aws

Lustre ユーザーガイド

FSx for Lustre

FSx for Lustre: Lustre ユーザーガイド

Table of Contents

Amazon FSx for Lustre とは何ですか？

FSx for Lustre を使用すると、人気のあるハイパフォーマンス Lustre ファイルシステムを簡単かつ費用効果の高い方法で起動して実行できます。機械学習、ハイパフォーマンスコンピューティング (HPC)、ビデオ処理、財務モデリングなど、速度が重要なワークロードには Lustre を使用します。

Lustre ファイルシステムは、ストレージがコンピューティングに追随することが望まれる、高速ストレージを必要とするアプリケーション向けに設計されています。Lustre は、増加し続ける世界のデータセットを迅速かつ安価に処理するという問題を解決するために構築されました。Lustre は、世界最速のコンピュータ向けに設計された、広く使用されているファイルシステムです。ミリ秒未満のレイテンシー、最大数 TBps のスループット、および最大数百万の IOPS を提供します。Lustre の詳細については、Lustre のウェブサイトを参照してください。

フルマネージドサービスである Amazon FSx を使用すると、ストレージ速度が重要なワークロードで Lustre を簡単に使用できます。FSx for Lustre は、Lustre ファイルシステムの設定と管理における従来の複雑さを排除し、バトルテスト済みのハイパフォーマンスファイルシステムを数分でスピンアップおよび実行できるようにします。また、複数のデプロイオプションおよびストレージクラスが用意されているため、ニーズに合わせてコストを最適化できます。

FSx for Lustre は POSIX に準拠しているため、変更を加えなくても、現在の Linux ベースのアプリケーションを使用できます。FSx for Lustre は、ネイティブファイルシステムインターフェイスを提供し、他のあらゆるファイルシステムと同様に Linux オペレーティングシステムで機能します。また、書き込み後の読み込み整合性を実現し、ファイルのロックをサポートします。

トピック

複数のデプロイオプションとストレージクラス

Amazon FSx for Lustre は、さまざまなデータ処理のニーズに対応するために、スクラッチ ファイルシステムと 永続 ファイルシステムの選択肢を提供します。スクラッチファイルシステムは、テンポラリストレージと短期間のデータ処理に最適です。データはレプリケーションされず、ファイルサーバに障害が発生しても保持されません。整合性のあるファイルシステムは、長期的なストレージとスループット重視のワークロードに最適です。永続ファイルシステムでは、データがレプリケーションされ、障害が発生した場合はファイルサーバーが置き換えられます。詳細については、「FSx for Lustre ファイルシステムのデプロイおよびストレージクラスオプション」を参照してください。

Amazon FSx for Lustre は、さまざまなデータ処理要件に最適化されたソリッドステートドライブ (SSD)、Intelligent-Tiering、およびハードディスクドライブ (HDD) ストレージクラスを提供します:

- SSD ストレージクラスは、小規模でランダムなファイル操作を行い、最大 TBps のスループットを必要とするワークロード向けに最適化されています。データセット全体への一貫したミリ秒未満のレイテンシーアクセスを提供します。
- Intelligent-Tiering ストレージクラスは、データセット全体で一貫した低レイテンシーを必要としないほとんどのワークロードに適しており推奨されます。最大数 TBps のスループットと、オプションの SSD 読み取りキャッシュを使用して頻繁にアクセスされるデータへのミリ秒未満のレイテンシーアクセスを備えた、完全に伸縮自在でコスト効率の高いストレージを提供します。
- HDD ストレージクラスは、完全なデータセットに対して一貫した 1 桁のミリ秒レイテンシーと最大 数十 GBps のスループットを必要とするワークロードで使用できます。必要に応じて、HDD ストレージ容量の 20% のサイズに SSD 読み取りキャッシュをプロビジョニングできます。

詳細については、「FSx for Lustre ストレージクラス」を参照してください。

FSx for Lustre とデータリポジトリ

FSx for Lustre ファイルシステムを Simple Storage Service (Amazon S3) のデータリポジトリまたはオンプレミスのデータストアにリンクできます。

FSx for Lustre S3 データリポジトリ統合

FSx for Lustre は Amazon S3 と統合されているため、Lustre のハイパフォーマンスファイルシステムを使用してクラウドデータセットを簡単に処理できます。Simple Storage Service (Amazon S3) バケットにリンクすると、FSx for Lustre ファイルシステムは S3 オブジェクトをファイルとして透過

的に表示します。Amazon FSx は、ファイルシステムの作成時に S3 バケット内のすべての既存ファイルのリストをインポートします。Amazon FSx は、ファイルシステムの作成後にデータリポジトリに追加されたファイルのリストをインポートすることもできます。ワークフローのニーズに合わせてインポートプリファレンスを設定できます。ファイルシステムによって、ファイルシステムデータを S3 に書き戻すこともできます。データリポジトリタスクによって、FSx for Lustre ファイルシステムと Simple Storage Service (Amazon S3) 上の耐久性のあるデータリポジトリ間のデータとメタデータの転送が簡単に行えます。詳細については、「Amazon FSx for Lustre でデータリポジトリの使用」および「データリポジトリタスク」を参照してください。

FSx for Lustre およびオンプレミスのデータリポジトリ

Amazon FSx for Lustre では、 Direct Connect または を使用してデータをインポート AWS クラウドすることで、データ処理ワークロードをオンプレミスから にバーストできます Site-to-Site VPN。詳細については、「オンプレミスのデータに対する Amazon FSx の使用」を参照してください。

FSx for Lustre ファイルシステムへのアクセス

単一の FSx for Lustre ファイルシステムに接続されているコンピューティングインスタンスタイプと Linux Amazon マシンイメージ (AMI) を混在させて一致させることができます。

Amazon FSx for Lustre ファイルシステムには、Amazon Elastic Compute Cloud (Amazon EC2) インスタンス、Amazon Elastic Container Service (Amazon ECS) Docker コンテナ、および Amazon Elastic Kubernetes Service (Amazon EKS) で実行中のコンテナからアクセスできます。

- Amazon EC2 - オープンソースの Lustre クライアントを使用して、Amazon EC2 コンピューティングインスタンスからファイルシステムにアクセスします。Amazon EC2 インスタンスは、同じ Amazon Virtual Private Cloud (Amazon VPC) 内の他のアベイラビリティーゾーンからファイルシステムにアクセスできます。ただし、ネットワーク設定が VPC 内のサブネットを越えてアクセスできるように設定されている場合に限ります。Amazon FSx for Lustre がマウントされたら、ローカルファイルシステムと同じように、ファイルやディレクトリを操作できるようになります。
- Amazon EKS - Amazon EKS ユーザーガイド で説明されているように、オープンソースの FSx for Lustre CSI ドライバー を使用して、Amazon EKS で実行されているコンテナから、Amazon FSx for Lustre にアクセスします。Amazon EKS で実行されているコンテナは、Amazon FSx for Lustre によってバックアップされた高性能永続ボリューム (PV) を使用できます。
- Amazon ECS - Amazon EC2 インスタンス上の Amazon ECS Docker コンテナから、Amazon FSx for Lustre にアクセスします。詳細については、「Amazon Elastic Container Service からのマウント」を参照してください。

Amazon FSx for Lustre は、Amazon Linux 2023 および Amazon Linux 2、Red Hat Enterprise Linux (RHEL)、CentOS、Ubuntu、SUSE Linux など、最も人気の高い Linux ベースの AMI と互換性があります。Lustre クライアントは、Amazon Linux 2023 および Amazon Linux 2 に含まれています。RHEL、CentOS、Ubuntu の場合 AWS Lustre、クライアントリポジトリはこれらのオペレーティングシステムと互換性のあるクライアントを提供します。

FSx for Lustre を使用すると、 Direct Connect または 経由でデータをインポート AWS クラウド することで、コンピューティング負荷の高いワークロードをオンプレミスから にバーストできます AWS Virtual Private Network。オンプレミスから Amazon FSx ファイルシステムにアクセスし、必要に応じてデータをファイルシステムにコピーし、クラウド内のインスタンスでコンピューティング集約型のワークロードを実行できます。

FSx for Lustre ファイルシステムにアクセスできるクライアント、コンピューティングインスタンス、および環境の詳細については、「ファイルシステムへのアクセス」を参照してください。

AWS サービスとの統合

Amazon FSx for Lustre は、入力データソースとして Amazon SageMaker AI と統合されています。SageMaker AI を FSx for Lustre で使用する場合、Simple Storage Service (Amazon S3) からの最初のダウンロードステップを除外することで、機械学習トレーニングジョブが高速化されます。さらに、S3 リクエストのコストを節約することで、同じデータセットで反復ジョブの一般的なオブジェクトが繰り返しダウンロードされるのを防ぐことができるため、総保有コスト (TCO) を削減することができます。詳細については、「Amazon SageMaker AI デベロッパーガイド」の「What Is SageMaker AI?」を参照してください。Amazon FSx for Lustre を SageMaker AI のデータソースとして使用する方法のチュートリアルについては、AWS 機械学習ブログの「Speed up training on Amazon SageMaker AI using Amazon FSx for Lustre and Amazon EFS file systems」(Amazon FSx for Lustre および Amazon EFS ファイルシステムを使用して Amazon SageMaker でトレーニングを高速化する) を参照してください。

FSx for Lustre は EC2 起動テンプレート AWS Batch を使用して と統合します。 AWS Batch では、ハイパフォーマンスコンピューティング (HPC) AWS クラウド、機械学習 (ML)、その他の非同期ワークロードなど、バッチコンピューティングワークロードを実行できます。 は、ジョブリソースの要件に基づいてインスタンスを AWS Batch 自動的かつ動的にサイズ設定します。詳細については、「 AWS Batch ユーザーガイド」の「What is AWS Batch?」を参照してください。

FSx for Lustre AWS ParallelCluster は AWS ParallelCluster、HPC クラスターのデプロイと管理に使用される AWSがサポートするオープンソースクラスター管理ツールです。クラスター作成プロセス中に、FSx for Lustre ファイルシステムを自動的に作成したり、既存のファイルシステムを使用したりできます。

セキュリティとコンプライアンス

FSx for Lustre ファイルシステムでは、保管時と転送中の暗号化がサポートされています。Amazon FSx は、 AWS Key Management Service () で管理されるキーを使用して、保管中のファイルシステムデータを自動的に暗号化しますAWS KMS。サポートされている Amazon EC2 インスタンスからアクセスされる特定の AWS リージョン では、転送中のデータも、ファイルシステム上で自動的に暗号化されます。転送中のデータの暗号化がサポートされている AWS リージョン 場所など、FSx for Lustre のデータ暗号化の詳細については、「」を参照してくださいAmazon FSx for Lustre でのデータの暗号化。Amazon FSx は、ISO、PCI-DSS、および SOC の認定に準拠していると評価されており、HIPAA の対象となります。詳細については、「Amazon FSx for Lustre のセキュリティ」を参照してください。

前提

このガイドでは、以下の仮定を行います。

- Amazon Elastic Compute Cloud (Amazon EC2) を使用する場合は、そのサービスに慣れていることを前提としています。Amazon EC2 の使用方法の詳細については、「Amazon EC2 ドキュメント」を参照してください。
- Amazon Virtual Private Cloud (Amazon VPC) の使用に慣れていることを前提としています。Amazon VPC の使用方法の詳細については、「Amazon VPC ユーザーガイド」を参照してください。
- Amazon VPC サービスに基づいて、VPC のデフォルトのセキュリティグループのルールを変更していないことを前提としています。セキュリティグループのルールを変更している場合は、Amazon EC2 インスタンスから Amazon FSx for Lustre ファイルシステムへのネットワークトラフィックを許可するために必要なルールを必ず追加してください。詳細については、「Amazon VPC を使用したファイルシステムアクセスコントロール」を参照してください。

Amazon FSx for Lustre の料金

Amazon FSx for Lustre では、ハードウェアやソフトウェアの前払い費用は発生しません。最低コミットメント、セットアップコスト、追加料金なしで、使用したリソースに対してのみ、お支払いいただきます。サービスに関連した料金や費用については、「 Amazon FSx for Lustre の料金」を参照してください。

Amazon FSx for Lustre フォーラム

Amazon FSx for Lustre の使用中に問題が発生した場合は、フォーラム を確認してください。

Amazon FSx for Lustre を初めてお使いですか？

Amazon FSx for Lustre を初めて使用する方は、以下のセクションを順に読むことをお勧めします。

1. 最初の Amazon FSx for Lustre ファイルシステムを作成する準備ができたら、「Amazon FSx for Lustre の使用開始」をお試しください。
2. パフォーマンスの詳細については、「Amazon FSx for Lustre のパフォーマンス」を参照してください。
3. ファイルシステムを Simple Storage Service (Amazon S3) バケットデータリポジトリにリンクする方法については、「Amazon FSx for Lustre でデータリポジトリの使用」を参照してください。
4. Amazon FSx for Lustre セキュリティの詳細については、「Amazon FSx for Lustre のセキュリティ」を参照してください。
5. スループットやファイルシステムのサイズを含む Amazon FSx for Lustre のスケーラビリティ制限の詳細については、「Amazon FSx for Lustre の Service quotas」を参照してください。
6. Amazon FSx for Lustre API の詳細については、「Amazon FSx for Lustre API リファレンス」を参照してください。

Amazon FSx for Lustre を設定する

Amazon FSx for Lustre を初めて使用する前に、「Amazon Web Services へのサインアップ」セクションのタスクを完了してください。入門チュートリアルを完了するには、ファイルシステムにリンクする Amazon S3 バケットに、「Simple Storage Service (Amazon S3) でデータリポジトリを使用する許可を追加する」に一覧表示されているアクセス許可があることを確認してください。

トピック

Amazon Web Services へのサインアップ

を設定するには AWS、次のタスクを実行します。

1. にサインアップする AWS アカウント
2. 管理アクセスを持つユーザーを作成する

にサインアップする AWS アカウント

がない場合は AWS アカウント、次の手順を実行して作成します。

にサインアップするには AWS アカウント

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

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

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

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

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

管理アクセスを持つユーザーを作成する

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

を保護する AWS アカウントのルートユーザー

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

   ルートユーザーを使用してサインインする方法については、「AWS サインイン ユーザーガイド」の「ルートユーザーとしてサインインする」を参照してください。
2. ルートユーザーの多要素認証 (MFA) を有効にします。

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

管理アクセスを持つユーザーを作成する

1. IAM アイデンティティセンターを有効にします。

   手順については、「AWS IAM アイデンティティセンター ユーザーガイド」の「AWS IAM アイデンティティセンターの有効化」を参照してください。
2. IAM アイデンティティセンターで、ユーザーに管理アクセスを付与します。

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

管理アクセス権を持つユーザーとしてサインインする

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

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

追加のユーザーにアクセス権を割り当てる

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

   手順については、「AWS IAM アイデンティティセンター ユーザーガイド」の「アクセス許可セットを作成する」を参照してください。
2. グループにユーザーを割り当て、そのグループにシングルサインオンアクセス権を割り当てます。

   手順については、「AWS IAM アイデンティティセンター ユーザーガイド」の「グループを追加する」を参照してください。

Simple Storage Service (Amazon S3) でデータリポジトリを使用する許可を追加する

Amazon FSx for Lustre は Simple Storage Service (Amazon S3) と深く統合しています。この統合により、FSx for Lustre ファイルシステムにアクセスするアプリケーションは、リンクされた Simple Storage Service (Amazon S3) バケットに保存されているオブジェクトにもシームレスにアクセスできます。詳細については、「Amazon FSx for Lustre でデータリポジトリの使用」を参照してください。

データリポジトリを使用するには、まず、管理者ユーザーのアカウントに関連付けられたロールで、Amazon FSx for Lustre に特定の IAM アクセス許可を許可する必要があります。

コンソールを使用するロールのインラインポリシーを埋め込むには

1. にサインイン AWS マネジメントコンソール し、https://console.aws.amazon.com/iam/ で IAM コンソールを開きます。
2. ナビゲーションペインで Roles (ロール) を選択してください。

3. 一覧で、ポリシーを埋め込むロールの名前を選択します。
4. [Permissions] (アクセス許可) タブを選択します。
5. ページ下部までスクロールし、[Add inline policy] (インラインポリシーの追加) を選択します。

> **Note**
>
> IAM のサービスリンクロールにインラインポリシーを埋め込むことはできません。リンクされたサービスは、ロールの許可を変更できるかどうかを定義するため、サービスコンソール、API、または AWS CLIからポリシーを追加できる場合があります。サービスに関するサービスリンクロールのドキュメントを表示するには、「IAM と連携するAWS サービス」を参照し、対象サービスの [サービスリンクロール] 列で [はい] を選択します。

6. [Creating Policies with the Visual Editor] (ビジュアルエディタでポリシーを作成する) を選択します
7. 以下のアクセス許可ポリシーステートメントに追加します。

   JSON

```
{
    "Version":"2012-10-17",
    "Statement": {
        "Effect": "Allow",
        "Action": [
            "iam:CreateServiceLinkedRole",
            "iam:AttachRolePolicy",
            "iam:PutRolePolicy"
        ],
        "Resource": "arn:aws:iam::*:role/aws-service-role/s3.data-
source.lustre.fsx.amazonaws.com/*"
    }
}
```

インラインポリシーを作成した後は、自動的にロールに埋め込まれます。サービスにリンクされたロールの詳細については、「Amazon FSx のサービスリンクロールの使用」を参照してください。

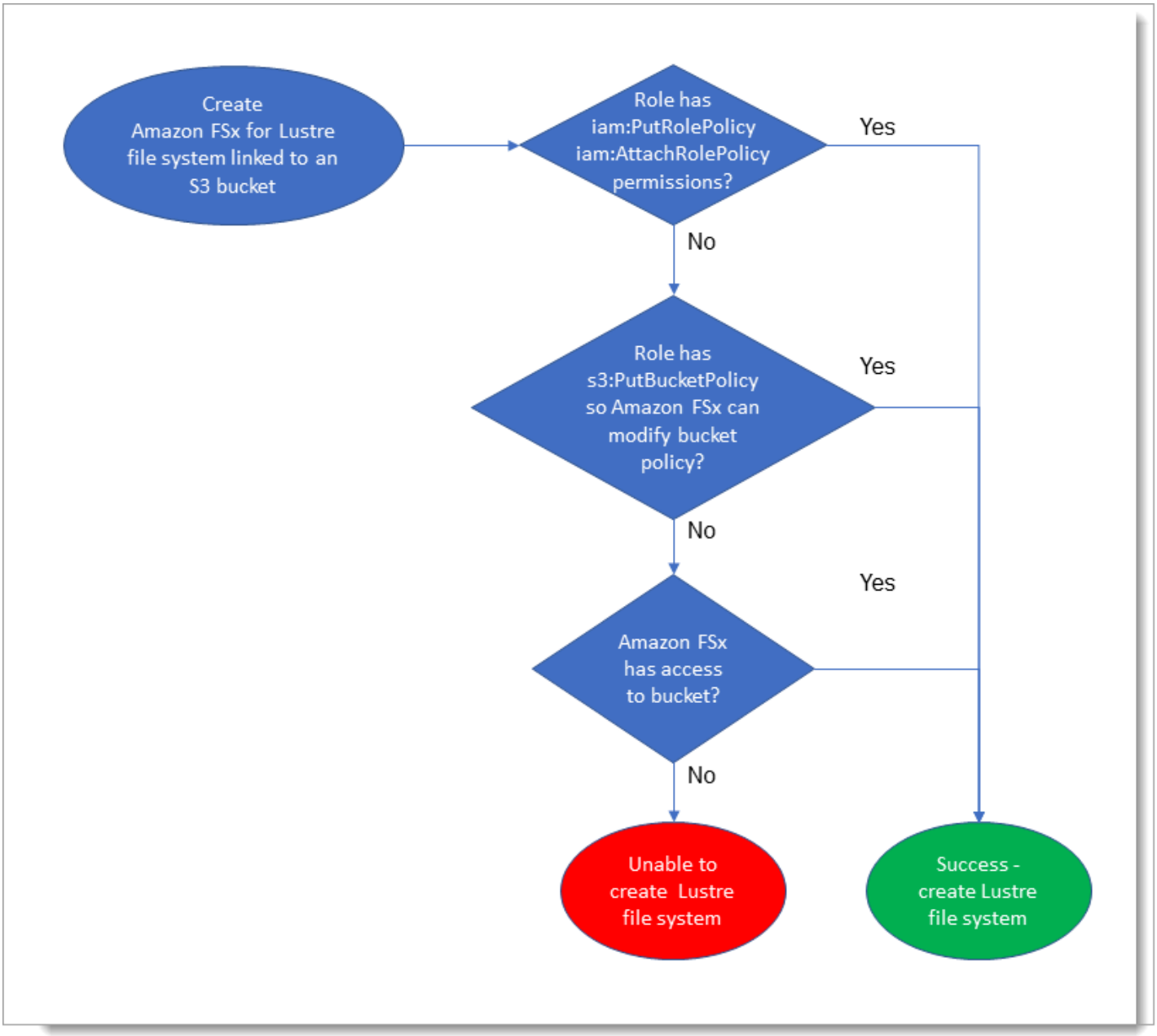
FSx for Lustre がリンクされた S3 バケットへのアクセスをチェックする方法

FSx for Lustre ファイルシステムの作成に使用する IAM ロールに `iam:AttachRolePolicy` および `iam:PutRolePolicy` 許可がない場合、Amazon FSx は S3 バケットポリシーを更新できるかどうかを確認します。Amazon FSx は、Amazon FSx ファイルシステムが S3 バケットへのデータのインポートまたはエクスポートを許可する `s3:PutBucketPolicy` アクセス許可が IAM ロールに含まれる場合に、バケットポリシーを更新できます。バケットポリシーの変更が許可されている場合、Amazon FSx はバケットポリシーに次のアクセス許可を追加します。

- `s3:AbortMultipartUpload`
- `s3:DeleteObject`
- `s3:PutObject`
- `s3:Get*`
- `s3:List*`
- `s3:PutBucketNotification`
- `s3:PutBucketPolicy`
- `s3:DeleteBucketPolicy`

Amazon FSx がバケットポリシーを変更できない場合、既存のバケットポリシーがバケットへの Amazon FSx アクセスを許可しているかどうかを確認します。

オプションがすべて失敗すると、ファイルシステムを作成するリクエストは失敗します。次の図表は、ファイルシステムがリンク先の S3 バケットにアクセスできるかどうかを判断するときに Amazon FSx が実行するチェックを示しています。



次のステップ

FSx for Lustre の使用を開始するには、「Amazon FSx for Lustre の使用開始」で Amazon FSx for Lustre リソースを作成するための手順を参照してください。

Amazon FSx for Lustre の使用開始

次に、Amazon FSx for Lustre の使用方法を学びます。ステップでは、Amazon FSx for Lustre ファイルシステムを作成し、コンピューティングインスタンスからアクセスするステップを説明します。オプションで、Amazon FSx for Lustre ファイルシステムを使用して、ファイルベースのアプリケーションで Simple Storage Service (Amazon S3) バケット内のデータを処理する方法を説明します。

この入門演習では、次のステップが含まれます。

トピック

前提条件

この入門演習を実行するには、次のものが必要です。

- Amazon FSx for Lustre ファイルシステムおよび Amazon EC2 インスタンスを作成するために必要なアクセス許可を持つ AWS アカウント。詳細については、「Amazon FSx for Lustre を設定する」を参照してください。
- FSx for Lustre ファイルシステムに関連付ける Amazon VPC セキュリティグループを作成します。これは、ファイルシステムの作成後は変更しないでください。詳細については、「Amazon FSx ファイルシステムのセキュリティグループを作成するには」を参照してください。
- Amazon VPC サービスに基づいて、仮想プライベートクラウド (VPC) でサポートされている Linux リリースを実行する Amazon EC2 インスタンス。この入門演習では、Amazon Linux 2023 を使用することをお勧めします。この EC2 インスタンスに Lustre クライアントをインストールし、EC2 インスタンスに FSx for Lustre ファイルシステムをマウントします。EC2 インスタンスの作成の詳細については、「Amazon EC2 ユーザーガイド」の「開始方法: インスタンスを起動する」または「インスタンスを起動する」を参照してください。

Amazon Linux 2023 に加えて、Lustre クライアントは Amazon Linux 2、Red Hat Enterprise Linux (RHEL)、CentOS、Rocky Linux、SUSE Linux Enterprise Server、Ubuntu オペレーティングシステムをサポートしています。詳細については、「Lustre ファイルシステムとクライアントカーネルの互換性」を参照してください。

- この入門演習用に Amazon EC2 インスタンスを作成するときは、次の点に注意してください。
  - デフォルトの VPC でインスタンスを作成することをお勧めします。
  - EC2 インスタンスを作成する場合は、デフォルトのセキュリティグループを使用することをお勧めします。
- スクラッチ または 永続的、どちらの Amazon FSx for Lustre ファイルシステムタイプを作成するか決定します 。詳細については、「FSx for Lustre ファイルシステムのデプロイおよびストレージクラスオプション」を参照してください。
- 各 FSx for Lustre ファイルシステムには、各メタデータサーバー (MDS) 用に 1 つの IP アドレスと、各ストレージサーバー (OSS) 用に 1 つの IP アドレスが必要です。詳細については、「ファイルシステムの IP アドレス」を参照してください。
- ワークロードが処理するデータを格納する Simple Storage Service (Amazon S3) バケット。S3 バケットは、FSx for Lustre ファイルシステムがリンクされた耐久性のあるデータリポジトリになります。

ステップ 1: FSx for Lustre ファイルシステムの作成

Amazon FSx コンソールでファイルシステムを作成します。Amazon FSx コンソールを使用して作成されるすべての FSx for Lustre ファイルシステムが、Lustre バージョン 2.15 で構築されるようになったことにご注意ください。

ファイルシステムを作成するには

1. https://console.aws.amazon.com/fsx/ で Amazon FSx コンソールを開きます。
2. ダッシュボードで [Create file system] (ファイルシステムの作成) を選択して、ファイルシステム作成ウィザードをスタートします。
3. FSx for Lustreを選択し、Next を選択して、Create File System ページを表示します。

   [File·system·details] (ファイルシステムの詳細) セクションで設定を開始します。
4. ファイルシステム名-オプション で、ファイルシステム名を入力します。最大 256 文字の Unicode 文字、空白、数字、特殊文字 + - = . _ : / を使用できます。

5. [デプロイとストレージクラス] で、いずれかのオプションを選択します:

   - 長期ストレージとレイテンシーの影響を受けやすいワークロードには、永続、SSD を選択します。SSD ストレージでは、プロビジョニングしたストレージの量に対して課金されます。

     必要に応じて、EFA が有効 を選択し、ファイルシステムの Elastic Fabric Adapter (EFA) サポートを有効にします。EFA の詳細については、「EFA 対応ファイルシステムの使用」を参照してください。

   - 長期ストレージには、[永続 インテリジェント階層化] を選択します。インテリジェント階層化ストレージクラスは、ほとんどのワークロードに適した完全に伸縮自在で費用対効果の高いストレージと、頻繁にアクセスされるデータの読み取りに SSD レイテンシーを提供するオプションの SSD 読み取りキャッシュを提供します。Intelligent-Tiering では、データセットのサイズに応じて保存したデータに対して課金され、ファイルシステムのサイズを指定する必要はありません。

     必要に応じて、EFA が有効 を選択し、ファイルシステムの Elastic Fabric Adapter (EFA) サポートを有効にします。

   - テンポラリストレージとデータの短期間の処理のために、[スクラッチ、SSD] デプロイを選択します。SSD ストレージでは、プロビジョニングしたストレージの量に対して課金されます。

6. ファイルシステムのスループットのマウントを選択します。プロビジョニングしたスループットに対して支払いが発生します。

   - 永続 SSD ストレージの場合は、ストレージの単位あたりのスループット 値を選択します。ストレージ単位あたりのスループットは、プロビジョニングされたストレージごとの読み取り、および書き込みスループットの量 (MB / TiB) です。
   - スクラッチ SSD ストレージの場合は、ストレージ単位あたりのスループットを選択します。
   - インテリジェント階層化ストレージの場合は、[スループットキャパシティ] の値を選択します。

7. [ストレージ容量] (SSD ストレージクラスのみ) については、ファイルシステムのストレージ容量を TB で設定します:

   - 永続、SSD デプロイタイプの場合、これを 1.2 TiB、2.4 TiB、または 2.4 TiB の増分の値に設定します。
   - EFA 対応、永続、SSD デプロイタイプの場合、この値を 1000、500、250、125 MBps/TiB のスループット階層ごとに 4.8 TiB、9.6 TiB、19.2 TiB、38.4 TiB の増分で設定します。

ファイルシステムを作成した後、必要に応じてストレージ容量を増やすことができます。詳細については、「ストレージ容量の管理」を参照してください。

8. メタデータ設定には、ファイルシステムのメタデータ IOPS の数をプロビジョニングするための以下のオプションから 1 つ選びます:

   - Amazon FSx for Lustre でファイルシステムのストレージ容量に基づいてファイルシステムのメタデータ IOPS を自動的にプロビジョニングおよびスケーリングする場合は、[自動] (SSD ストレージクラスのみ) を選択します。
   - メタデータ IOPS の数を指定してファイルシステムを SSD またはインテリジェント階層化でプロビジョニングする場合は、[ユーザープロビジョニング] を選択します。有効な値は次のとおりです。
     - SSD ファイルシステムでは、有効な値は、`1500`、`3000`、`6000`、`12000`、および `192000` までの `12000` の倍数です。
     - インテリジェント階層化ファイルシステムの場合、有効な値は `6000` と `12000` です。

   メタデータ IOPS の詳細については、「Lustre メタデータパフォーマンス設定」を参照してください。

9. [SSD 読み取りキャッシュ] (インテリジェント階層化のみ) の場合は、[自動 (スループットキャパシティに比例)] または [カスタム (ユーザーによるプロビジョニング)] のいずれかを選択します。自動オプションを使用すると、Amazon FSx for Lustre はプロビジョニングされたスループットに基づいて読み取りキャッシュサイズを自動的に選択します。アクティブな作業データセットのおおよそのサイズがわかっている場合は、カスタムを選択して SSD 読み取りキャッシュのサイズをカスタマイズできます。詳細については、「プロビジョニングされた SSD 読み取りキャッシュの管理」を参照してください。

10. データ圧縮タイプ で、[NONE] (なし) を選択してデータ圧縮をオフにするか、LZ4 を選択して LZ4 アルゴリズムでデータ圧縮をオンにします。詳細については、「Lustre データ圧縮」を参照してください。

11. [Network & security] (ネットワークとセキュリティ) セクションで、次のネットワークおよびセキュリティグループ情報を入力します。

    - [Virtual Private Cloud (VPC)] (仮想プライベートクラウド (VPC)) で、ファイルシステムに関連付ける VPC を選択します。この入門演習では、Amazon EC2 インスタンスと同じ VPC を選択します。

- VPC セキュリティグループ の場合は、VPC のデフォルトのセキュリティグループの ID がすでに追加されている必要があります。

  デフォルトのセキュリティグループを使用していない場合は、この入門演習で使用するセキュリティグループに次のインバウンドルールが追加されていることを確認してください。

タイプ	プロトコル	ポート範囲	ソース	説明
すべての TCP	TCP	0〜65535	カスタム *the_ID_of_this_security_group*	インバウンド Lustre トラフィックルール

> **Important**
>
> - 使用しているセキュリティグループが、「Amazon VPC を使用したファイルシステムアクセスコントロール」に記載の設定手順に従っていることを確認します。セキュリティグループを設定して、ポート 988 および 1018 ~ 1023 で、セキュリティグループ自体またはフルサブネット CIDR からのインバウンドトラフィックを許可する必要があります。これは、ファイルシステムホストが相互に通信できるようにするために必要です。
> - EFA 対応のファイルシステムを作成する場合は、必ず EFA 対応のセキュリティグループ を指定してください。

- [Subnet] (サブネット) に関して、使用可能なサブネットのリストから任意の値を選択します。

12. [Encryption] (暗号化) セクションで使用できるオプションは、作成するファイルシステムの種類によって異なります。

    - 永続ファイルシステムの場合、AWS Key Management Service (AWS KMS) 暗号化キーを選択して、保存中のファイルシステム上のデータを暗号化できます。
    - スクラッチファイルシステムの場合、保管中のデータは AWS によって管理されるキーを使用して暗号化されます。

- スクラッチ 2 および永続的ファイルシステムでは、サポートされている Amazon EC2 インスタンスタイプからファイルシステムにアクセスすると、転送中のデータが自動的に暗号化されます。詳細については、「転送中のデータの暗号化」を参照してください。

13. データリポジトリの Import/Export (オプション)] のセクションでは、ファイルシステムを Simple Storage Service (Amazon S3) データリポジトリにリンクすることはデフォルトで無効になっています。このオプションを有効にして、既存の S3 バケットへのデータリポジトリアソシエーションを作成する方法については、「ファイルシステムの作成中に S3 バケットをリンクするには (コンソール)」を参照してください。

> ⚠ Important
>
> - このオプションを選択すると、バックアップが無効になり、ファイルシステムの作成中にバックアップを有効にできなくなります。
> - 1 つ以上の Amazon FSx for Lustre ファイルシステムを Simple Storage Service (Amazon S3) バケットにリンクする場合は、リンクされているすべてのファイルシステムが削除されるまで Simple Storage Service (Amazon S3) バケットを削除しないでください。
> - インテリジェント階層化ファイルシステムは、Amazon S3 データリポジトリへのリンクをサポートしていません。

14. [ログ記録 (オプション)] では、デフォルトでログ記録が有効化されています。有効にすると、ファイルシステムのデータリポジトリアクティビティの障害と警告が Amazon CloudWatch Logs にログ記録されます。ログの設定の詳細については、「ロギングを管理する」を参照してください。

15. [バックアップとメンテナンス (オプション)] では、以下を実行できます。

    - 毎日の自動バックアップ を無効にします。このオプションは、[データリポジトリの Import/Export] を有効にしていない限り、デフォルトで有効になっています。
    - 毎日の自動バックアップウィンドウ の開始時刻を設定します。
    - 自動バックアップ保持期間 を 1～35 日に設定します。
    - 毎週のメンテナンス期間 のスタート時刻を設定するか、デフォルトの [No preference] (設定なし) に設定したままにします。

    詳細については、「バックアップでデータを保護する。」および「Amazon FSx for Lustre メンテナンスウィンドウ」を参照してください。

16. [ルートスカッシュ (オプション)] では、デフォルトでルートスカッシュが無効化されています。ルートスカッシュの有効化と設定の詳細については、「ファイルシステムの作成時にルートスカッシュを有効にするには (コンソール)」を参照してください。
17. ファイルシステムに適用するタグを作成します。
18. [Next] (次へ) を選択して、ファイルシステムの概要を作成する ページを表示します。
19. Amazon FSx for Lustreファイルシステムの設定を確認し、[Create file system] (ファイルシステムの作成) を選択します。

ファイルシステムが作成されたので、後のステップのために完全修飾ドメイン名とマウント名をメモします。ファイルシステムの完全修飾ドメイン名とマウント名を見つけるには、[ファイルシステム] のダッシュボードでファイルシステム名を選択し、添付 を選択します。

ステップ 2: Lustre クライアントをインストールおよび設定する

Amazon EC2 インスタンスから Amazon FSx for Lustre ファイルシステムにアクセスする前に、以下を実行する必要があります:

- EC2 インスタンスが最小カーネル要件を満たしていることを確認します。
- 必要に応じてカーネルを更新します。
- Lustre クライアントのダウンロードとインストール

カーネルバージョンを確認して Lustre クライアントをダウンロードするには

1. EC2 インスタンスでターミナルウィンドウを開きます。
2. 次のコマンドを実行して、コンピューティングインスタンスで現在実行されているカーネルを特定します。

   ```
   uname -r
   ```

3. 次のいずれかを実行します。

   - x86 ベースの EC2 インスタンスにコマンドが `6.1.79-99.167.amzn2023.x86_64` を返した場合、または Graviton2 ベースの EC2 インスタンスに `6.1.79-99.167.amzn2023.aarch64` またはそれ以上を返した場合は、次のコマンドを使用して Lustre クライアントをダウンロードしてインストールします。

```
sudo dnf install -y lustre-client
```

- コマンドが x86 ベースの EC2 インスタンスの場合は `6.1.79-99.167.amzn2023.x86_64` 未満、Graviton2 ベースの EC2 インスタンスの場合は `6.1.79-99.167.amzn2023.aarch64` 未満の結果を返す場合は、次のコマンドを実行してカーネルを更新し、Amazon EC2 インスタンスを再起動します。

  ```
  sudo dnf -y update kernel && sudo reboot
  ```

  uname -r コマンドを使用して、カーネルが更新されていることを確認します。次に、上記の説明に従って Lustre クライアントをダウンロードしてインストールします。

他の Linux ディストリビューションに Lustre クライアントをインストールする方法については、「Lustre クライアントのインストール」を参照してください。

ステップ 3: ファイルシステムをマウントする

ファイルシステムをマウントするには、マウントディレクトリまたはマウントポイントを作成し、ファイルシステムをクライアントにマウントし、クライアントがファイルシステムにアクセスできることを確認します。

ファイルシステムをマウントするには

1. 次のコマンドを使用して、マウントポイントのディレクトリを作成します。

   ```
   sudo mkdir -p /mnt/fsx
   ```

2. 作成したディレクトリに Amazon FSx for Lustre ファイルシステムをマウントします。次のコマンドを使用して、次のアイテムを置き換えます。

   - 実際のファイルシステムのドメインネームシステム (DNS) 名で *`file_system_dns_name`* を置き換えます。
   - *`mountname`* を、describe-file-systems AWS CLI コマンドまたは DescribeFileSystems API オペレーションを実行して取得できるファイルシステムのマウント名に置き換えます。

```
sudo mount -t lustre -o relatime,flock file_system_dns_name@tcp:/mountname /mnt/fsx
```

このコマンドは、-o relatime と flock の 2 つのオプションでファイルシステムをマウントします。

- relatime — atime オプションでは、ファイルがアクセスされるたびに atime (inode アクセス時間) のデータが保持されるのに対し、relatime オプションでも atime のデータが保持されますが、ファイルがアクセスされるたびに保持されるわけではありません。relatime オプションを有効にすると、atime のデータが最後に更新されてからファイルが変更された場合 (mtime)、またはファイルが一定時間以上 (デフォルトでは 6 時間) 前に最後にアクセスされた場合にのみ、atime のデータがディスクに書き込まれます。relatime または atime のオプションを使用すると、ファイルのリリース プロセスが最適化されます。

  > **Note**
  >
  > ワークロードに正確なアクセス時間の精度が必要な場合は、atime マウントオプションを使用してマウントできます。ただし、これを行うと、正確なアクセス時間値を維持するために必要なネットワークトラフィックが増加し、ワークロードのパフォーマンスに影響する可能性があります。
  > ワークロードにメタデータのアクセス時間が必要ない場合は、noatime マウントオプションを使用してアクセス時間の更新を無効にすると、パフォーマンスが向上する可能性があります。ファイルのリリースやデータの有効性のリリースなど、atime に焦点を絞ったプロセスでは、リリース時に不正確さが生じることに注意してください。

- flock - ファイルシステムのファイルロックを有効にします。ファイルロックを有効にしたくない場合は、flock なしで mount コマンドを使用します。

3. 次のコマンドを使用して、ファイルシステム /mnt/fsx をマウントしたディレクトリの内容を一覧表示し、マウントコマンドが成功したことを確認します。

```
ls /mnt/fsx
import-path  lustre
$
```

以下の df コマンドを使用することもできます。

```
df
Filesystem                      1K-blocks    Used  Available Use% Mounted on
devtmpf                           1001808       0    1001808   0% /dev
tmpfs                             1019760       0    1019760   0% /dev/shm
tmpfs                             1019760     392    1019368   1% /run
tmpfs                             1019760       0    1019760   0% /sys/fs/cgroup
/dev/xvda1                        8376300 1263180    7113120  16% /
123.456.789.0@tcp:/mountname  3547698816   13824 3547678848   1% /mnt/fsx
tmpfs                              203956       0     203956   0% /run/user/1000
```

結果は、/mnt/fsx にマウントされている Amazon FSx ファイルシステムを示しています。

ステップ 4: ワークフローを実行する

ファイルシステムが作成され、コンピューティングインスタンスにマウントされたので、それを使用して高パフォーマンスのコンピューティングワークロードを実行できます。

データリポジトリの関連付けを作成して、ファイルシステムを Simple Storage Service (Amazon S3) データリポジトリにリンクできます。詳細については、「Amazon S3 バケットにファイルシステムにリンクする」を参照してください。

ファイルシステムを Simple Storage Service (Amazon S3) データリポジトリにリンクしたら、ファイルシステムに書き込んだデータを Simple Storage Service (Amazon S3) バケットにいつでもエクスポートできます。コンピューティングインスタンスのいずれかのターミナルから、次のコマンドを実行して Simple Storage Service (Amazon S3) バケットにファイルをエクスポートします。

```
sudo lfs hsm_archive file_name
```

フォルダまたはファイルの大規模なコレクションでこのコマンドをすばやく実行する方法の詳細については、「HSM コマンドを使用したファイルのエクスポート」を参照してください。

ステップ 5: リソースをクリーンアップする

この演習が完了したら、以下のステップに従ってリソースをクリーンアップし、AWS アカウントを保護します。

リソースをクリーンアップするには

1. 最終的なエクスポートを行うには、次のコマンドを実行します。

```
nohup find /mnt/fsx -type f -print0 | xargs -0 -n 1 sudo lfs hsm_archive &
```

2. Amazon EC2 コンソールで、インスタンスを終了します。詳細については、「Amazon EC2 ユーザーガイド」の「インスタンスの終了」を参照してください。
3. Amazon FSx for Lustre コンソールで、次の手順でファイルシステムを削除します。
   a. ナビゲーションペインで、[File systems] (ファイルシステム) を選択します。
   b. ダッシュボードのファイルシステムのリストから削除するファイルシステムを選択します。
   c. [Actions] (アクション) で、[Delete file system] (ファイルシステムの削除) を選択します。
   d. 表示されるダイアログボックスで、ファイルシステムの最終バックアップを作成するかどうかを選択します。次に、削除を確定するために、ファイルシステム ID を入力します。[Delete file system] (ファイルシステムの削除) を選択します。
4. この演習用に Simple Storage Service (Amazon S3) バケットを作成して、エクスポートしたデータを保持したくない場合は、これで削除できます。詳細については、「Amazon Simple Storage Service ユーザーガイド」の「バケットの削除」を参照してください。

FSx for Lustre ファイルシステムのデプロイおよびストレージクラスオプション

Amazon FSx for Lustre には、2 つのファイルシステムデプロイオプション (永続 と スクラッチ) の 2 種類のファイルシステムデプロイオプションを提供します。SSD (ソリッドステートドライブ)、Intelligent-Tiering、HDD (ハードディスクドライブ) の 3 つのストレージクラスを提供します。

ファイルシステムのデプロイタイプとストレージクラスは、、 AWS マネジメントコンソール AWS Command Line Interface (AWS CLI)、または Amazon FSx for Lustre API を使用して新しいファイルシステムを作成するときに選択します。詳細については、「Amazon FSx API リファレンス」の「ステップ 1: FSx for Lustre ファイルシステムの作成」、および「CreateFileSystem」を参照してください。

永続的ファイルシステム

永続的ファイルシステム は、長期ストレージとワークロード用に設計されています。SSD および HDD ベースのファイルシステムについては、データはファイルシステムが存在するアベイラビリティーゾーン内で自動的にレプリケーションされます。インテリジェント階層化ファイルシステムでは、データは複数の利用可能領域に複製されます。ファイルサーバに添付されているデータボリュームは、添付先のファイルサーバとは別にレプリケーションされます。

Amazon FSx は、ハードウェア障害について永続的ファイルシステムを継続的にモニタリングし、障害発生時にインフラストラクチャのコンポーネントを自動的に置き換えます。永続的ファイルシステムでは、ファイルサーバが使用できなくなると、障害が発生してから数分以内にファイルサーバが自動的に置き換えられます。その間、クライアントはそのサーバー上のデータに対するリクエストを透過的に再試行し、最終的にファイルサーバーを交換した後に成功します。永続的ファイルシステム上のデータはディスク上にレプリケートされ、障害が発生したディスクはすべて自動的に透過的に置き換えられます。

長期ストレージや、長期間または無期限に実行され、可用性の中断の影響を受けやすいスループット重視のワークロードには、永続的ファイルシステムを使用します。

永続的デプロイタイプは、転送中の暗号化をサポートする Amazon EC2 インスタンスからアクセスされると、転送中のデータを自動的に暗号化します。

Amazon FSx for Lustre では、永続 1 と 永続 2 という 2 つの永続的なデプロイタイプがサポートされています。

Persistent 2 デプロイタイプ

永続 2 は最新世代の Persistent デプロイタイプで、長期ストレージを必要とし、最高レベルの IOPS とスループットを必要とするユースケースに最適です。永続 2 ファイルシステムは、SSD および インテリジェント階層化ストレージクラスをサポートします。

Amazon FSx コンソール、および Amazon FSx API を使用して AWS Command Line Interface、メタデータ設定と EFA を有効にした永続 2 ファイルシステムを作成できます。

Persistent 1 デプロイタイプ

永続型 1 デプロイタイプは、長期ストレージを必要とするユースケースに適しています。永続 1 デプロイタイプは SSD (ソリッドステートドライブ) と HDD (ハードディスクドライブ) のストレージクラスをサポートします。

永続 1 デプロイタイプは、 AWS CLI と Amazon FSx API を使用してのみ作成できます。

スクラッチファイルシステム

スクラッチファイルシステム は、データのテンポラリストレージと短期間の処理のために設計されています。データはレプリケーションされず、ファイルサーバーに障害が発生しても永続しません。スクラッチファイルシステムでは、ストレージ容量 TiB あたり 200 MBps のベースラインスループットの最大 6 倍の高バーストスループットを提供します。詳細については、「SSD および HDD ストレージクラスのパフォーマンス特性」を参照してください。

短期的で処理負荷の高いワークロードにコスト最適化されたストレージが必要な場合は、スクラッチファイルシステムを使用します。

スクラッチファイルシステムでは、ファイルサーバーが失敗し、データがレプリケーションされない場合、ファイルサーバーは置き換えられません。スクラッチファイルシステム上でファイルサーバまたはストレージディスクが使用できなくなった場合でも、他のサーバに保存されているファイルには引き続きアクセスできます。クライアントが使用できないサーバーまたはディスク上のデータにアクセスしようとすると、クライアントは即座に I/O エラーを発生します。

次の表に、サンプルサイズのスクラッチファイルシステムに想定される 1 日および 1 週間の可用性と耐久性を示します。大規模なファイルシステムでは、ファイルサーバとディスクが多くなるため、障害が発生する可能性が高くなります。

ファイルシステムサイズ (TiB)	ファイルサーバーの数	1 日の可用性 / 耐久性	1 週間の可用性 / 耐久性
1.2	2	99.9%	99.4%
2.4	2	99.9%	99.4%
4.8	3	99.8%	99.2%
9.6	5	99.8%	98.6%
50.4	22	99.1%	93.9%

ファイルシステムの IP アドレス

各 FSx for Lustre ファイルシステムには、各メタデータサーバー (MDS) 用に 1 つの IP アドレスと、各ストレージサーバー (OSS) 用に 1 つの IP アドレスが必要です。

SSD または HDD ストレージクラスを使用するファイルシステム

ファイルシステムタイプ	スループット、MBps/TiB	OSS あたりのストレージ
永続型 2 EFA*	125	OSS あたり 38.4 TiB
	250	OSS あたり 19.2 TiB
	500	OSS あたり 9.6 TiB
	1,000	OSS あたり 4.8 TiB
永続 2 非 EFA*	125、250、500、1000	OSS あたり 2.4 TiB
永続型 1 SSD	50、100、200	OSS あたり 2.4 TiB
永続 HDD	12	OSS あたり 6 TiB

ファイルシステムタイプ	スループット、MBps/TiB	OSS あたりのストレージ
	40	OSS あたり 1.8 TiB
スクラッチ 2	200	OSS あたり 2.4 TiB
Scratch 1	200	OSS あたり 3.6 TiB

インテリジェント階層化ストレージクラスを使用するファイルシステム

ファイルシステムタイプ	OSS あたりのスループット
Intelligent-Tiering*	OSS あたり 4000 MBps

> **Note**
>
> * Amazon FSx は、永続型 2 SSD およびメタデータ設定で設定された Intelligent-Tiering ファイルシステム上で、12,000 メタデータ IOPS ごとにメタデータサーバーをプロビジョニングします。
> Amazon FSx for Lustre Intelligent-Tiering ファイルシステムは、OSS あたり最大 512 TiB のストレージをサポートします。

FSx for Lustre ストレージクラス

Amazon FSx for Lustre は、さまざまなデータ処理要件に最適化されたソリッドステートドライブ (SSD)、Intelligent-Tiering、およびハードディスクドライブ (HDD) ストレージクラスを提供します:

- SSD ストレージクラスは、データセット全体への低レイテンシー (ミリ秒未満) アクセスを提供します。SSD ストレージクラスはプロビジョニングされます。つまり、ファイルシステムサイズを指定し、プロビジョニングされたストレージ量に対してストレージコストを支払います。SSD ス

トレージクラスは、すべてのデータにわたってオールフラッシュストレージのパフォーマンスを必要とするレイテンシーの影響を受けやすいワークロードに使用します。

SSD ストレージを備えた永続 2 ファイルシステムは、永続 1 ファイルシステムと比較して、ストレージ単位あたりのスループット (つまり、TiB あたり 250、500、または 1000 MBps) が高くなります。SSD ストレージを備えた 永続 1 ファイルシステムの場合、ストレージ単位あたりのスループットは、TiB あたり 50、100、または 200 MBps のいずれかになります。SSD ストレージを搭載した Scratch ファイルシステムの場合、ストレージ単位あたりのスループットは TiB あたり 200 MBps です。

- インテリジェント階層化ストレージクラスは、柔軟性が高く、アクセスパターンに応じて自動的に階層化されるストレージを提供します。柔軟性とは、保存するデータ量に応じて料金を支払う方式であり、ファイルシステムのサイズを事前に指定する必要がないことを意味します。インテリジェント階層化とは、最近アクセスしていないデータを保存するために自動的に支払う料金を低減することを意味します。このストレージクラスは、コールドデータを低コストのストレージ階層に階層化することで、コストを自動的に最適化します。頻繁にアクセスされるデータに対して、低レイテンシー(サブミリ秒)でアクセスするためのオプションの SSD 読み取りキャッシュをプロビジョニングできます。インテリジェント階層化ストレージクラスは、ほとんどのワークロードにおいて、価格とパフォーマンスの最適なバランスを提供します。キャッシュフレンドリで、すべてのデータにわたるオールフラッシュストレージのパフォーマンスを必要としないワークロードには、Intelligent-Tiering ストレージクラスを使用します。インテリジェント階層化ファイルシステムは、スループットキャパシティを 4,000 MBps 単位でサポートします。
- HDD ストレージクラスは、すべてのデータにわたって一貫した 1 桁の ミリ秒 レイテンシーを必要とするワークロードで使用できます。HDD ストレージ容量の 20% に相当するオプションの SSD リードキャッシュをプロビジョニングすることで、頻繁にアクセスされるデータへ低レイテンシーなアクセスが可能になります。HDD ストレージでは、ファイルシステムのサイズを指定し、プロビジョニングしたストレージの量に対して支払います。SSD ストレージを備えた 永続 1 ファイルシステムの場合、ストレージ単位あたりのスループットは、TiB あたり 12 または 40 MBps のいずれかになります。

ストレージクラスのパフォーマンスについては、「SSD および HDD ストレージクラスのパフォーマンス特性」および「インテリジェント階層化 ストレージクラスのパフォーマンス特性」を参照してください。

インテリジェント階層化ストレージクラスがデータを階層化する方法

Amazon FSx Intelligent-Tiering ストレージクラスは、3 つのアクセス階層に自動的にデータを保存します。これは、パフォーマンスへの影響や運用のオーバーヘッドなしに、データを最も費用対効果の高いアクセス階層に自動的に移動することにより、ストレージコストを最適化できるように設計されています。インテリジェント階層化ストレージクラスは、最終アクセス時間に基づいてデータを自動的に階層化し、アクティブでないデータのコストを自動的に最適化します:

- 過去 30 日間にアクセスされたデータは、高頻度アクセス階層に保存されます。
- 連続 30 日間アクセスされなかったデータは、低頻度アクセス階層に自動的に移行され、高頻度アクセス階層のデータよりもコストが低くなります。
- 連続 90 日間アクセスされなかったデータは、自動的にアーカイブインスタントアクセス階層に移動し、低頻度アクセス階層のデータよりもコストが低くなります。

低頻度アクセス階層またはアーカイブインスタントアクセス階層のデータにアクセスすると、データは自動的に高頻度アクセス階層に戻ります。さらに、スループットキャパシティの変更 (OSTs)、ファイルやディレクトリの再ストライピング、「lfs migrate」の使用などのオペレーションでは、一部のデータが高頻度アクセス階層に戻る可能性があります。

キャッシュされていないデータへのアクセスはすべて、データの階層に関係なく同じパフォーマンス特性を持ち、通常の読み取り/書き込みオペレーションコストを超える追加の IOPS、取得、移行コストはありません。

デプロイタイプの可用性

スクラッチ 2、永続 1、永続 2 のデプロイタイプを以下に示します AWS リージョン。

AWS リージョン	永続 2	永続 1	スクラッチ 2
米国東部(オハイオ)	✓	✓	✓
米国東部 (バージニア北部)	✓	✓	✓
米国東部 (アトランタ) ローカルゾーン	✓ *		

AWS リージョン	永続 2	永続 1	スクラッチ 2
米国東部 (ダラス) ローカルゾーン	✓ *		
米国西部 (北カリフォルニア)	✓	✓	✓
米国西部 (ロサンゼルス) ローカルゾーン		✓	✓
米国西部 (オレゴン)	✓	✓	✓
米国西部 (フェニックス) ローカルゾーン	✓ *		
アフリカ (ケープタウン)		✓	✓
アジアパシフィック (香港)	✓	✓	✓
アジアパシフィック (ハイデラバード)		✓	✓
アジアパシフィック (ジャカルタ)		✓	✓
アジアパシフィック (マレーシア)	✓ *		
アジアパシフィック (メルボルン)		✓	✓
アジアパシフィック (ムンバイ)	✓	✓	✓
アジアパシフィック (大阪)		✓	✓
アジアパシフィック (ソウル)	✓	✓	✓
アジアパシフィック (シンガポール)	✓	✓	✓
アジアパシフィック (シドニー)	✓	✓	✓
アジアパシフィック (台北)	✓ *		
アジアパシフィック (タイ)	✓ *		
アジアパシフィック (東京)	✓	✓	✓

AWS リージョン	永続 2	永続 1	スクラッチ 2
カナダ (中部)	✓	✓	✓
カナダ西部 (カルガリー)	✓ *		
欧州 (フランクフルト)	✓	✓	✓
欧州 (アイルランド)	✓	✓	✓
欧州 (ロンドン)	✓	✓	✓
欧州 (ミラノ)		✓	✓
欧州 (パリ)		✓	✓
欧州 (スペイン)		✓	✓
欧州 (ストックホルム)	✓	✓	✓
欧州 (チューリッヒ)		✓	✓
イスラエル (テルアビブ)	✓ *		✓
メキシコ (中部)	✓ *		
中東 (バーレーン)		✓	✓
中東 (アラブ首長国連邦)		✓	✓
南米 (サンパウロ)		✓	✓
AWS GovCloud (米国東部)		✓	✓
AWS GovCloud (米国西部)		✓	✓

Note

* EFA を使用しない SSD ストレージクラスを備えた Persistent-125 および Persistent-250 ファイルシステム AWS リージョン をサポートしています。

Amazon FSx for Lustre でデータリポジトリの使用

Amazon FSx for Lustre は、高速ワークロード処理用に最適化された高性能ファイルシステムを提供します。これは、機械学習、ハイパフォーマンスコンピューティング (HPC)、ビデオ処理、財務モデリング、Electronic Design Automation (EDA) などのワークロードをサポートすることができます。通常、ワークロードでは、データアクセスのために高速でスケーラブルなファイルシステムインターフェイスを介してデータを表示する必要があります。多くの場合、これらのワークロードに使用されるデータセットは Amazon S3 の長期データリポジトリに保存されます。FSx for Lustre は Amazon S3 などのデータリポジトリとネイティブに統合されており、データセットを Lustre ファイルシステムで簡単に処理できます。

> **Note**
>
> - ファイルシステムのバックアップは、Amazon S3 データリポジトリにリンクされているファイルシステムではサポートされません。詳細については、「バックアップでデータを保護する。」を参照してください。
> - インテリジェント階層化ファイルシステムは、Amazon S3 データリポジトリへのリンクをサポートしていません。

トピック

データリポジトリの概要

Amazon FSx for Lustre をデータリポジトリで使用する場合、自動インポートおよびデータリポジトリのインポートタスクを使用して、高パフォーマンスのファイルシステムで大量のファイルデータを取り込み、処理できます。同時に、データリポジトリの自動エクスポートまたはエクスポートタスクを使用して、結果をデータリポジトリに書き込むことができます。この方法を使用することで、データリポジトリに保存されている最新のデータを使用して、いつでもワークロードを再起動できます。

> **Note**
>
> データリポジトリの関連付け、自動エクスポート、複数のリポジトリのサポートは、FSx for Lustre 2.10 ファイルシステムと `Scratch 1` ファイルシステムでは使用できません。

FSx for Lustre は Amazon S3 と緊密に統合されています。この統合により、Amazon FSx ファイルシステムをマウントするアプリケーションから Amazon S3 バケットに保存されているオブジェクトにシームレスにアクセスできます。また、 AWS クラウド の Amazon EC2 インスタンスでコンピューティング集約型のワークロードを実行し、ワークロードの完了後に結果をデータリポジトリにエクスポートすることもできます。

Amazon S3 データリポジトリ内のオブジェクトにファイルシステム上のファイルおよびディレクトリとしてアクセスするには、ファイルおよびディレクトリのメタデータをファイルシステムにロードする必要があります。データリポジトリの関連付けを作成するときに、リンクされたデータリポジトリからメタデータをロードできます。

また、自動インポートまたはデータリポジトリのインポートタスクを使用して、リンクされたデータリポジトリからファイルシステムにファイルおよびディレクトリのメタデータをインポートすることもできます。データリポジトリの関連付けで自動インポートを有効にすると、S3 データリポジトリでファイルが作成、変更、または削除されたときに、ファイルシステムによってファイルのメタデータが自動的にインポートされます。または、データリポジトリのインポートタスクを使用して、新しいファイルまたは変更されたファイルとディレクトリのメタデータをインポートすることもできます。

> **Note**
>
> データリポジトリの自動インポートおよびインポートタスクは、ファイルシステム上で同時に使用できます。

また、自動エクスポートまたはデータリポジトリのエクスポートタスクを使用して、ファイルおよびそれに関連するメタデータをデータリポジトリにエクスポートすることもできます。データリポジトリの関連付けで自動エクスポートを有効にすると、ファイルデータおよびメタデータが作成、変更、または削除されたときに、ファイルシステムによってファイルデータとメタデータが自動的にエクスポートされます。また、データリポジトリのエクスポートタスクを使用して、ファイルまたはディレクトリをエクスポートすることもできます。データリポジトリのエクスポートタスクを使用すると、そのような最後のタスク以降に作成または変更されたファイルデータとメタデータがエクスポートされます。

> **Note**
>
> - 自動エクスポートおよびエクスポートデータリポジトリタスクは、ファイルシステム上で同時に使用することはできません。
> - データリポジトリの関連付けは、通常のファイル、シンボリックリンク、ディレクトリのみをエクスポートします。つまり、その他の種類のファイル (FIFO スペシャル、ブロックスペシャル、キャラクタスペシャル、ソケット) はすべて、自動エクスポートやデータリポジトリタスクのエクスポートといったエクスポートプロセスの一部としてエクスポートされません。

FSx for Lustre は、 Direct Connect または VPN を使用してオンプレミスクライアントからデータをコピーできるようにすることで、オンプレミスファイルシステムによるクラウドバーストワークロードもサポートします。

> **Important**
>
> 1 つ以上の Amazon FSx ファイルシステムを Amazon S3 のデータリポジトリにリンクしている場合は、リンクされているすべてのファイルシステムが削除またはリンク解除されるまで、Amazon S3 バケットを削除しないでください。

リンクされた S3 バケットのリージョンとアカウントのサポート

S3 バケットへのリンクを作成するときは、次のリージョンとアカウントのサポートの制限に注意してください。

- 自動エクスポートは、クロスリージョン設定をサポートします。Amazon FSx ファイルシステムとリンクされた S3 バケットは、同じ AWS リージョン または異なる に配置できます AWS リージョン。
- 自動インポートは、クロスリージョン設定をサポートしません。Amazon FSx ファイルシステムとリンクされた S3 バケットの両方が同じ AWS リージョンに配置されている必要があります。
- 自動エクスポートと自動インポートの両方で、クロスアカウント設定がサポートされています。Amazon FSx ファイルシステムとリンクされた S3 バケットは、同じ AWS アカウント または異なる に属します AWS アカウント。

データリポジトリの POSIX メタデータのサポート

Amazon FSx for Lustre は、Amazon S3 上のリンクされたデータリポジトリとの間でデータをインポートおよびエクスポートする際に、ファイル、ディレクトリ、シンボリックリンク (symlink) の Portable Operating System Interface (POSIX) メタデータを自動的に転送します。ファイルシステム内の変更をリンクされたデータリポジトリにエクスポートすると、FSx for Lustre は POSIX メタデータの変更も S3 オブジェクトのメタデータとして エクスポートします。つまり、別の FSx for Lustre ファイルシステムが S3 から同じファイルをインポートした場合、それらのファイルには、所有権やアクセス許可を含む、そのファイルシステム内にあるものと同じ POSIX メタデータが含まれるということです。

FSx for Lustre は、次のような POSIX 準拠のオブジェクトキーを持つ S3 オブジェクトのみをインポートします。

```
mydir/
mydir/myfile1
mydir/mysubdir/
mydir/mysubdir/myfile2.txt
```

FSx for Lustre は、ディレクトリおよびシンボリックリンクを個別のオブジェクトとして S3 上のリンクされたデータリポジトリに保存します。ディレクトリの場合、FSx for Lustre は、次のようにスラッシュ (「/」) で終わるキー名を持つ S3 オブジェクトを作成します。

- S3 オブジェクトキー `mydir/` は、FSx for Lustre ディレクトリ `mydir/` にマッピングされます。
- S3 オブジェクトキー `mydir/mysubdir/` は、FSx for Lustre ディレクトリ `mydir/mysubdir/` にマッピングされます。

シンボリックリンクの場合、FSx for Lustre は次の Amazon S3 スキーマを使用します。

- S3 オブジェクトキー - FSx for Lustre マウントディレクトリのリンク先を指定する相対パス
- S3 オブジェクトデータ - このシンボリックリンクのターゲットパス
- S3 オブジェクトメタデータ - シンボリックリンクのメタデータ

FSx for Lustre は、次のようなファイル、ディレクトリ、シンボリックリンクの所有権、アクセス許可、タイムスタンプなどの POSIX メタデータを S3 オブジェクトに保存します。

- `Content-Type` - ウェブブラウザのリソースのメディアタイプを示すために使用される HTTP エンティティヘッダー。
- `x-amz-meta-file-permissions` - Linux stat (2) のマニュアルページ の `st_mode` と一致する、`<octal file type><octal permission mask>` 形式のファイルタイプとアクセス許可。

> **Note**
>
> FSx for Lustre は `setuid` の情報をインポートまたは保持しません。

- `x-amz-meta-file-owner` - 整数で表された所有者ユーザー ID (UID)。
- `x-amz-meta-file-group` - 整数で表されるグループ ID (GID)。
- `x-amz-meta-file-atime` - Unix エポックの開始後のナノ秒単位の最終アクセス時間。時間値を `ns` で終了します。それ以外の場合、FSx for Lustre は値をミリ秒として解釈します。
- `x-amz-meta-file-mtime` - Unix エポックの開始後の最終修正時間。時間値を `ns` で終了します。それ以外の場合、FSx for Lustre は値をミリ秒として解釈します。
- `x-amz-meta-user-agent` - Amazon FSx のインポート中に無視されるユーザーエージェント。エクスポート中、FSx for Lustre はこの値を `aws-fsx-lustre` に設定します。

関連付けられた POSIX アクセス許可のないオブジェクトを S3 からインポートする場合、FSx for Lustre がファイルに割り当てるデフォルトの POSIX アクセス許可は `755` です。この許可は、すべてのユーザーに対する読み取りおよび実行アクセスと、ファイルの所有者に対する書き込みアクセスを許可します。

> **Note**
>
> FSx for Lustre は、S3 オブジェクト上のユーザー定義のカスタムメタデータを保持しません。

ハードリンクおよび Amazon S3 へのエクスポート

ファイルシステムの DRA で自動エクスポート (新規および変更されたポリシーを含む) が有効になっている場合、DRA に含まれる各ハードリンクは、ハードリンクごとに個別の S3 オブジェクトとして Amazon S3 にエクスポートされます。複数のハードリンクを含むファイルをファイルシステムで変更すると、ファイルの変更時にどのハードリンクが使用されたかに関係なく、S3 内のすべてのコピーが更新されます。

データリポジトリタスク (DRT) を使用してハードリンクを S3 にエクスポートする場合、DRT に指定されたパスに含まれる各ハードリンクは、ハードリンクごとに個別の S3 オブジェクトとして S3 にエクスポートされます。複数のハードリンクを含むファイルをファイルシステムで変更すると、ファイルの変更時にどのハードリンクが使用されたかに関係なく、S3 内の各コピーがそれぞれのハードリンクがエクスポートされた時点で更新されます。

> **Important**
>
> 新しい FSx for Lustre ファイルシステムが、別の FSx for Lustre ファイルシステム、 AWS DataSync、または Amazon FSx ファイルゲートウェイによって以前にハードリンクがエクスポートされた S3 バケットにリンクされると、ハードリンクはその後、新しいファイルシステムに個別のファイルとしてインポートされます。

ハードリンクとリリースされたファイル

リリースされたファイルとは、メタデータはファイルシステムに存在し、コンテンツは S3 にのみ保存されているファイルのことです。リリースされたファイルの詳細については、ファイルのリリースを参照してください。

> **Important**
>
> データリポジトリアソシエーション (DRA) 持つファイルシステムでハードリンクを使用することには、次の制限があります。
>
> - 複数のハードリンクを持つリリース済みファイルを削除して再作成すると、すべてのハードリンクの内容が上書きされる可能性があります。
> - リリースされたファイルを削除すると、データリポジトリアソシエーションの外部にあるすべてのハードリンクからコンテンツが削除されます。

- 対応する S3 オブジェクトが S3 Glacier Flexible Retrieval または S3 Glacier Deep Archive ストレージクラスのいずれかにあるリリース済みファイルへのハードリンクを作成しても、ハードリンク用に S3 に新しいオブジェクトが作成されることはありません。

チュートリアル: Simple Storage Service (Amazon S3) バケットにオブジェクトをアップロードする際の POSIX アクセス許可を付与する

次の手順では、POSIX アクセス許可を使用してオブジェクトを Simple Storage Service (Amazon S3) にアップロードするプロセスについて説明します。これにより、その S3 バケットにリンクされている Amazon FSx ファイルシステムを作成する際に POSIX アクセス許可をインポートできます。

POSIX アクセス許可を持つオブジェクトを Simple Storage Service (Amazon S3) にアップロードするには

1. ローカルコンピュータまたはマシンから、次のコマンド例を使用して、S3バケットにアップロードされるテストディレクトリ (`s3cptestdir`) とファイル (`s3cptest.txt`) を作成します。

   ```
   $ mkdir s3cptestdir
   $ echo "S3cp metadata import test" >> s3cptestdir/s3cptest.txt
   $ ls -ld s3cptestdir/ s3cptestdir/s3cptest.txt
   drwxr-xr-x 3 500 500 96 Jan 8 11:29 s3cptestdir/
   -rw-r--r-- 1 500 500 26 Jan 8 11:29 s3cptestdir/s3cptest.txt
   ```

   新しく作成されたファイルとディレクトリには、前の例に示すように、ファイル所有者のユーザー ID (UID) とグループ ID (GID) が 500 で、アクセス許可があります。

2. Simple Storage Service (Amazon S3) API を呼び出して、メタデータ許可を持つディレクトリ `s3cptestdir` を作成します。ディレクトリ名は、末尾にスラッシュ (/) を付けて指定する必要があります。サポートされている POSIX メタデータについては、「データリポジトリの POSIX メタデータのサポート」を参照してください。

   `bucket_name` を実際の S3 バケット名に置き換えます。

   ```
   $ aws s3api put-object --bucket bucket_name --key s3cptestdir/ --metadata '{"user-
   agent":"aws-fsx-lustre" , \
         "file-atime":"1595002920000000000ns" , "file-owner":"500" , "file-
   permissions":"0100664","file-group":"500" , \
   ```

```
      "file-mtime":"1595002920000000000ns"}'
```

3. POSIX アクセス許可が S3 オブジェクトメタデータにタグ付けされていることを確認します。

```
$ aws s3api head-object --bucket bucket_name --key s3cptestdir/
{
    "AcceptRanges": "bytes",
    "LastModified": "Fri, 08 Jan 2021 17:32:27 GMT",
    "ContentLength": 0,
    "ETag": "\"d41d8cd98f00b204e9800998ecf8427e\"",
    "VersionId": "bAlhCoWq7aIEjc3R6Myc6UOb8sHHtJkR",
    "ContentType": "binary/octet-stream",
    "Metadata": {
        "user-agent": "aws-fsx-lustre",
        "file-atime": "1595002920000000000ns",
        "file-owner": "500",
        "file-permissions": "0100664",
        "file-group": "500",
        "file-mtime": "1595002920000000000ns"
    }
}
```

4. メタデータのアクセス許可を使用して、コンピュータから S3 バケットにテストファイル (ステップ 1 で作成した) をアップロードします。

```
$ aws s3 cp s3cptestdir/s3cptest.txt s3://bucket_name/s3cptestdir/s3cptest.txt \
      --metadata '{"user-agent":"aws-fsx-lustre" , "file-
atime":"1595002920000000000ns" , \
      "file-owner":"500" , "file-permissions":"0100664","file-group":"500" , "file-
mtime":"1595002920000000000ns"}'
```

5. POSIX アクセス許可が S3 オブジェクトメタデータにタグ付けされていることを確認します。

```
$ aws s3api head-object --bucket bucket_name --key s3cptestdir/s3cptest.txt
{
    "AcceptRanges": "bytes",
    "LastModified": "Fri, 08 Jan 2021 17:33:35 GMT",
    "ContentLength": 26,
    "ETag": "\"eb33f7e1f44a14a8e2f9475ae3fc45d3\"",
    "VersionId": "w9ztRoEhB832m8NC3a_JTlTyIx7Uzql6",
    "ContentType": "text/plain",
    "Metadata": {
        "user-agent": "aws-fsx-lustre",
```

```
        "file-atime": "1595002920000000000ns",
        "file-owner": "500",
        "file-permissions": "0100664",
        "file-group": "500",
        "file-mtime": "1595002920000000000ns"
    }
}
```

6. S3 バケットにリンクされている Amazon FSx ファイルシステムに対するアクセス許可を確認します。

```
$ sudo lfs df -h /fsx
UUID                       bytes        Used   Available Use% Mounted on
3rnxfbmv-MDT0000_UUID       34.4G        6.1M       34.4G   0% /fsx[MDT:0]
3rnxfbmv-OST0000_UUID        1.1T        4.5M        1.1T   0% /fsx[OST:0]

filesystem_summary:          1.1T        4.5M        1.1T   0% /fsx

$ cd /fsx/s3cptestdir/
$ ls -ld s3cptestdir/
drw-rw-r-- 2 500 500 25600 Jan  8 17:33 s3cptestdir/

$ ls -ld s3cptestdir/s3cptest.txt
-rw-rw-r-- 1 500 500 26 Jan 8 17:33 s3cptestdir/s3cptest.txt
```

`s3cptestdir` ディレクトリと `s3cptest.txt` ファイルの両方に POSIX 許可がインポートされています。

Amazon S3 バケットにファイルシステムにリンクする

Amazon FSx for Lustre ファイルシステムを Simple Storage Service (Amazon S3) のデータリポジトリにリンクできます。リンクは、ファイルシステムの作成時、またはファイルシステムの作成後いつでも作成できます。

ファイルシステム上のディレクトリと S3 バケットまたはプレフィックス間のリンクは、データリポジトリの関連付け (DRA) と呼ばれます。FSx for Lustre ファイルシステムには、最大 8 つのデータリポジトリの関連付けを設定できます。最大 8 つの DRA リクエストをキューに入れることができますが、ファイルシステムに対して一度に処理できるリクエストは 1 つだけです。各 DRA には、一意の FSx for Lustre ファイルシステムディレクトリと、それに関連付けられた一意の S3 バケットまたはプレフィックスが必要です。

> **Note**
>
> データリポジトリの関連付け、自動エクスポート、複数のリポジトリのサポートは、FSx for Lustre 2.10 ファイルシステムと `Scratch 1` ファイルシステムでは使用できません。

S3 データリポジトリ上のオブジェクトにファイルシステム上のファイルとディレクトリとしてアクセスするには、ファイルおよびディレクトリのメタデータをファイルシステムにロードする必要があります。DRA を作成する際に、リンク先のデータリポジトリからメタデータをロードしたり、データリポジトリのインポートタスクを使用して、FSx for Lustre ファイルシステムを使用してアクセスするファイルやディレクトリのバッチのメタデータを後でロードしたりできます。また、自動エクスポートを使用して、オブジェクトがデータリポジトリに追加、変更、削除された場合にメタデータを自動的にロードすることもできます。

DRA は、自動インポートのみ、自動エクスポートのみ、またはその両方に設定できます。自動インポートと自動エクスポートの両方で設定されたデータリポジトリの関連付けは、ファイルシステムとリンクされた S3 バケット間で両方向にデータを転送します。S3 バケット内のデータに変更を加えると、FSx for Lustre が変更を検出し、その変更をファイルシステムに自動的にインポートします。ファイルを作成、変更、または削除すると、アプリケーションがファイルの変更を完了した後、FSx for Lustre が変更を非同期的に Amazon S3 に自動でエクスポートします。

> **Important**
>
> - ファイルシステムと S3 バケットの両方で同じファイルを変更する場合は、アプリケーションレベルを調整して競合を防ぐ必要があります。FSx for Lustre では、複数の場所での競合する書き込みを防止できません。
> - 不変属性でマークされたファイルの場合、FSx for Lustre は、FSx for Lustre ファイルシステムと、ファイルシステムにリンクされた S3 バケット間の変更を同期できません。不変フラグを長期間設定すると、Amazon FSx と S3 間のデータ移動のパフォーマンスが低下する可能性があります。

データリポジトリの関連付けを作成すると、次のプロパティを設定できます。

- ファイルシステムパス - ディレクトリを指すファイルシステム上のローカルパス (`/ns1/` など) またはサブディレクトリ (`/ns1/subdir/` など) を指すファイルシステムのローカルパスを入力します。入力されたデータは、以下の指定されたデータリポジトリパスで 特定のデータに 1 対 1 で

マッピングされます。名前の先頭のスラッシュは必須です。2 つのデータリポジトリの関連付けは、重複するファイルシステムパスを持つことはできません。例えば、データリポジトリがファイルシステムパス /ns1 に関連付けられている場合、ファイルシステムパス /ns1/ns2 に別のデータリポジトリをリンクすることはできません。

> **Note**
>
> ファイルシステムパスとしてスラッシュ (/) のみを指定した場合、ファイルシステムにリンクできるデータリポジトリは 1 つだけです。「/」は、ファイルシステムに関連付けられた最初のデータリポジトリのファイルシステムパスとしてのみ指定できます。

- データリポジトリパス - S3 データリポジトリにパスを入力します。パスには、次の S3 バケットまたは `s3://bucket-name/prefix/` 形式のプレフィックスを使用できます。このプロパティは、S3 データリポジトリのファイルのインポート先またはエクスポート先を指定します。特に指定しなければ、FSx for Lustre はデータリポジトリのパスに末尾の「/」を追加します。例えば、`s3://amzn-s3-demo-bucket/my-prefix` のデータリポジトリのパスを指定すると、FSx for Lustre はそれを `s3://amzn-s3-demo-bucket/my-prefix/` として解釈します。

  2 つのデータリポジトリの関連付けは、重複するデータリポジトリパスを持つことはできません。例えば、パス `s3://amzn-s3-demo-bucket/my-prefix/` があるデータリポジトリがファイルシステムにリンクされている場合、データリポジトリのパス `s3://amzn-s3-demo-bucket/my-prefix/my-sub-prefix` と別のデータリポジトリの関連付けを作成することはできません。
- リポジトリからメタデータをインポートする - このオプションを選択すると、データリポジトリの関連付けを作成した直後にデータリポジトリ全体からメタデータをインポートできます。または、データリポジトリのインポートタスクを実行して、データリポジトリの関連付けが作成された後でも、リンクされたデータリポジトリのメタデータのすべてまたはサブセットをファイルシステムにロードできます。
- 設定のインポート - リンクされた S3 バケットからファイルシステムに自動的にインポートされる、更新されたオブジェクトのタイプ (新規、変更、および削除の任意の組み合わせ) を指定するインポートポリシーを選択します。自動インポート (新規、変更、削除) は、コンソールからデータリポジトリを追加するとデフォルトで有効になりますが、 AWS CLI または Amazon FSx API を使用する場合はデフォルトで無効になります。
- 設定をエクスポートする - S3 バケットに自動的にエクスポートされる更新されたオブジェクトのタイプ (新規、変更、および削除の任意の組み合わせ) を指定するエクスポートポリシーを選択します。自動エクスポート (新規、変更、削除) は、コンソールからデータリポジトリを追加すると

デフォルトで有効になりますが、 AWS CLI または Amazon FSx API を使用する場合はデフォルトで無効になります。

ファイルシステムパスとデータリポジトリパスの設定により、Amazon FSx のパスと S3 のオブジェクトキーが 1 対 1 でマッピングされます。

トピック

S3 バケットへのリンクの作成

以下の手順では、 AWS マネジメントコンソール と AWS Command Line Interface () を使用して、FSx for Lustre ファイルシステムのデータリポジトリを既存の S3 バケットに関連付けるプロセスについて説明しますAWS CLI。S3 バケットをファイルシステムがリンクする方法については、「Simple Storage Service (Amazon S3) でデータリポジトリを使用する許可を追加する」を参照してください。

> **Note**
>
> データリポジトリは、ファイルシステムバックアップが有効になっているファイルシステムにリンクすることはできません。データリポジトリをリンクする前にバックアップを無効にします。

ファイルシステムの作成中に S3 バケットをリンクするには (コンソール)

1. https://console.aws.amazon.com/fsx/ で Amazon FSx コンソールを開きます。
2. 「はじめに」ステップ 1: FSx for Lustre ファイルシステムの作成 セクションで説明されている新しいファイルシステムを作成する手順に従います。
3. [Data Repository Import/Export -optional] (データリポジトリ インポート / エクスポート - オプション) セクションを開きます。この機能は、デフォルトでは無効になっています。

4. [Import data from and export data to S3] (データを S3 からインポートし、データを S3 にエクスポートする) を選択します。
5. データリポジトリ関連付け情報 ダイアログで、以下のフィールドに情報を入力します。

   - ファイルシステムパス: S3 データリポジトリに関連付けられる Amazon FSx ファイルシステム内に、ハイレベルディレクトリの名前 (/ns1 など) またはサブディレクトリの名前 (/ns1/subdir など) を入力します。パスの先頭のスラッシュが必要です。2 つのデータリポジトリの関連付けは、重複するファイルシステムパスを持つことはできません。例えば、データリポジトリがファイルシステムパス /ns1 に関連付けられている場合、ファイルシステムパス /ns1/ns2 に別のデータリポジトリをリンクすることはできません。ファイルシステムパス 設定は、ファイルシステムのすべてのデータリポジトリの関連付けで一意である必要があります。
   - データリポジトリパス: ファイルシステムに関連付ける既存の S3 バケットまたはプレフィックスのパスを入力します (例えば、s3://amzn-s3-demo-bucket/my-prefix)。2 つのデータリポジトリの関連付けは、重複するデータリポジトリパスを持つことはできません。データリポジトリパス 設定は、ファイルシステムのすべてのデータリポジトリの関連付けで一意である必要があります。
   - リポジトリからメタデータをインポートする: このプロパティを選択すると、オプションで、リンクが作成された直後にメタデータをインポートするデータリポジトリのインポートタスクを実行できます。

6. 設定のインポート-オプション で、[Import Policy] (ポリシーのインポート) を設定します。これにより、S3 バケット内のオブジェクトを追加、変更、または削除する際に、ファイルおよびディレクトリのリストを最新の状態に保つ方法が決定されます。例えば、[New] (新規) をクリックして、S3 バケットで作成された新しいオブジェクトのメタデータをファイルシステムにインポートします。インポートポリシーの詳細については、「S3 バケットから更新を自動的にインポートする」を参照してください。
7. [Export Policy] (ポリシーをエクスポートする) で、ファイルシステム内のオブジェクトを追加、変更、または削除するときに、リンクされた S3 バケットにファイルをエクスポートする方法を決定するエクスポートポリシーを設定します。例えば、[Changed] (変更済) を選択して、ファイルシステム上でコンテンツまたはメタデータが変更されたオブジェクトをエクスポートします。ポリシーのエクスポートの詳細については、「S3 バケットに更新を自動的にエクスポートする」を参照してください。
8. ファイルシステム作成ウィザードの次のセクションに進みます。

S3 バケットを既存のファイルシステム (コンソール) にリンクするには

1. https://console.aws.amazon.com/fsx/ で Amazon FSx コンソールを開きます。
2. ダッシュボードから、[File systems] (ファイルシステム) を選択します。次に、データリポジトリの関連付けを作成する対象のファイルシステムを選択します。
3. [Data repository] (データリポジトリ) タブを選択します。
4. [Data repository associations] (データリポジトリ関連) ペインで、[Create data repository association] (データリポジトリの関連付けを作成する) を選択します。
5. [Data repository association information] (データリポジトリ関連付け情報) ダイアログで、以下のフィールドに情報を入力します。

   - ファイルシステムパス: S3 データリポジトリに関連付けられる Amazon FSx ファイルシステム内に、ハイレベルディレクトリの名前 (`/ns1` など) またはサブディレクトリの名前 (`/ns1/subdir` など) を入力します。パスの先頭のスラッシュが必要です。2 つのデータリポジトリの関連付けは、重複するファイルシステムパスを持つことはできません。例えば、データリポジトリがファイルシステムパス `/ns1` に関連付けられている場合、ファイルシステムパス `/ns1/ns2` に別のデータリポジトリをリンクすることはできません。ファイルシステムパス 設定は、ファイルシステムのすべてのデータリポジトリの関連付けで一意である必要があります。
   - データリポジトリパス: ファイルシステムに関連付ける既存の S3 バケットまたはプレフィックスのパスを入力します (例えば、`s3://amzn-s3-demo-bucket/my-prefix`)。2 つのデータリポジトリの関連付けは、重複するデータリポジトリパスを持つことはできません。データリポジトリパス 設定は、ファイルシステムのすべてのデータリポジトリの関連付けで一意である必要があります。
   - リポジトリからメタデータをインポートする: このプロパティを選択すると、オプションで、リンクが作成された直後にメタデータをインポートするデータリポジトリのインポートタスクを実行できます。

6. 設定のインポート-オプション で、[Import Policy] (ポリシーのインポート) を設定します。これにより、S3 バケット内のオブジェクトを追加、変更、または削除する際に、ファイルおよびディレクトリのリストを最新の状態に保つ方法が決定されます。例えば、[New] (新規) をクリックして、S3 バケットで作成された新しいオブジェクトのメタデータをファイルシステムにインポートします。インポートポリシーの詳細については、「S3 バケットから更新を自動的にインポートする」を参照してください。
7. [Export Policy] (ポリシーをエクスポートする) で、ファイルシステム内のオブジェクトを追加、変更、または削除するときに、リンクされた S3 バケットにファイルをエクスポートする方法を

決定するエクスポートポリシーを設定します。例えば、[Changed] (変更済) を選択して、ファイルシステム上でコンテンツまたはメタデータが変更されたオブジェクトをエクスポートします。ポリシーのエクスポートの詳細については、「S3 バケットに更新を自動的にエクスポートする」を参照してください。

8. [作成] を選択します。

ファイルシステムを S3 バケット (AWS CLI) にリンクするには

次の例では、Amazon FSx ファイルシステムを S3 バケットにリンクするデータリポジトリの関連付けを作成します。これには、新規または変更されたファイルをファイルシステムにインポートするインポートポリシーと、新規、変更、または削除したファイルをリンクされた S3 バケットにエクスポートするエクスポートポリシーを使用します。

- データリポジトリの関連付けを作成するには、以下に示すように、Amazon FSx CLI コマンド `create-data-repository-association` を使用します。

```
$ aws fsx create-data-repository-association \
      --file-system-id fs-0123456789abcdef0 \
      --file-system-path /ns1/path1/ \
      --data-repository-path s3://amzn-s3-demo-bucket/myprefix/ \
      --s3
 "AutoImportPolicy={Events=[NEW,CHANGED,DELETED]},AutoExportPolicy={Events=[NEW,CHANGED,DEL
```

Amazon FSx は、DRA の JSON 記述をすぐに返します。DRA は非同期に作成されます。

このコマンドを使用すると、ファイルシステムの作成が完了する前でも、データリポジトリの関連付けを作成できます。ファイルシステムが使用可能になった後、リクエストはキューに入れられ、データリポジトリの関連付けが作成されます。

データリポジトリの関連付け設定の更新

既存のデータリポジトリの関連付けの設定は、以下の手順で示すように AWS マネジメントコンソール、 AWS CLI、および Amazon FSx API を使用して更新できます。

> **Note**
>
> DRA の作成後は DRA の `File system path` または `Data repository path` を更新することはできません。`File system path` または `Data repository path` を変更する場合は、DRA を削除してから再度作成する必要があります。

既存のデータリポジトリ関連付けの設定を更新するには (コンソール)

1. https://console.aws.amazon.com/fsx/ で Amazon FSx コンソールを開きます。
2. ダッシュボードから、[File systems] (ファイルシステム) をクリックし、管理するファイルシステムを選択します。
3. [Data repository] (データリポジトリ) タブを選択します。
4. [Data repository associations] (データリポジトリ関連) ペインで、変更するデータリポジトリ関連付けを選択します。
5. [Update] (更新) を選択します。データリポジトリの関連付けの編集ダイアログが表示されます。
6. [Import settings - optional] (設定のインポート-オプション) で、[Import Policy] (ポリシーのインポート) を更新することができます。インポートポリシーの詳細については、「S3 バケットから更新を自動的にインポートする」を参照してください。
7. [Export settings - optional] (エクスポート設定 - オプション) でエクスポートポリシーを更新できます。ポリシーのエクスポートの詳細については、「S3 バケットに更新を自動的にエクスポートする」を参照してください。
8. [Update] (更新) を選択します。

既存のデータリポジトリの関連付けの設定を更新するには (CLI)

- データリポジトリの関連付けを更新するには、以下に示すように Amazon FSx CLI コマンド `update-data-repository-association` を使用します。

```
$ aws fsx update-data-repository-association \
      --association-id 'dra-872abab4b4503bfc2' \
      --s3
 "AutoImportPolicy={Events=[NEW,CHANGED,DELETED]},AutoExportPolicy={Events=[NEW,CHANGED,DEL
```

データリポジトリ関連付けのインポートポリシーおよびエクスポートポリシーが正常に更新されると、Amazon FSx は更新されたデータリポジトリ関連付けの説明を JSON として返します。

S3 バケットへの関連付けを削除する

次の手順では、 AWS マネジメントコンソール と AWS Command Line Interface () を使用して、既存の Amazon FSx ファイルシステムから既存の S3 バケットにデータリポジトリの関連付けを削除するプロセスについて説明しますAWS CLI。データリポジトリの関連付けを削除すると、S3 バケットからファイルシステムのリンクが解除されます。

ファイルシステムから S3 バケットへのリンクを削除するには (コンソール)

1. https://console.aws.amazon.com/fsx/ で Amazon FSx コンソールを開きます。
2. ダッシュボードから、[File systems] (ファイルシステム) を選択します。次に、データリポジトリの関連付けを削除するファイルシステムを選択します。
3. [Data repository] (データリポジトリ) タブを選択します。
4. [Data repository associations] (データリポジトリ関連) ペインで、削除するデータリポジトリの関連付けを選択します。
5. [Actions] (アクション) で、[Delete association] (関連付けを削除する) を選択します。
6. [削除] ダイアログで、[ファイルシステム内のデータを削除する] を選択して、データリポジトリの関連付けに対応するファイルシステム内のデータを物理的に削除することができます。

   同一のファイルシステムパスを使用しつつ、別の S3 バケットプレフィックスを参照する新しいデータリポジトリ関連付けを作成する予定がある場合、またはファイルシステム内のデータが不要になった場合は、このオプションを選択します。
7. [Delete] (削除) をクリックして、ファイルシステムからデータリポジトリの関連付けを削除します。

ファイルシステムから S3 バケット (AWS CLI) へのリンクを削除するには

次の例では、Amazon FSx ファイルシステムを S3 バケットにリンクするデータリポジトリの関連付けを削除します。`--association-id` パラメータは、削除するデータリポジトリの関連付けの ID を指定します。

- データリポジトリの関連付けを削除するには、以下に示すように Amazon FSx CLI コマンド `delete-data-repository-association` を使用します。

```
$ aws fsx delete-data-repository-association \
```

```
    --association-id dra-872abab4b4503bfc \
    --delete-data-in-file-system false
```

データリポジトリの関連付けを正常に削除すると、Amazon FSx はその説明を JSON として返します。

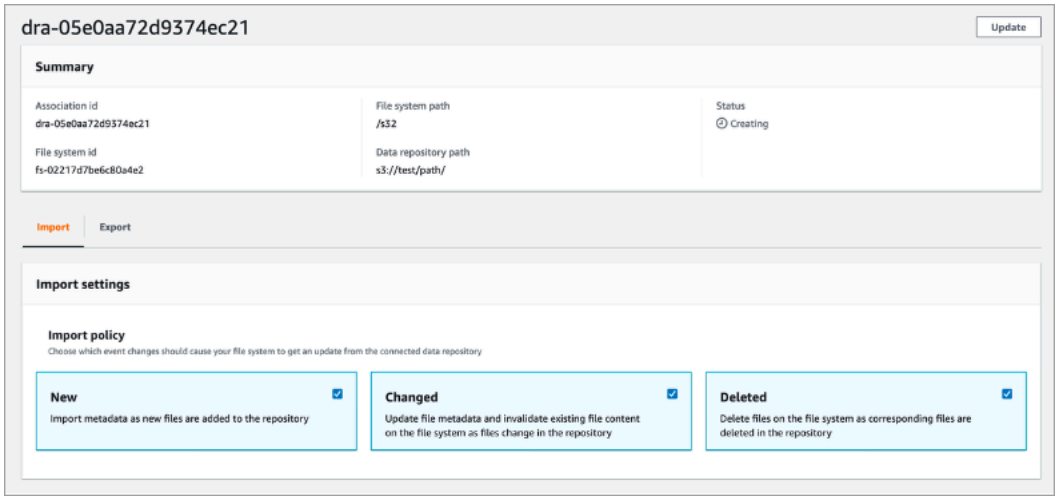
> **同じファイルシステムパスで DRAs を再作成する**
>
> 同じファイルシステムパスを使用するデータリポジトリの関連付けを削除して再作成することはお勧めしません。DRA を削除し、後で同じファイルシステムパスを使用して新しい DRA を作成すると、一部のファイルは以前に削除された DRA から HSM 状態を保持する場合があります。
> 以前に削除された DRA によって管理された再作成された DRA からファイルをエクスポートする必要がある場合は、以下のコマンドを使用してそれらのファイルをダーティとしてマークし、データリポジトリのエクスポートタスクを実行する必要があります。
>
> ```
> sudo lfs hsm_set --dirty file_path
> ```

データリポジトリの関連付けの詳細の表示

FSx for Lustre コンソール、、および API を使用して AWS CLI、データリポジトリの関連付けの詳細を表示できます。詳細には、DRA の関連付け ID、ファイルシステムパス、データリポジトリパス、インポート設定、エクスポート設定、ステータス、および関連するファイルシステムの ID が含まれます。

DRA の詳細を表示するには (コンソール)

1. https://console.aws.amazon.com/fsx/ で Amazon FSx コンソールを開きます。
2. ダッシュボードから、[File system] (ファイルシステム) を選択してから、データリポジトリの関連付けの詳細を表示するファイルシステムを選択します。
3. [Data repository] (データリポジトリ) タブを選択します。
4. [Data repository associations] (データリポジトリ関連) ペインで、表示するデータリポジトリの関連付けを選択します。[Summary] (概要) ページが表示され、DRA の詳細が表示されます。



DRA の詳細を表示するには (CLI)

- 特定のデータリポジトリ関連付けの詳細を表示するには、以下に示すように Amazon FSx CLI `describe-data-repository-associations` コマンドを使用します。

```
$ aws fsx describe-data-repository-associations \
      --association-ids dra-872abab4b4503bfc2
```

  Amazon FSx は、データリポジトリの関連付けの説明を JSON として返します。

データリポジトリの関連付けのライフサイクル状態

データリポジトリの関連付けのライフサイクル状態は、特定の DRA に関するステータス情報を提供します。データリポジトリ関連付けには、次の ライフサイクル状態 があります。

- 作成中 - Amazon FSx は、ファイルシステムとリンクされたデータリポジトリの間にデータリポジトリの関連付けを作成しています。データリポジトリは使用できません。
- 使用可能 - データリポジトリの関連付けを使用できます。
- 更新中 - データリポジトリの関連付けは、可用性に影響する可能性があるお客様が開始した更新を実行しています。
- 削除中 - データリポジトリの関連付けは、お客様が開始した削除を実行しています。
- 設定が不適切です - Amazon FSx は、データリポジトリの関連付け設定が修正されるまで、S3 バケットから更新を自動的にインポートしたり、S3 バケットに更新を自動的に エクスポートしたりすることはできません。

DRA は、以下の理由で 設定ミス になる可能性があります:

- Amazon FSx には、S3 バケットにアクセスするために必要な IAM アクセス許可がありません。
- これは、S3 バケットの FSx イベント通知設定が削除または変更されます。
- S3 バケットには、FSx イベントタイプと重複する既存のイベント通知があります。

根本的な問題を解決した後、DRA は 15 分以内に自動的に Available 状態に戻るか、update-data-repository-association AWS CLI コマンドを使用してすぐに状態変更をトリガーできます。

- 失敗 - データリポジトリの関連付けは、回復できないターミナル状態にあります (例えば、ファイルシステムパスが削除される、S3 バケットが削除されるなど)。

Amazon FSx コンソール、、および Amazon FSx API を使用して AWS Command Line Interface、データリポジトリの関連付けのライフサイクル状態を表示できます。詳細については、「データリポジトリの関連付けの詳細の表示」を参照してください。

サーバー側で暗号化された Simple Storage Service (Amazon S3) バケットの使用

FSx for Lustre は、S3 マネージドキーによるサーバー側の暗号化 (SSE-S3)、および (SSE-KMS) に保存されている を使用する Amazon S3 バケットをサポートしています。 S3-managed AWS KMS keys AWS Key Management Service

S3 バケットに書き込むときに Amazon FSx でデータを暗号化するには、S3 バケットのデフォルトの暗号化を SSE-S3 または SSE-KMS に設定する必要があります。詳細については、「Amazon S3 ユーザーガイド」の「デフォルトの暗号化の設定」を参照してください。S3 バケットにファイルを書き込む場合、Amazon FSx は S3 バケットのデフォルトの暗号化ポリシーに従います。

デフォルトでは、Amazon FSx は SSE-S3 を使用して暗号化された S3 バケットをサポートします。SSE-KMS 暗号化を使用して暗号化された S3 バケットに Amazon FSx ファイルシステムをリンクする場合は、Amazon FSx が KMS キーを使用して S3 バケット内のオブジェクトを暗号化および復号化できるようにするステートメントをカスタマー管理キーポリシーに追加する必要があります。

次のステートメントは、特定の Amazon FSx ファイルシステムで、特定の S3 バケットのオブジェクトを暗号化および復号化することを許可します。*bucket_name*。

```
{
    "Sid": "Allow access through S3 for the FSx SLR to use the KMS key on the objects
 in the given S3 bucket",
```

```
    "Effect": "Allow",
    "Principal": {
        "AWS": "arn:aws:iam::aws_account_id:role/aws-service-role/s3.data-
source.lustre.fsx.amazonaws.com/AWSServiceRoleForFSxS3Access_fsx_file_system_id"
    },
    "Action": [
        "kms:Encrypt",
        "kms:Decrypt",
        "kms:ReEncrypt*",
        "kms:GenerateDataKey*",
        "kms:DescribeKey"
    ],
    "Resource": "*",
    "Condition": {
        "StringEquals": {
            "kms:CallerAccount": "aws_account_id",
            "kms:ViaService": "s3.bucket-region.amazonaws.com"
        },
        "StringLike": {
            "kms:EncryptionContext:aws:s3:arn": "arn:aws:s3:::bucket_name/*"
        }
    }
}
```

Note

CMK で KMS を使用して S3 バケットキーを有効にして S3 バケットを暗号化する場合は、次の例で示すとおり、EncryptionContext をオブジェクト ARN ではなくバケット ARN に設定します。

```
"StringLike": {
    "kms:EncryptionContext:aws:s3:arn": "arn:aws:s3:::bucket_name"
}
```

次のポリシーステートメントでは、アカウント内のすべての Amazon FSx ファイルシステムが特定の S3 バケットにリンクすることを許可します。

```
{
      "Sid": "Allow access through S3 for the FSx SLR to use the KMS key on the objects
 in the given S3 bucket",
```

```
      "Effect": "Allow",
      "Principal": {
        "AWS": "*"
      },
      "Action": [
        "kms:Encrypt",
        "kms:Decrypt",
        "kms:ReEncrypt*",
        "kms:GenerateDataKey*",
        "kms:DescribeKey"
      ],
      "Resource": "*",
      "Condition": {
        "StringEquals": {
          "kms:ViaService": "s3.bucket-region.amazonaws.com",
          "kms:CallerAccount": "aws_account_id"
        },
        "StringLike": {
            "kms:EncryptionContext:aws:s3:arn": "arn:aws:s3:::bucket_name/*"
        },
        "ArnLike": {
          "aws:PrincipalArn": "arn:aws_partition:iam::aws_account_id:role/aws-service-
role/s3.data-source.lustre.fsx.amazonaws.com/AWSServiceRoleForFSxS3Access_fs-*"
        }
      }
}
```

別の AWS アカウント または共有 VPC からのサーバー側の暗号化された Amazon S3 バケットへのアクセス

FSx for Lustre ファイルシステムを暗号化した Simple Storage Service (Amazon S3) バケットを作成したら、`AWSServiceRoleForFSxS3Access_fs-01234567890` リンクされた S3 バケットからデータを読み書きする前に S3 バケットを暗号化するために使用される KMS キーへのサービスリンクロール (SLR) アクセス。KMS キーに対するアクセス許可が既にある IAM ロールを使用できます。

> **Note**
>
> この IAM ロールは、KMS キー / S3 バケットが属するアカウントではなく、FSx for Lustre ファイルシステムが作成されたアカウント (S3 SLR と同じアカウント) に存在する必要があります。

IAM ロールを使用して次の AWS KMS API を呼び出し、S3 SLR が S3 オブジェクトへのアクセス許可を取得できるように S3 SLR の許可を作成します。SLR に関連付けられている ARN を見つけるには、ファイルシステム ID を検索文字列として使用して IAM ロールを検索します。

```
$ aws kms create-grant --region fs_account_region \
      --key-id arn:aws:kms:s3_bucket_account_region:s3_bucket_account:key/key_id \
      --grantee-principal arn:aws:iam::fs_account_id:role/aws-service-role/s3.data-
source.lustre.fsx.amazonaws.com/AWSServiceRoleForFSxS3Access_file-system-id \
      --operations "Decrypt" "Encrypt" "GenerateDataKey"
 "GenerateDataKeyWithoutPlaintext" "CreateGrant" "DescribeKey" "ReEncryptFrom"
 "ReEncryptTo"
```

サービスにリンクされたロールの詳細については、「Amazon FSx のサービスリンクロールの使用」を参照してください。

データリポジトリからの変更のインポート

データおよび POSIX メタデータを含むメタデータの変更を、リンクされたデータリポジトリから Amazon FSx ファイルシステムにエクスポートできます。関連する POSIX メタデータには、所有権、許可、およびタイムスタンプが含まれます。

変更をファイルシステムにインポートするには、次のいずれかの方法を使用します。

- リンクされたデータリポジトリに新規ファイル、変更されたファイル、または削除されたファイルを自動的にエクスポートできるようにファイルシステムを設定します。詳細については、「S3 バケットから更新を自動的にインポートする」を参照してください。
- データリポジトリの関連付けを作成する際に、メタデータをインポートするためのオプションを選択します。これにより、データリポジトリの関連付けを作成した直後に、データリポジトリのインポートタスクが開始されます。
- オンデマンドのデータリポジトリのインポートタスクを使用します。詳細については、「データリポジトリのタスクを使用して変更をインポートする」を参照してください。

自動インポートとデータリポジトリのインポートタスクは同時に実行できます。

データリポジトリの関連付けで自動インポートを有効にすると、S3 でオブジェクトが作成、変更、または削除されたときに、ファイルシステムによってファイルメタデータが自動的に更新されます。データリポジトリの関連付けの作成時にメタデータをインポートするオプションを選択すると、デー

タリポジトリ内の全オブジェクトのメタデータがファイルシステムによってインポートされます。データリポジトリのインポートタスクを使用してインポートする場合、前回のインポート以降に作成または変更されたオブジェクトのメタデータのみがファイルシステムによってインポートされます。

FSx for Lustre は、アプリケーションがファイルシステム内のファイルに最初にアクセスする際に、データリポジトリからファイルの内容を自動的にコピーしてファイルシステムにロードします。このデータの移動は FSx for Lustre によって管理されており、アプリケーションに対して透過的に行われます。その後に行われるファイルの読み取りは、ミリ秒未満のレイテンシーでファイルシステムから直接提供されます。

ファイルシステム全体またはファイルシステム内のディレクトリをプリロードすることもできます。詳細については、「ファイルシステムへのファイルのプリロード」を参照してください。複数のファイルのプリロードを同時にリクエストすると、FSx for Lustre は Amazon S3 データリポジトリからファイルを並行してロードします。

FSx for Lustre は、POSIX 準拠のオブジェクトキーを持つ S3 オブジェクトのみをインポートします。自動インポートおよびデータリポジトリのインポートタスクの両方で、POSIX メタデータがインポートされます。詳細については、「データリポジトリの POSIX メタデータのサポート」を参照してください。

> **Note**
>
> FSx for Lustre では、S3 Glacier Flexible Retrieval および S3 Glacier Deep Archive ストレージクラスからのシンボリックリンク (symlink) メタデータのインポートはサポートされていません。シンボリックリンクではない S3 Glacier Flexible Retrieval オブジェクトまたは S3 Glacier Deep Archive オブジェクトのメタデータをインポートできます (つまり、正しいメタデータを使用して FSx for Lustre ファイルシステムで inode が作成されます)。ただし、ファイルシステムからこのデータを読み取るには、始めに S3 Glacier Flexible Retrieval または S3 Glacier Deep Archive オブジェクトを復元する必要があります。S3 Glacier Flexible Retrieval または S3 Glacier Deep Archive ストレージクラスの Amazon S3 オブジェクトから FSx for Lustre へのファイルデータの直接インポートはサポートされていません。

S3 バケットから更新を自動的にインポートする

FSx for Lustre は、オブジェクトが S3 バケットに追加、変更されたとき、または S3 バケットから削除されたときに、ファイルシステムのメタデータを自動的に更新するように設定できます。FSx for Lustre は、S3 の変更に対応して、ファイルとディレクトリのリストを作成、更新、または削除

します。S3 バケット内の変更されたオブジェクトにメタデータが含まれなくなった場合、FSx for Lustre は、現在のアクセス許可を含む現在のメタデータの値を保持します。

> **Note**
>
> 更新を自動でインポートするためには、FSx for Lustre ファイルシステムとリンクされた S3 バケットが同じ AWS リージョン に配置されている必要があります。

データリポジトリの関連付けを作成するときに自動インポートを設定し、FSx マネジメントコンソール、、 AWS CLIまたは AWS API を使用していつでも自動インポート設定を更新できます。

> **Note**
>
> 同じデータリポジトリの関連付けで、自動インポートと自動エクスポートの両方を設定できます。このトピックでは、自動インポート機能についてのみ説明します。

> **Important**
>
> - すべての自動インポートのポリシーが有効になっており、自動エクスポートが無効になっている状態で S3 でオブジェクトが変更された場合、そのオブジェクトのコンテンツは常にファイルシステム内の対応するファイルにインポートされます。ターゲットの場所にファイルが既に存在する場合、そのファイルは上書きされます。
> - 自動インポートと自動エクスポートのポリシーがすべて有効になっている状態で、ファイルシステムと S3 の両方でファイルを変更すると、ファイルシステム内のファイルまたは S3 内のオブジェクトのいずれかが他方で上書きされる可能性があります。ある場所で後から編集しても、別の場所で行った以前の編集が上書きされない場合があります。ファイルシステムと S3 バケットの両方で同じファイルを変更する場合は、このような競合を防ぐためにアプリケーションレベルの調整を行う必要があります。FSx for Lustre では、複数の場所での競合する書き込みを防止できません。

インポートポリシーでは、リンクされた S3 バケットの内容が変更されたときに、FSx for Lustre がファイルシステムをどのように更新するかを指定します。データリポジトリの関連付けには、次のいずれかのインポートポリシーがあります。

- 新規 — FSx for Lustre は、リンクされた S3 データリポジトリに新しいオブジェクトが追加された場合にのみ、ファイルとディレクトリのメタデータを自動的に更新します。
- 変更済み - FSx for Lustre は、データリポジトリ内の既存のオブジェクトが変更された場合にのみ、ファイルとディレクトリのメタデータを自動的に更新します。
- 削除済み — FSx for Lustre は、データリポジトリ内のオブジェクトが削除された場合にのみ、ファイルとディレクトリのメタデータを自動的に更新します。
- 新規、変更済み、削除済みの任意の組み合わせ - FSx for Lustre は、S3 データリポジトリで指定されたアクションのいずれかが発生した場合に、ファイルとディレクトリのメタデータを自動的に更新します。例えば、S3 リポジトリでオブジェクトが [新規] に追加されたとき、または [削除済み] から削除されたときにファイルシステムが更新され、オブジェクトが変更されたときには更新されないように指定できます。
- ポリシーの設定なし - FSx for Lustre は、S3 データリポジトリにオブジェクトが追加、変更されたとき、または S3 データリポジトリから削除されたときに、S3 データリポジトリのメタデータを更新しません。インポートポリシーを設定しない場合、データリポジトリの関連付けの自動インポートは無効になります。「データリポジトリのタスクを使用して変更をインポートする」で説明されているように、データリポジトリのインポートタスクを使用して、メタデータの変更を手動でインポートできます。

> **Important**
>
> 自動インポートは、次の S3 アクションはリンクされている FSx for Lustre ファイルシステムに同期しません。
>
> - S3 オブジェクトライフサイクルの有効期限を使用したオブジェクトの削除
> - バージョニングが有効なバケット内の現在のオブジェクトの永久削除
> - バージョニングが有効なバケット内のオブジェクトの削除キャンセル

インポートポリシーを [新規]、[変更済み]、[削除済み] に設定することをお勧めします。このポリシーにより、リンクされた S3 データリポジトリで行われたすべての更新が、ファイルシステムに自動的にインポートされます。

リンクされた S3 バケットの変更に基づいてファイルシステムのファイルとディレクトリのリストを更新するようにインポートポリシーを設定すると、FSx for Lustre はリンクされた S3 バケットにイベント通知設定を作成します。イベント通知設定には FSx という名前が付けられています。S3

バケットの FSx イベント通知設定を変更または削除しないでください。これを行うと、更新されたファイルとディレクトリのメタデータがファイルシステムに自動的にインポートされなくなります。

FSx for Lustre がリンクされた S3 バケットで変更されたファイルリストを更新した場合、ファイルの書き込みがロックされていても、ローカルファイルは更新されたバージョンで上書きされます。

FSx for Lustre は、ファイルシステムを更新するために最善を尽くします。FSx for Lustre は、次の状況ではファイルシステムを更新できません。

- FSx for Lustre に、変更された、または新しい S3 オブジェクトを開くためのアクセス許可がない場合です。この場合、FSx for Lustre はオブジェクトをスキップして続行します。DRA のライフサイクルステータスは影響を受けません。
- FSx for Lustre に、`GetBucketAcl` 用などのバケットレベルのアクセス許可がない場合です。この場合、データリポジトリのライフサイクルの状態が [Misconfigured] (設定ミス) になります。詳細については、「データリポジトリの関連付けのライフサイクル状態」を参照してください。
- リンクされた S3 バケットの FSx イベント通知設定が削除または変更された場合。この場合、データリポジトリのライフサイクルの状態が [Misconfigured] (設定ミス) になります。詳細については、「データリポジトリの関連付けのライフサイクル状態」を参照してください。

CloudWatch Logs への ログ記録を有効にして、自動的にインポートできなかったファイルやディレクトリに関する情報をログに記録することをお勧めします。ログ内の警告とエラーには、失敗の理由に関する情報が含まれています。詳細については、「データリポジトリのイベントログ」を参照してください。

前提条件

FSx for Lustre がリンクされた S3 バケットから新規、変更済み、または削除済みのファイルを自動的にインポートするには、次の条件が必要です。

- ファイルシステムとそれにリンクされたS3バケットは同じ AWS リージョンにあります。
- S3 バケットには、誤って設定された [ライフサイクル状態] はありません。詳細については、「データリポジトリの関連付けのライフサイクル状態」を参照してください。
- アカウントには、リンクされた S3 バケットでイベント通知を設定および受信するために必要なアクセス許可があります。

サポートされているファイル変更のタイプ

FSx for Lustre は、リンクされた S3 バケットで発生したファイルとディレクトリへの、次のような変更のインポートをサポートしています。

- ファイル内容の変更
- ファイルまたはディレクトリのメタデータの変更
- シンボリックリンクターゲットまたはメタデータの変更。
- ファイルおよびディレクトリの削除 (ファイルシステム内のディレクトリに対応するリンクされた S3 バケット内のオブジェクト、つまりキー名がスラッシュで終わるオブジェクトを削除すると、FSx for Lustre はファイルシステム上の対応するディレクトリが空の場合にのみ、それを削除します)

インポート設定の更新

データリポジトリの関連付けを作成する際に、リンクされた S3 バケットのファイルシステムのインポート設定を設定できます。詳細については、「S3 バケットへのリンクの作成」を参照してください。

また、インポートポリシーを含め、いつでもインポート設定を更新できます。詳細については、「データリポジトリの関連付け設定の更新」を参照してください。

自動インポートのモニタリング

S3 バケット内の変化率が自動インポートで処理可能な変化率を超えた場合、FSx for Lustre ファイルシステムにインポートされる当該メタデータの変更に遅延が生じます。その場合、`AgeOfOldestQueuedMessage` メトリクスを使用して、自動インポートによる処理を待機している最も古い変更の経過時間をモニタリングできます。このメトリクスの詳細については、「FSx for Lustre S3 リポジトリメトリクス」を参照してください。

メタデータの変更のインポートの遅延が (`AgeOfOldestQueuedMessage` メトリクスを使用して測定される) 14 日を超えるた場合、自動インポートによって処理されていない S3 バケット内の変更はファイルシステムにインポートされません。さらに、データリポジトリの関連付けのライフサイクルが MISCONFIGURED とマークされ、自動インポートが停止します。自動エクスポートを有効にしている場合、自動エクスポートは引き続き FSx for Lustre ファイルシステムの変更をモニタリングします。ただし、追加の変更は FSx for Lustre ファイルシステムから S3 に同期されません。

データリポジトリの関連付けを MISCONFIGRED のライフサイクル状態から AVAILABLE のライフサイクル状態に戻すには、データリポジトリの関連付けを更新する必要があります。デー

タリポジトリの関連付けは、update-data-repository-association CLI コマンド (または対応する UpdateDataRepositoryAssociation API オペレーション) を使用して更新できます。唯一必要なリクエストパラメータは、更新するデータリポジトリの関連付けの `AssociationID` だけです。

データリポジトリの関連付けのライフサイクル状態が AVAILABLE に変わると、自動インポート (および有効化されている場合は自動エクスポート) が再起動されます。再起動すると、自動エクスポートは S3 に対するファイルシステムの変更の同期を再開します。インポートされていない FSx for Lustre ファイルシステムまたはデータリポジトリの関連付けが構成が間違っていた状態の FSx for Lustre ファイルシステムに対して S3 内の新規オブジェクトと変更されたオブジェクトのメタデータを同期させるには、データリポジトリのインポートタスクを実行します。インポートデータリポジトリタスクでは、S3 バケット内の削除は FSx for Lustre ファイルシステムと同期されません。S3 をファイルシステムと (削除を含めて) 完全に同期する場合は、ファイルシステムを再作成する必要があります。

メタデータの変更のインポートの遅延が 14 日を超えないようにするために、`AgeOfOldestQueuedMessage` メトリクスにアラームを設定し、`AgeOfOldestQueuedMessage` メトリクスがアラームしきい値を超えた場合に S3 バケット内のアクティビティを減らすことをお勧めします。最大数の変更を S3 から継続的に送信する単一のシャードで S3 バケットに接続されている FSx for Lustre ファイルシステムで自動インポートのみを実行している場合、自動インポートでは 14 日以内に 7 時間分の S3 変更のバックログを処理できます。

さらに、自動インポートが 14 日で処理できるよりも多くの変更が 1 つの S3 アクションで生成されることがあります。このような種類のアクションの例には、S3 への AWS Snowball アップロードや大規模な削除などがあります。S3 バケットで行った大規模な変更を FSx for Lustre ファイルシステムと同期させる場合、自動インポートの変更が 14 日を超えないようにするには、ファイルシステムを削除し、S3 の変更が完了した後に再作成する必要があります。

`AgeOfOldestQueuedMessage` メトリクスが増大している場合、S3 バケットの `GetRequests`、`PutRequests`、`PostRequests`、`DeleteRequests` メトリクスを参照して、変化率や自動インポートに送信される変更の数を増加させるアクティビティ変更を確認してください。利用可能な S3 メトリクスについては、「Amazon S3 ユーザーガイド」の「Amazon S3 のモニタリング」を参照してください。

使用可能なすべての FSx for Lustre メトリクスの一覧については、「Amazon CloudWatch によるモニタリング」を参照してください。

データリポジトリのタスクを使用して変更をインポートする

データリポジトリのインポートタスクでは、S3 データリポジトリに新規または変更されたオブジェクトのメタデータをインポートし、S3 データリポジトリ内の新しいオブジェクトに対し、新規のファイルまたはディレクトリのリストを作成します。データリポジトリで変更されたオブジェクトについては、対応するファイルまたはディレクトリのリストが新しいメタデータで更新されます。データリポジトリから削除されたオブジェクトに対するアクションは実行されません。

Amazon FSx コンソールと CLI を使用してメタデータの変更をインポートするには、以下の手順に従います。複数の DRA に対して 1 つのデータリポジトリタスクを使用できることに注意してください。

メタデータの変更をインポートするには (コンソール)

1. Amazon FSx コンソール (https://console.aws.amazon.com/fsx/) を開きます。
2. ナビゲーションペインで [ファイルシステム] をクリックし、Lustre ファイルシステムを選択します。
3. [Data repository] (データリポジトリ) タブを選択します。
4. [Data repository associations] (データリポジトリ関連) ペインで、インポートタスクを作成する対象のデータリポジトリの関連付けを選択します。
5. [Actions] (アクション) メニューから、[Import Task] (タスクのインポート) を選択します。ファイルシステムがデータリポジトリにリンクされていない場合、この選択は利用できません。[Create import data repository task] (データリポジトリのインポートタスクの作成) ページが表示されます。
6. (オプション) [Data repository paths to import] (インポートするデータリポジトリパス) のディレクトリまたはファイルへのパスを指定することで、リンクされた S3 バケットからインポートするディレクトリまたはファイルを最大 32 個指定します。

   > **Note**
   >
   > 指定したパスが有効でない場合、タスクは失敗します。

7. (オプション) [Completion report] (完了レポート) 直下の [Enable] (有効化) をクリックして、タスクの完了後にタスク完了レポートを生成します。[task completion report] (タスク完了レポート) に、[Report scope] (レポートスコープ) に示されている範囲を満たすタスクによって処理されるファイルの詳細が表示されます。Amazon FSx がレポートを配信する場所を指定するに

は、[Report path] (レポートパス) にリンクされた S3 データリポジトリの相対パスを入力します。

8. [Create] (作成) を選択します。

   [File systems] (ファイルシステム) ページの上部にある通知には、先ほど作成したタスクが表示されます。

タスクのステータスと詳細を表示するには、ファイルシステム用の [Data Repository] (データリポジトリ) タブの [Data Repository Tasks] (データリポジトリタスク) ペインを下にスクロールします。デフォルトのソート順では、最新のタスクがリストの最上部に表示されます。

このページからタスクサマリーを表示するには、先ほど作成したタスクの [Task ID] (タスク ID) を選択します。[Summary] (概要) タスクのページが表示されます。

メタデータの変更をインポートするには (CLI)

- `create-data-repository-task` CLI コマンドを使用して FSx for Lustre ファイルシステムにメタデータの変更をインポートします。対応する API オペレーションは `CreateDataRepositoryTask` です。

  ```
  $ aws fsx create-data-repository-task \
      --file-system-id fs-0123456789abcdef0 \
      --type IMPORT_METADATA_FROM_REPOSITORY \
      --paths s3://bucketname1/dir1/path1 \
      --report Enabled=true,Path=s3://bucketname1/dir1/
  path1,Format=REPORT_CSV_20191124,Scope=FAILED_FILES_ONLY
  ```

  データリポジトリタスクが正常に作成されると、Amazon FSx はタスクの説明を JSON として返します。

リンクされたデータリポジトリからメタデータをインポートするタスクを作成した後、データリポジトリのインポートタスクのステータスを確認できます。データリポジトリタスクを表示する方法の詳細については、「データリポジトリタスクへのアクセス」を参照してください。

ファイルシステムへのファイルのプリロード

オプションで、個々のファイルまたはディレクトリの内容をファイルシステムに事前ロードできます。

HSM コマンドを使用したファイルのインポート

Amazon FSx は、ファイルが最初にアクセスされたときに Simple Storage Service (Amazon S3) データリポジトリからデータをコピーします。このアプローチにより、ファイルへの最初の読み取りまたは書き込みにはわずかなレイテンシーが発生します。アプリケーションがこのレイテンシーの影響を受けやすく、アプリケーションがアクセスする必要があるファイルやディレクトリがわかっている場合は、オプションで、個々のファイルまたはディレクトリのコンテンツをプリロードできます。以下のように `hsm_restore` コマンドを実行します。

`hsm_action` コマンド (`lfs` ユーザーユーティリティで発行される) を使用して、ファイルの内容がファイルシステムへのロードが完了したことを確認します。`NOOP` の戻り値は、ファイルが正常にロードされたことを示します。ファイルシステムがマウントされたコンピューティングインスタンスから次のコマンドを実行します。*`## / ## / ####`* ファイルシステムにプリロードするファイルのパスを使用して置き換えます。

```
sudo lfs hsm_restore path/to/file
sudo lfs hsm_action path/to/file
```

次のコマンドを使用して、ファイルシステム全体またはファイルシステム内のディレクトリ全体をプリロードできます。(末尾にアンパサンドをつけると、コマンドはバックグラウンドプロセスとして実行されます。) 複数のファイルのプリロードを同時にリクエストすると、Amazon FSx はファイルを Simple Storage Service (Amazon S3) データリポジトリから並行してロードします。ファイルが既にファイルシステムにロードされている場合、`hsm_restore` コマンドはそのファイルを再ロードしません。

```
nohup find local/directory -type f -print0 | xargs -0 -n 1 -P 8 sudo lfs hsm_restore &
```

> **Note**
>
> リンクされた S3 バケットがファイルシステムより大きい場合は、すべてのファイルメタデータをファイルシステムにインポートできるはずです。ただし、ファイルシステムの残りのストレージ領域に収まる実際のファイルデータだけを読み込むことができます。ファイルシステムにストレージが残っていないときにファイルデータにアクセスしようとすると、エラーが発生します。この場合、必要に応じてストレージ容量を増やすことができます。詳細については、「ストレージ容量の管理」を参照してください。

検証ステップ

以下に示す bash スクリプトを実行して、アーカイブ (リリース) 状態にあるファイルまたはオブジェクトの数を検出できます。

特に多数のファイルがあるファイル システム全体でスクリプトのパフォーマンスを向上させるために、CPU スレッドは /proc/cpuproc ファイルに基づいて自動的に決定されます。つまり、vCPU 数の多い Amazon EC2 インスタンスでは、パフォーマンスが向上します。

1. Bash スクリプトをセットアップします。

```
#!/bin/bash

# Check if a directory argument is provided
if [ $# -ne 1 ]; then
    echo "Usage: $0 /path/to/lustre/mount"
    exit 1
fi

# Set the root directory from the argument
ROOT_DIR="$1"

# Check if the provided directory exists
if [ ! -d "$ROOT_DIR" ]; then
    echo "Error: Directory $ROOT_DIR does not exist."
    exit 1
fi

# Automatically detect number of CPUs and set threads
if command -v nproc &> /dev/null; then
    THREADS=$(nproc)
elif [ -f /proc/cpuinfo ]; then
    THREADS=$(grep -c ^processor /proc/cpuinfo)
else
    echo "Unable to determine number of CPUs. Defaulting to 1 thread."
    THREADS=1
fi

# Output file
OUTPUT_FILE="released_objects_$(date +%Y%m%d_%H%M%S).txt"

echo "Searching in $ROOT_DIR for all released objects using $THREADS threads"
echo "This may take a while depending on the size of the filesystem..."
```

```
# Find all released files in the specified lustre directory using parallel
# If you  get false positives for file names/paths that include the word
 'released',
# you can grep 'released exists archived' instead of just 'released'
time sudo lfs find "$ROOT_DIR" -type f | \
parallel --will-cite -j "$THREADS" -n 1000 "sudo lfs hsm_state {} | grep released"
 > "$OUTPUT_FILE"

echo "Search complete. Released objects are listed in $OUTPUT_FILE"
echo "Total number of released objects: $(wc -l <"$OUTPUT_FILE")"
```

2. スクリプトを実行可能にします:

```
$ chmod +x find_lustre_released_files.sh
```

3. このスクリプトを以下の例にあるように、実行します:

```
$ ./find_lustre_released_files.sh /fsx1/sample
Searching in /fsx1/sample for all released objects using 16 threads
This may take a while depending on the size of the filesystem...
real 0m9.906s
user 0m1.502s
sys 0m5.653s
Search complete. Released objects are listed in
 released_objects_20241121_184537.txt
Total number of released objects: 30000
```

リリースされたオブジェクトが存在する場合は、以下の例のように、目的のディレクトリで一括復元を実行して、ファイルを S3 から FSx for Lustre に取り込みます:

```
$ DIR=/path/to/lustre/mount
$ nohup find $DIR -type f -print0 | xargs -0 -n 1 -P 8 sudo lfs hsm_restore &
```

数百万のファイルがある場合、hsm_restore には時間がかかることに留意してください。

データリポジトリへの変更のエクスポート

データへの変更および POSIX メタデータの変更を、FSx for Lustre ファイルシステムからリンクされたデータリポジトリにエクスポートできます。関連する POSIX メタデータには、所有権、許可、およびタイムスタンプが含まれます。

ファイルシステムから変更をエクスポートするには、次のいずれかの方法を使用します。

- ファイルシステムを設定して、リンクされたデータリポジトリに新規、変更済み、または削除済みのファイルを自動的にエクスポートできるようにします。詳細については、「S3 バケットに更新を自動的にエクスポートする」を参照してください。
- オンデマンドのデータリポジトリのエクスポートタスクを使用します。詳細については、「データリポジトリのタスクを使用した変更のエクスポート」を参照してください

データリポジトリの自動エクスポートタスクとエクスポートタスクは同時に実行できません。

> **Important**
>
> 対応するオブジェクトが S3 Glacier Flexible Retrieval に保存されている場合、自動エクスポートではファイルシステム上の以下のメタデータオペレーションは S3 と同期されません。
>
> - chmod
> - chown
> - rename

データリポジトリの関連付けで自動エクスポートを有効にすると、ファイルデータとメタデータが作成、変更、削除された場合に、ファイルシステムによってそれらが自動的にエクスポートされます。データリポジトリのエクスポートタスクを使用してファイルまたはディレクトリをエクスポートする場合、最後のエクスポート以降に作成または変更されたデータファイルとメタデータのみが、ファイルシステムによってエクスポートされます。

自動エクスポートおよびデータリポジトリのエクスポートタスクの両方で、POSIX メタデータがエクスポートされます。詳細については、「データリポジトリの POSIX メタデータのサポート」を参照してください。

Important

- FSx for Lustre が S3 バケットにデータをエクスポートするには、UTF-8 互換形式で保存されている必要があります。
- S3 オブジェクトキーの最大長は 1,024 バイトです。FSx for Lustre では、対応する S3 オブジェクトキーが 1,024 バイトを超えるファイルはエクスポートされません。

Note

自動エクスポートおよびデータリポジトリのエクスポートタスクによって作成されたすべてのオブジェクトは、S3 標準ストレージクラスを使用して書き込まれます。

トピック

S3 バケットに更新を自動的にエクスポートする

ファイルシステムでファイルが追加、変更、または削除されるときに、リンクされた S3 バケットの内容を自動的に更新するように FSx for Lustre ファイルシステムを設定できます。FSx for Lustre は、ファイルシステムの変更に応じて S3 内のオブジェクトを作成、更新、削除します。

Note

自動エクスポートは FSx for Lustre 2.10 ファイルシステムと `Scratch 1` ファイルシステムでは使用できません。

ファイルシステム AWS リージョン と同じ または別の にあるデータリポジトリにエクスポートできます AWS リージョン。

データリポジトリの関連付けを作成するときに自動エクスポートを設定し、FSx マネジメントコンソール、、 AWS CLIおよび AWS API を使用していつでも自動エクスポート設定を更新できます。

> **Important**
>
> - すべての自動エクスポートポリシーが有効で、自動インポートが無効になっているファイルシステムでファイルが変更された場合、そのファイルの内容は常に S3 の対応するオブジェクトにエクスポートされます。オブジェクトがターゲットの場所に既に存在する場合、そのオブジェクトは上書きされます。
> - 自動インポートと自動エクスポートのポリシーがすべて有効になっている状態で、ファイルシステムと S3 の両方でファイルを変更すると、ファイルシステム内のファイルまたは S3 内のオブジェクトのいずれかが他方で上書きされる可能性があります。ある場所で後から編集しても、別の場所で行った以前の編集が上書きされない場合があります。ファイルシステムと S3 バケットの両方で同じファイルを変更する場合は、このような競合を防ぐためにアプリケーションレベルの調整を行う必要があります。FSx for Lustre では、複数の場所での競合する書き込みを防止できません。

エクスポートポリシーでは、ファイルシステムの内容が変更されたときに、FSx for Lustre がリンクされた S3 バケットを更新する方法を指定します。データリポジトリの関連付けには、次のいずれかの自動エクスポートポリシーを設定できます。

- 新規 - FSx for Lustre は、ファイルシステム上に新しいファイル、ディレクトリ、またはシンボリックリンクが作成された場合にのみ、S3 データリポジトリを更新します。
- 変更済み - FSx for Lustre は、ファイルシステム内の既存のオブジェクトが変更された場合にのみ、S3 データリポジトリを自動的に更新します。ファイルコンテンツの変更については、S3 リポジトリに転送される前に、ファイルを閉じる必要があります。メタデータの変更 (名前の変更、所有権、許可、タイムスタンプ) は、オペレーションが完了すると、反映されます。名前を変更する場合 (移動を含む)、既存の (名前が変更された) S3 オブジェクトが削除され、新しい名前の新しい S3 オブジェクトが作成されます。
- 削除済み - FSx for Lustre は、ファイルシステムでファイル、ディレクトリ、またはシンボリックリンクが削除された場合にのみ、S3 データリポジトリを自動的に更新します。
- 新規、変更済み、削除済みの任意の組み合わせ - FSx for Lustre は、指定されたアクションのいずれかがファイルシステムで発生した場合に、自動的に S3 データリポジトリを更新します。例えば、ファイルシステムでオブジェクトが [新規] に追加されたとき、または [削除済み] から削除されたときに S3 リポジトリが更新され、ファイルが変更されたときには更新されないように指定できます。

- ポリシーの設定なし - FSx for Lustre は、ファイルシステムにファイルが追加、変更されたとき、またはファイルシステムから削除されたときに、S3 データリポジトリを自動的に更新しません。エクスポートポリシーを設定しない場合、自動エクスポートは無効になります。「データリポジトリのタスクを使用した変更のエクスポート」の説明に従って、データリポジトリのエクスポートタスクを使用して、変更を手動でエクスポートできます。

ほとんどのユースケースで、エクスポートポリシーを [新規]、[変更済み]、[削除済み] に設定することをお勧めします。このポリシーにより、ファイルシステムで行われたすべての更新が、リンクされた S3 データリポジトリに自動的にエクスポートされます。

CloudWatch Logs への ログ記録を有効にして、自動的にエクスポートできなかったファイルやディレクトリに関する情報をログに記録することをお勧めします。ログ内の警告とエラーには、失敗の理由に関する情報が含まれています。詳細については、「データリポジトリのイベントログ」を参照してください。

> **Note**
>
> エクスポートオペレーション中にアクセス時間 (atime) と変更時間 (mtime) が S3 と同期している間、これらのタイムスタンプのみを変更しても自動エクスポートはトリガーされません。ファイルコンテンツやその他のメタデータ (所有権やアクセス許可など) の変更のみが S3 への自動エクスポートをトリガーします。

エクスポート設定の更新

データリポジトリの関連付けを作成するときに、リンクされた S3 バケットへのファイルシステムのエクスポート設定を設定できます。詳細については、「S3 バケットへのリンクの作成」を参照してください。

また、エクスポートポリシーを含め、いつでもエクスポート設定を更新できます。詳細については、「データリポジトリの関連付け設定の更新」を参照してください。

自動インポートのモニタリング

Amazon CloudWatch に公開された一連のメトリクスを使用して、自動エクスポートが有効化されているデータリポジトリの関連付けをモニタリングできます。`AgeOfOldestQueuedMessage` メトリクスは、ファイルシステムに対して行われた後、S3 にまだエクスポートされていない最も古い更新の経過時間を表します。`AgeOfOldestQueuedMessage` が長期間にわたってゼロより大きい場合は、メッセージキューが減少するまで、ファイルシステムに対してアクティブに行われている変更

(特にディレクトリ名の変更) の数を一時的に減らすことをお勧めします。詳細については、「FSx for Lustre S3 リポジトリメトリクス」を参照してください。

> **Important**
>
> 自動エクスポートが有効化されているデータリポジトリの関連付けまたはファイルシステムを削除する場合、まず、AgeOfOldestQueuedMessage がゼロであること、つまりまだエクスポートされていない変更が存在しないことを確認する必要があります。データリポジトリの関連付けまたはファイルシステムを削除したときに AgeOfOldestQueuedMessage がゼロより大きい場合、まだエクスポートされていない変更は、リンクされた S3 バケットに到達していません。これを回避するには、AgeOfOldestQueuedMessage がゼロになるまで待ってから、データリポジトリの関連付けまたはファイルシステムを削除します。

データリポジトリのタスクを使用した変更のエクスポート

データリポジトリのエクスポートタスクは、ファイルシステムの新規または変更されたファイルをエクスポートします。ファイルシステムの新しいファイル用に、新しいオブジェクトが S3 に作成されます。ファイルシステムで変更されたファイル、またはメタデータが変更されたファイルの場合、S3 内の対応するオブジェクトは、新しいデータとメタデータを持つ新しいオブジェクトに置き換えられます。ファイルシステムから削除されたファイルに対するアクションは実行されません。

> **Note**
>
> データリポジトリのエクスポートタスクを使用する際は、以下の点に留意してください。
>
> - エクスポートするファイルを追加または除外するためのワイルドカードの使用はサポートされていません。
> - mv 操作を実行すると、移動後のターゲットファイルは、UID、GID、アクセス許可、または内容の変更がない場合でも、S3 にエクスポートされます。

Amazon FSx コンソールと CLI を使用して、ファイルシステム上のデータとメタデータの変更をリンクされた S3 バケットにエクスポートするには、以下の手順に従います。複数の DRA に対して 1 つのデータリポジトリタスクを使用できることに注意してください。

変更をエクスポートするには (コンソール)

1. Amazon FSx コンソール (https://console.aws.amazon.com/fsx/) を開きます。

2. ナビゲーションペインで [ファイルシステム] をクリックし、Lustre ファイルシステムを選択します。
3. [Data repository] (データリポジトリ) タブを選択します。
4. [Data repository associations] (データリポジトリ関連付け) ペインで、エクスポートタスクを作成する対象のデータリポジトリの関連付けを選択します。
5. [Actions] (アクション) で、[Export task] (タスクのエクスポート) を選択します。ファイルシステムが S3 のデータリポジトリにリンクされていない場合、この選択は使用できません。[Create export data repository task] (データリポジトリのエクスポートタスクの作成) ダイアログが表示されます。
6. (オプション) [File system paths to export] (エクスポートするためのファイルシステムパス) のディレクトリまたはファイルへのパスを指定して、Amazon FSx ファイルシステムからエクスポートするディレクトリまたはファイルを最大 32 個指定します。指定するパスは、ファイルシステムのマウントポイントに対する相対パスである必要があります。マウントポイントが `/mnt/fsx` で、`/mnt/fsx/path1` がエクスポートするファイルシステム上のディレクトリまたはファイルである場合、提供するパスは `path1` です。

   > **Note**
   >
   > 指定したパスが有効でない場合、タスクは失敗します。

7. (オプション) [Completion report] (完了レポート) 直下の [Enable] (有効化) をクリックして、タスクの完了後にタスク完了レポートを生成します。[task completion report] (タスク完了レポート) に、[Report scope] (レポートスコープ) に示されている範囲を満たすタスクによって処理されるファイルの詳細が表示されます。Amazon FSx がレポートを配信する場所を指定するには、ファイルシステムのリンクされた S3 データリポジトリの [Report path] (レポートパス) の相対パスを入力します。
8. [Create] (作成) を選択します。

   [File systems] (ファイルシステム) ページの上部にある通知には、先ほど作成したタスクが表示されます。

タスクのステータスと詳細を表示するには、ファイルシステム用の [Data Repository] (データリポジトリ) タブの [Data Repository Tasks] (データリポジトリタスク) ペインを下にスクロールします。デフォルトのソート順では、最新のタスクがリストの最上部に表示されます。

このページからタスクサマリーを表示するには、先ほど作成したタスクの [Task ID] (タスク ID) を選択します。[Summary] (概要) タスクのページが表示されます。

変更をエクスポートするには (CLI)

- FSx for Lustreファイルシステムのデータとメタデータの変更をエクスポートするには create-data-repository-task CLI コマンドを使用します。対応する API オペレーションは CreateDataRepositoryTask。

```
$ aws fsx create-data-repository-task \
    --file-system-id fs-0123456789abcdef0 \
    --type EXPORT_TO_REPOSITORY \
    --paths path1,path2/file1 \
    --report Enabled=true
```

次の例に示すように、データリポジトリタスクを正常に作成すると、Amazon FSx は JSON としてデータの説明を返します。

```
{
    "Task": {
        "TaskId": "task-123f8cd8e330c1321",
        "Type": "EXPORT_TO_REPOSITORY",
        "Lifecycle": "PENDING",
        "FileSystemId": "fs-0123456789abcdef0",
        "Paths": ["path1", "path2/file1"],
        "Report": {
            "Path":"s3://dataset-01/reports",
            "Format":"REPORT_CSV_20191124",
            "Enabled":true,
            "Scope":"FAILED_FILES_ONLY"
        },
        "CreationTime": "1545070680.120",
        "ClientRequestToken": "10192019-drt-12",
        "ResourceARN": "arn:aws:fsx:us-
east-1:123456789012:task:task-123f8cd8e330c1321"
    }
}
```

リンクされたデータリポジトリにデータをエクスポートするタスクを作成すると、データリポジトリのエクスポートタスクのステータスを確認できます。データリポジトリタスクを表示する方法の詳細については、「データリポジトリタスクへのアクセス」を参照してください。

HSM コマンドを使用したファイルのエクスポート

> **Note**
>
> FSx for Lustre ファイルシステムのデータおよびメタデータの変更を Simple Storage Service (Amazon S3) の耐久性のあるデータリポジトリにエクスポートするには、「S3 バケットに更新を自動的にエクスポートする」で説明されている自動エクスポート機能を使用します。「データリポジトリのタスクを使用した変更のエクスポート」の説明に従って、データリポジトリのエクスポートタスクを使用することもできます。

個々のファイルをデータリポジトリにエクスポートし、ファイルがデータリポジトリに正常にエクスポートされたことを確認するには、以下に示すコマンドを実行します。`states: (0x00000009) exists archived` の戻り値はのファイルが正常にエクスポートされたことを示します。

```
sudo lfs hsm_archive path/to/export/file
sudo lfs hsm_state path/to/export/file
```

> **Note**
>
> ルートユーザーとして、または sudo を使用して HSM コマンド (`hsm_archive` など) を実行する必要があります。

ファイルシステム全体またはファイルシステム内のディレクトリ全体をエクスポートするには、次のコマンドを実行します。複数のファイルを同時にエクスポートする場合、Amazon FSx for Lustre はファイルを Simple Storage Service (Amazon S3) データリポジトリに並行してエクスポートします。

```
nohup find local/directory -type f -print0 | xargs -0 -n 1 sudo lfs hsm_archive &
```

エクスポートが完了したかどうかを判定するには、次のコマンドを実行します。

```
find path/to/export/file -type f -print0 | xargs -0 -n 1 -P 8 sudo lfs hsm_state | awk
 '!/\<archived\>/ || /\<dirty\>/' | wc -l
```

ファイルを残さないでコマンドが返ると、エクスポートは完了です。

データリポジトリタスク

データリポジトリのインポートおよびエクスポートタスクを使用すると、FSx for Lustre ファイルシステムと Amazon S3 で耐久性のあるデータリポジトリ間のデータおよびメタデータの転送を管理できます。

[Data Repository Tasks] (データリポジトリタスク) は、FSx for Lustre ファイルシステムと S3 上のデータリポジトリ間のデータとメタデータの転送を最適化します。これを行う方法の 1 つは、Amazon FSx ファイルシステムとそのリンクされたデータリポジトリ間の変更を追跡することです。また、並列転送技術を用いることで、データを最大で数百 GB /秒の速度で転送することも可能です。データリポジトリタスクを作成および表示するには AWS CLI、Amazon FSx コンソール、、および Amazon FSx API を使用します。

データリポジトリタスクは、所有権、許可、タイムスタンプなど、ファイルシステムのポータブルオペレーティングシステムインターフェイス (POSIX) メタデータを維持します。タスクはこのメタデータを保持するため、FSx for Lustre ファイルシステムとそのリンクされたデータリポジトリ間のアクセスコントロールを実装および維持できます。

リリースデータリポジトリタスクを使用して、Amazon S3 にエクスポートされたファイルをリリースすることで、新しいファイル用にファイルシステムのスペースを解放することができます。リリースされたファイルの内容は削除されますが、リリースされたファイルのメタデータはファイルシステムに残ります。ユーザーやアプリケーションは、リリースされたファイルを再度読み込むことで引き続きアクセスできます。ユーザーまたはアプリケーションがリリースされたファイルを読み取ると、FSx for Lustre はファイルコンテンツを Amazon S3 から透過的に取得します。

データリポジトリタスクのタイプ

データリポジトリタスクには 3 つのタイプがあります。

- データリポジトリの [エクスポート] タスクは、Lustre ファイルシステムからリンクされた S3 バケットにエクスポートします。
- データリポジトリの [インポート] タスクは、リンクされた S3 バケットから Lustre ファイルシステムにインポートします。

- データリポジトリの [リリース] タスクは、リンクされている S3 バケットにエクスポートされたファイルを Lustre ファイルシステムからリリースします。

詳細については、「データリポジトリタスクの作成」を参照してください。

トピック

- タスクのステータスと詳細を理解する
- データリポジトリタスクの使用
- タスク完了レポートの使用
- データリポジトリタスク失敗のトラブルシューティング

タスクのステータスと詳細を理解する

データリポジトリタスクには、説明的な情報とライフサイクルステータスがあります。

タスクを作成した後、Amazon FSx コンソール、CLI、または API を使用して、データリポジトリタスクの次の詳細情報を表示できます。

- タスクタイプ :
  - `EXPORT_TO_REPOSITORY` はエクスポートタスクを示します。
  - `IMPORT_METADATA_FROM_REPOSITORY` はインポートタスクを示します。
  - `RELEASE_DATA_FROM_FILESYSTEM` はリリースタスクを示します。
- タスクが実行されたファイルシステム。
- タスクの作成時刻。
- タスクのステータス。
- タスクが処理されたファイルの総数。
- タスクが正常に処理されたファイルの総数。
- タスクの処理に失敗したファイルの総数。タスクステータスが FAILED (失敗) の場合、この値はゼロより大きくなります。障害ファイルに関する詳細情報は、タスク完了レポートで確認できます。詳細については、「タスク完了レポートの使用」を参照してください。
- タスクが開始された時刻。
- タスクのステータスが最後に更新された時刻。タスクのステータスは 30 秒ごとに更新されます。

データリポジトリタスクのステータスは、次のいずれかです。

- [PENDING] (保留中) は、Amazon FSx がタスクを開始していないことを示します。
- [EXECUTING] (実行中) は、Amazon FSx がタスクを処理中であることを示します。
- [FAILED] (失敗) は、Amazon FSx でタスクが正常に処理されなかったことを示します。例えば、タスクの処理に失敗したファイルがある可能性があります。タスクの詳細は、障害に関する詳細情報を提供します。障害タスクの詳細については、「データリポジトリタスク失敗のトラブルシューティング」を参照してください。
- [SUCCEEDED] (成功) は、Amazon FSx がタスクを正常に完了したことを示します。
- [CANCELED] (キャンセル済み) は、タスクがキャンセルされて完了していないことを示します。
- [CANCELING] (キャンセル中) は、Amazon FSx がタスクをキャンセル中であることを示します。

データリポジトリのタスク情報は、タスク終了後 14 日間保持されます。既存のデータリポジトリタスクへのアクセス方法の詳細については、「データリポジトリタスクへのアクセス」を参照してください。

データリポジトリタスクの使用

以下のセクションでは、データリポジトリタスクの管理に関する詳細情報について説明します。Amazon FSx コンソール、CLI、または API を使用して、データリポジトリタスクの作成、複製、詳細の表示、およびキャンセルを行うことができます。

トピック

データリポジトリタスクの作成

Amazon FSx コンソール、CLI、または API を使用してデータリポジトリタスクを作成できます。タスクを作成したら、コンソール、CLI、または API を使用してタスクの進行状況とステータスを確認できます。

データリポジトリタスクには、次の 3 種類のデータリポジトリタスクを作成できます。

- データリポジトリの [エクスポート] タスクは、Lustre ファイルシステムからリンクされた S3 バケットにエクスポートします。詳細については、「データリポジトリのタスクを使用した変更のエクスポート」を参照してください。
- データリポジトリの [インポート] タスクは、リンクされた S3 バケットから Lustre ファイルシステムにインポートします。詳細については、「データリポジトリのタスクを使用して変更をインポートする」を参照してください。
- データリポジトリの [リリース] タスクは、リンクされている S3 バケットにエクスポートされたファイルを Lustre ファイルシステムからリリースします。詳細については、「データリポジトリタスクを使用してファイルをリリースする」を参照してください。

タスクの複製

Amazon FSx コンソールで、既存のデータリポジトリタスクを複製できます。タスクを複製すると、既存のタスクの正確なコピーが インポートデータリポジトリタスクの作成 または エクスポートデータリポジトリタスクの作成 ページに表示されます。新しいタスクを作成して実行する前に、必要に応じてエクスポートまたはインポートするパスを変更できます。

> **Note**
>
> そのタスクの正確なコピーがすでに実行されている場合、重複タスクを実行するリクエストは失敗します。すでに実行されているタスクの正確なコピーには、エクスポートタスクの場合は同じファイルシステムパスが含まれ、インポートタスクの場合は同じデータリポジトリパスが含まれます。

タスクは、タスクの詳細ビュー、ファイルシステムの [Data Repository] (データリポジトリ) タブの [Data Repository Tasks] (データリポジトリタスク) ペイン、または [Data Repository Tasks] (データリポジトリタスク) ページから複製できます。

既存のタスクを複製するには

1. ファイルシステムの [Data Repository] (データリポジトリ) タブの [Data Repository Tasks] (データリポジトリタスク) ペインでタスクを選択します。
2. [Duplicate task] (タスクの複製) を選択します。選択したタスクのタイプに応じて、インポートデータリポジトリの作成 タスクまたは エクスポートデータリポジトリの作成 タスクページが表示されます。新しいタスクの設定はすべて、複製するタスクの設定と同じです。
3. インポート元またはエクスポート先のパスを変更または追加します。

4. [Create] (作成) を選択します。

データリポジトリタスクへのアクセス

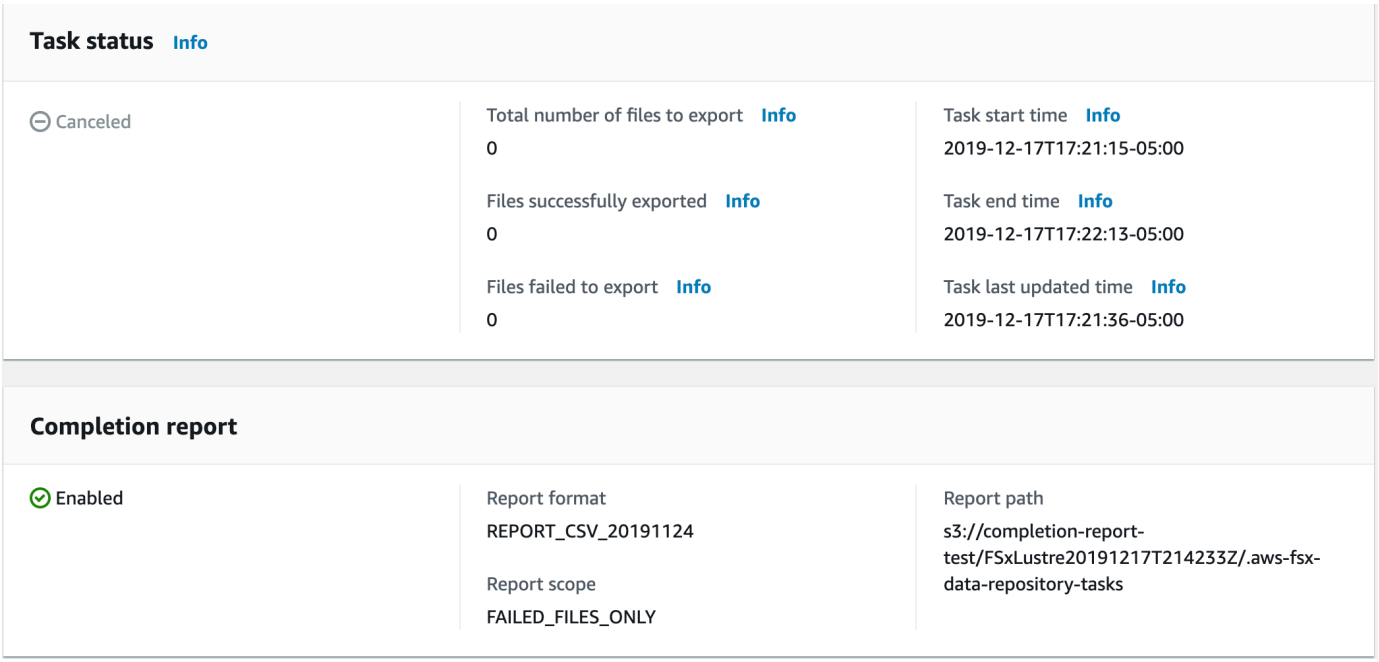
データリポジトリタスクを作成したら、Amazon FSx コンソール、CLI、API を使用して、タスク、およびアカウント内のすべての既存のタスクにアクセスできます。Amazon FSx は、次の詳細なタスク情報を提供します。

- 既存のすべてのタスク。
- 特定のファイルシステムに関するすべてのタスク。
- 特定のデータリポジトリ関連付けに関するすべてのタスク。
- 特定のライフサイクルステータスを持つすべてのタスク。タスクのライフサイクルステータス値の詳細については、「タスクのステータスと詳細を理解する」を参照してください。

以下に記載されているように、Amazon FSx コンソール、CLI、または API を使用して、アカウント内のすべての既存のデータリポジトリタスクにアクセスできます。

データリポジトリのタスクとタスクの詳細を表示するには (コンソール)

1. https://console.aws.amazon.com/fsx/ で Amazon FSx コンソールを開きます。
2. ナビゲーションページで、データリポジトリタスクを表示するファイルシステムを選択します。ファイルシステムの詳細ページが表示されます。
3. ファイルシステムの詳細ページで、[Data repository] (データリポジトリ) タブを選択します。このファイルシステムのタスクはすべて、[Data Repository Tasks] (データリポジトリタスク) パネルに表示されます。
4. タスクの詳細を表示するには、[データリポジトリタスク] パネルで [タスク ID] または [タスク名] を選択します。タスクの詳細ページが表示されます。



データリポジトリのタスクとタスクの詳細を取得するには (CLI)

Amazon FSx [describe-data-repository-tasks](#) CLI コマンドを使用すると、アカウント内のすべてのデータリポジトリタスクとその詳細を表示できます。[DescribeDataRepositoryTasks](#) は、同等の API コマンドです。

- アカウント内のすべてのデータリポジトリタスクオブジェクトを確認するには、次のコマンドを使用します。

```
aws fsx describe-data-repository-tasks
```

コマンドが成功すると、Amazon FSx は JSON 形式でレスポンスを返します。

```
{
    "DataRepositoryTasks": [
        {
            "Lifecycle": "EXECUTING",
            "Paths": [],
            "Report": {
                "Path":"s3://dataset-01/reports",
                "Format":"REPORT_CSV_20191124",
                "Enabled":true,
                "Scope":"FAILED_FILES_ONLY"
            },
```

```
            "StartTime": 1591863862.288,
            "EndTime": ,
            "Type": "EXPORT_TO_REPOSITORY",
            "Tags": [],
            "TaskId": "task-0123456789abcdef3",
            "Status": {
                "SucceededCount": 4255,
                "TotalCount": 4200,
                "FailedCount": 55,
                "LastUpdatedTime": 1571863875.289
            },
            "FileSystemId": "fs-0123456789a7",
            "CreationTime": 1571863850.075,
            "ResourceARN": "arn:aws:fsx:us-east-1:1234567890:task/
task-0123456789abcdef3"
        },
        {
            "Lifecycle": "FAILED",
            "Paths": [],
            "Report": {
                "Enabled": false,
            },
            "StartTime": 1571863862.288,
            "EndTime": 1571863905.292,
            "Type": "EXPORT_TO_REPOSITORY",
            "Tags": [],
            "TaskId": "task-0123456789abcdef1",
            "Status": {
                "SucceededCount": 1153,
                "TotalCount": 1156,
                "FailedCount": 3,
                "LastUpdatedTime": 1571863875.289
            },
            "FileSystemId": "fs-0123456789abcdef0",
            "CreationTime": 1571863850.075,
            "ResourceARN": "arn:aws:fsx:us-east-1:1234567890:task/
task-0123456789abcdef1"
        },
        {
            "Lifecycle": "SUCCEEDED",
            "Paths": [],
            "Report": {
                "Path":"s3://dataset-04/reports",
                "Format":"REPORT_CSV_20191124",
```

```
                "Enabled":true,
                "Scope":"FAILED_FILES_ONLY"
            },
            "StartTime": 1571863862.288,
            "EndTime": 1571863905.292,
            "Type": "EXPORT_TO_REPOSITORY",
            "Tags": [],
            "TaskId": "task-04299453935122318",
            "Status": {
                "SucceededCount": 258,
                "TotalCount": 258,
                "FailedCount": 0,
                "LastUpdatedTime": 1771848950.012,
            },
            "FileSystemId": "fs-0123456789abcdef0",
            "CreationTime": 1771848950.012,
            "ResourceARN": "arn:aws:fsx:us-east-1:1234567890:task/
task-0123456789abcdef0"
        }
    ]
}
```

ファイルシステムによるタスクの表示

以下に記載されているように、Amazon FSx コンソール、CLI、または API を使用して、特定のファイルシステムのすべてのタスクを表示できます。

ファイルシステム別にタスクを表示するには (コンソール)

1. ナビゲーションペインで [File systems] (ファイルシステム) を選択します。[File systems] (ファイルシステム) ページが表示されます。
2. データリポジトリタスクを表示するファイルシステムを選択します。ファイルシステムの詳細ページが表示されます。
3. ファイルシステムの詳細ページで、[Data repository] (データリポジトリ) タブを選択します。このファイルシステムのタスクはすべて、[Data Repository Tasks] (データリポジトリタスク) パネルに表示されます。

ファイルシステム別にタスクを取得するには (CLI)

- 次のコマンドを使用して、ファイルシステム fs-0123456789abcdef0 のすべてのデータリポジトリタスクを表示します。

```
aws fsx describe-data-repository-tasks \
    --filters Name=file-system-id,Values=fs-0123456789abcdef0
```

コマンドが成功すると、Amazon FSx は JSON 形式でレスポンスを返します。

```
{
    "DataRepositoryTasks": [
        {
            "Lifecycle": "FAILED",
            "Paths": [],
            "Report": {
                "Path":"s3://dataset-04/reports",
                "Format":"REPORT_CSV_20191124",
                "Enabled":true,
                "Scope":"FAILED_FILES_ONLY"
            },
            "StartTime": 1571863862.288,
            "EndTime": 1571863905.292,
            "Type": "EXPORT_TO_REPOSITORY",
            "Tags": [],
            "TaskId": "task-0123456789abcdef1",
            "Status": {
                "SucceededCount": 1153,
                "TotalCount": 1156,
                "FailedCount": 3,
                "LastUpdatedTime": 1571863875.289
            },
            "FileSystemId": "fs-0123456789abcdef0",
            "CreationTime": 1571863850.075,
            "ResourceARN": "arn:aws:fsx:us-east-1:1234567890:task/
task-0123456789abcdef1"
        },
        {
            "Lifecycle": "SUCCEEDED",
            "Paths": [],
            "Report": {
                "Enabled": false,
```

```
            },
            "StartTime": 1571863862.288,
            "EndTime": 1571863905.292,
            "Type": "EXPORT_TO_REPOSITORY",
            "Tags": [],
            "TaskId": "task-0123456789abcdef0",
            "Status": {
                "SucceededCount": 258,
                "TotalCount": 258,
                "FailedCount": 0,
                "LastUpdatedTime": 1771848950.012,
            },
            "FileSystemId": "fs-0123456789abcdef0",
            "CreationTime": 1771848950.012,
            "ResourceARN": "arn:aws:fsx:us-east-1:1234567890:task/
task-0123456789abcdef0"
        }
    ]
}
```

データリポジトリタスクのキャンセル

データリポジトリタスクを、保留中または実行中状態のときにキャンセルできます。タスクをキャンセルすると、次のことが発生します。

- Amazon FSx は、処理対象のキューにあるファイルを処理しません。
- Amazon FSx は、現在処理中のファイルの処理を続行します。
- Amazon FSx は、タスクが既に処理したファイルを元に戻せません。

データリポジトリタスクをキャンセルするには (コンソール)

1. https://console.aws.amazon.com/fsx/ で Amazon FSx コンソールを開きます。
2. データリポジトリタスクをキャンセルするファイルシステムをクリックします。
3. [Data Repository] (データリポジトリ) タブを開き、下にスクロールして データリポジトリタスク パネルを表示します。
4. キャンセルしたいタスクの [Task ID] (タスク ID) または [Task name] (タスク名) を選択します。
5. [Cancel task] (タスクのキャンセル) を選択してタスクをキャンセルします。
6. タスク ID を入力して、キャンセルリクエストを確認します。

データリポジトリタスクをキャンセルするには (CLI)

Amazon FSx `cancel-data-repository-task` CLI コマンドを使用して、タスクをキャンセルします。`CancelDataRepositoryTask` は、同等の API コマンドです。

- 次のコマンドを使用して、データリポジトリのタスクをキャンセルします。

```
aws fsx cancel-data-repository-task \
    --task-id task-0123456789abcdef0
```

コマンドが成功すると、Amazon FSx は JSON 形式でレスポンスを返します。

```
{
    "Status": "CANCELING",
    "TaskId": "task-0123456789abcdef0"
}
```

タスク完了レポートの使用

タスク完了レポートは、データリポジトリのエクスポート、インポート、またはリリースタスクの結果に関する詳細を示します。レポートには、レポートのスコープに一致するタスクで処理されたファイルの結果が含まれます。`Enabled` パラメータを使用して、タスクに関するレポートを生成するかどうかを指定できます。

Amazon FSx は、タスクのレポートを有効にするときに指定したパスを使用して、Simple Storage Service (Amazon S3) 内のファイルシステムのリンクされたデータリポジトリにレポートを配信します。レポートのファイル名は、インポートタスクの場合は `report.csv`、エクスポートまたはリリースタスクの場合は `failures.csv` です。

レポート形式は、`FilePath`、`FileStatus`、および `ErrorCode` の 3 つのフィールドを持つカンマ区切り値 (CSV) ファイルです。

レポートは、RFC-4180 形式のエンコードを使用して次のようにエンコードされます。

- 次の文字のいずれかで始まるパスは、一重引用符で囲まれています: `@ + - =`
- 次の文字の少なくとも 1 つを含む文字列は、二重引用符で囲まれています: `" ,`
- すべての二重引用符は、追加の二重引用符で工スケープされます。

レポートのエンコードの例をいくつか示します。

- `@filename.txt` が `"""@filename.txt"""` になります
- `+filename.txt` が `"""+filename.txt"""` になります
- `file,name.txt` が `"file,name.txt"` になります
- `file"name.txt` が `"file""name.txt"` になります

RFC-4180 エンコードの詳細については、「RFC-4180 - カンマ区切り値 (CSV) ファイルの一般形式と MIME タイプ」を参照してください。

次に、失敗したファイルのみを含むタスク完了レポートに表示される情報の例を示します。

```
myRestrictedFile,failed,S3AccessDenied
dir1/myLargeFile,failed,FileSizeTooLarge
dir2/anotherLargeFile,failed,FileSizeTooLarge
```

失敗したタスクとその解決方法の詳細については、「データリポジトリタスク失敗のトラブルシューティング」を参照してください。

データリポジトリタスク失敗のトラブルシューティング

CloudWatch Logs へのログを有効にすると、データリポジトリタスクを使用してファイルをインポートまたはエクスポートする際に発生した、任意の障害に関する情報をログに記録できます。CloudWatch Logs のイベントログの詳細については、「データリポジトリのイベントログ」を参照してください。

データリポジトリタスクが失敗した場合、Amazon FSx が処理に失敗したファイルの数は、コンソールの タスクステータス ページの エクスポートに失敗したファイル で確認できます。または、CLI や API を使用してタスクの `Status: FailedCount` プロパティを表示することもできます。この情報へのアクセスについては、「データリポジトリタスクへのアクセス」を参照してください。

データリポジトリタスクの場合、Amazon FSx は、完了レポートで失敗した特定のファイルやディレクトリに関する情報もオプションで提供します。タスク完了レポートには、障害が発生した Lustre ファイルシステム上のファイルまたはディレクトリパス、そのステータス、および失敗の理由が含まれます。詳細については、「タスク完了レポートの使用」を参照してください。

データリポジトリタスクは、以下に示すものを含む、いくつかの理由で失敗することがあります。

エラーコード	説明
FileSizeTooLarge	Amazon S3 でサポートされているオブジェクトの最大サイズは 5 TiB です。
InternalError	インポート、エクスポート、またはリリースタスクの Amazon FSx ファイルシステム内でエラーが発生しました。通常、このエラーコードは失敗したタスクが実行された Amazon FSx ファイルシステムが、失敗したライフサイクル状態にあることを意味します。この問題が発生すると、データ損失のために影響を受けるファイルを回復できないことがあります。それ以外の場合は、階層ストレージ管理 (HSM) コマンドを使用して、ファイルとディレクトリを S3 のデータリポジトリにエクスポートできます。詳細については、「HSM コマンドを使用したファイルのエクスポート」を参照してください。
OperationNotPermitted	リンクされている S3 バケットにファイルがエクスポートされていないため、Amazon FSx はファイルをリリースできませんでした。自動エクスポートまたはデータリポジトリのエクスポートタスクを使用して、ファイルが最初にリンクされた Amazon S3 バケットにエクスポートされるようにする必要があります。
PathSizeTooLong	エクスポートのパスが長すぎます。S3 でサポートされているオブジェクトキーの最大長は 1,024 文字です。
ResourceBusy	Amazon FSx は、ファイルシステムで別のクライアントによってアクセスされているため、ファイルをエクスポートまたはリリースできませんでした。ワークフローがファイルへの書き込

エラーコード	説明
	みを完了した後、データリポジトリタスクを再試行できます。

エラーコード	説明
S3AccessDenied	データリポジトリのエクスポートまたはインポートタスクに対する Simple Storage Service (Amazon S3) へのアクセスが拒否されました。 エクスポートタスクの場合、Amazon FSx ファイルシステムでは、S3 上のリンクされたデータリポジトリにエクスポートする S3:PutObject オペレーションを実行する許可が必要です。この許可は、AWSServiceRoleForFSxS3Access_ *fs-0123456789abcdef0* サービスにリンクされたロールで許可されています。詳細については、「Amazon FSx のサービスリンクロールの使用」を参照してください。 エクスポートタスクの場合、エクスポートタスクではファイルシステムの VPC の外側にデータが流れる必要があるため、ターゲットリポジトリに aws:SourceVpc または aws:SourceVpce IAM グローバル条件キーの 1 つを含むバケットポリシーがある場合に発生する可能性があります。 インポートタスクの場合、Amazon FSx ファイルシステムに S3 上のリンクされたデータリポジトリからインポートするため、S3:HeadObject および S3:GetObject オペレーションを実行する許可が必要です。 インポートタスクで、S3 バケットが AWS Key Management Service (SSE-KMS) に保存されているカスタマーマネージドキーによるサーバー側の暗号化を使用している場合は、 のポリシー設定に従う必要がありますサーバー側で暗号化された Simple Storage Service (Amazon S3) バケットの使用。

エラーコード	説明
	S3 バケットに、ファイルシステムにリンクされた S3 バケットアカウント AWS アカウントとは異なる からアップロードされたオブジェクトが含まれている場合、どのアカウントがアップロードしたかに関係なく、データリポジトリタスクで S3 メタデータを変更したり、S3 オブジェクトを上書きしたりできます。S3 バケットで、S3 オブジェクト所有権の機能を有効にすることをお勧めします。この機能を使用すると、アップロードに`-/-acl bucket-owner-full-control` 既定 ACL の提供を強制することで、他の がバケット AWS アカウント にアップロードする新しいオブジェクトの所有権を取得できます。S3 バケットで、バケット所有者優先 内のオプションを選択することにより、S3 オブジェクトの所有権を有効にします。詳細については、「Simple Storage Service (Amazon S3) ユーザーガイド」の「S3 オブジェクトの所有権を使用してアップロードされたオブジェクトの所有権をコントロールする」を参照してください。
`S3Error`	Amazon FSx で、`S3AccessDenied` ではない S3 に関連するエラーが発生しました。
`S3FileDeleted`	Amazon FSx は、ハードリンクファイルをエクスポートできませんでした。ソースファイルがデータリポジトリ内に存在しません。

エラーコード	説明
S3ObjectInUnsupportedTier	Amazon FSx が S3 Glacier または S3 Glacier Deep Archive ストレージクラスから、シンボリックリンク以外のオブジェクトを正常にインポートしました。FileStatus は、タスク完了レポートで succeeded with warning になります。データを取得するには、まず S3 Glacier Flexible Retrieval または S3 Glacier Deep Archive オブジェクトを復元し、hsm_restore コマンドを使用してオブジェクトをインポートする必要があることを、警告が示します。
S3ObjectNotFound	Amazon FSx は、データリポジトリに存在しないため、ファイルをインポートまたはエクスポートできませんでした。
S3ObjectPathNotPosixCompliant	Simple Storage Service (Amazon S3) オブジェクトは存在しますが、POSIX に準拠しているオブジェクトではないため、インポートできません。サポートされている POSIX メタデータについては、「データリポジトリの POSIX メタデータのサポート」を参照してください。
S3ObjectUpdateInProgressFromFileRename	自動エクスポートがファイルの名前変更を処理しているため、Amazon FSx はファイルをリリースできませんでした。ファイルをリリースする前に、エクスポートの自動名前変更プロセスを終了する必要があります。

エラーコード	説明
`S3SymlinkInUnsupportedTier`	Amazon FSx は、シンボリックリンクオブジェクトをインポートできませんでした。このオブジェクトは、S3 Glacier Flexible Retrieval や S3 Glacier Deep Archive ストレージクラスなど、サポートされていない Amazon S3 ストレージクラスにあります。`FileStatus` は、タスク完了レポートで `failed` になります。
`SourceObjectDeletedBeforeReleasing`	Amazon FSx は、ファイルがリリースされる前にデータリポジトリから削除されたため、ファイルシステムからファイルをリリースできませんでした。

ファイルのリリース

リリースデータリポジトリタスクは、FSx for Lustre ファイルシステムからファイルのデータをリリースして、新しいファイルのためにスペースを解放します。ファイルを解放すると、ファイルのリストとメタデータは保持されますが、そのファイルのコンテンツのローカルコピーは削除されます。ユーザーまたはアプリケーションがリリースされたファイルにアクセスすると、データはリンクされた Amazon S3 バケットからファイルシステムに自動的かつ透過的に再ロードされます。

> **Note**
>
> リリースデータリポジトリタスクは FSx for Lustre 2.10 ファイルシステムでは使用できません。

[リリースするファイルシステムパス] と [前回のアクセスからの最小期間] パラメータによって、リリースされるファイルが決まります。

- [リリースするファイルシステムパス]: リリースされるファイルのパスを指定します。
- [前回のアクセスからの最小期間]: その期間内にアクセスされなかったファイルがリリースされるように、期間を日単位で指定します。ファイルが前回アクセスされてからの期間は、リリースタスクの作成時刻と前回ファイルにアクセスした時刻 (`atime`、`mtime`、`ctime` の最大値) との差をとって計算されます。

ファイルパスに従ってファイルがリリースされるのは、ファイルが S3 にエクスポートされており、前回のアクセスからの期間が [前回のアクセスからの最小期間] の値よりも大きい場合のみです。[前回のアクセスからの最小期間] を `0` 日間と指定すると、前回のアクセスからの期間とは無関係にファイルがリリースされます。

> **Note**
>
> リリースするファイルを含めたり除外したりするためのワイルドカードの使用はサポートされていません。

リリースデータリポジトリタスクは、リンクされた S3 データリポジトリに既にエクスポートされているファイルのデータのみをリリースします。自動エクスポート機能、エクスポートデータリポジトリタスク、または HSM コマンドのいずれかを使用してデータを S3 にエクスポートできます。ファイルがデータリポジトリにエクスポートされたことを確認するには、次のコマンドを実行します。`states: (0x00000009) exists archived` の戻り値はのファイルが正常にエクスポートされたことを示します。

```
sudo lfs hsm_state path/to/export/file
```

> **Note**
>
> HSM コマンドは、ルートユーザーとして、または sudo を使用して実行する必要があります。

ファイルデータを定期的にリリースするには、Amazon EventBridge スケジューラを使用してリリースデータリポジトリタスクの繰返しをスケジュールします。詳細については、「Amazon EventBridge スケジューラユーザーガイド」の「Getting started with EventBridge Scheduler」を参照してください。

トピック

- データリポジトリタスクを使用してファイルをリリースする

データリポジトリタスクを使用してファイルをリリースする

Amazon FSx コンソールと CLI を使用してファイルシステムからファイルをリリースするタスクを作成するには、以下の手順に従います。ファイルを解放すると、ファイルのリストとメタデータは保持されますが、そのファイルのコンテンツのローカルコピーは削除されます。

ファイルをリリースするには (コンソール)

1. https://console.aws.amazon.com/fsx/ で Amazon FSx コンソールを開きます。
2. 左のナビゲーションペインで [ファイルシステム] を選択し、Lustre ファイルシステムを選択します。
3. [Data repository] (データリポジトリ) タブを選択します。
4. [データリポジトリの関連付け] ペインで、リリースタスクを作成する対象のデータリポジトリの関連付けを選択します。
5. [アクション] で [リリースタスクの作成] を選択します。ファイルシステムが S3 のデータリポジトリにリンクされている場合のみ、この選択が使用できます。[データリポジトリのリリースタスクの作成] ダイアログが表示されます。
6. [リリースするためのファイルシステムパス] で、Amazon FSx ファイルシステムからリリースするディレクトリまたはファイルへのパスを指定して、最大 32 個のディレクトリまたはファイルを指定します。指定するパスは、ファイルシステムのマウントポイントに対する相対パスである必要があります。たとえば、マウントポイントが `/mnt/fsx` で、`/mnt/fsx/path1` がリリースするファイルシステムのファイルである場合、指定するパスは `path1` です。ファイルシステム内のすべてのファイルをリリースするには、パスとしてフォワードスラッシュ (/) を指定します。

   > **Note**
   >
   > 指定したパスが有効でない場合、タスクは失敗します。

7. [最終アクセスからの最小の期間] には、その期間内にアクセスされなかったファイルがリリースされるように、期間を日単位で指定します。最終アクセス時刻は、`atime`、`mtime`、`ctime` の最大値を使用して計算されます。最終アクセス期間が、前回のアクセスからの最小期間 (タスク作成時刻を基準とする) よりも長いファイルはリリースされます。この日数より短い期間にアクセスされたファイルは、[リリースするファイルシステムパス] フィールドに含まれている場合でも、リリースされません。最終アクセスからの期間とは無関係に、ファイルをリリースするまでの日数を `0` 日間と指定します。

8. (オプション) [完了レポート] で [有効化] を選択すると、[レポートスコープ] で提供されたスコープを満たすファイルの詳細を提供するタスク完了レポートが生成されます。Amazon FSx がレポートを配信する場所を指定するには、ファイルシステムのリンクされた S3 データリポジトリの [レポートパス] の相対パスを入力します。
9. [データリポジトリタスクを作成] を選択します。

   [File systems] (ファイルシステム) ページの上部にある通知には、先ほど作成したタスクが表示されます。

タスクのステータスと詳細を表示するには、[データリポジトリ] タブで [データリポジトリタスク] まで下にスクロールします。デフォルトのソート順では、最新のタスクがリストの最上部に表示されます。

このページからタスクサマリーを表示するには、先ほど作成したタスクの [Task ID] (タスク ID) を選択します。

ファイルをリリースするには (CLI)

- `create-data-repository-task` CLI コマンドを使用して、FSx for Lustre ファイルシステムのファイルをリリースするタスクを作成します。対応する API オペレーションは `CreateDataRepositoryTask` です。

  以下のパラメータを設定します:

  - ファイルをリリースするファイルシステムの ID に `--file-system-id` をセットします。
  - データをリリースするファイルシステムのパスに `--paths` をセットします。ディレクトリを指定すると、そのディレクトリ内のファイルがリリースされます。ファイルパスが指定されている場合、そのファイルのみがリリースされます。リンクされている S3 バケットにエクスポートされているファイルシステム内のすべてのファイルをリリースするには、パスにフォワードスラッシュ (/) を指定します。
  - `--type` を `RELEASE_DATA_FROM_FILESYSTEM` に設定します。
  - 以下のように `--release-configuration DurationSinceLastAccess` オプションを設定します。
    - `Unit` – `DAYS` に設定します。
    - `Value` — その期間にアクセスされなかったファイルがリリースされるまでの期間を表す整数を日単位で指定します。この日数より短い期間にアクセスされたファイルは、たとえ `--`

paths パラメータに含まれている場合でもリリースされません。最終アクセスからの期間とは無関係に、ファイルをリリースするまでの日数を 0 日間と指定します。

このサンプルコマンドは、リンクされている S3 バケットにエクスポートされ、--release-configuration 基準を満たしているファイルが、指定されたパスのディレクトリからリリースされるように指定します。

```
$ aws fsx create-data-repository-task \
    --file-system-id fs-0123456789abcdef0 \
    --type RELEASE_DATA_FROM_FILESYSTEM \
    --paths path1,path2/file1 \
    --release-configuration '{"DurationSinceLastAccess":
{"Unit":"DAYS","Value":10}}' \
    --report Enabled=false
```

データリポジトリタスクが正常に作成されると、Amazon FSx はタスクの説明を JSON として返します。

ファイルをリリースするタスクを作成したら、タスクの状態を確認できます。データリポジトリタスクを表示する方法の詳細については、「データリポジトリタスクへのアクセス」を参照してください。

オンプレミスのデータに対する Amazon FSx の使用

Amazon FSx を使用して、オンプレミスのデータをクラウド内のコンピューティングインスタンスで処理できます。FSx for Lustre は、 Direct Connect および VPN へのアクセスをサポートしているため、オンプレミスクライアントからファイルシステムをマウントできます。

オンプレミスのデータで Amazon FSx を使用するには

1. ファイルシステムを作成します。詳細については、「使用開始」の演習の「ステップ 1: FSx for Lustre ファイルシステムの作成」を参照してください。
2. オンプレミスのクライアントからファイルシステムをマウントします。詳細については、「オンプレミスまたはピアリングされた Amazon VPC から Amazon FSx ファイルシステムをマウントする」を参照してください。
3. 処理するデータを FSx for Lustre ファイルシステムにコピーします。

4. ファイルシステムをマウントしているクラウド内 Amazon EC2 インスタンスで、コンピューティング集約型のワークロードを実行します。
5. 完了したら、ファイルシステムからオンプレミスのデータロケーションに最終結果をコピーし、FSx for Lustre ファイルシステムを削除します。

データリポジトリのイベントログ

CloudWatch Logs へのログ記録を有効にすると、自動のインポートとエクスポート、およびデータリポジトリタスクおよび復元イベントを使用して行う、ファイルのインポートまたはエクスポート中に発生した障害に関する情報をログ記録できます。詳細については、「Amazon CloudWatch Logs でのロギング」を参照してください。

> **Note**
>
> データリポジトリタスクが失敗した場合には、Amazon FSx が、その失敗に関する情報をタスク完了レポートに書き込みます。完了レポート内の障害情報の詳細ついては、「データリポジトリタスク失敗のトラブルシューティング」を参照してください。

トピック

インポートイベント

エラータイプ	ログレベル	ログメッセージ	根本原因	完了レポート内のエラーコード
オブジェクト一覧表示の失敗	ERROR	S3 バケット *bucket_name* 内で、プレフィクス *prefix* を持つ	Amazon FSx は、S3 バケット内の S3 オブジェクトをリストできま	該当なし

エラータイプ	ログレベル	ログメッセージ	根本原因	完了レポート内のエラーコード
		S3 オブジェクトのリスト作成に失敗しました。	せんでした。S3 バケットポリシーが Amazon FSx に十分なアクセス許可を付与していない場合に、このエラーが発生することがあります。	
サポートされていない S3 ストレージクラス	WARN	S3 バケット *bucket_name* 内のキー *key_value* を持つ S3 オブジェクトのインポートに失敗しました。このオブジェクトはサポートされていない階層 *S3_tier_name* にあります。	Amazon FSx は、S3 オブジェクトをインポートできませんでした。このオブジェクトは、S3 Glacier Flexible Retrieval や S3 Glacier Deep Archive ストレージクラスなど、サポートされていない Amazon S3 ストレージクラス内にあります。	`S3ObjectInUnsupportedTier`

エラータイプ	ログレベル	ログメッセージ	根本原因	完了レポート内のエラーコード
サポートされていないシンボリックリンクのストレージクラス	ERROR	S3 バケット *bucket_name* 内の、キー *key_value* を持つ S3 シンボリックリンクオブジェクトのインポートに失敗しました。このオブジェクトは、サポートされていない階層 *S3_tier_name* にあります。	Amazon FSx は、シンボリックリンクオブジェクトをインポートできませんでした。このオブジェクトは、S3 Glacier Flexible Retrieval や S3 Glacier Deep Archive ストレージクラスなど、サポートされていない Amazon S3 ストレージクラスにあります。	`S3SymlinkInUnsupportedTier`

エラータイプ	ログレベル	ログメッセージ	根本原因	完了レポート内のエラーコード
S3 アクセスが拒否されました	ERROR	S3 バケット *bucket_name* 内のキー *key_value* を持つ S3 オブジェクトのインポートに失敗しました。この S3 オブジェクトへのアクセスが拒否されました。	データリポジトリのエクスポートまたはインポートタスクにおいて、Amazon S3 へのアクセスが拒否されました。 インポートタスクの場合、Amazon FSx ファイルシステムに S3 上のリンクされたデータリポジトリからインポートするため、`s3:HeadObject` および `s3:GetObject` オペレーションを実行する許可が必要です。 インポートタスクの場合、S3 バケットが AWS Key Management	`S3AccessDenied`

エラータイプ	ログレベル	ログメッセージ	根本原因	完了レポート内のエラーコード
			Service (SSE-KMS) に保存されているカスタマー管理キーを使ってサーバー側の暗号化を使用する場合は、サーバー側で暗号化された Simple Storage Service (Amazon S3) バケットの使用 のポリシー設定に従う必要があります。	

エラータイプ	ログレベル	ログメッセージ	根本原因	完了レポート内のエラーコード
アクセス拒否を削除する	ERROR	S3 バケット *bucket_name* 内のキー *key_value* を持つ S3 オブジェクトの、ローカルファイルの削除に失敗しました。この S3 オブジェクトへのアクセスが拒否されました。	自動インポートによる S3 オブジェクトへのアクセスが拒否されました。	該当なし

エラータイプ	ログレベル	ログメッセージ	根本原因	完了レポート内のエラーコード
POSIX 非準拠オブジェクト	ERROR	S3 バケット *bucket_name* 内のキー *key_value* を持つ S3 オブジェクトのインポートに失敗しました。この S3 オブジェクトは POSIX 準拠ではありません。	Simple Storage Service (Amazon S3) オブジェクトは存在しますが、POSIX に準拠しているオブジェクトではないため、インポートできません。サポートされている POSIX メタデータについては、「データリポジトリの POSIX メタデータのサポート」を参照してください。	`S3ObjectPathNotPosixCompliant`

エラータイプ	ログレベル	ログメッセージ	根本原因	完了レポート内のエラーコード
オブジェクトタイプの不一致	ERROR	S3 バケット *bucket_name* 内のキー *key_value* を持つ S3 オブジェクトのインポートに失敗しました。同じ名前の S3 オブジェクトがファイルシステムに既にインポートされています。	インポートしようとした S3 オブジェクトのタイプ (ファイルまたはディレクトリ) が、ファイルシステム内に既存の、同じ名前を持つオブジェクトと異っています。	`S3ObjectTypeMismatch`
ディレクトリメタデータの更新の失敗	ERROR	内部エラーが発生したため、ローカルディレクトリのメタデータの更新に失敗しました。	内部エラーが発生したため、ディレクトリメタデータをインポートできませんでした。	該当なし

エラータイプ	ログレベル	ログメッセージ	根本原因	完了レポート内のエラーコード
S3 オブジェクトが見つかりません	ERROR	キー *key_value* を持つ S3 オブジェクトのインポートに失敗しました。S3 バケット *bucket_name* 内で、このオブジェクトが見つかりません。	Amazon FSx は、ファイルメタデータをインポートできませんでした。対応するオブジェクトがデータリポジトリ内に存在しません。	`S3FileDeleted`
S3 バケットが見つかりません	ERROR	S3 バケット *bucket_name* 内のキー *key_value* を持つ S3 オブジェクトのインポートに失敗しました。このバケットは存在しません。	S3 バケットが見つからないため、Amazon FSx は、ファイルシステムに S3 オブジェクトを自動インポートできません。	該当なし

エラータイプ	ログレベル	ログメッセージ	根本原因	完了レポート内のエラーコード
S3 バケットが見つかりません	ERROR	S3 バケット *bucket_name* 内のキー *key_value* を持つ S3 オブジェクトで、ローカルファイルの削除に失敗しました。このバケットは存在しません。	S3 バケットが見つからないため、Amazon FSx はファイルシステム上の S3 オブジェクトにリンクされたファイルを削除できません。	該当なし
ディレクトリ作成の失敗	ERROR	内部エラーが発生したため、ローカルディレクトリの作成に失敗しました。	内部エラーが発生したため、Amazon FSx による、ファイルシステム上へのディレクトリ作成の自動インポートが失敗しました。	該当なし

エラータイプ	ログレベル	ログメッセージ	根本原因	完了レポート内のエラーコード
ディスク容量がいっぱいです	ERROR	S3 バケット *bucket_name* 内のキー *key_value* を持つ S3 オブジェクトをインポートできませんでした。このファイルシステムに空き容量がありません。	ファイルまたはディレクトリの作成中に、ファイルシステムが使用可能なメタデータサーバーのディスク領域が終了しました。	該当なし

エクスポートイベント

エラータイプ	ログレベル	ログメッセージ	根本原因	完了レポート内のエラーコード
アクセスが拒否される	ERROR	S3 バケット *bucket_name* 内のキー *key_value* を持つ S3 オブジェクトへのアクセスが拒否されたため、このファイルのエクスポートが失敗しました。	データリポジトリのエクスポートタスクにおいて、Amazon S3 へのアクセスが拒否されました。 エクスポートタスクの場合、Amazon FSx ファイルシステムでは、S3 上のリンクさ	`S3AccessDenied`

エラータイプ	ログレベル	ログメッセージ	根本原因	完了レポート内のエラーコード
			れたデータリポジトリにエクスポートする s3:PutObject オペレーションを実行する許可が必要です。この許可は、AWSServiceRoleForFSxS3Access_*fs-0123456789abcdef0* サービスにリンクされたロールで許可されています。詳細については、「Amazon FSx のサービスリンクロールの使用」を参照してください。 エクスポートタスクでは、ファイルシステムの VPC から外部にデータが転送される必要があります。このエラーは、ターゲットリポジトリが使用するバ	

エラータイプ	ログレベル	ログメッセージ	根本原因	完了レポート内のエラーコード
			ケットポリシーに、aws:SourceVpc または aws:SourceVpce IAM グローバル条件キーのいずれかが含まれている場合に発生する可能性があります。 S3 バケットに、ファイルシステムにリンクされた S3 バケットアカウントとは別の AWS アカウント からアップロードされたオブジェクトが含まれている場合、アップロードされたアカウントに関係なく、データリポジトリタスクが S3 メタデータを変更したり、S3 オブジェクトを上書きしたりできます。S3 バケットで、S3 オブジェクト所	

エラータイプ	ログレベル	ログメッセージ	根本原因	完了レポート内のエラーコード
			有権の機能を有効にすることをお勧めします。この機能を使用すると、アップロードに `--acl bucket-owner-full-control` 固定 ACL を提供させることにより、他の AWS アカウント がバケットにアップロードする新しいオブジェクトの所有権を取得できます。S3 バケットで、バケット所有者優先 内のオプションを選択することにより、S3 オブジェクトの所有権を有効にします。詳細については、「Simple Storage Service (Amazon S3) ユーザーガイド」の「S3 オブジェクトの所有権を使用してアップロード	

エラータイプ	ログレベル	ログメッセージ	根本原因	完了レポート内のエラーコード
			されたオブジェクトの所有権をコントロールする」を参照してください。	
エクスポートパスが長すぎます	ERROR	ファイルのエクスポートに失敗しました。ローカルファイルのパスサイズが、S3 でサポートされている最大オブジェクトキー長を超えています。	エクスポートのパスが長すぎます。S3 でサポートされているオブジェクトキーの最大長は 1,024 文字です。	`PathSizeTooLong`
ファイルが大きすぎます	ERROR	ファイルサイズがサポートされている S3 オブジェクトの最大サイズを超えているため、このファイルのエクスポートに失敗しました。	Amazon S3 でサポートされているオブジェクトの最大サイズは 5 TiB です。	`FileSizeTooLarge`

エラータイプ	ログレベル	ログメッセージ	根本原因	完了レポート内のエラーコード
KMS キーが見つかりませんでした	ERROR	バケットの KMS キーが見つからなかったため、S3 バケット *bucket_name* 内のキー *key_value* を持つ S3 オブジェクトのファイルのエクスポートが失敗しました。	AWS KMS key が見つからなかったため、Amazon FSx はファイルをエクスポートできませんでした。必ず、S3 バケットと同じ AWS リージョン にあるキーを使用してください。KMS キーの作成の詳細については、「AWS Key Management Service デベロッパーガイド」の「Creating keys」を参照してください。	N/A

エラータイプ	ログレベル	ログメッセージ	根本原因	完了レポート内のエラーコード
リソースビジー	ERROR	ファイルをエクスポートできませんでした。このファイルは、別のプロセスで使用されています。	Amazon FSx は、ファイルシステム上の別のクライアントによって変更されているため、ファイルをエクスポートできませんでした。ワークフローによるファイルへの書き込みが完了した後、このタスクを再試行できます。	`ResourceBusy`
ファイルがリリースされました	WARN	エクスポートはスキップされました: ローカルファイルがリリース済み状態であり、かつキー *key_value* を持ちリンクされた S3 オブジェクトが、バケット *bucket_name* 内に見つかりませんでした。	このファイルはファイルシステム上でリリース済み状態であるため、Amazon FSx によるエクスポートが行えませんでした。	該当なし

エラータイプ	ログレベル	ログメッセージ	根本原因	完了レポート内のエラーコード
データリポジトリパスの不一致	WARN	エクスポートがスキップされました: このローカルファイルは、データリポジトリにリンクするファイルシステムパスに属していません。	オブジェクトがデータリポジトリにリンクされているファイルシステムパスに属していないため、Amazon FSx は、このオブジェクトをエクスポートできませんでした。	該当なし
内部エラー	ERROR	ファイルシステムオブジェクトのエクスポート中に、自動エクスポート内でエラーが発生しました	内部 (自動エクスポートまたは Lustre レベルの) エラーのため、エクスポートが失敗しました。	該当なし
完了レポートのアップロードの失敗	ERROR	*bucket_name* への、データリポジトリタスク完了レポートのアップロードに失敗しました。	Amazon FSx は完了レポートをアップロードできませんでした。	該当なし

エラータイプ	ログレベル	ログメッセージ	根本原因	完了レポート内のエラーコード
完了レポートの検証の失敗	ERROR	データリポジトリタスク完了レポートをバケット *bucket_name* にアップロードできませんでした。このファイルシステムに関連付けられたデータリポジトリに、完了レポートのパス *report_path* が属していません	お客様が指定した S3 パスが、リンクされたデータリポジトリに属していないため、Amazon FSx は、この完了レポートをアップロードできませんでした。	該当なし

HSM 復元イベント

エラータイプ	ログレベル	ログメッセージ	根本原因
アクセスが拒否される	ERROR	S3 バケット *bucket_name* 内の S3 オブジェクト *object_name* へのアクセスが拒否されたため、このファイルのエクスポートが失敗しました。	HSM コマンドを使用してファイルを復元しようとすると、Amazon S3 へのアクセスが拒否されました。ファイルシステムに S3 上のリンクされたデータリポジトリから復元するため、`s3:HeadObject` および

エラータイプ	ログレベル	ログメッセージ	根本原因
			`s3:GetObject` オペレーションを実行する許可が必要です。
サポートされていない S3 ストレージクラス	WARN	バケット *`bucket_name`* の S3 オブジェクト *`object_name`* がサポートされていない *`S3_storage_class _name`* にあったため、ファイルの復元に失敗しました。	対応する S3 オブジェクトが S3 Glacier Flexible Retrieval や S3 Glacier Deep Archive などの S3 サポートされていないストレージクラスにあるため、Amazon FSx はファイルを復元できませんでした。`hsm_restore` を使用する前に、まず Glacier ストレージクラスからオブジェクトを復元する必要があります。
S3 オブジェクトが見つかりません	ERROR	S3 バケット *`bucket_name`* 内で、このオブジェクトが見つからなかったため、キー *`key_value`* を持つ S3 オブジェクトのインポートに失敗しました。	対応する S3 オブジェクトがデータリポジトリ内に存在しなかったため、Amazon FSx は、ファイルを復元できませんでした。

エラータイプ	ログレベル	ログメッセージ	根本原因
S3 バケットが見つかりません	ERROR	S3 バケット *bucket_name* が存在しないため、ファイルの復元に失敗しました。	リンクされた S3 バケットが存在しないため、Amazon FSx はファイルを復元できません。
ディスク容量がいっぱいです	ERROR	ファイルシステムに使用可能なストレージ領域がないため、ファイルの復元に失敗しました。	S3 からファイルデータを復元しようとしましたが、ファイルシステムの使用可能なストレージ容量が不足しています。ファイルシステムのストレージ容量を増やすか、ファイルを解放して領域を解放することを検討してください。

以前のデプロイタイプでの使用

このセクションは、Scratch 1 デプロイタイプのファイルシステムに加えて、データリポジトリの関連付けを使用しない `Scratch 2` または `Persistent 1` デプロイタイプのファイルシステムにも適用されます。自動エクスポートおよび複数のデータリポジトリのサポートは、データリポジトリの関連付けを使用していない FSx for Lustre ファイルシステムでは利用できないことに留意してください。

トピック

- Simple Storage Service (Amazon S3) バケットにファイルシステムにリンクする
- S3 バケットから更新を自動的にインポートする

Simple Storage Service (Amazon S3) バケットにファイルシステムにリンクする

Amazon FSx for Lustre ファイルシステムを作成すると、Simple Storage Service (Amazon S3) の耐久性のあるデータリポジトリにリンクできます。ファイルシステムを作成する前に、リンク先の Simple Storage Service (Amazon S3) バケットがすでに作成されていることを確認してください。[Create file system] (ファイルシステムの作成) ウィザードでは、オプションの データリポジトリ Import/Export ペインに、次のデータリポジトリ設定プロパティを設定します。

- ファイルシステムの作成後に S3 バケットのオブジェクトを追加または変更するときに、Amazon FSx がファイルとディレクトリのリストを最新の状態に保つ方法を選択します。詳細については、「S3 バケットから更新を自動的にインポートする」を参照してください。
- バケットをインポートする: リンクされたリポジトリに使用している S3 バケットの名前を入力します。
- インポートプレフィックス: S3 バケット内のデータの一部ファイルとディレクトリリストのみをファイルシステムにインポートする場合は、オプションのインポートプレフィックスを入力します。インポートプレフィックスは、S3 バケット内のデータのインポート元を定義します。
- エクスポートプレフィックス: Amazon FSx がファイルシステムの内容をリンクされた S3 バケットにエクスポートする場所を定義します。

Amazon FSx が、FSx for Lustre ファイルシステムからインポート元の S3 バケットの同じディレクトリにデータをエクスポートする 1:1 マッピングを作成できます。1:1 のマッピングを作成するには、ファイルシステムを作成するときに、プレフィックスなしで S3 バケットへのエクスポートパスを指定します。

- コンソールを使用してファイルシステムを作成する場合は、プレフィックスのエクスポート > 指定したプレフィックス オプションを選択し、プレフィックスフィールドを空白のままにします。
- AWS CLI または API を使用してファイルシステムを作成する場合は、エクスポートパスを S3 バケットの名前として、`ExportPath=s3://amzn-s3-demo-bucket/` などの追加のプレフィックスなしで指定します。

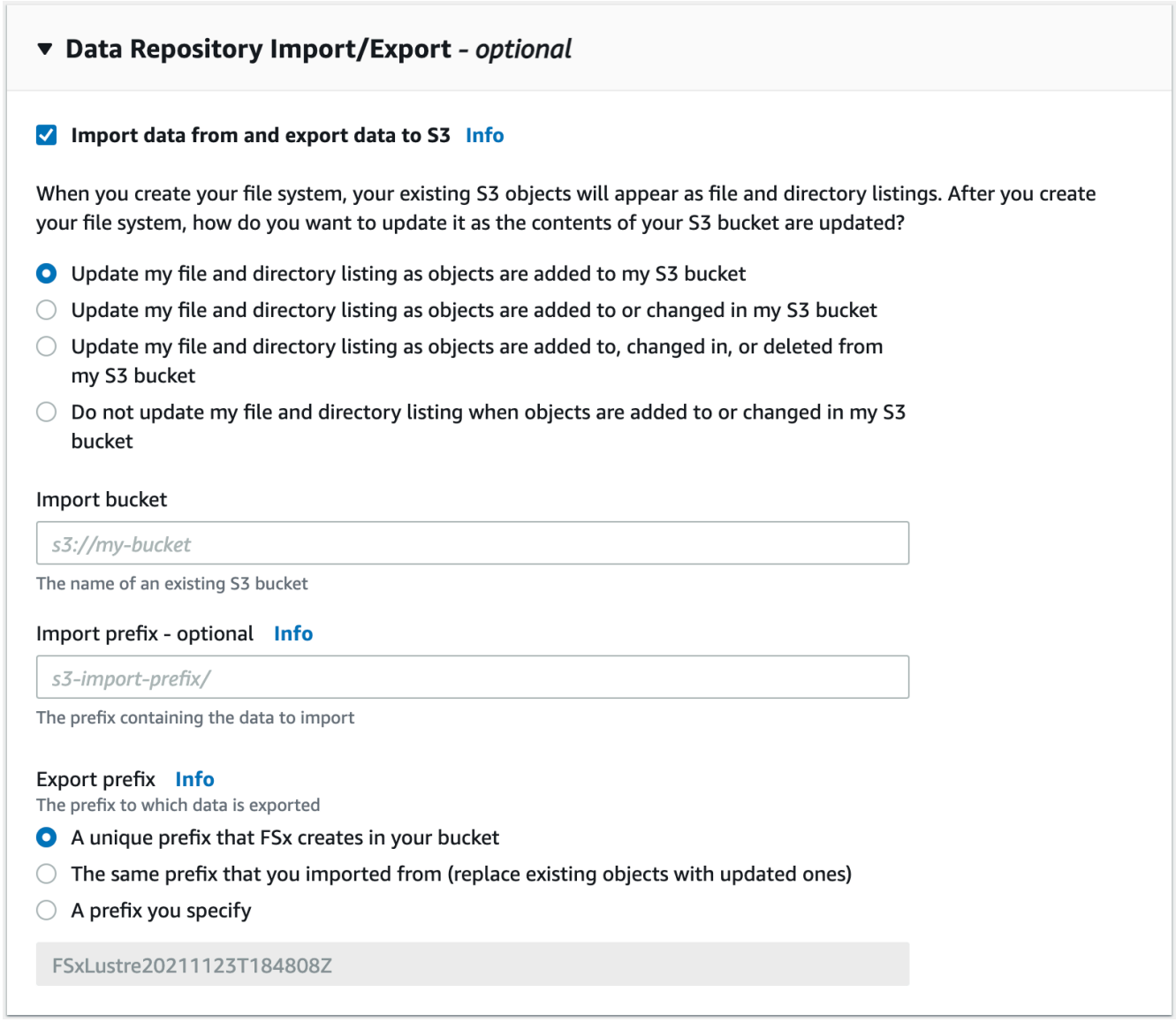
この方法を使用すると、インポートパスを指定するときにインポートプレフィックスを含めることができ、エクスポートの 1:1 マッピングには影響しません。

S3 バケットにリンクされているファイルシステムの作成

以下の手順では、AWS 管理コンソールと AWS コマンドラインインターフェイス (AWS CLI) を使用して、S3 バケットにリンクされた Amazon FSx ファイルシステムを作成するプロセスについて説明します。

Console

1. https://console.aws.amazon.com/fsx/ で Amazon FSx コンソールを開きます。
2. ダッシュボードから、[Create file system] (ファイルシステムの作成) を選択します。
3. ファイルシステムタイプで、FSx for Lustre を選択してから、[Next] (次へ) を選択します。
4. [File system details] (ファイルシステムの詳細) と [Network and Security] (ネットワークおよびセキュリティ) セクションに必要な情報を入力します。詳細については、「ステップ 1: FSx for Lustre ファイルシステムの作成」を参照してください。
5. Simple Storage Service (Amazon S3) にリンクされたデータリポジトリを設定するための [Data repository import/export] (データリポジトリ Import/Export) パネルを使用します。[Import data from and export data to S3] (S3 からのデータインポートと S3 へのデータエクスポート) を展開して [Data Repository Import/Export] (データリポジトリの Import/Export) セクションを開き、データリポジトリを設定します。



6. S3 バケットのオブジェクトを追加または変更したときに、Amazon FSx がファイルとディレクトリのリストを最新の状態に保つ方法を選択します。ファイルシステムを作成すると、既存の S3 オブジェクトがファイルとディレクトリのリストとして表示されます。

   - S3 バケットにオブジェクトが追加されると、ファイルとディレクトリのリストを更新する: (デフォルト) Amazon FSx は、リンクされた S3 バケットに追加された新しいオブジェクトのうち、現在 FSx ファイルシステム内に存在しないオブジェクトのファイルとディレクトリのリストを自動的に更新します。Amazon FSx は、S3 バケット内で変更されたオブジェクトのリストを更新しません。Amazon FSx では、S3 バケットで削除されたオブジェクトのリストは削除されません。

> **Note**
>
> CLI と API を使用してリンクされた S3 バケットからデータをインポートするためのデフォルトのインポート設定は NONE です。コンソールを使用する場合のデフォルトのインポート設定は、新しいオブジェクトが S3 バケットに追加されたときに Lustre を更新することです。

- S3 バケットにオブジェクトが追加または変更されると、ファイルとディレクトリのリストを更新する: このオプションの選択後は、S3 バケットに追加された新しいオブジェクトや S3 バケットで変更された既存のオブジェクトのファイルとディレクトリのリストが、Amazon FSx によって自動的に更新されます Amazon FSx では、S3 バケットで削除されたオブジェクトのリストは削除されません。
- S3 バケットにオブジェクトが追加、変更、または削除されると、ファイルとディレクトリのリストを更新する: このオプションの選択後は、S3 バケットに追加された新しいオブジェクトや S3 バケットで削除された既存のオブジェクトのファイルとディレクトリのリストが、Amazon FSx によって自動的に更新されます
- S3 バケットにオブジェクトが追加、変更、削除されたときに、ファイルを更新したり、直接リスト表示したりしない - Amazon FSx は、ファイルシステムの作成時に、リンクされた S3 バケットのファイルとディレクトリのリストのみを更新します。このオプションの選択後は、新しいオブジェクトや変更、削除されたオブジェクトのファイルとディレクトリのリストは FSx で更新されません。

7. S3 バケット内のデータの一部ファイルとディレクトリリストのみをファイルシステムにインポートする場合は、オプションの インポートプレフィックス を入力します。インポートプレフィックスは、S3 バケット内のデータのインポート元を定義します。詳細については、「S3 バケットから更新を自動的にインポートする」を参照してください。
8. 使用可能な エクスポートプレフィックス オプションの 1 つを選択します。

   - Amazon FSx がバケット内に作成する一意のプレフィックス: このオプションを選択すると、FSx for Lustre によって生成されたプレフィックスを使用して、新規および変更されたオブジェクトをエクスポートできます。プレフィックスは、次の「/FSxLustre*file-system-creation- timestamp*」のようになります。タイムスタンプは UTC 形式です (例えば、FSxLustre20181105T222312Z)。
   - インポート元と同じプレフィックス (既存のオブジェクトを更新されたオブジェクトに置き換える): 既存のオブジェクトを更新されたオブジェクトに置き換えるには、このオプションを選択します。

- 指定するプレフィックス: インポートしたデータを保持し、指定したプレフィックスを使用して新規および変更されたオブジェクトをエクスポートするには、このオプションを選択します。S3 バケットにデータをエクスポートするときに 1:1 のマッピングを実現するには、このオプションを選択し、プレフィックスフィールドを空白のままにします。FSx は、インポート元のディレクトリと同じディレクトリにデータをエクスポートします。

9. (オプション) メンテナンスプリファレンス を設定するか、またはシステムのデフォルトを使用します。
10. [Next] (次へ) を選択してから、ファイルシステム設定を確認します。必要に応じて変更を加えます。
11. [Create file system] (ファイルシステムを作成する) を選択します。

AWS CLI

次の例では、amzn-s3-demo-bucket にリンクされた Amazon FSx ファイルシステムを作成します。ファイルシステムの作成後に、リンクされたデータリポジトリ内の新規、変更、および削除されたファイルをインポートするインポートプリファレンスを使用します。

> **Note**
>
> CLI と API を使用して、リンクされた S3 バケットからデータをインポートするためのデフォルトのインポートプリファレンスの設定は NONE です。これは、コンソール使用時のデフォルトの動作とは異なります。

FSx for Lustre ファイルシステムを作成するには、以下に示すような Amazon FSx CLI コマンド create-file-system を使用します。対応する API オペレーションは CreateFileSystem です。

```
$ aws fsx create-file-system \
--client-request-token CRT1234 \
--file-system-type LUSTRE \
--file-system-type-version 2.10 \
--lustre-configuration
AutoImportPolicy=NEW_CHANGED_DELETED,DeploymentType=SCRATCH_1,ImportPath=s
3://amzn-s3-demo-bucket/,ExportPath=s3://amzn-s3-demo-bucket/export,
PerUnitStorageThroughput=50 \
--storage-capacity 2400 \
--subnet-ids subnet-123456 \
```

```
--tags Key=Name,Value=Lustre-TEST-1 \
--region us-east-2
```

次の例に示すように、ファイルシステムを正常に作成すると、Amazon FSx はファイルシステムの説明を JSON として返します。

```
{
    "FileSystems": [
        {
            "OwnerId": "owner-id-string",
            "CreationTime": 1549310341.483,
            "FileSystemId": "fs-0123456789abcdef0",
            "FileSystemType": "LUSTRE",
            "FileSystemTypeVersion": "2.10",
            "Lifecycle": "CREATING",
            "StorageCapacity": 2400,
            "VpcId": "vpc-123456",
            "SubnetIds": [
                "subnet-123456"
            ],
            "NetworkInterfaceIds": [
                "eni-039fcf55123456789"
            ],
            "DNSName": "fs-0123456789abcdef0.fsx.us-east-2.amazonaws.com",
            "ResourceARN": "arn:aws:fsx:us-east-2:123456:file-system/
fs-0123456789abcdef0",
            "Tags": [
                {
                    "Key": "Name",
                    "Value": "Lustre-TEST-1"
                }
            ],
            "LustreConfiguration": {
                "DeploymentType": "PERSISTENT_1",
                "DataRepositoryConfiguration": {
                    "AutoImportPolicy": "NEW_CHANGED_DELETED",
                    "Lifecycle": "UPDATING",
                    "ImportPath": "s3://amzn-s3-demo-bucket/",
                    "ExportPath": "s3://amzn-s3-demo-bucket/export",
                    "ImportedFileChunkSize": 1024
                },
                "PerUnitStorageThroughput": 50
            }
```

```
        }
    ]
}
```

ファイルシステムのエクスポートパスの表示

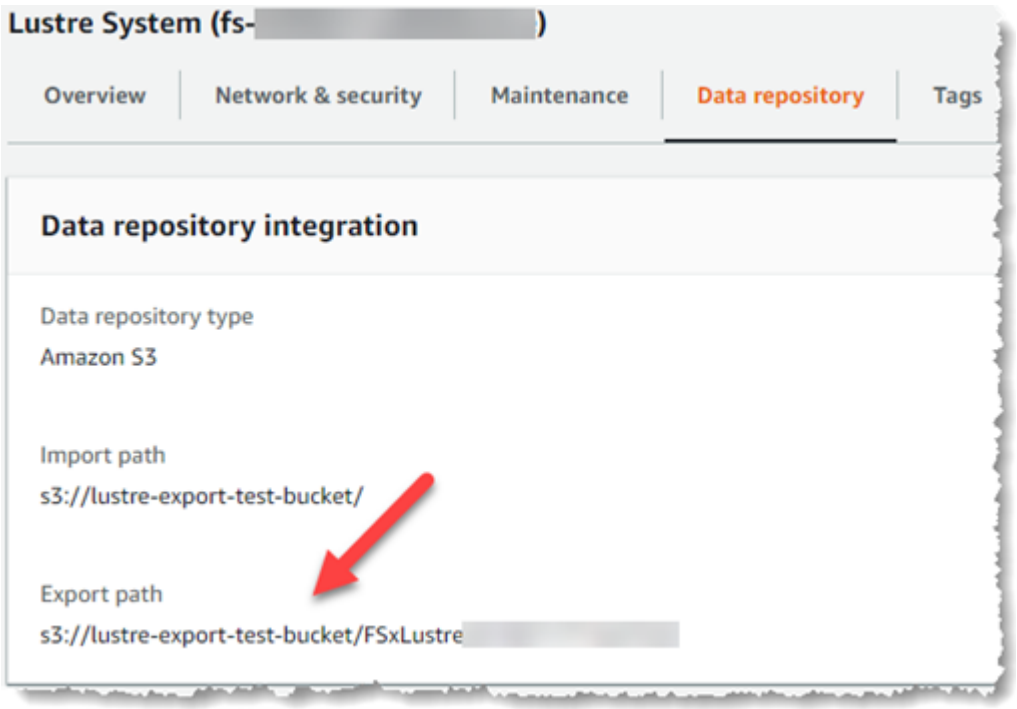
FSx for Lustre コンソール、AWS CLI、および API を使用して、ファイルシステムのエクスポートパスを表示できます。

Console

1. https://console.aws.amazon.com/fsx/ で Amazon FSx コンソールを開きます
2. FSx for Lustreファイルシステムの場合は、[File system name] (ファイルシステム名) または [File system ID] (ファイルシステム ID) を選択して、エクスポートパスを表示します。

   そのファイルシステムの詳細ページが表示されます。
3. [Data repository] (データリポジトリ) タブを選択します。

   [Data repository integration] (データリポジトリ統合) パネルが表示され、インポートパスとエクスポートパスが表示されます。



CLI

ファイルシステムのエクスポートパスを特定するには、describe-file-systems AWS CLI コマンドを使用します。

```
aws fsx describe-file-systems
```

レスポンスで LustreConfiguration の下の ExportPath プロパティを探します。

```
{
    "OwnerId": "111122223333",
    "CreationTime": 1563382847.014,
    "FileSystemId": "",
    "FileSystemType": "LUSTRE",
    "Lifecycle": "AVAILABLE",
    "StorageCapacity": 2400,
    "VpcId": "vpc-6296a00a",
    "SubnetIds": [
        "subnet-1111111"
    ],
    "NetworkInterfaceIds": [
        "eni-0c288d5b8cc06c82d",
        "eni-0f38b702442c6918c"
    ],
    "DNSName": "fs-0123456789abcdef0.fsx.us-east-2.amazonaws.com",
    "ResourceARN": "arn:aws:fsx:us-east-2:267731178466:file-system/
fs-0123456789abcdef0",
    "Tags": [
        {
          "Key": "Name",
          "Value": "Lustre System"
        }
    ],
 "LustreConfiguration": {
    "DeploymentType": "SCRATCH_1",
    "DataRepositoryConfiguration": {
    "AutoImportPolicy": " NEW_CHANGED_DELETED",
    "Lifecycle": "AVAILABLE",
    "ImportPath": "s3://amzn-s3-demo-bucket/",
    "ExportPath": "s3://amzn-s3-demo-bucket/FSxLustre20190717T164753Z",
    "ImportedFileChunkSize": 1024
    }
  },
```

```
  "PerUnitStorageThroughput": 50,
  "WeeklyMaintenanceStartTime": "6:09:30"
}
```

データリポジトリのライフサイクル状態

データリポジトリのライフサイクル状態は、ファイルシステムのリンクされたデータリポジトリに関するステータス情報を提供します。データリポジトリには、以下のライフサイクル状態があります。

- 作成中: Amazon FSx は、ファイルシステムとリンクされたデータリポジトリの間にデータリポジトリの設定を作成しています。データリポジトリは使用できません。
- 使用可能: データリポジトリを使用できます。
- 更新中: データリポジトリの設定では、可用性に影響する可能性があるお客様が開始した更新を実行しています。
- 設定ミス: Amazon FSx は、データリポジトリの設定が修正されるまで S3 バケットから更新を自動的にインポートできません。詳細については、「正しく設定されていないリンクされた S3 バケットのトラブルシューティング」を参照してください。

Amazon FSx コンソール、AWS コマンドラインインターフェイス、および Amazon FSx API を使用して、ファイルシステムのリンクされたデータリポジトリのライフサイクル状態を表示できます。Amazon FSx コンソールでは、ファイルシステムの [Data Repository] (データリポジトリ) タブの [Data Repository Integration] (データリポジトリ統合) ペインで、データリポジトリの ライフサイクル状態 にアクセスできます。`Lifecycle` のプロパティは、`describe-file-systems` CLI コマンド (同等の API アクションは `DescribeFileSystems`) のレスポンスの `DataRepositoryConfiguration` オブジェクトに配置されます。

S3 バケットから更新を自動的にインポートする

デフォルトでは、新しいファイルシステムを作成すると、Amazon FSx はファイルシステムの作成時に、リンクされた S3 バケット内のオブジェクトのファイルメタデータ (名前、所有権、タイムスタンプ、アクセス許可) をインポートします。FSx for Lustre ファイルシステムは、ファイルシステムの作成後に、S3 バケットに追加、変更、または S3 バケットから削除されたオブジェクトのメタデータを自動的にインポートするように設定できます。FSx for Lustre は、ファイルシステムの作成時にファイルメタデータをインポートするのと同じ方法で、作成後に変更されたオブジェクトのファイルとディレクトリリストを更新します。Amazon FSx は、変更されたオブジェクトのファイルとディレクトリリストを更新します。S3 バケット内の変更されたオブジェクトにメタデータが含まれ

なくなった場合、Amazon FSx はデフォルトのアクセス許可を使用するのではなく、ファイルの現在のメタデータ値を保持します。

> **Note**
>
> インポート設定は、2020 年 7 月 23 日の東部標準時午後 3 時以降に作成されたLustreファイルシステムのFSxで利用できます。

新しいファイルシステムの作成時にインポートプリファレンスを設定でき、FSx 管理コンソール、AWS CLI、および AWS API、を使用して、既存のファイルシステムの設定を更新できます。ファイルシステムを作成すると、既存の S3 オブジェクトがファイルとディレクトリのリストとして表示されます。ファイルシステム作成後、S3 バケットのコンテンツが更新されるときに、どのように更新しますか? ファイルシステムには、以下のいずれかのインポート設定があります。

> **Note**
>
> 更新を自動でインポートするためには、FSx for Lustreファイルシステムとリンクされた S3 バケットが同じ AWS リージョンに配置されている必要があります。

- S3 バケットにオブジェクトが追加されると、ファイルとディレクトリのリストを更新する: (デフォルト) Amazon FSx は、リンクされた S3 バケットに追加された新しいオブジェクトのうち、現在 FSx ファイルシステム内に存在しないオブジェクトのファイルとディレクトリのリストを自動的に更新します。Amazon FSx は、S3 バケット内で変更されたオブジェクトのリストを更新しません。Amazon FSx では、S3 バケットで削除されたオブジェクトのリストは削除されません。

  > **Note**
  >
  > CLI と API を使用してリンクされた S3 バケットからデータをインポートするためのデフォルトのインポート設定は `NONE` です。コンソールを使用する場合のデフォルトのインポート設定は、新しいオブジェクトが S3 バケットに追加されたときに Lustre を更新することです。

- S3 バケットにオブジェクトが追加または変更されると、ファイルとディレクトリのリストを更新する: このオプションの選択後は、S3 バケットに追加された新しいオブジェクトや S3 バケットで変更された既存のオブジェクトのファイルとディレクトリのリストが、Amazon FSx によって自動

的に更新されます Amazon FSx では、S3 バケットで削除されたオブジェクトのリストは削除されません。

- S3 バケットにオブジェクトが追加、変更、または削除されると、ファイルとディレクトリのリストを更新する: このオプションの選択後は、S3 バケットに追加された新しいオブジェクトや S3 バケットで削除された既存のオブジェクトのファイルとディレクトリのリストが、Amazon FSx によって自動的に更新されます
- S3 バケットにオブジェクトが追加、変更、削除されたときに、ファイルを更新したり、直接リスト表示したりしない - Amazon FSx は、ファイルシステムの作成時に、リンクされた S3 バケットのファイルとディレクトリのリストのみを更新します。このオプションの選択後は、新しいオブジェクトや変更、削除されたオブジェクトのファイルとディレクトリのリストは FSx で更新されません。

リンクされた S3 バケットの変更に基づいてファイルシステムファイルとディレクトリのリストを更新するようにインポートプリファレンスを設定すると、Amazon FSx は FSx という名前のリンクされた S3 バケットにイベント通知設定を作成します。S3 バケットのイベント通知設定 FSx を変更または削除しないでください - それにより、新しい、または変更されたファイルとディレクトリの一覧がファイルシステムに自動的にインポートされなくなります。

Amazon FSx がリンクされた S3 バケットで変更されたファイルリストを更新すると、ファイルが書き込みロックされていても、更新されたバージョンでローカルファイルが上書きされます。同様に、リンクされた S3 バケットで対応するオブジェクトが削除されたときに、Amazon FSx がファイルリストを更新すると、ファイルが書き込みロックされていても、ローカルファイルが削除されます。

Amazon FSx は、ファイルシステムを更新するために最善の努力を払います。Amazon FSx は、次の状況での変更でファイルシステムを更新できません。

- Amazon FSx に変更または新規の S3 オブジェクトを開く許可がない場合。
- リンクされた S3 バケットの FSx イベント通知設定が削除または変更された場合。

これらのいずれの場合も、データリポジトリのライフサイクルの状態は 設定ミス になります。詳細については、「データリポジトリのライフサイクル状態」を参照してください。

前提条件

Amazon FSx がリンクされた S3 バケットから新規、変更、または削除されたファイルを自動的にインポートするには、次の条件が必要です。

- ファイルシステムとそれにリンクされた S3 バケットは、同じ AWS リージョンに配置する必要があります。
- S3バケットには、設定ミスされたライフサイクル状態はありません。詳細については、「データリポジトリのライフサイクル状態」を参照してください。
- アカウントには、リンクされた S3 バケットでイベント通知を設定および受信するために必要なアクセス許可が必要です。

サポートされているファイル変更のタイプ

Amazon FSx では、リンクされた S3 バケットで発生するファイルおよびフォルダーへの以下の変更のインポートがサポートされています。

- ファイル内容の変更
- ファイルまたはフォルダのメタデータの変更
- シンボリックリンクターゲットまたはメタデータの変更

インポートプリファレンスの更新

新しいファイルシステムを作成するときに、ファイルシステムのインポートプリファレンスを設定できます。詳細については、「Amazon S3 バケットにファイルシステムにリンクする」を参照してください。

ファイルシステムのインポートプリファレンスは、以下の手順に示すように、AWS マネジメントコンソール、AWS CLI、Amazon FSx API を使用して作成後に更新することもできます。

Console

1. https://console.aws.amazon.com/fsx/ で Amazon FSx コンソールを開きます。
2. ダッシュボードから、[File systems] (ファイルシステム) を選択します。
3. 管理するファイルシステムを選択し、ファイルシステムの詳細を表示します。
4. [Data repository] (データリポジトリ) を選択して、データリポジトリの設定を表示します。ライフサイクル状態が [AVAILABLE] (利用可能) または [MISCONFIGURED] (設定ミス) の場合は、インポートプリファレンスを変更できます。詳細については、「データリポジトリのライフサイクル状態」を参照してください。

5. [Actions] (アクション) を選択してから、[Update import preferences] (インポートプリファレンスを更新) を選択して [Update import preferences] (インポートプリファレンスを更新) ダイアログボックスを表示します。
6. 新しい設定を選択し、[Update] (更新) を選択して変更を加えます。

CLI

インポートプリファレンスを更新するには、`update-file-system` CLI コマンドを使用します。対応する API オペレーションは `UpdateFileSystem` です。

ファイルシステムの `AutoImportPolicy` を正常に更新すると、Amazon FSx は更新されたファイルシステムの説明を JSON として、以下のように返します。

```
{
    "FileSystems": [
        {
            "OwnerId": "111122223333",
            "CreationTime": 1549310341.483,
            "FileSystemId": "fs-0123456789abcdef0",
            "FileSystemType": "LUSTRE",
            "Lifecycle": "UPDATING",
            "StorageCapacity": 2400,
            "VpcId": "vpc-123456",
            "SubnetIds": [
                "subnet-123456"
            ],
            "NetworkInterfaceIds": [
                "eni-039fcf55123456789"
            ],
            "DNSName": "fs-0123456789abcdef0.fsx.us-east-2.amazonaws.com",
            "ResourceARN": "arn:aws:fsx:us-east-2:123456:file-system/
fs-0123456789abcdef0",
            "Tags": [
                {
                    "Key": "Name",
                    "Value": "Lustre-TEST-1"
                }
            ],
            "LustreConfiguration": {
                "DeploymentType": "SCRATCH_1",
                "DataRepositoryConfiguration": {
                    "AutoImportPolicy": "NEW_CHANGED_DELETED",
```

```
                    "Lifecycle": "UPDATING",
                    "ImportPath": "s3://amzn-s3-demo-bucket/",
                    "ExportPath": "s3://amzn-s3-demo-bucket/export",
                    "ImportedFileChunkSize": 1024
                }
                "PerUnitStorageThroughput": 50,
                "WeeklyMaintenanceStartTime": "2:04:30"
            }
        }
    ]
}
```

Amazon FSx for Lustre のパフォーマンス

本章では、Amazon FSx for Lustre のパフォーマンスに関するトピックを紹介し、ファイルシステムのパフォーマンスを最大化するための重要なヒントおよび推奨事項を提示します。

トピック

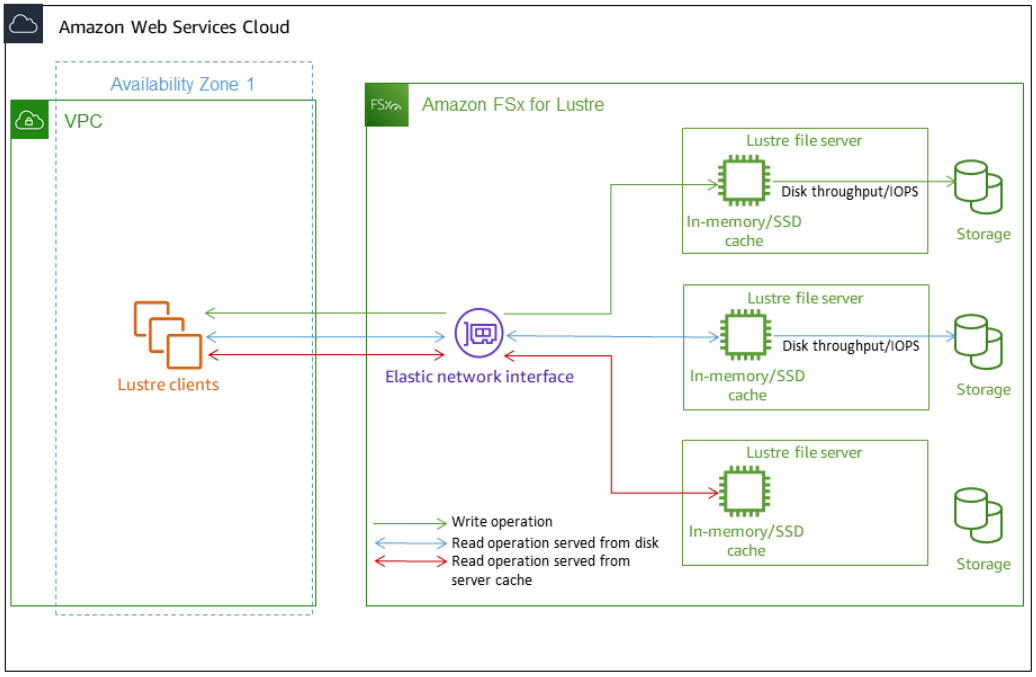
概要:

Amazon FSx for Lustre は、一般的な高性能ファイルシステムである Lustre をベースに構築されており、ファイルシステムのサイズに応じて直線的に増加するスケールアウトパフォーマンスを提供します。Lustre ファイルシステムは複数のファイルサーバーとディスクをまたいで水平にスケールします。このスケーリングにより、各クライアントは各ディスクに保存されているデータに直接アクセスして、従来のファイルシステムに存在するボトルネックの多くを取り除くことができます。Amazon FSx for Lustre は、Lustre のスケーラブルなアーキテクチャに基づいて構築され、多数のクライアントで高いレベルのパフォーマンスをサポートします。

FSx for Lustre のファイルシステム用のしくみ

各 FSx for Lustre ファイルシステムは、クライアントが通信するファイルサーバと、データを格納する各ファイルサーバに接続されたディスクのセットで設定されます。各ファイルサーバは、高速のインメモリキャッシュを使用して、最も頻繁にアクセスされるデータのパフォーマンスを向上させます。ストレージクラスに応じて、ファイルサーバーにはオプションの SSD 読み取りキャッシュをプ

ロビジョニングできます。クライアントがメモリ内キャッシュまたは SSD キャッシュに格納されているデータにアクセスする場合、ファイルサーバーはディスクから読み取る必要がないため、レイテンシーが減少し、ドライブ可能なスループットの合計量が増加します。次の図表は、書き込み操作、ディスクから実行される読み取り操作、およびインメモリまたは SSD キャッシュから実行される読み取り操作のパスを示しています。



ファイルサーバーのインメモリまたは SSD キャッシュに保存されているデータを読み取る場合、ファイルシステムのパフォーマンスはネットワークスループットによって決まります。ファイルシステムにデータを書き込むとき、またはインメモリキャッシュに保存されていないデータを読み取る場合、ファイルシステムのパフォーマンスは、ネットワークスループットとディスクスループットの低い方によって決まります。

SSD および HDD ストレージクラスのネットワークスループット、ディスクスループット、および IOPS の特性の詳細については、「SSD および HDD ストレージクラスのパフォーマンス特性」および「インテリジェント階層化 ストレージクラスのパフォーマンス特性」を参照してください。

ファイルシステムのメタデータパフォーマンス

ファイルシステムメタデータ IO オペレーション (IOPS) は、1 秒あたりに作成、一覧表示、読み取り、削除できるファイルとディレクトリの数を決定します。

永続 2 ファイルシステムでは、ストレージ容量とは独立してメタデータ IOPS をプロビジョニングできるほか、クライアントインスタンスがファイルシステム上で生成しているメタデータ IOPS の数および種類をより詳細に把握することが可能です。SSD ファイルシステムでは、メタデータ IOPS は、プロビジョニングしたストレージ容量に基づいて自動的にプロビジョニングされます。インテリジェント階層化ストレージクラスのファイルシステムはは、自動モードをサポートしていません。

FSx for Lustre の 永続 2 ファイルシステムでは、プロビジョニングしたメタデータ IOPS の数およびメタデータ操作の種類によって、ファイルシステムがサポートできるメタデータ操作の速度が決まります。プロビジョニングするメタデータ IOPS のレベルによって、ファイルシステムのメタデータディスクにプロビジョニングされる IOPS の数が決まります。

操作タイプ	プロビジョニングされたメタデータ IOPS ごとに 1 秒あたりに駆動できるオペレーション
ファイルの作成、開く、閉じる	2
ファイルの削除	1
ディレクトリの作成、名前の変更	0.1
ディレクトリの削除	0.2

SSD ファイルシステムでは、自動モードにより、メタデータ IOPS のプロビジョニングを行うことができます。自動モードでは、Amazon FSx は、次の表に従ってファイルシステムのストレージ容量に基づいてメタデータ IOPS を自動的にプロビジョニングします。

ファイルシステムのストレージ容量	自動モードでのメタデータ IOPS を含む
1200 GiB	1500
2400 GiB	3000
4800 ~ 9600 GiB	6000
12000 ~ 45600 GiB	12000
48,000 GiB 以上	24000 GiB あたり 12000 IOPS

ユーザープロビジョニングモードでは、オプションでプロビジョニングするメタデータ IOPS の数を指定できます。有効な値は次のとおりです。

- SSD ファイルシステムでは、有効な値は、1500、3000、6000、12000、および 192000 までの 12000 の倍数です。
- インテリジェント階層化ファイルシステムの場合、有効な値は 6000 と 12000 です。

メタデータ IOPS を設定する方法については、「メタデータパフォーマンスの管理」を参照してください。ファイルシステムのデフォルトのメタデータ IOPS 数を超えてプロビジョニングしたメタデータ IOPS については、別途料金が発生することにご留意ください。

個々のクライアントインスタンスへのスループット

スループットキャパシティが 10 GBps を超えるファイルシステムを作成する場合は、Elastic Fabric Adapter (EFA) を有効にして、クライアントインスタンスあたりのスループットを最適化することをお勧めします。クライアントインスタンスごとのスループットをさらに最適化するために、EFA 対応のファイルシステムでは、EFA 対応の NVIDIA GPU 搭載クライアントインスタンス向けの GPUDirect Storage および ENA Express 対応クライアントインスタンス向けの ENA Express もサポートされています。

単一のクライアントインスタンスに対して得られるスループットは、選択したファイルシステムの種類およびクライアントインスタンスのネットワークインターフェイスに依存します。

ファイルシステムタイプ	クライアントインスタンスのネットワークインターフェイス	クライアントあたりの最大スループット、Gbps
EFA 非対応	いずれか	100 Gbps*
EFA 対応	ENA	100 Gbps*
EFA 対応	ENA Express	100 Gbps
EFA 対応	EFA	700 Gbps
EFA 対応	GDS を使用した EFA	1200 Gbps

> **Note**
>
> * 個々のクライアントインスタンスと個々の FSx for Lustre オブジェクトストレージサーバー間のトラフィックは 5 Gbps に制限されています。FSx for Lustre ファイルシステムを支えるオブジェクトストレージサーバーの数については、「ファイルシステムの IP アドレス」を参照してください。

ファイルシステムストレージレイアウト

Lustre のすべてのファイルデータは、オブジェクトストレージターゲット (OST) と呼ばれるストレージボリュームに格納されます。すべてのファイルメタデータ (ファイル名、タイムスタンプ、アクセス許可などを含む) は、メタデータターゲット (MDT) と呼ばれるストレージボリュームに保存されます。Amazon FSx for Lustre ファイルシステムは、1 つ以上の MDT と複数の OST で設定されます。Amazon FSx for Lustre は、ストレージ容量とスループットと IOPS 負荷のバランスをとるために、ファイルシステムを設定する OST にファイルデータを分散します。

ファイルシステムを設定する MDT および OST のストレージ使用状況を表示するには、ファイルシステムがマウントされているクライアントから次のコマンドを実行します。

```
lfs df -h mount/path
```

このコマンドの出力は以下のようになります。

Example

```
UUID                       bytes        Used   Available Use% Mounted on
mountname-MDT0000_UUID      68.7G        5.4M        68.7G   0% /fsx[MDT:0]
mountname-OST0000_UUID       1.1T        4.5M         1.1T   0% /fsx[OST:0]
mountname-OST0001_UUID       1.1T        4.5M         1.1T   0% /fsx[OST:1]

filesystem_summary:          2.2T        9.0M         2.2T   0% /fsx
```

ファイルシステム内のデータのストライピング

ファイルストライピングにより、ファイルシステムのスループットパフォーマンスを最適化できます。Amazon FSx for Lustre は、データがすべてのストレージサーバーから確実に提供されるように、OST 間で自動的にファイルを分散します。複数の OST にまたがるファイルのストライピング方法を設定することで、同じ概念をファイルレベルで適用できます。

ストライピングとは、ファイルを複数のチャンクに分割して、異なる OST に格納できることを意味します。ファイルが複数の OST にストライプされると、ファイルへの読み取りまたは書き込みリクエストがそれらの OST にまたがって分散され、アプリケーションがそれを介して処理できる総スループットまたは IOPS が向上します。

Amazon FSx for Lustre ファイルシステムのデフォルトのレイアウトを次に示します。

- 2020 年 12 月 18 日より前に作成されたファイルシステムでは、デフォルトのレイアウトでストライプカウントが 1 に指定されています。つまり、異なるレイアウトを指定しない限り、標準の Linux ツールを使用して Amazon FSx for Lustre で作成された各ファイルは 1 つのディスクに格納されます。
- 2020 年 12 月 18 日以降に作成されたファイルシステムのデフォルトレイアウトは、プログレッシブファイルレイアウトであり、サイズが 1 GiB 未満のファイルは 1 つのストライプに保存され、大きいファイルにはストライプカウント 5 が割り当てられます。
- 2023 年 8 月 25 日以降に作成されたファイルシステムのデフォルトレイアウトは、プログレッシブファイルのレイアウト で説明されている 4 コンポーネントのプログレッシブファイルレイアウトです。
- すべてのファイルシステムでは、作成日に関係なく、Amazon S3 からインポートされたファイルはデフォルトのレイアウトを使用せずに、ファイルシステムの `ImportedFileChunkSize` パラメータにあるレイアウトを使用します。`ImportedFileChunkSize` より大きい S3 インポートされたファイルは、`(FileSize / ImportedFileChunksize) + 1` のストライプカウントで複数の OST に格納されます。`ImportedFileChunkSize` のデフォルト値は 1 GiB です。

`lfs getstripe` コマンドを使用してファイルまたはディレクトリのレイアウト設定を表示できます。

```
lfs getstripe path/to/filename
```

このコマンドは、ファイルのストライプカウント、ストライプサイズ、およびストライプオフセットを報告します。ストライプカウントは、ファイルがストライプされている OST の数です。ストライプサイズは、OST に保存されている連続データの量です。ストライプオフセットは、ファイルがストライプされる最初の OST のインデックスです。

ストライピング設定の変更

ファイルのレイアウトパラメータは、ファイルが最初に作成されたときに設定されます。`lfs setstripe` コマンドを使用すると、指定したレイアウトで新しい空のファイルを作成します。

```
lfs setstripe filename --stripe-count number_of_OSTs
```

lfs setstripe コマンドは、新しいファイルのレイアウトにのみ影響します。これを使用して、ファイルを作成する前にファイルのレイアウトを指定します。ディレクトリのレイアウトを定義することもできます。ディレクトリに設定されると、そのレイアウトはそのディレクトリに追加されたすべての新しいファイルに適用されますが、既存のファイルには適用されません。作成した新しいサブディレクトリも新しいレイアウトを継承し、そのサブディレクトリ内に作成した新しいファイルまたはディレクトリに適用されます。

既存のファイルのレイアウトを変更するには、lfs migrate コマンドを使用します。このコマンドは、必要に応じてファイルをコピーし、コマンドで指定したレイアウトに従ってコンテンツを配信します。例えば、ファイルに追加されたファイルやサイズが増加しても、ストライプカウントは変更されないため、ファイルのレイアウトを変更するにはそれらを移行する必要があります。または、lfs setstripe コマンドを使用して、レイアウトを指定し、元のコンテンツを新しいファイルにコピーし、新しいファイルの名前を変更して元のファイルと置き換えます。

デフォルトのレイアウト設定がワークロードに最適ではない場合があります。例えば、数十個の OST と多数のマルチギガバイトファイルがあるファイルシステムでは、デフォルトのストライプカウント値である 5 OST を超えるファイルをストライピングすることで、パフォーマンスが向上します。ストライプカウントの少ない大きなファイルを作成すると、I/O パフォーマンスのボトルネックが発生し、OST がいっぱいになる可能性もあります。この場合、ファイルのストライプカウントが多いディレクトリを作成できます。

大きなファイル (特にサイズが 1 ギガバイトを超えるファイル) のストライプレイアウトを設定することは、次の理由で重要です。

- 大きなファイルの読み取りと書き込み時に、複数の OST とその関連サーバーが IOPS、ネットワーク帯域幅、および CPU リソースを提供できるようにすることで、スループットが向上します。
- OST の小さなサブセットが全体的なワークロードパフォーマンスを制限するホットスポットになる可能性を低減します。
- 1 つの大きなファイルが OST を埋め尽くし、ディスクフルエラーを引き起こす可能性を防ぎます。

すべてのユースケースに単一の最適なレイアウト設定はありません。ファイルレイアウトに関する詳細なガイダンスについては、「Lustre.org ドキュメント」の「ファイルレイアウト (ストライピング) と空き領域の管理」を参照してください。一般的なガイドラインを次に示します。

- ストライプのレイアウトは、大きなファイル、特にファイルのサイズが数百メガバイト以上のユースケースで最も重要です。このため、新しいファイルシステムのデフォルトのレイアウトでは、サイズが 1 GiB を超えるファイルに対してストライプカウント 5 が割り当てられます。
- ストライプカウントは、大きなファイルをサポートするシステム用に調整する必要があるレイアウトパラメータです。ストライプカウントは、ストライプファイルのチャンクを保持する OST ボリュームの数を指定します。例えば、ストライプ数が 2、ストライプサイズが 1 MiB の場合、Lustre はファイルの代替の 1 MiB チャンクを 2 つの OST のそれぞれに書き込みます。
- 有効なストライプカウントは、実際の OST ボリュームの数と指定したストライプカウント値のうち小さい方です。特別なストライプカウント値の -1 を使用できます。これは、ストライプをすべての OST ボリュームに配置する必要があることを示します。
- 特定の操作では、ファイルが小さすぎてすべての OST ボリュームの容量を消費できない場合でも、Lustre はレイアウト内のすべての OST へのネットワークラウンドトリップを必要とするため、小さなファイルに対して大きなストライプカウントを設定することは最適ではありません。
- プログレッシブファイルレイアウト (PFL) を設定して、ファイルのレイアウトをサイズに応じて変更することができます。PFL 設定では、各ファイルに対して明示的に設定しなくても、大小のファイルを組み合わせたファイルシステムの管理を簡素化できます。詳細については、「プログレッシブファイルのレイアウト」を参照してください。
- ストライプサイズは、デフォルトで 1 MiB です。ストライプオフセットを設定すると、特殊な状況では便利ですが、通常は指定しないままにしておき、デフォルトを使用するのが最善です。

プログレッシブファイルのレイアウト

ディレクトリのプログレッシブファイルレイアウト (PFL) 設定を指定して、小さなファイルと大きなファイルに対して異なるストライプ設定を指定してから、それを入力できます。例えば、データが新しいファイルシステムに書き込まれる前に、最上位ディレクトリに PFL を設定できます。

PFL 設定を指定するには、`lfs setstripe` コマンドで `-E` オプションを使用して、以下のコマンドのように、異なるサイズのファイルのレイアウトコンポーネントを指定します。

```
lfs setstripe -E 100M -c 1 -E 10G -c 8 -E 100G -c 16 -E -1 -c 32 /mountname/directory
```

このコマンドは、4 つのレイアウトコンポーネントを設定します。

- 最初のコンポーネント (`-E 100M -c 1`) は、最大 100 MiB のファイルのストライプカウント値 1 を示します。

- 2 番目のコンポーネント (`-E 10G -c 8`) は、サイズが 10 GiB までのファイルのストライプカウントを 8 であることを示します。
- 3 番目のコンポーネント (`-E 100G -c 16`) は、サイズが 100 GiB までのファイルのストライプカウントを 16 であることを示します。
- 4 番目の要素 (`-E -1 -c 32`) は、100 GiB を超えるファイルのストライプカウントが 32 であることを示しています。

> **Important**
>
> PFL レイアウトで作成されたファイルにデータを追加すると、そのレイアウトコンポーネントがすべて入力されます。例えば、上記の 4 コンポーネントコマンドで、1 MiB のファイルを作成し、その末尾にデータを追加すると、ファイルのレイアウトが展開され、ストライプカウントが -1 になります。これは、システム内のすべての OST を指します。これは、データがすべての OST に書き込まれるという意味ではありませんが、ファイル長の読み取りなどのオペレーションは、すべての OST に並行してリクエストを送信し、ファイルシステムに大きなネットワークロードを追加します。
>
> したがって、その後にデータを追加できる小またはミディアムの長さのファイルのストライプカウントを制限するように注意してください。通常、ログファイルは新しいレコードを追加することで増加するため、Amazon FSx for Lustre では、親ディレクトリで指定されたデフォルトのストライプ設定に関係なく、追加モードで作成されたファイルに、デフォルトのストライプカウント 1 が割り当てられます。

2023 年 8 月 25 日以降に作成された Amazon FSx for Lustre ファイルシステムのデフォルトの PFL 設定は、次のコマンドを実行して設定します。

```
lfs setstripe -E 100M -c 1 -E 10G -c 8 -E 100G -c 16 -E -1 -c 32 /mountname
```

中 ~ 大規模ファイルへの同時アクセスが多いワークロードを持つお客様は、前述の 4 つのコンポーネントのレイアウト例で示したように、ファイルサイズが小さい場合はストライプカウントが多いレイアウトを使用し、ファイルサイズが最大の場合はすべての OST にまたがるストライピングのレイアウトを使用することでメリットが得られます。

パフォーマンスと使用状況のモニタリング

Amazon FSx for Lustre は 1 分ごとに、各ディスク (MDT および OST) の使用状況メトリクスを Amazon CloudWatch に発行します。

ファイルシステムの総使用状況の詳細を表示するには、各メトリクスの Sum 統計を調べます。例えば、`DataReadBytes` 統計は、ファイルシステム内のすべての OST で見られる総読み取りスループットを報告します。同様に、`FreeDataStorageCapacity` 統計は、ファイルシステム内のファイルデータに使用可能なストレージ容量の合計を報告します。

ファイルシステムのパフォーマンスのモニタリングの詳細については、「Amazon FSx for Lustre ファイルシステムのモニタリング」を参照してください。

SSD および HDD ストレージクラスのパフォーマンス特性

SSD または HDD ストレージクラスを持つ FSx for Lustre ファイルシステムがサポートするスループットは、そのストレージ容量に比例します。Amazon FSx for Lustre ファイルシステムでは、数 TBps のスループット とと数百万の IOPS に拡張できます。Amazon FSx for Lustre では、何千ものコンピュートインスタンスから同じファイルまたはディレクトリへの同時アクセスもサポートしています。このアクセスにより、ハイパフォーマンスコンピューティング (HPC) で一般的な手法であるアプリケーションメモリからストレージへの迅速なデータチェックポイントが可能になります。ファイルシステムを作成した後、いつでも必要なときにスループットキャパシティとスループットキャパシティを増やすことができます。詳細については、「ストレージ容量の管理」を参照してください。

FSx for Lustre ファイルシステムは、ネットワーク I/O クレジットメカニズムを使用してバースト読み取りスループットの値を提供し、平均帯域幅使用率に基づいて、ネットワーク帯域幅を割り当てます。インスタンスでは、帯域幅がベースライン帯域幅を下回るとクレジットを獲得し、クレジットをネットワークデータ転送を実行するときに使用できます。

以下の表は、SSD および HDD ストレージクラスを使用した FSx for Lustre の展開オプションが想定して設計されたパフォーマンスを示しています。

SSD ストレージオプションのファイルシステムパフォーマンス

デプロイタイプ	ネットワークスループット (MBps/TiB のプロビジョニングされたストレージ)		ネットワーク IOPS (プロビジョニングされたストレージの IOPS / TiB)	キャッシュストレージ (GiB の RAM / TiB のストレージのプロビジョニング)	ファイルオペレーションあたりのディスクレイテンシー (ミリ秒、P50)	ディスクスループット (MBps/TiB、プロビジョニングされたストレージまたは SSD キャッシュ)	
	[Baseline] (ベースライン)	[Burst] (バースト)				[Baseline] (ベースライン)	バースト
SCRATCH_2	200	1300	数万規模のベースライン 数十万規模のバースト	6.7	メタデータ: sub-ms データ: sub-ms	200 (読み取り) 100 (書き込み)	-
PERSISTENT-125	320	1300		3.4		125	500
PERSISTENT-250	640	1300		6.8		250	500
PERSISTENT-500	1300	-		13.7		500	-
PERSISTENT-1000	2600	-		27.3		1000	-

HDD ストレージオプションのファイルシステムパフォーマンス

デプロイタイプ	ネットワークスループット (MBps/TiB のストレージまたは SSD キャッシュのプロビジョニング)		ネットワーク IOPS (プロビジョニングされたストレージの IOPS / TiB)	キャッシュストレージ (GiB の RAM / TiB のストレージのプロビジョニング)	ファイルオペレーションあたりのディスクレイテンシー (ミリ秒、P50)	ディスクスループット (MBps/TiB、プロビジョニングされたストレージまたは SSD キャッシュ)	
	[Baseline] (ベースライン)	[Burst] (バースト)				[Baseline] (ベースライン)	バースト
PERSISTENT-12							
HDD ストレージ	40	375*	数万規模のベースライン 数十万規模のバースト	0.4 memory	メタデータ: sub-ms データ: 一桁ミリ秒	12	80 (読み取り) 50 (書き込み)
SSD キャッシュの読み取り	200	1,900		200 SSD キャッシュ	データ: sub-ms	200	-
PERSISTENT-40							
HDD ストレージ	150	1,300*	数万規模のベースライン 数十万規模のバースト	1.5	メタデータ: sub-ms データ: 一桁ミリ秒	40	250 (読み取り) 150 (書き込み)
SSD キャッシュの読み	750	6500		200 SSD	データ: sub-	200	-

旧世代の SSD ストレージオプションのファイルシステムパフォーマンス

デプロイタイプ	ネットワークスループット (プロビジョニングされたストレージの TiB あたり MBps)		ネットワーク IOPS (プロビジョニングされたストレージの TiB あたりの IOPS)	キャッシュストレージ (プロビジョニングされたストレージの TiB あたり GiB)	ファイルオペレーションあたりのディスクレイテンシー (ミリ秒、P50)	ディスクスループット (プロビジョニングされたストレージまたは SSD キャッシュの TiB あたり MBps)	
	[Baseline] (ベースライン)	[Burst] (バースト)				[Baseline] (ベースライン)	[Burst] (バースト)
PERSISTENT-50	250	1,300*	数万規模のベースライン	2.2 RAM	メタデータ: sub-ms	50	240
PERSISTENT-100	500	1,300*		4.4 RAM	データ: sub-ms	100	240
PERSISTENT-200	750	1,300*	数十万規模のバースト	8.8 RAM		200	240

> **Note**
>
> * 以下の永続的ファイルシステムは、ストレージの 1 TiB あたり最大 530 MBps のネットワークバースト AWS リージョン を提供します。アフリカ (ケープタウン) 、アジアパシフィック (香港) 、アジアパシフィック (大阪) 、アジアパシフィック (シンガポール) 、カナダ (中部) 、欧州 (フランクフルト) 、欧州 (ロンドン) 、欧州 (ミラノ) 、欧州 (ストックホルム) 、中東 (バーレーン) 、南米 (サンパウロ) 、中国、米国西部 (ロサンゼルス) 。

例: ベースラインとバーストスループットの集計

次の例は、ストレージ容量とディスクスループットがファイルシステムのパフォーマンスに与える影響を示しています。

ストレージ容量が 4.8 TiB、ストレージ単位あたりのスループットが 1 TiB あたり 50 MBps の永続型ファイルシステムでは、合計ベースラインディスクスループットが 240 MBps、バーストディスクスループットが 1.152 GBps となります。

ファイルシステムのサイズについては、Amazon FSx for Lustre は、ファイルオペレーションに対して一貫したミリ秒未満のレイテンシーを提供します。

インテリジェント階層化 ストレージクラスのパフォーマンス特性

FSx for Lustre のインテリジェント階層化ストレージクラスは、従来 HDD ベースまたは HDD/SSD 混在型の高性能ファイルストレージファイルシステムで実行されるワークロード向けに、弾力性がありコスト最適化されたストレージを提供します。インテリジェント階層化ストレージクラスを使用するファイルシステムは、完全に弾力性のある、インテリジェントに階層化されたリージョナルストレージを利用しており、ワークロードの変化に応じて自動的に容量が増減します。データの階層化方法については、「インテリジェント階層化ストレージクラスがデータを階層化する方法」を参照してください。

インテリジェント階層化ストレージクラスを使用する FSx for Lustre ファイルシステムがサポートするスループットは、ストレージ容量に依存しません。インテリジェント階層化ファイルシステムは、複数 TBps のスループットおよび数百万の IOPS までスケール可能です。インテリジェント階層化ストレージクラスを使用するファイルシステムは、頻繁にアクセスされるデータへの低レイテンシアクセスのために、オプションとしてプロビジョニングされた SSD 読み取りキャッシュも提供します。デフォルトでは、Amazon FSx for Lustre は頻繁にアクセスされるメタデータ用に SSD 読み取りキャッシュをプロビジョニングします。ほとんどのワークロードは読み込み量が多く、データセッ

ト全体の小さなサブセットのみと常にアクティブに連携する傾向があるため、Intelligent-Tiering ストレージクラスを使用するファイルシステムで Intelligent-Tiering ストレージと SSD リードキャッシュのハイブリッドモデルを使用することで、ほとんどのワークロードに対して SSD ファイルシステムに相当するレベルで実行されるストレージを提供しつつ、SSD および HDD ストレージクラスよりもストレージコストを削減できます。

Intelligent-Tiering ファイルシステムへのデータの読み取りと書き込み、特に最近または頻繁にアクセスされていないことでファイルサーバーのインメモリキャッシュに存在していないデータの場合、パフォーマンスは SSD 読み取りキャッシュのサイズによって異なります。Intelligent-Tiering ストレージからのデータアクセスでは、time-to-first-byte レイテンシーが約数十ミリ秒、リクエストあたりのコストがかかります。一方、SSD リードキャッシュからのアクセスでは、ミリ秒未満のレイテンシーとなり、リクエストあたりのコストなしで返されます。

ファイルシステムの SSD 読み取りキャッシュのサイズを設定するときは、ワークロード内のアクセス頻度の高いデータセットのサイズと、アクセス頻度の低いデータの読み取りに対するワークロードの高いレイテンシーの感度の両方を考慮する必要があります。ファイルシステムの作成後に SSD 読み取りキャッシュのサイズ設定モード間を切り替えて、キャッシュをスケールアップまたはスケールダウンできます。SSD 読み取りキャッシュを変更する方法の詳細については、「プロビジョニングされた SSD 読み取りキャッシュの管理」を参照してください。

FSx for Lustre がインテリジェント階層化ストレージにデータのブロックを書き込むと、書き込みリクエストが発生します。ファイルシステムへのデータ書き込み時には、書き込みリクエストが集約されてインテリジェント階層化ストレージへ書き込まれるため、スループットが向上し、リクエストコストが削減されます。読み取りは、ファイルサーバーのインメモリキャッシュ、SSD 読み取りキャッシュ、またはインテリジェント階層化ストレージから直接提供できます。インテリジェント階層化ストレージから読み取りが提供されると、取得されたデータのブロックごとに読み取りリクエストが発生します。データを順番に読み取ると、FSx for Lustre はパフォーマンスを向上させるためにデータをプリフェッチします。

インテリジェント階層化ストレージクラスを使用するファイルシステムのインメモリキャッシュからのデータは、ネットワーク I/O としてリクエスト元のクライアントに直接提供されます。 クライアントがインメモリキャッシュにないデータにアクセスすると、SSD リードキャッシュまたはインテリジェント階層化ストレージからディスク I/O として読み取り、ネットワーク I/O としてクライアントに提供されます。

インテリジェント階層化のファイルシステムパフォーマンス

以下の表は、FSx for Lustre のインテリジェント階層化ストレージクラスを使用するファイルシステム向けに設計された性能を示しています。

プロビジョンドスループットキャパシティ (MBps)	ネットワークスループット (MBps)		ネットワーク IOPS	インメモリキャッシュストレージ (GB)	SSD キャッシュディスクの最大スループット (MBps)	SSD キャッシュディスクの最大 IOPS	
	[Baseline] (ベースライン)	[Burst] (バースト)				[Baseline] (ベースライン)	[Burst] (バースト)
4000 ごと	12500	-	数十万	76.8	4000	160000	-

パフォーマンスのヒント

Amazon FSx for Lustre を使用する場合は、次のパフォーマンスのヒントに留意してください。サービスの制限については、「Amazon FSx for Lustre の Service quotas」を参照してください。

- 平均 I/O サイズ - Amazon FSx for Lustre はネットワークファイルシステムであるため、各ファイルオペレーションはクライアントと Amazon FSx for Lustre の間でラウンドトリップされるため、レイテンシーのオーバーヘッドが小さくなります。このオペレーションあたりのレイテンシーのため、通常は平均 I/O サイズの増加に応じて全体のスループットが向上します。大量のデータにオーバーヘッドが分散するためです。
- リクエストモデル -ファイルシステムへの非同期書き込みを有効にすることで、保留中の書き込みオペレーションは、Amazon FSx for Lustre に非同期で書き込まれる前に、Amazon EC2 インスタンスでバッファリングされます。非同期書き込みは、通常レイテンシーが低くなります。非同期書き込みを実行するとき、カーネルはキャッシュの追加のメモリを使用します。同期書き込みを有効にしているファイルシステムは、Amazon FSx for Lustre に同期リクエストを発行します。各オペレーションはクライアントと Amazon FSx for Lustre の間のラウンドトリップを通過します。

  > **Note**
  >
  > 選択したリクエストモデルでは、整合性 (複数の Amazon EC2 インスタンスを使用している場合) と速度にトレードオフがあります。

- ディレクトリサイズの制限 – 永続 2 FSx for Lustre ファイルシステムで最適なメタデータパフォーマンスを実現するには、各ディレクトリを 100K ファイル未満に制限します。ディレクトリ内のファイル数を制限すると、ファイルシステムが親ディレクトリのロックを取得するのに必要な時間が短縮されます。
- Amazon EC2 インスタンス - 多数の読み取りおよび書き込みオペレーションを実行するアプリケーションは、そうでないアプリケーションよりも多くのメモリまたはコンピューティング容量ーを必要とします。コンピューティングを多用するワークロードのために Amazon EC2 インスタンスを起動するときは、アプリケーションが必要とする量のリソースを持つインスタンスタイプを選択します。Amazon FSx for Lustre ファイルシステムのパフォーマンス特性は、Amazon EBS 最適化インスタンスの使用に依存しません。
- 最適なパフォーマンスを得るために推奨されるクライアントインスタンスの調整
  1. メモリが 64 GiB を超えるクライアントインスタンスタイプでは、次の調整を適用することをお勧めします。

     ```
     sudo lctl set_param ldlm.namespaces.*.lru_max_age=600000
     ```

```
sudo lctl set_param ldlm.namespaces.*.lru_size=<100 * number_of_CPUs>
```

2. 64 vCPU コアを超えるクライアントインスタンスタイプでは、次の調整を適用することをお勧めします。

```
echo "options ptlrpc ptlrpcd_per_cpt_max=32" >> /etc/modprobe.d/modprobe.conf
echo "options ksocklnd credits=2560" >> /etc/modprobe.d/modprobe.conf

# reload all kernel modules to apply the above two settings
sudo reboot
```

クライアントをマウントした後、次の調整を適用する必要があります。

```
sudo lctl set_param osc.*OST*.max_rpcs_in_flight=32
sudo lctl set_param mdc.*.max_rpcs_in_flight=64
sudo lctl set_param mdc.*.max_mod_rpcs_in_flight=50
```

3. ディレクトリリスト (ls) のパフォーマンスを最適化するには、次の調整を適用する必要があります。

```
sudo lctl set_param llite.*.statahead_max=512
sudo lctl set_param llite.*.statahead_agl=1
if sudo lctl get_param llite.*.statahead_xattr > /dev/null 2>&1; then
    sudo lctl set_param llite.*.statahead_xattr=1
else
    echo "Warning: Xattr statahead is not supported on this Lustre client. Please
 upgrade to the latest Lustre 2.15 client to apply this tuning"
fi
```

注: `lctl set_param` は再起動すると持続しないことが知られています。これらのパラメータはクライアント側から永続的に設定することはできないため、ブート cron ジョブを実装して、お勧めの調整を使用して設定することをお勧めします。

- OST 間でのワークロードバランス - 場合によっては、ワークロードによってファイルシステムが提供できる総スループット (ストレージ TiB あたり 200 MBps) が駆動されないことがあります。その場合は、CloudWatch メトリクスを使用して、ワークロードの I/O パターンの不均衡によってパフォーマンスが影響を受けるかどうかをトラブルシューティングできます。これが原因であるかどうかを特定するには、Amazon FSx for Lustre の最大 CloudWatch メトリクスを参照してください。

場合によっては、この統計は 240 MBps のスループット (単一の 1.2 TiB Amazon FSx for Lustre ディスクのスループットキャパシティ) 以上の負荷を示します。このような場合、ワークロードはディスク全体に均等に分散されません。この場合、`lfs setstripe` コマンドを実行して、ワークロードが頻繁にアクセスしているファイルのストライピングを変更します。最適なパフォーマンスを得るには、ファイルシステムを設定するすべての OST で、スループット要件が高いファイルをストライプ化します。

ファイルをデータリポジトリからインポートする場合は、別の方法を使用して、高スループットファイルを OST 全体で均等にストライプできます。そのためには、次の Amazon FSx for Lustre ファイルシステムを作成するときの `ImportedFileChunkSize` パラメータを変更できます。

例えば、ワークロードが 7.0 TiB ファイルシステム (6 x 1.17 TiB OST で設定されている) を使用し、2.4 GiB ファイル間で高いスループットを駆動する必要があるとします。この場合、`ImportedFileChunkSize` の値を `(2.4 GiB / 6 OSTs) = 400 MiB` に設定できます。これにより、ファイルがファイルシステムの OST 全体に均等に分散されます。

- Lustre メタデータ IOPS 用クライアント – ファイルシステムにメタデータ設定が指定されている場合は、Amazon Linux 2023、Amazon Linux 2、Red Hat/Rocky Linux 8.9、8.10、または 9.x、CentOS 8.9 または 8.10、6.2、6.5 または 6.8 カーネルを持つ Ubuntu 22 以降、または Ubuntu 20 のいずれかの OS バージョンを持つ Lustre 2.15 クライアントまたは Lustre 2.12 クライアントをインストールすることをお勧めします。

インテリジェント階層化 のパフォーマンスに関する考慮事項

インテリジェント階層化 ストレージクラスを使用してファイルシステムを操作する場合の重要なパフォーマンス上の考慮事項を以下に示します:

- I/O サイズが小さいデータを読み取るワークロードは、インテリジェント階層化 ストレージ階層からのレイテンシーが高いため、I/O サイズが大きいワークロードと同じスループットを実現するために、より高い同時実行性を必要とし、より多くのリクエストコストが発生します。小さい IO サイズを使用する場合、より高い同時実行性とスループットをサポートするのに十分な大きさの SSD 読み取りキャッシュを設定することをお勧めします。
- インテリジェント階層化ファイルシステムでクライアントが駆動できる最大ディスク IOPS は、ワークロードの特定のアクセスパターンと、SSD 読み取りキャッシュをプロビジョニングしたかどうかによって異なります。ランダムアクセスのワークロードでは、データが SSD リードキャッシュにキャッシュされている場合、クライアントは通常、データがキャッシュにない場合よりもはるかに高い IOPS を駆動できます。

- インテリジェント階層化ストレージクラスは先行読み込みをサポートし、シーケンシャル読み込みリクエストのパフォーマンスを最適化します。データアクセスパターンを可能な限り順番に設定して、データのプリフェッチとパフォーマンスの向上を可能にすることをお勧めします。

ファイルシステムへのアクセス

Amazon FSx を使用すると、 Direct Connect または VPN 経由でデータをインポートすることで、コンピューティング負荷の高いワークロードをオンプレミスから Amazon Web Services クラウドにバーストできます。オンプレミスから Amazon FSx ファイルシステムにアクセスし、必要に応じてデータをファイルシステムにコピーし、クラウド内のインスタンスでコンピューティング集約型のワークロードを実行できます。

次のセクションでは、Linux インスタンスに Amazon FSx for Lustre ファイルシステムをアクセスする方法を説明します。加えて、システムの再起動後に `fstab` ファイルを使用してファイルシステムを自動的に再マウントする方法を説明します。

ファイルシステムをマウントする前に、関連する AWS リソースを作成、設定、および起動する必要があります。詳細な手順については、「Amazon FSx for Lustre の使用開始」を参照してください。次に、コンピューティングインスタンスに Lustre クライアントをインストールして設定できます。

トピック

Lustre ファイルシステムとクライアントカーネルの互換性

FSx for Lustre ファイルシステムには、クライアントインスタンスの Linux カーネルバージョンと互換性のある Lustre バージョンを使用することを強くお勧めします。

Amazon Linux クライアント

オペレーティングシステム	OS バージョン	最小カーネルバージョン	最大カーネルバージョン	Lustre クライアントバージョン	Lustre ファイルシステムバージョン		
					2.10	2.12	2.15
Amazon Linux 2023	6.12	*	*	2.15	いいえ	はい	はい
	6.1	6.1.79-99.167	6.1.79-99.167+	2.15	いいえ	はい	はい
Amazon Linux 2	5.10	5.10.144-127.601	5.10.144-127.601+	2.12	はい	はい	はい
			<5.10.144-127.601	2.10	はい	はい	いいえ
	5.4	5.4.214-120.368	5.4.214-120.368+	2.12	はい	はい	はい
			<5.4.214-120.368	2.10	はい	はい	いいえ
	4.14	4.14.294-220.533	4.14.294-220.533+	2.12	はい	はい	はい
			<4.14.294-220.533	2.10	はい	はい	いいえ

Ubuntu クライアント

オペレーティングシステム	OS バージョン	最小カーネルバージョン	最大カーネルバージョン	Lustre クライアントバージョン	Lustre ファイルシステムバージョン		
					2.10	2.12	2.15
Ubuntu	24	6.14.0-1012	6.14.0*	2.15	いいえ	はい	はい
		6.8.0-1024	6.8.0*	2.15	いいえ	はい	はい
	22	6.8.0-1017	6.8.0*	2.15	いいえ	はい	はい
		6.5.0-1023	6.5.0*	2.15	いいえ	はい	はい
		6.2.0-1017	6.2.0*	2.15	いいえ	はい	はい
		5.15.0-1015-aws	5.15.0-1051-aws	2.12	はい	はい	はい
	20	5.15.0-1015-aws	5.15.0*	2.12	はい	はい	はい
		5.4.0-1011-aws	5.13.0-1031-aws	2.10	はい	はい	いいえ

RHEL/CentOS/Rocky Linux クライアント

オペレーティングシステム	OS バージョン	アーキテクチャ	最小カーネルバージョン	最大カーネルバージョン	Lustre クライアントバージョン	Lustre ファイルシステムバージョン		
						2.10	2.12	2.15
RHEL/ Rocky Linux	9.7	Arm + x86	5.14.0-6 1.5.1	5.14.0-6 1*	2.15	いいえ	はい	はい
	9.6	Arm + x86	5.14.0-5 0.12.1	5.14.0-5 0*	2.15	いいえ	はい	はい
	9.5	Arm + x86	5.14.0-5(3.19.1	5.14.0-5(3*	2.15	いいえ	はい	はい
	9.4	Arm + x86	5.14.0-4 7.13.1	5.14.0-4 7*	2.15	いいえ	はい	はい
	9.3	Arm + x86	5.14.0-3(2.18.1	5.14.0-3(2.18.1	2.15	いいえ	はい	はい
	9.0	Arm + x86	5.14.0-7(.13.1	5.14.0-7(.30.1	2.15	いいえ	はい	はい
RHEL/ Cent OS/ Rocky Li	8.10	Arm + x86	4.18.0-5 3	4.18.0-5 3*	2.12	はい	はい	はい
	8.9	Arm + x86	4.18.0-5 3*	4.18.0-5 3*	2.12	はい	はい	はい
	8.8	Arm + x86	4.18.0-4 7*	4.18.0-4 7*	2.12	はい	はい	はい

オペレーティングシステム	OS バージョン	アーキテクチャ	最小カーネルバージョン	最大カーネルバージョン	Lustre クライアントバージョン	Lustre ファイルシステムバージョン		
	8.7	Arm + x86	4.18.0-42 5*	4.18.0-42 5*	2.12	はい	はい	はい
	8.6	Arm + x86	4.18.0-37 2*	4.18.0-37 2*	2.12	はい	はい	はい
	8.5	Arm + x86	4.18.0-34 8*	4.18.0-34 8*	2.12	はい	はい	はい
	8.4	Arm + x86	4.18.0-30 5*	4.18.0-30 5*	2.12	はい	はい	はい
RHEL/ CentOS	8.3	Arm + x86	4.18.0-24 0*	4.18.0-24 0*	2.10	はい	はい	いいえ
	8.2	Arm + x86	4.18.0-19 3*	4.18.0-19 3*	2.10	はい	はい	いいえ
	7.9	x86	3.10.0-11 60*	3.10.0-11 60*	2.12	はい	はい	はい
	7.8	x86	3.10.0-11 27*	3.10.0-11 27*	2.10	はい	はい	いいえ
	7.7	x86	3.10.0-10 62*	3.10.0-10 62*	2.10	はい	はい	いいえ
CentOS	7.9	Arm	4.18.0-19 3*	4.18.0-19 3*	2.12	はい	はい	はい
	7.8	Arm	4.18.0-14 7*	4.18.0-14 7*	2.12	はい	はい	はい

Lustre クライアントのインストール

Linux インスタンスから Amazon FSx for Lustre ファイルシステムをマウントするには、まずオープンソースの Lustre クライアントをインストールします。次に、オペレーティングシステムのバージョンに応じて、次のいずれかの手順を使用します。カーネルのサポート情報については、「Lustre ファイルシステムとクライアントカーネルの互換性」を参照してください。

EFA (Elastic Fabric Adapter) で Lustre クライアントを使用している場合は、「」を参照してくださいEFA クライアントの設定。

コンピューティングインスタンスがインストール手順で指定された Linux カーネルを実行しておらず、カーネルを変更できない場合は、独自の Lustre クライアントを構築できます。詳細については、Lustre Wiki の「Lustre のコンパイル」を参照してください。

Amazon Linux

Amazon Linux 2023 で Lustre クライアントをインストールするには

1. クライアントのターミナルを開きます。
2. 次のコマンドを実行して、コンピューティングインスタンスで現在実行されているカーネルを特定します。

   ```
   uname -r
   ```

3. システムレスポンスを確認し、Amazon Linux 2023 に Lustre クライアントをインストールするための以下の最小カーネル要件と比較します。

   - 6.12 カーネルの最小要件 - 6.12*
   - 6.1 カーネルの最小要件 - 6.1.79-99.167.amzn2023

   EC2 インスタンスが最小カーネル要件を満たしている場合は、ステップを進め、Lustre クライアントをインストールします。

   コマンドがカーネルの最小要件に満たない結果を返す場合は、カーネルを更新し、次のコマンドを実行して Amazon EC2 インスタンスを再起動します。

   ```
   sudo dnf -y update kernel && sudo reboot
   ```

   uname -r コマンドを使用して、カーネルが更新されていることを確認します。

4. Lustre クライアントをダウンロードしてインストールするには、以下のコマンドを使用します。

```
sudo dnf install -y lustre-client
```

Amazon Linux 2 で Lustre クライアントをインストールするには

1. クライアントのターミナルを開きます。
2. 次のコマンドを実行して、コンピューティングインスタンスで現在実行されているカーネルを特定します。

```
uname -r
```

3. システムレスポンスを確認し、Amazon Linux 2 に Lustre クライアントをインストールするための以下の最小カーネル要件と比較します。

   - 5.10 カーネル最小要件 - 5.10.144-127.601.amzn2
   - 5.4 カーネル最小要件 - 5.4.214-120.368.amzn2
   - 4.14 カーネル最小要件 - 4.14.294-220.533.amzn2

   EC2 インスタンスが最小カーネル要件を満たしている場合は、ステップを進め、Lustre クライアントをインストールします。

   コマンドがカーネルの最小要件に満たない結果を返す場合は、カーネルを更新し、次のコマンドを実行して Amazon EC2 インスタンスを再起動します。

```
sudo yum -y update kernel && sudo reboot
```

   uname -r コマンドを使用して、カーネルが更新されていることを確認します。
4. Lustre クライアントをダウンロードしてインストールするには、以下のコマンドを使用します。

```
sudo amazon-linux-extras install -y lustre
```

   カーネルをカーネル最小要件にアップグレードできない場合は、以下のコマンドでレガシー 2.10 クライアントをインストールできます。

```
sudo amazon-linux-extras install -y lustre2.10
```

Amazon Linux で Lustre クライアントをインストールするには

1. クライアントのターミナルを開きます。
2. 次のコマンドを実行して、コンピューティングインスタンスで現在実行されているカーネルを特定します。Lustre クライアントには Amazon Linux カーネル `4.14, version 104` 以上が必要です。

   ```
   uname -r
   ```
3. 次のいずれかを行います。

   - コマンドが `4.14.104-78.84.amzn1.x86_64` または 4.14 以降のバージョンに戻った場合、次のコマンドを使用して Lustre クライアントをダウンロードしてインストールします。

     ```
     sudo yum install -y lustre-client
     ```
   - コマンドが `4.14.104-78.84.amzn1.x86_64` より小さい結果を返した場合、次のコマンドを実行してカーネルを更新し、Amazon EC2 インスタンスを再起動します。

     ```
     sudo yum -y update kernel && sudo reboot
     ```

     uname -r コマンドを使用して、カーネルが更新されていることを確認します。次に、先の説明に従って Lustre クライアントをダウンロードしてインストールします。

CentOS、Rocky Linux、および Red Hat

Red Hat および Rocky Linux 9.0 または 9.3 ~ 9.7 にLustreクライアントをインストールするには

Red Hat Enterprise Linux (RHEL)、Rocky Linux と互換性がある Lustre クライアントパッケージは、Amazon FSx Lustre クライアント yum パッケージリポジトリからインストールおよび更新できます。パッケージは、ダウンロード前またはダウンロード中に改ざんされていないことを確認するために署名されています。対応する公開キーをシステムにインストールしないと、リポジトリのインストールは失敗します。

Amazon FSx Lustre クライアント yum パッケージリポジトリを追加するには

1. クライアントのターミナルを開きます。
2. 次のコマンドを使用して、Amazon FSx をインストールします。

```
curl https://fsx-lustre-client-repo-public-keys.s3.amazonaws.com/fsx-rpm-public-key.asc -o /tmp/fsx-rpm-public-key.asc
```

3. 次のコマンドを使用して、キーをインポートします。

```
sudo rpm --import /tmp/fsx-rpm-public-key.asc
```

4. 次のコマンドを使用してリポジトリを追加し、パッケージマネージャーを更新します。

```
sudo curl https://fsx-lustre-client-repo.s3.amazonaws.com/el/9/fsx-lustre-client.repo -o /etc/yum.repos.d/aws-fsx.repo
```

Amazon FSx Lustre クライアント yum リポジトリを設定するには

Amazon FSx Lustre クライアント yum パッケージリポジトリは、サポートされている最新の Rocky Linux、RHEL 9 リリースとともに最初に出荷されたカーネルバージョンと互換性がある Lustre クライアントをインストールするようにデフォルトで設定されています。使用しているカーネルバージョンと互換性がある Lustre クライアントをインストールするには、リポジトリ設定ファイルを編集します。

このセクションでは、実行中のカーネルの判別方法、リポジトリ設定を編集する必要があるかどうか、および設定ファイルの編集方法について説明します。

1. 次のコマンドを使用して、コンピューティングインスタンスで現在実行されているカーネルを特定します。

```
uname -r
```

2. 次のいずれかを実行します。

   - コマンドを `5.14.0-611*` に返した場合では、リポジトリの設定を変更する必要はありません。「Lustre クライアントをインストールするには」の手順に進んでください。
   - コマンドが を返す場合は`5.14.0-570*`、Rocky Linux および RHEL 9.6 リリースのLustreクライアントを指すようにリポジトリ設定を編集する必要があります。

- コマンドが を返す場合は`5.14.0-503*`、Rocky Linux および RHEL 9.5 リリースの Lustreクライアントを指すようにリポジトリ設定を編集する必要があります。
- コマンドが を返す場合は`5.14.0-427*`、Rocky Linux および RHEL 9.4 リリースの Lustreクライアントを指すようにリポジトリ設定を編集する必要があります。
- コマンドが `5.14.0-362.18.1` を返した場合、Rocky Linux および RHEL 9.3 リリースの Lustre クライアントを指定するように、リポジトリ設定を編集する必要があります。
- コマンドが `5.14.0-70*` を返した場合、Rocky Linux および RHEL 9.0 リリースの Lustre クライアントを指定するように、リポジトリ設定を編集する必要があります。

3. 次のコマンドを使用して、特定のバージョンの RHEL を指すようにリポジトリ設定ファイルを編集します。*specific_RHEL_version* を、使用する必要がある RHEL バージョンに置き換えます。

```
sudo sed -i 's#9#specific_RHEL_version#' /etc/yum.repos.d/aws-fsx.repo
```

たとえば、リリース 9.6 を参照するには、次の例のように、 コマンド`9.6`で *specific_RHEL_version*を に置き換えます。

```
sudo sed -i 's#9#9.6#' /etc/yum.repos.d/aws-fsx.repo
```

4. 次のコマンドを使用して yum キャッシュをクリアします。

```
sudo yum clean all
```

Lustre クライアントをインストールするには

- 次のコマンドを使用してリポジトリからパッケージをインストールします。

```
sudo yum install -y kmod-lustre-client lustre-client
```

追加情報 (Rocky Linux および Red Hat 9.0 以降)

前述のコマンドは、Amazon FSx ファイルシステムをマウントして操作するために必要な 2 つのパッケージをインストールします。リポジトリには、ソースコードを含むパッケージやテストを含むパッケージなど、追加の Lustre パッケージが含まれており、必要に応じてインストールできます。リポジトリで使用できるすべてのパッケージを一覧表示するには、次のコマンドを使用します。

```
yum --disablerepo="*" --enablerepo="aws-fsx" list available
```

上流の出典コードの tarball と、適用したパッチのセットを含む出典 rpm をダウンロードするには、次のコマンドを使用します。

```
 sudo yumdownloader --source kmod-lustre-client
```

yum 更新を実行すると、使用可能な場合は新しいバージョンのモジュールがインストールされ、既存のバージョンが置き換えられます。現在インストールされているバージョンが更新時に削除されないようにするには、次のような行を追加して /etc/yum.conf ファイルを開きます。

```
installonlypkgs=kernel, kernel-PAE, installonlypkg(kernel), installonlypkg(kernel-
module),
              installonlypkg(vm), multiversion(kernel), kmod-lustre-client
```

このリストには、yum.conf man ページ、および kmod-lustre-client パッケージ で指定したデフォルトのインストール専用パッケージが含まれます。

CentOS および Red Hat 8.2 ~ 8.10、または Rocky Linux 8.4 ~ 8.10 に Lustre クライアントをインストールするには

Red Hat Enterprise Linux (RHEL)、Rocky Linux、および CentOS と互換性がある Lustre クライアントパッケージは、Amazon FSx Lustre クライアント yum パッケージリポジトリからインストールおよび更新できます。パッケージは、ダウンロード前またはダウンロード中に改ざんされていないことを確認するために署名されています。対応する公開キーをシステムにインストールしないと、リポジトリのインストールは失敗します。

Amazon FSx Lustre クライアント yum パッケージリポジトリを追加するには

1. クライアントのターミナルを開きます。
2. 次のコマンドを使用して、Amazon FSx をインストールします。

   ```
   curl https://fsx-lustre-client-repo-public-keys.s3.amazonaws.com/fsx-rpm-public-
   key.asc -o /tmp/fsx-rpm-public-key.asc
   ```

3. 次のコマンドを使用して、キーをインポートします。

   ```
   sudo rpm --import /tmp/fsx-rpm-public-key.asc
   ```

4. 次のコマンドを使用してリポジトリを追加し、パッケージマネージャーを更新します。

```
sudo curl https://fsx-lustre-client-repo.s3.amazonaws.com/el/8/fsx-lustre-
client.repo -o /etc/yum.repos.d/aws-fsx.repo
```

Amazon FSx Lustre クライアント yum リポジトリを設定するには

Amazon FSx Lustre クライアント yum パッケージリポジトリは、サポートされている最新の CentOS、Rocky Linux、および RHEL 8 リリースに最初に出荷されたカーネルバージョンと互換性がある Lustre クライアントをインストールするようにデフォルトで設定されています。使用しているカーネルバージョンと互換性がある Lustre クライアントをインストールするには、リポジトリ設定ファイルを編集します。

このセクションでは、実行中のカーネルの判別方法、リポジトリ設定を編集する必要があるかどうか、および設定ファイルの編集方法について説明します。

1. 次のコマンドを使用して、コンピューティングインスタンスで現在実行されているカーネルを特定します。

```
uname -r
```

2. 次のいずれかを実行します。

   - コマンドを `4.18.0-553*` に返した場合では、リポジトリの設定を変更する必要はありません。「Lustre クライアントをインストールするには」の手順に進んでください。
   - コマンドが `4.18.0-513*` を返した場合、CentOS、Rocky Linux、RHEL 8.9 リリースの Lustre クライアントを指定するように、リポジトリ設定を編集する必要があります。
   - コマンドが `4.18.0-477*` を返した場合、CentOS、Rocky Linux、RHEL 8.8 リリースの Lustre クライアントを指定するように、リポジトリ設定を編集する必要があります。
   - コマンドが `4.18.0-425*` を返した場合、CentOS、Rocky Linux、RHEL 8.7 リリースの Lustre クライアントを指定するように、リポジトリ設定を編集する必要があります。
   - コマンドが `4.18.0-372*` を返した場合、CentOS、Rocky Linux、および RHEL 8.6 リリースの Lustre クライアントを指定するように、リポジトリ設定を編集する必要があります。
   - コマンドが `4.18.0-348*` を返した場合、CentOS、Rocky Linux、および RHEL 8.5 リリースの Lustre クライアントを指定するように、リポジトリ設定を編集する必要があります。
   - コマンドが `4.18.0-305*` に返した場合、CentOS、Rocky Linux、および RHEL 8.4 リリースの Lustre クライアントを指すように、リポジトリ設定を編集する必要があります。

- コマンドが `4.18.0-240*` を返した場合、CentOS および RHEL 8.3 リリースの Lustre クライアントを指すように、リポジトリ設定を編集する必要があります。
- コマンドが `4.18.0-193*` を返した場合、CentOS および RHEL 8.2 リリースの Lustre クライアントを指すように、リポジトリ設定を編集する必要があります。

3. 次のコマンドを使用して、特定のバージョンの RHEL を指すようにリポジトリ設定ファイルを編集します。

```
sudo sed -i 's#8#specific_RHEL_version#' /etc/yum.repos.d/aws-fsx.repo
```

例えば、リリース 8.9 を指定するには、コマンド内の *specific_RHEL_version* を `8.9` に置き換えます。

```
sudo sed -i 's#8#8.9#' /etc/yum.repos.d/aws-fsx.repo
```

4. 次のコマンドを使用して yum キャッシュをクリアします。

```
sudo yum clean all
```

Lustre クライアントをインストールするには

- 次のコマンドを使用してリポジトリからパッケージをインストールします。

```
sudo yum install -y kmod-lustre-client lustre-client
```

追加情報 (CentOS、Rocky Linux、および Red Hat 8.2 以降)

前述のコマンドは、Amazon FSx ファイルシステムをマウントして操作するために必要な 2 つのパッケージをインストールします。リポジトリには、ソースコードを含むパッケージやテストを含むパッケージなど、追加の Lustre パッケージが含まれており、必要に応じてインストールできます。リポジトリで使用できるすべてのパッケージを一覧表示するには、次のコマンドを使用します。

```
yum --disablerepo="*" --enablerepo="aws-fsx" list available
```

上流の出典コードの tarball と、適用したパッチのセットを含む出典 rpm をダウンロードするには、次のコマンドを使用します。

```
sudo yumdownloader --source kmod-lustre-client
```

yum 更新を実行すると、使用可能な場合は新しいバージョンのモジュールがインストールされ、既存のバージョンが置き換えられます。現在インストールされているバージョンが更新時に削除されないようにするには、次のような行を追加して /etc/yum.conf ファイルを開きます。

```
installonlypkgs=kernel, kernel-PAE, installonlypkg(kernel), installonlypkg(kernel-
module),
              installonlypkg(vm), multiversion(kernel), kmod-lustre-client
```

このリストには、yum.conf man ページ、および kmod-lustre-client パッケージ で指定したデフォルトのインストール専用パッケージが含まれます。

CentOS と Red Hat 7.7、7.8 または 7.9 (x86_64 インスタンス) に Lustre クライアントをインストールするには

Red Hat Enterprise Linux (RHEL) および CentOS と互換性がある Lustre クライアントパッケージは、Amazon FSx Lustre クライアント yum パッケージリポジトリからインストールおよび更新できます。パッケージは、ダウンロード前またはダウンロード中に改ざんされていないことを確認するために署名されています。対応する公開キーをシステムにインストールしないと、リポジトリのインストールは失敗します。

Amazon FSx Lustre クライアント yum パッケージリポジトリを追加するには

1. クライアントのターミナルを開きます。
2. 次のコマンドを使用して、Amazon FSx rpm 公開キーをインストールします。

   ```
   curl https://fsx-lustre-client-repo-public-keys.s3.amazonaws.com/fsx-rpm-public-
   key.asc -o /tmp/fsx-rpm-public-key.asc
   ```

3. 次のコマンドを使用して、キーをインポートします。

   ```
   sudo rpm --import /tmp/fsx-rpm-public-key.asc
   ```

4. 次のコマンドを使用してリポジトリを追加し、パッケージマネージャーを更新します。

   ```
   sudo curl https://fsx-lustre-client-repo.s3.amazonaws.com/el/7/fsx-lustre-
   client.repo -o /etc/yum.repos.d/aws-fsx.repo
   ```

Amazon FSx Lustre クライアント yum リポジトリを設定するには

Amazon FSx Lustre クライアント yum パッケージリポジトリは、サポートされている最新の CentOS および RHEL 7 リリースに最初に出荷されたカーネルバージョンと互換性がある Lustre クライアントをインストールするようにデフォルトで設定されています。使用しているカーネルバージョンと互換性がある Lustre クライアントをインストールするには、リポジトリ設定ファイルを編集します。

このセクションでは、実行中のカーネルの判別方法、リポジトリ設定を編集する必要があるかどうか、および設定ファイルの編集方法について説明します。

1. 次のコマンドを使用して、コンピューティングインスタンスで現在実行されているカーネルを特定します。

   ```
   uname -r
   ```

2. 次のいずれかを実行します。

   - コマンドを `3.10.0-1160*` に返した場合では、リポジトリの設定を変更する必要はありません。「Lustre クライアントをインストールするには」の手順に進んでください。
   - コマンドが `3.10.0-1127*` を返した場合、CentOS および RHEL 7.8 リリースの Lustre クライアントを指定するように、リポジトリ設定を編集する必要があります。
   - コマンドが `3.10.0-1062*` を返した場合、CentOS および RHEL 7.7 リリースの Lustre クライアントを指定するように、リポジトリ設定を編集する必要があります。

3. 次のコマンドを使用して、特定のバージョンの RHEL を指すようにリポジトリ設定ファイルを編集します。

   ```
   sudo sed -i 's#7#specific_RHEL_version#' /etc/yum.repos.d/aws-fsx.repo
   ```

   リリース 7.8 をポイントするには、コマンド内の *`specific_RHEL_version`* と `7.8` を置換えます。

   ```
   sudo sed -i 's#7#7.8#' /etc/yum.repos.d/aws-fsx.repo
   ```

   リリース 7.7 をポイントするには、コマンド内の *`specific_RHEL_version`* と `7.7` を置換えます。

   ```
   sudo sed -i 's#7#7.7#' /etc/yum.repos.d/aws-fsx.repo
   ```

4. 次のコマンドを使用して yum キャッシュをクリアします。

```
sudo yum clean all
```

Lustre クライアントをインストールするには

- 次のコマンドを使用してリポジトリから Lustre クライアントパッケージをインストールします。

```
sudo yum install -y kmod-lustre-client lustre-client
```

追加情報 (CentOS および Red Hat 7.7 以降)

前述のコマンドは、Amazon FSx ファイルシステムをマウントして操作するために必要な 2 つのパッケージをインストールします。リポジトリには、ソースコードを含むパッケージやテストを含むパッケージなど、追加の Lustre パッケージが含まれており、必要に応じてインストールできます。リポジトリで使用できるすべてのパッケージを一覧表示するには、次のコマンドを使用します。

```
yum --disablerepo="*" --enablerepo="aws-fsx" list available
```

上流の出典コードの tarball と、適用したパッチのセットを含む出典 rpm をダウンロードするには、次のコマンドを使用します。

```
 sudo yumdownloader --source kmod-lustre-client
```

yum 更新を実行すると、使用可能な場合は新しいバージョンのモジュールがインストールされ、既存のバージョンが置き換えられます。現在インストールされているバージョンが更新時に削除されないようにするには、次のような行を追加して /etc/yum.conf ファイルを開きます。

```
installonlypkgs=kernel, kernel-big-mem, kernel-enterprise, kernel-smp,
              kernel-debug, kernel-unsupported, kernel-source, kernel-devel, kernel-
PAE,
              kernel-PAE-debug, kmod-lustre-client
```

このリストには、yum.conf man ページ、および kmod-lustre-client パッケージ で指定したデフォルトのインストール専用パッケージが含まれます。

CentOS 7.8 または 7.9 (Arm ベースの Graviton 搭載インスタンス) AWS にLustreクライアントをインストールするには

Arm ベースの AWS Graviton 搭載 EC2 インスタンスの CentOS 7 と互換性がある Amazon FSx Lustre クライアント yum パッケージリポジトリから、Lustre クライアントパッケージをインストールして更新できます。パッケージは、ダウンロード前またはダウンロード中に改ざんされていないことを確認するために署名されています。対応する公開キーをシステムにインストールしないと、リポジトリのインストールは失敗します。

Amazon FSx Lustre クライアント yum パッケージリポジトリを追加するには

1. クライアントのターミナルを開きます。
2. 次のコマンドを使用して、Amazon FSx rpm 公開キーをインストールします。

```
curl https://fsx-lustre-client-repo-public-keys.s3.amazonaws.com/fsx-rpm-public-key.asc -o /tmp/fsx-rpm-public-key.asc
```

```
curl https://fsx-lustre-client-repo-public-keys.s3.amazonaws.cn/fsx-rpm-public-key.asc -o /tmp/fsx-rpm-public-key.asc
```

3. 次のコマンドを使用して、キーをインポートします。

```
sudo rpm --import /tmp/fsx-rpm-public-key.asc
```

4. 次のコマンドを使用してリポジトリを追加し、パッケージマネージャーを更新します。

```
sudo curl https://fsx-lustre-client-repo.s3.amazonaws.com/centos/7/fsx-lustre-client.repo -o /etc/yum.repos.d/aws-fsx.repo
```

Amazon FSx Lustre クライアント yum リポジトリを設定するには

Amazon FSx Lustre クライアント yum パッケージリポジトリは、サポートされている最新の CentOS 7 リリースに最初に出荷されたカーネルバージョンと互換性がある Lustre クライアントをインストールするようにデフォルトで設定されています。使用しているカーネルバージョンと互換性がある Lustre クライアントをインストールするには、リポジトリ設定ファイルを編集します。

このセクションでは、実行中のカーネルの判別方法、リポジトリ設定を編集する必要があるかどうか、および設定ファイルの編集方法について説明します。

1. 次のコマンドを使用して、コンピューティングインスタンスで現在実行されているカーネルを特定します。

```
uname -r
```

2. 次のいずれかを実行します。

   - コマンドを `4.18.0-193*` に返した場合では、リポジトリの設定を変更する必要はありません。「Lustre クライアントをインストールするには」の手順に進んでください。
   - コマンドが `4.18.0-147*` を返した場合、CentOS 7.8 リリースの Lustre クライアントを指定するように、リポジトリ設定を編集する必要があります。

3. 次のコマンドを使用して、リポジトリ設定ファイルを CentOS 7.8 リリースをポイントするように編集します。

```
sudo sed -i 's#7#7.8#' /etc/yum.repos.d/aws-fsx.repo
```

4. 次のコマンドを使用して yum キャッシュをクリアします。

```
sudo yum clean all
```

Lustre クライアントをインストールするには

- 次のコマンドを使用してリポジトリからパッケージをインストールします。

```
sudo yum install -y kmod-lustre-client lustre-client
```

追加情報 (Arm ベースの Graviton 搭載 EC2 インスタンスの場合は CentOS 7.8 または AWS 7.9)

前述のコマンドは、Amazon FSx ファイルシステムをマウントして操作するために必要な 2 つのパッケージをインストールします。リポジトリには、ソースコードを含むパッケージやテストを含むパッケージなど、追加の Lustre パッケージが含まれており、必要に応じてインストールできます。リポジトリで使用できるすべてのパッケージを一覧表示するには、次のコマンドを使用します。

```
yum --disablerepo="*" --enablerepo="aws-fsx" list available
```

上流の出典コードの tarball と、適用したパッチのセットを含む出典 rpm をダウンロードするには、次のコマンドを使用します。

```
sudo yumdownloader --source kmod-lustre-client
```

yum 更新を実行すると、使用可能な場合は新しいバージョンのモジュールがインストールされ、既存のバージョンが置き換えられます。現在インストールされているバージョンが更新時に削除されないようにするには、次のような行を追加して /etc/yum.conf ファイルを開きます。

```
installonlypkgs=kernel, kernel-big-mem, kernel-enterprise, kernel-smp,
              kernel-debug, kernel-unsupported, kernel-source, kernel-devel, kernel-
PAE,
              kernel-PAE-debug, kmod-lustre-client
```

このリストには、yum.conf man ページ、および kmod-lustre-client パッケージ で指定したデフォルトのインストール専用パッケージが含まれます。

デフォルトのページサイズ (4KB) の Ubuntu

デフォルトのページサイズ (4KB) で Ubuntu 18.04、20.04、22.04、または 24.04 にLustreクライアントをインストールするには

Lustre パッケージは Amazon FSx Ubuntu リポジトリから入手できます。リポジトリのコンテンツがダウンロード前またはダウンロード中に改ざんされていないことを検証するために、GNU Privacy Guard (GPG) 署名がリポジトリのメタデータに適用されます。正しい公開キーをシステムにインストールしないと、リポジトリのインストールは失敗します。

1. クライアントのターミナルを開きます。
2. Amazon FSx Ubuntu リポジトリを追加するには、次の手順に従います。

   a. クライアントインスタンスに Amazon FSx Ubuntu リポジトリをまだ登録していない場合は、必要なパブリックキーをダウンロードしてインストールします。以下のコマンドを使用します。

      ```
      wget -O - https://fsx-lustre-client-repo-public-keys.s3.amazonaws.com/fsx-
      ubuntu-public-key.asc | gpg --dearmor | sudo tee /usr/share/keyrings/fsx-
      ubuntu-public-key.gpg >/dev/null
      ```

   b. 次のコマンドを使用して、Amazon FSx パッケージリポジトリをローカルパッケージマネージャに追加します。

```
sudo bash -c 'echo "deb [signed-by=/usr/share/keyrings/fsx-ubuntu-public-
key.gpg] https://fsx-lustre-client-repo.s3.amazonaws.com/ubuntu $(lsb_release -
cs) main" > /etc/apt/sources.list.d/fsxlustreclientrepo.list && apt-get update'
```

3. クライアントインスタンスで現在実行中のカーネルを特定し、必要に応じて更新します。x86 ベースの EC2 インスタンスと AWS Graviton プロセッサを搭載した Arm ベースの EC2 インスタンスの両方で、Ubuntu の Lustre クライアントに必要なカーネルのリストに関しては、「Ubuntu クライアント」を参照してください。

   a. カーネルが実行中であるかどうかを判断するために次のコマンドを実行します。

   ```
   uname -r
   ```

   b. 次のコマンドを実行して、最新の Ubuntu カーネルと Lustre バージョンに更新し、再起動します。

   ```
   sudo apt install -y linux-aws lustre-client-modules-aws && sudo reboot
   ```

   お使いのカーネルバージョンが、x86 ベースの EC2 インスタンスと Graviton ベースの EC2 インスタンスの両方で必要とされる最小カーネルバージョンよりも新しく、かつ最新のカーネルバージョンに更新したくない場合は、以下のコマンドで現在のカーネル用のLustreをインストールできます。

   ```
   sudo apt install -y lustre-client-modules-$(uname -r)
   ```

   Amazon FSx for Lustre ファイルシステムのマウントと操作に必要な 2 つの Lustre パッケージがインストールされます。出典コードを含むパッケージや、リポジトリ内のテストを含むパッケージなど、追加の関連したパッケージを必要に応じてインストールできます。

   c. リポジトリで使用できるすべてのパッケージを一覧表示するには、次のコマンドを使用します。

   ```
   sudo apt-cache search ^lustre
   ```

   d. (オプション) システムアップグレードで Lustre クライアントモジュールも常にアップグレードする場合は、`lustre-client-modules-aws` パッケージは、次のコマンドを使用してインストールされます。

```
sudo apt install -y lustre-client-modules-aws
```

> **Note**
>
> Module Not Found エラーが表示される場合は、「モジュールが見つからないというエラーのトラブルシューティングを行うには」を参照してください。

モジュールが見つからないというエラーのトラブルシューティングを行うには

任意のバージョンの Ubuntu のインストール中に Module Not Found エラーが表示された場合、以下の操作を実行します:

カーネルを最新のサポートされているバージョンにダウングレードします。lustre-client-modules パッケージのすべての利用可能なバージョンをリストし、対応するカーネルをインストールします。これを行うには、次のコマンドを使用します。

```
sudo apt-cache search lustre-client-modules
```

例えば、リポジトリに含まれる最新バージョンが lustre-client-modules-5.4.0-1011-aws の場合、次の作業を行います:

1. 次のコマンドを使用してこのパッケージを構築したカーネルをインストールします。

   ```
   sudo apt-get install -y linux-image-5.4.0-1011-aws
   ```

   ```
   sudo sed -i 's/GRUB_DEFAULT=.\+/GRUB\_DEFAULT="Advanced options for Ubuntu>Ubuntu,
    with Linux 5.4.0-1011-aws"/' /etc/default/grub
   ```

   ```
   sudo update-grub
   ```

2. 次のコマンドを実行して、インスタンスを再起動します。

   ```
   sudo reboot
   ```

3. 次のコマンドを使用して、Lustre クライアントをインストールします。

```
sudo apt-get install -y lustre-client-modules-$(uname -r)
```

ページサイズが 64KB の Ubuntu

64 KB のページサイズで Ubuntu24.04 (ARM64) にLustreクライアントをインストールするには 64KB

Lustre パッケージは Amazon FSx Ubuntu リポジトリから入手できます。リポジトリのコンテンツがダウンロード前またはダウンロード中に改ざんされていないことを検証するために、GNU Privacy Guard (GPG) 署名がリポジトリのメタデータに適用されます。正しい公開キーをシステムにインストールしないと、リポジトリのインストールは失敗します。

1. クライアントのターミナルを開きます。
2. インスタンスが 64KB のページサイズを使用していることを確認します。出力は である必要があります65536。

   ```
   getconf PAGESIZE
   ```

3. Amazon FSx Ubuntu リポジトリを追加するには、次の手順に従います。

   a. クライアントインスタンスに Amazon FSx Ubuntu リポジトリをまだ登録していない場合は、必要なパブリックキーをダウンロードしてインストールします。以下のコマンドを使用します。

      ```
      wget -O - https://fsx-lustre-client-repo-public-keys.s3.amazonaws.com/fsx-ubuntu-public-key.asc | gpg --dearmor | sudo tee /usr/share/keyrings/fsx-ubuntu-public-key.gpg >/dev/null
      ```

   b. 次のコマンドを使用して、Amazon FSx パッケージリポジトリをローカルパッケージマネージャに追加します。

      ```
      sudo bash -c 'echo "deb [signed-by=/usr/share/keyrings/fsx-ubuntu-public-key.gpg] https://fsx-lustre-client-repo.s3.amazonaws.com/ubuntu $(lsb_release -cs) main" > /etc/apt/sources.list.d/fsxlustreclientrepo.list && apt-get update'
      ```

4. クライアントインスタンスで現在実行中のカーネルを特定し、必要に応じて更新します。Ubuntu 24 カーネルのバージョンは 6.14.0-1018-aws-64k 以降である必要があります。

a. カーネルが実行中であるかどうかを判断するために次のコマンドを実行します。

```
uname -r
```

b. 次のコマンドを実行して、最新の Ubuntu カーネルと Lustre バージョンに更新し、再起動します。

```
sudo apt install -y linux-aws-64k lustre-client-modules-aws-64k && sudo reboot
```

お使いのカーネルバージョンが Graviton ベースの EC2 インスタンス`6.14.0-1018-aws-64k`で より大きく、最新のカーネルバージョンに更新しない場合は、次のコマンドを使用して現在のカーネルLustreに をインストールできます。

```
sudo apt install -y lustre-client-modules-$(uname -r)
```

Amazon FSx for Lustre ファイルシステムのマウントと操作に必要な 2 つの Lustre パッケージがインストールされます。出典コードを含むパッケージや、リポジトリ内のテストを含むパッケージなど、追加の関連したパッケージを必要に応じてインストールできます。

c. リポジトリで使用できるすべてのパッケージを一覧表示するには、次のコマンドを使用します。

```
sudo apt-cache search ^lustre
```

d. (オプション) システムアップグレードで Lustre クライアントモジュールも常にアップグレードする場合は、`lustre-client-modules-aws-64k` パッケージは、次のコマンドを使用してインストールされます。

```
sudo apt install -y lustre-client-modules-aws-64k
```

SUSE Linux

SUSE Linux 12 SP3、SP4、または SP5 に Lustre クライアントをインストールするには

SUSE Linux 12 SP3 に Lustre クライアントをインストールするには

1. クライアントのターミナルを開きます。
2. 次のコマンドを使用して、Amazon FSx をインストールします。

```
sudo wget https://fsx-lustre-client-repo-public-keys.s3.amazonaws.com/fsx-sles-public-key.asc
```

3. 次のコマンドを使用して、キーをインポートします。

```
sudo rpm --import fsx-sles-public-key.asc
```

4. 次のコマンドを使用して Lustre クライアントレポジトリを追加します。

```
sudo wget https://fsx-lustre-client-repo.s3.amazonaws.com/suse/sles-12/SLES-12/fsx-lustre-client.repo
```

5. Lustre クライアントをダウンロードしてインストールするには、以下のコマンドを使用します。

```
sudo zypper ar --gpgcheck-strict fsx-lustre-client.repo
sudo sed -i 's#SLES-12#SP3#' /etc/zypp/repos.d/aws-fsx.repo
sudo zypper refresh
sudo zypper in lustre-client
```

SUSE Linux 12 SP4 に Lustre クライアントをインストールするには

1. クライアントのターミナルを開きます。
2. 次のコマンドを使用して、Amazon FSx をインストールします。

```
sudo wget https://fsx-lustre-client-repo-public-keys.s3.amazonaws.com/fsx-sles-public-key.asc
```

3. 次のコマンドを使用して、キーをインポートします。

```
sudo rpm --import fsx-sles-public-key.asc
```

4. 次のコマンドを使用して Lustre クライアントレポジトリを追加します。

```
sudo wget https://fsx-lustre-client-repo.s3.amazonaws.com/suse/sles-12/SLES-12/fsx-lustre-client.repo
```

5. 次のいずれかを行います。

- SP4 を直接インストールするには、以下のコマンドを使用して Lustre クライアントをダウンロードし、インストールします。

```
sudo zypper ar --gpgcheck-strict fsx-lustre-client.repo
sudo sed -i 's#SLES-12#SP4#' /etc/zypp/repos.d/aws-fsx.repo
sudo zypper refresh
sudo zypper in lustre-client
```

- SP3 から SP4 に移行し、以前に SP3 用の Amazon FSx リポジトリを追加した場合は、次のコマンドを使用して Lustre クライアントをダウンロードしてインストールします。

```
sudo zypper ar --gpgcheck-strict fsx-lustre-client.repo
sudo sed -i 's#SP3#SP4#' /etc/zypp/repos.d/aws-fsx.repo
sudo zypper ref
sudo zypper up --force-resolution lustre-client-kmp-default
```

SUSE Linux 12 SP5 に Lustre クライアントをインストールするには

1. クライアントのターミナルを開きます。
2. 次のコマンドを使用して、Amazon FSx をインストールします。

```
sudo wget https://fsx-lustre-client-repo-public-keys.s3.amazonaws.com/fsx-sles-public-key.asc
```

3. 次のコマンドを使用して、キーをインポートします。

```
sudo rpm --import fsx-sles-public-key.asc
```

4. 次のコマンドを使用して Lustre クライアントレポジトリを追加します。

```
sudo wget https://fsx-lustre-client-repo.s3.amazonaws.com/suse/sles-12/SLES-12/fsx-lustre-client.repo
```

5. 次のいずれかを行います。

- SP5 を直接インストールするには、以下のコマンドを使用して Lustre クライアントをダウンロードし、インストールします。

```
sudo zypper ar --gpgcheck-strict fsx-lustre-client.repo
```

```
sudo zypper refresh
sudo zypper in lustre-client
```

- SP4 から SP5 に移行し、以前に SP4 用の Amazon FSx リポジトリを追加した場合は、次のコマンドを使用して Lustre クライアントをダウンロードしてインストールします。

```
sudo sed -i 's#SP4#SLES-12' /etc/zypp/repos.d/aws-fsx.repo
sudo zypper ref
sudo zypper up --force-resolution lustre-client-kmp-default
```

> **Note**
>
> インストールを完了するには、コンピューティングインスタンスを再起動する必要がある場合があります。

Amazon Elastic Compute Cloud インスタンスのマウント

Amazon EC2 インスタンスからファイルシステムをマウントできます。

Amazon EC2 からファイルシステムをマウントするには

1. Amazon EC2 インスタンスに接続します。
2. 次のコマンドを使用して、FSx for Lustre ファイルシステムでマウントポイントのディレクトリを作成します。

```
$ sudo mkdir -p /fsx
```

3. 作成したディレクトリに Amazon FSx for Lustre ファイルシステムをマウントします。次のコマンドを使用して、次のアイテムを置き換えます。

   - 実際のファイルシステムのシステムの DNS 名で *file_system_dns_name* を置き換えます。
   - ファイルシステムのマウント名で *mountname* を置き換えます。このマウント名は、`CreateFileSystem` API オペレーションレスポンスに返します。コマンドのレスポンスdescribe-file-systems AWS CLI 、および DescribeFileSystems API オペレーションでも返されます。

```
sudo mount -t lustre -o relatime,flock file_system_dns_name@tcp:/mountname /fsx
```

このコマンドは、-o relatime と flock の 2 つのオプションでファイルシステムをマウントします。

- relatime — atime オプションでは、ファイルがアクセスされるたびに atime (inode アクセス時間) のデータが保持されるのに対し、relatime オプションでも atime のデータが保持されますが、ファイルがアクセスされるたびに保持されるわけではありません。relatime オプションを有効にすると、atime のデータが最後に更新されてからファイルが変更された場合 (mtime)、またはファイルが一定時間以上 (デフォルトでは 6 時間) 前に最後にアクセスされた場合にのみ、atime のデータがディスクに書き込まれます。relatime または atime のオプションを使用すると、ファイルのリリース プロセスが最適化されます。

  > **Note**
  >
  > ワークロードに正確なアクセス時間の精度が必要な場合は、atime マウントオプションを使用してマウントできます。ただし、これを行うと、正確なアクセス時間値を維持するために必要なネットワークトラフィックが増加し、ワークロードのパフォーマンスに影響する可能性があります。
  > ワークロードにメタデータのアクセス時間が必要ない場合は、noatime マウントオプションを使用してアクセス時間の更新を無効にすると、パフォーマンスが向上する可能性があります。ファイルのリリースやデータの有効性のリリースなど、atime に焦点を絞ったプロセスでは、リリース時に不正確さが生じることに注意してください。

- flock - ファイルシステムのファイルロックを有効にします。ファイルロックを有効にしない場合は、mount なしの flock コマンドを使用します。

4. 次のコマンドを使用し、ファイルシステム、/mnt/fsx をマウントしたディレクトリの内容を一覧表示して、mount コマンドが正常に実行されたことを確認します。

```
$ ls /fsx
import-path  lustre
$
```

以下の df コマンドを使用することもできます。

```
$ df
Filesystem                     1K-blocks    Used  Available Use% Mounted on
devtmpfs                         1001808       0    1001808   0% /dev
tmpfs                            1019760       0    1019760   0% /dev/shm
tmpfs                            1019760     392    1019368   1% /run
tmpfs                            1019760       0    1019760   0% /sys/fs/cgroup
/dev/xvda1                       8376300 1263180    7113120  16% /
123.456.789.0@tcp:/mountname 3547698816   13824 3547678848   1% /fsx
tmpfs                             203956       0     203956   0% /run/user/1000
```

結果は、/fsx にマウントされている Amazon FSx ファイルシステムを示しています。

EFA クライアントの設定

以下の手順を使用して、Elastic Fabric Adapter (EFA) 経由で FSx for Lustre ファイルシステムにアクセスするように Lustre クライアントを設定します。

EFA は、以下のオペレーティングシステムを実行する Lustre クライアントでサポートされています:

- Amazon Linux 2023 (AL2023)
- Red Hat Enterprise Linux (RHEL) 9.5 以降
- カーネル 6.8 の Ubuntu 22.04 以降

EFA は、以下に示す Lustre クライアントでサポートされています。詳細については、「Lustre クライアントのインストール」を参照してください。

EFA は、trn2 インスタンスファミリーを除き、EFA をサポートする Nitro v4 (またはそれ以上) の EC2 インスタンスでサポートされています。「Amazon EC2 ユーザーガイド」の「サポートされているインスタンスタイプ」を参照してください。

トピック

- ステップ 1: 必要なドライバーをインストールする
- ステップ 2: Lustre クライアントの EFA を設定する
- ステップ 3: EFA インターフェイス

ステップ 1: 必要なドライバーをインストールする

> **Note**
>
> Deep Learning AMI を使用している場合は、Lustre クライアント、EFA ドライバー、NVIDIA GPUDirect Storage (GDS) ドライバーがプリインストールされているため、このステップをスキップできます。

Lustre クライアントと EFA ドライバーをインストールする

Lustre クライアントと EFA ドライバーをすばやくインストールするには

1. インストールスクリプトを含む ファイルをダウンロードして解凍します。

   ```
   curl -O https://docs.aws.amazon.com/fsx/latest/LustreGuide/samples/install-fsx-lustre-client.zip
   unzip install-fsx-lustre-client.zip
   ```

2. `install-fsx-lustre-client` フォルダに変更し、インストールスクリプトを実行します。

   ```
   cd install-fsx-lustre-client
   sudo ./bin/install-fsx-lustre-client.sh --install-lustre --install-efa
   ```

   スクリプトは以下を自動的に実行します:

   - Lustre クライアントをインストールします
   - EFA ドライバーをインストールします
   - Lustre クライアントと EFA ドライバーのインストールを検証します

   `install-fsx-lustre-client.sh` スクリプトで使用できるオプションと使用例のリストについては、zip ファイルの `README.md` ファイルを参照してください。

GDS ドライバーをインストールする (オプション)

このステップは、FSx for Lustre で NVIDIA GPUDirect Storage (GDS) を使用する場合にのみ必要です。

要件:

- Amazon EC2 P5, P5e, P5en、または P6-B200 インスタンス
- NVIDIA GDS ドライバーバージョン 2.24.2 以上

クライアントインスタンスに NVIDIA GPUDirect Storage ドライバーをインストールするには

1. NVIDIA GDS リポジトリのクローンを作成します:

```
git clone https://github.com/NVIDIA/gds-nvidia-fs.git
```

2. ドライバーをビルドしてインストールします:

```
cd gds-nvidia-fs/src/
export NVFS_MAX_PEER_DEVS=128
export NVFS_MAX_PCI_DEPTH=16
sudo -E make
sudo insmod nvidia-fs.ko
```

ステップ 2: Lustre クライアントの EFA を設定する

EFA インターフェイスを使用して FSx for Lustre ファイルシステムにアクセスするには、Lustre EFA モジュールをインストールし、EFA インターフェイスを設定する必要があります。

Quick Setup

Lustre クライアントをすばやく設定するには

1. Amazon EC2 インスタンスに接続します。
2. 設定スクリプトを含むファイルをダウンロードして解凍します:

```
curl -O https://docs.aws.amazon.com/fsx/latest/LustreGuide/samples/configure-efa-fsx-lustre-client.zip
unzip configure-efa-fsx-lustre-client.zip
```

3. `configure-efa-fsx-lustre-client` フォルダに変更し、セットアップスクリプトを実行します:

```
cd configure-efa-fsx-lustre-client
```

```
# for regular IO
sudo ./setup.sh

# for NVIDIA GPUDirect Storage (GDS) IO
sudo ./setup.sh --optimized-for-gds
```

スクリプトは以下を自動的に実行します:

- Lustre モジュールをインポートする
- TCP および EFA インターフェイスを設定する
- 再起動時に自動設定用の systemd サービスを作成する

`setup.sh` スクリプトで使用できるオプションと使用例のリストについては、zip ファイルの `README.md` ファイルを参照してください。

systemd サービスの手動管理

systemd サービスファイルは、/etc/systemd/system/configure-efa-fsx-lustre-client.service に作成されます。以下は、systemd 関連の便利なコマンドです:

```
# Check status
sudo systemctl status configure-efa-fsx-lustre-client.service

# View logs
sudo journalctl -u configure-efa-fsx-lustre-client.service
# View warnings/errors from dmesg
sudo dmesg
```

詳細については、zip ファイルの `README.md` ファイルを参照ください。

自動マウント設定 (オプション)

起動時の Amazon FSx for Lustre ファイルシステムの自動マウントの詳細については、「Amazon FSx ファイルシステムの自動マウント」を参照してください。

ステップ 3: EFA インターフェイス

各 FSx for Lustre ファイルシステムは、すべてのクライアントインスタンスを合わせた EFA 接続数の最大限界が 1024 です。

`configure-efa-fsx-lustre-client.sh` スクリプトは、インスタンスタイプに基づいて EFA インターフェイスを自動的に設定します。

インスタンスタイプ	EFA インターフェイスのデフォルト数
p6e-gb200.36xlarge	8
p6-b200.48xlarge	8
p5en.48xlarge	8
p5e.48xlarge	8
p5.48xlarge	8
複数のネットワークカードを持つ他のインスタンス	2
1 つのネットワークカードを持つ他のインスタンス	1

クライアントインスタンスで設定された各 EFA インターフェイスは、FSx for Lustre ファイルシステムに接続されている場合、1024 EFA 接続制限に対して 1 つの接続としてカウントされます。

EFA インターフェイスの手動管理

通常、より多くの EFA インターフェイスを持つインスタンスは、より高いスループットをサポートします。EFA 接続の合計制限内であれば、インターフェイスの数をカスタマイズして、特定のワークロードのパフォーマンスを最適化できます。

以下のコマンドを使用して、EFA インターフェイスを手動で管理できます。

1. 使用可能な EFA インターフェイスを表示します。

```
for interface in /sys/class/infiniband/*; do
    if [ ! -e "$interface/device/driver" ]; then continue; fi
    driver=$(basename "$(realpath "$interface/device/driver")")
    if [ "$driver" != "efa" ]; then continue; fi
    echo $(basename $interface)
done
```

2. 現在設定されているインターフェイスを表示します:

```
sudo lnetctl net show
```

3. EFA インターフェイスを追加します:

```
sudo lnetctl net add --net efa --if device_name --peer-credits 32
```

device_name をステップ 1 のリストから実際のデバイス名に置き換えます。

4. EFA インターフェイスを削除します:

```
sudo lnetctl net del --net efa --if device_name
```

device_name をステップ 2 のリストから実際のデバイス名に置き換えます。

Amazon Elastic Container Service からのマウント

FSx for Lustre ファイルシステムには、Amazon EC2 インスタンス上の Amazon Elastic Container Service (Amazon ECS) Docker コンテナからアクセスできます。これを行うには、次のオプションのいずれかを使用します。

1. Amazon ECS タスクをホストしている Amazon EC2 インスタンスから FSx for Lustre ファイルシステムをマウントし、このマウントポイントをコンテナにエクスポートします。
2. ファイルシステムをタスクコンテナ内に直接マウントする。

Amazon ECS の詳細については、「Amazon Elastic Container Service デベロッパーガイド」の「Amazon Elastic Container Service とは」を参照してください。

特に同じ EC2 インスタンスで多数のコンテナ (5 つ以上) を起動する場合や、タスクの存続期間が短い (5 分未満) の場合、リソースの使用率を向上させるためには、オプション 1 (Amazon ECS タスクをホストする Amazon EC2 インスタンスからマウントする) を使用することをお勧めします。

EC2 インスタンスを設定できない場合、またはアプリケーションがコンテナの柔軟性を必要とする場合、オプション 2 (Docker コンテナからのマウント) を使用します。

> **Note**
>
> AWS Fargate 起動タイプへの FSx for Lustre のマウントはサポートされていません。

以下のセクションでは、Amazon ECS コンテナから FSx for Lustre ファイルシステムをマウントする各オプションの手順について説明します。

トピック

- Amazon ECS タスクをホストする Amazon EC2 インスタンスからマウントする
- Docker コンテナからのマウント

Amazon ECS タスクをホストする Amazon EC2 インスタンスからマウントする

この手順では、FSx for Lustre ファイルシステムをローカルにマウントするように EC2 インスタンス上の Amazon ECS を設定する方法を示します。この手順では `volumes` および `mountPoints` コンテナプロパティを使用して、リソースを共有し、ローカルで実行されているタスクがこのファイルシステムにアクセスできるようにします。詳細については、「Amazon Elastic Container Service デベロッパーガイド」の「Amazon ECS コンテナインスタンスの起動」を参照してください。

この手順は、Amazon ECS 最適化 Amazon Linux 2 AMI 用に書かれています。別の Linux ディストリビューションを使用している場合は、「Lustre クライアントのインストール」を参照してください。

EC2 インスタンスの Amazon ECS からファイルシステムをマウントするには

1. Amazon ECS インスタンスを手動で、または Auto Scaling グループを使用して起動する場合は、次のコード例の行を [User data] (ユーザーデータ) フィールドの最後に追加します。例の項目を以下に置き換えます。

   - 実際のファイルシステムのシステムの DNS 名で *`file_system_dns_name`* を置き換えます。
   - ファイルシステムのマウント名で *`mountname`* を置き換えます。
   - 作成する必要があるファイルシステムのマウントポイントを使用して、*`mountpoint`* を置き換えます。

```
#!/bin/bash

...<existing user data>...

fsx_dnsname=file_system_dns_name
fsx_mountname=mountname
fsx_mountpoint=mountpoint
amazon-linux-extras install -y lustre
mkdir -p "$fsx_mountpoint"
mount -t lustre ${fsx_dnsname}@tcp:/${fsx_mountname} ${fsx_mountpoint} -o
 relatime,flock
```

2. Amazon ECS タスクを作成するときは、以下の JSON 定義の `volumes` および `mountPoints` コンテナプロパティを追加します。ファイルシステムのマウントポイント (`/mnt/fsx` など) で *`mountpoint`* を置き換えます。

```
{
    "volumes": [
           {
                 "host": {
                      "sourcePath": "mountpoint"
                 },
                 "name": "Lustre"
           }
    ],
    "mountPoints": [
           {
                 "containerPath": "mountpoint",
                 "sourceVolume": "Lustre"
           }
    ],
}
```

Docker コンテナからのマウント

次の手順で、Amazon ECS タスクコンテナを設定して `lustre-client` パッケージをインストールし、FSx for Lustre ファイルシステムをマウントします。この手順では、Amazon Linux (`amazonlinux`) Docker イメージを使用しますが、他のディストリビューションでも同様のアプローチが機能します。

ファイルシステムを Docker コンテナからマウントするには

1. Docker コンテナで、`lustre-client` パッケージをインストールし、FSx for Lustre ファイルシステムを `command` プロパティでマウントします。例の項目を以下に置き換えます。

   - 実際のファイルシステムのシステムの DNS 名で *`file_system_dns_name`* を置き換えます。
   - ファイルシステムのマウント名で *`mountname`* を置き換えます。
   - ファイルシステムのマウントポイントで *`mountpoint`* を置き換えます。

   ```
   "command": [
     "/bin/sh -c \"amazon-linux-extras install -y lustre; mount -t
    lustre file_system_dns_name@tcp:/mountname mountpoint -o relatime,flock;\""
   ],
   ```

2. `linuxParameters` プロパティを使用して、FSx for Lustre ファイルシステムをマウントすることをコンテナに許可する `SYS_ADMIN` 機能を追加します。

   ```
   "linuxParameters": {
     "capabilities": {
         "add": [
           "SYS_ADMIN"
         ]
      }
   }
   ```

オンプレミスまたはピアリングされた Amazon VPC から Amazon FSx ファイルシステムをマウントする

Amazon FSx ファイルシステムには、2 つの方法でアクセスできます。1 つは、ファイルシステムの VPC にピアリングされる Amazon VPC にある Amazon EC2 インスタンスからのものです。もう 1 つは、 Direct Connect または VPN を使用してファイルシステムの VPC に接続されているオンプレミスクライアントからのものです。

クライアントの VPC と Amazon FSx ファイルシステムの VPC を接続するには、VPC ピアリング接続または VPC トランジットゲートウェイを使用します。VPC ピアリング接続またはトランジットゲートウェイを使用して VPC を接続すると、VPC が別のアカウントに属している場合でも、ある

VPC にある Amazon EC2 インスタンスが別の VPC にある Amazon FSx ファイルシステムにアクセスできます。

次の手順を使用する前に、VPC ピアリング接続または VPC トランジットゲートウェイを設定する必要があります。

トランジットゲートウェイは、VPC とオンプレミスネットワークを相互接続するために使用できるネットワークの中継ハブです。VPC Transit Gateway の使用の詳細については、「Amazon VPC Transit Gateway ガイド」の「Transit Gateway の開始方法」を参照してください。

VPC ピアリング接続は、2 つの VPC 間のネットワーク接続です。このタイプの接続では、インターネットプロトコルバージョン 4 (IPv4) またはインターネットプロトコルバージョン 6 (IPv6) のプライベートアドレスを使用して、2 つの VPC 間でトラフィックをルーティングできます。VPC ピアリングを使用して、同じ AWS リージョン内または AWS リージョン間で VPCs を接続できます。VPC ピアリングについての詳細については、「Amazon VPC ピアリング ガイド」の「VPC ピア機能とは」を参照してください。

プライマリネットワークインターフェイスの IP アドレスを使用して、ファイルシステムをその VPC 外部からマウントできます。プライマリネットワークインターフェイスは、`aws fsx describe-file-systems` AWS CLI コマンドの実行時に返される最初のネットワークインターフェイスです。また、Amazon Web Services マネジメントコンソールからこの IP アドレスを取得することもできます。

次の表に、ファイルシステムの VPC 外にあるクライアントを使用して Amazon FSx ファイルシステムにアクセスするための IP アドレス要件を示します。

以下に所在するクライアントの場合	2020 年 12 月 17 日より前に作成されたファイルシステムへのアクセス	2020 年 12 月 17 日以降に作成されたファイルシステムへのアクセス
VPC ピアリングまたは AWS Transit Gatewayを使用した VPC のピアリング	RFC 1918 プライベート IP アドレス範囲に IP アドレスを持つクライアント: • 10.0.0.0/8 • 172.16.0.0/12 • 192.168.0.0/16	✓
Direct Connect または を使用したピアネットワーク Site-to-Site VPN		✓

2020 年 12 月 17 日より前に作成された Amazon FSx ファイルシステムに、非プライベート IP アドレス範囲を使用してアクセスする必要がある場合は、ファイルシステムのバックアップを復元して、新しいファイルシステムを作成できます。詳細については、「バックアップでデータを保護する。」を参照してください。

ファイルシステムのプライマリネットワークインターフェイスの IP アドレスを取得するには

1. https://console.aws.amazon.com/fsx/ で Amazon FSx コンソールを開きます。
2. ナビゲーションペインで、[File systems] (ファイルシステム) を選択します。
3. ダッシュボードから、ファイルシステムを選択します。
4. ファイルシステム詳細ページで、[Network & security] (ネットワークとセキュリティ) を選択します。
5. [Network interface] (ネットワークインターフェイス) で、プライマリ elastic network interface の ID を選択します。これにより、Amazon EC2 コンソールに移動します。
6. [Details] (詳細) タブで、[Primary private IPv4 IP] (プライマリプライベート IPv4 IP) を参照します。これは、プライマリネットワークインターフェイスの IP アドレスです。

> **Note**
>
> 関連付けられている VPC 外部から Amazon FSx ファイルシステムをマウントするときは、ドメインネームシステム (DNS) 名前解決を使用できません。

Amazon FSx ファイルシステムの自動マウント

Amazon EC2 インスタンスの `/etc/fstab` ファイルを初めてインスタンスに接続した後に、再起動のたびに Amazon FSx ファイルシステムをマウントします。

/etc/fstab を使用して FSx for Lustre を自動マウントする

Amazon EC2 インスタンスの再起動時に Amazon FSx ファイルシステムディレクトリを自動的に再マウントするには、`fstab` ファイルを使用できます。`fstab` ファイルには、ファイルシステムに関する情報が含まれています。インスタンスの起動中に実行される `mount -a` コマンドは、`fstab` ファイルに示されているファイルシステムをマウントします。

> **Note**
>
> - EC2 インスタンスの /etc/fstab ファイルを更新する前に、Amazon FSx ファイルシステムがすでに作成済みであることを確認してください。詳細については、「入門編エクササイズ」の「ステップ 1: FSx for Lustre ファイルシステムの作成」を参照してください。
> - EFA 対応ファイルシステムでは、systemd の設定が前提条件です。詳細については、「Quick Setup」を参照してください。

EC2 インスタンスの /etc/fstab ファイルを更新するには

1. EC2 インスタンスに接続して、エディタで /etc/fstab ファイルを開きます。
2. 次の行を /etc/fstab ファイルに追加します。

   作成したディレクトリに Amazon FSx for Lustre ファイルシステムをマウントします。次のコマンドを使用して、以下を置き換えます。

   - */fsx* を置き換えるには、Amazon FSx ファイルシステムをマウントするディレクトリを使用します。
   - 実際のファイルシステムのシステムの DNS 名で *file_system_dns_name* を置き換えます。
   - ファイルシステムのマウント名で *mountname* を置き換えます。このマウント名は、CreateFileSystem API オペレーションレスポンスに返します。また、 describe-file-systems AWS CLI コマンドのレスポンス、および DescribeFileSystems API オペレーションでも返されます。

   EFA 非対応のファイルシステムの場合:

   ```
   file_system_dns_name@tcp:/mountname /fsx lustre defaults,relatime,flock,_netdev,x-systemd.automount,x-systemd.requires=network.service 0 0
   ```

   EFA 対応ファイルシステムの場合:

   ```
   file_system_dns_name@tcp:/mountname /fsx lustre defaults,relatime,flock,_netdev,x-systemd.automount,x-systemd.requires=configure-efa-fsx-lustre-client.service,x-systemd.after=configure-efa-fsx-lustre-client.service 0 0
   ```

> **Warning**
>
> ファイルシステムを自動的にマウントする場合、ネットワークファイルシステムを識別するために使用された _netdev オプションを使用します。_netdev が見つからない場合、EC2 インスタンスはレスポンスを停止する可能性があります。この結果は、コンピューティングインスタンスがネットワークを開始後、ネットワークファイルシステムを初期化する必要があるためです。詳細については、「自動マウントが失敗してインスタンスがレスポンスしない」を参照してください。

3. 変更をファイルに保存します。

EC2 インスタンスは、再起動するたびに Amazon FSx ファイルシステムをマウントするように設定されました。

> **Note**
>
> 場合によっては、マウントされた Amazon FSx ファイルシステムのステータスに関係なく、Amazon EC2 インスタンスの起動が必要になることがあります。これらの場合は、/etc/fstab ファイルに記載されているファイルシステムのエントリに nofail オプションを追加します。

/etc/fstab ファイルに追加したコードの行のフィールドは以下のようになります。

フィールド	説明
file_system_dns_name @tcp:/	Amazon FSx ファイルシステムの DNS 名は、ファイルシステムを識別します。この名前は、 コンソールから取得することも、 AWS CLI または SDK からプログラムで取得することもできます AWS 。
mountname	ファイルシステムのマウント名。この名前は、 コンソールから取得することも、 describe-file-systems コマンド AWS CLI を使用して からプログラムで取得することも、 DescribeFileSystems オペレーションを使用して AWS API または SDK から取得することもできます。

フィールド	説明
/fsx	EC2 インスタンスの Amazon FSx ファイルシステムのマウントポイントです。
lustre	ファイルシステムのタイプは、Amazon FSx です。
mount options	ファイルシステムのマウントオプションは、次のオプションのカンマ区切りのリストとして表示されます。 • defaults - この値は、デフォルトのマウントオプションを使用するようにオペレーティングシステムに指示します。mount コマンドの出力を表示してファイルシステムがマウントされた後で、デフォルトのマウントオプションを一覧表示できます。 • relatime — このオプションは atime (inode アクセス時間) のデータを保持しますが、ファイルがアクセスされるたびに保持するわけではありません。このオプションを有効にすると、atime のデータが最後に更新されてからファイルが変更された場合 (mtime)、またはファイルが一定期間以上 (デフォルトでは 1 日) 前に最後にアクセスされた場合にのみ、atime のデータがディスクに書き込まれます。inode アクセス時間の更新を無効にする場合は、代わりに noatime マウントオプションを使用します。 • flock - ファイルロックを有効にしてファイルシステムをマウントします。ファイルロックを有効にしない場合は、noflock マウントオプションを使用します。 • _netdev - この値は、ファイルシステムがネットワークアクセスを必要とするデバイスに存在することをオペレーティングシステムに通知します。このオプションは、クライアント上でネットワークが有効になるまで、インスタンスがファイルシステムをマウントするのを防ぎます。

フィールド	説明
`x-systemd.automount,x-systemd.requires=network.service`	EFA 非対応のファイルシステムのためのこれらのオプションにより、ネットワーク接続がオンラインになるまで、自動マウンタが動作しないことが保証されます。 **Note** Amazon Linux 2023 および Ubuntu 22.04 以上の場合、`x-systemd.requires=network.service` オプション代わりに `x-systemd.requires=systemd-networkd-wait-online.service` を使用してください。
`x-systemd.automount,x-systemd.requires=configure-efa-fsx-lustre-client.service,x-systemd.after=configure-efa-fsx-lustre-client.service`	EFA 対応ファイルシステムのこれらのオプションにより、EFA クライアント設定が完了するまでオートマウンタが実行されなくなります。
`0`	ファイルシステムを dump でバックアップする必要があるかどうかを示す値。Amazon FSx の場合、この値は `0` です。
`0`	起動時に `fsck` がファイルシステムをチェックする順序を示す値。Amazon FSx ファイルシステムの場合、スタートアップ時に `fsck` を実行すべきでないことを示すにはこの値を `0` にします

特定のファイルセットのマウント

Lustre ファイルセット機能を使用すると、ファイルシステム名前空間のサブセットのみをマウントでき、これをファイルセットと呼びます。ファイルシステムのファイルセットをマウントするには、クライアントでファイルシステム名の後にサブディレクトリパスを指定します。ファイルセットマウ

ント (サブディレクトリマウントとも呼ばれる) は、特定のクライアントでのファイルシステムの名前空間の可視性を制限します。

例 - Lustre ファイルセットをマウントする

1. 次のディレクトリを持つ FSx for Lustre ファイルシステムがあるとします。

```
team1/dataset1/
team2/dataset2/
```

2. `team1/dataset1` ファイルセットだけをマウントし、ファイルシステムのこの部分のみをクライアント上でローカルに表示します。次のコマンドを使用して、次のアイテムを置き換えます。

   - 実際のファイルシステムのシステムの DNS 名で *`file_system_dns_name`* を置き換えます。
   - ファイルシステムのマウント名で *`mountname`* を置き換えます。このマウント名は、`CreateFileSystem` API オペレーションレスポンスに返します。また、 describe-file-systems AWS CLI コマンドのレスポンス、および DescribeFileSystems API オペレーションでも返されます。

```
mount -t lustre file_system_dns_name@tcp:/mountname/team1/dataset1 /fsx
```

Lustre ファイルセット機能を使用する際、以下に注意してください。

- クライアントが別のファイルセットを使用してファイルシステムを再マウントすること、またはファイルセットを全く使用しないことを妨げる制約はありません。
- ファイルセットを使用する場合、`.lustre/` ディレクトリへのアクセスを必要とする一部の Lustre 管理コマンドは (`lfs fid2path` コマンドなど) 動作しない場合があります。
- 同じホスト上の同じファイルシステムから複数のサブディレクトリをマウントする場合は、単一のマウントポイントよりも多くのリソースを消費し、代わりにファイルシステムのルートディレクトリを一度だけマウントする方が効率的であることに注意してください。

Lustre ファイルセット機能の詳細については、Lustre ドキュメントウェブサイトの「Lustre Operations Manual」を参照してください。

ファイルシステムをアンマウントする

FSx for Lustre ファイルシステムを削除する前に、FSx for Lustre ファイルシステムをマウントしたすべての Amazon EC2 インスタンスからアンマウントされていることを確認し、Amazon EC2 インスタンスをシャットダウンまたは終了する前に、マウントした FSx for Lustre ファイルシステムがそのインスタンスからアンマウントされていることを確認します。

FSx for Lustre サーバーは、I/O オペレーション中にクライアントに一時ファイルとディレクトリのロックを付与します。クライアントは、サーバーがクライアントにそのロックを解除して他のクライアントの I/O オペレーションのブロックを解除するよう要求したときに、すみやかに応答する必要があります。クライアントが応答しなくなった場合、他のクライアントがリクエストされた I/O オペレーションを続行できるように、数分後に強制的に切断される場合があります。これらの待機期間を回避するには、クライアントインスタンスをシャットダウンまたは終了する前に、および FSx for Lustre ファイルシステムを削除する前に、常にクライアントインスタンスからファイルシステムをアンマウントする必要があります。

インスタンス自体で umount コマンドを実行することで、Amazon EC2 インスタンスのファイルシステムをアンマウントできます。Amazon FSx ファイルシステムは、 AWS CLI、、 AWS マネジメントコンソールまたは AWS SDKs を使用してアンマウントすることはできません。Linux を実行する Amazon EC2 インスタンスに接続されている Amazon FSx ファイルシステムをアンマウントするには、次のように umount コマンドを使用します。

```
umount /mnt/fsx
```

他の umount オプションを指定しないことをお勧めします。デフォルトと異なる umount オプションを設定しないでください。

df コマンドを実行すると、Amazon FSx ファイルシステムのマウントが解除されたことを確認できます。このコマンドを実行すると、Linux ベースの Amazon EC2 インスタンスに現在マウントされているファイルシステムのディスク使用状況の統計情報が表示されます。アンマウントする Amazon FSx ファイルシステムが df コマンドの出力にリストされていない場合、ファイルシステムがアンマウントされていることを意味します。

Example- Amazon FSx ファイルシステムのマウントステータスを特定してアンマウントする

```
$ df -T
Filesystem Type 1K-blocks Used Available Use% Mounted on
file-system-id.fsx.aws-region.amazonaws.com@tcp:/mountname /fsx 3547708416 61440
 3547622400 1% /fsx
```

```
      /dev/sda1 ext4 8123812 1138920 6884644 15% /
```

```
$ umount /fsx
```

```
$ df -T
```

```
Filesystem Type 1K-blocks Used Available Use% Mounted on
/dev/sda1 ext4 8123812 1138920 6884644 15% /
```

Amazon EC2 スポットインスタンスの使用

FSx for Lustre を EC2 スポットインスタンスとともに使用すると、Amazon EC2 のコストを大幅に削減できます。スポットインスタンスは、オンデマンド料金より低価で利用できる未使用の EC2 インスタンスです。スポット料金が上限を超えた場合や、スポットインスタンスの需要が増加した場合、あるいはスポットインスタンスの供給が減少した場合には、Amazon EC2 がスポットインスタンスを中断する可能性があります。

Amazon EC2 によりスポットインスタンスが中断される際には、スポットインスタンスの中断通知が送信されます。それにより、Amazon EC2 が中断する 2 分前にインスタンスに対して警告を出します。詳細については、「Amazon EC2 ユーザーガイド」の「スポットインスタンス」を参照してください。

EC2 スポットインスタンスの中断によって Amazon FSx ファイルシステムが影響を受けないように、EC2 スポットインスタンスを終了または休止する前に Amazon FSx ファイルシステムをアンマウントすることをお勧めします。詳細については、「ファイルシステムをアンマウントする」を参照してください。

Amazon EC2 スポットインスタンスの中断

FSx for Lustre は、サーバーとクライアントインスタンスが協力してパフォーマンスと信頼性の高いファイルシステムを提供する分散ファイルシステムです。これらは、クライアントインスタンスとサーバーインスタンスの両方で配信されたコヒーレント状態を維持します。FSx for Lustre サーバは、I / O およびファイルシステムデータのキャッシュを積極的に実行している間、クライアントにテンポラリアクセス許可を委任します。クライアントは、サーバーがテンポラリアクセス許可の取り消しをリクエストすると、短期間でレスポンスすることが期待されます。クライアントの不正動作からファイルシステムを保護するために、サーバーは数分後にレスポンスしない Lustre クライアントを削除できます。レスポンスしないクライアントがサーバーリクエストにレスポンスするまで数分待

つ必要がないようにするには、特に EC2 スポットインスタンスを終了する前に、Lustre クライアントをきれいにアンマウントすることが重要です。

EC2 スポットは、インスタンスをシャットダウンする前に 2 分前に終了通知を送信します。EC2 スポットインスタンスを終了する前に、Lustre クライアントをクリーンにアンマウントするプロセスを自動化することをお勧めします。

Example- 終了する EC2 スポットインスタンスをクリーンにマウント解除するスクリプト

このサンプルスクリプトは、次の操作を実行して、終了する EC2 スポットインスタンスをクリーンにアンマウントします。

- スポット終了通知をモニタリングします。
- 終了通知が届くと、次のようになります。
  - ファイルシステムにアクセスしているアプリケーションを停止します。
  - インスタンスが終了する前にファイルシステムをアンマウントします。

必要に応じて、特にアプリケーションを正常にシャットダウンするために、スクリプトを適応させることができます。スポットインスタンスの中断を処理するためのベストプラクティスの詳細については、「EC2 スポットインスタンスの中断を処理するためのベストプラクティス」を参照してください。

```
#!/bin/bash

# TODO: Specify below the FSx mount point you are using
*FSXPATH=/fsx*

cd /

TOKEN=$(curl -s -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-
metadata-token-ttl-seconds: 21600")
if [ "$?" -ne 0 ]; then
    echo "Error running 'curl' command" >&2
    exit 1
fi

# Periodically check for termination
while sleep 5
do
```

```
    HTTP_CODE=$(curl -H "X-aws-ec2-metadata-token: $TOKEN" -s -w %{http_code} -o /dev/
null http://169.254.169.254/latest/meta-data/spot/instance-action)

    if [[ "$HTTP_CODE" -eq 401 ]] ; then
        # Refreshing Authentication Token
        TOKEN=$(curl -s -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-
metadata-token-ttl-seconds: 30")
        continue
    elif [[ "$HTTP_CODE" -ne 200 ]] ; then
        # If the return code is not 200, the instance is not going to be interrupted
        continue
    fi

    echo "Instance is getting terminated. Clean and unmount '$FSXPATH' ..."
    curl -H "X-aws-ec2-metadata-token: $TOKEN" -s http://169.254.169.254/latest/meta-
data/spot/instance-action
    echo

    # Gracefully stop applications accessing the filesystem
    #
    # TODO*: Replace with the proper command to stop your application if possible*

    # Kill every process still accessing Lustre filesystem
    echo "Kill every process still accessing Lustre filesystem..."
    fuser -kMm -TERM "${FSXPATH}"; sleep 2
    fuser -kMm -KILL "${FSXPATH}"; sleep 2

    # Unmount FSx For Lustre filesystem
    if ! umount -c "${FSXPATH}"; then
        echo "Error unmounting '$FSXPATH'. Processes accessing it:" >&2
        lsof "${FSXPATH}"

        echo "Retrying..."
        continue
    fi

    # Start a graceful shutdown of the host
    shutdown now

done
```

ファイルシステムの管理

FSx for Lustre は、管理タスクのパフォーマンスを簡素化する一連の機能を提供します。これには、ポイントインタイムバックアップの作成、ファイルシステムのストレージクォータの管理、ストレージとスループットキャパシティの管理、データ圧縮の管理、およびシステムの定期的なソフトウェアパッチ適用を実行するためのメンテナンス時間の設定が含まれます。

FSx for Lustre ファイルシステムは、Amazon FSx 管理コンソール、AWS Command Line Interface (AWS CLI)、Amazon FSx API、または AWS SDK を使用して管理できます。

トピック

EFA 対応ファイルシステムの使用

スループットキャパシティが 10 GBps を超えるファイルシステムを作成する場合は、Elastic Fabric Adapter (EFA) を有効にして、クライアントインスタンスあたりのスループットを最適化することをお勧めします。EFA は、カスタム構築のオペレーティングシステムバイパス手法と AWS スケーラブルな信頼性の高いデータグラム (SRD) ネットワークプロトコルを使用してパフォーマンスを向上させる高性能ネットワークインターフェイスです。EFA の詳細については、「Amazon EC2 ユー

ザーガイド」の「Amazon EC2 の AI/ML および HPC ワークロード用の Elastic Fabric Adapter」を参照してください。

EFA 対応ファイルシステムは、GPUDirect Storage (GDS) と ENA Express の 2 つの追加パフォーマンス機能をサポートしています。GDS サポートは EFA 上に構築されており、ファイルシステムと GPU メモリ間の直接データ転送を有効にして CPU をバイパスすることで、パフォーマンスをさらに強化します。この直接パスにより、冗長なメモリコピーやデータ転送オペレーションへの CPU の関与が不要になります。EFA と GDS のサポートにより、個々の EFA 対応クライアントインスタンスに対してより高いスループットを実現できます。ENA Express は、高度な経路選択アルゴリズムと強化された輻輳制御メカニズムを用いて、Amazon EC2 インスタンス向けに最適化されたネットワーク通信を提供します。ENA Express サポートを使用すると、個々の ENA Express 対応クライアントインスタンスに対してより高いスループットを実現できます。詳細については、「Amazon EC2 ユーザーガイド」の「ENA Express を使用して EC2 インスタンス間のネットワークパフォーマンスを高める」を参照してください。

トピック

- EFA 対応ファイルシステム使用時の考慮事項
- EFA 対応ファイルシステムを使用するための前提条件
- EFA 対応ファイルシステムの作成

EFA 対応ファイルシステム使用時の考慮事項

EFA 対応ファイルシステムを作成するときに考慮すべき重要な項目を以下に示します:

- 複数の接続オプション: EFA 対応ファイルシステムは、ENA、ENA Express、EFA を使用してクライアントインスタンスと通信できます。
- デプロイタイプ: EFA は、Intelligent-Tiering ストレージクラスを使用するファイルシステムなど、メタデータ設定が指定された永続 2 ファイルシステムでサポートされています。
- EFA 設定の更新: 新しいファイルシステムを作成するときに EFA を有効にすることはできますが、既存のファイルシステムで EFA を有効または無効にすることはできません。
- ストレージ容量によるスループットのスケーリング: EFA 対応の SSD ベースのファイル システムのストレージ容量を拡張してスループット容量を増やすことはできますが、EFA 対応のファイルシステムのスループット階層を変更することはできません。
- AWS リージョン: EFA 対応の永続 2 ファイルシステム AWS リージョン をサポートする のリストについては、「」を参照してくださいデプロイタイプの可用性。

EFA 対応ファイルシステムを使用するための前提条件

EFA 対応ファイルシステムを使用するための前提条件は以下のとおりです:

EFA 対応ファイルシステムを作成するには:

- EFA 対応のセキュリティグループを使用します。詳細については、「EFA 対応セキュリティグループ」を参照してください。
- Amazon VPC 内の EFA 対応クライアントインスタンスと同じアベイラビリティーゾーンと /16 CIDR を使用します。
- インテリジェント階層化ファイルシステムでは、EFA はスループットキャパシティが 4,000 MBps または 4,000 MBps の増分でのみサポートされています。

Elastic Fabric Adapter (EFA) を使用してファイルシステムにアクセスするには:

- EFA をサポートする Nitro v4 (またはそれ以上) EC2 インスタンスを使用します。ただし、trn2 インスタンスファミリーは除きます。「Amazon EC2 ユーザーガイド」の「サポートされているインスタンスタイプ」を参照してください。
- AL2023、RHEL 9.5 以降、またはカーネルバージョン 6.8 以降の Ubuntu 22 以降で実行します。詳細については、「Lustre クライアントのインストール」を参照してください。
- EFA モジュールをインストールし、クライアントインスタンスに EFA インターフェイスを設定します。詳細については、「EFA クライアントの設定」を参照してください。

GPUDirect Storage (GDS) を使用してファイルシステムにアクセスするには:

- Amazon EC2 P5, P5e, P5en、または P6-B200 クライアントインスタンスを使用します。
- NVIDIA Compute Unified Device Architecture (CUDA) パッケージ、オープンソースの NVIDIA ドライバー、NVIDIA GPUDirect Storage Driver をクライアントインスタンスにインストールします。詳細については、「GDS ドライバーをインストールする (オプション)」を参照してください。

ENA Express を使用してファイルシステムにアクセスするには:

- ENA Express をサポートする Amazon EC2 インスタンスを使用します。「Amazon EC2 ユーザーガイド」の「ENA Express でサポートされているインスタンスタイプ」を参照してください。
- Linux インスタンスの設定を更新します。詳細については、「Amazon EC2 ユーザーガイド」の「Linux インスタンスの前提条件」を参照してください。

- クライアントインスタンスのネットワークインターフェイスで ENA Express を有効にします。詳細については、「Amazon EC2 ユーザーガイド」の「EC2 インスタンスの ENA Express 設定を確認する」を参照してください。

EFA 対応ファイルシステムの作成

このセクションでは、 AWS CLIを使用して FSx for Lustre EFA 対応ファイルシステムを作成する方法について説明します。Amazon FSx コンソールを使用して EFA 対応ファイルシステムを作成する方法については、「ステップ 1: FSx for Lustre ファイルシステムの作成」を参照してください。

EFA 対応ファイルシステムを作成するには (CLI)

create-file-system CLI コマンド (または同等の CreateFileSystem API オペレーション) を使用します。以下の例では、PERSISTENT_2 デプロイタイプを使用した EFA 対応の FSx for Lustre ファイルシステムを作成しています。

```
aws fsx create-file-system\
   --storage-capacity 4800 \
   --storage-type SSD \
   --file-system-type LUSTRE \
   --file-system-type-version 2.15 \
   --subnet-ids subnet-01234567890 \
   --security-group-ids sg-0123456789abcdefg \
   --lustre-configuration '{"DeploymentType": "PERSISTENT_2", "EfaSupport": true}'
```

ファイルシステムが正常に作成されると、Amazon FSx はファイルシステムの説明を JSON 形式で返します。

Lustre ストレージクォータの使用

FSx for Lustre ファイルシステムでは、ユーザー、グループ、プロジェクトに対してストレージクォータを作成できます。ストレージクォータを設定すると、ユーザー、グループ、またはプロジェクトが消費できるディスク容量とファイル数を制限することができます。ストレージクォータは、ユーザーレベル、グループレベル、プロジェクトレベルの使用状況を自動的に追跡するため、ストレージ制限を設定するかどうかにかかわらず、消費量をモニタリングすることができます。

Amazon FSx はクォータを適用し、クォータを超えたユーザーがストレージスペースに書き込むのを防ぎます。ユーザーがクォータを超えた場合、クォータ制限を下回るまでファイルを削除して、ファイルシステムに再度書き込みができるようにする必要があります。

トピック

クォータの適用

ユーザー、グループ、プロジェクトに対するクォータの適用は、すべての FSx for Lustre ファイルシステムで自動的に有効になります。クォータの適用を無効にすることはできません。

クォータの種類

AWS アカウントルートユーザーの認証情報を持つシステム管理者は、次のタイプのクォータを作成できます。

- ユーザークォータ は、個々のユーザーに適用されます。特定のユーザーのユーザークォータを、他のユーザーのクォータとは異なるようにできます。
- グループクォータは、特定のグループのメンバーであるすべてのユーザーに適用されます。
- プロジェクトクォータは、プロジェクトに関連するすべてのファイルまたはディレクトリに適用されます。プロジェクトには、ファイルシステム内の異なるディレクトリにある複数のディレクトリや個々のファイルを含めることができます。

> **Note**
>
> プロジェクトクォータは、FSx for Lustre ファイルシステムの Lustre バージョン 2.15 でのみサポートされています。

- ブロッククォータは、ユーザー、グループ、プロジェクトが消費できるディスク容量を制限します。ストレージサイズはキロバイト単位で設定します。
- Inode クォータは、ユーザー、グループ、プロジェクトが作成できるファイルまたはディレクトリの数を制限します。Inode の最大数を整数として設定します。

> **Note**
>
> デフォルトクォータはサポートされていません。

特定のユーザーおよびグループにクォータを設定し、そのユーザーがそのグループのメンバーである場合、ユーザーのデータ使用量は両方のクォータに適用されます。また、両方のクォータによって制限されます。いずれかのクォータ制限に達すると、ユーザーはファイルシステムへの書き込みをブロックされます。

> **Note**
>
> Root ユーザーに設定されたクォータは強制されません。同様に、sudo コマンドを使用して root ユーザーとしてデータを書き込むと、クォータの適用がバイパスされます。

クォータ制限と猶予期間

Amazon FSx では、ユーザー、グループ、プロジェクトのクォータをハード制限として、または設定可能な猶予期間を持つソフト制限として適用します。

ハードリミットは絶対制限です。ユーザーがハードリミットを超えると、ブロックまたは i ノードの割り当ては ディスククォータの超過 メッセージを表示して拒否します。クォータのハード制限に達したユーザーは、クォータ制限を下回るようにファイルやディレクトリを削除してから、ファイルシステムに再度書き込みする必要があります。猶予期間が設定されている場合、ハードリミット未満であれば、ユーザーは猶予期間内にソフトリミットを超えることができます。

ソフトリミットでは、猶予期間を秒単位で設定します。ソフトリミットはハードリミットよりも小さくする必要があります。

Inode とブロッククォータに異なる猶予期間を設定できます。また、ユーザークォータ、グループクォータ、プロジェクトクォータに異なる猶予期間を設定することもできます。ユーザークォータ、グループクォータ、プロジェクトクォータの猶予期間が異なる場合、これらのクォータの猶予期間が経過すると、ソフト制限からハード制限に変更されます。

ユーザーがソフトリミットを超えた場合、Amazon FSx では、猶予期間が経過するまで、またはハードリミットに達するまで、クォータを超え続けることができます。猶予期間が終了すると、ソフトリミットはハードリミットに変換され、ストレージ使用量が定義されたブロッククォータまたは inode

クォータ制限を下回るまで、ユーザーはそれ以上の書き込み操作をブロックされます。猶予期間の開始時に、ユーザーは通知や警告を受けません。

クォータの設定と表示

ストレージクォータは、Linux ターミナルで Lustre ファイルシステム `lfs` コマンドを使用して設定します。`lfs setquota` コマンドはクォータ制限を設定し、`lfs quota` コマンドは、クォータ情報を表示します。

Lustre クォータコマンドの詳細については、Lustre ドキュメントサイトの Lustre オペレーションマニュアル を参照してください。

ユーザー、グループ、プロジェクトのクォータを設定する

ユーザー、グループ、プロジェクトのクォータを設定する `setquota` コマンドの構文は次のとおりです。

```
lfs setquota {-u|--user|-g|--group|-p|--project} username|groupname|projectid
             [-b block_softlimit] [-B block_hardlimit]
             [-i inode_softlimit] [-I inode_hardlimit]
             /mount_point
```

実行する条件は以下のとおりです。

- `-u` または `--user` は、クォータを設定するユーザーを指定します。
- `-g` または `--group` は、クォータを設定するグループを指定します。
- `-p` または `--project` は、クォータを設定するプロジェクトを指定します。
- `-b` は、ソフトリミットでブロッククォータを設定します。`-B` は、ハードリミットでブロッククォータを設定します。*`block_softlimit`* および *`block_hardlimit`* の両方はキロバイト単位で表され、最小値は 1024 KB です。
- `-i` は、ソフトリミットで i ノードクォータを設定します。`-I` は、ハードリミットで inode クォータを設定します。*`inode_softlimit`* および *`inode_hardlimit`* の両方は、inode の数で表され、最小値は 1024 inode です。
- *`mount_point`* は、ファイルシステムがマウントされたディレクトリです。

ユーザークォータの例: 次のコマンドは、`/mnt/fsx` にマウントされたファイルシステム上の `user1` に対して、5,000 KB のソフトブロック制限、8,000 KB のハードブロック制限、2,000 のソフト inode 制限、3,000 のハード inode 制限クォータを設定します。

```
sudo lfs setquota -u user1 -b 5000 -B 8000 -i 2000 -I 3000 /mnt/fsx
```

グループクォータの例: 次のコマンドは、/mnt/fsx にマウントされたファイルシステム上の group1 という名前のグループに対して、100,000 KB のハードブロック制限を設定します。

```
sudo lfs setquota -g group1 -B 100000 /mnt/fsx
```

プロジェクトクォータの例: まず、project コマンドを使用して、目的のファイルとディレクトリをプロジェクトに関連付けたことを確認してください。例えば、次のコマンドは、/mnt/fsxfs/dir1 ディレクトリのすべてのファイルとサブディレクトリを、プロジェクト ID が 100 のプロジェクトに関連付けます。

```
sudo lfs project -p 100 -r -s /mnt/fsxfs/dir1
```

次に、setquota コマンドを使用してプロジェクトクォータを設定します。次のコマンドは、/mnt/fsx にマウントされたファイルシステム上のプロジェクト 250 に対して、307,200 KB のソフトブロック制限、309,200 KB のハードブロック制限、10,000 のソフト inode 制限、11,000 のハード inode 制限クォータを設定します。

```
sudo lfs setquota -p 250 -b 307200 -B 309200 -i 10000 -I 11000 /mnt/fsx
```

猶予期間の設定

デフォルトの猶予期間は 1 週間です。次の構文を使用して、ユーザー、グループ、プロジェクトのデフォルトの猶予期間を調整できます。

```
lfs setquota -t {-u|-g|-p}
             [-b block_grace]
             [-i inode_grace]
             /mount_point
```

コードの説明は以下のとおりです。

- -t は、猶予期間が設定されることを示します。
- -u は、すべてのユーザーの猶予期間を設定します。
- -g は、すべてのグループの猶予期間を設定します。

- -p は、すべてのプロジェクトの猶予期間を設定します。
- -b は、ブロッククォータの猶予期間を設定します。-i は、inode クォータの猶予期間を設定します。*block_grace* および *inode_grace* の両方は、整数秒か XXwXXdXXhXXmXXs の形式で表されます。
- *mount_point* は、ファイルシステムがマウントされたディレクトリです。

次のコマンドは、ユーザーブロッククォータに対して 1,000 秒、ユーザー inode クォータに対して 1 週間と 4 日間の猶予期間を設定します。

```
sudo lfs setquota -t -u -b 1000 -i 1w4d /mnt/fsx
```

クォータの表示

quota コマンドは、ユーザークォータ、グループクォータ、プロジェクトクォータ、猶予期間に関する情報を表示します。

クォータコマンドの表示	表示されるクォータ情報
lfs quota */mount_point*	コマンドを実行するユーザーおよびユーザーのプライマリグループに関する一般的なクォータ情報 (ディスク使用量と制限)。
lfs quota -u *username* */mount_point*	特定のユーザーの一般的なクォータ情報。AWS アカウントの root ユーザーの認証情報を持つユーザーは、すべてのユーザーに対してこのコマンドを実行できますが、root 以外のユーザーはこのコマンドを実行して他のユーザーに関するクォータ情報を取得することはできません。
lfs quota -u *username* -v */mount_point*	特定のユーザーの一般的なクォータ情報と、各オブジェ

クォータコマンドの表示	表示されるクォータ情報
	クトストレージターゲット (OST) およびメタデータターゲット (MDT) の詳細なクォータ統計。AWS アカウントの root ユーザーの認証情報を持つユーザーは、すべてのユーザーに対してこのコマンドを実行できますが、root 以外のユーザーはこのコマンドを実行して他のユーザーに関するクォータ情報を取得することはできません。
`lfs quota -g groupname /mount_point`	特定のグループの一般的なクォータ情報。
`lfs quota -p projectid /mount_point`	特定のプロジェクトに関する一般的なクォータ情報。
`lfs quota -t -u /mount_point`	ユーザークォータのブロックと inode の猶予期間。
`lfs quota -t -g /mount_point`	グループクォータのブロックと inode の猶予期間。
`lfs quota -t -p /mount_point`	プロジェクトクォータのブロックと inode の猶予期間。

クォータおよび Simple Storage Service (Amazon S3) リンクバケット

FSx for Lustre ファイルシステムは Simple Storage Service (Amazon S3) データリポジトリにリンクできます。詳細については、「Amazon S3 バケットにファイルシステムにリンクする」を参照してください。

オプションで、ファイルシステムへのインポートパスとして、リンクされた S3 バケット内の特定のフォルダまたはプレフィックスを選択できます。Simple Storage Service (Amazon S3) のフォルダが

指定され、S3 からファイルシステムにインポートされると、そのフォルダのデータのみがクォータに適用されます。バケット全体のデータは、クォータ制限に対してカウントされません。

リンクされた S3 バケット内のファイルメタデータは、Simple Storage Service (Amazon S3) からインポートされたフォルダーと一致する構造を持つフォルダーにインポートされます。ファイルは、ファイルを所有するユーザーおよびグループの inode クォータにカウントされます。

ユーザーが `hsm_restore` または、ファイルを遅延ロードすると、ファイルのフルサイズは、ファイルの所有者に関連付けられたブロッククォータにカウントされます。例えば、ユーザー A がユーザー B が所有するファイルを遅延ロードすると、ストレージと inode の使用量はユーザー B のクォータにカウントされます。同様に、ユーザーが Amazon FSx API を使用してファイルをリリースすると、そのファイルを所有するユーザーまたはグループのブロッククォータからデータが解放されます。

HSM リストアと遅延ロードは root アクセスで実行されるため、クォータの強制を回避します。データがインポートされると、S3 で設定された所有権に基づいてユーザーまたはグループにカウントされます。これにより、ユーザーまたはグループがブロック制限を超える可能性があります。この場合、ファイルシステムに再度書き込みできるようにファイルを解放する必要があります。

同様に、自動インポートが有効になっているファイルシステムでは、S3 に追加されたオブジェクトの新しい inode が自動的に作成されます。新しい inode は、ルートアクセスで作成され、作成中にクォータ強制を回避します。新しい inode は、S3 内のオブジェクトの所有者に基づいて、ユーザーとグループにカウントされます。ユーザーとグループが自動インポートアクティビティに基づいて inode クォータを超えた場合、追加の容量を解放してクォータ制限を下回るためにファイルを削除する必要があります。

クォータとバックアップの復元

バックアップを復元すると、元のファイルシステムのクォータ設定が復元されたファイルシステムに実装されます。例えば、ファイルシステム A にクォータが設定され、ファイルシステム B がファイルシステム A のバックアップから作成されている場合、ファイルシステム A のクォータはファイルシステム B に適用されます。

ストレージ容量の管理

追加のストレージとスループットが必要になるので、FSx for Lustre ファイルシステムで設定されている SSD または HDD ストレージ容量を増やすことができます。FSx for Lustre ファイルシステムのスループットは、スループットキャパシティに応じて直線的に拡張されるため、スループット

キャパシティも同程度増加します。ストレージ容量を増やすには、Amazon FSx コンソール、AWS Command Line Interface (AWS CLI)、または Amazon FSx API を使用できます。

ファイルシステムのストレージ容量の更新をリクエストすると、Amazon FSx は自動的に新しいネットワークファイルサーバーを追加し、メタデータサーバーを拡張します。ストレージ容量のスケーリング中に、ファイルシステムが数分間使用できなくなる場合があります。ファイルシステムが利用できないときにクライアントによって発行されたファイルオペレーションは、透過的に再試行され、ストレージのスケーリングの完了後に成功します。ファイルシステムが使用できない間、ファイルシステムのステータスは `UPDATING` に設定されます。ストレージのスケーリングが完了すると、ファイルシステムのステータスは `AVAILABLE` に設定されます。

Amazon FSx は、既存のファイルサーバーと新しく追加されたファイルサーバー間でデータを透過的にリバランスするストレージ最適化プロセスを実行します。リバランシングは、ファイルシステムの可用性に影響を与えることなく、バックグラウンドで実行されます。リバランシング中に、データ移動のためにリソースが消費されるにつれて、ファイルシステムのパフォーマンスが低下することがあります。ほとんどのファイルシステムでは、ストレージの最適化には数時間から数日かかります。最適化フェーズでは、ファイルシステムにアクセスして使用できます。

Amazon FSx コンソール、CLI、および API を使用して、ストレージ最適化の進行状況をいつでも追跡できます。詳細については、「ストレージ容量の拡張をモニタリングする」を参照してください。

トピック

ストレージ容量を増やすときの考慮事項

ストレージ容量を増やすときに考慮すべき重要な事項をいくつか挙げます:

- 増加のみ - ファイルシステムのストレージ容量を 増やす ことしかできません。ストレージ容量を減らすことはできません。
- インクリメントの増加 - ストレージ容量を増やす場合は、[Increase storage capacity] (ストレージ容量増加) ダイアログボックスに記載されている増分値を使用します。

- 拡張するまでの時間 - 最後の拡張がリクエストされてから 6 時間経過するまでは、ファイルシステムのストレージ容量をさらに増やすことはできません。
- スループットキャパシティ – スループットキャパシティを増やすとスループットキャパシティが自動的に増加します。SSD キャッシュを使用する永続的な HDD ファイルシステムでは、HDD ストレージ容量の 20% のサイズの SSD キャッシュを維持するために、リードキャッシュのストレージ容量も同様に増加します。Amazon FSx は、ストレージおよびスループットキャパシティー単位の新しい値を計算し、[Increase storage capacity] ダイアログボックスに記入します。

> **Note**
>
> ファイルシステムのスループットキャパシティを更新しなくても、永続的な SSD ベースのファイルシステムのスループットキャパシティを個別に変更できます。詳細については、「プロビジョンドスループットキャパシティの管理」を参照してください。

- デプロイタイプ - スクラッチ 1 ファイルシステムを除くすべてのデプロイタイプのストレージ容量を増やすことができます。

ストレージ容量を増やす場合

空きストレージ容量が不足している場合は、ファイルシステムのストレージ容量を増やします。`FreeStorageCapacity` CloudWatch メトリクスを使用して、ファイルシステムで使用可能な空きストレージの量をモニタリングします。このメトリクスで Amazon CloudWatch アラームを作成し、特定のしきい値を下回ったときに通知を受け取ることができます。詳細については、「Amazon CloudWatch によるモニタリング」を参照してください。

CloudWatch メトリクスを使用して、ファイルシステムの継続的なスループット使用量をモニタリングすることができます。ファイルシステムに、より高いスループットキャパシティが必要であると判断した場合は、メトリクス情報を使用して、スループットキャパシティを増やす量を決定できます。ファイルシステムの現在のスループットを確認する方法については、「Amazon FSx for Lustre CloudWatch メトリクスを使用する方法」を参照してください。ストレージ容量がスループット容量にどのように影響するかについては、「Amazon FSx for Lustre のパフォーマンス」を参照してください。

また、ファイルシステムのストレージ容量と総スループットは、ファイルシステム詳細ページの [Summary] (概要) パネルで表示できます。

ストレージのスケーリングおよびバックアップリクエストの同時処理方法

ストレージスケーリングワークフローの開始直前、または進行中にバックアップをリクエストできます。Amazon FSx が 2 つのリクエストを処理する順序は次のとおりです。

- ストレージスケーリングワークフローが進行中の場合 (ストレージスケーリングのステータスは `IN_PROGRESS` およびファイルシステムのステータスは `UPDATING`) およびバックアップをリクエストすると、バックアップリクエストがキューに入れられます。バックアップタスクは、ストレージのスケーリングがストレージ最適化フェーズにあるときに開始されます (ストレージスケーリングのステータスは `UPDATED_OPTIMIZING` およびファイルシステムのステータスは `AVAILABLE`)。
- バックアップが進行中で、(バックアップのステータスは `CREATING`) ストレージスケーリングをリクエストすると、ストレージスケーリングリクエストがキューに入れられます。ストレージスケーリングワークフローは、Amazon FSx が Simple Storage Service (Amazon S3) にバックアップを転送するときに開始されます (バックアップステータスは `TRANSFERRING`)。

ストレージスケーリングリクエストが保留中であり、ファイルシステムのバックアップリクエストも保留中の場合、バックアップタスクの優先順位が高くなります。ストレージスケーリングタスクは、バックアップタスクが完了するまでスタートされません。

ストレージ容量を増やす

Amazon FSx コンソール、AWS CLI、または Amazon FSx API を使用して、ファイルシステムのストレージ容量を増やすことができます。

ファイルシステムのストレージ容量を増やすには (コンソール)

1. https://console.aws.amazon.com/fsx/ で Amazon FSx コンソールを開きます。
2. [ファイルシステム] に移動して、ストレージ容量を増やす Lustre ファイルシステムを選択します。
3. [Actions] (アクション) で、[Update storage capacity] (ストレージ容量更新) を選択します。または [概要] パネルで、ファイルシステムの [ストレージ容量] の横にある [更新] を選択して [ストレージ容量を増加] ダイアログボックスを表示します。
4. [Desired storage capacity] (希望するストレージ容量) で、ファイルシステムの現在のストレージ容量よりも大きい新しいストレージ容量を GiB 単位で指定します。

- 永続的な SSD またはスクラッチ 2 ファイルシステムの場合、この値は 2400 GiB の倍数にする必要があります。
- 永続 HDD ファイルシステムの場合、この値は 12 MBps/TiB ファイルシステムの場合は 6000 GiB の倍数、40 MBps/TiB ファイルシステムの場合は 1800 GiB の倍数にする必要があります。
- EFA 対応ファイルシステムの場合、この値は 125 MBps/TiB ファイルシステムの場合は 38400 GiB の倍数、250 MBps/TiB ファイルシステムの場合は 19200 GiB の倍数、500 MBps/TiB ファイルシステムの場合は 9600 GiB の倍数、1000 MBps/TiB ファイルシステムの場合は 4800 GiB の倍数である必要があります。

> **Note**
>
> スクラッチ 1 ファイルシステムのストレージ容量を増やすことはできません。

5. [Update] (更新) をクリックして、ストレージ容量の更新を開始します。
6. アップデートの進行状況は、[Update] (更新) タブのファイルシステム詳細ページでモニタリングできます。

ファイルシステムのストレージ容量を増やすには (CLI)

1. FSx for Lustre ファイルシステムのストレージ容量を増やすには、AWS CLI コマンド update-file-system を使用します。以下のパラメータを設定します:

   更新するファイルシステムの ID に `--file-system-id` を設定します。

   ストレージ容量の増加の量 (GiB 単位) の整数値に `--storage-capacity` を設定します。永続的な SSD またはスクラッチ 2 ファイルシステムの場合、この値は 2400 の倍数にする必要があります。永続 HDD ファイルシステムの場合、この値は 12 MBps/TiB ファイルシステムの場合は 6000 の倍数、40 MBps/TiB ファイルシステムの場合は 1800 の倍数にする必要があります。新しいターゲット値は、ファイルシステムの現在のストレージ容量よりも大きい値である必要があります。

   このコマンドは、永続的な SSD またはスクラッチ 2 ファイルシステムのストレージ容量目標値 9600 GiB を指定します。

```
$ aws fsx update-file-system \
```

```
    --file-system-id fs-0123456789abcdef0 \
    --storage-capacity 9600
```

2. AWS CLI コマンド describe-file-systems を使用して、更新の進捗状況をモニタリングすることができます。出力で `administrative-actions` を探します。

   詳細については、「AdministrativeAction」を参照してください。

ストレージ容量の拡張をモニタリングする

Amazon FSx コンソール、API、または AWS CLI を使用してストレージ容量拡張の進捗状況をモニタリングできます。

コンソールで拡大をモニタリングする

ファイルシステムの詳細ページの [Update] (更新) タブで、各更新タイプの最新の 10 件の更新ケースを表示できます。

表示できる情報は次のとおりです。

Update type] (更新タイプ

サポートされているタイプは [Storage capacity] (ストレージ容量) と [Storage optimization] (ストレージの最適化) です。

Target value] (ターゲット値

ファイルシステムのストレージ容量を更新する希望値です。

Status] (ステータス

ストレージ容量の現在のステータスが更新されます。指定できる値は次のとおりです。

- [Pending] (保留中) - Amazon FSx は更新リクエストを受信しましたが、処理をスタートしていません。
- [In progress] (進行中) - Amazon FSx が更新リクエストを処理しています。
- [Updated、Optimizing] (アップデート済み、最適化) - Amazon FSx により、ファイルシステムのストレージ容量が増加しました。ストレージ最適化プロセスでは、ファイルサーバ間でデータの再バランシングが行われています。
- [Completed] (完了) - ストレージ容量の増加は正常に完了しました。

- 失敗 - ストレージ容量の拡張に失敗しました。疑問符 (?) を選択し、ストレージの更新が失敗した理由の詳細を確認します。

進行 %

ストレージ最適化プロセスの進行状況を、完了率として表示します。

リクエスト時間

Amazon FSx が更新アクションリクエストを受信した時刻。

モニタリングは、AWS CLI と API で増加します

ファイルシステムストレージ容量の拡張リクエストを表示およびモニタリングするには、describe-file-systems AWS CLI コマンドと DescribeFileSystems API アクションを使用します。`AdministrativeActions` 配列には、管理アクションタイプごとに最新の更新アクションが 10 個表示されます。ファイルシステムのストレージ容量を増やすと、`FILE_SYSTEM_UPDATE` および `STORAGE_OPTIMIZATION` アクションの 2 つの `AdministrativeActions` が生成されます。

次の例は、describe-file-systems CLI コマンドのレスポンスの抜粋を示しています。ファイルシステムのストレージ容量は 4800 GB で、ストレージ容量を 9600 GB に増やすための保留中の管理アクションがあります。

```
{
    "FileSystems": [
        {
            "OwnerId": "111122223333",
            .
            .
            .
            "StorageCapacity": 4800,
            "AdministrativeActions": [
                {
                     "AdministrativeActionType": "FILE_SYSTEM_UPDATE",
                     "RequestTime": 1581694764.757,
                     "Status": "PENDING",
                     "TargetFileSystemValues": {
                         "StorageCapacity": 9600
                     }
                },
                {
                    "AdministrativeActionType": "STORAGE_OPTIMIZATION",
                    "RequestTime": 1581694764.757,
```

```
                    "Status": "PENDING",
                }
            ]
```

Amazon FSx はまず、FILE_SYSTEM_UPDATE アクションを処理し、新しいファイルサーバーをファイルシステムに追加します。新しいストレージがファイルシステムで使用可能になると、FILE_SYSTEM_UPDATE ステータスが UPDATED_OPTIMIZING に変わります。ストレージ容量は新しい大きな値を示し、Amazon FSx は STORAGE_OPTIMIZATION 管理アクションの処理を開始します。これは、describe-file-systems CLI コマンドのレスポンスの次の抜粋を示しています。

ProgressPercent プロパティには、ストレージ最適化プロセスの進行状況が表示されます。ストレージ最適化プロセスが正常に完了すると、FILE_SYSTEM_UPDATE アクションが COMPLETED に変更され、STORAGE_OPTIMIZATION アクションは表示されなくなります。

```
{
    "FileSystems": [
        {
            "OwnerId": "111122223333",
            .
            .
            .
            "StorageCapacity": 9600,
            "AdministrativeActions": [
                {
                    "AdministrativeActionType": "FILE_SYSTEM_UPDATE",
                    "RequestTime": 1581694764.757,
                    "Status": "UPDATED_OPTIMIZING",
                    "TargetFileSystemValues": {
                        "StorageCapacity": 9600
                }
                },
                {
                    "AdministrativeActionType": "STORAGE_OPTIMIZATION",
                    "RequestTime": 1581694764.757,
                    "Status": "IN_PROGRESS",
                    "ProgressPercent": 50,
                }
            ]
```

ストレージ容量の拡張に失敗した場合、FILE_SYSTEM_UPDATE アクションが FAILED に変更されます。FailureDetails プロパティでは、次の例に示すように、障害に関する情報を提供します。

```
{
    "FileSystems": [
        {
            "OwnerId": "111122223333",
            .
            .
            .
            "StorageCapacity": 4800,
            "AdministrativeActions": [
                {
                    "AdministrativeActionType": "FILE_SYSTEM_UPDATE",
                    "FailureDetails": {
                        "Message": "string"
                    },
                    "RequestTime": 1581694764.757,
                    "Status": "FAILED",
                    "TargetFileSystemValues":
                        "StorageCapacity": 9600
                }
            ]
```

プロビジョニングされた SSD 読み取りキャッシュの管理

インテリジェント階層化 ストレージクラスを使用してファイルシステムを作成する場合、頻繁にアクセスされるデータの読み取りに SSD レイテンシーを提供する SSD ベースの読み取りキャッシュを、GiB あたり最大 3 IOPS までプロビジョニングすることもできます。

頻繁にアクセスされるデータ用に、SSD 読み取りキャッシュを以下のいずれかのサイズ設定モードで構成できます:

- 自動 (スループットキャパシティに比例)。Automatic では、Amazon FSx for Lustre は、プロビジョニングされたスループットキャパシティに基づいて SSD データ読み取りキャッシュサイズを自動的に選択します。
- カスタム (ユーザープロビジョニングされた)。カスタムを使用すると、SSD 読み取りキャッシュのサイズをカスタマイズすることができ、ワークロードの要件に応じていつでもそのサイズを増減させることができます。
- ファイルシステムで SSD データ読み取りキャッシュを使用したくない場合は、キャッシュなし を選択します。

自動 (スループットキャパシティに比例) モードでは、Amazon FSx がファイルシステムのスループットキャパシティに基づいて、以下のデフォルト読み取りキャッシュサイズを自動的にプロビジョニングします。

プロビジョンドスループットキャパシティ (MBps)	自動 (スループットキャパシティに比例) モードの SSD 読み取りキャッシュ (GiB)	サポートされている SSD 読み取りキャッシュサイズ	
		最小 (GiB)	最大 (GiB)
4000 ごと	20000	32	131072

ファイルシステムを作成したら、読み取りキャッシュのサイジングモードとストレージ容量をいつでも変更できます。

トピック

SSD 読み取りキャッシュの更新時の考慮事項

SSD データ読み取りキャッシュを変更する際の重要な考慮事項を以下に示します:

- SSD 読み取りキャッシュを変更すると、その内容はすべて消去されます。つまり、SSD 読み取りキャッシュが再度満たされるまで、パフォーマンスレベルが低下する可能性があります。
- SSD 読み取りキャッシュの容量サイズを増減できます。ただし、これは 6 時間に 1 回しか実行できません。SSD 読み取りキャッシュをファイルシステムに追加または削除する場合、時間制限はありません。
- SSD 読み取りキャッシュのサイズは、変更するたびに 10% 以上増減する必要があります。

プロビジョニングされた SSD 読み取りキャッシュの更新

Amazon FSx コンソール、AWS CLI、または Amazon FSx API を使用して、SSD データ読み取りキャッシュを更新できます。

インテリジェント階層化ファイルシステムの SSD 読み取りキャッシュを更新します (CLI)

1. https://console.aws.amazon.com/fsx/ で Amazon FSx コンソールを開きます。
2. 左のナビゲーションペインで [File system] (ファイルシステム) を選択します。[ファイルシステム] リストで、SSD 読み取りキャッシュを更新する FSx for Lustre ファイルシステムを選択します。
3. SSD [概要] パネルで、ファイルシステムの [SSD 読み取りキャッシュ] 値の横にある [更新] を選択します。

   [SSD 読み取りキャッシュの更新] ダイアログボックスが表示されます。
4. 以下のように、データ読み取りキャッシュに使用する新しいサイジングモードを選択します:

- [自動 (スループットキャパシティに比例)] を選択して、スループットキャパシティに基づいてデータ読み取りキャッシュを自動的にサイズ設定します。
- データセットのおおよそのサイズがわかっていて、データ読み取りキャッシュをカスタマイズする場合は、カスタム (ユーザープロビジョニング) を選択します。カスタム を選択した場合は、希望する読み取りキャッシュ容量 を GiB 単位で指定する必要があります。
- インテリジェント階層化 ファイルシステムで SSD データ読み取りキャッシュを使用しない場合は、[なし] を選択します。

5. [更新] を選択します。

インテリジェント階層化ファイルシステムの SSD 読み取りキャッシュを更新します (CLI)

インテリジェント階層化 ファイルシステムの SSD データ読み取りキャッシュを更新するには、AWS CLI コマンド update-file-system または同等の UpdateFileSystem API アクションを使用します。以下のパラメータを設定します:

- `--file-system-id` を更新するファイルシステムの ID に設定します。
- SSD 読み取りキャッシュを変更するには、`--lustre-configuration DataReadCacheConfiguration` プロパティを使用します。このプロパティには、`SizeGiB` と `SizingMode` の 2 つのパラメータがあります:
  - SizeGiB - `USER_PROVISIONED` モードを使用するときに SSD 読み取りキャッシュのサイズを GiB 単位で設定します。
  - SizingMode - SSD 読み取りキャッシュのサイズ設定モードを設定します。
    - Intelligent-Tiering ファイルシステムで SSD 読み取りキャッシュを使用しない場合は、`NO_CACHE` に設定します。
    - SSD 読み取りキャッシュの正確なサイズを指定するには、`USER_PROVISIONED` に設定します。
    - SSD データ読み取りキャッシュをスループットキャパシティに基づいて自動的にサイズ設定するには、`PROPORTIONAL_TO_THROUGHPUT_CAPACITY` に設定します。

以下の例では、SSD 読み取りキャッシュを `USER_PROVISIONED` モードに更新し、サイズを 524288 GiB に設定します。

```
aws fsx update-file-system \
   --file-system-id fs-0123456789abcdef0 \
```

```
  --lustre-configuration
'DataReadCacheConfiguration={SizeGiB=524288,SizingMode=USER_PROVISIONED}'
```

describe-file-systems の AWS CLI コマンドを使用して、更新の進捗状況をモニタリングすることができます。出力で `AdministrativeActions` セクションを探します。

詳細については、「Amazon FSx リファレンス」の「AdministrativeAction」を参照してください。

SSD 読み取りキャッシュの更新のモニタリング

Amazon FSx コンソール、API、または AWS CLI を使用して、SSD 読み取りキャッシュの更新の進捗状況をモニタリングできます。

コンソールで更新をモニタリングする

ファイルシステムの更新は、[ファイルシステムの詳細] ページの [更新] タブでモニタリングできます。

SSD 読み取りキャッシュの更新では、以下の情報を表示できます:

Update type] (更新タイプ

サポートしているタイプは、SSD リードキャッシュサイジングモード と SSD リードキャッシュサイズ です。

Target value] (ターゲット値

ファイルシステムの SSD リードキャッシュサイジングモードまたは SSD リードキャッシュサイズの更新された値。

Status] (ステータス

更新の現在のステータス。指定できる値は次のとおりです。

- [Pending] (保留中) - Amazon FSx は更新リクエストを受信しましたが、処理をスタートしていません。
- [In progress] (進行中) - Amazon FSx が更新リクエストを処理しています。
- [Completed] (完了) - 更新は正常に終了しました。
- [Failed] (失敗) - 更新リクエストが失敗する。疑問符 (?) を選択して、ストレージの更新が失敗した理由の詳細を確認します。

Request time] (リクエスト時間

Amazon FSx が更新アクションリクエストを受信した時刻。

AWS CLI と API を使用した SSD 読み取りキャッシュの更新のモニタリング

describe-file-systems AWS CLI コマンドと DescribeFileSystems API オペレーションを使用し、ファイルシステム SSD 読み取りキャッシュ更新リクエストを表示してモニタリングできます。AdministrativeActions 配列には、管理アクションタイプごとに最新の更新アクションが 10 件を表示されます。ファイルシステムの SSD 読み取りキャッシュを更新すると、FILE_SYSTEM_UPDATE AdministrativeActions が生成されます。

次の例は、describe-file-systems CLI コマンドのレスポンスの抜粋を示しています。ファイルシステムには、SSD 読み取りキャッシュのサイズ設定モードを USER_PROVISIONED に、SSD 読み取りキャッシュサイズを 524288 に変更する保留中の管理アクションがあります。

```
"AdministrativeActions": [
    {
        "AdministrativeActionType": "FILE_SYSTEM_UPDATE",
        "RequestTime": 1586797629.095,
        "Status": "PENDING",
        "TargetFileSystemValues": {
            "LustreConfiguration": {
                "DataReadCacheConfiguration": {
                     "SizingMode": "USER_PROVISIONED"
                     "SizeGiB": 524288,
                }
            }
        }
    }
]
```

新しい SSD 読み取りキャッシュ設定がファイルシステムで使用可能になると、FILE_SYSTEM_UPDATE ステータスが COMPLETED に変わります。SSD の読み取りキャッシュ更新リクエストが失敗すると、FILE_SYSTEM_UPDATE アクションのステータスは FAILED に変わります。

メタデータパフォーマンスの管理

Amazon FSx コンソール、Amazon FSx API、または AWS Command Line Interface (AWS CLI) を使用して、エンドユーザーまたはアプリケーションを中断することなく、FSx for Lustre ファイルシステムのメタデータ設定を更新できます。更新手順では、ファイルシステムのプロビジョニングされたメタデータ IOPS の数を増やします。

> **Note**
>
> 拡張メタデータは 2.15 ファイルシステムでのみ使用できます。メタデータパフォーマンスは、永続 2 デプロイタイプと指定されたメタデータ設定で作成された FSx for Lustre ファイルシステムでのみ向上できます。ファイルシステムの作成時にメタデータ設定が指定されていない場合、FSx for Lustre ファイルシステムのメタデータ設定を追加または更新することはできません。これは、拡張メタデータパフォーマンスをサポートしていなかった 2.12 ファイルシステムのバックアップから復元されたファイルシステム、またはメタデータ構成が指定されていない 2.15 ファイルシステムから復元されたファイルシステムにも適用されます。

ファイルシステムのメタデータパフォーマンスの向上は、数分で使用できます。メタデータのパフォーマンス向上リクエストが少なくとも 6 時間離れていれば、メタデータのパフォーマンスはいつでも更新できます。メタデータパフォーマンスのスケーリング中に、ファイルシステムが数分間使用できなくなる場合があります。ファイルシステムが利用できないときにクライアントによって発行されたファイルオペレーションは、透過的に再試行され、メタデータパフォーマンスのスケーリングの完了後に成功します。新しいメタデータパフォーマンス向上は、利用可能になった後に請求されます。

Amazon FSx コンソール、CLI、および API を使用して、メタデータのパフォーマンス向上の進行状況をいつでも追跡できます。詳細については、「メタデータ設定の更新のモニタリング」を参照してください。

トピック

Lustre メタデータパフォーマンス設定

プロビジョニングされたメタデータ IOPS の数によって、ファイルシステムでサポートできるメタデータオペレーションの最大レートが決まります。

ファイルシステムを作成するときは、メタデータ設定モードを選択します:

- SSD ファイルシステムでは、Amazon FSx でファイルシステムのストレージ容量に基づいてファイルシステムのメタデータ IOPS を自動的にプロビジョニングおよびスケーリングする場合は、[自動] を選択します。インテリジェント階層化 ファイルシステムは自動モードをサポートしていないことに留意してください。
- SSD ファイルシステムでは、メタデータ IOPS の数を指定してファイルシステムのプロビジョニングする場合は、[ユーザープロビジョニング] を選択します。
- インテリジェント階層化ファイルシステムの場合は、ユーザープロビジョニングモードを選択する必要があります。ユーザープロビジョニングモードでは、ファイルシステム用にプロビジョニングするメタデータ IOPS の数を指定できます。

SSD ファイルシステムでは、自動モードからユーザープロビジョニングモードにいつでも切り替えることができます。ファイルシステムでプロビジョニングされたメタデータ IOPS の数が、自動モードでプロビジョニングされたメタデータ IOPS のデフォルトの数と一致する場合、ユーザープロビジョニングモードから自動モードに切り替えることもできます。インテリジェント階層化 ファイルシステムがサポートするのはユーザープロビジョニングモードのみです。そのためメタデータ設定モードを切り替えることはできません。

有効なメタデータ IOPS 値は以下のとおりです:

- SSD ファイルシステムでは、有効なメタデータ IOPS 値は、1500、3000、6000、および 192000 までの 12000 の倍数です。
- インテリジェント階層化 ファイルシステムの場合、有効なメタデータ IOPS 値は 6000 と 12000 です。

ワークロードのメタデータパフォーマンスが自動モードでプロビジョニングされたメタデータ IOPS の数を超える場合は、ユーザープロビジョニングモードを使用してファイルシステムのメタデータ IOPS 値を増やすことができます。

ファイルシステムのメタデータサーバー設定の現在の値は、次のように表示できます。

- コンソールの使用 – ファイルシステムの詳細ページの [概要] パネルの、[メタデータ IOPS] フィールドには、プロビジョニングされたメタデータ IOPS の現在の値と、ファイルシステムの現在のメタデータ設定モードが表示されます。
- CLI または API の使用 – describe-file-systems CLI コマンドまたは DescribeFileSystems API オペレーションを使用して、`MetadataConfiguration` プロパティを検索します。

メタデータパフォーマンスを向上させる際の考慮事項

メタデータパフォーマンスを向上させる際の重要な考慮事項をいくつか示します。

- メタデータパフォーマンスの向上のみ — ファイルシステムのメタデータ IOPS の数を増やすのみで、メタデータ IOPS の数を減らすことはできません。
- 自動モードでのメタデータ IOPS の指定はサポートされていません – 自動モードのファイルシステム上のメタデータ IOPS の数を指定することはできません。ユーザープロビジョニングモードに切り替えてから、リクエストを行う必要があります。詳細については、「メタデータ設定モードの変更」を参照してください。
- スケーリング前に書き込まれたデータのメタデータ IOPS – メタデータ IOPS を 12000 を超えてスケーリングすると、FSx for Lustre はファイルシステムに新しいメタデータサーバーを追加します。新しいメタデータは、パフォーマンスを向上させるためにすべてのサーバーに自動的に分散されます。ただし、スケーリング前に作成された既存のメタデータとサブディレクトリは元のサーバーに残り、メタデータ IOPS は増加しません。
- 向上リクエスト間の時間 — 最後の向上がリクエストされてから 6 時間後まで、ファイルシステムでメタデータパフォーマンスをさらに向上させることはできません。
- メタデータパフォーマンスと SSD ストレージの同時向上 – メタデータパフォーマンスとファイルシステムストレージ容量を同時にスケール時間枠することはできません。

メタデータパフォーマンスを向上させるタイミング

ファイルシステムでデフォルトでプロビジョニングされているよりも高いレベルのメタデータパフォーマンスを必要とするワークロードを実行する必要がある場合は、メタデータ IOPS の数を増やします。ファイルシステムで消費しているプロビジョニングされたメタデータサーバーのパフォーマンスの割合を示す `Metadata IOPS Utilization` グラフを使用して、AWS マネジメントコンソール のメタデータパフォーマンスをモニタリングできます。

また、より詳細な CloudWatch メトリクスを使用してメタデータパフォーマンスをモニタリングすることもできます。CloudWatch メトリクスには、ディスク IO を必要とするメタデータサーバーオペレーションのボリュームを提供する `DiskReadOperations` および `DiskWriteOperations` が含まれます。また、ファイルとディレクトリの作成、統計、読み取り、削除などのメタデータオペレー時間枠ションの詳細なメトリクスも提供します。詳細については、「FSx for Lustre メタデータメトリクス」を参照してください。

メタデータパフォーマンスの向上

Amazon FSx コンソール、AWS CLI、または Amazon FSx API を使用して、ファイルシステムのメタデータパフォーマンスを向上させることができます。

ファイルシステムのメタデータパフォーマンスを向上させるには (コンソール)

1. https://console.aws.amazon.com/fsx/ で Amazon FSx コンソールを開きます。
2. 左のナビゲーションペインで [File system] (ファイルシステム) を選択します。[ファイルシステム] リストで、メタデータのパフォーマンスを向上させる FSx for Lustre ファイルシステムを選択します。
3. [アクション] で、[メタデータ IOPS の更新] を選択します。または [概要] パネルで、ファイルシステムの [メタデータ IOPS] の横にある [更新] を選択します。

   [メタデータ IOPS の更新] ダイアログボックスが表示されます。
4. [ユーザープロビジョニング] を選択します。
5. [希望するメタデータ IOPS] の場合は、新しいメタデータ IOPS 値を選択します。値は、メタデータ IOPS 値と同等かそれ以上である必要があります。

   - SSD ファイルシステムでは、有効な値は、`1500`、`3000`、`6000`、`12000`、および `192000` までの `12000` の倍数です。
   - インテリジェント階層化ファイルシステムの場合、有効な値は `6000` と `12000` です。
6. [更新] を選択します。

ファイルシステム (CLI) のメタデータパフォーマンスを向上させるには

FSx for Lustre ファイルシステムのメタデータパフォーマンスを向上させるには、update-file-system の AWS CLI コマンドを使用します (UpdateFileSystem は同等の API アクションです)。以下のパラメータを設定します:

- `--file-system-id` を更新するファイルシステムの ID に設定します。
- メタデータパフォーマンスを向上させるには、`--lustre-configuration MetadataConfiguration` プロパティを使用します。このプロパティには、`Mode` と `Iops` の 2 つのパラメータがあります。
  1. ファイルシステムが USER_PROVISIONED モードの場合、`Mode` の使用はオプションです (使用する場合は `Mode` を `USER_PROVISIONED` に設定します)。

SSD ファイルシステムが AUTOMATIC モードの場合は、Mode を USER_PROVISIONED に設定します (これにより、メタデータ IOPS 値が増加するだけでなく、ファイルシステムモードが USER_PROVISIONED に切り替わります)。

2. SSD ファイルシステムでは、Iops を 1500、3000、6000、12000、または 192000 までの 12000 の倍数の値に設定します。インテリジェント階層化 ファイルシステムの場合は、Iops を 6000 または 12000 に設定します。値は、メタデータ IOPS 値と同等かそれ以上である必要があります。

次の例では、プロビジョニングされたメタデータ IOPS を 12000 に更新します。

```
aws fsx update-file-system \
    --file-system-id fs-0123456789abcdef0 \
    --lustre-configuration 'MetadataConfiguration={Mode=USER_PROVISIONED,Iops=12000}'
```

メタデータ設定モードの変更

SSD ベースのファイルシステムでは、以下の手順で説明するように、AWS コンソールと CLI を使用して既存のファイルシステムのメタデータ設定モードを変更できます。

自動モードからユーザープロビジョニングモードに切り替える場合は、現在のファイルシステムのメタデータ IOPS 値以上のメタデータ IOPS 値を指定する必要があります。

ユーザープロビジョニングモードから自動モードに切り替えることをリクエストし、現在のメタデータ IOPS 値が自動デフォルトより大きい場合、Amazon FSx はリクエストを拒否します。これは、メタデータ IOPS のダウンスケーリングがサポートされていないためです。モードスイッチのブロックを解除するには、モードスイッチを再度有効にするために、現在のメタデータ IOPS を自動モードで一致させるようにストレージ容量を増やす必要があります。

Amazon FSx コンソール、AWS CLI、または Amazon FSx API を使用して、ファイルシステムのメタデータ設定モードを変更できます。

ファイルシステムのメタデータ設定モードを変更するには (コンソール)

1. https://console.aws.amazon.com/fsx/ で Amazon FSx コンソールを開きます。
2. 左のナビゲーションペインで [File system] (ファイルシステム) を選択します。[ファイルシステム] リストで、メタデータ設定モードを変更する FSx for Lustre ファイルシステムを選択します。

3. [アクション] で、[メタデータ IOPS の更新] を選択します。または [概要] パネルで、ファイルシステムの [メタデータ IOPS] の横にある [更新] を選択します。

   [メタデータ IOPS の更新] ダイアログボックスが表示されます。

4. 以下のいずれかを行ってください。

   - ユーザープロビジョニングモードから自動モードに切り替えるには、[自動] を選択します。
   - 自動モードからユーザープロビジョニングモードに切り替えるには、[ユーザープロビジョニング] を選択します。次に、[目的のメタデータ IOPS] に、現在のファイルシステムのメタデータ IOPS 値以上のメタデータ IOPS 値を指定します。

5. [更新] を選択します。

SSD ファイルシステムのメタデータ構成モードを変更するには (CLI)

SSD FSx for Lustre ファイルシステムのメタデータ設定モードを変更するには、update-file-system の AWS CLI コマンドを使用します (UpdateFileSystem は同等の API アクションです)。以下のパラメータを設定します:

- `--file-system-id` を更新するファイルシステムの ID に設定します。
- SSD ベースのファイルシステムでメタデータ設定モードを変更するには、`--lustre-configuration MetadataConfiguration` プロパティを使用します。このプロパティには、`Mode` と `Iops` の 2 つのパラメータがあります。
  - SSD ファイルシステムから AUTOMATIC モードから USER_PROVISIONED モードに切り替えるには、`Mode` を `USER_PROVISIONED` に、`Iops` をメタデータ IOPS 値を現在のファイルシステムのメタデータ IOPS 値以上に設定します。例:

```
aws fsx update-file-system \
    --file-system-id fs-0123456789abcdef0 \
    --lustre-configuration
 'MetadataConfiguration={Mode=USER_PROVISIONED,Iops=96000}'
```

  - USER_PROVISIONED モードから AUTOMATIC モードに切り替えるには、`Mode` を `AUTOMATIC` に設定し、`Iops` パラメータを使用しないでください。例:

```
aws fsx update-file-system \
    --file-system-id fs-0123456789abcdef0 \
    --lustre-configuration 'MetadataConfiguration={Mode=AUTOMATIC}'
```

メタデータ設定の更新のモニタリング

Amazon FSx コンソール、API、または AWS CLI を使用して、メタデータ設定の更新の進捗状況をモニタリングできます。

メタデータ設定の更新のモニタリング (コンソール)

メタデータ設定の更新は、[ファイルシステムの詳細] ページの [更新] タブでモニタリングできます。

メタデータ設定の更新について、次の情報を表示できます。

Update type] (更新タイプ

サポートされているタイプは、[メタデータ IOPS] と [メタデータ設定モード] です。

Target value] (ターゲット値

ファイルシステムのメタデータ IOPS またはメタデータ設定モードの更新値。

Status] (ステータス

更新の現在のステータス。指定できる値は次のとおりです。

- [Pending] (保留中) - Amazon FSx は更新リクエストを受信しましたが、処理をスタートしていません。
- [In progress] (進行中) - Amazon FSx が更新リクエストを処理しています。
- [Completed] (完了) - 更新は正常に終了しました。
- [Failed] (失敗) - 更新リクエストが失敗する。疑問符 (?) を選択して、ストレージの更新が失敗した理由の詳細を確認します。

Request time] (リクエスト時間

Amazon FSx が更新アクションリクエストを受信した時刻。

メタデータ設定の更新のモニタリング (CLI)

メタデータ設定の更新リクエストを表示してモニタリングするには、describe-file-systems の AWS CLI コマンドと DescribeFileSystems API アクションを使用します。`AdministrativeActions` 配列には、管理アクションタイプごとに最新の更新アクションが 10 件を表示されます。ファイルシステムのメタデータパフォーマンスまたはメタデータ設定モードを更新すると、`FILE_SYSTEM_UPDATE` `AdministrativeActions` が生成されます。

次の例は、describe-file-systems CLI コマンドのレスポンスの抜粋を示しています。ファイルシステムには、メタデータ IOPS を 96000 に、メタデータ設定モードを USER_PROVISIONED に増やすための保留中の管理アクションがあります。

```
"AdministrativeActions": [
    {
        "AdministrativeActionType": "FILE_SYSTEM_UPDATE",
        "RequestTime": 1678840205.853,
        "Status": "PENDING",
        "TargetFileSystemValues": {
            "LustreConfiguration": {
                "MetadataConfiguration": {
                    "Iops": 96000,
                    "Mode": USER_PROVISIONED
                }
            }
        }
    }
]
```

Amazon FSx は FILE_SYSTEM_UPDATE アクションを処理し、ファイルシステムのメタデータ IOPS とメタデータ設定モードを変更します。新しいメタデータリソースがファイルシステムで使用可能になると、FILE_SYSTEM_UPDATE ステータスが COMPLETED に変わります。

メタデータ設定の更新リクエストが失敗した場合、次の例に示すように、FILE_SYSTEM_UPDATE アクションが FAILED に変更されます。FailureDetails プロパティでは、障害に関する情報が表示されます。

```
"AdministrativeActions": [
    {
        "AdministrativeActionType": "FILE_SYSTEM_UPDATE",
        "RequestTime": 1678840205.853,
        "Status": "FAILED",
        "TargetFileSystemValues": {
            "LustreConfiguration": {
                "MetadataConfiguration": {
                    "Iops": 96000,
                    "Mode": USER_PROVISIONED
                }
            }
        },
        "FailureDetails": {
```

```
            "Message": "failure-message"
        }
    }
]
```

プロビジョンドスループットキャパシティの管理

すべての FSx for Lustre ファイルシステムには、ファイルシステムの作成時に設定されたスループットキャパシティがあります。SSD または HDD ストレージを使用するファイルシステムの場合、スループットキャパシティは 1 秒あたりのメガバイト/テビバイト (MBps/TiB) で測定されます。インテリジェント階層化ストレージを使用するファイルシステムの場合、スループットキャパシティはファイルシステムのメガバイト/秒 (MBps) で測定されます。スループット容量は、ファイルシステムをホストしているファイルサーバーがファイルデータを提供できる速度を決定する要素の 1 つです。スループット容量では、秒ごとの I/O オペレーション (IOPS) が高くなり、ファイルサーバー上のデータをキャッシュするためのメモリが増えます。詳細については、「Amazon FSx for Lustre のパフォーマンス」を参照してください。

永続的な SSD ベースのファイルシステムのスループット階層は、ストレージ単位あたりのファイルシステムのスループットの値を増減することで変更できます。有効な値は、ファイルシステムのデプロイタイプによって以下のように異なります。

- 永続 1 SSD ベースのデプロイタイプの場合、有効な値は 50、100、および 200 MBps/TiB です。
- 永続 2 SSD ベースのデプロイタイプの場合、有効な値は 125、250、500、および 1000 MBps/TiB です。

ファイルシステムの合計スループットキャパシティの値を増やすことで、インテリジェント階層化 ファイルシステムのスループットキャパシティを変更できます。有効な値は 4,000 MBps または 4,000 MBps の増分で、最大 2,000,000 MBps です。

ファイルシステムのスループットキャパシティの現在の値は、以下のようにして表示できます:

- コンソールの使用 – ファイルシステムの詳細ページの [概要] パネルで、[ストレージ単位あたりのスループット] フィールドには SSD ベースのファイルシステムの現在の値が表示され、[スループットキャパシティ] フィールドには インテリジェント階層化 ファイルシステムの現在の値が表示されます。
- CLI または API の使用 – describe-file-systems CLI コマンドまたは DescribeFileSystems API オペレーションを使用して、`PerUnitStorageThroughput` プロパティを検索します。

ファイルシステムのスループットキャパシティを変更すると、背後で Amazon FSx は SSD ファイルシステムのシステムファイルサーバーを切り替えるか Intelligent-Tiering ファイルシステムの新しいファイルサーバーを追加ます。スループットキャパシティのスケーリング中、ファイルシステムは最大 1 時間使用できなくなります。ファイルシステムで使用可能になると、新しいスループット容量が課金されます。

トピック

スループットキャパシティを更新する際の考慮事項

スループットキャパシティを更新する際に考慮すべき重要な事項は次のとおりです。

- 増減する – SSD ベースのファイルシステムのスループットキャパシティの量を増減できます。Intelligent-Tiering ファイルシステムのスループットキャパシティの量を増やすことだけができます。
- 更新増分 – スループット容量を変更する場合、SSD ベースのファイル システムの場合は [スループット階層の更新] ダイアログ ボックス、Intelligent-Tiering ファイル システムの場合は [スループット容量の更新] ダイアログ ボックスにリストされている増分を使用してください。
- 増加させる時間間隔 – 最後のリクエストから 6 時間経過し、かつ、スループット最適化プロセスが完了するまでは、ファイルシステムでスループットキャパシティを変更することはできません。
- SSD 読み取りキャッシュの自動スケーリング – SSD 読み取りキャッシュのデフォルトモード (スループットキャパシティに比例) の場合、Amazon FSx はプロビジョニングするスループットキャパシティの MBps ごとに 5 GiB のデータストレージを自動的にプロビジョニングします。ファイルシステムのスループットキャパシティをスケールすると、Amazon FSx は新しく追加されたファイルサーバーに追加のキャッシュストレージをアタッチすることで、SSD データキャッシュを自動的にスケールします。
- デプロイタイプ – 永続 SSD ベースまたは Intelligent-Tiering のデプロイタイプのスループットキャパシティしか更新できません。EFA 対応 SSD ベースのファイルシステムのスループットキャパシティを変更することはできません。

スループット容量を変更するタイミング

Amazon FSx は Amazon CloudWatch と統合され、ファイルシステムの継続的なスループット使用レベルをモニタリングできます。ファイルシステムを介してドライブできるパフォーマンス (スループットと IOPS) は、ファイルシステムのスループットキャパシティ、ストレージ容量、ストレージクラスに加えて、特定のワークロードの特性によって異なります。ファイルシステムの現在のスループットを確認する方法については、「Amazon FSx for Lustre CloudWatch メトリクスを使用する方法」を参照してください。CloudWatch メトリクスの詳細については、「Amazon CloudWatch によるモニタリング」を参照してください。

スループットキャパシティの変更

Amazon FSx コンソール、 AWS Command Line Interface (AWS CLI)、または Amazon FSx API を使用して、FSx for Lustre ファイルシステムのスループットキャパシティを変更できます。

SSD ファイルシステムのスループットキャパシティを変更するには (コンソール)

1. https://console.aws.amazon.com/fsx/ で Amazon FSx コンソールを開きます。
2. [ファイルシステム] に移動し、スループットキャパシティを変更する FSx for Lustre ファイルシステムを選択します。
3. [アクション] には、[スループット階層を更新] を選択します。または、[概要] パネルで、ファイルシステムの [ストレージ単位あたりのスループット] の横にある [更新] を選択します。

   [スループット階層を更新] ウィンドウが表示されます。
4. 一覧から [希望するストレージ単位あたりのスループット] の新しい値を選択します。
5. [Update] (更新) を選択して、スループット容量の更新を開始します。

> **Note**
>
> ファイルシステムは更新中、ごくわずかな期間利用できないことがあります。

SSD ファイルシステムのスループットキャパシティを変更するには (CLI)

- ファイルシステムのスループットキャパシティを変更するには、update-file-system CLI コマンド (または同等の UpdateFileSystem API オペレーション) を使用します。以下のパラメータを設定します:
  - `--file-system-id` を更新するファイルシステムの ID に設定します。

- `--lustre-configuration PerUnitStorageThroughput` は、永続 1 SSD ファイルシステムの場合は `50`、`100`、または `200` MBps/TiB の値に、永続 2 SSD ファイルシステムの場合は `125`、`250`、`500`、または `1000` MBps/TiB の値に設定します。

このコマンドは、ファイルシステムに対してスループットキャパシティを 1000 MBps/TiB に設定することを指定します。

```
aws fsx update-file-system \
    --file-system-id fs-0123456789abcdef0 \
    --lustre-configuration PerUnitStorageThroughput=1000
```

インテリジェント階層化 ファイルシステムのスループットキャパシティを変更するには (コンソール)

1. https://console.aws.amazon.com/fsx/ で Amazon FSx コンソールを開きます。
2. [ファイルシステム] に移動し、スループットキャパシティを変更する FSx for Lustre ファイルシステムを選択します。
3. [Actions] (アクション) の場合、[Update throughput capacity] (スループット容量の更新) を選択します。または、[Summary] (概要) パネルでファイルシステムの [Throughput capacity] (スループット容量) の横にある [Update] (更新) を選択します。

   スループットキャパシティの更新 ウィンドウが表示されます。
4. リストから [希望するスループットキャパシティ] の新しい値を選択します。

   Amazon FSx は、キャッシュの内容をクリアしないように、データ読み取りキャッシュを自動的にスケーリングします。
5. [Update] (更新) を選択して、スループット容量の更新を開始します。

> **Note**
>
> ファイルシステムは更新中、ごくわずかな期間利用できないことがあります。

インテリジェント階層化 ファイルシステムのスループットキャパシティを変更する (CLI)

- ファイルシステムのスループットキャパシティを変更するには、update-file-system CLI コマンド (または同等の UpdateFileSystem API オペレーション) を使用します。以下のパラメータを設定します:

  - `--file-system-id` を更新するファイルシステムの ID に設定します。
  - データ読み取りキャッシュがスループットキャパシティモードに比例して設定されている場合は、`--lustre-configuration ThroughputCapacity` を最大 `4000` MBps の刻み幅のスループットレベルで、最大 `2000000` MBps までで設定します。

    データ読み取りキャッシュがユーザープロビジョニングモードで設定されている場合は、`--lustre-configuration DataReadCacheConfiguration` プロパティを使用してデータ読み取りキャッシュを指定する必要があります。サーバーごとのキャッシュストレージ比率を同じに維持し、新しい SizeGiB を指定する必要があります。そうしないと、リクエストは拒否されます。

  このコマンドは、スループットキャパシティに比例するモードで設定された読み取りキャッシュを使用するファイルシステムに対して、スループットキャパシティを 8000 MBps に設定するよう指定します。

```
aws fsx update-file-system \
    --file-system-id fs-0123456789abcdef0 \
    --lustre-configuration '{
      "ThroughputCapacity": 8000
      }'
```

  このコマンドは、ユーザープロビジョニングモードで設定された読み取りキャッシュを使用するファイルシステムのスループットキャパシティを 8000 MBps に設定することを指定します。

```
aws fsx update-file-system \
    --file-system-id fs-0123456789abcdef0 \
    --lustre-configuration {
        "ThroughputCapacity": 8000,
        "DataReadCacheConfiguration": '{
             "SizingMode":"USER_PROVISIONED"
             "SizeGiB":1000
             # New size should be cache storage allocated per server multiplied by
 number of file servers
```

```
        }
}'
```

スループット容量の変更のモニタリング

Amazon FSx コンソール、API、および AWS CLIを使用して、スループット容量変更プロセスをモニタリングできます。

スループットキャパシティの変更のモニタリング (コンソール)

- [ファイルシステムの詳細] ページの [更新] タブに、更新アクションタイプごとに最新の更新アクションを 10 件表示できます。

  スループット容量の更新アクションでは、次の情報を表示できます。

  [Update type] (更新タイプ)

  サポートされているタイプは [単位あたりのストレージスループット] です。

  [Target value] (ターゲット値)

  ファイルシステムのストレージ単位あたりスループット容量の変更後の値として望ましい値。

  [Status] (ステータス)

  更新の現在のステータス。スループット容量の更新では、指定できる値は次のとおりです:

  - [Pending] (保留中) - Amazon FSx は更新リクエストを受信しましたが、処理を開始していません。
  - [In progress] (進行中) - Amazon FSx が更新リクエストを処理しています。
  - [更新、最適化] – Amazon FSx は、ファイルシステムのネットワーク I/O、CPU、メモリリソースを更新しました。新しいディスク I/O パフォーマンスレベルを書き込み操作に利用できます。読み取り操作では、ファイルシステムがこの状態ではなくなるまで、前のレベルと新しいレベル間でディスク I/O パフォーマンスが表示されます。
  - [Completed] (完了) - スループット容量の更新が正常に完了しました。
  - [Failed] (失敗) - スループット容量の更新に失敗しました。疑問符 (?) を選択して、スループットの更新が失敗した理由の詳細を確認します。

[Request time] (リクエストタイム)

Amazon FSx が更新リクエストを受信した時刻。

ファイルシステムの更新のモニタリング (CLI)

- describe-file-systems CLI コマンドおよび DescribeFileSystems API アクションを使用して、ファイルシステムのスループット容量変更リクエストを表示し、モニタリングできます。AdministrativeActions 配列には、管理アクションタイプごとに最新の更新アクションが 10 件を表示されます。ファイルシステムのスループット容量を変更すると、FILE_SYSTEM_UPDATE 管理アクションが生成されます。

  次の例は、describe-file-systems CLI コマンドのレスポンスの抜粋を示しています。ファイルシステムのストレージ単位あたりの目標スループットは 500 MBps/TiB です。

```
.
.
.
"AdministrativeActions": [
    {
        "AdministrativeActionType": "FILE_SYSTEM_UPDATE",
        "RequestTime": 1581694764.757,
        "Status": "PENDING",
        "TargetFileSystemValues": {
          "LustreConfiguration": {
            "PerUnitStorageThroughput": 500
          }
        }
    }
]
```

  Amazon FSx でアクションが正常に処理されると、ステータスは COMPLETED に変更されます。新しいスループット容量がファイルシステムで使用可能になり、PerUnitStorageThroughput プロパティで表示されます。

  スループット容量の変更が失敗した場合、ステータスは FAILED に代わり、FailureDetails プロパティは障害に関する情報を提供します。

Lustre データ圧縮

Lustre データ圧縮機能を使用すると、高性能な Amazon FSx for Lustre ファイルシステムおよびバックアップストレージのコスト削減を実現できます。データ圧縮が有効になっている場合、Amazon FSx for Lustre は、新しく書き込まれたファイルをディスクに書き込む前に自動的に圧縮し、読み取り時に自動的に解凍します。

データ圧縮は LZ4 アルゴリズムを使用します。LZ4 アルゴリズムは、ファイルシステムのパフォーマンスに悪影響を及ぼすことなく、高レベルの圧縮を実現するように最適化されています。LZ4 は、圧縮速度と圧縮ファイルサイズのバランスをとり、Lustre コミュニティに信頼されているパフォーマンス指向のアルゴリズムです。通常、データ圧縮を有効にしても、レイテンシーに対して測定可能な影響は生じません。

データ圧縮は、Amazon FSx for Lustre ファイルサーバーとストレージ間で転送されるデータの量を減らします。圧縮ファイル形式をまだ使用していない場合は、データ圧縮を使用するときにファイルシステム全体のスループットキャパシティが増加します。データ圧縮に関連するスループットキャパシティの増加は、フロントエンドネットワークのインターフェイスカードを飽和させた後に制限されます。

例えば、ファイルシステムが PERSISTENT-50 SSD デプロイタイプの場合、ネットワークスループットのベースラインはストレージの TiB あたり 250 MBps です。ディスクスループットのベースラインは、TiB あたり 50 MBps です。データ圧縮を使用すると、ディスクスループットが TiB あたり 50 MBps から、ベースラインネットワークスループット制限である TiB あたり最大 250 MBps に増加する可能性があります。ネットワークおよびディスクのスループット制限の詳細については、「SSD および HDD ストレージクラスのパフォーマンス特性」の「ファイルシステムのパフォーマンス表」を参照してください。データ圧縮パフォーマンスの詳細については、AWS ストレージブログの「Amazon FSx for Lustre データ圧縮を使用してパフォーマンスを向上させながらコストを削減する」を参照してください。

トピック

- データ圧縮を管理する
- 以前に書き込まれたファイルの圧縮
- ファイルサイズの表示
- CloudWatch メトリクスの使用

データ圧縮を管理する

新しい Amazon FSx for Lustre ファイルシステムを作成するときに、データ圧縮を有効または無効にすることができます。コンソール、AWS CLI、または API から Amazon FSx for Lustre ファイルシステムを作成すると、データ圧縮はデフォルトで無効になっています。

ファイルシステムを作成するときにデータ圧縮を有効にするには (コンソール)

1. https://console.aws.amazon.com/fsx/ で Amazon FSx コンソールを開きます。
2. 開始方法 セクションの「ステップ 1: FSx for Lustre ファイルシステムの作成」で説明されている新しいファイルシステムを作成する手順に従います。
3. [File·system·details] (ファイルシステムの詳細) セクションの [Data compression type] (データ圧縮タイプ) で、LZ4 を選択します。
4. 新しいファイルシステムを作成する場合と同様に、ウィザードを完了します。
5. [Review and create] (レビューと作成) を選択します。
6. Amazon FSx for Lustre ファイルシステム用に選択した設定を確認し、ファイルシステムの作成を選択します。

ファイルシステムが 使用可能 の場合は、データ圧縮が有効になっています。

ファイルシステム (CLI) の作成時にデータ圧縮を有効にするには

- データ圧縮を有効にして FSx for Lustre ファイルシステムを作成するには、次のような `DataCompressionType` パラメータを指定して Amazon FSx CLI コマンド `create-file-system` を使用します。対応する API オペレーションは CreateFileSystem です。

```
$ aws fsx create-file-system \
      --client-request-token CRT1234 \
      --file-system-type LUSTRE \
      --file-system-type-version 2.12 \
      --lustre-configuration
 DeploymentType=PERSISTENT_1,PerUnitStorageThroughput=50,DataCompressionType=LZ4 \
      --storage-capacity 3600 \
      --subnet-ids subnet-123456 \
      --tags Key=Name,Value=Lustre-TEST-1 \
      --region us-east-2
```

次の例に示すように、ファイルシステムを正常に作成すると、Amazon FSx はファイルシステムの説明を JSON として返します。


```
{

    "FileSystems": [
        {
            "OwnerId": "111122223333",
            "CreationTime": 1549310341.483,
            "FileSystemId": "fs-0123456789abcdef0",
            "FileSystemType": "LUSTRE",
            "FileSystemTypeVersion": "2.12",
            "Lifecycle": "CREATING",
            "StorageCapacity": 3600,
            "VpcId": "vpc-123456",
            "SubnetIds": [
                "subnet-123456"
            ],
            "NetworkInterfaceIds": [
                "eni-039fcf55123456789"
            ],
            "DNSName": "fs-0123456789abcdef0.fsx.us-east-2.amazonaws.com",
            "ResourceARN": "arn:aws:fsx:us-east-2:123456:file-system/
fs-0123456789abcdef0",
            "Tags": [
                {
                    "Key": "Name",
                    "Value": "Lustre-TEST-1"
                }
            ],
            "LustreConfiguration": {
                "DeploymentType": "PERSISTENT_1",
                "DataCompressionType": "LZ4",
                "PerUnitStorageThroughput": 50
            }
        }
    ]
}
```


既存のファイルシステムのデータ圧縮設定を変更することもできます。既存のファイルシステムに対してデータ圧縮をオンにすると、新しく書き込まれたファイルのみが圧縮され、既存のファイルは圧縮されません。詳細については、「以前に書き込まれたファイルの圧縮」を参照してください。

既存のファイルシステムでのデータ圧縮を更新するには (コンソール)

1. https://console.aws.amazon.com/fsx/ で Amazon FSx コンソールを開きます。
2. [ファイルシステム] に移動し、データ圧縮を管理する Lustre ファイルシステムを選択します。
3. アクション で データ圧縮タイプの更新 を選択します。
4. データ圧縮タイプの更新 ダイアログボックスで、LZ4 を選択してデータ圧縮をオンにするか、[NONE] (なし) を選択してオフにします。
5. [Update] (更新) を選択します。
6. [Updates] (更新) タブのファイルシステムの詳細ページで更新の進行状況をモニタリングできます。

既存のファイルシステム (CLI) のデータ圧縮を更新するには

Lustre ファイルシステム用の既存の FSx のデータ圧縮設定を更新するには、AWS CLI コマンド update-file-system を使用します。以下のパラメータを設定します:

- `--file-system-id` を更新するファイルシステムの ID に設定します。
- データ圧縮をオフにするには `--lustre-configuration DataCompressionType` を `NONE` に設定し、LZ4 アルゴリズムでデータ圧縮をオンにするには `LZ4` を設定します。

このコマンドは、LZ4 アルゴリズムでデータ圧縮がオンになっていることを指定します。

```
$ aws fsx update-file-system \
    --file-system-id fs-0123456789abcdef0 \
    --lustre-configuration DataCompressionType=LZ4
```

バックアップからファイルシステムを作成するときのデータ圧縮設定

使用可能なバックアップを使用して、新しい Amazon FSx for Lustre ファイルシステムを作成できます。バックアップから新しいファイルシステムを作成する場合、`DataCompressionType` を指定する必要はありません。設定は、バックアップの `DataCompressionType` 設定を使用して適用されます。バックアップから作成するときに `DataCompressionType` を指定する場合、値はバックアップの `DataCompressionType` 設定と一致する必要があります。

バックアップの設定を表示するには、Amazon FSx コンソールの [Backups] (バックアップ) タブを選択します。バックアップの詳細は、バックアップの [Summary] (概要) ページに表示されま

す。describe-backups AWS CLI コマンドを実行することもできます (同等の API アクションは DescribeBackups)。

以前に書き込まれたファイルの圧縮

Amazon FSx for Lustre ファイルシステムでデータ圧縮が無効になったときに作成されたファイルは、圧縮解除されます。データ圧縮を有効にしても、既存の非圧縮データは自動的に圧縮されません。

Lustre クライアントのインストールの一部としてインストールされる `lfs_migrate` コマンドを使用して、既存のファイルを圧縮することができます。この例については、GitHub で入手可能な「FSxl 圧縮」を参照してください。

ファイルサイズの表示

次のコマンドを使用して、ファイルとディレクトリの非圧縮サイズと圧縮サイズを表示できます。

- du 圧縮サイズを表示します。
- `du --apparent-size` 非圧縮サイズを表示します。
- `ls -l` 非圧縮サイズを表示します。

次の例では、同じファイルでの各コマンドの出力を示します。

```
$ du -sh samplefile
272M samplefile
$ du -sh --apparent-size samplefile
1.0G samplefile
$ ls -lh samplefile
-rw-r--r-- 1 root root 1.0G May 10 21:16 samplefile
```

-h オプションは、人間が読める形式でサイズを出力するため、コマンドに役立ちます。

CloudWatch メトリクスの使用

Amazon CloudWatch Logs メトリクスを使用して、ファイルシステムの使用状況を表示できます。`LogicalDiskUsage` メトリクスは、論理的なディスクの総使用量 (圧縮なし) を示し、`PhysicalDiskUsage` メトリクスは、物理ディスクの総使用量 (圧縮あり) を示します。これら 2 つのメトリクスは、ファイルシステムでデータ圧縮が有効になっているか、以前に有効にしていた場合にのみ使用できます。

LogicalDiskUsage 統計の Sum を PhysicalDiskUsage 統計の Sum で割ることにより、ファイルシステムの圧縮率を決定できます。

ファイルシステムのパフォーマンスのモニタリングの詳細については、「Amazon FSx for Lustre ファイルシステムのモニタリング」を参照してください。

Lustre ルートスカッシュ

ルートスカッシュは、既存のネットワークベースのアクセス制御と POSIX ファイルに対するアクセス許可の上に、ファイルアクセス制御の新たなレイヤーを追加する管理機能です。ルートスカッシュ機能を使用すると、ルートとして FSx for Lustre ファイルシステムへのアクセスを試みるクライアントに対し、ルートレベルのアクセスを制限することができます。

FSx for Lustre のファイルシステムにおけるアクセス許可の管理など、管理アクションを実行するには、ルートユーザーとしてのアクセス許可が必要です。ただし、ルートアクセスでは、ユーザーに無制限のアクセス権を付与されます。また、ファイルシステムオブジェクトへのアクセス、変更、または削除に関するパーミッションチェックを、バイパスすることも可能になります。ルートスカッシュ機能を使用すると、ファイルシステムに対して非ルートのユーザー ID (UID) とグループ ID (GID) を指定することで、データの不正なアクセスや削除を防ぐことができます。ファイルシステムにアクセスするルートユーザーは、低い権限が指定されたユーザーやグループに自動的に変換され、ストレージ管理者が設定する制限付きの権限を使用します。

ルートスカッシュ機能では、ルートスカッシュ設定の影響を受けないクライアントのリストを、オプションで設定することもできます。これらのクライアントは、権限に制限なく、ルートとしてファイルシステムにアクセスできます。

トピック

- ルートスカッシュの仕組み
- ルートスカッシュの管理

ルートスカッシュの仕組み

ルートスカッシュ機能は、ルートユーザーのユーザー ID (UID) とグループ ID (GID) を、Lustre システム管理者が指定した UID と GID に再マッピングすることで機能します。ルートスカッシュ機能では、UID/GID の再マッピングが適用されないクライアントのセットを、オプションで指定することも可能です。

新しく作成した FSx for Lustre ファイルシステムでは、デフォルトでルートスカッシュが無効化されています。ルートスカッシュを有効にするには、FSx for Lustre ファイルシステムに対し、UID および GID のルートスカッシュ設定を行います。UID と GID の値は、0 ~ 4294967294 整数です。

- UID と GID にゼロ以外の値を指定すると、ルートスカッシュが有効化されます。UID と GID を異なる値にすることは可能ですが、ともにゼロ以外の値を設定する必要があります。
- UID と GID の値が 0 (ゼロ) の場合はルートを表します。この場合、ルートスカッシュは無効化されます。

ファイルシステムの作成時に、Amazon FSx コンソールを使用して、ファイルシステムの作成時にルートスカッシュを有効にするには (コンソール) に示すように、[ルートスカッシュ] プロパティにルートスカッシュ UID および GID 値を指定できます。ファイルシステムの作成時にルートスカッシュを有効にするには (CLI) に示すように、AWS CLI または API で RootSquash パラメータを使用して UID 値と GID 値を指定することもできます。

オプションで、ルートスカッシュが適用されないクライアントのために、NID のリストを指定することもできます。クライアントの NID は、Lustre Network でクライアントを一意に識別するために使用される識別子です。NID は、単一のアドレスとして指定する、またはアドレスの範囲として指定することができます。

- 単一のアドレスは、クライアントの IP アドレスに続いて Lustre のネットワーク ID を指定する、Lustre NID の標準的な形式で記述します (例: 10.0.1.6@tcp)。
- アドレス範囲は、範囲の区切りにダッシュを使用して記述します (例えば 10.0.[2-10].[1-255]@tcp)。
- クライアントの NID を指定しない場合、ルートスカッシュに対する例外は設定されません。

ファイルシステムを作成または更新する際に、Amazon FSx コンソールで [ルートスカッシュの例外] プロパティを使用して、クライアントの NID のリストを指定できます。AWS CLI または API で、NoSquashNids パラメータを使用します。詳細については、「ルートスカッシュの管理」の手順を参照してください。

ルートスカッシュの管理

ファイルシステムの作成中、デフォルトでは、ルートスカッシュは無効になっています。Amazon FSx コンソール、AWS CLI、または API を使用して新しい Amazon FSx for Lustre ファイルシステムを作成する際に、ルートスカッシュを有効にできます。

ファイルシステムの作成時にルートスカッシュを有効にするには (コンソール)

1. https://console.aws.amazon.com/fsx/ で Amazon FSx コンソールを開きます。
2. 開始方法 セクションの「ステップ 1: FSx for Lustre ファイルシステムの作成」で説明されている新しいファイルシステムを作成する手順に従います。
3. [ルートスカッシュ - オプション]セクションを開きます。
4. [ルートスカッシュ] には、ルートユーザーがファイルシステムにアクセスできるユーザー ID とグループ ID を指定します。1 ~ 4294967294 の範囲の任意の整数を次のように指定できます。

   1. [ユーザー ID] には、ルートユーザーが使用するユーザー ID を指定します。
   2. [グループ ID] には、ルートユーザーが使用するグループ ID を指定します。
5. (オプション) [ルートスカッシュの例外] については、次のようにします:

   1. [クライアントのアドレスを追加] を選択します。
   2. [クライアントのアドレス] フィールドに、ルートスカッシュが適用されないクライアントの IP アドレスを指定します。IP アドレスの形式については、「ルートスカッシュの仕組み」を参照してください。
   3. 必要に応じて繰り返し、クライアントの IP アドレスをさらに追加します。
6. 新しいファイルシステムを作成する場合と同様に、ウィザードを完了します。
7. [Review and create] (レビューと作成) を選択します。
8. Amazon FSx for Lustre ファイルシステム用に選択した設定を確認し、ファイルシステムの作成を選択します。

ファイルシステムが [利用可能] になると、ルートスカッシュが有効になります。

ファイルシステムの作成時にルートスカッシュを有効にするには (CLI)

- ルートスカッシュが有効化された FSx for Lustre ファイルシステムを作成するには、`RootSquashConfiguration` パラメータを指定しながら Amazon FSx CLI コマンド `create-file-system` を使用します。対応する API オペレーションは CreateFileSystem です。

  `RootSquashConfiguration` パラメータでは、以下のオプションを設定します。

  - `RootSquash` – ルートユーザーが使用するためのユーザー ID とグループ ID を指定する、コロンで区切られた値 (UID:GID) です。`0` ~ `4294967294` の範囲内であれば、それぞれの ID のために任意の整数 (0 はルート) を指定できます (例えば `65534:65534`)。

- NoSquashNids – ルートスカッシュが適用されないクライアントの Lustre Network 識別子 (NID) を指定します。クライアントの NID の形式については、「ルートスカッシュの仕組み」を参照してください。

次の例では、ルートスカッシュが有効化された FSx for Lustre ファイルシステムを作成しています。

```
$ aws fsx create-file-system \
      --client-request-token CRT1234 \
      --file-system-type LUSTRE \
      --file-system-type-version 2.15 \
      --lustre-configuration
 "DeploymentType=PERSISTENT_2,PerUnitStorageThroughput=250,DataCompressionType=LZ4,
\
          RootSquashConfiguration={RootSquash="65534:65534",\
          NoSquashNids=["10.216.123.47@tcp", "10.216.12.176@tcp"]}" \
      --storage-capacity 2400 \
      --subnet-ids subnet-123456 \
      --tags Key=Name,Value=Lustre-TEST-1 \
      --region us-east-2
```

次の例に示すように、ファイルシステムを正常に作成すると、Amazon FSx はファイルシステムの説明を JSON として返します。

```
{

    "FileSystems": [
        {
            "OwnerId": "111122223333",
            "CreationTime": 1549310341.483,
            "FileSystemId": "fs-0123456789abcdef0",
            "FileSystemType": "LUSTRE",
            "FileSystemTypeVersion": "2.15",
            "Lifecycle": "CREATING",
            "StorageCapacity": 2400,
            "VpcId": "vpc-123456",
            "SubnetIds": [
                "subnet-123456"
            ],
            "NetworkInterfaceIds": [
```

```
                "eni-039fcf55123456789"
            ],
            "DNSName": "fs-0123456789abcdef0.fsx.us-east-2.amazonaws.com",
            "ResourceARN": "arn:aws:fsx:us-east-2:123456:file-system/
fs-0123456789abcdef0",
            "Tags": [
                {
                    "Key": "Name",
                    "Value": "Lustre-TEST-1"
                }
            ],
            "LustreConfiguration": {
                "DeploymentType": "PERSISTENT_2",
                "DataCompressionType": "LZ4",
                "PerUnitStorageThroughput": 250,
                "RootSquashConfiguration": {
                    "RootSquash": "65534:65534",
                    "NoSquashNids": "10.216.123.47@tcp 10.216.29.176@tcp"
            }
        }
    ]
}
```

また、Amazon FSx コンソール、AWS CLI、または API を使用して、既存のファイルシステムのルートスカッシュ設定を更新することもできます。例えば、ルートスカッシュの UID と GID の値を変更したり、クライアントの NID を追加または削除したり、ルートスカッシュを無効化したりできます。

既存のファイルシステムでルートスカッシュの設定を更新するには (コンソール)

1. https://console.aws.amazon.com/fsx/ で Amazon FSx コンソールを開きます。
2. [ファイルシステム] に移動し、ルートスカッシュ管理の対象にする Lustre ファイルシステムを選択します。
3. [アクション] で [ルートスカッシュを更新] を選択します。または、[概要] パネルで、ファイルシステムの [ルートスカッシュ] フィールドの横にある [更新] を選択して、[ルートスカッシュ設定を更新] ダイアログボックスを表示します。
4. [ルートスカッシュ] では、ルートユーザーがファイルシステムにアクセスできるユーザー ID とグループ ID を更新します。`0` ~ `4294967294` の範囲の任意の整数を指定できます。ルートスカッシュを無効にするには、両方の ID に `0` (ゼロ) を指定します。

   1. [ユーザー ID] には、ルートユーザーが使用するユーザー ID を指定します。

2. [グループ ID] には、ルートユーザーが使用するグループ ID を指定します。

5. [ルートスカッシュの例外] では、次の操作を行います:

   1. [クライアントのアドレスを追加] を選択します。
   2. [クライアントのアドレス] フィールドに、ルートスカッシュが適用されないクライアントの IP アドレスを指定します。
   3. 必要に応じて繰り返し、クライアントの IP アドレスをさらに追加します。

6. [更新] を選択します。

> **Note**
>
> ルートスカッシュが有効になっていて無効にする場合は、ステップ 4～6 を実行しないで [無効化] を選択します。

[Updates] (更新) タブのファイルシステムの詳細ページで更新の進行状況をモニタリングできます。

既存のファイルシステムでルートスカッシュの設定を更新するには (CLI)

既存の FSx for Lustre ファイルシステムで、ルートスカッシュの設定を更新するには、AWS CLI コマンド update-file-system を使用します。API オペレーションでは、UpdateFileSystem がこれに相当します。

以下のパラメータを設定します:

- `--file-system-id` を更新するファイルシステムの ID に設定します。
- 以下のように `--lustre-configuration RootSquashConfiguration` オプションを設定します。
  - `RootSquash` – ルートユーザーが使用するためのユーザー ID とグループ ID を、コロンで区切った値 (UID:GID) で指定します。それぞれの ID には、`0`～`4294967294` の範囲内であれば任意の整数 (0 はルート) を指定できます。ルートスカッシュを無効にするには、UID:GID の値に `0:0` を指定します。
  - `NoSquashNids` – ルートスカッシュが適用されないクライアントの Lustre Network 識別子 (NID) を指定します。`[]` を使用すると、すべてのクライアント NID が削除されます。この場合、ルートスカッシュの例外は設定されません。

このコマンドでは、65534 をルートユーザーのユーザー ID とグループ ID の値として使用することで、ルートスカッシュの有効化を指定しています。

```
$ aws fsx update-file-system \
    --file-system-id fs-0123456789abcdef0 \
    --lustre-configuration RootSquashConfiguration={RootSquash="65534:65534", \
          NoSquashNids=["10.216.123.47@tcp", "10.216.12.176@tcp"]}
```

コマンドが正常に実行されると、Amazon FSx for Lustre は JSON 形式でレスポンスを返します。

ファイルシステムのルートスカッシュ設定は、Amazon FSx コンソールの [ファイルシステムの詳細] ページの [概要] パネル、または `describe-file-systems` CLI コマンド (同等の API アクションは `DescribeFileSystems`) の応答で確認できます。

FSx for Lustre ファイルシステムのステータス

Amazon FSx ファイルシステムのステータスを表示するには、Amazon FSx コンソールの AWS CLI コマンド describe-file-systems または API オペレーション DescribeFileSystems を使用します。

ファイルシステムのステータス	説明
AVAILABLE (利用可能)	ファイルシステムは正常な状態にあり、到達可能であり、使用可能です。
CREATING (作成)	Amazon FSx は新しいファイルシステムを作成しています。
[DELETING] (削除中)	Amazon FSx は既存のファイルシステムを削除しています。
UPDATING (更新)	ファイルシステムは、お客様によって開始される更新を受けています。
MISCONFIGURED (設定ミス)	ファイルシステムに障害が発生していますが、リカバリ可能な状態です。
FAILED (失敗)	このステータスは、次のいずれかを意味します。

ファイルシステムのステータス	説明
	• ファイルシステムに障害が発生したため、Amazon FSx では復旧できません。 • 新しいファイルシステムを作成するとき、Amazon FSx はファイルシステムを作成できませんでした。

Amazon FSx for Lustre リソースのタグ付け

ファイルシステムや Amazon FSx for Lustre リソースを管理しやすくするために、タグ形式で各リソースに独自のメタデータを割り当てることができます。タグを使用すると、例えば用途別、所有者別、環境別などのさまざまな方法で AWS リソースを分類できます。これは同じタイプのリソースが多数ある場合に役立ちます。割り当てたタグに基づいて、特定のリソースをすばやく識別できます。このトピックでは、タグとその作成方法について説明します。

トピック

タグの基本

タグとはAWS リソースに割り当てるラベルです。タグはそれぞれ、1 つのキーとオプションの 1 つの値で設定されており、どちらもお客様側が定義します。

タグを使用すると、例えば用途別、所有者別、環境別などのさまざまな方法で AWS リソースを分類できます。例えば、アカウントの Amazon FSx for Lustre ファイルシステムに一連のタグを定義して、各インスタンスの所有者とスタックレベルを追跡しやすくすることができます。

各リソースタイプのニーズを満たす一連のタグキーを考案することをお勧めします。一貫性のある一連のタグキーを使用することで、リソースの管理が容易になります。追加したタグに基づいてリソースを検索およびフィルタリングできます。

タグは Amazon FSx に対してセマンティックな意味は持たず、文字列として厳密に解釈されます。また、タグは自動的にリソースに割り当てられます。タグのキーと値は編集でき、タグはリソースか

らいつでも削除できます。タグの値を空の文字列に設定することはできますが、タグの値を null に設定することはできません。特定のリソースについて既存のタグと同じキーを持つタグを追加した場合、以前の値は新しい値によって上書きされます。リソースを削除すると、リソースのタグも削除されます。

Amazon FSx for Lustre API、AWS CLI、または AWS SDK を使用している場合、TagResource API アクションを使用してタグを既存のリソースに適用できます。さらに、リソース作成アクションによっては、リソースの作成時にリソースのタグを指定できます。リソースの作成時にタグを適用できない場合はリソース作成プロセスがロールバックされます。これにより、リソースがタグ付きで作成されるか、まったく作成されないようになるため、タグ付けされていないリソースが存在することがなくなります。作成時にリソースにタグ付けすることで、リソース作成後にカスタムタグ付けスクリプティングを実行する必要がなくなります。作成時にユーザーがリソースにタグ付けできるようにする方法については、「作成中にリソースにタグを付ける許可を付与する」を参照してください。

リソースのタグ付け

アカウントに存在する Amazon FSx for Lustre リソースにタグ付けできます。Amazon FSx コンソールを使用している場合は、関連するリソース画面のタグタブを使用して、リソースにタグを適用できます。リソースを作成するときは、Name キーに値を適用できます。また、新しいファイルシステムを作成するときに、選択したタグを適用できます。コンソールではリソースを Name タグに応じて整理できますが、このタグには Amazon FSx for Lustre サービスに対する意味論的意味はありません。

IAM ポリシーでタグベースのリソースレベルアクセス許可を、作成時のタグ付けをサポートする Amazon FSx for Lustre API アクションに適用し、作成時にリソースにタグ付けできるユーザーとグループを細かくコントロールできます。リソースは、作成時から適切に保護されます。タグはリソースに即座に適用されるため、リソースの使用をコントロールするタグベースのリソースレベルアクセスコントロールがただちに有効になります。リソースはより正確に追跡および報告されます。新しいリソースにタグ付けの使用を適用し、リソースで設定されるタグキーと値をコントロールできます。

さらに、リソースレベルのアクセス許可を IAM ポリシーの TagResource および UntagResource Amazon FSx for Lustre API アクションに適用し、既存のリソースで設定されるタグキーと値をコントロールすることもできます。

請求用リソースへのタグ付けの詳細については、「AWS Billing ユーザーガイド」の「コスト割り当てタグの使用」を参照してください。

タグの制限

タグには以下のような基本制限があります。

- リソースあたりのタグの最大数 - 50 件
- タグキーはリソースごとにそれぞれ一意である必要があります。また、各タグキーに設定できる値は 1 つのみです。
- キーの最大長 - UTF-8 の 128 Unicode 文字
- 値の最大長 - UTF-8 の 256 Unicode 文字
- Amazon FSx for Lustre のタグに使用できる文字は、UTF-8 で表現できる文字、数字、およびスペースに加えて、 + - = . _ : / @ です。
- タグのキーと値は大文字と小文字が区別されます。
- `aws:` プレフィックスは AWS 用に限定されています。タグにこのプレフィックスが付いたタグキーがある場合、タグのキーまたは値を編集、削除することはできません。`aws:` プレフィックスを持つタグは、リソースあたりのタグ数の制限にはカウントされません。

タグのみに基づいてリソースを削除することはできません。削除するには、リソース識別子を指定する必要があります。例えば、`DeleteMe` というタグキーでタグ付けされたファイルシステムを削除するには、fs-1234567890abcdef0 などのファイルシステムリソース識別子で `DeleteFileSystem` アクションを使用する必要があります。

公開リソースまたは共有リソースにタグを付ける場合、割り当てるタグは AWS アカウント でのみ使用できます。他の AWS アカウント はタグにアクセスできません。共有リソースへのタグベースのアクセスコントロールの場合、各 AWS アカウント は、リソースへのアクセスをコントロールするために独自のタグのセットを割り当てる必要があります。

許可とタグ

作成時に Amazon FSx リソースにタグ付けするために必要なアクセス許可の詳細については、「作成中にリソースにタグを付ける許可を付与する」を参照してください。また、タグを使用して IAM ポリシーで Amazon FSx リソースへのアクセスを制限する方法の詳細については、「タグを使用した Amazon FSx リソースへのアクセスのコントロール」を参照してください。

Amazon FSx for Lustre メンテナンスウィンドウ

Amazon FSx for Lustre は、管理する Lustre ソフトウェアに対して定期的なソフトウェアパッチを適用します。パッチ適用はまれで、通常は数週間に 1 回行われます。メンテナンスウィンドウは、このソフトウェアパッチが発生する曜日と時刻をコントロールする機会です。ファイルシステムの作成時にメンテナンスウィンドウを選択します。時間設定がない場合は、30 分のデフォルトウィンドウが割り当てられます。

パッチ適用には、30 分のメンテナンスウィンドウのほんの一部しか必要ありません。この数分間、ファイルシステムは一時的に使用できなくなります。ファイルシステムが利用できない間にクライアントから発行されたファイル操作は、自動的に再試行され、メンテナンス完了後に正常に完了します。メンテナンス中はインメモリキャッシュが消去されるため、メンテナンス完了後まではレイテンシが通常より高くなることに留意してください。

FSx for Lustre では、ワークロードと運用要件に合わせて、必要に応じてメンテナンスウィンドウを調整できます。メンテナンスウィンドウは、少なくとも 14 日ごとに 1 回スケジュールされていれば、必要に応じて何度でも移動できます。14 日以内にメンテナンス期間が設定されていない状態でパッチがリリースされた場合、FSx for Lustre は、セキュリティと信頼性を確保するためにファイルシステムのメンテナンスを続行します。

Amazon FSx 管理コンソール、AWS CLI、AWS API、またはいずれかの AWS SDK を使用して、ファイルシステムのメンテナンスウィンドウを変更できます。

コンソールを使用してメンテナンスウィンドウを変更するには

1. https://console.aws.amazon.com/fsx/ で Amazon FSx コンソールを開きます。
2. ナビゲーションペインで、[File systems] (ファイルシステム) を選択します。
3. メンテナンスウィンドウを変更するファイルシステムを選択します。ファイルシステムの詳細ページが表示されます。
4. [Maintenance] (メンテナンス) タブを選択します。メンテナンスウィンドウの [Settings] (設定) パネルが表示されます。
5. [Edit] (編集) をクリックし、メンテナンスウィンドウを開始する新しい日時を入力します。
6. [Save] (保存) を選択して変更を保存します。新しいメンテナンス開始時刻が [Settings] (設定) パネルに表示されます。

ファイルシステムのメンテナンスウィンドウは、update-file-system CLI コマンドを使用して変更できます。次のコマンドを実行し、ファイルシステム ID をユーザーのファイルシステムの ID に、日時をメンテナンス期間を開始する日時に置き換えます。

```
aws fsx update-file-system --file-system-id fs-01234567890123456 --lustre-configuration
 WeeklyMaintenanceStartTime=1:01:30
```

Lustre バージョンの管理

FSx for Lustre は現在、Lustre コミュニティによってリリースされた複数のロングタームサポート (LTS) Lustre バージョンをサポートしています。新しい LTS バージョンでは、クライアントインスタンス向けにパフォーマンス向上、新機能、および最新の Linux カーネルバージョンのサポートなどの利点が提供されます。AWS マネジメントコンソール、AWS CLI、または AWS SDK を使用して、数分以内にファイルシステムを新しい Lustre バージョンにアップグレードできます。

FSx for Lustre は現在、Lustre LTS バージョン 2.10、2.12、2.15 をサポートしています。FSx for Lustre ファイルシステムの LTS バージョンは、AWS マネジメントコンソール または describe-file-systems AWS CLI コマンドを使用して決定できます。

Lustre バージョンアップグレードを行う前に、「Lustre バージョンアップグレードのベストプラクティス」で説明されているステップを実行することをお勧めします。

トピック

Lustre バージョンアップグレードのベストプラクティス

FSx for Lustre ファイルシステムの Lustre バージョンをアップグレードする前に、以下のベストプラクティスに従うことをお勧めします:

- 非本番稼働環境でテストする: 本番稼働用ファイルシステムをアップグレードする前に、本番稼働用ファイルシステムの複製で Lustre バージョンアップグレードをテストします。これにより、本稼働ワークロードに対してスムーズなアップグレードプロセスが確保されます。
- クライアントの互換性を確認する: クライアントインスタンスで実行されている Linux カーネルバージョンが、アップグレード先の Lustre バージョンと互換性があることを確認します。詳細については、「Lustre ファイルシステムとクライアントカーネルの互換性」を参照してください。
- データをバックアップします。
  - S3 にリンクされていないファイルシステムの場合: Lustre バージョンをアップグレードする前に FSx バックアップを作成して、ファイルシステムの既知の復元ポイントを確保することをお勧めします。ファイルシステムで日次自動バックアップが有効になっている場合、Amazon FSx はアップグレード前にファイルシステムのバックアップを自動的に作成します。
  - S3 にリンクされたファイルシステムの場合 アップグレードする前に、すべての変更が S3 にエクスポートされていることを確認することをお勧めします。自動エクスポートを有効にしてい

る場合は、AgeOfOldestQueuedMessage AutoExport メトリクスがゼロであることを確認することで、すべての変更が正常に S3 にエクスポートされたことを確認できます。自動エクスポートを有効にしていない場合は、手動データリポジトリタスク (DRT) エクスポートを実行して、アップグレード前にファイルシステムを S3 バケットと同期できます。

アップグレードの実行

FSx for Lustre ファイルシステムを新しいバージョンにアップグレードするには、以下の手順に従ってください:

1. すべてのクライアントをアンマウントする: アップグレードを開始する前に、ファイルシステムにアクセスするすべてのクライアントインスタンスからファイルシステムをアンマウントする必要があります。Amazon CloudWatch の ClientConnections メトリクスを使用して、すべてのクライアントが正常にアンマウントされたことを確認できます。このメトリクスにはゼロ接続が表示されます。ファイルシステムに接続中のクライアントが存在する場合、アップグレードプロセスは進行しません。

   ファイルシステムのルートに保存されている .fsx/clientConnectionsファイル内の、ファイルシステムに接続されたクライアントネットワーク識別子 (NID) のリストを表示することができます。このファイルは 5 分ごとに更新されます。この例のように、cat コマンドを使用してファイルの内容を表示できます:

   ```
   cat /test/.fsx/clientConnections
   ```

2. Lustre バージョンのアップグレード: Amazon FSx コンソール、AWS CLI、または Amazon FSx API を使用して、FSx for Lustre ファイルシステムの Lustre バージョンをアップグレードできます。ファイルシステムを FSx for Lustre でサポートされている最新の Lustre バージョンにアップグレードすることをお勧めします。

   ファイルシステムの Lustre バージョンを更新するには (コンソール)

   a. https://console.aws.amazon.com/fsx/ で Amazon FSx コンソールを開きます。

   b. 左のナビゲーションペインで [File system] (ファイルシステム) を選択します。[ファイルシステム] リストで、Lustre バージョンを更新する FSx for Lustre ファイルシステムを選択します。

   c. [アクション] で、[ファイルシステムの更新] を選択します。または [概要] パネルで、ファイルシステムの [Lustre バージョン] フィールドの横にある [更新] を選択します。ファイル

システム Lustre バージョンを更新 ダイアログボックスが表示されます。ファイルシステム Lustre バージョンを更新 ダイアログボックスが表示されます。

d. 新しい Lustre バージョンの選択 フィールドで、Lustre バージョンを選択します。選択する値は、現在の Lustre バージョンよりも新しい必要があります。

e. [更新] を選択します。

ファイルシステムの Lustre バージョンを更新するには (CLI)

FSx for Lustre ファイルシステムの Lustre バージョンを更新するには、 AWS CLI コマンド update-file-system を使用します。(同等の API アクションは UpdateFileSystem です)。以下のパラメータを設定します:

- `--file-system-id` を更新するファイルシステムの ID に設定します。
- 更新するファイル システムの Lustre の新しいバージョンに `--file-system-type-version` を設定します。

以下の例では、ファイルシステムの Lustre バージョンを 2.12 から 2.15 に更新します:

```
aws fsx update-file-system \
    --file-system-id fs-0123456789abcdef0 \
    --file-system-type-version "2.15"
```

3. すべてのクライアントをマウントする: Amazon FSx コンソールの [更新] タブまたは AWS CLI の `describe-file-systems` を使用して、Lustre バージョンの更新の進行状況を監視できます。Lustre バージョンのアップグレードステータスが `Completed` と表示されたら、クライアントインスタンスにファイルシステムを安全に再マウントし、ワークロードを再開できます。

ファイルシステムの削除

Amazon FSx for Lustre ファイルシステムは、Amazon FSx コンソール、AWS CLI、Amazon FSx API を使用して削除することができます。FSx for Lustre ファイルシステムを削除する前に、ファイルシステムが接続されているすべての Amazon EC2 インスタンスからアンマウントする必要があります。S3 にリンクされたファイルシステムでは、ファイルシステムを削除する前にすべてのデータが S3 に書き込まれていることを確認するために、AgeOfOldestQueuedMessage メトリクスがゼロになるまでモニタリングする (自動エクスポートを使用している場合) か、データリポジトリのエクスポートタスクを実行することができます。自動エクスポートを有効にしていて、データリポジトリ

のエクスポートタスクを使用する場合は、データリポジトリのエクスポートタスクを実行する前に自動エクスポートを無効にする必要があります。

すべての Amazon EC2 インスタンスからアンマウントした後にファイルシステムを削除するには

- コンソールの使用 - ステップ 5: リソースをクリーンアップするで説明されている手順に従います。
- API または CLI の使用 – DeleteFileSystem API オペレーションまたは delete-file-system CLI コマンドを使用します。

バックアップでデータを保護する。

Amazon FSx for Lustre を使用すると、Simple Storage Service (Amazon S3) 耐久データリポジトリにリンクされていない永続ファイルシステムの、自動日次バックアップとユーザー主導バックアップを実行できます。Amazon FSx バックアップは、ファイルシステムの一貫性、高い耐久性、および増分されます。高い耐久性を確保するために、Amazon FSx for Lustre では 99.999999999% (イレブンナイン) の耐久性で Amazon Simple Storage Service (Amazon S3) にバックアップを保存します。

FSx for Lustreファイルシステムのバックアップは、自動日次バックアップかユーザー主導のバックアップ機能を使用して生成されたものであるかを問わない、ブロックベースの増分バックアップです。つまり、バックアップを取得する際に、Amazon FSx はファイルシステム上のデータと以前のバックアップをブロックレベルで比較します。その後、Amazon FSx は、すべてのブロックレベルの変更のコピーを新しいバックアップに保存します。以前のバックアップが新しいバックアップに保存されないため、変更されないブロックレベルのデータ。バックアッププロセスの期間は、前回のバックアップが実行されてから変更されたデータの量によって異なり、ファイルシステムのストレージ容量の影響を受けません。次のリストは、さまざまな状況下でのバックアップ時間を示しています。

- データがほとんどない真新しいファイルシステムの初期バックアップは数分で完了します。
- TB のデータをロードした後に実行される新しいファイルシステムの初期バックアップは、完了までに数時間かかります。
- ブロックレベルのデータに対する最小限の変更 (作成 / 変更が比較的少ない) で、TB のデータを用いたファイルシステムの 2 回目のバックアップは、完了までに数秒かかります。
- 大量のデータが追加および変更された後、同じファイルシステムの 3 回目のバックアップが完了するまでに数時間かかります。

バックアップを削除すると、そのバックアップに固有のデータだけが削除されます。各 FSx for Lustre バックアップには、バックアップから新しいファイルシステムを作成するために必要なすべての情報が含まれており、ファイルシステムのポイントインタイムスナップショットを効果的に復元します。

ファイルシステムの定期的なバックアップを作成することは、Amazon FSx for Lustre がファイルシステムに対して実行するレプリケーションを補完するベストプラクティスです。Amazon FSx バックアップは、バックアップの保持とコンプライアンスのニーズのサポートに役立ちます。Amazon FSx for Lustre バックアップの使用は、バックアップの作成、バックアップのコピー、バックアップからのファイルシステムの復元、バックアップの削除など、簡単です。

スクラッチ ファイルシステムは、データのテンポラリストレージと短期間の処理用に設計されているため、バックアップはサポートされていません。S3 バケットがプライマリデータリポジトリとして機能し、Lustre ファイルシステムに常に完全なデータセットが含まれているとは限らないため、Amazon S3 バケットにリンクされたファイルシステムではバックアップはサポートされていません。

トピック

FSx for Lustreでのバックアップサポート

バックアップは、Simple Storage Service (Amazon S3) データリポジトリにリンクされていない FSx for Lustre 永続ファイルシステムでのみサポートされています。

スクラッチ ファイルシステムはテンポラリストレージと短期間のデータ処理用に設計されているため、Amazon FSx はスクラッチ ファイルシステムでのバックアップをサポートしていません。S3 バケットはプライマリデータリポジトリとして機能し、ファイルシステムには必ずしも常に完全なデータセットが含まれているとは限らないため、Amazon FSx は Simple Storage Service (Amazon S3) バケットにリンクされたファイルシステムでのバックアップをサポートしていません。詳細については、「デプロイとストレージクラスオプション」および「データリポジトリの使用」を参照してください。

自動日次バックアップの使用

Amazon FSx for Lustre は、ファイルシステムの自動日次バックアップを取ることができます。自動日次バックアップは、ファイルシステムの作成時に設定された日次バックアップウィンドウ中に実行されます。日次バックアップウィンドウのある時点で、バックアッププロセスが初期化される間、ストレージ I / O が一時的に中断することがあります (通常は数秒以下)。日次バックアップウィンドウ

を選択するときは、1日の中で都合の良い時間帯を選択することをお勧めします。この時間は、ファイルシステムを使用するアプリケーションの通常の動作時間外であることが理想的です。

自動日次バックアップは、保持期間 と呼ばれる一定期間、保存されます。保持期間は、0～90 日間で設定できます。保持期間を 0 (ゼロ) 日 に設定すると、自動日次 バックアップが行われなくなります。自動日次バックアップのデフォルトの保持期間は 0 日です。自動日次バックアップは、ファイルシステムの削除時に削除されます。

> **Note**
>
> 保持期間を 0 日に設定すると、ファイルシステムが自動的にバックアップされることはありません。関連したすべてのレベルの重要な機能を持つファイルシステムには、自動日次バックアップを使用することを強くお勧めします。

AWS CLI またはいずれかの AWS SDKs を使用して、ファイルシステムのバックアップウィンドウとバックアップ保持期間を変更できます。`UpdateFileSystem` API オペレーションまたは `update-file-system` CLI コマンドを使用します。

ユーザー主導のバックアップ機能

Amazon FSx for Lustre では、いつでもファイルシステムのバックアップを手動で作成できます。これを行うには、Amazon FSx for Lustre コンソール、API、または AWS Command Line Interface (CLI) を使用します。ユーザーが作成した Amazon FSx ファイルシステムのバックアップは期限切れにならず、保存したい期間利用できます。ユーザーによるバックアップは、バックアップされたファイルシステムを削除した後も保持されます。ユーザーが作成したバックアップは、Amazon FSx for Lustre コンソール、API、または CLI を使用してのみ削除でき、Amazon FSx によって自動的に削除されることはありません。詳細については、「バックアップの削除」を参照してください。

ユーザーによるバックアップの作成

次の手順では、既存のファイルシステムの Amazon FSx コンソールでユーザー主導のバックアップを作成する方法についてガイドします。

ユーザー主導のファイルシステムバックアップを作成するには

1. https://console.aws.amazon.com/fsx/ で Amazon FSx for Lustre コンソールを開きます。
2. コンソールダッシュボードから、バックアップするファイルシステムの名前を選択します。

3. [Actions] (アクション)から、[Create backup] (バックアップの作成) を選択します。
4. 開いた [Create backup] (バックアップの作成) ダイアログボックスで、バックアップの名前を入力します。バックアップ 名は、英字、空白、数字、特殊文字. + - = _ : / を含む最大 256 の Unicode 文字を使用できます。
5. [Create backup] (バックアップの作成) を選択します。

これで、ファイルシステムのバックアップが作成されました。Amazon FSx for Lustre コンソールの左側のナビゲーション [Backups] (バックアップバックアップ) を選択すると、すべてのバックアップのテーブルが表示されます。バックアップに付けた名前と、一致する結果のみを表示するようにテーブルフィルターを検索できます。

この手順で説明したように、ユーザー主導バックアップを作成すると、タイプは `USER_INITIATED` になり、Amazon FSx がバックアップを [Creating]] (作成中) している間は作成ステータスになります。完全に利用可能になるまで、バックアップが Simple Storage Service (Amazon S3) に転送されている間は、ステータスが [Transferring] (転送中) に変わります。

Amazon FSx AWS Backup での の使用

AWS Backup は、Amazon FSx ファイルシステムをバックアップしてデータを保護するシンプルで費用対効果の高い方法です。 AWS Backup は、作成を簡素化するために設計された統合バックアップサービスです。 コピー、 復元、 バックアップの削除、 レポートと監査を改善しながら、リーガル AWS Backup な 規制、 また、 AWS Backup は AWS ストレージボリュームを保護します。 データベース、 と ファイルシステムは、以下を実行できる一元的な場所を提供することで、よりシンプルになります。

- バックアップする AWS リソースを設定して監査します。
- バックアップスケジュールのオートメーション。
- 保持ポリシーの設定。
- AWS リージョン間および AWS アカウント間でバックアップをコピーします。
- 最近のすべてのバックアップと復元アクティビティのモニタリング。

AWS Backup は、Amazon FSx の組み込みバックアップ機能を使用します。 AWS Backup コンソールから取得されるバックアップは、Amazon FSx コンソールを介して取得されるバックアップと同じレベルのファイルシステムの整合性とパフォーマンス、および同じ復元オプションを持ちます。AWS Backup を使用してこれらのバックアップを管理すると、無制限の保持オプションや、1 時間ごとにスケジュールされたバックアップを作成する機能など、追加の機能を利用できます。さらに、

ソースファイルシステムが削除された後でも、 はイミュータブルバックアップ AWS Backup を保持します。これにより、偶発的または悪意のある削除から保護できます。

によって作成されたバックアップ AWS Backup は、バックアップタイプAWS_BACKUPを持ち、ファイルシステムで取得する他の Amazon FSx バックアップと比較して増分的です。によって作成されたバックアップ AWS Backup はユーザー主導のバックアップと見なされ、Amazon FSx のユーザー主導のバックアップクォータにカウントされます。によって作成されたバックアップは、Amazon FSx コンソール、CLI、および API AWS Backup で表示および復元できます。ただし、Amazon FSx コンソール、CLI、または API AWS Backup で によって作成されたバックアップを削除することはできません。 AWS Backup を使用して Amazon FSx ファイルシステムをバックアップする方法の詳細については、 AWS Backup デベロッパーガイドの「Amazon FSx ファイルシステムの使用」を参照してください。

バックアップのコピー

Amazon FSx を使用して、同じ AWS アカウント内のバックアップを別の AWS リージョン (クロスリージョンコピー) または同じ AWS リージョン (リージョン内コピー) に手動でコピーできます。クロスリージョンコピーは、同じ AWS パーティション内でのみ作成できます。ユーザー主導のバックアップコピーは、Amazon FSx コンソール AWS CLI、または API を使用して作成できます。ユーザー主導バックアップコピーを作成するときは、タイプ USER_INITIATED があります。

AWS Backup を使用して、 AWS リージョン AWS アカウント間でバックアップをコピーすることもできます。 AWS Backup は、ポリシーベースのバックアッププランの一元的なインターフェイスを提供するフルマネージドバックアップ管理サービスです。クロスアカウント管理では、バックアップポリシーを自動的に使用して、組織内の アカウント全体にバックアッププランを適用できます。

クロスリージョンバックアップコピー は、クロスリージョン災害対策に特に役立ちます。プライマリで災害が発生した場合にバックアップから復元し AWS リージョン、他の AWS リージョンですぐに可用性を回復できるように、バックアップを作成して別の AWS リージョンにコピーします。バックアップコピーを使用して、ファイルデータセットを別の AWS リージョン または同じ 内にクローンすることもできます AWS リージョン。Amazon FSx コンソール、または Amazon FSx for Lustre API を使用して AWS CLI、同じ AWS アカウント (クロスリージョンまたはインリージョン) 内にバックアップコピーを作成します。また、AWS Backup を使用して、オンデマンドまたはポリシーベースのバックアップコピーを実行することもできます。

クロスアカウントバックアップコピー は、独立したアカウントにバックアップをコピーするという規制コンプライアンスの要件を満たすのに役立ちます。また、バックアップの偶発的または悪意のある削除、認証情報の喪失、または AWS KMS キーの侵害を防ぐために、データ保護の追加レイヤー

も提供します。クロスアカウントバックアップは、ファンイン (複数のプライマリアカウントから 1 つの独立したバックアップ コピーアカウントにバックアップをコピーすること) および ファンアウト (1 つのプライマリアカウントから複数の独立したバックアップ コピーアカウントにバックアップをコピーすること) をサポートします。

サポート AWS Backup で を使用して、クロスアカウントバックアップコピーを作成できます AWS Organizations 。クロスアカウントコピーのアカウント境界は、 AWS Organizations ポリシーによって定義されます。 AWS Backup を使用してクロスアカウントバックアップコピーを作成する方法の詳細については、 AWS Backup デベロッパーガイドの「 でのバックアップコピーの作成 AWS アカウント」を参照してください。

バックアップコピーの制約

バックアップをコピーする際の制約は以下のとおりです。

- インテリジェント階層化 ストレージクラスを使用したファイルシステムのバックアップは、バックアップコピーをサポートしていません。
- クロスリージョンバックアップコピーは、2 つの商用リージョン間 AWS リージョン、中国 (北京) と中国 (寧夏) リージョン間、および AWS GovCloud (米国東部) と AWS GovCloud (米国西部) リージョン間でのみサポートされますが、これらのリージョンのセット間ではサポートされません。
- クロスリージョンバックアップコピーは、オプトインリージョンではサポートされていません。
- リージョン内のバックアップコピーは、任意の 内で作成できます AWS リージョン。
- コピーする前に、出典バックアップは、`AVAILABLE` のステータスである必要があります。
- コピー中の出典バックアップは削除できません。デスティネーション· バックアップが利用可能になってから、出典バックアップを削除できるようになるまでの間に、短い遅延が発生する場合があります。出典バックアップの削除を再試行する場合は、この遅延に注意する必要があります。
- アカウントあたり 1 つのコピー先 AWS リージョン に対して最大 5 つのバックアップコピーリクエストを実行できます。

クロスリージョンのバックアップコピーの許可

IAM ポリシーステートメントを使用して、バックアップコピーオペレーションを実行するためのアクセス許可を付与します。ソース AWS リージョンと通信してクロスリージョンバックアップコピーをリクエストするには、リクエスタ (IAM ロールまたは IAM ユーザー) がソースバックアップとソース AWS リージョンにアクセスできる必要があります。

ポリシーを使用して、バックアップコピーオペレーションの CopyBackup アクションにアクセス許可を付与します。次の例のように、ポリシーの Action フィールドでアクションを指定し、ポリシーの Resource フィールドでリソース値を指定します。

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "fsx:CopyBackup",
            "Resource": "arn:aws:fsx:*:111122223333:backup/*"
        }
    ]
}
```

IAM ポリシーの詳細については、IAM ユーザーガイド の「IAM ポリシーと許可」を参照してください。

フルコピーと増分コピー

バックアップをソースバックアップ AWS リージョン とは異なる にコピーする場合、最初のコピーはフルバックアップコピーです。最初のバックアップコピー後、同じ AWS アカウント内の同じコピー先リージョンへの後続のすべてのバックアップコピーは増分されます。ただし、そのリージョンで以前にコピーされたすべてのバックアップを削除しておらず、同じ AWS KMS キーを使用していることが条件です。両方の条件が満たされていない場合、コピーオペレーションによって (増分ではない) フルバックアップコピーになります。

同じ 内でバックアップをコピーする AWS アカウント

次の手順で説明するように AWS マネジメントコンソール、、CLI、および API を使用して FSx for Lustre ファイルシステムのバックアップをコピーできます。

コンソールを使用して、同じアカウント (クロスリージョンまたはインリージョン) 内のバックアップをコピーするには

1. https://console.aws.amazon.com/fsx/ で Amazon FSx コンソールを開きます。

2. ナビゲーションペインで、[Backups] (バックアップ) を選択します。
3. [Backups] (バックアップ) テーブルで、コピーするバックアップを選択し、[Copy backup] (バックアップのコピー)を選択します。
4. [Settings] (設定) セクションで、以下の手順を実行します:
   - 送信先リージョンリストで、バックアップをコピーする送信先 AWS リージョンを選択します。送信先は、別の AWS リージョン (クロスリージョンコピー) または同じ AWS リージョン内 (リージョン内コピー) にすることができます。
   - (オプション) [Copy Tags] (タグのコピー) を選択して、出典バックアップから宛先バックアップにタグをコピーします。ステップ 6 で [Copy Tags] (タグのコピー) を選択し、タグを追加すると、すべてのタグがマージされます。
5. 暗号化 では、コピーしたバックアップを暗号化する AWS KMS 暗号化キーを選択します。
6. [Tags - optional] (タグ - オプション) で、キーと値を入力して、コピーしたバックアップにタグを追加します。ここにタグを追加し、またステップ 4 で [Copy Tags] (タグのコピー) を選択すると、すべてのタグがマージされます。
7. [Copy backup] (バックアップのコピー) を選択します。

   バックアップは、同じ 内で選択した AWS アカウント にコピーされます AWS リージョン。

CLI を使用して同じアカウント内 (クロスリージョンまたはインリージョン) 内でバックアップをコピーするには

- copy-backup CLI コマンドまたは CopyBackup API オペレーションを使用して、 AWS リージョン間または AWS リージョン内で、同じ AWS アカウント内のバックアップをコピーします。

  次のコマンドは、us-east-1 リージョンからbackup-0abc123456789cba7 の ID でバックアップをコピーします。

  ```
  aws fsx copy-backup \
    --source-backup-id backup-0abc123456789cba7 \
    --source-region us-east-1
  ```

  レスポンスには、コピーされたバックアップの説明が表示されます。

Amazon FSx コンソールまたはプログラムで `describe-backups` CLI コマンドあるいは DescribeBackups (バックアップの説明) の API オペレーションを使用してバックアップを見ることができます。

バックアップの復元

可能なバックアップを使用して新しいファイルシステムを作成し、別のファイルシステムのポイントインタイム スナップショット を効果的に復元できます。コンソール AWS CLI、またはいずれかの AWS SDKs を使用してバックアップを復元できます。新しいファイルシステムへのバックアップの復元には、新しいファイルシステムの作成と同じ時間がかかります。バックアップから復元されたデータは、ファイルシステムにレイジーロードされ、その間、レイテンシーがわずかに長くなります。

> **Note**
>
> バックアップは、元のデプロイタイプ、ストレージクラス、スループットキャパシティ、ストレージキャパシティ、データ圧縮タイプ、および AWS リージョン と同じデプロイタイプのファイルシステムにのみ復元できます。復元されたファイルシステムのストレージ容量は、利用可能になった後、増やすことができます。

コンソールを使用してバックアップからファイルシステムを復元するには

1. https://console.aws.amazon.com/fsx/ で Amazon FSx for Lustre コンソールを開きます。
2. コンソールダッシュボードで、左側のナビゲーションから [Backups] (バックアップ) を選択します。
3. [Backups] (バックアップ) テーブルから復元するバックアップを選択し、[Restore backup] (バックアップの復元) を選択します。

   ファイルシステム作成ウィザードが開き、バックアップが作成されたファイルシステムの設定に基づいて、ほとんどの設定が事前に入力されています。オプションで、仮想プライベートクラウド (VPC) 設定を変更するか、新しい Lustre バージョンを選択できます。デプロイタイプやストレージ単位あたりのスループットなど、他の構成設定は復元中に変更できないことに注意してください。
4. 新しいファイルシステムを作成する場合と同様に、ウィザードを完了します。
5. [Review and create] (レビューと作成) を選択します。

6. Amazon FSx for Lustre ファイルシステム用に選んだ設定を確認し、[Create file system] (ファイルシステムの作成) を選択します。

バックアップから復元し、新しいファイルシステムを作成中です。ステータスが `AVAILABLE` に変わると、ファイルシステムを通常どおり使用できます。

バックアップの削除

バックアップの削除は、永久的で回復不能なアクションです。削除されたバックアップ内のデータもすべて削除されます。今後そのバックアップが必要でないということが確かでない限り、バックアップを削除しないでください。Amazon FSx コンソール、CLI、または API AWS Backup で によって作成されたバックアップを削除することはできません。

バックアップを削除するには

1. https://console.aws.amazon.com/fsx/ で Amazon FSx for Lustre コンソールを開きます。
2. コンソールダッシュボードで左側のナビゲーションから [Backups] (バックアップ) を選択します。
3. [Backups] (バックアップ) テーブルから削除するバックアップを選択してから、[Delete backup] (バックアップの 削除) を選択します。
4. 開いた [Delete backups] (バックアップの削除) ダイアログボックスで、バックアップの ID が削除するバックアップを識別していることを確認します。
5. 削除するバックアップのチェックボックスがチェックされていることを確認します。
6. [Delete backups] (バックアップの削除) を選択します。

これで、バックアップと含まれているすべてのデータが完全に復元不能に削除されます。

Amazon FSx for Lustre ファイルシステムのモニタリング

モニタリングは、FSx for Lustre ファイルシステムおよび他の AWS ソリューションの信頼性、可用性、パフォーマンスを維持する上で重要です。マルチポイント障害が発生した場合は、AWS ソリューションのすべての部分からモニタリングデータを収集することにより、その障害をより簡単にデバッグできます。FSx for Lustres ファイルシステムを監視して異常を検出した場合に報告し、必要に応じて、次のツールを使用して自動的に対処することができます。

- Amazon CloudWatch – AWS のリソースおよび AWS で実行しているアプリケーションをリアルタイムでモニタリングします。メトリクスの収集と追跡、カスタマイズしたダッシュボードの作成、および指定したメトリクスが指定したしきい値に達したときに通知するアラームの設定を行うことができます。例えば、CloudWatch で Amazon FSx for Lustre インスタンスのストレージ容量などのメトリクスを追跡し、必要に応じて新しいインスタンスを自動的に起動できます。
- Lustre のログ - ファイルシステムに対して有効になっているログイベントをモニタリングします。Lustre のログは、イベントを Amazon CloudWatch Logs に書き込みます。
- AWS CloudTrail - AWS アカウント により、またはそのアカウントに代わって行われた API コールおよび関連イベントを取得し、指定した Amazon S3 バケットにログファイルを配信します。AWS を呼び出したユーザーとアカウント、呼び出し元の IP アドレス、および呼び出し日時を特定できます。

以下のセクションでは、FSx for Lustre ファイルシステムでツールを使用する方法について説明します。

トピック

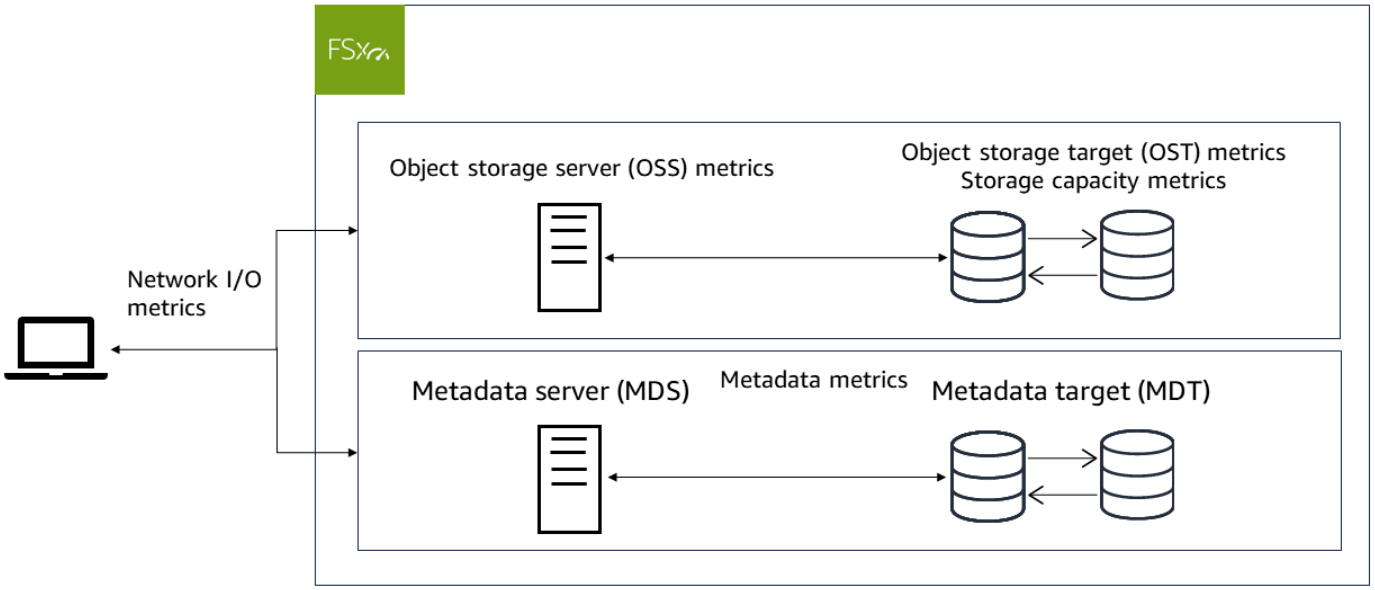
Amazon CloudWatch によるモニタリング

CloudWatch を使用して Amazon FSx for Lustre をモニターできます。CloudWatch は、Amazon FSx for Lustre から raw データを収集して処理し、読み取り可能なほぼリアルタイムのメトリクスにします。これらの統計は 15 か月間保持されるため、履歴情報にアクセスして、アプリケーションまたはサービスのパフォーマンスをより正確に把握できます。CloudWatch の詳細については、「Amazon CloudWatch ユーザーガイド」の「Amazon CloudWatch とは」を参照してください。

FSx for Lustre の CloudWatch メトリクスは、次の 6 つのカテゴリに分類されます:

- ネットワーク I/O メトリクス – クライアントとファイルシステム間のアクティビティを測定します。
- オブジェクトストレージサーバーのメトリクス – オブジェクトストレージサーバー (OSS) のネットワークスループットとディスクスループット使用率を測定します。
- オブジェクトストレージターゲットメトリクス – オブジェクトストレージターゲット (OST) のディスクスループットとディスク IOPS 使用率を測定します。
- メタデータメトリクス – メタデータサーバー (MDS) CPU 使用率、メタデータターゲット (MDT) IOPS 使用率、およびクライアントメタデータオペレーションを測定します。
- ストレージ容量メトリクス – ストレージ容量の使用率を測定します。
- S3 データリポジトリメトリクス – インポートまたはエクスポートを待っている最も古いメッセージの経過時間を測定し、ファイルシステムによって処理される名前を変更します。

次の図は、FSx for Lustre ファイルシステム、そのコンポーネント、およびメトリクスカテゴリを示しています。



FSx for Lustre は、メトリクスデータを 1 分間隔で CloudWatch に送信します。

> **Note**
>
> メトリクスは、Amazon FSx for Lustre ファイルシステムのファイルシステムメンテナンスウィンドウ中に公開されない場合があります。

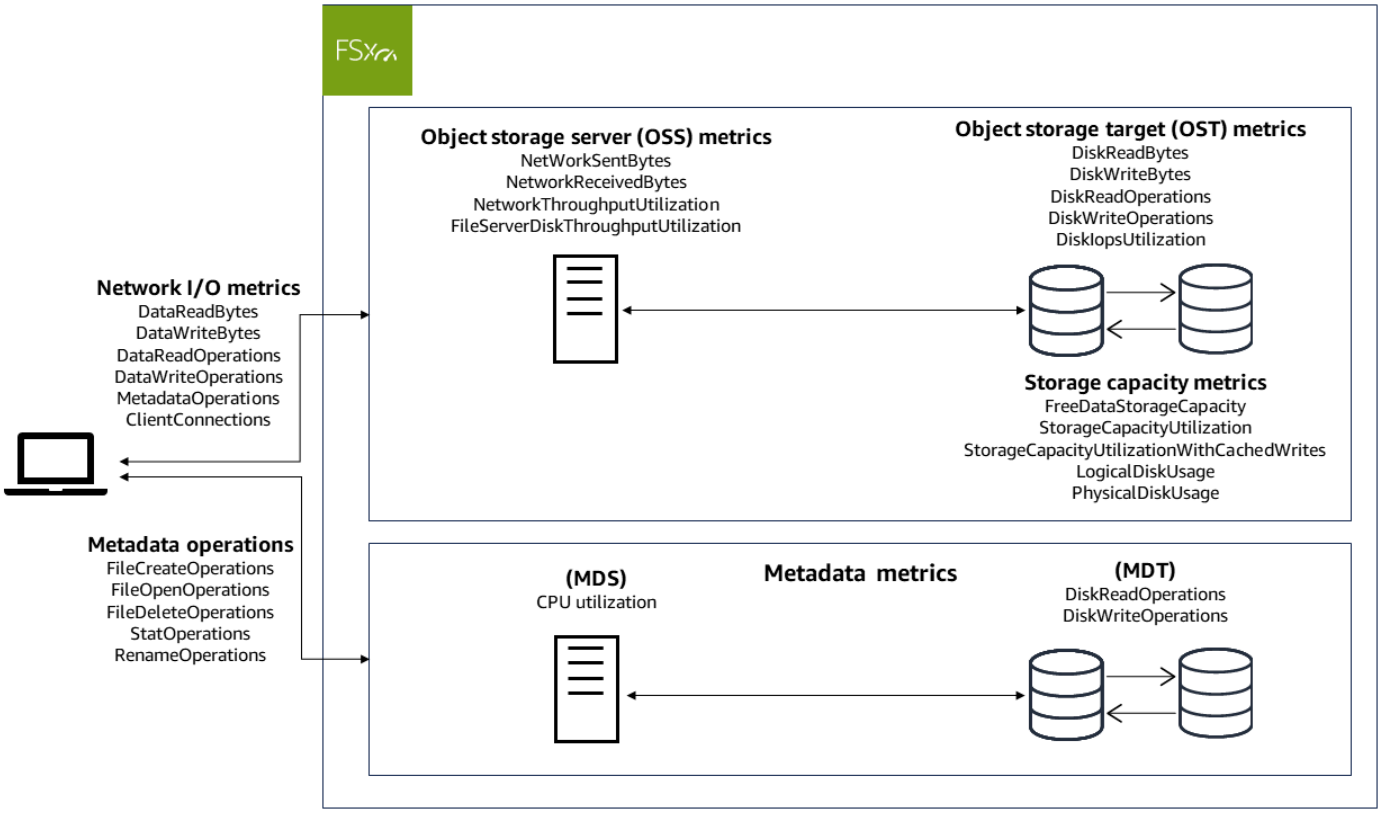
トピック

Amazon FSx for Lustre CloudWatch メトリクスを使用する方法

各 Amazon FSx for Lustre ファイルシステムには、2 つの主要なアーキテクチャコンポーネントがあります。

- ファイルシステムにアクセスするクライアントにデータを提供する 1 つ以上のオブジェクトストレージサーバー (OSSs)。各 OSS は、ファイルシステム内のデータをホストするオブジェクトストレージターゲット (OSTs) と呼ばれる 1 つ以上のストレージボリュームにアタッチされます。
- ファイルシステムにアクセスするクライアントにメタデータを提供する 1 つ以上のメタデータサーバー (MDSs)。各 MDS は、メタデータターゲット (MDT) と呼ばれるストレージボリュームにアタッチされ、ファイル名、ディレクトリ、アクセス許可、ファイルレイアウトなどのメタデータを格納します。

FSx for Lustre は、ファイルシステムのストレージとメタデータサーバーのパフォーマンスおよびリソース使用率を追跡するメトリクスを CloudWatch でレポートします。次の図は、Amazon FSx for Lustre ファイルシステムとそのアーキテクチャコンポーネント、およびモニタリングに使用できるパフォーマンスとリソースの CloudWatch メトリクスを示しています。



Amazon FSx for Lustre コンソールのファイルシステムのダッシュボードにある [モニタリングとパフォーマンス] パネルを使用して、以下の表に記載されているメトリクスを表示することができます。詳細については、「CloudWatch メトリクスへのアクセス」を参照してください。

ファイルシステムアクティビティ (概要タブ内)

方法を教えてください	チャート	関連するメトリクス
ファイルシステムで使用可能なストレージ容量を判別するにはどうすればよいですか?	使用可能なストレージ容量 (バイト)	`FreeDataStorageCapacity`
ファイルシステムの合計スループットはどのように決定すればよいですか?	合計クライアントスループット (バイト/秒)	`SUM(DataReadBytes + DataWriteBytes )/PERIOD` (秒単位)
ファイルシステムの合計クライアント IOPS を決定しますか?	合計クライアント	`SUM(DataReadOperations + DataWriteOperation`

方法を教えてください	チャート	関連するメトリクス
	IOPS (オペレーション/秒)	s + MetadataOperations)/PERIOD (in seconds)
クライアントとファイルサーバー間で確立されている接続の数を判別しますか?	クライアント接続 (数)	ClientConnections
ファイルシステムのメタデータパフォーマンス使用率を決定しますか?	メタデータ IOPS 使用率 (%)	MAX(MDT Disk IOPS)

ストレージタブ

方法を教えてください	チャート	関連するメトリクス
使用可能なストレージの容量を決定しますか?	使用可能なストレージ容量 (バイト)	FreeDataStorageCapacity
クライアントでのキャッシュされた書き込み用に予約されたスペースを除き、ファイルシステムの使用済みストレージの割合を決定しますか?	ストレージ容量の合計使用率 (%)	StorageCapacityUtilization
クライアントでのキャッシュされた書き込み用に予約されたスペースを含む、ファイルシステムの使用済みストレージの割合を決定しますか?	ストレージ容量の合計使用率 (%)	StorageCapacityUtilizationWithCachedWrites
クライアントでのキャッシュされた書き込み用に予約されたスペースを除き、ファイルシステムの OSTs の使用済みストレージの割合を決定しますか?	OST あたりのストレージ容量の合計使用率 (%)	StorageCapacityUtilization
クライアントへのキャッシュ書き込み用に予約されたスペースを含む、ファイルシステムの	クライアント許可による OST あ	StorageCapacityUtilizationWithCachedWrites

方法を教えてください	チャート	関連するメトリクス
OSTsの使用済みストレージの割合を決定しますか?	たりのストレージ容量の合計使用率 (%)	
ファイルシステムのデータ圧縮率を決定しますか?	圧縮設定	100*(LogicalDiskUsage - PhysicalDiskUSage)/LogicalDiskUsage

オブジェクトストレージのパフォーマンス (パフォーマンスタブ内)

方法を教えてください	チャート	関連するメトリクス
クライアントと OSS 間のネットワークスループットをプロビジョニングされた制限の割合として決定しますか?	ネットワークスループット (%)	NetworkThroughputUtilization
OSS とその OST 間のディスクスループットをプロビジョニングされた制限の割合として決定しますか?	ディスクスループット (%)	FileServerDiskThroughputUtilization
OST にアクセスするオペレーションの IOPS をプロビジョニングされた制限の割合として決定しますか?	ディスク IOPS (%)	DiskIopsUtilization

メタデータパフォーマンス (パフォーマンスタブ内)

方法を教えてください	チャート	関連するメトリクス
メタデータサーバーの CPU 使用率を決定しますか?	CPU 使用率 (%)	CPUUtilization
メタデータ IOPS 使用率をプロビジョニングされた制限の割合として決定しますか?	メタデータ IOPS 使用率	MAX(MDT Disk IOPS)

CloudWatch メトリクスへのアクセス

CloudWatch の Amazon FSx for Lustre メトリクスは、次の方法で確認できます。

- Amazon FSx for Lustre コンソール。
- CloudWatch コンソール。
- CloudWatch コマンドラインインターフェイス (CLI)。
- CloudWatch API。

次の手順は、これらのツールを使用してメトリクスにアクセスする方法を示しています。

Amazon FSx for Lustre コンソールの使用

Amazon FSx for Lustre コンソールを使用してメトリクスを表示するには

1. https://console.aws.amazon.com/fsx/ で Amazon FSx コンソールを開きます。
2. ナビゲーションペインで、[ファイルシステム] を選択し、メトリクスを表示するファイルシステムを選択します。
3. [概要] ページで [モニタリングとパフォーマンス] を選択して、ファイルシステムのメトリクスを確認します。

   [モニタリングとパフォーマンス] パネルには 4 つのタブがあります。

   - [概要] (デフォルトタブ) には、アクティブな警告、CloudWatch アラーム、[ファイルシステムのアクティビティ] のグラフが表示されます。
   - [ストレージ] を選択して、ストレージ容量、使用率、アクティブな警告のメトリクスを表示します。
   - [パフォーマンス] には、ファイルサーバーとストレージのパフォーマンスメトリクスとアクティブな警告が表示されます。
   - [CloudWatch アラーム] には、ファイルシステムに設定されたアラームのグラフが表示されます。

CloudWatch コンソールの使用

CloudWatch コンソールを使用してメトリクスを表示するには

1. CloudWatch コンソール を開きます。

2. ナビゲーションペインで [Metrics] (メトリクス) を選択します。
3. FSx 名前空間を選択します。
4. (オプション) メトリクスを表示するには、検索フィールドにその名前を入力します。
5. (オプション) メトリクスを調べるには、質問に最も一致するカテゴリを選択します。

の使用AWS CLI

AWS CLI からメトリクスにアクセスするには

- `--namespace "AWS/FSx"` 名前空間で `list-metrics` コマンドを使用します。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。

CloudWatch API の使用

CloudWatch API からメトリクスにアクセスするには

- `GetMetricStatistics` を呼び出します。詳細については、「Amazon CloudWatch API リファレンス」を参照してください。

Amazon FSx for Lustre のメトリクスとディメンション

Amazon FSx for Lustre は、すべての FSx for Lustre ファイルシステムの Amazon CloudWatch の `AWS/FSx` 名前空間で、次の表で説明されているメトリクスを発行します。

トピック

FSx for Lustre ネットワーク I/O のメトリクス

AWS/FSx 名前空間には、次のネットワーク I/O メトリクスが含まれます。メトリクスはすべて、FileSystemId という 1 つのディメンションを取ります。

メトリクス	説明
DataReadBytes	クライアントによる読み取りからファイルシステムへのバイト数。 Sum 統計は、指定した期間に読み取りオペレーションと関連付けられているバイト数の合計です。Minimum 統計は、1 つの OST で読み取りオペレーションと関連付けられる総バイト数です。Maximum 統計は、OST で読み込みオペレーションと関連付けられる総バイト数です。Average 統計は、1 つの OST あたりで読み取りオペレーションと関連付けられる総バイト数です。SampleCount 統計は OST の数です。 その期間の平均スループット (バイト / 秒) を算出するには、Sum 統計をその期間の秒数で割ります。 単位: • Sum、Minimum、Maximum、Average ではバイト。 • SampleCount では、カウントです。 有効な統計:Sum 、Minimum、Maximum、Average、SampleCount
DataWriteBytes	クライアントによるファイルシステムへの書き込みからのバイト数。 Sum 統計は書き込みオペレーションと関連付けられる総バイト数です。Minimum 統計は、1 つの OST で書き込みオペレーションと関連付けられる総バイト数です。Maximum 統計は、OST で書き込みオペレーションと関連付けられる総バイト数です。Average 統計は、1 つの OST あたりで書き込みオペレーションと関連付けられる総バイト数です。SampleCount 統計は OST の数です。 その期間の平均スループット (バイト / 秒) を算出するには、Sum 統計をその期間の秒数で割ります。 単位:

メトリクス	説明
	• Sum、Minimum、Maximum、Average ではバイト。 • SampleCount では、カウントです。 有効な統計:Sum 、Minimum、Maximum、Average、SampleCount
DataRead0 perations	ディスク読み取りオペレーションの回数。 Sum 統計は読み取りオペレーションの総回数です。Minimum 統計は、1 つの OST での読み取りオペレーションの最小回数です。Maximum 統計は、OST での読み取りオペレーションの最大回数です。Average 統計は、1 つのOST あたりの読み取りオペレーションの平均回数です。SampleCount 統計は OST の数です。 ある期間の 1 秒あたりの読み取りオペレーションの平均回数を算出するには、Sum 統計をその期間の秒数で割ります。 単位: • Sum、Minimum、Maximum、Average、SampleCount では、カウントです。 有効な統計:Sum 、Minimum、Maximum、Average、SampleCount

メトリクス	説明
DataWrite Operations	書き込みオペレーションの回数。 Sum 統計は書き込みオペレーションの総回数です。Minimum 統計は、1 つの OST での書き込みオペレーションの最小回数です。Maximum 統計は、OST での書き込みオペレーションの最大回数です。Average 統計は、1 つの OST あたりの書き込みオペレーションの平均回数です。SampleCount 統計は OST の数です。 ある期間の 1 秒あたりの書き取りオペレーションの平均回数を算出するには、Sum 統計をその期間の秒数で割ります。 単位: • Sum、Minimum、Maximum、Average、SampleCount では、カウントです。 有効な統計:Sum 、Minimum、Maximum、Average、SampleCount
MetadataOperations	メタデータオペレーションの回数。 Sum 統計はメタデータオペレーションの回数です。Minimum 統計は、1 つの MDT あたりのメタデータオペレーションの最小回数です。Maximum 統計は、1 つの MDT あたりのメタデータオペレーションの最大回数です。Average 統計は、1 つの MDT あたりのメタデータオペレーションの平均回数です。SampleCount 統計は MDT の数です。 その期間の 1 秒あたりの平均メタデータオペレーション回数を算出するには、Sum 統計をその期間の秒数で割ります。 単位: • Sum、Minimum、Maximum、Average、SampleCount では、カウントです。 有効な統計:Sum 、Minimum、Maximum、Average、SampleCount

メトリクス	説明
ClientConnections	クライアントとファイルシステム間のアクティブな接続の数。 単位: 数

FSx for Lustre オブジェクトストレージサーバーのメトリクス

AWS/FSx 名前空間には、次のオブジェクトストレージサーバー (OSS) のメトリクスが含まれます。メトリクスはすべて、FileSystemId と FileServer の 2 つのディメンションを取ります。

- FileSystemId — ファイルシステムの AWS リソース ID。
- FileServer – Lustre ファイルシステム内のオブジェクトストレージサーバー (OSS) の名前。各 OSS は、1 つ以上のオブジェクトストレージターゲット (OST) でプロビジョニングされます。OSS は OSS<HostIndex> の命名規則を使用します。*HostIndex* は 4 桁の 16 進値を表します (例: OSS0001)。OSS の ID は、OSS にアタッチされた最初の OST の ID です。例えば、OST0000 と OST0001 にアタッチされた最初の OSSは OSS0000 を使用し、OST0002、OST0003 にアタッチされた 2 番目の OSSは OSS0002 を使用します。

メトリクス	説明
NetworkThroughputUtilization	ファイルシステムで利用可能なネットワークスループットの割合で表される、ネットワークスループットの使用率です。このメトリクスは、ファイルシステムにおける 1 つの OSS のネットワークスループットキャパシティの割合で表される、NetworkSentBytes および NetworkReceivedBytes の合計に相当します。ファイルシステムの OSS ごと、1 分間で 1 つのメトリクスが出力されます。 Average 統計は、指定した期間での特定の OSS におけるネットワークスループットの平均使用率です。 Minimum 統計は、指定した期間での 1 分間の特定の OSS におけるネットワークスループットの最小使用率です。

メトリクス	説明
	Maximum 統計は、指定した期間での 1 分間の特定の OSS におけるネットワークスループットの最大使用率です。 単位: パーセント 有効な統計:Average 、Minimum、Maximum
NetworkSentBytes	ファイルシステムから送信されたバイト数。リンクされたデータリポジトリとの間でのデータ移動など、すべてのトラフィックはこのメトリクスで考慮されます。ファイルシステムの OSS ごと、1 分間で 1 つのメトリクスが出力されます。 Sum 統計は、指定した期間に特定の OSS からネットワーク経由で送信されたバイト数の合計です。 Average 統計は、指定した期間に特定の OSS からネットワーク経由で送信されたバイト数の平均です。 Minimum 統計は、指定した期間に特定の OSS からネットワーク経由で送信された最小のバイト数です。 Maximum 統計は、指定した期間に特定の OSS からネットワーク経由で送信された最大のバイト数です。 統計の送信スループット (バイト/秒) を算出するには、統計を指定した期間の秒数で割ります。 単位: バイト 有効な統計:Sum 、Average、Minimum、Maximum

メトリクス	説明
NetworkReceivedBytes	ファイルシステムが受信したバイト数。リンクされたデータリポジトリとの間でのデータ移動など、すべてのトラフィックはこのメトリクスで考慮されます。ファイルシステムの OSS ごと、1 分間で 1 つのメトリクスが出力されます。 Sum 統計は、指定した期間にネットワーク経由で特定の OSS によって受信されたバイト数の合計です。 Average 統計は、指定した期間にネットワークを通じて、特定のOSS によって受信されたバイト数の平均です。 Minimum 統計は、指定した期間にネットワークを通じて、特定の OSS によって受信された最小のバイト数です。 Maximum 統計は、指定した期間にネットワークを通じて、特定の OSS によって受信された最大のバイト数です。 統計のスループット (バイト/秒) を算出するには、統計を指定した期間の秒数で割ります。 単位: バイト 有効な統計:Sum 、Average、Minimum、Maximum

メトリクス	説明
FileServerDiskThroughputUtilization	スループットキャパシティによって決定されるプロビジョニング制限に対する割合 (%) で表される、OSS と関連付けられた OST の間のディスクスループットです。このメトリクスは、ファイルシステムにおける OSS のディスクスループットキャパシティの割合で表される、DiskReadBytes および DiskWriteBytes の合計に相当します。ファイルシステムの OSS ごと、1 分間で 1 つのメトリクスが出力されます。 Average 統計は、指定した期間における特定の OSS での OSS ディスクスループットの平均使用率です。 Minimum 統計は、指定した期間における特定の OSS での OSS ディスクスループットの最小使用率です。 Maximum 統計は、指定した期間における特定の OSS での OSS ディスクスループットの最大使用率です。 単位: パーセント 有効な統計:Average 、Minimum、Maximum

FSx for Lustre オブジェクトストレージターゲットメトリクス

AWS/FSx 名前空間には、次のオブジェクトストレージターゲット (OST) のメトリクスが含まれます。メトリクスはすべて、FileSystemId と StorageTargetId の 2 つのディメンションを取ります。

> **Note**
>
> DiskReadOperations および DiskWriteOperations メトリクスは Scratch ファイルシステムでは使用できず、DiskIopsUtilization メトリクスは Scratch および Persistent HDD ファイルシステムでは利用できません。

メトリクス	説明
DiskReadBytes	この OST から読み取られた任意のディスクのバイト数 (ディスク IO)。ファイルシステムの OST ごと、1 分間で 1 つのメトリクスが出力されます。 Sum 統計は、指定した期間に 1 分間で特定の OST から読み込まれたバイト数の合計です。 Average 統計は、指定した期間に 1 分間で特定の OST から読み込まれたバイト数の平均です。 Minimum 統計は、指定した期間に 1 分間で特定の OST から読み込まれた最小のバイト数です。 Maximum 統計は、指定した期間に 1 分間で特定の OST から読み込まれた最大のバイト数です。 統計のディスク読み取りスループット (バイト/秒) を算出するには、統計をその期間の秒数で割ります。 単位: バイト 有効な統計: Sum、Average、Minimum、Maximum
DiskWriteBytes	この OST から書き込まれた任意のディスクのバイト数 (ディスク IO)。ファイルシステムの OST ごと、1 分間で 1 つのメトリクスが出力されます。 Sum 統計は、指定した期間に 1 分間で特定の OST から書き込まれた合計バイト数です。 Average 統計は、指定した期間に 1 分間で特定の OST から書き込まれたバイト数の平均です。 Minimum 統計は、指定した期間に 1 分間で特定の OST から書き込まれた最小のバイト数です。 Maximum 統計は、指定した期間に 1 分間で特定の OST から書き込まれた最大のバイト数です。

メトリクス	説明
	統計のディスク読み取りスループット (バイト/秒) を算出するには、統計をその期間の秒数で割ります 単位: バイト 有効な統計: Sum、Average、Minimum、Maximum
DiskRead0perations	この OST への読み取りオペレーション (ディスク IO) の数。ファイルシステムの OST ごと、1 分間で 1 つのメトリクスが出力されます。 Sum 統計は、指定した期間で特定の OST により実行された読み取りオペレーションの合計数です。 Average 統計は、指定した期間に 1 分間で特定の OST により実行された読み取りオペレーションの平均数です。 Minimum 統計は、指定した期間に 1 分間で特定の OST により実行された読み取りオペレーションの最小数です。 Maximum 統計は、指定した期間に 1 分間で特定の OST により実行された読み取りオペレーションの最大数です。 その期間におけるディスク IOPS の平均を計算するには、Average 統計を使用してその結果を 60 (秒) で割ります。 単位: カウント 有効な統計: Sum、Average、Minimum、Maximum

メトリクス	説明
DiskWriteOperations	この OST への書き込みオペレーション (ディスク IO) の数。ファイルシステムの OST ごと、1 分間で 1 つのメトリクスが出力されます。 Sum 統計は、指定した期間で特定の OST により実行された書き込みオペレーションの合計数です。 Average 統計は、指定した期間に 1 分間で特定の OST により実行された書き込みオペレーションの平均数です。 Minimum 統計は、指定した期間に 1 分間で特定の OST により実行された書き込みオペレーションの最低数です。 Maximum 統計は、指定した期間に 1 分間で特定の OST により実行された書き込みオペレーションの最高数です。 その期間におけるディスク IOPS の平均を計算するには、Average 統計を使用してその結果を 60 (秒) で割ります。 単位: カウント 有効な統計: Sum、Average、Minimum、Maximum
DiskIopsUtilization	OST のディスク IOPS の制限に対する 1 つの OST のディスク IOPS 使用率。ファイルシステムの OST ごと、1 分間で 1 つのメトリクスが出力されます。 Average 統計は、指定した期間における特定の OST でのディスク IOPS の平均使用率です。 Minimum 統計は、指定した期間における特定の OST でのディスク IOPS の最小使用率です。 Maximum 統計は、指定した期間における特定の OST でのディスク IOPS の最大使用率です。 単位: パーセント 有効な統計: Average、Minimum、Maximum

FSx for Lustre メタデータメトリクス

AWS/FSx 名前空間には、次のメタデータメトリクスが含まれます。CPUUtilization メトリクスは FileSystemId および FileServer ディメンションを受け取り、他のメトリクスは FileSystemId および StorageTargetId ディメンションを受け取ります。

- FileSystemId — ファイルシステムの AWS リソース ID。
- StorageTargetId – メタデータターゲット (MDT) の名前。MDT は、MDT<MDTIndex> の命名規則を使用します (例: MDT0001)。
- FileServer – Lustre ファイルシステム内のメタデータサーバー (MDS) の名前。各 MDS は、1 つのメタデータターゲット (MDT) でプロビジョニングされます。MDS は、MDS<HostIndex> の命名規則を使用します。ここで、HostIndex は、サーバーの MDT インデックスを使用して導出された 4 桁の 16 進値を表します。例えば、MDT0000 でプロビジョニングされた最初の MDS は MDS0000 を使用し、MDT0001 でプロビジョニングされた 2 番目の MDSは MDS0001 を使用します。ファイルシステムにメタデータ設定が指定されている場合、ファイルシステムには複数のメタデータサーバーが含まれます。

メトリクス	説明
CPUUtilization	ファイルシステムの MDS CPU リソースの使用率 (%)。ファイルシステムの MDS ごと、1 分間で 1 つのメトリクスが出力されます。 Average 統計は、指定した期間における MDS の平均 CPU 使用率です。 Minimum 統計は、指定した期間における特定の MDS の最小 CPU 使用率です。 Maximum 統計は、指定した期間における特定の MDS の最大 CPU 使用率です。 単位: パーセント 有効な統計: Average、Minimum、Maximum

メトリクス	説明
FileCreateOperations	ファイル作成オペレーションの合計数。 単位: 数
FileOpenOperations	ファイルオープンオペレーションの合計数。 単位: 数
FileDeleteOperations	ファイル削除オペレーションの合計数。 単位: 数
StatOperations	統計オペレーションの合計数。 単位: 数
RenameOperations	インプレースディレクトリの名前変更かクロスディレクトリの名前変更かにかかわらず、ディレクトリの名前変更の合計数。 単位: 数

FSx for Lustre ストレージ容量のメトリクス

AWS/FSx 名前空間には、次のストレージ容量のメトリクスが含まれます。これらのメトリクスはすべて、FileSystemId ディメンションを受け取る LogicalDiskUsage と PhysicalDiskUsage を除く FileSystemId と StorageTargetId の 2 つのディメンションを受け取ります。

メトリクス	説明
FreeDataStorageCapacity	この OST で利用可能なストレージ容量。ファイルシステムの OST ごと、1 分間で 1 つのメトリクスが出力されます。 Sum 統計は、指定した期間に特定の OST で利用可能なバイト数の合計です。

メトリクス	説明
	Average 統計は、指定した期間に特定の OST で利用可能なバイト数の平均です。 Minimum 統計は、指定した期間に特定の OST で利用可能な最小のバイト数です。 Maximum 統計は、指定した期間に特定の OST で利用可能な最大のバイト数です。 単位: バイト 有効な統計: Sum、Average、Minimum、Maximum
StorageCapacityUtilization	特定のファイルシステムの OST におけるストレージ容量の使用率。ファイルシステムの OST ごと、1 分間で 1 つのメトリクスが出力されます。 Average 統計は、指定した期間における特定の OST でのストレージ容量の平均使用量です。 Minimum 統計は、指定した期間における特定の OST でのストレージ容量の最小使用量です。 Maximum 統計は、指定した期間における特定の OST でのストレージ容量の最大使用量です。 単位: パーセント 有効な統計:Average 、Minimum、Maximum

メトリクス	説明
StorageCapacityUtilizationWithCachedWrites	クライアントでのキャッシュされた書き込み用に予約されたスペースを含む、特定のファイルシステム OST のストレージ容量使用率。ファイルシステムの OST ごと、1 分間で 1 つのメトリクスが出力されます。 Average 統計は、指定した期間における特定の OST でのストレージ容量の平均使用量です。 Minimum 統計は、指定した期間における特定の OST でのストレージ容量の最小使用量です。 Maximum 統計は、指定した期間における特定の OST でのストレージ容量の最大使用量です。 単位: パーセント 有効な統計:Average 、Minimum、Maximum
LogicalDiskUsage	格納された (非圧縮) 論理的なデータの量。 Sum 統計は、ファイルシステムに格納された論理的な総バイト数です。Minimum 統計は、ファイルシステムの OST に格納された、論理的な最小バイト数です。Maximum 統計は、ファイルシステムの OST に格納された、論理的な最大バイト数です。Average 統計は、1 つの OST あたりの、格納された論理的な平均バイト数です。SampleCount 統計は OST の数です。 単位: • ではバイト。。SumMinimumMaximum • SampleCount のカウント 有効な統計:Sum 、Minimum、Maximum、Average、SampleCount

メトリクス	説明
PhysicalDiskUsage	ファイルシステムデータ (圧縮) によって物理的に占有されたストレージの量。 Sum 統計は、ファイルシステムの OST に占有された総バイト数です。Minimum 統計は、空の OST に占有された総バイト数です。Maximum 統計は、満杯の OST に占有された総バイト数です。Average 統計は、1 つの OST あたりの、占有された平均バイト数です。SampleCount 統計は OST の数です。 単位: • ではバイト。。SumMinimumMaximum • SampleCount のカウント 有効な統計:Sum、Minimum、Maximum、Average、SampleCount

FSx for Lustre S3 リポジトリメトリクス

FSx for Lustre は、CloudWatch 内の FSx 名前空間に以下の AutoImport (自動インポート) と AutoExport (自動エクスポート) メトリクスを公開します。これらのメトリクスでは、より詳細なデータ測定を行うためにディメンションが使用されます。AutoImport と AutoExport の両方には FileSystemId と Publisher のディメンションがあります。

メトリクス	説明
AgeOfOldestQueuedMessage ディメンション: AutoExport	エクスポートを待機している最も古いメッセージの経過時間 (秒)。 Average 統計は、エクスポートを待機している最も古いメッセージの平均経過時間です。Maximum 統計は、エクスポートキュー内にある 1 つのメッセージの最大秒数です。Minimum 統計は、エクスポートキュー

メトリクス	説明
	内にある 1 つのメッセージの最小秒数です。値が 0 の場合は、エクスポートを待機しているメッセージがないことを示します。 単位: 秒 有効な統計:Average、Minimum、Maximum
RepositoryRenameOperations ディメンション: AutoExport	より大きいディレクトリの名前変更によってファイルシステムによって処理された処理された名前変更の数。 Sum 統計は、ディレクトリの名前変更から生じる名前変更オペレーションの総数です。Average 統計は、ファイルシステムの名前変更オペレーションの平均回数です。Maximum 統計は、ファイルシステムでのディレクトリ名変更と関連付けられている名前変更オペレーションの最大数です。Minimum 統計は、ファイルシステムでのディレクトリ名変更と関連付けられている名前変更の最小数です。 単位: カウント 有効な統計: Sum、Average、Minimum、Maximum、

メトリクス	説明
AgeOfOldestQueuedMessage ディメンション: AutoImport	インポートを待機している最も古いメッセージの経過時間 (秒)。 Average 統計は、インポートを待機している最も古いメッセージの平均経過時間です。Maximum 統計は、インポートキュー内にある 1 つのメッセージの最大秒数です。Minimum 統計は、インポートキュー内にある 1 つのメッセージの最小秒数です。値が 0 の場合は、インポートを待機しているメッセージがないことを示します。 単位: 秒 有効な統計:Average、Minimum、Maximum

FSx for Lustre のディメンション

Amazon FSx for Lustre メトリクスは AWS/FSx 名前空間を使用し、次のディメンションを使用します。

- FileSystemId ディメンションはファイルシステムの ID を示し、その個々のファイルシステムにリクエストするメトリクスをフィルタリングします。ID は、Amazon FSx コンソールの、[ファイルシステム ID] フィールドのファイルシステムの詳細ページの [概要] パネルにあります。ファイルシステム ID は、*fs-01234567890123456* の形式です。describe-file-systems CLI コマンドのレスポンスで ID を確認することもできます (同等の API アクションは DescribeFileSystems)。
- StorageTargetId ディメンションは、メタデータメトリクスを発行した OST (オブジェクトストレージターゲット) または MDT (メタデータターゲット) を示します。StorageTargetId は OSTxxxx (例: OST0001) または MDTxxxx (例: MDT0001) の形式です。
- FileServer ディメンションは以下を示します。
  - OSS メトリクスの場合: オブジェクトストレージサーバー (OSS) の名前。OSS は OSSxxxx 命名規則を使用します (例: OSS0002)。

- CPUUtilization メトリクスの場合: メタデータサーバー (MDS) の名前。MDS は `MDSxxxx` 命名規則を使用します (例: `MDS0002`)。
- CloudWatch と AWS CLI では、`AutoImport` と `AutoImport` のメトリクスを発行したサービスを示す `Publisher` ディメンションを使用できます。

ディメンションの詳細については、「Amazon CloudWatch ユーザーガイド」の「ディメンション」を参照してください。

パフォーマンスの警告と推奨事項

FSx for Lustre では、CloudWatch メトリクスのいずれかが連続した複数のデータポイントで事前に設定されたしきい値に近づいたり超過したりすると、そのメトリクスに対して警告が表示されます。これらの警告により、ファイルシステムのパフォーマンスを最適化するために使用できる実用的な推奨事項が示されます。

警告は、Amazon FSx for Lustre コンソールの [モニタリングとパフォーマンス] ダッシュボードのいくつかのエリアからアクセスできます。Amazon FSx のパフォーマンスに関するアクティブな警告や最新の警告すべて、およびファイルシステム用に設定されたアラーム状態 にある CloudWatch アラームすべてが、[概要] セクションの [モニタリングとパフォーマンス] パネルに表示されます。この警告は、メトリクスグラフが表示されているダッシュボードのセクションにも表示されます。これらの警告は、基盤となるメトリクスが警告しきい値を下回ってから 24 時間後にダッシュボードから自動的に消えます。

Amazon FSx のどのメトリクスに対しても、CloudWatch アラームを作成できます。詳しくは、「メトリクスをモニタリングする CloudWatch アラームを作成する」を参照してください。

パフォーマンスの警告を使用してファイルシステムのパフォーマンスを向上させる

Amazon FSx は、ファイルシステムのパフォーマンスを最適化するために使用できる実用的な推奨事項を提供します。アクティビティが今後も続くと予想される場合、またはその問題がファイルシステムのパフォーマンスに影響を及ぼしている場合は、推奨されるアクションを実行します。警告をトリガーしたメトリクスに応じて、次の表に示すように、ファイルシステムのスループットキャパシティ、ストレージ容量またはメタデータ IOPS を増やすことで解決できます。

ダッシュボードセクション	このメトリクスに対応する警告が存在する場合	この操作を行います
収納家具	`Storage capacity utilization`	ファイルシステムのストレージ容量を増やします。 ファイルシステムのオブジェクトストレージターゲット (OST) のサブセットでストレージ容量使用率が高い場合、ワークロードのバランスを再調整して、ファイルシステム全体でストレージ容量使用率をより均等に調整することもできます。
	`Storage capacity utilization with cached writes`	クライアントで max_dirty_mb パラメータを設定すると、クライアント書き込みキャッシュのサイズを小さくできます。
オブジェクトストレージのパフォーマンス	`Network throughput`	ファイルシステムのスループットキャパシティを増やします。 ファイルシステムのオブジェクトストレージサーバー (OSS) のサブセットでスループット使用率が高い場合は、ワークロードのバランスを再調整して、ファイルシステム全体でスループット使用率をより均等に調整することもできます。
	`Disk throughput`	ファイルシステムのスループットキャパシティを増やします。 ファイルシステムのオブジェクトストレージサーバー (OSS)

ダッシュボードセクション	このメトリクスに対応する警告が存在する場合	この操作を行います
		のサブセットでディスクスループット使用率が高い場合は、ワークロードのバランスを再調整して、ファイルシステム全体でディスクスループット使用率をより均等に調整することもできます。
	Disk IOPS	ファイルシステムのストレージ容量を増やします。 ファイルシステムのオブジェクトストレージターゲット (OST) のサブセットでディスク IOPS 使用率が高い場合は、ワークロードのバランスを再調整して、ディスク IOPS 使用率をファイルシステム全体で均等に調整することもできます。
メタデータパフォーマンス	CPU utilization	ファイルシステムのストレージ容量を増やします。 ストレージ容量とは無関係にメタデータパフォーマンスをスケールする必要がある場合は、MetadataConfiguration パラメータを使用して、ストレージ容量とは無関係にメタデータパフォーマンスのプロビジョニングをサポートする新しいファイルシステムに移行できます。

ダッシュボードセクション	このメトリクスに対応する警告が存在する場合	この操作を行います
	`Metadata IOPS`	ファイルシステムのメタデータ IOPS を増やします。

ファイルシステムのパフォーマンスに関する詳細については、「Amazon FSx for Lustre のパフォーマンス」を参照してください。

メトリクスをモニタリングする CloudWatch アラームを作成する

アラームの状態が変わったら、Amazon SNS メッセージを送信する Amazon CloudWatch のアラームを作成することができます。アラームは、指定期間にわたって単一のメトリクスを監視し、指定したしきい値に対応したメトリクスの値に基づいて、期間数にわたって 1 つ以上のアクションを実行します。アクションは、Amazon SNS のトピックまたは Auto Scaling のポリシーに送信される通知です。

アラームは持続している状態変化に対してのみアクションを呼び出します。CloudWatch アラームは、特定の状態にあるという理由ではアクションを呼び出しません。状態は変わるはずで、指定された期間変更されたままになっている必要があります。Amazon FSx コンソールまたは CloudWatch コンソールでアラームを作成できます。

次の手順は、コンソール、AWS CLI、および API を使用して Amazon FSx for Lustre のアラームを作成する方法を示しています。

Amazon FSx for Lustre コンソールを使用してアラームを設定するには

1. https://console.aws.amazon.com/fsx/ で Amazon FSx コンソールを開きます。
2. ナビゲーションペインで、[ファイルシステム] を選択し、アラームに対して作成したいファイルシステムを選択します。
3. [概要] ページで、[モニタリング] を選択します。
4. [Create CloudWatch alarm] (CloudWatch アラームの作成) を選択します。CloudWatch コンソールにリダイレクトされます。
5. [Select metrics] (メトリクスの選択) を選択し、[Next] (次へ) を選択します。
6. [メトリクス] セクションで、[FSX] を選択します。

7. [ファイルシステムメトリクス] を選択し、アラームを設定するメトリクスを選択し、[メトリクスの選択] を選択します。
8. [条件] セクションで、アラームに使用する条件を選択し、[次へ] を選択します。

> **Note**
>
> ファイルシステムのメンテナンス中は、メトリクスが公開されない場合があります。不必要で誤解を招くようなアラーム条件の変更を防ぎ、欠落しているデータポイントに対する回復力を持つようにアラームを設定するには、「Amazon CloudWatch ユーザーガイド」の「CloudWatch アラームによる欠落データの扱い方を設定する」を参照してください。

9. アラーム状態がアクションをトリガーした際に、CloudWatch から E メール または SNS 通知を受け取りたい場合は、[このアラーム状態がいかなる場合も] を選択します。

   [SNS トピックの選択] (SNS トピックの選択) で、既存の SNS トピックを選択します。[Create topic] (トピックの作成) を選択すると、新しいメールサブスクリプションリストの名前とメールアドレスを設定できます。このリストは保存され、今後のアラーム用のフィールドに表示されます。[Next] (次へ) を選択します。

> **Warning**
>
> [Create Topic] (トピックの作成) を使用して新しい Amazon SNS トピックを作成する場合、メールアドレスを検証しなければ、そのアドレスで通知を受け取ることができません。メールは、アラームがアラーム状態になったときにのみ送信されます。アラーム状態になった際に、メールアドレスの検証がまだ完了していない場合は、そのアドレスで通知を受け取ることはできません。

10. [Name] (名前)、[Description] (説明)、[Whenever] (いつでも) のそれぞれにメトリクスの値を入力し、[Next] (次へ) を選択します。
11. [プレビューと作成] ページでアラームをレビューし、[アラームの作成] を選択します。

CloudWatch コンソールを使用してアラームを設定するには

1. AWS マネジメントコンソール にサインインして、CloudWatch コンソール (https://console.aws.amazon.com/cloudwatch/) を開きます。

2. [Create Alarm] (アラームの作成) を選択して、[Create Alarm Wizard] (アラームウィザードの作成) を起動します。
3. [FSx メトリクス] を選択してメトリクスを見つけます。結果を絞り込むには、ファイルシステム ID を検索します。アラームを作成する対象のメトリクスを選択し、[次へ] を選択します。
4. [名前]、[説明]、[次の時] のそれぞれにメトリクスの値を入力します。
5. アラーム状態に達したときに CloudWatch から E メールを受け取るには、[アラームが次の時] で、[状態: 警告] を選択します。[Send notification to] (通知の宛先) に、既存の SNS トピックを選択します。[トピックの作成] を選択すると、新しいメールサブスクリプションリストの名前とメールアドレスを設定できます。このリストは保存され、今後のアラーム用のフィールドに表示されます。

   > **Warning**
   >
   > [トピックの作成] を使用して新しい Amazon SNS トピックを作成する場合、メールアドレスを検証しなければ、そのアドレスで通知を受け取ることができません。メールは、アラームがアラーム状態になったときにのみ送信されます。アラーム状態になった際に、メールアドレスの検証がまだ完了していない場合は、そのアドレスで通知を受け取ることはできません。

6. [アラームプレビュー] を表示し、[アラームの作成] を選択するか、戻って変更を加えます。

AWS CLI を使用してアラームを設定するには

- `put-metric-alarm` を呼び出します。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。

CloudWatch を使用してアラームを設定するには

- `PutMetricAlarm` を呼び出します。詳細については、「Amazon CloudWatch API リファレンス」を参照してください。

Amazon CloudWatch Logs でのロギング

FSx for Lustre では、ファイルシステムに関連付けられたデータリポジトリのエラーイベントと警告イベントの Amazon CloudWatch Logs へのロギングをサポートします。

> **Note**
>
> Amazon CloudWatch Logs を使用したロギングは、2021 年 11 月 30 日の午後 3 時 PST 以降に作成された Amazon FSx for Lustre ファイルシステムでのみ使用できます。

トピック

ロギングの概要

FSx for Lustre ファイルシステムにデータリポジトリがリンクされている場合は、Amazon CloudWatch Logs へのデータリポジトリイベントのロギングが有効にできます。エラーイベントと警告イベントは、インポート、エクスポート、復元イベントについてログに記録できます。オペレーションおよびデータリポジトリへのリンクの詳細については、「Amazon FSx for Lustre でデータリポジトリの使用」を参照してください。

Amazon FSx がログに記録するログレベルを設定できます。つまり、Amazon FSx がエラーイベントのみ、警告イベントのみ、またはエラーイベントと警告イベントの両方を記録するかどうかを設定できます。イベントログをいつでもオフにすることもできます。

> **Note**
>
> 関連したすべてのレベルの重要な機能を持つファイルシステムには、ログを有効にすることを強くお勧めします。

ログの宛先

ログが有効な場合、FSx for Lustre を Amazon CloudWatch Logs の宛先で設定する必要があります。イベントログの宛先は Amazon CloudWatch Logs ロググループであり、Amazon FSx はこのロググループ内にファイルシステムのログストリーミングを作成します。CloudWatch Logs を使用すると、Amazon CloudWatch コンソールで監査イベントログを保存、表示、検索した

り、CloudWatch Logs インサイトを使用してログに対してクエリを実行したり、CloudWatch アラームまたは Lambda 関数をトリガーしたりできます。

FSx for Lustre ファイルシステムを作成するとき、または後で更新するときに、ログの宛先を選択します。詳細については、「ロギングを管理する」を参照してください。

デフォルトでは、Amazon FSx はアカウントにデフォルトの CloudWatch Logs ロググループを作成し、イベントログの宛先として使用します。イベントログの宛先としてカスタム CloudWatch Logs ロググループを使用する場合は、イベントログの宛先の名前と場所の要件を以下に示します。

- CloudWatch Logs ロググループの名前は、`/aws/fsx/` プレフィックスで始まる必要があります。
- コンソールでファイルシステムを作成または更新するときに既存の CloudWatch Logs ロググループがない場合、Amazon FSx for Lustre は CloudWatch Logs `/aws/fsx/lustre` ロググループでデフォルトのログストリーミングを作成して使用できます。ログストリーミングは、`datarepo_`*`file_system_id`* の形式で作成されます (例えば、`datarepo_fs-0123456789abcdef0`)。
- デフォルトのロググループを使用しない場合は、コンソールでファイルシステムを作成または更新する際に、設定 UI を使用して CloudWatch Logs ロググループを作成できます。
- 宛先 CloudWatch Logs ロググループは、Amazon FSx for Lustre ファイルシステムと同じ AWS パーティション、AWS リージョン、および AWS アカウント に存在している必要があります。

イベントログの宛先はいつでも変更できます。そうすると、新しいイベントログは新しい宛先にのみ送信されます。

ロギングを管理する

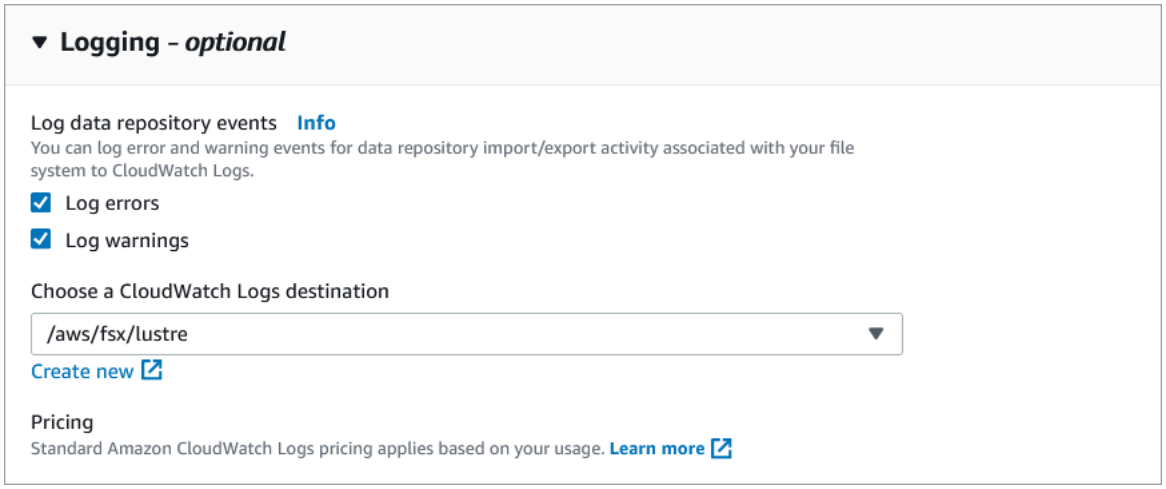
新しい FSx for Lustre ファイルシステムを作成する際や、後で更新する際にログを有効にできます。Amazon FSx コンソールからファイルシステムを作成すると、デフォルトでロギングはオンになります。ただし、AWS CLI または Amazon FSx API を使用してファイルシステムを作成する場合、ロギングはデフォルトでオフになっています。

ロギングが有効になっている既存のファイルシステムでは、イベントを記録するログレベルやログの宛先など、ログイベントの設定を変更できます。タスクは Amazon FSx コンソール、AWS CLI、または Amazon FSx API を使用して実行できます。。

ファイルシステム作成時にロギングを有効にするには (コンソール)

1. https://console.aws.amazon.com/fsx/ で Amazon FSx コンソールを開きます。

2. 「開始方法」セクションの「ステップ 1: FSx for Lustre ファイルシステムの作成」で説明されている新しいファイルシステムを作成する手順に従います。
3. [Logging - optional] (ログ-オプション) セクションを開きます。ロギングはデフォルトで有効になっています。



4. ファイルシステム作成ウィザードの次のセクションに進みます。

ファイルシステムが [Available] (使用可能) の場合、ログが有効になります。

ファイルシステム (CLI) の作成時にログを有効にするには

1. 新しいファイルシステムを作成する場合は、`LogConfiguration` プロパティとの CreateFileSystem オペレーションを実行して、新しいファイルシステムのロギングを有効にします。

```
create-file-system --file-system-type LUSTRE \
    --storage-capacity 1200 --subnet-id subnet-08b31917a72b548a9  \
    --lustre-configuration "LogConfiguration={Level=WARN_ERROR, \
        Destination="arn:aws:logs:us-east-1:234567890123:log-group:/aws/fsx/
testEventLogging"}"
```

2. ファイルシステムが [Available] (使用可能) になると、ログ機能が有効になります。

ログ設定を変更するには (コンソール)

1. https://console.aws.amazon.com/fsx/ で Amazon FSx コンソールを開きます。
2. [ファイルシステム] に移動し、ログを管理する Lustre ファイルシステムを選択します。
3. [Data repository] (データリポジトリ) タブを選択します。

4. ログのパネルで、[Update] (更新) を選択します。
5. ログ設定の更新ダイアログで、目的の設定を変更します。

   a. [Log errors] (エラーのログ) を選択してエラーイベントのみをログに記録するか、[Log warnings] (警告のログ) を選択して警告イベントのみをログに記録するか、またはその両方を選択します。選択を行わないと、ログは無効になります。

   b. 既存の CloudWatch Logs のログ宛先を選択するか、新しいログ宛先を作成します。
6. [Save] (保存) を選択します。

ログ設定を変更するには (CLI)

- `update-file-system` CLI コマンドまたは同等の `UpdateFileSystem` API オペレーションを使用します。

```
update-file-system --file-system-id fs-0123456789abcdef0 \
    --lustre-configuration "LogConfiguration={Level=WARN_ERROR, \
        Destination="arn:aws:logs:us-east-1:234567890123:log-group:/aws/fsx/
testEventLogging"}"
```

ログの表示

Amazon FSx がログの出力を開始した後、ログを表示できます。以下のようにログを表示できます。

- ログを表示するには、Amazon CloudWatch コンソールに移動し、イベントログの宛先となるロググループとログストリーミングを選択します。詳細については、「Amazon CloudWatch Logs ユーザーガイド」の「Cloud Watch Logs に送信されたログデータの表示」を参照してください。
- CloudWatch Logs Insights を使用してログデータをインタラクティブに検索および分析できます。詳細については、「Amazon CloudWatch Logs ユーザーガイド」の「 CloudWatch Logs Insights による ログデータの分析」を参照してください。
- ログを Simple Storage Service (Amazon S3) にエクスポートすることもできます。詳細については、「Amazon CloudWatch Logs ユーザーガイド」の「 Simple Storage Service (Amazon S3) へのログデータのエクスポート」を参照してください。

障害の原因の詳細については、「データリポジトリのイベントログ」を参照してください。

AWS CloudTrail での FSx for Lustre API コールのログロギング

Amazon FSx for Lustre は、AWS CloudTrail と統合されています。これは、Amazon FSx for Lustre のユーザー、ロール、または AWS のサービスで実行されたアクションをレコードするためのサービスです。CloudTrail は、Amazon FSx for Lustre へのすべての API コールをイベントとしてキャプチャします。キャプチャされた呼び出しには、Amazon FSx for Lustre コンソールからの呼び出しと、Amazon FSx for Lustre API オペレーションへのコード呼び出しが含まれます。

追跡を作成する場合は、Amazon FSx for Lustre のイベントなど、Simple Storage Service (Amazon S3) バケットへの CloudTrail イベントの継続的な配信を有効にすることができます。追跡を設定しない場合でも、CloudTrail コンソールの [Event history] (イベント履歴) で最新のイベントを表示できます。CloudTrail で収集された情報に基づいて、Amazon FSx for Lustre に対して行われたリクエストを判断できます。リクエストの実行元 IP アドレス、実行者、実行日時、および追加の詳細を判断することもできます。

CloudTrail に関する詳細は、AWS CloudTrail ユーザーガイド を参照してください。

CloudTrail での Amazon FSx for Lustre の情報

CloudTrail は、アカウントを作成すると AWS アカウントで有効になります。Amazon FSx for Lustre で API アクティビティが発生すると、そのアクティビティは [Event history] (イベント履歴) で AWS のその他のサービスのイベントと共に CloudTrail イベントにレコードされます。最近のイベントは、AWS アカウントで表示、検索、ダウンロードできます。詳細については、「CloudTrail イベント履歴でのイベントの表示」を参照してください。

Amazon FSx for Lustre のイベントなど、AWS アカウントのイベントの継続的なレコードについては、追跡を作成します。[Trail] (追跡) により、CloudTrail はログファイルを Simple Storage Service (Amazon S3) バケットに配信できます。デフォルトでは、コンソールで証跡を作成すると、すべての AWS リージョンに証跡が適用されます。追跡では、AWS パーティション内のすべての AWS リージョンからのイベントを ログに記録し、指定した Simple Storage Service (Amazon S3)バケットにログファイルを配信します。さらに、CloudTrail· ログで収集したイベントデータをより詳細に分析し、それに基づく対応するためにその他の AWS サービスを設定できます。詳細については、『AWS CloudTrail ユーザーガイド:』の以下のトピックを参照してください。

- 追跡作成の概要
- CloudTrail がサポートするサービスと統合
- CloudTrail 用 Amazon SNS 通知の構成

- 複数のリージョンから CloudTrail ログファイルを受け取る および 複数のアカウントから CloudTrail ログファイルを受け取る

すべての Amazon FSx for Lustre の API コール は、CloudTrail によってログに記録されます。例えば、`CreateFileSystem` と `TagResource` オペレーションへのコールは、CloudTrail ログファイルにエントリを生成します。

各イベントまたはログエントリには、誰がリクエストを生成したかという情報が含まれます。アイデンティティ情報は、以下を判別するのに役立ちます:

- リクエストが、ルート認証情報と AWS Identity and Access Management (IAM) ユーザー認証情報のどちらを使用して送信されたか。
- リクエストがロールまたはフェデレーションユーザーのテンポラリなセキュリティ認証情報を使用して行われたかどうか。
- リクエストが別の AWS サービスによって行われたかどうか。

詳細については、「AWS CloudTrail ユーザーガイド」の「CloudTrail userIdentity 要素」を参照してください。

Amazon FSx for Lustre ログファイルエントリの理解

[Trail] (追跡) は、指定した Simple Storage Service (Amazon S3) バケットにイベントをログファイルとして配信するように設定できます。CloudTrail のログファイルは、単一か複数のログエントリを含みます。イベント は、任意の出典からの単一のリクエストを表し、リクエストされたアクション、アクションの日時、リクエストパラメータなどに関する情報が含まれます。CloudTrail· ログファイルは、パブリック API コールの順序付けられたスタックトレースではないため、特定の順序では表示されません。

次の例は、ファイルシステムのタグがコンソールから作成されたときの `TagResource` オペレーションを示す CloudTrail ログエントリを示しています。

```
{
    "eventVersion": "1.05",
    "userIdentity": {
        "type": “Root”,
        "principalId": “111122223333”,
        "arn": "arn:aws:sts::111122223333:root”,
        "accountId": “111122223333”,
```

```
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE”,
        "sessionContext": {
            "attributes": {
                "mfaAuthenticated": "false",
                "creationDate": "2018-11-14T22:36:07Z"
            }
        }
    },
    "eventTime": "2018-11-14T22:36:07Z",
    "eventSource": "fsx.amazonaws.com",
    "eventName": "TagResource",
    "awsRegion": "us-east-1",
    "sourceIPAddress": “192.0.2.0”,
    "userAgent": “console.amazonaws.com”,
    "requestParameters": {
        "resourceARN": "arn:aws:fsx:us-east-1:111122223333:file-system/fs-
ab12cd34ef56gh789”
    },
    "responseElements": null,
    "requestID": “aEXAMPLE-abcd-1234-56ef-b4cEXAMPLE51”,
    "eventID": “bEXAMPLE-gl12-3f5h-3sh4-ab6EXAMPLE9p”,
    "eventType": "AwsApiCall",
    "apiVersion": "2018-03-01",
    "recipientAccountId": “111122223333”
}
```

次の例は、ファイルシステムのタグがコンソールから削除されたときの UntagResource アクションを示す CloudTrail ログエントリを示しています。

```
{
    "eventVersion": "1.05",
    "userIdentity": {
        "type": “Root”,
        "principalId": "111122223333",
        "arn": "arn:aws:sts::111122223333:root",
        "accountId": "111122223333",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "sessionContext": {
            "attributes": {
                "mfaAuthenticated": "false",
                "creationDate": "2018-11-14T23:40:54Z"
```

```
            }
        }
    },
    "eventTime": "2018-11-14T23:40:54Z",
    "eventSource": "fsx.amazonaws.com",
    "eventName": "UntagResource",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "192.0.2.0",
    "userAgent": "console.amazonaws.com",
    "requestParameters": {
        "resourceARN": "arn:aws:fsx:us-east-1:111122223333:file-system/fs-
ab12cd34ef56gh789"
    },
    "responseElements": null,
    "requestID": "aEXAMPLE-abcd-1234-56ef-b4cEXAMPLE51",
    "eventID": "bEXAMPLE-gl12-3f5h-3sh4-ab6EXAMPLE9p",
    "eventType": "AwsApiCall",
    "apiVersion": "2018-03-01",
    "recipientAccountId": "111122223333"
}
```

AWS DataSync を使用した Amazon FSx for Lustre への移行

FSx for Lustre ファイルシステム間のデータ移行には、AWS DataSync を使用できます。DataSync は、インターネットまたは Direct Connect 経由でセルフマネージド型のストレージシステムと AWS ストレージサービス間のデータの移動とレプリケーションを簡素化、自動化、および高速化するデータ転送サービスです。DataSync は、所有権、タイムスタンプ、アクセス許可などのファイルシステムデータおよびメタデータを転送できます。

AWS DataSync を使用して既存のファイルを FSx for Lustre に移行する方法

一度限りのデータ移行、配信ワークロード用の定期的なデータ取り込み、データ保護および回復用のレプリケーションのスケジューリングに、Lustre ファイルシステム用 FSx で DataSync を使用できます。特定の転送シナリオに関する情報については、AWS DataSync ユーザーガイドの「AWS DataSync を使用してデータを転送できる場所」を参照してください。

前提条件

FSx for Lustre のセットアップにデータを移行するには、DataSync 要件を満たすサーバーとネットワークが必要です。詳細については、AWS DataSync ユーザーガイドの「AWS DataSync を使用したセットアップ」を参照してください。

- 送信先の FSx for Lustre ファイルシステムを作成しました。詳細については、「ステップ 1: FSx for Lustre ファイルシステムの作成」を参照してください。
- 送信元のファイルシステムと送信先のファイルシステムは、同じ仮想プライベートクラウド (VPC) 内に接続されています。送信元のファイルシステムは、オンプレミスまたは別の Amazon VPC 内 (AWS アカウント か AWS リージョン) に配置することができますが、Amazon VPC ピアリング、Transit Gateway、AWS Direct Connect、または Site-to-Site VPN を使用して、送信先ファイルシステムのネットワークとピア接続されている必要があります。詳細については、Amazon VPC Peering Guideの「VPC ピア機能とは」を参照してください。

> **Note**
>
> DataSync は、片方の転送場所が Amazon S3 の場合にのみ、FSx for Lustre と AWS アカウント の間で転送できます。

DataSync を使用してファイルを移行するためのベーシックなステップ

DataSync を使用して、送信元から送信先へファイルを転送するには、次のベーシックステップを行います。

1. ご使用の環境にエージェントをダウンロードしてデプロイし、アクティブ化します (AWS のサービス 間で転送する場合は不要)。
2. 送信元と送信先の場所を作成します。
3. タスクを作成します。
4. タスクを実行して、ソースから宛先にファイルを転送します。

詳細については、『AWS DataSync ユーザーガイド:』の以下のトピックを参照してください。

- 「オンプレミスストレージと AWS の間で転送」
- Amazon FSx for Lustre での AWS DataSync 転送の設定。
- 「Amazon EC2 にエージェントをデプロイする」

Amazon FSx for Lustre のセキュリティ

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

セキュリティは、AWS とお客様との間での責任共有です。責任共有モデルでは、これをクラウド のセキュリティおよびクラウド 内 のセキュリティとして説明しています。

- クラウドのセキュリティ - AWS は、Amazon ウェブサービスクラウドで AWS サービスを実行するインフラストラクチャを保護する責任があります。AWS は、安全に使用できるサービスも提供します。サードパーティーの監査人は、AWS コンプライアンスプログラム の一環として、セキュリティの有効性を定期的にテストおよび検証します。「Amazon FSx for Lustre」 に適用されるコンプライアンスプログラムの詳細については、「コンプライアンスプログラムによる対象範囲内の「AWS」 のサービス」を参照してください。
- クラウド内のセキュリティ - ユーザーの責任は、使用する 「AWS」 のサービスに応じて異なります。また、お客様は、データの機密性、お客様の会社の要件、および適用される法律および規制など、その他の要因についても責任を負います。

このドキュメントは、Amazon FSx for Lustre を使用する際に責任共有モデルを適用する方法を理解するのに役立ちます。以下のトピックでは、セキュリティとコンプライアンスの目標を達成するように Amazon FSx を設定する方法について説明します。Amazon FSx for Lustre リソースのモニタリングおよびセキュリティ保護に役立つその他の Amazon サービスを使用する方法も説明します。

以下は、Amazon FSx を操作する際のセキュリティ上の考慮事項についての説明です。

トピック

でのデータ保護Amazon FSx for Lustre

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

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

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

お客様の E メールアドレスなどの極秘または機密情報は、タグ、または名前フィールドなどの自由形式のテキストフィールドに配置しないことを強くお勧めします。これは、コンソール、API、AWS CLI、または AWS SDK で Amazon FSx または他の AWS のサービス を使用する場合も同様です。タグ、または名前に使用される自由形式のテキストフィールドに入力されるデータは、請求または診断ログに使用される場合があります。外部サーバーに URL を提供する場合、そのサーバーへのリクエストを検証できるように、認証情報を URL に含めないことを強くお勧めします。

トピック

Amazon FSx for Lustre でのデータの暗号化

Amazon FSx for Lustre は、ファイルシステムの 2 つの暗号化形式、保管中のデータと転送時の暗号化の暗号化をサポートします。保管中のデータの暗号化は、Amazon FSx ファイルシステムの作成時に自動的に有効になります。この機能をサポートする Amazon EC2 インスタンス から Amazon FSx ファイルシステムにアクセスすると、転送中のデータの暗号化が自動的に有効になります。

暗号化を使用するタイミング

保管時のデータとメタデータの暗号化が必要な企業、または規制ポリシーの対象となる組織の場合は、暗号化されたファイルシステムを作成し、転送中のデータの暗号化を使用してファイルシステムをマウントすることをおすすめします。

コンソールを使用して保管中に暗号化されたファイルシステムを作成する方法の詳細については、「Amazon FSx for Lustre ファイルシステムの作成」を参照してください。

トピック

保管中のデータの暗号化

保管中のデータの暗号化は、AWS マネジメントコンソール、AWS CLI を介して、またはプログラムで Amazon FSx API または AWS SDK の 1 つを介して Amazon FSx for Lustre ファイルシステムを作成すると、自動的に有効になります。組織では、特定の分類に合致する、または特定のアプリケーション、ワークロード、環境に関連するすべてのデータを暗号化する必要が生じる場合があります。永続的ファイルシステムを作成する場合は、データを暗号化する AWS KMS キーを指定できます。スクラッチファイルシステムを作成すると、データは Amazon FSx によって管理されるキーを使用して暗号化されます。コンソールを使用して保管中に暗号化されたファイルシステムを作成する方法の詳細については、「Amazon FSx for Lustre ファイルシステムの作成」を参照してください。

> **Note**
>
> AWS キー管理インフラストラクチャは、連邦情報処理標準 (FIPS) 140-2 で承認された暗号化アルゴリズムを使用します。このインフラストラクチャは、米国標準技術局 (NIST) 800-57 レコメンデーションに一致しています。

FSx for Lustre が AWS KMS を使用する方法の詳細については「Amazon FSx for Lustre で AWS KMS を使用する方法」を参照してください。

保存時の暗号化の方法

暗号化されたファイルシステムの場合、データとメタデータはファイルシステムに書き込まれる前に自動的に暗号化されます。同様に、データとメタデータが読み取られると、アプリケーションに提示される前に自動的に復号化されます。このプロセスは Amazon FSx for Lustre で透過的に処理されるため、アプリケーションを変更する必要はありません。

Amazon FSx for Lustre は、保管中のファイルシステムデータの暗号化に、業界スタンダードの AES-256 暗号化アルゴリズムを使用します。詳細については、「AWS Key Management Service デベロッパーガイド」の「暗号化の基本」を参照してください。

Amazon FSx for Lustre で AWS KMS を使用する方法

Amazon FSx for Lustre はデータがファイルシステムに書き込まれる前にデータを自動的に暗号化し、読み取り時に自動的に復号化します。データは XTS-AES-256 ブロック暗号を使用して暗号化されます。すべてのスクラッチ FSx for Lustre ファイルシステムは、AWS KMS によって管理されるキーを使用して保管時に暗号化されます。Amazon FSx for Lustre は、キーの管理のために AWS KMS と統合します。保管時にスクラッチファイルシステムの暗号化に使用されるキーは、ファイルシステムごとに一意であり、ファイルシステムの削除後に破棄されます。永続ファイルシステムの場合、データの暗号化と復号化に使用される KMS キーを選択します。永続ファイルシステムを作成するときに使用するキーを指定します。この KMS キーの許可を有効化、無効化、または削除することができます。この KMS キーは、以下の 2 つのタイプのいずれかになります。

- Amazon FSx の AWS マネージドキー - これはデフォルトの KMS キーです。KMS キーの作成と保存には料金はかかりませんが、利用料金はかかります。詳細については、「AWS Key Management Service 料金表」を参照してください。
- カスタマー管理キー - これは、キーポリシーと許可を複数のユーザーまたはサービスに設定できる、最も柔軟性のある KMS キーです。カスタマーマネージドキーの作成の詳細については、「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。

ファイルデータ暗号化と復号化の KMS キーとして顧客管理キーを使用する場合は、キーローテーションを有効にできます。キーローテーションを有効にすると、AWS KMS は 1 年に 1 回キーを自動的にローテーションします。また、カスタマー管理キーでは、カスタマー管理キーへのアクセスを無効にしたり、再び有効化したり、削除したり、取り消すタイミングを随時選択することができます。

> **Important**
>
> Amazon FSx は、KMS の対称暗号化キーのみを受け入れます。Amazon FSx では、非対称 KMS キーを使用することはできません。

AWS KMSの Amazon FSx キーポリシー

キーポリシーは、KMS キーへのアクセスをコントロールするための主要な方法です。キーポリシーの詳細については、「AWS Key Management Service デベロッパーガイド」の「AWS KMS でのキーポリシーの使用」を参照してください。次のリストで、Amazon FSx でサポートされる保管時のファイルシステムの暗号化 AWS KMS に関連するすべてのアクセス許可について説明します。

- kms:Encrypt - (オプション) プレーンテキストを暗号化テキストに暗号化します。この許可は、デフォルトのキーポリシーに含まれています。
- kms:Decrypt - (必須) 暗号化テキストを復号します。暗号文は、以前に暗号化された平文です。この許可は、デフォルトのキーポリシーに含まれています。
- kms:ReEncrypt - (オプション) クライアント側にデータのプレーンテキストを公開することなく、サーバー側で新しい KMS キーを使用してデータを暗号化します。データは最初に復号化され、次に再暗号化されます。この許可は、デフォルトのキーポリシーに含まれています。
- kms:GenerateDataKeyWithoutPlaintext - (必須) KMS キーで暗号化されたデータ暗号化キーを返します。この許可は、kms:GenerateDataKey* のデフォルトのキーポリシーに含まれています。
- kms:CreateGrant - (必須) キーを使用できるユーザーとその条件を指定する許可をキーに付与します。付与は、主要なポリシーに対する代替の許可メカニズムです。権限の詳細については、「AWS Key Management Service開発者ガイド」の「権限の使用」を参照してください。この許可は、デフォルトのキーポリシーに含まれています。
- kms:DescribeKey - (必須) 指定された KMS キーに関する詳細情報を提供します。この許可は、デフォルトのキーポリシーに含まれています。
- kms:ListAliases - (オプション) アカウント内のキーエイリアスをすべて一覧表示します。コンソールを使用して暗号化されたファイルシステムを作成すると、このアクセス許可により KMS キーを選択するためのリストに入力されます。最高のユーザーエクスペリエンスを提供するには、この許

可を使用することをお勧めします。このアクセス許可は、デフォルトのキーポリシーに含まれています。

転送中のデータの暗号化

スクラッチ 2 および永続ファイルシステムは、転送中の暗号化をサポートする Amazon EC2 インスタンスからアクセスされると、およびファイルシステム内のホスト間のすべてのコミュニケーションについて、転送中のデータを自動的に暗号化します。転送中の暗号化をサポートする EC2 インスタンスについては、Amazon EC2 ユーザーガイドの「転送中の暗号化」を参照してください。

Amazon FSx for Lustre を使用できる AWS リージョン のリストについては、「デプロイタイプの可用性」を参照してください。

ネットワーク間のトラフィックのプライバシー

このトピックでは、Amazon FSx でサービスから他の口ケーションまでの接続を保護する方法について説明します。

Amazon FSx とオンプレミスクライアント間のトラフィック

プライベートネットワークと AWS との間には 2 つの接続オプションがあります。

- AWS Site-to-Site VPN 接続。詳細については、「AWS Site-to-Site VPN とは?」を参照してください。
- AWS Direct Connect 接続。詳細については、「AWS Direct Connect とは?」を参照してください。

ネットワーク経由で FSx for Lustre にアクセスして、管理タスクを実行するための AWS-公開 API オペレーションと、ファイルシステムと対話するための Lustre ポートにアクセスできます。

API トラフィックの暗号化

AWS 公開型 API オペレーションにアクセスするには、クライアントが Transport Layer Security (TLS) 1.2 以降をサポートしている必要があります。TLS 1.2 が必須で、TLS 1.3 をお勧めします。クライアントは、Ephemeral Diffie-Hellman (DHE) や Elliptic Curve Diffie-Hellman Ephemeral (ECDHE) などの Perfect Forward Secrecy (PFS) を備えた暗号スイートもサポートする必要があります。モードは、Java 7 以降など、最近のほとんどのシステムでサポートされています。また、リクエストは、アクセスキー ID と、IAM プリンシパルに関連付けられているシークレットアクセスキーを使用

して署名する必要があります。または、AWS Security Token Service (STS) を使用して、リクエストに署名するためのテンポラリセキュリティ認証情報を生成できます。

データトラフィックの暗号化

転送中のデータの暗号化は、AWS クラウド 内からファイルシステムにアクセスするサポートされている EC2 インスタンスから有効になります。詳細については、「転送中のデータの暗号化」を参照してください。FSx for Lustre は、オンプレミスのクライアントとファイルシステム間の転送中に暗号化をネイティブに提供しません。

Amazon FSx for Lustre 向けの Identity and Access Management

AWS Identity and Access Management (IAM) は、管理者が AWS リソースへのアクセスを安全に制御 AWS のサービス するのに役立つ です。IAM 管理者は、Amazon FSx リソースを使用するための認証 (サインイン) および認可(許可を持つ) できるユーザーをコントロールします。IAM は、追加料金なしで使用できる AWS のサービス です。

トピック

- オーディエンス
- アイデンティティを使用した認証
- ポリシーを使用したアクセスの管理
- Amazon FSx for Lustre と IAM の連携の仕組み
- Amazon FSx for Lustre のアイデンティティベースのポリシー例
- AWS Amazon FSx for Lustre の マネージドポリシー
- Amazon FSx for Lustre のアイデンティティとアクセスのトラブルシューティング
- Amazon FSx でのタグの使用
- Amazon FSx のサービスリンクロールの使用

オーディエンス

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

- サービスユーザー - 機能にアクセスできない場合は、管理者にアクセス許可をリクエストします (「Amazon FSx for Lustre のアイデンティティとアクセスのトラブルシューティング」を参照してください)

- サービス管理者 - ユーザーアクセスを決定し、アクセス許可リクエストを送信します (「Amazon FSx for Lustre と IAM の連携の仕組み」を参照してください)
- IAM 管理者 - アクセスを管理するポリシーを記述します (「Amazon FSx for Lustre のアイデンティティベースのポリシー例」を参照してください)

アイデンティティを使用した認証

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

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

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

AWS アカウント ルートユーザー

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

フェデレーテッドアイデンティティ

ベストプラクティスとして、人間のユーザーが一時的な認証情報 AWS のサービス を使用して にアクセスするには、ID プロバイダーとのフェデレーションを使用する必要があります。

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

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

IAM ユーザーとグループ

IAM ユーザーは、1 人のユーザーまたは 1 つのアプリケーションに対して特定のアクセス許可を持つ ID です。長期認証情報を持つ IAM ユーザーの代わりに一時的な認証情報を使用することをお勧めします。詳細については、IAM ユーザーガイドの「ID プロバイダーとのフェデレーションを使用してにアクセスすることを人間 AWS のユーザーに要求する」を参照してください。

IAM グループは、IAM ユーザーの集合を指定し、大量のユーザーのアクセス許可の管理を容易にします。詳細については、「IAM ユーザーガイド」の「IAM ユーザーのユースケース」を参照してください。

IAM ロール

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

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

ポリシーを使用したアクセスの管理

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

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

デフォルトでは、ユーザーやロールにアクセス許可はありません。IAM 管理者は IAM ポリシーを作成し、それらをユーザーが担うことができるロールに追加します。IAM ポリシーは、オペレーションの実行方法を問わず、アクセス許可を定義します。

アイデンティティベースのポリシー

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

ID ベースのポリシーは、インラインポリシー (単一の ID に直接埋め込む) または 管理ポリシー (複数の ID にアタッチされるスタンドアロンポリシー) にすることができます。マネージドポリシーまたはインラインポリシーのいずれかを選択する方法については、「IAM ユーザーガイド」の「管理ポリシーとインラインポリシーのいずれかを選択する」を参照してください。

リソースベースのポリシー

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

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

その他のポリシータイプ

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

- アクセス許可の境界 – アイデンティティベースのポリシーで IAM エンティティに付与することのできるアクセス許可の数の上限を設定します。詳細については、「IAM ユーザーガイド」の「IAM エンティティのアクセス許可境界」を参照してください。
- サービスコントロールポリシー (SCP) - 組織または組織単位の最大限のアクセス許可を AWS Organizationsで指定します。詳細については、「AWS Organizations ユーザーガイド」の「サービスコントロールポリシー」を参照してください。
- リソースコントロールポリシー (RCP) – アカウント内のリソースで利用できる最大限のアクセス許可を設定します。詳細については、「AWS Organizations ユーザーガイド」の「リソースコントロールポリシー (RCP)」を参照してください。
- セッションポリシー – ロールまたはフェデレーションユーザーの一時セッションを作成する際にパラメータとして渡される高度なポリシーです。詳細については、「IAM ユーザーガイド」の「セッションポリシー」を参照してください。

複数のポリシータイプ

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

Amazon FSx for Lustre と IAM の連携の仕組み

IAM を使用して Amazon FSx へのアクセスを管理する前に、Amazon FSx で使用できる IAM 機能について理解しておく必要があります。

Amazon FSx for Lustre で使用できる IAM の機能

IAM の特徴量	Amazon FSx のサポート
アイデンティティベースのポリシー	あり
リソースベースのポリシー	なし
ポリシーアクション	あり
ポリシーリソース	あり
ポリシー条件キー	あり
ACL	なし
ABAC (ポリシー内のタグ)	あり
一時的な認証情報	あり
転送アクセスセッション (FAS)	あり
サービスロール	いいえ
サービスリンクロール	はい

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

Amazon FSx のアイデンティティベースの ポリシー

アイデンティティベースのポリシーのサポート: あり

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

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

Amazon FSx のアイデンティティベースのポリシー例

Amazon FSx のアイデンティティベースポリシーの例を確認するには、「Amazon FSx for Lustre のアイデンティティベースのポリシー例」を参照してください。

Amazon FSx 内のリソースベースのポリシー

リソースベースのポリシーのサポート: なし

Amazon FSx のポリシーアクション

ポリシーアクションのサポート: あり

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

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

Amazon FSx のアクションの一覧を確認するには、「サービス認可リファレンス」の「Actions defined by Amazon FSx for Lustre」を参照してください。

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

```
fsx
```

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

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

Amazon FSx のアイデンティティベースポリシーの例を確認するには、「Amazon FSx for Lustre のアイデンティティベースのポリシー例」を参照してください。

Amazon FSx のポリシーリソース

ポリシーリソースのサポート: あり

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

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

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

Amazon FSx リソースのタイプとその ARN の一覧を確認するには、「サービス認可リファレンス」の「Actions defined by Amazon FSx for Lustre」を参照してください。リソースごとの ARN を指定するためのアクションについては、「Amazon FSx for Lustre で定義されるアクション」を参照してください。

Amazon FSx のアイデンティティベースポリシーの例を確認するには、「Amazon FSx for Lustre のアイデンティティベースのポリシー例」を参照してください。

Amazon FSx のポリシー条件キー

サービス固有のポリシー条件キーのサポート: あり

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

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

Amazon FSx での条件キーの一覧を確認するには、「サービス認可リファレンス」の「Actions defined by Amazon FSx for Lustre」を参照してください。条件キーを使用できるアクションとリソースについては、「Amazon FSx for Lustre で定義されるアクション」を参照してください。

Amazon FSx のアイデンティティベースポリシーの例を確認するには、「Amazon FSx for Lustre のアイデンティティベースのポリシー例」を参照してください。

Amazon FSx アクセスコントロールリスト (ACL)

ACL のサポート: なし

Amazon FSx での属性ベースのアクセスコントロール (ABAC)

ABAC (ポリシー内のタグ) のサポート: あり

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

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

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

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

Amazon FSx リソースのタグ付けの詳細については、「Amazon FSx for Lustre リソースのタグ付け」を参照してください。

リソースのタグに基づいてリソースへのアクセスを制限するためのアイデンティティベースのポリシーの例を表示するには、「タグを使用した Amazon FSx リソースへのアクセスのコントロール」を参照してください。

Amazon FSx でのテンポラリ認証情報の使用

一時的な認証情報のサポート: あり

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

Amazon FSx の転送アクセスセッション

転送アクセスセッション (FAS) のサポート: あり

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

Amazon FSx のサービスロール

サービスロールのサポート: なし

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

> **Warning**
>
> サービスロールの許可を変更すると、Amazon FSx の機能が破損する可能性があります。Amazon FSx が指示する場合以外は、サービスロールを編集しないでください。

Amazon FSx のサービスにリンクされたロール

サービスリンクロールのサポート: あり

サービスにリンクされたロールは、 にリンクされたサービスロールの一種です AWS のサービス。サービスは、ユーザーに代わってアクションを実行するロールを引き受けることができます。サービスにリンクされたロールは に表示され AWS アカウント 、サービスによって所有されます。IAM 管理者は、サービスリンクロールのアクセス許可を表示できますが、編集することはできません。

Amazon FSx サービスにリンクされたロールの作成と管理の詳細については、「Amazon FSx のサービスリンクロールの使用」を参照してください。

Amazon FSx for Lustre のアイデンティティベースのポリシー例

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

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

リソースタイプごとの ARN の形式を含む、Amazon FSx で定義されるアクションとリソースタイプの詳細については、「サービス認可リファレンス」の「Amazon FSx for Lustre のアクション、リソース、および条件キー」を参照してください。

トピック

- ポリシーに関するベストプラクティス
- Amazon FSx コンソールの使用
- ユーザーが自分の許可を表示できるようにする

ポリシーに関するベストプラクティス

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

- AWS 管理ポリシーを開始し、最小特権のアクセス許可に移行 – ユーザーとワークロードにアクセス許可の付与を開始するには、多くの一般的なユースケースにアクセス許可を付与するAWS 管理ポリシーを使用します。これらは で使用できます AWS アカウント。ユースケースに固有の AWS カスタマー管理ポリシーを定義することで、アクセス許可をさらに減らすことをお勧めします。詳細については、IAM ユーザーガイド の AWS マネージドポリシー または ジョブ機能のAWS マネージドポリシー を参照してください。
- 最小特権を適用する – IAM ポリシーでアクセス許可を設定する場合は、タスクの実行に必要な許可のみを付与します。これを行うには、特定の条件下で特定のリソースに対して実行できるアクションを定義します。これは、最小特権アクセス許可とも呼ばれています。IAM を使用して許可を適用する方法の詳細については、IAM ユーザーガイド の IAM でのポリシーとアクセス許可 を参照してください。
- IAM ポリシーで条件を使用してアクセスをさらに制限する - ポリシーに条件を追加して、アクションやリソースへのアクセスを制限できます。たとえば、ポリシー条件を記述して、すべてのリクエストを SSL を使用して送信するように指定できます。条件を使用して、サービスアクションが などの特定の を通じて使用されている場合に AWS のサービス、サービスアクションへのアクセスを許可することもできます CloudFormation。詳細については、IAM ユーザーガイド の IAM JSON ポリシー要素:条件 を参照してください。
- IAM Access Analyzer を使用して IAM ポリシーを検証し、安全で機能的なアクセス権限を確保する - IAM Access Analyzer は、新規および既存のポリシーを検証して、ポリシーが IAM ポリシー言語 (JSON) および IAM のベストプラクティスに準拠するようにします。IAM アクセスアナライザーは 100 を超えるポリシーチェックと実用的な推奨事項を提供し、安全で機能的なポリシーの作成をサポートします。詳細については、IAM ユーザーガイド の IAM Access Analyzer でポリシーを検証する を参照してください。
- 多要素認証 (MFA) を要求する – で IAM ユーザーまたはルートユーザーを必要とするシナリオがある場合は AWS アカウント、MFA をオンにしてセキュリティを強化します。API オペレーションが呼び出されるときに MFA を必須にするには、ポリシーに MFA 条件を追加します。詳細については、IAM ユーザーガイド の MFA を使用した安全な API アクセス を参照してください。

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

Amazon FSx コンソールの使用

Amazon FSx for Lustre コンソールにアクセスするには、一連の最小限のアクセス許可が必要です。これらのアクセス許可により、 の Amazon FSx リソースの詳細を一覧表示および表示できます AWS アカウント。最小限必要な許可よりも制限が厳しいアイデンティティベースのポリシーを作成

すると、そのポリシーを持つエンティティ (ユーザーまたはロール) に対してコンソールが意図したとおりに機能しません。

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

ユーザーとロールが引き続き Amazon FSx コンソールを使用できるようにするには、エンティティに AmazonFSxConsoleReadOnlyAccess AWS 管理ポリシーもアタッチします。詳細については、「IAM ユーザーガイド」の「ユーザーへの許可の追加」を参照してください。

AmazonFSxConsoleReadOnlyAccess およびその他の AWS Amazon FSx for Lustre の マネージドポリシー の Amazon FSx マネージドサービスポリシーが表示されます。

ユーザーが自分の許可を表示できるようにする

この例では、ユーザーアイデンティティにアタッチされたインラインおよびマネージドポリシーの表示を IAM ユーザーに許可するポリシーの作成方法を示します。このポリシーには、コンソールで、または AWS CLI または AWS API を使用してプログラムでこのアクションを実行するアクセス許可が含まれています。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "ViewOwnUserInfo",
            "Effect": "Allow",
            "Action": [
                "iam:GetUserPolicy",
                "iam:ListGroupsForUser",
                "iam:ListAttachedUserPolicies",
                "iam:ListUserPolicies",
                "iam:GetUser"
            ],
            "Resource": ["arn:aws:iam::*:user/${aws:username}"]
        },
        {
            "Sid": "NavigateInConsole",
            "Effect": "Allow",
            "Action": [
                "iam:GetGroupPolicy",
                "iam:GetPolicyVersion",
```

```
                "iam:GetPolicy",
                "iam:ListAttachedGroupPolicies",
                "iam:ListGroupPolicies",
                "iam:ListPolicyVersions",
                "iam:ListPolicies",
                "iam:ListUsers"
            ],
            "Resource": "*"
        }
    ]
}
```

AWS Amazon FSx for Lustre の マネージドポリシー

AWS マネージドポリシーは、AWS が作成および管理するスタンドアロンポリシーです。AWS マネージドポリシーは、多くの一般的なユースケースに対してアクセス許可を提供するように設計されているため、ユーザー、グループ、ロールへのアクセス権の割り当てを開始できます。

AWS マネージドポリシーは、ご利用の特定のユースケースに対して最小特権のアクセス許可を付与しない場合があることにご注意ください。これは、すべての AWS ユーザーが使用できるようになるのを避けるためです。ユースケースに固有のカスタマー管理ポリシーを定義して、アクセス許可を絞り込むことをお勧めします。

AWS マネージドポリシーで定義されたアクセス許可は変更できません。AWS が AWS マネージドポリシーに定義されている権限を更新すると、更新はポリシーがアタッチされているすべてのプリンシパルアイデンティティ (ユーザー、グループ、ロール) に影響します。新しい AWS のサービス を起動するか、既存のサービスで新しい API オペレーションが使用可能になると、AWS が AWS マネージドポリシーを更新する可能性が最も高くなります。

詳細については、「IAM ユーザーガイド」の「AWS マネージドポリシー」を参照してください。

AmazonFSxServiceRolePolicy

Amazon FSx がユーザーに代わって AWS リソースを管理できるようにします。詳細については、「Amazon FSx のサービスリンクロールの使用」を参照してください。

AWS マネージドポリシー: AmazonFSxDeleteServiceLinkedRoleAccess

IAM エンティティには `AmazonFSxDeleteServiceLinkedRoleAccess` をアタッチできません。このポリシーはサービスにリンクされ、そのサービス用のサービスにリンクされたロールでのみ使用

されます。このポリシーをアタッチ、デタッチ、変更、または削除することはできません。詳細については、「Amazon FSx のサービスリンクロールの使用」を参照してください。

このポリシーは、Amazon FSx for Lustre によって Amazon FSx でのみ使用する Simple Storage Service (Amazon S3) アクセスのサービスリンクロールを削除できるようにする管理者許可を付与します。

許可の詳細

このポリシーには、Amazon FSx が Simple Storage Service (Amazon S3) アクセスの FSx サービスリンクロールの削除ステータスを表示、削除、および表示できる `iam` での許可が含まれます。

このポリシーの許可を確認するには、AWS マネージドポリシーリファレンスガイドの「AmazonFSxDeleteServiceLinkedRoleAccess」を参照してください。

AWS マネージドポリシー: AmazonFSxFullAccess

IAM エンティティに AmazonFSxFullAccess をアタッチできます。また、このポリシーはユーザーに代わってアクションを実行できることを Amazon FSx に許可するためのサービスロールにも添付されます。

Amazon FSx へのフルアクセスと、関連する AWS サービスへのアクセスを提供します。

許可の詳細

このポリシーには、以下のアクセス許可が含まれています。

- `fsx` – プリンシパルに、Amazon FSx のすべてのアクション (`BypassSnaplockEnterpriseRetention` を除く) を実行するためのフルアクセスを付与します。
- `ds` - プリンシパルに、Directory Service ディレクトリに関する情報の表示を許可します。
- `ec2`
  - プリンシパルが指定した条件下でタグを作成できるようにします。
  - VPC で使用できるすべてのセキュリティグループのセキュリティグループ検証を強化します。
- `iam` - プリンシパルに、ユーザーに代わって Amazon FSx サービスにリンクされたロールを作成することを許可します。これは必須なため、Amazon FSx はユーザーに代わって AWS リソースを管理できます。
- `firehose` - プリンシパルに Amazon Data Firehose へのレコード書き込みを許可します。これは、ユーザーが Firehose に監査アクセスログを送信して、FSx for Windows File Server のファイルシステムアクセスをモニタリングできるようにするために必要です。

- `logs` - プリンシパルに、ロググループ、ログストリームの作成、ログストリームへのイベントの書き込みを許可します。これは、ユーザーが CloudWatch Logs に監査アクセスログを送信して、FSx for Windows File Server のファイルシステムアクセスをモニタリングできるようにするために必要です。

このポリシーの許可を確認するには、AWS マネージドポリシーリファレンスガイドの「AmazonFSxFullAccess」を参照してください。

AWS マネージドマネージドポリシー: AmazonFSxConsoleFullAccess

`AmazonFSxConsoleFullAccess` ポリシーは IAM アイデンティティにアタッチできます。

このポリシーは、Amazon FSx へのフルアクセス許可および AWS マネジメントコンソール 経由での AWS に関連するサービスへのアクセスを許可する管理者許可を付与します。

許可の詳細

このポリシーには、以下のアクセス許可が含まれています。

- `fsx` – プリンシパルに、Amazon FSx マネジメントコンソールのすべてのアクション (`BypassSnaplockEnterpriseRetention` を除く) を実行することを許可します。
- `cloudwatch` – プリンシパルが、Amazon FSx マネジメントコンソールで CloudWatch Alarms およびメトリクスを表示できるようにします。
- `ds` - プリンシパルが、Directory Service ディレクトリに関する情報を一覧表示できるようにします。
- `ec2`
  - プリンシパルが、ルートテーブルにタグを作成し、ネットワークインターフェイス、ルートテーブル、セキュリティグループ、サブネット、および Amazon FSx ファイルシステムに関連付けられた VPC を一覧表示できるようにします。
  - プリンシパルは、VPC で使用できるすべてのセキュリティグループの高度なセキュリティグループ検証を提供します。
  - Amazon FSx ファイルシステムに関連付けられた Elastic Network Interfaces をプリンシパルに表示できるようにします。
- `kms` - プリンシパルに AWS Key Management Service キーのエイリアスを一覧表示できるようにします。

- s3 - プリンシパルが、Simple Storage Service (Amazon S3) バケット内のオブジェクトの一部またはすべてを一覧表示できるようにします (最大 1000)。
- iam - Amazon FSx がユーザーに代わってアクションを実行できるようにするサービスリンクロールを作成する許可を付与します。

このポリシーの許可を確認するには、AWS マネージドポリシーリファレンスガイドの「AmazonFSxConsoleFullAccess」を参照してください。

AWS マネージドポリシー: AmazonFSxConsoleReadOnlyAccess

AmazonFSxConsoleReadOnlyAccess ポリシーは IAM アイデンティティにアタッチできます。

このポリシーは Amazon FSx および関連 AWS サービスへの読み取り専用のアクセス許可を付与し、ユーザーが AWS マネジメントコンソール のサービスに関する情報を表示できるようにします。

許可の詳細

このポリシーには、以下の許可が含まれています。

- fsx - プリンシパルが Amazon FSx マネジメントコンソールで、すべてのタグを含む Amazon FSx ファイルシステムに関する情報を表示できるようにします。
- cloudwatch – プリンシパルが、Amazon FSx マネジメントコンソールで CloudWatch Alarms およびメトリクスを表示できるようにします。
- ds - プリンシパルが、Directory Service Amazon FSx マネジメントコンソール内のディレクトリに関する情報を表示できるようにします。
- ec2
  - Amazon FSx マネジメントコンソールで、プリンシパルが Amazon FSx ファイルシステムに関連付けられている、ネットワークインターフェイス、セキュリティグループ、サブネット、および VPC を表示できるようにします。
  - プリンシパルは、VPC で使用できるすべてのセキュリティグループの高度なセキュリティグループ検証を提供します。
  - Amazon FSx ファイルシステムに関連付けられた Elastic Network Interfaces をプリンシパルに表示できるようにします。
- kms - プリンシパルが AWS Key Management Service Amazon FSx マネジメントコンソールのキーのエイリアスを表示できるようにします。

- `log` - プリンシパルが、リクエストを行うアカウントに関連付けられた Amazon CloudWatch Logs ロググループを記述できるようにします。これは、プリンシパルが FSx for Windows File Server ファイルシステムの既存のファイルアクセス監査の設定を表示できるために必要です。
- `firehose` - プリンシパルが、リクエストを行うアカウントに関連付けられた Amazon Data Firehose 配信ストリームを記述できるようにします。これは、プリンシパルが FSx for Windows File Server ファイルシステムの既存のファイルアクセス監査の設定を表示できるために必要です。

このポリシーの許可を確認するには、AWS マネージドポリシーリファレンスガイドの「AmazonFSxConsoleReadOnlyAccess」を参照してください。

AWS マネージドポリシー: AmazonFSxReadOnlyAccess

`AmazonFSxReadOnlyAccess` ポリシーは IAM アイデンティティにアタッチできます。

- `fsx` - プリンシパルが Amazon FSx マネジメントコンソールで、すべてのタグを含む Amazon FSx ファイルシステムに関する情報を表示できるようにします。
- `ec2` – VPC で使用できるすべてのセキュリティグループのセキュリティグループ検証を強化します。

このポリシーの許可を確認するには、AWS マネージドポリシーリファレンスガイドの「AmazonFSxReadOnlyAccess」を参照してください。

Amazon FSx の AWS マネージドポリシーへの更新

このサービスが変更の追跡を開始してからの、Amazon FSx の AWS マネージドポリシーの更新に関する詳細を表示します。このページへの変更に関する自動アラートについては、Amazon FSx ドキュメント履歴 ページの RSS フィードを購読してください。

変更	説明	日付
AmazonFSxServiceRolePolicy— 既存のポリシーへの更新	Amazon FSx は、プリンシパルが `AmazonFSx.FileSystemId` タグを持つカスタマーネットワークインターフェイスに IPv6 アドレスを割り	2025 年 7 月 22 日

変更	説明	日付
	当てることを許可する新しいアクセス許可 `ec2:AssignIpv6Addresses` を追加しました。	
AmazonFSxServiceRolePolicy - 既存のポリシーへの更新	Amazon FSx は、プリンシパルが `AmazonFSx.FileSystemId` タグを持つカスタマーネットワークインターフェイスから IPv6 アドレスの割り当てを解除できるようにする新しいアクセス許可 `ec2:UnassignIpv6Addresses` を追加しました。	2025 年 7 月 22 日
AmazonFSxConsoleFullAccess - 既存のポリシーへの更新	Amazon FSx に新しいアクセス許可 `fsx:CreateAndAttachS3AccessPoint` が追加されました。これにより、プリンシパルは S3 アクセスポイントを作成し、FSx ボリュームにアタッチできます。	2025 年 6 月 25 日
AmazonFSxConsoleFullAccess - 既存のポリシーへの更新	Amazon FSx は、プリンシパルが AWS リージョン の AWS アカウント 内のすべての S3 アクセスポイントを一覧表示できるようにする新しいアクセス許可`fsx:DescribeS3AccessPointAttachments` を追加しました。	2025 年 6 月 25 日

変更	説明	日付
AmazonFSxConsoleFullAccess - 既存のポリシーへの更新	Amazon FSx に新しいアクセス許可 `fsx:DetachAndDeleteS3AccessPoint` が追加されました。これにより、プリンシパルは S3 アクセスポイントを削除できます。	2025 年 6 月 25 日
AmazonFSxFullAccess - 既存のポリシーへの更新	Amazon FSx に新しいアクセス許可 `fsx:CreateAndAttachS3AccessPoint` が追加されました。これにより、プリンシパルは S3 アクセスポイントを作成し、FSx ボリュームにアタッチできます。	2025 年 6 月 25 日
AmazonFSxFullAccess - 既存のポリシーへの更新	Amazon FSx は、プリンシパルが AWS リージョン の AWS アカウント 内のすべての S3 アクセスポイントを一覧表示できるようにする新しいアクセス許可`fsx:DescribeS3AccessPointAttachments` を追加しました。	2025 年 6 月 25 日
AmazonFSxFullAccess - 既存のポリシーへの更新	Amazon FSx に新しいアクセス許可 `fsx:DetachAndDeleteS3AccessPoint` が追加されました。これにより、プリンシパルは S3 アクセスポイントを削除できます。	2025 年 6 月 25 日

変更	説明	日付
AmazonFSxConsoleReadOnlyAccess - 既存のポリシーへの更新	Amazon FSx に新しいアクセス許可 `ec2:DescribeNetworkInterfaces` が追加されました。これにより、プリンシパルはファイルシステムに関連付けられた Elastic Network Interface を表示できます。	2025 年 2 月 25 日
AmazonFSxConsoleFullAccess - 既存のポリシーへの更新	Amazon FSx に新しいアクセス許可 `ec2:DescribeNetworkInterfaces` が追加されました。これにより、プリンシパルはファイルシステムに関連付けられた Elastic Network Interface を表示できます。	2025 年 2 月 7 日
AmazonFSxServiceRolePolicy - 既存のポリシーへの更新	Amazon FSx に新しいアクセス許可、`ec2:GetSecurityGroupsForVpc` が追加されました。これにより、プリンシパルは VPC で使用できるすべてのセキュリティグループの拡張セキュリティグループ検証を提供できます。	2024 年 1 月 9 日

変更	説明	日付
AmazonFSxReadOnlyAccess – 既存のポリシーへの更新	Amazon FSx に新しいアクセス許可、`ec2:GetSecurityGroupsForVpc` が追加されました。これにより、プリンシパルは VPC で使用できるすべてのセキュリティグループの拡張セキュリティグループ検証を提供できます。	2024 年 1 月 9 日
AmazonFSxConsoleReadOnlyAccess - 既存のポリシーへの更新	Amazon FSx に新しいアクセス許可、`ec2:GetSecurityGroupsForVpc` が追加されました。これにより、プリンシパルは VPC で使用できるすべてのセキュリティグループの拡張セキュリティグループ検証を提供できます。	2024 年 1 月 9 日
AmazonFSxFullAccess - 既存のポリシーへの更新	Amazon FSx に新しいアクセス許可、`ec2:GetSecurityGroupsForVpc` が追加されました。これにより、プリンシパルは VPC で使用できるすべてのセキュリティグループの拡張セキュリティグループ検証を提供できます。	2024 年 1 月 9 日

変更	説明	日付
AmazonFSxConsoleFullAccess - 既存のポリシーへの更新	Amazon FSx に新しいアクセス許可、`ec2:GetSecurityGroupsForVpc` が追加されました。これにより、プリンシパルは VPC で使用できるすべてのセキュリティグループの拡張セキュリティグループ検証を提供できます。	2024 年 1 月 9 日
AmazonFSxFullAccess - 既存のポリシーへの更新	Amazon FSx に、ユーザーが FSx for OpenZFS ファイルシステムに対してクロスリージョンおよびクロスアカウントのデータレプリケーションを実行できるようにする新しいアクセス許可が追加されました。	2023 年 12 月 20 日
AmazonFSxConsoleFullAccess - 既存のポリシーへの更新	Amazon FSx に、ユーザーが FSx for OpenZFS ファイルシステムに対してクロスリージョンおよびクロスアカウントのデータレプリケーションを実行できるようにする新しいアクセス許可が追加されました。	2023 年 12 月 20 日
AmazonFSxFullAccess - 既存のポリシーへの更新	Amazon FSx は、ユーザーが FSx for OpenZFS ファイルシステムのボリュームのオンデマンドレプリケーションを実行できるように、新しいアクセス許可を追加しました。	2023 年 11 月 26 日

変更	説明	日付
AmazonFSxConsoleFullAccess - 既存のポリシーへの更新	Amazon FSx は、ユーザーが FSx for OpenZFS ファイルシステムのボリュームのオンデマンドレプリケーションを実行できるように、新しいアクセス許可を追加しました。	2023 年 11 月 26 日
AmazonFSxFullAccess - 既存のポリシーへの更新	Amazon FSx に、ユーザーが FSx for ONTAP マルチ AZ ファイルシステムに対して共有 VPC サポートを表示、有効化、無効化できるようにする新しいアクセス許可が追加されました。	2023 年 11 月 14 日
AmazonFSxConsoleFullAccess - 既存のポリシーへの更新	Amazon FSx に、ユーザーが FSx for ONTAP マルチ AZ ファイルシステムに対して共有 VPC サポートを表示、有効化、無効化できるようにする新しいアクセス許可が追加されました。	2023 年 11 月 14 日
AmazonFSxFullAccess - 既存のポリシーへの更新	Amazon FSx は、Amazon FSx に FSx for OpenZFS Multi-AZ ファイルシステムのネットワーク設定を管理できるように、新しいアクセス許可を追加しました。	2023 年 8 月 9 日

変更	説明	日付
AWS マネージドポリシー : AmazonFSxServiceRolePolicy — 既存のポリシーの更新	Amazon FSx は、Amazon FSx が CloudWatch メトリクスを AWS/FSx 名前空間に公開するように既存の cloudwatch:PutMetricData アクセス許可を変更しました。	2023 年 7 月 24 日
AmazonFSxFullAccess - 既存のポリシーへの更新	Amazon FSx のポリシーが更新され、fsx:* アクセス権限が削除され、特定の fsx アクションが追加されました。	2023 年 7 月 13 日
AmazonFSxConsoleFullAccess - 既存のポリシーへの更新	Amazon FSx のポリシーが更新され、fsx:* アクセス権限が削除され、特定の fsx アクションが追加されました。	2023 年 7 月 13 日
AmazonFSxConsoleReadOnlyAccess - 既存のポリシーへの更新	Amazon FSx は、FSx for Windows File Server ファイルシステム用の強化されたパフォーマンスメトリクスと推奨アクションをユーザーが Amazon FSx コンソールで表示できるように、新しいアクセス許可を追加しました。	2022 年 9 月 21 日
AmazonFSxConsoleFullAccess - 既存のポリシーへの更新	Amazon FSx は、FSx for Windows File Server ファイルシステム用の強化されたパフォーマンスメトリクスと推奨アクションをユーザーが Amazon FSx コンソールで表示できるように、新しいアクセス許可を追加しました。	2022 年 9 月 21 日

変更	説明	日付
AmazonFSxReadOnlyAccess - トラッキングポリシーをスタートしました	このポリシーにより、すべての Amazon FSx のリソースと、それらに関連付けられたすべてのタグへの読み取り専用アクセスを許可します。	2022 年 2 月 4 日
AmazonFSxDeleteServiceLinkedRoleAccess - トラッキングポリシーをスタートしました	このポリシーは、Amazon FSx が Simple Storage Service (Amazon S3) アクセスのサービスにリンクされたロールを削除することを許可する管理者許可を付与します。	2022 年 1 月 7 日
AmazonFSxServiceRolePolicy - 既存のポリシーへの更新	Amazon FSx は、Amazon FSx for NetApp ONTAP ファイルシステムのネットワーク設定を管理できるように、新しいアクセス許可を追加しました。	2021 年 9 月 2 日
AmazonFSxFullAccess - 既存のポリシーへの更新	Amazon FSx は、Amazon FSx がスコープダウン呼び出し用の EC2 ルートテーブルにタグを作成できるように、新しいアクセス許可を追加しました。	2021 年 9 月 2 日
AmazonFSxConsoleFullAccess - 既存のポリシーへの更新	Amazon FSx は、Amazon FSx が Amazon FSx for NetApp ONTAP マルチ AZ を作成できるように、新しいアクセス許可を追加しました。	2021 年 9 月 2 日

変更	説明	日付
AmazonFSxConsoleFullAccess - 既存のポリシーへの更新	Amazon FSx は、Amazon FSx がスコープダウン呼び出し用の EC2 ルートテーブルにタグを作成できるように、新しいアクセス許可を追加しました。	2021 年 9 月 2 日
AmazonFSxServiceRolePolicy - 既存のポリシーへの更新	Amazon FSx は、Amazon FSx が CloudWatch Logs ログストリームを記述および書き込むことを許可にする新しいパーミッションを追加しました。 これは、ユーザーが CloudWatch Logs を使用して FSx for Windows File Server ファイルシステムのファイルアクセス監査ログを表示できるようにするために必要です。	2021 年 6 月 8 日
AmazonFSxServiceRolePolicy - 既存のポリシーへの更新	Amazon FSx は、Amazon FSx が Amazon Data Firehose 配信ストリームを記述および書き込みできるようにする新しいアクセス許可を追加しました。 これは、ユーザーが Amazon Data Firehose を使用して FSx for Windows File Server ファイルシステムのファイルアクセス監査ログを表示できるようにするために必要です。	2021 年 6 月 8 日

変更	説明	日付
AmazonFSxFullAccess - 既存のポリシーへの更新	Amazon FSx では、プリンシパルが CloudWatch Logs ロググループ、ログストリーミング、およびログストリームへのイベントの書き込みを記述および作成できる新しいアクセス許可が追加されました。 これは、プリンシパルが CloudWatch Logs を使用して FSx for Windows File Server ファイルシステムのファイルアクセス監査ログを表示できるようにするために必要です。	2021 年 6 月 8 日
AmazonFSxFullAccess - 既存のポリシーへの更新	Amazon FSx は、プリンシパルが Amazon Data Firehose にレコードを記述および書き込むことを許可する新しい許可を追加しました。 これは、ユーザーが Amazon Data Firehose を使用して FSx for Windows File Server ファイルシステムのファイルアクセス監査ログを表示できるようにするために必要です。	2021 年 6 月 8 日

変更	説明	日付
AmazonFSxConsoleFullAccess - 既存のポリシーへの更新	Amazon FSx は、プリンシパルがリクエストを行うアカウントに関連付けられた Amazon CloudWatch Logs ロググループを記述できるように、新しいアクセス許可を追加しました。 これは、FSx for Windows File Server ファイルシステムのファイルアクセス監査を設定するときに、プリンシパルが既存の CloudWatch Logs ロググループを選択できるために必要です。	2021 年 6 月 8 日
AmazonFSxConsoleFullAccess - 既存のポリシーへの更新	Amazon FSx は、プリンシパルがリクエストを行うアカウントに関連付けられた Amazon Data Firehose 配信ストリームを記述できるように、新しいアクセス許可を追加しました。 これは、FSx for Windows File Server ファイルシステムのファイルアクセス監査を設定する際に、プリンシパルが既存の Firehose 配信ストリームを選択できるようにするために必要です。	2021 年 6 月 8 日

変更	説明	日付
AmazonFSxConsoleReadOnlyAccess - 既存のポリシーへの更新	Amazon FSx は、プリンシパルがリクエストを行うアカウントに関連付けられた Amazon CloudWatch Logs ロググループを記述できるように、新しいアクセス許可を追加しました。 これは、プリンシパルが FSx for Windows File Server ファイルシステムの既存のファイルアクセス監査の設定を表示できるために必要です。	2021 年 6 月 8 日
AmazonFSxConsoleReadOnlyAccess - 既存のポリシーへの更新	Amazon FSx は、プリンシパルがリクエストを行うアカウントに関連付けられた Amazon Data Firehose 配信ストリームを記述できるように、新しいアクセス許可を追加しました。 これは、プリンシパルが FSx for Windows File Server ファイルシステムの既存のファイルアクセス監査の設定を表示できるために必要です。	2021 年 6 月 8 日
Amazon FSx が変更の追跡をスタートしました	Amazon FSx が AWS マネージドポリシーの変更の追跡をスタートしました。	2021 年 6 月 8 日

Amazon FSx for Lustre のアイデンティティとアクセスのトラブルシューティング

Amazon FSx と IAM の使用に伴って発生する可能性がある一般的な問題の診断や修復には、次の情報を利用してください。

トピック

- Amazon FSx でアクションを実行する認可がありません
- iam:PassRole を実行する権限がありません
- 自分の 以外のユーザーに Amazon FSx リソース AWS アカウント へのアクセスを許可したい

Amazon FSx でアクションを実行する認可がありません

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

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

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

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

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

iam:PassRole を実行する権限がありません

iam:PassRole アクションを実行する権限がないというエラーが表示された場合は、ポリシーを更新して Amazon FSx にロールを渡せるようにする必要があります。

一部の AWS のサービス では、新しいサービスロールまたはサービスにリンクされたロールを作成する代わりに、既存のロールをそのサービスに渡すことができます。そのためには、サービスにロールを渡すアクセス許可が必要です。

以下の例のエラーは、marymajor という IAM ユーザーがコンソールを使用して Amazon FSx でアクションを実行しようする場合に発生します。ただし、このアクションをサービスが実行するには、サービスロールから付与されたアクセス許可が必要です。Mary には、ロールをサービスに渡すアクセス許可がありません。

```
User: arn:aws:iam::123456789012:user/marymajor is not authorized to perform:
 iam:PassRole
```

この場合、Mary のポリシーを更新してメアリーに iam:PassRole アクションの実行を許可する必要があります。

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

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

他のアカウントのユーザーや組織外の人が、リソースにアクセスするために使用できるロールを作成できます。ロールの引き受けを委託するユーザーを指定できます。リソースベースのポリシーまたはアクセスコントロールリスト (ACL) をサポートするサービスの場合、それらのポリシーを使用して、リソースへのアクセスを付与できます。

詳細については、以下を参照してください。

- Amazon FSx が機能をサポートしているかどうかを確認するには、「Amazon FSx for Lustre と IAM の連携の仕組み」を参照してください。
- 所有 AWS アカウント している のリソースへのアクセスを提供する方法については、IAM ユーザーガイドの「所有 AWS アカウント している別の の IAM ユーザーへのアクセスを提供する」を参照してください。
- リソースへのアクセスをサードパーティーに提供する方法については AWS アカウント、IAM ユーザーガイドの「サードパーティー AWS アカウント が所有する へのアクセスを提供する」を参照してください。
- ID フェデレーションを介してアクセスを提供する方法については、IAM ユーザーガイド の 外部で認証されたユーザー (ID フェデレーション) へのアクセスの許可 を参照してください。
- クロスアカウントアクセスにおけるロールとリソースベースのポリシーの使用方法の違いについては、IAM ユーザーガイド の IAM でのクロスアカウントのリソースへのアクセス を参照してください。

Amazon FSx でのタグの使用

タグを使用すると、Amazon FSx リソースへのアクセスをコントロールしたり、属性ベースのアクセスコントロール (ABAC) を実装したりできます。作成中に Amazon FSx リソースにタグを適用するには、ユーザーは特定の AWS Identity and Access Management (IAM) 許可を持っている必要があります。

作成中にリソースにタグを付ける許可を付与する

リソースを作成する一部の Amazon FSx for Lustre の API アクションでは、リソースの作成時にタグを指定することができます。リソースタグを使用して、属性ベースのアクセスコントロール (ABAC) を実装できます。詳細については、IAM ユーザーガイドの「ABAC とは AWS」を参照してください。

ユーザーが作成時にタグを付けるには、`fsx:CreateFileSystem` などのリソースを作成するアクションを使用するためのアクセス許可が必要です。リソース作成アクションでタグが指定されている場合、IAM は `fsx:TagResource` アクションに対して追加の認可を実行して、ユーザーがタグを作成する認可を持っているかどうかを確認します。そのため、ユーザーには、`fsx:TagResource` アクションを使用する明示的なアクセス許可も必要です。

次のポリシー例では、特定の での作成時に、ユーザーがファイルシステムを作成し、タグを適用することを許可します AWS アカウント。

```
{
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
         "fsx:CreateFileSystem",
         "fsx:TagResource"
      ],
      "Resource": [
         "arn:aws:fsx:region:account-id:file-system/*"
      ]
    }
  ]
}
```

同様に、次のポリシーでは、ユーザーが特定のファイルシステム上でバックアップを作成し、バックアップ作成時にバックアップにタグを適用することができます。

```
{
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
         "fsx:CreateBackup"
      ],
      "Resource": "arn:aws:fsx:region:account-id:file-system/file-system-id*"
    },
    {
      "Effect": "Allow",
      "Action": [
         "fsx:TagResource"
      ],
      "Resource": "arn:aws:fsx:region:account-id:backup/*"
    }
  ]
}
```

`fsx:TagResource` アクションは、タグがリソース作成アクション時に適用された場合のみ評価されます。したがって、リクエストでタグが指定されていない場合、リソースを作成するアクセス許可を持つユーザー (タグ付け条件がないと仮定) には、`fsx:TagResource` アクションを実行するアクセス許可は必要ありません。ただし、ユーザーがタグ付きリソースを作成しようとした場合、ユーザーが `fsx:TagResource` アクションを使用するアクセス許可を持っていない場合はリクエストに失敗します。

Amazon FSx リソースのタグ付けの詳細については、「Amazon FSx for Lustre リソースのタグ付け」を参照してください。タグを使用した Amazon FSx for Lustre リソースへのアクセスコントロールの詳細については、「タグを使用した Amazon FSx リソースへのアクセスのコントロール」を参照してください。

タグを使用した Amazon FSx リソースへのアクセスのコントロール

Amazon FSx とアクションへのアクセスをコントロールするには、タグに基づいて IAM ポリシーを使用できます。コントロールは 2 つの方法で可能です。

- それらのリソースのタグに基づいて、Amazon FSx へのアクセスをコントロールできます。
- IAM リクエストの条件でどのタグを渡すかをコントロールできます。

タグを使用して AWS リソースへのアクセスを制御する方法については、IAM ユーザーガイドの「タグを使用したアクセスの制御」を参照してください。作成時の Amazon FSx リソースのタグ付けの詳細については、「作成中にリソースにタグを付ける許可を付与する」を参照してください。リソースのタグ付けの詳細については、「Amazon FSx for Lustre リソースのタグ付け」を参照してください

リソースのタグに基づいてアクセスのコントロール

ユーザーまたはロールが Amazon FSx リソースで実行できるアクションをコントロールするには、リソースでタグを使用できます。例えば、リソースのタグのキーバリューのペアに基づいて、ファイルシステムリソースに対する特定の API オペレーションを許可または拒否することが必要な場合があります。

Exampleポリシーの例 - 特定のタグを指定するときにファイルシステムを作成する

このポリシーにより、ユーザーは特定のタグとキーバリューのペア (この例では `key=Department, value=Finance`) でタグ付けした場合にのみファイルシステムを作成できます。

```
{
    "Effect": "Allow",
    "Action": [
        "fsx:CreateFileSystem",
        "fsx:TagResource"
    ],
    "Resource": "arn:aws:fsx:region:account-id:file-system/*",
    "Condition": {
        "StringEquals": {
            "aws:RequestTag/Department": "Finance"
        }
    }
}
```

Exampleポリシーの例 - 特定のタグを持つファイルシステムでのみバックアップを作成する

このポリシーにより、ユーザーはキーと値のペア `key=Department, value=Finance` でタグ付けされたファイルシステムでのみバックアップを作成でき、バックアップはタグ `Deparment=Finance` で作成されます。

JSON

```
{
```

```
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "fsx:CreateBackup"
            ],
            "Resource": "arn:aws:fsx:us-east-1:111122223333:file-system/*",
            "Condition": {
                "StringEquals": {
                    "aws:ResourceTag/Department": "Finance"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "fsx:TagResource",
                "fsx:CreateBackup"
            ],
            "Resource": "arn:aws:fsx:us-east-1:111122223333:backup/*",
            "Condition": {
                "StringEquals": {
                    "aws:RequestTag/Department": "Finance"
                }
            }
        }
    ]
}
```

Exampleポリシーの例 - 特定のタグを持つバックアップから特定のタグを持つファイルシステムを作成する

このポリシーにより、ユーザーは、Department=Finance でタグ付けされたバックアップからのみ Department=Finance でタグ付けされたファイルシステムを作成できます。

JSON

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

```
        {
            "Effect": "Allow",
            "Action": [
                "fsx:CreateFileSystemFromBackup",
                "fsx:TagResource"
            ],
            "Resource": "arn:aws:fsx:us-east-1:111122223333:file-system/*",
            "Condition": {
                "StringEquals": {
                    "aws:RequestTag/Department": "Finance"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "fsx:CreateFileSystemFromBackup"
            ],
            "Resource": "arn:aws:fsx:us-east-1:111122223333:backup/*",
            "Condition": {
                "StringEquals": {
                    "aws:ResourceTag/Department": "Finance"
                }
            }
        }
    ]
}
```

Exampleポリシーの例 - 特定のタグを持つファイルシステムの削除

このポリシーにより、ユーザーは Department=Finance でタグ付けされたファイルシステムのみを削除できます。最終バックアップを作成する場合は、それは Department=Finance でタグ付けされる必要があります。FSx for Lustre ファイルシステムの場合、ユーザーは最終バックアップを作成するために fsx:CreateBackup 特権が必要です。

JSON

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

```
            "Effect": "Allow",
            "Action": [
                "fsx:DeleteFileSystem"
            ],
            "Resource": "arn:aws:fsx:us-east-1:111122223333:file-system/*",
            "Condition": {
                "StringEquals": {
                    "aws:ResourceTag/Department": "Finance"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "fsx:CreateBackup",
                "fsx:TagResource"
            ],
            "Resource": "arn:aws:fsx:us-east-1:111122223333:backup/*",
            "Condition": {
                "StringEquals": {
                    "aws:RequestTag/Department": "Finance"
                }
            }
        }
    ]
}
```

Exampleポリシーの例 - 特定のタグを持つファイルシステム上にデータリポジトリタスクを作成する

このポリシーにより、ユーザーは Department=Finance でタグ付けされたデータリポジトリタスクを作成でき、Department=Finance でタグ付けされたファイルシステムでのみ作成できます。

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "fsx:CreateDataRepositoryTask"
            ],
```

```
            "Resource": "arn:aws:fsx:us-east-1:111122223333:file-system/*",
            "Condition": {
                "StringEquals": {
                    "aws:ResourceTag/Department": "Finance"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "fsx:CreateDataRepositoryTask",
                "fsx:TagResource"
            ],
            "Resource": "arn:aws:fsx:us-east-1:111122223333:task/*",
            "Condition": {
                "StringEquals": {
                    "aws:RequestTag/Department": "Finance"
                }
            }
        }
    ]
}
```

Amazon FSx のサービスリンクロールの使用

Amazon FSx は AWS Identity and Access Management (IAM) サービスにリンクされたロールを使用します。サービスリンクロールは、Amazon FSx に直接リンクされているユニークなタイプの IAM ロールです。サービスにリンクされたロールは Amazon FSx によって事前定義されており、サービスがユーザーに代わって他の AWS サービスを呼び出すために必要なすべてのアクセス許可が含まれています。

サービスにリンクされたロールを使用すると、必要な許可を手動で追加する必要がないため、Amazon FSx のセットアップが簡単になります。サービスリンクロールの許可は Amazon FSx が定義し、特に定義されない限り、Amazon FSx のみがそのロールを引き受けることができます。定義される許可には信頼ポリシーと許可ポリシーが含まれ、その許可ポリシーを他のIAM エンティティに添付することはできません。

サービスリンクロールを削除するには、まずその関連リソースを削除しなければなりません。これは、リソースにアクセスするための許可を不用意に削除できないため、Amazon FSx リソースを保護できます。

サービスにリンクされたロールをサポートする他のサービスの詳細については、AWS「IAM と連携するサービス」を参照し、「サービスにリンクされたロール」列で「はい」があるサービスを探します。サービスリンクロールに関するドキュメントをサービスで表示するには、リンクで [はい] を選択します。

Amazon FSx のサービスリンクロール許可

Amazon FSx は、アカウント内で特定のアクションを実行する `AWSServiceRoleForAmazonFSx` および `AWSServiceRoleForFSxS3Access_`*`fs-01234567890`* という名前の 2 つのサービスリンクロールを使用します。アクションの例としては、VPC 内のファイルシステム用の Elastic Network Interface を作成したり、Simple Storage Service (Amazon S3) バケットのデータリポジトリにアクセスしたりすることが挙げられます。`AWSServiceRoleForFSxS3Access_`*`fs-01234567890`* では、S3 バケットにリンクされている Amazon FSx for Lustre ファイルシステムを作成するごとに、このサービスにリンクされたロールが作成されます。

AWSServiceRoleForAmazonFSx アクセス許可の詳細

の場合`AWSServiceRoleForAmazonFSx`、ロールのアクセス許可ポリシーにより、Amazon FSx は該当するすべての AWS リソースに対してユーザーに代わって以下の管理アクションを実行できます。

このポリシーの更新については、「AmazonFSxServiceRolePolicy」を参照してください

> **Note**
>
> AWSServiceRoleForAmazonFSx は、すべての Amazon FSx ファイルシステムタイプで使用されます。リストされたアクセス許可の一部は、FSx for Lustre には適用されません。

- ds – Amazon FSx が Directory Service ディレクトリ内のアプリケーションを表示、認可、および認可解除できるようにします。
- ec2 - Amazon FSx に以下のことを許可します:
  - Amazon FSx ファイルシステムに関連付けられたネットワークインターフェイスを表示、作成、および関連付け解除します。
  - Amazon FSx ファイルシステムに関連付けられた 1 つ以上の Elastic IP アドレスを表示します。
  - Amazon FSx ファイルシステムに関連付けられている Amazon VPC、セキュリティグループ、およびサブネットを表示します。

  - AmazonFSx.FileSystemId タグを持つカスタマーネットワークインターフェイスに IPv6 アドレスを割り当てます。
  - AmazonFSx.FileSystemId タグを持つカスタマーネットワークインターフェイスから IPv6 アドレスの割り当てを解除します。
  - VPC で使用できるすべてのセキュリティグループのセキュリティグループ検証を強化します。
  - AWS認可されたユーザーがネットワークインターフェイスで特定のオペレーションを実行するためのアクセス許可を作成します。
- cloudwatch – Amazon FSx がメトリクスデータポイントを AWS/FSx 名前空間の CloudWatch に発行できるようにします。
- route53 - Amazon FSx に Amazon VPC をプライベートホストゾーンに関連付けることを許可します。
- logs - Amazon FSx が CloudWatch Logs のログストリーミングを記述して書き込むことを許可します。これは、ユーザーが FSx for Windows File Server ファイルシステムのファイルアクセス監査ログを CloudWatch Logs ストリーミングに送信できるようにするためです。
- firehose - Amazon FSx に Amazon Data Firehose 配信ストリームを記述して書き込むことを許可します。これは、ユーザーが FSx for Windows File Server ファイルシステムのファイルアクセス監査ログを Amazon Data Firehose 配信ストリームに公開できるようにするためです。

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "CreateFileSystem",
            "Effect": "Allow",
            "Action": [
                "ds:AuthorizeApplication",
                "ds:GetAuthorizedApplicationDetails",
                "ds:UnauthorizeApplication",
                "ec2:CreateNetworkInterface",
                "ec2:CreateNetworkInterfacePermission",
                "ec2:DeleteNetworkInterface",
                "ec2:DescribeAddresses",
                "ec2:DescribeDhcpOptions",
                "ec2:DescribeNetworkInterfaces",
                "ec2:DescribeRouteTables",
```

```
                "ec2:DescribeSecurityGroups",
                "ec2:DescribeSubnets",
                "ec2:DescribeVPCs",
                "ec2:DisassociateAddress",
                "ec2:GetSecurityGroupsForVpc",
                "route53:AssociateVPCWithHostedZone"
            ],
            "Resource": "*"
        },
        {
            "Sid": "PutMetrics",
            "Effect": "Allow",
            "Action": [
                "cloudwatch:PutMetricData"
            ],
            "Resource": [
                "*"
            ],
            "Condition": {
                "StringEquals": {
                    "cloudwatch:namespace": "AWS/FSx"
                }
            }
        },

        {
            "Sid": "TagResourceNetworkInterface",
            "Effect": "Allow",
            "Action": [
                "ec2:CreateTags"
            ],
            "Resource": [
                "arn:aws:ec2:*:*:network-interface/*"
            ],
            "Condition": {
                "StringEquals": {
                    "ec2:CreateAction": "CreateNetworkInterface"
                },
                "ForAllValues:StringEquals": {
                    "aws:TagKeys": "AmazonFSx.FileSystemId"
                }
            }
        },
        {
```

```
            "Sid": "ManageNetworkInterface",
            "Effect": "Allow",
            "Action": [
                "ec2:AssignPrivateIpAddresses",
                "ec2:ModifyNetworkInterfaceAttribute",
                "ec2:UnassignPrivateIpAddresses"
            ],
            "Resource": [
                "arn:aws:ec2:*:*:network-interface/*"
            ],
            "Condition": {
                "Null": {
                    "aws:ResourceTag/AmazonFSx.FileSystemId": "false"
                }
            }
        },
        {
            "Sid": "ManageRouteTable",
            "Effect": "Allow",
            "Action": [
                "ec2:CreateRoute",
                "ec2:ReplaceRoute",
                "ec2:DeleteRoute"
            ],
            "Resource": [
                "arn:aws:ec2:*:*:route-table/*"
            ],
            "Condition": {
                "StringEquals": {
                    "aws:ResourceTag/AmazonFSx": "ManagedByAmazonFSx"
                }
            }
        },
        {
            "Sid": "PutCloudWatchLogs",
            "Effect": "Allow",
            "Action": [
                "logs:DescribeLogGroups",
                "logs:DescribeLogStreams",
                "logs:PutLogEvents"
            ],
            "Resource": "arn:aws:logs:*:*:log-group:/aws/fsx/*"
        },
        {
```

```
            "Sid": "ManageAuditLogs",
            "Effect": "Allow",
            "Action": [
                "firehose:DescribeDeliveryStream",
                "firehose:PutRecord",
                "firehose:PutRecordBatch"
            ],
            "Resource": "arn:aws:firehose:*:*:deliverystream/aws-fsx-*"
        }
    ]
}
```

本ポリシーの更新については、Amazon FSx の AWS マネージドポリシーへの更新 に記載されています。

サービスリンクロールの作成、編集、削除を IAM エンティティ (ユーザー、グループ、ロールなど) に許可するにはアクセス許可を設定する必要があります。詳細については、「IAM ユーザーガイド」の「サービスにリンクされたロールのアクセス許可」を参照してください。

AWSServiceRoleForFSxS3Access アクセス許可の詳細

AWSServiceRoleForFSxS3Access_*file-system-id* では、ロールのアクセス許可ポリシーにより、Amazon FSx が、Amazon FSx for Lustre ファイルシステムのデータリポジトリをホストしている Amazon S3 バケット上で以下のアクションを実行することを許可します。

- s3:AbortMultipartUpload
- s3:DeleteObject
- s3:Get*
- s3:List*
- s3:PutBucketNotification
- s3:PutObject

サービスリンクロールの作成、編集、削除を IAM エンティティ (ユーザー、グループ、ロールなど) に許可するにはアクセス許可を設定する必要があります。詳細についてはIAM ユーザーガイド の「サービスにリンクされた役割のアクセス許可」を参照してください。

Amazon FSx のサービスリンクロールの作成

サービスリンクロールを手動で作成する必要はありません。 AWS マネジメントコンソール、、AWS CLIまたは AWS API でファイルシステムを作成すると、Amazon FSx によってサービスにリンクされたロールが作成されます。

> **Important**
>
> このサービスリンクロールは、このロールでサポートされている機能を使用する別のサービスでアクションが完了した場合にアカウントに表示されます。詳細については、「IAM アカウントに新しいロールが表示される」を参照してください。

このサービスリンクロールを削除した後で再度作成する必要が生じた場合は、同じ手順でアカウントにロールを再作成できます。サービスリンクロールは、ファイルシステムの作成時に Amazon FSx で自動的に再作成されます。

Amazon FSx のサービスにリンクされたロールの編集

Amazon FSx では、サービスにリンクされたロールを編集することはできません。サービスリンクロールの作成後は、さまざまなエンティティがロールを参照する可能性があるため、ロール名を変更することはできません。ただし、IAM を使用してロールの説明を編集することはできます。詳細については、「IAM ユーザーガイド」の「サービスリンクロールの編集」を参照してください。

Amazon FSx のサービスリンクロールの削除

サービスにリンクされたロールを必要とする機能またはサービスが不要になった場合には、そのロールを削除することをお勧めします。これにより、積極的にモニタリングまたは保守されない未使用のエンティティを排除できます。ただし、サービスにリンクされたロールを手動で削除する前に、すべての ファイルシステムおよびバックアップを削除する必要があります。

> **Note**
>
> リソースを削除しようとしたときに Amazon FSx サービスがロールを使用している場合は、削除が失敗する可能性があります。その場合は、数分待ってからオペレーションを再試行してください。

IAM を使用してサービスリンクロールを手動で削除するには

IAM コンソール、IAM CLI、または IAM API を使用して、AWSServiceRoleForAmazonFSx サービスにリンクされたロールを削除します。詳細については、「IAM ユーザーガイド」の「サービスにリンクされたロールの削除」を参照してください。

Amazon FSx サービスリンクロールがサポートされるリージョン

Amazon FSx は、サービスが利用可能なすべてのリージョンでサービスにリンクされたロールの使用をサポートします。詳細については、「AWS リージョンとエンドポイント」を参照してください。

Amazon VPC を使用したファイルシステムアクセスコントロール

Amazon FSx ファイルシステムは、ファイルシステムに関連付ける Amazon VPC サービスに基づいて仮想プライベートクラウド (VPC) 内に存在する Elastic Network Interface を通してアクセスできます。Amazon FSx ファイルシステムにアクセスするには、ファイルシステムのネットワークインターフェイスにマッピングされる DNS 名を使用します。関連付けられた VPC 内のリソースまたはピアリングされた VPC のみが、ファイルシステムのネットワークインターフェイスにアクセスできます。詳細については、「Amazon VPC ユーザーガイド」の「Amazon VPC とは」を参照してください。

> **Warning**
>
> Amazon FSx Elastic Network Interface のネットワークインターフェイスを変更または削除しないでください。このネットワークインターフェイスを変更または削除すると、VPC とファイルシステムとの間の接続が完全に失われる可能性があります。

Amazon VPC セキュリティグループ

VPC 内のファイルシステムのネットワークインターフェイスを通過するネットワークトラフィックをさらにコントロールするには、セキュリティグループを使用してファイルシステムへのアクセスを制限します。セキュリティグループ は、仮想ファイアウォールとして機能し、関連付けられたインスタンスへのトラフィックを管理します。この場合、関連付けられたリソースはファイルシステムのネットワークインターフェイスです。VPC セキュリティグループを使用して Lustre クライアントのネットワークトラフィックをコントロールします。

EFA 対応セキュリティグループ

EFA 対応 FSx for Lustre を作成する場合は、まず EFA 対応セキュリティグループを作成し、ファイルシステムのセキュリティグループとして指定する必要があります。EFA にはセキュリティグルー

プ自体とのインバウンドおよびアウトバウンドのトラフィックをすべて許可するセキュリティグループと、クライアントが異なるセキュリティグループにある場合は、クライアントのセキュリティグループが必要です。詳細については、Amazon EC2 ユーザーガイド の ステップ 1: EFA 対応のセキュリティグループを準備する を参照してください。

インバウンドルールとアウトバンドルールを使用したアクセスのコントロール

セキュリティグループを使用して Amazon FSx ファイルシステムと Lustre クライアントへのアクセスをコントロールするには、インバウンドルール、およびファイルシステムと Lustre クライアントから送信されるトラフィックをコントロールするアウトバンドルールを追加します。Amazon FSx ファイルシステムのファイル共有を、サポートされているコンピューティングインスタンス上のフォルダーにマッピングするために、セキュリティグループに適切なネットワークトラフィックルールがあることを確認します。

セキュリティグループルールの詳細については、Amazon EC2 ユーザーガイドの「セキュリティグループルール」を参照してください。

Amazon FSx ファイルシステムのセキュリティグループを作成するには

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、[セキュリティグループ] を選択します。
3. [Create Security Group] (セキュリティグループの作成) を選択します。
4. セキュリティグループの名前と説明を指定します。
5. VPC については、Amazon FSx ファイルシステムに関連付けられている VPC を選択し、その VPC 内にセキュリティグループを作成します。
6. [Create] (作成) を選択して、セキュリティグループを作成します。

次に、作成したセキュリティグループにインバウンドルールを追加して、FSx for Lustre ファイルサーバー間の Lustre トラフィックを有効にします。

セキュリティグループへのインバウンドルールの追加

1. 作成したセキュリティグループが選択されていない場合は、そのセキュリティグループを選択します。[Actions] (アクション) メニューで、[Edit inbound rules] (インバウンドルールの編集) を選択します。
2. 次のインバウンドルールを追加します。

タイプ	プロトコル	ポート範囲	ソース	説明
カスタム TCP ルール	TCP	988	[Custom] (カスタム) を選択して、作成したセキュリティグループのセキュリティグループ ID を入力します。	FSx for Lustre ファイルサーバ間の Lustre トラフィックを許可します
カスタム TCP ルール	TCP	988	[カスタム] を選択して、Lustre クライアントに関連付けられたセキュリティグループのセキュリティグループ ID を入力します。	FSx for Lustre ファイルサーバーと Lustre クライアント間の Lustre トラフィックを許可します
カスタム TCP ルール	TCP	1018-1023	[Custom] (カスタム) を選択して、作成したセキュリティグループのセキュリティグループ ID を入力します。	FSx for Lustre ファイルサーバ間の Lustre トラフィックを許可します

タイプ	プロトコル	ポート範囲	ソース	説明
カスタム TCP ルール	TCP	1018-1023	[カスタム] を選択して、Lustre クライアントに関連付けられたセキュリティグループのセキュリティグループ ID を入力します。	FSx for Lustre ファイルサーバーと Lustre クライアント間の Lustre トラフィックを許可します

3. [Save] (保存) をクリックして、新しいインバウンドルールを保存して適用します。

デフォルトでは、セキュリティグループルールは、すべてのアウトバウンドトラフィック (すべて、0.0.0.0/0) を許可します。セキュリティグループがすべてのアウトバウンドトラフィックを許可していない場合は、次のアウトバウンドルールをセキュリティグループに追加します。ルールでは、FSx for Lustre ファイルサーバーと Lustre クライアント間、および Lustre ファイルサーバー間のトラフィックが許可されます。

セキュリティグループにアウトバウンドルールを追加するには

1. インバウンドルールを追加したのと同じセキュリティグループを選択します。[Actions] (アクション) メニューで、[Edit outbound rules] (アウトバウンドルールの編集) を選択します。
2. 次のアウトバウンドルールを追加します。

タイプ	プロトコル	ポート範囲	ソース	説明
カスタム TCP ルール	TCP	988	[Custom] (カスタム) を選択して、作成したセキュリティグループのセキュリティグループ ID を入力します。	FSx for Lustre ファイルサーバ間の Lustre トラフィックを許可する

タイプ	プロトコル	ポート範囲	ソース	説明
カスタム TCP ルール	TCP	988	[カスタム] を選択して、Lustre クライアントに関連付けられたセキュリティグループのセキュリティグループ ID を入力します。	FSx for Lustre ファイルサーバーと Lustre クライアント間の Lustre トラフィックを許可します
カスタム TCP ルール	TCP	1018-1023	[Custom] (カスタム) を選択して、作成したセキュリティグループのセキュリティグループ ID を入力します。	FSx for Lustre ファイルサーバ間の Lustre トラフィックを許可します
カスタム TCP ルール	TCP	1018-1023	[カスタム] を選択して、Lustre クライアントに関連付けられたセキュリティグループのセキュリティグループ ID を入力します。	FSx for Lustre ファイルサーバーと Lustre クライアント間の Lustre トラフィックを許可します

3. [Save] (保存) を選択して、新しいアウトバウンドルールを保存して適用します。

Amazon FSx ファイルシステムに関連付けられているセキュリティグループを関連付けるには

1. https://console.aws.amazon.com/fsx/で Amazon FSx コンソールを開きます。
2. コンソールダッシュボードで、ファイルシステムを選択して詳細を表示します。

3. [ネットワークとセキュリティ] タブで、[ネットワークインターフェイス] の下にある [Amazon EC2 コンソール] リンクをクリックし、ファイルシステムのすべてのネットワークインターフェイスを表示します。
4. ネットワークインターフェイスごとに、[アクション] で [セキュリティグループを変更] を選択します。
5. [セキュリティグループの変更] ダイアログボックスで、ネットワークインターフェイスに関連付けるセキュリティグループを選択します。
6. [保存] を選択します。

Lustre クライアント VPC セキュリティグループのルール

VPC セキュリティグループを使用して、Lustre クライアントへのアクセスをコントロールします。これには、Lustre クライアントから送信されるトラフィックをコントロールするインバウンドルール、およびアウトバンドルールを追加します。Lustre トラフィックが Lustre クライアントと Amazon FSx ファイルシステム間を流れることができるように、セキュリティグループに適切なネットワークトラフィックルールがあることを確認してください。

Lustre クライアントに適用されるセキュリティグループに、次のインバウンドルールを追加します。

タイプ	プロトコル	ポート範囲	ソース	説明
カスタム TCP ルール	TCP	988	[カスタム] を選択して、Lustre クライアントに適用されたセキュリティグループのセキュリティグループ ID を入力します。	Lustre クライアント間の Lustre トラフィックを許可する
カスタム TCP ルール	TCP	988	[Custom] (カスタム) を選択して、FSx for Lustre ファイルシステムに関連	FSx for Lustre ファイルサーバーと Lustre クライアント間の Lustre トラ

タイプ	プロトコル	ポート範囲	ソース	説明
			付けられたセキュリティグループのセキュリティグループ ID を入力します。	フィックを許可します
カスタム TCP ルール	TCP	1018-1023	[カスタム] を選択して、Lustre クライアントに適用されたセキュリティグループのセキュリティグループ ID を入力します。	Lustre クライアント間の Lustre トラフィックを許可する
カスタム TCP ルール	TCP	1018-1023	[Custom] (カスタム) を選択して、FSx for Lustre ファイルシステムに関連付けられたセキュリティグループのセキュリティグループ ID を入力します。	FSx for Lustre ファイルサーバーと Lustre クライアント間の Lustre トラフィックを許可します

Lustre クライアントに適用されるセキュリティグループに、次のアウトバウンドルールを追加します。

タイプ	プロトコル	ポート範囲	ソース	説明
カスタム TCP ルール	TCP	988	[カスタム] を選択して、Lustre クライアン	Lustre クライアント間の Lustre

タイプ	プロトコル	ポート範囲	ソース	説明
			トに適用されたセキュリティグループのセキュリティグループ ID を入力します。	トラフィックを許可する
カスタム TCP ルール	TCP	988	[Custom] (カスタム) を選択して、FSx for Lustre ファイルシステムに関連付けられたセキュリティグループのセキュリティグループ ID を入力します。	FSx for Lustre ファイルサーバーと Lustre クライアント間の Lustre トラフィックを許可します
カスタム TCP ルール	TCP	1018-1023	[カスタム] を選択して、Lustre クライアントに適用されたセキュリティグループのセキュリティグループ ID を入力します。	Lustre クライアント間の Lustre トラフィックを許可する

タイプ	プロトコル	ポート範囲	ソース	説明
カスタム TCP ルール	TCP	1018-1023	[Custom] (カスタム) を選択して、FSx for Lustre ファイルシステムに関連付けられたセキュリティグループのセキュリティグループ ID を入力します。	FSx for Lustre ファイルサーバーと Lustre クライアント間の Lustre トラフィックを許可します

Amazon VPC ネットワーク ACL

VPC 内のファイルシステムへのアクセスを保護するためのもう 1 つのオプションは、ネットワークアクセスコントロールリスト (ネットワーク ACL) を確立することです。ネットワーク ACL はセキュリティグループとは別のものですが、VPC のリソースにセキュリティのレイヤーを追加するための同様の機能があります。ネットワーク ACL を使用したアクセスコントロールの実装の詳細については、「Amazon VPC ユーザーガイド」の「ネットワーク ACL を使用してサブネットへのトラフィックを制御する」を参照してください。

Amazon FSx for Lustre のコンプライアンス検証

AWS のサービス が特定のコンプライアンスプログラムの対象であるかどうかを確認するには、「コンプライアンスプログラムによる対象範囲内の AWS のサービス」をご覧いただき、関心のあるコンプライアンスプログラムを選択してください。一般的な情報については、「AWSコンプライアンスプログラム」を参照してください。

AWS Artifact を使用して、サードパーティーの監査レポートをダウンロードできます。詳細については、「AWS Artifact でレポートをダウンロードする」を参照してください。

AWS のサービス を使用する際のお客様のコンプライアンス責任は、お客様のデータの機密性や貴社のコンプライアンス目的、適用可能な法律および規制によって決定されます。AWS のサービスを使用する際のコンプライアンス責任の詳細については、「AWS セキュリティドキュメント」を参照してください。

Amazon FSx for Lustre とインターフェイス VPC エンドポイント (AWS PrivateLink)

インターフェイス VPC エンドポイントを使用するように Amazon FSx を設定することで、VPC のセキュリティ体制を強化できます。インターフェイス VPC エンドポイントは、インターネットゲートウェイ、NAT デバイス、VPN 接続、Direct Connect 接続のいずれも必要とせずに Amazon FSx API にプライベートにアクセスできるテクノロジー、AWS PrivateLink を利用しています。VPC のインスタンスは、パブリック IP アドレスがなくても Amazon FSx API と通信できます。VPC と Amazon FSx 間のトラフィックは、AWS ネットワークを離れません。

各インターフェイス VPC エンドポイントは、サブネット内の 1 つ以上の Elastic Network Interface によって表されます。ネットワークインターフェイスは、Amazon FSx API へのトラフィックのエントリポイントとなるプライベート IP アドレスを提供します。

Amazon FSx インターフェイス VPC エンドポイントに関する考慮事項

Amazon FSx のインターフェイス VPC エンドポイントを設定する前に、「Amazon VPC ユーザーガイド」の「インターフェイス VPC エンドポイントのプロパティと制限」を確認してください。

VPC から任意の Amazon FSx API オペレーションを呼び出すことができます。例えば、VPC 内で CreateFileSystem API を呼び出すことで、FSx for Lustre ファイルシステムを作成することができます。Amazon FSx API の詳細なリストについては、「Amazon FSx API Reference」(Amazon FSx API リファレンス) の「Actions」(アクション) を参照してください。

VPC ピアリングに関する考慮事項

他の VPC には、インターフェイス VPC エンドポイントを使用して、VPC ピアリングによって接続できます。VPC ピアリングは2 つの VPC 間のネットワーク接続です。自分が所有者である 2 つの VPC 間や、他の AWS アカウント アカウント内の VPC との間で、VPC ピアリング接続を確立できます。VPC は 2 つの異なる AWS リージョン の間でも使用できます。

ピア接続された VPC 間のトラフィックは AWS ネットワーク上に留まり、パブリックインターネットを経由しません。VPC がピア接続されると、双方の VPC にある Amazon Elastic Compute Cloud (Amazon EC2) インスタンスは、いずれかの VPC で作成されたインターフェイス VPC エンドポイントを介して Amazon FSx API にアクセスできます。

Amazon FSx API 用のインターフェイス VPC エンドポイントの作成

Amazon FSx API 用の VPC エンドポイントは、Amazon VPC コンソールまたは AWS Command Line Interface (AWS CLI) で作成できます。詳細については、Amazon VPC ユーザーガイド の インターフェイス VPC エンドポイントの作成 をご参照ください。

Amazon FSx エンドポイントの完全なリストについては、「Amazon Web Services 全般のリファレンス」の「Amazon FSx エンドポイントとクォータ」を参照してください。

Amazon FSx のインターフェイス VPC エンドポイントを作成するには、次のいずれかを使用します。

- **com.amazonaws.*region*.fsx** – Amazon FSx API オペレーションのエンドポイントを作成します。
- **com.amazonaws.*region*.fsx-fips** — 連邦情報処理規格 (FIPS) 140-2 に準拠した Amazon FSx API のエンドポイントを作成します。

オプションとしてプライベート DNS を使用するには、VPC の `enableDnsHostnames` および `enableDnsSupport` 属性を設定する必要があります。詳細については、Amazon VPC ユーザーガイド の VPC の DNS サポートを表示および更新する を参照してください。

中国の AWS リージョン を除き、エンドポイントでプライベート DNS を有効にすると、AWS リージョン のデフォルト DNS 名 (`fsx.us-east-1.amazonaws.com` など) を使用して、VPC エンドポイントで Amazon FSx に API リクエストを行うことができます。中国 (北京) および 中国 (寧夏) AWS リージョン の場合、それぞれ `fsx-api.cn-north-1.amazonaws.com.cn` および `fsx-api.cn-northwest-1.amazonaws.com.cn` を使用して VPC エンドポイントで API リクエストを行うことができます。

詳細については、Amazon VPC ユーザーガイドの「インターフェイス VPC エンドポイントを介したサービスへのアクセス」をご参照ください。

Amazon FSx 用の VPC エンドポイントポリシーの作成

Amazon FSx API へのアクセスをさらに制御するために VPC エンドポイントに AWS Identity and Access Management (IAM) ポリシーをアタッチすることも可能です。本ポリシーでは、以下を規定します。

- アクションを実行できるプリンシパル。
- 実行可能なアクション。

- このアクションを実行できるリソース。

詳細については、「Amazon VPC ユーザーガイド」の「VPC エンドポイントでサービスへのアクセスを制御する」を参照してください。

Amazon FSx for Lustre の Service quotas

Amazon FSx for Lustre を使用する際のクォータについて以下に説明します。

トピック

増やすことができるクォータ

増やすことができる Amazon FSx for Lustre の AWS アカウントあたり、AWS リージョンあたりのクォータは次のとおりです。

リソース	デフォルト	説明
Lustre 永続 1 ファイルシステム	100	このアカウントで作成できる Amazon FSx for Lustre 永続 1 ファイルシステムの最大数。
Lustre 永続 2 ファイルシステム	100	このアカウントで作成できる Amazon FSx for Lustre 永続 2 ファイルシステムの最大数。
Lustre Persistent HDD ストレージ容量 (ファイルシステムあたり)	102000	Amazon FSx for Lustre 永続的ファイルシステムに設定できる HDD ストレージ容量 (GiB 単位) の最大容量。
Lustre 永続 1 ファイルストレージ容量	100800	このアカウントで設定できる、すべての Amazon FSx for Lustre Persistent 1 ファイルシステムの最大ストレージ容量 (GiB 単位)です。
Lustre 永続 2 ファイルストレージ容量	100800	このアカウントで設定できる、すべての Amazon FSx for

リソース	デフォルト	説明
		Lustre Persistent 2 ファイルシステムの最大ストレージ容量 (GiB 単位)です。
Lustre スクラッチファイルシステム	100	このアカウントで作成できる Amazon FSx for Lustre スクラッチ ファイルシステムの最大数。
Lustre スクラッチストレージ容量	100800	このアカウントのすべての Amazon FSx for Lustre スクラッチ ファイルシステムに設定できるストレージ容量 (GiB 単位) の最大容量。
Lustre 永続 インテリジェント階層化 スループットキャパシティ	100000	このアカウントで許可されている、すべての Amazon FSx for Lustre インテリジェント階層化 ファイルシステムの合計スループットキャパシティ (MBps 単位)です。
Lustre 永続 インテリジェント階層化 SSD 読み取りキャッシュストレージ容量	100800	このアカウントで設定できる、すべての Amazon FSx for Lustre インテリジェント階層化 ファイルシステムに対するプロビジョニング済み SSD 読み取りキャッシュの最大ストレージ容量(GiB 単位)です。
Lustre バックアップ	500	このアカウントのすべての Amazon FSx for Lustre ファイルシステムに対して保持できるユーザー主導バックアップの最大数。

クォータの増加をリクエストするには

1. Service Quotas コンソール を開きます。
2. ナビゲーションペインで、[AWSサービス] を選択します。
3. を選択してください。。Lustre
4. クォータを選択します。
5. [Request quota increase] (クォータ引き上げリクエスト) を選択して、指示に従ってクォータの引き上げをリクエストします。
6. クォータリクエストのステータスを表示するには、コンソールのナビゲーションペインの [Quota request history] (クォータ依頼履歴) を選択します。

詳細については、「Service Quotas ユーザーガイド」の「クォータ引き上げのリクエスト」を参照してください。

ファイルシステムあたりのリソースクォータ

以下に、AWS リージョンでの各ファイルシステムの Amazon FSx for Lustre リソースに対する制限を示します。

リソース	ファイルシステムあたりの制限
タグの最大数	50
自動バックアップの最大保持期間	90 日間
単一の宛先リージョンに対して同時に送信できるバックアップコピーリクエストの 1 アカウントあたりの最大数。	5
ファイルシステムごとのリンクされた S3 バケットからのファイル更新数	1000 万 / 月
最小ストレージ容量、SSD ファイルシステム	1.2 TiB
最小ストレージ容量、HDD ファイルシステム	6 TiB
ストレージ単位あたりの最小スループット、SSD	50 MBps

リソース	ファイルシステムあたりの制限
ストレージ単位あたりの最大スループット、SSD	1,000 MBps
ストレージ単位あたりの最小スループット、HDD	12 MBps
ストレージ単位あたりの最大スループット、HDD	40 MBps

追加の考慮事項

さらに、以下の点にも注意してください。

- 最大 125 の Amazon FSx for Lustre ファイルシステムに対して AWS Key Management Service (AWS KMS) キーのそれぞれを使用できます。
- ファイルシステムを作成できる AWS リージョンのリストについては、「AWS 全般のリファレンス」の「Amazon FSx エンドポイントとクォータ」を参照してください。

Amazon FSx for Lustre のトラブルシューティング

このセクションでは、Amazon FSx for Lustre ファイルシステムのさまざまなトラブルシューティングシナリオとソリューションについて説明します。

以下に記載されていない問題が発生した場合は、Amazon FSx for Lustre フォーラムで質問してみてください。

トピック

FSx for Lustre ファイルシステムの作成に失敗する

ファイルシステムの作成リクエストが失敗する場合、次のトピックで説明するように、いくつかの原因が考えられます。

セキュリティグループの設定不備により、EFA 有効化ファイルシステムを作成できません

FSx for Lustre EFA 対応ファイルシステムの作成が失敗し、以下のエラーメッセージが表示されます:

```
Insufficient security group permissions to create an EFA-enabled file system.
Update security group to allow all internal inbound and outbound traffic.
```

実行するアクション

作成操作に使用する VPC セキュリティグループが、「EFA 対応セキュリティグループ」で説明されているとおりに設定されていることを確認してください。EFA にはセキュリティグループ自体とのインバウンドおよびアウトバウンドのトラフィックをすべて許可するセキュリティグループと、クライアントが異なるセキュリティグループにある場合は、クライアントのセキュリティグループが必要です。

セキュリティグループの設定が間違っているため、ファイルシステムを作成できない

FSx for Lustre ファイルシステムの作成が失敗し、次のエラーメッセージが表示されます。

```
The file system cannot be created because the default security group in the subnet
 provided
or the provided security groups do not permit Lustre LNET network traffic on port 988
```

実行するアクション

作成操作に使用する VPC セキュリティグループが、「Amazon VPC を使用したファイルシステムアクセスコントロール」で説明されているとおりに設定されていることを確認してください。セキュリティグループを設定して、ポート 988 および 1018 ~ 1023 で、セキュリティグループ自体またはフルサブネット CIDR からのインバウンドトラフィックを許可する必要があります。これは、ファイルシステムホストが相互に通信できるようにするために必要です。

容量不足エラーによりファイルシステムを作成できません

新しいファイルシステムの作成、スループットキャパシティの更新、スループットキャパシティの変更を試みると、容量不足エラーが発生することがあります。

原因:

このエラーは、リクエストされたアベイラビリティーゾーンにおいて FSx for Lustre がリクエストを満たすために現在十分な利用可能なハードウェア容量を持っていない場合に発生します。

解決策:

この問題を解決するには、以下の手順を実行します。

- 容量の利用可能状況は頻繁に変動しますので、数分後に再度リクエストしてください。
- 別のアベイラビリティーゾーンでリクエストを試してください。

- 小さいストレージサイズまたは低いスループットレベルでオペレーションを試行する

S3 バケットにリンクされたファイルシステムを作成できません。

S3 バケットにリンクされた新しいファイルシステムを作成すると、次のようなエラーメッセージが表示されて失敗します。

```
User: arn:aws:iam::012345678901:user/username is not authorized to perform:
 iam:PutRolePolicy on resource: resource ARN
```

このエラーは、必要な IAM アクセス許可なしで Simple Storage Service (Amazon S3) バケットにリンクされたファイルシステムを作成しようとした場合に発生する可能性があります。必要な IAM アクセス許可は、ユーザーに代わって指定された Simple Storage Service (Amazon S3) バケットにアクセスするために使用される Amazon FSx for Lustre サービスにリンクされたロールをサポートします。

実行するアクション

IAM エンティティ (ユーザー、グループ、またはロール) にファイルシステムを作成するための適切なアクセス許可があることを確認します。これには、Amazon FSx for Lustre サービスにリンクされたロールをサポートするアクセス許可ポリシーの追加が含まれます。詳細については、「Simple Storage Service (Amazon S3) でデータリポジトリを使用する許可を追加する」を参照してください。

サービスにリンクされたロールの詳細については、「Amazon FSx のサービスリンクロールの使用」を参照してください。

ファイルシステムのマウントに関する問題のトラブルシューティング

ファイルシステムのマウントコマンドが失敗する場合、次のトピックで説明するように、いくつかの原因が考えられます。

ファイルシステムのマウントがすぐに失敗する

ファイルシステムのマウントコマンドはすぐに失敗します。コードの例を以下に示します。

```
mount.lustre: mount fs-0123456789abcdef0.fsx.us-east-1.aws@tcp:/fsx at /lustre
```

```
failed: No such file or directory

Is the MGS specification correct?
Is the filesystem name correct?
```

このエラーは、mount コマンドを使用してパーシステントまたはスクラッチ 2 ファイルシステムをマウントするときの正しい mountname 値を使用していない場合に発生する可能性があります。mountname値は、describe-file-systems AWS CLIコマンドまたは DescribeFileSystemsAPI 操作の応答から取得できます。

ファイルシステムのマウントがハングした後、タイムアウトエラーで失敗する

ファイルシステムのマウントコマンドが 1、2 分間ハングし、タイムアウトエラーで失敗します。

次のコードは例を示しています。

```
sudo mount -t lustre file_system_dns_name@tcp:/mountname /mnt/fsx

[2+ minute wait here]
Connection timed out
```

このエラーは、Amazon EC2 インスタンスまたはファイルシステムのセキュリティグループが正しく設定されていないために発生する可能性があります。

実行するアクション

ファイルシステムのセキュリティグループに、Amazon VPC セキュリティグループ で指定したインバウンドルールがあることを確認します。

自動マウントが失敗してインスタンスがレスポンスしない

場合によっては、ファイルシステムの自動マウントが失敗し、Amazon EC2 インスタンスがレスポンスしなくなる場合があります。

この問題は、_netdev オプションは宣言されていません。_netdev が見つからない場合、Amazon EC2 インスタンスはレスポンスを停止する可能性があります。この結果は、コンピューティングインスタンスがネットワークを開始後、ネットワークファイルシステムを初期化する必要があるためです。

実行するアクション

この問題が発生した場合は、AWS サポート にお問い合わせください。

システムのブート中にファイルシステムのマウントが失敗する

ファイルシステムのマウントは、システムのブート中に失敗します。マウントは、/etc/fstab を使用してオートメーション化されています。ファイルシステムがマウントされていない場合、インスタンスの起動時間枠の syslog に次のエラーが表示されます。

```
LNetError: 3135:0:(lib-socket.c:583:lnet_sock_listen()) Can't create socket: port 988
 already in use
LNetError: 122-1: Can't start acceptor on port 988: port already in use
```

このエラーは、ポート 988 が使用できない場合に発生することがあります。インスタンスが NFS ファイルシステムをマウントするように設定されている場合、NFS マウントがクライアントポートをポート 988 にバインドする可能性があります。

実行するアクション

可能な場合は、NFS クライアントの noresvport および noauto マウントオプションをチューニングすることで、この問題を回避できます。

DNS 名を使用したファイルシステムのマウントが失敗する

次のシナリオに示すように、ドメインネームサービス (DNS) 名の設定が間違っていると、ファイルシステムのマウントエラーが発生する可能性があります。

シナリオ 1: ドメインネームサービス (DNS) 名を使用しているファイルシステムのマウントが失敗します。次のコードは例を示しています。

```
sudo mount -t lustre file_system_dns_name@tcp:/mountname /mnt/fsx
mount.lustre: Can't parse NID
'file_system_dns_name@tcp:/mountname'
```

実行するアクション

仮想プライベートクラウド (VPC) の設定を確認します。カスタム VPC を使用している場合は、DNS 設定が有効であることを確認します。詳細については、「Amazon VPC ユーザーガイド」の「VPC での DNS の使用」を参照してください。

mount コマンドで DNS 名を指定するには、以下を実行する必要があります:

- Amazon EC2 インスタンスが Amazon FSx for Lustre ファイルシステムと同じ VPC 内にあることを確認します。
- Amazon が提供する DNS サーバーを使用するように設定された VPC 内でAmazon EC2 インスタンスを接続します。詳細については、Amazon VPC ユーザーガイド の「DHCP オプション設定」を参照してください。
- 接続する Amazon EC2 インスタンスの Amazon VPC で、DNS ホスト名が有効であることを確認します。詳細については、Amazon VPC ユーザーガイド の「VPC の DNS サポートを更新する」を参照してください。

シナリオ 2: ドメインネームサービス (DNS) 名を使用しているファイルシステムのマウントが失敗します。次のコードは例を示しています。

```
mount -t lustre file_system_dns_name@tcp:/mountname /mnt/fsx
mount.lustre: mount file_system_dns_name@tcp:/mountname at /mnt/fsx failed: Input/
output error Is the MGS running?
```

実行するアクション

クライアントの VPC セキュリティグループに、正しいアウトバウンドトラフィックルールが適用されていることを確認します。このレコメンデーションは、特にデフォルトのセキュリティグループを使用していない場合、またはデフォルトのセキュリティグループを変更した場合に当てはまります。詳細については、「Amazon VPC セキュリティグループ」を参照してください。

ファイルシステムにアクセスできない

次のように、ファイルシステムにアクセスできない原因はいくつか考えられますが、それぞれ独自の解像度があります。

ファイルシステムの Elastic Network Interface に接続されている Elastic IP アドレスが削除されました

Amazon FSx は、公開インターネットからのファイルシステムへのアクセスをサポートしていません。Amazon FSx は、インターネットから到達可能なパブリック IP アドレスである Elastic IP アドレスを自動的にデタッチし、ファイルシステムの Elastic Network Interface に接続します。

ファイルシステムの Elastic Network Interface が変更または削除されました

ファイルシステムの Elastic Network Interface 変更または削除しないでください。このネットワークインターフェイスを変更または削除すると、VPC とファイルシステムとの間の接続が完全に失われる可能性があります。新しいファイルシステムを作成し、FSx Elastic Network Interface は変更または削除しないでください。詳細については、「Amazon VPC を使用したファイルシステムアクセスコントロール」を参照してください。

DRA を作成するときに S3 バケットへのアクセスを検証できない

Amazon FSx コンソールから、または `create-data-repository-association` CLI コマンド (CreateDataRepositoryAssociation は同等の API アクション) を使用してデータリポジトリアソシエーション (DRA) を作成すると、次のエラーメッセージが表示されて失敗します。

```
Amazon FSx is unable to validate access to the S3 bucket. Ensure the IAM role or user
you are using has s3:Get*, s3:List* and s3:PutObject permissions to the S3 bucket
 prefix.
```

> **Note**
>
> Amazon FSx コンソールまたは `create-file-system` CLI コマンド (CreateFileSystem) は同等の API アクション) を使用してデータリポジトリ (S3 バケットまたはプレフィックス) にリンクされているスクラッチ 1、スクラッチ 2、または 永続 1 ファイルシステムを作成するときにも上記のエラーが発生する可能性があります。

実行するアクション

FSx for Lustre ファイルシステムが S3 バケットと同じアカウントにある場合、このエラーは、作成リクエストに使用した IAM ロールに S3 バケットへのアクセスに必要なアクセス許可がないことを意味します。IAM ロールに、エラーメッセージにリストされたアクセス許可があることを確認します。許可は、ユーザーに代わって指定された Simple Storage Service (Amazon S3) バケットにアクセスするために使用される Amazon FSx for Lustre サービスにリンクされたロールをサポートします。

FSx for Lustre ファイルシステムが S3 バケットとは異なるアカウントにある場合 (クロスアカウントの場合)、使用した IAM ロールに必要なアクセス許可があることを確認するだけでなく、FSx for

Lustre が作成されるアカウントからのアクセスを許可するように S3 バケットポリシーを設定する必要があります。

S3 クロスアカウントバケットパーミッションの詳細については、Amazon Simple Storage Service ユーザーガイド の 例 2: クロスアカウントバケットパーミッションを付与するバケット所有者 を参照してください。

ディレクトリの名前変更に長い時間がかかる

質問

Amazon S3 バケットにリンクされているファイルシステム上のディレクトリの名前を変更し、自動エクスポートを有効にしました。このディレクトリ内のファイルが S3 バケットで名前変更されるのに長い時間がかかるのはなぜですか?

回答

ファイルシステム上のディレクトリの名前を変更すると、FSx for Lustre は、名前が変更されたディレクトリ内のすべてのファイルとディレクトリに対して新しい S3 オブジェクトを作成します。ディレクトリの名前変更を S3 に伝播するのにかかる時間は、名前が変更されるディレクトリの子孫であるファイルとディレクトリの量に直接関係します。

正しく設定されていないリンクされた S3 バケットのトラブルシューティング

場合によっては、FSx for Lustre ファイルシステムのリンク S3 バケットのデータリポジトリのライフサイクル状態が誤って設定されている可能性があります。

考えられる原因

このエラーは、リンクされたデータリポジトリへのアクセスに必要な AWS Identity and Access Management (IAM) アクセス許可を、Amazon FSx が必要としない場合に発生する可能性があります。必要な IAM アクセス許可は、ユーザーに代わって指定された Simple Storage Service (Amazon S3) バケットにアクセスするために使用される Amazon FSx for Lustre サービスにリンクされたロールをサポートします。

実行するアクション

1. IAM エンティティ (ユーザー、グループ、またはロール) にファイルシステムを作成するための適切なアクセス許可があることを確認します。これには、Amazon FSx for Lustre サービスにリンクされたロールをサポートするアクセス許可ポリシーの追加が含まれます。詳細については、「Simple Storage Service (Amazon S3) でデータリポジトリを使用する許可を追加する」を参照してください。
2. Amazon FSx CLI または API を使用して、次のように update-file-system CLI コマンド (UpdateFileSystem は同等の API アクション) でファイルシステムの AutoImportPolicy を更新します。

```
aws fsx update-file-system \
--file-system-id fs-0123456789abcdef0 \
--lustre-configuration AutoImportPolicy=the_existing_AutoImportPolicy
```

サービスにリンクされたロールの詳細については、「Amazon FSx のサービスリンクロールの使用」を参照してください。

考えられる原因

このエラーは、リンクされた Simple Storage Service (Amazon S3) データリポジトリに、Amazon FSx イベント通知設定 (s3:ObjectCreated:*、s3:ObjectRemoved:*) と重複するイベントタイプを持つ既存のイベント通知設定がある場合に発生する可能性があります。

これは、リンクされた S3 バケットの Amazon FSx イベント通知設定が削除または変更された場合にも発生します。

実行するアクション

1. FSx イベント設定が使用する s3:ObjectCreated:* および s3:ObjectRemoved:* のイベントタイプのいずれかまたは両方を使用するリンクされた S3 バケット上の既存のイベント通知を削除します。
2. リンクされた S3 バケットに、名前がFSx、イベントタイプがs3:ObjectCreated:*およびs3:ObjectRemoved:*の S3 イベント通知設定があることを確認し、ARN:*topic_arn_returned_in_API_response*を使用して SNS トピックに送信します。
3. Amazon FSx CLI または API を使用して、S3 バケットに FSx イベント通知設定を再適用し、ファイルシステムの AutoImportPolicy をリフレッシュさせます。次のように、update-file-system CLI コマンド (UpdateFileSystem) は同等の API アクションです) を使用してこれを行います。

```
aws fsx update-file-system \
--file-system-id fs-0123456789abcdef0 \
--lustre-configuration AutoImportPolicy=the_existing_AutoImportPolicy
```

ストレージ問題のトラブルシューティング

場合によっては、ファイルシステムのストレージの問題が発生することがあります。問題は、`lfs migrate` コマンドなどの `lfs` コマンドを使用してトラブルシューティングできます。

ストレージターゲットにスペースがないことによる書き込みエラー

ファイルシステムストレージレイアウト で説明されているように、`lfs df -h` コマンドを使用して、ファイルシステムのストレージ使用量を確認できます。`filesystem_summary` フィールドには、ファイルシステムのストレージ使用量の合計が報告されます。

ファイルシステムのディスク使用率が 100% の場合は、ファイルシステムのストレージ容量を増やすことを検討してください。詳細については、「ストレージ容量の管理」を参照してください。

ファイルシステムのストレージ使用率が 100% でなくても、書き込みエラーが発生する場合は、書き込み先のファイルが、いっぱいの OST でストライプ化されている可能性があります。

実行するアクション

- 多くの OST がいっぱいになっている場合は、ファイルシステムのストレージ容量を増やしてください。OST 上のアンバランスストレージ セクションのアクションに従って、OST のアンバランスストレージをチェックします。
- OST がいっぱいになっていない場合は、すべてのクライアントインスタンスに次の調整を適用して、クライアントのダーティページのバッファサイズを調整します。

```
sudo lctl set_param osc.*.max_dirty_mb=64
```

OST 上のアンバランスストレージ

Amazon FSx for Lustre は、新しいファイルストライプを OST 全体に均等に分散します。ただし、I/O パターンまたはファイルストレージレイアウトが原因で、ファイルシステムのバランスが崩れる可能性があります。その結果、一部のストレージターゲットが満杯になり、他のストレージターゲットは比較的空のままになる可能性があります。

lfs migrate コマンドを使用して、ファイルやディレクトリを満杯の OST から空きのある OST に移動します。lfs migrate コマンドは、ブロックモードでも非ブロックモードでも使用できます。

- ブロックモードは、lfs migrate コマンドのデフォルトモードです。ブロックモードで実行すると、lfs migrate はデータの移行前にまずファイルおよびディレクトリのグループロックを取得してファイルへの変更を防ぎ、移行が完了するとロックを解除します。ブロックモードは、他のプロセスがファイルを変更できないようにすることで、これらのプロセスによって移行が中断されるのを防ぎます。このモードの欠点は、アプリケーションがファイルを変更できないようにすると、アプリケーションに遅延や エラーが発生する可能性があることです。
- 非ブロックモードは、lfs migrate コマンドで -n オプションを指定すると有効になります。非ブロックモードで lfs migrate を実行すると、他のプロセスでも移行中のファイルを変更できます。lfs migrate がファイルの移行を完了する前にプロセスがファイルを変更した場合、lfs migrate はそのファイルの移行に失敗し、ファイルは元のストライプレイアウトのままになります。

このコマンドはアプリケーションに干渉する可能性が低いため、非ブロックモードを使用することをお勧めします。

実行するアクション

1. 比較的大きなクライアントインスタンス (Amazon EC2 c5n.4xlarge インスタンスタイプなど) を起動して、ファイルシステムにマウントします。
2. 非ブロックモードスクリプトまたはブロックモードスクリプトを実行する前に、各クライアントインスタンスで次のコマンドを実行し、プロセスを高速化します。

   ```
   sudo lctl set_param 'mdc.*.max_rpcs_in_flight=60'
   sudo lctl set_param 'mdc.*.max_mod_rpcs_in_flight=59'
   ```

3. スクリーンセッションを開始し、非ブロックモードスクリプトまたはブロックモードスクリプトを実行します。スプリクト内の変数は必ず適切なものに変更してください。

   - 非ブロックモードスクリプト

     ```
     #!/bin/bash

     # UNCOMMENT THE FOLLOWING LINES:
     #
     # TRY_COUNT=0
     ```

```
# MAX_MIGRATE_ATTEMPTS=100
# OSTS="fsname-OST0000_UUID"
# DIR_OR_FILE_MIGRATED="/mnt/subdir/"
# BATCH_SIZE=10
# PARALLEL_JOBS=16 # up to max-procs processes, set to 16 if client is
 c5n.4xlarge with 16 vcpu
# LUSTRE_STRIPING_CONFIG="-E 100M -c 1 -E 10G -c 8 -E 100G -c 16 -E -1 -c 32" #
 should be consistent with the existing striping setup
#

if [ -z "$TRY_COUNT" -o -z "$MAX_MIGRATE_ATTEMPTS" -o -z "$OSTS" -o -z
 "$DIR_OR_FILE_MIGRATED" -o -z "$BATCH_SIZE" -o -z "$PARALLEL_JOBS" -o -z
 "$LUSTRE_STRIPING_CONFIG" ]; then
    echo "Some variables are not set."
    exit 1
fi

echo "lfs migrate starts"
while true; do
    output=$(sudo lfs find ! -L released --ost $OSTS --print0
 $DIR_OR_FILE_MIGRATED | shuf -z | /bin/xargs -0 -P $PARALLEL_JOBS -n $BATCH_SIZE
 sudo lfs migrate -n $LUSTRE_STRIPING_CONFIG 2>&1)
    if [[ $? -eq 0 ]]; then
        echo "lfs migrate succeeds for $DIR_OR_FILE_MIGRATED at the $TRY_COUNT
 attempt, exiting."
        exit 0
    elif [[ $? -eq 123 ]]; then
        echo "WARN: Target data objects are not located on these OSTs. Skipping
 lfs migrate"
        exit 1
    else
        echo "lfs migrate fails for $DIR_OR_FILE_MIGRATED at the $TRY_COUNT
 attempt, retrying..."
        if (( ++TRY_COUNT >= MAX_MIGRATE_ATTEMPTS )); then
            echo "WARN: Exceeds max retry attempt. Skipping lfs migrate for
 $DIR_OR_FILE_MIGRATED. Failed with the following error"
            echo $output
            exit 1
        fi
    fi
done
```

- ブロックモードスクリプト

- OSTS の値を OST の値に置き換えます。
- nproc に整数値を指定し、同時に実行する max-procs プロセスの数を設定します。例えば、Amazon EC2 c5n.4xlarge インスタンスタイプには 16 の vCPUs があるため、nproc には 16 (または 16 未満の値) を使用できます。
- mnt_dir_path にマウントディレクトリパスを指定します。

```
# find all OSTs with usage above a certain threshold; for example, greater than
 or equal to 85% full
for OST in $(lfs df -h |egrep '( 8[5-9]| 9[0-9]|100)%'|cut -d' ' -f1); do echo
 ${OST};done|tr '\012' ','

# customer can also just pass OST values directly to OSTS variable
OSTS='dzfevbmv-OST0000_UUID,dzfevbmv-OST0002_UUID,dzfevbmv-OST0004_UUID,dzfevbmv-
OST0005_UUID,dzfevbmv-OST0006_UUID,dzfevbmv-OST0008_UUID'

nproc=<Run up to max-procs processes if client is c5n.4xlarge with 16 vcpu, this
 value can be set to 16>

mnt_dir_path=<mount dir, e.g. '/my_mnt'>

lfs find ${mnt_dir_path} --ost ${OSTS}| xargs -P ${nproc} -n2 lfs migrate -E 100M
 -c 1 -E 10G -c 8 -E 100G -c 16 -E -1 -c 32
```

メモ

- ファイルシステムの読み取りパフォーマンスに影響が出ることに気付いた場合は、ctrl-c または kill -9 を使用していつでも移行を中止できます。スレッドの数を (nproc 値) を小さい数 (8 など) まで減らしたら、ファイルの移行を再開します。
- lfs migrate コマンドを実行すると、クライアントワークロードによっても開かれているファイルで失敗します。エラーをスローして次のファイルに移動します。したがって、アクセスされているファイルがたくさんある場合、スクリプトはファイルを移行させることができず、移行の進行が非常に遅くなるという形でそれが反映される可能性があります。
- OST の使用状況は、次のいずれかの方法でモニタリングできます。
  - クライアントマウントで、次のコマンドを実行して OST の使用状況をモニタリングし、使用率が 85% を超える OST を検索します。

```
lfs df -h |egrep '( 8[5-9]| 9[1-9]|100)%'
```

  - Amazon CloudWatch メトリクス、`OST FreeDataStorageCapacity` をチェックし、`Minimum` をチェックします。スクリプトが 85% を超える OST を検出すると、メトリクスが 15% に近づいたとき、`ctrl-c` または `kill -9` を使用して移行を停止します。
- また、新しいファイルが複数のストレージターゲットにストライプされるように、ファイルシステムまたはディレクトリのストライプ設定を変更することを検討することもできます。詳細については、「ファイルシステム内のデータのストライピング」を参照してください。

FSx for Lustre CSI ドライバーの問題のトラブルシューティング

Amazon FSx for Lustre は、オープンソースの FSx for Lustre CSI ドライバー を使用して、Amazon EKS で実行されているコンテナからのにアクセスをサポートしていします。デプロイの詳細については、「Amazon EKS ユーザーガイド」の「Amazon FSx for Lustre ストレージ」を参照してください。

Amazon EKS で実行されているコンテナの FSx for Lustre CSI ドライバーに関する問題が発生している場合は、GitHub にある「CSI ドライバーのトラブルシューティング (一般的な問題)」を参照してください。

追加情報

このセクションでは、サポートされているが非推奨の Amazon FSx 機能のリファレンスについて説明します。

トピック

カスタムバックアップスケジュールの設定

AWS Backup を使用して、ファイルシステムのカスタムバックアップスケジュールを設定することをお勧めします。ここで提供される情報は、AWS Backup の使用時よりもバックアップを頻繁にスケジュールする必要がある場合の参考用です。

有効になっている場合、Amazon FSx は毎日のバックアップ期間中に 1 日 1 回、ファイルシステムのバックアップを自動的に取得します。Amazon FSx では、自動バックアップに対して指定した保持期間が適用されます。また、ユーザーによるバックアップもサポートしているため、いつでもバックアップを作成できます。

以下に、カスタムバックアップスケジューリングをデプロイするためのリソースと設定を示します。カスタムバックアップスケジューリングは、ユーザーが定義したカスタムスケジュールに基づいて Amazon FSx for Lustre ファイルシステム上でユーザー主導のバックアップを実行します。例えば、6 時間に 1 回、毎週 1 回などです。このスクリプトは、指定した保持期間以前のバックアップの削除も設定します。

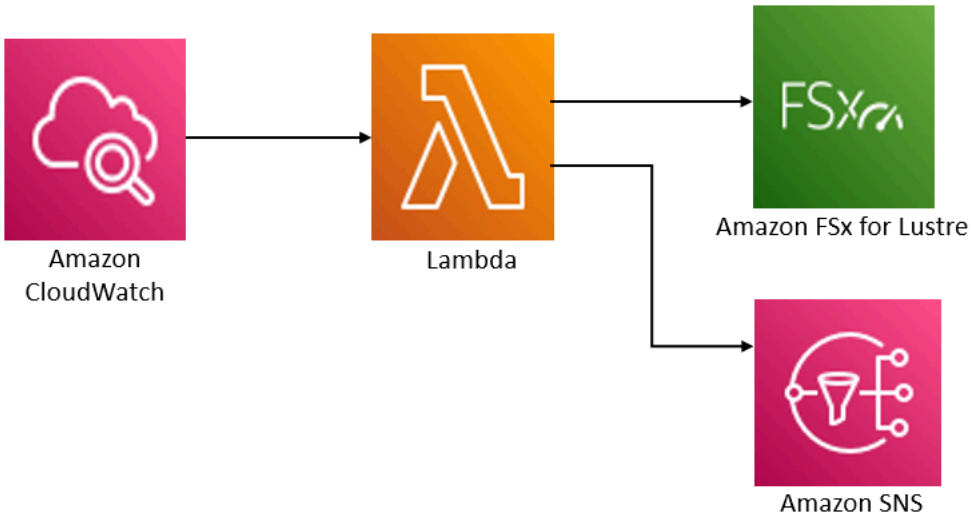
このソリューションは、必要なすべてのコンポーネントを自動的にデプロイし、以下のパラメータを受け取ります。

- ファイルシステム
- バックアップを実行するための CRON スケジュールパターン
- バックアップ保持期間 (日数)
- バックアップネームタグ

CRON スケジュールパターンの詳細については、「Amazon CloudWatch ユーザーガイド」の「ルールのスケジュール表現」を参照してください。

アーキテクチャの概要

このソリューションをデプロイすると、AWS クラウド に以下のリソースが構築されます。



このソリューションは以下の処理を実行します。

1. CloudFormation テンプレートは、CloudWatch イベント、Lambda 関数、Amazon SNS キュー、および IAM ロールをデプロイします。IAM ロールは、Amazon FSx for Lustre API オペレーションを呼び出すためのアクセス許可を Lambda 関数に与えます。
2. CloudWatch イベントは、最初のデプロイ時に CRON パターンとして定義したスケジュールで実行されます。このイベントは、バックアップを開始するために Amazon FSx for Lustre `CreateBackup` API オペレーションを呼び出して、ソリューションのバックアップマネージャー Lambda 関数を呼び出します。
3. バックアップマネージャーは、`DescribeBackups` を使用して、指定されたファイルシステムの既存のユーザー主導バックアップのリストを取得します。次に、初期デプロイ中に指定した保存期間より以前のバックアップを削除します。
4. 最初のデプロイ時に通知するオプションを選択すると、バックアップマネージャーは、正常なバックアップ時に Amazon SNS キューに通知メッセージを送信します。障害が発生した場合は常に通知が送信されます。

CloudFormation テンプレート

このソリューションは CloudFormation を使用して、Amazon FSx for Lustre カスタムバックアップスケジューリングソリューションのデプロイを自動化します。このソリューションを使用するに

は、fsx-scheduled-backup.テンプレートCloudFormation テンプレートをダウンロードしてください。

オートメーションデプロイ

次の手順では、このカスタムバックアップスケジューリングソリューションを設定および展開します。デプロイには約 5 分かかります。スタートする前に、自分の AWS アカウントの Amazon Virtual Private Cloud (Amazon VPC) で実行されている Amazon FSx for Lustre ファイルシステムの ID が必要です。リソースを作成するための詳細については、「Amazon FSx for Lustre の使用開始」を参照してください。

> **Note**
>
> このソリューションを実行すると、関連する AWS のサービスに料金が発生します。詳細については、それらのサービスの料金詳細ページを参照してください。

カスタムバックアップソリューションスタックを起動するには

1. fsx-scheduled-backup.template CloudFormation テンプレートをダウンロードします。CloudFormation スタックの作成の詳細については、「AWS CloudFormation ユーザーガイド」の「AWS CloudFormation コンソールでのスタックの作成」を参照してください。

   > **Note**
   >
   > デフォルトでは、このテンプレートは米国東部 (バージニア北部) AWS リージョンで起動します。Amazon FSx for Lustre は現在、特定の AWS リージョン でのみ利用可能です。本ソリューションは、Amazon FSx for Lustre が利用可能な AWS リージョンで起動する必要があります。詳細については、「AWS 全般のリファレンス」の「AWS リージョン とエンドポイント」の Amazon FSx セクションを参照してください。

2. [Parameters] (パラメータ) については、テンプレートのパラメータを確認し、ファイルシステムのニーズに合わせて変更します。このソリューションは以下のデフォルト値を使用します。

パラメータ	デフォルト	説明
Amazon FSx for Lustre ファイルシステム ID	デフォルト値なし	バックアップするファイルシステムのファイルシステム ID。
バックアップの CRON スケジュールパターン。	0 0/4 * * ? *	CloudWatch イベントを実行するスケジュールで、新しいバックアップをトリガーし、保持期間外の古いバックアップを削除します。
バックアップ 保持期間 (日数)	7	ユーザーによるバックアップを保持する日数。Lambda 関数は、この日数より古いユーザーによるバックアップを削除します。
バックアップの名前	ユーザースケジュールのバックアップ	バックアップの名前は、Amazon FSx for Lustre 管理コンソールの、バックアップ名 の欄に表示されます。
バックアップの通知	はい	バックアップが正常に開始されたときに通知するかどうかを選択します。エラーが発生した場合は、常に通知が送信されます。
Eメールアドレス	デフォルト値なし	SNS 通知をサブスクライブするためのEメールアドレス。

3. [Next] (次へ) を選択します。
4. [Options] (オプション) には、[Next] (次へ) を選択します。

5. [Review] (確認) で、設定を確認して確定します。テンプレートが IAM リソースを作成することを確認するチェックボックスを選択する必要があります。
6. [Create] (作成) を選択してスタックをデプロイします。

CloudFormation コンソールの [Status] (ステータス) 欄でスタックのステータスを表示できます。約 5 分後に CREATE_COMPLETE のステータスが表示されます。

追加のオプション

このソリューションで作成された Lambda 関数を使用して、複数の Amazon FSx for Lustre ファイルシステムのカスタムスケジュールバックアップを実行できます。ファイルシステム ID は、CloudWatch イベントの入力 JSON で Amazon FSx for Lustre 関数に渡されます。Lambda 関数に渡されるデフォルトの JSON は次のとおりです。ここで、`FileSystemId` と `SuccessNotification` の値は、CloudFormation スタックの起動時に指定されたパラメータから渡されます。

```
{
 "start-backup": "true",
 "purge-backups": "true",
 "filesystem-id": "${FileSystemId}",
 "notify_on_success": "${SuccessNotification}"
}
```

追加の Amazon FSx for Lustre ファイルシステムのバックアップをスケジュールするには、別の CloudWatch イベントルールを作成します。このソリューションで作成された Lambda 関数をターゲットとして用い、スケジュールイベント出典を使用します。[Configure input] (入力の設定) で [Constant (JSON text)] (定数 (JSON テキスト)) を選択します。JSON 入力の場合は、バックアップする Amazon FSx for Lustre ファイルシステムのファイルシステム ID を `${FileSystemId}` の代わりに入力するだけです。また、上記の JSON の `${SuccessNotification}` の代わりに `Yes` または `No` のどちらかを入力してください。

手動で作成する追加の CloudWatch イベント ルールは、Amazon FSx for Lustre カスタムスケジュールバックアップソリューション CloudFormation スタックのパートではありません。したがって、スタックを削除してもそれらは削除されません。

ドキュメント履歴

- API バージョン: 2018 年 3 月 1 日
- 最新のドキュメント更新: 2025 年 9 月 30 日

以下の表は、Amazon FSx for Lustre ユーザーガイド の重要な変更点を示します。ドキュメントの更新に関する通知については、RSS フィードにサブスクライブできます。

変更	説明	日付
永続 2 デプロイタイプに AWS リージョン サポートが追加されました	永続 2 SSD FSx for Lustre ファイルシステムが米国西部 (フェニックス) ローカルゾーンで利用可能になりました。詳細については、「デプロイタイプの可用性」を参照してください。	2025 年 9 月 30 日
Ubuntu 24 Kernel 6.14.0 の Lustre クライアントサポートが追加されました	FSx for Lustre クライアントが Ubuntu 24.04 Kernel 6.14.0 を実行する Amazon EC2 インスタンスをサポートするようになりました。詳細については、「Lustre クライアントのインストール」を参照してください。	2025 年 9 月 24 日
LustreAmazon Linux 2023 Kernel 6.12 のクライアントサポートが追加されました	FSx for Lustre クライアントが Amazon Linux 2023 Kernel 6.12 を実行する Amazon EC2 インスタンスをサポートするようになりました。詳細については、「Lustre クライアントのインストール」を参照してください。	2025 年 9 月 9 日

追加の AWS リージョン サポートの追加	FSx for Lustre ファイルシステムがアジアパシフィック (台北) で利用可能になりました。詳細については、「デプロイタイプの可用性」を参照してください。	2025 年 8 月 18 日
Amazon FSx が AmazonFSx ServiceRolePolicy AWS マネージドポリシーを更新	Amazon FSx は、AmazonFSxServiceRolePolicy の `ec2:AssignIpv6Addresses` および `ec2:UnassignIpv6Addresses` アクセス許可を追加しました。詳細については、「Amazon FSx の AWS マネージドポリシーに関する更新」を参照してください。	2025 年 7 月 22 日
CentOS、Rocky Linux、Red Hat Enterprise Linux (RHEL) 9.6 に対する Lustre クライアントサポートが追加されました	FSx for Lustre のクライアントが、Rocky Linux および Red Hat Enterprise Linux (RHEL) 9.6 を実行する Amazon EC2 インスタンスをサポートするようになりました。詳細については、「Lustre クライアントのインストール」を参照してください。	2025 年 7 月 1 日

Amazon FSx が AmazonFSx FullAccess AWS マネージドポリシーを更新	AmazonFSxFullAccess 管理ポリシーが更新され、`fsx:CreateAndAttachS3AccessPoint` 、`fsx:DescribeS3AccessPointAttachments` 、および`fsx:DetachAndDeleteS3AccessPoint` アクセス許可が追加されました。	2025 年 6 月 25 日
Amazon FSx が AmazonFSx ConsoleFullAccess AWS マネージドポリシーを更新	AmazonFSxConsoleFullAccess 管理ポリシーが更新され、`fsx:CreateAndAttachS3AccessPoint` 、`fsx:DescribeS3AccessPointAttachments` 、および`fsx:DetachAndDeleteS3AccessPoint` アクセス許可が追加されました。	2025 年 6 月 25 日
インテリジェント階層化ストレージクラスのサポートが追加されました	インテリジェント階層化ストレージクラスを使用して FSx for Lustre ファイルシステムを作成できるようになりました。インテリジェント階層化は、頻繁にアクセスされるデータへの低レイテンシーアクセスのためのオプションの SSD キャッシュを備えた完全伸縮自在なストレージを提供します。詳細については、インテリジェント階層化ストレージクラスのパフォーマンス特性を参照してください。	2025 年 5 月 29 日

追加の AWS リージョン サポートの追加	FSx for Lustre ファイルシステムが、アジアパシフィック (タイ) およびメキシコ (中部) で利用可能になりました。詳細については、「デプロイタイプの可用性」を参照してください。	2025 年 5 月 8 日
Ubuntu 24 の Lustre クライアントのサポートが追加されました	FSx for Lustre クライアントが Ubuntu 24.04 を実行する Amazon EC2 インスタンスをサポートするようになりました。詳細については、「Lustre クライアントのインストール」を参照してください。	2025 年 3 月 19 日
Amazon FSx が AmazonFSxConsoleReadOnlyAccess AWS マネージドポリシーを更新	Amazon FSx は、AmazonFSxConsoleReadOnlyAccess ポリシーに `ec2:DescribeNetworkInterfaces` 権限を追加するように更新しました。詳細は「AmazonFSxConsoleReadOnlyAccess」のポリシーをご参照ください。	2025 年 2 月 25 日
Lustre バージョンのアップグレードのサポートが追加されました	FSx for Lustre ファイルシステムの Lustre バージョンを新しいバージョンにアップグレードできるようになりました。バージョン管理の詳細については、「Managing Lustre version」を参照してください。	2025 年 2 月 12 日

Amazon FSx が AmazonFSx ConsoleFullAccess AWS マネージドポリシーを更新	Amazon FSx が AmazonFSx ConsoleFullAccess ポリシーを更新し、`ec2:DescribeNetworkInterfaces` アクセス権限を追加しました。詳細は「AmazonFSx ConsoleFullAccess」のポリシーをご参照ください。	2025 年 2 月 7 日
永続 2 デプロイタイプに AWS リージョン サポートが追加されました	永続 2 SSD FSx for Lustre ファイルシステムがアジアパシフィック (マレーシア) AWS リージョン で利用可能になりました。詳細については、「デプロイタイプの可用性」を参照してください。	2025 年 1 月 2 日
CentOS、Rocky Linux、Red Hat Enterprise Linux (RHEL) 9.5 に対する Lustre クライアントサポートが追加されました	FSx for Lustre のクライアントが、Rocky Linux および Red Hat Enterprise Linux (RHEL) 9.5 を実行する Amazon EC2 インスタンスをサポートするようになりました。詳細については、「Lustre クライアントのインストール」を参照してください。	2024 年 12 月 26 日

EFA のサポートが追加されました	EFA をサポートするクライアントインスタンスのネットワークパフォーマンスを向上させる Elastic Fabric Adapter (EFA) をサポートする FSx for Lustre 永続型 2 ファイルシステムを作成できるようになりました。EFA を有効にすると、GPUDirect Storage (GDS) と ENA Express もサポートされます。詳細については、「Working with EFA-enabled file systems」を参照してください。	2024 年 11 月 27 日
永続 2 デプロイタイプに AWS リージョン サポートが追加されました	永続 2 SSD FSx for Lustre ファイルシステムが米国西部 (北カリフォルニア) AWS リージョン で利用可能になりました。詳細については、「デプロイタイプの可用性」を参照してください。	2024 年 11 月 27 日
Ubuntu 22 Kernel 6.8.0 の Lustre クライアントサポートが追加されました	FSx for Lustre クライアントが Ubuntu 22.04 Kernel 6.8.0 を実行する Amazon EC2 インスタンスをサポートするようになりました。詳細については、「Lustre クライアントのインストール」を参照してください。	2024 年 11 月 8 日

Amazon CloudWatch メトリクスの追加とモニタリングダッシュボードの強化のサポートを追加	FSx for Lustre では、ネットワーク、パフォーマンス、ストレージメトリクスの追加とモニタリングダッシュボードの強化によって、ファイルシステムのアクティビティを容易に把握できるようになりました。詳細については、「Amazon CloudWatch によるモニタリング」を参照してください。	2024 年 9 月 25 日
永続 2 デプロイタイプに AWS リージョン サポートが追加されました	永続 2 SSD FSx for Lustre ファイルシステムが米国東部 (ダラス) ローカルゾーンで利用可能になりました。詳細については、「デプロイタイプの可用性」を参照してください。	2024 年 9 月 20 日
Ubuntu 22 Kernel 6.5.0 の Lustre クライアントサポートが追加されました	FSx for Lustre クライアントが Ubuntu 22.04 Kernel 6.5.0 を実行する Amazon EC2 インスタンスをサポートするようになりました。詳細については、「Lustre クライアントのインストール」を参照してください。	2024 年 8 月 1 日

CentOS、Rocky Linux および Red Hat Enterprise Linux (RHEL) 8.10 に対する Lustre クライアントサポートが追加されました	FSx for Lustre のクライアントが、CentOS、Rocky Linux、Red Hat Enterprise Linux (RHEL) 8.10 を実行する Amazon EC2 インスタンスをサポートするようになりました。詳細については、「Lustre クライアントのインストール」を参照してください。	2024 年 6 月 18 日
メタデータのパフォーマンスを向上させるためのサポートが追加されました	メタデータのパフォーマンスを向上させる機能を提供するメタデータ設定を使用して、FSx for Lustre 永続 2 ファイルシステムを作成できるようになりました。詳細については、「ファイルシステムのメタデータパフォーマンス」と「メタデータパフォーマンスの管理」を参照してください。	2024 年 6 月 6 日
永続 2 デプロイタイプに AWS リージョン サポートが追加されました	永続 2 SSD FSx for Lustre ファイルシステムが米国東部 (アトランタ) ローカルゾーンで利用可能になりました。詳細については、「デプロイタイプの可用性」を参照してください。	2024 年 5 月 29 日

CentOS、Rocky Linux、Red Hat Enterprise Linux (RHEL) 9.4 に対する Lustre クライアントサポートが追加されました	FSx for Lustre のクライアントが、Rocky Linux および Red Hat Enterprise Linux (RHEL) 9.4 を実行する Amazon EC2 インスタンスをサポートするようになりました。詳細については、「Lustre クライアントのインストール」を参照してください。	2024 年 5 月 16 日
永続 2 デプロイタイプに AWS リージョン サポートが追加されました	永続 2 SSD FSx for Lustre ファイルシステムがカナダ西部 (カルガリー) AWS リージョン で利用可能になりました。詳細については、「デプロイタイプの可用性」を参照してください。	2024 年 5 月 3 日
Amazon Linux 2023 の Lustre クライアントサポートが追加されました	FSx for Lustre クライアントが Amazon Linux 2023 を実行する Amazon EC2 インスタンスをサポートするようになりました。詳細については、「Lustre クライアントのインストール」を参照してください。	2024 年 3 月 25 日
CentOS、Rocky Linux、Red Hat Enterprise Linux (RHEL) 8.9 に対する Lustre クライアントサポートが追加されました	FSx for Lustre のクライアントが、CentOS、Rocky Linux、Red Hat Enterprise Linux (RHEL) 8.9 を実行する Amazon EC2 インスタンスをサポートするようになりました。詳細については、「Lustre クライアントのインストール」を参照してください。	2024 年 1 月 9 日

Amazon FSx はAmazonFSxFullAccess、AmazonFSxConsoleFullAccess、AmazonFSxReadOnlyAccess、AmazonFSxConsoleReadOnlyAccess、AmazonFSxServiceRolePolicy AWS マネージドポリシーを更新しました	Amazon FSx に、AmazonFSxFullAccess、AmazonFSxConsoleFullAccess、AmazonFSxReadOnlyAccess、AmazonFSxConsoleReadOnlyAccess、および AmazonFSxServiceRolePolicy ポリシーを更新して、`ec2:GetSecurityGroupsForVpc` アクセス許可が追加されました。詳細については、「Amazon FSx の AWS マネージドポリシーに関する更新」を参照してください。	2024 年 1 月 9 日
CentOS、Rocky Linux、Red Hat Enterprise Linux (RHEL) 9.0 と 9.3 に対する Lustre クライアントサポートが追加されました	FSx for Lustre のクライアントが、Rocky Linux および Red Hat Enterprise Linux (RHEL) 9.0 と 9.3 を実行する Amazon EC2 インスタンスをサポートするようになりました。詳細については、「Lustre クライアントのインストール」を参照してください。	2023 年 12 月 20 日

Amazon FSx for Lustre で、AmazonFSxFullAccess および AmazonFSxConsoleFullAccess AWS マネージドポリシーが更新されました	Amazon FSx で、AmazonFSxFullAccess ポリシーと AmazonFSxConsoleFullAccess ポリシーが更新され、`ManageCrossAccountDataReplication` アクションが追加されました。詳細については、「Amazon FSx の AWS マネージドポリシーに関する更新」を参照してください。	2023 年 12 月 20 日
Amazon FSx で、AmazonFSxFullAccess および AmazonFSxConsoleFullAccess AWS マネージドポリシーが更新されました	Amazon FSx で、AmazonFSxFullAccess ポリシーと AmazonFSxConsoleFullAccess ポリシーが更新され、`fsx:CopySnapshotAndUpdateVolume` アクセス許可が追加されました。詳細については、「Amazon FSx の AWS マネージドポリシーに関する更新」を参照してください。	2023 年 11 月 26 日
スループットキャパシティスケーリングの追加	既存の FSx for Lustre の永続的な SSD ベースファイルシステムのスループットキャパシティを、スループット要件の進展に応じて変更できるようになりました。詳細については、「スループットキャパシティの管理」を参照してください。	2023 年 11 月 16 日

Amazon FSx で、AmazonFSxFullAccess および AmazonFSxConsoleFullAccess AWS マネージドポリシーが更新されました	Amazon FSx で、AmazonFSxFullAccess ポリシーと AmazonFSxConsoleFullAccess ポリシーが更新され、`fsx:DescribeSharedVPCConfiguration` アクセス許可と `fsx:UpdateSharedVPCConfiguration` アクセス許可が追加されました。詳細については、「Amazon FSx の AWS マネージドポリシーに関する更新」を参照してください。	2023 年 11 月 14 日
プロジェクトクォータのサポートの追加	プロジェクトのストレージクォータを作成できるようになりました。プロジェクトクォータは、プロジェクトに関連するすべてのファイルまたはディレクトリに適用されます。詳細については、「ストレージのクォータ」を参照してください。	2023 年 8 月 29 日
Lustre のバージョン 2.15 のサポートが追加されました	Amazon FSx コンソールを使用して作成されるすべての FSx for Lustre ファイルシステムが、Lustre バージョン 2.15 で構築されるようになりました。詳細については、「ステップ 1: Amazon FSx for Lustre ファイルシステムを作成する」を参照してください。	2023 年 8 月 29 日

永続 2 デプロイタイプに AWS リージョン サポートが追加されました	永続 2 FSx for Lustre ファイルシステムがイスラエル (テルアビブ) AWS リージョン で利用可能になりました。詳細については、「FSx for Lustre ファイルシステムのデプロイオプション」を参照してください。	2023 年 8 月 24 日
データリポジトリのリリースタスクのサポートを追加	FSx for Lustre は、S3 データリポジトリにリンクされたファイルシステムからアーカイブファイルをリリースするためのデータリポジトリのリリースタスクを提供するようになりました。ファイルを解放すると、ファイルのリストとメタデータは保持されますが、そのファイルのコンテンツのローカルコピーは削除されます。詳細については、「データリポジトリタスクを使用してファイルをリリースする」を参照してください。	2023 年 8 月 9 日
Amazon FSx が AmazonFSx ServiceRolePolicy AWS マネージドポリシーを更新	Amazon FSx は、AmazonFSxServiceRolePolicy の `cloudwatch:PutMetricData` アクセス許可を更新しました。詳細については、「Amazon FSx の AWS マネージドポリシーに関する更新」を参照してください。	2023 年 7 月 24 日

Amazon FSx が AmazonFSx FullAccess AWS マネージドポリシーを更新	Amazon FSx が AmazonFSx FullAccess ポリシーを更新し、`fsx:*` アクセス権限を削除し、特定の `fsx` アクションを追加しました。詳細については、「AmazonFSx FullAccess」のポリシーを参照してください。	2023 年 7 月 13 日
Amazon FSx が AmazonFSx ConsoleFullAccess AWS マネージドポリシーを更新	Amazon FSx が AmazonFSx ConsoleFullAccess ポリシーを更新し、`fsx:*` アクセス権限を削除し、特定の `fsx` アクションを追加しました。詳細については、「AmazonFSx ConsoleFullAccess」のポリシーを参照してください。	2023 年 7 月 13 日
CentOS、Rocky Linux、Red Hat Enterprise Linux (RHEL) 8.8 に対する Lustre クライアントサポートが追加されました	FSx for Lustre のクライアントが、CentOS、Rocky Linux、Red Hat Enterprise Linux (RHEL) 8.8 を実行する Amazon EC2 インスタンスをサポートするようになりました。詳細については、「Lustre クライアントのインストール」を参照してください。	2023 年 5 月 25 日

AutoImport (自動インポート) と AutoExport (自動エクスポート) メトリクスのサポートの追加	FSx for Lustre が、データリポジトリにリンクされたファイルシステムの自動インポートと自動エクスポートの更新をモニタリングする Amazon CloudWatch メトリクスを提供するようになりました。詳細については、「Amazon CloudWatch によるモニタリング」を参照してください。	2023 年 3 月 31 日
永続 1 と スクラッチ 2 デプロイタイプの DRA サポートの追加されました	データリポジトリの関連付けを作成して、永続 1 または スクラッチ 2 のデプロイタイプでデータリポジトリを Lustre 2.12 ファイルシステムにリンクできるようになりました。詳細については、「Amazon FSx for Lustre でデータリポジトリを使用する」を参照してください。	2023 年 3 月 29 日
CentOS、Rocky Linux、および Red Hat Enterprise Linux (RHEL) 8.7 の Lustre クライアントサポートが追加されました	FSx for Lustre クライアントが、CentOS、Rocky Linux、Red Hat Enterprise Linux (RHEL) 8.7 を実行する Amazon EC2 インスタンスをサポートするようになりました。詳細については、「Lustre クライアントのインストール」を参照してください。	2022 年 12 月 5 日

永続 2 デプロイタイプに AWS リージョン サポートが追加されました	次世代の 永続 2 SSD FSx for Lustre ファイルシステムが、欧州 (ストックホルム)、アジアパシフィック (香港)、アジアパシフィック (ムンバイ)、アジアパシフィック (ソウル) AWS リージョン で利用できるようになりました。詳細については、「FSx for Lustre ファイルシステムのデプロイオプション」を参照してください。	2022 年 11 月 10 日
CentOS、Rocky Linux、および Red Hat Enterprise Linux (RHEL) 8.6 の Lustre クライアントサポートが追加されました	FSx for Lustre クライアントが、CentOS、Rocky Linux、Red Hat Enterprise Linux (RHEL) 8.6 を実行する Amazon EC2 インスタンスをサポートするようになりました。詳細については、「Lustre クライアントのインストール」を参照してください。	2022 年 9 月 8 日
Ubuntu 22 の Lustre クライアントのサポートが追加されました	FSx for Lustre クライアントが Ubuntu 22.04 を実行する Amazon EC2 インスタンスをサポートするようになりました。詳細については、「Lustre クライアントのインストール」を参照してください。	2022 年 7 月 28 日

Rocky Linux の Lustre クライアントサポートが追加されました	FSx for Lustre クライアントが Rocky Linux を実行する Amazon EC2 インスタンスをサポートするようになりました。詳細については、「Lustre クライアントのインストール」を参照してください。	2022 年 7 月 8 日
Lustre ルートスカッシュのサポートが追加されました	今後は、Lustre ルートスカッシュ機能を使用することで、FSx for Lustre ファイルシステムへのアクセスを (ルートとして) 試みるクライアントに対し、ルートレベルのアクセスを制限できるようになりました。詳細については、「Lustre ルートスカッシュ」を参照してください。	2022 年 5 月 25 日
永続 2 デプロイタイプに AWS リージョン サポートが追加されました	次世代の 永続 2 SSD FSx for Lustre ファイルシステムが、欧州 (ロンドン)、アジアパシフィック (シンガポール)、アジアパシフィック (シドニー) AWS リージョン で利用できるようになりました。詳細については、「FSx for Lustre ファイルシステムのデプロイオプション」を参照してください。	2022 年 4 月 19 日

Amazon FSx for Lustre ファイルシステムのファイルを移行する際の、AWS DataSync の使用に関するサポートが追加されました	既存のファイルシステムから FSx for Lustre ファイルシステムにファイルを移行する際に、AWS DataSync を使用できるようになりました。詳細については、「How to migrate existing files to FSx for Lustre using AWS DataSync」を参照してください。	2022 年 4 月 5 日
AWS PrivateLink インターフェイス VPC エンドポイントのサポートが追加に	インターフェイス VPC エンドポイントを使用し、インターネット経由でトラフィックを送信せずに、VPC から Amazon FSx API にアクセスできます。詳細については、「Amazon FSx and interface VPC endpoints」を参照してください。	2022 年 4 月 5 日
Lustre DRA キューイングのサポートが追加されました	FSx for Lustre のファイルシステムを作成する際に、DRA (データリポジトリアソシエーション) を作成できるようになりました。リクエストはキューに入れられ、ファイルシステムが使用可能になると DRA が作成されます。詳細については、「ファイルシステムを S3 バケットにリンクする」を参照してください。	2022 年 2 月 28 日

CentOS および Red Hat Enterprise Linux (RHEL) 8.5 の Lustre クライアントサポートが追加されました	FSx for Lustre クライアントは、CentOS および Red Hat Enterprise Linux (RHEL) 8.5 を実行する Amazon EC2 インスタンスをサポートするようになりました。詳細については、「Lustre クライアントのインストール」を参照してください。	2021 年 12 月 20 日
リンクされたデータリポジトリへの FSx for Lustre からの変更のエクスポートに関するサポート	FSx for Lustre を設定して、ファイルシステムからリンクされた Simple Storage Service (Amazon S3) データリポジトリへ、新規、変更、および削除されたファイルを自動的にエクスポートできるようになりました。データリポジトリタスクを使用して、データおよびメタデータの変更をデータリポジトリにエクスポートできます。複数のデータリポジトリへのリンクを設定することもできます。詳細については、「データリポジトリへの変更のエクスポート」を参照してください。	2021 年 11 月 30 日

Lustre ログのサポートが追加されました	FSx for Lustre を設定して、ファイルシステムに関連付けられているデータリポジトリのエラーイベントと警告イベントを Amazon CloudWatch Logs に口グに記録できるようになりました。詳細については、「Amazon CloudWatch Logs を使用した口ギング」を参照してください。	2021 年 11 月 30 日
永続的な SSD ファイルシステムは、より高いスループットとより小さなストレージ容量をサポートします	次世代 Persistent SSD FSx for Lustre ファイルシステムには、より高いスループットオプションがあり、より小さい最小ストレージ容量を備えています。詳細については、「FSx for Lustre ファイルシステムのデプロイオプション」を参照してください。	2021 年 11 月 30 日
Lustre のバージョン 2.12 のサポートが追加されました	FSx for Lustre のファイルシステムを作成するときに、Lustre のバージョン 2.12 を選択できるようになりました。詳細については、「ステップ 1: Amazon FSx for Lustre ファイルシステムを作成する」を参照してください。	2021 年 10 月 5 日

CentOS および Red Hat Enterprise Linux (RHEL) 8.4 の Lustre クライアントサポートが追加されました	FSx for Lustre クライアントが CentOS および Red Hat Enterprise Linux (RHEL) 8.4 を実行する Amazon EC2 インスタンスをサポートするようになりました。詳細については、「Lustre クライアントのインストール」を参照してください。	2021 年 6 月 9 日
データ圧縮サポートが追加されました	FSx for Lustre ファイルシステムを作成する際に、データ圧縮を有効にできるようになりました。既存の FSx for Lustre ファイルシステム上でデータ圧縮を有効または無効にすることもできます。詳細については、「Lustre データ圧縮」を参照してください。	2021 年 5 月 27 日
バックアップのコピーのサポートが追加されました	Amazon FSx を使用して、同じ AWS アカウント 内のバックアップを別の AWS リージョン (リージョン間コピー)、または同じ内部 AWS リージョン (リージョン内コピー) にコピーできるようになりました。詳細については、「バックアップのコピー」を参照してください。	2021 年 4 月 12 日

Lustre ファイルセットの Lustre クライアントサポート	FSx for Lustre クライアントでは、ファイルシステム名前空間のサブセットのみをマウントするファイルセットの使用がサポートされるようになりました。詳細については、「特定のファイルセットのマウント」を参照してください。	2021 年 3 月 18 日
非プライベート IP アドレスを使用したクライアントアクセスのサポートが追加されました	非プライベート IP アドレスを使用して、オンプレミスクライアントから FSx for Lustre ファイルシステムにアクセスできます。詳細については、「オンプレミスまたはピアリングされた Amazon VPC から Amazon FSx ファイルシステムをマウントする」を参照してください。	2020 年 12 月 17 日
ARM ベースの CentOS 7.9 の Lustre クライアントサポートが追加されました	FSx for Lustre クライアントが、ARM ベースの CentOS 7.9 を実行する Amazon EC2 インスタンスをサポートするようになりました。詳細については、「Lustre クライアントのインストール」を参照してください。	2020 年 12 月 17 日

CentOS および Red Hat Enterprise Linux (RHEL) 8.3 の Lustre クライアントサポートが追加されました	FSx for Lustre クライアントが、CentOS および Red Hat Enterprise Linux (RHEL) 8.3 を実行する Amazon EC2 インスタンスをサポートするようになりました。詳細については、「Lustre クライアントのインストール」を参照してください。	2020 年 12 月 16 日
ストレージとスループットキャパシティスケーリングのサポートが追加されました	ストレージとスループット要件の展開に応じて、既存の FSx for Lustre ファイルシステムのストレージおよびスループットキャパシティを増やすことができます。詳細については、「ストレージとスループットキャパシティの管理」を参照してください。	2020 年 11 月 24 日
ストレージクォータのサポートが追加されました	ユーザーおよびグループのストレージクォータを作成できるようになりました。ストレージのクォータは、FSx for Lustre ファイルシステム上でユーザーまたはグループが消費できるディスク容量とファイル数を制限します。詳細については、「ストレージのクォータ」を参照してください。	2020 年 11 月 9 日

Amazon FSx が AWS Backup と統合されました	ネイティブの Amazon FSx バックアップの使用に加えて、FSx ファイルシステムのバックアップおよび復元に AWS Backup を使用できるようになりました。詳細については Amazon FSx で AWS Backup を使用するを参照してください。	2020 年 11 月 9 日
HDD (ハードディスクドライブ) ストレージオプションのサポートが追加されました	SSD (ソリッドステートドライブ) ストレージオプションに加えて、FSx for Lustre は HDD (ハードディスクドライブ) ストレージオプションをサポートするようになりました。大規模でシーケンシャルなファイル操作を伴うスループット集約型のワークロードに、 HDD を使用するようにファイルシステムを設定できます。詳細については、「複数のストレージオプション」を参照してください。	2020 年 8 月 12 日
リンクされたデータリポジトリの変更を FSx for Lustre にインポートするためのサポート	ファイルシステムの作成後に、追加された新しいファイルとリンクされたデータレポジトリで変更されたファイルを自動的にインポートするように FSx for Lustre ファイルシステムを設定できるようになりました。詳細については、「データレポジトリから更新を自動的にインポートする」を参照してください。	2020 年 7 月 23 日

SUSE Linux SP4 および SP5 の Lustre クライアントサポートが追加されました	FSx for Lustre クライアントが SUSE Linux SP4 および SP5 を実行する Amazon EC2 インスタンスをサポートするようになりました。詳細については、「Lustre クライアントのインストール」を参照してください。	2020 年 7 月 20 日
CentOS および Red Hat Enterprise Linux (RHEL) 8.2 の Lustre クライアントサポートが追加されました	FSx for Lustre クライアントが、CentOS および Red Hat Enterprise Linux (RHEL) 8.2 を実行する Amazon EC2 インスタンスをサポートするようになりました。詳細については、「Lustre クライアントのインストール」を参照してください。	2020 年 7 月 20 日
自動および手動のファイルシステムバックアップサポートが追加されました	Simple Storage Service (Amazon S3) 耐久データリポジトリにリンクされていないファイルシステムの自動デイリーバックアップと手動バックアップを実行できるようになりました。詳細については、「バックアップの使用」を参照してください。	2020 年 6 月 23 日

つの新しいファイルシステムデプロイタイプがリリースされました	スクラッチファイルシステムは、データのテンポラリストレージと短期間の処理のために設計されています。永続的ファイルシステムは、長期ストレージとワークロード用に設計されています。詳細については、「FSx for Lustre デプロイオプション」を参照してください。	2020 年 2 月 12 日
POSIX メタデータのサポートが追加されました	FSx for Lustre は、Simple Storage Service (Amazon S3) 上のリンクされた耐久性のあるデータリポジトリにファイルをインポートおよびエクスポートするときに、関連する POSIX メタデータを保持します。詳細については、「データリポジトリの POSIX メタデータサポート」を参照してください。	2019 年 12 月 23 日
新しいデータリポジトリタスク機能がリリースされました	データリポジトリタスクを使用して、変更されたデータおよび関連する POSIX メタデータをリンクされた Simple Storage Service (Amazon S3) 上の耐久性のあるデータリポジトリにエクスポートできるようになりました。詳細については、「データリポジトリタスク」を参照してください。	2019 年 12 月 23 日

追加の AWS リージョン サポートが追加されました	FSx for Lustre が欧州 (ロンドン) リージョン AWS リージョン で利用可能になりました。FSx for Lustre のリージョン固有の制限については、「制限」を参照してください。	2019 年 7 月 9 日
さらなる AWS リージョン サポートが追加されました	FSx for Lustre がアジアパシフィック (シンガポール) AWS リージョン で使用できるようになりました。FSx for Lustre のリージョン固有の制限については、「制限」を参照してください。	2019 年 6 月 26 日
Amazon Linux および Amazon Linux 2 の Lustre クライアントサポートが追加されました	FSx for Lustre クライアントが Amazon Linux および Amazon Linux 2 を実行する Amazon EC2 インスタンスをサポートするようになりました。詳細については、「Lustre クライアントのインストール」を参照してください。	2019 年 3 月 11 日

ユーザー定義のデータエクスポートパスのサポートが追加されました	これで、ユーザーは Simple Storage Service (Amazon S3) バケット内の元のオブジェクトを上書きしたり、指定したプレフィックスに新しいファイルや変更されたファイルを書き込むことができるようになりました。このオプションを使用すると、FSx for Lustre をデータ処理ワークフローに組み込む柔軟性が向上します。詳細については、「Simple Storage Service (Amazon S3) バケットへのデータのエクスポート」を参照してください。	2019 年 2 月 6 日
合計ストレージのデフォルト制限が増加しました	すべての FSx for Lustre ファイルシステムのデフォルトの合計ストレージは 100,800 GiB に増加しました。詳細については、「制限」を参照してください。	2019 年 1 月 11 日
Amazon FSx for Lustre が一般利用可能になりました	Amazon FSx for Lustre は、高性能コンピューティング、機械学習、メディア処理ワークフローなど、コンピューティング集約型のワークロードに最適化されたフルマネージドのファイルシステムです。	2018 年 11 月 28 日

翻訳は機械翻訳により提供されています。提供された翻訳内容と英語版の間で齟齬、不一致または矛盾がある場合、英語版が優先します。