



AWS Snowcone

# AWS Snowcone ユーザーガイド



# AWS Snowcone ユーザーガイド: AWS Snowcone

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

AWS Snowconeとは .....	1
ユースケース .....	2
料金 .....	3
AWS Snowcone の仕組み .....	4
AWS Snowcone ワークフロー .....	4
インポートジョブの仕組み .....	7
を使用した Snowcone と AWS ストレージサービス間のオンラインデータ転送 DataSync .....	7
Snowcone と Amazon S3 間のオフラインデータ転送 .....	8
インポートジョブストレージの場合 .....	8
コンピューティングジョブストレージの場合 .....	8
デバイスの仕様 .....	10
[概要] .....	10
ハードウェアとネットワーク .....	11
上面図 .....	12
背面パネル .....	12
前面パネル .....	13
AWS Snowcone Wi-Fi 仕様 .....	14
堅牢化仕様 .....	15
Snowcone の電源およびアクセサリ .....	15
ディスクと CPU のパフォーマンス .....	16
セットアップ .....	17
にサインアップする AWS .....	17
注文する前に .....	18
ローカル環境について .....	18
特殊文字の使用 .....	19
Amazon EC2 .....	20
Snow Family デバイス上の Amazon EC2 と Amazon EC2 互換インスタンスの違い .....	21
Snowcone のコンピューティングインスタンスの料金 .....	21
前提条件 .....	21
AWS Marketplace AMIs .....	22
インスタンスからの Linux AMI の作成 .....	23
スナップショットからの Linux AMI の作成 .....	23
開始方法 .....	28
Snow Family デバイスを注文するジョブの作成 .....	28

ステップ 1: ジョブタイプを選択する .....	29
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 .....	30
ステップ 3: 機能とオプションを選択する .....	33
ステップ 4: セキュリティ、配送、通知の設定を選択する .....	33
ステップ 5: ジョブの概要を確認してジョブを作成する .....	36
ダウンロード AWS OpsHub .....	37
Snow Family デバイスを注文するためのジョブのキャンセル .....	37
Snow Family デバイスにアクセスするための認証情報の取得 .....	38
Snow Family デバイスのロック解除 .....	39
Snow Family デバイスのロック解除のトラブルシューティング .....	41
Snow Family デバイスの再起動 .....	42
AWS OpsHub を使用してデバイスを管理する .....	47
Snow Family デバイス AWS OpsHub 用の のダウンロード .....	48
デバイスのロック解除 .....	49
デバイスをローカルでロック解除する .....	49
デバイスをリモートでロック解除する .....	52
の署名の検証 AWS OpsHub .....	56
AWS サービスの管理 .....	59
コンピューティングインスタンスのローカル使用 .....	60
NFS インターフェイスの管理 .....	68
AWS DataSync を使用してファイルを に転送する AWS .....	76
ファイルの転送 .....	77
デバイスの管理 .....	78
デバイスの再起動 .....	78
デバイスのシャットダウン .....	79
デバイスのエイリアスの編集 .....	80
更新の取得 .....	81
プロファイルの管理 .....	82
デバイスの NTP タイムサーバーの設定 .....	84
AWS Snowball Edge クライアントの使用 .....	87
Snowball Edge クライアントのダウンロードとインストール .....	87
Snowball Edge クライアントのコマンド .....	87
Snowball Edge クライアント用プロファイルの設定 .....	89
NFC 検証の QR コードの取得 .....	90
AWS Snowcone デバイスのロック解除 .....	90
Snowcone の更新 .....	91

認証情報の取得 .....	94
Snowcone Device でのサービスの開始 .....	95
Snowcone Device でのサービスの停止 .....	96
データ転送の証明書の取得 .....	96
AWS Snowcone ログ .....	97
デバイスステータスの取得 .....	99
サービスステータスの取得 .....	101
AWS DataSync AMI の起動 .....	102
NFS の開始とアクセスの制限 .....	105
NFS 実行中に NFS 共有へのアクセスを制限する .....	106
Amazon S3 バケットのエクスポートパスの取得 .....	107
ローカル AWS オペレーターのデバッグを有効にする .....	107
ローカル AWS オペレーターのデバッグを無効にする .....	107
ダイレクトネットワークインターフェイスの作成 .....	107
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 .....	108
ダイレクトネットワークインターフェイスの更新 .....	108
ダイレクトネットワークインターフェイスの削除 .....	109
機能ステータスのチェック .....	109
機能ステータスの変更 .....	110
タイムサーバーの設定 .....	111
タイムソースのチェック .....	111
Snow Device Management を使用したデバイスの管理 .....	114
Snow Family デバイスの注文時の Snow Device Management 状態の選択 .....	115
Snow Device Management のアクティブ化 .....	116
Snow Device Management のアクセス許可を IAM ロールに追加する .....	117
Snow Device Management CLI コマンド .....	118
タスクを作成する .....	119
タスクのステータスを確認する .....	120
デバイス情報を確認する .....	121
Amazon EC2 互換インスタンスのステータスを確認する .....	123
タスクのメタデータを確認する .....	125
タスクをキャンセルする .....	126
コマンドと構文を一覧表示する .....	127
リモート管理可能なデバイスを一覧表示する .....	128
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 .....	129
利用可能なリソースを一覧表示する .....	130

デバイスタグまたはタスクタグを一覧表示する .....	131
タスクをステータス別に一覧表示する .....	132
タグを適用する .....	133
タグを削除する .....	134
AWS サービスの使用 .....	135
Amazon EC2 をコンピューティングに使用する .....	135
概要 .....	136
前提条件 .....	137
コンピューティングインスタンスを含むジョブを作成する .....	138
コンピューティングインスタンスのネットワーク設定 .....	143
SSH を使用して Snowcone のコンピューティングインスタンスに接続する .....	150
コンピューティングインスタンスの Snowcone クライアントのコマンド .....	150
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 ...	156
Amazon EC2 互換エンドポイントの使用 .....	166
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 .....	182
EC2 互換インスタンスでのブロックストレージの使用 .....	184
Snow デバイスのセキュリティグループ .....	185
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 .....	186
Amazon ECS のトラブルシューティング .....	188
DataSync を使用してファイルを転送する .....	190
NFS インターフェイスの管理 .....	191
Snow Family デバイスの NFS 設定 .....	192
EC2 互換インスタンス AWS IoT Greengrass での の使用 EC2-compatible .....	197
Amazon EC2 互換インスタンスの設定 .....	197
AWS サービスを使用するために必要なポート .....	200
Snowcone Device の返却 .....	202
Snowcone Device の切断 .....	202
デバイスのデータの保護 .....	203
の保護 AWS Snowcone .....	203
NFC タグの検証 .....	204
ジョブステータスについて .....	206
通知 .....	208
Snow が Amazon SNS を使用する仕組み .....	208
Snow ジョブのステータス変更の SNS トピックを暗号化する .....	208
カスタマーマネージド KMS キーポリシーの設定 .....	209
SNS 通知の例 .....	210

注文プロセスについて .....	223
配送プロセスについて .....	223
Snowcone デバイスの返却 .....	223
の使用 AWS Management Console .....	223
コンソールから Snowcone を注文する .....	224
ジョブ管理 API の使用 .....	224
JMAPI の一般的な用途 .....	224
JMAPI 必須の文字列 .....	224
JMAPI エンドポイント .....	225
JMAPI CLI コマンド .....	225
例 .....	226
EC2 ジョブ .....	227
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 .....	228
コンソールを使用してジョブを作成する .....	229
AWS CLI を使用してジョブを作成する .....	229
配送に関する考慮事項 .....	230
配送用 AWS Snowcone デバイスの準備 .....	230
リージョンごとの配送制限 .....	231
AWS Snowcone デバイスの配送 .....	231
配送業者 .....	232
Snowcone デバイスの更新 .....	240
前提条件 .....	241
更新をダウンロードする .....	241
更新をインストールする .....	243
SSL 証明書を更新する .....	246
Snow Family デバイスの Amazon Linux 2 AMI を更新する .....	247
ベストプラクティス .....	249
セキュリティ .....	249
ネットワーク .....	250
リソース管理 .....	250
EC2 互換インスタンスの管理 .....	250
パフォーマンス .....	251
Snowcone のクォータ .....	252
コンピューティングリソースのクォータ .....	252
Snowcone デバイスの配送に関する制限事項 .....	253

インポート用に返却される Snowcone デバイスの処理の制限 .....	253
使用可能 AWS リージョン .....	254
トラブルシューティング .....	255
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 .....	255
ネットワークの問題 .....	255
IP アドレスが 0.0.0.0 と表示される .....	255
デバイスのロックを解除できない .....	256
Datasync での EC2 互換インスタンスの問題 .....	256
エラー: インスタンスの起動に失敗しました .....	256
データ転送の問題 .....	256
サーバーによってアクセスが拒否されました .....	256
データ転送中に接続がタイムアウトする .....	257
Spawn Showmount ENOENT .....	257
Snow Family デバイスの返却に関する問題のトラブルシューティング .....	258
API リファレンス .....	259
ドキュメント履歴 .....	260
AWS 用語集 .....	266
.....	cclxvii

# AWS Snowconeとは

AWS Snowcone は、エッジコンピューティングとデータ転送用のポータブルで堅牢、安全なデバイスです。Snowcone デバイスを使用して、オフラインでデバイスを に配送するか AWS クラウド、を使用してオンラインでデータを収集、処理 AWS、および に移動できます AWS DataSync。

制約の厳しい (データセンター以外の) エッジ環境や一貫したネットワーク接続がない場所でアプリケーションを実行するのは難しいことがあります。こうした場所では、データセンターの IT 機器に必要なスペース、電力、冷却機能が不足しがちです。

Snowcone には 2 つのタイプがあります。

- Snowcone — Snowcone は、2 つの vCPU、4 GB のメモリ、8 TB のハードディスクドライブ (HDD) ベースストレージを備えています。
- Snowcone SSD — Snowcone SSD は、2 つの vCPU、4 GB のメモリ、14 TB のソリッドステートドライブ (SSD) ベースストレージを備えています。

2 つの CPU とテラバイトのストレージを備えた Snowcone デバイスは、Amazon Elastic Compute Cloud (Amazon EC2) インスタンスを使用してデータを安全に格納する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。

Snowcone デバイスは小さい (8.94 インチ x 5.85 インチ x 3.25 インチ / 227 mm x 148.6 mm x 82.65 mm) ため、工場内の機械の隣に配置してデータを収集、フォーマットし、ストレージと分析 AWS のために に戻すことができます。Snowcone デバイスの重さは約 4.5 lbs. (2kg) ですので、リュックに入れて持ち歩いたり、バッテリーで操作したり、Wi-Fi インターフェイスでセンサーデータを収集したりすることができます。

## Note

Wi-Fi は北米 AWS リージョン のでのみ利用できます。

Snowcone デバイスは、ネットワークファイルシステム (NFS) サポートとのインターフェイスを提供します。Snowcone デバイスは、オンプレミスの Windows、Linux、macOS サーバーからのデータ転送、および NFS インターフェイスを介したファイルベースのアプリケーションからのデータ転送をサポートしています。

と同様に AWS Snowball、AWS Snowcone には複数のセキュリティ暗号化機能レイヤーがあります。これらのサービスのいずれかを使用して、データを収集、処理、に転送し AWS、Amazon EC2-compatible インスタンスを使用する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。Snowcone は、最大で数十テラバイトのデータ移行に対応するように設計されています。Snowball Edge デバイスを収容できない、スペースに制約のある環境でご使用いただけます。

## ユースケース

デバイスは次のユースケース AWS Snowcone に使用できます。

- エッジコンピューティングアプリケーションの場合、データを収集し、そのデータを処理して即時にインサイトを取得し、AWS へオンラインでデータを転送する。
- オンラインのセンサーやマシンによって継続的に生成されるデータを、ファクトリーまたは他のエッジロケーション AWS の に転送します。
- メディア、科学、またはその他のコンテンツを AWS ストレージサービスからパートナーや顧客に配信するため。
- メディア、科学、その他のコンテンツを、エッジロケーションから AWS に転送してコンテンツを集約する。
- データの転送準備が整っている 1 回限りのデータ移行シナリオでは、Snowcone は、デバイスを返送 AWS クラウド することで、最大 8 TB または 14 TB のデータを に転送する迅速かつ低コストの方法を提供します AWS。

モバイル展開の場合、Snowcone デバイスは指定されたバッテリー電源で動作できます。CPU 使用率が 25% の軽いワークロードの場合、デバイスはバッテリーで最大 6 時間稼働できます。Snowcone デバイスの Wi-Fi インターフェイスを使用して、ワイヤレスセンサーからデータを収集できます。AWS Snowcone デバイスは、低電力、ポータブル、軽量、耐振動性を備えているため、さまざまなリモートおよびコーステの場所で使用できます。

### Note

Wi-Fi は北米 AWS リージョン の のみ利用できます。

## 料金

Snowcone デバイスは、使用ごとのご注文が可能であり、最大 4 年間保持できます。AWS Snowcone 料金と手数料については、[AWS Snowcone 「の料金」](#)を参照してください。

# AWS Snowcone の仕組み

AWS Snowcone は、エッジコンピューティングとデータ転送に使用するポータブルデバイスです。開始するには、転送する必要があるデータの量と必要なコンピューティングパフォーマンス AWS Management Console に基づいて、で 1 つ以上の Snowcone デバイスをリクエストします。選択した Amazon Simple Storage Service (Amazon S3) バケット、データ、Amazon Elastic Compute Cloud (Amazon EC2) Amazon マシンイメージ (AMI) は、デバイスに自動的に設定、暗号化、プリインストールされます。AWS DataSync エージェントは、デバイスが配送される前にプリインストールされています。

デバイスが到着したら、オンプレミスネットワークに接続し、動的ホスト構成プロトコル (DHCP) を使用して、IP アドレスを手動または自動で設定します。Snowcone デバイスを管理するためのグラフィカルユーザーインターフェイス (GUI) アプリケーション AWS OpsHub for Snow Family である をダウンロードしてインストールする必要があります。ノートパソコンなどの、Windows または macOS クライアントマシンにインストールできます。

デバイスを開いて AWS OpsHub ロック解除すると、デバイスとそのシステムメトリクスを示すダッシュボードが表示されます。その後、AWS OpsHub を数回クリックするだけで、インスタンスを起動してエッジアプリケーションをデプロイしたり、データをデバイスに移行したりできます。

コンピューティングジョブまたはデータ転送ジョブが完了し、デバイスを返却する準備ができると、E Ink 配送ラベルは自動的に返送先住所を更新し、Snowcone デバイスが正しい AWS 施設に配送されるようにします。デバイスが発送されると、Amazon Simple Notification Service (Amazon SNS) から送信されたメッセージ、生成されたテキストや E メール、または直接コンソールにより、追跡ステータスを受け取ることができます。

## トピック

- [AWS Snowcone ワークフロー](#)
- [インポートジョブの仕組み](#)
- [インポートジョブストレージの場合](#)
- [コンピューティングジョブストレージの場合](#)

## AWS Snowcone ワークフロー

3 種類のジョブタイプを作成できます。ユースケースによってジョブタイプが異なる場合も、デバイスの注文、受領、返却のワークフローはすべて次のとおりです。

## ワークフロー

1. ジョブの作成 – 各ジョブでは、AWS Snow ファミリーマネジメントコンソール またはプログラムでジョブ管理 API を使用して、デバイスタイプを選択します (Snowcone または Snowcone SSD、ユースケースによって異なります)。ジョブのステータスは、AWS Management Console または Snowcone API を使用して追跡できます。
2. デバイスはジョブの準備が整いました。AWS はジョブ用の AWS Snowcone デバイスを準備し、ジョブのステータスは「Snowcone の準備」に変わります。
3. リージョンのキャリアによるデバイスの配送 – ここからは配送キャリアが引継ぎ、ジョブのステータスは [In transit to you] (お客様へ配送中) になります。追跡番号と追跡ウェブサイトへのリンクは、AWS Snow ファミリーマネジメントコンソール コンソールまたはジョブ管理 API で確認できます。お客様のリージョンのキャリアについての詳細は、[の配送に関する考慮事項 AWS Snowcone](#) を参照してください。
4. Snowcone 用の電源を入手する – 最小のフットプリントを維持するために、Snowcone デバイスには電源が付属していません。Snowcone は 45 ワットの USB-C 接続電源を使用します。ポータブルバッテリーで駆動することもできます。詳細については、「[AWS Snowcone 電源とアクセサリ](#)」を参照してください。
5. デバイスの受信 – 数日後、リージョンの配送業者は、ジョブの作成時に指定した住所に AWS Snowcone デバイスを配送します。ジョブのステータスは、[Delivered to you] (お客様へ配送済み) に変わります。デバイスは、それ自体が独自の配送用コンテナであるため、梱包されていません。
6. Snow Family アプリケーション用の 認証情報を取得し、AWS OpsHub または Snowball Edge クライアントをダウンロードする – 認証情報、ジョブマニフェスト、マニフェストのロック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します。
  - デバイス用のマニフェストは、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にあるときに、ジョブ管理 API で取得できます。マニフェストは、デバイスへのアクセスを認証するために使用されます。マニフェストは暗号化されており、復号できるのは解除コードだけです。
  - 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。お客様の施設にある間、デバイスへ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、安全な場所に保管することをお勧めします。
  - AWS OpsHub for Snow Family は、Snowcone を含む Snow Family デバイスを管理するためのアプリケーションです。AWS OpsHub for Snow Family GUI は Snowcone デバイスの設定と管

理をサポートし、エッジコンピューティングワークロードを迅速に実行して、AWS クラウドへデータを移行します。数回クリックするだけで、AWS OpsHub を使用して Snowcone デバイスのロック解除と設定、データのドラッグアンドドロップ、アプリケーションの起動、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。Windows または macOS クライアントマシン (ノートパソコンなど) にダウンロードしてインストールできます。AWS OpsHubの使用にコストはかかりません。

リソース AWS OpsHub から をダウンロードします。 [AWS Snowball](#)の詳細については、AWS OpsHub 「」を参照してください[AWS OpsHub for Snow Family を使用してデバイスを管理する](#)。

- 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。
7. 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。
  8. デバイスの電源オン – 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。
  9. 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て、AWS Snowcone デバイスの IP アドレスを取得します。
  10. AWS OpsHub を使用してデバイスのロックを解除する – AWS Snowcone デバイスのロックを解除するには、デバイスの IP アドレスを入力し、マニフェストをアップロードし、ロック解除コードを入力します。はマニフェストを復 AWS OpsHub 号し、それを使用してデバイスへのアクセスを認証します。の詳細については、AWS OpsHub 「」を参照してください[AWS OpsHub for Snow Family を使用してデバイスを管理する](#)。
  11. デバイスの使用 – AWS OpsHub を使用して AWS Snowcone デバイスをセットアップおよび管理し、エッジコンピューティングワークロードをすばやく実行して にデータを転送できるようにします AWS クラウド。数回クリックするだけで、AWS OpsHub を使用して Snowcone デバイスのロック解除と設定、データのドラッグアンドドロップ、アプリケーションの起動、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。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。
  12. イーサネットケーブルまたは電源ケーブルを抜かない – データ転送中またはコンピューティング操作中は、イーサネットケーブル、USB-C 電源、バッテリーを取り外さないでください。データ転送またはコンピューティングジョブの完了後に Snowcone デバイスをオフにするには、電源ボタンを押します。
  13. デバイスの返送の準備 – 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て、デバイスの電源をオフにします。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収納スペースに収納して、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。

14.リージョンのキャリアがデバイスを に返 AWSす – キャリアが AWS Snowcone デバイスを持っている場合、ジョブのステータスは への転送中に AWS変わります。

## インポートジョブの仕組み

Snowcone を使用して、 を使用してデバイスと AWS ストレージサービス間でデータをオンラインで転送できます AWS DataSync。オンプレミスのストレージデバイスから Snowcone デバイスに、オフラインでデータを転送することもできます。

### を使用した Snowcone と AWS ストレージサービス間のオンラインデータ転送 DataSync

[AWS DataSync](#) は、インターネットまたは を介した AWS ストレージサービスとの間で大量のデータのコピーを簡素化、自動化、高速化するオンラインデータ転送サービスです AWS Direct Connect。AWS DataSync エージェントは Snowcone デバイスにプリインストールされており、デバイスと Amazon S3 バケット、Amazon Elastic File System (Amazon EFS) ファイルシステム、および Amazon FSx for Windows File Server 間でデータを転送するために使用します。 は、ファイルとオブジェクトの移動、データ転送のスケジュール、転送の進行状況のモニタリング、データの暗号化、データ転送の検証、および問題に関する顧客への通知 DataSync を自動的に処理します。

DataSync エージェントは、Snowcone ジョブの準備中に Amazon マシンイメージ (AMI) として Snowcone デバイスにプリインストールされます。データを にオンラインで転送するには AWS、Snowcone デバイスを外部ネットワークに接続し、AWS OpsHub for Snow Family または AWS Command Line Interface ( AWS CLI) を使用して DataSync エージェント AMI を起動します。AWS Management Console または を使用して DataSync エージェントをアクティブ化します AWS CLI。次に、AWS Snowcone ネットワークファイルシステム (NFS) ストアと Amazon S3、Amazon EFS、または Amazon FSx の間でオンラインデータ転送タスクを設定します。

Snowcone で DataSync 実行されている は、次の目的で使用できます。

- エッジコンピューティングアプリケーションの場合、データを収集し、そのデータを処理して即時にインサイトを取得し、AWSへオンラインでデータを転送する。
- オンラインのセンサーまたはマシンが継続的に生成するデータを、ファクトリーまたはその他エッジロケーションにある AWS に転送する。

- メディア、科学、またはその他のコンテンツを AWS ストレージサービスからパートナーや顧客にオンラインで配信するため。
- メディア、科学、その他のコンテンツを、オンラインでエッジロケーションから AWS に転送してコンテンツを集約する。

ワンタイムエッジコンピューティングまたはデータ転送ワークフロー、またはワイドエリアネットワーク (WAN) リンクや不適切な WAN 帯域幅のないエッジロケーションの Snowcone ワークフローの場合は、データ転送を完了 AWS するために Snowcone デバイスを に返送することをお勧めします。

## Snowcone と Amazon S3 間のオフラインデータ転送

オフラインデータインポートジョブでは、Snowcone デバイスをオンプレミスネットワークに接続し、AWS OpsHub を使用してデバイスのロックを解除します。[AWS Snowball リソースページ](#) AWS OpsHub から をダウンロードします。NFS インターフェイスを使用して、オンプレミスストレージデバイスから Snowcone デバイスにデータをコピーできます。Snowcone デバイスにデータをコピーすると、デバイスの E Ink 配送ラベルにより、デバイスが自動的に正しい AWS 施設に送信されるようになります。Snowcone デバイスを追跡するには、Amazon SNS で生成されるテキストメッセージ、E メール、およびコンソールを使用します。

## インポートジョブストレージの場合

Snowcone デバイス内部には、8 TB または 14 TB のディスクストレージが装備されています。このディスクストレージは、ローカルの Amazon Elastic Block Store (Amazon EBS) ポリユームプレゼンテーションを通じて、内部ネットワークファイルシステム (NFS) サービスまたはローカルの Amazon EC2 互換インスタンスで使用できます。NFS ストレージポリユームには 8 TB または 14 TB を使用できます。容量が最適化された HDD (sbg1) Amazon EBS ストレージポリユームには 150 GB を使用できます。

## コンピューティングジョブストレージの場合

ジョブタイプがローカルコンピューティングの場合、合計 8 TB または 14 TB のローカル容量に最適化された HDD (sbg1) Amazon EBS ポリユームを作成して、Amazon EC2 互換インスタンスにアタッチします。Amazon EBS ポリユームを使用すると、ローカルの Amazon EC2 互換インスタンスは、ルートポリユーム単体よりも多くのローカル容量にアクセスできます。これはローカルストレ

ジのみであるため、Amazon EBS ボリュームに書き込まれたデータは、Amazon S3 にインポートできない AWS ため、デバイスが返されると失われます。 Amazon S3

 Note

NFS サーバーはコンピューティングジョブには使用できません。との間でデータをインポートまたはエクスポートする必要がある場合 AWS クラウド、または AWS DataSync エージェントを実行する必要がある場合は、注文時にローカルコンピューティングジョブタイプを選択しないでください。

# AWS Snowcone デバイス仕様

このセクションでは、ハードウェア、ネットワーク、Wi-Fi、電源の AWS Snowcone デバイスの仕様と要件について説明します。

## トピック

- [特徴と仕様の概要](#)
- [ハードウェアとネットワーク](#)
- [AWS Snowcone Wi-Fi 仕様](#)
- [堅牢化仕様](#)
- [AWS Snowcone 電源とアクセサリ](#)
- [ディスクと CPU のパフォーマンス](#)

## 特徴と仕様の概要

次の表に、Snowcone デバイスの特徴と仕様の概要を示します。

項目	仕様
用途	産業用 IoT、輸送、ヘルスケア IoT、コンテンツ配信、戦術的エッジコンピューティング、物流、自動運転車、データ移行
デバイスサイズ	長さ 9 インチ、幅 6 インチ、高さ 3 インチ (227 mm × 148.6 mm × 82.65 mm)
デバイスの重量	4.5 ポンド。Snowcone SSD の場合は (2.04 kg)、Snowcone SSD の場合は 4.6 ポンド (2.09 kg)
ストレージキャパシティ	8 TB 使用可能 (Snowcone)、14 TB 使用可能 (Snowcone SSD)
オンボードコンピューティングオプション	Amazon EC2 Amazon マシンイメージ (AMI)
暗号化	はい、256 ビット

項目	仕様
Network File System (NFS) を介した転送	はい
Amazon S3 API を介した転送	いいえ
ポータブル性	バッテリーベースのオペレーション
ワイヤレス	Wi-Fi <div style="border: 1px solid #ccc; border-radius: 10px; padding: 10px; margin-top: 10px;"> <p> Note Wi-Fi は北米 AWS リージョン のでのみ利用できます。</p> </div>
使用可能な vCPU 数	2 vCPU
使用可能メモリ	4 GB
ネットワークインターフェイス	2x 1/10 ギガビット (Gb) - RJ45
AWS DataSync エージェントはプリインストールされています	はい
標準的なジョブの存続期間	オフラインまたはオンラインのデータ転送: 数日から数週間 エッジコンピューティング: 数週間から数ヶ月
最大ジョブ長	エッジコンピューティングまたは継続的なデータ転送: 最長 360 日

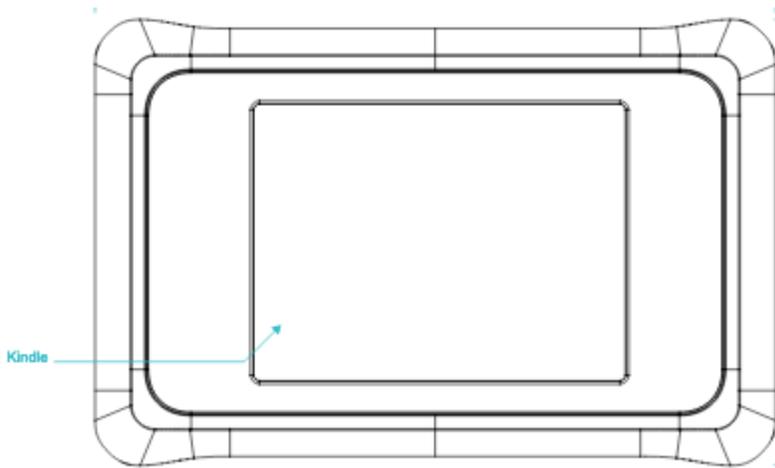
## ハードウェアとネットワーク

Snowcone デバイスは 8 TB または 14 TB 使用可能なストレージを備えています。2 つの使用可能な CPUs と 4 GB の使用可能なメモリを備えた特定の Amazon Elastic Compute Cloud (Amazon EC2) インスタンスを実行して、アプリケーションと AWS IoT Greengrass 機能をサポートします。この

セクションでは、デバイスの各種ビューに表示されるインターフェイス、電源ボタン、電源要件など、物理デバイスに関する情報について説明します。

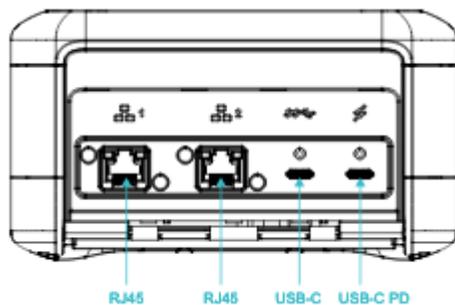
## 上面図

Snowcone デバイスの上面には、有線および無線ネットワークの両方を設定するためのオペレーターインターフェイスとして使用される、統合型 E-Ink タッチディスプレイが備えられています。また、電子配送ラベルのディスプレイとしても機能します。電子配送ラベルには、デバイスの電源を最初に入れた後に、自動的に変更されるアウトバウンドとインバウンドの両方の配送ラベルが事前にプログラムされています。



## 背面パネル

すべてのケーブル接続は背面パネルで行います。このセクションでは、各コネクタについて説明します。



## 電源

45 W 以上を供給できる適切な電源アダプターにより、右端の USB-C 接続からデバイスに電源が供給されます。

**Note**

AWS Snowcone には、可能な限り小さいフォームファクタが付属しているため、電源は含まれていません。詳細については、「[AWS Snowcone 電源とアクセサリ](#)」を参照してください。

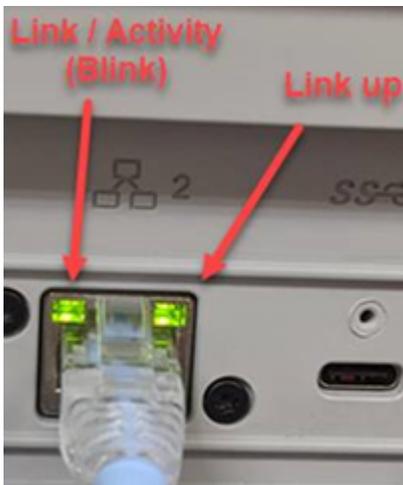
## USB

最初の USB-C 接続はアクティブではありません。

## イーサネットコネクタ 1 および 2

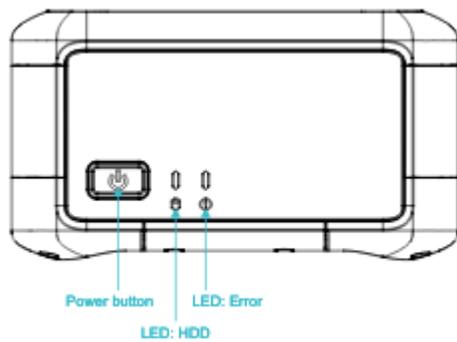
有線ネットワークの場合、Snowcone デバイスは 1 ギガビット (Gb) または 10 Gb イーサネットネットワークに対して自動ネゴシエートする 2 つのポートを提供します。

RJ45 10/1G Base-T イーサネットポート: これらのポートは、遠端接続機能に基づいて 10 Gb と 1 Gb の間で自動ネゴシエートされます。これらのポートは、1 Gb 未満の速度 (100 Mb や 10 Mb など) をネゴシエートしません。リンク LED は各コネクタの右側にあり、アクティビティ LED は左側にあります。



## 前面パネル

前面パネルには、電源ボタンとステータス LED ディスプレイが備えられています。



## 電源スイッチ

デバイスの電源を入れるには、電源ボタンを短く押します。ボタンが点灯し、E-Ink ディスプレイが進行状況バーに変わり、[Please wait] (お待ちください) と表示されます。

デバイスの電源を切るには、電源スイッチを 2 秒間押すか、E-Ink ディスプレイに [Please wait] (お待ちください) と表示されるまで長押しします。

### Note

電源ボタンを 5 秒以上押すと、デバイスの電源が強制的にオフになります。この操作により、バッファ内のデータが失われるおそれがあるため、推奨されません。AC 電源の喪失後、デバイスは電源を自動的に最後のオペレーション状態に復元します。

## ステータス LED

電源ボタンの横に 2 つのステータス LED があります。左側の LED はディスクアクティビティにより点滅し、故障条件がある場合は右側の LED が点灯します。

## AWS Snowcone Wi-Fi 仕様

AWS Snowcone は、IEEE 802.11ac 標準を備えたギガビット Wi-Fi ネットワークと、802.11abgn ネットワークをサポートしています。Snowcone のワイヤレススループットを最大化するには、2x2 802.11ac 160 MHz チャンネルを使用してください。これにより、ベースラインの 1x1 802.11bgn ネットワークよりも最大 10 倍高速になります。Snowcone の Wi-Fi の仕様は IEEE 802.11abgn +ac、2x2、MIMO、デュアルバンド 2 GHz、5 G (160MHz) です。動作周波数は 2 GHz と 5 GHz です。

物理イーサネットケーブルを使用して Snowcone をオンプレミスネットワークに接続することも、Wi-Fi を使用してワイヤレスで接続することもできます。Wi-Fi 接続を使用すると、を使用して

Snowcone デバイスを管理し AWS OpsHub、Snowcone とオンプレミスストレージデバイス間でデータを転送できます。

#### Note

Wi-Fi は北米 AWS リージョン の のみ利用できます。

## 堅牢化仕様

AWS Snowcone デバイスは、ISTA-3A、HIPAA D4169、MIL-STD-810G など、耐障害性、動作振動などの厳しい基準を満たすように設計されています。最大 3.8 フィート (1.15 メートル) からの落下に耐えられるよう設計されています。また、IP65 国際保護マーキング IEC 規格に準拠しています。つまり、防塵性 (密閉時にエンクロージャ内にほこりがない状態) と耐水性 (すべての側面の水噴射からの保護を含む) の両方を備えています。

このデバイスは、凍結 (摂氏 0 度または華氏 32 度) から砂漠のような状態 (Snowcone: 摂氏 38 度または華氏 100 度、Snowcone SSD: 摂氏 45 度または華氏 113 度) までの、幅広い動作温度範囲に対応します。保管中または配送中、Snowcone デバイスはさらに過酷な温度 (摂氏 -32 度または華氏 -25.6 度から摂氏 63 度または華氏 145.4 度) に耐えることができます。

## AWS Snowcone 電源とアクセサリ

AWS Snowcone デバイスには電源やイーサネットケーブル (RJ45) は含まれていません。これは、可能な限り小さいフォームファクタで出荷されるためです。Snowcone デバイスを、プラグイン電源またはバッテリーで稼働するオプションがあります。以下に、電源およびイーサネットケーブルを注文する際の詳細を示します。

USB-C 電源アダプタ – プラグイン電源または固定式 (モバイル以外) の動作環境では、Snowcone デバイスに USB-C 電源アダプタを使用します。Snowcone デバイスに電力を供給するには、以下の AWS テスト済み USB-C 電源アダプターのいずれかを購入できます。

- [Apple 61W USB-C 電源アダプタ](#)
- [Lenovo USB-C 65W スタンダード AC アダプタ](#)

または、45 W 以上の定格を備え、環境温度に適した USB-C 電源アダプタを使用できます。

USB-Cバッテリー – モバイルまたはポータブルの動作環境では、USB-C バッテリーを使用して、Snowcone デバイスに電源を供給します。Snowcone デバイスに電力を供給するには、[Dell Notebook Power Bank Plus – USB C、65Wh – PW7018LC](#)、または最低 45 W の定格の USB-C バッテリーを使用できます。

イーサネットケーブル (RJ45) – Snowcone デバイスをローカルネットワークに接続するには、イーサネットケーブル (RJ45) を使用します。ケーブルをお持ちでない場合は、購入する必要があります。

## ディスクと CPU のパフォーマンス

AWS Snowcone は、専用のエッジコンピューティング/データ転送デバイスです。ディスクと CPU のパフォーマンスは、さまざまな要因によって異なります。転送のサンプルパフォーマンスに関する数字の一部は「[Snowcone のパフォーマンス](#)」で確認できます。

# AWS のセットアップ AWS Snowcone

最初の AWS Snowcone ジョブを作成する前に、以下の手順に従って環境を適切に準備してください。

トピック

- [にサインアップする AWS](#)

## にサインアップする AWS

Amazon Web Services (AWS) にサインアップすると、アカウントは自動的にすべての AWS サービスにサインアップされます。は、使用するサービスに対して AWS のみ課金します。アカウントを設定したら、を使用して AWS Snowcone デバイスを注文、設定、管理できます AWS Snow ファミリー マネジメント コンソール。Snowcone のコストおよび料金の詳細については、[AWS Snowcone の料金](#)を参照してください。

が既にある場合は AWS アカウント、AWS アカウント 番号を書き留めます。をお持ちでない場合は AWS アカウント、以下の手順に従ってください。

を作成するには 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。

次のステップに必要な AWS アカウント 番号を書き留めます。

# Snowcone Device を注文する前に

AWS Snowcone はリージョン固有のサービスであるため、ジョブを計画する前に、そのサービスがリージョンで利用可能であることを確認してください。デバイスの注文能力に影響する AWS リージョン ため、ロケーションと Amazon S3 バケットが同じ 内にあることを確認してください。Snowcone デバイスの、リージョンの国境外への出荷には制限があります。詳細については、「[リージョンごとの配送制限](#)」を参照してください。

以外のロケーションへの国際出荷 AWS リージョン は、特定のリージョン (米国からメキシコなど) の一部のロケーションについて AWS Snow ファミリーマネジメントコンソール、許可リストに登録されているお客様のでサポートされています。これらのリクエストに対応するためのターゲットの送信先、コスト、タイミングを、アカウントチームと話し合う必要があります。

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 では、各デバイスでデータを保護するために使用する暗号化キーを保護します。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。

## トピック

- [ローカル環境に関する質問](#)
- [特殊文字を使用したファイル操作](#)
- [Snowcone での Amazon EC2 の使用](#)

## 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。

転送中のデータにアクセスするか。

データの破損を防ぐため、データの転送中は Snowcone デバイスを切り離したり、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。

## 特殊文字を使用したファイル操作

ファイル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 ("|")

ファイルにこれらの文字が 1 つ以上ある場合は、AWS Snowcone デバイスにコピーする前に名前を変更します。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。名前にスペースが含まれている個々のオブジェクト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: <b>test file.txt</b>
Windows	"C:\Users\ <username>\desktop\test file.txt"</username>
Mac	/Users/<username>/test\ file.txt

オペレーティングシステム	ファイル名: <b>test file.txt</b>
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。ただし、はアクセスコントロールリスト (ACL) AWS DataSync を保持します。ACLs 詳細については、「[AWS DataSync ユーザーガイド](#)」の「[ガメタデータと特殊ファイル DataSync を処理する方法](#)」を参照してください。

## Snowcone での Amazon EC2 の使用

このセクションでは、AWS Snowcone デバイスでの Amazon EC2 互換コンピューティングインスタンスの使用の概要について説明します。

仮想マシン (Amazon マシンイメージ、AMI) として管理およびデプロイされたアプリケーションをエッジで実行する場合は、Amazon EC2 互換インスタンスを使用する必要があります。Snowcone は、snc1.micro (1 CPU と 1 GB RAM)、snc1.small (1 CPU と 2 GB RAM)、snc1.medium (2 CPU と 4 GB RAM) の 3 つのインスタンスで SNC1 インスタンスタイプをサポートします。Snowcone における EC2 互換インスタンスの Support により、Amazon EC2 でアプリケーションを構築およびテストできます。、AWS Snowball SDK、またはを使用して AWS Management Console、AWS Snowcone ジョブの作成中に EC2-compatible AMIs を有効にしてプロビジョニングできます AWS CLI。

サポートされている Amazon EC2 インスタンスタイプ

コンピューティングジョブには、次の Amazon EC2 インスタンスタイプを使用します。

snc1.micro — 1 CPU コア、1 GB RAM

snc1.small — 1 CPU コア、2 GB RAM

snc1.medium — 2 CPU コア、4 GB RAM

を使用して AWS OpsHub、Snowcone でインスタンスを管理します。[AWS Snowball リソース](#) ウェブサイト AWS OpsHub から をダウンロードします。を使用してデバイスのロックを解除したら

AWS OpsHub、Amazon EC2 ページに移動します。[インスタンスの作成] を選択して、ジョブ作成時にデバイスにプリロードした AMI に基づいて、EC2 互換インスタンスを作成します。その後、インスタンスに接続してエッジアプリケーションを実行できます。また、EC2-compatible インスタンスを開始、停止、終了、再起動するためのワンクリックボタン AWS OpsHub も用意されています。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。

デバイスの使用が終了したら、に戻します AWS。デバイスをインポートジョブで使用した場合、ファイルインターフェイスを使用して転送されたデータは、Snowcone NFS インターフェイスを使用して、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

Snowcone 上で実行されるコンピューティングインスタンスのデータが、AWS にインポートされることはありません。

## 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

## Snowcone 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 Snowcone の料金](#)」を参照してください。

## 前提条件

ジョブを作成するには、以下の点を考慮する必要があります。

- 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 を AWS アカウントで所有する必要があります。現在、サポートされている AMI は、[Amazon Linux 2](#)、[CentOS 7 \(x86\\_64\) - with Updates HVM](#)、[Ubuntu 16.04 LTS - Xenial \(HVM\)](#) イメージに基づいています。これらのイメージは、[AWS Marketplace](#) から取得できます。

ジョブクエストに AMI を追加する前に、1 つ以上のサポートされる AMI が AWS アカウントにあることを確認してください。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。詳細については、「[AWS Marketplace AMIs](#)」を参照してください。

- AMI はいずれも 1 つのボリュームを使用した Amazon Elastic Block Store (Amazon EBS) ベースである必要があります。
- Snowcone で実行されるコンピューティングインスタンスに接続している場合は、Secure Shell (SSH) を使用する必要があります。これを行うには、最初にキーペアを追加します。

## 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	b3dl4415quatdnd14qa6kcu45
CentOS 7 (x86-64)	aw0evgkw8e5c1q413zgy5pjce
Ubuntu 16.04 LTS	csv6h7oyg29b7epjzg7qdr7no
Amazon Linux 2	avyfzznywe""kg15qv5f57ska
Ubuntu 20.04 LTS	a8jyyfn4hjutohctm41o2z18m
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

## インスタンスからの Linux AMI の作成

コンソールまたはコマンドラインを使用して、AMI を作成できます。既存の AMI から開始して、インスタンスを起動してカスタマイズし、そこから新しい AMI を作成して、最後に新しい AMI のインスタンスを起動します。

コンソールを使用してインスタンスから AMI を作成するには

1. 新しい AMI の開始点として、適切な EBS-backed AMI を選択し、起動前に必要に応じて設定します。詳細については、[インスタンス起動ウィザードを使用したインスタンスの起動](#)を参照してください。

2. [起動] を選択して、選択した EBS-backed AMI のインスタンスを起動します。デフォルト値をそのまま使ってウィザードを完了します。詳細については、[インスタンス起動ウィザードを使用したインスタンスの起動](#)を参照してください。
3. インスタンスの実行中に、それに接続します。必要に応じて、インスタンスで次のアクションを実行し、インスタンスをカスタマイズできます。
  - ソフトウェアやアプリケーションをインストールする
  - データをコピーする
  - 起動時間を短縮するために一時ファイルの消去、ハードディスクのデフラグ、占有領域の開放処理を行う。
  - 追加の Amazon EBS ボリュームをアタッチする
4. (オプション) インスタンスにアタッチされているすべてのボリュームのスナップショットを作成する。スナップショットの作成の詳細については、[Amazon EBS スナップショットの作成](#)を参照してください。
5. ナビゲーションペインで [Instances] を選択し、インスタンスを選択します。[Actions] (アクション) で、[Image] (イメージ)、[Create Image] (イメージの作成) の順に選択します

 Tip

このオプションが無効になっている場合、そのインスタンスは Amazon EBS-Backed インスタンスではありません。

6. [Create Image] ダイアログボックスで、以下の情報を指定し、[Create Image] を選択します。
  - [Image name] (イメージ名) - イメージの一意の名前。
  - [Image description] (イメージの説明) - (オプション) イメージの説明 (最大 255 文字)。
  - [No reboot] (再起動しない) - このオプションはデフォルトでは選択されません。Amazon EC2 はインスタンスをシャットダウンし、アタッチされていたすべてのボリュームのスナップショットを作成し、AMI を作成および登録して、インスタンスを再起動します。インスタンスをシャットダウンしない場合は、[No reboot] (再起動しない) を選択します。

 Warning

[再起動しない] を選択した場合は、作成されたイメージのファイルシステムの整合性は保証されません。

- [Instance Volumes] (インスタンスボリューム) - このセクションのフィールドでは、ルートボリュームを変更し、Amazon EBS およびインスタンスストアボリュームを追加できます。各フィールドについては、各フィールドの横の **i** アイコンで一時停止するとフィールドのヒントが表示されます。いくつかの重要な点を以下に一覧表示します。
  - ルートボリュームのサイズを変更するには、[Volume Type] (ボリュームタイプ) 列で [Root] (ルート) を検索します。[Size (GiB)] (サイズ (GiB)) に、必要な値を入力します。
  - [Delete on Termination] (終了時に削除) を選択した場合、この AMI から作成されたインスタンスを終了すると、Amazon EBS ボリュームが削除されます。[Delete on Termination] (終了時に削除) をオフにした場合は、インスタンスを終了しても、Amazon EBS ボリュームは削除されません。詳細については、「Amazon EC2 ユーザーガイド」の「[インスタンスの削除で Amazon EBS ボリュームを保持する](#)」を参照してください。
  - Amazon EBS ボリュームを追加するには、[Add New Volume] (新しいボリュームの追加) を選択します (これにより新しい行が追加されます)。[Volume Type] で [EBS] を選択し、行のフィールドに入力します。作成した AMI からインスタンスを起動すると、追加したボリュームは自動的にそのインスタンスにアタッチされます。空のボリュームはフォーマットしてマウントする必要があります。スナップショットベースのボリュームはマウントする必要があります。
  - インスタンスストアボリュームの追加については、[AMI にインスタンスストアボリュームを追加する](#)を参照してください。その後新しい AMI からインスタンスを起動すると、追加されたボリュームは自動的に初期化されてマウントされます。これらのボリュームには、AMI の作成に使用された実行中のインスタンスのインスタンスストアボリュームのデータは含まれません。
7. 作成中に AMI のステータスを表示するには、ナビゲーションペインで [AMI] を選択します。最初、ステータスは保留中ですが、数分後に使用可能に変わります。
- (オプション) 新しい AMI に作成されたスナップショットを表示するには、[Snapshots (スナップショット)] を選択します。ユーザーがこの AMI からインスタンスを起動すると、Amazonはこのスナップショットを使用して、ルートデバイスボリュームを作成します。
8. 新しい AMI からインスタンスを起動します。詳細については、[インスタンス起動ウィザードを使用したインスタンスの起動](#)を参照してください。
9. 新しい実行中インスタンスには、前のステップで適用したカスタム設定がすべて含まれます。

## コマンドラインを使用してインスタンスから AMI を作成するには

次のいずれかのコマンドを使用できます。これらのコマンドラインインターフェイスの詳細については、Amazon EC2 へのアクセスを参照してください。

- [create-image](#) (AWS CLI)
- [New-EC2Image](#) (AWS Tools for Windows PowerShell)

## スナップショットからの Linux AMI の作成

インスタンスのルートデバイスボリュームのスナップショットがある場合は、AWS Management Console またはコマンドラインを使用して、このスナップショットから AMI を作成できます。

コンソールを使用してスナップショットから AMI を作成するには

1. Amazon EC2 コンソール (<https://console.aws.amazon.com/ec2/>) を開きます。
2. ナビゲーションペインで [Elastic Block Store] の [Snapshots] を選択します。
3. スナップショットを選択し、[Actions]、[Create Image] を選択します。
4. [Create Image from EBS Snapshot] (EBS スナップショットからイメージを作成する) ダイアログボックスで、フィールドに入力して AMI を作成し、[Create] (作成) を選択します。親インスタンスを再作成する場合は、親インスタンスと同じオプションを選択します。
  - Architecture: 32 ビットの場合は [i386] を、64 ビットの場合は [x86\_64] を選択します。
  - Root device name: ルートボリュームの適切な名前を入力します。詳細については、[Linux インスタンスでのデバイスの名前付け](#)を参照してください。
  - Virtualization type: この AMI から起動されるインスタンスで準仮想化 (PV) またはハードウェア仮想マシン (HVM) のいずれの仮想化を使用するかを選択します。詳細については、「[Linux AMI 仮想化タイプ](#)」を参照してください。
  - (PV 仮想化タイプのみ) Kernel ID および RAM disk ID: リストから AKI と ARI を選択します。デフォルトの AKI を選択するか、AKI を選択しない場合、この AMI を使用してインスタンスを起動するたびに AKI を指定する必要があります。また、デフォルトの AKI にインスタンスとの互換性がない場合、インスタンスのヘルスチェックが失敗する可能性があります。
  - (オプション) Block Device Mappings: ボリュームを追加するか、AMI のルートボリュームのデフォルト容量を増やします。インスタンスのボリューム増加時のファイルシステムサイズ変更の詳細については、[ボリュームサイズ変更後の Linux ファイルシステムの拡張](#)を参照してください。

## コマンドラインを使用してスナップショットから AMI を作成するには

次のいずれかのコマンドを使用できます。これらのコマンドラインインターフェイスの詳細については、[Amazon EC2 へのアクセス](#)を参照してください。

- [register-image](#) (AWS CLI)
- [Register-EC2Image](#) (AWS Tools for Windows PowerShell)

# 開始方法

このセクションでは、で最初の AWS Snowcone ジョブを作成して完了するための一般的な手順について説明します AWS Snow ファミリーマネジメントコンソール。AWS Snowcone デバイスの概要については、「」を参照してください [AWS Snowcone の仕組み](#)。

この入門ドキュメントでは、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、Snowball Edge クライアントまたは AWS OpsHub for Snow Family アプリケーションを使用して AWS Snowcone デバイスのロックを解除する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 Snowcone 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、「[AWS のセットアップ AWS Snowcone](#)」を参照してください。

の使用を開始するには AWS Snowcone、「[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。

## 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](#)
- [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 デバイスのロック解除](#)
- [Snow Family デバイスの再起動](#)

## 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、AWS 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記述するために使用する用語です。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、「[ジョブのステータス](#)」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 へのインポート] – AWS によって空の Snowcone デバイスが発送されます。デバイスをローカルネットワークに接続し、を使用してデバイスを設定します OpsHub。NFS 共有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 Amazon S3 にアップロードされます。
- [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 <a href="#">Info</a>					
	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

- [Snowcone power supply] (Snowcone 電源供給) セクションで、[I will provide my own power supply and Ethernet cable] (自分の電源とイーサネットケーブルを指定) を選択します。電源の詳細については、[AWS Snowcone の電源およびアクセサリ](#)を参照してください。
- [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。デバイスの料金については、「[AWS Snowcone の料金](#)」を参照してください。
- [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。
  - 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、「[ユーザーガイド](#)」の「[オフラインデータ転送に NFS AWS Snowcone を使用する](#)」を参照してください。

5. 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。

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

6. [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。

詳細は、本ガイドの「[コンピューティングインスタンスを含むジョブを作成する](#)」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。

7. [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. Snowcone デバイスでワイヤレスネットワークを有効にするには、[Snowcone でワイヤレスを有効にする] を選択します。
2. 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。
3. [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) リージョンで新しい 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。

### Note

Snowcone デバイスには電源コードは付属されないため、別途入手する必要があります。詳細については、[AWS Snowcone 「電源とアクセサリ」](#) 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

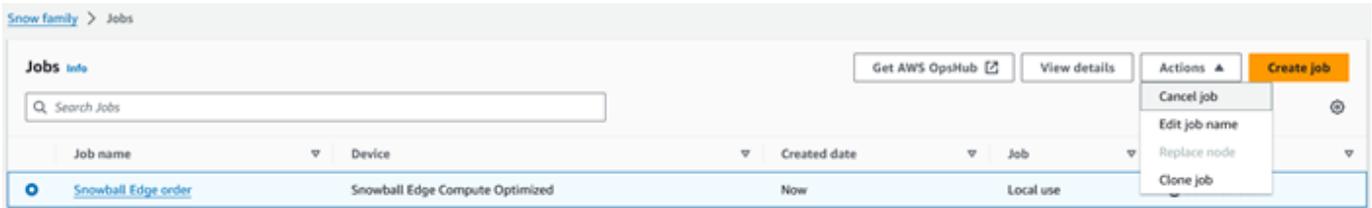
1. [AWS Snowball リソース](#) で、 をクリックします AWS OpsHub。ダウンロードリンク AWS OpsHub があるセクションで、 AWS OpsHub オペレーティングシステムにインストールする適切なダウンロードリンクを選択します。
2. [AWS OpsHub] セクションで、お使いのオペレーティングシステムの [Download] (ダウンロード) を選択し、インストール手順に従います。完了したら、[Next] を選択します。

## 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

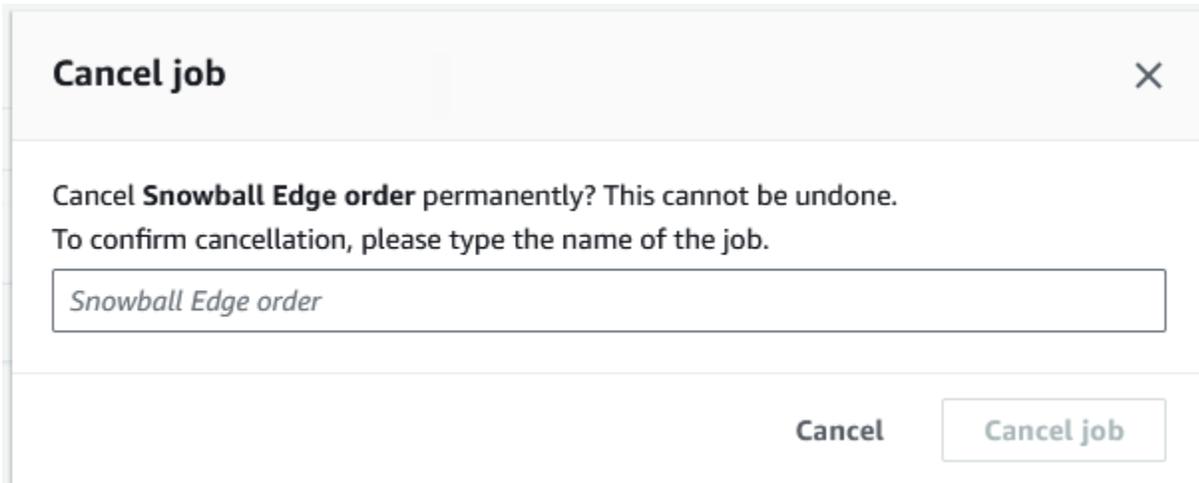
Snow Family デバイスを注文するジョブを作成したら、 を使用してジョブをキャンセルできます AWS Snow ファミリーマネジメントコンソール。ジョブをキャンセルすると、注文したデバイスを受信しません。ジョブをキャンセルできるのは、ジョブのステータスが Job created の場合のみです。ジョブがこのステータスを過ぎた後、ジョブをキャンセルすることはできません。詳細については、「[ジョブのステータス](#)」を参照してください。

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。



## 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、「[AWS Snowcone ジョブステータスについて](#)」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] が含まれます。

#### 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、「[AWS Snowcone デバイスの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: [AWS Snowball Edge クライアントの使用](#)

## 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](#)」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したら、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。[AWS 「Snowcone デバイスの仕様」](#)を参照してください。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、「[AWS Snowball Edge クライアントの使用](#)」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。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、「[AWS Snowcone デバイス仕様](#)」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. デバイスの LCD ディスプレイで Snowcone 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 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
```

## 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。

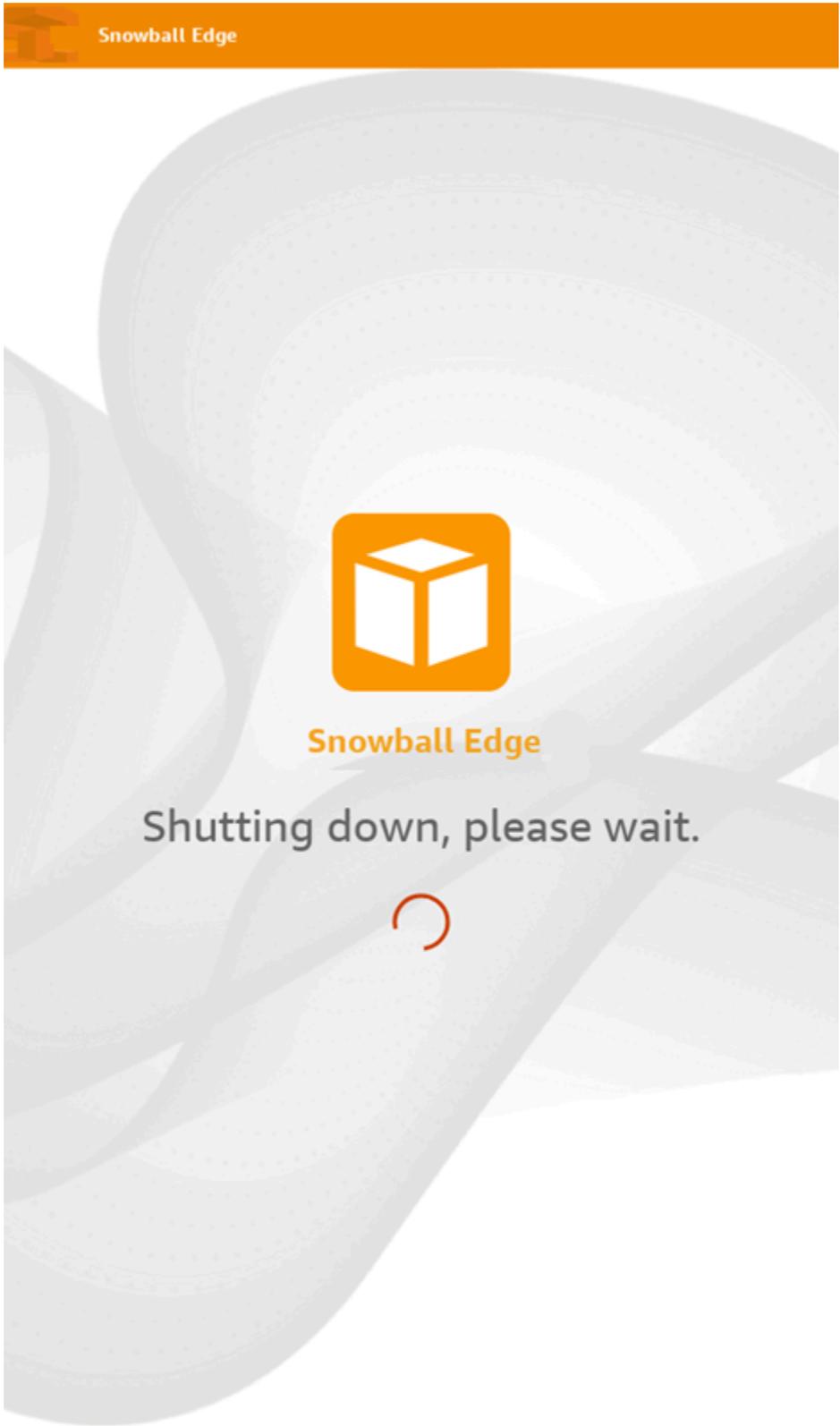
これで、Snow Family デバイスの使用を開始できます。

## 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。NFS インターフェイスを使用してデータを転送していた場合は、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停止](#)」を参照してください。

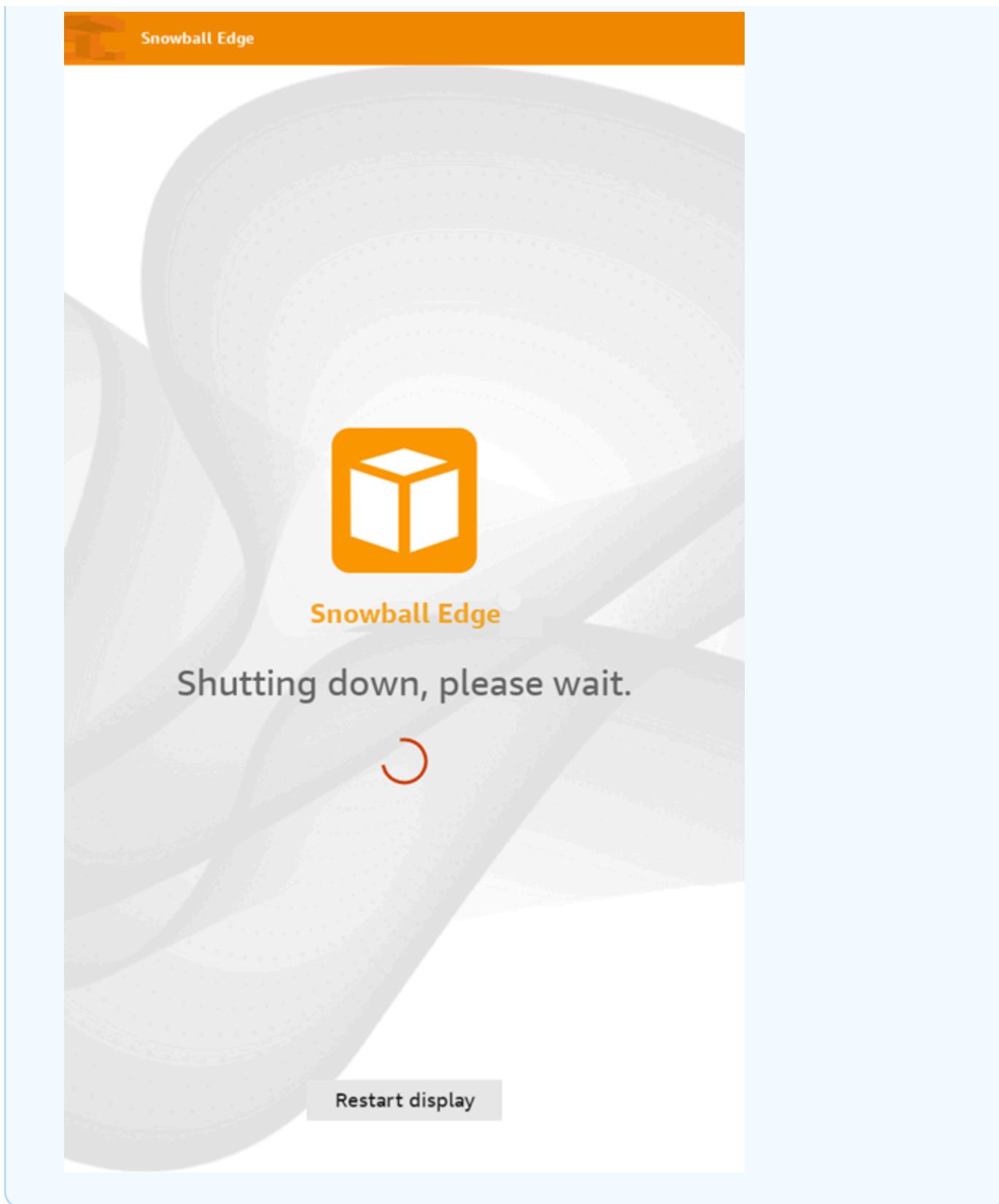
電源ボタンを使用してデバイスを再起動するには

1. デバイスとの通信がすべて終了したら、デバイスの前面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。[AWS Snowcone 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。「[AWS Snowcone デバイスのロック解除](#)」を参照してください。

# 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。

[AWS OpsHub アプリケーションをダウンロード](#)してクライアントマシンにインストールすると、AWS OpsHub は AWS Snowcone、デバイスが Wi-Fi または物理ケーブルで接続されているかどうかにかかわらず、同じネットワーク上のデバイスに接続できます。次に、デバイスを開いて AWS OpsHub ロック解除します。次に、デバイスとシステムメトリクスを示すダッシュボードが表示されます。その後、エッジアプリケーションのデプロイまたはデバイスへのデータの移行を開始できます。AWS OpsHub は、デバイス上の drag-and-drop ファイルまたはフォルダを許可することで、Snowcone デバイスへのデータ転送を簡素化します。を使用すると AWS OpsHub、デバイスに保存された内容を簡単に確認することもできます。

## トピック

- [Snow Family デバイス AWS OpsHub 用の のダウンロード](#)
- [デバイスのロック解除](#)
- [の PGP AWS OpsHub 署名の検証 \(オプション\)](#)
- [デバイスの AWS サービスの管理](#)
- [DataSync を使用してファイルを に転送する AWS](#)
- [デバイスの管理](#)
- [デバイスの NTP タイムサーバーの設定](#)

## Snow Family デバイス AWS OpsHub 用の のダウンロード

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。

The screenshot shows the AWS OpsHub documentation page. On the left is a navigation menu with links for 'What's New', 'OpsHub', 'Snowball Edge Client', 'Featured Content', 'Documentation', 'Webinars', 'Videos', 'Blog Posts', and 'Free Training'. The main content area is titled 'OpsHub' and contains a description of the tool and a list of operating systems with corresponding download links.

	OpsHub
Windows 7 or higher	<a href="#">Download</a>
Mac OS X 10.10 or higher	<a href="#">Download</a>
Linux (Ubuntu version 14 or higher, and Fedora version 24 or higher)	<a href="#">Download</a>
	<a href="#">(Signature)</a>

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、デバイスを選択して、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を表示します。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、「[デバイスの AWS サービスの管理](#)」を参照してください。

**Note**

Snowcone デバイスで使用可能なストレージ容量は、NFS サービスが開始されるまでは正確ではありません。[NFS インターフェイスの管理](#) 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. 「開始方法 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

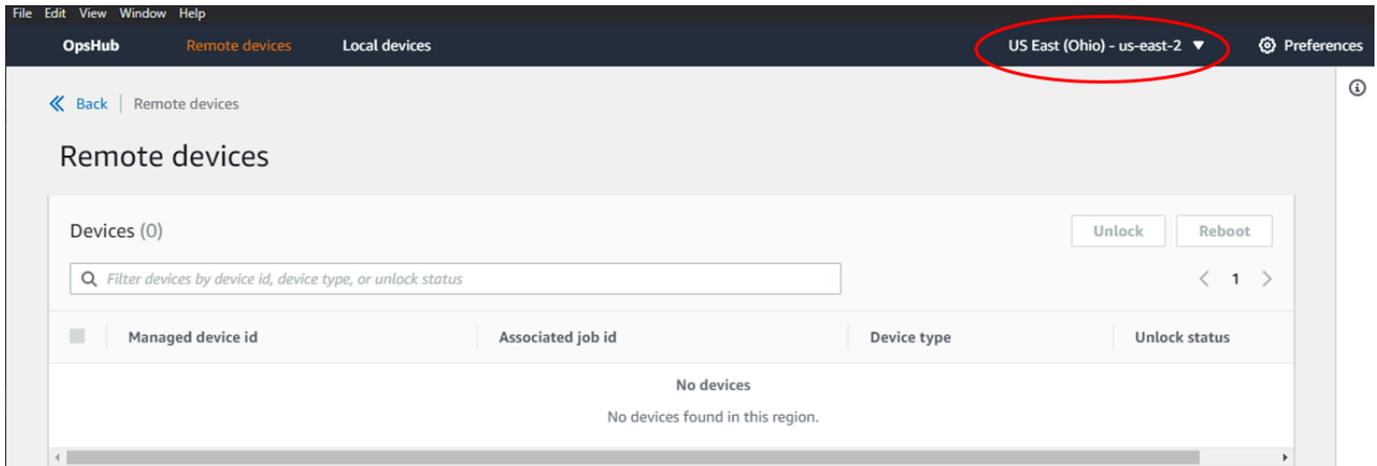
Access key

Secret key

Back

Sign in

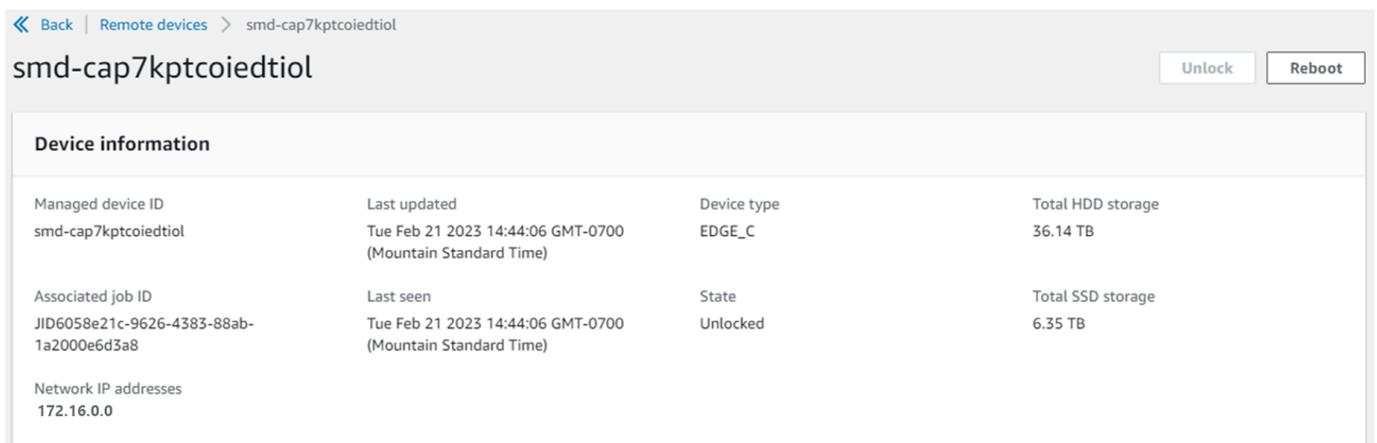
6. [リモートデバイス] タブの上部で、リモートでロック解除する Snow デバイスのリージョンを選択します。



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。

#### Note

Snowcone デバイスで使用可能なストレージ容量は、NFS サービスが開始されるまでは正確ではありません。[NFS インターフェイスの管理](#) を参照してください。



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、「[デバイスの AWS 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/5mea1VqJzd8liFpIFQnl+X7U2x6emDbM01yJWV3aMmPwhtQ7iBdt5a4x82
EF5bZJ8HSRMvANDILD/9VTN8VfUQGKFjFY2GdX9ERwvFTb47bbv9Z28V1284
4lw2w1B1007Fo02v/Y0ukrN3VHCpmJQS1IiqZbYRa0DVK6UR5QNvUlJ5fwWs
4qW9UDPhT/HDuaMrMFCejEn/7wvRUrGVtzCT9F56A1/dwRSxBejQQEb1AC8j
uuyi7gJaPdyNntR0EFTD7i02L6X2jB4YLfvGxP7Xeq1Y37t8NKF8CYTp0ry/
Wvw0iKZFbo4AkiI0aLyBCk9HBXhUKA9x06g0nhh1UFQrPGrk60RPQKqL76HA
E2ewzGDa90wLRBUAt2nRQpyNYjoASBvz/cAr3e0nuWsIzopZienrxI5ffcjY
f6UWA/OK3ITHtYHewVhseDyEqTQ4MUIWQS4NAwARAQABzTlBV1MgT3BzSHVi
IGZvciBTbm93IEZhbWlseSA8YXdzLW9wc2h1Yi1zaWduZXJAYW1hem9uLmNv
bT7CwY0EEAEIACAFA1/hGf8GCwkHCAMCBBUICgIEFgIBAAIQAQIbAwIeAQAh
CRAHgc9adPNF8RYhBDcvpelIaY930b0vqiGBz1p080XxGbcP+gPZX7LzKc1Y
w9CT3UHgkAIawOSXYktujzoYVxAz8/j3jEkCY0dKnfyqvWZDiJAXnzmXWwBg
cxg1g0GXNXCM4lAd68CmbA0LoLTaWSQX30ZbswzhbtX2ADAlOpV8RLBik7fm
bS9FyuubDRhfYRQq0fpjUGXFiEgwg6aMFxsrGLlv4QD7t+6ftFie/mxLbjR4
iMgtr8FIPXbgn05YYY/LeF4NIgX4iLEqRbAnfWjPzqQ1spFWAotIzDmZqby+
WdWThrH4K1rwtYM8sDhqRnMnqJrGFZzk7aDhVPwF+FOVMmPeEN5JRazEeUr1
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
1sS8JU5rindEsKUrkTwcG2wyZFoe1z1E8xPkLRSRN5ZbbgKsTz16l1HgCCId
Do+WJdDkWGwXmtDvzjm32EI/PVBd108ga9aPwXdhLw0dKAjZ4JrJXLUQJjRI
IVDSyM0bEH0UM6a/+mWNZazNfo0LsGwqGva6Xn5WJWlwR1S78vPNf03BQYu0
YRjaVQR+kPtB9aSAZni5sWfk6N1rRNd1Q78d067uhhejsjRt7Mja2fEL4Kb1X
nK4U/ps7X103o/VjblneZ0hJK6kAKU172tnPJTJ31Jb0xX73wsMWDYZRZVcK
9X9+GFrpwKHwKkpjpm0t/FRxNepvqR172TkgBPqGH2TM0FdB1f/uQprivqge
PBbS0JrmBIH9/anIqgtMdtcNQB/0erLdCDqI5af0uD10LcLwdJwG9/bSrfwT
TVEE3WbXmJ8pZgMz1HUizE6V2DSadV/YItk50I0jJR0VH0Hv1FMwGCEAIFzf
9P/pNi8hpEm1RphRi0VVcdQ30bH0M0gPHU5V9f1IhyCL1zU3LjYTHkq0yJD5
YDA1x01MYq3DcSM5130VBbLmuVS2GpcsTCYqlQA6h/zzMwz+/70wU0EX+EZ
/wEQA0AY8ULmcJIQWIr14V0jy1pJeD3qwj7wd+QsBzJ+mOp0B/3ZFAhQiN01
9yCDLHeiZeAmWYX90IXrNiIdcHy+WTAp4G+NaMpqE52qhbDjz+IbvLp11yDH
bYEHpjnTHXEy21bvKAJ0Kkw/2RcQ0i4dodGnq5icyYj+9gcuHvnVwbrQ96Ia
0D7c+b5T+bzFqk90nIcztMrRuhDLJnJpi70jpvQwfq/TkkZA+mzupxfSkq/Y
N9qXNEToT/VI2gn/LS0X4Ar112KxBjzNEsQkwGSiWSYtMA5J+Tj5ED0uZ/qe
omNblA1D4bm7Na8NAoLxCtAiDq/f3To9Xb18lHsnd0mfLCb/BVgP4edQKTii
C/OZH9QJ1fmN0aq7JVLQAuvQNEL88RKW6YZBqkPd3P6zdc7sWDLTMXM0d3I
e6NUvU7pW0E9NyRfUF+oT4s9wAJhAodinAi8Zi9rEfhK1VCJ76j7bcQyZe0
jXD3IJ7T+X2XA8M/BmypwMw0Soljzhwh044RAasr/fAzpKNPB318JwcQunIz
u2N3CeJ+zrsomjCpXzehwsSVq1lzaL2ureJBL0KkBgYxUJYXpbS01ax1TsFG
091dAN0s9Ej8CND37GsNnuygj0gWXB6MNgbvPs3H3zi/AbMunQ1VB1w07JX
zdM1hBQZh6w+NeiEsK1T6wHi7IhxABEBAAHCwXYEGAEIAAKFAL/hGf8CGwwA
IQkQIYHPWnTzRfEWIQQ3L6XpSGmPd9Gzr6ohgc9adPNF8TMBD/9TbU/+PVbF
ywKvwi3GL0lpY7BXn8lQaHyunMGUavm080faRR0ynkH0ZqLHCp6bIajF0fvF
b7c0Jamzx8Hg+SIId16yRpRY+fa4RQ6PNnmT93ZgWW3EbjPyJG1m0/rt03SR
+0yn4/ldlg2KfBX4ppMoPCMKUdWxGmDETXsGihwZ0gmCZqXe81K122PYkSN
JQQ+L1fjKvCaxfPKEjXYTbIbfyyhCR6NzA0VZxCrzSz2xDrYwP/V002K1xda
0ix6r2aEHf+xYEUh0aBt80HY5nXTuRReCVU789MUVtCMqD2u6amdo4BR0kWA
QNg4yavKwV+LVtyYh2Iju9VSyv4xL1Q4xKHvcAUrSH73bHG7b7jkUJckD0f4
twhjJk/Lfwe6RdnVo2WoeTvE93w+NAq2FXmvbiG7eltl0XfQecvQU3QNbRvH
U8B96W0w8UXJdvTKg4f0NbjSw7iJ3x5naixQ+rA8hLV8x0gn2LX6wvxT/SEu
mn20KX+fPtJELK7v/NheFLX1jsKLXYo4jHrkfIXNsNUhg/x2E71kAjbET3s+
t9kCtxt2iXDDZvpIbmG04QkvLFvoR0aSmN6+8fupe3e+e2yN0e6xGTuE60gX
I2+X1p1g9IduDYTpoI20XleHyyMqGeeIb4g0iisloTp5oi3EuAYRGf1XuqAT
VA19bKnpkBsJ0A==
=tD2T
-----END PGP PUBLIC KEY BLOCK-----
```

## 2. 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

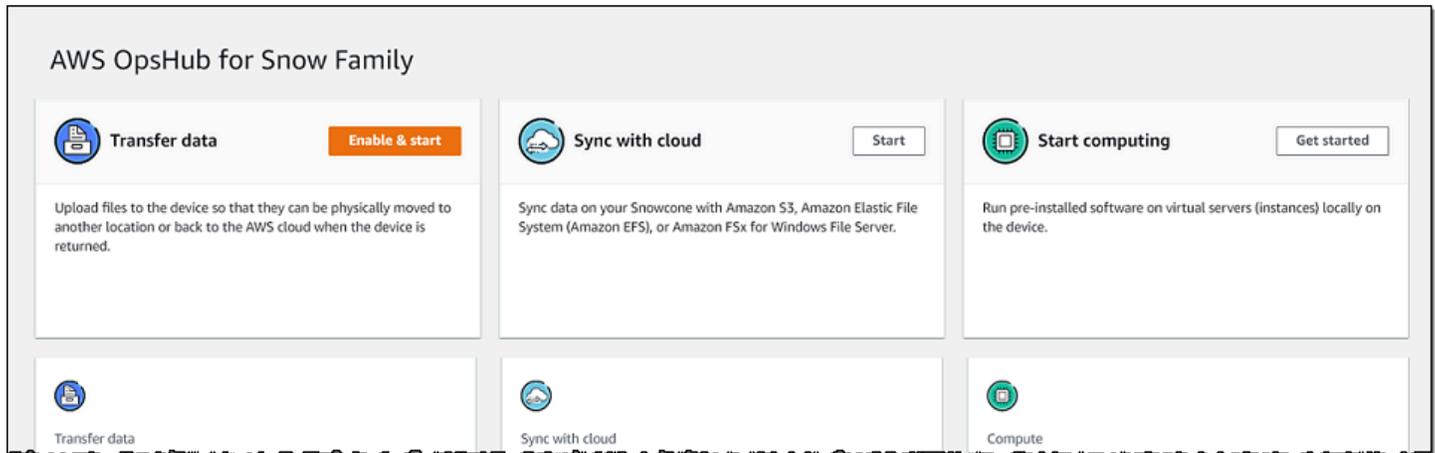
```
gpg: Signature made Mon Dec 21 13:44:47 2020 PST
gpg:          using RSA key 1655BBDE2B770256
gpg: Good signature from "AWS OpsHub for Snow Family <aws-opshub-
signer@amazon.com>" [unknown]
gpg: WARNING: This key is not certified with a trusted signature!
gpg:          There is no indication that 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。

## デバイスの AWS 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。
- AWS DataSync- オンプレミスストレージとファイルシステムや Amazon S3 などの他の AWS クラウド 場所の間で、多数のファイルを転送します。
- 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。



## 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](#)
- [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。

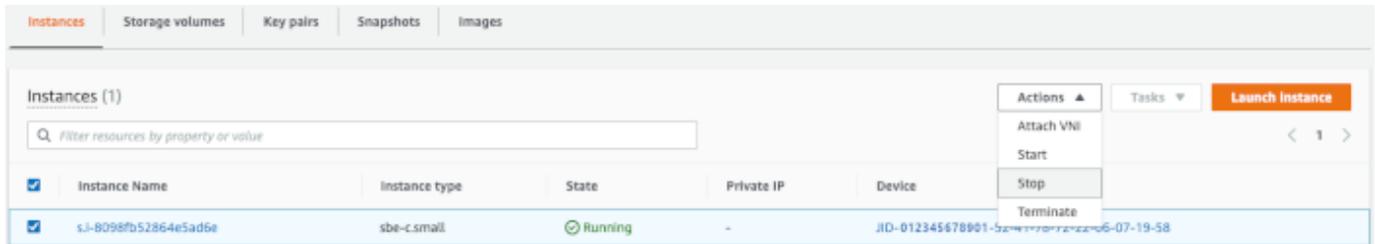
Amazon EC2 互換インスタンスを停止するには

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

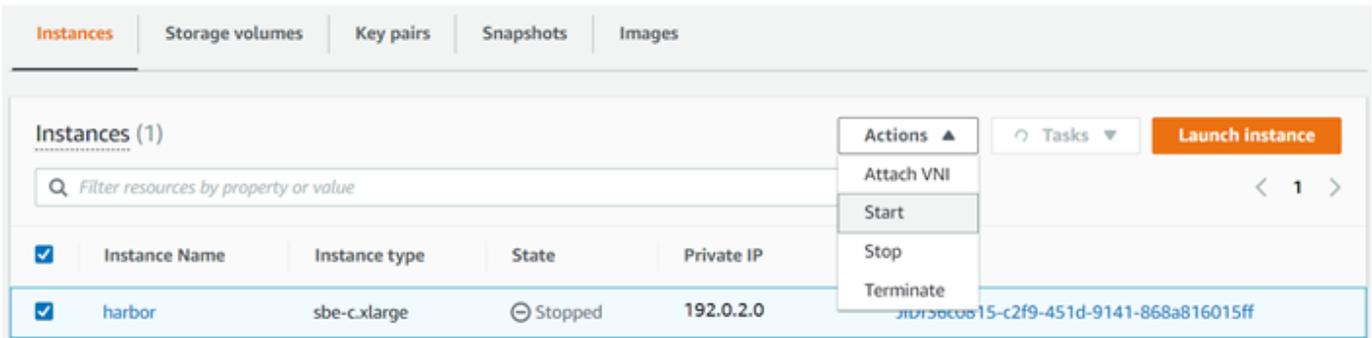
を使用して Amazon EC2-compatible インスタンスを起動するには、次の手順に従います AWS OpsHub。

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

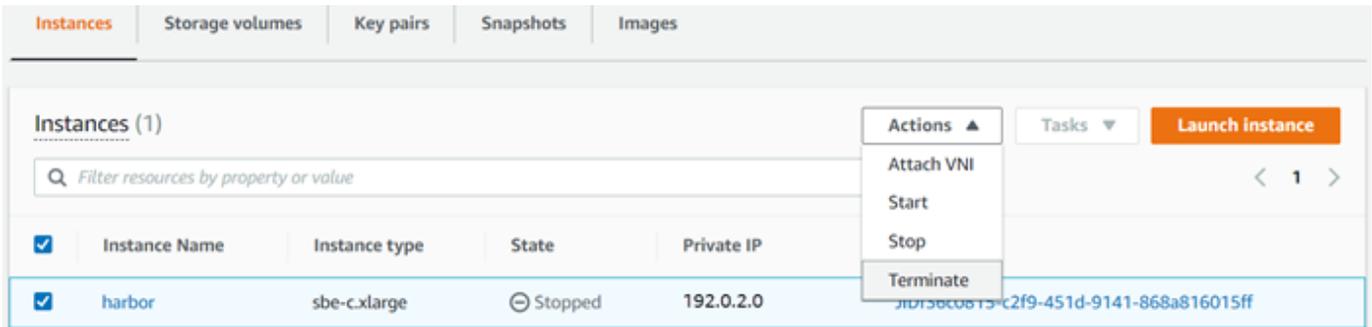
1. AWS OpsHub ダッシュボードでコンピューティングを開きます。
2. ナビゲーションペインで、[Compute (EC2)] ((EC2) のコンピューティング) ページ、[Key Pairs] (キーペア) タブの順に選択します。Amazon EC2 コンソールにリダイレクトされ、キーペアを作成、インポート、管理できます。
3. キーペアを作成してインポートする方法については、[Amazon EC2 ユーザーガイド](#)の「[Amazon EC2 キーペアと Linux インスタンス](#) Amazon EC2」を参照してください。

## Amazon EC2 互換インスタンスを終了する

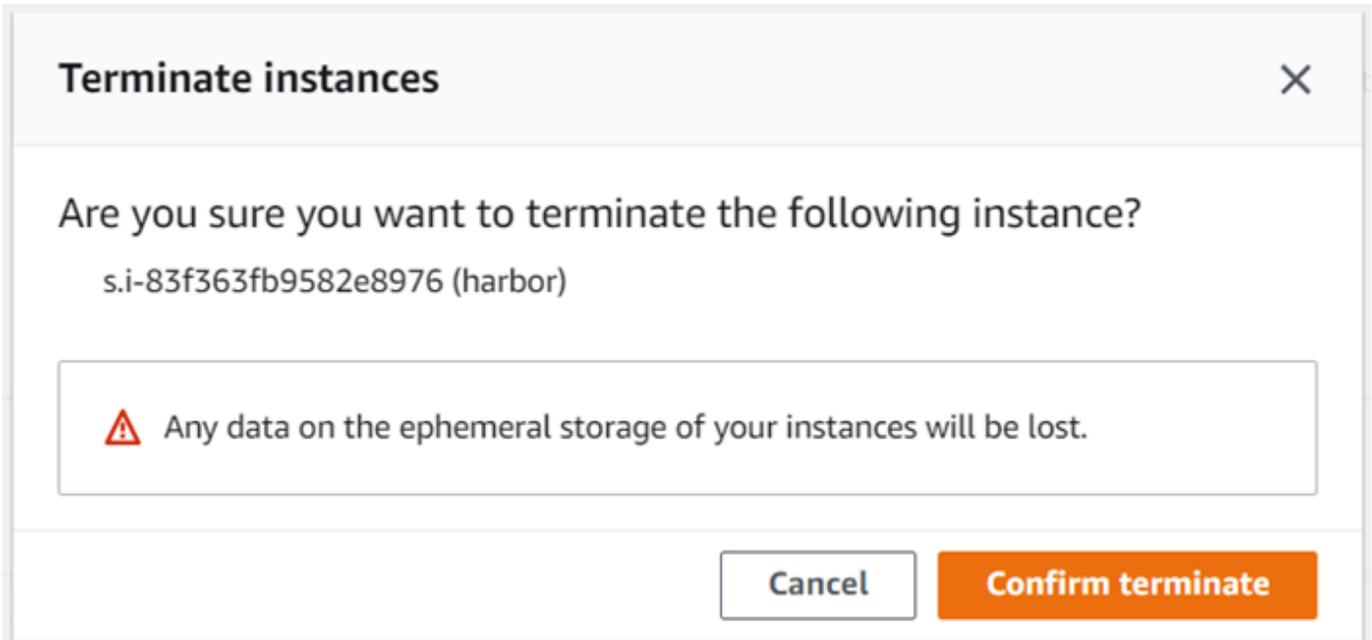
Amazon EC2 互換インスタンスを終了すると、再起動できなくなります。

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

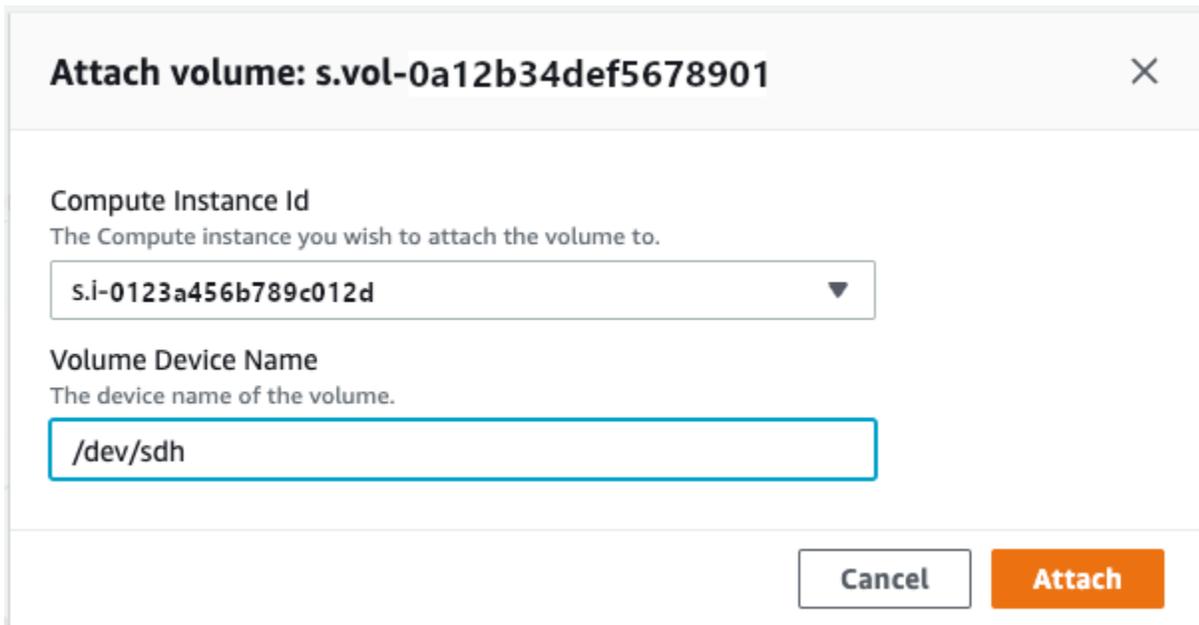
Volume Type  
Select a performance type for your volume.  
Capacity-optimized HDD volume (sbg1)

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。

## 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。

起動すると、NFS インターフェイスは 1 GB のメモリと 1 つの 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、[AWS Snowcone の仕組み](#)」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。

#### Note

のデバイスの詳細ページで AWS OpsHub、Snowcone デバイスの使用可能なストレージ領域は、NFS インターフェイスが開始されるまで正確ではありません。

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 インターフェイスを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 インターフェイスの停止

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。

## DataSync を使用してファイルに転送する AWS

AWS OpsHub を使用して、Snowcone デバイスで AWS DataSync エージェントを作成できます。これを使用して、デバイスと AWS クラウドの Amazon S3、Amazon Elastic File System (Amazon EFS)、FSx for Windows File Server 間でファイルを転送できます。

AWS DataSync は、AWS ストレージサービスとの間で大量のデータのコピーを簡素化、自動化、高速化するために設計されたオンラインデータ転送サービスです。は、インターネットまたは 経由でデータ DataSync をコピーします AWS Direct Connect。フルマネージドサービスとして、はアプリケーションの変更、スクリプトの開発、インフラストラクチャの管理の必要性の大部分 DataSync を排除します。

DataSync は、ネットワークファイルシステム (NFS) と Amazon EFS、Amazon S3、または Amazon FSx for Windows File Server 間のデータ転送をサポートします。

でサポートされている送信元と送信先のロケーションの組み合わせについては AWS DataSync、[「ユーザーガイド」の「ロケーション」の使用AWS DataSync](#)」を参照してください。

Snowcone には、オンプレミスストレージシステムからのデータを読み書きするために使用される仮想マシン (VM) である DataSync エージェントが付属しています。を使用するには DataSync、まずエージェントを起動し、DataSync コンソールに移動してアクティブ化します。の詳細については DataSync、[「の開始方法 AWS DataSync」](#)を参照してください。

DataSync エージェントを開始するには

1. DataSync エージェントを起動する前に、Snowcone デバイスで NFS を有効にします。「[NFS インターフェイスの自動設定](#)」および「[NFS の開始とアクセスの制限](#)」を参照してください。
2. AWS OpsHub ダッシュボードで、クラウドと同期セクションで開始 を選択し、DataSync エージェント開始ウィザードを開きます。エージェント開始 DataSync フォームには、デバイス IP アドレス と物理ネットワークインターフェイスフィールドが入力されます。

3. [Create IP address (VNI)] を選択して仮想 IP アドレスを作成するか、[Use existing IP address] を選択します。
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。  
  
[Use existing IP address] を選択した場合、[Virtual network interface] リストボックスから仮想インターフェイスを選択します。
5. [Start agent] を選択します。DataSync リソースページにリダイレクトされます。エージェントの IP アドレスが表示されるまで最大 5 分かかることがあります。
6. コピーアイコンを使用して、エージェントの IP アドレスファイルからエージェントの IP アドレス値をコピーし、DataSync コンソールを開く を選択します。

これにより、DataSync コンソールが開き、エージェントをアクティブ化してファイルを転送します。残りのセットアップは AWS DataSync コンソールで行われます。

## を使用したファイル転送 DataSync

AWS Snowcone はエージェントを既に作成しているため、エージェントをアクティブ化し、送信元と送信先の場所を設定し、タスクを作成して、タスクを開始するだけで済みます。

DataSync エージェントをアクティブ化して DataSync サービスを使用するには

1. <https://console.aws.amazon.com/datasync/> で AWS DataSync コンソールを開きます。
2. [Create agent page] (エージェントの作成) ページの [Activation] (アクティベーション) セクションで、コピーした IP アドレスを [Agent address] (エージェントアドレス) ボックスに貼り付け、[Get key] (キーの取得) を選択します。ブラウザは、IP アドレスに接続され、エージェントから一意のアクティベーションキーを取得します。
3. エージェントがアクティブ化されたら、Snowcone デバイスで実行されている NFS をのソースロケーションとして設定します DataSync。手順については、AWS DataSync ユーザーガイドの[送信元の場所設定](#)を参照してください。

### Note

Snowcone デバイスで実行されている DataSync エージェントは、ネットワーク上の到達可能な場所との間でファイルを転送できます。

4. [Configure a destination] (送信先の設定) ページで、ファイルの転送先を選択して設定します。手順については、AWS DataSync ユーザーガイドの[送信先の場所設定](#)を参照してください。
5. タスク設定を構成します。手順については、AWS DataSync ユーザーガイドの[タスク設定の構成](#)を参照してください。
6. 設定を確認してタスクを作成します。手順については、AWS DataSync ユーザーガイドの[設定の確認とタスクの作成](#)を参照してください。
7. タスクを開始し、ファイルが転送されるのを待ちます。手順については、AWS DataSync ユーザーガイドの[タスクの開始](#)を参照してください。

## デバイスの管理

を使用して Snow Family デバイスを AWS OpsHub 管理できます。デバイスの詳細ページでは、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](#)
- [デバイスと AWS OpsHub アプリケーションの更新の取得](#)
- [プロファイルの管理](#)

## 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

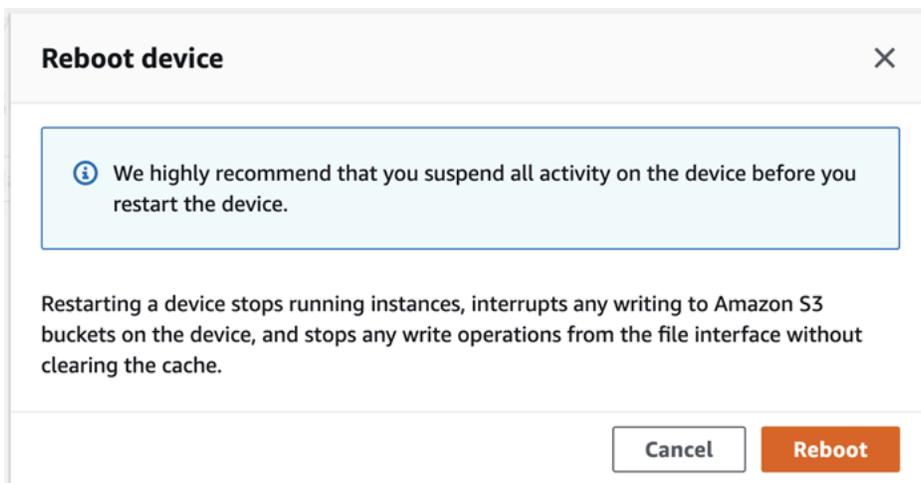
デバイスを再起動する前に、デバイスでのすべてのアクティビティを停止することが強く推奨されます。デバイスを再起動すると、実行中のインスタンスが停止し、デバイス上の Amazon S3 バケットへの書き込みが中断されます。

## 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



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

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**

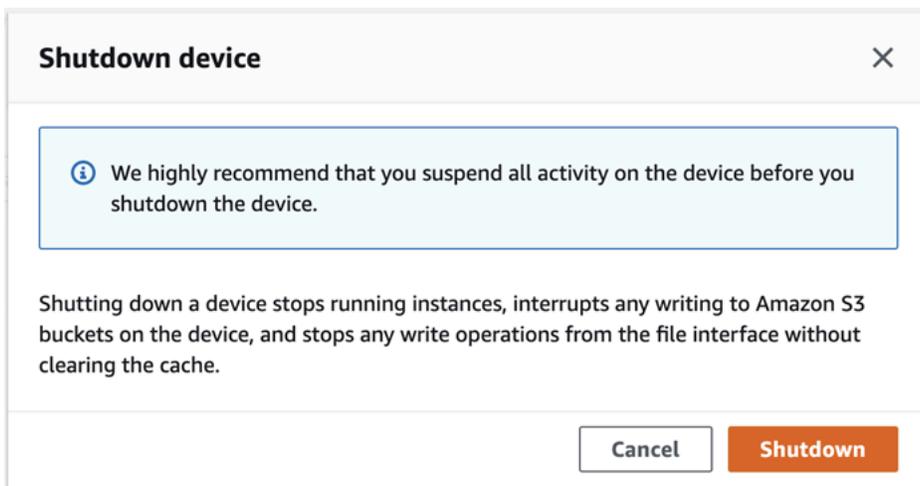
デバイスをシャットダウンする前に、デバイスですべてのアクティビティを停止することが強く推奨されます。デバイスをシャットダウンすると、実行中のインスタンスが停止し、デバイス上の Amazon S3 バケットへの書き込みが中断されます。

## 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。

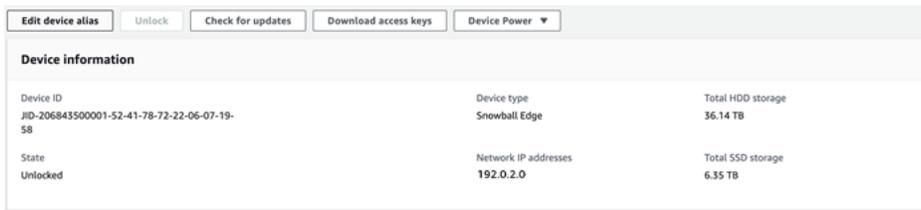


## デバイスのエイリアスの編集

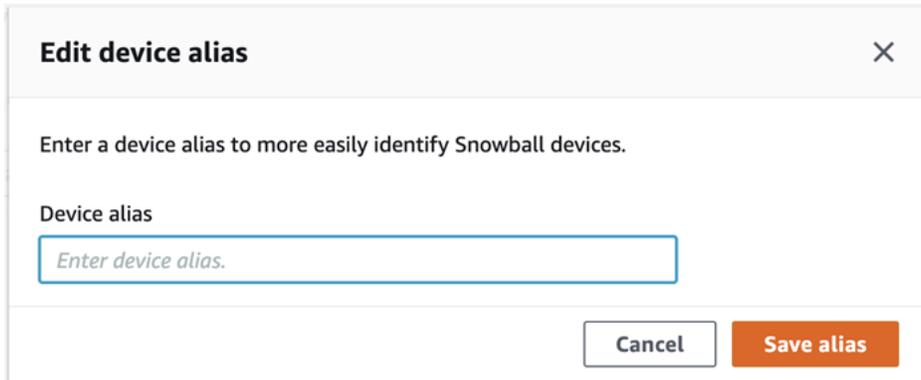
を使用してデバイスエイリアスを編集するには、次の手順に従います AWS OpsHub。

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。



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



## 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。

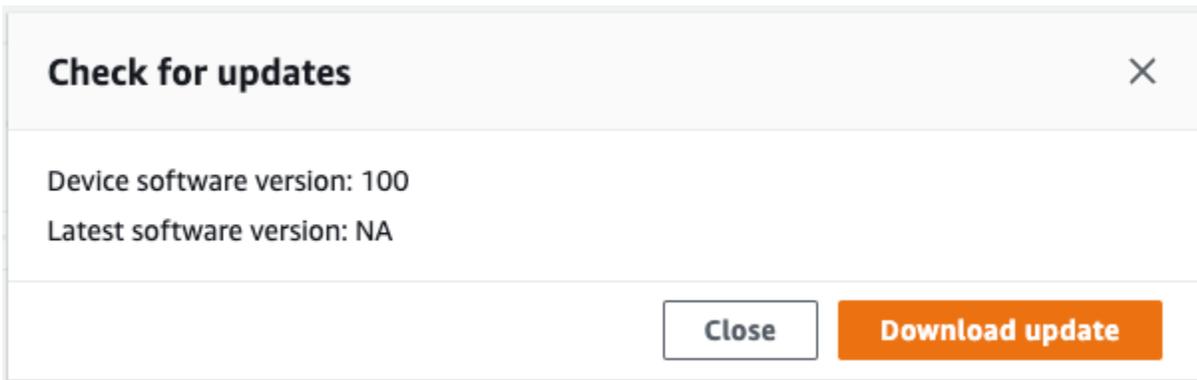
### デバイスの更新

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。



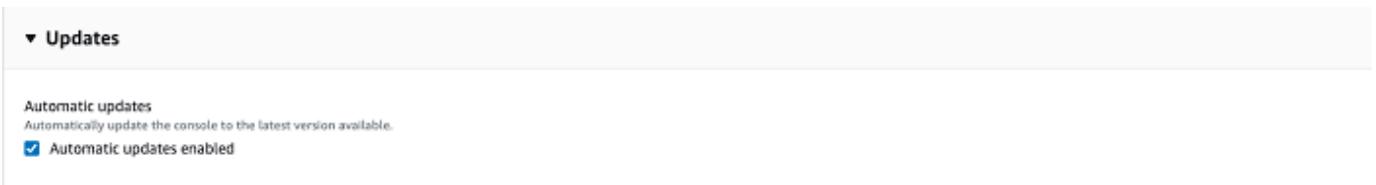
3. 更新が利用可能な場合は、[更新をダウンロード] を選択します。それ以外の場合は、[Close (閉じる)] を選択します。

## 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。

## 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.

# AWS Snowball Edge クライアントの使用

Snowball Edge クライアントは、ローカルサーバーで実行するスタンドアロンのターミナルアプリケーションで、AWS Snowcone デバイスのロックを解除し、認証情報、ログ、ステータス情報を取得します。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。

AWS Snowcone デバイスにデータを読み書きするときは、NFS インターフェイスを使用します。AWS OpsHub for Snow Family アプリケーションを使用して、Snowcone デバイスを含む Snow Family デバイスを管理することもできます。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。

## 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 リソース](#)からダウンロードしてインストールできます。そのページで、オペレーティングシステム用のインストールパッケージを検索し、指示に従って 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 クライアントのコマンド

以下で、使用例と出力サンプルを含む、Snowball Edge クライアントのコマンドを説明します。

**Note**

AWS Snowcone デバイスは同じ Snowball Edge CLI コマンドを使用しますが、クラスタリングに適用されるコマンドはサポートしていません。

## 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](#)
- [AWS Snowcone デバイスのロック解除](#)
- [Snowcone の更新](#)
- [認証情報の取得](#)
- [Snowcone Device でのサービスの開始](#)
- [Snowcone Device でのサービスの停止](#)
- [データ転送の証明書の取得](#)
- [AWS Snowcone 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](#)
- [AWS DataSync AMI の起動](#)
- [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](#)
- [Amazon S3 バケットのエクスポートパスの取得](#)
- [ローカル AWS オペレーターのデバッグを有効にする](#)
- [ローカル AWS オペレーターのデバッグを無効にする](#)
- [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](#)
- [機能ステータスのチェック](#)
- [機能ステータスの変更](#)
- [タイムサーバーの設定](#)
- [タイムソースのチェック](#)

## 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、「[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-.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 Snowcone デバイス LCD ディスプレイで AWS Snowcone デバイスの IP アドレスを見つけることができます。AWS Snowcone デバイスをお使いのネットワークに初めて接続するときに DHCP サーバーを使用できる場合、自動的に DHCP IP アドレスを取得します。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、「[での AWS サービスの使用 AWS Snowcone](#)」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、Snowcone デバイス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 出力例

```
Configuration will be stored at home directory\.aws\snowball\config\snowball-.config  
Snowcone 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、複数のプロフィー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、「AWS Command Line Interface 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 Snowcone の検証アプリケーションで使用するデバイス固有の QR コードを生成することができます。このアプリは、Apple App Store または Google Play ストアからダウンロードできます。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 出力例

```
QR code is saved to ~/downloads/snowball-qr-code.png
```

## AWS Snowcone デバイスのロック解除

スタンドアロン AWS Snowcone 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。これらのコマンドは、AWS Snowcone デバイスへのアクセスを認証します。

これらのロック解除コマンドのいずれかを実行すると、マニフェストファイルへのパス、29 文字のロック解除コード、スタンドアロンデバイスの IP アドレスを手動で入力できます。このプロセスは面倒なため、代わりに Snowball Edge クライアントを設定することをお勧めします。Snowball Edge クライアントをすでに設定している場合は、マニフェストへのパス、ロック解除コード、IP アドレスを入力することなく、コマンド自体を入力するだけで済みます。

### 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 Snowcone 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。

## 使用法 (Snowball Edge クライアント設定済み)

```
snowballEdge unlock-device
```

### Example

#### Example 出力のロック解除

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

## Snowcone の更新

次のコマンドを使用して、Snowcone デバイスの更新をダウンロードしてインストールします。これらのコマンドを使用する手順については、「[Snowcone の更新](#)」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 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 出力例

```
Latest version: 102
Installed version: 101
```

`snowballEdge describe-device-software` – デバイスの現在のソフトウェアバージョンを返します。また、更新のダウンロード中は、ダウンロード状態も表示されます。更新が進行中の場合は、更新のバージョンマニフェスト、およびインストール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 出力例

```
Installed version: 101
Installing version: 102
Install State: Downloading
```

`snowballEdge download-updates` – Snowcone デバイスの最新の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 出力例

```
Download started. Run describe-device-software API for additional information.
```

`snowballEdge install-updates` – 既にダウンロードされた Snowcone デバイス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 出力例

```
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 出力例

```
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 出力例

```
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 出力例

```
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 コマンド snowballEdge get-secret-access-key とコマンドを使用して、ローカル認証情報を取得できます。これらを使用して、AWS CLI または AWS SDK を使用するときリクエストを認証します。これらの認証情報は、Snowcone の個々のジョブにのみ関連付けられており、デバイスでのみ使用できます。デバイスには、AWS クラウドの AWS Identity and Access Management (IAM) 許可はありません。

詳細については、「[AWS 認証情報](#)」と「[Amazon EC2 エンドポイントの使用](#)」を参照してください。

### Note

Snowball AWS CLI 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。CLI の認証情報を設定する方法については、AWS Command Line Interface ユーザーガイドの[クイック設定](#)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 出力例

```
{
  "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 出力例

```
[snowballEdge]
aws_access_key_id = AKIAIOSFODNN7EXAMPLE
aws_secret_access_key = wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY
```

## Snowcone Device でのサービスの開始

Snowcone は、コンピューティングインスタンス、NFS ファイルインターフェイス、Amazon EC2、など、複数のサービスをサポートしています AWS DataSync。これらのサービスは、snowballEdge start-service コマンドで開始できます。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 出力例

```
Starting the AWS service on your Snowball Edge . You can determine the status of the
AWS service using the describe-service command.
```

## Snowcone Device でのサービスの停止

Snowcone デバイス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 EC2 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。

使用法 (Snowball Edge クライアント設定済み)

```
snowballEdge stop-service --service-id service_id
```

Example 出力例

```
Stopping the AWS service on your Snowball . You can determine the status of the AWS
service using the describe-service command.
```

## データ転送の証明書の取得

Snowcone デバイスにデータを転送するには、NFS インターフェイスまたは を使用します AWS DataSync。Snowcone デバイスのロックを別の IP アドレスで解除すると、新しい証明書が生成され、古い証明書のエンドポイントでの使用が無効になります。 `get-certificate` コマンドを使用して、Snowcone デバイスから再度、新しい更新済みの証明書を取得できます。

これらの証明書は以下のコマンドで一覧表示し、Snowcone デバイスからダウンロードできます。

- `list-certificates` – 使用できる証明書の Amazon リソースネーム (ARN) をリストします。

使用法 (Snowball Edge クライアント設定済み)

```
snowballEdge list-certificates
```

## Example 出力例

```
{
  "Certificates" : [ {
    "CertificateArn" : "arn:aws:snowball-
device:::certificate/78EXAMPLE516EXAMPLEf538EXAMPLEa7",
    "SubjectAlternativeNames" : [ "192.0.2.0" ]
  } ]
}
```

- `get-certificate` – 指定された ARN に基づいて、特定の証明書を取得します。

### 使用法 (Snowball Edge クライアント設定済み)

```
snowballEdge get-certificate --certificate-arn arn:aws:snowball-
device:::certificate/78EXAMPLE516EXAMPLEf538EXAMPLEa7
```

## Example 出力例

```
-----BEGIN CERTIFICATE-----
Certificate
-----END CERTIFICATE-----
```

## AWS Snowcone ログ

オンプレミスのデータセンターと Snowcone デバイス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 出力例

```
{
  "Logs" : [ {
    "LogArn" : "arn:aws:snowball-device::log/s3-storage-JIEXAMPLE2f-1234-4953-a7c4-dfEXAMPLE709",
    "LogType" : "SUPPORT",
    "ServiceId" : "datasync",
    "EstimatedSizeBytes" : 53132614
  }, {
    "LogArn" : "arn:aws:snowball-device::log/fileinterface-JIDEXAMPLEf-1234-4953-a7c4-dfEXAMPLE709",
    "LogType" : "CUSTOMER",
    "ServiceId" : "nfs",
    "EstimatedSizeBytes" : 4446
  }
]
```

- `get-log` – Snowcone デバイス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 出力例

```
Logs are being saved to download/path/snowball--logs-1515EXAMPLE88.bin
```

- `get-support-logs` – Snowcone デバイスから 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)

```
snowballEdge get-support-logs
```

## Example 出力例

```
Logs are being saved to download/path/snowball--logs-1515716135711.bin
```

### Important

CUSTOMER ログには、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。この機密情報を保護するために、これらのログが不要になった場合は、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、Snowcone 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 出力例

```
{
  "DeviceId" : "JIDbEXAMPLE-7eed-1234-ABCD-7EXAMPLE123c",
  "UnlockStatus" : {
    "State" : "UNLOCKED"
  },
  "ActiveNetworkInterface" : {
    "IpAddress" : "192.168.1.2"
  },
  "PhysicalNetworkInterfaces" : [ {
    "PhysicalNetworkInterfaceId" : "s.ni-8bEXAMPLE5EXAMPLE",
    "PhysicalConnectorType" : "RJ45",
    "IpAddressAssignment" : "DHCP",
    "IpAddress" : "192.168.1.13",
    "Netmask" : "255.255.255.0",
    "DefaultGateway" : "192.168.1.1",
```

```
    "MacAddress" : "EX:AM:PL:E0:12:34"
  }, {
    "PhysicalNetworkInterfaceId" : "s.ni-84EXAMPLE3EXAMPLE",
    "PhysicalConnectorType" : "RJ45_2",
    "IpAddressAssignment" : "STATIC",
    "IpAddress" : "0.0.0.0",
    "Netmask" : "0.0.0.0",
    "DefaultGateway" : "192.168.1.1",
    "MacAddress" : "EX:AM:PL:E0:12:34"
  }, {
    "PhysicalNetworkInterfaceId" : "s.ni-87EXAMPLE5EXAMPLE",
    "PhysicalConnectorType" : "WIFI",
    "IpAddressAssignment" : "STATIC",
    "IpAddress" : "0.0.0.0",
    "Netmask" : "0.0.0.0",
    "DefaultGateway" : "192.168.1.1",
    "MacAddress" : "EX:AM:PL:E0:12:34"
  } ],
  "DeviceCapacities" : [ {
    "Name" : "HDD Storage",
    "Unit" : "Byte",
    "Total" : 157242114048,
    "Used" : 81604378624,
    "Available" : 75637735424
  }, {
    "Name" : "SSD Storage",
    "Unit" : "Byte",
    "Total" : 0,
    "Used" : 0,
    "Available" : 0
  }, {
    "Name" : "vCPU",
    "Unit" : "Number",
    "Total" : 3,
    "Used" : 3,
    "Available" : 0
  }, {
    "Name" : "Memory",
    "Unit" : "Byte",
    "Total" : 5368709120,
    "Used" : 5368709120,
    "Available" : 0
  }, {
    "Name" : "GPU",
```

```
"Unit" : "Number",
"Total" : 0,
"Used" : 0,
"Available" : 0
} ],
"DeviceType" : "SNC1_HDD"
}
```

## サービスステータスの取得

describe-service コマンドを使用して、Snowcone 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 出力例

```
{
  "ServiceIds" : [ "nfs", "datasync", "ec2" ]
}
```

- 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。
- 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 出力例

```
{
  "ServiceId" : "ec2",
  "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"
    }
  } ]
}
```

## AWS DataSync AMI の起動

Snowcone で AWS DataSync AMI を起動します。

使用法 (Snowball Edge クライアント設定済み)

AWS DataSync は、snc1.medium インスタンスタイプで起動する必要があります。別のインスタンスタイプ DataSync で起動すると、オペレーションが不安定になり、データが失われる可能性があります。describe-images コマンドを使用して、インスタンスを起動するイメージを検索します。出力は次のようになります。

```
{
  "ImageId": "s.ami-0c046f119de4f752f",
  "Public": false,
```

```

"State": "AVAILABLE",
"BlockDeviceMappings": [
  {
    "DeviceName": "/dev/sda",
    "Ebs": {
      "DeleteOnTermination": true,
      "Iops": 0,
      "SnapshotId": "s.snap-0d7558ce444ab09bf",
      "VolumeSize": 20,
      "VolumeType": "sbp1"
    }
  }
],
>Description": "AWS DataSync AMI for online data transfer",
>EnaSupport": false,
>Name": "scn-datasync-ami",
>RootDeviceName": "/dev/sda"
}

```

```
aws ec2 describe-instances --endpoint http://${snowcone_ip}:8008
```

## Example 出力例

```

{
  "Reservations": [
    {
      "Instances": [
        {
          "AmiLaunchIndex": 0,
          "ImageId": "s.image id",
          "InstanceId": "s.instance id",
          "InstanceType": "snc1.medium",
          "LaunchTime": "2020-03-06T18:58:36.609Z",
          "PrivateIpAddress": "ip address",
          "State": {
            "Code": 16,
            "Name": "running"
          },
          "BlockDeviceMappings": [
            {
              "DeviceName": "/dev/sda",
              "Ebs": {
                "AttachTime": "2020-03-06T19:14:21.336Z",

```

```

        "DeleteOnTermination": true,
        "Status": "attached",
        "VolumeId": "s.volume id"
    }
}
],
"EbsOptimized": false,
"EnaSupport": false,
"RootDeviceName": "/dev/sda",
"SecurityGroups": [
    {
        "GroupName": "default",
        "GroupId": "s.security group id"
    }
],
"SourceDestCheck": false,
"CpuOptions": {
    "CoreCount": 2,
    "ThreadsPerCore": 1
}
},
"ReservationId": "s.r-80c8ee6b041b29eb4"
},
]
}

```

インスタンスを実行します。

```
aws ec2 run-instances --image-id s.ami id \--instance-type snc1.medium --endpoint
http://${snowcone_ip}:8008
```

Example 出力例

```

{
  "Instances": [
    {
      "AmiLaunchIndex": 0,
      "ImageId": "s.ami-0623310b494365cc5",
      "InstanceId": "s.i-80c8ee6b041b29eb4",
      "InstanceType": "snc1.medium",
      "State": {
        "Code": 0,

```

```
        "Name": "pending"
    },
    "EbsOptimized": false,
    "EnaSupport": false,
    "RootDeviceName": "/dev/sda",
    "SecurityGroups": [
        {
            "GroupName": "default",
            "GroupId": "s.sg-80c8ee6b041b29eb4"
        }
    ],
    "SourceDestCheck": false,
    "CpuOptions": {
        "CoreCount": 2,
        "ThreadsPerCore": 1
    }
}
],
"ReservationId": "s.r-80c8ee6b041b29eb4"
}
```

## 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。

### Note

Snowcone デバイスで使用可能なストレージ容量は、NFS サービスが開始されるまでは正確ではありません。

デバイスによって公開される NFS 共有ヘマウントする IP アドレス範囲に、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
```

## Amazon S3 バケットのエクスポートパスの取得

Amazon S3 バケットのエクスポートパスを取得するための特定の Snowcone コマンドはありません。エクスポートパスの形式は次のようになります。

```
/buckets/bucket-name.
```

## ローカル AWS オペレーターのデバッグを有効にする

- `enable-local-aws-operator-debugging` — SSH ポート 22 を開いて、ローカル AWS オペレーターのデバッグのためにデバイスを有効にします。

使用法 (Snowball Edge クライアント設定済み)

```
snowballEdge enable-local-aws-operator-debugging
```

## ローカル AWS オペレーターのデバッグを無効にする

- `disable-local-aws-operator-debugging` — SSH ポート 22 を閉じて、デバイスのローカル AWS 演算子のデバッグを無効にします。デフォルトでは、SSH ポート 22 は閉じています。Snowcone デバイスがオフになっているか、電源が入れ替わると、ローカル AWS オペレーターのデバッグは無効になります。

使用法 (Snowball Edge クライアント設定済み)

```
snowballEdge disable-local-aws-operator-debugging
```

## 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` - [ダイレクトネットワークインターフェイス \(DNI\) を作成します。](#) デバイス上の Amazon EC2 コンピューティングインスタンスで使用する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。`describe-direct-network-interfaces` コマンドを使用して、デバイスで使用可能なダイレクトネットワークインターフェイスを確認できます。

## 使用法 (Snowball Edge クライアント設定済み)

```
create-direct-network-interface [--endpoint endpoint] [--instance-id instanceId] [--mac macAddress]
                                [--manifest-file manifestFile] [--physical-network-
interface-id physicalNetworkInterfaceId]
                                [--profile profile] [--unlock-code unlockCode] [--
vlan vlanId]
```

## 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` - デバイス上のダイレクトネットワークインターフェイスを取得します。ダイレクトネットワークインターフェイスを使用して、デバイス上の Amazon EC2 コンピューティングインスタンスおよびサービスのネットワークを設定できます。`create-direct-network-interface` コマンドを使用して、新しいダイレクトネットワークインターフェイスを作成できます。

## 使用法 (Snowball Edge クライアント設定済み)

```
describe-direct-network-interfaces [--endpoint endpoint] [--manifest-file manifestFile]
  [--profile profile] [--unlock-code unlockCode]
```

## 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` - ダイレクトネットワークインターフェイスを更新します。このコマンドを使用して、デバイス上の Amazon EC2 コンピューティングインスタンスで使用するダイレクトネットワークインターフェイスを更新します。`describe-direct-network-interfaces` コマンドを使用して、デバイスで使用可能なダイレクトネットワークインターフェイスを確認できます。Amazon EC2 インスタンスにアタッチされているネットワークインターフェイスを変更する場合、そのインターフェイスは最初にデタッチされます。

## 使用法 (Snowball Edge クライアント設定済み)

```
update-direct-network-interface [--direct-network-interface-arn directNetworkInterfaceArn] [--endpoint endpoint]
  [--mac macAddress]
  [--manifest-file manifestFile] [--profile profile] [--
unlock-code unlockCode]
```

```
[--vlan vlanId] [--attach-instance-id instanceId | --  
detach]
```

## 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` – 使用されなくなったダイレクトネットワークインターフェイスを削除します。Amazon EC2 コンピューティングインスタンスに関連付けられているダイレクトネットワークインターフェイスを削除するには、まずインスタンスからダイレクトネットワークインターフェイスの関連付けを解除する必要があります。

使用法 (Snowball Edge クライアント設定済み)

```
delete-direct-network-interface [--direct-network-interface-  
arn directNetworkInterfaceArn] [--endpoint endpoint]  
                                [--manifest-file manifestFile] [--profile profile] [--  
unlock-code unlockCode]
```

## 機能ステータスのチェック

Snowcone デバイスとローカル AWS サービスをリモートで管理できる など AWS Snow Device Management、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  
}
```

## 機能ステータスの変更

AWS Snowcone デバイスで使用可能な機能のステータスを変更するには、`set-features` コマンドを使用します。Snowcone デバイスとローカル AWS サービスをリモートで管理 AWS Snow Device Management できる を有効または無効にするには、`--remote-management-state` パラメータを使用します。このコマンドを実行する前に、デバイスのロックを解除する必要があります。

Snow Device Management は次の状態に設定できます。

- `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、デバイスは電源がオン AWS リージョン になっているときにへの接続を試みます。

### Note

`NOT_INSTALLED` の状態は、Snow Device Management をサポートしないデバイス、または起動前に既にフィールドに存在していたデバイスを識別するためだけに存在します。すでにデプロイされているデバイスにこの機能をインストールすること、またはデバイスからアンインストールすることはできません。Snow Device Management を使用するには、この機能がプリインストールされた新しいデバイスを注文する必要があります。

## 使用法 (Snowball Edge クライアント設定済み)

```
snowballEdge set-features  
--remote-management-state INSTALLED_AUTOSTART  
--manifest-file ./JID2bf11d5a-fict-414a-b5b1-3bf7e6a6e83d_manifest.bin  
--unlock-code 73bb0-f8ke1-69a4a-f4288-4f88d  
--endpoint https://10.0.0.25
```

## 出力例

```
{
  "RemoteManagementState" : "INSTALLED_AUTOSTART"
}
```

## 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。Edge AWS Snowcone 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。

## 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。

# 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に、デバイスを受け取ったときに Snow Device Management がどの状態になるかを選択できます。インストール済みでも、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ません。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 を使用できます。、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」[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、[AWS Snowcone デバイスの仕様への接続](#)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、[する](#)を参照してください。<https://docs.aws.amazon.com/snowball/latest/snowcone-guide/connect-unlock-local.html>

```
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-ficthmqoc2pht1ef \
--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
```

## での AWS サービスの使用 AWS Snowcone

以下に、AWS Snowcone デバイスの概要を示します。AWS Snowcone 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できます。

AWS Snowcone デバイスのロック解除については、「」を参照してください [AWS Snowball Edge クライアントの使用](#)。

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。

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

- [Amazon EC2 互換コンピューティングインスタンスの使用](#)
- [AWS DataSync 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 Snowcone デバイスで AWS サービスを使用するために必要なポート](#)

## Amazon EC2 互換コンピューティングインスタンスの使用

このトピックでは、AWS Snowcone デバイスでの Amazon Elastic Compute Cloud (Amazon EC2) コンピューティングインスタンスの使用に関する概要を説明します。このトピックには、概念情報、手順、例などが含まれています。

**Note**

これらの機能は、アジアパシフィック (ムンバイ) ではサポートされていません AWS リージョン。

## 概要

Snowcone 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は、サポートされている EC2 互換インスタンスタイプを使用することで実行できます。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、Snowcone ジョブを作成します。サポート対象のインスタンスタイプの詳細については、「[Snowcone での Amazon EC2 の使用](#)」を参照してください。

ジョブタイプがローカルコンピューティングである場合は、合計 8 TiB のローカルの EBS ボリュームを作成し、それを Amazon EC2 互換インスタンスにアタッチします。これにより、ローカルの Amazon EC2 互換インスタンスは、ルートボリューム単体よりも多くのローカル容量にアクセスできます。これはローカルストレージのみであるため、デバイスが返されると、EBS ボリュームに書き込まれたデータは失われます。これは Amazon S3 にインポートできない AWS ためです。

**Note**

NFS サーバーはコンピューティングジョブには使用できません。との間でデータをインポートまたはエクスポートする必要がある場合は AWS クラウド、注文時にローカルコンピューティングジョブタイプを選択しないでください。

Snowcone 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 から実行できます。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ー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。Amazon EC2 と互換性のあるエンドポイントを使用して、Snowcone でコンピューティングインスタンスを管理できます。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。ツールを使用して AWS OpsHub for Snow Family、AMIs、コンピューティングインスタン

ス、および AWS サービスを管理します。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。

デバイスの使用が終了したら、に戻します AWS。デバイスがインポートジョブで使用された場合、NFS インターフェイスを介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、デバイス内のデータは AWS に返却された時点で完全消去されます。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。

#### Important

- Snowcone 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。
- Snowcone 上で実行されるコンピューティングインスタンスのデータが、AWS にインポートされることはありません。

## Snowcone 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 Snowcone の料金](#)」を参照してください。

## 前提条件

ジョブを作成するには、以下の点を考慮する必要があります。

- ジョブに AMIs を追加する前に、に AMI があり AWS アカウント、サポートされているイメージタイプで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](#)

#### 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 \(HVM\)](#)、[Ubuntu 20.04 LTS - Focal](#)、または [Ubuntu 22.04 LTS - Jammy](#)

#### 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 Snow Family 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。

ジョブ作成リクエストに AMI を追加する前に、1 つ以上のサポートされている AMI が AWS アカウントにあることを確認します。

- AMI はいずれも 1 つのボリュームを使用した Amazon Elastic Block Store (Amazon EBS) ベースである必要があります。
- Snowcone で実行されるコンピューティングインスタンスに接続する予定の場合は、Secure Shell (SSH) を使用する必要があります。これを行うには、最初にキーペアを追加します。

## コンピューティングインスタンスを含むジョブを作成する

このセクションでは、最初の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。

- から AMI を使用する場合は AWS Marketplace、サポートされている製品コードと使用オペレーションコードがあることを確認してください。詳細については、「[AWS Marketplace AMIs](#)」を参照してください。
- 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。
- Snowcone でインスタンスを起動した後に 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。
- AWS Snowcone デバイスでの暗号化 AMIs または暗号化された Amazon EBS ボリュームの使用はサポートされていません。

## 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 の製品コードと使用コードが 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。お客様は場所に関係なく、使用できるリージョンをどれでも選択できます。
3. ナビゲーションペインで [AMIs] 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、 が提供するすべての Linux AMIs 一覧表示するには AWS、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、AMI のリストに表示される範囲を指定します。

(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。検索バーをもう一度クリックし、[Platform] (プラットフォーム)、 = 演算子の順に選択した上で、表示されたリストからオペレーティングシステムを選択します。

(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。[Search] (検索) バーをもう一度選択し、[Platform] (プラットフォーム) を選択します。次に、表示されたリストからオペレーティングシステムを選択します。

### 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。

### Note

製品コード avyfzznywe""kgI5qv5f57ska は、すべてのリージョンでサポートされています。

## 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
Amazon Linux 2	avyfzznywe""kgl5qv5f57ska
Ubuntu 20.04 LTS	a8jyynf4hjutohctm41o2z18m
Ubuntu 22.04 LTS	47xbqns9xujfkkjt189a13aqe

- すべてのリージョンで、使用状況オペレーションコードがであることを確認しますRunInstances。

## 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": "avyfzznywektkgl5qv5f57ska",
        "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。

## デバイスで起動しているコンピューティングインスタンスに接続するために SSH を使用するように AMI を設定する

Secure Shell (SSH) を使用して Snowcone 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。

SSH キーを AMI に追加するには

1. [Snow Family 用の Amazon Linux 2](#)、[CentOS 7 \(x86\\_64\) - アップデート HVM](#)、または Ubuntu 16.04 LTS - Xenial (HVM) イメージ AWS クラウドに基づいて、で新しいインスタンスを起動します。 <https://aws.amazon.com/marketplace/pp/B01JBL2M00>

インスタンスを起動する際に、インスタンスに割り当てるストレージサイズが、Snowcone での使用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。Snowcone 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、サポートされているサイズのリストについては、[AWS Snowcone クォータ](#) を参照してください。

2. Snowcone で実行するアプリケーションをインストールして設定し、正常に動作することをテストします。
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。Snowcone 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。[AWS Snow ファミリーマネジメントコンソール](#) を使用するには、[開始方法](#) の手順に従います。

## 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、ターミナルを開き、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 snowballEdge 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 SNOWCONE
```

デバイスが届き、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。

Snowcone デバイスは EC2 互換インスタンスごとに 8 つの DNI をサポートし、デバイスあたり最大 8 つの DNI をサポートします。

## 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、[Snowcone 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。

- 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。

- 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。
- 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
```

## 追加コマンド

デバイスで使用できるすべてのコマンドについては AWS Snowcone、[「Snowball Edge Client の使用」](#)を参照してください。

## SSH を使用して Snowcone のコンピューティングインスタンスに接続する

SSH を使用して Snowcone デバイス上のコンピューティングインスタンスに接続するには、ジョブを作成する前に、SSH キーを AMI に付与する必要があります。この手順の詳細については、[「デバイスで起動しているコンピューティングインスタンスに接続するために SSH を使用するように AMI を設定する」](#)を参照してください。この手順を行わない場合、SSH を使用してインスタンスに接続することはできません。

SSH を使用して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。
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

## コンピューティングインスタンスの Snowcone 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。これにより、Snowcone デバイスで一部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方法の詳細については、[AWS Snowball Edge クライアントの使用](#) を参照してください。

ここでは、コンピューティングインスタンスに固有の Snowball Edge コマンドについて、その使用例を含め、説明します。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 Snowcone、「」を参照してください [Snowcone EC2-compatible AWS CLI コマンド](#)。

#### Note

クラスターに関連するコマンドはサポートされていないので、エラーが表示されます。

## 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 Snowcone 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。起動設定を作成するには、`snowballEdge create-autostart-configuration` コマンドを使用します。その使用 방법은以下のとおりです。

### 使用方法

```
snowballEdge create-autostart-configuration --physical-connector-type  
[SFP_PLUS or RJ45] --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

Snowcone 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 방법은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]]  
[--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

使用されなくなった起動設定を削除するには、`snowballEdge delete-autostart-configuration` 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の一覧表示

Snowcone で作成した起動設定を一覧表示するには、`describe-autostart-configurations` 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

Snowcone でコンピューティングインスタンスを実行するか、Snowcone で NFS インターフェイスを起動するには、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 Snowcone 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 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"
  }
}
```

```
}
```

## 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"
  },{
    "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"
  }
]
```

]

## 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。

**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、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 組み合わされたコマンド**

```
[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。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   --:--:--
--:--:-- --:--:-- 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-idhostnameinstance-idinstance-typelocal-hostnamelocal-ipv4macnetwork/reservation-idsecurity-groups
```

### IMDSv1

```
[ec2-user ~]$ curl http://192.0.2.0/latest/meta-data/ami-idhostnameinstance-idinstance-typelocal-hostnamelocal-ipv4macnetwork/reservation-idsecurity-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 互換エンドポイントの使用

以下は、Amazon Elastic Compute Cloud (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。

### トピック

- [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](#)
- [Snowcone でサポートされていない Amazon EC2 の機能](#)
- [Snowcone EC2-compatible AWS CLI コマンド](#)
- [サポートされる Amazon EC2 API オペレーション](#)

### 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 Snowcone 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、ロックが解除されるたびに、Snowcone によって生成される証明書を使用して保持されます。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. Snowcone 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、Snowcone デバイスに 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、[データ転送の証明書の取得](#) 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.snowcone.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。

## Snowcone でサポートされていない Amazon EC2 の機能

Amazon EC2 互換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、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。ただし、すべての機能と API オペレーションが Snowcone デバイスでサポートされているわけではありません。

このガイドでサポート対象として明示的にリストされていない機能またはアクションは、サポートされていません。たとえば、以下の Amazon EC2 アクションは、Snowcone での使用がサポートされていません。

- [create-nat-gateway](#)
- [create-key-pair](#)

## Snowcone 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。

## Snowcone でサポートされる Amazon EC2 互換 AWS CLI コマンドの一覧

以下は、Snowcone 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](#) – Amazon EBS ボリュームを AWS Snowcone デバイス上の停止または実行中のインスタンスにアタッチし、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 インスタンスの ID。
  - -volume-id value – EBS ボリュームの ID。
- [authorize-security-group-egress](#) – Snowcone 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、「[Snow 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。AWS Snowcone 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"
  }
]
}

```

- `[--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 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 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](#) — Snowcone デバイス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](#) — AWS Snowcone 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 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 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。
- [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](#) – 利用可能な 1 つ以上のイメージ (AMI) について記述します。使用可能なイメージは、ジョブの作成時に Snowcone 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-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 IDs のリスト。
- `-- launch-template-names ( リスト) "string" "string"` – 起動テンプレートの名前前のリスト。
- `--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 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 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 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 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に 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 Snowcone 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、Snowcone デバイス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 Snowcone 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 IDs。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。
- `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。

## サポートされる Amazon EC2 API オペレーション

以下に、Snowcone デバイスで使用できる Amazon EC2 API オペレーションについて説明します (Amazon EC2 API リファレンス内の説明へのリンクあり)。Amazon EC2 API コールでは、署名バージョン 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。

- [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](#) – Snowcone 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。を呼び出すときは AuthorizeSecurityGroupIngress、GroupName または の値を指定する必要がありますGroupId。
- [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](#) – Snowcone 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
- [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 `group-id` の値を指定する必要があります。
- [RunInstances](#) –

 Note

インスタンスのサイズとタイプに応じて、Snowcone 上でコンピューティングインスタンスを起動するために最大 1 時間半程度かかります。

- [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 Snowcone 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。インスタンスは、終了すると、自動起動によって起動しますが、インスタンスを削除するか、インスタンスの自動起動設定を更新した場合、この自動起動によって新しいインスタンスが起動します。

起動テンプレートには、Snowcone に 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、Snowcone で EC2 互換インスタンスを起動するたびにそれらのパラメータを指定する必要はありません。

Snowcone で自動起動設定を使用する場合、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。Snowcone を設定した後、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されます。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。後続のすべてのインスタンス起動（計画的または計画外の再起動後）は、AWS Snowcone デバイスのロックが解除された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。

Snowcone で EC2 互換インスタンスを自動起動するには、以下のステップに従います。

1. AWS Snowcone 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インスタンスを含むジョブを作成する](#)」を参照してください。
2. Snowcone を入手したら、そのロックを解除します。
3. EC2 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。詳細については、「[Snowcone でサポートされる Amazon EC2 互換 AWS CLI コマンドの一覧](#)」を参照してください。

#### Note

Amazon EC2 エンドポイントは、デバイスエンドポイントです。

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. 再起動し、AWS Snowcone デバイス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、Snowcone 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されます。

実行中のインスタンスのステータスを表示するには、EC2 API コマンド `describe-autostart-configurations` を使用します。

#### Note

テンプレートの起動 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 Snowcone 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 および 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のブロックストレージの使用

Snowcone 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 エンドポイントを使用してサポートされます。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、「[Snowcone でサポートされる Amazon EC2 互換 AWS CLI コマンドの一覧](#)」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 AWS Snowcone 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、ボリュームをホストする AWS Snowcone デバイスで実行されている EC2-compatible 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) に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。

- Amazon EC2 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 Snowcone 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 ボリュームの合計容量は、AWS Snowcone デバイスの使用可能な領域によって制限されます。

## Snow 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「[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。Amazon EC2

Snowcone 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、Snowcone デバイスではサポートされません。

次に、Snowcone 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との上記以外の違いを説明します。

- 各 Snowcone では、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[Snowcone でサポートされる Amazon EC2 互換 AWS CLI コマンドの一覧](#)。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 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。Snowcone では、コンピューティングインスタンス用に、インスタンスメタデータのカテゴリのサブセットがサポートされています。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。

Snowcone でサポートされているインスタンスメタデータのカテゴリ

[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。
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。

[データ]	説明
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。

### Snowcone でサポートされているインスタンス動的データのカテゴリ

[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 <a href="#">ユーザーガイド</a> 」の「 <a href="#">インスタンス ID ドキュメント</a> 」を参照してください。 Amazon EC2

## Snowcone コンピューティングインスタンスのユーザーデータの変更

ユーザーデータは、Snowcone デバイスの、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。Snowcone 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。

## Snowcone 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

次に、コンピューティングインスタンスを持つ Snowcone ジョブの、トラブルシューティングのヒントをご紹介します。

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を起動したとき Snowcone が応答しなくなる](#)
- [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 RJ45 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。Snowcone には RJ45 インターフェイスが 2 つあるため、間違った物理インターフェイスを指定している可能性があります。

### 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を起動したとき Snowcone が応答しなくなる

Snowcone が、インスタンスの起動を停止させている可能性があります。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。Snowcone の HTTP または HTTPS Amazon EC2 エンドポイントに対して AWS CLI コマンド `aws ec2 describe-instances` を実行すると、インスタンスのステータスを確認できます。

## 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 インスタンスには、1 つのルートボリュームがあります。

ボリュームの追加に関する追加情報については、「<https://docs.aws.amazon.com/snowball/latest/snowcone-guide/snowcone-snowcone-efs.html>」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
```

## AWS DataSync を使用してファイルを転送する

AWS DataSync は、インターネットまたは 経由での AWS ストレージサービスとの間の大量のデータのコピーを簡素化、自動化、高速化するオンラインデータ転送サービスです AWS Direct Connect。DataSync エージェントは、Snowcone デバイスにプリインストールされています。デバイスと Amazon S3 バケット、Amazon EFS Amazon FSx for Windows File Server の間でデータを転送できます。は、ファイルやオブジェクトの移動、データ転送のスケジュール、転送の進行状況のモニタリング、暗号化、データ転送の検証、および問題が発生した場合の顧客への通知 AWS DataSync を自動的に処理します。

DataSync エージェントを起動する前に、Snowcone デバイスでネットワークファイルシステム (NFS) を有効にします。「[NFS インターフェイスの自動設定](#)」および「[NFS の開始とアクセスの制限](#)」を参照してください。

DataSync エージェントは、Snowcone ジョブの準備中に AMI として Snowcone デバイスにプリインストールされています。データを にオンラインで転送するには AWS、Snowcone デバイスを外部ネットワークに接続し、AWS OpsHub または CLI を使用して DataSync エージェント AMI を起動します。DataSync エージェントを AWS マネジメントコンソールでアクティブ化するか、CLI を使用して、Snowcone NFS ストアと Amazon S3、Amazon EFS、または Amazon FSx for Windows File Server の間でオンラインデータ転送タスクを設定します。

Snowcone で AWS DataSync 実行されている は、次の目的で使用できます。

- エッジコンピューティングアプリケーション、データの収集および処理、迅速なインサイトの取得、オンラインによる AWS へのデータ転送。
- オンラインのセンサーやマシンによって継続的に生成されるデータを、ファクトリーまたは他のエッジロケーション AWS の に転送します。
- メディア、科学、またはその他のコンテンツを AWS ストレージサービスからパートナーや顧客にオンラインで配信します。
- エッジロケーションから にメディア、科学、またはその他のコンテンツをオンラインで転送してコンテンツを集約します AWS。

- DataSync エージェントの VNI に静的 IP アドレスを使用していることを確認してください。DHCP を使用すると、デバイスの再起動により、別の IP アドレスを使用していることを意味する可能性があるため、問題が発生する可能性があります。これにより、DataSync エージェントと DataSync AWS クラウドでのアクティベーションの両方を再設定する必要が強制されます。
- Snowcone で Datasync エージェントを使用するには、Snowcone の NFS 設定に AllowedHosts 「」 エントリに DataSync エージェント IP アドレスが含まれていることを確認する必要があります。これには、エージェントを有効にする DataSync ために Snowcone の NFS サービスを停止して再起動する必要があります。
- NFS サービスを停止するたびに、DataSync エージェントも停止する必要があります。注: リストされた DataSync エージェントを許可する NFS サービスを停止すると、Ddatasync エージェントが動作しなくなります。
- 「[AWS DataSync task quotas](#)」で、Snowcone デバイスでのタスクあたりの最大ファイル数を確認してください。200,000 ファイルの制限を超えると、DataSync タスクはメモリ割り当てエラーを報告し、実行を中止します。

ワンタイムエッジコンピューティングまたはデータ転送ワークフロー、またはワイドエリアネットワーク (WAN) リンクや不適切な WAN 帯域幅のないエッジロケーションの Snowcone ワークフローの場合、データ転送を完了するには Snowcone デバイスを AWS に返送する必要があります。

## 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。

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、[AWS Snowcone の仕組み](#)」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。

#### Note

Snowcone デバイスで使用可能なストレージ容量は、NFS サービスが開始されるまでは正確ではありません。

GUI ツールである `awscli` 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」を参照してください。<https://docs.aws.amazon.com/snowball/latest/snowcone-guide/manage-nfs.html>

## 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 を返しますINACTIVE。

```
{
  "ServiceId" : "nfs",
  "Status" : {
    "State" : "ACTIVE"
  }
}
```

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は次のとおりです。](#)

- 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
```

」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 VNIs がない場合は、`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
```

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を許可されている 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は次のとおりです。

- 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 サービスを停止してください。

```
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 Snowcone デバイスで AWS サービスを使用するために必要なポート

AWS サービスが AWS Snowcone 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	SSH	デバイスのヘルスチェックと EC2 SSH 用
2049	NFS	NFS エンドポイント
8008	HTTP	EC2 HTTP エンドポイント
8243	HTTPS	EC2 HTTPS エンドポイント
9091	HTTP	デバイス管理用エンドポイント

## Snowcone Device の返却

Snowcone デバイスへのデータ転送が終了したら、AWSに返送する準備をします。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。

デバイスとの通信がすべて終了したら、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。

## Snowcone Device の切断

Snowcone ケーブルを取り外します。デバイス上部の E Ink ディスプレイに返送用配送ラベルが表示されたら、返送できます。リージョンのキャリアを確認するには、[の配送に関する考慮事項 AWS Snowcone](#) を参照してください。

### ジョブ固有の考慮事項

#### Important

データをインポートする場合は、プロセスの最後へのインポート AWS が成功し、データ転送の結果を検証できるようになるまで、転送されたデータのローカルコピーを削除しないでください。

#### Note

Snow デバイスを返送して Amazon S3 にインポートすると、デバイスが改ざんされておらず、デバイスが正常であることを確認すると、データの取り込み AWS を開始します。デバイス上のデータを送信先の S3 バケットに取り込みたくない場合は、Snow ジョブのキャンセルをリクエストできます。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。

デバイスの配送方法については、[の配送に関する考慮事項 AWS Snowcone](#) を参照してください。

# デバイスのデータの保護

デバイス上のデータを保護するために、次の推奨事項を検討してください AWS Snowcone 。

トピック

- [の保護 AWS Snowcone](#)
- [NFC タグの検証](#)

## の保護 AWS Snowcone

以下は、Snowcone 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 Snowcone デバイスは頑丈に作られていますが、天候によってハードウェアの耐久性に支障が生じる場合があります。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

Snowcone 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。

- 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。Snowcone では、データ保護のために、不正開封防止筐体、256 ビットの暗号化、およびデータのセキュリティと完全な保管継続性を提供するための業界標準である TPM など、数重に設計されたセキュリティ機能を採用しています。
- データ転送ジョブが処理および検証されると、 は、メディアのサニタイズに関する米国国立標準技術研究所 (NIST) ガイドラインに従って Snowcone デバイスのソフトウェア消去 AWS を実行します。

## NFC タグの検証

AWS Snowcone 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 で利用可能な Snowcone 検証アプリケーション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。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。

以下に、Snowcone デバイスで NFC タグを検証する方法を説明します。開始する前に、「使用開始」演習の以下の最初のステップを実行したことを確認してください。

1. 最初の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。
3. ローカルネットワークに接続します。
4. 認証情報とツールを取得します。詳細については、「[認証情報の取得](#)」を参照してください。
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、「[AWS Snowball Edge クライアントの使用](#)」を参照してください。

デバイス内の NFC タグを検証するには AWS Snowcone

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。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. 電話を使用して NFC タグをスキャンするには、Snowcone Verification App をダウンロードしてインストールします。Android スマートフォンを使用している場合は、Google Play ストアからそのアプリをダウンロードします。
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。

- 別のスマートフォンでアプリをダウンロードして、もう一度試します。
- 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。

# AWS Snowcone ジョブステータスについて

AWS Snowcone ジョブを作成すると、ジョブのステータスが遷移し、ステータスが 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。

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。

## 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。

## AWS Snowcone デバイスジョブのステータス

[Job Status]	意味
Job created	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	AWS はジョブ用のデバイスを準備しています。
Preparing shipment	AWS は、デバイスの配送を準備しています。
In transit to you	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。

[Job Status]	意味
Delivered to you	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。
への転送中 AWS	デバイスを に返送しました AWS。
仕分け施設にあります	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	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	AWS は Amazon S3 にデータをインポートしています。
完了	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	ジョブがキャンセルされました。

# 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" : "dc1e94d9-56c5-5e96-808d-cc7f68faa162",
  "TopicArn" : "arn:aws:sns:us-east-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" : "FMG5t1ZhJNHLHUXvZgtZz1k24FzVa7oX0T4P03neeXw8ZEXZx6z35j2F0TuNYShn2h0bKNC/zLTnMyIxEzmi2X1sh0BWsJHkrW2xkR58ABZF+4uWHEE73yDVR4SyYAIkP9jstZzDRm+bcVs8+T0yaLiEGLrIIIL4esi11lhIkgErCuy5btPcWXBdio2fpCRD5x9oR6gmE/rd5071X1c1uvnv4r1Lkk4pqP2/iUfxFZva1xLSRvgyfm6D9hNk1VyPfy+7Ta1MD01zmJu0rExtnSIbZew3foxgx8GT+1bZkLd0ZdtRj1IyPRP44eyq78sU0Eo/LsDr0Iak4ZDpg8dXg==",
  "SigningCertURL" : "https://sns.us-east-1.amazonaws.com/SimpleNotificationService-010a507c1833636cd94bdb98bd93083a.pem",
  "UnsubscribeURL" : "https://sns.us-east-2.amazonaws.com/?Action=Unsubscribe&SubscriptionArn=arn:aws:sns:us-east-2:111122223333:ExampleTopic1:e1039402-24e7-40a3-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 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&amp;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
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
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 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&amp;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 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&amp;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/
zLTnMyIxEzmi2X1sh0BWsJHkrW2xkR58ABZ
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&amp;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	<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 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&amp;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
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+T0yaLiEGLrIIIL4esi11lhIkq 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&amp;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"
}
```

# AWS Snowcone 注文プロセスについて

AWS Snowconeの注文には、次の2つのオプションがあります。注文には、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 (JMAPI) を使用できます。

## 配送プロセスについて

このセクションでは、AWS Snowcone デバイスの配送処理方法に関する情報と、AWS リージョン サポートされているリストについて説明します。サポートされるリージョンおよび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。ジョブで選択した配送レートが、ジョブで使用する Snowcone デバイスの配送および受領に適用されます。配送料金の詳細については、[AWS Snowcone の料金](#)を参照してください。

### Note

Snowcone デバイスは、デバイスを注文した AWS リージョン内での、データのインポートまたはエクスポートのみに使用できます。

## Snowcone デバイスの返却

E Ink ディスプレイの前払い配送情報に、デバイスの返送先の正しい住所が記載されています。Snowcone デバイスの返送方法については、[配送業者](#)を参照してください。Snowcone デバイスは、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。パッケージの追跡にはリージョンのキャリアをご使用いただけます。AWS Snow ファミリー マネジメントコンソールを使用してジョブのステータスの変化を追跡できます。

### Important

から別途指示がない限り AWS、Snowcone 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。

## の使用 AWS Management Console

AWS Snow ファミリー マネジメントコンソールを使用して、Snowcone デバイスを注文できます。

## コンソールから Snowcone を注文する

AWS Snowball コンソールを使用して Snowcone を注文する手順については、「」を参照してください [開始方法](#)。

## ジョブ管理 API の使用

ジョブ管理 API (JMAPI) は、AWS Snow ファミリーマネジメントコンソールで利用できる、プログラムによる同じ機能へのアクセスを提供します。これにより、ジョブの機能を自動化できます。JMAPI を使用すると、ジョブのステータスの表示、ジョブの作成、マニフェストファイルのダウンロード、コードのロック解除、ジョブ完了レポートの表示を行うことができます。呼び出しは API を介して行われるため、これらの呼び出しをカスタムアプリケーションまたはウェブフロントエンドに統合できます。

### トピック

- [JMAPI の一般的な用途](#)
- [JMAPI 必須の文字列](#)
- [JMAPI エンドポイント](#)
- [JMAPI CLI コマンド](#)
- [例](#)

### JMAPI の一般的な用途

- Snowcone デバイスの注文の自動化
- マニフェストファイルのダウンロード
- ロック解除ファイルのダウンロード
- 現在の Snowcone ジョブの一覧表示
- Snowcone ジョブ完了レポートのダウンロード

### JMAPI 必須の文字列

ジョブ管理 API を使用して注文を行う場合は、例で表示される、次の必須パラメータを使用します。

```
--job-type
```

```
--resources
--address-id
--region
--role-arn
--kms-key-arn
--shipping-option
--device-type
--description
```

## JMAPI エンドポイント

### API エンドポイント

各エンドポイントを呼び出す場合、形式は `snowballEdge.region.amazonaws.com` になります。以下は、エンドポイントの内訳の理解に役立ついくつかの例です。

#### Example

リージョン	エンドポイント
米国東部 (バージニア北部)	snowball.us-east-1.amazonaws.com
米国西部 (オレゴン)	snowball.us-west-2.amazonaws.com

## JMAPI CLI コマンド

### ジョブ管理 CLI

以下は、ジョブ管理 API に対して実行できる CLI 呼び出しの例です。

Command	例
ジョブの一覧表示	aws snowball list-jobs
ジョブの説明	aws snowball describe-job --job-id <i>[JOB ID]</i>
アドレスの説明	aws snowball describe-address -- address-id

Command	例
住所の作成	<code>aws snowball create-address --cli-input-json file://create-address.json</code>
ジョブを作成する	<code>aws snowball create-job --cli-input-json file://create-job.json</code>
ジョブをキャンセルする	<code>aws snowball cancel-job --job-id <i>[JOB ID]</i></code>

## 例

以下は、ジョブ管理 API を使用するコマンドの例です。

### KMS JSON の例

以下の JSON の例は、AWS KMS ポリシーファイルを使用するために適切にフォーマットされた JSON ファイルです。

```
{
  "KeyMetadata": {
    "Origin": "AWS_KMS",
    "KeyId": "1234abcd-12ab-34cd-56ef-1234567890ab",
    "Description": "",
    "KeyManager": "CUSTOMER",
    "Enabled": true,
    "KeyUsage": "ENCRYPT_DECRYPT",
    "KeyState": "Enabled",
    "CreationDate": 1502910355.475,
    "Arn": "arn:aws:kms:us-west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
    "AWSAccountId": "111122223333"
  }
}
```

### アドレス作成の例

以下は、コマンドをフォーマットしてアドレスを作成し、成功した場合の応答の例です。

```
aws snowball create-address --address "Name=Bob,Company=AWS,Street1=1234 Fake
St.,City=All,StateOrProvince=Any,Country=US,PostalCode=12345,PhoneNumber=1234567890"
```

## 出力例

```
{
  "AddressId": "ADID3be640c8-1111-1111-1111-917f201ffa42"
}
```

## ジョブ作成の例

以下は、create-jobコマンド実行のためのコマンドの例です。

```
aws snowball create-job --job-type IMPORT --resources file://path/to/resources.json
--address-id ADID3be640c8-1111-1111-1111-917f201ffa42 --region us-east-1 --role-
arn arn:aws:iam::123456789123:role/example_role --kms-key-arn arn:aws:kms:us-
west-2:000000000000:key/Example --snowball-capacity-preference T14 --device-
configuration file://path/to/configuration.json --shipping-option SECOND_DAY --
snowball-type SNC1_SSD
```

上記の create-job コマンドは、us-east-1 リージョンで、SECOND\_DAY 配送オプションがあり、T14 容量の設定を持つ SNC1\_SSD タイプの Snowcone デバイスでインポートジョブを作成します。

## AMI をエクスポートして Amazon EC2 ジョブで使用する

このセクションでは、AWS Snowcone デバイスの Amazon EC2-compatible をエクスポートする方法の概要を説明します。

### トピック

- [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](#)
- [コンソールを使用してジョブを作成する](#)
- [AWS CLI を使用してジョブを作成する](#)

## 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 Snowcone 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、Snowcone デバイスを受け取っても、SSH を使用してインスタンスに接続することはできません。

SSH キーを AMI に追加するには

1. 互換性のある AMI イメージ AWS クラウド を使用して 新しいインスタンスを起動します (「」を参照<https://docs.aws.amazon.com/snowball/latest/developer-guide/using-ami.html>)。

インスタンスの起動時に、インスタンスに割り当てるストレージサイズが、後ほど Snowcone デバイスで使用する際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。Snowcone のコンピューティングインスタンスストレージ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、「[ec2-snowcone-limits](#)」を参照してください。

2. Snowcone で実行するアプリケーションをインストールして設定し、正常に動作することをテストします。
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。Snowcone 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 EBS-backed Linux AMI を作成する](#)」を参照してください。
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。

## コンソールを使用してジョブを作成する

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追加します。

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。

## AWS CLI を使用してジョブを作成する

AWS Command Line Interface () を使用してジョブを作成することもできますAWS CLI。これを行うには、ターミナルを開き、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::123456789012:role/snowcone-import-snowcone-role --shipping-option SECOND_DAY --snowball-type SNC1_HDD --snowball-capacity-preference T8 --device-configuration '{"SnowconeDeviceConfiguration":{"WirelessConnection":{"IsWifiEnabled": false}}}'
```

デバイスの到着後に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。

## の配送に関する考慮事項 AWS Snowcone

以下に、AWS Snowcone デバイスの配送処理方法に関する情報と、AWS リージョン サポートされている各を示すリストを示します。ジョブに選択した配送レートは、そのジョブに使用される AWS Snowcone デバイスの送受信の両方に適用されます。配送料金の詳細については、[AWS Snowcone の料金](#)を参照してください。

### トピック

- [配送用 AWS Snowcone デバイスの準備](#)
- [リージョンごとの配送制限](#)
- [AWS Snowcone デバイスの配送](#)

Snow Family デバイスを注文するジョブを作成するときは、配送先住所と配送速度を指定します。この配送速度は、ジョブを作成した日から AWS Snowcone デバイスがどのくらい早く届くかを示すものではありません。デバイスが AWS と配送先住所の間で転送されている時間のみが表示されます。この時間には、処理にかかる時間は一切含まれていません。処理時間はジョブタイプ (エクスポートは通常インポートより長くなります) などの要素によって異なります。また、通信事業者は通常、1日に1回だけ送信 AWS Snowcone デバイスを集荷します。したがって、配送前の処理に1日以上かかることが可能です。

#### Note

Snow Family デバイスは、注文したのと同じ AWS リージョン にのみ返すことができます。一部の AWS リージョンでは、Snow Family デバイスを注文した国とは異なる国に送信できます。詳細については、「[リージョンごとの配送制限](#)」を参照してください。

## 配送用 AWS Snowcone デバイスの準備

以下で、Snowcone を AWS に返送するための準備方法を説明します。

AWS Snowcone デバイスを配送用に準備するには

1. デバイスとの間でこのジョブのすべてのデータの転送が完了していることを確認します AWS Snowcone 。 [デバイスのロックを解除します。](#)

2. デバイス前面のインジケータライトの近く、ネットワークポートの反対側にある電源ボタンを押します。約 20 秒後にデバイスの電源がオフになります。

Snowcone デバイスの電源を切って電源ケーブルを抜いた後、約 1 分以内に E Ink 画面に配送情報が表示されない場合は、「[Snow Family デバイスの返却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。

## 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、AWS リージョン データが存在する からコンソールにサインインする必要があります。Snow Family デバイスは、アジアパシフィック (インド) からアジアパシフィック (オーストラリア) など、国際配送はされません。

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

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。

## AWS Snowcone デバイスの配送

E Ink ディスプレイの前払い配送情報に、AWS Snowcone デバイスの返送先の正しい住所が記載されています。AWS Snowcone デバイスを返却する方法については、「」を参照してください [the section called “配送業者”](#)。AWS Snowcone デバイスは、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。パッケージの追跡にはリージョンのキャリアをご使用いただけます。

を使用して、ジョブのステータス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。

Snowcone デバイスの電源を切って電源ケーブルを抜いた後、約 1 分以内に E Ink 画面に配送情報が表示されない場合は、「[Snow Family デバイスの返却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。

## 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 Snowcone 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 AWS お客様への AWS Snowcone デバイスの配送を処理し、に帰ります AWS。AWS Snowcone デバイスが出荷されると、追跡番号が取得されます。各ジョブの追跡番号や追跡を行うウェブサイトへのリンクは、[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 への API コールを使用して確認できます。

以下は、リージョン別の AWS Snowcone デバイスでサポートされているキャリアのリストです。

- 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、イスラエル、シンガポールの配送業者は、Kuehne + Nagel です。
- 中国の配送業者は、S.F. Express です。
- その他のリージョンのキャリアはすべて、[UPS](#) です。

## AWS Snowcone カナダ、欧州、南アフリカ、米国での集荷

カナダ、欧州、南アフリカ、米国では、UPS が AWS Snowcone デバイスを集荷できるように、次の情報に注意してください。

- UPS で直接集荷を手配 AWS Snowcone するか、デバイスを UPS の集荷施設に持ち込んでに配送するように手配します AWS。
- E Ink ディスプレイの前払い UPS 配送ラベルには、AWS Snowcone 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 Snowcone 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 から配送の追跡番号が自動的に提供されます。

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

## AWS Snowcone ブラジルでの集荷

ブラジルでは、UPS による Snowcone の集荷について以下の点に注意してください。

- Snowcone 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。
- Snowcone 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。
- Snowcone デバイスを返送するときは適切な文書を発行する必要があります。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、宣言書のコピーを 4 つ作成することをお勧めします。1 つは記録用、もう 1 つは移送用です。

## AWS Snowcone イスラエルでの集荷

イスラエルでは、[snowball-shipping@amazon.com](mailto:snowball-shipping@amazon.com) に AWS メールで連絡して集荷を手配します。件名に「Snowcone Pickup Request」と入力し、次の情報を記載してください。

- ジョブ ID – 返送する Snow デバイスに関連付けられたジョブ ID。ジョブ ID は、[AWS Snow ファミリーマネジメントコンソール](#) で検索できます。
- 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。
- 連絡先情報の詳細 – 配送サービスの Kuehne + Nagel が、必要に応じて集荷の情報として使用できる名前、E メールアドレス、および市内の国内電話番号。

すぐに、 からフォローアップの E メールが届きます。この E メール AWS Support には、指定した住所のデバイスの集荷に関する情報が記載されています。通常は、現地時間の 12:00 までにデバイスの出荷準備を終えて、集荷に備えてください。[配送用 AWS Snowcone デバイスの準備](#) を参照してください。

## AWS Snowcone 英国での集荷

英国では、UPS が AWS Snowcone デバイスを集荷するために、次の情報に注意してください。

- UPS と直接集荷を手配 AWS Snowcone するか、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 Snowcone 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 Snowcone 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レポートされます。

### Important

が別途指示しない限り AWS、 AWS Snowcone デバイス に個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。

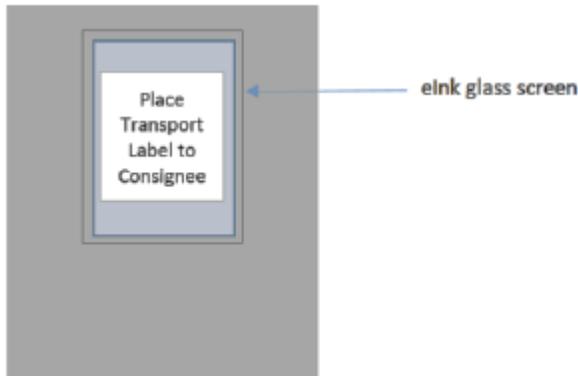
Snow Family 製品に対する UPS サービスは、その国の国内でのみ提供されます。

### 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。

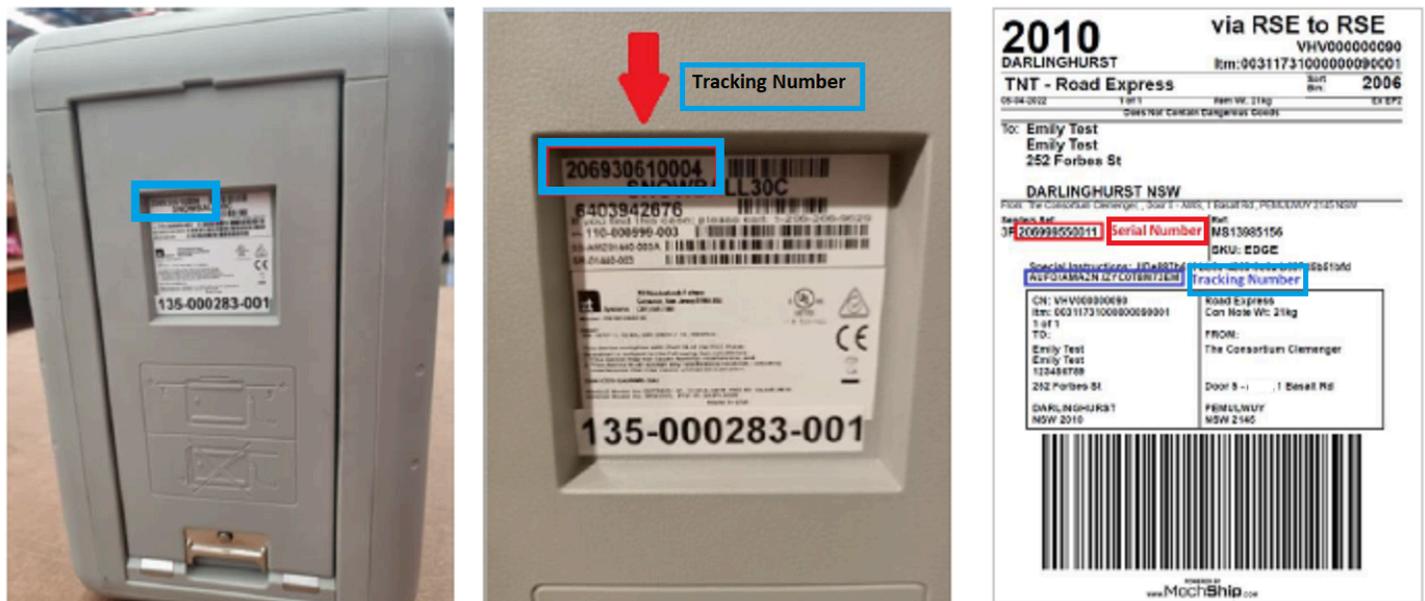
## AWS Snowcone オーストラリアでの集荷

オーストラリアでは、AWS Snowcone デバイスを に返送する場合は AWS、Snow デバイスの eInk 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。



Snow 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](mailto:knau.snowball_return@kuehne-nagel.com) に送信してください。

*Example where to locate tracking number and serial number*



Snow デバイスの返送を手配するには、以下の情報を添えて、[knau.snowball\\_return@kuehne-nagel.com](mailto:knau.snowball_return@kuehne-nagel.com) に E メールを送信してください。

- 氏名
- 追跡番号 (青で表示)
- デバイスのシリアル番号 (赤で表示)

- 集荷先の完全な住所
- 集荷時の連絡先担当者
- 集荷時の連絡先電話番号
- 集荷日
- 集荷時間帯 (営業時間内に 3 時間以上)

#### Note

集荷日時は、営業日の営業時間内である必要があります。  
Kuehne + Nagel チームが返信し、集荷依頼を受け取ったことを確認します。

## AWS Snowcone インドでの集荷

インドでは、Blue Dart が Snowcone デバイスを集荷します。Snowcone デバイスの返送準備ができたなら、電源をオフにして、返送の準備をしてください。集荷を手配するには、件名を「Snowcone Pickup Request」とし、[snowball-pickup@amazon.com](mailto:snowball-pickup@amazon.com) 宛てに E メールを送信します。E メールには、以下の情報を記載してください。

- ジョブ ID – AWS に返送する Snowcone のジョブ ID。
- 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) – Snowcone 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) – Snowcone 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、Snowcone の集荷に関する特記事項。

Snowcone 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 は配送ラベル用紙を提供し、Snowcone デバイスを集荷します。

#### Important

インドで Snowcone 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 Snowcone 韓国での集荷

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、件名を「Snowcone Pickup Request」とし、[snowball-shipping@amazon.com](mailto:snowball-shipping@amazon.com) 宛に E メールを送信していただくと、当社が集荷を手配いたします。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 Snowcone 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、 からフォローアップメールが届きます。このメール AWS Support には、指定した住所のデバイスの集荷に関する情報が記載されています。デバスの出荷準備をして (「[配送用 AWS Snowcone デバイスの準備](#)」を参照)、集荷に備えてください (通常は午後 1 時～午後 3 時の間)。

## AWS Snowcone 香港での集荷

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 Snowcone 集荷リクエストを記載した E メールを [snowball-shipping-ap-east-1@amazon.com](mailto: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
- 希望のデバイス集荷日
- 集荷が可能な最早時間
- 集荷が可能な最遅時間
- 集荷先住所

### Note

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 によって配信されます。返送の S.F. Express の追跡番号を使用して、配送完了日時を確認してください。

## AWS Snowcone インドネシア、日本、シンガポールでの集荷

インドネシア、日本、シンガポールでは、デバイスを返却する準備ができたなら、E Ink の返品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

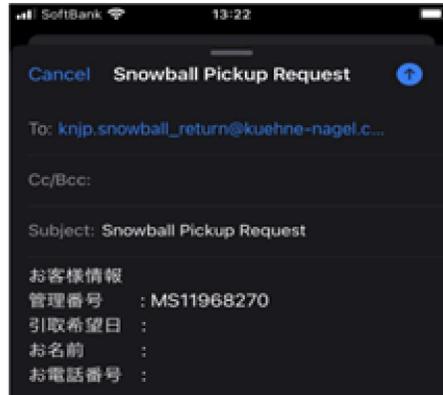
AWS Jobs ID QF6LNZGKTZPF  
シリアル番号 2R 207138750022  
管理番号 MS14003547



#### 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



## 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 Snowcone 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、インドネシア、日本、シンガポール、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せん。通常、至急便で発送された Snowcone 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。

- インド – インド国内で配送する場合、AWS Snowcone デバイスは関連するすべての税務書類 AWS を受け取ってから 7 営業日以内に発送されます。
- イスラエル – イスラエル国内での配送の場合、同日配送を利用できます。
- アメリカ合衆国 (米国) およびカナダ – 米国およびカナダ国内の配送の場合、翌日配送と翌々日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 Snowcone 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。

# Snowcone 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 Snowcone 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、デバイスで実行されているサービスを停止し、更新の完了後に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) ものとします。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ます。

## 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。
- Snowcone 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、[AWS「Snowball Edge Client 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、「」<https://docs.aws.amazon.com/snowball/latest/snowcone-guide/snowcone-using-client-commands.html#snowcone-client-configuration>を参照してください。

上記のタスクが完了すると、Snowcone 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。Snowcone デバイスの自動更新に関する詳細は、「[Snowcone の更新](#)」の「`configure-auto-update-strategy`」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
```

Snowcone ソフトウェアの更新をチェック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、Snowcone 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
```

- 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
```

## 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。

### 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

- ターミナルウィンドウを開き、Snowcone 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。
- 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りますを示しています。

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 `serviceId` があります。

```
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]
```

Snowcone 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、「」を参照してください<https://docs.aws.amazon.com/snowball/latest/snowcone-guide/reboot.html>。デバイスのロック解除の詳細については、ロック解除 Snow [Family デバイスのロック解除](#)」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、Snowcone 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せん。

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。

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. コードとマニフェストファイルを[使用します](#)。

## 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 Snowcone デバイスのベストプラクティス

デバイスのメリットと満足度を最大限に高めるために AWS Snowcone、以下のベストプラクティスに従うことをお勧めします。

## トピック

- [セキュリティ](#)
- [ネットワーク](#)
- [リソース管理](#)
- [EC2 互換インスタンスの管理](#)

## セキュリティ

- AWS Snowcone デバイスについて不審な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 Snowcone 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。これらを個別に保存すると、権限のないユーザーが AWS Snowcone 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。このアプローチでは、AWS Snowcone 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そのユーザーの 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の `snowballEdge list-access-keys` および `snowballEdge get-secret-access-key` を実行すると表示される認証情報は、アクセスキーとシークレットキーのキーペアです。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 や他のにはマッピングされません AWS アカウント。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。
- NFS 共有へのアクセスを制限することができます。詳細については、「[NFS 実行中に NFS 共有へのアクセスを制限する](#)」を参照してください。
- Snowcone デバイスの電源を切ったり、電源を入れ直したりすると、ロック状態になります。

## ネットワーク

- 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 AWS Snowcone、一度に 1 つの方法のみを使用することをお勧めします。NFS マウントと

同じ S3 バケット DataSync 上の NFS マウントと を同時に使用すると、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 Snowcone 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。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場合があります。
- AWS Snowcone 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 [Snowcone のパフォーマンス](#)。

## リソース管理

- オンプレミスデータ転送を実行するための 5 日間の無料期間は、AWS Snowcone 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。

## EC2 互換インスタンスの管理

AWS Snowcone 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、保持する Amazon EC2 互換インスタンスを一時停止します。

## Snowcone のパフォーマンス

次の表は、ネットワークの転送速度が、AWS Snowcone にデータを埋めるのにかかる時間にどのように影響するかを示しています。

レート (MB/秒)	8 TB の転送時間	4 TB の転送時間
100	21.17 時間	10.59 時間
60	36.57 時間	18.29 時間
30	68.57 時間	34.29 時間
10	210.29 時間	105.15 時間

# AWS Snowcone クォータ

以下は、Snowcone デバイス使用時のクォータに関する情報です。

## Snowcone ディスクストレージ

Snowcone デバイス内部には、8 TB または 14 TB のディスクストレージが装備されています。このディスクストレージは、ローカルの Amazon Elastic Block Store (Amazon EBS) ボリュームプレゼンテーションを通じて、内部ネットワークファイルシステム (NFS) サービスまたはローカルの Amazon EC2 互換インスタンスで使用できます。このストレージは NFS または Amazon EBS のどちらかに使用できますが、両方には使用できません。ユースケースに応じて、ストレージを割り当てることができます。Snowcone の注文時に、ストレージが事前に割り当てられていることに注意してください。

## コンピューティングジョブストレージ

ジョブタイプがローカルコンピューティングの場合、合計 8 TB または 14 TB のローカルの Amazon EBS ボリュームを作成して、Amazon EC2 互換インスタンスにアタッチできます。Amazon EBS ボリュームを使用すると、ローカルの Amazon EC2 インスタンスは、ルートボリューム単体よりも多くのローカル容量にアクセスできます。これはローカルストレージ単体であり、Amazon S3 へインポートできないため、デバイスを AWS に返却すると、Amazon EBS ボリュームに書き込まれたデータは失われます。

## トピック

- [コンピューティングリソースのクォータ](#)
- [Snowcone デバイスの配送に関する制限事項](#)
- [インポート用に返却される Snowcone デバイスの処理の制限](#)
- [使用可能 AWS リージョン](#)

## コンピューティングリソースのクォータ

ストレージで使用可能な Snowcone デバイスクォータのコンピューティングリソース

次の表に、Snowcone デバイスで使用可能なコンピューティングリソースとそのストレージクォータの概要を示します。

インスタンスタイプ	vCPU コア	メモリ (GiB)	クォータ
snc1.micro	1	1	2
snc1.small	1	2	2
snc1.medium	2	4	1

## Snowcone デバイス用の Amazon マシンイメージ (AMI) と Amazon EC2 容量クォータ

次の表に、Snowcone デバイスの AMI および Amazon EC2 互換インスタンスの容量クォータの概要を示します。

Amazon EC2	サイズ
Amazon EC2 互換 AMI	125 GB (お客様のジョブで使用されるすべての AMI の合計サイズ)
Amazon EC2 互換インスタンス	150 GB (インスタンスのバックイングストレージボリュームで使用可能な領域を示します)

## Snowcone デバイスの配送に関する制限事項

Snowcone デバイスの配送に関する制限事項は次のとおりです。

- AWS は、Snowcone デバイスを私書箱に配送しません。
- ジョブ作成時に指定された国以外の住所への Snowcone の移動は許可されず、AWS サービス条件に対する違反となります。

配送の詳細については、[の配送に関する考慮事項 AWS Snowcone](#) を参照してください。

## インポート用に返却される Snowcone デバイスの処理の制限

データをインポートするには AWS クラウド、Snowcone デバイスが次の要件を満たしている必要があります。

- Snowcone デバイスが侵害されていないこと。前面と背面のドアを開ける場合を除き、いかなる理由でも Snowcone を開けないでください。
- デバイスが物理的に破損していないこと。損傷を防ぐため、Snowcone デバイスの 2 つのドアは、ラッチでクリック音がするまで閉じてください。
- によって配送ラベルが提供されていない限り AWS、Snowcone デバイスの E Ink ディスプレイが表示され、Snowcone 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返送ラベルが表示される必要があります。

#### Note

上記の要件を満たしていない返却 Snowcone デバイスはすべて、作業を実行せずに消去されます。

## 使用可能 AWS リージョン

サポートされている AWS リージョン および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。

# Snowcone の問題のトラブルシューティング

このセクションでは、AWS Snowcone デバイス/サービスに関するガイダンスとインサイトを提供し、発生した問題のトラブルシューティングを行います。

## トピック

- [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](#)
- [ネットワークの問題のトラブルシューティング](#)
- [Datasync での Amazon EC2 互換インスタンスのトラブルシューティング](#)
- [データ転送の問題のトラブルシューティング](#)
- [Snow Family デバイスの返却に関する問題のトラブルシューティング](#)

## 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

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ドキュメントは、[こちら](#)をご覧ください。

## ネットワークの問題のトラブルシューティング

### IP アドレスが 0.0.0.0 と表示される

ネットワークと電源ケーブルは接続されているのに、デバイスには IP アドレスが 0.0.0.0 と表示されている

#### 実行するアクション

Snowcone とネットワークデバイス間のケーブル接続が正しくチェックされていることを確認します。

それでもデバイスに IP アドレスが表示されない場合は、ルーターで、IP 設定の失敗につながる可能性のある誤動作や、設定に関連する問題がないか確認します。

ネットワークで DHCP が有効になっていない場合は、Snowcone ディスプレイの STATIC オプションを使用してデバイスに静的 IP を設定し、適切な静的 IP アドレスでデバイスをプログラムしていることを確認します。

## デバイスのロックを解除できない

OpsHub または Snowball エッジクライアントを使用してデバイスのロックを解除することはできません。

### 実行するアクション

クライアントが、AWS Snowcone デバイスと同じサブネット、同じネットワーク内にあることを確認します。

Snowcone に複数の IP アドレス割り当て (イーサネットなど WiFi) がある場合は、複数のクライアントが Snowcone に応答しようとしている場合に、同じ IP アドレスを使用して接続していることを確認します。一度に使用できるインターフェイスは 1 つだけです。

上記の手順が機能しない場合は、デバイスからのログ AWS Support の提供にエンゲージしてください。Snowball Edge クライアントとコマンド snowballEdge を使用します。get-support-logs

## Datasync での Amazon EC2 互換インスタンスのトラブルシューティング

### エラー: インスタンスの起動に失敗しました

Snowcone に、このリクエストのインスタンスを起動するための容量が不足しています。

### 実行するアクション

エージェントとして AWS Snowcone デバイスで実行する要件では DataSync、デフォルトのインスタンス snc1.medium を使用します。このインスタンスは、2 つの CPU コアと 4 GiB のメモリを提供します。Snowcone デバイスに、新しい Amazon EC2 互換インスタンス起動リクエストをプロビジョニングするのに十分なリソースがあることを確認してください。

## データ転送の問題のトラブルシューティング

### サーバーによってアクセスが拒否されました

エラー: mount.nfs - 192.168.1.214 のマウント中にサーバーによってアクセスが拒否されました

## 実行するアクション

Quick Setup を使用して NFS を設定した場合、デフォルトでは、このデバイスにファイルを転送できるのはシステムだけです。他のホストが Snowcone にデータをアップロードできるようにする場合は、NFS サービスを無効にし、NFS 設定で許可する IP アドレスのリストを使用してサービスを再度有効にします。

## データ転送中に接続がタイムアウトする

Amazon S3 インターフェイス (経由) AWS Snowcone を使用して にデータを転送しようとする、接続がタイムアウトします AWS CLI。

### この正常な動作の理由

AWS Snowcone デバイスは、オフライン転送とオンライン転送の NFS マウント経由 AWS DataSync の転送のみをサポートします。Amazon S3 インターフェイスは現在サポートされていません。AWS CLIを使用して接続するとタイムアウトが発生するのは、これが理由です。

## Spawn Showmount ENOENT

エラー: Uncaught Error - spawn showmount ENOENT これは Error NO ENTry (または Error NO ENTity) の省略形で、実際にはファイルやディレクトリ以外でも使用されます。

### 実行するアクション

このエラーは、NFS クライアントがワークステーションで実行されていない場合に発生する可能性があります。で NFS を設定する前に NFS サービスが開始されなかった場合は AWS OpsHub、このエラーが表示されることがあります。

Windows クライアントでは、次の手順を実行して NFS サービスが実行されているかどうかを確認できます。

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。
3. [Services for NFS] で、[Client for NFS] を選択し、[OK] を選択します。 [「Windows クライアントへの NFS のマウント」](#) を参照してください。

このエラーは、実行中のワークステーション AWS OpsHub とデバイス の間に接続をブロックする可能性のあるファイアウォール/ウイルス対策がある場合にも発生する可能性があります。

ワークステーションから、NFS ポート番号 2049 の AWS Snowcone デバイス IP アドレスコマンドに telnet を実行して、接続が正常に確立されたかどうかを確認します。

## Snow Family デバイスの返却に関する問題のトラブルシューティング

Snow Family デバイスの電源を切って返却の準備をしても、E Ink ディスプレイに返品配送情報が表示されないことがあります。

### 実行するアクション

1. にログインします AWS Snow ファミリーマネジメントコンソール。
2. デバイスのジョブを表示します。
3. [詳細] セクションの [配送ラベル] セクションで、[返品ラベルを表示して印刷するには、次のリンクをクリックしてください] を選択します。
4. ラベルを印刷し、デバイスに貼付します。
5. [Snowcone デバイスの返却](#) に従ってデバイスを返却します。

# ジョブ管理 API リファレンス

- [ジョブ管理 API リファレンス](#)
  - [アクション](#)
  - [データ型](#)
  - [共通パラメータ](#)
  - [共通エラー](#)

# AWS Snowcone ユーザーガイドのドキュメント履歴

次の表に、の今回のリリースのドキュメントを示します AWS Snowcone。

- API バージョン: 最新
- ドキュメントの最終更新日: 2023 年 8 月 25 日

変更	説明	日付
<a href="#">デバイスの注文時にカスタム AMIs を含める</a>	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、「 <a href="#">Snowcone での Amazon EC2 の使用</a> 」を参照してください。	2023 年 11 月 15 日
<a href="#">新たに AWS リージョン サポート</a>	AWS Snowcone SSD デバイスがイスラエル (テルアビブ) で利用可能になりました AWS リージョン。このリージョンの集荷情報が追加されました。詳細については、「AWS 全般のリファレンス」の「 <a href="#">AWS Snow Family エンドポイントとクォータ</a> 」を参照してください。配送の詳細については、「 <a href="#">AWS Snowcone の配送に関する考慮事項</a> 」を参照してください。	2023 年 8 月 25 日
<a href="#">新たに AWS リージョン サポート</a>	Snowcone デバイスが欧州 (パリ) 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「 <a href="#">AWS Snow Family</a>	2022 年 1 月 29 日

[「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、「[AWS Snowconeの配送に関する考慮事項](#)」を参照してください。

### [Snowcone のトラブルシューティングに関する新しいヒント](#)

AWS Snowcone Snowball Edge と同様のトラブルシューティングのヒントセクションを追加しました。追加された新しいセクションについては、「[Snowcone の問題のトラブルシューティング](#)」を参照してください。

2022 年 4 月 15 日

### [新たに AWS リージョン サポート](#)

AWS Snowcone が、アジアパシフィック (ムンバイ) およびブラジルリージョンで利用可能になりました。EU、カナダ、シンガポール、ブラジルの集荷情報が追加されました。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。

2022 年 2 月 23 日

### [新たに AWS リージョン サポート](#)

AWS Snowcone が欧州 (ロンドン) リージョンで利用可能になりました。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。

2022 年 1 月 5 日

## [Network Time Protocol \(NTP\) サーバー設定をサポート](#)

Snowcone デバイスは、外部のNetwork Time Protocol (NTP) のサーバー設定をサポートするようになりました。

2021 年 11 月 16 日

## [新たに AWS リージョン サポート](#)

AWS Snowcone SSD が米国西部 (北カリフォルニア)、米国東部 (オハイオ)、アジアパシフィック (シンガポール)、アジアパシフィック (東京)、およびアジアパシフィック (シドニー) の各リージョンで利用可能になりました。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。

2021 年 11 月 3 日

## [新たに AWS リージョン サポート](#)

AWS Snowcone が、米国西部 (北カリフォルニア)、米国東部 (オハイオ)、南米 (サンパウロ) の各リージョンで利用可能になりました。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。

2021 年 9 月 29 日

## [新たに AWS リージョン サポート](#)

AWS Snowcone が、アジアパシフィック (シンガポール) およびアジアパシフィック (東京) リージョンで利用可能になりました。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。

2021 年 8 月 26 日

## [Snowcone デバイスを使用した、Amazon S3 からのオフラインデータエクスポートをサポート](#)

Snowcone デバイス AWS に転送して Amazon S3 データをエクスポートするようになり、リクエストできるようになりました。Snowcone デバイスはお客様の場所に物理的に配送されます。詳細については、[インポートおよびエクスポートジョブの仕組み](#)を参照してください。

2021 年 8 月 4 日

## [の紹介 AWS Snow Device Management](#)

Snow Device Management を使用すると、AWS Snowcone 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 Snowcone デバイスは Snow Device Management をサポートしており、Snowcone 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、「[を使用してデバイスを管理する AWS Snow Device Management](#)」を参照してください。

2021 年 8 月 4 日

### 新たに AWS リージョン サポートされる

AWS Snowcone がカナダ (中部) リージョンで利用可能になりました。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。

2021 年 4 月 28 日

### 新たに AWS リージョン サポート

AWS Snowcone がアジアパシフィック (シドニー) リージョンで利用可能になりました。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。

2021 年 3 月 24 日

### ダイレクトネットワークインターフェイスをサポート

AWS Snowcone では、ダイレクトネットワークインターフェイス (DNI) 設定が追加されました。これは、マルチキャストストリーム、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。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。

2021 年 1 月 12 日

## [新たに AWS リージョン サポート](#)

AWS Snowcone が欧州 (フランスフルト) リージョンで利用可能になりました。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。

2020 年 11 月 18 日

## [新たに AWS リージョン サポート](#)

AWS Snowcone が欧州 (アイルランド) リージョンで利用可能になりました。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。

2020 年 9 月 16 日

## [の紹介 AWS Snowcone](#)

AWS Snowcone は、エッジコンピューティングとデータ転送用のポータブルで堅牢、安全なデバイスです。を使用して、デバイスを AWS Snowcone に配送することでオフラインで AWS、またはを使用してオンラインで、データを収集 AWS、処理、およびに移動できます AWS DataSync。詳細については、「[AWS Snowcone とは](#)」を参照してください。

2020 年 6 月 17 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。