



Unable to locate subtitle

AWS Snowball Edge デベロッパーガイド



AWS Snowball Edge デベロッパーガイド: ***Unable to locate subtitle***

Copyright © 2024 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Amazon の商標とトレードドレスは、Amazon 以外の製品またはサービスとの関連において、顧客に混乱を招いたり、Amazon の名誉または信用を毀損するような方法で使用することはできません。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

Snowball Edge とは	1
AWS Snowball エッジ機能	1
関連 サービス	2
Snow Family サービスへのアクセス	3
AWS Snowball Edge デバイスへのアクセス	4
AWS Snowball Edge の料金	4
デバイスモニタリング	4
初めて使用する AWS Snowball ユーザーのリソース	4
デバイスハードウェア情報	5
デバイス設定	5
デバイスのユースケース	8
デバイスの仕様	10
ネットワークハードウェア	18
Snow Family デバイスを使用するための前提条件	21
にサインアップする AWS アカウント	22
管理アクセスを持つユーザーを作成する	22
ローカル環境について	24
特殊文字の使用	24
を使用した Amazon S3 暗号化 AWS KMS	26
サーバー側の暗号化による Amazon S3 の暗号化	30
Snow Family デバイスで Amazon S3 アダプターをインポートジョブ/エクスポートジョブに 使用するための前提条件	30
Snow Family デバイスの Amazon S3 互換ストレージを使用するための前提条件	31
Snow Family デバイスでコンピューティングインスタンスを使用するための前提条件	32
Snowball Edge の仕組み	34
インポートジョブの仕組み	36
エクスポートジョブの仕組み	36
ローカルコンピューティングおよびストレージジョブの仕組み	37
クラスター化されたローカルコンピューティングおよびストレージジョブの仕組み	37
Snowball Edge の動画とブログ	38
Snowball Edge デバイスの長期料金	39
長期料金期間中のデバイスの交換	39
配送に関する考慮事項	41
リージョンごとの配送制限	41

開始方法	43
Snow Family デバイスを注文するジョブの作成	44
ステップ 1: ジョブタイプを選択する	45
ステップ 2: コンピューティングとストレージのオプションを選択する	46
ステップ 3: 機能とオプションを選択する	51
ステップ 4: セキュリティ、配送、通知の設定を選択する	52
ステップ 5: ジョブの概要を確認してジョブを作成する	55
ダウンロード AWS OpsHub	56
Snow Family デバイスを注文するためのジョブのキャンセル	57
Snowball Edge の受領	57
ローカルネットワークへの接続	59
Snow Family デバイスにアクセスするための認証情報の取得	61
Snowball Edge クライアントのダウンロードとインストール	62
Snow Family デバイスのロック解除	63
Snow Family デバイスのロック解除のトラブルシューティング	66
ローカルユーザーの設定	66
Snow Family デバイスの再起動	68
Snowball Edge の電源をオフにする	73
デバイスの返送	77
Snow Family デバイスの返送	78
配送業者	78
インポートステータスのモニタリング	87
ジョブ完了レポートとログの取得	88
大規模なデータ移行	91
大規模な転送の計画	91
ステップ 1: クラウドに移行する内容を把握する	92
ステップ 2: 目標転送速度を計算する	92
ステップ 3: Snow Family デバイスの必要数を決定する	92
ステップ 4: ジョブを作成する	93
ステップ 5: データを転送セグメントに分割する	93
大規模な転送のキャリブレーション	94
大規模データ移行プランの作成	95
ステップ 1: 移行の詳細を選択する	95
ステップ 2: 配送、セキュリティ、通知の設定を選択する	101
ステップ 3: プランを確認して作成する	102
大規模データ移行プランの使用	102

推奨ジョブ注文スケジュール	102
注文済みジョブリスト	104
モニタリングダッシュボード	104
AWS OpsHub を使用してデバイスを管理する	106
Snow Family デバイスの AWS OpsHub をダウンロードする	107
デバイスのロック解除	107
デバイスをローカルでロック解除する	108
デバイスをリモートでロック解除する	111
の署名の検証 AWS OpsHub	114
AWS サービスの管理	118
コンピューティングインスタンスのローカル使用	119
クラスターの管理	133
Snow Family デバイスの Amazon S3 互換ストレージをセットアップする	134
S3 ストレージの管理	141
NFS インターフェイスの管理	144
デバイスの管理	153
デバイスの再起動	153
デバイスのシャットダウン	156
デバイスのエイリアスの編集	158
を使用したパブリックキー証明書の管理 OpsHub	158
更新の取得	160
プロフィールの管理	162
管理タスクの自動化	164
タスクの作成と開始	164
タスクの詳細の表示	167
タスクの削除	168
デバイスの NTP タイムサーバーの設定	168
Snowball Edge Device の使用	170
Snowball Edge クライアントコマンドの使用	172
Snowball Edge クライアント用プロフィールの設定	173
NFC 検証の QR コードの取得	174
Snowball Edge クライアントバージョン	175
Snowball Edge デバイスのロック解除	175
Snowball Edge の更新	176
認証情報の取得	179
Snowball Edge でのサービスの開始	180

Snowball Edge でのサービスの停止	181
NFS の開始とアクセスの制限	182
NFS 実行中に NFS 共有へのアクセスを制限する	183
AWS Snowball Edge ログ	183
デバイスステータスの取得	185
サービスステータスの取得	189
クラスターからノードを削除する	194
クラスターへのノードの追加	195
デバイス用のタグの作成	195
デバイスからのタグの削除	196
デバイスのタグの説明	196
ダイレクトネットワークインターフェイスの作成	197
ダイレクトネットワークインターフェイスに関する情報の取得	198
ダイレクトネットワークインターフェイスの更新	198
ダイレクトネットワークインターフェイスの削除	199
仮想ネットワークインターフェイス (VNI) を作成する	200
仮想ネットワークインターフェイスに関する情報の取得	201
仮想ネットワークインターフェイスの更新	201
仮想ネットワークインターフェイスの削除	202
機能ステータスのチェック	203
タイムサーバーの設定	203
タイムソースのチェック	204
MTU サイズの更新	206
S3 アダプターを使用したファイルの転送	207
Amazon S3 AWS CLI アダプターで使用するバージョン 1.16.14 のダウンロードとインストール Amazon S3	208
Snowball Edge デバイスで AWS CLI および API オペレーションを使用する	209
ローカルの Amazon S3 認証情報の取得と使用	210
Amazon S3 アダプターではサポートされていない Amazon S3 の機能	212
小さなファイルのバッチ処理	212
サポートされている CLI コマンド	215
サポートされている REST API アクション	219
NFS インターフェイスの管理	222
Snow Family デバイスの NFS 設定	224
EC2 互換インスタンス AWS IoT Greengrass での の使用 EC2-compatible	228
Amazon EC2 互換インスタンスの設定	228

の使用 AWS Lambda	231
開始する前に	232
Snowball Edge デバイスに Lambda 関数をデプロイする	234
Amazon EC2 互換コンピューティングインスタンスの使用	234
概要	235
Snow Family デバイス上の Amazon EC2 と Amazon EC2 互換インスタンスの違い	236
Snowball Edge のコンピューティングインスタンスの料金	237
Snow Family デバイスでの AMI の使用	237
Snow Family デバイスに VM イメージをインポートする	247
AWS CLI および API オペレーションの使用	263
コンピューティングインスタンスのクォータ	264
コンピューティングジョブの作成	268
コンピューティングインスタンスのネットワーク設定	270
SSH を使用してコンピューティングインスタンスに接続する	277
同じデバイス上で、コンピューティングインスタンスからバケットにデータを転送する	278
コンピューティングインスタンスの Snowball Edge クライアントのコマンド	279
Amazon EC2 互換エンドポイントの使用	284
EC2 互換インスタンスの自動起動	304
Amazon EC2 互換インスタンスで Snow 用インスタンスメタデータサービスを使用する	305
EC2 互換インスタンスでのブロックストレージの使用	315
セキュリティグループ	316
サポートされるインスタンスメタデータとユーザーデータ	317
EC2 互換インスタンスの停止	319
コンピューティングインスタンスのトラブルシューティング	319
Snow Family デバイスの Amazon S3 互換ストレージを使用する	321
Snow Family デバイスの Amazon S3 互換ストレージを注文する	325
Snow Family デバイスでの Amazon S3 互換ストレージのセットアップと起動	325
Snowball Edge デバイスで S3 バケットを操作する	330
Snowball Edge デバイスで S3 オブジェクトを操作する	338
Snow Family デバイスの Amazon S3 互換ストレージでサポートされる REST API アクション	345
Snow Family デバイスの Amazon S3 互換ストレージを Snow デバイスのクラスターで使用 する	346
Snow Family デバイスの Amazon S3 互換ストレージでのイベント通知の設定	363
ローカル SMTP 通知の設定	366
AWS Snow での Amazon EKS Anywhere の使用	367

Amazon EKS Anywhere on AWS Snow 用の Snowball Edge デバイスを注文する前に完了すべきアクション	369
Amazon EKS Anywhere on Snow で使用する AWS Snowball Edge デバイスの注文	370
Snowball Edge デバイスで Amazon EKS Anywhere を設定して実行する	371
切断されたオペレーションのための Amazon EKS Anywhere on AWS Snow の設定	383
クラスターの作成と保守	383
ローカルでの IAM の使用	384
AWS CLI および API オペレーションの使用	385
サポートされている IAM AWS CLI コマンド	386
IAM ポリシーの例	390
TrustPolicy 例	394
の使用 AWS STS	395
Snowball Edge での AWS CLI および API オペレーションの使用	395
Snowball Edge でサポートされている AWS STS AWS CLI コマンド	396
サポートされている AWS STS API オペレーション	397
公開鍵証明書の管理	397
証明書の一覧表示	398
証明書の取得	399
証明書の削除	399
AWS サービスを使用するために必要なポート	400
Snow Device Management を使用したデバイスの管理	402
Snow Family デバイスの注文時の Snow Device Management 状態の選択	403
Snow Device Management のアクティブ化	404
Snow Device Management のアクセス許可を IAM ロールに追加する	405
Snow Device Management CLI コマンド	406
タスクを作成する	407
タスクのステータスを確認する	408
デバイス情報を確認する	409
Amazon EC2 互換インスタンスのステータスを確認する	411
タスクのメタデータを確認する	413
タスクをキャンセルする	414
コマンドと構文を一覧表示する	415
リモート管理可能なデバイスを一覧表示する	416
デバイスを横断してタスクのステータスを一覧表示する	417
利用可能なリソースを一覧表示する	418
デバイスタグまたはタスクタグを一覧表示する	419

タスクをステータス別に一覧表示する	420
タグを適用する	421
タグを削除する	422
AWS Snowball Edge ジョブについて	423
ジョブの詳細	423
ジョブステータス	426
クラスターステータス	429
Amazon S3 へのジョブのインポート	430
Amazon S3 からのジョブのエクスポート	431
エクスポート範囲の使用	432
エクスポートジョブのベストプラクティス	439
ローカルコンピューティングおよびストレージ専用ジョブ	439
ローカルストレージジョブ	440
ローカルクラスターオプション	440
コンソールでのジョブのクローン	441
ベストプラクティス	442
セキュリティ	442
リソース管理	443
パフォーマンス	443
パフォーマンスに関する推奨事項	445
データ転送の高速化	445
Snowball Edge デバイスの更新	447
前提条件	448
更新をダウンロードする	448
更新をインストールする	452
SSL 証明書を更新する	459
Snow Family デバイスの Amazon Linux 2 AMI を更新する	460
セキュリティ	461
データ保護	462
クラウドでのデータの保護	463
デバイスのデータの保護	467
Identity and Access Management	470
コンソールとジョブのアクセス制御	470
ログ記録とモニタリング	509
コンプライアンス検証	510
耐障害性	511

インフラストラクチャセキュリティ	512
データ検証	513
転送されたデータのチェックサムの検証	513
Snowball 転送中のローカルインベントリの作成	513
一般的な検証エラー	514
Amazon S3 にデータをインポートした後、Snowball Edge を手動検証する	514
通知	516
Snow が Amazon SNS を使用する仕組み	516
Snow ジョブのステータス変更の SNS トピックを暗号化する	516
カスタマーマネージド KMS キーポリシーの設定	517
SNS 通知の例	518
でのログ記録 AWS CloudTrail	531
AWS Snowball のエッジ情報 CloudTrail	531
AWS Snowball Edge のログファイルエントリの概要	532
クォータ	534
のリージョンの可用性 AWS Snowball Edge	534
AWS Snowball Edge ジョブの制限事項	535
のレート制限 AWS Snowball Edge	536
Amazon Snow S3 アダプターの接続制限	536
Snowball Edge デバイスでオンプレミスのデータを転送する際の制限	536
Snowball Edge の配送の制限	537
インポート用に返却された Snowball Edge の処理の制限	537
トラブルシューティング	538
デバイスの識別	539
起動に関する問題のトラブルシューティング	541
起動中の LCD ディスプレイの問題のトラブルシューティング	542
起動時の E Ink ディスプレイに関する問題のトラブルシューティング	544
接続の問題	544
unlock-device コマンドの問題のトラブルシューティング	545
マニフェストファイルの問題	545
認証情報の問題	545
認証情報が見つかりません AWS CLI	546
エラーメッセージ: シークレットアクセスキーの確認と署名	546
NFS インターフェイスの問題のトラブルシューティング	546
データ転送の問題	548
AWS CLI 問題	548

AWS CLI エラーメッセージ : 「Profile Cannot Be Null」	549
を使用してデータを転送するときの Null ポインタエラー AWS CLI	549
インポートジョブの問題	549
エクスポートジョブの問題	550
ドキュメント履歴	552
AWS 用語集	560
.....	dlxi

AWS Snowball Edge とは

AWS Snowball Edge は、一部の AWS 機能用のオンボードストレージとコンピューティング能力を備えた Snowball デバイスの一種です。Snowball Edge は、ローカル環境と AWS クラウドまたはとローカル環境間でデータを転送するだけでなく、ローカルでデータを処理 AWS クラウドしてエッジコンピューティングワークロードを実行できます。

各 Snowball Edge デバイスは、インターネットよりも高速でデータを転送できます。この転送は、デバイス内のデータをリージョンのキャリアが配送することで行われます。アプライアンスは頑丈で、E Ink 配送ラベルを完備しています。

Snowball Edge デバイスのデバイス設定には、ストレージの最適化、コンピューティングの最適化、GPU によるコンピューティングの最適化の 3 つのオプションが用意されています。このガイドでは、Snowball Edge デバイスとはこのデバイスのすべてのオプションを指しています。特定の情報がデバイスの 1 つ (あるいは複数) の設定のみに適用される場合 (GPU を使用した Snowball Edge にオンボード GPU 周辺機能が用意される詳細など) には、限定して呼び出します。詳細については、「[Snowball Edge デバイス設定](#)」を参照してください。

トピック

- [AWS Snowball エッジ機能](#)
- [AWS Snowball Edge に関連するサービス](#)
- [Snow Family サービスへのアクセス](#)
- [AWS Snowball Edge の料金](#)
- [デバイスモニタリング](#)
- [初めて使用する AWS Snowball ユーザーのリソース](#)
- [AWS Snowball エッジデバイスのハードウェア情報](#)
- [Snow Family デバイスを使用するための前提条件](#)

AWS Snowball エッジ機能

Snowball Edge デバイスには次のような特徴があります。

- デバイス用の大容量のストレージ容量とコンピューティング機能。これは、ジョブの作成時に選択するオプションによって異なります。

- 最大で 100 Gbit/秒の転送速度のネットワークアダプタ。
- 暗号化が強制であり、保管中や輸送中のデータを保護します。
- ローカル環境と Amazon S3 の間でデータをインポートまたはエクスポートし、1 つ以上のデバイスでインターネットを使用せずにデータを物理的に転送できます。
- Snowball Edge デバイスは、それ自体が頑丈なボックスです。デバイスが発送可能になると、組み込みの E Ink ディスプレイが変化して配送ラベルを表示します。
- Snowball Edge デバイスには、ネットワーク接続を管理し、サービスステータスの情報を取得するために使用できるオンボードの LCD ディスプレイが装備されています。
- Snowball Edge デバイスは、ローカルストレージおよびコンピューティングジョブ用にクラスター化できます。これにより、3~16 台のデバイス間でデータの耐久性を実現し、必要に応じてストレージをローカルで拡大または縮小できます。
- Kubernetes ワークロード用の Snowball Edge デバイスでは Amazon EKS Anywhere を使用できません。
- Snowball Edge デバイスには Amazon S3 および Amazon EC2 と互換性のあるエンドポイントが用意されており、ユースケースをプログラムで実行できます。
- Snowball Edge デバイスでは、新しい sbe1、sbe-c、および sbe-g インスタンスタイプがサポートされています。このインスタンスタイプでは Amazon マシンイメージ (AMI) を使用してデバイスでコンピューティングインスタンスを実行できます。
- Snowball Edge は、データ移行で次のデータ転送プロトコルをサポートしています。
 - NFSv3
 - NFSv4
 - NFSv4.1
 - Amazon S3 over HTTP または HTTPS (AWS CLI バージョン 1.16.14 以前と互換性のある API 経由)

AWS Snowball Edge に関連するサービス

AWS Snowball Edge デバイスは、以下の関連 AWS サービスで使用できます。

- Amazon S3 アダプター — Amazon S3 API オペレーションのサブセットをサポートする Snowball Edge 用 Amazon S3 API AWS を使用して、との間でプログラムによるデータ転送に使用します。このロールでは、データは AWS ユーザーに代わってによって Snow デバイスに転送され、デバイスは (エクスポートジョブの場合) ユーザーに配送されるか、空の Snow デバイスをユー

ザーに AWS 配送し、オンプレミスのソースからデバイスにデータを転送して AWS (インポートジョブの場合) に返送します。

- Snow Family デバイスの Amazon S3 互換ストレージ — Amazon EC2、Amazon EKS Anywhere on Snow などのコンピューティングサービスのデータニーズに対応するために使用します。この機能は Snowball Edge デバイスで使用でき、拡張された Amazon S3 API セットと、3 ~ 16 ノードの柔軟なクラスター設定による耐障害性の向上、ローカルバケット管理、ローカル通知などの機能を提供します。
- Amazon EC2 - Amazon EC2 API オペレーションのサブセットをサポートする Amazon EC2 と互換性のあるエンドポイントを使用して、Snowball Edge デバイスでコンピューティングインスタンスを実行します。AWSで Amazon EC2 を使用方法の詳細については、[Amazon EC2 Linux インスタンスの開始方法](#)を参照してください。
- Amazon EKS Anywhere on Snow - Snow Family デバイス上で Kubernetes クラスターを作成して運用できます。[AWS Snow での Amazon EKS Anywhere の使用](#) を参照してください。
- AWS Lambda による – AWS IoT Greengrass Snow Family デバイスの Amazon S3 互換ストレージに基づいて、AWS Snowball Edge デバイスで行われたストレージアクションに基づいて Lambda 関数を呼び出します。Lambda の使用方法の詳細については、「[AWS Lambda デベロッパーガイド](#)」の「[AWS Snowball Edge AWS Lambda での の使用](#)」を参照してください。
- Amazon Elastic Block Store (Amazon EBS) - EC2 互換インスタンスで使用するためのブロックレベルのストレージボリュームを提供します。詳細については、[Amazon Elastic Block Store \(Amazon EBS\)](#) を参照してください。
- AWS Identity and Access Management (IAM) — このサービスを使用して、AWS リソースへのアクセスを安全に制御します。詳細については、[IAM とは](#)を参照してください。
- AWS Security Token Service (AWS STS) – IAM ユーザーまたは認証するユーザー (フェデレーテッドユーザー) の一時的な権限が制限された認証情報をリクエストします。詳細については、「[IAM の一時的セキュリティ認証情報](#)」を参照してください。
- Amazon EC2 Systems Manager - このサービスを使用して、AWSのインフラストラクチャを表示および制御します。詳細については、[AWS Systems Manager とは](#)を参照してください。

Snow Family サービスへのアクセス

[AWS Snow ファミリーマネジメントコンソール](#) またはジョブ管理 API を使用して、ジョブを作成および管理できます。[AWS Snow ファミリーマネジメントコンソール](#) の使用方法の詳細については、「[開始方法](#)」を参照してください。ジョブ管理 API の詳細については、[AWS Snowballのジョブ管理 API リファレンス](#)を参照してください。

AWS Snowball Edge デバイスへのアクセス

Snowball Edge デバイスがオンサイトに到着したら、LCD 画面を使用してデバイスで IP アドレスを設定し、Snowball Edge クライアントまたは AWS OpsHub for Snow Family を使用してデバイスのロックを解除できます。その後、データ転送やエッジコンピューティングタスクを実行できます。詳細については、「[AWS Snowball エッジデバイスの使用](#)」を参照してください。

AWS Snowball Edge の料金

サービスや関連デバイスに関する料金については、「[AWS Snowball Edge の料金](#)」を参照してください。

デバイスモニタリング

AWS は Snow デバイスをモニタリングし、Snow デバイスが に接続されているときにメトリクスと使用状況情報を収集することがあります AWS リージョン。Snow デバイスが に接続されていない場合 AWS リージョン、AWS は Snow デバイスをモニタリングしません。

が修復不可能な問題 AWS を検出し、物理的な機器を交換する必要がある場合は、 から通知 AWS されます。その後、お客様のサイトに配送される代替ジョブを配置できます。Snow デバイスのモニタリングは Snow デバイスサービス料金の一部として含まれるため、追加料金はかかりません。

初めて使用する AWS Snowball ユーザーのリソース

AWS Snow Family サービスを初めて使用する場合は、以下のセクションを順番に読むことをお勧めします。

1. デバイスタイプとオプションの詳細については、[AWS Snowball エッジデバイスのハードウェア情報](#) を参照してください。
2. ジョブのタイプに関する詳細については、「[AWS Snowball Edge ジョブについて](#)」を参照してください。
3. AWS Snowball Edge デバイスの使用方法 end-to-end の概要については、「」を参照してください [AWS Snowball Edge の仕組み](#)。
4. 始める準備ができれば、「[開始方法](#)」を参照してください。
5. デバイスでのコンピューティングインスタンスの使用方法については、「[Amazon EC2 互換コンピューティングインスタンスの使用](#)」を参照してください。

AWS Snowball エッジデバイスのハードウェア情報

すべての Snowball Edge デバイスにはサイズや重量などの物理的特性がありますが、用途に合わせて異なるタイプのハードウェアが含まれています。データ転送用に設計されたデバイスはより多くのストレージで設定され、コンピューティング用に設計されたデバイスはより多くの仮想 CPUs とメモリで設定されます。このセクションでは、Snowball Edge デバイスの物理的特性と、それらのコンピューティングとストレージの仕様について説明します。

トピック

- [Snowball Edge デバイス設定](#)
- [デバイスのユースケース](#)
- [AWS Snowball エッジデバイスの仕様](#)
- [サポートされているネットワークハードウェア](#)

Snowball Edge デバイス設定

Snowball Edge デバイスでは、以下のデバイス構成オプションが用意されています。

- Snowball Edge ストレージ最適化 (データ転送用) – この Snowball Edge デバイスオプションには、使用可能なストレージ容量が 80 TB あります。
- Snowball Edge ストレージ最適化 210 TB – この Snowball Edge デバイスオプションには、210 TB の使用可能なストレージ容量があります。
- Snowball Edge ストレージ最適化 (EC2-compatible のコンピューティング機能付き) – この Snowball Edge デバイスオプションには、コンピューティング機能用に最大 80 TB の使用可能なストレージ容量、40 個の vCPUs、80 GB のメモリがあります。また、Amazon EC2-compatible AMIs。
- Snowball Edge コンピューティング最適化 – この Snowball Edge デバイス (AMD EPYC Gen2 搭載) は、104 個の vCPU、416 GB のメモリ、コンピューティングインスタンス専用の 28 TB の NVMe SSD を備え、最も多くのコンピューティング機能を提供します。

Snowball Edge のコンピューティング最適化 (AMD EPYC Gen1 を使用) には、最大 52 vCPUs、208 GB のメモリ、39.5 TB の使用可能なストレージ容量、コンピューティングインスタンス用の 7.68 TB の専用 NVMe SSD があります。

- Snowball Edge コンピューティング最適化 (GPU 搭載) – この Snowball Edge デバイスオプションはコンピューティング最適化 (AMD EPYC Gen1 搭載) オプションと同じで、グラフィックプロ

セッシングユニット (GPU) がインストールされています。GPU は P3 Amazon EC2 互換インスタンスタイプで使用可能なものと同様です。

Note

これらのデバイスで Snow Family デバイスで Amazon S3 互換ストレージを使用する場合、使用可能なストレージは異なります。Snow [Family デバイスの Amazon S3 互換ストレージでのストレージ容量については、「Snow Family デバイスの Amazon S3 互換ストレージの使用」](#)を参照してください。

これらの3つのオプションのコンピューティング機能の詳細については、「[Amazon EC2 互換コンピューティングインスタンスの使用](#)」を参照してください。[こちら](#)で、ジョブ作成とテラバイト単位のディスク容量の違いについて説明しています。

Note

このガイドで Snowball Edge デバイスに言及する場合は、デバイスのすべての種類のオプションを指しています。情報がデバイスの1つまたは複数の特定のオプション構成にのみ適用される場合は (GPU 搭載の Snowball Edge コンピューティング最適化オプションに搭載されている GPU 周辺機能など)、そのことが明示されています。

次の表は、さまざまなデバイスオプションの違いをまとめたものです。ハードウェアの仕様については、「[AWS Snowball エッジデバイスの仕様](#)」を参照してください。


	Snowball Edge ストレージ最適化 (データ転送用)	Snowball Edge ストレージ最適化 210 TB	Snowball Edge ストレージ最適化 (EC2 コンピューティング機能搭載)	Snowball Edge コンピューティング最適化 (AMD EPYC Gen2 および NVME 搭載)	Snowball Edge コンピューティング最適化 (AMD EPYC Gen1、HDD、オプションの GPU 搭載)
CPU	AMD Naples、32 コア、3.4 Ghz	AMD Rome、64 コア、2 GHz	AMD Naples、32 コア、3.4 Ghz	AMD Rome、64 コア、2 GHz	AMD Naples、32 コア、3.4 Ghz
vCPUs	40	104	40	104	52
使用可能メモリ	80 GB	416 GB	80 GB	416 GB	208 GB
セキュリティカード	はい	はい	はい	はい	はい
GPU (オプション)	なし	なし	なし	なし	NVidia V100
[SSD]	1 TB SATA	210 TB NVMe	1 TB SATA	28 TB NVMe	7.68 TB NVMe
使用可能な HDD	80 TB	該当しない	80 TB	該当しない	39.5 TB 使用可能
ネットワークインターフェイス	<ul style="list-style-type: none"> • 2x 10 Gbit – RJ45 (1 つ使用可能) • 1x 25 Gbit – SFP28 	<ul style="list-style-type: none"> • 2x 10 Gbit – RJ45 (1 つ使用可能) • 1x 25 Gbit – SFP28 	<ul style="list-style-type: none"> • 2x 10 Gbit – RJ45 (1 つ使用可能) • 1x 25 Gbit – SFP28 	<ul style="list-style-type: none"> • 2x 10 Gbit – RJ45 (1 つ使用可能) • 1x 25 Gbit – SFP28 	<ul style="list-style-type: none"> • 2x 10 Gbit – RJ45 (1 つ使用可能) • 1x 25 Gbit – SFP28

	Snowball Edge ストレージ最適化 (データ転送用)	Snowball Edge ストレージ最適化 210 TB	Snowball Edge ストレージ最適化 (EC2 コンピューティング機能搭載)	Snowball Edge コンピューティング最適化 (AMD EPYC Gen2 および NVME 搭載)	Snowball Edge コンピューティング最適化 (AMD EPYC Gen1、HDD、オプションの GPU 搭載)
	<ul style="list-style-type: none"> 1x 100 Gbit – QSFP28 	<ul style="list-style-type: none"> 1x 100 Gbit – QSFP28 	<ul style="list-style-type: none"> 1x 100 Gbit – QSFP28 	<ul style="list-style-type: none"> 1x 100 Gbit – QSFP28 	<ul style="list-style-type: none"> 1x 100 Gbit – QSFP28
物理的セキュリティ機能	<ul style="list-style-type: none"> 隠し磁気スクリュー 侵入スイッチ NFC タグ 改ざん防止インサート 改ざん検出のための Android アプリ GPS と携帯電話 コンフォーマルコーティング 	<ul style="list-style-type: none"> 隠し磁気スクリュー 侵入スイッチ NFC タグ 改ざん防止インサート 改ざん検出のための Android アプリ コンフォーマルコーティング 	<ul style="list-style-type: none"> 隠し磁気スクリュー 侵入スイッチ NFC タグ 改ざん防止インサート 改ざん検出のための Android アプリ GPS と携帯電話 コンフォーマルコーティング 	<ul style="list-style-type: none"> 隠し磁気スクリュー 侵入スイッチ NFC タグ 改ざん防止インサート 改ざん検出のための Android アプリ コンフォーマルコーティング 	<ul style="list-style-type: none"> 隠し磁気スクリュー 侵入スイッチ NFC タグ 改ざん防止インサート 改ざん検出のための Android アプリ コンフォーマルコーティング

デバイスのユースケース

次の表は、さまざまなユースケースを示しています AWS Snow Family devices。

ユースケース	Snowball Edge	AWS Snowcone
データを Amazon S3 にインポートする	✓	✓
Amazon S3 からエクスポートする	✓	
耐久性のあるローカルストレージ	✓	
を使用したローカルコンピューティング AWS Lambda	✓	✓
ローカルコンピューティングインスタンス	✓	✓
デバイスのクラスター内の耐久性のある Amazon S3 ストレージ	✓	
で使用する AWS IoT Greengrass (IoT)	✓	✓
NFS を介しての GUI によるファイル転送	✓	✓
GPU ワークロード	✓	

 Note

GPU のサポートが必要なワークロードでは、Snowball Edge コンピューティング最適化 (オプションの GPU 搭載) が必要です。

Snowball Edge ストレージ最適化 210TB は、Snow Family デバイスの NFS、S3 アダプター、Amazon S3 互換ストレージを介したデータ転送をサポートします。

AWS Snowball エッジデバイスの仕様

このセクションでは、AWS Snowball エッジデバイスタイプとハードウェアの仕様について説明します。

トピック

- [Snowball Edge Storage Optimized \(データ転送用\) の仕様](#)
- [Snowball Edge Storage Optimized 210 TB の仕様](#)
- [Snowball Edge Storage Optimized \(EC2 搭載\) の仕様](#)
- [Snowball Edge Compute Optimized デバイスの仕様](#)

Snowball Edge Storage Optimized (データ転送用) の仕様

次の表に、Snowball Edge Storage Optimized デバイスのハードウェア仕様が記載されています。

項目	Snowball Edge Storage Optimized (データ転送用) の仕様
ストレージ仕様	
HDD ストレージ容量	80 TB 使用可能
電源仕様	
電源	AWS リージョン 米国の : NEMA 5 ~ 15p 100 ~ 220 ボルト。すべての AWS リージョンで、電源ケーブルが同梱されます。
消費電力	平均的なユースケースには 304 ワットです。ただし、電源は 1200 ワットに設定されています。
電圧	100 ~ 240V AC

項目	Snowball Edge Storage Optimized (データ転送用) の仕様
頻度	47/63 Hz
データおよびネットワーク接続	2x 10 Gbit – RJ45 (1 つ使用可能) 1x 25 Gbit – SFP28 1x 100 Gbit – QSFP28
ケーブル	各 AWS Snowball Edge デバイスには、国固有の電源ケーブルが付属しています。その他の付属ケーブルやオプティクスは付属していません。詳細については、「 サポートされているネットワークハードウェア 」を参照してください。
熱要件	AWS Snowball エッジデバイスはオフィスの運用用に設計されており、データセンターの運用に最適です。
デシベル出力	平均して、AWS Snowball Edge デバイスは 68 デシベルの音を生成し、通常はバキュームクリーナーやリビングルームの音楽よりも静かです。
寸法と重量の仕様	
(重量)	49.7 ポンド (22.54 Kg)
高さ	15.5 インチ (394 mm)
幅	10.6 インチ (265 mm)
長さ。	28.3 インチ (718 mm)
環境仕様	

項目	Snowball Edge Storage Optimized (データ転送用) の仕様
振動	ASTM D4169トラックレベル I 0.73 GRMS に相当する非オペレーション上の使用
衝撃	70 G (MIL-S-901) に相当するオペレーション上の使用 50 G (ISTA-3A) に相当する非オペレーション上の使用
高度	0 ~ 3,000 メートル (0 ~ 10,000 フィート) に相当するオペレーション上の使用 0 ~ 12,000 メートルに相当する非オペレーション上の使用
温度範囲	0 ~ 45°C (オペレーション上)

Snowball Edge Storage Optimized 210 TB の仕様

次の表で、Snowball Edge Storage Optimized 210 TB デバイスのハードウェア仕様を示します。

項目	Snowball Edge Storage Optimized 210 TB の仕様
コンピューティングとメモリの仕様	
CPU	104 個の vCPU
RAM	416 GB
ストレージ仕様	
NVME ストレージ容量	210 TB 使用可能 (オブジェクトおよび NFS データ転送用)
SSD ストレージ容量	なし

項目	Snowball Edge Storage Optimized 210 TB の仕様
電源仕様	
電源	AWS リージョン 米国の : NEMA 5 ~ 15p 100 ~ 220 ボルト。すべての AWS リージョンで、電源ケーブルが同梱されます。
消費電力	平均的なユースケースは 304 ワットですが、電源の定格は 1,200 ワットです。
電圧	100 ~ 240V AC
頻度	47/63 Hz
データおよびネットワーク接続	2x 10 Gbit – RJ45 (1 つ使用可能) 1x 25 Gbit – SFP28 1x 100 Gbit – QSFP28
ケーブル	各 AWS Snowball Edge デバイスには、国固有の電源ケーブルが付属しています。その他の付属ケーブルやオプティクスは付属していません。詳細については、「 サポートされているネットワークハードウェア 」を参照してください。
熱要件	AWS Snowball Edge デバイスはオフィスの運用用に設計されており、データセンターの運用に最適です。
デシベル出力	平均して、AWS Snowball Edge デバイスは 68 デシベルの音を生成し、通常はバキュームクリーナーやリビングルームの音楽よりも静かです。
寸法と重量の仕様	
(重量)	49.7 ポンド (22.54 Kg)
高さ	15.5 インチ (394 mm)
幅	10.6 インチ (265 mm)
長さ。	28.3 インチ (718 mm)

項目	Snowball Edge Storage Optimized 210 TB の仕様
環境仕様	
振動	ASTM D4169 トラックレベル I 0.73 GRMS に相当する非オペレーション上の使用
衝撃	70 G (MIL-S-901) に相当するオペレーション上の使用 50 G (ISTA-3A) に相当する非オペレーション上の使用
高度	0 ~ 3,000 メートル (0 ~ 10,000 フィート) に相当するオペレーション上の使用 0 ~ 12,000 メートルに相当する非オペレーション上の使用
温度範囲	0 ~ 30°C (オペレーション上)

Snowball Edge Storage Optimized (EC2 搭載) の仕様

以下の表に、Snowball Edge Storage Optimized (EC2 搭載) デバイスのハードウェア仕様が記載されています。

項目	Snowball Edge Storage Optimized (EC2 搭載) の仕様
コンピューティングとメモリの仕様	
CPU	40 個の vCPU
RAM	80 GiB
ストレージ仕様	
HDD ストレージ容量	80 TB 使用可能 (オブジェクトおよびブロックストレージ用)
SSD ストレージ容量	1 TB 使用可能な SATA SSD ストレージ (ブロックストレージ用)
電源仕様	

項目	Snowball Edge Storage Optimized (EC2 搭載) の仕様
電源	AWS リージョン 米国の : NEMA 5 ~ 15p 100 ~ 220 ボルト。すべての AWS リージョンで、電源ケーブルが同梱されます。
消費電力	平均的なユースケースは 304 ワットですが、電源の定格は 1,200 ワットです。
電圧	100 ~ 240V AC
頻度	47/63 Hz
データおよびネットワーク接続	2x 10 Gbit – RJ45 (1 つ使用可能) 1x 25 Gbit – SFP28 1x 100 Gbit – QSFP28
ケーブル	各 AWS Snowball Edge デバイスには、国固有の電源ケーブルが付属しています。その他の付属ケーブルやオプティクスは付属していません。詳細については、「 サポートされているネットワークハードウェア 」を参照してください。
熱要件	AWS Snowball Edge デバイスはオフィスの運用用に設計されており、データセンターの運用に最適です。
デシベル出力	平均して、AWS Snowball Edge デバイスは 68 デシベルの音を生成し、通常はバキュームクリーナーやリビングルームの音楽よりも静かです。
寸法と重量の仕様	
(重量)	49.7 ポンド (22.54 Kg)
高さ	15.5 インチ (394 mm)
幅	10.6 インチ (265 mm)
長さ。	28.3 インチ (718 mm)
環境仕様	

項目	Snowball Edge Storage Optimized (EC2 搭載) の仕様
振動	ASTM D4169トラックレベル I 0.73 GRMS に相当する非オペレーション上の使用
衝撃	70 G (MIL-S-901) に相当するオペレーション上の使用 50 G (ISTA-3A) に相当する非オペレーション上の使用
高度	0 ~ 3,000 メートル (0 ~ 10,000 フィート) に相当するオペレーション上の使用 0 ~ 12,000 メートルに相当する非オペレーション上の使用
温度範囲	0 ~ 45°C (オペレーション上)

Snowball Edge Compute Optimized デバイスの仕様

項目	Snowball Edge Compute Optimized の仕様
コンピューティングとメモリの仕様	
CPU	最大 104 個の vCPU (52 個または 104 個の vCPU の構成が可能)
RAM	512 GB RAM (最大 416 GB RAM - お客様による使用が可能)
GPU	nVidia V100 (GPU 搭載のコンピューティング最適化で使用可能 - 52 vCPU でのみ提供)
ストレージ仕様	
SSD ストレージ容量	28 TB の NVMe SSD または 42 TB の HDD (39.5 TB 使用可能)
電源仕様	

項目	Snowball Edge Compute Optimized の仕様
電源	AWS リージョン 米国の : NEMA 5 ~ 15p 100 ~ 220 ボルト。すべての AWS リージョンで、電源ケーブルが同梱されます。
消費電力	平均的なユースケースは 304 ワットですが、電源の定格は 1,200 ワットです。
電圧	100 ~ 240V AC
頻度	47/63 Hz
データおよびネットワーク接続	2x 10 Gbit – RJ45 (1 つ使用可能) 1x 25 Gbit – SFP28 1x 100 Gbit – QSFP28
ケーブル	各 AWS Snowball Edge デバイスには、国固有の電源ケーブルが付属しています。その他の付属ケーブルやオプティクスは付属していません。詳細については、「 サポートされているネットワークハードウェア 」を参照してください。
熱要件	AWS Snowball Edge デバイスはオフィスの運用用に設計されており、データセンターの運用に最適です。
デシベル出力	平均して、AWS Snowball Edge デバイスは 68 デシベルの音を生成し、通常はバキュームクリーナーやリビングルームの音楽よりも静かです。
寸法と重量の仕様	
(重量)	49.7 ポンド (22.54 Kg)
高さ	15.5 インチ (394 mm)
幅	10.6 インチ (265 mm)

項目	Snowball Edge Compute Optimized の仕様
長さ。	28.3 インチ (718 mm)
環境仕様	
振動	ASTM D4169トラックレベル I 0.73 GRMS に相当する非オペレーション上の使用
衝撃	70 G (MIL-S-901) に相当するオペレーション上の使用 50 G (ISTA-3A) に相当する非オペレーション上の使用
高度	0~3,000メートル (0~10,000 フィート) に相当するオペレーション上の使用 0~12,000メートルに相当する非オペレーション上の使用
温度範囲	0~45°C (オペレーション上)

サポートされているネットワークハードウェア

AWS Snowball Edge デバイスを使用するには、独自のネットワークケーブルが必要です。RJ45ケーブルには、特定の推奨はありません。Mellanox および Finisar の SFP+ ケーブル、QSFP+ ケーブルおよびモジュールはデバイスとの互換性を検証済みです。

AWS Snowball Edge デバイスのバックパネルを開くと、次のスクリーンショットに示すポートと同様のネットワークポートが表示されます。



デバイスで一度に使用できるネットワークインターフェイス AWS Snowball Edge は 1 つだけです。そのため、以下のネットワークハードウェアをサポートするには、いずれか 1 つのポートを使用してください。

SFP

このポートは、SFP28 および SFP+ トランシーバー・モジュール、および直接接続カッパー (DAC) ケーブルと互換性のある 10G/25G SFP28 インターフェイスを提供します。トランシーバーまたは DAC ケーブルはお客様が用意する必要があります。

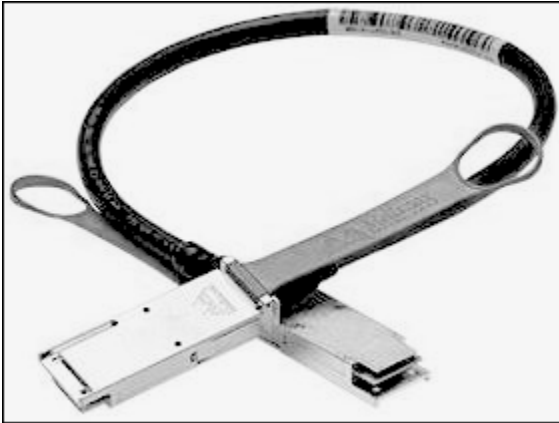
- 10G オペレーションの場合は、すべての SFP+ オプションを使用できます。その例を以下に示します。
 - 10Gbase-LR (シングルモードファイバー) トランシーバー
 - 10Gbase-SR (マルチモードファイバー) トランシーバー
 - SFP+ DAC ケーブル
- 25G オペレーションの場合は、すべての SFP28 オプションを使用できます。その例を以下に示します。
 - 25Gbase-LR (シングルモードファイバー) トランシーバー
 - 25Gbase-SR (マルチモードファイバー) トランシーバー
 - SFP28 DAC ケーブル



QSFP

このポートでは、ストレージ最適化デバイスに 40G QSFP+ インターフェイスが提供され、コンピューティング最適化デバイスに 40/50/100G QSFP+ インターフェイスが提供されます。どちらも、QSFP+ トランシーバーモジュールおよび DAC ケーブルと互換性があります。トランシーバーまたは DAC ケーブルはお客様が用意する必要があります。次に例を示します。

- 40Gbase-LR4 (シングルモードファイバー) トランシーバー
- 40Gbase-SR4 (マルチモードファイバー) トランシーバー
- QSFP+ DAC

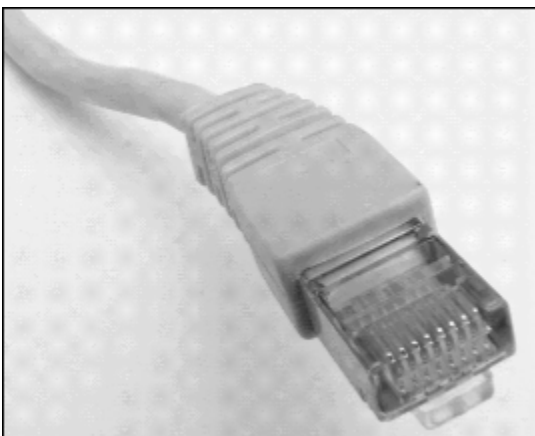


RJ45

このポートは 1Gbase-TX/10Gbase-TX オペレーションを提供します。これは終端が RJ45 コネクタの UTP ケーブル経由で接続します。Snowball Edge デバイスには 2 つの RJ45 ポートがあります。使用するポートを 1 つ選択します。

1G オペレーションは点滅するオレンジ色のライトで示されます。1G オペレーションは、データ転送にかかる時間が大幅に増加するため、Snowball Edge デバイスへの大規模なデータ転送には推奨されません。

10G オペレーションは点滅する緑色のライトで示されます。最大動作距離 180 フィート (55 メートル) の Cat6A UTP ケーブルが必要です。



Snow Family デバイスを使用するための前提条件

Snow Family デバイスの使用を開始する前に、AWS アカウントをお持ちでない場合はサインアップする必要があります。また、Snow Family デバイスで使用するデータインスタンスとコンピューティングインスタンスを設定する方法についても学習することをお勧めします。

AWS Snowball Edge はリージョン固有のサービスです。そのため、ジョブを計画する前に、そのサービスがお客様の AWS リージョンで利用可能であることを確認してください。ロケーションと Amazon S3 バケットが同じ AWS リージョン または同じ国にあることを確認します。これは、デバイスの注文能力に影響するためです。

ローカルエッジコンピューティングとストレージジョブ用にコンピューティング最適化されたデバイスで、Snow Family デバイスの Amazon S3 互換ストレージを使用するには、注文時にデバイスに S3 容量をプロビジョニングする必要があります。Snow Family デバイスの Amazon S3 互換ストレージは、ローカルバケット管理をサポートしているため、デバイスを受け取った後、デバイスまたはクラスターに S3 バケットを作成できます。

注文プロセスの一環として、AWS Identity and Access Management (IAM) ロールと AWS Key Management Service (AWS KMS) キーを作成します。KMS キーは、ジョブのロック解除コードの暗号化に使用されます。IAM ロールと KMS キーの作成の詳細については、「[Snow Family デバイスを注文するジョブの作成](#)」を参照してください。

Note

アジアパシフィック (ムンバイ) では、Amazon on Internet Services Private Limited (AISPL) によって AWS リージョン サービスが提供されます。アジアパシフィック (ムンバイ) で Amazon Web Services にサインアップする方法については AWS リージョン、「[AISPL にサインアップする](#)」を参照してください。

トピック

- [にサインアップする AWS アカウント](#)
- [管理アクセスを持つユーザーを作成する](#)
- [ローカル環境に関する質問](#)
- [特殊文字を含むファイル名の処理](#)
- [を使用した Amazon S3 暗号化 AWS KMS](#)
- [サーバー側の暗号化による Amazon S3 の暗号化](#)

- [Snow Family デバイスで Amazon S3 アダプターをインポートジョブ/エクスポートジョブに使用するための前提条件](#)
- [Snow Family デバイスの Amazon S3 互換ストレージを使用するための前提条件](#)
- [Snow Family デバイスでコンピューティングインスタンスを使用するための前提条件](#)

にサインアップする AWS アカウント

がない場合は AWS アカウント、次の手順を実行して作成します。

にサインアップするには AWS アカウント

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

サインアップ手順の一環として、通話呼び出しを受け取り、電話キーパッドで検証コードを入力するように求められます。

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。セキュリティのベストプラクティスとして、ユーザーに管理アクセスを割り当て、ルートユーザーのみを使用して[ルートユーザーアクセスが必要なタスク](#)を実行してください。

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 [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、 を保護し AWS アカウントのルートユーザー、 を有効にして AWS IAM Identity Center、日常的なタスクにルートユーザーを使用しないように管理ユーザーを作成します。

のセキュリティ保護 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「[ルートユーザーの仮想 MFA デバイスを有効にする \(コンソール\)](#)」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、「[ユーザーガイド](#)」の「[デフォルトで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](#)」AWS IAM Identity Center」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 「アクセスポータルへのサインイン」](#) 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「[グループの参加](#)」を参照してください。

ローカル環境に関する質問

データセットとローカル環境のセットアップについて理解しておく、データ転送を実行する際に役立ちます。ご注文の前に、次の点について考慮してください。

どのようなデータを転送するか。

AWS Snowball Edge では、小さなファイルを大量に転送しようとするすると正常に動作しません。Snowball Edge では、個々のオブジェクトが暗号化されるためです。小さなファイルとは、1 MB 未満のファイルなどのことです。Edge AWS Snowball デバイスに転送する前に、圧縮することをお勧めします。また、各ディレクトリ内のファイルやディレクトリの数は、500,000 を超えないようにすることをお勧めします。

転送中のデータにアクセスするか。

静的なデータセットであることが重要です (転送中にユーザーやシステムによってデータにアクセスされることがないこと)。そうでない場合、チェックサムの不一致によってファイルの転送が失敗する可能性があります。ファイルは転送されず、Failed のようにマークされます。

データの破損を防ぐため、データ転送中に AWS Snowball Edge デバイスを切断したり、ネットワーク設定を変更したりしないでください。ファイルは、デバイスへの書き込み中、静的な状態になっている必要があります。デバイスへの書き込み中にファイルが変更されると、読み取り/書き込み競合が発生する可能性があります。

ネットワークは AWS Snowball データ転送をサポートしますか？

Snowball Edge は、RJ45、SFP+、または QSFP+ ネットワークアダプタをサポートしています。スイッチがギガビットスイッチであることを確認してください。スイッチのブランドによりませんが、ギガビットまたは 10/100/1000 などと表示されています。Snowball Edge デバイスは、メガビットスイッチまたは 10/100 スイッチをサポートしていません。

特殊文字を含むファイル名の処理

オブジェクトの名前に特殊文字が含まれていると、エラーが発生する可能性があることに注意してください。Amazon S3 では特殊文字を使用できますが、次の文字の使用は避けることを強くお勧めします。

- バックスラッシュ ("\")
- 左中括弧 ("{"
- 右中括弧 ("}")
- 左角括弧 ("["
- 右角括弧 ("]")
- 小なり記号 ("<")
- 大なり記号 (">")
- 表示不可能な ASCII 文字 (10 進数の 128 ~ 255 の文字)
- カレット ("^")
- パーセント記号 ("%")
- アクサングラーブ/バックティック ("`")
- 引用符
- チルダ ("~")
- シャープ記号 ("#")
- 縦棒/パイプ ("|")

ファイルにオブジェクト名にこれらの文字が 1 つ以上含まれている場合は、オブジェクトの名前を変更してから AWS Snowball Edge デバイスにコピーします。ファイル名にスペースが含まれている Windows ユーザーは、個々のオブジェクトをコピーしたり、再帰コマンドを実行したりする際に注意が必要です。コマンドでは、名前にスペースを含むオブジェクトの名前を引用符で囲みます。このようなファイルの例を以下に示します。

オペレーティングシステム	ファイル名: test file.txt
Windows	"C:\Users\ <username>\desktop\test file.txt"</username>
iOS	/Users/<username>/test\ file.txt
Linux	/home/<username>/test\ file.txt

Note

転送されるオブジェクトメタデータは、オブジェクト名とサイズのみです。

を使用した Amazon S3 暗号化 AWS KMS

デフォルトの AWS マネージド暗号化キーまたはカスターマネージド暗号化キーを使用して、データのインポートまたはエクスポート時にデータを保護できます。

AWS KMS マネージドキーでの Amazon S3 デフォルトバケット暗号化の使用

で AWS マネージド暗号化を有効にするには AWS KMS

1. <https://console.aws.amazon.com/s3/>でAmazon S3 コンソールを開きます。
2. 暗号化する Amazon S3 バケットを選択します。
3. 右側に表示されるウィザードで、[Properties] (プロパティ)を選択します。
4. [Default encryption] (デフォルトの暗号化) ボックスで、[Disabled] (無効)を選択し (このオプションは灰色で表示されます)、デフォルトの暗号化を有効にします。
5. 暗号化方法として [AWS-KMS] を選択してから、使用する KMS キーを選択します このキーは、バケットに配置されたオブジェクトを暗号化するために使用されます。
6. [保存] を選択します。

Snowball Edge ジョブを作成したら、データをインポートする前に既存の IAM ロールポリシーにステートメントを追加します。これは、注文プロセス中に作成したロールです。デフォルトのロール名は、ジョブタイプに応じて Snowball-import-s3-only-role または Snowball-export-s3-only-role のようになります。

以下は、このようなステートメントの使用例です。

データをインポートする場合

AWS KMS マネージドキーによるサーバー側の暗号化 (SSE-KMS) を使用してインポートジョブに関連付けられた Amazon S3 バケットを暗号化する場合は、IAM ロールに次のステートメントも追加する必要があります。

Example Snowball インポート用の IAM ロールの例

```
{
  "Effect": "Allow",
  "Action": [
    "kms: GenerateDataKey",
    "kms: Decrypt"
  ]
}
```

```
  ],
  "Resource": "arn:aws:kms:us-west-2:123456789012:key/abc123a1-abcd-1234-efgh-111111111111"
}
```

データをエクスポートする場合

AWS KMS マネージドキーによるサーバー側の暗号化を使用してエクスポートジョブに関連付けられた Amazon S3 バケットを暗号化する場合は、IAM ロールに次のステートメントも追加する必要があります。

Example Snowball エクスポート用の IAM ロール

```
{
  "Effect": "Allow",
  "Action": [
    "kms:Decrypt"
  ],
  "Resource": "arn:aws:kms:us-west-2:123456789012:key/abc123a1-abcd-1234-efgh-111111111111"
}
```

AWS KMS カスタマーキーでの S3 デフォルトバケット暗号化の使用

独自の KMS キーでデフォルトの Amazon S3 バケット暗号化を使用して、インポートおよびエクスポートするデータを保護できます。

データをインポートする場合

でカスターマネージド暗号化を有効にするには AWS KMS

1. にサインイン AWS Management Console し、<https://console.aws.amazon.com/kms> で AWS Key Management Service (AWS KMS) コンソールを開きます。
2. を変更するには AWS リージョン、ページの右上隅にあるリージョンセレクターを使用します。
3. 左側のナビゲーションペインで [Customer managed keys] (カスターマネージド型キー) を選択し、使用するバケットに関連付けられている KMS キーを選択します。
4. まだ展開されていない場合、[Key Policy] (キーポリシー) を展開します。
5. [Key Users] (キーユーザー) セクションで [Add] (追加) を選択し、IAM ロールを検索します。IAM ロールを選択し、[Add] (追加) を選択します。

6. または、[Switch to Policy view] (ポリシービューへの切り替え) を選択してキーポリシードキュメントを表示し、キーポリシーにステートメントを追加することもできます。以下はポリシーの例です。

Example AWS KMS カスタマーマネージドキーのポリシーの

```
{
  "Sid": "Allow use of the key",
  "Effect": "Allow",
  "Principal": {
    "AWS": [
      "arn:aws:iam::111122223333:role/snowball-import-s3-only-role"
    ]
  },
  "Action": [
    "kms:Decrypt",
    "kms:GenerateDataKey"
  ],
  "Resource": "*"
}
```

このポリシーが AWS KMS カスタマーマネージドキーに追加されると、Snowball ジョブに関連付けられた IAM ロールも更新する必要があります。デフォルトのロールは snowball-import-s3-only-role です。

Example Snowball インポート用の IAM ロール

```
{
  "Effect": "Allow",
  "Action": [
    "kms:GenerateDataKey",
    "kms:Decrypt"
  ],
  "Resource": "arn:aws:kms:us-west-2:123456789012:key/abc123a1-abcd-1234-efgh-111111111111"
}
```

詳細については、「[でのアイデンティティベースのポリシー \(IAM ポリシー\) の使用 AWS Snowball](#)」を参照してください。

使用されている KMS キーは、次のように表示されます。

```
"Resource": "arn:aws:kms:region:AccountID:key/*"
```

データをエクスポートする場合

Example AWS KMS カスタマーマネージドキーのポリシーの

```
{
  "Sid": "Allow use of the key",
  "Effect": "Allow",
  "Principal": {
    "AWS": [
      "arn:aws:iam::111122223333:role/snowball-import-s3-only-role"
    ]
  },
  "Action": [
    "kms:Decrypt",
    "kms:GenerateDataKey"
  ],
  "Resource": "*"
}
```

このポリシーが AWS KMS カスタマーマネージドキーに追加されると、Snowball ジョブに関連付けられた IAM ロールも更新する必要があります。デフォルトのロールは次のように表示されます。

snowball-export-s3-only-role

Example Snowball エクスポート用の IAM ロール

```
{
  "Effect": "Allow",
  "Action": [
    "kms:GenerateDataKey",
    "kms:Decrypt"
  ],
  "Resource": "arn:aws:kms:us-west-2:123456789012:key/abc123a1-abcd-1234-efgh-111111111111"
}
```

このポリシーが AWS KMS カスタマーマネージドキーに追加されると、Snowball ジョブに関連付けられた IAM ロールも更新する必要があります。デフォルトのロールは snowball-export-s3-only-role です。

サーバー側の暗号化による Amazon S3 の暗号化

AWS Snowball は、Amazon S3 マネージド暗号化キー (SSE-S3) によるサーバー側の暗号化をサポートします。サーバー側の暗号化は、保管中のデータを保護するものです。SSE-S3 では、Amazon S3 に保管中のデータを保護するための強力な多要素暗号化を採用しています。詳細については、Amazon Simple Storage Service ユーザーガイドの [Amazon S3 が管理する暗号化キーによるサーバー側の暗号化 \(SSE-S3\) を使用したデータの保護](#) を参照してください。

Note

現在、はお客様が用意したキーによるサーバー側の暗号化 (SSE-C) をサポート AWS Snowball していません。ただし、その SSE タイプを使用して、インポートされたデータを保護したい場合や、エクスポートするデータにすでにこのタイプを使用している場合があります。このような場合は、以下の点に注意してください。

- インポート - S3 にインポートしたオブジェクトを SSE-C で暗号化する場合は、それらのオブジェクトを別のバケットにコピーします。コピー先のバケットでは、バケットポリシーの一部として SSE-KMS または SSE-S3 暗号化が設定されているものとします。
- エクスポート - SSE-C で暗号化されたオブジェクトをエクスポートする場合は、最初にそれらのオブジェクトを別のバケットにコピーします。コピー先のバケットでは、バケットポリシーにサーバー側の暗号化が指定されていないか、SSE-KMS または SSE-S3 が指定されているものとします。

Snow Family デバイスで Amazon S3 アダプターをインポートジョブ/エクスポートジョブに使用するための前提条件

Snow Family デバイスで S3 アダプターを使用すると、デバイスを使用してオンプレミスのデータソースからクラウドまたはクラウドからオンプレミスのデータストレージにデータを移動できます。詳細については、「[データ移行に Amazon S3 アダプターを使用してファイルを転送する](#)」を参照してください。

ジョブに関連付けられた Amazon S3 バケットでは、Amazon S3 Standard ストレージクラスを使用する必要があります。最初のジョブを作成する前に、以下の点に留意してください。

Amazon S3 にデータをインポートするジョブについては、以下のステップに従ってください。

- 転送するファイルやフォルダは、Amazon S3 の [オブジェクトキーの命名のガイドライン](#) に従って命名されます。このガイドラインを満たしていない名前のファイルやフォルダは、Amazon S3 にインポートされません。
- Amazon S3 にどのデータをインポートするのかをプランニングします。詳細については、「[大規模な転送の計画](#)」を参照してください。

Amazon S3 からデータをエクスポートする前に、以下のステップに従ってください。

- ジョブの作成時に、エクスポートされるデータを確認します。詳細については、「[エクスポート範囲の使用](#)」を参照してください。
- ファイル名にコロン (:) を含むファイルがある場合は、Amazon S3 でファイル名を変更してから、これらを取得するエクスポートジョブを作成します。ファイル名にコロンが含まれている場合は、ファイルを Microsoft Windows Server にエクスポートすることはできません。

Snow Family デバイスの Amazon S3 互換ストレージを使用するための前提条件

Snow Family デバイスの Amazon S3 互換ストレージは、エッジロケーションのデバイスにデータを保存し、そのデータをローカルコンピューティングオペレーションに使用するときを使用します。ローカルコンピューティングオペレーションに使用されるデータは、デバイスが返されたときに Amazon S3 にインポートされません。

ローカルコンピューティング用および Amazon S3 互換ストレージを備えたストレージ用の Snow デバイスを注文する場合は、次の点に注意してください。

- デバイスの注文時に、Amazon S3 ストレージ容量をプロビジョニングします。そのため、デバイスを注文する前に、ストレージのニーズを検討してください。
- Amazon S3 バケットは、Snow Family デバイスの注文時ではなく、受領後にデバイス上で作成できます。
- Snow Family デバイスで Amazon S3 互換ストレージを使用するには、最新バージョンの AWS CLI (v2.11.15 以降)、Snowball Edge クライアント、または をダウンロード AWS OpsHub し、コンピュータにインストールする必要があります。Amazon S3
- デバイスを受け取ったら、本ガイドの「[Snow Family デバイスの Amazon S3 互換ストレージを使用する](#)」に従って、Snow Family デバイスの Amazon S3 互換ストレージを設定、起動、使用します。

Snow Family デバイスでコンピューティングインスタンスを使用するための前提条件

Edge でホストされている Amazon EC2-compatibleコンピューティングインスタンスは sbe1、sbe-c、および sbe-g インスタンスタイプ AWS Snowball で実行できます。

- sbe1 インスタンスタイプは、Snowball Edge Storage Optimized オプションを備えたデバイスで動作します。
- sbe-c インスタンスタイプは、Snowball Edge Compute Optimized オプションを備えたデバイスで動作します。
- sbe-c および sbe-g インスタンスタイプはいずれも、GPU オプション付きの Snowball Edge Compute Optimized を備えたデバイスで動作します。

Snowball Edge デバイスのオプションでサポートされるすべてのコンピューティングインスタンスタイプは、AWS Snowball Edge デバイスに固有です。これらのインスタンスでは、クラウドベースのインスタンスと同様に、Amazon Machine Image (AMI) を起動する必要があります。インスタンスの AMI を選択してから、Snowball Edge ジョブを作成します。

Snowball Edge でコンピューティングインスタンスを使用するには、Snow Family デバイスを注文するジョブを作成し、AMIs を指定します。これを行うには、AWS Snowball マネジメントコンソール、AWS Command Line Interface (AWS CLI)、または AWS SDKs のいずれかを使用します。一般的に、インスタンスを使用するには、ジョブの作成前にハウスキープिंगの前提条件をいくつか実行する必要があります。

コンピューティングインスタンスを使用するジョブでは、ジョブに任意の AMI を追加する前に、サポートされるイメージタイプの AMI が AWS アカウントにある必要があります。現在、サポートされている AMI は以下のオペレーティングシステムに基づいています。

- [Amazon Linux 2](#)
- [CentOS 7 \(x86_64\) - with Updates HVM](#)
- Ubuntu 16.04 LTS - Xenial (HVM)
- [Ubuntu 20.04 LTS - Focal](#)
- [Ubuntu 22.04 LTS - Jammy](#)
- [Microsoft Windows Server 2012 R2](#)
- [Microsoft Windows Server 2016](#)
- [Microsoft Windows Server 2019](#)

Note

Ubuntu 16.04 LTS - Xenial (HVM) イメージは、ではサポートされなくなりましたが AWS Marketplace、Amazon EC2 VM Import/Export を介して Snowball Edge デバイスで使用でき、AMIs でローカルで実行できます。

これらのイメージは、[AWS Marketplace](#) から取得できます。

SSH を使用して Snowball Edge で実行中のインスタンスに接続している場合は、独自のキーペアを使用するか、Snowball Edge でキーペアを作成できます。AWS OpsHub を使用してデバイスにキーペアを作成するには、「」を参照してください[キーペアによる作業](#)。を使用してデバイスにキーペア AWS CLI を作成するには、「」の create-key-pair 「」を参照してください[Snowball Edge でサポートされる Amazon EC2 互換の AWS CLI コマンドのリスト](#)。キーペアと Amazon Linux 2 の詳細については、[Amazon EC2 ユーザーガイド](#) の「[Amazon EC2 キーペアと Linux インスタンス](#)」を参照してください。Amazon EC2

デバイスでコンピューティングインスタンスを使用する際の固有の情報については、「[Amazon EC2 互換コンピューティングインスタンスの使用](#)」を参照してください。

AWS Snowball Edge の仕組み

AWS Snowball エッジデバイスは、によって所有され AWS、使用中はオンプレミスの場所に存在します。

AWS Snowball Edge デバイスで使用できるジョブタイプは 3 つあります。ユースケースによってジョブのタイプが異なっても、各ジョブタイプでのデバイスの注文、受領、返却のワークフローは同じです。ジョブタイプにかかわらず、ジョブの完了後、すべてのジョブは、米国国立標準技術研究所 (NIST) 800-88 標準のデータ消去に従います。

共有ワークフロー

1. ジョブの作成 - 各ジョブは AWS Snow ファミリーマネジメントコンソール で作成するか、ジョブ管理 API を介してプログラムで作成します。ジョブのステータスは、コンソールまたは API を介して追跡できます。
2. ジョブ用のデバイスの準備 - お客様のジョブ用の AWS Snowball Edge デバイスが用意され、ジョブのステータスが [Snowball の準備中] になります。
3. リージョンのキャリアによるデバイスの配送 - キャリアによって集荷され、ジョブのステータスが [In transit to you] (お客様へ配送中) になります。コンソールまたはジョブ管理 API で、追跡番号と追跡ウェブサイトへのリンクを確認できます。お客様のリージョンのキャリアについての詳細は、[Snow Family デバイスの配送に関する考慮事項](#) を参照してください。
4. デバイスの受信 - 数日後、リージョンのキャリアは、ジョブの作成時に指定したアドレスに AWS Snowball Edge デバイスを配信し、ジョブのステータスが 配信済み に変わります。デバイスが到着すると、梱包箱がないことがわかります。それ自体が配送用コンテナであるためです。
5. 認証情報の取得と Snowball Edge クライアントのダウンロード - 認証情報、ジョブマニフェスト、マニフェストの解除コードを取得し、Snowball Edge クライアントをダウンロードして、データ転送を開始する準備を整えます。
 - Snowball Edge クライアントは、デバイスからオンプレミスのデータ送信先へのデータの流れの管理に使用するツールです。

Snowball Edge クライアントは、[AWS Snowball リソース](#) ページからダウンロードできます。

「[AWS Snowball Edge のリソース](#)」ページから Snowball Edge クライアントをダウンロードし、お客様が所有する強力なワークステーションにインストールする必要があります。

- マニフェストは、デバイスへのアクセスの認証に使用されます。暗号化されていて解除コードでのみ復号できます。マニフェストは、コンソールから取得するか、デバイスがお客様のオンプレミスの場所にあるときにジョブ管理 API で取得できます。

- 解除コードは、マニフェストの復号に使用される 29 文字のコードです。解除コードは、コンソールから、またはジョブ管理 API を使用して、取得できます。デバイスがお客様の施設にある間の不正アクセスを防ぐため、解除コードをマニフェストとは別の場所に保存しておくことをお勧めします。
- 6. ハードウェアの配置 – デバイスをデータセンターに移動し、ケース上に記載された指示に従って開きます。デバイスを電源とローカルネットワークに接続します。
- 7. デバイスの電源を入れる – 次に、LCD ディスプレイの上にある電源ボタンを押して、デバイスの電源をオンにします。数分後に [Ready] 画面が表示されます。
- 8. デバイスの IP アドレスの取得 – LCD ディスプレイに [CONNECTION] (接続) タブがあります。このタブをタップし、AWS Snowball Edge デバイスの IP アドレスを取得します。
- 9. Snowball Edge クライアントを使用してデバイスのロックを解除する – Snowball Edge クライアントを使用して AWS Snowball Edge デバイスのロックを解除する場合は、デバイスの IP アドレス、マニフェストへのパス、およびロック解除コードを入力します。Snowball Edge クライアントによってマニフェストが復号され、これを使ってデバイスへのアクセスが認証されます。
- 10. デバイスの使用 – デバイスが稼働します。Amazon S3 アダプターまたはネットワークファイルシステム (NFS) マウントポイントを使用してデータを転送できます。または、Snow Family デバイスの Amazon S3 互換ストレージでのローカルコンピューティングとストレージに使用できます。
- 11. デバイスを返送する準備をする – オンプレミスの場所でデバイスを操作したら、LCD ディスプレイの上にある電源ボタンを押します。約 20 秒程度でデバイスの電源がオフになります。デバイスを切り離し、電源ケーブルをデバイス上部のケーブルフックに収納して、デバイスの 3 つのドアをすべて閉じます。デバイスの返却準備完了です。
- 12. リージョンの通信事業者がデバイスを に返 AWSす – 通信事業者が AWS Snowball Edge デバイスを持っている場合、ジョブのステータスは への転送 AWS中になります。

Note

エクスポートおよびクラスタージョブには追加のステップがあります。詳細については、「[エクスポートジョブの仕組み](#)」および「[クラスター化されたローカルコンピューティングおよびストレージジョブの仕組み](#)」を参照してください。

トピック

- [インポートジョブの仕組み](#)
- [エクスポートジョブの仕組み](#)
- [ローカルコンピューティングおよびストレージジョブの仕組み](#)

- [Snowball Edge の動画とブログ](#)

インポートジョブの仕組み

各インポートジョブごとに 1 つの Snowball アプライアンスを使用します。AWS Snow ファミリー マネジメントコンソール またはジョブ管理 API で Snow Family デバイスを注文するジョブを作成すると、Snowball がお客様に配送されます。数日後に Snowball がお手元に届いたら、Snowball Edge デバイスをネットワークに接続して、Amazon S3 にインポートするデータをデバイスに転送します。データ転送が完了したら、Snowball を に返送し AWS、Amazon S3 にデータをインポートします。

エクスポートジョブの仕組み

各エクスポートジョブは、任意の数の AWS Snowball Edge デバイスを使用できます。リストに含まれるデータが 1 台のデバイスに収まる数を超える場合は、複数のデバイスがお客様に提供されます。各ジョブパートに関連付けられるデバイスは 1 つのみです。ジョブパートが作成されると、最初のジョブパートのステータスが [Preparing Snowball] (Snowball の準備中) になります。

Note

ジョブをパートに分割するリストオペレーションは Amazon S3 の機能です。他の Amazon S3 オペレーションと同様に料金が発生します。

その後すぐに、デバイスへのデータのエクスポートが開始されます。データのエクスポートにかかる時間は、データセットの特性によって異なります。例えば、多くの小さいファイル (10 MB 未満) をエクスポートする場合は、かなり長い時間がかかります。エクスポートが完了すると、 はリージョン AWS のキャリアによるデバイスの集荷準備をします。到着したら、AWS AWS Snowball Edge デバイスをネットワークに接続し、デバイスからネットワーク上のストレージにデータを転送します。

データ転送が完了したら、デバイスを に返送します AWS。当社は、エクスポートジョブパートの デバイスを受領後、データを完全に消去します。この消去作業は National Institute of Standards and Technology (NIST) 800-88 基準に準拠しています。このステップにより、特定のジョブパートが完了します。

- キーリストの場合

S3 バケット内のオブジェクトをエクスポートする前に、バケットをスキャンします。スキャン後にバケットが変更された場合、欠落または変更されたオブジェクトをスキャンするため、ジョブに遅延が発生する可能性があります。

- S3 Glacier Flexible Retrieval の場合

は S3 Glacier ストレージクラスにあるオブジェクトをエクスポート AWS Snowball できないことに注意してください。これらのオブジェクトは、AWS Snowball がバケット内のオブジェクトを正常にエクスポートする前に復元する必要があります。

ローカルコンピューティングおよびストレージジョブの仕組み

Amazon EKS Anywhere on Snow で AWS EC2-compatible コンピューティングインスタンスまたは Kubernetes コンテナを実行することで、AWS Snowball Edge デバイスのローカルコンピューティングおよびストレージ機能を使用できます。コンピューティング機能には、Snow Family デバイスの Amazon S3 互換ストレージによってデータストレージが提供されます。

Snowball Edge デバイスで Amazon S3 バケットを作成して、ローカルデータアクセス、ローカルデータ処理、データレジデンシーを必要とするアプリケーション用に、オンプレミスのオブジェクトを保存および取得できます。Snow Family デバイスの Amazon S3 互換ストレージは、SNOW という新しいストレージクラスを提供します。このストレージクラスは、S3 API を使用し、複数の Snowball Edge デバイス間でデータを永続的かつ冗長的に保存するように設計されています。Snowball Edge バケットで、Amazon S3 と同じ API と機能 (バケットのライフサイクルポリシー、暗号化、タグ付けなど) を使用できます。デバイスまたはデバイスが返されると AWS、Snow Family デバイスの Amazon S3 互換ストレージに作成または保存されたすべてのデータが消去されます。詳細については、[「ローカルコンピューティングとストレージ専用ジョブ」](#)を参照してください。

詳細については、[「ローカルコンピューティングおよびストレージ専用ジョブ」](#)を参照してください。

クラスター化されたローカルコンピューティングおよびストレージジョブの仕組み

クラスタージョブは、ローカルストレージおよびコンピューティング専用の特別な種類のジョブです。これは、データの耐久性を高め、ストレージ容量を増やす必要のあるワークロード用です。詳細については、[「ローカルクラスターオプション」](#)を参照してください。

Note

スタンドアロンのローカルストレージおよびコンピューティングのジョブの場合と同様に、クラスターに保存されているデータを Amazon S3 にインポートするには、個別のインポートジョブの一部として追加のデバイスを注文する必要があります。これらのデバイスを注文する場合、インポートジョブのデバイスを返送する際に、クラスターのデータをそのデバイスに転送し、データをインポートすることができます。

クラスターには、ノードと呼ばれる 3~16 AWS Snowball Edge デバイスがあります。リージョンのキャリアからノードを受け取ったら、すべてのノードを電源とネットワークに接続して IP アドレスを取得します。これらの IP アドレスを使用して、いずれかのノードの IP アドレスを使用し、クラスターのノードを 1 つのロック解除コマンドですべて同時にロック解除します。詳細については、「[Snowball Edge クライアントコマンドの使用](#)」を参照してください。

Snow Family デバイスの Amazon S3 互換ストレージと、他のノードに分散されたデータを使用して、ロック解除されたクラスターにデータを書き込むことができます。

クラスターが完了したら、すべてのノードを に返送します AWS。返送されたクラスターノードを受領したら、当社が Snowball を完全に消去します。この消去作業は National Institute of Standards and Technology (NIST) 800-88 基準に準拠しています。

Snowball Edge の動画とブログ

- [snow-transfer-tool AWS Snowball Edge デバイスの との混合ファイルサイズの移行](#)
- [AWS Snowball エッジデータ移行](#)
- [AWS OpsHub for Snow Family](#)
- [Novetta が、災害対応のための IoT と機械学習をエッジに提供](#)
- [DMS と を使用して大規模なデータベース移行を有効にする AWS Snowball](#)
- [によるデータ移行のベストプラクティス AWS Snowball Edge](#)
- [AWS Snowball resources](#)
- [AWS Snowball エッジコンピューティング最適化デバイスの Amazon S3 互換ストレージが一般利用可能に](#)
- [AWS Snowball Edge デバイスの Snow Family デバイスの Amazon S3 互換ストレージの使用を開始する](#)

Snowball Edge デバイスの長期料金

Snowball Edge デバイスを注文する際、ユースケースに最適な料金オプションを選択できます。2つの料金設定があります。1つはデバイスを保持する日単位のオンデマンド料金で、もう1つはデバイスの種類に応じて月単位、1年または3年単位で設定される前払いの長期料金です。デバイスの使用が中断されないように、1年または3年単位で長期料金オプションを自動的に更新することができます。これにより、前の期間が終了すると新しい前払い期間が開始されます。月額長期料金オプションは、デバイスを保持する間は自動的に更新されます。デバイスの注文の詳細については、このガイドの「[Snow Family デバイスを注文するジョブの作成](#)」を参照してください。

予算面での利便性に加えて、長期料金オプションでは、運用要件の変化に応じて料金期間中に Snowball Edge デバイスを交換できます。例えば、新しいデバイスに新しい AMI や Amazon S3 からの新しいデータが含まれるようにデバイスの交換を依頼したり、故障したデバイスの交換を依頼したりできます。[長期料金期間中のデバイスの交換](#) を参照してください。

Note

Snow サービスに起因するハードウェアまたはソフトウェアの問題以外の理由で、1年または3年のコミット料金プランで AWS Snowball Edge デバイスの交換または交換をリクエストした場合、デバイスサイクル料金が請求されます。このデバイスサイクリング料金は、お客様の設定に応じた月額料金 (Snowball Edge Compute Optimized の場合) またはオンデマンドジョブ料金として決定されます。

長期料金の詳細については、「[Optimizing cost with long-term pricing options for AWS Snowball](#)」を参照してください。の AWS Snowball 料金については AWS リージョン、「[の AWS Snowball 料金](#)」を参照してください。

長期料金期間中のデバイスの交換

長期料金期間中に Snowball Edge デバイスを交換するには、新しいデバイスを注文し、現在保持しているデバイスをすぐに返却する必要があります。

1. 交換用の Snowball Edge デバイス用の新しいジョブを作成します。交換するデバイスは、使用しているデバイスと同じジョブタイプで、コンピューティングオプションとストレージオプションが同じである必要があります。このガイドの「[Snow Family デバイスを注文するジョブの作成](#)」を参照してください。

2. お持ちのデバイスをすぐに返却します。AWS [Snowball Edge の電源をオフにする](#) 「」と「」を参照してください。[Snowball Edge デバイスの返却](#)。はデバイス交換のロジスティクスを管理し、この交換にはデバイスサイクル料金が課金されます。

Snow Family デバイスの配送に関する考慮事項

Snow Family デバイスを注文するジョブを作成するときは、配送先住所を指定し、配送速度を選択します。この配送速度は、ジョブを作成した日から デバイスを受領する日までの予定日数を示すものではありません。むしろ、デバイスが AWS と配送先住所の間で転送されている時間を示します。デバイスが出荷される前に、AWS はジョブのデバイスを処理します。ジョブの処理に必要な時間は、ジョブタイプや規模などの要因によって異なります。また、配送業者は通常、1日に1回のみ Snow Family デバイスを集荷し、配送業者は週末に発送用デバイスを集荷しません。したがって、配送前の処理に1日以上かけることが可能です。AWS がデバイスを出荷する準備をしている間、および返却後にデバイスを受け取った後、を通じてジョブのステータスをモニタリングできます AWS Snow ファミリーマネジメントコンソール。詳細については、「[ジョブステータス](#)」を参照してください。

Note

選択した配送速度は、AWS がデバイスをお客様に送信するときと、デバイスを に返送するときに適用されます AWS。

Snowball Edge デバイスは、デバイスが注文された AWS リージョン内のデータをインポートまたはエクスポートするためにのみ使用できます。

Snow Family デバイスを注文するジョブの作成時に配送速度を選択し、配送先住所を入力する方法の詳細については、「」を参照してください [ステップ 4: セキュリティ、配送、通知の設定を選択する](#)。Snow Family デバイスを に返す方法の詳細については、AWS 「」を参照してください [Snowball Edge デバイスの返却](#)。

配送料金の詳細については、[AWS Snowball Edge の料金](#)を参照してください。

リージョンごとの配送制限

Snow Family デバイスを注文するジョブを作成する前に、Amazon S3 データ AWS リージョンと同じからコンソールにサインインする必要があります。は、アジアパシフィック (インド) AWS リージョンからアジアパシフィック (オーストラリア) など、同じ内の国間で Snow Family デバイスを配送 AWS しません。


国間の配送の例外は、欧州連合 (EU) 加盟国のみです。欧州 AWS リージョンでのデータ転送の場合、以下の EU メンバー国にのみデバイスを配送します。

オーストリア、ベルギー、ブルガリア、クロアチア、キプロス、チェコ共和国、デンマーク、エストニア、フィンランド、フランス、ドイツ、ギリシャ、ハンガリー、イタリア、アイルランド、ラトビア、リトアニア、ルクセンブルク、マルタ、オランダ、ポーランド、ポルトガル、ルーマニア、スロバキア、スロベニア、スペイン、スウェーデン

Snow Family デバイスは、デバイスが注文されたのと同じ AWS リージョンにのみ返すことができます。

同じ国内への発送は許可されています。例:

- 英国リージョンでのデータ転送のために、英国国内でデバイスを配送する。
- アジアパシフィック (ムンバイ) でのデータ転送のために、インド国内でデバイスを発送する。

 Note

AWS は、Snow Family デバイスを私書箱に配送しません。

開始方法

AWS Snowball Edge デバイスを使用すると、インターネットへの接続がオプションではない可能性がある場所で、AWS クラウド ローカルおよびコスト効率の高い のストレージおよびコンピューティング能力にアクセスできます。また、オンプレミスのデータセンターと Amazon Simple Storage Service (Amazon S3) の間で数百テラバイトからペタバイトのデータを転送できます。

以下に、AWS Snow ファミリーマネジメントコンソールで最初の AWS Snowball Edge デバイス ジョブを作成して完了する一般的な手順を示します。コンソールには、ジョブタイプで分けられたもっとも一般的なワークフローが提示されます。AWS Snowball Edge デバイスの特定のコンポーネントの詳細については、このドキュメントで確認できます。サービス全体の概要については、「[AWS Snowball Edge の仕組み](#)」を参照してください。

入門演習では、を使用してジョブ AWS Snow ファミリーマネジメントコンソール を作成し、AWS OpsHub for Snow Family を使用して AWS Snowball Edge デバイスのロックを解除および管理し、Amazon S3 インターフェイスを使用してデータの読み取りと書き込みを行うことを前提としています。ジョブをプログラムで作成して、作成するジョブのオプションを増やしたい場合は、ジョブ管理 API を使用できます。詳細については、「[AWS Snowball API リファレンス](#)」を参照してください。

開始する前に、AWS Identity and Access Management (IAM) で AWS アカウント と管理者ユーザーを作成する必要があります。詳細については、「[Snow Family デバイスを使用するための前提条件](#)」を参照してください。

トピック

- [Snow Family デバイスを注文するジョブの作成](#)
- [によるジョブのキャンセル AWS Snow ファミリーマネジメントコンソール](#)
- [Snowball Edge の受領](#)
- [ローカルネットワークへの接続](#)
- [Snow Family デバイスにアクセスするための認証情報の取得](#)
- [Snowball Edge クライアントのダウンロードとインストール](#)
- [Snow Family デバイスのロック解除](#)
- [ローカルユーザーの設定](#)
- [Snow Family デバイスの再起動](#)
- [Snowball Edge の電源をオフにする](#)

- [Snowball Edge デバイスの返却](#)
- [Snow Family デバイスの返送](#)
- [インポートステータスのモニタリング](#)
- [ジョブ完了レポートとログの取得](#)

Snow Family デバイスを注文するジョブの作成

Snow Family デバイスを注文するには、 で Snow Family デバイスを注文するジョブを作成します。AWS Snow ファミリーマネジメントコンソール。ジョブとは、 がお客様による Snow Family デバイスの使用のライフサイクルを説明するために AWS 使用する用語です。ジョブは、デバイスの注文時に開始され、 がデバイスを AWS 準備してお客様に配送し、使用するときに行き続けられ、 がデバイスを返送した後にデバイス AWS を受信して処理した後に完了します。ジョブは、エクスポート、インポート、ローカルコンピューティングとストレージのタイプ別に分類されます。詳細については、「[AWS Snowball Edge ジョブについて](#)」を参照してください。

デバイスを注文するジョブを作成したら、 を使用してジョブのステータス AWS Snow ファミリーマネジメントコンソール を表示し、注文したデバイスの進捗状況をモニタリングできます。 は、デバイスからお客様に配送する AWS 準備をし、返却された後です。詳細については、「[ジョブステータス](#)」を参照してください。デバイスが によって返されて処理されたら AWS、 を介してジョブ完了レポートとログにアクセスできます AWS Snow ファミリーマネジメントコンソール。詳細については、「[コンソールでのジョブ完了レポートとログの取得](#)」を参照してください。

ジョブの作成および管理は、ジョブマネジメント API を使って行うことも可能です。詳細については、「[APIリファレンスAWS Snowball](#)」を参照してください。

トピック

- [ステップ 1: ジョブタイプを選択する](#)
- [ステップ 2: コンピューティングとストレージのオプションを選択する](#)
- [ステップ 3: 機能とオプションを選択する](#)
- [ステップ 4: セキュリティ、配送、通知の設定を選択する](#)
- [ステップ 5: ジョブの概要を確認してジョブを作成する](#)
- [ダウンロード AWS OpsHub](#)

ステップ 1: ジョブタイプを選択する

ジョブを作成する最初のステップは、必要なジョブのタイプを決定し、AWS Snow ファミリーマネジメントコンソールを使用して計画を始めることです。

ジョブタイプを選択するには

1. にサインインし AWS Management Console、を開きます [AWS Snow ファミリーマネジメントコンソール](#)。この でジョブを初めて作成する場合は AWS リージョン、AWS Snow Family ページが表示されます。それ以外の場合は、既存のジョブのリストが表示されます。
2. これが最初のジョブの場合は、AWS Snow Family デバイスの注文 を選択します。複数のジョブで 500 TB を超えるデータを移行することが予想される場合は、[500 TB を超える大規模データ移行プランを作成] を選択します。それ以外の場合は、左側のナビゲーションバーで [ジョブを作成] を選択します。[Next step] (次のステップ) を選択し、[Plan your job] (ジョブの計画) ページを開きます。
3. [ジョブに名前を付ける] セクションで、[ジョブ名] ボックスにジョブの名前を入力します。
4. ニーズに応じて、次のいずれかのジョブタイプを選択します。
 - Amazon S3 にインポートする — 空の Snowball Edge デバイスを に AWS 配送するには、このオプションを選択します。デバイスをローカルネットワークに接続し、Snowball Edge クライアントを実行します。NFS 共有または S3 アダプターを使用してデバイスにデータをコピーし、 に返送すると AWS、データは にアップロードされます AWS。
 - [Export from Amazon S3] (Amazon S3 からのエクスポート) — このオプションを選択すると、Amazon S3 バケットからデバイスにデータがエクスポートされます。AWS がデータをデバイスに読み込み、お客様に発送します。デバイスをローカルネットワークに接続し、Snowball Edge クライアントを実行します。デバイスからサーバーにデータをコピーします。完了したら、デバイスを に発送し AWS、データはデバイスから消去されます。
 - [ローカルコンピューティングおよびストレージのみ] – データを転送せずに、デバイス上でコンピューティングおよびストレージワークロードを実行します。

Choose a job type

- Import into Amazon S3** [Info](#)
AWS will ship an empty device to you for storage and compute workloads. You'll transfer your data onto it, and ship it back. After AWS gets it, your data will be moved.
- Export from Amazon S3** [Info](#)
Choose what data you want to export from your S3 buckets for storage and compute workloads. AWS will load that data onto a device and ship it to you. When you're done ship the device back for erasing.
- Local compute and storage only** [Info](#)
Perform local compute and storage workloads without transferring data. You can order multiple devices in a cluster for increased durability and storage capacity. Includes rugged and rack-mountable devices.

5. [次へ] を選択して続行します。

ステップ 2: コンピューティングとストレージのオプションを選択する

Snow Family デバイスのハードウェア仕様、使用する Amazon EC2 互換インスタンス、データの保存方法、料金を選択してください。

デバイスのコンピューティングオプションとストレージオプションを選択するには

1. [Snow デバイス] セクションで、注文する Snow Family デバイスを選択します。

Note

AWS リージョン 注文元のと選択したジョブタイプによっては、一部の Snow Family デバイスを使用できない場合があります。

Snow devices Info					
	Name	Compute	Memory	Storage (HDD)	Storage (SSD)
<input checked="" type="radio"/>	Snowcone	2 vCPUs	4 GB	8 TB	-
<input type="radio"/>	Snowcone SSD	2 vCPUs	4 GB	-	14 TB
<input type="radio"/>	Snowball Edge Compute Optimized	52 vCPUs	208 GB	39.5 TB	7.68 TB
<input type="radio"/>	Snowball Edge Compute Optimized with GPU	52 vCPUs, GPU	208 GB	39.5 TB	7.68 TB
<input type="radio"/>	Snowball Edge Compute Optimized	104 vCPUs	416 GB	-	28 TB

2. [料金オプションを選択] セクションの [料金オプションを選択] メニューから、このジョブに適用する料金の種類を選択します。1年または3年の確約前払い料金プランを選択した場合、[自動更新] で [オン] を選択すると、現在の期間終了時に料金プランが自動的に更新され、[オフ] を選択すると、現在の期間終了時に料金プランが自動的に更新されません。Snowball Edge デバイスの長期料金オプションの詳細については、このガイドの「[Snowball Edge デバイスの長期料金](#)」を参照してください。のデバイス料金については AWS リージョン、「[のAWS Snowball 料金](#)」を参照してください。
3. [ストレージタイプの選択] セクションで、ニーズに応じて以下を選択してください。
 - Amazon S3 アダプター: Amazon S3 アダプターを使用し、Amazon S3 REST API アクションを使用して Snow Family デバイスとの間でプログラムによりデータを転送します。
 - Amazon S3 互換ストレージ: Amazon S3 互換ストレージを使用して、S3 互換の耐久性とスケラブルなオブジェクトストレージを単一の Snowball Edge デバイスまたはマルチデバイスクラスターにデプロイします。
 - NFS ベースのデータ転送: ネットワークファイルシステム (NFS) ベースのデータ転送を使用すると、コンピュータから Snow Family デバイスの Amazon S3 バケットにファイルをドラッグアンドドロップできます。

⚠ Warning

NFS ベースのデータ転送は S3 アダプターをサポートしていません。NFS ベースのデータ転送を続行する場合は、NFS 共有をマウントしてオブジェクトを転送する必要があります。AWS CLI を使用してオブジェクトを転送すると失敗します。

詳細については、AWS Snowball エッジデベロッパーガイドの[「オフラインデータ転送に NFS を使用する」](#)を参照してください。

ℹ Note

使用可能なストレージタイプオプションは、選択したジョブタイプと Snow デバイスによって異なります。

4.

ストレージタイプとして [S3 アダプター] を選択した場合、またはブロックストレージをサポートするデバイスを選択した場合は、次の手順を実行して、デバイスに含める 1 つまたは複数の S3 バケットを選択します。

- [S3 バケットを選択] セクションで、次の 1 つまたは複数の手順を実行して 1 つまたは複数の S3 バケットを選択します。
 1. [S3 バケット名] リストで、使用する S3 バケットを選択します。
 2. [項目を検索] フィールドにバケット名の全体または一部を入力し、入力に基づいて表示されるバケットのリストをフィルタリングして、バケットを選択します。
 3. 新しいバケットを作成するには、[新しい S3 バケットを作成] を選択します。[バケット名] リストに、新しいバケット名が表示されます。これを選択します。

1 つ以上の S3 バケットを含めることができます。これらのバケットは、デバイスにローカル S3 バケットとして表示されます。

Select your S3 buckets [Info](#)

The S3 buckets you select will appear as directories on your device. Data stored in these buckets on the device will not be transferred to S3 on return.

[Create a new S3 bucket](#)

🔍

<input type="checkbox"/>	S3 bucket name	Date created
<input type="checkbox"/>	my-gobally-unique-bucket-name	3/15/2023, 5:20:20 PM EDT
<input type="checkbox"/>	do-not-delete-gatedgarden-audit-669419309129	3/11/2023, 5:13:13 PM EST

5. ストレージタイプとして [Amazon S3 互換ストレージ] を選択した場合は、[S3 ストレージ容量] セクションで次の操作を行います。
 - a. Snow Family デバイスの Amazon S3 互換ストレージを単一のデバイスで使用するか、デバイスのクラスターで使用するかを選択します。このガイドの [「AWS Snowball Edge クラスターの使用」](#) を参照してください。
 - b. Snow Family デバイスの Amazon S3 互換ストレージで使用するデバイスストレージの容量を選択します。

Note

Snow Family デバイスの Amazon S3 互換ストレージを使用する場合、デバイスの受領後に Amazon S3 バケットを管理・作成できるため、注文時にバケットを選択する必要はありません。本ガイドの [「Snow Family デバイスの Amazon S3 互換ストレージ」](#) を参照してください。

S3 storage capacity

Select device type

Single device

Cluster

Select storage amount

2.5 TB

Single device

Block storage: 41 TB


Single device ▼

6. ストレージタイプとして [NFS ベースのデータ転送] を選択した場合は、[S3 バケットを選択] セクションで、次の 1 つまたは複数の手順を実行して 1 つまたは複数の S3 バケットを選択します。
 - a. [S3 バケット名] リストで、使用する S3 バケットを選択します。
 - b. [項目を検索] フィールドにバケット名の全体または一部を入力し、入力に基づいて表示されるバケットのリストをフィルタリングして、バケットを選択します。
 - c. 新しいバケットを作成するには、[新しい S3 バケットを作成] を選択します。[バケット名] リストに、新しいバケット名が表示されます。これを選択します。
 - d. NFS データ転送に使用する S3 バケットを選択したら、AMI のブロックストレージとして使用する S3 バケットも選択します。[S3](#) バケットを選択する手順を参照してください。

1 つ以上の S3 バケットを含めることができます。これらのバケットは、デバイスにローカル S3 バケットとして表示されます。

Choose your NFS storage

These S3 buckets will appear on directories on your device. You can transfer data onto these buckets using NFS.

 Only data stored in these directories will be ingested to your S3 buckets in the cloud.

The NFS storage limit is 80 TB

[Create a new S3 bucket](#)

<input type="checkbox"/>	S3 bucket name	Date created
<input type="checkbox"/>	this-unique-bucket-name	6/14/2023, 12:20:08 PM EDT

7. [コンピューティングに EC2 互換インスタンスを使用する – オプション] セクションで、デバイスに含める Amazon EC2 互換 AMI をアカウントから選択します。または、検索フィールドに AMI の名前の全体または一部を入力し、入力に基づいて表示される AMI のリストをフィルタリングして、AMI を選択します。

詳細については、本ガイドの「[デバイスの注文時に AMI を追加する](#)」を参照してください。

この機能には追加料金が発生します。詳細については、「[AWS Snowball Edge の料金](#)」を参照してください。

8. [Next] ボタンを選択します。

ステップ 3: 機能とオプションを選択する

Amazon EKS Anywhere for AWS Snow、AWS IoT Greengrass インスタンス、リモートデバイス管理機能など、Snow Family デバイスジョブに含める機能とオプションを選択します。

機能とオプションを選択するには

1. Amazon EKS Anywhere on AWS Snow セクションで、Amazon EKS Anywhere on AWS Snow を含めるには、Include Amazon EKS Anywhere on Snow を選択し、次の操作を行います。

Note

Amazon EKS Anywhere でサポートされている最新の Kubernetes バージョンで Kubernetes クラスターを作成することをお勧めします。詳細については、「[Amazon EKS-Anywhere バージョニング](#)」を参照してください。アプリケーションに特定のバー

ジョンの Kubernetes が必要な場合は、Amazon EKS の標準サポートまたは拡張サポートで提供されている任意のバージョンの Kubernetes を使用してください。デプロイのライフサイクルを計画する際は、Kubernetes バージョンのリリース日とサポート日を考慮してください。これにより、使用する予定の Kubernetes のバージョンのサポートが失われる可能性を回避できます。詳細については、「[Amazon EKS Kubernetes リリースカレンダー](#)」を参照してください。

- a. [独自の AMI を構築する] セクションで、Amazon EKS Anywhere 用に構築した AMI を選択します。[Amazon EKS Anywhere on AWS Snow 用の Snowball Edge デバイスを注文する前に完了すべきアクション](#) を参照してください。
 - b. Amazon EKS Anywhere クラスターを複数の Snowball Edge デバイスで運用するには、[高可用性] セクションで、注文に含めるデバイスの数を選択します。
2. AWS IoT Greengrass Snow の セクションで、IoT ワークロード用の検証済み AMI を含めるには、Snow デバイスに AWS IoT Greengrass 検証済み AMI をインストールする を選択します。
 3. AWS OpsHub または Snowball Edge Client による Snow Family デバイスのリモート管理を有効にするには、AWS OpsHub または Snowball クライアント を使用して Snow デバイスをリモートで管理を選択します。
 4. [次へ] ボタンを選択します。

ステップ 4: セキュリティ、配送、通知の設定を選択する

トピック

- [セキュリティ設定を選択する](#)
- [配送設定を選択する](#)
- [通知設定を選択する](#)

セキュリティ設定を選択する

セキュリティ設定では、AWS Snow Family デバイスのジョブに対するアクセス許可と暗号化設定を追加し、転送中のデータを保護できるようにします。

ジョブのセキュリティを設定するには

1. [Encryption] (暗号化) セクションで、使用する [KMS key] (KMS キー) を選択します。

- デフォルト AWS Key Management Service (AWS KMS) キーを使用する場合は、AWS/importexport (デフォルト) を選択します。これは、他のキーが定義されていないときにインポートジョブとエクスポートジョブを保護するデフォルトのキーです。
 - 独自の AWS KMS キーを指定する場合は、キー ARN の入力 を選択し、キー ARN ボックスに Amazon リソースネーム (ARN) を指定し、この KMS キーの使用 を選択します。キーの ARN がリストに追加されます。
2. [サービスアクセスタイプを選択] セクションで、次のいずれかの操作を行います。
 - Snow コンソールを選択すると、サービスにリンクされたロールが作成され、ユーザーに代わって AWS リソースにアクセスします。は、ユーザーに代わって Amazon S3 と Amazon Simple Notification Service (Amazon SNS) を使用するための許可を AWS Snow Family に付与します。このロールは、Snow サービスに AWS Security Token Service (AWS STS) AssumeRole の信頼を付与します。
 - [使用する既存のサービスロールを追加] を選択して、必要な [ロール ARN] を指定するか、デフォルトのロールを使用できます。
 3. [次へ] をクリックします。

配送設定を選択する

Snow Family デバイスの受け取りと返却には、発送と返送をともなうため、正確な配送情報を提供することが重要です。

配送の詳細を指定するには

1. [配送先住所] のセクションで、既存の住所を選択するか、新しい住所を追加します。
 - [Use recent address] (最近使用した住所を使用する) を選択すると、ファイル内にある住所が表示されます。リストから必要な住所を注意深く選択します。
 - [Add a new address] (新しい住所を追加する) を選択した場合は、要求されている住所情報を入力します。は新しい配送情報 AWS Snow ファミリーマネジメントコンソール を保存します。

Note

住所に入力した国は、デバイスの配送先の国と一致し、その国で有効でなければなりません。

2. [Shipping speed] (配送速度) セクションで、ジョブの配送速度を選択します。この速度は、デバイスが目的地間で配送される速さを示すもので、何日後に到着するのかを反映したものではありません。選択できる配送速度は次のとおりです。
 - [One-Day Shipping (1 business day)] (翌日配送 (1 営業日))
 - [Two-Day Shipping (2 business day)] (翌々日配送 (2 営業日))
 - 「[配送業者](#)」を参照してください。

通知設定を選択する

通知により、AWS Snow Family デバイスジョブの最新のステータスが更新されます。SNS トピックを作成し、ジョブのステータスが変更されると、Amazon Simple Notification Service (Amazon SNS) から E メールが届きます。

通知をセットアップするには

- [Set notifications] (通知の設定) セクションで、次のいずれかを実行します。
 - 既存の SNS トピックを使用する場合は、[Use an existing SNS topic] (既存の SNS トピックを使用する) を選択し、リストからトピックの Amazon リソースネーム (ARN) を選択します。
 - 新しい SNS トピックを作成する場合は、[Create a new SNS topic] (新しい SNS トピックの作成) を選択します。トピックの名前と E メールアドレスを入力します。

Note

米国西部 (北カリフォルニア) および米国西部 (オレゴン) リージョンで作成された Snow デバイスを注文するジョブは、米国東部 (バージニア北部) リージョン経由でルーティングされます。このため、Amazon SNS などのサービスコールも米国東部 (バージニア北部) を経由します。最高のエクスペリエンスを得るには、米国東部 (バージニア北部) リージョンに新しい SNS トピックを作成することをお勧めします。

ジョブの以下のいずれかのステータスについて通知されます。

- Job created
- Preparing device
- Preparing shipment
- In transit to you
- Delivered to you
- への転送中 AWS
- 仕分け施設にあります
- で AWS
- インポート中
- 完了
- キャンセル

ジョブステータス変更通知と暗号化された SNS トピックの詳細については、このガイドの [「Snow Family デバイスの通知」](#) を参照してください。

[次へ] を選択します。

ステップ 5: ジョブの概要を確認してジョブを作成する

AWS Snow Family デバイスジョブに必要なすべての情報を提供したら、ジョブを確認して作成します。ジョブを作成すると、AWS は Snow Family デバイスの出荷準備を開始します。

一部の国では、ジョブは輸出規制法の対象となります。輸出許可が必要になる場合もあります。また、米国の輸出および再輸出法が適用されます。特定の国および米国の法規制を迂回することは禁じられています。

1. ジョブの概要ページで、ジョブを作成する前にすべてのセクションを確認してください。変更を加えたい場合は、該当するセクションで [Edit] (編集) を選択し、情報を編集します。
2. 確認と編集が完了したら、[Create job] (ジョブを作成) を選択します。

Note

Snow Family デバイスを注文するジョブを作成したら、ジョブ作成状態のときにキャンセルでき、料金は発生しません。詳細については、[「によるジョブのキャンセル AWS Snow ファミリーマネジメントコンソール」](#)を参照してください。

ジョブが作成されると、[Job status] (ジョブステータス) のセクションにジョブのステータスが表示されます。ジョブのステータスの詳細については、[ジョブステータス](#)を参照してください。

ダウンロード AWS OpsHub

AWS Snow Family デバイスは AWS OpsHub for Snow Family、デバイスとローカルの管理に使用できる使いやすいツールを提供します AWS のサービス。

をクライアントコンピュータに AWS OpsHub インストールすると、次のようなタスクを実行できます。

- 単一のデバイスまたはクラスター化されたデバイスのロック解除や設定
- ファイルの転送
- Snow Family デバイスで実行されているインスタンスの起動と管理

詳細については、「[AWS OpsHub for Snow Family を使用してデバイスを管理する](#)」を参照してください。

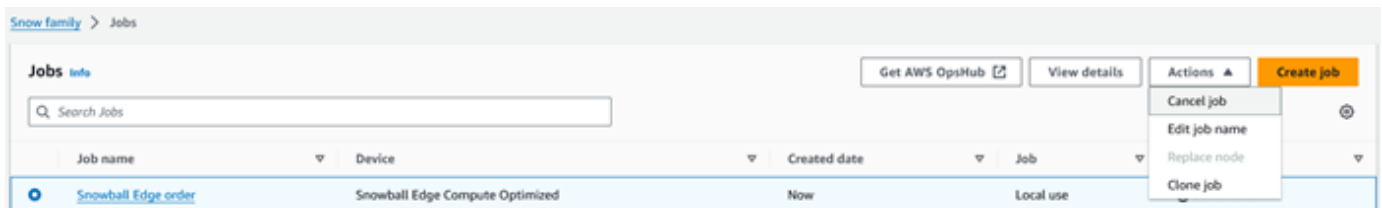
ダウンロードしてインストールするには AWS OpsHub for Snow Family

1. [AWS Snowball リソース](#) で、 をクリックします AWS OpsHub。ダウンロードリンク AWS OpsHub があるセクションで、オペレーティングシステム AWS OpsHub にインストールする適切なダウンロードリンクを選択します。
2. [AWS OpsHub] セクションで、お使いのオペレーティングシステムの [Download] (ダウンロード) を選択し、インストール手順に従います。完了したら、[Next] を選択します。

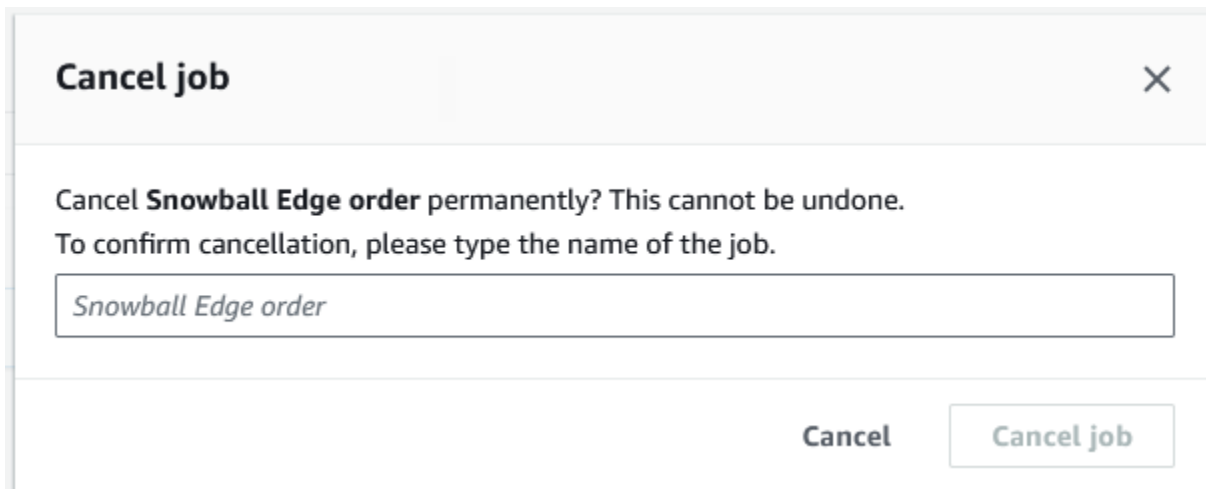
によるジョブのキャンセル AWS Snow ファミリーマネジメントコンソール

Snow Family デバイスを注文するジョブを作成したら、を使用してジョブをキャンセルできます AWS Snow ファミリーマネジメントコンソール。ジョブをキャンセルすると、注文したデバイスを受信しません。ジョブをキャンセルできるのは、ジョブのステータスが Job created の場合のみです。ジョブがこのステータスを過ぎた後、ジョブをキャンセルすることはできません。詳細については、「[ジョブのステータス](#)」を参照してください。

1. [AWS Snow ファミリーマネジメントコンソール](#) にログインします。
2. キャンセルするジョブを選択します。
3. [アクション] を選択します。表示されるメニューから、ジョブのキャンセル を選択します。



4. ジョブのキャンセルウィンドウが表示されます。ジョブのキャンセルを確認するには、**job name**を入力し、ジョブのキャンセル を選択します。ジョブのリストの「ステータス」列に「キャンセル済み」が表示されます。



Snowball Edge の受領

デバイスを受け取ると、その AWS Snowball Edge デバイスがボックスに入っていないことに気付くかもしれません。デバイスはそれ自体が物理的に頑丈な配送用コンテナです。デバイスが到着した

ら、まず、損傷や明らかな改ざんがないか検査してください。デバイスに疑わしい点が見つかった場合は、内部ネットワークに接続しないでください。代わりに、[AWS Support](#) まで問題をお知らせいただければ、新しいデバイスをお客様宛に発送します。

⚠ Important

AWS Snowball Edge デバイスは のプロパティです AWS。AWS Snowball Edge デバイスへの改ざんは、AWS 「利用規定」 ポリシーに違反します。詳細については、[AWS の適正利用規約](#)を参照してください。

デバイスは、次の図のような外観です。



内部ネットワークにデバイスを接続する準備ができたなら、次のセクションを参照してください。

Next: [ローカルネットワークへの接続](#)

ローカルネットワークへの接続

以下の手順を使用して、AWS Snowball Edge デバイスをローカルネットワークに接続します。デバイスをインターネットに接続する必要はありません。デバイスは、前面、背面、上部の3か所にドアがあります。

デバイスをネットワークに接続するには

1. デバイスの前面および背面のドアを、アプライアンスのドアスロット内にスライドさせて開きます。こうすることで、デバイス前面に埋め込まれた LCD ディスプレイのタッチスクリーンと、背面の電源ポートおよびネットワークポートにアクセスできます。

Note

Snowball Edge デバイスを使用している間は、前面ドアと背面ドアを閉めないでください。ドアが開いていると、空気でデバイスが冷却されます。デバイスの使用中にドアを閉めると、過熱を防ぐためにデバイスがシャットダウンする場合があります。

2. 上部ドアを開いて付属の電源ケーブルをケーブルフックから取り外し、デバイスを電源に接続します。
3. RJ45、SFP+、または QSFP+ ネットワークケーブルのいずれかを選択し、デバイスをネットワークに接続します。ネットワークポートは、デバイスの背面にあります。
4. LCD ディスプレイの上にある電源ボタンを押して、AWS Snowball Edge デバイスの電源を入れます。
5. デバイスの準備ができると、デバイスの起動準備中に、LCD ディスプレイに短いビデオが表示されます。約 10 分後に、デバイスをロック解除する準備が整います。
6. (オプション) デフォルトのネットワーク設定を変更するには、LCD ディスプレイで [CONNECTION] を選択します。

次の手順を使用して、IP アドレスを別の静的 IP アドレスに変更できます。

起動時の問題をトラブルシューティングするには、「[起動に関する問題のトラブルシューティング](#)」を参照してください。

AWS Snowball Edge デバイスの IP アドレスを変更するには

1. LCD ディスプレイで、[CONNECTION (接続)] を選択します。

AWS Snowball Edge デバイスの現在のネットワーク設定が画面に表示されます。ドロップダウンボックスの下の IP アドレスは、AWS Snowball Edge デバイスがリクエストした DHCP アドレスを反映するように自動的に更新されます。

2. (オプション) IP アドレスを静的 IP アドレスに変更します。そのままにしておくこともできます。

デバイスはネットワークに接続されました。

Important

データの破損を防ぐため、使用中は AWS Snowball Edge デバイスを切断したり、接続設定を変更したりしないでください。

Next: [Snow Family デバイスにアクセスするための認証情報の取得](#)

Snow Family デバイスにアクセスするための認証情報の取得

各ジョブには、Snow Family デバイスへのアクセスを認証するために AWS Snow ファミリーマネジメントコンソール またはジョブ管理 API から取得する必要がある一連の認証情報があります。これらの認証情報は、暗号化されたマニフェストファイルおよび関連するロック解除コードです。マニフェストファイルには、ジョブとそれに関連付けられたアクセス権限に関する重要な情報が含まれています。

Note

認証情報は、デバイスが発送された後に取得します。ジョブのステータスは AWS Snow ファミリーマネジメントコンソールで確認できます。詳細については、「[ジョブステータス](#)」を参照してください。

コンソールを使用して認証情報を取得するには

1. にサインイン AWS Management Console し、 を開きます [AWS Snow ファミリーマネジメントコンソール](#)。
2. コンソールで、テーブルを検索してジョブマニフェストをダウンロードする特定のジョブを選択します。
3. ジョブステータスペインを展開し、ジョブの詳細を表示 を選択します。
4. 表示された詳細ペインで、[Credentials] を展開してから、次の操作を行います。
 - デバイスのロックを解除するには 29 文字すべてを指定する必要があるため、ロック解除コード (ハイフンを含む) を書き留めておきます。

- ダイアログボックスで [Download manifest] を選択し、指示に従ってジョブマニフェストファイルをコンピュータにダウンロードします。マニフェストファイル名には [Job ID] が含まれません。

Note

ロック解除コードのコピーは、そのジョブのマニフェストと同じコンピュータの場所に保存しないことをお勧めします。詳細については、「[Snowball Edge デバイスの使用に関するベストプラクティス](#)」を参照してください。

認証情報を取得したら、次のステップとして、AWS Snowball Edge デバイスのロック解除に使用される Snowball Edge クライアントをダウンロードします。

Next: [Snowball Edge クライアントのダウンロードとインストール](#)

Snowball Edge クライアントのダウンロードとインストール

Snowball Edge クライアントは、[AWS Snowball Edge リソース](#)からダウンロードしてインストールできます。そのページに、お使いのオペレーティングシステムに適したインストールパッケージが用意されています。手順に従って、Snowball Edge クライアントをインストールします。ワークステーション内のターミナルから Snowball Edge クライアントを実行する場合、お使いのオペレーティングシステムに応じて、特定のパスを使用する必要がある場合があります。

- Microsoft Windows – クライアントをインストールすると、追加の準備なしに任意のディレクトリから実行できます。
- Linux – Snowball Edge クライアントを `~/snowball-client-linux-build_number/bin/` ディレクトリから実行する必要があります。Snowball Edge クライアントは、64 ビット Linux ディストリビューションでのみサポートされます。
- MacOS – `install.sh` スクリプトにより、Snowball Edge クライアント `.tar` ファイルから `/usr/local/bin/snowball` ディレクトリにフォルダがコピーされます。このスクリプトを実行する場合、`/usr/local/bin` が `bash_profile` のパスであれば、Snowball Edge クライアントを任意のディレクトリから実行できます。パスは、`echo $PATH` コマンドで確認できます。

Snowball Edge クライアントコマンドの詳細については、「」を参照してください [Snowball Edge クライアントコマンドの使用](#)。

Next: [Snow Family デバイスのロック解除](#)

Snow Family デバイスのロック解除

このセクションでは、Snowball Edge CLI を使用した Snow Family デバイスのロック解除について説明します。Snow Family デバイス用のグラフィカルユーザーインターフェイス (GUI) ツールであるを使用してデバイスのロックを解除するには [AWS OpsHub](#)、[「デバイスのロック解除」](#)デバイスの」を参照してください。

Snow Family デバイスを使用してデータを転送したり、エッジコンピューティングタスクを実行する前に、デバイスのロックを解除する必要があります。デバイスのロックを解除するときは、29 桁のロック解除コードとマニフェストファイルの 2 つの形式の認証情報を提供することで、デバイスのアクセス機能を認証します。デバイスのロックを解除すると、デバイスをさらに設定したり、デバイスからデータを移動したり、Amazon EC2-compatible インスタンスを設定して使用したりできます。

デバイスのロックを解除する前に、デバイスを電源とネットワークに接続し、電源を入れ、IP アドレスを割り当てる必要があります。[ローカルネットワークへの接続](#)。Snow Family デバイスに関する次の情報が必要になります。

- Snowball Edge クライアントのダウンロードとインストール 詳細については、「[Snowball Edge クライアントのダウンロードとインストール](#)」を参照してください。
- から認証情報を取得します AWS Snow ファミリーマネジメントコンソール。1 つ以上のスタンドアロンデバイスの場合、各 Snow Family デバイスのロック解除コードとマニフェストファイル。Snowball Edge デバイスのクラスターの場合、クラスターの 1 つのロック解除コードと 1 つのマニフェストファイル。認証情報のダウンロードの詳細については、「」を参照してください [Snow Family デバイスにアクセスするための認証情報の取得](#)。
- 各デバイスに電源を入れ、ネットワークに接続します。詳細については、「[ローカルネットワークへの接続](#)」を参照してください。

Snowball Edge クライアントを使用してスタンドアロンデバイスのロックを解除するには

1. AWS Snowball Edge デバイスの LCD ディスプレイの接続タブで、AWS Snowball Edge デバイスの IP アドレスを見つけます。その IP アドレスをメモします。
2. `unlock-device` コマンドを使用して、Snow Family デバイスの IP アドレスと認証情報を使用して、次のように Snow Family デバイスへのアクセスを認証します。

```
snowballEdge unlock-device --endpoint https://ip-address-of-device --manifest-file /Path/to/manifest/file.bin --unlock-code 29-character-unlock-code
```

デバイスは、次のメッセージで正常にロック解除されたことを示します。

```
Your Snowball Edge device is unlocking. You may determine the unlock state of your device using the describe-device command. Your Snowball Edge device will be available for use when it is in the UNLOCKED state.
```

コマンドが を返す場合は connection refused、「」を参照してください [Snow Family デバイスのロック解除のトラブルシューティング](#)。

Example **unlock-device** コマンドの

この例では、デバイスの IP アドレスは 192.0.2.0、マニフェストファイル名は JID2EXAMPLE-0c40-49a7-9f53-916aEXAMPLE81-manifest.bin、29 文字のロック解除コードは です 12345-abcde-12345-ABCDE-12345。

```
snowballEdge unlock-device --endpoint https://192.0.2.0 --manifest-file /Downloads/JID2EXAMPLE-0c40-49a7-9f53-916aEXAMPLE81-manifest.bin /--unlock-code 12345-abcde-12345-ABCDE-12345
```

Snowball Edge クライアントを使用して Snowball Edge デバイスのクラスターをロック解除するには

1. 各デバイスの LCD ディスプレイの接続タブで、クラスター内の各 AWS Snowball Edge デバイスの IP アドレスを見つけます。IP アドレスをメモします。
2. snowballEdge unlock-cluster コマンドを使用して、次のように、クラスター内のいずれかの AWS Snowball Edge デバイスの IP アドレス、認証情報、およびクラスター内のすべてのデバイスの IP アドレスを使用して、デバイスデバイスのクラスターへのアクセスを認証します。

```
snowballEdge unlock-cluster --endpoint https://ip-address-of-device --manifest-  
file Path/to/manifest/file.bin --unlock-code 29-character-unlock-code --device-ip-  
addresses ip-address-of-cluster-device-1 ip-address-of-cluster-device-2 ip-address-  
of-cluster-device-3
```

デバイスのクラスターは、次のメッセージで正常にロック解除されたことを示します。

```
Your Snowball Edge Cluster is unlocking. You may determine the unlock state of your  
cluster using the describe-cluster command. Your Snowball Edge Cluster will be  
available for use when your Snowball Edge devices are in the UNLOCKED state.
```

コマンドが を返す場合はconnection refused、「」を参照してください[Snow Family デバイスのロック解除のトラブルシューティング](#)。

Example **unlock-cluster** コマンドの

この例では、5つのデバイスのクラスターの場合、クラスター内のデバイスの1つのIPアドレスは192.0.2.0、マニフェストファイル名はJID2EXAMPLE-0c40-49a7-9f53-916aEXAMPLE81-manifest.bin、29文字のロック解除コードは 12345-abcde-12345-ABCDE-12345。

```
snowballEdge unlock-cluster --endpoint https://192.0.2.0 --manifest-file /  
Downloads/JID2EXAMPLE-0c40-49a7-9f53-916aEXAMPLE81-manifest.bin /  
  
--unlock-code 12345-abcde-12345-ABCDE-12345 --device-ip-addresses 192.0.2.0  
192.0.2.1 192.0.2.2 192.0.2.3 192.0.2.4
```

Snow Family デバイスのロック解除のトラブルシューティング

unlock-device コマンドが を返す場合 connection refused、コマンド構文の入力を間違えているか、コンピュータまたはネットワークの設定によってコマンドが Snow デバイスに到達できない可能性があります。問題を解決するには、次のアクションを実行します。

1. コマンドが正しく入力されたことを確認します。
 - a. デバイスの LCD 画面を使用して、コマンドで使用されている IP アドレスが正しいことを確認します。
 - b. コマンドで使用されるマニフェストファイルへのパスが正しいことを確認します。これには、ファイル名も含まれます。
 - c. を使用して [AWS Snow ファミリーマネジメントコンソール](#)、コマンドで使用されているロック解除コードが正しいことを確認します。
2. 使用しているコンピュータが Snow デバイスと同じネットワークとサブネットにあることを確認します。
3. 使用しているコンピュータとネットワークが Snow デバイスへのアクセスを許可するように設定されていることを確認します。オペレーティングシステムの ping コマンドを使用して、コンピュータがネットワーク経由で Snow デバイスにアクセスできるかどうかを確認します。ウイルス対策ソフトウェアの設定、ファイアウォール設定、仮想プライベートネットワーク (VPN)、またはコンピュータとネットワークのその他の設定を確認します。

これで、Snow Family デバイスの使用を開始できます。

Next: [ローカルユーザーの設定](#)

ローカルユーザーの設定

以下は、AWS Snowball Edge デバイスでローカル管理者を設定する手順です。

1. ルートユーザーの認証情報を取得する

snowballEdge list-access-keys と snowballEdge get-secret-access-key を使用して、ローカルの認証情報を取得します。詳細については、「[認証情報の取得](#)」を参照してください。

2. **aws configure** を使用してルートユーザーの認証情報を設定する

AWS Access Key ID、AWS Secret Access Key、および Default region name を指定します。リージョン名は snow である必要があります。オプションで、Default output format を指定します。の設定の詳細については AWS CLI、「ユーザーガイド」の「[AWS CLI の設定](#)」を参照してください。

3. デバイスで 1 人以上のローカルユーザーを作成する

create-user コマンドを使用して、デバイスにユーザーを追加します。

```
aws iam create-user --endpoint endpointIPAddress:6078 --profile ProfileID --region snow --user-name UserName
```

ビジネスニーズに応じてユーザーを追加したら、AWS ルート認証情報を安全な場所に保存します。この認証情報は、アカウントとサービスの管理タスクにのみ使用します。IAM ユーザーの作成方法の詳細については、IAM ユーザーガイドの [AWS アカウント内での IAM ユーザーの作成](#) を参照してください。

4. ユーザーのアクセスキーの作成

Warning

このシナリオでは、プログラムによるアクセスと長期的な認証情報を持つ IAM ユーザーが必要です。これはセキュリティ上のリスクをもたらします。このリスクを軽減するために、これらのユーザーにはタスクの実行に必要な権限のみを付与し、不要になったユーザーを削除することをお勧めします。アクセスキーは、必要に応じて更新できます。詳細については、「[IAM ユーザーガイド](#)」の「アクセスキーの更新」を参照してください。

create-access-key コマンドを使用して、ユーザーのアクセスキーを作成します。

```
aws iam create-access-key --endpoint endpointIPAddress:6078 --profile ProfileID --region snow --user-name UserName
```

アクセスキー情報をファイルに保存し、ユーザーに配布します。

5. アクセスポリシーの作成

ユーザーごとに、デバイスの機能へのアクセスレベルを変えたい場合があります。次の例では、s3-only-policy という名前のポリシードキュメントを作成し、ユーザーにアタッチします。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "s3:*",
      "Resource": "*"
    }
  ]
}
```

```
aws iam create-policy --endpoint endpointIPAddress:6078 --profile ProfileID --
region snow --policy-name s3-only-policy --policy-document file://s3-only-policy
```

6. ユーザーへのポリシーのアタッチ

attach-user-policy を使用して、s3-only-policy をユーザーに適用します。

```
aws iam attach-user-policy --endpoint endpointIPAddress:6078 --profile ProfileID
--region snow --user-name UserName --policy-arn arn:aws:iam::AccountID:policy/POLICYNAME
```

ローカルでの IAM の使用の詳細については、[ローカルでの IAM の使用](#) を参照してください。

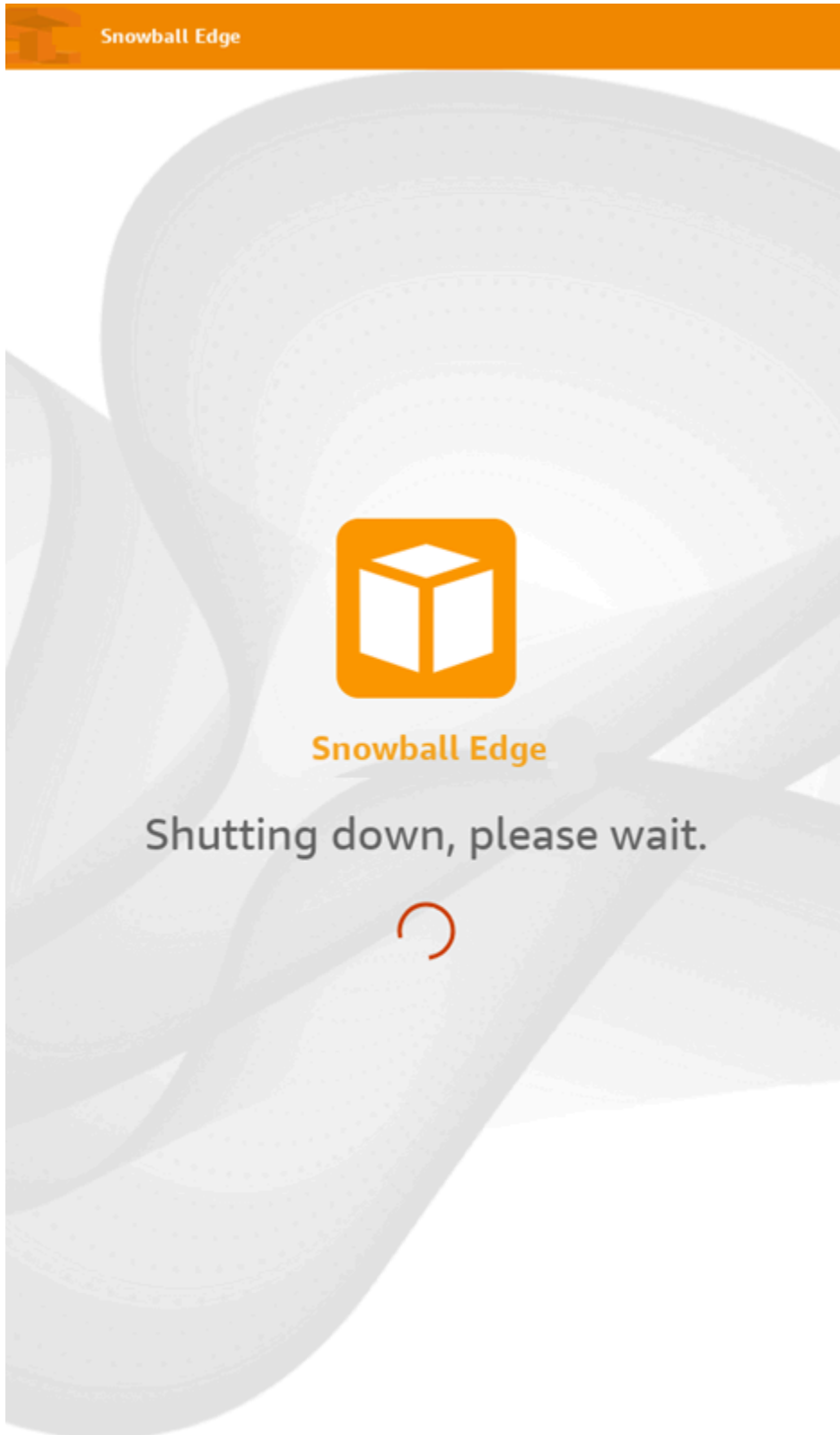
Next: [AWS Snowball エッジデバイスの使用](#)

Snow Family デバイスの再起動

Snow Family デバイスを再起動する前に、デバイスへのすべてのデータ転送が停止していることを確認します。

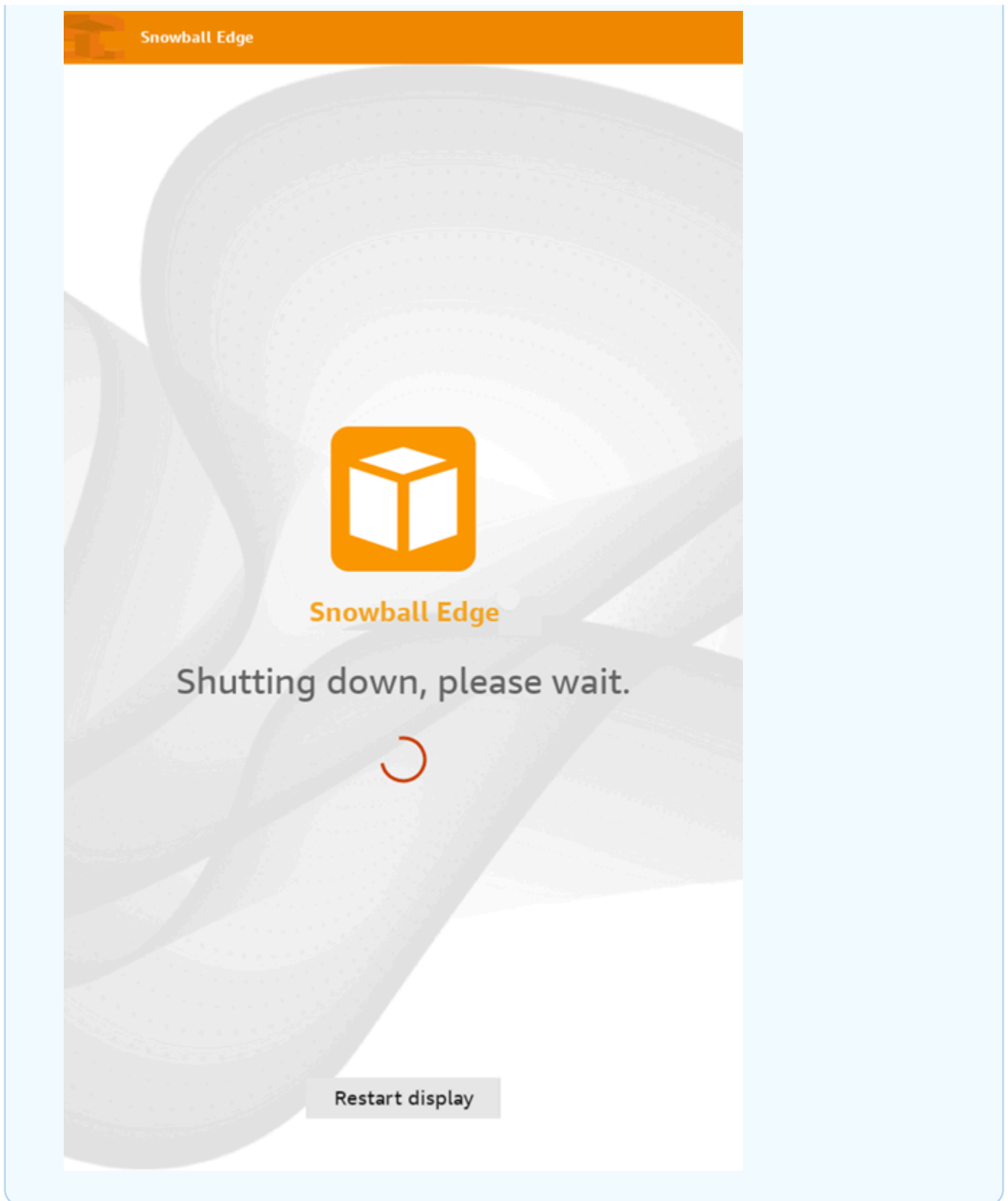
電源ボタンを使用してデバイスを再起動するには

1. デバイスとの通信がすべて終了したら、LCD 画面の上にある電源ボタンを押して、電源をオフにします。デバイスがシャットダウンするまでに約 20 秒かかります。デバイスのシャットダウン中は、デバイスがシャットダウン中であることを示すメッセージが LCD 画面に表示されません。



Note

デバイスが実際にシャットダウンされていないのに LCD 画面にシャットダウンメッセージが表示される場合は、画面の [ディスプレイの再起動] ボタンを押して画面を通常の動作に戻します。



2. 電源ボタンを押します。デバイスの準備ができると、デバイスの起動準備中に、LCD ディスプレイに短いビデオが表示されます。約 10 分後に、デバイスをロック解除する準備が整います。

3. デバイスのロックを解除します。[Snow Family デバイスのロック解除](#) を参照してください。

Snowball Edge クライアントを使用してデバイスを再起動するには

1. デバイスとの通信がすべて終了したら、reboot-device コマンドを使用して再起動します。デバイスの準備ができると、デバイスの起動準備中に、LCD ディスプレイに短いビデオが表示されます。約 10 分後に、デバイスをロック解除する準備が整います。

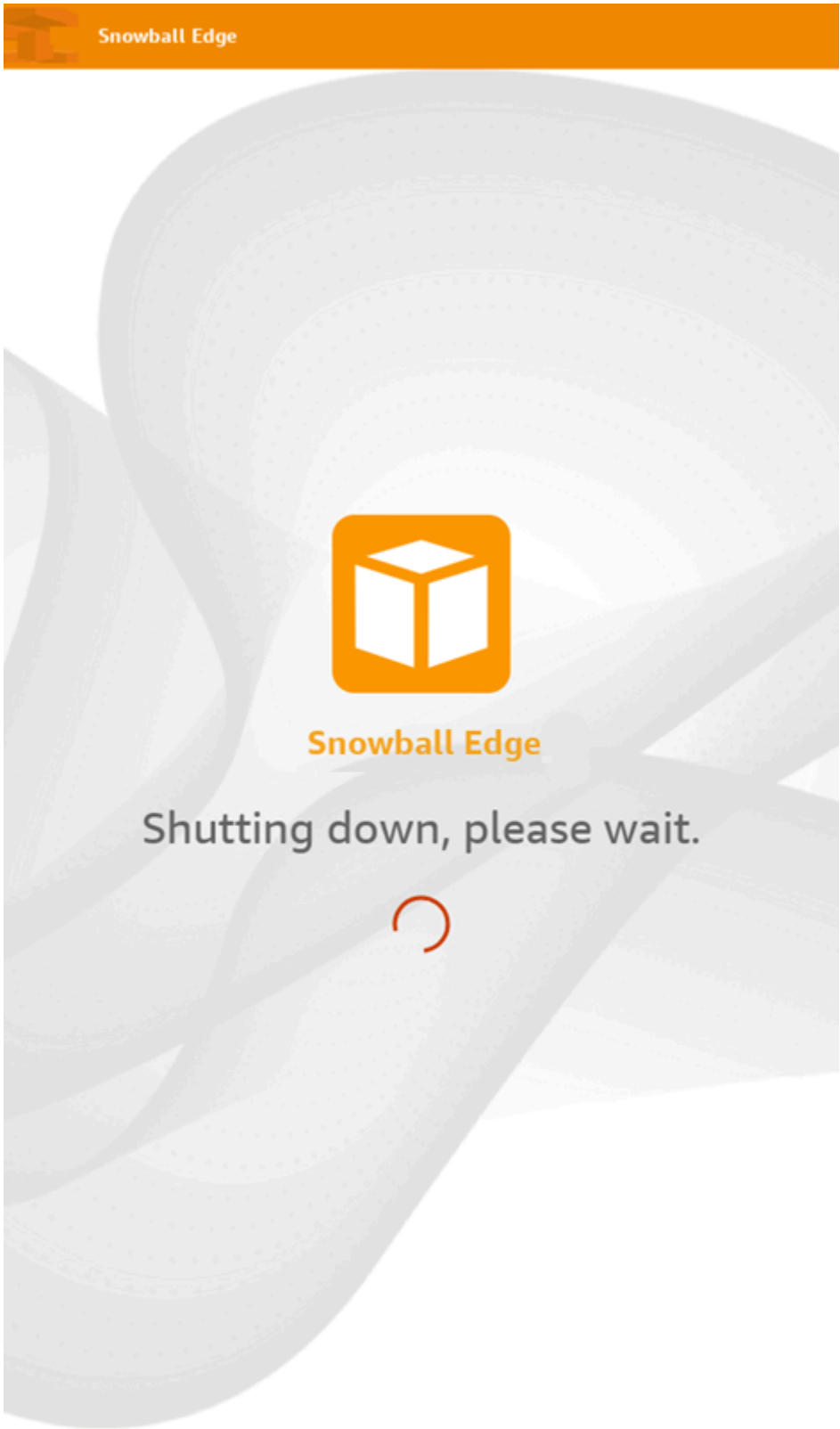
```
snowballEdge reboot-device
```

2. デバイスのロックを解除します。[Snow Family デバイスのロック解除](#) を参照してください。

Snowball Edge の電源をオフにする

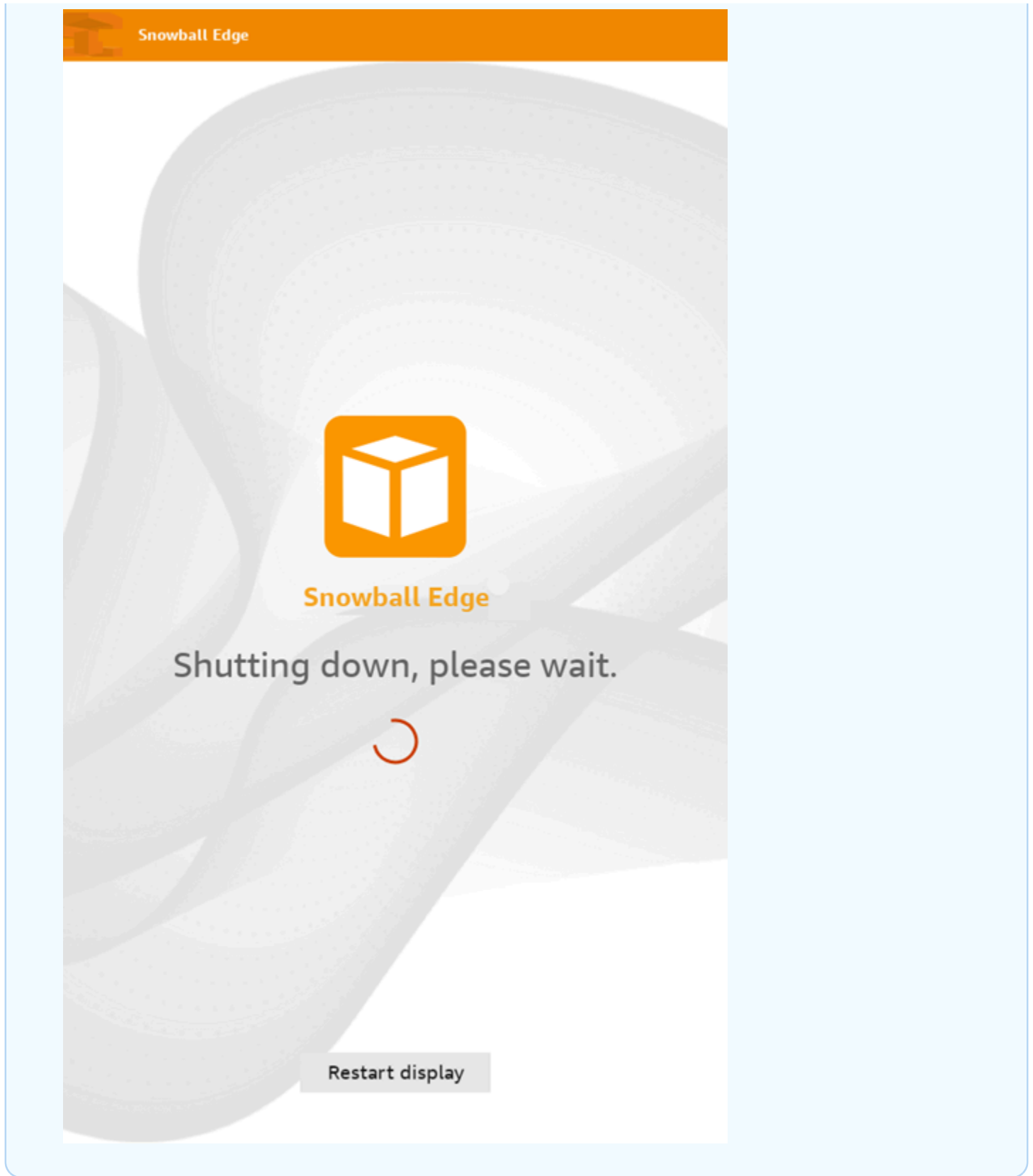
デバイスのデータ転送が完了したら AWS Snowball Edge、への返送に備えて準備します AWS。作業を始める前に、デバイスへのすべてのデータ転送が停止したことを確認します。NFS インターフェイスを使用してデータを転送する場合は、デバイスの電源をオフにする前に無効にします。詳細については、「[NFS インターフェイスの管理](#)」を参照してください。

デバイスとの通信がすべて終了したら、LCD 画面の上にある電源ボタンを押して、電源をオフにします。デバイスがシャットダウンするまでに約 20 秒かかります。デバイスのシャットダウン中は、デバイスがシャットダウン中であることを示すメッセージが LCD 画面に表示されます。



Note

デバイスが実際にシャットダウンされていないのに LCD 画面にシャットダウンメッセージが表示される場合は、画面の [ディスプレイの再起動] ボタンを押して画面を通常の動作に戻します。



デバイスのシャットダウン後、E Ink ディスプレイに配送情報が表示されます。E Ink ディスプレイに返送配送情報が表示されない場合は、[お問い合わせ](#)ください AWS Support。

Next: [Snowball Edge デバイスの返却](#)

Snowball Edge デバイスの返却

Snowball Edge の使用が完了し、電源をオフにすると、配送業者はそれを に返送します AWS。配送業者は、デバイスの配送の追跡番号を自動的に提供します。追跡番号が に表示されます AWS Snow ファミリーマネジメントコンソール。追跡番号とキャリアの追跡ウェブサイトへのリンクにアクセスするには、コンソールでジョブのステータスの詳細を表示します。詳細については、[「Snow Family デバイスの返送」](#)を参照してください。

キャリアはデバイスを仕 AWS 分け施設に配送し、デバイスは AWS データセンターに転送されます。データセンターで、 は AWS、配送中にデバイスが改ざんされていないことと、デバイスが正常であることを確認します。デバイスに Amazon S3 にインポートするデータが含まれている場合、AWS はインポートを開始します。そうしないと、デバイス上のデータは安全に消去されます。でデバイスが AWS 処理されるたびに、ステータスの変更を追跡できます AWS Snow ファミリーマネジメントコンソール。デバイスを注文するジョブの作成時にそのオプションを選択すると、ステータス変更に関する Amazon SNS 通知が送信されます。詳細については、[「インポートステータスのモニタリング」](#)を参照してください。

最終ステータス値には、AWS Snowball Edge デバイスが に よって受信された日時 AWS、データインポートが開始された日時、ジョブが完了した日時が含まれます。

Note

デバイスに Amazon S3 にインポートする予定のデータが含まれており、デバイス上のデータをインポートしたくない場合は、AWS Support に連絡して Snow ジョブのキャンセルをリクエストしてください。ジョブをキャンセルすると、データ転送はスキップされ、所定のプロセスに従ってデバイスが安全に消去されます。厳格な過程管理と運用手順のため、お客様のデータが保存されているデバイスを当社の施設に保管することはできません。

返送用の AWS Snowball Edge デバイスを準備するには

1. デバイスの電源をオフにします。詳細については、[「Snowball Edge の電源をオフにする」](#)を参照してください。
2. デバイ스에 接続されているネットワークケーブルをすべて切断します。

3. 電源ケーブルを切り離します。AWS Snowball Edge デバイス上部のケーブルフックに格納します。
4. AWS Snowball Edge デバイスの背面、上部、前面のドアを閉じます。クリック音が聞こえて聞こえるまで、各ドアを押します。

Next: [Snow Family デバイスの返送](#)

Snow Family デバイスの返送

AWS Snowball Edge デバイスは データセンターから出荷され、AWS データセンターに配送されます。デバイスの E Ink 画面の前払い配送情報には、AWS Snowball Edge デバイスを返送するための住所が含まれています。返送の配送速度は、デバイスの受領時にかかった配送速度と同じです。ステータスの変更は AWS Snow ファミリーマネジメントコンソールで追跡でき、配送の進行状況は各リージョンの配送業者を通じて追跡できます。

AWS Snowball Edge デバイスの返却方法の詳細については、「」を参照してください [配送業者](#)。

Important

で特に指示されていない限り AWS、AWS Snowball Edge デバイスに別の配送ラベルを貼り付けしないでください。AWS Snowball Edge デバイス E Ink ディスプレイに表示される配送情報を常に使用します。

配送業者

Snow Family デバイスを注文するジョブを作成するときは、AWS Snowball Edge デバイスの配送先住所を指定します。お客様のリージョンをサポートする配送業者が、からお客様 AWS へのデバイスの配送と、お客様からへの返送を処理します AWS。ジョブのステータスが「出荷準備中」になると、発送の配送情報を確認できます。

出荷されるすべての AWS Snowball Edge デバイスに追跡番号があります。追跡番号や追跡を行うウェブサイトのリンクは、[AWS Snow ファミリーマネジメントコンソール](#) のジョブダッシュボード、またはジョブ管理 API を使用して確認できます。

これらのキャリアは AWS Snowball Edge デバイスでサポートされています。

- インドの配送業者は Blue Dart です。

- 韓国、日本、オーストラリア、インドネシアの配送業者は、Kuehne + Nagel です。
- 中国および香港の場合、S.F. Express がキャリアです。
- その他のリージョンのキャリアはすべて、[UPS](#) です。

トピック

- [AWS Snowball Edge EU、米国、英国、南アフリカ、カナダでの UPS の集荷](#)
- [AWS Snowball 英国での集荷](#)
- [AWS Snowball ブラジルでの集荷](#)
- [AWS Snowball オーストラリアでの集荷](#)
- [AWS Snowball インドでの集荷](#)
- [AWS Snowball 韓国でのエッジ集荷](#)
- [AWS Snowball 香港でのエッジ集荷](#)
- [AWS Snowball シンガポール、日本、インドネシアでの集荷](#)
- [AWS Snowball アラブ首長国連邦ドバイでの受け取りと返却](#)
- [配送速度](#)

AWS Snowball Edge EU、米国、英国、南アフリカ、カナダでの UPS の集荷

UPS は、多くの場合、EU、米国、英国、南アフリカ、カナダでお客様のデバイスの集荷に対応します。以下の有用なガイドラインを参照してください。

- UPS で直接集荷を手配するか、AWS Snowball Edge デバイスを UPS パッケージの持ち込み施設に持ち込み、に配送します AWS。
- E Ink ディスプレイの前払い UPS 配送ラベルには、AWS Snowball Edge デバイスの返送先住所が記載されています。
- AWS Snowball Edge デバイスは の AWS 仕分け施設に配送され、AWS データセンターに転送されます。UPS が追跡番号を提供します。

Important

で特に指示されていない限り AWS、AWS Snowball Edge デバイスに別の配送ラベルを貼り付けしないでください。必ず、デバイスの E Ink ディスプレイに表示される配送情報を使用してください。

UPS は次の EU 加盟国に Snowball Edge デバイスを配送します: オーストリア、ベルギー、ブルガリア、クロアチア、キプロス共和国、チェコ共和国、デンマーク、エストニア、フィンランド、フランス、ドイツ、ギリシャ、ハンガリー、イタリア、アイルランド、ラトビア、リトアニア、ルクセンブルク、マルタ、オランダ、ポーランド、ポルトガル、ルーマニア、スロバキア、スロベニア、スペイン、スウェーデン。

Note

現在、英国と欧州連合 (EU) 諸国間の注文は国際注文とみなされ、特別な国際プロセスを通じた承認が必要です。英国と EU 間でのデバイス配送が必要な場合は、UPS の集荷または持ち込みを手配する前に、<snowball-shipping@amazon.com> に E メールを送信し、商業送り状をリクエストしてください。

Snow Family 製品に対する UPS サービスは、その国の国内でのみ提供されます。

AWS Snowball 英国での集荷

英国において、UPS に Snowball Edge の集荷を依頼する際は、次の情報に注意してください。

- UPS で直接集荷を手配 AWS Snowball Edge するか、AWS Snowball Edge デバイスを UPS の集荷施設に持ち込み、に配送します AWS。
- E Ink ディスプレイの前払い UPS 配送ラベルには、AWS Snowball Edge デバイスを返送するための正しい住所が含まれています。
- AWS Snowball Edge デバイスは、仕 AWS 分け施設に配送され、AWS データセンターに転送されます。UPS からジョブの追跡番号が自動的にレポートされます。

Important

が別途指示しない限り AWS、AWS Snowball Edge デバイスに別の配送ラベルを貼り付けないでください。必ず、デバイスの E Ink ディスプレイに表示される配送情報を使用してください。

Snow Family 製品に対する UPS サービスは、その国の国内でのみ提供されます。

Note

2021年1月以降、英国はEUの一部ではなくなりました。英国と他のEU諸国間の注文は国際注文であり、一般的でないアベイラビリティプロセスは特別な国際プロセスを通じてのみ承認されます。お客様が承認を受け、デバイスをEU加盟国からLHRへ、または英国からEU加盟国に返送する場合は、UPSへの集荷/持ち込みを手配する前に商業送り状が提供されるよう、まず <snowball-shipping@amazon.com> に返送をリクエストする必要があります。

AWS Snowball ブラジルでの集荷

ブラジルでのUPSによるSnowball Edgeデバイスの集荷に関するガイドラインは以下のとおりです。

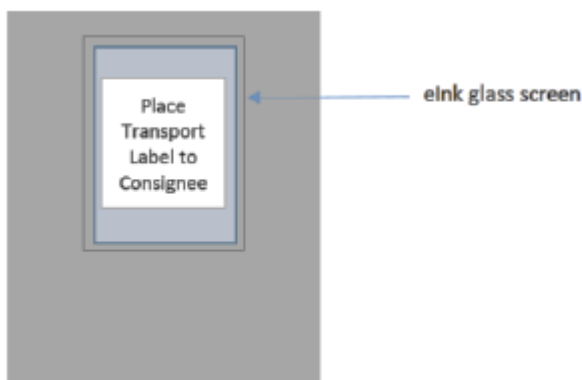
- Snowball Edge デバイスの返送準備ができたなら、0800-770-9035 に電話で連絡し、UPS に集荷を手配します。
- Snowball Edge は、26 の州と連邦直轄区を含むブラジル国内で使用できます。
- すでにお持ちの場合は、ジョブを作成する前に、Cadastro Nacional de Pessoa Juridica (CNPJ) ID を確認してください。
- 適切なドキュメントを発行して、Snowball Edge デバイスを返送する必要があります。Imposto sobre Circulação de Mercadorias e Serviços (ICMS) 登録に従って、お客様の州に必要な書類を税務署に確認してください。
 - サンパウロ市内 – 通常、非 ICMS 申告と電子税申告 (NF-e) が必要です。
 - サンパウロ市外 – 通常、次の書類が必要です。
 - 非 ICMS 宣言書
 - Nota Fiscal Avulsa
 - 電子税金請求書 (NF-e)

Note

非 ICMS 納税者申告の場合、宣言書を4部作成することをお勧めします。1部はお客様の記録用、他3部は移送用です。

AWS Snowball オーストラリアでの の集荷

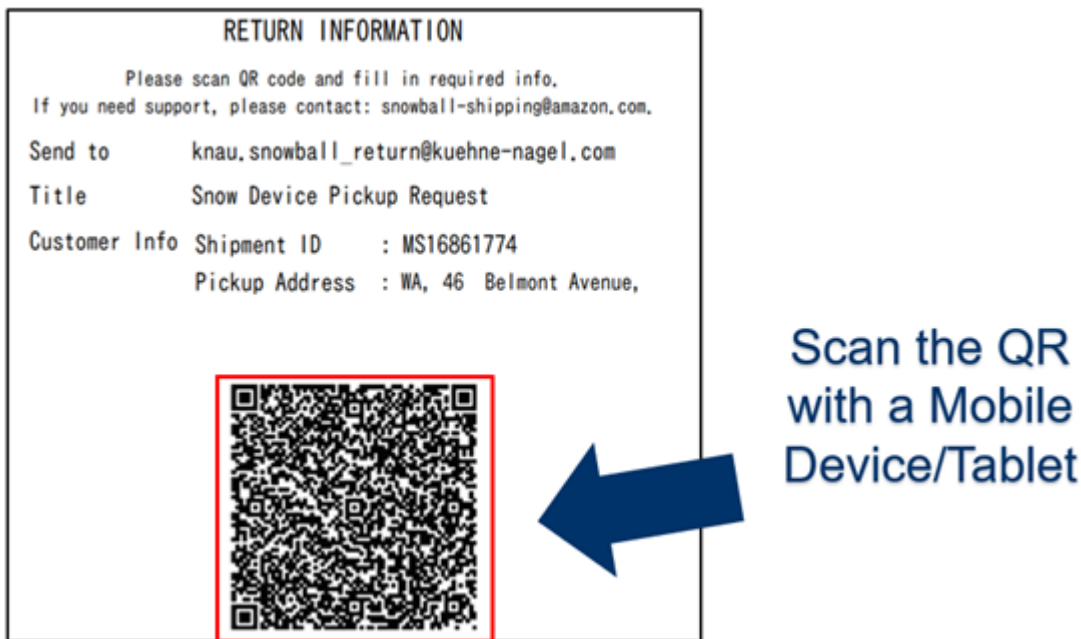
オーストラリアでは、AWS Snowball Edge デバイスを に返送する場合は AWS、Snow デバイスの E Ink ラベルの上に返送用輸送ラベル (これらの手順を含む箱にあります) を置きます。



Note

デバイスと一緒に返送用ラベルを受け取らなかった場合は、デバイスのシリアル番号または参照番号を記載した E メールを knau.snowball_return@kuehne-nagel.com に送信してください。

Snow Family デバイスの返送を手配するには、返品手順書に記載されている QR コードをモバイルデバイスでスキャンしてください。デバイスには、E メールメッセージへのハイパーリンクが表示されます。メッセージには、メールアドレス、件名、管理番号または委託番号などの情報が含まれます。集荷日、名前、連絡先の詳細を記入するか、変更があった場合は新しい集荷住所を入力します。



AWS Snowball インドでの の集荷

インドでは、Blue Dart が Snowball デバイスを集荷します。Snowball デバイスの返送準備ができたら、電源をオフにして、返送出荷の準備をします。集荷を手配するには、件名に Snowball Pickup Request と記載の上、snowball-pickup@amazon.com 宛てに E メールにてご連絡ください。E メールには、以下の情報を記載してください。

- ジョブ ID — に返す Snowball に関連付けられたジョブ ID AWS。
- AWS アカウント ID – ジョブを作成した AWS アカウントの ID。
- 最早集荷時間 (お客様の現地時間) – Snowball の集荷が可能な最早時間。
- 最遅集荷時間 (お客様の現地時間) – Snowball の集荷が可能な最遅時間。
- 特記事項 (オプション) – 集荷を調整するための連絡先の詳細など、Snowball の集荷に関する特記事項。

Snowball チームが Blue Dart に集荷を手配し、お客様に確認メールを送信します。Blue Dart は紙の配送ラベルを提供し、Snowball デバイスを集荷します。

⚠ Important

インドで Snowball を使用する場合は、必ず、すべての税関係書類を管轄の州に提出してください。

AWS Snowball 韓国でのエッジ集荷

韓国では、Kuehne + Nagel が集荷を行います。デバイスを返送する準備ができたなら、「Snowball Pickup Request」という件名で snowball-shipping@amazon.com 宛に E メールを送信してください。集荷日を調整いたします。Eメールの本文には、以下の情報を記載してください。

- ジョブ ID – に返す Snowball に関連付けられたジョブ ID AWS。
- 集荷先住所 – デバイスの集荷場所の住所。
- 集荷日 – デバイスの集荷を希望する最早日。
- 連絡先の詳細 – Kuehne + Nagel が必要に応じて連絡を取るために使用できる名前、E メールアドレス、国内電話番号。

すぐに Snowball チームからフォローアップ E メールが届きます。この E メールには、お知らせいただいた住所での集荷に関する情報が含まれています。デバイスの電源を入れ直します。通常 1,300 ~ 1,500 で集荷可能になります。

AWS Snowball 香港でのエッジ集荷

香港では、S.F. Express が集荷を行います。デバイスを返送する準備ができたなら、件名に Snowball Pickup Request を記載した E メールを snowball-shipping-ap-east-1@amazon.com に送信して、集荷をスケジュールできるようにします。Eメールの本文には、以下の情報を記載してください。

- ジョブ ID
- AWS アカウント ID
- 問い合わせ名
- 連絡先電話番号
- 連絡先 E メールアドレス
- デバイスの集荷を希望する日
- 最早集荷時間
- 最遅集荷時間
- 集荷先住所

S.F. Express にお申し込み後、集荷日時を変更することはできません。

デバイスは S.F. Express AWS によって に配信されます。S.F. Express 返送時の追跡番号は、申し込み時に通知されます。

AWS Snowball シンガポール、日本、インドネシアでの集荷

シンガポール、日本、インドネシアでは、デバイスを返送する準備ができたなら、E Ink の返送ラベルに表示されている QR コードを携帯電話でスキャンします。E メールテンプレートが表示されます。集荷の日時と連絡先の詳細を記入してください。

RETURN

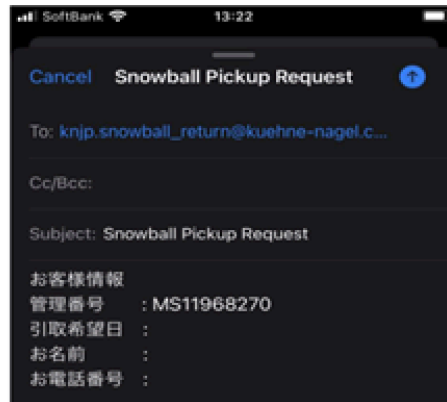
AWS Jobs ID QF6LNZGKTZPF
シリアル番号 2R 207138750022
管理番号 MS14003547



返送のご案内

“以下のQRコードをスキャンし情報を入力の上、メールにてご連絡をお願い致します。”

送信先アドレス knjp.snowball_return@kuehne-nagel.com
件名 Snow Ball Pickup Request
お客様情報
管理番号 : MS14003547
引取希望日 : 要記入
お名前 : 要記入
お電話番号 : 要記入



Note

集荷住所がデバイスの配送先住所と異なる場合は、指定された配送業者に通知できるように、E メール本文に新しい住所を追加してください。

Note

日本では、配送業者により 120.00 USD の配送料が請求されます。料金の説明には Snowball と記載されますが、この料金はすべての Snow Family デバイスの配送に適用されます。

AWS Snowball アラブ首長国連邦ドバイでの の受け取りと返却

ドバイでの AWS Snowball Edge デバイスの受け取りと返送に関するガイドラインは以下のとおりです。

Snowball Edge デバイスの受け取り

Snowball Edge デバイスをフリーゾーンで受け取る場合、UPS から荷物の配送準備が整ったという通知を受けたら、フリーゾーンのゲートパスを申請、取得、共有します。

フリーゾーンまたは本土にいる場合は、デバイスの受け取り時に配達証明 (POD) に署名してください。

Snowball Edge デバイスの返送

Snowball Edge デバイスを返送する際は、600 544 743 で、または UPS のウェブサイトから UPS と直接デバイスの集荷を手配します。デバイスの集荷前に、E Ink ディスプレイに返送先情報が表示されていることを確認してください。[Snowball Edge デバイスの返却](#) を参照してください。フリーゾーンでは、デバイスの集荷に UPS ドライバーが割り当てられたという通知を受けたら、フリーゾーンのゲートパスを申請、取得、共有します。

E Ink ディスプレイの UPS 前払い配送ラベルに、Snowball Edge デバイスの正しい返送先住所が記載されています。

Snowball Edge デバイスは の AWS 仕分け施設に配送され、AWS データセンターに転送されます。UPS からジョブの追跡番号が自動的に提供されます。

Important

が別途指示しない限り AWS、Snowball Edge デバイスに別の配送ラベルを貼り付けないでください。必ず、デバイスの E Ink ディスプレイに表示される配送ラベルを使用してください。

Snow Family 製品に対する UPS サービスは、その国の国内でのみ提供されます。

配送速度

利用できる配送速度は国によって異なります。これらの配送速度は、AWS Snowball Edge デバイスを配送する国によって異なります。配送速度は次のとおりです。

- オーストラリア、日本、シンガポール、インドネシア、韓国 — これらの国内での配送の場合、1 ~ 3 日の標準配送速度をご利用いただけます。
- ブラジル – ブラジル国内に配送する場合、UPS Domestic Express Saver をご利用いただけます。この配送方法では、2 営業日以内に配送されます。配送日数は、州境での遅延の影響を受ける可能性があります。

- EU – EU 各国内への配送には、至急便を利用できます。通常、エクスプレス配送された AWS Snowball Edge デバイスは約 1 日で配送されます。また、EU のほとんどの国で通常便も利用できます。こちらは通常片道 1 週間以内です。
- 香港 – 香港内部の配送には、至急便を利用できます。
- インド – インド国内の配送の場合、AWS が関連する税務書類をすべて受領してから 7 営業日以内に Snowball Edge デバイスが発送されます。
- ドバイ、アラブ首長国連邦 – 配送業者の Express Saver 配送を利用できます。
- 英国 (UK) – 英国国内での配送には、至急便を利用できます。通常、至急便で発送された Snowball Edge デバイスは約 1 日で配送されます。また、通常配送も利用できます。こちらは通常、片道 1 週間以内に配送されます。
- アメリカ合衆国 (米国) およびカナダ – 米国およびカナダ内の配送の場合、翌日配送と翌々日配送を利用できます。

インポートステータスのモニタリング

コンソールでインポートジョブのステータスをモニタリングするには、ジョブ AWS リージョンが作成された [AWS Snow ファミリーマネジメントコンソール](#) のにサインインします。表から追跡するジョブを選択するか、表の上部にある検索バーで選択したパラメータでジョブを検索します。ジョブを選択すると、ジョブの詳細情報が表に示され、その中に、ジョブのリアルタイムのステータスを示すグラフが表示されます。

Note

設定したアクセス権限の問題により Snow デバイスからデータセンターにデータをインポートできない場合、AWS はその旨をお客様に通知するよう試みます。通知を受けた日から 30 日以内に問題を解決してください。問題が解決されない場合、AWS Snow Family ジョブをキャンセルし、デバイスからデータを削除することがあります。

デバイスが に到着すると AWS、ジョブのステータスは転送中から AWS AWS に変わります。Amazon S3 へのデータのインポートが開始されるまで、平均で 1 日かかります。開始されると、ジョブのステータスが [Importing] に変わります。が Snow Family デバイスからデータをインポート AWS するまでに、Snow Family デバイスに移動するのとほぼ同じ時間がかかります。データがインポートされると、ジョブのステータスは完了ステータスに変わります。

これで、を使用した Amazon S3 への最初のデータインポートジョブ AWS Snowball が完了しました。データ転送に関するレポートはコンソールから取得できます。コンソールからこのレポートにアクセスするには、ジョブをテーブルから選択して展開し、詳細情報を表示します。[Get report] を選択して、ジョブ完了レポートを PDF ファイルでダウンロードします。詳細については、「[ジョブ完了レポートとログの取得](#)」を参照してください。

Next: [ジョブ完了レポートとログの取得](#)

ジョブ完了レポートとログの取得

Snow Family デバイスを使用して Amazon S3 にデータをインポートしたり、Amazon S3 からデータをエクスポートしたりすると、ダウンロード可能な PDF ジョブレポートが表示されます。インポートジョブの場合、このレポートはインポートプロセスの最終段階で使用できるようになります。エクスポートジョブの場合、通常、ジョブパートの AWS Snowball Edge デバイスが配信されている間、ジョブレポートが使用可能になります。ジョブ完了レポートは、ローカルのコンピューティングおよびストレージのみのジョブでは使用できません。

ジョブレポートには、Amazon S3 データ転送の状態の詳細が記載されています。レポートには、お客様の記録用にジョブまたはジョブパートの詳細が含まれています。ジョブレポートには、デバイスと Amazon S3 の間で転送されたオブジェクトの総数および総バイト数の高レベルの概要を示すテーブルも含まれています。

転送されたオブジェクトのステータスに関するより詳細な可視記録は、2 つの関連ログ、成功ログと失敗ログで確認できます。これらのログはカンマ区切り値 (CSV) 形式で保存され、各ログの名前には、そのログが記録したジョブまたはジョブパートのジョブ ID が含まれています。

レポートおよびログは AWS Snow ファミリーマネジメントコンソールからダウンロードできます。以下はサンプルレポートです。

Snow Family Job Completion Report



Region: us-gov-east-1(OSU)

Job ID: JIDd6d95004-fe1a-42d3-895d-684f357ef840

Snow Device Serial ID: 207117851234

Job type: IMPORT

Device type: Snowball Edge Storage Optimized

Storage type: S3

Job creation date: 2022-06-02 19:32:27.831 GMT

Job state: Completed

Customer address:

123 Any Street
Any Town, USA

Transfer details:

Transfer type	Total	Success	Failed
Objects	2,635	2,635	0
Bytes	32.2 TB	32.2 TB	0 B

Job state transition details:

The job was created on 2022-06-02 19:32:27.831 GMT
 The snowball got allocated on 2022-06-06 19:10:43.670 GMT
 The snowball was shipped on 2022-06-07 21:59:50.937 GMT
 The snowball was at customer on 2022-06-08 14:04:45.856 GMT
 The snowball was shipped to AWS on 2022-06-28 20:57:42.246 GMT
 The snowball was at our sorting facility on 2022-06-29 14:06:20.737 GMT
 The snowball was at AWS on 2022-06-30 23:12:45.017 GMT
 The data transfer started on 2022-06-30 23:21:34.805 GMT
 The data transfer was completed on +54473-09-10 22:23:46 GMT

Please review your job's status from the console.

For Snow job details, please see: <https://docs.aws.amazon.com/snowball/>

ジョブレポートやログを取得するには

1. にサインイン AWS Management Console し、 を開きます [AWS Snow ファミリーマネジメントコンソール](#)。
2. テーブルからジョブまたはジョブパートを選択し、ステータスペインを展開します。

ジョブレポートやログの取得についての 3 つのオプションが表示されます。[Get job report]、[Download success log]、[Download failure log] です。

3. ダウンロードするログを選択します。

次のリストは、レポートに使用される値について説明しています。

- 完了済み - 転送が正常に完了しました。成功ログで詳細情報を確認できます。
- エラーで完了 - 一部またはすべてのデータが転送されませんでした。失敗ログで詳細情報を確認できます。

Next: [AWS Snowball エッジデバイスの使用](#)

を使用した大規模なデータ移行 AWS Snow Family devices

オンプレミスの場所から大規模なデータを移行するには、データが AWS に正常に移行されるようにするために、慎重な計画、オーケストレーション、実行が必要です。

移行を開始する前にデータ移行戦略を立てて、期限を過ぎたり、予算を超過したり、移行が失敗したりする可能性を避けることをお勧めします。AWS Snow サービスは、の Snow Family Large Data Migration Manager (LDMM) 機能を使用して、大規模なデータ移行プロジェクトを配置、注文、追跡するのに役立ちます AWS Snow ファミリーマネジメントコンソール。

「[大規模な転送の計画](#)」および「[大規模な転送のキャリブレーション](#)」のトピックで手動のデータ移行プロセスについて説明しています。Snow Family LDMM 移行プランを使用すると、手動による手順を効率化できます。

トピック

- [大規模な転送の計画](#)
- [大規模な転送のキャリブレーション](#)
- [大規模データ移行プランの作成](#)
- [大規模データ移行プランの使用](#)

大規模な転送の計画

以下のセクションのガイドラインに従って、オンサイトの AWS Snowball Edge デバイスとサーバー間の大規模なデータ転送を計画し、調整することをお勧めします。

トピック

- [ステップ 1: クラウドに移行する内容を把握する](#)
- [ステップ 2: 目標転送速度を計算する](#)
- [ステップ 3: Snow Family デバイスの必要数を決定する](#)
- [ステップ 4: ジョブを作成する](#)
- [ステップ 5: データを転送セグメントに分割する](#)

ステップ 1: クラウドに移行する内容を把握する

を使用して最初のジョブを作成する前に AWS Snow ファミリーマネジメントコンソール、転送する必要があるデータの量、現在保存されている場所、および転送先を必ず評価してください。ペタバイト規模以上のデータを転送する場合、このような準備をしておくことで、Snow Family デバイスの使用を開始する際の作業がはるかに楽になります。

データを AWS クラウド 初めて に移行する場合は、クラウド移行モデルを設計することをお勧めします。クラウド移行は一晩では達成できません。すべてのシステムが期待どおりに動作するには、入念な計画プロセスが必要です。

このステップが終わると、クラウドに移行するデータの総量がわかります。

ステップ 2: 目標転送速度を計算する

各サーバーに接続されている Snow Family デバイスへのデータ転送の速度を推定しておくことが重要です。この推定速度 (MB/秒 単位) によって、ローカルネットワークインフラストラクチャを使用してデータソースから Snowball Edge デバイスにデータを転送できる速度が決まります。

Note

大容量のデータを転送する場合は、Amazon S3 データ移行を使用することをお勧めします。AWS Snow ファミリーマネジメントコンソールでデバイスを注文する場合は、このオプションを選択する必要があります。

ベースラインの転送速度を特定するには、データの小さなサブセットを Snowball Edge デバイスに転送するか、10 GB のサンプルファイルを転送してスループットを観察します。

目標転送速度を特定するときは、ネットワークの速度、転送するファイルのサイズ、ローカルサーバーからの読み出し速度を変更して、ネットワーク設定を含む環境を調整することでスループットを向上できることに注意してください。Amazon S3 アダプターは、条件が許す限り高速で Snow Family デバイスにデータをコピーします。

ステップ 3: Snow Family デバイスの必要数を決定する

クラウドに移行する予定のデータの合計量、推定転送速度、およびデータを に移動することを許可する日数を使用して AWS、大規模なデータ移行に必要な Snow Family デバイスの数を決定します。デバイスのタイプに応じて、Snowball Edge デバイスの使用可能なストレージ容量は約 39.5 TB、80

TB、210 TB です。例えば、300 TB のデータを 10 AWS 日に以上 に移動し、転送速度が 250 MB/秒の場合、4 台の Snowball Edge デバイスが必要です。転送するデータが 40 TB 未満の場合、AWS Snowcone デバイス (使用可能な領域が 14TB) が推奨されます。

Note

AWS Snow Family devices LDMM には、同時にサポートできる の数を見積 AWS Snow Family devices もるウィザードが用意されています。詳細については、「[大規模データ移行プランの作成](#)」を参照してください。

ステップ 4: ジョブを作成する

必要な Snow Family デバイスの台数がわかったら、デバイスごとにインポートジョブを作成する必要があります。Snow Family LDMM を使用すると、複数のジョブの作成が簡単になります。詳細については、「[次のジョブを注文する](#)」を参照してください。

Note

次のジョブを注文すると、推奨ジョブ注文スケジュールから直接プランに自動的に追加できます。詳細については、「[推奨ジョブ注文スケジュール](#)」を参照してください。

ステップ 5: データを転送セグメントに分割する

複数のジョブを使った大規模なデータ転送の場合、ベストプラクティスとして、データを処理しやすいサイズのデータセットに論理的に分割することをお勧めします。これにより、パーティションを 1 つずつ転送することも、複数のパーティションを並列転送することもできます。パーティションを計画するときは、組み合わせたパーティションのデータがそのジョブの Snow Family デバイ스에 収まることを確認してください。例えば、次のいずれかの方法で、転送をパーティションに分割できます。

- 1 台の Snowball Edge にそれぞれ 8 TB のパーティションを 10 個作成できます。
- ファイルが大きい場合は、各ファイルを個々のパーティションにすることもできますが、Amazon S3 のオブジェクトでは 5 TB の制限があります。
- 各パーティションのサイズは同じでなくて構いません。また、個々のパーティションは同じ種類のデータで構成することができます。例えば、あるパーティションには小さいファイル、別のパー

パーティションには圧縮したアーカイブ、また別のパーティションには大きいファイル、といった方法で分けることができます。このアプローチは、ファイルの種類ごとに平均転送速度を特定するうえで役立ちます。

Note

転送する各ファイルに対し、メタデータオペレーションが実行されます。このオーバーヘッドは、ファイルサイズとは無関係に一定です。そのため、小さいファイルを大きいバンドルに圧縮したり、ファイルをバッチ処理したり、大きいファイルを個別に転送したりすることでパフォーマンスが向上します。

セグメントに分けてデータ転送を行うと、転送中に生じた問題をすばやく解決することができます。転送を 1 日以上実行した後で、大規模な異種混在の転送をトラブルシューティングすることは複雑になる可能性があるためです。

ペタバイト規模のデータ転送の計画が完了したら、サーバーから Snow Family デバイスにセグメントをいくつか転送し、速度と合計転送時間を調整することをお勧めします。

大規模な転送のキャリブレーション

データパーティションの代表的なセットを転送することで、転送パフォーマンスを調整できます。定義した複数のパーティションを選択し、Snow Family デバイスに転送します。各操作の転送速度と転送時間の合計を記録します。結果が目標転送速度を下回る場合は、複数のセグメントを同時にコピーできる可能性があります。この場合、データセットを追加したパーティションでキャリブレーションを繰り返します。

データ転送を実行中のすべてのインスタンスの転送速度の合計に見合った増加が見られなくなるまで、キャリブレーション中の並行コピー操作を継続します。最後に実行したインスタンスを終了し、新しい目標転送速度を確認、記録します。

次のいずれかのシナリオを使用してデータを並列転送することで、Snow Family デバイスへのデータを高速化できます。

- 1 台の Snow Family デバイスに対して、1 つのワークステーションで S3 アダプターの複数のセッションを使用する。
- 1 台の Snow Family デバイスに対して、複数のワークステーションで S3 アダプターの複数のセッションを使用する。

- 複数の Snow Family デバイスを対象とする S3 インターフェイスの複数のセッションを使用する (1 台または複数のワークステーションを使用)。

これらのステップを実行すると、Snow Family デバイスへのデータ転送の速度を把握できます。

大規模データ移行プランの作成

AWS Snow Family 大規模なデータ移行計画機能を使用すると、複数の Snow Family サービス製品を使用して、500 TB から複数のペタバイトへの大規模なデータ移行を計画、追跡、モニタリング、管理できます。

大規模なデータ移行計画機能を使用して、移行するデータのサイズ AWS や、データの同時移行に必要な Snow Family デバイスの数など、データ移行の目標に関する情報を収集します。このプランを使用して、データ移行プロジェクトの予定スケジュールと、目標を達成するための推奨ジョブ注文スケジュールを作成します。

Note

現在、データ移行プランは 500 TB を超えるインポートジョブで利用できます。

トピック


- [ステップ 1: 移行の詳細を選択する](#)
- [ステップ 2: 配送、セキュリティ、通知の設定を選択する](#)
- [ステップ 3: プランを確認して作成する](#)

ステップ 1: 移行の詳細を選択する

Note

大規模データ移行プランは、500 TB を超えるデータ移行で利用できます。500 TB 未満のデータ転送プロジェクトでは、Snow Family デバイスで個別にジョブ注文を作成します。詳細については、このガイドの [「Snow Family デバイスを注文するジョブの作成」](#) を参照してください。

1. [AWS Snow ファミリーマネジメントコンソール](#)にサインインします。この AWS Snow ファミリーマネジメントコンソールで初めて使用する場合は AWS リージョン、AWS Snow Family ページが表示されます。それ以外の場合は、既存のジョブのリストが表示されます。
2. 初めてのデータ移行プランを行う場合は、メインページから [大規模データ移行プランの作成] を選択します。それ以外の場合は、[大規模データ移行プラン] を選択します。[データ移行プランを作成] を選択し、プラン作成ウィザードを開きます。
3. [データ移行プランに名前を付ける] で、データ移行プランの名前を入力します。プラン名の最大長は 64 文字です。有効な文字は、A-Z、a-z、0-9、および . _ (ハイフン) です。プラン名の最初に **aws:** を付けることはできません。
4. 移行するデータの合計に AWS、に移行するデータの量を入力します AWS。
5. [Snow デバイス] で Snow Family デバイスを選択します。

 Note

サポートされているデバイスオプションは、特定の AWS リージョンの可用性によって異なる場合があります。

Snow devices <small>Info</small>					
	Name	Compute	Memory	Storage (HDD)	Storage
<input checked="" type="radio"/>	Snowcone	2 vCPUs	4 GB	8 TB	-
<input type="radio"/>	Snowcone SSD	2 vCPUs	4 GB	-	14 TB
<input type="radio"/>	Snowball Edge Compute Optimized	52 vCPUs	208 GB	39.5 TB	7.68 TB
<input type="radio"/>	Snowball Edge Compute Optimized with GPU	52 vCPUs, GPU	208 GB	39.5 TB	7.68 TB
<input type="radio"/>	Snowball Edge Compute Optimized	104 vCPUs	416 GB	-	28 TB

6. [同時実行デバイス] では、現在の場所で同時にデータをコピーできる Snow Family デバイスの数を入力します。よくわからない場合は、次のセクションに進み、同時実行デバイス見積もりウィザードを使用してこれを判断する方法を確認してください。
7. [次へ] をクリックします。

同時実行デバイス推定ウィザードの使用

同時実行デバイス推定ウィザードは、大規模なデータの移行中に使用できる同時実行デバイスの数を判断するのに役立ちます。

前提条件:

- 概念実証を実施して、環境内の Snow Family デバイスを使用してデータ転送方法をテストし、パフォーマンスを測定済みであること。
- ネットワークとバックエンドストレージへの接続について理解していること。

ステップ 1: データソース情報を入力する

まず、ストレージソースからデータをコピーする際の理論上の最大スループットを特定します。

1. [移行データの合計] に、移行する予定のデータ量を入力します。

[単位] で、移行するデータ量の測定単位 (GB または TB) を選択します。

2. [アクティブなネットワークインターフェイスの数] に、ストレージソースからのデータ移行で使用可能なアクティブなネットワークインターフェイスの数を入力します。

Number of active network interfaces [Info](#)

The number of network interfaces that can be used for migrations

Number of active network interfaces used for data migration

3. [ネットワークインターフェイス速度] では、ストレージソースのネットワークインターフェイスの速度を選択します。ネットワーク速度の単位は Gb/秒です。

Network interface speed [Info](#)

The speed of the network interfaces used for migrations

Network interface speed (Gb/s)

4. [最大ネットワークスループット] には、概念実証で特定された、ストレージソースに対するテスト済みの最大ネットワークスループットを入力します。スループットの単位は MB/秒です。

Maximum network throughput [Info](#)

The maximum sustainable throughput for the data source

Maximum tested throughput of data source (MB/s)

5. [ストレージバックエンドネットワークの使用] では、ストレージソースがバックエンドストレージとネットワークを共有しているかどうかを指定します。
- ネットワークが共有されていない場合は [はい] を選択します。1 つのストリームのストレージ相互接続の速度を入力する必要はありません。
 - ネットワークが共有されている場合は [いいえ] を選択します。1 つのストリームのストレージ相互接続の速度を MB/秒単位で入力します。

選択した内容に基づいて、ウィザードはページの下部にある [データソースの最大移行スループット (MB/秒)] 値を更新します。

Storage backend network usage [Info](#)

Network shared with storage backend traffic?
Is the network used for migration being shared with your storage backend?

Yes

Speed of storage interconnection for single stream (MB/s)
This is a single connection throughput that can be sustained from source to destination

6. [次へ] をクリックします。

ステップ 2: 移行ワークステーションパラメータを入力する

Snow Family デバイスをストレージソース (Microsoft Windows サーバーなど) に直接接続できます。代わりに Snow Family デバイスを 1 台または複数のワークステーションに接続して、ストレージソースからデータをコピーすることもできます。

1. [移行ワークステーションの使用] では、ワークステーションの使用を選択します。
 - ワークステーションを使用せずにデータソースから直接データを転送する場合は、[なし - データソースを直接使用] を選択して、[次へ] を選択します。
 - データ転送に 1 台または複数のワークステーションを使用する場合は、[その他 - コピーワークステーションを使用] を選択します。

Migration workstation usage [Info](#)

Type of migration source used

Other - Use copy workstation(s) ▼

2. [アクティブなネットワークインターフェイスの数] に、データ移行に使用するポートの数を入力します。

Number of active network interfaces [Info](#)

The number of network interfaces that can be used for migrations

Number of active network interfaces on the migration workstation

1

3. [ネットワークインターフェイス速度] では、ネットワークインターフェイスの速度 (GB/秒) を選択します。

Network interface speed [Info](#)

Your workstations Network card speeds

Network interface speed (Gb/s)

10 ▼

4. [ストレージバックエンドネットワークの使用] では、ワークステーションのあるネットワークがバックエンドストレージと共有されているかどうかを指定します。
 - 共有されている場合は [はい] を選択します。

- 共有されていない場合は [いいえ] を選択します。1 つのストリームのストレージ相互接続の速度を MB/秒単位で入力します。

Storage backend network usage [Info](#)

Network shared with storage backend traffic?
Is the network used for migration being shared with your storage backend?

Yes ▼

Speed of storage interconnection for single stream (MB/s)
This is a single connection throughput that can be sustained from source to destination

入力内容に基づいて、ウィザードは [移行ワークステーションの数] に推奨数を表示します。推奨数に同意しない場合は、手動で数を変更できます。この数は、大規模データ移行プランの [同時実行デバイス] に表示されます。

Number of migration workstations [Info](#)

Recommended number of migration workstations used

ステップ 3: Snow Family デバイスの平均転送スループットを入力する

- [Snow デバイスの平均転送スループット] フィールドに、概念実証で確認した転送スループットを MB/秒単位で入力します。

Average Snow device transfer throughput [Info](#)

This is the throughput from your migration workstation to the Snow device you saw during the proof of concept

Average Snow device transfer throughput (MB/s)

平均スループットに基づいて、ウィザードは移行プランの詳細にある [推奨される同時実行 Snow デバイス数] と [同時実行デバイスの最大数] を更新します。

2. [この数を使用] を選択して続行し、移行の詳細の選択に戻ります。[次へ] を選択して、次のステップ ([ステップ 2: 配送、セキュリティ、通知の設定を選択する](#)) に進みます。

Note

最大 5 台の Snow デバイスを同時に使用できます。

ステップ 2: 配送、セキュリティ、通知の設定を選択する

1. [配送先住所] セクションで、既存の住所を選択するか、新しい住所を追加します。

Note

住所に入力する国は、デバイスの配送先の国と一致し、その国で有効なものでなければなりません。

2. [サービスアクセスタイプの選択] で、次のいずれかを行います。
 - Snow Family が、Snow Family ジョブの CloudWatch メトリクスと Amazon SNS 通知を発行するために必要なすべてのアクセス許可を持つ新しいサービスにリンクされたロールを作成できるようにします。
 - 必要なアクセス許可を持つ既存のサービスロールを追加する。このロールの設定方法の例については、「[例 4: 期待されるロールのアクセス許可と信頼ポリシー](#)」を参照してください。
3. [通知を送信] では、通知を送信するかどうかを選択します。[データ移行プランに関する通知を送信しない] を選択すると、このプランからの通知は受信しませんが、ジョブの通知は受信します。
4. [通知を設定] では、
 - [既存の SNS トピックを使用] または
 - [新しい SNS トピックを作成] をクリックします。

ステップ 3: プランを確認して作成する

1. [プランの詳細] と [配送、セキュリティ、通知の設定] で情報を確認し、必要に応じて編集します。
2. [データ移行プランを作成] を選択してプランを作成します。

大規模データ移行プランの使用

大規模データ移行プランを作成したら、作成されたスケジュールとダッシュボードを使用して、残りの移行プロセスを進めることができます。

推奨ジョブ注文スケジュール

AWS Snow Family devices 大規模な移行計画を作成したら、推奨されるジョブの順序付けスケジュールを使用して新しいジョブを作成できます。

Note

データサイズや同時実行デバイスの数を手動で更新すると、スケジュールが調整されます。ジョブが推奨注文日までに注文されなかった場合や、推奨注文日より前に注文された場合、スケジュールは自動的に調整されます。推奨注文日より前にジョブが返送された場合、スケジュールは自動的に調整されます。

Recommended job ordering schedule		Jobs ordered	
Recommended job ordering schedule <small>This list provides an estimated schedule to place Snow Jobs in order to achieve your data migration goals. The estimated ordering schedule is automatically adjusted based on your data migration speed.</small>			
<input type="text" value="Filter by a date and time range"/>		<input type="checkbox"/> Hide Ordered	
Recommended date to order	Number of devices to order	Number of ordered devices	Device type
<input type="radio"/> Thu Mar 23 2023	2	-	Snowball Edge Storage Optimized with 210TB
<input type="radio"/> Fri Mar 31 2023	2	-	Snowball Edge Storage Optimized with 210TB
<input type="radio"/> Sat Apr 08 2023	2	-	Snowball Edge Storage Optimized with 210TB
<input type="radio"/> Sun Apr 16 2023	2	-	Snowball Edge Storage Optimized with 80TB
<input type="radio"/> Mon Apr 24 2023	2	-	Snowball Edge Storage Optimized with 80TB
<input type="radio"/> Tue May 02 2023	2	-	Snowball Edge Storage Optimized with 80TB
<input type="radio"/> Wed May 10 2023	2	-	Snowball Edge Storage Optimized with 80TB
<input type="radio"/> Thu May 18 2023	2	-	Snowball Edge Storage Optimized with 80TB
<input type="radio"/> Fri May 26 2023	1	-	Snowball Edge Storage Optimized with 80TB
<input type="radio"/> Fri May 26 2023	1	-	Snowcone SSD

次のジョブを注文する

次のジョブを注文するには、手動でジョブを作成してプランに追加する代わりに、以前注文したジョブのクローンを作成するか、事前に入力されているジョブを作成するかを選択できます。

ジョブのクローンを作成するには

1. 推奨ジョブ注文スケジュールから次の注文 (ステータスが [未注文] の最初の推奨項目) を選択し、[アクション] メニューから [ジョブのクローンを作成] を選択します。[ジョブのクローンを作成] ウィンドウが表示されます。
2. [ジョブのクローンを作成] ウィンドウの [注文済みジョブ] セクションで、クローンを作成するジョブを選択します。
3. [新規ジョブの詳細] セクションで、注文するデバイスを選択します。選択したデバイスごとに、選択したジョブに基づいて [ジョブ名] が自動的に入力されます。ジョブ名は上書きできます。
4. [確認] を選択して、選択したデバイスのジョブ注文を行います。システムにより各デバイスのジョブのクローンが作成されます。

新しいジョブを作成するには

1. 推奨ジョブ注文スケジュールから次の注文 (ステータスが [未注文] の最初の推奨項目) を選択し、[アクション] メニューから [新しいジョブを作成] を選択します。[新しいジョブを作成] ウィンドウが表示されます。

Recommended date to order	Number of devices to order	Number of ordered devices	Device type	Status
Thu Mar 23 2023	2	-	Snowball Edge Storage Optimized with 210TB	⊗ Not Ordered
Fri Mar 31 2023	2	-	Snowball Edge Storage Optimized with 210TB	⊗ Not Ordered
Sat Apr 08 2023	2	-	Snowball Edge Storage Optimized with 210TB	⊗ Not Ordered
Sun Apr 16 2023	2	-	Snowball Edge Storage Optimized with 80TB	⊗ Not Ordered
Mon Apr 24 2023	2	-	Snowball Edge Storage Optimized with 80TB	⊗ Not Ordered
Tue May 02 2023	2	-	Snowball Edge Storage Optimized with 80TB	⊗ Not Ordered
Wed May 10 2023	2	-	Snowball Edge Storage Optimized with 80TB	⊗ Not Ordered
Thu May 18 2023	2	-	Snowball Edge Storage Optimized with 80TB	⊗ Not Ordered
Fri May 26 2023	1	-	Snowball Edge Storage Optimized with 80TB	⊗ Not Ordered
Fri May 26 2023	1	-	Snowcone SSD	⊗ Not Ordered

2. [デバイスの選択] セクションで、注文するデバイスを選択します。[Continue] を選択します。

The screenshot shows a dialog box titled "Create New Jobs" with a close button (X) in the top right corner. Below the title is a section titled "Device Selection (2/2)" with the instruction "Select which devices you would like to order". There is a list of device options, each with a checked checkbox:

- Device type
- Snowball Edge Storage Optimized with 210TB
- Snowball Edge Storage Optimized with 210TB

At the bottom of the dialog, there are two buttons: "Cancel" and "Continue".

3. [新規作成] ページが表示されます。ジョブタイプ、配送先住所、デバイスタイプなど、ほとんどのパラメータはプランに基づいて設定されます。システムによってデバイスごとにジョブが作成されます。

1 つまたは複数のジョブが正常に作成されたかどうかを確認できます。正常に作成されたジョブは、自動的にプランに追加されます。

注文済みジョブリスト

各プランに注文済みジョブリストが表示されます。このリストは最初は空です。ジョブの注文を開始したら、[アクション] メニューから [ジョブを追加] を選択してプランにジョブを追加できます。ここで追加したジョブは、モニタリングダッシュボードで追跡されます。

同様に、[アクション] メニューから [ジョブを削除] を選択することで、注文済みジョブリストからジョブを削除できます。

データ移行をスムーズに行うため、プランで提供されるジョブ注文スケジュールを使用することをお勧めします。

モニタリングダッシュボード

プランにジョブを追加すると、ジョブが取り込み AWS のために に戻ると、ダッシュボードにメトリクスが表示されます。次のメトリクスによって進捗状況を追跡できます。

- に移行されたデータ AWS – AWS これまでに移行されたデータの量。
- ジョブごとの平均データ移行量 – ジョブ 1 件あたりの平均データ量 (テラバイト単位)。
- Snow ジョブの合計数 – 注文済みの Snowball Edge ジョブの数と、未注文の残りのジョブの数の比較。
- 移行ジョブの平均所要時間 – 1 件のジョブの平均所要時間 (日単位)。
- Snow ジョブのステータス – 各ステータスのジョブ数。

AWS OpsHub for Snow Family を使用してデバイスを管理する

Snow Family デバイスは、デバイスとローカル AWS サービスの管理に使用できる AWS OpsHub for Snow Family 使いやすいツールを提供するようになりました。クライアントコンピュータ AWS OpsHub を使用して、単一またはクラスター化されたデバイスのロック解除と設定、ファイルの転送、Snow Family デバイスで実行されているインスタンスの起動と管理などのタスクを実行します。を使用して AWS OpsHub、Storage Optimized と Compute Optimized Snow の両方のデバイスタイプを管理できます。AWS OpsHub アプリケーションは追加料金なしで利用できます。

AWS OpsHub は、Snowball API で利用可能な既存のすべてのオペレーションを受け取り、グラフィカルユーザーインターフェイスとして表示します。このインターフェイスは、データをにすばやく移行 AWS クラウドし、Snow Family デバイスにエッジコンピューティングアプリケーションをデプロイするのに役立ちます。

AWS OpsHub は、Snow Family デバイスで実行されている AWS のサービスの統合ビューを提供し、を通じて運用タスクを自動化します AWS Systems Manager。を使用すると AWS OpsHub、さまざまなレベルの技術的専門知識を持つユーザーは、多数の Snow Family デバイスを管理できます。数回クリックするだけで、デバイスのロック解除、ファイルの転送、Amazon EC2 互換インスタンスの管理、デバイスメトリクスのモニタリングを行うことができます。

Snow デバイスがお客様のサイトに到着したら、ラップトップなどのクライアントマシンで AWS OpsHub アプリケーションをダウンロード、インストール、および起動します。インストール後、デバイスのロックを解除し、ローカルでサポートされている AWS のサービスを使用して管理を開始できます。AWS OpsHub は、デバイス上のストレージ容量やアクティブなインスタンスなどの主要なメトリクスをまとめたダッシュボードを提供します。また、Snow Family デバイスでサポートされる AWS の一部のサービスも提供されます。数分以内に、デバイスへのファイル転送を開始できます。

トピック

- [Snow Family デバイスの AWS OpsHub をダウンロードする](#)
- [デバイスのロック解除](#)
- [の PGP AWS OpsHub 署名の検証 \(オプション\)](#)
- [デバイスで AWS のサービスの管理](#)
- [デバイスの管理](#)
- [管理タスクの自動化](#)

- [デバイスの NTP タイムサーバーの設定](#)

Snow Family デバイスの AWS OpsHub をダウンロードする

ダウンロードするには AWS OpsHub

1. [AWS Snowball のリソースウェブサイト](#)に移動します。

OpsHub

OpsHub is a graphical user interface you can use to manage Snowball devices. OpsHub makes it easy to setup and manage Snowball devices enabling you to rapidly deploy edge computing workloads and simplify data migration to the cloud. With just a few clicks in OpsHub, you have the full functionality of the Snow Family of devices at your fingertips; you can unlock and configure devices, drag-and-drop data to devices, launch applications, and monitor device metrics.

- [OpsHub documentation](#)

	OpsHub
Windows 7 or higher	Download
Mac OS X 10.10 or higher	Download
Linux (Ubuntu version 14 or higher, and Fedora version 24 or higher)	Download (Signature)

2. AWS OpsHub セクションで、オペレーティングシステムのダウンロードを選択し、インストール手順に従います。

デバイスのロック解除

デバイスがサイトに到着したら、まず接続してロックを解除します。AWS OpsHub では次の方法を使用して、デバイスのサインイン、ロック解除、管理を行うことができます。

- ローカル — デバイスにローカルでサインインするには、デバイスの電源を入れ、ローカルネットワークに接続する必要があります。次に、ロック解除コードとマニフェストファイルを入力します。
- リモート — デバイスにリモートでサインインするには、デバイスの電源を入れ、デバイスがネットワークを介して *device-order-region*.amazonaws.com に接続できることを確認します。

次に、デバイスにリンク AWS アカウント されている の AWS Identity and Access Management (IAM) 認証情報 (アクセスキーとシークレットキー) を指定します。

リモート管理を有効にし、関連付けられたアカウントを作成する方法については、「[Snow Device Management のアクティブ化](#)」を参照してください。

トピック

- [デバイスをローカルでロック解除する](#)
- [デバイスをリモートでロック解除する](#)

デバイスをローカルでロック解除する

デバイスをローカルで接続してロック解除するには

1. デバイスのフラップを開き、電源コードを探し、コードを電源に接続します。
2. ネットワークケーブル (通常はイーサネット RJ45 ケーブル) を使用してデバイスをネットワークに接続し、前面パネルを開いてデバイスの電源を入れます。
3. AWS OpsHub アプリケーションを開きます。初めて使用するときは、言語の選択を求められます。次いで、[次へ] を選択します。
4. 「使用開始 OpsHub」 ページで、「ローカルデバイス にサインイン」を選択し、「 にサインイン」を選択します。



Get started with OpsHub

Sign into local devices
You'll need an unlock code and manifest file

Sign into remote devices
You'll need an access key & secret key

Sign in

5. [ローカルデバイスにサインイン] ページで、お使いの Snow Family デバイスタイプを選択して [サインイン] を選択します。
6. [サインイン] ページで、デバイスの IP アドレスとロック解除コードを入力します。デバイスマニフェストを選択するには、[Choose file] (ファイルの選択)、[Sign in] (サインイン) の順に選択します。



Sign into your Snowball Edge

Sign in with an unlock code and manifest file


Device IP address

Eg 12.34.45.678

Unlock code

7c0e1-bab84-f7675-0a2b6-bfcc3

Manifest file

 Choose file

No file chosen

Back

Sign in

7. (オプション) デバイスの認証情報を [profile] (プロファイル) として保存します。プロファイルに名前を付け、[プロファイル名を保存] を選択します。プロファイルの詳細については、「[プロファイルの管理](#)」を参照してください。
8. ローカルデバイス タブでデバイスを選択すると、デバイスで実行されているネットワークインターフェイスや AWS サービスなどの詳細が表示されます。このタブからクラスターの詳細を表示したり、AWS Command Line Interface () と同様にデバイスを管理したりすることもできます。AWS CLI。詳細については、「[デバイスで AWS のサービスの管理](#)」を参照してください。

AWS Snow Device Management がインストールされているデバイスの場合、リモート管理を有効にするを選択して機能をオンにできます。詳細については、「[AWS Snow Device Management を使用してデバイスを管理する](#)」を参照してください。

デバイスをリモートでロック解除する

Snow Family デバイスのロックを解除するには

デバイスを接続してリモートでロックを解除するには

1. デバイスのフラップを開き、電源コードを探し、コードを電源に接続します。
2. イーサネットケーブル (通常は RJ45 ケーブル) を使用してデバイスをネットワークに接続し、フロントパネルを開いてデバイスの電源を入れます。

Note

リモートでロックを解除するには、デバイスが *device-order-region*.amazonaws.com に接続できる必要があります。

3. AWS OpsHub アプリケーションを開きます。初めて使用するときは、言語の選択を求められます。次いで、[次へ] を選択します。
4. 「使用開始 OpsHub」ページで、「リモートデバイスにサインイン」を選択し、「 にサインイン」を選択します。



Get started with OpsHub

Sign into local devices
You'll need an unlock code and manifest file

Sign into remote devices
You'll need an access key & secret key

Sign in

5. [Sign in to remote devices] (リモートデバイスにサインインする) ページで、AWS アカウントの AWS Identity and Access Management (IAM) 認証情報 (アクセスキーおよびシークレットキー) を入力し、[Sign in] (サインイン) を選択します。



Sign into remote devices

Sign in with an access key and secret key

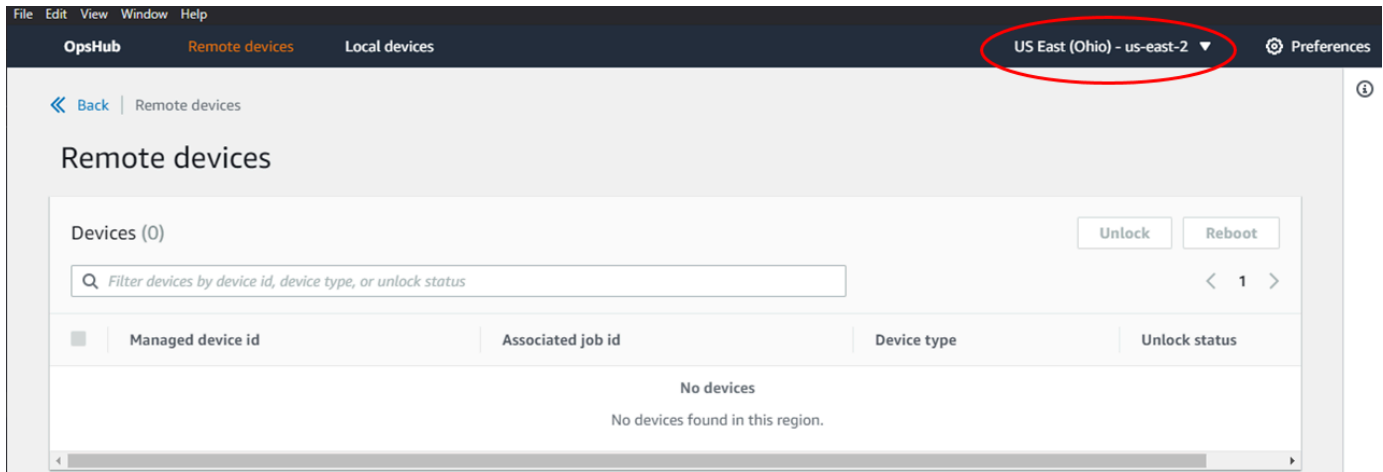
Access key

Secret key

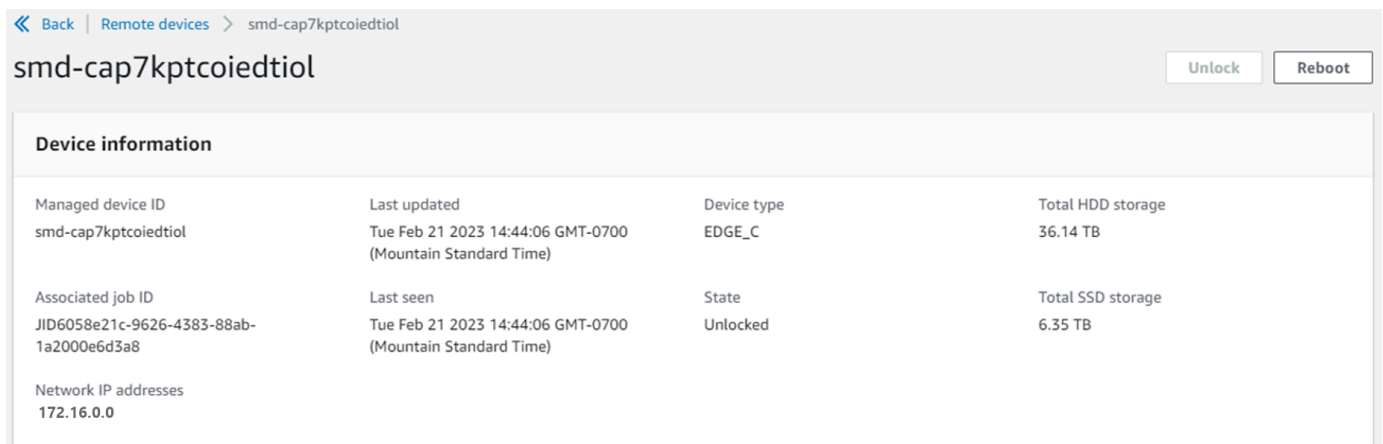
Back

Sign in

6. [リモートデバイス] タブの上部で、リモートでロック解除する Snow デバイスのリージョンを選択します。



7. [Remote devices] (リモートデバイス) タブでデバイスを選択し、状態やネットワークインターフェイスなどの詳細を表示します。次に、[Unlock] (ロック解除) を選択して、デバイスのロックを解除します。



リモートデバイスの詳細ページから、AWS Command Line Interface () と同様にデバイスを再起動して管理することもできますAWS CLI。別の でリモートデバイスを表示するには AWS リージョン、ナビゲーションバーで現在のリージョンを選択し、表示するリージョンを選択します。詳細については、「[デバイスで AWS のサービスの管理](#)」を参照してください。

の PGP AWS OpsHub 署名の検証 (オプション)

Linux オペレーティングシステム用の AWS OpsHub アプリケーションインストーラパッケージは、暗号で署名されています。公開キーを使用して、インストーラパッケージがオリジナルで、変更されていないことを検証できます。ファイルに損傷や変更がある場合、検証は失敗します。GNU Privacy Guard (GPG) を使用して、インストーラパッケージの署名を検証できます。この検証はオプションです。アプリケーションの署名検証の選択は、いつでも実行できます。

Linux オペレーティングシステムインストーラの SIGNATURE ファイルは、[AWS Snowcone リソース](#) または [Snowball Edge リソース](#) からダウンロードできます。

Linux オペレーティングシステムで AWS OpsHub インストールパッケージを確認するには

1. 次のパブリックキーをコピーしてファイルに保存し、ファイルに名前を付けます。例えば `opshub-public-key.pgp` です。

```
-----BEGIN PGP PUBLIC KEY BLOCK-----
xsFNBF/hGf8BEAC9HCDV8uljDX02Jxspi6kmPu4xqf4ZZLQsSqJcHU61oL/c
/zAN+mUqJT9aJ1rr0QFGVD1bMogecUPf1TW1DkEEpG8ZbX5P8vR+EE10/rW/
WtqizSudy6qy59ZRK+YVSDx7DZyuJmI07j00UADCL+95ZQN9vqwHNjBHsgfQ
l/1Tqhy81ozTZXCi/+u+99YLaugJIP6ZYIEdfpxnghqyVtaappBFTAyfG67Y
N/5mea1VqJzd8liFpIFQn1+X7U2x6emDbM01yJWV3aMmPwhtQ7iBdt5a4x82
EF5bZJ8HSRMvANDILD/9VTN8VfUQGKFjFY2GdX9ERwvfTb47bbv9Z28V1284
4lw2w1B1007Fo02v/Y0ukrN3VHCpmJQ51IiqZbYRa0DVK6UR5QNvUlj5fwWs
4qW9UDPHt/HDuaMrMFCejEn/7wvRUrGvtzCT9F56A1/dwRSxBejQQEb1AC8j
uuyi7gJaPdyNntR0EFTD7i02L6X2jB4YLfvGxP7Xeq1Y37t8NKF8CYTp0ry/
Wvw0iKZFbo4AkiI0aLyBCK9HBXhUKa9x06g0nhh1UFQrPGrk60RPQKqL76HA
E2ewzGDa90w1RBuAt2nRQpyNYjoASBvz/cAr3e0nuWsIzopZIenrxI5ffcjY
f6UWA/OK3ITHtYHewVhseDyEqTQ4MUIWQs4NAwARAQABzTlBV1MgT3BzSHVi
IGZvciBTbm93IEZhbWlseSA8YXdzLW9wc2h1Yi1zaWduZXJAYW1hem9uLmNv
bT7CwY0EEAEIACAFA1/hGf8GCwkHCAMCBBUICgIEFgIBAAIZAQIbAwIeAQAh
CRAhgC9adPNF8RYhBDcvpelIaY930b0vqiGBz1p080XxGbcP+gPZX7LzKc1Y
w9CT3UHgkAIaw0SXYktujzoYVxAz8/j3jEkCY0dKnfyqvWZDiJAXnzmXWwBg
cxg1g0GXNXCM4lAd68CmbA0LoLTaWSQX30ZbswzhbtX2ADAlOpV8RLBik7fm
bS9FYuubDRhfYRQq0fPjUGXFiEgwg6aMFxsrGLlv4QD7t+6ftFIe/mxLbjR4
iMgtr8FIPXbgn05YYY/LeF4NIgX4iLEqRbAnfWjPzqQ1spFWAotIzDmZqby+
WdWThrH4K1rwtYM8sDhqRnMnqJrGFZzk7aDhVPwF+F0VMmPeEN5JRazEeUr1
VZaSw6mu0n4FMGSXuwGgdvmkqnMe6I5/xLdU4IOPNhp0UmakDW0q/a1dREDE
ZLMQDMINphmeQno4inGmwbRo63gitD4ZNR5sWwfuwty251o8Ekv7jkkp3mSv
pdxn5tptttnPaSPcSIX/4ED119Tu0i7aup+v30t7eikYDSZG6g9+jHB3Va9e
/VWShFSgy8Jm2+qq/ujUQDAGTcfSuY9jg1ITsog6ayEza/2upDJ1m+40HK4p
8DrEzP/3jTahT8q5ofFWSRDL17d31TSU+JBmPE3mz311FNXgi08w+taY320z
+irHtb3iSiiukbjS8s0maVgzszRqS9mhaEn4LL0zoqrUicmXgTyFB7n2LuYv
07vxM05xxhGQwsF2BBABCAAJBQJf4RoCAhsDACEJEBFVzT/tDi5FiEEi+09
V+UAYN9Gnw36EVm/NP+00LnnEQ/+J4C0Mn8j0AebXrwBiFs83sQo2q+WHL1S
MRc1g5gRFDXs6h1Gv+TGXRen7j1oeaddWvg0tUBxqmC0jr+8AKH00tiBWSu0
lsS8JU5rindEsKUrKTwcG2wyZFoe1z1E8xPkLRSRN5ZbbgKsTz1611HgCCId
Do+WJdDkWGwXmtDvzjM32EI/PVBd108ga9aPwXdhLw0dKAjZ4JrJXLUQJjRI
IVDSyM0bEH0UM6a/+mWNZazNfo0LsGWqGva6Xn5WJWlR1S78vPNf03BQYU0
YRjaVQR+kPtB9aSAZni5sWfk6NrrNd1Q78d067uhhejsjRt7Mja2fEL4Kb1X
nK4U/ps7X103o/VjblneZ0hJK6kAKU172tnPJTJ31Jb0xX73wsMWDYZRZVcK
```

```

9X9+GFrpwhKHwKKPjpm0t/FRxNepvqR172TkgBPqGH2TM0FdB1f/uQprivqge
PBbS0JrmBIH9/anIqgtMdtcNQB/0erLdCdQI5af0uD10LcLwdJwG9/bSrfwT
TVEE3WbXmJ8pZgmZlHUIzE6V2DSadV/YItk50I0j jr0VH0Hv1FMwGCEAIFzf
9P/pNi8hpEmLRphRi0VVcdQ30bH0M0gPHu5V9f1IhyCL1zU3LjYTHkq0yJD5
YDA1x01MYq3DcSM5130VBbLmuVS2GpcTCYq1gQA6h/zzMwz+/70wU0EX+EZ
/wEQA0AY8ULmcJIQWIr14V0jy1pJeD3qwj7wd+QsBzJ+m0p0B/3ZFAhQiN01
9yCD1HeiZeAmWYX90IXrNiIdcHy+WTAp4G+NaMpqE52qhbDjz+IbvLp11yDH
bYEHpjnthXEy21bvkAJ0Kkw/2RcQ0i4dodGnq5icyYj+9gcuHvnVwbrQ96Ia
0D7c+b5T+bzFqk90nIcztmrRuhDLJnJpi70jpvQwfq/TkkZA+mzupxfSkq/Y
N9qXNEToT/VI2gn/LS0X4Ar112KxBjzNESQkwGSiWSYtMA5J+Tj5ED0uZ/qe
omNb1A1D4bm7Na8NAoLxCTAiDq/f3To9Xb181Hsnd0mfLCb/BVgP4edQKTii
C/OZH9QJlfnM0aq7JVLQAuvQNEL88RKW6YZBqkPd3P6zdc7sWDLTMXM0d3I
e6NUvU7pW0E9NyRfUF+oT4s9wAJhAodinAi8Zi9rEfhK1VCJ76j7bcQyZe0
jXD3IJ7T+X2XA8M/BmypwMw0Soljzhwh044RAasr/fAzpKNPB318JwcQunIz
u2N3CeJ+zrsomjcPxzehwsSVq11zaL2ureJBL0KkBgYxUJYXpbS01ax1TsFG
09ldAN0s9Ej8CND37GsNnuygj0gWXbX6MNgbvPs3H3zi/AbMunQ1VB1w07JX
zdM1hBQZh6w+NeiEsK1T6wHi7IhxABEBAAHcWXYEGAEIAAkFA1/hGf8CGwwA
IQkQIYHPWnTzRfEWIQQ3L6XpSGmPd9Gzr6ohgc9adPNF8TMBD/9TbU/+PVbF
ywKvwi3GL01pY7BXn81QaHyunMGUavm080faRR0ynkH0ZqLHCp6bIajF0fvF
b7c0Jamzx8Hg+SIdl6yRpRY+fA4RQ6PNnmT93ZgWW3EbjPyJGlm0/rt03SR
+0yn4/ldlg2KfBX4ppMoPCMKUdWxGrimDETXsGihwZ0gmCZqXe81K122PYkSN
JQQ+LlfjKvCaxfPKEjXYTbIbfyyhCR6NzA0VZxCrzSz2xDrYWp/V002K1xda
0ix6r2aEHf+xYEuh0aBt80HY5nXTuRReCVU789MUVtCMqD2u6amdo4BR0kWA
QNg4yavKwV+LVtyYh2Iju9VSyv4xL1Q4xKHvcAUrSH73bHG7b7jkUJckD0f4
twhjJk/Lfwe6RdnVo2WoeTvE93w+NAq2FXmvbiG7elt10XfQecvQU3QNbrvH
U8B96W0w8UXJdvTKg4f0NbjSw7iJ3x5naixQ+rA8hLV8x0gn2LX6wvxT/SEu
mn20KX+fPtJELK7v/NheFLX1jsKLXYo4jHrkfIXNsNUhg/x2E71kAjbET3s+
t9kCtxt2iXDDZvpIbmG04QkvLFvoroASmN6+8fupe3e+e2yN0e6xGTuE60gX
I2+X1p1g9IduDYTpO120X1eHyyMqGEEIb4g0iisloTp5oi3EuAYRGf1XuqAT
VA19bKnpkBsJ0A==
=tD2T
-----END PGP PUBLIC KEY BLOCK-----

```

- パブリックキーをキーリングにインポートし、返されたキー値をメモします。

GPG

```
gpg --import opshub-public-key.pgp
```

出力例

```
gpg: key 1655BBDE2B770256: public key "AWS OpsHub for Snow Family <aws-opshub-
signer@amazon.com>" imported
```

```
gpg: Total number processed: 1
gpg:             imported: 1
```

3. フィンガープリントを確認します。必ず、*key-value* を前のステップの値と置き換えてください。GPG を使用してフィンガープリントを検証することをお勧めします。

```
gpg --fingerprint key-value
```

このコマンドは、次のような出力を返します。

```
pub  rsa4096 2020-12-21 [SC]
      372F A5E9 4869 8F77 D1B3  AFAA 2181 CF5A 74F3 45F1
uid  [ unknown] AWS OpsHub for Snow Family <aws-opshub-signer@amazon.com>
sub  rsa4096 2020-12-21 [E]
```

フィンガープリントは、次のものと一致する必要があります。

```
372F A5E9 4869 8F77 D1B3 AFAA 2181 CF5A 74F3 45F1
```

フィンガープリントが一致しない場合は、AWS OpsHub アプリケーションをインストールしないでください。AWS Supportに連絡する。

4. インスタンスのアーキテクチャとオペレーティングシステムに従って署名ファイルをダウンロードします (まだ実行していない場合)。
5. インストーラパッケージの署名を確認します。必ず *signature-filename* と *OpsHub-download-filename* を、署名ファイルと AWS OpsHub アプリケーションのダウンロード時に指定した値と置き換えてください。

GPG

```
gpg --verify signature-filename OpsHub-download-filename
```

このコマンドは、次のような出力を返します。

GPG

```
gpg: Signature made Mon Dec 21 13:44:47 2020 PST
gpg:             using RSA key 1655BBDE2B770256
gpg: Good signature from "AWS OpsHub for Snow Family <aws-opshub-
signer@amazon.com>" [unknown]
```



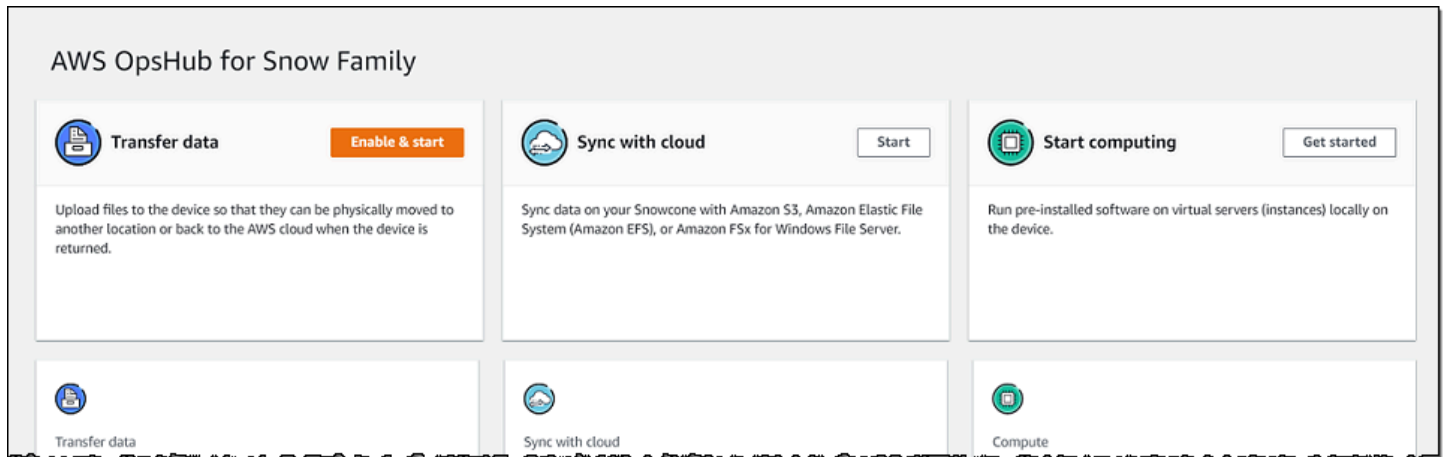
```
gpg: WARNING: This key is not certified with a trusted signature!  
gpg:          There is no indication that the signature belongs to the owner.  
Primary key fingerprint: 9C93 4C3B 61F8 C434 9F94 5CA0 1655 BBDE 2B77 0256
```

GPG を使用するとき、出力結果に「BAD signature」という句が含まれる場合、手順が正しいことをもう一度確認してください。このレスポンスが引き続き表示される場合は、に連絡し AWS Support、エージェントをインストールしないでください。信頼に関する警告メッセージは、署名が無効であることを意味するものではなく、公開キーを検証していないことを示しているだけです。キーは、自分や信頼する人が署名した場合にのみ信頼できます。

デバイスで AWS のサービスの管理

を使用すると AWS OpsHub、Snow Family デバイスで AWS サービスを使用および管理できます。現在、 は次のリソース AWS OpsHub をサポートしています。

- Amazon Elastic Compute Cloud (Amazon EC2) インスタンス – Amazon EC2 互換インスタンスを使用すると、仮想サーバーにインストールされたソフトウェアを AWS クラウド に送信せずに処理を実行することができます。
- ネットワークファイルシステム (NFS) — ファイル共有を使用してデータをデバイスに移動します。デバイスを AWS に配送して にデータを転送したり AWS クラウド、 DataSync を使用して他の AWS クラウド 場所に転送したりできます。
- Snow Family デバイスの Amazon S3 互換ストレージ – 耐障害性、スケールが向上し、Amazon S3 API 機能セットが堅牢でモバイルエッジ、切断された環境に拡張された安全なオブジェクトストレージを提供します。Snow Family デバイスで Amazon S3 互換ストレージを使用すると、データを格納し、エッジコンピューティング用に Snow Family デバイスで高可用性アプリケーションを実行できます。



トピック

- [Amazon EC2 互換コンピューティングインスタンスをローカルで使用する](#)
- [Amazon EC2 クラスターの管理](#)
- [Snow Family デバイスの Amazon S3 互換ストレージをセットアップする](#)
- [Amazon S3 アダプターストレージの管理](#)
- [NFS インターフェイスの管理](#)

Amazon EC2 互換コンピューティングインスタンスをローカルで使用する

を使用して AWS OpsHub、デバイスにローカルで仮想サーバー (インスタンス) にプリインストールされたソフトウェアを実行したり、デバイス上の Amazon EC2 インスタンスを管理したりできます。

トピック

- [Amazon EC2 互換インスタンスの起動](#)
- [Amazon EC2 互換インスタンスを停止する](#)
- [Amazon EC2 互換インスタンスを開始する](#)
- [キーペアによる作業](#)
- [Amazon EC2 互換インスタンスを終了する](#)
- [ストレージボリュームをローカルで使用する](#)
- [Amazon EC2 互換 AMI としてデバイスにイメージをインポートする](#)
- [スナップショットの削除](#)
- [AMI の登録解除](#)

Amazon EC2 互換インスタンスの起動

を使用して Amazon EC2-compatible インスタンスを起動するには、次の手順に従います AWS OpsHub。

Amazon EC2 互換インスタンスを起動するには

1. AWS OpsHub アプリケーションを開きます。
2. ダッシュボードの [Start computing] セクションで、[Get started] を選択します。または、上部の [Services] メニューを選択し、[Compute (EC2)] を選択して [Compute] ページを開きます。すべてのコンピューティングリソースが [Resources] セクションに表示されます。
3. デバイスで Amazon EC2 互換インスタンスを実行している場合、それらのインスタンスは、[インスタンス] の [インスタンス名] 列に表示されます。このページでは、各インスタンスの詳細を確認できます。
4. [インスタンスを起動] を選択します。インスタンス起動ウィザードが開きます。
5. [デバイス] で、Amazon EC2 互換インスタンスを起動する Snow デバイスを選択します。

Launch instance ✕

Device

192.0.2.0 ▼

Image (AMI)

snow-al2-test-ami-1.0.2 ▼

Instance type

sbe-c.small ▼

Create public IP address (VNI) Use existing IP address (VNI) Do not attach IP address

Physical network interface

SFP+:a.bc-1d2ef456gg678gi9j ▼

IP Address assignment

DHCP ▼

Key pair

Create key pair Use existing key pair Do not attach key pair

Name

test-instance-key-pair

The name can include up to 255 ASCII characters. It can't include leading or trailing spaces.

Create key pair

Cancel **Launch**

6. [Image (AMI)] で、リストから Amazon マシンイメージ (AMI) を選択します。この AMI は、インスタンスの起動に使用されます。
7. [インスタンスタイプ] で、リストからいずれかを選択します。
8. インスタンスに IP アドレスをアタッチする方法を選択します。次のオプションがあります。
 - [Create public IP address (VNI)] (パブリック IP アドレス (VNI) を作成する) — 物理ネットワークインターフェイスを使用して、新しい IP アドレスを作成するには、このオプションを選択します。物理ネットワークインターフェイスと IP アドレスの割り当てを選択します。
 - [Use existing IP address (VNI)] (既存の IP アドレス (VNI) を使用する) — 既存の IP アドレスを使用して、既存の仮想ネットワークインターフェイスを使用するには、このオプションを選択します。物理ネットワークインターフェイスと仮想ネットワークインターフェイスを選択します。
 - [Do not attach IP address] (IP アドレスをアタッチしない) — IP アドレスをアタッチしない場合は、このオプションを選択します。
9. インスタンスにキーペアをアタッチする方法を選択します。次のオプションがあります。

[Create key pair] (キーペアの作成) — 新しいキーペアを作成し、そのキーペアを使用して新しいインスタンスを起動するには、このオプションを選択します。

[Use existing key pair] (既存のキーペアを使用する) — 既存のキーペアを使用してインスタンスを起動するには、このオプションを選択します。

[Do not attach IP address] (IP アドレスをアタッチしない) — キーペアをアタッチしない場合は、このオプションを選択します。この AMI に組み込まれたパスワードが既知でない限り、このインスタンスに接続できないことを認識する必要があります。

詳細については、「[キーペアによる作業](#)」を参照してください。

10. [Launch] (起動する) を選択します。[Compute instances] セクションでインスタンスが起動されます。[State] は [Pending] で、完了すると [Running] に変わります。

Amazon EC2 互換インスタンスを停止する

を使用して Amazon EC2-compatible インスタンス AWS OpsHub を停止するには、次のステップに従います。

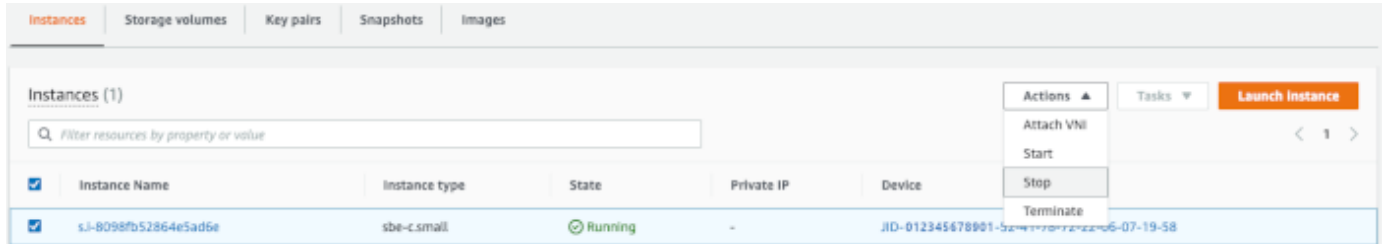
Amazon EC2 互換インスタンスを停止するには

1. AWS OpsHub アプリケーションを開きます。

2. ダッシュボードの [Start computing] セクションで、[Get started] を選択します。または、上部の [Services] メニューを選択し、[Compute (EC2)] を選択して [Compute] ページを開きます。

すべてのコンピューティングリソースが [Resources] セクションに表示されます。

3. デバイスで Amazon EC2 互換インスタンスを実行している場合、それらのインスタンスは、[インスタンス] の [インスタンス名] 列に表示されます。
4. 停止するインスタンスを選択し、[アクション]、[停止] の順に選択します。[State] が [Stopping] に変わり、完了すると [Stopped] に変わります。



Amazon EC2 互換インスタンスを開始する

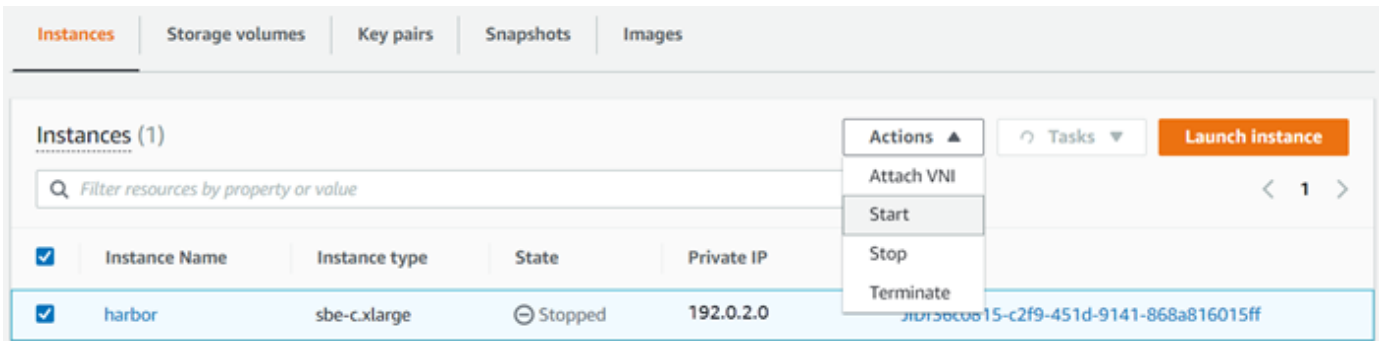
を使用して Amazon EC2-compatible インスタンスを起動するには、次の手順に従います AWS OpsHub。

Amazon EC2 互換インスタンスを開始するには

1. AWS OpsHub アプリケーションを開きます。
2. ダッシュボードの [Start computing] セクションで、[Get started] を選択します。または、上部の [Services] メニューを選択し、[Compute (EC2)] を選択して [Compute] ページを開きます。

コンピューティングリソースが [Resources] セクションに表示されます。

3. [インスタンス名] 列の [インスタンス] で、起動するインスタンスを見つけます。
4. インスタンスを選択し、[Start] を選択します。[State] が [Pending] に変わり、完了すると [Running] に変わります。



キーペアによる作業

Amazon EC2 互換インスタンスを起動し、SSH を使用してインスタンスに接続するには、キーペアを指定する必要があります。Amazon EC2 を使用して、新しいキーペアを作成したり、既存のキーペアをインポートしたり、キーペアを管理したりできます。

キーペアを作成、インポート、管理するには

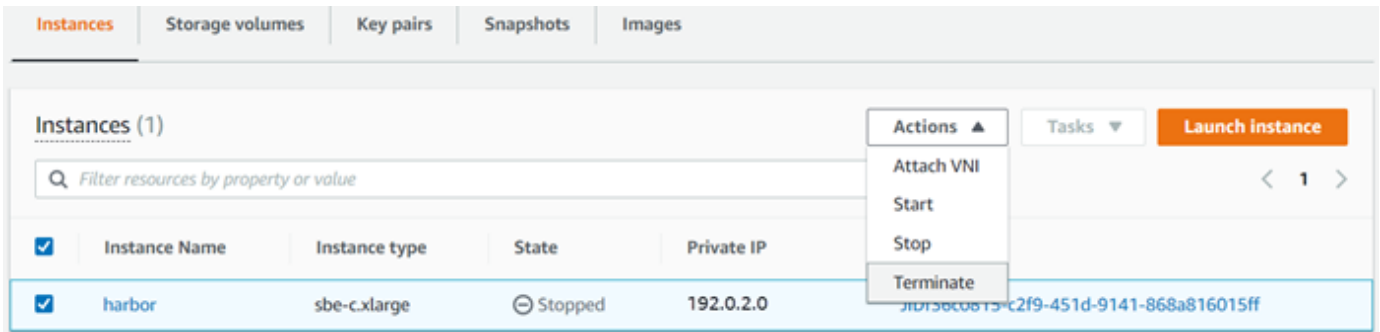
1. AWS OpsHub ダッシュボードでコンピューティングを開きます。
2. ナビゲーションペインで、[Compute (EC2)] ((EC2) のコンピューティング) ページ、[Key Pairs] (キーペア) タブの順に選択します。Amazon EC2 コンソールにリダイレクトされ、キーペアを作成、インポート、管理できます。
3. キーペアを作成してインポートする方法については、[Amazon EC2 ユーザーガイド](#)の「[Amazon EC2 キーペアと Linux インスタンス](#)」を参照してください。

Amazon EC2 互換インスタンスを終了する

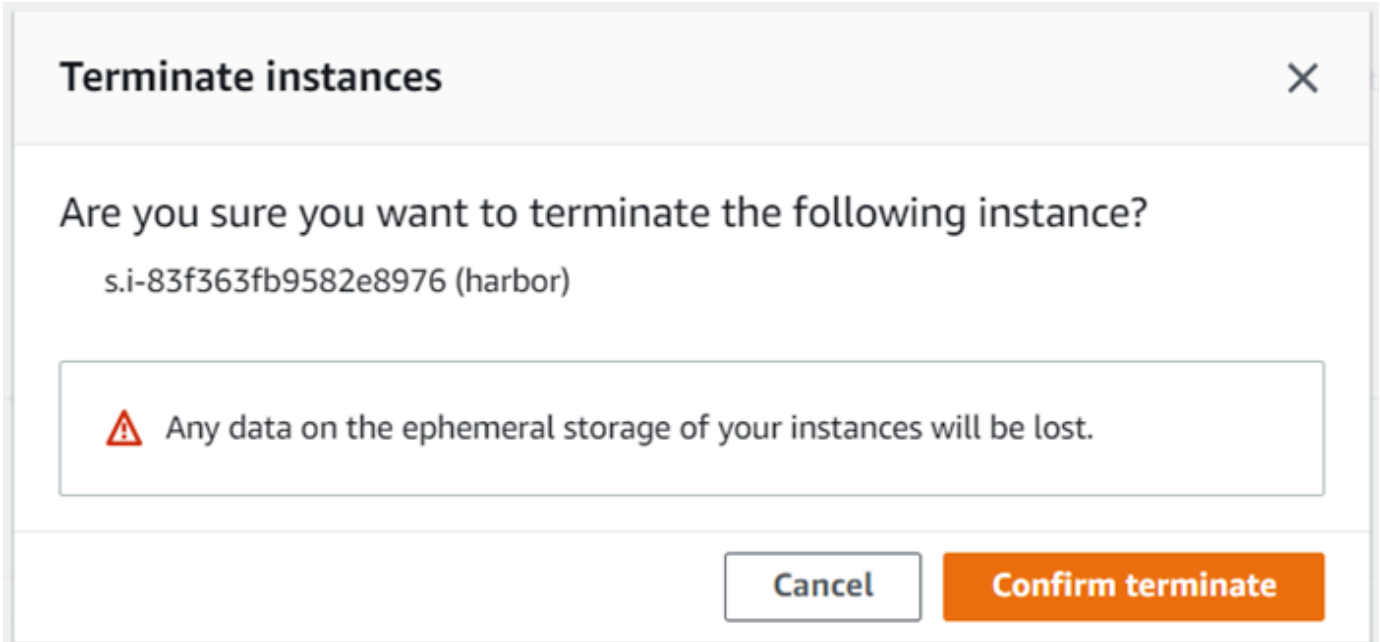
Amazon EC2 互換インスタンスを終了すると、再起動できなくなります。

Amazon EC2 互換インスタンスを終了するには

1. AWS OpsHub アプリケーションを開きます。
2. ダッシュボードの [Start computing] セクションで、[Get started] を選択します。または、上部の [Services] メニューを選択し、[Compute (EC2)] を選択して [Compute] ページを開きます。すべてのコンピューティングリソースは、[Resources] セクションに表示されます。
3. [インスタンス名] 列の [インスタンス] で、削除するインスタンスを見つけます。
4. インスタンスを選択し、[アクション] メニューを選択します。[アクション] メニューから [終了] を選択します。



5. [インスタンスを終了] ウィンドウで、[終了を確認] を選択します。



Note

インスタンスを削除すると、再起動できなくなります。

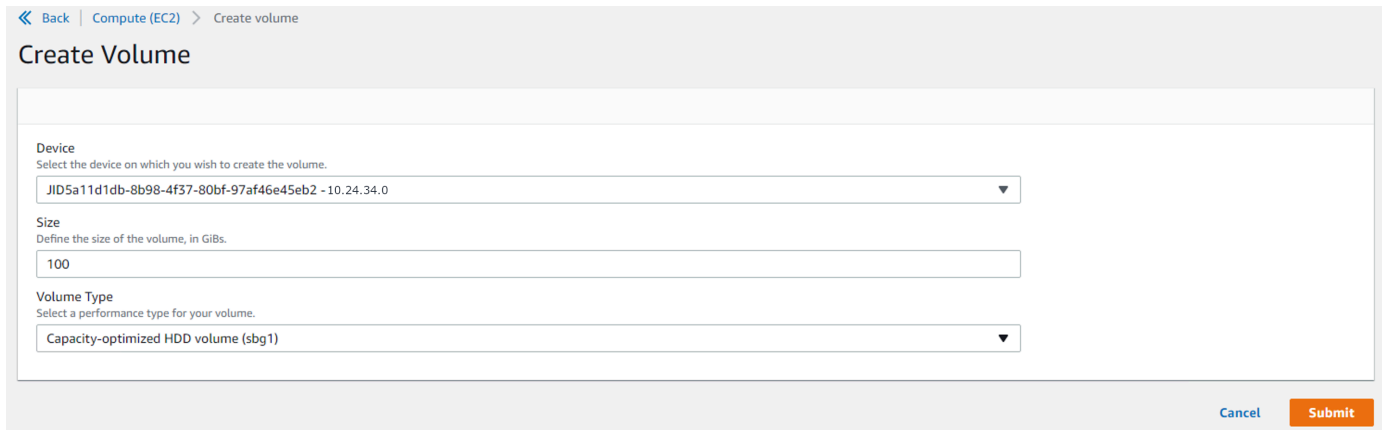
[State] が [Terminating] に変わり、完了すると [Terminated] に変わります。

ストレージボリュームをローカルで使用する

Amazon EC2 互換インスタンスは、ストレージに Amazon EBS ボリュームを使用します。この手順では、を使用してストレージボリュームを作成し、インスタンスにアタッチします AWS OpsHub。

ストレージボリュームを作成するには

1. AWS OpsHub アプリケーションを開きます。
2. ダッシュボードの [Start computing] セクションで、[Get started] を選択します。または、上部の [Services] メニューを選択し、[Compute (EC2)] を選択して [Compute] ページを開きます。
3. [ストレージボリューム] タブを選択します。デバイスにストレージボリュームがある場合は、ボリュームの詳細が [ストレージボリューム] の下に表示されます。
4. [Create volume] を選択して、[Create volume] ページを開きます。



Device
Select the device on which you wish to create the volume.
JID5a11d1db-8b98-4f37-80bf-97af46e45eb2 -10.24.34.0

Size
Define the size of the volume, in GiBs.
100

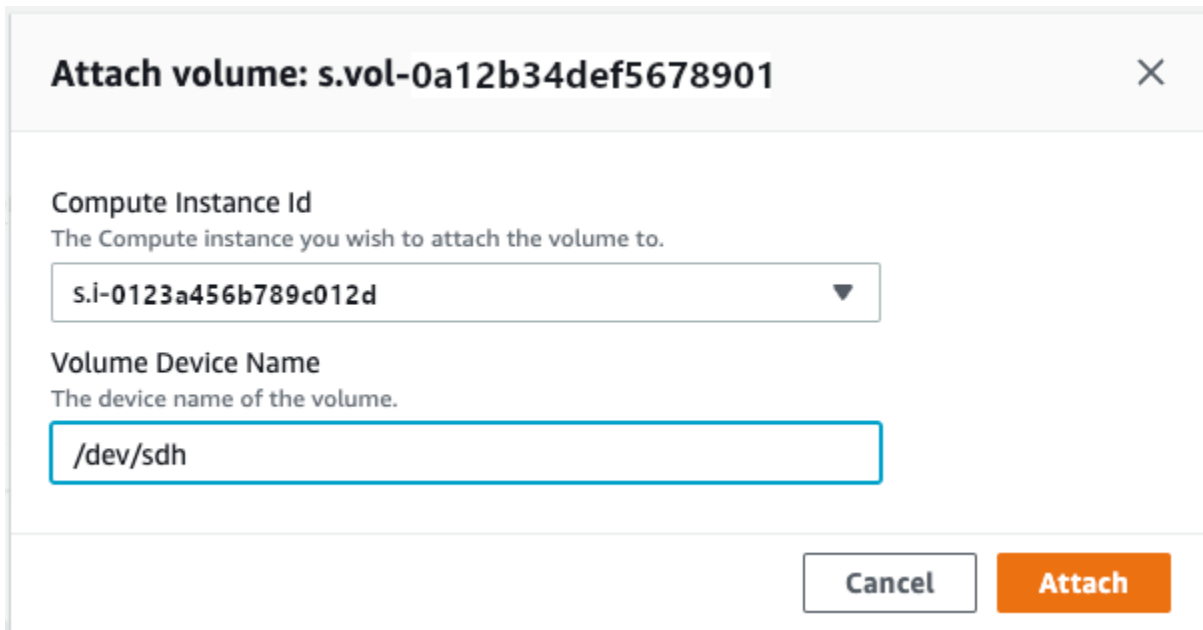
Volume Type
Select a performance type for your volume.
Capacity-optimized HDD volume (sbg1)

Cancel Submit

5. ボリュームを作成するデバイスを選択し、作成するサイズ (内 GiBs) を入力し、ボリュームのタイプを選択します。
6. [送信] を選択します。[State] は [Creating] で、完了すると [Available] に変わります。ボリュームとその詳細は、[Volumes] タブで確認できます。

インスタンスにストレージボリュームをアタッチするには

1. 作成したボリュームを選択し、[Attach volume] を選択します。



Attach volume: s.vol-0a12b34def5678901 ✕

Compute Instance Id
The Compute instance you wish to attach the volume to.

s.i-0123a456b789c012d ▼

Volume Device Name
The device name of the volume.

/dev/sdh

Cancel Attach

2. [コンピューティングインスタンス ID] で、ボリュームをアタッチする先のインスタンスを選択します。
3. [Volume Device Name] に、ボリュームのデバイス名 (`/dev/sdh`、`xvdh` など) を入力します。
4. 添付を選択します。

ボリュームが不要になった場合は、インスタンスからデタッチしてから削除できます。

Amazon EC2 互換 AMI としてデバイスにイメージをインポートする

イメージのスナップショットを Snowball Edge デバイスにインポートし、Amazon EC2 互換 Amazon マシンイメージ (AMI) として登録できます。スナップショットは基本的に、AMI または別のストレージボリュームの作成に使用できるストレージボリュームのコピーです。これにより、外部ソースからデバイスに独自のイメージを取り込み、Amazon EC2 互換インスタンスとして起動できます。

次のステップに従って、イメージのインポートを完了します。

1. デバイス上の Amazon S3 バケットにスナップショットをアップロードします。
2. Amazon S3、Amazon EC2、VM Import/Export、スナップショットを Import/Export するために使用する機能へのアクセス権を付与するために必要なアクセス許可を設定します。
3. S3 バケットから、スナップショットをイメージとしてデバイスにインポートします。
4. イメージを Amazon EC2 互換 AMI として登録します。
5. Amazon EC2 互換インスタンスとして AMI を起動します。

Note

Snow Family デバイスにスナップショットをアップロードする際は、次の制限事項に注意してください。

- Snow Family デバイスは現在、RAW イメージ形式のスナップショットのインポートのみをサポートしています。
- Snow Family デバイスは現在、1 GB から 1 TB までのサイズのスナップショットのインポートのみをサポートしています。

ステップ 1: デバイスの S3 バケットにスナップショットをアップロードする

スナップショットをインポートする前に、デバイス上の Amazon S3 にスナップショットをアップロードする必要があります。これは、スナップショットがデバイスまたはクラスターで利用可能な Amazon S3 からしか、インポートできないためです。インポートプロセス中にイメージを保存する、デバイス上の S3 バケットを選択します。

スナップショットを Amazon S3 にアップロードするには

- S3 バケットを作成するには、[Amazon S3 ストレージの作成](#)を参照してください。

S3 バケットにスナップショットをアップロードするには、[Amazon S3 ストレージへのファイルのアップロード](#)を参照してください。

ステップ 2: S3 バケットからスナップショットをインポートする

スナップショットが Amazon S3 にアップロードされると、そのスナップショットをデバイスにインポートできます。インポートされた、またはインポート中のすべてのスナップショットは、[Snapshots] (スナップショット) タブに表示されます。

スナップショットをデバイスにインポートするには

1. AWS OpsHub アプリケーションを開きます。
2. ダッシュボードの [Start computing] セクションで、[Get started] を選択します。または、上部の [Services] メニューを選択し、[Compute (EC2)] を選択して [Compute] ページを開きます。すべてのコンピューティングリソースが [Resources] セクションに表示されます。
3. [Snapshots] (スナップショット) タブを選択して、デバイスにインポートされたすべてのスナップショットを表示します。Amazon S3 のイメージファイルは、スナップショットとしてデバイ

スにインポートされる .raw ファイルです。スナップショット ID またはスナップショットの状態
でフィルタリングして、特定のスナップショットを検索できます。スナップショット ID を選択
すると、そのスナップショットの詳細を表示できます。

4. インポートするスナップショットを選択し、[Import snapshot] (スナップショットをインポート
する) を選択して、[Import snapshot] (スナップショットをインポートする) ページを開きます。
5. [Device] (デバイス) で、インポートする Snow Family デバイスの IP アドレスを選択します。
6. [Import description] (説明のインポート) および[Snapshot description] (スナップショットの説明)
に、それぞれの説明を入力します。
7. [Role] (ロール) リストで、インポートに使用するロールを選択します。Snow Family デバイスは
VM Import/Export を使用してスナップショットをインポートします。はこのロール AWS を引
き受け、それを使用してユーザーに代わってスナップショットをインポートします。にロールが
設定されていない場合は AWS Snowball Edge、(ローカル IAM ロールを作成 AWS OpsHub で
きるで IAM を開きます AWS Identity and Access Management。ロールには、インポートを実
行するために必要なVM Import/Export のアクセス許可を持つポリシーも必要です。また、ロー
ルにこのポリシーをアタッチする必要があります。詳細については、「[ローカルでの IAM の使
用](#)」を参照してください。

以下は ポリシーの例です。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "vmie.amazonaws.com"
      },
      "Action": "sts:AssumeRole"
    }
  ]
}
```

にサインイン AWS Management Console し、<https://console.aws.amazon.com/iam/> で IAM コ
ンソールを開きます。

作成するロールには、Amazon S3 にアクセスするための最小限のアクセス許可が必要です。以
下は最小ポリシーの例です。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "s3:GetBucketLocation",
        "s3:GetObject",
        "s3:ListBucket",
        "s3:GetMetadata"
      ],
      "Resource": [
        "arn:aws:s3:::import-snapshot-bucket-name",
        "arn:aws:s3:::import-snapshot-bucket-name/*"
      ]
    }
  ]
}
```

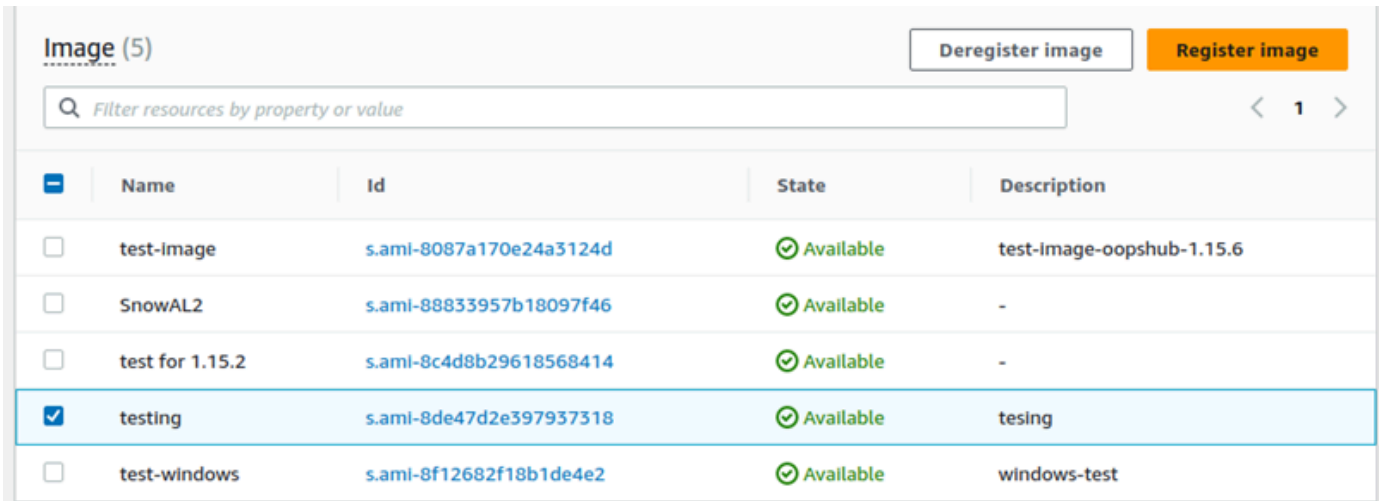
8. [Browse S3] (S3 を参照する) を選択し、インポートするスナップショットが含まれる S3 バケットを選択します。[スナップショット]、[Submit] (送信) の順に選択します。デバイスでスナップショットのダウンロードが始まります。スナップショット ID を選択して詳細を表示できます。このページからインポートプロセスをキャンセルできます。

ステップ 3: Amazon EC2 互換 AMI としてスナップショットを登録する

スナップショットとしてインポートされたイメージから Amazon EC2 互換 AMI を作成するプロセスを登録と呼びます。デバイスにインポートされたイメージは、Amazon EC2 互換インスタンスとして起動する前に登録する必要があります。

スナップショットとしてインポートされたイメージを登録するには

1. AWS OpsHub アプリケーションを開きます。
2. ダッシュボードの [Start computing] セクションで、[Get started] を選択します。または、上部の [Services] メニューを選択し、[Compute (EC2)] を選択して [Compute] ページを開きます。すべてのコンピューティングリソースが [Resources] セクションに表示されます。
3. [Images] (イメージ) タブを選択します。名前、ID、状態でイメージをフィルタリングして、特定のイメージを検索することができます。
4. 登録するイメージを選択し、[Register image] (イメージの登録) を選択します。



	Name	Id	State	Description
<input type="checkbox"/>	test-image	s.ami-8087a170e24a3124d	Available	test-image-oopshub-1.15.6
<input type="checkbox"/>	SnowAL2	s.ami-88833957b18097f46	Available	-
<input type="checkbox"/>	test for 1.15.2	s.ami-8c4d8b29618568414	Available	-
<input checked="" type="checkbox"/>	testing	s.ami-8de47d2e397937318	Available	tesing
<input type="checkbox"/>	test-windows	s.ami-8f12682f18b1de4e2	Available	windows-test

- [Register image] (イメージの登録) ページで、[Name] (名前) および [Description] (説明) を入力します。
- [Root volume] (ルートボリューム) で、ルートデバイスの名前を指定します。

[Block device] (ブロックデバイス) セクションで、ボリュームサイズとボリュームタイプを変更できます。
- インスタンス終了時にボリュームを削除する場合は、[Delete on termination] (終了時に削除) を選択します。
- さらにボリュームを追加する場合は、[Add new volume] (新しいボリュームの追加) を選択します。
- 終了したら、[Submit] (送信) を選択します。

ステップ 4: Amazon EC2 互換 AMI を起動する

- 詳細については、「[Amazon EC2 互換インスタンスを起動する](#)」を参照してください。

スナップショットの削除

スナップショットが不要になった場合は、デバイスから削除できます。Amazon S3 のイメージファイルは、スナップショットとしてデバイスにインポートされる .raw ファイルです。削除するスナップショットがイメージで使用されている場合、そのスナップショットは削除できません。インポートが完了したら、デバイス上の Amazon S3 にアップロードした .raw ファイルも削除することができます。

スナップショットを削除するには

1. AWS OpsHub アプリケーションを開きます。
2. ダッシュボードの [Start computing] セクションで、[Get started] を選択します。または、上部の [Services] メニューを選択し、[Compute (EC2)] を選択して [Compute] ページを開きます。すべてのコンピューティングリソースが [Resources] セクションに表示されます。
3. [Snapshot] (スナップショット) タブを選択して、インポートされたすべてのスナップショットを表示します。スナップショット ID またはスナップショットの状態でフィルタリングして、特定のスナップショットを検索できます。
4. 削除するスナップショットを選択して、[Delete] (削除) を選択します。複数のスナップショットを選択できます。

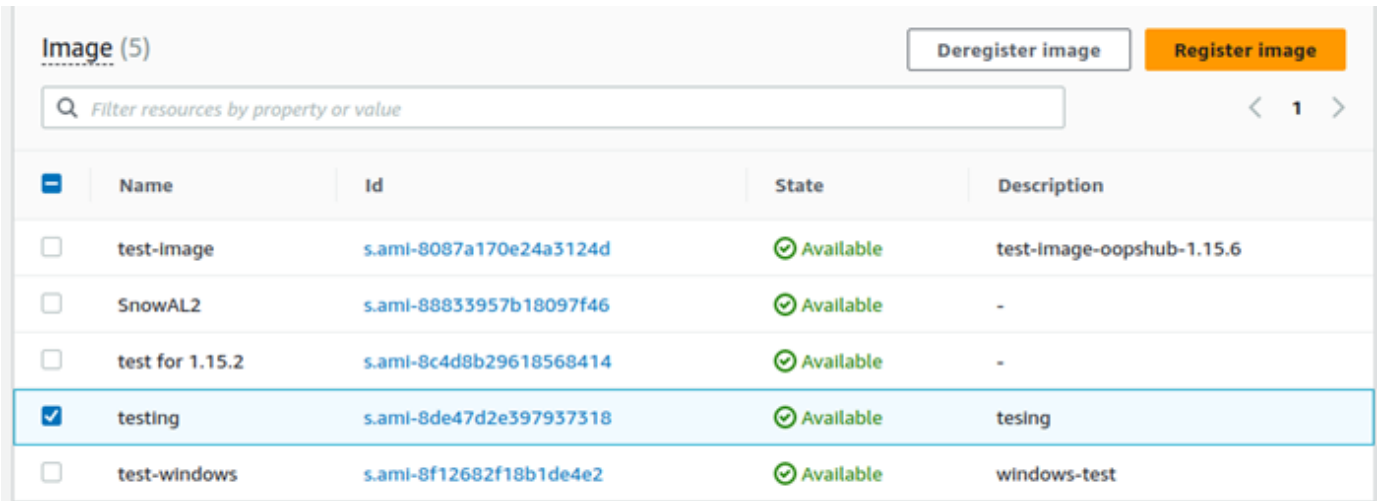
Snapshot ID	State	Capacity	Description	Start Time
s-snap-01a2b3c4567d8e9f0g	Completed	8 GBs	Snow marketplace AL2 AMI 1.0.2	Thu, 09 Jun 2022 16:27:29 GMT

5. [Delete snapshot confirmation] (スナップショット削除の確認) ボックスで、[Delete snapshot] (スナップショットの削除) を選択します。削除が成功すると、スナップショットは [Snapshots] (スナップショット) タブのリストから削除されます。

AMI の登録解除

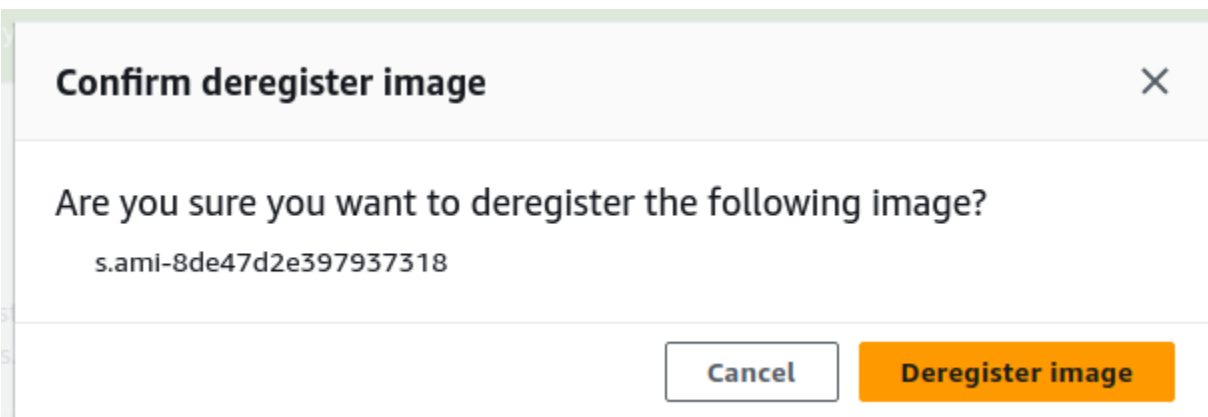
AMI の登録を解除するには

1. AWS OpsHub アプリケーションを開きます。
2. ダッシュボードの [Start computing] セクションで、[Get started] を選択します。または、上部の [Services] メニューを選択し、[Compute (EC2)] を選択して [Compute] ページを開きます。すべてのコンピューティングリソースが [Resources] セクションに表示されます。
3. [Images] (イメージ) タブを選択します。すべてのイメージが一覧表示されます。名前、ID、状態でイメージをフィルタリングして、特定のイメージを検索することができます。
4. 登録解除するイメージを選択し、[Deregister] (登録解除) を選択します。



	Name	Id	State	Description
<input type="checkbox"/>	test-image	s.ami-8087a170e24a3124d	Available	test-image-oopshub-1.15.6
<input type="checkbox"/>	SnowAL2	s.ami-88833957b18097f46	Available	-
<input type="checkbox"/>	test for 1.15.2	s.ami-8c4d8b29618568414	Available	-
<input checked="" type="checkbox"/>	testing	s.ami-8de47d2e397937318	Available	tesing
<input type="checkbox"/>	test-windows	s.ami-8f12682f18b1de4e2	Available	windows-test

5. [イメージの登録解除を確認] ウィンドウで、イメージ ID を確認し、[イメージの登録解除] を選択します。登録解除が成功すると、イメージはイメージリストから削除されます。



Amazon EC2 クラスターの管理

Amazon EC2 クラスターは、デバイスのクラスターとして一緒にプロビジョニングするデバイスのグループです。クラスターを使用するには、デバイスの AWS サービスがデフォルトのエンドポイントで実行されている必要があります。また、通信するクラスター内の特定のデバイスを選択する必要があります。クラスターはデバイスごとに使用します。

Amazon EC2 クラスターを作成するには

1. Snow デバイスに接続してログインします。デバイスにログインする方法については、「[デバイスのロック解除](#)」を参照してください。
2. [Choose device] (デバイスの選択) ページで、[Snowball Edge cluster] (Snowball Edge クラスター)、[Next] (次へ) の順に選択します。

3. [デバイスへの接続] ページで、デバイスの IP アドレスと、クラスター内の他のデバイスの IP アドレスを入力します。
4. [Add another device] を選択してデバイスを追加し、[次へ] を選択します。
5. [Provide the keys] ページで、デバイスクライアントのロック解除コードを入力し、デバイスマニフェストをアップロードして、[Unlock device] を選択します。

Snowball Edge デバイスは 256 ビット暗号化を使用して、データのセキュリティとフルの両方を確保 chain-of-custody します。

6. (オプション) 名前を入力してプロファイルを作成してから、[Save profile name] (プロファイル名の保存) を選択します。ダッシュボードが表示されます。ここには、すべてのクラスターが表示されます。

AWS サービスの使用とクラスターの管理を開始できるようになりました。クラスター内のインスタンスは、個々のインスタンスを管理するのと同じ方法で管理します。手順については、「[デバイスで AWS のサービスの管理](#)」または「[デバイスの管理](#)」を参照してください。

Snow Family デバイスの Amazon S3 互換ストレージをセットアップする

Snow Family デバイスの Amazon S3 互換ストレージサービスは、デフォルトではアクティブになっていません。デバイスまたはクラスターでサービスを開始するには、各デバイスに 2 つの仮想ネットワークインターフェイス (vNIC) を作成して、s3control エンドポイントと s3api エンドポイントにアタッチする必要があります。

トピック

- [前提条件](#)
- [簡易セットアップオプションを使用する](#)
- [詳細セットアップオプションを使用する](#)
- [Snow Family デバイスの Amazon S3 互換ストレージサービスを自動起動するように設定するサービス](#)
- [Snow Family デバイスの Amazon S3 互換ストレージにバケットを作成する](#)
- [Snow Family デバイスバケットの Amazon S3 互換ストレージにファイルとフォルダをアップロードする](#)
- [Snow Family デバイスバケットの Amazon S3 互換ストレージからファイルとフォルダを削除する](#)
- [Snow Family デバイスの Amazon S3 互換ストレージからバケットを削除する](#)

前提条件

を使用してデバイスまたはクラスターをセットアップする前に AWS OpsHub for Snow Family、次の操作を行います。

- Snowball Edge デバイスの電源を入れ、ネットワークに接続します。
- ローカルマシンで、最新バージョンの [AWS OpsHub](#) をダウンロードしてインストールします。デバイスまたはクラスターを接続し、マニフェストファイルを使用してロックを解除します。デバイスのロック解除に関する詳細については、「[デバイスのロック解除](#)」を参照してください。

簡易セットアップオプションを使用する

ネットワークで DHCP を使用している場合は、簡易セットアップオプションを使用してください。このオプションでは、サービスの開始時に VNIC が各デバイスに自動的に作成されます。

1. にログインし AWS OpsHub、ストレージの管理 を選択します。

Snow Family デバイスの Amazon S3 互換ストレージのランディングページに移動します。

2. [サービスのセットアップ開始タイプ] で [簡易] を選択します。
3. [Start service] (サービスの開始) を選択します。

Note

完了に数分かかり、使用しているデバイスの数によって異なります。

サービスが開始されると、サービスの状態はアクティブになり、エンドポイントが表示されま

す。

Amazon S3 compatible storage on Snow Stop service Start service setup

Use Amazon S3 compatible storage on Snow to manage files and folders on your device(s). Add files to the device so they can be accessed locally.

Amazon S3 compatible storage on Snow resources Refresh Enable service auto-start

Amazon S3 compatible storage on Snow uses one GB of RAM and one of your device CPUs, limiting the amount of compute instances available.

Service state Info Active	Service auto-start Info Disabled	S3 storage available -
S3 endpoint status Active	S3 endpoint Info 10.0.0.8	S3Control endpoint status Active
		S3Control endpoint Info 10.0.0.1

Buckets (8) [Info](#) Refresh Empty Delete Create bucket

Buckets are containers for data stored in Amazon S3 compatible storage on Snow.

Find buckets by name

Name	Creation date
1bucket	Thu, 16 Mar 2023 00:51:53 GMT

詳細セットアップオプションを使用する

ネットワークで静的 IP アドレスを使用している場合や、既存の VNI を再利用したい場合は、詳細セットアップオプションを使用してください。このオプションでは、デバイスごとに手動で VNIC を作成します。

1. にログインし AWS OpsHub、ストレージの管理 を選択します。

Snow Family デバイスの Amazon S3 互換ストレージのランディングページに移動します。

2. [サービスのセットアップ開始タイプ] で [詳細] を選択します。
3. VNIC を作成する必要があるデバイスを選択します。

クラスターの場合、Snow Family デバイスの Amazon S3 互換ストレージサービスを開始するには、最小数のデバイス数のクォーラムが必要です。3 ノードクラスターのクォーラムは 2 つです。

Note

クラスター設定でサービスを初めて起動する際は、クラスター内のすべてのデバイスを設定し、サービスで使用を開始できるようにしておく必要があります。次回以降の起動では、クォーラムを満たしていればデバイスのサブセットを使用できますが、サービスはデグレードした状態で開始されます。

4. デバイスごとに、既存の VNIC を選択するか、[VNI の作成] を選択します。

各デバイスには、オブジェクト操作用の S3 エンドポイント用と、バケット操作用の S3Control エンドポイント用の VNIC が必要です。

5. VNIC を作成する場合は、物理ネットワークインターフェイスを選択し、ステータス IP アドレスとサブネットマスクを入力してから、[仮想ネットワークインターフェイスを作成] を選択します。

6. VNIC を作成したら、[サービスを開始] を選択します。

Note

完了に数分かかり、使用しているデバイスの数によって異なります。

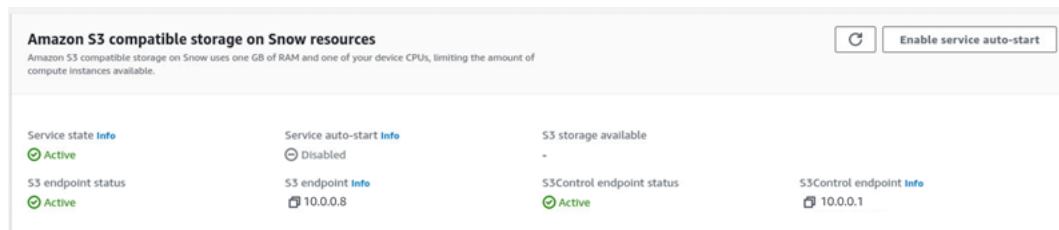
サービスが開始されると、サービスの状態はアクティブになり、エンドポイントが表示されます。

Snow Family デバイスの Amazon S3 互換ストレージサービスを自動起動するように設定するサービス

1. にログインし AWS OpsHub、ストレージの管理 を選択します。

Snow Family デバイスの Amazon S3 互換ストレージのランディングページに移動します。

2. [Snow リソースの Amazon S3 互換ストレージ] で、[サービスの自動起動を有効にする] を選択します。システムが、今後サービスが自動起動するように設定します。



Snow Family デバイスの Amazon S3 互換ストレージにバケットを作成する

AWS OpsHub インターフェイスを使用して、Snow Family デバイスに Amazon S3 バケットを作成します。

1. を開きます AWS OpsHub。
2. [ストレージの管理] で [開始] を選択します。[Snow の Amazon S3 互換ストレージ] ページが表示されます。
3. [Buckets] (バケット) で、[Create bucket] (バケットの作成) を選択します。[バケットの作成] 画面が表示されます。

Create bucket

Bucket settings

Bucket name [Info](#)

test123

Bucket names must be unique within your Snowball device or cluster and must not contain spaces or uppercase letters.

Default encryption

Automatically encrypt new objects uploaded to this snow bucket. [Learn more](#)

S3 compatible storage on Snow buckets are encrypted at all times and this setting cannot be changed.

Default encryption
Enabled

Encryption type
Amazon S3 key (SSE-S3)

Cancel **Create bucket**

4. [バケット名] にバケットの名前を入力します。

Note

バケット名は Snowball デバイスまたはクラスター内で一意である必要があり、スペースや大文字を含めることはできません。

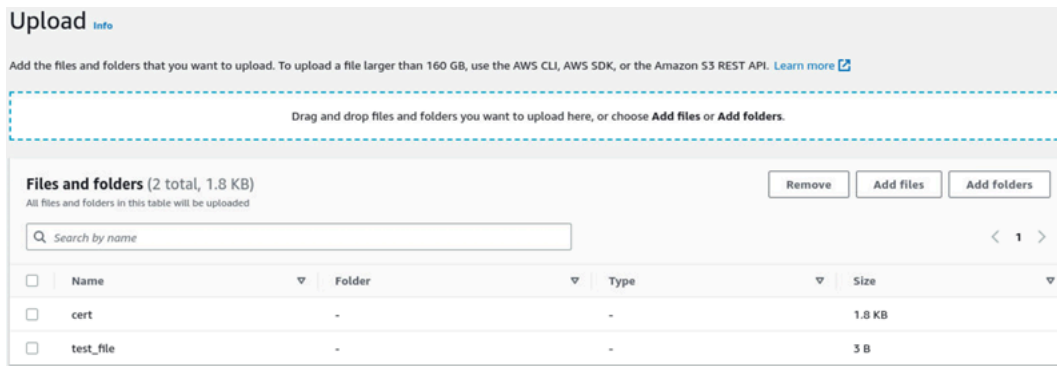
5. [バケットを作成] を選択します。システムによりバケットが作成され、[Snow の Amazon S3 互換ストレージ] ページの [バケット] に表示されます。

Snow Family デバイスバケットの Amazon S3 互換ストレージにファイルとフォルダをアップロードする

AWS OpsHub インターフェイスを使用して、ファイルとフォルダを Snow Family デバイスバケットの Amazon S3 互換ストレージにアップロードします。ファイルとフォルダは個別にアップロードすることも、まとめてアップロードすることもできます。

1. AWS OpsHubを開きます。
2. [ストレージの管理] の [バケット] で、ファイルをアップロードするバケットを選択します。そのバケットのページが表示されます。

3. バケットのページで [ファイルをアップロード] を選択します。[アップロード] ページが表示されます。



4. オペレーティングシステムのファイルマネージャーから AWS OpsHub ウィンドウにドラッグしてファイルまたはフォルダをアップロードするか、次の手順を実行します。
 - a. [ファイルを追加] または [フォルダを追加] を選択します。
 - b. アップロードするファイルまたはフォルダを 1 つ以上選択します。[Open] (開く) を選択します。

システムが、選択したファイルとフォルダをデバイスのバケットにアップロードします。アップロードが完了すると、ファイルとフォルダの名前が [ファイルとフォルダ] リストに表示されます。

Snow Family デバイスバケットの Amazon S3 互換ストレージからファイルとフォルダを削除する

AWS OpsHub インターフェイスを使用して、Snow Family デバイスのバケットからファイルとフォルダを削除し、完全に削除します。

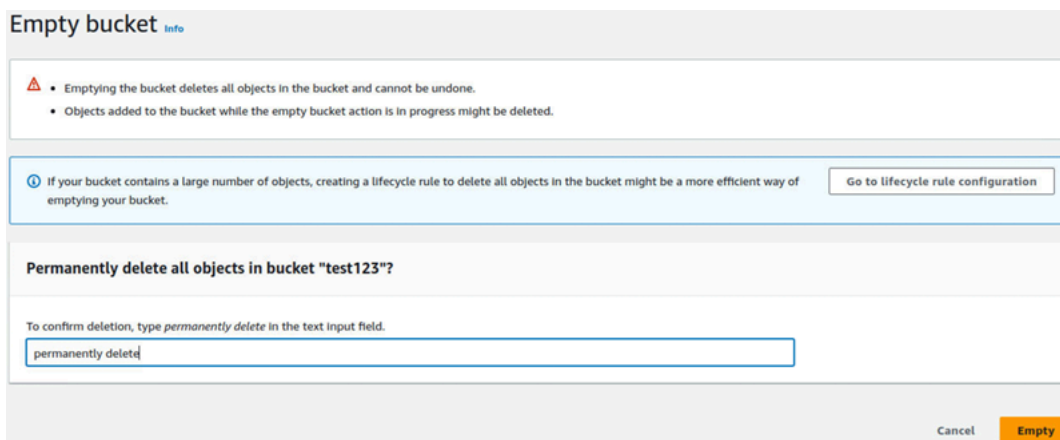
1. を開きます AWS OpsHub。
2. [ストレージの管理] の [バケット] で、ファイルとフォルダを削除するバケットの名前を選択します。そのバケットのページが表示されます。
3. [ファイルとフォルダ] で、完全に削除するファイルとフォルダのチェックボックスをオンにします。
4. [削除] を選択します。システムが、デバイス上のバケットからファイルまたはフォルダを削除します。

Snow Family デバイスの Amazon S3 互換ストレージからバケットを削除する

デバイスからバケットを削除するには、バケットが空である必要があります。バケットからファイルやフォルダを削除するか、バケットを空にするツールを使用してください。ファイルやフォルダを削除するには、「[Snow Family デバイスバケットの Amazon S3 互換ストレージからファイルとフォルダを削除する](#)」を参照してください。

バケットを空にするツールを使用するには

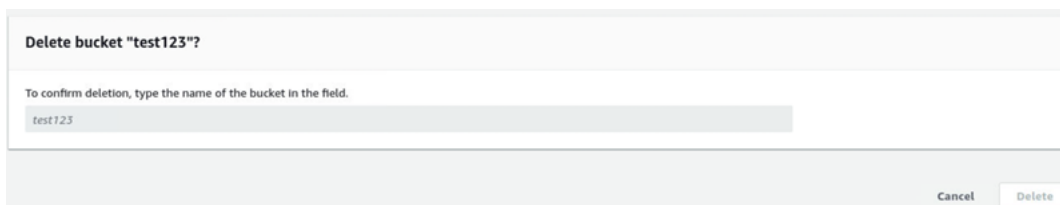
1. を開きます AWS OpsHub。
2. [ストレージの管理] の [バケット] で、空にするバケットのラジオボタンを選択します。
3. [空にする] を選択します。[バケットを空にする] ページが表示されます。



4. [バケットを空にする] ページのテキストボックスに、**permanently delete** と入力します。
5. [空にする] を選択します。システムによってバケットが空にされます。

空のバケットを削除するには

1. [ストレージの管理] の [バケット] で、削除するバケットのラジオボタンを選択します。
2. [削除] を選択します。[バケットの削除] ページが表示されます。



3. [バケットの削除] ページのテキストボックスに、バケットの名前を入力します。
4. [削除] を選択します。システムによってデバイスからバケットが削除されます。

Amazon S3 アダプターストレージの管理

を使用して AWS OpsHub、インポートジョブとエクスポートジョブに Amazon S3 (S3) ストレージを作成および管理できます。

トピック

- [Amazon S3 ストレージへのアクセス](#)
- [Amazon S3 ストレージへのファイルのアップロード](#)
- [Amazon S3 ストレージからのファイルのダウンロード](#)
- [Amazon S3 ストレージからのファイルの削除](#)

Amazon S3 ストレージへのアクセス

デバイスにファイルをアップロードし、ローカルでファイルにアクセスできます。デバイスの別の場所に物理的に移動したり、デバイスが返却された AWS クラウド ときに にインポートし直すことができます。

Snow Family デバイスは、Amazon S3 バケットを使用して、デバイス上のファイルを保存および管理します。

S3 バケットにアクセスするには

1. AWS OpsHub アプリケーションを開きます。
2. ダッシュボードの [Manage file storage] セクションで、[Get started] を選択します。

デバイスが Amazon S3 転送メカニズムで注文された場合、そのデバイスは [ファイルとオブジェクトストレージ] ページの [バケット] セクションに表示されます。[ファイルとオブジェクトストレージ] ページでは、各バケットの詳細を確認できます。

Note

デバイスが NFS 転送メカニズムで注文された場合、NFS サービスを設定してアクティブ化した後に、バケット名がマウントポイントセクションに表示されます。ファイルインターフェイスの詳細については、「[NFS インターフェイスの管理](#)」を参照してください。

File & object storage

Use Amazon S3 to manage files and objects stored on your device. Add files to the device so they can be accessed locally. For import jobs, the files will be transferred to AWS when the device is sent back.

Resources

Storage available

925.85 GB available of 925.93 GB

99%

Select a bucket below to start transferring files to your device.

Buckets (7)

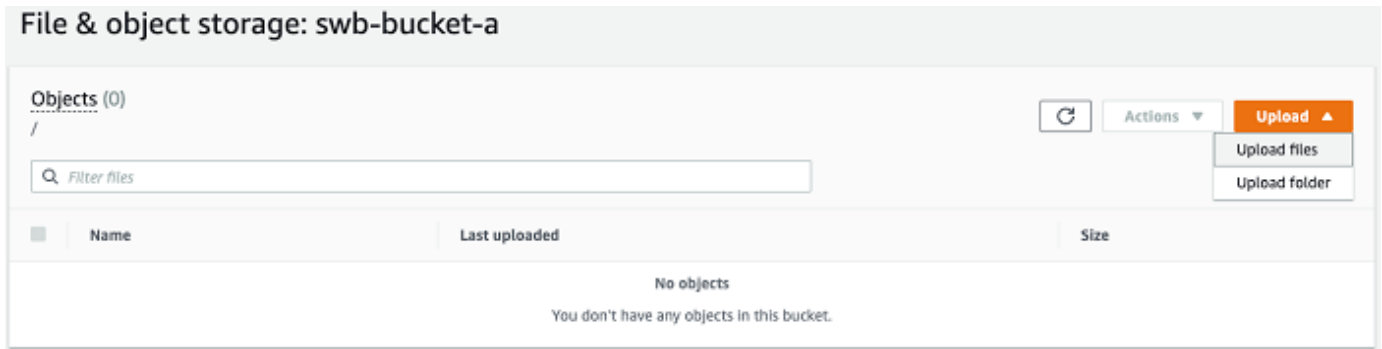
Filter buckets

Bucket name	Date created
sbw-output	Mon, 12 Oct 2009 17:50:30 GMT
swb-bucket-a	Mon, 12 Oct 2009 17:50:30 GMT
swb-bucket-b	Mon, 12 Oct 2009 17:50:30 GMT
swb-bucket-c	Mon, 12 Oct 2009 17:50:30 GMT
swb-bucket-d	Mon, 12 Oct 2009 17:50:30 GMT
swb-bucket-e	Mon, 12 Oct 2009 17:50:30 GMT
swb-bucket-f	Mon, 12 Oct 2009 17:50:30 GMT

Amazon S3 ストレージへのファイルのアップロード

ファイルをアップロードするには

1. ダッシュボードの [Manage file storage] セクションで、[Get started] を選択します。デバイスに Amazon S3 バケットがある場合、それらは [File storage] (ファイルストレージ) ページの [Buckets] (バケット) セクションに表示されます。ページでは、各バケットの詳細を確認できます。
2. ファイルをアップロードするバケットを選択します。
3. [アップロード]、[ファイルをアップロード] の順に選択するか、バケットにファイルをドラッグアンドドロップして、[OK] を選択します。



Note

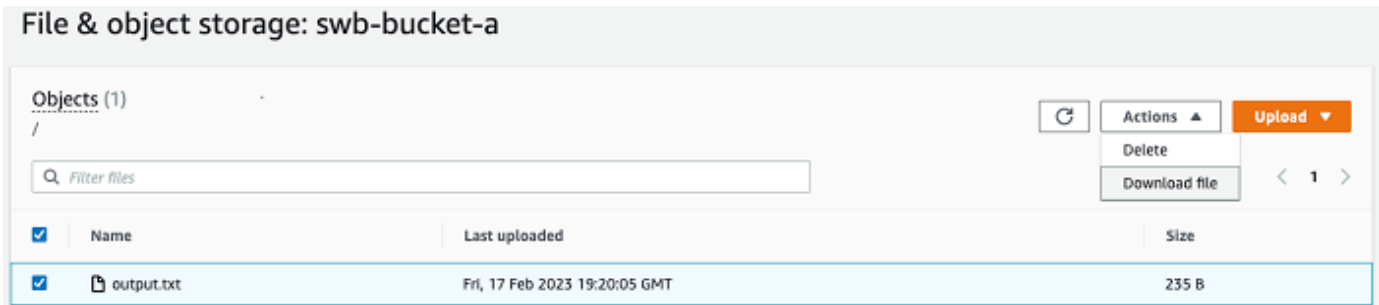
サイズの大きいファイルをアップロードするには、AWS CLIを使用して Amazon S3 のマルチパートアップロード機能を使用できます。S3 CLI 設定の詳細については、[「CLI S3 設定」](#)を参照してください。マルチパートアップロードの詳細については、Amazon Simple Storage Service ユーザーガイドの[「マルチパートアップロードの概要」](#)を参照してください。

を使用したローカルマシンから Snowball Edge へのフォルダのアップロード AWS OpsHub がサポートされています。フォルダサイズが非常に大きい場合、ファイル/フォルダの選択を読み取る OpsHub までに少し時間がかかります。OpsHub がファイルとフォルダを読み取っている間は、進行状況トラッカーは表示されません。ただし、アップロードプロセスが開始されると進行状況トラッカーが表示されます。

Amazon S3 ストレージからのファイルのダウンロード

ファイルをダウンロードするには

1. ダッシュボードの [Manage file storage] セクションで、[Get started] を選択します。デバイスに S3 バケットがある場合、それらは [File storage] (ファイルストレージ) ページの [Buckets] (バケット) セクションに表示されます。ページでは、各バケットの詳細を確認できます。
2. ファイルをダウンロードするバケットを選択し、ダウンロードするファイルに移動します。1 つ以上のファイルを選択します。



3. [Actions] メニューで、[Download] を選択します。
4. ファイルをダウンロードする場所を選択し、[OK] を選択します。

Amazon S3 ストレージからのファイルの削除

不要になったファイルは、Amazon S3 バケットから削除できます。

ファイルを削除するには

1. ダッシュボードの [Manage file storage] セクションで、[Get started] を選択します。デバイスに Amazon S3 バケットがある場合、それらは [File storage] (ファイルストレージ) ページの [Buckets] (バケット) セクションに表示されます。ページでは、各バケットの詳細を確認できます。
2. ファイルを削除するバケットを選択し、削除するファイルに移動します。
3. Actions メニューで、Delete を選択します。
4. 表示されるダイアログボックスで、[Confirm delete] を選択します。

NFS インターフェイスの管理

ネットワークファイルシステム (NFS) インターフェイスを使用して、デバイスがオペレーティングシステムにローカルストレージであるかのように、Snow Family デバイスにファイルをアップロードします。これにより、ファイルのコピー、ドラッグアンドドロップ、その他のグラフィカルユーザーインターフェイス機能など、オペレーティングシステムの機能を使用できるため、データ転送に対するよりユーザーフレンドリーなアプローチが可能になります。デバイス上の各 S3 バケットは NFS インターフェイスエンドポイントとして利用でき、データをコピーするためにマウントできます。NFS インターフェイスはインポートジョブで使用できます。

Snowball Edge デバイスが、デバイスを注文するジョブの作成時にそれを含めるように設定されている場合は、NFS インターフェイスを使用できます。デバイスが NFS インターフェイスを含める

ように設定されていない場合は、S3 Family デバイスの S3 アダプターまたは Amazon S3 互換ストレージを使用してデータを転送します。S3 アダプターの詳細については、「」を参照してください[Amazon S3 アダプターストレージの管理](#)。Snow Family デバイスの Amazon S3 互換ストレージの詳細については、「」を参照してください[Snow Family デバイスの Amazon S3 互換ストレージをセットアップする](#)。

起動すると、NFS インターフェイスは 1 GB のメモリと 1 CPU を使用します。これにより、Snow Family デバイスで実行されている他のサービスの数や、が実行できる EC2-compatible インスタンスの数が制限される場合があります。

NFS インターフェイスを介して転送されるデータは、転送中に暗号化されません。NFS インターフェイスを設定する場合、CIDR ブロックを提供できます。Snow Family デバイスは、それらのブロック内のアドレスを持つクライアントコンピュータからの NFS インターフェイスへのアクセスを制限します。

デバイス上のファイルは、に返されると Amazon S3 に転送されます AWS。詳細については、[Amazon S3 へのジョブのインポート](#)」を参照してください。

コンピュータオペレーティングシステムで NFS を使用方法の詳細については、オペレーティングシステムのドキュメントを参照してください。

NFS インターフェイスを使用する場合は、次の詳細に注意してください。

- ファイル名は、Snow Family デバイス上のローカル S3 バケット内のオブジェクトキーです。キー名は一連の Unicode 文字で、UTF-8 にエンコードすると最大で 1,024 バイト長になります。可能な場合は NFSv4.1 を使用し、Unicode UTF-8 でファイル名をエンコードして、データを正常にインポートすることをお勧めします。UTF-8 でエンコードされていないファイル名は S3 にアップロードされないか、使用する NFS エンコーディングに応じて、異なるファイル名で S3 にアップロードされる場合があります。
- ファイルパスの最大長は 1,024 文字であることに注意してください。Snow Family デバイスでは、1,024 文字を超えるファイルパスはサポートされません。このファイルパスの長さを超えると、ファイルのインポートエラーが発生します。
- 詳細については、「Amazon Simple Storage Service ユーザーガイド」の「[オブジェクトキー](#)」を参照してください。
- NFS ベースの転送の場合、Snow Family デバイスから Amazon S3 にインポートされると、標準の POSIX スタイルのメタデータがオブジェクトに追加されます。さらに、NFS オプションを使用した Snow Family デバイスの Amazon S3 への内部インポートメカニズムの一部として現在使用している AWS DataSync メタデータ `x-amz-meta-user-agent-aws-datasync` が表示されます。

- 単一の Snowball Edge デバイスを使用して、最大 40M のファイルを転送できます。1 つのジョブで 4,000 万を超えるファイルを転送する必要がある場合は、転送ごとのファイル数を減らすためにファイルをバッチ処理してください。拡張 NFS インターフェイスまたは S3 インターフェイスを搭載した Snowball Edge デバイスの場合、個々のファイルのサイズは任意で、最大ファイルサイズは 5 TB です。

コマンドラインインターフェイス (CLI) ツールである Snowball Edge クライアントを使用して NFS インターフェイスを設定および管理することもできます。詳細については、[「NFS インターフェイスの管理」](#)を参照してください。

トピック

- [Windows オペレーティングシステムでの NFS サービスの起動](#)
- [NFS インターフェイスの自動設定](#)
- [NFS インターフェイスを手動で設定する](#)
- [Snow Family デバイスの NFS エンドポイントの管理](#)
- [クライアントコンピュータへの NFS エンドポイントのマウント](#)
- [NFS インターフェイスの停止](#)

Windows オペレーティングシステムでの NFS サービスの起動

クライアントコンピュータが Windows 10 Enterprise または Windows 7 Enterprise オペレーティングシステムを使用している場合は、AWS OpsHub アプリケーションで NFS を設定する前に、クライアントコンピュータで NFS サービスを起動します。

1. クライアントコンピュータで、[スタート] ボタンをクリックし、[コントロールパネル]、[プログラム] の順に選択します。
2. [Turn Windows features on or off] を選択します。

Note

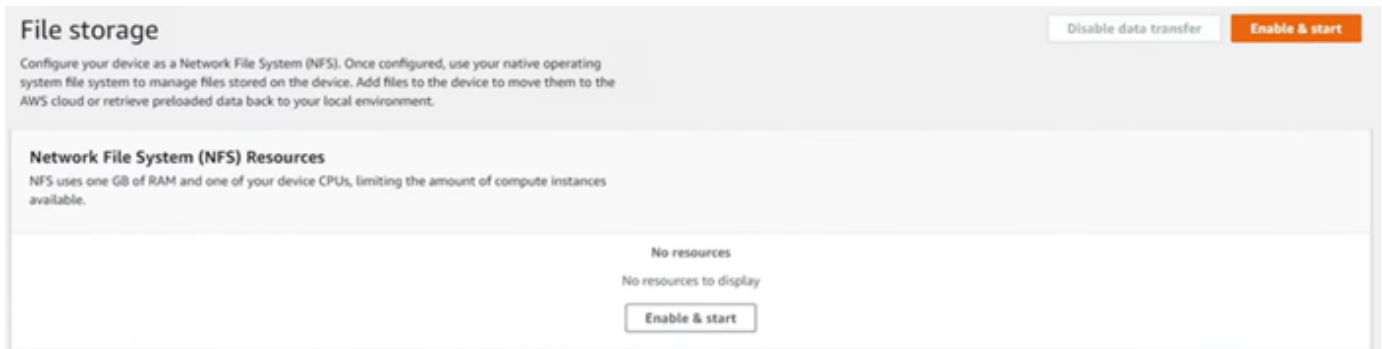
Windows 機能を有効にするには、コンピュータの管理者ユーザー名とパスワードの入力が必要になる場合があります。

3. [Services for NFS] で、[Client for NFS] を選択し、[OK] を選択します。

NFS インターフェイスの自動設定

NFS インターフェイスは Snow Family デバイスでデフォルトでは実行されていないため、デバイスでデータ転送を有効にするには起動する必要があります。数回クリックするだけで、Snow Family デバイスは NFS インターフェイスを迅速かつ自動的に設定できます。NFS インターフェイスは自分で設定することもできます。詳細については、「[NFS インターフェイスを手動で設定する](#)」を参照してください。

1. ダッシュボードの [データの転送] セクションで、[有効にして開始] を選択します。この処理が完了するまでに 1~2 分かかることがあります。



2. NFS サービスが開始されると、NFS インターフェイスの IP アドレスがダッシュボードに表示され、転送データセクションにサービスがアクティブであることを示します。
3. Open in Explorer (Windows または Linux オペレーティングシステムを使用している場合) を選択して、オペレーティングシステムのファイルブラウザでファイル共有を開き、Snow Family デバイスへのファイルの転送を開始します。クライアントコンピュータからファイル共有にファイルをコピーして貼り付けたり、ドラッグアンドドロップしたりできます。Windows オペレーティングシステムでは、ファイル共有は次の のようになります buckets(\\12.123.45.679) (Z:)

Note

Linux オペレーティングシステムでは、NFS エンドポイントをマウントするにはルートアクセス許可が必要です。

NFS インターフェイスを手動で設定する

NFS インターフェイスは Snow Family デバイスでデフォルトでは実行されていないため、デバイスでデータ転送を有効にするには起動する必要があります。Snow Family デバイスで実行されている仮

仮想ネットワークインターフェイス (VNI) の IP アドレスを指定し、必要に応じてファイル共有へのアクセスを制限することで、NFS インターフェイスを手動で設定できます。NFS インターフェイスを手動で設定する前に、Snow Family デバイスで仮想ネットワークインターフェイス (VNI) を設定します。詳細については、[コンピューティングインスタンスのネットワーク設定](#)を参照してください。

Snow Family デバイスで NFS インターフェイスを自動的に設定することもできます。詳細については、「[NFS インターフェイスの自動設定](#)」を参照してください。

1. [データの転送] セクションの下部にあるダッシュボードで、[手動設定] をクリックします。

2. [有効にして開始] をクリックして、[NFS を開始] ウィザードを開きます。[物理ネットワークインターフェイス] フィールドにデータが入力されま

Start NFS ✕

Physical network interface

RJ45: s.ni-8459d6c7273eed333 ▼

Create IP address (VNI) Use existing IP address (VNI)

IP Address assignment

DHCP ▼

Restrict NFS to allowed hosts Allow all hosts

Allowed hosts

Provide a set of CIDR blocks allowed to connect to the NFS service.

192.0.2.0/24 ✕

0.0.0.0/0 ✕

Add allowed hosts

Allow instances on this device to access NFS

Enable

Cancel Start NFS

す。

3. [IP アドレス (VNI) の作成] をクリックするか、[既存の IP アドレスを使用] をクリックします。

4. [IP Address assignment] (IP アドレスの割り当て) リストボックスで、[Create IP address (VNI)] (IP アドレス (VNI) の作成)、[DHCP] または [Static IP] (静的 IP) の順に選択します。

⚠ Important

DHCP ネットワークを使用する場合、NFS インターフェイスの IP アドレスが DHCP サーバーによって再割り当てされる可能性があります。これにより、デバイスの切断後に IP アドレスがリサイクルされる可能性があります。許可されたホスト範囲を設定し、クライアントのアドレスが変更されると、別のクライアントがそのアドレスを取得できます。この場合、新しいクライアントは共有にアクセスできます。これを防ぐには、DHCP 予約または静的 IP アドレスを使用します。

既存の IP アドレスを使用する を選択した場合は、仮想ネットワークインターフェイスのリストから仮想ネットワークインターフェイスを選択します。

5. NFS インターフェイスへのアクセスを制限して許可されたネットワークアドレスのブロックを提供するか、ネットワーク上の任意のデバイスが Snow Family デバイスの NFS インターフェイスにアクセスすることを許可するかを選択します。
 - Snow Family デバイスの NFS インターフェイスへのアクセスを制限するには、NFS を許可されたホストに制限する を選択します。許可されたホストには、一連の CIDR ブロックを入力します。複数の CIDR ブロックへのアクセスを許可する場合は、別のブロックセットを入力します。ブロックのセットを削除するには、ブロックを含むフィールドの横にある X を選択します。「許可されたホストの追加」を選択します。

i Note

許可されたホストに NFS を制限する を選択し、許可された CIDR ブロックを指定しない場合、Snow Family デバイスは NFS インターフェイスをマウントするすべてのリクエストを拒否します。

- ネットワーク上の任意のデバイスが NFS インターフェイスにアクセスできるようにするには、すべてのホストを許可する を選択します。
6. Snow Family デバイスで実行されている EC2-compatible インスタンスが NFS アダプターにアクセスできるようにするには、 を有効にするを選択します。
 7. [Start NFS] を選択します。開始まで 1~2 分かかることがあります。

⚠ Important

NFS インターフェイスの起動中は、Snow Family デバイスをオフにしないでください。

ネットワークファイルシステム (NFS) リソースセクションから、NFS インターフェイスの状態がアクティブと表示されます。クライアントコンピュータにローカルストレージとしてインターフェイスをマウントするには、IP アドレスのリストが必要です。

Snow Family デバイスの NFS エンドポイントの管理

Snow Family デバイスの各 S3 バケットはエンドポイントとして表され、マウントパスに一覧表示されます。NFS インターフェイスを起動したら、エンドポイントをマウントして、そのエンドポイントとの間でファイルを転送する。一度にマウントできるエンドポイントは 1 つだけです。別のエンドポイントをマウントするには、まず現在のエンドポイントをアンマウントします。

エンドポイントをマウントするには

1. 「マウントパス」セクションで、次のいずれかを実行してエンドポイントを選択します。
 - 「エンドポイントのフィルタリング」フィールドに、バケット名の全部または一部を入力して、エントリで使用可能なエンドポイントのリストをフィルタリングし、エンドポイントを選択します。
 - マウントパスリストでマウントするエンドポイントを選択します。
2. NFS エンドポイントのマウント を選択します。Snow Family デバイスは、エンドポイントをマウントして使用します。

エンドポイントをアンマウントするには

1. 「マウントパス」セクションで、アンマウントするエンドポイントを選択します。
2. エンドポイントのアンマウント を選択します。Snow Family デバイスはエンドポイントをアンマウントし、使用できなくなります。

Note

エンドポイントをアンマウントする前に、エンドポイントとの間でデータのコピーが行われていないことを確認してください。

クライアントコンピュータへの NFS エンドポイントのマウント

NFS インターフェイスを起動してエンドポイントをマウントしたら、エンドポイントをローカルストレージとしてクライアントコンピュータにマウントします。

1. マウントパスで、マウントするエンドポイントのコピーアイコンを選択します。エンドポイントをマウントするときにオペレーティングシステムに貼り付けます。
2. Windows、Linux、macOS オペレーティングシステム用のデフォルトのマウントコマンドを次に示します。

- Windows :

```
mount -o nolock rsize=128 wsize=128 mtype=hard nfs-interface-ip-address:/  
buckets/BucketName *
```

- Linux:

```
mount -t nfs nfs-interface-ip-address:/buckets/BucketName mount_point
```

- macOS:

```
mount -t nfs -o vers=3,rsize=131072,wsize=131072,nolocks,hard,retrans=2 nfs-  
interface-ip-address:/buckets/$bucketname mount_point
```

NFS インターフェ이스の停止

Snow Family デバイスの NFS インターフェイスとの間でファイルの転送が完了したら、NFS インターフェイスを停止します。

1. ダッシュボードから [サービス] を選択し、[ファイルストレージ] をクリックします。
2. [ファイルストレージ] ページで、[データ転送を無効にする] をクリックします。通常、NFS エンドポイントがダッシュボードから消去されるまで最大 2 分かかります。

デバイスの管理

を使用して AWS OpsHub Snow Family デバイスを管理できます。デバイスの詳細ページで、デバイスのエイリアスの変更 AWS CLI、デバイスの再起動、更新の確認など、を使用して実行するのと同じタスクを実行できます。

トピック

- [デバイスの再起動](#)
- [デバイスのシャットダウン](#)
- [デバイスのエイリアスの編集](#)
- [を使用したパブリックキー証明書の管理 OpsHub](#)
- [デバイスと AWS OpsHub アプリケーションの更新の取得](#)
- [プロフィールの管理](#)

デバイスの再起動

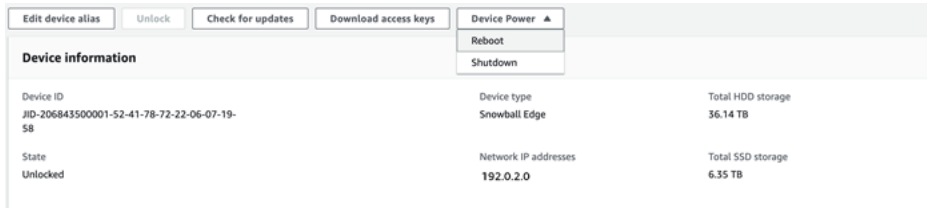
を使用して Snow デバイスを再起動するには AWS OpsHub、次の手順に従います。

Important

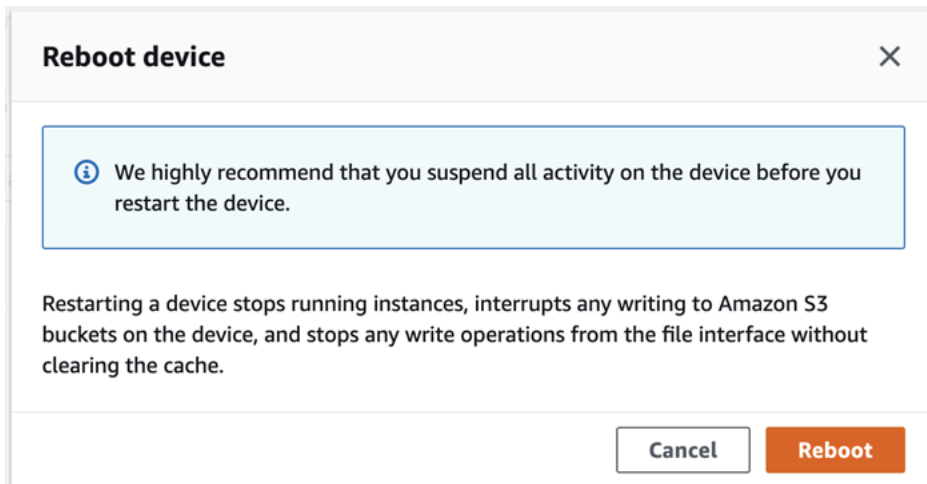
デバイスを再起動する前に、デバイスでのすべてのアクティビティを停止することが強く推奨されます。デバイスを再起動すると、実行中のインスタンスが停止し、デバイス上の Amazon S3 バケットへの書き込みが中断されます。

デバイスを再起動するには

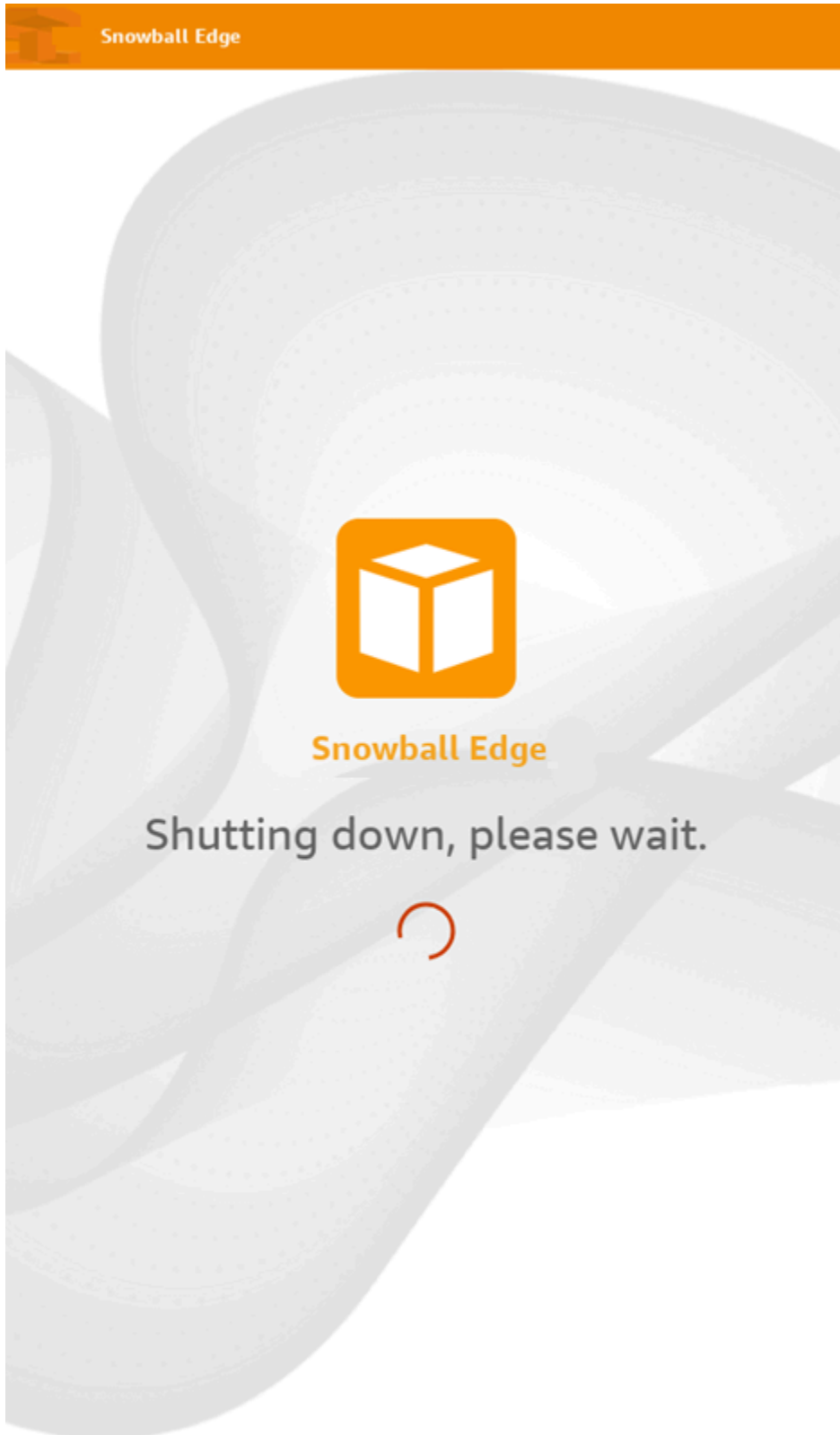
1. AWS OpsHub ダッシュボードで、デバイス でデバイスを見つけます。次に、デバイスを選択して、デバイスの詳細ページを開きます。
2. [デバイスの電源] メニューを選択し、[再起動] を選択します。ダイアログボックスが表示されます。



3. ダイアログボックスで [再起動] を選択します。デバイスが再起動を開始します。



デバイスのシャットダウン中、LCD 画面にはデバイスがシャットダウン中であることを示すメッセージが表示されます。



デバイスのシャットダウン

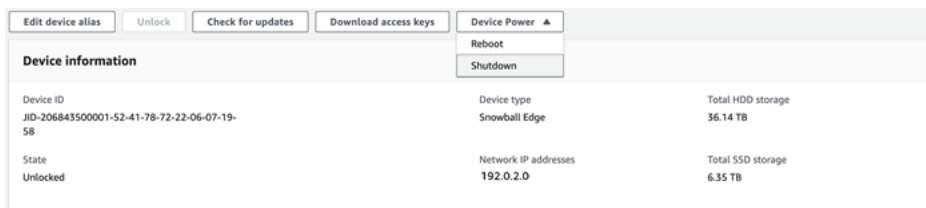
を使用して Snow デバイスをシャットダウンするには AWS OpsHub 、次の手順に従います。

⚠ Important

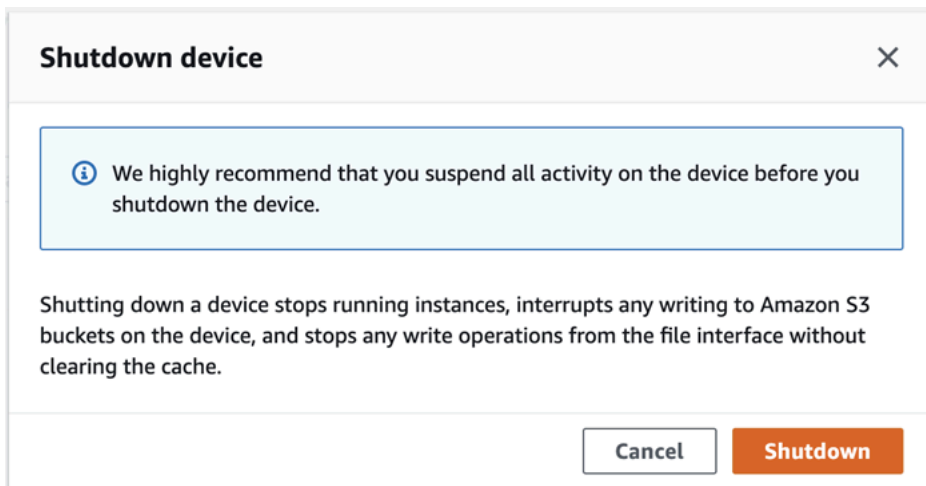
デバイスをシャットダウンする前に、デバイスですべてのアクティビティを停止することが強く推奨されます。デバイスをシャットダウンすると、実行中のインスタンスが停止し、デバイス上の Amazon S3 バケットへの書き込みが中断されます。

デバイスをシャットダウンするには

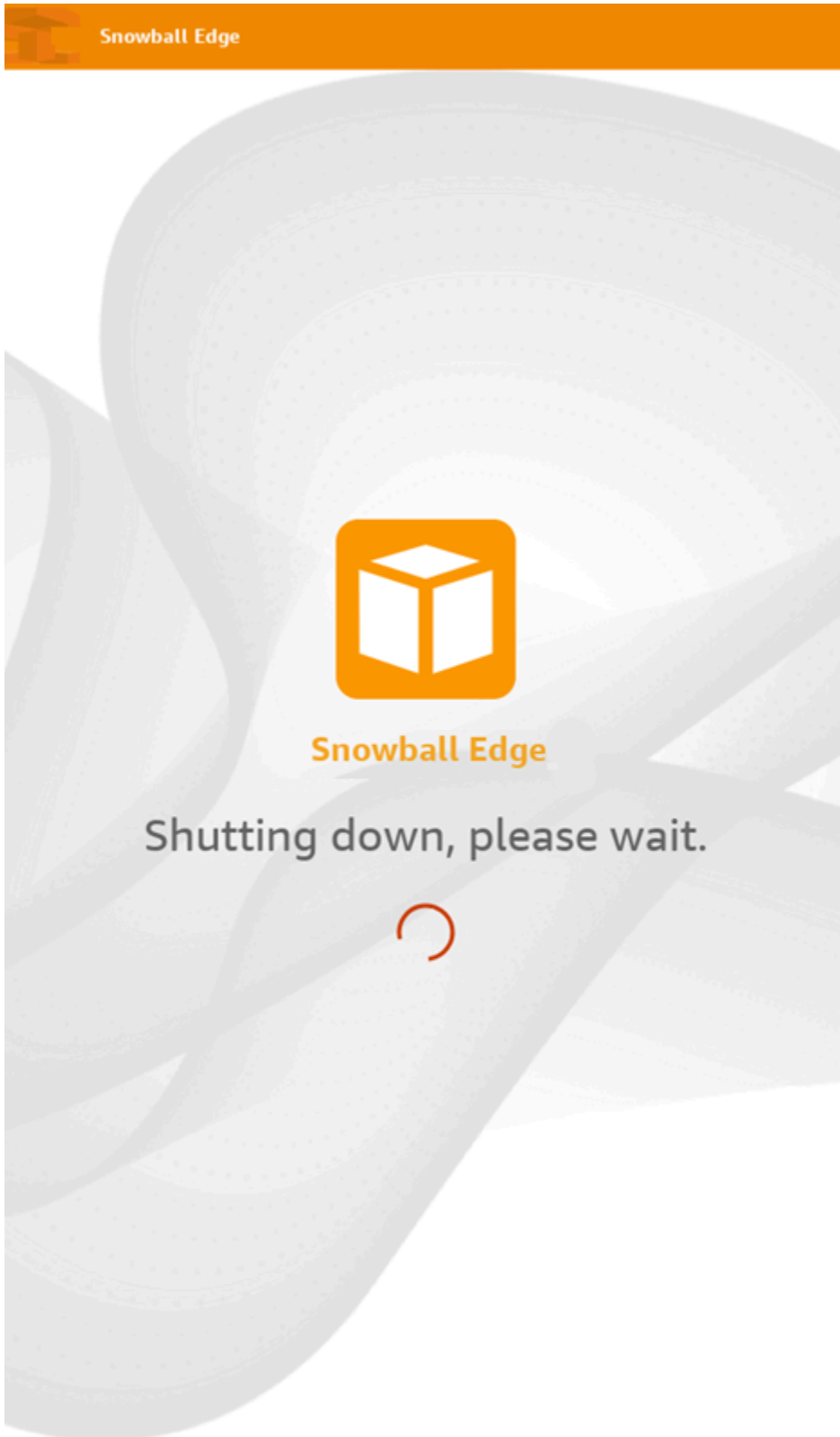
1. AWS OpsHub ダッシュボードで、デバイス でデバイスを見つけます。次に、デバイスを選択して、デバイスの詳細ページを開きます。
2. [デバイスの電源] メニューを選択し、[シャットダウン] を選択します。ダイアログボックスが表示されます。



3. ダイアログボックスで、[シャットダウン] を選択します。デバイスのシャットダウンが開始されます。



デバイスのシャットダウン中、LCD 画面にはデバイスがシャットダウン中であることを示すメッセージが表示されます。

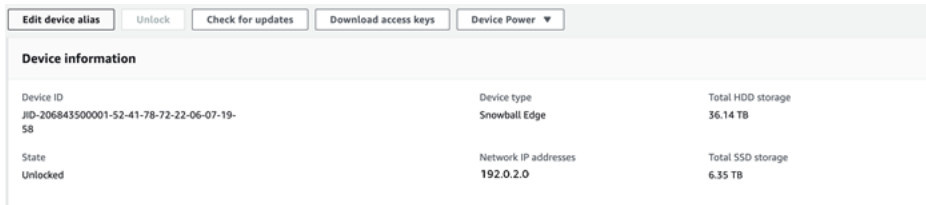


デバイスのエイリアスの編集

を使用してデバイスエイリアスを編集するには、次の手順に従います AWS OpsHub。

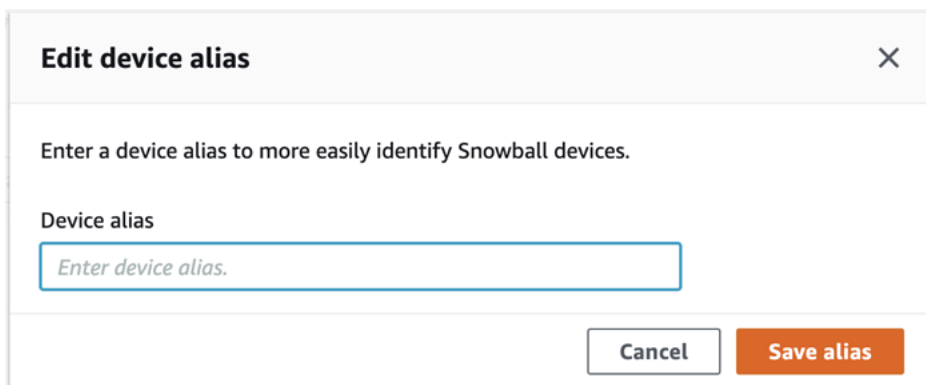
デバイスのエイリアスを編集するには

1. AWS OpsHub ダッシュボードで、デバイス でデバイスを見つけます。デバイスを選択して、デバイスの詳細ページを開きます。
2. [Edit device alias] タブを選択します。



Device information		
Device ID JID-206843500001-52-41-78-72-22-06-07-19-58	Device type Snowball Edge	Total HDD storage 36.14 TB
State Unlocked	Network IP addresses 192.0.2.0	Total SSD storage 6.35 TB

3. [Device alias] に新しい名前を入力し、[Save alias] を選択します。



Edit device alias

Enter a device alias to more easily identify Snowball devices.

Device alias

Cancel Save alias

を使用したパブリックキー証明書の管理 OpsHub

パブリックキー証明書を提供することで、HTTPS プロトコルを介して Snowball Edge デバイスまたは Snowball Edge デバイスのクラスターで実行されている AWS サービスと安全にやり取りできます。HTTPS プロトコルを使用して、Snow Family デバイスの IAM、Amazon EC2、S3 アダプター、Amazon S3 互換ストレージ、Amazon EC2 Systems Manager、Snowball Edge デバイス AWS STS などの AWS サービスとやり取りできます。デバイスのクラスターの場合、1つの証明書が必要で、クラスター内のどのデバイスでも生成できます。Snowball Edge デバイスで証明書が生成され、デバイスのロックを解除すると、Snowball Edge クライアントコマンドを使用して証明書を一覧表示、取得、削除できます。

Snowball Edge デバイスは、以下のイベントが発生すると証明書を生成します。

- Snowball Edge デバイスまたはクラスターが初めてロック解除される。
- Snowball Edge デバイスまたはクラスターは、証明書を削除した後 (delete-certificate コマンドまたは の証明書の更新を使用) にロック解除されます AWS OpsHub。
- 証明書の有効期限が切れた後に Snowball Edge デバイスまたはクラスターが再起動およびロック解除されます。

新しい証明書が生成されるたびに、古い証明書は無効になります。証明書は、生成された日から 1 年間有効です。

Snowball Edge クライアントを使用して公開鍵証明書を管理することもできます。詳細については、「[公開鍵証明書の管理](#)」を参照してください。

トピック

- [を使用してパブリックキー証明書をダウンロードする OpsHub](#)
- [を使用したパブリックキー証明書の更新 OpsHub](#)

を使用してパブリックキー証明書をダウンロードする OpsHub

アクティブな公開鍵証明書はコンピュータにダウンロードできます。

1. AWS OpsHub ダッシュボードで、デバイス でデバイスを見つけます。デバイスを選択して、デバイスの詳細ページを開きます。
2. デバイスの詳細ページで、[証明書を管理] メニューを選択します。メニューから [証明書をダウンロード] を選択します。
3. 表示されるウィンドウで、ダウンロードする証明書ファイルに名前を付け、コンピュータ上のダウンロード先の場所を選択できます。[保存] を選択します。

を使用したパブリックキー証明書の更新 OpsHub

公開鍵証明書を更新する前に、Snow Family デバイスとの間のデータ転送をすべて停止し、実行中の EC2 互換インスタンスをすべて停止してください。詳細については、このガイドの [「Amazon EC2-compatible インスタンスの停止」](#) を参照してください。

1. AWS OpsHub ダッシュボードで、デバイス でデバイスを見つけます。デバイスを選択して、デバイスの詳細ページを開きます。

2. デバイスの詳細ページで、[証明書管理] メニューを選択します。メニューから [証明書を更新] を選択します。
3. [証明書を更新] ウィンドウで、フィールドに **Renew** と入力し、[更新] を選択します。Snow Family デバイスが既存の公開鍵証明書を削除し、デバイスまたはクラスターを再起動します。

Renew certificate



The following certificate will be deleted:

arn:aws:snowball-device:::certificate/example



Stop all activity on the Snow device or cluster before proceeding.

Clicking **Renew** will automatically reboot **all devices attached to this certificate** and terminate any ongoing data transfers and other running processes. A new certificate will be generated when you unlock the device or cluster after it reboots.

To confirm, enter **Renew** in the field and then choose **Renew**

Cancel

Renew

デバイスと AWS OpsHub アプリケーションの更新の取得

デバイスの更新をチェックしてインストールできます。アプリケーションを自動的に最新バージョン AWS OpsHub に更新するようにを設定することもできます。

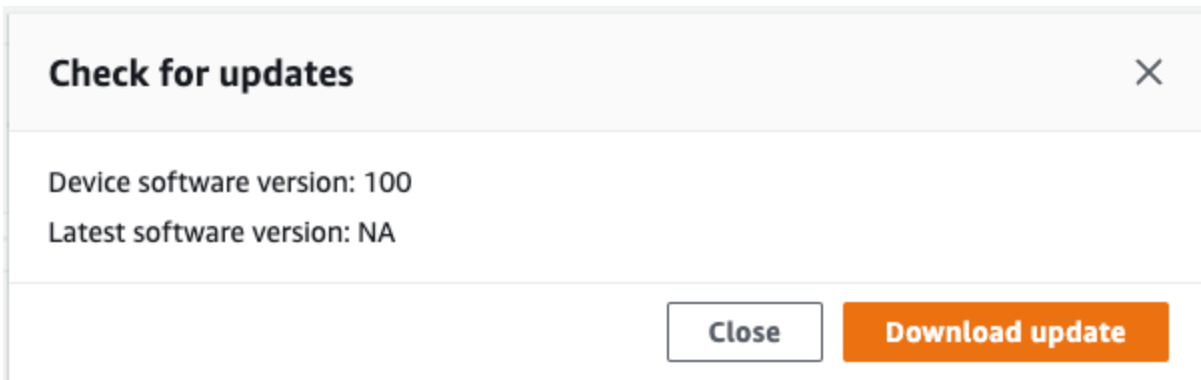
デバイスの更新

を使用して Snow デバイスを更新するには AWS OpsHub 、次の手順に従います。

デバイスを更新するには

1. AWS OpsHub ダッシュボードで、デバイス でデバイスを見つけます。デバイスを選択して、デバイスの詳細ページを開きます。
2. [Check for updates] タブを選択します。

[Check for updates] ページには、デバイスの現在のソフトウェアバージョンと、最新のソフトウェアバージョン (利用可能な場合) が表示されます。



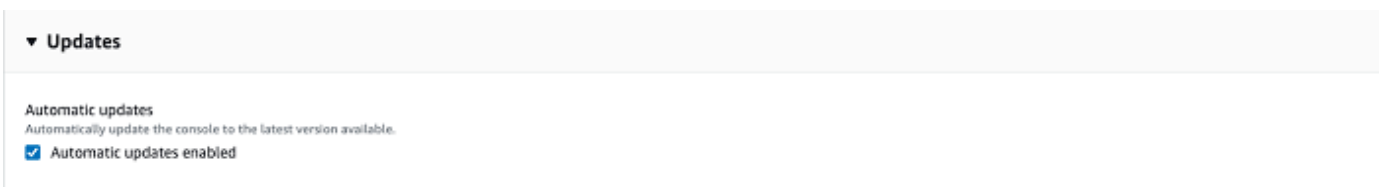
3. 更新が利用可能な場合は、[更新をダウンロード] を選択します。それ以外の場合は、[Close (閉じる)] を選択します。

AWS OpsHub アプリケーションの更新

AWS OpsHub はアプリケーションを自動的に最新バージョンに更新します。自動更新が有効になっていることを確認するには、次のステップを実行します。

で自動更新が有効になっていることを確認するには AWS OpsHub

1. AWS OpsHub ダッシュボードで、設定 を選択します。
2. [Updates] (更新) タブを開きます。
3. [Automatic updates enabled] (自動更新の有効化) が選択されていることを確認します。自動更新はデフォルトで有効になっています。



自動更新が有効になっていない場合、AWS OpsHub アプリケーションの最新バージョンは取得されません。

プロファイルの管理

ローカルファイルシステム上に認証情報を永続的に保存するためのプロファイルを作成できます。を使用すると AWS OpsHub、デバイスの IP アドレス、ロック解除コード、マニフェストファイルを使用してデバイスのロックを解除するたびに、新しいプロファイルを作成できます。

Snowball Edge クライアントを使用して、任意にプロファイルを作成することもできます。

「[Snowball Edge クライアント用プロファイルの設定](#)」を参照してください。

プロファイルを編集または削除するには、テキストエディタでプロファイルファイルを編集します。

Example `snowball-edge.config` ファイルの例

この例は、SnowDevice1profile、SnowDevice2profile、SnowDevice3profile の 3 つのプロファイルを含むプロファイルファイルを示しています。

```
{"version":1,"profiles":
  {
    "SnowDevice1profile":
      {
        "name":"SnowDevice1profile",
        "jobId":"JID12345678-136f-45b4-b5c2-847db8adc749",
        "unlockCode":"db223-12345-dbe46-44557-c7cc2",
        "manifestPath":"C:\\Users\\Administrator\\.aws\\ops-hub\\manifest\\
\\JID12345678-136f-45b4-b5c2-847db8adc749_manifest-1670622989203.bin",
        "defaultEndpoint":"https://10.16.0.1",
        "isCluster":false,
        "deviceIps":[]
      },
    },
    "SnowDevice2profile":
      {
        "name":"SnowDevice2profile",
        "jobId":"JID12345678-fdb2-436a-a4ff-7c510dec1bae",
        "unlockCode":"b893b-54321-0f65c-6c5e1-7f748",
        "manifestPath":"C:\\Users\\Administrator\\.aws\\ops-hub\\manifest\\JID12345678-
fdb2-436a-a4ff-7c510dec1bae_manifest-1670623746908.bin",
```

```
    "defaultEndpoint": "https://10.16.0.2",
    "isCluster": false,
    "deviceIps": []
  },
  "SnowDevice3profile":
  {
    "name": "SnowDevice3profile",
    "jobId": "JID12345678-c384-4a5e-becd-ab5f38888463",
    "unlockCode": "64c89-13524-4d054-13d93-c1b80",
    "manifestPath": "C:\\Users\\Administrator\\.aws\\ops-hub\\manifest\\JID12345678-
c384-4a5e-becd-ab5f38888463_manifest-1670623999136.bin",
    "defaultEndpoint": "https://10.16.0.3",
    "isCluster": false,
    "deviceIps": []
  }
}
```

プロファイルを作成するには

1. デバイスをローカルでロック解除し、「[デバイスのロック解除](#)」の手順に従ってサインインします。
2. プロファイルに名前を付け、[プロファイル名を保存] を選択します。

プロファイルを編集するには

1. テキストエディタで home directory\\.aws\\snowball\\config から snowball-edge.config を開きます。
2. 必要に応じてファイルを編集します。例えば、プロファイル内のデバイスの IP アドレスを変更するには、defaultEndpoint エントリを変更します。
3. ファイルを保存して閉じます。

プロファイルを削除するには

1. テキストエディタを使用して、home directory\\.aws\\snowball\\config から snowball-edge.config を開きます。
2. プロファイル名を含む行、プロファイル名の後に続く中括弧 {}、括弧内の内容を削除します。
3. ファイルを保存して閉じます。

管理タスクの自動化

を使用して AWS OpsHub、Snow Family デバイスで頻繁に実行する運用タスクを自動化できます。仮想サーバーの再起動、Amazon EC2 互換インスタンスの停止など、リソースで繰り返し実行するアクションのタスクを作成できます。オペレーションタスクを安全に実行し、AWS リソースに対してオペレーションを一括で実行するオートメーションドキュメントを提供します。一般的な IT ワークフローをスケジュールすることもできます。

Note

タスクの自動化は、クラスターではサポートされていません。
タスクを使用するには、まず Amazon EC2 Systems Manager サービスを開始する必要があります。Snowball Edge でサービスを開始するには、「[Snowball Edge でのサービスの開始](#)」を参照してください。

トピック

- [タスクの作成と開始](#)
- [タスクの詳細の表示](#)
- [タスクの削除](#)

タスクの作成と開始

タスクを作成するときは、タスクを実行するリソースの種類を指定し、タスクを実行する手順が示されたタスクドキュメントを提供します。タスクドキュメントは YAML 形式または JSON 形式のいずれかです。次に、タスクに必要なパラメータを指定し、タスクを開始します。

タスクを作成するには

1. ダッシュボードの [Launch tasks] セクションで、[Get started] を選択して [タスク] ページを開きます。作成したタスクは、[タスク] の下に表示されます。
2. [タスクの作成] を選択し、タスクの詳細を入力します。
3. [名前] に、タスクの一意な名前を入力します。

i Tip

名前は 3~128 文字にしてください。有効な文字は a-z、A-Z、0-9、.、_、および - です。

- 必要に応じて、[Target type-optional] リストからターゲットタイプを選択できます。これは、タスクを実行するリソースの種類です。

例えば、Amazon EC2 互換インスタンスで実行するタスクには `/AWS::EC2::Instance` を、すべてのリソースタイプで実行するには `/` を指定できます。

- [Content] (コンテンツ) セクションで [YAML] または [JSON] を選択し、タスクを実行するスクリプトを指定します。YAML 形式または JSON 形式の 2 つのオプションがあります。例については、「[タスクの例](#)」を参照してください。
- [作成] を選択します。作成したタスクが [タスク] ページに表示されます。

タスクを開始します。

- ダッシュボードの [Launch tasks] セクションで、[Get started] を選択して [タスク] ページを開きます。タスクが [タスク] の下に表示されます。
- タスクを選択して、[Start task] ページを開きます。
- ターゲットで実行するには、[シンプルな実行] を選択します。

複数のターゲットで安全に実行し、同時実行性とエラーのしきい値を定義するには、[レート制御] を選択します。このオプションでは、[レート制御] セクションで追加のターゲットおよびエラーのしきい値情報を指定します。

- 必要な入力パラメータを指定し、[Start task] を選択します。

タスクの状態は [Pending] になり、タスクが正常に実行されると [Success] に変わります。

タスクの例

次の例では、Amazon EC2 互換インスタンスを再起動します。これには、`endpoint` と `instance ID` の 2 つの入力パラメータが必要です。

YAML の例


```
description: Restart EC2 instance
schemaVersion: '0.3'
parameters:
  Endpoint:
    type: String
    description: (Required) EC2 Service Endpoint URL
  Id:
    type: String
    description: (Required) Instance Id
mainSteps:
- name: restartInstance
  action: aws:executeScript
  description: Restart EC2 instance step
  inputs:
    Runtime: python3.7
    Handler: restart_instance
    InputPayload:
      Endpoint: "{{ Endpoint }}"
      Id: "{{ Id }}"
    TimeoutSeconds: 30
    Script: |-
      import boto3
      import time
      def restart_instance(payload, context):
          ec2_endpoint = payload['Endpoint']
          instance_id = payload['Id']
          ec2 = boto3.resource('ec2', endpoint_url=ec2_endpoint)
          instance = ec2.Instance(instance_id)
          if instance.state['Name'] != 'stopped':
              instance.stop()
              instance.wait_until_stopped()
          instance.start()
          instance.wait_until_running()
          return {'InstanceState': instance.state}
```

JSON の例

```
{
  "description" : "Restart EC2 instance",
  "schemaVersion" : "0.3",
  "parameters" : {
    "Endpoint" : {
```

```
    "type" : "String",
    "description" : "(Required) EC2 Service Endpoint URL"
  },
  "Id" : {
    "type" : "String",
    "description" : "(Required) Instance Id"
  }
},
"mainSteps" : [ {
  "name" : "restartInstance",
  "action" : "aws:executeScript",
  "description" : "Restart EC2 instance step",
  "inputs" : {
    "Runtime" : "python3.7",
    "Handler" : "restart_instance",
    "InputPayload" : {
      "Endpoint" : "{{ Endpoint }}",
      "Id" : "{{ Id }}"
    }
  },
  "TimeoutSeconds" : 30,
  "Script" : "import boto3\nimport time\ndef restart_instance(payload, context):\n\n    ec2_endpoint = payload['Endpoint']\n    instance_id = payload['Id']\n    ec2 = boto3.resource('ec2', endpoint_url=ec2_endpoint)\n    instance = ec2.Instance(instance_id)\n    if instance.state['Name'] != 'stopped':\n    instance.stop()\n    instance.wait_until_stopped()\n    instance.start()\n    instance.wait_until_running()\n    return {'InstanceState': instance.state}"
}
] ]
}
```

タスクの詳細の表示

タスクの実行に必要な説明やパラメータなど、管理タスクの詳細を表示できます。

タスクの詳細を表示するには

1. ダッシュボードの [Launch tasks] セクションで、[Get started] を選択して [タスク] ページを開きます。

2. [タスク] ページで、詳細を表示するタスクを見つけて選択します。
3. [詳細を表示] を選択し、いずれかのタブを選択して詳細を表示します。たとえば、[パラメータ] タブには、スクリプトの入力パラメータが表示されます。

タスクの削除

管理タスクを削除するには、次のステップに従います。

タスクを削除するには

1. ダッシュボードの [Launch tasks] セクションで、[Get started] を選択して [タスク] ページを開きます。
2. 削除するタスクを見つけます。タスクを選択し、[削除] を選択します。

デバイスの NTP タイムサーバーの設定

次の手順に従って、デバイスが時刻を同期する必要があるタイムサーバーを表示および更新します。

タイムソースをチェックするには

1. AWS OpsHub ダッシュボードで、デバイス でデバイスを見つけます。デバイスを選択して、デバイスの詳細ページを開きます。
2. デバイスが時刻を同期しているタイムソースのリストが、[Time sources] (タイムソース) テーブルに表示されます。

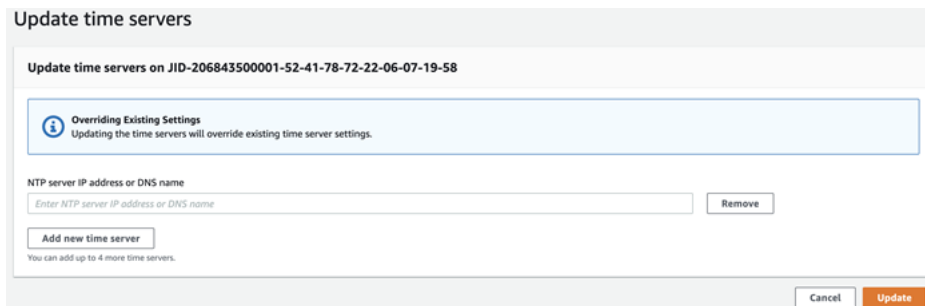
タイムソーステーブルには 4 つの列があります。

- [Address] (アドレス): タイムソースの DNS 名/ IP アドレス
- [State] (状態): デバイスとそのタイムソース間の現在の接続ステータス。5 つの状態があります。
 - [CURRENT]: タイムソースは、現在、時刻の同期に使用されています
 - [COMBINED]: タイムソースは現在のソースと組み合わせられています
 - [EXCLUDED]: タイムソースは結合アルゴリズムによって除外されています
 - [LOST]: タイムソースとの接続が失われています
 - [UNAVAILABILITY]: 結合アルゴリズムが偽ティッカーであるかばらつきが多すぎるとみなされた、無効なタイムソースです

- [Type]: ネットワークタイムプロトコル (NTP) ソースはサーバーでもピアでもかまいません。サーバーは、ユーザーが `update-time-server` コマンドを使用して設定できます。一方、ピアはクラスター内の他の Snowball Edge デバイスを一時停止することでしか設定できず、クラスターが関連付けられると自動的に設定されます。
- [Stratum] (階層): ソースの階層。[Stratum 1] (階層 1) は、ローカルでアタッチされたリファレンスクロックを持つソースを示します。階層 1 のソースに同期されたソースは [Stratum 2] (階層 2) に設定されます。階層 2 のソースに同期されたソースは [Stratum 3] (階層 3)、というように設定されます。

タイムサーバーを更新するには

1. AWS OpsHub ダッシュボードで、デバイス でデバイスを見つけます。デバイスを選択して、デバイスの詳細ページを開きます。
2. デバイスが時刻を同期しているタイムソースのリストが、[Time sources] (タイムソース) テーブルに表示されます。
3. [Time sources] (タイムソース) テーブルで、[Update time servers] (タイムサーバーの更新) を選択します。
4. デバイスの時刻を同期させるタイムサーバーの DNS 名または IP アドレスを入力し、[Update] (更新) を選択します。



Update time servers

Update time servers on JID-206843500001-52-41-78-72-22-06-07-19-58

ⓘ Overriding Existing Settings
Updating the time servers will override existing time server settings.

NTP server IP address or DNS name

You can add up to 4 more time servers.

サポートされている NTP デバイスタイプとソフトウェアバージョン

NTP は、バージョン 2 のどのストレージデバイスやコンピューティングデバイスでも使用できません。ただし、ソフトウェアバージョン 77 以降の Snowball Edge バージョン 3 ストレージおよびコンピューティングデバイスタイプは NTP をサポートします。NTP が有効になっているかどうかを確認するには、Snowball Edge CLI コマンド `describe-time-sources` を使用します。

AWS Snowball エッジデバイスの使用

以下に、AWS Snowball Edge デバイスの概要を示します。Snowball Edge は、AWS Key Management Service (AWS KMS) で保護された物理的に堅牢なデバイスで、ローカルストレージとコンピューティング、またはオンプレミスサーバーと Amazon Simple Storage Service (Amazon S3) 間のデータ転送に使用します。

AWS Snowball Edge デバイスのロック解除については、「」を参照してください[Snow Family デバイスのロック解除](#)。

デバイスが到着したら、まず、損傷や明らかな改ざんがないか検査してください。

Warning

デバイスに疑わしい点が見つかった場合は、内部ネットワークに接続しないでください。[AWS Support](#) にお問い合わせいただければ、新しいデバイスをお客様宛に配送します。

次の画像は、AWS Snowball Edge デバイスの様子を示しています。



デバイスには、前面、背面、上部にラッチで開閉する 3 つのドアがあります。上部ドアにはデバイスの電源コードが入っています。他の 2 つのドアは、開いてデバイスの中にスライドさせることができるため、使用中に邪魔になりません。ドアを開くと、デバイス前面に埋め込まれた LCD E Ink ディスプレイと、背面の電源ポートおよびネットワークポートにアクセスできます。

デバイス到着後に電源をオンにしたら、使用準備完了です。

トピック

- [Snowball Edge クライアントコマンドの使用](#)
- [データ移行に Amazon S3 アダプターを使用してファイルを転送する](#)
- [NFS インターフェイスの管理](#)
- [AWS IoT Greengrass を使用して、Amazon EC2 互換インスタンスでプリインストールされたソフトウェアを実行する](#)
- [AWS Snowball Edge AWS Lambda での使用](#)

- [Amazon EC2 互換コンピューティングインスタンスの使用](#)
- [Snow Family デバイスの Amazon S3 互換ストレージを使用する](#)
- [AWS Snow での Amazon EKS Anywhere の使用](#)
- [ローカルでの IAM の使用](#)
- [の使用 AWS Security Token Service](#)
- [公開鍵証明書の管理](#)
- [エッジデバイスで AWS サービス AWS Snowball を使用するために必要なポート](#)

Snowball Edge クライアントコマンドの使用

以下は、Snow Family デバイスで使用する Snowball Edge クライアントコマンドに関する情報です。各コマンドには、使用例とサンプル出力が含まれています。Snowball Edge クライアントは、デバイスで実行するスタンドアロンのコマンドラインアプリケーションであり、Snow Family デバイスのロックを解除して設定および制御します。Snow Family デバイスのクラスターで クライアントを使用することもできます。Snowball Edge クライアントの使用中に snowballEdge help コマンドを実行することにより、追加のサポート情報を取得できます。

Snowball Edge クライアントをダウンロードするには、「」を参照してください[Snowball Edge クライアントのダウンロードとインストール](#)。

トピック

- [Snowball Edge クライアント用プロファイルの設定](#)
- [NFC 検証の QR コードの取得](#)
- [Snowball Edge クライアントバージョン](#)
- [Snowball Edge デバイスのロック解除](#)
- [Snowball Edge の更新](#)
- [認証情報の取得](#)
- [Snowball Edge でのサービスの開始](#)
- [Snowball Edge でのサービスの停止](#)
- [NFS の開始とアクセスの制限](#)
- [NFS 実行中に NFS 共有へのアクセスを制限する](#)
- [AWS Snowball Edge ログ](#)
- [デバイスステータスの取得](#)

- [サービスステータスの取得](#)
- [クラスターからノードを削除する](#)
- [クラスターへのノードの追加](#)
- [デバイス用のタグの作成](#)
- [デバイスからのタグの削除](#)
- [デバイスのタグの説明](#)
- [ダイレクトネットワークインターフェイスの作成](#)
- [ダイレクトネットワークインターフェイスに関する情報の取得](#)
- [ダイレクトネットワークインターフェイスの更新](#)
- [ダイレクトネットワークインターフェイスの削除](#)
- [仮想ネットワークインターフェイス \(VNI\) を作成する](#)
- [仮想ネットワークインターフェイスに関する情報の取得](#)
- [仮想ネットワークインターフェイスの更新](#)
- [仮想ネットワークインターフェイスの削除](#)
- [機能ステータスのチェック](#)
- [タイムサーバーの設定](#)
- [タイムソースのチェック](#)
- [MTU サイズの更新](#)

Snowball Edge クライアント用プロファイルの設定

Snowball Edge クライアントのコマンドを実行するたびに、マニフェストファイル、解除コード、IP アドレスを指定します。これらのうち最初の 2 つは、AWS Snow ファミリーマネジメントコンソール または ジョブ管理 API から取得できます。マニフェストとロック解除コードの取得の詳細については、「[Snow Family デバイスにアクセスするための認証情報の取得](#)」を参照してください。

`snowballEdge configure` コマンドを使用して、マニフェストへのパス、29 文字のロック解除コード、およびエンドポイントをプロファイルとして保存するオプションがあります。設定すると、特定のジョブ用にこれらの値を手動で入力することなく、他の Snowball Edge クライアントのコマンドを使用できます。Snowball Edge クライアントの設定後、その情報はプレーンテキストの JSON 形式で `home directory/.aws/snowball/config/snowball-edge.config` に保存されます。

エンドポイントは IP アドレスで、`https://` が追加されます。AWS Snowball Edge デバイス LCD ディスプレイで AWS Snowball Edge デバイスの IP アドレスを見つけることができます。AWS Snowball Edge デバイスが初めてネットワークに接続されると、DHCP サーバーが使用可能な場合、DHCP IP アドレスが自動的に取得されます。別の IP アドレスを使用する場合は、LCD ディスプレイから変更できます。詳細については、「[AWS Snowball エッジデバイスの使用](#)」を参照してください。

⚠ Important

設定ファイルにアクセスできるユーザーは誰でも、Snowball Edge デバイスまたはクラスター上のデータにアクセスできます。このファイルのローカルアクセスコントロールを管理することは、管理者の責任の 1 つです。

使用方法

このコマンドは 2 つの方法で (インライン、または求められたとき) 使用できます。この使用例は、求められたときの方法を示しています。

```
snowballEdge configure
```

Example 出力

```
Configuration will stored at home directory\.aws\snowball\config\snowball-edge.config  
Snowball Edge Manifest Path: /Path/to/manifest/file  
Unlock Code: 29 character unlock code  
Default Endpoint: https://192.0.2.0
```

同時に複数のジョブがある場合や、別のエンドポイントからクラスターを管理するオプションが必要な場合は、複数のプロファイルを持つことができます。複数の AWS CLI プロファイルの詳細については、「[ユーザーガイド](#)」の「[名前付きプロファイル](#)」を参照してください。

NFC 検証の QR コードの取得

このコマンドを使用して、AWS Snowball Edge Verification App で使用するデバイス固有の QR コードを生成できます。NFC 検証の詳細は、[NFC タグの検証](#) を参照してください。

使用方法

```
snowballEdge get-app-qr-code --output-file ~/downloads/snowball-qr-code.png
```

Example 出力

```
QR code is saved to ~/downloads/snowball-qr-code.png
```

Snowball Edge クライアントバージョン

version コマンドを使用して、Snowball Edge コマンドラインインターフェイス (CLI) クライアントのバージョンを確認します。

使用方法

```
snowballEdge version
```

出力例

```
Snowball Edge client version: 1.2.0 Build 661
```

Snowball Edge デバイスのロック解除

スタンドアロン AWS Snowball Edge デバイスのロックを解除するには、snowballEdge unlock-device コマンドを実行します。クラスターをロック解除するには、snowballEdge unlock-cluster コマンドを使用します。このコマンドにより、AWS Snowball Edge デバイスへのアクセスを認証します。

Note

ジョブに関連付けられたデバイスのロックを解除するには、デバイスがオンサイトにあり、電源とネットワークに接続され、電源が入っている必要があります。さらに、AWS Snowball Edge デバイスの前面にある LCD ディスプレイには、デバイスが使用可能であることを示す必要があります。

使用方法

```
snowballEdge unlock-device --endpoint https://192.0.2.0 --manifest-file Path/to/manifest/file --unlock-code 01234-abcde-ABCDE-01234
```

Example 1 つのデバイスをロック解除する入力

```
snowballEdge unlock-device --endpoint https://192.0.2.0 --manifest-file /usr/home/manifest.bin --unlock-code 01234-abcde-ABCDE-01234
```

Example 1 つのデバイスをロック解除する出力

Your Snowball Edge device is unlocking. You may determine the unlock state of your device using the describe-device command. Your Snowball Edge device will be available for use when it is in the UNLOCKED state.

クラスターの使用

クラスターのロックを解除するときは、いずれかのノードのエンドポイントと、クラスター内の他のデバイスのすべての IP アドレスを指定します。

```
snowballEdge unlock-cluster --endpoint https://192.0.2.0 --manifest-file Path/to/manifest/file --unlock-code 01234-abcde-ABCDE-01234 --device-ip-addresses 192.0.2.0 192.0.2.1 192.0.2.2 192.0.2.3 192.0.2.4
```

Example クラスターのロックを解除する出力

Your Snowball Edge Cluster is unlocking. You may determine the unlock state of your cluster using the describe-device command. Your Snowball Edge Cluster will be available for use when your Snowball Edge devices are in the UNLOCKED state.

Snowball Edge の更新

次のコマンドを使用して、Snowball Edge デバイスの更新をダウンロードしてインストールできます。これらのコマンドを使用する手順については、「[Snowball Edge デバイスのソフトウェア更新](#)」を参照してください。

`snowballEdge check-for-updates` – クラウド上で利用可能な Snowball Edge ソフトウェアに関するバージョン情報、およびデバイスにインストールされている現在のバージョンを返します。

使用法 (Snowball Edge クライアント設定済み)

```
snowballEdge check-for-updates
```

Example 出力

```
Latest version: 102  
Installed version: 101
```

`snowballEdge describe-device-software` – デバイスの SSL 証明書の現在のソフトウェアバージョンと有効期限を返します。また、ソフトウェア更新をダウンロード中またはインストール中の場合、状態も表示されます。可能性のある出力のリストは、次のとおりです。

- NA – 現在進行中のソフトウェアの更新はありません。
- Downloading – 新しいソフトウェアをダウンロードしています。
- Installing – 新しいソフトウェアをインストールしています。
- Requires Reboot – 新しいソフトウェアがインストールされ、デバイスを再起動する必要があります。

Warning

デバイスを再起動する前に、デバイスでのすべてのアクティビティを停止することが強く推奨されます。デバイスを再起動すると、実行中のインスタンスが停止し、デバイス上の Amazon S3 バケットへの書き込みが中断されます。これらのプロセスのすべてのデータが失われる可能性があります。

使用法 (Snowball Edge クライアント設定済み)

```
snowballEdge describe-device-software
```

Example 出力

```
Installed version: 101  
Installing version: 102  
Install State: Downloading  
CertificateExpiry: Thur Jan 01 00:00:00 UTC 1970
```

`snowballEdge download-updates` – Snowball Edge の最新ソフトウェア更新のダウンロードを開始します。

使用法 (Snowball Edge クライアント設定済み)

```
snowballEdge download-updates
```

Example 出力

```
Download started. Run describe-device-software API for additional information.
```

`snowballEdge install-updates` – 既にダウンロードされた Snowball Edge の最新のソフトウェア更新のインストールを開始します。

使用法 (Snowball Edge クライアント設定済み)

```
snowballEdge install-updates
```

Example 出力

```
Installation started.
```

`snowballEdge reboot-device` – デバイスを再起動します。

Warning

デバイスを再起動する前に、デバイスでのすべてのアクティビティを停止することが強く推奨されます。デバイスを再起動すると、実行中のインスタンスが停止し、デバイス上の Amazon S3 バケットへの書き込みが中断されます。これらのプロセスのすべてのデータが失われる可能性があります。

使用法 (Snowball Edge クライアント設定済み)

```
snowballEdge reboot-device
```

Example 出力

```
Rebooting device now.
```

`snowballEdge configure-auto-update-strategies` – 自動更新戦略を設定します。

使用法 (Snowball Edge クライアント設定済み)

```
snowballEdge configure-auto-update-strategy --auto-check autoCheck [--auto-check-  
frequency  
autoCheckFreq] --auto-download autoDownload  
[--auto-download-frequency autoDownloadFreq]  
--auto-install autoInstall  
[--auto-install-frequency autoInstallFreq]  
--auto-reboot autoReboot [--endpoint  
endpoint]
```

Example 出力

```
Successfully configured auto update strategy. Run describe-auto-update-strategies for  
additional information.
```

snowballEdge describe-auto-update-strategies – 現在設定されているすべての自動更新戦略を返します。

使用法 (Snowball Edge クライアント設定済み)

```
snowballEdge describe-auto-update-strategies
```

Example 出力

```
auto-update-strategy {[  
  auto-check:true,  
  auto-check-frequency: "0 0 * * FRI", // CRON Expression String, Every Friday at  
  midnight  
  auto-download:true,  
  auto-download-frequency: "0 0 * * SAT", // CRON Expression String, Every Saturday at  
  midnight  
  auto-install:true,  
  auto-install-frequency: "0 13 * * Sun", // CRON Expression String, Every Saturday at  
  midnight  
  auto-reboot: false;  
]}
```

認証情報の取得

snowballEdge list-access-keys および snowballEdge get-secret-access-key コマンドを使用して、AWS アカウント Snowball Edge 上の の管理者ユーザーの認証情報を取得できま

す。これらの認証情報を使用して、AWS Identity and Access Management (IAM ユーザー) とロールを作成し、AWS CLI または AWS SDK で を使用するときにはリクエストを認証できます。これらの認証情報は、Snowball Edge の個々のジョブにのみ関連付けられ、デバイスまたはデバイスのクラスターでのみ使用できます。デバイスには、AWS クラウドの IAM アクセス許可はありません。

Note

Snowball Edge AWS CLI で を使用している場合は、CLI を設定するときにこれらの認証情報を使用する必要があります。の認証情報の設定については AWS CLI、「ユーザーガイド」の「[AWS CLI の設定](#) AWS Command Line Interface 」を参照してください。

使用法 (Snowball Edge クライアント設定済み)

```
snowballEdge list-access-keys
```

Example 出力

```
{
  "AccessKeyIds" : [ "AKIAIOSFODNN7EXAMPLE" ]
}
```

使用法 (Snowball Edge クライアント設定済み)

```
snowballEdge get-secret-access-key --access-key-id Access Key
```

Example 出力

```
[snowballEdge]
aws_access_key_id = AKIAIOSFODNN7EXAMPLE
aws_secret_access_key = wJalrXUtnFEMI/K7MDENG/bPxrFiCYEXAMPLEKEY
```

Snowball Edge でのサービスの開始

Snowball Edge デバイスは Amazon S3 に加えて、複数のサービスをサポートします。これには、コンピューティングインスタンス、ファイルインターフェイス、およびが含まれます AWS IoT Greengrass。デフォルトでは、Amazon S3 と Amazon EC2 は常にオンになっており、Snowball Edge クライアントで停止または再開はできません。ただし、ファイルインターフェイスとは

snowballEdge start-service コマンドで開始 AWS IoT Greengrass できます。各サービスのサービス ID を取得するには、snowballEdge list-services コマンドを使用できます。

このコマンドを実行する前に、1つの仮想ネットワークインターフェイスを作成して、開始するサービスに結合します。詳細については、「[仮想ネットワークインターフェイスの作成](#)」を参照してください。

使用法 (Snowball Edge クライアント設定済み)

```
snowballEdge start-service --service-id service_id --virtual-network-interface-arns virtual-network-interface-arn
```

Example 出力

```
Starting the AWS service on your Snowball Edge. You can determine the status of the AWS service using the describe-service command.
```

Snowball Edge でのサービスの停止

Snowball Edge で実行中のサービスを停止するには、snowballEdge stop-service コマンドを使用します。

Amazon S3 アダプター、Amazon EC2 AWS STS、IAM サービスは停止できません。

Warning

バッファされた残りのデータがデバイスに書き込まれる前にファイルインターフェイスが停止すると、データ損失が発生する可能性があります。ファイルインターフェイスの詳細については、「[NFS インターフェイスの管理](#)」を参照してください。

Note

Snow Family デバイスの Amazon S3 互換ストレージサービスを停止すると、デバイスまたはクラスターの S3 バケットに保存されているデータにアクセスできなくなります。Snow Family デバイスの Amazon S3 互換ストレージを再起動すると、アクセスが回復します。Snow Family デバイスの Amazon S3 互換ストレージが有効になっているデバイスでは、Snowball Edge デバイスの電源を入れた後にサービスを開始することをお勧めします。本ガイドの「[Snowball Edge のセットアップ](#)」を参照してください。

使用法 (Snowball Edge クライアント設定済み)

```
snowballEdge stop-service --service-id service_id
```

Example 出力

```
Stopping the AWS service on your Snowball Edge. You can determine the status of the AWS service using the describe-service command.
```

NFS の開始とアクセスの制限

⚠ Important

Amazon Elastic Block Store (Amazon EBS) を使用する場合は、NFS サービスを開始しないでください。NFS を初めて起動すると、すべてのストレージが NFS に割り当てられます。NFS サービスが停止している場合でも、NFS ストレージを Amazon EBS に再割り当てすることはできません。

ℹ Note

デバイスによって公開されている NFS 共有のマウントを許可される IP 範囲に CIDR ブロックを提供できます。例えば、10.0.0.0/16。許可された CIDR ブロックを指定しない場合、すべてのマウントリクエストが拒否されます。

NFS 経由で転送されるデータは、転送中に暗号化されないことに注意してください。

CIDR ブロックによって許可されるホストを除き、Snowcone は NFS 共有の認証や認可のメカニズムを提供しません。

snowballEdge start-service コマンドで NFS を起動します。NFS サービスのサービス ID を取得するには、snowballEdge list-services コマンドを使用します。

このコマンドを実行する前に、1つの仮想ネットワークインターフェイスを作成して、開始するサービスに結合します。詳細については、[仮想ネットワークインターフェイスの作成](#)を参照してください。Amazon S3 バケット内のファイル共有とデータへのアクセスを制限し、現在実施されている制限を確認できます。これを行うには、NFS サービスの開始時に、ファイル共有および S3 バケットにアクセスできる許可されたホストに、CIDR ブロックを割り当てます。

使用法 (Snowball Edge クライアント設定済み)

```
snowballEdge start-service --service-id nfs --virtual-network-interface-arns
arn:aws:snowball-device:::interface/s.ni-12345fgh45678j --service-configuration
AllowedHosts=ip address-1/32,ip address-2/24
```

Example 出力例

```
Starting the service on your Snowball Edge. You can determine the status of the service
using the describe-service command.
```

NFS 実行中に NFS 共有へのアクセスを制限する

NFS 起動後に、Amazon S3 バケット内のファイル共有とデータへのアクセスを制限できます。現在実施されている制限を確認し、バケットごとに異なるアクセス制限をかけることができます。これを行うには、NFS サービスの開始時に、ファイル共有および S3 バケットにアクセスできるホストに、CIDR ブロックを割り当てます。以下にサンプルコマンドを示します。

使用法 (Snowball Edge クライアント設定済み)

```
snowballEdge start-service \  
  --service-id nfs \  
  --virtual-network-interface-arns virtual-network-interface-arn --service-configuration
AllowedHosts=ip-address-1/32,ip-address-1/24
```

現在の制限を確認するには、describe-service コマンドを使用します。

```
snowballEdge describe-service --service-id nfs
```

AWS Snowball Edge ログ

オンプレミスのデータセンターと Snowball Edge 間でデータを転送すると、自動的にログが生成されます。デバイスへのデータ転送中に予期しないエラーが発生した場合は、次のコマンドを使用してログのコピーをローカルサーバーに保存できます。

ログに関連するコマンドは、以下の 3 つです。

- list-logs – JSON 形式でログのリストを返します。このリストには、ログのサイズ (バイト単位)、ログの ARN、ログのサービス ID、およびログのタイプが報告されます。

使用法 (Snowball Edge クライアント設定済み)

```
snowballEdge list-logs
```

Example 出力

```
{
  "Logs" : [ {
    "LogArn" : "arn:aws:snowball-device::log/s3-storage-JIEXAMPLE2f-1234-4953-a7c4-
dfEXAMPLE709",
    "LogType" : "SUPPORT",
    "ServiceId" : "s3",
    "EstimatedSizeBytes" : 53132614
  }, {
    "LogArn" : "arn:aws:snowball-device::log/fileinterface-JIDEXAMPLEf-1234-4953-
a7c4-dfEXAMPLE709",
    "LogType" : "CUSTOMER",
    "ServiceId" : "fileinterface",
    "EstimatedSizeBytes" : 4446
  }
]
```

- `get-log` – Snowball Edge から特定のログのコピーを指定されたパスでサーバーにダウンロードします。CUSTOMERログは .zip形式で保存され、このタイプのログを抽出してその内容を表示できます。SUPPORT ログは暗号化され、AWS Support エンジニアのみが読み取ることができます。ログの名前とパスを指定するオプションがあります。

使用法 (Snowball Edge クライアント設定済み)

```
snowballEdge get-log --log-arn arn:aws:snowball-device::log/fileinterface-
JIDEXAMPLEf-1234-4953-a7c4-dfEXAMPLE709
```

Example 出力

```
Logs are being saved to download/path/snowball-edge-logs-1515EXAMPLE88.bin
```

- `get-support-logs` – Snowball Edge から、SUPPORT タイプのすべてのログのコピーを指定したパスのサービスにダウンロードします。

使用法 (Snowball Edge クライアント設定済み)

Snowball Edge クライアント

```
snowballEdge get-support-logs
```

Example 出力

```
Logs are being saved to download/path/snowball-edge-logs-1515716135711.bin
```

Important

CUSTOMER タイプには、お客様独自のデータに関する機密情報が含まれている可能性があります。個の機密情報を保持するために、これらのログが不要になったら削除することを強くお勧めします。

デバイスステータスの取得

以下の Snowball Edge クライアントコマンドを使用して、Snowball Edge デバイスのステータスと一般的なヘルスを判断できます。

- describe-device

使用法 (Snowball Edge クライアント設定済み)

```
snowballEdge describe-device
```

Example 出力

```
{
  "DeviceId" : "JID-EXAMPLE12345-123-456-7-890",
  "UnlockStatus" : {
    "State" : "UNLOCKED"
  },
  "ActiveNetworkInterface" : {
    "IpAddress" : "192.0.2.0"
  }
}
```

```
},
"PhysicalNetworkInterfaces" : [ {
  "PhysicalNetworkInterfaceId" : "s.ni-EXAMPLEd9ecbf03e3",
  "PhysicalConnectorType" : "QSFP",
  "IpAddressAssignment" : "STATIC",
  "IpAddress" : "0.0.0.0",
  "Netmask" : "0.0.0.0",
  "DefaultGateway" : "192.0.2.1",
  "MacAddress" : "EX:AM:PL:E0:12:34",
  "MtuSize" : "1500"
}, {
  "PhysicalNetworkInterfaceId" : "s.ni-EXAMPLE4c3840068f",
  "PhysicalConnectorType" : "SFP_PLUS",
  "IpAddressAssignment" : "DHCP",
  "IpAddress" : "192.0.2.2",
  "Netmask" : "255.255.255.0",
  "DefaultGateway" : "192.0.2.1",
  "MacAddress" : "EX:AM:PL:E0:56:78",
  "MtuSize" : "5743"
}, {
  "PhysicalNetworkInterfaceId" : "s.ni-EXAMPLE0a3a6499fd",
  "PhysicalConnectorType" : "RJ45",
  "IpAddressAssignment" : "STATIC",
  "IpAddress" : "0.0.0.0",
  "Netmask" : "0.0.0.0",
  "DefaultGateway" : "192.0.2.1",
  "MacAddress" : "EX:AM:PL:E0:90:12",
  "MtuSize" : "1500"
} ],
"DeviceCapacities" : [ {
  "Name" : "HDD Storage",
  "Unit" : "Byte",
  "Total" : 39736350227824,
  "Available" : 39707789471744
}, {
  "Name" : "SSD Storage",
  "Unit" : "Byte",
  "Total" : 6979321856000,
  "Available" : 6884832575488
}, {
  "Name" : "vCPU",
  "Unit" : "Number",
  "Total" : 52,
  "Available" : 49
}
```

```
  }, {
    "Name" : "Memory",
    "Unit" : "Byte",
    "Total" : 223338299392,
    "Available" : 216895848448
  }, {
    "Name" : "GPU",
    "Unit" : "Number",
    "Total" : 0,
    "Available" : 0
  } ],
  "DeviceType" : "EDGE_C"
}
```

- describe-cluster

使用法 (Snowball Edge クライアント設定済み)

```
snowballEdge describe-cluster
```

Example 出力

```
{
  "ClusterId" : "CIDEXAMPLE7-5402-4c19-9feb-7c9EXAMPLEd5",
  "Devices" : [ {
    "DeviceId" : "JIDEXAMPLE2-bc53-4618-a538-917EXAMPLE94",
    "UnlockStatus" : {
      "State" : "UNLOCKED"
    },
    "ActiveNetworkInterface" : {
      "IpAddress" : "192.0.2.0"
    },
    "ClusterAssociation" : {
      "State" : "ASSOCIATED",
      "ClusterId" : "CIDEXAMPLE7-5402-4c19-9feb-7c9EXAMPLEd5"
    },
    "NetworkReachability" : {
      "State" : "REACHABLE"
    }
  } ],
  {
    "DeviceId" : "JIDEXAMPLE2-bc53-4618-a538-917EXAMPLE94",
    "UnlockStatus" : {
      "State" : "UNLOCKED"
    }
  }
}
```

```
    },
    "ActiveNetworkInterface" : {
      "IpAddress" : "192.0.2.1"
    },
    "ClusterAssociation" : {
      "State" : "ASSOCIATED",
      "ClusterId" : "CIDEXAMPLE7-5402-4c19-9feb-7c9EXAMPLEd5"
    },
    "NetworkReachability" : {
      "State" : "REACHABLE"
    }
  }, {
    "DeviceId" : "JIDEXAMPLE2-bc53-4618-a538-917EXAMPLE94",
    "UnlockStatus" : {
      "State" : "UNLOCKED"
    },
    "ActiveNetworkInterface" : {
      "IpAddress" : "192.0.2.2"
    },
    "ClusterAssociation" : {
      "State" : "ASSOCIATED",
      "ClusterId" : "CIDEXAMPLE7-5402-4c19-9feb-7c9EXAMPLEd5"
    },
    "NetworkReachability" : {
      "State" : "REACHABLE"
    }
  }, {
    "DeviceId" : "JIDEXAMPLE2-bc53-4618-a538-917EXAMPLE94",
    "UnlockStatus" : {
      "State" : "UNLOCKED"
    },
    "ActiveNetworkInterface" : {
      "IpAddress" : "192.0.2.3"
    },
    "ClusterAssociation" : {
      "State" : "ASSOCIATED",
      "ClusterId" : "CIDEXAMPLE7-5402-4c19-9feb-7c9EXAMPLEd5"
    },
    "NetworkReachability" : {
      "State" : "REACHABLE"
    }
  }, {
    "DeviceId" : "JIDEXAMPLE2-bc53-4618-a538-917EXAMPLE94",
    "UnlockStatus" : {
```

```
    "State" : "UNLOCKED"
  },
  "ActiveNetworkInterface" : {
    "IpAddress" : "192.0.2.4"
  },
  "ClusterAssociation" : {
    "State" : "ASSOCIATED",
    "ClusterId" : "CIDEXAMPLE7-5402-4c19-9feb-7c9EXAMPLEd5"
  },
  "NetworkReachability" : {
    "State" : "REACHABLE"
  }
} ]
}
```

サービスステータスの取得

`describe-service` コマンドを使用して、Snowball Edge デバイスで実行中のサービスのステータスと全般的なヘルスを判断できます。最初に `list-services` コマンドを実行して、実行中のサービスを確認します。

- `list-services`

使用法 (Snowball Edge クライアント設定済み)

```
snowballEdge list-services
```

Example 出力

```
{
  "ServiceIds" : [ "greengrass", "fileinterface", "s3", "ec2", "s3-snow" ]
}
```

- `describe-service`

このコマンドは、サービスのステータス値が返されます。また、サービスで発生した問題を解決するのに役立つ状態情報も含まれています。これらの状態は次のとおりです。

- **ACTIVE** – このサービスは実行されており、使用できます。
- **ACTIVATING** – このサービスは起動しますが、まだ使用することはできません。

- DEACTIVATING – このサービスはシャットダウン処理の途中です。
- DEGRADED – Snow Family デバイスの Amazon S3 互換ストレージの場合、このステータスはクラスター内の 1 つ以上のディスクまたはデバイスがダウンしていることを示します。Snow Family デバイスの Amazon S3 互換ストレージサービスは中断されずに稼働していますが、データ損失のリスクを最小限に抑えるため、クラスターのクォーラムが失われる前に影響を受けるデバイスを回復または交換する必要があります。本ガイドの「[クラスタリングの概要](#)」を参照してください。
- INACTIVE – このサービスは実行されておらず、使用することはできません。

使用法 (Snowball Edge クライアント設定済み)

```
snowballEdge describe-service --service-id service-id
```

Example 出力

```
{
  "ServiceId" : "s3",
  "Status" : {
    "State" : "ACTIVE"
  },
  "Storage" : {
    "TotalSpaceBytes" : 99608745492480,
    "FreeSpaceBytes" : 99608744468480
  },
  "Endpoints" : [ {
    "Protocol" : "http",
    "Port" : 8080,
    "Host" : "192.0.2.0"
  }, {
    "Protocol" : "https",
    "Port" : 8443,
    "Host" : "192.0.2.0",
    "CertificateAssociation" : {
      "CertificateArn" : "arn:aws:snowball-
device::certificate/6d955EXAMPLEdb71798146EXAMPLE3f0"
    }
  } ]
}
```

Example Snow Family デバイスの Amazon S3 互換ストレージのサービス出力

describe-service コマンドは、service-idパラメータの *s3-snow* 値に対して次の出力を提供します。

```
{
  "ServiceId" : "s3-snow",
  "Autostart" : false,
  "Status" : {
    "State" : "ACTIVE"
  },
  "ServiceCapacities" : [ {
    "Name" : "S3 Storage",
    "Unit" : "Byte",
    "Used" : 640303104,
    "Available" : 219571981512
  } ],
  "Endpoints" : [ {
    "Protocol" : "https",
    "Port" : 443,
    "Host" : "10.0.2.123",
    "CertificateAssociation" : {
      "CertificateArn" : "arn:aws:snowball-device::certificate/
a65ba817f2c5ac9683fc3bc1ae123456"
    },
    "Description" : "s3-snow bucket API endpoint",
    "DeviceId" : "JID6ebd4c50-c3a1-4b16-b32c-b254f9b7f2dc",
    "Status" : {
      "State" : "ACTIVE"
    }
  } ], {
    "Protocol" : "https",
    "Port" : 443,
    "Host" : "10.0.3.202",
    "CertificateAssociation" : {
      "CertificateArn" : "arn:aws:snowball-device::certificate/
a65ba817f2c5ac9683fc3bc1ae123456"
    },
    "Description" : "s3-snow object API endpoint",
    "DeviceId" : "JID6ebd4c50-c3a1-4b16-b32c-b254f9b7f2dc",
    "Status" : {
```

```
    "State" : "ACTIVE"
  }
}, {
  "Protocol" : "https",
  "Port" : 443,
  "Host" : "10.0.3.63",
  "CertificateAssociation" : {
    "CertificateArn" : "arn:aws:snowball-device::certificate/
a65ba817f2c5ac9683fc3bc1ae123456"
  },
  "Description" : "s3-snow bucket API endpoint",
  "DeviceId" : "JID2a1e0deb-38b1-41f8-b904-a396c62da70d",
  "Status" : {
    "State" : "ACTIVE"
  }
}, {
  "Protocol" : "https",
  "Port" : 443,
  "Host" : "10.0.2.243",
  "CertificateAssociation" : {
    "CertificateArn" : "arn:aws:snowball-device::certificate/
a65ba817f2c5ac9683fc3bc1ae123456"
  },
  "Description" : "s3-snow object API endpoint",
  "DeviceId" : "JID2a1e0deb-38b1-41f8-b904-a396c62da70d",
  "Status" : {
    "State" : "ACTIVE"
  }
}, {
  "Protocol" : "https",
  "Port" : 443,
  "Host" : "10.0.2.220",
  "CertificateAssociation" : {
    "CertificateArn" : "arn:aws:snowball-device::certificate/
a65ba817f2c5ac9683fc3bc1ae123456"
  },
  "Description" : "s3-snow bucket API endpoint",
  "DeviceId" : "JIDcc45fa8f-b994-4ada-a821-581bc35d8645",
  "Status" : {
    "State" : "ACTIVE"
  }
}, {
  "Protocol" : "https",
  "Port" : 443,
```

```
"Host" : "10.0.2.55",
"CertificateAssociation" : {
  "CertificateArn" : "arn:aws:snowball-device::certificate/
a65ba817f2c5ac9683fc3bc1ae123456"
},
"Description" : "s3-snow object API endpoint",
"DeviceId" : "JIDcc45fa8f-b994-4ada-a821-581bc35d8645",
"Status" : {
  "State" : "ACTIVE"
}
}, {
  "Protocol" : "https",
  "Port" : 443,
  "Host" : "10.0.3.213",
  "CertificateAssociation" : {
    "CertificateArn" : "arn:aws:snowball-device::certificate/
a65ba817f2c5ac9683fc3bc1ae123456"
  },
  "Description" : "s3-snow bucket API endpoint",
  "DeviceId" : "JID4ec68543-d974-465f-b81d-89832dd502db",
  "Status" : {
    "State" : "ACTIVE"
  }
}, {
  "Protocol" : "https",
  "Port" : 443,
  "Host" : "10.0.3.144",
  "CertificateAssociation" : {
    "CertificateArn" : "arn:aws:snowball-device::certificate/
a65ba817f2c5ac9683fc3bc1ae123456"
  },
  "Description" : "s3-snow object API endpoint",
  "DeviceId" : "JID4ec68543-d974-465f-b81d-89832dd502db",
  "Status" : {
    "State" : "ACTIVE"
  }
}, {
  "Protocol" : "https",
  "Port" : 443,
  "Host" : "10.0.2.143",
  "CertificateAssociation" : {
    "CertificateArn" : "arn:aws:snowball-device::certificate/
a65ba817f2c5ac9683fc3bc1ae123456"
  },
  },
```

```
"Description" : "s3-snow bucket API endpoint",
"DeviceId" : "JID6331b8b5-6c63-4e01-b3ca-eab48b5628d2",
"Status" : {
  "State" : "ACTIVE"
}
}, {
  "Protocol" : "https",
  "Port" : 443,
  "Host" : "10.0.3.224",
  "CertificateAssociation" : {
    "CertificateArn" : "arn:aws:snowball-device:::certificate/
a65ba817f2c5ac9683fc3bc1ae123456"
  },
  "Description" : "s3-snow object API endpoint",
  "DeviceId" : "JID6331b8b5-6c63-4e01-b3ca-eab48b5628d2",
  "Status" : {
    "State" : "ACTIVE"
  }
} ]
}
```

クラスターからノードを削除する

disassociate-device コマンドは、Snowball Edge クラスターからノードを削除します。異常なノードを置き換える場合は、このコマンドを使用します。クラスターの詳細については、本ガイドの「[クラスタリングの概要](#)」を参照してください。

Important

disassociate-device コマンドは、異常なノードを削除する場合にのみ使用します。正常なノードを削除しようとする、このコマンドは失敗し、エラーが返されます。

このコマンドを使用して、ノードの電源が誤ってオフになり、ネットワークから切断され、残りのクラスターが一時的に利用できないノードを削除しないでください。このコマンドで削除されたノードはいずれのクラスターにも追加できないため、AWSに返す必要があります。

誤って電源を切ったノードや、ネットワークから切断したノードについては、ノードを再度、電源およびネットワークに接続し、`associate-device` コマンドを使用します。電源が入っていて正常である場合、`disassociate-device` コマンドを使用してノードを切断することはできません。

使用法 (Snowball Edge クライアント設定済み)

```
snowballEdge disassociate-device --device-id Job ID for the Device
```

Example 出力

```
Disassociating your Snowball Edge device from the cluster. Your Snowball Edge device will be disassociated from the cluster when it is in the "DISASSOCIATED" state. You can use the describe-cluster command to determine the state of your cluster.
```

クラスターへのノードの追加

`associate-device` コマンドは、Snowball Edge デバイスのクラスターにノードを追加します。ノードの電源を切ると、ロック解除からロック状態に戻ります。そのノードのロックを解除するには、このコマンドを使用できます。このコマンドを使用して、使用できないノードを、代替品として注文した新しいノードと置き換えます。クラスターの詳細については、本ガイドの「[クラスタリングの概要](#)」を参照してください。

使用法 (Snowball Edge クライアント設定済み)

```
snowballEdge associate-device --device-ip-address IP Address
```

Example 出力

```
Associating your Snowball Edge device with the cluster. Your Snowball Edge device will be associated with the cluster when it is in the ASSOCIATED state. You can use the describe-cluster command to determine the state of your cluster.
```

デバイス用のタグの作成

デバイス上の指定されたタグを追加または上書きします。最大 50 個のタグを作成できます。タグは、キーと値のペアから構成されます。値はオプションです。

Note

タグに機密データを含めないようにしてください。

使用法 (Snowball Edge クライアント設定済み)

```
snowballEdge create-tags --tag Key=Name,Value=user-test --tag Key=Stage,Value=beta
```

詳細については、describe-tags コマンドを実行します。

Example 出力

```
Tag(s) [Key=Name,Value=test, Key=Stage,Value=beta] created.
```

デバイスからのタグの削除

delete-tags コマンドは、指定したタグを Snowball Edge デバイスから削除します。

使用法 (Snowball Edge クライアント設定済み)

```
snowballEdge delete-tags --tag Key=Stage,Value=beta  
Tag(s) [Key=Stage,Value=beta] deleted.
```

詳細については、describe-tags コマンドを実行します。

Note

複数のタグを同時に削除する場合、次のように複数のキーと値のペアを指定できます。
delete-tags --tag Key=Name,Value=test --tag Key=Stage,Value=Beta
タグ値を指定せずにタグキーを指定すると、その値に関係なく、このキーを持つタグはすべて削除されます。タグ値として空の文字列を持つタグキーを指定すると、値として空の文字列を持つタグのみが削除されます。

デバイスのタグの説明

describe-tags コマンドは、Snowball Edge デバイス上のタグを記述します。

使用法 (Snowball Edge クライアント設定済み)

```
snowballEdge describe-tags
```

詳細については、`describe-tags` コマンドを実行します。

Example 出力

```
{
  "Tags" : [ {
    "Key" : "Name",
    "Value" : "user-test"
  }, {
    "Key" : "Stage",
    "Value" : "beta"
  } ]
}
```

ダイレクトネットワークインターフェイスの作成

`create-direct-network-interface` コマンドを使用して、Snow Family デバイスにダイレクトネットワークインターフェイスを作成します。ダイレクトネットワークインターフェイス (DNI) は AMI インスタンスに関連付ける必要があります。vlan パラメータを使用して、仮想ローカルエリアネットワーク (VLAN) ID をインターフェイスに割り当て、インターフェイスからのすべてのトラフィックにその VLAN ID をタグ付けできます。また、mac パラメータを使用して DNI にメディアアクセスコントロール (MAC) アドレスを割り当てることもできます。mac パラメータと値を指定しない場合、MAC アドレスが自動的に割り当てられます。

`describe-device` コマンドを使用して、物理ネットワークインターフェイス ID を取得できます。詳細については、「[デバイスステータスの取得](#)」を参照してください。

```
snowballEdge create-direct-network-interface --instance-id AMI-instance-id --physical-network-interface-id physical-network-interface-id --vlan vlan-id --mac MAC-address
```

Example `create-direct-network-interface` コマンド出力の

```
{
  "DirectNetworkInterface" : {
```



```
    "DirectNetworkInterfaceArn" : "arn:aws:snowball-device:::interface/s.ni-
x8a3b6k1e9n4r2s7o",
    "PhysicalNetworkInterfaceId" : "s.ni-p5d2q8r3s9t4u7v1w",
    "InstanceId" : "s.i-g9h2j4k6l8m1n3p5q",
    "Driver" : "mlx5 core",
    "MacAddress" : "1A:2B:3C:4D:5E:6F",
    "MtuSize": "1500"
  }
}
```

ダイレクトネットワークインターフェイスに関する情報の取得

`describe-direct-network-interface` コマンドを使用して、Snow Family デバイスのダイレクトネットワークインターフェイスに関する情報を表示します。

```
snowballEdge describe-direct-network-interfaces --endpoint https://snow-device-ip-
address --manifest-file path/to/manifest/file.bin --unlock-code unlock-code
```

Example `describe-direct-network-interfaces` コマンド出力の

```
{
  "DirectNetworkInterface" : {
    "DirectNetworkInterfaceArn" : "arn:aws:snowball-device:::interface/s.ni-
x8a3b6k1e9n4r2s7o",
    "PhysicalNetworkInterfaceId" : "s.ni-p5d2q8r3s9t4u7v1w",
    "InstanceId" : "s.i-g9h2j4k6l8m1n3p5q",
    "Driver" : "mlx5 core",
    "MacAddress" : "1A:2B:3C:4D:5E:6F",
    "MtuSize": "1500"
  }
}
```

ダイレクトネットワークインターフェイスの更新

`update-direct-network-interface` を使用して、ダイレクトネットワークインターフェイス (DNI) のプロパティを変更します。Amazon EC2-compatible を変更すると、インターフェイスはデタッチされます。DNI が使用している物理インターフェイスの最大送信単位 (MTU) を変更するに

は、`update-mtu-size` コマンドを使用します。詳細については、「[MTU サイズの更新](#)」を参照してください。

```
snowballEdge update-direct-network-interface --direct-network-interface-arn directNetworkInterfaceArn //
--endpoint https://snow-device-ip-address [--mac macAddress] //
--manifest-file path/to/manifest/file.bin --unlock-code unlock-code //
[--vlan vlanId] [--attach-instance-id instanceId | --detach]
```

Example `update-direct-network-interface` コマンド出力の

```
{
  "DirectNetworkInterface" : {
    "DirectNetworkInterfaceArn" : "arn:aws:snowball-device:::interface/s.ni-x8a3b6k1e9n4r2s7o",
    "PhysicalNetworkInterfaceId" : "s.ni-p5d2q8r3s9t4u7v1w",
    "InstanceId" : "s.i-g9h2j4k6l8m1n3p5q",
    "Driver" : "mlx5 core",
    "MacAddress" : "2A:3B:5C:5D:6E:7F",
    "MtuSize": "1500"
  }
}
```

ダイレクトネットワークインターフェイスの削除

`delete-direct-network-interface` コマンドを使用して、ダイレクトネットワークインターフェイス (DNI) を削除します。Amazon EC2-compatible コンピューティングインスタンスに関連付けられた DNI を削除するには、まず `update-direct-network-interface` コマンドの `detach` パラメータを使用してインスタンスから DNI をデタッチします。詳細については、「[ダイレクトネットワークインターフェイスの更新](#)」を参照してください。

```
snowballEdge delete-direct-network-interface --direct-network-interface-arn directNetworkInterfaceArn //
--endpoint https://snow-device-ip-address --manifest-file path/to/manifest/file.bin //
[--profile profile] --unlock-code unlock-code
```

Example `delete-direct-network-interface` コマンド出力の

The direct network interface has been deleted from your Snowball Edge. You can determine the direct network interfaces available on your Snowball Edge using the `describe-direct-network-interfaces` command.

仮想ネットワークインターフェイス (VNI) を作成する

`create-virtual-network-interface` コマンドを使用して、Snowball Edge デバイスに仮想ネットワークインターフェイスを作成します。`describe-device` コマンドを使用して、物理ネットワークインターフェイス ID を取得できます。詳細については、「[デバイスステータスの取得](#)」を参照してください。

Note

`static-ip-address-configuration` パラメータは、`ip-address-assignment` パラメータに `STATIC` 値を使用する場合にのみ有効です。

```
snowballEdge create-virtual-network-interface --endpoint https://ip-address-of-snow-device --manifest-file /path/to/manifest/file.bin --unlock-code unlock-code --ip-address-assignment DHCP or STATIC --physical-network-interface-id [physical network interface id] --static-ip-address-configuration IpAddress=IP-address,NetMask=netmask
```

Example `create-virtual-network-interface` コマンド出力の

```
{
  "VirtualNetworkInterface": {
    "VirtualNetworkInterfaceArn": "arn:aws:snowball-device:::interface/s.ni-8EXAMPLE8EXAMPLEf",
    "PhysicalNetworkInterfaceId": "s.ni-8EXAMPLEeEXAMPLEd",
    "IpAddressAssignment": "DHCP",
    "IpAddress": "192.0.2.0",
    "Netmask": "255.255.255.0",
    "DefaultGateway": "192.0.2.1",
    "MacAddress": "EX:AM:PL:E1:23:45",
```

```
    "MtuSize" : "1500"
  }
}
```

仮想ネットワークインターフェイスに関する情報の取得

`describe-virtual-network-interface` コマンドを使用して、Snow Family デバイスの仮想ネットワークインターフェイスに関する情報を表示します。

```
snowballEdge describe-direct-network-interfaces --endpoint https://ip-address-of-snow-device --manifest-file path/to/manifest/file.bin --unlock-code unlock-code
```

Example `describe-virtual-network-interfaces` コマンド出力の

```
{
  "VirtualNetworkInterface": {
    "VirtualNetworkInterfaceArn": "arn:aws:snowball-device:::interface/s.ni-8EXAMPLE8EXAMPLEf",
    "PhysicalNetworkInterfaceId": "s.ni-8EXAMPLEaEXAMPLEd",
    "IpAddressAssignment": "DHCP",
    "IpAddress": "192.0.2.0",
    "Netmask": "255.255.255.0",
    "DefaultGateway": "192.0.2.1",
    "MacAddress": "EX:AM:PL:E1:23:45",
    "MtuSize" : "1500"
  }
}
```

仮想ネットワークインターフェイスの更新

`update-virtual-network-interface` コマンドを使用して、Snow Family デバイスの仮想ネットワークインターフェイスを更新します。DNI が使用している物理インターフェイスの最大送信単位 (MTU) を変更するには、`update-mtu-size` コマンドを使用します。詳細については、「[MTU サイズの更新](#)」を参照してください。

```
snowballEdge update-virtual-network-interface --direct-network-interface-arn directNetworkInterfaceArn --endpoint https://ip-address-of-snow-device // --unlock-code unlock-code [--mac macAddress] --manifest-file path/to/manifest/file.bin // [--vlan vlanId] [--attach-instance-id instanceId | --detach]
```

Example `update-virtual-network-interface` コマンド出力の

```
{
  "VirtualNetworkInterface": {
    "VirtualNetworkInterfaceArn": "arn:aws:snowball-device:::interface/s.ni-8EXAMPLE8EXAMPLEf",
    "PhysicalNetworkInterfaceId": "s.ni-8EXAMPLEaEXAMPLEd",
    "IpAddressAssignment": "DHCP",
    "IpAddress": "192.0.2.9",
    "Netmask": "255.255.255.0",
    "DefaultGateway": "192.0.2.1",
    "MacAddress": "EX:AM:PL:E1:23:45",
    "MtuSize" : "1500"
  }
}
```

仮想ネットワークインターフェイスの削除

`delete-direct-network-interface` コマンドを使用して、仮想ネットワークインターフェイス (VNI) を削除します。

```
snowballEdge delete-virtual-network-interface --virtual-network-interface-arn virtual-network-interface-ARN --endpoint https://endpoint // --manifest-file path/to/manifest/file.bin] [--profile profile] --unlock-code unlock-code]
```

Example `delte-direct-network-interface` コマンド出力の

The virtual network interface has been deleted from your Snowball Edge. You can determine the virtual network interfaces available on your Snowball Edge using the `describe-virtual-network-interfaces` command.

機能ステータスのチェック

デバイスで使用可能な機能のステータスを一覧表示するには、`describe-features` コマンドを使用します。

`RemoteManagementState` は、Snow Device Management のステータスを表示し、次のいずれかの状態を返します。

- `INSTALLED_ONLY` – 機能はインストールされていますが、有効になっていません。
- `INSTALLED_AUTOSTART` – この機能が有効で、デバイスは電源が入っている AWS リージョン ときにへの接続を試みます。
- `NOT_INSTALLED` – デバイスはこの機能をサポートしないか、起動前に既にフィールドに入っていました。

使用法 (Snowball Edge クライアント設定済み)

```
snowballEdge describe-features \  
  --manifest-file manifest.bin path \  
  --unlock-code unlock-code \  
  --endpoint https://device-local-ip:9091
```

出力例

```
{  
  "RemoteManagementState" : String  
}
```

タイムサーバーの設定

外部ネットワークタイムプロトコル (NTP) サーバーを設定できます。NTP CLI コマンドは、デバイスがロック状態とロック解除状態の両方の場合に使用できます。マニフェストと解除コードが必要です。これらは、`snowballEdge configure` コマンド、または `--manifest-file` および `--unlock-code` オプションを使用して設定できます。AWS Snowcone Edge との両方で `snowballEdge CLI` を使用できることに注意してください AWS Snowcone。

セキュリティで保護された NTP タイムサーバーは、お客様が用意する必要があります。デバイスを接続する NTP タイムサーバーを設定するには、`update-time-servers` CLI コマンドを使用します。

Note

`update-time-servers` コマンドは、以前の NTP タイムサーバーの設定を上書きします。

サポートされている NTP デバイスタイプとソフトウェアバージョン

NTP は、バージョン 2 のどのストレージデバイスやコンピューティングデバイスでも使用できません。ただし、ソフトウェアバージョン 77 以降の Snowball Edge バージョン 3 ストレージおよびコンピューティングデバイスタイプは NTP をサポートします。NTP が有効になっているかどうかを確認するには、Snowball Edge CLI コマンド `describe-time-sources` を使用します。

使用方法

```
snowballEdge update-time-servers time.google.com
```

Example 出力例

```
Updating time servers now.
```

タイムソースのチェック

デバイスが現在接続されている NTP タイムソースを確認するには、`describe-time-sources` Snowball Edge CLI コマンドを使用します。

使用方法

```
snowballEdge describe-time-sources
```

Example 出力例

```
{
  "Sources" : [ {
    "Address" : "172.31.2.71",
    "State" : "LOST",
    "Type" : "PEER",
```

```
"Stratum" : 10
}, {
  "Address" : "172.31.3.203",
  "State" : "LOST",
  "Type" : "PEER",
  "Stratum" : 10
}, {
  "Address" : "172.31.0.178",
  "State" : "LOST",
  "Type" : "PEER",
  "Stratum" : 10
}, {
  "Address" : "172.31.3.178",
  "State" : "LOST",
  "Type" : "PEER",
  "Stratum" : 10
}, {
  "Address" : "216.239.35.12",
  "State" : "CURRENT",
  "Type" : "SERVER",
  "Stratum" : 1
} ]
}
```

`describe-time-sources` コマンドは、タイムソースの状態のリストを返します。ソース状態は、毎回、Address、State、Type、Stratum フィールドを含みます。これらのフィールドの意味は以下のとおりです。

- Address – タイムソースの DNS 名/ IP アドレス。
- State – デバイスとそのタイムソース間の現在の接続ステータス。5 つの状態が想定されます。
 - CURRENT – タイムソースは現在、時刻の同期に使用されている。
 - COMBINED – タイムソースは現在のソースと結合されている。
 - EXCLUDED – タイムソースが結合アルゴリズムによって除外された。
 - LOST – タイムソースとの接続が切断された。
 - UNACCEPTABLE – 結合アルゴリズムが偽ティッカーであるか、ばらつきが多すぎるとみなされた無効なタイムソース。
- Type – NTP タイムソースは、サーバーまたはピアのいずれかになります。サーバーは、`update-time-servers` コマンドにより設定できます。ピアは、クラスター内の他の Snowball Edge デバイスのみになり、クラスターが関連付けられると自動的に設定されます。

- Stratum – このフィールドには、ソースの階層が表示されます。階層 1 は、ローカルでアタッチされたリファレンスクロックを持つソースを示します。階層 1 のソースに同期されたソースは、階層 2 にあります。階層 2 のソースに同期されたソースは、階層 3 などにあります。

NTP タイムソースは、サーバーまたはピアのいずれかになります。サーバーは、`update-time-servers` コマンドを使用するユーザーによって設定されます。一方、ピアはクラスター内の他の Snowball Edge デバイスのみとなります。出力例では、`describe-time-sources` は、5 のクラスター内にある Snowball Edge で呼び出されます。出力には、4 つのピアと 1 台のサーバーが含まれます。ピアの階層は 10 で、サーバーの階層は 1 であるため、サーバーが現在のタイムソースとして選択されます。

MTU サイズの更新

`update-mtu-size` コマンドを使用して、Snow Family デバイスの物理インターフェイスの最大送信単位 (MTU) のバイト単位のサイズを変更します。この物理ネットワークインターフェイスに関連付けられているすべての仮想ネットワークインターフェイスとダイレクトネットワークインターフェイスは、同じ MTU サイズで設定されます。

Note

最小 MTU サイズは 1500 バイト、最大サイズは 9216 バイトです。

`describe-device` コマンドを使用して、物理ネットワークインターフェイス IDs とそれらのインターフェイスの現在の MTU サイズを取得できます。詳細については、「[デバイスステータスの取得](#)」を参照してください。

`describe-direct-network-interface` および `describe-virtual-network-interface` コマンドを使用して、それらのインターフェイスの現在の MTU サイズを取得できます。詳細については、「[ダイレクトネットワークインターフェイスに関する情報の取得](#)」および「[仮想ネットワークインターフェイスに関する情報の取得](#)」を参照してください。

使用方法

```
snowballEdge update-mtu-size --physical-network-interface-id physical-network-interface-id --mtu-size size-in-bytes
```

Example `update-mtu-size` の出力

```
{
  "PhysicalNetworkInterface": {
    "PhysicalNetworkInterfaceId": "s.ni-8c1f891d7f5b87cfe",
    "PhysicalConnectorType": "SFP_PLUS",
    "IpAddressAssignment": "DHCP",
    "IpAddress": "192.0.2.0",
    "Netmask": "255.255.255.0",
    "DefaultGateway": "192.0.2.255",
    "MacAddress": "8A:2r:5G:9p:6Q:4s",
    "MtuSize": "5743"
  }
}
```

データ移行に Amazon S3 アダプターを使用してファイルを転送する

以下は、Amazon S3 アダプターの概要です。このアダプターを使用して、Amazon S3 REST API アクションを使用して、AWS Snowball Edge デバイス上に既にある S3 バケットとの間でプログラムでデータを転送できます。Amazon S3 この Amazon S3 REST API サポートは、アクションのサブセットのみに限定されています。このアクションのサブセットを AWS SDKs の 1 つで使用して、プログラムでデータを転送できます。また、Amazon S3 でサポートされる AWS Command Line Interface (AWS CLI) コマンドのサブセットを使用して、プログラムでデータを転送できます。

ソリューションが AWS SDK for Java バージョン 1.11.0 以降を使用している場合は、次のを使用する必要があります `S3ClientOptions`。

- `disableChunkedEncoding()` – インターフェイスでは、チャンクエンコーディングがサポートされないことを示します。
- `setPathStyleAccess(true)` – すべてのリクエストに対して、パススタイルのアクセスを使用するようにインターフェイスを設定します。

詳細については、Amazon SDK for [Java の「Class S3ClientOptions.Builder」](#) を参照してください。
AppStream

⚠ Important

デバイス上のローカルバケットへのデータの読み取りと書き込みには、一度に1つの方法のみを使用することをお勧めします AWS Snowball Edge。ファイルインターフェイスと Amazon S3 アダプターの両方を同じバケットで同時に使用すると、読み取り/書き込みの競合が発生する可能性があります。

「[AWS Snowball Edge クォータ](#)」で、制限の詳細を説明しています。

AWS サービスが Snowball Edge で正しく動作するには、サービスのポートを許可する必要があります。詳細については、「[エッジデバイスで AWS サービス AWS Snowball を使用するために必要なポート](#)」を参照してください。

トピック

- [Amazon S3 AWS CLI アダプターで使用するバージョン 1.16.14 のダウンロードとインストール Amazon S3](#)
- [Snowball Edge デバイスで AWS CLI および API オペレーションを使用する](#)
- [ローカルの Amazon S3 認証情報の取得と使用](#)
- [Amazon S3 アダプターではサポートされていない Amazon S3 の機能](#)
- [小さなファイルのバッチ処理](#)
- [サポートされている AWS CLI コマンド](#)
- [サポートされている REST API アクション](#)

Amazon S3 AWS CLI アダプターで使用するバージョン 1.16.14 のダウンロードとインストール Amazon S3

現在、Snowball Edge デバイスで Amazon S3 アダプターの使用がサポートされているのは、AWS CLI 1.16.14 以前のバージョンのみです。の新しいバージョン AWS CLI は、Amazon S3 S3 アダプターと互換性がありません。

📌 Note

Snow Family デバイスの Amazon S3 互換ストレージを使用している場合は、AWS CLIの最新バージョンを使用できます。最新バージョンをダウンロードして使用するには、「[AWS Command Line Interface ユーザーガイド](#)」を参照してください。

Linux オペレーティングシステム AWS CLI をインストールする

この連鎖コマンドを実行します。

```
curl "https://s3.amazonaws.com/aws-cli/awscli-bundle-1.16.14.zip" -o "awscli-bundle.zip";unzip awscli-bundle.zip;sudo ./awscli-bundle/install -i /usr/local/aws -b /usr/local/bin/aws;/usr/local/bin/aws --version;
```

Windows オペレーティングシステム AWS CLI をインストールする

お使いのオペレーティングシステム用のインストーラファイルをダウンロードして実行します。

- [32 ビット](#)
- [64 ビット](#)

Snowball Edge デバイスで AWS CLI および API オペレーションを使用する

AWS CLI または API オペレーションを使用して Snowball Edge で IAM、Amazon S3、および Amazon EC2 コマンドを発行する場合は、リージョンを「」と指定する必要がありますsnow。これを行うには、次の例のように、コマンド自体で AWS configure または を使用します。

```
aws configure --profile abc
AWS Access Key ID [None]: AKIAIOSFODNN7EXAMPLE
AWS Secret Access Key [None]: 1234567
Default region name [None]: snow
Default output format [None]: json
```

または

```
aws s3 ls --profile snowballEdge --endpoint http://192.0.2.0:8080 --region snow
```

AWS Snowball の Amazon S3 API インターフェイスを使用した認可

Amazon S3 アダプターを使用する場合、すべてのインタラクションはデフォルトで Signature Version AWS 4 アルゴリズムで署名されます。この認可は、ソースからインターフェイスに移動す

るデータを検証するためにのみ使用されます。すべての暗号化と復号はデバイス上で行います。暗号化されていないデータは、デバイスに保存されません。

インターフェイスを使用する際は、次の点に注意が必要です。

- AWS Snowball Edge デバイスへのリクエストに署名するためにローカルの Amazon S3 認証情報を取得するには、`snowballEdge list-access-keys` および `snowballEdge get-secret-access-keys` Snowball Edge クライアントコマンドを実行します。詳細については、「[Snowball Edge クライアントコマンドの使用](#)」を参照してください。これらのローカル Amazon S3 認証情報には、アクセスキーとシークレットキーのキーペアが含まれます。これらのキーは、ジョブに関連付けられたデバイスでのみ有効です。AWS Identity and Access Management (IAM) 対応がないため、AWS クラウド では使用できません。
- 暗号化キーは、使用する AWS 認証情報によって変更されません。署名バージョン 4 のアルゴリズムを使用した署名は、ソースからインターフェイスに移動するデータを検証するためだけに使用されます。このため、この署名は Snowball でデータを暗号化するために使用される暗号化キーには関係しません。

ローカルの Amazon S3 認証情報の取得と使用

Snowball Edge とのやり取りはすべて、署名バージョン AWS 4 アルゴリズムで署名されます。アルゴリズムの詳細については、「AWS 全般のリファレンス」の「[Signature Version 4 Signing Process](#)」を参照してください。

Snowball Edge クライアントエッジデバイスに対するリクエストに署名するローカル Amazon S3 認証情報は、`snowballEdge list-access-keys` および `snowballEdge get-secret-access-key` Snowball Edge クライアント情報を実行して取得できます。[認証情報の取得](#) を参照してください。これらのローカル Amazon S3 認証情報には、アクセスキー ID とシークレットキーのキーペアが含まれます。これらの認証情報は、ジョブに関連付けられたデバイスでのみ有効です。これらは IAM 対応がないため、AWS クラウド では使用できません。

これらの認証情報は、サーバーの AWS 認証情報ファイルに追加できます。通常、デフォルトの認証情報プロファイルファイルは `~/.aws/credentials` にありますが、この場所はプラットフォームごとに異なる場合があります。このファイルは、多くの AWS SDKs とによって共有されます AWS CLI。ローカル認証情報は、次の例のようにプロファイル名を付けて保存できます。

```
[snowballEdge]
aws_access_key_id = AKIAIOSFODNN7EXAMPLE
aws_secret_access_key = wJalrXUtnFEMI/K7MDENG/bPxrFiCYEXAMPLEKEY
```

S3 アダプターを AWS CLI エンドポイントとして指定する

を使用して AWS Snowball Edge デバイスにコマンド AWS CLI を発行する場合、エンドポイントが Amazon S3 アダプターであることを指定します。次に示すように、HTTPS エンドポイント、またはセキュリティで保護されていない HTTP エンドポイントの使用を選択できます。

セキュリティで保護された HTTPS エンドポイント

```
aws s3 ls --profile snowballEdge --endpoint https://192.0.2.0:8443 --ca-bundle path/to/certificate
```

セキュリティで保護されていない HTTP エンドポイント

```
aws s3 ls --profile snowballEdge --endpoint http://192.0.2.0:8080
```

8443の HTTPS エンドポイントを使用すると、データはサーバーから Snowball Edge に安全に転送されます。この暗号化は、新しい IP アドレスを取得する際に Snowball Edge によって生成される証明書を使用して行われます。証明書を取得したら、ローカルの `ca-bundle.pem` ファイルに保存できます。その後、以下で説明するように、証明書へのパスを含めるように AWS CLI プロファイルを設定できます。

証明書をインターフェイスエンドポイントに関連付けるには

1. Snowball Edge を電源とネットワークに接続し、電源を入れます。
2. デバイスの起動が終了したら、ローカルネットワークでの IP アドレスを書き留めておきます。
3. ネットワーク上のターミナルから Snowball Edge に ping を送信できることを確認します。
4. ターミナルで `snowballEdge get-certificate` コマンドを実行します。このコマンドの詳細については、「[公開鍵証明書の管理](#)」を参照してください。
5. `snowballEdge get-certificate` コマンドの出力をファイルに保存します (例: `ca-bundle.pem`)。
6. ターミナルから次のコマンドを実行します。

```
aws configure set profile.snowballEdge.ca_bundle /path/to/ca-bundle.pem
```

この手順を完了したら、次の例のように、これらのローカル認証情報、証明書、および指定されたエンドポイントを使用して CLI コマンドを実行できます。

```
aws s3 ls --profile snowballEdge --endpoint https://192.0.2.0:8443
```

Amazon S3 アダプターではサポートされていない Amazon S3 の機能

Amazon S3 アダプターを使用して、Amazon S3 API アクションにより、Snowball Edge との間でデータをプログラムで転送できます。ただし、Amazon S3 アダプターを使用する場合、すべての Amazon S3 転送機能および API アクションが、Snowball Edge デバイスで使用できるわけではありません。例えば、以下の機能とアクションは、Snowball Edge での使用がサポートされません。

- [TransferManager](#) — このユーティリティは、SDK for Java を使用してローカル環境から Amazon S3 にファイルを転送します。代わりに、インターフェイスを使用して、サポートされる API アクションまたは AWS CLI コマンドを使用することを検討してください。
- [GET Bucket \(オブジェクトのリスト化\) バージョン 2](#) – この GET アクションを実装すると、バケット内の一部またはすべて (最大 1,000) のオブジェクトが返されます。[GET Bucket \(List Objects\) バージョン 1](#) アクションまたは `ls` AWS CLI コマンドを使用することを検討してください。
- [ListBuckets](#) – オブジェクトエンドポイント ListBuckets を持つ はサポートされていません。次のコマンドは、Snow Family デバイスの Amazon S3 互換ストレージでは機能しません。

```
aws s3 ls --endpoint https://192.0.2.0 --profile profile
```

小さなファイルのバッチ処理

各コピーオペレーションには、暗号化のために多少のオーバーヘッドがあります。小さなファイルを AWS Snowball Edge デバイスに転送するプロセスを高速化するために、それらを 1 つのアーカイブにまとめてバッチ処理できます。ファイルをバッチ処理する際に、サポートされるアーカイブ形式のいずれかでファイルがバッチ処理済みの場合は、Amazon S3 へのインポート時に自動抽出されます。

通常は、1 MB 以下のファイルがバッチ処理に含まれている必要があります。バッチ処理に含めるファイル数にハード制限はありませんが、バッチ処理に 10000 ファイルを上限とすることが推奨されます。バッチに 10 万以上のファイルがあると、デバイスに戻った後で Amazon S3 にインポートされるファイルの速度に影響します。各バッチの合計サイズは 100 GB 以下にすることをお勧めします。

ファイルのバッチ処理は手動によって管理するプロセスです。ファイルをバッチ処理したら、コマンドと `--metadata snowball-auto-extract=true` オプションを使用して AWS CLI `cp` Snowball Edge デバイスに転送します。 `snowball-auto-extract=true` を指定すると、バッチ処理されたファイルのサイズが 100 GB 以下であれば、データが Amazon S3 にインポートされる際に、アーカイブファイルのコンテンツが自動的に抽出されます。

Note

100 GB を超えるバッチは、Amazon S3 にインポートされる際に抽出されません。

小さなファイルのバッチ処理をするには

1. バッチ処理をする小さなファイルの形式を決めます。自動抽出機能は、TAR、ZIP、`tar.gz` 形式でサポートされます。
2. サイズやまとめてバッチ処理するファイルの数を含め、まとめてバッチ処理する小さなファイルを特定します。
3. 次の例に示すように、コマンドラインでファイルをバッチ処理します。
 - Linux では、デバイスにファイルを転送するときに使用するのと同じコマンドラインで、ファイルのバッチ処理ができます。

```
tar -cf - /Logs/April | aws s3 cp - s3://mybucket/batch01.tar --metadata snowball-auto-extract=true --endpoint http://192.0.2.0:8080
```

Note

または、任意のアーカイブユーティリティを使用して 1 つ以上の大きなアーカイブにファイルをバッチ処理することもできます。ただし、この方法では、Snowball に転送する前にアーカイブを保存するための追加のローカルストレージが必要となります。

- Windows では、すべてのファイルがコマンドの実行元と同じディレクトリにある場合、以下のコマンド例を使用してファイルをバッチ処理します。

```
7z a -tzip -so "test" | aws s3 cp - s3://mybucket/batch01.zip --metadata snowball-auto-extract=true --endpoint http://192.0.2.0:8080
```


コマンドの実行元とは別のディレクトリのファイルをバッチ処理する場合は、以下のコマンド例を使用します。

```
7z a -tzip -so "test" "c:\temp" | aws s3 cp - s3://mybucket/batch01.zip --  
metadata snowball-auto-extract=true --endpoint http://10.x.x.x:8080
```

Note

Microsoft Windows 2016の場合、Tar を利用できませんが、Windows 用 Tar ウェブサイトからダウンロードできます。

7ZIP は、7ZIP ウェブサイトからダウンロードできます。

4. Snowball Edge を使用して、Amazon S3 に転送する小さなファイルをすべてアーカイブするまで、操作を繰り返します。
5. アーカイブしたファイルを Snowball に転送します。データを自動抽出し、ステップ 1 で前述したサポートされているアーカイブ形式のいずれかを使用した場合は、`--metadata snowball-auto-extract=true` オプションを指定してコマンドを使用します AWS CLI `cp`。

Note

非アーカイブファイルがある場合は、このコマンドを使用しないでください。

アーカイブファイルを作成する場合、抽出によって現在のデータ構造が維持されます。つまり、ファイルとフォルダを含むアーカイブファイルを作成すると、Snowball Edge は Amazon S3 プロセスへの取り込み中にこれを再作成します。

アーカイブファイルは、保存されているディレクトリと同じディレクトリに抽出され、それに応じてフォルダ構造が構築されます。アーカイブファイルをコピーするときは、`--metadata snowball-auto-extract=true` フラグを設定することが重要です。設定しないと、Snowball Edge は、Amazon S3 にインポートされるときにデータを抽出しません。

ファイル `a.txt`、`b.txt`、`c.txt` を含む `/Logs/April/` のフォルダ構造がある場合、ステップ 3 の例を使用します。このアーカイブファイルが `/mybucket/` のルートに配置されている場合、抽出後のデータは次のようになります。

```
/mybucket/Logs/April/a.txt  
/mybucket/Logs/April/b.txt
```

```
/mybucket/Logs/April/c.txt
```

アーカイブファイルが /mybucket/test/ に配置されている場合、抽出は次のようになります。

```
/mybucket/Test/Logs/April/a.txt  
/mybucket/Test/Logs/April/b.txt  
/mybucket/Test/Logs/April/c.txt
```

サポートされている AWS CLI コマンド

以下に、該当する AWS Command Line Interface (AWS CLI) コマンドのエンドポイントとして Snow Family デバイスの Amazon S3 アダプターまたは Amazon S3 互換ストレージを指定する方法について説明します。また、Snow Family AWS Snowball Edge デバイスのアダプターまたは Amazon S3 互換ストレージを使用して、デバイスへのデータ転送がサポートされている Amazon S3 の AWS CLI コマンドのリストも参照できます。

Note

AWS CLI 呼び出し先のリージョンの指定など AWS CLI、のインストールと設定については、[AWS Command Line Interface 「ユーザーガイド」](#) を参照してください。

現在、Snowball Edge デバイスは、バージョン 1.16.14 以前のみの AWS CLI で Amazon S3 アダプターの使用をサポートしています。[Snowball Edge クライアントバージョン](#) を参照してください。Snow Family デバイスの Amazon S3 互換ストレージを使用している場合は、AWS CLI の最新バージョンを使用できます。最新バージョンをダウンロードして使用するには、「[AWS Command Line Interface ユーザーガイド](#)」を参照してください。

Note

AWS CLI のバージョン 1.16.14 をインストールする前に、Python のバージョン 2.6.5+ または 3.4 をインストールしていることを確認してください。

Amazon S3 でサポートされている AWS CLI コマンド

以下は、AWS Snowball Edge デバイスがサポートする Amazon S3 の AWS CLI コマンドとオプションのサブセットの説明です。コマンドまたはオプションが以下に一覧表示されていない場合、そ

のコマンドまたはオプションはサポートされません。--sse や --storage-class など、一部のサポートされていないオプションを、コマンドとともに宣言できます。ただし、それらのオプションは無視され、データのインポート方法に影響はありません。

- **cp** - ファイルまたはオブジェクトを AWS Snowball Edge デバイスとの間でコピーします。このコマンドのオプションは、次のとおりです。
 - --dryrun (ブール値) – 指定したコマンドによって実行されなかったオペレーションが表示されます。
 - --quiet (ブール値) – 指定したコマンドによって実行されたオペレーションは表示されません。
 - --include (文字列) – 指定したパターンと一致するファイルまたはオブジェクトをコマンドから除外しないでください。詳細については、AWS CLI コマンドリファレンスの [Use of Exclude and Include Filters](#) を参照してください。
 - --exclude (文字列) – 指定したパターンと一致するすべてのファイルまたはオブジェクトをコマンドから除外します。
 - --follow-symlinks | --no-follow-symlinks (ブール値) – ローカルファイルシステムから Amazon S3 にアップロードする場合にのみ、シンボリックリンク (symlinks) をたどります。Amazon S3 はシンボリックリンクをサポートしないため、リンクターゲットのコンテンツはリンク名でアップロードされます。どちらのオプションも指定されていない場合、デフォルトでは、シンボリックリンクをたどります。
 - --only-show-errors (ブール値) – エラーと警告のみが表示されます。その他の出力は非表示になります。
 - --recursive (ブール値) – コマンドは、指定したディレクトリまたはプレフィックスのすべてのファイルまたはオブジェクトに対して実行されます。
 - --page-size (整数) – 各リスト操作に回答して返される結果の数。デフォルト値は 1000 です (許可される最大値)。オペレーションがタイムアウトする場合は、それより低い値を使用すると有効な可能性があります。
 - --metadata (マップ) – Amazon S3 内のオブジェクトに保存するメタデータのマップ。このマップは、このリクエストに含まれているすべてのオブジェクトに適用されます。同期では、この機能では、変更されていないファイルは新しいメタデータを受け取らないことを意味します。2 つの Amazon S3 の場所の間でコピーする際は、特に指定がない限り、metadata-directive の引数はデフォルトで REPLACE になります。
- **ls** – AWS Snowball Edge デバイス上のオブジェクトを一覧表示します。このコマンドのオプションは、次のとおりです。

- `--human-readable` (ブール値) – ファイルサイズが、人が読み取れる形式で表示されます。
- `--summarize` (ブール値) – 概要情報が表示されます。この情報は、オブジェクトの数とその合計サイズです。
- `--recursive` (ブール値) – コマンドは、指定したディレクトリまたはプレフィックスのすべてのファイルまたはオブジェクトに対して実行されます。
- `--page-size` (整数) – 各リスト操作に応答して返される結果の数。デフォルト値は 1000 です (許可される最大値)。オペレーションがタイムアウトする場合は、それより低い値を使用すると有効な可能性があります。
- `rm` – AWS Snowball Edge デバイス上のオブジェクトを削除します。このコマンドのオプションは、次のとおりです。
 - `--dryrun` (ブール値) – 指定したコマンドによって実行されなかったオペレーションが表示されます。
 - `--include` (文字列) – 指定したパターンと一致するファイルまたはオブジェクトをコマンドから除外しないでください。詳細については、AWS CLI コマンドリファレンスの [Use of Exclude and Include Filters](#) を参照してください。
 - `--exclude` (文字列) – 指定したパターンと一致するすべてのファイルまたはオブジェクトをコマンドから除外します。
 - `--recursive` (ブール値) – コマンドは、指定したディレクトリまたはプレフィックスのすべてのファイルまたはオブジェクトに対して実行されます。
 - `--page-size` (整数) – 各リスト操作に応答して返される結果の数。デフォルト値は 1000 です (許可される最大値)。オペレーションがタイムアウトする場合は、それより低い値を使用すると有効な可能性があります。
 - `--only-show-errors` (ブール値) – エラーと警告のみが表示されます。その他の出力は非表示になります。
 - `--quiet` (ブール値) – 指定したコマンドによって実行されたオペレーションは表示されません。
- `sync` – ディレクトリとプレフィックスを同期します。このコマンドは、新しいファイルおよび更新されたファイルを送信元ディレクトリから送信先にコピーします。このコマンドが送信先にディレクトリを作成するのは、ファイルが 1 つ以上含まれる場合に限られます。

⚠ Important

1 つのディレクトリから同じ Snowball Edge の別のディレクトリへの同期はサポートされません。

ある AWS Snowball デバイスから別の AWS Snowball デバイスへの同期はサポートされていません。
このオプションは、オンプレミスのデータストレージと Snowball Edge の間でコンテンツを同期する場合にのみ使用できます。

- `--dryrun` (ブール値) – 指定したコマンドによって実行されなかったオペレーションが表示されます。
- `--quiet` (ブール値) – 指定したコマンドによって実行されたオペレーションは表示されません。
- `--include` (文字列) – 指定したパターンと一致するファイルまたはオブジェクトをコマンドから除外しないでください。詳細については、AWS CLI コマンドリファレンスの [Use of Exclude and Include Filters](#) を参照してください。
- `--exclude` (文字列) – 指定したパターンと一致するすべてのファイルまたはオブジェクトをコマンドから除外します。
- `--follow-symlinks` または `--no-follow-symlinks` (ブール値) – ローカルファイルシステムから Amazon S3 にアップロードする場合にのみ、シンボリックリンク (symlinks) をたどります。Amazon S3 はシンボリックリンクをサポートしないため、リンクターゲットのコンテンツはリンク名でアップロードされます。どちらのオプションも指定されていない場合、デフォルトでは、シンボリックリンクをたどります。
- `--only-show-errors` (ブール値) – エラーと警告のみが表示されます。その他の出力は非表示になります。
- `--no-progress` (ブール値) – ファイル転送の進行状況は表示されません。このオプションは、`--quiet` オプションと `--only-show-errors` オプションが指定されていない場合にのみ適用されます。
- `--page-size` (整数) – 各リスト操作に回答して返される結果の数。デフォルト値は 1000 です (許可される最大値)。オペレーションがタイムアウトする場合は、それより低い値を使用すると有効な可能性があります。
- `--metadata` (マップ) – Amazon S3 内のオブジェクトに保存するメタデータのマップ。このマップは、このリクエストに含まれているすべてのオブジェクトに適用されます。同期では、この機能では、変更されていないファイルは新しいメタデータを受け取らないことを意味します。2 つの Amazon S3 の場所の間でコピーする際は、特に指定がない限り、`metadata-directive` の引数はデフォルトで `REPLACE` になります。

⚠ Important

1つのディレクトリから同じ Snowball Edge の別のディレクトリへの同期はサポートされません。
ある AWS Snowball デバイスから別の AWS Snowball デバイスへの同期はサポートされていません。
このオプションは、オンプレミスのデータストレージと Snowball Edge の間でコンテンツを同期する場合にのみ使用できます。

- `--size-only` (ブール値) – このオプションにより、各キーのサイズだけを基準として、送信元から送信先へ同期するかどうか決定されます。
- `--exact-timestamps` (ブール値) – Amazon S3 からローカルストレージに同期する際、タイムスタンプが完全に一致する場合にのみ、同サイズの項目が無視されます。デフォルトの動作では、ローカルバージョンが Amazon S3 バージョンより新しい場合を除き、同サイズの項目が無視されます。
- `--delete` (ブール値) – 送信先だけに存在し、送信元には存在しないファイルが、同期中に削除されます。

ファイルまたはフォルダは、「my photo.jpg」や「My Documents」のように名前にスペースを含めて使用できます。ただし、AWS CLI コマンドでスペースを適切に処理してください。詳細については、AWS Command Line Interface ユーザーガイドの[AWS CLIのパラメータ値の指定](#)を参照してください。

サポートされている REST API アクション

以下に、AWS Snowball Edge デバイスと Amazon S3 で使用できる REST API アクションを示します。

トピック

- [Snowball Edge デバイスでサポートされる REST API アクション](#)
- [Amazon S3 アダプターでサポートされる REST API アクション](#)

Snowball Edge デバイスでサポートされる REST API アクション

HEAD Snowball Edge

説明

現在のところ、Snowball Edge REST API オペレーションは 1 つだけです。このオペレーションは、特定のデバイスのステータス情報を返すために使用できます。このオペレーションは、Snowball Edge のステータスを返します。このステータスには、トラブルシューティング AWS Support の目的でで使用できる情報が含まれます。

このオペレーションは AWS SDKs または では使用できません AWS CLI。curl または HTTP クライアントを使用することをお勧めします。このオペレーションに対しては、リクエストに署名する必要はありません。

リクエスト

以下の例では、Snowball Edge の IP アドレスは `192.0.2.0` です。この値を実際のデバイスの IP アドレスに置き換えます。

```
curl -X HEAD http://192.0.2.0:8080
```

レスポンス

```
<Status xsi:schemaLocation="http://s3.amazonaws.com/doc/2006-03-01/" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <snowballIp>127.0.0.1</snowballIp>
  <snowballPort>8080</snowballPort>
  <snowballId>device-id</snowballId>
  <totalSpaceInBytes>499055067136</totalSpaceInBytes>
  <freeSpaceInBytes>108367699968</freeSpaceInBytes>
  <jobId>job-id</jobId>
  <snowballServerVersion>1.0.1</snowballServerVersion>
  <snowballServerBuild>DevBuild</snowballServerBuild>
  <snowballClientVersion>Version 1.0</snowballClientVersion>
  <snowballRoundTripLatencyInMillis>33</snowballRoundTripLatencyInMillis>
</Status>
```

Amazon S3 アダプターでサポートされる REST API アクション

以下は、Amazon S3 アダプターの使用でサポートされる Amazon S3 REST API アクションの一覧です。このリストには、Amazon S3 での API アクションの動作に関する情報へのリンクが含まれてい

ます。このリストでは、Amazon S3 API アクションと対応する AWS Snowball Edge デバイスの動作の違いについても説明します。以下の例が示すように、AWS Snowball Edge デバイスから返されるレスポンスはすべて、Server を AWSSnowball として宣言します。

```
HTTP/1.1 201 OK
x-amz-id-2: JuKZqmXuiwFeDQxhD7M8KtsKobSzWA1QEjLbTMTagkKdBX2z7I1/jGhDeJ3j6s80
x-amz-request-id: 32FE2CEB32F5EE25
Date: Fri, 08 2016 21:34:56 GMT
Server: AWSSnowball
```

Amazon S3 REST API コールでは、SigV4 署名が必要です。AWS CLI または AWS SDK を使用してこれらの API コールを行う場合、SigV4 署名が処理されます。それ以外の場合は、お客様独自の SigV4 署名ソリューションを実装する必要があります。詳細については、Amazon Simple Storage Service ユーザーガイドの「[リクエストの認証 \(AWS 署名バージョン 4\)](#)」を参照してください。

- [GET Bucket \(オブジェクトのリスト化\) バージョン 1](#)は、サポートされます。ただし、この GET オペレーションの実装では、以下はサポートされません。
 - ページ分割
 - マーカー
 - 区切り記号
 - リストが返される際、リストはソートされません。

Version 1 のみがサポートされます。GET Bucket (List Objects) version 2 はサポートされません。

- [GET Service](#)
- [HEAD Bucket](#)
- [HEAD Object](#)
- [GET Object](#) – Snow デバイスの S3 バケットからオブジェクトをダウンロードします。
- [PUT Object](#) – オブジェクトが を使用して AWS Snowball Edge デバイスにアップロードされると PUT Object、ETag が生成されます。

ETag は、オブジェクトのハッシュです。ETag は、オブジェクトのコンテンツに加えた変更のみを反映し、メタデータに加えた変更は反映しません。ETag は、オブジェクトデータの MD5 ダイジェストである場合と、そうでない場合があります。ETags の詳細については、Amazon Simple Storage Service API リファレンスの[共通の応答ヘッダー](#)を参照してください。

- [DELETE Object](#)

- [マルチパートアップロードの開始](#) – この実装では、AWS Snowball Edge デバイス上に既にあるオブジェクトに対してマルチパートアップロードリクエストを開始すると、まずそのオブジェクトが削除されます。次に、デバイスを部分的にコピーします AWS Snowball Edge。
- [マルチパートアップロードのリスト](#)
- [Upload Part](#)
- [Complete Multipart Upload](#)
- [マルチパートアップロードの中止](#)

Note

ここに記載されていない Amazon S3 アダプター REST API アクションはサポートされません。サポートされていない REST API アクションを Snowball Edge で使用すると、アクションがサポートされていないことを示すエラーメッセージが返されます。

NFS インターフェイスの管理

ネットワークファイルシステム (NFS) インターフェイスを使用して、デバイスがオペレーティングシステムにローカルストレージであるかのように、Snow Family デバイスにファイルをアップロードします。これにより、ファイルのコピー、ドラッグアンドドロップ、その他のグラフィカルユーザーインターフェイス機能など、オペレーティングシステムの機能を使用できるため、データ転送に対するよりユーザーフレンドリーなアプローチが可能になります。デバイス上の各 S3 バケットは NFS インターフェイスエンドポイントとして利用でき、データをコピーするためにマウントできます。NFS インターフェイスはインポートジョブで使用できます。

Snowball Edge デバイスが、デバイスを注文するジョブの作成時にそれを含めるように設定されている場合は、NFS インターフェイスを使用できます。デバイスが NFS インターフェイスを含めるように設定されていない場合は、S3 Family デバイスの S3 アダプターまたは Amazon S3 互換ストレージを使用してデータを転送します。S3 アダプターの詳細については、「」を参照してください [Amazon S3 アダプターストレージの管理](#)。Snow Family デバイスの Amazon S3 互換ストレージの詳細については、「」を参照してください [Snow Family デバイスの Amazon S3 互換ストレージをセットアップする](#)。

起動すると、NFS インターフェイスは 1 GB のメモリと 1 CPU を使用します。これにより、Snow Family デバイスで実行されている他のサービスの数や、が実行できる EC2-compatible インスタンスの数が制限される場合があります。

NFS インターフェイスを介して転送されるデータは、転送中に暗号化されません。NFS インターフェイスを設定する場合、CIDR ブロックを提供できます。Snow Family デバイスは、それらのブロック内のアドレスを持つクライアントコンピュータからの NFS インターフェイスへのアクセスを制限します。

デバイス上のファイルは、に返されると Amazon S3 に転送されます AWS。詳細については、[Amazon S3へのジョブのインポート](#)」を参照してください。

コンピュータオペレーティングシステムで NFS を使用する方法の詳細については、オペレーティングシステムのドキュメントを参照してください。

NFS インターフェイスを使用する場合は、次の詳細に注意してください。

- ファイル名は、Snow Family デバイス上のローカル S3 バケット内のオブジェクトキーです。キー名は一連の Unicode 文字で、UTF-8 にエンコードすると最大で 1,024 バイト長になります。可能な場合は NFSv4.1 を使用し、Unicode UTF-8 でファイル名をエンコードして、データを正常にインポートすることをお勧めします。UTF-8 でエンコードされていないファイル名は S3 にアップロードされないか、使用する NFS エンコーディングに応じて、異なるファイル名で S3 にアップロードされる場合があります。
- ファイルパスの最大長は 1,024 文字であることに注意してください。Snow Family デバイスでは、1,024 文字を超えるファイルパスはサポートされません。このファイルパスの長さを超えると、ファイルのインポートエラーが発生します。
- 詳細については、「Amazon Simple Storage Service ユーザーガイド」の「[オブジェクトキー](#)」を参照してください。
- NFS ベースの転送の場合、Snow Family デバイスから Amazon S3 にインポートされると、標準の POSIX スタイルのメタデータがオブジェクトに追加されます。さらに、NFS オプションを使用した Snow Family デバイスの Amazon S3 への内部インポートメカニズムの一部として現在使用している AWS DataSync メタデータ `x-amz-meta-user-agent-aws-datasync` が表示されます。
- 単一の Snowball Edge デバイスを使用して、最大 40M のファイルを転送できます。1 つのジョブで 4,000 万を超えるファイルを転送する必要がある場合は、転送ごとのファイル数を減らすためにファイルをバッチ処理してください。拡張 NFS インターフェイスまたは S3 インターフェイスを搭載した Snowball Edge デバイスの場合、個々のファイルのサイズは任意で、最大ファイルサイズは 5 TB です。

GUI ツールである を使用して AWS OpsHub NFS インターフェイスを設定および管理することもできます。詳細については、「[オフラインファイル転送に NFS を使用する NFS インターフェイスの管理](#)」を参照してください。

Snow Family デバイスの NFS 設定

NFS インターフェイスは Snow Family デバイスでデフォルトでは実行されていないため、デバイスへのデータ転送を有効にするには起動する必要があります。Snow Family デバイスで実行されている仮想ネットワークインターフェイス (VNI) の IP アドレスを指定し、必要に応じてファイル共有へのアクセスを制限することで、NFS インターフェイスを設定できます。NFS インターフェイスを設定する前に、Snow Family デバイスで仮想ネットワークインターフェイス (VNI) を設定します。詳細については、[コンピューティングインスタンスのネットワーク設定](#)を参照してください。

NFS インターフェイス用に Snow Family デバイスを設定する

- describe-service コマンドを使用して、NFS インターフェイスがアクティブかどうかを判断します。

```
snowballEdge describe-service --service-id nfs
```

コマンドは NFS サービスの状態、ACTIVE または INACTIVE を返します。

```
{
  "ServiceId" : "nfs",
  "Status" : {
    "State" : "ACTIVE"
  }
}
```

State 名前の値が ACTIVE の場合、NFS インターフェイスサービスはアクティブであり、Snow Family デバイス NFS ボリュームをマウントできます。詳細については、「

[NFS インターフェイスを起動したら、エンドポイントをローカルストレージとしてクライアントコンピュータにマウントします。](#)

[Windows、Linux、macOS オペレーティングシステム用のデフォルトのマウントコマンドを次に示します。](#)

- Windows :

```
mount -o nolock rsize=128 wsize=128 mtype=hard nfs-interface-ip-address:/buckets/BucketName *
```

- Linux:

```
mount -t nfs nfs-interface-ip-address:/buckets/BucketName mount_point
```

- macOS:

```
mount -t nfs -o vers=3,rsize=131072,wsize=131072,nolocks,hard,retrans=2 nfs-interface-ip-address:/buckets/$bucketname mount_point
```

」を参照してください。値が の場合はINACTIVE、サービスを開始する必要があります。

Snow Family デバイスで NFS サービスを開始する

必要に応じて仮想ネットワークインターフェイス (VNI) を起動し、Snow Family デバイスで NFS サービスを起動します。必要に応じて、NFS サービスを開始するときに、許可されたネットワークアドレスのブロックを指定します。アドレスを指定しない場合、NFS エンドポイントへのアクセスは制限されません。

1. describe-virtual-network-interface コマンドを使用して、Snow Family デバイスで使用可能な VNI を確認します。

```
snowballEdge describe-virtual-network-interfaces
```

Snow Family デバイスで 1 つ以上の VNI がアクティブな場合、コマンドは以下を返します。

```
snowballEdge describe-virtual-network-interfaces
[
  {
    "VirtualNetworkInterfaceArn" : "arn:aws:snowball-device:::interface/
s.ni-8EXAMPLE8EXAMPLE8",
```

```
"PhysicalNetworkInterfaceId" : "s.ni-8EXAMPLEaEXAMPLEd",
"IpAddressAssignment" : "DHCP",
"IpAddress" : "192.0.2.0",
"Netmask" : "255.255.255.0",
"DefaultGateway" : "192.0.2.1",
"MacAddress" : "EX:AM:PL:E1:23:45"
},{
  "VirtualNetworkInterfaceArn" : "arn:aws:snowball-device:::interface/
s.ni-1EXAMPLE1EXAMPLE1",
  "PhysicalNetworkInterfaceId" : "s.ni-8EXAMPLEaEXAMPLEd",
  "IpAddressAssignment" : "DHCP",
  "IpAddress" : "192.0.2.2",
  "Netmask" : "255.255.255.0",
  "DefaultGateway" : "192.0.2.1",
  "MacAddress" : "12:34:5E:XA:MP:LE"
}
]
```

NFS インターフェイスで使用する VNI `VirtualNetworkInterfaceArn` の名前の値を書き留めます。

2. 使用可能な VNI がいない場合は、`create-virtual-network-interface` コマンドを使用して NFS インターフェイスの VNI を作成します。詳細については、[「仮想ネットワークインターフェイス \(VNI\) のセットアップ」](#) を参照してください。
3. `start-service` コマンドを使用して NFS サービスを開始し、それを VNI に関連付けます。NFS インターフェイスへのアクセスを制限するには、コマンドに `service-configuration` および `AllowedHosts` パラメータを含めます。

```
snowballEdge start-service --virtual-network-interface-arns arn-of-vni --service-id
nfs --service-configuration AllowedHosts=CIDR-address-range
```

4. `describe-service` コマンドを使用して、サービスのステータスを確認します。State 名前の値が の場合に実行されますACTIVE。

```
snowballEdge describe-service --service-id nfs
```

このコマンドは、サービスの状態、NFS エンドポイントの IP アドレスとポート番号、およびエンドポイントへのアクセスが許可されている CIDR 範囲を返します。

```
{
  "ServiceId" : "nfs",
  "Status" : {
    "State" : "ACTIVE"
  },
  "Endpoints" : [ {
    "Protocol" : "nfs",
    "Port" : 2049,
    "Host" : "192.0.2.0"
  } ],
  "ServiceConfiguration" : {
    "AllowedHosts" : [ "10.24.34.0/23", "198.51.100.0/24" ]
  }
}
```

クライアントコンピュータへの NFS エンドポイントのマウント

NFS インターフェイスを起動したら、エンドポイントをローカルストレージとしてクライアントコンピュータにマウントします。

Windows、Linux、macOS オペレーティングシステム用のデフォルトのマウントコマンドを次に示します。

- Windows :

```
mount -o nolock rsize=128 wsize=128 mtype=hard nfs-interface-ip-address:/
buckets/BucketName *
```

- Linux:

```
mount -t nfs nfs-interface-ip-address:/buckets/BucketName mount_point
```

- macOS:

```
mount -t nfs -o vers=3,rsiz=131072,wsiz=131072,nolocks,hard,retrans=2 nfs-  
interface-ip-address:/buckets/$bucketname mount_point
```

NFS インターフェイスの停止

NFS インターフェイス経由でファイルの転送が完了したら、Snow Family デバイスの電源をオフにする前に、`stop-service` コマンドを使用して NFS サービスを停止します。

```
snowballEdge stop-service --service-id nfs
```

AWS IoT Greengrass を使用して、Amazon EC2 互換インスタンスでプリインストールされたソフトウェアを実行する

AWS IoT Greengrass は、オープンソースのモノのインターネット (IoT) エッジランタイムとクラウドサービスであり、デバイスで IoT アプリケーションを構築、デプロイ、管理するのに役立ちます。を使用して AWS IoT Greengrass、デバイスが生成したデータに対してローカルで動作し、機械学習モデルに基づいて予測を実行し、デバイスデータをフィルタリングして集計できるようにするソフトウェアを構築できます。の詳細については [AWS IoT Greengrass、 「AWS IoT Greengrass Version 2 デベロッパーガイド」の「とは AWS IoT Greengrass」](#) を参照してください。

Snow Family AWS IoT Greengrass デバイスで を使用すると、デバイスが生成された場所に近いデータを収集および分析し、ローカルイベントに自律的に反応し、ローカルネットワーク上の他のデバイスと安全に通信できるようになります。

Amazon EC2 互換インスタンスの設定

Note

Snow Family AWS IoT Greengrass Version 2 デバイスに をインストールするには、デバイスがインターネットに接続されていることを確認します。インストール後、Snow Family デバイスが と連携するためにインターネットは必要ありません AWS IoT Greengrass。

の EC2-compatible インスタンスを設定するには AWS IoT Greengrass V2

- パブリック IP アドレスと SSH キーを使用して、AWS IoT Greengrass 検証済みの AMI を起動します。
 - の使用 AWS CLI: [run-instances](#)。
 - の使用 AWS OpsHub: [Amazon EC2-compatible インスタンス を起動します](#)。

Note

インスタンスに関連付けられているパブリック IP アドレスと SSH キー名を書き留めます。

- SSH を使用して EC2 互換インスタンスに接続します。これを行うには、デバイスに接続されているコンピュータで、次のコマンドを実行します。`ssh-key` を、EC2 互換インスタンスの起動に使用したキーに置き換えます。`public-ip-address` を EC2-compatible インスタンスのパブリック IP アドレス `public-ip-address` に置き換えます。

```
ssh -i ssh-key ec2-user@ public-ip-address
```

Important

コンピュータで Microsoft Windows の以前のバージョンを使用している場合は、SSH コマンドがないか、SSH があっても EC2 互換インスタンスに接続できない可能性があります。EC2 互換インスタンスに接続するには、PuTTY をインストールして設定します。PuTTY は無料のオープンソース SSH クライアントです。SSH キーを .pem 形式から PuTTY 形式に変換して、EC2 インスタンスに接続する必要があります。から .pem PuTTY 形式に変換する方法については、Amazon EC2 [ユーザーガイドの PuTTYgen を使用してプライベートキーを変換する](#) を参照してください。

のインストール AWS IoT Greengrass

次に、EC2-compatible インスタンスをローカル開発に使用できる AWS IoT Greengrass Core デバイスとして設定します。

をインストールするには AWS IoT Greengrass

1. 次のコマンドを使用して、 の前提条件ソフトウェアをインストールします AWS IoT Greengrass。このコマンドは、 AWS Command Line Interface (AWS CLI) v2、 Python 3、 および Java 8 をインストールします。

```
curl "https://awscli.amazonaws.com/awscli-exe-linux-x86_64.zip" -o "awscliv2.zip"
&& unzip awscliv2.zip && sudo ./aws/install && sudo yum -y install python3
java-1.8.0-openjdk
```

2. AWS IoT Greengrass ソフトウェアを実行するアクセス許可をルートユーザーに付与し、 sudoers 設定ファイルの から root ALL=(ALL:ALL) ALL root ALL=(ALL) ALLにルートアクセス許可を変更します。

```
sudo sed -in 's/root\tALL=(ALL)/root\tALL=(ALL:ALL)/' /etc/sudoers
```

3. AWS IoT Greengrass Core ソフトウェアをダウンロードするには、 次のコマンドを使用します。

```
curl -s https://d2s8p88vqu9w66.cloudfront.net/releases/greengrass-nucleus-
latest.zip > greengrass-nucleus-latest.zip && unzip greengrass-nucleus-latest.zip -
d GreengrassCore && rm greengrass-nucleus-latest.zip
```

4. 次のコマンドを使用して、 AWS IoT Greengrass Core ソフトウェアをインストールするための認証情報を提供します。例の値は実際の認証情報に置き換えてください。

```
export AWS_ACCESS_KEY_ID=AKIAIOSFODNN7EXAMPLE
export AWS_SECRET_ACCESS_KEY=wJalrXUtnFEMI/K7MDENG/bPxrFiCYEXAMPLEKEY
```

Note

これらは、 Snow Family デバイスではなく、 AWS リージョンの IAM ユーザーからの認証情報です。

5. AWS IoT Greengrass Core ソフトウェアをインストールするには、 次のコマンドを使用します。コマンドは、 コアソフトウェアが動作するために必要な AWS リソースを作成し、 AMI の起動時に実行されるシステムサービスとしてコアソフトウェアを設定します。

コマンド内の以下のパラメータを置き換えてください。

- `region`: リソースを検索または作成する AWS リージョン。
- `MyGreengrassCore`: AWS IoT Greengrass コアデバイスの AWS IoT モノの名前。
- `MyGreengrassCoreGroup`: AWS IoT Greengrass コアデバイスの AWS IoT モノグループの名前。

```
sudo -E java -Droot="/greengrass/v2" -Dlog.store=FILE \  
-jar ./GreengrassInstaller/lib/Greengrass.jar \  
--aws-region region \  
--thing-name MyGreengrassCore \  
--thing-group-name MyGreengrassCoreGroup \  
--thing-policy-name GreengrassV2IoTThingPolicy \  
--tes-role-name GreengrassV2TokenExchangeRole \  
--tes-role-alias-name GreengrassCoreTokenExchangeRoleAlias \  
--component-default-user ggc_user:ggc_group \  
--provision true \  
--setup-system-service true \  
--deploy-dev-tools true
```

Note

このコマンドは、Amazon Linux 2 AMI を実行している Amazon EC2 互換インスタンス用です。Windows AMI については、[AWS IoT Greengrass 「Core ソフトウェアのインストール」](#)を参照してください。

完了すると、ローカルで使用できるように Snow Family デバイスで AWS IoT Greengrass コアが実行されます。

AWS Snowball Edge AWS Lambda での の使用

AWS Lambda を搭載した AWS IoT Greengrass は、サーバーレスコード (Lambda 関数) を Snowball Edge デバイスでローカルに実行できるコンピューティングサービスです。Lambda を使用して、Message Queuing Telemetry Transport (MQTT) メッセージを使用して Snowball Edge デバイス

で Lambda 関数を呼び出し、Lambda 関数で Python コードを実行し、それらを使用してクラウド内のパブリック AWS サービスエンドポイントを呼び出すことができます。Snowball Edge デバイスで Lambda 関数を使用するには、で AWS リージョン サポートされている で Snowball Edge ジョブを作成する必要があります AWS IoT Greengrass。有効な のリストについては AWS リージョン、「」の [AWS IoT Greengrass](#) 「」を参照してください AWS 全般のリファレンス。Lambda on Snowball Edge は、Lambda と Snowball Edge デバイスを利用可能なリージョンで使用できます。

Note

各関数に最小推奨値である 128 MB のメモリを割り当てた場合、1 つのジョブに最大 7 つの Lambda 関数を使用することができます。

トピック

- [開始する前に](#)
- [Snowball Edge デバイスに Lambda 関数をデプロイする](#)

開始する前に

Snowball Edge で実行する Python 言語の Lambda 関数を作成する前に、次のサービス、概念、関連トピックを十分に理解しておくことをお勧めします。

の前提条件 AWS IoT Greengrass

AWS IoT Greengrass は、AWS クラウド 機能をローカルデバイスに拡張するソフトウェアです。AWS IoT Greengrass を使用すると、ローカルデバイスは、ローカルネットワーク上で相互に安全に通信しながら、情報ソースに近いデータを収集および分析できます。具体的には、を使用するデベロッパー AWS IoT Greengrass は、でサーバーレスコード (Lambda 関数) を作成できます AWS クラウド。そして開発者は簡単にこのコードをデバイスにデプロイし、ローカルでアプリケーションを実行できます。

Snowball Edge AWS IoT Greengrass でを使用する際に理解しておくべき重要な AWS IoT Greengrass 概念は次のとおりです。

- AWS IoT Greengrass 要件 – AWS IoT Greengrass 要件の完全なリストについては、「AWS IoT Greengrass Version 2 デベロッパーガイド」の [「要件」](#) を参照してください。

- AWS IoT Greengrass core – AWS IoT Greengrass コアソフトウェアをダウンロードし、デバイスで実行されている EC2 インスタンスにインストールします。このガイドの[AWS IoT Greengrass Amazon EC2 インスタンスでの の使用](#)」を参照してください。

Snowball Edge デバイスで Lambda 関数を使用するには、まずデバイスの Amazon EC2 インスタンスに AWS IoT Greengrass Core ソフトウェアをインストールする必要があります。Snowball Edge デバイスで使用する予定の Lambda 関数は、Snowball Edge デバイス AWS IoT Greengrass へのインストールに使用するのと同じアカウントで作成する必要があります。Snowball Edge デバイスに をインストールする方法については、AWS IoT Greengrass 「」を参照してください[AWS IoT Greengrass を使用して、Amazon EC2 互換インスタンスでプリインストールされたソフトウェアを実行する](#)。

- AWS IoT Greengrass グループ – Snowball Edge デバイスは、AWS IoT Greengrass グループのコアデバイスとしてグループの一部です。グループの詳細については、「AWS IoT Greengrass デベロッパーガイド」の「[AWS Greengrass IoT Groups](#)」を参照してください。
- MQTT – 業界標準 AWS IoT Greengrass の軽量 MQTT プロトコルを使用して、グループ内で通信します。AWS IoT Greengrass グループ内の MQTT と互換性のあるデバイスまたはソフトウェアは、MQTT メッセージを呼び出すことができます。関連する MQTT メッセージが関数を呼び出すように指定した場合、これらのメッセージで Lambda 関数を呼び出すことができます。

の前提条件 AWS Lambda

AWS Lambda は、サーバーのプロビジョニングや管理を行わずにコードを実行できるようにするコンピューティングサービスです。Snowball Edge で Lambda を使用する際には、以下の Lambda 概念を理解していることが重要です。

- Lambda 関数 - Lambda にアップロードされ、公開されるカスタムコードで、Snowball Edge で使用されます。詳細については、AWS Lambda デベロッパーガイドの [Lambda 関数](#)を参照してください。
- Lambda コンソール – Snowball Edge で使用する Python 言語 Lambda 関数をアップロード、更新、発行するコンソール。[Lambda コンソール](#)の詳細については、「AWS Lambda デベロッパーガイド」の「[Lambda コンソール](#)」を参照してください。
- Python – Snowball Edge AWS IoT Greengrass の を搭載した Lambda 関数に使用される高レベルのプログラミング言語。 は Python バージョン 3.8.x AWS IoT Greengrass をサポートしています。

Snowball Edge デバイスに Lambda 関数をデプロイする

AWS IoT Greengrass グループ内の Snowball Edge デバイスで Lambda 関数を実行するには、関数をコンポーネントとしてインポートします。AWS IoT Greengrass コンソールを使用して関数をコンポーネントとしてインポートする方法の詳細については、「AWS IoT Greengrass Version 2 デベロッパーガイド」の「[コンポーネントとして Lambda 関数をインポートする \(コンソール\)](#)」を参照してください。

1. AWS IoT コンソールの Greengrass コンポーネントページで、コンポーネントの作成 を選択します。
2. [コンポーネントソース] で、[Lambda 関数をインポート] を選択します。[Lambda 関数] で Lambda 関数の名前を選択します。[Lambda 関数バージョン] で、関数のバージョンを選択します。
3. アクションの対象となるメッセージに関数をサブスクライブするには、[イベントソースを追加] を選択し、イベントを選択します。[タイムアウト (秒)] に、タイムアウト時間を秒単位で入力します。
4. [固定] で、関数を固定するかどうかを選択します。
5. [コンポーネントを作成] を選択します。
6. [デプロイ] を選択します。
7. [デプロイ] で [既存のデプロイに追加] を選択し、Greengrass グループを選択します。[次へ] をクリックします。
8. [パブリックコンポーネント] で、以下のコンポーネントを選択します。
 - aws.greengrass.Cli
 - aws.greengrass.LambdaLauncher
 - aws.greengrass.LambdaManager
 - aws.greengrass.LambdaRuntimes
 - aws.greengrass.Nucleus
9. [デプロイ] を選択します。

Amazon EC2 互換コンピューティングインスタンスの使用

このセクションでは、概念情報、手順、例など、AWS Snowball Edge デバイスで Amazon EC2-compatibleコンピューティングインスタンスを使用する方法の概要を説明します。

トピック

- [概要](#)
- [Snow Family デバイス上の Amazon EC2 と Amazon EC2 互換インスタンスの違い](#)
- [Snowball Edge のコンピューティングインスタンスの料金](#)
- [Snow Family デバイスでの Amazon EC2 互換 AMI の使用](#)
- [Snow Family デバイスに仮想マシンイメージをインポートする](#)
- [Snowball Edge での AWS CLI および API オペレーションの使用](#)
- [Snowball Edge デバイスのコンピューティングインスタンスのクォータ](#)
- [コンピューティングジョブの作成](#)
- [コンピューティングインスタンスのネットワーク設定](#)
- [SSH を使用して Snow Family デバイスのコンピューティングインスタンスに接続する](#)
- [同じ Snowball Edge 上の EC2 互換コンピューティングインスタンスから S3 バケットにデータを転送する](#)
- [コンピューティングインスタンスの Snowball Edge クライアントのコマンド](#)
- [Amazon EC2 互換エンドポイントの使用](#)
- [起動テンプレートを使用した Amazon EC2 互換インスタンスの自動起動](#)
- [Amazon EC2 互換インスタンスで Snow 用インスタンスメタデータサービスを使用する](#)
- [Amazon EC2 互換インスタンスでのブロッックストレージの使用](#)
- [Snowball Edge デバイスのセキュリティグループ](#)
- [サポートされるインスタンスメタデータとユーザーデータ](#)
- [EC2 互換インスタンスの停止](#)
- [Snowball Edge デバイスのコンピューティングインスタンスのトラブルシューティング](#)

概要

sbe1、sbe-c、sbe-g のインスタンスタイプを使用して、Snowball Edge でホストされる Amazon EC2 互換コンピューティングインスタンスを実行できます。sbe1 インスタンスタイプは、Snowball Edge Storage Optimized オプションを備えたデバイスで動作します。sbe-c インスタンスタイプは、Snowball Edge Compute Optimized オプションを備えたデバイスで動作します。sbe-c および sbe-g インスタンスタイプはいずれも、GPU オプション付きの Snowball Edge Compute Optimized を備えたデバイスで動作します。サポートされているインスタンスタイプについては、[Snowball Edge デバイスのコンピューティングインスタンスのクォータ](#)を参照してください。

Snowball Edge デバイスオプションの使用をサポートする 3 つのコンピューティングインスタンスタイプは、すべて Snowball Edge デバイ스에固有です。これらのインスタンスでは、クラウドベースのインスタンスと同様に、Amazon Machine Image (AMI) を起動する必要があります。クラウド上のインスタンスのベースイメージとなる AMI を選択してから、Snowball Edge ジョブを作成します。

Snowball Edge でコンピューティングインスタンスを使用するには、Snow Family デバイスを注文するジョブを作成し、AMIs を指定します。これは、[AWS Snow ファミリーマネジメントコンソール](#)、AWS CLI または AWS SDKs のいずれかを使用して実行できます。通常の場合、インスタンスを使用するには、ジョブの作成前にハウスキピングの前提条件をいくつか実行する必要があります。

デバイスが到着したら、AMI とインスタンスの管理を開始できます。Snowball Edge 上のインスタンスは、Amazon EC2 と互換性のあるエンドポイントを使用して管理できます。このタイプのエンドポイントは、Amazon EC2 互換の CLI コマンドや AWS SDK のアクションの大部分をサポートします。Snowball Edge AWS Management Console を使用して AMIs とコンピューティングインスタンスを管理することはできません。

デバイスの使用が終了したら、に 戻ります AWS。デバイスをインポートジョブで使用した場合、Amazon S3 アダプターまたは NFS インターフェイスを使用して転送されたデータは、Amazon S3 にインポートされます。それ以外の場合は、デバイスが に 返されたときに完全な消去を実行します AWS。この消去作業は National Institute of Standards and Technology (NIST) 800-88 基準に準拠しています。

Important

- Snowball Edge デバイスでの暗号化された AMI の使用は、サポートされません。
- Snowball Edge で実行されているコンピューティングインスタンスのデータは、にインポートされません AWS。

Snow Family デバイス上の Amazon EC2 と Amazon EC2 互換インスタンスの違い

AWS Snow Family EC2-compatible インスタンスを使用すると、お客様は EC2-compatible AMIs、EC2 APIs

Snowball Edge のコンピューティングインスタンスの料金

コンピューティングインスタンスの使用には追加料金がかかります。詳細については、「[AWS Snowball Edge の料金](#)」を参照してください。

Snow Family デバイスでの Amazon EC2 互換 AMI の使用

AWS Snow Family デバイスで Amazon マシンイメージ (AMI) を使用するには、まずデバイスに追加する必要があります。AMI は次の方法で追加できます。

- デバイスの注文時に、AMI をアップロードします。
- デバイスがお客様のサイトに到着したら、AMI を追加します。

Snow Family デバイスに付属する Amazon EC2 コンピューティングインスタンスは、デバイスに追加した Amazon EC2 AMI に基づいて起動されます。Amazon EC2 互換 AMI は、Linux と Microsoft Windows の両方のオペレーティングシステムをサポートします。

Linux

以下の Linux オペレーティングシステムがサポートされます。

- [Snow Family 用 Amazon Linux 2](#)

Note

この AMI の最新バージョンは、Snow Family デバイスが によって出荷される準備が整っているときに提供されます AWS。受信時にデバイス上のこの AMI のバージョンを確認するには、「」を参照してください [Snow Family 用 Amazon Linux 2 AMI のバージョンの確認](#)。

- [CentOS 7 \(x86_64\) - with Updates HVM](#)
- Ubuntu 16.04 LTS - Xenial (HVM)

Note

Ubuntu 16.04 LTS - Xenial (HVM) イメージは、ではサポートされなくなりましたが AWS Marketplace、Amazon EC2 VM Import/Export を介して Snowball Edge デバイスで使用でき、AMIs でローカルで実行できます。

- [Ubuntu 20.04 LTS - Focal](#)
- [Ubuntu 22.04 LTS - Jammy](#)

セキュリティのベストプラクティスとして、新しい Amazon Linux 2 AMIs up-to-date がリリースされたら、Amazon Linux 2 AMIs を Snow Family デバイスに保持します。[Snow Family デバイスの Amazon Linux 2 AMI を更新する](#) を参照してください。

Windows

以下の Windows オペレーティングシステムがサポートされます。

- Windows Server 2012 R2
- Windows Server 2016
- [Windows Server 2019]

VM Import/Export AWS を使用して Windows 仮想マシン (VM) イメージを にインポートすることで、デバイスに Windows AMIs を追加できます。または、デバイスがサイトにデプロイされた直後に、デバイスにイメージをインポートすることもできます。詳細については、「[Microsoft Windows AMI の追加](#)」を参照してください。

Note

を起点とする Windows AMIs AWS をデバイスに追加することはできません。UEFI はサポートされていないため、ローカルにインポートする AMI は BIOS ブートモードにする必要があります。

Snow Family は Bring-Your-Own-License (BYOL) モデルをサポートします。詳細については、「[Microsoft Windows AMI の追加](#)」を参照してください。

Note

AWS Snow Family EC2-compatible インスタンスを使用すると、お客様は EC2-compatible AMIs。 EC2 APIs

トピック

- [デバイスの注文時に AMI を追加する](#)
- [からの AMI の追加 AWS Marketplace](#)
- [AMI をローカルで追加する](#)
- [Microsoft Windows AMI の追加](#)
- [デバイスへの VM イメージのインポート](#)
- [最新の Amazon Linux 2 AMI のエクスポート](#)

デバイスの注文時に AMI を追加する

デバイスを注文する際、AWS Snow ファミリーマネジメントコンソールの [コンピューティングに EC2 インスタンスを使用 - オプション] セクションで AMI を選択することで、デバイスに AMI を追加できます。[コンピューティングに EC2 インスタンスを使用 - オプション] には、デバイスにロードできるすべての AMI が一覧表示されます。AMI は以下のカテゴリに分類されます。

- AWS Marketplace の AMIs — これらは、サポートされている AMIs のリストから作成された AMIs。AWS Marketplace でサポートされている AMI から AMIs 「」を参照してください [からの AMI の追加 AWS Marketplace](#)。
- VM Import/Export を使用してアップロードされた AMI — デバイスの注文時に、VM Import/Export を使用してアップロードされた AMI がコンソールに一覧表示されます。詳細については、「VM Import/Export ユーザーガイド」の「[VM Import/Export を使用してイメージとして VM をインポートする](#)」を参照してください。サポートされる仮想化環境の詳細については、[VM Import/Export 要件](#)を参照してください。

からの AMI の追加 AWS Marketplace

AWS Marketplace インスタンスを起動 AWS Marketplace し、そのインスタンスから AMIs を作成し、Snow デバイスの注文元と同じリージョンで AMI を設定することで、から Snow Family デバイスに多くの AMI を追加できます。次に、デバイスを注文するジョブを作成するときに、デバイスに AMI を含めることを選択できます。Marketplace から AMI を選択するときは、サポートされている製品コードとプラットフォームがあることを確認してください。

トピック

- [AWS Marketplace AMIs](#)
- [Snow Family 用 Amazon Linux 2 AMI のバージョンの確認](#)
- [Snow Family デバイスの AMI を設定する](#)

AWS Marketplace AMIs

から AWS Marketplace Snow Family デバイスに AMI を追加するプロセスを開始する前に、AMI の製品コードとプラットフォームの詳細が でサポートされていることを確認してください AWS リージョン。

1. Amazon EC2 コンソール (<https://console.aws.amazon.com/ec2/>) を開きます。
2. ナビゲーションバーから、インスタンスを起動するリージョンと、Snow Family デバイスを注文するジョブを作成するリージョンを選択します。場所に関係なく、利用可能な任意のリージョンを選択できます。
3. ナビゲーションペインで [AMI] を選択します。
4. フィルターオプションと検索オプションを使用して、表示される AMIs のリストをスコープし、条件に一致する AMIs のみを表示します。例えば、 が提供する AMIs の場合は AWS Marketplace、パブリックイメージ を選択します。次に、検索オプションを使用して、表示される AMIs。
 - (新しいコンソール) 検索バーを選択し、メニューから「所有者エイリアス」、「 = 演算子」、「値」を選択します。
 - (旧コンソール) [Search] (検索) バーを選択し、メニューから [Owner] (所有者)、次に値として [Amazon images] (Amazon イメージ) を選択します。

Note

からの AMIs ソース列に aws-marketplace AWS Marketplace が含まれます。

5. AMI ID 列で、AMI の AMI ID を選択します。
6. AMI のイメージ概要で、製品コードがリージョンでサポートされていることを確認します。詳細については、以下の表を参照してください。

サポートされている AWS Marketplace AMI 製品コード

AMI オペレーティングシステム	製品コード
Ubuntu Server 14.04 LTS	b3dl4415quatdndl4qa6kcu45
CentOS 7 (x86-64)	aw0evgkw8e5c1q413zgy5pjce
Ubuntu 16.04 LTS	csv6h7oyg29b7epjzg7qdr7no

AMI オペレーティングシステム	製品コード
Amazon Linux 2	avyfzznywe""kgl5qv5f57ska
Ubuntu 20.04 LTS	a8jyyfnf4hjutohctm41o2z18m
Ubuntu 22.04 LTS	47xbqns9xujfkkjt189a13aqe

7. 次に、プラットフォームの詳細に以下のリストからのエントリのいずれかが含まれていることも確認します。

- Amazon Linux、Ubuntu、または Debian
- Red Hat Linux bring-your-own-license
- Amazon RDS for Oracle bring-your-own-license
- Windows bring-your-own-license

Snow Family 用 Amazon Linux 2 AMI のバージョンの確認

Snow Family デバイスの Snow Family 用 Amazon Linux 2 AMI のバージョンを確認するには、次の手順に従います。続行する AWS CLI 前に、の最新バージョンをインストールします。詳細については、「[ユーザーガイド](#)」の「[の最新バージョンのインストールまたは更新 AWS CLI AWS Command Line Interface](#)」を参照してください。

- `describe-images` AWS CLI コマンドを使用して、AMI の説明を表示します。バージョンは説明に含まれています。前のステップのパブリックキー証明書を指定します。詳細については、コマンドリファレンスの「[describe-images AWS CLI](#)」を参照してください。

```
aws ec2 describe-images --endpoint http://snow-device-ip:8008 --region snow
```

Example `describe-images` コマンドの出力の

```
{
  "Images": [
    {
      "CreationDate": "2024-02-12T23:24:45.705Z",
```

```
    "ImageId": "s.ami-02ba84cb87224e16e",
    "Public": false,
    "ProductCodes": [
      {
        "ProductCodeId": "avyfzzywekktkg15qv5f57ska",
        "ProductCodeType": "marketplace"
      }
    ],
    "State": "AVAILABLE",
    "BlockDeviceMappings": [
      {
        "DeviceName": "/dev/xvda",
        "Ebs": {
          "DeleteOnTermination": true,
          "Iops": 0,
          "SnapshotId": "s.snap-0efb49f2f726fde63",
          "VolumeSize": 8,
          "VolumeType": "sbp1"
        }
      }
    ],
    "Description": "Snow Family Amazon Linux 2 AMI 2.0.20240131.0 x86_64
HVM gp2",
    "EnaSupport": false,
    "Name": "amzn2-ami-snow-family-hvm-2.0.20240131.0-x86_64-gp2-
b7e7f8d2-1b9e-4774-a374-120e0cd85d5a",
    "RootDeviceName": "/dev/xvda"
  }
]
}
```

この例では、Snow Family 用 Amazon Linux 2 AMI のバージョンは です**2.0.20240131.0**。これは名前の値にありますDescription。

Snow Family デバイスの AMI を設定する

1. Amazon EC2 コンソール (<https://console.aws.amazon.com/ec2/>) を開きます。
2. でサポートされている AMI の新しいインスタンスを起動します AWS Marketplace。

Note

インスタンスを起動する際、インスタンスに割り当てるストレージサイズがユースケースに適切であることを確認してください。確認するには、Amazon EC2 コンソールの [Add storage] (ストレージの追加) ステップを実行します。

3. Snowball Edge で実行するアプリケーションをインストールおよび設定し、正常に動作することを確認します。

Important

- 単一のボリューム AMI のみがサポートされます。
- AMI の EBS ボリュームは 10 TB 以下にする必要があります。AMI のデータに必要な EBS ボリュームサイズをプロビジョニングすることをお勧めします。これにより、AMI をエクスポートしてデバイスにロードするためにかかる時間を短縮できます。デバイスのデプロイ後、インスタンスのボリュームサイズを変更したり、ボリュームを追加したりできます。
- AMI の EBS スナップショットを暗号化しないでください。

4. このインスタンスを作成する際は、SSH キーペアで使用する PEM または PPK ファイルのコピーを作成します。Snowball Edge デバイスとの通信に使用する予定のサーバーに、このファイルを保存します。このファイルへのパスをメモします。これは、SSH を使用してデバイス上の EC2 互換インスタンスに接続するときになります。

Important

この手順を行わないと、Snowball Edge デバイスを受け取った際に、SSH を使用してインスタンスに接続することができません。

5. このインスタンスを AMI として保存します。詳細については、[Amazon EC2 ユーザーガイド](#)の「Linux インスタンス用 Amazon EC2 ユーザーガイド」を参照してください。
6. SSH を使用して接続するインスタンスごとに、1~4 のステップを繰り返します。必ず、各 SSH キーペアのコピーを作成し、関連付けられている AMI を追跡してください。
7. これで、デバイスの注文時に、これらの AMI をデバイスに追加できるようになります。

AMI をローカルで追加する

デバイスがサイトに到着したら、新しい AMI をデバイスに追加できます。手順については、「[Snow Family デバイスに仮想マシンイメージをインポートする](#)」を参照してください。すべての仮想マシンがサポートされますが、完全な機能についてテスト済みなのは、サポートされる AMI のみです。

Note

VM Import/Export を使用してデバイスに AMI を追加したり、デバイスのデプロイ後に仮想マシンをインポートしたりする際に、任意のオペレーティングシステムを使用する仮想マシンを追加できます。ただし、サポートされるオペレーティングシステムのみが Snow Family デバイスでテストおよび検証されています。お客様は、デバイスにインポートする仮想イメージに含まれるオペレーティングシステムまたはソフトウェアの利用規約を遵守する責任を負うものとします。

Important

AWS サービスが Snowball Edge で正しく機能するには、サービスのポートを許可する必要があります。詳細については、「[エッジデバイスで AWS サービス AWS Snowball を使用するために必要なポート](#)」を参照してください。

Microsoft Windows AMI の追加

サポートされている Windows オペレーティングシステムを使用する仮想マシン (VMs) の場合、VM Import/Export AWS を使用して Windows VM イメージをインポートするか、サイトにデプロイされた直後にデバイスにインポートすることで、AMI を追加できます。

Bring-Your-Own-License (BYOL)

Snowball Edge では、独自のライセンスを使用して Microsoft Windows AMI をデバイスにインポートできます。Bring Your Own License (BYOL) は、オンプレミスライセンスで所有している AMI を持ち込むプロセスです AWS。は、BYOL オプションの共有デプロイオプションと専用デプロイオプションの両方 AWS を提供します。

Windows VM イメージをデバイスに追加するには、VM Import/Export AWS を使用してインポートするか、サイトにデプロイされた直後にデバイスにインポートします。を起点とする Windows AMIs を追加することはできません AWS。したがって、Snow Family デバイスで AMI を使用する場

合は、独自の Windows VM イメージを作成してインポートし、Bring-Your-Own-License を実行する必要があります。Windows ライセンスおよび BYOL の詳細については、[Amazon Web Services and Microsoft: Frequently Asked Questions](#) を参照してください。

デバイスにインポートする Windows VM イメージの作成

Windows VM イメージを作成するには、Windows および macOS オペレーティングシステムで VirtualBox サポートされているなどの仮想化環境が必要です。Snow デバイス用の VM を作成するときは、4 GB 以上の RAM で、2 つ以上のコアを割り当てることをお勧めします。VM を起動して実行する場合は、オペレーティングシステム (Windows Server 2012、2016、2019) をインストールする必要があります。Snow Family デバイスに必要なドライバーをインストールするには、このセクションの手順に従います。

Windows AMI を Snow デバイスで実行するに

は、VirtIO、FLR、NetVCM、Vioinput、Viorng、Vioscsi、Vioserial、および VioStor ドライバーを追加する必要があります。これらのドライバーを [Windows イメージにインストールするための Microsoft Software Installer \(virtio-win-guest-tools-installer\)](#) は、[のリポジトリからダウンロードできます](#) GitHub。virtio-win-pkg-scripts

Note

VM イメージを、デプロイした Snow デバイスに直接インポートする場合は、VM イメージファイルが RAW 形式である必要があります。

Windows イメージを作成するには

1. Microsoft Windows コンピュータで、[Start] (スタート) を選択し、**devmgmt.msc**を入力して [Device Manager] (デバイスマネージャー) を開きます。
2. メインメニューで、[Actions] (アクション)、[Add legacy hardware] (レガシーハードウェアの追加) の順に選択します。
3. ウィザードで、[Next] (次へ) を選択します。
4. [Install the hardware that I manually select from a list (advanced)] (リストから手動で選択したハードウェアをインストールする (アドバンスト))、[Next] (次へ) の順に選択します。
5. [Show All Devices] (すべてのデバイスを表示)、[Next] (次へ) の順に選択します。
6. [Have Disk] (ディスクの使用) を選択し、リストから [Copy manufacturer's files] (製造元のファイルをコピーする) を開いて、ISO ファイルを検索します。

7. ISO ファイル内で Driver\W2K8R2\amd64 ディレクトリを検索し、.INF ファイルを開きます。
8. [.INF] ファイル、[Open] (開く)、[OK] の順に選択します。
9. ドライバー名が表示されたら、[Next] (次へ) を選択し、さらに 2 回、[Next] (次へ) を選択します。次に、[Finish] (終了) を選択します。

これにより、新しいドライバーを使用して、デバイスがインストールされます。実際のハードウェアは存在しないため、デバイスの問題を示す黄色の感嘆符が表示されます。この問題を解決する必要があります。

ハードウェアの問題を解決するには

1. 感嘆符が付いたデバイスのコンテキスト (右クリック) メニューを開きます。
2. [Uninstall] (アンインストール) を選択し、[Delete the driver software for this device] (このデバイスのドライバーソフトウェアを削除する) を選択解除して、[OK] を選択します。

ドライバーがインストールされ、デバイスで AMI を起動する準備が整います。

デバイスへの VM イメージのインポート

VM イメージの準備が完了したら、いずれかのオプションを使用して、イメージをデバイスにインポートできます。

- VM Import/Export を使用したクラウド内 — VM イメージを にインポート AWS して AMI として登録すると、 から注文するときにはデバイスにイメージを追加できます AWS Snow ファミリーマネジメントコンソール。詳細については、「VM Import/Export ユーザーガイド」の「[VM Import/Export を使用してイメージとして VM をインポートする](#)」を参照してください。
- サイトにデプロイされているデバイスにローカルに - VM イメージをデバイスに直接インポートするには、AWS OpsHub for Snow Family または AWS Command Line Interface () を使用します AWS CLI。

の使用の詳細については AWS OpsHub、[「Amazon EC2-compatible コンピューティングインスタンスをローカルで使用する](#)」を参照してください。

の使用については、AWS CLI「」を参照してください [Snow Family デバイスに仮想マシンイメージをインポートする](#)。

最新の Amazon Linux 2 AMI のエクスポート

Amazon Linux 2 AMIs を最新バージョンに更新するには、まず から最新の Amazon Linux 2 VM イメージをエクスポートしてから AWS Marketplace、その VM イメージを Snow デバイスにインポートします。

1. `ssm get-parameters` AWS CLI コマンドを使用して、 で Amazon Linux 2 AMI の最新のイメージ ID を検索します AWS Marketplace。

```
aws ssm get-parameters --names /aws/service/ami-amazon-linux-latest/amzn2-ami-hvm-x86_64-gp2 --query 'Parameters[0].[Value]' --region your-region
```

このコマンドは、AMI の最新のイメージ ID を返します。例えば `ami-0ccb473bada910e74` です。

2. 最新の Amazon Linux 2 イメージをエクスポートします。 [「Amazon EC2 ユーザーガイド」の「Amazon マシンイメージ \(AMI\) から直接 VM をエクスポートする」](#) を参照してください。 Amazon EC2 `ec2 export-image` コマンドの `image-id` パラメータの値として、Amazon Linux 2 AMI の最新のイメージ ID を使用します。
3. AWS CLI または を使用して、VM イメージを Snow デバイスにインポートします AWS OpsHub。
 - の使用の詳細については、AWS CLI 「」 を参照してください [Snow Family デバイスに仮想マシンイメージをインポートする](#)。
 - の使用の詳細については、AWS OpsHub 「」 を参照してください [Amazon EC2 互換 AMI としてデバイスにイメージをインポートする](#)。

Snow Family デバイスに仮想マシンイメージをインポートする

AWS CLI および VM Import/Export サービスを使用して、Amazon マシンイメージ (AMI) として Snow Family デバイスに仮想マシン (VM) イメージをインポートできます。VM イメージをインポートしたら、イメージを AMI として登録し、Amazon EC2-compatible インスタンスとして起動します。

Snow Family デバイスを注文するジョブを作成するときに、Amazon EC2 からデバイスに AMIs を追加できます。Snow Family デバイスを受け取ったら、この手順を使用します。詳細については、 [「ステップ 2: コンピューティングとストレージのオプションを選択する」](#) を参照してください。

AWS OpsHub を使用して VM イメージファイルをアップロードすることもできます。詳細については、このガイドの「[Amazon EC2-compatible AMI としてデバイスにイメージをインポートする](#)」を参照してください。

トピック

- [ステップ 1: VM イメージを準備し、Snow Family デバイスにアップロードする](#)
- [ステップ 2: 必要なアクセス許可を設定する](#)
- [ステップ 3: VM イメージをデバイス上のスナップショットとしてインポートする](#)
- [ステップ 4: スナップショットを AMI として登録する](#)
- [ステップ 5: AMI からインスタンスを起動する](#)
- [追加の AMI アクション](#)

ステップ 1: VM イメージを準備し、Snow Family デバイスにアップロードする

VM Import/Export AWS クラウド を使用して の Amazon EC2 AMI またはインスタンスから VM イメージをエクスポートするか、選択した仮想化プラットフォームを使用して VM イメージをローカルに生成して VM イメージを準備します。

VM Import/Export を使用して Amazon EC2 インスタンスを VM イメージとしてエクスポートするには、VM Import/[Export ユーザーガイドの VM Import/Export を使用してインスタンス](#)を VM としてエクスポートするを参照してください。VM Import/Export を使用して Amazon EC2 AMI を VM イメージとしてエクスポートするには、VM Import/Export [ユーザーガイドの「Amazon マシンイメージ \(AMI\) から直接 VM をエクスポートする](#)」を参照してください。

ローカル環境から VM イメージを生成する場合は、そのイメージが Snow Family デバイスの AMI として使用するよう設定されていることを確認します。環境によっては、次の項目を設定する必要がある場合があります。

- オペレーティングシステムを設定および更新します。
- ホスト名を設定します。
- ネットワークタイムプロトコル (NTP) が設定されていることを確認します。
- 必要に応じて SSH パブリックキーを含めます。キーペアのローカルコピーを作成します。詳細については、「[SSH を使用して Snowball Edge のコンピューティングインスタンスに接続する](#)」を参照してください。
- Snow Family デバイスで使用するソフトウェアをインストールして設定します。

Note

Snow Family デバイスのディスクスナップショットを準備するときは、次の制限に注意してください。

- Snow Family デバイスは現在、RAW イメージ形式のスナップショットのインポートのみをサポートしています。
- Snow Family デバイスは現在、1 GB から 1 TB までのサイズのスナップショットのインポートのみをサポートしています。

Snow Family デバイスの Amazon S3 バケットへの VM イメージのアップロード

VM イメージを準備したら、Snow Family デバイスまたはクラスターの S3 バケットにアップロードします。Snow Family デバイスの S3 アダプターまたは Amazon S3 互換ストレージを使用して、スナップショットをアップロードできます。

S3 アダプターを使用して仮想マシンイメージをアップロードするには

- `cp` コマンドを使用して、VM イメージファイルをデバイスのバケットにコピーします。

```
aws s3 cp image-path s3://S3-bucket-name --endpoint http://S3-object-API-endpoint:443 --profile profile-name
```

詳細については、このガイドの「[サポートされている AWS CLI コマンド](#)」を参照してください。

Snow Family デバイスで Amazon S3 互換ストレージを使用して VM イメージをアップロードするには

- `put-object` コマンドを使用して、スナップショットファイルをデバイスのバケットにコピーします。

```
aws s3api put-object --bucket bucket-name --key path-to-snapshot-file --body snapshot-file --profile your-profile --endpoint-url s3api-endpoint-ip
```

詳細については、「[Snowball Edge デバイスの S3 オブジェクトの使用](#)」を参照してください。

ステップ 2: 必要なアクセス許可を設定する

インポートを成功させるには、Snow Family デバイス、Amazon EC2、およびユーザーに VM Import/Export のアクセス許可を設定する必要があります。

Note

これらのアクセス許可を提供するサービスロールとポリシーは、Snow Family デバイスにあります。

VM Import/Export に必要なアクセス許可

インポートプロセスを開始する前に、Snow Family デバイスで VM Import/Export がロールを引き受けることを許可する信頼ポリシーを持つ IAM ロールを作成する必要があります。デバイス上の VM Import/Export がデバイスの S3 バケットに保存されているイメージにアクセスできるようにする追加のアクセス許可がロールに付与されます。

信頼ポリシー JSON ファイルを作成します。

次に、VM Import/Export が S3 バケットからインポートする必要があるスナップショットにアクセスするために、ロールにアタッチする必要がある信頼ポリシーの例を示します。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "vmie.amazonaws.com"
      },
      "Action": "sts:AssumeRole"
    }
  ]
}
```

信頼ポリシー JSON ファイルを使用してロールを作成します。

ロール名は `vmimport` とすることができます。コマンドで `--role-name` オプションを使って、これを変更できます。

```
aws iam create-role --role-name role-name --assume-role-policy-document file:///trust-policy-json-path --profile profile-name --endpoint http://snowball-ip:6078 --region snow
```

`create-role` コマンドからの出力例を次に示します。

```
{
  "Role": {
    "AssumeRolePolicyDocument": {
      "Version": "2012-10-17",
      "Statement": [
        {
          "Action": "sts:AssumeRole",
          "Effect": "Allow",
          "Principal": {
            "Service": "vmie.amazonaws.com"
          }
        }
      ]
    },
    "MaxSessionDuration": 3600,
    "RoleId": "AROACEMGEZDGNBVG3TQ0JQGEZAAAABQBB6NSGNAAAABPSVLTREPY3FPAFOLKJ3",
    "CreateDate": "2022-04-19T22:17:19.823Z",
    "RoleName": "vmimport",
    "Path": "/",
    "Arn": "arn:aws:iam::123456789012:role/vmimport"
  }
}
```

ロールのポリシーを作成します。

次の例のポリシーは、Amazon S3 へのアクセスに必要な最小限のアクセス許可を備えています。Amazon S3 バケット名を、イメージが保存されているバケット名に変更します。スタンドアロンの Snowball Edge デバイスの場合は、`snow-id` をジョブ ID に変更します。デバイスのクラスターの場合は、`snow-id` をクラスター ID に変更します。プレフィックスを使用して、VM Import/Export でスナップショットをインポートできる場所をさらに絞り込むことができます。このようなポリシー JSON ファイルを作成します。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "s3:GetBucketLocation",
        "s3:GetObject",
        "s3:ListBucket",
        "s3:GetMetadata"
      ],
      "Resource": [
        "arn:aws:s3:snow:account-id:snow/snow-id/bucket/import-snapshot-bucket-name",
        "arn:aws:s3:snow:account-id:snow/snow-id/bucket/import-snapshot-bucket-name/*"
      ]
    }
  ]
}
```

ポリシーファイルを使用してポリシーを作成します。

```
aws iam create-policy --policy-name policy-name --policy-document file:///policy-json-file-path --profile profile-name --endpoint http://snowball-ip:6078 --region snow
```

create-policy コマンドからの出力例を次に示します。

```
{
  "Policy": {
    "PolicyName": "vmimport-resource-policy",
    "PolicyId": "ANPACEMGEZDGNBVGY3TQ0JQGEZAAAAB00EE3IIHAAAABWZJPI2VW4UUTFEDBC2R",
    "Arn": "arn:aws:iam::123456789012:policy/vmimport-resource-policy",
    "Path": "/",
    "DefaultVersionId": "v1",
    "AttachmentCount": 0,
    "IsAttachable": true,
    "CreateDate": "2020-07-25T23:27:35.690000+00:00",
    "UpdateDate": "2020-07-25T23:27:35.690000+00:00"
  }
}
```

ポリシーをロールにアタッチします。

前述のロールにポリシーをアタッチし、必要なリソースにアクセスする許可を付与します。これにより、ローカル VM Import/Export サービスが、デバイスの Amazon S3 からスナップショットをダウンロードできるようになります。

```
aws iam attach-role-policy --role-name role-name --policy-arn
arn:aws:iam::123456789012:policy/policy-name --profile profile-name --endpoint
http://snowball-ip:6078 --region snow
```

発信者が必要とするアクセス許可

Snowball Edge VM Import/Export が引き受けるロールに加えて、ロールを VMIE に渡す許可がユーザーに付与されていることも確認する必要があります。デフォルトのルートユーザーを使用してインポートを実行する場合、ルートユーザーに必要なすべてのアクセス許可は既に存在するため、このステップをスキップしてステップ 3 に進むことができます。

インポートを実行するユーザーに、次の 2 つの IAM アクセス許可をアタッチします。

- `pass-role`
- `get-role`

ロールのポリシーを作成します。

以下は、IAM ロールに対する `get-role` および `pass-role` のアクションの実行を、ユーザーに許可するポリシーの例です。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "iam:GetRole",
      "Resource": "*"
    },
    {
      "Sid": "iamPassRole",
      "Effect": "Allow",
      "Action": "iam:PassRole",
      "Resource": "*",
      "Condition": {
        "StringEquals": {
          "iam:PassedToService": "importexport.amazonaws.com"
        }
      }
    }
  ]
}
```



```

    }
  }
}
}
}

```

ポリシーファイルを使用してポリシーを作成します。

```
aws iam create-policy --policy-name policy-name --policy-document file:///policy-json-  
file-path --profile profile-name --endpoint http://snowball-ip:6078 --region snow
```

create-policy コマンドからの出力例を次に示します。

```
{
  "Policy":{
    "PolicyName":"caller-policy",
    "PolicyId":"ANPACEMGEZDGNBVG3TQ0JQGEZAAAAB000TU0E3AAAAAAPPBEUM7Q7ARPUE53C6R",
    "Arn":"arn:aws:iam::123456789012:policy/caller-policy",
    "Path":"/",
    "DefaultVersionId":"v1",
    "AttachmentCount":0,
    "IsAttachable":true,
    "CreateDate":"2020-07-30T00:58:25.309000+00:00",
    "UpdateDate":"2020-07-30T00:58:25.309000+00:00"
  }
}
```

ポリシーの生成後、Amazon EC2 API または CLI オペレーションを呼び出してスナップショットをインポートする IAM ユーザーに、ポリシーをアタッチします。

```
aws iam attach-user-policy --user-name your-user-name --policy-arn  
arn:aws:iam::123456789012:policy/policy-name --profile profile-name --endpoint  
http://snowball-ip:6078 --region snow
```

デバイスで Amazon EC2 APIs呼び出すために必要なアクセス許可

スナップショットをインポートするには、IAM ユーザーに `ec2:ImportSnapshot` のアクセス許可が必要です。ユーザーのアクセスを制限する必要がない場合は、`ec2:*` のアクセス許可を使用して、Amazon EC2 のフルアクセス許可を付与します。次に、デバイス上の Amazon EC2 用に付与または制限できるアクセス許可を示します。以下の内容を含むポリシーファイルを作成します。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "ec2:ImportSnapshot",
        "ec2:DescribeImportSnapshotTasks",
        "ec2:CancelImportTask",
        "ec2:DescribeSnapshots",
        "ec2>DeleteSnapshot",
        "ec2:RegisterImage",
        "ec2:DescribeImages",
        "ec2:DeregisterImage"
      ],
      "Resource": "*"
    }
  ]
}
```

ポリシーファイルを使用してポリシーを作成します。

```
aws iam create-policy --policy-name policy-name --policy-document file:///policy-json-  
file-path --profile profile-name --endpoint http://snowball-ip:6078 --region snow
```

create-policy コマンドからの出力例を次に示します。

```
{
  "Policy": {
    "PolicyName": "ec2-import.json",
    "PolicyId": "ANPACEMGEZDGNBVG3TQ0JQGEZAAAAABQBGPDQC5AAAAATYN62UNBFYTF5WVCSCZS",
    "Arn": "arn:aws:iam::123456789012:policy/ec2-import.json",
    "Path": "/",
    "DefaultVersionId": "v1",
    "AttachmentCount": 0,
    "IsAttachable": true,
    "CreateDate": "2022-04-21T16:25:53.504000+00:00",
    "UpdateDate": "2022-04-21T16:25:53.504000+00:00"
  }
}
```

ポリシーの生成後、Amazon EC2 API または CLI オペレーションを呼び出してスナップショットをインポートする IAM ユーザーに、ポリシーをアタッチします。

```
aws iam attach-user-policy --user-name your-user-name --policy-arn
arn:aws:iam::123456789012:policy/policy-name --profile profile-name --endpoint
http://snowball-ip:6078 --region snow
```

ステップ 3: VM イメージをデバイス上のスナップショットとしてインポートする

次のステップでは、VM イメージをスナップショットとしてデバイスにインポートします。S3Bucket パラメータの値は、VM イメージを含むバケットの名前です。S3Key パラメータの値は、このバケット内の VM イメージファイルへのパスです。

```
aws ec2 import-snapshot --disk-container "Format=RAW,UserBucket={S3Bucket=bucket-name,S3Key=image-file}" --profile profile-name --endpoint http://snowball-ip:8008 --
region snow
```

詳細については、AWS CLI 「コマンドリファレンス」の「[import-snapshot](#)」を参照してください。

このコマンドでは、以下のスイッチはサポートされません。

- [--client-data value]
- [--client-token value]
- [--dry-run]
- [--no-dry-run]
- [--encrypted]
- [--no-encrypted]
- [--kms-key-id value]
- [--tag-specifications value]

Example **import-snapshot** コマンドの出力

```
{
  "ImportTaskId": "s.import-snap-1234567890abc",
  "SnapshotTaskDetail": {
    "DiskImageSize": 2.0,
    "Encrypted": false,
```

```
"Format": "RAW",
"Progress": "3",
"Status": "active",
"StatusMessage": "pending",
"UserBucket": {
  "S3Bucket": "bucket",
  "S3Key": "vmimport/image01"
}
}
```

Note

Snow Family デバイスでは、現在、デバイスごとに一度に実行できるアクティブなインポートジョブは 1 つだけです。新しいインポートタスクを開始するには、現在のタスクが終了するのを待つか、クラスター内の別の使用可能なノードを選択します。必要に応じて、現在のインポートをキャンセルすることもできます。遅延を防ぐため、インポート中は Snow Family デバイスを再起動しないでください。デバイスを再起動すると、インポートは失敗し、デバイスがアクセス可能になると進行状況が削除されます。スナップショットインポートタスクステータスのステータスをチェックするには、次のコマンドを使用します。

```
aws ec2 describe-import-snapshot-tasks --import-task-ids id --profile profile-name --endpoint http://snowball-ip:8008 --region snow
```

ステップ 4: スナップショットを AMI として登録する

デバイスへのスナップショットのインポートが成功したら、`register-image` コマンドを使用して登録できます。

Note

AMI を登録できるのは、そのすべてのスナップショットが利用可能な場合のみです。

詳細については、コマンドリファレンスの「[register-image](#) AWS CLI」を参照してください。

Example **register-image** コマンドの

```
aws ec2 register-image \  

```

```
--name ami-01 \  
--description my-ami-01 \  
--block-device-mappings "[{\\"DeviceName\\": \"/dev/sda1\\",\\"Ebs\\":{\\"Encrypted\\":false, \  
\\"DeleteOnTermination\\":true,\\"SnapshotId\\":\\"snapshot-id\\",\\"VolumeSize\\":30}}]" \  
--root-device-name /dev/sda1 \  
--profile profile-name \  
--endpoint http://snowball-ip:8008 \  
--region snow
```

ブロックデバイスマッピング JSON の例を以下に示します。詳細については、AWS CLI コマンドリファレンス [block-device-mapping](#) の [register-image](#) の [パラメータ](#) を参照してください。

```
[  
  {  
    "DeviceName": "/dev/sda",  
    "Ebs":  
      {  
        "Encrypted": false,  
        "DeleteOnTermination": true,  
        "SnapshotId": "snapshot-id",  
        "VolumeSize": 30  
      }  
  }  
]
```

Example **register-image** コマンドの

```
{  
  "ImageId": "s.ami-8de47d2e397937318"  
}
```

ステップ 5: AMI からインスタンスを起動する

インスタンスを起動するには、AWS CLI コマンドリファレンスの「[run-instances](#)」を参照してください。

image-id パラメータの値は、register-image コマンドの出力として ImageId の名前の値です。

```
aws ec2 run-instances --image-id image-id --instance-type instance-type --  
profile profile-name --endpoint http://snowball-ip:8008 --region snow
```

```
{
  "Instances": [
    {
      "SourceDestCheck": false,
      "CpuOptions": {
        "CoreCount": 1,
        "ThreadsPerCore": 2
      },
      "InstanceId": "s.i-12345a73123456d1",
      "EnaSupport": false,
      "ImageId": "s.ami-1234567890abcdefg",
      "State": {
        "Code": 0,
        "Name": "pending"
      },
      "EbsOptimized": false,
      "SecurityGroups": [
        {
          "GroupName": "default",
          "GroupId": "s.sg-1234567890abc"
        }
      ],
      "RootDeviceName": "/dev/sda1",
      "AmiLaunchIndex": 0,
      "InstanceType": "sbe-c.large"
    }
  ],
  "ReservationId": "s.r-1234567890abc"
}
```

Note

AWS OpsHub を使用してインスタンスを起動することもできます。詳細については、このガイドの [「Amazon EC2-compatible インスタンスの起動」](#) を参照してください。

追加の AMI アクション

追加の AWS CLI コマンドを使用して、スナップショットのインポートステータスのモニタリング、インポートされたスナップショットの詳細の取得、スナップショットのインポートのキャンセル、インポート後のスナップショットの削除または登録解除を行うことができます。

スナップショットのインポートステータスのモニタリング

インポートの進行状況の現在の状態を確認するには、Amazon EC2 `describe-import-snapshot-tasks` コマンドを実行します。このコマンドは、でのページ分割とフィルタリングをサポートします `task-state`。

Example `describe-import-snapshot-tasks` コマンドの

```
aws ec2 describe-import-snapshot-tasks --import-task-ids id --profile profile-name --  
endpoint http://snowball-ip:8008 --region snow
```

Example `describe-import-snapshot-tasks` コマンド出力の

```
{  
  "ImportSnapshotTasks": [  
    {  
      "ImportTaskId": "s.import-snap-8f6bfd7fc9ead9aca",  
      "SnapshotTaskDetail": {  
        "Description": "Created by AWS-Snowball-VMImport service for  
s.import-snap-8f6bfd7fc9ead9aca",  
        "DiskImageSize": 8.0,  
        "Encrypted": false,  
        "Format": "RAW",  
        "Progress": "3",  
        "SnapshotId": "s.snap-848a22d7518ad442b",  
        "Status": "active",  
        "StatusMessage": "pending",  
        "UserBucket": {  
          "S3Bucket": "bucket1",  
          "S3Key": "image1"  
        }  
      }  
    }  
  ]  
}
```

Note

このコマンドは、過去 7 日以内に正常に完了したタスク、または削除済みとしてマークされたタスクの出力のみを表示します。フィルタリングは `Name=task-`

state、Values=active | deleting | deleted | completed のみをサポートします

このコマンドは、以下のパラメータをサポートしていません。

- [--dry-run]
- [--no-dry-run]

インポートタスクのキャンセル

インポートタスクをキャンセルするには、cancel-import-task コマンドを実行します。

Example **cancel-import-task** コマンドの

```
aws ec2 cancel-import-task --import-task-id import-task-id --profile profile-name --  
endpoint http://snowball-ip:8008 --region snow
```

Example **cancel-import-task** コマンド出力の

```
{  
  "ImportTaskId": "s.import-snap-8234ef2a01cc3b0c6",  
  "PreviousState": "active",  
  "State": "deleting"  
}
```

Note

完了状態でないタスクのみをキャンセルできます。

このコマンドは、以下のパラメータをサポートしていません。

- [--dry-run]
- [--no-dry-run]

スナップショットの説明

スナップショットのインポート後、このコマンドを使用してスナップショットを説明できます。スナップショットをフィルタリングするには、前回のインポートタスクレスポンスからのスナップショット ID に `snapshot-ids` を渡します。このコマンドは、`start-time`、`volume-idstatus` とフィルターをサポートします。

Example `describe-snapshots` コマンドの

```
aws ec2 describe-snapshots --snapshot-ids snapshot-id --profile profile-name --endpoint http://snowball-ip:8008 --region snow
```

Example `describe-snapshots` コマンド出力の

```
{
  "Snapshots": [
    {
      "Description": "Created by AWS-Snowball-VMImport service for s.import-snap-8f6bfd7fc9ead9aca",
      "Encrypted": false,
      "OwnerId": "123456789012",
      "SnapshotId": "s.snap-848a22d7518ad442b",
      "StartTime": "2020-07-30T04:31:05.032000+00:00",
      "State": "completed",
      "VolumeSize": 8
    }
  ]
}
```

このコマンドは、以下のパラメータをサポートしていません。

- `[--restorable-by-user-ids value]`
- `[--dry-run]`
- `[--no-dry-run]`

Snow Family デバイスからのスナップショットの削除

所有していて不要になったスナップショットを削除するには、`delete-snapshot` コマンドを使用します。

Example `delete-snapshot` コマンドの

```
aws ec2 delete-snapshot --snapshot-id snapshot-id --profile profile-name --endpoint
http://snowball-ip:8008 --region snow
```

Note

Snowball Edge では、[PENDING] (保留) 状態、または AMI のルートデバイスとして指定されたスナップショットの削除はサポートされません。

このコマンドは、以下のパラメータをサポートしていません。

- [--dry-run]
- [--no-dry-run]

AMI の登録解除

不要になったAMI の登録を解除するには、`deregister-image` コマンドを実行します。現在、[PENDING] (保留) 状態の AMI の登録解除はサポートされていません。

Example `deregister-image` コマンドの

```
aws ec2 deregister-image --image-id image-id --profile profile-name --endpoint
http://snowball-ip:8008 --region snow
```

このコマンドは、以下のパラメータをサポートしていません。

- [--dry-run]
- [--no-dry-run]

Snowball Edge での AWS CLI および API オペレーションの使用

AWS Command Line Interface (AWS CLI) または API オペレーションを使用して Snowball Edge で IAM、Amazon S3、および Amazon EC2 コマンドを発行する場合は、`region` 「」と指定する必要があります snow。これは、次の例のように、コマンド自体で `AWS configure` または `aws configure` を使用して実行できます。

```
aws configure --profile ProfileName
AWS Access Key ID [None]: defgh
AWS Secret Access Key [None]: 1234567
Default region name [None]: snow
Default output format [None]: json
```

または

```
aws s3 ls --profile ProfileName --endpoint http://192.0.2.0:8080 --region snow
```

Snowball Edge デバイスのコンピューティングインスタンスのクォータ

以下は、AWS Snowball Edge デバイスのコンピューティングリソースのストレージクォータと共有リソースの制限です。

ストレージクォータ

コンピューティングリソースで使用できるストレージは、Snowball Edge デバイスの Amazon S3 専用ストレージとは別のリソースです。ストレージのクォータは次のとおりです。

- Snowball Edge Storage Optimized オプション用ストレージクォータ – Amazon S3 で利用できる合計ストレージは、デバイスでコンピューティングインスタンスを使用しているかどうかに応じて、60 TB ~ 80 TB となります。sbe1 コンピューティングインスタンスを使用している場合は、Snowball Edge Storage Optimized オプション用の使用可能な専用ストレージの合計は、1,000 GB です。
- Snowball Edge Compute Optimized および GPU オプションのストレージクォータ – sbe-c および sbe-g インスタンス用の使用可能な専用ストレージの合計は、7.68 TB です。残りの使用可能なストレージの合計は 42 TB です。

次の表は、Snowball Edge デバイスの使用可能なコンピューティングリソースの概要を示しています。

機能	制限
単一の Snowball Edge Storage Optimized オプションの AMI の数	10

機能	制限
単一の Snowball Edge Compute Optimized オプションの AMI の数	20
GPU オプション付きの単一の Snowball Edge Compute Optimized の AMI の数	20
インスタンスあたりのボリュームの数	10
現在実行中 (または停止した) インスタンス	利用可能なリソースによって異なる

インスタンスタイプ	vCPU コア	メモリ (GiB)	GPU	サポートされているデバイスオプション
sbe1.small	1	1	0	ストレージの最適化
sbe1.medium	1	2	0	ストレージの最適化
sbe1.large	2	4	0	ストレージの最適化
sbe1.xlarge	4	8	0	ストレージの最適化
sbe1.2xlarge	8	16	0	ストレージの最適化
sbe1.4xlarge	16	32	0	ストレージの最適化
sbe1.6xlarge	24	32	0	ストレージの最適化
sbe-c.small	1	2	0	コンピューティングの最適化
sbe-c.medium	1	4	0	コンピューティングの最適化
sbe-c.large	2	8	0	コンピューティングの最適化

インスタンスタイプ	vCPU コア	メモリ (GiB)	GPU	サポートされているデ バイスオプション
sbe-c.xlarge	4	16	0	コンピューティングの 最適化
sbe-c.2xlarge	8	32	0	コンピューティングの 最適化
sbe-c.4xlarge	16	64	0	コンピューティングの 最適化
sbe-c.8xlarge	32	128	0	コンピューティングの 最適化
sbe-c.12xlarge	48	192	0	コンピューティングの 最適化
sbe-c.16xlarge	64	256	0	コンピューティングの 最適化
sbe-c.24xlarge	96	384	0	コンピューティングの 最適化
sbe-g.small	1	2	1	GPU を使用
sbe-g.medium	1	4	1	GPU を使用
sbe-g.large	2	8	1	GPU を使用
sbe-g.xlarge	4	16	1	GPU を使用
sbe-g.2xlarge	8	32	1	GPU を使用
sbe-g.4xlarge	16	64	1	GPU を使用
sbe-g.8xlarge	32	128	1	GPU を使用
sbe-g.12xlarge	48	192	1	GPU を使用

共有コンピューティングリソースの制限

Snowball Edge デバイスのすべてのサービスは、デバイスの有限リソースの一部を使用します。使用可能なコンピューティングリソースが最大化された Snowball Edge デバイスは、新しいコンピューティングリソースを起動できません。例えば、ストレージ最適化デバイスで sbe1.4xlarge コンピューティングインスタンスの実行中に NFS インターフェイスを起動しようとしても、NFS インターフェイスサービスは起動しません。別のデバイスオプションで利用可能なリソースの概要および各サービスのリソース要件を次に示します。

- いずれのコンピューティングサービスも ACTIVE ではない場合。
 - ストレージの最適化オプションでは、コンピューティングインスタンス用に 24 個の vCPU と 32 GiB のメモリがあります。
 - コンピューティングの最適化オプションでは、コンピューティングインスタンス用に 52 個の vCPU と 208 GiB のメモリがあります。これは、GPU オプションを使用した場合にも同様です。
- AWS IoT Greengrass と AWS Lambda を搭載した AWS IoT Greengrass は **です**ACTIVE。
 - ストレージの最適化オプションでは、これらのサービスは 4 個の vCPU コアと 8 GiB のメモリを使用します。
 - コンピューティングの最適化オプションでは、これらのサービスは 1 個の vCPU コアと 1 GiB のメモリを使用します。これは、GPU オプションを使用した場合にも同様です。
 - NFS インターフェイスが ACTIVE の場合、Snowball Edge デバイスで 8 個の vCPU コアと 16 GiB のメモリを使用します。
- Snow Family デバイスの Amazon S3 互換ストレージがアクティブの場合：
 - AMD EPYC Gen2 と NVME 搭載の Snowball Edge Compute Optimized では、Snow Family デバイス上の、最小構成の 3 TB の Amazon S3 互換ストレージを備えた単一ノードで 8 個の vCPU コアと 16 GB のメモリを使用します。3 TB を超える Snow Family デバイスの Amazon S3 互換ストレージを備えた単一ノードでは、20 個の vCPU コアと 40 GB のメモリを使用します。クラスターの場合、20 個の vCPU コアと 40 GB のメモリを使用します。
 - AMD EPYC Gen1、HDD、およびオプションの GPU 搭載の Snowball Edge Compute Optimized では、単一ノードで 8 個の vCPU コアと 16 GB のメモリを使用します。クラスターの場合、20 個の vCPU コアと 40 GB のメモリを使用します。

Snowball Edge のサービスが ACTIVE であるかどうかを確認するには、Snowball Edge クライアントでコマンド `snowballEdge describe-service` を使用します。詳細については、「[サービスステータスの取得](#)」を参照してください。

コンピューティングジョブの作成

このセクションでは、AWS Snowball Edge デバイスの、最初の Amazon EC2 互換コンピューティングインスタンスジョブを作成します。

Important

ジョブを作成する前に以下の点に注意してください。

- AMI に関連付けられている vCPU、メモリ、ストレージの値が、作成するインスタンスタイプと一致していることを確認します。
- Snowball Edge でインスタンスの起動後、Secure Shell (SSH) を使用してインスタンスに接続する場合は、まず以下の手順を実行する必要があります。事後に Snowball Edge 上の AMI を更新することはできません。このステップは、ジョブの作成前に行う必要があります。

AMI を設定して SSH を使用し、デバイスで起動するコンピューティングインスタンスに接続する

Secure Shell (SSH) を使用して Snowball Edge デバイスのコンピューティングインスタンスに接続するには、次の手順を実行する必要があります。この手順では、ジョブを作成する前に AMI に SSH キーを追加します。また、この手順を使用して、ジョブの AMI として使用する予定のインスタンスのアプリケーションをセットアップすることをお勧めします。

Important

この手順を行わないと、Snowball Edge デバイスを受け取った際に、SSH を使用してインスタンスに接続することができません。

SSH キーを AMI に追加するには

1. [CentOS 7 \(x86_64\) - with Updates HVM](https://docs.aws.amazon.com/snowball/latest/developer-guide/using-ami.html)、[Ubuntu 16.04 LTS - Xenial \(HVM\)](https://docs.aws.amazon.com/snowball/latest/developer-guide/using-ami.html)、Amazon Linux 2 AMI イメージ、または Windows AWS クラウドに基づいて、新しいインスタンスを起動します。<https://docs.aws.amazon.com/snowball/latest/developer-guide/using-ami.html>
<https://aws.amazon.com/about-aws/whats-new/2020/11/aws-snowball-edge-supports-windows-operating-systems/>

インスタンスを起動する際、インスタンスに割り当てるストレージサイズが、後ほど Snowball Edge で使用するために適切かどうかを確認します。確認するには、Amazon EC2 コンソールの [Step 4: Add Storage] (ステップ 4: ストレージの追加) を実行します。Snowball Edge のコンピューティングインスタンスストレージのボリュームでサポートされるサイズのリストについては、[Snowball Edge デバイスのコンピューティングインスタンスのクォータ](#) を参照してください。

2. Snowball Edge で実行するアプリケーションをインストールおよび設定し、正常に動作することを検証します。
3. このインスタンスを作成するには、SSH キーペアで使用する PEM/PPK ファイルのコピーを作成します。Snowball Edge との通信に使用する予定のサーバーに、このファイルを保存します。このファイルは、SSH を使用して、デバイス上の起動されたインスタンスへの接続に必要なため、このファイルへのパスをメモします。
4. このインスタンスを AMI として保存します。詳細については、「[Amazon EC2 ユーザーガイド](#)」の「[Amazon EBS-Backed Linux AMI](#) の作成」を参照してください。Amazon EC2
5. SSH を使用して接続するインスタンスごとに、この手順を繰り返します。別々の SSH キーペアをコピーしたことを確認し、関連付けられた AMI を書き留めます。

コンソールでジョブを作成する

次のステップでは、Snow Family デバイスを注文するジョブを作成します。ジョブは任意のタイプ (例: クラスター) で作成できます。を使用して[AWS Snow ファミリーマネジメントコンソール](#)、「[Snow Family デバイスを注文するジョブの作成](#)」に記載されている手順に従ってください。ジョブ作成ウィザードの [Step 3: Give job details] (ステップ 3: ジョブの詳細の入力) ページまで進んだら、次の追加ステップを実行します。

1. [Enable compute with EC2 (EC2 でコンピューティングを有効にする)] を選択します。
2. [Add an AMI (AMI の追加)] を選択します。
3. 表示されたダイアログボックスで AMI を選択し、[Save] (保存) を選択します。
4. デバイスタイプに応じて、最大 20 AMIs をジョブに追加します。
5. 通常どおりにジョブの作成を続行します。

でのジョブの作成 AWS CLI

ジョブは、AWS CLIを使って作成することもできます。これを行うには、ターミナルを開いて次のコマンドを実行します。その際、赤色の文字を実際の値に置き換えます。

```
aws snowball create-job --job-type IMPORT --resources '{"S3Resources": [{"BucketArn": "arn:aws:s3:::bucket-name"}], "Ec2AmiResources": [{"AmiId": "ami-12345678"}]}' --description Example --address-id ADIEXAMPLE60-1234-1234-5678-41fEXAMPLE57 --kms-key-arn arn:aws:kms:us-west-2:012345678901:key/eEXAMPLE-1234-1234-5678-5b4EXAMPLE8e --role-arn arn:aws:iam::012345678901:role/snowball-local-s3-lambda-us-west-2-role --snowball-capacity-preference T100 --shipping-option SECOND_DAY --snowball-type EDGE
```

これを受信し、デバイスのロックを解除したら、Snowball Edge クライアントを使用してローカル認証情報を取得します。詳細については、「[認証情報の取得](#)」を参照してください。

コンピューティングインスタンスのネットワーク設定

Snow Family デバイスでコンピューティングインスタンスを起動した後、ネットワークインターフェイスを作成して IP アドレスを指定する必要があります。Snow Family デバイスは、仮想ネットワークインターフェイスとダイレクトネットワークインターフェイスの 2 種類のネットワークインターフェイスをサポートします。

仮想ネットワークインターフェイス (VNI)

仮想ネットワークインターフェイスは、Snow Family デバイス上の EC2 互換インスタンスに接続するための標準ネットワークインターフェイスです。ダイレクトネットワークインターフェイスを使用するかどうかにかかわらず、各 EC2 互換インスタンスに対して VNI を作成する必要があります。VNI を通過するトラフィックは、設定したセキュリティグループによって保護されます。VNI は、Snow Family デバイスの制御に使用する物理ネットワークポートにのみ関連付けることができます。

Note

VNI は Snow Family デバイスの管理に使用されているものと同じ物理インターフェイス (RJ45、SFP+、または QSFP) を使用します。デバイス管理に使用しているものとは異なる物理インターフェイスで VNI を作成すると、予期しない結果が生じる可能性があります。

ダイレクトネットワークインターフェイス (DNI)

ダイレクトネットワークインターフェイス (DNI) は、マルチキャストストリーム、推移的ルーティング、ロードバランシングなどのユースケースを可能にする高度なネットワーク機能です。中間変換やフィルタリングなしでインスタンスにレイヤー 2 ネットワークアクセスを提供することで、Snow Family デバイスのネットワーク構成に対する柔軟性を高め、ネットワークパフォーマンスを向上させることができます。DNI は VLAN タグと MAC アドレスのカスタマイズをサポートします。DNI 上のトラフィックは、セキュリティグループによって保護されません。

Snowball Edge デバイスでは、DNI を RJ45、SFP、または QSFP ポートに関連付けることができます。各物理ポートは、最大 63 個の DNIs をサポートします。DNIs は、Snow Family デバイスの管理に使用するのと同じ物理ネットワークポートに関連付ける必要はありません。

Note

Snowball Edge ストレージ最適化 (EC2 コンピューティング機能搭載) デバイスは DNI をサポートしません。

トピック

- [前提条件](#)
- [仮想ネットワークインターフェイス \(VNI\) の設定](#)
- [ダイレクトネットワークインターフェイス \(DNI\) の設定](#)

前提条件

VNI または DNI を設定する前に、次の前提条件が満たされていることを確認します。

1. デバイスの電源がオンになっており、物理的なネットワークインターフェイス (RJ45 ポートなど) のいずれかが IP アドレスと接続されていることを確認します。
2. Snow Family デバイスで使用している物理ネットワークインターフェイスに関連付けられている IP アドレスを取得します。
3. Snowball Edge クライアントを設定します。詳細については、[Snowball Edge クライアント用プロフィールの設定](#)を参照してください。
4. デバイスのロックを解除します。AWS OpsHub for Snow Family を使用してデバイスのロックを解除することをお勧めします。手順については、[デバイスのロック解除](#)を参照してください。

CLI コマンドを使用する場合は、次のコマンドを実行し、ダイアログボックスに表示される情報を入力します。

```
snowballEdge configure
```

Snowball Edge Manifest Path: `manifest.bin`

Unlock Code: `unlock code`

Default Endpoint: `https://device ip`

5. 以下のコマンドを実行します。

```
snowballEdge unlock-device
```

デバイスディスプレイの更新は、デバイスのロックが解除されていることを示します。

6. デバイス上の EC2 互換インスタンスを起動します。VNI をこのインスタンスに関連付けます。
7. `snowballEdge describe-device` コマンドを実行して、物理ネットワークインターフェイス ID のリストを取得します。
8. 使用する物理ネットワークインターフェイスの ID を特定し、これを書き留めます。

仮想ネットワークインターフェイス (VNI) の設定

物理ネットワークインターフェイス ID の識別後に、仮想ネットワークインターフェイス (VNI) を設定できます。次の手順に従って、VNI を設定します。VNI を作成する前に、前提条件となるタスクを実行していることを確認してください。

VNI を作成し、IP アドレスを関連付ける

1. `snowballEdge create-virtual-network-interface` コマンドを実行します。以下の例では、このコマンドの実行に 2 つの異なる IP アドレス割り当て方法 (DHCP、STATIC) を使用しています。DHCP メソッドは Dynamic Host Configuration Protocol (DHCP) を使用します。

```
snowballEdge create-virtual-network-interface \  
--physical-network-interface-id s.ni-abcd1234 \  
--ip-address-assignment DHCP
```

//OR//

```
snowballEdge create-virtual-network-interface \  
--physical-network-interface-id s.ni-abcd1234 \  
--ip-address-assignment STATIC \  
--static-ip-address-configuration IpAddress=192.0.2.0,Netmask=255.255.255.0
```

このコマンドによって、IP アドレスを含む JSON 構造が返ります。プロセスの後半で `ec2 associate-address` AWS CLI、コマンドの IP アドレスを書き留めます。

この IP アドレスが必要なときはいつでも、`snowballEdge describe-virtual-network-interfaces` Snowball Edge クライアントコマンド、または `aws ec2 describe-addresses` AWS CLI コマンドを使用して取得できます。

2. 新しく作成した IP アドレスをインスタンスに関連付けるには、次のコマンドを使用し、赤色のテキストを値に置き換えます。

```
aws ec2 associate-address --public-ip 192.0.2.0 --instance-id s.i-01234567890123456  
--endpoint http://Snow Family device physical IP address:8008
```

ダイレクトネットワークインターフェイス (DNI) の設定

Note

ダイレクトネットワークインターフェイス機能は、2021 年 1 月 12 日以降に利用でき、AWS リージョン Snow Family デバイスが利用可能なすべてので使用できます。

前提条件

ダイレクトネットワークインターフェイス (DNI) を設定する前に、前提条件セクションのタスクを実行する必要があります。

1. DNI を設定する前に、前提条件となるタスクを実行します。手順については、「[前提条件](#)」を参照してください。
2. さらに、デバイスでインスタンスを起動し、VNI を作成してインスタンスに関連付ける必要があります。手順については、「[仮想ネットワークインターフェイス \(VNI\) の設定](#)」を参照してください。

Note

in-the-field ソフトウェア更新を実行して既存のデバイスに直接ネットワークを追加した場合は、デバイスを完全に有効にするためにデバイスを 2 回再起動する必要があります。

DNI を作成し、IP アドレスを関連付ける

1. 次のコマンドを実行して、ダイレクトネットワークインターフェイスを作成し、Amazon EC2 互換インスタンスにアタッチします。次のステップでは、デバイスの MAC アドレスが必要です。

```
create-direct-network-interface [--endpoint endpoint] [--instance-id instanceId]
  [--mac macAddress]
  [--physical-network-interface-
id physicalNetworkInterfaceId]
  [--unlock-code unlockCode] [--vlan vlanId]
```

OPTIONS

--endpoint <endpoint> このリクエストを送信するエンドポイント。デバイスのエンドポイントは、https スキームの後に IP アドレスが続く URL です。例えば、デバイスの IP アドレスが 123.0.1.2 の場合、デバイスのエンドポイントは https://123.0.1.2 になります。

--instance-id <instanceId> インターフェイスをアタッチする EC2 互換インスタンス ID (オプション)。

--mac <macAddress> ネットワークインターフェイスの MAC アドレスを設定します (オプション)。

--physical-network-interface-id <physicalNetworkInterfaceId> 新しい仮想ネットワークインターフェイスを作成する物理ネットワークインターフェイスの ID。describe-device コマンドを使用して、Snowball Edge で使用可能な物理ネットワークインターフェイスを決定できます。

--vlan <vlanId> インターフェイスに、割り当てられた VLAN を設定します (オプション)。指定すると、インターフェイスから送信されるすべてのトラフィックは、指定した VLAN ID で

タグ付けされます。着信トラフィックは、指定した VLAN ID でフィルタリングされ、インスタンスに渡される前にすべての VLAN タグが除去されます。

2. ステップ 1 で DNI をインスタンスに関連付けなかった場合は、[ダイレクトネットワークインターフェイスの更新](#) コマンドを実行して関連付けることができます。
3. DNI を作成し、EC2 互換インスタンスに関連付けた後、Amazon EC2 互換インスタンス内で 2 つの設定変更を行う必要があります。
 - 1 つ目の変更は、EC2 互換インスタンスに関連付けられた VNI 用のパケットが eth0 を介して送信されるように設定することです。
 - 2 つ目の変更は、ブート時に DHCP または静的 IP を使用するようにダイレクトネットワークインターフェイスを設定することです。

以下に、これらの設定変更を行う Amazon Linux 2 と CentOS Linux のシェルスクリプトの例を示します。

Amazon Linux 2

```
# Mac address of the direct network interface.
# You got this when you created the direct network interface.
DNI_MAC=[MAC ADDRESS FROM CREATED DNI]

# Configure routing so that packets meant for the VNI always are sent through
eth0.
PRIVATE_IP=$(curl -s http://169.254.169.254/latest/meta-data/local-ipv4)
PRIVATE_GATEWAY=$(ip route show to match 0/0 dev eth0 | awk '{print $3}')
ROUTE_TABLE=10001
echo "from $PRIVATE_IP table $ROUTE_TABLE" > /etc/sysconfig/network-scripts/
rule-eth0
echo "default via $PRIVATE_GATEWAY dev eth0 table $ROUTE_TABLE" > /etc/
sysconfig/network-scripts/route-eth0
echo "169.254.169.254 dev eth0" >> /etc/sysconfig/network-scripts/route-eth0

# Query the persistent DNI name, assigned by udev via ec2net helper.
# changable in /etc/udev/rules.d/70-persistent-net.rules
DNI=$(ip --oneline link | grep -i $DNI_MAC | awk -F ': ' '{ print $2 }')

# Configure DNI to use DHCP on boot.
cat << EOF > /etc/sysconfig/network-scripts/ifcfg-$DNI
DEVICE="$DNI"
NAME="$DNI"
```

```
HWADDR=$DNI_MAC
ONBOOT=yes
NOZEROCONF=yes
BOOTPROTO=dhcp
TYPE=Ethernet
MAINROUTETABLE=no
EOF

# Make all changes live.
systemctl restart network
```

CentOS Linux

```
# Mac address of the direct network interface. You got this when you created the
direct network interface.
DNI_MAC=[MAC ADDRESS FROM CREATED DNI]
# The name to use for the direct network interface. You can pick any name that
isn't already in use.
DNI=eth1

# Configure routing so that packets meant for the VNIC always are sent through
eth0
PRIVATE_IP=$(curl -s http://169.254.169.254/latest/meta-data/local-ipv4)
PRIVATE_GATEWAY=$(ip route show to match 0/0 dev eth0 | awk '{print $3}')
ROUTE_TABLE=10001
echo from $PRIVATE_IP table $ROUTE_TABLE > /etc/sysconfig/network-scripts/rule-
eth0
echo default via $PRIVATE_GATEWAY dev eth0 table $ROUTE_TABLE > /etc/sysconfig/
network-scripts/route-eth0

# Configure your direct network interface to use DHCP on boot.
cat << EOF > /etc/sysconfig/network-scripts/ifcfg-$DNI
DEVICE="$DNI"
NAME="$DNI"
HWADDR="$DNI_MAC"
ONBOOT=yes
NOZEROCONF=yes
BOOTPROTO=dhcp
TYPE=Ethernet
EOF

# Rename DNI device if needed.
```

```
CURRENT_DEVICE_NAME=$(LANG=C ip -o link | awk -F ':' -vIGNORECASE=1 '!/link\/  
ieee802\.\11/ && /'"$DNI_MAC"'/ { print $2 }')  
ip link set $CURRENT_DEVICE_NAME name $DNI  
  
# Make all changes live.  
systemctl restart network
```

SSH を使用して Snow Family デバイスのコンピューティングインスタンスに接続する

Secure Shell (SSH) を使用して Snow Family デバイスのコンピューティングインスタンスに接続するには、SSH キーを提供または作成するための以下のオプションがあります。

- デバイスを注文するジョブを作成するときに、Amazon マシンイメージ (AMI) の SSH キーを指定できます。詳細については、「[AMI を設定して SSH を使用し、デバイスで起動するコンピューティングインスタンスに接続する](#)」を参照してください。
- Snow Family デバイ스에 インポートする仮想マシンイメージを作成するときに、AMI の SSH キーを指定できます。詳細については、「[Snow Family デバイ스에 仮想マシンイメージをインポートする](#)」を参照してください。
- Snow Family デバイ스에서 キーペアを作成し、ローカルで生成されたパブリックキーを使用してインスタンスを起動することを選択できます。詳細については、「[Amazon EC2 ユーザーガイド](#)」の「[Amazon EC2 を使用してキーペアを作成する Amazon EC2](#)」を参照してください。

SSH 経由でインスタンスに接続するには

1. デバイスの電源が入っており、ネットワークに接続され、ロックが解除されていることを確認します。詳細については、「[ローカルネットワークへの接続](#)」を参照してください。
2. コンピューティングインスタンス用にネットワークが設定されていることを確認します。詳細については、「[コンピューティングインスタンスのネットワーク設定](#)」を参照してください。
3. メモを確認して、この特定のインスタンスに使用した PEM または PPK キーペアを見つけます。コンピュータ上の場所にこれらのファイルのコピーを作成します。PEM ファイルへのパスを書き留めます。
4. 次のコマンド例のように、SSH を使用してインスタンスに接続します。この IP アドレスは、「[コンピューティングインスタンスのネットワーク設定](#)」で設定した仮想ネットワークインターフェイス (VNIC) の IP アドレスです。


```
ssh -i path/to/PEM/key/file instance-user-name@192.0.2.0
```

詳細については、「Amazon EC2 [ユーザーガイド](#)」の「[SSH を使用した Linux インスタンスへの接続](#)」を参照してください。Amazon EC2

同じ Snowball Edge 上の EC2 互換コンピューティングインスタンスから S3 バケットにデータを転送する

同じ Snowball Edge 上のコンピューティングインスタンスと Amazon S3 バケット間で、データを転送することができます。これを行うには、サポートされている AWS CLI コマンドと適切なエンドポイントを使用します。例えば、sbe1.xlarge インスタンスのディレクトリから同じデバイスの Amazon S3 バケット、myBucket にデータを移動するとします。Snow Family デバイスエンドポイント <https://S3-object-API-endpoint:443> で Amazon S3 互換ストレージを使用していると仮定します。次の手順に従ってください。

Note

この手順は、「[AMI を設定して SSH を使用し、デバイスで起動するコンピューティングインスタンスに接続する](#)」の手順に従っている場合のみ使用できます。

同じ Snowball Edge 上のコンピューティングインスタンスとバケット間でデータを転送するには

1. SSH を使用してコンピューティングインスタンスに接続します。
2. をダウンロードしてインストールします AWS CLI。インスタンスに AWS CLIがない場合は、これをダウンロードしてインストールします。詳細については、「[AWS Command Line Interface のインストール](#)」を参照してください。
3. Snowball Edge AWS CLI の Amazon S3 エンドポイントで動作するようにコンピューティングインスタンスでを設定します。詳細については、「[ローカルの Amazon S3 認証情報の取得と使用](#)」を参照してください。
4. Snow Family デバイスのサポートされている Amazon S3 互換ストレージのコマンドを使用して、データを転送します。例:

```
aws s3 cp ~/june2018/results s3://myBucket/june2018/results --recursive --endpoint
https://S3-object-API-endpoint:443
```

コンピューティングインスタンスの Snowball Edge クライアントのコマンド

Snowball Edge クライアントは、ローカルサーバー上で実行できるスタンドアロンのターミナルアプリケーションです。このアプリケーションにより、Snowball Edge デバイスまたはデバイスのクラスターで、いくつかの管理タスクを実行することができます。Snowball Edge クライアントを使用したサービスの開始および停止方法など、このアプリケーションの使用法の詳細については、[Snowball Edge クライアントコマンドの使用](#) を参照してください。

以下で、コンピューティングインスタンスに固有の Snowball Edge コマンドに関する情報 (使用例など) を確認できます。

デバイスで使用できる Amazon EC2-compatible コマンドのリストについては AWS Snowball Edge、「」を参照してください [Snowball Edge でサポートされている Amazon EC2-compatible AWS CLI コマンド](#)。

Amazon EC2 互換インスタンスを自動起動するための起動設定の作成

ロック解除後に AWS Snowball Edge デバイスで Amazon EC2-compatible コンピューティングインスタンスを自動的に起動するには、起動設定を作成します。これを行うには、以下のよう
に、snowballEdge create-autostart-configuration コマンドを使用します。

使用方法

```
snowballEdge create-autostart-configuration --physical-connector-type [SFP_PLUS or RJ45
or QSFP] --ip-address-assignment [DHCP or STATIC] [--static-ip-address-configuration
IpAddress=[IP address],NetMask=[Netmask]] --launch-template-id [--launch-template-
version]
```

EC2 互換インスタンスを自動起動するための起動設定の更新

Snowball Edge の既存の起動設定を更新するには、snowballEdge update-autostart-configuration コマンドを使用します。その使用法は以下のとおりです。起動設定を有効または無効にするには、--enabled パラメータを指定します。

使用方法

```
snowballEdge update-autostart-configuration --autostart-configuration-arn [--physical-connector-type [SFP_PLUS or RJ45 or QSFP]] [--ip-address-assignment [DHCP or STATIC]] [--static-ip-address-configuration IpAddress=[IP address],NetMask=[Netmask]][--launch-template-id] [--launch-template-version] [--enabled]
```

EC2 互換インスタンスを自動起動するための起動設定の削除

使用しなくなった起動設定を削除するには、以下のように、`snowballEdge delete-autostart-configuration` コマンドを使用します。

使用方法

```
snowballEdge delete-autostart-configuration --autostart-configuration-arn
```

EC2 互換インスタンスを自動起動するための起動設定を一覧表示する

Snowball Edge 上で作成した起動設定を一覧表示するには、以下のように、`describe-autostart-configurations` コマンドを使用します。

使用方法

```
snowballEdge describe-autostart-configurations
```

仮想ネットワークインターフェイスの作成

Snowball Edge でコンピューティングインスタンスを実行したり、NFS インターフェイスを起動したりするには、まず仮想ネットワークインターフェイス (VNIC) を作成します。各 Snowball Edge には、デバイスの物理的ネットワークインターフェイスコントローラーである、3 つのネットワークインターフェイス (NIC) があります。デバイスの背面に、RJ45、SFP、および QSFP ポートがあります。

各 VNIC は物理的なものに基づいており、各 NIC に関連付けられている VNIC の数に制限はありません。仮想ネットワークインターフェイスを作成するには、`snowballEdge create-virtual-network-interface` コマンドを使用します。

Note

--static-ip-address-configuration パラメータは、--ip-address-assignment パラメータに STATIC オプションを使用している場合にのみ有効です。

使用

このコマンドは、Snowball Edge クライアントの設定ありか、設定なしの、2つの方法で使用できます。以下の使用例は、Snowball Edge クライアントの設定ありの方法を示しています。

```
snowballEdge create-virtual-network-interface --ip-address-assignment [DHCP or STATIC]
--physical-network-interface-id [physical network interface id] --static-ip-address-
configuration IpAddress=[IP address],NetMask=[Netmask]
```

以下の使用例は、Snowball Edge クライアントの設定なしの方法を示しています。

```
snowballEdge create-virtual-network-interface --endpoint https://[ip address]
--manifest-file /path/to/manifest --unlock-code [unlock code] --ip-address-
assignment [DHCP or STATIC] --physical-network-interface-id [physical network interface
id] --static-ip-address-configuration IpAddress=[IP address],NetMask=[Netmask]
```

Example 例: VNIC の作成 (DHCP を使用)

```
snowballEdge create-virtual-network-interface --ip-address-assignment dhcp --physical-
network-interface-id s.ni-8EXAMPLEaEXAMPLEd
{
  "VirtualNetworkInterface" : {
    "VirtualNetworkInterfaceArn" : "arn:aws:snowball-device:::interface/
s.ni-8EXAMPLE8EXAMPLEf",
    "PhysicalNetworkInterfaceId" : "s.ni-8EXAMPLEaEXAMPLEd",
    "IpAddressAssignment" : "DHCP",
    "IpAddress" : "192.0.2.0",
    "Netmask" : "255.255.255.0",
    "DefaultGateway" : "192.0.2.1",
    "MacAddress" : "EX:AM:PL:E1:23:45",
    "MtuSize" : "1500"
  }
}
```

仮想ネットワークインターフェイスの記述

デバイスで以前に作成した VNIC を定義するには、`snowballEdge describe-virtual-network-interfaces` コマンドを使用します。その使用方法は以下のとおりです。

使用

このコマンドは、Snowball Edge クライアントの設定ありか、設定なしの、2 つの方法で使用できます。以下の使用例は、Snowball Edge クライアントの設定ありの方法を示しています。

```
snowballEdge describe-virtual-network-interfaces
```

以下の使用例は、Snowball Edge クライアントの設定なしの方法を示しています。

```
snowballEdge describe-virtual-network-interfaces --endpoint https://[ip address] --manifest-file /path/to/manifest --unlock-code [unlock code]
```

Example 例: VNIC の定義

```
snowballEdge describe-virtual-network-interfaces
[
  {
    "VirtualNetworkInterfaceArn" : "arn:aws:snowball-device:::interface/
s.ni-8EXAMPLE8EXAMPLE8",
    "PhysicalNetworkInterfaceId" : "s.ni-8EXAMPLEaEXAMPLEd",
    "IpAddressAssignment" : "DHCP",
    "IpAddress" : "192.0.2.0",
    "Netmask" : "255.255.255.0",
    "DefaultGateway" : "192.0.2.1",
    "MacAddress" : "EX:AM:PL:E1:23:45",
    "MtuSize" : "1500"
  },{
    "VirtualNetworkInterfaceArn" : "arn:aws:snowball-device:::interface/
s.ni-1EXAMPLE1EXAMPLE1",
    "PhysicalNetworkInterfaceId" : "s.ni-8EXAMPLEaEXAMPLEd",
    "IpAddressAssignment" : "DHCP",
    "IpAddress" : "192.0.2.2",
    "Netmask" : "255.255.255.0",
    "DefaultGateway" : "192.0.2.1",
    "MacAddress" : "12:34:5E:XA:MP:LE",
    "MtuSize" : "1500"
  }
]
```

]

仮想ネットワークインターフェイスの更新

仮想ネットワークインターフェイス (VNIC) の作成後、`snowballEdge update-virtual-network-interface` コマンドを使用して、その設定を更新できます。特定の VNIC の Amazon リソースネーム (ARN) を指定してから、更新する要素に対してのみ値を指定します。

使用方法

このコマンドは、Snowball Edge クライアントの設定ありか、設定なしの、2つの方法で使用できます。以下の使用例は、Snowball Edge クライアントの設定ありの方法を示しています。

```
snowballEdge update-virtual-network-interface --virtual-network-interface-arn [virtual-network-interface-arn] --ip-address-assignment [DHCP or STATIC] --physical-network-interface-id [physical network interface id] --static-ip-address-configuration IpAddress=[IP address],NetMask=[Netmask]
```

以下の使用例は、Snowball Edge クライアントの設定なしの方法を示しています。

```
snowballEdge update-virtual-network-interface --endpoint https://[ip address] --manifest-file /path/to/manifest --unlock-code [unlock code] --virtual-network-interface-arn [virtual network-interface-arn] --ip-address-assignment [DHCP or STATIC] --physical-network-interface-id [physical network interface id] --static-ip-address-configuration IpAddress=[IP address],NetMask=[Netmask]
```

Example 例: VNIC の更新 (DHCP を使用)

```
snowballEdge update-virtual-network-interface --virtual-network-interface-arn arn:aws:snowball-device:::interface/s.ni-8EXAMPLEbEXAMPLEd --ip-address-assignment dhcp
```

仮想ネットワークインターフェイスの削除

仮想ネットワークインターフェイスを削除するには、`snowballEdge delete-virtual-network-interface` コマンドを使用できます。

使用方法

このコマンドは、Snowball Edge クライアントの設定ありか、設定なしの、2つの方法で使用できます。以下の使用例は、Snowball Edge クライアントの設定ありの方法を示しています。

```
snowballEdge delete-virtual-network-interface --virtual-network-interface-arn [virtual network-interface-arn]
```

以下の使用例は、Snowball Edge クライアントの設定なしの方法を示しています。

```
snowballEdge delete-virtual-network-interface --endpoint https://[ip address] --manifest-file /path/to/manifest --unlock-code [unlock code] --virtual-network-interface-arn [virtual network-interface-arn]
```

Example 例: VNIC の削除

```
snowballEdge delete-virtual-network-interface --virtual-network-interface-arn arn:aws:snowball-device:::interface/s.ni-8EXAMPLEbEXAMPLEd
```

Amazon EC2 互換エンドポイントの使用

以下は、Amazon EC2 互換エンドポイントの概要です。このエンドポイントでは、Amazon EC2 互換 API オペレーションを使用して、Amazon マシンイメージ (AMI) とコンピューティングインスタンスをプログラムで管理できます。

Amazon EC2-compatible エンドポイントを AWS CLI エンドポイントとして指定する

を使用して AWS Snowball Edge デバイスにコマンド AWS CLI を発行する場合、エンドポイントが Amazon EC2-compatible エンドポイントであることを指定できます。次に示すように、HTTPS エンドポイント、またはセキュリティで保護されていない HTTP エンドポイントの使用を選択できます。

セキュリティで保護された HTTPS エンドポイント

```
aws ec2 describe-instances --endpoint https://192.0.2.0:8243 --ca-bundle path/to/certificate
```

セキュリティで保護されていない HTTP エンドポイント

```
aws ec2 describe-instances --endpoint http://192.0.2.0:8008
```

8243 の HTTPS エンドポイントを使用すると、転送中のデータは暗号化されます。この暗号化は、ロックの解除時に Snowball Edge によって生成される証明書を使用して行われます。証明書を取得

したら、ローカルの `ca-bundle.pem` ファイルに保存できます。次に、以下に示すように AWS CLI プロファイルを設定し、証明書へのパスを含めます。

証明書を Amazon EC2 互換エンドポイントに関連付けるには

1. Snowball Edge を電源とネットワークに接続し、電源を入れます。
2. デバイスのロック解除が終了したら、ローカルネットワークでの IP アドレスを書き留めておきます。
3. ネットワーク上のターミナルから Snowball Edge に ping を送信できることを確認します。
4. ターミナルで `snowballEdge get-certificate` コマンドを実行します。このコマンドの詳細については、「[公開鍵証明書の管理](#)」を参照してください。
5. `snowballEdge get-certificate` コマンドの出力をファイルに保存します (例: `ca-bundle.pem`)。
6. ターミナルから次のコマンドを実行します。

```
aws configure set profile.snowballEdge.ca_bundle /path/to/ca-bundle.pem
```

この手順を完了したら、これらのローカル認証情報、証明書、および指定されたエンドポイントを使用して CLI コマンドを実行できます。

Snowball Edge でサポートされている Amazon EC2-compatible AWS CLI コマンド

Snow Family デバイス上のコンピューティングインスタンスは、Amazon EC2 互換エンドポイントを使用して管理できます。このタイプのエンドポイントは、AWS SDKs の多くの Amazon EC2 CLI コマンドとアクションをサポートします。AWS リージョン AWS CLI の呼び出し先を指定するなど AWS CLI、のインストールと設定については、[AWS Command Line Interface 「ユーザーガイド」](#) を参照してください。

Snowball Edge でサポートされる Amazon EC2 互換の AWS CLI コマンドのリスト

以下は、Snowball Edge デバイスでサポートされている Amazon EC2 の AWS CLI コマンドとオプションのサブセットの説明です。コマンドまたはオプションが以下に示されていない場合、そのコマンドまたはオプションはサポートされていません。サポートされていない一部のオプションを、コマンドとともに宣言できます。ただし、それらのオプションは無視されます。

- [associate-address](#) – デバイスの 3 つの物理ネットワークインターフェイスのいずれかで使用するために、仮想 IP アドレスをインスタンスに関連付けます。

- `--instance-id` – 単一の sbe インスタンスの ID。
- `--public-ip` – インスタンスへのアクセスに使用する仮想 IP アドレス。
- [attach-volume](#) – デバイ스에서停止したか実行中のインスタンスに Amazon EBS ボリュームをアタッチし、指定したデバイス名でインスタンスに公開します。
 - `--device value` – デバイス名。
 - `--instance-id` – ターゲットの Amazon EC2 互換インスタンスの ID。
 - `-volume-id value` – EBS ボリュームの ID。
- [authorize-security-group-egress](#) – Snowball Edge デバイスで使用する 1 つ以上の出カールールをセキュリティグループに追加します。具体的には、このアクションでは、1 つ以上の IPv4 CIDR アドレス範囲にインスタンスがトラフィックを送信できます。詳細については、「[Snowball Edge デバイスのセキュリティグループ](#)」を参照してください。
 - `--group-id value` – セキュリティグループの ID
 - `[--ip-permissions value]` – IP アクセス許可の 1 つ以上のセット。
- [authorize-security-group-ingress](#) – セキュリティグループに 1 つ以上の進入ルールを追加します。authorize-security-group-ingress を呼び出すときは、group-name または group-id のどちらかの値を指定する必要があります。
 - `[--group-name value]` – セキュリティグループ名。
 - `[--group-id value]` – セキュリティグループの ID
 - `[--ip-permissions value]` – IP アクセス許可の 1 つ以上のセット。
 - `[--protocol value]` IP プロトコル。指定できる値は tcp、udp、および icmp です。「すべてのプロトコル」が指定されている場合を除き、`--port` 引数は必須です (-1)。
 - `[--port value]` – TCP または UDP 用に許可するポートの範囲。この値は、1 つの整数または範囲 (最小/最大) です。

ICMP の場合、1 つの整数または範囲 (type-code) です。この場合、type は ICMP タイプ番号を表し、code は ICMP コード番号を表します。-1 の値は、すべての ICMP タイプのすべての ICMP コードを示します。type だけの -1 の値は、特定の ICMP タイプのすべての ICMP コードを示します。
 - `[--cidr value]` – CIDR IP の範囲。
- [create-launch-template](#) – 起動テンプレートを作成します。起動テンプレートには、インスタンスを起動するためのパラメータが含まれています。RunInstances を使用してインスタンスを起動するときに、リクエストで起動パラメータを指定する代わりに起動テンプレートを指定できます。
デバイスごとに最大 100 個のテンプレートを作成できます。

- `--launch-template-name string` – 起動テンプレートの名前。
- `--launch-template-data structure` – 起動テンプレートの情報。以下の属性がサポートされています。
 - `ImageId`
 - `InstanceType`
 - `SecurityGroupIds`
 - `TagSpecifications`
 - `UserData`

JSON 構文:

```
{
  "ImageId":"string",
  "InstanceType":"sbe-c.large",
  "SecurityGroupIds":["string", ...],
  "TagSpecifications":[{"ResourceType":"instance","Tags":
    [{"Key":"Name","Value":"Test"},
     {"Key":"Stack","Value":"Gamma"}]},
  "UserData":"this is my user data"
}
```

- `[-version-description string]` — 起動テンプレートの最初のバージョンの説明。
- `--endpoint snowballEndpoint` – Amazon EC2 互換 API オペレーションを使用して、コンピューティングインスタンスをプログラムで管理できるようにする値。詳細については、[「Amazon EC2-compatible エンドポイントを AWS CLI エンドポイントとして指定する」](#)を参照してください。
- [create-launch-template-version](#) – 起動テンプレートの新しいバージョンを作成します。新しいバージョンの基になる起動テンプレートの既存のバージョンを指定できます。起動テンプレートのバージョンには、作成された順序で番号が付けられます。起動テンプレートのバージョン番号を指定、変更、または置き換えることはできません。起動テンプレートごとに最大 100 個のバージョンを作成できます。

リクエストで起動テンプレート ID または起動テンプレート名を指定します。

- `--launch-template-id string` – 起動テンプレートの ID。
- `--launch-template-name string` – 起動テンプレートの名前。

- `--launch-template-data structure` – 起動テンプレートの情報。以下の属性がサポートされています。
 - ImageId
 - InstanceType
 - SecurityGroupIds
 - TagSpecifications
 - UserData

JSON 構文:

```
{
  "ImageId":"string",
  "InstanceType":"sbe-c.large",
  "SecurityGroupIds":["string", ...],
  "TagSpecifications":[{"ResourceType":"instance","Tags":
[{"Key":"Name","Value":"Test"},
 {"Key":"Stack","Value":"Gamma"}]}],
  "UserData":"this is my user data"
}
```

- `[-source-version string]` — 新しいバージョンの基となる起動テンプレートのバージョン番号。新しいバージョンは、`launch-template-data` で指定したパラメータを除いて、基になるバージョンと同じ起動パラメータを継承します。
- `[-version-description string]` — 起動テンプレートの最初のバージョンの説明。
- `--endpoint snowballEndpoint` – Amazon EC2 互換 API オペレーションを使用して、コンピューティングインスタンスをプログラムで管理できるようにする値。詳細については、「[Amazon EC2-compatible エンドポイントを AWS CLI エンドポイントとして指定する](#)」を参照してください。
- [create-tags](#) — 指定したリソースの 1 つ以上のタグを追加または上書きします。各リソースには、最大 50 個のタグを設定できます。各タグはキーとオプションの値で構成されます。タグキーは、リソースごとに一意にする必要があります。以下のリソースがサポートされています。
 - AMI
 - インスタンス
 - 起動テンプレート
 - セキュリティグループ

- [create-security-group](#) – Snowball Edge にセキュリティグループを作成します。最大で 50 個のセキュリティグループを作成できます。セキュリティグループを作成するときに、任意のフレンドリ名を指定します。
 - `--group-name value` — セキュリティグループ名。
 - `--description value` — セキュリティグループの説明。これは、情報提供のみを目的としています。この値の長さは最大 255 文字とすることができます。
- [create-volume](#) — デバイスのインスタンスにアタッチできる Amazon EBS ボリュームを作成します。
 - `[--size value]` – のボリュームのサイズ GiBs。1 GiB から 1 TB (1000) までです GiBs。
 - `[--snapshot-id value]` — ボリュームの作成元となるスナップショット。
 - `[--volume-type value]` — ボリュームのタイプ。値が指定されていない場合、デフォルト値は `sbg1` です。以下に示しているのは、可能な値です。
 - マグネティックボリュームの `sbg1`
 - SSD ボリュームの `sbp1`
 - `[--tag-specification value]` — 作成中にボリュームに適用するタグのリスト。
- [delete-launch-template](#) – 起動テンプレートを削除します。起動テンプレートを削除すると、すべてのバージョンが削除されます。

リクエストで起動テンプレート ID または起動テンプレート名を指定します。

- `--launch-template-id string` – 起動テンプレートの ID。
- `--launch-template-name string` – 起動テンプレートの名前。
- `--endpoint snowballEndpoint` – Amazon EC2 互換 API オペレーションを使用して、コンピューティングインスタンスをプログラムで管理できるようにする値。詳細については、「[Amazon EC2-compatible エンドポイントを AWS CLI エンドポイントとして指定する](#)」を参照してください。
- [delete-launch-template-version](#) – 起動テンプレートの 1 つ以上のバージョンを削除します。起動テンプレートのデフォルトバージョンは削除できません。まずデフォルトとして別のバージョンを割り当てる必要があります。デフォルトバージョンが起動テンプレートの唯一のバージョンである場合は、`delete-launch-template` コマンドを使用して起動テンプレート全体を削除します。

リクエストで起動テンプレート ID または起動テンプレート名を指定します。

- `--launch-template-id string` – 起動テンプレートの ID。
- `--launch-template-name string` – 起動テンプレートの名前。

- `--versions (list) "string" "string"` — 削除する 1 つ以上の起動テンプレートバージョンのバージョン番号。
- `--endpoint snowballEndpoint` – Amazon EC2 互換 API オペレーションを使用して、コンピューティングインスタンスをプログラムで管理できるようにする値。詳細については、「[Amazon EC2-compatible エンドポイントを AWS CLI エンドポイントとして指定する](#)」を参照してください。
- [delete-security-group](#) – セキュリティグループを削除します。

インスタンスに関連付けられている、または他のセキュリティグループによって参照されているセキュリティグループを削除しようとする、DependencyViolation でそのオペレーションは失敗します。

- `--group-name value` — セキュリティグループ名。
- `--description value` — セキュリティグループの説明。これは、情報提供のみを目的としています。この値の長さは最大 255 文字とすることができます。
- [delete-tags](#) — 指定したリソース (AMI、コンピューティングインスタンス、起動テンプレート、セキュリティグループ) から、指定したタグのセットを削除します。
- [delete-volume](#) — 指定した Amazon EBS ボリュームを削除します。ボリュームは available 状態である必要があります (インスタンスにアタッチされていません)。
 - `--volume-id value` — ボリュームの ID。
- [describe-addresses](#) — デバイスで同じ数の sbe インスタンスに関連付けられている 1 つ以上の仮想 IP アドレスを定義します。
 - `--public-ips` – インスタンスに関連付けられている 1 つ以上の仮想 IP アドレス。
- [describe-images](#) — 使用可能な 1 つ以上のイメージ (AMI) について説明します。使用可能なイメージは、ジョブの作成時に Snowball Edge デバイスに追加されます。
 - `--image-id` — AMI の Snowball AMI ID。
- [describe-instance-attribute](#) – 指定されたインスタンスの指定された属性を記述します。一度に 1 つの属性のみ指定できます。以下の属性がサポートされています。
 - `instanceInitiatedShutdownBehavior`
 - `instanceType`
 - `userData`
- [describe-instances](#) – 1 つ以上のインスタンスを記述します。レスポンスは、インスタンスに割り当てられているすべてのセキュリティグループを返します。
 - `--instance-ids` – デバイスで停止した 1 つ以上の sbe インスタンスの ID。

- `--page-size` — 呼び出しで取得する各ページのサイズ。この値は、コマンドの出力で返された項目数には影響を与えません。ページサイズを小さく設定すると、デバイスへの呼び出しが多くなり、各呼び出しで取得する項目が少なくなります。こうすることで、呼び出しのタイムアウトを防ぐことができます。
- `--max-items` — コマンドの出力で返される項目の総数。使用可能な項目の総数が指定された値を上回る場合、コマンドの出力で `NextToken` が提供されます。ページ分割を再開するには、後続コマンドの `starting-token` 引数で `NextToken` 値を指定します。
- `--starting-token` — ページ割りを始める場所を指定するトークン。このトークンは、以前に切り詰められたレスポンスからの `NextToken` 値です。
- [describe-instance-status](#) – 指定されたインスタンスまたはすべてのインスタンスのステータスを記述します。デフォルトでは、すべてのインスタンスのステータスを返すよう特に指定しない限り、実行中のインスタンスのみが説明されます。インスタンスステータスには、以下の要素が含まれます。
 - ステータスチェック – Snow デバイスは、実行中の Amazon EC2 互換インスタンスのステータスチェックを実行して、ハードウェアおよびソフトウェアの問題を特定します。
 - インスタンスの状態 – インスタンスを起動した瞬間から終了時までインスタンスを管理できません。

このコマンドでは、次のフィルタがサポートされます。

- `[--filters]` (リスト)

フィルター。

- `instance-state-code` – 16 ビットの符号なし整数としてのインスタンスの状態のコード。上位バイトは内部サービスレポート目的で使用されるため、無視します。下位バイトは表示されている状態に基づいて設定されます。有効な値は 0 (保留中)、16 (実行中)、32 (シャットダウン中)、48 (終了)、64 (停止中)、80 (停止) です。
- `instance-state-name` – インスタンスの状態 (`pending` | `running` | `shutting-down` | `terminated` | `stopping` | `stopped`)。
- `instance-status.reachability` – 名前が `reachability` (`passed` | `failed` | `initializing` | `insufficient-data`) のインスタンスステータスに基づいてフィルタリングします。
- `instance-status.status` – インスタンスのステータス (`ok` | `impaired` | `initializing` | `insufficient-data` | `not-applicable`)。
- `system-status.reachability` – 名前が `reachability` (`passed` | `failed` | `initializing` | `insufficient-data`) のシステムステータスに基づいてフィルタリングします。

- `system-status.status` – インスタンスのシステムステータス (`ok` | `impaired` | `initializing` | `insufficient-data` | `not-applicable`)。
- JSON 構文:

```
[
  {
    "Name": "string",
    "Values": ["string", ...]
  }
  ...
]
```

- `[--instance-ids]` (リスト)

インスタンス ID。

デフォルト: すべてのインスタンスについて説明します。

- `[--dry-run|--no-dry-run]` (ブーリアン)

実際にリクエストを行わずに、ユーザーがアクションに必要なアクセス許可を持っているかどうかをチェックし、エラーレスポンスを返します。必要なアクセス許可がある場合、エラーレスポンスは `DryRunOperation` です。

それ以外の場合は、`UnauthorizedOperation` です。

- `[--include-all-instances | --no-include-all-instances]` (ブーリアン)

`true` の場合、すべてのインスタンスのヘルスステータスが含まれます。`false` の場合、実行中のインスタンスのみのヘルスステータスが含まれます。

デフォルト: `false`

- `[--page-size]` (整数) – 呼び出しで取得する各ページのサイズ。この値は、コマンドの出力で返された項目数には影響を与えません。ページサイズを小さく設定すると、デバイスへの呼び出しが多くなり、各呼び出しで取得する項目が少なくなります。こうすることで、呼び出しのタイムアウトを防ぐことができます。
- `[--max-items]` (整数) – コマンドの出力で返される項目の総数。使用可能な項目の総数が指定された値を上回る場合、コマンドの出力で `NextToken` が提供されます。ページ分割を再開するには、後続コマンドの `starting-token` 引数で `NextToken` 値を指定します。

- `[--starting-token]` (文字列) – ページ分割を始める場所を指定するトークン。このトークンは、以前に切り詰められたレスポンスからの `NextToken` 値です。
- [describe-launch-templates](#) – 1 つ以上の起動テンプレートについて説明します。 `describe-launch-templates` コマンドはページ分割されたオペレーションです。複数の呼び出しを行って、結果のデータセット全体を取得できます。

リクエストで起動テンプレート ID または起動テンプレート名を指定します。

- `--launch-template-ids` (リスト) "string" "string" – 起動テンプレートの IDs のリスト。
- `--launch-template-names` (リスト) "string" "string" – 起動テンプレートの名前のリスト。
- `--page-size` – 呼び出しで取得する各ページのサイズ。この値は、コマンドの出力で返された項目数には影響を与えません。ページサイズを小さく設定すると、デバイスへの呼び出しが多くなり、各呼び出しで取得する項目が少なくなります。こうすることで、呼び出しのタイムアウトを防ぐことができます。
- `--max-items` – コマンドの出力で返される項目の総数。使用可能な項目の総数が指定された値を上回る場合、コマンドの出力で `NextToken` が提供されます。ページ分割を再開するには、後続コマンドの `starting-token` 引数で `NextToken` 値を指定します。
- `--starting-token` – ページ割り始める場所を指定するトークン。このトークンは、以前に切り詰められたレスポンスからの `NextToken` 値です。
- `--endpoint snowballEndpoint` – Amazon EC2 互換 API オペレーションを使用して、コンピューティングインスタンスをプログラムで管理できるようにする値。詳細については、「[Amazon EC2-compatible エンドポイントを AWS CLI エンドポイントとして指定する](#)」を参照してください。
- [describe-launch-template-versions](#) – 指定された起動テンプレートの 1 つ以上のバージョンを記述します。すべてのバージョン、個々のバージョン、またはバージョンの範囲を定義できます。 `describe-launch-template-versions` コマンドはページ分割されたオペレーションです。複数の呼び出しを行って、結果のデータセット全体を取得できます。

リクエストで起動テンプレート ID または起動テンプレート名を指定します。

- `--launch-template-id string` – 起動テンプレートの ID。
- `--launch-template-name string` – 起動テンプレートの名前。
- `[--versions (list) "string" "string"]` – 削除する 1 つ以上の起動テンプレートバージョンのバージョン番号。
- `[--min-version string]` – 起動テンプレートバージョンを説明する最小のバージョン番号。

- `[--max-version string]` — 起動テンプレートバージョンを説明する最大のバージョン番号。
- `--page-size` — 呼び出しで取得する各ページのサイズ。この値は、コマンドの出力で返された項目数には影響を与えません。ページサイズを小さく設定すると、デバイスへの呼び出しが多くなり、各呼び出しで取得する項目が少なくなります。こうすることで、呼び出しのタイムアウトを防ぐことができます。
- `--max-items` — コマンドの出力で返される項目の総数。使用可能な項目の総数が指定された値を上回る場合、コマンドの出力で NextToken が提供されます。ページ分割を再開するには、後続コマンドの `starting-token` 引数で NextToken 値を指定します。
- `--starting-token` — ページ割りを始める場所を指定するトークン。このトークンは、以前に切り詰められたレスポンスからの NextToken 値です。
- `--endpoint snowballEndpoint` – Amazon EC2 互換 API オペレーションを使用して、コンピューティングインスタンスをプログラムで管理できるようにする値。詳細については、「[Amazon EC2-compatible エンドポイントを AWS CLI エンドポイントとして指定する](#)」を参照してください。
- [describe-security-groups](#) – 1 つ以上のセキュリティグループについて説明します。

`describe-security-groups` コマンドはページ分割されたオペレーションです。複数の API コールを発行して、結果のデータセット全体を取得できます。

- `[--group-name value]` — セキュリティグループ名。
- `[--group-id value]` — セキュリティグループの ID。
- `[--page-size value]` – AWS サービスコールで取得する各ページのサイズ。このサイズは、コマンドの出力で返された項目数には影響を与えません。ページサイズを小さく設定すると、AWS サービスへの呼び出しが多くなり、各呼び出しで取得する項目が少なくなります。このアプローチは、AWS サービスコールがタイムアウトするのを防ぐのに役立ちます。使用例については、AWS Command Line Interface ユーザーガイドの[ページ割り](#)を参照してください。
- `[--max-items value]` — コマンドの出力で返される項目の総数。使用可能な項目の総数が指定された値を上回る場合、コマンドの出力で NextToken が提供されます。ページ分割を再開するには、後続コマンドの `starting-token` 引数で NextToken 値を指定します。AWS CLI の範囲外で NextToken レスポンス要素を直接使用しないでください。使用例については、AWS Command Line Interface ユーザーガイドの[ページ割り](#)を参照してください。
- `[--starting-token value]` — ページ割りを始める場所を指定するトークン。このトークンは、以前に切り詰められたレスポンスからの NextToken 値です。使用例については、AWS Command Line Interface ユーザーガイドの[ページ割り](#)を参照してください。

- [describe-tags](#) — 指定したリソース (image、instance、またはセキュリティグループ) の1 つ以上のタグを説明します。このコマンドでは、次のフィルタがサポートされています。
 - launch-template
 - resource-id
 - resource-type – image または instance
 - キー
 - value
- [describe-volumes](#) — 指定した Amazon EBS ボリュームについて説明します。
 - [--max-items value] — コマンドの出力で返される項目の総数。使用可能な項目の総数が指定された値を上回る場合、コマンドの出力で NextToken が提供されます。ページ分割を再開するには、後続コマンドの starting-token 引数で NextToken 値を指定します。
 - [--starting-token value] — ページ割りを始める場所を指定するトークン。このトークンは、以前に切り詰められたレスポンスからの NextToken 値です。
 - [--volume-ids value] — 1 つ以上のボリュームの ID。
- [detach-volume](#) — 停止したか、実行中のインスタンスから Amazon EBS ボリュームをデタッチします。
 - [--device value] — デバイス名。
 - [--instance-id] — ターゲットの Amazon EC2 インスタンスの ID。
 - --volume-id value — ボリュームの ID。
- [disassociate-address](#) – 仮想 IP アドレスとインスタンスの関連付けを解除します。
 - --public-ip — インスタンスからの関連付けを解除する仮想 IP アドレス。
- [get-launch-template-data](#) – 指定されたインスタンスの設定データを取得します。このデータを使用して起動テンプレートを作成できます。
 - --instance-id – 単一の sbe インスタンスの ID。
 - --endpoint snowballEndpoint – Amazon EC2 互換 API オペレーションを使用して、コンピューティングインスタンスをプログラムで管理できるようにする値。詳細については、「[Amazon EC2-compatibleエンドポイントを AWS CLI エンドポイントとして指定する](#)」を参照してください。
- [modify-launch-template](#) – 起動テンプレートを変更します。起動テンプレートのどのバージョンをデフォルトバージョンとして設定するかを指定できます。起動テンプレートのバージョンを指定せずにインスタンスを起動すると、起動テンプレートのデフォルトバージョンが適用されます。

リクエストで起動テンプレート ID または起動テンプレート名を指定します。

- `--launch-template-id string` – 起動テンプレートの ID。
- `--launch-template-name string` – 起動テンプレートの名前。
- `--default-version string` – デフォルトバージョンとして設定された起動テンプレートのバージョン番号。
- `--endpoint snowballEndpoint` – Amazon EC2 互換 API オペレーションを使用して、コンピューティングインスタンスをプログラムで管理できるようにする値。詳細については、「[Amazon EC2-compatible エンドポイントを AWS CLI エンドポイントとして指定する](#)」を参照してください。
- [modify-instance-attribute](#) – 指定されたインスタンスの属性を変更します。以下の属性がサポートされています。
 - `instanceInitiatedShutdownBehavior`
 - `userData`
- [revoke-security-group-egress](#) – セキュリティグループから 1 つ以上の出カールールを削除します。
 - `[--group-id value]` – セキュリティグループの ID
 - `[--ip-permissions value]` – IP アクセス許可の 1 つ以上のセット。
- [revoke-security-group-ingress](#) – セキュリティグループへの 1 つ以上の進入ルールを取り消します。 `revoke-security-group-ingress` を呼び出すときは、 `group-name` または `group-id` のどちらかの値を指定する必要があります。
 - `[--group-name value]` – セキュリティグループ名。
 - `[--group-id value]` – セキュリティグループの ID。
 - `[--ip-permissions value]` – IP アクセス許可の 1 つ以上のセット。
 - `[--protocol value]` IP プロトコル。指定できる値は `tcp`、`udp`、および `icmp` です。「すべてのプロトコル」が指定されている場合を除き、 `--port` 引数は必須です (-1)。
 - `[--port value]` – TCP または UDP 用に許可するポートの範囲。単一の整数または範囲 (最小/最大)。

ICMP の場合、1 つの整数または範囲 (type-code) です。この場合、 `type` は ICMP タイプ番号を表し、 `code` は ICMP コード番号を表します。 -1 の値は、すべての ICMP タイプのすべての ICMP コードを示します。 `type` だけの -1 の値は、特定の ICMP タイプのすべての ICMP コードを示します。
- `[--cidr value]` – CIDR IP の範囲。
- [run-instances](#) – AMI 用の Snowball AMI ID を使用して、複数のコンピューティングインスタンスを起動します。

Note

インスタンスのサイズとタイプに応じて、Snowball Edge 上でコンピューティングインスタンスを起動する際に、最大で 1 時間半かかることがあります。

- `--block-device-mappings (list)` – ブロックデバイスマッピングエントリ。パラメータ `DeleteOnTermination`、`VolumeSize`、および `VolumeType` がサポートされています。ブートボリュームのタイプは `sbg1` である必要があります。

このコマンドの JSON 構文は次のとおりです。

```
{
  "DeviceName": "/dev/sdh",
  "Ebs":
  {
    "DeleteOnTermination": true|false,
    "VolumeSize": 100,
    "VolumeType": "sbp1"|"sbg1"
  }
}
```

- `--count` — 起動するインスタンスの数。単一の数字が指定されている場合は、起動する最小値と見なされます (デフォルトは 1)。`min:max` の形式で範囲が指定されている場合、最初の数字は起動するインスタンスの最小数と解釈され、2 番目は起動するインスタンスの最大数として解釈されます。
- `--image-id` — `describe-images` を呼び出して取得できる、AMI の Snowball AMI ID。インスタンスを起動するには、AMI が必要です。
- `-- InstanceInitiatedShutdownBehavior` – デフォルトでは、インスタンスからシャットダウンを開始すると (シャットダウンや電源オフなどのコマンドを使用)、インスタンスは停止します。この動作を変更して、インスタンスの停止ではなく終了させることができます。パラメータ `stop` および `terminate` がサポートされています。デフォルトは `stop` です。詳細については、Amazon EC2 Linux インスタンス用ユーザーガイドの [インスタンスによって開始されるシャットダウン動作の変更](#) を参照してください。
- `--instance-type` – `sbe` インスタンスタイプ。
- `--launch-template structure` — インスタンスの起動に使用する起動テンプレート。`run-instances` コマンドで指定したパラメータはすべて、起動テンプレート内の同じパラメータを

上書きします。起動テンプレートの名前または ID のいずれかを指定できますが、両方を指定することはできません。

```
{
  "LaunchTemplateId": "string",
  "LaunchTemplateName": "string",
  "Version": "string"
}
```

- -- security-group-ids 1 つ以上のセキュリティグループ ID。を使用してセキュリティグループを作成できます [CreateSecurityGroup](#)。値を指定しない場合、デフォルトのセキュリティグループの ID が作成したインスタンスに割り当てられます。
- --tag-specifications — 起動中にリソースに適用されるタグ。起動時にのみインスタンスにタグ付けできます。指定されたタグは、起動時に作成されたすべてのインスタンスに適用されます。作成後のリソースにタグを付けるには、create-tags を使用します。
- --user-data — インスタンスで使用可能なユーザーデータ。を使用している場合は AWS CLI、base64 エンコードが実行され、ファイルからテキストをロードできます。それ以外の場合は、base64 でエンコードされたテキストを指定する必要があります。
- --key-name (文字列) — キーペア名。キーペアは、CreateKeyPair または ImportKeyPair を使用して作成できます。。

Warning

キーペアを指定しない場合、ユーザーが別の方法でログインすることを許可する設定の AMI を選択した場合を除き、インスタンスに接続することはできません。

- [start-instances](#) — 以前に停止した sbe インスタンスを開始します。インスタンスにアタッチされたすべてのリソースは、開始や停止の間は保持されますが、インスタンスを終了させると消去されます。
- --instance-ids – デバイスで停止した 1 つ以上の sbe インスタンスの ID。
- [stop-instances](#) — 実行中の sbe インスタンスを停止します。インスタンスにアタッチされたすべてのリソースは、開始や停止の間は保持されますが、インスタンスを終了させると消去されます。
- --instance-ids – デバイスで停止する 1 つ以上の sbe インスタンスの ID。
- [terminate-instances](#) — 1 つ以上のインスタンスをシャットダウンします。このオペレーションはべき等です。インスタンスを複数回終了させると、各コールは成功します。インスタンスにアタッチされたすべてのリソースは、開始や停止の間は保持されますが、インスタンスを終了させるとデータは消去されます。

Note

デフォルトでは、`shutdown` や `poweroff` などのコマンドを使用してインスタンスからシャットダウンを開始すると、インスタンスは停止します。ただし、`InstanceInitiatedShutdownBehavior` 属性を使用してこの動作を変更し、これらのコマンドでインスタンスを終了させることができます。詳細については、Amazon EC2 Linux インスタンス用ユーザーガイドの [インスタンスによって開始されるシャットダウン動作の変更](#) を参照してください。

- `--instance-ids` — デバイスで終了する 1 つ以上の sbe インスタンスの ID。これらのインスタンスに保存されたすべての関連するデータは失われます。
- [create-key-pair](#) — 指定された名前の 2048 ビット RSA キーペアを作成します。Amazon EC2 は公開キーを保存し、ファイルに保存するプライベートキーを表示します。PEM エンコードされた PKCS#1 形式 (暗号化なし) で、プライベートキーが返されます。指定した名前のキーがすでに存在する場合、Amazon EC2 からエラーが返されます。
- `--key-name` (文字列) — キーペアの一意の名前。

制約: 最大 255 字の ASCII 文字。

- `[-tag-specifications]` (リスト) — 新しいキーペアに適用するタグ。

```
{
  "ResourceType": "image|"instance|"key-pair|"launch-template|"security-group",
  "Tags": [
    {
      "Key": "string",
      "Value": "string"
    }
    ...
  ]
}
...
```

- [import-key-pair](#) —
 - `--key-name` (文字列) — キーペアの一意の名前。

制約: 最大 255 字の ASCII 文字。

- `--public-key-material` (BLOB) – パブリックキー。API コールの場合、テキストは base64 でエンコードされている必要があります。コマンドラインツールでは、base64 エンコーディングが自動的に実行されます。
- `[-tag-specifications]` (リスト) — 新しいキーペアに適用するタグ。

```
{
  "ResourceType": "image"|"instance"|"key-pair"|"launch-template"|"security-group",
  "Tags": [
    {
      "Key": "string",
      "Value": "string"
    }
    ...
  ]
}
```

- [describe-key-pairs](#) –

`[-filters]` (リスト) — フィルター。

- `key-pair-id` – キーペアの ID。
- `key-name` — キーペア名。
- `tag-key` — リソースに割り当てられたタグのキー。このフィルターを使用して、タグの値に関係なく、特定のキーでタグが割り当てられているすべてのリソースを検索します。
- `[-tag-specifications]` (リスト) — 新しいキーペアに適用するタグ。
- `tag:key` — リソースに割り当てられたタグのキーと値の組み合わせ。フィルター名にタグキーを使用し、タグ値をフィルター値として使用します。例えば、キー `Owner` と値 `Team A` のタグを持つすべてのリソースを検索するには、`tag:Owner` のフィルター名と `Team A` のフィルター値を指定します。

```
{
  "Name": "string",
  "Values": ["string", ...]
}
...
```

- `[-key-names]` (リスト) — キーペア名。

デフォルト: すべてのキーペアを説明します。

- `[-key-pair-ids]` (リスト) – キーペアIDs。

- [delete-key-pair](#) –
 - [--key-name] (文字列) — キーペア名。
 - [--key-pair-id] (文字列) – キーペアの ID。

サポートされる Amazon EC2 互換 API オペレーション

以下は、Snowball Edge で使用できる Amazon EC2 互換 API オペレーションと、「Amazon EC2 API リファレンス」内の説明へのリンクです。Amazon EC2 互換 API コールでは、Signature Version 4 (SigV4) 署名が必要です。AWS CLI または AWS SDK を使用してこれらの API コールを行う場合、SigV4 署名が処理されます。それ以外の場合は、お客様独自の SigV4 署名ソリューションを実装する必要があります。詳細については、「[ローカルの Amazon S3 認証情報の取得と使用](#)」を参照してください。

- [AssociateAddress](#) – Elastic IP アドレスをインスタンスまたはネットワークインターフェイスに関連付けます。
- [AttachVolume](#) – 以下のリクエストパラメータがサポートされています。
 - Device
 - InstanceId
 - VolumeId
- [AuthorizeSecurityGroupEgress](#) — Snowball Edge デバイスで使用する 1 つ以上の出カールールをセキュリティグループに追加します。具体的には、このアクションでは、1 つ以上の IPv4 CIDR アドレス範囲にインスタンスがトラフィックを送信できます。
- [AuthorizeSecurityGroupIngress](#) – セキュリティグループに 1 つ以上の進入ルールを追加します。呼び出すときは `AuthorizeSecurityGroupIngress`、`GroupName` または `GroupId` の値を指定する必要があります。
- [CreateVolume](#) – 以下のリクエストパラメータがサポートされています。
 - SnapshotId
 - Size
 - VolumeType
 - TagSpecification.N
- [CreateLaunchTemplate](#) – 以下のリクエストパラメータがサポートされています。
 - ImageId
 - InstanceType

- SecurityGroupIds
- TagSpecifications
- UserData
- [CreateLaunchTemplateVersion](#)
- [CreateTags](#) – 以下のリクエストパラメータがサポートされています。
 - AMI
 - Instance
 - Launch template
 - Security group
- [CreateSecurityGroup](#) – Snowball Edge にセキュリティグループを作成します。最大で 50 個のセキュリティグループを作成できます。セキュリティグループを作成するときに、任意のフレンドリ名を指定します。
- [DeleteLaunchTemplate](#)
- [DeleteLaunchTemplateVersions](#)
- [DeleteSecurityGroup](#) – セキュリティグループを削除します。インスタンスに関連付けられている、または他のセキュリティグループによって参照されているセキュリティグループを削除しようとすると、DependencyViolation でそのオペレーションは失敗します。
- [DeleteTags](#) – 指定されたリソースセットから指定されたタグセットを削除します。
- [DeleteVolume](#) – 以下のリクエストパラメータがサポートされています。
 - VolumeId
- [DescribeAddresses](#)
- [DescribeImages](#)
- [DescribeInstanceAttribute](#) – 以下の属性がサポートされています。
 - instanceType
 - userData
- [DescribeInstanceStatus](#)
- [DescribeLaunchTemplates](#)
- [DescribeLaunchTemplateVersions](#)
- [DescribeInstances](#)

- [DescribeSecurityGroups](#) – 1 つ以上のセキュリティグループについて説明します。
DescribeSecurityGroupsはページ分割されたオペレーションです。複数の API コールを発行して、結果のデータセット全体を取得できます。
- [DescribeTags](#) – このコマンドでは、次のフィルターがサポートされています。
 - resource-id
 - resource-type — AMI あるいはコンピューティングインスタンスのみ
 - key
 - value
- [DescribeVolume](#) – 以下のリクエストパラメータがサポートされています。
 - MaxResults
 - NextToken
 - VolumeId.N
- [DetachVolume](#) – 以下のリクエストパラメータがサポートされています。
 - Device
 - InstanceId
 - VolumeId
- [DisassociateAddress](#)
- [GetLaunchTemplateData](#)
- [ModifyLaunchTemplate](#)
- [ModifyInstanceAttribute](#) – userData 属性のみがサポートされています。
- [RevokeSecurityGroupEgress](#) – セキュリティグループから 1 つ以上の出カールールを削除します。
- [RevokeSecurityGroupIngress](#) – セキュリティグループへの 1 つ以上の進入ルールを取り消します。を呼び出すときは RevokeSecurityGroupIngress、group-nameまたは の値を指定する必要がありますgroup-id。
- [RunInstances](#) –

Note

インスタンスのサイズとタイプに応じて、Snowball Edge 上でコンピューティングインスタンスを起動する際に、最大で 1 時間半かかることがあります。

StartInstances

- [StopInstances](#) – 停止したインスタンスに関連付けられたリソースは保持されます。これらのリソースを解放するには、インスタンスを終了させます。ただし、関連付けられているすべてのデータが削除されます。
- [TerminateInstances](#)

起動テンプレートを使用した Amazon EC2 互換インスタンスの自動起動

起動テンプレートと Snowball Edge クライアント起動設定コマンドを使用して、AWS Snowball Edge デバイスで Amazon EC2-compatible インスタンスを自動的に起動できます。

起動テンプレートには、Snowball Edge で Amazon EC2 互換インスタンスを作成するために必要な設定情報が含まれています。起動テンプレートを使用して起動パラメータを保存できるため、Snowball Edge で EC2 互換インスタンスを起動するたびにパラメータを指定する必要がありません。

Snowball Edge で自動起動設定を使用する際に、Amazon EC2 互換インスタンスを起動するためのパラメータを設定します。Snowball Edge の設定後、再起動してロックを解除すると、自動起動設定を使用して、指定したパラメータでインスタンスが起動します。自動起動設定を使用して起動されたインスタンスが停止した場合は、デバイスのロックを解除すると、インスタンスの実行が開始されます。

Note

自動起動設定を初めて指定した後は、デバイスを再起動してインスタンスを起動します。以降、(計画済みまたは計画外の再起動後の) インスタンスの起動はすべて、デバイスのロック解除後に自動的に行われます。

起動テンプレートを使用して、Amazon EC2 互換インスタンスを起動するとき、そのインスタンスの Amazon マシンイメージ (AMI) ID、インスタンスタイプ、ユーザーデータ、セキュリティグループ、タグを指定できます。サポートされているインスタンスタイプについては、「[Snowball Edge デバイスのコンピューティングインスタンスのクォータ](#)」を参照してください。

Snowball Edge で EC2 互換インスタンスを自動起動させるには、以下のステップに従います。

1. AWS Snowball Edge デバイスを注文するときは、コンピューティングインスタンスを備えた Snow Family デバイスを注文するジョブを作成します。詳細については、「[コンピューティングジョブの作成](#)」を参照してください。

2. Snowball Edge を入手したら、そのロックを解除します。
3. EC2 互換 API コマンド `aws ec2 create-launch-template` を使用して、起動テンプレートを作成します。
4. Snowball Edge クライアントコマンド `snowballEdge create-autostart-configuration` を使用して、EC2 互換インスタンスの起動テンプレートをネットワーク設定にバインドします。詳細については、「[Amazon EC2 互換インスタンスを自動起動するための起動設定の作成](#)」を参照してください。
5. デバイスを再起動し、そのロックを解除します。起動テンプレートで指定した属性と、Snowball Edge クライアントコマンド `create-autostart-configuration` を使用して、EC2 互換インスタンスが自動起動します。

実行中のインスタンスのステータスを表示するには、EC2 互換 API コマンド `describe-autostart-configurations` を使用します。

Note

起動テンプレート AWS Snowball をサポートするためのコンソールまたはジョブ管理 API はありません。AWS Snowball Edge デバイスで EC2 互換インスタンスを自動起動させるには、EC2 互換コマンドと Snowball Edge クライアント CLI コマンドを使用します。

Amazon EC2 互換インスタンスで Snow 用インスタンスメタデータサービスを使用する

IMDS for Snow は、Snow 上の Amazon EC2 互換インスタンスにインスタンスメタデータサービス (IMDS) を提供します。インスタンスメタデータは、インスタンスに関する情報のカテゴリです。ホスト名、イベント、セキュリティグループなどのカテゴリが含まれます。IMDS for Snow を使用すると、インスタンスメタデータを使用して、Amazon EC2 互換インスタンスの起動時に指定したユーザーデータにアクセスすることもできます。例えば、IMDS for Snow を使用して、インスタンスを設定するためのパラメータを指定したり、単純なスクリプトにこれらのパラメータを含めたりすることができます。汎用 AMI をビルドし、ユーザーデータを使って起動時に提供された設定ファイルを変更することができます。

インスタンスメタデータ、ユーザーデータ、および Snow EC2 互換インスタンスについては、本ガイドの「[サポートされているインスタンスメタデータとユーザーデータ](#)」を参照してください。

⚠ Important

インスタンスメタデータおよびユーザーデータにはそのインスタンス自体内からのみアクセスできるものの、データは認証または暗号化手法によって保護されていません。インスタンス、そしてインスタンス上で実行される任意のソフトウェアに対して直接アクセス権がある可能性がある人は、メタデータを表示できます。そのため、パスワードまたは存続期間の長い暗号化キーなどの機密データは、ユーザーデータとして保管しないようにしてください。

ℹ Note

このセクションの例では、インスタンスメタデータサービスの IPv4 アドレスを使用します (169.254.169.254)。リンクローカル IPv6 アドレスを使用したインスタンスメタデータの取得はサポートされていません。

トピック

- [IMDS バージョン](#)
- [IMDSv1 と IMDSv2 を使用したインスタンスメタデータの取得の例](#)

IMDS バージョン

IMDS バージョン 2 または IMDS バージョン 1 を使用して、実行中のインスタンスからインスタンスメタデータにアクセスできます。

- インスタンスメタデータサービスバージョン 2 (IMDSv2) – セッション指向メソッド
- インスタンスメタデータサービスバージョン 1 (IMDSv1) – リクエスト/レスポンス方式

Snow ソフトウェアのバージョンによって、IMDSv1 または IMDSv2、あるいは両方を使用できます。これは、EC2 互換インスタンスで実行されている AMI の種類によっても異なります。Ubuntu 20.04 を実行している AMI など、一部の AMI では IMDSv2 が必要です。インスタンスメタデータサービスは、PUT または GET ヘッダーの有無によって IMDSv1 と IMDSv2 リクエストを区別します。IMDSv2 はこれら両方のヘッダーを使用します。IMDSv1 は GET ヘッダーのみを使用します。

AWS では、IMDSv2 のセキュリティが高いため、IMDSv1 ではなく IMDSv2 の使用を推奨しています。詳細については、「[EC2 Instance Metadata Service の拡張により、オープンファイアウォール](#)

[ル、リバースプロキシ、および SSRF の脆弱性に対して多層防御を追加する](#)」を参照してください。

IMDSv2

IMDSv2 は、セッション指向リクエストを使用します。セッション指向リクエストでは、セッション期間を定義するセッショントークンを作成します。セッション期間は、最短で 1 秒、最長で 6 時間に設定可能です。この期間中、以降のリクエストで同じセッショントークンを使用できます。この期間の期限が切れたら、将来のリクエストに使用する新しいセッショントークンを作成する必要があります。

次の例では、Linux シェルスクリプトと IMDSv2 を使って、最上位インスタンスメタデータ項目を取得します。この例では以下を行います。

1. PUT リクエストを使って、6 時間 (21,600 秒) 持続するセッショントークンを作成する
2. セッショントークンヘッダーを TOKEN という変数に格納する
3. トークンを使って最上位のメタデータ項目をリクエストする

2 つの個別のコマンドを実行することも、それらを組み合わせることもできます。

個別のコマンド

最初に、次のコマンドを使用してトークンを生成します。

Note

X-aws-ec2-metadata-token-ttl-seconds は必須ヘッダーです。このヘッダーが含まれていない場合は、「400 - Missing or Invalid Parameters」エラーコードが表示されます。

```
[ec2-user ~]$ TOKEN=$(curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-metadata-token-ttl-seconds: 21600")
```

その後、次のコマンドを使用して、トークンを使用して最上位のメタデータ項目を生成します。

```
[ec2-user ~]$ curl -H "X-aws-ec2-metadata-token: $TOKEN" -v http://169.254.169.254/latest/meta-data/
```

組み合わせられたコマンド

トークンを保存し、コマンドを組み合わせることができます。次の例では、上記の 2 つのコマンドを組み合わせ、セッショントークンヘッダーを TOKEN という変数に格納します。

Note

トークンの作成時にエラーが発生した場合は、有効なトークンではなくエラーメッセージが変数に格納され、コマンドは機能しません。

Example 組み合わせられたコマンド

```
[ec2-user ~]$ TOKEN=$(curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-metadata-token-ttl-seconds: 21600" \
&& curl -H "X-aws-ec2-metadata-token: $TOKEN" -v http://169.254.169.254/latest/meta-data/)
```

トークンを作成した後、期限切れになるまで再使用することができます。次のコマンド例では、インスタンスの起動に使用される AMI の ID を取得し、前の例で \$TOKEN で作成されたトークンに格納します。

Example トークンの再利用

```
[ec2-user ~]$ curl -H "X-aws-ec2-metadata-token: $TOKEN" -v http://169.254.169.254/latest/meta-data/ami-id
```

IMDSv2 を使用してインスタンスメタデータをリクエストする場合、リクエストは次のルールに従う必要があります。

1. PUT リクエストを使って、インスタンスメタデータサービスに対してセッションを開始します。PUT リクエストは、インスタンスメタデータサービスに対する後続の GET リクエストに含ま

れている必要のあるトークンを返します。このトークンは、IMDSv2を使ってメタデータにアクセスするのに必要です。

2. トークンを、インスタンスメタデータサービスに対するすべてのGETリクエストに含めます。
 - a. トークンはインスタンス固有のキーです。トークンは他の EC2 互換インスタンスでは有効ではなく、生成されたインスタンスの外部で使用しようとするすると拒否されます。
 - b. PUTリクエストには、トークンの有効期限 (TTL) を最大 6 時間 (21,600 秒) まで秒単位で指定するヘッダーが含まれている必要があります。トークンは論理的セッションを表します。TTL は、トークンが有効な時間の長さ、つまりセッションの期間を指定します。
 - c. トークンの期限が切れた後、インスタンスメタデータにアクセスし続けるためには、別の PUT リクエストを使用して新しいセッションを作成する必要があります。
 - d. 各リクエストについてトークンを再使用するか、あるいは新しいトークンを作成することを選択できます。少数のリクエストでは、インスタンスメタデータサービスにアクセスする必要があるたびに、トークンを生成してすぐに使用するほうが簡単かもしれませんが、ただし、効率を重視するなら、インスタンスメタデータをリクエストする必要があるたびにPUTリクエストを書くより、トークン期間を長く指定して再使用することができます。それぞれが独自のセッションを表すトークンを同時に使用できる数については、実際的に制限がありません。

HTTP GET および HEAD メソッドは IMDSv2 インスタンスメタデータリクエストで許可されています。PUT リクエストは、X-Forwarded-For ヘッダーが含まれている場合、拒否されます。

デフォルトで、PUT リクエストに対するレスポンスのホップ制限 (有効期限) は IP プロトコルレベルで 1 です。IMDS for Snow には、PUT レスポンスのホップ制限を変更する機能はありません。

IMDSv1

IMDSv1 はリクエスト/レスポンスモデルを使用します。インスタンスメタデータをリクエストするには、GET リクエストをインスタンスメタデータサービスに送信します。

```
[ec2-user ~]$ curl http://169.254.169.254/latest/meta-data/
```

インスタンスメタデータの取得

インスタンスメタデータは実行中のインスタンスから利用できるため、Amazon EC2 コンソールや AWS CLI を使用してアクセスする必要はありません。これは、インスタンスから実行するスクリプトを記述しているときに便利です。例えば、インスタンスメタデータからインスタンスのローカ

ル IP アドレスにアクセスして、外部アプリケーションへの接続を管理できます。インスタンスメタデータはいくつかのカテゴリに分けられます。各インスタンスメタデータカテゴリの説明については、本ガイドの「[サポートされているインスタンスメタデータとユーザーデータ](#)」を参照してください。

実行中のインスタンス内からインスタンスメタデータのすべてのカテゴリを表示するには、次の IPv4 URI を使用します。

```
http://169.254.169.254/latest/meta-data/
```

これらの IP アドレスは、リンクローカルアドレスであり、インスタンスからのみ使用することが可能です。詳細については、Wikipedia の「[リンクローカルアドレス](#)」を参照してください。

レスポンスおよびエラーメッセージ

すべてのインスタンスメタデータがテキスト (HTTP コンテンツタイプ text/plain) として返されます。

特定のメタデータリソースのリクエストでは、適切な値が返されます。リソースが使用できない場合は「404 - Not Found」HTTP エラーコードが返されます。

一般的なメタデータリソースに対するリクエスト (URI が / 文字で終わる場合) は、使用可能なリソースのリストを返すか、そのようなリソースがない場合は「404 - Not Found」HTTP エラーコードを返します。リスト項目は個別の行に表示され、各行の末尾には改行記号 (ASCII 文字コード 10) が付いています。

IMDSv1 を使って行われたリクエストでは、次の HTTP エラーコードが返されます。

- 400 - Missing or Invalid Parameters – PUT リクエストは無効です。
- 401 - Unauthorized – GET リクエストで無効なトークンが使用されています。推奨されるアクションは新しいトークンを生成することです。
- 403 - Forbidden – リクエストが許可されていないか、インスタンスメタデータサービスがオフになっています。

IMDSv1 と IMDSv2 を使用したインスタンスメタデータの取得の例

次の例は、Linux インスタンスで使用できるコマンドを示しています。

Example 使用可能なインスタンスメタデータのバージョンの取得

次の例では、使用できるインスタンスメタデータのバージョンを取得しています。各バージョンは、新しいインスタンスのメタデータカテゴリがリリースされたときのインスタンスメタデータビルドを参照します。以前のバージョンに存在する構造および情報に依存するスクリプトがある場合は、以前のバージョンを使用することができます。

IMDSv2

```
[ec2-user ~]$ TOKEN=`curl -X PUT "http://192.0.2.0/latest/api/token" -H "X-aws-ec2-
metadata-token-ttl-seconds: 21600"` && curl -H "X-aws-ec2-metadata-token: $TOKEN" -v
http://192.0.2.0/
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
 Dload  Upload  Total  Spent    Left  Speed
 100    56    100    56      0     0    3733    0    0  --:--:--
--:--:-- --:--:-- 3733
* Trying 192.0.2.0...
* TCP_NODELAY set
* Connected to 192.0.2.0 (192.0.2.0) port 80 (#0)
> GET / HTTP/1.1
> Host: 192.0.2.0
> User-Agent: curl/7.61.1
> Accept: */*
> X-aws-ec2-metadata-token:
MDAXcxNFLbAwJIYx8KzgNckcHTdxT4Tt69TzpKExlXKTULHIQnjEtXvD
>
* HTTP 1.0, assume close after body
< HTTP/1.0 200 OK
< Date: Mon, 12 Sep 2022 21:58:03 GMT
< Content-Length: 274
< Content-Type: text/plain
< Server: EC2ws
<
1.0
2007-01-19
2007-03-01
2007-08-29
2007-10-10
2007-12-15
2008-02-01
2008-09-01
2009-04-04
```

```
2011-01-01
2011-05-01
2012-01-12
2014-02-25
2014-11-05
2015-10-20
2016-04-19
2016-06-30
2016-09-02
2018-03-28
2018-08-17
2018-09-24
2019-10-01
2020-10-27
2021-01-03
2021-03-23
* Closing connection 0
```

IMDSv1

```
[ec2-user ~]$ curl http://192.0.2.0/
1.0
2007-01-19
2007-03-01
2007-08-29
2007-10-10
2007-12-15
2008-02-01
2008-09-01
2009-04-04
2011-01-01
2011-05-01
2012-01-12
2014-02-25
2014-11-05
2015-10-20
2016-04-19
2016-06-30
2016-09-02
2018-03-28
2018-08-17
```

```
2018-09-24
2019-10-01
2020-10-27
2021-01-03
2021-03-23
latest
```

Example 最上位のメタデータ項目の取得

次の例では、最上位のメタデータ項目を取得します。最上位のメタデータ項目の詳細については、本ガイドの「[サポートされているインスタンスメタデータとユーザーデータ](#)」を参照してください。

IMDSv2

```
[ec2-user ~]$ TOKEN=`curl -X PUT "http://192.0.2.0/latest/api/token" -H "X-aws-ec2-metadata-token-ttl-seconds: 21600"` && curl -H "X-aws-ec2-metadata-token: $TOKEN" -v http://192.0.2.0/latest/meta-data/
ami-id
hostname
instance-id
instance-type
local-hostname
local-ipv4
mac
network/
reservation-id
security-groups
```

IMDSv1

```
[ec2-user ~]$ curl http://192.0.2.0/latest/meta-data/
ami-id
hostname
instance-id
instance-type
local-hostname
local-ipv4
mac
```

```
network/  
reservation-id  
security-groups
```

Example 最上位メタデータの値の取得

次の例では、前の例で取得された一部の最上位メタデータ項目の値を取得します。IMDSv2 リクエストは、前の例のコマンドで作成された保管済みトークン (期限内であると仮定) を使用します。

ami-id IMDSv2

```
curl -H "X-aws-ec2-metadata-token: $TOKEN" -v http://192.0.2.0/latest/meta-data/  
ami-id ami-0abcdef1234567890
```

ami-id IMDSv1

```
curl http://192.0.2.0/latest/meta-data/ami-id ami-0abcdef1234567890
```

reservation-id IMDSv2

```
[ec2-user ~]$ curl -H "X-aws-ec2-metadata-token: $TOKEN" -v http://192.0.2.0/  
latest/meta-data/reservation-id r-0efghijk987654321
```

reservation-id IMDSv1

```
[ec2-user ~]$ curl http://192.0.2.0/latest/meta-data/reservation-id \  
r-0efghijk987654321
```

local-hostname IMDSv2

```
[ec2-user ~]$ curl -H "X-aws-ec2-metadata-token: $TOKEN" -v http://192.0.2.0/latest/meta-data/local-hostname ip-00-000-00-00
```

local-hostname IMDSv1

```
[ec2-user ~]$ curl http://192.0.2.0/latest/meta-data/local-hostname ip-00-000-00-00
```

Amazon EC2 互換インスタンスでのブロックストレージの使用

Snowball Edge のブロックストレージでは、アプリケーションのニーズに基づいてブロックストレージを追加または削除できます。Amazon EC2 互換インスタンスにアタッチしたボリュームは、インスタンスの有効期間とは無関係に存続するストレージボリュームとして公開されます。使い慣れた Amazon EBS API を使用して、ブロックストレージを管理できます。

特定の Amazon EBS コマンドは、EC2 互換エンドポイントを使用することでサポートされます。サポートされているコマンドには、attach-volume、create-volume、delete-volume、detach-volume、および describe-volumes があります。これらのコマンドの詳細については、[Snowball Edge でサポートされる Amazon EC2 互換の AWS CLI コマンドのリスト](#) を参照してください。

Important

ボリュームをデタッチする前に、デバイスに存在するすべてのファイルシステムをオペレーティングシステム内でアンマウントしてください。この操作を行わないと、データが失われる可能性があります。

Amazon EBS ボリュームのクォータ、およびデバイスの Amazon EBS ボリュームとクラウドの Amazon EBS ボリュームの違いを以下に示します。

- Amazon EBS ボリュームは、このボリュームをホストしているデバイスで実行中の EC2 互換インスタンスでのみ使用できます。
- ボリュームタイプは、キャパシティー最適化 HDD (sbg1) またはパフォーマンス最適化 SSD (sbp1) のいずれかに制限されます。デフォルトのボリュームタイプは sbg1 です。

- Snowball Edge は、Amazon S3 オブジェクトと Amazon EBS の間で HDD メモリを共有します。で HDD ベースのブロックストレージを使用すると AWS Snowball Edge、Amazon S3 オブジェクトで使用できるメモリの量が減ります。同様に、Amazon S3 オブジェクトを使用すると、HDD ボリュームの Amazon EBS ブロックストレージに使用可能なメモリの量が減ります。
- Amazon EC2 互換のルートボリュームは常に IDE ドライバーを使用します。追加の Amazon EBS ボリュームは、Virtio ドライバー (使用可能な場合) を優先的に使用します。Virtio ドライバーを使用できない場合、SBE はデフォルトで IDE ドライバーを使用します。パフォーマンスが向上するため、Virtio ドライバーをお勧めします。
- Amazon EBS ボリュームを作成する際、encrypted パラメータはサポートされません。ただし、デバイス上のすべてのデータは、デフォルトで暗号化されます。。
- ボリュームの範囲は 1 GB ~ 10 TB です。
- 最大 10 個の Amazon EBS ボリュームを 1 つの EC2 互換インスタンスにアタッチできます。
- AWS Snowball Edge デバイスで保持できる Amazon EBS ボリュームの数に正式な制限はありません。ただし、Amazon EBS ボリュームの総容量は、デバイスで使用可能なスペースによって制限されます。

Snowball Edge デバイスのセキュリティグループ

セキュリティグループは、1 つ以上のインスタンスのトラフィックを制御する仮想ファイアウォールとして機能します。インスタンスを起動するときに、1 つ以上のセキュリティグループとインスタンスを関連付けます。各セキュリティグループに対してルールを追加して、関連付けられたインスタンスに対するトラフィックを許可できます。詳細については、「Amazon EC2 ユーザーガイド」の「[Amazon EC2 security groups for Linux instances](#)」(Linux インスタンス用の Amazon EC2 セキュリティグループ) を参照してください。

Snowball Edge デバイスのセキュリティグループは、AWS クラウドのセキュリティグループに類似しています。Virtual Private Cloud (VPC) は、Snowball Edge デバイスではサポートされません。

Snowball Edge のセキュリティグループと EC2-VPC のセキュリティグループの上記以外の違いを次に示します。

- 各 Snowball Edge のセキュリティグループは、最大 50 個に制限されています。
- デフォルトのセキュリティグループでは、すべて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許可します。
- ローカルインスタンス間のトラフィックは、プライベートインスタンス IP アドレスまたはパブリック IP アドレスのいずれかを使用します。たとえば、SSH を使用してインスタンス A からイ

インスタンス B に接続するとします。この場合、ターゲットの IP アドレスは、セキュリティグループのルールでこのトラフィックが許可される場合に、インスタンス B のパブリック IP あるいはプライベート IP アドレスのいずれかを使用できます。

- AWS CLI アクションと API コールにリストされているパラメータのみがサポートされます。これらは、通常は EC2-VPC インスタンスでサポートされているパラメータのサブセットです。

サポートされている AWS CLI アクションの詳細については、「」を参照してください[Snowball Edge でサポートされる Amazon EC2 互換の AWS CLI コマンドのリスト](#)。サポートされている API オペレーションの詳細については、「[サポートされる Amazon EC2 互換 API オペレーション](#)」を参照してください。

サポートされるインスタンスメタデータとユーザーデータ

インスタンスメタデータは、インスタンスに関するデータで、実行中のインスタンスを設定または管理するために使用します。Snowball Edge では、コンピューティングインスタンス用の、インスタンスメタデータカテゴリのサブセットがサポートされます。詳細については、「Amazon EC2 ユーザーガイド」の「[Instance metadata and user data](#)」（インスタンスメタデータとユーザーデータ）を参照してください。

以下のカテゴリがサポートされています。他のカテゴリを使用すると、404 エラーメッセージが返されます。

Snowball Edge でサポートされるインスタンスメタデータカテゴリ

[データ]	説明
ami-id	インスタンスの起動に使用される AMI ID。
hostname	インスタンスのプライベート IPv4 DNS ホスト名。
instance-id	このインスタンスの ID。
instance-type	インスタンスの種類。
local-hostname	インスタンスのプライベート IPv4 DNS ホスト名。
local-ipv4	インスタンスのプライベート IPv4 アドレス。

[データ]	説明
mac	インスタンスのメディアアクセスコントロール (MAC) アドレス。
network/interfaces/macs/ <i>mac</i> /local-hostname	インターフェイスのローカルホスト名。
network/interfaces/macs/ <i>mac</i> /local-ipv4s	インターフェイスに関連付けられたプライベート IPv4 アドレス。
network/interfaces/macs/ <i>mac</i> /mac	インスタンスの MAC アドレス。
network/interfaces/macs/ <i>mac</i> /public-ipv4s	インターフェイスに関連付けられた Elastic IP アドレス。
public-ipv4	パブリック IPv4 アドレス。
public-keys/0/openssh-key	パブリックキー。インスタンスの起動時に指定された場合のみ返されます。
reservation-id	予約の ID。
userData	起動時にインスタンスに指示を送るシェルスクリプト。

Snowball Edge でサポートされるインスタンス動的データカテゴリ

[データ]	説明
インスタンスアイデンティティ/ドキュメント	インスタンスの属性を含む JSON。instanceId、imageId、privateIp、および instanceType のみが値を持ちます。返されるその他の属性は null です。詳細については、「Amazon EC2 ユーザーガイド 」の「 インスタンス ID ドキュメント 」を参照してください。 Amazon EC2

Snowball コンピューティングインスタンスのユーザーデータ

ユーザーデータは、Snowball Edge のコンピューティングインスタンスのシェルスクリプトで使用するためにサポートされます。シェルスクリプトを使用して、起動時にインスタンスに指示を送ることができます。ユーザーデータは、`modify-instance-attribute` AWS CLI コマンドまたは `ModifyInstanceAttribute` API アクションを使用して変更できます。

ユーザーデータを変更するには

1. `stop-instances` AWS CLI コマンドを使用してコンピューティングインスタンスを停止します。
2. `modify-instance-attribute` AWS CLI コマンドを使用して、`userData` 属性を変更します。
3. `start-instances` AWS CLI コマンドを使用してコンピューティングインスタンスを再起動します。

コンピューティングインスタンスではシェルスクリプトのみがサポートされています。Snowball Edge で実行されるコンピューティングインスタンスでは、`cloud-init` パッケージディレクティブはサポートされません。AWS CLI コマンドの操作の詳細については、[AWS CLI 「コマンドリファレンス」](#)を参照してください。

EC2 互換インスタンスの停止

デバイスで作成した Amazon EC2 互換インスタンスを誤って削除しないように、オペレーティングシステムからインスタンスをシャットダウンしないでください。たとえば、`shutdown` または `reboot` コマンドを使用しないでください。オペレーティングシステム内からインスタンスをシャットダウンすると、[terminate-instances](#) コマンドを呼び出すのと同じ結果になります。

代わりに、[stop-instances](#) コマンドを使用して、保持したい Amazon EC2 互換インスタンスを一時停止します。

Snowball Edge デバイスのコンピューティングインスタンスのトラブルシューティング

以下で、コンピューティングインスタンスを持つ Snowball Edge ジョブのトラブルシューティングに関するヒントを確認できます。

トピック

- [仮想ネットワークインターフェイスの IP アドレスが 0.0.0.0 です](#)
- [大型のコンピューティングインスタンス起動時に Snowball Edge がハングする](#)
- [インスタンスに 1 つのルートボリュームがあります](#)
- [保護されていないプライベートキーファイルエラー](#)

仮想ネットワークインターフェイスの IP アドレスが 0.0.0.0 です

この問題は、仮想ネットワークインターフェイス (VNIC) に関連付けられた物理ネットワークインターフェイス (NIC) の IP アドレスも 0.0.0.0. である場合に発生します。この状況は、NIC で IP アドレスが設定されていない場合 (たとえば、デバイスの電源をオンにしたばかりの場合など) に発生します。また、誤ったインターフェイスを使用している場合に発生することがあります。たとえば、SFP+ インターフェイスの IP アドレスを取得しようとしたが RJ45 インターフェイスがネットワークに接続されていない場合などです。

実行するアクション

これが発生した場合、次のことを実行できます。

- IP アドレスを持つ NIC に関連付けられている、新しい VNIC を作成します。詳細については、「[コンピューティングインスタンスのネットワーク設定](#)」を参照してください。
- 既存の VNIC を更新します。詳細については、「[仮想ネットワークインターフェイスの更新](#)」を参照してください。

大型のコンピューティングインスタンス起動時に Snowball Edge がハングする

Snowball Edge がインスタンスの起動を停止した可能性があります。これは実際の状況とは異なります。最大のコンピューティングインスタンスが起動するまでには 1 時間以上かかることがあります。

インスタンスのステータスを確認するには、Snowball Edge の HTTP または HTTPS Amazon EC2-compatible エンドポイントに対して AWS CLI コマンド `aws ec2 describe-instances` を実行します。

インスタンスに 1 つのルートボリュームがあります

インスタンスには、設計上 1 つのルートボリュームがあります。すべての sbe インスタンスのルートボリュームは 1 つですが、Snowball Edge ではアプリケーションのニーズに基づいて、ブロック

ストレージを追加または削除できます。詳細については、「[Amazon EC2 互換インスタンスでのブロックストレージの使用](#)」を参照してください。

保護されていないプライベートキーファイルエラー

このエラーは、コンピューティングインスタンス上の .pem ファイルに十分な読み取り/書き込みのアクセス許可がない場合に発生する可能性があります。

実行するアクション

この問題は、以下の手順に従ってファイルのアクセス許可を変更することで解決できます。

1. ターミナルを開き、.pem ファイルを保存した場所に移動します。
2. 次のコマンドを入力します。

```
chmod 400 filename.pem
```

Snow Family デバイスの Amazon S3 互換ストレージを使用する

Snow Family デバイスの Amazon S3 互換ストレージは、高い耐障害性とスケーラビリティを備えた安全なオブジェクトストレージを提供し、Amazon S3 API 機能セットを接続されていない堅牢なモバイルエッジ環境にも拡張します。Snow Family デバイスの Amazon S3 互換ストレージを使用すると、お客様は Snow Family デバイスでエッジコンピューティング向けにデータを保存し、可用性の高いアプリケーションを実行できます。

Snowball Edge デバイスで Amazon S3 バケットを作成して、ローカルデータアクセス、ローカルデータ処理、データレジデンシーを必要とするアプリケーション用に、オンプレミスのオブジェクトを保存および取得できます。Snow Family デバイスの Amazon S3 互換ストレージは、SNOW という新しいストレージクラスを提供します。このストレージクラスは、S3 API を使用し、複数の Snowball Edge デバイス間でデータを永続的かつ冗長的に保存するように設計されています。Snowball Edge バケットで、Amazon S3 と同じ API と機能 (バケットのライフサイクルポリシー、暗号化、タグ付けなど) を使用できます。デバイスまたはデバイスが返されると AWS、Snow Family デバイスの Amazon S3 互換ストレージに作成または保存されたすべてのデータが消去されます。詳細については、「[ローカルコンピューティングとストレージ専用ジョブ](#)」を参照してください。

Snow Family デバイスの Amazon S3 互換ストレージは、スタンドアロン構成またはクラスター構成でデプロイできます。スタンドアロン構成では、デバイスに S3 容量をプロビジョニングでき、残りはブロックストレージとして利用できます。クラスター構成では、すべてのデータディスク容量が

S3 ストレージに使用されます。クラスターは、最低 3 台、最大 16 台のデバイスで構成できます。クラスターのサイズによって、S3 サービスは 1 台または 2 台のデバイスの耐障害性を維持するように設計されています。

を使用すると AWS DataSync、Snowball Edge デバイス上の Snow Family デバイスの Amazon S3 互換ストレージと AWS ストレージサービス間でオブジェクトを転送できます。詳細については、「ユーザーガイド」の「[Snowball Edge での S3 互換ストレージを使用した転送の設定](#) AWS DataSync」を参照してください。

以下は、Snow Family デバイスの Amazon S3 互換ストレージのストレージ容量と、Snow Family デバイスの Amazon S3 互換ストレージを使用するスタンドアロンデバイスのブロックストレージ容量です。クラスターの耐障害性とストレージ容量については、「」を参照してください[this table](#)。

Snowball Edge Compute Optimized and Compute Optimized with GPU

Snow Family デバイスの Amazon S3 互換ストレージのストレージ容量と、Snowball Edge Compute Optimized (AMD EPYC Gen1、HDD、オプションの GPU 搭載) デバイスのブロックストレージのストレージ容量

Snow Family デバイスの Amazon S3 互換ストレージのストレージ容量 (TB 単位)	ブロックストレージ容量 (TB 単位)
2.5	41
5.5	37
8.5	33
11	29
14	25
17	21
19.5	17
22.5	13
25.5	9
28.5	5

Snow Family デバイスの Amazon S3 互換ストレージのストレージ容量 (TB 単位)	ブロックストレージ容量 (TB 単位)
31	1

Snowball Edge Compute Optimized with NVMe storage

Snow Family デバイスの Amazon S3 互換ストレージのストレージ容量と、Snowball Edge Compute Optimized (AMD EPYC Gen2、NVMe 搭載のコンピューティング最適化) デバイスのブロックストレージのストレージ容量

Snow Family デバイスの Amazon S3 互換ストレージのストレージ容量 (TB 単位)	ブロックストレージ容量 (TB 単位)
3	17.5
5.5	14.5
10.5	8.5
12	6.5
13	5.5
16.5	1.5

Snowball Edge storage optimized 210 TB

Snow Family デバイスの Amazon S3 互換ストレージのストレージ容量と Snowball Edge ストレージ最適化 210 TB デバイスのブロックストレージ

Snow Family デバイスの Amazon S3 互換ストレージのストレージ容量 (TB 単位)	ブロックストレージ容量 (TB 単位)
20	2.0.6
40	182
60	158

Snow Family デバイスの Amazon S3 互換ストレージのストレージ容量 (TB 単位)	ブロックストレージ容量 (TB 単位)
80	134
100	110
120	86
140	62
160	38
180	14
190	2

Snow Family デバイスの Amazon S3 互換ストレージの仕様

- Snow Family デバイスバケットの最大数は、デバイスまたはクラスターあたり 100 です。
- Snow Family デバイスの S3 のバケット所有者アカウントは、バケット内のすべてのオブジェクトを所有します。
- バケットに対し操作ができるのは、Snow Family デバイスの S3 のバケット所有者アカウントのみです。
- オブジェクトのサイズの制限は、Amazon S3 と同じです。
- Snow Family デバイスの S3 に保存されるすべてのオブジェクトは、ストレージクラスとして SNOW を使用します。
- デフォルトでは、SNOW ストレージクラスに格納されているすべてのオブジェクトは、Amazon S3 管理の暗号化キー (SSE-S3) によるサーバー側の暗号化を使用して保存されます。また、ユーザーが用意した暗号化キー (SSE-C) で、サーバー側の暗号化を使用してオブジェクトを保存するように明示的に選択することもできます。
- Snow Family デバイスにオブジェクトを保存する十分なスペースがない場合、API は容量不足の例外 (ICE) を返します。

トピック

- [Snow Family デバイスの Amazon S3 互換ストレージを注文する](#)

- [Snow Family デバイスでの Amazon S3 互換ストレージのセットアップと起動](#)
- [Snowball Edge デバイスで S3 バケットを操作する](#)
- [Snowball Edge デバイスで S3 オブジェクトを操作する](#)
- [Snow Family デバイスの Amazon S3 互換ストレージでサポートされる REST API アクション](#)
- [Snow Family デバイスの Amazon S3 互換ストレージを Snow デバイスのクラスターで使用する](#)
- [Snow Family デバイスの Amazon S3 互換ストレージでのイベント通知の設定](#)
- [ローカル SMTP 通知の設定](#)

Snow Family デバイスの Amazon S3 互換ストレージを注文する

Snow Family デバイスの Amazon S3 互換ストレージ用にデバイスを注文する手順は、Snowball Edge の注文プロセスと似ています。注文するには、本ガイドの「[Snow Family デバイスを注文するジョブの作成](#)」参照し、注文時に以下の点に留意してください。

- [ジョブタイプを選択] で、[ローカルコンピューティングおよびストレージのみ] を選択します。
- [Snow デバイス] で [Snowball Edge Compute Optimized] を選択します
- [ストレージタイプを選択] で、[Snow Family デバイスの Amazon S3 互換ストレージ] を選択します。
- スタンドアロンデバイスの場合は、[ストレージ容量] で [単一デバイス] を選択し、希望のストレージ容量を選択します。
- クラスターの場合は、[ストレージ容量] で [クラスター] を選択し、希望のストレージ容量と耐障害性を選択します。

Snow Family デバイスでの Amazon S3 互換ストレージのセットアップと起動

からローカル環境にソフトウェアツールをインストールして設定し、Snowball Edge デバイスまたはデバイスのクラスター、および Snow Family デバイスの Amazon S3 互換ストレージとやり取り AWS します。次に、これらのツールを使用して Snowball Edge デバイスまたはクラスターを設定し、Snow Family デバイスで Amazon S3 互換ストレージを起動します。

前提条件

Snow Family デバイスの Amazon S3 互換ストレージでは、Snowball Edge Client と をローカル環境に AWS CLI インストールする必要があります。AWS SDK for .NET および AWS Tools for Windows

を使用して PowerShell、Snow Family デバイスの Amazon S3 互換ストレージを使用することもできます。AWS では、これらのツールの次のバージョンを使用することをお勧めします。

- Snowball Edge Client – 最新バージョンを使用します。詳細については、このガイドの [「Snowball Edge Client のダウンロードとインストール」](#) を参照してください。
- AWS CLI – バージョン 2.11.15 以降。詳細については、[「ユーザーガイド」の「のインストール、更新、アンインストール AWS CLI」](#) を参照してください。AWS Command Line Interface
- AWS SDK for .NET – AWSSDK.S3Control 3.7.304.8 以降。詳細については、[「AWS SDK for .NET」](#) を参照してください。
- AWS Tools for Windows PowerShell – バージョン 4.1.476 以降。詳細については、[AWS Tools for Windows PowerShell ユーザーガイド](#) を参照してください。

ローカル環境のセットアップ

このセクションでは、Snow Family デバイスの Amazon S3 互換ストレージで使用する Snowball Edge Client とローカル環境をセットアップおよび設定する方法について説明します。

使用する環境をセットアップするには

1. Snowball Edge Client の最新バージョンをダウンロードしてインストールします。詳細については、このガイドの [「Snowball Edge Client のダウンロードとインストール」](#) を参照してください。
2. 以下のコマンドを実行してフォルダを設定します。

```
chmod u+x new_cli/bin/snowballEdge
chmod u+x new_cli/jre/bin/java
```

3. new_cli/bin を \$PATH に追加します。
4. snowballEdge configure コマンドを実行します。以下のようなレスポンスが返されます。

```
Configuration will be stored at /home/user/.aws/snowball/config/snowball-
edge.config
```

5. 次の情報を入力します。
 - マニフェストパス。

- ロック解除コード。
- デフォルトのエンドポイント。スタンドアロン Snowball Edge デバイスの場合は、デバイスの IP アドレスを使用します。デバイスのクラスターの場合、はクラスター内の任意のデバイスの IP アドレスを指定します。クライアントからデフォルトのエンドポイントが使用可能かどうかをテストするには、次のようなコマンドを使用します。ポート番号には、9091 (アクティベーションポート)、22 (SSH)、および 8080 (s3 の HTTP エンドポイント) を使用します。

```
telnet snowball_ip port_number
```

6. を使用している場合は AWS SDK for .NET、`clientConfig.AuthenticationRegion` パラメータ値を次のように設定します。

```
clientConfig.AuthenticationRegion = "snow"
```

Snowball Edge デバイスのセットアップ

Snowball Edge での IAM のセットアップ

AWS Identity and Access Management (IAM) は、Snowball Edge デバイスで実行される AWS リソースへのきめ細かなアクセスを有効にするのに役立ちます。IAM を使用して、誰を認証 (サインイン) し、誰にリソースの使用を認可する (アクセス許可を付与する) かを制御します。

IAM は Snowball Edge でローカルでサポートされています。ローカル IAM サービスを使用してロールを作成し、IAM ポリシーをアタッチできます。これらのポリシーを使用して、割り当てられたタスクの実行に必要なアクセスを許可できます。

以下の例では、Amazon S3 API へのフルアクセスを許可します。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "VisualEditor0",
      "Effect": "Allow",
      "Action": "s3:*"
```

```
        "Resource": "*"
    }
]
}
```

その他の IAM ポリシーの例については、「[AWS Snowball Edge デベロッパーガイド](#)」を参照してください。

Snow Family デバイスの Amazon S3 互換ストレージサービスを開始する

Snowball Edge デバイスまたはクラスターで Snow Family デバイスサービスで Amazon S3 互換ストレージを起動するには、以下の手順に従います。

Note

よりユーザーフレンドリーなエクスペリエンスが必要な場合は、を使用してスタンドアロンデバイスまたはデバイスのクラスターの Snow Family デバイスサービスで Amazon S3 互換ストレージを起動できます AWS OpsHub。 [Snow Family デバイスの Amazon S3 互換ストレージをセットアップする](#) を参照してください。

1. 以下のコマンドを実行して、Snowball Edge デバイスまたはデバイスのクラスターのロックを解除します。

- 単一のデバイスの場合

```
snowballEdge unlock-device --endpoint https://snow-device-ip
```

- クラスターの場合

```
snowballEdge unlock-cluster
```

2. 次のコマンドを実行し、Snowball Edge デバイスまたはデバイスのクラスターがロック解除されていることを確認します。

- 単一のデバイスの場合

```
snowballEdge describe-device --endpoint https://snow-device-ip
```

- クラスターの場合

```
snowballEdge describe-cluster --device-ip-addresses [snow-device-1-ip] [snow-device-2-ip] /  
[snow-device-3-ip] [snow-device-4-ip] [snow-device-5-ip] /  
[snow-device-6-ip]
```


3. Snow Family デバイスの Amazon S3 互換ストレージを起動するには、各デバイス (1 台のデバイスかクラスターかにかかわらず) について、以下を実行します。
 - a. 以下の `PhysicalNetworkInterfaceId` コマンドを実行してデバイスの `describe-device` を取得します。

```
snowballEdge describe-device --endpoint https://snow-device-ip
```

- b. 以下の `create-virtual-network-interface` コマンドを 2 回実行して、`s3control` (バケット操作) エンドポイントと `s3api` (オブジェクト操作) エンドポイント用の仮想ネットワークインターフェイス (VNI) を作成します。

```
snowballEdge create-virtual-network-interface --ip-address-assignment  
dhcp --manifest-file manifest --physical-network-interface-id  
"PhysicalNetworkInterfaceId" --unlock-code unlockcode --endpoint https://snow-device-ip
```

これらのコマンドについて詳しくは、「[仮想ネットワークインターフェイスの作成](#)」を参照してください。

 Note

Snow Family デバイスの Amazon S3 互換ストレージを起動すると、デバイスリソースが消費されます。

4. 次の `start-service` コマンドを実行して、Snow Family デバイスサービスで Amazon S3 互換ストレージを起動します。これには、デバイスの IP アドレスと、`s3control` および `s3api` エンドポイント用に作成した VNI の Amazon リソースネーム (ARNs) が含まれます。

1 台のデバイスでサービスを開始するには

```
snowballEdge start-service --service-id s3-snow --device-ip-addresses snow-device-1-ip --virtual-network-interface-arns vni-arn-1 vni-arn-2
```

クラスターでサービスを開始するには :

```
snowballEdge start-service --service-id s3-snow --device-ip-addresses snow-device-1-ip snow-device-2-ip snow-device-3-ip --virtual-network-interface-arns vni-arn-1 vni-arn-2 vni-arn-3 vni-arn-4 vni-arn-5 vni-arn-6
```

--virtual-network-interface-arns では、前のステップで作成したすべての VNI の ARN を含めます。スペースを使用して各 ARN を区切ります。

5. 1 台のデバイスの場合は、以下の describe-service コマンドを実行します。

```
snowballEdge describe-service --service-id s3-snow
```

サービスステータスが Active になるまで待ちます。

クラスターの場合は、以下の describe-service コマンドを入力します。

```
snowballEdge describe-service --service-id s3-snow \  
--device-ip-addresses snow-device-1-ip snow-device-2-ip snow-device-3-ip
```

Snowball Edge デバイスで S3 バケットを操作する

Snowball Edge デバイスで Amazon S3 バケットを作成して、ローカルデータアクセス、ローカルデータ処理、データレジデンシーを必要とするアプリケーション用に、オンプレミスでオブジェクトを保存および取得できます。Snow Family デバイスの Amazon S3 互換ストレージは、SNOW という新しいストレージクラスを提供します。このストレージクラスは、S3 API を使用し、複数の Snowball Edge デバイス間でデータを永続的かつ冗長的に保存するように設計されています。Snowball Edge バケットで、Amazon S3 と同じ API と機能 (バケットのライフサイクルポリシー、暗号化、タグ付けなど) を使用できます。

の使用 AWS CLI

AWS CLIを使用して、デバイスの Amazon S3 バケットを操作するには、以下の手順に従います。

を設定するには AWS CLI

1. `~/.aws/config` でオブジェクトエンドポイントのプロファイルを作成します。

```
[profile your-profile]  
aws_access_key_id = your-access-id  
aws_secret_access_key = your-access-key  
region = snow  
ca_bundle = dev/apps/ca-certs/your-ca_bundle
```

2. デバイスから証明書を取得します。詳細については、「[Snowball Edge デベロッパーガイド](#)」を参照してください。
3. SDK を仮想環境にインストールした場合は、以下のコマンドを使用してアクティブ化します。

```
source your-virtual-environment-name/bin/activate
```

オペレーションを設定したら、AWS CLIの API コールを使用してオペレーションにアクセスできます。以下の例で、*cert* は IAM を使用して取得したデバイス証明書です。

オブジェクト操作へのアクセス

```
aws s3api --profile your-profile list-objects-v2 --endpoint-url  
https://s3api-endpoint-ip
```

バケット操作へのアクセス操作

```
aws s3control --profile your-profile list-regional-buckets --account-id  
bucket-owner --endpoint-url https://s3ctrlapi-endpoint-ip
```

Java SDK の使用

以下の例に従って、Java SDK を使用して Amazon S3 オブジェクトを操作します。

```
import software.amazon.awssdk.services.s3.S3Client;
```

```
import software.amazon.awssdk.auth.credentials.AwsBasicCredentials;
import software.amazon.awssdk.auth.credentials.StaticCredentialsProvider;
import software.amazon.awssdk.http.SdkHttpClient;
import software.amazon.awssdk.http.apache.ApacheHttpClient;
import software.amazon.awssdk.regions.Region;

import java.net.URI;

AwsBasicCredentials creds = AwsBasicCredentials.create(accessKey, secretKey); // set
creds by getting Access Key and Secret Key from snowball edge
SdkHttpClient httpClient =
    ApacheHttpClient.builder().tlsTrustManagersProvider(trustManagersProvider).build(); //
set trust managers provider with client certificate from snowball edge
String s3SnowEndpoint = "10.0.0.0"; // set s3-snow object api endpoint from describe
service

S3Client s3Client =
    S3Client.builder().httpClient(httpClient).region(Region.of("snow")).endpointOverride(new
URI(s3SnowEndpoint)).credentialsProvider(StaticCredentialsProvider.create(creds)).build();
```

バケット ARN 形式

以下に示す Amazon リソースネーム (ARN) 形式を使用して、Snowball Edge デバイスの Amazon S3 バケットを識別できます。

```
arn:partition:s3:snow:account-id:device/device-id/bucket/bucket-name
```

ここで、*partition* は Snowball Edge デバイスを注文したリージョンのパーティションです。*device-id* は、デバイスがスタンドアロンの Snowball Edge デバイスの場合は *job_id*、Snowball Edge クラスターの場合は *cluster_id* です。

Snowball Edge デバイスで S3 バケットを作成する

Snowball Edge デバイスで Amazon S3 バケットを作成して、ローカルデータアクセス、ローカルデータ処理、データレジデンシーを必要とするアプリケーション用に、エッジでオブジェクトを保存および取得できます。Snow Family デバイスの Amazon S3 互換ストレージは、SNOW という新しいストレージクラスを提供します。このストレージクラスは、Amazon S3 を使用し、複数のデバイス間でデータを永続的かつ冗長的に保存するように設計されています。Amazon S3 バケットと同じ API と機能 (バケットのライフサイクルポリシー、暗号化、タグ付けなど) を使用できます。

以下の例では、AWS CLIを使用して Snowball Edge デバイス用の Amazon S3 バケットを作成します。このコマンドを実行するには、ユーザー入力プレースホルダーをユーザー自身の情報に置き換えます。

```
aws s3control --profile your-profile create-bucket --bucket your-snow-bucket --  
endpoint-url https://s3ctrlapi-endpoint-ip
```

を使用したオブジェクトライフサイクル設定の作成と管理 AWS CLI

Amazon S3 ライフサイクルを使用して、Snow Family デバイスの Amazon S3 互換ストレージのストレージ容量を最適化できます。ライフサイクルルールを作成して、オブジェクトが古くなったり、新しいバージョンに置き換えられたりしたときに期限切れにすることができます。ライフサイクルルールを作成、有効化、無効化、または削除できます。Amazon S3 ライフサイクルの詳細については、「[ストレージのライフサイクルの管理](#)」を参照してください。

Note

バケット AWS アカウント を作成する はバケットを所有し、ライフサイクルルールを作成、有効化、無効化、または削除できる唯一のものです。

AWS Command Line Interface (AWS CLI) を使用して、Snow Family デバイスバケットの Amazon S3 互換ストレージのライフサイクル設定を作成および管理するには、以下の例を参照してください。

Snowball Edge バケットにライフサイクル設定を配置する

次の AWS CLI 例では、Snowball Edge バケットにライフサイクル設定ポリシーを配置します。このポリシーでは、フラグ付きのプレフィックス (*myprefix*) とタグを持つすべてのオブジェクトが 10 日後に期限切れになることを指定します。この例を使用するには、各ユーザー入力プレースホルダーをユーザー自身の情報に置き換えます。

まず、ライフサイクル設定ポリシーを JSON ファイルに保存します。この例では、ファイル名は **lifecycle-example.json** です。

```
{  
  "Rules": [{  
    "ID": "id-1",  
    "Filter": {
```



```
    "And": {
      "Prefix": "myprefix",
      "Tags": [{
        "Value": "mytagvalue1",
        "Key": "mytagkey1"
      },
      {
        "Value": "mytagvalue2",
        "Key": "mytagkey2"
      }
    ],
  },
  "Status": "Enabled",
  "Expiration": {
    "Days": 10
  }
}]
}
```

ファイルを保存したら、`put-bucket-lifecycle-configuration` コマンドの一部として JSON ファイルを送信します。このコマンドを使用するには、各ユーザー入力プレースホルダーを独自の情報に置き換えます。

```
aws s3control put-bucket-lifecycle-configuration --bucket
    example-snow-bucket --profile your-profile
    --lifecycle-configuration file://lifecycle-example.json --endpoint-url
    https://s3ctrlapi-endpoint-ip
```

このコマンドの詳細については、「[コマンドリファレンス `put-bucket-lifecycle-configuration`](#)」の「」を参照してください。AWS CLI

Snowball Edge デバイスで S3 バケットを操作する

Snow Family デバイスの Amazon S3 互換ストレージでは、Snowball Edge デバイスで Amazon S3 バケットを作成して、ローカルデータアクセス、ローカルデータ処理、データレジデンシーを必要とするアプリケーション用に、オンプレミスでオブジェクトを保存および取得できます。Snow Family デバイスの Amazon S3 互換ストレージは、SNOW という新しいストレージクラスを提供します。このストレージクラスは、S3 API を使用し、複数の Snowball Edge デバイス間でデータを永続的かつ冗長的に保存するように設計されています。Snowball Edge バケットで、Amazon S3 と同じ API と機能 (バケットのライフサイクルポリシー、暗号化、タグ付けなど) を使用できます。AWS

Command Line Interface (AWS CLI) または SDK を使用して、Snow Family デバイスの Amazon S3 互換ストレージを使用できます。AWS SDKs

Snow Family デバイスバケットの Amazon S3 互換ストレージにアクセスできるかどうかを判断する

以下の例では、head-bucket コマンドを使用して、Amazon S3 バケットが存在し、AWS CLI を使用してそのバケットにアクセスする許可があるかどうかを調べます。このコマンドを使用するには、各ユーザー入力プレースホルダーを独自の情報に置き換えます。

```
aws s3api head-bucket --bucket sample-bucket --profile your-profile --endpoint-url https://s3api-endpoint-ip
```

バケットまたはリージョンバケットのリストを取得する

list-regional-buckets または list buckets を使用して、を使用して Snow Family デバイスバケットの Amazon S3 互換ストレージを一覧表示します AWS CLI。

```
aws s3control list-regional-buckets --account-id 123456789012 --profile your-profile --endpoint-url https://s3ctrlapi-endpoint-ip
```

list-regional-buckets コマンドの詳細については、「[コマンドリファレンスlist-regional-buckets](#)」の「」を参照してください。AWS CLI

```
aws s3 list-buckets --account-id 123456789012 --endpoint-url https://s3api-endpoint-ip
```

list-buckets コマンドの詳細については、[コマンドAWS CLI リファレンスの「list-buckets」](#)を参照してください。

次の SDK for Java の例では、Snowball Edge デバイスのバケットのリストを取得します。詳細については、Amazon Simple Storage Service API リファレンス[ListBuckets](#)の「」を参照してください。

```
import com.amazonaws.services.s3.model.*;
public void listBuckets() {
    ListBucketsRequest reqListBuckets = new ListBucketsRequest()
        .withAccountId(AccountId)
    ListBucketsResult respListBuckets = s3APIClient.RegionalBuckets(reqListBuckets);
    System.out.printf("ListBuckets Response: %s\n", respListBuckets.toString());
}
```

次の PowerShell 例では、Snowball Edge デバイスのバケットのリストを取得します。

```
Get-S3CRegionalBucketList -AccountId 012345678910 -Endpoint "https://snowball_ip" -  
Region snow
```

次の .NET の例では、Snowball Edge デバイスのバケットのリストを取得します。

```
using Amazon.S3Control;  
using Amazon.S3Control.Model;  
  
namespace SnowTest;  
  
internal class Program  
{  
    static async Task Main(string[] args)  
    {  
        var config = new AmazonS3ControlConfig  
        {  
            ServiceURL = "https://snowball_ip",  
            AuthenticationRegion = "snow" // Note that this is not RegionEndpoint  
        };  
  
        var client = new AmazonS3ControlClient(config);  
  
        var response = await client.ListRegionalBucketsAsync(new  
ListRegionalBucketsRequest()  
        {  
            AccountId = "012345678910"  
        });  
    }  
}
```

バケットを取得する

以下の例では、AWS CLIを使用して Snow Family デバイスバケットの Amazon S3 互換ストレージを取得します。このコマンドを使用するには、各ユーザー入力プレースホルダーを独自の情報に置き換えます。

```
aws s3control get-bucket --account-id 123456789012 --bucket DOC-EXAMPLE-BUCKET --  
profile your-profile --endpoint-url https://s3ctrlapi-endpoint-ip
```

このコマンドの詳細については、「AWS CLI コマンドリファレンス」の「[get-bucket](#)」を参照してください。

以下の Snow Family デバイスの Amazon S3 互換ストレージの例では、SDK for Java を使用してバケットを取得します。詳細については、「[Amazon Simple Storage Service API リファレンスGetBucket](#)」の「」を参照してください。

```
import com.amazonaws.services.s3control.model.*;  
  
public void getBucket(String bucketName) {  
  
    GetBucketRequest reqGetBucket = new GetBucketRequest()  
        .withBucket(bucketName)  
        .withAccountId(AccountId);  
  
    GetBucketResult respGetBucket = s3ControlClient.getBucket(reqGetBucket);  
    System.out.printf("GetBucket Response: %s%n", respGetBucket.toString());  
}
```

バケットを削除する

Important

- バケット AWS アカウント を作成する がバケットを所有しており、バケットを削除できる唯一のバケットです。
- Snow Family デバイスバケットは、削除する前に空にする必要があります。
- バケットが削除された後は、そのバケットを復元することはできません。

以下の例では、AWS CLIを使用して Snow Family デバイスバケットの Amazon S3 互換ストレージを削除します。このコマンドを使用するには、各ユーザー入力プレースホルダーを独自の情報に置き換えます。

```
aws s3control delete-bucket --account-id 123456789012 --bucket DOC-EXAMPLE-BUCKET --
profile your-profile --endpoint-url https://s3ctrlapi-endpoint-ip
```

このコマンドの詳細については、「AWS CLI コマンドリファレンス」の「[delete-bucket](#)」を参照してください。

Snowball Edge デバイスで S3 オブジェクトを操作する

このセクションでは、Snow Family デバイスの Amazon S3 互換ストレージ上のオブジェクトに対して実行できるさまざまな操作について説明します。

Snow Family デバイスバケットの Amazon S3 互換ストレージにオブジェクトをコピーする

次の例では、*sample-object.xml* という名前のファイルを、AWS CLIを使用するための書き込み許可がある Snow Family デバイスバケットの Amazon S3 互換ストレージにアップロードします。このコマンドを使用するには、各ユーザー入力プレースホルダーを独自の情報に置き換えます。

```
aws s3api put-object --bucket sample-bucket --key sample-object.xml --body sample-
object.xml --profile your-profile --endpoint-url s3api-endpoint-ip
```

以下の Snow Family デバイスの Amazon S3 互換ストレージの例では、SDK for Java を使用して同じバケット内の新しいオブジェクトにオブジェクトをコピーします。このコマンドを使用するには、各ユーザー入力プレースホルダーを独自の情報に置き換えます。

```
import com.amazonaws.AmazonServiceException;
import com.amazonaws.SdkClientException;
import com.amazonaws.services.s3.AmazonS3;
import com.amazonaws.services.s3.AmazonS3ClientBuilder;
import com.amazonaws.services.s3.model.CopyObjectRequest;
add : import java.io.IOException;

public class CopyObject {
    public static void main(String[] args) {
        String bucketName = "*** Bucket name ***";
        String sourceKey = "*** Source object key ***";
        String destinationKey = "*** Destination object key ***";

        try {
            // This code expects that you have AWS credentials set up per:
```

```
// https://docs.aws.amazon.com/sdk-for-java/v1/developer-guide/setup-credentials.html
AmazonS3 s3Client = AmazonS3ClientBuilder.standard()
    .enableUseArnRegion()
    .build();

// Copy the object into a new object in the same bucket.
CopyObjectRequest copyObjectRequest = new CopyObjectRequest(sourceKey,
destinationKey);
s3Client.copyObject(copyObjectRequest);
CopyObjectRequest copyObjectRequest = CopyObjectRequest.builder()
    .sourceKey(sourceKey)
    .destinationKey(destKey)
    .build();
} catch (AmazonServiceException e) {
    // The call was transmitted successfully, but Amazon S3 couldn't process
    // it, so it returned an error response.
    e.printStackTrace();
} catch (SdkClientException e) {
    // Amazon S3 couldn't be contacted for a response, or the client
    // couldn't parse the response from Amazon S3.
    e.printStackTrace();
}
}
```

バケットからオブジェクトを取得する

以下の例では、AWS CLIを使用して Snow Family デバイスバケットの Amazon S3 互換ストレージから *sample-object.xml* というオブジェクトを取得します。SDK コマンドは `s3-snow:GetObject` です。このコマンドを使用するには、各ユーザー入力プレースホルダーを独自の情報に置き換えます。

```
aws s3api get-object --bucket sample-bucket --key sample-object.xml --profile your-profile --endpoint-url s3api-endpoint-ip
```

このコマンドの詳細については、「AWS CLI コマンドリファレンス」の「[get-object](#)」を参照してください。

以下の Snow Family デバイスの Amazon S3 互換ストレージの例では、SDK for Java を使用してオブジェクトを取得します。このコマンドを使用するには、各ユーザー入力プレースホルダーを

独自の情報に置き換えます。詳細については、[「Amazon Simple Storage Service API リファレンスGetObject」](#)の「」を参照してください。

```
import com.amazonaws.AmazonServiceException;
import com.amazonaws.SdkClientException;
import com.amazonaws.services.s3.AmazonS3;
import com.amazonaws.services.s3.AmazonS3ClientBuilder;
import com.amazonaws.services.s3.model.GetObjectRequest;
import com.amazonaws.services.s3.model.ResponseHeaderOverrides;
import com.amazonaws.services.s3.model.S3Object;

import java.io.BufferedReader;
import java.io.IOException;
import java.io.InputStream;
import java.io.InputStreamReader;

public class GetObject {
    public static void main(String[] args) throws IOException {
        String bucketName = "**** Bucket name ****";
        String key = "**** Object key ****";

        S3Object fullObject = null, objectPortion = null, headerOverrideObject = null;
        try {
            // This code expects that you have AWS credentials set up per:
            // https://docs.aws.amazon.com/sdk-for-java/v1/developer-guide/setup-
credentials.html
            AmazonS3 s3Client = AmazonS3ClientBuilder.standard()
                .enableUseArnRegion()
                .build();
            GetObjectRequest getObjectRequest = GetObjectRequest.builder()
                .bucket(bucketName)
                .key(key)
                .build();

s3Client.getObject(getObjectRequest);
        } catch (AmazonServiceException e) {
            // The call was transmitted successfully, but Amazon S3 couldn't process
            // it, so it returned an error response.
            e.printStackTrace();
        } catch (SdkClientException e) {
            // Amazon S3 couldn't be contacted for a response, or the client
            // couldn't parse the response from Amazon S3.

```

```
        e.printStackTrace();
    } finally {
        // To ensure that the network connection doesn't remain open, close any
open input streams.
        if (fullObject != null) {
            fullObject.close();
        }
        if (objectPortion != null) {
            objectPortion.close();
        }
        if (headerOverrideObject != null) {
            headerOverrideObject.close();
        }
    }
}

private static void displayTextInputStream(InputStream input) throws IOException {
    // Read the text input stream one line at a time and display each line.
    BufferedReader reader = new BufferedReader(new InputStreamReader(input));
    String line = null;
    while ((line = reader.readLine()) != null) {
        System.out.println(line);
    }
    System.out.println();
}
}
```

バケット内のオブジェクトを一覧表示する

以下の例では、AWS CLIを使用して、Snow Family デバイスバケットの Amazon S3 互換ストレージ内のオブジェクトを一覧表示します。SDK コマンドは `s3-snow:ListObjectsV2` です。このコマンドを使用するには、各ユーザー入力プレースホルダーを独自の情報に置き換えます。

```
aws s3api list-objects-v2 --bucket sample-bucket --profile your-profile --endpoint-  
url s3api-endpoint-ip
```

このコマンドの詳細については、「[コマンドリファレンス](#)」の [list-objects-v 「2」](#) を参照してください。AWS CLI

以下の Snow Family デバイスの Amazon S3 互換ストレージの例では、SDK for Java を使用してバケット内のオブジェクトを一覧表示します。このコマンドを使用するには、各ユーザー入力スペースホルダーを独自の情報に置き換えます。

この例では、ListObjects API オペレーションの最新リビジョンである [ListObjectsV2](#) を使用しています。この改訂版の API オペレーションをアプリケーション開発に使用することをお勧めします。下位互換性のために、Amazon S3 はこの API の以前のバージョンであるオペレーションを引き続きサポートします。

```
import com.amazonaws.AmazonServiceException;
import com.amazonaws.SdkClientException;
import com.amazonaws.services.s3.AmazonS3;
import com.amazonaws.services.s3.AmazonS3ClientBuilder;
import com.amazonaws.services.s3.model.ListObjectsV2Request;
import com.amazonaws.services.s3.model.ListObjectsV2Result;
import com.amazonaws.services.s3.model.S3ObjectSummary;

public class ListObjectsV2 {

    public static void main(String[] args) {
        String bucketName = "*** Bucket name ***";

        try {
            // This code expects that you have AWS credentials set up per:
            // https://docs.aws.amazon.com/sdk-for-java/v1/developer-guide/setup-credentials.html
            AmazonS3 s3Client = AmazonS3ClientBuilder.standard()
                .enableUseArnRegion()
                .build();

            System.out.println("Listing objects");

            // maxKeys is set to 2 to demonstrate the use of
            // ListObjectsV2Result.getNextContinuationToken()
            ListObjectsV2Request req = new
ListObjectsV2Request().withBucketName(bucketName).withMaxKeys(2);
            ListObjectsV2Result result;

            do {
                result = s3Client.listObjectsV2(req);

                for (S3ObjectSummary objectSummary : result.getObjectSummaries()) {
```

```
        System.out.printf(" - %s (size: %d)\n", objectSummary.getKey(),
objectSummary.getSize());
    }
    // If there are more than maxKeys keys in the bucket, get a
continuation token
    // and list the next objects.
    String token = result.getNextContinuationToken();
    System.out.println("Next Continuation Token: " + token);
    req.setContinuationToken(token);
    } while (result.isTruncated());
} catch (AmazonServiceException e) {
    // The call was transmitted successfully, but Amazon S3 couldn't process
// it, so it returned an error response.
    e.printStackTrace();
} catch (SdkClientException e) {
    // Amazon S3 couldn't be contacted for a response, or the client
// couldn't parse the response from Amazon S3.
    e.printStackTrace();
}
}
}
```

バケット内のオブジェクトを削除する

Snow Family デバイスバケットの Amazon S3 互換ストレージから 1 つまたは複数のオブジェクトを削除できます。以下の例では、AWS CLIを使用して *sample-object.xml* という名前のオブジェクトを削除します。このコマンドを使用するには、各ユーザー入力プレースホルダーを独自の情報に置き換えます。

```
aws s3api delete-object --bucket sample-bucket --key key --profile your-profile --
endpoint-url s3api-endpoint-ip
```

このコマンドの詳細については、「AWS CLI コマンドリファレンス」の「[delete-object](#)」を参照してください。

以下の Snow Family デバイスの Amazon S3 互換ストレージの例では、SDK for Java を使用してバケット内のオブジェクトを削除します。この例を使用するには、削除するオブジェクトのキー名を指定します。詳細については、Amazon Simple Storage Service API リファレンス [DeleteObject](#) の「」を参照してください。

```
import com.amazonaws.AmazonServiceException;
import com.amazonaws.SdkClientException;
import com.amazonaws.services.s3.AmazonS3;
import com.amazonaws.services.s3.AmazonS3ClientBuilder;
import com.amazonaws.services.s3.model.DeleteObjectRequest;

public class DeleteObject {
    public static void main(String[] args) {
        String bucketName = "*** Bucket name ***";
        String keyName = "*** key name *****";

        try {
            // This code expects that you have AWS credentials set up per:
            // https://docs.aws.amazon.com/sdk-for-java/v1/developer-guide/setup-
credentials.html
            AmazonS3 s3Client = AmazonS3ClientBuilder.standard()
                .enableUseArnRegion()
                .build();

            DeleteObjectRequest deleteObjectRequest = DeleteObjectRequest.builder()
                .bucket(bucketName)
                .key(keyName)
                .build());
            s3Client.deleteObject(deleteObjectRequest);
        } catch (AmazonServiceException e) {
            // The call was transmitted successfully, but Amazon S3 couldn't process
            // it, so it returned an error response.
            e.printStackTrace();
        } catch (SdkClientException e) {
            // Amazon S3 couldn't be contacted for a response, or the client
            // couldn't parse the response from Amazon S3.
            e.printStackTrace();
        }
    }
}
```

Snow Family デバイスの Amazon S3 互換ストレージでサポートされる REST API アクション

以下のリストは、Snow Family デバイスの Amazon S3 互換ストレージでサポートされている API オペレーションと、AWS リージョンの Amazon S3 の関連オペレーションへのリンクを示しています。

サポートされているバケット API オペレーション

- [CreateBucket](#)
- [DeleteBucket](#)
- [DeleteBucketLifecycle](#)
- [GetBucket](#)
- [GetBucketLifecycleConfiguration](#)
- [ListBuckets](#)
- [PutBucketLifecycleConfiguration](#)

サポートされているオブジェクト API オペレーション

- [AbortMultipartUpload](#)
- [CompleteMultipartUpload](#)
- [CopyObject](#)
- [CreateMultipartUpload](#)
- [DeleteObject](#)
- [DeleteObjects](#)
- [DeleteObjectTagging](#)
- [GetObject](#)
- [GetObjectTagging](#)
- [HeadBucket](#)
- [HeadObject](#)
- [ListMultipartUploads](#)
- [ListObjects](#)
- [ListObjectsV2](#)

- [ListParts](#)
- [PutObject](#)
- [PutObjectTagging](#)
- [UploadPart](#)
- [UploadPartCopy](#)

Snow Family デバイスの Amazon S3 互換ストレージを Snow デバイスのクラスターで使用する

クラスターは、ローカルストレージとコンピューティングの目的で単一の論理単位として使用される 3 つ以上の Snowball Edge デバイスのコレクションです。ローカルストレージおよびコンピューティングにおいて、クラスターには、スタンドアロンの Snowball Edge デバイスに比べ主な利点が 2 つあります。

- 耐久性の向上 – Snowball Edge デバイスのクラスターに保存されている S3 データは、単一のデバイスよりも耐久性が高くなります。さらに、クラスターに影響を与える可能性のあるハードウェアの停止にもかかわらず、クラスター上のデータは安全で実行可能なままです。クラスターは、データが危険にさらされる前に、3 台と 4 台のデバイスのクラスターで 1 台のデバイスの損失に、5 ~ 16 台のデバイスのクラスターで最大 2 台のデバイスの損失に耐えられます。異常なノードを置き換えて、クラスターに保存されているデータの耐久性と安全性を維持できます。
- ストレージの増加 – Snowball Edge ストレージ最適化デバイスを使用すると、最大 2.6 PB の使用可能な S3-compatible ストレージ容量を持つ 16 個のノードクラスターを作成できます。Snowball Edge コンピューティング最適化デバイスを使用すると、最大 501 TB の使用可能な S3-compatible ストレージ容量を持つ 16 個のノードクラスターを作成できます。

Snowball Edge デバイスのクラスターは、リーダーレスノードで構成されています。どのノードでもクラスター全体にデータを書き込んだり、クラスター全体からデータを読み取ったりでき、すべてのノードでクラスター behind-the-scenes の管理を実行できます。

Snowball Edge デバイスのクラスターの使用を計画する際は、次の考慮事項に注意してください。

- クラスターの潜在的なパフォーマンスと安定性の問題を減らすために、クラスター内のすべてのデバイスに冗長電源を提供することをお勧めします。
- スタンドアロンのローカルストレージおよびコンピューティングのジョブの場合と同様に、クラスターに保存されているデータを Amazon S3 にインポートするには、個々のインポートジョブの一

部として追加のデバイスを注文する必要があります。インポートジョブとして追加のデバイスを注文すると、クラスターからインポートジョブデバイスにデータを転送できます。

- Amazon S3 からクラスターにデータを取得するには、Amazon S3 API を使用してクラスターに Amazon S3 バケットを作成し、S3 からオブジェクトを保存および取得します。また、AWS DataSync を使用して、Snowball Edge デバイスの Snow Family デバイスの AWS ストレージサービスと Amazon S3 互換ストレージ間でオブジェクトを転送することもできます。詳細については、「[Snowball Edge での S3 互換ストレージを使用した転送の設定](#)」を参照してください。
- 、AWS Snow ファミリーマネジメントコンソール、AWS CLI または AWS SDKs のいずれかからデバイスのクラスターを注文するジョブを作成できます。詳細については、「[開始方法](#)」を参照してください。
- クラスター内の各デバイスにはノード ID があります。ノード ID は、スタンドアロンデバイスのジョブ ID など、クラスター内の各デバイスの一意の識別子です。ノード IDs は、AWS Snow ファミリーマネジメントコンソール、SDK AWS CLI、および Snowball Edge クライアントから取得できます。AWS SDKs Snowball Edge クライアントコマンド describe-device および IDs describe-cluster を返します。
- クラスターの寿命は、クラスターがプロビジョンされる際にクラスターデバイスに付与されるセキュリティ証明書によって制限されています。Snowball Edge デバイスは、デフォルトで最大 360 日間使用できますが、その後は返却していただく必要があります。この期間が終了すると、デバイスは、読み込み/書き込みリクエストへの応答を停止します。1 つ以上のデバイスを 360 日以上保持する必要がある場合は、[お問い合わせ](#) してください AWS Support。
- がクラスターの一部である返されたデバイス AWS を受信すると、デバイスの完全な消去が実行されます。この消去作業は National Institute of Standards and Technology (NIST) 800-88 基準に準拠しています。

Snow Family デバイスクラスターの Amazon S3 互換ストレージの耐障害性とストレージ容量

クラスターサイズ	耐障害性	Snowball Edge Compute Optimized (AMD EPYC Gen1、HDD、オプションの GPU 搭載) デバイスのストレージ容量 (TB 単位)	Snowball Edge Compute Optimized (AMD EPYC Gen2 および NVMe で最適化されたコンピューティング) デバイスのストレージ容量 (TB 単位)	Snowball Edge ストレージ最適化 210 TB デバイスのストレージ容量 (TB 単位)
3	最大 1 ノードの損失	83	38	438
4	最大 1 ノードの損失	125	57	657
5	最大 2 ノードの損失	125	57	657
6	最大 2 ノードの損失	167	76	904
7	最大 2 ノードの損失	209	95	1096
8	最大 2 ノードの損失	250	114	1315
9	最大 2 ノードの損失	292	133	1534
10	最大 2 ノードの損失	334	152	1754
11	最大 2 ノードの損失	370	165	1970

クラスターサイズ	耐障害性	Snowball Edge Compute Optimized (AMD EPYC Gen1、HDD、オプションの GPU 搭載) デバイスのストレージ容量 (TB 単位)	Snowball Edge Compute Optimized (AMD EPYC Gen2 および NVMe で最適化されたコンピュティング) デバイスのストレージ容量 (TB 単位)	Snowball Edge ストレージ最適化 210 TB デバイスのストレージ容量 (TB 単位)
12	最大 2 ノードの損失	376	171	1973
13	最大 2 ノードの損失	418	190	2192
14	最大 2 ノードの損失	459	209	2411
15	最大 2 ノードの損失	495	225	2625
16	最大 2 ノードの損失	501	228	2631

クラスターのロックを解除すると、そのクラスターにデータを保存してアクセスできるようになります。Amazon S3 互換エンドポイントを使用して、クラスターとの間でデータを読み書きできます。

クラスターにデータを読み書きするには、デバイスのクラスターに、使用できないノードの許容数を超えない読み取り/書き込みクォーラムがある必要があります。

Snowball Edge クラスターのクォーラム

クォーラムとは、読み取り/書き込みのクォーラムを維持するために相互に通信する必要がある、クラスター内の Snowball Edge デバイスの最小数を表します。

クラスター内のすべてのデバイスが正常であれば、クラスターの読み取り/書き込みクォーラムがあります。これらのデバイスの 1 つまたは 2 つがオフラインになった場合は、クラスターの運用キャ

パシティーを減らします。ただし、クラスターの読み書きは引き続き行うことができます。クラスターが動作している 1 つまたは 2 つのデバイスを除くすべてのデバイスでは、クラスターには読み取り/書き込みクォーラムが残ります。クラスターの運用能力に影響が出る前にオフラインになることのできるノードの数は、[this table](#) で確認できます。

クラスターが示されているデバイス数を超えて失われると、クォーラムが失われる可能性があります[this table](#)。クォーラムが失われると、クラスターはオフラインになり、クラスター内のデータは使用できなくなります。イベントの重要度によっては、この問題を解決できる場合があります。解決できない場合、データは完全に失われます。一時的な外部電源イベントであり、Snowball Edge デバイスに電源を再びオンにして、クラスター内のすべてのノードをロック解除できる場合、データは再び利用できます。

Important

正常なノードが最小数存在しない場合は、AWS Supportにお問い合わせください。

`describe-cluster` コマンドを使用して、各ノードのロック状態とネットワーク到達可能性を表示できます。クラスター内のデバイスが正常で接続されていることを確認することは、クラスターストレージを使用する際にユーザーが引き受ける管理責任です。詳細については、「[デバイスステータスの取得](#)」を参照してください。

1 つ以上のノードが異常であると判断した場合は、クラスター内のノードを置き換えて、クォーラムとデータの正常性と安定性を維持できます。詳細については、「[クラスター内のノードの置き換え](#)」を参照してください。

使用できないクラスターノードを再接続する

電源やネットワークの切断などの問題により、ノード、またはクラスター内のデバイスが一時的に使用できなくなることがありますが、ノード上のデータは損なわれません。これが発生すると、クラスターのステータスに影響を及ぼします。ノードのネットワーク到達可能性とロックステータスは、`snowballEdge describe-cluster` コマンドを使用して Snowball Edge クライアントに報告されます。

物理的にクラスターを配置しているため、すべてのノードの前面、背面、上部を利用することをお勧めします。これにより、背面で電源ケーブルやネットワークケーブルに、上部の配送ラベルでノード ID に、前面の LCD 画面で IP アドレスやその他の管理情報にアクセスすることができます。

ノードが使用不可と検出された場合は、使用不可の原因となったシナリオに応じて、次のいずれかの手順を試してみることをお勧めします。

使用不可のノードを再接続するには

1. ノードの電源が入っていることを確認します。
2. ノードが、残りのクラスターと同じ内部ネットワークに接続されていることを確認します。
3. ノードの電源を入れる必要がある場合は、終了するまで最大 20 分待ちます。
4. `snowballEdge unlock-cluster` コマンドまたは `snowballEdge associate-device` コマンドを実行します。例については、「[Snowball Edge デバイスのロック解除](#)」を参照してください。

ネットワーク接続は失ったが電源を失っていない使用不可のノードを再接続するには

1. ノードが、残りのクラスターと同じ内部ネットワークに接続されていることを確認します。
2. `snowballEdge describe-device` コマンドを実行して、以前使用可能だったノードがクラスターに再度追加された日時を確認します。例については、「[デバイスステータスの取得](#)」を参照してください。

前の手順を実行すると、ノードが正常に動作するはずですが、読み取り/書き込みクォーラムも満たす必要があります。このケースに該当しない場合は、1 つ以上のノードに深刻な問題が存在している可能性があるため、クラスターからの削除が必要な場合があります。

クラスター内のノードの置き換え

ノードを置き換えるには、まず置き換えを注文する必要があります。代替ノードは、コンソール、AWS CLI、または SDKs のいずれか AWS から注文できます。コンソールから代替ノードを注文する場合は、キャンセルされていない、または完了していないジョブの代替品を注文できます。次に、クラスターから異常なノードを関連付け解除し、置換ノードをネットワークに接続して、置換ノードを含むクラスターをロック解除し、置換ノードをクラスターに関連付けて、Snow Family デバイスサービスで Amazon S3 互換ストレージを再起動します。

コンソールから代替ノードを注文するには

1. [AWS Snow ファミリーマネジメントコンソール](#) にサインインします。
2. ジョブダッシュボードから作成したクラスターに属するノードのジョブを検索して、選択します。
3. [Actions] で、[Replace node] を選択します。

この操作を行うと、ジョブ作成ウィザードの最終ステップが開きます。設定は、元のクラスター作成時とすべて同じです。

4. [ジョブの作成] を選択します。

Snowball Edge の代替品を配送中です。クラスターから異常なノードを削除するには、次の手順に従います。

クラスターからノードを削除するには

1. 削除するノードの電源をオフにします。詳細については、[「Snowball Edge の電源をオフにする」](#)を参照してください。
2. `describe-cluster` コマンドを使用して、異常なノードに到達できないようにします。これは、`NetworkReachability` オブジェクト `State` の名前 `UNREACHABLE` の値で示されます。

```
snowballEdge describe-cluster --manifest-file path/to/manifest/file.bin --unlock-code unlock-code --endpoint https://ip-address-of-device-in-cluster
```

Example `describe-cluster` の出力

```
{
  "ClusterId": "CID12345678-1234-1234-1234-123456789012",
  "Devices": [
    {
      "DeviceId": "JID12345678-1234-1234-1234-123456789012",
      "UnlockStatus": {
        "State": "UNLOCKED"
      },
      "ActiveNetworkInterface": {
        "IpAddress": "10.0.0.0"
      },
      "ClusterAssociation": {
        "ClusterId": "CID12345678-1234-1234-1234-123456789012",
        "State": "ASSOCIATED"
      },
      "NetworkReachability": {
        "State": "REACHABLE"
      }
    }
  ]
}
```

```
    },
    "Tags": []
  },
  {
    "DeviceId": "JID12345678-1234-1234-1234-123456789013",
    "UnlockStatus": {
      "State": "UNLOCKED"
    },
    "ActiveNetworkInterface": {
      "IpAddress": "10.0.0.1"
    },
    "ClusterAssociation": {
      "ClusterId": "CID12345678-1234-1234-1234-123456789012",
      "State": "ASSOCIATED"
    },
    "NetworkReachability": {
      "State": "REACHABLE"
    },
    "Tags": []
  },
  {
    "DeviceId": "JID12345678-1234-1234-1234-123456789014",
    "ClusterAssociation": {
      "ClusterId": "CID12345678-1234-1234-1234-123456789012",
      "State": "ASSOCIATED"
    },
    "NetworkReachability": {
      "State": "UNREACHABLE"
    }
  }
]
}
```

3. `describe-service` コマンドを使用して、`s3-snow`サービスのステータスが `DEGRADED` であることを確認します。

```
snowballEdge describe-service --service-id s3-snow --device-ip-addresses snow-device-1-address snow-device-2-address --manifest-file path/to/manifest/file.bin --unlock-code unlock-code --endpoint https://snow-device-ip-address
```

Example **describe-service** コマンドの出力の

```
{
  "ServiceId": "s3-snow",
  "Autostart": true,
  "Status": {
    "State": "DEGRADED"
  },
  "ServiceCapacities": [
    {
      "Name": "S3 Storage",
      "Unit": "Byte",
      "Used": 38768180432,
      "Available": 82961231819568
    }
  ],
  "Endpoints": [
    {
      "Protocol": "https",
      "Port": 443,
      "Host": "10.0.0.10",
      "CertificateAssociation": {
        "CertificateArn": "arn:aws:snowball-
device:::certificate/7Rg2lP9tQaHnW4sC6xUzF1vGyD3jB5kN8MwEiYpT"
      },
      "Description": "s3-snow bucket API endpoint",
      "DeviceId": "JID-beta-207012320001-24-02-05-17-17-26",
      "Status": {
        "State": "ACTIVE"
      }
    },
    {
      "Protocol": "https",
      "Port": 443,
      "Host": "10.0.0.11",
      "CertificateAssociation": {
        "CertificateArn": "arn:aws:snowball-
device:::certificate/7Rg2lP9tQaHnW4sC6xUzF1vGyD3jB5kN8MwEiYpT"
      },
      "Description": "s3-snow object API endpoint",
      "DeviceId": "JID-beta-207012320001-24-02-05-17-17-26",
      "Status": {
```

```
        "State": "ACTIVE"
      }
    },
    {
      "Protocol": "https",
      "Port": 443,
      "Host": "10.0.0.12",
      "CertificateAssociation": {
        "CertificateArn": "arn:aws:snowball-
device:::certificate/7Rg2lP9tQaHnW4sC6xUzF1vGyD3jB5kN8MwEiYpT"
      },
      "Description": "s3-snow bucket API endpoint",
      "DeviceId": "JID-beta-207012240003-24-02-05-17-17-27",
      "Status": {
        "State": "ACTIVE"
      }
    },
    {
      "Protocol": "https",
      "Port": 443,
      "Host": "10.0.0.13",
      "CertificateAssociation": {
        "CertificateArn": "arn:aws:snowball-
device:::certificate/7Rg2lP9tQaHnW4sC6xUzF1vGyD3jB5kN8MwEiYpT"
      },
      "Description": "s3-snow object API endpoint",
      "DeviceId": "JID-beta-207012320001-24-02-05-17-17-27",
      "Status": {
        "State": "ACTIVE"
      }
    }
  ]
}
```

4. `disassociate-device` コマンドを使用して、異常なノードの関連付けを解除し、クラスターから削除します。

```
snowballEdge disassociate-device --device-id device-id --manifest-file path/to/manifest/file.bin --unlock-code unlock-code --endpoint https://ip-address-of-unhealthy-device
```

Example `disassociate-device` コマンドの出力

Disassociating your Snowball Edge device from the cluster. Your Snowball Edge device will be disassociated from the cluster when it is in the "DISASSOCIATED" state. You can use the `describe-cluster` command to determine the state of your cluster.

5. コマンド `describe-cluster` を再度使用して、異常なノードとクラスターの関連付けが解除されていることを確認します。

```
snowballEdge describe-cluster --manifest-file path/to/manifest/file.bin --unlock-code unlock-code --endpoint https:ip-address-of-healthy-device
```

Example ノードの関連付けが解除されていることを示す `describe-cluster` コマンドの

```
{
  "ClusterId": "CID12345678-1234-1234-1234-123456789012",
  "Devices": [
    {
      "DeviceId": "JID12345678-1234-1234-1234-123456789012",
      "UnlockStatus": {
        "State": "UNLOCKED"
      },
      "ActiveNetworkInterface": {
        "IpAddress": "10.0.0.0"
      },
      "ClusterAssociation": {
        "ClusterId": "CID12345678-1234-1234-1234-123456789012",
        "State": "ASSOCIATED"
      },
      "NetworkReachability": {
        "State": "REACHABLE"
      },
      "Tags": []
    },
  ]
}
```

```
    "DeviceId": "JID12345678-1234-1234-1234-123456789013",
    "UnlockStatus": {
      "State": "UNLOCKED"
    },
    "ActiveNetworkInterface": {
      "IpAddress": "10.0.0.1"
    },
    "ClusterAssociation": {
      "ClusterId": "CID12345678-1234-1234-1234-123456789012",
      "State": "ASSOCIATED"
    },
    "NetworkReachability": {
      "State": "REACHABLE"
    },
    "Tags": []
  },
  {
    "DeviceId": "JID12345678-1234-1234-1234-123456789014",
    "ClusterAssociation": {
      "ClusterId": "CID12345678-1234-1234-1234-123456789012",
      "State": "DISASSOCIATED"
    }
  }
]
```

6. 電源をオフにして、異常なデバイスを に戻します AWS。詳細については、[「Snowball Edge の電源をオフにする」](#) および [「Snowball Edge デバイスの返却」](#) を参照してください。

代替デバイスが到着したら、次の手順を使用してクラスターに追加します。

代替デバイスを追加するには

1. すべてのデバイスの前面、背面、上部にアクセスできるように、クラスターの代替デバイスを配置します。
2. ノードの電源を入れ、ノードがクラスターの残りの部分と同じ内部ネットワークに接続されていることを確認します。詳細については、[「ローカルネットワークへの接続」](#) を参照してください。
3. `unlock-cluster` コマンドを使用して、新しいノードの IP アドレスを含めます。


```
snowballEdge unlock-cluster --manifest-file path/to/manifest/file.bin --unlock-code unlock-code --endpoint https://ip-address-of-cluster-device --device-ip-addresses node-1-ip-address node-2-ip-address new-node-ip-address
```

新しいノードの状態は、次のステップでクラスターに関連付けるDEGRADEDまでになります。

4. `associate-device` コマンドを使用して、置換ノードをクラスターに関連付けます。

```
snowballEdge associate-device --device-ip-address new-node-ip-address
```

Example `associate-device` コマンド出力の

Associating your Snowball Edge device with the cluster. Your Snowball Edge device will be associated with the cluster when it is in the ASSOCIATED state. You can use the `describe-device` command to determine the state of your devices.

5. `describe-cluster` コマンドを使用して、新しいノードがクラスターに関連付けられていることを確認します。

```
snowballEdge describe-cluster --manifest-file path/to/manifest/file.bin --unlock-code unlock-code --endpoint https://node-ip-address
```

Example `describe-cluster` コマンド出力の

```
{
  "ClusterId": "CID12345678-1234-1234-1234-123456789012",
  "Devices": [
    {
      "DeviceId": "JID12345678-1234-1234-1234-123456789012",
      "UnlockStatus": {
```

```
        "State": "UNLOCKED"
    },
    "ActiveNetworkInterface": {
        "IpAddress": "10.0.0.0"
    },
    "ClusterAssociation": {
        "ClusterId": "CID12345678-1234-1234-1234-123456789012",
        "State": "ASSOCIATED"
    },
    "NetworkReachability": {
        "State": "REACHABLE"
    },
    "Tags": []
},
{
    "DeviceId": "JID-CID12345678-1234-1234-1234-123456789013",
    "UnlockStatus": {
        "State": "UNLOCKED"
    },
    "ActiveNetworkInterface": {
        "IpAddress": "10.0.0.1"
    },
    "ClusterAssociation": {
        "ClusterId": "CID12345678-1234-1234-1234-123456789012",
        "State": "ASSOCIATED"
    },
    "NetworkReachability": {
        "State": "REACHABLE"
    },
    "Tags": []
},
{
    "DeviceId": "JID-CID12345678-1234-1234-1234-123456789015",
    "UnlockStatus": {
        "State": "UNLOCKED"
    },
    "ActiveNetworkInterface": {
        "IpAddress": "10.0.0.2"
    },
    "ClusterAssociation": {
        "ClusterId": "CID12345678-1234-1234-1234-123456789012",
        "State": "ASSOCIATED"
    },
    "NetworkReachability": {
```

```
        "State": "REACHABLE"
      },
      "Tags": []
    }
  ]
}
```

6. 新しいノードで、2つの仮想ネットワークインターフェイス (VNI)を作成します。詳細については、「[Snow Family デバイスの Amazon S3 互換ストレージサービスを開始する](#)」を参照してください。
7. `stop-service` コマンドを使用して `s3-snow` サービスを停止します。

```
snowballEdge stop-service --service-id s3-snow --device-ip-addresses cluster-device-1-ip-address cluster-device-2-ip-address cluster-device-3-ip-address --manifest-file path/to/manifest/file.bin --unlock-code unlock-code --endpoint https://snow-device-ip-address
```

Example `stop-service` コマンド出力の

Stopping the AWS service on your Snowball Edge. You can determine the status of the AWS service using the `describe-service` command.

8. `start-service` コマンドを使用して、新しいノードをクラスターに追加した後に `s3-snow` サービスを開始します。

```
snowballEdge start-service --service-id s3-snow --device-ip-addresses cluster-device-1-ip-address cluster-device-2-ip-address cluster-device-3-ip-address --virtual-network-interface-arns "device-1-vni-ip-address-a" "device-1-vni-ip-address-b" "device-2-vni-ip-address-a" "device-2-vni-ip-address-b" "device-3-vni-ip-address-a" "device-3-vni-ip-address-b" --manifest-file path/to/manifest/file.bin --unlock-code unlock-code --endpoint https://snow-device-ip-address
```

Example **start-service** コマンド出力の

Starting the AWS service on your Snowball Edge. You can determine the status of the AWS service using the describe-service command.

9. describe-service コマンドを使用して、s3-snow サービスが開始されたことを確認します。

```
snowballEdge describe-service --service-id s3-snow --device-ip-addresses snow-device-1-address snow-device-2-address snow-device-3-address --manifest-file path/to/manifest/file.bin --unlock-code unlock-code --endpoint https://snow-device-ip-address
```

Example **describe-service** コマンド出力の

```
{
  "ServiceId": "s3-snow",
  "Autostart": true,
  "Status": {
    "State": "ACTIVE"
  },
  "ServiceCapacities": [{
    "Name": "S3 Storage",
    "Unit": "Byte",
    "Used": 38768180432,
    "Available": 82961231819568
  }],
  "Endpoints": [{
    "Protocol": "https",
    "Port": 443,
    "Host": "10.0.0.10",
    "CertificateAssociation": {
      "CertificateArn": "arn:aws:snowball-
device::certificate/7Rg2lP9tQaHnW4sC6xUzF1vGyD3jB5kN8MwEiYpT"
    },
    "Description": "s3-snow bucket API endpoint",
    "DeviceId": "JID12345678-1234-1234-1234-123456789012",
    "Status": {
```

```
        "State": "ACTIVE"
    }
}, {
    "Protocol": "https",
    "Port": 443,
    "Host": "10.0.0.11",
    "CertificateAssociation": {
        "CertificateArn": "arn:aws:snowball-
device:::certificate/7Rg2lP9tQaHnW4sC6xUzF1vGyD3jB5kN8MwEiYpT"
    },
    "Description": "s3-snow object API endpoint",
    "DeviceId": "JID12345678-1234-1234-1234-123456789013",
    "Status": {
        "State": "ACTIVE"
    }
}, {
    "Protocol": "https",
    "Port": 443,
    "Host": "10.0.0.12",
    "CertificateAssociation": {
        "CertificateArn": "arn:aws:snowball-
device:::certificate/7Rg2lP9tQaHnW4sC6xUzF1vGyD3jB5kN8MwEiYpT"
    },
    "Description": "s3-snow bucket API endpoint",
    "DeviceId": "JID12345678-1234-1234-1234-123456789015",
    "Status": {
        "State": "ACTIVE"
    }
}, {
    "Protocol": "https",
    "Port": 443,
    "Host": "10.0.0.13",
    "CertificateAssociation": {
        "CertificateArn": "arn:aws:snowball-
device:::certificate/7Rg2lP9tQaHnW4sC6xUzF1vGyD3jB5kN8MwEiYpT"
    },
    "Description": "s3-snow object API endpoint",
    "DeviceId": "JID-beta-207012320001-24-02-05-17-17-27",
    "Status": {
        "State": "ACTIVE"
    }
}, {
    "Protocol": "https",
    "Port": 443,
```

```
    "Host": "10.0.0.14",
    "CertificateAssociation": {
      "CertificateArn": "arn:aws:snowball-
device:::certificate/7Rg2lP9tQaHnW4sC6xUzF1vGyD3jB5kN8MwEiYpT"
    },
    "Description": "s3-snow bucket API endpoint",
    "DeviceId": "JID-beta-207012240003-24-02-05-17-17-28",
    "Status": {
      "State": "ACTIVE"
    }
  }, {
    "Protocol": "https",
    "Port": 443,
    "Host": "10.0.0.15",
    "CertificateAssociation": {
      "CertificateArn": "arn:aws:snowball-
device:::certificate/7Rg2lP9tQaHnW4sC6xUzF1vGyD3jB5kN8MwEiYpT"
    },
    "Description": "s3-snow object API endpoint",
    "DeviceId": "JID-beta-207012320001-24-02-05-17-17-28",
    "Status": {
      "State": "ACTIVE"
    }
  }
}]
}
```

Snow Family デバイスの Amazon S3 互換ストレージでのイベント通知の設定

Snow Family デバイスの Amazon S3 互換ストレージは、Message Queuing Telemetry Transport (MQTT) プロトコルに基づくオブジェクト API コールの Amazon S3 イベント通知をサポートします。

Snow Family デバイスの Amazon S3 互換ストレージを使用して、S3 バケットで特定のイベントが発生したときに通知を受け取ることができます。通知を有効にするには、サービスから発行するイベントを指定する通知設定を追加します。

Snow Family デバイスの Amazon S3 互換ストレージは、以下の通知タイプをサポートしています。

- 新しいオブジェクトがイベントを作成しました。
- オブジェクトの削除イベント
- オブジェクトのタグ付けイベント

Amazon S3 イベント通知を設定する

1. 開始する前に、ネットワークに MQTT インフラストラクチャが必要です。
2. Snowball Edge クライアントで、`snowballEdge configure` コマンドを実行して Snowball Edge デバイスをセットアップします。

プロンプトが表示されたら、次の情報を入力します。

- マニフェストファイルへのパス。
 - デバイスのロック解除コード。
 - デバイスのエンドポイント (`https://10.0.0.1` など)。
3. 以下の `put-notification-configuration` コマンドを実行して、外部ブローカーに通知を送信します。

```
snowballEdge put-notification-configuration --broker-endpoint ssl://mqtt-broker-ip-address:8883 --enabled true --service-id s3-snow --ca-certificate file:path-to-mqtt-broker-ca-cert
```

4. 以下の `get-notification-configuration` コマンドを実行して、すべてが正しく設定されていることを確認します。

```
snowballEdge get-notification-configuration --service-id s3-snow
```

これにより、ブローカーのエンドポイントと `enabled` フィールドが返されます。

ネットワーク内の MQTT ブローカーに通知を送信するようにクラスター全体を設定すると、すべてのオブジェクト API コールでイベント通知が生成されます。

Note

トピック `s3SnowEvents/Device ID` (#####の場合はクラスター ID)/`bucketName` をサブスクライブする必要があります。ワイルドカードを使用することもできます。例えば、トピック名は `#` または `s3SnowEvents/#` です。

以下は、Snow Family デバイスの Amazon S3 互換ストレージのイベントログの例です。

```
{
  "eventDetails": {
    "additionalEventData": {
      "AuthenticationMethod": "AuthHeader",
      "CipherSuite": "ECDHE-RSA-AES128-GCM-SHA256",
      "SignatureVersion": "SigV4",
      "bytesTransferredIn": 1205,
      "bytesTransferredOut": 0,
      "x-amz-id-2": "uLdTfvdGTK1X6TBgCZtDd9Beef8wzUurA+Wpht7rKtfdaNsnxeLILg=="
    },
    "eventName": "PutObject",
    "eventTime": "2023-01-30T14:13:24.772Z",
    "requestAuthLatencyMillis": 40,
    "requestBandwidthKBs": 35,
    "requestID": "140CD93455CB62B4",
    "requestLatencyMillis": 77,
    "requestLockLatencyNanos": 1169953,
    "requestParameters": {
      "Content-Length": "1205",
      "Content-MD5": "GZdTU0hYHvHgQgmaw2gl4w==",
      "Host": "10.0.2.251",
      "bucketName": "buckett",
      "key": "file-key"
    },
    "requestTTFBLatencyMillis": 77,
    "responseElements": {
      "ETag": "\"19975350e8581ef1e042099ac36825e3\"",
      "Server": "AmazonS3",
      "x-amz-id-2": "uLdTfvdGTK1X6TBgCZtDd9Beef8wzUurA+Wpht7rKtfdaNsnxeLILg==",
      "x-amz-request-id": "140CD93455CB62B4"
    },
    "responseStatusCode": 200,
  }
}
```



```
    "sourceIPAddress": "172.31.37.21",
    "userAgent": "aws-cli/1.27.23 Python/3.7.16 Linux/4.14.301-224.520.amzn2.x86_64
botocore/1.29.23",
    "userIdentity": {
        "identityType": "IAMUser",
        "principalId": "531520547609",
        "arn": "arn:aws:iam::531520547609:root",
        "userName": "root"
    }
}
}
```

Amazon S3 イベント通知の詳細については、「[Amazon S3 イベント通知](#)」を参照してください。

ローカル SMTP 通知の設定

Simple Mail Transfer Protocol (SMTP) を使用して、Snowball Edge デバイスのローカル通知を設定できます。サービスの状態 (アクティブ、低下、非アクティブ) が変化した場合や、80%、90%、100% の容量使用率のしきい値を超えた場合、ローカル通知は設定済みのサーバーに E メールを送信します。

開始する前に、以下を確認してください。

- 最新の Snowball Edge クライアントにアクセスできる。
- デバイスのロックが解除され、使用可能になっている。
- デバイスはインターネット (Amazon Simple Email Service または外部 SMTP サーバーを使用している場合) またはローカル SMTP サーバーに接続できる。

デバイスの設定

E メール通知を送信するようにデバイスを設定します。

デバイスを SMTP 通知用に設定するには

1. 以下のコマンドを実行して SMTP 設定をデバイスに追加します。

```
# If you don't specify a port, port 587 is the default.
SMTP_ENDPOINT=your-local-smtp-server-endpoint:port

# For multiple email recipients, separate with commas
```

```
RECIPIENTS_LIST=your-email-address

snowballEdge put-notification-configuration \
  --service-id local-monitoring \
  --enabled true \
  --type smtp \
  --broker-endpoint "$SMTP_ENDPOINT" \
  --sender example-sender@domain.com \
  --recipients "$RECIPIENTS_LIST"
```

成功すると、`example-sender@domain.com` からテストメールが届きます。

- 以下の `get-notification-configuration` コマンドを使用して設定をテストします。

```
snowballEdge get-notification-configuration \
  --service-id local-monitoring
```

パスワードや証明書は、ユーザーが指定しても応答には含まれません。

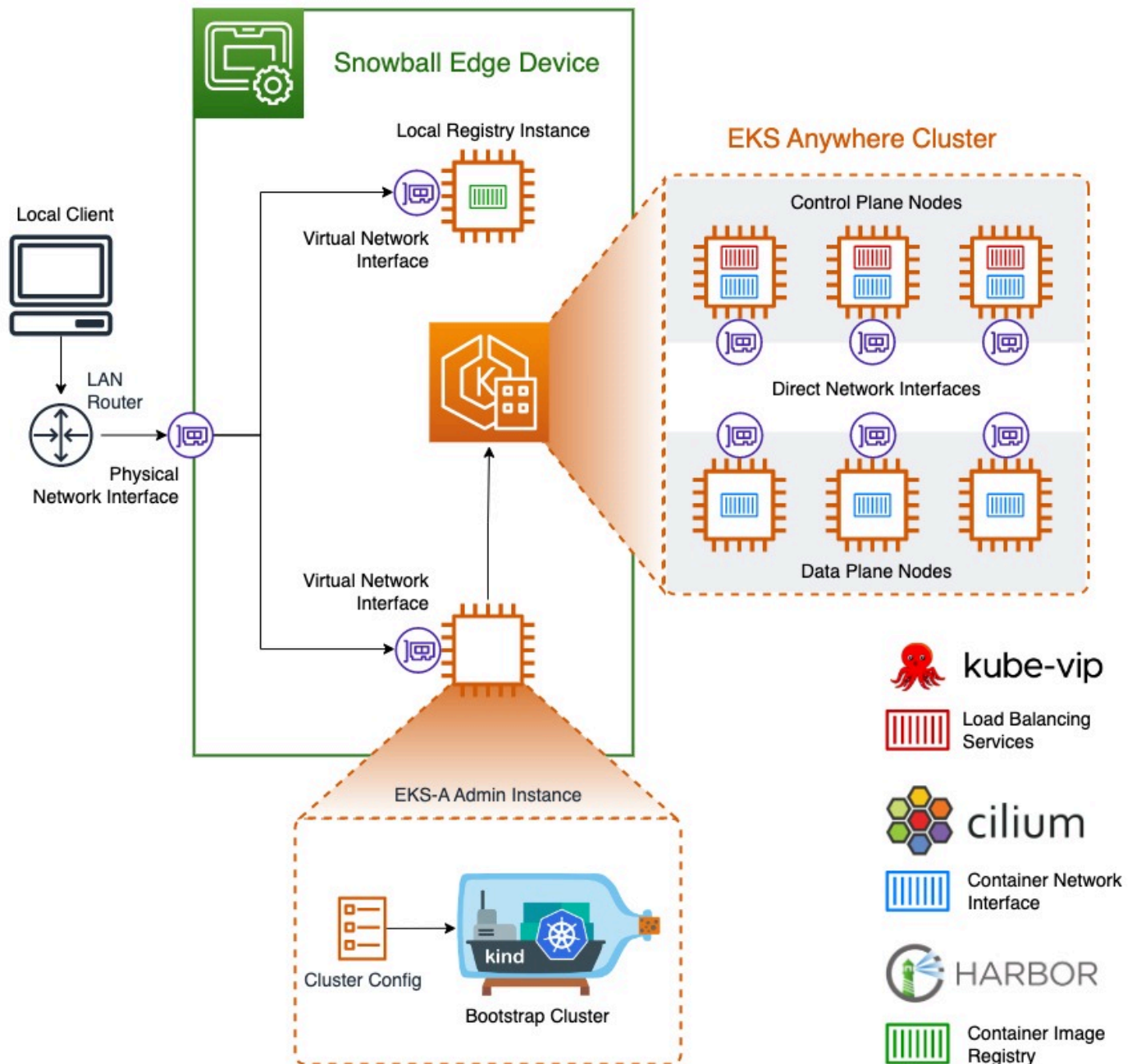
AWS Snow での Amazon EKS Anywhere の使用

Amazon EKS Anywhere on AWS Snow は、Snow Family デバイスで Kubernetes クラスターを作成および運用するのに役立ちます。Kubernetes は、コンテナ化されたアプリケーションのデプロイ、スケーリング、および管理の自動化に使用されるオープンソースソフトウェアです。Amazon EKS Anywhere は、外部ネットワーク接続の有無にかかわらず、Snowball Edge デバイス上で使用できます。外部ネットワーク接続のないデバイスで Amazon EKS Anywhere を使用するには、Snowball Edge デバイスで実行するコンテナレジストリを指定します。Amazon EKS Anywhere に関する一般的な情報については、「[Amazon EKS Anywhere documentation](#)」を参照してください。

Amazon EKS Anywhere on AWS Snow を使用すると、次の機能を利用できます。

- Amazon EKS Anywhere CLI (`eksctl anywhere`) を使用して、Snowball Edge コンピューティング最適化デバイスに Kubernetes (K8s) クラスターをプロビジョニングする。Amazon EKS Anywhere を 1 台の Snowball Edge デバイスにプロビジョニングすることも、高可用性実現のために 3 台以上のデバイスにプロビジョニングすることもできます。
- Cilium Container Network Interface (CNI) のサポート
- ノードオペレーティングシステムとしての Ubuntu 20.04 のサポート

この図は、Snowball Edge デバイスにデプロイされた Amazon EKS Anywhere クラスターを示しています。



Amazon EKS Anywhere でサポートされている最新の Kubernetes バージョンで Kubernetes クラスターを作成することをお勧めします。詳細については、「[Amazon EKS-Anywhere バージョニング](#)」を参照してください。アプリケーションに特定のバージョンの Kubernetes が必要な場合は、Amazon EKS の標準サポートまたは拡張サポートで提供されている任意のバージョンの Kubernetes を使用してください。デプロイのライフサイクルを計画する際は、Kubernetes バージョ

ンのリリース日とサポート日を考慮してください。これにより、使用する予定の Kubernetes のバージョンのサポートが失われる可能性を回避できます。詳細については、[「Amazon EKS Kubernetes リリースカレンダー」](#)を参照してください。

Amazon EKS Anywhere on AWS Snow の詳細については、[Amazon EKS Anywhere のドキュメント](#)を参照してください。

トピック

- [Amazon EKS Anywhere on AWS Snow 用の Snowball Edge デバイスを注文する前に完了すべきアクション](#)
- [Amazon EKS Anywhere on Snow で使用する AWS Snowball Edge デバイスの注文](#)
- [Snowball Edge デバイスで Amazon EKS Anywhere を設定して実行する](#)
- [切断されたオペレーションのための Amazon EKS Anywhere on AWS Snow の設定](#)
- [Snowball Edge デバイスでのクラスターの作成と保守](#)

Amazon EKS Anywhere on AWS Snow 用の Snowball Edge デバイスを注文する前に完了すべきアクション

現時点では、Amazon EKS Anywhere は、Snowball Edge コンピューティング最適化デバイスとグラフィックスプロセッシングユニット (GPU) 搭載のコンピューティング最適化デバイスと互換性があります。Snowball Edge デバイスを注文する前に、準備しておくべきことがいくつかあります。

- デバイス上に仮想マシンを作成するために使用するオペレーティングシステムイメージを構築して提供します。
- ネットワークには、K8s コントロールプレーンエンドポイントで使用できる静的 IP アドレスがあり、アドレス解決プロトコル (ARP) を許可する必要があります。
- Snowball Edge デバイスでは、特定のポートが開いている必要があります。ポートの詳細については、「Amazon EKS Anywhere documentation」の「[Ports and protocols](#)」を参照してください。

トピック

- [Ubuntu EKS Distro AMI を作成する](#)
- [Harbor AMI を構築する](#)

Ubuntu EKS Distro AMI を作成する

Ubuntu EKS Distro AMI を作成するには、「[Build Snow node images](#)」を参照してください。

生成された AMI の名前は `capa-ami-ubuntu-20.04-version-timestamp` のパターンに従います。例えば、`capa-ami-ubuntu-20.04-v1.24-1672424524` となります。

Harbor AMI を構築する

Snowball Edge デバイスに含めるように Harbor プライベートレジストリ AMI を設定すると、外部ネットワークに接続しなくてもデバイスで Amazon EKS Anywhere を使用できます。Snowball Edge デバイスが外部ネットワークから切断されているときに Amazon EKS Anywhere を使用しない場合、またはデバイスで使用する AMI にプライベート Kubernetes レジストリがある場合は、このセクションをスキップできます。

Harbor ローカルレジストリ AMI を作成するには、「[Build a Harbor AMI](#)」を参照してください。

Amazon EKS Anywhere on Snow で使用する AWS Snowball Edge デバイスの注文

Snowball Edge コンピューティング最適化デバイス、または GPU 搭載のコンピューティング最適化デバイスを注文するには、本ガイドの「[Snow Family デバイスを注文するジョブの作成](#)」を参照し、注文プロセス中に以下の点に留意してください。

- ステップ 1 で、[ローカルコンピューティングとストレージのみ] のジョブタイプを選択します。
- ステップ 2 で、[Snowball Edge Compute Optimized] または [Snowball Edge Compute Optimized (GPU 搭載)] のデバイスタイプを選択します。
- ステップ 3 で、AWS Snow 上の Amazon EKS Anywhere を選択し、必要な Kubernetes バージョンを選択します。

Note

最新のソフトウェアを配信するために、現在利用可能なバージョンよりも新しいバージョンの EKS Anywhere を使用してデバイスを設定する場合があります。詳細については、「[Amazon EKS ユーザーガイド](#)」の「[バージョンニング](#)」を参照してください。Amazon EKS Anywhere でサポートされている最新の Kubernetes バージョンで Kubernetes クラスターを作成することをお勧めします。詳細については、「[Amazon EKS-Anywhere バージョニング](#)」を参照してください。アプリケーションに特定のバー

ジョンの Kubernetes が必要な場合は、Amazon EKS の標準サポートまたは拡張サポートで提供されている任意のバージョンの Kubernetes を使用してください。デプロイのライフサイクルを計画する際は、Kubernetes バージョンのリリース日とサポート日を考慮してください。これにより、使用する予定の Kubernetes のバージョンのサポートが失われる可能性を回避できます。詳細については、[「Amazon EKS Kubernetes リリースカレンダー」](#)を参照してください。

- EKS Distro AMI ([「Ubuntu EKS Distro AMI を作成する」](#) を参照) や、オプションで構築した Harbor AMI ([「Harbor AMI を構築する」](#) を参照) など、デバイスに含める AMI を選択します。
- 高可用性を実現するために複数の Snowball Edge デバイスが必要な場合は、[高可用性] から必要なデバイスの数を選択します。

Snowball Edge デバイス (1 台または複数) を受け取ったら、[Snowball Edge デバイスで Amazon EKS Anywhere を設定して実行する](#) に従って Amazon EKS Anywhere を設定します。

Snowball Edge デバイスで Amazon EKS Anywhere を設定して実行する

以下の手順に従って、Snowball Edge デバイスで Amazon EKS Anywhere を設定して起動します。次に、接続されていないデバイスでも動作するように Amazon EKS Anywhere を設定するには、デバイスを外部ネットワークから切断する前に追加の手順を実行します。詳細については、[「切断されたオペレーションのための Amazon EKS Anywhere on AWS Snow の設定」](#)を参照してください。

トピック

- [初期セットアップ](#)
- [Snowball Edge デバイスで Amazon EKS Anywhere を自動的に設定して実行する](#)
- [Snowball Edge デバイスで Amazon EKS Anywhere を手動で設定して実行する](#)

初期セットアップ

デバイスをローカルネットワークに接続し、Snowball Edge クライアントをダウンロードし、認証情報を取得し、デバイスのロックを解除して、各 Snowball Edge デバイスで初期セットアップを実行します。

初期セットアップを実行する

1. Snowball Edge クライアントのダウンロードとインストール 詳細については、[「Snowball Edge クライアントのダウンロードとインストール」](#)を参照してください。

2. デバイスをローカルネットワークに接続します。詳細については、「[ローカルネットワークへの接続](#)」を参照してください。
3. 認証情報を取得して、デバイスのロックを解除します。詳細については、「[Snow Family デバイスにアクセスするための認証情報の取得](#)」を参照してください。
4. デバイスのロックを解除します。詳細については、「[Snow Family デバイスのロック解除](#)」を参照してください。デバイスを手動でロック解除する代わりにスクリプトツールを使用することもできます。「[Unlock devices](#)」を参照してください。

Snowball Edge デバイスで Amazon EKS Anywhere を自動的に設定して実行する

サンプルスクリプトツールを使用して環境を設定し、Amazon EKS Anywhere 管理インスタンスを実行するか、手動で実行することができます。スクリプトツールを使用するには、「[Unlock devices and setup environment for Amazon EKS Anywhere](#)」を参照してください。環境を設定して Amazon EKS Anywhere 管理インスタンスを実行した後、ネットワークに接続されていない Snowball Edge デバイス上で動作するように Amazon EKS Anywhere を設定する必要がある場合は、「[切断されたオペレーションのための Amazon EKS Anywhere on AWS Snow の設定](#)」を参照してください。それ以外の場合は、「[Snowball Edge デバイスでのクラスターの作成と保守](#)」を参照してください。

手動で環境を設定し、Amazon EKS Anywhere 管理インスタンスを実行するには、「[Snowball Edge デバイスで Amazon EKS Anywhere を手動で設定して実行する](#)」を参照してください。

Snowball Edge デバイスで Amazon EKS Anywhere を手動で設定して実行する

トピック

- [AWS CLI プロファイルを作成する](#)
- [Amazon EKS Anywhere IAM ローカルユーザーを作成する](#)
- [\(オプション\) Secure Shell キーを作成してインポートする](#)
- [Amazon EKS Anywhere 管理インスタンスを実行し、認証情報と証明書ファイルをそのインスタンスに転送する](#)

AWS CLI プロファイルを作成する

Snowball Edge デバイスと Amazon EKS Anywhere 管理インスタンスを設定するプロセス全体で使用する認証情報を保存する AWS CLI プロファイルを作成します。AWS CLI プロファイルの詳細については、ユーザーガイドの「[の名前付きプロファイル AWS CLI AWS Command Line Interface](#)」を参照してください。

サンプルスクリプトツールを使用して、AWS CLI プロファイルと Amazon EKS Anywhere ローカル IAM ユーザーを自動的に作成できます。「[Create credentials and certificates file](#)」を参照してください。スクリプトを使用した後、[\(オプション\) Secure Shell キーを作成してインポートする](#) から再開します。それ以外の場合は、次の手順を実行してから、「[Amazon EKS Anywhere IAM ローカルユーザーを作成する](#)」の手順を実行してください。

 Note

設定する Snowball Edge デバイスごとにこれを行います。

```
PATH_TO_Snowball_Edge_CLIENT/bin/snowballEdge list-access-keys --endpoint
https://snowball-ip --manifest-file path-to-manifest-file --unlock-code unlock-code
{
  "AccessKeyIds" : [ "xxxx" ]
}
```

AccessKeyIds の値を get-secret-access-key コマンドの access-key-id パラメータの値として使用します。

```
PATH_TO_Snowball_Edge_CLIENT/bin/snowballEdge get-secret-access-key --access-key-
id ACCESS_KEY_ID --endpoint https://snowball-ip --manifest-file path-to-manifest-file
--unlock-code unlock-code
[snowballEdge]
aws_access_key_id = xxx
aws_secret_access_key = xxx
```

aws_access_key_id および の値を AWS CLI プロファイルの AWS Access Key ID および AWS Secret Access Key の値 aws_secret_access_key として使用します。

```
aws configure --profile profile-name
AWS Access Key ID [None]: aws_access_key_id
AWS Secret Access Key [None]: aws_secret_access_key
Default region name [None]: snow
```


Amazon EKS Anywhere IAM ローカルユーザーを作成する

セキュリティ上のベストプラクティスとして、Snowball Edge デバイスで Amazon EKS Anywhere のローカル IAM ユーザーを作成します。これは、次の手順を使用して手動で行います。

Note

使用する Snowball Edge デバイスごとにこれを行います。

ローカルユーザーを作成する

create-user コマンドを使用して Amazon EKS Anywhere IAM ユーザーを作成します。

```
aws iam create-user --user-name user-name --endpoint http://snowball-ip:6078 --
profile profile-name
{
  "User": {
    "Path": "/",
    "UserName": "eks-a-user",
    "UserId": "AIDACKCEVSQ6C2EXAMPLE",
    "Arn": "arn:aws:iam::123456789012:user/eks-a-user",
    "CreateDate": "2022-04-06T00:13:35.665000+00:00"
  }
}
```

ローカルユーザー用のポリシーを作成します。

ポリシードキュメントを作成し、これを使用して IAM ポリシーを作成し、そのポリシーを Amazon EKS Anywhere ローカルユーザーにアタッチします。

ポリシードキュメントを作成して Amazon EKS Anywhere ローカルユーザーにアタッチするには

1. ポリシードキュメントを作成してコンピュータに保存します。以下のポリシーをドキュメントにコピーします。

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

```

{
  "Effect": "Allow",
  "Action": [
    "snowballdevice:DescribeDevice",
    "snowballdevice:CreateDirectNetworkInterface",
    "snowballdevice>DeleteDirectNetworkInterface",
    "snowballdevice:DescribeDirectNetworkInterfaces",
    "snowballdevice:DescribeDeviceSoftware"
  ],
  "Resource": ["*"]
},
{
  "Effect": "Allow",
  "Action": [
    "ec2:RunInstances",
    "ec2:DescribeInstances",
    "ec2:TerminateInstances",
    "ec2:ImportKeyPair",
    "ec2:DescribeKeyPairs",
    "ec2:DescribeInstanceTypes",
    "ec2:DescribeImages",
    "ec2>DeleteTags"
  ],
  "Resource": ["*"]
}
]
}

```

2. `create-policy` コマンドを使用して、ポリシードキュメントに基づいて IAM ポリシーを作成します。`--policy-document` パラメータの値には、ポリシーファイルへの絶対パスを使用する必要があります。例えば、次のようになります: `file:///home/user/policy-name.json`

```

aws iam create-policy --policy-name policy-name --policy-document file:///home/user/policy-name.json --endpoint http://snowball-ip:6078 --profile profile-name
{
  "Policy": {
    "PolicyName": "policy-name",
    "PolicyId":
"ANPACEMGEZDGNBVG3TQ0JQGEZAAAABP76TE5MKAAAABCC0TR2IJ43NBTJRZBU",
    "Arn": "arn:aws:iam::123456789012:policy/policy-name",
    "Path": "/",

```

```
    "DefaultVersionId": "v1",
    "AttachmentCount": 0,
    "IsAttachable": true,
    "CreateDate": "2022-04-06T04:46:56.907000+00:00",
    "UpdateDate": "2022-04-06T04:46:56.907000+00:00"
  }
}
```

3. `attach-user-policy` コマンドを使用して、IAM ポリシーを Amazon EKS Anywhere ローカルユーザーにアタッチします。

```
aws iam attach-user-policy --policy-arn policy-arn --user-name user-name --endpoint
http://snowball-ip:6078 --profile profile-name
```

アクセスキーと認証情報ファイルを作成する

Amazon EKS Anywhere IAM ローカルユーザーのアクセスキーを作成します。次に、認証情報ファイルを作成し、ローカルユーザー用に生成された `AccessKeyId` と `SecretAccessKey` の値をその中に含めます。認証情報ファイルは、後で Amazon EKS Anywhere 管理インスタンスによって使用されます。

1. `create-access-key` コマンドを使用して、Amazon EKS Anywhere ローカルユーザーのアクセスキーを作成します。

```
aws iam create-access-key --user-name user-name --endpoint http://snowball-ip:6078
--profile profile-name
{
  "AccessKey": {
    "UserName": "eks-a-user",
    "AccessKeyId": "AKIAIOSFODNN7EXAMPLE",
    "Status": "Active",
    "SecretAccessKey": "RTT/wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY",
    "CreateDate": "2022-04-06T04:23:46.139000+00:00"
  }
}
```

2. 認証情報ファイルを作成します。その中に、AccessKeyId と SecretAccessKey の値を次の形式で保存します。

```
[snowball-ip]
aws_access_key_id = ABCDEFGHIJKLMNOPQR2T
aws_secret_access_key = AfSD7sYz/TBZtzkReBl6PuuISzJ2WtNkeePw+nNzJ
region = snow
```

Note

複数の Snowball Edge デバイスを使用している場合、ファイル内の認証情報の順序は関係ありませんが、すべてのデバイスの認証情報を 1 つのファイルに含める必要があります。

管理インスタンス用の証明書ファイルを作成する

Amazon EKS Anywhere 管理インスタンスを実行するには、Snowball Edge デバイスの証明書が必要です。Snowball Edge デバイスにアクセスするための証明書を含む証明書ファイルを作成し、後で Amazon EKS Anywhere 管理インスタンスで使用できるようにします。

証明書ファイルを作成するには

1. `list-certificates` コマンドを使用して、使用を予定している各 Snowball Edge デバイスの証明書を取得します。

```
PATH_TO_Snowball_Edge_CLIENT/bin/snowballEdge list-certificates --endpoint
https://snowball-ip --manifest-file path-to-manifest-file --unlock-code unlock-
code
{
  "Certificates" : [ {
    "CertificateArn" : "arn:aws:snowball-device:::certificate/xxx",
    "SubjectAlternativeNames" : [ "ID:JID-xxx" ]
  } ]
}
```

2. CertificateArn の値を get-certificate コマンドの --certificate-arn パラメータの値として使用します。

```
PATH_TO_Snowball_Edge_CLIENT/bin/snowballEdge get-certificate --certificate-arn ARN
--endpoint https://snowball-ip --manifest-file path-to-manifest-file --unlock-
code unlock-code
```

3. デバイス証明書ファイルを作成します。get-certificate の出力を証明書ファイルに含めます。以下は、出力の保存方法の例です。

Note

複数の Snowball Edge デバイスを使用している場合、ファイル内の認証情報の順序は関係ありませんが、すべてのデバイスの認証情報を 1 つのファイルに含める必要があります。

```
-----BEGIN CERTIFICATE-----
ZWtzYSBzBm93IHRlc3QgY2VydGlmYWdhGUgZWtzYSBzBm93IHRlc3QgY2VydGlm
aWNhdGV1a3NhIHhNub3cgdGVzdCBjZXJ0aWZpY2F0ZWVrc2Egc25vdyB0ZXN0IGN1
cnRpZm1jYXRlZWtzYSBzBm93IHRlc3QgY2VydGlmYWdhGUgV1a3NhIHhNub3cgdGVz
dCBjZXJ0aWZpY2F0ZQMIIDXCcAkSgAwIBAgIJJAISM0nTVmbj+MA0GCSqGSIb3DQ
...
-----END CERTIFICATE-----
```

4. 「[Amazon EKS Anywhere IAM ローカルユーザーを作成する](#)」の手順を繰り返して、すべての Snowball Edge デバイスで Amazon EKS Anywhere の IAM ローカルユーザーを作成します。

(オプション) Secure Shell キーを作成してインポートする

このオプションの手順を使用して、すべての Amazon EKS Anywhere ノードインスタンスにアクセスするための Secure Shell (SSH) キーを作成し、そのパブリックキーをすべての Snowball Edge デバイスにインポートします。このキーファイルは安全に保管してください。

この手順をスキップした場合、Amazon EKS Anywhere は必要なときに SSH キーを自動的に作成してインポートします。このキーは `${PWD}/${CLUSTER_NAME}/eks-a-id_rsa` の管理インスタンスに保存されます。

SSH キーを作成して Amazon EKS Anywhere インスタンスにインポートする

1. `ssh-keygen` コマンドを使用して SSH キーを生成します。

```
ssh-keygen -t rsa -C "key-name" -f path-to-key-file
```

2. `import-key-pair` コマンドを使用して、コンピュータから Snowball Edge デバイ스에 キーをインポートします。

Note

キーをすべてのデバイスにインポートする場合、`key-name` パラメータの値は同じでなければなりません。

```
aws ec2 import-key-pair --key-name key-name --public-key-material file:///path/to/key-file --endpoint http://snowball-ip:8008 --profile profile-name
{
  "KeyFingerprint": "5b:0c:fd:e1:a0:69:05:4c:aa:43:f3:3b:3e:04:7f:51",
  "KeyName": "default",
  "KeyPairId": "s.key-85edb5d820c92a6f8"
}
```

Amazon EKS Anywhere 管理インスタンスを実行し、認証情報と証明書ファイルをそのインスタンスに転送する

Amazon EKS Anywhere 管理インスタンスを実行する

以下の手順に従って、Amazon EKS Anywhere 管理インスタンスを手動で実行し、管理インスタンスに仮想ネットワークインターフェイス (VNI) を設定し、インスタンスのステータスを確認し、SSH キーを作成し、これを使用して管理インスタンスに接続します。サンプルスクリプトツールを使用す

ると、Amazon EKS Anywhere 管理インスタンスの作成と、このインスタンスへの認証情報と証明書ファイルの転送を自動化できます。「[Create Amazon EKS Anywhere admin instance](#)」を参照してください。スクリプトツールが完了したら、「[Snowball Edge デバイスでのクラスターの作成と保守](#)」を参照してインスタンスに SSH 接続し、クラスターを作成できます。Amazon EKS Anywhere インスタンスを手動でセットアップする場合は、以下のステップに従います。

Note

クラスターのプロビジョニングに複数の Snowball Edge デバイスを使用している場合は、どの Snowball Edge デバイスでも Amazon EKS Anywhere 管理インスタンスを起動できます。

Amazon EKS Anywhere 管理インスタンスを実行するには

1. `create-key-pair` コマンドを使用して Amazon EKS Anywhere 管理インスタンス用の SSH キーを作成します。このコマンドはキーを `$PWD/key-file-name` に保存します。

```
aws ec2 create-key-pair --key-name key-name --query 'KeyMaterial' --output text --endpoint http://snowball ip:8008 --profile profile-name > key-file-name
```

2. `describe-images` コマンドを使用して、`eks-anywhere-admin` で始まるイメージ名を出力で検索します。

```
aws ec2 describe-images --endpoint http://snowball-ip:8008 --profile profile-name
```

3. `run-instance` コマンドを使用して、Amazon EKS Anywhere 管理イメージで `eks-a` 管理インスタンスを起動します。

```
aws ec2 run-instances --image-id eks-a-admin-image-id --key-name key-name --instance-type sbe-c.xlarge --endpoint http://snowball-ip:8008 --profile profile-name
```

4. Amazon EKS Anywhere インスタンスのステータスを確認するには、`describe-instances` コマンドを使用します。コマンドが `running` のインスタンスの状態を示すまで待つから、次に進みます。

```
aws ec2 describe-instances --instance-id instance-id --endpoint http://snowball-ip:8008 --profile profile-name
```

5. describe-device コマンドの出力から、ネットワークに接続されている物理ネットワークインターフェイスの PhysicalNetworkInterfaceId の値を書き留めます。これを使用して VNI を作成します。

```
PATH_TO_Snowball_Edge_CLIENT/bin/snowballEdge describe-device --endpoint https://snowball-ip --manifest-file path-to-manifest-file --unlock-code unlock-code
```

6. Amazon EKS Anywhere 管理インスタンス用の VNI を作成します。PhysicalNetworkInterfaceId の値を physical-network-interface-id パラメータの値として使用します。

```
PATH_TO_Snowball_Edge_CLIENT/bin/snowballEdge create-virtual-network-interface --ip-address-assignment dhcp --physical-network-interface-id PNI --endpoint https://snowball-ip --manifest-file path-to-manifest-file --unlock-code unlock-code
```

7. IPAddress の値を associate-address コマンドの public-ip パラメータの値として使用して、パブリックアドレスを Amazon EKS Anywhere 管理インスタンスに関連付けます。

```
aws ec2 associate-address --instance-id instance-id --public-ip VNI-IP --endpoint http://snowball-ip:8008 --profile profile-name
```

8. Amazon EKS Anywhere の管理インスタンスに SSH で接続します。

```
ssh -i path-to-key ec2-user@VNI-IP
```


証明書と認証情報ファイルを管理インスタンスに転送する

Amazon EKS Anywhere 管理インスタンスが実行されたら、Snowball Edge デバイスの認証情報と証明書を管理インスタンスに転送します。「[アクセスキーと認証情報ファイルを作成する](#)」と「[管理インスタンス用の証明書ファイルを作成する](#)」で認証情報と証明書ファイルを保存したディレクトリで、次のコマンドを実行します。

```
scp -i path-to-key path-to-credentials-file path-to-certificates-file ec2-user@eks-admin-instance-ip:~
```

Amazon EKS Anywhere 管理インスタンス上のファイルの内容を確認します。以下は認証情報と証明書ファイルの例です。

```
[192.168.1.1]
aws_access_key_id = EMGEZDGNBVGy3TQ0JQGEZB5ULEAAIWHWUJDXEXAMPLE
aws_secret_access_key = AUHpqj00GZQHEyXDbN0neLN1fR0gEXAMPLE
region = snow

[192.168.1.2]
aws_access_key_id = EMGEZDGNBVGy3TQ0JQGEZG507F3FJUCMYRMI4KPIEXAMPLE
aws_secret_access_key = kY4C18+RJA wq/bu28Y8fUJepwqhDEXAMPLE
region = snow
```

```
-----BEGIN CERTIFICATE-----
ZWtzYSBzBm93IHR1c3QgY2VydG1maWNhdGUgZWtzYSBzBm93IHR1c3QgY2VydG1m
aWNhdGV1a3NhIHhNub3cGdGVzdCBjZXJ0aWZpY2F0ZWVrc2Egc25vdyB0ZXN0IGN1
cnRpZm1jYXRlZWtzYSBzBm93IHR1c3QgY2VydG1maWNhdGV1a3NhIHhNub3cGdGVz
dCBjZXJ0aWZpY2F0ZQMIIDXCcAkSgAwIBAgIJAISM0nTVmbj+MA0GCSqGSIb3DQ
...
-----END CERTIFICATE-----

-----BEGIN CERTIFICATE-----
KJ0FP12PAYPEjxr81/PoCXfZeARBzN9WLUH5yz1ta+sYUJouzhzWuLJYA1xqcCPY
mhV1kRsN4hVd1BNRNCCpRF766yjdJeibKVzXQxoXoZBjr0kuGwqRy3d3ndjK77h4
OR5Fv9mjGf7CjcaSjk/4iwmZvRSaQacb0YG5Gveb4mfUAuVtuFoMeYfnAgMBAAGj
azBpMAwGA1UdEqFMAMBAf8wHQYDVR00BBYEFL/bRcnBRuSM5+FcYFa8HfIBomdF
...
```

```
-----END CERTIFICATE-----
```

切断されたオペレーションのための Amazon EKS Anywhere on AWS Snow の設定

外部ネットワーク接続のない環境で Amazon EKS Anywhere を実行できるようにするには、ネットワークに接続されている Snowball Edge デバイスでこの Amazon EKS Anywhere の追加設定を行います。

独自のローカルのプライベート Kubernetes レジストリを使用して Amazon EKS Anywhere をオフラインで使用できるように設定するには、「EKS Anywhere documentation」の「[Registry Mirror configuration](#)」を参照してください。

Harbor プライベートレジストリ AMI を作成した場合は、このセクションの以下の手順に従ってください。

トピック

- [Snowball Edge デバイスで Harbor レジストリを設定する](#)
- [Amazon EKS Anywhere 管理インスタンスで Harbor レジストリを使用する](#)

Snowball Edge デバイスで Harbor レジストリを設定する

「[Configure Harbor on a Snowball Edge device](#)」を参照してください。

Amazon EKS Anywhere 管理インスタンスで Harbor レジストリを使用する

「[Import Amazon EKS Anywhere container images to the local Harbor registry on a Snowball Edge device](#)」を参照してください。

Snowball Edge デバイスでのクラスターの作成と保守

クラスター作成のベストプラクティス

Amazon EKS Anywhere クラスターを作成するには、「[Snow クラスターの作成](#)」を参照してください。

Snowball Edge デバイスで Amazon EKS Anywhere クラスターを作成するときは、次のベストプラクティスに注意してください。

- 静的 IP アドレス範囲にクラスターを作成する前に、同じ IP アドレス範囲を使用している Snowball Edge デバイスに他のクラスターがないことを確認してください。
- Snowball Edge デバイスで DHCP アドレス指定を使用してクラスターを作成する前に、クラスターに使用されているすべての静的 IP アドレス範囲が DHCP プールサブネットにないことを確認してください。
- 複数のクラスターを作成する場合は、1つのクラスターが正常にプロビジョニングされて実行されるまで待ってから、別のクラスターを作成します。

クラスターのアップグレード

Amazon EKS Anywhere 管理者 AMI または EKS Distro AMI をアップグレードするには、お問い合わせください AWS Support。AWS Support は、アップグレードされた AMI を含む Snowball Edge 更新を提供します。その後、Snowball Edge の更新をダウンロードしてインストールします。「[更新をダウンロードする](#)」および「[更新をインストールする](#)」を参照してください。

Amazon EKS Anywhere AMI をアップグレードしたら、新しい Amazon EKS Anywhere 管理者インスタンスを起動する必要があります。[Amazon EKS Anywhere 管理インスタンスを実行する](#) を参照してください。次に、キーファイル、クラスターフォルダ、認証情報、証明書を以前の管理インスタンスからアップグレードしたインスタンスにコピーします。これらは、クラスターにちなんだ名前のフォルダにあります。

クラスターリソースのクリーンアップ

Snowball Edge デバイスで複数のクラスターを作成して正しく削除しなかった場合や、クラスターに問題があり、再開後にクラスターが代替ノードを作成した場合、リソースリークが発生します。サンプルスクリプトツールを使用して、Amazon EKS Anywhere 管理者インスタンスと Snowball Edge デバイスのクリーニングを変更および使用できます。[Amazon EKS Anywhere on AWS Snow クリーンアップツール](#)」を参照してください。

ローカルでの IAM の使用

AWS Identity and Access Management (IAM) は、AWS Snowball Edge デバイスで実行される AWS リソースへのアクセスを安全に制御するのに役立ちます。IAM により、誰を認証 (サインイン) し、誰にリソースの使用を承認する (アクセス許可を持たせる) かを制御します。

IAM は、デバイスでローカルにサポートされています。ローカル IAM サービスを使用して新しいユーザーを作成し、IAM ポリシーをアタッチできます。これらのポリシーを使用して、割り当てら

れたタスクの実行に必要なアクセスを許可できます。例えば、ユーザーにデータの転送を許可する一方で、新しい Amazon EC2 互換インスタンスの作成は制限することができます。

さらに、デバイスで AWS Security Token Service (AWS STS) を使用して、ローカルのセッションベースの認証情報を作成できます。IAM サービスの詳細については、IAM ユーザーガイドの[使用開始](#)を参照してください。

デバイスのルート認証情報を無効にすることはできません。また、アカウント内のポリシーを使用して AWS アカウント ルートユーザーへのアクセスを明示的に拒否することはできません。ルートユーザーのアクセスキーを保護し、デバイスを日常的に操作するための IAM ユーザー認証情報を作成することをお勧めします。

Important

このセクションのドキュメントは、AWS Snowball Edge デバイス上で IAM をローカルで使用する場合に適用されます。IAM を使用方法については、AWS クラウド「」を参照してください [での Identity and Access Management AWS Snowball](#)。

AWS サービスが Snowball Edge で正しく動作するには、サービスのポートを許可する必要があります。詳細については、「[エッジデバイスで AWS サービス AWS Snowball を使用するために必要なポート](#)」を参照してください。

トピック

- [Snowball Edge での AWS CLI および API オペレーションの使用](#)
- [Snowball Edge でサポートされている IAM AWS CLI コマンドのリスト](#)
- [IAM ポリシーの例](#)
- [TrustPolicy 例](#)

Snowball Edge での AWS CLI および API オペレーションの使用

AWS CLI または API オペレーションを使用して Snowball Edge で IAM AWS STS、Amazon S3、および Amazon EC2 コマンドを発行する場合は、`region`「」と指定する必要があります。これは、次の例のように、コマンド自体で `aws configure` または `aws sts get-session-token` を使用して実行できます。

```
aws configure --profile abc
AWS Access Key ID [None]: AKIAIOSFODNN7EXAMPLE
```

```
AWS Secret Access Key [None]: 1234567
Default region name [None]: snow
Default output format [None]: json
```

または

```
aws iam list-users --profile snowballEdge --endpoint http://192.0.2.0:6078 --region
snow
```

Note

AWS Snowball Edge でローカルで使用されるアクセスキー ID とアクセスキーは、のキーと交換できません AWS クラウド。

Snowball Edge でサポートされている IAM AWS CLI コマンドのリスト

以下は、Snowball Edge デバイスでサポートされている IAM の AWS CLI コマンドとオプションのサブセットの説明です。コマンドまたはオプションが以下に示されていない場合、そのコマンドまたはオプションはサポートされていません。コマンドでサポートされていないパラメータについては、説明に記載されています。

- [attach-role-policy](#) – 指定された IAM ロールに指定された管理ポリシーをアタッチします。
- [attach-user-policy](#) – 指定された管理ポリシーを指定されたユーザーにアタッチします。
- [create-access-key](#) – 指定されたユーザーの新しいローカル IAM シークレットアクセスキーと対応する AWS アクセスキー ID を作成します。
- [create-policy](#) - デバイスの新しい IAM 管理ポリシーを作成します。
- [create-role](#) — デバイスの新しいローカル IAM ロールを作成します。以下のパラメータはサポートされていません。
 - Tags
 - PermissionsBoundary
- [create-user](#) — デバイスの新しいローカル IAM ユーザーを作成します。以下のパラメータはサポートされていません。

- Tags
- PermissionsBoundary
- [delete-access-key](#) - 指定されたユーザーの新しいローカル IAM シークレットアクセスキーと対応する AWS アクセスキー ID を削除します。
- [delete-policy](#) - 指定した管理ポリシーを削除します。
- [delete-role](#) - 指定したロールを削除します。
- [delete-user](#) - 指定したユーザーを削除します。
- [detach-role-policy](#) - 指定されたロールから指定された管理ポリシーを削除します。
- [detach-user-policy](#) - 指定されたユーザーから指定された管理ポリシーを削除します。
- [get-policy](#) - 指定した管理ポリシーについて、ポリシーのデフォルトバージョン、ポリシーがアタッチされているローカル IAM ユーザー、グループ、ロールの総数などの情報を取得します。
- [get-policy-version](#) - ポリシードキュメントを含む、指定された管理ポリシーの指定されたバージョンに関する情報を取得します。
- [get-role](#) - 指定したロールについて、そのロールのパス、GUID、ARN、ロールを継承するアクセス許可を付与する信頼ポリシーなどの情報を取得します。
- [get-user](#) - 指定した IAM ユーザーについて、ユーザーの作成日、パス、一意の ID、ARN などの情報を取得します。
- [list-access-keys](#) - 指定された IAM ユーザーに関連付けられたアクセスキー IDs に関する情報を返します。
- [list-attached-role-policies](#) - 指定された IAM ロールにアタッチされているすべての管理ポリシーを一覧表示します。
- [list-attached-user-policies](#) - 指定された IAM ユーザーにアタッチされているすべての管理ポリシーを一覧表示します。
- [list-entities-for-policy](#) - 指定された管理ポリシーがアタッチされているすべてのローカル IAM ユーザー、グループ、ロールを一覧表示します。
 - --EntityFilter: 値として user および role のみがサポートされます。
- [list-policies](#) - ローカル AWS アカウントで使用可能なすべての管理ポリシーを一覧表示します。次のパラメータはサポートされていません。
 - --PolicyUsageFilter
- [list-roles](#) - 指定したパスプレフィックスを持つローカル IAM ロールを一覧表示します。
- [list-users](#) - 指定したパスのプレフィックスを持つ IAM ユーザーを一覧表示します。

- [update-access-key](#) – 指定されたアクセスキーのステータスをアクティブから非アクティブに変更するか、またはその逆に変更します。
- [update-assume-role-policy](#) – ロールを引き受けるアクセス許可を IAM エンティティに付与するポリシーを更新します。
- [update-role](#) – ロールの説明または最大セッション継続時間の設定を更新します。
- [update-user](#) – 指定した IAM ユーザーの名前やパスを更新します。

サポートされている IAM API オペレーション

以下は、Snowball Edge で使用できる IAM API オペレーションと、IAM API リファレンスの説明へのリンクです。

- [AttachRolePolicy](#) – 指定された IAM ロールに指定された管理ポリシーをアタッチします。
- [AttachUserPolicy](#) – 指定された管理ポリシーを指定されたユーザーにアタッチします。
- [CreateAccessKey](#) – 指定されたユーザーの新しいローカル IAM シークレットアクセスキーと対応する AWS アクセスキー ID を作成します。
- [CreatePolicy](#) – デバイスの新しい IAM 管理ポリシーを作成します。
- [CreateRole](#) – デバイスの新しいローカル IAM ロールを作成します。
- [CreateUser](#) – デバイスの新しいローカル IAM ユーザーを作成します。

以下のパラメータはサポートされていません。

- Tags
- PermissionsBoundary
- [DeleteAccessKey](#) – 指定されたアクセスキーを削除します。
- [DeletePolicy](#) – 指定された管理ポリシーを削除します。
- [DeleteRole](#) – 指定されたロールを削除します。
- [DeleteUser](#) – 指定されたユーザーを削除します。
- [DetachRolePolicy](#) – 指定されたロールから指定された管理ポリシーを削除します。
- [DetachUserPolicy](#) – 指定されたユーザーから指定された管理ポリシーを削除します。
- [GetPolicy](#) – ポリシーのデフォルトバージョン、ポリシーがアタッチされているローカル IAM ユーザー、グループ、ロールの合計数など、指定された管理ポリシーに関する情報を取得します。
- [GetPolicyVersion](#) – ポリシードキュメントを含む、指定された管理ポリシーの指定されたバージョンに関する情報を取得します。

- [GetRole](#) – ロールのパス、GUID、ARN、ロールを引き受けるアクセス許可を付与するロールの信頼ポリシーなど、指定されたロールに関する情報を取得します。
- [GetUser](#) – ユーザーの作成日、パス、一意の ID、ARN など、指定された IAM ユーザーに関する情報を取得します。
- [ListAccessKeys](#) – 指定された IAM ユーザーに関連付けられたアクセスキー IDs に関する情報を返します。
- [ListAttachedRolePolicies](#) – 指定された IAM ロールにアタッチされているすべての管理ポリシーを一覧表示します。
- [ListAttachedUserPolicies](#) – 指定された IAM ユーザーにアタッチされているすべての管理ポリシーを一覧表示します。
- [ListEntitiesForPolicy](#) – ユーザーの作成日、パス、一意の ID、ARN など、指定された IAM ユーザーに関する情報を取得します。
 - --EntityFilter: 値として user および role のみがサポートされます。
- [ListPolicies](#) – ローカルで利用可能なすべてのマネージドポリシーを一覧表示します AWS アカウント。次のパラメータはサポートされていません。
 - --PolicyUsageFilter
- [ListRoles](#) – 指定されたパスプレフィックスを持つローカル IAM ロールを一覧表示します。
- [ListUsers](#) – 指定されたパスプレフィックスを持つ IAM ユーザーを一覧表示します。
- [UpdateAccessKey](#) – 指定されたアクセスキーのステータスをアクティブから非アクティブに変更するか、またはその逆に変更します。
- [UpdateAssumeRolePolicy](#) – ロールを引き受けるアクセス許可を IAM エンティティに付与するポリシーを更新します。
- [UpdateRole](#) – ロールの説明または最大セッション期間の設定を更新します。
- [UpdateUser](#) – 指定された IAM ユーザーの名前やパスを更新します。

サポートされる IAM ポリシーのバージョンと文法

以下は、IAM ポリシーのローカル IAM サポートバージョン 2012-10-17 と、ポリシーの文法のサブセットです。

ポリシータイプ	サポートされている文法
ID ベースのポリシー (ユーザー/ロールポリシー)	「Effect」、「Action」および「Resource」

ポリシータイプ	サポートされている文法
	<p>Note</p> <p>ローカル IAM は、"Condition"、"NotAction"、"NotResource"、および "Principal" をサポートしていません。</p>
リソースベースのポリシー (ロールの信頼ポリシー)	<p>「Effect」、「Action」および「Principal」</p> <p>Note</p> <p>プリンシパルの場合、AWS アカウント ID またはプリンシパル ID のみが許可されます。</p>

IAM ポリシーの例

Note

AWS Identity and Access Management (IAM) ユーザーには、[AWS OpsHub for Snow Family アプリケーション](#)を使用して Snow Family デバイスを管理するための "snowballdevice:*" アクセス許可が必要です。

次に、Snowball Edge デバイスにアクセス許可を付与するポリシーの例を示します。

例 1: IAM API を使用したサンプルユーザーの GetUser 呼び出しを許可する

次のポリシーを使用して、IAM API を介したサンプルユーザーの GetUser 呼び出しを許可します。

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

```
        "Sid": "VisualEditor0",
        "Effect": "Allow",
        "Action": "iam:GetUser",
        "Resource": "arn:aws:iam:::user/example-user"
    }
]
}
```

例 2: Amazon S3 API へのフルアクセスを許可する

Amazon S3 API へのフルアクセスを許可するには、以下のポリシーを使用します。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "VisualEditor0",
      "Effect": "Allow",
      "Action": "s3:*",
      "Resource": "*"
    }
  ]
}
```

例 3: 特定の Amazon S3 バケットへの読み取りと書き込みアクセスを許可する

特定のバケットへの読み取りおよび書き込みアクセスを許可するには、次のポリシーを使用します。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "ListObjectsInBucket",
      "Effect": "Allow",
      "Action": "s3:ListBucket",
      "Resource": "arn:aws:s3:::bucket-name"
    },
    {
      "Sid": "AllObjectActions",
      "Effect": "Allow",
      "Action": "s3:*Object",
      "Resource": "arn:aws:s3:::bucket-name/*"
    }
  ]
}
```

```
    }  
  ]  
}
```

例 4: 特定の Amazon S3 バケットへの List、Get、および Put アクセスを許可する

次のポリシーを使用して、特定の S3 バケットへの List、Get、および Put アクセスを許可します。

```
{  
  "Version": "2012-10-17",  
  "Statement": [  
    {  
      "Effect": "Allow",  
      "Action": [  
        "s3:PutObject",  
        "s3:GetObject",  
        "s3:List*"  
      ],  
      "Resource": "arn:aws:s3:::examplebucket/*"  
    }  
  ]  
}
```

例 5: Amazon EC2 API へのフルアクセスを許可する

Amazon EC2 へのフルアクセスを許可するには、以下のポリシーを使用します。

```
{  
  "Version": "2012-10-17",  
  "Statement": [  
    {  
      "Effect": "Allow",  
      "Action": "ec2:*",  
      "Resource": "*"   
    }  
  ]  
}
```

例 6: Amazon EC2 互換インスタンスを開始および停止するためのアクセスを許可する

次のポリシーを使用して、Amazon EC2 インスタンスの開始と停止のアクセスを許可します。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "ec2:StartInstances",
        "ec2:StopInstances"
      ],
      "Resource": "*"
    }
  ]
}
```

例 7: への呼び出しを拒否する DescribeLaunchTemplates が、 へのすべての呼び出しを許可する DescribeImages

DescribeLaunchTemplates への呼び出しを拒否するが、DescribeImages へのすべての呼び出しを許可するには、次のポリシーを使用します。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Deny",
      "Action": [
        "ec2:DescribeLaunchTemplates"
      ],
      "Resource": "*"
    },
    {
      "Effect": "Allow",
      "Action": [
        "ec2:DescribeImages"
      ],
      "Resource": "*"
    }
  ]
}
```

例 8: API コールのポリシー

ユーザー定義のマネージドポリシーを含む、Snow デバイスで使用可能なすべてのマネージドポリシーを一覧表示します。詳細は、「[list-policies](#)」を参照してください。

```
aws iam list-policies --endpoint http://ip-address:6078 --profile snowballEdge --region snow
{
  "Policies": [
    {
      "PolicyName": "Administrator",
      "Description": "Root user admin policy for Account 123456789012",
      "CreateDate": "2020-03-04T17:44:59.412Z",
      "AttachmentCount": 1,
      "IsAttachable": true,
      "PolicyId": "policy-id",
      "DefaultVersionId": "v1",
      "Path": "/",
      "Arn": "arn:aws:iam::123456789012:policy/Administrator",
      "UpdateDate": "2020-03-04T19:10:45.620Z"
    }
  ]
}
```

TrustPolicy 例

信頼ポリシーは、通常アクセスできない AWS リソースにアクセスするために使用できる一時的なセキュリティ認証情報のセットを返します。これらの一時的な認証情報は、アクセスキー ID、シークレットアクセスキー、およびセキュリティトークンで構成されています。通常、クロスアカウントアクセスには、アカウントで AssumeRole を使用します。

以下に示しているのは、信頼ポリシーの例です。信頼ポリシーの詳細については、AWS Security Token Service API リファレンス [AssumeRole](#) の「」を参照してください。

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

```
        "arn:aws:iam::AccountId:root" //You can use the Principal ID
instead of the account ID.
    ]
  },
  "Action": [
    "sts:AssumeRole"
  ]
}
]
```

の使用 AWS Security Token Service

AWS Security Token Service (AWS STS) は、IAM ユーザーの権限が制限された一時的な認証情報をリクエストするのに役立ちます。

Important

AWS サービスが Snowball Edge で正しく動作するには、サービスのポートを許可する必要があります。詳細については、「[エッジデバイスで AWS サービス AWS Snowball を使用するために必要なポート](#)」を参照してください。

トピック

- [Snowball Edge での AWS CLI および API オペレーションの使用](#)
- [Snowball Edge でサポートされている AWS STS AWS CLI コマンド](#)
- [サポートされている AWS STS API オペレーション](#)

Snowball Edge での AWS CLI および API オペレーションの使用

AWS CLI または API オペレーションを使用して Snowball Edge デバイスで IAM AWS STS、Amazon S3、および Amazon EC2 コマンドを発行する場合は、を region 「」と指定する必要があります snow。これは、次の例のように、コマンド自体内で AWS configure または を使用して実行できます。

```
aws configure --profile snowballEdge
AWS Access Key ID [None]: defgh
AWS Secret Access Key [None]: 1234567
```

```
Default region name [None]: snow
Default output format [None]: json
```

または

```
aws iam list-users --profile snowballEdge --endpoint http://192.0.2.0:6078 --region
snow
```

Note

AWS Snowball Edge でローカルで使用するアクセスキー ID とアクセスシークレットキーは、のキーと交換できません AWS クラウド。

Snowball Edge でサポートされている AWS STS AWS CLI コマンド

[assume-role](#) コマンドのみがローカルでサポートされています。

assume-role では以下のパラメータがサポートされます。

- role-arn
- role-session-name
- duration-seconds

コマンドの例

ロールを引き受けるには、以下のコマンドを使用します。

```
aws sts assume-role --role-arn "arn:aws:iam::123456789012:role/example-role" --
role-session-name AWSCLI-Session --endpoint http://snow-device-IP-address:7078
```

assume-role コマンド使用の詳細については、[AWS CLIを使用して IAM ロールを引き受ける方法を教えてください。](#)を参照してください。

の使用の詳細については AWS STS、IAM [ユーザーガイドの「一時的なセキュリティ認証情報の使用」](#)を参照してください。

サポートされている AWS STS API オペレーション

[AssumeRole](#) API のみがローカルでサポートされています。

AssumeRole では以下のパラメータがサポートされます。

- RoleArn
- RoleSessionName
- DurationSeconds

例

ロールを引き受けるには、以下を使用します。

```
https://sts.amazonaws.com/  
?Version=2011-06-15  
&Action=AssumeRole  
&RoleSessionName=session-example  
&RoleArn=arn:aws:iam::123456789012:role/demo  
&DurationSeconds=3600
```

公開鍵証明書の管理

パブリックキー証明書を提供することで、HTTPS プロトコルを介して Snowball Edge デバイスまたは Snowball Edge デバイスのクラスターで実行されている AWS サービスと安全にやり取りできます。HTTPS プロトコルを使用して、Snow Family デバイスの IAM、Amazon EC2、S3 アダプター、Amazon S3 互換ストレージ、Amazon EC2 Systems Manager、Snowball Edge デバイス AWS STS などの AWS サービスとやり取りできます。デバイスのクラスターの場合、1 つの証明書が必要で、クラスター内のどのデバイスでも生成できます。Snowball Edge デバイスで証明書が生成され、デバイスのロックを解除すると、Snowball Edge クライアントコマンドを使用して証明書を一覧表示、取得、削除できます。

Snowball Edge デバイスは、以下のイベントが発生すると証明書を生成します。

- Snowball Edge デバイスまたはクラスターが初めてロック解除される。
- Snowball Edge デバイスまたはクラスターは、証明書を削除した後 (delete-certificate コマンドまたは の証明書の更新を使用) にロック解除されます AWS OpsHub。

- 証明書の有効期限が切れた後に Snowball Edge デバイスまたはクラスターが再起動およびロック解除されます。

新しい証明書が生成されるたびに、古い証明書は無効になります。証明書は、生成された日から 1 年間有効です。

AWS OpsHub for Snow Family を使用してパブリックキー証明書を管理することもできます。詳細については、このガイドの「[を使用したパブリックキー証明書の管理 OpsHub](#)」を参照してください。

トピック

- [証明書の一覧表示](#)
- [証明書の取得](#)
- [証明書の削除](#)

証明書の一覧表示

`list-certificates` コマンドを使用して、現在の証明書の Amazon リソースネーム (ARN) を確認します。

```
snowballEdge list-certificates
```

Example `list-certificates` の出力

```
{
  "Certificates" : [ {
    "CertificateArn" : "arn:aws:snowball-
device:::certificate/78EXAMPLE516EXAMPLEf538EXAMPLEa7",
    "SubjectAlternativeNames" : [ "192.0.2.0" ]
  } ]
}
```

証明書の取得

get-certificate コマンドを使用して、指定された ARN に基づいて証明書の内容を確認します。list-certificates コマンドを使用して、certificate-arn パラメータとして使用する証明書の ARN を取得します。

```
snowballEdge get-certificate --certificate-arn arn:aws:snowball-  
device:::certificate/78EXAMPLE516EXAMPLEf538EXAMPLEa7
```

Example get-certificate の出力

```
-----BEGIN CERTIFICATE-----  
Certificate  
-----END CERTIFICATE-----
```

証明書の更新に関する情報は、[S3 アダプターを AWS CLI エンドポイントとして指定する](#) を参照してください。

証明書の削除

delete-certificate コマンドを使用して、現在の証明書を削除します。list-certificates コマンドを使用して、certificate-arn パラメータとして使用する証明書の ARN を取得します。新しい証明書を生成するには、Snowball Edge またはクラスター内の各 Snowball Edge を再起動します。「[Snow Family デバイスの再起動](#)」を参照するか、snowballEdge reboot-device コマンドを使用してください。

```
snowballEdge delete-certificate --certificate-arn arn:aws:snowball-  
device:::certificate/78EXAMPLE516EXAMPLEf538EXAMPLEa7
```

Example delete-certificate の出力

```
The certificate has been deleted from your Snow device. Please reboot your Snowball  
Edge or Snowball Edge cluster to generate a new certificate.
```

エッジデバイスで AWS サービス AWS Snowball を使用するために必要なポート

AWS サービスが AWS Snowball Edge デバイスで正しく動作するには、サービスのネットワークポートを許可する必要があります。

以下は、各 AWS サービスに必要なネットワークポートのリストです。

[ポート]	[プロトコル]	コメント
22 (HTTP)	TCP	デバイスのヘルスチェックと EC2 SSH 用
443 (HTTPS)	TCP	S3 API と S3 コントロール API HTTPS エンドポイント
2049 (HTTP)	TCP	NFS エンドポイント
6078 (HTTP)	TCP	IAM HTTP エンドポイント
6089 (HTTPS)	TCP	IAM HTTPS エンドポイント
7078 (HTTP)	TCP	STS HTTP エンドポイント
7089 (HTTPS)	TCP	STS HTTPS エンドポイント
8080 (HTTP)	TCP	S3 アダプター HTTP エンドポイント
8008 (HTTP)	TCP	EC2 HTTP エンドポイント
8243 (HTTPS)	TCP	EC2 HTTPS エンドポイント
8443 (HTTPS)	TCP	S3 Adapter HTTPS エンドポイント

[ポート]	[プロトコル]	コメント
9091 (HTTP)	TCP	デバイス管理用エンドポイント
9092	TCP	EKS Anywhere と CAPAS デバイスコントローラーのインバウンド
8242	TCP	EKS Anywhere の EC2 HTTPS エンドポイントのインバウンド
6443	TCP	EKS Anywhere Kubernetes API エンドポイントのインバウンド
2379	TCP	EKS Anywhere Etcd API エンドポイントのインバウンド
2380	TCP	EKS Anywhere Etcd API エンドポイントのインバウンド

AWS Snow Device Management を使用してデバイスを管理する

AWS Snow Device Management では、Snow Family デバイスとローカル AWS サービスをリモートで管理できます。すべての Snow Family デバイスは Snow Device Management をサポートしており、Snow Family デバイスが利用可能なほとんどの AWS リージョン の新しいデバイスにインストールされています。

Snow Device Management では、次のタスクを実行できます。

- タスクを作成する
- タスクのステータスを確認する
- タスクのメタデータを確認する
- タスクをキャンセルする
- デバイス情報を確認する
- Amazon EC2 互換インスタンスのステータスを確認する
- コマンドと構文を一覧表示する
- リモート管理可能なデバイスを一覧表示する
- デバイスを横断してタスクのステータスを一覧表示する
- 利用可能なリソースを一覧表示する
- タスクをステータス別に一覧表示する
- デバイスタグまたはタスクタグを一覧表示する
- タグを適用する
- タグを削除する

トピック

- [Snow Family デバイスの注文時の Snow Device Management 状態の選択](#)
- [Snow Device Management のアクティブ化](#)
- [Snow Device Management のアクセス許可を IAM ロールに追加する](#)
- [Snow Device Management CLI コマンド](#)

Snow Family デバイスの注文時の Snow Device Management 状態の選択

Snow デバイスを注文するジョブを作成するときは、デバイスを受信したときに Snow Device Management がどの状態になるかを選択できます。インストールされているが、アクティブ化またはインストールされていないか、アクティブ化されていないかを選択できます。インストールされているがアクティブ化されていない場合は、AWS OpsHub または Snowball Edge クライアントを使用してアクティブ化する必要があります。インストールしてアクティブ化している場合は、デバイスを受信してローカルネットワークに接続した後に Snow Device Management を使用できません。Snowball Edge クライアント AWS Snow ファミリーマネジメントコンソール、AWS CLI または Snow ジョブ管理 API を使用してデバイスを注文するジョブを作成するときに、Snow Device Management の状態を選択できます。

から Snow Device Management の状態を選択するには AWS Snow ファミリーマネジメントコンソール

1. Snow Device Management をインストールしてアクティブ化するには、AWS OpsHub または Snowball クライアント を使用して Snow デバイスをリモートで管理を選択します。
2. Snow Device Management をインストールしてもアクティブ化しない場合は、AWS OpsHub または Snowball クライアント を使用して Snow デバイスをリモートで管理 を選択しないでください。

詳細については、このガイドの [「ステップ 3: 機能とオプションを選択する」](#) を参照してください。

、Snowball Edge クライアント AWS CLI、または Snow ジョブ管理 API から Snow Device Management の状態を選択するには :

- Snow Device Management の状態を指定するには、`remote-management`パラメータを使用します。パラメータの`INSTALLED_ONLY`値は、Snow Device Management がインストールされているが、アクティブ化されていないことを意味します。パラメータの`INSTALLED_AUTOSTART`値は、Snow Device Management がインストールされ、アクティブ化されていることを意味します。このパラメータの値を指定しない場合、`INSTALLED_ONLY`がデフォルト値です。

Example `create-job` コマンドの `remote-management`パラメータの構文の

```
aws snowball create-job \  
  --job-type IMPORT \  
  --remote-management INSTALLED_AUTOSTART \  
  --device-configuration '{"SnowconeDeviceConfiguration": {"WirelessConnection":  
{"IsWifiEnabled": false} } }' \  
  --resources '{"S3Resources": [{"BucketArn": "arn:aws:s3:::bucket-name"}]}' \  
  --description "Description here" \  
  --address-id ADID00000000-0000-0000-0000-000000000000 \  
  --kms-key-arn arn:aws:kms:us-west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab \  
  --role-arn arn:aws:iam::000000000000:role/SnowconeImportGamma \  
  --snowball-capacity-preference T8 \  
  --shipping-option NEXT_DAY \  
  --snowball-type SNC1_HDD \  
  --region us-west-2 \  

```

詳細については、[「API リファレンス」の「ジョブ管理 AWS Snowball API リファレンス」](#)を参照してください。

Snow Device Management のアクティブ化

Snowball Edge クライアントを使用して Snow Device Management をアクティブ化するには、次の手順に従います。

この手順を使用する前に、次の操作を行います。

- Snowball Edge クライアントの最新バージョンをダウンロードしてインストールします。詳細については、[Downloading and Installing the Snowball Client](#) を参照してください。
- マニフェストファイルをダウンロードし、Snow Family デバイスのロック解除コードを取得します。詳細については、[「認証情報とツールの取得」](#)を参照してください。
- Snow Family デバイスをローカルネットワークに接続します。詳細については、[「ローカルネットワークへの接続」](#)を参照してください。
- Snow Family デバイスのロックを解除します。詳細については、[「デバイスをローカルでロック解除する」](#)を参照してください。

```
snowballEdge set-features \  
  --remote-management-state INSTALLED_AUTOSTART \  
  --manifest-file JID1717d8cc-2dc9-4e68-aa46-63a3ad7927d2_manifest.bin \  

```

```
--unlock-code 7c0e1-bab84-f7675-0a2b6-f8k33 /  
--endpoint https://192.0.2.0:9091
```

コマンドが成功すると、Snowball Edge クライアントは以下を返します。

```
{  
  "RemoteManagementState" : "INSTALLED_AUTOSTART"  
}
```

Snow Device Management のアクセス許可を IAM ロールに追加する

デバイスの注文 AWS アカウント 元ので、(IAM) AWS Identity and Access Management ロールを作成し、次のポリシーをロールに追加します。次に、Snow Device Management を使って、デバイスをリモート管理するためにログインする IAM ユーザーに、このロールを割り当てます。詳細については、[IAM ロールの作成](#)および[AWS アカウントでの IAM ユーザーの作成](#)を参照してください。

Policy

```
{  
  "Version": "2012-10-17",  
  "Statement": [  
    {  
      "Effect": "Allow",  
      "Action": [  
        "snow-device-management:ListDevices",  
        "snow-device-management:DescribeDevice",  
        "snow-device-management:DescribeDeviceEc2Instances",  
        "snow-device-management:ListDeviceResources",  
        "snow-device-management:CreateTask",  
        "snow-device-management:ListTasks",  
        "snow-device-management:DescribeTask",  
        "snow-device-management:CancelTask",  
        "snow-device-management:DescribeExecution",  
        "snow-device-management:ListExecutions",  
        "snow-device-management:ListTagsForResource",  
      ]  
    }  
  ]  
}
```



```
        "snow-device-management:TagResource",
        "snow-device-management:UntagResource"
    ],
    "Resource": "*"
}
]
```

Snow Device Management CLI コマンド

このセクションでは、Snow Device Management を使用して Snow Family デバイスをリモートで管理するために使用できる AWS CLI コマンドについて説明します。を使用してリモート管理タスクを実行することもできます AWS OpsHub for Snow Family。詳細については、「[デバイス](#)」「[デバイスでの AWS サービスの管理](#)」を参照してください。

Note

デバイスを管理する前に、電源がオンになっており、ネットワークに接続されており、プロビジョニングされた AWS リージョン に接続できることを確認してください。

トピック

- [タスクを作成する](#)
- [タスクのステータスを確認する](#)
- [デバイス情報を確認する](#)
- [Amazon EC2 互換インスタンスのステータスを確認する](#)
- [タスクのメタデータを確認する](#)
- [タスクをキャンセルする](#)
- [コマンドと構文を一覧表示する](#)
- [リモート管理可能なデバイスを一覧表示する](#)
- [デバイスを横断してタスクのステータスを一覧表示する](#)
- [利用可能なリソースを一覧表示する](#)
- [デバイスタグまたはタスクタグを一覧表示する](#)

- [タスクをステータス別に一覧表示する](#)
- [タグを適用する](#)
- [タグを削除する](#)

タスクを作成する

1 台以上のターゲットデバイスに、ロック解除や再起動などのタスクを実行するよう指示を出すときは、`create-task` を使用します。ターゲットデバイスを指定するときは、`--targets` パラメータを使って管理対象のデバイスの ID の一覧を提供し、`--command` パラメータを使って実行するタスクを指定します。1 台のデバイスで実行できるコマンドは、1 回に 1 つのみです。

サポートされているコマンド

- `unlock` (引数なし)
- `reboot` (引数なし)

ターゲットデバイスごとに実行するタスクを作成するには、次のコマンドを使用します。`user input placeholder` を、ユーザー自身の情報に置き換えます。

コマンド

```
aws snow-device-management create-task
--targets smd-fictbgr3rbcjeqa5
--command reboot={}
```

例外

```
ValidationException
ResourceNotFoundException
InternalServerError
ThrottlingException
AccessDeniedException
ServiceQuotaExceededException
```

出力

```
{
  "taskId": "st-ficthmqoc2pht111",
  "taskArn": "arn:aws:snow-device-management:us-west-2:000000000000:task/st-
cjkwhmqoc2pht111"
}
```

タスクのステータスを確認する

1 台以上のターゲットデバイスで実行されている、リモートタスクのステータスを確認するには、`describe-execution` コマンドを使用します。

タスクは、次のいずれかのステータスになります。

- QUEUED
- IN_PROGRESS
- CANCELED
- FAILED
- COMPLETED
- REJECTED
- TIMED_OUT

タスクのステータスを確認するには、次のコマンドを使用します。*user input placeholder* を、ユーザー自身の情報に置き換えます。

コマンド

```
aws snow-device-management describe-execution \
--taskId st-ficthmqoc2pht111 \
--managed-device-id smd-fictqic6gcldf111
```

出力

```
{
  "executionId": "1",
  "lastUpdatedAt": "2021-07-22T15:29:44.110000+00:00",
  "managedDeviceId": "smd-fictqic6gcldf111",
  "startedAt": "2021-07-22T15:28:53.947000+00:00",
  "state": "SUCCEEDED",
  "taskId": "st-ficthmqoc2pht111"
}
```

デバイス情報を確認する

デバイスに固有の情報 (デバイスタイプ、ソフトウェアバージョン、IP アドレス、ロック状態など) を確認するには、`describe-device` コマンドを使用します。出力には、以下も含まれています。

- `lastReachedOutAt` - デバイスが最後に AWS クラウドに連絡した日時。デバイスがオンラインであることを示します。
- `lastUpdatedAt` — デバイスでデータが最後に更新された日時。デバイスキャッシュがいつ更新されたかを示します。

デバイス情報を確認するには、次のコマンドを使用します。*user input placeholder* を、ユーザー自身の情報に置き換えます。

コマンド

```
aws snow-device-management describe-device \  
--managed-device-id smd-fictqic6gcldf111
```

例外

```
ValidationException  
ResourceNotFoundException  
InternalServerError  
ThrottlingException  
AccessDeniedException
```

出力

```
{
  "associatedWithJob": "JID2bf11d5a-ea1e-414a-b5b1-3bf7e6a6e111",
  "deviceCapacities": [
    {
      "available": 158892032000,
      "name": "HDD Storage",
      "total": 158892032000,
      "unit": "Byte",
      "used": 0
    },
    {
      "available": 0,
      "name": "SSD Storage",
      "total": 0,
      "unit": "Byte",
      "used": 0
    },
    {
      "available": 3,
      "name": "vCPU",
      "total": 3,
      "unit": "Number",
      "used": 0
    },
    {
      "available": 5368709120,
      "name": "Memory",
      "total": 5368709120,
      "unit": "Byte",
      "used": 0
    },
    {
      "available": 0,
      "name": "GPU",
      "total": 0,
      "unit": "Number",
      "used": 0
    }
  ],
  "deviceState": "UNLOCKED",
  "deviceType": "SNC1_HDD",
}
```

```
"lastReachedOutAt": "2021-07-23T21:21:56.120000+00:00",
"lastUpdatedAt": "2021-07-23T21:21:56.120000+00:00",
"managedDeviceId": "smd-fictqic6gcldf111",
"managedDeviceArn": "arn:aws:snow-device-management:us-west-2:000000000000:managed-
device/smd-fictqic6gcldf111"
"physicalNetworkInterfaces": [
  {
    "defaultGateway": "10.0.0.1",
    "ipAddress": "10.0.0.2",
    "ipAddressAssignment": "DHCP",
    "macAddress": "ab:cd:ef:12:34:56",
    "netmask": "255.255.252.0",
    "physicalConnectorType": "RJ45",
    "physicalNetworkInterfaceId": "s.ni-530f866d526d4b111"
  },
  {
    "defaultGateway": "10.0.0.1",
    "ipAddress": "0.0.0.0",
    "ipAddressAssignment": "STATIC",
    "macAddress": "ab:cd:ef:12:34:57",
    "netmask": "0.0.0.0",
    "physicalConnectorType": "RJ45",
    "physicalNetworkInterfaceId": "s.ni-8abc787f0a6750111"
  }
],
"software": {
  "installState": "NA",
  "installedVersion": "122",
  "installingVersion": "NA"
},
"tags": {
  "Project": "PrototypeA"
}
}
```

Amazon EC2 互換インスタンスのステータスを確認する

Amazon EC2 インスタンスの現在のステータスを確認するには、`describe-ec2-instances` コマンドを使用します。出力は `describe-device` コマンドの出力と似ていますが、結果は のデバイス キャッシュから取得 AWS クラウド され、使用可能なフィールドのサブセットが含まれます。

Amazon EC2 互換インスタンスのステータスを確認するには、次のコマンドを使用します。*user input placeholder* を、ユーザー自身の情報に置き換えます。

コマンド

```
aws snow-device-management describe-device-ec2-instances \  
--managed-device-id smd-fictbgr3rbcje111 \  
--instance-ids s.i-84fa8a27d3e15e111
```

例外

```
ValidationException  
ResourceNotFoundException  
InternalServerError  
ThrottlingException  
AccessDeniedException
```

出力

```
{  
  "instances": [  
    {  
      "instance": {  
        "amiLaunchIndex": 0,  
        "blockDeviceMappings": [  
          {  
            "deviceName": "/dev/sda",  
            "ebs": {  
              "attachTime": "2021-07-23T15:25:38.719000-07:00",  
              "deleteOnTermination": true,  
              "status": "ATTACHED",  
              "volumeId": "s.vol-84fa8a27d3e15e111"  
            }  
          }  
        ],  
        "cpuOptions": {  
          "coreCount": 1,  
          "threadsPerCore": 1  
        }  
      }  
    }  
  ]  
}
```

```
    },
    "createdAt": "2021-07-23T15:23:22.858000-07:00",
    "imageId": "s.ami-03f976c3cadaa6111",
    "instanceId": "s.i-84fa8a27d3e15e111",
    "state": {
      "name": "RUNNING"
    },
    "instanceType": "snc1.micro",
    "privateIpAddress": "34.223.14.193",
    "publicIpAddress": "10.111.60.160",
    "rootDeviceName": "/dev/sda",
    "securityGroups": [
      {
        "groupId": "s.sg-890b6b4008bdb3111",
        "groupName": "default"
      }
    ],
    "updatedAt": "2021-07-23T15:29:42.163000-07:00"
  },
  "lastUpdatedAt": "2021-07-23T15:29:58.
071000-07:00"
}
]
}
```

タスクのメタデータを確認する

デバイスの任意のタスクのメタデータを確認するには、`describe-task` コマンドを使用します。タスクのメタデータには、次の項目が含まれています。

- ターゲットデバイス
- タスクのステータス
- タスクが作成された日時
- デバイスでデータが最後に更新された日時
- タスクが完了した日時
- タスクの作成時に記入された説明 (もしある場合)

タスクのメタデータを確認するには、次のコマンドを使用します。*user input placeholder* を、ユーザー自身の情報に置き換えます。

コマンド

```
aws snow-device-management describe-task \  
--task-id st-ficthmqoc2pht111
```

例外

```
ValidationException  
ResourceNotFoundException  
InternalServerError  
ThrottlingException  
AccessDeniedException
```

出力

```
{  
  "completedAt": "2021-07-22T15:29:46.758000+00:00",  
  "createdAt": "2021-07-22T15:28:42.613000+00:00",  
  "lastUpdatedAt": "2021-07-22T15:29:46.758000+00:00",  
  "state": "COMPLETED",  
  "tags": {},  
  "targets": [  
    "smd-fictbgr3rbcje111"  
  ],  
  "taskId": "st-ficthmqoc2pht111",  
  "taskArn": "arn:aws:snow-device-management:us-west-2:000000000000:task/st-ficthmqoc2pht111"  
}
```

タスクをキャンセルする

特定のタスクのキャンセルリクエストを送信するには、`cancel-task` コマンドを使用します。キャンセルできるのは、ステータスが `QUEUED` である、まだ実行されていないタスクのみです。既に実行されたタスクはキャンセルできません。

Note

キャンセルしようとしたタスクが、`cancel-task` コマンドでステータスを変更する前にキューから処理されると、その実行が継続される可能性があります。

タスクをキャンセルするには、次のコマンドを使用します。`user input placeholder` を、ユーザー自身の情報に置き換えます。

コマンド

```
aws snow-device-management cancel-task \  
--task-id st-ficthmqoc2pht111
```

例外

```
ValidationException  
ResourceNotFoundException  
InternalServerError  
ThrottlingException  
AccessDeniedException
```

出力

```
{  
  "taskId": "st-ficthmqoc2pht111"  
}
```

コマンドと構文を一覧表示する

Snow Device Management API でサポートされているすべてのコマンドの一覧を返すには、`help` コマンドを使用します。任意のコマンドの詳細情報および構文を返すときは、`help` コマンドも使用できます。

サポートされているすべてのコマンドを一覧表示するには、次のコマンドを使用します。

コマンド

```
aws snow-device-management help
```

コマンドの詳細情報および構文を返すときは、次のコマンドを使用します。*command* を、ユーザーが希望するコマンドの名前に置き換えます。

コマンド

```
aws snow-device-management command help
```

リモート管理可能なデバイスを一覧表示する

コマンドを実行している AWS リージョン で Snow Device Management を有効にしている、自身のアカウント上のすべてのデバイスの一覧を返すには、`list-devices` コマンドを使用します。`--max-results` と `--next-token` はオプションです。詳細については、[AWS CLI 「コマンドラインインターフェイスユーザーガイド」の「ページ分割オプションの使用」](#)を参照してください。AWS

リモート管理可能なデバイスを一覧表示するには、次のコマンドを使用します。*user input placeholder* を、ユーザー自身の情報に置き換えます。

コマンド

```
aws snow-device-management list-devices \  
--max-results 10
```

例外

```
ValidationException  
InternalServerError  
ThrottlingException
```

```
AccessDeniedException
```

出力

```
{
  "devices": [
    {
      "associatedWithJob": "ID2bf11d5a-ea1e-414a-b5b1-3bf7e6a6e111",
      "managedDeviceId": "smd-fictbgr3rbcjeqa5",
      "managedDeviceArn": "arn:aws:snow-device-management:us-
west-2:000000000000:managed-device/smd-fictbgr3rbcje111"
      "tags": {}
    }
  ]
}
```

デバイスを横断してタスクのステータスを一覧表示する

1 台以上のターゲットデバイスの、タスクのステータスを返すには、`list-executions` コマンドを使用します。返されたリストをフィルタリングして、現在、特定の状態にあるタスクを表示するには、`--state` パラメータを使用します。`--max-results` と `--next-token` はオプションです。詳細については、[AWS CLI 「コマンドラインインターフェイスユーザーガイド」の「ページ分割オプションの使用」](#)を参照してください。AWS

タスクは、次のいずれかのステータスになります。

- QUEUED
- IN_PROGRESS
- CANCELED
- FAILED
- COMPLETED
- REJECTED
- TIMED_OUT

デバイスを横断してタスクのステータスを一覧表示するには、次のコマンドを使用します。`user input placeholder` を、ユーザー自身の情報に置き換えます。

コマンド

```
aws snow-device-management list-executions \  
--taskId st-ficthmqoc2pht1ef \  
--state SUCCEEDED \  
--max-results 10
```

例外

```
ValidationException  
InternalServerError  
ThrottlingException  
AccessDeniedException
```

出力

```
{  
  "executions": [  
    {  
      "executionId": "1",  
      "managedDeviceId": "smd-fictbgr3rbcje111",  
      "state": "SUCCEEDED",  
      "taskId": "st-ficthmqoc2pht111"  
    }  
  ]  
}
```

利用可能なリソースを一覧表示する

デバイスで使用できる AWS リソースのリストを返すには、`list-device-resources` コマンドを使用します。特定のタイプのリソースでリストをフィルタリングするには、`--type` パラメータを使用します。現在、Amazon EC2 互換インスタンスは唯一サポートされているリソースタイプです。`--max-results` と `--next-token` はオプションです。詳細については、[AWS CLI 「コマンドラインインターフェイスユーザーガイド」の「ページ分割オプションの使用」](#)を参照してください。AWS

デバイスで利用可能なリソースを一覧表示するには、次のコマンドを使用します。*user input placeholder* を、ユーザー自身の情報に置き換えます。

コマンド

```
aws snow-device-management list-device-resources \
--managed-device-id smd-fictbgr3rbcje111 \
--type AWS::EC2::Instance
--next-
token YAQGPwAT9L3wVKaGYjt4yS34MiQLWvzcShe9oIeDJr05AT4rXSprqcqQhhBEYRfcerAp0YYbJmRT=
--max-results 10
```

例外

```
ValidationException
InternalServerError
ThrottlingException
AccessDeniedException
```

出力

```
{
  "resources": [
    {
      "id": "s.i-84fa8a27d3e15e111",
      "resourceType": "AWS::EC2::Instance"
    }
  ]
}
```

デバイスタグまたはタスクタグを一覧表示する

管理対象デバイスまたはタスクの、タグの一覧を返すには、`list-tags-for-resource` コマンドを使用します。

デバイスのタグを一覧表示するには、次のコマンドを使用します。例にある Amazon リソースネーム (ARN) を、お使いのデバイスの ARN に置き換えます。

コマンド

```
aws snow-device-management list-tags-for-resource
--resource-arn arn:aws:snow-device-management:us-west-2:123456789012:managed-device/
smd-fictbgr3rbcjeqa5
```

例外

```
AccessDeniedException
InternalServerError
ResourceNotFoundException
ThrottlingException
```

出力

```
{
  "tags": {
    "Project": "PrototypeA"
  }
}
```

タスクをステータス別に一覧表示する

`list-tasks` コマンドを使用して、コマンドが実行されている AWS リージョンのデバイスからタスクのリストを返します。IN_PROGRESS、COMPLETED、または CANCELED のステータス別に結果をフィルタリングするには、`--state` パラメータを使用します。`--max-results` と `--next-token` はオプションです。詳細については、[AWS CLI「コマンドラインインターフェイスユーザーガイド」の「ページ分割オプションの使用」](#)を参照してください。AWS

タスクをステータス別に一覧表示するには、次のコマンドを使用します。*user input placeholder* を、ユーザー自身の情報に置き換えます。

コマンド

```
aws snow-device-management list-tasks \
--state IN_PROGRESS \
--next-token K8VAMqKiP2Cf4xGkmH8GMyZrg0F8FUb+d10KTP9+P4pUb+8PhW+6MiXh4= \
```

```
--max-results 10
```

例外

```
ValidationException  
InternalServerError  
ThrottlingException  
AccessDeniedException
```

出力

```
{  
  "tasks": [  
    {  
      "state": "IN_PROGRESS",  
      "tags": {},  
      "taskId": "st-ficthmqoc2phtlef",  
      "taskArn": "arn:aws:snow-device-management:us-west-2:000000000000:task/st-ficthmqoc2phtlef"  
    }  
  ]  
}
```

タグを適用する

デバイスのタグ、またはデバイス上のタスクのタグを、追加または置換するには、`tag-resource` コマンドを使用します。`--tags` パラメータは、Key=Value ペアの、コンマ区切りのリストを受け入れます。

デバイスのタグを適用するには、次のコマンドを使用します。*user input placeholder* を、ユーザー自身の情報に置き換えます。

コマンド

```
aws snow-device-management tag-resource \  
--resource-arn arn:aws:snow-device-management:us-west-2:123456789012:managed-device/  
smd-fictbgr3rbcjeqa5 \  

```



```
--tags Project=PrototypeA
```

例外

```
AccessDeniedException  
InternalServerError  
ResourceNotFoundException  
ThrottlingException
```

タグを削除する

デバイスのタグ、またはデバイス上のタスクのタグを削除するには、`untag-resources` コマンドを使用します。

デバイスのタグを削除するには、次のコマンドを使用します。*user input placeholder* を、ユーザー自身の情報に置き換えます。

コマンド

```
aws snow-device-management untag-resources \  
--resource-arn arn:aws:snow-device-management:us-west-2:123456789012:managed-device/  
smd-fictbgr3rbcjeqa5 \  
--tag-keys Project
```

例外

```
AccessDeniedException  
InternalServerError  
ResourceNotFoundException  
ThrottlingException
```

AWS Snowball Edge ジョブについて

のジョブ AWS Snowball は個別の作業単位であり、コンソールまたはジョブ管理 API で作成するときに定義されます。AWS Snowball Edge デバイスには 3 つの異なるジョブタイプがあり、これらはすべてローカルストレージおよびコンピューティング機能を備えています。この機能では、ファイルインターフェイスまたは Amazon S3 インターフェイスを使用してデータを読み書きします。これは、AWS Snowball Edge デバイスでローカルに実行される Amazon S3 PUT オブジェクト API アクションに基づいて Lambda 関数をトリガーします。

- [Amazon S3 へのジョブのインポート](#) – 80 TB 以下のローカルデータを、単一のデバイスにコピーしてから Amazon S3 に移動させるデータ転送です。インポートジョブの場合、Snowball デバイスとジョブには one-to-one 関係があります。各ジョブに関連付けられるデバイスは 1 つのみです。より多くのデータをインポートする必要がある場合は、新しいインポートジョブを作成するか、既存のジョブをクローンできます。このジョブタイプのデバイスが返却されると、デバイス上のデータが Amazon S3 にインポートされます。
- [Amazon S3 からのジョブのエクスポート](#) – 任意の量のデータ (Amazon S3 内) を任意の数の Snowball Edge デバイスにコピーし、一度に 1 つの AWS Snowball Edge デバイスをオンプレミスのデータ送信先に移動すること。エクスポートジョブを作成すると、ジョブパートに分割されます。各ジョブパートのサイズは 80 TB 以下で、各ジョブパートには 1 つの AWS Snowball Edge デバイスしか関連付けられていません。このジョブタイプのデバイスが返却されると、内容は消去されます。
- [ローカルコンピューティングおよびストレージ専用ジョブ](#) – これらのジョブには、1 つの AWS Snowball Edge デバイス、またはクラスターで使用される複数のデバイスが含まれます。これらのジョブは、エクスポートジョブのようにバケットのデータから開始することはありません。また、インポートジョブのように、最後に Amazon S3 にデータをインポートすることもできません。このジョブタイプのデバイスが返却されると、内容は消去されます。このジョブタイプでは、デバイスのクラスターを作成することもできます。クラスターによってローカルストレージの耐久性を向上させ、ローカルストレージ容量を拡大縮小できます。

Lambda を使用できないリージョンでは、このジョブのタイプはローカルストレージ専用と呼ばれます。

ジョブの詳細

ジョブを作成する前に、[前提条件](#)を満たしていることを確認してください。各ジョブは、作成時に指定した詳細で定義されます。次の表で、ジョブのすべての詳細を説明します。

コンソール識別子	API 識別子	詳細情報
ジョブ名	Description	英数字、スペース、任意の Unicode 特殊文字からなるジョブの名前。
ジョブタイプ	JobType	ジョブのタイプ。インポート、エクスポート、またはローカルコンピューティングおよびストレージです。
ジョブ ID	JobId	ジョブを識別する一意の 39 文字のラベル。ジョブ ID は E Ink ディスプレイに表示される配送ラベルの下部に表示されています。また、ジョブのマニフェストファイル名にも入っています。
Address	AddressId	デバイスの送付先住所。API の場合は、アドレスデータ型の ID です。
作成日	CreationDate	このジョブを作成した日。
配送速度	ShippingOption	速度オプションはリージョンによります。詳細については、「 配送速度 」を参照してください。
IAM ロール ARN	RoleARN	この Amazon リソースネーム (ARN) は、Amazon S3 バケットの書き込み権限を持つジョブの作成時に作成される AWS Identity and Access Management (IAM) ロールです。Amazon S3 作成プロセスは自動的に行われ、引き受

コンソール識別子	API 識別子	詳細情報
		<p>けること AWS Snowball を許可する IAM ロールは、S3 バケットと Snowball 間でデータをコピーする場合にのみ使用されます。詳細については、「AWS Snowball コンソールを使用するために必要なアクセス許可」を参照してください。</p>
AWS KMS キー	KmsKeyARN	<p>では AWS Snowball、AWS Key Management Service (AWS KMS) は各 Snowball のキーを暗号化します。ジョブを作成するときは、所有する AWS KMS 暗号化キーの ARN も選択または作成します。詳細については、「AWS Key Management ServiceAWS Snowball Edge の」を参照してください。</p>
Snowball のキャパシティー	SnowballCapacityPreference	<p>このジョブで注文された AWS Snowball デバイスのストレージ容量。使用可能なサイズは、によって異なります AWS リージョン。</p>
ストレージサービス	該当なし	<p>このジョブに関連付けられた AWS ストレージサービス、この場合は Amazon S3。</p>

コンソール識別子	API 識別子	詳細情報
リソース	Resources	ジョブに関連付けられた AWS ストレージサービスリソース。この場合は、データ転送を行う Amazon S3 バケットです。
ジョブタイプ	JobType	ジョブのタイプ。インポート、エクスポート、またはローカルコンピューティングおよびストレージです。
Snowball type	SnowballType	このジョブで注文された Snow Family デバイスのタイプ。
クラスター ID	ClusterId	クラスターを識別する一意の 39 文字のラベル。

ジョブステータス

各 AWS Snowball Edge デバイスジョブのステータスは、ジョブの現在の状態を示すように変更されます。このジョブステータス情報は、正常性、現在の処理状態、関連デバイスで使用されているストレージは反映していません。

ジョブのステータスを確認するには

1. [AWS Snow ファミリーマネジメントコンソール](#) にログインします。
2. [Job dashboard] (ジョブダッシュボード) でジョブを選択します。
3. コンソール内のジョブ名をクリックします。
4. [ジョブステータス] ペインは上部近くにあり、ジョブのステータスを反映します。

AWS Snowball Edge デバイスジョブのステータス

コンソール識別子	API 識別子	ステータス情報
Job created	New	ジョブが作成されました。ジョブがエクスポートジョブの場合、このステータスの間のみジョブまたはジョブパートをキャンセルできます。
アプライアンスの準備	PreparingAppliance	AWS はジョブ用のデバイスを準備しています。
Exporting	InProgress	AWS は、Amazon S3 からデバイスにデータをエクスポートしています。
Preparing shipment	PreparingShipment	AWS は、デバイスの配送を準備しています。このステータスのお客様には、配送予定追跡情報が表示されます。
In transit to you	InTransitToCustomer	デバイスはジョブの作成時に指定された住所宛に配送中です。
Delivered to you	WithCustomer	デバイスがジョブ作成時に指定された住所に到着しています。

コンソール識別子	API 識別子	ステータス情報
への転送中 AWS	InTransitToAWS	デバイスを に返送しました AWS。
仕分け施設にあります	WithAWSSortingFacility	このジョブのデバイスは当社の内部仕分け施設にあります。Amazon S3 へのインポートジョブの追加プロセスが間もなく開始されます (通常は 2 日以内)。
で AWS	WithAWS	配送物が AWSに到着しました。データをインポートする場合、通常は到着日中にインポートが開始されます。
インポート中	InProgress	AWS は Amazon Simple Storage Service (Amazon S3) にデータをインポートしています。
完了	Complete	ジョブまたはジョブの一部が正常に完了しました。
キャンセル	Cancelled	ジョブがキャンセルされました。

クラスターステータス

各クラスターには、ジョブの現在の状態を示して変化するステータスがあり、クラスターの現在の全体的な進行状態を示して変化します。クラスター内の個別のノードには、独自のジョブステータスがあることに注意してください。

このクラスターステータス情報は、正常性、現在の処理状態、クラスターやそのノードで使用されているストレージは反映していません。

コンソール識別子	API 識別子	ステータス情報
Awaiting Quorum	AwaitingQuorum	クラスターのリクエストを処理するためのノードが十分にならないため、クラスターはまだ作成されていません。クラスターが作成されるには、少なくとも5つのノードが必要です。
保留中	Pending	クラスターが作成され、ノードを発送する準備をしています。各ノードのステータスはノードのジョブステータスで追跡できます。
Delivered to you	InUse	クラスターの少なくとも1つのノードが、ジョブの作成時に指定された住所に到着しています。
完了	Complete	クラスターのすべてのノードが返されました AWS。

コンソール識別子	API 識別子	ステータス情報
キャンセル	Cancelled	クラスター作成のリクエストがキャンセルされました。クラスターリクエストは、Pending 状態に入る前にのみキャンセルできます。

Amazon S3 へのジョブのインポート

インポートジョブでは、データは組み込みの Amazon S3 アダプターまたは NFS マウントポイントを使用して AWS Snowball Edge デバイスにコピーされます。インポートジョブで使用するデータソースはオンプレミスである必要があります。つまり、転送するデータを保持するストレージデバイスは、ジョブの作成時に指定されたアドレスに物理的に存在する必要があります。

ファイルをインポートすると、各ファイルは Amazon S3 のオブジェクトになり、各ディレクトリはプレフィックスになります。既存のバケットにデータをインポートした場合は、新たにインポートされたオブジェクトと同じ名前の既存のオブジェクトが上書きされます。インポートジョブタイプには、ローカルストレージやコンピューティング機能もあります。この機能は、ファイルインターフェイスまたは Amazon S3 アダプターを使用してデータの読み取りと書き込みを行い、AWS Snowball Edge デバイスでローカルに実行される Amazon S3 PUT オブジェクト API アクションに基づいて Lambda 関数をトリガーします。

すべてのデータが内の指定された Amazon S3 バケットにインポートされると AWS クラウド、AWS はデバイスを完全に消去します。この消去作業は NIST 800-88 基準に準拠しています。

インポートが完了したら、ジョブレポートをダウンロードできます。インポートプロセスに失敗したオブジェクトがある場合はこのレポートで警告されます。成功したログと失敗したログでは、追加情報を記載できます。

Important

転送データのローカルコピーは、ジョブ完了レポートの結果を検証しインポートログを確認するまで削除しないでください。

Amazon S3 からのジョブのエクスポート

Note

タグとメタデータは現在サポートされていないため、S3 バケットからオブジェクトをエクスポートすると、すべてのタグとメタデータが削除されます。

エクスポートジョブのデータソースは、1 つまたは複数の Amazon S3 バケットです。ジョブパートのデータが Amazon S3 から AWS Snowball Edge デバイスに移動されたら、ジョブレポートをダウンロードできます。デバイスへの転送が失敗したオブジェクトがあれば、このレポートで警告されます。ジョブの成功と失敗のログで詳細情報を参照できます。

各エクスポートジョブで、転送の完了に必要な数のデバイスを使用して、任意の数のオブジェクトをエクスポートできます。エクスポートジョブのジョブパートの各 AWS Snowball Edge デバイスは、次々と配信され、前のジョブパートがステータスに移行 AWS した後、後続のデバイスはユーザーに配信されます。

Amazon S3 アダプターまたは NFS マウントポイントを使用してデバイスからオンプレミスのデータ送信先にオブジェクトをコピーすると、これらのオブジェクトはファイルとして保存されます。すでにファイルのある場所にオブジェクトをコピーした場合、同じ名前の既存のファイルは上書きされます。エクスポートジョブタイプには、ローカルストレージやコンピューティング機能もあります。この機能は、ファイルインターフェイスまたは Amazon S3 アダプターを使用してデータの読み取りと書き込みを行い、AWS Snowball Edge デバイスでローカルに実行される Amazon S3 PUT オブジェクト API アクションに基づいて Lambda 関数をトリガーします。

が返されたデバイス AWS を受信すると、NIST 800-88 標準に従って完全に消去されます。

Important

Snow デバイスにエクスポートするデータは Amazon S3 に格納されている必要があります。Snow デバイスにエクスポート Amazon S3 Glacier する予定の のデータは、エクスポートする前に解凍するか、S3 ストレージクラスに移動する必要があります。これは Snow エクスポートジョブを作成する前に実行してください。

すべてのジョブのすべてのコンテンツがオンプレミスのデータ送信先にコピーされたことが確認できるまで、エクスポートされた Amazon S3 オブジェクトを変更、更新、削除しないでください。

エクスポートジョブを作成すると、Amazon S3 バケット全体または特定の範囲のオブジェクトキーをエクスポートできます。

エクスポート範囲の使用

[AWS Snow ファミリーマネジメントコンソール](#) またはジョブ管理 API でエクスポートジョブを作成する場合、Amazon S3 バケット全体または特定の範囲のオブジェクトキーをエクスポートできます。オブジェクトキー名によって、バケット内のオブジェクトは一意的に識別されます。範囲をエクスポートする場合は、範囲の開始、範囲の終了、またはその両方を指定して範囲の長さを定義します。

範囲はソートされた UTF-8 バイナリです。UTF-8 バイナリデータは次の方式でソートされます。

- 数字 0~9 は大文字と小文字の英字の前に来ます。
- 大文字の英字はすべての小文字の英字の前に来ます。
- 小文字の英字は、大文字英字と数字よりも後になります。
- 特殊文字は、他の文字セットの間にソートされます。

UTF-8 の仕様の詳細については、[Wikipedia の UTF-8](#) を参照してください。

エクスポート範囲の例

UTF-8 バイナリ順にソートされた、以下のオブジェクトとプレフィックスを含むバケットがあるとします。

- 01
- Aardvark
- Aardwolf
- Aasvogel/apple
- Aasvogel/arrow/object1
- Aasvogel/arrow/object2
- Aasvogel/banana
- Aasvogel/banker/object1
- Aasvogel/banker/object2
- Aasvogel/cherry

- Banana
- Car

指定された範囲の開始	指定された範囲の終了	エクスポートされるオブジェクトの範囲
(none)	(none)	バケット内のすべてのオブジェクト
(none)	Aasvogel	01 Aardvark Aardwolf Aasvogel/apple Aasvogel/arrow/object1 Aasvogel/arrow/object2 Aasvogel/banana Aasvogel/banker/object1 Aasvogel/banker/object2 Aasvogel/cherry
(none)	Aasvogel/banana	01 Aardvark

指定された範囲の開始	指定された範囲の終了	エクスポートされるオブジェクトの範囲
		Aardwolf Aasvogel/apple Aasvogel/arrow/object1 Aasvogel/arrow/object2 Aasvogel/banana
Aasvogel	(none)	Aasvogel/apple Aasvogel/arrow/object1 Aasvogel/arrow/object2 Aasvogel/banana Aasvogel/banker/object1 Aasvogel/banker/object2 Aasvogel/cherry Banana Car

指定された範囲の開始	指定された範囲の終了	エクスポートされるオブジェクトの範囲
Aardwolf	(none)	Aardwolf Aasvogel/apple Aasvogel/arrow/object1 Aasvogel/arrow/object2 Aasvogel/banana Aasvogel/banker/object1 Aasvogel/banker/object2 Aasvogel/cherry Banana Car

指定された範囲の開始	指定された範囲の終了	エクスポートされるオブジェクトの範囲
Aar	(none)	Aardvark Aardwolf Aasvogel/apple Aasvogel/arrow/object1 Aasvogel/arrow/object2 Aasvogel/banana Aasvogel/banker/object1 Aasvogel/banker/object2 Aasvogel/cherry Banana Car

指定された範囲の開始	指定された範囲の終了	エクスポートされるオブジェクトの範囲
car	(none)	オブジェクトは何もエクスポートされません。ジョブを作成しようとするときエラーメッセージが返されません。UTF-8 バイナリ値に従って、car は Car の下にソートされることに注意してください。
Aar	Aarr	Aardvark Aardwolf
Aasvogel/arrow	Aasvogel/arrox	Aasvogel/arrow/object1 Aasvogel/arrow/object2
Aasvogel/apple	Aasvogel/banana	Aasvogel/apple Aasvogel/arrow/object1 Aasvogel/arrow/object2 Aasvogel/banana

指定された範囲の開始	指定された範囲の終了	エクスポートされるオブジェクトの範囲
Aasvogel/apple	Aasvogel/banker	Aasvogel/apple Aasvogel/arrow/object1 Aasvogel/arrow/object2 Aasvogel/banana Aasvogel/banker/object1 Aasvogel/banker/object2
Aasvogel/apple	Aasvogel/cherry	Aasvogel/apple Aasvogel/arrow/object1 Aasvogel/arrow/object2 Aasvogel/banana Aasvogel/banker/object1 Aasvogel/banker/object2 Aasvogel/cherry

これら 3 つのバケットがあり、folder2 からすべてのオブジェクトをコピーするとします。

- s3://bucket/folder1/
- s3://bucket/folder2/
- s3://bucket/folder3/

指定された範囲の開始	指定された範囲の終了	エクスポートされるオブジェクトの範囲
folder2/	folder2/	バケット folder2 内のすべてのオブジェクト

エクスポートジョブのベストプラクティス

- データが Amazon S3 にあることを確認し、ジョブを注文する前に小さなファイルをバッチ処理します。
- バケットに何百万ものオブジェクトがある場合は、エクスポートジョブの定義にキー範囲が指定されていることを確認してください。
- データ名の末尾にスラッシュ (/ または \) の付いたオブジェクトは Snowball Edge に転送されないため、スラッシュを削除してオブジェクトキー名を更新します。
- S3 バケットの場合、オブジェクトの長さの上限は 255 文字です。
- バージョン対応の S3 バケットでは、現在のバージョンのオブジェクトのみがエクスポートされます。
- 削除マーカーはエクスポートされません。

ローカルコンピューティングおよびストレージ専用ジョブ

ローカルコンピューティングおよびストレージジョブを使用すると、インターネット接続なしで、Snow Family デバイスの Amazon S3 互換ストレージをローカルで使用できます。デバイスが返されたときに、Amazon S3 からデバイスにデータをエクスポートしたり Amazon S3 にデータをインポートしたりすることはできません。

トピック

- [ローカルストレージジョブ](#)
- [ローカルクラスターオプション](#)

ローカルストレージジョブ

Snow Family AWS Snowball Edge デバイスの Amazon S3 互換ストレージまたは S3 アダプターを使用して、デバイスにオブジェクトを読み書きできます。デバイスの注文時に S3 アダプターを使用する場合は、受信時にデバイスに含める Amazon S3 バケットも選択します。Snow Family デバイスで Amazon S3 互換ストレージを使用することを選択した場合、受信時に Amazon S3 バケットはデバイスに含まれません。

Snowball Edge デバイスで Amazon S3 バケットを作成して、ローカルデータアクセス、ローカルデータ処理、データレジデンシーを必要とするアプリケーション用に、オンプレミスのオブジェクトを保存および取得できます。Snow Family デバイスの Amazon S3 互換ストレージは、SNOW という新しいストレージクラスを提供します。このストレージクラスは、S3 API を使用し、複数の Snowball Edge デバイス間でデータを永続的かつ冗長的に保存するように設計されています。Snowball Edge バケットで、Amazon S3 と同じ API と機能 (バケットのライフサイクルポリシー、暗号化、タグ付けなど) を使用できます。デバイスまたはデバイスが返されると AWS、Snow Family デバイスの Amazon S3 互換ストレージに作成または保存されたすべてのデータが消去されます。詳細については、[「ローカルコンピューティングとストレージ専用ジョブ」](#)を参照してください。

詳細については、本ガイドの「[Snow Family デバイスの Amazon S3 互換ストレージ](#)」を参照してください。

デバイスの使用が終了したら、に返すと AWS、デバイスは消去されます。この消去作業は National Institute of Standards and Technology (NIST) 800-88 基準に準拠しています。

ローカルクラスターオプション

クラスターは、Snowball Edge デバイスの論理グループで、3～16 台のデバイスで構成されています。クラスターは 1 つのジョブとして作成され、他の AWS Snowball ジョブサービスと比較して耐久性とストレージサイズが向上します。クラスタージョブの詳細については、本ガイドの「[クラスターリングの概要](#)」を参照してください。

コンソールでのジョブのクローン

インポートジョブまたはローカルコンピューティングおよびストレージジョブを初めて作成すると、複数の AWS Snowball Edge デバイスが必要であることがわかります。インポートジョブやローカルコンピューティングおよびストレージジョブは 1 つのデバイスに関連付けられるため、複数のデバイスが必要な場合は、複数のジョブを作成する必要があります。追加ジョブの作成時には、コンソールで再度ジョブの作成ウィザードを実行するか、既存のジョブをクローンできます。

Note

ジョブのクローンはコンソールで使用できるショートカットで、追加ジョブを簡単に作成できます。ジョブ管理 API を使用してジョブを作成する場合は、ジョブ作成コマンドを再度実行するだけです。

ジョブのクローン作成とは、自動的に変更される名前を除いて正確に再作成することを意味します。クローンのプロセスは簡単です。

コンソールでジョブのクローンを作成するには

1. で AWS Snow ファミリーマネジメントコンソール、テーブルからジョブを選択します。
2. [Actions] で、[Clone job] を選択します。

[Create job] ウィザードの最終ページ [Step 6: Review] が開きます。

3. 情報を確認し、該当する [Edit] ボタンを選択して変更を加えます。
4. クローンされたジョブを作成するには、[Create job] を選択します。

クローンされたジョブは **####-clone-##** 形式で命名されます。番号は自動的にジョブ名に追加され、このジョブのクローンを最初に作成した後、クローンを作成した回数を表示します。例えば、AprilFinanceReports-clone はジョブの最初のクローンジョブを表し AprilFinanceReports、DataCenterMigration-clone-42 は DataCenterMigration ジョブの 40 秒のクローンを表します。

Snowball Edge デバイスの使用に関するベストプラクティス

AWS Snowball Edge デバイスの利点と満足度を最大限に高めるために、以下のベストプラクティスに従うことをお勧めします。

セキュリティ

以下は、AWS Snowball Edge デバイスの使用中にセキュリティを維持するための推奨事項とベストプラクティスです。

一般的なセキュリティ

- AWS Snowball Edge デバイスについて疑わしい点に気付いた場合は、内部ネットワークに接続しないでください。[AWS Support](#) にお問い合わせいただければ、新しい AWS Snowball Edge をお客様宛に配送いたします。
- 解除キーのコピーは、ワークステーション上のそのジョブ用のマニフェストと同じ場所に保存しないことをお勧めします。これらを別の場所に保存すると、権限のないユーザーが AWS Snowball Edge デバイスにアクセスできなくなります。たとえば、マニフェストのコピーはローカルサーバーに保存し、コードはメールでデバイスをロック解除するユーザーに送信することができます。この方法では、AWS Snowball Edge デバイスへのアクセスを、サーバーに保存されているファイルとユーザーの E メールアドレスにアクセスできる個人に制限します。
- Snowball Edge クライアントコマンド `list-access-keys` および `unlock` を実行するときに表示される認証情報は `get-secret-access-key`、デバイスへのアクセスに使用されるアクセスキーのペアです。

これらのキーは、デバイスのジョブおよびローカルリソースにのみ関連付けられます。これらは AWS アカウント や他の AWS アカウントにはマッピングされません。これらのキーを使用してのサービスとリソースにアクセスしようとする AWS クラウド、ジョブに関連付けられたローカルリソースでのみ機能するため失敗します。

- 認証情報を紛失した、または認証情報が侵害されたと思われる場合は、デバイスの SSL 証明書を更新するプロセスに従って、新しいマニフェストファイルとロック解除コードをリクエストしてください。[SSL 証明書を更新する](#) を参照してください。

AWS Identity and Access Management (IAM) ポリシーを使用してアクセスを制御する方法については、「[AWS Snowball Edge の管理 \(事前定義\) ポリシー](#)」を参照してください。

ネットワークセキュリティ

- デバイス上のローカルバケットへのデータの読み取りと書き込みには、一度に 1 つの方法のみを使用することをお勧めします AWS Snowball Edge。ファイルインターフェイスと Amazon S3 アダプターの両方を同じバケットで同時に使用すると、読み取り/書き込みの競合が発生する可能性があります。
- データの破損を防ぐため、データ転送中に AWS Snowball Edge デバイスを切断したり、ネットワーク設定を変更したりしないでください。
- デバイスへの書き込み中、ファイルは静的な状態になっている必要があります。書き込み中にファイルが変更されると、読み取り/書き込み競合が発生する可能性があります。
- AWS Snowball Edge デバイスのパフォーマンス向上の詳細については、「」を参照してください [パフォーマンス](#)。

リソース管理

AWS Snowball Edge デバイスでジョブとリソースを管理するには、次のベストプラクティスを考慮してください。

- オンプレミスデータ転送を実行する 10 日間の無料期間は、AWS Snowball Edge デバイスがデータセンターに到着した翌日から始まります。これは、すべての Snowball Edge デバイスタイプに適用されます。
- [Job created] ステータスは、ジョブをキャンセルできる唯一のステータスです。ジョブが別のステータスに変わったら、ジョブをキャンセルすることはできません。これはクラスターに適用されます。
- ジョブのインポートでは、プロセスの最終段階で Amazon S3 へのインポートが成功するまで、転送データのローカルコピーを削除しないでください。プロセスの一部として、データ転送の結果を確認するようにしてください。

パフォーマンス

Note

データ転送のパフォーマンスは、ネットワーク環境、オペレーティングシステム、コピー方法、プロトコル、ソースデータの読み取りパフォーマンス、ファイルサイズなどのデータセットの特性によって異なります。正確なデータ転送速度とデータ転送時間を決定するに

は、環境での proof-of-concept テストを通じてパフォーマンスを測定することをお勧めします。

以下に、AWS Snowball Edge デバイスのパフォーマンスに関する推奨事項と情報を示します。ここでは、一般的なパフォーマンスについて説明します。オンプレミス環境は、ネットワークテクノロジー、ハードウェア、オペレーティングシステム、手順など、異なるさまざまな方法で動作しているためです。

次の表は、ネットワークの転送速度によって Snowball Edge デバイスにデータが収まるまでの時間がどのように影響を受けるかを概説しています。小さいファイルを転送すると、オーバーヘッドが増加するため転送速度が落ちます。小さいファイルが多数ある場合は、大きなアーカイブに圧縮してから、Snowball Edge デバイスに転送することをお勧めします。

レート (MB/秒)	82 TB の転送時間
800	1.22 日間
450	2.11 日間
400	2.37 日間
300	3.16 日間
277	3.42 日間
200	4.75 日間
100	9.49 日間
60	15.53 日間
30	31.06 日間
10	85.42 日間

パフォーマンスに関する有意義なガイダンスを提供するために、以下のセクションでは、AWS Snowball Edge デバイスを使用するタイミングを決定する方法と、サービスを最大限に活用する方法について説明します。

トピック

- [パフォーマンスに関する推奨事項](#)
- [データ転送の高速化](#)

パフォーマンスに関する推奨事項

次の推奨事項は、データ転送のパフォーマンス向上に大きな影響を与えるため、強くお勧めします。

- 各ディレクトリ内のファイルまたはディレクトリの数が 500,000 を超えないようにすることをお勧めします。
- Snowball Edge デバイスに転送するすべてのファイルは、1 MB より小さいサイズにすることをお勧めします。
- サイズが 1 MB より小さいファイルが多数ある場合は、大きなアーカイブに圧縮してから Snowball Edge デバイスに転送することをお勧めします。

データ転送の高速化

デバイスのパフォーマンスを向上させる最善の方法の 1 つは、AWS Snowball Edge デバイスとの間で送受信されるデータ転送を高速化することです。一般的に、データソースからデバイスへの転送速度は、次の方法で向上します。次のリストは、パフォーマンスへのプラスの影響が大きいものから小さいものの順に並べられています。

1. 一度に複数の書き込みオペレーションを実行する - これを行うには、単一の AWS Snowball Edge デバイスにネットワーク接続されたコンピュータの複数のターミナルウィンドウから、各コマンドを実行します。
2. 小容量ファイルのバッチ転送 - コピー操作によっては、暗号化によりオーバーヘッドが生じる場合があります。このプロセスを高速化するには、ファイルをバッチ処理して単一のアーカイブにまとめます。ファイルをバッチ処理してまとめると、Amazon S3 にインポートしたときに自動抽出することができます。詳細については、「[小さなファイルのバッチ処理](#)」を参照してください。
3. 転送中にファイル上で他のオペレーションを実行しない - 転送時のファイル名の変更、ファイルメタデータの変更、コピーオペレーション時のファイルへのデータ書き込みは、転送パフォーマンスに悪影響を及ぼします。転送時、ファイルは静的な状態にされることをお勧めします。
4. ローカルネットワークの使用を減らす - AWS Snowball Edge デバイスは、ローカルネットワークを介して通信します。したがって、AWS Snowball Edge デバイス、接続されているスイッチ、お

よびデータソースをホストしているコンピュータ間のローカルネットワークトラフィックを減らすことで、データ転送速度を向上させることができます。

5. 不要なホップを排除する – AWS Snowball Edge デバイス、データソース、およびそれらの間でターミナル接続を実行しているコンピュータをセットアップして、単一のスイッチ間で通信する唯一のマシンになるようにすることをお勧めします。これにより、データ転送速度が向上します。

Snowball Edge デバイスのソフトウェア更新

AWS Snow Family デバイスで新しいソフトウェアが使用可能になると、 から通知されます。通知は、E メールで AWS Health Dashboard、イベントとして提供されます CloudWatch。E メール通知は、Amazon Web Services, Inc. から、Snow Family デバイスの注文に使用した AWS アカウントにアタッチされた E メールアドレスに送信されます。通知を受け取ったら、このトピックの手順に従って、デバイスの使用が中断されないように、できるだけ早く更新をダウンロードしてインストールします。の詳細については AWS Health Dashboard、「[AWS Health ユーザーガイド](#)」を参照してください。CloudWatch イベントの詳細については、「[Amazon CloudWatch Events ユーザーガイド](#)」を参照してください。

からソフトウェアアップデートをダウンロードし AWS、オンプレミス環境の Snowball Edge デバイスにインストールできます。これらの更新はバックグラウンドで行われます。最新のソフトウェアが からデバイスに安全にダウンロードされている間も、通常どおり AWS デバイスを引き続き使用できます。ただし、ダウンロードした更新を適用するには、デバイスのワークロードを停止して再起動する必要があります。

Snowball Edge/Snowcone デバイス (アプライアンス) AWS 用に が提供するソフトウェア更新は、サービス条件のセクション 9 に基づくアプライアンスソフトウェアです。

ソフトウェア更新は、AWSに代わって該当するアプライアンスにソフトウェア更新をインストールすることのみを目的として提供されます。お客様は、(i) 当該アプライアンスへのソフトウェア更新のインストールに必要なもの以外のソフトウェア更新のコピーを作成すること、または (ii) ソフトウェア更新に含まれる機能または手段 (ソフトウェア更新に適用される暗号化を含むがこれに限定されない) を回避または無効化することを行わず (または試みない)、サードパーティがこれを行うことを許可または承認しない (または試みない) ものとし、ソフトウェア更新が当該アプライアンスにインストールされたら、アプライアンスへのソフトウェア更新のインストールに使用されたすべてのメディアからソフトウェア更新を削除することに同意するものとし、

Warning

更新をインストールする前に、デバイスのすべてのアクティビティを一時停止することを強くお勧めします。デバイスを更新して再起動すると、インスタンスの実行が停止し、ローカル Amazon S3 バケットへの書き込みが中断されます。

トピック

- [前提条件](#)
- [更新をダウンロードする](#)
- [更新をインストールする](#)
- [SSL 証明書を更新する](#)
- [Snow Family デバイスの Amazon Linux 2 AMI を更新する](#)

前提条件

デバイスを更新する前に、次の前提条件を満たしている必要があります。

- ジョブが作成され、オンプレミスのデバイスがあり、そのロックが解除されていること。詳細については、「[開始方法](#)」を参照してください。
- Snowball Edge デバイスの更新は、Snowball Edge クライアントを使って行います。Snowball Edge クライアントの最新バージョンをダウンロードして、更新するデバイスへのネットワーク接続があるローカル環境のコンピュータにインストールする必要があります。詳細については、「[Snowball Edge クライアントの使用](#)」を参照してください。
- (オプション) Snowball Edge クライアントのプロファイルを設定することをお勧めします。詳細については、「[Snowball Edge クライアント用プロファイルの設定](#)」を参照してください。
- クラスター化された Snowball Edge デバイス上の Snow Family デバイスの Amazon S3 互換ストレージの場合は、S3-Snow サービスを停止し、自動起動を無効にします。[Snow Family デバイスの Amazon S3 互換ストレージサービスを自動起動するように設定するサービス](#) を参照してください。

Note

クラスター化されたデバイスの場合、すべてのコマンドをデバイスごとに実行する必要があります。

上記のタスクが完了すると、Snowball Edge デバイスの更新をダウンロードしてインストールできるようになります。

更新をダウンロードする

Snow Family デバイスの更新をダウンロードするには、主に 2 つの方法があります。

- 特定の Snowball Edge クライアントのコマンドを使用して、随時手動で更新をトリガーできます。
- デバイスを自動的に更新する時間をプログラムで決定できます。

以下の処理は、手動で更新をダウンロードするプロセスを説明しています。Snowball Edge デバイスの自動更新については、「Snowball Edge configure-auto-update-strategy [の更新](#)」の「」を参照してください。

Note

デバイスがインターネットにアクセスできない場合は、[GetSoftwareUpdates](#) API を使用して更新ファイルをダウンロードできます。次に、次の例のように、uriパラメータdownload-updatesを使用して を呼び出すときに、ローカルファイルの場所をポイントします。

```
snowballEdge download-updates --uri file:///tmp/local-update
```

Windows オペレーティングシステムの場合、uriパラメータの値を次のようにフォーマットします。

```
snowballEdge download-updates --uri file:/C:/path/to/local-update
```

スタンドアロンデバイスの Snowball Edge ソフトウェア更新を確認してダウンロードするには

1. ターミナルウィンドウを開き、Snowball Edge デバイスが describe-device コマンドでロック解除されていることを確認します。デバイスがロックされている場合は、unlock-device コマンドを使用してロック解除します。詳細については、「[Snow Family デバイスのロック解除](#)」を参照してください。
2. デバイスのロックが解除されている場合は、snowballEdge check-for-updates コマンドを実行します。このコマンドは、Snowball Edge ソフトウェアの利用可能な最新バージョン、およびデバイスにインストールされている現在のバージョンも返します。
3. デバイスのソフトウェアが最新のものではない場合、snowballEdge download-updates コマンドを実行します。

Note

デバイスがインターネットに接続されていない場合は、まず [GetSoftwareUpdates](#) API を使用して更新ファイルをダウンロードします。次に、次の例のように、ダウンロードしたファイルへのローカルパスを持つ uri パラメータを使用して snowballEdge download-updates コマンドを実行します。

```
snowballEdge download-updates --uri file:///tmp/local-update
```

Windows オペレーティングシステムの場合、uri パラメータの値を次のようにフォーマットします。

```
snowballEdge download-updates --uri file://C:/path/to/local-update
```

4. このダウンロードのステータスは、snowballEdge describe-device-software コマンドで確認できます。更新のダウンロード中は、このコマンドでステータスを表示します。

Example describe-device-software コマンドの出力

```
Install State: Downloading
```

デバイスのクラスターの Snowball Edge ソフトウェア更新を確認してダウンロードするには

1. ターミナルウィンドウを開き、snowballEdge describe-device コマンドを使用して、クラスター内のすべての Snowball Edge デバイスのロックが解除されていることを確認します。デバイスがロックされている場合は、snowballEdge unlock-cluster コマンドを使用してロックを解除します。詳細については、[「Snowball Edge のロック解除」](#)を参照してください。
2. クラスター内のすべてのデバイスがロック解除されたら、クラスター内のデバイスごとに check-for-updates コマンドを実行します。このコマンドは、Snowball Edge ソフトウェアの利用可能な最新バージョン、およびデバイスにインストールされている現在のバージョンも返します。

```
snowballEdge check-for-updates --unlock-code 29-character-unlock-code --manifest-file path/to/manifest/file.bin --endpoint https://ip-address-of-snow-device
```

Note

ロック解除コードとマニフェストファイルは、クラスター内のすべてのデバイスで同じです。

Example `check-for-updates` コマンドの

```
{  
  "InstalledVersion" : "118",  
  "LatestVersion" : "119"  
}
```

LatestVersion 名前の値がInstalledVersion名前の値より大きい場合、更新を使用できません。

3. クラスター内のデバイスごとに、`download-updates` コマンドを使用して更新をダウンロードします。

```
snowballEdge download-updates --uri file:///tmp/local-update
```

Note

Windows オペレーティングシステムの場合、`uri`パラメータの値を次のようにフォーマットします。

```
snowballEdge download-updates --uri file://C:/path/to/local-update
```

4. クラスター内のデバイスごとにこのダウンロードのステータスを確認するには、`describe-device-software` コマンドを使用します。

```
snowballEdge describe-device-software --unlock-code 29-character-unlock-code --manifest-file path/to/manifest/file.bin --endpoint https://ip-address-of-snow-device
```

Example **describe-device-software** コマンドの出力の

```
{
  "InstalledVersion" : "118",
  "InstallingVersion" : "119",
  "InstallState" : "DOWNLOADED",
  "CertificateExpiry" : "Sat Mar 30 16:47:51 UTC 2024"
}
```

InstallState 名前の値が の場合DOWNLOADED、更新はダウンロードされ、インストールできるようになります。

更新をインストールする

更新をダウンロードしたら、インストールして、更新が有効になるようにデバイスを再起動する必要があります。次の手順では、更新を手動でインストールする方法を説明します。

Snowball Edge デバイスのクラスターの場合、クラスター内の各デバイスに更新をダウンロードしてインストールする必要があります。

Note

ソフトウェア更新をインストールする前に、デバイス上のすべてのアクティビティを中断してください。更新をインストールすると、実行中のインスタンスが停止し、デバイス上の Amazon S3 バケットへの書き込みが中断されます。これにより、データが失われる可能性があります。

スタンドアロン Snow Family デバイスに既にダウンロードされているソフトウェア更新をインストールするには

1. ターミナルウィンドウを開き、Snowball Edge デバイスが `describe-device` コマンドでロック解除されていることを確認します。デバイスがロックされている場合は、`unlock-device` コマンドを使用してロック解除します。詳細については、[「Snowball Edge のロック解除」](#)を参照してください。
2. `list-services` コマンドを実行して、デバイスで利用可能なサービスを確認します。コマンドは、デバイスで使用可能な各サービスのサービス IDs を返します。

```
snowballEdge list-services
```

Example **list-services** コマンドの出力の

```
{
  "ServiceIds" : [ "greengrass", "fileinterface", "s3", "ec2", "s3-snow" ]
}
```

3. `list-services` コマンドで識別される各サービス ID について、`describe-service` コマンドを実行してステータスを確認します。この情報を使用して、停止するサービスを識別します。

```
snowballEdge describe-service --service-id service-id
```

Example **describe-service** コマンドの出力の

```
{
  "ServiceId" : "s3",
  "Status" : {
    "State" : "ACTIVE"
  },
  "Storage" : {
    "TotalSpaceBytes" : 99608745492480,
    "FreeSpaceBytes" : 99608744468480
  }
}
```



```
},
"Endpoints" : [ {
  "Protocol" : "http",
  "Port" : 8080,
  "Host" : "192.0.2.0"
}, {
  "Protocol" : "https",
  "Port" : 8443,
  "Host" : "192.0.2.0",
  "CertificateAssociation" : {
    "CertificateArn" : "arn:aws:snowball-
device::certificate/6d955EXAMPLEdb71798146EXAMPLE3f0"
  }
} ]
}
```

この出力は、s3サービスがアクティブであり、stop-service コマンドを使用して停止する必要があることを示しています。

4. stop-service コマンドを使用して、State名前の値がlist-servicesコマンドの出力ACTIVEにある各サービスを停止します。複数のサービスが実行されている場合は、続行する前に各サービスを停止してください。

Note

Amazon S3 アダプター、Amazon EC2 AWS STS、IAM サービスは停止できません。Snow Family デバイスの Amazon S3 互換ストレージが稼働している場合は、更新をインストールする前に停止してください。Snow Family デバイスの Amazon S3 互換ストレージには、s3-snow として がありますserviceId。

```
snowballEdge stop-service --service-id service-id --device-ip-addresses snow-
device-1-ip-address snow-device-device-2-ip-address snow-device-3-ip-address --
manifest-file path/to/manifest/file.bin --unlock-code 29-character-unlock-code --
endpoint https://snow-device-ip-address
```

Example **stop-service** コマンドの出力の

Stopping the AWS service on your Snowball Edge. You can determine the status of the AWS service using the describe-service command.

5. snowballEdge install-updates コマンドを実行します。
6. このインストールのステータスは、snowballEdge describe-device-software コマンドで確認できます。更新のインストール中は、このコマンドでステータスを表示します。

出力例

```
Install State: Installing //Possible values[NA, Installing, Requires Reboot]
```

Snowball Edge デバイスのソフトウェア更新が正しくインストールされました。更新のインストールは、この更新が自動的にデバイスに適用されることではありません。更新のインストールを完了するには、デバイスを再起動する必要があります。

Warning

デバイス上のすべてのアクティビティを停止せずに Snow Family デバイスを再起動すると、データが失われる可能性があります。

7. デバイス上のすべてのサービスが停止したら、デバイスを再起動し、デバイスのロックを解除して、再度再起動します。これにより、ダウンロードしたソフトウェア更新のインストールが完了します。デバイスを再起動する方法の詳細については、[「Snow Family デバイスを再起動する」](#)を参照してください。デバイスのロック解除の詳細については、[「Snowball Edge のロック解除 Snow」](#)を参照してください。
8. 2 回目の再起動後にデバイスの電源がオンになったら、デバイスのロックを解除します。
9. check-for-updates コマンドを実行します。このコマンドは、Snowball Edge ソフトウェアの利用可能な最新バージョン、およびデバイスにインストールされている現在のバージョンも返します。

Snowball Edge デバイスのクラスターに既にダウンロードされているソフトウェア更新をインストールするには

1. クラスター内のデバイスごとに、`describe-device` コマンドを実行して、デバイスがロック解除されているかどうかを確認します。デバイスがロックされている場合は、`unlock-cluster` コマンドを使用してロックを解除します。詳細については、[「Snowball Edge のロック解除」](#)を参照してください。
2. クラスター内のデバイスごとに、`list-services` コマンドを実行して、デバイスで利用可能なサービスを確認します。コマンドは、デバイスで使用可能な各サービスのサービス IDs を返します。

```
snowballEdge list-services
```

Example **list-services** コマンドの出力の

```
{
  "ServiceIds" : [ "greengrass", "fileinterface", "s3", "ec2", "s3-snow" ]
}
```

3. `list-services` コマンドで識別される各サービス ID について、`describe-service` コマンドを実行してステータスを確認します。この情報を使用して、停止するサービスを識別します。

```
snowballEdge describe-service --service-id service-id
```

Example **describe-service** コマンドの出力の

```
{
  "ServiceId" : "s3",
  "Status" : {
    "State" : "ACTIVE"
  },
  "Storage" : {
    "TotalSpaceBytes" : 99608745492480,
```

```
"FreeSpaceBytes" : 99608744468480
},
"Endpoints" : [ {
  "Protocol" : "http",
  "Port" : 8080,
  "Host" : "192.0.2.0"
}, {
  "Protocol" : "https",
  "Port" : 8443,
  "Host" : "192.0.2.0",
  "CertificateAssociation" : {
    "CertificateArn" : "arn:aws:snowball-
device:::certificate/6d955EXAMPLEdb71798146EXAMPLE3f0"
  }
} ]
}
```

この出力は、s3サービスがアクティブであり、`stop-service` コマンドを使用して停止する必要があることを示しています。

4. クラスター内のデバイスごとに、`stop-service` コマンドを使用して、State名前の値が`list-services`コマンドの出力ACTIVEにある各サービスを停止します。複数のサービスが実行されている場合は、続行する前に各サービスを停止してください。

Note

Amazon S3 アダプター、Amazon EC2 AWS STS、IAM サービスは停止できません。Snow Family デバイスの Amazon S3 互換ストレージが稼働している場合は、更新をインストールする前に停止してください。Snow Family デバイスの Amazon S3 互換ストレージには、`s3-snow` として `serviceId` がありません。

```
snowballEdge stop-service --service-id service-id --device-ip-addresses snow-
device-1-ip-address snow-device-device-2-ip-address snow-device-3-ip-address --
manifest-file path/to/manifest/file.bin --unlock-code 29-character-unlock-code --
endpoint https://snow-device-ip-address
```

Example **stop-service** コマンドの出力の

```
Stopping the AWS service on your Snowball Edge. You can determine the status of the AWS service using the describe-service command.
```

5. クラスター内のデバイスごとに、`install-updates` コマンドを実行します。

```
snowballEdge install-updates
```

6. このインストールのステータスは、`describe-device-software` コマンドで確認できます。

```
snowballEdge describe-device-software
```

Example **describe-device-service** コマンドの出力の

```
Install State: Installing //Possible values[NA, Installing, Requires Reboot]
```

Install State が `Requires Reboot` の場合、Snowball Edge デバイスのソフトウェア更新が正常にインストールされました。更新のインストールは、この更新が自動的にデバイスに適用されることはありません。更新のインストールを完了するには、デバイスを再起動する必要があります。

Warning

デバイス上のすべてのアクティビティを停止せずに Snowball Edge デバイスを再起動すると、データが失われる可能性があります。

7. クラスター内のすべてのデバイスを再起動し、クラスターのロックを解除して、クラスター内のすべてのデバイスを再起動します。これにより、ダウンロードしたソフトウェア更新のインストールが完了します。デバイスを再起動する方法の詳細については、「[Snow Family デバイスを再起動する](#)」を参照してください。デバイスのクラスターのロック解除の詳細については、「[Snowball Edge のロック解除](#)」を参照してください。

8. クラスター内の各デバイスが 2 回再起動されたら、クラスターのロックを解除し、`check-for-updates` コマンドを使用してデバイスが更新されたことを確認します。このコマンドは、Snowball Edge ソフトウェアの利用可能な最新バージョン、およびデバイスにインストールされている現在のバージョンも返します。現在のバージョンと利用可能な最新バージョンが同じ場合、デバイスは正常に更新されました。

これで、Snow Family デバイスまたはデバイスのクラスターが正常に更新され、最新の Snow Family ソフトウェアへの更新が確認されました。

SSL 証明書を更新する

Snow Family デバイスを 360 日以上保持する場合は、デバイスの使用が中断されないように、デバイスの Secure Sockets Layer (SSL) 証明書を更新する必要があります。証明書の有効期限が切れると、デバイスを使用できなくなり、AWSに返却する必要があります。

AWS は、お持ちの Snow Family デバイスの SSL 証明書の有効期限が切れる 30 日前に通知します。通知は、E メールで AWS Health Dashboard、イベントとして提供されます CloudWatch。E メール通知は、Amazon Web Services, Inc. から、Snow Family デバイスの注文に使用した AWS アカウントにアタッチされた E メールアドレスに送信されます。通知を受け取ったら、このトピックの指示に従って、デバイスの使用が中断されないように、できるだけ早く更新をリクエストしてください。の詳細については AWS Health Dashboard、「[AWS Health ユーザーガイド](#)」を参照してください。CloudWatch イベントの詳細については、「[Amazon CloudWatch Events ユーザーガイド](#)」を参照してください。

SSL 証明書の更新は、Snowball Edge クライアントを介して行われます。Snowball Edge クライアントの最新バージョンをダウンロードして、更新するデバイスへのネットワーク接続があるローカル環境のコンピュータにインストールする必要があります。詳細については、「[Snowball Edge Client の使用](#)」を参照してください。

このトピックでは、証明書の有効期限を確認する方法と、デバイスを更新する方法について説明します。

1. 証明書の有効期限を特定するには `snowballEdge describe-device-software` コマンドを使用します。コマンドの出力では、`CertificateExpiry` の値に証明書の有効期限の日時が含まれます。

Example `describe-device-software` の出力

```
Installed version: 101
Installing version: 102
Install State: Downloading
CertificateExpiry : Thur Jan 01 00:00:00 UTC 1970
```

2. に連絡して AWS Support、SSL 証明書の更新をリクエストします。
3. AWS Support は更新ファイルを提供します。更新ファイルを [ダウンロード](#) して [インストール](#) します。
4. [Snowball Edge をロック解除するとき、新しいロック解除コードとマニフェストファイル](#)。

Snow Family デバイスの Amazon Linux 2 AMI を更新する

セキュリティのベストプラクティスとして、Amazon Linux 2 AMIs up-to-date Snow Family デバイスに保持します。の更新 AWS Marketplace については、の [Amazon Linux 2 AMI \(HVM\)、SSD ボリュームタイプ \(64 ビット x86\)](#) を定期的を確認してください。AMI を更新する必要があると判断したら、最新の Amazon Linux 2 イメージを Snow デバイスにインポートします。「[Amazon EC2 互換 AMI としてデバイスにイメージをインポートする](#)」を参照してください。

AWS CLI で `ssm get-parameters` コマンドを使用して最新の Amazon Linux 2 イメージ ID を取得することもできます。

```
aws ssm get-parameters --names /aws/service/ami-amazon-linux-latest/amzn2-ami-hvm-x86_64-gp2 --query 'Parameters[0].[Value]' --region your-region
```

このコマンドは、AMI の最新のイメージ ID を返します。例:

```
ami-0ccb473bada910e74
```

AWS Snowball Edge のセキュリティ

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からメリットを得られます。

セキュリティは、AWS とユーザーの間で共有される責任です。[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して説明しています。

- クラウドのセキュリティ – AWS は、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責任を担います AWS クラウド。また、は、お客様が安全に使用できるサービス AWS も提供します。[AWS コンプライアンスプログラム](#)の一環として、サードパーティーの監査が定期的にセキュリティの有効性をテストおよび検証しています。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 AWS Snowball、「[コンプライアンスAWS プログラムによる対象範囲内のサービス](#)」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、の使用時に責任共有モデルを適用する方法を理解するのに役立ちます AWS Snowball。以下のトピックでは、セキュリティおよびコンプライアンスの目的 AWS Snowball を達成するためにを設定する方法を示します。また、AWS Snowball リソースのモニタリングや保護に役立つ他の AWS のサービスの使用方法についても説明します。

トピック

- [AWS Snowball Edge でのデータ保護](#)
- [での Identity and Access Management AWS Snowball](#)
- [AWS Snowballでのログ記録とモニタリング](#)
- [のコンプライアンス検証 AWS Snowball](#)
- [耐障害性](#)
- [のインフラストラクチャセキュリティ AWS Snowball](#)

AWS Snowball Edge でのデータ保護

AWS Snowball は、データ保護に関する規制とガイドラインを含む AWS [責任共有モデル](#) に準拠しています。は、すべての AWS サービス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 AWS 責任を負います。AWS は、顧客コンテンツや個人データを処理するためのセキュリティ設定コントロールなど、このインフラストラクチャでホストされるデータの制御を維持します。AWS のお客様および APN パートナーは、データ管理者またはデータ処理者として、に格納される個人データに対して責任を負います AWS クラウド。

データ保護の目的で、AWS アカウント 認証情報を保護し、AWS Identity and Access Management (IAM) を使用して個々のユーザーをセットアップして、各ユーザーに各自の職務を果たすために必要なアクセス許可のみを付与することをお勧めします。また、次の方法でデータを保護することもお勧めします:

- 各アカウントで多要素認証 (MFA) を使用します。
- SSL/TLS を使用して AWS リソースと通信します。TLS 1.2 以降が推奨されます。
- 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。

顧客のアカウント番号などの機密の識別情報は、[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、AWS Snowball または AWS SDK を使用して AWS CLI または他の のサービスを使用する場合も同様です。AWS SDKs AWS Snowball または他のサービスに入力したデータはすべて、診断ログの内容として取得される可能性があります。外部サーバーへの URL を指定するときは、そのサーバーへのリクエストを検証するための認証情報を URL に含めないでください。

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

トピック

- [クラウドでのデータの保護](#)
- [デバイスのデータの保護](#)

クラウドでのデータの保護

AWS Snowball は、Amazon S3 にデータをインポートまたはエクスポートするとき、Snow Family デバイスを注文するジョブを作成するとき、およびデバイスが更新されたときにデータを保護します。以下のセクションでは、Snowball Edge を使用してオンラインまたはクラウド AWS でとやり取りするときデータを保護する方法について説明します。

トピック

- [AWS Snowball Edge の暗号化](#)
- [AWS Key Management ServiceAWS Snowball Edge の](#)

AWS Snowball Edge の暗号化

Snowball Edge を使用してデータを S3 にインポートする際、デバイスに転送されるすべてのデータは、ネットワーク経由で SSL 暗号化によって保護されます。AWS Snowball Edge では、保管中のデータを保護するため、サーバー側の暗号化 (SSE) が使用されます。

AWS Snowball Edge でのサーバー側の暗号化

AWS Snowball Edge は、Amazon S3 が管理する暗号化キー (SSE-S3) によるサーバー側の暗号化をサポートしています。サーバー側の暗号化は、保管中のデータを保護するものです。SSE-S3 では、Amazon S3 に保管中のデータを保護するための強力な多要素暗号化を採用しています。SSE-S3 の詳細については、Amazon Simple Storage Service ユーザーガイドの [Amazon S3 が管理する暗号化キーによるサーバー側の暗号化 \(SSE-S3\) を使用したデータの保護](#) を参照してください。

現在、AWS Snowball Edge は、お客様が用意したキーによるサーバー側の暗号化 (SSE-C) を提供していません。Snow Family デバイスの Amazon S3 互換ストレージは、ローカルコンピューティングおよびストレージジョブ用の SSE-C を提供します。ただし、その SSE タイプを使用して、インポートされたデータを保護したい場合や、エクスポートするデータにすでにこのタイプを使用している場合があります。このような場合は、以下の点に注意してください。

- インポート –

S3 にインポートしたオブジェクトを SSE-C で暗号化する場合は、代わりに、そのバケットのバケットポリシーの一部として確立されている SSE-KMS または SSE-S3 暗号化を使用することを

検討してください。ただし、Amazon S3 にインポートしたオブジェクトを SSE-C で暗号化する必要がある場合は、バケット内のオブジェクトをコピーして SSE-C で暗号化する必要があります。これを実現する CLI コマンドの例を以下に示します。

```
aws s3 cp s3://mybucket/object.txt s3://mybucket/object.txt --sse-c --sse-c-key 1234567891SAMPLEKEY
```

または

```
aws s3 cp s3://mybucket s3://mybucket --sse-c --sse-c-key 1234567891SAMPLEKEY --recursive
```

- エクスポート - SSE-C で暗号化されたオブジェクトをエクスポートする場合は、最初にそれらのオブジェクトを別のバケットにコピーします。コピー先のバケットでは、バケットポリシーにサーバー側の暗号化が指定されていないか、SSE-KMS または SSE-S3 が指定されているものとしません。

Snowball Edge から Amazon S3 内にインポートしたデータに対する SSE-S3 の有効化

Amazon S3 にインポートするデータに対して SSE-S3 を有効にするには、Amazon S3 マネジメントコンソールで次の手順を使用します。AWS Snow ファミリーマネジメントコンソールまたは Snowball デバイス自体の設定は必要ありません。

Amazon S3 内にインポートするデータに対して SSE-S3 暗号化を有効にするには、データのインポート先となるすべてのバケットに対してバケットポリシーを設定するだけです。アップロードリクエストに `s3:PutObject` ヘッダーが含まれていない場合は、ポリシーを更新してオブジェクトのアップロード (`x-amz-server-side-encryption`) 許可を拒否します。

Amazon S3 にインポートしたデータに対して SSE-S3 を有効にするには

1. にサインイン AWS Management Console し、<https://console.aws.amazon.com/s3/> で Amazon S3 コンソールを開きます。
2. データをインポートする先のバケットを、バケットのリストから選択します。
3. [Permissions] を選択します。
4. [バケットポリシー] を選択します。
5. [Bucket policy editor] に、次のポリシーを入力します。このポリシーで *YourBucket* のすべてのインスタンスを実際のバケット名に置き換えます。

```
{
  "Version": "2012-10-17",
  "Id": "PutObjPolicy",
  "Statement": [
    {
      "Sid": "DenyIncorrectEncryptionHeader",
      "Effect": "Deny",
      "Principal": "*",
      "Action": "s3:PutObject",
      "Resource": "arn:aws:s3:::YourBucket/*",
      "Condition": {
        "StringNotEquals": {
          "s3:x-amz-server-side-encryption": "AES256"
        }
      }
    },
    {
      "Sid": "DenyUnEncryptedObjectUploads",
      "Effect": "Deny",
      "Principal": "*",
      "Action": "s3:PutObject",
      "Resource": "arn:aws:s3:::YourBucket/*",
      "Condition": {
        "Null": {
          "s3:x-amz-server-side-encryption": "true"
        }
      }
    }
  ]
}
```

6. [保存] を選択します。

Amazon S3 バケットの設定が終了しました。このバケットにデータをインポートすると、データは SSE-S3 によって保護されます。必要に応じて、その他のバケットに対してこの手順を繰り返します。

AWS Key Management ServiceAWS Snowball Edge の

AWS Key Management Service (AWS KMS) は、データの暗号化に使用される暗号化キーの作成と制御を容易にするマネージドサービスです。は、ハードウェアセキュリティモジュール (HSMs)

AWS KMS を使用してキーのセキュリティを保護します。具体的には、AWS Snowball Edge のジョブ用に選択した AWS KMS キーの Amazon リソースネーム (ARN) が KMS キーに関連付けられます。その KMS キーは、ジョブの解除コードの暗号化に使用されます。解除コードは、マニフェストファイルで暗号化の最上位レイヤーを復号するために使用されます。マニフェストファイル内に保存された暗号化キーは、デバイス上のデータを暗号化および復号するために使用されます。

AWS Snowball Edge では、各 AWS Snowball Edge デバイスのデータを保護するために使用される暗号化キー AWS KMS を保護します。ジョブの作成時に、既存の KMS キーを選択することもできます。AWS KMS キーの ARN を指定すると、AWS KMS keys AWS Snowball Edge デバイスの一意のキーの暗号化 AWS Snowball に使用する が指示されます。AWS Snowball Edge でサポートされている Amazon S3 server-side-encryption オプションの詳細については、「」を参照してください [AWS Snowball Edge でのサーバー側の暗号化](#)。

Snowball Edge AWS KMS keys 用 マネージドカスタマーの使用

アカウント用に作成された Snowball Edge AWS KMS keys の マネージドカスタマーを使用する場合は、次の手順に従います。

ジョブで AWS KMS keys を選択するには

1. で AWS Snow ファミリーマネジメントコンソール、ジョブの作成 を選択します。
2. ジョブタイプを選択し、[Next] を選択します。
3. 配送の詳細を入力し、[Next] を選択します。
4. ジョブの詳細を入力し、[Next] を選択します。
5. セキュリティオプションを設定します。暗号化で、KMS キーには、AWS マネージドキーまたは で以前に作成されたカスタムキーを選択するか AWS KMS、別のアカウントが所有するキーを入力する必要がある場合はキー ARN を入力を選択します。

Note

AWS KMS key ARN はカスタマーマネージドキーのグローバル意識別子です。

6. 次へ を選択して、 の選択を終了します AWS KMS key。
7. Snow デバイスの IAM ユーザーに KMS キーへのアクセスを許可します。
 - a. IAM コンソール (<https://console.aws.amazon.com/iam/>) で [暗号化キー] に移動し、デバイス上のデータの暗号化に使用する KMS キーを開きます。

- b. [キーユーザー] で [追加] を選択し、Snow デバイスの IAM ユーザーを検索して [アタッチ] を選択します。

カスタム KMS エンベロープ暗号化キーの作成

AWS Snowball Edge で独自のカスタム AWS KMS エンベロープ暗号化キーを使用することもできます。独自のキーの作成を選択した場合は、ジョブを作成したのと同じリージョンで作成する必要があります。

ジョブ用に独自の AWS KMS キーを作成するには、「AWS Key Management Service デベロッパーガイド」の「[キーの作成](#)」を参照してください。

デバイスのデータの保護

AWS Snowball Edge の保護

AWS Snowball Edge を使用する際に考慮すべきセキュリティポイントと、デバイスが処理 AWS のために に到着したときに実行するその他のセキュリティ上の予防策に関する大まかな情報は次のとおりです。

以下のセキュリティ手段をお勧めします。

- デバイスが到着したら、まず、損傷や明らかな改ざんがないか検査してください。デバイスに疑わしい点が見つかった場合は、内部ネットワークに接続しないでください。[AWS Support](#) にお問い合わせいただければ、新しいデバイスをお客様宛に配送します。
- ジョブの認証情報が漏れないように保護するための取り組みをしてください。ジョブのマニフェストと解除コードにアクセスできる個人は、そのジョブに送信されたデバイスのコンテンツにもアクセスできます。
- 配送ドックにデバイスを放置しないでください。配送ドックに置いたままにすると、屋外環境にさらされることとなります。各 AWS Snowball Edge デバイスは耐久性がありますが、天気はハードウェアの最も堅牢なものにダメージを与える可能性があります。デバイスの盗難、紛失、破損はできるだけ速やかにご報告ください。問題の報告が早いほど、代替品をすぐにお送りしてお客様のジョブを完了させることができます。

Note

AWS Snowball Edge デバイスは のプロパティです AWS。デバイスへの改ざんは、AWS 「利用規定」 ポリシーに違反します。詳細については、<http://aws.amazon.com/aup/> をご覧ください。

私たちは次のセキュリティステップを実行しています。

- Amazon S3 アダプターでデータを転送する場合、オブジェクトメタデータは保持されません。同一のまま残るメタデータは filename および filesize のみです。そのほかのメタデータはすべて、次の例のように設定されます: `-rw-rw-r-- 1 root root [filesize] Dec 31 1969 [path/filename]`
- ファイルインターフェイスでデータを転送する場合、オブジェクトメタデータは保持されます。
- デバイスが に到着すると AWS、改ざんの兆候がないか検査し、トラステッドプラットフォームモジュール (TPM) によって変更が検出されていないことを確認します。AWS Snowball Edge は、不正開封防止筐体、256 ビット暗号化、データのセキュリティと完全な保管チェーンの両方を提供するように設計された業界標準の TPM など、データを保護するように設計された複数のセキュリティレイヤーを使用します。
- データ転送ジョブの処理と検証が完了すると、AWS では National Institute of Standards and Technology (NIST) のメディア衛生のためのガイドラインに従って Snowball デバイスのソフトウェアを消去します。

NFC タグの検証

Snowball Edge Compute Optimized と Snowball Edge Storage Optimized (データ転送用) デバイスには、NFC タグが組み込まれています。これらのタグは、Android 用 AWS Snowball Edge Verification アプリでスキャンできます。これらの NFC タグをスキャンして検証すると、使用する前に、デバイスが改ざんされていないことを確認できます。

NFC タグの検証では、Snowball Edge クライアントを使用してデバイス固有の QR コードを生成し、スキャンするタグが正しいデバイス用のものであることを検証します。

Snowball Edge デバイスで NFC タグを確認する手順を以下で説明します。開始する前に、「使用開始」演習の以下の最初の 5 つのステップを実行したことを確認します。

1. Snowball Edge ジョブを作成します。詳細については、「[Snow Family デバイスを注文するジョブの作成](#)」を参照してください。

2. デバイスを受け取ります。詳細については、「[Snowball Edge の受領](#)」を参照してください。
3. ローカルネットワークに接続します。詳細については、「[ローカルネットワークへの接続](#)」を参照してください。
4. 認証情報とツールを取得します。詳細については、「[Snow Family デバイスにアクセスするための認証情報の取得](#)」を参照してください。
5. Snowball Edge クライアントのダウンロードとインストール 詳細については、「[Snowball Edge クライアントのダウンロードとインストール](#)」を参照してください。

NFC タグを検証するには

1. `snowballEdge get-app-qr-code` Snowball Edge クライアントのコマンドを実行します。クラスターのノードに対してこのコマンドを実行する場合は、シリアル番号 (`--device-sn`) を指定して、1つのノードの QR コードを取得します。クラスターの各ノードについて、このステップを繰り返します。このコマンドの使用の詳細については、「[NFC 検証の QR コードの取得](#)」を参照してください。

QR コードは選択した場所に .png ファイルとして保存されます。

2. 保存した .png ファイルに移動し、これを開き、アプリで QR コードをスキャンできるようにします。
3. これらのタグは、Android の AWS Snowball エッジ検証アプリを使用してスキャンできます。

Note

AWS Snowball エッジ検証アプリはダウンロードできませんが、アプリが既にインストールされているデバイスがある場合は、アプリを使用できます。

4. アプリを起動し、画面の指示に従います。

これでデバイスの NFC タグを正しくスキャンし、検証しました。

スキャン中に問題が発生した場合は、次の操作を試してください。

- デバイスに Snowball Edge Compute Optimized のオプションがあることを確認します (GPU がある場合とない場合)。
- アプリがインストールされている別のデバイスがある場合は、そのデバイスを使用してみてください。

- 他の NFC タグの干渉を受けない、部屋の隔離された場所にデバイスを移動して、もう一度試してください。
- 問題が解決しない場合は、[AWS Support](#) にお問い合わせください。

での Identity and Access Management AWS Snowball

すべての AWS Snowball ジョブが認証されている必要があります。これを行うには、アカウント内に IAM ユーザーを作成し、管理します。IAM を使用して、AWS でユーザーとアクセス許可を作成、管理できます。

AWS Snowball ユーザーは、ジョブを作成するために にアクセス AWS Snowball AWS Management Console するための特定の IAM 関連のアクセス許可を持っている必要があります。インポートまたはエクスポートジョブを作成する IAM ユーザーは、ジョブに使用する Amazon S3 バケット、リソース、Amazon SNS トピック、エッジコンピューティングジョブ用の Amazon EC2-compatible AMI など、適切な Amazon Simple Storage Service (AWS KMS Amazon S3) リソースにもアクセスできる必要があります。

Important

デバイスでローカルに IAM を使用方法については、[ローカルでの IAM の使用](#) を参照してください。

トピック

- [Snow ファミリーコンソールのアクセス制御とジョブの作成](#)

Snow ファミリーコンソールのアクセス制御とジョブの作成

すべての AWS サービスと同様に、へのアクセスには、リクエストの認証に AWS が使用できる認証情報 AWS Snowball が必要です。これらの認証情報には、Amazon S3 バケットや function.differs などの AWS リソースへのアクセス許可が 2 つの方法で必要です。AWS Lambda AWS Snowball

1. のジョブには Amazon リソースネーム (ARNs AWS Snowball がありません。
2. オンプレミスのデバイスに対する物理/ネットワークアクセスコントロールは、お客様の責任で行っていただきます。

[AWS Identity and Access Management \(IAM\)](#) の使用方法と、でリソースにアクセスできるユーザーを制御することでリソース AWS Snowball を保護する方法、およびローカルアクセスコントロールの推奨事項の [Identity and Access Management AWS Snow Family](#) の詳細については AWS クラウド、「」を参照してください。

の Identity and Access Management AWS Snow Family

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

トピック

- [対象者](#)
- [アイデンティティを使用した認証](#)
- [ポリシーを使用したアクセスの管理](#)
- [と IAM の AWS Snow Family 連携方法](#)
- [のアイデンティティベースのポリシーの例 AWS Snow Family](#)
- [AWS Snow Family ID とアクセスのトラブルシューティング](#)

対象者

AWS Identity and Access Management (IAM) の使用法は、で行う作業によって異なります AWS Snow Family。

サービスユーザー – AWS Snow Family サービスを使用してジョブを実行する場合、管理者から必要な認証情報とアクセス許可が与えられます。さらに多くの AWS Snow Family 機能を使用して作業を行う場合は、追加のアクセス許可が必要になることがあります。アクセスの管理方法を理解すると、管理者から適切な権限をリクエストするのに役に立ちます。AWS Snow Family機能にアクセスできない場合は、「[AWS Snow Family ID とアクセスのトラブルシューティング](#)」を参照してください。

サービス管理者 – 社内の AWS Snow Family リソースを担当している場合は、通常、へのフルアクセスがあります AWS Snow Family。サービスユーザーがどの AWS Snow Family 機能やリソースにアクセスするかを決めるのは管理者の仕事です。その後、IAM 管理者にリクエストを送信して、サービスユーザーの権限を変更する必要があります。このページの情報を点検して、IAM の基本概

念を理解してください。会社で IAM を使用する方法の詳細については、AWS Snow Family 「」を参照してくださいと [IAM の AWS Snow Family 連携方法](#)。

IAM 管理者 - 管理者は、AWS Snow Family へのアクセスを管理するポリシーの書き込み方法の詳細について確認する場合があります。IAM で使用できる AWS Snow Family アイデンティティベースのポリシーの例を表示するには、「」を参照してくださいの [アイデンティティベースのポリシーの例 AWS Snow Family](#)。

アイデンティティを使用した認証

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

ID ソースを介して提供された認証情報を使用して、フェデレーテッド ID AWS としてにサインインできます。AWS IAM Identity Center (IAM Identity Center) ユーザー、会社のシングルサインオン認証、Google または Facebook の認証情報は、フェデレーション ID の例です。フェデレーテッドアイデンティティとしてサインインする場合、IAM ロールを使用して、前もって管理者により ID フェデレーションが設定されています。フェデレーション AWS を使用してにアクセスすると、間接的にロールを引き受けることとなります。

ユーザーのタイプに応じて、AWS Management Console または AWS アクセスポータルにサインインできます。へのサインインの詳細については AWS、「ユーザーガイド」の「[にサインインする方法 AWS アカウント](#) AWS サインイン」を参照してください。

AWS プログラムでにアクセスする場合、は Software Development Kit (SDK) とコマンドラインインターフェイス (CLI) AWS を提供し、認証情報を使用してリクエストに暗号で署名します。AWS ツールを使用しない場合は、リクエストに自分で署名する必要があります。推奨される方法を使用してリクエストを自分で署名する方法の詳細については、IAM [ユーザーガイドの API AWS リクエスト](#) の署名を参照してください。

使用する認証方法を問わず、追加セキュリティ情報の提供をリクエストされる場合もあります。例えば、AWS では、多要素認証 (MFA) を使用してアカウントのセキュリティを向上させることをお勧めします。詳細については、『AWS IAM Identity Center ユーザーガイド』の「[Multi-factor authentication](#)」(多要素認証) および『IAM ユーザーガイド』の「[AWS における多要素認証 \(MFA\) の使用](#)」を参照してください。

AWS アカウント ルートユーザー

を作成するときは AWS アカウント、アカウント内のすべての およびリソースへの AWS のサービス 完全なアクセス権を持つ 1 つのサインインアイデンティティから始めます。この ID は AWS アカウ

ント ルートユーザーと呼ばれ、アカウントの作成に使用した E メールアドレスとパスワードでサインインすることでアクセスできます。日常的なタスクには、ルートユーザーを使用しないことを強くお勧めします。ルートユーザーの認証情報は保護し、ルートユーザーでしか実行できないタスクを実行するときに使用します。ルートユーザーとしてサインインする必要があるタスクの完全なリストについては、『IAM ユーザーガイド』の「[ルートユーザー認証情報が必要なタスク](#)」を参照してください。

フェデレーティッドアイデンティティ

ベストプラクティスとして、管理者アクセスを必要とするユーザーを含む人間のユーザーに、一時的な認証情報を使用してにアクセスするための ID プロバイダーとのフェデレーションの使用を要求 AWS のサービスします。

フェデレーティッド ID は、エンタープライズユーザーディレクトリ、ウェブ ID プロバイダー、AWS Directory Service、Identity Center ディレクトリのユーザー、または ID ソースを通じて提供された認証情報 AWS のサービス を使用してにアクセスするユーザーです。フェデレーティッド ID がにアクセスすると AWS アカウント、ロールを引き受け、ロールは一時的な認証情報を提供します。

アクセスを一元管理する場合は、AWS IAM Identity Centerを使用することをお勧めします。IAM Identity Center でユーザーとグループを作成することも、独自の ID ソース内のユーザーとグループのセットに接続して同期して、すべての AWS アカウント とアプリケーションで使用できるようにすることもできます。IAM Identity Center の詳細については、『AWS IAM Identity Center ユーザーガイド』の「[What is IAM Identity Center?](#)」(IAM Identity Center とは) を参照してください。

IAM ユーザーとグループ

[IAM ユーザー](#)は、単一のユーザーまたはアプリケーションに対して特定のアクセス許可 AWS アカウント を持つ 内のアイデンティティです。可能であれば、パスワードやアクセスキーなどの長期的な認証情報を保有する IAM ユーザーを作成する代わりに、一時認証情報を使用することをお勧めします。ただし、IAM ユーザーでの長期的な認証情報が必要な特定のユースケースがある場合は、アクセスキーをローテーションすることをお勧めします。詳細については、「IAM ユーザーガイド」の「[長期的な認証情報を必要とするユースケースのためにアクセスキーを定期的にローテーションする](#)」を参照してください。

[IAM グループ](#)は、IAM ユーザーの集団を指定するアイデンティティです。グループとしてサインインすることはできません。グループを使用して、複数のユーザーに対して一度に権限を指定できます。多数のユーザーグループがある場合、グループを使用することで権限の管理が容易になります。例えば、IAMAdmins という名前のグループを設定して、そのグループに IAM リソースを管理する権限を与えることができます。

ユーザーは、ロールとは異なります。ユーザーは 1 人の人または 1 つのアプリケーションに一意に関連付けられますが、ロールはそれを必要とする任意の人が引き受けるようになっています。ユーザーには永続的な長期の認証情報がありますが、ロールでは一時的な認証情報が提供されます。詳細については、『IAM ユーザーガイド』の「[IAM ユーザー \(ロールではなく\) の作成が適している場合](#)」を参照してください。

IAM ロール

[IAM ロール](#)は、特定のアクセス許可 AWS アカウント を持つ 内のアイデンティティです。これは IAM ユーザーに似ていますが、特定のユーザーには関連付けられていません。ロール を切り替える AWS Management Console ことで、[で IAM ロール](#)を一時的に引き受けることができます。ロールを引き受けるには、または AWS API AWS CLI オペレーションを呼び出すか、カスタム URL を使用します。ロールを使用する方法の詳細については、「IAM ユーザーガイド」の「[IAM ロールの使用](#)」を参照してください。

IAM ロールと一時的な認証情報は、次の状況で役立ちます:

- フェデレーションユーザーアクセス - フェデレーテッドアイデンティティに権限を割り当てるには、ロールを作成してそのロールの権限を定義します。フェデレーテッドアイデンティティが認証されると、そのアイデンティティはロールに関連付けられ、ロールで定義されている権限が付与されます。フェデレーションの詳細については、『IAM ユーザーガイド』の「[サードパーティーアイデンティティプロバイダー向けロールの作成](#)」を参照してください。IAM アイデンティティセンターを使用する場合、権限セットを設定します。アイデンティティが認証後にアクセスできるものを制御するため、IAM Identity Center は、権限セットを IAM のロールに関連付けます。権限セットの詳細については、『AWS IAM Identity Center ユーザーガイド』の「[権限セット](#)」を参照してください。
- 一時的な IAM ユーザー権限 - IAM ユーザーまたはロールは、特定のタスクに対して複数の異なる権限を一時的に IAM ロールで引き受けることができます。
- クロスアカウントアクセス - IAM ロールを使用して、自分のアカウントのリソースにアクセスすることを、別のアカウントの人物 (信頼済みプリンシパル) に許可できます。クロスアカウントアクセス権を付与する主な方法は、ロールを使用することです。ただし、一部の では AWS のサービス、(ロールをプロキシとして使用する代わりに) リソースにポリシーを直接アタッチできません。クロスアカウントアクセスにおけるロールとリソースベースのポリシーの違いについては、『IAM ユーザーガイド』の「[IAM ロールとリソースベースのポリシーとの相違点](#)」を参照してください。
- クロスサービスアクセス — 一部の は、他の の機能 AWS のサービス を使用します AWS のサービス。例えば、あるサービスで呼び出しを行うと、通常そのサービスによって Amazon EC2 でアプリケーションが実行されたり、Amazon S3 にオブジェクトが保存されたりします。サービスで

は、呼び出し元プリンシパルの権限、サービスロール、またはサービスにリンクされたロールを使用してこれを行う場合があります。

- 転送アクセスセッション (FAS) – IAM ユーザーまたはロールを使用してアクションを実行する場合 AWS、ユーザーはプリンシパルと見なされます。一部のサービスを使用する際に、アクションを実行することで、別のサービスの別のアクションがトリガーされることがあります。FAS は、 を呼び出すプリンシパルのアクセス許可を AWS のサービス、ダウンストリームサービス AWS のサービス へのリクエストリクエストリクエストと組み合わせて使用します。FAS リクエストは、サービスが他の AWS のサービス またはリソースとのやり取りを完了する必要があるリクエストを受け取った場合にのみ行われます。この場合、両方のアクションを実行するためのアクセス許可が必要です。FAS リクエストを行う際のポリシーの詳細については、「[転送アクセスセッション](#)」を参照してください。
- サービスロール - サービスがユーザーに代わってアクションを実行するために引き受ける [IAM ロール](#)です。IAM 管理者は、IAM 内からサービスロールを作成、変更、削除できます。詳細については、「IAM ユーザーガイド」の「[AWS のサービスに権限を委任するロールの作成](#)」を参照してください。
- サービスにリンクされたロール – サービスにリンクされたロールは、 にリンクされたサービスロールの一種です AWS のサービス。サービスは、ユーザーに代わってアクションを実行するロールを引き受けることができます。サービスにリンクされたロールは に表示され AWS アカウント、サービスによって所有されます。IAM 管理者は、サービスにリンクされたロールの権限を表示できますが、編集することはできません。
- Amazon EC2 で実行されているアプリケーション – IAM ロールを使用して、EC2 インスタンスで実行され、AWS CLI または AWS API リクエストを行うアプリケーションの一時的な認証情報を管理できます。これは、EC2 インスタンス内でのアクセスキーの保存に推奨されます。AWS ロールを EC2 インスタンスに割り当て、そのすべてのアプリケーションで使用できるようにするには、インスタンスにアタッチされたインスタンスプロファイルを作成します。インスタンスプロファイルにはロールが含まれ、EC2 インスタンスで実行されるプログラムは一時的な認証情報を取得できます。詳細については、『IAM ユーザーガイド』の「[Amazon EC2 インスタンスで実行されるアプリケーションに IAM ロールを使用して権限を付与する](#)」を参照してください。

IAM ロールと IAM ユーザーのどちらを使用するかについては、『IAM ユーザーガイド』の「[\(IAM ユーザーではなく\) IAM ロールをいつ作成したら良いのか?](#)」を参照してください。

ポリシーを使用したアクセスの管理

でアクセスを制御する AWS には、ポリシーを作成し、AWS ID またはリソースにアタッチします。ポリシーは AWS、アイデンティティまたはリソースに関連付けられているときにアクセス許可を

定義する のオブジェクトです。 は、プリンシパル(ユーザー、ルートユーザー、またはロールセッション) がリクエストを行うときに、これらのポリシー AWS を評価します。ポリシーでの権限により、リクエストが許可されるか拒否されるかが決まります。ほとんどのポリシーは JSON ドキュメント AWS として に保存されます。JSON ポリシードキュメントの構造と内容の詳細については、「IAM ユーザーガイド」の「[JSON ポリシー概要](#)」を参照してください。

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

デフォルトでは、ユーザーやロールに権限はありません。IAM 管理者は、リソースで必要なアクションを実行するための権限をユーザーに付与する IAM ポリシーを作成できます。その後、管理者はロールに IAM ポリシーを追加し、ユーザーはロールを引き継ぐことができます。

IAM ポリシーは、オペレーションの実行方法を問わず、アクションの権限を定義します。例えば、iam:GetRole アクションを許可するポリシーがあるとします。そのポリシーを持つユーザーは、AWS Management Console、AWS CLI または AWS API からロール情報を取得できます。

アイデンティティベースのポリシー

アイデンティティベースポリシーは、IAM ユーザー、ユーザーのグループ、ロールなど、アイデンティティにアタッチできる JSON 権限ポリシードキュメントです。これらのポリシーは、ユーザーとロールが実行できるアクション、リソース、および条件をコントロールします。アイデンティティベースのポリシーを作成する方法については、「IAM ユーザーガイド」の「[IAM ポリシーの作成](#)」を参照してください。

アイデンティティベースポリシーは、さらにインラインポリシーまたはマネージドポリシーに分類できます。インラインポリシーは、単一のユーザー、グループ、またはロールに直接埋め込まれています。管理ポリシーは、内の複数のユーザー、グループ、ロールにアタッチできるスタンドアロンポリシーです AWS アカウント。管理ポリシーには、AWS 管理ポリシーとカスタマー管理ポリシーが含まれます。マネージドポリシーまたはインラインポリシーのいずれかを選択する方法については、『IAM ユーザーガイド』の「[マネージドポリシーとインラインポリシーの比較](#)」を参照してください。

リソースベースのポリシー

リソースベースのポリシーは、リソースに添付する JSON ポリシードキュメントです。リソースベースのポリシーには例として、IAM ロールの信頼ポリシーや Amazon S3 バケットポリシーがあげられます。リソースベースのポリシーをサポートするサービスでは、サービス管理者はポリシーを使用して特定のリソースへのアクセスを制御できます。ポリシーがアタッチされているリソースの場合、指定されたプリンシパルがそのリソースに対して実行できるアクションと条件は、ポリシーに

よって定義されます。リソースベースのポリシーでは、[プリンシパルを指定する](#)必要があります。プリンシパルには、アカウント、ユーザー、ロール、フェデレーテッドユーザー、またはを含めることができます AWS のサービス。

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

アクセスコントロールリスト (ACL)

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

Amazon S3、AWS WAF、および Amazon VPC は、ACLs。ACL の詳細については、『Amazon Simple Storage Service デベロッパーガイド』の「[アクセスコントロールリスト \(ACL\) の概要](#)」を参照してください。

その他のポリシータイプ

AWS は、一般的ではない追加のポリシータイプをサポートします。これらのポリシータイプでは、より一般的なポリシータイプで付与された最大の権限を設定できます。

- **アクセス許可の境界** - アクセス許可の境界は、アイデンティティベースのポリシーによって IAM エンティティ (IAM ユーザーまたはロール) に付与できる権限の上限を設定する高度な機能です。エンティティにアクセス許可の境界を設定できます。結果として得られる権限は、エンティティのアイデンティティベースポリシーとそのアクセス許可の境界の共通部分になります。Principal フィールドでユーザーまたはロールを指定するリソースベースのポリシーでは、アクセス許可の境界は制限されません。これらのポリシーのいずれかを明示的に拒否した場合、権限は無効になります。許可の境界の詳細については、「IAM ユーザーガイド」の「[IAM エンティティの許可の境界](#)」を参照してください。
- **サービスコントロールポリシー (SCPs)** - SCPs は、の組織または組織単位 (OU) に対する最大アクセス許可を指定する JSON ポリシーです AWS Organizations。AWS Organizations は、AWS アカウント ビジネスが所有する複数のをグループ化して一元管理するサービスです。組織内のすべての機能を有効にすると、サービスコントロールポリシー (SCP) を一部またはすべてのアカウントに適用できます。SCP は、各を含むメンバーアカウントのエンティティのアクセス許可を制限します AWS アカウントのルートユーザー。Organizations と SCP の詳細については、『AWS Organizations ユーザーガイド』の「[SCP の仕組み](#)」を参照してください。
- **セッションポリシー** - セッションポリシーは、ロールまたはフェデレーションユーザーの一時的なセッションをプログラムで作成する際にパラメータとして渡す高度なポリシーです。結果として

セッションの権限は、ユーザーまたはロールのアイデンティティベースポリシーとセッションポリシーの共通部分になります。また、リソースベースのポリシーから権限が派生する場合があります。これらのポリシーのいずれかを明示的に拒否した場合、権限は無効になります。詳細については、「IAM ユーザーガイド」の「[セッションポリシー](#)」を参照してください。

複数のポリシータイプ

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

と IAM の AWS Snow Family 連携方法

IAM を使用してへのアクセスを管理する前に AWS Snow Family、で使用できる IAM 機能について学びます AWS Snow Family。

で使用できる IAM の機能 AWS Snow Family

IAM 機能	AWS Snow Family サポート
アイデンティティベースのポリシー	Yes
リソースベースのポリシー	はい
ポリシーアクション	Yes
ポリシーリソース	はい
ポリシー条件キー (サービス固有)	はい
ACL	No
ABAC (ポリシー内のタグ)	部分的
一時的な認証情報	はい
転送アクセスセッション (FAS)	はい
サービスロール	あり

IAM 機能	AWS Snow Family サポート
サービスリンクロール	いいえ

AWS Snow Family およびその他の AWS のサービスがほとんどの IAM 機能と連携する方法の概要を把握するには、「IAM ユーザーガイド」の「IAM [AWS と連携する のサービス](#)」を参照してください。

のアイデンティティベースのポリシー AWS Snow Family

アイデンティティベースポリシーをサポートする **Yes**

アイデンティティベースポリシーは、IAM ユーザー、ユーザーグループ、ロールなど、アイデンティティにアタッチできる JSON 権限ポリシードキュメントです。これらのポリシーは、ユーザーとロールが実行できるアクション、リソース、および条件をコントロールします。アイデンティティベースのポリシーを作成する方法については、『IAM ユーザーガイド』の「[IAM ポリシーの作成](#)」を参照してください。

IAM アイデンティティベースのポリシーでは、許可または拒否するアクションとリソース、およびアクションを許可または拒否する条件を指定できます。プリンシパルは、それが添付されているユーザーまたはロールに適用されるため、アイデンティティベースのポリシーでは指定できません。JSON ポリシーで使用できるすべての要素については、「IAM ユーザーガイド」の「[IAM JSON ポリシーの要素のリファレンス](#)」を参照してください。

のアイデンティティベースのポリシーの例 AWS Snow Family

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

内のリソースベースのポリシー AWS Snow Family

リソースベースのポリシーのサポート **はい**

リソースベースのポリシーは、リソースに添付する JSON ポリシードキュメントです。リソースベースのポリシーには例として、IAM ロールの信頼ポリシーや Amazon S3 バケットポリシーがあげ

られます。リソースベースのポリシーをサポートするサービスでは、サービス管理者はポリシーを使用して特定のリソースへのアクセスを制御できます。ポリシーがアタッチされているリソースの場合、指定されたプリンシパルがそのリソースに対して実行できるアクションと条件は、ポリシーによって定義されます。リソースベースのポリシーでは、[プリンシパルを指定する](#)必要があります。プリンシパルには、アカウント、ユーザー、ロール、フェデレーテッドユーザー、または [を含めることができます](#) AWS のサービス。

クロスアカウントアクセスを有効にするには、アカウント全体、または別のアカウントの IAM エンティティをリソースベースのポリシーのプリンシパルとして指定します。リソースベースのポリシーにクロスアカウントのプリンシパルを追加しても、信頼関係は半分しか確立されない点に注意してください。プリンシパルとリソースが異なる がある場合 AWS アカウント、信頼されたアカウントの IAM 管理者は、プリンシパルエンティティ (ユーザーまたはロール) にリソースへのアクセス許可も付与する必要があります。IAM 管理者は、アイデンティティベースのポリシーをエンティティにアタッチすることで権限を付与します。ただし、リソースベースのポリシーで、同じアカウントのプリンシパルへのアクセス権が付与されている場合は、アイデンティティベースのポリシーを追加する必要はありません。詳細については、『IAM ユーザーガイド』の「[IAM ロールとリソースベースのポリシーとの相違点](#)」を参照してください。

のポリシーアクション AWS Snow Family

ポリシーアクションに対するサポート	はい
-------------------	----

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

JSON ポリシーの Action 要素には、ポリシー内のアクセスを許可または拒否するために使用できるアクションが記述されます。ポリシーアクションの名前は通常、関連付けられた AWS API オペレーションと同じです。一致する API オペレーションのない権限のみのアクションなど、いくつかの例外があります。また、ポリシーに複数アクションが必要なオペレーションもあります。これらの追加アクションは、[依存アクション](#) と呼ばれます。

このアクションは、関連付けられたオペレーションを実行するための権限を付与するポリシーで使用されます。

AWS Snow Family アクションのリストを確認するには、「サービス認証リファレンス」の「[で定義されるアクション AWS Snow Family](#)」を参照してください。

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

```
snowball
```

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

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

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

のポリシーリソース AWS Snow Family

ポリシーリソースに対するサポート	はい
------------------	----

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

Resource JSON ポリシー要素は、アクションが適用されるオブジェクトを指定します。ステートメントには、Resource または NotResource 要素を含める必要があります。ベストプラクティスとして、[Amazon リソースネーム \(ARN\)](#) を使用してリソースを指定します。これは、リソースレベルの権限と呼ばれる特定のリソースタイプをサポートするアクションに対して実行できます。

オペレーションのリスト化など、リソースレベルの権限をサポートしないアクションの場合は、ステートメントがすべてのリソースに適用されることを示すために、ワイルドカード (*) を使用します。

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

AWS Snow Family リソースタイプとその ARNs」の「[で定義されるリソース AWS Snow Family](#)」を参照してください。どのアクションで各リソースの ARN を指定できるかについては、「[AWS Snow Family で定義されるアクション](#)」を参照してください。

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

のポリシー条件キー AWS Snow Family

サービス固有のポリシー条件キーのサポート	はい
----------------------	----

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

Condition 要素 (または Condition ブロック) を使用すると、ステートメントが有効な条件を指定できます。Condition 要素はオプションです。イコールや未満などの [条件演算子](#) を使用して条件式を作成することで、ポリシーの条件とリクエスト内の値を一致させることができます。

1 つのステートメントに複数の Condition 要素を指定するか、1 つの Condition 要素に複数のキーを指定すると、AWS は AND 論理演算子を使用してそれら进行评估します。1 つの条件キーに複数の値を指定すると、は論理ORオペレーションを使用して条件 AWS を评估します。ステートメントの権限が付与される前にすべての条件が満たされる必要があります。

条件を指定する際にプレースホルダー変数も使用できます。例えば IAM ユーザーに、IAM ユーザー名がタグ付けされている場合のみリソースにアクセスできる権限を付与することができます。詳細については、『IAM ユーザーガイド』の「[IAM ポリシーの要素: 変数およびタグ](#)」を参照してください。

AWS は、グローバル条件キーとサービス固有の条件キーをサポートします。すべての AWS グローバル条件キーを確認するには、「IAM ユーザーガイド」の[AWS 「グローバル条件コンテキストキー」](#)を参照してください。

AWS Snow Family 条件キーのリストを確認するには、「サービス認証リファレンス」の「[の条件キー AWS Snow Family](#)」を参照してください。条件キーを使用できるアクションとリソースについては、「[で定義されるアクション AWS Snow Family](#)」を参照してください。

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

ACLs AWS Snow Family

ACL のサポート	No
-----------	----

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

での ABAC AWS Snow Family

ABAC (ポリシー内のタグ) のサポート	部分的
-----------------------	-----

属性ベースのアクセスコントロール (ABAC) は、属性に基づいて権限を定義する認可戦略です。では AWS、これらの属性はタグと呼ばれます。タグは、IAM エンティティ (ユーザーまたはロール) および多くの AWS リソースにアタッチできます。エンティティとリソースのタグ付けは、ABAC の最初の手順です。その後、プリンシパルのタグがアクセスしようとしているリソースのタグと一致した場合に操作を許可するように ABAC ポリシーを設計します。

ABAC は、急成長する環境やポリシー管理が煩雑になる状況で役立ちます。

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

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

ABAC の詳細については、『IAM ユーザーガイド』の「[ABAC とは?](#)」を参照してください。ABAC をセットアップするステップを説明するチュートリアルについては、「IAM ユーザーガイド」の「[属性に基づくアクセスコントロール \(ABAC\) を使用する](#)」を参照してください。

での一時的な認証情報の使用 AWS Snow Family

一時的な認証情報のサポート	はい
---------------	----

一部の は、一時的な認証情報を使用してサインインすると機能 AWS のサービスしません。一時的な認証情報 AWS のサービス を使用する などの詳細については、IAM ユーザーガイドの [AWS のサービス「IAM と連携する](#)」を参照してください。

ユーザー名とパスワード以外の AWS Management Console 方法で にサインインする場合、一時的な認証情報を使用します。例えば、会社の Single Sign-On (SSO) リンク AWS を使用して にアクセ

スすると、そのプロセスによって一時的な認証情報が自動的に作成されます。また、ユーザーとしてコンソールにサインインしてからロールを切り替える場合も、一時的な認証情報が自動的に作成されます。ロールの切り替えに関する詳細については、「IAM ユーザーガイド」の「[ロールへの切り替え \(コンソール\)](#)」を参照してください。

一時的な認証情報は、AWS CLI または AWS API を使用して手動で作成できます。その後、これらの一時的な認証情報を使用して、AWS recommends にアクセスできます AWS。この際、長期的なアクセスキーを使用する代わりに、一時的な認証情報を動的に生成することをお勧めします。詳細については、「[IAM の一時的セキュリティ認証情報](#)」を参照してください。

の転送アクセスセッション AWS Snow Family

転送アクセスセッション (FAS) をサポート はい

IAM ユーザーまたはロールを使用してアクションを実行すると AWS、プリンシパルと見なされます。一部のサービスを使用する際に、アクションを実行することで、別のサービスの別のアクションがトリガーされることがあります。FAS は、 を呼び出すプリンシパルのアクセス許可を AWS のサービス、ダウンストリームサービス AWS のサービス へのリクエストリクエストと組み合わせて使用します。FAS リクエストは、サービスが他の AWS のサービス またはリソースとのやり取りを完了する必要があるリクエストを受け取った場合にのみ行われます。この場合、両方のアクションを実行するためのアクセス許可が必要です。FAS リクエストを行う際のポリシーの詳細については、「[転送アクセスセッション](#)」を参照してください。

AWS Snow Familyのサービスロール

サービスロールに対するサポート あり

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

⚠ Warning

サービスロールのアクセス許可を変更すると、AWS Snow Family 機能が破損する可能性があります。が指示する場合以外 AWS Snow Family は、サービスロールを編集しないでください。

のサービスにリンクされたロール AWS Snow Family

サービスにリンクされたロールのサポート いいえ

サービスにリンクされたロールは、にリンクされたサービスロールの一種です AWS のサービス。サービスは、ユーザーに代わってアクションを実行するロールを引き受けることができます。サービスにリンクされたロールは に表示され AWS アカウント、サービスによって所有されます。IAM 管理者は、サービスにリンクされたロールの権限を表示できますが、編集することはできません。

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

のアイデンティティベースのポリシーの例 AWS Snow Family

デフォルトでは、ユーザーおよびロールには、AWS Snow Family リソースを作成または変更する権限はありません。また、AWS Command Line Interface (AWS CLI) AWS Management Console、または AWS API を使用してタスクを実行することはできません。IAM 管理者は、リソースで必要なアクションを実行するための権限をユーザーに付与する IAM ポリシーを作成できます。その後、管理者はロールに IAM ポリシーを追加し、ユーザーはロールを引き受けることができます。

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

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

トピック

- [ポリシーのベストプラクティス](#)
- [AWS Snow Family コンソールを使用する](#)
- [自分の権限の表示をユーザーに許可する](#)

ポリシーのベストプラクティス

ID ベースのポリシーは、ユーザーのアカウントで誰かが AWS Snow Family リソースを作成、アクセス、または削除できるかどうかを決定します。これらのアクションを実行すると、AWS アカウントに料金が発生する可能性があります。アイデンティティベースポリシーを作成したり編集したりする際には、以下のガイドラインと推奨事項に従ってください:

- AWS 管理ポリシーを開始し、最小特権のアクセス許可に移行する – ユーザーとワークロードにアクセス許可を付与するには、多くの一般的なユースケースにアクセス許可を付与する AWS 管理ポリシーを使用します。これらはで使用できます AWS アカウント。ユースケースに固有の AWS カスタマー管理ポリシーを定義して、アクセス許可をさらに減らすことをお勧めします。詳細については、『IAM ユーザーガイド』の「[AWS マネージドポリシー](#)」または「[AWS ジョブ機能の管理ポリシー](#)」を参照してください。
- 最小特権を適用する – IAM ポリシーで権限を設定するときは、タスクの実行に必要な権限のみを付与します。これを行うには、特定の条件下で特定のリソースに対して実行できるアクションを定義します。これは、最小特権権限とも呼ばれています。IAM を使用して権限を適用する方法の詳細については、『IAM ユーザーガイド』の「[IAM でのポリシーと権限](#)」を参照してください。
- IAM ポリシーで条件を使用してアクセスをさらに制限する - ポリシーに条件を追加して、アクションやリソースへのアクセスを制限できます。例えば、ポリシー条件を記述して、すべてのリクエストを SSL を使用して送信するように指定できます。条件を使用して、などの特定の を介してサービスアクションが使用される場合に AWS のサービス、サービスアクションへのアクセスを許可することもできます AWS CloudFormation。詳細については、『IAM ユーザーガイド』の「[IAM JSON policy elements: Condition](#)」(IAM JSON ポリシー要素 : 条件) を参照してください。
- IAM Access Analyzer を使用して IAM ポリシーを検証し、安全で機能的な権限を確保する - IAM Access Analyzer は、新規および既存のポリシーを検証して、ポリシーが IAM ポリシー言語 (JSON) および IAM のベストプラクティスに準拠するようにします。IAM アクセスアナライザーは 100 を超えるポリシーチェックと実用的な推奨事項を提供し、安全で機能的なポリシーの作成をサポートします。詳細については、『IAM ユーザーガイド』の「[IAM Access Analyzer ポリシーの検証](#)」を参照してください。
- 多要素認証 (MFA) を要求する – で IAM ユーザーまたはルートユーザーを必要とするシナリオがある場合は AWS アカウント、セキュリティを強化するために MFA を有効にします。API オペレー

ションが呼び出されるときに MFA を必須にするには、ポリシーに MFA 条件を追加します。詳細については、『IAM ユーザーガイド』の「[MFA 保護 API アクセスの設定](#)」を参照してください。

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

AWS Snow Family コンソールを使用する

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

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

ユーザーとロールが AWS Snow Family 引き続きコンソールを使用できるようにするには、エンティティに AWS Snow Family *ConsoleAccess* または *ReadOnly* AWS 管理ポリシーもアタッチします。詳細については、「IAM ユーザーガイド」の「[ユーザーへのアクセス許可の追加](#)」を参照してください。

自分の権限の表示をユーザーに許可する

この例では、ユーザーアイデンティティにアタッチされたインラインおよびマネージドポリシーの表示を IAM ユーザーに許可するポリシーの作成方法を示します。このポリシーには、コンソールで、または AWS CLI または AWS API を使用してプログラムでこのアクションを実行するアクセス許可が含まれています。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "ViewOwnUserInfo",
      "Effect": "Allow",
      "Action": [
        "iam:GetUserPolicy",
        "iam:ListGroupsWithUser",
        "iam:ListAttachedUserPolicies",
```

```
        "iam:ListUserPolicies",
        "iam:GetUser"
    ],
    "Resource": ["arn:aws:iam::*:user/${aws:username}"]
},
{
    "Sid": "NavigateInConsole",
    "Effect": "Allow",
    "Action": [
        "iam:GetGroupPolicy",
        "iam:GetPolicyVersion",
        "iam:GetPolicy",
        "iam:ListAttachedGroupPolicies",
        "iam:ListGroupPolicies",
        "iam:ListPolicyVersions",
        "iam:ListPolicies",
        "iam:ListUsers"
    ],
    "Resource": "*"
}
]
```

AWS Snow Family ID とアクセスのトラブルシューティング

次の情報は、 と IAM の使用時に発生する可能性がある一般的な問題の診断 AWS Snow Family と修正に役立ちます。

トピック

- [でアクションを実行する権限がない AWS Snow Family](#)
- [iam を実行する権限がありません。PassRole](#)
- [自分の 以外のユーザーに自分の AWS Snow Family リソース AWS アカウント へのアクセスを許可したい](#)

でアクションを実行する権限がない AWS Snow Family

「I am not authorized to perform an action in Amazon Bedrock」というエラーが表示された場合、そのアクションを実行できるようにポリシーを更新する必要があります。

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

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

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

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

iam を実行する権限がありません。PassRole

iam:PassRole アクションを実行する権限がないというエラーが表示された場合は、ポリシーを更新して AWS Snow Family にロールを渡すことができるようにする必要があります。

一部の AWS のサービスでは、新しいサービスロールまたはサービスにリンクされたロールを作成する代わりに、そのサービスに既存のロールを渡すことができます。そのためには、サービスにロールを渡す権限が必要です。

以下の例のエラーは、marymajor という IAM ユーザーがコンソールを使用して AWS Snow Family でアクションを実行しようする場合に発生します。ただし、このアクションをサービスが実行するには、サービスロールから付与された権限が必要です。Mary には、ロールをサービスに渡す権限がありません。

```
User: arn:aws:iam::123456789012:user/marymajor is not authorized to perform:
iam:PassRole
```

この場合、Mary のポリシーを更新してメアリーに `iam:PassRole` アクションの実行を許可する必要があります。

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

自分の 以外のユーザーに自分の AWS Snow Family リソース AWS アカウント へのアクセスを許可したい

他のアカウントのユーザーや組織外の人が、リソースにアクセスするために使用できるロールを作成できます。ロールの引き受けを委託するユーザーを指定できます。リソースベースのポリシーまた

はアクセスコントロールリスト (ACL) をサポートするサービスの場合、それらのポリシーを使用して、リソースへのアクセスを付与できます。

詳細については、以下を参照してください:

- がこれらの機能 AWS Snow Family をサポートしているかどうかを確認するには、「」を参照してくださいと [IAM の AWS Snow Family 連携方法](#)。
- 所有 AWS アカウント しているのリソースへのアクセスを提供する方法については、[IAM ユーザーガイドの「所有 AWS アカウント している別の の IAM ユーザーへのアクセスを提供する」](#)を参照してください。
- リソースへのアクセスをサードパーティー に提供する方法については AWS アカウント、IAM ユーザーガイドの「[サードパーティー AWS アカウント が所有する へのアクセスを提供する](#)」を参照してください。
- ID フェデレーションを介してアクセスを提供する方法については、『IAM ユーザーガイド』の「[外部で認証されたユーザー \(ID フェデレーション\) へのアクセス権限](#)」を参照してください。
- クロスアカウントアクセスでのロールとリソースベースのポリシーの使用の違いの詳細については、『IAM ユーザーガイド』の「[IAM ロールとリソースベースのポリシーとの相違点](#)」を参照してください。

のアクセスコントロール AWS クラウド

AWSでリクエストを認証するための有効な認証情報を持つことができます。ただし、アクセス許可がない限り、AWS リソースを作成またはアクセスすることはできません。例えば、Snow Family デバイスを注文するジョブを作成するアクセス許可が必要です。

以下のセクションでは、AWS Snowballのクラウドベースのアクセス権限を管理する方法について説明します。最初に概要のセクションを読むことをお勧めします。

- [のリソースへのアクセス許可の管理の概要 AWS クラウド](#)
- [でのアイデンティティベースのポリシー \(IAM ポリシー\) の使用 AWS Snowball](#)

のリソースへのアクセス許可の管理の概要 AWS クラウド

すべてのAWS リソースは によって所有され AWS アカウント、リソースを作成またはアクセスするためのアクセス許可はアクセス許可ポリシーによって管理されます。アカウント管理者は、IAM ID (ユーザー、グループ、ロール) にアクセス許可ポリシーをアタッチできます。一部のサービス (など AWS Lambda) では、リソースへのアクセス許可ポリシーのアタッチもサポートされています。

Note

アカウント管理者 (または管理者ユーザー) は、管理者権限を持つユーザーです。詳細については、「IAM ユーザーガイド」の「[IAM のベストプラクティス](#)」を参照してください。

トピック

- [リソースおよびオペレーション](#)
- [リソース所有権について](#)
- [でのリソースへのアクセスの管理 AWS クラウド](#)
- [ポリシー要素 \(アクション、効果、プリンシパル\) の指定](#)
- [ポリシーでの条件の指定](#)

リソースおよびオペレーション

では AWS Snowball、プライマリリソースはジョブです。AWS Snowball には Snowball や デバイスなどの AWS Snowball Edge デバイスもありますが、これらのデバイスは既存のジョブのコンテキストでのみ使用できます。Amazon S3 バケットおよび Lambda 関数は、それぞれ Amazon S3 と Lambda のリソースです。

前述のとおり、ジョブには関連付けられた Amazon リソースネーム (ARN) はありません。ただし、Amazon S3 バケットなどの他のサービスのリソースには、以下の表に示すように、一意の ARN が関連付けられています。

AWS Snowball は、ジョブを作成および管理するための一連のオペレーションを提供します。使用可能なオペレーションのリストについては、[AWS Snowball API リファレンス](#)を参照してください。

リソース所有権について

は、リソースを作成したユーザーに関係なく、アカウントで作成されたリソース AWS アカウントを所有します。具体的には、リソース所有者は、リソース作成リクエスト AWS アカウント を認証する [プリンシパルエンティティ](#) (ルートアカウント、IAM ユーザー、または IAM ロール) のです。次の例は、この仕組みを示しています。

- のルートアカウントの認証情報 AWS アカウント を使用して S3 バケットを作成する場合、AWS アカウント はリソースの所有者です (では AWS Snowball、リソースはジョブです)。
- で IAM ユーザーを作成し AWS アカウント、そのユーザーに Snow Family デバイスを注文するジョブを作成するアクセス許可を付与すると、そのユーザーは Snow Family デバイスを注文する

ジョブを作成できます。ただし AWS アカウント、ユーザーが属する がジョブリソースを所有します。

- でジョブを作成するアクセス許可 AWS アカウント を持つ IAM ロールを作成すると、ロールを引き受けることのできるすべてのユーザーが、Snow Family デバイスを注文するジョブを作成できます。ロールが属 AWS アカウントする がジョブリソースを所有します。

でのリソースへのアクセスの管理 AWS クラウド

アクセス権限ポリシー では、誰が何にアクセスできるかを記述します。以下のセクションで、アクセス許可ポリシーを作成するために使用可能なオプションについて説明します。

Note

このセクションでは、 のコンテキストでの IAM の使用について説明します AWS Snowball。これは、IAM サービスに関する詳細情報を取得できません。完全な IAM ドキュメンテーションについては、「IAM ユーザーガイド」の「[IAM とは](#)」を参照してください。IAM ポリシー構文の詳細と説明については、「IAM ユーザーガイド」の「[AWS IAM ポリシーリファレンス](#)」を参照してください。

IAM アイデンティティにアタッチされたポリシーはアイデンティティベースのポリシー (IAM ポリシー) と呼ばれ、リソースにアタッチされたポリシーはリソースベースのポリシーと呼ばれます。はアイデンティティベースのポリシー (IAM ポリシー) のみ AWS Snowball をサポートします。

トピック

- [リソースベースのポリシー](#)

リソースベースのポリシー

Amazon S3 などの他のサービスでは、リソースベースの許可ポリシーもサポートされています。例えば、S3 バケットにポリシーをアタッチして、そのバケットへのアクセス許可を管理できます。AWS Snowball はリソースベースのポリシーをサポートしていません。

ポリシー要素 (アクション、効果、プリンシパル) の指定

各ジョブで ([リソースおよびオペレーション](#) を参照)、サービスは一連の API オペレーション ([AWS Snowball API リファレンス](#)を参照) を定義してそのジョブを作成、管理します。これらの API オペレーションのアクセス許可を付与するために、 はポリシーで指定できる一連のア

アクション AWS Snowball を定義します。たとえば、ジョブの場合、次のアクションが定義されます。CreateJob、CancelJob、および DescribeJob。API オペレーションを実行する場合に、複数のアクションで権限が必要となる場合があることに注意してください。

最も基本的なポリシーの要素を次に示します。

- リソース – ポリシーで Amazon リソースネーム (ARN) を使用して、ポリシーを適用するリソースを識別します。詳細については、「[リソースおよびオペレーション](#)」を参照してください。

Note

これは、Amazon S3、Amazon EC2、AWS Lambda AWS KMS、およびその他の多くのサービスでサポートされています。

Snowball では、IAM ポリシーステートメントの Resource 要素でのリソース ARN の指定はサポートされていません。Snowball へのアクセスを許可するには、ポリシーで “Resource”: “*” を指定します。

- アクション – アクションキーワードを使用して、許可または拒否するリソース操作を特定します。たとえば、指定した Effect に応じて、snowball:* は、すべてのオペレーションの実行をユーザーに許可または拒否します。

Note

これは、Amazon EC2、Amazon S3、および IAM でサポートされています。

- 効果 – ユーザーが特定のアクションを要求する際の効果を指定します。許可または拒否のいずれかになります。リソースへのアクセスを明示的に付与 (許可) していない場合、アクセスは暗黙的に拒否されます。また、明示的にリソースへのアクセスを拒否すると、別のポリシーによってアクセスが許可されている場合でも、ユーザーはそのリソースにアクセスできなくなります。

Note

これは、Amazon EC2、Amazon S3、および IAM でサポートされています。

- プリンシパル – ID ベースのポリシー (IAM ポリシー) で、ポリシーがアタッチされているユーザーが黙示的なプリンシパルとなります。リソースベースのポリシーでは、アクセス許可を受け取るユーザー、アカウント、サービス、またはその他のエンティティを指定します (リソースベースのポリシーにのみ適用されます)。リソースベースのポリシー AWS Snowball はサポートされていません。

IAM ポリシーの構文と記述の詳細については、「IAM ユーザーガイド」の「[AWS IAM ポリシーリファレンス](#)」を参照してください。

すべての AWS Snowball API アクションを示す表については、「」を参照してください [AWS Snowball API アクセス許可: アクション、リソース、および条件リファレンス](#)。

ポリシーでの条件の指定

許可を付与するとき、IAM ポリシー言語を使用して、ポリシーが有効になる必要がある条件を指定できます。例えば、特定の日付の後にのみ適用されるポリシーが必要になる場合があります。ポリシー言語での条件の指定の詳細については、「IAM ユーザーガイド」の「[条件](#)」を参照してください。

条件を表すには、あらかじめ定義された条件キーを使用します。AWS Snowballに固有の条件キーはありません。ただし、必要に応じて使用できる AWS 全体の条件キーがあります。AWS 全体のキーの完全なリストについては、IAM ユーザーガイドの「[条件に使用可能なキー](#)」を参照してください。

でのアイデンティティベースのポリシー (IAM ポリシー) の使用 AWS Snowball

このトピックでは、アカウント管理者が IAM アイデンティティ (ユーザー、グループ、ロール) にアクセス許可ポリシーをアタッチする方法を示す、アイデンティティベースのポリシーの例について説明します。これらのポリシーは、の AWS Snowball リソースに対してオペレーションを実行するアクセス許可を付与します AWS クラウド。

Important

初めに、AWS Snowball リソースへのアクセスを管理するための基本概念と使用可能なオプションについて説明する概要トピックをお読みになることをお勧めします。詳細については、「[のリソースへのアクセス許可の管理の概要 AWS クラウド](#)」を参照してください。

このセクションでは、次のトピックを対象としています。

- [AWS Snowball コンソールを使用するために必要なアクセス許可](#)
- [AWS Snowball Edge の管理 \(事前定義\) ポリシー](#)
- [お客様が管理するポリシーの例](#)

以下に示しているのは、アクセス許可ポリシーの例です。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "s3:GetBucketLocation",
        "s3:GetObject",
        "s3:ListBucket"
      ],
      "Resource": "arn:aws:s3::*:*"
    },
    {
      "Effect": "Allow",
      "Action": [
        "snowball:*",
        "importexport:*"
      ],
      "Resource": "*"
    }
  ]
}
```

このポリシーには以下の2つのステートメントがあります。

- 最初のステートメントは、arn:aws:s3::*:* のAmazon リソースネーム (ARN) を使用して、すべての Amazon S3 バケットで3つの Amazon S3 のアクション (s3:GetBucketLocation、s3:GetObject、および s3:ListBucket) を実行するためのアクセス許可を付与します。ARN でワイルドカード (*) が指定されているため、ユーザーは任意の、またはすべての Amazon S3 バケットを選択してデータをエクスポートできます。
- 2番目のステートメントは、すべての AWS Snowball アクションに対するアクセス許可を付与します。これらのアクションではリソースレベルのアクセス権限はサポートされていないため、ポリシーでワイルドカード文字 (*) が指定され、Resource の値にもワイルドカード文字が指定されます。

ID ベースのポリシーでアクセス許可を得るプリンシパルを指定していないため、ポリシーでは Principal 要素を指定していません。ユーザーにポリシーをアタッチすると、そのユーザーが暗黙のプリンシパルになります。IAM ロールにアクセス権限ポリシーをアタッチすると、ロールの信頼ポリシーで識別されたプリンシパルがアクセス権限を得ることになります。

すべての AWS Snowball ジョブ管理 API アクションとそれらが適用されるリソースを示す表については、「」を参照してください[AWS Snowball API アクセス許可: アクション、リソース、および条件リファレンス](#)。

AWS Snowball コンソールを使用するために必要なアクセス許可

アクセス許可リファレンステーブルには、AWS Snowball ジョブ管理 API オペレーションと、各オペレーションに必要なアクセス許可が一覧表示されます。ジョブ管理 API オペレーションの詳細については、「[AWS Snowball API アクセス許可: アクション、リソース、および条件リファレンス](#)」を参照してください。

を使用するには AWS Snow ファミリーマネジメントコンソール、次のアクセス許可ポリシーに示すように、追加のアクションのアクセス許可を付与する必要があります。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "s3:GetBucketLocation",
        "s3:GetBucketPolicy",
        "s3:ListBucket",
        "s3:ListBucketMultipartUploads",
        "s3:ListAllMyBuckets"
      ],
      "Resource": "arn:aws:s3:::*"
    },
    {
      "Effect": "Allow",
      "Action": [
        "s3:CreateBucket",
        "s3:PutObject",
        "s3:AbortMultipartUpload",
        "s3:ListMultipartUploadParts",
        "s3:PutObjectAcl"
      ],
      "Resource": "arn:aws:s3:::*"
    },
    {
      "Effect": "Allow",
      "Action": [
        "lambda:GetFunction",
```

```
        "lambda:GetFunctionConfiguration"
    ],
    "Resource": "arn:aws:lambda:*::function:*"
},
{
    "Effect": "Allow",
    "Action": [
        "lambda:ListFunctions"
    ],
    "Resource": "*"
},
{
    "Effect": "Allow",
    "Action": [
        "kms:CreateGrant",
        "kms:GenerateDataKey",
        "kms:Decrypt",
        "kms:Encrypt",
        "kms:RetireGrant",
        "kms:ListKeys",
        "kms:DescribeKey",
        "kms:ListAliases"
    ],
    "Resource": [
        "*"
    ]
},
{
    "Effect": "Allow",
    "Action": [
        "iam:AttachRolePolicy",
        "iam:CreatePolicy",
        "iam:CreateRole",
        "iam:ListRoles",
        "iam:ListRolePolicies",
        "iam:PutRolePolicy"
    ],
    "Resource": [
        "*"
    ]
},
{
    "Effect": "Allow",
    "Action": "iam:PassRole",
```

```
    "Resource": "*",
    "Condition": {
      "StringEquals": {
        "iam:PassedToService": "importexport.amazonaws.com"
      }
    }
  },
  {
    "Effect": "Allow",
    "Action": [
      "ec2:DescribeImages",
      "ec2:ModifyImageAttribute"
    ],
    "Resource": [
      "*"
    ]
  },
  {
    "Effect": "Allow",
    "Action": [
      "sns:CreateTopic",
      "sns:ListTopics",
      "sns:GetTopicAttributes",
      "sns:SetTopicAttributes",
      "sns:ListSubscriptionsByTopic",
      "sns:Subscribe"
    ],
    "Resource": [
      "*"
    ]
  },
  {
    "Effect": "Allow",
    "Action": [
      "greengrass:getServiceRoleForAccount"
    ],
    "Resource": [
      "*"
    ]
  },
  {
    "Effect": "Allow",
    "Action": [
      "snowball:*"
    ]
  }
}
```

```
    ],
    "Resource": [
        "*"
    ]
}
]
```

AWS Snowball コンソールには、次の理由でこれらの追加のアクセス許可が必要です。

- **ec2:** - ユーザーは、Amazon EC2 互換インスタンスを記述し、ローカルコンピューティングの目的でそれらの属性を変更できます。詳細については、「[Amazon EC2 互換コンピューティングインスタンスの使用](#)」を参照してください。
- **kms:** - ユーザーは、データを暗号化する KMS キーを作成または選択できます。詳細については、「[AWS Key Management ServiceAWS Snowball Edge の](#)」を参照してください。
- **iam:** - これにより、ジョブの作成と処理に関連する AWS リソースにアクセスするために AWS Snowball が引き受ける IAM ロール ARN を作成または選択できます。
- **sns:** - ユーザーは、作成したジョブの Amazon SNS 通知を作成または選択できます。詳細については、「[Snow Family デバイスの通知](#)」を参照してください。

AWSAWS Snowball Edge の管理 (事前定義) ポリシー

AWS は、によって作成および管理されるスタンドアロン IAM ポリシーを提供することで、多くの一般的なユースケースに対処します AWS。マネージドポリシーは、一般的ユースケースに必要な許可を付与することで、どの許可が必要なのかをユーザーが調査する必要をなくすることができます。詳細については、「IAM ユーザーガイド」の「[AWS マネージドポリシー](#)」を参照してください。

では、次の AWS管理ポリシーを使用できます AWS Snowball。

Snowball Edge 用の IAM ロールポリシーの作成

IAM ロールポリシーは、Amazon S3 バケットに対する読み取り/書き込みのアクセス許可を付与して作成する必要があります。また、IAM ロールには、Snowball との信頼関係も必要です。信頼関係を持つことは、データのインポートとエクスポートのどちらを行うかに応じて、が Snowball と Amazon S3 バケットにデータを書き込む AWS ことができることを意味します。

で Snow Family デバイスを注文するジョブを作成する場合 AWS Snow ファミリーマネジメントコンソール、必要な IAM ロールの作成は、アクセス許可セクションのステップ 4 で行われます。これは自動プロセスです。Snowball に許可した IAM ロールは、お客様の転送データが入った Snowball が

AWSに到着したときに、バケットにデータを書き込むためにのみ使用されます。このプロセスを以下の手順で示します。

インポートジョブ用の IAM ロールを作成するには

1. にサインイン AWS Management Console し、 <https://console.aws.amazon.com/importexport/> で AWS Snowball コンソールを開きます。
2. [ジョブの作成] を選択します。
3. 最初のステップで、Amazon S3 へのインポートジョブの詳細を入力してから、[Next (次へ)] を選択します。
4. 2 番目のステップでは、[Permission] で、[Create/Select IAM Role] を選択します。

IAM マネジメントコンソールが開き、指定した Amazon S3 バケットにオブジェクトをコピーするために AWS で使用する IAM ロールが表示されます。

5. このページで詳細を確認し、[許可] を選択します。

に戻ります。ここで AWS Snow ファミリーマネジメントコンソール、選択した IAM ロール ARN には、先ほど作成した IAM ロールの Amazon リソースネーム (ARN) が含まれます。

6. [Next] (次へ) を選択して、ロールの作成を終了します。

前述の手順により、データをインポート先となる Amazon S3 バケットへの書き込み権限を持つ IAM ロールが作成されます。作成する IAM ロールは、インポートジョブ用であるか、エクスポートジョブ用であるかによって、次のいずれかの構造になります。

インポートジョブ用の IAM ロール

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "s3:GetBucketLocation",
        "s3:ListBucketMultipartUploads"
      ],
      "Resource": "arn:aws:s3:::*"
    }
  ],
  {
```

```
    "Effect": "Allow",
    "Action": [
      "s3:GetBucketPolicy",
      "s3:PutObject",
      "s3:AbortMultipartUpload",
      "s3:ListMultipartUploadParts",
      "s3:PutObjectAcl",
      "s3:ListBucket",
      "s3:HeadBucket"
    ],
    "Resource": "arn:aws:s3:::*"
  }
]
```

AWS KMS マネージドキーによるサーバー側の暗号化 (SSE-KMS) を使用してインポートジョブに関連付けられた Amazon S3 バケットを暗号化する場合は、IAM ロールに次のステートメントも追加する必要があります。

```
{
  "Effect": "Allow",
  "Action": [
    "kms:GenerateDataKey"
  ],
  "Resource": "arn:aws:kms:us-west-2:123456789012:key/abc123a1-abcd-1234-efgh-111111111111"
}
```

オブジェクトのサイズが大きい場合、インポートプロセスに使用される Amazon S3 クライアントはマルチパートアップロードを使用します。SSE-KMS を使用してマルチパートアップロードを開始すると、アップロードされたすべてのパートは指定された AWS KMS キーを使用して暗号化されます。パートは暗号化されているため、マルチパートアップロードを完了するために組み立てる前に、パートを復号する必要があります。そのため、SSE-KMS を使用して Amazon S3 へのマルチパートアップロードを実行するときに、AWS KMS キー (kms:Decrypt) を復号化するアクセス許可が必要です。

以下に示しているのは、kms:Decrypt アクセス許可が求められるインポートジョブに必要な IAM ロールの例です。

```
{
  "Effect": "Allow",
```



```
"Action": [
    "kms:GenerateDataKey","kms:Decrypt"
],
"Resource": "arn:aws:kms:us-west-2:123456789012:key/abc123a1-abcd-1234-efgh-111111111111"
}
```

以下に示しているのは、エクスポートジョブに必要な IAM ロールの例です。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "s3:GetBucketLocation",
        "s3:GetBucketPolicy",
        "s3:GetObject",
        "s3:ListBucket"
      ],
      "Resource": "arn:aws:s3:::*"
    }
  ]
}
```

AWS KMS マネージドキーでサーバー側の暗号化を使用してエクスポートジョブに関連付けられた Amazon S3 バケットを暗号化する場合は、IAM ロールに次のステートメントも追加する必要があります。

```
{
  "Effect": "Allow",
  "Action": [
    "kms:Decrypt"
  ],
  "Resource": "arn:aws:kms:us-west-2:123456789012:key/abc123a1-abcd-1234-efgh-111111111111"
}
```

独自のカスタム IAM ポリシーを作成して、AWS Snowball ジョブ管理用の API オペレーションのアクセス許可を許可できます。これらのカスタムポリシーは、それらのアクセス許可が必要な IAM ユーザーまたはグループにアタッチできます。

お客様が管理するポリシーの例

このセクションでは、さまざまな AWS Snowball ジョブ管理アクションのアクセス許可を付与するユーザーポリシーの例を示します。これらのポリシーは、AWS SDK または AWS CLI を使用しているときに機能します。コンソールを使用している場合は、「[AWS Snowball コンソールを使用するために必要なアクセス許可](#)」で説明しているコンソールに固有の追加のアクセス権限を付与する必要があります。

Note

すべての例で、us-west-2 リージョンを使用し、架空のアカウント ID を含めています。

例

- [例 1: ユーザーが API を使用して Snow Family デバイスを注文するためのジョブの作成を許可するロールポリシー](#)
- [例 2: インポートジョブを作成するためのロールポリシー](#)
- [例 3: エクスポートジョブを作成するためのロールポリシー](#)
- [例 4: 期待されるロールのアクセス許可と信頼ポリシー](#)
- [AWS Snowball API アクセス許可: アクション、リソース、および条件リファレンス](#)

例 1: ユーザーが API を使用して Snow Family デバイスを注文するためのジョブの作成を許可するロールポリシー

以下のアクセス権限ポリシーは、ジョブ管理 API を使用したジョブまたはクラスター作成のアクセス権限を付与するために使用される、すべてのポリシーで必須の構成要素です。このステートメントは、Snowball IAM ロールの信頼関係ポリシーステートメントとして必要です。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "importexport.amazonaws.com"
      },
      "Action": "sts:AssumeRole"
    }
  ]
}
```

```
}  
]  
}
```

例 2: インポートジョブを作成するためのロールポリシー

次のロール信頼ポリシーを使用して、AWS IoT Greengrass 関数 AWS Lambda を搭載した を使用する Snowball Edge のインポートジョブを作成します。

```
{  
  "Version": "2012-10-17",  
  "Statement": [  
    {  
      "Effect": "Allow",  
      "Action": [  
        "s3:GetBucketLocation",  
        "s3:ListBucketMultipartUploads"  
      ],  
      "Resource": "arn:aws:s3:::*"  
    },  
    {  
      "Effect": "Allow",  
      "Action": [  
        "s3:GetBucketPolicy",  
        "s3:GetBucketLocation",  
        "s3:ListBucketMultipartUploads",  
        "s3:ListBucket",  
        "s3:HeadBucket",  
        "s3:PutObject",  
        "s3:AbortMultipartUpload",  
        "s3:ListMultipartUploadParts",  
        "s3:PutObjectAcl",  
        "s3:GetObject"  
      ],  
      "Resource": "arn:aws:s3:::*"  
    },  
    {  
      "Effect": "Allow",  
      "Action": [  
        "snowball:*"  
      ],  
      "Resource": [  

```

```
        "*"
    ]
},
{
    "Effect": "Allow",
    "Action": [
        "iot:AttachPrincipalPolicy",
        "iot:AttachThingPrincipal",
        "iot:CreateKeysAndCertificate",
        "iot:CreatePolicy",
        "iot:CreateThing",
        "iot:DescribeEndpoint",
        "iot:GetPolicy"
    ],
    "Resource": [
        "*"
    ]
},
{
    "Effect": "Allow",
    "Action": [
        "lambda:GetFunction"
    ],
    "Resource": [
        "*"
    ]
},
{
    "Effect": "Allow",
    "Action": [
        "greengrass:CreateCoreDefinition",
        "greengrass:CreateDeployment",
        "greengrass:CreateDeviceDefinition",
        "greengrass:CreateFunctionDefinition",
        "greengrass:CreateGroup",
        "greengrass:CreateGroupVersion",
        "greengrass:CreateLoggerDefinition",
        "greengrass:CreateSubscriptionDefinition",
        "greengrass:GetDeploymentStatus",
        "greengrass:UpdateGroupCertificateConfiguration",
        "greengrass:CreateGroupCertificateAuthority",
        "greengrass:GetGroupCertificateAuthority",
        "greengrass:ListGroupCertificateAuthorities",
        "greengrass:ListDeployments",
```

```
        "greengrass:GetGroup",
        "greengrass:GetGroupVersion",
        "greengrass:GetCoreDefinitionVersion"
    ],
    "Resource": [
        "*"
    ]
}
]
```

例 3: エクスポートジョブを作成するためのロールポリシー

次のロール信頼ポリシーを使用して、AWS IoT Greengrass 関数 AWS Lambda を搭載した を使用する Snowball Edge のエクスポートジョブを作成します。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "s3:GetBucketLocation",
        "s3:GetObject",
        "s3:ListBucket"
      ],
      "Resource": "arn:aws:s3:::*"
    },
    {
      "Effect": "Allow",
      "Action": [
        "snowball:*"
      ],
      "Resource": [
        "*"
      ]
    },
    {
      "Effect": "Allow",
      "Action": [
```

```
        "iot:AttachPrincipalPolicy",
        "iot:AttachThingPrincipal",
        "iot:CreateKeysAndCertificate",
        "iot:CreatePolicy",
        "iot:CreateThing",
        "iot:DescribeEndpoint",
        "iot:GetPolicy"
    ],
    "Resource": [
        "*"
    ]
},
{
    "Effect": "Allow",
    "Action": [
        "lambda:GetFunction"
    ],
    "Resource": [
        "*"
    ]
},
{
    "Effect": "Allow",
    "Action": [
        "greengrass:CreateCoreDefinition",
        "greengrass:CreateDeployment",
        "greengrass:CreateDeviceDefinition",
        "greengrass:CreateFunctionDefinition",
        "greengrass:CreateGroup",
        "greengrass:CreateGroupVersion",
        "greengrass:CreateLoggerDefinition",
        "greengrass:CreateSubscriptionDefinition",
        "greengrass:GetDeploymentStatus",
        "greengrass:UpdateGroupCertificateConfiguration",
        "greengrass:CreateGroupCertificateAuthority",
        "greengrass:GetGroupCertificateAuthority",
        "greengrass:ListGroupCertificateAuthorities",
        "greengrass:ListDeployments",
        "greengrass:GetGroup",
        "greengrass:GetGroupVersion",
        "greengrass:GetCoreDefinitionVersion"
    ],
    "Resource": [
        "*"
    ]
}
```

```
    ]
  }
]
}
```

例 4: 期待されるロールのアクセス許可と信頼ポリシー

既存のサービスロールを使用するには、以下の期待されるロールのアクセス許可ポリシーが必要です。これは 1 回限りの設定です。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "sns:Publish",
      "Resource": ["[[snsArn]]"]
    },
    {
      "Effect": "Allow",
      "Action": [
        "cloudwatch:ListMetrics",
        "cloudwatch:GetMetricData",
        "cloudwatch:PutMetricData"
      ],
      "Resource": [
        "*"
      ],
      "Condition": {
        "StringEquals": {
          "cloudwatch:namespace": "AWS/SnowFamily"
        }
      }
    }
  ]
}
```

既存のサービスロールを使用するには、以下の期待されるロールの信頼ポリシーが必要です。これは 1 回限りの設定です。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "importexport.amazonaws.com"
      },
      "Action": "sts:AssumeRole"
    }
  ]
}
```

AWS Snowball API アクセス許可: アクション、リソース、および条件リファレンス

[アクセスコントロール AWS クラウド](#) をセットアップし、IAM アイデンティティにアタッチできるアクセス権限ポリシー (アイデンティティベースのポリシー) を作成するときは、以下のリストをリファレンスとして使用できます。には、各 AWS Snowball ジョブ管理 API オペレーションと、アクションを実行するためのアクセス許可を付与できる対応するアクションが含まれています。また、アクセス許可を付与できる AWS リソースの API オペレーションごとにも含まれます。ポリシーの Action フィールドでアクションを指定し、ポリシーの Resource フィールドでリソースの値を指定します。

AWS Snowball ポリシーで AWS 全体の条件キーを使用して条件を表現できます。AWS 全体のキーの完全なリストについては、「IAM ユーザーガイド」の「[使用可能なキー](#)」を参照してください。

Note

アクションを指定するには、API オペレーション名 (snowball:CreateJob など) の前に snowball: プレフィックスを使用します。

AWS Snowballでのログ記録とモニタリング

モニタリングは、および AWS ソリューションの信頼性、可用性、パフォーマンスを維持する AWS Snowball 上で重要な部分です。マルチポイント障害が発生した場合は、その障害をより簡単にデバッグできるように、モニタリングデータを収集する必要があります。AWS には、AWS Snowball 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の複数のツールが用意されています。

AWS CloudTrail ログ

CloudTrail は、AWS Snowball ジョブ管理 API または AWS コンソールの使用時に、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。によって収集された情報を使用して CloudTrail、AWS Snowball サービスに対して行われた API リクエスト、リクエスト元の IP アドレス、リクエスト者、リクエスト日時などの詳細を確認できます。詳細については、「[を使用した AWS Snowball Edge API コールのログ記録 AWS CloudTrail](#)」を参照してください。

のコンプライアンス検証 AWS Snowball

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

を使用して、サードパーティ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、「[でのレポートのダウンロード AWS Artifact](#)」の」を参照してください。

を使用する際のお客様のコンプライアンス責任 AWS のサービスは、お客様のデータの機密性、貴社のコンプライアンス目的、適用される法律および規制によって決まります。は、コンプライアンスに役立つ以下のリソース AWS を提供しています。

- [セキュリティとコンプライアンスのクイックスタートガイド](#) – これらのデプロイガイドでは、アーキテクチャ上の考慮事項について説明し、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 AWS を にデプロイする手順について説明します。
- [アマゾン ウェブ サービスにおける HIPAA セキュリティとコンプライアンスのアーキテクチャー](#) – このホワイトペーパーでは、企業が AWS を使用して HIPAA 対象アプリケーションを作成する方法について説明します。

Note

すべて AWS のサービス HIPAA の対象となるわけではありません。詳細については、「[HIPAA 対応サービスのリファレンス](#)」を参照してください。

- [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、お客様の業界や地域に適用される場合があります。

- [AWS カスタマーコンプライアンスガイド](#) – コンプライアンスの観点から責任共有モデルを理解します。このガイドでは、ガイダンスを保護し AWS のサービス、複数のフレームワーク (米国国立標準技術研究所 (NIST)、Payment Card Industry Security Standards Council (PCI)、国際標準化機構 (ISO) を含む) のセキュリティコントロールにマッピングするためのベストプラクティスをまとめています。
- 「[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](#) – これにより AWS のサービス、内のセキュリティ状態を包括的に確認できます AWS。Security Hub では、セキュリティコントロールを使用して AWS リソースを評価し、セキュリティ業界標準とベストプラクティスに対するコンプライアンスをチェックします。サポートされているサービスとコントロールのリストについては、「[Security Hub のコントロールリファレンス](#)」を参照してください。
- [Amazon GuardDuty](#) – これにより AWS アカウント、疑わしいアクティビティや悪意のあるアクティビティがないか環境を監視することで、、、ワークロード、コンテナ、データに対する潜在的な脅威 AWS のサービスを検出します。GuardDuty は、特定のコンプライアンスフレームワークで義務付けられている侵入検知要件を満たすことで、PCI DSS などのさまざまなコンプライアンス要件への対応に役立ちます。
- [AWS Audit Manager](#) – これにより AWS のサービス、AWS 使用状況を継続的に監査し、リスクの管理方法と規制や業界標準への準拠を簡素化できます。

耐障害性

AWS グローバルインフラストラクチャは、AWS リージョン およびアベイラビリティゾーンを中心に構築されています。は、低レイテンシー、高スループット、および高度に冗長なネットワークで接続された、物理的に分離および分離された複数のアベイラビリティゾーン AWS リージョンを提供します。アベイラビリティゾーンでは、アベイラビリティゾーン間で中断せずに、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、耐障害性、およびスケーラビリティが優れています。

AWS リージョン およびアベイラビリティ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。

のインフラストラクチャセキュリティ AWS Snowball

マネージドサービスである AWS Snow Family 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、「Security Pillar AWS Well-Architected Framework」の [「Infrastructure Protection」](#) を参照してください。

が AWS 公開している API コールを使用して、ネットワーク AWS Snow Family 経由で 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 は必須で TLS 1.3 がお勧めです。
- 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。

Snowball Edge ジョブのデータ検証

次に、AWS Snowball Edge がデータ転送を検証する方法と、ジョブ中およびジョブ後にデータの整合性を確保するために実行できる手動手順について説明します。

トピック

- [転送されたデータのチェックサムの検証](#)
- [Snowball 転送中のローカルインベントリの作成](#)
- [一般的な検証エラー](#)
- [Amazon S3 にデータをインポートした後、Snowball Edge を手動検証する](#)

転送されたデータのチェックサムの検証

Amazon S3 インターフェイスを使用してローカルのデータソースから Snowball Edge にファイルをコピーする際、多数のチェックサムが作成されます。これらのチェックサムを使用して、転送中のデータが自動的に検証されます。

おおまかには、これらのチェックサムはファイルごと (または大きなファイルの部分ごと) に作成されます。Snowball Edge の場合、デバイスのバケットに対して次の AWS CLI コマンドを実行すると、これらのチェックサムが表示されます。チェックサムは、転送プロセス全体でデータの整合性を検証し、データが正しくコピーされたことを確認するために使用されます。

```
aws s3api list-objects --bucket bucket-name --endpoint http://ip:8080 --profile edge-profile
```

これらのチェックサムが一致しない場合、関連データは Amazon S3 にインポートされません。

Snowball 転送中のローカルインベントリの作成

Amazon S3 アダプターまたは CLI を使用する場合は、Snowball にコピーされたファイルのローカルインベントリを作成します。ローカルインベントリの内容を使用して、ローカルストレージまたはサーバーにある内容と比較できます。

例えば、などです

```
aws s3 cp folder/ s3://bucket --recursive > inventory.txt
```

一般的な検証エラー

検証エラーが発生すると、対応するデータ (ファイルまたは大きなファイルの一部) は送信先に書き込まれません。検証エラーの一般的な原因は以下のとおりです。

- シンボリックリンクのコピーしようとした。
- アクティブに変更中のファイルをコピーしようとした。試行がチェックサム検証に失敗し、失敗した転送としてマークされた。
- 5 TB を超えるサイズのファイルをコピーしようとした。
- 2 GiB を超えるパートサイズをコピーしようとした。
- データストレージ容量がすでにいっぱいになっている Snowball Edge デバイスにファイルをコピーしようとした。
- Amazon S3 の [オブジェクトキーの命名のガイドライン](#) に従っていない Snowball Edge デバイスに、ファイルをコピーしようとした。

これらの検証エラーのいずれか 1 つが発生すると、ログに記録されます。検証に失敗したファイルとその理由は、ステップに従って手動で特定できます。詳細については、「[Amazon S3 にデータをインポートした後、Snowball Edge を手動検証する](#)」を参照してください。

Amazon S3 にデータをインポートした後、Snowball Edge を手動検証する

インポートジョブの完了後に Amazon S3 内のデータを手動検証するいくつかのオプションについて以下に説明します。

ジョブ完了レポートおよび関連ログの確認

データが Amazon S3 でインポートされたりエクスポートされたりするたびに、ダウンロード可能な PDF 形式のジョブレポートが生成されます。インポートジョブの場合、このレポートはインポートプロセスの最後に生成されます。詳細については、「[ジョブ完了レポートとログの取得](#)」を参照してください。

S3 インベントリ

複数のジョブで大量のデータを Amazon S3 内に転送した場合、ジョブ完了レポートを個別に確認することは効率的ではありません。代わりに、1 つまたは複数の Amazon S3 バケット内の、すべて

のオブジェクトのインベントリを取得できます。Amazon S3 インベントリでは、オブジェクトおよび対応するメタデータが日単位または週単位で、カンマ区切り値 (CSV) ファイルに表示されます。このファイルでは、Amazon S3 バケットのオブジェクトまたは共有プレフィックスのオブジェクト (名前が共通の文字列で始まるオブジェクト) が対象となります。

データをインポートした先の Amazon S3 バケットのインベントリを取得すると、このインベントリを、ソースデータの転送元のファイルと容易に比較できます。この方法により、どの場所にあるどのファイルが転送されなかったかをすばやく確認できます。

Amazon S3 sync コマンドの使用

ワークステーションがインターネットに接続できる場合は、AWS CLI コマンドを実行して、転送されたすべてのファイルの最終検証を実行できます `aws s3 sync`。このコマンドは、ディレクトリと S3 プレフィックスを同期させます。新しいファイルおよび更新されたファイルを送信元ディレクトリから送信先に再帰的にコピーします。詳細については、AWS CLI コマンドリファレンスの[同期](#)を参照してください。

Important

このコマンドの送信先としてローカルストレージを指定する場合は、同期先のファイルのバックアップを必ず作成してください。これらのファイルは、指定した Amazon S3 の送信元のコンテンツで上書きされます。

Snow Family デバイスの通知

Snow が Amazon SNS を使用する仕組み

Snow デバイスは、Amazon Simple Notification Service (Amazon SNS) で配信される堅牢な通知機能を活用できるように設計されています。Snow デバイスを注文するジョブを作成するときに、ジョブのステータス変更の通知を受け取る E メールアドレスを指定できます。その際、新しい Amazon SNS トピックを作成することも、既存のトピックを使用することもできます。SNS トピックが暗号化されている場合は、トピックに対するカスターマネージド KMS 暗号化を有効にし、カスターマネージド KMS キーポリシーを設定する必要があります。[通知設定を選択する](#) を参照してください。

ジョブを作成すると、Amazon SNS 通知を取得するために指定したすべての E メールアドレスに、トピックサブスクリプションの確認を求める E メールメッセージが AWS 通知から送信されます。その E メールアカウントのユーザーは [サブスクリプションを確認] を選択してサブスクリプションを確定する必要があります。Amazon SNS の通知 E メールはジョブの各ステータスに対して作成され、[AWS Snow ファミリーマネジメントコンソール](#) へのリンクが記載されています。

Amazon SNS コンソールから、ステータス変更通知のテキストメッセージを送信するように Amazon SNS を設定することもできます。詳細については、Amazon Simple Notification Service デベロッパーガイドの「[Mobile text messaging \(SMS\)](#)」を参照してください。

Snow ジョブのステータス変更の SNS トピックを暗号化する

Snow ジョブのステータス変更通知の SNS トピックでカスターマネージド KMS 暗号化を有効にします。Snow インポート IAM ロールは マネージド AWS KMS キーにアクセスして Decrypt および GenerateDataKey アクションを実行できないため、AWS マネージド暗号化で暗号化された SNS トピックは Snow ジョブのステータス変更を受信できません。さらに、AWS マネージド KMS キーのポリシーは編集できません。

Amazon SNS マネジメントコンソールを使用して Amazon SNS トピックのサーバー側の暗号化を有効にするには

1. にサインイン AWS Management Console し、<https://console.aws.amazon.com/sns/v3/home> で Amazon SNS コンソールを開きます。
2. ナビゲーションペインで、[トピック] を選択します。

3. [トピック] ページで、ジョブのステータス変更通知に使用するトピックを選択し、[編集] を選択します。
4. [暗号化] セクションを展開し、以下の操作を実行します。
 - a. [暗号化の有効化] を選択します。
 - b. AWS KMS キーを指定します。次を参照してください。
 - c. KMS タイプごとに、[説明]、[アカウント]、および [KMS ARN] が表示されます。
5. AWS アカウントからカスタムキーを使用するには、AWS KMS キーフィールドを選択し、リストからカスタム KMS kms を選択します。カスタム KMS の作成手順については、「AWS Key Management Service デベロッパーガイド」の「[キーの作成](#)」を参照してください。

アカウントまたは別の AWS アカウントのカスタム KMS ARN を使用するには AWS、KMS キーフィールドに AWS KMS キー ARN を入力します。
6. [変更を保存] を選択します。トピックに対するサーバー側の暗号化が有効になり、トピックページが表示されます。

カスタマーマネージド KMS キーポリシーの設定

Snow ジョブのステータス変更の通知を受け取る SNS トピックの暗号化を有効にしたら、SNS トピック暗号化の KMS ポリシーを更新し、"mks:Decrypt" および "mks:GenerateDataKey*" アクションに対する Snow サービスプリンシパル "importexport.amazonaws.com" を許可します。

KMS キーポリシーでインポート/エクスポートのサービスロールを許可するには

1. にサインイン AWS Management Console し、<https://console.aws.amazon.com/kms> で AWS Key Management Service (AWS KMS) コンソールを開きます。
2. を変更するには AWS リージョン、ページの右上隅にあるリージョンセレクターを使用します。
3. コンソールの右上隅で、コンソール AWS リージョン の を Snow デバイスの注文元と同じリージョンに変更します。
4. ナビゲーションペインで、[カスタマーマネージドキー] を選択します。
5. KMS キーのリストで、更新する KMS キーのエイリアスまたはキー ID を選択します。
6. [キーポリシー] タブを選択すると、キーポリシーステートメントで、キーポリシーによって KMS キーへのアクセス許可を付与されたプリンシパルと、それらが実行できるアクションを確認できます。

7. Snow サービスプリンシパル "importexport.amazonaws.com" には、"kms:Decrypt" および "kms:GenerateDataKey*" アクションの以下のポリシーステートメントを追加します。

```
{
  "Effect": "Allow",
  "Principal": {
    "Service": "service.amazonaws.com"
  },
  "Action": [
    "kms:Decrypt",
    "kms:GenerateDataKey"
  ],
  "Resource": "*",
  "Condition": {
    "ArnLike": {
      "aws:SourceArn": "arn:aws:service:region:customer-account-id:resource-type/customer-resource-id"
    }
  },
  "StringEquals": {
    "kms:EncryptionContext:aws:sns:topicArn": "arn:aws:sns:your_region:customer-account-id:your_sns_topic_name"
  }
}
```

8. [変更を保存] を選択して変更を適用し、ポリシーエディタを終了します。

SNS 通知の例

Amazon SNS 通知は、ジョブのステータスが変化すると次の E メールメッセージを生成します。これらのメッセージは Email-JSON SNS トピックプロトコルの一例です。

ジョブステータス	SNS 通知 JSON
Job created	<pre>{ "Type" : "Notification",</pre>

ジョブステータス

SNS 通知 JSON

```
"MessageId" : "dc1e94d9-56c5-5e9
6-808d-cc7f68faa162",
  "TopicArn" : "arn:aws:sns:us-ea
st-2:111122223333:ExampleTopic1",
  "Message" : "Your job Job-name
(JID8bca334a-6c2f-4cd0-97e2
-3f5a4dc9bd6d) has been created. More
info - https://console.aws.amazon.
com/importexport",
  "Timestamp" : "2023-02-23T00:27:
58.831Z",
  "SignatureVersion" : "1",
  "Signature" : "FMG5t1ZhJNHLHUXvZ
gtZz1k24FzVa7oX0T4P03neeXw8
ZEXZx6z35j2F0TuNYShn2h0bKNC/
zLTnMyIxEzmi2X1sh0BWsJHkrW2xkR58ABZ
F+4uWHEE73yDVR4SyYAIkP9jstZzDRm
+bcVs8+T0yaLiEGLrIIIL4esi11lhIkg
ErCuy5btPcWXBdio2fpCRD5x9oR
6gmE/rd5071X1c1uvnv4r1Lkk4pqP2/
iUfxFZva1xLSRvgyfm6D9hNk1VyPfy+7
Ta1MD01zmJu0rExtnSIbZew3foxgx8GT
+1bZkLd0ZdtRj1IyPRP44eyq78sU0Eo/
LsDr0Iak4ZDpg8dXg==",
  "SigningCertURL" : "https://
sns.us-east-1.amazonaws.com/
SimpleNotificationService-010a507c1
833636cd94bdb98bd93083a.pem",
  "UnsubscribeURL" : "https://
sns.us-east-2.amazonaws.com/?
Action=Unsubscribe&SubscriptionArn
=arn:aws:sns:us-east-2:1111
22223333:ExampleTopic1:e103
9402-24e7-40a3-a0d4-797da162b297"
}
```

ジョブステータス	SNS 通知 JSON
アプライアンスの準備	<pre>{ "Type" : "Notification", "MessageId" : "dc1e94d9-56c5-5e9 6-808d-cc7f68faa162", "TopicArn" : "arn:aws:sns:us-ea st-2:111122223333:ExampleTopic1", "Message" : "Your job Job-name (JID8bca334a-6c2f-4cd0-97e2 -3f5a4dc9bd6d) is being prepared. More info - https://console.aw s.amazon.com/importexport", "Timestamp" : "2023-02-23T00:27: 58.831Z", "SignatureVersion" : "1", "Signature" : "FMG5t1ZhJNHLHUXvZ gtZz1k24FzVa7oX0T4P03neeXw8 ZEXZx6z35j2F0TuNYShn2h0bKNC/ zLTnMyIxEzmi2X1sh0BWsJHkrW2xkr58ABZ F+4uWHEE73yDVR4SyYAIkP9jstZzDRm +bcVs8+T0yaLiEGLrIIIL4esi111hIkg ErCuy5btPcWBdio2fpCRD5x9oR 6gmE/rd5071X1c1uvnv4r1Lkk4pqP2/ iUfxFZva1xLSRvgyfm6D9hNk1VyPfy+7 Ta1MD01zmJu0rExtnSIbZew3foxgx8GT +1bZkLd0ZdtRj1IyPRP44eyq78sU0Eo/ LsDr0Iak4ZDpg8dXg==", "SigningCertURL" : "https:// sns.us-east-1.amazonaws.com/ SimpleNotificationService-010a507c1 833636cd94bdb98bd93083a.pem", "UnsubscribeURL" : "https:// sns.us-east-2.amazonaws.com/? Action=Unsubscribe&SubscriptionArn =arn:aws:sns:us-east-2:1111 22223333:ExampleTopic1:e103 9402-24e7-40a3-a0d4-797da162b297" }</pre>

ジョブステータス

SNS 通知 JSON

Exporting

```
{
  "Type" : "Notification",
  "MessageId" : "dc1e94d9-56c5-5e9
6-808d-cc7f68faa162",
  "TopicArn" : "arn:aws:sns:us-ea
st-2:111122223333:ExampleTopic1",
  "Message" : "Your job Job-name
(JID8bca334a-6c2f-4cd0-97e2
-3f5a4dc9bd6d) is being Exported.
More info - https://console.aw
s.amazon.com/importexport",
  "Timestamp" : "2023-02-23T00:27:
58.831Z",
  "SignatureVersion" : "1",
  "Signature" : "FMG5t1ZhJNHLHUXvZ
gtZz1k24FzVa7oX0T4P03neeXw8
ZEXZx6z35j2F0TuNYShn2h0bKNC/
zLTnMyIxEzmi2X1sh0BWsJHkrW2xkr58ABZ
F+4uWHEE73yDVR4SyYAIkP9jstZzDRm
+bcVs8+T0yaLiEGLrIIIL4esi111hIkG
ErCuy5btPcWXBdio2fpCRD5x9oR
6gmE/rd5071X1c1uvnv4r1Lkk4pqP2/
iUfxFZva1xLSRvgyfm6D9hNk1VyPfy+7
Ta1MD01zmJu0rExtnSIbZew3foxgx8GT
+1bZkLd0ZdtRj1IyPRP44eyq78sU0Eo/
LsDr0Iak4ZDpg8dXg==",
  "SigningCertURL" : "https://
sns.us-east-1.amazonaws.com/
SimpleNotificationService-010a507c1
833636cd94bdb98bd93083a.pem",
  "UnsubscribeURL" : "https://
sns.us-east-2.amazonaws.com/?
Action=Unsubscribe&SubscriptionArn
=arn:aws:sns:us-east-2:1111
22223333:ExampleTopic1:e103
9402-24e7-40a3-a0d4-797da162b297"
}
```

ジョブステータス	SNS 通知 JSON
In transit to you	<pre>{ "Type" : "Notification", "MessageId" : "dc1e94d9-56c5-5e9 6-808d-cc7f68faa162", "TopicArn" : "arn:aws:sns:us-ea st-2:111122223333:ExampleTopic1", "Message" : "Your job Job-name (JID8bca334a-6c2f-4cd0-97e2 -3f5a4dc9bd6d) is in transit to you. More info - https://console.aw s.amazon.com/importexport", "Timestamp" : "2023-02-23T00:27: 58.831Z", "SignatureVersion" : "1", "Signature" : "FMG5t1ZhJNHLHUXvZ gtZz1k24FzVa7oX0T4P03neeXw8 ZEXZx6z35j2F0TuNYShn2h0bKNC/ zLTnMyIxEzmi2X1sh0BWsJHkrW2xkr58ABZ F+4uWHEE73yDVR4SyYAIkP9jstZzDRm +bcVs8+T0yaLiEGLrIIIL4esi111hIkG ErCuy5btPcWXBdio2fpCRD5x9oR 6gmE/rd5071X1c1uvnv4r1Lkk4pqP2/ iUfxFZva1xLSRvgyfm6D9hNk1VyPfy+7 Ta1MD01zmJu0rExtN5IbZew3foxgx8GT +1bZkLd0ZdtRj1IyPRP44eyq78sU0Eo/ LsDr0Iak4ZDpg8dXg==", "SigningCertURL" : "https:// sns.us-east-1.amazonaws.com/ SimpleNotificationService-010a507c1 833636cd94bdb98bd93083a.pem", "UnsubscribeURL" : "https:// sns.us-east-2.amazonaws.com/? Action=Unsubscribe&SubscriptionArn =arn:aws:sns:us-east-2:1111 22223333:ExampleTopic1:e103 9402-24e7-40a3-a0d4-797da162b297" }</pre>

ジョブステータス

SNS 通知 JSON

Delivered to you

```
{
  "Type" : "Notification",
  "MessageId" : "dc1e94d9-56c5-5e9
6-808d-cc7f68faa162",
  "TopicArn" : "arn:aws:sns:us-ea
st-2:111122223333:ExampleTopic1",
  "Message" : "Your job Job-name
(JID8bca334a-6c2f-4cd0-97e2
-3f5a4dc9bd6d) was delivered to
you. More info - https://console.aw
s.amazon.com/importexport",
  "Timestamp" : "2023-02-23T00:27:
58.831Z",
  "SignatureVersion" : "1",
  "Signature" : "FMG5t1ZhJNHLHUXvZ
gtZz1k24FzVa7oX0T4P03neeXw8
ZEXZx6z35j2F0TuNYShn2h0bKNC/
zLTnMyIxEzmi2X1sh0BWsJHkrW2xkr58ABZ
F+4uWHEE73yDVR4SyYAIkP9jstZzDRm
+bcVs8+T0yaLiEGLrIIIL4esi111hIkG
ErCuy5btPcWXBdio2fpCRD5x9oR
6gmE/rd5071X1c1uvnv4r1Lkk4pqP2/
iUfxFZva1xLSRvgyfm6D9hNk1VyPfy+7
Ta1MD01zmJu0rExtnSIbZew3foxgx8GT
+1bZkLd0ZdtRj1IyPRP44eyq78sU0Eo/
LsDr0Iak4ZDpg8dXg==",
  "SigningCertURL" : "https://
sns.us-east-1.amazonaws.com/
SimpleNotificationService-010a507c1
833636cd94bdb98bd93083a.pem",
  "UnsubscribeURL" : "https://
sns.us-east-2.amazonaws.com/?
Action=Unsubscribe&SubscriptionArn
=arn:aws:sns:us-east-2:1111
22223333:ExampleTopic1:e103
9402-24e7-40a3-a0d4-797da162b297"
}
```

ジョブステータス	SNS 通知 JSON
への転送中 AWS	<pre>{ "Type" : "Notification", "MessageId" : "dc1e94d9-56c5-5e9 6-808d-cc7f68faa162", "TopicArn" : "arn:aws:sns:us-ea st-2:111122223333:ExampleTopic1", "Message" : "Your job Job-name (JID8bca334a-6c2f-4cd0-97e2 -3f5a4dc9bd6d) is in transit to AWS. More info - https://console.aw s.amazon.com/importexport", "Timestamp" : "2023-02-23T00:27: 58.831Z", "SignatureVersion" : "1", "Signature" : "FMG5t1ZhJNHLHUXvZ gtZz1k24FzVa7oX0T4P03neeXw8 ZEXZx6z35j2F0TuNYShn2h0bKNC/ zLTnMyIxEzmi2X1sh0BWsJHkrW2xkr58ABZ F+4uWHEE73yDVR4SyYAIkP9jstZzDRm +bcVs8+T0yaLiEGLrIIIL4esi111hIkG ErCuy5btPcWBdio2fpCRD5x9oR 6gmE/rd5071X1c1uvnv4r1Lkk4pqP2/ iUfxFZva1xLSRvgyfm6D9hNk1VyPfy+7 Ta1MD01zmJu0rExtnSIbZew3foxgx8GT +1bZkLd0ZdtRj1IyPRP44eyq78sU0Eo/ LsDr0Iak4ZDpg8dXg==", "SigningCertURL" : "https:// sns.us-east-1.amazonaws.com/ SimpleNotificationService-010a507c1 833636cd94bdb98bd93083a.pem", "UnsubscribeURL" : "https:// sns.us-east-2.amazonaws.com/? Action=Unsubscribe&SubscriptionArn =arn:aws:sns:us-east-2:1111 22223333:ExampleTopic1:e103 9402-24e7-40a3-a0d4-797da162b297" }</pre>

ジョブステータス

SNS 通知 JSON

仕分け施設にあります

```
{
  "Type" : "Notification",
  "MessageId" : "dc1e94d9-56c5-5e9
6-808d-cc7f68faa162",
  "TopicArn" : "arn:aws:sns:us-ea
st-2:111122223333:ExampleTopic1",
  "Message" : "Your job Job-name
(JID8bca334a-6c2f-4cd0-97e2
-3f5a4dc9bd6d) is at AWS sorting
facility. More info - https://
console.aws.amazon.com/impor
texport",
  "Timestamp" : "2023-02-23T00:27:
58.831Z",
  "SignatureVersion" : "1",
  "Signature" : "FMG5t1ZhJNHLHUXvZ
gtZz1k24FzVa7oX0T4P03neeXw8
ZEXZx6z35j2F0TuNYShn2h0bKNC/
zLTnMyIxEzmi2X1shOBWsJHkrW2xkR58ABZ
F+4uWHEE73yDVR4SyYAIkP9jstZzDRm
+bcVs8+T0yaLiEGLrIIIL4esi11lhIkG
ErCuy5btPcWXBdio2fpCRD5x9oR
6gmE/rd507lX1c1uvnv4r1Lkk4pqP2/
iUfxFZva1xLSRvgyfm6D9hNk1VyPfy+7
Ta1MD0lzmJu0rExtnSIbZew3foxgx8GT
+1bZkLd0ZdtdRJIyPRP44eyq78sU0Eo/
LsDr0Iak4ZDpg8dXg==",
  "SigningCertURL" : "https://
sns.us-east-1.amazonaws.com/
SimpleNotificationService-010a507c1
833636cd94bdb98bd93083a.pem",
  "UnsubscribeURL" : "https://
sns.us-east-2.amazonaws.com/?
Action=Unsubscribe&SubscriptionArn
=arn:aws:sns:us-east-2:1111
22223333:ExampleTopic1:e103
9402-24e7-40a3-a0d4-797da162b297"
}
```


ジョブステータス	SNS 通知 JSON
で AWS	<pre>{ "Type" : "Notification", "MessageId" : "dc1e94d9-56c5-5e9 6-808d-cc7f68faa162", "TopicArn" : "arn:aws:sns:us-ea st-2:111122223333:ExampleTopic1", "Message" : "Your job Job-name (JID8bca334a-6c2f-4cd0-97e2 -3f5a4dc9bd6d) is at AWS. More info - https://console.aws.amazon.com/ importexport", "Timestamp" : "2023-02-23T00:27: 58.831Z", "SignatureVersion" : "1", "Signature" : "FMG5t1ZhJNHLHUXvZ gtZz1k24FzVa7oX0T4P03neeXw8 ZEXZx6z35j2F0TuNYShn2h0bKNC/ zLTnMyIxEzmi2X1sh0BWsJHkrW2xkr58ABZ F+4uWHEE73yDVR4SyYAIkP9jstZzDRm +bcVs8+T0yaLiEGLrIIIL4esi111hIkG ErCuy5btPcWXBdio2fpCRD5x9oR 6gmE/rd5071X1c1uvnv4r1Lkk4pqP2/ iUfxFZva1xLSRvgyfm6D9hNk1VyPfy+7 Ta1MD01zmJu0rExtnSIbZew3foxgx8GT +1bZkLd0ZdtRj1IyPRP44eyq78sU0Eo/ LsDr0Iak4ZDpg8dXg==", "SigningCertURL" : "https:// sns.us-east-1.amazonaws.com/ SimpleNotificationService-010a507c1 833636cd94bdb98bd93083a.pem", "UnsubscribeURL" : "https:// sns.us-east-2.amazonaws.com/? Action=Unsubscribe&SubscriptionArn =arn:aws:sns:us-east-2:1111 22223333:ExampleTopic1:e103 9402-24e7-40a3-a0d4-797da162b297" }</pre>

ジョブステータス

SNS 通知 JSON

インポート中

```
{
  "Type" : "Notification",
  "MessageId" : "dc1e94d9-56c5-5e9
6-808d-cc7f68faa162",
  "TopicArn" : "arn:aws:sns:us-ea
st-2:111122223333:ExampleTopic1",
  "Message" : "Your job Job-name
(JID8bca334a-6c2f-4cd0-97e2
-3f5a4dc9bd6d) is being imported.
More info - https://console.aw
s.amazon.com/importexport",
  "Timestamp" : "2023-02-23T00:27:
58.831Z",
  "SignatureVersion" : "1",
  "Signature" : "FMG5t1ZhJNHLHUXvZ
gtZz1k24FzVa7oX0T4P03neeXw8
ZEXZx6z35j2F0TuNYShn2h0bKNC/
zLTnMyIxEzmi2X1sh0BWsJHkrW2xkr58ABZ
F+4uWHEE73yDVR4SyYAIkP9jstZzDRm
+bcVs8+T0yaLiEGLrIIIL4esi111hIkg
ErCuy5btPcWXBdio2fpCRD5x9oR
6gmE/rd5071X1c1uvnv4r1Lkk4pqP2/
iUfxFZva1xLSRvgyfm6D9hNk1VyPfy+7
Ta1MD01zmJu0rExtN5IbZew3foxgx8GT
+1bZkLd0ZdtDRJlIyPRP44eyq78sU0Eo/
LsDr0Iak4ZDpg8dXg==",
  "SigningCertURL" : "https://
sns.us-east-1.amazonaws.com/
SimpleNotificationService-010a507c1
833636cd94bdb98bd93083a.pem",
  "UnsubscribeURL" : "https://
sns.us-east-2.amazonaws.com/?
Action=Unsubscribe&SubscriptionArn
=arn:aws:sns:us-east-2:1111
22223333:ExampleTopic1:e103
9402-24e7-40a3-a0d4-797da162b297"
}
```

ジョブステータス	SNS 通知 JSON
完了	<pre>{ "Type" : "Notification", "MessageId" : "dc1e94d9-56c5-5e9 6-808d-cc7f68faa162", "TopicArn" : "arn:aws:sns:us-ea st-2:111122223333:ExampleTopic1", "Message" : "Your job Job-name (JID8bca334a-6c2f-4cd0-97e2 -3f5a4dc9bd6d) complete.\nThanks for using AWS Snow Family.\nCan you take a quick survey on your experienc e? Survey here: http://bit.ly/1pLQ JMY. More info - https://console.aw s.amazon.com/importexport", "Timestamp" : "2023-02-23T00:27: 58.831Z", "SignatureVersion" : "1", "Signature" : "FMG5t1ZhJNHLHUXvZ gtZz1k24FzVa7oX0T4P03neeXw8 ZEXZx6z35j2F0TuNYShn2h0bKNC/ zLTnMyIxEzmi2X1shOBWsJHkrW2xkR58ABZ F+4uWHEE73yDVR4SyYAIkP9jstZzDRm +bcVs8+T0yaLiEGLrIIIL4esi11lhIkG ErCuy5btPcWXBdio2fpCRD5x9oR 6gmE/rd5071X1c1uvnv4r1Lkk4pqP2/ iUfxFZva1xLSRvgyfm6D9hNk1VyPfy+7 Ta1MD01zmJu0rExtnSIbZew3foxgx8GT +1bZkLd0ZdtdRJIyPRP44eyq78sU0Eo/ LsDr0Iak4ZDpg8dXg==", "SigningCertURL" : "https:// sns.us-east-1.amazonaws.com/ SimpleNotificationService-010a507c1 833636cd94bdb98bd93083a.pem", "UnsubscribeURL" : "https:// sns.us-east-2.amazonaws.com/? Action=Unsubscribe&SubscriptionArn =arn:aws:sns:us-east-2:1111 22223333:ExampleTopic1:e103 9402-24e7-40a3-a0d4-797da162b297" }</pre>

ジョブステータス	SNS 通知 JSON

ジョブステータス

SNS 通知 JSON

キャンセル

```
{
  "Type" : "Notification",
  "MessageId" : "dc1e94d9-56c5-5e9
6-808d-cc7f68faa162",
  "TopicArn" : "arn:aws:sns:us-ea
st-2:111122223333:ExampleTopic1",
  "Message" : "Your job Job-name
(JID8bca334a-6c2f-4cd0-97e2
-3f5a4dc9bd6d) was canceled. More
info - https://console.aws.amazon.
com/importexport",
  "Timestamp" : "2023-02-23T00:27:
58.831Z",
  "SignatureVersion" : "1",
  "Signature" : "FMG5t1ZhJNHLHUXvZ
gtZz1k24FzVa7oX0T4P03neeXw8
ZEXZx6z35j2F0TuNYShn2h0bKNC/
zLTnMyIxEzmi2X1sh0BWsJHkrW2xkr58ABZ
F+4uWHEE73yDVR4SyYAIkP9jstZzDRm
+bcVs8+T0yaLiEGLrIIIL4esi111hIkG
ErCuy5btPcWXBdio2fpCRD5x9oR
6gmE/rd5071X1c1uvnv4r1Lkk4pqP2/
iUfxFZva1xLSRvgyfm6D9hNk1VyPfy+7
Ta1MD01zmJu0rExtN5IbZew3foxgx8GT
+1bZkLd0ZdtDRJlIyPRP44eyq78sU0Eo/
LsDr0Iak4ZDpg8dXg==",
  "SigningCertURL" : "https://
sns.us-east-1.amazonaws.com/
SimpleNotificationService-010a507c1
833636cd94bdb98bd93083a.pem",
  "UnsubscribeURL" : "https://
sns.us-east-2.amazonaws.com/?
Action=Unsubscribe&SubscriptionArn
=arn:aws:sns:us-east-2:1111
22223333:ExampleTopic1:e103
9402-24e7-40a3-a0d4-797da162b297"
}
```

を使用した AWS Snowball Edge API コールのログ記録 AWS CloudTrail

AWS Snowball または Snow Family サービスは、ユーザー AWS CloudTrail、ロール、または service. CloudTrail captures によって実行されたアクションを記録するサービスであると統合されます。AWS Snow Family サービスのすべての API コールをキャプチャします。キャプチャされた呼び出しには、AWS Snowball ファミリーコンソールからの呼び出しと、ファミリージョブ管理 API AWS Snowball 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、ファミリー API コールの CloudTrail イベントなど、Amazon S3 バケットへのイベントの AWS Snowball 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。によって収集された情報を使用して CloudTrail、AWS Snowball Family API で行われたリクエスト、行われたリクエストの IP アドレス、リクエスト者、リクエスト日時などの詳細を確認できます。

の詳細については CloudTrail、[「AWS CloudTrail ユーザーガイド」](#)を参照してください。

AWS Snowball のエッジ情報 CloudTrail

CloudTrail アカウントを作成する AWS アカウントと、で有効になります。AWS Snowball Edge でアクティビティが発生すると、そのアクティビティは CloudTrail イベント履歴の他の AWS サービスイベントとともにイベントに記録されます。で最近のイベントを表示、検索、ダウンロードできます AWS アカウント。詳細については、「[ユーザーガイド](#)」の「[イベント履歴を使用した CloudTrail イベントの表示](#)」を参照してください。AWS CloudTrail

Edge 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 AWS Snowball、証跡を作成します。証跡により CloudTrail、はログファイルを Amazon S3 バケットに配信できます。デフォルトでは、コンソールで証跡を作成すると、すべての AWS リージョンに証跡が適用されます。証跡は AWS、パーティション AWS リージョン内のすべての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、『AWS CloudTrail ユーザーガイド:』の以下のトピックを参照してください。

- [追跡作成の概要](#)
- [CloudTrail サポートされているサービスと統合](#)
- [の Amazon SNS 通知の設定 CloudTrail](#)

- [複数のリージョンからの CloudTrail ログファイルの受信](#)と[複数のアカウントからのログファイルの受信 CloudTrail](#)

すべてのジョブ管理アクションは [AWS Snowball API リファレンス](#) に文書化されており、以下の例外 CloudTrail を除いて によってログに記録されます。

- [CreateAddress](#) オペレーションは、顧客の機密情報を保護するためにログに記録されません。
- すべての読み取り専用 API コール (Get、Describe、List というプレフィックスで始まる API オペレーション) ではレスポンス要素は記録されません。

各イベントまたはログエントリには、誰がリクエストを生成したかという情報が含まれます。アイデンティティ情報は、以下を判別するのに役立ちます：

- リクエストがルート認証情報または AWS Identity and Access Management (IAM ユーザー) 認証情報のどちらを使用して行われたか。
- 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テンポラリなセキュリティ認証情報を使用して行われたかどうか。
- リクエストが別の AWS サービスによって行われたかどうか。

詳細については、「ユーザーガイド」の [CloudTrail userIdentity 要素](#) AWS CloudTrail 」を参照してください。

AWS Snowball Edge のログファイルエントリの概要

証跡は、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できるようにする設定です。CloudTrail ログファイルには 1 つ以上のログエントリが含まれます。イベントは任意のソースからの単一のリクエストを表し、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に関する情報が含まれます。CloudTrail ログファイルはパブリック API コールの順序付けられたスタックトレースではないため、特定の順序では表示されません。

次の例は、[DescribeJob](#) オペレーションを示す CloudTrail ログエントリを示しています。

```
    {"Records": [
  {
    "eventVersion": "1.05",
    "userIdentity": {
```

```
    "type": "Root",
    "principalId": "111122223333",
    "arn": "arn:aws:iam::111122223333:root",
    "accountId": "111122223333",
    "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
    "sessionContext": {"attributes": {
      "mfaAuthenticated": "false",
      "creationDate": "2019-01-22T21:58:38Z"
    }},
    "invokedBy": "signin.amazonaws.com"
  },
  "eventTime": "2019-01-22T22:02:21Z",
  "eventSource": "snowball.amazonaws.com",
  "eventName": "DescribeJob",
  "awsRegion": "eu-west-1",
  "sourceIPAddress": "192.0.2.0",
  "userAgent": "signin.amazonaws.com",
  "requestParameters": {"jobId": "JIDa1b2c3d4-0123-abcd-1234-0123456789ab"},
  "responseElements": null,
  "requestID": "12345678-abcd-1234-abcd-ab0123456789",
  "eventID": "33c7ff7c-3efa-4d81-801e-7489fe6fff62",
  "eventType": "AwsApiCall",
  "recipientAccountId": "444455556666"
}
]}
```


AWS Snowball Edge クォータ

以下に、AWS Snowball Edge デバイスの使用に関する制限について説明します。

Important

Snowball Edge を使用して Amazon Simple Storage Service (Amazon S3) にデータを転送する場合、個々の Amazon S3 オブジェクトのサイズは最低 0 バイトから最大 5 テラバイト (TB) までの範囲に設定することに留意してください。

のリージョンの可用性 AWS Snowball Edge

次の表 AWS Snowball Edge は、 が利用可能なリージョンを示しています。

リージョン	Snowball Edge の可用性
米国東部 (オハイオ)	✓
米国東部 (バージニア北部)	✓
米国西部 (北カリフォルニア)	✓
米国西部 (オレゴン)	✓
AWS GovCloud (米国東部)	✓
AWS GovCloud (米国西部)	✓
カナダ (中部)	✓
アジアパシフィック (ジャカルタ)	✓
アジアパシフィック (ムンバイ)	✓
アジアパシフィック (大阪)	✓
アジアパシフィック (ソウル)	✓
アジアパシフィック (シンガポール)	✓

リージョン	Snowball Edge の可用性
アジアパシフィック (シドニー)	✓
アジアパシフィック (東京)	✓
欧州 (フランクフルト)	✓
欧州 (アイルランド)	✓
欧州 (ロンドン)	✓
欧州 (ミラノ)	✓
ヨーロッパ (パリ)	✓
欧州 (ストックホルム)	✓
中東 (アラブ首長国連邦)	✓
南米 (サンパウロ)	✓

サポートされている AWS リージョンとエンドポイントの詳細については、「」の[AWS 「Snow Family エンドポイントとクォータ」](#)を参照してください。AWS 全般のリファレンス

AWS Snowball Edge ジョブの制限事項

AWS Snowball Edge デバイスジョブの作成には、次の制限があります。

- セキュリティ上の理由から、AWS Snowball Edge デバイスを使用するジョブは、準備から 360 日以内に完了する必要があります。1 つ以上のデバイスを 360 日以上保持する必要がある場合は、「」を参照してください[SSL 証明書を更新する](#)。360 日を過ぎると、デバイスがロックされてアクセスできなくなるため、返送していただく必要が生じます。インポートジョブ中に AWS Snowball Edge デバイスがロックされた場合でも、デバイス上の既存のデータを Amazon S3 に転送できます。
- AWS Snowball Edge は、Amazon S3-managed暗号化キーによるサーバー側の暗号化 (SSE-S3) と、AWS Key Management Service マネージドキーによるサーバー側の暗号化 (SSE-KMS) をサポートしています。Snow Family デバイスの Amazon S3 互換ストレージは、ローカルのコンピューティングジョブとストレージジョブで SSS-C をサポートしています。詳細については、

「Amazon Simple Storage Service ユーザーガイド」の「[サーバー側の暗号化を使用したデータの保護](#)」を参照してください。

- デバイスを使用してデータ AWS Snowball Edge をインポートしていて、単一の Snowball Edge デバイ스에 収まる量を超えるデータを転送する必要がある場合は、追加のジョブを作成します。各エクスポートジョブでは、複数の Snowball Edge デバイスを使用できます。
- 一度に使用できる Snowball Edge デバイスの数のデフォルトのサービス制限は、各 AWS リージョンにつき 1 アカウントあたり 1 台です。サービス制限を増加する、またはクラスタージョブを作成するには、[AWS Support](#) までお問い合わせください。
- デバイスに転送されたオブジェクトのメタデータは保持されません。同一のまま残るメタデータは filename および filesize のみです。その他のすべてのメタデータは、次の例のように設定されます。

```
-rw-rw-r-- 1 root root [filesize] Dec 31 1969 [path/filename]
```

のレート制限 AWS Snowball Edge

レートリミッターは、サーバークラスター環境におけるリクエストのレートを制御するために使用されます。

Amazon Snow S3 アダプターの接続制限

Amazon S3 の Snowball Edge の最大接続制限は 1000 です。1000 を超える接続はすべてドロップされます。

Snowball Edge デバイスでオンプレミスのデータを転送する際の制限

オンプレミスの AWS Snowball Edge デバイスとの間でデータを転送する場合、次の制限があります。

- ファイルは、書き込み中、静的な状態になっている必要があります。転送中に変更されたファイルは Amazon S3 にインポートされません。
- ジャンボフレーム (1,500 バイトを越えるペイロードのあるイーサネットフレーム) はサポートされません。

- エクスポートするデータを選択する際、データ名の末尾にスラッシュの付いたオブジェクト (/ または \) は転送されないことにご注意ください。末尾にスラッシュのあるオブジェクトをエクスポートする前に、スラッシュを削除して名前を更新してください。
- マルチパートデータ転送を使用する場合、最大パートサイズは 2 GiB です。

Snowball Edge の配送の制限

AWS Snowball Edge デバイスの配送には、次の制限があります。

- AWS は、Snowball Edge デバイスを私書箱に配送しません。
- AWS は、欧州 (アイルランド) から欧州 (フランクフルト)、アジアパシフィック (シドニー) など、米国以外のリージョン間で Snowball Edge デバイスを配送しません。
- Snowball Edge デバイスをジョブの作成時に指定された国以外の住所に移動することは許可されておらず、AWS サービス条件に違反します。

配送の詳細については、[Snow Family デバイスの配送に関する考慮事項](#) を参照してください。

インポート用に返却された Snowball Edge の処理の制限

データをインポートするには AWS、デバイスが次の要件を満たしている必要があります。

- AWS Snowball Edge デバイスが侵害されてはいけません。前面、背面、上部の 3 つのドアを開いたり、オプションのエアフィルターを追加および交換したりする場合を除き、何らかの理由で AWS Snowball Edge デバイスを開かないでください。
- デバイスが物理的に破損していないこと。Snowball Edge デバイスの 3 つのドアを、ラッチがカチッと音を立てるまで閉めることで、損傷を防ぐことができます。
- Snowball Edge デバイスの E Ink ディスプレイが表示されている必要があります。また、AWS Snowball Edge デバイスへのデータ転送が完了したときに自動的に生成されたリターンラベルも表示する必要があります。

Note

上記の要件を満たさずに返却されたすべての Snowball Edge デバイスは、作業を行うことなく消去されます。

AWS Snowball Edge のトラブルシューティング

トラブルシューティングを行う際は、次の一般的なガイドラインを念頭に置いてください。

- Amazon S3 のオブジェクトの最大ファイルサイズは 5 TB です。
- AWS Snowball Edge デバイスに転送されるオブジェクトの最大キー長は 933 バイトです。1 文字あたり 1 バイトを超える文字を含むキー名の場合も、最大キー長は 933 バイトです。キー長には、ファイル名やオブジェクト名とともにパスやプレフィックスも含まれます。したがって、ファイル名が短くても、深く入れ子になっているパスの中にある場合は、キー長が 933 バイトを超える場合があります。キー長の決定には、バケット名はパスとして考慮されません。以下にいくつかの例を示します。

オブジェクト名	バケット名	パスとバケット名	キー長
sunflower-1.jpg	pictures	sunflower-1.jpg	15 文字
receipts.csv	MyTaxInfo	/Users/Eric/Documents/2016/January/	47 文字
bhv.1	\$7\$zWwwXKQj\$gLA0oZCj\$r8p	/.VfV/FqGC3QN\$7BXys3KHYePfuIOMNjY83dVxugPY1xVg/evpcQEJLT/rSwZc\$M1VVf/\$hwefVISRqwepB\$/BiiD/PPF\$twRAjrD	135 文字

オブジェクト名	バケット名	パスとバケット名	キー長
		/fIMp/0NY	

- セキュリティ上の理由から、AWS Snowball Edge デバイスを使用するジョブは、準備から 360 日以内に完了する必要があります。1 つ以上のデバイスを 360 日以上保持する必要がある場合は、「」を参照してください[SSL 証明書を更新する](#)。それ以外の場合、360 日以降はデバイスがロックされてアクセスできなくなるため、返却する必要があります。インポートジョブ中に AWS Snowball Edge デバイスがロックされた場合でも、デバイス上の既存のデータを Amazon S3 に転送できます。
- AWS Snowball Edge デバイスを使用して予期しないエラーが発生した場合は、そのことを確認する必要があります。関連するログをコピーし、発生した問題の簡単な説明とともにへのメッセージに含めます AWS Support。ログの詳細については、[Snowball Edge クライアントコマンドの使用](#) を参照してください。

トピック

- [デバイスの識別方法](#)
- [起動に関する問題のトラブルシューティング](#)
- [接続の問題のトラブルシューティング](#)
- [unlock-device コマンドの問題のトラブルシューティング](#)
- [マニフェストファイルの問題のトラブルシューティング](#)
- [認証情報の問題のトラブルシューティング](#)
- [NFS インターフェイスの問題のトラブルシューティング](#)
- [データ転送に関する問題のトラブルシューティング](#)
- [AWS CLI 問題のトラブルシューティング](#)
- [インポートジョブの問題のトラブルシューティング](#)
- [エクスポートジョブの問題のトラブルシューティング](#)

デバイスの識別方法

describe-device コマンドを使用してデバイスタイプを調べ、以下の表で DeviceType の戻り値を調べて構成を判断してください。

```
snowballEdge describe-device
```

Example **describe-device** の出力

```
{
  "DeviceId" : "JID-206843500001-35-92-20-211-23-06-02-18-24",
  "UnlockStatus" : {
    "State" : "UNLOCKED"
  },
  "ActiveNetworkInterface" : {
    "IpAddress" : "127.0.0.1"
  },
  "PhysicalNetworkInterfaces" : [ {
    "PhysicalNetworkInterfaceId" : "s.ni-8d0ef958ec860ac7c",
    "PhysicalConnectorType" : "RJ45",
    "IpAddressAssignment" : "DHCP",
    "IpAddress" : "172.31.25.194",
    "Netmask" : "255.255.240.0",
    "DefaultGateway" : "172.31.16.1",
    "MacAddress" : "02:38:30:12:a3:7b",
    "MtuSize" : "1500"
  } ],
  "DeviceCapacities" : [ {
    "Name" : "HDD Storage",
    "Unit" : "Byte",
    "Total" : 39736350227824,
    "Available" : 985536581632
  }, {
    "Name" : "SSD Storage",
    "Unit" : "Byte",
    "Total" : 6979321856000,
    "Available" : 6979321856000
  }, {
    "Name" : "vCPU",
    "Unit" : "Number",
    "Total" : 52,
    "Available" : 52
  }, {
    "Name" : "Memory",
    "Unit" : "Byte",
```

```

    "Total" : 223338299392,
    "Available" : 223338299392
  }, {
    "Name" : "GPU",
    "Unit" : "Number",
    "Total" : 0,
    "Available" : 0
  } ],
  "DeviceType" : "EDGE_C"
}

```

DeviceTypeと Snow Family のデバイス構成

DeviceType 値	デバイス設定
EDGE	Snowball Edge ストレージ最適化 (EC2 コンピューティング機能搭載)
EDGE_C	Snowball Edge コンピューティング最適化 (AMD EPYC Gen1 および HDD 搭載)
EDGE_CG	Snowball Edge コンピューティング最適化 (AMD EPYC Gen1、HDD、GPU 搭載)
EDGE_S	Snowball Edge ストレージ最適化
V3_5C	Snowball Edge コンピューティング最適化 (AMD EPYC Gen2 および NVME 搭載)
V3_5S	Snowball Edge ストレージ最適化 210 TB

Snowball Edge デバイス構成の詳細については、「[AWS Snowball エッジデバイスのハードウェア情報](#)」を参照してください。

起動に関する問題のトラブルシューティング

Snow Family デバイスの起動で発生する可能性のある特定の問題のトラブルシューティングには、以下の情報が役立ちます。

- デバイスが起動するまで 10 分ほど待ちます。起動中は、デバイスを動かしたり使用したりしないでください。
- 電源を供給するケーブルの両端がしっかりと接続されていることを確認してください。
- 電力を供給するケーブルを、問題がないことを確認済みの別のケーブルと交換します。
- 電力を供給するケーブルを、問題がないことを確認済みの別の電源に接続します。

起動中の LCD ディスプレイの問題のトラブルシューティング

Snowball Edge デバイスの電源を入れると、LCD ディスプレイで問題が発生することがあります。

- LCD 画面は黒で、Snowball Edge デバイスを電源に接続して LCD 画面の上にある電源ボタンを押しても画像は表示されません。
- LCD 画面が Snowball Edge の設定を過ぎても進行しないため、数分かかる場合があります。メッセージとネットワーク設定画面は表示されません。

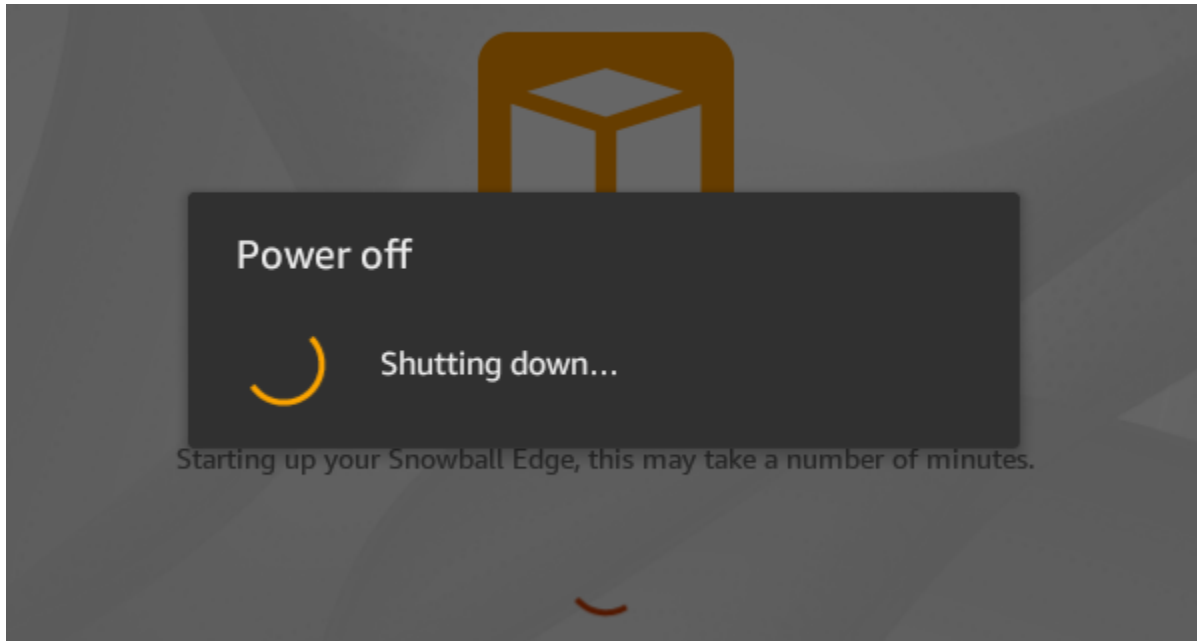


電源ボタンを押した後に LCD 画面が黒くなったときに実行するアクション

1. Snowball Edge デバイスが電源に接続され、電源が電源を提供していることを確認します。
2. デバイスを 1~2 時間電源に接続したままにします。デバイスの前面と背面のドアが開いていることを確認します。
3. デバイスに戻ると、LCD 画面を使用する準備が整います。

Snowball Edge がネットワーク設定画面に進まない場合に実行するアクション

1. [Snowball Edge の設定中です。これには数分かかることがあります] というメッセージが表示されたまま 10 分ほど待ちます。
2. 画面で [ディスプレイを再起動] ボタンを選択します。[シャットダウンしています...] メッセージが表示された後、[Snowball Edge の設定中です。これには数分かかることがあります] メッセージが表示され、デバイスは正常に起動します。



[ディスプレイを再起動] ボタンを使用した後も、LCD 画面が [Snowball Edge の設定中です。これには数分かかることがあります] から先に進まない場合は、次の手順を実行してください。

実行するアクション

1. LCD 画面の上にある電源ボタンを押して、デバイスの電源をオフにします。
2. デバイスからすべてのケーブルを取り外します。
3. デバイスの電源を切り、そのまま 20 分間待ちます。
4. 電源とネットワークケーブルを接続します。
5. LCD 画面の上にある電源ボタンを押して、デバイスの電源を入れます。

問題が解決しない場合は、AWS Support に連絡してデバイスを返却し、新しい Snowball Edge デバイスを受け取ります。

起動時の E Ink ディスプレイに関する問題のトラブルシューティング

Snowball Edge デバイスの電源を入れると、デバイス上部の E Ink ディスプレイに次のメッセージが表示されることがあります。

```
The appliance has timed out
```

このメッセージは、デバイスに問題があることを示すものではありません。通常どおり使用し、オフにしてに返送すると AWS、返送の配送情報が想定どおりに表示されます。

接続の問題のトラブルシューティング

Snowball Edge への接続で発生する可能性のある特定の問題のトラブルシューティングには、以下の情報が役立ちます。

- 1 秒あたり 100 メガバイトの速度で動作するルーターおよびスイッチは、Snowball Edge では使用できません。1 秒あたり 1 GB (またはそれ以上) の速度で動作するスイッチを使用することをお勧めします。
- デバイスとの異常な接続エラーが発生した場合は、Snowball Edge の電源をオフにし、すべてのケーブルを抜いて、10 分間そのままにしておきます。10 分が経過したら、デバイスを再起動してもう一度お試しください。
- ウイルス対策ソフトウェアやファイアウォールが、Snowball Edge デバイスのネットワーク接続をブロックしていないことを確認します。
- ファイルインターフェイスおよび Amazon S3 インターフェイスは、異なる IP アドレスを持つことに注意してください。

より高度な接続のトラブルシューティングについては、次の手順を実行します。

- Snowball Edge と通信できない場合は、デバイスの IP アドレスに ping を送信します。ping が no connect を返す場合、デバイスの IP アドレスとローカルネットワーク設定を確認してください。
- IP アドレスが正しく、デバイスの裏側のライトが点滅している場合は、telnet を使用して、ポート 22、9091、8080 でデバイスをテストします。ポート 22 をテストして、Snowball Edge が正常に動作しているかどうかを判断します。ポート 9091 をテストすると、を使用してデバイスにコマンドを送信 AWS CLI できるかどうかが決まります。ポート 8080 をテストすると、デバイスが Amazon S3 バケットに書き込めることを確認できます。ポート 22 で接続できてもポート 8080 では接続できない場合、まず Snowball Edge の電源を切ってから、すべてのケーブルを抜きま。10 分間デバイスをそのままにします。それから、再接続してもう一度開始します。

unlock-device コマンドの問題のトラブルシューティング

unlock-device コマンドが を返す場合 connection refused、コマンド構文の入力を間違えているか、コンピュータまたはネットワークの設定によってコマンドが Snow デバイスに到達できない可能性があります。問題を解決するには、次のアクションを実行します。

1. コマンドが正しく入力されたことを確認します。
 - a. デバイスの LCD 画面を使用して、コマンドで使用されている IP アドレスが正しいことを確認します。
 - b. コマンドで使用されるマニフェストファイルへのパスが正しいことを確認します。これには、ファイル名も含まれます。
 - c. を使用して [AWS Snow ファミリーマネジメントコンソール](#)、コマンドで使用されているロック解除コードが正しいことを確認します。
2. 使用しているコンピュータが Snow デバイスと同じネットワークとサブネットにあることを確認します。
3. 使用しているコンピュータとネットワークが Snow デバイスへのアクセスを許可するように設定されていることを確認します。オペレーティングシステムの ping コマンドを使用して、コンピュータがネットワーク経由で Snow デバイスにアクセスできるかどうかを確認します。ウイルス対策ソフトウェアの設定、ファイアウォール設定、仮想プライベートネットワーク (VPN)、またはコンピュータとネットワークのその他の設定を確認します。

マニフェストファイルの問題のトラブルシューティング

各ジョブには、特定のマニフェストファイルが関連付けられています。複数のジョブを作成する場合は、どのマニフェストがどのジョブに関連しているかを追跡します。

マニフェストファイルを紛失した場合、またはマニフェストファイルが破損している場合は、特定のジョブのマニフェストファイルを再度ダウンロードできます。これを行うには、コンソール AWS CLI、または AWS APIs のいずれかを使用します。

認証情報の問題のトラブルシューティング

以下のトピックを使用して、Snowball Edge で発生した認証情報の問題を解決します。

認証情報が見つかりません AWS CLI

を使用して Amazon S3 インターフェイスを介して AWS Snowball Edge デバイスと通信している場合 AWS CLI、認証情報を見つけることができないというエラーメッセージが表示されることがあります。[aws configure] (aws の設定) を実行して、認証情報を設定できます。

実行するアクション

がコマンドの実行 AWS CLI に使用する AWS 認証情報を設定します。詳細については、「[AWS CLI ユーザーガイド](#)」の「AWS Command Line Interface の設定。」を参照してください。

エラーメッセージ: シークレットアクセスキーの確認と署名

Amazon S3 インターフェイスを使用して Snowball Edge にデータを転送する場合、次のエラーメッセージが表示される可能性があります。

```
An error occurred (SignatureDoesNotMatch) when calling the CreateMultipartUpload operation: The request signature we calculated does not match the signature you provided.
```

Check your AWS secret access key and signing method. For more details go to:

[http://docs.aws.amazon.com/AmazonS3/latest/dev/](http://docs.aws.amazon.com/AmazonS3/latest/dev/RESTAuthentication.html#ConstructingTheAuthenticationHeader)

[RESTAuthentication.html#ConstructingTheAuthenticationHeader](http://docs.aws.amazon.com/AmazonS3/latest/dev/RESTAuthentication.html#ConstructingTheAuthenticationHeader)

実行するアクション

Snowball Edge クライアントから認証情報を取得します。詳細については、「[認証情報の取得](#)」を参照してください。

NFS インターフェイスの問題のトラブルシューティング

Snow Family デバイスは、NFS インターフェイスのステータスがであることを示している場合があります DEACTIVATED。これは、最初に NFS インターフェイスを停止せずに Snow Family デバイスの電源がオフになった場合に発生する可能性があります。

実行するアクション

この問題を修正するには、次のステップにより、NFS サービスを停止して再起動します。

1. describe-service コマンドを使用して、サービスのステータスを確認します。

```
snowballEdge describe-service --service-id nfs
```

コマンドにより以下が返されます。

```
{
  "ServiceId" : "nfs",
  "Status" : {
    "State" : "DEACTIVATED"
  }
}
```

2. stop-service コマンドを使用して NFS サービスを停止します。

```
snowballEdge stop-service --service-id nfs
```

3. start-service コマンドを使用して NFS サービスを開始します。詳細については、「[Snow Family デバイスで NFS サービスを開始する](#)」を参照してください。

```
snowballEdge start-service --virtual-network-interface-arns vni-arn --service-id
nfs --service-configuration AllowedHosts=0.0.0.0/0
```

4. describe-service コマンドを使用して、サービスが実行されていることを確認します。

```
snowballEdge describe-service --service-id nfs
```

State 名前の値が の場合ACTIVE、NFS インターフェイスサービスはアクティブです。

```
{
  "ServiceId" : "nfs",
  "Status" : {
    "State" : "ACTIVE"
  },
}
```

```
"Endpoints" : [ {  
  "Protocol" : "nfs",  
  "Port" : 2049,  
  "Host" : "192.0.2.0"  
} ],  
"ServiceConfiguration" : {  
  "AllowedHosts" : [ "10.24.34.0/23", "198.51.100.0/24" ]  
}  
}
```

データ転送に関する問題のトラブルシューティング

Snowball Edge との間のデータ転送中にパフォーマンスの問題が発生した場合は、転送パフォーマンスの改善の推奨事項とガイダンスについて、「[パフォーマンス](#)」を参照してください。Snowball Edge との間のデータ転送で発生する可能性のある問題のトラブルシューティングには、以下が役立ちます。

- Snowball Edge のルートディレクトリにデータを転送することはできません。デバイスへのデータ転送に問題が発生した場合は、サブディレクトリにデータを転送していることを確認してください。最上位のサブディレクトリには、ジョブに含めた Amazon S3 バケットの名前があります。それらのサブディレクトリにデータを配置します。
- Linux を使用していて、UTF-8 文字を含むファイルを AWS Snowball Edge デバイスにアップロードできない場合は、Linux サーバーで UTF-8 文字エンコードが認識されていない可能性があります。この問題は、`locales` パッケージを Linux サーバーにインストールし、`en_US.UTF-8` のように UTF-8 ロケールのいずれかを使用するように設定することで解決できます。`locales` パッケージは環境変数 `LC_ALL` (例: `export LC_ALL=en_US.UTF-8`) をエクスポートして設定できます。
- で Amazon S3 インターフェイスを使用すると AWS CLI、や など、名前にスペースがあるファイル `my photo.jpg` やフォルダを操作できます `My Documents`。ただし、必ずスペースが適切に処理されていることを確認してください。詳細については、AWS Command Line Interface ユーザーガイドの [AWS CLI のパラメータ値の指定](#) を参照してください。

AWS CLI 問題のトラブルシューティング

AWS Snowball Edge デバイスと AWS CLI を使用する際の問題は、以下のトピックを参照して解決してください。

AWS CLI エラーメッセージ : 「Profile Cannot Be Null」

を操作するとき AWS CLI、プロファイルが null にできないというエラーメッセージが表示されることがあります。このエラー AWS CLI は、 がインストールされていないか、 AWS CLI プロファイルが設定されていない場合に発生する可能性があります。

実行するアクション

ワークステーション AWS CLI をダウンロードして設定していることを確認します。詳細については、[「ユーザーガイド」の「バンドルされたインストーラ \(Linux、macOS、または Unix\) AWS CLI を使用して](#) をインストールするAWS Command Line Interface」を参照してください。

を使用してデータを転送するときの Null ポインタエラー AWS CLI

を使用してデータ AWS CLI を転送すると、null ポインタエラーが発生する可能性があります。このエラーは、次の条件で発生することがあります。

- 指定されたファイル名にタイプミスがあった場合 (例: flowwer.png の代わりに flower.npg や flower.png)
- 指定されたパスが間違っていた場合 (例: C:\Documents\flower.png の代わりに C:\Documents\flower.png)
- ファイルが破損している場合

実行するアクション

ファイル名とパスが正しいことを確認して、もう一度試してください。この問題が続く場合は、ファイルが破損していないことを確認して、転送を中止するか、またはファイルの修復を試みます。

インポートジョブの問題のトラブルシューティング

ファイルを Amazon S3 にインポートできなくなることがあります。次の問題が発生した場合は、指定されたアクションをお試しの上、問題を解決してください。ファイルのインポートに失敗した場合は、再度インポートが必要になる場合があります。再度インポートするには、Snowball Edge で新しいジョブの作成が必要になる場合があります。

オブジェクト名に無効な文字が含まれているため、ファイルを Amazon S3 にインポートできない

この問題は、ファイル名またはフォルダ名に Amazon S3 でサポートされていない文字が含まれている場合に発生します。Amazon S3 には、オブジェクト名に使用できる文字についてのルールがあり

ます。詳細については、「Amazon S3 ユーザーガイド」の「[オブジェクトキー名の作成](#)」を参照してください。

実行するアクション

この問題が発生した場合は、インポートに失敗したファイルとフォルダのリストがジョブ完了レポートに表示されます。

場合によっては、リストまたはリスト内のファイルの容量が非常に大きいために、インターネット経由で転送できないことがあります。このような場合は、Snowball インポートジョブを新しく作成するか、ファイル名およびフォルダ名を変更して Amazon S3 にインポートし、ファイルを再度転送します。

ファイルが小さく、ファイル数も多くない場合は、AWS CLI または `awscli` を使用して Amazon S3 にコピーできます AWS Management Console。詳細については、Amazon Simple Storage Service ユーザーガイドの [How do I upload files and folders to an S3 bucket?](#) を参照してください。

エクスポートジョブの問題のトラブルシューティング

ファイルをワークステーションにエクスポートできないことがあります。次の問題が発生した場合は、指定されたアクションをお試しの上、問題を解決してください。ファイルのエクスポートに失敗した場合は、再度エクスポートが必要になる場合があります。再度エクスポートするには、Snowball Edge で新しいジョブの作成が必要になる場合があります。

Microsoft Windows Server にファイルをエクスポートできませんでした

該当フォルダ名または関連フォルダ名が Windows でサポートされていない形式の場合、ファイルを Microsoft Windows Server にエクスポートできないことがあります。たとえば、ファイル名またはフォルダ名にコロン (:) が含まれている場合、Windows のファイル名またはフォルダ名はそのような文字に対応していないため、エクスポートは失敗します。

実行するアクション

1. エラーが発生している名前のリストを作成します。エクスポートに失敗したファイル名やフォルダ名は、ログで確認できます。詳細については、「[AWS Snowball Edge ログ](#)」を参照してください。
2. 問題が発生している Amazon S3 内のオブジェクト名を変更し、サポートされていない文字を削除するか、置き換えます。
3. 名前のリストまたはそのリスト内のファイルの容量が非常に大きいために、インターネット経由で転送できない場合は、特に該当するオブジェクトの新しいエクスポートジョブを作成します。

ファイルのサイズが小さく、ファイルの数も多くない場合は、名前を変更したオブジェクトを Amazon S3 から AWS CLI または を通じてコピーします AWS Management Console。詳細については、Amazon Simple Storage Service ユーザーガイドの [How Do I Download an Object from an S3 Bucket?](#) を参照してください。

ドキュメント履歴

- API バージョン: 1.0
- ドキュメントの最終更新日: 2024 年 3 月 14 日

次の表は、2018 年 7 月以降の AWS Snowball Edge デベロッパーガイドの重要な変更点をまとめたものです。ドキュメントの更新に関する通知については、RSS フィードにサブスクライブできます。

変更	説明	日付
Snowball Edge デバイスのテープゲートウェイは廃止されました	テープゲートウェイの機能は、Snowball Edge デバイスでは使用できません。	2024 年 3 月 14 日
ファイルインターフェイスが非推奨になりました	ファイルインターフェイスはデータ転送に使用できなくなりました。	2024 年 3 月 1 日
Snowball Edge ストレージ最適化 210 TB デバイスで利用可能な Snow Family デバイスの Amazon S3 互換ストレージ	Snow Family デバイスの Amazon S3 互換ストレージは、Snowball Edge ストレージ最適化 210 TB デバイスの S3 ストレージで使用できません。詳細については、 「Snow Family デバイスの Amazon S3 互換ストレージの使用」 を参照してください。	2024 年 2 月 26 日
デバイスの注文時にカスタム AMIs を含める	AWS Snow Family カスタム Amazon マシンイメージをジョブの注文中にプリロードできるようになりました。詳細については、 「からの AMI の追加 AWS Marketplace」 を参照してください。	2023 年 11 月 15 日

[Snow Family デバイスの Amazon S3 互換ストレージの一般提供開始](#)

Snow Family デバイスの Amazon S3 互換ストレージは、Snowball Edge コンピューティング最適化デバイスでサポートされています。詳細については、「[Snow Family デバイスの Amazon S3 互換ストレージ](#)」を参照してください。

2023 年 4 月 20 日

[新たに AWS リージョン サポートされる](#)

AWS Snowball が中東 (UAE) リージョンでサポートされるようになりました。このリージョンのエンドポイントの詳細については、「AWS 全般のリファレンス」の「[AWS Snow Family エンドポイントとクォータ](#)」を参照してください。配送の詳細については、「[Snowball Edge の配送に関する考慮事項](#)」を参照してください。

2023 年 3 月 6 日

[新たに AWS リージョン サポートされる](#)

AWS Snowball がアジアパシフィック (ジャカルタ) リージョンでサポートされるようになりました。このリージョンのエンドポイントの詳細については、「AWS 全般のリファレンス」の「[AWS Snow Family エンドポイントとクォータ](#)」を参照してください。配送の詳細については、「[Snowball Edge の配送に関する考慮事項](#)」を参照してください。

2022 年 9 月 7 日

[Snowball Edge の大規模データ移行](#)

Snowball Edge が大規模データ移行プランの自動化をサポートするようになりました。詳細については、「[大規模なデータ移行](#)」(手動手順)と、必要に応じて「[大規模データ移行プランの作成](#)」を参照して、自動化を開始してください。

2022 年 4 月 27 日

[の紹介 AWS Snow Device Management](#)

Snow Device Management を使用すると、Snowball Edge デバイスとローカル AWS サービスをリモートで管理できます。すべての Snowball Edge デバイスは Snow Device Management をサポートしており、Snowball Edge AWS リージョン が利用可能なほとんどの の新しいデバイスにプリインストールされています。詳細については、「[を使用したデバイスの管理 AWS Snow Device Management](#)」を参照してください。

2022 年 4 月 27 日

[Snowball Edge の NFS 設定](#)

ストレージ最適化デバイスに [Snowball Edge の NFS 設定](#)が追加されました。

2022 年 4 月 21 日

[ロードバランサーのレート制限](#)

Snowball Edge が、サーバークラスター環境でリクエストを分散するための [レート制限](#)をサポートするようになりました。

2022 年 4 月 19 日

[テープゲートウェイによる Snowball Edge をサポート](#)

テープゲートウェイサービスをホストするように特別に設定された Snowball Edge デバイスを注文できるようになりました。このテクノロジーの組み合わせにより、オフラインのテープデータの移行を安全に行えるようになります。

2021 年 11 月 30 日

[Network Time Protocol \(NTP\) サーバー設定をサポート](#)

Snowball Edge デバイスが、外部の Network Time Protocol (NTP) のサーバー設定をサポートできるようになりました。

2021 年 11 月 16 日

[NFS によるオフラインのデータ転送をサポート](#)

Snowball Edge デバイスが、NFS を使用したオフラインのデータ転送をサポートできるようになりました。詳細については、[オフラインのデータ転送に NFS を使用する](#)を参照してください。

2021 年 8 月 4 日

[新たに AWS リージョン サポート](#)

Snowball Edge デバイスがアフリカ (ケープタウン) で利用可能になりました AWS リージョン。詳細については、「AWS 全般のリファレンス」の「[AWS Snow Family エンドポイントとクォータ](#)」を参照してください。配送の詳細については、[Snowball Edge の配送についての考慮事項](#)を参照してください。

2020 年 11 月 23 日

[独自のイメージのデバイスへのインポートをサポート](#)

イメージのスナップショットを Snowball Edge デバイスにインポートし、Amazon EC2 互換 Amazon マシンイメージ (AMI) として登録できるようになりました。詳細については、「[Amazon EC2 AMI としてデバイスにイメージをインポートする](#)」を参照してください。

2020 年 11 月 9 日

[新たに AWS リージョン サポート](#)

Snowball Edge デバイスが欧州 (ミラノ) で利用可能になりました AWS リージョン。詳細については、「AWS 全般のリファレンス」の「[AWS Snow Family エンドポイントとクォータ](#)」を参照してください。配送の詳細については、[Snowball Edge の配送についての考慮事項](#)を参照してください。

2020 年 9 月 30 日

[コンテンツの再編成](#)

AWS Snow ファミリーマネジメントコンソールワークフローに沿った開始方法のセクションを作成し、わかりやすくするために他のセクションを更新しました。詳細については、[AWS Snowball Edge の開始方法](#)を参照してください。

2020 年 9 月 17 日

[の紹介 AWS OpsHub for Snow Family](#)

Snow Family デバイスは、デバイスとローカル AWS サービスの管理に使用できる AWS OpsHub for Snow Family、使いやすいツールを提供するようになりました。詳細については、[「AWS OpsHub for Snow Family を使用して Snowball デバイスを管理する」](#)を参照してください。

2020 年 4 月 16 日

[AWS Identity and Access Management \(IAM\) が AWS Snowball Edge デバイス上でローカルで利用可能になりました](#)

AWS Identity and Access Management (IAM) を使用して、AWS Snowball Edge デバイスで実行されているリソースへのアクセス AWS を安全に制御できるようになりました。詳細については、「[Using IAM Locally](#)」を参照してください。

2020 年 4 月 16 日

[新しい Snowball Edge Storage Optimized \(データ転送用\) デバイスオプションの導入](#)

Snowball では、既存のコンピューティング最適化デバイスと GPU デバイスに基づく、新しいストレージ最適化デバイスが追加されました。詳細については、[Snowball Edge のデバイスオプション](#)を参照してください。

2020 年 3 月 23 日

[NFC タグ検証のサポート](#)

Snowball Edge Compute Optimized デバイス (GPU ありまたはなし) には、NFC タグが組み込まれています。これらのタグは、Android で利用可能な AWS Snowball エッジ検証アプリでスキャンできます。詳細については、「[NFC タグの検証](#)」を参照してください。

2018 年 12 月 13 日

[セキュリティグループがコンピューティングインスタンスで利用可能になりました。](#)

Snowball Edge デバイスのセキュリティグループは AWS クラウドのセキュリティグループと似ていますが、微妙な違いがいくつかあります。詳細については、[Snowball Edge デバイスのセキュリティグループ](#)を参照してください。

2018 年 11 月 26 日

[オンプレミスの更新の紹介](#)

Snowball Edge デバイスをローカル環境で実行するためのソフトウェアを更新できるようになりました。オンプレミスの更新では、インターネットに接続する必要があります。詳細については、[Snowball Edge の更新](#)を参照してください。

2018 年 11 月 26 日

[Snowball Edge の新しいデバイスオプションのご紹介](#)

Snowball Edge デバイスには、ストレージ最適化、コンピューティング最適化、GPU の使用の 3 つのオプションがあります。詳細については、[Snowball Edge のデバイスオプション](#)を参照してください。

2018 年 11 月 15 日

[新たに AWS リージョン サポート](#)

Snowball Edge デバイスが、アジアパシフィック (ムンバイ) で利用可能になりました。このリージョンでは、AWS Lambda を使用するコンピューティングインスタンスと AWS IoT Greengrass はサポートされていないことに注意してください。

2018 年 9 月 24 日

[Snowball Edge デバイスに、Amazon EC2 互換コンピューティングインスタンスのサポートを導入](#)

AWS Snowball は、Snowball Edge デバイスで実行されている [Amazon EC2 コンピューティングインスタンス](#)を使用したローカルジョブをサポートするようになりました。

2018 年 7 月 17 日

[トラブルシューティングに関するコンテンツの改善](#)

トラブルシューティングに関する章を更新し、再構成しています。

2018 年 7 月 11 日

AWS 用語集

最新の AWS 用語については、「AWS の用語集 リファレンス」の[AWS 「用語集」](#)を参照してください。

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。