AWS ホワイトペーパー

# Amazon Elastic File System を使用したファイルデータの暗号化

# Amazon Elastic File System を使用したファイルデータの暗号化: AWS ホワイトペーパー

# Table of Contents

# Amazon Elastic File System を使用したファイルデータの暗号化

公開日: 2021 年 2 月 22 日 (ドキュメント履歴と寄稿者)

## 要約

AWS にとってセキュリティはジョブゼロ (どんな最優先事項よりも重要なこと) であり、AWS は企業内でセキュリティをジョブゼロとして実行するためのツールをお客様に提供しています。政府の規制や業界または企業のコンプライアンスポリシーでは、暗号化ポリシー、暗号化アルゴリズム、適切なキー管理を使ってさまざまな分類のデータを保護することが必要とされる場合があります。このホワイトペーパーでは、Amazon Elastic File System (Amazon EFS) の暗号化に関するベストプラクティスの概要を説明します。

## はじめに

Amazon Elastic File System (Amazon EFS) は、シンプルかつスケーラブルで可用性が高く、耐久性に優れた共有ファイルシステムをクラウドで提供します。Amazon EFS を使用して作成するファイルシステムは伸縮自在で、データの追加や削除に応じて自動的に拡張および縮小できます。複数のアベイラビリティーゾーン (AZ) にあるストレージサーバーは数の制約がなく、データを分散して、ペタバイト規模まで拡大できます。

これらのファイルシステムに保存されているデータは、Amazon EFS を使って保管中および転送中に暗号化できます。保管中のデータを暗号化するために、AWS マネジメントコンソールまたは AWS Command Line Interface (AWS CLI) から暗号化ファイルシステムを作成できます。また、Amazon EFS API または AWS SDK の 1 つを使用して、暗号化ファイルシステムをプログラムで作成することもできます。

保管中のデータを暗号化するために、Amazon EFS は AWS Key Management Service (AWS KMS) と統合してキー管理を行います。ファイルシステムをマウントし、すべての NFS トラフィックを Transport Layer Security (TLS) 経由で転送することによって、転送中のデータの暗号化を有効にできます。

このホワイトペーパーでは、Amazon EFS の暗号化に関するベストプラクティスの概要を説明します。クライアント接続レイヤーで転送中のデータの暗号化を有効にする方法と、AWS マネジメントコンソールと AWS CLI で暗号化ファイルシステムを作成する方法について説明します。

> **Note**
>
> API と SDK を使用した暗号化ファイルシステムの作成は、このホワイトペーパーでは扱いません。この方法の詳細については、Amazon EFS ユーザーガイドの「Amazon EFS API」または SDK ドキュメントを参照してください。

# 基本概念と用語

このセクションでは、このホワイトペーパーで参照されている概念と用語を定義します。

- Amazon Elastic File System (Amazon EFS) — シンプルでスケーラブルな共有ファイルストレージをAWS クラウド内で提供する、可用性と耐久性に優れたサービスです。Amazon EFS には、標準のファイルシステムインターフェイスとファイルシステムセマンティクスが用意されています。複数のアベイラビリティーゾーンにある無数のストレージサーバー上で、データを実質無制限に保存できます。
- AWS Identity and Access Management (IAM) — AWS サービス API へのアクセスを安全かつきめ細かに制御できるサービスです。ポリシーが作成され、個々のユーザー、グループ、ロールのアクセスを制限するために使用されます。AWS KMS キーは IAM コンソールから管理できます。
- AWS KMS — データの暗号化に使用される暗号化キーであるカスタマーマスターキー (CMK) の作成と管理を容易にするマネージドサービスです。AWS KMS CMK は、中国 (北京) および中国 (寧夏) リージョンを除き、FIPS 140-2 暗号化モジュール検証プログラムによる検証済みのハードウェアセキュリティモジュール (HSM) で保護されています。AWS KMS は、データを暗号化する別の AWS のサービスと統合されています。また、AWS CloudTrail と完全に統合され、AWS KMS がお客様に代わって実行した API コールのログを提供します。これは、お客様の組織に適用されるコンプライアンスまたは規制要件を満たすのに役立ちます。
- カスタマーマスターキー (CMK) — キー階層の最上位を表します。これには、データの暗号化および復号化のためのキーマテリアルが含まれています。AWS KMS はこのキーマテリアルを生成できます。また、ユーザーが生成したものを AWS KMS にインポートすることもできます。CMK は AWS アカウントと AWS リージョンに固有で、カスタマー管理型または AWS 管理型を使用できます。
- AWS 管理型 CMK — AWS がユーザーに代わって生成する CMK。AWS 管理の CMK は、統合された AWS のサービスのリソースの暗号化を有効にすると作成されます。AWS 管理型 CMK キーポリシーは AWS によって管理され、ユーザーは変更できません。AWS 管理型 CMK の作成や保存には料金がかかりません。
- カスタマー管理型 CMK — AWS マネジメントコンソールか API、AWS CLI、または SDK を使用して作成する CMK。CMK をより細かく制御する必要がある場合は、カスタマー管理型 CMK を使用できます。
- KMS キーポリシー — カスタマー管理型 CMK へのアクセスを制御するリソースポリシー。キーポリシー、または IAM ポリシーとキーポリシーの組み合わせを使用して、これらのアクセス権限をお客様が定義します。詳細については、AWS KMS デベロッパーガイドの「アクセス管理の概要」を参照してください。

- データキー — AWS KMS の外部でデータを暗号化するために AWS KMS によって生成される暗号化キー。AWS KMS では、CMK で保護されたデータキーを、承認されたエンティティ (ユーザーまたはサービス) で取得可能です。
- Transport Layer Security (TLS) — Secure Sockets Layer (SSL) の後継である TLS は、ネットワーク上でやり取りされる情報の暗号化に不可欠な暗号化プロトコルです。
- EFS マウントヘルパー — EFS ファイルシステムのマウントを簡略化するために使用される Linux クライアントエージェント (`amazon-efs-utils`)。TLS トンネル経由のすべての NFS トラフィックを設定、保守、ルーティングするために使用できます。

基本的な概念と用語の詳細については、AWS KMS デベロッパーガイドの「AWS Key Management Service の概念」を参照してください。

# 保管時のデータの暗号化

AWS には、業界標準の AES-256 暗号化アルゴリズムを使用して、保管中のすべてのデータとメタデータを暗号化する、暗号化ファイルシステムを作成するためのツールが用意されています。暗号化ファイルシステムは、暗号化と復号化を自動的かつ透過的に処理するように設計されているため、ユーザーがアプリケーションを変更する必要はありません。保管中のデータやメタデータの暗号化を要求する社内ポリシーや規制ポリシーの対象となっている組織では、暗号化ファイルシステムを作成することをお勧めします。

トピック



## キーの管理

Amazon EFS は、AWS KMS と統合されており、暗号化ファイルシステムの暗号化キーを管理します。また AWS KMS は、Amazon Simple Storage Service (Amazon S3)、Amazon Elastic Block Store (Amazon EBS)、Amazon Relational Database Service (Amazon RDS)、Amazon Aurora、Amazon Redshift、Amazon WorkMail、WorkSpaces など、他の AWS のサービスによる暗号化もサポートしています。Amazon EFS では、ファイルシステムのコンテンツを暗号化するために、XTS モードと 256 ビットキー (XTS-AES-256) を使用した高度な暗号化標準アルゴリズムが使用されます。

暗号化ポリシーを採用して保管中のデータを保護する方法を検討する際には、3 つの重要な質問に答える必要があります。3 つの質問は、マネージドサービスでも、Amazon EBS などのアンマネージドサービスでも、保存されているデータに対しては同じように有効です。

キーの保存場所

AWS KMS では、マスターキーは必要なときに確実に取得できるように、暗号化された形式で耐久性の高いストレージに保存されます。

キーが使用される場所

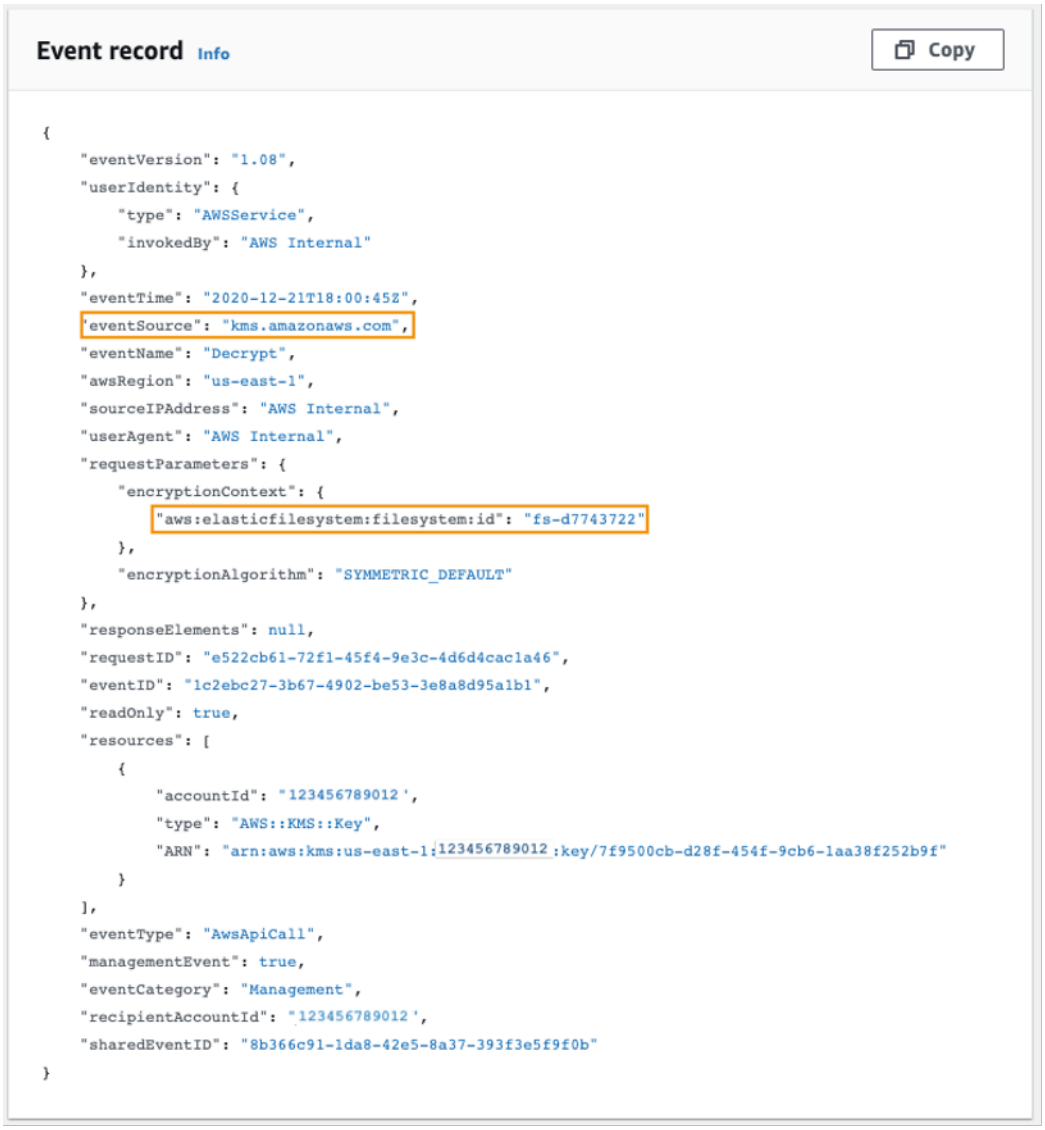
暗号化された Amazon EFS ファイルシステムの使用は、ファイルシステムをマウントするクライアントに対して透過的です。データはディスクに書き込まれる前に暗号化され、クライアントが読み取り要求を発行した後に復号化されるというように、すべての暗号化操作は EFS サービス内で行われます。

キーを使用できるユーザー

AWS KMS キーポリシーは、暗号化キーへのアクセスを制御します。

このキーポリシーを IAM ポリシーと組み合わせて、別の制御レイヤーを提供することをお勧めします。各キーにキーポリシーがあります。キーが AWS 管理の CMK の場合、AWS がキーポリシーを管理します。キーがカスタマー管理の CMK である場合は、お客様がキーポリシーを管理します。キーポリシーは、CMK へのアクセスを制御するための主要な方法です。キーの使用と管理を統制する権限はキーポリシーによって定義されます。

Amazon EFS を使用して暗号化ファイルシステムを作成する場合、お客様に代わって CMK を使用するアクセス権限を Amazon EFS に付与します。Amazon EFS がお客様の代わりに AWS KMS に対して行った呼び出しは、お客様の AWS アカウントから発信されたものとして CloudTrail ログに表示されます。次のスクリーンショットは、Amazon EFS によって行われた KMS Decrypt 呼び出しのサンプル CloudTrail イベントを示しています。




KMS Decrypt の CloudTrail ログ

AWS KMS と暗号化キーへのアクセスを管理する方法の詳細は、AWS KMS デベロッパーガイドの「AWS KMS CMK へのアクセスの管理」を参照してください。

AWS KMS による暗号化の管理方法の詳細は、ホワイトペーパー「AWS KMS の暗号化の詳細説明」を参照してください。

管理者 IAM ユーザーおよびグループを作成する方法の詳細は、IAM ユーザーガイドの「最初の IAM 管理者のユーザーおよびグループの作成」を参照してください。

# 暗号化ファイルシステムの作成

暗号化ファイルシステムは、AWS マネジメントコンソール、AWS CLI、Amazon EFS API、または AWS SDK を使用して作成できます。ファイルシステムの暗号化を有効にできるのは、作成時のみです。

Amazon EFS はキー管理のために AWS KMS と統合され、CMK を使用してファイルシステムを暗号化します。ファイル名、ディレクトリ名、ディレクトリコンテンツなどのファイルシステムメタデータは、AWS 管理の CMK を使用して暗号化および復号化されます。

ファイルやファイルデータのコンテンツは、お客様が選択した CMK を使用して暗号化および復号化されます。CMK は次の 3 つのタイプのいずれかです。

- Amazon EFS 用の AWS 管理の CMK
- お客様の AWS アカウントからのカスタマー管理の CMK
- 別の AWS アカウントからのカスタマー管理の CMK

組織は、CMK のアクセス制御と使用ポリシーだけでなく、作成、ローテーション、削除に関して完全な制御を必要とする社内ポリシーや規制ポリシーの対象になる場合があります。対象になる場合は、カスタマー管理の CMK の使用をお勧めします。対象にならない場合は、AWS 管理の CMK を使用できます。

すべてのユーザーが Amazon EFS の AWS 管理型 CMK を保有しています。そのエイリアスは `aws/elasticfilesystem` です。この CMK のキーポリシーは AWS によって管理され、お客様は変更できません。AWS 管理の CMK の作成と保存にコストは発生しません。

カスタマー管理の CMK を使用してファイルシステムを暗号化する場合は、所有するカスタマー管理の CMK のキーエイリアスを選択します。または、別のアカウントが所有するカスタマー管理の CMK の Amazon リソースネーム (ARN) を入力することもできます。お客様が所有するカスタマー管理の CMK を使って、キーを使用できるユーザーとサービスをキーポリシーとキー付与によって制御できます。

また、キーへのアクセスを無効化、再有効化、削除、または取り消すタイミングを選択することで、これらのキーの有効期間とローテーションを制御することもできます。他の AWS アカウントのキーへのアクセスを管理する方法については、AWS KMS デベロッパーガイドの「キーポリシーの変更」を参照してください。

カスタマー管理の CMK を管理する方法の詳細は、AWS KMS デベロッパーガイドの「カスタマーマスターキー (CMK)」を参照してください。

次のセクションでは、AWS マネジメントコンソールと AWS CLI を使用して、暗号化ファイルシステムを作成する方法について説明します。

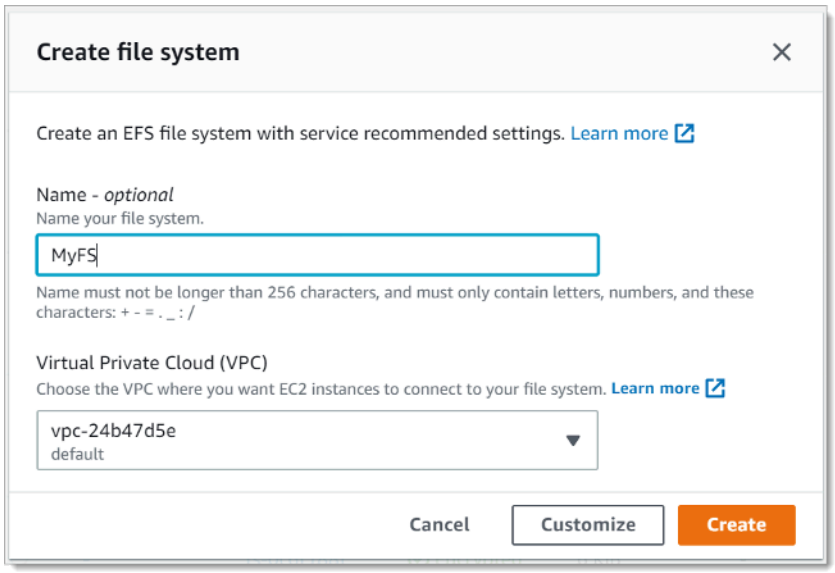
## AWS マネジメントコンソールを使用した暗号化ファイルシステムの作成

AWS マネジメントコンソールを使用して、暗号化された Amazon EFS ファイルシステムを作成するには、以下の手順に従います。

### ステップ 1.ファイルシステムの設定

このステップでは、ライフサイクル管理、パフォーマンスモード、スループットモード、保管中のデータの暗号化など、ファイルシステムの全般的な設定を行います。

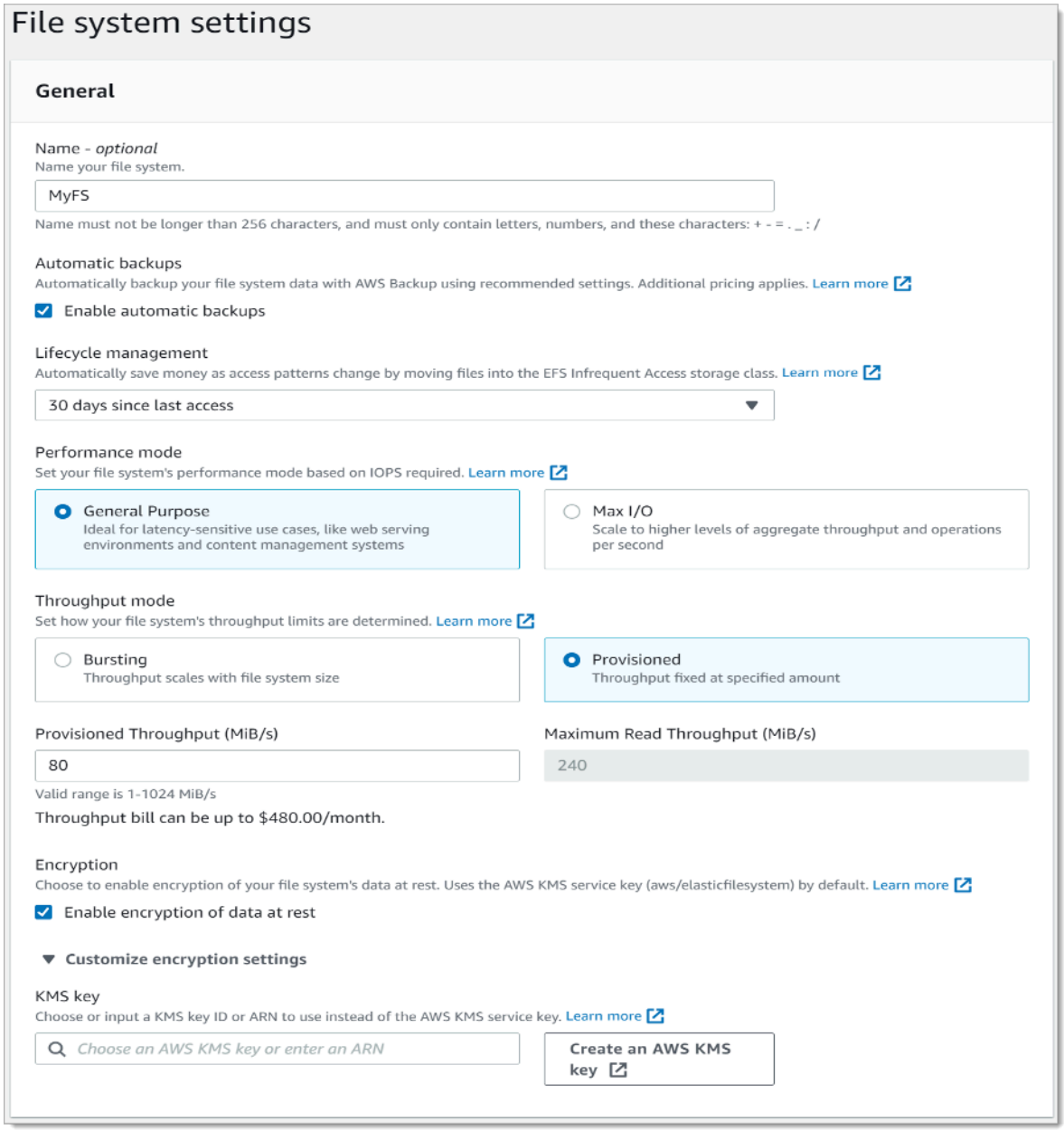
1. AWS マネジメントコンソールにサインインし、Amazon EFS コンソールを開きます。
2. [ファイルシステムの作成] をクリックして、[ファイルシステムの作成] ダイアログボックスを表示します。デフォルトで暗号化を有効にするなど、推奨設定を使用してファイルシステムを作成する方法の詳細は、「Amazon EFS ファイルシステムを作成する」を参照してください。




EFS ファイルシステムの作成

3. (オプション) サービスの推奨設定を使用してファイルシステムを作成する代わりに、カスタマイズしたファイルシステムを作成するには、[カスタマイズ] を選択します。

   [ファイルシステムの設定] ページが表示されます。




EFS ファイルシステムの作成: 全般的な設定

4. [全般] 設定で、以下の詳細を入力します。
   - (オプション) [名前] にファイルシステムの名前を入力します。
   - [自動バックアップ] は、デフォルトで有効になっています。チェックボックスをオフにすると、自動バックアップをオフにすることができます。詳細については、「Amazon EFS での AWS Backup の使用」を参照してください。
   - [ライフサイクル管理] で、ライフサイクル管理ポリシーを選択します。Amazon EFS のライフサイクル管理では、ファイルシステムに対して費用対効果の高いファイルストレージが自動的

に管理されます。有効にすると、ライフサイクル管理により、設定された期間アクセスされなかったファイルは、低頻度アクセス (IA) ストレージクラスに移行されます。期間は、ライフサイクルポリシーを使用して定義します。ライフサイクル管理を有効にしない場合は、[なし] を選択します。詳細は、Amazon EFS ユーザーガイドの「EFS のライフサイクル管理」を参照してください。

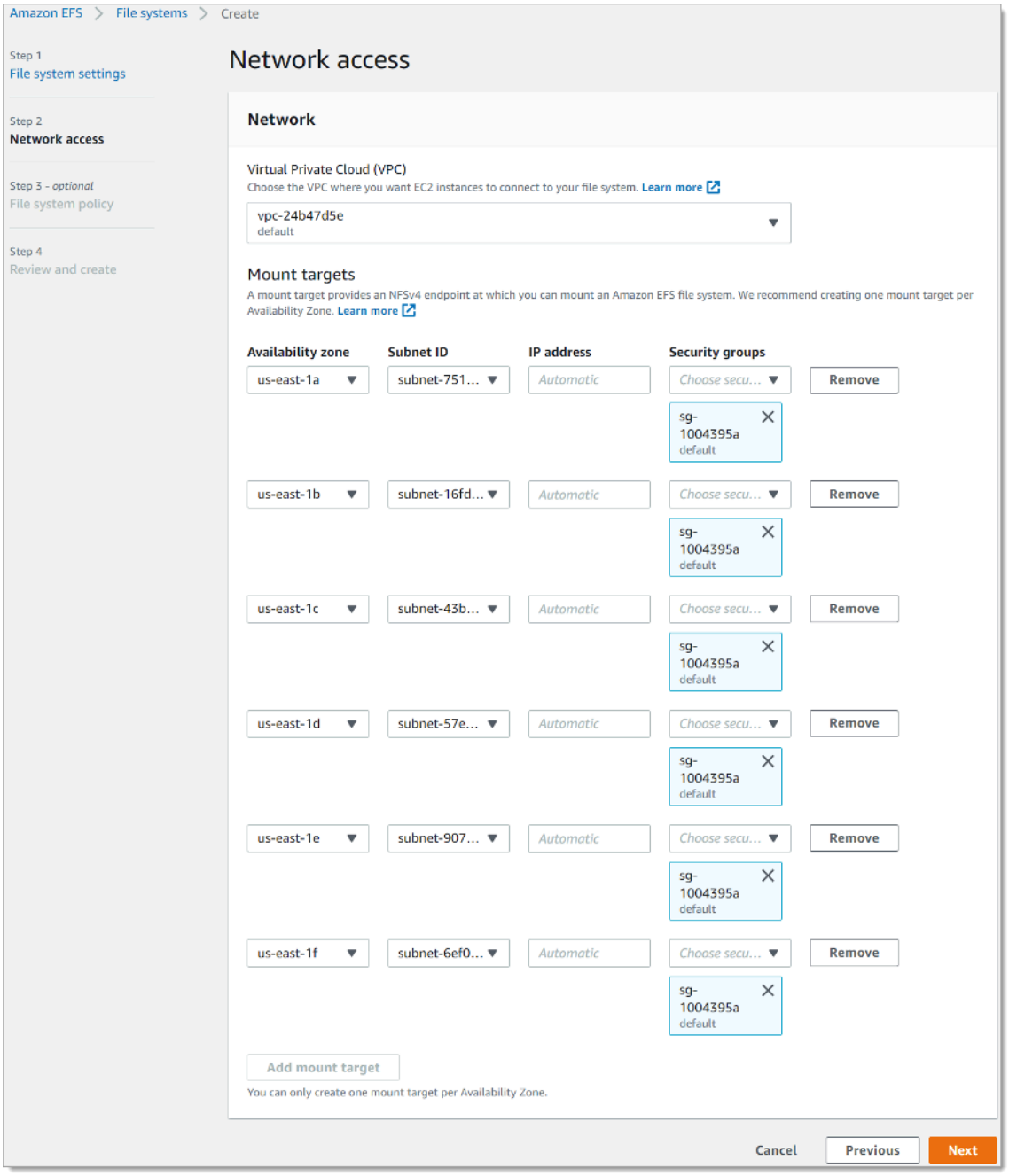
- [パフォーマンスモード] で、デフォルトの [汎用] モードか [最大 I/O] モードのいずれかを選択します。詳細は、Amazon EFS ユーザーガイドの「パフォーマンスモード」を参照してください。
- [スループットモード] で、デフォルトの [バースト] モードか [プロビジョニング済み] モードのいずれかを選択します。
- [プロビジョニング済み] を選択した場合は、[プロビジョニングされたスループット (MiB/s)] フィールドが表示されます。ファイルシステムにプロビジョニングするスループットの量を入力します。スループットを入力すると、フィールドの横に月額コストの推定額が表示されます。詳細は、Amazon EFS ユーザーガイドの「スループットモード」を参照してください。
- [暗号化] について、保管中のデータの暗号化はデフォルトで有効になっています。デフォルトでは、AWS Key Management Service (AWS KMS) の EFS サービスキー (`aws/elasticfilesystem`) が使用されます。暗号化に別の KMS キーを使用するには、[暗号化設定のカスタマイズ] を展開して、リストからキーを選択します。または、使用する KMS キーの KMS キー ID または Amazon リソースネーム (ARN) を [KMS キー] に入力します。

  新しいキーを作成する必要がある場合は、[AWS KMS キーを作成] をクリックして AWS KMS コンソールを起動し、新しいキーを作成します。

5. (オプション) [タグを追加] をクリックして、キーと値のペアをファイルシステムに追加します。
6. [次へ] をクリックして、設定プロセスの [ネットワークアクセス] ステップに進みます。

## ステップ 2.ネットワークアクセスの設定

このステップでは、Virtual Private Cloud (VPC) やマウントターゲットなど、ファイルシステムのネットワーク設定を構成します。マウントターゲットごとに、アベイラビリティーゾーン、サブネット、IP アドレス、セキュリティグループを設定します。




EFS ファイルシステムの作成: ネットワークアクセス

1. [Virtual Private Cloud (VPC)] で、EC2 インスタンスとファイルシステムを接続する仮想プライベートクラウドを選択します。詳細は、Amazon EFS ユーザーガイドの「ファイルシステムネットワークのアクセシビリティの管理」を参照してください。
   - [アベイラビリティーゾーン] – デフォルトでは、AWS リージョンの各アベイラビリティーゾーンにマウントターゲットが設定されます。特定のアベイラビリティーゾーンにマウントターゲットを作成しない場合は、[削除] をクリックしてそのアベイラビリティーゾーンのマウント

ターゲットを削除します。ファイルシステムへのアクセスの予定があるすべてのアベイラビリティーゾーンにマウントターゲットを作成してください。作成してもコストは発生しません。

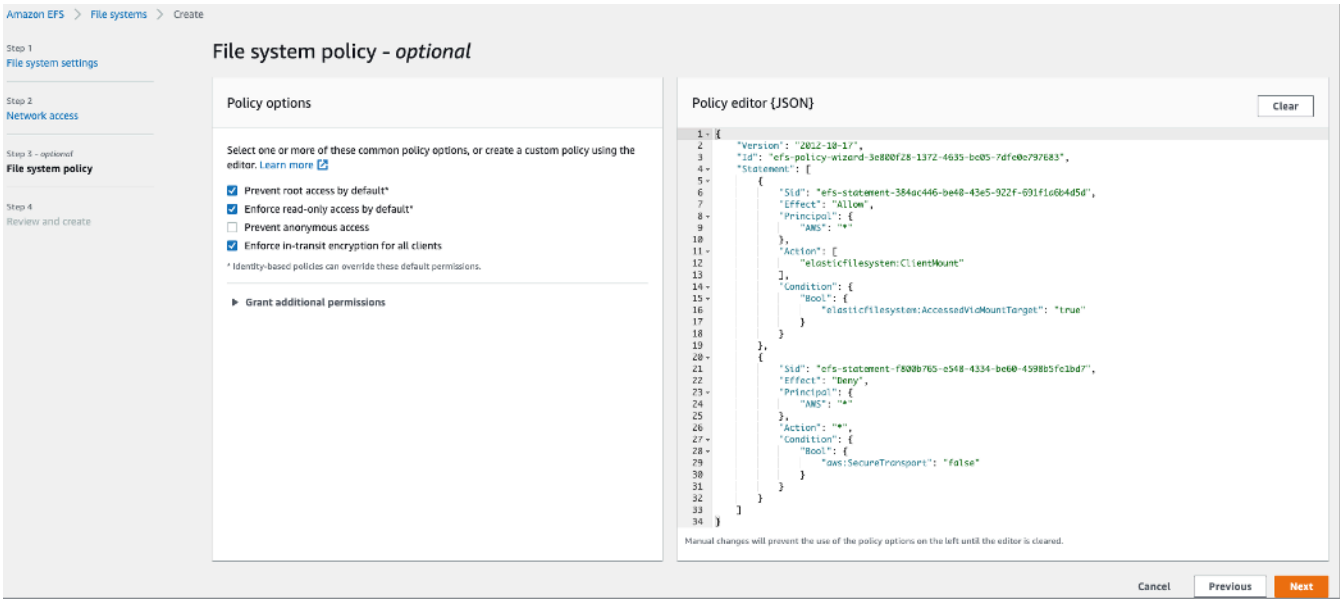
- [サブネット ID] – アベイラビリティーゾーンで使用可能なサブネットから選択します。デフォルトのサブネットはあらかじめ選択されています。ベストプラクティスとして、選択したサブネットをセキュリティ要件に基づいてパブリックまたはプライベートにしてください。
- [IP アドレス] – デフォルトでは、Amazon EFS によって、サブネット内の使用可能な IP アドレスから自動で選択されます。または、サブネット内の特定の IP アドレスを入力することができます。マウントターゲットは、保有する IP アドレスは 1 つですが、冗長性があり、可用性の高いネットワークリソースです。
- [セキュリティグループ] – マウントターゲット に 1 つ以上のセキュリティグループを指定できます。ベストプラクティスとして、セキュリティグループは EFS マウントの目的 (NFS ポート 2049) にのみ使用されるように、またインバウンドルールは、他の VPC CIDR ブロック範囲からはポート 2049 のみを許可するか、EFS にアクセスする必要があるリソースのソースとしてセキュリティグループを使用するようにしてください。詳細は、Amazon EFS ユーザーガイドの「Amazon EC2 インスタンスとマウントターゲットのセキュリティグループの使用」を参照してください。

  他のセキュリティグループを追加する、もしくはセキュリティグループを変更するには、[セキュリティグループの選択] を選んで、リストから他のセキュリティグループを追加します。デフォルトのセキュリティグループを使用しない場合は、削除してかまいません。詳細は、Amazon EFS ユーザーガイドの「セキュリティグループの作成」を参照してください。

2. [マウントターゲットを追加] をクリックして、マウントターゲットがないアベイラビリティーゾーンにマウントターゲットを作成します。各アベイラビリティーゾーンにマウントターゲットが設定されている場合は、このボタンは使用できません。
3. [次へ] をクリックして続行します。[ファイルシステムポリシー] ページが表示されます。

## ステップ 3.ファイルシステムポリシーの作成

このステップでは、ファイルシステムへの NFS クライアントのアクセスを制御するファイルシステムポリシーを作成します。EFS のファイルシステムポリシーは、ファイルシステムへの NFS クライアントのアクセスを制御するために使用する IAM リソースポリシーです。詳細は、Amazon EFS ユーザーガイドの「IAM を使用した Amazon EFS への NFS アクセスのコントロール」を参照してください。

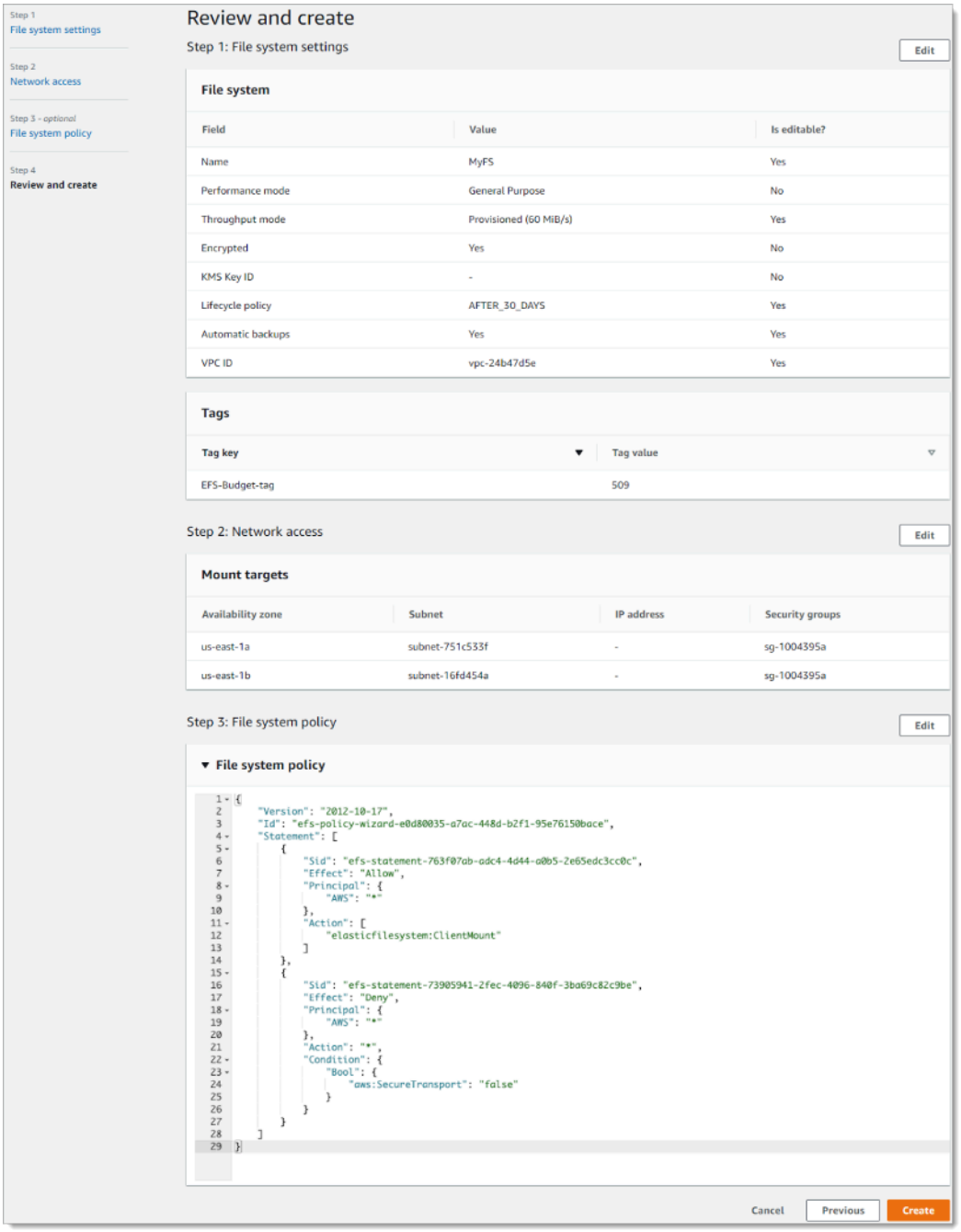



EFS ファイルシステムの作成: ファイルシステムポリシー

1. [ポリシーオプション] で、使用可能な事前構成済みポリシーオプションを以下から選択することをお勧めします。
   - [デフォルトでルートアクセスを禁止する]
   - [デフォルトで読み取り専用アクセスを強制する]
   - [すべてのクライアントに転送時の暗号化を強制する]
2. [追加のアクセス許可を付与] を使用して、別の AWS アカウントを含む追加の IAM プリンシパルにファイルシステムのアクセス権限を付与します。[追加] をクリックして、アクセス権限を付与するエンティティのプリンシパル ARN を入力し、[アクセス許可] をクリックして付与します。
3. 事前設定済みのポリシーのカスタマイズや、要件に基づく独自ポリシーの作成には、[ポリシーエディタ] を使用します。事前設定済みのポリシーのいずれかを選択すると、JSON のポリシー定義が [ポリシーエディタ] に表示されます。
4. [次へ] をクリックして続行します。[確認して作成する] ページが表示されます。

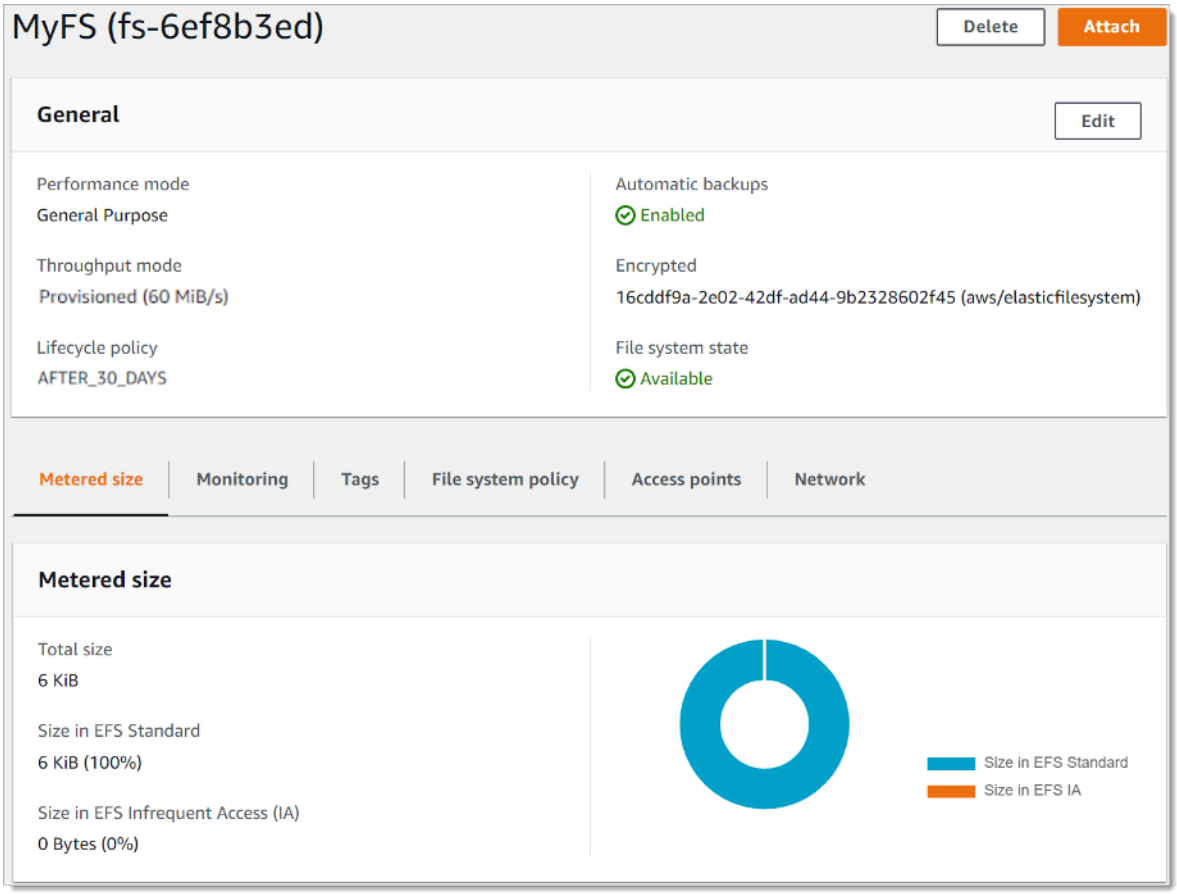
## ステップ 4.確認して作成する

このステップでは、ファイルシステムの設定を確認し、必要に応じて変更を加えて、ファイルシステムを作成します。




EFS ファイルシステムの作成 : 確認して作成する

1. それぞれのファイルシステム設定グループを確認します。[編集] をクリックすると、それぞれのグループに変更を加えることができます。
2. [作成] をクリックしてファイルシステムを作成し、[ファイルシステム] ページに戻ります。
3. [ファイルシステム] ページに、ファイルシステムとその設定の詳細が次のように表示されます。




ファイルシステム

## AWS CLI を使用した暗号化ファイルシステムの作成

AWS CLI を使用して暗号化ファイルシステムを作成する場合、追加のパラメータを使用して暗号化ステータスとカスタマー管理の CMK を設定できます。AWS CLI の最新バージョンを使用していることを確認してください。AWS CLI のアップグレード方法については、AWS コマンドラインインターフェイスユーザーガイドの「AWS CLI のインストール、更新、アンインストール」を参照してください。

`CreateFileSystem` のオペレーションでは、`--encrypted` パラメータはブール値で、暗号化ファイルシステムの作成に必要です。`--kms-key-id` は、カスタマー管理の CMK を使用し、キーのエイリアスまたは ARN を含める場合にのみ必要です。AWS 管理の CMK を使用している場合は、このパラメータを含めないでください。

```
$ aws efs create-file-system \
```

```
        --creation-token $(uuidgen) \
        --performance-mode generalPurpose \
        --encrypted \
        --kms-key-id user/customer-managedCMKalias
```

AWS マネジメントコンソール、AWS CLI、AWS SDK、または Amazon EFS API を使用して Amazon EFS ファイルシステムを作成する方法の詳細は、Amazon EFS ユーザーガイドの「Amazon Elastic File System とは」を参照してください。

## 保管中のデータの暗号化を強制する

暗号化が I/O レイテンシーとスループットに与える影響は最小限です。暗号化と復号化は、ユーザーおよびアプリケーション、サービスに対して透過的です。すべてのデータとメタデータは、お客様の代わりに Amazon EFS によって、ディスクに書き込まれる前に暗号化され、クライアントから読み取られる前に復号化されます。暗号化ファイルシステムにアクセスするために、クライアントツールやアプリケーション、もしくはサービスを変更する必要はありません。

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やワークロード、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。Amazon EFS ファイルシステムリソースに対して、AWS Identity and Access Management (IAM) ID ベースポリシーを使用して保管中のデータの暗号化を強制できます。IAM 条件キーを使用すると、暗号化されていない EFS ファイルシステムがユーザーによって作成されるのを阻止できます。

たとえば、暗号化された EFS ファイルシステムのみの作成をユーザーに明示的に許可する IAM ポリシーでは、Effect (効果)、Action (アクション)、Condition (条件) を次のように組み合わせて使用します。

- `Effect` は、`Allow` です。
- `Action` は、`elasticfilesystem:CreateFileSystem` です。
- `Condition elasticfilesystem:Encrypted` は、`true` です。

次の例は、暗号化ファイルシステムのみを作成することをプリンシパルに許可する IAM ID ベースポリシーを示しています。

```
{
  “Version”: “2012-10-17”,
  “Statement”: [
    {
```

```
      “Sid”: “VisualEditior0”,
      “Effect”: “Allow”,
      “Action”: “elasticfilesystem:CreateFileSystem”,
      “Condition”: {
        “Bool”: {
          “elasticfilesystem:Encrypted”: “true”
        }
      },
      “Resource”: “*”
    }
}
```

Resource 属性が * に設定されている場合は、作成されるすべての EFS リソースに IAM ポリシーが適用されることを意味します。データ分類が必要な EFS リソースのサブセットにのみ IAM ポリシーを強制するために、タグに基づく条件付き属性を追加することができます。

また、組織内のすべての AWS アカウントまたは OU に対してサービスコントロールポリシーを使用することで、暗号化された Amazon EFS ファイルシステムを AWS Organizations レベルで作成するよう強制することも可能です。AWS Organizations のサービスコントロールポリシーの詳細は、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。

# すべての EFS ファイルシステムの暗号化を要求する IAM ポリシーの作成

コンソール、AWS CLI、または API を使用して、暗号化された Amazon EFS ファイルシステムのみを作成することをユーザーに許可する IAM ID ベースポリシーを作成できます。次の手順では、IAM コンソールを使用してこのようなポリシーを作成し、そのポリシーをアカウント内のユーザーに適用する方法を説明します。

暗号化された EFS ファイルシステムを強制する IAM ポリシーを作成するには:

1. AWS マネジメントコンソールにサインインし、IAM コンソールを開きます。
2. ナビゲーションペインの [アクセス管理] で、[ポリシー] をクリックします。
3. [ポリシーの作成] をクリックして、[ポリシーの作成] ページを表示します。
4. [ビジュアルエディタ] タブで、次の情報を入力します。

   - [サービス] で、[EFS] を選択します。
   - [アクション] で、検索フィールドに create を入力し、[CreateFileSystem] をクリックします。

- [リクエスト条件] で [条件の追加] リンクをクリックし、[条件キー] は `elasticfilesystem:Encrypted` 、[演算子]は `Bool` 、[値] は `true` で検索します。

5. ポリシーの [名前] と [説明] を入力します。[Encrypted] リクエスト条件を含む、ポリシーサマリーを確認します。
6. [ポリシーの作成] をクリックしてポリシーを保存します。

アカウント内のユーザーにポリシーを適用するには:

1. IAM コンソールの [アクセス管理] で、[ユーザー] をクリックします。
2. ポリシーを適用するユーザーを選択します。
3. [アクセス権限の追加] をクリックして [アクセス権限を追加] ページを表示します。
4. [既存のポリシーを直接アタッチ] を選択します。
5. 前の手順で作成した EFS ポリシーの名前を入力します。
6. ポリシーを選択して展開します。次に [{} JSON] をクリックしてポリシーの内容を確認します。次の JSON ポリシーのように表示されます。

```
{
  “Version”: “2012-10-17”,
  “Statement”: [
    {
      “Sid”: “VisualEditior0”,
      “Effect”: “Allow”,
      “Action”: “elasticfilesystem:CreateFileSystem”,
      “Condition”: {
        “Bool”: {
          “elasticfilesystem:Encrypted”: “true”
        }
      },
      “Resource”: “*”
    }
}
```

# 暗号化されていないファイルシステムの検出

組織によっては、暗号化されていない Amazon EFS リソースの特定が必要な場合があります。AWS Config マネージドルールを使って、暗号化されていないファイルシステムを検出できます。AWS

Config には、AWS 管理のルールが用意されています。これは定義済みのカスタマイズ可能なルールで、AWS リソースが一般的なベストプラクティスに準拠しているかどうかを評価し、ルール違反のリソースに NON_COMPLIANT のフラグを付けるために AWS Config によって使用されます。

AWS 管理の Config ルール `efs-encrypted-check` を使用して、Amazon Elastic File System (Amazon EFS) が AWS Key Management Service (AWS KMS) を使ってファイルデータを暗号化するように設定されているかどうかを確認できます。AWS 管理のルールの設定とアクティブ化の詳細は、「AWS Config マネージドルールの使用」を参照してください。

# 転送時のデータの暗号化

業界標準の AES-256 暗号化方式を使ったTransport Layer Security 1.2 (TLS) を使用してすべての NFS トラフィックが転送中に暗号化されるように、ファイルシステムをマウントできます。TLS は、ネットワーク上でやり取りされる情報の暗号化に使用される、業界標準の暗号化プロトコル一式です。AES-256 は、TLS でのデータ送信に使用される 256 ビットの暗号化方式です。AWS では、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で、転送中の暗号化を設定することを推奨しています。

IAM ポリシーを使用して、NFS クライアントから Amazon EFS へのアクセスに対して転送中の暗号化を強制できます。クライアントがファイルシステムに接続するとAmazon EFS は、ファイルシステムの IAM リソースポリシー (ファイルシステムポリシー) とID ベースの IAM ポリシーを評価し、付与するのに適切なファイルシステムアクセス権限を決定します。ファイルシステムリソースポリシーで `aws:SecureTransport` 条件キーを使用すると、EFS ファイルシステムへの接続時に、NFS クライアントに TLS の使用を強制できます。

> **Note**
>
> IAM 認証を使用して NFS クライアントによるアクセスをコントロールするには、EFS マウントヘルパーを使用して Amazon EFS ファイルシステムをマウントする必要があります。詳細は、Amazon EFS ユーザーガイドの「IAM 認証によるマウント」を参照してください。

次の EFS ファイルシステムポリシーの例では、転送中の暗号化が強制されており、次のような特徴があります。

- `effect` は、`allow` です。
- プリンシパルは、すべての IAM エンティティを対象とする `*` に設定されています。
- アクションは `ClientMount`、`ClientWrite`、`ClientRootAccess` に設定されています。
- 権限を付与する条件は `SecureTransport` に設定されています。TLS を使用してファイルシステムに接続する NFS クライアントだけが、アクセス権を付与されます。

```
{
  “Version”: “2012-10-17”,
  “Id”: “ExamplePolicy01”,
  “Statement”: [
    {
```

```
      “Sid”: “VisualEditior0”,
      “Effect”: “Allow”,
      “Principal”: {
        “AWS”: “*”,
      },
      “Action”: [
“elasticfilesystem:ClientRootAccess”,
“elasticfilesystem:ClientMount”,
“elasticfilesystem:ClientWrite”
      ],
      “Condition”: {
        “Bool”: {
          “aws:SecureTransport”: “true”
        }
      }
    }
  ]
}
```

ファイルシステムポリシーは、Amazon EFS コンソールまたは AWS CLI を使用して作成できます。

EFS コンソールを使用してファイルシステムポリシーを作成するには:

1. Amazon EFS コンソールを開きます。
2. [ファイルシステム] を選択します。
3. [ファイルシステム] ページで、ファイルシステムポリシーを作成または編集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. [ファイルシステムポリシー]、[編集] の順にクリックします。 [ファイルシステムポリシー] ページが表示されます。


ファイルシステムポリシーの作成

5. [ポリシーオプション] では、次の使用可能な構成済みポリシーオプションを選択することをお勧めします。

   - [デフォルトでルートアクセスを禁止する]
   - [デフォルトで読み取り専用アクセスを強制する]
   - [すべてのクライアントに転送時の暗号化を強制する]

   事前設定済みのポリシーを選択すると、ポリシーの JSON オブジェクトが [ポリシーエディタ] パネルに表示されます。

6. [追加のアクセス許可を付与] を使用して、別の AWS アカウントを含む追加の IAM プリンシパルにファイルシステムのアクセス権限を付与します。[追加] をクリックして、アクセス権限を付与するエンティティのプリンシパル ARN を入力し、[アクセス許可] をクリックして付与します。

7. 事前設定済みのポリシーのカスタマイズや、要件に基づく独自ポリシーの作成には、[ポリシーエディタ] を使用します。エディタを使用すると、事前設定済みのポリシーのオプションが選択できなくなります。ポリシーの変更を元に戻すには、[消去] をクリックします。

   エディタをクリアすると、事前設定済みのポリシーがまた選択できるようになります。

8. ポリシーの編集または作成が完了したら、[保存] をクリックします。

   ファイルシステムの詳細ページが開き、[ファイルシステムポリシー] にポリシーが表示されます。

   AWS CloudFormation や AWS SDK、Amazon EFS API を直接使用してプログラムでファイルシステムポリシーを作成することもできます。ファイルシステムポリシーの作成の詳細は、Amazon EFS ユーザーガイドの「ファイルシステムポリシーの作成」を参照してください。

## 転送中のデータの暗号化の設定

転送中のデータの暗号化を設定するには、各クライアントに EFS マウントヘルパーをダウンロードすることをお勧めします。EFS マウントヘルパーは、転送中のデータの暗号化の設定など、EFS の使用を簡素化するために AWS が提供するオープンソースのユーティリティです。マウントヘルパーはデフォルトで EFS 推奨マウントオプションを使用します。

EFS マウントヘルパーは、以下の Linux ディストリビューションでサポートされています。

- Amazon Linux 2017.09 以降
- Amazon Linux 2 以降
- Debian 9 以降
- Fedora 28 以降
- Red Hat Enterprise Linux / CentOS 7 以降
- Ubuntu 16.04 以降

転送中のデータの暗号化を設定するには:

1. EFS マウントヘルパーをインストールします。
   - Amazon Linux の場合は、下記のコマンドを使用します。

```
sudo yum install -y amazon-efs-utils
```

- その他の Linux ディストリビューションの場合は、GitHub からダウンロードしてインストールします。

  amazon-efs-utils パッケージは、依存関係がある NFS クライアント (nfs-utils)、ネットワークリレー (stunnel)、OpenSSL、Python を自動的にインストールします。

2. ファイルシステムをマウントします。

```
sudo mount -t efs -o tls file-system-id
efs-mount-point
```

- `mount -t efs` で EFS マウントヘルパーを呼び出します。
- EFS マウントヘルパーを使用してマウントする場合、ファイルシステムの DNS 名またはマウントターゲットの IP アドレスの使用はサポートされていません。代わりにファイルシステム ID を使用してください。
- EFS マウントヘルパーは、AWS が推奨するマウントオプションをデフォルトで使用します。デフォルトのマウントオプションをオーバーライドすることは推奨されませんが、場合によってはオーバーライドも可能で柔軟に対応できます。マウントオプションをオーバーライドする場合は、徹底的にテストして、変更によりファイルシステムのアクセスとパフォーマンスに生じる影響を把握することをお勧めします。
- 次の表に、EFS マウントヘルパーが使用するデフォルトのマウントオプションを示します。

| オプション | 説明 | | | |
|---|---|---|---|---|
| nfsvers=4.1 | NFS プロトコルのバージョン | | | |
| rsize=1048576 | ネットワーク読み取りリクエストごとに NFS クライアントが受信できるデー | | | |

| オプション | 説明 |
| --- | --- |
| | タの最大バイト数です。 |
| wsize=1048576 | ネットワーク書き込みリクエストごとに NFS クライアントが送信できるデータの最大バイト数を設定します。 |
| hard | NFS リクエストがタイムアウトした後の NFS クライアントのリカバリ動作です。これにより、NFS リクエストは、サーバーが応答するまで無期限に再試行されます。 |
| timeo=600 | NFS クライアントが、NFS リクエストを再試行する前にレスポンスを待機する際に使用するタイムアウト値です。デシ秒 (1/10秒) で設定します。 |

| オプション | 説明 |
| --- | --- |
| retrans=2 | NFS クライアントがリカバリアクションを試みる前に、リクエストを再試行する回数です。 |
| noresvport | ネットワーク接続が再確立された場合は新しい TCP 送信元ポートを使用するように NFS クライアントに指示します。 |

- システムの再起動後にファイルシステムを自動的に再マウントするには、次の行を /etc/fstab に追加します。

```
file-system-id efs-mount-point efs _netdev, tls, iam 0 0
```

## 転送中のデータの暗号化の使用

転送中のデータの暗号化を要求する社内ポリシーまたは規制ポリシーの対象となっている組織では、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で転送中のデータの暗号化を使用することをお勧めします。暗号化と復号化は接続レベルで構成され、セキュリティのレイヤーがさらに追加されます。

EFS マウントヘルパーを使用してファイルシステムをマウントすると、クライアントと Amazon EFS 間の TLS 1.2 トンネルが設定されて維持されるため、すべての NFS トラフィックがこの暗号化されたトンネルにルーティングされます。暗号化された TLS 接続の確立に使用される証明書は、Amazon 認証局 (CA) によって署名され、最新の Linux ディストリビューションのほとんどで信

頼されています。また、EFS マウントヘルパーは watchdog プロセスを生成して、各ファイルシステムへのすべてのセキュアなトンネルをモニタリングし、実行中であることを確認します。

EFS マウントヘルパーを使用して Amazon EFS への暗号化された接続を確立した後は、その他のユーザー入力や設定は不要です。暗号化は、ファイルシステムにアクセスするユーザー接続やアプリケーションに対して透過的です。

EFS マウントヘルパーを使用して EFS ファイルシステムへの暗号化された接続が正常にマウントされて確立されると、ファイルシステムがマウントされ、localhost (127.0.0.1) をネットワーク中継として使用して暗号化されたトンネルが確立されたことが mount コマンドの出力に示されます。次の出力例を参照してください。

```
127.0.0.1:/ on efs-mount-point type nfs4

(rw,relatime,vers=4.1,rsize=1048576,wsize=1048576,namlen=255,hard,proto=tcp,port=20059,timeo=6
```

efs-mount-point を EFS ファイルシステムにマップするには、/var/log/amazon/efs にある mount.log ファイルを照会して、最後に成功したマウント操作を見つけます。これは、次の簡単な grep コマンドを使って見つけられます。

```
grep -E "Successfully
mounted.*efs-mount-point"
/var/log/amazon/efs/mount.log | tail -1
```

この grep コマンドの出力では、マウントされた EFS ファイルシステムの DNS 名が返されます。次の出力例を参照してください。

```
2018-03-15 07:03:42,363 - INFO - Successfully mounted
file-system-id.efs.region.amazonaws.com
at efs-mount-point
```

# 結論

Amazon EFS ファイルシステムのデータは、保管中も転送中も暗号化できます。AWS KMS を使って制御や管理ができる CMK を使用して、保管中のデータを暗号化できます。暗号化ファイルシステムの作成は、AWS マネジメントコンソールの Amazon EFS ファイルシステム作成ウィザードでチェックボックスを選択するか、AWS CLI、AWS SDK、または Amazon EFS API の `CreateFileSystem` オペレーションにパラメータを 1 つ追加するだけで簡単に実行できます。

AWS IAM ID ベースのポリシーとファイルシステムポリシーを使用して保管中および転送中に暗号化を強制することで、セキュリティ要件をさらに強化し、コンプライアンスのニーズを満たすことができます。また、暗号化ファイルシステムの使用は、サービスやアプリケーション、ユーザーに対して透過的であり、ファイルシステムのパフォーマンスへの影響は最小限に抑えられています。転送中のデータを暗号化するには、EFS マウントヘルパーを使用して各クライアントで暗号化された TLS トンネルを確立し、クライアントとマウントされた EFS ファイルシステムの間のすべての NFS トラフィックを暗号化します。Amazon EFS データの暗号化の強制は、保管中のデータには IAM ID ポリシーを、転送中のデータには EFS ファイルシステムポリシーをそれぞれ使って、追加コストなしで可能です。

# リソース

- AWS KMS 暗号化の詳細についてのホワイトペーパー
- Amazon EFS ユーザーガイド

# ドキュメント履歴と寄稿者

## ドキュメント履歴

このホワイトペーパーの更新に関する通知を受け取るには、RSS フィードを購読してください。

| update-history-change | update-history-description | update-history-date |
| --- | --- | --- |
| マイナーな更新 | ページレイアウトの調整 | 2021 年 4 月 30 日 |
| ホワイトペーパーの更新 | IAM を使用した保存中および転送中の暗号化の強制を追加 | 2021 年 2 月 22 日 |
| ホワイトペーパーの更新 | 転送中のデータの暗号化を追加 | 2018 年 4 月 1 日 |
| 初版公開 | Amazon EFS 暗号化ファイルシステムを使用した保管中のデータの暗号化を公開 | 2017 年 9 月 1 日 |

> **Note**
>
> RSS 更新を購読するには、使用しているブラウザで RSS プラグインを有効にする必要があります。

## 寄稿者

本書の寄稿者は次のとおりです。

- AWS、ストレージスペシャリストソリューションアーキテクト、Darryl S. Osborne
- Amazon EFS、シニアプロダクトマネージャー、Joseph Travaglini
- AWS、プリンシパルソリューションアーキテクト、Peter Buonora
- AWS、シニアソリューションアーキテクト、Siva Rajamani