aws

API Reference

# Amazon Elastic File System

**API Version 2015-02-01**

# Amazon Elastic File System: API Reference

# Table of Contents

# Welcome

Amazon Elastic File System (Amazon EFS) provides simple, scalable file storage for use with Amazon EC2 Linux and Mac instances in the AWS Cloud. With Amazon EFS, storage capacity is elastic, growing and shrinking automatically as you add and remove files, so that your applications have the storage they need, when they need it. For more information, see the Amazon Elastic File System API Reference and the Amazon Elastic File System User Guide.

This document was last published on June 19, 2026.

# Actions

The following actions are supported:

- CreateAccessPoint
- CreateFileSystem
- CreateMountTarget
- CreateReplicationConfiguration
- CreateTags
- DeleteAccessPoint
- DeleteFileSystem
- DeleteFileSystemPolicy
- DeleteMountTarget
- DeleteReplicationConfiguration
- DeleteTags
- DescribeAccessPoints
- DescribeAccountPreferences
- DescribeBackupPolicy
- DescribeFileSystemPolicy
- DescribeFileSystems
- DescribeLifecycleConfiguration
- DescribeMountTargets
- DescribeMountTargetSecurityGroups
- DescribeReplicationConfigurations
- DescribeTags
- ListTagsForResource
- ModifyMountTargetSecurityGroups
- PutAccountPreferences
- PutBackupPolicy
- PutFileSystemPolicy
- PutLifecycleConfiguration

- TagResource
- UntagResource
- UpdateFileSystem
- UpdateFileSystemProtection

# CreateAccessPoint

Creates an EFS access point. An access point is an application-specific view into an EFS file system that applies an operating system user and group, and a file system path, to any file system request made through the access point. The operating system user and group override any identity information provided by the NFS client. The file system path is exposed as the access point's root directory. Applications using the access point can only access data in the application's own directory and any subdirectories. A file system can have a maximum of 10,000 access points unless you request an increase. To learn more, see Mounting a file system using EFS access points.

> **Note**
>
> If multiple requests to create access points on the same file system are sent in quick succession, and the file system is near the limit of access points, you may experience a throttling response for these requests. This is to ensure that the file system does not exceed the stated access point limit.

This operation requires permissions for the `elasticfilesystem:CreateAccessPoint` action.

Access points can be tagged on creation. If tags are specified in the creation action, IAM performs additional authorization on the `elasticfilesystem:TagResource` action to verify if users have permissions to create tags. Therefore, you must grant explicit permissions to use the `elasticfilesystem:TagResource` action. For more information, see Granting permissions to tag resources during creation.

## Request Syntax

```
POST /2015-02-01/access-points HTTP/1.1
Content-type: application/json

{
   "ClientToken": "string",
   "FileSystemId": "string",
   "PosixUser": {
      "Gid": number,
      "SecondaryGids": [ number ],
      "Uid": number
   },
   "RootDirectory": {
```

```
      "CreationInfo": {
         "OwnerGid": number,
         "OwnerUid": number,
         "Permissions": "string"
      },
      "Path": "string"
   },
   "Tags": [
      {
         "Key": "string",
         "Value": "string"
      }
   ]
}
```

## URI Request Parameters

The request does not use any URI parameters.

## Request Body

The request accepts the following data in JSON format.

### ClientToken

A string of up to 64 ASCII characters that Amazon EFS uses to ensure idempotent creation.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: `.+`

Required: Yes

### FileSystemId

The ID of the EFS file system that the access point provides access to.

Type: String

Length Constraints: Maximum length of 128.

Pattern: `^(arn:aws[-a-z]*:elasticfilesystem:[0-9a-z-:]+:file-system/fs-[0-9a-f]{8,40}|fs-[0-9a-f]{8,40})$`

Required: Yes

**PosixUser**

The operating system user and group applied to all file system requests made using the access point.

Type: PosixUser object

Required: No

**RootDirectory**

Specifies the directory on the EFS file system that the access point exposes as the root directory of your file system to NFS clients using the access point. The clients using the access point can only access the root directory and below. If the `RootDirectory` > `Path` specified does not exist, Amazon EFS creates it and applies the `CreationInfo` settings when a client connects to an access point. When specifying a `RootDirectory`, you must provide the `Path`, and the `CreationInfo`.

Amazon EFS creates a root directory only if you have provided the CreationInfo: OwnUid, OwnGID, and permissions for the directory. If you do not provide this information, Amazon EFS does not create the root directory. If the root directory does not exist, attempts to mount using the access point will fail.

Type: RootDirectory object

Required: No

**Tags**

Creates tags associated with the access point. Each tag is a key-value pair, each key must be unique. For more information, see Tagging AWS resources in the *AWS General Reference Guide*.

Type: Array of Tag objects

Required: No

## Response Syntax

```
HTTP/1.1 200
Content-type: application/json
```

```
{
   "AccessPointArn": "string",
   "AccessPointId": "string",
   "ClientToken": "string",
   "FileSystemId": "string",
   "LifeCycleState": "string",
   "Name": "string",
   "OwnerId": "string",
   "PosixUser": {
      "Gid": number,
      "SecondaryGids": [ number ],
      "Uid": number
   },
   "RootDirectory": {
      "CreationInfo": {
         "OwnerGid": number,
         "OwnerUid": number,
         "Permissions": "string"
      },
      "Path": "string"
   },
   "Tags": [
      {
         "Key": "string",
         "Value": "string"
      }
   ]
}
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

### AccessPointArn

The unique Amazon Resource Name (ARN) associated with the access point.

Type: String

Length Constraints: Maximum length of 128.

Pattern: `^arn:aws[-a-z]*:elasticfilesystem:[0-9a-z-:]+:access-point/fsap-[0-9a-f]{8,40}$`

### AccessPointId

The ID of the access point, assigned by Amazon EFS.

Type: String

Length Constraints: Maximum length of 128.

Pattern: `^(arn:aws[-a-z]*:elasticfilesystem:[0-9a-z-:]+:access-point/fsap-[0-9a-f]{8,40}|fsap-[0-9a-f]{8,40})$`

### ClientToken

The opaque string specified in the request to ensure idempotent creation.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: `.+`

### FileSystemId

The ID of the EFS file system that the access point applies to.

Type: String

Length Constraints: Maximum length of 128.

Pattern: `^(arn:aws[-a-z]*:elasticfilesystem:[0-9a-z-:]+:file-system/fs-[0-9a-f]{8,40}|fs-[0-9a-f]{8,40})$`

### LifeCycleState

Identifies the lifecycle phase of the access point.

Type: String

Valid Values: `creating | available | updating | deleting | deleted | error`

### Name

The name of the access point. This is the value of the `Name` tag.

Type: String

**OwnerId**

Identifies the AWS account that owns the access point resource.

Type: String

Length Constraints: Maximum length of 14.

Pattern: `^(\d{12})|(\d{4}-\d{4}-\d{4})$`

**PosixUser**

The full POSIX identity, including the user ID, group ID, and secondary group IDs on the access point that is used for all file operations by NFS clients using the access point.

Type: PosixUser object

**RootDirectory**

The directory on the EFS file system that the access point exposes as the root directory to NFS clients using the access point.

Type: RootDirectory object

**Tags**

The tags associated with the access point, presented as an array of Tag objects.

Type: Array of Tag objects

## Errors

For information about the errors that are common to all actions, see Common Error Types.

**AccessPointAlreadyExists**

Returned if the access point that you are trying to create already exists, with the creation token you provided in the request.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 409

**AccessPointLimitExceeded**

Returned if the AWS account has already created the maximum number of access points allowed per file system. For more informaton, see https://docs.aws.amazon.com/efs/latest/ug/limits.html#limits-efs-resources-per-account-per-region.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 403

**BadRequest**

Returned if the request is malformed or contains an error such as an invalid parameter value or a missing required parameter.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end

user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 400

**FileSystemNotFound**

Returned if the specified `FileSystemId` value doesn't exist in the requester's AWS account.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 404

**IncorrectFileSystemLifeCycleState**

Returned if the file system's lifecycle state is not "available".

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 409

**InternalServerError**

Returned if an error occurred on the server side.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 500

**ThrottlingException**

Returned when the `CreateAccessPoint` API action is called too quickly and the number of Access Points on the file system is nearing the limit of 120.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 429

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4

- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

# CreateFileSystem

Creates a new, empty file system. The operation requires a creation token in the request that Amazon EFS uses to ensure idempotent creation (calling the operation with same creation token has no effect). If a file system does not currently exist that is owned by the caller's AWS account with the specified creation token, this operation does the following:

- Creates a new, empty file system. The file system will have an Amazon EFS assigned ID, and an initial lifecycle state `creating`.
- Returns with the description of the created file system.

Otherwise, this operation returns a `FileSystemAlreadyExists` error with the ID of the existing file system.

> **Note**
>
> For basic use cases, you can use a randomly generated UUID for the creation token.

The idempotent operation allows you to retry a `CreateFileSystem` call without risk of creating an extra file system. This can happen when an initial call fails in a way that leaves it uncertain whether or not a file system was actually created. An example might be that a transport level timeout occurred or your connection was reset. As long as you use the same creation token, if the initial call had succeeded in creating a file system, the client can learn of its existence from the `FileSystemAlreadyExists` error.

For more information, see Creating a file system in the *Amazon EFS User Guide*.

> **Note**
>
> The `CreateFileSystem` call returns while the file system's lifecycle state is still `creating`. You can check the file system creation status by calling the DescribeFileSystems operation, which among other things returns the file system state.

This operation accepts an optional `PerformanceMode` parameter that you choose for your file system. We recommend `generalPurpose` `PerformanceMode` for all file systems. The `maxIO` mode is a previous generation performance type that is designed for highly parallelized workloads

that can tolerate higher latencies than the `generalPurpose` mode. `MaxIO` mode is not supported for One Zone file systems or file systems that use Elastic throughput.

The `PerformanceMode` can't be changed after the file system has been created. For more information, see Amazon EFS performance modes.

You can set the throughput mode for the file system using the `ThroughputMode` parameter.

After the file system is fully created, Amazon EFS sets its lifecycle state to `available`, at which point you can create one or more mount targets for the file system in your VPC. For more information, see CreateMountTarget. You mount your Amazon EFS file system on an EC2 instances in your VPC by using the mount target. For more information, see Amazon EFS: How it Works.

This operation requires permissions for the `elasticfilesystem:CreateFileSystem` action.

File systems can be tagged on creation. If tags are specified in the creation action, IAM performs additional authorization on the `elasticfilesystem:TagResource` action to verify if users have permissions to create tags. Therefore, you must grant explicit permissions to use the `elasticfilesystem:TagResource` action. For more information, see Granting permissions to tag resources during creation.

## Request Syntax

```
POST /2015-02-01/file-systems HTTP/1.1
Content-type: application/json

{
   "AvailabilityZoneName": "string",
   "Backup": boolean,
   "CreationToken": "string",
   "Encrypted": boolean,
   "KmsKeyId": "string",
   "PerformanceMode": "string",
   "ProvisionedThroughputInMibps": number,
   "Tags": [
      {
         "Key": "string",
         "Value": "string"
      }
   ],
   "ThroughputMode": "string"
}
```

## URI Request Parameters

The request does not use any URI parameters.

## Request Body

The request accepts the following data in JSON format.

**AvailabilityZoneName**

For One Zone file systems, specify the AWS Availability Zone in which to create the file system. Use the format `us-east-1a` to specify the Availability Zone. For more information about One Zone file systems, see EFS file system types in the *Amazon EFS User Guide*.

> **Note**
>
> One Zone file systems are not available in all Availability Zones in AWS Regions where Amazon EFS is available.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: `.+`

Required: No

**Backup**

Specifies whether automatic backups are enabled on the file system that you are creating. Set the value to `true` to enable automatic backups. If you are creating a One Zone file system, automatic backups are enabled by default. For more information, see Automatic backups in the *Amazon EFS User Guide*.

Default is `false`. However, if you specify an `AvailabilityZoneName`, the default is `true`.

> **Note**
>
> AWS Backup is not available in all AWS Regions where Amazon EFS is available.

Type: Boolean

Required: No

### CreationToken

A string of up to 64 ASCII characters. Amazon EFS uses this to ensure idempotent creation.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: `.+`

Required: Yes

### Encrypted

A Boolean value that, if true, creates an encrypted file system. When creating an encrypted file system, you have the option of specifying an existing AWS Key Management Service key (KMS key). If you don't specify a KMS key, then the default KMS key for Amazon EFS, `/aws/elasticfilesystem`, is used to protect the encrypted file system.

Type: Boolean

Required: No

### KmsKeyId

The ID of the KMS key that you want to use to protect the encrypted file system. This parameter is required only if you want to use a non-default KMS key. If this parameter is not specified, the default KMS key for Amazon EFS is used. You can specify a KMS key ID using the following formats:

- Key ID - A unique identifier of the key, for example `1234abcd-12ab-34cd-56ef-1234567890ab`.
- ARN - An Amazon Resource Name (ARN) for the key, for example `arn:aws:kms:us-west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab`.
- Key alias - A previously created display name for a key, for example `alias/projectKey1`.
- Key alias ARN - An ARN for a key alias, for example `arn:aws:kms:us-west-2:444455556666:alias/projectKey1`.

If you use `KmsKeyId`, you must set the CreateFileSystem:Encrypted parameter to true.

> **Important**
>
> EFS accepts only symmetric KMS keys. You cannot use asymmetric KMS keys with Amazon EFS file systems.

Type: String

Length Constraints: Maximum length of 2048.

Pattern: `^([0-9a-f]{8}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{12}|mrk-[0-9a-f]{32}|alias/[a-zA-Z0-9/_-]+|(arn:aws[-a-z]*:kms:[a-z0-9-]+:\d{12}:((key/[0-9a-f]{8}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{12})|(key/mrk-[0-9a-f]{32})|(alias/[a-zA-Z0-9/_-]+))))$`

Required: No

**PerformanceMode**

The performance mode of the file system. We recommend `generalPurpose` performance mode for all file systems. File systems using the `maxIO` performance mode can scale to higher levels of aggregate throughput and operations per second with a tradeoff of slightly higher latencies for most file operations. The performance mode can't be changed after the file system has been created. The `maxIO` mode is not supported on One Zone file systems.

> **Important**
>
> Due to the higher per-operation latencies with Max I/O, we recommend using General Purpose performance mode for all file systems.

Default is `generalPurpose`.

Type: String

Valid Values: `generalPurpose | maxIO`

Required: No

### ProvisionedThroughputInMibps

The throughput, measured in mebibytes per second (MiBps), that you want to provision for a file system that you're creating. Required if `ThroughputMode` is set to `provisioned`. Valid values are 1-3414 MiBps, with the upper limit depending on Region. To increase this limit, contact Support. For more information, see Amazon EFS quotas that you can increase in the *Amazon EFS User Guide*.

Type: Double

Valid Range: Minimum value of 1.0.

Required: No

### Tags

Use to create one or more tags associated with the file system. Each tag is a user-defined key-value pair. Name your file system on creation by including a `"Key":"Name","Value":"{value}"` key-value pair. Each key must be unique. For more information, see Tagging AWS resources in the *AWS General Reference Guide*.

Type: Array of Tag objects

Required: No

### ThroughputMode

Specifies the throughput mode for the file system. The mode can be `bursting`, `provisioned`, or `elastic`. If you set `ThroughputMode` to `provisioned`, you must also set a value for `ProvisionedThroughputInMibps`. After you create the file system, you can decrease your file system's Provisioned throughput or change between the throughput modes, with certain time restrictions. For more information, see Specifying throughput with provisioned mode in the *Amazon EFS User Guide*.

Default is `bursting`.

Type: String

Valid Values: `bursting | provisioned | elastic`

Required: No

## Response Syntax

```
HTTP/1.1 201
Content-type: application/json

{
   "AvailabilityZoneId": "string",
   "AvailabilityZoneName": "string",
   "CreationTime": number,
   "CreationToken": "string",
   "Encrypted": boolean,
   "FileSystemArn": "string",
   "FileSystemId": "string",
   "FileSystemProtection": {
      "ReplicationOverwriteProtection": "string"
   },
   "KmsKeyId": "string",
   "LifeCycleState": "string",
   "Name": "string",
   "NumberOfMountTargets": number,
   "OwnerId": "string",
   "PerformanceMode": "string",
   "ProvisionedThroughputInMibps": number,
   "SizeInBytes": {
      "Timestamp": number,
      "Value": number,
      "ValueInArchive": number,
      "ValueInIA": number,
      "ValueInStandard": number
   },
   "Tags": [
      {
         "Key": "string",
         "Value": "string"
      }
   ],
   "ThroughputMode": "string"
}
```

## Response Elements

If the action is successful, the service sends back an HTTP 201 response.

The following data is returned in JSON format by the service.

### AvailabilityZoneId

The unique and consistent identifier of the Availability Zone in which the file system is located, and is valid only for One Zone file systems. For example, `use1-az1` is an Availability Zone ID for the us-east-1 AWS Region, and it has the same location in every AWS account.

Type: String

### AvailabilityZoneName

Describes the AWS Availability Zone in which the file system is located, and is valid only for One Zone file systems. For more information, see Using EFS storage classes in the *Amazon EFS User Guide*.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: `.+`

### CreationTime

The time that the file system was created, in seconds (since 1970-01-01T00:00:00Z).

Type: Timestamp

### CreationToken

The opaque string specified in the request.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: `.+`

### Encrypted

A Boolean value that, if true, indicates that the file system is encrypted.

Type: Boolean

### FileSystemArn

The Amazon Resource Name (ARN) for the EFS file system, in the format `arn:aws:elasticfilesystem:`*`region`*`:`*`account-id`*`:file-system/`*`file-system-id`* . Example with sample data: `arn:aws:elasticfilesystem:us-west-2:1111333322228888:file-system/fs-01234567`

Type: String

### FileSystemId

The ID of the file system, assigned by Amazon EFS.

Type: String

Length Constraints: Maximum length of 128.

Pattern: `^(arn:aws[-a-z]*:elasticfilesystem:[0-9a-z-:]+:file-system/fs-[0-9a-f]{8,40}|fs-[0-9a-f]{8,40})$`

### FileSystemProtection

Describes the protection on the file system.

Type: FileSystemProtectionDescription object

### KmsKeyId

The ID of an AWS KMS key used to protect the encrypted file system.

Type: String

Length Constraints: Maximum length of 2048.

Pattern: `^([0-9a-f]{8}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{12}|mrk-[0-9a-f]{32}|alias/[a-zA-Z0-9/_-]+|(arn:aws[-a-z]*:kms:[a-z0-9-]+:\d{12}:((key/[0-9a-f]{8}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{12})|(key/mrk-[0-9a-f]{32})|(alias/[a-zA-Z0-9/_-]+))))$`

### LifeCycleState

The lifecycle phase of the file system.

Type: String

Valid Values: `creating | available | updating | deleting | deleted | error`

**Name**

You can add tags to a file system, including a `Name` tag. For more information, see CreateFileSystem. If the file system has a `Name` tag, Amazon EFS returns the value in this field.

Type: String

Length Constraints: Maximum length of 256.

Pattern: `^([\p{L}\p{Z}\p{N}_.:/=+\-@]*)$`

**NumberOfMountTargets**

The current number of mount targets that the file system has. For more information, see CreateMountTarget.

Type: Integer

Valid Range: Minimum value of 0.

**OwnerId**

The AWS account that created the file system.

Type: String

Length Constraints: Maximum length of 14.

Pattern: `^(\d{12})|(\d{4}-\d{4}-\d{4})$`

**PerformanceMode**

The performance mode of the file system.

Type: String

Valid Values: `generalPurpose | maxIO`

**ProvisionedThroughputInMibps**

The amount of provisioned throughput, measured in MiBps, for the file system. Valid for file systems using `ThroughputMode` set to `provisioned`.

Type: Double

Valid Range: Minimum value of 1.0.

**SizeInBytes**

The latest known metered size (in bytes) of data stored in the file system, in its `Value` field, and the time at which that size was determined in its `Timestamp` field. The `Timestamp` value is the integer number of seconds since 1970-01-01T00:00:00Z. The `SizeInBytes` value doesn't represent the size of a consistent snapshot of the file system, but it is eventually consistent when there are no writes to the file system. That is, `SizeInBytes` represents actual size only if the file system is not modified for a period longer than a couple of hours. Otherwise, the value is not the exact size that the file system was at any point in time.

Type: FileSystemSize object

**Tags**

The tags associated with the file system, presented as an array of `Tag` objects.

Type: Array of Tag objects

**ThroughputMode**

Displays the file system's throughput mode. For more information, see Throughput modes in the *Amazon EFS User Guide*.

Type: String

Valid Values: `bursting | provisioned | elastic`

## Errors

For information about the errors that are common to all actions, see Common Error Types.

**BadRequest**

Returned if the request is malformed or contains an error such as an invalid parameter value or a missing required parameter.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 400

**FileSystemAlreadyExists**

Returned if the file system you are trying to create already exists, with the creation token you provided.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 409

**FileSystemLimitExceeded**

Returned if the AWS account has already created the maximum number of file systems allowed per account.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end

user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 403

**InsufficientThroughputCapacity**

Returned if there's not enough capacity to provision additional throughput. This value might be returned when you try to create a file system in provisioned throughput mode, when you attempt to increase the provisioned throughput of an existing file system, or when you attempt to change an existing file system from Bursting Throughput to Provisioned Throughput mode. Try again later.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 503

**InternalServerError**

Returned if an error occurred on the server side.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle.

Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 500

**ThroughputLimitExceeded**

Returned if the throughput mode or amount of provisioned throughput can't be changed because the throughput limit of 1024 MiB/s has been reached.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 400

**UnsupportedAvailabilityZone**

Returned if the requested Amazon EFS functionality is not available in the specified Availability Zone.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 400

# Examples

## Create an encrypted EFS file system

The following example sends a POST request to create a file system in the `us-west-2` Region with automatic backups enabled. The request specifies `myFileSystem1` as the creation token for idempotency.

### Sample Request

```
POST /2015-02-01/file-systems HTTP/1.1
Host: elasticfilesystem.us-west-2.amazonaws.com
x-amz-date: 20140620T215117Z
Authorization: <...>
Content-Type: application/json
Content-Length: 42

{
  "CreationToken" : "myFileSystem1",
  "PerformanceMode" : "generalPurpose",
  "Backup": true,
  "Encrypted": true,
  "Tags":[
      {
         "Key": "Name",
         "Value": "Test Group1"
      }
   ]
}
```

### Sample Response

```
HTTP/1.1 201 Created
x-amzn-RequestId: 01234567-89ab-cdef-0123-456789abcdef
Content-Type: application/json
Content-Length: 319

{
   "ownerId":"251839141158",
   "CreationToken":"myFileSystem1",
   "Encrypted": true,
   "PerformanceMode" : "generalPurpose",
```

```
   "fileSystemId":"fs-01234567",
   "CreationTime":"1403301078",
   "LifeCycleState":"creating",
   "numberOfMountTargets":0,
   "SizeInBytes":{
       "Timestamp": 1403301078,
       "Value": 29313618372,
       "ValueInArchive": 201156,
       "ValueInIA": 675432,
       "ValueInStandard": 29312741784
   },
   "Tags":[
      {
        "Key": "Name",
        "Value": "Test Group1"
      }
   ],
   "ThroughputMode": "elastic"
}
```

## Create an encrypted EFS file system with One Zone availability

The following example sends a POST request to create a file system in the `us-west-2` Region with automatic backups enabled. The file system will have One Zone storage in the `us-west-2b` Availability Zone.

### Sample Request

```
POST /2015-02-01/file-systems HTTP/1.1
Host: elasticfilesystem.us-west-2.amazonaws.com
x-amz-date: 20140620T215117Z
Authorization: <...>
Content-Type: application/json
Content-Length: 42

{
  "CreationToken" : "myFileSystem2",
  "PerformanceMode" : "generalPurpose",
  "Backup": true,
  "AvailabilityZoneName": "us-west-2b",
  "Encrypted": true,
  "ThroughputMode": "elastic",
  "Tags":[
```

```
      {
         "Key": "Name",
         "Value": "Test Group1"
      }
   ]
}
```

## Sample Response

```
HTTP/1.1 201 Created
x-amzn-RequestId: 01234567-89ab-cdef-0123-456789abcdef
Content-Type: application/json
Content-Length: 319

{
   "ownerId":"251839141158",
   "CreationToken":"myFileSystem1",
   "Encrypted": true,
   "AvailabilityZoneId": "usew2-az2",
   "AvailabilityZoneName": "us-west-2b",
   "PerformanceMode" : "generalPurpose",
   "fileSystemId":"fs-01234567",
   "CreationTime":"1403301078",
   "LifeCycleState":"creating",
   "numberOfMountTargets":0,
   "SizeInBytes":{
       "Timestamp": 1403301078,
       "Value": 29313618372,
       "ValueInArchive": 201156,
       "ValueInIA": 675432,
       "ValueInStandard": 29312741784
   },
   "Tags":[
      {
        "Key": "Name",
        "Value": "Test Group1"
      }
   ],
   "ThroughputMode": "elastic"
}
```

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

# CreateMountTarget

Creates a mount target for a file system. You can then mount the file system on EC2 instances by using the mount target. For more information about mount targets, see Managing mount targets.

To create a mount target for an EFS file system, the file system's lifecycle state must be `available`. For more information, see DescribeFileSystems.

After creating the mount target, Amazon EFS returns a response that includes a `MountTargetId` and an IP address (`IpAddress` or `Ipv6Address`). You use this IP address when mounting the file system in an EC2 instance. You can also use the mount target's DNS name when mounting the file system. The EC2 instance on which you mount the file system by using the mount target can resolve the mount target's DNS name to its IP address. For more information, see How Amazon EFS works.

Note that you can create mount targets for a file system in only one VPC, and there can be only one mount target per Availability Zone. For more information, see Creating mount targets.

If the request satisfies the requirements, Amazon EFS does the following:

- Creates a new mount target in the specified subnet.
- Creates a new network interface in the subnet with the folowing:
  - The description `Mount target` *`fsmt-id`* `for file system` *`fs-id`* where *`fsmt-id`* is the mount target ID, and *`fs-id`* is the `FileSystemId`.
  - The `requesterManaged` property of the network interface set to `true`, and the `requesterId` value set to `EFS`.

  Each mount target has one corresponding requester-managed EC2 network interface. After the network interface is created, Amazon EFS sets the `NetworkInterfaceId` field in the mount target's description to the network interface ID, and the IP address to its address. If network interface creation fails, the entire `CreateMountTarget` operation fails.

> **Note**
>
> The `CreateMountTarget` call returns only after creating the network interface, but while the mount target state is still `creating`, you can check the mount target creation status by

calling the DescribeMountTargets operation, which among other things returns the mount target state.

This operation requires permissions for the following action on the file system:

- `elasticfilesystem:CreateMountTarget`

This operation also requires permissions for the following Amazon EC2 actions:

- `ec2:DescribeSubnets`
- `ec2:DescribeNetworkInterfaces`
- `ec2:CreateNetworkInterface`

## Request Syntax

```
POST /2015-02-01/mount-targets HTTP/1.1
Content-type: application/json

{
   "FileSystemId": "string",
   "IpAddress": "string",
   "IpAddressType": "string",
   "Ipv6Address": "string",
   "SecurityGroups": [ "string" ],
   "SubnetId": "string"
}
```

## URI Request Parameters

The request does not use any URI parameters.

## Request Body

The request accepts the following data in JSON format.

### FileSystemId

The ID of the file system for which to create the mount target.

Type: String

Length Constraints: Maximum length of 128.

Pattern: `^(arn:aws[-a-z]*:elasticfilesystem:[0-9a-z-:]+:file-system/fs-[0-9a-f]{8,40}|fs-[0-9a-f]{8,40})$`

Required: Yes

**IpAddress**

If the `IpAddressType` for the mount target is IPv4 ( `IPV4_ONLY` or `DUAL_STACK`), then specify the IPv4 address to use. If you do not specify an `IpAddress`, then Amazon EFS selects an unused IP address from the subnet specified for `SubnetId`.

Type: String

Length Constraints: Minimum length of 7. Maximum length of 15.

Pattern: `^[0-9]{1,3}\.[0-9]{1,3}\.[0-9]{1,3}\.[0-9]{1,3}$`

Required: No

**IpAddressType**

The IP address type for the mount target. The possible values are `IPV4_ONLY` (only IPv4 addresses), `IPV6_ONLY` (only IPv6 addresses), and `DUAL_STACK` (dual-stack, both IPv4 and IPv6 addresses). If you don't specify an `IpAddressType`, then `IPV4_ONLY` is used.

> **Note**
>
> The `IPAddressType` must match the IP type of the subnet. Additionally, the `IPAddressType` parameter overrides the value set as the default IP address for the subnet in the VPC. For example, if the `IPAddressType` is `IPV4_ONLY` and `AssignIpv6AddressOnCreation` is `true`, then IPv4 is used for the mount target. For more information, see Modify the IP addressing attributes of your subnet.

Type: String

Valid Values: `IPV4_ONLY | IPV6_ONLY | DUAL_STACK`

Required: No

### Ipv6Address

If the `IPAddressType` for the mount target is IPv6 (`IPV6_ONLY` or `DUAL_STACK`), then specify the IPv6 address to use. If you do not specify an `Ipv6Address`, then Amazon EFS selects an unused IP address from the subnet specified for `SubnetId`.

Type: String

Length Constraints: Minimum length of 3. Maximum length of 39.

Required: No

### SecurityGroups

VPC security group IDs, of the form `sg-xxxxxxxx`. These must be for the same VPC as the subnet specified. The maximum number of security groups depends on account quota. For more information, see Amazon VPC Quotas in the *Amazon VPC User Guide* (see the **Security Groups** table). If you don't specify a security group, then Amazon EFS uses the default security group for the subnet's VPC.

Type: Array of strings

Array Members: Maximum number of 100 items.

Length Constraints: Minimum length of 11. Maximum length of 43.

Pattern: `^sg-[0-9a-f]{8,40}`

Required: No

### SubnetId

The ID of the subnet to add the mount target in. For One Zone file systems, use the subnet that is associated with the file system's Availability Zone. The subnet type must be the same type as the `IpAddressType`.

Type: String

Length Constraints: Minimum length of 15. Maximum length of 47.

Pattern: `^subnet-[0-9a-f]{8,40}$`

Required: Yes

## Response Syntax

```
HTTP/1.1 200
Content-type: application/json

{
   "AvailabilityZoneId": "string",
   "AvailabilityZoneName": "string",
   "FileSystemId": "string",
   "IpAddress": "string",
   "Ipv6Address": "string",
   "LifeCycleState": "string",
   "MountTargetId": "string",
   "NetworkInterfaceId": "string",
   "OwnerId": "string",
   "SubnetId": "string",
   "VpcId": "string"
}
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

### AvailabilityZoneId

The unique and consistent identifier of the Availability Zone that the mount target resides in. For example, `use1-az1` is an AZ ID for the us-east-1 Region and it has the same location in every AWS account.

Type: String

### AvailabilityZoneName

The name of the Availability Zone in which the mount target is located. Availability Zones are independently mapped to names for each AWS account. For example, the Availability Zone `us-east-1a` for your AWS account might not be the same location as `us-east-1a` for another AWS account.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: `.+`

**FileSystemId**

The ID of the file system for which the mount target is intended.

Type: String

Length Constraints: Maximum length of 128.

Pattern: `^(arn:aws[-a-z]*:elasticfilesystem:[0-9a-z-:]+:file-system/fs-[0-9a-f]{8,40}|fs-[0-9a-f]{8,40})$`

**IpAddress**

The IPv4 address for the mount target.

Type: String

Length Constraints: Minimum length of 7. Maximum length of 15.

Pattern: `^[0-9]{1,3}\.[0-9]{1,3}\.[0-9]{1,3}\.[0-9]{1,3}$`

**Ipv6Address**

The IPv6 address for the mount target.

Type: String

Length Constraints: Minimum length of 3. Maximum length of 39.

**LifeCycleState**

Lifecycle state of the mount target.

Type: String

Valid Values: `creating | available | updating | deleting | deleted | error`

**MountTargetId**

System-assigned mount target ID.

Type: String

Length Constraints: Minimum length of 13. Maximum length of 45.

Pattern: `^fsmt-[0-9a-f]{8,40}$`

**NetworkInterfaceId**

The ID of the network interface that Amazon EFS created when it created the mount target.

Type: String

**OwnerId**

AWS account ID that owns the resource.

Type: String

Length Constraints: Maximum length of 14.

Pattern: `^(\d{12})|(\d{4}-\d{4}-\d{4})$`

**SubnetId**

The ID of the mount target's subnet.

Type: String

Length Constraints: Minimum length of 15. Maximum length of 47.

Pattern: `^subnet-[0-9a-f]{8,40}$`

**VpcId**

The virtual private cloud (VPC) ID that the mount target is configured in.

Type: String

## Errors

For information about the errors that are common to all actions, see Common Error Types.

**AvailabilityZonesMismatch**

Returned if the Availability Zone that was specified for the mount target is different from the file system's Availability Zone. For more information, see Regional and One Zone storage redundancy.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 400

**BadRequest**

Returned if the request is malformed or contains an error such as an invalid parameter value or a missing required parameter.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 400

**FileSystemNotFound**

Returned if the specified `FileSystemId` value doesn't exist in the requester's AWS account.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 404

**IncorrectFileSystemLifeCycleState**

Returned if the file system's lifecycle state is not "available".

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 409

**InternalServerError**

Returned if an error occurred on the server side.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 500

**IpAddressInUse**

Returned if the request specified an IP address (`IpAddress` or `Ipv6Address`) that is already in use in the subnet.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 409

**MountTargetConflict**

Returned if the mount target would violate one of the specified restrictions based on the file system's existing mount targets.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 409

**NetworkInterfaceLimitExceeded**

The calling account has reached the limit for elastic network interfaces for the specific AWS Region. Either delete some network interfaces or request that the account quota be raised. For

more information, see Amazon VPC Quotas in the *Amazon VPC User Guide* (see the **Network interfaces per Region** entry in the **Network interfaces** table).

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 409

**NoFreeAddressesInSubnet**

Returned no `IpAddress` or `Ipv6Address` was provided in the request and there are no free IP addresses in the specified subnet.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 409

**SecurityGroupLimitExceeded**

Returned if the number of `SecurityGroups` specified in the request is greater than the limit, which is based on account quota. Either delete some security groups or request that the account quota be raised. For more information, see Amazon VPC Quotas in the *Amazon VPC User Guide* (see the **Security Groups** table).

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 400

**SecurityGroupNotFound**

Returned if one of the specified security groups doesn't exist in the subnet's virtual private cloud (VPC).

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 400

**SubnetNotFound**

Returned if there is no subnet with ID `SubnetId` provided in the request.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 400

**UnsupportedAvailabilityZone**

Returned if the requested Amazon EFS functionality is not available in the specified Availability Zone.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 400

# Examples

## Create a mount target at an available IPv4 address on a subnet

The following request specifies only the file system and subnet for the mount target. The target is created at an available IPv4 address on the specified subnet, with the default security group associated with the VPC.

### Sample Request

```
POST /2015-02-01/mount-targets HTTP/1.1
```

```
Host: elasticfilesystem.us-west-2.amazonaws.com
x-amz-date: 20140620T221118Z
Authorization: <...>
Content-Type: application/json
Content-Length: 160

{"SubnetId": "subnet-748c5d03", "FileSystemId": "fs-01234567"}
```

### Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: 01234567-89ab-cdef-0123-456789abcdef
Content-Type: application/json
Content-Length: 252

{
    "OwnerId": "231243201240",
    "MountTargetId": "fsmt-55a4413c",
    "FileSystemId": "fs-01234567",
    "SubnetId": "subnet-01234567",
    "LifeCycleState": "available",
    "IpAddress": "172.31.22.183"
    "NetworkInterfaceId":"eni-1bcb7772"
    "AvailabilityZoneId": "eus1-az2",
    "AvailabilityZoneName": "eu-south-1b",
    "VpcId": "vpc-08d45b31fa009a15e"
}
```

## Create a mount target at a specific IPv4 address

The following request specifies the file system, subnet, security group, and IPv4 address to use for the mount target.

### Sample Request

```
POST /2015-02-01/mount-targets HTTP/1.1
Host: elasticfilesystem.us-west-2.amazonaws.com
x-amz-date: 20140620T221118Z
Authorization: <...>
Content-Type: application/json
Content-Length: 160
```

```
{
   "FileSystemId":"fs-01234567",
   "SubnetId":"subnet-01234567",
   "IpAddress":"10.0.2.42",
   "SecurityGroups":[
      "sg-01234567"
   ]
}
```

### Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: 01234567-89ab-cdef-0123-456789abcdef
Content-Type: application/json
Content-Length: 252

{
   "OwnerId":"251839141158",
   "MountTargetId":"fsmt-9a13661e",
   "FileSystemId":"fs-01234567",
   "SubnetId":"subnet-fd04ff94",
   "LifeCycleState":"available",
   "IpAddress":"10.0.2.42",
   "NetworkInterfaceId":"eni-1bcb7772"
   "AvailabilityZoneId": "eus1-az2",
   "AvailabilityZoneName": "eu-south-1b",
   "VpcId": "vpc-08d45b31fa009a15e"
}
```

## Create a mount target at a specific IPv6 address

The following request specifies the file system, subnet, security group, and IPv6 address to use for the mount target.

### Sample Request

```
POST /2015-02-01/mount-targets HTTP/1.1
Host: elasticfilesystem.us-west-2.amazonaws.com
x-amz-date: 20140620T221118Z
Authorization: <...>
Content-Type: application/json
Content-Length: 160
```

```
{
   "FileSystemId":"fs-01234567",
   "SubnetId":"subnet-01234567",
   "Ipv6Address":"2001:0db8:85a3:0000:0000:8a2e:0370:7334",
   "IpAddressType": "IPV6_ONLY",
   "SecurityGroups":[
      "sg-01234567"
   ]
}
```

### Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: 01234567-89ab-cdef-0123-456789abcdef
Content-Type: application/json
Content-Length: 252

{
   "OwnerId":"251839141158",
   "MountTargetId":"fsmt-9a13661e",
   "FileSystemId":"fs-01234567",
   "SubnetId":"subnet-fd04ff94",
   "LifeCycleState":"available",
   "Ipv6Address":"2001:0db8:85a3:0000:0000:8a2e:0370:7334",
   "NetworkInterfaceId":"eni-1bcb7772"
   "AvailabilityZoneId": "eus1-az2",
   "AvailabilityZoneName": "eu-south-1b",
   "VpcId": "vpc-08d45b31fa009a15e"
}
```

## Create a mount target at an available IPv4 address on dual-stack subnet

The following request specifies the file system, subnet, and address type for the mount target. The target is created at an available IPv4 address on the specified dual-stack subnet, with the default security group associated with the VPC.

### Sample Request

```
POST /2015-02-01/mount-targets HTTP/1.1
Host: elasticfilesystem.us-west-2.amazonaws.com
x-amz-date: 20140620T221118Z
```

```
Authorization: <...>
Content-Type: application/json
Content-Length: 160

{"SubnetId": "subnet-748c5d03", "FileSystemId": "fs-01234567"}
```

### Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: 01234567-89ab-cdef-0123-456789abcdef
Content-Type: application/json
Content-Length: 252

{
    "OwnerId": "251839141158",
    "MountTargetId": "fsmt-55a4413c",
    "FileSystemId": "fs-01234567",
    "SubnetId":"subnet-fd04ff94",
    "LifeCycleState": "available",
    "IpAddress": "172.31.22.183"
    "Ipv6Address": "2a05:d01a:419:8611:3e57:75ab:5719:b238",
    "NetworkInterfaceId": "eni-01234567",
    "AvailabilityZoneId": "eus1-az2",
    "AvailabilityZoneName": "eu-south-1b",
    "VpcId": "vpc-08d45b31fa009a15e"
}
```

## Create mount target at specific Ipv4 and IPv6 addresses on dual-stack subnet

The following request specifies the file system, subnet, security group, IPv4 address, IPv6 address for the mount target. The target is created at the specified IPv4 and IPv6 addresses on a dual-stack subnet.

### Sample Request

```
POST /2015-02-01/mount-targets HTTP/1.1
Host: elasticfilesystem.us-west-2.amazonaws.com
x-amz-date: 20140620T221118Z
Authorization: <...>
Content-Type: application/json
Content-Length: 160
```

```
{
   "FileSystemId":"fs-01234567",
   "SubnetId":"subnet-01234567",
   "IpAddressType": "DUAL_STACK",
   "IpAddress": "10.0.1.25",
   "Ipv6Address":"2001:0db8:85a3:0000:0000:8a2e:0370:7334",
   "SecurityGroups":[
      "sg-01234567"
   ]
}
```

### Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: 01234567-89ab-cdef-0123-456789abcdef
Content-Type: application/json
Content-Length: 252

{
    "OwnerId": "231243201240",
    "MountTargetId": "fsmt-55a4413c",
    "FileSystemId": "fs-01234567",
    "SubnetId": "subnet-01234567",
    "LifeCycleState": "available",
    "IpAddress": "10.0.1.25",
    "Ipv6Address":"2001:0db8:85a3:0000:0000:8a2e:0370:7334",
    "NetworkInterfaceId": "eni-01234567",
    "AvailabilityZoneId": "eus1-az2",
    "AvailabilityZoneName": "eu-south-1b",
    "VpcId": "vpc-08d45b31fa009a15e"
}
```

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4
- AWS SDK for C++
- AWS SDK for Go v2

- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

# CreateReplicationConfiguration

Creates a replication configuration to either a new or existing EFS file system. For more information, see Amazon EFS replication in the *Amazon EFS User Guide*. The replication configuration specifies the following:

- **Source file system** – The EFS file system that you want to replicate.
- **Destination file system** – The destination file system to which the source file system is replicated. There can only be one destination file system in a replication configuration.

> **Note**
>
> A file system can be part of only one replication configuration.

The destination parameters for the replication configuration depend on whether you are replicating to a new file system or to an existing file system, and if you are replicating across AWS accounts. See DestinationToCreate for more information.

This operation requires permissions for the `elasticfilesystem:CreateReplicationConfiguration` action. Additionally, other permissions are required depending on how you are replicating file systems. For more information, see Required permissions for replication in the *Amazon EFS User Guide*.

## Request Syntax

```
POST /2015-02-01/file-systems/SourceFileSystemId/replication-configuration HTTP/1.1
Content-type: application/json

{
   "Destinations": [
      {
         "AvailabilityZoneName": "string",
         "FileSystemId": "string",
         "KmsKeyId": "string",
         "Region": "string",
         "RoleArn": "string"
      }
   ]
```

```
}
```

## URI Request Parameters

The request uses the following URI parameters.

### SourceFileSystemId

Specifies the Amazon EFS file system that you want to replicate. This file system cannot already be a source or destination file system in another replication configuration.

Length Constraints: Maximum length of 128.

Pattern: `^(arn:aws[-a-z]*:elasticfilesystem:[0-9a-z-:]+:file-system/fs-[0-9a-f]{8,40}|fs-[0-9a-f]{8,40})$`

Required: Yes

## Request Body

The request accepts the following data in JSON format.

### Destinations

An array of destination configuration objects. Only one destination configuration object is supported.

Type: Array of DestinationToCreate objects

Required: Yes

## Response Syntax

```
HTTP/1.1 200
Content-type: application/json

{
   "CreationTime": number,
   "Destinations": [
      {
```

```
            "FileSystemId": "string",
            "LastReplicatedTimestamp": number,
            "OwnerId": "string",
            "Region": "string",
            "RoleArn": "string",
            "Status": "string",
            "StatusMessage": "string"
        }
    ],
    "OriginalSourceFileSystemArn": "string",
    "SourceFileSystemArn": "string",
    "SourceFileSystemId": "string",
    "SourceFileSystemOwnerId": "string",
    "SourceFileSystemRegion": "string"
}
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

### CreationTime

Describes when the replication configuration was created.

Type: Timestamp

### Destinations

An array of destination objects. Only one destination object is supported.

Type: Array of Destination objects

### OriginalSourceFileSystemArn

The Amazon Resource Name (ARN) of the original source EFS file system in the replication configuration.

Type: String

### SourceFileSystemArn

The Amazon Resource Name (ARN) of the current source file system in the replication configuration.

Type: String

**SourceFileSystemId**

The ID of the source Amazon EFS file system that is being replicated.

Type: String

Length Constraints: Maximum length of 128.

Pattern: `^(arn:aws[-a-z]*:elasticfilesystem:[0-9a-z-:]+:file-system/fs-[0-9a-f]{8,40}|fs-[0-9a-f]{8,40})$`

**SourceFileSystemOwnerId**

ID of the AWS account in which the source file system resides.

Type: String

Length Constraints: Maximum length of 14.

Pattern: `^(\d{12})|(\d{4}-\d{4}-\d{4})$`

**SourceFileSystemRegion**

The AWS Region in which the source EFS file system is located.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: `^[a-z]{2}-((iso[a-z]{0,1}-)|(gov-)){0,1}[a-z]+-{0,1}[0-9]{0,1}$`

## Errors

For information about the errors that are common to all actions, see Common Error Types.

**BadRequest**

Returned if the request is malformed or contains an error such as an invalid parameter value or a missing required parameter.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 400

**ConflictException**

Returned if the source file system in a replication is encrypted but the destination file system is unencrypted.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 409

**FileSystemLimitExceeded**

Returned if the AWS account has already created the maximum number of file systems allowed per account.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle.

Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 403

**FileSystemNotFound**

Returned if the specified `FileSystemId` value doesn't exist in the requester's AWS account.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 404

**IncorrectFileSystemLifeCycleState**

Returned if the file system's lifecycle state is not "available".

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 409

**InsufficientThroughputCapacity**

Returned if there's not enough capacity to provision additional throughput. This value might be returned when you try to create a file system in provisioned throughput mode, when you

attempt to increase the provisioned throughput of an existing file system, or when you attempt to change an existing file system from Bursting Throughput to Provisioned Throughput mode. Try again later.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 503

**InternalServerError**

Returned if an error occurred on the server side.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 500

**ReplicationNotFound**

Returned if the specified file system does not have a replication configuration.

**ErrorCode**

ReplicationNotFound

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 404

**ThroughputLimitExceeded**

Returned if the throughput mode or amount of provisioned throughput can't be changed because the throughput limit of 1024 MiB/s has been reached.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 400

**UnsupportedAvailabilityZone**

Returned if the requested Amazon EFS functionality is not available in the specified Availability Zone.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle.

Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 400

**ValidationException**

Returned if the AWS Backup service is not available in the AWS Region in which the request was made.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 400

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python

- AWS SDK for Ruby V3

# CreateTags

> **Note**
>
> DEPRECATED - `CreateTags` is deprecated and not maintained. To create tags for EFS resources, use the TagResource API action.

Creates or overwrites tags associated with a file system. Each tag is a key-value pair. If a tag key specified in the request already exists on the file system, this operation overwrites its value with the value provided in the request. If you add the `Name` tag to your file system, Amazon EFS returns it in the response to the DescribeFileSystems operation.

This operation requires permission for the `elasticfilesystem:CreateTags` action.

## Request Syntax

```
POST /2015-02-01/create-tags/FileSystemId HTTP/1.1
Content-type: application/json

{
   "Tags": [
      {
         "Key": "string",
         "Value": "string"
      }
   ]
}
```

## URI Request Parameters

The request uses the following URI parameters.

### FileSystemId

The ID of the file system whose tags you want to modify (String). This operation modifies the tags only, not the file system.

Length Constraints: Maximum length of 128.

Pattern: `^(arn:aws[-a-z]*:elasticfilesystem:[0-9a-z-:]+:file-system/fs-[0-9a-f]{8,40}|fs-[0-9a-f]{8,40})$`

Required: Yes

## Request Body

The request accepts the following data in JSON format.

**Tags**

An array of Tag objects to add. Each Tag object is a key-value pair.

Type: Array of Tag objects

Required: Yes

## Response Syntax

```
HTTP/1.1 204
```

## Response Elements

If the action is successful, the service sends back an HTTP 204 response with an empty HTTP body.

## Errors

For information about the errors that are common to all actions, see Common Error Types.

**BadRequest**

Returned if the request is malformed or contains an error such as an invalid parameter value or a missing required parameter.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 400

**FileSystemNotFound**

Returned if the specified `FileSystemId` value doesn't exist in the requester's AWS account.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 404

**InternalServerError**

Returned if an error occurred on the server side.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 500

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

# DeleteAccessPoint

Deletes the specified access point. After deletion is complete, new clients can no longer connect to the access points. Clients connected to the access point at the time of deletion will continue to function until they terminate their connection.

This operation requires permissions for the `elasticfilesystem:DeleteAccessPoint` action.

## Request Syntax

```
DELETE /2015-02-01/access-points/AccessPointId HTTP/1.1
```

## URI Request Parameters

The request uses the following URI parameters.

**AccessPointId**

The ID of the access point that you want to delete.

Length Constraints: Maximum length of 128.

Pattern: `^(arn:aws[-a-z]*:elasticfilesystem:[0-9a-z-:]+:access-point/fsap-[0-9a-f]{8,40}|fsap-[0-9a-f]{8,40})$`

Required: Yes

## Request Body

The request does not have a request body.

## Response Syntax

```
HTTP/1.1 204
```

## Response Elements

If the action is successful, the service sends back an HTTP 204 response with an empty HTTP body.

# Errors

For information about the errors that are common to all actions, see Common Error Types.

**AccessPointNotFound**

Returned if the specified `AccessPointId` value doesn't exist in the requester's AWS account.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 404

**BadRequest**

Returned if the request is malformed or contains an error such as an invalid parameter value or a missing required parameter.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 400

**InternalServerError**

Returned if an error occurred on the server side.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 500

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

# DeleteFileSystem

Deletes a file system, permanently severing access to its contents. Upon return, the file system no longer exists and you can't access any contents of the deleted file system.

You need to manually delete mount targets attached to a file system before you can delete an EFS file system. This step is performed for you when you use the AWS console to delete a file system.

> **Note**
>
> You cannot delete a file system that is part of an EFS replication configuration. You need to delete the replication configuration first.

You can't delete a file system that is in use. That is, if the file system has any mount targets, you must first delete them. For more information, see DescribeMountTargets and DeleteMountTarget.

> **Note**
>
> The `DeleteFileSystem` call returns while the file system state is still `deleting`. You can check the file system deletion status by calling the DescribeFileSystems operation, which returns a list of file systems in your account. If you pass file system ID or creation token for the deleted file system, the DescribeFileSystems returns a `404 FileSystemNotFound` error.

This operation requires permissions for the `elasticfilesystem:DeleteFileSystem` action.

## Request Syntax

```
DELETE /2015-02-01/file-systems/FileSystemId HTTP/1.1
```

## URI Request Parameters

The request uses the following URI parameters.

### FileSystemId

The ID of the file system you want to delete.

Length Constraints: Maximum length of 128.

Pattern: `^(arn:aws[-a-z]*:elasticfilesystem:[0-9a-z-:]+:file-system/fs-[0-9a-f]{8,40}|fs-[0-9a-f]{8,40})$`

Required: Yes

## Request Body

The request does not have a request body.

## Response Syntax

```
HTTP/1.1 204
```

## Response Elements

If the action is successful, the service sends back an HTTP 204 response with an empty HTTP body.

## Errors

For information about the errors that are common to all actions, see Common Error Types.

**BadRequest**

Returned if the request is malformed or contains an error such as an invalid parameter value or a missing required parameter.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 400

**FileSystemInUse**

Returned if a file system has mount targets.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 409

**FileSystemNotFound**

Returned if the specified `FileSystemId` value doesn't exist in the requester's AWS account.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 404

**InternalServerError**

Returned if an error occurred on the server side.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 500

# Examples

## Delete a file system

The following example sends a DELETE request to the `file-systems` endpoint (`elasticfilesystem.us-west-2.amazonaws.com/2015-02-01/file-systems/fs-01234567`) to delete a file system whose ID is `fs-01234567`.

### Sample Request

```
DELETE /2015-02-01/file-systems/fs-01234567 HTTP/1.1
Host: elasticfilesystem.us-west-2.amazonaws.com
x-amz-date: 20140622T233021Z
Authorization: <...>
```

### Sample Response

```
HTTP/1.1 204 No Content
x-amzn-RequestId: a2d125b3-7ebd-4d6a-ab3d-5548630bff33
Content-Length: 0
```

# See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4
- AWS SDK for C++

- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

# DeleteFileSystemPolicy

Deletes the `FileSystemPolicy` for the specified file system. The default `FileSystemPolicy` goes into effect once the existing policy is deleted. For more information about the default file system policy, see Using Resource-based Policies with EFS.

This operation requires permissions for the `elasticfilesystem:DeleteFileSystemPolicy` action.

## Request Syntax

```
DELETE /2015-02-01/file-systems/FileSystemId/policy HTTP/1.1
```

## URI Request Parameters

The request uses the following URI parameters.

### FileSystemId

Specifies the EFS file system for which to delete the `FileSystemPolicy`.

Length Constraints: Maximum length of 128.

Pattern: `^(arn:aws[-a-z]*:elasticfilesystem:[0-9a-z-:]+:file-system/fs-[0-9a-f]{8,40}|fs-[0-9a-f]{8,40})$`

Required: Yes

## Request Body

The request does not have a request body.

## Response Syntax

```
HTTP/1.1 200
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

# Errors

For information about the errors that are common to all actions, see Common Error Types.

**BadRequest**

Returned if the request is malformed or contains an error such as an invalid parameter value or a missing required parameter.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 400

**FileSystemNotFound**

Returned if the specified `FileSystemId` value doesn't exist in the requester's AWS account.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 404

**IncorrectFileSystemLifeCycleState**

Returned if the file system's lifecycle state is not "available".

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 409

**InternalServerError**

Returned if an error occurred on the server side.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 500

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4
- AWS SDK for C++

- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

# DeleteMountTarget

Deletes the specified mount target.

This operation forcibly breaks any mounts of the file system by using the mount target that is being deleted, which might disrupt instances or applications using those mounts. To avoid applications getting cut off abruptly, you might consider unmounting any mounts of the mount target, if feasible. The operation also deletes the associated network interface. Uncommitted writes might be lost, but breaking a mount target using this operation does not corrupt the file system itself. The file system you created remains. You can mount an EC2 instance in your VPC by using another mount target.

This operation requires permissions for the following action on the file system:

- `elasticfilesystem:DeleteMountTarget`

> **Note**
>
> The `DeleteMountTarget` call returns while the mount target state is still `deleting`. You can check the mount target deletion by calling the DescribeMountTargets operation, which returns a list of mount target descriptions for the given file system.

The operation also requires permissions for the following Amazon EC2 action on the mount target's network interface:

- `ec2:DeleteNetworkInterface`

## Request Syntax

```
DELETE /2015-02-01/mount-targets/MountTargetId HTTP/1.1
```

## URI Request Parameters

The request uses the following URI parameters.

**MountTargetId**

The ID of the mount target to delete (String).

Length Constraints: Minimum length of 13. Maximum length of 45.

Pattern: `^fsmt-[0-9a-f]{8,40}$`

Required: Yes

## Request Body

The request does not have a request body.

## Response Syntax

```
HTTP/1.1 204
```

## Response Elements

If the action is successful, the service sends back an HTTP 204 response with an empty HTTP body.

## Errors

For information about the errors that are common to all actions, see Common Error Types.

**BadRequest**

Returned if the request is malformed or contains an error such as an invalid parameter value or a missing required parameter.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle.

Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 400

**DependencyTimeout**

The service timed out trying to fulfill the request, and the client should try the call again.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 504

**InternalServerError**

Returned if an error occurred on the server side.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 500

**MountTargetNotFound**

Returned if there is no mount target with the specified ID found in the caller's AWS account.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 404

## Examples

### Remove a file system's mount target

The following example sends a DELETE request to delete a specific mount target.

#### Sample Request

```
DELETE /2015-02-01/mount-targets/fsmt-9a13661e HTTP/1.1
Host: elasticfilesystem.us-west-2.amazonaws.com
x-amz-date: 20140622T232908Z
Authorization: <...>
```

#### Sample Response

```
HTTP/1.1 204 No Content
x-amzn-RequestId: 01234567-89ab-cdef-0123-456789abcdef
```

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2

- AWS SDK for .NET V4
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

# DeleteReplicationConfiguration

Deletes a replication configuration. Deleting a replication configuration ends the replication process. After a replication configuration is deleted, the destination file system becomes `Writeable` and its replication overwrite protection is re-enabled. For more information, see Delete a replication configuration.

This operation requires permissions for the `elasticfilesystem:DeleteReplicationConfiguration` action.

## Request Syntax

```
DELETE /2015-02-01/file-systems/SourceFileSystemId/replication-configuration?
deletionMode=DeletionMode HTTP/1.1
```

## URI Request Parameters

The request uses the following URI parameters.

### DeletionMode

When replicating across AWS accounts or across AWS Regions, Amazon EFS deletes the replication configuration from both the source and destination account or Region (`ALL_CONFIGURATIONS`) by default. If there's a configuration or permissions issue that prevents Amazon EFS from deleting the replication configuration from both sides, you can use the `LOCAL_CONFIGURATION_ONLY` mode to delete the replication configuration from only the local side (the account or Region from which the delete is performed).

> **Note**
>
> Only use the `LOCAL_CONFIGURATION_ONLY` mode in the case that Amazon EFS is unable to delete the replication configuration in both the source and destination account or Region. Deleting the local configuration leaves the configuration in the other account or Region unrecoverable.
> Additionally, do not use this mode for same-account, same-region replication as doing so results in a BadRequest exception error.

Valid Values: `ALL_CONFIGURATIONS | LOCAL_CONFIGURATION_ONLY`

**SourceFileSystemId**

The ID of the source file system in the replication configuration.

Length Constraints: Maximum length of 128.

Pattern: `^(arn:aws[-a-z]*:elasticfilesystem:[0-9a-z-:]+:file-system/fs-[0-9a-f]{8,40}|fs-[0-9a-f]{8,40})$`

Required: Yes

## Request Body

The request does not have a request body.

## Response Syntax

```
HTTP/1.1 204
```

## Response Elements

If the action is successful, the service sends back an HTTP 204 response with an empty HTTP body.

## Errors

For information about the errors that are common to all actions, see Common Error Types.

**BadRequest**

Returned if the request is malformed or contains an error such as an invalid parameter value or a missing required parameter.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle.

Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 400

**FileSystemNotFound**

Returned if the specified `FileSystemId` value doesn't exist in the requester's AWS account.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 404

**InternalServerError**

Returned if an error occurred on the server side.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 500

**ReplicationNotFound**

Returned if the specified file system does not have a replication configuration.

**ErrorCode**

ReplicationNotFound

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 404

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

# DeleteTags

> **Note**
>
> DEPRECATED - `DeleteTags` is deprecated and not maintained. To remove tags from EFS resources, use the UntagResource API action.

Deletes the specified tags from a file system. If the `DeleteTags` request includes a tag key that doesn't exist, Amazon EFS ignores it and doesn't cause an error. For more information about tags and related restrictions, see Tag restrictions in the *AWS Billing and Cost Management User Guide*.

This operation requires permissions for the `elasticfilesystem:DeleteTags` action.

## Request Syntax

```
POST /2015-02-01/delete-tags/FileSystemId HTTP/1.1
Content-type: application/json

{
   "TagKeys": [ "string" ]
}
```

## URI Request Parameters

The request uses the following URI parameters.

### FileSystemId

The ID of the file system whose tags you want to delete (String).

Length Constraints: Maximum length of 128.

Pattern: `^(arn:aws[-a-z]*:elasticfilesystem:[0-9a-z-:]+:file-system/fs-[0-9a-f]{8,40}|fs-[0-9a-f]{8,40})$`

Required: Yes

## Request Body

The request accepts the following data in JSON format.

**TagKeys**

A list of tag keys to delete.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 50 items.

Length Constraints: Minimum length of 1. Maximum length of 128.

Pattern: `^(?![aA]{1}[wW]{1}[sS]{1}:)([\p{L}\p{Z}\p{N}_.:/=+\-@]+)$`

Required: Yes

## Response Syntax

```
HTTP/1.1 204
```

## Response Elements

If the action is successful, the service sends back an HTTP 204 response with an empty HTTP body.

## Errors

For information about the errors that are common to all actions, see Common Error Types.

**BadRequest**

Returned if the request is malformed or contains an error such as an invalid parameter value or a missing required parameter.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 400

**FileSystemNotFound**

Returned if the specified `FileSystemId` value doesn't exist in the requester's AWS account.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 404

**InternalServerError**

Returned if an error occurred on the server side.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 500

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

# DescribeAccessPoints

Returns the description of a specific Amazon EFS access point if the `AccessPointId` is provided. If you provide an EFS `FileSystemId`, it returns descriptions of all access points for that file system. You can provide either an `AccessPointId` or a `FileSystemId` in the request, but not both.

This operation requires permissions for the `elasticfilesystem:DescribeAccessPoints` action.

## Request Syntax

```
GET /2015-02-01/access-points?
AccessPointId=AccessPointId&FileSystemId=FileSystemId&MaxResults=MaxResults&NextToken=NextToken
 HTTP/1.1
```

## URI Request Parameters

The request uses the following URI parameters.

**AccessPointId**

(Optional) Specifies an EFS access point to describe in the response; mutually exclusive with `FileSystemId`.

Length Constraints: Maximum length of 128.

Pattern: `^(arn:aws[-a-z]*:elasticfilesystem:[0-9a-z-:]+:access-point/fsap-[0-9a-f]{8,40}|fsap-[0-9a-f]{8,40})$`

**FileSystemId**

(Optional) If you provide a `FileSystemId`, EFS returns all access points for that file system; mutually exclusive with `AccessPointId`.

Length Constraints: Maximum length of 128.

Pattern: `^(arn:aws[-a-z]*:elasticfilesystem:[0-9a-z-:]+:file-system/fs-[0-9a-f]{8,40}|fs-[0-9a-f]{8,40})$`

**MaxResults**

(Optional) When retrieving all access points for a file system, you can optionally specify the `MaxItems` parameter to limit the number of objects returned in a response. The default value is 100.

Valid Range: Minimum value of 1.

**NextToken**

`NextToken` is present if the response is paginated. You can use `NextMarker` in the subsequent request to fetch the next page of access point descriptions.

Length Constraints: Minimum length of 1. Maximum length of 128.

Pattern: `.+`

## Request Body

The request does not have a request body.

## Response Syntax

```
HTTP/1.1 200
Content-type: application/json

{
   "AccessPoints": [
      {
         "AccessPointArn": "string",
         "AccessPointId": "string",
         "ClientToken": "string",
         "FileSystemId": "string",
         "LifeCycleState": "string",
         "Name": "string",
         "OwnerId": "string",
         "PosixUser": {
            "Gid": number,
            "SecondaryGids": [ number ],
            "Uid": number
         },
         "RootDirectory": {
            "CreationInfo": {
```

```
                "OwnerGid": number,
                "OwnerUid": number,
                "Permissions": "string"
             },
             "Path": "string"
          },
          "Tags": [
             {
                "Key": "string",
                "Value": "string"
             }
          ]
       }
    ],
    "NextToken": "string"
}
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**AccessPoints**

An array of access point descriptions.

Type: Array of AccessPointDescription objects

**NextToken**

Present if there are more access points than returned in the response. You can use the NextMarker in the subsequent request to fetch the additional descriptions.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 128.

Pattern: `.+`

## Errors

For information about the errors that are common to all actions, see Common Error Types.

**AccessPointNotFound**

Returned if the specified `AccessPointId` value doesn't exist in the requester's AWS account.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 404

**BadRequest**

Returned if the request is malformed or contains an error such as an invalid parameter value or a missing required parameter.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 400

**FileSystemNotFound**

Returned if the specified `FileSystemId` value doesn't exist in the requester's AWS account.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 404

**InternalServerError**

Returned if an error occurred on the server side.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 500

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3

- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

# DescribeAccountPreferences

Returns the account preferences settings for the AWS account associated with the user making the request, in the current AWS Region.

## Request Syntax

```
GET /2015-02-01/account-preferences HTTP/1.1
Content-type: application/json

{
   "MaxResults": number,
   "NextToken": "string"
}
```

## URI Request Parameters

The request does not use any URI parameters.

## Request Body

The request accepts the following data in JSON format.

**MaxResults**

(Optional) When retrieving account preferences, you can optionally specify the `MaxItems` parameter to limit the number of objects returned in a response. The default value is 100.

Type: Integer

Valid Range: Minimum value of 1.

Required: No

**NextToken**

(Optional) You can use `NextToken` in a subsequent request to fetch the next page of AWS account preferences if the response payload was paginated.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 128.

Pattern: `.+`

Required: No

## Response Syntax

```
HTTP/1.1 200
Content-type: application/json

{
   "NextToken": "string",
   "ResourceIdPreference": {
      "ResourceIdType": "string",
      "Resources": [ "string" ]
   }
}
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

### NextToken

Present if there are more records than returned in the response. You can use the `NextToken` in the subsequent request to fetch the additional descriptions.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 128.

Pattern: `.+`

### ResourceIdPreference

Describes the resource ID preference setting for the AWS account associated with the user making the request, in the current AWS Region.

Type: ResourceIdPreference object

## Errors

For information about the errors that are common to all actions, see Common Error Types.

**InternalServerError**

Returned if an error occurred on the server side.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 500

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

# DescribeBackupPolicy

Returns the backup policy for the specified EFS file system.

## Request Syntax

```
GET /2015-02-01/file-systems/FileSystemId/backup-policy HTTP/1.1
```

## URI Request Parameters

The request uses the following URI parameters.

### FileSystemId

Specifies which EFS file system for which to retrieve the `BackupPolicy`.

Length Constraints: Maximum length of 128.

Pattern: `^(arn:aws[-a-z]*:elasticfilesystem:[0-9a-z-:]+:file-system/fs-[0-9a-f]{8,40}|fs-[0-9a-f]{8,40})$`

Required: Yes

## Request Body

The request does not have a request body.

## Response Syntax

```
HTTP/1.1 200
Content-type: application/json

{
   "BackupPolicy": {
      "Status": "string"
   }
}
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**BackupPolicy**

Describes the file system's backup policy, indicating whether automatic backups are turned on or off.

Type: BackupPolicy object

## Errors

For information about the errors that are common to all actions, see Common Error Types.

**BadRequest**

Returned if the request is malformed or contains an error such as an invalid parameter value or a missing required parameter.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 400

**FileSystemNotFound**

Returned if the specified `FileSystemId` value doesn't exist in the requester's AWS account.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 404

**InternalServerError**

Returned if an error occurred on the server side.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 500

**PolicyNotFound**

Returned if the specified IAM policy or resource policy cannot be located or accessed. Depending on the action being performed, the error may also be returned if the default file system policy is in effect for the EFS file system or if no backup policy is specified.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle.

Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 404

**ValidationException**

Returned if the AWS Backup service is not available in the AWS Region in which the request was made.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 400

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python

- AWS SDK for Ruby V3

# DescribeFileSystemPolicy

Returns the `FileSystemPolicy` for the specified EFS file system.

This operation requires permissions for the `elasticfilesystem:DescribeFileSystemPolicy` action.

## Request Syntax

```
GET /2015-02-01/file-systems/FileSystemId/policy HTTP/1.1
```

## URI Request Parameters

The request uses the following URI parameters.

### FileSystemId

Specifies which EFS file system to retrieve the `FileSystemPolicy` for.

Length Constraints: Maximum length of 128.

Pattern: `^(arn:aws[-a-z]*:elasticfilesystem:[0-9a-z-:]+:file-system/fs-[0-9a-f]{8,40}|fs-[0-9a-f]{8,40})$`

Required: Yes

## Request Body

The request does not have a request body.

## Response Syntax

```
HTTP/1.1 200
Content-type: application/json

{
   "FileSystemId": "string",
   "Policy": "string"
```

```
}
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**FileSystemId**

Specifies the EFS file system to which the `FileSystemPolicy` applies.

Type: String

Length Constraints: Maximum length of 128.

Pattern: `^(arn:aws[-a-z]*:elasticfilesystem:[0-9a-z-:]+:file-system/fs-[0-9a-f]{8,40}|fs-[0-9a-f]{8,40})$`

**Policy**

The JSON formatted `FileSystemPolicy` for the EFS file system.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 20000.

Pattern: `[\s\S]+`

## Errors

For information about the errors that are common to all actions, see Common Error Types.

**BadRequest**

Returned if the request is malformed or contains an error such as an invalid parameter value or a missing required parameter.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 400

**FileSystemNotFound**

Returned if the specified `FileSystemId` value doesn't exist in the requester's AWS account.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 404

**InternalServerError**

Returned if an error occurred on the server side.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 500

**PolicyNotFound**

Returned if the specified IAM policy or resource policy cannot be located or accessed. Depending on the action being performed, the error may also be returned if the default file system policy is in effect for the EFS file system or if no backup policy is specified.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 404

# Examples

## Example

This example illustrates one usage of DescribeFileSystemPolicy.

### Sample Request

```
GET /2015-02-01/file-systems/fs-01234567/policy HTTP/1.1
```

### Sample Response

```
{
    "FileSystemId": "fs-01234567",
    "Policy": "{
        "Version": "2012-10-17",
        "Id": "efs-policy-wizard-cdef0123-aaaa-6666-5555-444455556666",
        "Statement": [
            {
```

```
                "Sid": "efs-statement-abcdef01-1111-bbbb-2222-111122224444",
                "Effect" : "Deny",
                "Principal": {
                "AWS": "*"
                },
                "Action": "*",
                "Resource": "arn:aws:elasticfilesystem:us-east-2:111122223333:file-
system/fs-01234567",
                "Condition": {
                "Bool": {
                    "aws:SecureTransport": "false"
                    }
                }
            },
            {
                "Sid": "efs-statement-01234567-aaaa-3333-4444-111122223333",
                "Effect": "Allow",
                "Principal": {
                    "AWS": "*"
                },
                "Action": [
                    "elasticfilesystem:ClientMount",
                    "elasticfilesystem:ClientWrite"
                ],
                "Resource" : "arn:aws:elasticfilesystem:us-east-2:111122223333:file-
system/fs-01234567"
            }
        ]
    }
}
```

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2

- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

# DescribeFileSystems

Returns the description of a specific Amazon EFS file system if either the file system `CreationToken` or the `FileSystemId` is provided. Otherwise, it returns descriptions of all file systems owned by the caller's AWS account in the AWS Region of the endpoint that you're calling.

When retrieving all file system descriptions, you can optionally specify the `MaxItems` parameter to limit the number of descriptions in a response. This number is automatically set to 100. If more file system descriptions remain, Amazon EFS returns a `NextMarker`, an opaque token, in the response. In this case, you should send a subsequent request with the `Marker` request parameter set to the value of `NextMarker`.

To retrieve a list of your file system descriptions, this operation is used in an iterative process, where `DescribeFileSystems` is called first without the `Marker` and then the operation continues to call it with the `Marker` parameter set to the value of the `NextMarker` from the previous response until the response has no `NextMarker`.

The order of file systems returned in the response of one `DescribeFileSystems` call and the order of file systems returned across the responses of a multi-call iteration is unspecified.

This operation requires permissions for the `elasticfilesystem:DescribeFileSystems` action.

## Request Syntax

```
GET /2015-02-01/file-systems?
CreationToken=CreationToken&FileSystemId=FileSystemId&Marker=Marker&MaxItems=MaxItems
 HTTP/1.1
```

## URI Request Parameters

The request uses the following URI parameters.

### CreationToken

(Optional) Restricts the list to the file system with this creation token (String). You specify a creation token when you create an Amazon EFS file system.

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: `.+`

**FileSystemId**

(Optional) ID of the file system whose description you want to retrieve (String).

Length Constraints: Maximum length of 128.

Pattern: `^(arn:aws[-a-z]*:elasticfilesystem:[0-9a-z-:]+:file-system/fs-[0-9a-f]{8,40}|fs-[0-9a-f]{8,40})$`

**Marker**

(Optional) Opaque pagination token returned from a previous `DescribeFileSystems` operation (String). If present, specifies to continue the list from where the returning call had left off.

Length Constraints: Minimum length of 1. Maximum length of 128.

Pattern: `.+`

**MaxItems**

(Optional) Specifies the maximum number of file systems to return in the response (integer). This number is automatically set to 100. The response is paginated at 100 per page if you have more than 100 file systems.

Valid Range: Minimum value of 1.

## Request Body

The request does not have a request body.

## Response Syntax

```
HTTP/1.1 200
Content-type: application/json

{
   "FileSystems": [
      {
         "AvailabilityZoneId": "string",
         "AvailabilityZoneName": "string",
         "CreationTime": number,
         "CreationToken": "string",
```

```
         "Encrypted": boolean,
         "FileSystemArn": "string",
         "FileSystemId": "string",
         "FileSystemProtection": {
            "ReplicationOverwriteProtection": "string"
         },
         "KmsKeyId": "string",
         "LifeCycleState": "string",
         "Name": "string",
         "NumberOfMountTargets": number,
         "OwnerId": "string",
         "PerformanceMode": "string",
         "ProvisionedThroughputInMibps": number,
         "SizeInBytes": {
            "Timestamp": number,
            "Value": number,
            "ValueInArchive": number,
            "ValueInIA": number,
            "ValueInStandard": number
         },
         "Tags": [
            {
               "Key": "string",
               "Value": "string"
            }
         ],
         "ThroughputMode": "string"
      }
   ],
   "Marker": "string",
   "NextMarker": "string"
}
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

### FileSystems

An array of file system descriptions.

Type: Array of FileSystemDescription objects

**Marker**

Present if provided by caller in the request (String).

Type: String

Length Constraints: Minimum length of 1. Maximum length of 128.

Pattern: `.+`

**NextMarker**

Present if there are more file systems than returned in the response (String). You can use the `NextMarker` in the subsequent request to fetch the descriptions.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 128.

Pattern: `.+`

## Errors

For information about the errors that are common to all actions, see Common Error Types.

**BadRequest**

Returned if the request is malformed or contains an error such as an invalid parameter value or a missing required parameter.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 400

**FileSystemNotFound**

Returned if the specified `FileSystemId` value doesn't exist in the requester's AWS account.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 404

**InternalServerError**

Returned if an error occurred on the server side.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 500

# Examples

## Retrieve a list of 10 file systems

The following example sends a GET request to the `file-systems` endpoint (`elasticfilesystem.us-west-2.amazonaws.com/2015-02-01/file-systems`). The

request specifies a `MaxItems` query parameter to limit the number of file system descriptions to 10.

### Sample Request

```
GET /2015-02-01/file-systems?MaxItems=10 HTTP/1.1
Host: elasticfilesystem.us-west-2.amazonaws.com
x-amz-date: 20140622T191208Z
Authorization: <...>
```

### Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: 01234567-89ab-cdef-0123-456789abcdef
Content-Type: application/json
Content-Length: 499
{
   "FileSystems":[
      {
         "OwnerId":"251839141158",
         "CreationToken":"MyFileSystem1",
         "FileSystemId":"fs-01234567",
         "PerformanceMode" : "generalPurpose",
         "CreationTime":"1403301078",
         "LifeCycleState":"created",
         "Name":"my first file system",
         "NumberOfMountTargets":1,
         "SizeInBytes":{
            "Timestamp": 1403301078,
            "Value": 29313618372,
            "ValueInArchive": 201156,
            "ValueInIA": 675432,
            "ValueInStandard": 29312741784
         }
      }
   ]
}
```

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

# DescribeLifecycleConfiguration

Returns the current `LifecycleConfiguration` object for the specified EFS file system. Lifecycle management uses the `LifecycleConfiguration` object to identify when to move files between storage classes. For a file system without a `LifecycleConfiguration` object, the call returns an empty array in the response.

This operation requires permissions for the `elasticfilesystem:DescribeLifecycleConfiguration` operation.

## Request Syntax

```
GET /2015-02-01/file-systems/FileSystemId/lifecycle-configuration HTTP/1.1
```

## URI Request Parameters

The request uses the following URI parameters.

### FileSystemId

The ID of the file system whose `LifecycleConfiguration` object you want to retrieve (String).

Length Constraints: Maximum length of 128.

Pattern: `^(arn:aws[-a-z]*:elasticfilesystem:[0-9a-z-:]+:file-system/fs-[0-9a-f]{8,40}|fs-[0-9a-f]{8,40})$`

Required: Yes

## Request Body

The request does not have a request body.

## Response Syntax

```
HTTP/1.1 200
Content-type: application/json
```

```
{
   "LifecyclePolicies": [
      {
         "TransitionToArchive": "string",
         "TransitionToIA": "string",
         "TransitionToPrimaryStorageClass": "string"
      }
   ]
}
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**LifecyclePolicies**

An array of lifecycle management policies. EFS supports a maximum of one policy per file system.

Type: Array of LifecyclePolicy objects

Array Members: Maximum number of 3 items.

## Errors

For information about the errors that are common to all actions, see Common Error Types.

**BadRequest**

Returned if the request is malformed or contains an error such as an invalid parameter value or a missing required parameter.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end

user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 400

**FileSystemNotFound**

Returned if the specified `FileSystemId` value doesn't exist in the requester's AWS account.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 404

**InternalServerError**

Returned if an error occurred on the server side.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 500

# Examples

## Retrieve the lifecycle configuration for a file system

The following request retrieves the `LifecycleConfiguration` object for the specified file system.

### Sample Request

```
GET /2015-02-01/file-systems/fs-01234567/lifecycle-configuration HTTP/1.1
Host: elasticfilesystem.us-west-2.amazonaws.com
x-amz-date: 20181120T221118Z
Authorization: <...>
```

### Sample Response

```
HTTP/1.1 200 OK
        x-amzn-RequestId: 01234567-89ab-cdef-0123-456789abcdef
        Content-Type: application/json
        Content-Length: 86
{
  "LifecyclePolicies": [
    {
        "TransitionToArchive": "AFTER_270_DAYS"
    },
    {
        "TransitionToIA": "AFTER_14_DAYS"
    },
    {
        "TransitionToPrimaryStorageClass": "AFTER_1_ACCESS"
    }
  ]
}
```

# See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2

- AWS SDK for .NET V4
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

# DescribeMountTargets

Returns the descriptions of all the current mount targets, or a specific mount target, for a file system. When requesting all of the current mount targets, the order of mount targets returned in the response is unspecified.

This operation requires permissions for the `elasticfilesystem:DescribeMountTargets` action, on either the file system ID that you specify in `FileSystemId`, or on the file system of the mount target that you specify in `MountTargetId`.

The corresponding CLI command is `describe-mount-targets`.

## Request Syntax

```
GET /2015-02-01/mount-targets?
AccessPointId=AccessPointId&FileSystemId=FileSystemId&Marker=Marker&MaxItems=MaxItems&MountTarg
 HTTP/1.1
```

## URI Request Parameters

The request uses the following URI parameters.

### AccessPointId

(Optional) The ID of the access point whose mount targets that you want to list. It must be included in your request if a `FileSystemId` or `MountTargetId` is not included in your request. Accepts either an access point ID or ARN as input.

Length Constraints: Maximum length of 128.

Pattern: `^(arn:aws[-a-z]*:elasticfilesystem:[0-9a-z-:]+:access-point/fsap-[0-9a-f]{8,40}|fsap-[0-9a-f]{8,40})$`

### FileSystemId

(Optional) ID of the file system whose mount targets you want to list (String). It must be included in your request if an `AccessPointId` or `MountTargetId` is not included. Accepts either a file system ID or ARN as input.

Length Constraints: Maximum length of 128.

Pattern: `^(arn:aws[-a-z]*:elasticfilesystem:[0-9a-z-:]+:file-system/fs-[0-9a-f]{8,40}|fs-[0-9a-f]{8,40})$`

**Marker**

(Optional) Opaque pagination token returned from a previous `DescribeMountTargets` operation (String). If present, it specifies to continue the list from where the previous returning call left off.

Length Constraints: Minimum length of 1. Maximum length of 128.

Pattern: `.+`

**MaxItems**

(Optional) Maximum number of mount targets to return in the response. Currently, this number is automatically set to 10, and other values are ignored. The response is paginated at 100 per page if you have more than 100 mount targets.

Valid Range: Minimum value of 1.

**MountTargetId**

(Optional) ID of the mount target that you want to have described (String). It must be included in your request if `FileSystemId` is not included. Accepts either a mount target ID or ARN as input.

Length Constraints: Minimum length of 13. Maximum length of 45.

Pattern: `^fsmt-[0-9a-f]{8,40}$`

## Request Body

The request does not have a request body.

## Response Syntax

```
HTTP/1.1 200
Content-type: application/json

{
   "Marker": "string",
```

```
   "MountTargets": [
      {
         "AvailabilityZoneId": "string",
         "AvailabilityZoneName": "string",
         "FileSystemId": "string",
         "IpAddress": "string",
         "Ipv6Address": "string",
         "LifeCycleState": "string",
         "MountTargetId": "string",
         "NetworkInterfaceId": "string",
         "OwnerId": "string",
         "SubnetId": "string",
         "VpcId": "string"
      }
   ],
   "NextMarker": "string"
}
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

### Marker

If the request included the `Marker`, the response returns that value in this field.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 128.

Pattern: `.+`

### MountTargets

Returns the file system's mount targets as an array of `MountTargetDescription` objects.

Type: Array of MountTargetDescription objects

### NextMarker

If a value is present, there are more mount targets to return. In a subsequent request, you can provide `Marker` in your request with this value to retrieve the next set of mount targets.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 128.

Pattern: `.+`

## Errors

For information about the errors that are common to all actions, see Common Error Types.

**AccessPointNotFound**

Returned if the specified `AccessPointId` value doesn't exist in the requester's AWS account.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 404

**BadRequest**

Returned if the request is malformed or contains an error such as an invalid parameter value or a missing required parameter.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle.

Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 400

**FileSystemNotFound**

Returned if the specified `FileSystemId` value doesn't exist in the requester's AWS account.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 404

**InternalServerError**

Returned if an error occurred on the server side.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 500

**MountTargetNotFound**

Returned if there is no mount target with the specified ID found in the caller's AWS account.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 404

# Examples

## Retrieve descriptions of mount targets created for a file system

The following request retrieves descriptions of mount targets created for the specified file system.

### Sample Request

```
GET /2015-02-01/mount-targets?FileSystemId=fs-01234567 HTTP/1.1
Host: elasticfilesystem.us-west-2.amazonaws.com
x-amz-date: 20140622T191252Z
Authorization: <...>
```

### Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: 01234567-89ab-cdef-0123-456789abcdef
Content-Type: application/json
Content-Length: 357

{
   "MountTargets":[
      {
         "OwnerId":"251839141158",
         "MountTargetId":"fsmt-01234567",
         "FileSystemId":"fs-01234567",
```

```
            "SubnetId":"subnet-01234567",
            "LifeCycleState":"added",
            "IpAddress":"10.0.2.42",
            "NetworkInterfaceId":"eni-1bcb7772"
         }
      ]
}
```

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

# DescribeMountTargetSecurityGroups

Returns the security groups currently in effect for a mount target. This operation requires that the network interface of the mount target has been created and the lifecycle state of the mount target is not `deleted`.

This operation requires permissions for the following actions:

- `elasticfilesystem:DescribeMountTargetSecurityGroups` action on the mount target's file system.
- `ec2:DescribeNetworkInterfaceAttribute` action on the mount target's network interface.

## Request Syntax

```
GET /2015-02-01/mount-targets/MountTargetId/security-groups HTTP/1.1
```

## URI Request Parameters

The request uses the following URI parameters.

**MountTargetId**

The ID of the mount target whose security groups you want to retrieve.

Length Constraints: Minimum length of 13. Maximum length of 45.

Pattern: `^fsmt-[0-9a-f]{8,40}$`

Required: Yes

## Request Body

The request does not have a request body.

## Response Syntax

```
HTTP/1.1 200
Content-type: application/json
```

```
{
   "SecurityGroups": [ "string" ]
}
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**SecurityGroups**

An array of security groups.

Type: Array of strings

Array Members: Maximum number of 100 items.

Length Constraints: Minimum length of 11. Maximum length of 43.

Pattern: `^sg-[0-9a-f]{8,40}`

## Errors

For information about the errors that are common to all actions, see Common Error Types.

**BadRequest**

Returned if the request is malformed or contains an error such as an invalid parameter value or a missing required parameter.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle.

Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 400

**IncorrectMountTargetState**

Returned if the mount target is not in the correct state for the operation.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 409

**InternalServerError**

Returned if an error occurred on the server side.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 500

**MountTargetNotFound**

Returned if there is no mount target with the specified ID found in the caller's AWS account.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 404

# Examples

## Retrieve security groups in effect for a file system

The following example retrieves the security groups that are in effect for the network interface associated with a mount target.

### Sample Request

```
GET /2015-02-01/mount-targets/fsmt-9a13661e/security-groups HTTP/1.1
Host: elasticfilesystem.us-west-2.amazonaws.com
x-amz-date: 20140620T223513Z
Authorization: <...>
```

### Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: 01234567-89ab-cdef-0123-456789abcdef
Content-Length: 57

{
"SecurityGroups" : [
"sg-188d9f74"
]
}
```

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

# DescribeReplicationConfigurations

Retrieves the replication configuration for a specific file system. If a file system is not specified, all of the replication configurations for the AWS account in an AWS Region are retrieved.

## Request Syntax

```
GET /2015-02-01/file-systems/replication-configurations?
FileSystemId=FileSystemId&MaxResults=MaxResults&NextToken=NextToken HTTP/1.1
```

## URI Request Parameters

The request uses the following URI parameters.

### FileSystemId

You can retrieve the replication configuration for a specific file system by providing its file system ID. For cross-account,cross-region replication, an account can only describe the replication configuration for a file system in its own Region.

Length Constraints: Maximum length of 128.

Pattern: `^(arn:aws[-a-z]*:elasticfilesystem:[0-9a-z-:]+:file-system/fs-[0-9a-f]{8,40}|fs-[0-9a-f]{8,40})$`

### MaxResults

(Optional) To limit the number of objects returned in a response, you can specify the `MaxItems` parameter. The default value is 100.

Valid Range: Minimum value of 1.

### NextToken

`NextToken` is present if the response is paginated. You can use `NextToken` in a subsequent request to fetch the next page of output.

Length Constraints: Minimum length of 1. Maximum length of 128.

Pattern: `.+`

## Request Body

The request does not have a request body.

## Response Syntax

```
HTTP/1.1 200
Content-type: application/json

{
   "NextToken": "string",
   "Replications": [
      {
         "CreationTime": number,
         "Destinations": [
            {
               "FileSystemId": "string",
               "LastReplicatedTimestamp": number,
               "OwnerId": "string",
               "Region": "string",
               "RoleArn": "string",
               "Status": "string",
               "StatusMessage": "string"
            }
         ],
         "OriginalSourceFileSystemArn": "string",
         "SourceFileSystemArn": "string",
         "SourceFileSystemId": "string",
         "SourceFileSystemOwnerId": "string",
         "SourceFileSystemRegion": "string"
      }
   ]
}
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**NextToken**

You can use the `NextToken` from the previous response in a subsequent request to fetch the additional descriptions.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 128.

Pattern: `.+`

**Replications**

The collection of replication configurations that is returned.

Type: Array of ReplicationConfigurationDescription objects

## Errors

For information about the errors that are common to all actions, see Common Error Types.

**BadRequest**

Returned if the request is malformed or contains an error such as an invalid parameter value or a missing required parameter.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 400

**FileSystemNotFound**

Returned if the specified `FileSystemId` value doesn't exist in the requester's AWS account.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 404

**InternalServerError**

Returned if an error occurred on the server side.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 500

**ReplicationNotFound**

Returned if the specified file system does not have a replication configuration.

**ErrorCode**

ReplicationNotFound

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end

user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 404

**ValidationException**

Returned if the AWS Backup service is not available in the AWS Region in which the request was made.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 400

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3

- AWS SDK for Python
- AWS SDK for Ruby V3

# DescribeTags

> **Note**
>
> DEPRECATED - The `DescribeTags` action is deprecated and not maintained. To view tags associated with EFS resources, use the `ListTagsForResource` API action.

Returns the tags associated with a file system. The order of tags returned in the response of one `DescribeTags` call and the order of tags returned across the responses of a multiple-call iteration (when using pagination) is unspecified.

This operation requires permissions for the `elasticfilesystem:DescribeTags` action.

## Request Syntax

```
GET /2015-02-01/tags/FileSystemId/?Marker=Marker&MaxItems=MaxItems HTTP/1.1
```

## URI Request Parameters

The request uses the following URI parameters.

### FileSystemId

The ID of the file system whose tag set you want to retrieve.

Length Constraints: Maximum length of 128.

Pattern: `^(arn:aws[-a-z]*:elasticfilesystem:[0-9a-z-:]+:file-system/fs-[0-9a-f]{8,40}|fs-[0-9a-f]{8,40})$`

Required: Yes

### Marker

(Optional) An opaque pagination token returned from a previous `DescribeTags` operation (String). If present, it specifies to continue the list from where the previous call left off.

Length Constraints: Minimum length of 1. Maximum length of 128.

Pattern: `.+`

### MaxItems

(Optional) The maximum number of file system tags to return in the response. Currently, this number is automatically set to 100, and other values are ignored. The response is paginated at 100 per page if you have more than 100 tags.

Valid Range: Minimum value of 1.

## Request Body

The request does not have a request body.

## Response Syntax

```
HTTP/1.1 200
Content-type: application/json

{
   "Marker": "string",
   "NextMarker": "string",
   "Tags": [
      {
         "Key": "string",
         "Value": "string"
      }
   ]
}
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

### Marker

If the request included a `Marker`, the response returns that value in this field.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 128.

Pattern: `.+`

**NextMarker**

If a value is present, there are more tags to return. In a subsequent request, you can provide the value of `NextMarker` as the value of the `Marker` parameter in your next request to retrieve the next set of tags.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 128.

Pattern: `.+`

**Tags**

Returns tags associated with the file system as an array of `Tag` objects.

Type: Array of Tag objects

## Errors

For information about the errors that are common to all actions, see Common Error Types.

**BadRequest**

Returned if the request is malformed or contains an error such as an invalid parameter value or a missing required parameter.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 400

**FileSystemNotFound**

Returned if the specified `FileSystemId` value doesn't exist in the requester's AWS account.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 404

**InternalServerError**

Returned if an error occurred on the server side.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 500

# Examples

## Retrieve tags associated with a file system

The following request retrieves tags (key-value pairs) associated with the specified file system.

### Sample Request

```
GET /2015-02-01/tags/fs-01234567/ HTTP/1.1
Host: elasticfilesystem.us-west-2.amazonaws.com
x-amz-date: 20140620T215404Z
Authorization: <...>
```

### Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: 01234567-89ab-cdef-0123-456789abcdef
Content-Type: application/json
Content-Length: 288

{
    "Tags":[
        {
            "Key":"Name",
            "Value":"my first file system"
        },
        {
            "Key":"Fleet",
            "Value":"Development"
        },
        {
            "Key":"Developer",
            "Value":"Alice"
        }
    ]
}
```

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4
- AWS SDK for C++
- AWS SDK for Go v2

- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

# ListTagsForResource

Lists all tags for a top-level EFS resource. You must provide the ID of the resource that you want to retrieve the tags for.

This operation requires permissions for the `elasticfilesystem:DescribeAccessPoints` action.

## Request Syntax

```
GET /2015-02-01/resource-tags/ResourceId?MaxResults=MaxResults&NextToken=NextToken
 HTTP/1.1
```

## URI Request Parameters

The request uses the following URI parameters.

**MaxResults**

(Optional) Specifies the maximum number of tag objects to return in the response. The default value is 100.

Valid Range: Minimum value of 1.

**NextToken**

(Optional) You can use `NextToken` in a subsequent request to fetch the next page of access point descriptions if the response payload was paginated.

Length Constraints: Minimum length of 1. Maximum length of 128.

Pattern: `.+`

**ResourceId**

Specifies the EFS resource you want to retrieve tags for. You can retrieve tags for EFS file systems and access points using this API endpoint.

Length Constraints: Maximum length of 128.

Pattern: `^(arn:aws[-a-z]*:elasticfilesystem:[0-9a-z-:]+:(access-point/fsap|file-system/fs)-[0-9a-f]{8,40}|fs(ap)?-[0-9a-f]{8,40})$`

Required: Yes

## Request Body

The request does not have a request body.

## Response Syntax

```
HTTP/1.1 200
Content-type: application/json

{
   "NextToken": "string",
   "Tags": [
      {
         "Key": "string",
         "Value": "string"
      }
   ]
}
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**NextToken**

NextToken is present if the response payload is paginated. You can use NextToken in a subsequent request to fetch the next page of access point descriptions.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 128.

Pattern: .+

**Tags**

An array of the tags for the specified EFS resource.

Type: Array of Tag objects

## Errors

For information about the errors that are common to all actions, see Common Error Types.

**AccessPointNotFound**

Returned if the specified `AccessPointId` value doesn't exist in the requester's AWS account.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 404

**BadRequest**

Returned if the request is malformed or contains an error such as an invalid parameter value or a missing required parameter.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 400

**FileSystemNotFound**

Returned if the specified `FileSystemId` value doesn't exist in the requester's AWS account.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 404

**InternalServerError**

Returned if an error occurred on the server side.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 500

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2

- AWS SDK for .NET V4
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

# ModifyMountTargetSecurityGroups

Modifies the set of security groups in effect for a mount target.

When you create a mount target, Amazon EFS also creates a new network interface. For more information, see CreateMountTarget. This operation replaces the security groups in effect for the network interface associated with a mount target, with the `SecurityGroups` provided in the request. This operation requires that the network interface of the mount target has been created and the lifecycle state of the mount target is not `deleted`.

The operation requires permissions for the following actions:

- `elasticfilesystem:ModifyMountTargetSecurityGroups` action on the mount target's file system.
- `ec2:ModifyNetworkInterfaceAttribute` action on the mount target's network interface.

## Request Syntax

```
PUT /2015-02-01/mount-targets/MountTargetId/security-groups HTTP/1.1
Content-type: application/json

{
   "SecurityGroups": [ "string" ]
}
```

## URI Request Parameters

The request uses the following URI parameters.

**MountTargetId**

The ID of the mount target whose security groups you want to modify.

Length Constraints: Minimum length of 13. Maximum length of 45.

Pattern: `^fsmt-[0-9a-f]{8,40}$`

Required: Yes

## Request Body

The request accepts the following data in JSON format.

**SecurityGroups**

An array of VPC security group IDs.

Type: Array of strings

Array Members: Maximum number of 100 items.

Length Constraints: Minimum length of 11. Maximum length of 43.

Pattern: `^sg-[0-9a-f]{8,40}`

Required: No

## Response Syntax

```
HTTP/1.1 204
```

## Response Elements

If the action is successful, the service sends back an HTTP 204 response with an empty HTTP body.

## Errors

For information about the errors that are common to all actions, see Common Error Types.

**BadRequest**

Returned if the request is malformed or contains an error such as an invalid parameter value or a missing required parameter.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 400

**IncorrectMountTargetState**

Returned if the mount target is not in the correct state for the operation.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 409

**InternalServerError**

Returned if an error occurred on the server side.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle.

Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 500

**MountTargetNotFound**

Returned if there is no mount target with the specified ID found in the caller's AWS account.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 404

**SecurityGroupLimitExceeded**

Returned if the number of `SecurityGroups` specified in the request is greater than the limit, which is based on account quota. Either delete some security groups or request that the account quota be raised. For more information, see Amazon VPC Quotas in the *Amazon VPC User Guide* (see the **Security Groups** table).

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 400

**SecurityGroupNotFound**

Returned if one of the specified security groups doesn't exist in the subnet's virtual private cloud (VPC).

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 400

# Examples

## Replace a mount target's security groups

The following example replaces security groups in effect for the network interface associated with a mount target.

### Sample Request

```
PUT /2015-02-01/mount-targets/fsmt-9a13661e/security-groups HTTP/1.1
Host: elasticfilesystem.us-west-2.amazonaws.com
x-amz-date: 20140620T223446Z
Authorization: <...>
Content-Type: application/json
Content-Length: 57

{
  "SecurityGroups" : [
  "sg-188d9f74"
  ]
}
```

### Sample Response

```
HTTP/1.1 204 No Content
x-amzn-RequestId: 01234567-89ab-cdef-0123-456789abcdef
```

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

# PutAccountPreferences

Use this operation to set the account preference in the current AWS Region to use long 17 character (63 bit) or short 8 character (32 bit) resource IDs for new EFS file system and mount target resources. All existing resource IDs are not affected by any changes you make. You can set the ID preference during the opt-in period as EFS transitions to long resource IDs. For more information, see Managing Amazon EFS resource IDs.

> **Note**
>
> Starting in October, 2021, you will receive an error if you try to set the account preference to use the short 8 character format resource ID. Contact AWS support if you receive an error and must use short IDs for file system and mount target resources.

## Request Syntax

```
PUT /2015-02-01/account-preferences HTTP/1.1
Content-type: application/json

{
   "ResourceIdType": "string"
}
```

## URI Request Parameters

The request does not use any URI parameters.

## Request Body

The request accepts the following data in JSON format.

**ResourceIdType**

Specifies the EFS resource ID preference to set for the user's AWS account, in the current AWS Region, either `LONG_ID` (17 characters), or `SHORT_ID` (8 characters).

> **Note**
>
> Starting in October, 2021, you will receive an error when setting the account preference to SHORT_ID. Contact AWS support if you receive an error and must use short IDs for file system and mount target resources.

Type: String

Valid Values: `LONG_ID | SHORT_ID`

Required: Yes

## Response Syntax

```
HTTP/1.1 200
Content-type: application/json

{
   "ResourceIdPreference": {
      "ResourceIdType": "string",
      "Resources": [ "string" ]
   }
}
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**ResourceIdPreference**

Describes the resource type and its ID preference for the user's AWS account, in the current AWS Region.

Type: ResourceIdPreference object

# Errors

For information about the errors that are common to all actions, see Common Error Types.

**BadRequest**

Returned if the request is malformed or contains an error such as an invalid parameter value or a missing required parameter.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 400

**InternalServerError**

Returned if an error occurred on the server side.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 500

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

# PutBackupPolicy

Updates the file system's backup policy. Use this action to start or stop automatic backups of the file system.

## Request Syntax

```
PUT /2015-02-01/file-systems/FileSystemId/backup-policy HTTP/1.1
Content-type: application/json

{
   "BackupPolicy": {
      "Status": "string"
   }
}
```

## URI Request Parameters

The request uses the following URI parameters.

### FileSystemId

Specifies which EFS file system to update the backup policy for.

Length Constraints: Maximum length of 128.

Pattern: `^(arn:aws[-a-z]*:elasticfilesystem:[0-9a-z-:]+:file-system/fs-[0-9a-f]{8,40}|fs-[0-9a-f]{8,40})$`

Required: Yes

## Request Body

The request accepts the following data in JSON format.

### BackupPolicy

The backup policy included in the `PutBackupPolicy` request.

Type: BackupPolicy object

Required: Yes

## Response Syntax

```
HTTP/1.1 200
Content-type: application/json

{
   "BackupPolicy": {
      "Status": "string"
   }
}
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**BackupPolicy**

Describes the file system's backup policy, indicating whether automatic backups are turned on or off.

Type: BackupPolicy object

## Errors

For information about the errors that are common to all actions, see Common Error Types.

**BadRequest**

Returned if the request is malformed or contains an error such as an invalid parameter value or a missing required parameter.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end

user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 400

**FileSystemNotFound**

Returned if the specified `FileSystemId` value doesn't exist in the requester's AWS account.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 404

**IncorrectFileSystemLifeCycleState**

Returned if the file system's lifecycle state is not "available".

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 409

**InternalServerError**

Returned if an error occurred on the server side.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 500

**ValidationException**

Returned if the AWS Backup service is not available in the AWS Region in which the request was made.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 400

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4

- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

# PutFileSystemPolicy

Applies an Amazon EFS `FileSystemPolicy` to an Amazon EFS file system. A file system policy is an IAM resource-based policy and can contain multiple policy statements. A file system always has exactly one file system policy, which can be the default policy or an explicit policy set or updated using this API operation. EFS file system policies have a 20,000 character limit. When an explicit policy is set, it overrides the default policy. For more information about the default file system policy, see Default EFS file system policy.

> **Note**
>
> EFS file system policies have a 20,000 character limit.

This operation requires permissions for the `elasticfilesystem:PutFileSystemPolicy` action.

## Request Syntax

```
PUT /2015-02-01/file-systems/FileSystemId/policy HTTP/1.1
Content-type: application/json

{
   "BypassPolicyLockoutSafetyCheck": boolean,
   "Policy": "string"
}
```

## URI Request Parameters

The request uses the following URI parameters.

### FileSystemId

The ID of the EFS file system that you want to create or update the `FileSystemPolicy` for.

Length Constraints: Maximum length of 128.

Pattern: `^(arn:aws[-a-z]*:elasticfilesystem:[0-9a-z-:]+:file-system/fs-[0-9a-f]{8,40}|fs-[0-9a-f]{8,40})$`

Required: Yes

## Request Body

The request accepts the following data in JSON format.

**BypassPolicyLockoutSafetyCheck**

(Optional) A boolean that specifies whether or not to bypass the `FileSystemPolicy` lockout safety check. The lockout safety check determines whether the policy in the request will lock out, or prevent, the IAM principal that is making the request from making future `PutFileSystemPolicy` requests on this file system. Set `BypassPolicyLockoutSafetyCheck` to `True` only when you intend to prevent the IAM principal that is making the request from making subsequent `PutFileSystemPolicy` requests on this file system. The default value is `False`.

Type: Boolean

Required: No

**Policy**

The `FileSystemPolicy` that you're creating. Accepts a JSON formatted policy definition. EFS file system policies have a 20,000 character limit. To find out more about the elements that make up a file system policy, see Resource-based policies within Amazon EFS.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 20000.

Pattern: `[\s\S]+`

Required: Yes

## Response Syntax

```
HTTP/1.1 200
Content-type: application/json

{
   "FileSystemId": "string",
```

```
   "Policy": "string"
}
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**FileSystemId**

Specifies the EFS file system to which the `FileSystemPolicy` applies.

Type: String

Length Constraints: Maximum length of 128.

Pattern: `^(arn:aws[-a-z]*:elasticfilesystem:[0-9a-z-:]+:file-system/fs-[0-9a-f]{8,40}|fs-[0-9a-f]{8,40})$`

**Policy**

The JSON formatted `FileSystemPolicy` for the EFS file system.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 20000.

Pattern: `[\s\S]+`

## Errors

For information about the errors that are common to all actions, see Common Error Types.

**BadRequest**

Returned if the request is malformed or contains an error such as an invalid parameter value or a missing required parameter.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 400

**FileSystemNotFound**

Returned if the specified `FileSystemId` value doesn't exist in the requester's AWS account.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 404

**IncorrectFileSystemLifeCycleState**

Returned if the file system's lifecycle state is not "available".

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle.

Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 409

**InternalServerError**

Returned if an error occurred on the server side.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 500

**InvalidPolicyException**

Returned if the `FileSystemPolicy` is malformed or contains an error such as a parameter value that is not valid or a missing required parameter. Returned in the case of a policy lockout safety check error.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 400

# Examples

## Create an EFS FileSystemPolicy

The following request creates a `FileSystemPolicy` that allows all AWS principals to mount the specified EFS file system with read and write permissions.

### Sample Request

```
PUT /2015-02-01/file-systems/fs-01234567/file-system-policy HTTP/1.1
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "elasticfilesystem:ClientMount",
                "elasticfilesystem:ClientWrite"
            ],
            "Principal": {
                "AWS": ["*"]
            },
        }
    ]
}
```

### Sample Response

```
{
    "Version": "2012-10-17",
    "Id": "1",
    "Statement": [
        {
            "Sid": "efs-statement-abcdef01-1111-bbbb-2222-111122224444",
            "Effect": "Allow",
            "Action": [
                "elasticfilesystem:ClientMount",
                "elasticfilesystem:ClientWrite"
            ],
            "Principal": {
                "AWS": ["*"]
            },
```

```
            "Resource":"arn:aws:elasticfilesystem:us-east-1:1111222233334444:file-
system/fs-01234567"
        }
    ]
}
```

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

# PutLifecycleConfiguration

Use this action to manage storage for your file system. A `LifecycleConfiguration` consists of one or more `LifecyclePolicy` objects that define the following:

- **`TransitionToIA`** – When to move files in the file system from primary storage (Standard storage class) into the Infrequent Access (IA) storage.
- **`TransitionToArchive`** – When to move files in the file system from their current storage class (either IA or Standard storage) into the Archive storage.

  File systems cannot transition into Archive storage before transitioning into IA storage. Therefore, TransitionToArchive must either not be set or must be later than TransitionToIA.

  > **Note**
  >
  > The Archive storage class is available only for file systems that use the Elastic throughput mode and the General Purpose performance mode.

- **`TransitionToPrimaryStorageClass`** – Whether to move files in the file system back to primary storage (Standard storage class) after they are accessed in IA or Archive storage.

For more information, see Managing file system storage.

Each Amazon EFS file system supports one lifecycle configuration, which applies to all files in the file system. If a `LifecycleConfiguration` object already exists for the specified file system, a `PutLifecycleConfiguration` call modifies the existing configuration. A `PutLifecycleConfiguration` call with an empty `LifecyclePolicies` array in the request body deletes any existing `LifecycleConfiguration`. In the request, specify the following:

- The ID for the file system for which you are enabling, disabling, or modifying lifecycle management.
- A `LifecyclePolicies` array of `LifecyclePolicy` objects that define when to move files to IA storage, to Archive storage, and back to primary storage.

> **Note**
>
> Amazon EFS requires that each `LifecyclePolicy` object have only have a single transition, so the `LifecyclePolicies` array needs to be structured with separate `LifecyclePolicy` objects. See the example requests in the following section for more information.

This operation requires permissions for the `elasticfilesystem:PutLifecycleConfiguration` operation.

To apply a `LifecycleConfiguration` object to an encrypted file system, you need the same AWS Key Management Service permissions as when you created the encrypted file system.

## Request Syntax

```
PUT /2015-02-01/file-systems/FileSystemId/lifecycle-configuration HTTP/1.1
Content-type: application/json

{
   "LifecyclePolicies": [
      {
         "TransitionToArchive": "string",
         "TransitionToIA": "string",
         "TransitionToPrimaryStorageClass": "string"
      }
   ]
}
```

## URI Request Parameters

The request uses the following URI parameters.

### FileSystemId

The ID of the file system for which you are creating the `LifecycleConfiguration` object (String).

Length Constraints: Maximum length of 128.

Pattern: `^(arn:aws[-a-z]*:elasticfilesystem:[0-9a-z-:]+:file-system/fs-[0-9a-f]{8,40}|fs-[0-9a-f]{8,40})$`

Required: Yes

## Request Body

The request accepts the following data in JSON format.

### LifecyclePolicies

An array of `LifecyclePolicy` objects that define the file system's `LifecycleConfiguration` object. A `LifecycleConfiguration` object informs lifecycle management of the following:

- **`TransitionToIA`** – When to move files in the file system from primary storage (Standard storage class) into the Infrequent Access (IA) storage.
- **`TransitionToArchive`** – When to move files in the file system from their current storage class (either IA or Standard storage) into the Archive storage.

  File systems cannot transition into Archive storage before transitioning into IA storage. Therefore, TransitionToArchive must either not be set or must be later than TransitionToIA.

  > **Note**
  >
  > The Archive storage class is available only for file systems that use the Elastic throughput mode and the General Purpose performance mode.

- **`TransitionToPrimaryStorageClass`** – Whether to move files in the file system back to primary storage (Standard storage class) after they are accessed in IA or Archive storage.

> **Note**
>
> When using the `put-lifecycle-configuration` CLI command or the `PutLifecycleConfiguration` API action, Amazon EFS requires that each `LifecyclePolicy` object have only a single transition. This means that in a request body, `LifecyclePolicies` must be structured as an array of `LifecyclePolicy` objects, one object for each storage transition. See the example requests in the following section for more information.

Type: Array of LifecyclePolicy objects

Array Members: Maximum number of 3 items.

Required: Yes

## Response Syntax

```
HTTP/1.1 200
Content-type: application/json

{
   "LifecyclePolicies": [
      {
         "TransitionToArchive": "string",
         "TransitionToIA": "string",
         "TransitionToPrimaryStorageClass": "string"
      }
   ]
}
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**LifecyclePolicies**

An array of lifecycle management policies. EFS supports a maximum of one policy per file system.

Type: Array of LifecyclePolicy objects

Array Members: Maximum number of 3 items.

## Errors

For information about the errors that are common to all actions, see Common Error Types.

**BadRequest**

Returned if the request is malformed or contains an error such as an invalid parameter value or a missing required parameter.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 400

**FileSystemNotFound**

Returned if the specified `FileSystemId` value doesn't exist in the requester's AWS account.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 404

**IncorrectFileSystemLifeCycleState**

Returned if the file system's lifecycle state is not "available".

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 409

**InternalServerError**

Returned if an error occurred on the server side.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 500

# Examples

## Create a lifecycle configuration

The following example creates a `LifecyclePolicy` object using the `PutLifecycleConfiguration` action. This example creates a lifecycle policy that instructs EFS to do the following:

- Move all files in the file system that haven't been accessed in Standard storage within the last 30 days to IA storage.
- Move all files in the file system that haven't been accessed in Standard storage within the last 90 days to Archive storage.

- Move files back to Standard storage after they are accessed in IA or Archive storage. The Archive storage class is available only for file systems that use the Elastic throughput mode and the General Purpose performance mode.

For more information, see EFS storage classes and Managing file system storage.

### Sample Request

```
PUT /2015-02-01/file-systems/fs-0123456789abcdefb/lifecycle-configuration HTTP/1.1
Host: elasticfilesystem.us-west-2.amazonaws.com
x-amz-date: 20181122T232908Z
Authorization: <...>
Content-type: application/json
Content-Length: 86

{
   "LifecyclePolicies": [
      {
         "TransitionToArchive": "AFTER_90_DAYS"
      },
      {
         "TransitionToIA": "AFTER_30_DAYS"
      },
      {
         "TransitionToPrimaryStorage": "AFTER_1_ACCESS"
      }
   ]
}
```

### Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: 01234567-89ab-cdef-0123-456789abcdef
Content-type: application/json
Content-Length: 86

{
    "LifecyclePolicies": [
      {
         "TransitionToArchive": "AFTER_90_DAYS"
      },
```

```
      {
         "TransitionToIA": "AFTER_30_DAYS"
      },
      {
         "TransitionToPrimaryStorage": "AFTER_1_ACCESS"
      }
    ]
}
```

## Example put-lifecycle-configuration CLI request

This example illustrates one usage of PutLifecycleConfiguration.

### Sample Request

```
aws efs put-lifecycle-configuration \
   --file-system-id fs-0123456789abcdefb \
   --lifecycle-policies "[{"TransitionToArchive":"AFTER_90_DAYS"},
     {"TransitionToIA":"AFTER_30_DAYS"},
     {"TransitionToPrimaryStorageClass":"AFTER_1_ACCESS"}]
   --region us-west-2 \
   --profile adminuser
```

### Sample Response

```
{
   "LifecyclePolicies": [
       {
           "TransitionToArchive": "AFTER_90_DAYS"
       },
       {
           "TransitionToIA": "AFTER_30_DAYS"
       },
       {
           "TransitionToPrimaryStorageClass": "AFTER_1_ACCESS"
       }
   ]
}
```

## Disable lifecycle management

The following example disables lifecycle management for the specified file system.

### Sample Request

```
PUT /2015-02-01/file-systems/fs-01234567/lifecycle-configuration HTTP/1.1
Host: elasticfilesystem.us-west-2.amazonaws.com
x-amz-date: 20181122T232908Z
Authorization: <...>
Content-type: application/json
Content-Length: 86

{
   "LifecyclePolicies": [ ]
}
```

### Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: 01234567-89ab-cdef-0123-456789abcdef
Content-type: application/json
Content-Length: 86

{
   "LifecyclePolicies": [ ]
}
```

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3

- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

# TagResource

Creates a tag for an EFS resource. You can create tags for EFS file systems and access points using this API operation.

This operation requires permissions for the `elasticfilesystem:TagResource` action.

## Request Syntax

```
POST /2015-02-01/resource-tags/ResourceId HTTP/1.1
Content-type: application/json

{
   "Tags": [
      {
         "Key": "string",
         "Value": "string"
      }
   ]
}
```

## URI Request Parameters

The request uses the following URI parameters.

### ResourceId

The ID specifying the EFS resource that you want to create a tag for.

Length Constraints: Maximum length of 128.

Pattern: `^(arn:aws[-a-z]*:elasticfilesystem:[0-9a-z-:]+:(access-point/fsap|file-system/fs)-[0-9a-f]{8,40}|fs(ap)?-[0-9a-f]{8,40})$`

Required: Yes

## Request Body

The request accepts the following data in JSON format.

**Tags**

An array of Tag objects to add. Each Tag object is a key-value pair.

Type: Array of Tag objects

Required: Yes

## Response Syntax

```
HTTP/1.1 200
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

## Errors

For information about the errors that are common to all actions, see Common Error Types.

**AccessPointNotFound**

Returned if the specified AccessPointId value doesn't exist in the requester's AWS account.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 404

**BadRequest**

Returned if the request is malformed or contains an error such as an invalid parameter value or a missing required parameter.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 400

**FileSystemNotFound**

Returned if the specified `FileSystemId` value doesn't exist in the requester's AWS account.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 404

**InternalServerError**

Returned if an error occurred on the server side.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end

user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 500

# Examples

## Create Tags on a File System

The following request creates three tags ("key1", "key2", and "key3") on the specified file system.

### Sample Request

```
POST /2015-02-01/tag-resource/fs-01234567 HTTP/1.1
Host: elasticfilesystem.us-west-2.amazonaws.com
x-amz-date: 20140620T221118Z
Authorization: <...>
Content-Type: application/json
Content-Length: 160

{
    "Tags": [
        {
            "Key": "key1",
            "Value": "value1"
        },
        {
            "Key": "key2",
            "Value": "value2"
        },
        {
            "Key": "key3",
            "Value": "value3"
        }
    ]
}
```

### Sample Response

```
HTTP/1.1 204 no content
```

```
x-amzn-RequestId: 01234567-89ab-cdef-0123-456789abcdef
```

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

# UntagResource

Removes tags from an EFS resource. You can remove tags from EFS file systems and access points using this API operation.

This operation requires permissions for the `elasticfilesystem:UntagResource` action.

## Request Syntax

```
DELETE /2015-02-01/resource-tags/ResourceId?tagKeys=TagKeys HTTP/1.1
```

## URI Request Parameters

The request uses the following URI parameters.

**ResourceId**

Specifies the EFS resource that you want to remove tags from.

Length Constraints: Maximum length of 128.

Pattern: `^(arn:aws[-a-z]*:elasticfilesystem:[0-9a-z-:]+:(access-point/fsap|file-system/fs)-[0-9a-f]{8,40}|fs(ap)?-[0-9a-f]{8,40})$`

Required: Yes

**TagKeys**

The keys of the key-value tag pairs that you want to remove from the specified EFS resource.

Array Members: Minimum number of 1 item. Maximum number of 50 items.

Length Constraints: Minimum length of 1. Maximum length of 128.

Pattern: `^(?![aA]{1}[wW]{1}[sS]{1}:)([\p{L}\p{Z}\p{N}_.:/=+\-@]+)$`

Required: Yes

## Request Body

The request does not have a request body.

## Response Syntax

```
HTTP/1.1 200
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

## Errors

For information about the errors that are common to all actions, see Common Error Types.

**AccessPointNotFound**

Returned if the specified `AccessPointId` value doesn't exist in the requester's AWS account.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 404

**BadRequest**

Returned if the request is malformed or contains an error such as an invalid parameter value or a missing required parameter.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end

user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 400

**FileSystemNotFound**

Returned if the specified `FileSystemId` value doesn't exist in the requester's AWS account.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 404

**InternalServerError**

Returned if an error occurred on the server side.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 500

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

# UpdateFileSystem

Updates the throughput mode or the amount of provisioned throughput of an existing file system.

## Request Syntax

```
PUT /2015-02-01/file-systems/FileSystemId HTTP/1.1
Content-type: application/json

{
   "ProvisionedThroughputInMibps": number,
   "ThroughputMode": "string"
}
```

## URI Request Parameters

The request uses the following URI parameters.

### FileSystemId

The ID of the file system that you want to update.

Length Constraints: Maximum length of 128.

Pattern: `^(arn:aws[-a-z]*:elasticfilesystem:[0-9a-z-:]+:file-system/fs-[0-9a-f]{8,40}|fs-[0-9a-f]{8,40})$`

Required: Yes

## Request Body

The request accepts the following data in JSON format.

### ProvisionedThroughputInMibps

(Optional) The throughput, measured in mebibytes per second (MiBps), that you want to provision for a file system that you're creating. Required if `ThroughputMode` is set to `provisioned`. Valid values are 1-3414 MiBps, with the upper limit depending on Region. To increase this limit, contact Support. For more information, see Amazon EFS quotas that you can increase in the *Amazon EFS User Guide*.

Type: Double

Valid Range: Minimum value of 1.0.

Required: No

**ThroughputMode**

(Optional) Updates the file system's throughput mode. If you're not updating your throughput mode, you don't need to provide this value in your request. If you are changing the `ThroughputMode` to `provisioned`, you must also set a value for `ProvisionedThroughputInMibps`.

Type: String

Valid Values: `bursting | provisioned | elastic`

Required: No

## Response Syntax

```
HTTP/1.1 202
Content-type: application/json

{
   "AvailabilityZoneId": "string",
   "AvailabilityZoneName": "string",
   "CreationTime": number,
   "CreationToken": "string",
   "Encrypted": boolean,
   "FileSystemArn": "string",
   "FileSystemId": "string",
   "FileSystemProtection": {
      "ReplicationOverwriteProtection": "string"
   },
   "KmsKeyId": "string",
   "LifeCycleState": "string",
   "Name": "string",
   "NumberOfMountTargets": number,
   "OwnerId": "string",
   "PerformanceMode": "string",
   "ProvisionedThroughputInMibps": number,
   "SizeInBytes": {
```

```
      "Timestamp": number,
      "Value": number,
      "ValueInArchive": number,
      "ValueInIA": number,
      "ValueInStandard": number
   },
   "Tags": [
      {
         "Key": "string",
         "Value": "string"
      }
   ],
   "ThroughputMode": "string"
}
```

## Response Elements

If the action is successful, the service sends back an HTTP 202 response.

The following data is returned in JSON format by the service.

### AvailabilityZoneId

The unique and consistent identifier of the Availability Zone in which the file system is located, and is valid only for One Zone file systems. For example, `use1-az1` is an Availability Zone ID for the us-east-1 AWS Region, and it has the same location in every AWS account.

Type: String

### AvailabilityZoneName

Describes the AWS Availability Zone in which the file system is located, and is valid only for One Zone file systems. For more information, see Using EFS storage classes in the *Amazon EFS User Guide*.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: `.+`

### CreationTime

The time that the file system was created, in seconds (since 1970-01-01T00:00:00Z).

Type: Timestamp

**CreationToken**

The opaque string specified in the request.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: `.+`

**Encrypted**

A Boolean value that, if true, indicates that the file system is encrypted.

Type: Boolean

**FileSystemArn**

The Amazon Resource Name (ARN) for the EFS file system, in the format `arn:aws:elasticfilesystem:`*`region`*`:`*`account-id`*`:file-system/`*`file-system-id`* . Example with sample data: `arn:aws:elasticfilesystem:us-west-2:1111333322228888:file-system/fs-01234567`

Type: String

**FileSystemId**

The ID of the file system, assigned by Amazon EFS.

Type: String

Length Constraints: Maximum length of 128.

Pattern: `^(arn:aws[-a-z]*:elasticfilesystem:[0-9a-z-:]+:file-system/fs-[0-9a-f]{8,40}|fs-[0-9a-f]{8,40})$`

**FileSystemProtection**

Describes the protection on the file system.

Type: FileSystemProtectionDescription object

**KmsKeyId**

The ID of an AWS KMS key used to protect the encrypted file system.

Type: String

Length Constraints: Maximum length of 2048.

Pattern: `^([0-9a-f]{8}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{12}|mrk-[0-9a-f]{32}|alias/[a-zA-Z0-9/_-]+|(arn:aws[-a-z]*:kms:[a-z0-9-]+:\d{12}:((key/[0-9a-f]{8}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{12})|(key/mrk-[0-9a-f]{32})|(alias/[a-zA-Z0-9/_-]+))))$`

**LifeCycleState**

The lifecycle phase of the file system.

Type: String

Valid Values: `creating | available | updating | deleting | deleted | error`

**Name**

You can add tags to a file system, including a `Name` tag. For more information, see CreateFileSystem. If the file system has a `Name` tag, Amazon EFS returns the value in this field.

Type: String

Length Constraints: Maximum length of 256.

Pattern: `^([\p{L}\p{Z}\p{N}_.:/=+\-@]*)$`

**NumberOfMountTargets**

The current number of mount targets that the file system has. For more information, see CreateMountTarget.

Type: Integer

Valid Range: Minimum value of 0.

**OwnerId**

The AWS account that created the file system.

Type: String

Length Constraints: Maximum length of 14.

Pattern: `^(\d{12})|(\d{4}-\d{4}-\d{4})$`

### PerformanceMode

The performance mode of the file system.

Type: String

Valid Values: `generalPurpose | maxIO`

### ProvisionedThroughputInMibps

The amount of provisioned throughput, measured in MiBps, for the file system. Valid for file systems using `ThroughputMode` set to `provisioned`.

Type: Double

Valid Range: Minimum value of 1.0.

### SizeInBytes

The latest known metered size (in bytes) of data stored in the file system, in its `Value` field, and the time at which that size was determined in its `Timestamp` field. The `Timestamp` value is the integer number of seconds since 1970-01-01T00:00:00Z. The `SizeInBytes` value doesn't represent the size of a consistent snapshot of the file system, but it is eventually consistent when there are no writes to the file system. That is, `SizeInBytes` represents actual size only if the file system is not modified for a period longer than a couple of hours. Otherwise, the value is not the exact size that the file system was at any point in time.

Type: FileSystemSize object

### Tags

The tags associated with the file system, presented as an array of `Tag` objects.

Type: Array of Tag objects

### ThroughputMode

Displays the file system's throughput mode. For more information, see Throughput modes in the *Amazon EFS User Guide*.

Type: String

Valid Values: `bursting | provisioned | elastic`

## Errors

For information about the errors that are common to all actions, see Common Error Types.

**BadRequest**

Returned if the request is malformed or contains an error such as an invalid parameter value or a missing required parameter.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 400

**FileSystemNotFound**

Returned if the specified `FileSystemId` value doesn't exist in the requester's AWS account.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 404

**IncorrectFileSystemLifeCycleState**

Returned if the file system's lifecycle state is not "available".

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 409

**InsufficientThroughputCapacity**

Returned if there's not enough capacity to provision additional throughput. This value might be returned when you try to create a file system in provisioned throughput mode, when you attempt to increase the provisioned throughput of an existing file system, or when you attempt to change an existing file system from Bursting Throughput to Provisioned Throughput mode. Try again later.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 503

**InternalServerError**

Returned if an error occurred on the server side.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 500

**ThroughputLimitExceeded**

Returned if the throughput mode or amount of provisioned throughput can't be changed because the throughput limit of 1024 MiB/s has been reached.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 400

**TooManyRequests**

Returned if you don’t wait at least 24 hours before either changing the throughput mode, or decreasing the Provisioned Throughput value.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 429

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

# UpdateFileSystemProtection

Updates protection on the file system.

This operation requires permissions for the `elasticfilesystem:UpdateFileSystemProtection` action.

## Request Syntax

```
PUT /2015-02-01/file-systems/FileSystemId/protection HTTP/1.1
Content-type: application/json

{
   "ReplicationOverwriteProtection": "string"
}
```

## URI Request Parameters

The request uses the following URI parameters.

### FileSystemId

The ID of the file system to update.

Length Constraints: Maximum length of 128.

Pattern: `^(arn:aws[-a-z]*:elasticfilesystem:[0-9a-z-:]+:file-system/fs-[0-9a-f]{8,40}|fs-[0-9a-f]{8,40})$`

Required: Yes

## Request Body

The request accepts the following data in JSON format.

### ReplicationOverwriteProtection

The status of the file system's replication overwrite protection.

- ENABLED – The file system cannot be used as the destination file system in a replication configuration. The file system is writeable. Replication overwrite protection is ENABLED by default.

- DISABLED – The file system can be used as the destination file system in a replication configuration. The file system is read-only and can only be modified by EFS replication.
- REPLICATING – The file system is being used as the destination file system in a replication configuration. The file system is read-only and is only modified only by EFS replication.

If the replication configuration is deleted, the file system's replication overwrite protection is re-enabled and the file system becomes writeable.

Type: String

Valid Values: ENABLED | DISABLED | REPLICATING

Required: No

## Response Syntax

```
HTTP/1.1 200
Content-type: application/json

{
   "ReplicationOverwriteProtection": "string"
}
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

### ReplicationOverwriteProtection

The status of the file system's replication overwrite protection.

- ENABLED – The file system cannot be used as the destination file system in a replication configuration. The file system is writeable. Replication overwrite protection is ENABLED by default.
- DISABLED – The file system can be used as the destination file system in a replication configuration. The file system is read-only and can only be modified by EFS replication.
- REPLICATING – The file system is being used as the destination file system in a replication configuration. The file system is read-only and is modified only by EFS replication.

If the replication configuration is deleted, the file system's replication overwrite protection is re-enabled, the file system becomes writeable.

Type: String

Valid Values: `ENABLED | DISABLED | REPLICATING`

## Errors

For information about the errors that are common to all actions, see Common Error Types.

**BadRequest**

Returned if the request is malformed or contains an error such as an invalid parameter value or a missing required parameter.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 400

**FileSystemNotFound**

Returned if the specified `FileSystemId` value doesn't exist in the requester's AWS account.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end

user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 404

**IncorrectFileSystemLifeCycleState**

Returned if the file system's lifecycle state is not "available".

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 409

**InsufficientThroughputCapacity**

Returned if there's not enough capacity to provision additional throughput. This value might be returned when you try to create a file system in provisioned throughput mode, when you attempt to increase the provisioned throughput of an existing file system, or when you attempt to change an existing file system from Bursting Throughput to Provisioned Throughput mode. Try again later.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 503

**InternalServerError**

Returned if an error occurred on the server side.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 500

**ReplicationAlreadyExists**

Returned if the file system is already included in a replication configuration.>

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 409

**ThroughputLimitExceeded**

Returned if the throughput mode or amount of provisioned throughput can't be changed because the throughput limit of 1024 MiB/s has been reached.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 400

**TooManyRequests**

Returned if you don’t wait at least 24 hours before either changing the throughput mode, or decreasing the Provisioned Throughput value.

**ErrorCode**

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

HTTP Status Code: 429

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface V2
- AWS SDK for .NET V4

- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for Kotlin
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

# Data Types

The Amazon Elastic File System API contains several data types that various actions use. This section describes each data type in detail.

> **Note**
>
> The order of each element in a data type structure is not guaranteed. Applications should not assume a particular order.

The following data types are supported:

- AccessPointDescription
- BackupPolicy
- CreationInfo
- Destination
- DestinationToCreate
- FileSystemDescription
- FileSystemProtectionDescription
- FileSystemSize
- LifecyclePolicy
- MountTargetDescription
- PosixUser
- ReplicationConfigurationDescription
- ResourceIdPreference
- RootDirectory
- Tag

# AccessPointDescription

Provides a description of an EFS file system access point.

## Contents

**AccessPointArn**

The unique Amazon Resource Name (ARN) associated with the access point.

Type: String

Length Constraints: Maximum length of 128.

Pattern: `^arn:aws[-a-z]*:elasticfilesystem:[0-9a-z-:]+:access-point/fsap-[0-9a-f]{8,40}$`

Required: No

**AccessPointId**

The ID of the access point, assigned by Amazon EFS.

Type: String

Length Constraints: Maximum length of 128.

Pattern: `^(arn:aws[-a-z]*:elasticfilesystem:[0-9a-z-:]+:access-point/fsap-[0-9a-f]{8,40}|fsap-[0-9a-f]{8,40})$`

Required: No

**ClientToken**

The opaque string specified in the request to ensure idempotent creation.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: `.+`

Required: No

**FileSystemId**

The ID of the EFS file system that the access point applies to.

Type: String

Length Constraints: Maximum length of 128.

Pattern: `^(arn:aws[-a-z]*:elasticfilesystem:[0-9a-z-:]+:file-system/fs-[0-9a-f]{8,40}|fs-[0-9a-f]{8,40})$`

Required: No

**LifeCycleState**

Identifies the lifecycle phase of the access point.

Type: String

Valid Values: `creating | available | updating | deleting | deleted | error`

Required: No

**Name**

The name of the access point. This is the value of the `Name` tag.

Type: String

Required: No

**OwnerId**

Identifies the AWS account that owns the access point resource.

Type: String

Length Constraints: Maximum length of 14.

Pattern: `^(\d{12})|(\d{4}-\d{4}-\d{4})$`

Required: No

**PosixUser**

The full POSIX identity, including the user ID, group ID, and secondary group IDs on the access point that is used for all file operations by NFS clients using the access point.

Type: PosixUser object

Required: No

**RootDirectory**

The directory on the EFS file system that the access point exposes as the root directory to NFS clients using the access point.

Type: RootDirectory object

Required: No

**Tags**

The tags associated with the access point, presented as an array of Tag objects.

Type: Array of Tag objects

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

# BackupPolicy

The backup policy for the file system used to create automatic daily backups. If status has a value of ENABLED, the file system is being automatically backed up. For more information, see Automatic backups.

## Contents

**Status**

Describes the status of the file system's backup policy.

- **ENABLED** – EFS is automatically backing up the file system.
- **ENABLING** – EFS is turning on automatic backups for the file system.
- **DISABLED** – Automatic back ups are turned off for the file system.
- **DISABLING** – EFS is turning off automatic backups for the file system.

Type: String

Valid Values: ENABLED | ENABLING | DISABLED | DISABLING

Required: Yes

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

# CreationInfo

Required if the `RootDirectory` > `Path` specified does not exist. Specifies the POSIX IDs and permissions to apply to the access point's `RootDirectory` > `Path`. If the access point root directory does not exist, EFS creates it with these settings when a client connects to the access point. When specifying `CreationInfo`, you must include values for all properties.

Amazon EFS creates a root directory only if you have provided the CreationInfo: OwnUid, OwnGID, and permissions for the directory. If you do not provide this information, Amazon EFS does not create the root directory. If the root directory does not exist, attempts to mount using the access point will fail.

> **Important**
>
> If you do not provide `CreationInfo` and the specified `RootDirectory` does not exist, attempts to mount the file system using the access point will fail.

## Contents

**OwnerGid**

Specifies the POSIX group ID to apply to the `RootDirectory`. Accepts values from 0 to 2^32 (4294967295).

Type: Long

Valid Range: Minimum value of 0. Maximum value of 4294967295.

Required: Yes

**OwnerUid**

Specifies the POSIX user ID to apply to the `RootDirectory`. Accepts values from 0 to 2^32 (4294967295).

Type: Long

Valid Range: Minimum value of 0. Maximum value of 4294967295.

Required: Yes

**Permissions**

Specifies the POSIX permissions to apply to the `RootDirectory`, in the format of an octal number representing the file's mode bits.

Type: String

Length Constraints: Minimum length of 3. Maximum length of 4.

Pattern: `^[0-7]{3,4}$`

Required: Yes

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

# Destination

Describes the destination file system in the replication configuration.

## Contents

**FileSystemId**

The ID of the destination Amazon EFS file system.

Type: String

Length Constraints: Maximum length of 128.

Pattern: `^(arn:aws[-a-z]*:elasticfilesystem:[0-9a-z-:]+:file-system/fs-[0-9a-f]{8,40}|fs-[0-9a-f]{8,40})$`

Required: Yes

**Region**

The AWS Region in which the destination file system is located.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: `^[a-z]{2}-((iso[a-z]{0,1}-)|(gov-)){0,1}[a-z]+-{0,1}[0-9]{0,1}$`

Required: Yes

**Status**

Describes the status of the replication configuration. For more information about replication status, see Viewing replication details in the *Amazon EFS User Guide*.

Type: String

Valid Values: `ENABLED | ENABLING | DELETING | ERROR | PAUSED | PAUSING`

Required: Yes

**LastReplicatedTimestamp**

The time when the most recent sync was successfully completed on the destination file system. Any changes to data on the source file system that occurred before this time have been successfully replicated to the destination file system. Any changes that occurred after this time might not be fully replicated.

Type: Timestamp

Required: No

**OwnerId**

ID of the AWS account in which the destination file system resides.

Type: String

Length Constraints: Maximum length of 14.

Pattern: `^(\d{12})|(\d{4}-\d{4}-\d{4})$`

Required: No

**RoleArn**

Amazon Resource Name (ARN) of the IAM role in the source account that allows Amazon EFS to perform replication on its behalf. This is optional for same-account replication and required for cross-account replication.

Type: String

Length Constraints: Maximum length of 2048.

Pattern: `arn:(aws[a-zA-Z-]*)?:iam::\d{12}:role/?[a-zA-Z_0-9+=,.@\-_/]+`

Required: No

**StatusMessage**

Message that provides details about the PAUSED or ERRROR state of the replication destination configuration. For more information about replication status messages, see Viewing replication details in the *Amazon EFS User Guide*.

Type: String

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

# DestinationToCreate

Describes the new or existing destination file system for the replication configuration.

- If you want to replicate to a new file system, do not specify the File System ID for the destination file system. Amazon EFS creates a new, empty file system. For One Zone storage, specify the Availability Zone to create the file system in. To use an AWS Key Management Service key other than the default KMS key, then specify it. For more information, see Configuring replication to new Amazon EFS file system in the *Amazon EFS User Guide*.

  > **Note**
  >
  > After the file system is created, you cannot change the KMS key or the performance mode.

- If you want to replicate to an existing file system that's in the same account as the source file system, then you need to provide the ID or Amazon Resource Name (ARN) of the file system to which to replicate. The file system's replication overwrite protection must be disabled. For more information, see Replicating to an existing file system in the *Amazon EFS User Guide*.
- If you are replicating the file system to a file system that's in a different account than the source file system (cross-account replication), you need to provide the ARN for the file system and the IAM role that allows Amazon EFS to perform replication on the destination account. The file system's replication overwrite protection must be disabled. For more information, see Replicating across AWS accounts in the *Amazon EFS User Guide*.

## Contents

### AvailabilityZoneName

To create a file system that uses One Zone storage, specify the name of the Availability Zone in which to create the destination file system.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: `.+`

Required: No

**FileSystemId**

The ID or ARN of the file system to use for the destination. For cross-account replication, this must be an ARN. The file system's replication overwrite replication must be disabled. If no ID or ARN is specified, then a new file system is created.

> **Note**
>
> When you initially configure replication to an existing file system, Amazon EFS writes data to or removes existing data from the destination file system to match data in the source file system. If you don't want to change data in the destination file system, then you should replicate to a new file system instead. For more information, see https://docs.aws.amazon.com/efs/latest/ug/create-replication.html.

Type: String

Length Constraints: Maximum length of 128.

Pattern: `^(arn:aws[-a-z]*:elasticfilesystem:[0-9a-z-:]+:file-system/fs-[0-9a-f]{8,40}|fs-[0-9a-f]{8,40})$`

Required: No

**KmsKeyId**

Specify the AWS Key Management Service (AWS KMS) key that you want to use to encrypt the destination file system. If you do not specify a KMS key, Amazon EFS uses your default KMS key for Amazon EFS, `/aws/elasticfilesystem`. This ID can be in one of the following formats:

- Key ID - The unique identifier of the key, for example `1234abcd-12ab-34cd-56ef-1234567890ab`.
- ARN - The ARN for the key, for example `arn:aws:kms:us-west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab`.
- Key alias - A previously created display name for a key, for example `alias/projectKey1`.
- Key alias ARN - The ARN for a key alias, for example `arn:aws:kms:us-west-2:444455556666:alias/projectKey1`.

Type: String

Length Constraints: Maximum length of 2048.

Pattern: `^([0-9a-f]{8}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{12}|mrk-[0-9a-f]{32}|alias/[a-zA-Z0-9/_-]+|(arn:aws[-a-z]*:kms:[a-z0-9-]+:\d{12}:((key/[0-9a-f]{8}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{12})|(key/mrk-[0-9a-f]{32})|(alias/[a-zA-Z0-9/_-]+))))$`

Required: No

**Region**

To create a file system that uses Regional storage, specify the AWS Region in which to create the destination file system. The Region must be enabled for the AWS account that owns the source file system. For more information, see Managing AWS Regions in the *AWS General Reference Reference Guide*.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: `^[a-z]{2}-((iso[a-z]{0,1}-)|(gov-)){0,1}[a-z]+-{0,1}[0-9]{0,1}$`

Required: No

**RoleArn**

Amazon Resource Name (ARN) of the IAM role in the source account that allows Amazon EFS to perform replication on its behalf. This is optional for same-account replication and required for cross-account replication.

Type: String

Length Constraints: Maximum length of 2048.

Pattern: `arn:(aws[a-zA-Z-]*)?:iam::\d{12}:role/?[a-zA-Z_0-9+=,.@\-_/]+`

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++

- AWS SDK for Java V2
- AWS SDK for Ruby V3

# FileSystemDescription

A description of the file system.

## Contents

**CreationTime**

The time that the file system was created, in seconds (since 1970-01-01T00:00:00Z).

Type: Timestamp

Required: Yes

**CreationToken**

The opaque string specified in the request.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: `.+`

Required: Yes

**FileSystemId**

The ID of the file system, assigned by Amazon EFS.

Type: String

Length Constraints: Maximum length of 128.

Pattern: `^(arn:aws[-a-z]*:elasticfilesystem:[0-9a-z-:]+:file-system/fs-[0-9a-f]{8,40}|fs-[0-9a-f]{8,40})$`

Required: Yes

**LifeCycleState**

The lifecycle phase of the file system.

Type: String

Valid Values: `creating | available | updating | deleting | deleted | error`

Required: Yes

**NumberOfMountTargets**

The current number of mount targets that the file system has. For more information, see CreateMountTarget.

Type: Integer

Valid Range: Minimum value of 0.

Required: Yes

**OwnerId**

The AWS account that created the file system.

Type: String

Length Constraints: Maximum length of 14.

Pattern: `^(\d{12})|(\d{4}-\d{4}-\d{4})$`

Required: Yes

**PerformanceMode**

The performance mode of the file system.

Type: String

Valid Values: `generalPurpose | maxIO`

Required: Yes

**SizeInBytes**

The latest known metered size (in bytes) of data stored in the file system, in its `Value` field, and the time at which that size was determined in its `Timestamp` field. The `Timestamp` value is the integer number of seconds since 1970-01-01T00:00:00Z. The `SizeInBytes` value doesn't represent the size of a consistent snapshot of the file system, but it is eventually consistent when there are no writes to the file system. That is, `SizeInBytes` represents actual size only if

the file system is not modified for a period longer than a couple of hours. Otherwise, the value is not the exact size that the file system was at any point in time.

Type: FileSystemSize object

Required: Yes

**Tags**

The tags associated with the file system, presented as an array of Tag objects.

Type: Array of Tag objects

Required: Yes

**AvailabilityZoneId**

The unique and consistent identifier of the Availability Zone in which the file system is located, and is valid only for One Zone file systems. For example, use1-az1 is an Availability Zone ID for the us-east-1 AWS Region, and it has the same location in every AWS account.

Type: String

Required: No

**AvailabilityZoneName**

Describes the AWS Availability Zone in which the file system is located, and is valid only for One Zone file systems. For more information, see Using EFS storage classes in the *Amazon EFS User Guide*.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: .+

Required: No

**Encrypted**

A Boolean value that, if true, indicates that the file system is encrypted.

Type: Boolean

Required: No

**FileSystemArn**

The Amazon Resource Name (ARN) for the EFS file system, in the format `arn:aws:elasticfilesystem:region:account-id:file-system/file-system-id` . Example with sample data: `arn:aws:elasticfilesystem:us-west-2:1111333322228888:file-system/fs-01234567`

Type: String

Required: No

**FileSystemProtection**

Describes the protection on the file system.

Type: FileSystemProtectionDescription object

Required: No

**KmsKeyId**

The ID of an AWS KMS key used to protect the encrypted file system.

Type: String

Length Constraints: Maximum length of 2048.

Pattern: `^([0-9a-f]{8}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{12}|mrk-[0-9a-f]{32}|alias/[a-zA-Z0-9/_-]+|(arn:aws[-a-z]*:kms:[a-z0-9-]+:\d{12}:((key/[0-9a-f]{8}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{12})|(key/mrk-[0-9a-f]{32})|(alias/[a-zA-Z0-9/_-]+))))$`

Required: No

**Name**

You can add tags to a file system, including a `Name` tag. For more information, see CreateFileSystem. If the file system has a `Name` tag, Amazon EFS returns the value in this field.

Type: String

Length Constraints: Maximum length of 256.

Pattern: `^([\p{L}\p{Z}\p{N}_.:/=+\-@]*)$`

Required: No

**ProvisionedThroughputInMibps**

The amount of provisioned throughput, measured in MiBps, for the file system. Valid for file systems using `ThroughputMode` set to `provisioned`.

Type: Double

Valid Range: Minimum value of 1.0.

Required: No

**ThroughputMode**

Displays the file system's throughput mode. For more information, see Throughput modes in the *Amazon EFS User Guide*.

Type: String

Valid Values: `bursting | provisioned | elastic`

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

# FileSystemProtectionDescription

Describes the protection on a file system.

## Contents

**ReplicationOverwriteProtection**

The status of the file system's replication overwrite protection.

- `ENABLED` – The file system cannot be used as the destination file system in a replication configuration. The file system is writeable. Replication overwrite protection is `ENABLED` by default.
- `DISABLED` – The file system can be used as the destination file system in a replication configuration. The file system is read-only and can only be modified by EFS replication.
- `REPLICATING` – The file system is being used as the destination file system in a replication configuration. The file system is read-only and is modified only by EFS replication.

If the replication configuration is deleted, the file system's replication overwrite protection is re-enabled, the file system becomes writeable.

Type: String

Valid Values: `ENABLED | DISABLED | REPLICATING`

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

# FileSystemSize

The latest known metered size (in bytes) of data stored in the file system, in its `Value` field, and the time at which that size was determined in its `Timestamp` field. The value doesn't represent the size of a consistent snapshot of the file system, but it is eventually consistent when there are no writes to the file system. That is, the value represents the actual size only if the file system is not modified for a period longer than a couple of hours. Otherwise, the value is not necessarily the exact size the file system was at any instant in time.

## Contents

**Value**

The latest known metered size (in bytes) of data stored in the file system.

Type: Long

Valid Range: Minimum value of 0.

Required: Yes

**Timestamp**

The time at which the size of data, returned in the `Value` field, was determined. The value is the integer number of seconds since 1970-01-01T00:00:00Z.

Type: Timestamp

Required: No

**ValueInArchive**

The latest known metered size (in bytes) of data stored in the Archive storage class.

Type: Long

Valid Range: Minimum value of 0.

Required: No

**ValueInIA**

The latest known metered size (in bytes) of data stored in the Infrequent Access storage class.

Type: Long

Valid Range: Minimum value of 0.

Required: No

**ValueInStandard**

The latest known metered size (in bytes) of data stored in the Standard storage class.

Type: Long

Valid Range: Minimum value of 0.

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

# LifecyclePolicy

Describes a policy used by lifecycle management that specifies when to transition files into and out of storage classes. For more information, see Managing file system storage.

**Note**

When using the `put-lifecycle-configuration` CLI command or the `PutLifecycleConfiguration` API action, Amazon EFS requires that each `LifecyclePolicy` object have only a single transition. This means that in a request body, `LifecyclePolicies` must be structured as an array of `LifecyclePolicy` objects, one object for each transition. For more information, see the request examples in PutLifecycleConfiguration.

## Contents

**TransitionToArchive**

The number of days after files were last accessed in primary storage (the Standard storage class) at which to move them to Archive storage. Metadata operations such as listing the contents of a directory don't count as file access events.

Type: String

Valid Values: `AFTER_1_DAY | AFTER_7_DAYS | AFTER_14_DAYS | AFTER_30_DAYS | AFTER_60_DAYS | AFTER_90_DAYS | AFTER_180_DAYS | AFTER_270_DAYS | AFTER_365_DAYS`

Required: No

**TransitionToIA**

The number of days after files were last accessed in primary storage (the Standard storage class) at which to move them to Infrequent Access (IA) storage. Metadata operations such as listing the contents of a directory don't count as file access events.

Type: String

Valid Values: `AFTER_7_DAYS | AFTER_14_DAYS | AFTER_30_DAYS | AFTER_60_DAYS | AFTER_90_DAYS | AFTER_1_DAY | AFTER_180_DAYS | AFTER_270_DAYS | AFTER_365_DAYS`

Required: No

**TransitionToPrimaryStorageClass**

Whether to move files back to primary (Standard) storage after they are accessed in IA or Archive storage. Metadata operations such as listing the contents of a directory don't count as file access events.

Type: String

Valid Values: `AFTER_1_ACCESS`

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

# MountTargetDescription

Provides a description of a mount target.

## Contents

**FileSystemId**

The ID of the file system for which the mount target is intended.

Type: String

Length Constraints: Maximum length of 128.

Pattern: `^(arn:aws[-a-z]*:elasticfilesystem:[0-9a-z-:]+:file-system/fs-[0-9a-f]{8,40}|fs-[0-9a-f]{8,40})$`

Required: Yes

**LifeCycleState**

Lifecycle state of the mount target.

Type: String

Valid Values: `creating | available | updating | deleting | deleted | error`

Required: Yes

**MountTargetId**

System-assigned mount target ID.

Type: String

Length Constraints: Minimum length of 13. Maximum length of 45.

Pattern: `^fsmt-[0-9a-f]{8,40}$`

Required: Yes

**SubnetId**

The ID of the mount target's subnet.

Type: String

Length Constraints: Minimum length of 15. Maximum length of 47.

Pattern: `^subnet-[0-9a-f]{8,40}$`

Required: Yes

**AvailabilityZoneId**

The unique and consistent identifier of the Availability Zone that the mount target resides in. For example, `use1-az1` is an AZ ID for the us-east-1 Region and it has the same location in every AWS account.

Type: String

Required: No

**AvailabilityZoneName**

The name of the Availability Zone in which the mount target is located. Availability Zones are independently mapped to names for each AWS account. For example, the Availability Zone `us-east-1a` for your AWS account might not be the same location as `us-east-1a` for another AWS account.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: `.+`

Required: No

**IpAddress**

The IPv4 address for the mount target.

Type: String

Length Constraints: Minimum length of 7. Maximum length of 15.

Pattern: `^[0-9]{1,3}\.[0-9]{1,3}\.[0-9]{1,3}\.[0-9]{1,3}$`

Required: No

**Ipv6Address**

The IPv6 address for the mount target.

Type: String

Length Constraints: Minimum length of 3. Maximum length of 39.

Required: No

**NetworkInterfaceId**

The ID of the network interface that Amazon EFS created when it created the mount target.

Type: String

Required: No

**OwnerId**

AWS account ID that owns the resource.

Type: String

Length Constraints: Maximum length of 14.

Pattern: `^(\d{12})|(\d{4}-\d{4}-\d{4})$`

Required: No

**VpcId**

The virtual private cloud (VPC) ID that the mount target is configured in.

Type: String

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++

- AWS SDK for Java V2
- AWS SDK for Ruby V3

# PosixUser

The full POSIX identity, including the user ID, group ID, and any secondary group IDs, on the access point that is used for all file system operations performed by NFS clients using the access point.

## Contents

**Gid**

The POSIX group ID used for all file system operations using this access point.

Type: Long

Valid Range: Minimum value of 0. Maximum value of 4294967295.

Required: Yes

**Uid**

The POSIX user ID used for all file system operations using this access point.

Type: Long

Valid Range: Minimum value of 0. Maximum value of 4294967295.

Required: Yes

**SecondaryGids**

Secondary POSIX group IDs used for all file system operations using this access point.

Type: Array of longs

Array Members: Minimum number of 0 items. Maximum number of 16 items.

Valid Range: Minimum value of 0. Maximum value of 4294967295.

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

# ReplicationConfigurationDescription

Describes the replication configuration for a specific file system.

## Contents

**CreationTime**

Describes when the replication configuration was created.

Type: Timestamp

Required: Yes

**Destinations**

An array of destination objects. Only one destination object is supported.

Type: Array of Destination objects

Required: Yes

**OriginalSourceFileSystemArn**

The Amazon Resource Name (ARN) of the original source EFS file system in the replication configuration.

Type: String

Required: Yes

**SourceFileSystemArn**

The Amazon Resource Name (ARN) of the current source file system in the replication configuration.

Type: String

Required: Yes

**SourceFileSystemId**

The ID of the source Amazon EFS file system that is being replicated.

Type: String

Length Constraints: Maximum length of 128.

Pattern: `^(arn:aws[-a-z]*:elasticfilesystem:[0-9a-z-:]+:file-system/fs-[0-9a-f]{8,40}|fs-[0-9a-f]{8,40})$`

Required: Yes

**SourceFileSystemRegion**

The AWS Region in which the source EFS file system is located.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: `^[a-z]{2}-((iso[a-z]{0,1}-)|(gov-)){0,1}[a-z]+-{0,1}[0-9]{0,1}$`

Required: Yes

**SourceFileSystemOwnerId**

ID of the AWS account in which the source file system resides.

Type: String

Length Constraints: Maximum length of 14.

Pattern: `^(\d{12})|(\d{4}-\d{4}-\d{4})$`

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

# ResourceIdPreference

Describes the resource type and its ID preference for the user's AWS account, in the current AWS Region.

## Contents

**ResourceIdType**

Identifies the EFS resource ID preference, either `LONG_ID` (17 characters) or `SHORT_ID` (8 characters).

Type: String

Valid Values: `LONG_ID | SHORT_ID`

Required: No

**Resources**

Identifies the Amazon EFS resources to which the ID preference setting applies, `FILE_SYSTEM` and `MOUNT_TARGET`.

Type: Array of strings

Valid Values: `FILE_SYSTEM | MOUNT_TARGET`

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

# RootDirectory

Specifies the directory on the Amazon EFS file system that the access point provides access to. The access point exposes the specified file system path as the root directory of your file system to applications using the access point. NFS clients using the access point can only access data in the access point's `RootDirectory` and its subdirectories.

## Contents

**CreationInfo**

(Optional) Specifies the POSIX IDs and permissions to apply to the access point's `RootDirectory`. If the `RootDirectory` > `Path` specified does not exist, EFS creates the root directory using the `CreationInfo` settings when a client connects to an access point. When specifying the `CreationInfo`, you must provide values for all properties.

> **Important**
>
> If you do not provide `CreationInfo` and the specified `RootDirectory` > `Path` does not exist, attempts to mount the file system using the access point will fail.

Type: CreationInfo object

Required: No

**Path**

Specifies the path on the EFS file system to expose as the root directory to NFS clients using the access point to access the EFS file system. A path can have up to four subdirectories. If the specified path does not exist, you are required to provide the `CreationInfo`.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 100.

Pattern: `^(\/|(\/(?!\.)+[^$#<>;`|&?{}^*/\n]+){1,4})$`

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

# Tag

A tag is a key-value pair. Allowed characters are letters, white space, and numbers that can be represented in UTF-8, and the following characters: `+ - = . _ : /`.

## Contents

**Key**

The tag key (String). The key can't start with `aws:`.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 128.

Pattern: `^(?![aA]{1}[wW]{1}[sS]{1}:)([\p{L}\p{Z}\p{N}_.:/=+\-@]+)$`

Required: Yes

**Value**

The value of the tag key.

Type: String

Length Constraints: Maximum length of 256.

Pattern: `^([\p{L}\p{Z}\p{N}_.:/=+\-@]*)$`

Required: Yes

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

# Common Parameters

The following list contains the parameters that all actions use for signing Signature Version 4 requests with a query string. Any action-specific parameters are listed in the topic for that action. For more information about Signature Version 4, see Signing AWS API requests in the *IAM User Guide*.

**X-Amz-Algorithm**

The hash algorithm that you used to create the request signature.

Condition: Specify this parameter when you include authentication information in a query string instead of in the HTTP authorization header.

Type: string

Valid Values: `AWS4-HMAC-SHA256`

Required: Conditional

**X-Amz-Credential**

The credential scope value, which is a string that includes your access key, the date, the region you are targeting, the service you are requesting, and a termination string ("aws4_request"). The value is expressed in the following format: *access_key*/*YYYYMMDD*/*region*/*service*/aws4_request.

For more information, see Create a signed AWS API request in the *IAM User Guide*.

Condition: Specify this parameter when you include authentication information in a query string instead of in the HTTP authorization header.

Type: string

Required: Conditional

**X-Amz-Date**

The date that is used to create the signature. The format must be ISO 8601 basic format (YYYYMMDD'T'HHMMSS'Z'). For example, the following date time is a valid X-Amz-Date value: `20120325T120000Z`.

Condition: X-Amz-Date is optional for all requests; it can be used to override the date used for signing requests. If the Date header is specified in the ISO 8601 basic format, X-Amz-Date is not required. When X-Amz-Date is used, it always overrides the value of the Date header. For more information, see Elements of an AWS API request signature in the *IAM User Guide*.

Type: string

Required: Conditional

**X-Amz-Security-Token**

The temporary security token that was obtained through a call to AWS Security Token Service (AWS STS). For a list of services that support temporary security credentials from AWS STS, see AWS services that work with IAM in the *IAM User Guide*.

Condition: If you're using temporary security credentials from AWS STS, you must include the security token.

Type: string

Required: Conditional

**X-Amz-Signature**

Specifies the hex-encoded signature that was calculated from the string to sign and the derived signing key.

Condition: Specify this parameter when you include authentication information in a query string instead of in the HTTP authorization header.

Type: string

Required: Conditional

**X-Amz-SignedHeaders**

Specifies all the HTTP headers that were included as part of the canonical request. For more information about specifying signed headers, see Create a signed AWS API request in the *IAM User Guide*.

Condition: Specify this parameter when you include authentication information in a query string instead of in the HTTP authorization header.

Type: string

Required: Conditional

# Common Error Types

This section lists common error types that this AWS service may return. Not all services return all error types listed here. For errors specific to an API action for this service, see the topic for that API action.

**AccessDeniedException**

You don't have permission to perform this action. Verify that your IAM policy includes the required permissions.

HTTP Status Code: 403

**ExpiredTokenException**

The security token included in the request has expired. Request a new security token and try again.

HTTP Status Code: 403

**IncompleteSignature**

The request signature doesn't conform to AWS standards. Verify that you're using valid AWS credentials and that your request is properly formatted. If you're using an SDK, ensure it's up to date.

HTTP Status Code: 403

**InternalFailure**

The request can't be processed right now because of an internal server issue. Try again later. If the problem persists, contact AWS Support.

HTTP Status Code: 500

**MalformedHttpRequestException**

The request body can't be processed. This typically happens when the request body can't be decompressed using the specified content encoding algorithm. Verify that the content encoding header matches the compression format used.

HTTP Status Code: 400

**NotAuthorized**

You don't have permissions to perform this action. Verify that your IAM policy includes the required permissions.

HTTP Status Code: 401

**OptInRequired**

Your AWS account needs a subscription for this service. Verify that you've enabled the service in your account.

HTTP Status Code: 403

**RequestAbortedException**

The request was aborted before a response could be returned. This typically happens when the client closes the connection.

HTTP Status Code: 400

**RequestEntityTooLargeException**

The request entity is too large. Reduce the size of the request body and try again.

HTTP Status Code: 413

**RequestTimeoutException**

The request timed out. The server didn't receive the complete request within the expected time frame. Try again.

HTTP Status Code: 408

**ServiceUnavailable**

The service is temporarily unavailable. Try again later.

HTTP Status Code: 503

**ThrottlingException**

Your request rate is too high. The AWS SDKs automatically retry requests that receive this exception. Reduce the frequency of requests.

HTTP Status Code: 400

**UnknownOperationException**

The action or operation isn't recognized. Verify that the action name is spelled correctly and that it's supported by the API version you're using.

HTTP Status Code: 404

**UnrecognizedClientException**

The X.509 certificate or AWS access key ID you provided doesn't exist in our records. Verify that you're using valid credentials and that they haven't expired.

HTTP Status Code: 403

**ValidationError**

The input doesn't meet the required format or constraints. Check that all required parameters are included and that values are valid.

HTTP Status Code: 400