 ユーザーの Google ドライブを有効にする

Google ドライブを有効にする前に、以下を実行する必要があります。

- 有効な組織ドメインを持つアクティブな Google Workspace アカウントと、そのドメインのユーザーを AppStream 2.0 と一緒に使用します。
- 関連付けられているフリートを使用して AppStream 2.0 を設定します。

このフリートでは、2018 年 5 月 31 日以降にリリースされたバージョンの AppStream 2.0 エージェントを使用するイメージを使用する必要があります。詳細については、[AppStream 2.0 エージェントのリリースノート](#) を参照してください。このフリートでは、インターネットにアクセスできることも必要です。

- 信頼されているアプリとして、Amazon AppStream 2.0 を Google Workspace アカウントに関連付けられている 1 つ以上のドメインに追加します。最大 10 個のドメインに対して Google ドライブを有効にすることができます。
- Windows ベースのスタックを用意します。(Linux ベースのスタックはサポートされていません)。

Google Workspace ドメインに Amazon AppStream 2.0 を信頼されたアプリとして追加するには、以下のステップを実行します。

Amazon AppStream 2.0 を Google Workspace ドメインで信頼できるアプリとして追加するには

1. <https://admin.google.com/> で Google Workspace 管理者コンソールにサインインします。
2. 左側のナビゲーションサイドバーで、[Security] (セキュリティ)、[Access and data control] (アクセスとデータコントロール)、[API controls] (API コントロール) を選択します。
3. ページ上部の [App access control (アプリのアクセスコントロール)] セクションで、[MANAGE THIRD-PARTY APP ACCESS (サードパーティー製アプリのアクセスを管理)] を選択します。
4. [Add app] (新規アプリの追加) を選択し、[OAuth App Name Or Client ID] (OAuth アプリケーション名またはクライアント ID) を選択します。
5. AWS リージョンの Amazon AppStream 2.0 OAuth クライアント ID を入力し、SEARCH を選択します。クライアント ID のリストについては、この手順の後にある表を参照してください。
6. 検索結果から [Amazon AppStream 2.0] を選択し、[Select (選択)] を選択します。
7. [Client ID (クライアント ID)] ページの [OAuth Client ID (OAuth クライアント ID)] で、一覧に正しい ID が表示されていることを確認し、ID の左側にあるチェックボックスをオンにします。
8. ページの右下にある [SELECT (選択)] を選択します。
9. Google Workspace 組織内のどの組織部門にアクセス許可を与えるかを設定します。
10. [Access to Google Data] (Google Data へのアクセス) で、[Trusted: Can access all Google services] (信頼済み: すべての Google サービスにアクセス可能) を選択し、[CONTINUE] (続行) を選択します。
11. 選択した内容が正しいことを確認し、問題がなければ [FINISH] (完了) を選択します。
12. 接続済みアプリのリストに、正しい OAuth ID を持つ Amazon AppStream 2.0 アプリが表示されていることを確認します。

Amazon AppStream 2.0 OAuth2 クライアント ID

リージョン	Amazon AppStream 2.0 OAuth クライアント ID
米国東部 (バージニア北部)	266080779488-15n5q5nkiclp6m524qibnmhmbsg0hk92.apps.googleusercontent.com
米国東部 (オハイオ)	723951369598-6tvd1f52g2qh0qa141o4k1avasvnj51i.apps.googleusercontent.com

リージョン	Amazon AppStream 2.0 OAuth クライアント ID
米国西部 (オレゴン)	1026466167591-i4jmemrggsjomp9tnkkcs5tniggfuijb.apps.googleusercontent.com
アジアパシフィック (ムンバイ)	325827353178-coqs1c374mf388ctl1rl1s374dc1bmb2.apps.googleusercontent.com
アジアパシフィック (ソウル)	562383781419-am1i2dnvt050tmdltsvr36i8l2js40dj.apps.googleusercontent.com
アジアパシフィック (シンガポール)	856871139998-4eia2n1db5j6gtv4c1rdte1fh1gec8vs.apps.googleusercontent.com
アジアパシフィック (シドニー)	151535156524-b889372osskprm4dt1clpm53mo3m9omp.apps.googleusercontent.com
アジアパシフィック (東京)	922579247628-qpl9kpihg3hu5dul2lphbjs4qbg6mjm2.apps.googleusercontent.com
カナダ (中部)	872792838542-t39aqh72jv895c89thtk6v83sl6jugm2.apps.googleusercontent.com
欧州 (フランクフルト)	643727794574-1se5360a77i84je9j3ap12obov1ib76q.apps.googleusercontent.com
欧州 (アイルランド)	599492309098-098muc7ofjfo9vua5rm5u9q2k3ml0k3j.apps.googleusercontent.com
欧州 (ロンドン)	682555519925-usbn2sk1ffgo8odgf23nj66ri71na0k5.apps.googleusercontent.com

リージョン	Amazon AppStream 2.0 OAuth クライアント ID
AWS GovCloud (米国東部)	20306576244-gqqkappmhv9fj06sdk7as60he89e7ce.apps.googleusercontent.com
	 Note AWS GovCloud (US) リージョンでの AppStream 2.0 の使用の詳細については、AWS GovCloud (US) ユーザーガイドのAmazon AppStream 2.0」を参照してください。
AWS GovCloud (米国西部)	996065833880-litfkb2vfd7c65nt7s24r7t81e5bc9b1.apps.googleusercontent.com
	 Note AWS GovCloud (US) リージョンでの AppStream 2.0 の使用の詳細については、AWS GovCloud (US) ユーザーガイドのAmazon AppStream 2.0」を参照してください。
南米 (サンパウロ)	891888628791-1ltbtedva29esqvqadiatlj4htcgcjfo.apps.googleusercontent.com

AppStream 2.0 ユーザーに対して Google ドライブを有効にするには、次の手順に従います。

スタックの作成時に Google ドライブを有効にするには

- [Amazon AppStream 2.0 でスタックを作成する](#) の手順に従って、[Google ドライブを有効にする] が選択されていること、Google Workspace アカウントに関連付けられている組織ドメインを 1 つ以上指定していることを確認します。

既存のスタックで Google ドライブを有効にするには

1. AppStream 2.0 コンソールを <https://console.aws.amazon.com/appstream2> で開きます。

2. 左側のナビゲーションペインで、[Stacks] を選択し、Google ドライブを有効にするスタックを選択します。
3. スタックリストの下で [ストレージ] を選択し、[Google Drive for Google Workspace を有効化] を選択します。
4. [Google Drive for Google Workspace を有効化] ダイアログボックスの、[Google Workspace ドメイン名] で、Google Workspace アカウントに関連付けられている組織ドメイン名を 1 つ以上入力します。別のドメインを指定するには、[Add another domain] を選択し、ドメインの名前を入力します。
5. ドメイン名を追加したら、[Enable (有効化)] を選択します。

Note

AppStream 2.0 ストリーミングセッション中の Google Drive の使用を開始するのに役立つようにユーザーに提供できるガイダンスについては、[Google ドライブを使用する](#) を参照してください。

AppStream 2.0 ユーザーの Google ドライブを無効にする

Google ドライブに保存したユーザーコンテンツを失うことなく、スタックに対して Google ドライブを無効にすることができます。スタックで Google ドライブを無効にすると、次のような影響があります。

- スタックのアクティブなストリーミングセッションに接続されているユーザーはエラーメッセージを受け取ります。Google ドライブにアクセスする権限がないことが通知されます。
- Google ドライブを無効化したスタックを使用する、新しいセッションでは、Google ドライブは表示されません。
- Google ドライブが無効になっている特定のスタックのみが影響を受けます。
- すべてのスタックに対して Google ドライブが無効になっている場合でも、AppStream 2.0 によって、Google Drive に保存されているユーザーコンテンツが削除されることはありません。

既存のスタックで Google ドライブを無効にするには、以下の手順を実行します。

既存のスタックの Google ドライブを無効にするには

1. AppStream 2.0 コンソールを <https://console.aws.amazon.com/appstream2> で開きます。

2. 左側のナビゲーションペインで、[Stacks] を選択し、Google ドライブを無効にするスタックを選択します。
3. スタックリストの下で [ストレージ] を選択し、[Google Drive for Google Workspace を有効化] オプションの選択を解除します。
4. [Google Drive for Google Workspace を無効化] ダイアログボックスで、CONFIRM (大文字と小文字は区別される) と入力します。選択内容を確認したら、[無効化] を選択します。

次の AppStream 2.0 ストリーミングセッションを開始すると、スタックのユーザーは、その後のセッション内から Google Drive フォルダにアクセスできなくなります。

AppStream 2.0 ユーザーに対して OneDrive for Business を有効にして管理する

AppStream 2.0 は、組織内のユーザーに対して以下の永続的ストレージをサポートしています。

- OneDrive for Business
- Google Drive for Google Workspace
- ホームフォルダ

組織に対して、1 つ以上のオプションを有効にすることができます。AppStream 2.0 スタックに対して OneDrive for Business を有効にすると、スタックのユーザーは OneDrive for Business アカウントを AppStream 2.0 にリンクできます。その後、ユーザーは、OneDrive for Business のアカウントにサインインして、アプリケーションストリーミングセッション中に OneDrive フォルダにアクセスできます。このセッション中に、OneDrive でユーザーがファイルやフォルダに行う変更は、自動的にバックアップ、同期されます。結果として、ストリーミングセッション外部から利用できるようになります。

Important

OneDrive for Business は、OneDrive ドメインのアカウントに対してのみ有効にすることができます。個人アカウントに対して設定することはできません。AppStream 2.0 では、エンドユーザーがアプリケーションに同意できるように Microsoft Azure Active Directory 環境を構成する必要があります。詳細については、Azure Active Directory [アプリケーション管理](#) ドキュメントの「[エンドユーザーがアプリケーションに同意する方法を設定する](#)」を参照してください。

管理者同意ワークフローを使用すると、管理者は管理者の承認を必要とするアプリケーションへのアクセスを許可できます。管理者の同意ワークフローが Azure Active Directory 環境で設定されている場合は、「」の手順に従って[AppStream 2.0 ユーザーの OneDrive を有効にする](#)、管理者の同意が必要なドメインを指定します。

 Note

OneDrive for Business は、Windows スタックに対しては有効にすることができますが、Linux スタックやマルチセッションフリートに関連付けられたスタックに対して有効にすることはできません。

内容

- [AppStream 2.0 ユーザーの OneDrive を有効にする](#)
- [AppStream 2.0 ユーザーの OneDrive を無効にする](#)

AppStream 2.0 ユーザーの OneDrive を有効にする

OneDrive を有効にする前に、以下を実行する必要があります。

- 有効な組織ドメインを持つアクティブな Microsoft Office 365 アカウントまたは OneDrive for Business アカウントと、そのドメインのユーザーを AppStream 2.0 と一緒に使用します。
- 関連付けられているフリートを使用して AppStream 2.0 を設定します。

このフリートでは、2018 年 7 月 26 日以降にリリースされたバージョンの AppStream 2.0 エージェントを使用するイメージを使用する必要があります。詳細については、[AppStream 2.0 エージェントのリリースノート](#)を参照してください。このフリートでは、インターネットにアクセスできることも必要です。

- Windows ベースのスタックを用意します。(Linux ベースのスタックはサポートされていません)。

AppStream 2.0 ユーザーに対して OneDrive を有効にするには、次の手順に従います。

スタックの作成時に OneDrive を有効にするには

- [Amazon AppStream 2.0 でスタックを作成する](#) の手順に従って、[Enable OneDrive (OneDrive を有効にする)] が選択されていること、OneDrive for Business アカウントに関連付けられている組織ドメインを 1 つ以上指定していることを確認します。

既存のスタックで OneDrive を有効にするには

1. AppStream 2.0 コンソールを <https://console.aws.amazon.com/appstream2> で開きます。
2. 左側のナビゲーションペインで、[Stacks] を選択し、OneDrive を有効にするスタックを選択します。
3. スタックリストの下で [Storage] をクリックし、[Enable OneDrive for Business] を選択します。
4. [Enable OneDrive for Business] ダイアログボックスの、[OneDrive domain name] に、OneDrive アカウントに関連付けられている組織ドメイン名を 1 つ以上入力します。別のドメインを指定するには、[Add another domain] を選択し、ドメインの名前を入力します。
5. ドメインごとに、OneDrive for Business アカウントを AppStream 2.0 にリンクする前に、ユーザーが管理者の同意を得る必要があるかどうかを指定できます。OneDrive for Business 管理者の同意はデフォルトで無効になっている必要があります。このチェックボックスをオンにすると、ユーザーは OneDrive for Business アカウントをリンクする前に管理者の同意を得るように求められます。
6. OneDrive ドメイン名を追加したら、[Enable (有効化)] を選択します。

ユーザーが AppStream 2.0 で OneDrive を使用するには、サードパーティーのウェブアプリケーションを使用して OneDrive アカウントをリンクするアクセス許可を付与する必要があります。そのためには、次のセクションのステップに従います。

 Important

エンドユーザーがアプリケーションに同意できるようにするには、Microsoft Azure Active Directory 環境を設定する必要があります。詳細については、Azure Active Directory [アプリケーション管理](#) ドキュメントの「[エンドユーザーがアプリケーションに同意する方法を設定する](#)」を参照してください。

AppStream 2.0 で OneDrive をリンクするアクセス許可をユーザーに付与する

ユーザーが OneDrive for Business を AppStream 2.0 にリンクできるようにするには、Office 365 または OneDrive for Business 管理コンソールで統合アプリを有効にする必要があります。

1. Business 管理コンソールで、Office 365 または OneDrive にサインインします。
2. コンソールの左のナビゲーションペインで [Settings (設定)]、[Services & add-ins] の順に選択します。
3. サービスとアドインのリストから、[Integrated Apps] を選択します。
4. [Integrated apps] ページで、組織のユーザーがサードパーティーのウェブアプリを使用して、Office 365 情報にアクセスすることを許可するオプションをオンにします。

Note

AppStream 2.0 ストリーミングセッション中の OneDrive の使用を開始するのに役立つようにユーザーに提供できるガイダンスについては、[OneDrive for Business を使用する](#) を参照してください。

AppStream 2.0 ユーザーの OneDrive を無効にする

OneDrive に保存したユーザーコンテンツを失うことなく、スタックに対して OneDrive を無効にすることができます。スタックで OneDrive を無効にすると、次のような影響があります。

- スタックのアクティブなストリーミングセッションに接続されているユーザーはエラーメッセージを受け取ります。OneDrive にアクセスするアクセス許可がないことが通知されます。
- OneDrive を無効化したスタックを使用する、新しいセッションでは、OneDrive は表示されません。
- OneDrive が無効になっている特定のスタックのみが影響を受けます。
- すべてのスタックに対して OneDrive が無効になっている場合でも、AppStream 2.0 によって、OneDrive に保存されているユーザーコンテンツが削除されることはありません。

既存のスタックで OneDrive を無効にするには、以下の手順を実行します。

既存のスタックで OneDrive を無効にするには

1. AppStream 2.0 コンソールを <https://console.aws.amazon.com/appstream2> で開きます。

2. 左側のナビゲーションペインで、[Stacks] を選択し、OneDrive を無効にするスタックを選択します。
3. スタックリストの下で [Storage] をクリックし、[Enable OneDrive for Business] オプションを解除します。
4. [Disable OneDrive for Business] ダイアログボックスで、CONFIRM (大文字と小文字は区別される) と入力します。選択内容を確認したら、[Disable] を選択します。

次の AppStream 2.0 ストリーミングセッションを開始すると、スタックのユーザーは、その後のセッション内から OneDrive フォルダにアクセスできなくなります。

AppStream 2.0 ユーザーのカスタム共有フォルダ (サーバーメッセージブロック (SMB) ネットワークドライブ) の有効化と管理

組織に対して、1 つ以上のオプションを有効にすることができます。サーバーメッセージブロック (SMB) ネットワークドライブを有効にしてマッピングすると、複数のユーザーが Windows AppStream 2.0 セッションから同じデータにアクセスできます。これらのセッション中にユーザーが SMB ネットワークドライブに加えた変更は、自動的にバックアップおよび同期されます。

Note

- サーバーメッセージブロック (SMB) ネットワークドライブマッピングは、ドメインに参加しているフリートでのみサポートされています。
- この機能を使用するには、2024 年 9 月 18 日以降にリリースされた AppStream 2.0 エージェントを利用する AppStream 2.0 イメージを使用する必要があります。詳細については、「[AppStream 2.0 エージェントのバージョンを管理する](#)」および「[AppStream 2.0 ベースイメージとマネージド型イメージの更新に関するリリースノート](#)」を参照してください。

サーバーメッセージブロック (SMB) ネットワークドライブをマッピングする前に、インバウンドルールについて、ユーザーがフリートへの接続に使用するセキュリティグループが TCP ポート 445 (SMB プロトコル) をドメインコントローラーとセキュリティグループに公開していることを確認します。

内容

- [マップサーバーメッセージブロック \(SMB\) ネットワークドライブ](#)

マップサーバーメッセージブロック (SMB) ネットワークドライブ

SMB のターゲットネットワーク下にある任意のマシンを使用できます。セッションスクリプトを使用して設定する場合は、まず、ユーザーがログオンしたときに呼び出されるスクリプトを作成する必要があります。セッションスクリプトはイメージごとに設定されます。

サーバーメッセージブロック (SMB) ネットワークドライブをマッピングするには、次の手順を実行します。

ステップ 1: サービスが実行されていることを確認する

スタートメニューから [services.msc] を開き、次のサービスがすべて実行されていることを確認します。

- DNS クライアント
- 関数検出リソースの発行
- SSDP 検出
- UPnP デバイスホスト

ステップ 2: SMB フォルダを作成する

ファイルエクスプローラーを使用して SMB を作成できます。

ファイルエクスプローラーを使用して SMB 共有フォルダを設定するには

1. SMB フォルダを右クリックし、[プロパティ]、[共有] を選択します。
2. [高度な共有] を選択します。
3. [高度な共有] で、[このフォルダを共有] をチェックし、[アクセス許可] を選択します。
4. すべてのユーザーにアクセス許可を付与する場合は、デフォルト設定のままにします。

特定のユーザーを追加する場合は、[アクセス許可の共有] で、[全員]、[削除] を選択します。次に、[追加] を選択し、ファイル共有にアクセスするユーザーまたはグループを入力します。

追加するユーザーまたはグループごとに、[許可] を選択して、[フルコントロール]、[変更]、または [読み取りアクセス許可] を割り当てます。

5. [適用]、[OK]、[OK]、[閉じる] を選択します。

ステップ 3: SMB がドメイン内でアクセス可能であることを確認する

同じセキュリティグループを使用し、同じドメインに結合する別のサーバーからファイルエクスプローラーを開きます。ネットワークパスフォルダに移動して、提供されたネットワークパス経由でネットワーク共有にアクセスします。[プロパティ]、[共有]、[ネットワークパス] を選択します。

ステップ 4: ユーザーがローカル/ドメイングループポリシーからシンボリックリンクを作成できるようにする

ユーザーのローカル/ドメイングループポリシーからのシンボリックリンクの作成を有効にして、グループポリシーで定義されているセッションスクリプトまたはログオンスクリプトを確認します。これにより、ユーザーアクセス許可を持つスクリプトをステップ 5 で作成できます。

ユーザーがローカル/ドメイングループポリシーからシンボリックリンクを作成できるようにするには

1. このポリシーの定義に使用される GPO では、[コンピュータ設定]、[Windows 設定]、[セキュリティ設定]、[ユーザー権限の割り当て]、[ポリシー]、[シンボリックリンクの作成] を選択します。次に、追加するユーザーのアクセス許可を更新します。シンボリックリンクの作成の詳細については、「[Create symbolic links](#)」を参照してください。
2. デフォルトでは、リモートからリモートへの (例えば、別の同様のシンボリックリンク内のネットワーク共有へのシンボリックリンクマッピング) およびリモートからローカルへの (例えば、シンボリックリンクからネットワーク共有へのシンボリックリンクマッピング) アクセスは無効になっています。シンボリックリンクマッピングが必要な場合は、以下のコマンドを実行します。
 - リモートからリモートへのアクセスを有効にする場合 - `fsutil behavior set SymlinkEvaluation R2R:1`
 - リモートからローカルへのアクセスを有効にする場合 - `fsutil behavior set SymlinkEvaluation R2L:1`

ステップ 5: ユーザーがログオンしたときに呼び出されるスクリプトを作成する

AppStream 2.0 セッションスクリプトまたは GPO ログオンスクリプトを使用して、ユーザーがログオンしたときに呼び出されるスクリプトを作成します。AppStream 2.0 セッションスクリプトを使用することを選択した場合、セッションスクリプトはその特定の AppStream 2.0 イメージにのみ適用されます。GPO ログオンスクリプトを使用する場合、GPO はドメイン/OU に適用され、フリー

に設定できます。これにより、所有するすべてのイメージにスクリプトを設定する必要はありません。

オプション 1: セッションスクリプトを使用して SMB 共有フォルダをマイファイルにマウントする (Powershell を使用)

セッションスクリプトを使用して SMB 共有フォルダをマイファイルにマウントするには (Powershell を使用)

1. ユーザーアクセス許可を正常に定義したら、ユーザーコンテキストまたはシステムコンテキストを使用して次のサンプルスクリプトを設定します。

ユーザーコンテキストを使用する config.json スクリプトの例を次に示します。

```
"SessionStart": {
  "executables": [
    {
      "context": "system",
      "filename": "",
      "arguments": "",
      "s3LogEnabled": true
    },
    {
      "context": "user",
      "filename": "C:\\\\Windows\\\\System32\\\\WindowsPowerShell\\\\v1.0\\\\
powershell.exe",
      "arguments": "-File \\\\\"C:\\\\AppStream\\\\SessionScripts\\\\userStart.ps1\\\\\"",
      "s3LogEnabled": true
    }
  ],
  "waitingTime": 30
}
```

以下は、システムコンテキストを使用するスクリプトの例です。

```
"SessionStart": {
  "executables": [
    {
      "context": "system",
      "filename": "C:\\\\Windows\\\\System32\\\\WindowsPowerShell\\\\v1.0\\\\
powershell.exe",
      "arguments": "-File \\\\\"C:\\\\AppStream\\\\SessionScripts\\\\systemStart.ps1\\\\\"",
      "s3LogEnabled": true
    }
  ]
}
```

```
    },  
    {  
        "context": "user",  
        "filename": "",  
        "arguments": "",  
        "s3LogEnabled": true  
    }  
],  
"waitingTime": 30
```

2. マルチセッションフリートを使用している場合は、システム環境変数

\$env:AppStream_Session_UserName を使用してユーザーのマイファイルフォルダに移動できます。これにより、システムコンテキスト \$env:USERNAME を使用する際に、ユーザー名の代わりに Admin にマッピングできます。

```
# Define the target application path  
$targetPathes = "<SMB-PATH>"  
  
# Define the shortcut location  
$symlinkLocation = "C:\Users\$Env:AppStream_Session_UserName\My Files\Custom  
Folder"  
  
# Create the junction for Custom Home Folder under MyFiles  
New-Item -ItemType SymbolicLink -Path $symlinkLocation -Target $targetPathes
```

オプション 2: GPO ログオンスクリプトを使用して SMB 共有フォルダをマイファイルの下にマウントする

1. ファイルまたはフォルダにシンボリックリンクを作成して、SMB 共有フォルダをマウントします。詳細については、「[例 7: ファイルまたはフォルダへのシンボリックリンクを作成する](#)」を参照してください。
2. [ユーザーログオンスクリプトを割り当てます。](#)
3. 次のスクリプトを追加して、マイファイルの下にカスタムホームフォルダのジャンクションを作成します。

```
# Define the target application path  
$targetPathes = "<SMB-PATH>"  
  
# Define the shortcut location  
$symlinkLocation = "C:\Users\$env:Username\My Files\Custom Folder"
```

```
# Create the junction for Custom Home Folder under MyFiles
New-Item -ItemType SymbolicLink -Path $symlinkLocation -Target $targetPaths
```

Windows Server 2022 イメージを使用している場合、ログオンスクリプトが正常に完了するまでマイファイルフォルダが作成されない問題が発生する可能性があります。これにより、SMB マウントオペレーションがログオンスクリプトを介して実行されるとタイムアウトが発生する可能性があります。この問題を解決するには、SMB もマウントしながら、以下を実行してログオンスクリプトを使用して独立したプロセス (Start-Process) をトリガーします。

- a. ログオンスクリプトを作成します。

```
# Define the log file path
$logFilePath = "<This-is-where-your-log-files-are-saved>"

# Function to write log messages
function Write-Log {
    param (
        [string]$message
    )
    $timestamp = get-date -format "yyyy-MM-dd HH:mm:ss"
    $logMessage = "$timestamp - $message"
    $logMessage | Out-File -FilePath $logFilePath -Append -Encoding UTF8
}

try {
    Write-Log "Setting execution policy..."
    Set-ExecutionPolicy -ExecutionPolicy Bypass -Scope Process -Force
    Write-Log "Unblocking logon script file..."
    $filePath = "<This-is-where-your-actual-logon-script-is-linked>"
    Unblock-File -Path $filePath
    Write-Log "Running actual logon script..."
    Start-Process -FilePath 'Powershell.exe' -ArgumentList "-File `"$filePath`""
} catch {
    Write-Log "An error occurred: $_" "ERROR"
}
```

- b. 必要に応じて、グループポリシーを使用して、このログオンスクリプトの遅延設定を更新します。詳細については、「[ログオン スクリプトの遅延を構成する](#)」を参照してください。ログオンスクリプトの遅延時間は、非同期ログオンスクリプトをトリガーするまでの遅延時間です。デフォルトの遅延は 5 分です。

- c. フリートを再起動して、ログオンスクリプトの遅延を適用します。

AppStream 2.0 ユーザーのアプリケーション設定の永続化を有効にする

AppStream 2.0 は、Windows ベースのスタックに対して永続的なアプリケーション設定をサポートしています。つまり、ユーザーのアプリケーションのカスタマイズや Windows 設定は各ストリーミングセッション後に自動的に保存され、次のセッションで適用されます。ユーザーが設定できる永続的なアプリケーション設定の例としては、ブラウザのお気に入り、設定、ウェブページのセッション、アプリケーション接続プロファイル、プラグイン、UI のカスタマイズなどが挙げられます。これらの設定は、アプリケーション設定の永続化が有効になっている AWS リージョン内のアカウントの Amazon Simple Storage Service (Amazon S3) バケットに保存されます。これらは、各 AppStream 2.0 ストリーミングセッションで使用できます。

Note

アプリケーション設定の永続化の有効化は、現在 Linux ベースのスタックではサポートされていません。

Note

S3 バケットに保存されているデータには、標準 Amazon S3 料金が適用される場合があります。詳細については、[Amazon S3 の料金](#) を参照してください。

内容

- [アプリケーション設定の永続化の仕組み](#)
- [アプリケーション設定の永続化の有効化](#)
- [ユーザーのアプリケーション設定 VHD の管理](#)

アプリケーション設定の永続化の仕組み

永続的なアプリケーション設定は Virtual Hard Disk (VHD) ファイルに保存されます。このファイルは、アプリケーション設定の永続化が有効になっているスタックから、ユーザーが初めてアプリケーションをストリーミングしたときに作成されます。スタックに関連付けられているフリートがデフォルトのアプリケーションおよび Windows 設定が含まれているイメージに基づいている場合、このデ

フォルト設定がユーザーの最初のストリーミングセッションで使用されます。デフォルト設定の詳細については、[チュートリアル: AppStream 2.0 コンソールを使用してカスタム AppStream 2.0 イメージを作成する](#) の Step 3: Create Default Application and Windows Settings を参照してください。

ストリーミングセッションが終了すると、VHD はアンマウントされ、アカウント内の Amazon S3 バケットにアップロードされます。バケットは、AWS リージョンのスタックで永続的なアプリケーション設定を初めて有効にしたときに作成されます。そのバケットは AWS アカウントとリージョンに固有のものです。VHD は、伝送時には Amazon S3 SSL エンドポイントを使用して暗号化され、保管時には [AWS マネージド CMK](#) を使用して暗号化されます。

VHD は、C:\Users\%username% および D:\%username% の両方でストリーミングインスタンスにマウントされます。インスタンスが Active Directory ドメインに参加していない場合、Windows ユーザー名は PhotonUser になります。インスタンスが Active Directory ドメインに参加している場合、Windows ユーザー名はログインユーザーの名前になります。

アプリケーション設定の永続性は複数のオペレーティングシステムのバージョン間では機能しません。たとえば、スタックでアプリケーション設定の永続性を有効にして、そのスタックが Windows Server 2012 R2 イメージを使用するフリートに関連付けられている場合、別のオペレーティングシステム (Windows Server 2016 など) で実行されるイメージを使用するようにフリートを更新すると、以前のストリーミングセッションの設定はそのスタックのユーザーには保存されません。代わりに、新しいイメージを使用するようにフリートを更新した後、ユーザーがフリートインスタンスからストリーミングセッションを起動するときに、新しい Windows ユーザープロファイルが作成されます。ただし、イメージで同じオペレーティングシステムに更新を適用すると、以前のストリーミングセッションからのユーザーのカスタマイズと設定が保存されます。同じオペレーティングシステムへの更新がイメージに適用されると、ユーザーがフリートからストリーミングセッションを起動するときに、同じ Windows ユーザープロファイルが使用されます。

Important

AppStream 2.0 は、ストリーミングインスタンスが Microsoft Active Directory ドメインに参加している場合にのみ、[Microsoft Data Protection API](#) に依存するアプリケーションをサポートします。ストリーミングインスタンスが Active Directory ドメインに参加していない場合、Windows ユーザーの PhotonUser はフリートインスタンスごとに異なります。DPAPI セキュリティモデルの機能上の理由から、このシナリオで DPAPI を使用するアプリケーションではユーザーのパスワードは保持されません。ストリーミングインスタンスが Active Directory ドメインに参加していて、そのユーザーがドメインユーザーである場合、Windows ユーザー名はログインしているユーザーの名前であり、DPAPI を使用するアプリケーションではユーザーのパスワードは保持されます。

AppStream 2.0 は、次のフォルダを除いて、このパスにあるすべてのファイルとフォルダを自動的に保存します。

- 問い合わせ
- Desktop
- ドキュメント
- ダウンロード
- リンク
- 画像
- Saved Games
- 検索
- 動画

これらのフォルダ外で作成されたファイルとフォルダは、VHD 内に保存され、Amazon S3 と同期されます。デフォルトの VHD 最大サイズは、Elastic フリートの場合は 1 GB、常時オンフリートおよびオンデマンドフリートの場合は 5 GB です。保存された VHD のサイズは、それに含まれるファイルとフォルダの合計サイズです。AppStream 2.0 は、ユーザーの HKEY_CURRENT_USER レジストリハイブを自動的に保存します。新規ユーザー (Amazon S3 にプロファイルが存在しないユーザー) の場合、AppStream 2.0 はデフォルトのプロファイルを使用して初期プロファイルを作成します。このプロファイルは、Image Builder の C:\users\default に作成されます。

 Note

ストリーミングセッションが開始する前に、VHD 全体をストリーミングインスタンスにダウンロードする必要があります。このため、VHD に大量のデータが保持されていると、ストリーミングセッションの開始が遅れる場合があります。詳細については、「[アプリケーション設定の永続化を有効にするためのベストプラクティス](#)」を参照してください。

アプリケーション設定の永続化を有効にする場合、設定グループを指定する必要があります。設定グループは、このスタックに保存されているどのアプリケーション設定をストリーミングセッションで使用するかを決定します。AppStream 2.0 は、AWS アカウントの S3 バケット内に個別に保存される設定グループの新しい VHD ファイルを作成します。設定グループを複数のスタック間で共有すると、同じアプリケーション設定が各スタックで使用されます。スタックが独自のアプリケーション設定を必要とする場合は、このスタック限定の設定グループを指定します。

アプリケーション設定の永続化の有効化

AppStream 2.0 ユーザーのアプリケーション設定の永続化を有効にする方法については、以下のトピックを参照してください。

内容

- [アプリケーション設定の永続化を有効にするための前提条件](#)
- [アプリケーション設定の永続化を有効にするためのベストプラクティス](#)
- [アプリケーション設定の永続化を有効にする方法](#)

アプリケーション設定の永続化を有効にするための前提条件

アプリケーション設定の永続化を有効にするには、まず、以下のことを行う必要があります。

- Amazon S3 アクションに対する正しい AWS Identity and Access Management (IAM) アクセス許可があることを確認します。詳細については、[Amazon AppStream 2.0 向け Identity and Access Management](#) の「ホームフォルダ用の IAM ポリシーと Amazon S3 バケット」を参照してください。
- 2017 年 12 月 7 日以降 AWS に よって公開されたベースイメージから作成されたイメージを使用します。リリースされた AWS ベースイメージの最新リストについては、「」を参照してください[AppStream 2.0 ベースイメージとマネージド型イメージの更新に関するリリースノート](#)。
- この機能を有効にするスタックと (2018 年 8 月 29 日以降にリリースされたバージョンの AppStream 2.0 エージェントを使用するイメージに基づく) フリートを関連付けます。詳細については、「[AppStream 2.0 エージェントのリリースノート](#)」を参照してください。
- インターネットアクセスまたは Amazon S3 の VPC エンドポイントを設定して、Virtual Private Cloud (VPC) から Amazon S3 へのネットワーク接続を有効にします。詳細については、[Amazon AppStream 2.0 のネットワーキングとアクセス](#) の「ホームフォルダと VPC エンドポイント」セクションを参照してください。

アプリケーション設定の永続化を有効にするためのベストプラクティス

インスタンスへのインターネットアクセスを提供せずに、アプリケーション設定の永続化を有効にするには、VPC エンドポイントを使用します。このエンドポイントは、AppStream 2.0 インスタンスの接続先の VPC 内に存在する必要があります。エンドポイントへの AppStream 2.0 アクセスを有効にするには、カスタムポリシーをアタッチする必要があります。カスタムポリシーを作成する方法については、[Amazon AppStream 2.0 のネットワーキングとアクセス](#) の「ホームフォルダと VPC

エンドポイント」を参照してください。プライベート Amazon S3 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 Endpoints](#) および [Endpoints for Amazon S3](#) を参照してください。

アプリケーション設定の永続化を有効にする方法

AppStream 2.0 コンソール、AppStream 2.0 AppStream API、AWS SDK、または コマンドライン インターフェイス (CLI) AWS を使用して、スタックの作成中またはスタックの作成後にアプリケーション設定の永続化を有効または無効にできます。AWS リージョンごとに、永続的なアプリケーション設定はアカウントの S3 バケットに保存されます。

リージョンのスタックに対してアプリケーション設定の永続化を AWS 初めて有効にすると、AppStream 2.0 は同じリージョンの AWS アカウントに S3 バケットを作成します。同じバケットに、その AWS リージョン内のすべてのユーザーとすべてのスタックのアプリケーション設定 VHD ファイルが保存されます。詳細については、[ユーザーのアプリケーション設定 VHD の管理](#) の Amazon S3 バケットストレージ を参照してください。

スタックの作成時にアプリケーション設定の永続化を有効にするには

- 「[Amazon AppStream 2.0 でスタックを作成する](#)」の手順に従い、[Enable Application Settings Persistence (アプリケーション設定の永続化を有効にする)] が選択されていることを確認します。

既存のスタックでアプリケーション設定の永続化を有効にするには

- AppStream 2.0 コンソール (<https://console.aws.amazon.com/appstream2>) を開きます。
- 左のナビゲーションペインで、[Stacks (スタック)] を選択し、アプリケーション設定の永続化を有効にするスタックを選択します。
- スタックのリストの下で、[User Settings (ユーザー設定)]、[Application Settings Persistence (アプリケーション設定の永続化)]、[Edit (編集)] の順に選択します。
- [Application Settings Persistence (アプリケーション設定の永続化)] ダイアログボックスで、[Enable Application Settings Persistence (アプリケーション設定の永続化を有効にする)] を選択します。
- 現在の設定グループを選択して確定するか、新しい設定グループの名前を入力します。完了したら、[Update (更新)] を選択します。

これにより、新しいストリーミングセッションでアプリケーション設定の永続化が有効になります。

ユーザーのアプリケーション設定 VHD の管理

AppStream 2.0 ユーザーのアプリケーション設定の Virtual Hard Disks (VHD) ファイルを管理する方法については、以下のトピックを参照してください。

内容

- [Amazon S3 バケットのストレージ](#)
- [ユーザーのアプリケーション設定のリセット](#)
- [Amazon S3 オブジェクトのバージョニングを有効にしてユーザーのアプリケーション設定を元に戻す](#)
- [アプリケーション設定 VHD のサイズ拡大](#)

Amazon S3 バケットのストレージ

アプリケーション設定の永続化を有効にすると、ユーザーのアプリケーションのカスタマイズと Windows 設定は、AWS アカウントで作成された Amazon S3 バケットに保存されている Virtual Hard Disk (VHD) ファイルに自動的に保存されます。AWS リージョンごとに、アカウントおよびリージョンに固有のバケットが AppStream 2.0 によってアカウント内に作成されます。ユーザーが行ったすべてのアプリケーション設定が該当リージョンのバケットに保存されます。

これらの S3 バケットを管理するための設定タスクは一切不要です。AppStream 2.0 サービスによって完全に管理されます。各バケットに保存された VHD ファイルは、伝送時には Amazon S3 の SSL エンドポイントを使用して暗号化され、保管時には [AWS マネージド CMK](#) を使用して暗号化されます。バケットは、以下にあるような特定の形式で命名されます。

```
appstream-app-settings-region-code-account-id-without-hyphens-random-identifier
```

region-code

これは、アプリケーション設定の永続化を使用してスタックが作成される AWS リージョンコードです。

account-id-without-hyphens

AWS アカウント ID。ランダムな識別子により、該当リージョンで他のバケットとの競合が発生することはありません。バケット名の最初の部分 appstream-app-settings は、複数のアカウントやリージョンにまたがる場合でも変更されません。

たとえば、アカウント番号 123456789012 で、米国西部 (オレゴン) リージョン (us-west-2) のスタックに対してアプリケーション設定の永続化を有効にすると、AppStream 2.0 は、該当リージョンのアカウント内に次に示す名前で Amazon S3 バケットを作成します。適切なアクセス許可を持つ管理者のみが、このバケットを削除できます。

```
appstream-app-settings-us-west-2-1234567890123-abcdefg
```

アプリケーション設定の永続化を無効にしても、S3 バケットに保存された VHD は削除されません。設定 VHD を完全に削除するには、Amazon S3 コンソールまたは API を使用して、ユーザーまたは適切なアクセス許可を持つ別の管理者が削除する必要があります。AppStream 2.0 は、バケットの誤った削除を防止するバケットポリシーを追加します。詳細については、[Amazon AppStream 2.0 向け Identity and Access Management](#) の「アプリケーション設定の永続化用の IAM ポリシーと Amazon S3 バケット」を参照してください。

アプリケーション設定の永続化を有効にすると、設定 VHD を保存するために設定グループごとに固有のフォルダが作成されます。S3 バケットのフォルダの階層は、次のセクションで説明するように、ユーザーがストリーミングセッションを起動する方法によって異なります。

アカウントの S3 バケットで設定 VHD が保存されているフォルダへのパスは、次の構造になります。

```
bucket-name/Windows/prefix/settings-group/access-mode/user-id-SHA-256-hash
```

bucket-name

ユーザーのアプリケーション設定が保存されている S3 バケットの名前。名前の形式については、このセクションで先ほど説明しました。

prefix

Windows バージョン固有のプレフィックス。例えば、v4 for Windows Server 2012 R2 です。

settings-group

設定グループの値。この値は、同じアプリケーション設定を共有する 1 つ以上のスタックに適用されます。

access-mode

ユーザーの ID メソッド: AppStream 2.0 API または CLI の場合は custom、SAML の場合は federated、ユーザープールのユーザーの場合は userpool。

user-id-SHA-256-hash

ユーザー固有のフォルダ名。この名前は、ユーザー ID から生成された小文字の SHA-256 ハッシュ 16 進数文字列を使用して作成されます。

次のフォルダ構造の例は、ユーザー ID が、ID が testuser@mydomain.com、米国西部 (オレゴン) リージョン (us-west-2) AWS アカウント 123456789012 の設定グループ test-stack で API または CLI を使用してアクセスされるストリーミングセッションに適用されます。

```
appstream-app-settings-us-west-2-1234567890123-abcdefg/Windows/v4/test-stack/custom/a0bcb1da11f480d9b5b3e90f91243143eac04cfccfbdc777e740fab628a1cd13
```

ユーザーのフォルダを確認するには、ウェブサイトを使用するか、オンラインで入手できるオープンソースコーディングライブラリを使用して、ユーザー ID の小文字の SHA-256 ハッシュ値を生成します。

ユーザーのアプリケーション設定のリセット

ユーザーのアプリケーション設定をリセットするには、AWS アカウントの S3 バケットから VHD および関連するメタデータファイルを検索して削除する必要があります。ユーザーのアクティブなストリーミングセッションが進行中は、この操作を実行しないでください。ユーザーの VHD とメタデータファイルを削除すると、次回にユーザーがアプリケーション設定の永続化が有効になっているストリーミングインスタンスからセッションを起動すると、AppStream 2.0 によって当該ユーザーの新しい設定 VHD が作成されます。

ユーザーのアプリケーション設定をリセットするには

1. <https://console.aws.amazon.com/s3/> で Amazon S3 コンソールを開きます。
2. [Bucket name (バケット名)] リストで、リセットするアプリケーション設定 VHD が含まれている S3 バケットを選択します。
3. VHD が含まれているフォルダを見つけます。S3 バケットのフォルダ構造内を移動する詳しい方法については、このトピックの前半にある「Amazon S3 バケットのストレージ」を参照してください。
4. [名前] のリストで、VHD と REG の横にあるチェックボックスをオンにし、[詳細]、[削除] の順に選択します。
5. [Delete objects (オブジェクトの削除)] ダイアログボックスで、VHD と REG が表示されていることを確認し、[削除] を選択します。

該当する設定グループに基づいてアプリケーション設定の永続化が有効になっているフリートから次回にユーザーがストリーミングすると、新しいアプリケーション設定 VHD が作成されます。この VHD は、セッションの最後に S3 バケットに保存されます。

Amazon S3 オブジェクトのバージョニングを有効にしてユーザーのアプリケーション設定を元に戻す

Amazon S3 オブジェクトのバージョニングとライフサイクルポリシーを使用して、ユーザーによるアプリケーション設定の変更を管理できます。Amazon S3 オブジェクトのバージョニングを使用すると、あらゆるバージョンの設定 VHD を保持、取得、復元できます。これにより、意図しないユーザーのアクションとアプリケーションの障害の両方から復旧できます。バージョニングを有効にすると、各ストリーミングセッション後に、新しいバージョンのアプリケーション設定 VHD が Amazon S3 と同期されます。新しいバージョンは以前のバージョンを上書きしないため、ユーザーの設定に問題が生じた場合は、以前のバージョンの VHD に戻すことができます。

Note

各バージョンのアプリケーション設定 VHD は、別個のオブジェクトとして Amazon S3 に保存され、相応に課金されます。

S3 バケットでのオブジェクトのバージョニングは、デフォルトでは有効にならないため、明示的に有効にする必要があります。

アプリケーション設定 VHD でオブジェクトのバージョニングを有効にするには

1. <https://console.aws.amazon.com/s3/> で Amazon S3 コンソールを開きます。
2. [Bucket name (バケット名)] リストで、オブジェクトのバージョニングを有効にするアプリケーション設定 VHD が含まれている S3 バケットを選択します。
3. [Properties] (プロパティ) を選択します。
4. [Versioning (バージョニング)]、[Enable versioning (バージョニングの有効化)]、[Save (保存)] の順に選択します。

古いバージョンのアプリケーション設定 VHD を失効させるには、Amazon S3 ライフサイクルポリシーを使用できます。詳細については、Amazon Simple Storage Service ユーザーガイドの「[S3 バケットのライフサイクルポリシーを作成する方法を教えてください](#)」を参照してください。

ユーザーのアプリケーション設定 VHD を前のバージョンに戻すには

ユーザーのアプリケーション設定 VHD を前のバージョンに戻すには、該当する S3 バケットから以降のバージョンの VHD を削除します。ユーザーがアクティブなストリーミングセッションを進行中は、この操作を実行しないでください。

1. <https://console.aws.amazon.com/s3/> で Amazon S3 コンソールを開きます。
2. [Bucket name (バケット名)] リストで、前のバージョンに戻すユーザーのアプリケーション設定 VHD が含まれている S3 バケットを選択します。
3. VHD が含まれているフォルダを見つけて選択します。S3 バケットのフォルダ構造内を移動する詳しい方法については、このトピックの前半にある「Amazon S3 バケットのストレージ」を参照してください。

フォルダを選択すると、設定 VHD および関連するメタデータファイルが表示されます。

4. VHD とメタデータファイルのバージョンを一覧表示するには、[Show (表示)] を選択します。
5. 以前に戻す VHD のバージョンを見つけます。
6. [Name (名前)] リストで、以降のバージョンの VHD および関連するメタデータファイルの横にあるチェックボックスをオンにし、[More (詳細)]、[Delete (削除)] の順に選択します。
7. 前のバージョンに戻すアプリケーション設定 VHD および関連するメタデータファイルが以降のバージョンであることを確認します。

該当する設定グループに基づいてアプリケーション設定の永続化が有効になっているフリートから次回にユーザーがストリーミングを行うと、前のバージョンに戻したユーザー設定が表示されます。

アプリケーション設定 VHD のサイズ拡大

デフォルトの VHD 最大サイズは、Elastic フリートの場合は 1 GB、常時オンフリートおよびオンデマンドフリートの場合は 5 GB です。ユーザーがアプリケーション設定の領域を増やす必要がある場合は、該当するアプリケーション設定 VHD を Windows コンピュータにダウンロードして拡大できます。次に、S3 バケット内の現在の VHD を、拡大したものに置き換えます。ユーザーがアクティブなストリーミングセッションを進行中は、この操作を実行しないでください。

Note

Virtual Hard Disk (VHD) の物理サイズを減らすには、セッションを終了する前にごみ箱を空にします。これにより、アップロードとダウンロードの時間も短縮され、全体的なユーザーエクスペリエンスが向上します。

アプリケーション設定 VHD のサイズを拡大するには

Note

ユーザーがアプリケーションのストリーミングを行う前に、VHD 全体をダウンロードする必要があります。アプリケーション設定 VHD のサイズを拡大すると、ユーザーがアプリケーションのストリーミングセッションを開始するまでの所要時間が長くなる場合があります。

1. <https://console.aws.amazon.com/s3/> で Amazon S3 コンソールを開きます。
2. [Bucket name (バケット名)] リストで、拡大するアプリケーション設定 VHD が含まれている S3 バケットを選択します。
3. VHD が含まれているフォルダを見つけて選択します。S3 バケットのフォルダ構造内を移動する詳しい方法については、このトピックの前半にある「Amazon S3 バケットのストレージ」を参照してください。

フォルダを選択すると、設定 VHD および関連するメタデータファイルが表示されます。

4. Profile.vhdx ファイルを Windows コンピュータのディレクトリにダウンロードします。ダウンロードが完了しても、ブラウザを閉じないでください。拡大した VHD をアップロードするためにブラウザを後で再び使用します。
5. Diskpart を使用して VHD のサイズを 7 GB に拡大するには、管理者としてコマンドプロンプトを開き、以下のコマンドを入力します。

```
diskpart
```

```
select vdisk file="C:\path\to\application\settings\profile.vhdx"
```

```
expand vdisk maximum=7000
```

6. 次に、以下の Diskpart コマンドを入力して、VHD を見つけてアタッチし、ボリュームを一覧表示します。

```
select vdisk file="C:\path\to\application\settings\profile.vhdx"

attach vdisk

list volume
```

出力で、「AppStreamUsers」というラベルが付いたボリューム番号を書き留めます。次のステップで、このボリュームを選択して拡大します。

7. 次のコマンドを入力します。

```
select volume ###
```

は出力に表示されたボリュームの番号です。

8. 次のコマンドを入力します。

```
extend
```

9. 以下のコマンドを入力して、VHD のパーティションのサイズが正常に拡大したこと (この例では 2 GB) を確認します。

```
diskpart
```

```
select vdisk file="C:\path\to\application\settings\profile.vhdx"

list volume
```

10. 次のコマンドを入力して VHD をデタッチし、アップロードできるようにします。

```
detach vdisk
```

11. Amazon S3 コンソールのブラウザに戻り、[Upload (アップロード)]、[Add files (ファイルの追加)] の順に選択し、拡大した VHD を選択します。

12. [アップロード] を選択します。

VHD をアップロードすると、該当する設定グループに基づいてアプリケーション設定の永続化が有効になっているフリートから次回にユーザーがストリーミングを行ったときに、拡大したアプリケーション設定 VHD を使用されます。

AppStream 2.0 ユーザー用のリージョン設定を有効にする

AppStream 2.0 を使用すると、お客様またはお客様のユーザーは、ユーザーの場所や言語に固有の特定の Windows 設定を構成できます。AppStream 2.0 では、Linux イメージの作成中にリージョン設定を指定することもできます。詳細については、「[Linux イメージの日本語サポートを有効にする](#)」を参照してください。

内容

- [AppStream 2.0 ユーザーのためのデフォルトのリージョン設定を構成する](#)
- [AppStream 2.0 ユーザーが自分のリージョンを構成できるようにする](#)

AppStream 2.0 ユーザーのためのデフォルトのリージョン設定を構成する

Note

このページの説明は、Windows フリートにのみ適用されます。デフォルトのリージョン設定は、Elastic フリートではサポートされていません。

AppStream 2.0 では、Windows スタック内のユーザーが、自分の所在地や言語に固有の設定を使用するようにストリーミングセッションを設定できます。詳細については、「[AppStream 2.0 ユーザーが自分のリージョンを構成できるようにする](#)」を参照してください。また、ユーザーの場所や言語に固有のデフォルト設定を使用するようにフリートを設定することもできます。特に、次の Windows 設定をフリートに適用することができます。

- Time zone (タイムゾーン) — オペレーティングシステムの時刻に依存する Windows および任意のアプリケーションで使用されるシステム時間を決定します。AppStream 2.0 では、Windows Server 2012 R2、Windows Server 2016、および Windows Server 2019 と同じオプションを使用できます。
- Display Language (表示言語) — Windows オペレーティングシステムと特定の Windows アプリケーションで使用される表示言語を決定します。
- System Locale (システムロケール) — Windows が異なる言語で非 Unicode アプリケーションに使用するコードページ (ANSI、MS-DOS、および Macintosh) とビットマップフォントファイルを決定します。

- User Locale (ユーザーロケール) (またはカルチャー) — 日付、数値、通貨のフォーマットや文字列のソート時に Windows や任意のアプリケーションが Windows カルチャに照会して使用する規則を決定します。
- Input Method (入力方法) — 別の言語で文字を入力するときに使用できるキーストロークの組み合わせを決定します。

現在、AppStream 2.0 はこれらの言語設定でのみ英語と日本語をサポートしています。

内容

- [デフォルトのタイムゾーンを指定する](#)
- [デフォルトの表示言語を指定する](#)
- [デフォルトのシステムロケールを指定する](#)
- [デフォルトのユーザーロケールを指定する](#)
- [デフォルトの入力メソッドを指定する](#)
- [アプリケーション設定の永続化に関する特別な考慮事項](#)
- [日本語の言語設定に関する特別な考慮事項](#)

デフォルトのタイムゾーンを指定する

ユーザーのストリーミングセッションで使用するデフォルトのタイムゾーンを指定するには、次の 2 つの手順のうち、いずれかのステップを実行します。

手順

- [デフォルトのタイムゾーンを指定します \(Windows Server 2012 R2\)](#)
- [デフォルトのタイムゾーンを指定します \(Windows サーバー 2016、Windows サーバー 2019 および Windows サーバー 2022\)](#)

Note

現在、AppStream 2.0 は [UTC] と [(UTC+9:00) 大阪、札幌、東京] のみでサポートされています。

デフォルトのタイムゾーンを指定します (Windows Server 2012 R2)

1. 使用する Image Builder に接続し、ローカル管理者権限を持つユーザーでサインインします。そのためには、次のいずれかを実行します。

- [AppStream 2.0 コンソールを使用する](#) (ウェブ接続のみ)
- [ストリーミング URL を作成する](#) (ウェブまたは AppStream 2.0 クライアント接続用)

Note

接続先の Image Builder が Active Directory ドメインに参加していて、組織でスマートカードサインインが必要な場合は、ストリーミング URL を作成し、接続に AppStream 2.0 クライアントを使用する必要があります。スマートカードサインインの詳細については、「[スマートカード](#)」をご参照ください。

2. Image Builder のデスクトップで、Windows の [スタート] ボタンをクリックしてから、[コントロールパネル] を選択します。
3. [時計、言語、およびリージョン]、[日付と時刻]、[タイムゾーンの変更] を選択します。
4. [タイムゾーン] リストで、タイムゾーンを選択し、[OK] を選択します。
5. タイムゾーン設定の変更を適用するには、Image Builder を再起動します。それには、Windows の [スタート] ボタンを選択し、[Windows PowerShell] を選択します。PowerShell では、`restart-computer` コマンドレットを使用します。
6. Windows の再起動中に、AppStream 2.0 ログインプロンプトが表示されます。10 分待ってから、再度 Image Builder にログインします。それ以外の場合は、エラーが発生する場合があります。10 分後に [管理者] としてログインすることができます。
7. 必要に応じて、追加のデフォルトのリージョン設定または言語設定を構成します。それ以外の場合は、Image Builder デスクトップで Image Assistant を開き、ストリーミング用にアプリケーションをインストールして設定します。
8. Image Builder の設定が終了した後、Image Assistant で必要なステップに従って、イメージの作成を完了します。イメージを作成する方法については、「[チュートリアル: AppStream 2.0 コンソールを使用してカスタム AppStream 2.0 イメージを作成する](#)」を参照してください。
9. 次のいずれかを行います：
 - 新しいフリートを作成し、フリートの新しいイメージを選択します。詳細については、「[Amazon AppStream 2.0 フリートとスタックを作成する](#)」を参照してください。
 - 新しいイメージを使用するように既存のフリートを更新します。

10. デフォルトの設定を行うユーザーに割り当てられたスタックにフリートを関連付けます。

設定したデフォルトのタイムゾーン設定は、フリートインスタンスと、そのインスタンスから起動されるユーザーストリーミングセッションに適用されます。

デフォルトのタイムゾーンを指定します (Windows サーバー 2016、Windows サーバー 2019 および Windows サーバー 2022)

1. 使用する Image Builder に接続し、ローカル管理者権限を持つアカウントでサインインします。そのためには、次のいずれかを実行します。

- [AppStream 2.0 コンソールを使用する](#) (ウェブ接続のみ)
- [ストリーミング URL を作成する](#) (ウェブまたは AppStream 2.0 クライアント接続用)

Note

接続先の Image Builder が Active Directory ドメインに参加していて、組織でスマートカードサインインが必要な場合は、ストリーミング URL を作成し、接続に AppStream 2.0 クライアントを使用する必要があります。スマートカードサインインの詳細については、「[スマートカード](#)」をご参照ください。

2. Image Builder のデスクトップで、Windows の [スタート] ボタンをクリックしてから、[コントロールパネル] を選択します。
3. PowerShell または Windows ユーザーインターフェイスを使用して、デフォルトのタイムゾーンを指定します。

- PowerShell
 - PowerShell を開き、次のコマンドを実行します。

```
Run Set-TimeZone -Id "Tokyo Standard Time"
```

Note

このコマンドを実行するには、該当するコンピュータに管理者としてログインする必要があります。

- Windows ユーザーインターフェイス

1. Image Builder のデスクトップで、Windows の [Start (開始)] ボタンを選択し、「**timedate.cpl**」と入力して [Date and Time (日付と時刻)] のコントロールパネル項目を開きます。
2. [日付と時刻] アイコンを右クリックし、[管理者として実行] を選択します。
3. [ユーザーアカウント制御] によって、アプリケーションがデバイスに変更を加えることを許可するかどうかを選択するメッセージが表示されたら、[はい] を選択します。
4. [Change time zone (タイムゾーンの変更)] を選択します。
5. [タイムゾーン] リストで、タイムゾーンを選択し、[OK] を選択します。
4. 必要に応じて、追加のデフォルトのリージョン設定または言語設定を構成します。それ以外の場合は、Image Builder デスクトップで Image Assistant を開き、ストリーミング用にアプリケーションをインストールして設定します。
5. Image Builder の設定が終了した後、Image Assistant で必要なステップに従って、イメージの作成を完了します。イメージを作成する方法については、「[チュートリアル: AppStream 2.0 コンソールを使用してカスタム AppStream 2.0 イメージを作成する](#)」を参照してください。
6. 次のいずれかを行います：
 - 新しいフリートを作成し、フリートの新しいイメージを選択します。詳細については、「[Amazon AppStream 2.0 フリートとスタックを作成する](#)」を参照してください。
 - 新しいイメージを使用するように既存のフリートを更新します。
7. デフォルトの設定を行うユーザーに割り当てられたスタックにフリートを関連付けます。

設定したデフォルトのタイムゾーン設定は、フリートインスタンスと、そのインスタンスから起動されるユーザーストリーミングセッションに適用されます。

Note

ユーザーは、設定したデフォルト設定からタイムゾーンを変更できます。これらのアドレスは、アプリケーションストリーミングセッション中に、リージョンの設定を構成します。手順については、「[AppStream 2.0 ユーザーが自分のリージョンを構成できるようにする](#)」を参照してください。また、同じ AWS リージョン内のフリートインスタンスからストリーミングするときにユーザーが以前にタイムゾーンを選択した場合、ユーザーが指定したタイムゾーン設定は、Image Builder で指定したデフォルトのタイムゾーン設定を自動的に上書きします。

デフォルトの表示言語を指定する

ユーザーのストリーミングセッションにデフォルトの表示言語を指定するには、2通りの方法があります。AppStream 2.0 のデフォルトアプリケーションおよび Windows 設定機能を使用するか、ローカル管理者権限を持つアカウントでサインインしているときに Image Builder を構成します。このセクションの手順では、AppStream 2.0 のデフォルトアプリケーションおよび Windows 設定機能を使用して、デフォルトの表示言語を指定する方法について説明します。

Note

Windows で表示言語を変更すると、表示言語の言語とリージョンに合わせてユーザーロケールと入力方法が自動的に変更されます。3つの設定をすべて一致させるために、ユーザーロケールまたは入力方法を個別に変更する必要はありません。

1. 使用する Image Builder に接続し、[Template User (テンプレートユーザー)] アカウントでサインインします。そのためには、次のいずれかを実行します。
 - [AppStream 2.0 コンソールを使用する](#) (ウェブ接続のみ)
 - [ストリーミング URL を作成する](#) (ウェブまたは AppStream 2.0 クライアント接続用)

Note

接続先の Image Builder が Active Directory ドメインに参加していて、組織でスマートカードサインインが必要な場合は、ストリーミング URL を作成し、接続に AppStream 2.0 クライアントを使用する必要があります。スマートカードサインインの詳細については、「[スマートカード](#)」をご参照ください。

[Template User (テンプレートユーザー)] では、ユーザー向けのデフォルトアプリケーションと Windows の設定を作成できます。詳細については、[Amazon AppStream 2.0 でのデフォルトのアプリケーションおよび Windows 設定とアプリケーション起動パフォーマンス](#)の「Creating Default Application and Windows Settings for Your AppStream 2.0 Users」を参照してください。

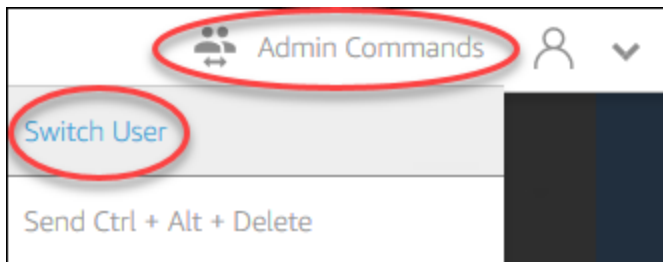
2. Image Builder のデスクトップで、Windows の [スタート] ボタンをクリックしてから、[コントロールパネル] を選択します。
3. [時計、言語、およびリージョン]、[言語]、[言語を追加] を選択します。

4. 言語を選択して、[追加] を選択します。

Note

現時点では、AppStream 2.0 は [英語 (米国)] と [日本語] のみをサポートしています。

5. 選択した言語が、Windows に追加した言語のリストに表示されます。先ほど追加した言語を選択します。次にその言語が言語リストの一番上に表示されるまで、[上へ移動] を選択します。
6. [詳細設定] を選択します。[Override for Windows display language (Windows の表示言語を上書きする)] で、使用する言語をリストから選択します。
7. 追加した言語に関連付けられている入力メソッドを使用する場合は、[Override for default input method (デフォルトの入力メソッドを上書き)] で、その言語の入力メソッドを選択します。
8. [Save] を選択します。ログオフするように求められたら、[Log off now (今すぐログオフ)] を選択します。
9. プロンプトが表示されたら、Image Builder に再度ログインし、[テンプレートのユーザー] を選択します。Windows で選択した表示言語が使用されていることを確認してください。
10. Image Builder デスクトップの右上領域で、[Admin Commands]、[Switch User] の順に選択します。



11. プロンプトが表示されたら、[管理者] としてログインします。
12. 必要に応じて、追加のデフォルトのリージョン設定または言語設定を構成します。それ以外の場合は、Image Builder デスクトップで Image Assistant を開き、ストリーミング用にアプリケーションをインストールして設定します。
13. Image Assistant プロセスのステップ 2 で、[設定の保存] を選択します。
14. Image Assistant で、必要な手順に従って、イメージの作成を完了します。イメージを作成する方法については、「[チュートリアル: AppStream 2.0 コンソールを使用してカスタム AppStream 2.0 イメージを作成する](#)」を参照してください。
15. 次のいずれかを行います：

- 新しいフリートを作成し、フリートの新しいイメージを選択します。詳細については、[Amazon AppStream 2.0 フリートとスタックを作成する](#) を参照してください。
- 新しいイメージを使用するように既存のフリートを更新します。

16. デフォルトの設定を行うユーザーに割り当てられたスタックにフリートを関連付けます。

設定したデフォルトの表示言語、関連するユーザーロケール、および入力メソッドの設定は、フリートインスタンスおよびそれらのインスタンスから起動されるユーザーストリーミングセッションに適用されます。

あるいは、[管理者] として Image Builder にログインしているときにデフォルトの表示言語を設定することもできます。[Template User] (テンプレートユーザー) アカウントと [[Administrator] (管理者) アカウントでログインしているときに別の表示言語を選択し、Image Assistant プロセスのステップ 2 で [Save settings] (設定の保存) を選択した場合は、[Template User] (テンプレートユーザー) の設定が優先されます。

Note

ユーザーは、自分が設定したデフォルト設定からユーザーロケールと入力メソッドを変更できます。サポートされている 11 種類のロケールと 9 種類のサポートされている入力メソッドのいずれかに変更できます。そうするには、「[AppStream 2.0 ユーザーが自分のリージョンを構成できるようにする](#)」にあるように、アプリケーションストリーミングセッションの間にリージョン設定を構成することができます。また、同じリージョン内のフリートインスタンスからストリーミングするときに、ユーザーが以前にユーザーロケールまたは入力メソッドを選択した場合、それらのユーザー指定の設定は、Image Builder で指定したデフォルトのユーザーロケールおよび入力メソッドを自動的に上書きします。

デフォルトのシステムロケールを指定する

ユーザーのストリーミングセッションのデフォルトのシステムロケールを指定するには、次のステップを実行します。

1. 使用する Image Builder に接続し、ローカル管理者権限を持つアカウントでサインインします。そのためには、次のいずれかを実行します。
 - [AppStream 2.0 コンソールを使用する](#) (ウェブ接続のみ)

- [ストリーミング URL を作成する](#) (ウェブまたは AppStream 2.0 クライアント接続用)

 Note

接続先の Image Builder が Active Directory ドメインに参加していて、組織でスマートカードサインインが必要な場合は、ストリーミング URL を作成し、接続に AppStream 2.0 クライアントを使用する必要があります。スマートカードサインインの詳細については、「[スマートカード](#)」をご参照ください。

2. Image Builder のデスクトップで、Windows の [スタート] ボタンをクリックしてから、[コントロールパネル] を選択します。
3. [時計、言語、およびリージョン] を選択し、[リージョン] を選択します。
4. [リージョン] ダイアログボックスの [形式] タブを選択します。
5. [システムロケールの変更] を選択します。
6. [リージョン設定] ダイアログボックスの [現在のシステムロケール] リストで、言語と地域を選択します。

 Note

現時点では、AppStream 2.0 は [英語 (米国)] と [日本語 (日本)] のみをサポートしています。

7. [OK] を選択して [Region Settings] (リージョン設定) ダイアログボックスを閉じ、再度 [OK] を選択して [Region] (リージョン) ダイアログボックスを閉じます。
8. コンピュータを再起動するように求められたら、Windows を再起動します。
9. Windows の再起動中に、AppStream 2.0 ログインプロンプトが表示されます。10 分待つから、再度 Image Builder にログインします。それ以外の場合は、エラーが発生する場合があります。10 分後に [管理者] としてログインすることができます。
10. 必要に応じて、追加のデフォルトのリージョン設定または言語設定を構成します。それ以外の場合は、Image Builder デスクトップで Image Assistant を開き、ストリーミング用にアプリケーションをインストールして設定します。Image Builder の設定が終了した後、Image Assistant で必要なステップに従って、イメージの作成を完了します。イメージを作成する方法については、「[チュートリアル: AppStream 2.0 コンソールを使用してカスタム AppStream 2.0 イメージを作成する](#)」を参照してください。
11. 次のいずれかを行います：

- 新しいフリートを作成し、フリートの新しいイメージを選択します。詳細については、「[Amazon AppStream 2.0 フリートとスタックを作成する](#)」を参照してください。
- 新しいイメージを使用するように既存のフリートを更新します。

12. デフォルトの設定を行うユーザーに割り当てられたスタックにフリートを関連付けます。

設定したデフォルトのシステムロケール設定は、フリートインスタンスと、そのインスタンスから起動されるユーザーストリーミングセッションに適用されます。

デフォルトのユーザーロケールを指定する

ユーザーのストリーミングセッションのデフォルトのユーザーロケールを指定するには、次のステップを実行します。

Note

表示言語を設定する予定で、ユーザーロケールと表示言語を一致させたい場合は、ユーザーロケールを変更する必要はありません。表示言語を変更すると、それに合わせてユーザーロケールも自動的に変更されます。

1. 使用する Image Builder に接続し、ローカル管理者権限を持つアカウントでサインインします。そのためには、次のいずれかを実行します。
 - [AppStream 2.0 コンソールを使用する](#) (ウェブ接続のみ)
 - [ストリーミング URL を作成する](#) (ウェブまたは AppStream 2.0 クライアント接続用)

Note

接続先の Image Builder が Active Directory ドメインに参加していて、組織でスマートカードサインインが必要な場合は、ストリーミング URL を作成し、接続に AppStream 2.0 クライアントを使用する必要があります。スマートカードサインインの詳細については、「[スマートカード](#)」をご参照ください。

2. Image Builder のデスクトップで、Windows の [スタート] ボタンをクリックしてから、[コントロールパネル] を選択します。
3. [時計、言語、およびリージョン] を選択し、[リージョン] を選択します。

4. [リージョン] ダイアログボックスの [形式] タブを選択します。
5. [形式] リストで、言語とリージョンを選択します。

 Note

現時点では、AppStream 2.0 は [英語 (米国)] と [日本語 (日本)] のみをサポートしています。

6. [OK] を選択し、[リージョン] ダイアログボックスを閉じます。
7. 必要に応じて、追加のデフォルトのリージョン設定または言語設定を構成します。それ以外の場合は、Image Builder デスクトップで Image Assistant を開き、ストリーミング用にアプリケーションをインストールして設定します。
8. Image Assistant プロセスのステップ 2 で、[設定の保存] を選択します。
9. Image Assistant で、必要な手順に従って、イメージの作成を完了します。イメージを作成する方法については、「[チュートリアル: AppStream 2.0 コンソールを使用してカスタム AppStream 2.0 イメージを作成する](#)」を参照してください。
10. 次のいずれかを行います：
 - 新しいフリートを作成し、フリートの新しいイメージを選択します。詳細については、「[Amazon AppStream 2.0 フリートとスタックを作成する](#)」を参照してください。
 - 新しいイメージを使用するように既存のフリートを更新します。
11. デフォルトの設定を行うユーザーに割り当てられたスタックにフリートを関連付けます。

設定したデフォルトのユーザーロケール設定は、フリートインスタンスと、そのインスタンスから起動されるユーザーストリーミングセッションに適用されます。

 Note

ユーザーは、ユーザーロケールを構成したデフォルト設定から、サポートされている 11 の異なるロケールのいずれかに変更できます。そうするには、「[AppStream 2.0 ユーザーが自分のリージョンを構成できるようにする](#)」にあるように、アプリケーションストリーミングセッションの間にリージョン設定を構成することができます。また、同じリージョン内のフリートインスタンスからストリーミングするときに、ユーザーが以前にユーザーロケールまたは入力メソッドを選択した場合、それらのユーザー指定の設定は、Image Builder で指定したデフォルトのユーザーロケール設定を自動的に上書きします。

デフォルトの入力メソッドを指定する

ユーザーのストリーミングセッションで使用するデフォルトの入力メソッドを指定するには、次のステップを実行します。

Note

表示言語を設定する予定で、入力メソッドと表示言語を一致させたい場合は、入力メソッドを変更する必要はありません。Windows で表示言語を変更すると、表示言語の言語とリージョンに合わせてユーザーロケールと入力方法が自動的に変更されます。3 つの設定をすべて一致させるために、ユーザーロケールまたは入力方法を個別に変更する必要はありません。

1. 使用する Image Builder に接続し、ローカル管理者権限を持つアカウントでサインインします。そのためには、次のいずれかを実行します。
 - [AppStream 2.0 コンソールを使用する](#) (ウェブ接続のみ)
 - [ストリーミング URL を作成する](#) (ウェブまたは AppStream 2.0 クライアント接続用)

Note

接続先の Image Builder が Active Directory ドメインに参加していて、組織でスマートカードサインインが必要な場合は、ストリーミング URL を作成し、接続に AppStream 2.0 クライアントを使用する必要があります。スマートカードサインインの詳細については、「[スマートカード](#)」をご参照ください。

2. Image Builder のデスクトップで、Windows の [スタート] ボタンをクリックしてから、[コントロールパネル] を選択します。
3. [時計、言語、およびリージョン]、[言語]、[言語を追加] を選択します。
4. 言語を選択して、[追加] を選択します。

Note

現時点では、AppStream 2.0 は [英語 (米国)] と [日本語] のみをサポートしています。

5. 選択した言語が、Windows に追加した言語のリストに表示されます。

6. [詳細設定] を選択します。[Override for default input method (デフォルトの入力メソッドを上書き)] で、追加した言語の入力メソッドを選択します。
7. [Save] を選択します。
8. ログオフして、もう一度ログインします。これを行うには、Image Builder デスクトップで Windows の [スタート] ボタンを選択します。[ImageBuilderAdmin] を選択し、[サインアウト] を選択します。プロンプトが表示されたら、[管理者] としてログインします。
9. 必要に応じて、追加のデフォルトのリージョン設定または言語設定を構成します。それ以外の場合は、Image Builder デスクトップで Image Assistant を開き、ストリーミング用にアプリケーションをインストールして設定します。
10. Image Assistant プロセスのステップ 2 で、[設定の保存] を選択します。
11. Image Assistant で、必要な手順に従って、イメージの作成を完了します。イメージを作成する方法については、「[チュートリアル: AppStream 2.0 コンソールを使用してカスタム AppStream 2.0 イメージを作成する](#)」を参照してください。
12. 次のいずれかを行います：
 - 新しいフリートを作成し、フリートの新しいイメージを選択します。詳細については、[Amazon AppStream 2.0 フリートとスタックを作成する](#) を参照してください。
 - 新しいイメージを使用するように既存のフリートを更新します。
13. デフォルトの設定を行うユーザーに割り当てられたスタックにフリートを関連付けます。

設定したデフォルトの入力メソッドは、フリートインスタンスと、そのインスタンスから起動されるユーザーストリーミングセッションに適用されます。

Note

ユーザーは、入力メソッドを構成したデフォルト設定から、サポートされている 9 の異なる入力メソッドのいずれかに変更できます。「[AppStream 2.0 ユーザーが自分のリージョンを構成できるようにする](#)」で説明されているように、アプリケーションストリーミングセッション中にリージョン設定を構成することで、この設定を構成できます。また、同じリージョン内のフリートインスタンスからストリーミングするときに、ユーザーが以前に入力メソッドを選択した場合、それらのユーザー指定の設定は、Image Builder で指定したデフォルトの入力メソッドを自動的に上書きします。

アプリケーション設定の永続化に関する特別な考慮事項

Step 3: User Settings で AppStream 2.0 コンソールでスタックを作成するときは、異なるリージョン設定を使用する別のスタックとして [Application settings persistence (アプリケーション設定の永続化)] の下にある同じ設定グループを使用する場合、両方のスタックに使用されるリージョン設定は 1 セットだけです。各ユーザーについて、ユーザーが最初にログインしたスタックのデフォルトのリージョン設定が、同じアプリケーション設定グループ内の他のスタックのデフォルトのリージョン設定を自動的に上書きします。この問題を回避するには、リージョン設定が異なる 2 つの異なるスタックに同じアプリケーション設定グループを使用しないでください。

日本語の言語設定に関する特別な考慮事項

このセクションでは、AppStream 2.0 ユーザー用に日本語を設定する際に注意すべき重要事項について説明します。

AWS CLI

Windows システムロケールを日本語に変更するには、Image Builder AWS に コマンドラインインターフェイス (AWS CLI) バージョン 1.16.30 以降がインストールされている必要があります。Image Builder で AWS CLI のバージョンを更新するには、[AWS 「コマンドラインインターフェイスのインストール」](#) の手順に従います。

日本語キーボード

イメージを作成するときに Image Builder の入力メソッドが日本語に設定されていると、AppStream 2.0 は自動的に日本語キーボードを使用するようにイメージを構成します。また、イメージを使用するフリートは日本語キーボードを使用するように自動的に構成されます。ただし、Image Builder セッションで日本語キーボードを使用する場合は、HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\i8042prt\Parameters registry key レジストリキーの次のレジストリ設定を更新してください。

名前	型	[データ]
LayerDriver JPN	REG_SZ	kbd106.dll
OverrideKeyboardIdentifier	REG_SZ	PCAT_106KEY
OverrideKeyboardSubtype	DWORD	2

名前	型	[データ]
OverrideKeyboardType	DWORD	7

これらの設定を変更した後、Image Builder を再起動します。それには、Windows の [スタート] ボタンを選択し、[Windows PowerShell] を選択します。PowerShell では、restart-computer コマンドレットを使用します。

AppStream 2.0 ユーザーが自分のリージョンを構成できるようにする

Note

現在、ユーザーがリージョン設定の指定を有効にすることは Linux ベースのストリーミングセッションではサポートされていません。

ユーザーは、自分の所在地や言語に固有の設定を使用するように Amazon AppStream 2.0 Windows ストリーミングセッションを設定できます。特に、ユーザーは以下の項目を設定できます。

- Time zone (タイムゾーン) — オペレーティングシステムの時刻に依存する Windows および任意のアプリケーションで使用されるシステム時間を決定します。AppStream 2.0 では、フリートで使用されている Windows Server のバージョンと同じオプションをこの設定で使用できます。
- ストリーミングセッションのタイムゾーンをデバイスに設定されているタイムゾーンと同期するには、[デバイスに基づいて自動的にタイムゾーンを設定] を選択します。これは、シングルセッションフリートとマルチセッションフリートの両方でデフォルトで有効になっており、無効にすることができます。
- シングルセッションフリートの場合、自動リダイレクトを使用する代わりに、ストリーミングセッションの特定のタイムゾーンを選択できます。カスタムタイムゾーンを設定するには、[リージョン設定] の [デバイスオプションに基づいてタイムゾーンを自動的に設定] を無効にし、使用可能なリストから任意のタイムゾーンを選択します。
- Locale (国) (カルチャ) — 日付、数値、通貨のフォーマットや文字列のソート時に Windows や任意のアプリケーションが Windows カルチャに照会して使用する規則を決定します。AppStream 2.0 がサポートする国一覧については、[サポート対象ロケール](#) を参照してください。

- Input method (入力方法) — 別の言語で文字を入力するときに使用できるキーストロークの組み合わせを決定します。

ユーザーがストリーミングセッション中にその地域の設定を変更する場合、同じ AWS リージョンで行われる今後のストリーミングセッションにもその変更が適用されます。

 Note

リージョンの設定を開始するのに役立つようにユーザーに提供できるガイダンスについては、[地域の設定を指定する](#) を参照してください。

内容

- [サポート対象ロケール](#)
- [AppStream 2.0 ユーザー用のリージョン設定を有効にする](#)

サポート対象ロケール

AppStream 2.0 では、以下のロケールがサポートされています。

[Locale] (国)	言語カルチャ名
中国語 (簡体字、中国)	zh-CN
中国語 (簡体字、シンガポール)	zh-SG
繁体字中国語	zh-TW
オランダ語 (オランダ)	nl-NL
英語 (オーストラリア)	en-AU
英語 (カナダ)	en-CA
英語 (英国)	en-GB
英語 (米国)	en-US

[Locale] (国)	言語カルチャ名
フランス語 (フランス)	fr-FR
ドイツ語 (ドイツ)	de-DE
イタリア語 (イタリア)	it-IT
日本語 (日本)	ja-JP
韓国語 (韓国)	ko-KR
ポルトガル語 (ブラジル)	pt-BR
スペイン語 (スペイン、インターナショナルソート)	es-ES
タイ語 (タイ)	th-TH

AppStream 2.0 ユーザー用のリージョン設定を有効にする

AppStream 2.0 ストリーミングセッション中に、該当スタックのリージョン設定をユーザーが指定可能にするには、2018 年 6 月 6 日以降にリリースされた AppStream 2.0 エージェントのバージョンを使用するイメージに基づくフリートに、スタックが関連付けられている必要があります。詳細については、「[AppStream 2.0 エージェントのリリースノート](#)」を参照してください。また、イメージに Windows PowerShell 5.1 以降がインストールされている必要があります。2018 年 6 月 12 日以降に公開された、AppStream 2.0 ベースイメージから作成されたイメージは、両方の条件を満たします。2018 年 6 月 12 日より前に公開された、AppStream 2.0 ベースイメージから作成されたイメージには、デフォルトでは Windows PowerShell 5.1 が含まれません。

Windows PowerShell 5.1 が含まれるように、既存のイメージを更新するには

1. ベースイメージとして既存のイメージを使用して、新しい Image Builder を起動し、次の手順を実行します。
 - a. AppStream 2.0 コンソールの左側のナビゲーションペインで、[Images (イメージ)] を選択します。
 - b. [Image Builder] タブ、[Launch Image Builder] の順に選択し、既存のイメージを選択します。

- c. Image Builder を起動するときに、AppStream 2.0 エージェントを更新するかどうかを確認するメッセージが表示された場合は、チェックボックスをオンにして [Start (開始)] を選択します。
2. Image Builder の実行後に接続して、ローカル管理者権限を持つアカウントでサインインします。そのためには、次のいずれかを実行します。
 - [AppStream 2.0 コンソールを使用する](#) (ウェブ接続のみ)
 - [ストリーミング URL を作成する](#) (ウェブまたは AppStream 2.0 クライアント接続用)

 Note

接続先の Image Builder が Active Directory ドメインに参加していて、組織でスマートカードサインインが必要な場合は、ストリーミング URL を作成し、接続に AppStream 2.0 クライアントを使用する必要があります。スマートカードサインインの詳細については、「[スマートカード](#)」をご参照ください。

3. Image Builder デスクトップから、Windows PowerShell を開きます。Windows の [スタート] ボタンをクリックし、[Windows PowerShell] を選択します。
4. PowerShell コマンドプロンプトで \$PSVersionTable コマンドを入力し、Image Builder にインストールされている Windows PowerShell のバージョンを確認します。Image Builder に Windows PowerShell 5.1 以降が含まれていない場合は、次の手順でインストールします。
5. ウェブブラウザを開き、Microsoft のドキュメントの [Install and Configure WMF 5.1](#) の手順に従います。必ず Windows Server 2012 R2 用 Windows Management Framework (WMF) 5.1 パッケージをダウンロードします。WMF 5.1 には Windows PowerShell 5.1 が含まれます。
6. WMF 5.1 のインストールプロセスが終了すると、コンピュータを再起動するかどうかを確認するメッセージが表示されます。Image Builder を再起動する場合は、[今すぐ再起動] を選択します。
7. 約 10 分間待機してから、Image Builder にログインします。これは、AppStream 2.0 ですぐにプロンプトが表示された場合でも同じです。そうでない場合は、エラーが発生する可能性があります。
8. Image Builder に再ログインした後、Windows PowerShell を開き、\$PSVersionTable コマンドを入力して、Image Builder に Windows PowerShell 5.1 がインストールされていることを確認します。

9. Image Builder を使用して新しいイメージを作成します。これでこの新しいイメージに、最新バージョンの AppStream 2.0 エージェントと Windows PowerShell が含まれるようになります。
10. 次の手順を実行して、新しいイメージを使用するようにフリートを更新します。
 - a. AppStream 2.0 コンソールの左のナビゲーションペインで、[Fleets (フリート)] を選択し、リージョン設定を有効にする、スタックに関連付けられているフリートを選択します。
 - b. [Fleet Details] タブで、[Edit] を選択します。
 - c. [Image name] で、フリートに使用する新しいイメージを選択します。

Image Builder を使用してイメージを作成する方法の詳細については、「[チュートリアル: AppStream 2.0 コンソールを使用してカスタム AppStream 2.0 イメージを作成する](#)」を参照してください。

アプリケーションの使用権限の管理

Amazon AppStream 2.0 では、アプリケーションカタログを動的に構築して、ユーザーがアクセスできる AppStream 2.0 アプリケーションを表示できます。アプリケーションの使用権限は、サードパーティーの SAML 2.0 アイデンティティプロバイダーを使用するか、AppStream 2.0 動的アプリケーションフレームワークを使用して、属性に基づいて割り当てることができます。以下のセクションでは、アプリケーションの使用権限を管理する方法について説明します。

Note

ほとんどのシナリオで、サードパーティーの SAML 2.0 アイデンティティプロバイダーを使用する属性ベースのアプリケーションの使用権限が推奨されます。アプリケーションを AppStream 2.0 イメージにインストールする必要があるように、使用権限に加えてアプリケーションパッケージの配信を管理する既存の動的アプリケーションプロバイダーを使用する場合は、動的アプリケーションフレームワークをお勧めします。詳細については、「[動的アプリケーションプロバイダーと動的アプリケーションフレームワークについて学ぶための追加リソース](#)」を参照してください。

内容

- [サードパーティーの SAML 2.0 アイデンティティプロバイダーを使用した属性ベースのアプリケーションの使用権限](#)
- [動的アプリケーションフレームワークを使用した動的アプリケーションプロバイダーからのアプリケーションの使用権限の適用](#)

サードパーティーの SAML 2.0 アイデンティティプロバイダーを使用した属性ベースのアプリケーションの使用権限

アプリケーション使用権限は、AppStream 2.0 スタック内の特定のアプリケーションへのアクセスを制御します。これは、サードパーティーの SAML 2.0 アイデンティティプロバイダーからの SAML 2.0 属性アサーションを使用して動作します。アサーションは、ユーザーアイデンティティが AppStream 2.0 SAML アプリケーションにフェデレートされるときに一致します。使用権限が true で、属性の名前と値が一致する場合、スタック内の 1 つ以上のアプリケーションに対するユーザーアイデンティティへのアクセス権が付与されます。

以下のシナリオでは、サードパーティーの SAML 2.0 アイデンティティプロバイダーを使用する属性ベースのアプリケーションの使用権限は適用されません。つまり、次のような場合、使用権限は無視されます。

- AppStream 2.0 ユーザープール認証。詳細については、「[Amazon AppStream 2.0 ユーザープール](#)」を参照してください。
- AppStream 2.0 ストリーミング URL 認証。詳細については、「[ストリーミング URL](#)」を参照してください。
- AppStream 2.0 フリートがデスクトップストリームビュー用に設定されている場合のデスクトップアプリケーション。詳細については、「[Amazon AppStream 2.0 フリートとスタックを作成する](#)」を参照してください。
- 動的アプリケーションフレームワークを使用するスタック。動的アプリケーションフレームワークは、個別のアプリケーションの使用権限機能を提供します。詳細については、「[動的アプリケーションフレームワークを使用した動的アプリケーションプロバイダーからのアプリケーションの使用権限の適用](#)」を参照してください。
- ユーザーが AppStream 2.0 アプリケーションカタログにフェデレートすると、アプリケーションの使用権限には、ユーザーが資格が付与されているアプリケーションのみが表示されます。AppStream 2.0 セッション内でのアプリケーションの実行に制限はありません。例えば、デスクトップストリームビュー用に設定されたフリートでは、ユーザーはデスクトップから直接アプリケーションを起動できます。

アプリケーションの使用権限の作成

アプリケーションの使用権限を作成する前に、以下を実行する必要があります。

- AppStream 2.0 フリートを作成し、ニーズを満たす 1 つ以上のアプリケーション（常時オンまたはオンデマンドフリート）または割り当てられたアプリケーション（Elastic フリート）を含むイメージでスタックします。詳細については、「[Amazon AppStream 2.0 フリートとスタックを作成する](#)」を参照してください。
- サードパーティーの SAML 2.0 アイデンティティプロバイダーを使用して、スタックへのユーザーアクセスを提供します。詳細については、「[Amazon AppStream 2.0 と SAML 2.0 の統合](#)」を参照してください。以前にセットアップした既存の SAML 2.0 アイデンティティプロバイダーを使用している場合は、「[ステップ 2: SAML 2.0 フェデレーション IAM ロールを作成する](#)」の IAM ロールの信頼ポリシーに sts:TagSession 許可を追加する手順を参照してください。詳細については、「[AWS STSでのセッションタグの受け渡し](#)」を参照してください。この許可は、アプリケーションの使用権限を使用するために必要です。

アプリケーション使用権限を作成するには

1. [AppStream 2.0 コンソールを開きます](#)。
2. 左のナビゲーションペインで、[Stacks] (スタック) を選択し、アプリケーションの使用権限を管理するスタックを選択します。
3. [Application Entitlements] (アプリケーションの使用権限) ダイアログボックスで、[Create] (作成) を選択します。
4. 使用権限の[Name] (名前) と[Description] (説明) を入力します。
5. 使用権限の属性名と値を定義します。

属性をマッピングする場合は、`https://aws.amazon.com/SAML/Attributes/PrincipalTag:{TagKey}` の形式で属性を指定します。{TagKey} は次の属性のいずれかです。

- ロール
- department
- 組織
- グループ
- title
- costCenter
- userType

定義した属性は、アプリケーションが AppStream 2.0 セッションにフェデレートされるときに、スタック内のアプリケーションをユーザーに付与するために使用されます。使用権限は、フェデレーション中に作成された SAML アサーションのキーバリュー名と、属性名を一致させることによって機能します。詳細については、「[SAML PrincipalTag 属性](#)」を参照してください。

Note

1 つ以上の値をコロン (:) で区切って、サポートされている属性に含めることができます。

たとえば、グループ情報を SAML 属性名 `https://aws.amazon.com/SAML/Attributes/PrincipalTag:groups` に「group1: group2: group3」という値で渡すことができます。使用権限は、単一のグループ値、例えば「group1」に基づいてアプリケーションを許可できます。詳細については、「[SAML PrincipalTag 属性](#)」を参照してください。

6. スタック内のアプリケーション設定を構成して、すべてのアプリケーションに資格を付与するか、アプリケーションを選択します。[All applications (*)] (すべてのアプリケーション (*)) を選択すると、将来追加されるアプリケーションを含め、スタックで使用可能なすべてのアプリケーションを適用します。[Select applications] (アプリケーションの選択) を選択すると、特定のアプリケーション名をフィルタリングします。
7. 設定を確認して使用権限を作成します。このプロセスを繰り返して、追加の使用権限を作成できます。スタック内のアプリケーションに対する使用権限は、属性名と値に基づいてユーザーと一致するすべての使用権限の組み合わせとなります。
8. SAML 2.0 アイデンティティプロバイダーで、使用権限で定義された属性と値を送信するように、AppStream 2.0 SAML アプリケーション属性マッピングを設定します。ユーザーが AppStream 2.0 アプリケーションカタログにフェデレートすると、アプリケーションの使用権限には、ユーザーが資格が付与されているアプリケーションのみが表示されます。

SAML 2.0 マルチスタックアプリケーションカタログ

サードパーティー SAML 2.0 アイデンティティプロバイダーを使用する属性ベースのアプリケーションの使用権限を使用すると、単一のリレー状態 URL から複数のスタックへのアクセスを有効にできます。次のように、リレー状態 URL からスタックと アプリケーション (存在する場合) パラメータを削除します。

```
https://relay-state-region-endpoint?accountId=aws-account-id-without-hyphens
```

ユーザーが AppStream 2.0 アプリケーションカタログにフェデレートすると、スタックが配置されているリージョンに関連付けられたアカウント ID およびリレー状態エンドポイントについて、アプリケーションの使用権限が 1 つ以上のアプリケーションをユーザーに一致させたすべてのスタックが表示されます。ユーザーがカタログを選択すると、アプリケーションの使用権限には、そのユーザーが資格を持つアプリケーションのみが表示されます。詳細については、「[ステップ 6: フェデレーションのリレーステートを設定する](#)」を参照してください。

Note

SAML 2.0 マルチスタックアプリケーションカタログを使用するには、SAML 2.0 フェデレーション IAM ロールのインラインポリシーを設定する必要があります。詳細については、「[ステップ 3: IAM ロールにインラインポリシーを埋め込む](#)」を参照してください。

動的アプリケーションフレームワークを使用した動的アプリケーションプロバイダーからのアプリケーションの使用権限の適用

Note

現在、動的アプリケーションフレームワークでのアプリケーション使用権限の管理は Linux ベースのスタック向けにサポートされていません。

Amazon AppStream 2.0 は、ユーザーが AppStream 2.0 スタックからストリーミングするときにユーザーに表示されるアプリケーションカタログの動的構築をサポートします。AppStream 2.0 によって提供される API オペレーションを使用して、ストリーミングインスタンス上でユーザーがアクセスできるアプリケーションをリアルタイムで変更する動的アプリプロバイダーを開発できます。あるいは、これらの API オペレーションを使用するサードパーティーの動的アプリプロバイダーを実装することもできます。

Note

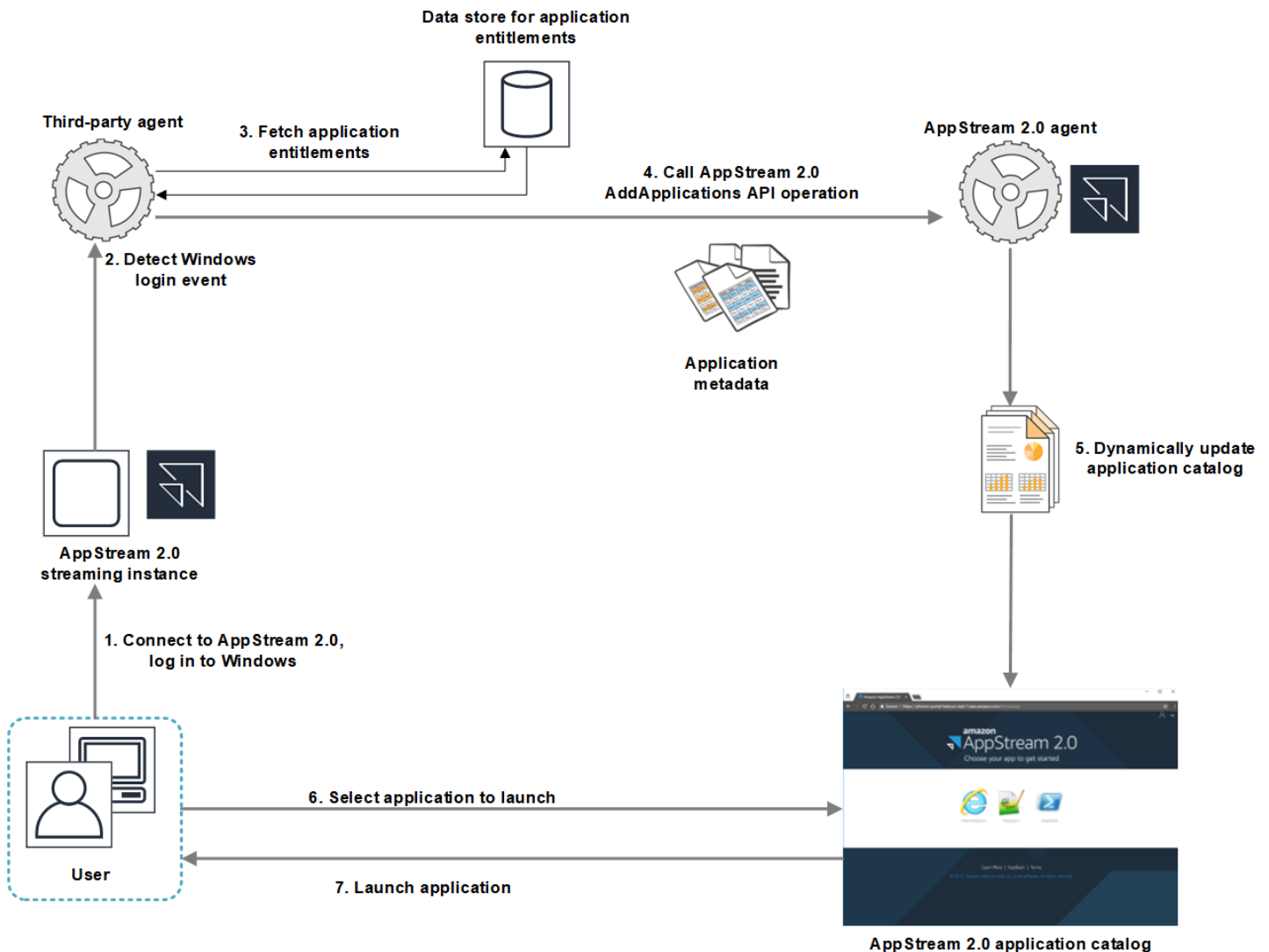
この機能には、Microsoft アクティブディレクトリドメインに参加済みの AppStream 2.0 常時オンまたはオンデマンドフリートが必要です。詳細については、「[AppStream 2.0 でのアクティブディレクトリの使用](#)」を参照してください。この機能は、マルチセッションフリートでは使用できません。

内容

- [動的アプリケーションフレームワークの API オペレーションのワークフロー例](#)
- [AppStream 2.0 動的アプリケーションフレームワークを使用して動的アプリケーションプロバイダーを構築する](#)
- [動的アプリケーションプロバイダーの有効化](#)
- [動的アプリケーションプロバイダーのテスト \(オプション\)](#)
- [動的アプリケーションプロバイダーと動的アプリケーションフレームワークについて学ぶための追加リソース](#)

動的アプリケーションフレームワークの API オペレーションのワークフロー例

次の図は、AppStream 2.0 とサードパーティーアプリケーションプロバイダー間の API オペレーションフローの例です。



1. ユーザーは AppStream 2.0 に接続します。フリートストリーミングインスタンスがユーザーに割り当てられ、Windows ログインが発生します。
2. サービスまたはエージェントが Windows のログオンイベントを検出し、Windows にログインしているユーザーを特定します。
3. サービスまたはエージェントは、ユーザーのアプリケーション使用権限を取得します。図の例では、アプリケーションの使用権限はデータベースに保存されます。この情報はさまざまな方法で保存し取得できます。たとえば、アプリケーションの使用期限をサーバーソフトウェアから取得

したり、Active Directory 内のグループ名を解析してアプリケーション識別子 (ID) を見つけることができます。

4. 動的アプリプロバイダーは、ユーザーが持つべきアプリケーションのアプリケーションメタデータを使用して AppStream 2.0 エージェント AddApplications API オペレーションを呼び出します。
5. AppStream 2.0 エージェントは、変更されたアプリケーションリストでアプリケーションカタログを動的に更新します。
6. ユーザーは、起動するアプリケーションを選択します。
7. アプリケーションは、サービスまたはエージェントによって指定されたアプリケーションメタデータを使用して起動されます。

ユーザーの立場では、この処理を意識することはありません ユーザーは AppStream 2.0 に接続し、フリーストインスタンスにログインします。ログイン後、画像に指定されて動的アプリケーションプロバイダーによって提供されたアプリケーションのリストがユーザーに表示されます。

AppStream 2.0 動的アプリケーションフレームワークを使用して動的アプリケーションプロバイダーを構築する

AppStream 2.0 動的アプリケーションフレームワークは、動的アプリケーションプロバイダーを構築するために使用できる、AppStream 2.0 ストリーミングインスタンス内の API オペレーションを提供します。動的アプリケーションプロバイダーは、提供されている API オペレーションを使用して、ユーザーがリアルタイムでアクセスできるアプリケーションのカタログを変更できます。動的アプリケーションプロバイダーによって管理されるアプリケーションは、イメージ内に存在することも、Windows ファイル共有やアプリケーション仮想化テクノロジーなどからインスタンス外に存在することもあります。

Note

この機能には、Microsoft アクティブディレクトリドメインに参加済みの AppStream 2.0 常時オンまたはオンデマンドフリーストが必要です。詳細については、「[AppStream 2.0 でのアクティブディレクトリの使用](#)」を参照してください。

内容

- [動的アプリケーションフレームワークについて](#)
- [動的アプリケーションフレームワーク Thrift 定義と名前付きパイプ名](#)

- [AppStream 2.0 のアプリケーション使用権限管理のための API アクション](#)

動的アプリケーションフレームワークについて

動的アプリケーションフレームワークでは、プロセス間メッセージで [Apache Thrift ソフトウェアフレームワーク](#) を使用します。Windows の名前付きパイプを介して公開されます。Thrift フレームワークを使用して、選択したソフトウェア開発言語で動的アプリケーションプロバイダーを構築できます。動的アプリケーションフレームワークは、3 つの API オペレーション AddApplications、RemoveApplications、ClearApplications で構成されます。

動的アプリケーションフレームワーク Thrift 定義と名前付きパイプ名

Thrift では、AppStream 2.0 が提供する単純な定義ファイルを使用して RPC クライアントをコンパイルできます。RPC クライアントを使用すると、ストリーミングインスタンスで実行されている AppStream 2.0 エージェントソフトウェアと通信できます。RPC クライアントを自分の言語用にコンパイルする方法については、[Apache Thrift ドキュメント](#) を参照してください。選択した言語用に Thrift ライブラリをコンパイルしたら、名前付きパイプトランスポートを使用して Thrift クライアントを構築します。D56C0258-2173-48D5-B0E6-1EC85AC67893 をパイプ名として使用します。

AppStreamServer.thrift

```
namespace netstd AppStream.ApplicationCatalogService.Model

const string ServiceEndpoint = "D56C0258-2173-48D5-B0E6-1EC85AC67893";

struct AddApplicationsRequest
{
    1: required string userSid;
    2: required list<Application> applications;
}

struct AddApplicationsResponse
{
}

struct RemoveApplicationsRequest
{
    1: required string userSid;
    2: required list<string> applicationIds;
}
```

```
struct RemoveApplicationsResponse
{
}

struct ClearApplicationsRequest
{
    1: required string userSid;
}

struct ClearApplicationsResponse
{
}

struct Application
{
    1: required string id;
    2: required string displayName;
    3: required string launchPath;
    4: required string iconData;
    5: string launchParams;
    6: string workingDirectory;
}

exception AppStreamClientException
{
    1: string errorMessage,
    2: ErrorCode errorCode
}

exception AppStreamServerException
{
    1: string errorMessage,
    2: ErrorCode errorCode
}

enum ErrorCode
{
}

service ApplicationCatalogService
{
    AddApplicationsResponse AddApplications(1:AddApplicationsRequest request)
    throws (1: AppStreamClientException ce, 2: AppStreamServerException se),
```



```
RemoveApplicationsResponse RemoveApplications(1:RemoveApplicationsRequest request)
throws (1: AppStreamClientException ce, 2: AppStreamServerException se),

ClearApplicationsResponse ClearApplications(1:ClearApplicationsRequest request)
throws (1: AppStreamClientException ce, 2: AppStreamServerException se),
}
```

AppStream 2.0 のアプリケーション使用権限管理のための API アクション

以下の API オペレーションを使用して、AppStream 2.0 のアプリケーション使用権限を管理できます。

AddApplicationsRequest オペレーション

AppStream 2.0 ユーザーのアプリケーションカタログにアプリケーションを追加します。AppStream 2.0 によって表示されるアプリケーションカタログには、この API オペレーションを使用して追加したアプリケーションと、イメージに追加したアプリケーションが含まれています。これらの方法の一方または両方を使用してアプリケーションを追加した後、ユーザーはアプリケーションを起動できます。

リクエストの構文

*userSid*;

list<Application> applications;

リクエストパラメータ

userSid

リクエストが適用されるユーザーの SID。

型: 文字列

必須: はい

長さの制限: 最小長は 1、最大長は 208 文字。

applications

リクエストが適用されるアプリケーションのリスト。

型: 文字列

必須: はい

Application オブジェクト

アプリケーションの表示と起動に必要なアプリケーションメタデータを記述します。アプリケーションの識別子は一意である必要があり、API オペレーションまたはイメージを介して指定された他のアプリケーションと競合してはなりません。

id

指定されているアプリケーションの ID。この値は、AppStream 2.0 アプリケーションレポートの `application_name` の値に対応し、ユーザーがアプリケーションを起動したときに提供されます。ユーザーがストリーミングセッション中に 1 つ以上のアプリケーションを起動した日ごとに、[使用状況レポート](#) を有効にすると、AppStream 2.0 によってアプリケーションレポートが Amazon S3 バケットにエクスポートされます。アプリケーションレポートの詳細については、「[アプリケーションレポートのフィールド](#)」を参照してください。

型: 文字列

必須: はい

長さの制限: 最小長は 1、最大長は 512 文字。

displayName

指定されているアプリケーションの表示名。この名前は、アプリケーションカタログでユーザーに表示されます。

型: 文字列

必須: はい

長さの制限: 最小長は 1、最大長は 512 文字。

LaunchPath

起動するアプリケーションの実行可能ファイルへの Windows ファイルシステムパス。

型: 文字列

必須: はい

長さの制限: 最小長は 1、最大長は 32,767 文字。

iconData

アプリケーションカタログに表示する Base 64 エンコードイメージ。このイメージは .png、.jpeg、または .jpg のいずれかの形式にする必要があります。

型: 文字列

必須: はい

長さの制限: 最小長は 1、最大長は 1,000,000 文字。

launchParams

アプリケーションを起動するために使用されるパラメータです。

型: 文字列

必須: いいえ

長さ制限: 最大長は 32,000 文字です。

workingDirectory

アプリケーションを起動する作業ディレクトリへの Windows ファイルシステムパス。

型: 文字列

必須: いいえ

長さ制限: 最大長は 32,767 文字です。

RemoveApplicationsRequest オペレーション

AddApplicationsRequest オペレーションを使用して追加されたアプリケーションを削除します。アプリケーションは、ユーザーのアプリケーションカタログから削除されます。アプリケーションが削除されたら、起動することはできません。アプリケーションがまだ実行中の場合は、AppStream 2.0 はそのアプリケーションを閉じません。AppStream 2.0 イメージで直接指定されているアプリケーションは、削除できません。

リクエストの構文

```
### userSid;
```

```
list<Application> applications;
```

リクエストパラメータ

userSid

リクエストが適用されるユーザーの SID。

型: 文字列

必須: はい

長さの制限: 最小長は 1、最大長は 208 文字。

applications

リクエストが適用されるアプリケーションのリスト。

型: 文字列

必須: はい

ClearApplicationsRequest オペレーション

AddApplicationsRequest オペレーションを使用して、アプリケーションカタログに追加されたすべてのアプリケーションを削除する必要があります。アプリケーションが削除されたら、起動することはできません。ClearApplicationsRequest オペレーションが使用されているときにアプリケーションが実行されている場合、AppStream 2.0 はそれらを閉じません。AppStream 2.0 イメージで直接指定されているアプリケーションは、削除できません。

リクエストの構文

```
### userSid;
```

リクエストパラメータ

userSid

リクエストが適用されるユーザーの SID。

型: 文字列

必須: はい

長さの制限: 最小長は 1、最大長は 208 文字。

動的アプリケーションプロバイダーの有効化

動的アプリケーションプロバイダーを AppStream 2.0 イメージ内で最初に有効にする必要があります。これらのプロバイダーを有効にした後、それらはストリーミングインスタンス上のユーザーのアプリケーションを管理できます。

この機能を有効にするには、動的アプリケーションプロバイダーの詳細を Image Builder の設定ファイルに追加する必要があります。Image Builder は、Microsoft Active Directory ドメインに参加している必要があります。Image Builder で次のステップを実行してから、動的アプリケーションをテストして、期待どおりに機能することを確認できます。最後に、イメージの作成を完了します。

Note

サードパーティーの動的アプリケーションプロバイダーは、インストール中に設定ファイルを変更することがあります。インストールの手順については、アプリケーションプロバイダーのドキュメントを参照してください。

動的アプリケーションプロバイダーを有効にするには

1. 使用する Image Builder に接続し、Image Builder に対するローカル管理者権限を持つドメインアカウントでサインインします。そのためには、次のいずれかを実行します。
 - [AppStream 2.0 コンソールを使用する](#) (ウェブ接続のみ)
 - [ストリーミング URL を作成する](#) (ウェブまたは AppStream 2.0 クライアント接続用)

Note

組織でスマートカードサインインが必要な場合は、ストリーミング URL を作成し、接続に AppStream 2.0 クライアントを使用する必要があります。スマートカードサインインの詳細については、「[スマートカード](#)」をご参照ください。

2. C:\ProgramData\Amazon\AppStream\AppCatalogHelper\DynamicAppCatalog\ に移動して、[Agents.json] 設定ファイルを開きます。
3. [Agents.json] ファイルで、次のエントリを追加します。

「DisplayName」: 「<Uninstall hive display name value>」、

「Path」: 「<C:\path\to\client\application>」

DisplayName は、アプリケーション用に作成した [HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\Uninstall] キーの [DisplayName] レジストリ値と一致する必要があります。

4. 動的アプリケーションプロバイダーをインストールします。
5. Image Builder デスクトップから、Image Assistant を開きます。
6. 必要に応じて、イメージに含める他のアプリケーションをインストールします。
7. Image Assistant の [1] で、Add Apps] (1. アプリケーションを追加する) ページで、[Enable dynamic app providers] (動的アプリケーションプロバイダーの有効化) チェックボックスをオンにします。
8. 同じページで、ステップ 8 の説明に従って他のアプリケーションをインストールした場合は、[+Add App] を選択し、追加するアプリケーションを指定します。

 Note

動的アプリケーションプロバイダーを使用するときは、イメージ内のアプリケーションを指定する必要はありません。イメージでアプリケーションを指定した場合、それらを動的アプリケーションプロバイダーで削除することはできません。

9. 次のセクションの手順に進み、動的アプリケーションプロバイダーをテストします。

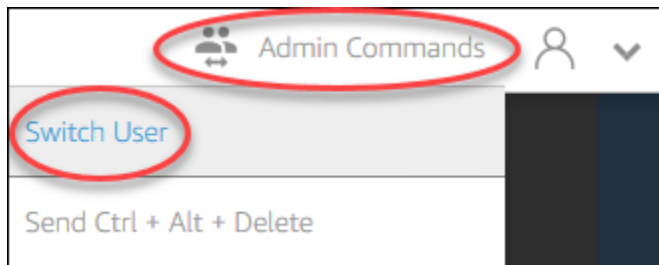
動的アプリケーションプロバイダーのテスト (オプション)

Image Builder で動的アプリケーションプロバイダーを有効にしたら、プロバイダーをテストして期待どおりに機能することを確認できます。これを行うには、イメージの作成を終了する前に以下の手順を実行してください。

動的アプリケーションプロバイダーをテストするには

1. 次のいずれかを行います：
 - 動的アプリケーションプロバイダーを有効にした Image Builder に接続して、管理者としてすでにログオンしている場合は、Image Builder のローカル管理者アクセス許可を持たないアカウントに切り替える必要があります。これを行うには、Image Builder セッションツールバー

の右上で [Admin Commands (管理者コマンド)]、[Switch User (ユーザーの切り替え)] を選択します。



- Image Builder にまだ接続していない場合は、[AppStream 2.0 コンソールを使用](#) (ウェブ接続のみ) するか、[ストリーミング URL を作成](#) (ウェブまたは AppStream 2.0 クライアント接続の場合) して接続します。

Note

サインインを求められたら、[ディレクトリユーザー] を選択し、Image Builder に対してローカル管理者アクセス許可を持たないドメインアカウントでサインインします。

2. Image Builder デスクトップで Image Assistant を開きます (開いていない場合)。
3. [Test Apps (アプリケーションのテスト)] ページで、動的アプリケーションプロバイダー以外のアプリケーションをイメージ内で指定した場合は、それがリストの最初に表示されます。動的アプリケーションプロバイダーからのアプリケーションがリストに表示されるまでに、少し時間がかかる場合があります。
4. リストからアプリケーションを選択して開き、期待どおりに機能することを確認します。
5. テストが完了したら、[Test Apps (アプリケーションのテスト)] ページの右下で、[Switch user (ユーザーの切り替え)] を選択します。
6. [管理者] を選択して、Image Builder に再度ログインします。
7. Image Assistant で、必要な手順に従って、イメージの作成を完了します。イメージを作成する方法については、「[チュートリアル: AppStream 2.0 コンソールを使用してカスタム AppStream 2.0 イメージを作成する](#)」を参照してください。

AppStream 2.0 は、[Agents.json] 構成ファイルに指定されているエージェントを自動的に最適化します。

動的アプリケーションプロバイダーと動的アプリケーションフレームワークについて学ぶための追加リソース

次のリンクでは、動的アプリケーションプロバイダーと動的アプリケーションフレームワークの詳細を学べるよう情報を提供します。

ソリューション	説明
Liquidware FlexApp	FlexApp — Liquidware FlexApp の概要を説明します。 FlexApp は、AppStream 2.0 動的アプリケーションフレームワークを使用して、アプリケーションの使用権限と配信をリアルタイムで管理しているサードパーティープロバイダーです。FlexApp レイヤーは、Windows オペレーティングシステムのバージョンや配信プラットフォームに関係なく、あらゆる Windows デスクトップ環境にアプリケーションを配信します。
App-V	動的アプリケーションフレームワークを使用して App-V パッケージを AppStream 2.0 に持ち込む — 動的アプリケーションフレームワークを使用して App-V を AppStream 2.0 と統合する方法について説明します。
AppStream 2.0	AppStream 2.0 動的アプリケーションフレームワークを使用して動的アプリケーションプロバイダーを構築する — AppStream 2.0 動的アプリケーションフレームワークを使用して、独自の動的アプリプロバイダーを開発する方法について説明します。

ユーザーに AppStream 2.0 へのアクセスを提供する

ユーザーは、サポートされているデバイス上のウェブブラウザまたは AppStream 2.0 クライアントを使用して AppStream 2.0 ストリーミングセッションにアクセスできます。

組織の要件に応じて、SAML 2.0 を使用して ID フェデレーションを設定するか、AppStream 2.0 ユーザープールを使用するか、ストリーミング URL を作成することにより、AppStream 2.0 ストリーミングセッションへのユーザーアクセスを有効にすることができます。接続方法を選択する際の推奨事項を次に示します。

- [SAML 2.0](#): ユーザーを管理し、SAML 2.0 フェデレーションをサポートする ID プロバイダーがある場合は、この接続方法を使用します。

Note

この接続方法は、AppStream 2.0 フリートが Microsoft Active Directory ドメインに参加している場合に必要です。

- [AppStream 2.0 ユーザープール](#): 次の場合にこの接続方法を使用します。
 - SAML 2.0 準拠の ID プロバイダーを構成する前に、概念証明 (POC) をすばやくセットアップする必要があります。
 - SAML 2.0 に準拠する ID プロバイダーがありません。
 - AppStream 2.0 コンソール内で直接ユーザーを管理する場合。
- [ストリーミング URL](#): この接続方法は、一時的な URL を使用して AppStream 2.0 へのアクセスをプログラムで提供する場合に使用します。既存の ID プロバイダーを使用して AppStream 2.0 へのプログラムによるアクセスを提供する場合は、この接続方法をお勧めします。

サポートされている機能

次の表は、さまざまなアクセスタイプでサポートされている機能を比較したものです。

機能	ブラウザベースのアクセス	Windows のクライアントベースのアクセス	macOS のクライアントベースのアクセス	メモ
エンタープライズデプロイツール	✗	✓	✗	詳細については、「 the section called “チュートリアル: クライアントをインストールし、クライアントエクスペリエンスをカスタマイズする” 」を参照してください。
HIPAA/PCI への準拠	✓	✓	✓	詳細については、「 コンプライアンス 」を参照してください。
Active Directory 認証	✓	✓	✓	詳細については、「 アクティブディレクトリの使用 」を参照し

機能	ブラウザ ベースの アクセス	Windows のクライ アントベ ースの アクセス	macOS の クライ アントベ ースの アクセス	メモ
				てください。
MFA (多要素認証)	✓	✓	✓	AppStream 2.0 では、MFA が SAML 2.0 経由でサポートされています。
スマートカード (CAC および PIV リーダー)	✗	✓	✗	詳細については、「 the section called “スマートカードリダイレクト” 」を参照してください。
アクセスコントロール用の証明書 (OS ベース)	✓	✓	✓	AppStream 2.0 では、証明書認証は SAML 2.0 経由でサポートされています。

機能	ブラウザベースのアクセス	Windows のクライアントベースのアクセス	macOS のクライアントベースのアクセス	メモ
証明書ベースの認証	✓	✓	✓	詳細については、「 the section called “証明書ベースの認証” 」を参照してください。
クライアントのカスタマイズ	利用可能 (例外あり)	利用可能 (例外あり)	利用可能 (例外あり)	AppStream 2.0 はウェブベースのブランドとカスタム URL をサポートしています。詳細については、「 カスタムブランディングを追加する 」を参照してください。
デスクトップビュー接続モード	✓	✓	✓	
クラシックアプリケーションモード	✓	✓	✓	
ネイティブアプリケーションモード	✗	✓	✗	

機能	ブラウザ ベースの アクセス	Windows のクライ アントベ ースの アクセス	macOS の クライ アントベ ースの アクセス	メモ
USB リダイレクト	✗	✓	✗	Windows ベースのフ リートにア クセスする AppStream 2.0 クライ アントでサ ポートされ ます。詳細 については、「 the section called “USB リダイレクト” 」を参照してください。
音声入力 (ウェブ会議や通話用)	✓	✓	✓	Linux では サポート されませ ん。AppStr eam 2.0 は USB マイ クをサポ ートして います。
ビデオ入力 (会議アプリケーション)	✓	✓	✓	

機能	ブラウザベースのアクセス	Windows のクライアントベースのアクセス	macOS のクライアントベースのアクセス	メモ
ストレージのリダイレクト	x	✓	x	詳細については、「 the section called “ファイルシステムリダイレクトの有効化” 」を参照してください。
USB/ローカルプリンターへのリダイレクト	利用可能 (例外あり)	✓	✓	ブラウザでの AppStream 2.0 の間接的な印刷。フルリダイレクトは、Linux ベースのスタックではサポートされていません。
クリップボードのリダイレクト	✓	✓	✓	
タブレットの描画	✓	✓	x	

機能	ブラウザ ベースの アクセス	Windows のクライ アントベ ースの アクセス	macOS の クライ アントベ ースの アクセス	メモ
YubiKey のサポート	x	利用可能 (例外あり)	x	AppStream 2.0 クライ アントでサ ポートされ ています。 詳細につい ては、「 the section called “ス トリーミ ングアプリ ケーション で使用する USB デバ イスを認証 する” 」を 参照してく ださい。

機能	ブラウザベースのアクセス	Windows のクライアントベースのアクセス	macOS のクライアントベースのアクセス	メモ
モニターのサポート	ウェブアクセスデュアルモニターのサポート。詳細については、「 the section called “デュアルモニターのサポート” 」を参照してください。	✓	✓	詳細については、「 the section called “マルチモニター” 」を参照してください。
ユーザープール	✓	✓	✗	詳細については、「 the section called “ユーザープール” 」を参照してください。

機能	ブラウザベースのアクセス	Windows のクライアントベースのアクセス	macOS のクライアントベースのアクセス	メモ
App Block Builder に接続する	✓	✓	x	詳細については、「 the section called “App Block Builder” 」を参照してください。
マルチスタックアプリケーションにアクセスする	✓	✓	✓	詳細については、「 the section called “属性ベースのアプリケーションの使用権限” 」を参照してください。

次のトピックでは、アプリケーションストリーミング用に AppStream 2.0 へのユーザーアクセスを設定する方法について説明します。

アプリケーションストリーミングの使用を開始する際に役に立つようにユーザーに提供できるガイダンスについては、[AppStream 2.0 ユーザー向けのガイダンス](#) を参照してください。

ウェブブラウザを介したアクセスを許可する

ユーザーは、サポートされているデバイスのウェブブラウザまたは AppStream 2.0 クライアントアプリケーションを使用して AppStream 2.0 ストリーミングセッションを開始できます。以下のトピックでは、ユーザーにウェブブラウザを介したアクセスを許可できるように役立つ情報を提供します。

内容

- [システムの要件と機能のサポート \(ウェブブラウザ\)](#)
- [AppStream 2.0 ユーザーの接続方法を構成する \(ウェブブラウザ\)](#)

AppStream 2.0 クライアントを介して AppStream 2.0 へのユーザーアクセスを提供する方法については、[AppStream 2.0 クライアント経由でアクセスを提供する](#) を参照してください。

システムの要件と機能のサポート (ウェブブラウザ)

このトピックでは、ウェブブラウザを使用して AppStream 2.0 へのユーザーアクセスを提供するための要件を理解するのに役立つ情報を提供します。また、サポートされている機能についても説明しています。

トピック

- [システム要件と考慮事項](#)
- [機能とデバイスのサポート](#)

システム要件と考慮事項

ユーザーは、Windows、Mac、Chromebook、Linux コンピュータなどのデスクトップコンピュータから HTML5 対応のウェブブラウザを介して AppStream 2.0 にアクセスできます。使用できる HTML5 対応のウェブブラウザには、次のものがあります。

- Google Chrome
- Mozilla Firefox
- Safari
- Microsoft Edge

ウェブブラウザで AppStream 2.0 を使用する際に、ブラウザの拡張機能やプラグインは必要ありません。

ユーザーは、次のブラウザおよびデバイス上で AppStream 2.0 フリートのストリーミングセッションにアクセスすることもできます。

- iPad (iOS 11 以降) の Chrome または Safari
- Android (Android 8 以降)

- Microsoft Surface Pro (Windows 10) タブレット

AppStream 2.0 は、画面解像度が 1,024x768 ピクセル未満のデバイスではサポートされません。

機能とデバイスのサポート

AppStream 2.0 は、ウェブブラウザから AppStream 2.0 にアクセスするユーザーに対して、次の機能と周辺機器のサポートを提供します。

トピック

- [デュアルモニターのサポート](#)
- [タッチスクリーンデバイスのサポート](#)
- [ドローイングタブレットのサポート](#)
- [相対マウスオフセット](#)

デュアルモニターのサポート

AppStream 2.0 では、異なる解像度のモニターを含め、ストリーミングセッション中に複数のモニターを使用できます。最適なストリーミング体験を実現するために、解像度の異なるモニターを使用しているユーザーは、モニターのディスプレイスケールを 100% に設定することをお勧めします。

デュアルモニターは、次のウェブブラウザで開始されるストリーミングセッションでサポートを提供します。

- Google Chrome
- Mozilla Firefox
- Safari

デュアルモニターでのブラウザベースのストリーミングセッションでは、モニターごとに最大ディスプレイ解像度 2,560 x 1,600 ピクセルがサポートされます。ユーザーが 2 つ以上のモニターを必要とする場合、またはモニターあたり 2,560 x 1,600 ピクセルを超えるディスプレイ解像度を必要とする場合は、AppStream 2.0 クライアントを使用できます。

Note

デュアルモニターは、モバイルデバイスや埋め込み AppStream 2.0 ストリーミングセッションではサポートされません。

ストリーミングセッションのユーザー接続に加えて、AppStream 2.0 では、Image Builder への管理接続用のデュアルモニターの使用もサポートしています。

タッチスクリーンデバイスのサポート

AppStream 2.0 は、タッチ対応の iPad、Android タブレット、Windows デバイスでのジェスチャをサポートしています。すべてのタッチイベントはストリーミングセッションに渡され、Windows の規約に従って処理されます。サポートされているタッチジェスチャは、ロングタップによる右クリック、スワイプによるスクロール、ピンチによるズーム、2 本指での回転などであり、アプリケーションで使用できます。

Note

タッチ対応デバイスでジェスチャのサポートを有効にするには、AppStream 2.0 イメージで、2019 年 3 月 7 日以降にリリースされたバージョンの AppStream 2.0 エージェントを使用する必要があります。詳細については、「[AppStream 2.0 エージェントのリリースノート](#)」を参照してください。

AppStream 2.0 ストリーミングセッションでタッチ対応デバイスの使用を開始するために役立つユーザー向けのガイダンスについては、[タッチスクリーンデバイス](#) を参照してください。

ドローイングタブレットのサポート

ドローイングタブレット (ペンタブレットとも呼ばれます) は、スタイラス (ペン) を使って絵を描くことができるコンピュータ入力デバイスです。AppStream 2.0 では、ユーザーが Wacom ペンタブレットなどのドローイングタブレットを、ローカルコンピュータに接続してストリーミングアプリケーションで使用できるようになりました。

次に、ユーザーがストリーミングアプリケーションでドローイングタブレットを使用するための要件と考慮事項を示します。

- 顧客がこの機能を利用できるようにするには、Windows Server 2019 を実行するためのイメージを使用するように AppStream 2.0 フリートを設定する必要があります。

- この機能を使用するには、ユーザーは Google Chrome または Mozilla Firefox ブラウザのみ、または AppStream 2.0 クライアントから AppStream 2.0 にアクセスする必要があります。
- ストリーミングアプリケーションは、Windows Ink テクノロジーをサポートしている必要があります。詳細については、「[Windows アプリでのペン操作と Windows インク](#)」を参照してください。
- GIMP などの一部のアプリケーションでは、圧力感度をサポートするために、ストリーミングインスタンス上のドロ잉タブレットを検出する必要があります。この場合、ユーザーは AppStream 2.0 クライアントを使用して AppStream 2.0 にアクセスし、これらの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。さらに、ユーザーのドロ잉タブレットを認定する必要があります。ユーザーは、新しいストリーミングセッションを開始するたびに、ドロ잉タブレットを AppStream 2.0 と共有する必要があります。詳しいガイダンスについては、[ストリーミングアプリケーションで使用する USB デバイスを認証する](#) を参照してください。
- この機能は Chromebook ではサポートされていません。

アプリケーションストリーミングセッション中にドロ잉タブレットの使用を開始するには、ユーザーは USB を使用してドロ잉タブレットをローカルコンピュータに接続し、サポートされているウェブブラウザまたは AppStream 2.0 クライアント (インストールされている場合) を使用してストリーミングセッションを開始します。この機能を使用するために、USB リダイレクトは必要ありません。

相対マウスオフセット

デフォルトでは、ユーザーのストリーミングセッション中に、AppStream 2.0 は絶対座標を使用し、マウスの動きをローカルにレンダリングすることによって、マウスの動きに関する情報をストリーミングインスタンスに送信します。コンピュータ支援設計 (CAD)/コンピュータ支援製造 (CAM) ソフトウェアやビデオゲームなど、グラフィックを多用するアプリケーションの場合、相対的マウスモードが有効になっているときにマウスのパフォーマンスが向上します。相対的マウスモードでは、ウィンドウまたは画面内の XY 座標の絶対値ではなく、最終フレームからマウスが移動した距離を表す相対座標が使用されます。相対的マウスモードが有効になっている場合、AppStream 2.0 はマウスの動きをリモートでレンダリングします。

ユーザーは、次のいずれかの操作を行って、AppStream 2.0 ストリーミングセッション中にこの機能を有効にできます。

- Windows: Ctrl + Shift + F8 を押す
- Mac: Ctrl + Fn + Shift + F8 を押す

AppStream 2.0 ユーザーの接続方法を構成する (ウェブブラウザ)

組織の要件に応じて、SAML 2.0 を使用した ID フェデレーションの設定、AppStream 2.0 ユーザープールの使用、ストリーミング URL の作成のいずれかを実行して、ウェブブラウザから AppStream 2.0 へのアクセスをユーザーに提供できます。

内容

- [SAML 2.0](#)
- [AppStream 2.0 ユーザープール](#)
- [ストリーミング URL](#)
- [次のステップ](#)

SAML 2.0

ユーザーは、内部の組織ポータルにアクセスするためにユーザーに提供された URL を入力します。組織の認証情報を入力すると、AppStream 2.0 にリダイレクトされます。

詳細については、「[SAML のセットアップ](#)」を参照してください。

Note

組織で、Active Directory に参加しているストリーミングインスタンスへの Windows サインイン用のスマートカードと、ストリーミングアプリケーションのセッション内認証が必要な場合は、ユーザーが AppStream 2.0 クライアントをインストールして使用する必要があります。詳細については、「[スマートカード](#)」を参照してください。

AppStream 2.0 ユーザープール

AppStream 2.0 ユーザープールで新規ユーザーを作成するか、ユーザープールユーザーを AppStream 2.0 スタックに割り当てると、AppStream 2.0 はユーザーに代わって E メールを送信します。ユーザーは、ようこそメールに記載された URL を入力し、認証情報を入力して、[Connect (接続)] を選択します。

詳細については、「[Amazon AppStream 2.0 ユーザープール](#)」を参照してください。

ストリーミング URL

ストリーミング URL を作成するには、以下のいずれかの方法を使用します。

- AppStream 2.0 コンソール
- [CreateStreamingURL](#) API アクション
- [create-streaming-url](#) AWS CLI コマンド

AppStream 2.0 コンソールを使用してストリーミング URL を作成するには、以下の手順を実行します。

AppStream 2.0 コンソールを使用してストリーミング URL を作成するには

1. AppStream 2.0 コンソールを <https://console.aws.amazon.com/appstream2> で開きます。
2. ナビゲーションペインの [Fleets] を選択します。
3. フリートのリストで、ストリーミング URL を作成するスタックに関連付けられているフリートを選択します。フリートのステータスが [Running (実行中)] であることを確認します。
4. ナビゲーションペインで、[Stacks] を選択します。スタックを選択してから、[Actions (アクション)]、[Create Streaming URL (ストリーミング URL の作成)] を選択します。
5. [User id (ユーザーID)] に、ユーザー ID を入力します。
6. [URL Expiration (URL の有効期限)] で、有効期限を選択し、生成された URL が有効な期間の長さを指定します。この URL は最大 7 日間有効です。
7. [Get URL (URL を取得)] を選択します。
8. URL をコピーし、アクセス可能な場所に保存して、ユーザーに提供します。

次のステップ

ウェブブラウザ接続方法を構成したら、AppStream 2.0 に接続してストリーミングセッションを開始できるように、[AppStream 2.0 に接続する](#) のステップバイステップガイダンスをユーザーに提供できます。

AppStream 2.0 クライアント経由でアクセスを提供する

ユーザーはサポートされているデバイスに、AppStream 2.0 クライアントアプリケーションまたはウェブブラウザを使用して AppStream 2.0 ストリーミングセッションを開始できます。

AppStream 2.0 クライアントは、AppStream 2.0 ストリーミングセッション中に次の機能を必要とするユーザー用に設計されたネイティブアプリケーションです。

- 2 台以上のモニターまたは 4K 解像度のサポートが必要なユーザー。

- AppStream 2.0 を介してストリーミングされたアプリケーションで USB デバイスを使用しているユーザー。
- ストリーミングセッション内のビデオ会議にローカルウェブカメラを使用すると、使用中のブラウザはビデオまたは音声入力をサポートしません。
- ストリーミングセッション中にキーボードショートカットを使用しているユーザー。
- ストリーミングセッション中にローカルドライブとフォルダへのシームレスなアクセスが必要なユーザー。
- プリントジョブをストリーミングアプリケーションから、ローカルコンピュータに接続されているプリンターにリダイレクトする機能が必要です。
- ローカルにインストールされたアプリケーションを操作するのとほぼ同じ方法で、リモートストリーミングアプリケーションを操作することを希望するユーザー。

以下のトピックでは、ユーザーに AppStream 2.0 クライアントを介したアクセスを許可するのに役立つ情報を提供します。ユーザーにウェブブラウザを介した AppStream 2.0 へのアクセスを許可する方法については、[ウェブブラウザを介したアクセスを許可する](#) を参照してください。

目次

- [システム要件と機能のサポート \(AppStream 2.0 クライアント\)](#)
- [AppStream 2.0 クライアントをインストールして構成する](#)

システム要件と機能のサポート (AppStream 2.0 クライアント)

このトピックでは、AppStream 2.0 クライアントの要件とサポートされる機能を理解するのに役立つ情報を提供します。

トピック

- [システム要件と考慮事項](#)
- [機能とデバイスのサポート](#)

システム要件と考慮事項

AppStream 2.0 クライアントには以下の要件があります。

- AppStream 2.0 クライアントを起動する時は、最小特権の原則に従ってください。クライアントは、タスクを完了するために必要な特権レベルのみで実行される必要があります。

- クライアントの要件
 - Windows クライアント
 - オペレーティングシステム — Windows 10 (32 ビットまたは 64 ビット)、Windows 11 (64 ビット)
 - Microsoft Visual C++ 2019 バージョン 14.20.xx 再配布可能バージョン以降。詳細については、Microsoft サポートドキュメントの「[最新の Microsoft Visual C++ 再頒布可能バージョン](#)」を参照してください。
 - RAM — 2 GB 以上
 - ハードドライブ容量 — 200 MB 以上
 - macOS クライアント
 - オペレーティングシステム — macOS 13 (Ventura)、macOS 14 (Sonoma)、macOS 15 (Sequoia)
 - ハードドライブ容量 — 200 MB 以上
- ローカル管理者権限 — USB ドライバーのサポートとして AppStream 2.0 USB ドライバーをインストールする場合に使用します。

 Note

macOS クライアントでは、ローカル管理者権限はサポートされていません。

- 2018 年 11 月 14 日以降に発行された最新の AppStream 2.0 エージェントまたはエージェントバージョンを使用する AppStream 2.0 イメージ。AppStream 2.0 エージェントのバージョンについては、「[AppStream 2.0 エージェントのリリースノート](#)」を参照してください。
- クライアントは、UDP と NICE DCV 経由のデフォルトの TCP ベースのストリーミングをサポートします。NICE DCV および UDP の詳細については、「[QUIC UDP トラnsポートプロトコルを有効にする](#)」を参照してください。クライアントの UDP ストリーミングを有効にする場合は、次の要件を満たしていることを確認してください。次の要件を満たさない場合、クライアントはデフォルトで TCP ベースのストリーミングに戻ります。
- スタックは、ストリーミング設定エクスペリエンスセクションで UDP を優先するように設定されています。詳細については、「[Amazon AppStream 2.0 フリートとスタックを作成する](#)」を参照してください。
- ネットワークでは、AWS IP 範囲のポート 8433 で UDP トラフィックが許可されます。詳細については、「[AWS および IP アドレスの範囲](#)」を参照してください。

- フリートを作成するときに最新のベースイメージを使用しています。詳細については、「[AppStream 2.0 ベースイメージとマネージド型イメージの更新に関するリリースノート](#)」を参照してください。
- エンドユーザーが最新のクライアントを使用している。詳細については、「[サポートされるクライアント](#)」を参照してください。

Note

AppStream 2.0 クライアントのインストールには、インターネット接続を使用することをお勧めします。インターネットに接続されていないコンピュータにはクライアントをインストールできない場合や、AppStream 2.0 からストリーミングされたアプリケーションを USB デバイスで使用できない場合があります。詳細については、「[AppStream 2.0 ユーザーの問題のトラブルシューティング](#)」を参照してください。

機能とデバイスのサポート

AppStream 2.0 クライアントは、次の機能とデバイスをサポートしています。

トピック

- [ネイティブアプリケーションモード](#)
- [診断ログの自動およびオンデマンドのアップロード](#)
- [周辺機器](#)

ネイティブアプリケーションモード

Note

Linux インスタンスからストリーミングする場合、または Amazon AppStream 2.0 macOS クライアントアプリケーションを使用する場合、ネイティブアプリケーションモードは使用できません。

ネイティブアプリケーションモードでは、AppStream 2.0 ストリーミングセッション中にユーザーは使い慣れたエクスペリエンスを得られます。ユーザーがこのモードで AppStream 2.0 に接続すると、ローカルコンピュータにインストールされているアプリケーションを操作するのとほぼ同じ方法

で、リモートストリーミングアプリケーションを操作できます。ネイティブアプリケーションモードの各ストリーミングアプリケーションは、独自のウィンドウで開き、ユーザーのローカル PC のタスクバーにアプリケーションのアイコンが表示されます。

ユーザーにクラシックモードでのみ AppStream 2.0 に接続させる場合は、ネイティブアプリケーションモードを無効にするように `NativeAppModeDisabled` レジストリを設定できます。詳細については、「[ネイティブアプリケーションモードを無効にするかどうかを選択する](#)」を参照してください。

ネイティブアプリケーションモードとクラシックモードの詳細、およびユーザーに提供できるガイドンスについては、[AppStream 2.0 クライアント接続モード](#) を参照してください。

Note

[Desktop (デスクトップ)] ストリームビューでフリートが有効になっている場合、ネイティブアプリケーションモードは使用できません。[Desktop] (デスクトップ) ストリームビューを設定する方法については、「[Amazon AppStream 2.0 でフリートを作成する](#)」をご参照ください。

要件

ユーザーに対してこの機能を有効にするには、2020 年 2 月 19 日以降にリリースされた [AppStream 2.0 エージェントのバージョン](#) を使用するイメージを使用する必要があります。また、AppStream 2.0 クライアントのバージョン 1.1.129 以降をユーザーの PC にインストールする必要があります。クライアントのバージョンの詳細については、[AppStream 2.0 クライアントリリースノート](#) を参照してください。

AppStream 2.0 クライアントのバージョン 1.1.129 以降がユーザーのコンピュータにインストールされているが、使用しているイメージで 2020 年 2 月 19 日以降にリリースされたエージェントバージョンが使用されていない場合は、ネイティブアプリケーションモードを選択していても、クライアントはクラシックモードに戻ります。

既知の問題

ネイティブアプリケーションモードでのストリーミングセッション中に、ユーザーがあるブラウザウィンドウのタブを別のウィンドウに対してドッキングまたはドッキング解除しようとする、リモートストリーミングブラウザはローカルブラウザと同じようには機能しません。ネイティブアプリケーションモードでのストリーミングセッション中にこのタスクを実行するには、ユーザーは Alt キーを押しながら、ブラウザタブを別のブラウザウィンドウにドッキングする必要があります。

診断ログの自動およびオンデマンドのアップロード

ユーザーが AppStream 2.0 クライアントを使用しているときに問題が発生した場合のトラブルシューティングに役立てるために、診断ログの自動またはオンデマンドのアップロードを自分で有効にするか、ユーザーに有効にしてもらうことができます。

Note

診断ログに機密情報は含まれません。自分が管理しているユーザー PC で診断ログの自動およびオンデマンドのアップロードを無効にしたり、これらの機能を無効にすることをユーザーに許可したりすることはできません。

診断ログの自動アップロード

自分が管理する PC にクライアントをインストールするときに、診断ログを自動的にアップロードするように AppStream 2.0 クライアントを設定できます。これにより、クライアントで問題が発生した場合も、ユーザーによる操作なしで、ログが AppStream 2.0 (AWS) に送信されます。詳細については、「[ユーザーの追加の AppStream 2.0 クライアント設定を設定する](#)」を参照してください。

または、ユーザーによる AppStream 2.0 クライアントのインストール時またはインストール後に、診断ログの自動アップロードを有効にするかどうかをユーザーに選択させることができます。このタスクの実行に役立つようにユーザーに提供できるガイダンスについては、「[Windows のセットアップ](#)」をご参照ください。

診断ログのオンデマンドアップロード

ログ記録をより詳細に制御する場合は、自動のログ記録を無効にし、診断ログのオンデマンドアップロードを有効にすることができます。診断ログのオンデマンドアップロードをユーザーに許可することで、例外が発生した場合やクライアントが応答しなくなった場合に、ミニダンプ (エラーレポート) を AppStream 2.0 (AWS) に送信するかどうか、ユーザーが選択できるようになります。

このタスクの実行に役立つユーザー向けのガイダンスについては、[ログ記録](#) を参照してください。

周辺機器

AppStream 2.0 クライアントは、モニター、ウェブカメラ、マウス、キーボード、ドローイングタブレットなどの周辺機器に対して次のサポートを提供します。

Note

特定の例外を除き、AppStream 2.0 クライアントが USB デバイスをサポートするには USB リダイレクトが必要です。また、ほとんどの場合、デバイスに USB リダイレクトが必要な場合は、AppStream 2.0 ストリーミングセッションで使用する前に、デバイスを修飾する必要があります。詳細については、「[USB リダイレクト](#)」を参照してください。

トピック

- [マルチモニター](#)
- [リアルタイムオーディオビデオ](#)
- [USB デバイス](#)
- [ドローイングタブレット](#)
- [キーボードショートカット](#)
- [相対マウスオフセット](#)

マルチモニター

AppStream 2.0 では、異なる解像度のモニターを含め、ストリーミングセッション中に複数のモニターを使用できます。最適なストリーミング体験を実現するために、解像度の異なるモニターを使用しているユーザーは、モニターのディスプレイスケールを 100% に設定することをお勧めします。

Note

[ネイティブアプリケーションモード](#)を使用する AppStream 2.0 ストリーミングセッションでは、最大 2K 解像度のモニターがサポートされます。高解像度のモニターをストリーミングセッションに使用すると、AppStream 2.0 クライアントはクラシックモードに戻ります。このシナリオでは、AppStream 2.0 クラシックモードのストリーミングビューが、画面の 2K を占有し、画面の残りの部分は黒になります。

マルチモニター (最大 2K 解像度)

次の AppStream 2.0 インスタンスタイプは、最大 4 台のモニターをサポートし、モニターあたり最大 2560 x 1600 ピクセルの表示解像度をサポートします。汎用、メモリ最適化、コンピューティング最適化、グラフィックス G4dn、グラフィックス G5、グラフィックスデザイン、グラフィックスプロ。

マルチモニター (最大 4K 解像度)

次の AppStream 2.0 インスタンスタイプは、最大 2 台のモニターをサポートし、モニターあたりの最大ディスプレイ解像度は 4096 x 2160 ピクセルです: Graphics G4dn、Graphics G5、Graphics Design、Graphics Pro。

Note

グラフィック以外のインスタンスタイプ (汎用、メモリ最適化、コンピューティング最適化) は、モニターあたり 2,560 x 1,600 ピクセルの最大ディスプレイ解像度をサポートします。

リアルタイムオーディオビデオ

AppStream 2.0 は、ローカルウェブカメラ動画入力を AppStream 2.0 ストリーミングセッションにリダイレクトすることにより、リアルタイム音声動画 (AV) をサポートしています。この機能により、ユーザーは AppStream 2.0 ストリーミングセッション内でビデオ会議や音声会議にローカルウェブカメラを使用できます。リアルタイム AV とリアルタイムオーディオのサポートにより、ユーザーは AppStream 2.0 ストリーミングセッションを離れることなく、使い慣れたビデオおよび音声会議アプリケーションを使用して共同作業を行うことができます。

ユーザーが AppStream 2.0 ストリーミングセッション内からビデオ会議を開始すると、AppStream 2.0 は、ウェブカメラのビデオとマイクのオーディオ入力をローカルで圧縮してから、このデータを安全なチャネル経由でストリーミングインスタンスに送信します。ストリーミングセッション中に、ユーザーは AppStream 2.0 ツールバーを使用して音声および動画入力を有効にできます。ユーザーが複数のウェブカメラを持っている場合 (ローカルコンピュータに接続されている USBウェブカメラと組み込みのウェブカメラがある場合など)、ストリーミングセッション中に使用するウェブカメラを選択することもできます。

Note

マルチセッションフリートでは、イン/アウト機能のみにアクセスできます。動画入力 (ウェブカメラサポート) は、マルチセッションフリートではまだ利用できません。

リアルタイム AV のサポートを設定およびテストするには、次の手順を実行します。

リアルタイム AV サポートの設定とテスト

1. 新しい Image Builder を作成するか、次の要件を満たす既存の Image Builder に接続します。

- Image Builder は、Windows Server 2016 または Windows Server 2019 を実行する必要があります。
- Image Builder は、2021 年 6 月 1 日以降にリリースされたバージョンの AppStream 2.0 エージェントを使用する必要があります。
- 2021 年 5 月 17 日以降にリリースされた AppStream 2.0 エージェントの場合、リアルタイム AV はデフォルトで有効になっています。テスト用のストリーミング URL を作成するには、ステップ 3~6 を省略し、Image Builder からの接続を解除します。リアルタイム AV を無効にする必要がある場合は、ステップ 4 でウェブカメラのアクセス許可を無効にしながら、すべてのステップを完了します。
- Image Builder が、ウェブブラウザへのアクセスを使用して動画に接続する場合には、2021 年 6 月 24 日以降にリリースされたバージョンの AppStream 2.0 エージェントを使用する必要があります。サポートされるウェブブラウザの詳細については、「[the section called “ウェブブラウザアクセス”](#)」を参照してください。

Image Builder を作成する方法については、[Image Builder を起動し、ストリーミングアプリケーションをインストールして設定する](#) を参照してください。

2. 使用する Image Builder に接続し、管理者としてサインインします。Image Builder に接続するには、次のいずれかを行います。
 - [AppStream 2.0 コンソールを使用する](#) (ウェブ接続のみ)
 - [ストリーミング URL を作成する](#) (ウェブまたは AppStream 2.0 クライアント接続用)

 Note

接続先の Image Builder が Active Directory ドメインに参加していて、組織でスマートカードサインインが必要な場合は、ストリーミング URL を作成し、接続に AppStream 2.0 クライアントを使用する必要があります。スマートカードサインインの詳細については、「[スマートカード](#)」をご参照ください。

3. Image Builder で、レジストリエディタを開きます。これを行うには、Image Builder のデスクトップで、タスクバーの検索ボックスに「**regedit**」と入力してください。次に、レジストリエディタで最上位の結果を選択します。
4. HKEY_LOCAL_MACHINE\SOFTWARE\Amazon\AppStream\ で、次の種類、名前、および値のデータを持つ新しいレジストリ値を作成します。

- レジストリ値の種類: DWORD
 - レジストリ値の名前: WebCamPermission
 - レジストリ値のデータ (ヘキサデミカル): ウェブカメラのアクセスを許可 (有効化) する場合は 1 を、拒否 (無効化) する場合には 0 を設定
5. レジストリ値を作成した後、[テンプレートユーザー] または Image Builder の管理者権限を持たないドメインアカウントに切り替えます。[Template User (テンプレートユーザー)] に切り替えるには、セッションウィンドウの右上にあるツールバーで、[Admin Commands (管理コマンド)]、[Switch User (ユーザーの切り替え)]、[Template User (テンプレートユーザー)] の順に選択します。
 6. [Administrator (管理者)] に戻ります。
 7. Image Builder から接続解除し、Image Builder のストリーミング URL を作成します。そのためには、次の操作を行います。
 - a. AppStream 2.0 コンソールを <https://console.aws.amazon.com/appstream2> で開きます。
 - b. ナビゲーションペインで、[Images (イメージ)]、[Image Builder] の順に選択します。
 - c. 接続解除した Image Builder を選択し、[Actions (アクション)]、[Create streaming URL (ストリーミング URL の作成)] の順に選択します。
 - d. [Copy Link (リンクをコピー)] を選択し、セキュリティで保護されているアクセス可能な場所にリンクを保存します。次の手順のリンクを使用して、Image Builder に接続します。
 8. 作成したストリーミング URL を介して、AppStream 2.0 クライアントもしくはウェブブラウザへのアクセスを使用しながら Image Builder に接続します。
 9. [ビデオ会議と音声会議](#) の手順に従って、Image Builder でリアルタイム AV エクスペリエンスをテストします。
 10. リアルタイム AV が予想どおりに動作していることを確認したら、ストリーミングセッションの接続を解除してから、Image Builder に再接続し、Image Assistant の必要な手順に従ってイメージの作成を完了します。イメージを作成する方法については、「[チュートリアル: AppStream 2.0 コンソールを使用してカスタム AppStream 2.0 イメージを作成する](#)」を参照してください。

Image Builder の設定とリアルタイム AV をサポートするイメージの作成が完了したら、AppStream 2.0 フリート内のユーザーに対し、この機能の使用を許可できるようになります。ユーザーのコンピュータに AppStream 2.0 クライアントのバージョン 1.1.257 以降がインストールされていることを確認します。

 Note

AppStream 2.0 クライアントでリアルタイム AV を使用する場合は、AppStream 2.0 のベースイメージとエージェントには、2021 年 6 月 1 日以降のバージョンを使用する必要があります。AppStream 2.0 クライアントには、最新バージョンの使用をお勧めします。リアルタイム AV を使用するためにユーザーに提供できるガイダンスについては、「[ビデオ会議と音声会議](#)」をご参照ください。

ウェブブラウザへのアクセスを通じてリアルタイム AV を使用する場合は、AppStream 2.0 イメージ用の AppStream 2.0 エージェントには、2021 年 6 月 24 日以降にリリースされたバージョンを使用する必要があります。サポートされるウェブブラウザの詳細については、「[the section called “ウェブブラウザアクセス”](#)」を参照してください。

USB デバイス

次のセクションでは、USB デバイスに対する AppStream 2.0 のサポートについて説明します。

目次

- [USB リダイレクト](#)
- [スマートカード](#)

USB リダイレクト

AppStream 2.0 ストリーミングセッション中にほとんどのローカル USB デバイスを使用するには、USB リダイレクトが必要です。USB リダイレクトが必要な場合は、ユーザーが AppStream 2.0 ストリーミングセッション中にデバイスを使用する前に、その[デバイスを認定](#)する必要があります。デバイスの認定後、ユーザーは [AppStream 2.0 でデバイスを共有する](#)必要があります。USB リダイレクトでは、AppStream 2.0 ストリーミングセッション中に、ユーザーのデバイスにアクセスしてローカルアプリケーションで使用することはできなくなります。

それ以外の場合は、USB デバイスは AppStream 2.0 で既に有効になっているため、それ以上の構成は必要ありません。たとえば、AppStream 2.0 クライアントをインストールすると、スマートカードリダイレクトはデフォルトで有効になっています。この機能を有効にすると、USB リダイレクトは使用されないため、スマートカードリーダーを認定する必要はなく、ユーザーがストリーミングセッション中にデバイスを使用するために AppStream 2.0 でこれらのデバイスを共有する必要はありません。

Note

USB リダイレクトは現在、Linux ベースのフリースタンス、または AppStream 2.0 macOS クライアントアプリケーションを使用する場合にはサポートされていません。

スマートカード

AppStream 2.0 では、Active Directory に参加しているストリーミングインスタンスへの Windows サインイン用のスマートカードの使用と、ストリーミングアプリケーション用のセッション内認証がサポートされています。スマートカードリダイレクトはデフォルトで有効になっているため、ユーザーは USB リダイレクトなしで、ローカルコンピュータとスマートカードに接続されているスマートカードリーダーを使用できます。

目次

- [Windows サインインとセッション内認証](#)
- [スマートカードリダイレクト](#)

Windows サインインとセッション内認証

AppStream 2.0 では、Active Directory ドメインパスワード、または AppStream 2.0 ストリーミングインスタンス (フリースタンスおよび Image Builder) への Windows サインイン用の[共通アクセスカード \(CAC\)](#) や[個人識別検証 \(PIV\)](#) スマートカードなどのスマートカードの使用をサポートしています。ユーザーは、ローカルコンピュータに接続されたスマートカードリーダーとスマートカードを使用して、Microsoft Active Directory ドメインに参加している AppStream 2.0 ストリーミングインスタンスにサインインできます。また、ローカルのスマートカードリーダーとスマートカードを使用して、ストリーミングセッション内のアプリケーションにサインインすることもできます。

ユーザーが Windows で Active Directory に参加しているストリーミングインスタンスにサインインし、ストリーミングアプリケーションのセッション内認証にスマートカードを使用できるようにするには、次のことを行う必要があります。

- 次の要件を満たすイメージを使用します。
 - イメージは、2020 年 12 月 28 日以降 AWS に によって公開されたベースイメージから作成する必要があります。詳細については、「[AppStream 2.0 ベースイメージとマネージド型イメージの更新に関するリリースノート](#)」を参照してください。

- イメージでは、2021 年 1 月 4 日以降にリリースされた AppStream 2.0 エージェントのバージョンを使用する必要があります。詳細については、「[AppStream 2.0 エージェントのリリースノート](#)」を参照してください。
- このセクションで説明するように、ユーザーがストリーミングセッションにアクセスする AppStream 2.0 スタックで Smart card sign in for Active Directory (Active Directory のスマートカードサインイン) を有効にします。

 Note

この設定では、Windows が AppStream 2.0 ストリーミングインスタンス (フリートまたは Image Builder) にサインインするために使用できる認証方法のみを制御します。ユーザーは、ストリーミングインスタンスにサインインした後、セッション内認証に使用できる認証方法を制御しません。

- ユーザーに AppStream 2.0 クライアントバージョン 1.1.257 以降がインストールされていることを確認します。詳細については、「[AppStream 2.0 クライアントリリースノート](#)」を参照してください。

デフォルトでは、Active Directory のパスワードサインインは AppStream 2.0 スタックで有効になっています。AppStream 2.0 コンソールで次の手順を実行すると、Active Directory のスマートカードサインインを有効にできます。

AppStream 2.0 コンソールを使用して Active Directory のスマートカードサインインを有効にするには

1. AppStream 2.0 コンソールを <https://console.aws.amazon.com/appstream2> で開きます。
2. 左のナビゲーションペインで [スタック] を選択します。
3. Active Directory のスマートカード認証を有効にするスタックを選択します。
4. [User Settings (ユーザー設定)] タブを選択し、[Clipboard, file transfer, print to local device, and authentication permissions (クリップボード、ファイル転送、ローカルデバイスへのプリント、および認証のアクセス許可)] セクションを展開します。
5. [Smart card sign in for Active Directory (Active Directory のスマートカードサインイン)] で、[Enabled (有効)] を選択します。

[Password sign in for Active Directory (Active Directory のパスワードサインイン)] が有効になっていない場合は、[パスワードサインイン] を有効にすることもできます。少なくとも 1 つの認証方法を有効にしてください。

6. [更新] を選択します。

または、AppStream 2.0 API、AWS SDK、または AWS Command Line Interface () を使用して、Active Directory のスマートカードサインインを有効にすることもできますAWS CLI。

スマートカードリダイレクト

デフォルトでは、AppStream 2.0 クライアントがインストールされている場合、スマートカードのリダイレクトが有効になっています。この機能を有効にすると、ユーザーは USB リダイレクトなしで AppStream 2.0 ストリーミングセッション中に、ローカルコンピュータとスマートカードに接続されているスマートカードリーダーを使用できます。AppStream 2.0 ストリーミングセッション中、ユーザーのスマートカードリーダーとスマートカードは、ローカルアプリケーションで使用するために引き続きアクセスできます。AppStream 2.0 クライアントは、スマートカード API 呼び出しをユーザーのストリーミングアプリケーションからローカルのスマートカードにリダイレクトします。

Note

スマートカードリダイレクトは現在、Linux ベースのフリーインスタンスまたはマルチセッションフリーインスタンス、または AppStream 2.0 macOS クライアントアプリケーションを使用する場合にはサポートされていません。

Note

スマートカードの動作にミドルウェアソフトウェアが必要な場合は、ユーザーのデバイスと AppStream 2.0 ストリーミングインスタンスの両方にミドルウェアソフトウェアをインストールする必要があります。

管理対象デバイスでのクライアントインストール時に、スマートカードリダイレクトを無効にすることができます。詳細については、「[スマートカードリダイレクトを無効にするかどうかを選択する](#)」を参照してください。スマートカードリダイレクトを無効にすると、ユーザーは USB リダイレクトなしで AppStream 2.0 ストリーミングセッション中にスマートカードリーダーとスマートカードを使用できません。この場合、[デバイスを認定する](#)必要があります。デバイスの認定後、ユーザーは

[AppStream 2.0 でデバイスを共有する](#) 必要があります。スマートカードリダイレクトが無効になっている場合、ユーザーの AppStream 2.0 ストリーミングセッション中に、スマートカードリーダーとスマートカードにアクセスしてローカルアプリケーションで使用することはできません。

ドローイングタブレット

ドローイングタブレット (ペンタブレットとも呼ばれます) は、スタイラス (ペン) を使って絵を描くことができるコンピュータ入力デバイスです。AppStream 2.0 では、ユーザーが Wacom ペンタブレットなどのドローイングタブレットを、ローカルコンピュータに接続してストリーミングアプリケーションで使えるようになりました。

Note

AppStream 2.0 macOS クライアントアプリケーションを使用する場合、ドローイングタブレットはサポートされていません。

次に、ユーザーがストリーミングアプリケーションでドローイングタブレットを使用するための要件と考慮事項を示します。

- 顧客がこの機能を利用できるようにするには、Windows Server 2019 を実行するためのイメージを使用するように AppStream 2.0 フリートを設定する必要があります。
- この機能を使用するには、ユーザーは AppStream 2.0 クライアントを使用するか、Google Chrome または Mozilla Firefox ブラウザでのみ AppStream 2.0 にアクセスする必要があります。
- ストリーミングアプリケーションは、Windows Ink テクノロジーをサポートしている必要があります。詳細については、「[Windows アプリでのペン操作と Windows インク](#)」を参照してください。
- GIMP などの一部のアプリケーションでは、圧力感度をサポートするために、ストリーミングインスタンス上のドローイングタブレットを検出する必要があります。この場合、ユーザーは AppStream 2.0 クライアントを使用して AppStream 2.0 にアクセスし、これらの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。さらに、ユーザーのドローイングタブレットを認定する必要があります。ユーザーは、新しいストリーミングセッションを開始するたびに、ドローイングタブレットを AppStream 2.0 と共有する必要があります。詳細については、「[ストリーミングアプリケーションで使用する USB デバイスを認証する](#)」を参照してください。
- この機能は Chromebook ではサポートされていません。

アプリケーションストリーミングセッション中にドローイングタブレットの使用を開始するには、ユーザーは USB を使用してドローイングタブレットをローカルコンピュータに接続し、圧力感度の

検出に必要な場合は AppStream 2.0 とデバイスを共有します。次に、AppStream 2.0 クライアントまたは[サポートされているウェブブラウザ](#)を使用して、AppStream 2.0 ストリーミングセッションを開始します。

キーボードショートカット

Windows クライアントでは、ほとんどのオペレーティングシステムのキーボードショートカットがサポートされています。サポートされているキーボードショートカットには、Alt + Tab、クリップボードショートカット (Ctrl + X、Ctrl + C、Ctrl + V)、Esc、Alt + F4 などがあります。

macOS クライアントでは、サポートされているキーボードショートカットにはクリップボードショートカット (コマンド + X、コマンド + C、コマンド + V、コマンド + A、コマンド Z) が含まれます。

相対マウスオフセット

デフォルトでは、ユーザーのストリーミングセッション中に、AppStream 2.0 は絶対座標を使用し、マウスの動きをローカルにレンダリングすることによって、マウスの動きに関する情報をストリーミングインスタンスに送信します。コンピュータ支援設計 (CAD)/コンピュータ支援製造 (CAM) ソフトウェアやビデオゲームなど、グラフィックを多用するアプリケーションの場合、相対的マウスモードが有効になっているときにマウスのパフォーマンスが向上します。相対的マウスモードでは、ウィンドウまたは画面内の XY 座標の絶対値ではなく、最終フレームからマウスが移動した距離を表す相対座標が使用されます。相対的マウスモードが有効になっている場合、AppStream 2.0 はマウスの動きをリモートでレンダリングします。

ユーザーは、次のいずれかの操作を行って、AppStream 2.0 ストリーミングセッション中にこの機能を有効にできます。

- Windows クライアントアプリケーションで Ctrl+Shift+F8 を押すか、macOS クライアントアプリケーションで Ctrl+Shift+Fn+F8 を押す
- ストリーミングセッションウィンドウの左上にある AppStream 2.0 ツールバーの [Settings (設定)] メニューから [Relative Mouse Position (相対的マウスの位置)] [Ctrl + Shift + F8]] を選択します。この方法は、クラシックモードまたは [Desktop View (デスクトップビュー)] を使用している場合に機能します。

AppStream 2.0 クライアントをインストールして構成する

ユーザーに AppStream 2.0 クライアントをインストールさせるか、PowerShell スクリプトをリモートで実行して AppStream 2.0 クライアントをインストールできます。

ユーザーがストリーミングセッションでできるようにする USB デバイスを認定する必要があります。その USB デバイスが認定されていない場合、それは AppStream 2.0 によって検出されず、セッションと共有することはできません。

以下のトピックでは、AppStream 2.0 クライアントをインストールして設定する方法について説明します。

目次

- [ユーザーに AppStream 2.0 クライアントをインストールさせる](#)
- [チュートリアル: Amazon AppStream 2.0 クライアントをインストールし、ユーザーのクライアントエクスペリエンスをカスタマイズする](#)
- [AppStream 2.0 エンタープライズデプロイツール、クライアント、USB ドライバーを手動で更新する](#)
- [ストリーミングアプリケーションで使用する USB デバイスを認証する](#)
- [AppStream 2.0 クライアントユーザーの接続方法を構成する](#)
- [ユーザーが AppStream 2.0 ストリーミングセッションで USB デバイスを共有できるようにする](#)
- [ストリーミングセッションをウェブブラウザから AppStream 2.0 クライアントにリダイレクトする](#)
- [AppStream 2.0 ユーザーのファイルシステムリダイレクトを有効にする](#)
- [AppStream 2.0 ユーザーのローカルプリンターリダイレクトを有効にする](#)

ユーザーに AppStream 2.0 クライアントをインストールさせる

AppStream 2.0 クライアントのインストールに役立つようにユーザーに提供できるstep-by-stepのガイダンスについては、[Windows のセットアップ](#)「」または「」を参照してください[the section called “macOS のセットアップとインストール”](#)。

Important

Windows クライアントの場合、ユーザーが .exe ファイルを実行できないようにウイルス対策ソフトウェアをデプロイしている場合は、例外を追加して、ユーザーが AppStream 2.0 クライアントインストール .exe プログラムを実行できるようにする必要があります。それ以外の場合、ユーザーがクライアントをインストールしようとする、何も起こらないか、インストールプログラムの開始後にエラーが発生します。

ユーザーがクライアントをインストールした後、AppStream 2.0 ストリーミングセッション中に USB デバイスを使用することをユーザーに許可する場合は、以下の要件を満たす必要があります。

- AppStream 2.0 で使用できる USB デバイスを認証する必要があります。詳細については、「[ストリーミングアプリケーションで使用する USB デバイスを認証する](#)」を参照してください。
- デバイスが認証された後、ユーザーは新しいストリーミングセッションを開始するたびにデバイスを AppStream 2.0 と共有する必要があります。このタスクの完了に役立つようにユーザーに提供できるガイダンスについては、「[USB デバイス](#)」をご参照ください。

チュートリアル: Amazon AppStream 2.0 クライアントをインストールし、ユーザーのクライアントエクスペリエンスをカスタマイズする

以下のセクションでは、AppStream 2.0 クライアントをインストールし、ユーザーのクライアントエクスペリエンスをカスタマイズする方法について説明します。ユーザーのクライアントをダウンロードしてインストールする場合は、まずエンタープライズデプロイツールをダウンロードします。次に PowerShell スクリプトを実行して、リモートで AppStream 2.0 クライアントをインストールし、クライアントの設定を行います。

Note

AppStream 2.0 macOS クライアントでのエンタープライズデプロイツールの使用はサポートされていません。

内容

- [エンタープライズデプロイツールのダウンロード](#)
- [AppStream 2.0 クライアントと USB ドライバーをインストールする](#)
- [AppStream 2.0 クライアントで AppStream 2.0 にアクセスする](#)
- [AppStream 2.0 クライアントユーザーの StartURL レジストリ値を設定する](#)
- [TrustedDomains レジストリ値を設定して AppStream 2.0 クライアントのその他のドメインを有効にする](#)
- [AS2TrustedDomains DNS TXT レコードを作成して、レジストリを変更せずに AppStream 2.0 クライアントのドメインを有効にします。](#)
- [信頼されたドメインの DNS TXT レコード検索を無効にする](#)
- [自動クライアント更新を無効にするかどうかの選択](#)

- [診断ログのオンデマンドアップロードを無効にするかどうかの選択](#)
- [ネイティブアプリケーションモードを無効にするかどうかを選択する](#)
- [ローカルプリンターリダイレクトを無効にするかどうかを選択する](#)
- [スマートカードリダイレクトを無効にするかどうかを選択する](#)
- [ユーザーの追加の AppStream 2.0 クライアント設定を設定する](#)
- [グループポリシーを使用した AppStream 2.0 クライアントエクスペリエンスのカスタマイズ](#)

エンタープライズデプロイツールのダウンロード

エンタープライズデプロイツールには、AppStream 2.0 クライアントインストールファイルとグループポリシー管理用テンプレートが含まれています。

1. エンタープライズデプロイツールをダウンロードするには、[AppStream 2.0 でサポートされているクライアント](#)ページの右下にある [Enterprise Deployment Tool (エンタープライズデプロイツール)] リンクを選択します。このリンクを選択すると、最新バージョンのツールに必要なファイルを含む .zip ファイルが開きます。
2. 必要なファイルを抽出するには、ツールをダウンロードした場所に移動し、[AmazonAppStreamClient_EnterpriseSetup_<version>] フォルダを右クリックして、[すべて抽出] を選択します。フォルダには、2 つのインストールプログラムとグループポリシー管理用テンプレートが含まれています。
 - AppStream 2.0 クライアントインストーラ (AmazonAppStreamClientSetup_<version>.msi) — AppStream 2.0 クライアントをインストールします。
 - AppStream 2.0 USB ドライバーインストーラ (AmazonAppStreamUsbDriverSetup_<version>.exe) — AppStream 2.0 を介してストリーミングされるアプリケーションで USB デバイスを使用するために必要な AppStream 2.0 USB ドライバーをインストールします。
 - AppStream 2.0 クライアントグループポリシー管理テンプレート (as2_client_config.adm) — グループポリシーを使用して AppStream 2.0 クライアントを設定できます。

AppStream 2.0 クライアントと USB ドライバーをインストールする

AppStream 2.0 クライアントのインストールファイルをダウンロードしたら、ユーザーのコンピュータで次の PowerShell スクリプトを実行して、AppStream 2.0 クライアントのインストールファイル、AppStreamClient.exe、および USB ドライバーをサイレントインストールします。

Note

このスクリプトを実行するには、該当するコンピュータに管理者権限でログインしている必要があります。起動時に [システム] アカウントの下でスクリプトをリモートで実行することもできます。

```
Start-Process msixexec.exe -Wait -ArgumentList '/i  
AmazonAppStreamClientSetup_<version>.msi /quiet'  
  
Start-Process AmazonAppStreamUsbDriverSetup_<version>.exe -Wait -ArgumentList '/quiet'
```

ユーザーのコンピュータにエンタープライズデプロイツールをインストールした後、AppStream 2.0 クライアントは次のようにインストールされます。

1. AppStream 2.0 クライアントのインストールファイルは、ユーザーのコンピュータ上の次のパスにコピーされます: C:\Program Files (x86)\Amazon AppStream 2.0 Client Installer\AppStreamClient.exe。
2. エンタープライズデプロイツールのインストール後にユーザーがコンピュータに初めてログオンすると、AppStream 2.0 クライアントがインストールされます。

Note

エンタープライズデプロイツールは、AppStream 2.0 クライアントフォルダである AppStreamClient が既に %localappdata% に存在することを検出した場合、ツールはクライアントをインストールしません。

ユーザーが AppStream 2.0 クライアントをアンインストールした場合、AppStream 2.0 エンタープライズデプロイツールを更新するまで、クライアントは再度インストールされません。

AppStream 2.0 クライアントで AppStream 2.0 にアクセスする

デフォルトでは、ユーザーは AppStream 2.0 クライアントを起動すると、AppStream 2.0 ドメインを含む URL、または接続を有効にする DNS TXT レコードを含む URL にのみ接続できます。クライアントユーザーが AppStream 2.0 ドメイン以外のドメインにアクセスできるようにするには、以下のいずれかの操作を行います。

- ユーザーからアクセス可能なカスタム URL (組織のログインポータル URL など) を指定するように StartURL レジストリ値を設定します。
- ユーザーからアクセス可能な信頼されたドメインを指定するように TrustedDomains レジストリ値を設定します。
- ユーザーからアクセス可能な信頼されたドメインを指定するための AS2TrustedDomains DNS TXT を作成します。この方法では、レジストリを変更せずに済みます。

Note

AppStream 2.0 クライアントおよび DNS TXT レコード設定により、管理者が指定したドメインまたは URL に、ユーザーが他の接続方法を使用してアクセスできなくなることはありません。たとえば、ユーザーは指定されたドメインまたは URL へのネットワークアクセスが可能であれば、ウェブブラウザを使用してそれらのドメインまたは URL にアクセスできます。

AppStream 2.0 クライアントユーザーの StartURL レジストリ値を設定する

StartUrl レジストリ値を使用して、ユーザーが AppStream 2.0 クライアントを起動したときに自動入力されるカスタム URL を設定できます。クライアントのインストール中にこの HKLM レジストリキーを作成すると、ユーザーがクライアントを起動するときに URL を指定する必要がなくなります。

AppStream 2.0 クライアントをインストールしたら、次の PowerShell スクリプトを実行してこのレジストリキーを作成するか、AppStream 2.0 クライアントのエンタープライズデプロイツールに含まれている管理用テンプレートを使用して、グループポリシーを通じてクライアントを設定できます。

StartUrl の値を ID プロバイダー (IdP) の URL に置き換えます。URL では、デバイスによって信頼されている証明書を使用する必要があります。つまり、StartUrl ウェブページで使用する証明書には、URL のドメイン名が含まれたサブジェクト代替名 (SAN) を含める必要があります。たとえば、StartUrl を <https://appstream.example.com> に設定した場合、SSL 証明書には appstream.example.com が含まれた SAN を含める必要があります。

 Note

このスクリプトを実行するには、該当するコンピュータに管理者権限でログインしている必要があります。起動時に [システム] アカウントの下でスクリプトをリモートで実行することもできます。

```
$registryPath="HKLM:\Software\Amazon\AppStream Client"
New-Item -Path "HKLM:\Software\Amazon" -Name "AppStream Client" -Force

New-ItemProperty -Path $registryPath -Name "StartUrl" -Value "https://www.example.com"
-PropertyType String -Force | Out-Null
```

TrustedDomains レジストリ値を設定して AppStream 2.0 クライアントのその他のドメインを有効にする

指定した信頼されたドメインの URL に接続するように AppStream 2.0 クライアントを設定できます。たとえば、組織ドメインの URL や 1 つ以上の IdP ドメインの URL に接続することをユーザーを許可できます。URL を指定するときは、**.example-idp.com* の形式を使用します。

信頼されたドメインのリストはカンマ区切り形式で指定できます。このリストをレジストリ値として AppStream 2.0 TrustedDomains HKLM レジストリキーに追加します。AppStream 2.0 クライアントをインストールするときや、Microsoft Active Directory を使用している場合は、このレジストリキーを作成し、信頼されたドメインのリストを指定することをお勧めします。これにより、ユーザーは、クライアントがインストールされた直後に、信頼されたドメインの URL に接続できます。

AppStream 2.0 クライアントをインストールしたら、以下の PowerShell スクリプトを実行して、このレジストリキーを作成できます。または、AppStream 2.0 クライアントのエンタープライズデプロイツールに含まれている管理用テンプレートを使用して、グループポリシーを通じてクライアントを設定できます。

TrustedDomains 値を 1 つ以上の組織ドメインや IdP ドメインのカンマ区切りリストに置き換えます。信頼されたドメインのウェブページで使用する証明書には、URL のドメインを含む SAN が含まれている必要があります。たとえば、信頼されたドメインに **.example.com* が含まれていて、ユーザーが *https://appstream.example.com* を指定している場合、SSL 証明書には *appstream.example.com* を含む SAN が必要です。

 Note

このスクリプトを実行するには、該当するコンピュータに管理者権限でログインしている必要があります。起動時に [システム] アカウントの下でスクリプトをリモートで実行することもできます。

```
$registryPath="HKLM:\Software\Amazon\AppStream Client"
New-Item -Path "HKLM:\Software\Amazon" -Name "AppStream Client" -Force

New-ItemProperty -Path $registryPath -Name "TrustedDomains" -Value "*.example1.com,
*.example2.com, aws.amazon.com" -PropertyType String -Force | Out-Null
```

以下に示しているのは、信頼されたドメイン名の形式を設定するための要件と考慮事項です。

- a ~ z、0 ~ 9、-、* の文字がサポートされています。
- DNS は、ドメイン名のどこに表示されるかに応じて、* 文字をワイルドカードまたはアスタリスク文字 (ASCII 42) として扱います。DNS レコードの名前で * をワイルドカードとして使用する場合は制限は以下のとおりです。
 - * は、ドメイン名の左端のラベルを置き換えます。たとえば、「*.example.com」または「*.prod.com」となります。prod.*.example.com のように「*」を他のどのような位置に含めても、DNS はこれをワイルドカードとしてではなく、アスタリスク文字 (ASCII 42) として扱います。
 - * は、ラベル全体を置き換える必要があります。例えば、*prod.example.com や prod*.example.com と指定することはできません。
 - * は、* を含むサブドメインレベル、およびそのサブドメインのすべてのサブドメインに適用されます。たとえば、*.example.com というエントリがある場合、AppStream 2.0 クライアントは zenith.example.com、acme.zenith.example.com、pinnacle.acme.zenith.example.com を許可します。

AS2TrustedDomains DNS TXT レコードを作成して、レジストリを変更せずに AppStream 2.0 クライアントのドメインを有効にします。

ユーザーが組織ドメインの任意の URL (*.example.com など)、または IdP ドメインの任意の URL (*.example-idp.com など) に接続できるようにするには、そのドメインに DNS TXT レコードを作成します。DNS TXT レコードを作成すると、StartURL または TrustedDomains レジストリ値を使用せずに URL への接続をユーザーに許可できます。

信頼されたサブドメインのリストは、プレフィックス `AS2TrustedDomains=` を付けて、カンマ区切り形式で指定できます。その後、該当するドメインの DNS TXT レコードを作成します。AS2TrustedDomains DNS TXT レコードを使用して有効にすることができるドメインは、この DNS TXT レコードを作成したドメイン (またはそのサブドメイン) のみです。DNS TXT レコードを使用して、他のドメインを有効にすることはできません。

DNS レコードの設定の詳細については、「[Enable your organizational domain for the AppStream 2.0 client with a Route 53 DNS TXT record](#)」および「[Creating an AS2TrustedDomains DNS TXT record to redirect the AppStream 2.0 native client to a third-party identity provider](#)」を参照してください。

Note

DNS TXT レコードを作成すると、StartURL または TrustedDomains レジストリ値に含まれていない有効なドメインから、任意のユーザーがストリーミングできます。AppStream 2.0 クライアントおよび DNS TXT レコード設定により、管理者が指定したドメインまたは URL に、ユーザーが他の接続方法を使用してアクセスできなくなることはありません。たとえば、ユーザーは指定されたドメインまたは URL へのネットワークアクセスが可能であれば、ウェブブラウザを使用してそれらのドメインまたは URL にアクセスできます。

DNS TXT レコード設定の例

以下に示しているのは、DNS TXT レコード設定の例です。この設定の例を使用すると、ユーザーは AppStream 2.0 クライアントを起動し、`appstream.example.com` または `appstream-dev.example.com` に接続できます。ただし、`example.com` には接続できません。

- Domains to enable — `appstream.example.com`、`appstream-dev.example.com`
- DNS TXT record location — `example.com`
- DNS TXT record value — `AS2TrustedDomains=appstream.example.com,appstream-dev.example.com`

要件と考慮事項

以下に示しているのは、DNS TXT レコードを作成するための要件と考慮事項です。

- 第 2 レベルドメインで TXT レコードを作成する必要があります。たとえば、ドメインが `prod.appstream.example.com` である場合、`example.com` で DNS TXT レコードを作成する必要があります。

- TXT レコードの値は、AS2TrustedDomains= で始める必要があります。
- a ~ z、0 ~ 9、-、* の文字がサポートされています。
- DNS は、ドメイン名のどこに表示されるかに応じて、* 文字をワイルドカードまたはアスタリスク文字 (ASCII 42) として扱います。DNS レコードの名前で * をワイルドカードとして使用する場合は制限は以下のとおりです。
 - * は、ドメイン名の左端のラベルを置き換えます。たとえば、「*.example.com」または「*.prod.com」となります。prod.*.example.com のように「*」を他のどのような位置に含めても、DNS はこれをワイルドカードとしてではなく、アスタリスク文字 (ASCII 42) として扱います。
 - * は、ラベル全体を置き換える必要があります。例えば、*prod.example.com や prod*.example.com と指定することはできません。
 - * は、* を含むサブドメインレベル、およびそのサブドメインのすべてのサブドメインに適用されます。たとえば、*.example.com というエントリがある場合、AppStream 2.0 クライアントはドメイン zenith.example.com、acme.zenith.example.com、pinnacle.acme.zenith.example.com への接続を許可します。

信頼されたドメインの DNS TXT レコード検索を無効にする

デフォルトでは、ユーザーが AppStream 2.0 を起動し、AppStream 2.0 ドメインではない URL を指定すると、クライアントは DNS TXT レコード検索を実行します。この検索は URL の第 2 レベルドメインで実行されるため、クライアントは AS2TrustedDomains リストにドメインが含まれているかどうかを判断できます。この動作により、ユーザーは StartURL レジストリキーや TrustedDomains レジストリキーに指定されていないドメインに接続したり、AppStream 2.0 ドメインに接続したりできます。

この動作を無効にするには、DnsTxtRecordQueryDisabled レジストリキーの値を true に設定します。このレジストリキーは、AppStream 2.0 クライアントのインストール時に作成できます。これにより、クライアントは、StartURL レジストリキーまたは TrustedDomains レジストリキーに指定されている URL にのみ接続します。

AppStream 2.0 クライアントをインストールしたら、以下の PowerShell スクリプトを実行して、このレジストリキーを作成できます。または、AppStream 2.0 クライアントのエンタープライズデプロイツールに含まれている管理用テンプレートを使用して、グループポリシーを通じてクライアントを設定できます。

Note

このスクリプトを実行するには、該当するコンピュータに管理者権限でログインしている必要があります。起動時に [システム] アカウントの下でスクリプトをリモートで実行することもできます。

```
$registryPath="HKLM:\Software\Amazon\AppStream Client"
New-Item -Path "HKLM:\Software\Amazon" -Name "AppStream Client" -Force

New-ItemProperty -Path $registryPath -Name "DnsTxtRecordQueryDisabled" -Value "true" -
PropertyType String -Force | Out-Null
```

自動クライアント更新を無効にするかどうかの選択

デフォルトでは、新しいバージョンの AppStream 2.0 クライアントが使用可能になると、クライアントは自動的に最新バージョンに更新されます。AutoUpdateDisabled レジストリキーの値を true に設定することで、自動更新を無効にすることができます。このレジストリキーは、AppStream 2.0 クライアントのインストール時に作成できます。こうすることで、新しいバージョンが利用可能になっても、クライアントは自動的に更新されません。

AppStream 2.0 クライアントをインストールしたら、以下の PowerShell スクリプトを実行して、このレジストリキーを作成できます。または、AppStream 2.0 クライアントのエンタープライズデプロイツールに含まれている管理用テンプレートを使用して、グループポリシーを通じてクライアントを設定できます。

Note

このスクリプトを実行するには、該当するコンピュータに管理者権限でログインしている必要があります。起動時に [システム] アカウントの下でスクリプトをリモートで実行することもできます。

```
$registryPath="HKLM:\Software\Amazon\AppStream Client"
New-Item -Path "HKLM:\Software\Amazon" -Name "AppStream Client" -Force

New-ItemProperty -Path $registryPath -Name "AutoUpdateDisabled" -Value "True" -
PropertyType String -Force | Out-Null
```


診断ログのオンデマンドアップロードを無効にするかどうかの選択

デフォルトでは、AppStream 2.0 クライアントは、診断ログとミニダンプをオンデマンドで AppStream 2.0 (AWS) にアップロードすることをユーザーに許可します。また、例外が発生するか、AppStream 2.0 クライアントが応答しなくなった場合、ミニダンプおよび関連ログをアップロードするかどうかを選択するようユーザーに求めます。オンデマンドの診断ログ記録の詳細については、[診断ログの自動およびオンデマンドのアップロード](#) を参照してください。

このような動作を無効にするには、UserUploadOfClientLogsAllowed レジストリキーの値を false に設定します。この HKLM レジストリキーは、AppStream 2.0 クライアントをインストールするときに作成できます。

AppStream 2.0 クライアントをインストールしたら、以下の PowerShell スクリプトを実行して、このレジストリキーを作成できます。または、AppStream 2.0 クライアントのエンタープライズデプロイツールに含まれている管理用テンプレートを使用して、グループポリシーを通じてクライアントを設定できます。

Note

このスクリプトを実行するには、該当するコンピュータに管理者権限でログインしている必要があります。起動時に [システム] アカウントの下でスクリプトをリモートで実行することもできます。

```
$registryPath="HKLM:\Software\Amazon\AppStream Client"
New-Item -Path "HKLM:\Software\Amazon" -Name "AppStream Client" -Force

New-ItemProperty -Path $registryPath -Name "UserUploadOfClientLogsAllowed" -Value
"false" -PropertyType String -Force | Out-Null
```

ネイティブアプリケーションモードを無効にするかどうかを選択する

デフォルトでは、AppStream 2.0 クライアントはクラシックモードまたはネイティブアプリケーションモードで実行できます。NativeAppModeDisabled レジストリキーの値を true に設定することで、ネイティブアプリケーションモードを無効にすることができます。この HKLM レジストリキーは、AppStream 2.0 クライアントをインストールするときに作成できます。その値を true に設定すると、クライアントはクラシックモードでのみ実行されます。ネイティブアプリケーションモードの詳細については、[ネイティブアプリケーションモード](#) を参照してください。

AppStream 2.0 クライアントをインストールしたら、以下の PowerShell スクリプトを実行して、このレジストリキーを作成できます。または、AppStream 2.0 クライアントのエンタープライズデプロイツールに含まれている管理用テンプレートを使用して、グループポリシーを通じてクライアントを設定できます。

Note

このスクリプトを実行するには、該当するコンピュータに管理者権限でログインしている必要があります。起動時に [システム] アカウントの下でスクリプトをリモートで実行することもできます。

```
$registryPath="HKLM:\Software\Amazon\AppStream Client"
New-Item -Path "HKLM:\Software\Amazon" -Name "AppStream Client" -Force

New-ItemProperty -Path $registryPath -Name "NativeAppModeDisabled" -Value "True" -
PropertyType String -Force | Out-Null
```

ローカルプリンターリダイレクトを無効にするかどうかを選択する

デフォルトでは、AppStream 2.0 クライアントを使用すると、ユーザーはプリントジョブをストリーミングアプリケーションからローカルコンピュータに接続されているプリンターにリダイレクトできます。PrinterRedirectionDisabled レジストリキーの値を true に設定することで、ローカルプリンターのリダイレクトを無効にすることができます。この HKLM レジストリキーは、AppStream 2.0 クライアントをインストールするときに作成できます。値が true に設定されている場合、クライアントはプリントジョブをユーザーのストリーミングアプリケーションからローカルコンピュータに接続されているプリンターにリダイレクトしません。

AppStream 2.0 クライアントをインストールしたら、以下の PowerShell スクリプトを実行して、このレジストリキーを作成できます。または、AppStream 2.0 クライアントのエンタープライズデプロイツールに含まれている管理用テンプレートを使用して、グループポリシーを通じてクライアントを設定できます。

Note

このスクリプトを実行するには、該当するコンピュータに管理者権限でログインしている必要があります。起動時に [システム] アカウントの下でスクリプトをリモートで実行することもできます。

```
$registryPath="HKLM:\Software\Amazon\AppStream Client"
New-Item -Path "HKLM:\Software\Amazon" -Name "AppStream Client" -Force

New-ItemProperty -Path $registryPath -Name "PrinterRedirectionDisabled" -Value "True" -
PropertyType String -Force | Out-Null
```

スマートカードリダイレクトを無効にするかどうかを選択する

デフォルトでは、スマートカードのリダイレクトは AppStream 2.0 クライアントに対して有効になっています。この機能を有効にすると、ユーザーは USB リダイレクトなしで AppStream 2.0 ストリーミングセッション中に、ローカルコンピュータとスマートカードに接続されているスマートカードリーダーを使用できます。AppStream 2.0 ストリーミングセッション中、ユーザーのスマートカードリーダーとスマートカードは、ローカルアプリケーションで使用するために引き続きアクセスできます。クライアントは、スマートカード API 呼び出しをユーザーのストリーミングアプリケーションからローカルのスマートカードにリダイレクトします。スマートカードリダイレクトを無効にするには、SmartCardRedirectionDisabled レジストリキーの値を true に設定します。この HKLM レジストリキーは、AppStream 2.0 クライアントをインストールするときに作成できます。

値が true に設定されている場合、ユーザーは USB リダイレクトなしでは AppStream 2.0 ストリーミングセッション中にスマートカードリーダーとスマートカードを使用できません。この場合、[デバイスを認定しない限り](#)、ユーザーはローカルコンピュータに接続されているスマートカードを使用してストリーミングアプリケーションにサインインできません。デバイスの認定後、ユーザーは[AppStream 2.0 でデバイスを共有する](#)必要があります。スマートカードリダイレクトを無効にすると、ユーザーの AppStream 2.0 ストリーミングセッション中に、スマートカードリーダーとスマートカードにアクセスしてローカルアプリケーションで使用することはできません。

AppStream 2.0 クライアントをインストールしたら、以下の PowerShell スクリプトを実行して、このレジストリキーを作成できます。または、AppStream 2.0 クライアントのエンタープライズデプロイツールに含まれている管理用テンプレートを使用して、グループポリシーを通じてクライアントを設定できます。

Note

このスクリプトを実行するには、該当するコンピュータに管理者権限でログインしている必要があります。起動時に [システム] アカウントの下でスクリプトをリモートで実行することもできます。

```
$registryPath="HKLM:\Software\Amazon\AppStream Client"
```

```
New-Item -Path "HKLM:\Software\Amazon" -Name "AppStream Client" -Force
```

```
New-ItemProperty -Path $registryPath -Name "SmartCardRedirectionDisabled" -Value "True"
-PropertyType String -Force | Out-Null
```

ユーザーの追加の AppStream 2.0 クライアント設定を設定する

AppStream 2.0 クライアントは、レジストリキーを使用して、次の追加のクライアント設定を設定します。

- AppStream 2.0 クライアントエンドユーザーライセンス契約 (EULA) の受諾
- AppStream 2.0 クライアント EULA バージョンが受諾されました
- AppStream 2.0 クライアントへの診断ログの自動アップロード
- USB ドライバーを AppStream 2.0 に渡すために使用される USB ドライバーの自動更新
- AppStream 2.0 クライアントでのハードウェアレンダリングの有効化
- AppStream 2.0 クライアントでのファイルシステムリダイレクトのカスタムフォルダパスの設定
- システムのデフォルトブラウザで ID プロバイダー (IdP) の URL を開く

次の表は、ユーザーの AppStream 2.0 クライアントエクスペリエンスをカスタマイズするために使用できる追加のクライアント設定のレジストリ値をまとめたものです。

Note

これらの値は大文字と小文字が区別されます。

値	レジストリパス	タイプ	説明	データ
EULAAccepted	HKCU\Software\Amazon\Appstream Client	文字列	ユーザーに代わって AppStream 2.0 クライアント EULA を受諾するには、この	true/false

値	レジストリパス	タイプ	説明	データ
			値を <code>true</code> に設定します。	
AcceptedEULAVersion	HKCU\Software\Amazon\Appstream Client	文字列	受け入れられた EULA のバージョン。AppStream 2.0 クライアントの最新バージョンの EULA が受諾済みバージョンの EULA と異なる場合、ユーザーは最新バージョンの EULA を受諾するように求められます。	1.0
DiagnosticInfoCollectionAllowed	HKCU\Software\Amazon\Appstream Client	String	AppStream 2.0 が AppStream 2.0 クライアントから AppStream 2.0 (AWS) に診断ログを自動的に送信できるようにするには、この値を <code>true</code> に設定します。	true/false

値	レジストリパス	タイプ	説明	データ
USBDriverOptIn	HKCU\Software\Amazon\Appstream Client	String	USB ドライバーを AppStream 2.0 に渡すために使用する USB ドライバーを AppStream 2.0 で自動的に更新するには、この値を true に設定します。	true/false
HardwareRenderingEnabled	HKCU\Software\Amazon\Appstream Client	文字列	AppStream 2.0 クライアントでハードウェアレンダリングを有効にするには、この値を true に設定します。	true/false

値	レジストリパス	タイプ	説明	データ
FileRedirectionCustomDefaultFolders	HKCU\Software\Amazon\Appstream Client	文字列	この値を設定して、ファイルシステムのリダイレクトに少なくとも1つのフォルダパスを含めます。' 'を使用して複数のフォルダパスを区切ります。デフォルトでは、次のフォルダパスが指定されています。%USERPROFILE%\Desktop %USERPROFILE%\Documents %USERPROFILE%\Downloads	#####

値	レジストリパス	タイプ	説明	データ
OpenIdpUrlInSystemBrowser	HKCU\Software\Amazon\Appstream Client	String	AppStream 2.0 クライアントがシステムのデフォルトブラウザで IdP URL を開くことができるようにするには、この値を true に設定します。この機能は、クライアントバージョン 1.1.1360 以降でサポートされています。	true/false

AppStream 2.0 クライアントをインストールしたら、次の PowerShell スクリプトを実行して、これらのレジストリキーを作成できます。すべてのレジストリキーを作成しない場合は、必要に応じてスクリプトを変更して、必要なレジストリキーのみを作成します。または、AppStream 2.0 クライアントのエンタープライズデプロイツールで提供されている管理用テンプレートを使用して、グループポリシーを通じてクライアントを設定できます。

Note

各ユーザーに対して次のエントリを設定する必要があります。

```
$registryPath="HKCU:\Software\Amazon\AppStream Client"
New-Item -Path "HKCU:\Software\Amazon" -Name "AppStream Client" -Force
New-ItemProperty -Path $registryPath -Name "EULAAccepted" -Value "true" -PropertyType String -Force | Out-Null
```



```
New-ItemProperty -Path $registryPath -Name "AcceptedEULAVersion" -Value "1.0" -
PropertyType String -Force | Out-Null
New-ItemProperty -Path $registryPath -Name "DiagnosticInfoCollectionAllowed" -Value
"true" -PropertyType String -Force | Out-Null
New-ItemProperty -Path $registryPath -Name "USBDriverOptIn" -Value "true" -PropertyType
String -Force | Out-Null
New-ItemProperty -Path $registryPath -Name "HardwareRenderingEnabled" -Value "true" -
PropertyType String -Force | Out-Null
New-ItemProperty -Path $registryPath -Name "FileRedirectionCustomDefaultFolders" -Value
"%USERPROFILE%\Desktop|%USERPROFILE%\Documents|%USERPROFILE%\Downloads" -PropertyType
String -Force | Out-Null
New-ItemProperty -Path $registryPath -Name "OpenIdpUrlInSystemBrowser" -Value "true" -
PropertyType String -Force | Out-Null
```

グループポリシーを使用した AppStream 2.0 クライアントエクスペリエンスのカスタマイズ

AppStream 2.0 クライアントのエンタープライズデプロイツールで提供されている管理用テンプレートを使用して、グループポリシーを通じてクライアントを設定できます。管理用テンプレートをグループポリシー管理コンソールに読み込む方法については、Microsoft サポートドキュメントの「[Recommendations for managing Group Policy administrative template \(.adm\) files](#)」を参照してください。

AppStream 2.0 エンタープライズデプロイツール、クライアント、USB ドライバーを手動で更新する

デフォルトでは、AppStream 2.0 クライアントと USB ドライバーは、新しいクライアントバージョンがリリースされると自動的に更新されます。ただし、エンタープライズデプロイツールを使用してユーザーの AppStream 2.0 クライアントをインストールし、自動更新を無効にした場合は、AppStream 2.0 エンタープライズデプロイツール、クライアント、および USB ドライバーを手動で更新する必要があります。そのためには、以下のステップを実行して、ユーザーのコンピュータで必要な PowerShell コマンドを実行します。

Note

これらのコマンドを実行するには、該当するコンピュータに管理者としてログインしている必要があります。または、起動時に SYSTEM アカウントによりスクリプトをリモートで実行できます。

エンタープライズデプロイツールを使用した AppStream 2.0 macOS クライアントの管理はサポートされていません。

1. AppStream 2.0 エンタープライズデプロイツールをサイレントでアンインストールします。

```
Start-Process msixexec.exe -Wait -ArgumentList '/x  
AmazonAppStreamClientSetup_<existing_version>.msi /quiet'
```

2. AppStream 2.0 USB ドライバーを表示せずにアンインストールします。

```
Start-Process -Wait AmazonAppStreamUsbDriverSetup_<existing_version>.exe -  
ArgumentList '/uninstall /quiet /norestart'
```

3. AppStream 2.0 クライアントを表示せずにアンインストールします。

```
Start-Process "$env:LocalAppData\AppData\AmazonAppStreamClient\Update.exe" -ArgumentList '--  
uninstall'
```

Note

このプロセスでは、AppStream 2.0 クライアントの設定に使用されるレジストリキーも削除されます。AppStream 2.0 クライアントを再インストールしたら、これらのキーを再作成する必要があります。

4. アプリケーションのインストールディレクトリをクリーンアップします。

```
Remove-Item -Path $env:LocalAppData\AppData\AmazonAppStreamClient -Recurse -Confirm:$false -  
Force
```

5. コンピュータを再起動します。

```
Restart-computer
```

6. 最新バージョンの AppStream 2.0 エンタープライズデプロイツールをサイレントインストールします。

```
Start-Process msixexec.exe -Wait -ArgumentList '/i  
AmazonAppStreamClientSetup_<new_version>.msi /quiet'
```

7. 最新バージョンの AppStream 2.0 USB ドライバーをサイレントインストールします。

```
Start-Process AmazonAppStreamUsbDriverSetup_<new_version>.exe -Wait -ArgumentList  
'/quiet'
```

ストリーミングアプリケーションで使用する USB デバイスを認証する

ユーザーがその AppStream 2.0 ストリーミングインスタンスにリダイレクトできる USB デバイスを指定する方法は 2 つあります。

Note

USB リダイレクトは、現在 Windows AppStream 2.0 ストリーミングインスタンスでのみサポートされています。macOS クライアントではサポートされていません。

- イメージに保存された設定ファイル内に USB デバイスフィルター文字列を作成できます。この方法を使用できるのは、常時オンおよびオンデマンドフリートのみです。
- フリートの作成時に、AWS マネジメントコンソールまたは CreateFleet API を使用して USB デバイスフィルター文字列を指定できます。これらの文字列の詳細については、以下のセクションを参照してください。この方法を使用できるのは、Elastic フリートのみです。

AppStream 2.0 イメージで、ユーザーがそのストリーミングアプリケーション向けに利用可能にできる USB デバイスを指定するファイルを作成できます。ユーザーの USB デバイスを認定してストリーミングアプリケーションで使えるようにするには、以下の手順を実行します。

Note

セキュリティ上の理由から、承認済みの信頼できるソースからの USB デバイスのみを認定するようにしてください。すべての汎用デバイスまたはデバイスクラスを認定すると、ストリーミングアプリケーションで未承認のデバイスが使用されることになる可能性があります。

1. まだの場合は、AppStream 2.0 クライアントをインストールしてください。詳細については、[AppStream 2.0 クライアントをインストールして構成する](#) を参照してください。
2. 認証する USB デバイスをコンピュータに接続します。
3. C:\Users\<logged-in-user>\AppData\Local\AppStreamClient に移動し、[dcvusblist.exe] をダブルクリックします。
4. [DCV - USB デバイス] ダイアログボックスに、ローカルコンピュータに接続された USB デバイスのリストが表示されます。[フィルター] 列に、USB デバイスごとにフィルター文字列が表示

- されます。有効にする USB デバイスのリストエントリを右クリックして、[フィルター文字列のコピー] を選択します。
5. デスクトップで、Windows の [スタート] ボタンを選択し、メモ帳を検索します。[メモ帳] をダブルクリックして新しいファイルを開き、フィルター文字列をファイルにコピーして保存します。後で、フィルター文字列を使用して USB デバイスを認証します。
 6. 新しい Image Builder を起動します。詳細については、「[Image Builder を起動し、ストリーミングアプリケーションをインストールして設定する](#)」を参照してください。
 7. Image Builder が [Running (実行中)] 状態になったら、次のステップを実行してストリーミング URL を作成し、AppStream 2.0 クライアントを使用して Image Builder に接続します。
 - a. リストでイメージビルダーを選択した状態で、[アクション]、[ストリーミング URL の作成] の順に選択します。
 - b. [ストリーミング URL の作成] ダイアログボックスで、[リンクのコピー] を選択し、ウェブアドレスをコピーして、後でできるように別のファイルに貼り付けておきます。この URL を使用して、ステップ 12 でイメージビルダーに再接続します。
 - c. [Launch in Client (クライアントで起動)] を選択します。
 - d. [Launch Application] ダイアログボックスが表示され、リンクを開くときに使用するアプリケーションを選択するように求められた場合は、[Amazon AppStream]、[Open link] の順に選択します。次にこのステップを実行してイメージビルダーに接続するときにこのダイアログボックスが表示されないようにするには、[Remember my choice for amazonappstream links (amazonappstream リンクの選択を記憶する)] チェックボックスをオンにします。
 - e. AppStream 2.0 クライアントが AWS カスタマーアグリーメント、AWS サービス条件、AWS プライバシー通知、およびサードパーティー通知へのリンクを表示している場合は、この情報を確認してから、完了を選択します。
 - f. クライアントのサインインページが表示されると、ウェブアドレスフィールドにストリーミング URL があらかじめ入力されています。[接続] を選択します。
 - g. プロンプトが表示されたら、管理者としてイメージビルダーにログインします。
 8. イメージビルダーに接続した後、USB デバイスを使用する前にドライバーをインストールする必要がある場合は、イメージビルダーにドライバーをダウンロードしてインストールしてください。たとえば、3D マウスの接続を使用する場合は、必要なドライバーをダウンロードしてインストールする必要があり、Image Builder に接続します。
 9. Image Builder で、Windows の [スタート] ボタンを選択し、メモ帳を検索します。[メモ帳] を右クリックし、[管理者として実行] を選択します。

10. [ファイル]、[開く] の順に選択し、ファイル C:\ProgramData\Amazon\Photon\DCV\usb_device_allowlist.txt を選択します。また、usb_device_allowlist.txt ファイルでワイルドカード式を使用し、デバイスのカテゴリ全体や、特定メーカーのすべてのデバイスを許可することもできます。
11. ローカルコンピュータから Image Builder にフィルター文字列をコピーします。特定の USB デバイスのフィルター文字列は、[Name (名前)]、[Base Class (基本クラス)]、[SubClass (サブクラス)]、[Protocol (プロトコル)]、[ID Vendor (ベンダー ID)]、[ID Product (製品 ID)]、[Support Autoshare (自動共有のサポート)]、[Skip Reset (リセットのスキップ)] のフィールドをカンマで区切った文字列です。これらの文字列の詳細については、[USB デバイスフィルター文字列の操作](#) を参照してください。
12. Image Builder から切断し、再起動して、AppStream 2.0 クライアントを使用して再接続します。これを行うには、AppStream 2.0 クライアントを開き、ステップ 7 で作成したストリーミング URL をクライアントのサインインウェブアドレスフィールドに貼り付け、[Connect (接続)] を選択します。
13. Image Builder は、USB デバイスをテストして、正常に動作することを確認します。
14. ユーザーは、USB デバイスを AppStream 2.0 セッションで使用する前に、まずデバイスをセッションと共有する必要があります。このタスクの実行に役立つようにユーザーに提供できるガイダンスについては、「[USB デバイス](#)」をご参照ください。
15. USB デバイスが Image Builder と正常に動作する場合は、イメージを作成します。詳細については、「[チュートリアル: AppStream 2.0 コンソールを使用してカスタム AppStream 2.0 イメージを作成する](#)」を参照してください。
16. イメージの作成が完了したら、新しいイメージを使用するように AppStream 2.0 フリートを更新します。

USB デバイスフィルター文字列の操作

このセクションでは、AppStream 2.0 ストリーミングセッションの USB デバイスの認証に使用できるフィルター文字列について説明します。また、これらの文字列を使用するためのガイダンスも提供します。以下のフィルター文字列を使用できます。

- Name — このフィルター文字列の値は、デフォルトではデバイス名になっていますが、独自の値を指定することもできます。
- Base Class, SubClass, Protocol — デバイスの USB クラスコード。詳細については、「[定義済みのクラスコード](#)」を参照してください。

- ID Vendor (VID) — USB 組織によって USB デバイスのメーカーに割り当てられた一意の識別子。
- ID Product (PID) — メーカーによって USB デバイスに割り当てられた一意の識別子。
- Support Autoshare — ストリーミングセッションの開始時にデバイスを自動的に共有することを AppStream 2.0 クライアントに許可します。デバイスの自動共有を許可する場合は、この値を 1 に設定します。デバイスの自動共有を禁止する場合は、この値を 0 に設定します。
- Skip Reset — デフォルトでは、AppStream 2.0 が USB デバイスをストリーミングセッションと共有すると、デバイスが正常に機能するようにリセットされます。ただし、一部の USB デバイスはリセットされた場合、ストリーミングセッション中に正しく機能しません。この問題を防止するには、このフィルター文字列の値を 1 に設定して、ストリーミングセッションと共有している間はデバイスをリセットしないように AppStream 2.0 クライアントに指示します。ストリーミングセッションと共有している間にデバイスをリセットする場合は、この値を 0 に設定します。Skip Reset の値を設定する場合は、Support Autoshare の値を必ず 0 または 1 に設定してください。

ローカルコンピュータからコピーされるフィルター文字列は、USB デバイ스에固有です。必要に応じて、該当する USB デバイスを個別に許可せずに、デバイスのクラス全体を許可することもできます。たとえば、任意の種類の Wacom デザインタブレットや、任意の USB 大容量ストレージデバイスを使用することをユーザーに許可できます。このようなシナリオでは、特定のフィルター文字列フィールドにワイルドカード文字を指定できます。USB デバイスの VID と PID がわからない場合は、この情報を [USB ID データベース](#) で検索できます。

以下の例では、ストリーミングセッション中に USB デバイスを共有するためのフィルター文字列を設定する方法を示しています。

- ストリーミングセッションの開始時にすべての大容量記憶デバイスを許可する — 「Mass storage, 8, *, *, *, *, 1,0」
- ストリーミングセッションの開始時にすべての Wacom デバイスを自動的に許可する — 「Wacom tablets, 3, *, *, 1386, *, 1,0」
- オーディオインターフェイスを提供するすべてのデバイスを許可する — 「Audio, 1, *, *, *, *, 1,0」
- デバイス X を許可するが、デバイスの共有中はリセットしない。ストリーミングセッションの開始時にデバイスを自動的に共有しない — 「"X, Y, *, *, 1386, *, 0,1"」

AppStream 2.0 クライアントユーザーの接続方法を構成する

ユーザーのローカルコンピュータに AppStream 2.0 クライアントをインストールした後、ユーザーはその AppStream 2.0 クライアントを使用してストリーミングセッションに接続できます。組織の要件に応じて、SAML 2.0 を使用した ID フェデレーションの設定、AppStream 2.0 ユーザープールの使用、ストリーミング URL の作成のいずれかを実行して、クライアントユーザーに AppStream 2.0 へのアクセスを提供できます。

目次

- [SAML 2.0](#)
- [AppStream 2.0 ユーザープール](#)
- [ストリーミング URL](#)
- [次のステップ](#)

SAML 2.0

外部 ID プロバイダーを使用してユーザーを AppStream 2.0 スタックにフェデレーションする場合は、クライアントが起動されるたびに事前入力された URL で AppStream 2.0 クライアントを構成するためのレジストリ値を作成する必要があります。URL では、デバイスによって信頼されている証明書を使用する必要があります。証明書には、URL のドメイン名を含むサブジェクト代替名 (SAN) を含める必要があります。

詳細については、以下を参照してください。

- [SAML のセットアップ](#)
- [AppStream 2.0 クライアントユーザーの StartURL レジストリ値を設定する](#)

AppStream 2.0 ユーザープール

AppStream 2.0 ユーザープールで新規ユーザーを作成するか、ユーザープールユーザーを AppStream 2.0 スタックに割り当てると、AppStream 2.0 はユーザーに代わって E メールを送信します。ユーザーは、ようこそメールに記載された URL を入力し、認証情報を入力して、[Connect (接続)] を選択します。

詳細については、「[Amazon AppStream 2.0 ユーザープール](#)」を参照してください。

Note

ユーザープールのユーザーは、AppStream 2.0 macOS クライアントから AppStream 2.0 にアクセスできません。

ストリーミング URL

ストリーミング URL を作成するには、以下のいずれかの方法を使用します。

- AppStream 2.0 コンソール
- [CreateStreamingURL](#) API アクション
- [create-streaming-url](#) AWS CLI コマンド

AppStream 2.0 コンソールを使用してストリーミング URL を作成するには、以下の手順を実行します。

AppStream 2.0 コンソールを使用してストリーミング URL を作成するには

1. AppStream 2.0 コンソールを <https://console.aws.amazon.com/appstream2> で開きます。
2. ナビゲーションペインの [Fleets] を選択します。
3. フリートのリストで、ストリーミング URL を作成するスタックに関連付けられているフリートを選択します。フリートのステータスが [Running (実行中)] であることを確認します。
4. ナビゲーションペインで、[Stacks] を選択します。スタックを選択してから、[Actions (アクション)]、[Create Streaming URL (ストリーミング URL の作成)] を選択します。
5. [User id (ユーザーID)] に、ユーザー ID を入力します。
6. [URL Expiration (URL の有効期限)] で、有効期限を選択し、生成された URL が有効な期間の長さを指定します。この URL は最大 7 日間有効です。
7. [Get URL (URL を取得)] を選択します。
8. URL をコピーし、アクセス可能な場所に保存して、ユーザーに提供します。

AppStream 2.0 クライアントサインインページで、ユーザーは作成されたストリーミング URL をウェブアドレスとして入力し、[Connect (接続)] を選択します。

次のステップ

クライアント接続方法を設定したら、AppStream 2.0 に接続してストリーミングセッションを開始するのに役立つstep-by-stepガイダンスをユーザーに提供できます。 [Windows クライアントで AppStream 2.0 に接続する](#) または [the section called “macOS クライアントで AppStream 2.0 に接続する”](#)。

ユーザーが AppStream 2.0 ストリーミングセッションで USB デバイスを共有できるようにする

ユーザーが USB デバイスを AppStream 2.0 セッションで共有する前に、USB デバイスを認証する必要があります。それ以外の場合、ユーザーがストリーミングセッションを開始すると、その USB デバイスは AppStream 2.0 で検出されず、セッションと共用できません。詳細については、「[ストリーミングアプリケーションで使用する USB デバイスを認証する](#)」を参照してください。

Note

AppStream 2.0 ストリーミングセッションでの USB デバイスの共有は、macOS クライアントではサポートされていません。

ストリーミングセッションをウェブブラウザから AppStream 2.0 クライアントにリダイレクトする

ウェブブラウザから AppStream 2.0 クライアントにストリーミングセッションをリダイレクトするように AppStream 2.0 を構成できます。こうして、ユーザーが AppStream 2.0 にサインインし、ウェブブラウザでストリーミングセッションを開始すると、そのセッションは AppStream 2.0 クライアントにリダイレクトされます。そのためには、以下の手順を実行します。

1. AppStream 2.0 CreateStreamingURL API アクションを使用して、ストリーミング URL を生成します。
2. ストリーミング URL にカスタム AppStream 2.0 クライアントハンドラの次のプレフィックスを追加します: **amazonappstream:**

プレフィックスとストリーミング URL は、一緒に次のようにフォーマットされます。

amazonappstream:*base64encoded(streamingURL)*

 Note

URL をエンコードするときは、エンコードが UTF-8 であることを確認します。

エンコードする Powershell サンプル:

```
[Convert]::ToBase64String([System.Text.Encoding]::UTF8.GetBytes("Streaming  
IdpURL"))
```

3. ユーザーがストリーミング URL にリダイレクトされると、ブラウザは AppStream 2.0 クライアントによってリンクを開く必要があることを検出します。
4. ユーザーは、AppStream 2.0 クライアントを使用してストリーミングセッションを開始するかどうかを選択するように求められます。
5. プロンプトが表示されたら、次のいずれかの状況が発生します。
 - AppStream 2.0 クライアントがインストールされている場合、ユーザーは AppStream 2.0 クライアントを使用してストリーミングセッションを続行することを選択できます。
 - AppStream 2.0 クライアントがインストールされていない場合、ブラウザの動作は次のように異なります。
 - Chrome — メッセージは表示されません。
 - Firefox — Amazon AppStream を開くには新しいアプリが必要であるというメッセージが表示されます。
 - Microsoft Edge — メッセージは表示されません。
 - Internet Explorer — AppStream 2.0 クライアントがインストールされていないことをユーザーに通知するメッセージが表示されます。

この場合、ユーザーは [AppStream Client をダウンロード] リンクを選択してクライアントをダウンロードできます。クライアントをダウンロードした後はインストールできます。ブラウザを最新表示にして、クライアントを使用してストリーミングセッションを開始することができます。

デフォルトのブラウザを使用して Windows デスクトップショートカットを作成する

デフォルトのブラウザを使用して Windows デスクトップショートカットを作成し、クライアントを起動するには、次のサンプル Powershell スクリプトを使用します。

```
$StringToEncode = 'your URL string'
```

```
$encodedUrl =  
[Convert]::ToBase64String([System.Text.Encoding]::UTF8.GetBytes($StringToEncode))  
  
$shortcutContent = "  
[{000214A0-0000-0000-C000-000000000046}]  
Prop3=19,0  
[InternetShortcut]  
IDList=  
URL=amazonappstream:$encodedUrl  
IconIndex=0  
HotKey=0  
IconFile=$env:USERPROFILE\AppData\Local\AppStreamClient\appstreamclient.exe  
"  
  
Set-Content -Path "$env:USERPROFILE\Desktop\AppStream 2.0 Client Launcher.url" -Value  
$shortcutContent
```

AppStream 2.0 ユーザーのファイルシステムリダイレクトを有効にする

AppStream 2.0 ファイルシステムのリダイレクトにより、AppStream 2.0 クライアントをインストールしたユーザーは、ストリーミングセッション内からローカルコンピュータのファイルにアクセスできます。ファイルシステムのリダイレクトを有効にすると、ユーザーがアクセスできるローカルドライブとフォルダのリストを指定できます。ユーザーが AppStream 2.0 にサインインしてストリーミングセッションを開始するときに、アクセスするドライブまたはフォルダをリストから選択できます。その後、ドライブまたはフォルダを AppStream 2.0 と共有できます。ドライブまたはフォルダは、ストリーミングセッション中も引き続きアクセスできます。ユーザーは、いつでもローカルドライブまたはフォルダの共有を停止できます。

Note

ファイルシステムのリダイレクトは現在、Linux ベースのフリートインスタンス、マルチセッションフリートインスタンス、または macOS クライアントを使用する場合にはサポートされていません。

トピック

- [ファイルシステムリダイレクトの前提条件](#)
- [ファイルシステムリダイレクトを有効にする方法](#)

- [デフォルトのドライブとフォルダをユーザーが共有できるようにする](#)
- [ファイルシステムリダイレクトを使用するためのガイダンスを AppStream 2.0 ユーザーに提供する](#)

ファイルシステムリダイレクトの前提条件

AppStream 2.0 のファイルリダイレクトを有効にする方法:

- 2019 年 8 月 8 日以降にリリースされたバージョンの AppStream 2.0 エージェントを使用するイメージを使用する必要があります。詳細については、「[AppStream 2.0 エージェントのリリースノート](#)」を参照してください。
- ユーザーには、AppStream 2.0 クライアントバージョン 1.0.480 以降がインストールされている必要があります。詳細については、「[AppStream 2.0 クライアントリリースノート](#)」を参照してください。
- ファイルのアップロードとダウンロードは、ストリーミングセッション用にユーザーがアクセスするスタックで有効にする必要があります。後述の手順を参照してください。

ファイルシステムリダイレクトを有効にする方法

以下のステップを実行して、ユーザーがストリーミングセッションでアクセスするスタックでファイルのアップロードとダウンロードの両方を有効にします。

1. AppStream 2.0 コンソールを <https://console.aws.amazon.com/appstream2> で開きます。
2. 左のナビゲーションペインで [スタック] を選択します。
3. ファイルシステムリダイレクトを有効にするスタックを選択します。
4. [User Settings (ユーザー設定)] タブを選択し、[Clipboard, file transfer, and local print permissions (クリップボード、ファイル転送、およびローカルプリントのアクセス許可)] セクションを展開します。
5. [File transfer] で [Upload and download] が選択されていることを確認します。そうでない場合は、[編集] を選択し、[Upload and download (アップロードとダウンロード)] を選択します。
6. [更新] を選択します。

デフォルトのドライブとフォルダをユーザーが共有できるようにする

デフォルトでは、スタックのユーザーへのファイルリダイレクトを有効にすると、これらのユーザーは以下のドライブとフォルダをストリーミングセッションで共有できます。

- ドライブ:
 - すべてのローカルハードディスク (C ドライブや D ドライブなどの物理ドライブ)
 - すべての仮想ドライブ (マップされたドライブ文字、Google ドライブ、OneDrive などのネットワークおよび仮想ドライブ)
 - すべてのローカル USB ドライブ
- フォルダ
 - %USERPROFILE%\Desktop
 - %USERPROFILE%\Documents
 - %USERPROFILE%\Downloads

これらのドライブとフォルダのパスは、[Share your local drives and folders (ローカルドライブとフォルダを共有する)] ダイアログボックスに事前入力されています。このダイアログボックスは、ユーザーが AppStream 2.0 にサインインしてストリーミングセッションを開始し、[Setting (設定)]、[Local Resources (ローカルリソース)]、[Local Drives and Folders (ローカルドライブとフォルダ)] の順に選択したときに表示されます。

レジストリを編集することで、デフォルトのドライブおよびフォルダを変更したり、独自に定義したりできます。AppStream 2.0 クライアントのエンタープライズデプロイツールで提供されている管理用テンプレートファイルを使用することもできます。このテンプレートでは、グループポリシーを使用してクライアントを設定できます。詳細については、「[AppStream 2.0 クライアントをインストールして構成する](#)」を参照してください。

ユーザーがストリーミングセッション中に共有ローカルドライブとフォルダにアクセスすると、対応するパスはバックスラッシュでアンダースコアに置き換えられます。また、ローカルコンピュータの名前とドライブ文字のサフィックスが付きます。たとえば、ユーザー名が janedoe で、コンピュータ名が ExampleCorp-123456 のユーザーの場合、デフォルトの [Desktop]、[Documents]、および [Downloads] フォルダパスは次のように表示されます。

C_Users_janedoe_Desktop (\\ExampleCorp-123456) (F:)

C_Users_janedoe_Documents (\\ExampleCorp-123456) (G:)

C_Users_janedoe_Downloads (\\ExampleCorp-123456) (H:)

ファイルシステムリダイレクトを使用するためのガイダンスを AppStream 2.0 ユーザーに提供する

ストリーミングセッション中にファイルリダイレクトを実行する方法をユーザーが理解しやすくするために、[ローカルファイルアクセス](#) の情報を提供することができます。

AppStream 2.0 ユーザーのローカルプリンターリダイレクトを有効にする

ローカルプリンターリダイレクトを使用すると、AppStream 2.0 ユーザーは、ユーザーがマッピングしたネットワークプリンターを含め、各自のローカルコンピュータに接続されているプリンターに、ストリーミングアプリケーションからプリントジョブをリダイレクトできます。ユーザーがストリーミングセッション中にドキュメントを印刷できるようにするために、AppStream 2.0 ストリーミングインスタンスにプリンタードライバをインストールする必要はありません。

Note

ローカルプリンターリダイレクトの有効化は、現在 Linux ベースのスタックではサポートされていません。

トピック

- [ローカルプリンターリダイレクトの前提条件](#)
- [ローカルプリンターへのリダイレクトを有効にするには](#)
- [ローカルプリンターへのリダイレクトを無効にするには](#)

ローカルプリンターリダイレクトの前提条件

ユーザーがローカルプリンターリダイレクトを使用できるようにするには、次の操作を行う必要があります。

- 2020 年 7 月 30 日以降にリリースされたバージョンの AppStream 2.0 エージェントを使用するイメージを使用してください。詳細については、「[AppStream 2.0 エージェントのリリースノート](#)」を参照してください。
- ユーザーに AppStream 2.0 クライアントバージョン 1.1.179 以降がインストールされていることを確認します。詳細については、「[AppStream 2.0 クライアントリリースノート](#)」を参照してください。
- ユーザーがストリーミングセッションでアクセスするスタックで、プリンターリダイレクトが有効になっていることを確認します。

ローカルプリンターへのリダイレクトを有効にするには

デフォルトでは、AppStream 2.0 クライアントがインストールされると、ローカルプリンターのリダイレクトが有効になります。ただし、ユーザーがストリーミングセッションでアクセスするスタック

でローカルプリンターリダイレクトが有効になっていない場合は、次の手順を実行して AppStream 2.0 コンソールでローカルプリンターリダイレクトを有効にできます。

AppStream 2.0 コンソールを使用してローカルプリンターのリダイレクトを有効にするには

1. AppStream 2.0 コンソールを <https://console.aws.amazon.com/appstream2> で開きます。
2. 左のナビゲーションペインで [スタック] を選択します。
3. ローカルプリンターリダイレクトを有効にするスタックを選択します。
4. [User Settings (ユーザー設定)] タブを選択し、[Clipboard, file transfer, print to local device, and authentication permissions (クリップボード、ファイル転送、ローカルデバイスへのプリント、および認証のアクセス許可)] セクションを展開します。
5. [Print to local device (ローカルデバイスへプリント)] で、[Enabled (有効)] が選択されていることを確認します。そうでない場合は、[Edit (編集)] を選択し、[Enabled (有効)] を選択します。
6. [更新] を選択します。

または、AppStream 2.0 API、AWS SDK、または AWS Command Line Interface () を使用して、ローカルプリンターリダイレクトを有効にすることもできますAWS CLI。

ローカルプリンターへのリダイレクトを無効にするには

ローカルプリンターへのリダイレクトを無効にするには以下のステップに従います。

ローカルプリンターへのリダイレクトを無効にするには

次のいずれかの方法で、ローカルプリンターリダイレクトを無効にすることができます。

- マネージド型デバイスへのクライアントのインストール中です。詳細については、「[ローカルプリンターリダイレクトを無効にするかどうかを選択する](#)」を参照してください。
- AppStream 2.0 コンソールを使用して、AppStream 2.0 スタックでこのオプションを無効にします。
- AppStream 2.0 API、AWS SDK、または AWS Command Line Interface (AWS CLI) を使用して、AppStream 2.0 スタックでこのオプションを無効にします。

Amazon AppStream 2.0 リソースにタグを付ける

AWS を使用すると、タグ形式で AWS リソースにメタデータを割り当てることができます。これらのタグを使用して、AppStream 2.0 Image Builder、イメージ、フリート、スタックなどの管理に役立てることができます。また、請求データなどのデータの整理することもできます。

次のようにできます：

- リソースを (目的、所有者、環境など) さまざまな方法で論理的にグループ化することができます。

これは、同じ種類のリソースが多い場合に役立ちます。

- リソースに割り当てたタグに基づいて特定のリソースをすばやく特定します。
- AWS コストの特定と管理

たとえば、異なる環境 (開発環境や実稼働環境など) にある AppStream 2.0 フリートや、異なるビジネス部門 (人事やマーケティングなど) に割り当てられた AppStream 2.0 フリートを確認してグループ化できます。その後、これらのフリートの関連 AWS コストを詳細レベルで追跡できます。そのためには、サインアップして、タグキー値が含まれたアマゾン ウェブ サービスアカウントの請求書を取得する必要があります。タグによるコスト配分レポートの設定の詳細については、AWS Billing ユーザーガイドの「[コスト配分月次レポート](#)」を参照してください。

内容

- [Amazon AppStream 2.0 のタグ付けの基本](#)
- [Amazon AppStream 2.0 のタグ制限](#)
- [Amazon AppStream 2.0 コンソールでのリソース作成中のタグの追加](#)
- [Amazon AppStream 2.0 コンソールでの既存のリソースのタグの追加、編集、削除](#)
- [Amazon AppStream 2.0 API、AWS SDK、または CLI AWS を使用したタグの操作](#)

Amazon AppStream 2.0 のタグ付けの基本

タグは、他の AWS サービスタグと同様に、キーと値のペアで構成されます。リソースにタグを付けるには、各タグのキーと値を指定します。キーは、「プロジェクト」、「所有者」、「環境」など一般的なカテゴリとすることができ、特定の値を指定できます。また、複数のリソース間で同じキーと値を共有できます。作成後すぐに AppStream 2.0 リソースにタグを付けるか、後で付けることがで

きます。リソースを削除すると、そのリソースからタグが削除されます。ただし、同じタグキーを持つ他の AppStream 2.0 および AWS リソースは影響を受けません。

タグのキーと値は編集でき、タグはリソースからいつでも削除できます。タグの値を空の文字列に設定することはできますが、タグの名前を null に設定することはできません。特定のリソースについて既存のタグと同じキーを持つタグを追加した場合、以前の値は新しい値によって上書きされます。リソースを削除すると、リソースのタグも削除されます。

Note

AppStream 2.0 リソースの AWS コストを追跡するために毎月のコスト配分レポートを設定する場合は、既存の AppStream 2.0 リソースに追加されたタグが、その月に更新されるリソースの翌月 1 日にコスト配分レポートに表示されることに注意してください。

Amazon AppStream 2.0 のタグ制限

- AppStream 2.0 リソースあたりのタグの最大数は 50 です。
- キーの最大長は UTF-8 で 128 Unicode 文字です。
- 値の最大長は UTF-8 で 256 Unicode 文字です。
- タグのキーと値は大文字と小文字が区別されます。
- タグ名または値に「aws:」プレフィックスを使用しないでください。これは、AWS 用に予約されたシステムタグです。このプレフィックスが含まれるタグの名前や値は編集または削除できません。このプレフィックスを持つタグは、リソースあたりのタグ数の制限には計算されません。
- 通常使用できる文字は、UTF-8 で表現できる文字、数字、およびスペースと、特殊文字 +、-、=、:、_、.、/、@ です。
- 同じキーと値を複数のリソースで共有できますが、同じリソースで重複したキーを持つことはできません。
- リソースの作成中にリソースのタグを追加できます。作成済みのリソースのタグを追加、編集、削除することもできます。

Amazon AppStream 2.0 コンソールでのリソース作成中のタグの追加

AppStream 2.0 コンソールでリソースを作成するときに、リソースの管理用に 1 つ以上のタグを追加できます。詳細については、以下の各トピックを参照してください。

- Image builders — [Image Builder を起動し、ストリーミングアプリケーションをインストールして設定する](#)、ステップ 4
- イメージ — [ステップ 6: イメージの作成を完了する](#)、ステップ 1
- フリート — [Amazon AppStream 2.0 でフリートを作成する](#)、ステップ 3
- スタック — [Amazon AppStream 2.0 でスタックを作成する](#)、ステップ 2

Amazon AppStream 2.0 コンソールでの既存のリソースのタグの追加、編集、削除

AppStream 2.0 コンソールを使用して、既存のリソースのタグを追加、編集、削除できます。

既存の AppStream 2.0 リソースのタグを追加、編集、または削除するには

1. AppStream 2.0 コンソール (<https://console.aws.amazon.com/appstream2>) を開きます。
2. ナビゲーションバーから、タグを追加、編集、または削除するリソースを含むリージョンを選択します。
3. ナビゲーションペインで、リソースタイプを選択します。リソースタイプは、Image Builder、イメージ、フリート、またはスタックとすることができます。
4. リソースリストからリソースを選択します。
5. [Tags]、[Add/Edit Tags] を選択し、次のうち 1 つ以上を実行します。
 - タグを追加するには、[Add Tag (タグを追加)] を選択し、各タグのキーと値を入力します。
 - タグを編集するには、必要に応じて、タグのキーと値を変更します。
 - タグを削除するには、タグの Delete (削除) アイコン (X) を選択します。
6. [Save] を選択します。

Amazon AppStream 2.0 API、AWS SDK、または CLI AWS を使用したタグの操作

AppStream 2.0 API、AWS SDK、または AWS コマンドラインインターフェイス (AWS CLI) を使用している場合は、tagsパラメータで次の AppStream 2.0 オペレーションを使用して、新しいリソースを作成するときにタグを追加できます。

Note

タグのキーと値にはスペースを使用できます。AWS CLI を使用するときにはスペースを指定するには、「\s」を使用します (引用符なし)。

タスク	AWS CLI	API 操作
新しいフリートに 1 つ以上のタグを追加する	create-fleet	CreateFleet
新しいイメージビルダーに 1 つ以上のタグを追加する	create-imagebuilder	CreateImageBuilder
新しいスタックに 1 つ以上のタグを追加する	create-stack	CreateStack

以下の AppStream 2.0 オペレーションを使用して、既存のリソースのタグを追加、編集、削除、または一覧表示できます。

タスク	AWS CLI	API 操作
リソースの 1 つ以上のタグを追加または上書きする	タグリソース	TagResource
リソースの 1 つ以上のタグを削除する	タグなしリソース	UntagResource
リソースの 1 つ以上のタグを一覧表示する	list-tags-for-resource	ListTagsForResource

AppStream 2.0 API、AWS SDK、または CLI AWS アクションを使用して既存の AppStream 2.0 リソースのタグを追加、編集、削除、または一覧表示する場合は、Amazon リソースネーム (ARN) を使用してリソースを指定します。ARN は AWS リソースを一意に識別し、次の一般的な構文を使用します。

```
arn:aws:appstream:region:account:resourceType/resourceName
```

region

リソースが作成された AWS リージョン (例: us-east-1)。

account

ハイフンなしの AWS アカウント ID (例: 123456789012)。

resourceType

リソースのタイプ。AppStream 2.0 リソースのタイプとして、image-builder、image、fleet、および stack にタグを付けることができます。

resourceName

リソースの名前。

例えば、CLI [describe-fleets](#) コマンドを使用して、AppStream 2.0 フリーの ARN AWS を取得できます。次のコマンドをコピーします。

```
aws appstream describe-fleets
```

TestFleet という名前の単一のフリーを含む環境の場合、このリソースの ARN は、次のような JSON 出力に表示されます。

```
"Arn": "arn:aws:appstream:us-east-1:123456789012:fleet/TestFleet"
```

このリソースの ARN を取得した後、[tag-resource](#) コマンドを使用して 2 つのタグを追加できます。

```
aws appstream tag-resource --resource arn:awsappstream:us-east-1:123456789012:fleet/TestFleet --tags Environment=Test,Department=IT
```

最初のタグ Environment=Test は、フリーがテスト環境であることを示します。2 番目のタグ Department=IT は、フリーが IT 部門に属していることを示します。

次のコマンドを使用して、フリートに追加した 2 つのタグをリストできます。

```
aws appstream list-tags-for-resource --resource arn:aws:appstream:us-east-1:123456789012:fleet/TestFleet
```

この例の JSON 出力は次のとおりです。

```
{
  "Tags": {
    "Environment" : "Test",
    "Department" : "IT"
  }
}
```

モニタリングとレポート

モニタリングとレポートは、Amazon AppStream 2.0 ストリーミングインスタンスの信頼性、可用性、パフォーマンスを維持し、ユーザーに応答性の高いストリーミングエクスペリエンスを提供するための重要な部分です。

内容

- [Amazon AppStream 2.0 リソースのモニターリング](#)
- [AppStream 2.0 使用状況レポート](#)
- [を使用した AppStream 2.0 API コールのログ記録 AWS CloudTrail](#)

詳細については次を参照してください:

- [AppStream 2.0 でカスタムロギングと CloudWatch アラートを作成する](#)
- [AWS Health Dashboard の開始方法 – アカウントのヘルス](#)
- [EventBridge による AWS ヘルスイベントのモニタリング](#)

Amazon AppStream 2.0 リソースのモニターリング

AppStream 2.0 は Amazon CloudWatch にメトリクスを公開し、詳細な追跡と深層ダイブの分析を有効にします。これらの統計情報は、一定期間記録されるため、履歴情報にアクセスしてフリートのパフォーマンスをよりの確に把握できます。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。

内容

- [コンソールを使用してフリートの使用状況を表示する](#)
- [コンソールを使用してインスタンスおよびセッションのパフォーマンスメトリクスを表示する](#)
- [AppStream 2.0 のメトリクスとディメンション](#)

コンソールを使用してフリートの使用状況を表示する

Amazon AppStream 2.0 フリートの使用状況は、AppStream 2.0 コンソールまたは CloudWatch コンソールでモニタリングできます。

AppStream 2.0 コンソールでフリートの使用状況を表示するには

1. AppStream 2.0 コンソール (<https://console.aws.amazon.com/appstream2>) を開きます。
2. 左のペインの [Fleets] を選択します。
3. フリートを選択して、[Fleet Usage] タブを選択します。
4. デフォルトでは、以下のメトリクスがグラフに表示されます。

- シングルセッションフリートの

ActualCapacity、InUseCapacity、DesiredCapacity、AvailableCapacity、PendingCapacity

- マルチセッションフリートの

ActualUserSessionCapacity、ActiveUserSessionCapacity、AvailableUserSessionCapacity

CloudWatch コンソールでフリートの使用状況を表示するには

1. CloudWatch コンソール (<https://console.aws.amazon.com/cloudwatch/>) を開きます。
2. 左のペインで [メトリクス] を選択します。
3. [AppStream] 名前空間を選択し、[Fleet Metrics] を選択します。
4. グラフ化するメトリクスを選択します。

コンソールを使用してインスタンスおよびセッションのパフォーマンスメトリクスを表示する

Amazon AppStream 2.0 フリートのインスタンスおよびセッションのパフォーマンスは、AppStream 2.0 コンソールまたは CloudWatch コンソールを使用してモニタリングできます。

パフォーマンスメトリクスは 5 分間隔で収集されます。新しいセッションがプロビジョニングされると、5 分後に最初のメトリクスデータポイントが表示されます。以降のメトリクスデータポイントは 5 分間隔で利用可能になります。

Note

現在、パフォーマンスメトリクスはマルチセッションフリートでのみ利用可能です。

AppStream 2.0 コンソールでインスタンスとセッションを表示するには

1. AppStream 2.0 コンソール (<https://console.aws.amazon.com/appstream2>) を開きます。

2. 左のペインの [Fleets] を選択します。
3. フリートを選択し、[詳細を表示] と [セッションを表示] を選択します。
4. セッションを選択してメトリクスを表示します。
5. デフォルトでは、以下のメトリクスがグラフに表示されます。

- インスタンスメトリクス
 - CpuUtilizationInstance
 - MemoryUtilizationInstance
 - PagingFileUtilizationInstance
 - DiskUtilizationInstance
- セッションメトリクス
 - CpuUtilizationSession
 - MemoryUtilizationSession

CloudWatch コンソールでインスタンスとセッションのパフォーマンスを表示するには

1. CloudWatch コンソール (<https://console.aws.amazon.com/cloudwatch/>) を開きます。
2. 左のペインで [メトリクス] を選択します。
3. [AppStream] 名前空間を選択し、[フリートインスタンスのメトリクス] または [フリートセッションのメトリクス] を選択します。
4. グラフ化するメトリクスを選択します。

AppStream 2.0 のメトリクスとディメンション

Amazon AppStream 2.0 は、次のメトリクスとディメンション情報を Amazon CloudWatch に送信します。

以下の `InsufficientConcurrencyLimitError` 以外のメトリクスはすべて、常時オンデマンドフリートに適用されます。Elastic フリートに適用される唯一のメトリクスは `InUseCapacity` と `InsufficientCapacityError` です。

AppStream 2.0 は、メトリクスを CloudWatch に毎分 1 回送信します。AWS/AppStream 名前空間には、次のメトリクスが含まれます。

トピック

- [シングルセッションフリートのフリート使用状況メトリクス](#)
- [マルチセッションフリートのフリート使用状況メトリクス](#)
- [マルチセッションフリートのインスタンスとセッションのパフォーマンスメトリクス](#)
- [Amazon AppStream 2.0 メトリクスのディメンション](#)

シングルセッションフリートのフリート使用状況メトリクス

以下はシングルセッションフリートのフリート使用状況メトリクスです。

メトリクス	説明
ActualCapacity	ストリーミングに使用可能であるか、現在ストリーミング中のインスタンスの合計数。 ActualCapacity = AvailableCapacity + InUseCapacity 単位: カウント 有効な統計: Average、Minimum、Maximum
AvailableCapacity	現在、ユーザーセッションに使用可能なアイドル状態のインスタンスの数。 AvailableCapacity = ActualCapacity - InUseCapacity 単位: カウント 有効な統計: Average、Minimum、Maximum
CapacityUtilization	次の数式を使用した、フリートで使用中のインスタンスの割合 (%)。 CapacityUtilization = (InUseCapacity/ActualCapacity) * 100 このメトリクスをモニタリングすると、フリートの目的の容量の値を増減する決定に役立ちます。 単位: パーセント

メトリクス	説明
	有効な統計: Average、Minimum、Maximum
DesiredCapacity	実行中または保留中のインスタンスの合計数。これはフリートが一定の状態でサポートできる同時ストリーミングセッションの合計数を表します。 $\text{DesiredCapacity} = \text{ActualCapacity} + \text{PendingCapacity}$ 単位: カウント 有効な統計: Average、Minimum、Maximum
InUseCapacity	ストリーミングセッションに現在使用中のインスタンスの数。InUseCapacity の数が 1 の場合、1 つのストリーミングセッションを表します。 単位: カウント 有効な統計: Average、Minimum、Maximum
PendingCapacity	AppStream 2.0 によってプロビジョニング中のインスタンスの数。プロビジョニングの完了後にフリートがサポートできるストリーミングセッションの追加の数を表します。プロビジョニングが開始されると、通常はインスタンスがストリーミングに使用可能になるまでに、10～20 分かかります。 単位: カウント 有効な統計: Average、Minimum、Maximum

メトリクス	説明
RunningCapacity	現在実行中のインスタンスの合計数。現在の状態のフリートでサポートできる同時ストリーミングセッションの数を表します。 このメトリクスは、常時オンのフリートのみに提供され、ActualCapacity メトリクスと同じ値になります。 単位: カウント 有効な統計: Average、Minimum、Maximum
InsufficientCapacityError	容量不足により拒否されたセッションリクエストの数。 このメトリクスを使用して、ストリーミングセッションを待機中のユーザーを通知するようアラームを設定できます。 単位: カウント 有効な統計: Average、Minimum、Maximum、Sum
InsufficientConcurrencyLimitError	最大同時ストリーミング容量に達したために拒否された Elastic フリートセッションリクエストの数。 このメトリクスを使用して、ストリーミングセッションを待機中のユーザーを通知するようアラームを設定できます。 単位: カウント 有効な統計: Average、Minimum、Maximum、Sum

マルチセッションフリートのフリート使用状況メトリクス

以下はマルチセッションフリートのフリート使用状況メトリクスです。

メトリクス	説明
CapacityUtilization	次の数式を使用した、フリートで使用中のセッションの割合 (%)。

メトリクス	説明
	$\text{CapacityUtilization} = (\text{ActiveUserSessionCapacity} / \text{ActualUserSessionCapacity}) * 100$ このメトリクスをモニタリングすると、フリートの目的の容量の値を増減する決定に役立ちます。 単位: パーセント 有効な統計: Average、Minimum、Maximum
ActualUserSessionCapacity	ストリーミングに使用可能であるか、現在ストリーミング中であるセッションスロットの合計数。 $\text{ActualUserSessionCapacity} = \text{AvailableUserSessionCapacity} + \text{ActiveUserSessionCapacity}$ 単位: カウント 有効な統計: Average、Minimum、Maximum
AvailableUserSessionCapacity	現在、ユーザーセッションに使用可能なアイドル状態のセッションスロットの数。 $\text{AvailableUserSessionCapacity} = \text{ActualUserSessionCapacity} - \text{ActiveUserSessions}$ 単位: カウント 有効な統計: Average、Minimum、Maximum

メトリクス	説明
DesiredUserSessionCapacity	実行中または保留中のセッションスロットの合計数。これはフリートが一定の状態でサポートできる同時ストリーミングセッションの合計数を表します。 $\text{DesiredUserSessionCapacity} = \text{ActualUserSessionCapacity} + \text{PendingUserSessionCapacity}$ 単位: カウント 有効な統計: Average、Minimum、Maximum
ActiveUserSessionCapacity	ストリーミングセッションに現在使用中のユーザーセッションの数。 単位: カウント 有効な統計: Average、Minimum、Maximum
PendingUserSessionCapacity	AppStream 2.0 によってプロビジョニング中のセッションスロットの数。プロビジョニングの完了後にフリートがサポートできるストリーミングセッションの追加の数を表します。プロビジョニングが開始されると、通常はインスタンスがストリーミングに使用可能になるまでに、10 ~ 20 分かかります。 単位: カウント 有効な統計: Average、Minimum、Maximum
RunningUserSessionCapacity	ストリーミングに使用可能であるか、現在ストリーミング中であるセッションスロットの合計数。現在の状態のフリートでサポートできる同時ストリーミングセッションの数を表します。 このメトリクスは、常時オンのフリートのみに提供され、ActualUserSessionCapacity メトリクスと同じ値になります。 単位: カウント 有効な統計: Average、Minimum、Maximum

マルチセッションフリートのインスタンスとセッションのパフォーマンスメトリクス

以下はマルチセッションフリートのインスタンスとセッションのパフォーマンスメトリクスです。

メトリクス	説明
CpuUtilizationInstance	割り当てられたコンピューティングユニットのうち、現在インスタンスで使用中であるものの割合。 単位: パーセント
MemoryUtilizationInstance	割り当てられた物理メモリユニットのうち、現在インスタンスで使用中であるものの割合。 単位: パーセント
PagingFileUtilizationInstance	メモリ (RAM) 容量を拡張するために現在使用中であるページングファイルの割合。 単位: パーセント
DiskUtilizationInstance	インスタンスでプログラムの実行とタスクの実行に現在使用中であるディスクユニットの割合。 単位: パーセント
CpuUtilizationSession	割り当てられたコンピューティングユニットのうち、セッションで現在使用中であるものの割合。 単位: パーセント
MemoryUtilizationSession	割り当てられた物理メモリユニットのうち、セッションで現在使用中であるものの割合。 単位: パーセント

Amazon AppStream 2.0 メトリクスのディメンション

Amazon AppStream 2.0 が提供するメトリクスをフィルタリングするには、以下のディメンションを使用します。

メトリクスタイプ	ディメンション	説明	メトリクス
フリートメトリクス	Fleet	フリートの名前。	フリート容量メトリクス
フリートインスタンスのメトリクス	Fleet Name	フリートの名前。	フリートインスタンスのパフォーマンスメトリクス
フリートインスタンスのメトリクス	Instance Id	インスタンス識別子。	フリートインスタンスのパフォーマンスメトリクス
フリートセッションのメトリクス	Fleet Name	フリートの名前。	フリートセッションのパフォーマンスメトリクス
フリートセッションのメトリクス	Instance Id	インスタンス識別子。	フリートセッションのパフォーマンスメトリクス
フリートセッションのメトリクス	Session Id	セッション識別子。	フリートセッションのパフォーマンスメトリクス

AppStream 2.0 使用状況レポート

Amazon AppStream 2.0 使用状況レポートをサブスクライブして、ユーザーがサービスをどのように使用しているかについて詳細なレポートを受け取ることができます。2 つの .csv ファイルは、毎日アカウントの Amazon Simple Storage Service (Amazon S3) バケットにエクスポートされます。

Note

AppStream 2.0 使用状況レポートを有効にするには、使用するイメージで、2019 年 5 月 7 日以降にリリースされたバージョンの AppStream 2.0 エージェントが実行されている必要があります。

目次

- [AppStream 2.0 の使用状況レポートを有効にする](#)
- [AppStream 2.0 使用状況レポートのフィールド](#)
- [カスタムレポートを作成して AppStream 2.0 使用状況データを分析する](#)

AppStream 2.0 の使用状況レポートを有効にする

使用状況レポートを受け取るには、AppStream 2.0 コンソール、AWS Command Line Interface (AWS CLI)、または CreateUsageReportSubscription API オペレーションを使用して、それらのレポートをサブスクライブします。使用状況データを受信する AWS リージョンごとに、使用状況レポートを個別に有効にする必要があります。

Note

使用状況レポートのサブスクライブはいつでも開始または停止できます。使用状況レポートのサブスクライブは無料ですが、S3 バケットに保存されたレポートには標準の Amazon S3 料金が適用されます。詳細については、[Amazon S3 の料金](#) を参照してください。

AppStream 2.0 コンソールを使用して AppStream 2.0 の使用状況レポートをサブスクライブするには、次の手順を実行します。

1. AppStream 2.0 コンソールを <https://console.aws.amazon.com/appstream2> で開きます。
2. 使用状況レポートを有効にする AWS リージョンを選択します。

3. ナビゲーションペインで [Usage Reports (使用状況レポート)] を選択します。
4. [有効] を選択し、[適用] を選択してください。

インスタンスセッションスクリプトを有効にし、セッションスクリプト設定に対して Amazon S3 ログを有効にした場合、AppStream 2.0 によってスクリプト出力の保存用に S3 バケットが作成されています。そのバケットはアカウントとリージョンに固有のものです。この場合に使用状況レポートを有効にすると、AppStream 2.0 によって使用状況レポートの保存にその同じバケットが使用されます。インスタンスセッションスクリプトをまだ有効にしていない場合、使用状況レポートを有効にすると、AppStream 2.0 によって以下の場所に新しい S3 バケットが作成されます。

```
appstream-logs-region-code-account-id-without-hyphens-random-identifier
```

region-code

使用状況レポートが有効になっている AWS リージョンのリージョンコード。

account-id-without-hyphens

ご自身の Amazon Web Services アカウント ID ランダムな ID により、同じリージョンで他のバケットとの競合が発生することはありません。バケット名の最初の部分 appstream-logs は、複数のアカウントやリージョンにまたがる場合でも変更されません。

たとえば、米国西部 (オレゴン) リージョン (us-west-2)、アカウント番号 123456789012 で、使用状況レポートを有効にした場合は、AppStream 2.0 によって、以下の例に示しているような名前で、そのリージョンのアカウント内に Amazon S3 バケットが作成されます。

```
appstream-logs-us-west-2-1234567890123-abcdefg
```

適切なアクセス許可を持つ管理者のみが、このバケットを削除できます。

トピック

- [AppStream 2.0 セッションレポート](#)
- [AppStream 2.0 アプリケーションレポート](#)

AppStream 2.0 セッションレポート

お客様の Amazon Web Services アカウントで、ユーザーが 1 つ以上のストリーミングセッションを開始した日ごとに、AppStream 2.0 によってセッションレポートが Amazon S3 バケットにエクス

ポートされます。daily-session-report-[YYYY]-[MM]-[DD].csv という名前のレポートが、以下のフォルダパスを使用して、Amazon S3 アカウント内のネストフォルダ構造に保存されます。

```
[bucket_name]/sessions/schedule=DAILY/year=[YYYY]/month=[MM]/day=[DD]/
```

Amazon Athena を使用してレポートをクエリする場合、このネスト構造により、パーティション分割が容易になります。Athena はサーバーレスのインタラクティブなクエリサービスであり、このサービスを使用すると、S3 バケットに保存されているデータを標準の SQL により分析できます。詳細については、「[カスタムレポートを作成して AppStream 2.0 使用状況データを分析する](#)」を参照してください。

各ユーザーセッションはセッションレポートの 1 つのレコードに記述されます。セッションレポートは毎日、レポート対象日の終了 (UTC 時間) から 24 時間以内に生成されます。セッションが複数日にまたがる場合、セッションレコードは、セッションが終了した日に対応するセッションレポートに含まれます。セッションレポートに含まれるデータについては、「[セッションレポートのフィールド](#)」を参照してください。

AppStream 2.0 アプリケーションレポート

ユーザーがストリーミングセッション中に 1 つ以上のアプリケーションを起動した日ごとに、AppStream 2.0 によってアプリケーションレポートが Amazon S3 バケットにエクスポートされます。daily-app-report-[YYYY]-[MM]-[DD].csv という名前のレポートが、以下のフォルダパスを使用して、Amazon S3 アカウント内のネストフォルダ構造に保存されます。

```
[bucket_name]/applications/schedule=DAILY/year=[YYYY]/month=[MM]/day=[DD]/
```

Amazon Athena を使用してレポートをクエリする場合、このネスト構造により、パーティション分割が容易になります。Athena はサーバーレスのインタラクティブなクエリサービスであり、このサービスを使用すると、S3 バケットに保存されているデータを標準の SQL により分析できます。詳細については、「[カスタムレポートを作成して AppStream 2.0 使用状況データを分析する](#)」を参照してください。

各アプリケーション起動はアプリケーションレポートの 1 つのレコードに記述されます。たとえば、ユーザーがセッション中に 5 つの別々のアプリケーションを起動した場合は、アプリケーションレポートに 5 つの別々のレコードが含まれます。以下のいずれかのイベントが発生した場合、アプリケーションは起動されたと記録されます。

- アプリケーション ID がストリーミング URL またはリレー状態のいずれかに埋め込まれているため、セッション開始時にアプリケーションが直接起動される。

- ユーザーが新しいストリーミングセッションの開始時に、アプリケーションカタログからアプリケーションを選択する。
- ユーザーがストリーミングセッション中に、アプリケーションカタログリストからアプリケーションを選択する。

アプリケーションレポートには、その他の方法で起動されたアプリケーションは含まれません。たとえば、ユーザーに Windows Explorer、PowerShell、または Windows デスクトップの [Start (開始)] メニューへのアクセスを許可しており、ユーザーがそれらのツールを使用して直接アプリケーションを起動した場合や、別のプログラムまたはスクリプトによってアプリケーションが起動された場合、それらのアプリケーションの起動はアプリケーションレポートに含まれません。

アプリケーションレポートは毎日、レポート対象日の終了 (UTC 時間) から 24 時間以内に生成されます。セッションが複数日にまたがる場合、セッション中に起動されたアプリケーションは、セッションが終了した日に対応するアプリケーションレポートに反映されます。アプリケーションレポートに含まれるデータについては、「[アプリケーションレポートのフィールド](#)」を参照してください。

AppStream 2.0 使用状況レポートのフィールド

このトピックでは、AppStream 2.0 使用状況レポートに含まれるフィールドに関する情報を提供します。

目次

- [セッションレポートのフィールド](#)
- [アプリケーションレポートのフィールド](#)

セッションレポートのフィールド

以下の表では、AppStream 2.0 セッションレポートに含まれるフィールドについて説明しています。

フィールド名	説明
user_session_id	セッションの一意の識別子 (ID)。
aws_account_id	Amazon Web Services アカウント ID

フィールド名	説明
region	AWS リージョン。
session_start_time	セッションが開始された日時。ISO 8601 標準形式および UTC 時間で指定する必要があります。
session_end_time	セッションが終了した日時。ISO 8601 標準形式および UTC 時間で指定する必要があります。
session_duration_in_seconds	セッションの期間 (秒)。
user_id	認証タイプ内のユーザーの固有 ID。
user_arn	ユーザーの Amazon リソースネーム (ARN)
authentication_type	ユーザーの認証方法。 使用できる値: CUSTOM SAML USERPOOL
authentication_type_user_id	ユーザー ID と認証タイプの連結したもの。ユーザーの料金を見積もる目的でユーザーを一意に識別します。詳細については、 AppStream 2.0 の料金 をご覧ください。
fleet_name	セッションに関連付けられたフリートの名前。
stack_name	セッションに関連付けられたスタックの名前。

フィールド名	説明
instance_type	セッションで使用される AppStream 2.0 インスタンスタイプ。インスタンスタイプの一覧については、 AppStream 2.0 の料金 を参照してください。
eni_private_ip_address	AppStream 2.0 インスタンスによってネットワーク通信に使用された Elastic Network Interface の IP アドレス。
connected_at_least_once	ユーザーが 1 回以上セッションに接続したかどうかを示します。 使用できる値: true false
client_ip_addresses	セッションへの接続に使用されたユーザーデバイスに関連付けられた IP アドレス。ユーザーが 2 回以上、セッションに対する接続と切断を行った場合、最後の 10 個までの個別の IP アドレスがセミコロンで区切られて保存されます。

フィールド名	説明
google_drive_enabled	Google Drive がセッションの永続的ストレージオプションとして有効になっていたかどうかを示します。詳細については、「AppStream 2.0 ユーザーに対して Google ドライブを有効にして管理する」を参照してください。 使用できる値: true false
one_drive_enabled	OneDrive がセッションの永続的ストレージオプションとして有効になっていたかどうかを示します。詳細については、「AppStream 2.0 ユーザーに対して Google ドライブを有効にして管理する」を参照してください。 使用できる値: true false
home_folders_storage_location	ホームフォルダに保存されたファイルに使用された Amazon S3 バケット。
user_settings_clipboard_copy_from_local_device	セッション中にユーザーがクリップボードを使用してローカルデバイスからストリーミングセッションにデータをコピーできたかどうかを示します。 使用できる値: ENABLED DISABLED

フィールド名	説明
<code>user_settings_clipboard_copy_to_local_device</code>	セッション中にユーザーがクリップボードを使用してストリーミングセッションからローカルデバイスにデータをコピーできたかどうかを示します。 使用できる値: ENABLED DISABLED
<code>user_settings_file_upload</code>	セッション中にユーザーがローカルデバイスからストリーミングセッションにファイルをアップロードできたかどうかを示します。 使用できる値: ENABLED DISABLED
<code>user_settings_file_download</code>	セッション中にユーザーがストリーミングセッションからローカルデバイスにファイルをダウンロードできたかどうかを示します。 使用できる値: ENABLED DISABLED
<code>user_settings_printing_to_local_device</code>	セッション中にユーザーがストリーミングセッションからローカルデバイスにファイルを印刷できたかどうかを示します。 使用できる値: ENABLED DISABLED

フィールド名	説明
application_settings_enabled	セッションに対してアプリケーション設定の永続性が有効になっていたかどうかを示します。 使用できる値: true false
domain_joined	セッション起動時に AppStream 2.0 ストリーミングインスタンスが Active Directory ドメインに参加したかどうかを示します。詳細については、「AppStream 2.0 でのアクティブディレクトリの使用」を参照してください。 使用できる値: Y N
max_session_duration	セッションの最大許容期間(秒)。
session_type	セッションのタイプ。 使用できる値: ALWAYS_ON ON_DEMAND
stream_view	ストリームビュー。 使用できる値: APPLICATION DESKTOP
streaming_experience_settings_protocol	セッションがストリーミングを終了したプロトコル。 使用できる値: UDP TCP

フィールド名	説明
instance_id	ユーザーセッションに関連付けられたインスタンス ID。
is_multisession	セッションがマルチセッションフリートに属しているかどうかを示します。 使用できる値: true false

アプリケーションレポートのフィールド

以下の表では、AppStream 2.0 アプリケーションレポートに含まれるフィールドについて説明しています。

フィールド名	説明
user_session_id	セッションの一意の識別子 (ID)。
application_name	Image Assistant で指定されたアプリケーションの名前。この値は、ユーザーが AppStream 2.0 インターフェイスを通じてアプリケーションを起動したときに提供されます。
schedule	レポートが生成される頻度。 可能な値: DAILY
year	レポートの年。
か月	レポートの月。
日	レポートの日。

カスタムレポートを作成して AppStream 2.0 使用状況データを分析する

Amazon Athena はサーバーレスのインタラクティブなクエリサービスであり、このサービスを使用すると、S3 バケットに保存されているデータを標準の SQL により分析できます。Athena を使用して、使用状況レポートを集計したり、他のタイプのカスタムレポートを生成したりできます。

内容

- [AWS Glue クローラを作成する](#)
- [AWS Glue クローラを使用してデータカタログを作成する](#)
- [Athena クエリの作成と実行](#)
- [アテナクエリの操作](#)

AWS Glue クローラを作成する

AWS Glue は、Amazon S3 データからデータベースを作成し、Athena を使用してそのデータベースをクエリできる、フルマネージド型の抽出、変換、ロード (ETL) サービスです。このデータベースは、AWS Glue データカタログとも呼ばれます。AWS Glue クローラは、Amazon S3 データのスキーマを自動的に検出し、対応するデータベースとテーブルを作成できます。AppStream 2.0 には、必要な AWS Glue リソースの作成に使用できる AWS CloudFormation テンプレートが用意されています。

Important

次の手順のステップを完了すると、AWS Glue クローラが作成されます。ただし、これらのステップではクローラを起動しません。クローラを起動するには、次の手順のステップを実行する必要があります。AWS Glue クローラの詳細については、「[クローラの定義](#)」を参照してください。

AWS Glue クローラを作成するには

1. AppStream 2.0 コンソール (<https://console.aws.amazon.com/appstream2>) を開きます。
2. 使用状況レポートをサブスクライブしている AWS リージョンを選択します。
3. ナビゲーションペインで、[Usage Reports (使用状況レポート)] を選択し、使用状況レポートのログ記録が有効になっていることを確認します。

4. [Report Details (レポートの詳細)] タブの [分析] の横にある [CloudFormation テンプレート] リンクをクリックします。

リンクを選択すると AWS CloudFormation コンソールが開き、テンプレートを実行する前に、テンプレートで指定された AWS CloudFormation スタックのパラメータを確認できます。テンプレートを実行すると、AWS Glue クローラといくつかのサンプル Athena クエリが作成されます。

5. [詳細の指定] ページの [ScheduleExpression] の横にあるデフォルト値をそのまま使用するか、クローラを実行する頻度に別の cron 式の値を指定します。他のデフォルト値は変更しないでください。終了したら、[Next (次へ)] を選択します。

クローラはデフォルトで、毎日実行されるようにスケジュールされていますが、毎週、毎月、または別の頻度で実行されるように設定できます。cron 構文については、「[Cron 式](#)」を参照してください。

6. [オプション] ページで、すべてのデフォルト値を受け入れ、[Next (次へ)] を選択します。
7. レビューページで、「**カスタム名で IAM リソースを作成する AWS CloudFormation 場合があることを承認します**」の横にあるチェックボックスを選択し、作成を選択します。

AWS CloudFormation スタックを作成して実行するには、十分な および AWS Glue AWS Identity and Access Management (IAM) アクセス許可が必要です。必要なアクセス許可がない場合は、お客様の Amazon Web Services アカウント管理者に対し、アカウントでこれらのステップを実行するか、以下のアクセス許可をお客様に付与するかのどちらかを依頼します。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "athena:CreateNamedQuery",
        "athena:BatchGetNamedQuery",
        "athena:GetNamedQuery",
        "athena:StartQueryExecution",
        "athena:GetQueryResults",
        "athena:GetQueryExecution",
        "athena:ListNamedQueries",
        "cloudformation:DescribeStacks",
        "cloudformation:GetStackPolicy",
        "cloudformation:DescribeStackEvents",
        "cloudformation:CreateStack",
```

```

        "cloudformation:GetTemplate",
        "cloudformation:ListChangeSets",
        "cloudformation:ListStackResources",
        "iam:GetRole",
        "iam:CreateRole",
        "iam:GetRolePolicy",
        "s3:GetBucketLocation",
        "s3:ListBucketMultipartUploads",
        "s3:ListBucket",
        "s3:ListMultipartUploadParts",
        "s3:PutObject",
        "s3:GetObject",
        "s3:AbortMultipartUpload"
    ],
    "Resource": [
        "arn:aws:iam::*:role/AppStreamUsageReports-AppStreamUsageReportGlueRole*",
        "arn:aws:cloudformation::*:stack/AppStreamUsageReports/*",
        "arn:aws:athena::*:workgroup/primary",
        "arn:aws:s3::aws-athena-query-results-*"
    ]
},
{
    "Effect": "Allow",
    "Action": [
        "iam:AttachRolePolicy",
        "iam:PutRolePolicy",
        "s3:GetObject",
        "s3:ListBucket"
    ],
    "Resource": [
        "arn:aws:s3::appstream-logs-*",
        "arn:aws:iam::*:role/AppStreamUsageReports-AppStreamUsageReportGlueRole*"
    ]
},
{
    "Effect": "Allow",
    "Action": [
        "iam:PassRole"
    ],
    "Resource": [
        "arn:aws:iam::*:role/AppStreamUsageReports-AppStreamUsageReportGlueRole*"
    ]
}

```

```

    ],
    "Condition": {
        "StringEquals": {
            "iam:PassedToService": "glue.amazonaws.com"
        }
    }
},
{
    "Effect": "Allow",
    "Action": [
        "cloudformation:GetTemplateSummary",
        "glue:GetResourcePolicy",
        "glue:GetCrawlers",
        "glue:BatchGetCrawlers",
        "glue:GetClassifiers",
        "glue:CreateClassifier",
        "glue:ListCrawlers",
        "glue:GetTags",
        "glue:GetCrawlerMetrics",
        "glue:GetClassifier",
        "tag:GetResources"
    ],
    "Resource": "*"
},
{
    "Effect": "Allow",
    "Action": "athena:RunQuery",
    "Resource": "arn:aws:athena:*:*:workgroup/primary"
},
{
    "Effect": "Allow",
    "Action": [
        "glue:GetTables",
        "glue:GetPartitions",
        "glue:GetTable"
    ],
    "Resource": [
        "arn:aws:glue:*:*:table/appstream-usage/*",
        "arn:aws:glue:*:*:database/appstream-usage",
        "arn:aws:glue:*:*:catalog"
    ]
},
{
    "Effect": "Allow",

```

```
        "Action": [
            "glue:GetDatabase",
            "glue:CreateDatabase",
            "glue:GetDatabases"
        ],
        "Resource": [
            "arn:aws:glue:*:*:database/appstream-usage",
            "arn:aws:glue:*:*:catalog"
        ]
    },
    {
        "Effect": "Allow",
        "Action": [
            "glue:GetCrawler",
            "glue:StartCrawler",
            "glue:CreateCrawler"
        ],
        "Resource": "arn:aws:glue:*:*:crawler/appstream-usage*"
    },
    {
        "Effect": "Allow",
        "Action": "glue:GetCatalogImportStatus",
        "Resource": "arn:aws:glue:*:*:catalog"
    }
]
```

AWS Glue クローラを使用してデータカタログを作成する

AWS Glue クローラを実行すると、セッションとアプリケーションレポートの構造にマッピングされたデータカタログとスキーマが作成されます。新しいレポートが Amazon S3 バケットに保存されるたびに、クローラを実行して、新しいレポートのデータで AWS Glue Data Catalog を更新する必要があります。

Note

AWS Glue クローラの実行には料金が発生する場合があります。詳細については、「[AWS Glue 料金](#)」を参照してください。

1. <https://console.aws.amazon.com/glue/> で AWS Glue コンソールを開きます。

2. 使用状況レポートをサブスクライブしている AWS リージョンを選択します。
3. appstream-usage-sessions-crawler という名前のクローラの横にあるチェックボックスをオンにし、[クローラの実行] を選択します。appstream-usage-apps-crawler という名前のクローラに対してこのステップを繰り返します。

これらのステップを実行すると、クローラーが実行され、AWS CloudFormation スタックで指定されたスケジュールに従って自動的に実行されるようにスケジュールされます。

4. 両方のクローラーの実行が完了したら、ナビゲーションペインで [データベース] を選択します。使用状況レポートを表す [appstream-usage] という名前のデータベースが表示されます。このデータベースは、appstream-usage-sessions-crawler と appstream-usage-apps-crawler が実行されたときに作成された AWS Glue データカタログです。
5. データベース内のテーブルを表示するには、[appstream-usage]、[テーブル] の順に選択します。それぞれアプリケーションとセッションの使用状況レポートを表す 2 つのテーブル、applications と sessions が表示されます。どちらかのテーブルを選択して、そのスキーマを表示します。

これで、SQL を使用して、Athena でこれらのテーブルに対してクエリを実行できるようになりました。

Athena クエリの作成と実行

Athena を使用して使用状況レポートに対してクエリを実行するには、以下のステップを実行します。

Note

実行した Athena クエリに対しては料金が発生します。詳細については、[Amazon Athena 料金](#) を参照してください。

1. <https://console.aws.amazon.com/athena/> で Athena コンソールを開きます。
2. [Database (データベース)] で、[appstream-usage] を選択します。
3. クエリペインで SQL クエリを入力し、[クエリの実行] を選択します。

アテナクエリの操作

このセクションでは、Athena で実行して Amazon S3 バケット内の使用状況レポートのデータを分析できる SQL クエリを示します。

特定の月のすべてのセッションの統合レポートを作成するには、以下のクエリを実行します。

```
SELECT *
FROM "appstream-usage"."sessions"
WHERE year='four-digit-year'
AND month='two-digit-month'
```

クエリで applications テーブルと sessions テーブルの結合オペレーションを実行することもできます。たとえば、特定の月に各アプリケーションを起動した個別のユーザーを表示するには、以下のクエリを実行します。

```
SELECT DISTINCT apps.application_name, sessions.user_id
FROM "appstream-usage"."applications" apps
  INNER JOIN "appstream-usage"."sessions" sessions ON (apps.user_session_id =
    sessions.user_session_id AND sessions.year='four-digit-year' AND sessions.month='two-
    digit-month')
WHERE apps.year='four-digit-year'
  AND apps.month='two-digit-month'
ORDER BY 1, 2
```

Athena のクエリ結果は、アカウントの Amazon S3 バケットに aws-athena-query-results-*account-id-without-hyphens-region-code* という名前の .csv ファイルとして保存されます。クエリ結果を見つけやすくするために、[名前を付けて保存] を選択し、クエリにその実行前に名前を付けます。[Athena Results (Athena 結果)] ペインでダウンロードアイコンを選択して、クエリの結果を .csv ファイルとしてダウンロードすることもできます。

パフォーマンスを向上させコストを削減するために、Athena ではパーティション分割を使用して、クエリでスキャンされるデータの量を減らします。詳細については、「[データのパーティション分割](#)」を参照してください。使用状況レポートは、年、月、日ごとに Amazon S3 バケットにパーティション分割されます。クエリの条件として [year (年)]、[month (月)]、および [day (日)] フィールドを使用して、クエリを特定の日付範囲パーティションに制限できます。たとえば、以下のクエリは 2019 年 5 月 19 日の週のセッションレポートのみを取り込みます。

```
SELECT SUBSTRING(session_start_time, 1, 10) AS report_date,
  COUNT(DISTINCT user_session_id) AS num_sessions
```



```
FROM "appstream-usage"."sessions"  
WHERE year='2019'  
      AND month='05'  
      AND day BETWEEN '19' and '25'  
GROUP BY 1  
ORDER BY 1
```

これに対して、以下のクエリは同じ結果を生成しますが、パーティションに制限されていないため、Amazon S3 バケットに保存されているすべてのセッションレポートを取り込みます。

```
SELECT SUBSTRING(session_start_time, 1, 10) AS report_date,  
       COUNT(DISTINCT user_session_id) AS num_sessions  
FROM "appstream-usage"."sessions"  
WHERE session_end_time BETWEEN '2019-05-19' AND '2019-05-26'  
GROUP BY 1  
ORDER BY 1
```

セッションが複数日にまたがる場合、セッションレコードとアプリケーションレコードは、セッションが終了した日に対応するセッションレポートにそれぞれ含まれます。このため、特定の日付範囲内でアクティブだったすべてのセッションに関連するレコードを見つける必要がある場合は、フリートに設定した最大セッション長までクエリのパーティションセットを延長することを検討してください。

たとえば、1 か月内で特定のフリートに対してアクティブだったすべてのセッションを表示する場合、フリートの最大セッション期間が 100 時間であれば、以下のようにパーティションセットを 5 日間延長したクエリを実行します。

```
SELECT *  
FROM "appstream-usage"."sessions"  
WHERE fleet_name = 'fleet_name'  
      AND session_start_time BETWEEN '2019-05-01' AND '2019-06-01'  
      AND year='2019'  
      AND (month='05' OR (month='06' AND day<='05'))  
ORDER BY session_start_time
```

AWS Glue クローラを作成した AWS CloudFormation テンプレートは、使用状況データの分析に使用できるいくつかのサンプルクエリを Athena アカウントに作成して保存しました。サンプルクエリとしては以下のものがあります。

- 1 か月あたりの集計セッションレポート

- 1 スタックあたりの平均セッション長
- 1 日あたりのセッション数
- ユーザーあたりの総ストリーミング時間

 Note

オンデマンドの使用料金は、セッションごとに 1 時間単位で切り上げられます。

- アプリ別のユーザー数

これらのクエリを使用するには、以下のステップを実行します。

1. <https://console.aws.amazon.com/athena/> で Athena コンソールを開きます。
2. [保存したクエリ] を選択します。この手順で先ほど示した 5 つのクエリが表示されます。各クエリの名前は「AS2」で始まっています。たとえば、「AS2_users_per_app_curr_mo」と入力します。
3. クエリを実行するには、名前の横にあるオプションではなくクエリ名を選択します。
4. クエリのテキストがクエリペインに表示されます。[Run query] (クエリの実行) を選択します。

これらのクエリを別の AWS CloudFormation テンプレートで表示するには、コードサンプルカタログの[athena-sample-queries-appstream-usage-data_template.yml](#)」を参照してください。AWS

を使用した AppStream 2.0 API コールのログ記録 AWS CloudTrail

Amazon AppStream 2.0 はと統合されています AWS CloudTrail。CloudTrail は、AppStream 2.0 のユーザー、ロール、または AWS サービスによって実行されたアクションを記録するサービスです。CloudTrail は AppStream 2.0 の API 呼び出しをイベントとしてキャプチャします。キャプチャされたコールには、AppStream 2.0 コンソールからの呼び出しと、AppStream 2.0 API オペレーションへのコード呼び出しが含まれます。証跡を作成する場合は、AppStream 2.0 のイベントなど、Amazon S3 バケットへの CloudTrail イベントの継続的な配信を有効にすることができます。証跡を設定しない場合でも、CloudTrail コンソールの [イベント履歴] で最新のイベントを表示できます。CloudTrail によって収集された情報を使用して、リクエスト情報などの詳細を決定できます。たとえば、CloudTrail は AppStream 2.0 に対してどのようなリクエストが行われたか、リクエストの実行元 IP アドレス、実行者、実行日時などの情報を収集します。

設定や有効化の方法など、CloudTrail の詳細については、「[AWS CloudTrail ユーザーガイド](#)」を参照してください。

トピック

- [CloudTrail での AppStream 2.0 情報](#)
- [例: AppStream 2.0 ログファイルのエントリ](#)

CloudTrail での AppStream 2.0 情報

CloudTrail は、AWS アカウントの作成時にアカウントで有効になります。AppStream 2.0 でサポートされているイベントアクティビティが発生すると、そのアクティビティは、[イベント履歴] の他の AWS のサービスイベントとともに CloudTrail イベントに記録されます。AWS アカウントで最近のイベントを表示、検索、ダウンロードできます。詳細については、「[CloudTrail イベント履歴でのイベントの表示](#)」を参照してください。

AppStream 2.0 のイベントなど、AWS アカウントのイベントの継続的な記録については、証跡を作成します。追跡により、CloudTrail はログファイルを Amazon S3 バケットに配信できます。デフォルトでは、コンソールで証跡を作成すると、証跡はすべての AWS リージョンに適用されます。証跡は、AWS パーティション内のすべてのリージョンからのイベントをログに記録し、指定した Amazon S3 バケットにログファイルを配信します。さらに、CloudTrail ログで収集されたイベントデータをより詳細に分析し、それに基づいて行動するように、他の AWS サービスを設定できます。詳細については、次を参照してください:

- [証跡の作成のための概要](#)
- [CloudTrail がサポートするサービスと統合](#)
- [CloudTrail 用 Amazon SNS 通知の構成](#)
- 「[複数のリージョンから CloudTrail ログファイルを受け取る](#)」および「[複数のアカウントから CloudTrail ログファイルを受け取る](#)」

AppStream 2.0 は、CloudTrail ログファイルのイベントとして以下のアクションを記録します。

- [AssociateFleet](#)
- [BatchAssociateUserStack](#)
- [BatchDisassociateUserStack](#)
- [CopyImage](#)
- [CreateDirectoryConfig](#)

- [CreateFleet](#)
- [CreateImageBuilder](#)
- [CreateImageBuilderStreamingURL](#)
- [CreateStack](#)
- [CreateStreamingURL](#)
- [DeleteDirectoryConfig](#)
- [DeleteFleet](#)
- [DeleteImage](#)
- [DeleteImageBuilder](#)
- [DeleteImagePermissions](#)
- [DeleteStack](#)
- [DescribeDirectoryConfigs](#)
- [DescribeFleets](#)
- [DescribeImageBuilders](#)
- [DescribeImagePermissions](#)
- [DescribeImages](#)
- [DescribeSessions](#)
- [DescribeStacks](#)
- [DescribeUserStackAssociations](#)
- [ExpireSession](#)
- [ListAssociatedFleets](#)
- [ListAssociatedStacks](#)
- [ListTagsForResource](#)
- [StartFleet](#)
- [StartImageBuilder](#)
- [StopFleet](#)
- [StopImageBuilder](#)
- [TagResource](#)
- [UntagResource](#)
- [UpdateDirectoryConfig](#)

- [UpdateFleet](#)
- [UpdateImagePermissions](#)
- [UpdateStack](#)

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。同一性情報は次の判断に役立ちます。

- リクエストが、ルートと IAM ユーザー認証情報のどちらを使用して送信されたか。
- 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一時的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 サービスによって行われたかどうか。

詳細については、[CloudTrail userIdentity 要素](#)を参照してください。

例: AppStream 2.0 ログファイルのエントリ

「トレイル」は、指定した Amazon S3 バケットにイベントをログファイルとして配信するように設定できます。CloudTrail のログファイルは、単一か複数のログエントリを含みます。イベントは任意ソースからの単一リクエストを表し、リクエストされたアクション、アクションの日時、リクエストパラメータなどの情報を含みます。CloudTrail ログファイルは、パブリック API 呼び出しの順序付けられたスタックトレースではないため、特定の順序では表示されません。

次は、AssociateFleet イベントを示す CloudTrail ログエントリの例です。

```
{
  "eventVersion": "1.05",
  "userIdentity": {
    "type": "AssumedRole",
    "principalId": "AIDACKCEVSQ6C2EXAMPLE:janeroe",
    "arn": "arn:aws:sts::123456789012:assumed-role/Admin/janeroe",
    "accountId": "123456789012",
    "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
    "sessionContext": {
      "attributes": {
        "mfaAuthenticated": "false",
        "creationDate": "2019-03-12T06:41:50Z"
      },
      "sessionIssuer": {
        "type": "Role",
```

```

        "principalId": "AIDACKCEVSQ6C2EXAMPLE",
        "arn": "arn:aws:iam:: 123456789012:role/Admin",
        "accountId": "123456789012",
        "userName": "Admin"
    }
}
},
"eventTime": "2019-03-12T06:58:09Z",
"eventSource": "appstream.amazonaws.com",
"eventName": "AssociateFleet",
"awsRegion": "us-east-1",
"sourceIPAddress": "198.51.100.15",
"userAgent": "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_11_6) AppleWebKit/537.36
(KHTML, like Gecko) Chrome/72.0.3626.121 Safari/537.36",
"requestParameters": {
    "fleetName": "ExampleFleet1",
    "stackName": "ExampleStack1"
},
"responseElements": null,
"requestID": "3210a159-4494-11e9-8017-873084baf125",
"eventID": "a6fbde60-a55a-46fe-87d4-89ead558dfffd",
"eventType": "AwsApiCall",
"recipientAccountId": "123456789012"
}

```

以下の例は、AppStream 2.0 Image Builder を使用してイメージが作成されたときの CreateImage イベントを示す CloudTrail ログエントリです。

```

{
    "eventVersion": "1.05",
    "userIdentity": {
        "arn": "arn:aws:appstream:us-east-1: 123456789012:image-builder/
ExampleImageBuilder",
        "accountId": "123456789012"
    },
    "eventTime": "2019-03-21T22:32:05Z",
    "eventSource": "appstream.amazonaws.com",
    "eventName": "CreateImage",
    "awsRegion": "us-east-1",
    "requestParameters": null,
    "responseElements": null,
    "eventID": "12b2d6e2-c9a9-402e-8886-2c388d3df610",
    "readOnly": false,

```

```
"eventType": "AwsServiceEvent",
"recipientAccountId": "123456789012",
"serviceEventDetails": {
  "ImageName": "ExampleImage1",
  "ImagePlatform": "WINDOWS",
  "PublicBaseImageReleasedDate": "Tue Jan 15 22:19:56 UTC 2019",
  "ImageDisplayName": "Example Image 1",
  "ImageBuilderSupported": "True",
  "ImageCreatedTime": "Thu Mar 21 22:32:05 UTC 2019",
  "ImageDescription": "Example image for testing",
  "ImageState": "PENDING"
}
}
```

Amazon AppStream 2.0 でのセキュリティ

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 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 クラウドで 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 AWS 責任があります。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。AppStream 2.0 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。また、お客様は、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、AppStream 2.0 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。ここでは、セキュリティとコンプライアンスの目標を満たすように AppStream 2.0 を設定する方法について説明します。また、AppStream 2.0 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。

内容

- [Amazon AppStream 2.0 におけるデータ保護](#)
- [Amazon AppStream 2.0 向け Identity and Access Management](#)
- [Amazon AppStream 2.0 でのログ記録とモニタリング](#)
- [Amazon AppStream 2.0 のコンプライアンスの検証](#)
- [Amazon AppStream 2.0 の耐障害性](#)
- [Amazon AppStream 2.0 のインフラストラクチャセキュリティ](#)
- [Amazon AppStream 2.0 のセキュリティグループ](#)
- [Amazon AppStream 2.0 での更新管理](#)
- [Amazon AppStream 2.0 のサービス間の混乱した代理の防止](#)

- [Amazon AppStream 2.0 のセキュリティのベストプラクティス](#)

Amazon AppStream 2.0 におけるデータ保護

Amazon AppStream 2.0 でのデータ保護には、AWS [責任共有モデル](#)が適用されます。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。お客様は、このインフラストラクチャでホストされているコンテンツに対する管理を維持する責任があります。このコンテンツには、使用される AWS のサービスのセキュリティ設定と管理タスクが含まれます。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のよくある質問](#)」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 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許可のみを各ユーザーに付与できます。また、次の方法でデータを保護することをお勧めします。

- 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 をお勧めします。
- 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。
- AWS 暗号化ソリューションと、サービス内のすべての AWS デフォルトのセキュリティコントロールを使用します。
- Amazon Macie などのアドバンスドマネージドセキュリティサービスを使用します。これは、Amazon S3 に保存されている個人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 140-2 検証済みの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 \(FIPS\) 140-2](#)を参照してください。

顧客の E メールアドレスなどの機密情報やセンシティブ情報は、タグや [Name] (名前) フィールドなどの自由形式のフィールドに配置しないことを強くお勧めします。これは、コンソール、API、または SDK を使用して AppStream 2.0 AWS CLIまたは他の AWS のサービスを使用する場合も同様です。AWS SDKs タグまたは名前に使用する自由記入欄に入力したデータは、課金や診断ログに使用される場合があります。外部サーバーへ URL を供給する場合は、そのサーバーへのリクエストを検証するために、認証情報を URL に含めないことを強くお勧めします。

トピック

- [保管時の暗号化](#)
- [転送時の暗号化](#)
- [管理者のコントロール](#)
- [アプリケーションアクセス](#)

保管時の暗号化

AppStream 2.0 フリートインスタンスは、本質的に一時的なものです。ユーザーのストリーミングセッションが終了すると、基礎となるインスタンスとそれに関連付けられた Amazon Elastic Block Store (Amazon EBS) ボリュームが終了します。また、AppStream 2.0 は鮮度のために未使用のインスタンスを定期的にリサイクルします。

[アプリケーション設定の永続性](#)、[ホームフォルダ](#)、[セッションスクリプト](#)、または[使用状況レポート](#)を有効にすると、ユーザーによって生成され、Amazon Simple Storage Service バケットに保存されるデータは、保管時に暗号化されます。AWS Key Management Service は、安全で可用性の高いハードウェアとソフトウェアを組み合わせ、クラウド向けに拡張されたキー管理システムを提供するサービスです。Amazon S3 は、[AWS マネージド CMK](#) を使用して Amazon S3 オブジェクトデータを暗号化します。

転送時の暗号化

次のテーブルに、転送中のデータの暗号化方法に関する情報を示します。該当する場合は、AppStream 2.0 の他のデータ保護方法もリストされます。

データ	ネットワークパス	保護方法
ウェブ資産 このトラフィックには、イメージや JavaScript ファイルなどのアセットが含まれます。	AppStream 2.0 ユーザーと AppStream 2.0 の間で	TLS 1.2 を使用して暗号化
ピクセルおよび関連するストリーミングトラフィック	AppStream 2.0 ユーザーと AppStream 2.0 の間で	256 ビット高度暗号化規格 (AES-256) を使用して暗号化

データ	ネットワークパス	保護方法
		TLS 1.2 を使用して転送
API トラフィック	AppStream 2.0 ユーザーと AppStream 2.0 の間で	TLS 1.2 を使用して暗号化 接続を作成するリクエストは、SigV4 を使用して署名されます。
ユーザーが生成したアプリケーション設定とホームフォルダデータ アプリケーション設定の永続性とホームフォルダが有効になっている場合に適用されます。	AppStream 2.0 ユーザーと Amazon S3 の間で	Amazon S3 SSL エンドポイントを使用して暗号化
AppStream 2.0 で管理されたトラフィック	AppStream 2.0 ストリーミングインスタンスと: • AppStream 2.0 管理サービス • AWS Amazon Web Services アカウントの サービスとリソース • 以外のAWS サービスとリソース (Google Drive や Microsoft OneDrive など)	TLS 1.2 を使用して暗号化 接続を作成するリクエストは、該当する場合は SigV4 を使用して署名されます。

管理者のコントロール

AppStream 2.0 には、ローカルコンピュータと AppStream 2.0 フリートインスタンス間でユーザーがデータを転送する方法を制限するために使用できる管理コントロールが用意されています。[AppStream 2.0 スタックの作成または更新時](#)に、次の項目を制限または無効にできます。

- クリップボード/コピーと貼り付けアクション
- ファイルのアップロードとダウンロード (フォルダとドライブのリダイレクトを含む)

- 印刷

AppStream 2.0 イメージを作成するときに、Windows 用 AppStream 2.0 クライアントから AppStream 2.0 フリートインスタンスにリダイレクトできる USB デバイスを指定できます。指定する USB デバイスは、ユーザーの AppStream 2.0 ストリーミングセッション中に使用できます。詳細については、「[ストリーミングアプリケーションで使用する USB デバイスを認証する](#)」を参照してください。

アプリケーションアクセス

デフォルトでは、AppStream 2.0 はイメージで指定するアプリケーションが、Image Builder および フリートインスタンス上の他のアプリケーションと実行可能ファイルを起動できるようにします。これにより、他のアプリケーションに依存するアプリケーション (たとえば、ブラウザを起動して製品のウェブサイトに移動するアプリケーション) が想定どおりに機能します。管理コントロール、セキュリティグループ、およびその他のセキュリティソフトウェアを設定して、リソースにアクセスし、ローカルコンピュータとフリートインスタンス間でデータを転送するために必要な最小限のアクセス許可をユーザーに付与します。

[Microsoft AppLocker](#) などのアプリケーション制御ソフトウェアおよびポリシーを使用して、ユーザーが実行できるアプリケーションとファイルを制御できます。アプリケーション制御ソフトウェアとポリシーは、ユーザーが AppStream 2.0 Image Builder とフリートインスタンスで実行できる実行可能ファイル、スクリプト、Windows インストーラーファイル、ダイナミックリンクライブラリ、およびアプリケーションパッケージを制御するのに役立ちます。

Note

AppStream 2.0 エージェントソフトウェアは、Windows コマンドプロンプトと Windows Powershell に依存して、ストリーミングインスタンスをプロビジョニングします。ユーザーが Windows コマンドプロンプトまたは Windows Powershell を起動できないように選択する場合は、Windows NT AUTHORITY\SYSTEM または管理者グループのユーザーにはポリシーを適用しないでください。

ルールタイプ	アクション	Windows ユーザーまたは グループ	名前/パス	条件	説明
実行可能ファイル	許可	NT AUTHORITY \System	*	パス	AppStream 2.0 エージェントソフトウェアに必要
実行可能ファイル	許可	BUILTIN\Administrators	*	パス	AppStream 2.0 エージェントソフトウェアに必要
実行可能ファイル	許可	全員	%PROGRAMFILES%\nodejs*	パス	AppStream 2.0 エージェントソフトウェアに必要
実行可能ファイル	許可	全員	%PROGRAMFILES%\NICE*	パス	AppStream 2.0 エージェントソフトウェアに必要
実行可能ファイル	許可	全員	%PROGRAMFILES%\Amazon*	パス	AppStream 2.0 エージェントソフトウェアに必要
実行可能ファイル	許可	全員	%PROGRAMFILES%\<default-browser>*	パス	Google ドライブや Microsoft OneDrive for Business などの永続的ストレージソリューションを使用す

ルールタイプ	アクション	Windows ユーザーまたは グループ	名前/パス	条件	説明
					る場合、AppStream 2.0 エージェントソフトウェアに必要です。この例外は、AppStream 2.0 ホームフォルダを使用する場合には必要ありません。

Amazon AppStream 2.0 向け Identity and Access Management

セキュリティ認証情報は、のサービスを識別 AWS し、AppStream 2.0 AWS リソースなどのリソースを無制限に使用できるようにします。AppStream 2.0 および AWS Identity and Access Management (IAM) の機能を使用して、他のユーザー、サービス、およびアプリケーションがお客様の AppStream 2.0 リソースを使用できるようにします。その際、お客様のセキュリティ認証情報は共有されません。

他のユーザーが Amazon Web Services アカウント内のリソースをどのように使用するかを制御するには IAM を、AppStream 2.0 ストリーミングインスタンスへのアクセスを制御するにはセキュリティグループを使用できます。AppStream 2.0 のリソースの完全使用または制限付き使用のどちらを許可するか選択できます。

内容

- [ストリーミングインスタンスへのネットワークアクセス](#)
- [AWS 管理ポリシーとリンクされたロールを使用して AppStream 2.0 リソースへの管理者アクセスを管理する](#)
- [IAM ポリシーを使用して Application Auto Scaling への管理者アクセスを管理する](#)

- [IAM ポリシーを使用したホームフォルダおよびアプリケーション設定の永続化用の Amazon S3 バケットへの管理者アクセスの管理](#)
- [IAM ロールを使用して、AppStream 2.0 ストリーミングインスタンスで実行されるアプリケーションとスクリプトにアクセス許可を付与する](#)
- [Red Hat Enterprise Linux および Rocky Linux での SELinux](#)
- [Amazon AppStream 2.0 での Cookie ベースの認証](#)

ストリーミングインスタンスへのネットワークアクセス

セキュリティグループは、ストリーミングインスタンスに到達できるトラフィックを制御するステータフルなファイアウォールとして機能します。AppStream 2.0 ストリーミングインスタンスを起動するときは、1 つ以上のセキュリティグループに割り当てます。セキュリティグループのそれぞれに、そのインスタンスへのトラフィックを制御するルールを追加できます。セキュリティグループのルールはいつでも変更できます。新しいルールは、セキュリティグループが割り当てられているすべてのインスタンスに自動的に適用されます。

詳細については、「[Amazon AppStream 2.0 のセキュリティグループ](#)」を参照してください。

AWS 管理ポリシーとリンクされたロールを使用して AppStream 2.0 リソースへの管理者アクセスを管理する

デフォルトでは、AppStream 2.0 リソースの作成や変更、または AppStream 2.0 API を使用したタスクの実行に必要なアクセス許可は IAM ユーザーにはありません。つまり、これらのユーザーは AppStream 2.0 コンソールまたは AppStream 2.0 AWS CLI コマンドを使用してこれらのアクションを実行できません。IAM ユーザーがリソースを作成または変更し、タスクを実行できるようにするには、それらのアクセス許可を必要とする IAM ユーザーまたはグループに IAM ポリシーをアタッチします。

ポリシーをユーザー、ユーザーのグループ、または IAM ロールにアタッチする場合、ポリシーによって特定リソースの特定タスクを実行するユーザーの権限が許可または拒否されます。

内容

- [AWS AppStream 2.0 リソースにアクセスするために必要な管理ポリシー](#)
- [AppStream 2.0、Application Auto Scaling、および AWS Certificate Manager Private CA に必要なロール](#)
- [AmazonAppStreamServiceAccess のサービスロールおよびポリシーを確認する](#)

- [ApplicationAutoScalingForAmazonAppStreamAccess](#) サービスロールとポリシーの確認
- [AWSServiceRoleForApplicationAutoScaling_AppStreamFleet](#) サービスにリンクされたロールとポリシーの確認
- [AmazonAppStreamPCAAccess](#) のサービスロールおよびポリシーを確認する

AWS AppStream 2.0 リソースにアクセスするために必要な管理ポリシー

AppStream 2.0 への完全な管理アクセスまたは読み取り専用アクセスを提供するには、これらのアクセス許可を必要とする IAM ユーザーまたはグループに、次のいずれか AWS の管理ポリシーをアタッチする必要があります。AWS 管理ポリシーは、AWS が作成および管理するスタンドアロンポリシーです。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](#)」を参照してください。

Note

では AWS、IAM ロールを使用して AWS サービスにアクセス許可を付与し、AWS リソースにアクセスできるようにします。ロールにアタッチされているポリシーによって、サービスがアクセスできる AWS リソースと、それらのリソースで何ができるかが決まります。AppStream 2.0 の場合、AmazonAppStreamFullAccess ポリシーで定義されたアクセス許可に加えて、AWS アカウントに必要なロールも必要です。詳細については、「[the section called “AppStream 2.0、Application Auto Scaling、および AWS Certificate Manager Private CA に必要なロール”](#)」を参照してください。

AmazonAppStreamFullAccess

この管理ポリシーは、AppStream 2.0 リソースへの完全な管理アクセスを提供します。AppStream 2.0 リソースを管理し、コマンドラインインターフェイス (AWS CLI)、AWS SDK、または AWS マネジメントコンソールを使用して API AWS アクションを実行するには、このポリシーで定義されているアクセス許可が必要です。

IAM ユーザーとして AppStream 2.0 コンソールにサインインする場合は、このポリシーを にアタッチする必要があります AWS アカウント。コンソールのフェデレーションを使用してサインインする場合は、フェデレーションで使した IAM ロールにこのポリシーをアタッチする必要があります。

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



```

        "Action": "iam:ListRoles",
        "Effect": "Allow",
        "Resource": ""
    },
    {
        "Action": "iam:PassRole",
        "Effect": "Allow",
        "Resource": "arn:aws:iam::role/service-role/
ApplicationAutoScalingForAmazonAppStreamAccess",
        "Condition": {
            "StringLike": {
                "iam:PassedToService": "application-autoscaling.amazonaws.com"
            }
        }
    },
    {
        "Action": "iam:CreateServiceLinkedRole",
        "Effect": "Allow",
        "Resource": "arn:aws:iam::role/aws-service-role/appstream.application-
autoscaling.amazonaws.com/AWSServiceRoleForApplicationAutoScaling_AppStreamFleet
(http://appstream.application-autoscaling.amazonaws.com/
AWSServiceRoleForApplicationAutoScaling_AppStreamFleet)",
        "Condition": {
            "StringLike": {
                "iam:AWSServiceName": "appstream.application-
autoscaling.amazonaws.com"
            }
        }
    }
]
}

```

AmazonAppStreamReadOnlyAccess

この管理ポリシーは、AppStream 2.0 リソースへの読み取り専用アクセスを提供します。

```

{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Action": [
                "appstream:Get*",
                "appstream:List*",
                "appstream:Describe*"
            ]
        }
    ]
}

```

```
    ],  
    "Effect": "Allow",  
    "Resource": "*"    
  }  
]  
}
```

AppStream 2.0 コンソールは、CLI または AWS SDK では利用できない機能を提供する 2 AWS 用の追加アクションを使用します。AmazonAppStreamFullAccess ポリシーと AmazonAppStreamReadOnlyAccess ポリシーのどちらも、これらのアクションのアクセス権限を提供します。

アクション	説明	アクセスレベル
GetImageBuilders	イメージビルダー名が設定されている場合、1 つ以上の指定したイメージビルダーを記述するリストを取得する許可を付与。それ以外の場合は、アカウントのすべてのイメージビルダーが記述されます。	読み取り
GetParametersForThemeAssetUpload	カスタムブランドのテーマアセットをアップロードするアクセス許可を付与します。詳細については、「 カスタムブランディングを Amazon AppStream 2.0 に追加する 」を参照してください。	書き込み

AmazonAppStreamPCAAccess

この管理ポリシーは、証明書ベースの認証のために、AWS アカウントの Certificate Manager プライベート CA リソースへの AWS 完全な管理アクセスを提供します。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "acm-pca:IssueCertificate",
        "acm-pca:GetCertificate",
        "acm-pca:DescribeCertificateAuthority"
      ],
      "Resource": "arn:*:acm-pca:*:*:*:*",
      "Condition": {
        "StringLike": {
          "aws:ResourceTag/euc-private-ca": "*"
        }
      }
    }
  ]
}
```

AmazonAppStreamServiceAccess

このマネージドポリシーは、AppStream 2.0 サービスロールのデフォルトポリシーです。

このロールのアクセス許可ポリシーにより、AppStream 2.0 は次のアクションを実行できます。

- AppStream 2.0 フリートのアカウントでサブネットを使用する場合、AppStream 2.0 はサブネット、VPCs、アベイラビリティゾーンを記述し、それらのサブネット内のフリートインスタンスに関連付けられたすべての Elastic Network Interface のライフサイクルを作成および管理できます。これには、これらのサブネットからそれらの Elastic Network Interface にセキュリティグループと IP アドレスをアタッチすることも含まれます。
- UPP や HomeFolders などの機能を使用する場合、AppStream 2.0 はアカウントで Amazon S3 バケット、オブジェクトとそのライフサイクル、ポリシー、暗号化設定を作成および管理できます。これらのバケットには、次の命名プレフィックスが含まれます。
 - "arn:aws:s3:::appstream2-36fb080bb8-",

- "arn:aws:s3:::appstream-app-settings-",
- "arn:aws:s3:::appstream-logs-"

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "ec2:DescribeVpcs",
        "ec2:DescribeSubnets",
        "ec2:DescribeAvailabilityZones",
        "ec2:CreateNetworkInterface",
        "ec2:DescribeNetworkInterfaces",
        "ec2>DeleteNetworkInterface",
        "ec2:DescribeSubnets",
        "ec2:AssociateAddress",
        "ec2:DisassociateAddress",
        "ec2:DescribeRouteTables",
        "ec2:DescribeSecurityGroups",
        "ec2:DescribeVpcEndpoints",
        "s3:ListAllMyBuckets",
        "ds:DescribeDirectories"
      ],
      "Resource": "*"
    },
    {
      "Effect": "Allow",
      "Action": [
        "s3:CreateBucket",
        "s3:ListBucket",
        "s3:GetObject",
        "s3:PutObject",
        "s3>DeleteObject",
        "s3:GetObjectVersion",
        "s3>DeleteObjectVersion",
        "s3:GetBucketPolicy",
        "s3:PutBucketPolicy",
        "s3:PutEncryptionConfiguration"
      ],
      "Resource": [
        "arn:aws:s3:::appstream2-36fb080bb8-*",
        "arn:aws:s3:::appstream-app-settings-*",

```

```

        "arn:aws:s3:::appstream-logs-*"
    ]
}

```

ApplicationAutoScalingForAmazonAppStreamAccess

このマネージドポリシーにより、AppStream 2.0 でアプリケーションの自動スケーリングが可能になります。

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "appstream:UpdateFleet",
        "appstream:DescribeFleets"
      ],
      "Resource": [
        "*"
      ]
    },
    {
      "Effect": "Allow",
      "Action": [
        "cloudwatch:DescribeAlarms"
      ],
      "Resource": [
        "*"
      ]
    }
  ]
}

```

AWSApplicationAutoscalingAppStreamFleetPolicy

このマネージドポリシーは、AppStream 2.0 と CloudWatch へのアクセス権限を Application Auto Scaling に付与します。

```

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

```

```
"Statement": [  
  {  
    "Effect": "Allow",  
    "Action": [  
      "appstream:UpdateFleet",  
      "appstream:DescribeFleets",  
      "cloudwatch:PutMetricAlarm",  
      "cloudwatch:DescribeAlarms",  
      "cloudwatch>DeleteAlarms"  
    ],  
    "Resource": [  
      "*"   
    ]  
  }  
]
```

AWS 管理ポリシーに対する AppStream 2.0 の更新

このサービスがこれらの変更の追跡を開始してからの AppStream 2.0 の AWS マネージドポリシーの更新に関する詳細を表示します。このページへの変更に関する自動アラートについては、[ドキュメント履歴](#) ページの RSS フィードを購読してください。

変更	説明	日付
AppStream 2.0 が変更の追跡を開始しました	AppStream 2.0 が AWS マネージドポリシーの変更の追跡を開始	2022 年 10 月 31 日

AppStream 2.0、Application Auto Scaling、および AWS Certificate Manager Private CA に必要なロール

では AWS、IAM ロールを使用して AWS サービスにアクセス許可を付与し、AWS リソースにアクセスできるようにします。ロールにアタッチされているポリシーによって、サービスがアクセスできる AWS リソースと、それらのリソースで何ができるかが決まります。AppStream 2.0 では、AmazonAppStreamFullAccess ポリシーで定義されたアクセス許可に加えて、AWS アカウントに次のロールも必要です。

ロール

- [AmazonAppStreamServiceAccess](#)
- [ApplicationAutoScalingForAmazonAppStreamAccess](#)
- [AWSServiceRoleForApplicationAutoScaling_AppStreamFleet](#)
- [AmazonAppStreamPCAAccess](#)

AmazonAppStreamServiceAccess

このロールは、AWS リージョンで AppStream 2.0 の使用を開始すると自動的に作成されるサービスロールです。サービスロールの詳細については、IAM ユーザーガイドの「[AWS サービスにアクセス許可を委任するロールの作成](#)」を参照してください。

AppStream 2.0 リソースの作成中に、AppStream 2.0 サービスは、このロールを引き受けることで、ユーザーに代わって他の AWS サービスへの API コールを行います。フリートを作成するには、アカウントにこのロールが必要です。このロールが AWS アカウントになく、必要な IAM アクセス許可と信頼関係ポリシーがアタッチされていない場合、AppStream 2.0 フリートを作成することはできません。

詳細については、[AmazonAppStreamServiceAccess のサービスロールおよびポリシーを確認する](#)「」を参照して、AmazonAppStreamServiceAccess サービスロールが存在し、正しいポリシーがアタッチされているかどうかを確認します。

Note

このサービスロールには、AppStream 2.0 の使用を開始する最初のユーザーとは異なるアクセス許可を持つことができます。このロールのアクセス許可の詳細については、「」の「AmazonAppStreamServiceAccess」を参照してください [the section called “AWS AppStream 2.0 リソースにアクセスするために必要な管理ポリシー”](#)。

ApplicationAutoScalingForAmazonAppStreamAccess

このロールは、AWS リージョンで AppStream 2.0 の使用を開始すると自動的に作成されるサービスロールです。サービスロールの詳細については、IAM ユーザーガイドの「[AWS サービスにアクセス許可を委任するロールの作成](#)」を参照してください。

自動スケーリングは、AppStream 2.0 フリートの機能です。スケーリングポリシーを設定するには、AWS アカウントにこのサービスロールが必要です。このサービスロールが AWS アカウントにな

く、必要な IAM アクセス許可と信頼関係ポリシーがアタッチされていない場合、AppStream 2.0 フリートをスケールすることはできません。

詳細については、「[ApplicationAutoScalingForAmazonAppStreamAccess サービスロールとポリシーの確認](#)」を参照してください。

AWSServiceRoleForApplicationAutoScaling_AppStreamFleet

このロールは、自動的に作成されるサービスにリンクされたロールです。詳細については、アプリケーション Auto Scaling ユーザーガイドの「[サービスにリンクされたロール](#)」を参照してください。

Application Auto Scaling は、サービスにリンクされたロールを使用して、ユーザーに代わって自動スケーリングを実行します。サービスにリンクされたロールは、AWS サービスに直接リンクされた IAM ロールです。このロールには、サービスがユーザーに代わって他の AWS サービスを呼び出すために必要なすべてのアクセス許可が含まれます。

詳細については、「[AWSServiceRoleForApplicationAutoScaling_AppStreamFleet サービスにリンクされたロールとポリシーの確認](#)」を参照してください。

AmazonAppStreamPCAAccess

このロールは、AWS リージョンで AppStream 2.0 の使用を開始すると自動的に作成されるサービスロールです。サービスロールの詳細については、IAM ユーザーガイドの「[AWS サービスにアクセス許可を委任するロールの作成](#)」を参照してください。

証明書ベースの認証は、Microsoft Active Directory ドメインに結合している AppStream 2.0 フリーットの機能です。証明書ベースの認証を有効にして使用するには、AWS アカウントにこのサービスロールが必要です。このサービスロールが AWS アカウントになく、必要な IAM アクセス許可と信頼関係ポリシーがアタッチされていない場合、証明書ベースの認証を有効化または使用することはできません。

詳細については、「[the section called “AmazonAppStreamPCAAccess のサービスロールおよびポリシーを確認する”](#)」を参照してください。

AmazonAppStreamServiceAccess のサービスロールおよびポリシーを確認する

AmazonAppStreamServiceAccess サービスロールの有無と、適切なポリシーがアタッチされているかどうかを確認するには、このセクションのステップを実行します。このロールがアカウントにな

く、作成する必要がある場合、お客様または必要なアクセス許可を持つ管理者がこのステップを実行し、Amazon Web Services アカウントで AppStream 2.0 の使用を開始する必要があります。

AmazonAppStreamServiceAccess の IAM サービスロールの有無を確認するには

1. IAM コンソール (<https://console.aws.amazon.com/iam/>) を開きます。
2. ナビゲーションペインで Roles (ロール) を選択してください。
3. 検索ボックスに amazonappstreamservice と入力して、選択するロールのリストを絞り込み、[AmazonAppStreamServiceAccess] を選択します。表示されたら、このロールを選択して、ロールの [概要] ページを表示します。
4. [アクセス許可] タブで、AmazonAppStreamServiceAccess アクセス許可ポリシーがアタッチされているかどうかを確認します。
5. ロールの [概要] ページに戻ります。
6. [信頼関係] タブで、[Show policy document (ポリシードキュメントの表示)] を選択して、AmazonAppStreamServiceAccess 信頼関係ポリシーがアタッチされていて、適切な形式に従っているかどうかを確認します。アタッチされている場合、信頼関係は正しく設定されています。[Cancel (キャンセル)] を選択して、IAM コンソールを閉じます。

AmazonAppStreamServiceAccess 信頼関係ポリシー

AmazonAppStreamServiceAccess 信頼関係ポリシーには、AppStream 2.0 サービスをプリンシパルとして含める必要があります。プリンシパルは、アクションを実行し AWS、リソースにアクセスできるのエンティティです。このポリシーには sts:AssumeRole アクションも含める必要があります。次のポリシー設定は、AppStream 2.0 を信頼されたエンティティとして定義します。

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

ApplicationAutoScalingForAmazonAppStreamAccess サービスロールとポリシーの確認

ApplicationAutoScalingForAmazonAppStreamAccess サービスロールの有無と、適切なポリシーがアタッチされているかどうかを確認するには、このセクションのステップを実行します。このロールがアカウントになく、作成する必要がある場合、お客様または必要なアクセス許可を持つ管理者がこのステップを実行し、Amazon Web Services アカウントで AppStream 2.0 の使用を開始する必要があります。

ApplicationAutoScalingForAmazonAppStreamAccess IAM サービスロールが存在するかどうかを確認するには

1. IAM コンソール (<https://console.aws.amazon.com/iam/>) を開きます。
2. ナビゲーションペインで Roles (ロール) を選択してください。
3. 検索ボックスに applicationautoscaling と入力して、選択するロールのリストを絞り込み、ApplicationAutoScalingForAmazonAppStreamAccess を選択します。表示されたら、このロールを選択して、ロールの [概要] ページを表示します。
4. [Permissions] (アクセス許可) タブで、ApplicationAutoScalingForAmazonAppStreamAccess アクセス許可ポリシーがアタッチされているかどうかを確認します。
5. ロールの [概要] ページに戻ります。
6. [Trust relationships] (信頼関係) タブで、[Show policy document] (ポリシードキュメントの表示) を選択して、ApplicationAutoScalingForAmazonAppStreamAccess 信頼関係ポリシーがアタッチされていて、適切な形式に従っているかどうかを確認します。アタッチされている場合、信頼関係は正しく設定されています。[Cancel (キャンセル)] を選択して、IAM コンソールを閉じます。

ApplicationAutoScalingForAmazonAppStreamAccess 信頼関係ポリシー

ApplicationAutoScalingForAmazonAppStreamAccess 信頼関係ポリシーには、Application Auto Scaling サービスをプリンシパルとして含める必要があります。このポリシーには sts:AssumeRole アクションも含める必要があります。次のポリシー設定では、Application Auto Scaling を信頼されたエンティティとして定義しています。

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

```
"Principal": {
  "Service": "application-autoscaling.amazonaws.com"
},
"Action": "sts:AssumeRole"
}
]
}
```

AWSServiceRoleForApplicationAutoScaling_AppStreamFleet サービスにリンクされたロールとポリシーの確認

AWSServiceRoleForApplicationAutoScaling_AppStreamFleet サービスにリンクされたロールの有無と、適切なポリシーがアタッチされているかどうかを確認するには、このセクションのステップを実行します。このロールがアカウントになく、作成する必要がある場合、お客様または必要なアクセス許可を持つ管理者がこのステップを実行し、Amazon Web Services アカウントで AppStream 2.0 の使用を開始する必要があります。

AWSServiceRoleForApplicationAutoScaling_AppStreamFleet IAM サービスにリンクされたロールが存在するかどうかを確認するには

1. IAM コンソール (<https://console.aws.amazon.com/iam/>) を開きます。
2. ナビゲーションペインで Roles (ロール) を選択してください。
3. 検索ボックスに applicationautoscaling と入力して、選択するロールのリストを絞り込み、AWSServiceRoleForApplicationAutoScaling_AppStreamFleet を選択します。表示されたら、このロールを選択して、ロールの [概要] ページを表示します。
4. [Permissions] (アクセス許可) タブで、AWSApplicationAutoscalingAppStreamFleetPolicy アクセス許可ポリシーがアタッチされているかどうかを確認します。
5. [ロール] の概要ページに戻ります。
6. [Trust relationships] (信頼関係) タブで、[Show policy document] (ポリシードキュメントの表示) を選択して、AWSServiceRoleForApplicationAutoScaling_AppStreamFleet 信頼関係ポリシーがアタッチされていて、適切な形式に従っているかどうかを確認します。アタッチされている場合、信頼関係は正しく設定されています。[Cancel (キャンセル)] を選択して、IAM コンソールを閉じます。

AWSServiceRoleForApplicationAutoScaling_AppStreamFleet 信頼関係ポリシー

AWSServiceRoleForApplicationAutoScaling_AppStreamFleet 信頼関係ポリシーには、プリンシパルとして `appstream.application-autoscaling.amazonaws.com` を含める必要があります。このポリシーには `sts:AssumeRole` アクションも含める必要があります。次のポリシー設定では、`appstream.application-autoscaling.amazonaws.com` を信頼されたエンティティとして定義しています。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "appstream.application-autoscaling.amazonaws.com"
      },
      "Action": "sts:AssumeRole"
    }
  ]
}
```

AmazonAppStreamPCAAccess のサービスロールおよびポリシーを確認する

AmazonAppStreamPCAAccess サービスロールの有無と、適切なポリシーがアタッチされているかどうかを確認するには、このセクションのステップを実行します。このロールがアカウントになく、作成する必要がある場合、お客様または必要なアクセス許可を持つ管理者がこのステップを実行し、Amazon Web Services アカウントで AppStream 2.0 の使用を開始する必要があります。

AmazonAppStreamPCAAccess の IAM サービスロールの有無を確認するには

1. IAM コンソール (<https://console.aws.amazon.com/iam/>) を開きます。
2. ナビゲーションペインで Roles (ロール) を選択してください。
3. 検索ボックスに「appstreampca」と入力して、選択するロールのリストを絞り込み、[AmazonAppStreamPCAAccess] を選択します。表示されたら、このロールを選択して、ロールの [概要] ページを表示します。
4. [アクセス許可] タブで、AmazonAppStreamPCAAccess アクセス権限ポリシーがアタッチされているかどうかを確認します。
5. [ロール] の概要ページに戻ります。

6. [信頼関係] タブで、[Show policy document] (ポリシードキュメントの表示) を選択して、AmazonAppStreamPCAAccess 信頼関係ポリシーがアタッチされていて、適切な形式に従っているかどうかを確認します。アタッチされている場合、信頼関係は正しく設定されています。[Cancel (キャンセル)] を選択して、IAM コンソールを閉じます。

AmazonAppStreamPCAAccess 信頼関係ポリシー

AmazonAppStreamPCAAccess 信頼関係ポリシーには、prod.euc.ecm.amazonaws.com をプリンシパルとして含める必要があります。このポリシーには sts:AssumeRole アクションも含める必要があります。次のポリシー設定は、ECM を信頼されたエンティティとして定義します。

CLI を使用して AmazonAppStreamPCAAccess AWS 信頼関係ポリシーを作成するには

1. AmazonAppStreamPCAAccess.json という名前の JSON ファイルを次の内容で作成します。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": [
          "prod.euc.ecm.amazonaws.com"
        ]
      },
      "Action": "sts:AssumeRole",
      "Condition": {}
    }
  ]
}
```

2. 必要に応じてAmazonAppStreamPCAAccess.jsonパスを調整し、次の AWS CLI コマンドを実行して信頼関係ポリシーを作成し、AmazonAppStreamPCAAccess 管理ポリシーをアタッチします。管理ポリシーの詳細については、[「the section called “AWS AppStream 2.0 リソースにアクセスするために必要な管理ポリシー”」](#)を参照してください。

```
aws iam create-role --path /service-role/ --role-name AmazonAppStreamPCAAccess --assume-role-policy-document file://AmazonAppStreamPCAAccess.json
```

```
aws iam attach-role-policy --role-name AmazonAppStreamPCAAccess --policy-arn
arn:aws:iam::aws:policy/AmazonAppStreamPCAAccess
```

IAM ポリシーを使用して Application Auto Scaling への管理者アクセスを管理する

フリートの自動スケーリングは、AppStream 2.0、Amazon CloudWatch、および Application Auto Scaling API の組み合わせによって可能になります。AppStream 2.0 フリートは AppStream 2.0 で作成され、アラームは CloudWatch で作成され、スケーリングポリシーは Application Auto Scaling で作成されます。

[AmazonAppStreamFullAccess](#) ポリシーで定義されたアクセス許可に加えて、フリートスケーリング設定にアクセスする IAM ユーザーには、動的スケーリングをサポートするサービスに必要なアクセス許可が必要です。IAM ユーザーには、次のポリシー例に示すアクションを使用するためのアクセス許可が必要です。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "appstream:*",
        "application-autoscaling:*",
        "cloudwatch:DeleteAlarms",
        "cloudwatch:DescribeAlarmsForMetric",
        "cloudwatch:DisableAlarmActions",
        "cloudwatch:DescribeAlarms",
        "cloudwatch:EnableAlarmActions",
        "cloudwatch:ListMetrics",
        "cloudwatch:PutMetricAlarm",
        "iam:ListRoles"
      ],
      "Resource": "*"
    },
    {
      "Sid": "iamPassRole",
      "Effect": "Allow",
      "Action": [
```

```
        "iam:PassRole"
      ],
      "Resource": "*",
      "Condition": {
        "StringEquals": {
          "iam:PassedToService": "application-autoscaling.amazonaws.com"
        }
      }
    }
  ]
}
```

独自の IAM ポリシーを作成して、Application Auto Scaling API への呼び出しに対してより具体的なアクセス許可を設定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 Auto Scaling ユーザーガイドの「認証とアクセスコントロール」を参照してください。<https://docs.aws.amazon.com/autoscaling/application/userguide/auth-and-access-control.html>

IAM ポリシーを使用したホームフォルダおよびアプリケーション設定の永続化用の Amazon S3 バケットへの管理者アクセスの管理

次の例では、IAM ポリシーを使用して、ホームフォルダとアプリケーション設定の永続化のために Amazon S3 バケットへのアクセスを管理する方法を示します。

例

- [ホームフォルダおよびアプリケーション設定の永続化用の Amazon S3 バケットの削除](#)
- [ホームフォルダおよびアプリケーション設定の永続化用の Amazon S3 バケットへの管理者アクセスの制限](#)

ホームフォルダおよびアプリケーション設定の永続化用の Amazon S3 バケットの削除

AppStream 2.0 は、作成するバケットに Amazon S3 バケットポリシーを追加することで、バケットが誤って削除されるのを防いでいます。S3 バケットを削除するには、最初に S3 バケットポリシーを削除する必要があります。ホームフォルダおよびアプリケーション設定の永続化用のバケットポリシーとして削除するものは、以下のとおりです。

ホームフォルダポリシー

```
{
```



```

"Version": "2012-10-17",
"Statement": [
  {
    "Sid": "PreventAccidentalDeletionOfBucket",
    "Effect": "Deny",
    "Principal": "*",
    "Action": "s3:DeleteBucket",
    "Resource": "arn:aws:s3:::appstream2-36fb080bb8-region-code-account-id-without-hyphens"
  }
]
}

```

アプリケーション設定の永続化ポリシー

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "PreventAccidentalDeletionOfBucket",
      "Effect": "Deny",
      "Principal": "*",
      "Action": "s3:DeleteBucket",
      "Resource": "arn:aws:s3:::appstream-app-settings-region-code-account-id-without-hyphens-unique-identifier"
    }
  ]
}

```

詳細については、Amazon Simple Storage Service ユーザーガイドの「[Deleting or Emptying a Bucket](#)」を参照してください。

ホームフォルダおよびアプリケーション設定の永続化用の Amazon S3 バケットへの管理者アクセスの制限

デフォルトでは、AppStream 2.0 で作成された Amazon S3 バケットにアクセスできる管理者は、ユーザーのホームフォルダと永続的なアプリケーション設定の一部のコンテンツを表示および変更できます。ユーザーファイルが含まれている S3 バケットへの管理者アクセスを制限するには、次のテンプレートに基づく S3 バケットアクセスポリシーを適用することをお勧めします。

```

{

```

```

{
  "Sid": "RestrictedAccess",
  "Effect": "Deny",
  "NotPrincipal": {
    "AWS": [
      "arn:aws:iam::account:role/service-role/AmazonAppStreamServiceAccess",
      "arn:aws:sts::account:assumed-role/AmazonAppStreamServiceAccess/PhotonSession",
      "arn:aws:iam::account:user/IAM-user-name"
    ]
  },
  "Action": "s3:*",
  "Resource": "arn:aws:s3::home-folder-or-application-settings-persistence-s3-bucket-region-account"
}

```

このポリシーでは、指定したユーザーと AppStream 2.0 サービスにのみ S3 バケットへのアクセスを許可します。アクセス権が必要な IAM ユーザーごとに、次の行をレプリケートします。

```
"arn:aws:iam::account:user/IAM-user-name"
```

次のポリシー例では、marymajor と johnstiles を除くすべての IAM ユーザーに対して、ホームフォルダの S3 バケットへのアクセスを制限します。また、アカウント ID 123456789012 の米国西部 (オレゴン) AWS リージョンの AppStream 2.0 サービスへのアクセスも許可します。

```

{
  "Sid": "RestrictedAccess",
  "Effect": "Deny",
  "NotPrincipal": {
    "AWS": [
      "arn:aws:iam::123456789012:role/service-role/AmazonAppStreamServiceAccess",
      "arn:aws:sts::123456789012:assumed-role/AmazonAppStreamServiceAccess/PhotonSession",
      "arn:aws:iam::123456789012:user/marymajor",
      "arn:aws:iam::123456789012:user/johnstiles"
    ]
  },
  "Action": "s3:*",
  "Resource": "arn:aws:s3:::appstream2-36fb080bb8-us-west-2-123456789012"
}

```

```
]
}
```

IAM ロールを使用して、AppStream 2.0 ストリーミングインスタンスで実行されるアプリケーションとスクリプトにアクセス許可を付与する

AppStream 2.0 ストリーミングインスタンスで実行されるアプリケーションとスクリプトは、AWS API リクエストに AWS 認証情報を含める必要があります。IAM ロールを作成して、これらの認証情報を管理できます。IAM ロールは、AWS リソースへのアクセスに使用できる一連のアクセス許可を指定します。ただし、このロールは 1 人のユーザーに一意に関連付けられるわけではありません。代わりに、それを必要とするすべてのユーザーが引き受けることができます。

IAM ロールを AppStream 2.0 ストリーミングインスタンスに適用できます。ストリーミングインスタンスがロールに切り替える (引き受ける) と、ロールは一時的なセキュリティ認証情報を提供します。アプリケーションまたはスクリプトはこれらの認証情報を使用して、ストリーミングインスタンスで API アクションおよび管理タスクを実行します。AppStream 2.0 は、一時的な認証情報スイッチを管理します。

内容

- [AppStream 2.0 ストリーミングインスタンスで IAM ロールを使用するためのベストプラクティス](#)
- [AppStream 2.0 ストリーミングインスタンスで使用するための既存の IAM ロールの設定](#)
- [AppStream 2.0 ストリーミングインスタンスで使用する IAM ロールを作成する方法](#)
- [AppStream 2.0 ストリーミングインスタンスで IAM ロールを使用する方法](#)

AppStream 2.0 ストリーミングインスタンスで IAM ロールを使用するためのベストプラクティス

AppStream 2.0 ストリーミングインスタンスで IAM ロールを使用する場合は、以下のプラクティスに従うことをお勧めします。

- AWS API アクションとリソースに付与するアクセス許可を制限します。

IAM ポリシーを作成し、AppStream 2.0 ストリーミングインスタンスに関連付けられた IAM ロールにアタッチするときは、最小限の権限の原則に従います。AWS API アクションまたはリソースへのアクセスを必要とするアプリケーションまたはスクリプトを使用する場合は、必要な特定のアクションとリソースを決定します。次に、アプリケーションまたはスクリプトがこれらのアク

シヨンのみを実行できるようにするポリシーを作成します。詳細については、「IAM ユーザーガイド」の「[Grant Least Privilege](#)」(最小権限を付与する)を参照してください。

- AppStream 2.0 リソースごとに IAM ロールを作成します。

AppStream 2.0 リソースごとに一意の IAM ロールを作成することは、最小権限の原則に従うプラクティスです。これにより、他のリソースに影響を与えることなく、リソースのアクセス許可を変更することもできます。

- 認証情報を使用できる場所を制限します。

IAM ポリシーでは、IAM ロールを使用してリソースにアクセスするための条件を定義できます。たとえば、リクエスト元の IP アドレスの範囲を指定する条件を含めることができます。これにより、認証情報が環境外で使用されなくなります。詳細については、IAM ユーザーガイドの「[追加セキュリティに対するポリシー条件を使用する](#)」を参照してください。

AppStream 2.0 ストリーミングインスタンスで使用するための既存の IAM ロールの設定

このトピックでは、Image Builder およびフリートストリーミングインスタンスでできるように、既存の IAM ロールを設定する方法について説明します。

前提条件

AppStream 2.0 Image Builder またはフリートストリーミングインスタンスで使用する IAM ロールは、以下の前提条件を満たす必要があります。

- IAM ロールは、AppStream 2.0 ストリーミングインスタンスと同じ Amazon Web Services アカウントに存在する必要があります。
- IAM ロールをサービスロールにすることはできません。
- IAM ロールにアタッチされた信頼関係ポリシーには、プリンシパルとして AppStream 2.0 サービスが含まれている必要があります。プリンシパルは、アクションを実行し AWS、リソースにアクセスできるのエンティティです。ポリシーには sts:AssumeRole アクションも含める必要があります。このポリシー設定は、AppStream 2.0 を信頼されたエンティティとして定義します。
- IAM ロールを Image Builder に適用する場合、Image Builder は 2019 年 9 月 3 日以降にリリースされたバージョンの AppStream 2.0 エージェントを実行する必要があります。フリートに IAM ロールを適用する場合、フリートは同じ日付以降にリリースされたバージョンのエージェントを使用するイメージを使用する必要があります。詳細については、「[AppStream 2.0 エージェントのリリースノート](#)」を参照してください。

AppStream 2.0 サービスプリンシパルが既存の IAM ロールを引き受けるようにするには

以下のステップを実行するには、IAM ロールを一覧表示および更新するために必要なアクセス許可を持つ IAM ユーザーとしてアカウントにサインインする必要があります。必要なアクセス許可がない場合は、お客様の Amazon Web Services アカウント管理者に対し、アカウントでこれらのステップを実行するか、必要なアクセス許可をお客様に付与するかのどちらかを依頼します。

1. IAM コンソール (<https://console.aws.amazon.com/iam/>) を開きます。
2. ナビゲーションペインで Roles (ロール) を選択してください。
3. アカウントのロールの一覧で、変更するロールの名前を選択します。
4. [Trust relationships] タブを選択し、続いて [Edit trust relationship] を選択します。
5. [Policy Document (ポリシードキュメント)] で、信頼関係ポリシーに `appstream.amazonaws.com` サービスプリンシパルの `sts:AssumeRole` アクションが含まれていることを確認します。

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

6. 信頼ポリシーの編集を完了したら、[信頼ポリシーの更新] を選択して変更を保存します。
7. 選択した IAM ロールが AppStream 2.0 コンソールに表示されます。このロールは、ストリーミングインスタンスで API アクションおよび管理タスクを実行するアクセス許可をアプリケーションとスクリプトに付与します。

AppStream 2.0 ストリーミングインスタンスで使用する IAM ロールを作成する方法

このトピックでは、Image Builder およびフリートストリーミングインスタンスで使えるように、新しい IAM ロールを作成する方法について説明します。

1. IAM コンソール (<https://console.aws.amazon.com/iam/>) を開きます。
2. ナビゲーションペインで [Roles] (ロール) を選択してから、[Create role] (ロールを作成する) を選択します。
3. 信頼できるエンティティの種類の選択 で、AWS サービス を選択します。
4. AWS サービスのリストから、AppStream 2.0 を選択します。
5. 「ユースケースの選択」で、AppStream 2.0 — AppStream 2.0 インスタンスがユーザーに代わって AWS サービスを呼び出すことを許可はすでに選択されています。[Next: Permissions] (次のステップ: 許可) を選択します。
6. 可能な場合は、アクセス許可ポリシーとして使用するポリシーを選択するか、[ポリシーの作成] を選択して新しいブラウザタブを開き、新しいポリシーをゼロから作成します。詳細については、IAM ユーザーガイドの「[IAM ポリシーの作成 \(コンソール\)](#)」のステップ 4 を参照してください。

ポリシーを作成したら、そのタブを閉じて元のタブに戻ります。AppStream 2.0 に割り当てるアクセス許可ポリシーの横にあるチェックボックスをオンにします。

7. (オプション) アクセス許可の境界を設定します。このアドバンスド機能は、サービスロールで使用できますが、サービスにリンクされたロールではありません。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。
8. [Next: Tags] (次へ: タグ) を選択します。オプションで、タグをキーと値のペアとしてアタッチできます。詳細については、IAM ユーザーガイドの「[IAM リソースのタグ付け](#)」を参照してください。
9. [次へ: レビュー] を選択します。
10. [Role name] (ロール名) に、Amazon Web Services アカウント内で一意のロール名を入力します。他の AWS リソースはロールを参照する可能性があるため、ロールの作成後にロールの名前を編集することはできません。
11. [ロールの説明] に、デフォルトのロールの説明をそのまま使用するか、新しいロールの説明を入力します。
12. ロールを確認したら、[ロールを作成] を選択します。

AppStream 2.0 ストリーミングインスタンスで IAM ロールを使用する方法

IAM ロールを作成したら、Image Builder を起動するとき、またはフリートを作成するとき、そのロールを Image Builder またはフリートストリーミングインスタンスに適用できます。既存のフリートに IAM ロールを適用することもできます。Image Builder を起動するとき IAM ロールを適用する方法については、[Image Builder を起動し、ストリーミングアプリケーションをインストール](#)

[して設定する](#) を参照してください。フリートを作成するときに IAM ロールを適用する方法については、[Amazon AppStream 2.0 でフリートを作成する](#) を参照してください。

Image Builder またはフリートストリーミングインスタンスに IAM ロールを適用すると、AppStream 2.0 は一時的な認証情報を取得し、インスタンスに [appstream_machine_role] 認証情報プロファイルを作成します。一時的な認証情報は 1 時間有効で、新しい認証情報は 1 時間ごとに取得されます。以前の認証情報は失効しないため、有効である限り使用できます。認証情報プロファイルを使用して、選択した言語で コマンドラインインターフェイス (AWS CLI)、AWS Tools for PowerShell、または AWS SDK AWS を使用してプログラムで AWS サービスを呼び出すことができます。

API コールを行う場合、認証情報プロファイルとして appstream_machine_role を指定します。それ以外の場合、アクセス許可が不十分なため、オペレーションは失敗します。

ストリーミングインスタンスがプロビジョニングされている間、AppStream 2.0 は指定されたロールを引き受けます。AppStream 2.0 は VPC にアタッチされている Elastic Network Interface を AWS API コールに使用するため、アプリケーションまたはスクリプトは、Elastic Network Interface が利用可能になるまで待ってから AWS API コールを行う必要があります。Elastic Network Interface が使用可能になる前に API 呼び出しが行われると、呼び出しは失敗します。

以下の例では、appstream_machine_role 認証情報プロファイルを使用して、ストリーミングインスタンス (EC2 インスタンス) を記述し、Boto クライアントを作成する方法を示します。Boto は、Amazon Web Services (AWS) SDK for Python です。

CLI を使用してストリーミングインスタンス (EC2 インスタンス) AWS を記述する

```
aws ec2 describe-instances --region us-east-1 --profile appstream_machine_role
```

AWS Tools for PowerShell を使用してストリーミングインスタンス (EC2 インスタンス) を記述する

AWS Tools for PowerShell バージョン 3.3.563.1 以降、Amazon Web Services SDK for .NET バージョン 3.3.103.22 以降を使用する必要があります。AWS Tools for PowerShell と Amazon Web Services SDK for .NET を含む AWS Tools for Windows インストーラは、[AWS Tools for PowerShell](#) ウェブサイトからダウンロードできます。

```
Get-EC2Instance -Region us-east-1 -ProfileName appstream_machine_role
```

AWS SDK for Python を使用した Boto クライアントの作成

```
session = boto3.Session(profile_name='appstream_machine_role')
```


Red Hat Enterprise Linux および Rocky Linux での SELinux

デフォルトでは、Security Enhanced Linux (SELinux) は `disabled` であり、Red Hat Enterprise Linux enabled および Rocky Linux を搭載した AppStream 2.0 Image Builder およびストリーミングインスタンスでは `enforcing` モードに設定されています。`enforcing` モードでは、アクセスの拒否が強制されます。SELinux は、カーネルの主要なサブシステムに強力な柔軟性のある強制アクセス制御 (MAC) アーキテクチャを提供するためのカーネル機能とユーティリティの集まりです。

SELinux は、機密性と完全性の要件に基づいて情報を分離するように強化されたメカニズムを備えています。このように情報を分離することで、アプリケーションのセキュリティメカニズムの改ざんやバイパスの脅威が軽減されます。また、悪意のあるアプリケーションや欠陥のあるアプリケーションによって引き起こされる可能性のあるダメージも回避できます。

SELinux には、日々のセキュリティの目標を満たすように設計されたサンプルセキュリティポリシー設定ファイルのセットが含まれています。SELinux の機能と機能の詳細については、「[SELinux とは](#)」を参照してください。

Amazon AppStream 2.0 での Cookie ベースの認証

AppStream 2.0 は、ブラウザの Cookie を使用してストリーミングセッションを認証し、ユーザーは毎回サインイン認証情報に再入力しなくても、アクティブなセッションに再接続できます。認証トークンは、すべての認証シナリオについてブラウザ Cookie に保存されます。Cookie は多くのオンラインサービスに必要ですが、Cookie 盗難攻撃に対して脆弱である可能性があります。ユーザーのデバイスに堅牢なエンドポイント保護ソリューションを実装するなど、Cookie の盗難を防ぐための予防的な対策を講じることを強くお勧めします。さらに、Cookie の盗難による潜在的な影響を軽減するために、以下のアクションを検討することをお勧めします。

- シングルセッション制限を適用する: AppStream 2.0 Windows イメージの場合、一度に 1 つの接続のみを許可するように、`max-concurrent-clients` という名前を 1 に設定して `HKEY_USERS \S-1-5-18\Software\GSettings\com\nicesoftware\dcv\session-management` にレジストリキーを作成します。これにより、同時セッションの数が 1 に制限され、アクティブなセッションのミラーリングがブロックされます。詳細については、「[session-management パラメータ](#)」を参照してください。
- セッションの有効期限と再認証を強制する
 - ユーザーがストリーミングセッションを正常に開始した後に認証トークンの有効期限が切れるように、`SessionDuration` 値を小さくします。`sessionDuration` の有効期限が切れた後に認証 Cookie を再利用するには、ユーザーが自身を再認証する必要があります。`SessionDuration` は、再認証が必要となるまでに、ユーザーのフェデレーティッドストリーミングセッションをアク

タイプにしておくことができる最大時間を指定します。デフォルト値は 60 分です。詳細については、「[the section called “ステップ 5: SAML 認証レスポンスのアサーションを作成する”](#)」を参照してください。

- セキュリティを最大化するために、ユーザーはストリーミングウィンドウを閉じるのではなく、ツールバー (セッションを終了) でセッションを適切に終了する必要があります。ツールバーを使用してセッションを終了すると、ユーザーセッションとストリーミングインスタンスの両方が終了します。そのためには、今後のアクセスのために再認証が必要であり、Cookie の誤用を防ぎます。ユーザーがセッションを終了せずにストリーミングウィンドウを閉じると、セッションとインスタンスは設定可能な切断タイムアウト期間 (分単位) の間アクティブのままになります。切断タイムアウトは 1~5760 の数値で、デフォルト値は 15 分である必要があります。非アクティブなセッションの誤用を防ぐため、短い切断タイムアウトを設定することをお勧めします。詳細については、「[the section called “フリートを作成する”](#)」を参照してください。
- AppStream 2.0 アプリケーションへのアクセスを IP 範囲に制限する: IP ベースの IAM ポリシーを実装することをお勧めします。これにより、AppStream 2.0 セッションには、IP アドレスが承認された IP 範囲に属するクライアントからのみアクセスできます。クライアントの IP アドレスが認可範囲外であるユーザーが開始した接続試行はすべて、それ以外の有効な認証 Cookie (ユーザーから盗まれた可能性がある) を提示している場合でも拒否されます。詳細については、「[Limit access to stream Amazon AppStream 2.0 applications to your IP ranges](#)」を参照してください。
- 追加の認証を追加する: ドメイン参加済みのストリーミングインスタンスを起動するには、AppStream 2.0 の常時オンおよびオンデマンド Windows フリートと Image Builder を Microsoft アクティブディレクトリのドメインに参加させ、既存のアクティブディレクトリドメイン (クラウドベースまたはオンプレミス) を使用します。最初の SAML ベースの認証後、ユーザーは組織ドメインに対する追加の認証のためにドメイン認証情報を提供するように求められます。詳細については、「[アクティブディレクトリの使用](#)」を参照してください。

心配なことやサポートが必要な場合は、[AWS サポート センター](#)にお問い合わせください。

Amazon AppStream 2.0 でのログ記録とモニタリング

モニタリングは、Amazon AppStream 2.0 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。このトピックでは、AppStream 2.0 リソースをモニタリングし、潜在的なインシデントに対応するために AWS が提供するサービスとツールについて説明します。

Amazon CloudWatch アラーム

Amazon CloudWatch アラームを使用して、指定した期間にわたって 1 つのメトリクスを確認できます。メトリクスが特定のしきい値を超えると、Amazon Simple Notification Service トピック

または AWS Auto Scaling ポリシーに通知が送信されます。CloudWatch アラームは、特定の状態にあるアクションを呼び出しません。その代わりに、状態が変更され、指定期間にわたって維持される必要があります。詳細については、「[Amazon AppStream 2.0 リソースのモニタリング](#)」を参照してください。

 Note

現在、AppStream 2.0 は CloudWatch Events のターゲットとして設定できません。CloudWatch Events のターゲットとして設定できるサービスのリストについては、[What Is Amazon CloudWatch Events](#) を参照してください。

AWS CloudTrail

AWS CloudTrail は、AppStream 2.0 のユーザー、ロール、または AWS サービスによって実行されたアクションの記録を提供します。これにより、AppStream 2.0 に対して行われたリクエスト、リクエスト元の IP アドレス、リクエストの実行者、リクエストの実行日時、および追加の詳細を判別できます。詳細については、「[を使用した AppStream 2.0 API コールのログ記録 AWS CloudTrail](#)」を参照してください。

AWS Trusted Advisor

AWS Trusted Advisor は AWS 環境を検査し、コスト削減、システムの可用性とパフォーマンスの向上、セキュリティギャップの解消に役立つ方法を推奨します。Trusted Advisor は、さまざまな AWS のお客様から収集されたベストプラクティスを使用します。すべての AWS お客様は 5 つの Trusted Advisor チェックにアクセスできます。ビジネスまたはエンタープライズサポートプランをお持ちの場合は、すべての Trusted Advisor チェックを表示できます。

ユーザーの[アプリケーション設定の永続化](#)または[ホームフォルダ](#)を有効にすると、ユーザーによって生成されたデータは Amazon S3 バケットに保存されます。Trusted Advisor には、Amazon S3 に関連する以下のチェックが含まれます。

- Amazon S3 バケットのログ記録設定のチェック。
- オープンなアクセス許可がある Amazon S3 バケットのセキュリティチェック。
- バージョニングが有効になっていない、またはバージョニングが停止されている Amazon S3 バケットの耐障害性チェック。

詳細については、AWS サポート ユーザーガイドの [AWS Trusted Advisor](#) を参照してください。

Amazon S3 アクセスログ

ユーザーがアプリケーション設定データまたはホームフォルダのデータを Amazon S3 バケットに保存している場合は、Amazon S3 サーバーアクセスログを表示してアクセスをモニタリングすることを検討してください。これらのログでは、バケットに対して行われたリクエストの詳細なレコードが提供されます。サーバーアクセスのログは、多くのアプリケーションに役立ちます。例えば、アクセスのログ情報は、セキュリティやアクセスの監査に役立ちます。詳細については、Amazon Simple Storage Service ユーザーガイドの「[Amazon S3 Server Access Logging](#)」を参照してください。

AppStream 2.0 使用状況レポート

AppStream 2.0 使用状況レポートをサブスクライブして、ユーザーがサービスをどのように使用しているかについて詳細なレポートを受け取ることができます。レポートには、ユーザーがストリーミングする時間と、ユーザーが起動するアプリケーションが含まれます。詳細については、「[AppStream 2.0 使用状況レポート](#)」を参照してください。

Amazon AppStream 2.0 のコンプライアンスの検証

サードパーティーの監査者は、複数のコンプライアンスプログラムの一環として Amazon AppStream 2.0 のセキュリティと AWS コンプライアンスを評価します。これには、[SOC](#)、[PCI](#)、[ISO](#)、[FedRAMP](#)、[HIPAA](#)、[MTCS](#)、[ENS High](#)、[HITRUST CSF](#)、[VPAT](#) などがあります。

Note

AppStream 2.0 は [FIPS 140-2](#) をサポートしています。管理用またはストリーミング用に AppStream 2.0 FIPS エンドポイントを使用する方法については、[the section called “FIPS エンドポイント”](#) を参照してください。

AppStream 2.0 は、[国防総省 \(DoD\) クラウドコンピューティングセキュリティ要求事項ガイド \(SRG\)](#) の評価も受けています。

特定のコンプライアンスプログラムの対象となる AWS サービスのリストについては、「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。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、サードパーティー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 AWS Artifact Reports](#)」を参照してください。

AppStream 2.0 を使用する場合のお客様のコンプライアンス責任は、データの機密性、企業のコンプライアンス目的、適用法規によって決まります。AWS は、コンプライアンスに役立つ次のリソースを提供しています。

- [「セキュリティ & コンプライアンス クイックリファレンス ガイド」](#) – これらのデプロイガイドには、アーキテクチャ上の考慮事項の説明と、AWS でセキュリティとコンプライアンスに重点を置いたベースライン環境をデプロイするための手順が記載されています。
- [「HIPAA のセキュリティとコンプライアンスのための設計」ホワイトペーパー](#) – このホワイトペーパーでは、企業が AWS を使用して HIPAA 準拠のアプリケーションを作成する方法について説明します。
- [AWS コンプライアンス リソース](#) – このワークブックとガイドのコレクションは、お客様の業界と場所に適用される場合があります。
- [「デベロッパーガイド」の「ルールによるリソースの評価」](#) – この AWS Config サービスは、リソース設定が社内プラクティス、業界ガイドライン、および規制にどの程度準拠しているかを評価します。AWS Config
- [AWS Security Hub](#) – この AWS サービスは、内のセキュリティ状態を包括的に把握し、セキュリティ業界標準とベストプラクティスへの準拠を確認するのに役立ちます。

Amazon AppStream 2.0 の耐障害性

AWS グローバルインフラストラクチャは、AWS リージョンとアベイラビリティーゾーンを中心に構築されています。リージョンには、低レイテンシー、高いスループット、そして高度の冗長ネットワークで接続されている複数の物理的に独立および隔離されたアベイラビリティーゾーンがあります。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イ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が高く、フォールトトレラントで、スケーラブルです。

AWS リージョンと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。

Amazon AppStream 2.0 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 AppStream 2.0 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 AWS 環境を設計するには、「セキュリティの柱 AWS Well-Architected Framework」の「[Infrastructure Protection](#)」を参照してください。

AWS が公開した API コールを使用して、ネットワーク経由で AppStream 2.0 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。

また、リクエストにはアクセスキー ID と、IAM プリンシパルに関連付けられているシークレットアクセスキーを使用して署名する必要があります。または[AWS Security Token Service](#) (AWS STS) を使用して、一時的なセキュリティ認証情報を生成し、リクエストに署名することもできます。

次のトピックでは、AppStream 2.0 インフラストラクチャセキュリティに関する追加情報を提供します。

目次

- [ネットワークの隔離](#)
- [物理ホストでの分離](#)
- [ネットワークトラフィックの制御](#)
- [AppStream 2.0 インターフェイス VPC エンドポイント](#)
- [FIPS エンドポイントを使用した転送中のデータの保護](#)

ネットワークの隔離

Virtual Private Cloud (VPC) は、Amazon Web Services クラウド内の論理的に隔離された領域にある仮想ネットワークです。ワークロードまたは組織エンティティ単位でインフラストラクチャを隔離するには、個別の VPC を使用します。

サブネットはある範囲の IP アドレスが示す VPC 内の領域です。インスタンスを起動する場合には VPC 内のあるサブネットにおいて起動することになります。サブネットを使用すると、単一の VPC 内で多階層ウェブアプリケーションの各階層 (ウェブサーバー、アプリケーションサーバーおよびデータベースサーバーなど) を隔離できます。インターネットからの直接アクセスを認めるべきでないインスタンスには、プライベートサブネットを使用します。

パブリックインターネットを経由せずに、VPC 内の AppStream 2.0 ストリーミングインスタンスからストリーミングできます。これを行うには、インターフェイス VPC エンドポイント (インターフェイスエンドポイント) を使用します。詳細については、[チュートリアル: インターフェイス VPC エンドポイントからの作成とストリーミング](#) を参照してください。

また、インターフェイスエンドポイントを使用して、パブリックインターネット経由でトラフィックを送信せずに VPC から AppStream 2.0 API オペレーションを呼び出すことができます。詳細については、[インターフェイス VPC エンドポイント経由で AppStream 2.0 API オペレーションおよび CLI コマンドにアクセスする](#) を参照してください。

物理ホストでの分離

同じ物理ホストで実行される異なるストリーミングインスタンスは、個別の物理ホストで実行されるかのように隔離されます。ハイパーバイザーが CPU およびメモリを隔離し、各インスタンスには、生ディスクデバイスへのアクセスに代わる仮想ディスクへのアクセスが提供されます。

ストリーミングインスタンスを停止または終了すると、そのストリーミングインスタンスに割り当てられていたメモリをハイパーバイザーがスクラブ (ゼロに設定するということです) し、そのメモリが新たなインスタンスに割り当てられ、すべてのストレージブロックがリセットされます。これにより、データが別のインスタンスに公開されることがなくなります。

ネットワークトラフィックの制御

AppStream 2.0 ストリーミングインスタンスへのネットワークトラフィックを制御するには、次のオプションを検討してください。

- Amazon AppStream ストリーミングインスタンスを起動する場合には、VPC 内のあるサブネットにおいて起動することになります。インターネットからアクセスできないようにするには、プライベートサブネットにストリーミングインスタンスをデプロイします。
- プライベートサブネットのストリーミングインスタンスへのインターネットアクセスを提供するには、NAT ゲートウェイを使用します。詳細については、[プライベートサブネットの VPC および NAT ゲートウェイを設定する](#) を参照してください。
- VPC に属するセキュリティグループにより、AppStream 2.0 ストリーミングインスタンスと、ライセンスサーバー、ファイルサーバー、データベースサーバーなどの VPC リソースの間でネットワークトラフィックを制御できます。また、セキュリティグループは、ストリーミングインスタンスと AppStream 2.0 管理サービス間のトラフィックを分離します。

セキュリティグループを使用して、ストリーミングインスタンスへのアクセスを制限します。この方法を使うと、たとえば、社内ネットワークのアドレス範囲に属するアドレスからのトラフィック

のみ認めるといったことができます。詳細については、[Amazon AppStream 2.0 のセキュリティグループ](#) を参照してください。

- パブリックインターネットを経由せずに、VPC 内の AppStream 2.0 ストリーミングインスタンスからストリーミングできます。これを行うには、インターフェイス VPC エンドポイント (インターフェイスエンドポイント) を使用します。詳細については、[チュートリアル: インターフェイス VPC エンドポイントからの作成とストリーミング](#) を参照してください。

また、インターフェイスエンドポイントを使用して、パブリックインターネット経由でトラフィックを送信せずに VPC から AppStream 2.0 API オペレーションを呼び出すことができます。詳細については、[インターフェイス VPC エンドポイント経由で AppStream 2.0 API オペレーションおよび CLI コマンドにアクセスする](#) を参照してください。

- IAM ロールとポリシーを使用して、AppStream 2.0、Application Auto Scaling、および Amazon S3 バケットへの管理者アクセスを管理します。詳細については、次のトピックを参照してください。
 - [AWS 管理ポリシーとリンクされたロールを使用して AppStream 2.0 リソースへの管理者アクセスを管理する](#)
 - [IAM ポリシーを使用して Application Auto Scaling への管理者アクセスを管理する](#)
 - [ホームフォルダおよびアプリケーション設定の永続化用の Amazon S3 バケットへの管理者アクセスの制限](#)
- SAML 2.0 を使用して、認証を AppStream 2.0 にフェデレートできます。詳細については、[Amazon AppStream 2.0 Service Quotas](#) を参照してください。

Note

小規模な AppStream 2.0 デプロイの場合は、AppStream 2.0 ユーザープールを使用できます。デフォルトでは、ユーザープールは最大 50 人のユーザーをサポートします。AppStream 2.0 クォータ (制限とも呼ばれる) の詳細については、[Amazon AppStream 2.0 Service Quotas](#) を参照してください。100 人以上の AppStream 2.0 ユーザーをサポートする必要があるデプロイメントの場合は、SAML 2.0 を使用することをお勧めします。

AppStream 2.0 インターフェイス VPC エンドポイント

Virtual Private Cloud (VPC) は、Amazon Web Services クラウド内の論理的に隔離された領域にある仮想ネットワークです。Amazon Virtual Private Cloud を使用して AWS リソースをホストする場合は、VPC と AppStream 2.0 の間にプライベート接続を確立できます。この接続を使用すると、AppStream 2.0 はパブリックインターネットを経由せずに、VPC のリソースと通信できます。

インターフェイスエンドポイントは、プライベート IP アドレスを使用して指定した VPC 内でトラフィックをストリーミングし続けることができるテクノロジーである AWS PrivateLink を利用しています。AWS Direct Connect または AWS Virtual Private Network トンネルで VPC を使用すると、ストリーミングトラフィックをネットワーク内に保持できます。

次のトピックでは、AppStream 2.0 インターフェイスエンドポイントについて説明します。

内容

- [チュートリアル: インターフェイス VPC エンドポイントからの作成とストリーミング](#)
- [インターフェイス VPC エンドポイント経由で AppStream 2.0 API オペレーションおよび CLI コマンドにアクセスする](#)

チュートリアル: インターフェイス VPC エンドポイントからの作成とストリーミング

Amazon Web Services アカウントのインターフェイス VPC エンドポイントを使用することで、Amazon VPC と AppStream 2.0 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できます。このエンドポイントを作成したら、これを使用するために AppStream 2.0 スタックまたはイメージビルダーを設定します。

前提条件

AppStream 2.0 用のインターフェイス VPC エンドポイントを設定する前に、以下の前提条件に注意してください。

- ユーザーを認証し、AppStream 2.0 が機能するために必要なウェブアセットを配信するためには、インターネットに接続できることが必須です。ストリーミングインターフェイスエンドポイントは、VPC 内のストリーミングトラフィックを維持します。ストリーミングトラフィックには、ピクセル、USB、ユーザー入力、オーディオ、クリップボード、ファイルのアップロードとダウンロード、プリンターのトラフィックが含まれます。このトラフィックを許可するには、「[許可されたドメイン](#)」に示されたドメインを許可する必要があります。VPC エンドポイントを作成したら、AppStream 2.0 ユーザー認証ドメインを許可する必要があります。ただし、ストリーミングゲートウェイでは、アクセスを <vpc-endpoint-id>.streaming.appstream.<aws-region>.vpce.amazonaws.com のみに制限できます。* への出品を許可します。amazonappstream.com://www.com.」は必要ありません。VPC エンドポイントの完全修飾ドメイン名は、その依存関係を置き換えます。
- ユーザーのデバイスが接続されているネットワークは、インターフェイスエンドポイントにトラフィックをルーティングできる必要があります。

- インターフェイスエンドポイントに関連付けられているセキュリティグループは、ユーザーが接続する IP アドレス範囲からポート 443(TCP) とポート 1400 ~ 1499 (TCP) へのインバウンドアクセスを許可する必要があります。
- サブネットのネットワークアクセスコントロールリストでは、一時ネットワークポート 1024 ~ 65535 (TCP) から、ユーザーが接続する IP アドレス範囲へのアウトバウンドトラフィックを許可する必要があります。
- API `ec2:DescribeVpcEndpoints` アクションを実行するためのアクセス許可 AWS アカウントを提供する IAM アクセス許可ポリシーがある必要があります。デフォルトでは、このアクセス許可は `AmazonAppStreamServiceAccess` ロールにアタッチされている IAM ポリシーで定義されます。必要なアクセス許可がある場合、AWS リージョンで AppStream 2.0 サービスを開始すると、このサービスロールは AppStream 2.0 によって自動的に作成され、必要な IAM ポリシーがアタッチされます。詳細については、「[Amazon AppStream 2.0 向け Identity and Access Management](#)」を参照してください。

インターフェイスエンドポイントを作成するには

1. Amazon VPC コンソールの <https://console.aws.amazon.com/vpc/> を開いてください。
2. ナビゲーションペインで、[エンドポイント]、[エンドポイントを作成] の順に選択します。
3. [Create Endpoint] (エンドポイントの作成) を選択します。
4. サービスカテゴリで、AWS サービスが選択されていることを確認します。
5. [Service Name] (サービス名)には **com.amazonaws.<AWS #####>.appstream.streaming** を選択します。
6. 以下の情報を指定します。終了したら、[Create Endpoint] を選択します。
 - [VPC] で、インターフェイスエンドポイントを作成する VPC を選択します。AppStream 2.0 リソースを持つ VPC とは異なる VPC を選択できます。
 - [サブネット] で、エンドポイントネットワークインターフェイスを作成する先のサブネット (アベイラビリティーゾーン) を選択します。少なくとも 2 つのアベイラビリティーゾーンでサブネットを選択することをお勧めします。
 - [Enable Private DNS Name] チェックボックスが選択されていることを確認します。

 Note

ユーザーがネットワークプロキシを使用してストリーミングインスタンスにアクセスする場合は、プライベートエンドポイントに関連付けられているドメインと DNS 名

のプロキシキャッシュを無効にします。VPC エンドポイントの DNS 名は、プロキシを介して許可する必要があります。

- [Security group] で、エンドポイントネットワークインターフェイスに関連付けるセキュリティグループを選択します。

 Note

セキュリティグループは、ユーザーが接続する IP アドレス範囲からポートへのインバウンドアクセスを提供する必要があります。

インターフェイスエンドポイントの作成中、コンソールのエンドポイントのステータスは、[Pending] と表示されます。エンドポイントが作成されると、ステータスは [Available] に変わります。

ストリーミングセッション用に作成したインターフェイスエンドポイントを使用するようにスタックを更新するには、次のステップを実行します。

新しいインターフェイスエンドポイントを使用するようスタックを更新するには

1. AppStream 2.0 コンソールを <https://console.aws.amazon.com/appstream2> で開きます。

使用するインターフェイスエンドポイントと同じ AWS リージョンでコンソールを開いてください。

2. ナビゲーションペインで [Stacks] を選択し、希望するスタックを選択します。
3. [VPC Endpoints (VPC エンドポイント)] タブを選択し、[Edit (編集)] を選択します。
4. [Edit VPC Endpoint (VPC エンドポイントの編集)] ダイアログボックスの [Streaming Endpoint (ストリーミングエンドポイント)] で、ストリーミングトラフィックを介するエンドポイントを選択します。
5. [更新] を選択します。

新しいストリーミングセッションのトラフィックは、このエンドポイントを介してルーティングされます。ただし、現在のストリーミングセッションのトラフィックは、引き続き以前に指定したエンドポイントを介してルーティングされます。

Note

インターフェイスエンドポイントが指定されている場合、ユーザーはインターネットエンドポイントを使用してストリーミングできません。

インターフェイス VPC エンドポイント経由で AppStream 2.0 API オペレーションおよび CLI コマンドにアクセスする

Amazon Virtual Private Cloud を使用して AWS リソースをホストする場合、インターネット経由で接続する代わりに、仮想プライベートクラウド (VPC) の [インターフェイス VPC エンドポイント](#) (インターフェイスエンドポイント) を介して AppStream 2.0 API オペレーションまたはコマンドラインインターフェイス (CLI) コマンドに直接接続できます。インターフェイスエンドポイントは、プライベート IP アドレスを使用して指定した VPC 内でトラフィックをストリーミングできるテクノロジーである AWS PrivateLink を利用しています。インターフェイスエンドポイントを使用すると、VPC と AppStream 2.0 間の通信は AWS、ネットワーク内で完全かつ安全に実施されます。

Note

このトピックでは、インターフェイスエンドポイント経由で AppStream 2.0 API オペレーションおよび CLI コマンドにアクセスする方法を説明します。AppStream 2.0 インターフェイスエンドポイントを作成し、そこからストリーミングする方法については、[チュートリアル: インターフェイス VPC エンドポイントからの作成とストリーミング](#) を参照してください。

前提条件

インターフェイスエンドポイントを使用するには、次の前提条件を満たす必要があります。

- インターフェイスエンドポイントに関連付けられているセキュリティグループは、ユーザーが接続する IP アドレス範囲からポート 443(TCP) のインバウンドアクセスを許可する必要があります。
- サブネットのネットワークアクセスコントロールリストでは、一時ネットワークポート 1024 ~ 65535 (TCP) から、ユーザーが接続する IP アドレス範囲へのアウトバウンドトラフィックを許可する必要があります。

トピック

- [インターフェイスエンドポイントを作成して AppStream 2.0 API オペレーションおよび CLI コマンドにアクセスする](#)
- [インターフェイスエンドポイントを使用して AppStream 2.0 API オペレーションおよび CLI コマンドにアクセスする](#)

インターフェイスエンドポイントを作成して AppStream 2.0 API オペレーションおよび CLI コマンドにアクセスする

インターフェイスエンドポイントを作成するには、次のステップを実行します。

1. Amazon VPC コンソールの <https://console.aws.amazon.com/vpc/> を開いてください。
2. ナビゲーションペインで、[エンドポイント]、[エンドポイントを作成] の順に選択します。
3. [Create Endpoint] (エンドポイントの作成) を選択します。
4. サービスカテゴリで、AWS サービスが選択されていることを確認します。
5. [Service Name] (サービス名) には **com.amazonaws.<AWS #####>.appstream.api** を選択します。
6. 以下の情報を指定します。終了したら、[Create Endpoint] を選択します。
 - [VPC] で、インターフェイスエンドポイントを作成する VPC を選択します。
 - [サブネット] は、エンドポイントネットワークインターフェイスを作成する先の、サブネット (アベイラビリティーゾーン) を選択します。少なくとも 2 つのアベイラビリティーゾーンでサブネットを選択することをお勧めします。
 - オプションで、[Enable Private DNS Name] チェックボックスを選択できます。

 Note

このオプションを選択する場合は、必要に応じて、プライベート DNS をサポートするように VPC および DNS を必ず設定してください。詳細については、Amazon VPC ユーザーガイドの [Private DNS](#) を参照してください。

- [Security group] (セキュリティグループ) で、エンドポイントネットワークインターフェイスに関連付けるセキュリティグループを選択します。

 Note

セキュリティグループは、ユーザーが接続する IP アドレス範囲からポートへのインバウンドアクセスを提供する必要があります。

インターフェイスエンドポイントの作成中、コンソールのエンドポイントのステータスは、[Pending] と表示されます。エンドポイントが作成されると、ステータスは [Available] に変わります。

インターフェイスエンドポイントを使用して AppStream 2.0 API オペレーションおよび CLI コマンドにアクセスする

作成した VPC エンドポイントのステータスが [Available (利用可能)] に変更されたら、エンドポイントを使用して AppStream 2.0 API オペレーションおよび CLI コマンドにアクセスできます。これを実行するには、これらのオペレーションおよびコマンドを使用するときに、インターフェイスエンドポイントの DNS 名がある `endpoint-url` パラメータを指定します。DNS 名はパブリックに解決可能ですが、VPC 内のトラフィックのみを正常にルーティングします。

次の例は、`describe-fleets` CLI コマンドを使用するときにインターフェイスエンドポイントの DNS 名を指定する方法を示しています。

```
aws appstream describe-fleets --endpoint-url <vpc-endpoint-id>.api.appstream.<aws-region>.vpce.amazonaws.com
```

次の例は、AppStream 2.0 Boto3 Python クライアントをインスタンス化するときにインターフェイスエンドポイントの DNS 名を指定する方法を示しています。

```
appstream2client = boto3.client('appstream', region_name='<aws-region>', endpoint_url='<vpc-endpoint-id>.api.appstream.<aws-region>.vpce.amazonaws.com')
```

`appstream2client` オブジェクトを使用する後続のコマンドでは、指定したインターフェイスエンドポイントが自動的に使用されます。

インターフェイスエンドポイントのプライベート DNS ホスト名を有効にした場合は、エンドポイント URL を指定する必要はありません。AppStream 2.0 API および CLI がデフォルトで使用する API DNS ホスト名。VPC 内で解決されます。プライベート DNS ホスト名の詳細については、Amazon VPC ユーザーガイドの [Private DNS](#) を参照してください。

FIPS エンドポイントを使用した転送中のデータの保護

デフォルトでは、AppStream 2.0 サービスと通信するとき、管理者として AppStream 2.0 コンソール、AWS コマンドラインインターフェイス (AWS CLI)、または AWS SDK を使用するか、Image Builder またはフリートインスタンスからストリーミングするユーザーとして、転送中のすべてのデータは TLS 1.2 を使用して暗号化されます。

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 140-2 検証済みの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。AppStream 2.0 は、AppStream 2.0 が利用可能なすべての米国 AWS リージョンで FIPS エンドポイントを提供します。FIPS エンドポイントを使用する場合、転送中のすべてのデータは、連邦情報処理標準 (FIPS)140-2 に準拠する暗号化標準を使用して暗号化されます。AppStream 2.0 エンドポイントのリストなど、FIPS エンドポイントの詳細については、[Federal Information Processing Standard \(FIPS\) 140-2](#) を参照してください。

トピック

- [管理用の FIPS エンドポイント](#)
- [ユーザーストリーミングセッションの FIPS エンドポイント](#)
- [例外](#)

管理用の FIPS エンドポイント

AppStream 2.0 の AWS CLI コマンドを実行するときに FIPS エンドポイントを指定するには、`endpoint-url`パラメータを使用します。以下の例では、米国西部 (オレゴン) リージョンの AppStream 2.0 FIPS エンドポイントを使用して、そのリージョン内のすべてのスタックのリストを取得します。

```
aws appstream describe-stacks --endpoint-url https://appstream2-fips.us-west-2.amazonaws.com
```

AppStream 2.0 API オペレーションの FIPS エンドポイントを指定するには、AWS SDK の手順を使用してカスタムエンドポイントを指定します。

ユーザーストリーミングセッションの FIPS エンドポイント

SAML 2.0 またはストリーミング URL を使用してユーザーを認証する場合は、ユーザーのストリーミングセッション用に FIPS 準拠の接続を設定できます。

SAML 2.0 を使用して認証するユーザーに FIPS 準拠の接続を使用するには、フェデレーションのリリースステートを設定するときに AppStream 2.0 FIPS エンドポイントを指定します。SAML 2.0 を使用した ID フェデレーション用のリリースステート URL の生成の詳細については、[SAML のセットアップ](#) を参照してください。

ストリーミング URL を介して認証するユーザーの FIPS 準拠の接続を設定するには、CLI AWS または SDK から AppStream または [CreateImageBuilderStreamingURL](#) オペレーションを呼び出すときに AppStream 2.0 FIPS AWS エンドポイントを指定します。[CreateStreamingURL](#) 生成された URL を使用してストリーミングインスタンスに接続するユーザーは、FIPS 準拠の接続を使用します。以下の例では、米国東部 (バージニア) リージョンの AppStream 2.0 FIPS エンドポイントを使用して、FIPS 準拠のストリーミング URL を生成します。

```
aws appstream create-streaming-url --stack-name stack-name --fleet-name fleet-name --  
user-id user-id --endpoint-url https://appstream2-fips.us-east-1.amazonaws.com
```

例外

FIPS 準拠の接続は以下のシナリオではサポートされません。

- AppStream 2.0 コンソール経由の AppStream 2.0 の管理
- AppStream 2.0 ユーザープール機能を使用して認証するユーザーのストリーミングセッション
- インターフェイス VPC エンドポイントを使用したストリーミング
- AppStream 2.0 コンソールからの FIPS 準拠のストリーミング URL の生成
- ストレージプロバイダーが FIPS エンドポイントを提供しない、Google Drive または OneDrive ストレージアカウントへの接続

Amazon AppStream 2.0 のセキュリティグループ

VPC セキュリティグループと関連付けることで、フリートのストリーミングインスタンスまたは Amazon AppStream 2.0 の Image Builder から VPC に追加のアクセスコントロールを提供できます。VPC に属するセキュリティグループにより、AppStream 2.0 ストリーミングインスタンスと、ライセンスサーバー、ファイルサーバー、データベースサーバーなどの VPC リソースの間でネットワークトラフィックを制御できます。詳細については、Amazon VPC ユーザーガイドの「[VPC のセキュリティグループ](#)」を参照してください。

セキュリティグループがフリートまたは Image Builder に関連付けられている場合、VPC セキュリティグループに対して定義するルールが適用されます。セキュリティグループのルールにより、ス

ストリーミングインスタンスから許可されるネットワークトラフィックが決まります。詳細については、Amazon VPC ユーザーガイドの [Security Group Rules](#) を参照してください。

新しい Image Builder の起動中、または新しいフリートの作成中に、最大 5 つのセキュリティグループを関連付けることができます。セキュリティグループを既存のフリートに関連付けるか、フリートのセキュリティグループを変更することもできます (フリートのセキュリティグループを変更するには、フリートを停止する必要があります)。詳細については、「Amazon VPC ユーザーガイド」の「[セキュリティグループの操作](#)」を参照してください。

セキュリティグループを選択しないと、Image Builder またはフリートは VPC のデフォルトのセキュリティグループに関連付けられます。詳細については、Amazon VPC ユーザーガイドの「[VPC のデフォルトセキュリティグループ](#)」を参照してください。

AppStream 2.0 でセキュリティグループを使用する際は、以下の追加の考慮事項に注意してください。

- インターネットトラフィック、ホームフォルダのデータ、VPC リソースとのアプリケーションの通信など、すべてのエンドユーザーデータは、ストリーミングインスタンスに関連付けられたセキュリティグループから影響を受けます。
- ストリーミングピクセルデータはセキュリティグループの影響を受けません。
- フリートまたは Image Builder に対してデフォルトのインターネットアクセスを有効にしている場合、関連付けられたセキュリティグループのルールで、インターネットアクセスを許可する必要があります。

Amazon VPC コンソールを使用して、セキュリティグループのルールを作成または編集したり、新しいセキュリティグループを作成したりできます。

- セキュリティグループを Image Builder に関連付けるには、[Image Builder を起動し、ストリーミングアプリケーションをインストールして設定する](#) の手順に従います。
- フリートにセキュリティグループを関連付けるには
 - While creating the fleet (フリートの作成中) — [Amazon AppStream 2.0 でフリートを作成する](#) の手順に従います。
 - 既存のフリートの場合 — AWS Management Consoleを使用してフリート設定を編集します。

AWS CLI および SDKs を使用して、セキュリティグループをフリートに関連付けることもできます。

- AWS CLI — [create-fleet](#) コマンドと [update-fleet](#) コマンドを使用します。
- AWS SDKs [CreateFleet](#) および [UpdateFleet](#) API オペレーションを使用します。

詳細については、[AWS Command Line Interface ユーザーガイド](#)および「[Tools for Amazon Web Services](#)」を参照してください。

Amazon AppStream 2.0 での更新管理

AppStream 2.0 には、Image Builder をより新しい AppStream 2.0 エージェントソフトウェアで自動的に更新する方法が用意されています。常に最新バージョンの AppStream 2.0 エージェントを使用するようにイメージを設定すると、ストリーミングインスタンスは自動的に更新されて、AWSで利用可能な最新の機能、機能強化、およびセキュリティ更新プログラムが反映されます。AppStream 2.0 エージェントのバージョンを管理する方法については、[AppStream 2.0 エージェントのバージョンを管理する](#) を参照してください。

Windows オペレーティングシステムの更新プログラム、お客様のアプリケーション、それらの依存関係のインストールとメンテナンスは、お客様の責任で行います。詳細については、「[Amazon AppStream 2.0 イメージを最新の状態に保つ](#)」を参照してください。

AppStream 2.0 イメージは、AppStream 2.0 マネージドイメージ更新を使用して最新の状態に保つことができます。この更新機能では、最新の Windows オペレーティングシステムの更新とドライバーの更新、および最新の AppStream 2.0 エージェントソフトウェアが提供されます。詳細については、「[AppStream 2.0 のマネージド型イメージアップデートを使用してイメージを更新する](#)」を参照してください。

ストリーミングインスタンス上のアプリケーションの更新を管理するには、提供される自動更新サービスを使用できます。また、アプリケーションベンダーが提供する更新プログラムをインストールするための推奨事項に従うこともできます。

Amazon AppStream 2.0 のサービス間の混乱した代理の防止

混乱した代理問題とは、アクションを実行する許可を持たないエンティティが、より高い特権を持つエンティティにそのアクションの実行を強制するというセキュリティ問題です。AWSでは、サービス間でのなりすましにより、混乱した代理問題に対してアカウントリソースが脆弱になることがあります。サービス間でのなりすましは、1つのサービス(呼び出し元サービス)が、別のサービス(呼び出されたサービス)を呼び出すときに発生します。呼び出し元サービスが呼び出されたサービスを操

作し、そのアクセス許可を使用して、呼び出し元サービスが自身で実行するアクセス許可を持っていない方法で、顧客のリソースに対して処理を実行する可能性があります。これを防ぐために、は、アカウント内のリソースにアクセスできるサービスプリンシパルを使用して、すべてのサービスのデータを保護するのに役立つツール AWS を提供します。

リソースポリシーで `aws:SourceArn` および `aws:SourceAccount` のグローバル条件コンテキストキーを使用して、それらのリソースにアクセスするときにアクセス許可を制限することをお勧めします。以下のガイドラインでは、これらのキーを使用してリソースを保護する場合の推奨事項と要件を詳しく説明しています。

- クロスサービスのアクセスにリソースを 1 つだけ関連付けたい場合は、`aws:SourceArn` を使用します。
- 指定されたアカウント内の任意のリソースを、クロスサービスによる使用に関連付ける場合は、`aws:SourceAccount` を使用します。
- `aws:SourceArn` キーにアカウント ID が含まれていない場合、アクセス許可を制限するためには、これら両方のグローバル条件コンテキストキー (`aws:SourceArn` および `aws:SourceAccount`) を使用する必要があります。
- 両方のグローバル条件コンテキストキーを使用しており、`aws:SourceArn` の値にアカウント ID が含まれる場合、それらが同じポリシーステートメントで使用されるときは、`aws:SourceAccount` キーは同じアカウント ID を使用する必要があります。

混乱した代理問題を回避するための最も効果的な方法は、許可するリソースに正確な Amazon リソースネーム (ARN) を使用することです。リソースの完全な ARN が不明な場合は、グローバルコンテキスト条件キー `aws:SourceArn` で、ARN の不明な部分を示すためにワイルドカード (* など) を使用します。複数のリソースを指定する場合は、ARN でワイルドカードを使用することもできます。例えば、ARN を `arn:aws:servicename::region-name::your AWS ##### ID:*` のようにフォーマットできます。

トピック

- [例: AppStream 2.0 サービスロールにおけるサービス間の混乱した代理の防止](#)
- [例: AppStream 2.0 フリートマシンロールにおけるサービス間の混乱した代理の防止](#)
- [例: AppStream 2.0 Elastic Fleets セッションスクリプトの Amazon S3 バケットポリシーにおけるサービス間の混乱した代理の防止](#)
- [例: AppStream 2.0 Application の Amazon S3 バケットポリシーにおけるサービス間の混乱した代理の防止](#)

例: AppStream 2.0 サービスロールにおけるサービス間の混乱した代理の防止

AppStream 2.0 では、さまざまなリソース ARN を使用するサービスロールを想定しているため、複雑な条件文になります。AppStream 2.0 リソースの予期しない障害を防ぐため、ワイルドカードリソースタイプを使用することをお勧めします。

Example **aws:SourceAccount** 条件付き:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": [
          "appstream.amazonaws.com"
        ]
      },
      "Action": "sts:AssumeRole",
      "Condition": {
        "StringEquals": {
          "aws:SourceAccount": "your AWS ##### ID"
        }
      }
    }
  ]
}
```

Example **aws:SourceArn** 条件付き:

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

```

        "Condition": {
            "ArnLike": {
                "aws:SourceArn": "arn:{aws partition}:appstream:{your region name}:
{your AWS ##### ID}:*"
            }
        }
    ]
}

```

例: AppStream 2.0 フリートマシンロールにおけるサービス間の混乱した代理の防止

Example **aws:SourceAccount** 条件付き:

```

{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
                "Service": [
                    "appstream.amazonaws.com"
                ]
            },
            "Action": "sts:AssumeRole",
            "Condition": {
                "StringEquals": {
                    "aws:SourceAccount": "your AWS ##### ID"
                }
            }
        }
    ]
}

```

Example **aws:SourceArn** 条件付き:

 Note

複数のフリートで 1 つの IAM ロールを使用する場合は、ワイルドカード (*) 付きの **aws:SourceArn** グローバルコンテキスト条件キーを使用して、複数の AppStream 2.0 フリートリソースと一致させることをお勧めします。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": [
          "appstream.amazonaws.com"
        ]
      },
      "Action": "sts:AssumeRole",
      "Condition": {
        "ArnLike": {
          "aws:SourceArn": "arn:{aws partition}:appstream:{your region name}:
{your AWS ##### ID}:fleet/{your fleet name}"
        }
      }
    }
  ]
}
```

例: AppStream 2.0 Elastic Fleets セッションスクリプトの Amazon S3 バケットポリシーにおけるサービス間の混乱した代理の防止

Example **aws:SourceAccount** 条件付き:

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

```

    "Principal": {
      "Service": [
        "appstream.amazonaws.com"
      ]
    },
    "Action": "s3:GetObject",
    "Resource": "your session script S3 path",
    "Condition": {
      "StringEquals": {
        "aws:SourceAccount": "your AWS ##### ID"
      }
    }
  }
]
}

```

Example **aws:SourceArn** 条件付き:

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": [
          "appstream.amazonaws.com"
        ]
      },
      "Action": "s3:GetObject",
      "Resource": "your session script S3 path",
      "Condition": {
        "ArnLike": {
          "aws:SourceArn": "arn:{aws partition}:appstream:{your region name}:  

{your AWS ##### ID}:fleet/{your fleet name}"
        }
      }
    }
  ]
}

```

例: AppStream 2.0 Application の Amazon S3 バケットポリシーにおけるサービス間の混乱した代理の防止

Amazon S3 バケットにデータを保存すると、バケットで混乱した代理問題が発生する可能性があります。これにより、Elastic フリート、App Block、セットアップスクリプト、アプリケーションアイコン、セッションスクリプトなどのデータが悪意のある攻撃者に対して脆弱になる可能性があります。

混乱した代理問題を防ぐため、ELASTIC-FLEET-EXAMPLE-BUCKET の Amazon S3 バケットポリシーで、aws:SourceAccount 条件または aws:SourceArn 条件を指定できます。

以下のリソースポリシーは、以下のいずれかの問題に関する混乱した代理問題を防止する方法を説明しています。

- AWS アカウント ID aws:SourceAccount を持つ
- グローバル条件コンテキストキー aws:SourceArn

AppStream 2.0 は現在、アプリケーションアイコンの混乱した代理の防止をサポートしていません。このサービスは VHD ファイルとセットアップスクリプトのみをサポートします。アプリケーションアイコンに条件を追加しようとしても、アイコンはエンドユーザーに表示されません。

次の例では、バケットポリシーは、所有者のアカウントの AppStream 2.0 Elastic フリートリソースにのみ、ELASTIC_FLEET_EXAMPLE_BUCKET へのアクセスを許可します。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "ConfusedDeputyPreventionExamplePolicy",
      "Effect": "Allow",
      "Principal": {
        "Service": "appstream.amazonaws.com"
      },
      "Action": "s3:GetObject",
      "Resource": [
        "arn:aws:s3:::ELASTIC-FLEET-EXAMPLE-BUCKET/vhd-folder/*",
        "arn:aws:s3:::ELASTIC-FLEET-EXAMPLE-BUCKET/scripts/*"
      ],
      "Condition": {
        "StringEquals": {
```

```

        "aws:SourceAccount": "your AWS ##### ID"
    }
}
},
{
    "Sid": "AllowRetrievalPermissionsToS3AppIconsForAppStream",
    "Effect": "Allow",
    "Principal": {
        "Service": "appstream.amazonaws.com"
    },
    "Action": "s3:GetObject",
    "Resource": "arn:aws:s3:::ELASTIC-FLEET-EXAMPLE-BUCKET/app-icons/*"
}
]
}

```

また、aws:SourceArn 条件を使用して、特定リソースへのリソースアクセスを制限できます。

Note

リソースの完全な ARN が不明な場合や、複数のリソースを指定する場合には、グローバルコンテキスト条件キー aws:SourceArn で、ARN の不明な部分を示すためにワイルドカード (*) を使用します。

```

{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "ConfusedDeputyPreventionExamplePolicy",
            "Effect": "Allow",
            "Principal": {
                "Service": "appstream.amazonaws.com"
            },
            "Action": "s3:GetObject",
            "Resource": [
                "arn:aws:s3:::ELASTIC-FLEET-EXAMPLE-BUCKET/vhd-folder/*",
                "arn:aws:s3:::ELASTIC-FLEET-EXAMPLE-BUCKET/scripts/*"
            ],
            "Condition": {
                "ArnLike": {

```



```

        "aws:SourceArn": "arn:{aws-partition}:appstream:{your region name}:
{your AWS account ID}:app-block/*"
    }
},
{
    "Sid": "AllowRetrievalPermissionsToS3AppIconsForAppStream",
    "Effect": "Allow",
    "Principal": {
        "Service": "appstream.amazonaws.com"
    },
    "Action": "s3:GetObject",
    "Resource": "arn:aws:s3:::ELASTIC-FLEET-EXAMPLE-BUCKET/app-icons/*"
}
]
}

```

aws:SourceArn および aws:SourceAccount 条件を使用して、特定のリソースおよびアカウントへのリソースアクセスを制限できます。

Note

リソースの完全な ARN が不明な場合や、複数のリソースを指定する場合には、グローバル条件コンテキストキー aws:SourceArn で、ARN の不明な部分を示すためにワイルドカード (*) を使用します。

```

{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "ConfusedDeputyPreventionExamplePolicy",
            "Effect": "Allow",
            "Principal": {
                "Service": "appstream.amazonaws.com"
            },
            "Action": "s3:GetObject",
            "Resource": [
                "arn:aws:s3:::ELASTIC-FLEET-EXAMPLE-BUCKET/vhd-folder/*",
                "arn:aws:s3:::ELASTIC-FLEET-EXAMPLE-BUCKET/scripts/*"
            ],
            "Condition": {

```

```

        "ArnLike": {
            "aws:SourceArn": "arn:{aws partition}:appstream:{your region name}:
{your AWS account ID}:app-block/*"
        },
        "StringEquals": {
            "aws:SourceAccount": "your AWS account ID"
        }
    },
    {
        "Sid": "AllowRetrievalPermissionsToS3AppIconsForAppStream",
        "Effect": "Allow",
        "Principal": {
            "Service": "appstream.amazonaws.com"
        },
        "Action": "s3:GetObject",
        "Resource": "arn:aws:s3:::ELASTIC-FLEET-EXAMPLE-BUCKET/app-icons/*"
    }
]
}

```

Amazon AppStream 2.0 のセキュリティのベストプラクティス

クラウドセキュリティは Amazon Web Services (AWS) の最優先事項です。セキュリティとコンプライアンスは、AWS とお客様の間の責任共有です。詳細については、「[責任共有モデル](#)」を参照してください。AWS および AppStream 2.0 のお客様は、スタック、フリート、イメージ、ネットワークなどのさまざまなレイヤーにセキュリティ対策を実装することが重要です。

AppStream 2.0 は本質的に一時的なため、多くの場合、アプリケーションやデスクトップ配信の安全なソリューションとして選択されます。Windows デプロイで一般的なウイルス対策ソリューションが、ユーザーセッションの終了時に事前定義されて削除される環境のユースケースに適しているかどうかについて検討してください。ウイルス対策は仮想化されたインスタンスにオーバーヘッドを追加するため、不要なアクティビティを軽減することがベストプラクティスとなっています。例えば、起動時に (一時的な) システムボリュームをスキャンしても AppStream 2.0 の全体的なセキュリティは向上しません。

AppStream 2.0 のセキュリティに関する 2 つの重要な質問は、主に次の点に関連しています。

- セッション終了後もユーザーの状態を維持することが必須か？
- ユーザーはセッション内でどのくらいのアクセス権限を持つべきか？

トピック

- [永続データの保護](#)
- [エンドポイントのセキュリティとウイルス対策](#)
- [ネットワーク除外](#)
- [AppStream 2.0 セッションの保護](#)
- [ファイアウォールとルーティング](#)
- [データ損失の防止](#)
- [Egress トラフィックの制御](#)
- [AWS サービスの使用](#)

永続データの保護

AppStream 2.0 のデプロイでは、ユーザーの状態を何らかの形で保持しなければならない場合があります。個々のユーザーのデータを永続化する場合や、共有フォルダを使用してコラボレーション用にデータを保持する場合などです。AppStream 2.0 インスタンスストレージは一時的であり、暗号化オプションはありません。

AppStream 2.0 では、Amazon S3 のホームフォルダとアプリケーション設定を通じてユーザーの状態を永続化します。一部のユースケースでは、ユーザーの状態の永続化をより細かく制御する必要があります。このようなユースケースについては、AWS は、サーバーメッセージブロック (SMB) ファイル共有の使用を推奨しています。

ユーザーの状態とデータ

ほとんどの Windows アプリケーションは、ユーザーが作成したアプリケーションデータと同じ場所に配置すると最適かつ最も安全に動作するため、このデータを AppStream 2.0 フリートと同じ AWS リージョンに保持することがベストプラクティスです。このデータを暗号化することがベストプラクティスです。ユーザーホームフォルダのデフォルトの動作は、AWS キー管理サービス () の Amazon S3-managed暗号化キーを使用して保管中のファイルとフォルダを暗号化することですAWS KMS。AWS コンソールまたは Amazon S3 バケットにアクセスできる AWS 管理ユーザーは、これらのファイルに直接アクセスできることに注意してください。

ユーザーファイルやフォルダを保存するために Windows ファイル共有のサーバーメッセージブロック (SMB) ターゲットを必要とする設計では、この処理は自動で行われるか、または設定が必要です。

表 5 — ユーザーデータを保護するためのオプション

SMB ターゲット	保管時の暗号化	転送時の暗号化	ウイルス対策 (AV)
FSx for Windows File Server	KMS AWS による自動	SMB 暗号化によって自動	リモートインスタンスにインストールされた AV は、マップされたドライブでスキャンを実行します。
ファイルゲートウェイ、AWS Storage Gateway	デフォルトでは、S3 AWS Storage Gateway に よって保存されるすべてのデータは、Amazon S3-Managed を使用してサーバー側で暗号化されます。(AWS Key Management Service KMS) を使用して保存データを暗号化するように、オプションでさまざまなゲートウェイタイプを設定できます。	任意のタイプのゲートウェイアプライアンスと AWS ストレージ間で転送されるすべてのデータは、SSL を使用して暗号化されます。	リモートインスタンスにインストールされた AV は、マップされたドライブでスキャンを実行します。
EC2 ベースの Windows File Server	EBS 暗号化を有効にする	PowerShell、Set-SmbServer Configuration - EncryptData \$True	サーバーにインストールされた AV は、ローカルドライブでスキャンを実行します。

エンドポイントのセキュリティとウイルス対策

Amazon AppStream 2.0 インスタンスは本質的に一時的であり、データに永続性がないため、永続デスクトップで必要となるアクティビティによってユーザーのエクスペリエンスとパフォーマンスが損

なわれないようにするには、別のアプローチが必要です。Endpoint Security エージェントは、組織のポリシーがある場合、または E メール、ファイル入力、外部ウェブブラウジングなどの外部データ入力で使用される場合、AppStream 2.0 イメージにインストールされます。

一意識別子の削除

Endpoint Security エージェントにはグローバル一意識別子 (GUID) がある場合があります、フリートインスタンスの作成プロセス中にリセットする必要があります。ベンダーは、イメージから生成されたインスタンスごとに新しい GUID が確実に生成されるように、製品をイメージにインストールする手順を定めています。

GUID が生成されないようにするには、AppStream 2.0 Assistant を実行してイメージを生成する前の最後のアクションとして Endpoint Security エージェントをインストールします。

パフォーマンスの最適化

Endpoint Security ベンダーは、AppStream 2.0 のパフォーマンスを最適化するスイッチと設定を提供しています。設定はベンダーによって異なり、ベンダーのドキュメント (通常は VDI に関するセクション) に記載されています。一般的な設定には以下が含まれますが、これらに限定されません。

- 起動時のスキャンをオフにして、インスタンスの作成、起動、ログインにかかる時間を最小限に抑える
- 不要なスキャンを防ぐため、定期スキャンをオフにする
- ファイルが列挙されないように署名キャッシュをオフにする
- VDI に最適化された IO 設定を有効にする
- パフォーマンスを確保するためにアプリケーションが必要とする除外

Endpoint Security ベンダーは、パフォーマンスを最適化する仮想デスクトップ環境での使用方法を提供しています。

- Trend Micro Office Scan [仮想デスクトップインフラストラクチャのサポート - Apex One/OfficeScan \(trendmicro.com\)](#)
- CrowdStrike と [CrowdStrike Falcon をデータセンターに設置する方法](#)
- Sophos と [Sophos Central エンドポイント: ID の重複を避けるためにゴールドイメージにインストールする方法、および Sophos Central: 仮想デスクトップ環境に Windows エンドポイントをインストールする際のベストプラクティス](#)

- McAfee と [McAfee Agent の仮想デスクトップインフラストラクチャシステムへのプロビジョニングとデプロイ](#)
- Microsoft Endpoint Security と [非永続 VDI マシン用の Microsoft Defender Antivirus の設定](#) - Microsoft Tech Community

スキップの除外

セキュリティソフトウェアが AppStream 2.0 インスタンスにインストールされている場合、セキュリティソフトウェアがプロセスに干渉しないようにする必要があります。

表 6 — AppStream 2.0 プロセスセキュリティソフトウェアが干渉してはいけないプロセス。

サービス	プロセス
AmazonCloudWatchAgent	"C:\Program Files\Amazon\AmazonCloudWatchAgent\start-amazon-cloudwatch-agent.exe"
AmazonSSMAgent	"C:\Program Files\Amazon\SSM\amazon-ssm-agent.exe"
NICE DCV	"C:\Program Files\NICE\DCV\Server\bin\dcvserver.exe" "C:\Program Files\NICE\DCV\Server\bin\dcvagent.exe"
AppStream 2.0	「C:\Program Files\Amazon\AppStream2\StorageConnector\StorageConnector.exe」 フォルダ "C:\Program Files\Amazon\Photon\ ".\Agent\PhotonAgent.exe" ".\Agent\s5cmd.exe" ".\WebServer\PhotonAgentWebServer.exe" ".\CustomShell\PhotonWindowsAppSwitcher.exe" ".\CustomShell\PhotonWindowsCustomShell.exe"

サービス	プロセス
	".\CustomShell\PhotonWindowsCustomShellBackground.exe"

フォルダ

セキュリティソフトウェアが AppStream 2.0 インスタンスにインストールされている場合、そのソフトウェアは次のフォルダに干渉しないようにします。

Example

```
C:\Program Files\Amazon\*
C:\ProgramData\Amazon\*
C:\Program Files (x86)\AWS Tools\*
C:\Program Files (x86)\AWS SDK for .NET\*
C:\Program Files\NICE\*
C:\ProgramData\NICE\*
C:\AppStream\*
```

Endpoint Security コンソールの健全性

Amazon AppStream 2.0 は、ユーザーがアイドルタイムアウトと切断タイムアウトを超えて接続するたびに、新しい一意のインスタンスを作成します。インスタンスには一意の名前が付けられ、エンドポイントセキュリティ管理コンソールに蓄積されます。4 日以上経過した (または AppStream 2.0 のセッションタイムアウトによってはそれ以下) 未使用の古くなったマシンを削除するように設定すると、コンソールに表示される期限切れのインスタンス数を最小限に抑えることができます。

ネットワーク除外

AppStream 2.0 管理ネットワーク範囲 (198.19.0.0/16) とそれに続くポートとアドレスは、AppStream 2.0 インスタンス内のセキュリティ、ファイアウォール、ウイルス対策ソリューションでブロックしないでください。

表 7 — AppStream 2.0 ストリーミングインスタンスのセキュリティソフトウェアが干渉してはならないポート

[ポート]	使用方法
8300	これはストリーミング接続の確立に使用されます。
3128	これは、AppStream 2.0 によるストリーミングインスタンスの管理に使用されます。
8000	これは、AppStream 2.0 によるストリーミングインスタンスの管理に使用されます。
8443	これは、AppStream 2.0 によるストリーミングインスタンスの管理に使用されます。
53	DNS

表 8 — AppStream 2.0 のマネージドサービスアドレスセキュリティソフトウェアが干渉してはならないアドレス

[ポート]	使用方法
169.254.169.123	NTP
169.254.169.249	NVIDIA GRID ライセンスサービス
169.254.169.250	KMS
169.254.169.251	KMS
169.254.169.253	DNS
169.254.169.254	メタデータ

AppStream 2.0 セッションの保護

アプリケーションとオペレーティングシステムの制御の制限

AppStream 2.0 では、管理者はアプリケーションストリーミングモードでウェブページから起動できるアプリケーションを正確に指定できます。ただし、これによって、指定されたアプリケーションのみを実行できることが保証されるわけではありません。

Windows ユーティリティとアプリケーションは、追加の手段でオペレーティングシステムを介して起動できます。組織に必要なアプリケーションのみを実行できるように、[Microsoft AppLocker](#) を使用する AWS ことをお勧めします。デフォルトのルールでは、重要なシステムディレクトリへのパスアクセスがすべてのユーザーに許可されるため、変更する必要があります。

Note

Windows Server 2016 と 2019 では、AppLocker ルールを適用するために Windows Application Identity サービスを実行する必要があります。Microsoft AppLocker を使用する AppStream 2.0 からのアプリケーションアクセスについては、[AppStream 管理者ガイド](#)に詳しく記載されています。

Active Directory ドメインに参加しているフリートインスタンスの場合は、グループポリシーオブジェクト (GPO) を使用してユーザーとシステム設定を配信し、ユーザーのアプリケーションとリソースへのアクセスを保護します。

ファイアウォールとルーティング

AppStream 2.0 フリートを作成するときは、サブネットとセキュリティグループを割り当てる必要があります。サブネットには、ネットワークアクセスコントロールリスト (NACL) とルートテーブルが既に割り当てられています。新しい Image Builder の起動中、または新しいフリートの作成中に、[最大 5 つのセキュリティグループ](#)を関連付けることができます。セキュリティグループには、[既存のセキュリティグループから最大 5 つの割り当て](#)を関連付けることができます。セキュリティグループごとに、インスタンスとの間で送受信されるネットワークトラフィックのアウトバウンドとインバウンドを制御するルールを追加します。

NACL は 1 つまたは複数の サブネットのインバウンドトラフィックとアウトバウンドトラフィックを制御するためのファイアウォールとして機能する任意指定の VPC セキュリティレイヤーです。セキュリティの追加レイヤーを VPC に追加するには、セキュリティグループと同様のルールを指定し

たネットワーク ACL をセットアップできます。セキュリティグループとネットワーク ACL の違いの詳細については、「[セキュリティグループとネットワーク ACL を比較する](#)」のページを参照してください。

セキュリティグループと NACL のルールを設計して適用するときは、権限を最小限に抑えるための AWS Well-Architected のベストプラクティスを検討してください。最小特権とは、タスクを完了するために必要なアクセス許可のみを付与する原則です。

オンプレミス環境を (AWS Direct Connect 経由で) AWS に接続する高速プライベートネットワークを保有するカスタマーは、AppStream 用の VPC エンドポイントの使用を検討してください。この場合、ストリーミングトラフィックは、パブリックインターネットを経由するのではなく、プライベートネットワーク接続を介してルーティングされます。このトピックの詳細については、このドキュメントの「AppStream 2.0 ストリーミングインターフェイス VPC エンドポイント」のセクションを参照してください。

データ損失の防止

ここでは、2 種類のデータ損失防止について確認します。

クライアントから AppStream 2.0 インスタンスへのデータ転送の制御

表 9 — データの入出力の制御に関するガイダンス

設定	オプション	ガイダンス
クリップボード	- リモートセッションにのみコピーと貼り付け - ローカルデバイスにのみコピー - 無効	この設定を無効にしても、セッション内のコピーと貼り付けは無効になりません。セッションにデータをコピーする必要がある場合は、データ漏洩の可能性を最小限に抑えるため、[リモートセッションにのみ貼り付ける] を選択してください。
ファイル転送	- アップロードとダウンロード - アップロードのみ	データ漏えいを防ぐため、この設定は有効にしないでください。

設定	オプション	ガイダンス
	- ダウンロードのみ - 無効	
ローカルデバイスへの印刷	- 有効 - 無効	印刷が必要な場合は、組織によって制御およびモニタリングされているネットワークマッピングされたプリンタを使用してください。

既存の組織データ転送ソリューションがスタック設定よりも優れている点について検討してください。これらの設定は、包括的なセキュアデータ転送ソリューションに代わるものではありません。

Egress トラフィックの制御

データ損失が懸念される場合は、ユーザーが AppStream 2.0 インスタンス内に入ったときにアクセスできるものを非表示にすることが重要です。ネットワークの出口 (または出力) パスはどのようになっていますか？ エンドユーザーが AppStream 2.0 インスタンス内でパブリックインターネットにアクセスできるようにすることは一般的な要件であるため、ネットワークパスに WebProxy またはコンテンツフィルタリングソリューションを配置することを検討する必要があります。その他の考慮事項には、ローカルのウイルス対策アプリケーションと AppStream インスタンス内のその他のエンドポイントセキュリティ対策が含まれます (詳細については、「エンドポイントセキュリティとウイルス対策」のセクションを参照してください)。

AWS サービスの使用

AWS Identity and Access Management

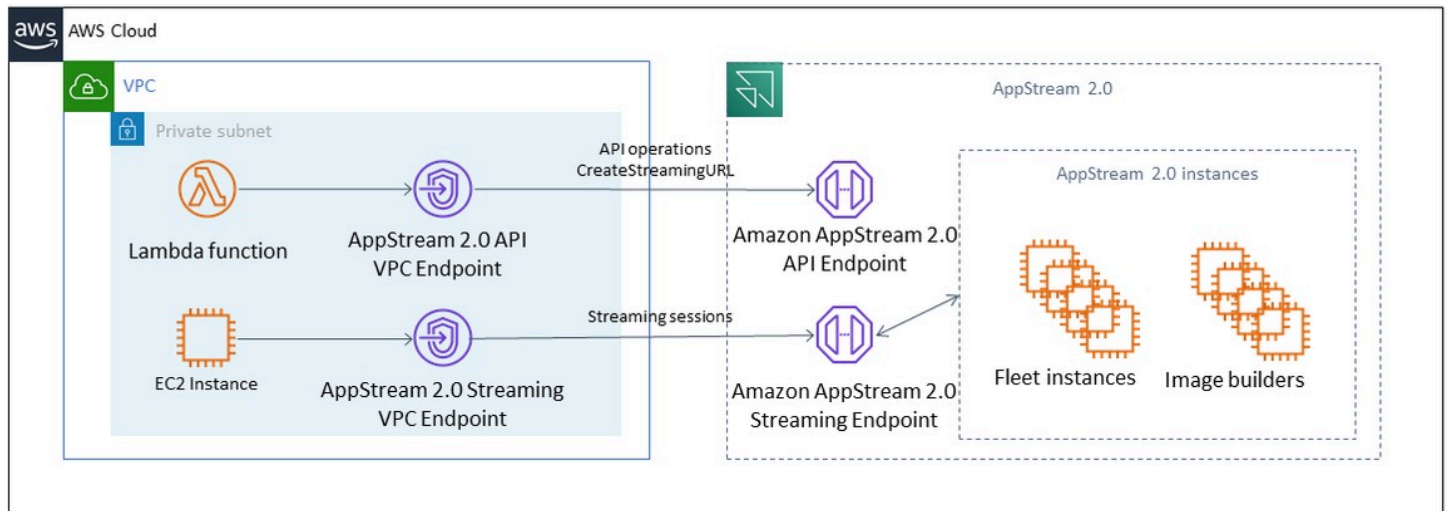
IAM ロールを使用して AWS サービスにアクセスし、それにアタッチされた IAM ポリシーに固有であることは、AppStream 2.0 セッションのユーザーのみが追加の認証情報を管理せずにアクセスできるようにするベストプラクティスです。[AppStream 2.0 で IAM ロールを使用する際のベストプラクティス](#)に従ってください。

ユーザーデータをホームフォルダとアプリケーション設定の永続化の両方に保持するために作成された [Amazon S3 バケットを保護するための IAM ポリシー](#)を作成します。これにより、[AppStream 2.0 管理者以外はアクセスできなくなります](#)。

VPC エンドポイント

VPC エンドポイントは、VPC とサポートされている AWS サービス、および を搭載した VPC エンドポイントサービス間のプライベート接続を有効にします AWS PrivateLink。AWS PrivateLink は、プライベート IP アドレスを使用して サービスにプライベートにアクセスできるテクノロジーです。VPC と他のサービス間のトラフィックは、Amazon ネットワークを離れません。AWS サービスにのみパブリックインターネットアクセスが必要な場合、VPC エンドポイントは NAT ゲートウェイとインターネットゲートウェイの要件を完全に削除します。

自動化ルーチンや開発者が AppStream 2.0 の API 呼び出しを行う必要がある環境では、[AppStream 2.0 API オペレーション用のインターフェイス VPC エンドポイントを作成します](#)。例えば、パブリックインターネットへのアクセスがないプライベートサブネットに EC2 インスタンスがある場合、AppStream 2.0 API の VPC エンドポイントを使用して [CreateStreamingURL](#) などの AppStream 2.0 API オペレーションを呼び出すことができます。次の図は、AppStream 2.0 API とストリーミング VPC エンドポイントが Lambda 関数と EC2 インスタンスによって使用される設定例を示しています。



VPC エンドポイント

ストリーミング VPC エンドポイントでは、VPC エンドポイントを介してセッションをストリーミングできます。ストリーミングインターフェイスエンドポイントは、VPC 内のストリーミングトラフィックを維持します。ストリーミングトラフィックには、ピクセル、USB、ユーザー入力、オーディオ、クリップボード、ファイルのアップロードとダウンロード、プリンターのトラフィックが含まれます。VPC エンドポイントを使用するには、AppStream 2.0 スタックで VPC エンドポイント設定が有効になっている必要があります。これは、インターネットアクセスが制限されていて、Direct Connect インスタンス経由でアクセスしたほうが有利な場所から、パブリックインターネット経由で

ユーザーセッションをストリーミングする代替手段となります。VPC エンドポイントを介してユーザーセッションをストリーミングするには、以下が必要です。

- インターフェイスエンドポイントに関連付けられているセキュリティグループは、ユーザーが接続する IP アドレス範囲からポート 443 (TCP) とポート 1400-1499 (TCP) へのインバウンドアクセスを許可する必要があります。
- サブネットのネットワークアクセスコントロールリストでは、一時ネットワークポート 1024-65535 (TCP) から、ユーザーが接続する IP アドレス範囲へのアウトバウンドトラフィックを許可する必要があります。
- ユーザーを認証し、AppStream 2.0 が機能するために必要なウェブアセットを配信するためには、インターネットに接続できることが必須です。

AppStream 2.0 による AWS サービスへのトラフィックの制限の詳細については、VPC [エンドポイントを作成してストリーミング](#)するための管理ガイドを参照してください。

パブリックインターネットへの完全なアクセスが必要な場合は、Image Builder で Internet Explorer のセキュリティ強化構成 (ESC) を無効にするのがベストプラクティスです。詳細については、「AppStream 2.0 管理ガイド」の「[Internet Explorer セキュリティ強化構成を無効にする](#)」を参照してください。

トラブルシューティング

Amazon AppStream 2.0 の操作中に問題が発生した場合は、次のトラブルシューティングのリソースを参照してください。

内容

- [一般的なトラブルシューティング](#)
- [Image Builder のトラブルシューティング](#)
- [フリートのトラブルシューティング](#)
- [Active Directory のトラブルシューティング](#)
- [AppStream 2.0 ユーザーの問題のトラブルシューティング](#)
- [永続ストレージ問題のトラブルシューティング](#)
- [通知コードのトラブルシューティング](#)

一般的なトラブルシューティング

以下は、Amazon AppStream 2.0 を使用する場合に発生する可能性がある一般的な問題です。

問題

- [SAML フェデレーションが機能していません。ユーザーに AppStream 2.0 アプリケーションを表示する権限がありません。](#)
- [ADFS ポータルからのフェデレーション後に、ストリーミングセッションが開始されません。「接続が切断されました」というエラーが表示されます。](#)
- [無効なリダイレクト URI エラーが発生します。](#)
- [自分のイメージビルダーとフリートが \[実行中\] 状態になることはありません。自分の DNS サーバーは Simple AD ディレクトリにあります。](#)
- [ユーザーのためにアプリケーション設定の永続化を有効にしましたが、永続的なアプリケーション設定が保存またはロードされません。](#)
- [ユーザーに対してアプリケーション設定の永続化を有効にしましたが、特定のストリーミングアプリケーションでは、ユーザーのパスワードがセッション間で永続化されません。](#)
- [Google Chrome データが、ユーザーの永続的なアプリケーション設定を保持する VHD ファイルに保存されます。このため、ユーザーの設定が保持されません。Chrome プロファイルを管理する方法を教えてください。](#)

- 埋め込み AppStream 2.0 ストリーミングセッション用にカスタムドメインを設定しましたが、AppStream 2.0 ストリーミング URL はカスタムドメインにリダイレクトされません。
- スマートカード対応の AppStream 2.0 フリートでアプリを起動しましたが、認証用に使用できる証明書数が限られています (またはありません)。
- 自分のスマートカード対応の AppStream 2.0 フリートで、Certificate Propagation サービスが開始されません。
- SAML 認証後に Active Directory のユーザー名またはパスワードでログインできない。

SAML フェデレーションが機能していません。ユーザーに AppStream 2.0 アプリケーションを表示する権限がありません。

このエラーは、SAML 2.0 フェデレーションの IAM ロール用に埋め込まれているインラインポリシーに、スタック ARN へのアクセス許可が含まれていないことが原因で発生する可能性があります。この IAM ロールは、AppStream 2.0 スタックにアクセスしているフェデレーティッドユーザーによって引き受けられます。ロールのアクセス許可を編集して、スタック ARN を含めます。詳細については、IAM ユーザーガイドの「[Amazon AppStream 2.0 と SAML 2.0 の統合](#)」および「[AWSでの SAML 2.0 フェデレーションのトラブルシューティング](#)」を参照してください。

ADFS ポータルからのフェデレーション後に、ストリーミングセッションが開始されません。「接続が切断されました」というエラーが表示されません。

NameID SAML 属性で、クレームルールの [Incoming Claim Type] を [UPN] に設定し、接続をやり直してください。

無効なリダイレクト URI エラーが発生します。

このエラーは、AppStream 2.0 スタックのリリースステート URL の形式が不正または無効な場合に発生します。フェデレーションのセットアップ時に設定されたリリースステートが、スタックの詳細に表示されているスタックのリリースステートと同じように設定されていることを AppStream 2.0 コンソールから確認してください。同じで、問題が解決しない場合は、お問い合わせください AWS サポート。詳細については、「[Amazon AppStream 2.0 と SAML 2.0 の統合](#)」を参照してください。

自分のイメージビルダーとフリートが [実行中] 状態になることはありません。自分の DNS サーバーは Simple AD デイレクトリにあります。

AppStream 2.0 は VPC 内の DNS サーバーに依存して、存在しないローカルドメイン名に対して存在しないドメイン (NXDOMAIN) レスポンスを返します。これにより、AppStream 2.0 マネージド型のネットワークインターフェイスは管理サーバーとやり取りできます。

Simple AD を使用してデイレクトリを作成すると、はユーザーに代わって DNS サーバーとしても機能する 2 つのドメインコントローラー AWS Directory Service を作成します。これらのドメインコントローラーは NXDOMAIN レスポンスを返さないため、AppStream 2.0 で使用することはできません。

ユーザーのためにアプリケーション設定の永続化を有効にしましたが、永続的なアプリケーション設定が保存またはロードされません。

AppStream 2.0 は、作成されたアプリケーション設定を Windows インスタンスの特定の場所に自動的に保存します。これらの設定が保存されるのは、アプリケーションによって以下のいずれかの場所に保存された場合に限ります。サポートされている場所のリストについては、「[アプリケーション設定の永続化の仕組み](#)」を参照してください。アプリケーションの保存先を C:\Users\%username% に設定していて、ユーザーのアプリケーション設定がセッション間で保持されないときは、マウントポイントが作成されていない場合があります。この場合は、ユーザーの永続的なアプリケーション設定の保存先である VHD ファイルに設定が保存されません。

この問題を解決するには、以下の手順を実行します。

1. フリートインスタンスで File Explorer を開き、ユーザープロファイルデイレクトリ (C:\Users\%username%) を参照します。
2. このデイレクトリにシンボリックリンクが含まれているかどうかを確認し、以下のいずれかの操作を行います。
 - シンボリックリンクがある場合は、それが D:\%username% を指していることを確認します。
 - シンボリックリンクがない場合は、C:\Users\%username% デイレクトリの削除を試行します。

このデイレクトリを削除できない場合は、このデイレクトリ内で削除を妨げているファイルを特定し、このファイルを作成したアプリケーションを特定します。次に、ファイルのアクセス許可を変更する方法またはファイルを移動する方法について、アプリケーションベンダーに問い合わせます。

このディレクトリを削除できる場合は、この問題を解決するための詳細なガイダンス AWS サポート について お問い合わせ してください。詳細については、[AWS サポート センター](#)を参照してください。

ユーザーに対してアプリケーション設定の永続化を有効にしましたが、特定のストリーミングアプリケーションでは、ユーザーのパスワードがセッション間で永続化されません。

次の場合にこの問題が発生します。

- ユーザーは、[Microsoft Data Protection API](#) を使用する Microsoft Outlook などのストリーミングアプリケーションです。
- アプリケーション設定の永続化は、Active Directory ドメインに参加していないストリーミングインスタンスに対して有効になります。

ストリーミングインスタンスが Active Directory ドメインに参加していない場合、Windows ユーザーの PhotonUser はフリートインスタンスごとに異なります。DPAPI セキュリティモデルの機能上の理由から、このシナリオで DPAPI を使用するアプリケーションではユーザーのパスワードは保持されません。ストリーミングインスタンスが Active Directory ドメインに参加していて、そのユーザーがドメインユーザーである場合、Windows ユーザー名はログインしているユーザーの名前であり、DPAPI を使用するアプリケーションではユーザーのパスワードは保持されます。

Google Chrome データが、ユーザーの永続的なアプリケーション設定を保持する VHD ファイルに保存されます。このため、ユーザーの設定が保持されません。Chrome プロファイルを管理する方法を教えてください。

デフォルトでは、Google Chrome はユーザーデータとローカルディスクキャッシュの両方を Windows ユーザープロファイルに保存します。ユーザーの永続的なアプリケーション設定が保持されている VHD ファイルにローカルディスクキャッシュデータが保存されないようにするには、ユーザーデータのみを保存するように Chrome を設定します。これを行うには、フリートインスタンスで管理者としてコマンドラインを開き、以下のパラメータを指定して Chrome を開始し、ディスクキャッシュの場所を変更します。

```
chrome.exe --disk-cache-dir C:\path-to-unsaved-location\
```

これらのパラメータを使用して Chrome を実行すると、ディスクキャッシュは AppStream 2.0 セッション間で保持されなくなります。

埋め込み AppStream 2.0 ストリーミングセッション用にカスタムドメインを設定しましたが、AppStream 2.0 ストリーミング URL はカスタムドメインにリダイレクトされません。

この問題を解決するには、AppStream 2.0 ストリーミング URL を作成したときに、AppStream 2.0 エンドポイントをカスタムドメインに置き換えたことを確認します。デフォルトでは、AppStream 2.0 ストリーミング URL は以下の形式になっています。

```
https://appstream2.region.aws.amazon.com/authenticate?parameters=authenticationcode
```

ストリーミング URL のデフォルトの AppStream 2.0 エンドポイントを置き換えるには、URL の `https://appstream2.region` をカスタムドメインに置き換えます。例えば、カスタムドメインが `training.example.com` の場合、新しいストリーミング URL は次の形式に従う必要があります。

```
https://training.example.com/authenticate?parameters=authenticationcode
```

埋め込み AppStream 2.0 ストリーミングセッションのカスタムドメインの設定の詳細については、「[カスタムドメインを使用するための設定要件](#)」をご参照ください。

スマートカード対応の AppStream 2.0 フリートでアプリを起動しましたが、認証用に使用できる証明書数が限られています (またはありません)。

この状況は、[Certificate Propagation](#) サービスが実行状態になる前にアプリケーションが起動された場合に発生します。

この問題を解決するには、PowerShell モジュール [Get-Service](#) を使用して Certificate Propagation サービスのステータスにクエリを実行し、アプリケーションを起動する前に実行状態になっていることを確認します。

例えば、次のスクリプトでは、Certificate Propagation サービスが実行されるまでアプリケーションは起動されません。

```
$logfile = "$Env:TEMP\AS2\Logging\$((Get-Date -Format "yyyy-MM-dd-HH-mm-ss"))_applaunch.log"
New-Item -path $logfile -ItemType File -Force | Out-Null

Function Write-Log {
    Param ([string]$message)
    $stamp = Get-Date -Format "yyyy/MM/dd HH:mm:ss"
    $logoutput = "$stamp $message"
    Add-content $logfile -value $logoutput
}

if (Get-Service -Name "CertPropSvc" | Where-Object -Property Status -eq Running) {

    Write-Log "The Certificate Propagation Service is running. Launching
Application..."
    try {
        Start-Process -FilePath "Path to Application" -WindowStyle Maximized -
ErrorAction Stop
    }
    catch {
        Write-Log "There was an error launching the application: $_"
    }
}

else {

    do {

        $status = Get-Service "CertPropSvc" | select-object -ExpandProperty Status
        Write-Log "The Certificate Propagation service status is currently $status"
        Start-Sleep -Seconds 2

    } until (Get-Service -Name "CertPropSvc" | Where-Object -Property Status -eq
Running)

    write-log "The Certificate Propagation Service is running. Launching
Application..."
    try {
        Start-Process -FilePath "Path to Application" -WindowStyle Maximized -
ErrorAction Stop
    }
    catch {
```

```
Write-Log "There was an error launching the application: $_"

    }
}
```

自分のスマートカード対応の AppStream 2.0 フリートで、Certificate Propagation サービスが開始されません。

[Certificate Propagation](#) サービスが開始されない場合、サービスのスタートアップタイプを [Disabled] (無効) に設定する必要があります。この問題を解決するには、フリートのイメージの作成に使用される AppStream 2.0 Image Builder で Windows サービス Microsoft 管理コンソールを起動し、Certificate Propagation サービスのスタートアップタイプが [Disabled] (無効) に設定されていることを確認します。

スタートアップタイプが [Disabled] (無効) に設定されておらず、AppStream 2.0 フリートでサービスがまだ開始されない場合は、PowerShell モジュール [Start-Service](#) を使用すると、フリートインスタンスの起動時に Certificate Propagation サービスが開始されます。

例えば、次の PowerShell スクリプトでは、サービスが停止状態であることを検出すると、サービスが開始されます。

```
$logfile = "C:\AppStream\Logging\$(Get-Date -Format "yyyy-MM-dd-HH-mm-ss")_certpropcheck.log"
New-Item -path $logfile -ItemType File -Force | Out-Null

Function Write-Log {
    Param ([string]$message)
    $stamp = Get-Date -Format "yyyy/MM/dd HH:mm:ss"
    $logoutput = "$stamp $message"
    Add-content $logfile -value $logoutput
}

if (Get-Service -Name "CertPropSvc" | Where-Object -Property Status -eq Running) {

    Write-Log "The Certificate Propagation Service is running. Exiting..."
    Exit
}
else {
    do {
```

```
if (Get-Service -Name "CertPropSvc" | Where-Object -Property Status -eq
Stopped) {

    Write-Log "The Certificate Propagation Service is stopped, attempting to
start..."
    try {
        Start-Service -Name "CertPropSvc" -ErrorAction Stop

    }
    catch {
        Write-Log "There was a problem starting the service: $_"
        break
    }

    $status = Get-Service "CertPropSvc" | select-object -ExpandProperty Status
    Write-Log "The Certificate Propagation service status is currently $status"

}
else {

    $status = Get-Service "CertPropSvc" | select-object -ExpandProperty Status
    Write-Log "The Certificate Propagation service status is currently $status"
    break

}

} until (Get-Service -Name "CertPropSvc" | Where-Object -Property Status -eq
Running)
}
```

SAML 認証後に Active Directory のユーザー名またはパスワードでログインできない。

SAML クレームの nameID は、Active Directory のユーザー名と一致する必要があります。一部の IdPs、特定の属性を調整した後、更新、更新、または再デプロイが必要です。調整を行い、SAML キャプチャに反映されない場合は、変更を有効にするために必要な特定のステップについて、IdP のドキュメントまたはサポートプログラムを参照してください。

Image Builder のトラブルシューティング

以下は、Amazon AppStream 2.0 Image Builder を使用する場合に発生する可能性がある問題です。

問題

- [Image Builder からインターネットに接続できない。](#)
- [アプリケーションをインストールするときに、「このオペレーティングシステムのバージョンはサポートされていません」というエラーが表示されます。](#)
- [自分のアプリケーションを開くために Windows PowerShell スクリプトを使いたいのですが。](#)
- [ユーザーが ClickOnce アプリケーションを利用できるようにしたいのですが。](#)
- [Image Builder に接続するときに、Ctrl+Alt+Delete を入力してログインするように求めるログイン画面が表示されます。しかし、キーストロークがローカルマシンにインターセプトされます。](#)
- [管理者モードおよびテストモードに切り替えるときに、パスワードが要求されます。パスワードを取得する方法がわかりません。](#)
- [インストールされているアプリケーションを追加するとエラーが発生します。](#)
- [Image Builder のバックグラウンドサービスを誤って終了して、切断されました。Image Builder に接続できなくなりました。](#)
- [アプリケーションのテストモードでの起動ができなくなります。](#)
- [VPC で、アプリケーションがネットワークリソースに接続できませんでした。](#)
- [Image Builder デスクトップをカスタマイズしましたが、作成したイメージからフリートを起動してセッションに接続した後、変更内容が使用できません。](#)
- [アプリケーションを起動するときにコマンドラインパラメータがなくなります。](#)
- [ウイルス対策アプリケーションをインストールした後、フリートでイメージを使用できません。](#)
- [イメージ作成に失敗しました。](#)
- [PrewarmManifest.txt へのアクセスが拒否されたというエラーメッセージが表示され、Image Assistant create-image オペレーションが失敗しました](#)

Image Builder からインターネットに接続できない。

Image Builder は、デフォルトではインターネットと通信することはできません。この問題を解決するには、インターネットにアクセスできる VPC サブネットに Image Builder を起動します。VPC サブネットからのインターネットアクセスを有効にするには、[NAT ゲートウェイ](#)を使用します。または、VPC にインターネットゲートウェイを設定し、Image Builder に Elastic IP アドレスをアタッチできます。詳細については、「[Amazon AppStream 2.0 のネットワークングとアクセス](#)」を参照してください。

アプリケーションをインストールするときに、「このオペレーティングシステムのバージョンはサポートされていません」というエラーが表示されます。

AppStream 2.0 イメージに追加できるのは、Windows Server 2012 R2、Windows Server 2016、および Windows Server 2019 にインストールできるアプリケーションだけです。使用する Image Builder に応じて、これら 3 つのオペレーティングシステムのいずれかでアプリケーションがサポートされているかどうかを確認してください。

自分のアプリケーションを開くために Windows PowerShell スクリプトを使いたいのですが。

Windows PowerShell スクリプトを使用して、フリートインスタンスでアプリケーションを開くことができます。アプリケーションを開く前に、これを実行してアプリケーションまたは環境を設定できます。アプリケーションの Windows PowerShell スクリプトを起動するには、Image Assistant の PowerShell で .exe ファイルを指定します。C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe に移動し、次の起動パラメータを指定します。

```
-file "C:\Path\To\PowerShell\Script.ps1"
```

Note

指定されたスクリプトでアプリケーションを起動できるようにするには、PowerShell スクリプトの実行ポリシーを上書きする必要があります。そのためには、**-ExecutionPolicy Bypass** を起動パラメータに追加します。

ユーザーが ClickOnce アプリケーションを利用できるようにしたいのですが。

ClickOnce アプリケーションを AppStream 2.0 ユーザーが利用できるようにするには、最初に管理者として、次にテンプレートユーザーとして、アプリケーションを Image Builder にインストールする必要があります。ClickOnce アプリケーションにはユーザー固有のインストールが必要なため、ユーザーがフリートインスタンスからアプリケーションを起動できるようにするには、テンプレートユーザーとしてアプリケーションをインストールする必要があります。ClickOnce アプリケーションをインストールするには、管理者として、次にテンプレートユーザーとして、次のステップを実行します。

1. AppStream 2.0 コンソールを <https://console.aws.amazon.com/appstream2> で開きます。
2. 左側のナビゲーションペインで、[Images]、[Image Builder] の順に選択します。
3. リストで、使用する Image Builder を選択し、管理者としてログインします。
4. appref-ms ファイルを呼び出すバッチファイルをユーザープロファイル内に作成します。%APPDATA% 環境変数を使用して、C:\Users\username\AppData\Roaming を置き換えます。バッチファイル呼び出しの例を次に示します。

```
explorer "%APPDATA%\Microsoft\Windows\Start Menu\Programs\Company\ClickOnce.appref-ms"
```

5. Image Builder デスクトップから、Image Assistant を開きます。
6. [Configure Apps (アプリケーションの構成)] ページで、[Switch user (ユーザーの切り替え)] を選択します。
7. [Local User] タブで、[Template User] を選択します。
8. テンプレートユーザーとしてログインした後、アプリケーションをもう一度インストールします。
9. Image Builder デスクトップから、Image Assistant を開きます。
10. [アプリの構成] ページで、ClickOnce アプリケーションを開いて正しく機能していることを確認します。テストが完了したら、[Switch user (ユーザーの切り替え)] を選択します。
11. 管理者として再度ログインし、Image Assistant で必要なステップを実行してイメージの作成を完了します。

Image Builder に接続するときに、Ctrl+Alt+Delete を入力してログインするように求めるログイン画面が表示されます。しかし、キーストロークがローカルマシンにインターセプトされます。

クライアントが、特定のキーの組み合わせを Image Builder セッションに送信せずに、ローカルでインターセプトする場合があります。Ctrl+Alt+Delete キーの組み合わせを Image Builder に確実に送信するために、[Admin Commands]、[Send Ctrl+Alt+Delete] の順に選択します。[Admin Commands] メニューは、Image Builder セッションツールバーの右上隅で利用できます。

管理者モードおよびテストモードに切り替えるときに、パスワードが要求されます。パスワードを取得する方法がわかりません。

AppStream 2.0 では、通常、選択したユーザーモードに自動的にログインされます。状況によっては、この切り替えが自動的に行われないことがあります。パスワードがリクエストされた場合は、[Admin Commands (管理コマンド)]、[Log me in (ログイン)] の順に選択します。ワンタイムパスワードを Image Builder に安全に送信し、[Password] フィールドに貼り付けます。

インストールされているアプリケーションを追加するとエラーが発生します。

アプリケーションのタイプがサポートされているかどうかを確認してください。アプリケーションのタイプ .exe、.lnk、および .bat を追加できます。

C:\Users フォルダの階層にアプリケーションがインストールされているかを確認してください。C:\Users にインストールされているアプリケーションはサポートされていません。アプリケーションをインストールするときは、C:\ の別のインストールフォルダを選択してください。

Image Builder のバックグラウンドサービスを誤って終了して、切断されました。Image Builder に接続できなくなりました。

Image Builder を停止して、再起動し、再度接続してください。問題が解決しない場合は、新しい Image Builder を起動 (作成) する必要があります。Image Builder インスタンスで実行されているバックグラウンドサービスを停止しないでください。停止すると、Image Builder セッションが中断されたり、イメージ作成に支障が生じたりする場合があります。

アプリケーションのテストモードでの起動ができなくなります。

アプリケーションで、昇格されたユーザー権限、または通常は管理者のみが使用できる特別なアクセス権限が必要かどうかを確認してください。Image Builder テストモードの Image Builder インスタンスの制限されたアクセス権限は、AppStream 2.0 のテストフリートのエンドユーザーのアクセス権限と同じです。昇格されたアクセス許可を必要とするアプリケーションは、Image Builder テストモードでは起動しません。

VPC で、アプリケーションがネットワークリソースに接続できませんでした。

Image Builder が、正しい VPC サブネットで起動されたかどうかを確認してください。また、必要に応じて VPC のルートテーブルが接続を許可するように設定されていることも確認してください。

Image Builder デスクトップをカスタマイズしましたが、作成したイメージからフリートを起動してセッションに接続した後、変更内容が使用できません。

時間の設定など、ローカルユーザーセッションの一部として保存される変更は、イメージを作成するときに保持されません。ローカルユーザーセッションの変更を保持するには、Image Builder インスタンスのローカルグループポリシーに追加します。

アプリケーションを起動するときにコマンドラインパラメータがなくなります。

Image Builder を使用してアプリケーションをイメージに追加するときに、コマンドラインパラメータを指定できます。アプリケーションの起動パラメータをユーザーごとに変更しない場合は、Image Builder インスタンスでアプリケーションをイメージに追加する際に起動パラメータを入力できます。

起動ごとに起動パラメータが異なる場合は、CreateStreamingURL API を使用して、プログラムによって指定できます。API フィールドで、sessionContext パラメータと applicationID パラメータを設定します。sessionContext はアプリケーションの起動時にコマンドラインオプションとして含まれています。

起動パラメータをオンザフライで計算する必要がある場合は、スクリプトを使用してアプリケーションを起動することができます。計算されたパラメータを使用してアプリケーションを起動する前に、スクリプト内で sessionContext パラメータを解析できます。

ウイルス対策アプリケーションをインストールした後、フリートでイメージを使用できません。

イメージを作成する前に、AppStream 2.0 スタックで Image Builder を使用して、ウイルス対策プログラムを含むツールをインストールできます。ただし、これらのプログラムが、AppStream 2.0 サー

ビスで使用するネットワークポートやプロセスをブロックしないようにする必要があります。イメージを作成して、これをフリートで使用する前に、アプリケーションを Image Builder テストモードでテストすることをお勧めします。

イメージ作成に失敗しました。

イメージ作成を開始する前に AppStream 2.0 サービスに変更を加えていないことを確認してください。イメージを再度作成してみてください。失敗した場合は、AWS サポートにお問い合わせください。詳細については、[AWS サポート センター](#)を参照してください。

PrewarmManifest.txt へのアクセスが拒否されたというエラーメッセージが表示され、Image Assistant **create-image** オペレーションが失敗しました

アプリケーションの最適化マニフェストは、昇格された権限で作成されています。イメージを作成するには、次のいずれかを実行してから、もう一度試してください。

- Image Assistant コマンドラインインターフェイス (CLI) 実行可能ファイル (Image-Assistant.exe) を管理者特権で実行する。
- アプリケーションの最適化マニフェストファイルを削除する。

フリートのトラブルシューティング

以下は、ユーザーがフリートインスタンスから起動された Amazon AppStream 2.0 ストリーミングセッションに接続する場合に発生する可能性のある問題です。

問題

- [フリートの容量を増やそうとしましたが、更新が有効になりません。](#)
- [デフォルト設定の Internet Explorer を使用しないと、アプリケーションが動作しない。Internet Explorer のデフォルト設定を復元する方法を教えてください。](#)
- [フリートインスタンス間で環境変数を永続化する必要がある。](#)
- [ユーザーの Internet Explorer のデフォルトホームページを変更したい。](#)
- [ユーザーがストリーミングセッションを終了してから新しいセッションを開始すると、ストリーミングリソースが利用できないというメッセージが表示される。](#)

フリートの容量を増やそうとしましたが、更新が有効になりません。

フリートの容量は、次の 2 つの方法のいずれかで増やすことができます。

- AppStream 2.0 コンソールで、フリートの [Scaling Policies] (スケーリングポリシー) タブの [Minimum capacity] (最小容量) の値を手動で増やします。
- フリートの容量を管理するフリートスケーリングポリシーを構成することにより、自動的に行います。

手動変更またはスケーリングポリシーが、フリートのインスタンスタイプやサイズにおいて、現在の AppStream 2.0 クォータを超えた場合、新しい値は有効になりません。この問題が発生した場合は、AWS コマンドラインインターフェイス (CLI) [describe-scaling-activities](#) コマンドを使用して、キャパシティリクエストが該当するフリートインスタンスタイプとサイズのクォータを超えているかどうかを確認できます。このコマンドでは、以下のフォーマットを使用します。

```
aws application-autoscaling describe-scaling-activities
--service-namespace appstream \
--resource-id fleet/fleetname \
```

たとえば、次のコマンドは、 の **TestFleet** フリートに関する情報を提供します **us-west-2** AWS リージョン。

```
aws application-autoscaling describe-scaling-activities --service-namespace appstream
--resource-id fleet/TestFleet --region us-west-2
```

次の JSON 出力は、[Minimum capacity (最小容量)] が 150 の **TestFleet** スケーリングポリシーが設定されたことを示しています。この値は **TestFleet** の制限 (クォータ) (100) を超えているため、新しいスケーリングポリシーは有効になりません。出力での StatusMessage パラメータは、フリートインスタンスタイプ (この場合は stream.standard.medium)、現在のクォータ (100) など、エラーの原因に関する詳細情報を提供します。

Note

AppStream 2.0 インスタンスタイプとサイズクォータは、Amazon Web Services アカウントごと、AWS リージョンごとです。同じリージョン内に同じインスタンスタイプとサイズを

使用するフリートが複数存在する場合、そのリージョン内のすべてのフリートインスタンスの総数は、該当するクォータ以下でなければなりません。

```
{
  "ScalingActivities": [
    {
      "ActivityId": "id",
      "ServiceNamespace": "appstream",
      "ResourceId": "fleet/TestFleet",
      "ScalableDimension": "appstream:fleet:DesiredCapacity",
      "Description": "Setting desired capacity to 150.",
      "Cause": "minimum capacity was set to 150",
      "StartTime": 1596828816.136,
      "EndTime": 1596828816.646,
      "StatusCode": "Failed",
      "StatusMessage": "Failed to set desired capacity to 150. Reason: The Instance type 'stream.standard.medium' capacity limit for fleet TestFleet' was exceeded. Requested: 150, Limit: 100 (Service: AmazonAppStream; Status Code: 400; Error Code: LimitExceededException; Request ID: id; Proxy: null)."
```

describe-scaling-activities コマンドを実行し、容量のリクエストが現在のクォータを超えていることが出力で示されている場合は、次の方法で問題を解決できます。

- 容量のリクエストをクォータを超えない値に変更します。
- クォータ引き上げのリクエスト。クォータの引き上げをリクエストするには、[AppStream 2.0 の制限フォーム](#)を使用します。

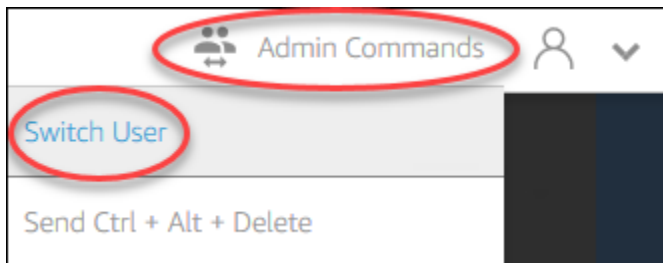
デフォルト設定の Internet Explorer を使用しないと、アプリケーションが動作しない。Internet Explorer のデフォルト設定を復元する方法を教えてください。

AppStream 2.0 環境に、要素をレンダリングするアプリケーションが含まれている場合は、Internet Explorer のデフォルト設定を復元して、インターネットへのフルアクセスを完全に有効にする必要がある場合があります。

Internet Explorer のデフォルト設定を自動的に復元するには

1. AppStream 2.0 コンソールを <https://console.aws.amazon.com/appstream2> で開きます。

2. 左側のナビゲーションペインで、[Images]、[Image Builder] の順に選択します。
3. デフォルト設定を復元する Internet Explorer で Image Builder を選択し、Running 状態であることを確認してから、[Connect] を選択します。
4. 次のいずれかを実行して、Image Builder にログインします。
 - Image Builder が Active Directory ドメインに参加していない場合、[Local User] タブで、[Template User] を選択します。
 - Image Builder が Active Directory ドメインに参加している場合は、[Directory User] タブを選択し、Image Builder に対するローカル管理者権限がないドメインユーザーの認証情報を入力して、[Log in] を選択します。
5. Internet Explorer を開いて次の手順を実行して設定をリセットします。
 - a. Internet Explorer ブラウザウィンドウの右上で、ツールアイコンを選択し、[インターネット オプション] を選択します。
 - b. [詳細設定] タブを選択し、[リセット] を選択します。
 - c. 選択の確認を求められたら、[リセット] を選択します。
 - d. [Internet Explorer の設定をリセット] メッセージが表示されたら、[閉じる] を選択します。
6. Image Builder デスクトップの右上領域で、[Admin Commands]、[Switch User] の順に選択します。



7. これにより現在のセッションが切断され、ログインメニューが開きます。次のいずれかを実行します。
 - Image Builder が Active Directory ドメインに参加していない場合、[Local User] タブで、[Administrator] を選択します。
 - Image Builder が Active Directory ドメインに参加している場合、[Directory User] タブを選択し、Image Builder でローカル監理者権限があるドメインユーザーとしてログインします。
8. Image Builder デスクトップから、Image Assistant を開きます。

9. Image Assistant で、必要な手順に従って、イメージの作成を完了します。詳細については、[「チュートリアル: AppStream 2.0 コンソールを使用してカスタム AppStream 2.0 イメージを作成する」](#)を参照してください。

フリートインスタンス間で環境変数を永続化する必要がある。

環境変数を使用すると、アプリケーション間で設定を動的に渡すことができます。これにより、フリートインスタンス間でユーザーの環境変数とシステムの環境変数を使用することができます。範囲を制限して環境変数を作成することもできます。これは、同じ環境変数をアプリケーションに応じて異なる値で使用する必要がある場合に便利です。詳細については、[「Amazon AppStream 2.0 で環境変数を保持する」](#)を参照してください。

ユーザーの Internet Explorer のデフォルトホームページを変更したい。

グループポリシーを使用して、ユーザーの Internet Explorer のデフォルトホームページを設定できます。設定したデフォルトページをユーザーが変更できるように設定することもできます。詳細については、[「Amazon AppStream 2.0 でユーザーのストリーミングセッション用に Internet Explorer のデフォルトホームページを変更する」](#)を参照してください。

ユーザーがストリーミングセッションを終了してから新しいセッションを開始すると、ストリーミングリソースが利用できないというメッセージが表示される。

ユーザーがセッションを終了すると、AppStream 2.0 では、基盤となるインスタンスを終了し、必要に応じて新しいインスタンスを作成して、フリートの必要な容量を満たします。AppStream 2.0 によって新しいインスタンスが作成され、他のすべてのインスタンスが使用される前にユーザーが新しいセッションを開始しようとする、ストリーミングリソースが利用できないというエラーがユーザーに表示されます。ユーザーがセッションを頻繁に開始および停止する場合は、フリートのキャパシティーの増加を検討してください。詳細については、[「Amazon AppStream 2.0 向け Fleet Auto Scaling」](#)を参照してください。または、セッションを終了せずに、フリートの最大セッション時間を延長して、アイドル状態の間はブラウザを閉じるようユーザーに指示することを検討してください。

Active Directory のトラブルシューティング

以下は、Amazon AppStream 2.0 で Active Directory を設定して使用するときに発生する可能性がある問題です。トラブルシューティングの通知コードのヘルプについては、「[通知コードのトラブルシューティング](#)」を参照してください。

問題

- [Image Builder とフリートインスタンスが「PENDING」状態でスタックする](#)
- [ユーザーが SAML アプリケーションを使用してログインできない](#)
- [フリートインスタンスが 1 人のユーザーに対しては機能するが、正しくサイクルしない](#)
- [ユーザーのグループポリシーオブジェクトが正常に適用されていない](#)
- [AppStream 2.0 ストリーミングインスタンスが Active Directory ドメインに参加していない。](#)
- [ドメインに参加しているストリーミングセッションでユーザーログインが完了するまで時間がかかる](#)
- [ユーザーは、ドメイン参加済みのストリーミングセッションではドメインリソースにアクセスできないが、ドメイン参加済みの Image Builder からはリソースにアクセスできる](#)
- [ユーザーに「証明書ベースの認証が利用できません」というエラーが表示され、ドメインパスワードの入力を求められる。または、証明書ベースの認証が有効になっているセッションを開始すると、「セッションから切断されました」というエラーが表示される](#)
- [Active Directory \(AD\) サービスアカウントを変更した後、ドメイン結合が失敗しています。](#)

Image Builder とフリートインスタンスが「PENDING」状態でスタックする

Image Builder およびフリートインスタンスは、準備完了状態に移行して使用できるようになるまで、最長で 25 分かかることがあります。インスタンスが使用できるようになるまでに 25 分以上かかっている場合は、Active Directory において、新しいコンピュータオブジェクトが適切な組織単位 (OU) で作成されているかどうかを確認します。新しいオブジェクトがある場合は、ストリーミングインスタンスは間もなく利用可能になります。オブジェクトがない場合は、AppStream 2.0 Directory Config でディレクトリ設定の詳細 (ディレクトリ名 (ディレクトリの完全修飾ドメイン名)、サービスアカウントのサインイン認証情報、OU 識別名)を確認します。

Image Builder とフリートのエラーは、フリートまたは Image Builder の [Notifications (通知)] タブの AppStream 2.0 コンソールに表示されます。フリートエラーは、[DescribeFleets](#) オペレーションまたは CLI コマンドの [describe-fleets](#) 経由で AppStream 2.0 API を使って利用することもできます。

ユーザーが SAML アプリケーションを使用してログインできない

AppStream 2.0 は ID プロバイダから提供される SAML_Subject の「NameID」属性に依存し、ユーザー名フィールドに入力してユーザーをログインさせます。ユーザー名は「*domain*\username」、または「user@domain.com」形式のいずれかを使用できます。*domain*\username 形式を使用している場合、*domain* は NetBIOS 名または完全修飾ドメイン名を使用できます。「user@domain.com」形式を使用する場合、UserPrincipalName 属性を使用できます。SAML_Subject 属性が正しく設定されていることを確認しても問題が解決しない場合は、AWS サポートにお問い合わせください。詳細については、[AWS サポート センター](#)を参照してください。

フリートインスタンスが 1 人のユーザーに対しては機能するが、正しくサイクルしない

フリートインスタンスは、ユーザーがセッションを完了するとサイクルし、各ユーザーが新しいインスタンスを使用するようにします。サイクルされたフリートインスタンスは、オンラインになると、以前のインスタンスのコンピュータ名を使用してドメインに参加します。このオペレーションが正常に発生するには、コンピュータオブジェクトが参加する組織単位 (OU) に対する Change Password アクセス許可と Reset Password アクセス許可がサービスアカウントに必要です。サービスアカウントのアクセス権限を確認して、もう一度試してください。問題が解決しない場合は、お問い合わせください AWS サポート。詳細については、[AWS サポート センター](#)を参照してください。

ユーザーのグループポリシーオブジェクトが正常に適用されていない

デフォルトでは、コンピュータオブジェクトは、そのコンピュータオブジェクトが存在する OU に基づいてコンピュータレベルポリシーを適用します。一方、ユーザーレベルポリシーはそのユーザーが存在する OU に基づいて適用されます。ユーザーレベルポリシーが適用されていない場合、以下のいずれかの処理を行うことができます。

- ユーザーレベルのポリシーを、ユーザーの Active Directory オブジェクトが存在する OU に移動する
- コンピュータレベルのループバック処理を有効にします。これにより、ユーザーレベルのポリシーがコンピュータオブジェクトの OU に適用されます。

詳細については、Microsoft サポートの [グループ ポリシーのループバック処理](#)を参照してください。

AppStream 2.0 ストリーミングインスタンスが Active Directory ドメインに参加していない。

AppStream 2.0 で使用する Active Directory ドメインは、ストリーミングインスタンスを起動した VPC を通じて完全修飾ドメイン名 (FQDN) を使用してアクセス可能であることが必要です。

ドメインにアクセスできることをテストするには

1. AppStream 2.0 で使用する同じ VPC、サブネット、セキュリティグループで Amazon EC2 インスタンスを起動します。
2. AppStream 2.0 で使用するサービスアカウントを使用して、EC2 インスタンスを FQDN (例: `yourdomain.example.com`) を使用して手動で Active Directory ドメインに結合します。次のコマンドを Windows PowerShell コンソールで実行します。

```
netdom join computer /domain:FQDN /OU:path /ud:user /pd:password
```

この手動による参加が失敗した場合は、次のステップに進みます。

3. 手動でドメインに結合できない場合は、コマンドプロンプトを開いて、`nslookup` コマンドを使用して FQDN を解決できることを確認します。例:

```
nslookup yourdomain.exampleco.com
```

名前解決が成功すると、有効な IP アドレスが返されます。FQDN を解決できない場合は、必要に応じてドメインの DHCP オプションセットを使用して VPC DNS サーバーを更新します。その後、このステップに戻ります。詳細については、Amazon VPC ユーザーガイドの[DHCP Options Sets](#)を参照してください。

4. FQDN が解決した場合は、`telnet` コマンドを使用して接続を検証します。

```
telnet yourdomain.exampleco.com 389
```

接続が成功した場合は、接続エラーがない空のコマンドプロンプトウィンドウが表示されます。必要に応じて EC2 インスタンスに Telnet クライアント機能をインストールします。詳細については、Microsoft ドキュメントの「[Install Telnet Client](#)」を参照してください。

EC2 インスタンスを手動でドメインに参加させることに失敗したが、FQDN の解決と Telnet クライアントとの接続テストに成功した場合は、VPC セキュリティグループがアクセスをブロックし

ている可能性があります。Active Directory では特定のネットワークポート設定が必要です。詳細については、Microsoft ドキュメントの [Active Directory and Active Directory Domain Services Port Requirements](#) を参照してください。

ドメインに参加しているストリーミングセッションでユーザーログインが完了するまで時間がかかる

AppStream 2.0 は、ユーザーがドメインパスワードを入力した後に Windows ログインアクションを実行します。認証に成功したら、AppStream 2.0 はアプリケーションを起動します。ログインと起動時間は、ドメインコントローラーへのネットワークの競合やグループポリシー設定をストリーミングインスタンスに適用するためにかかる時間など、多くの変動要素の影響を受けます。ドメイン認証の完了に時間がかかり過ぎる場合、次のアクションを実行してください。

- 正しいドメインコントローラーを選択して、AppStream 2.0 リージョンからドメインコントローラーへのネットワークのレイテンシーを最小限に抑えます。たとえば、フリートが us-east-1 にある場合は、[Active Directory サイトとサービス] ゾーンマッピングを使用して us-east-1 への帯域幅が広くレイテンシーが低いドメインコントローラーを使用します。詳細については、Microsoft ドキュメントの「[Active Directory サイトとサービス](#)」を参照してください。
- グループポリシー設定とユーザーログインスクリプトの適用や実行に著しく時間がかかっていないことを確認します。

ドメインユーザーによる AppStream 2.0 へのログインが失敗し、「不明なエラーが発生しました」というメッセージが表示された場合は、必要に応じて [Amazon AppStream 2.0 でアクティブディレクトリの使用を開始する前に](#) の説明に従ってグループポリシー設定を更新します。更新しないと、AppStream 2.0 によるドメインユーザーの認証とログインがブロックされる場合があります。

ユーザーは、ドメイン参加済みのストリーミングセッションではドメインリソースにアクセスできないが、ドメイン参加済みの Image Builder からはリソースにアクセスできる

Image Builder と同じ VPC、サブネット、およびセキュリティグループでフリートが作成されていること、およびドメインリソースにアクセスして使用するためのアクセス許可をユーザーに付与していることを確認します。

ユーザーに「証明書ベースの認証が利用できません」というエラーが表示され、ドメインパスワードの入力を求められる。または、証明書ベースの認証が有効になっているセッションを開始すると、「セッションから切断されました」というエラーが表示される

これらのエラーは、証明書ベースの認証がセッションで失敗した場合に発生します。証明書ベースの認証を有効にしてパスワードによるログオンにフォールバックできるようにすると、「証明書ベースの認証は利用できません」というエラーが表示されます。証明書ベースの認証がフォールバックなしで有効になっている場合、「セッションから切断されました」というエラーが表示されます。

ユーザーは、ウェブクライアントでページを更新するか、Windows 用クライアントから再接続できます。これは、証明書ベースの認証では断続的に発生する問題である可能性があるためです。問題が続く場合は、以下のいずれかの問題が原因で証明書ベースの認証が失敗する可能性があります。

- AppStream 2.0 が AWS プライベート CA と通信できなかったか、プライベート CA AWS が証明書を発行しませんでした。CloudTrail をチェックして、証明書が発行されたかどうかを確認します。詳細については、「[とは AWS CloudTrail](#)」および「[the section called “証明書ベースの認証の管理”](#)」を参照してください。
- ドメインコントローラには、スマートカードログオン用のドメインコントローラー証明書がないか、有効期限が切れている。詳細については、「[the section called “前提条件”](#)」のステップ 7. a を参照してください。
- 証明書が信頼されていない。詳細については、「[the section called “前提条件”](#)」のステップ 7. c を参照してください。
- SAML_Subject NameID の userPrincipalName 形式が正しくフォーマットされていないか、ユーザーの実際のドメインに解決しない。詳細については、「[the section called “前提条件”](#)」のステップ 1 を参照してください。
- SAML アサーションの (オプションの) ObjectSid 属性が、SAML_Subject NameID で指定されたユーザーの Active Directory セキュリティ識別子 (SID) と一致しない。SAML フェデレーションの属性マッピングが正しいこと、および SAML ID プロバイダーが Active Directory ユーザーの SID 属性を同期していることを確認します。
- AppStream 2.0 エージェントが証明書ベースの認証をサポートしていない。AppStream 2.0 エージェントのバージョン 10-13-2022 以降を使用します。
- スマートカードログオンのデフォルトの Active Directory 設定を変更したり、スマートカードがスマートカードリーダーから取り出された場合にアクションを実行したりするグループポリシー設定があります。これらの設定により、上記のエラー以外にも予期しない動作が発生する可能性があります。

があります。証明書ベースの認証では、インスタンスのオペレーティングシステムに仮想スマートカードが提示され、ログオンが完了すると削除されます。詳細については、「[スマートカードのプライマリグループポリシー設定](#)」および「[その他のスマートカードのグループポリシー設定とレジストリキー](#)」を参照してください。証明書ベースの認証を使用する場合は、スタック内で Active Directory のスマートカードサインインを有効にしないでください。詳細については、「[the section called “スマートカード”](#)」を参照してください。

- プライベート CA の CRL ディストリビューションポイントがオンラインになっていないか、AppStream 2.0 フリートインスタンスまたはドメインコントローラーからアクセスできない。詳細については、「[the section called “前提条件”](#)」のステップ 5 を参照してください。

その他のトラブルシューティング手順には、AppStream 2.0 インスタンスの Windows イベントログの確認が含まれます。ログオンに失敗したかどうかを確認する一般的なイベントとして、「[4625\(F\): アカウントがログオンできませんでした](#)」があります。ログ情報の取得の詳細については、「[アプリケーションログと Windows イベントログの永続化](#)」を参照してください。または、管理者としてアクティブな AppStream 2.0 セッションをトラブルシューティングするには、別のコンピューターでイベントビューアーを使用してログに接続できます。詳細については、「[イベントビューアーでコンピューターを選択する方法](#)」を参照してください。または、リモートデスクトップを使用して、AppStream 2.0 仮想プライベートクラウド (VPC) のリモートデスクトップサービスに接続できる別のコンピューターからインスタンスのプライベート IP アドレスに接続することもできます。AWS CLI を使用して、AWS リージョン、AppStream 2.0 スタック名、フリート名、ユーザー ID、認証タイプに基づいてセッションの IP アドレスを決定します。詳細については、「[AWS Command Line Interface](#)」を参照してください。

問題が解決しない場合は、お問い合わせください AWS サポート。詳細については、[AWS サポートセンター](#)を参照してください。

Active Directory (AD) サービスアカウントを変更した後、ドメイン結合が失敗しています。

2024 年 8 月の「[Microsoft Windows Server オペレーティングシステムの更新](#)」に基づくイメージを持つ既存のフリートがあり、そのフリートの Active Directory (AD) サービスアカウントを変更すると、フリートインスタンスがプロビジョニング中にドメイン結合に失敗する可能性があります。

Microsoft は、ドメイン結合オペレーションの動作を変更するパッチ [KB5020276](#) をリリースしました。AppStream 2.0 は、ストリーミングインスタンスを AD ドメインに結合するときに、既存のコンピュータオブジェクトを再利用します。このコンピュータオブジェクトは、AppStream 2.0 でフリートまたは Directory Config を作成するときに指定した AD サービスアカウントを使用して生成さ

れます。この Microsoft パッチ以前は、新しい AD サービスアカウントは、組織単位 (OU) で「コンピュータオブジェクトの作成」アクセス許可が設定されている限り、AppStream 2.0 によって作成された既存のコンピュータオブジェクトを再利用できます。

2024 年 8 月 13 日以降に Microsoft パッチが適用され、既存の AppStream 2.0 フリートの AD サービスアカウントを変更した場合、新しいサービスアカウントは AD 内の既存のコンピュータオブジェクトを再使用することはできません。これにより、AppStream 2.0 フリートでドメイン結合が失敗し、フリート通知に次のいずれかのエラーメッセージが表示されます。

- DOMAIN_JOIN_INTERNAL_SERVICE_ERROR 「グループ名が見つかりませんでした。」
- 同じ名前のアカウントが Active Directory に存在します。アカウントの再利用がセキュリティポリシーによってブロックされました

既存のコンピュータオブジェクトを再利用できるアカウントを制御するために、Microsoft はドメインコントローラー: ドメイン結合中のコンピュータアカウントの再利用を許可するという新しいグループポリシー設定を実装しました。この設定では、ドメイン結合オペレーション中にチェックをバイパスする信頼されたサービスアカウントのリストを指定できます。セルフマネージド AD 設定では、ドメインコントローラーのグループポリシーを使用して、[Microsoft が文書化した手順](#)に従って、AD サービスアカウントを新しい許可リストポリシーに追加することをお勧めします。

Managed Active Directory (MAD) の場合、AppStream 2.0 ドメイン参加サービスアカウントを変更した後、AppStream 2.0 フリートを再起動する必要があります。

問題が解決しない場合は、[お問い合わせ](#)してください AWS サポート。詳細については、[AWS サポートセンター](#)を参照してください。

AppStream 2.0 ユーザーの問題のトラブルシューティング

高度なログ記録を有効にする

ユーザーが経験する可能性のある問題のトラブルシューティングに役立つように、任意の AppStream 2.0 クライアントで高度なログ記録を有効にできます。高度なログ記録では、診断情報とデバッグレベルの詳細 (詳細なパフォーマンスデータなど) を含むログファイルが生成されます。

 Note

高度なログファイル AWS を確認し、AppStream 2.0 クライアントに関する問題のテクニカルサポートを受けるには、[お問い合わせください サポート](#)。詳細については、「[AWS Support Center Console](#)」を参照してください。

ウェブアクセスの高度なログ記録を有効にする

ユーザーが SAML、ユーザープールを使用している場合、または Application Catalogue ページにアクセスできる場合は、次の手順に従います。

1. カタログページをロードします。
2. デベロッパーツールを開き、コンソールタブを選択します。
3. ブラウザコンソールで、「」と入力 `window.siteConfig.logLevel = "INFO"` し、「Enter」を選択します。
4. アプリケーションを起動すると、コンソールタブにログ記録が表示されます。
5. 問題を再現します。
6. コンソールタブを右クリックし、すべてのメッセージをファイルに保存を選択します。

Windows クライアントの高度なログ記録を有効にする

Windows クライアントの高度なログ記録を有効にするには、次の手順に従います。

1. クライアントマシンで、に移動します `%localappdata%\AppStreamClient\app-<versionID>`。
2. メモ帳 `Log4Net.config` で を開きます。
3. ログ記録のルートレベルを INFO から DEBUG に変更します。
4. ファイルを保存します。
5. AppStream 2.0 クライアントを再起動し、接続を再試行してください。
6. 完全なフォルダを圧縮 `C:\Users\%USERNAME%\AppData\Local\Amazon\AppStreamClient\'` して、 からログを収集します。

以下は、ユーザーが AppStream 2.0 を使用する際に発生する可能性のある特定の問題です。

問題

- ユーザーによる AppStream 2.0 クライアントのインストールが失敗し、.NET Framework 4.6 が必要であるというメッセージが表示されます。
- ユーザーが AppStream 2.0 クライアントをインストールするときに、USB ドライバーのインストールが失敗し、AppStream 2.0 で USB デバイスを使用できなくなります。
- 私の AppStream 2.0 クライアントユーザーは、60 分ごとに AppStream 2.0 セッションから接続が解除されます。
- ユーザーがローカルデバイスとストリーミングセッションの間でコピーと貼り付けができません。
- 一部のキーボードショートカットは、ストリーミングセッション中に動作しません。
- ユーザーのドロ잉タブレットは、自分がデプロイしたストリーミングアプリケーションで動作していません。
- ユーザーのストリーミングセッション中に日本語入力方法が機能しない
- ユーザーがアプリケーションカタログからアプリケーションを起動しようとする、ストリーミングセッションの最大数到達に関するエラーが表示されます。
- ユーザーに黒い画面またはデスクトップが表示され、ユーザーのアプリケーションが Elastic フリートで起動されません。エラーは表示されません。

ユーザーによる AppStream 2.0 クライアントのインストールが失敗し、.NET Framework 4.6 が必要であるというメッセージが表示されます。

ユーザーが AppStream 2.0 クライアントをインストールすると、.NET Framework バージョン 4.6.2 も AppStream 2.0 によってインストールされます (このバージョン以降がまだインストールされていない場合)。クライアントのインストール先の PC がインターネットに接続されていない場合、.NET Framework をインストールすることはできません。この場合、.NET Framework バージョン 4.6 を手動でインストールするように求めるメッセージが表示されます。ただし、ユーザーが [インストール] を選択すると、インストールが失敗したことを示すエラーメッセージが表示されます。次に、ユーザーは最新バージョンの .NET Framework を手動でインストールするように求められます。ユーザーが [閉じる] を選択すると、インストールが終了します。

この問題を解決するには、クライアントのインストール先の PC からインターネット接続を確立し、同じ PC に .NET Framework バージョン 4.6.2 以降をダウンロードしてインストールする必要があります。ダウンロードできる .NET Framework のバージョンの一覧については、「[Download .NET Framework](#)」を参照してください。

Note

AppStream 2.0 クライアントバージョン 1.1.156 をインストールしているユーザーは、同じ PC に .NET Framework バージョン 4.7.2 以降をインストールする必要があります。

ユーザーが AppStream 2.0 クライアントをインストールするときに、USB ドライバーのインストールが失敗し、AppStream 2.0 で USB デバイスを使用できなくなります。

ユーザーは AppStream 2.0 クライアントをインストールするときに、AppStream 2.0 USB ドライバーをインストールするかどうかを選択します。ドライバーは、AppStream 2.0 を通じてストリーミングされたアプリケーションで USB デバイスを使用するために必要です。ただし、次の両方の問題があると、USB ドライバーのインストールは失敗します。

- AppStreamUsbDriver.exe ファイルの署名に使用するルート証明書が Windows 証明書ストアに存在しない。
- クライアントのインストール先の PC がインターネットに接続されていない。

この場合、Amazon AppStream USB ドライバーの証明書を検証できないため、USB ドライバーのインストールが失敗したというエラーメッセージがユーザーに表示されます。ユーザーが [OK] を選択すると、USB ドライバーなしで AppStream 2.0 クライアントのインストールが完了します。ユーザーは引き続き AppStream 2.0 クライアントでアプリケーションをストリーミングできますが、AppStream 2.0 を通じてストリーミングされたアプリケーションで USB デバイスを使用することはできません。

この問題を解決するには、ユーザーが AppStream 2.0 クライアントのインストール先の PC からインターネット接続を確立し、クライアントを再インストールする必要があります。

私の AppStream 2.0 クライアントユーザーは、60 分ごとに AppStream 2.0 セッションから接続が解除されます。

AppStream 2.0 へのアクセスに SAML 2.0 を使用して ID フェデレーションを設定している場合、ID プロバイダー (IdP) によっては、認証レスポンス AWS の一部として IdP が SAML 属性として渡す情報を設定する必要がある場合があります。これには、[Attribute] 要素の設定として、SessionDuration 属性を <https://aws.amazon.com/SAML/Attributes/SessionDuration> に設定することが含まれます。

SessionDuration は、再認証が必要となるまでに、ユーザーのフェデレーティッドストリーミングセッションをアクティブにしておくことができる最大時間を指定します。SessionDuration はオプションの属性ですが、これを SAML 認証レスポンスに含めることをお勧めします。この属性を指定しない場合、セッション時間はデフォルト値の 60 分に設定されます。

この問題を解決するには、SAML 認証レスポンスに SessionDuration 値を含めるように SAML 互換 IdP を設定し、必要に応じた値を設定します。詳細については、「[ステップ 5: SAML 認証レスポンスのアサーションを作成する](#)」を参照してください。

Note

ユーザーが AppStream 2.0 ネイティブクライアントを使用するか、新しいエクスペリエンスでウェブブラウザを使用して AppStream 2.0 のストリーミングアプリケーションにアクセスしている場合、ユーザーのセッション時間が期限切れになると、セッションは切断されます。ユーザーが古い/クラシックエクスペリエンスでウェブブラウザを使用して AppStream 2.0 のストリーミングアプリケーションにアクセスしている場合、ユーザーのセッション時間が期限切れになってユーザーがブラウザページを更新すると、セッションは切断されます。

ユーザーが SAML 互換 IdP を使用して新しいポータルエクスペリエンスにサインインし、ランダムな切断が続く場合、AppStream 2.0 セッションで使用されるセッション Cookie が、サブドメインとして `aws.amazon.com` を使用する他のウェブアプリケーションによって無効になっている可能性があります。以下に一般的なユーザーシナリオを示します。

- ユーザーが同じブラウザで新しい AppStream 2.0 セッションを開始すると、既存の AppStream 2.0 セッションが切断される。
- ユーザーが同じブラウザで他のウェブアプリケーションを起動し、`aws.amazon.com` ドメインで新しいユーザー認証が発生した場合、既存の AppStream 2.0 セッションが切断される。
- ユーザーが同じブラウザで新しい IAM 認証情報 AWS Management Console を使用してサインインすると、既存の AppStream 2.0 セッションは切断されます。

この問題を解決するには、新しいリレーステートエンドポイントを使用して SAML 2.0 フェデレーションを設定し、AppStream 2.0 クライアントバージョン 1.1.1300 以降を使用します。詳細については、「[the section called “ステップ 6: フェデレーションのリレーステートを設定する”](#)」の表 1 を参照してください。

ユーザーがローカルデバイスとストリーミングセッションの間でコピーと貼り付けができません。

AppStream 2.0 はウェブアプリケーションの非同期クリップボード操作を有効にするために [W3C 仕様](#) を利用します。これにより、ユーザーはローカルデバイスとストリーミングセッションの間でコンテンツをコピーして貼り付けることができます。これはキーボードショートカットの使用を含め、ローカルデバイスのアプリケーション間でコピーして貼り付けする方法と同じです。

W3C の非同期クリップボード仕様を現在サポートしている唯一のブラウザは Google Chrome バージョン 66 以降です。テキストのみのコピーと貼り付けをサポートします。他のすべてのブラウザでは、ユーザーは AppStream 2.0 ウェブポータルのクリップボード機能を使用できます。これにはコピー/貼り付けのためのダイアログボックスがあります。

ユーザーがストリーミングセッション中にクリップボードを使用して問題が発生した場合、次の情報を提供できます。

- Chrome バージョン 66 以降を使用しているが、キーボードショートカットが機能しない。

Chrome では、AppStream 2.0 に、クリップボードにコピーされたコンテンツへのアクセスを許可するかどうかを選択するためのプロンプトが表示されます。リモートセッションへの貼り付けを有効にするには、[Allow] を選択します。リモートセッションからローカルデバイスにテキストをコピーする場合は、ストリーミングセッションからローカルデバイスにテキストがコピーされるのに十分な時間、Chrome アプリケーションとストリーミングセッションを含むタブの両方に、フォーカスが留まっている必要があります。小さいサイズのテキストはほぼ瞬時にコピーされますが、大きいサイズのテキストの場合は、Chrome またはストリーミングセッションが含まれているタブから切り替わるまでに 1~2 秒待つことがあります。テキストコピーに要する時間は、ネットワーク状態によって異なります。

- 大きいサイズのテキストをコピーして貼り付けようとすると、コピーと貼り付けが機能しない。

AppStream 2.0 では、ローカルデバイスとストリーミングセッションの間でコピーして貼り付けることができるテキストのサイズはデフォルトで 20 MB に制限されます。20 MB を超えるテキストをコピーしようとした場合、何もコピーされません。ただし、管理者が制限を設定し、その制限を超えた場合、テキストは切り捨てられます。ローカルデバイスのアプリケーション間や、ストリーミングセッションのアプリケーション間では、テキストをコピーして貼り付ける場合に、この制限はありません。管理者は、ストリーミングセッションでコピー/貼り付けする文字数を制限することもできます。ローカルデバイスとストリーミングセッションの間で 20 MB または指定された制限より大きいテキストをコピーまたは貼り付ける場合は、小さなサイズに分割するか、ファイルとしてアップロードできます。

- AppStream 2.0 ウェブポータルクリップボード機能を使用して、テキストをストリーミングセッションに貼り付ける場合に、動作しない。

場合によっては、テキストをクリップボードダイアログボックスに貼り付け、ダイアログボックスを閉じた後、ストリーミングセッションでテキストを貼り付けるためにキーボードショートカットを使用しても何も起こりません。この問題が発生するのは、クリップボードダイアログボックスが表示されたとき、ストリーミングアプリケーションからフォーカスが離れるからです。ダイアログボックスを閉じた後、ストリーミングアプリケーションに自動的にフォーカスが返らない場合があります。ストリーミングアプリケーションをクリックするとフォーカスが返り、キーボードショートカットを使用して、ストリーミングセッションにテキストを貼り付けできるようになります。

一部のキーボードショートカットは、ストリーミングセッション中に動作しません。

以下のキーボードショートカットは、ユーザーのローカルコンピュータで動作しますが、AppStream 2.0 ストリーミングセッションに渡されません。

Windows:

- Win+L
- Ctrl+Alt+Del

Mac:

- Ctrl+F3
- Alt または オプションキーを使用する、すべてのショートカットキーの組み合わせ

この問題はユーザーのローカルコンピュータで以下の制限があるため発生します。

- ユーザーのローカルコンピュータで実行されているオペレーティングシステムにより、キーボードショートカットがフィルタ処理され、ユーザーが AppStream 2.0 にアクセスしているブラウザに渡されません。該当する動作は、Windows の Win+L および Ctrl+Alt+Del キーボードショートカット、および Mac の Ctrl+F3 キーボードショートカットです。
- ウェブアプリケーションとともに使用した場合、一部のキーボードショートカットはブラウザでフィルタ処理され、ウェブアプリケーションに対してイベントを生成しません。その結果、ウェブ

アプリケーションはユーザーによって入力されたキーボードショートカットに応答することはできません。

- キーボードイベントが生成される前にブラウザによって変換されたキーボードショートカットは、正しく変換されません。たとえば、Mac コンピューターでの Alt キーの組み合わせとオプションキーの組み合わせは、Windows での Alt グラフキーの組み合わせであるかのように変換されます。これが発生すると、ユーザーがこれらのキーの組み合わせを使用したときの意図とは異なる結果になります。

ユーザーのドローイングタブレットは、自分がデプロイしたストリーミングアプリケーションで動作していません。

ユーザーのドローイングタブレットがストリーミングアプリケーションで動作しない場合は、要件を満たしていること、およびこの機能を有効にするための追加の考慮事項を理解していることを確認してください。AppStream 2.0 ストリーミングセッション中にユーザーがドローイングタブレットを使用できるようにするための要件と考慮事項を次に示します。

Note

ドローイングタブレットは、AppStream 2.0 クライアントを使用するか、サポートされているウェブブラウザを使用して AppStream 2.0 にアクセスするユーザーに対してサポートされています。

- 顧客がこの機能を利用できるようにするには、Windows Server 2019 を実行するためのイメージを使用するように AppStream 2.0 フリートを設定する必要があります。
- この機能を使用するには、ユーザーは AppStream 2.0 クライアントを使用するか、Google Chrome または Mozilla Firefox ブラウザでのみ AppStream 2.0 にアクセスする必要があります。
- ストリーミングアプリケーションは、Windows Ink テクノロジーをサポートしている必要があります。詳細については、「[Windows アプリでのペン操作と Windows インク](#)」を参照してください。
- GIMP などの一部のアプリケーションでは、圧力感度をサポートするために、ストリーミングインスタンス上のドローイングタブレットを検出する必要があります。この場合、ユーザーは AppStream 2.0 クライアントを使用して AppStream 2.0 にアクセスし、これらの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。さらに、ユーザーのドローイングタブレットを認定する必要があります。ユーザーは、新しいストリーミングセッションを開始するたびに、ドローイングタブレットを AppStream 2.0 と共有する必要があります。

- この機能は Chromebook ではサポートされていません。

ユーザーのストリーミングセッション中に日本語入力方法が機能しない

AppStream 2.0 ストリーミングセッション中にユーザーが日本語入力方法を使用できるようにするには、次の手順を実行します。

- 日本語入力方法を使用するようにフリートを設定します。これを行うには、イメージの作成時に Image Builder で日本語入力方法を有効にし、そのイメージを使用するようにフリートを設定します。詳細については、「[デフォルトの入力メソッドを指定する](#)」を参照してください。これにより、AppStream 2.0 で日本語キーボードを使用するように自動的にイメージが構成されます。詳細については、「[日本語キーボード](#)」を参照してください。
- ユーザーのローカルコンピュータで、日本語入力方法も有効になっていることを確認します。

フリートインスタンスとユーザーのローカルコンピュータが同じ言語入力方法を使用していない場合、ユーザーのストリーミングセッション中にフリートインスタンスに予期しないキーボード入力が発生する可能性があります。たとえば、フリートインスタンスが日本語入力方法を使用し、ユーザーのローカルコンピュータが英語入力方法を使用する場合、ストリーミングセッション中に、ローカルコンピュータは、フリートインスタンスとは異なるキーマッピングを持つフリートインスタンスにキーを送信します。

フリートインスタンスで日本語入力方法が有効になっているかどうかを確認するには、フリートの [Desktop (デスクトップ)] ストリームビューを有効にします。詳細については、[Amazon AppStream 2.0 でフリートを作成する](#) のステップ 6 を参照してください。

Windows のキーボードショートカット

日本語入力モードの切り替えと日本語変換用の Windows キーボードショートカットを次に示します。これらのキーボードショートカットを使用するには、AppStream 2.0 ストリーミングセッションがアクティブである必要があります。

日本語入力モードを切り替えるための Windows キーボードショートカット

キーボードショートカット	説明
半角/全角/漢字 (半角/全角/漢字)	入力モードを英数字モードと日本語モードに切り替えます

キーボードショートカット	説明
もしくは Alt + `	
無変換 (無変換)	文字をひらがな、全角カタカナ、半角カタカナに順番に変換
カタカナ/ひらがな/ローマ字 (カタカナ/ひらがな/ローマ字)	入力モードをひらがなに変更します
Shift + カタカナ/ひらがな/ローマ字 (カタカナ/ひらがな/ローマ字)	入力モードをカタカナに変更
Alt + カタカナ/ひらがな/ローマ字 (カタカナ/ひらがな/ローマ字)	入力モードを日本語のローマ字と日本語仮名の間で切り替えます

日本語変換用の Windows キーボードショートカット

キーボードショートカット	説明
変換 (Henkan) + スペース	変換オプションを表示
F6	ひらがなに変換
F7	全角カタカナに変換
F8	半角カタカナに変換
F9	全角ローマ字に変換
F10	半角ローマ字に変換

Mac のキーボードショートカット

日本語入力方法を切り替えるための Mac キーボードショートカットおよび日本語変換については、Mac サポートドキュメントの次の記事を参照してください。

Note

AppStream 2.0 ストリーミングセッションは Windows インスタンスで実行されるため、Mac ユーザーには異なるキーマッピングが発生する可能性があります。

- 日本語入力方法を切り替えるためのキーボードショートカット — [Mac で日本語入力ソースを設定して切り替える](#)
- 日本語変換用のキーボードショートリンクカット — [Mac での日本語変換用キーボードショートカット](#)

ユーザーがアプリケーションカタログからアプリケーションを起動しようとする、ストリーミングセッションの最大数到達に関するエラーが表示されます。

AppStream 2.0 の Elastic フリートでは、最大同時実行数パラメータを使用して、同時にストリーミングできるユーザーの最大数を指定します。この値を超えてストリーミングしようとするユーザーには、このエラーが表示されます。この問題を解決するには、同時ストリームの最大数を増やすか、ユーザーに別のユーザーがストリーミングセッションを完了するまで待機するように勧めることができます。

Note

制限の引き上げをリクエストして、インスタンスタイプとサイズの制限を引き上げる必要が生じる場合もあります。

ユーザーに黒い画面またはデスクトップが表示され、ユーザーのアプリケーションが Elastic フリートで起動されません。エラーは表示されません。

この問題は、アプリケーションの起動パスが誤っていて、AppStream 2.0 がアプリケーションを起動できない場合に発生することがあります。アプリケーションの起動パスは、フリートで Desktop ビューを使用してルートボリュームをナビゲートすることによって検証できます。アプリケーションの実行可能ファイルが指定されたパスに存在することを検証します。

ストリーミングインスタンスで App Block の VHD またはセットアップスクリプトを見つけられない場合は、AppStream 2.0 がそれらを S3 バケットからダウンロードできなかった可能性があります。指定した VPC が S3 にアクセスできることを検証します。詳細については、「[AppStream 2.0 機能に対する Amazon S3 VPC エンドポイントの使用](#)」を参照してください。

永続ストレージ問題のトラブルシューティング

Amazon AppStream 2.0 では、永続的なストレージについて、ホームフォルダ、Google Drive for G Suite、ビジネス向け OneDrive のオプションがサポートされています。コンテンツ同期の動作は、これらの永続的なストレージソリューション全体で一貫しているため、予想される動作に関しては[ホームフォルダコンテンツの同期](#)で情報を確認することをお勧めします。

以下に、ユーザーまたはユーザーが AppStream 2.0 永続ストレージを使用する場合に発生する可能性がある問題を示します。

問題

- [スタックのホームフォルダが正常に機能していません。](#)
- [ユーザーがいずれかのアプリケーションからホームフォルダディレクトリにアクセスできない。](#)
- [ユーザーがアプリケーションの 1 つからホームフォルダにアクセスすると、「Device is not ready」というエラーメッセージが表示されます。](#)
- [Amazon S3 ユーザーのホームフォルダにあるファイルを削除したり置き換えたりしましたが、ストリーミングセッション中にフリートインスタンスのホームフォルダの変更がユーザーに表示されません。](#)
- [永続ストレージが预期したとおりに動作しません。ユーザーのファイルは、永続ストレージに保存するのに予想以上に時間がかかっています。](#)
- [私のユーザーは、ファイルを使用していないときにファイルがすでに使用されているというエラーを受け取っています。](#)
- [フォルダに何千ものファイルが含まれている場合、AppStream 2.0 でファイルの一覧を表示するときに時間がかかることがあります。](#)

スタックのホームフォルダが正常に機能していません。

S3 バケットに対するホームフォルダのバックアップの問題は、次のシナリオで発生することがあります。

- ストリーミングインスタンスからのインターネット接続がないか、プライベート Amazon S3 VPC エンドポイントにアクセスできない (該当する場合)。
- ネットワーク帯域幅の消費量が大きすぎる。たとえば、サービスを使用して、大きなファイルを含むホームフォルダを Amazon S3 にバックアップしようとしているときに、ユーザーが複数の大きなファイルをダウンロードまたはストリーミングしている場合があります。
- ファイルは 5 GB を超えています。
- 管理者が、サービスによって作成されたバケットを削除した。
- 管理者が AmazonAppStreamServiceAccess サービスロールの Amazon S3 アクセス許可を誤って編集した。

詳細については、[Amazon Simple Storage Service ユーザーガイド](#)を参照してください。

ユーザーがいずれかのアプリケーションからホームフォルダディレクトリにアクセスできない。

アプリケーションによっては、File Explorer の最上位フォルダとしてホームフォルダを表示するリダイレクトが認識されない場合があります。このような場合、ユーザーがストリーミングセッション中にアプリケーション内からホームフォルダにアクセスするには、アプリケーションインターフェイスから [File Open (ファイルを開く)] を選択して以下のいずれかのディレクトリを参照します。

- ドメインに参加していない Windows インスタンス: C:\Users\PhotonUser\My Files\Home Folder
- ドメインに参加済みの Windows インスタンス: C:\Users\%username%\My Files\Home Folder
- Linux インスタンス: ~/myfiles/HomeFolder

ユーザーがアプリケーションの 1 つからホームフォルダにアクセスすると、「Device is not ready」というエラーメッセージが表示されます。

永続的ストレージのマウントは、ユーザーがログインした後に行われ、数秒かかることがあります。永続的ストレージのマウントが完了する前に、アプリケーションがホームフォルダからファイルにアクセスしようとする、と、「デバイスの準備が整っていない」エラーが発生する可能性があります。数分待ってからもう一度試すことをお勧めします。

この問題を回避するには、セッションスクリプトを使用してストレージのマウントステータスをモニタリングできます。次に、マウントが完了したらストリーミングセッションを開始します。これによ

り、エンドユーザーのエクスペリエンスも向上します。詳細については、「[the section called “ユーザーのストリーミングエクスペリエンスを管理するセッションスクリプト”](#)」を参照してください。

Amazon S3 ユーザーのホームフォルダにあるファイルを削除したり置き換えたりしましたが、ストリーミングセッション中にフリートインスタンスのホームフォルダの変更がユーザーに表示されません。

S3 バケット内のユーザーのホームフォルダに保存されているコンテンツと、そのストリーミングセッション中にフリートインスタンス上のユーザーが利用できるコンテンツに違いが生まれるのは、Amazon S3 バケットに保存されたホームフォルダコンテンツと AppStream 2.0 フリートインスタンスに保存されたホームフォルダコンテンツと同期する方法が原因である可能性があります。

ユーザーの AppStream 2.0 ストリーミングセッションの開始時に、AppStream 2.0 が、Amazon Web Services アカウントとリージョンの Amazon S3 バケットに保存されているユーザーのホームフォルダファイルを格納します。ユーザーがストリーミングアプリケーションを使用してフリートインスタンスのホームフォルダにあるファイルを開くと、AppStream 2.0 はファイルをフリートインスタンスにダウンロードします。

アクティブなストリーミングセッション中にフリートインスタンス上のファイルに対してユーザーが加えた変更は、数秒ごとに S3 バケットのホームフォルダにアップロードされるか、ユーザーのストリーミングセッションの最後にアップロードされます。

ユーザーがストリーミングセッション中にフリートインスタンス上のホームフォルダにあるファイルを開き、変更を加えたりファイルを保存したりせずにファイルを閉じた場合、ストリーミングセッション中に S3 バケット内のユーザーのホームフォルダからファイルを削除すると、ユーザーがフォルダを更新したときにそのファイルがフリートインスタンスから削除されます。ユーザーがファイルを変更してローカルに保存した場合、ユーザーは、現在のストリーミングセッション中にフリートインスタンス上でそのファイルを引き続き使用できます。ファイルは S3 バケットに再びアップロードされます。ただし、フリートインスタンス上のユーザーが次のストリーミングセッション中にファイルを使用できない場合があります。

ユーザーの次のストリーミングセッション中にフリートインスタンスでファイルが利用可能かどうかは、S3 バケットでファイルを変更する前に、または変更した後に、ユーザーがフリートインスタンスでファイルを変更したかどうかによって異なります。

詳細については、「[ホームフォルダコンテンツの同期](#)」を参照してください。

永続ストレージが予期したとおりに動作しません。ユーザーのファイルは、永続ストレージに保存するのに予想以上に時間がかかっています。

AppStream 2.0 ストリーミングセッションでは、処計算集約型アプリケーションに関連付けられた大きなファイルとディレクトリを永続ストレージに保存すると、基本的な生産性アプリケーションに必要なファイルとディレクトリを保存するよりも時間がかかる場合があります。たとえば、アプリケーションが大量のデータを保存したり、同じファイルを頻繁に変更したりする場合は、1 回の書き込み操作を実行するアプリケーションによって作成されたファイルを保存する場合よりも時間がかかる場合があります。また、多くの小さなファイルを保存するのに時間がかかる場合があります。

コンピューティング集約型アプリケーションに関連付けられたファイルとディレクトリをユーザーが保存し、AppStream 2.0 永続ストレージオプションが期待どおりに動作しない場合は、Amazon FSx for Windows File Server や AWS Storage Gateway ファイルゲートウェイなどのサーバーメッセージブロック (SMB) ソリューションを使用することをお勧めします。以下は、これらの SMB ソリューションでの使用に適した、計算集約型アプリケーションに関連するファイルとディレクトリの例です。

- 統合開発環境 (IDE) 用の Workspace フォルダ
- ローカルデータベースファイル
- グラフィックシミュレーションアプリケーションによって作成されたスクラッチスペースフォルダ

詳細については、以下を参照してください。

- [Amazon FSx for Windows File Server Windows ユーザーガイド](#)
- [Using Amazon FSx with Amazon AppStream 2.0](#)
- AWS Storage Gateway ユーザーガイドの [ファイルゲートウェイ](#)

Note

さらにトラブルシューティングを進める前に、まず、ユーザーがファイルとディレクトリを保存する際に発生している問題が AppStream 2.0 永続ストレージにのみ関連しており、別の原因ではないことを確認してください。他の原因を除外するには、ストリーミングインスタンスで使用可能な [Temporary Files (一時ファイル)] ディレクトリに、ユーザーがファイルまたはディレクトリを保存できるように許可します。

私のユーザーは、ファイルを使用していないときにファイルがすでに使用されているというエラーを受け取っています。

この現象は、通常、次の場合に発生します。

- ファイルが最後に保存された後も、ユーザーのファイルがまだアップロードされているとき
- ファイルが頻繁に変更されるとき (データベースファイルなど)

サイズの大きいファイルのアップロードにはかなりの時間がかかる場合があります。また、ファイルをアップロードしようとするたびに、別のファイルが更新され、ファイルのアップロードが繰り返し試行されることがあります。

この問題を解決するには、Amazon FSx for Windows File Server や AWS Storage Gateway ファイルゲートウェイなどのサーバーメッセージブロック (SMB) ソリューションを使用することをお勧めします。詳細については、以下を参照してください。

- [Amazon FSx for Windows File Server Windows ユーザーガイド](#)
- [Using Amazon FSx with Amazon AppStream 2.0](#)
- AWS Storage Gateway ユーザーガイドの[ファイルゲートウェイ](#)

フォルダに何千ものファイルが含まれている場合、AppStream 2.0 でファイルの一覧を表示するときに時間がかかることがあります。

AppStream 2.0 は、API 呼び出しを使用して AppStream 2.0 永続ストレージに保存されているフォルダの内容を取得します。呼び出しを実行するたびに API 呼び出しで取得できる項目数には制限があります。このため、AppStream 2.0 は 1 つのフォルダ内の数千のファイルを取得する必要がある場合、ファイル数が少ないフォルダにあるファイルのリストを表示するよりも、すべてのファイルのリストを表示する方が時間がかかる場合があります。

この問題を解決するには、1 つのフォルダに数千のファイルがある場合、このコンテンツをより少ないファイルのグループに分割し、各グループを別のフォルダに保存することをお勧めします。これにより、各フォルダ内のファイルの一覧を表示するのに必要な API 呼び出しの数が減ります。

通知コードのトラブルシューティング

Amazon AppStream 2.0 の設定および使用に伴って表示される可能性がある通知の通知コードと解決手順を以下に示します。これらの通知は、Image Builder またはフリートを選択後、AppStream

2.0 コンソールの [Notifications (通知)] タブで確認できます。AppStream 2.0 API オペレーション [DescribeFleets](#) または [describe-fleets](#) CLI コマンドを使用して、フリート通知を取得することもできます。

Active Directory 内部サービス

Amazon AppStream 2.0 で Active Directory を設定および使用する際に内部サービスエラーが表示された場合は、次の手順に従います。

INTERNAL_SERVICE_ERROR

メッセージ: ユーザー名またはパスワードに誤りがあります。

解決策: このエラーは、リソースの Microsoft Active Directory ドメインで作成されたコンピュータオブジェクトを削除または無効化したときに発生する場合があります。このエラーを解決するには、Active Directory ドメインでコンピュータオブジェクトを有効にしてから、リソースを再度起動します。また、必要に応じて Active Directory ドメインでコンピュータオブジェクトアカウントをリセットします。このエラーが引き続き発生する場合は、[お問い合わせ](#) してください AWS サポート。詳細については、[AWS サポート センター](#) を参照してください。

Active Directory ドメイン参加

Amazon AppStream 2.0 で Active Directory を設定および使用する際に発生する可能性があるドメイン参加の問題の通知コードと解決手順を以下に示します。

DOMAIN_JOIN_ERROR_ACCESS_DENIED

メッセージ: アクセスが拒否されました。

解決策: ディレクトリ設定で指定されたサービスアカウントに、コンピュータオブジェクトを作成するアクセス許可、または既存のものを再利用するアクセス許可がありません。アクセス権限を検証し、Image Builder またはフリートを起動します。詳細については、「[アクティブディレクトリコンピュータオブジェクトを作成および管理するための許可の付与](#)」を参照してください。

DOMAIN_JOIN_ERROR_LOGON_FAILURE

メッセージ: ユーザー名またはパスワードに誤りがあります。

解決策: ディレクトリ設定で指定されたサービスアカウントのユーザー名またはパスワードが無効です。設定を更新して、エラーが発生した Image Builder またはフリートを再作成します。

DOMAIN_JOIN_NERR_PASSWORD_EXPIRED

メッセージ: このユーザーのパスワードの有効期限が切れています。

Resolution (解決策): AppStream 2.0 ディレクトリ設定で指定されたサービスアカウントのパスワードが期限切れです。Active Directory ドメインでサービスアカウントのパスワードを変更し、設定を更新して、エラーが発生した Image Builder またはフリートを再作成します。

DOMAIN_JOIN_ERROR_DS_MACHINE_ACCOUNT_QUOTA_EXCEEDED

メッセージ: コンピュータをドメインに結合できませんでした。このドメインで作成が許可されているコンピュータアカウントの最大数を超過しています。システム管理者に問い合わせ、この制限をリセットまたは引き上げます。

解決策: ディレクトリ設定で指定したサービスアカウントに、コンピュータオブジェクトの作成または既存のコンピュータオブジェクトの使用に必要なアクセス許可がありません。アクセス権限を検証し、Image Builder またはフリートを起動します。詳細については、「[アクティブディレクトリコンピュータオブジェクトを作成および管理するための許可の付与](#)」を参照してください。

DOMAIN_JOIN_ERROR_INVALID_PARAMETER

メッセージ: パラメータが正しくありません。このエラーは、LpName パラメータが NULL であるか、NameType パラメータが NetSetupUnknown または不明な名前タイプとして指定されている場合に返されます。

解決策: このエラーは、OU の識別名が正しくない場合に発生します。OU を検証して、もう一度試してください。このエラーが引き続き発生する場合は、[お問い合わせ](#)ください AWS サポート。詳細については、[AWS サポート センター](#)を参照してください。

DOMAIN_JOIN_ERROR_MORE_DATA

メッセージ: その他のデータを使用できます。

解決策: このエラーは、OU の識別名が正しくない場合に発生します。OU を検証して、もう一度試してください。このエラーが引き続き発生する場合は、[お問い合わせ](#)ください AWS サポート。詳細については、[AWS サポート センター](#)を参照してください。

DOMAIN_JOIN_ERROR_NO_SUCH_DOMAIN

メッセージ: 指定されたドメイン名が存在しないか、接続できませんでした。

解決策: ストリーミングインスタンスが Active Directory ドメインに接続できませんでした。ネットワーク接続を確保するには、VPC、サブネット、およびセキュリティグループ設定を確認しま

す。詳細については、「[AppStream 2.0 ストリーミングインスタンスが Active Directory ドメインに参加していない。](#)」を参照してください。

DOMAIN_JOIN_NERR_WORKSTATION_NOT_STARTED

メッセージ: Workstation サービスが開始されていません。

解決策: Workstation サービスの開始時にエラーが発生しました。イメージでサービスが有効になっていることを確認します。このエラーが引き続き発生する場合は、[お問い合わせ](#)してください。AWS サポート。詳細については、[AWS サポート センター](#)を参照してください。

DOMAIN_JOIN_ERROR_NOT_SUPPORTED

メッセージ: リクエストはサポートされていません。このエラーは、リモートコンピュータが lpServer パラメータで指定されており、この呼び出しがリモートコンピュータでサポートされていない場合に返されます。

解決策: サポート AWS サポート が必要な場合は、[お問い合わせ](#)してください。詳細については、[AWS サポート センター](#)を参照してください。

DOMAIN_JOIN_ERROR_FILE_NOT_FOUND

メッセージ: 指定されたファイルがシステムで見つかりません。

解決策: このエラーは、無効な組織単位 (OU) の識別子名が指定されている場合に発生します。識別子名の先頭には、**OU=** を付ける必要があります。OU 識別子名を検証し、再試行してください。詳細については、「[組織単位の識別子名を検索する](#)」を参照してください。

DOMAIN_JOIN_INTERNAL_SERVICE_ERROR

メッセージ: アカウントは既に存在しています。

Resolution (解決策): このエラーは、次の状況で発生する可能性があります。

- 問題がアクセス許可に関連していない場合は、Netdom ログでエラーがないか確認し、正しい OU を指定したことを確認してください。
- ディレクトリ設定で指定されたサービスアカウントに、コンピュータオブジェクトを作成するアクセス権限、または既存のものを再利用するアクセス権限がありません。この場合、アクセス権限を検証し、Image Builder またはフリートを起動します。詳細については、「[アクティブ ディレクトリコンピュータオブジェクトを作成および管理するための許可の付与](#)」を参照してください。
- AppStream 2.0 で作成したコンピュータオブジェクトは、作成後に作成先の OU から移動されます。この場合、最初の Image Builder またはフリートは正常に作成されますが、そのコン

コンピュータオブジェクトを使用する新しい Image Builder またはフリートは失敗します。Active Directory が指定先の OU でコンピュータオブジェクトを検索し、ドメイン内の別の場所で同じ名前のオブジェクトを検出すると、ドメイン参加は失敗します。

- AppStream 2.0 Directory Config で指定されている OU の名前には、ディレクトリ設定のカンマの前または後にスペースが含まれています。この場合、フリートまたは Image Builder が Active Directory ドメインに再参加しようとする、AppStream 2.0 はコンピュータオブジェクトを正しく循環できず、ドメインの再参加は成功しません。フリートでこの問題を解決するには、次の手順を実行します。
 1. フリートを停止します。
 2. フリートの Active Directory ドメイン設定を編集して、フリートが参加している Directory Config およびディレクトリ OU を削除します。詳細については、[「ステップ 3: ドメイン結合フリートを作成する」](#)を参照してください。
 3. AppStream 2.0 Directory Config を更新して、スペースを含まない OU を指定します。詳細については、[「ステップ 1: Directory Config オブジェクトを作成する」](#)を参照してください。
 4. フリートの Active Directory ドメイン設定を編集して、更新されたディレクトリ OU で Directory Config を指定します。

Image Builder でこの問題を解決するには、次の手順を実行します。

1. Image Builder を削除します。
2. AppStream 2.0 Directory Config を更新して、スペースを含まない OU を指定します。詳細については、[「ステップ 1: Directory Config オブジェクトを作成する」](#)を参照してください。
3. 新しい Image Builder を作成し、更新されたディレクトリ OU を使用して Directory Config を指定します。詳細については、[「Image Builder を起動し、ストリーミングアプリケーションをインストールして設定する」](#)を参照してください。

イメージの内部サービス

AppStream 2.0 でのマネージド型イメージ更新を使用してイメージの更新を開始した後に内部サービスエラーが表示される場合は、次の手順に従ってください。

INTERNAL_SERVICE_ERROR

メッセージ: AppStream 2.0 could not update image *image-name*. (AppStream 2.0 がイメージの image-name を更新できませんでした。) Failed to update/install/configure/disable <software name>. (update/install/configure/disable <ソフトウェア名> に失敗しました。) Check your source image and try again. (ソースイメージを確認して、もう一度試してください。) この問題が解決しない場合は、[お問い合わせ](#)してください AWS サポート。

解決方法: このエラーは、ソースイメージに問題がある場合に発生することがあります。イメージをもう一度更新してみてください。

それでも更新されない場合は、SSM Agent の最新バージョンを使用していることを確認してください。バージョンについては、「[the section called “ベースイメージおよびマネージド型イメージの更新版リリースノート”](#)」を参照してください。インストールの詳細については、「[Windows Server の EC2 インスタンスに SSM Agent を手動でインストールする](#)」を参照してください。

エラーが引き続き発生する場合は、イメージから Image Builder を起動してください。詳細については、「[Image Builder を起動し、ストリーミングアプリケーションをインストールして設定する](#)」を参照してください。イメージから Image Builder を起動できない場合は、イメージに関する別の問題があるということになります。その問題は、AppStream 2.0 でのマネージド型イメージ更新を使用してイメージを更新する前に解決する必要があります。このエラーが引き続き発生する場合は、[お問い合わせ](#)してください AWS サポート。詳細については、[AWS サポート センター](#)を参照してください。

セッションプロビジョニング

以下は、エンドユーザーがストリーミングセッションのプロビジョニングを試みたときに発生する可能性のあるセッションプロビジョニングに関する問題の通知コードと解決手順です。

Note

以下の「X」は、特定のエラーコードが発生したセッションの数に等しくなります。

USER_PROFILE_MOUNTING_FAILURE

メッセージ: X セッション (複数可) でユーザープロファイルのマウントに失敗しました。

解決策: この問題をトラブルシューティングするには、ユーザープロファイルが破損しているか、またはインスタンス上のサードパーティープロセスがユーザープロファイルのマウントを妨

げていないかを確認します。このエラーが引き続き発生する場合は、[お問い合わせ](#)してください
AWS サポート。詳細については、[AWS サポート センター](#)を参照してください。

USER_PROFILE_DOWNLOADING_FAILURE

メッセージ: X セッションでユーザープロファイルのダウンロードに失敗しました。

解決策: この問題をトラブルシューティングするには、ネットワーク設定を確認してください。
このエラーが引き続き発生する場合は、[お問い合わせ](#)してください AWS サポート。詳細については、[AWS サポート センター](#)を参照してください。

HOME_FOLDER_MOUNTING_FAILURE

メッセージ: X セッションでホームフォルダのマウントに失敗しました。

解決策: この問題をトラブルシューティングするには、ネットワーク設定を確認してください。
このエラーが引き続き発生する場合は、[お問い合わせ](#)してください AWS サポート。詳細については、「[AWS サポート センター](#)」を参照してください。

Amazon AppStream 2.0 Service Quotas

AppStream 2.0 にはさまざまなリソースが用意されており、それらを利用することができます。AppStream 2.0 リソースには、スタック、フリート、イメージ、および Image Builder が含まれます。Amazon Web Services アカウントを作成すると、作成できるリソースの数、および AppStream 2.0 サービスを使用できるユーザー数が、デフォルトのクォータ (制限とも言う) として設定されます。

クォータの引き上げをリクエストするには、<https://console.aws.amazon.com/servicequotas/> の [Service Quotas] コンソールを使用します。詳細については「Service Quotas ユーザーガイド」の「[クォータの引き上げのリクエスト](#)」を参照してください。

次の表は、各 AppStream 2.0 リソースと AppStream 2.0 ユーザープール内のユーザーのデフォルトクォータ一覧です。アカウントの実際のクォータは、アカウントを作成した時期に応じて、より大きくても低くてもかまいません。

Note

Graphics Pro インスタンスは、Graphics Pro インスタンスタイプをサポートするハードウェアのサポート終了により、10/31/2025 AWS 以降は使用できなくなります。AppStream 2.0 は、Graphics Pro インスタンスの制限引き上げリクエストを受け入れません。

Graphics Design インスタンスは、Graphics Design インスタンスタイプをサポートするハードウェアのサポートが終了するため、12/31/2025 AWS 以降は使用できなくなります。AppStream 2.0 は、Graphics Design インスタンスの制限引き上げリクエストを受け入れません。

名前	デフォルト	引き上げ可能
スタック	10	可能
群数	10	可能
コンピュート最適化フリートインスタンス *	- stream.compute.large: 5 - stream.compute.xlarge: 2 - stream.compute.2xlarge: 0 - stream.compute.4xlarge: 0	はい

名前	デフォルト	引き上げ可能
	stream.compute.8xlarge: 0	
グラフィックスフリートインスタンス *	- stream.graphics-design.large: 3 - stream.graphics-design.xlarge: 3 - stream.graphics-design.2xlarge: 3 - stream.graphics-design.4xlarge: 0 - stream.graphics-desktop.2xlarge: 0 - stream.graphics-pro.4xlarge: 0 - stream.graphics-pro.8xlarge: 0 - stream.graphics-pro.16xlarge: 0 - stream.graphics.g4dn.xlarge: 0 - stream.graphics.g4dn.2xlarge: 0 - stream.graphics.g4dn.4xlarge: 0 - stream.graphics.g4dn.8xlarge: 0 - stream.graphics.g4dn.12xlarge: 0 - stream.graphics.g4dn.16xlarge: 0 - stream.graphics.g5.xlarge: 0 - stream.graphics.g5.2xlarge: 0 - stream.graphics.g5.4xlarge: 0 - stream.graphics.g5.8xlarge: 0 - stream.graphics.g5.12xlarge: 0 - stream.graphics.g5.16xlarge: 0 - stream.graphics.g5.24xlarge: 0	はい

名前	デフォルト	引き上げ可能
メモリ最適化フリートインスタンス *	• stream.memory.large: 5 • stream.memory.xlarge: 2 • stream.memory.2xlarge: 0 • stream.memory.4xlarge: 0 • stream.memory.8xlarge: 0 • stream.memory.z1d.large: 5 • stream.memory.z1d.xlarge: 2 • stream.memory.z1d.2xlarge: 0 • stream.memory.z1d.3xlarge: 0 • stream.memory.z1d.6xlarge: 0 • stream.memory.z1d.12xlarge: 0	はい
スタンダードフリートインスタンス *	• stream.standard.small: 50 • stream.standard.medium: 50 • stream.standard.large: 50 • stream.standard.xlarge: 10 • stream.standard.2xlarge: 10	はい
イメージビルダー (合計)	10	可能
イメージ	10	可能
コンピューティング最適化 Image Builder インスタンス	• stream.compute.large: 3 • stream.compute.xlarge: 3 • stream.compute.2xlarge: 0 • stream.compute.4xlarge: 0 • stream.compute.8xlarge: 0	はい

名前	デフォルト	引き上げ可能
グラフィックス Image Builder インスタンス	• stream.graphics-design.large: 1 • stream.graphics-design.xlarge: 1 • stream.graphics-design.2xlarge: 1 • stream.graphics-design.4xlarge: 0 • stream.graphics-desktop.2xlarge: 0 • stream.graphics-pro.4xlarge: 0 • stream.graphics-pro.8xlarge: 0 • stream.graphics-pro.16xlarge: 0 • stream.graphics.g4dn.xlarge: 0 • stream.graphics.g4dn.2xlarge: 0 • stream.graphics.g4dn.4xlarge: 0 • stream.graphics.g4dn.8xlarge: 0 • stream.graphics.g4dn.12xlarge: 0 • stream.graphics.g4dn.16xlarge: 0	はい
メモリ最適化 Image Builder インスタンス	• stream.memory.large: 3 • stream.memory.xlarge: 3 • stream.memory.2xlarge: 0 • stream.memory.4xlarge: 0 • stream.memory.8xlarge: 0 • stream.memory.z1d.large: 3 • stream.memory.z1d.xlarge: 3 • stream.memory.z1d.2xlarge: 0 • stream.memory.z1d.3xlarge: 0 • stream.memory.z1d.6xlarge: 0 • stream.memory.z1d.12xlarge: 0	はい

名前	デフォルト	引き上げ可能
スタンダード Image Builder インスタンス	- stream.standard.small: 5 - stream.standard.medium: 5 - stream.standard.large: 5 - stream.standard.xlarge: 3 - stream.standard.2xlarge: 3	はい
共有できる AWS アカウント イメージの数	100	可能
送信先リージョンあたりの同時イメージコピー	2	可能
イメージの同時更新数	5	可能
ユーザープールのユーザー	50	可能
Elastic フリートの最大同時セッション数	Amazon Linux 2 - stream.standard.small: 10 - stream.standard.medium: 10 - stream.standard.large: 5 - stream.standard.xlarge: 2 - stream.standard.2xlarge: 2 [Windows Server 2019] - stream.standard.small: 10 - stream.standard.medium: 10 - stream.standard.large: 5 - stream.standard.xlarge: 2 - stream.standard.2xlarge: 2	はい
App Block Builder (合計)	10	[Yes (はい)]

名前	デフォルト	引き上げ可能
App Block Builder の最大数	- stream.standard.small: 1 - stream.standard.medium: 1 - stream.standard.large: 1 - stream.standard.xlarge: 1 - stream.standard.2xlarge: 1	はい

* AppStream 2.0 インスタンスタイプとサイズクォータは AWS 、リージョンごとに AWS アカウントごとに異なります。同じリージョン内に同じインスタンスタイプとサイズを使用するフリートが複数存在する場合、そのリージョン内のすべてのフリートインスタンスの総数は、該当するクォータ以下でなければなりません。リージョンまたはアベイラビリティゾーンで利用できるインスタンスタイプを確認するには、[AppStream 2.0](#) 料金表の AWS 「リージョン別の料金表 – 常時オン、オンデマンド、App Block Builder、および Image Builder インスタンス」を参照してください。

[Default Internet Access (デフォルトのインターネットアクセス)] が有効になっているフリートの場合、クォータは 100 フリートインスタンスです。デプロイで 100 を超える同時ユーザーをサポートする必要がある場合は、代わりに[NAT ゲートウェイ設定](#)を使用します。フリートに対してインターネットアクセスを有効にする方法の詳細については、「[インターネットアクセス](#)」を参照してください。

AppStream 2.0 ユーザー向けのガイダンス

AppStream 2.0 管理者である場合、AppStream 2.0 の使用を開始するのに役立つこのセクションのガイダンスをユーザーに提供できます。

AppStream 2.0 にアクセス可能になったユーザーである場合、このセクションのトピックはアプリケーションストリーミングに AppStream 2.0 を使用するのに役立ちます。AppStream 2.0 では、管理者がアプリケーションをリモートからアクセス可能にするため、ユーザーがアプリケーションを自分のデバイスにインストールする必要がありません。アプリケーションにアクセスするには、AppStream 2.0 に接続し、アプリケーションストリーミングセッションを開始します。

目次

- [AppStream 2.0 アクセス方法とクライアント](#)
- [ファイルストレージのオプション](#)
- [地域の設定を指定する](#)

AppStream 2.0 アクセス方法とクライアント

ウェブブラウザまたは Windows 用の AppStream 2.0 クライアントを使用して AppStream 2.0 に接続できます。

目次

- [ウェブブラウザアクセス](#)
- [Windows 用 AppStream 2.0 クライアントアプリケーション](#)
- [macOS 用 AppStream 2.0 クライアントアプリケーション](#)

ウェブブラウザアクセス

次の情報は、ウェブブラウザを使用して AppStream 2.0 に接続し、アプリケーションをストリーミングする際に役立ちます。

内容

- [要件](#)
- [セットアップ](#)

- [AppStream 2.0 に接続する](#)
- [AppStream 2.0 Web ブラウザアクセス \(バージョン 2\)](#)
- [モニターとディスプレイの解像度](#)
- [USB デバイス](#)
- [タッチスクリーンデバイス](#)
- [ファンクションキー](#)
- [Mac のオプションキーおよびコマンドキーを再マップする](#)
- [ビデオ会議と音声会議](#)
- [ドローイングタブレット](#)
- [相対マウスオフセット](#)
- [トラブルシューティング](#)

要件

HTML5 対応のウェブブラウザを使用して、任意の場所から AppStream 2.0 に接続できます。サポートされているブラウザは以下のとおりです。

- Google Chrome
- Mozilla Firefox
- Safari
- Microsoft Edge

AppStream 2.0 は、サポートされているすべてのブラウザの最新の 3 つの主要バージョンをサポートしています。古いブラウザバージョンでウェブクライアントにアクセスするユーザーは、最適なパフォーマンスを確保するために更新を推奨する通知を受け取ります。

Note

AppStream 2.0 ストリーミングセッション中にドローイングタブレットで利用できるのは、Google Chrome または Mozilla Firefox ブラウザのみです。ビデオおよび音声会議用のウェブカメラリダイレクトは、Google Chrome や Microsoft Edge などの Chromium ベースのウェブブラウザでサポートされています。

セットアップ

ウェブブラウザで AppStream 2.0 を使用する際に、ブラウザの拡張機能やプラグインは必要ありません。

AppStream 2.0 に接続する

AppStream 2.0 に接続してアプリケーションのストリーミングセッションを開始するには、次の手順に従います。

1. 管理者が組織のサインインページで最初にサインインする必要がある場合は、この手順のタスクを実行します。

管理者が組織のサインインページからサインインする必要がある場合は、この手順のタスクをスキップして手順 2 に進みます。

- a. 組織のサインインページに移動し、プロンプトが表示されたら、ドメインの認証情報を入力します。
 - b. サインインすると、AppStream 2.0 ストリーミングセッションで利用できる 1 つまたは複数のアプリケーションを表示するページにリダイレクトされます。管理者によって有効になっている場合は、[デスクトップビュー] も使用できます。
 - c. アプリケーションを選択するか、使用可能な場合は [デスクトップビュー] を選択します。
2. 管理者が組織のサインインページで最初にサインインする必要がある場合は、次のいずれかの操作を行います。

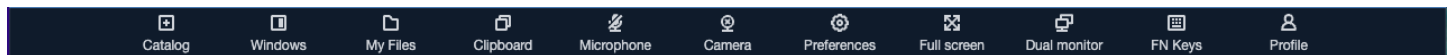
- AppStream 2.0 を初めて使用し、AppStream 2.0 を使用したアプリケーションへのアクセスを開始するように通知するウェルカムメールを受信した場合:

1. E メールを開き、[ログインページ] リンクを選択します。
2. 自分の E メールアドレスと E メールで提供された一時パスワードを入力し、[ログイン] を選択します。
3. プロンプトが表示されたら、新しいパスワードを入力して確認し、[Set Password] を選択します。
4. しばらくすると、AppStream 2.0 ポータルが開き、AppStream 2.0 ストリーミングセッションで利用できる 1 つまたは複数のアプリケーションが表示されます。管理者によって有効になっている場合は、[デスクトップビュー] も使用できます。
5. アプリケーションを選択するか、使用可能な場合は [Desktop View (デスクトップビュー)] を選択します。

- AppStream 2.0 を初めて使用したわけではなく、管理者が AppStream 2.0 ポータルのウェブアドレス (URL) が提供している場合は、次の手順を実行します。
 1. 管理者から提供された URL を入力して、AppStream 2.0 ポータルに移動します。
 2. プロンプトが表示されたら、パスワードを入力して [接続] を選択します。
 3. しばらくすると、AppStream 2.0 ポータルが開き、AppStream 2.0 ストリーミングセッションで使用する 1 つまたは複数のアプリケーションが表示されます。管理者によって有効になっている場合は、[デスクトップビュー] も使用できます。

AppStream 2.0 Web ブラウザアクセス (バージョン 2)

AppStream 2.0 ウェブブラウザアクセスバージョン 2 では、簡単に見つけられるメニューオプションやテキストによるエンドユーザー向けガイダンスなど、エンドユーザーエクスペリエンスが強化されています。新しいメニュー項目は追加されておらず、以前のバージョンで使用できたすべての設定オプションが新しいバージョンでも引き続き使用できます。これらの設定オプションは、AppStream 2.0 チームが実施した使いやすさのテストの結果、再編成されました。



エンドユーザーは、強化された AppStream 2.0 ツールバーに加えて、次の機能を [マイファイル] で使用できます。

- 複数のファイルをダウンロードまたは削除します。ファイルを選択し、[アクション]、[削除] または [ダウンロード] の順に選択します。
- ドラッグアンドドロップを使用してフォルダをアップロードします。
- [名前]、[最終更新日]、および [サイズ] に基づいてファイルを並べ替えます。
- ファイル名が長い場合は、ファイル名の列を折り返します。右上隅にある小さな歯車アイコンを選択し、[列の折り返し]、[確認] の順に選択します。

エンドユーザーは、以下の機能にアクセスすることもできます。

- 2 つのビジュアルモード (ライトとダーク) 間で切り替えるには、[設定]、[一般]、[テーマ]、[ライトモード] または [ダークモード] の順に選択します。
- ストリーミングセッションからフォーカスを離し、ツールバーの最初の要素にフォーカスを移してセッション中にキーボードベースでできるようにするには、キーボードショートカット **ctrl + alt + shift + F11** を使用します。

モニターとディスプレイの解像度

AppStream 2.0 では、異なる解像度のモニターを含め、ストリーミングセッション中に複数のモニターを使用できます。最適なストリーミング体験を実現するために、複数のモニターを使用する場合は、モニターの表示縮尺を 100% に設定することをお勧めします。

以下のウェブブラウザで開始されるアプリケーションストリーミングセッションにデュアルモニターを使用できます。

- Google Chrome
- Mozilla Firefox
- Safari
- Microsoft Edge

デュアルモニターでのブラウザベースのストリーミングセッションでは、モニターごとに最大ディスプレイ解像度 2,560 x 1,600 ピクセルがサポートされます。2 台以上のモニターを必要とする場合、またはモニターあたり 2,560 x 1,600 ピクセルを超えるディスプレイ解像度を必要とする場合は、AppStream 2.0 クライアントを使用する必要があります。

USB デバイス

USB デバイスは、ブラウザベースの AppStream 2.0 ストリーミングセッションではサポートされていません。AppStream 2.0 経由でストリーミングされるアプリケーションで USB デバイスを使用するには、AppStream 2.0 クライアントを使用する必要があります。詳細については、「[Windows 用 AppStream 2.0 クライアントアプリケーション](#)」を参照してください。

タッチスクリーンデバイス

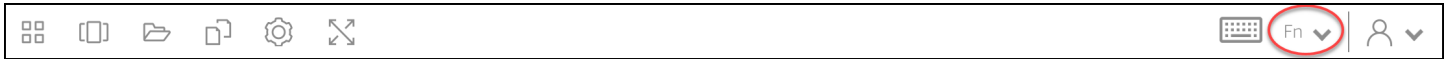
AppStream 2.0 は、タッチ対応の iPad、Android タブレット、Windows デバイスでのジェスチャをサポートしています。サポートされているタッチジェスチャは、ロングタップによる右クリック、スワイプによるスクロール、ピンチによるズーム、2 本指での回転などであり、アプリケーションで使用できます。

Note

画面サイズが 8 インチ未満のタッチスクリーンデバイスはサポートされていません。

iPad または Android タブレットにスクリーンキーボードを表示するには、AppStream 2.0 ツールバーのキーボードアイコンをタップします。キーボードアイコンが青色に変わると、スクリーンキーボードを使用してストリーミングアプリケーションにテキストを入力できます。キーボードアイコンをもう一度タップすると、スクリーンキーボードが非表示になります。

[Fn] アイコンをタップすると、Windows 固有のキーとキーボードショートカットの並びが表示されます。



タッチ対応デバイスの場合、AppStream 2.0 ツールバーのキーボードアイコンをタップしたときに表示されるリモートキーボードは、ローカルキーボードとは異なります。ローカルキーボードは、ローカルで実行されているアプリケーションの入力コントロール内をユーザーがタップしたときに、タッチ対応デバイスに自動的に表示されるスクリーンキーボードです。AppStream 2.0 ストリーミングセッション中は、リモートキーボードを使用してストリーミングアプリケーションにのみテキストを入力できます。ユーザーは AppStream 2.0 ツールバーのキーボードアイコンをタップすることによってのみ、リモートキーボードを表示または非表示にすることができます。AppStream 2.0 ツールバーの青色のキーボードアイコンは、リモートキーボードがアクティブであることを示します。

ローカルキーボードを使用して、[My Files (マイファイル)] ダイアログボックスなど、AppStream 2.0 ウェブポータル要素にテキストを入力できます。ただし、このキーボードを使用してテキストをストリーミングアプリケーションに入力することはできません。また、AppStream 2.0 ツールバーのキーボードアイコンを使用して、このキーボードを表示または非表示にすることはできません。

Note

Windows コンピュータにスクリーンキーボードを表示するには、Windows システムトレイのキーボードアイコンをタップします。Windows システムトレイにキーボードアイコンが表示されていない場合は、Windows タブレットモードに切り替えます。Windows システムトレイのキーボードアイコンをもう一度タップすると、スクリーンキーボードが表示されます。

ファンクションキーの終了についての詳しい情報は、次のセクションを参照してください。

ファンクションキー

AppStream 2.0 ストリーミングセッション中にキーボードショートカットを使用して、特殊なキーストロークまたはキーの組み合わせを入力できます。ストリーミングセッション中に Windows 固有の

キーとキーボードショートカットの行を表示するには、[Fn] アイコン (または AppStream 2.0 ウェブブラウザアクセス v2 の FN キー) を選択します。[Fn] アイコンは、セッションウィンドウの右上にある AppStream 2.0 ツールバーに表示されます。



以下に示しているのは、[Fn] アイコン (または AppStream 2.0 ウェブブラウザアクセス v2 の FN キー) を選択したときに Windows 固有のキーとキーボードショートカットがどのように表示されるかの例です。すべてのキーが表示されない場合は、ショートカットツールバーを右または左にスクロールして、さらにキーを表示できます。



Windows の Ctrl キーを含むキーの組み合わせを使用するには、ショートカットツールバーの Ctrl キーを選択して、ショートカットツールバー (タッチ対応デバイスを使用している場合は、スクリーンキーボード) で任意のキーを入力します。Ctrl キーを選択すると色が青に変わります。この場合、選択した別のキーはすべて、Control キーを含むキーの組み合わせとして解釈されます。

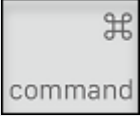


もう一度 Ctrl キーを選択すると、このキーの選択が解除されます。たとえば、キーボードショートカット Ctrl + F を使用するには、ショートカットツールバーの Ctrl キーを選択してから、f キーを押します。ショートカットツールバーの Ctrl キーをもう一度選択すると、Ctrl キーの選択が解除されます。Alt キーまたは Shift キーを含むショートカットを使用するには、同じ方法でショートカットツールバーの Alt キーまたは Shift キーを選択します。ショートカットツールバーの Shift キーは、キーボードショートカットにのみ使用できます。タッチ対応デバイスを使用している場合、このキーは、スクリーンキーボードで入力したキーの大文字と小文字には影響しません。

Mac のオプションキーおよびコマンドキーを再マップする

macOS または Mac OS X を実行するデバイスを使用して AppStream 2.0 に接続する場合、キーボードの Mac オプションキーとコマンドキーを再マップできます。

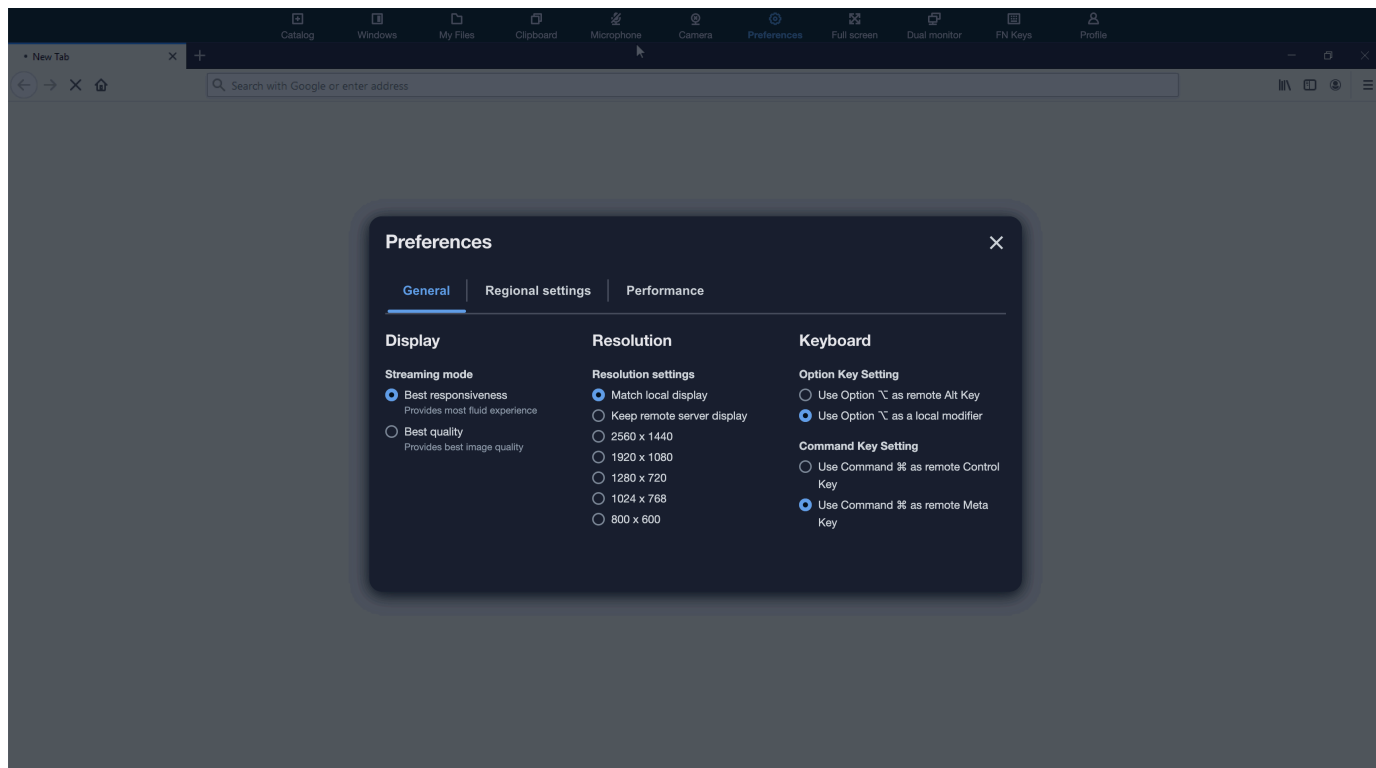
両方のキーと一緒に使用すると、修飾キーによって別のキーの操作が変更されます。修飾キーを別のキーと一緒に使用して、プリントなどのタスクを実行できます。[Meta key (メタキー)] は、修飾キーの特殊なタイプです。両方のキーと一緒に使用すると、メタキーを使用して別のキーの機能を一時的に変更できます。

この Mac キーを再マップできます	ストリーミングセッション中にこのキーへ
オプションキー 	• リモート Alt キー • ローカル修飾キー
コマンドキー 	• リモートコントロールキー • リモートメタキー

AppStream 2.0 ストリーミングセッション中に Mac オプションキーとコマンドキーを再マップするには、次の手順に従います。

Mac オプションキーとコマンドキーを再マップするには

1. ウェブブラウザを使用して AppStream 2.0 に接続します。
2. AppStream 2.0 ツールバーの左上にある [設定] アイコンを選択し、[キーボード設定] を選択します。
3. 再マップするキーに対応するオプションを選択します。



AppStream 2.0 ウェブブラウザアクセス v2 で Mac オプションキーとコマンドキーを再マップするには、次の手順に従います。

AppStream 2.0 ウェブブラウザアクセス v2 で Mac オプションキーとコマンドキーを再マップするには

1. ウェブブラウザを使用して AppStream 2.0 に接続します。
2. AppStream 2.0 ツールバーのトップメニューから、[設定] メニューを選択します。
3. [全般]、[キーボード]、および再マップするキーに対応するオプションを選択します。

ビデオ会議と音声会議

AppStream 2.0 リアルタイムオーディオビデオ (AV) は、ローカルなウェブカメラビデオおよびマイク音声からの入力を、AppStream 2.0 のストリーミングセッションにリダイレクトします。これにより、AppStream 2.0 ストリーミングセッション内で、ローカルデバイスを使用したビデオ会議や音声会議を行うことができます。

AppStream 2.0 ストリーミングセッション内でローカルのウェブカメラやマイクを使用するには

1. Google Chrome や Microsoft Edge など、Chrome ベースのウェブブラウザから AppStream 2.0 に接続します。

 Note

Chrome、Edge、Firefox などの一般に使用されるほとんどの HTML5 互換ブラウザは、AppStream 2.0 セッションでのオーディオ入力をサポートしています。

 Note

ウェブブラウザがビデオまたはオーディオからの入力をサポートしていない場合は、そのオプションが AppStream 2.0 ツールバーに表示されません。

2. デフォルトのデバイスとして、カメラとマイクへのウェブブラウザによるアクセス許可を設定した上で、AppStream 2.0 へのアクセスを許可します。

 Note

Google Chrome の設定方法の詳細については、「[カメラとマイクを使用する](#)」を参照してください。

3. AppStream 2.0 ツールバーの左上で [設定] アイコンを選択してから、[ウェブカメラを有効化] を選択します。AppStream 2.0 ウェブブラウザアクセス v2 の場合は、AppStream 2.0 ツールバーのオプションから [カメラ] を選択します (オプションが青色に変わります)。

 Note

[設定] メニューに、マイクまたはウェブカメラのアイコンが表示されない場合は、AppStream 2.0 の管理者にお問い合わせください。この原因としては、ウェブブラウザがビデオまたはオーディオ入力をサポートしていない場合や、管理者による追加の設定タスクが必要な場合が考えられます。詳細については、「[the section called “リアルタイムオーディオビデオ”](#)」を参照してください。

4. ウェブブラウザの設定によっては、カメラの使用を許可するよう促すメッセージが、そのブラウザに表示される場合があります。この場合は、[許可] を選択して、カメラを有効にします。

5. AppStream 2.0 ツールバーの左上で [設定] アイコンを選択してから、[マイクを有効化] を選択します。AppStream 2.0 ウェブブラウザアクセス v2 の場合は、AppStream 2.0 ツールバーのオプションから [マイク] を選択します (オプションが青色に変わります)。
6. ウェブブラウザの設定によっては、マイクロホンの使用を促すメッセージが、そのブラウザに表示される場合があります。[許可] を選択して、マイクを有効にします。

Note

複数のウェブカメラまたはマイクがあり、AppStream 2.0 セッション内でストリーミングに使用するデバイスを変更したい場合は、ウェブブラウザの AppStream 2.0 ウェブサイト URL に関する設定を消去し、デフォルトのデバイスを設定し直す必要があります。その後、変更を適用するためにブラウザを更新するか新しいセッションを開始し、前出のステップを繰り返してウェブカメラとマイクを有効にします。

ドローイングタブレット

ドローイングタブレット (ペンタブレットとも呼ばれます) は、スタイラス (ペン) を使って絵を描くことができるコンピュータ入力デバイスです。AppStream 2.0 では、Wacom ドローイングタブレットなどのドローイングタブレットをローカルコンピュータに接続し、そのタブレットをストリーミングアプリケーションで使用できます。

ストリーミングアプリケーションでドローイングタブレットを使用するための要件と考慮事項を次に示します。

- この機能を使用するには、Google Chrome または Mozilla Firefox ブラウザのみ、または [AppStream 2.0 クライアント](#) を使用して AppStream 2.0 に接続する必要があります。
- ストリーミングするアプリケーションは、Windows Ink テクノロジーをサポートしている必要があります。詳細については、「[Windows アプリでのペン操作と Windows インク](#)」を参照してください。
- 使用するストリーミングアプリケーションによっては、ドローイングタブレットが想定どおりに機能するために、USB リダイレクトが必要になる場合があります。これは、GIMP などの一部のアプリケーションでは、圧力感度をサポートするために USB リダイレクトが必要なためです。ストリーミングアプリケーションでこれが該当する場合は、AppStream 2.0 クライアントを使用して AppStream 2.0 に接続し、ドローイングタブレットをストリーミングセッションと共有する必要があります。USB デバイスをストリーミングセッションと共有する方法については、「[USB デバイス](#)」をご参照ください。

- この機能は Chromebook ではサポートされていません。

アプリケーションストリーミングセッション中にドローイングタブレットの使用を開始するには、USB を使用してドローイングタブレットをローカルコンピュータに接続し、圧力感度の検出に必要な場合は AppStream 2.0 とデバイスを共有してから、AppStream 2.0 ストリーミングセッションを開始します。サポートされているウェブブラウザまたは AppStream 2.0 クライアント (インストールされている場合) を使用して、ストリーミングセッションを開始できます。

相対マウスオフセット

デフォルトでは、ストリーミングセッション中、AppStream 2.0 は絶対座標を使用してマウスの動きをローカルにレンダリングすることにより、マウスの動きに関する情報を送信します。コンピュータ支援設計 (CAD)/コンピュータ支援製造 (CAM) ソフトウェアやビデオゲームなど、グラフィックを多用するアプリケーションの場合、相対的マウスモードが有効になっているときにマウスのパフォーマンスが向上します。相対的マウスモードでは、ウィンドウまたは画面内の XY 座標の絶対値ではなく、最終フレームからマウスが移動した距離を表す相対座標が使用されます。相対的マウスモードを有効にすると、AppStream 2.0 はマウスの動きをリモートでレンダリングします。

AppStream 2.0 ストリーミングセッション中にこの機能を有効にするには、次のいずれかの操作を行います。

- Windows: Ctrl + Shift + F8 を押す
- Mac: Ctrl + Fn + Shift + F8 を押す

トラブルシューティング

AppStream 2.0 の使用時に問題が発生した場合は、AppStream 2.0 セッション ID が管理者によるトラブルシューティングに役立ちます。このセクションでは、セッション ID を見つける方法について説明します。

セッション ID は、ストリーミングセッションをリクエストすると作成されます。セッション ID、および AppStream 2.0 によって使用されるその他の情報は、ブラウザのセッションストレージの場所に保存されます。ブラウザのインターフェイスで利用可能な開発者ツールを使用して、この場所を見つけることができます。

一般的なウェブブラウザで利用できる開発者ツールについては、以下のリソースを参照してください。

- [Apple Safari Developer Help: Storage tab](#)

- [View And Edit Session Storage With Chrome DevTools](#)
- [Firefox Developer Tools: Local Storage / Session Storage](#)
- [Microsoft Edge \(Chromium\) Developer Tools](#)
- [Microsoft Edge \(EdgeHTML\) Developer Tools](#)

ブラウザに対応したデベロッパーツールが見つかったら、AppStream 2.0 ウェブサイトのセッションストレージを検索します。ウェブサイトのドメインは `https://appstream2.<aws-region>.aws.amazon.com` です。ドメインを展開し、`[sessionStorage.as2SessionData]` を選択します。セッション ID はキー `[sessionId]` に保存されています。

Windows 用 AppStream 2.0 クライアントアプリケーション

次の情報は、Windows の AppStream 2.0 クライアントを使用して AppStream 2.0 に接続し、アプリケーションをストリーミングする際に役立ちます。

内容

- [機能](#)
- [要件](#)
- [Windows のセットアップ](#)
- [Windows クライアントで AppStream 2.0 に接続する](#)
- [監視](#)
- [USB デバイス](#)
- [ローカルファイルアクセス](#)
- [プリンターリダイレクト](#)
- [ビデオ会議と音声会議](#)
- [ドローイングタブレット](#)
- [相対マウスオフセット](#)
- [ログ記録](#)
- [トラブルシューティング](#)
- [AppStream 2.0 クライアントリリースノート](#)

機能

Windows 用 AppStream 2.0 クライアントは、Windows PC にインストールするアプリケーションです。このアプリケーションには、ウェブブラウザを使用して AppStream 2.0 にアクセスした場合には利用できない追加の機能があります。たとえば、AppStream 2.0 クライアントで以下のことを実行できます。

- 2 台以上のモニターまたは 4K 解像度を使用する。
- USB デバイスで AppStream 2.0 を介してストリーミングされるアプリケーションを使用する。
- ストリーミングセッション中にローカルドライブとフォルダにアクセスする。
- プリントジョブをストリーミングアプリケーションから、ローカルコンピュータに接続されているプリンターにリダイレクトします。
- ストリーミングセッション内のビデオ会議や音声会議には、ローカルのウェブカメラを使用します。
- ストリーミングセッション中にキーボードショートカットを使用する。
- ローカルにインストールされたアプリケーションを操作するのとほぼ同じ方法で、リモートストリーミングアプリケーションを操作する。

要件

Windows 用 AppStream 2.0 クライアントは、以下の要件を満たすコンピュータにインストールする必要があります。

- オペレーティングシステム — Windows 10 (32 ビットまたは 64 ビット)、Windows 11 (64 ビット)
- AppStream 2.0 クライアントバージョン 1.1.1066 以降用の Visual C++ 2019 再頒布可能パッケージ。Visual Studio 2015、2017、2019 の最新 Visual C++ の再配布可能パッケージについては、Microsoft サポートドキュメントの [The latest supported Visual C++ downloads](#) を参照してください。
- RAM — 2 GB 以上
- ハードドライブ容量 — 200 MB 以上

さらに、USB ドライバーをサポートするために AppStream 2.0 USB ドライバーをインストールするには、PC のローカル管理者権限が必要です。

Windows のセットアップ

次の手順に従って、クライアントをインストールします。

1. AppStream 2.0 クライアントをインストールする PC で、[AppStream 2.0 でサポートされているクライアント](#)から Windows アプリケーション用の AppStream 2.0 クライアントをダウンロードします。
2. アプリケーションの .exe ファイルをダウンロードした先に移動し、そのファイルをダブルクリックしてインストールを開始します。

Important

ファイルをダブルクリックしても何も起こらない場合や、エラーメッセージが表示された場合は、ネットワーク管理者に連絡してください。お客様の組織で使用されているウイルス対策ソフトウェアにより、AppStream 2.0 クライアントのインストールプログラムの実行が妨げられている可能性があります。

3. インストールウィザードには、AWS カスタマーアグリーメント、AWS サービス条件、AWS プライバシー通知、およびサードパーティー通知へのリンクが表示されます。この情報を確認して [Next] を選択します。
4. [クライアント診断] ページで、問題のトラブルシューティングに役立つデバイスログを AppStream 2.0 クライアントから自動的にアップロードするには、[クライアントログ記録] を選択したままで [次へ] を選択します。
5. [オプションコンポーネント] ページで、USB デバイスをストリーミングアプリケーションで使用するには、[AppStream 2.0 クライアント USB ドライバー] チェックボックスをオンにし、[終了] を選択します。
6. [AppStream 2.0 USB ドライバー] ウィザードの設定ウィザードが開いたら、[インストール] を選択します。
7. [ユーザーアカウント制御] によって、アプリケーションがデバイスに変更を加えることを許可するかどうかが選択するメッセージが表示されたら、[はい] を選択します。
8. USB ドライバーのインストールが完了したことを通知するメッセージが表示されたら、[Close] を選択します。

AppStream 2.0 のサインインページが開きます。AppStream 2.0 に接続してアプリケーションストリーミングセッションを開始する方法については、[AppStream 2.0 に接続する](#) を参照してください。

Windows クライアントで AppStream 2.0 に接続する

Windows 用 AppStream 2.0 クライアントが PC にインストールされたら、そのクライアントを使用して AppStream 2.0 に接続できます。

トピック

- [AppStream 2.0 クライアント接続モード](#)
- [AppStream 2.0 に接続する](#)
- [AppStream 2.0 接続モードを切り替える方法](#)

AppStream 2.0 クライアント接続モード

AppStream 2.0 クライアントには、[ネイティブアプリケーションモード] と [クラシックモード] の 2 つの接続モードがあります。選択した接続モードによって、アプリケーションのストリーミング中に使用できるオプション、およびストリーミングアプリケーションの機能と表示方法が決まります。また、管理者が [デスクトップビュー] を有効にしている場合は、[デスクトップビュー] も使用できます。

ネイティブアプリケーションモード

ネイティブアプリケーションモードでは、ローカル PC にインストールされているアプリケーションを操作するのとほぼ同じ方法で、リモートストリーミングアプリケーションを操作できます。

ネイティブアプリケーションモードで AppStream 2.0 に接続すると、AppStream 2.0 Application Launcher ウィンドウが開き、ストリーミング可能なアプリケーションのリストが表示されます。このモードでストリーミングアプリケーションを開くと、AppStream 2.0 Application Launcher ウィンドウは開いたままになり、アプリケーションは独自のウィンドウで開きます。ストリーミングセッション中、リモートストリーミングアプリケーションは、ローカルにインストールされたアプリケーションとほぼ同じように機能します。リモートストリーミングアプリケーションのアイコンはローカルアプリケーションのアイコンと同じように、ローカル PC のタスクバーに表示されます。ローカルアプリケーションのアイコンとは異なり、ネイティブアプリケーションモードのストリーミングアプリケーションのアイコンには AppStream 2.0 ロゴが含まれます。

AppStream 2.0 ストリーミングセッション中に、ローカルにインストールされたアプリケーションとリモートストリーミングアプリケーションをすばやく切り替えるには、操作するリモートアプリケーションまたはローカルアプリケーションのタスクバーアイコンをクリックします。AppStream 2.0 接続モードを切り替えることもできます。代わりにクラシックモードで作業する場合は、ネイティブアプリケーションモードからクラシックモードに切り替えることができます。

クラシックモード

クラシックアプリケーションモードを使用している場合は、AppStream 2.0 セッションウィンドウ内でリモートストリーミングアプリケーションを使用します。管理者が複数のアプリケーションを使用できるようにしている場合、セッション中に複数のアプリケーションを開くことができます。すべての開いているアプリケーションが、同じ AppStream 2.0 セッションウィンドウに表示されます。

クラシックモードで AppStream 2.0 に接続すると、AppStream 2.0 Application Launcher ウィンドウが開き、ストリーミング可能なアプリケーションのリストが表示されます。このモードでストリーミングアプリケーションを開くと、Application Launcher ウィンドウが閉じ、アプリケーションは AppStream 2.0 セッションウィンドウで開きます。

管理者がネイティブアプリケーションモードを無効にしていない場合は、クラシックモードからネイティブアプリケーションモードに切り替えることができます。詳細については、「[AppStream 2.0 接続モードを切り替える方法](#)」を参照してください。

Desktop view (デスクトップビュー)

AppStream 2.0 に接続して [Desktop view (デスクトップビュー)] を選択すると、AppStream 2.0 はストリーミングセッション用の標準 Windows デスクトップビューを提供します。ストリーミングできるアプリケーションのアイコンが Windows デスクトップに表示されます。さらに、ストリーミングセッションの設定を構成できる AppStream 2.0 ツールバーが、ストリーミングセッションウィンドウの左上の領域に表示されます。

AppStream 2.0 に接続する

AppStream 2.0 に接続してアプリケーションのストリーミングセッションを開始するには、次の手順に従います。

1. 管理者が組織のサインインページで最初にサインインする必要がある場合は、この手順のタスクを完了してから、手順 3 に進みます。

管理者が組織のサインインページからサインインする必要がない場合は、この手順のタスクをスキップして手順 2 に進みます。

- a. 組織のサインインページに移動し、プロンプトが表示されたら、ドメインの認証情報を入力します。
- b. サインインすると、AppStream 2.0 ポータルにリダイレクトされ、AppStream 2.0 ストリーミングセッションで利用できる 1 つ以上のアプリケーションが表示されます。管理者によって有効になっている場合は、[Desktop View (デスクトップビュー)] も使用できます。

- c. アプリケーションを選択するか、使用可能な場合は [Desktop View (デスクトップビュー)] を選択します。
2. 管理者が組織のサインインページで最初にサインインする必要がない場合は、以下の手順を実行して AppStream 2.0 クライアントを起動します。
 - a. AppStream 2.0 クライアントがインストールされているローカル PC の画面左下で、タスクバーの Windows 検索アイコンを選択し、検索ボックスに「**AppStream**」と入力します。
 - b. 検索結果から [Amazon AppStream] を選択して AppStream 2.0 クライアントを起動します。
 - c. AppStream 2.0 クライアントのサインインページで、クライアントをネイティブアプリケーションモードとクラシックモードのどちらで使用するかを選択できます。
 - ネイティブアプリケーションモードを使用するには、[Start in native application mode (ネイティブアプリケーションモードで起動)] チェックボックスをオンのままにします。
 - クラシックモードを使用するには、[Start in native application mode (ネイティブアプリケーションモードで起動)] チェックボックスをオフにします。
 - d. 次のいずれかを行います。
 - クライアントのサインインページにウェブアドレス (URL) があらかじめ入力されている場合は、[Connect (接続)] を選択します。
 - クライアントのサインインページに URL があらかじめ入力されていない場合は、AppStream 2.0 管理者が AppStream 2.0 用に提供した URL を入力し、[Connect (接続)] を選択します。URL がわからない場合は、管理者に問い合わせてください。
 - e. しばらくすると、AppStream 2.0 ポータルが開き、AppStream 2.0 ストリーミングセッションで利用できる 1 つまたは複数のアプリケーションが表示されます。管理者によって有効になっている場合は、[デスクトップビュー] も使用できます。
 - f. アプリケーションを選択するか、使用可能な場合は [Desktop View (デスクトップビュー)] を選択します。
3. AppStream 2.0 管理者が有効にした認証設定によっては、アプリケーションまたは [Desktop View (デスクトップビュー)] を選択した後、AppStream 2.0 セッションにサインインするために Active Directory ドメインの認証情報を入力するように求められる場合があります。このような場合は、次のいずれかの操作を行います。
 - 組織でパスワード認証が有効になっている場合は、Active Directory ドメインのパスワードを入力し、[Password sign in (パスワードサインイン)] を選択します。

- 組織がスマートカード認証を有効にしている場合は、[Choose a smart card (スマートカードの選択)] を選択し、指示に従ってスマートカード証明書を選んだら、スマートカードの PIN を入力して、[Smart card sign in (スマートカードサインイン)] を選択します。
- 組織が両方の認証方法を有効にしている場合は、Active Directory ドメインのパスワードを入力して [Smart card sign in] を選択するか、[Choose a smart card (スマートカードの選択)] を選択し、指示に従ってスマートカードのサインインを完了します。

AppStream 2.0 接続モードを切り替える方法

管理者がストリーミングセッションのネイティブアプリケーションモードを無効にしていない場合は、ネイティブアプリケーションモードとクラシックモード間で切り替えることができます。

ネイティブアプリケーションモードからクラシックモードに切り替えるには

1. AppStream 2.0 Application Launcher ウィンドウの左上で、[Settings (設定)] アイコンを選択してから、[Switch to classic mode (クラシックモードに切り替え)] を選択します。
2. クラシックモードに切り替えると、Application Launcher ウィンドウが閉じ、AppStream 2.0 セッションウィンドウが開きます。ネイティブアプリケーションモードでストリーミングしているアプリケーションが、AppStream 2.0 セッションウィンドウ内で開きます。

クラシックモードからネイティブアプリケーションモードに切り替えるには、次の手順に従います。

クラシックモードからネイティブアプリケーションモードに切り替えるには

1. AppStream 2.0 セッションウィンドウの左上で、[Settings (設定)] アイコンを選択し、[Switch to native application mode (ネイティブアプリケーションモードに切り替え)] を選択します。
2. クラシックモードからネイティブアプリケーションモードに切り替えると、AppStream 2.0 セッションウィンドウが閉じ、AppStream 2.0 Application Launcher ウィンドウが開きます。クラシックモードでストリーミングしているアプリケーションは別のウィンドウで開きます。

監視

モニターとディスプレイの解像度

AppStream 2.0 では、異なる解像度のモニターを含め、ストリーミングセッション中に複数のモニターを使用できます。最適なストリーミング体験を実現するために、複数のモニターを使用する場合は、モニターの表示縮尺を 100% に設定することをお勧めします。

AppStream 2.0 クライアントは、次の表示解像度を持つ複数のモニターをサポートしています。

- マルチモニター (最大 2K 解像度) — 最大 4 台のモニターと、モニターあたり最大 2,560 x 1,600 ピクセルの表示解像度
- マルチモニター (4K 解像度) — 最大 2 台のモニターと、モニターあたり最大 4,096 x 2,160 ピクセルの表示解像度

Note

ネイティブアプリケーションモードを使用して AppStream 2.0 ストリーミングセッションに接続している場合は、最大 2K 解像度のモニターを使用できます。高解像度のモニターを使用する場合、AppStream 2.0 クライアントはクラシックモードに戻ります。この場合、AppStream 2.0 クラシックモードのストリーミングビューが画面の 2K を占め、画面の残りの部分は黒になります。

マルチモニターの使用

複数のモニターを使用する場合は、次のオプションから選択できます。

- 1 つのモニターにフルスクリーンを拡張する
- すべてのモニターに全画面表示を拡張する
- 選択したモニターに全画面表示を拡張する

1 つのモニターにフルスクリーンを拡張する

複数のモニターがローカルコンピュータに接続されている場合にのみ、現在のモニターで全画面表示を拡張できます。この機能を有効にするには、以下のステップを完了します。

1. ウィンドウ上部のツールバーで、全画面 (交差した矢印) アイコンを選択します。
2. ドロップダウンメニューから、[1 台のモニターで] を選択します。

すべてのモニターに全画面表示を拡張

セッションの表示範囲を全画面解像度ですべてのモニターに拡張できます。拡張画面は物理的な画面アウトと画面解像度に一致します。たとえば、ローカルコンピュータに 3 つのモニターが接続され

ているとします。サーバーにより、3つの画面すべてにわたってセッションの表示範囲が拡張され、解像度が特定の画面解像度に調整されます。

この機能を有効にするには、以下のステップを完了します。

1. ウィンドウ上部のツールバーで、全画面 (交差した矢印) アイコンを選択します。
2. ドロップダウンメニューから [すべてのモニターで] を選択します。

選択したモニターに全画面表示を拡張

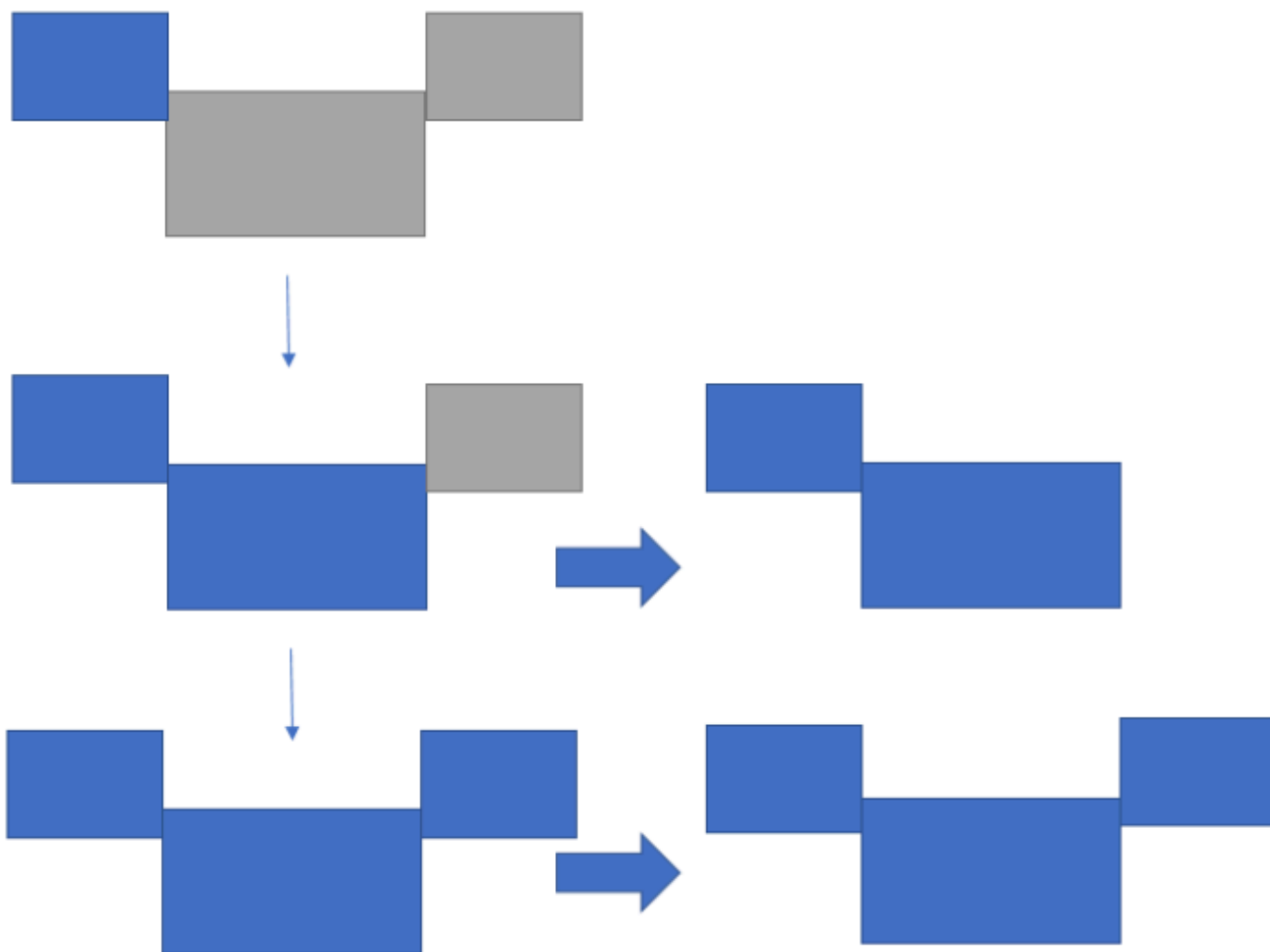
3台以上のモニターが接続されている場合、AppStream 2.0 は選択した使用可能なモニターに全画面表示を拡張することもできます。選択したモニターを全画面表示にできない場合はエラーメッセージが表示されるため、再度手順を実行する必要があります。選択したモニターは、お互いに隣り合うか、1つの側面を共有するようなディスプレイ設定にする必要があります。

以下は、隣接するモニターの配置例です。Windows ディスプレイ構成でモニターが隣接していない場合は AppStream 2.0 を終了し、ローカルマシンのディスプレイ設定を変更する必要があります。

 Note

青いボックスは AppStream 2.0 対応モニターでは、グレーのボックスは他のモニターです。

Examples of adjacent monitor placement



Examples of nonadjacent monitor placement



この機能を有効にするには、以下のステップを完了します。

1. ウィンドウ上部のツールバーで、全画面 (交差した矢印) アイコンを選択します。
2. ドロップダウンメニューから [選択したモニターで] を選択します。
3. [選択したモニターで] ウィンドウが開き、現在のモニターレイアウトが表示されます。DCV を全画面表示するモニターを選択し、[適用] を選択します。

USB デバイス

特定の例外を除き、AppStream 2.0 クライアントが USB デバイスをサポートするには USB リダイレクトが必要です。デバイスに USB リダイレクトが必要な場合は、新しいストリーミングセッションを開始するたびに、デバイスを AppStream 2.0 と共有する必要があります。

トピック

- [ストリーミングセッション中にスマートカードを使用する方法](#)
- [AppStream 2.0 で USB デバイスを共有する方法](#)

ストリーミングセッション中にスマートカードを使用する方法

管理者が有効にした認証設定によっては、AppStream 2.0 ストリーミングセッション中にスマートカードを認証に使用しなければならない場合があります。たとえば、ストリーミングセッション中にブラウザを開き、スマートカード認証を必要とする内部組織サイトに移動する場合は、スマートカードの認証情報を入力する必要があります。

デフォルトでは、スマートカードリダイレクトは AppStream 2.0 ストリーミングセッションで有効になっています。つまり、ローカルコンピュータに接続されているスマートカードリーダーは、AppStream 2.0 と共有しなくても使用できます。ストリーミングセッション中、スマートカードリーダーとスマートカードは、ローカルアプリケーションやストリーミングアプリケーションで使用できます。

管理者がスマートカードリダイレクトを無効にしている場合は、スマートカードリーダーを AppStream 2.0 と共有する必要があります。詳細については、次のセクションをご覧ください。

AppStream 2.0 で USB デバイスを共有する方法

ドローイングタブレットを使用している場合は、AppStream 2.0 での使用には USB リダイレクトは必要ないこともあります。ただし、Gnu 画像編集プログラム (GIMP) などのアプリケーションをストリーミングする場合は、圧力感度をサポートするために USB リダイレクトが必要となるため、ドローイングタブレットを AppStream 2.0 と共有する必要があります。ドローイングタブレットの詳細については、「[ドローイングタブレット](#)」をご参照ください。

USB デバイスを AppStream 2.0 と共有するには

1. AppStream 2.0 クライアントを使用してストリーミングセッションを開始します。
2. 左上の領域で、[設定] アイコンを選択し、[USB デバイス] を選択します。
3. USB デバイスがコンピュータに接続されている場合、USB デバイス名がダイアログボックスに表示されます。USB デバイスが検出されない場合は、AppStream 2.0 ネットワーク管理者に問い合わせてください。
4. ストリーミングセッションで共有する USB デバイスの名前の横にある [共有] トグルキーを切り替えます。

USB デバイスをストリーミングアプリケーションで使用できるようになりました。

Important

USB デバイスは、ローカルアプリケーションとリモートアプリケーションの間で同時に使用することはできません。したがって、USB デバイスをストリーミングセッションで共有した後は、ローカルコンピュータのアプリケーションで使用することはできません。ローカルコンピュータで USB デバイスを使用するには、ローカルで使用する USB デバイスの名前の横にある [共有] トグルキーを切り替えます。これにより、ストリーミングセッションとの共有が無効になります。

5. 新しいストリーミングセッションの開始時に自動的に接続するように USB デバイスを設定することもできます。そのためには、接続する USB デバイスのトグルキーの横にあるオプションを選択します。このオプションを有効にすると、次のストリーミングセッションの開始時に USB デバイスが自動的に接続されます。

ローカルファイルアクセス

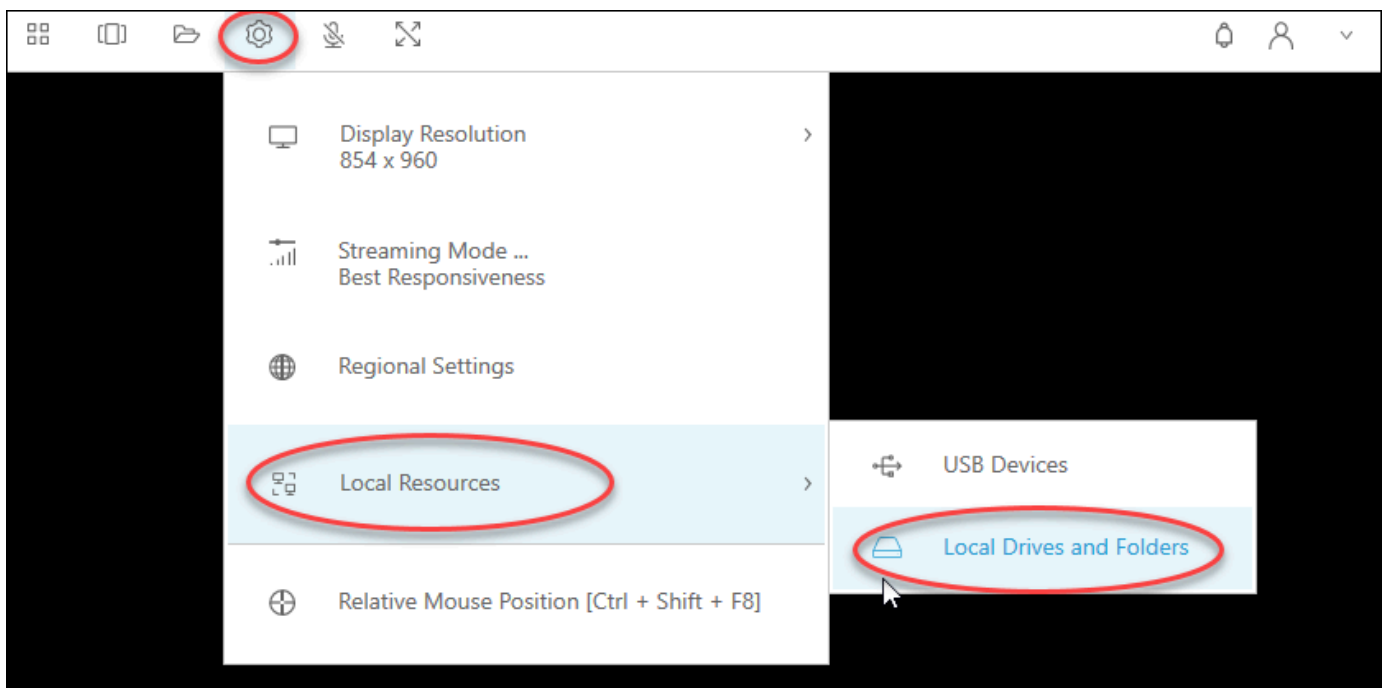
AppStream 2.0 ファイルリダイレクトを使用すると、AppStream 2.0 ストリーミングセッションからローカルコンピュータのファイルにアクセスできます。ファイルリダイレクトを使用するには、AppStream 2.0 クライアントを開き、ストリーミングセッションに接続して、共有するドライブとフォルダを選択します。ローカルドライブまたはフォルダを共有すると、ストリーミングセッションから共有ドライブまたは共有フォルダ内のすべてのファイルにアクセスできます。ローカルドライブとフォルダの共有はいつでも停止できます。

⚠ Important

AppStream 2.0 ファイルリダイレクトを使用するには、ローカルコンピュータに AppStream 2.0 クライアントがインストールされている必要があります。ウェブブラウザを使用して AppStream 2.0 に接続する場合、ファイルのリダイレクトは使用できません。

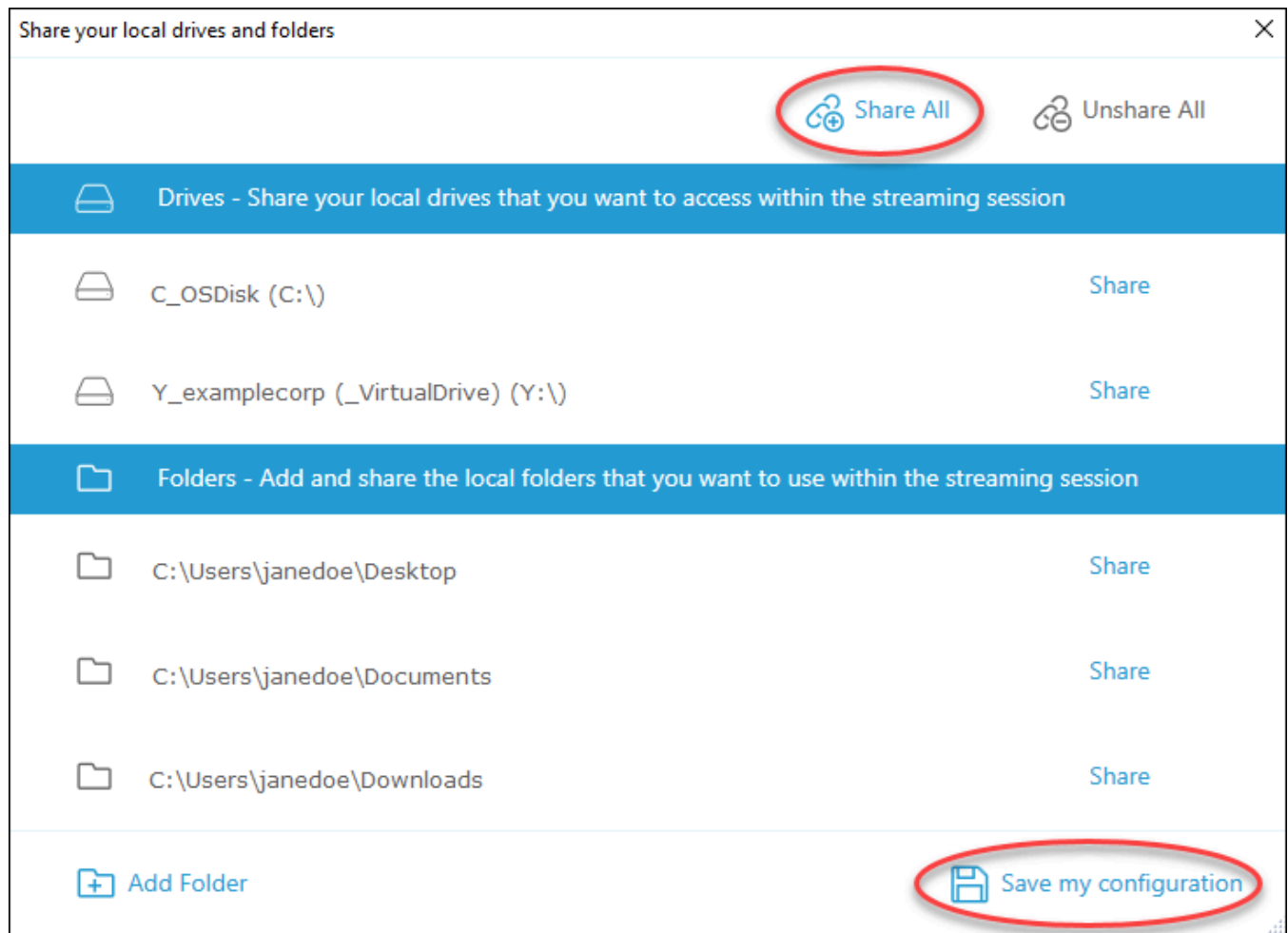
ローカルドライブとフォルダを共有するには

1. AppStream 2.0 クライアントを開き、ストリーミングセッションに接続します。
2. AppStream 2.0 セッションで、左上領域の [Settings (設定)] アイコンを選択し、[Local Resources (ローカルリソース)]、[Local Drives and Folders (ローカルドライブとフォルダ)]の順に選択します。

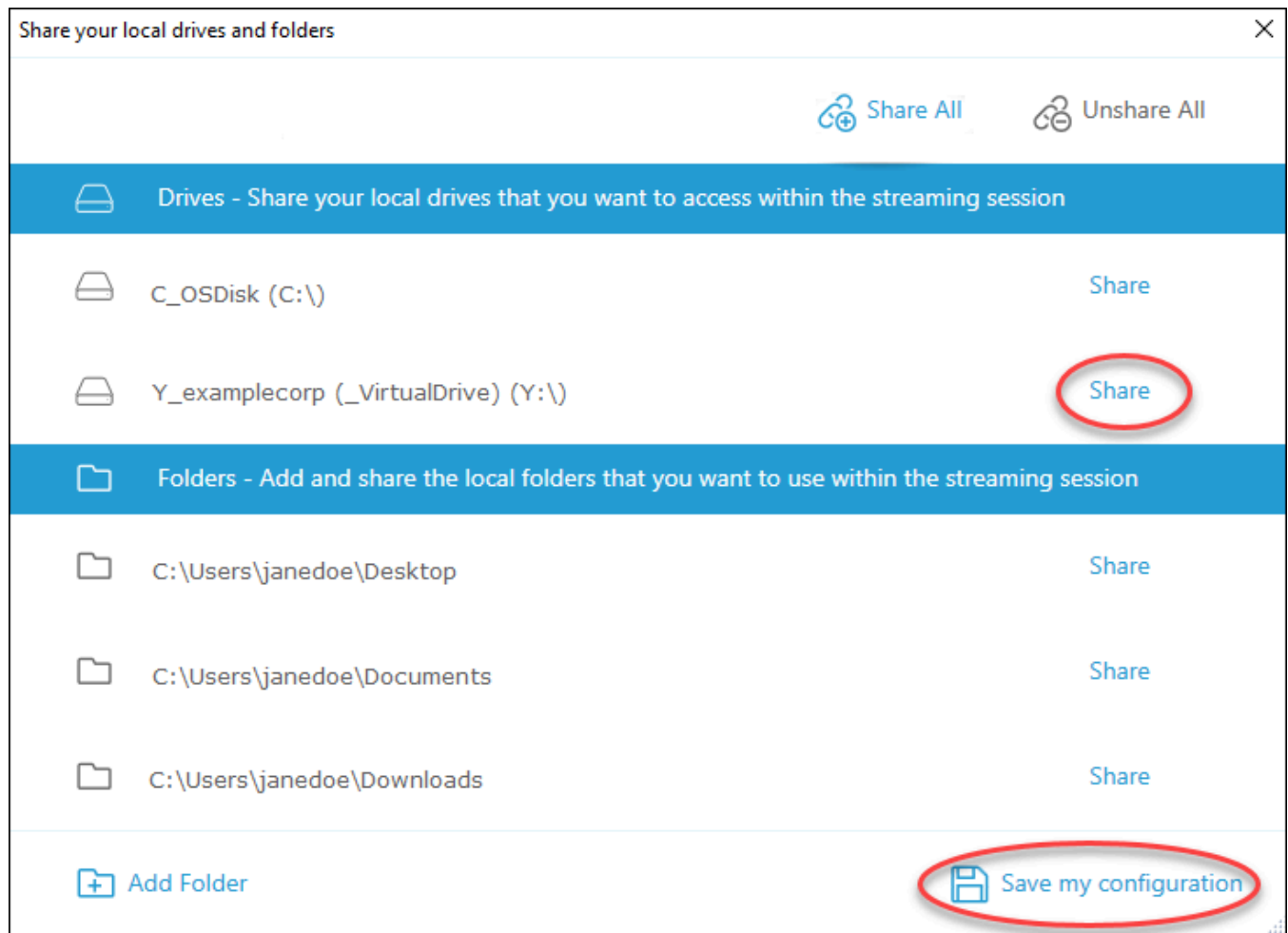


[Share your local drives and folders (ローカルドライブとフォルダを共有する)] ダイアログボックスに、管理者が共有できるようにしたドライブとフォルダが表示されます。すべてまたは特定のドライブとフォルダ、または 1 つのみを共有できます。独自のドライブとフォルダを追加することもできます。ドライブとフォルダを共有するには、次のいずれかを実行します。

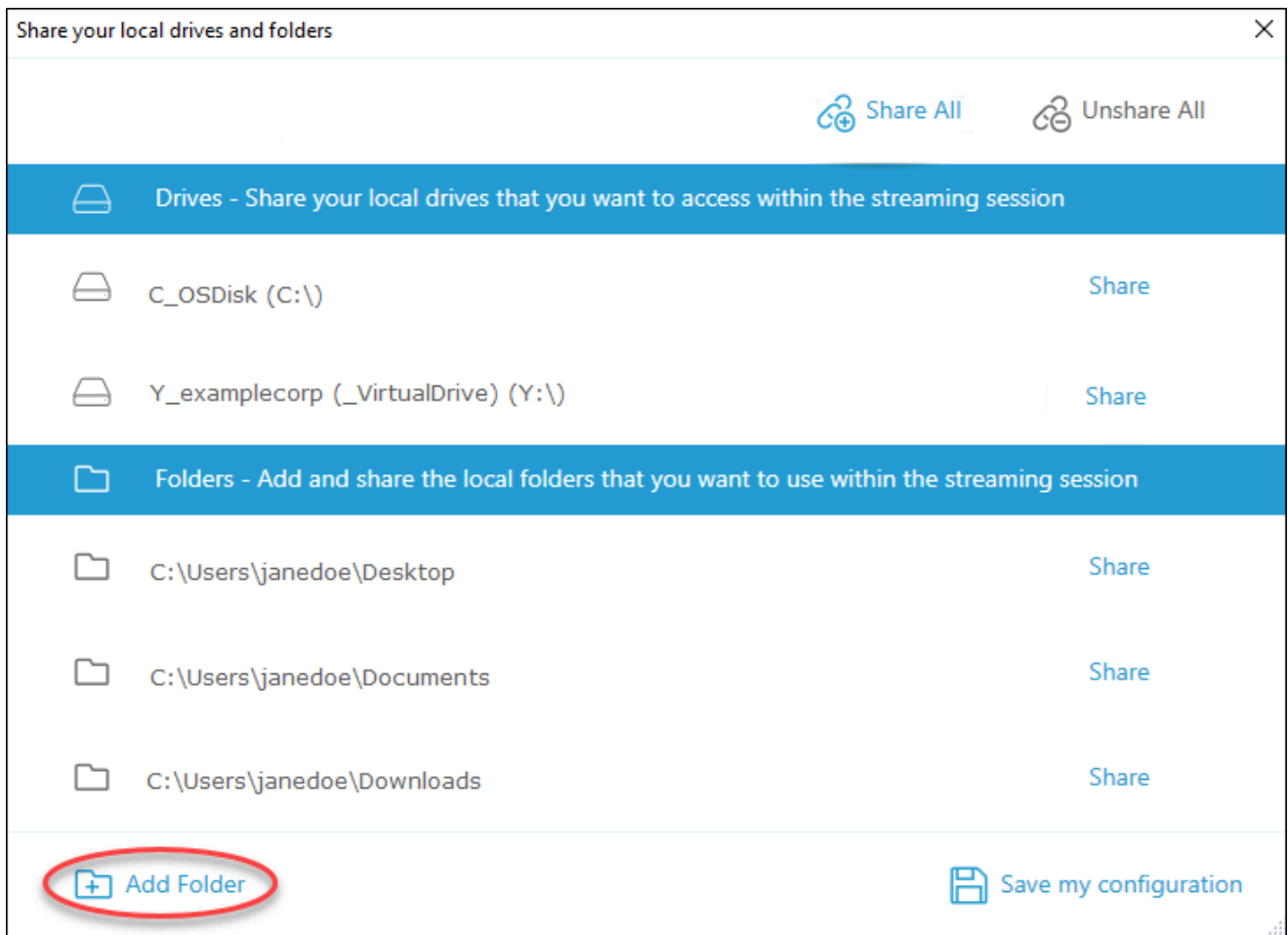
- [Share your local drives and folders] ダイアログボックスに表示されるすべてのローカルドライブおよびフォルダを共有するには、[Share All] を選択します。今後のストリーミングセッションに変更を適用するには、[Save my configuration] を選択します。



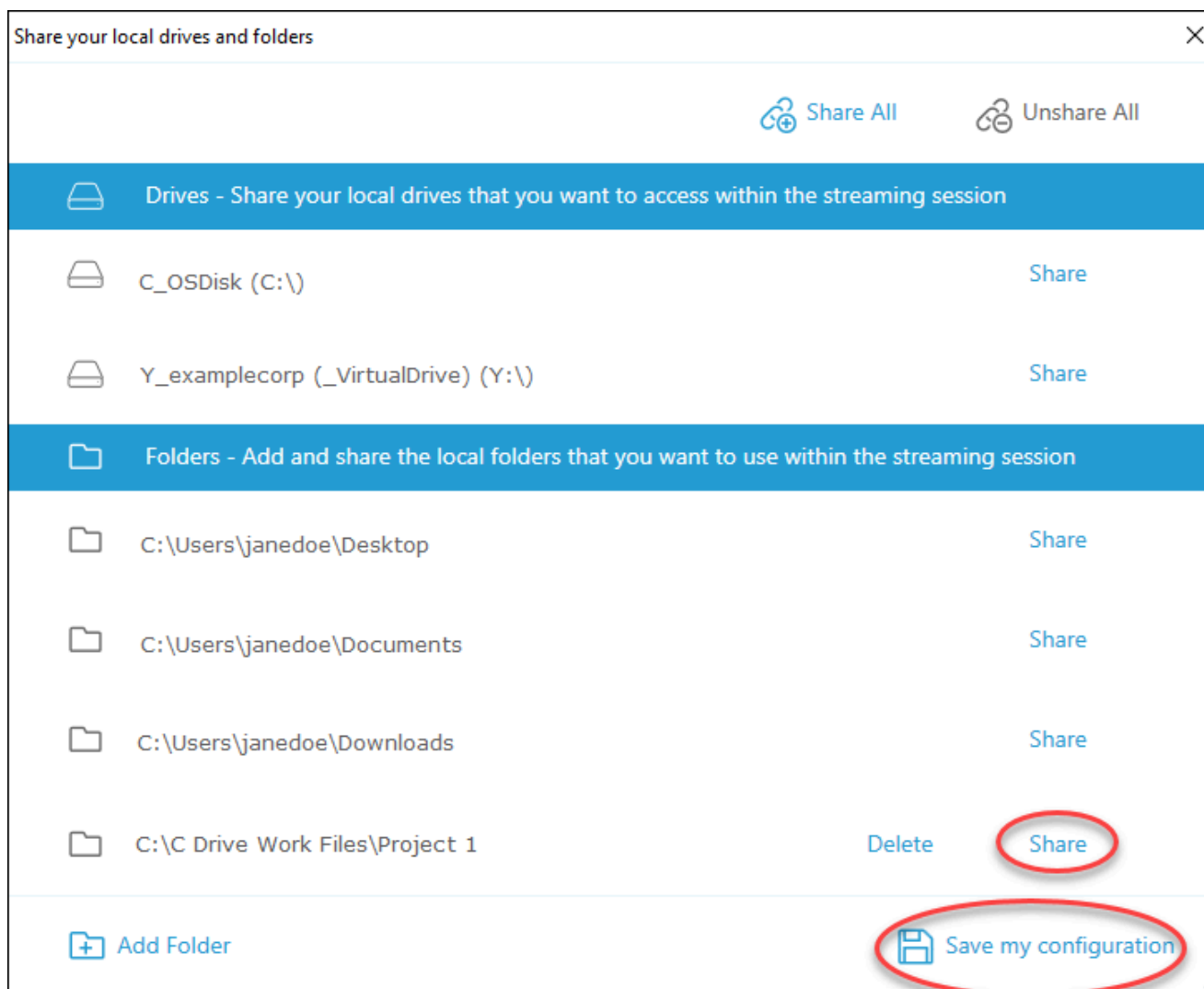
- 特定のローカルドライブまたはフォルダを共有するには、アクセスするドライブまたはフォルダを選択し、[Share (共有する)]、[Save my configuration (設定を保存する)] の順に選択します。別のローカルドライブまたはフォルダを共有するには、必要に応じてこれらのステップを繰り返します。



- 共有するローカルドライブやフォルダが表示されない場合は、それを追加できます。たとえば、管理者がローカルの C ドライブ全体を共有可能にしたとします。ただし、ユーザーがアクセスする必要があるのは、そのドライブの特定のフォルダのみであるとしています。この場合、ユーザーは必要なフォルダを追加し、そのフォルダのみを共有できます。フォルダを選択するには、次の操作を行います。
- [Share your local drives and folders] ダイアログボックスで、[Add Folder] を選択します。



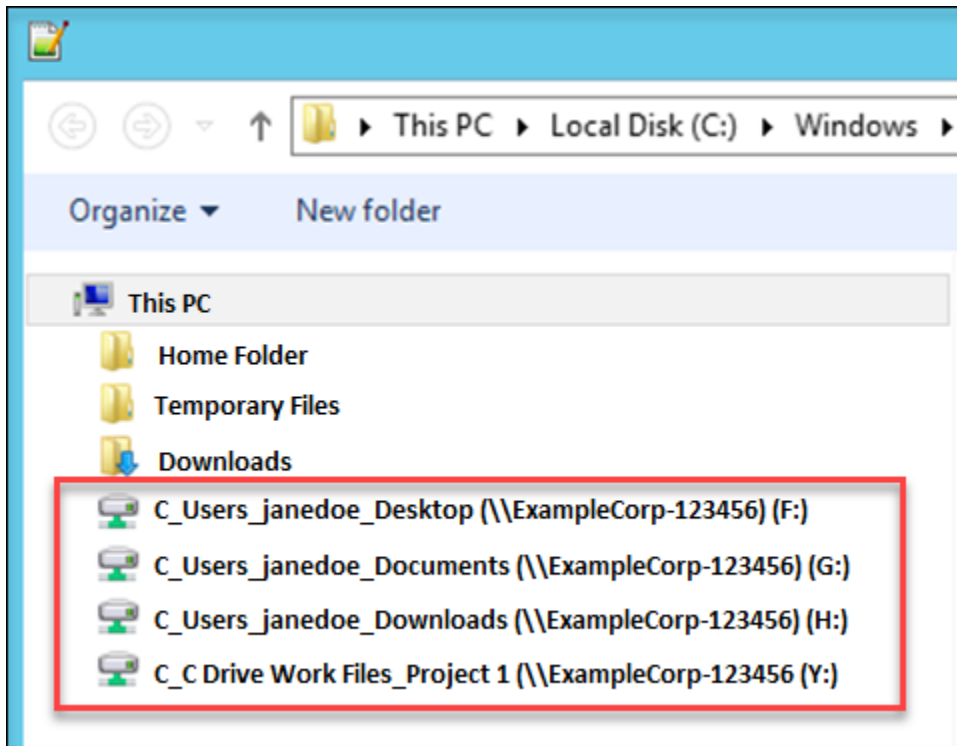
- 共有するフォルダを参照し、[OK] を選択します。
- これで、選択したフォルダを共有できるようになります。フォルダを選択し、[共有]、[Save my configuration (設定を保存する)] の順に選択します。別のローカルドライブまたはフォルダを追加するには、必要に応じてこれらのステップを繰り返します。



ローカルドライブまたはフォルダを共有したら、次の手順を実行して、ストリーミングセッションから共有ドライブまたは共有フォルダのファイルにアクセスします。

共有ローカルドライブまたはフォルダのファイルにアクセスするには

1. AppStream 2.0 クライアントを開き、ストリーミングセッションに接続します。
2. AppStream 2.0 セッションで、使用するアプリケーションを開きます。
3. アプリケーションインターフェイスから、[File Open (ファイルを開く)] を選択し、アクセスするファイルを参照します。次のスクリーンショットは、Jane Doe がメモ帳++ の参照ダイアログボックスでファイルを参照したときに、共有ローカルドライブおよびフォルダがどのように表示されるかを示しています。



参照ダイアログボックスで、共有ドライブとフォルダの対応するパスが赤色のボックスに表示されます。パスは、バックスラッシュをアンダースコアに置き換えて表示されます。各パスの末尾には、Jane のコンピュータ名、ExampleCorp-123456、およびドライブ文字があります。

4. ファイルの操作が完了したら、[File Save (ファイルを保存)] または [File Save As (名前を付けてファイルを保存)] コマンドを使用して、目的の場所に保存します。

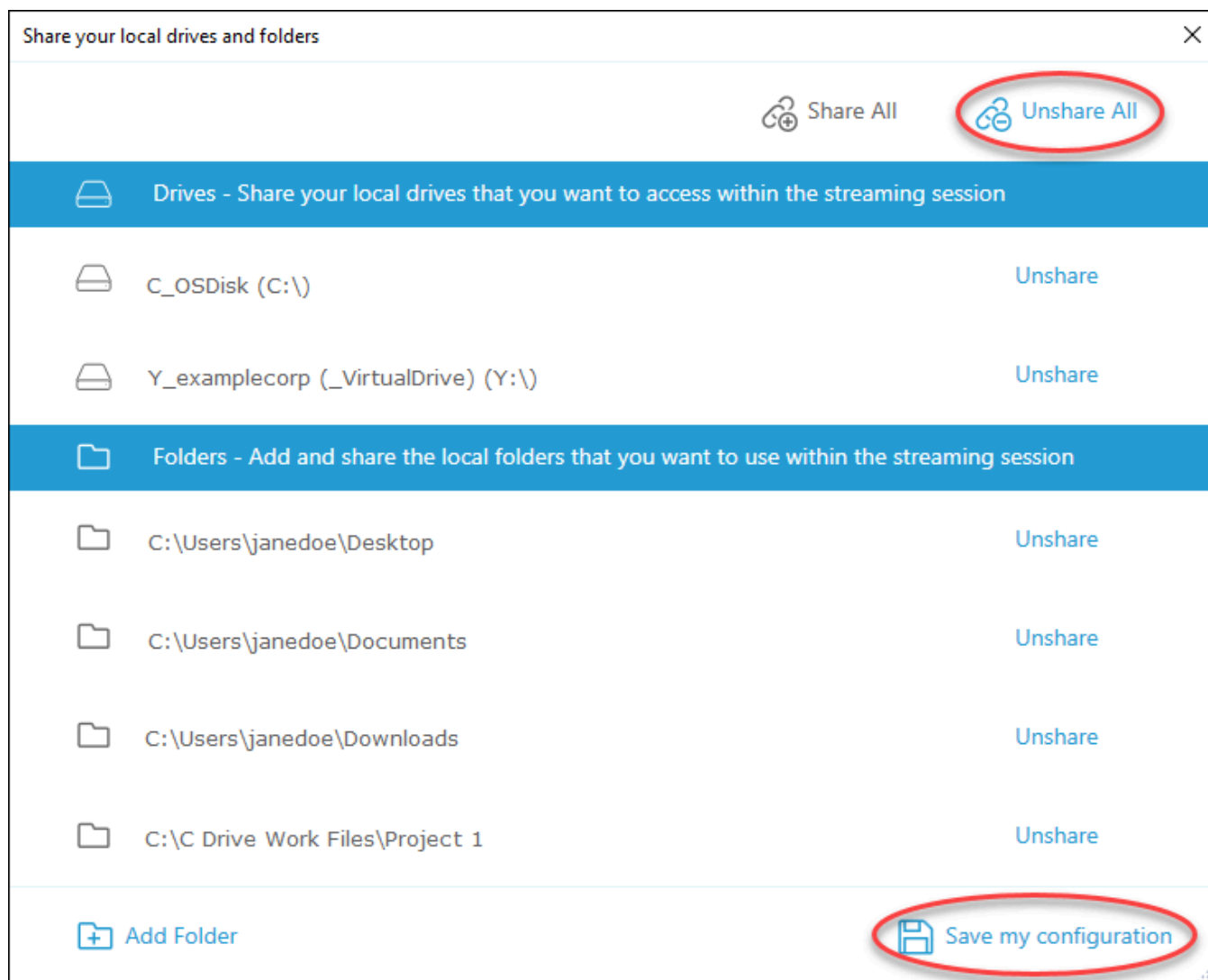
ローカルドライブまたはフォルダの共有を停止する場合は、以下の手順を実行します。

ローカルドライブとフォルダの共有を停止するには

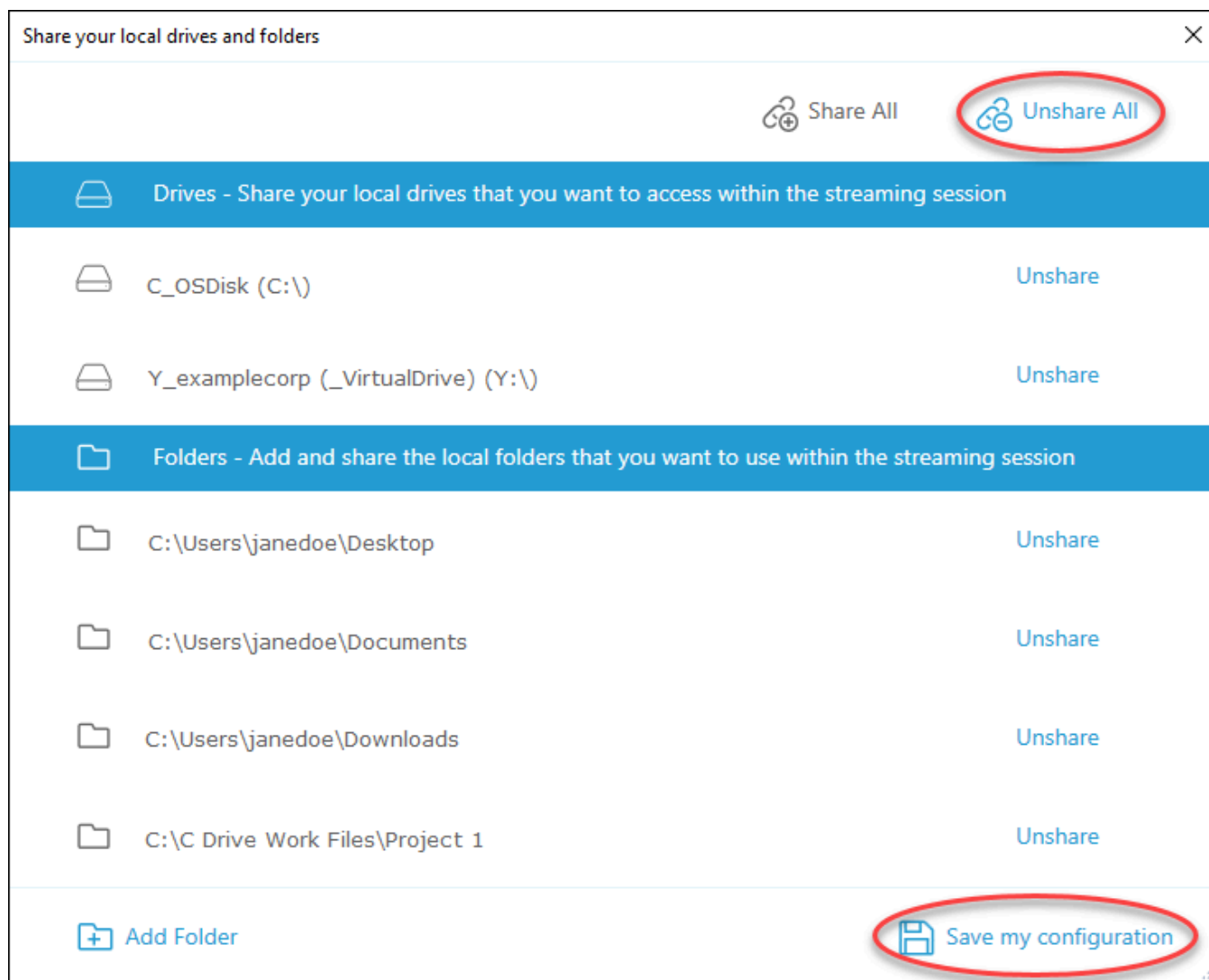
1. AppStream 2.0 クライアントを開き、ストリーミングセッションに接続します。
2. AppStream 2.0 セッションで、左上領域の [Settings (設定)] アイコンを選択し、[Local Resources (ローカルリソース)]、[Local Drives and Folders (ローカルドライブとフォルダ)] の順に選択します。

[Share your local drives and folders (ローカルドライブとフォルダを共有する)] ダイアログボックスに、管理者が共有できるようにしたドライブとフォルダ、および追加したドライブとフォルダ (該当する場合) が表示されます。1 つ以上のローカルドライブとフォルダの共有を停止するには、次のいずれかを実行します。

- すべての共有ローカルドライブとフォルダの共有を停止するには、[Unshare All (全てを共有解除する)]、[Save my configuration (設定を保存)]の順に選択します。



- 特定の共有ローカルドライブまたはフォルダの共有を停止するには、ドライブまたはフォルダを選択し、[Unshare (共有解除)]、[Save my configuration (設定を保存)] の順に選択します。その他のローカルドライブまたはフォルダの共有を停止するには、必要に応じてこれらのステップを繰り返します。



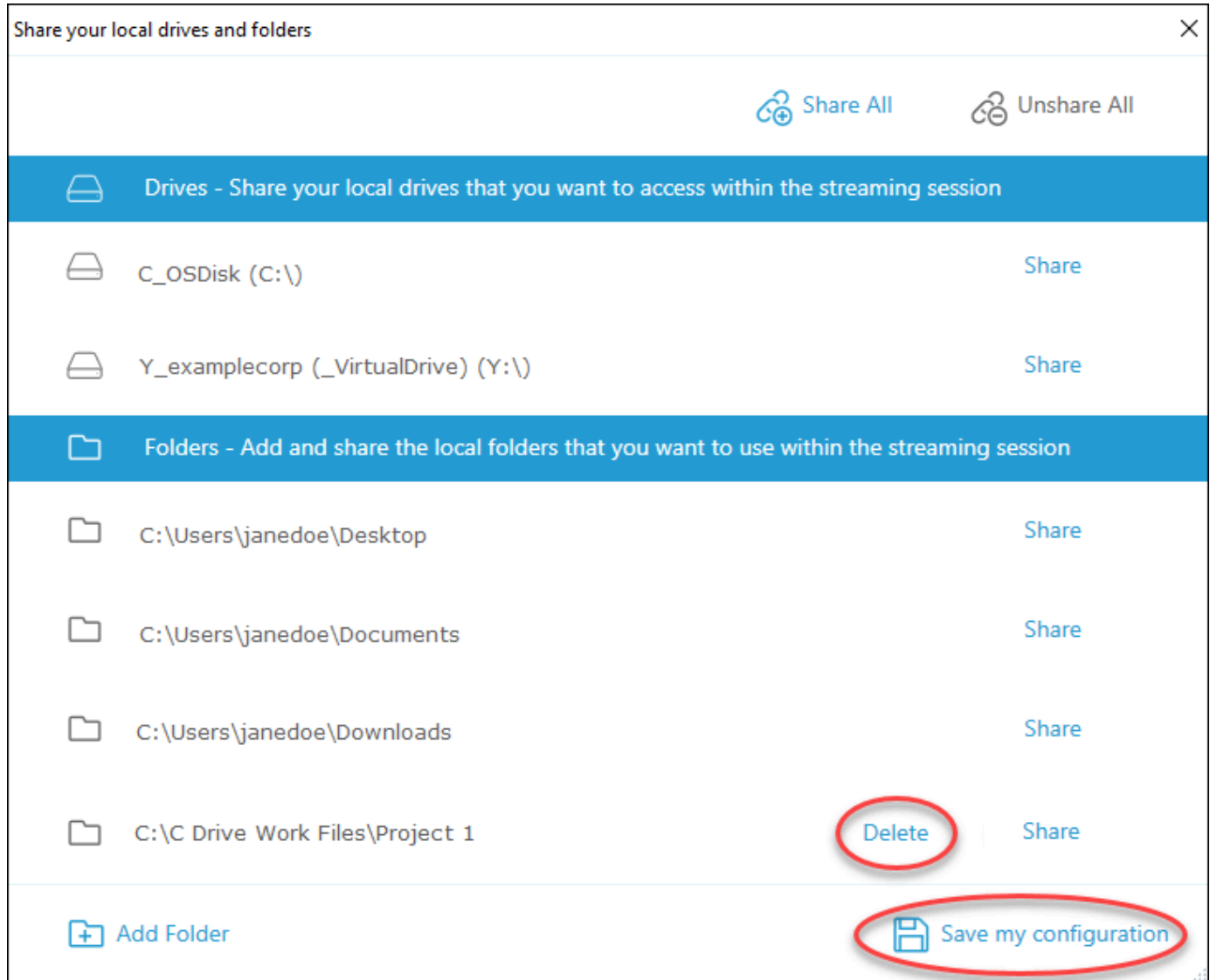
[Share your local drives and folders (ローカルドライブとフォルダを共有)] ダイアログボックスに追加するローカルドライブとフォルダを削除できます。ただし、管理者が共有できるようにしたローカルドライブまたはフォルダを削除することはできません。また、ローカルドライブまたはフォルダをすでに共有している場合は、削除する前に共有を停止する必要があります。

ローカルドライブとフォルダを削除するには

1. AppStream 2.0 クライアントを開き、ストリーミングセッションに接続します。
2. AppStream 2.0 セッションで、左上領域の [Settings (設定)] アイコンを選択し、[Local Resources (ローカルリソース)]、[Local Drives and Folders (ローカルドライブとフォルダ)] の順に選択します。

[Share your local drives and folders (ローカルドライブとフォルダを共有する)] ダイアログボックスに、管理者が共有できるようにしたドライブとフォルダが表示されます。ドライブまたはフォルダを追加した場合は、それらも表示されます。

3. 削除するローカルドライブまたはフォルダを選択し、[Delete (削除)]、[Save my configuration (設定を保存)] の順に選択します。



プリンターリダイレクト

AppStream 2.0 ローカルプリンターリダイレクトを使用すると、AppStream 2.0 ストリーミングセッションからローカルコンピュータに接続されているプリンターにアクセスできます。これにより、プリントジョブをストリーミングアプリケーションからローカルプリンター、またはマッピングしたネットワークプリンターにリダイレクトできます。

⚠ Important

AppStream 2.0 プリンターリダイレクトを使用するには、AppStream 2.0 クライアントをローカルコンピュータにインストールし、クライアントを使用してストリーミングセッションに接続する必要があります。ウェブブラウザを使用して AppStream 2.0 に接続する場合、ファイルのリダイレクトは使用できません。

プリントジョブをローカルプリンターにリダイレクトするには

1. AppStream 2.0 クライアントを開き、ストリーミングセッションに接続します。
2. ストリーミングアプリケーションで、[File (ファイル)]、[Print Now (今すぐプリント)] の順に選択します。
3. AppStream 2.0 セッションウィンドウの右上の領域で、通知アイコンの横に表示される新しい通知を選択します。
4. [Notifications (通知)] ダイアログボックスで、[Print Jobs (プリントジョブ)] タブを選択します。
5. [Print Jobs (プリントジョブ)] タブで、[Print (プリント)] を選択します。
6. ストリーミングアプリケーションの [Print (プリント)] ダイアログボックスが開きます。
7. [Print (プリント)] ダイアログボックスに、使用可能なローカルプリンター一覧が表示されます。使用するローカルプリンターを選択し、プリントを続行します。

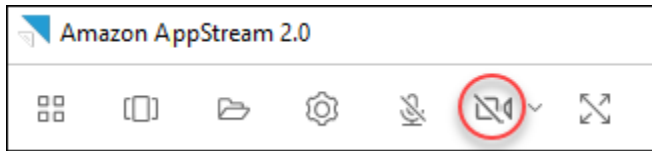
ビデオ会議と音声会議

AppStream 2.0 リアルタイムオーディオビデオ (AV) は、ローカルのウェブカメラビデオ入力を AppStream 2.0 ストリーミングセッションにリダイレクトします。これにより、AppStream 2.0 ストリーミングセッション内で、ローカルデバイスを使用したビデオ会議や音声会議を行うことができます。

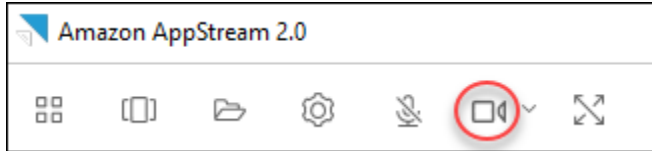
AppStream 2.0 ストリーミングセッション内でローカルのウェブカメラやマイクを使用するには

1. AppStream 2.0 クライアントを開き、ストリーミングセッションに接続します。
2. セッションウィンドウの左上にある AppStream 2.0 ツールバーで、次のいずれかの操作を行います。
 - 次のスクリーンショットに示すように、ビデオアイコンに斜線が付いている場合、AppStream 2.0 リアルタイム AV 機能は使用できませんが、ストリーミングセッションにウェブカメラが接

続されていないことを示します。ビデオアイコンを選択して、1 つまたは複数のウェブカメラをアタッチします。



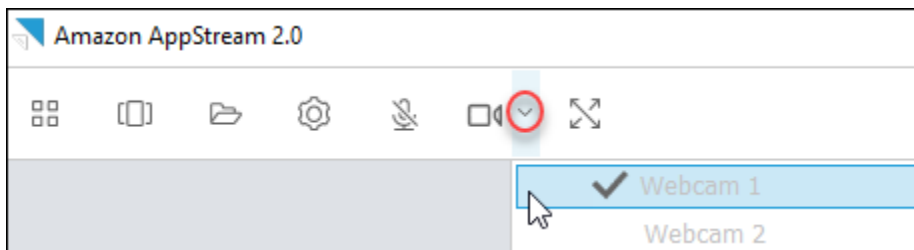
- 次のスクリーンショットに示すように、ビデオアイコンに斜線がない場合は、ストリーミングセッションに 1 つ以上のウェブカメラがすでにアタッチされています。この手順をスキップして、次の手順に進みます。



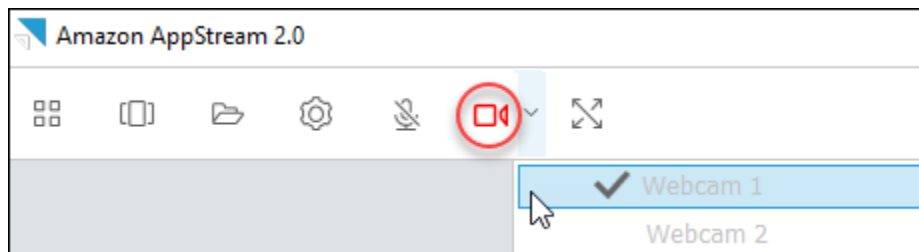
Note

AppStream 2.0 ツールバーにビデオアイコンが表示されない場合は、AppStream 2.0 の管理者にお問い合わせください。管理者は、[リアルタイムオーディオビデオ](#) の説明通りに、追加の設定タスクを実行しなければならない場合があります。

- ストリーミングセッションに接続されているウェブカメラの名前を表示するには、ビデオアイコンの横にある下向き矢印を選択してください。複数のウェブカメラがある場合 (ラップトップに接続されている USB ウェブカメラと組み込みのウェブカメラがある場合など)、ストリーミングセッション内のビデオ会議で使用するために選択されているウェブカメラの名前の横にチェックマークが表示されます。



- AppStream 2.0 ストリーミングセッション内で、選択したウェブカメラをビデオ会議で使用するには、使用するビデオ会議アプリケーションを起動してください。ウェブカメラがアクティブになっている場合 (ストリーミングセッション内のビデオ会議で使用されている)、ビデオアイコンは赤になります。



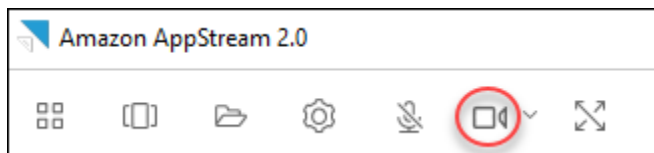
5. マイクを有効にするには、マイクのアイコンを選択します。

Note

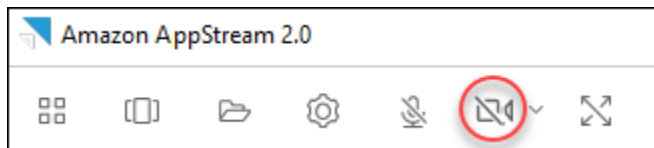
複数のウェブカメラがあり、AppStream 2.0 セッション内でストリーミングに使用するウェブカメラを変更する場合は、まずウェブカメラの接続をセッションから切り離す必要があります。詳細については、次の手順を参照してください。

AppStream 2.0 ストリーミングセッションで使用するようローカルウェブカメラを変更するには

1. AppStream 2.0 ストリーミングセッションで、セッションウィンドウの左上の AppStream 2.0 ツールバーで、次のいずれかの操作を行います。
 - 次のスクリーンショットに示すように、ビデオアイコンに斜線がない場合、これは AppStream 2.0 リアルタイム AV 機能が使用可能であり、ウェブカメラがストリーミングセッションに接続されていることを示します。ビデオアイコンを選択してウェブカメラを取り外します。



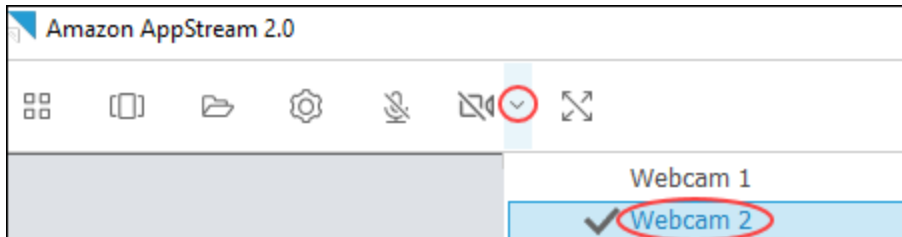
- 次のスクリーンショットに示すように、ビデオアイコンに斜線が付いている場合、ウェブカメラはすでにストリーミングセッションから接続を解除されています。この手順をスキップして、次の手順に進みます。



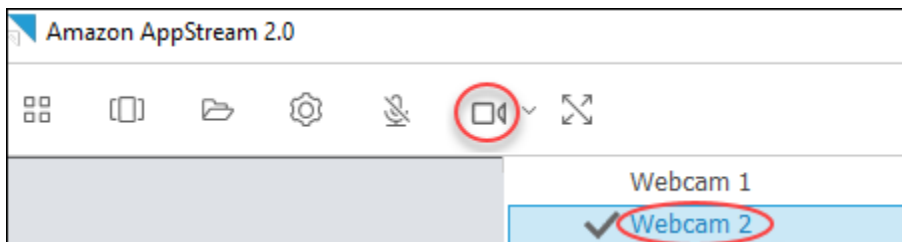
2. ビデオアイコンの横にある下向き矢印を選択してウェブカメラの名前を表示し、使用するウェブカメラの名前を選択します。

Note

使用するウェブカメラの名前を選択する必要があります。使用するウェブカメラの名前の横にあるチェックマークをオンにすると、ウェブカメラは変更されません。



3. ビデオアイコンを選択して、ウェブカメラを AppStream 2.0 ストリーミングセッションに再接続します。



ドローイングタブレット

ドローイングタブレット (ペントラレットとも呼ばれます) は、スタイラス (ペン) を使って絵を描くことができるコンピュータ入力デバイスです。AppStream 2.0 では、Wacom ドローイングタブレットなどのドローイングタブレットをローカルコンピュータに接続し、そのタブレットをストリーミングアプリケーションで使用できます。

ストリーミングアプリケーションでドローイングタブレットを使用するための要件と考慮事項を次に示します。

- この機能を使用するには、AppStream 2.0 クライアントを使用するか、Google Chrome または Mozilla Firefox ブラウザでのみ、AppStream 2.0 に接続する必要があります。
- ストリーミングするアプリケーションは、Windows Ink テクノロジーをサポートしている必要があります。詳細については、「[Windows アプリでのペン操作と Windows インク](#)」を参照してください。

- 使用するストリーミングアプリケーションによっては、ドローイングタブレットが想定どおりに機能するために、USB リダイレクトが必要になる場合があります。これは、GIMP などの一部のアプリケーションでは、圧力感度をサポートするために USB リダイレクトが必要なためです。ストリーミングアプリケーションでこれが該当する場合は、AppStream 2.0 クライアントを使用して AppStream 2.0 に接続し、ドローイングタブレットをストリーミングセッションと共有する必要があります。
- この機能は Chromebook ではサポートされていません。

アプリケーションストリーミングセッション中にドローイングタブレットの使用を開始するには、USB を使用してドローイングタブレットをローカルコンピュータに接続し、圧力感度の検出に必要な場合は AppStream 2.0 とデバイスを共有してから、AppStream 2.0 ストリーミングセッションを開始します。AppStream 2.0 クライアントまたは[サポートされているウェブブラウザ](#)を使用して、ストリーミングセッションを開始できます。

相対マウスオフセット

デフォルトでは、ストリーミングセッション中、AppStream 2.0 は絶対座標を使用してマウスの動きをローカルにレンダリングすることにより、マウスの動きに関する情報を送信します。コンピュータ支援設計 (CAD)/コンピュータ支援製造 (CAM) ソフトウェアやビデオゲームなど、グラフィックを多用するアプリケーションの場合、相対的マウスモードが有効になっているときにマウスのパフォーマンスが向上します。相対的マウスモードでは、ウィンドウまたは画面内の XY 座標の絶対値ではなく、最終フレームからマウスが移動した距離を表す相対座標が使用されます。相対的マウスモードを有効にすると、AppStream 2.0 はマウスの動きをリモートでレンダリングします。

AppStream 2.0 ストリーミングセッション中に、次のいずれかの方法でこの機能を有効にできます。

- Ctrl + Shift + F8 を押す
- ストリーミングセッションウィンドウの左上にある AppStream 2.0 ツールバーの [Settings (設定)] メニューから [Relative Mouse Position (相対的マウスの位置)] [Ctrl + Shift + F8]] を選択します。この方法は、クラシックモードまたは [Desktop view (デスクトップビュー)] を使用している場合に利用できます。

ログ記録

AppStream 2.0 クライアントで問題が発生した場合のトラブルシューティングに役立つように、診断ログを有効にすることができます。AppStream 2.0 (AWS) に送信されるログファイルには、デバイスおよび AWS ネットワークへの接続に関する詳細情報が含まれます。ログの自動アップロードを有

効にすると、これらのファイルは自動的に AppStream 2.0 (AWS) に送信されます。AppStream 2.0 ストリーミングセッションの実行前または実行中に、必要に応じてログファイルをアップロードすることもできます。

自動ログ記録

AppStream 2.0 クライアントのインストール時に、自動ログ記録を有効にすることができます。AppStream 2.0 クライアントのインストール時に自動ログ記録を有効にする方法については、[Windows のセットアップ](#) のステップ 5 を参照してください。

オンデマンドのログ記録

AppStream 2.0 ストリーミングセッション中に問題が発生した場合は、必要に応じてログファイルを送信することもできます。AppStream 2.0 クライアントが応答を停止するような問題が発生した場合は、エラーレポートおよび関連するログファイルを AppStream 2.0 (AWS) に送信するかどうかを選択するよう通知が表示されます。

次の手順では、AppStream 2.0 ストリーミングセッションにサインインする前と AppStream 2.0 ストリーミングセッション中にログファイルを送信する方法を示します。

AppStream 2.0 ストリーミングセッションの前にログファイルを送信するには

1. AppStream 2.0 クライアントがインストールされているローカル PC の画面左下で、タスクバーの Windows 検索アイコンを選択し、検索ボックスに **AppStream** と入力します。
2. 検索結果から [Amazon AppStream] を選択して AppStream 2.0 クライアントを起動します。
3. AppStream 2.0 サインインページの下部で、[Send Diagnostic Logs (診断ログの送信)] リンクを選択します。
4. AppStream 2.0 への接続を続行するには、AppStream 2.0 への接続に使用するウェブアドレス (URL) が AppStream 2.0 管理者からアプリケーションストリーミング用に提供されている場合、その URL を入力して [Connect (接続)] を選択します。

AppStream 2.0 ストリーミングセッション中にログファイルを送信するには

1. まだ AppStream 2.0 に接続してアプリケーションをストリーミングしていない場合は、AppStream 2.0 クライアントを使用してストリーミングセッションを開始します。
2. AppStream 2.0 セッションウィンドウの右上にある [Profiles (プロファイル)] アイコンを選択してから、[Send Diagnostic Logs (診断ログの送信)] を選択します。

トラブルシューティング

Windows 用の AppStream 2.0 クライアントを使用しているときに問題が発生した場合、AppStream 2.0 クライアント ID とバージョン番号が管理者によるトラブルシューティングに役立ちます。以下のセクションでは、クライアント ID とクライアントバージョン番号を見つける方法について説明します。

AppStream 2.0 クライアント ID を見つける方法

AppStream 2.0 クライアント ID はデバイスを一意に識別します。この ID は、Windows 用の AppStream 2.0 クライアントをインストールするときに作成されます。クライアント ID を確認するには、AppStream 2.0 クライアントを開きます。クライアントのサインインページの左下で、[Client Options (クライアントオプション)] リンクを選択します。クライアント ID は、[AppStream 2.0 Client Options (AppStream 2.0 クライアントオプション)] ダイアログボックスの上部に表示されます。マウスを使用してクライアント ID を選択し、キーボードまたはマウスを使用してクリップボードにコピーできます。

AppStream 2.0 クライアントのバージョン番号を確認する方法

AppStream 2.0 では、機能を追加したり問題を解決したりした新しいクライアントバージョンを定期的にリリースしています。インストールした AppStream 2.0 クライアントのバージョンを確認するには、AppStream 2.0 クライアントを開きます。クライアントのサインインページの下部にある [About Amazon AppStream 2.0 (Amazon AppStream 2.0 のバージョン情報)] リンクを選択します。クライアントバージョンは、Amazon AppStream 2.0 ロゴの下に表示されます。

AppStream 2.0 クライアントリリースノート

AppStream 2.0 クライアントは、AppStream 2.0 ストリーミングセッション中に追加機能を必要とするユーザー用に設計されたネイティブアプリケーションです。次の表は、AppStream 2.0 クライアントのリリース済みバージョンで利用可能な最新の更新プログラムを示しています。

クライアントの詳細については、[AppStream 2.0 クライアント経由でアクセスを提供する](#) を参照してください。

クライアントバージョン	リリース日	変更
1.1.1458	06-10-2025	埋め込み Chromium ブラウザをバージョン 135.0.170 にアップグレード

クライアントバージョン	リリース日	変更
		• バグ修正と改良が含まれます。
1.1.1440	06-06-2025	• 埋め込み Chromium ブラウザをバージョン 131.3.50 に戻します • バグ修正と改良が含まれます。
1.1.1437	05-21-2025	• 埋め込み Chromium ブラウザをバージョン 135.0.170 にアップグレード • バグ修正と改良が含まれます。
1.1.1423	03-31-2025	• 埋め込み Chromium ブラウザをバージョン 131.3.50 にアップグレード • バグ修正と改良が含まれます。
1.1.1414	01-16-2025	• 自動タイムゾーンリダイレクトのサポートを追加 • バグ修正と改良が含まれます。
1.1.1408	12-19-2024	• バグ修正と改良が含まれます。

クライアントバージョン	リリース日	変更
1.1.1403	12-12-2024	• ストリーミングセッション間でユーザー設定を保存するサポートを追加 • バグ修正と改良が含まれます。
1.1.1395	11-18-2024	• 埋め込み Chromium ブラウザをバージョン 129.0.110 にアップグレード • バグ修正と改良が含まれます。
1.1.1360	08-01-2024	• 選択したモニターに全画面表示を拡張するサポートを追加 • Red Hat Enterprise Linux イメージをストリーミングするサポートを追加 • 組み込みの Chromium ブラウザをバージョン 125.0.210 にアップグレードします。 • バグ修正と改良が含まれます。
1.1.1332	07-03-2024	• バグ修正と改良が含まれます。

クライアントバージョン	リリース日	変更
1.1.1326	06-17-2024	- システムブラウザでユーザーがサインインした後、クライアントを自動的に開くことで、IdP 主導の SSO ワークフローのユーザーエクスペリエンスを改善 - 他のバグ修正と機能向上
1.1.1303	04-03-2024	バグ修正と改良が含まれます。
1.1.1300	03-28-2024	- IdP 開始ストリーミングセッションから AppStream 2.0 クライアントを起動するサポートを追加 - 新しいリリースステートリージョンエンドポイントのサポートを追加 - 組み込みの Chromium ブラウザをバージョン 121.3.70 にアップグレード - バグ修正と改良が含まれます。
1.1.1259	02-08-2024	バグ修正と改良が含まれます。

クライアントバージョン	リリース日	変更
1.1.1246	01-18-2024	- アクセシビリティ機能の向上が含まれます。 - バグ修正と改良が含まれます。 - 組み込みの Chromium ブラウザをバージョン 119.4.30 にアップグレード
1.1.1228	11-01-2023	- バグ修正と改良が含まれます。 - 組み込みの Chromium ブラウザをバージョン 114.1.120 にアップグレード
1.1.1183	06-22-2023	- バグ修正と改良が含まれます。 - 組み込みの Chromium ブラウザをバージョン 111.2.20 にアップグレード
1.1.1159	05-09-2023	バグ修正と改良が含まれます。
1.1.1130	02-09-2023	組み込みの Chromium ブラウザをバージョン 108.4.130 にアップグレード
1.1.1118	11-07-2022	組み込みの Chromium ブラウザをバージョン 106.0.26 にアップグレードします。

クライアントバージョン	リリース日	変更
1.1.1099	10-13-2022	• バグ修正と改良が含まれます。
1.1.1066	08-17-2022	• 組み込みの Chromium ブラウザをバージョン 102.0.9 にアップグレードします。Microsoft Visual C++ 2019 再頒布可能パッケージを前提条件としてインストールする必要があります。
1.1.1025	06-29-2022	• UDP ストリーミングのサポートを追加します。詳細については、Amazon AppStream 2.0 は Windows ネイティブクライアントの UDP ストリーミングを有効にするを参照してください。
1.1.421	05-19-2022	• バグ修正を含む
1.1.414	04-26-2022	• バグ修正と UI の改良点を含む
1.1.398	02-23-2022	• バグ修正を含む
1.1.394	02-08-2022	• 組み込みの Chromium ブラウザをバージョン 97 にアップグレードします

クライアントバージョン	リリース日	変更
1.1.386	12-20-2021	- 組み込みの Chromium ブラウザをバージョン 94.4 にアップグレードします - バグ修正を含む
1.1.360	11-15-2021	- Linux アプリケーションストリーミングのサポートを追加 - Elastic フリートのサポートを追加しました。詳細については、「Amazon AppStream 2.0 が Elastic フリートを追加」を参照してください。 - 日本語キーボードのバグを修正
1.1.333	09-08-2021	組み込み Chromium ブラウザのバグ修正
1.1.319	08-16-2021	- キャップロック、数字ロック、スクロールロックキーの問題を解決しました。 - ドメイン参加サインインエクスペリエンスの問題を解決します。

クライアントバージョン	リリース日	変更
1.1.304	08-02-2021	- 組み込みの Chromium ブラウザをバージョン 91 にアップグレードします - 重要な修正を含む USB ドライバの更新
1.1.294	04-26-2021	- SAML 2.0 認証に関する問題を解決 - Windows 7 でのクライアントの安定性の問題を解決 - クライアント再接続時のフォルダ共有に関する問題を解決
1.1.285	2021 年 3 月 8 日	ウイルス対策ソフトウェアとの互換性を向上させる修正を含みます

クライアントバージョン	リリース日	変更
1.1.257	2020 年 12 月 28 日	- リアルタイムオーディオビデオ (AV) のサポートを追加 - Active Directory に参加しているストリーミングインスタンスに Windows でサインインするためのスマートカードの使用と、ストリーミングアプリケーションのセッション内認証のサポートを追加します。 - ストリーミングセッション中に Microsoft Excel シートがフォーカスを失う問題を解決します

クライアントバージョン	リリース日	変更
1.1.195	2020 年 8 月 18 日	• ストリーミングセッション中にファイルリダイレクトを使用する場合、OneDrive などのクラウドベースの永続ストレージソリューションに属するローカルドライブとフォルダを共有する際の操作性が向上します。 • 組み込みの Chromium ブラウザをバージョン 81 にアップグレードします • AS2TrustedDomain リストで指定されたドメインの AS2TrustedDomains DNS TXT レコード検索に失敗する問題を解決します。これらの障害は、一部の URI スキームで発生する可能性があります。詳細については、「AS2TrustedDomains DNS TXT レコードを作成して、レジストリを変更せずに AppStream 2.0 クライアントのドメインを有効にします。」を参照してください。 • オーディオが有効なときにクライアントが機

クライアントバージョン	リリース日	変更
		能を断続的に停止する問題を解決します
1.1.179	2020 年 7 月 8 日	- ローカルプリンターへのリダイレクトのサポートを追加する - いくつかのプロキシ設定でストリーミングを妨げる同時実行 HTTP 接続問題を解決します - 数ギガバイトを超えるファイルのダウンロードが停止し、失敗する問題を解決します。 - ユーザーが SAML 経由でサインインして AppStream 2.0 に接続し、AppStream 2.0 クライアントを閉じずにセッションから接続を解除し、新しい AppStream 2.0 ストリーミングセッションを開始しようとする、その後の AppStream 2.0 への接続試行が失敗する問題を解決します。

クライアントバージョン	リリース日	変更
1.1.160	04-28-2020	• .NET Framework バージョン 4.7.1 以前がインストールされている Windows PC でアプリケーションカタログページが開かない問題を解決します。 • ユーザーがクライアントアプリケーションを閉じると、クライアントからの応答が停止するという断続的な問題を解決します。

クライアントバージョン	リリース日	変更
1.1.156	04-22-2020	• DNS TXT レコードでユーザー接続用の信頼されたサブドメインを定義するためのサポートを追加しました。 • 診断ログとミニダンプのオンデマンドアップロードのサポートを追加しました。 • ネイティブアプリケーションモードでストリーミングするユーザーのカスタムブランドのサポートを追加しました。

 **Note**

このバージョンの AppStream 2.0 クライアントをインストールしているユーザーは、同じ PC に .NET Framework バージョン 4.7.2 以降をインストールする必要があります。ダウンロードできる .NET Framework のバージョンの一覧については、[「Download .NET」](#)

クライアントバージョン	リリース日	変更
		Framework 」を参照してください。
1.1.137	03-08-2020	バージョン 1.1.136 の更新プログラムに戻します。
1.1.136	03-05-2020	DNS TXT レコードでユーザー接続用の信頼されたサブドメインを定義するためのサポートを追加しました。
1.1.129	02-28-2020	- ネイティブアプリケーションモードのサポートを追加しました。 - DCV プリンターエクスペリエンスのユーザーインターフェイスを改良しました。 - ストリーミングアプリケーションで Surface Pro Pen を使用する際の問題を解決しました。 - 国際文字を含む名前の付いたファイルをダウンロードする際の問題を解決しました。

クライアントバージョン	リリース日	変更
1.0.525	12-12-2019	ストリーミングセッション中にユーザーがアプリケーションをクリックすると、マウスカーソルが間違っ場所を指す原因となる DPI の問題を解決しました。
1.0.511	10-16-2019	- 最大 4 台のモニターと、モニターあたり最大 2,560 x 1,600 ピクセルの表示解像度のサポートを追加 - Graphics Design および Graphics Pro インスタンスタイプで最大 2 台のモニターと、モニターあたり最大 4,096 x 2,160 ピクセルの表示解像度のサポートを追加 - カスタム Uniform Resource Identifier (URI) リダイレクトを使用して開始されたストリーミングセッションへのシームレスなユーザー接続のサポートを追加 - 開始 URL の信頼されたドメインを追加するためのサポートを追加

クライアントバージョン	リリース日	変更
1.0.499	09-26-2019	- クライアント側のハードウェアレンダリングに関する問題を解決しました。 - Bluetooth ヘッドセットがローカルコンピュータに接続されているときにクライアントが正しく動作しない問題を解決しました。
1.0.480	2019-08-20	AppStream 2.0 ファイルシステムリダイレクトのサポートを追加
1.0.467	07-29-2019	AppStream 2.0 ポータルエンドポイントに対して行われた更新との互換性を確保するための修正と機能強化が含まれています

クライアントバージョン	リリース日	変更
1.0.407	05-16-2019	• ユーザーがストリーミングセッションから切断されるまでにアイドル状態 (非アクティブ) となることができる時間を設定するためのサポートを追加しました。詳細については、Amazon AppStream 2.0 フリートとスタックを作成するの「Create a Fleet」を参照してください。 • SAML 2.0 セッションが期限切れになったときに表示されるセッションアラートウィンドウに関する問題を解決しました。 • プリントサーバーへのドキュメントの印刷に関するバグ修正が含まれます。

クライアントバージョン	リリース日	変更
1.0.375	03-07-2019	• Windows PC でのタッチスクリーンのサポートを追加 • 新しいストリーミングセッションの開始時に USB デバイスを自動的に接続することが可能に • セッションスクリプトの実行のサポートを追加 • AppStream 2.0 の動的アプリケーションフレームワーク API を使用して仮想化アプリケーションを配信するためのサポートを追加

クライアントバージョン	リリース日	変更
1.0.320	01-19-2019	- グラフィックデザインインスタンスのマルチモニターサポートを追加 100% を超えるクライアント表示倍率のサポートを追加 - AppStream 2.0 リージョン設定のサポートを追加します。 - AppStream 2.0 ユーザープールのサポートを追加 - クライアント側のプロキシ設定を尊重するためのサポートを追加
1.0.247	11-20-2018	初回リリース

macOS 用 AppStream 2.0 クライアントアプリケーション

以下の情報は、macOS 用の AppStream 2.0 クライアントを使用して AppStream 2.0 に接続し、アプリケーションをストリーミングするのに役立ちます。

内容

- [要件](#)
- [macOS のセットアップとインストール](#)
- [macOS クライアントで AppStream 2.0 に接続する](#)
- [監視](#)
- [ビデオ会議と音声会議](#)
- [相対マウスオフセット](#)
- [Windows ロゴキーまたはコマンドキーを再マッピングする](#)

- [設定を記憶する](#)
- [プリンターリダイレクト](#)
- [セッションの切断と終了](#)
- [トラブルシューティング](#)
- [AppStream 2.0 macOS クライアントリリースノート](#)

要件

macOS 用 AppStream 2.0 クライアントは、次の要件を満たすコンピュータにインストールする必要があります。

- オペレーティングシステム — macOS 13 (Ventura)、macOS 14 (Sonoma)、または macOS 15 (Sequoia)
- ハードドライブ容量 — 200 MB 以上

macOS のセットアップとインストール

クライアントアプリケーションをダウンロードしてインストールするには、次の手順に従います。

1. macOS デバイスで Amazon [Amazon AppStream 2.0 Downloads](#) を開き、macOS リンクを選択します。
2. アプリケーションをダウンロードし、インストールします。
3. AppStream 2.0 クライアントアプリケーションアイコンが Mac Launchpad に表示されることを確認するか、`/Users/username/Applications/` または `~/Applications` にチェックインします。

macOS クライアントで AppStream 2.0 に接続する

macOS 用 AppStream 2.0 クライアントが PC にインストールされたら、それを使用して AppStream 2.0 に接続できます。

トピック

- [AppStream 2.0 macOS クライアント接続モード](#)
- [AppStream 2.0 に接続する](#)

AppStream 2.0 macOS クライアント接続モード

AppStream 2.0 macOS クライアントは、クラシックモードとデスクトップビューの 2 つの接続モードをサポートしています。管理者が接続モードをセットアップします。

クラシックモード

クラシックアプリケーションモードを使用している場合は、AppStream 2.0 セッションウィンドウ内でリモートストリーミングアプリケーションを使用します。管理者が複数のアプリケーションを使用できるようにしている場合、セッション中に複数のアプリケーションを開くことができます。すべての開いているアプリケーションが、同じ AppStream 2.0 セッションウィンドウに表示されます。

クラシックモードで AppStream 2.0 に接続すると、AppStream 2.0 Application Launcher ウィンドウが開き、ストリーミング可能なアプリケーションのリストが表示されます。このモードでストリーミングアプリケーションを開くと、Application Launcher ウィンドウが閉じ、アプリケーションは AppStream 2.0 セッションウィンドウで開きます。

Desktop view (デスクトップビュー)

AppStream 2.0 に接続して [Desktop view (デスクトップビュー)] を選択すると、AppStream 2.0 はストリーミングセッション用の標準 Windows デスクトップビューを提供します。ストリーミングできるアプリケーションのアイコンが Windows デスクトップに表示されます。さらに、ストリーミングセッションの設定を構成できる AppStream 2.0 ツールバーが、ストリーミングセッションウィンドウの左上の領域に表示されます。

AppStream 2.0 に接続する

AppStream 2.0 に接続してアプリケーションのストリーミングセッションを開始するには、次の手順に従います。

1. 管理者が組織のサインインページで最初にサインインする必要がある場合は、この手順のタスクを完了してから、手順 3 に進みます。

管理者が組織のサインインページからサインインする必要がない場合は、この手順のタスクをスキップして手順 2 に進みます。

- a. 組織のサインインページに移動し、プロンプトが表示されたら、ドメインの認証情報を入力します。
- b. サインインすると、AppStream 2.0 Application Manager カタログページにリダイレクトされ、AppStream 2.0 ストリーミングセッションで利用できる 1 つ以上のアプリケーション

が表示されます。管理者によって有効になっている場合は、[デスクトップビュー] も使用できます。

- c. アプリケーションを選択するか、使用可能な場合は [Desktop View (デスクトップビュー)] を選択します。
2. 管理者が組織のサインインページで最初にサインインする必要がない場合は、以下の手順を実行して AppStream 2.0 クライアントを起動します。
 - a. AppStream 2.0 クライアントがインストールされているローカルコンピュータで、Amazon AppStream 2.0 を選択して AppStream 2.0 クライアントを起動します。
 - b. 次のいずれかを実行します。
 - クライアントのサインインページにウェブアドレス (URL) があらかじめ入力されている場合は、[Connect (接続)] を選択します。macOS クライアントのストリーミングセッションに移行する前に、認証のためにシステムのデフォルトのウェブブラウザにリダイレクトされます。
 - クライアントのサインインページに URL があらかじめ入力されていない場合は、AppStream 2.0 管理者が AppStream 2.0 用に提供した URL を入力し、[Connect (接続)] を選択します。また、macOS クライアントのストリーミングセッションに移行する前に、認証のためにシステムのデフォルトのウェブブラウザにリダイレクトされる場合があります。URL がわからない場合は、管理者に問い合わせてください。
 - c.しばらくすると、AppStream 2.0 Application Manager カタログページが開き、AppStream 2.0 ストリーミングセッションで利用できる 1 つ以上のアプリケーションが表示されます。管理者によって有効になっている場合は、[デスクトップビュー] も使用できます。
 - d. アプリケーションを選択するか、使用可能な場合は [Desktop View (デスクトップビュー)] を選択します。
3. AppStream 2.0 管理者が有効にした認証設定によっては、アプリケーションまたは [Desktop View (デスクトップビュー)] を選択した後、AppStream 2.0 セッションにサインインするために Active Directory ドメインの認証情報を入力するように求められる場合があります。この場合、Active Directory ドメインのパスワードを入力し、パスワードサインインを選択します。

監視

モニターとディスプレイの解像度

AppStream 2.0 クライアントは、次の表示解像度を持つ複数のモニターをサポートしています。

- マルチモニター (最大 2K 解像度) — 最大 4 台のモニターと、モニターあたり最大 2,560 x 1,600 ピクセルの表示解像度
- マルチモニター (4K 解像度) — 最大 2 台のモニターと、モニターあたり最大 4,096 x 2,160 ピクセルの表示解像度

クライアントウィンドウのサイズを変更しても変更されない固定解像度が必要な場合は、設定、表示、表示解像度を選択し、希望する解像度を指定します。自動サイズ変更を再度有効にするには、自動的に適応を選択します。

マルチモニターの使用

複数のモニターを使用する場合は、次のオプションから選択できます。

- 1 つのモニターにフルスクリーンを拡張する
- すべてのモニターに全画面表示を拡張する
- 選択したモニターに全画面表示を拡張する

1 つのモニターにフルスクリーンを拡張する

複数のモニターがローカルコンピュータに接続されている場合にのみ、現在のモニターで全画面表示を拡張できます。この機能を有効にするには、以下のステップを完了します。

1. ウィンドウ上部のツールバーで、全画面 (交差した矢印) アイコンを選択します。
2. ドロップダウンメニューから、全画面表示の現在のモニターを選択します。

すべてのモニターに全画面表示を拡張

セッションの表示範囲を全画面解像度ですべてのモニターに拡張できます。拡張画面は物理的な画面アウトと画面解像度に一致します。たとえば、ローカルコンピュータに 3 つのモニターが接続されているとします。サーバーにより、3 つの画面すべてにわたってセッションの表示範囲が拡張され、解像度が特定の画面解像度に調整されます。

この機能を有効にするには、以下のステップを完了します。

1. ウィンドウ上部のツールバーで、全画面 (交差した矢印) アイコンを選択します。
2. ドロップダウンメニューから、全画面表示のすべてのモニターを選択します。

選択したモニターに全画面表示を拡張

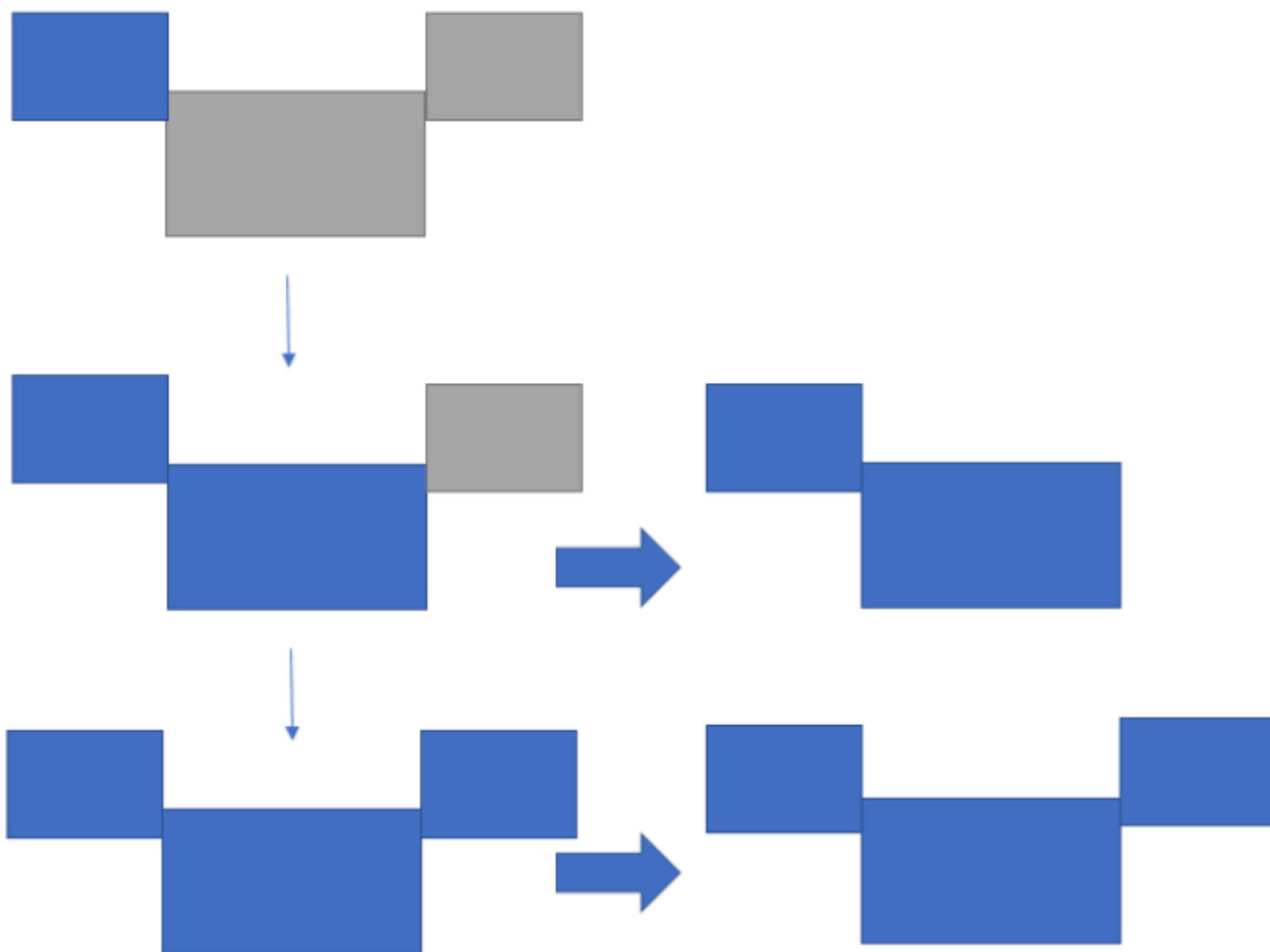
3 台以上のモニターが接続されている場合、AppStream 2.0 は選択した使用可能なモニターに全画面表示を拡張することもできます。選択したモニターを全画面表示にできない場合はエラーメッセージが表示されるため、再度手順を実行する必要があります。選択したモニターは、お互いに隣り合うか、1 つの側面を共有するようなディスプレイ設定にする必要があります。

以下は、隣接するモニターの配置例です。モニタがディスプレイ設定で隣接して設定されていない場合は、AppStream 2.0 を終了し、ローカルマシンでディスプレイ設定を変更する必要があります。

Note

青いボックスは AppStream 2.0 対応モニターでは、グレーのボックスは他のモニターです。

Examples of adjacent monitor placement



Examples of nonadjacent monitor placement



この機能を有効にするには、以下のステップを完了します。

1. ウィンドウ上部のツールバーで、全画面 (交差した矢印) アイコンを選択します。
2. ドロップダウンメニューから、選択したモニターの全画面表示を選択します。
3. 選択した全画面モニターウィンドウが表示され、現在のモニターレイアウトが表示されます。DCV を全画面表示するモニターを選択し、[適用] を選択します。

ビデオ会議と音声会議

AppStream 2.0 リアルタイムオーディオビデオ (AV) は、ローカルのウェブカメラビデオ入力を AppStream 2.0 ストリーミングセッションにリダイレクトします。これにより、AppStream 2.0 ストリーミングセッション内で、ローカルデバイスを使用したビデオ会議や音声会議を行うことができます。

AppStream 2.0 でウェブカメラまたはマイクを使用するには、システムメニューの設定を選択し、オーディオとビデオを選択してマイクとウェブカメラを有効または無効にします。複数のウェブカメラがある場合 (たとえば、ラップトップに接続されている USB ウェブカメラと組み込みウェブカメラがある場合)、Active Webcam から 1 つのウェブカメラを選択することもできます。

Show Toolbar を選択した場合は、マイクまたはビデオアイコンを選択して、マイクとウェブカメラを有効または無効にすることもできます。複数のウェブカメラがある場合は、ビデオアイコンの横にある下矢印を選択し、使用するウェブカメラを 1 つ選択できます。

Note

AppStream 2.0 ツールバーにビデオアイコンが表示されない場合は、AppStream 2.0 の管理者に問い合わせてください。管理者は、[the section called “リアルタイムオーディオビデオ”](#) の説明通りに、追加の設定タスクを実行しなければならない場合があります。

相対マウスオフセット

デフォルトでは、ストリーミングセッション中、AppStream 2.0 は絶対座標を使用してマウスの動きをローカルにレンダリングすることにより、マウスの動きに関する情報を送信します。コンピュータ支援設計 (CAD)/コンピュータ支援製造 (CAM) ソフトウェアやビデオゲームなど、グラフィックを多用するアプリケーションの場合、相対的マウスモードが有効になっているときにマウスのパフォーマンスが向上します。相対的マウスモードでは、ウィンドウまたは画面内の XY 座標の絶対値ではなく、最終フレームからマウスが移動した距離を表す相対座標が使用されます。相対的マウスモードを有効にすると、AppStream 2.0 はマウスの動きをリモートでレンダリングします。

AppStream 2.0 ストリーミングセッション中に、次のいずれかの方法でこの機能を有効にできます。

- Ctrl+Shift+Fn+F8 を押す
- 選択 設定から相対マウスを有効にし、有効にします。

Windows ロゴキーまたはコマンドキーを再マッピングする

キーボードで Mac オプションキーとコマンドキーを再マッピングできます。

両方のキーと一緒に使用すると、修飾キーによって別のキーの操作が変更されます。修飾キーを別のキーと一緒に使用して、プリントなどのタスクを実行できます。[Meta key (メタキー)] は、修飾キーの特殊なタイプです。両方のキーと一緒に使用すると、メタキーを使用して別のキーの機能を一時的に変更できます。

Mac オプションキーとコマンドキーを再マッピングするには、「設定」と「キーボードとマウス」を選択します。

ストリーミングセッション中に、オプションキーを次のキーに再マッピングできます。

- リモート Alt キー
- ローカル修飾キー

ストリーミングセッション中に、コマンドキーを次のキーに再マップできます。

- リモートコントロールキー
- メタキー

設定を記憶する

AppStream 2.0 macOS クライアントアプリケーションは、オーディオとビデオの設定を除き、将来のセッションの設定で設定した設定を保存できます。今後のセッションでオーディオ (マイク) とビデオ (ウェブカメラ) の設定を記憶する場合は、今後のセッションで設定、オーディオとビデオ、オーディオとビデオの記憶 (有効/無効) を選択します。この機能を有効にすると、オーディオ (マイク) とビデオ (ウェブカメラ) の設定が保存され、同じデバイス上の macOS クライアントから同じスタックにアクセスすると、セッション間で保持されます。

プリンターリダイレクト

AppStream 2.0 ローカルプリンターリダイレクトを使用すると、AppStream 2.0 ストリーミングセッションからローカルコンピュータに接続されているプリンターにアクセスできます。これにより、プリントジョブをストリーミングアプリケーションからローカルプリンター、またはマッピングしたネットワークプリンターにリダイレクトできます。

Important

AppStream 2.0 プリンターリダイレクトを使用するには、AppStream 2.0 クライアントをローカルコンピュータにインストールし、クライアントを使用してストリーミングセッションに接続する必要があります。ウェブブラウザを使用して AppStream 2.0 に接続する場合、ファイルのリダイレクトは使用できません。

プリントジョブをローカルプリンターにリダイレクトするには

1. AppStream 2.0 クライアントを開き、ストリーミングセッションに接続します。
2. ストリーミングアプリケーションで、[File (ファイル)]、[Print Now (今すぐプリント)] の順に選択します。
3. ストリーミングアプリケーションの [Print (プリント)] ダイアログボックスが開きます。
4. [Print (プリント)] ダイアログボックスに、使用可能なローカルプリンター一覧が表示されます。使用するローカルプリンターを選択し、プリントを続行します。

セッションの切断と終了

ストリーミングセッションを切断するには、次のいずれかのオプションを選択します。

- AppStream 2.0 ツールバーで、切断を選択します。
- Mac の画面上部のメニューバーで、Amazon AppStream 2.0 と切断を選択します。

タイムアウト時間間隔内の切断後に、以前のストリーミングセッションに再接続できます。切断後にストリーミングセッションがアクティブのままになる時間は、管理者によって設定されます。

現在のセッションを終了するには、次のいずれかのオプションを選択します。

- AppStream 2.0 ツールバーで、セッションの終了を選択します。

- Mac の画面上部のメニューバーで、Amazon AppStream 2.0 とセッションの終了を選択します。

セッションを終了すると、開いているドキュメントを保存するように求められ、ストリーミングインスタンスからすぐに切断されます。

トラブルシューティング

診断ログのアップロードを有効にし、クライアントバージョンとクライアント ID を確認するには、次の手順を実行します。

診断ログのアップロードを有効にする

AppStream 2.0 クライアントの問題をトラブルシューティングするには、診断ログを有効にできます。AppStream 2.0 に送信されるログファイルには、デバイスと AWS ネットワークへの接続に関する詳細情報が含まれています。AppStream 2.0 ストリーミングセッションの前または最中に診断ログのアップロードを有効にすると、これらのファイルは AppStream 2.0 に自動的に送信されます。ベストプラクティスとして、AppStream 2.0 チームが問題をトラブルシューティングできるように、ログのアップロードを有効にすることをお勧めします。

ファイルログ記録を有効にするには、次の手順に従います。

1. システムメニューバーから AppStream 2.0 を選択するか、Connect ページの右上隅に移動します。
2. クライアントオプションとクライアント自動ログ記録を選択します。

macOS 用 AppStream 2.0 クライアントのログを収集する

AppStream 2.0 ログは、管理者が設定の問題を特定してトラブルシューティングするために使用できます。また、AWS サポートによるケースの診断とトラブルシューティングにも役立ちます。ログを収集して共有するには、次のオプションから選択します。

- オプション 1: ターミナルを開き、「`」と入力します。 open ~/Library/Containers/com.amazon.appstreamclient/Data/logs`
- オプション 2: Finder を開き、ユーザー、User_Name、ライブラリ、コンテナ、Appstream、データ、ログを選択する
- オプション 3: Finder を開き、左上のシステムメニューバーから Go と を選択しますGo to folder。 **~/Library/Containers/com.amazon.appstreamclient/Data/logs** と入力します。

クライアントバージョンとクライアント ID を確認する

macOS 用 AppStream 2.0 クライアントの使用時に問題が発生した場合、AppStream 2.0 バージョン番号とクライアント ID は、管理者と AWS サポートチームのトラブルシューティングに役立ちます。インストールした AppStream 2.0 クライアントのバージョンを確認するには、AppStream 2.0 クライアントを開きます。システムメニューバーで、Amazon AppStream 2.0 と Amazon AppStream 2.0 についてを選択します。クライアントバージョンは、Amazon AppStream 2.0 ロゴの下に表示されます。

インストールした AppStream 2.0 クライアントのクライアント ID を検索するには、システムメニューバーで Amazon AppStream 2.0 を選択するか、Connect ページの右上隅に移動してクライアントオプションを選択します。

AppStream 2.0 macOS クライアントリリースノート

次の表は、macOS AppStream 2.0 クライアントのリリースバージョンで利用可能な最新の更新を示しています。

クライアントバージョン	リリース日	変更
1.1.0	06-02-2025	• アクセシビリティの修正と改善 • 動的アプリケーションプロバイダーのサポート • クライアントカスタマイズのサポート
1.0.1	03-25-2025	• バグ修正と機能向上 • SAML インスタンスの証明書ベースの認証とマルチスタックアクセスのサポート
1.0.0	12-19-2024	• 初回リリース

ファイルストレージのオプション

AppStream 2.0 管理者によって有効になっていれば、アプリケーションストリーミングセッション中にファイルとフォルダに、以下のストレージオプションのうち 1 つ以上を使用できます。

- [ホームフォルダ](#)
- [Google ドライブ](#)
- [OneDrive for Business](#)
- [カスタム共有フォルダ](#)

Note

現在、Google ドライブと OneDrive for Business は Linux ベースのストリーミングインスタンス向けにサポートされていません。

ホームフォルダを使用する

AppStream 2.0 管理者によってこのファイルストレージオプションが有効になっていれば、AppStream 2.0 ストリーミングセッションにサインインすると、ホームフォルダを使用できます。ホームフォルダで以下のことを実行できます。

- ホームフォルダに保存するファイルとフォルダを開き、編集します。ユーザーのホームフォルダに保存されているコンテンツに他のユーザーがアクセスすることはできません。
- ローカルコンピュータとホームフォルダの間でファイルをアップロードおよびダウンロードします。AppStream 2.0 は最近変更されたファイルとフォルダを継続的に確認し、ホームフォルダにバックアップします。
- アプリケーションで作業している場合、ホームフォルダに保存されているファイルやフォルダにアクセスできます。アプリケーションで [File Open (ファイルを開く)] を選択し、開くファイルまたはフォルダを参照します。ファイルの変更をホームフォルダに保存するには、アプリケーションインターフェイスから [File Save (ファイルの保存)] を選択し、ホームフォルダでファイルを保存する場所を参照します。
- ウェブビューセッションツールバーから [My Files] を選択して、ホームフォルダにアクセスすることもできます。

⚠ Warning

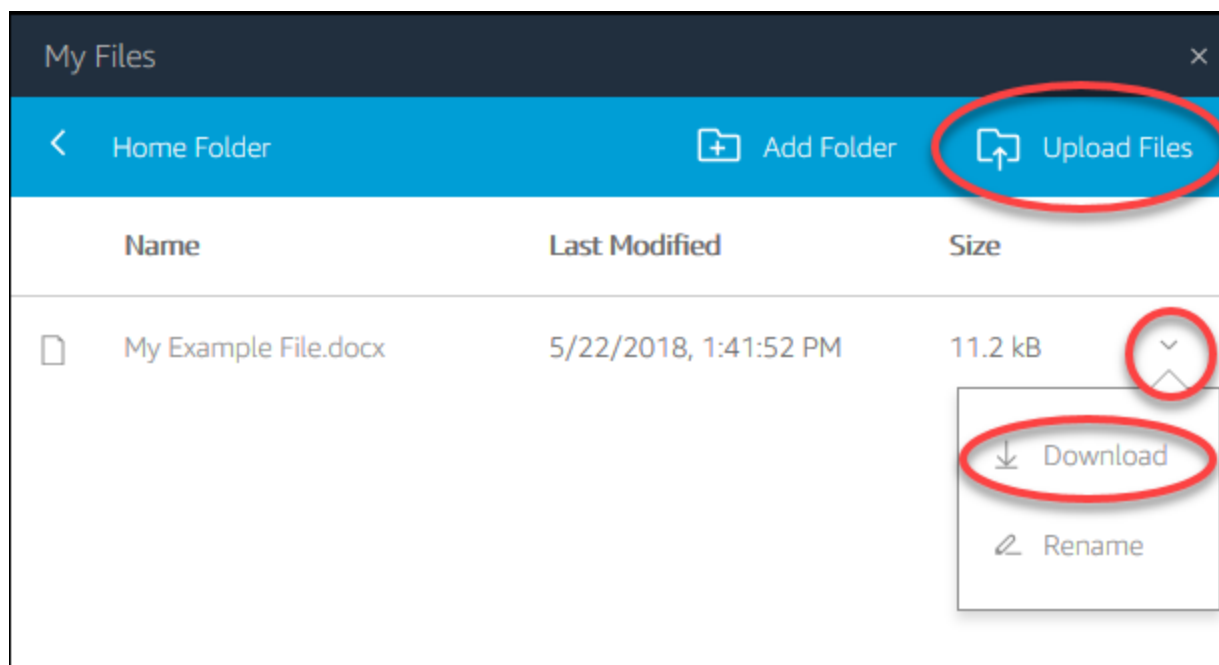
5 GB を超えるファイルは、AppStream 2.0 セッション間で保持されません。

i Note

ホームフォルダが表示されない場合は、ファイルエクスプローラの C:\Users\PhotonUser\My Files\Home Folder を参照することで、ホームフォルダのファイルを表示できます。

ローカルコンピュータとホームフォルダの間でファイルをアップロードおよびダウンロードするには

1. AppStream 2.0 ツールバーの左上で、[My Files (マイファイル)] アイコンを選択します。
2. 既存のフォルダに移動するか、[Add Folder] を選択してフォルダを作成します。
3. 対象のフォルダが表示されたら、次のいずれかを実行します。
 - ファイルをフォルダにアップロードするには、アップロードするファイルを選択し、[Upload] を選択します。
 - ファイルをフォルダからダウンロードするには、ダウンロードするファイルを選択し、ファイルの右にある下矢印を選択してから、[Download] を選択します。



Google ドライブを使用する

Note

Google API から受け取った情報の Amazon AppStream 2.0 による 使用および他のアプリケーションへの転送には、限定使用要件を含む [Google API サービスのユーザーデータポリシー](#)が適用されます。

AppStream 2.0 管理者によってこのファイルストレージオプションが有効になっていれば、Google Drive アカウントを AppStream 2.0 に追加できます。アカウントを追加して AppStream 2.0 ストリーミングセッションにサインインした後、Google Drive で以下のことを実行できます。

Note

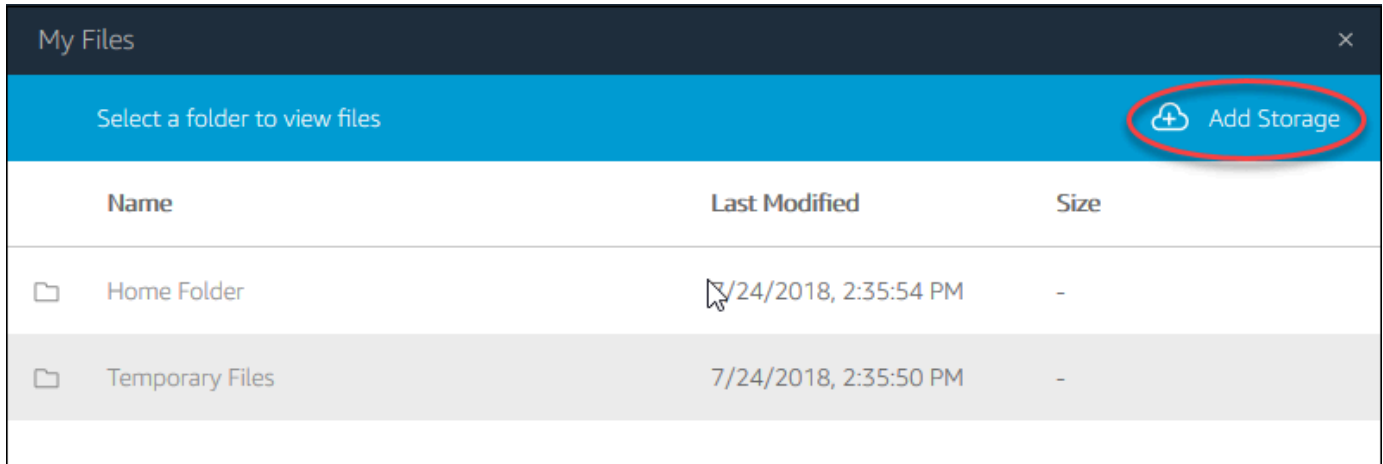
現在、Google ドライブは Linux ベースのストリーミングインスタンス向けにサポートされていません。

- Google ドライブに保存するファイルとフォルダを開き、編集します。共有することを選択しない限り、他のユーザーはアクセスできません。
- ローカルコンピュータと Google ドライブの間でファイルをアップロードおよびダウンロードします。ストリーミングセッション中に、Google ドライブのファイルやフォルダに行われた変更は、自動的にバックアップ、同期されます。その結果、ストリーミングセッション外部で、Google アカウントにサインインして、Google ドライブにアクセスしたユーザーが利用できるようになります。
- アプリケーションで作業している場合、Google ドライブに保存されているファイルやフォルダにアクセスできます。アプリケーションのインターフェイスから [File]、[Open] の順に選択し、開くファイルまたはフォルダを参照します。ファイルの変更を Google ドライブに保存するには、アプリケーションから [File]、[Save] を選択し、Google ドライブでファイルを保存する場所を参照します。
- AppStream 2.0 ツールバーの左上から [My Files (マイファイル)] を選択して、Google Drive にアクセスすることもできます。

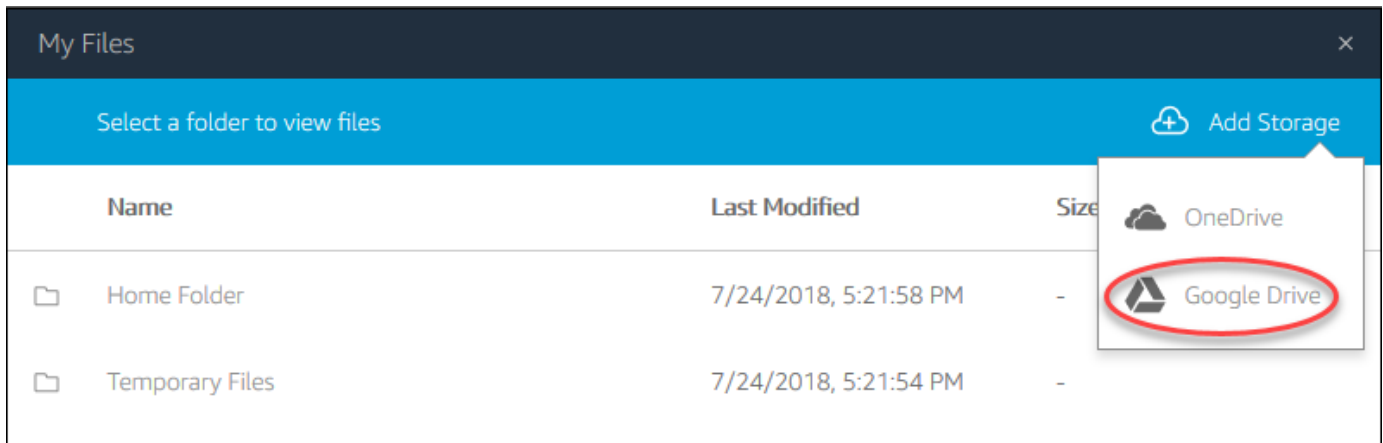
Google ドライブアカウントを AppStream 2.0 に追加するには

AppStream 2.0 ストリーミングセッション中に Google ドライブにアクセスするには、まず Google ドライブアカウントを AppStream 2.0 に追加する必要があります。

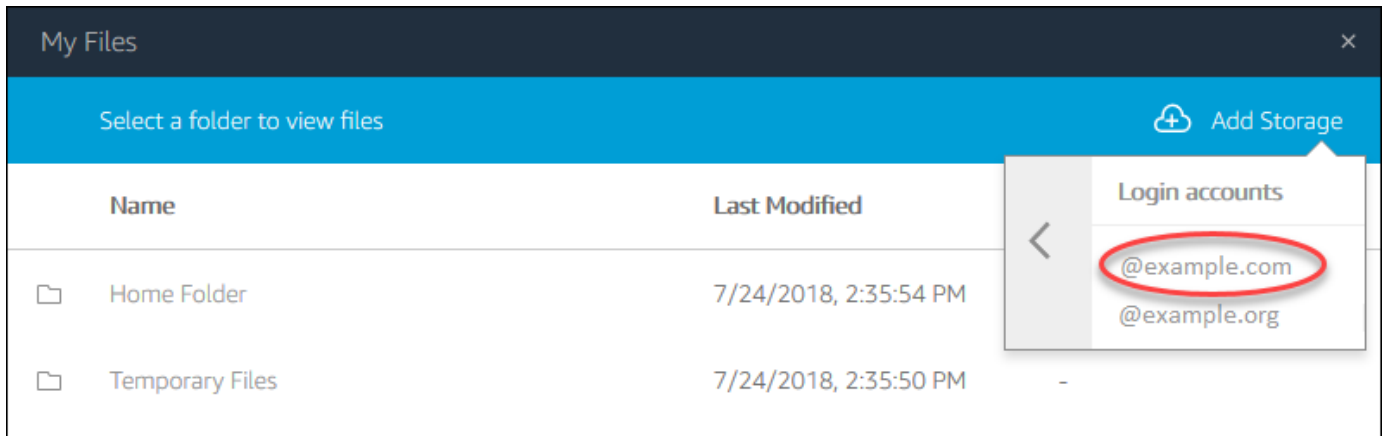
1. AppStream 2.0 ツールバーの左上で、[My Files (マイファイル)] アイコンを選択します。
2. [My Files] ダイアログボックスで、[Add Storage] を選択します。



3. [Google Drive] を選択します。

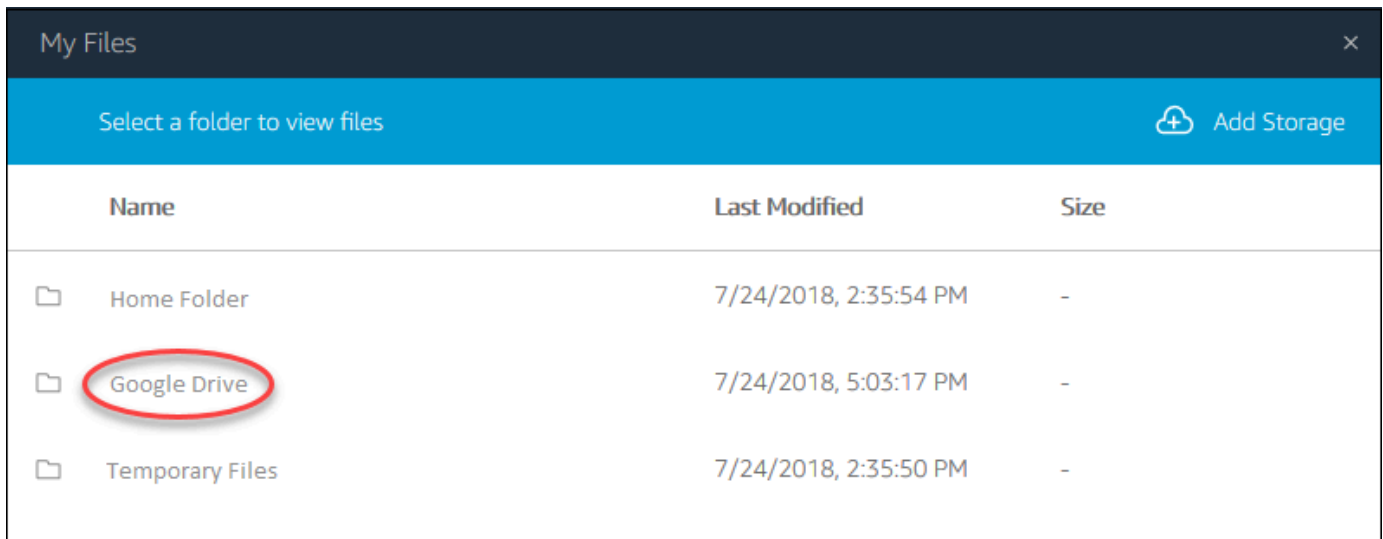


4. Google ドライブアカウントのドメインを選択します。



5. [Sign in with Google] ダイアログボックスが表示されます。プロンプトが表示されたら、Google ドライブアカウントのサインイン認証情報を入力します。

Google ドライブアカウントが AppStream 2.0 に追加されると、Google ドライブフォルダが [My Files (マイファイル)] に表示されます。



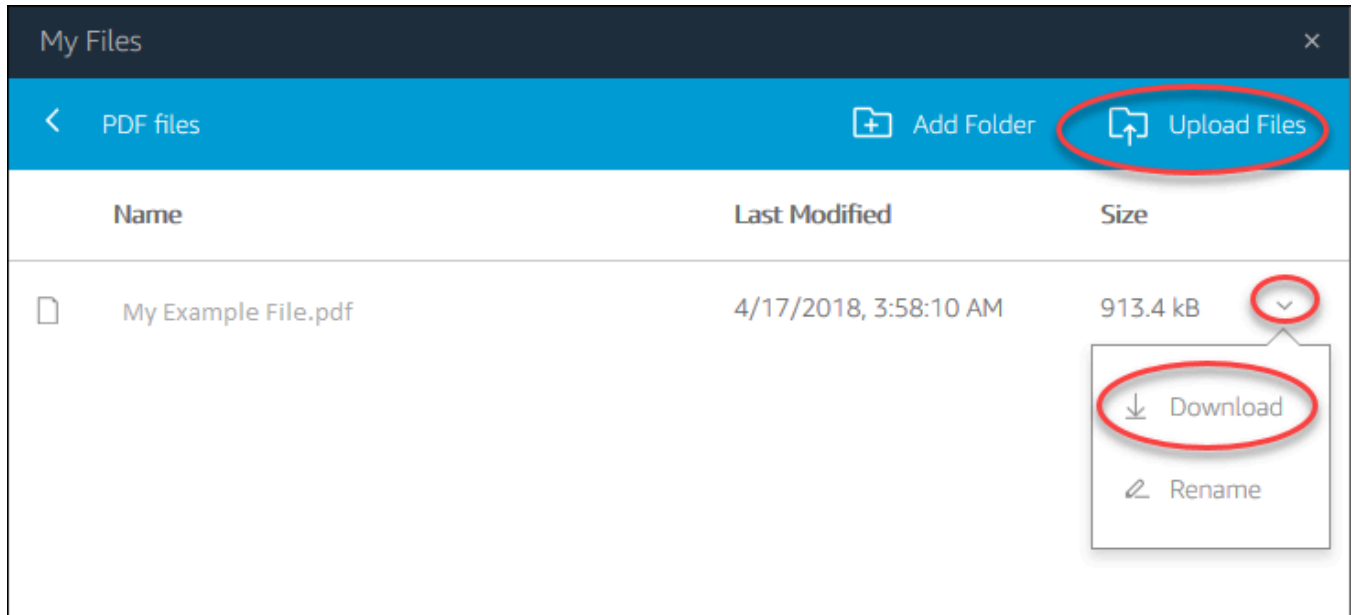
6. Google ドライブのファイルとフォルダを操作するには、[Google Drive (Google ドライブ)] フォルダを選択し、任意のファイルやフォルダを参照します。ストリーミングセッション中に Google ドライブのファイルで作業しない場合は、[My Files] ダイアログボックスを閉じます。

ローカルコンピュータと Google ドライブの間でファイルをアップロードおよびダウンロードするには

1. AppStream 2.0 ツールバーの左上で、[My Files (マイファイル)] アイコンを選択します。
2. [My Files] ダイアログボックスで、[Google Drive] を選択します。
3. 既存のフォルダに移動するか、[Add Folder] を選択してフォルダを作成します。

4. 対象のフォルダが表示されたら、次のいずれかを実行します。

- ファイルをフォルダにアップロードするには、アップロードするファイルを選択し、[Upload] を選択します。
- ファイルをフォルダからダウンロードするには、ダウンロードするファイルを選択し、ファイルの右にある下矢印を選択してから、[Download] を選択します。



OneDrive for Business を使用する

Note

現在、OneDrive for Business は Linux ベースのストリーミングインスタンス向けにサポートされていません。

AppStream 2.0 管理者によってこのファイルストレージオプションが有効になっていれば、OneDrive アカウントを AppStream 2.0 に追加できます。アカウントを追加して AppStream 2.0 ストリーミングセッションにサインインした後、OneDrive で以下のことを実行できます。

- OneDrive に保存するファイルとフォルダを開き、編集します。共有することを選択しない限り、他のユーザーはアクセスできません。
- ローカルコンピュータと OneDrive の間でファイルをアップロードおよびダウンロードします。ストリーミングセッション中に、OneDrive のファイルやフォルダに行った変更は、自動的にバック

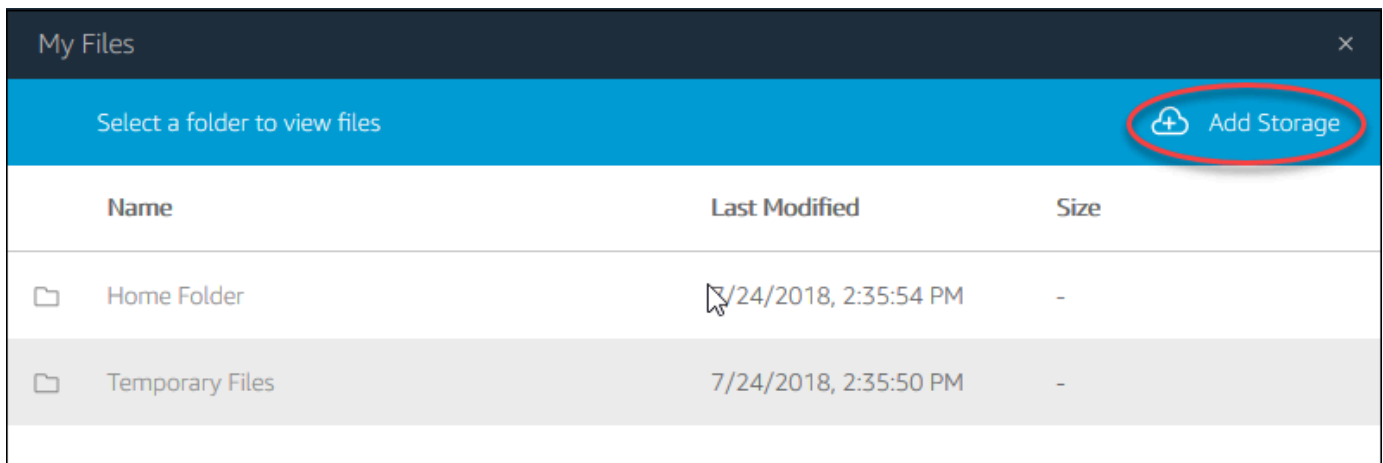
アップおよび同期されます。その結果、ストリーミングセッション外部で、OneDrive にサインインして、OneDrive にアクセスしたユーザーが利用できるようになります。

- アプリケーションで作業している場合、OneDrive に保存されているファイルやフォルダにアクセスできます。アプリケーションのインターフェイスから [File]、[Open] の順に選択し、開くファイルまたはフォルダを参照します。ファイルの変更を OneDrive に保存するには、アプリケーションから [File]、[Save] を選択し、OneDrive でファイルを保存する場所を参照します。
- AppStream 2.0 ツールバーの左上から [My Files (マイファイル)] を選択して、OneDrive にアクセスすることもできます。

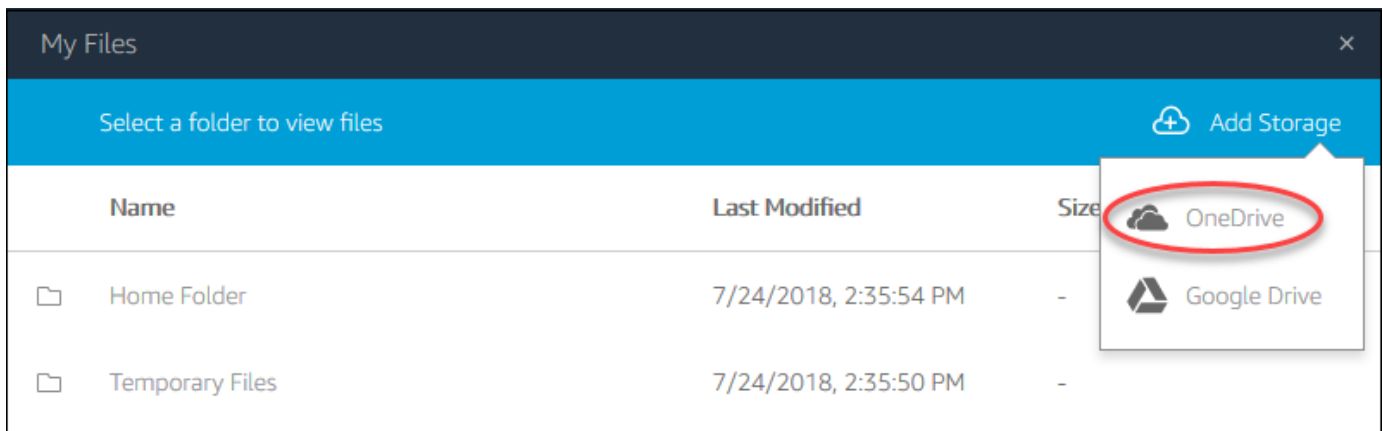
OneDrive アカウントを AppStream 2.0 に追加するには

AppStream 2.0 ストリーミングセッション中に OneDrive にアクセスするには、まず OneDrive アカウントを AppStream 2.0 に追加する必要があります。

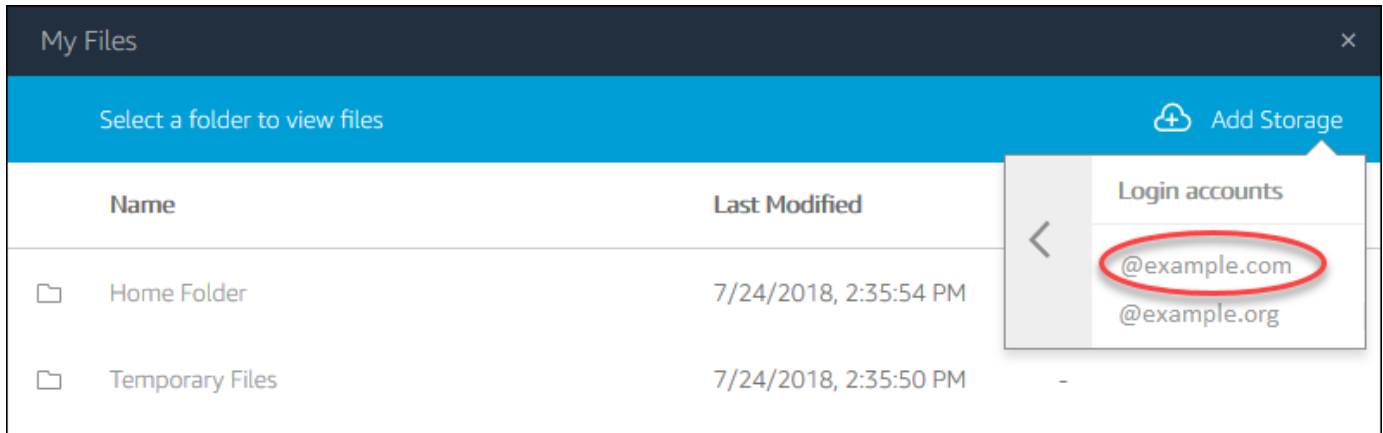
1. AppStream 2.0 ツールバーの左上で、[My Files (マイファイル)] アイコンを選択します。
2. [My Files] ダイアログボックスで、[Add Storage] を選択します。



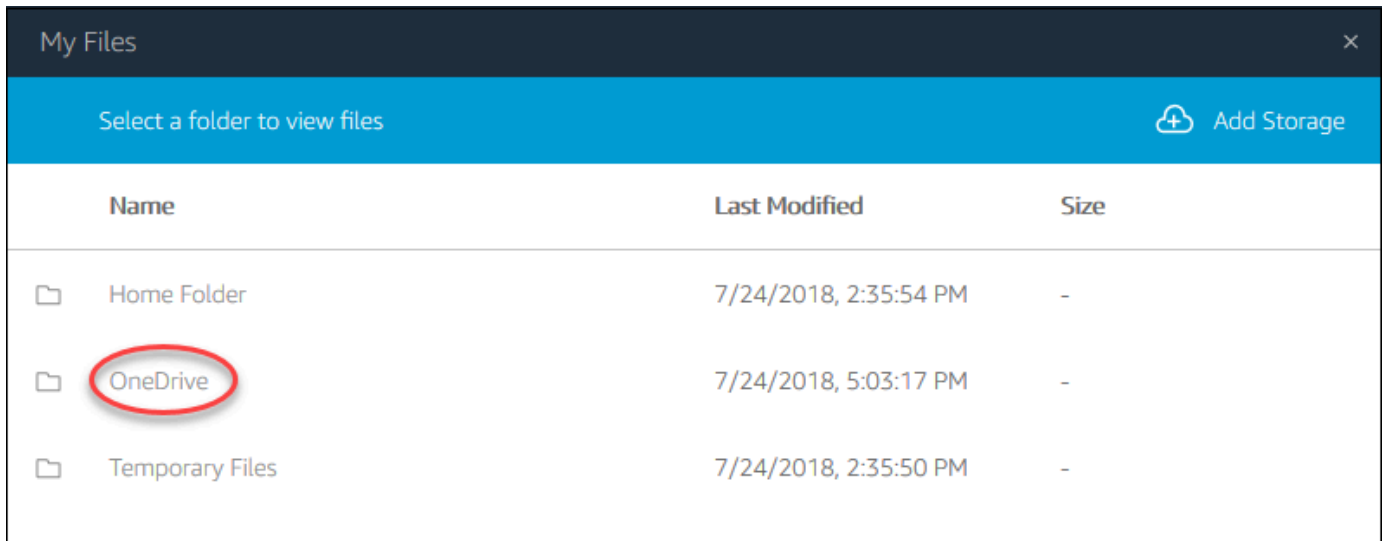
3. [OneDrive] を選択します。



4. [Login accounts] で、OneDrive アカountのドメインを選択します。



5. [サインイン] ダイアログボックスに、アカウントのサインイン認証情報を入力します。
6. OneDrive アカountが AppStream 2.0 に追加されると、OneDrive フォルダが [My Files (マイファイル)] に表示されます。

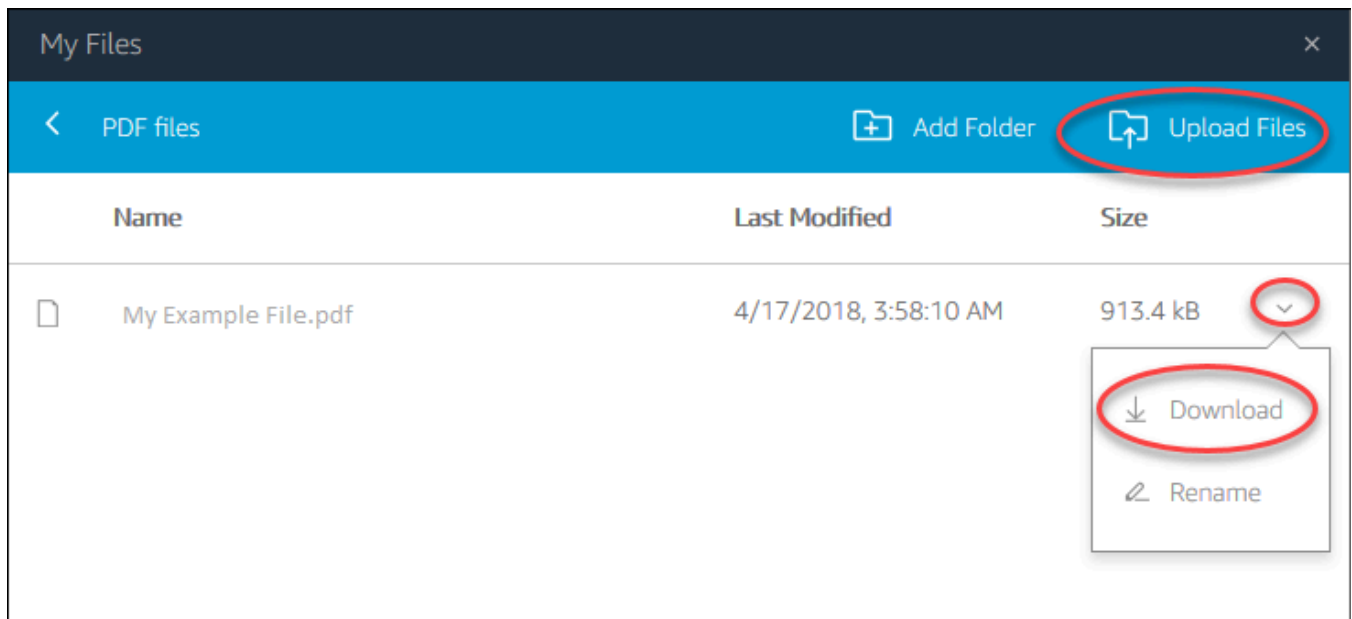


7. OneDrive のファイルとフォルダを操作するには、[OneDrive] フォルダを選択し、任意のファイルやフォルダを参照します。ストリーミングセッション中に OneDrive のファイルで作業しない場合は、[My Files] ダイアログボックスを閉じます。

ローカルコンピュータと OneDrive の間でファイルをアップロードおよびダウンロードするには

1. AppStream 2.0 ツールバーの左上で、[My Files (マイファイル)] アイコンを選択します。
2. [My Files] ダイアログボックスで、[OneDrive] を選択します。
3. 既存のフォルダに移動するか、[Add Folder] を選択してフォルダを作成します。
4. フォルダが表示されたら、次のいずれかの操作を行います。

- ファイルをフォルダにアップロードするには、アップロードするファイルを選択し、[Upload] を選択します。
- ファイルをフォルダからダウンロードするには、ダウンロードするファイルを選択し、ファイルの右にある下矢印を選択してから、[Download] を選択します。



AppStream 2.0 から OneDrive アクセス許可を削除するには

AppStream 2.0 ストリーミングセッション中に OneDrive が不要になった場合は、これらのステップを行い、AppStream 2.0 から OneDrive アクセス許可を削除します。

Note

これらのアクセス許可は、AppStream 2.0 ストリーミングセッション中にいつでも復元することができます。

1. Office 365 または OneDrive for Business にサインインします。
2. 右ペインの [My accounts] で、[My account] を選択します。
3. アカウントのダッシュボードページの [App permissions] で、[Change app permissions] を選択します。
4. [App permissions (アプリのアクセス許可)] ページの [Amazon AppStream 2.0] で、[Revoke (取り消し)] を選択します。

カスタム共有ネットワークフォルダの使用

AppStream 2.0 管理者がこのファイルストレージオプションを有効にしている場合、ストリーミングセッションにサインインすると、管理者にカスタム共有フォルダが設定され、名前が付けられます。共有フォルダの名前については、管理者にお問い合わせください。複数のユーザーが共有カスタムフォルダにアクセスしてコラボレーションできます。

カスタム共有フォルダでは、以下のことを実行できます。

- OneDrive に保存するファイルとフォルダを開き、編集します。共有することを選択しない限り、他のユーザーはアクセスできません。
- カスタム共有フォルダに保存するファイルとフォルダを開き、編集します。そこに保存されているコンテンツには、管理者が設定したアクセス許可に応じて、他のユーザーがアクセスできます。
- ローカルコンピュータとカスタム共有フォルダの間でファイルをアップロードおよびダウンロードします。
- アプリケーションで作業している場合、カスタム共有フォルダに保存されているファイルやフォルダにアクセスできます。

ローカルコンピュータとカスタム共有フォルダの間でファイルをアップロードおよびダウンロードするには

1. AppStream 2.0 ツールバーの左上で、[My Files (マイファイル)] アイコンを選択します。
2. 共有カスタムフォルダに移動します。フォルダ名については、管理者にお問い合わせください。
3. 共有カスタムフォルダを選択します。
 - ファイルをフォルダにアップロードするには、アップロードするファイルを選択し、[Upload] を選択します。
 - ファイルをフォルダからダウンロードするには、ダウンロードするファイルを選択し、ファイルの右にある下矢印を選択してから、[Download] を選択します。

地域の設定を指定する

AppStream 2.0 Windows ストリーミングセッションで、使用する場所または言語に固有の設定を使用するように、リージョン設定を指定できます。ストリーミングセッション中に行った変更は、以降のストリーミングセッションに適用されます。

Windows AppStream 2.0 ストリーミングセッションのリージョン設定を指定するには

1. AppStream 2.0 ツールバーの左上で [Settings (設定)] アイコンを選択してから、[Regional settings (リージョン設定)] を選択します。
2. [Regional settings] ダイアログボックスで、必要に応じて以下のオプションを設定します。完了したら、[Save] を選択します。
 - Time zone (タイムゾーン) — オペレーティングシステムの時刻に依存する Windows および任意のアプリケーションで使用されるシステム時間を決定します。以下のオプションのいずれかを選択してください：
 - ストリーミングセッションのタイムゾーンをデバイスに設定されているタイムゾーンと同期するには、[デバイスに基づいて自動的にタイムゾーンを設定する] を選択します。これは、シングルセッションフリートとマルチセッションフリートの両方でデフォルトで有効になっており、無効にすることができます。
 - シングルセッションフリートの場合、自動リダイレクトを使用する代わりに、ストリーミングセッションの特定のタイムゾーンを選択できます。カスタムタイムゾーンを設定するには、[リージョン設定] の [デバイスオプションに基づいてタイムゾーンを自動的に設定] を無効にし、使用可能なリストから任意のタイムゾーンを選択します。
 - Locale (国) (カルチャ) — 数値、通貨、時刻、および日付の Windows での表示方法を決定します。AppStream 2.0 では口ケールとして、中国語 (簡体字および繁体字)、オランダ語、英語、フランス語、ドイツ語、イタリア語、日本語、韓国語、ポルトガル語、スペイン語、タイ語をサポートします。
 - Input method (入力方法) — 別の言語で文字を入力するときに使用できるキーストロークの組み合わせを決定します。

リージョン設定の指定は、Linux ストリーミングセッションではまだサポートされていません。ただし、管理者が指定したショートカットキーの組み合わせを使用して、ストリーミングセッションで利用できるさまざまな入力方法を切り替えることができます。デフォルトのショートカットキーの組み合わせは、Super + Space と Shift + Super + Space です。Super は、Windows キーボードの Windows キー、または Apple キーボードの Command キーです。イメージの作成時に指定したショートカットキーについては、必ず管理者に確認してください。例えば、「[Linux イメージの日本語サポートを有効にする](#)」では、ショートカットキーの組み合わせが Control + Space と Shift + Control + Space に変更されています。

Amazon AppStream 2.0 用拡張機能 SDK デベロッパーガイド

Amazon AppStream 2.0 は、Amazon DCV テクノロジーを使用して、アプリケーションに安全で高性能なアクセスを提供します。Amazon DCV 拡張機能 SDK を使用すると、開発者は、以下のアクションを含め、エンドユーザー向けの AppStream 2.0 のエクスペリエンスをカスタマイズできます。

- カスタムハードウェアのサポートを容易にします。
- リモートセッションでのサードパーティアプリケーションの使いやすさを高めます。例えば、VoIP アプリケーションにローカルオーディオの削除を追加したり、会議アプリケーションにローカルビデオ再生を追加したりできます。
- スクリーンリーダーなどのアクセシビリティソフトウェアにリモートセッションやリモートで実行されているアプリケーションに関する情報を提供します。
- セキュリティソフトウェアに対して、ローカルエンドポイントのセキュリティ体制を分析して条件付きアクセスポリシーを許可します。
- 確立されたリモートセッションで任意のデータ転送を実行します。

Amazon DCV 拡張機能 SDK を使い始めるには、「[What is the Amazon DCV Extension SDK?](#)」を参照してください。SDK 自体は「[Amazon DCV Extension SDK Github repository](#)」にあります。また、SDK の統合例は、「[Amazon DCV Extension SDK Samples Github repository](#)」にあります。

トピック

- [Amazon AppStream 2.0 の拡張機能 SDK の前提条件](#)
- [Amazon AppStream 2.0 のサードパーティーベンダー拡張機能](#)

Amazon AppStream 2.0 の拡張機能 SDK の前提条件

Amazon DCV 拡張機能 SDK を使い始める前に、AppStream 2.0 クライアントアプリケーションと AppStream 2.0 サーバーが次の要件を満たしていることを確認してください。

サポートされている AppStream 2.0 クライアント:

- AppStream 2.0 Windows クライアントバージョン 1.1.1154 以降

 Note

AppStream 2.0 ウェブアクセスは Amazon DCV 拡張機能 SDK をサポートしていません。

サポートされている AppStream 2.0 ストリーミングインスタンス:

- Windows Server 2016 および 2019
- 2023 年 5 月 8 日以降にリリースされた Windows に対応する AppStream 2.0 エージェントのバージョン
- 2023 年 5 月 8 日以降にリリースされたマネージド AppStream 2.0 イメージ更新

Amazon AppStream 2.0 のサードパーティーベンダー拡張機能

AWS は、AppStream 2.0 ホストおよびクライアントプロセス内で Amazon DCV 拡張機能 SDK API をサポートします。ただし、サードパーティーの独立系ソフトウェアベンダー (ISV) によって開発された拡張機能は、AWSによって開発または維持されていないことに注意してください。したがって、インストール、設定、トラブルシューティング、更新を含む、拡張機能自体のサポートは、その拡張機能を開発したサードパーティーベンダーの責任となります。サードパーティーの拡張機能に関する問題や質問がある場合は、関連するサードパーティーベンダーにサポートを依頼してください。

Amazon AppStream 2.0 のドキュメント履歴

- API バージョン: 2016-12-01

次の表に、AppStream 2.0 サービス ([AppStream 2.0 ベースイメージ](#)、[AppStream 2.0 エージェント](#)、および [AppStream 2.0 クライアント](#) リリースを含む)、ならびに 2018 年 6 月 4 日以降の Amazon AppStream 2.0 管理ガイドのドキュメントに対する重要な追加事項について示します。また、お客様からいただいたフィードバックに対応するために、ドキュメントを頻繁に更新しています。

これらの更新に関する通知については、Amazon AppStream 2.0 RSS フィードにサブスクライブできます。

変更	説明	日付
最後にリリースされたイメージ	リリースされた最新のイメージを更新しました	2025 年 6 月 18 日
macOS クライアントの新しいバージョン	クライアントバージョン 1.1.0	2025 年 6 月 2 日
新しいクライアントバージョン	新しいクライアントバージョン 1.1.1437	2025 年 5 月 21 日
新しい許可されたドメイン	欧州 (パリ) の新しい許可されたドメイン	2025 年 5 月 6 日
新しいクライアントバージョン	新しいクライアントバージョン 1.1.1423	2025 年 3 月 31 日
macOS クライアントの新しいバージョン	クライアントバージョン 1.0.1	2025 年 3 月 26 日
グループポリシーの設定	追加のグループポリシー設定を追加	2025 年 2 月 20 日
新しいクライアントバージョン	新しいクライアントバージョン 1.1.1414	2025 年 1 月 16 日

新しいクライアントバージョン	新しいクライアントバージョン 1.1.1408	2024 年 12 月 20 日
Rocky Linux 8 のサポート	AppStream 2.0 が Rocky Linux 8 オペレーティングシステムをサポート	2024 年 12 月 19 日
macOS 用のクライアントアプリケーション	macOS 用の AppStream 2.0 クライアントを使用して AppStream 2.0 に接続し、アプリケーションをストリーミングする	2024 年 12 月 19 日
新しいクライアントバージョン	新しいクライアントバージョン 1.1.1403	2024 年 12 月 13 日
新しいエージェントのバージョン	AppStream 2.0 Agent の 10-31-2024 バージョン	2024 年 11 月 15 日
新しいエージェントのバージョン	AppStream 2.0 Agent の 10-21-2024 バージョン	2024 年 10 月 30 日
ウェブブラウザアクセスバージョン 2	ウェブブラウザアクセス v2 がデフォルトのエクスペリエンスになりました。	2024 年 10 月 4 日
デバイスに基づいてタイムゾーンを自動的に設定する	タイムゾーンを同期して、デバイスに設定されているタイムゾーンと一致させます。	2024 年 10 月 2 日
ホームフォルダのヘビーファイル同期モード	組織の Amazon Simple Storage Service ホームフォルダオプションを有効にする	2024 年 10 月 1 日
アプリケーション設定の更新	アプリケーション設定のさまざまな更新	2024 年 9 月 30 日
Cookie ベースの認証	Cookie の盗難を防ぐための事前対策	2024 年 9 月 25 日

ブランディングの更新	Amazon S3 バケットから組織ロゴまたはファビコンを選択する	2024 年 9 月 23 日
新しいクライアントバージョン	新しいクライアントバージョン 1.1.1360	2024 年 8 月 1 日
Red Hat Enterprise Linux のベースイメージ	Red Hat Enterprise Linux のベースイメージを追加	2024 年 7 月 30 日
新しいクライアントバージョン	新しいクライアントバージョン 1.1.1332	2024 年 7 月 3 日
最新の Windows ベースイメージ	2024 年 6 月 17 日にリリースされた Windows ベースイメージ	2024 年 7 月 2 日
新しいクライアントバージョン	新しいクライアントバージョン 1.1.1326	2024 年 6 月 17 日
新しいエージェントのバージョン	AppStream 2.0 Agent 向け 05-21-2024 バージョン	2024 年 5 月 30 日
マルチセッションのユースケース	マルチセッションのユースケースにオーディオ会議を追加	2024 年 5 月 30 日
最新のベースイメージ	2024 年 5 月 8 日にリリースされたベースイメージ	2024 年 5 月 30 日
AppStream 2.0 イメージのマネージド型更新機能	2024 年 4 月 25 日以降にリリースされたマネージド AppStream 2.0 イメージ更新	2024 年 5 月 15 日
新しいエージェントのバージョン	AppStream 2.0 Agent 向け 04-15-2024 バージョン	2024 年 4 月 26 日
最新のベースイメージ	2024 年 3 月 24 日にリリースされたベースイメージ	2024 年 4 月 26 日

クロスアカウント PCA 共有	集中型 CA を使用するためのアクセス許可を他のアカウントに付与する	2024 年 4 月 25 日
新しいクライアントバージョン	クライアントバージョン 1.1.1303	2024 年 4 月 4 日
新しいリリースステートリジョンエンドポイント	Windows クライアントアプリケーションバージョン 1.1.1300 以降のリリースステートエンドポイント	2024 年 4 月 1 日
新しいクライアントバージョン	クライアントバージョン 1.1.1300	2024 年 3 月 28 日
最新のベースイメージ	2024 年 1 月 26 日にリリースされたベースイメージ	2024 年 2 月 16 日
マルチセッションフリートでセッションスクリプトを使用する	マルチセッションフリートでセッションスクリプトを使用する場合、最適なパフォーマンスとセキュリティを確保するための追加の要件と考慮事項があります。	2024 年 2 月 15 日
新しいエージェントのバージョン	AppStream 2.0 Agent 向け 01-17-2024 バージョン	2024 年 2 月 15 日
クリップボードの更新	[ローカルデバイス文字数制限にコピー] または [リモートセッション文字数制限に貼り付け] を選択するか、またはその両方を選択して、クリップボードを使用する際にユーザーがコピーまたは貼り付けることができるデータの量を制限できます。	2024 年 2 月 15 日

新しいクライアントバージョン	クライアントバージョン 1.1.1259	2024 年 2 月 8 日
新しいクライアントバージョン	クライアントバージョン 1.1.1246	2024 年 1 月 18 日
Windows Server 2022 のサポート	Windows Server 2022 Base のサポートを追加しました。	2023 年 12 月 14 日
Web ブラウザアクセスバージョン 2	AppStream 2.0 ウェブブラウザアクセスバージョン 2 では、簡単に見つけられるメニューオプションやテキストによるエンドユーザー向けガイダンスなど、エンドユーザーエクスペリエンスが強化されています。	2023 年 12 月 11 日
新しいクライアントバージョン	クライアントバージョン 1.1.1228	2023 年 11 月 1 日
マルチセッションフリート	マルチセッションフリートでは、1 つのフリートインスタンスで複数のユーザーセッションをプロビジョニングできます。	2023 年 10 月 26 日
新しいエージェントのバージョン	AppStream 2.0 Agent 向け 08-22-2023 バージョン	2023 年 8 月 25 日
新しいインスタンスファミリー	新しい Graphics G5 インスタンスファミリー	2023 年 7 月 26 日
新しいエージェントのバージョン	AppStream 2.0 Agent 向け 06-11-2023 バージョン	2023 年 7 月 25 日

App Block Builder	App Block Builder は、アプリケーション (または App Block) のパッケージ化に使用できる再利用可能なリソースです	2023 年 6 月 29 日
新しいクライアントバージョン	クライアントバージョン 1.1.1183	2023 年 6 月 26 日
新しいエージェントのバージョン	AppStream 2.0 Agent 向け 05-30-2023 バージョン	2023 年 6 月 15 日
拡張機能 SDK デベロッパーガイド	Amazon AppStream 2.0 は、Amazon DCV テクノロジーを使用して、アプリケーションに安全で高性能なアクセスを提供します。	2023 年 5 月 26 日
新しいエージェントのバージョン	AppStream 2.0 Agent 向け 05-08-2023 バージョン	2023 年 5 月 12 日
新しいクライアントバージョン	クライアントバージョン 1.1.1159	2023 年 5 月 9 日
新しいエージェントのバージョン	AppStream 2.0 Agent 向け 04-13-2023 バージョン	2023 年 4 月 25 日
最新のベースイメージ	2023 年 3 月 15 日にリリースされた最新の Linux ベースイメージ	2023 年 4 月 5 日
ドキュメントの更新: 新しいリージョン — AWS GovCloud (米国東部)	AWS GovCloud (US) ユーザーガイドの「 Amazon AppStream 2.0 」を更新し、「SAML のセットアップ」内のリレーステートエンドポイントの一覧表および必要に応じて他のコンテンツを更新しました。	2023 年 4 月 5 日

新しいエージェントのバージョン	AppStream 2.0 Agent 向け 03-21-2023 バージョン	2023 年 4 月 3 日
新しいリージョンのサポート	南米 (サンパウロ) がサポート されました	2022 年 12 月 15 日
新しいクライアントバージョン	クライアントバージョン 1.1.1118	2022 年 11 月 7 日
クライアント機能表	さまざまなアクセスタイプで サポートされている機能を比 較した表を追加しました。	2022 年 11 月 7 日
証明書ベースの認証	Microsoft Active Directory に結 合している AppStream 2.0 フ リートでは、証明書ベースの 認証を使用できます。	2022 年 10 月 31 日
最新のベースイメージ	2022 年 10 月 5 日にリリース された最新の Linux ベースイ メージ	2022 年 10 月 27 日
新しいエージェントのバージョン	エージェントのバージョン 10-13-2022	2022 年 10 月 24 日
新しいクライアントバージョン	クライアントバージョン 1.1.1099	2022 年 10 月 13 日
Linux のウェブカメラのサポ ート	Linux ベースのイメージに対し てウェブカメラを有効または 無効にする	2022 年 10 月 5 日
最新のベースイメージ	2022 年 9 月 21 日にリリース された最新の Linux ベースイ メージ	2022 年 10 月 3 日
新しいクライアントバージョン	クライアントバージョン 1.1.1066	2022 年 8 月 17 日

最新のベースイメージ	2022 年 7 月 12 日にリリースされた最新の Windows ベースイメージ	2022 年 7 月 21 日
新しいエージェントのバージョン	AppStream 2.0 Agent 向け 06-20-2022 バージョン	2022 年 6 月 30 日
新しいクライアントバージョン	クライアントバージョン 1.1.421	2022 年 1 月 29 日
新しいリージョンのサポート	米国東部 (オハイオ) がサポートされるようになりました	2022 年 6 月 28 日
新しいデフォルトのクォータ	Elastic フリートの最大同時セッション数	2022 年 5 月 31 日
新しいクライアントバージョン	クライアントのバージョン 1.1.414	2022 年 4 月 27 日
日本語サポート	Linux イメージの日本語サポートを有効にする	2022 年 4 月 19 日
Elastic フリートのセッションスクリプト	Elastic フリートのセッションスクリプトを設定および指定するには	2022 年 4 月 14 日
カナダ (中部) のサポート	カナダ (中部) リージョンがサポートされました	2022 年 3 月 31 日
新しいエージェントのバージョン	AppStream 2.0 Agent 向け 03-14-2022 バージョン	2022 年 3 月 25 日
新しいエージェントのバージョン	エージェントのバージョン 03-14-2022	2022 年 3 月 19 日
最新のベースイメージ	2022 年 3 月 3 日にリリースされた最新の Windows ベースイメージ	2022 年 3 月 14 日

マネージドイメージの更新	ベースイメージ 02-18-2022 をリリース	2022 年 3 月 3 日
新しいクライアントバージョン	クライアントのバージョン 1.1.398	2022 年 2 月 23 日
新しいクライアントバージョン	クライアントのバージョン 1.1.394	2022 年 2 月 8 日
マネージドイメージの更新	AppStream 2.0 エージェントのバージョン 12-20-2021	2022 年 1 月 6 日
属性ベースのアプリケーションの使用権限	アプリケーション使用権限は、AppStream 2.0 スタック内の特定のアプリケーションへのアクセスを制御します。	2022 年 1 月 5 日
App Block とアプリケーション	Elastic フリートの使用時に App Block とアプリケーションを作成できます。	2021 年 11 月 19 日
Linux イメージを作成する	Linux ベースの Amazon AppStream 2.0 イメージを作成できるようになりました。	2021 年 11 月 15 日
新しいエージェントのバージョン	エージェントバージョン 10-19-2021。	2021 年 10 月 26 日
マネージドイメージとベースイメージの更新	各種更新。	2021 年 10 月 21 日
新しいクライアントバージョン	クライアントのバージョン 1.1.333	2021 年 9 月 14 日
マネージドイメージの更新	2021 年 8 月 12 日のマネージドイメージを更新しました。	2021 年 8 月 23 日
新しいエージェントのバージョン	エージェントのバージョン 08-02-2021。	2021 年 8 月 18 日

新しいクライアントバージョン	クライアントのバージョン 1.1.304	2021 年 8 月 2 日
AppStream 2.0 クライアント更新プログラム: バージョン 1.1.304	組み込みの Chromium ブラウザをバージョン 91 にアップグレードします	2021 年 8 月 2 日
AppStream 2.0 エージェントの更新	複数の問題点を解決	2021 年 8 月 2 日
ベースイメージの更新	ベースイメージの更新 (2021 年 7 月 19 日)	2021 年 7 月 23 日
AppStream 2.0 エージェントの更新	複数の問題点を解決	2021 年 7 月 1 日
ベースイメージの更新	ベースイメージの更新 (06-01-2021)	2021 年 6 月 10 日
Graphics Design インスタンスへの AMD ドライバーのインストール	Graphics Design インスタンスを使用している Image Builder で、AMD ドライバーを更新する必要がある場合は、最新の AppStream 2.0 Graphics Design のベースイメージを使用するか、AMD ドライバをダウンロードして Image Builder にインストールします。	2021 年 6 月 4 日
ドキュメントの更新: 新しい Graphics Design のベースイメージ	新しいイメージを 2 つ追加リリースし、必要に応じて他のコンテンツを更新しました。	2021 年 6 月 3 日
ドキュメントの更新: 新しい AppStream 2.0 エージェントのバージョン	エージェントのバージョン 05-17-2021 に関するエントリを追加し、必要に応じて他の内容を更新しました。	2021 年 5 月 26 日

AppStream 2.0 エージェントの更新	複数の問題点を解決	2021 年 5 月 17 日
AppStream 2.0 クライアントの更新プログラム: バージョン 1.1.294	SAML 2.0 認証、Windows 7 でのクライアントの安定性、およびクライアント再接続時のフォルダー共有に関する問題を解決	2021 年 4 月 26 日
新しいクライアントバージョン	クライアントバージョン 1.1.1154	2021 年 4 月 25 日
ドキュメントの更新: AppStream 2.0 でのマネージド型イメージ更新機能のサポート	「イメージの管理」内に「AppStream 2.0 のマネージド型イメージ更新機能を使用してイメージを更新する」セクションを新設し、必要に応じて他の内容を更新しました。	2021 年 4 月 8 日
AppStream 2.0 クライアントの更新プログラム: バージョン 1.1.285	ウイルス対策ソフトウェアとの互換性を向上させる修正を含みます	2021 年 3 月 8 日
AppStream 2.0 エージェントの更新	複数の問題点を解決	2021 年 3 月 4 日
新しいエージェントのバージョン	エージェントのバージョン 02-21-2022	2021 年 2 月 24 日
新しいクライアントバージョン	クライアントバージョン 1.1.1130	2021 年 2 月 9 日
ドキュメントの更新: スマートカードのサポート	「System Requirements and Feature Support」の「Smart Cards」セクションを作成し、必要に応じてその他の内容を更新しました。	2021 年 1 月 12 日

AppStream 2.0 エージェントの更新	ストリーミングインスタンスへの Windows サインインおよびセッション内認証用スマートカードを使用できるようサポートを追加	2021 年 1 月 4 日
ドキュメントの更新: リアルタイムオーディオビデオ (AV) のサポート	「System Requirements and Feature Support」で「Real-Time Audio-Video」セクションを作成し、必要に応じてその他の内容を更新しました。	2020 年 12 月 28 日
AppStream 2.0 クライアント更新プログラム: バージョン 1.1.257	リアルタイムオーディオビデオ (AV) とスマートカード認証のサポートを追加し、Excel の問題を解決	2020 年 12 月 28 日
AppStream 2.0 ベースイメージの更新	Base、Graphics Design、Graphics G4dn、Graphics Pro の更新: 2020 年 12 月 9 日までの Microsoft Windows 更新プログラム、AWS CLI バージョン 1.18.138、Amazon SSM Agent バージョン 3.0.431.0 が含まれます。	2020 年 12 月 28 日
AppStream 2.0 エージェントの更新	複数の問題点を解決	2020 年 12 月 17 日
AppStream 2.0 エージェントの更新	複数の問題点を解決	2020 年 10 月 8 日
AppStream 2.0 エージェントの更新	複数の問題点を解決	2020 年 9 月 1 日

AppStream 2.0 クライアント更新プログラム: バージョン 1.1.195	ファイルリダイレクトを使用するときのローカルドライブとフォルダの共有が改善され、その他の機能強化と修正が提供されます。	2020 年 8 月 18 日
ドキュメントの更新: ローカルプリンターリダイレクトのサポート	「System Requirements and Feature Support」の「Enable Local Printer Redirection」セクションを作成し、必要に応じて他の内容を更新しました。	2020 年 8 月 7 日
AppStream 2.0 エージェントの更新	AppStream 2.0 クライアントへのローカルプリンターリダイレクトのサポートを追加し、複数の問題を解決	2020 年 7 月 30 日
AppStream 2.0 ベースイメージの更新	Base、Graphics Design、Graphics G4dn、Graphics Pro の更新: 2020 年 6 月 9 日までの Microsoft Windows 更新プログラム、AWS CLI バージョン 1.18.86、Amazon SSM エージェントバージョン 2.3.1319.0 が含まれます。	2020 年 7 月 16 日
新しいリージョン: ムンバイ	「SAML のセットアップ」のリリースステートエンドポイントテーブルを更新、必要に応じて他の内容を更新。	2020 年 7 月 8 日
AppStream 2.0 クライアント更新プログラム: バージョン 1.1.179	ローカルプリンターリダイレクトのサポートを追加し、その他の機能強化と修正を提供します。	2020 年 7 月 8 日

[ドキュメントの更新: タブレット描画のサポート](#)

「System Requirements and Feature Support」の「Drawing Tablets」セクションを作成し、必要に応じてその他の内容を更新しました。

2020 年 6 月 26 日

[AppStream 2.0 エージェントの更新](#)

複数の問題点を解決

2020 年 5 月 27 日

[AppStream 2.0 クライアント更新プログラム: バージョン 1.1.160](#)

.NET Framework バージョン 4.7.1 以前がインストールされている Windows PC でアプリケーションカタログページが開かない問題を解決します。また、別の断続的な問題も解決します。

2020 年 4 月 28 日

[ドキュメントの更新: オンデマンドのログ記録のサポート](#)

「System Requirements and Feature Support」の Automatic and On-Demand Diagnostic Log Uploads セクションを作成し、必要に応じてその他の内容を更新しました。

2020 年 4 月 22 日

[ドキュメントの更新: DNS TXT レコードでユーザー接続用の信頼されたサブドメインを定義するためのサポート](#)

「System Requirements and Feature Support」の「Create the AS2TrustedDomains DNS TXT Record to Enable Your Domain for the AppStream 2.0 Client Without Registry Changes」セクションを作成し、必要に応じて他の内容を更新しました。

2020 年 4 月 22 日

[AppStream 2.0 クライアント
更新プログラム: バージョン
1.1.156](#)

診断ログとミニダンプのオンデマンドアップロードのサポート、DNS TXT レコードでのユーザー接続用の信頼されたサブドメインの定義、その他の機能強化を追加しました。

2020 年 4 月 22 日

[AppStream 2.0 エージェント
の更新](#)

ストリーミングセッションが失敗する問題を解決し、IAM ロールのパフォーマンスを改善します。

2020 年 4 月 20 日

[AppStream 2.0 ベースイメージ
の更新](#)

Base、Graphics Design、Graphics G4dn、Graphics Pro (Windows Server 2019):
2020 年 3 月 10 日までの Microsoft Windows 更新プログラム、AWS CLI バージョン 1.18.21、Amazon SSM エージェントバージョン 2.3.930.0 を含む

2020 年 4 月 18 日

[AppStream 2.0 ベースイメージ
の更新](#)

Base、Graphics Design、Graphics Pro の更新: 2020 年 2 月 11 日までの Microsoft Windows 更新プログラム、AWS CLI バージョン 1.17.5、Amazon SSM エージェントバージョン 2.3.842.0 が含まれます。

2020 年 3 月 18 日

AppStream 2.0 ベースイメージの更新	Graphics G4dn インスタンス (Windows Server 2012 R2) のサポートを追加。 2020 年 2 月 11 日までの Microsoft Windows 更新プログラム、AWS CLI バージョン 1.17.5、Amazon SSM エージェントバージョン 2.3.842.0 も含まれています。	2020 年 3 月 16 日
AppStream 2.0 クライアント更新プログラム: バージョン 1.1.137	バージョン 1.1.136 の更新プログラムに戻します。	2020 年 3 月 8 日
AppStream 2.0 クライアント更新プログラム: バージョン 1.1.136	DNS TXT レコードでユーザー接続用の信頼されたサブドメインを定義するためのサポートを追加し、その他の機能強化と修正を提供しました。	2020 年 3 月 5 日
AppStream 2.0 ベースイメージの更新	Graphics G4dn インスタンスのサポートを追加 (Windows Server 2016、Windows Server 2019)。Microsoft Windows の 2020 年 2 月 11 日までの更新プログラム、AWS CLI バージョン 1.17.5、および Amazon SSM Agent のバージョン 2.3.842.0 を含む	2020 年 3 月 5 日
ドキュメントの更新: ネイティブアプリケーションモードのサポートを追加しました。	「System Requirements and Feature Support」の「Native application mode」セクションを作成し、必要に応じてその他の内容を更新しました。	2020 年 2 月 28 日

AppStream 2.0 クライアント 更新プログラム: バージョン 1.1.129	ネイティブアプリケーション モードのサポートを追加し、 その他の機能強化と修正を提 供しました。	2020 年 2 月 28 日
AppStream 2.0 エージェント の更新	ネイティブアプリケーション モードと Desktop (デスクトッ プ) ストリームビューのサポー トを追加	2020 年 2 月 19 日
AppStream 2.0 ベースイメー ジの更新	Graphics Design の更新プロ グラム: Windows Server 2019 のサポートを追加。2019 年 11 月 12 日までの Microsoft Windows 更新プログラムを含 む	2020 年 1 月 13 日
AppStream 2.0 エージェント の更新	複数の問題点を解決	2020 年 1 月 13 日
ドキュメントの更新: AppStream 2.0 のセキュリ ティに関するドキュメントの 強化	「Security in Amazon AppStream 2.0」を作成し、必 要に応じてその他の内容を更 新しました。	2019 年 12 月 23 日
AppStream 2.0 クライアント 更新プログラム: バージョン 1.0.525	ストリーミングセッション中 にユーザーがアプリケーション をクリックすると、マウス カーソルが間違った場所を指 す原因となる DPI の問題を解 決しました。	2019 年 12 月 12 日
AppStream 2.0 ベースイメー ジの更新	2019 年 12 月 12 日までの Microsoft Windows 更新プロ グラム、AWS CLI バージョン 1.16.284、Amazon SSM エー ジェントバージョン 2.3.760.0 が含まれます。	2019 年 12 月 12 日

ドキュメントの更新: AppStream 2.0 z1d ベースの インスタンスのサポート	「AppStream 2.0 Instance Families」を更新	2019 年 11 月 21 日
AppStream 2.0 エージェント の更新	AppStream 2.0 アセンブリは現在、実行可能ファイルとインストーラーパッケージを含め、署名されています	2019 年 11 月 13 日
ドキュメントの更新: 埋め込み AppStream 2.0 ストリーミング セッション	「Embed AppStream 2.0 Streaming Sessions」を作成し、必要に応じて他のコンテンツを更新。	2019 年 11 月 1 日
AppStream 2.0 クライアント 更新プログラム: バージョン 1.0.511	最大 4 台のモニターのサポートを追加し、その他に機能を強化	2019 年 10 月 16 日
ドキュメントの更新: 新しい リージョン — AWS GovCloud (米国西部)	AWS GovCloud (US) ユーザーガイドに「 Amazon AppStream 2.0 」を新設し、「SAML のセットアップ」内のリレーステートエンドポイントの一覧表と、必要に応じて他のコンテンツの更新を行いました。	2019 年 10 月 9 日
AppStream 2.0 エージェント の更新	システムプロキシサーバーをバイパスしないように AppStream 2.0 ストレージコネクタを変更	2019 年 10 月 8 日
ドキュメントの更新: FIPS 準 拠のエンドポイント	「FIPS エンドポイントを使用した転送中のデータの保護」を作成し、必要に応じて他のコンテンツを更新しました。	2019 年 10 月 7 日

[AppStream 2.0 クライアント
更新プログラム: バージョン
1.0.499](#)

Bluetooth ヘッドセットがローカルコンピュータに接続されているときに、クライアント側のハードウェアレンダリングとクライアントが正しく動作しない問題を解決しました。

2019 年 9 月 26 日

[AppStream 2.0 エージェント
の更新](#)

複数の問題点を解決

2019 年 9 月 23 日

[AppStream 2.0 ベースイメー
ジの更新](#)

すべての Base インスタンスと Graphics Pro インスタンス、および Graphics Design Windows Server 2012 R2 インスタンスの更新: 2019 年 8 月 13 日までの Microsoft Windows 更新プログラムと AWS CLI バージョン 1.16.222 が含まれています。Base、Graphics Design、Graphics Pro インスタンスの追加更新

2019 年 9 月 18 日

[ドキュメントの更新:
AppStream 2.0 ストリーミング
インスタンスへの IAM ロール
の適用のサポート](#)

「Using an IAM Role to Grant Permissions to Applications and Scripts Running on AppStream 2.0 Streaming Instances」を新規作成、必要に応じて他の内容を更新。

2019 年 9 月 9 日

[AppStream 2.0 ベースイメージの更新](#)

Graphics Design インスタンスの更新: 2019 年 8 月 13 日までの Microsoft Windows 更新プログラム、AWS CLI バージョン 1.16.222、AMD ドライバー 24.20.13028.3002 が含まれています。

2019年9月5日

[AppStream 2.0 エージェントの更新](#)

AppStream 2.0 ストリーミングインスタンスに IAM ロールを適用するためのサポートを追加

2019 年 9 月 3 日

[ドキュメントの更新:
AppStream 2.0 ファイルシステムリダイレクト](#)

「Enable File System Redirection for Your AppStream 2.0 Users」を新規作成、必要に応じて他の内容を更新。

2019 年 8 月 20 日

[ドキュメントの更新: 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 エンドポイントからの作成とストリーミング」を新規作成。また、「インターフェイス VPC エンドポイント経由で AppStream 2.0 API オペレーションおよび CLI コマンドにアクセスする」を作成し、必要に応じて他の内容を更新。

2019 年 8 月 19 日

[AppStream 2.0 クライアント更新プログラム: バージョン 1.0.480](#)

AppStream 2.0 ファイルシステムリダイレクトのサポートを追加

2019 年 8 月 14 日

[AppStream 2.0 エージェントの更新](#)

AppStream 2.0 ファイルシステムリダイレクトのサポートを追加

2019 年 8 月 8 日

ドキュメントの更新: プログラムによる AppStream 2.0 イメージの作成	「Create Your AppStream 2.0 Image Programmatically」を作成し、必要に応じてその他のコンテンツを更新。	2019 年 8 月 1 日
AppStream 2.0 エージェントの更新	AppStream 2.0 イメージをプログラムで作成するためのサポートを追加します。	2019 年 7 月 26 日
ドキュメントの更新: Windows Server 2016 と Windows Server 2019 ベースイメージのサポートを追加しました	「AppStream 2.0 Base Image Version History」および必要に応じて他の内容を更新。	2019 年 6 月 28 日
AppStream 2.0 エージェントの更新	Windows Server 2016 と Windows Server 2019 ベースイメージのサポートを追加しました。	2019 年 6 月 19 日
AppStream 2.0 ベースイメージの更新	Windows Server 2016 および Windows Server 2019 のサポートを追加しました	2019 年 6 月 10 日
AppStream 2.0 ベースイメージの更新	2019 年 5 月 14 日までの Microsoft Windows 更新プログラムが含まれます。	2019 年 5 月 28 日
ドキュメントの更新: AppStream 2.0 の使用状況レポート	「AppStream 2.0 Usage Reports」を作成し、必要に応じて他の内容を更新。	2019 年 5 月 21 日
ドキュメントの更新: アイドルユーザー切断のサポート	「Create an AppStream 2.0 Fleet and Stack」の「Create a Fleet」を更新。	2019 年 5 月 17 日

[AppStream 2.0 クライアント
更新プログラム: バージョン
1.0.407](#)

ユーザーがストリーミングセッションから切断されるまでにアイドル状態 (非アクティブ) となることができる時間を設定するためのサポートを追加しました。

2019 年 5 月 16 日

[AppStream 2.0 エージェント
の更新](#)

ユーザーがストリーミングセッションから切断されるまでにアイドル状態 (非アクティブ) となることができる時間を設定するためのサポートを追加しました。また、AppStream 2.0 使用状況レポートをサブスクライブするためのサポートも追加しました。

2019 年 5 月 7 日

[AppStream 2.0 ベースイメージ
の更新](#)

2019 年 4 月 20 日までの Microsoft Windows 更新プログラム、AWS CLI バージョン 1.16.126、Graphics Pro インスタンス用の NVIDIA Graphics Driver 412.16 が含まれます。

2019 年 4 月 29 日

[ドキュメントの更新: を使用し
た AppStream 2.0 API コール
のログ記録 AWS CloudTrail](#)

「を使用した AppStream 2.0 API コールのログ記録」を作成しました AWS CloudTrail。

2019 年 4 月 25 日

[ドキュメントの更新: HIPAA
のコンプライアンス](#)

「AppStream 2.0 Compliance」を作成しました。

2019 年 3 月 28 日

[ドキュメントの更新: ジェス
チャのサポート](#)

「タッチスクリーンデバイス」を作成し、必要に応じて他のコンテンツを更新しました。

2019 年 3 月 13 日

[AppStream 2.0 クライアント 更新プログラム: バージョン 1.0.375](#)

Windows PC でのタッチスクリーンのサポートが追加されました。新しいストリーミングセッションの開始時に USB デバイスを自動的に接続できるようになりました。セッションスクリプトの実行のサポートが追加されました。AppStream 2.0 の動的アプリケーションフレームワーク API を使用して仮想化アプリケーションを配信できるようになりました。

2019 年 3 月 7 日

[AppStream 2.0 エージェント の更新](#)

タッチ対応の iPad、Android タブレット、Windows デバイスでのジェスチャのサポートを追加しました。

2019 年 3 月 7 日

[新しいリージョン: ソウル](#)

「SAML のセットアップ」のリリースステートエンドポイントテーブルを更新、必要に応じて他の内容を更新。

2019 年 2 月 13 日

[ドキュメントの更新: セッションスクリプト](#)

「Images (イメージ)」に「Use Session Scripts to Manage Your AppStream 2.0 Users' Streaming Experience」を作成し、必要に応じて他のコンテンツを更新。

2019 年 1 月 27 日

[AppStream 2.0 ベースイメージ の更新](#)

2018 年 12 月 10 日までの Microsoft Windows 更新プログラム、AWS CLI バージョン 1.16.84、Graphics Pro インスタンス用の NVIDIA Graphics Driver 391.58 が含まれます。

2019 年 1 月 22 日

[AppStream 2.0 エージェント の更新](#)

インスタンスセッションスクリプトの使用のサポートを追加しました。また、リソース作成中に AppStream 2.0 リソースタイプ (Image Builder、イメージ、フリート、スタック) にタグを追加できるようになりました。

2019 年 1 月 22 日

[AppStream 2.0 クライアント 更新プログラム: バージョン 1.0.320](#)

AppStream 2.0 動的アプリケーションフレームワーク API、AppStream 2.0 のリージョン設定、AppStream 2.0 ユーザープールのサポートを追加し、その他の機能強化を提供

2019 年 1 月 19 日

[ドキュメントの更新: デフォルト のリージョン設定](#)

「Configure Default Regional Settings for Your AppStream 2.0 Users」の新規作成、必要に応じて他の内容を更新。

2018 年 12 月 13 日

[ドキュメントの最新更新日: 動的 アプリケーションフレーム ワーク](#)

「アプリケーションの使用権限の管理」を新規作成、必要に応じて他の内容を更新。

2018 年 12 月 7 日

[AppStream 2.0 エージェント の更新](#)

AppStream 2.0 動的アプリケーションフレームワークを使用して動的アプリケーションプロバイダーを構築するためのサポートを追加しました。また、Windows を実行するウェブクライアントで日本語キーボードを使用するためのサポートも追加されています。

2018 年 12 月 4 日

ドキュメントの更新: AppStream 2.0 クライアント	「The AppStream 2.0 Client」を作成し、必要に応じて他のコンテンツを更新。	2018 年 11 月 20 日
AppStream 2.0 クライアント 更新プログラム: バージョン 1.0.247	初回リリース	2018 年 11 月 20 日
AppStream 2.0 エージェント の更新	AppStream 2.0 Windows クライアントを使用してストリーミングセッションを開始するためのサポートを追加します。	2018 年 11 月 14 日
ドキュメントの更新: イメージ の共有	「Administer Your AppStream 2.0 Images」を新設し、他のコンテンツを必要に応じて更新。	2018 年 9 月 14 日
ドキュメントの更新: アプリ ケーション設定の永続化	「Enable Application Settings Persistence for Your AppStream 2.0 Users」を新設し、他のコンテンツを必要に応じて更新。	2018 年 9 月 5 日
AppStream 2.0 エージェント の更新	アプリケーション設定の永続化に関するサポートを追加します。	2018 年 8 月 29 日
ドキュメントの更新: OneDrive のサポート	「Enable and Administer OneDrive for Your AppStream 2.0 Users」の新規作成、必要に応じて他の内容を更新。	2018 年 7 月 31 日
AppStream 2.0 エージェント の更新	OneDrive の永続的ストレージのサポートを追加	2018 年 7 月 26 日

AppStream 2.0 エージェントの更新	アプリケーションの起動用イメージの最適化に関する問題を解決します。	2018 年 6 月 19 日
ドキュメントの更新: 地域の設定	「Enable Regional Settings for Your AppStream 2.0 Users」の新規作成、必要に応じて他の内容を更新。	2018 年 6 月 14 日
ドキュメントの更新: デフォルトのアプリケーションと Windows の設定	「ステップ 4: デフォルトのアプリケーションと Windows の設定」を「チュートリアル: カスタムイメージの作成」に追加、必要に応じて他の内容を更新。	2018 年 14 月 6 日
AppStream 2.0 ベースイメージの更新	2018 年 5 月 9 日までの Microsoft Windows 更新プログラムおよび Windows PowerShell 5.1 が含まれます。	2018 年 12 月 6 日
AppStream 2.0 エージェントの更新	地域の設定とデフォルトのアプリケーションと Windows 設定のサポートを追加します。	2018 年 6 月 6 日
ドキュメントの更新: Google ドライブのサポート	「Enable and Administer Google Drive for Your AppStream 2.0 Users」の新規作成、必要に応じて他の内容を更新。	2018 年 6 月 4 日

以前の更新

次の表では、2018 年 6 月 4 日以前の AppStream 2.0 サービスおよび Amazon AppStream 2.0 管理ガイドドキュメントへの重要な追加事項について説明します。

Note

個々の AppStream 2.0 ベースイメージと AppStream 2.0 エージェントのリリースは、この表には含まれていません。これらのリリースの詳細については、「[AppStream 2.0 ベースイメージとマネージド型イメージの更新に関するリリースノート](#)」および「[AppStream 2.0 エージェントのリリースノート](#)」を参照してください。

変更	説明	日付
データ転送の管理コントロール	「Create AppStream 2.0 Fleets and Stacks」の「Create a Stack」セクションを更新、必要に応じて他の内容を更新。	2018 年 5 月 24 日
新リージョン: フランクフルト	「Setting Up SAML」のリレーステートエンドポイントテーブルを更新、必要に応じて他の内容を更新。	2018 年 3 月 28 日
カスタムブランド	「Add Your Custom Branding to Amazon AppStream 2.0」を新規作成、必要に応じて他の内容を更新。	2018 年 3 月 26 日
イメージのコピー	「チュートリアル: カスタムイメージの作成」を更新、必要に応じて他の内容を更新。	2018 年 2 月 23 日
新しいリージョン: シンガポールとシドニー	「SAML のセットアップ」のリレーステートエンドポイントテーブルを更新、必要に応じて他の内容を更新。	2018 年 1 月 24 日
リソースへのタグ付け	「Tagging Your Amazon AppStream 2.0 Resources」を新規作成、必要に応じて他の内容を更新。	2017 年 15 月 12 日
管理された AppStream 2.0 エージェントの更新	「Amazon AppStream 2.0 Agent Version History」を新規作成、必要に応じて他の内容を更新。	2017 年 12 月 7 日

変更	説明	日付
オンデマンドフリート	「Amazon AppStream 2.0 Fleets and Stacks」に「Fleet Type」セクションを追加、必要に応じて他の内容を更新。	2017 年 9 月 19 日
インスタンスタイプ	「Amazon AppStream 2.0 Instance Families」を新規作成、必要に応じて他の内容を更新。	2017 年 7 月 25 日
Active Directory	「Using Active Directory with Amazon AppStream 2.0」を新規作成、必要に応じて他の内容を更新。	2017 年 7 月 24 日
ユーザープール	「Manage Access Using the AppStream 2.0 User Pool」を新規作成、必要に応じて他の内容を更新。	2017 年 15 月 6 日
セキュリティグループ	「Network Settings for Amazon AppStream 2.0」に「Security Groups」セクションを追加、必要に応じて他の内容を更新。	2017 年 5 月 26 日
ホームフォルダ	「Enable and Administer Home Folders for Your AppStream 2.0 Users」を新規作成、必要に応じて他の内容を更新。	2017 年 5 月 18 日
デフォルトのインターネットアクセス	「Network Settings for Amazon AppStream 2.0」を新規作成、必要に応じて他の内容を更新。	2017 年 4 月 21 日
フリートの自動スケーリング	「Fleet Auto Scaling for Amazon AppStream 2.0」を新規作成、必要に応じて他の内容を更新。	2017 年 3 月 23 日
フリートの管理	「Amazon AppStream 2.0 Fleets and Stacks」を新規作成、必要に応じて他の内容を更新。	2017 年 2 月 22 日
SAML 2.0 のサポート	「Single Sign-on Access to AppStream 2.0 Using SAML 2.0」を新規作成、必要に応じて他の内容を更新。	2017 年 2 月 15 日

変更	説明	日付
Image Builder	「AppStream 2.0 Image Builders」を新規作成、必要に応じて他の内容を更新。	2017 年 1 月 19 日
初回のドキュメントリリース	Amazon AppStream 2.0 管理ガイドの最初のリリースを作成しました。	2016 年 12 月 01 日

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。