Application Auto Scaling

API Reference

API Version 2016-02-06

Application Auto Scaling: API Reference

Table of Contents

Welcome 1
Actions 3
DeleteScalingPolicy 4
Request Syntax 4
Request Parameters 4
Response Elements 6
Errors 7
Examples 7
See Also 8
DeleteScheduledAction 9
Request Syntax 9
Request Parameters 9
Response Elements 11
Errors 11
See Also 12
DeregisterScalableTarget 13
Request Syntax 13
Request Parameters 13
Response Elements 15
Errors 15
Examples 16
See Also 16
DescribeScalableTargets 18
Request Syntax 18
Request Parameters 18
Response Syntax 21
Response Elements 21
Errors 21
Examples 22
See Also 23
DescribeScalingActivities 24
Request Syntax 24
Request Parameters 24
Response Syntax 27
Response Elements 27
Errors 27
Examples 28
See Also 30
DescribeScalingPolicies 32
Request Syntax 32
Request Parameters 32
Response Syntax 35
Response Elements 36
Errors 36
Examples 37
See Also 38
DescribeScheduledActions 40
Request Syntax 40
Request Parameters 40
Response Syntax 43
Response Elements 43
Errors 44
See Also 44
PutScalingPolicy 45

Request Syntax ... 45
Request Parameters ... 46
Response Syntax ... 49
Response Elements ... 49
Errors ... 49
Examples ... 50
See Also ... 53
PutScheduledAction ... 54
Request Syntax ... 54
Request Parameters ... 54
Response Elements ... 58
Errors ... 58
See Also ... 59
RegisterScalableTarget ... 60
Request Syntax ... 60
Request Parameters ... 60
Response Elements ... 64
Errors ... 64
Examples ... 64
See Also ... 65
Data Types ... 66
Alarm ... 67
Contents ... 67
See Also ... 67
CustomizedMetricSpecification ... 68
Contents ... 68
See Also ... 69
MetricDimension ... 70
Contents ... 70
See Also ... 70
PredefinedMetricSpecification ... 71
Contents ... 71
See Also ... 72
ScalableTarget ... 73
Contents ... 73
See Also ... 76
ScalableTargetAction ... 77
Contents ... 77
See Also ... 77
ScalingActivity ... 78
Contents ... 78
See Also ... 81
ScalingPolicy ... 82
Contents ... 82
See Also ... 85
ScheduledAction ... 86
Contents ... 86
See Also ... 89
StepAdjustment ... 91
Contents ... 91
See Also ... 92
StepScalingPolicyConfiguration ... 93
Contents ... 93
See Also ... 94
SuspendedState ... 95
Contents ... 95
See Also ... 95

TargetTrackingScalingPolicyConfiguration 96
Contents 96
See Also 97
Common Parameters 99
Common Errors 101
Logging API Calls with CloudTrail 103
Application Auto Scaling Information in CloudTrail 103
Understanding Application Auto Scaling Log File Entries 104
..... 104

Welcome

This is the *Application Auto Scaling API Reference*. With Application Auto Scaling, you can configure automatic scaling for the following resources:

- Amazon AppStream 2.0 fleets
- Amazon Aurora Replicas
- Amazon Comprehend document classification and entity recognizer endpoints
- Amazon DynamoDB tables and global secondary indexes throughput capacity
- Amazon ECS services
- Amazon ElastiCache for Redis clusters (replication groups)
- Amazon EMR clusters
- Amazon Keyspaces (for Apache Cassandra) tables
- AWS Lambda function provisioned concurrency
- Amazon Managed Streaming for Apache Kafka broker storage
- Amazon Neptune clusters
- Amazon SageMaker endpoint variants
- Spot Fleets (Amazon EC2)
- Custom resources provided by your own applications or services

API Summary

The Application Auto Scaling service API includes three key sets of actions:

- Register and manage scalable targets - Register AWS or custom resources as scalable targets (a resource that Application Auto Scaling can scale), set minimum and maximum capacity limits, and retrieve information on existing scalable targets.
- Configure and manage automatic scaling - Define scaling policies to dynamically scale your resources in response to CloudWatch alarms, schedule one-time or recurring scaling actions, and retrieve your recent scaling activity history.
- Suspend and resume scaling - Temporarily suspend and later resume automatic scaling by calling the RegisterScalableTarget API action for any Application Auto Scaling scalable target. You can suspend and resume (individually or in combination) scale-out activities that are triggered by a scaling policy, scale-in activities that are triggered by a scaling policy, and scheduled scaling.

The documentation for each action shows the Query API request syntax, the request parameters, and the response elements and provides links to language-specific SDK reference topics. For more information, see AWS SDKs.

To learn more about Application Auto Scaling, including information about granting IAM users required permissions for Application Auto Scaling actions, see the Application Auto Scaling User Guide.

API request rate

Application Auto Scaling uses the token bucket algorithm to implement API throttling. With this algorithm, your account has a bucket that holds a specific number of tokens. The number of tokens in the bucket represents your throttling limit at any given second. Application Auto Scaling throttles API requests based on a shared API bucket. For example, calls to the DescribeScalableTargets (p. 18) and DescribeScheduledActions (p. 40) API operations use tokens from the same bucket. Throttling means that Application Auto Scaling rejects a request because the request exceeds the service's limit for the number of requests per second. When a request is throttled, Application Auto Scaling returns a `RateExceeded` error. Application Auto Scaling does not guarantee a minimum request rate for APIs. For more information, see My Auto Scaling API calls are getting throttled. What can I do to avoid this? in the AWS Knowledge Center.

This document was last published on October 6, 2021.

Actions

The following actions are supported:

- DeleteScalingPolicy (p. 4)
- DeleteScheduledAction (p. 9)
- DeregisterScalableTarget (p. 13)
- DescribeScalableTargets (p. 18)
- DescribeScalingActivities (p. 24)
- DescribeScalingPolicies (p. 32)
- DescribeScheduledActions (p. 40)
- PutScalingPolicy (p. 45)
- PutScheduledAction (p. 54)
- RegisterScalableTarget (p. 60)

DeleteScalingPolicy

Deletes the specified scaling policy for an Application Auto Scaling scalable target.

Deleting a step scaling policy deletes the underlying alarm action, but does not delete the CloudWatch alarm associated with the scaling policy, even if it no longer has an associated action.

For more information, see Delete a step scaling policy and Delete a target tracking scaling policy in the *Application Auto Scaling User Guide*.

Request Syntax

```
{
   "PolicyName": "string",
   "ResourceId": "string",
   "ScalableDimension": "string",
   "ServiceNamespace": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 99).

The request accepts the following data in JSON format.

PolicyName (p. 4)

The name of the scaling policy.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1600.

Pattern: `[\u0020-\uD7FF\uE000-\uFFFD\uD800\uDC00-\uDBFF\uDFFF\r\n\t]*`

Required: Yes

ResourceId (p. 4)

The identifier of the resource associated with the scalable target. This string consists of the resource type and unique identifier.

- ECS service - The resource type is `service` and the unique identifier is the cluster name and service name. Example: `service/default/sample-webapp`.
- Spot Fleet - The resource type is `spot-fleet-request` and the unique identifier is the Spot Fleet request ID. Example: `spot-fleet-request/sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE`.
- EMR cluster - The resource type is `instancegroup` and the unique identifier is the cluster ID and instance group ID. Example: `instancegroup/j-2EEZNYKUA1NTV/ig-1791Y4E1L8YI0`.
- AppStream 2.0 fleet - The resource type is `fleet` and the unique identifier is the fleet name. Example: `fleet/sample-fleet`.
- DynamoDB table - The resource type is `table` and the unique identifier is the table name. Example: `table/my-table`.
- DynamoDB global secondary index - The resource type is `index` and the unique identifier is the index name. Example: `table/my-table/index/my-table-index`.

- Aurora DB cluster - The resource type is `cluster` and the unique identifier is the cluster name. Example: `cluster:my-db-cluster`.
- SageMaker endpoint variant - The resource type is `variant` and the unique identifier is the resource ID. Example: `endpoint/my-end-point/variant/KMeansClustering`.
- Custom resources are not supported with a resource type. This parameter must specify the `OutputValue` from the CloudFormation template stack used to access the resources. The unique identifier is defined by the service provider. More information is available in our GitHub repository.
- Amazon Comprehend document classification endpoint - The resource type and unique identifier are specified using the endpoint ARN. Example: `arn:aws:comprehend:us-west-2:123456789012:document-classifier-endpoint/EXAMPLE`.
- Amazon Comprehend entity recognizer endpoint - The resource type and unique identifier are specified using the endpoint ARN. Example: `arn:aws:comprehend:us-west-2:123456789012:entity-recognizer-endpoint/EXAMPLE`.
- Lambda provisioned concurrency - The resource type is `function` and the unique identifier is the function name with a function version or alias name suffix that is not `$LATEST`. Example: `function:my-function:prod` or `function:my-function:1`.
- Amazon Keyspaces table - The resource type is `table` and the unique identifier is the table name. Example: `keyspace/mykeyspace/table/mytable`.
- Amazon MSK cluster - The resource type and unique identifier are specified using the cluster ARN. Example: `arn:aws:kafka:us-east-1:123456789012:cluster/demo-cluster-1/6357e0b2-0e6a-4b86-a0b4-70df934c2e31-5`.
- Amazon ElastiCache replication group - The resource type is `replication-group` and the unique identifier is the replication group name. Example: `replication-group/mycluster`.
- Neptune cluster - The resource type is `cluster` and the unique identifier is the cluster name. Example: `cluster:mycluster`.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1600.

Pattern: `[\u0020-\uD7FF\uE000-\uFFFD\uD800\uDC00-\uDBFF\uDFFF\r\n\t]*`

Required: Yes

ScalableDimension (p. 4)

The scalable dimension. This string consists of the service namespace, resource type, and scaling property.

- `ecs:service:DesiredCount` - The desired task count of an ECS service.
- `elasticmapreduce:instancegroup:InstanceCount` - The instance count of an EMR Instance Group.
- `ec2:spot-fleet-request:TargetCapacity` - The target capacity of a Spot Fleet.
- `appstream:fleet:DesiredCapacity` - The desired capacity of an AppStream 2.0 fleet.
- `dynamodb:table:ReadCapacityUnits` - The provisioned read capacity for a DynamoDB table.
- `dynamodb:table:WriteCapacityUnits` - The provisioned write capacity for a DynamoDB table.
- `dynamodb:index:ReadCapacityUnits` - The provisioned read capacity for a DynamoDB global secondary index.
- `dynamodb:index:WriteCapacityUnits` - The provisioned write capacity for a DynamoDB global secondary index.
- `rds:cluster:ReadReplicaCount` - The count of Aurora Replicas in an Aurora DB cluster. Available for Aurora MySQL-compatible edition and Aurora PostgreSQL-compatible edition.
- `sagemaker:variant:DesiredInstanceCount` - The number of EC2 instances for an SageMaker model endpoint variant.

- `custom-resource:ResourceType:Property` - The scalable dimension for a custom resource provided by your own application or service.
- `comprehend:document-classifier-endpoint:DesiredInferenceUnits` - The number of inference units for an Amazon Comprehend document classification endpoint.
- `comprehend:entity-recognizer-endpoint:DesiredInferenceUnits` - The number of inference units for an Amazon Comprehend entity recognizer endpoint.
- `lambda:function:ProvisionedConcurrency` - The provisioned concurrency for a Lambda function.
- `cassandra:table:ReadCapacityUnits` - The provisioned read capacity for an Amazon Keyspaces table.
- `cassandra:table:WriteCapacityUnits` - The provisioned write capacity for an Amazon Keyspaces table.
- `kafka:broker-storage:VolumeSize` - The provisioned volume size (in GiB) for brokers in an Amazon MSK cluster.
- `elasticache:replication-group:NodeGroups` - The number of node groups for an Amazon ElastiCache replication group.
- `elasticache:replication-group:Replicas` - The number of replicas per node group for an Amazon ElastiCache replication group.
- `neptune:cluster:ReadReplicaCount` - The count of read replicas in an Amazon Neptune DB cluster.

Type: String

Valid Values: `ecs:service:DesiredCount | ec2:spot-fleet-request:TargetCapacity | elasticmapreduce:instancegroup:InstanceCount | appstream:fleet:DesiredCapacity | dynamodb:table:ReadCapacityUnits | dynamodb:table:WriteCapacityUnits | dynamodb:index:ReadCapacityUnits | dynamodb:index:WriteCapacityUnits | rds:cluster:ReadReplicaCount | sagemaker:variant:DesiredInstanceCount | custom-resource:ResourceType:Property | comprehend:document-classifier-endpoint:DesiredInferenceUnits | comprehend:entity-recognizer-endpoint:DesiredInferenceUnits | lambda:function:ProvisionedConcurrency | cassandra:table:ReadCapacityUnits | cassandra:table:WriteCapacityUnits | kafka:broker-storage:VolumeSize | elasticache:replication-group:NodeGroups | elasticache:replication-group:Replicas | neptune:cluster:ReadReplicaCount`

Required: Yes

ServiceNamespace (p. 4)

The namespace of the AWS service that provides the resource. For a resource provided by your own application or service, use `custom-resource` instead.

Type: String

Valid Values: `ecs | elasticmapreduce | ec2 | appstream | dynamodb | rds | sagemaker | custom-resource | comprehend | lambda | cassandra | kafka | elasticache | neptune`

Required: Yes

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 Errors (p. 101).

ConcurrentUpdateException

Concurrent updates caused an exception, for example, if you request an update to an Application Auto Scaling resource that already has a pending update.

HTTP Status Code: 400

InternalServiceException

The service encountered an internal error.

HTTP Status Code: 400

ObjectNotFoundException

The specified object could not be found. For any operation that depends on the existence of a scalable target, this exception is thrown if the scalable target with the specified service namespace, resource ID, and scalable dimension does not exist. For any operation that deletes or deregisters a resource, this exception is thrown if the resource cannot be found.

HTTP Status Code: 400

ValidationException

An exception was thrown for a validation issue. Review the available parameters for the API request.

HTTP Status Code: 400

Examples

If you plan to create requests manually, you must replace the Authorization header contents in the examples (`AUTHPARAMS`) with a signature. For more information, see Signature Version 4 Signing Process in the *Amazon Web Services General Reference*. If you plan to use the AWS CLI or one of the AWS SDKs, these tools sign the requests for you.

Example

The following example deletes a scaling policy for the Amazon ECS service `web-app` running in the `default` cluster.

Sample Request

```
POST / HTTP/1.1
Host: autoscaling.us-west-2.amazonaws.com
Accept-Encoding: identity
Content-Length: 152
X-Amz-Target: AnyScaleFrontendService.DeleteScalingPolicy
X-Amz-Date: 20190506T205712Z
User-Agent: aws-cli/1.10.23 Python/2.7.11 Darwin/15.4.0 botocore/1.4.8
Content-Type: application/x-amz-json-1.1
Authorization: AUTHPARAMS

{
    "PolicyName": "my-scale-out-policy",
    "ServiceNamespace": "ecs",
    "ScalableDimension": "ecs:service:DesiredCount",
    "ResourceId": "service/default/web-app"
}
```

See Also

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

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

DeleteScheduledAction

Deletes the specified scheduled action for an Application Auto Scaling scalable target.

For more information, see Delete a scheduled action in the *Application Auto Scaling User Guide*.

Request Syntax

```
{
   "ResourceId": "string",
   "ScalableDimension": "string",
   "ScheduledActionName": "string",
   "ServiceNamespace": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 99).

The request accepts the following data in JSON format.

ResourceId (p. 9)

The identifier of the resource associated with the scheduled action. This string consists of the resource type and unique identifier.

- ECS service - The resource type is `service` and the unique identifier is the cluster name and service name. Example: `service/default/sample-webapp`.
- Spot Fleet - The resource type is `spot-fleet-request` and the unique identifier is the Spot Fleet request ID. Example: `spot-fleet-request/sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE`.
- EMR cluster - The resource type is `instancegroup` and the unique identifier is the cluster ID and instance group ID. Example: `instancegroup/j-2EEZNYKUA1NTV/ig-1791Y4E1L8YI0`.
- AppStream 2.0 fleet - The resource type is `fleet` and the unique identifier is the fleet name. Example: `fleet/sample-fleet`.
- DynamoDB table - The resource type is `table` and the unique identifier is the table name. Example: `table/my-table`.
- DynamoDB global secondary index - The resource type is `index` and the unique identifier is the index name. Example: `table/my-table/index/my-table-index`.
- Aurora DB cluster - The resource type is `cluster` and the unique identifier is the cluster name. Example: `cluster:my-db-cluster`.
- SageMaker endpoint variant - The resource type is `variant` and the unique identifier is the resource ID. Example: `endpoint/my-end-point/variant/KMeansClustering`.
- Custom resources are not supported with a resource type. This parameter must specify the `OutputValue` from the CloudFormation template stack used to access the resources. The unique identifier is defined by the service provider. More information is available in our GitHub repository.
- Amazon Comprehend document classification endpoint - The resource type and unique identifier are specified using the endpoint ARN. Example: `arn:aws:comprehend:us-west-2:123456789012:document-classifier-endpoint/EXAMPLE`.
- Amazon Comprehend entity recognizer endpoint - The resource type and unique identifier are specified using the endpoint ARN. Example: `arn:aws:comprehend:us-west-2:123456789012:entity-recognizer-endpoint/EXAMPLE`.

- Lambda provisioned concurrency - The resource type is `function` and the unique identifier is the function name with a function version or alias name suffix that is not `$LATEST`. Example: `function:my-function:prod` or `function:my-function:1`.
- Amazon Keyspaces table - The resource type is `table` and the unique identifier is the table name. Example: `keyspace/mykeyspace/table/mytable`.
- Amazon MSK cluster - The resource type and unique identifier are specified using the cluster ARN. Example: `arn:aws:kafka:us-east-1:123456789012:cluster/demo-cluster-1/6357e0b2-0e6a-4b86-a0b4-70df934c2e31-5`.
- Amazon ElastiCache replication group - The resource type is `replication-group` and the unique identifier is the replication group name. Example: `replication-group/mycluster`.
- Neptune cluster - The resource type is `cluster` and the unique identifier is the cluster name. Example: `cluster:mycluster`.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1600.

Pattern: `[\u0020-\uD7FF\uE000-\uFFFD\uD800\uDC00-\uDBFF\uDFFF\r\n\t]*`

Required: Yes

ScalableDimension (p. 9)

The scalable dimension. This string consists of the service namespace, resource type, and scaling property.

- `ecs:service:DesiredCount` - The desired task count of an ECS service.
- `elasticmapreduce:instancegroup:InstanceCount` - The instance count of an EMR Instance Group.
- `ec2:spot-fleet-request:TargetCapacity` - The target capacity of a Spot Fleet.
- `appstream:fleet:DesiredCapacity` - The desired capacity of an AppStream 2.0 fleet.
- `dynamodb:table:ReadCapacityUnits` - The provisioned read capacity for a DynamoDB table.
- `dynamodb:table:WriteCapacityUnits` - The provisioned write capacity for a DynamoDB table.
- `dynamodb:index:ReadCapacityUnits` - The provisioned read capacity for a DynamoDB global secondary index.
- `dynamodb:index:WriteCapacityUnits` - The provisioned write capacity for a DynamoDB global secondary index.
- `rds:cluster:ReadReplicaCount` - The count of Aurora Replicas in an Aurora DB cluster. Available for Aurora MySQL-compatible edition and Aurora PostgreSQL-compatible edition.
- `sagemaker:variant:DesiredInstanceCount` - The number of EC2 instances for an SageMaker model endpoint variant.
- `custom-resource:ResourceType:Property` - The scalable dimension for a custom resource provided by your own application or service.
- `comprehend:document-classifier-endpoint:DesiredInferenceUnits` - The number of inference units for an Amazon Comprehend document classification endpoint.
- `comprehend:entity-recognizer-endpoint:DesiredInferenceUnits` - The number of inference units for an Amazon Comprehend entity recognizer endpoint.
- `lambda:function:ProvisionedConcurrency` - The provisioned concurrency for a Lambda function.
- `cassandra:table:ReadCapacityUnits` - The provisioned read capacity for an Amazon Keyspaces table.
- `cassandra:table:WriteCapacityUnits` - The provisioned write capacity for an Amazon Keyspaces table.

- `kafka:broker-storage:VolumeSize` - The provisioned volume size (in GiB) for brokers in an Amazon MSK cluster.
- `elasticache:replication-group:NodeGroups` - The number of node groups for an Amazon ElastiCache replication group.
- `elasticache:replication-group:Replicas` - The number of replicas per node group for an Amazon ElastiCache replication group.
- `neptune:cluster:ReadReplicaCount` - The count of read replicas in an Amazon Neptune DB cluster.

Type: String

Valid Values: `ecs:service:DesiredCount | ec2:spot-fleet-request:TargetCapacity | elasticmapreduce:instancegroup:InstanceCount | appstream:fleet:DesiredCapacity | dynamodb:table:ReadCapacityUnits | dynamodb:table:WriteCapacityUnits | dynamodb:index:ReadCapacityUnits | dynamodb:index:WriteCapacityUnits | rds:cluster:ReadReplicaCount | sagemaker:variant:DesiredInstanceCount | custom-resource:ResourceType:Property | comprehend:document-classifier-endpoint:DesiredInferenceUnits | comprehend:entity-recognizer-endpoint:DesiredInferenceUnits | lambda:function:ProvisionedConcurrency | cassandra:table:ReadCapacityUnits | cassandra:table:WriteCapacityUnits | kafka:broker-storage:VolumeSize | elasticache:replication-group:NodeGroups | elasticache:replication-group:Replicas | neptune:cluster:ReadReplicaCount`

Required: Yes

ScheduledActionName (p. 9)

The name of the scheduled action.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1600.

Pattern: `[\u0020-\uD7FF\uE000-\uFFFD\uD800\uDC00-\uDBFF\uDFFF\r\n\t]*`

Required: Yes

ServiceNamespace (p. 9)

The namespace of the AWS service that provides the resource. For a resource provided by your own application or service, use `custom-resource` instead.

Type: String

Valid Values: `ecs | elasticmapreduce | ec2 | appstream | dynamodb | rds | sagemaker | custom-resource | comprehend | lambda | cassandra | kafka | elasticache | neptune`

Required: Yes

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 Errors (p. 101).

ConcurrentUpdateException

Concurrent updates caused an exception, for example, if you request an update to an Application Auto Scaling resource that already has a pending update.

HTTP Status Code: 400

InternalServiceException

The service encountered an internal error.

HTTP Status Code: 400

ObjectNotFoundException

The specified object could not be found. For any operation that depends on the existence of a scalable target, this exception is thrown if the scalable target with the specified service namespace, resource ID, and scalable dimension does not exist. For any operation that deletes or deregisters a resource, this exception is thrown if the resource cannot be found.

HTTP Status Code: 400

ValidationException

An exception was thrown for a validation issue. Review the available parameters for the API request.

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

DeregisterScalableTarget

Deregisters an Application Auto Scaling scalable target when you have finished using it. To see which resources have been registered, use DescribeScalableTargets.

> **Note**
> Deregistering a scalable target deletes the scaling policies and the scheduled actions that are associated with it.

Request Syntax

```
{
   "ResourceId": "string",
   "ScalableDimension": "string",
   "ServiceNamespace": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 99).

The request accepts the following data in JSON format.

ResourceId (p. 13)

The identifier of the resource associated with the scalable target. This string consists of the resource type and unique identifier.

- ECS service - The resource type is `service` and the unique identifier is the cluster name and service name. Example: `service/default/sample-webapp`.
- Spot Fleet - The resource type is `spot-fleet-request` and the unique identifier is the Spot Fleet request ID. Example: `spot-fleet-request/sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE`.
- EMR cluster - The resource type is `instancegroup` and the unique identifier is the cluster ID and instance group ID. Example: `instancegroup/j-2EEZNYKUA1NTV/ig-1791Y4E1L8YI0`.
- AppStream 2.0 fleet - The resource type is `fleet` and the unique identifier is the fleet name. Example: `fleet/sample-fleet`.
- DynamoDB table - The resource type is `table` and the unique identifier is the table name. Example: `table/my-table`.
- DynamoDB global secondary index - The resource type is `index` and the unique identifier is the index name. Example: `table/my-table/index/my-table-index`.
- Aurora DB cluster - The resource type is `cluster` and the unique identifier is the cluster name. Example: `cluster:my-db-cluster`.
- SageMaker endpoint variant - The resource type is `variant` and the unique identifier is the resource ID. Example: `endpoint/my-end-point/variant/KMeansClustering`.
- Custom resources are not supported with a resource type. This parameter must specify the `OutputValue` from the CloudFormation template stack used to access the resources. The unique identifier is defined by the service provider. More information is available in our GitHub repository.
- Amazon Comprehend document classification endpoint - The resource type and unique identifier are specified using the endpoint ARN. Example: `arn:aws:comprehend:us-west-2:123456789012:document-classifier-endpoint/EXAMPLE`.

- Amazon Comprehend entity recognizer endpoint - The resource type and unique identifier are specified using the endpoint ARN. Example: `arn:aws:comprehend:us-west-2:123456789012:entity-recognizer-endpoint/EXAMPLE`.
- Lambda provisioned concurrency - The resource type is `function` and the unique identifier is the function name with a function version or alias name suffix that is not `$LATEST`. Example: `function:my-function:prod` or `function:my-function:1`.
- Amazon Keyspaces table - The resource type is `table` and the unique identifier is the table name. Example: `keyspace/mykeyspace/table/mytable`.
- Amazon MSK cluster - The resource type and unique identifier are specified using the cluster ARN. Example: `arn:aws:kafka:us-east-1:123456789012:cluster/demo-cluster-1/6357e0b2-0e6a-4b86-a0b4-70df934c2e31-5`.
- Amazon ElastiCache replication group - The resource type is `replication-group` and the unique identifier is the replication group name. Example: `replication-group/mycluster`.
- Neptune cluster - The resource type is `cluster` and the unique identifier is the cluster name. Example: `cluster:mycluster`.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1600.

Pattern: `[\u0020-\uD7FF\uE000-\uFFFD\uD800\uDC00-\uDBFF\uDFFF\r\n\t]*`

Required: Yes

ScalableDimension (p. 13)

The scalable dimension associated with the scalable target. This string consists of the service namespace, resource type, and scaling property.

- `ecs:service:DesiredCount` - The desired task count of an ECS service.
- `elasticmapreduce:instancegroup:InstanceCount` - The instance count of an EMR Instance Group.
- `ec2:spot-fleet-request:TargetCapacity` - The target capacity of a Spot Fleet.
- `appstream:fleet:DesiredCapacity` - The desired capacity of an AppStream 2.0 fleet.
- `dynamodb:table:ReadCapacityUnits` - The provisioned read capacity for a DynamoDB table.
- `dynamodb:table:WriteCapacityUnits` - The provisioned write capacity for a DynamoDB table.
- `dynamodb:index:ReadCapacityUnits` - The provisioned read capacity for a DynamoDB global secondary index.
- `dynamodb:index:WriteCapacityUnits` - The provisioned write capacity for a DynamoDB global secondary index.
- `rds:cluster:ReadReplicaCount` - The count of Aurora Replicas in an Aurora DB cluster. Available for Aurora MySQL-compatible edition and Aurora PostgreSQL-compatible edition.
- `sagemaker:variant:DesiredInstanceCount` - The number of EC2 instances for an SageMaker model endpoint variant.
- `custom-resource:ResourceType:Property` - The scalable dimension for a custom resource provided by your own application or service.
- `comprehend:document-classifier-endpoint:DesiredInferenceUnits` - The number of inference units for an Amazon Comprehend document classification endpoint.
- `comprehend:entity-recognizer-endpoint:DesiredInferenceUnits` - The number of inference units for an Amazon Comprehend entity recognizer endpoint.
- `lambda:function:ProvisionedConcurrency` - The provisioned concurrency for a Lambda function.

- `cassandra:table:ReadCapacityUnits` - The provisioned read capacity for an Amazon Keyspaces table.
- `cassandra:table:WriteCapacityUnits` - The provisioned write capacity for an Amazon Keyspaces table.
- `kafka:broker-storage:VolumeSize` - The provisioned volume size (in GiB) for brokers in an Amazon MSK cluster.
- `elasticache:replication-group:NodeGroups` - The number of node groups for an Amazon ElastiCache replication group.
- `elasticache:replication-group:Replicas` - The number of replicas per node group for an Amazon ElastiCache replication group.
- `neptune:cluster:ReadReplicaCount` - The count of read replicas in an Amazon Neptune DB cluster.

Type: String

Valid Values: `ecs:service:DesiredCount | ec2:spot-fleet-request:TargetCapacity | elasticmapreduce:instancegroup:InstanceCount | appstream:fleet:DesiredCapacity | dynamodb:table:ReadCapacityUnits | dynamodb:table:WriteCapacityUnits | dynamodb:index:ReadCapacityUnits | dynamodb:index:WriteCapacityUnits | rds:cluster:ReadReplicaCount | sagemaker:variant:DesiredInstanceCount | custom-resource:ResourceType:Property | comprehend:document-classifier-endpoint:DesiredInferenceUnits | comprehend:entity-recognizer-endpoint:DesiredInferenceUnits | lambda:function:ProvisionedConcurrency | cassandra:table:ReadCapacityUnits | cassandra:table:WriteCapacityUnits | kafka:broker-storage:VolumeSize | elasticache:replication-group:NodeGroups | elasticache:replication-group:Replicas | neptune:cluster:ReadReplicaCount`

Required: Yes

ServiceNamespace (p. 13)

The namespace of the AWS service that provides the resource. For a resource provided by your own application or service, use `custom-resource` instead.

Type: String

Valid Values: `ecs | elasticmapreduce | ec2 | appstream | dynamodb | rds | sagemaker | custom-resource | comprehend | lambda | cassandra | kafka | elasticache | neptune`

Required: Yes

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 Errors (p. 101).

ConcurrentUpdateException

Concurrent updates caused an exception, for example, if you request an update to an Application Auto Scaling resource that already has a pending update.

HTTP Status Code: 400

InternalServiceException

The service encountered an internal error.

HTTP Status Code: 400

ObjectNotFoundException

The specified object could not be found. For any operation that depends on the existence of a scalable target, this exception is thrown if the scalable target with the specified service namespace, resource ID, and scalable dimension does not exist. For any operation that deletes or deregisters a resource, this exception is thrown if the resource cannot be found.

HTTP Status Code: 400

ValidationException

An exception was thrown for a validation issue. Review the available parameters for the API request.

HTTP Status Code: 400

Examples

If you plan to create requests manually, you must replace the Authorization header contents in the examples (`AUTHPARAMS`) with a signature. For more information, see Signature Version 4 Signing Process in the *Amazon Web Services General Reference*. If you plan to use the AWS CLI or one of the AWS SDKs, these tools sign the requests for you.

Example

The following example deregisters a scalable target for an Amazon ECS service called `web-app` that is running in the `default` cluster.

Sample Request

```
POST / HTTP/1.1
Host: autoscaling.us-west-2.amazonaws.com
Accept-Encoding: identity
Content-Length: 117
X-Amz-Target: AnyScaleFrontendService.DeregisterScalableTarget
X-Amz-Date: 20190506T210150Z
User-Agent: aws-cli/1.10.23 Python/2.7.11 Darwin/15.4.0 botocore/1.4.8
Content-Type: application/x-amz-json-1.1
Authorization: AUTHPARAMS

{
    "ResourceId": "service/default/web-app",
    "ServiceNamespace": "ecs",
    "ScalableDimension": "ecs:service:DesiredCount"
}
```

See Also

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

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

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

DescribeScalableTargets

Gets information about the scalable targets in the specified namespace.

You can filter the results using `ResourceIds` and `ScalableDimension`.

Request Syntax

```
{
   "MaxResults": number,
   "NextToken": "string",
   "ResourceIds": [ "string" ],
   "ScalableDimension": "string",
   "ServiceNamespace": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 99).

The request accepts the following data in JSON format.

MaxResults (p. 18)

The maximum number of scalable targets. This value can be between 1 and 50. The default value is 50.

If this parameter is used, the operation returns up to `MaxResults` results at a time, along with a `NextToken` value. To get the next set of results, include the `NextToken` value in a subsequent call. If this parameter is not used, the operation returns up to 50 results and a `NextToken` value, if applicable.

Type: Integer

Required: No

NextToken (p. 18)

The token for the next set of results.

Type: String

Pattern: `[\u0020-\uD7FF\uE000-\uFFFD\uD800\uDC00-\uDBFF\uDFFF\r\n\t]*`

Required: No

ResourceIds (p. 18)

The identifier of the resource associated with the scalable target. This string consists of the resource type and unique identifier.

- ECS service - The resource type is `service` and the unique identifier is the cluster name and service name. Example: `service/default/sample-webapp`.
- Spot Fleet - The resource type is `spot-fleet-request` and the unique identifier is the Spot Fleet request ID. Example: `spot-fleet-request/sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE`.
- EMR cluster - The resource type is `instancegroup` and the unique identifier is the cluster ID and instance group ID. Example: `instancegroup/j-2EEZNYKUA1NTV/ig-1791Y4E1L8YI0`.

- AppStream 2.0 fleet - The resource type is `fleet` and the unique identifier is the fleet name. Example: `fleet/sample-fleet`.
- DynamoDB table - The resource type is `table` and the unique identifier is the table name. Example: `table/my-table`.
- DynamoDB global secondary index - The resource type is `index` and the unique identifier is the index name. Example: `table/my-table/index/my-table-index`.
- Aurora DB cluster - The resource type is `cluster` and the unique identifier is the cluster name. Example: `cluster:my-db-cluster`.
- SageMaker endpoint variant - The resource type is `variant` and the unique identifier is the resource ID. Example: `endpoint/my-end-point/variant/KMeansClustering`.
- Custom resources are not supported with a resource type. This parameter must specify the `OutputValue` from the CloudFormation template stack used to access the resources. The unique identifier is defined by the service provider. More information is available in our GitHub repository.
- Amazon Comprehend document classification endpoint - The resource type and unique identifier are specified using the endpoint ARN. Example: `arn:aws:comprehend:us-west-2:123456789012:document-classifier-endpoint/EXAMPLE`.
- Amazon Comprehend entity recognizer endpoint - The resource type and unique identifier are specified using the endpoint ARN. Example: `arn:aws:comprehend:us-west-2:123456789012:entity-recognizer-endpoint/EXAMPLE`.
- Lambda provisioned concurrency - The resource type is `function` and the unique identifier is the function name with a function version or alias name suffix that is not `$LATEST`. Example: `function:my-function:prod` or `function:my-function:1`.
- Amazon Keyspaces table - The resource type is `table` and the unique identifier is the table name. Example: `keyspace/mykeyspace/table/mytable`.
- Amazon MSK cluster - The resource type and unique identifier are specified using the cluster ARN. Example: `arn:aws:kafka:us-east-1:123456789012:cluster/demo-cluster-1/6357e0b2-0e6a-4b86-a0b4-70df934c2e31-5`.
- Amazon ElastiCache replication group - The resource type is `replication-group` and the unique identifier is the replication group name. Example: `replication-group/mycluster`.
- Neptune cluster - The resource type is `cluster` and the unique identifier is the cluster name. Example: `cluster:mycluster`.

Type: Array of strings

Array Members: Maximum number of 50 items.

Length Constraints: Minimum length of 1. Maximum length of 1600.

Pattern: `[\u0020-\uD7FF\uE000-\uFFFD\uD800\uDC00-\uDBFF\uDFFF\r\n\t]*`

Required: No

ScalableDimension (p. 18)

The scalable dimension associated with the scalable target. This string consists of the service namespace, resource type, and scaling property. If you specify a scalable dimension, you must also specify a resource ID.

- `ecs:service:DesiredCount` - The desired task count of an ECS service.
- `elasticmapreduce:instancegroup:InstanceCount` - The instance count of an EMR Instance Group.
- `ec2:spot-fleet-request:TargetCapacity` - The target capacity of a Spot Fleet.
- `appstream:fleet:DesiredCapacity` - The desired capacity of an AppStream 2.0 fleet.
- `dynamodb:table:ReadCapacityUnits` - The provisioned read capacity for a DynamoDB table.
- `dynamodb:table:WriteCapacityUnits` - The provisioned write capacity for a DynamoDB table.

- `dynamodb:index:ReadCapacityUnits` - The provisioned read capacity for a DynamoDB global secondary index.
- `dynamodb:index:WriteCapacityUnits` - The provisioned write capacity for a DynamoDB global secondary index.
- `rds:cluster:ReadReplicaCount` - The count of Aurora Replicas in an Aurora DB cluster. Available for Aurora MySQL-compatible edition and Aurora PostgreSQL-compatible edition.
- `sagemaker:variant:DesiredInstanceCount` - The number of EC2 instances for an SageMaker model endpoint variant.
- `custom-resource:ResourceType:Property` - The scalable dimension for a custom resource provided by your own application or service.
- `comprehend:document-classifier-endpoint:DesiredInferenceUnits` - The number of inference units for an Amazon Comprehend document classification endpoint.
- `comprehend:entity-recognizer-endpoint:DesiredInferenceUnits` - The number of inference units for an Amazon Comprehend entity recognizer endpoint.
- `lambda:function:ProvisionedConcurrency` - The provisioned concurrency for a Lambda function.
- `cassandra:table:ReadCapacityUnits` - The provisioned read capacity for an Amazon Keyspaces table.
- `cassandra:table:WriteCapacityUnits` - The provisioned write capacity for an Amazon Keyspaces table.
- `kafka:broker-storage:VolumeSize` - The provisioned volume size (in GiB) for brokers in an Amazon MSK cluster.
- `elasticache:replication-group:NodeGroups` - The number of node groups for an Amazon ElastiCache replication group.
- `elasticache:replication-group:Replicas` - The number of replicas per node group for an Amazon ElastiCache replication group.
- `neptune:cluster:ReadReplicaCount` - The count of read replicas in an Amazon Neptune DB cluster.

Type: String

Valid Values: `ecs:service:DesiredCount | ec2:spot-fleet-request:TargetCapacity | elasticmapreduce:instancegroup:InstanceCount | appstream:fleet:DesiredCapacity | dynamodb:table:ReadCapacityUnits | dynamodb:table:WriteCapacityUnits | dynamodb:index:ReadCapacityUnits | dynamodb:index:WriteCapacityUnits | rds:cluster:ReadReplicaCount | sagemaker:variant:DesiredInstanceCount | custom-resource:ResourceType:Property | comprehend:document-classifier-endpoint:DesiredInferenceUnits | comprehend:entity-recognizer-endpoint:DesiredInferenceUnits | lambda:function:ProvisionedConcurrency | cassandra:table:ReadCapacityUnits | cassandra:table:WriteCapacityUnits | kafka:broker-storage:VolumeSize | elasticache:replication-group:NodeGroups | elasticache:replication-group:Replicas | neptune:cluster:ReadReplicaCount`

Required: No

ServiceNamespace (p. 18)

The namespace of the AWS service that provides the resource. For a resource provided by your own application or service, use `custom-resource` instead.

Type: String

Valid Values: `ecs | elasticmapreduce | ec2 | appstream | dynamodb | rds | sagemaker | custom-resource | comprehend | lambda | cassandra | kafka | elasticache | neptune`

Required: Yes

Response Syntax

```
{
   "NextToken": "string",
   "ScalableTargets": [
      {
         "CreationTime": number,
         "MaxCapacity": number,
         "MinCapacity": number,
         "ResourceId": "string",
         "RoleARN": "string",
         "ScalableDimension": "string",
         "ServiceNamespace": "string",
         "SuspendedState": {
            "DynamicScalingInSuspended": boolean,
            "DynamicScalingOutSuspended": boolean,
            "ScheduledScalingSuspended": boolean
         }
      }
   ]
}
```

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 (p. 21)

The token required to get the next set of results. This value is `null` if there are no more results to return.

Type: String

Pattern: `[\u0020-\uD7FF\uE000-\uFFFD\uD800\uDC00-\uDBFF\uDFFF\r\n\t]*`

ScalableTargets (p. 21)

The scalable targets that match the request parameters.

Type: Array of ScalableTarget (p. 73) objects

Errors

For information about the errors that are common to all actions, see Common Errors (p. 101).

ConcurrentUpdateException

Concurrent updates caused an exception, for example, if you request an update to an Application Auto Scaling resource that already has a pending update.

HTTP Status Code: 400

InternalServiceException

The service encountered an internal error.

HTTP Status Code: 400

InvalidNextTokenException

The next token supplied was invalid.

HTTP Status Code: 400

ValidationException

An exception was thrown for a validation issue. Review the available parameters for the API request.

HTTP Status Code: 400

Examples

If you plan to create requests manually, you must replace the Authorization header contents in the examples (`AUTHPARAMS`) with a signature. For more information, see Signature Version 4 Signing Process in the *Amazon Web Services General Reference*. If you plan to use the AWS CLI or one of the AWS SDKs, these tools sign the requests for you.

Example

The following example describes the scalable targets for the `ecs` service namespace.

Sample Request

```
POST / HTTP/1.1
Host: autoscaling.us-west-2.amazonaws.com
Accept-Encoding: identity
Content-Length: 27
X-Amz-Target: AnyScaleFrontendService.DescribeScalableTargets
X-Amz-Date: 20190506T184921Z
User-Agent: aws-cli/1.10.23 Python/2.7.11 Darwin/15.4.0 botocore/1.4.8
Content-Type: application/x-amz-json-1.1
Authorization: AUTHPARAMS

{
    "ServiceNamespace": "ecs"
}
```

Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: 3f10dab0-13bb-11e6-a873-676fff004c09
Content-Type: application/x-amz-json-1.1
Content-Length: 272
Date: Fri, 06 May 2019 18:49:21 GMT

{
    "ScalableTargets": [
        {
            "CreationTime": 1462558906.199,
            "MaxCapacity": 10,
            "MinCapacity": 1,
            "ResourceId": "service/default/web-app",
            "RoleARN": "arn:aws:iam::012345678910:role/aws-service-role/ecs.application-
autoscaling.amazonaws.com/AWSServiceRoleForApplicationAutoScaling_ECSService",
            "ScalableDimension": "ecs:service:DesiredCount",
            "ServiceNamespace": "ecs",
            "SuspendedState": {
                "DynamicScalingInSuspended": false,
                "DynamicScalingOutSuspended": false,
                "ScheduledScalingSuspended": false
            }
        }
    ]
}
```

See Also

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

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

DescribeScalingActivities

Provides descriptive information about the scaling activities in the specified namespace from the previous six weeks.

You can filter the results using `ResourceId` and `ScalableDimension`.

Request Syntax

```
{
   "MaxResults": number,
   "NextToken": "string",
   "ResourceId": "string",
   "ScalableDimension": "string",
   "ServiceNamespace": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 99).

The request accepts the following data in JSON format.

MaxResults (p. 24)

The maximum number of scalable targets. This value can be between 1 and 50. The default value is 50.

If this parameter is used, the operation returns up to `MaxResults` results at a time, along with a `NextToken` value. To get the next set of results, include the `NextToken` value in a subsequent call. If this parameter is not used, the operation returns up to 50 results and a `NextToken` value, if applicable.

Type: Integer

Required: No

NextToken (p. 24)

The token for the next set of results.

Type: String

Pattern: `[\u0020-\uD7FF\uE000-\uFFFD\uD800\uDC00-\uDBFF\uDFFF\r\n\t]*`

Required: No

ResourceId (p. 24)

The identifier of the resource associated with the scaling activity. This string consists of the resource type and unique identifier.

- ECS service - The resource type is `service` and the unique identifier is the cluster name and service name. Example: `service/default/sample-webapp`.
- Spot Fleet - The resource type is `spot-fleet-request` and the unique identifier is the Spot Fleet request ID. Example: `spot-fleet-request/sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE`.

- EMR cluster - The resource type is `instancegroup` and the unique identifier is the cluster ID and instance group ID. Example: `instancegroup/j-2EEZNYKUA1NTV/ig-1791Y4E1L8YI0`.
- AppStream 2.0 fleet - The resource type is `fleet` and the unique identifier is the fleet name. Example: `fleet/sample-fleet`.
- DynamoDB table - The resource type is `table` and the unique identifier is the table name. Example: `table/my-table`.
- DynamoDB global secondary index - The resource type is `index` and the unique identifier is the index name. Example: `table/my-table/index/my-table-index`.
- Aurora DB cluster - The resource type is `cluster` and the unique identifier is the cluster name. Example: `cluster:my-db-cluster`.
- SageMaker endpoint variant - The resource type is `variant` and the unique identifier is the resource ID. Example: `endpoint/my-end-point/variant/KMeansClustering`.
- Custom resources are not supported with a resource type. This parameter must specify the `OutputValue` from the CloudFormation template stack used to access the resources. The unique identifier is defined by the service provider. More information is available in our GitHub repository.
- Amazon Comprehend document classification endpoint - The resource type and unique identifier are specified using the endpoint ARN. Example: `arn:aws:comprehend:us-west-2:123456789012:document-classifier-endpoint/EXAMPLE`.
- Amazon Comprehend entity recognizer endpoint - The resource type and unique identifier are specified using the endpoint ARN. Example: `arn:aws:comprehend:us-west-2:123456789012:entity-recognizer-endpoint/EXAMPLE`.
- Lambda provisioned concurrency - The resource type is `function` and the unique identifier is the function name with a function version or alias name suffix that is not `$LATEST`. Example: `function:my-function:prod` or `function:my-function:1`.
- Amazon Keyspaces table - The resource type is `table` and the unique identifier is the table name. Example: `keyspace/mykeyspace/table/mytable`.
- Amazon MSK cluster - The resource type and unique identifier are specified using the cluster ARN. Example: `arn:aws:kafka:us-east-1:123456789012:cluster/demo-cluster-1/6357e0b2-0e6a-4b86-a0b4-70df934c2e31-5`.
- Amazon ElastiCache replication group - The resource type is `replication-group` and the unique identifier is the replication group name. Example: `replication-group/mycluster`.
- Neptune cluster - The resource type is `cluster` and the unique identifier is the cluster name. Example: `cluster:mycluster`.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1600.

Pattern: `[\u0020-\uD7FF\uE000-\uFFFD\uD800\uDC00-\uDBFF\uDFFF\r\n\t]*`

Required: No

ScalableDimension (p. 24)

The scalable dimension. This string consists of the service namespace, resource type, and scaling property. If you specify a scalable dimension, you must also specify a resource ID.

- `ecs:service:DesiredCount` - The desired task count of an ECS service.
- `elasticmapreduce:instancegroup:InstanceCount` - The instance count of an EMR Instance Group.
- `ec2:spot-fleet-request:TargetCapacity` - The target capacity of a Spot Fleet.
- `appstream:fleet:DesiredCapacity` - The desired capacity of an AppStream 2.0 fleet.
- `dynamodb:table:ReadCapacityUnits` - The provisioned read capacity for a DynamoDB table.
- `dynamodb:table:WriteCapacityUnits` - The provisioned write capacity for a DynamoDB table.

- `dynamodb:index:ReadCapacityUnits` - The provisioned read capacity for a DynamoDB global secondary index.
- `dynamodb:index:WriteCapacityUnits` - The provisioned write capacity for a DynamoDB global secondary index.
- `rds:cluster:ReadReplicaCount` - The count of Aurora Replicas in an Aurora DB cluster. Available for Aurora MySQL-compatible edition and Aurora PostgreSQL-compatible edition.
- `sagemaker:variant:DesiredInstanceCount` - The number of EC2 instances for an SageMaker model endpoint variant.
- `custom-resource:ResourceType:Property` - The scalable dimension for a custom resource provided by your own application or service.
- `comprehend:document-classifier-endpoint:DesiredInferenceUnits` - The number of inference units for an Amazon Comprehend document classification endpoint.
- `comprehend:entity-recognizer-endpoint:DesiredInferenceUnits` - The number of inference units for an Amazon Comprehend entity recognizer endpoint.
- `lambda:function:ProvisionedConcurrency` - The provisioned concurrency for a Lambda function.
- `cassandra:table:ReadCapacityUnits` - The provisioned read capacity for an Amazon Keyspaces table.
- `cassandra:table:WriteCapacityUnits` - The provisioned write capacity for an Amazon Keyspaces table.
- `kafka:broker-storage:VolumeSize` - The provisioned volume size (in GiB) for brokers in an Amazon MSK cluster.
- `elasticache:replication-group:NodeGroups` - The number of node groups for an Amazon ElastiCache replication group.
- `elasticache:replication-group:Replicas` - The number of replicas per node group for an Amazon ElastiCache replication group.
- `neptune:cluster:ReadReplicaCount` - The count of read replicas in an Amazon Neptune DB cluster.

Type: String

Valid Values: `ecs:service:DesiredCount | ec2:spot-fleet-request:TargetCapacity | elasticmapreduce:instancegroup:InstanceCount | appstream:fleet:DesiredCapacity | dynamodb:table:ReadCapacityUnits | dynamodb:table:WriteCapacityUnits | dynamodb:index:ReadCapacityUnits | dynamodb:index:WriteCapacityUnits | rds:cluster:ReadReplicaCount | sagemaker:variant:DesiredInstanceCount | custom-resource:ResourceType:Property | comprehend:document-classifier-endpoint:DesiredInferenceUnits | comprehend:entity-recognizer-endpoint:DesiredInferenceUnits | lambda:function:ProvisionedConcurrency | cassandra:table:ReadCapacityUnits | cassandra:table:WriteCapacityUnits | kafka:broker-storage:VolumeSize | elasticache:replication-group:NodeGroups | elasticache:replication-group:Replicas | neptune:cluster:ReadReplicaCount`

Required: No

ServiceNamespace (p. 24)

The namespace of the AWS service that provides the resource. For a resource provided by your own application or service, use `custom-resource` instead.

Type: String

Valid Values: `ecs | elasticmapreduce | ec2 | appstream | dynamodb | rds | sagemaker | custom-resource | comprehend | lambda | cassandra | kafka | elasticache | neptune`

Required: Yes

Response Syntax

```
{
   "NextToken": "string",
   "ScalingActivities": [
      {
         "ActivityId": "string",
         "Cause": "string",
         "Description": "string",
         "Details": "string",
         "EndTime": number,
         "ResourceId": "string",
         "ScalableDimension": "string",
         "ServiceNamespace": "string",
         "StartTime": number,
         "StatusCode": "string",
         "StatusMessage": "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 (p. 27)

The token required to get the next set of results. This value is `null` if there are no more results to return.

Type: String

Pattern: `[\u0020-\uD7FF\uE000-\uFFFD\uD800\uDC00-\uDBFF\uDFFF\r\n\t]*`

ScalingActivities (p. 27)

A list of scaling activity objects.

Type: Array of ScalingActivity (p. 78) objects

Errors

For information about the errors that are common to all actions, see Common Errors (p. 101).

ConcurrentUpdateException

Concurrent updates caused an exception, for example, if you request an update to an Application Auto Scaling resource that already has a pending update.

HTTP Status Code: 400

InternalServiceException

The service encountered an internal error.

HTTP Status Code: 400

InvalidNextTokenException

The next token supplied was invalid.

HTTP Status Code: 400

ValidationException

An exception was thrown for a validation issue. Review the available parameters for the API request.

HTTP Status Code: 400

Examples

If you plan to create requests manually, you must replace the Authorization header contents in the examples (`AUTHPARAMS`) with a signature. For more information, see Signature Version 4 Signing Process in the *Amazon Web Services General Reference*. If you plan to use the AWS CLI or one of the AWS SDKs, these tools sign the requests for you.

Example of scaling activities for a scaling policy

The following example describes the scaling activities for an Amazon ECS service named `web-app` that is running in the `default` cluster. It shows the scaling activities for the scaling policy named `cpu75-target-tracking-scaling-policy`, which was triggered by the CloudWatch alarm named `TargetTracking-service/default/web-app-AlarmHigh-d4f0770c-b46e-434a-a60f-3b36d653feca`.

Sample Request

```
POST / HTTP/1.1
Host: autoscaling.us-west-2.amazonaws.com
Accept-Encoding: identity
Content-Length: 117
X-Amz-Target: AnyScaleFrontendService.DescribeScalingActivities
X-Amz-Date: 20190506T224112Z
User-Agent: aws-cli/1.10.26 Python/2.7.11 Darwin/15.4.0 botocore/1.4.8
Content-Type: application/x-amz-json-1.1
Authorization: AUTHPARAMS

{
  "ResourceId": "service/default/web-app",
  "ServiceNamespace": "ecs",
  "ScalableDimension": "ecs:service:DesiredCount"
}
```

Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: a2704130-13db-11e6-9fca-039a3edb2541
Content-Type: application/x-amz-json-1.1
Content-Length: 1784
Date: Fri, 06 May 2019 22:41:12 GMT

{
  "ScalingActivities": [
    {
      "ScalableDimension": "ecs:service:DesiredCount",
      "Description": "Setting desired count to 3.",
      "ResourceId": "service/default/web-app",
      "ActivityId": "4d759079-a31f-4d0c-8468-504c56e2eecf",
      "StartTime": 1462574194.658,
      "ServiceNamespace": "ecs",
      "EndTime": 1462574276.686,
      "Cause": "monitor alarm TargetTracking-service/default/web-app-AlarmHigh-d4f0770c-
b46e-434a-a60f-3b36d653feca in state ALARM triggered policy cpu75-target-tracking-scaling-
policy",
      "StatusMessage": "Successfully set desired count to 3. Change successfully fulfilled
 by ecs.",
      "StatusCode": "Successful"
    },
    {
      "ScalableDimension": "ecs:service:DesiredCount",
      "Description": "Setting desired count to 2.",
      "ResourceId": "service/default/web-app",
      "ActivityId": "90aff0eb-dd6a-443c-889b-b809e78061c1",
      "StartTime": 1462574254.223,
      "ServiceNamespace": "ecs",
      "EndTime": 1462574333.492,
      "Cause": "monitor alarm TargetTracking-service/default/web-app-AlarmHigh-d4f0770c-
b46e-434a-a60f-3b36d653feca in state ALARM triggered policy cpu75-target-tracking-scaling-
policy",
      "StatusMessage": "Successfully set desired count to 2. Change successfully fulfilled
 by ecs.",
      "StatusCode": "Successful"
    }
  ]
}
```

Example of scaling activities for scheduled actions

The following example describes the scaling activities for a DynamoDB table named `my-table`. It shows the scaling activities for scheduled actions named `my-first-scheduled-action` and `my-second-scheduled-action`.

Sample Request

```
POST / HTTP/1.1
Host: autoscaling.us-west-2.amazonaws.com
Accept-Encoding: identity
Content-Length: 117
X-Amz-Target: AnyScaleFrontendService.DescribeScalingActivities
X-Amz-Date: 20190526T110828Z
User-Agent: aws-cli/1.10.26 Python/2.7.11 Darwin/15.4.0 botocore/1.4.8
Content-Type: application/x-amz-json-1.1
Authorization: AUTHPARAMS

{
  "ResourceId": "table/my-table",
  "ServiceNamespace": "dynamodb",
  "ScalableDimension": "dynamodb:table:WriteCapacityUnits"
}
```

Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: a2704130-13db-11e6-9fca-039a3edb2541
Content-Type: application/x-amz-json-1.1
Content-Length: 1784
Date: Fri, 26 May 2019 11:08:28 GMT
```

```
{
  "ScalingActivities": [
    {
      "ScalableDimension": "dynamodb:table:WriteCapacityUnits",
      "Description": "Setting write capacity units to 10.",
      "ResourceId": "table/my-table",
      "ActivityId": "4d1308c0-bbcf-4514-a673-b0220ae38547",
      "StartTime": 1561574415.086,
      "ServiceNamespace": "dynamodb",
      "Cause": "maximum capacity was set to 10",
      "StatusMessage": "Successfully set write capacity units to 10. Waiting for change to
 be fulfilled by dynamodb.",
      "StatusCode": "InProgress"
    },
    {
      "ScalableDimension": "dynamodb:table:WriteCapacityUnits",
      "Description": "Setting min capacity to 5 and max capacity to 10",
      "ResourceId": "table/my-table",
      "ActivityId": "f2b7847b-721d-4e01-8ef0-0c8d3bacc1c7",
      "StartTime": 1561574414.644,
      "ServiceNamespace": "dynamodb",
      "Cause": "scheduled action name my-second-scheduled-action was triggered",
      "StatusMessage": "Successfully set min capacity to 5 and max capacity to 10",
      "StatusCode": "Successful"
    },
    {
      "ScalableDimension": "dynamodb:table:WriteCapacityUnits",
      "Description": "Setting write capacity units to 15.",
      "ResourceId": "table/my-table",
      "ActivityId": "d8ea4de6-9eaa-499f-b466-2cc5e681ba8b",
      "StartTime": 1561574108.904,
      "ServiceNamespace": "dynamodb",
      "EndTime": 1561574140.255,
      "Cause": "minimum capacity was set to 15",
      "StatusMessage": "Successfully set write capacity units to 15. Change successfully
 fulfilled by dynamodb.",
      "StatusCode": "Successful"
    },
    {
      "ScalableDimension": "dynamodb:table:WriteCapacityUnits",
      "Description": "Setting min capacity to 15 and max capacity to 20",
      "ResourceId": "table/my-table",
      "ActivityId": "3250fd06-6940-4e8e-bb1f-d494db7554d2",
      "StartTime": 1561574108.512,
      "ServiceNamespace": "dynamodb",
      "Cause": "scheduled action name my-first-scheduled-action was triggered",
      "StatusMessage": "Successfully set min capacity to 15 and max capacity to 20",
      "StatusCode": "Successful"
    }
  ]
}
```

See Also

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

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

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

DescribeScalingPolicies

Describes the Application Auto Scaling scaling policies for the specified service namespace.

You can filter the results using `ResourceId`, `ScalableDimension`, and `PolicyNames`.

For more information, see Target tracking scaling policies and Step scaling policies in the *Application Auto Scaling User Guide*.

Request Syntax

```
{
   "MaxResults": number,
   "NextToken": "string",
   "PolicyNames": [ "string" ],
   "ResourceId": "string",
   "ScalableDimension": "string",
   "ServiceNamespace": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 99).

The request accepts the following data in JSON format.

MaxResults (p. 32)

The maximum number of scalable targets. This value can be between 1 and 10. The default value is 10.

If this parameter is used, the operation returns up to `MaxResults` results at a time, along with a `NextToken` value. To get the next set of results, include the `NextToken` value in a subsequent call. If this parameter is not used, the operation returns up to 10 results and a `NextToken` value, if applicable.

Type: Integer

Required: No

NextToken (p. 32)

The token for the next set of results.

Type: String

Pattern: `[\u0020-\uD7FF\uE000-\uFFFD\uD800\uDC00-\uDBFF\uDFFF\r\n\t]*`

Required: No

PolicyNames (p. 32)

The names of the scaling policies to describe.

Type: Array of strings

Array Members: Maximum number of 50 items.

Length Constraints: Minimum length of 1. Maximum length of 1600.

Pattern: `[\u0020-\uD7FF\uE000-\uFFFD\uD800\uDC00-\uDBFF\uDFFF\r\n\t]*`

Required: No

ResourceId (p. 32)

The identifier of the resource associated with the scaling policy. This string consists of the resource type and unique identifier.

- ECS service - The resource type is `service` and the unique identifier is the cluster name and service name. Example: `service/default/sample-webapp`.
- Spot Fleet - The resource type is `spot-fleet-request` and the unique identifier is the Spot Fleet request ID. Example: `spot-fleet-request/sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE`.
- EMR cluster - The resource type is `instancegroup` and the unique identifier is the cluster ID and instance group ID. Example: `instancegroup/j-2EEZNYKUA1NTV/ig-1791Y4E1L8YI0`.
- AppStream 2.0 fleet - The resource type is `fleet` and the unique identifier is the fleet name. Example: `fleet/sample-fleet`.
- DynamoDB table - The resource type is `table` and the unique identifier is the table name. Example: `table/my-table`.
- DynamoDB global secondary index - The resource type is `index` and the unique identifier is the index name. Example: `table/my-table/index/my-table-index`.
- Aurora DB cluster - The resource type is `cluster` and the unique identifier is the cluster name. Example: `cluster:my-db-cluster`.
- SageMaker endpoint variant - The resource type is `variant` and the unique identifier is the resource ID. Example: `endpoint/my-end-point/variant/KMeansClustering`.
- Custom resources are not supported with a resource type. This parameter must specify the `OutputValue` from the CloudFormation template stack used to access the resources. The unique identifier is defined by the service provider. More information is available in our GitHub repository.
- Amazon Comprehend document classification endpoint - The resource type and unique identifier are specified using the endpoint ARN. Example: `arn:aws:comprehend:us-west-2:123456789012:document-classifier-endpoint/EXAMPLE`.
- Amazon Comprehend entity recognizer endpoint - The resource type and unique identifier are specified using the endpoint ARN. Example: `arn:aws:comprehend:us-west-2:123456789012:entity-recognizer-endpoint/EXAMPLE`.
- Lambda provisioned concurrency - The resource type is `function` and the unique identifier is the function name with a function version or alias name suffix that is not `$LATEST`. Example: `function:my-function:prod` or `function:my-function:1`.
- Amazon Keyspaces table - The resource type is `table` and the unique identifier is the table name. Example: `keyspace/mykeyspace/table/mytable`.
- Amazon MSK cluster - The resource type and unique identifier are specified using the cluster ARN. Example: `arn:aws:kafka:us-east-1:123456789012:cluster/demo-cluster-1/6357e0b2-0e6a-4b86-a0b4-70df934c2e31-5`.
- Amazon ElastiCache replication group - The resource type is `replication-group` and the unique identifier is the replication group name. Example: `replication-group/mycluster`.
- Neptune cluster - The resource type is `cluster` and the unique identifier is the cluster name. Example: `cluster:mycluster`.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1600.

Pattern: `[\u0020-\uD7FF\uE000-\uFFFD\uD800\uDC00-\uDBFF\uDFFF\r\n\t]*`

Required: No

ScalableDimension (p. 32)

The scalable dimension. This string consists of the service namespace, resource type, and scaling property. If you specify a scalable dimension, you must also specify a resource ID.

- `ecs:service:DesiredCount` - The desired task count of an ECS service.
- `elasticmapreduce:instancegroup:InstanceCount` - The instance count of an EMR Instance Group.
- `ec2:spot-fleet-request:TargetCapacity` - The target capacity of a Spot Fleet.
- `appstream:fleet:DesiredCapacity` - The desired capacity of an AppStream 2.0 fleet.
- `dynamodb:table:ReadCapacityUnits` - The provisioned read capacity for a DynamoDB table.
- `dynamodb:table:WriteCapacityUnits` - The provisioned write capacity for a DynamoDB table.
- `dynamodb:index:ReadCapacityUnits` - The provisioned read capacity for a DynamoDB global secondary index.
- `dynamodb:index:WriteCapacityUnits` - The provisioned write capacity for a DynamoDB global secondary index.
- `rds:cluster:ReadReplicaCount` - The count of Aurora Replicas in an Aurora DB cluster. Available for Aurora MySQL-compatible edition and Aurora PostgreSQL-compatible edition.
- `sagemaker:variant:DesiredInstanceCount` - The number of EC2 instances for an SageMaker model endpoint variant.
- `custom-resource:ResourceType:Property` - The scalable dimension for a custom resource provided by your own application or service.
- `comprehend:document-classifier-endpoint:DesiredInferenceUnits` - The number of inference units for an Amazon Comprehend document classification endpoint.
- `comprehend:entity-recognizer-endpoint:DesiredInferenceUnits` - The number of inference units for an Amazon Comprehend entity recognizer endpoint.
- `lambda:function:ProvisionedConcurrency` - The provisioned concurrency for a Lambda function.
- `cassandra:table:ReadCapacityUnits` - The provisioned read capacity for an Amazon Keyspaces table.
- `cassandra:table:WriteCapacityUnits` - The provisioned write capacity for an Amazon Keyspaces table.
- `kafka:broker-storage:VolumeSize` - The provisioned volume size (in GiB) for brokers in an Amazon MSK cluster.
- `elasticache:replication-group:NodeGroups` - The number of node groups for an Amazon ElastiCache replication group.
- `elasticache:replication-group:Replicas` - The number of replicas per node group for an Amazon ElastiCache replication group.
- `neptune:cluster:ReadReplicaCount` - The count of read replicas in an Amazon Neptune DB cluster.

Type: String

Valid Values: `ecs:service:DesiredCount | ec2:spot-fleet-request:TargetCapacity | elasticmapreduce:instancegroup:InstanceCount | appstream:fleet:DesiredCapacity | dynamodb:table:ReadCapacityUnits | dynamodb:table:WriteCapacityUnits | dynamodb:index:ReadCapacityUnits | dynamodb:index:WriteCapacityUnits | rds:cluster:ReadReplicaCount | sagemaker:variant:DesiredInstanceCount | custom-resource:ResourceType:Property | comprehend:document-classifier-endpoint:DesiredInferenceUnits | comprehend:entity-recognizer-`

endpoint:DesiredInferenceUnits | lambda:function:ProvisionedConcurrency | cassandra:table:ReadCapacityUnits | cassandra:table:WriteCapacityUnits | kafka:broker-storage:VolumeSize | elasticache:replication-group:NodeGroups | elasticache:replication-group:Replicas | neptune:cluster:ReadReplicaCount

Required: No

ServiceNamespace (p. 32)

The namespace of the AWS service that provides the resource. For a resource provided by your own application or service, use `custom-resource` instead.

Type: String

Valid Values: `ecs | elasticmapreduce | ec2 | appstream | dynamodb | rds | sagemaker | custom-resource | comprehend | lambda | cassandra | kafka | elasticache | neptune`

Required: Yes

Response Syntax

```
{
   "NextToken": "string",
   "ScalingPolicies": [
      {
         "Alarms": [
            {
               "AlarmARN": "string",
               "AlarmName": "string"
            }
         ],
         "CreationTime": number,
         "PolicyARN": "string",
         "PolicyName": "string",
         "PolicyType": "string",
         "ResourceId": "string",
         "ScalableDimension": "string",
         "ServiceNamespace": "string",
         "StepScalingPolicyConfiguration": {
            "AdjustmentType": "string",
            "Cooldown": number,
            "MetricAggregationType": "string",
            "MinAdjustmentMagnitude": number,
            "StepAdjustments": [
               {
                  "MetricIntervalLowerBound": number,
                  "MetricIntervalUpperBound": number,
                  "ScalingAdjustment": number
               }
            ]
         },
         "TargetTrackingScalingPolicyConfiguration": {
            "CustomizedMetricSpecification": {
               "Dimensions": [
                  {
                     "Name": "string",
                     "Value": "string"
                  }
               ],
               "MetricName": "string",
               "Namespace": "string",
               "Statistic": "string",
               "Unit": "string"
            },
            "DisableScaleIn": boolean,
            "PredefinedMetricSpecification": {
               "PredefinedMetricType": "string",
               "ResourceLabel": "string"
            },
            "ScaleInCooldown": number,
            "ScaleOutCooldown": number,
            "TargetValue": number
         }
      }
   ]
}
```

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 (p. 35)

The token required to get the next set of results. This value is `null` if there are no more results to return.

Type: String

Pattern: `[\u0020-\uD7FF\uE000-\uFFFD\uD800\uDC00-\uDBFF\uDFFF\r\n\t]*`

ScalingPolicies (p. 35)

Information about the scaling policies.

Type: Array of ScalingPolicy (p. 82) objects

Errors

For information about the errors that are common to all actions, see Common Errors (p. 101).

ConcurrentUpdateException

Concurrent updates caused an exception, for example, if you request an update to an Application Auto Scaling resource that already has a pending update.

HTTP Status Code: 400

FailedResourceAccessException

Failed access to resources caused an exception. This exception is thrown when Application Auto Scaling is unable to retrieve the alarms associated with a scaling policy due to a client error, for example, if the role ARN specified for a scalable target does not have permission to call the CloudWatch DescribeAlarms on your behalf.

HTTP Status Code: 400

InternalServiceException

The service encountered an internal error.

HTTP Status Code: 400

InvalidNextTokenException

The next token supplied was invalid.

HTTP Status Code: 400

ValidationException

An exception was thrown for a validation issue. Review the available parameters for the API request.

HTTP Status Code: 400

Examples

If you plan to create requests manually, you must replace the Authorization header contents in the examples (`AUTHPARAMS`) with a signature. For more information, see Signature Version 4 Signing Process in the *Amazon Web Services General Reference*. If you plan to use the AWS CLI or one of the AWS SDKs, these tools sign the requests for you.

Example

The following example describes the scaling policies for the `ecs` service namespace.

Sample Request

```
POST / HTTP/1.1
Host: autoscaling.us-west-2.amazonaws.com
Accept-Encoding: identity
Content-Length: 27
X-Amz-Target: AnyScaleFrontendService.DescribeScalingPolicies
X-Amz-Date: 20190506T194435Z
User-Agent: aws-cli/1.10.23 Python/2.7.11 Darwin/15.4.0 botocore/1.4.8
Content-Type: application/x-amz-json-1.1
Authorization: AUTHPARAMS

{
    "ServiceNamespace": "ecs"
}
```

Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: f662c515-13c2-11e6-add4-41b78770ca43
Content-Type: application/x-amz-json-1.1
Content-Length: 1393
Date: Fri, 06 May 2019 19:44:35 GMT

{
    "ScalingPolicies": [
        {
            "Alarms": [
                {
                    "AlarmARN": "arn:aws:cloudwatch:us-west-2:012345678910:alarm:step-
scaling-alarmhigh-ecs:service/default/web-app",
                    "AlarmName": "Step-Scaling-AlarmHigh-ECS:service/default/web-app"
                }
            ],
            "CreationTime": 1462561899.23,
            "PolicyARN": "arn:aws:autoscaling:us-
west-2:012345678910:scalingPolicy:ac542982-cbeb-4294-891c-a5a941dfa787:resource/ecs/
service/default/web-app:policyName/my-scale-out-policy",
            "PolicyName": "my-scale-out-policy",
            "PolicyType": "StepScaling",
            "ResourceId": "service/default/web-app",
            "ScalableDimension": "ecs:service:DesiredCount",
            "ServiceNamespace": "ecs",
            "StepScalingPolicyConfiguration": {
                "AdjustmentType": "PercentChangeInCapacity",
                "Cooldown": 60,
                "MetricAggregationType": "Average",
                "StepAdjustments": [
                    {
                        "MetricIntervalLowerBound": 0,
                        "ScalingAdjustment": 200
                    }
                ]
            }
        },
        {
            "Alarms": [
                {
                    "AlarmARN": "arn:aws:cloudwatch:us-west-2:012345678910:alarm:step-
scaling-alarmlow-ecs:service/default/web-app",
                    "AlarmName": "Step-Scaling-AlarmLow-ECS:service/default/web-app"
                }
            ],
            "CreationTime": 1462562575.099,
            "PolicyARN": "arn:aws:autoscaling:us-
west-2:012345678910:scalingPolicy:6d8972f3-efc8-437c-92d1-6270f29a66e7:resource/ecs/
service/default/web-app:policyName/my-scale-in-policy",
            "PolicyName": "my-scale-in-policy",
            "PolicyType": "StepScaling",
            "ResourceId": "service/default/web-app",
            "ScalableDimension": "ecs:service:DesiredCount",
            "ServiceNamespace": "ecs",
            "StepScalingPolicyConfiguration": {
                "AdjustmentType": "PercentChangeInCapacity",
                "Cooldown": 120,
                "MetricAggregationType": "Average",
                "MinAdjustmentMagnitude": 1,
                "StepAdjustments": [
                    {
                        "MetricIntervalLowerBound": -15,
                        "MetricIntervalUpperBound": 0
                        "ScalingAdjustment": -25,
                    },
                    {
                        "MetricIntervalUpperBound": -15,
                        "ScalingAdjustment": -50
                    }
                ]
            }
        }
    ]
}
```

See Also

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

- AWS Command Line Interface

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

DescribeScheduledActions

Describes the Application Auto Scaling scheduled actions for the specified service namespace.

You can filter the results using the `ResourceId`, `ScalableDimension`, and `ScheduledActionNames` parameters.

For more information, see Scheduled scaling and Managing scheduled scaling in the *Application Auto Scaling User Guide*.

Request Syntax

```
{
   "MaxResults": number,
   "NextToken": "string",
   "ResourceId": "string",
   "ScalableDimension": "string",
   "ScheduledActionNames": [ "string" ],
   "ServiceNamespace": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 99).

The request accepts the following data in JSON format.

MaxResults (p. 40)

The maximum number of scheduled action results. This value can be between 1 and 50. The default value is 50.

If this parameter is used, the operation returns up to `MaxResults` results at a time, along with a `NextToken` value. To get the next set of results, include the `NextToken` value in a subsequent call. If this parameter is not used, the operation returns up to 50 results and a `NextToken` value, if applicable.

Type: Integer

Required: No

NextToken (p. 40)

The token for the next set of results.

Type: String

Pattern: `[\u0020-\uD7FF\uE000-\uFFFD\uD800\uDC00-\uDBFF\uDFFF\r\n\t]*`

Required: No

ResourceId (p. 40)

The identifier of the resource associated with the scheduled action. This string consists of the resource type and unique identifier.

- ECS service - The resource type is `service` and the unique identifier is the cluster name and service name. Example: `service/default/sample-webapp`.

- Spot Fleet - The resource type is `spot-fleet-request` and the unique identifier is the Spot Fleet request ID. Example: `spot-fleet-request/sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE`.
- EMR cluster - The resource type is `instancegroup` and the unique identifier is the cluster ID and instance group ID. Example: `instancegroup/j-2EEZNYKUA1NTV/ig-1791Y4E1L8YI0`.
- AppStream 2.0 fleet - The resource type is `fleet` and the unique identifier is the fleet name. Example: `fleet/sample-fleet`.
- DynamoDB table - The resource type is `table` and the unique identifier is the table name. Example: `table/my-table`.
- DynamoDB global secondary index - The resource type is `index` and the unique identifier is the index name. Example: `table/my-table/index/my-table-index`.
- Aurora DB cluster - The resource type is `cluster` and the unique identifier is the cluster name. Example: `cluster:my-db-cluster`.
- SageMaker endpoint variant - The resource type is `variant` and the unique identifier is the resource ID. Example: `endpoint/my-end-point/variant/KMeansClustering`.
- Custom resources are not supported with a resource type. This parameter must specify the `OutputValue` from the CloudFormation template stack used to access the resources. The unique identifier is defined by the service provider. More information is available in our GitHub repository.
- Amazon Comprehend document classification endpoint - The resource type and unique identifier are specified using the endpoint ARN. Example: `arn:aws:comprehend:us-west-2:123456789012:document-classifier-endpoint/EXAMPLE`.
- Amazon Comprehend entity recognizer endpoint - The resource type and unique identifier are specified using the endpoint ARN. Example: `arn:aws:comprehend:us-west-2:123456789012:entity-recognizer-endpoint/EXAMPLE`.
- Lambda provisioned concurrency - The resource type is `function` and the unique identifier is the function name with a function version or alias name suffix that is not `$LATEST`. Example: `function:my-function:prod` or `function:my-function:1`.
- Amazon Keyspaces table - The resource type is `table` and the unique identifier is the table name. Example: `keyspace/mykeyspace/table/mytable`.
- Amazon MSK cluster - The resource type and unique identifier are specified using the cluster ARN. Example: `arn:aws:kafka:us-east-1:123456789012:cluster/demo-cluster-1/6357e0b2-0e6a-4b86-a0b4-70df934c2e31-5`.
- Amazon ElastiCache replication group - The resource type is `replication-group` and the unique identifier is the replication group name. Example: `replication-group/mycluster`.
- Neptune cluster - The resource type is `cluster` and the unique identifier is the cluster name. Example: `cluster:mycluster`.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1600.

Pattern: `[\u0020-\uD7FF\uE000-\uFFFD\uD800\uDC00-\uDBFF\uDFFF\r\n\t]*`

Required: No

ScalableDimension (p. 40)

The scalable dimension. This string consists of the service namespace, resource type, and scaling property. If you specify a scalable dimension, you must also specify a resource ID.

- `ecs:service:DesiredCount` - The desired task count of an ECS service.
- `elasticmapreduce:instancegroup:InstanceCount` - The instance count of an EMR Instance Group.
- `ec2:spot-fleet-request:TargetCapacity` - The target capacity of a Spot Fleet.
- `appstream:fleet:DesiredCapacity` - The desired capacity of an AppStream 2.0 fleet.

- `dynamodb:table:ReadCapacityUnits` - The provisioned read capacity for a DynamoDB table.
- `dynamodb:table:WriteCapacityUnits` - The provisioned write capacity for a DynamoDB table.
- `dynamodb:index:ReadCapacityUnits` - The provisioned read capacity for a DynamoDB global secondary index.
- `dynamodb:index:WriteCapacityUnits` - The provisioned write capacity for a DynamoDB global secondary index.
- `rds:cluster:ReadReplicaCount` - The count of Aurora Replicas in an Aurora DB cluster. Available for Aurora MySQL-compatible edition and Aurora PostgreSQL-compatible edition.
- `sagemaker:variant:DesiredInstanceCount` - The number of EC2 instances for an SageMaker model endpoint variant.
- `custom-resource:ResourceType:Property` - The scalable dimension for a custom resource provided by your own application or service.
- `comprehend:document-classifier-endpoint:DesiredInferenceUnits` - The number of inference units for an Amazon Comprehend document classification endpoint.
- `comprehend:entity-recognizer-endpoint:DesiredInferenceUnits` - The number of inference units for an Amazon Comprehend entity recognizer endpoint.
- `lambda:function:ProvisionedConcurrency` - The provisioned concurrency for a Lambda function.
- `cassandra:table:ReadCapacityUnits` - The provisioned read capacity for an Amazon Keyspaces table.
- `cassandra:table:WriteCapacityUnits` - The provisioned write capacity for an Amazon Keyspaces table.
- `kafka:broker-storage:VolumeSize` - The provisioned volume size (in GiB) for brokers in an Amazon MSK cluster.
- `elasticache:replication-group:NodeGroups` - The number of node groups for an Amazon ElastiCache replication group.
- `elasticache:replication-group:Replicas` - The number of replicas per node group for an Amazon ElastiCache replication group.
- `neptune:cluster:ReadReplicaCount` - The count of read replicas in an Amazon Neptune DB cluster.

Type: String

Valid Values: `ecs:service:DesiredCount | ec2:spot-fleet-request:TargetCapacity | elasticmapreduce:instancegroup:InstanceCount | appstream:fleet:DesiredCapacity | dynamodb:table:ReadCapacityUnits | dynamodb:table:WriteCapacityUnits | dynamodb:index:ReadCapacityUnits | dynamodb:index:WriteCapacityUnits | rds:cluster:ReadReplicaCount | sagemaker:variant:DesiredInstanceCount | custom-resource:ResourceType:Property | comprehend:document-classifier-endpoint:DesiredInferenceUnits | comprehend:entity-recognizer-endpoint:DesiredInferenceUnits | lambda:function:ProvisionedConcurrency | cassandra:table:ReadCapacityUnits | cassandra:table:WriteCapacityUnits | kafka:broker-storage:VolumeSize | elasticache:replication-group:NodeGroups | elasticache:replication-group:Replicas | neptune:cluster:ReadReplicaCount`

Required: No

ScheduledActionNames (p. 40)

The names of the scheduled actions to describe.

Type: Array of strings

Array Members: Maximum number of 50 items.

Length Constraints: Minimum length of 1. Maximum length of 1600.

Pattern: `[\u0020-\uD7FF\uE000-\uFFFD\uD800\uDC00-\uDBFF\uDFFF\r\n\t]*`

Required: No

ServiceNamespace (p. 40)

The namespace of the AWS service that provides the resource. For a resource provided by your own application or service, use `custom-resource` instead.

Type: String

Valid Values: `ecs | elasticmapreduce | ec2 | appstream | dynamodb | rds | sagemaker | custom-resource | comprehend | lambda | cassandra | kafka | elasticache | neptune`

Required: Yes

Response Syntax

```
{
   "NextToken": "string",
   "ScheduledActions": [
      {
         "CreationTime": number,
         "EndTime": number,
         "ResourceId": "string",
         "ScalableDimension": "string",
         "ScalableTargetAction": {
            "MaxCapacity": number,
            "MinCapacity": number
         },
         "Schedule": "string",
         "ScheduledActionARN": "string",
         "ScheduledActionName": "string",
         "ServiceNamespace": "string",
         "StartTime": number,
         "Timezone": "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 (p. 43)

The token required to get the next set of results. This value is `null` if there are no more results to return.

Type: String

Pattern: `[\u0020-\uD7FF\uE000-\uFFFD\uD800\uDC00-\uDBFF\uDFFF\r\n\t]*`

ScheduledActions (p. 43)

Information about the scheduled actions.

Type: Array of ScheduledAction (p. 86) objects

Errors

For information about the errors that are common to all actions, see Common Errors (p. 101).

ConcurrentUpdateException

Concurrent updates caused an exception, for example, if you request an update to an Application Auto Scaling resource that already has a pending update.

HTTP Status Code: 400

InternalServiceException

The service encountered an internal error.

HTTP Status Code: 400

InvalidNextTokenException

The next token supplied was invalid.

HTTP Status Code: 400

ValidationException

An exception was thrown for a validation issue. Review the available parameters for the API request.

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

PutScalingPolicy

Creates or updates a scaling policy for an Application Auto Scaling scalable target.

Each scalable target is identified by a service namespace, resource ID, and scalable dimension. A scaling policy applies to the scalable target identified by those three attributes. You cannot create a scaling policy until you have registered the resource as a scalable target.

Multiple scaling policies can be in force at the same time for the same scalable target. You can have one or more target tracking scaling policies, one or more step scaling policies, or both. However, there is a chance that multiple policies could conflict, instructing the scalable target to scale out or in at the same time. Application Auto Scaling gives precedence to the policy that provides the largest capacity for both scale out and scale in. For example, if one policy increases capacity by 3, another policy increases capacity by 200 percent, and the current capacity is 10, Application Auto Scaling uses the policy with the highest calculated capacity (200% of 10 = 20) and scales out to 30.

We recommend caution, however, when using target tracking scaling policies with step scaling policies because conflicts between these policies can cause undesirable behavior. For example, if the step scaling policy initiates a scale-in activity before the target tracking policy is ready to scale in, the scale-in activity will not be blocked. After the scale-in activity completes, the target tracking policy could instruct the scalable target to scale out again.

For more information, see Target tracking scaling policies and Step scaling policies in the *Application Auto Scaling User Guide*.

> **Note**
> If a scalable target is deregistered, the scalable target is no longer available to execute scaling policies. Any scaling policies that were specified for the scalable target are deleted.

Request Syntax

```
{
   "PolicyName": "string",
   "PolicyType": "string",
   "ResourceId": "string",
   "ScalableDimension": "string",
   "ServiceNamespace": "string",
   "StepScalingPolicyConfiguration": {
      "AdjustmentType": "string",
      "Cooldown": number,
      "MetricAggregationType": "string",
      "MinAdjustmentMagnitude": number,
      "StepAdjustments": [
         {
            "MetricIntervalLowerBound": number,
            "MetricIntervalUpperBound": number,
            "ScalingAdjustment": number
         }
      ]
   },
   "TargetTrackingScalingPolicyConfiguration": {
      "CustomizedMetricSpecification": {
         "Dimensions": [
            {
               "Name": "string",
               "Value": "string"
            }
         ],
         "MetricName": "string",
         "Namespace": "string",
         "Statistic": "string",
         "Unit": "string"
      },
      "DisableScaleIn": boolean,
      "PredefinedMetricSpecification": {
         "PredefinedMetricType": "string",
         "ResourceLabel": "string"
      },
      "ScaleInCooldown": number,
      "ScaleOutCooldown": number,
      "TargetValue": number
   }
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 99).

The request accepts the following data in JSON format.

PolicyName (p. 45)

The name of the scaling policy.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

Pattern: `\p{Print}+`

Required: Yes

PolicyType (p. 45)

The policy type. This parameter is required if you are creating a scaling policy.

The following policy types are supported:

`TargetTrackingScaling`—Not supported for Amazon EMR

`StepScaling`—Not supported for DynamoDB, Amazon Comprehend, Lambda, Amazon Keyspaces, Amazon MSK, Amazon ElastiCache, or Neptune.

For more information, see Target tracking scaling policies and Step scaling policies in the *Application Auto Scaling User Guide*.

Type: String

Valid Values: `StepScaling | TargetTrackingScaling`

Required: No

ResourceId (p. 45)

The identifier of the resource associated with the scaling policy. This string consists of the resource type and unique identifier.

- ECS service - The resource type is `service` and the unique identifier is the cluster name and service name. Example: `service/default/sample-webapp`.
- Spot Fleet - The resource type is `spot-fleet-request` and the unique identifier is the Spot Fleet request ID. Example: `spot-fleet-request/sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE`.

- EMR cluster - The resource type is `instancegroup` and the unique identifier is the cluster ID and instance group ID. Example: `instancegroup/j-2EEZNYKUA1NTV/ig-1791Y4E1L8YI0`.
- AppStream 2.0 fleet - The resource type is `fleet` and the unique identifier is the fleet name. Example: `fleet/sample-fleet`.
- DynamoDB table - The resource type is `table` and the unique identifier is the table name. Example: `table/my-table`.
- DynamoDB global secondary index - The resource type is `index` and the unique identifier is the index name. Example: `table/my-table/index/my-table-index`.
- Aurora DB cluster - The resource type is `cluster` and the unique identifier is the cluster name. Example: `cluster:my-db-cluster`.
- SageMaker endpoint variant - The resource type is `variant` and the unique identifier is the resource ID. Example: `endpoint/my-end-point/variant/KMeansClustering`.
- Custom resources are not supported with a resource type. This parameter must specify the `OutputValue` from the CloudFormation template stack used to access the resources. The unique identifier is defined by the service provider. More information is available in our GitHub repository.
- Amazon Comprehend document classification endpoint - The resource type and unique identifier are specified using the endpoint ARN. Example: `arn:aws:comprehend:us-west-2:123456789012:document-classifier-endpoint/EXAMPLE`.
- Amazon Comprehend entity recognizer endpoint - The resource type and unique identifier are specified using the endpoint ARN. Example: `arn:aws:comprehend:us-west-2:123456789012:entity-recognizer-endpoint/EXAMPLE`.
- Lambda provisioned concurrency - The resource type is `function` and the unique identifier is the function name with a function version or alias name suffix that is not `$LATEST`. Example: `function:my-function:prod` or `function:my-function:1`.
- Amazon Keyspaces table - The resource type is `table` and the unique identifier is the table name. Example: `keyspace/mykeyspace/table/mytable`.
- Amazon MSK cluster - The resource type and unique identifier are specified using the cluster ARN. Example: `arn:aws:kafka:us-east-1:123456789012:cluster/demo-cluster-1/6357e0b2-0e6a-4b86-a0b4-70df934c2e31-5`.
- Amazon ElastiCache replication group - The resource type is `replication-group` and the unique identifier is the replication group name. Example: `replication-group/mycluster`.
- Neptune cluster - The resource type is `cluster` and the unique identifier is the cluster name. Example: `cluster:mycluster`.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1600.

Pattern: `[\u0020-\uD7FF\uE000-\uFFFD\uD800\uDC00-\uDBFF\uDFFF\r\n\t]*`

Required: Yes

ScalableDimension (p. 45)

The scalable dimension. This string consists of the service namespace, resource type, and scaling property.

- `ecs:service:DesiredCount` - The desired task count of an ECS service.
- `elasticmapreduce:instancegroup:InstanceCount` - The instance count of an EMR Instance Group.
- `ec2:spot-fleet-request:TargetCapacity` - The target capacity of a Spot Fleet.
- `appstream:fleet:DesiredCapacity` - The desired capacity of an AppStream 2.0 fleet.
- `dynamodb:table:ReadCapacityUnits` - The provisioned read capacity for a DynamoDB table.
- `dynamodb:table:WriteCapacityUnits` - The provisioned write capacity for a DynamoDB table.

- `dynamodb:index:ReadCapacityUnits` - The provisioned read capacity for a DynamoDB global secondary index.
- `dynamodb:index:WriteCapacityUnits` - The provisioned write capacity for a DynamoDB global secondary index.
- `rds:cluster:ReadReplicaCount` - The count of Aurora Replicas in an Aurora DB cluster. Available for Aurora MySQL-compatible edition and Aurora PostgreSQL-compatible edition.
- `sagemaker:variant:DesiredInstanceCount` - The number of EC2 instances for an SageMaker model endpoint variant.
- `custom-resource:ResourceType:Property` - The scalable dimension for a custom resource provided by your own application or service.
- `comprehend:document-classifier-endpoint:DesiredInferenceUnits` - The number of inference units for an Amazon Comprehend document classification endpoint.
- `comprehend:entity-recognizer-endpoint:DesiredInferenceUnits` - The number of inference units for an Amazon Comprehend entity recognizer endpoint.
- `lambda:function:ProvisionedConcurrency` - The provisioned concurrency for a Lambda function.
- `cassandra:table:ReadCapacityUnits` - The provisioned read capacity for an Amazon Keyspaces table.
- `cassandra:table:WriteCapacityUnits` - The provisioned write capacity for an Amazon Keyspaces table.
- `kafka:broker-storage:VolumeSize` - The provisioned volume size (in GiB) for brokers in an Amazon MSK cluster.
- `elasticache:replication-group:NodeGroups` - The number of node groups for an Amazon ElastiCache replication group.
- `elasticache:replication-group:Replicas` - The number of replicas per node group for an Amazon ElastiCache replication group.
- `neptune:cluster:ReadReplicaCount` - The count of read replicas in an Amazon Neptune DB cluster.

Type: String

Valid Values: `ecs:service:DesiredCount | ec2:spot-fleet-request:TargetCapacity | elasticmapreduce:instancegroup:InstanceCount | appstream:fleet:DesiredCapacity | dynamodb:table:ReadCapacityUnits | dynamodb:table:WriteCapacityUnits | dynamodb:index:ReadCapacityUnits | dynamodb:index:WriteCapacityUnits | rds:cluster:ReadReplicaCount | sagemaker:variant:DesiredInstanceCount | custom-resource:ResourceType:Property | comprehend:document-classifier-endpoint:DesiredInferenceUnits | comprehend:entity-recognizer-endpoint:DesiredInferenceUnits | lambda:function:ProvisionedConcurrency | cassandra:table:ReadCapacityUnits | cassandra:table:WriteCapacityUnits | kafka:broker-storage:VolumeSize | elasticache:replication-group:NodeGroups | elasticache:replication-group:Replicas | neptune:cluster:ReadReplicaCount`

Required: Yes

ServiceNamespace (p. 45)

The namespace of the AWS service that provides the resource. For a resource provided by your own application or service, use `custom-resource` instead.

Type: String

Valid Values: `ecs | elasticmapreduce | ec2 | appstream | dynamodb | rds | sagemaker | custom-resource | comprehend | lambda | cassandra | kafka | elasticache | neptune`

Required: Yes

StepScalingPolicyConfiguration (p. 45)

A step scaling policy.

This parameter is required if you are creating a policy and the policy type is `StepScaling`.

Type: StepScalingPolicyConfiguration (p. 93) object

Required: No

TargetTrackingScalingPolicyConfiguration (p. 45)

A target tracking scaling policy. Includes support for predefined or customized metrics.

This parameter is required if you are creating a policy and the policy type is `TargetTrackingScaling`.

Type: TargetTrackingScalingPolicyConfiguration (p. 96) object

Required: No

Response Syntax

```
{
   "Alarms": [
      {
         "AlarmARN": "string",
         "AlarmName": "string"
      }
   ],
   "PolicyARN": "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.

Alarms (p. 49)

The CloudWatch alarms created for the target tracking scaling policy.

Type: Array of Alarm (p. 67) objects

PolicyARN (p. 49)

The Amazon Resource Name (ARN) of the resulting scaling policy.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1600.

Pattern: `[\u0020-\uD7FF\uE000-\uFFFD\uD800\uDC00-\uDBFF\uDFFF\r\n\t]*`

Errors

For information about the errors that are common to all actions, see Common Errors (p. 101).

ConcurrentUpdateException

Concurrent updates caused an exception, for example, if you request an update to an Application Auto Scaling resource that already has a pending update.

HTTP Status Code: 400

FailedResourceAccessException

Failed access to resources caused an exception. This exception is thrown when Application Auto Scaling is unable to retrieve the alarms associated with a scaling policy due to a client error, for example, if the role ARN specified for a scalable target does not have permission to call the CloudWatch DescribeAlarms on your behalf.

HTTP Status Code: 400

InternalServiceException

The service encountered an internal error.

HTTP Status Code: 400

LimitExceededException

A per-account resource limit is exceeded. For more information, see Application Auto Scaling service quotas.

HTTP Status Code: 400

ObjectNotFoundException

The specified object could not be found. For any operation that depends on the existence of a scalable target, this exception is thrown if the scalable target with the specified service namespace, resource ID, and scalable dimension does not exist. For any operation that deletes or deregisters a resource, this exception is thrown if the resource cannot be found.

HTTP Status Code: 400

ValidationException

An exception was thrown for a validation issue. Review the available parameters for the API request.

HTTP Status Code: 400

Examples

If you plan to create requests manually, you must replace the Authorization header contents in the examples (`AUTHPARAMS`) with a signature. For more information, see Signature Version 4 Signing Process in the *Amazon Web Services General Reference*. If you plan to use the AWS CLI or one of the AWS SDKs, these tools sign the requests for you.

Example of a target tracking scaling policy

The following example applies a target tracking scaling policy to an Amazon ECS service called `web-app` in the `default` cluster. The policy keeps the average CPU utilization of the service at 75 percent, with scale-out and scale-in cooldown periods of 60 seconds. The output contains the ARNs and names of the two CloudWatch alarms created on your behalf.

Sample Request

```
POST / HTTP/1.1
Host: autoscaling.us-west-2.amazonaws.com
Accept-Encoding: identity
```

```
Content-Length: 392
X-Amz-Target: AnyScaleFrontendService.PutScalingPolicy
X-Amz-Date: 20190506T191044Z
User-Agent: aws-cli/1.10.23 Python/2.7.11 Darwin/15.4.0 botocore/1.4.8
Content-Type: application/x-amz-json-1.1
Authorization: AUTHPARAMS

{
    "PolicyName": "cpu75-target-tracking-scaling-policy",
    "PolicyType": "TargetTrackingScaling",
    "TargetTrackingScalingPolicyConfiguration": {
        "TargetValue": 75.0,
        "PredefinedMetricSpecification": {
            "PredefinedMetricType": "ECSServiceAverageCPUUtilization"
        },
        "ScaleOutCooldown": 60,
        "ScaleInCooldown": 60
    },
    "ServiceNamespace": "ecs",
    "ScalableDimension": "ecs:service:DesiredCount",
    "ResourceId": "service/default/web-app"
}
```

Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: 4a0f8f18-cb5f-11e0-8364-37acb4b5a1b2
Content-Type: application/x-amz-json-1.1
Content-Length: 314
Date: Fri, 06 May 2019 19:10:44 GMT

{
    "PolicyARN": "arn:aws:autoscaling:us-west-2:012345678910:scalingPolicy:6d8972f3-
efc8-437c-92d1-6270f29a66e7:resource/ecs/service/default/web-app:policyName/cpu75-target-
tracking-scaling-policy",
    "Alarms": [
        {
            "AlarmARN": "arn:aws:cloudwatch:us-west-2:012345678910:alarm:TargetTracking-
service/default/web-app-AlarmHigh-d4f0770c-b46e-434a-a60f-3b36d653feca",
            "AlarmName": "TargetTracking-service/default/web-app-AlarmHigh-d4f0770c-
b46e-434a-a60f-3b36d653feca"
        },
        {
            "AlarmARN": "arn:aws:cloudwatch:us-west-2:012345678910:alarm:TargetTracking-
service/default/web-app-AlarmLow-1b437334-d19b-4a63-a812-6c67aaf2910d",
            "AlarmName": "TargetTracking-service/default/web-app-AlarmLow-1b437334-
d19b-4a63-a812-6c67aaf2910d"
        }
    ]
}
```

Example of a step scaling policy for scale out

The following example applies a step scaling policy to an Amazon ECS service called `web-app` in the `default` cluster. The policy increases the desired count of the service by 200%, with a cooldown period of 60 seconds. The output includes the ARN for the policy, which you need to create the CloudWatch alarm.

Sample Request

```
POST / HTTP/1.1
```

```
Host: autoscaling.us-west-2.amazonaws.com
Accept-Encoding: identity
Content-Length: 358
X-Amz-Target: AnyScaleFrontendService.PutScalingPolicy
X-Amz-Date: 20190506T191138Z
User-Agent: aws-cli/1.10.23 Python/2.7.11 Darwin/15.4.0 botocore/1.4.8
Content-Type: application/x-amz-json-1.1
Authorization: AUTHPARAMS

{
    "PolicyName": "my-scale-out-policy",
    "PolicyType": "StepScaling",
    "StepScalingPolicyConfiguration": {
        "AdjustmentType": "PercentChangeInCapacity",
        "Cooldown": 60,
        "MetricAggregationType": "Average",
        "StepAdjustments": [
            {
                "ScalingAdjustment": 200,
                "MetricIntervalLowerBound": 0
            }
        ]
    },
    "ServiceNamespace": "ecs",
    "ScalableDimension": "ecs:service:DesiredCount",
    "ResourceId": "service/default/web-app"
}
```

Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: 5ec6d08e-17ce-1e165a468-73cad4b5cel6
Content-Type: application/x-amz-json-1.1
Content-Length: 175
Date: Fri, 06 May 2019 19:11:38 GMT

{
    "PolicyARN": "arn:aws:autoscaling:us-west-2:012345678910:scalingPolicy:ac542982-
cbeb-4294-891c-a5a941dfa787:resource/ecs/service/default/web-app:policyName/my-scale-out-
policy"
}
```

Example of a step scaling policy for scale in

The following example applies a step scaling policy to the same Amazon ECS service as in the preceding example. The policy has two step adjustments that decrease the desired count of the service by 25% or 50%, depending on the size of the alarm breach, with a cooldown period of 120 seconds. The output includes the ARN for the policy, which you need to create the CloudWatch alarm.

Sample Request

```
POST / HTTP/1.1
Host: autoscaling.us-west-2.amazonaws.com
Accept-Encoding: identity
Content-Length: 431
X-Amz-Target: AnyScaleFrontendService.PutScalingPolicy
X-Amz-Date: 20190506T191152Z
User-Agent: aws-cli/1.10.23 Python/2.7.11 Darwin/15.4.0 botocore/1.4.8
Content-Type: application/x-amz-json-1.1
Authorization: AUTHPARAMS
```

```
{
    "PolicyName": "my-scale-in-policy",
    "PolicyType": "StepScaling",
    "StepScalingPolicyConfiguration": {
        "AdjustmentType": "PercentChangeInCapacity",
        "Cooldown": 120,
        "MetricAggregationType": "Average",
        "MinAdjustmentMagnitude": 1,
        "StepAdjustments": [
            {
                "ScalingAdjustment": -25,
                "MetricIntervalLowerBound": -15,
                "MetricIntervalUpperBound": 0
             },
            {
                "ScalingAdjustment": -50,
                "MetricIntervalUpperBound": -15
            }
        ]
    },
    "ServiceNamespace": "ecs",
    "ScalableDimension": "ecs:service:DesiredCount",
    "ResourceId": "service/default/web-app"
}
```

Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: 5a64c9e1-3cfe-11e74bfad-8d1c65ec6d08
Content-Type: application/x-amz-json-1.1
Content-Length: 174
Date: Fri, 06 May 2019 19:11:52 GMT

{
    "PolicyARN": "arn:aws:autoscaling:us-west-2:012345678910:scalingPolicy:6d8972f3-
efc8-437c-92d1-6270f29a66e7:resource/ecs/service/default/web-app:policyName/my-scale-in-
policy"
}
```

See Also

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

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

PutScheduledAction

Creates or updates a scheduled action for an Application Auto Scaling scalable target.

Each scalable target is identified by a service namespace, resource ID, and scalable dimension. A scheduled action applies to the scalable target identified by those three attributes. You cannot create a scheduled action until you have registered the resource as a scalable target.

When start and end times are specified with a recurring schedule using a cron expression or rates, they form the boundaries for when the recurring action starts and stops.

To update a scheduled action, specify the parameters that you want to change. If you don't specify start and end times, the old values are deleted.

For more information, see Scheduled scaling in the *Application Auto Scaling User Guide*.

> **Note**
> If a scalable target is deregistered, the scalable target is no longer available to run scheduled actions. Any scheduled actions that were specified for the scalable target are deleted.

Request Syntax

```
{
   "EndTime": number,
   "ResourceId": "string",
   "ScalableDimension": "string",
   "ScalableTargetAction": {
      "MaxCapacity": number,
      "MinCapacity": number
   },
   "Schedule": "string",
   "ScheduledActionName": "string",
   "ServiceNamespace": "string",
   "StartTime": number,
   "Timezone": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 99).

The request accepts the following data in JSON format.

EndTime (p. 54)

The date and time for the recurring schedule to end, in UTC.

Type: Timestamp

Required: No

ResourceId (p. 54)

The identifier of the resource associated with the scheduled action. This string consists of the resource type and unique identifier.

- ECS service - The resource type is `service` and the unique identifier is the cluster name and service name. Example: `service/default/sample-webapp`.

- Spot Fleet - The resource type is `spot-fleet-request` and the unique identifier is the Spot Fleet request ID. Example: `spot-fleet-request/sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE`.
- EMR cluster - The resource type is `instancegroup` and the unique identifier is the cluster ID and instance group ID. Example: `instancegroup/j-2EEZNYKUA1NTV/ig-1791Y4E1L8YI0`.
- AppStream 2.0 fleet - The resource type is `fleet` and the unique identifier is the fleet name. Example: `fleet/sample-fleet`.
- DynamoDB table - The resource type is `table` and the unique identifier is the table name. Example: `table/my-table`.
- DynamoDB global secondary index - The resource type is `index` and the unique identifier is the index name. Example: `table/my-table/index/my-table-index`.
- Aurora DB cluster - The resource type is `cluster` and the unique identifier is the cluster name. Example: `cluster:my-db-cluster`.
- SageMaker endpoint variant - The resource type is `variant` and the unique identifier is the resource ID. Example: `endpoint/my-end-point/variant/KMeansClustering`.
- Custom resources are not supported with a resource type. This parameter must specify the `OutputValue` from the CloudFormation template stack used to access the resources. The unique identifier is defined by the service provider. More information is available in our GitHub repository.
- Amazon Comprehend document classification endpoint - The resource type and unique identifier are specified using the endpoint ARN. Example: `arn:aws:comprehend:us-west-2:123456789012:document-classifier-endpoint/EXAMPLE`.
- Amazon Comprehend entity recognizer endpoint - The resource type and unique identifier are specified using the endpoint ARN. Example: `arn:aws:comprehend:us-west-2:123456789012:entity-recognizer-endpoint/EXAMPLE`.
- Lambda provisioned concurrency - The resource type is `function` and the unique identifier is the function name with a function version or alias name suffix that is not `$LATEST`. Example: `function:my-function:prod` or `function:my-function:1`.
- Amazon Keyspaces table - The resource type is `table` and the unique identifier is the table name. Example: `keyspace/mykeyspace/table/mytable`.
- Amazon MSK cluster - The resource type and unique identifier are specified using the cluster ARN. Example: `arn:aws:kafka:us-east-1:123456789012:cluster/demo-cluster-1/6357e0b2-0e6a-4b86-a0b4-70df934c2e31-5`.
- Amazon ElastiCache replication group - The resource type is `replication-group` and the unique identifier is the replication group name. Example: `replication-group/mycluster`.
- Neptune cluster - The resource type is `cluster` and the unique identifier is the cluster name. Example: `cluster:mycluster`.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1600.

Pattern: `[\u0020-\uD7FF\uE000-\uFFFD\uD800\uDC00-\uDBFF\uDFFF\r\n\t]*`

Required: Yes

ScalableDimension (p. 54)

The scalable dimension. This string consists of the service namespace, resource type, and scaling property.

- `ecs:service:DesiredCount` - The desired task count of an ECS service.
- `elasticmapreduce:instancegroup:InstanceCount` - The instance count of an EMR Instance Group.
- `ec2:spot-fleet-request:TargetCapacity` - The target capacity of a Spot Fleet.
- `appstream:fleet:DesiredCapacity` - The desired capacity of an AppStream 2.0 fleet.

- `dynamodb:table:ReadCapacityUnits` - The provisioned read capacity for a DynamoDB table.
- `dynamodb:table:WriteCapacityUnits` - The provisioned write capacity for a DynamoDB table.
- `dynamodb:index:ReadCapacityUnits` - The provisioned read capacity for a DynamoDB global secondary index.
- `dynamodb:index:WriteCapacityUnits` - The provisioned write capacity for a DynamoDB global secondary index.
- `rds:cluster:ReadReplicaCount` - The count of Aurora Replicas in an Aurora DB cluster. Available for Aurora MySQL-compatible edition and Aurora PostgreSQL-compatible edition.
- `sagemaker:variant:DesiredInstanceCount` - The number of EC2 instances for an SageMaker model endpoint variant.
- `custom-resource:ResourceType:Property` - The scalable dimension for a custom resource provided by your own application or service.
- `comprehend:document-classifier-endpoint:DesiredInferenceUnits` - The number of inference units for an Amazon Comprehend document classification endpoint.
- `comprehend:entity-recognizer-endpoint:DesiredInferenceUnits` - The number of inference units for an Amazon Comprehend entity recognizer endpoint.
- `lambda:function:ProvisionedConcurrency` - The provisioned concurrency for a Lambda function.
- `cassandra:table:ReadCapacityUnits` - The provisioned read capacity for an Amazon Keyspaces table.
- `cassandra:table:WriteCapacityUnits` - The provisioned write capacity for an Amazon Keyspaces table.
- `kafka:broker-storage:VolumeSize` - The provisioned volume size (in GiB) for brokers in an Amazon MSK cluster.
- `elasticache:replication-group:NodeGroups` - The number of node groups for an Amazon ElastiCache replication group.
- `elasticache:replication-group:Replicas` - The number of replicas per node group for an Amazon ElastiCache replication group.
- `neptune:cluster:ReadReplicaCount` - The count of read replicas in an Amazon Neptune DB cluster.

Type: String

Valid Values: `ecs:service:DesiredCount | ec2:spot-fleet-request:TargetCapacity | elasticmapreduce:instancegroup:InstanceCount | appstream:fleet:DesiredCapacity | dynamodb:table:ReadCapacityUnits | dynamodb:table:WriteCapacityUnits | dynamodb:index:ReadCapacityUnits | dynamodb:index:WriteCapacityUnits | rds:cluster:ReadReplicaCount | sagemaker:variant:DesiredInstanceCount | custom-resource:ResourceType:Property | comprehend:document-classifier-endpoint:DesiredInferenceUnits | comprehend:entity-recognizer-endpoint:DesiredInferenceUnits | lambda:function:ProvisionedConcurrency | cassandra:table:ReadCapacityUnits | cassandra:table:WriteCapacityUnits | kafka:broker-storage:VolumeSize | elasticache:replication-group:NodeGroups | elasticache:replication-group:Replicas | neptune:cluster:ReadReplicaCount`

Required: Yes

ScalableTargetAction (p. 54)

The new minimum and maximum capacity. You can set both values or just one. At the scheduled time, if the current capacity is below the minimum capacity, Application Auto Scaling scales out to the minimum capacity. If the current capacity is above the maximum capacity, Application Auto Scaling scales in to the maximum capacity.

Type: ScalableTargetAction (p. 77) object

Required: No

Schedule (p. 54)

The schedule for this action. The following formats are supported:

- At expressions - "`at(`*`yyyy-mm-dd`*`T`*`hh`*`:`*`mm`*`:`*`ss`*`)`"
- Rate expressions - "`rate(`*`value unit`*`)`"
- Cron expressions - "`cron(`*`fields`*`)`"

At expressions are useful for one-time schedules. Cron expressions are useful for scheduled actions that run periodically at a specified date and time, and rate expressions are useful for scheduled actions that run at a regular interval.

At and cron expressions use Universal Coordinated Time (UTC) by default.

The cron format consists of six fields separated by white spaces: [Minutes] [Hours] [Day_of_Month] [Month] [Day_of_Week] [Year].

For rate expressions, *value* is a positive integer and *unit* is `minute` | `minutes` | `hour` | `hours` | `day` | `days`.

For more information and examples, see Example scheduled actions for Application Auto Scaling in the *Application Auto Scaling User Guide*.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1600.

Pattern: `[\u0020-\uD7FF\uE000-\uFFFD\uD800\uDC00-\uDBFF\uDFFF\r\n\t]*`

Required: No

ScheduledActionName (p. 54)

The name of the scheduled action. This name must be unique among all other scheduled actions on the specified scalable target.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

Pattern: `(?!((^[ ]+.*)|(.*([\u0000-\u001f]|[\u007f-\u009f]|[:/|])+.*)|(.*[ ]+$))).+`

Required: Yes

ServiceNamespace (p. 54)

The namespace of the AWS service that provides the resource. For a resource provided by your own application or service, use `custom-resource` instead.

Type: String

Valid Values: `ecs | elasticmapreduce | ec2 | appstream | dynamodb | rds | sagemaker | custom-resource | comprehend | lambda | cassandra | kafka | elasticache | neptune`

Required: Yes

StartTime (p. 54)

The date and time for this scheduled action to start, in UTC.

Type: Timestamp

Required: No

Timezone (p. 54)

Specifies the time zone used when setting a scheduled action by using an at or cron expression. If a time zone is not provided, UTC is used by default.

Valid values are the canonical names of the IANA time zones supported by Joda-Time (such as `Etc/GMT+9` or `Pacific/Tahiti`). For more information, see https://www.joda.org/joda-time/timezones.html.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1600.

Pattern: `[\u0020-\uD7FF\uE000-\uFFFD\uD800\uDC00-\uDBFF\uDFFF\r\n\t]*`

Required: No

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 Errors (p. 101).

ConcurrentUpdateException

Concurrent updates caused an exception, for example, if you request an update to an Application Auto Scaling resource that already has a pending update.

HTTP Status Code: 400

InternalServiceException

The service encountered an internal error.

HTTP Status Code: 400

LimitExceededException

A per-account resource limit is exceeded. For more information, see Application Auto Scaling service quotas.

HTTP Status Code: 400

ObjectNotFoundException

The specified object could not be found. For any operation that depends on the existence of a scalable target, this exception is thrown if the scalable target with the specified service namespace, resource ID, and scalable dimension does not exist. For any operation that deletes or deregisters a resource, this exception is thrown if the resource cannot be found.

HTTP Status Code: 400

ValidationException

An exception was thrown for a validation issue. Review the available parameters for the API request.

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

RegisterScalableTarget

Registers or updates a scalable target.

A scalable target is a resource that Application Auto Scaling can scale out and scale in. Scalable targets are uniquely identified by the combination of resource ID, scalable dimension, and namespace.

When you register a new scalable target, you must specify values for minimum and maximum capacity. Current capacity will be adjusted within the specified range when scaling starts. Application Auto Scaling scaling policies will not scale capacity to values that are outside of this range.

After you register a scalable target, you do not need to register it again to use other Application Auto Scaling operations. To see which resources have been registered, use DescribeScalableTargets. You can also view the scaling policies for a service namespace by using DescribeScalableTargets. If you no longer need a scalable target, you can deregister it by using DeregisterScalableTarget.

To update a scalable target, specify the parameters that you want to change. Include the parameters that identify the scalable target: resource ID, scalable dimension, and namespace. Any parameters that you don't specify are not changed by this update request.

> **Note**
> If you call the `RegisterScalableTarget` API to update an existing scalable target, Application Auto Scaling retrieves the current capacity of the resource. If it is below the minimum capacity or above the maximum capacity, Application Auto Scaling adjusts the capacity of the scalable target to place it within these bounds, even if you don't include the `MinCapacity` or `MaxCapacity` request parameters.

Request Syntax

```
{
   "MaxCapacity": number,
   "MinCapacity": number,
   "ResourceId": "string",
   "RoleARN": "string",
   "ScalableDimension": "string",
   "ServiceNamespace": "string",
   "SuspendedState": {
      "DynamicScalingInSuspended": boolean,
      "DynamicScalingOutSuspended": boolean,
      "ScheduledScalingSuspended": boolean
   }
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 99).

The request accepts the following data in JSON format.

MaxCapacity (p. 60)

The maximum value that you plan to scale out to. When a scaling policy is in effect, Application Auto Scaling can scale out (expand) as needed to the maximum capacity limit in response to changing demand. This property is required when registering a new scalable target.

Although you can specify a large maximum capacity, note that service quotas may impose lower limits. Each service has its own default quotas for the maximum capacity of the resource. If you want to specify a higher limit, you can request an increase. For more information, consult the documentation for that service. For information about the default quotas for each service, see Service Endpoints and Quotas in the *Amazon Web Services General Reference*.

Type: Integer

Required: No

MinCapacity (p. 60)

The minimum value that you plan to scale in to. When a scaling policy is in effect, Application Auto Scaling can scale in (contract) as needed to the minimum capacity limit in response to changing demand. This property is required when registering a new scalable target.

For certain resources, the minimum value allowed is 0. This includes Lambda provisioned concurrency, Spot Fleet, ECS services, Aurora DB clusters, EMR clusters, and custom resources. For all other resources, the minimum value allowed is 1.

Type: Integer

Required: No

ResourceId (p. 60)

The identifier of the resource that is associated with the scalable target. This string consists of the resource type and unique identifier.

- ECS service - The resource type is `service` and the unique identifier is the cluster name and service name. Example: `service/default/sample-webapp`.
- Spot Fleet - The resource type is `spot-fleet-request` and the unique identifier is the Spot Fleet request ID. Example: `spot-fleet-request/sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE`.
- EMR cluster - The resource type is `instancegroup` and the unique identifier is the cluster ID and instance group ID. Example: `instancegroup/j-2EEZNYKUA1NTV/ig-1791Y4E1L8YI0`.
- AppStream 2.0 fleet - The resource type is `fleet` and the unique identifier is the fleet name. Example: `fleet/sample-fleet`.
- DynamoDB table - The resource type is `table` and the unique identifier is the table name. Example: `table/my-table`.
- DynamoDB global secondary index - The resource type is `index` and the unique identifier is the index name. Example: `table/my-table/index/my-table-index`.
- Aurora DB cluster - The resource type is `cluster` and the unique identifier is the cluster name. Example: `cluster:my-db-cluster`.
- SageMaker endpoint variant - The resource type is `variant` and the unique identifier is the resource ID. Example: `endpoint/my-end-point/variant/KMeansClustering`.
- Custom resources are not supported with a resource type. This parameter must specify the `OutputValue` from the CloudFormation template stack used to access the resources. The unique identifier is defined by the service provider. More information is available in our GitHub repository.
- Amazon Comprehend document classification endpoint - The resource type and unique identifier are specified using the endpoint ARN. Example: `arn:aws:comprehend:us-west-2:123456789012:document-classifier-endpoint/EXAMPLE`.
- Amazon Comprehend entity recognizer endpoint - The resource type and unique identifier are specified using the endpoint ARN. Example: `arn:aws:comprehend:us-west-2:123456789012:entity-recognizer-endpoint/EXAMPLE`.
- Lambda provisioned concurrency - The resource type is `function` and the unique identifier is the function name with a function version or alias name suffix that is not `$LATEST`. Example: `function:my-function:prod` or `function:my-function:1`.
- Amazon Keyspaces table - The resource type is `table` and the unique identifier is the table name. Example: `keyspace/mykeyspace/table/mytable`.

- Amazon MSK cluster - The resource type and unique identifier are specified using the cluster ARN. Example: `arn:aws:kafka:us-east-1:123456789012:cluster/demo-cluster-1/6357e0b2-0e6a-4b86-a0b4-70df934c2e31-5`.
- Amazon ElastiCache replication group - The resource type is `replication-group` and the unique identifier is the replication group name. Example: `replication-group/mycluster`.
- Neptune cluster - The resource type is `cluster` and the unique identifier is the cluster name. Example: `cluster:mycluster`.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1600.

Pattern: `[\u0020-\uD7FF\uE000-\uFFFD\uD800\uDC00-\uDBFF\uDFFF\r\n\t]*`

Required: Yes

RoleARN (p. 60)

This parameter is required for services that do not support service-linked roles (such as Amazon EMR), and it must specify the ARN of an IAM role that allows Application Auto Scaling to modify the scalable target on your behalf.

If the service supports service-linked roles, Application Auto Scaling uses a service-linked role, which it creates if it does not yet exist. For more information, see Application Auto Scaling IAM roles.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1600.

Pattern: `[\u0020-\uD7FF\uE000-\uFFFD\uD800\uDC00-\uDBFF\uDFFF\r\n\t]*`

Required: No

ScalableDimension (p. 60)

The scalable dimension associated with the scalable target. This string consists of the service namespace, resource type, and scaling property.

- `ecs:service:DesiredCount` - The desired task count of an ECS service.
- `elasticmapreduce:instancegroup:InstanceCount` - The instance count of an EMR Instance Group.
- `ec2:spot-fleet-request:TargetCapacity` - The target capacity of a Spot Fleet.
- `appstream:fleet:DesiredCapacity` - The desired capacity of an AppStream 2.0 fleet.
- `dynamodb:table:ReadCapacityUnits` - The provisioned read capacity for a DynamoDB table.
- `dynamodb:table:WriteCapacityUnits` - The provisioned write capacity for a DynamoDB table.
- `dynamodb:index:ReadCapacityUnits` - The provisioned read capacity for a DynamoDB global secondary index.
- `dynamodb:index:WriteCapacityUnits` - The provisioned write capacity for a DynamoDB global secondary index.
- `rds:cluster:ReadReplicaCount` - The count of Aurora Replicas in an Aurora DB cluster. Available for Aurora MySQL-compatible edition and Aurora PostgreSQL-compatible edition.
- `sagemaker:variant:DesiredInstanceCount` - The number of EC2 instances for an SageMaker model endpoint variant.
- `custom-resource:ResourceType:Property` - The scalable dimension for a custom resource provided by your own application or service.
- `comprehend:document-classifier-endpoint:DesiredInferenceUnits` - The number of inference units for an Amazon Comprehend document classification endpoint.

- `comprehend:entity-recognizer-endpoint:DesiredInferenceUnits` - The number of inference units for an Amazon Comprehend entity recognizer endpoint.
- `lambda:function:ProvisionedConcurrency` - The provisioned concurrency for a Lambda function.
- `cassandra:table:ReadCapacityUnits` - The provisioned read capacity for an Amazon Keyspaces table.
- `cassandra:table:WriteCapacityUnits` - The provisioned write capacity for an Amazon Keyspaces table.
- `kafka:broker-storage:VolumeSize` - The provisioned volume size (in GiB) for brokers in an Amazon MSK cluster.
- `elasticache:replication-group:NodeGroups` - The number of node groups for an Amazon ElastiCache replication group.
- `elasticache:replication-group:Replicas` - The number of replicas per node group for an Amazon ElastiCache replication group.
- `neptune:cluster:ReadReplicaCount` - The count of read replicas in an Amazon Neptune DB cluster.

Type: String

Valid Values: `ecs:service:DesiredCount | ec2:spot-fleet-request:TargetCapacity | elasticmapreduce:instancegroup:InstanceCount | appstream:fleet:DesiredCapacity | dynamodb:table:ReadCapacityUnits | dynamodb:table:WriteCapacityUnits | dynamodb:index:ReadCapacityUnits | dynamodb:index:WriteCapacityUnits | rds:cluster:ReadReplicaCount | sagemaker:variant:DesiredInstanceCount | custom-resource:ResourceType:Property | comprehend:document-classifier-endpoint:DesiredInferenceUnits | comprehend:entity-recognizer-endpoint:DesiredInferenceUnits | lambda:function:ProvisionedConcurrency | cassandra:table:ReadCapacityUnits | cassandra:table:WriteCapacityUnits | kafka:broker-storage:VolumeSize | elasticache:replication-group:NodeGroups | elasticache:replication-group:Replicas | neptune:cluster:ReadReplicaCount`

Required: Yes

ServiceNamespace (p. 60)

The namespace of the AWS service that provides the resource. For a resource provided by your own application or service, use `custom-resource` instead.

Type: String

Valid Values: `ecs | elasticmapreduce | ec2 | appstream | dynamodb | rds | sagemaker | custom-resource | comprehend | lambda | cassandra | kafka | elasticache | neptune`

Required: Yes

SuspendedState (p. 60)

An embedded object that contains attributes and attribute values that are used to suspend and resume automatic scaling. Setting the value of an attribute to `true` suspends the specified scaling activities. Setting it to `false` (default) resumes the specified scaling activities.

Suspension Outcomes

- For `DynamicScalingInSuspended`, while a suspension is in effect, all scale-in activities that are triggered by a scaling policy are suspended.
- For `DynamicScalingOutSuspended`, while a suspension is in effect, all scale-out activities that are triggered by a scaling policy are suspended.

- For ScheduledScalingSuspended, while a suspension is in effect, all scaling activities that involve scheduled actions are suspended.

For more information, see Suspending and resuming scaling in the *Application Auto Scaling User Guide*.

Type: SuspendedState (p. 95) object

Required: No

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 Errors (p. 101).

ConcurrentUpdateException

Concurrent updates caused an exception, for example, if you request an update to an Application Auto Scaling resource that already has a pending update.

HTTP Status Code: 400

InternalServiceException

The service encountered an internal error.

HTTP Status Code: 400

LimitExceededException

A per-account resource limit is exceeded. For more information, see Application Auto Scaling service quotas.

HTTP Status Code: 400

ValidationException

An exception was thrown for a validation issue. Review the available parameters for the API request.

HTTP Status Code: 400

Examples

If you plan to create requests manually, you must replace the Authorization header contents in the examples (AUTHPARAMS) with a signature. For more information, see Signature Version 4 Signing Process in the *Amazon Web Services General Reference*. If you plan to use the AWS CLI or one of the AWS SDKs, these tools sign the requests for you.

Example

The following example registers an Amazon ECS service with Application Auto Scaling.

Sample Request

```
POST / HTTP/1.1
```

```
Host: autoscaling.us-west-2.amazonaws.com
Accept-Encoding: identity
Content-Length: 229
X-Amz-Target: AnyScaleFrontendService.RegisterScalableTarget
X-Amz-Date: 20190506T182145Z
User-Agent: aws-cli/1.10.23 Python/2.7.11 Darwin/15.4.0 botocore/1.4.8
Content-Type: application/x-amz-json-1.1
Authorization: AUTHPARAMS

{
    "ScalableDimension": "ecs:service:DesiredCount",
    "ResourceId": "service/default/web-app",
    "ServiceNamespace": "ecs",
    "MinCapacity": 1,
    "MaxCapacity": 10
}
```

See Also

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

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

Data Types

The Application Auto Scaling 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:

- Alarm (p. 67)
- CustomizedMetricSpecification (p. 68)
- MetricDimension (p. 70)
- PredefinedMetricSpecification (p. 71)
- ScalableTarget (p. 73)
- ScalableTargetAction (p. 77)
- ScalingActivity (p. 78)
- ScalingPolicy (p. 82)
- ScheduledAction (p. 86)
- StepAdjustment (p. 91)
- StepScalingPolicyConfiguration (p. 93)
- SuspendedState (p. 95)
- TargetTrackingScalingPolicyConfiguration (p. 96)

Alarm

Represents a CloudWatch alarm associated with a scaling policy.

Contents

AlarmARN

The Amazon Resource Name (ARN) of the alarm.

Type: String

Pattern: `[\u0020-\uD7FF\uE000-\uFFFD\uD800\uDC00-\uDBFF\uDFFF\r\n\t]*`

Required: Yes

AlarmName

The name of the alarm.

Type: String

Pattern: `[\u0020-\uD7FF\uE000-\uFFFD\uD800\uDC00-\uDBFF\uDFFF\r\n\t]*`

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 Go
- AWS SDK for Java V2
- AWS SDK for Ruby V3

CustomizedMetricSpecification

Represents a CloudWatch metric of your choosing for a target tracking scaling policy to use with Application Auto Scaling.

For information about the available metrics for a service, see AWS Services That Publish CloudWatch Metrics in the *Amazon CloudWatch User Guide*.

To create your customized metric specification:

- Add values for each required parameter from CloudWatch. You can use an existing metric, or a new metric that you create. To use your own metric, you must first publish the metric to CloudWatch. For more information, see Publish Custom Metrics in the *Amazon CloudWatch User Guide*.
- Choose a metric that changes proportionally with capacity. The value of the metric should increase or decrease in inverse proportion to the number of capacity units. That is, the value of the metric should decrease when capacity increases, and increase when capacity decreases.

For more information about CloudWatch, see Amazon CloudWatch Concepts.

Contents

Dimensions

The dimensions of the metric.

Conditional: If you published your metric with dimensions, you must specify the same dimensions in your scaling policy.

Type: Array of MetricDimension (p. 70) objects

Required: No

MetricName

The name of the metric.

Type: String

Required: Yes

Namespace

The namespace of the metric.

Type: String

Required: Yes

Statistic

The statistic of the metric.

Type: String

Valid Values: `Average | Minimum | Maximum | SampleCount | Sum`

Required: Yes

Unit

The unit of the metric.

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 Go
- AWS SDK for Java V2
- AWS SDK for Ruby V3

MetricDimension

Describes the dimension names and values associated with a metric.

Contents

Name

The name of the dimension.

Type: String

Required: Yes

Value

The value of the dimension.

Type: String

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 Go
- AWS SDK for Java V2
- AWS SDK for Ruby V3

PredefinedMetricSpecification

Represents a predefined metric for a target tracking scaling policy to use with Application Auto Scaling.

Only the Amazon Web Services that you're using send metrics to Amazon CloudWatch. To determine whether a desired metric already exists by looking up its namespace and dimension using the CloudWatch metrics dashboard in the console, follow the procedure in Building dashboards with CloudWatch in the *Application Auto Scaling User Guide*.

Contents

PredefinedMetricType

The metric type. The `ALBRequestCountPerTarget` metric type applies only to Spot Fleet requests and ECS services.

Type: String

Valid Values: `DynamoDBReadCapacityUtilization | DynamoDBWriteCapacityUtilization | ALBRequestCountPerTarget | RDSReaderAverageCPUUtilization | RDSReaderAverageDatabaseConnections | EC2SpotFleetRequestAverageCPUUtilization | EC2SpotFleetRequestAverageNetworkIn | EC2SpotFleetRequestAverageNetworkOut | SageMakerVariantInvocationsPerInstance | ECSServiceAverageCPUUtilization | ECSServiceAverageMemoryUtilization | AppStreamAverageCapacityUtilization | ComprehendInferenceUtilization | LambdaProvisionedConcurrencyUtilization | CassandraReadCapacityUtilization | CassandraWriteCapacityUtilization | KafkaBrokerStorageUtilization | ElastiCachePrimaryEngineCPUUtilization | ElastiCacheReplicaEngineCPUUtilization | ElastiCacheDatabaseMemoryUsageCountedForEvictPercentage | NeptuneReaderAverageCPUUtilization`

Required: Yes

ResourceLabel

Identifies the resource associated with the metric type. You can't specify a resource label unless the metric type is `ALBRequestCountPerTarget` and there is a target group attached to the Spot Fleet request or ECS service.

You create the resource label by appending the final portion of the load balancer ARN and the final portion of the target group ARN into a single value, separated by a forward slash (/). The format of the resource label is:

`app/my-alb/778d41231b141a0f/targetgroup/my-alb-target-group/943f017f100becff.`

Where:

- app/<load-balancer-name>/<load-balancer-id> is the final portion of the load balancer ARN
- targetgroup/<target-group-name>/<target-group-id> is the final portion of the target group ARN.

To find the ARN for an Application Load Balancer, use the DescribeLoadBalancers API operation. To find the ARN for the target group, use the DescribeTargetGroups API operation.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1023.

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 Go
- AWS SDK for Java V2
- AWS SDK for Ruby V3

ScalableTarget

Represents a scalable target.

Contents

CreationTime

The Unix timestamp for when the scalable target was created.

Type: Timestamp

Required: Yes

MaxCapacity

The maximum value to scale to in response to a scale-out activity.

Type: Integer

Required: Yes

MinCapacity

The minimum value to scale to in response to a scale-in activity.

Type: Integer

Required: Yes

ResourceId

The identifier of the resource associated with the scalable target. This string consists of the resource type and unique identifier.

- ECS service - The resource type is `service` and the unique identifier is the cluster name and service name. Example: `service/default/sample-webapp`.
- Spot Fleet - The resource type is `spot-fleet-request` and the unique identifier is the Spot Fleet request ID. Example: `spot-fleet-request/sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE`.
- EMR cluster - The resource type is `instancegroup` and the unique identifier is the cluster ID and instance group ID. Example: `instancegroup/j-2EEZNYKUA1NTV/ig-1791Y4E1L8YI0`.
- AppStream 2.0 fleet - The resource type is `fleet` and the unique identifier is the fleet name. Example: `fleet/sample-fleet`.
- DynamoDB table - The resource type is `table` and the unique identifier is the table name. Example: `table/my-table`.
- DynamoDB global secondary index - The resource type is `index` and the unique identifier is the index name. Example: `table/my-table/index/my-table-index`.
- Aurora DB cluster - The resource type is `cluster` and the unique identifier is the cluster name. Example: `cluster:my-db-cluster`.
- SageMaker endpoint variant - The resource type is `variant` and the unique identifier is the resource ID. Example: `endpoint/my-end-point/variant/KMeansClustering`.
- Custom resources are not supported with a resource type. This parameter must specify the `OutputValue` from the CloudFormation template stack used to access the resources. The unique identifier is defined by the service provider. More information is available in our GitHub repository.
- Amazon Comprehend document classification endpoint - The resource type and unique identifier are specified using the endpoint ARN. Example: `arn:aws:comprehend:us-west-2:123456789012:document-classifier-endpoint/EXAMPLE`.

- Amazon Comprehend entity recognizer endpoint - The resource type and unique identifier are specified using the endpoint ARN. Example: `arn:aws:comprehend:us-west-2:123456789012:entity-recognizer-endpoint/EXAMPLE`.
- Lambda provisioned concurrency - The resource type is `function` and the unique identifier is the function name with a function version or alias name suffix that is not `$LATEST`. Example: `function:my-function:prod` or `function:my-function:1`.
- Amazon Keyspaces table - The resource type is `table` and the unique identifier is the table name. Example: `keyspace/mykeyspace/table/mytable`.
- Amazon MSK cluster - The resource type and unique identifier are specified using the cluster ARN. Example: `arn:aws:kafka:us-east-1:123456789012:cluster/demo-cluster-1/6357e0b2-0e6a-4b86-a0b4-70df934c2e31-5`.
- Amazon ElastiCache replication group - The resource type is `replication-group` and the unique identifier is the replication group name. Example: `replication-group/mycluster`.
- Neptune cluster - The resource type is `cluster` and the unique identifier is the cluster name. Example: `cluster:mycluster`.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1600.

Pattern: `[\u0020-\uD7FF\uE000-\uFFFD\uD800\uDC00-\uDBFF\uDFFF\r\n\t]*`

Required: Yes

RoleARN

The ARN of an IAM role that allows Application Auto Scaling to modify the scalable target on your behalf.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1600.

Pattern: `[\u0020-\uD7FF\uE000-\uFFFD\uD800\uDC00-\uDBFF\uDFFF\r\n\t]*`

Required: Yes

ScalableDimension

The scalable dimension associated with the scalable target. This string consists of the service namespace, resource type, and scaling property.

- `ecs:service:DesiredCount` - The desired task count of an ECS service.
- `elasticmapreduce:instancegroup:InstanceCount` - The instance count of an EMR Instance Group.
- `ec2:spot-fleet-request:TargetCapacity` - The target capacity of a Spot Fleet.
- `appstream:fleet:DesiredCapacity` - The desired capacity of an AppStream 2.0 fleet.
- `dynamodb:table:ReadCapacityUnits` - The provisioned read capacity for a DynamoDB table.
- `dynamodb:table:WriteCapacityUnits` - The provisioned write capacity for a DynamoDB table.
- `dynamodb:index:ReadCapacityUnits` - The provisioned read capacity for a DynamoDB global secondary index.
- `dynamodb:index:WriteCapacityUnits` - The provisioned write capacity for a DynamoDB global secondary index.
- `rds:cluster:ReadReplicaCount` - The count of Aurora Replicas in an Aurora DB cluster. Available for Aurora MySQL-compatible edition and Aurora PostgreSQL-compatible edition.
- `sagemaker:variant:DesiredInstanceCount` - The number of EC2 instances for an SageMaker model endpoint variant.

- `custom-resource:ResourceType:Property` - The scalable dimension for a custom resource provided by your own application or service.
- `comprehend:document-classifier-endpoint:DesiredInferenceUnits` - The number of inference units for an Amazon Comprehend document classification endpoint.
- `comprehend:entity-recognizer-endpoint:DesiredInferenceUnits` - The number of inference units for an Amazon Comprehend entity recognizer endpoint.
- `lambda:function:ProvisionedConcurrency` - The provisioned concurrency for a Lambda function.
- `cassandra:table:ReadCapacityUnits` - The provisioned read capacity for an Amazon Keyspaces table.
- `cassandra:table:WriteCapacityUnits` - The provisioned write capacity for an Amazon Keyspaces table.
- `kafka:broker-storage:VolumeSize` - The provisioned volume size (in GiB) for brokers in an Amazon MSK cluster.
- `elasticache:replication-group:NodeGroups` - The number of node groups for an Amazon ElastiCache replication group.
- `elasticache:replication-group:Replicas` - The number of replicas per node group for an Amazon ElastiCache replication group.
- `neptune:cluster:ReadReplicaCount` - The count of read replicas in an Amazon Neptune DB cluster.

Type: String

Valid Values: `ecs:service:DesiredCount | ec2:spot-fleet-request:TargetCapacity | elasticmapreduce:instancegroup:InstanceCount | appstream:fleet:DesiredCapacity | dynamodb:table:ReadCapacityUnits | dynamodb:table:WriteCapacityUnits | dynamodb:index:ReadCapacityUnits | dynamodb:index:WriteCapacityUnits | rds:cluster:ReadReplicaCount | sagemaker:variant:DesiredInstanceCount | custom-resource:ResourceType:Property | comprehend:document-classifier-endpoint:DesiredInferenceUnits | comprehend:entity-recognizer-endpoint:DesiredInferenceUnits | lambda:function:ProvisionedConcurrency | cassandra:table:ReadCapacityUnits | cassandra:table:WriteCapacityUnits | kafka:broker-storage:VolumeSize | elasticache:replication-group:NodeGroups | elasticache:replication-group:Replicas | neptune:cluster:ReadReplicaCount`

Required: Yes

ServiceNamespace

The namespace of the AWS service that provides the resource, or a `custom-resource`.

Type: String

Valid Values: `ecs | elasticmapreduce | ec2 | appstream | dynamodb | rds | sagemaker | custom-resource | comprehend | lambda | cassandra | kafka | elasticache | neptune`

Required: Yes

SuspendedState

Specifies whether the scaling activities for a scalable target are in a suspended state.

Type: SuspendedState (p. 95) object

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 Go
- AWS SDK for Java V2
- AWS SDK for Ruby V3

ScalableTargetAction

Represents the minimum and maximum capacity for a scheduled action.

Contents

MaxCapacity

The maximum capacity.

Although you can specify a large maximum capacity, note that service quotas may impose lower limits. Each service has its own default quotas for the maximum capacity of the resource. If you want to specify a higher limit, you can request an increase. For more information, consult the documentation for that service. For information about the default quotas for each service, see Service Endpoints and Quotas in the *Amazon Web Services General Reference*.

Type: Integer

Required: No

MinCapacity

The minimum capacity.

For certain resources, the minimum value allowed is 0. This includes Lambda provisioned concurrency, Spot Fleet, ECS services, Aurora DB clusters, EMR clusters, and custom resources. For all other resources, the minimum value allowed is 1.

Type: Integer

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 Go
- AWS SDK for Java V2
- AWS SDK for Ruby V3

ScalingActivity

Represents a scaling activity.

Contents

ActivityId

The unique identifier of the scaling activity.

Type: String

Pattern: `[\u0020-\uD7FF\uE000-\uFFFD\uD800\uDC00-\uDBFF\uDFFF\r\n\t]*`

Required: Yes

Cause

A simple description of what caused the scaling activity to happen.

Type: String

Pattern: `[\u0020-\uD7FF\uE000-\uFFFD\uD800\uDC00-\uDBFF\uDFFF\r\n\t]*`

Required: Yes

Description

A simple description of what action the scaling activity intends to accomplish.

Type: String

Pattern: `[\u0020-\uD7FF\uE000-\uFFFD\uD800\uDC00-\uDBFF\uDFFF\r\n\t]*`

Required: Yes

Details

The details about the scaling activity.

Type: String

Pattern: `[\u0020-\uD7FF\uE000-\uFFFD\uD800\uDC00-\uDBFF\uDFFF\r\n\t]*`

Required: No

EndTime

The Unix timestamp for when the scaling activity ended.

Type: Timestamp

Required: No

ResourceId

The identifier of the resource associated with the scaling activity. This string consists of the resource type and unique identifier.

- ECS service - The resource type is `service` and the unique identifier is the cluster name and service name. Example: `service/default/sample-webapp`.
- Spot Fleet - The resource type is `spot-fleet-request` and the unique identifier is the Spot Fleet request ID. Example: `spot-fleet-request/sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE`.

- EMR cluster - The resource type is `instancegroup` and the unique identifier is the cluster ID and instance group ID. Example: `instancegroup/j-2EEZNYKUA1NTV/ig-1791Y4E1L8YI0`.
- AppStream 2.0 fleet - The resource type is `fleet` and the unique identifier is the fleet name. Example: `fleet/sample-fleet`.
- DynamoDB table - The resource type is `table` and the unique identifier is the table name. Example: `table/my-table`.
- DynamoDB global secondary index - The resource type is `index` and the unique identifier is the index name. Example: `table/my-table/index/my-table-index`.
- Aurora DB cluster - The resource type is `cluster` and the unique identifier is the cluster name. Example: `cluster:my-db-cluster`.
- SageMaker endpoint variant - The resource type is `variant` and the unique identifier is the resource ID. Example: `endpoint/my-end-point/variant/KMeansClustering`.
- Custom resources are not supported with a resource type. This parameter must specify the `OutputValue` from the CloudFormation template stack used to access the resources. The unique identifier is defined by the service provider. More information is available in our GitHub repository.
- Amazon Comprehend document classification endpoint - The resource type and unique identifier are specified using the endpoint ARN. Example: `arn:aws:comprehend:us-west-2:123456789012:document-classifier-endpoint/EXAMPLE`.
- Amazon Comprehend entity recognizer endpoint - The resource type and unique identifier are specified using the endpoint ARN. Example: `arn:aws:comprehend:us-west-2:123456789012:entity-recognizer-endpoint/EXAMPLE`.
- Lambda provisioned concurrency - The resource type is `function` and the unique identifier is the function name with a function version or alias name suffix that is not `$LATEST`. Example: `function:my-function:prod` or `function:my-function:1`.
- Amazon Keyspaces table - The resource type is `table` and the unique identifier is the table name. Example: `keyspace/mykeyspace/table/mytable`.
- Amazon MSK cluster - The resource type and unique identifier are specified using the cluster ARN. Example: `arn:aws:kafka:us-east-1:123456789012:cluster/demo-cluster-1/6357e0b2-0e6a-4b86-a0b4-70df934c2e31-5`.
- Amazon ElastiCache replication group - The resource type is `replication-group` and the unique identifier is the replication group name. Example: `replication-group/mycluster`.
- Neptune cluster - The resource type is `cluster` and the unique identifier is the cluster name. Example: `cluster:mycluster`.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1600.

Pattern: `[\u0020-\uD7FF\uE000-\uFFFD\uD800\uDC00-\uDBFF\uDFFF\r\n\t]*`

Required: Yes

ScalableDimension

The scalable dimension. This string consists of the service namespace, resource type, and scaling property.

- `ecs:service:DesiredCount` - The desired task count of an ECS service.
- `elasticmapreduce:instancegroup:InstanceCount` - The instance count of an EMR Instance Group.
- `ec2:spot-fleet-request:TargetCapacity` - The target capacity of a Spot Fleet.
- `appstream:fleet:DesiredCapacity` - The desired capacity of an AppStream 2.0 fleet.
- `dynamodb:table:ReadCapacityUnits` - The provisioned read capacity for a DynamoDB table.
- `dynamodb:table:WriteCapacityUnits` - The provisioned write capacity for a DynamoDB table.

- `dynamodb:index:ReadCapacityUnits` - The provisioned read capacity for a DynamoDB global secondary index.
- `dynamodb:index:WriteCapacityUnits` - The provisioned write capacity for a DynamoDB global secondary index.
- `rds:cluster:ReadReplicaCount` - The count of Aurora Replicas in an Aurora DB cluster. Available for Aurora MySQL-compatible edition and Aurora PostgreSQL-compatible edition.
- `sagemaker:variant:DesiredInstanceCount` - The number of EC2 instances for an SageMaker model endpoint variant.
- `custom-resource:ResourceType:Property` - The scalable dimension for a custom resource provided by your own application or service.
- `comprehend:document-classifier-endpoint:DesiredInferenceUnits` - The number of inference units for an Amazon Comprehend document classification endpoint.
- `comprehend:entity-recognizer-endpoint:DesiredInferenceUnits` - The number of inference units for an Amazon Comprehend entity recognizer endpoint.
- `lambda:function:ProvisionedConcurrency` - The provisioned concurrency for a Lambda function.
- `cassandra:table:ReadCapacityUnits` - The provisioned read capacity for an Amazon Keyspaces table.
- `cassandra:table:WriteCapacityUnits` - The provisioned write capacity for an Amazon Keyspaces table.
- `kafka:broker-storage:VolumeSize` - The provisioned volume size (in GiB) for brokers in an Amazon MSK cluster.
- `elasticache:replication-group:NodeGroups` - The number of node groups for an Amazon ElastiCache replication group.
- `elasticache:replication-group:Replicas` - The number of replicas per node group for an Amazon ElastiCache replication group.
- `neptune:cluster:ReadReplicaCount` - The count of read replicas in an Amazon Neptune DB cluster.

Type: String

Valid Values: `ecs:service:DesiredCount | ec2:spot-fleet-request:TargetCapacity | elasticmapreduce:instancegroup:InstanceCount | appstream:fleet:DesiredCapacity | dynamodb:table:ReadCapacityUnits | dynamodb:table:WriteCapacityUnits | dynamodb:index:ReadCapacityUnits | dynamodb:index:WriteCapacityUnits | rds:cluster:ReadReplicaCount | sagemaker:variant:DesiredInstanceCount | custom-resource:ResourceType:Property | comprehend:document-classifier-endpoint:DesiredInferenceUnits | comprehend:entity-recognizer-endpoint:DesiredInferenceUnits | lambda:function:ProvisionedConcurrency | cassandra:table:ReadCapacityUnits | cassandra:table:WriteCapacityUnits | kafka:broker-storage:VolumeSize | elasticache:replication-group:NodeGroups | elasticache:replication-group:Replicas | neptune:cluster:ReadReplicaCount`

Required: Yes

ServiceNamespace

The namespace of the AWS service that provides the resource, or a `custom-resource`.

Type: String

Valid Values: `ecs | elasticmapreduce | ec2 | appstream | dynamodb | rds | sagemaker | custom-resource | comprehend | lambda | cassandra | kafka | elasticache | neptune`

Required: Yes

StartTime

The Unix timestamp for when the scaling activity began.

Type: Timestamp

Required: Yes

StatusCode

Indicates the status of the scaling activity.

Type: String

Valid Values: `Pending | InProgress | Successful | Overridden | Unfulfilled | Failed`

Required: Yes

StatusMessage

A simple message about the current status of the scaling activity.

Type: String

Pattern: `[\u0020-\uD7FF\uE000-\uFFFD\uD800\uDC00-\uDBFF\uDFFF\r\n\t]*`

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 Go
- AWS SDK for Java V2
- AWS SDK for Ruby V3

ScalingPolicy

Represents a scaling policy to use with Application Auto Scaling.

For more information about configuring scaling policies for a specific service, see Getting started with Application Auto Scaling in the *Application Auto Scaling User Guide*.

Contents

Alarms

The CloudWatch alarms associated with the scaling policy.

Type: Array of Alarm (p. 67) objects

Required: No

CreationTime

The Unix timestamp for when the scaling policy was created.

Type: Timestamp

Required: Yes

PolicyARN

The Amazon Resource Name (ARN) of the scaling policy.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1600.

Pattern: `[\u0020-\uD7FF\uE000-\uFFFD\uD800\uDC00-\uDBFF\uDFFF\r\n\t]*`

Required: Yes

PolicyName

The name of the scaling policy.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

Pattern: `\p{Print}+`

Required: Yes

PolicyType

The scaling policy type.

Type: String

Valid Values: `StepScaling | TargetTrackingScaling`

Required: Yes

ResourceId

The identifier of the resource associated with the scaling policy. This string consists of the resource type and unique identifier.

- ECS service - The resource type is `service` and the unique identifier is the cluster name and service name. Example: `service/default/sample-webapp`.
- Spot Fleet - The resource type is `spot-fleet-request` and the unique identifier is the Spot Fleet request ID. Example: `spot-fleet-request/sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE`.
- EMR cluster - The resource type is `instancegroup` and the unique identifier is the cluster ID and instance group ID. Example: `instancegroup/j-2EEZNYKUA1NTV/ig-1791Y4E1L8YI0`.
- AppStream 2.0 fleet - The resource type is `fleet` and the unique identifier is the fleet name. Example: `fleet/sample-fleet`.
- DynamoDB table - The resource type is `table` and the unique identifier is the table name. Example: `table/my-table`.
- DynamoDB global secondary index - The resource type is `index` and the unique identifier is the index name. Example: `table/my-table/index/my-table-index`.
- Aurora DB cluster - The resource type is `cluster` and the unique identifier is the cluster name. Example: `cluster:my-db-cluster`.
- SageMaker endpoint variant - The resource type is `variant` and the unique identifier is the resource ID. Example: `endpoint/my-end-point/variant/KMeansClustering`.
- Custom resources are not supported with a resource type. This parameter must specify the `OutputValue` from the CloudFormation template stack used to access the resources. The unique identifier is defined by the service provider. More information is available in our GitHub repository.
- Amazon Comprehend document classification endpoint - The resource type and unique identifier are specified using the endpoint ARN. Example: `arn:aws:comprehend:us-west-2:123456789012:document-classifier-endpoint/EXAMPLE`.
- Amazon Comprehend entity recognizer endpoint - The resource type and unique identifier are specified using the endpoint ARN. Example: `arn:aws:comprehend:us-west-2:123456789012:entity-recognizer-endpoint/EXAMPLE`.
- Lambda provisioned concurrency - The resource type is `function` and the unique identifier is the function name with a function version or alias name suffix that is not `$LATEST`. Example: `function:my-function:prod` or `function:my-function:1`.
- Amazon Keyspaces table - The resource type is `table` and the unique identifier is the table name. Example: `keyspace/mykeyspace/table/mytable`.
- Amazon MSK cluster - The resource type and unique identifier are specified using the cluster ARN. Example: `arn:aws:kafka:us-east-1:123456789012:cluster/demo-cluster-1/6357e0b2-0e6a-4b86-a0b4-70df934c2e31-5`.
- Amazon ElastiCache replication group - The resource type is `replication-group` and the unique identifier is the replication group name. Example: `replication-group/mycluster`.
- Neptune cluster - The resource type is `cluster` and the unique identifier is the cluster name. Example: `cluster:mycluster`.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1600.

Pattern: `[\u0020-\uD7FF\uE000-\uFFFD\uD800\uDC00-\uDBFF\uDFFF\r\n\t]*`

Required: Yes

ScalableDimension

The scalable dimension. This string consists of the service namespace, resource type, and scaling property.

- `ecs:service:DesiredCount` - The desired task count of an ECS service.
- `elasticmapreduce:instancegroup:InstanceCount` - The instance count of an EMR Instance Group.

- ec2:spot-fleet-request:TargetCapacity - The target capacity of a Spot Fleet.
- appstream:fleet:DesiredCapacity - The desired capacity of an AppStream 2.0 fleet.
- dynamodb:table:ReadCapacityUnits - The provisioned read capacity for a DynamoDB table.
- dynamodb:table:WriteCapacityUnits - The provisioned write capacity for a DynamoDB table.
- dynamodb:index:ReadCapacityUnits - The provisioned read capacity for a DynamoDB global secondary index.
- dynamodb:index:WriteCapacityUnits - The provisioned write capacity for a DynamoDB global secondary index.
- rds:cluster:ReadReplicaCount - The count of Aurora Replicas in an Aurora DB cluster. Available for Aurora MySQL-compatible edition and Aurora PostgreSQL-compatible edition.
- sagemaker:variant:DesiredInstanceCount - The number of EC2 instances for an SageMaker model endpoint variant.
- custom-resource:ResourceType:Property - The scalable dimension for a custom resource provided by your own application or service.
- comprehend:document-classifier-endpoint:DesiredInferenceUnits - The number of inference units for an Amazon Comprehend document classification endpoint.
- comprehend:entity-recognizer-endpoint:DesiredInferenceUnits - The number of inference units for an Amazon Comprehend entity recognizer endpoint.
- lambda:function:ProvisionedConcurrency - The provisioned concurrency for a Lambda function.
- cassandra:table:ReadCapacityUnits - The provisioned read capacity for an Amazon Keyspaces table.
- cassandra:table:WriteCapacityUnits - The provisioned write capacity for an Amazon Keyspaces table.
- kafka:broker-storage:VolumeSize - The provisioned volume size (in GiB) for brokers in an Amazon MSK cluster.
- elasticache:replication-group:NodeGroups - The number of node groups for an Amazon ElastiCache replication group.
- elasticache:replication-group:Replicas - The number of replicas per node group for an Amazon ElastiCache replication group.
- neptune:cluster:ReadReplicaCount - The count of read replicas in an Amazon Neptune DB cluster.

Type: String

Valid Values: ecs:service:DesiredCount | ec2:spot-fleet-request:TargetCapacity | elasticmapreduce:instancegroup:InstanceCount | appstream:fleet:DesiredCapacity | dynamodb:table:ReadCapacityUnits | dynamodb:table:WriteCapacityUnits | dynamodb:index:ReadCapacityUnits | dynamodb:index:WriteCapacityUnits | rds:cluster:ReadReplicaCount | sagemaker:variant:DesiredInstanceCount | custom-resource:ResourceType:Property | comprehend:document-classifier-endpoint:DesiredInferenceUnits | comprehend:entity-recognizer-endpoint:DesiredInferenceUnits | lambda:function:ProvisionedConcurrency | cassandra:table:ReadCapacityUnits | cassandra:table:WriteCapacityUnits | kafka:broker-storage:VolumeSize | elasticache:replication-group:NodeGroups | elasticache:replication-group:Replicas | neptune:cluster:ReadReplicaCount

Required: Yes

ServiceNamespace

The namespace of the AWS service that provides the resource, or a custom-resource.

Type: String

Valid Values: `ecs | elasticmapreduce | ec2 | appstream | dynamodb | rds | sagemaker | custom-resource | comprehend | lambda | cassandra | kafka | elasticache | neptune`

Required: Yes

StepScalingPolicyConfiguration

A step scaling policy.

Type: StepScalingPolicyConfiguration (p. 93) object

Required: No

TargetTrackingScalingPolicyConfiguration

A target tracking scaling policy.

Type: TargetTrackingScalingPolicyConfiguration (p. 96) object

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 Go
- AWS SDK for Java V2
- AWS SDK for Ruby V3

ScheduledAction

Represents a scheduled action.

Contents

CreationTime

The date and time that the scheduled action was created.

Type: Timestamp

Required: Yes

EndTime

The date and time that the action is scheduled to end, in UTC.

Type: Timestamp

Required: No

ResourceId

The identifier of the resource associated with the scaling policy. This string consists of the resource type and unique identifier.

- ECS service - The resource type is `service` and the unique identifier is the cluster name and service name. Example: `service/default/sample-webapp`.
- Spot Fleet - The resource type is `spot-fleet-request` and the unique identifier is the Spot Fleet request ID. Example: `spot-fleet-request/sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE`.
- EMR cluster - The resource type is `instancegroup` and the unique identifier is the cluster ID and instance group ID. Example: `instancegroup/j-2EEZNYKUA1NTV/ig-1791Y4E1L8YI0`.
- AppStream 2.0 fleet - The resource type is `fleet` and the unique identifier is the fleet name. Example: `fleet/sample-fleet`.
- DynamoDB table - The resource type is `table` and the unique identifier is the table name. Example: `table/my-table`.
- DynamoDB global secondary index - The resource type is `index` and the unique identifier is the index name. Example: `table/my-table/index/my-table-index`.
- Aurora DB cluster - The resource type is `cluster` and the unique identifier is the cluster name. Example: `cluster:my-db-cluster`.
- SageMaker endpoint variant - The resource type is `variant` and the unique identifier is the resource ID. Example: `endpoint/my-end-point/variant/KMeansClustering`.
- Custom resources are not supported with a resource type. This parameter must specify the `OutputValue` from the CloudFormation template stack used to access the resources. The unique identifier is defined by the service provider. More information is available in our GitHub repository.
- Amazon Comprehend document classification endpoint - The resource type and unique identifier are specified using the endpoint ARN. Example: `arn:aws:comprehend:us-west-2:123456789012:document-classifier-endpoint/EXAMPLE`.
- Amazon Comprehend entity recognizer endpoint - The resource type and unique identifier are specified using the endpoint ARN. Example: `arn:aws:comprehend:us-west-2:123456789012:entity-recognizer-endpoint/EXAMPLE`.
- Lambda provisioned concurrency - The resource type is `function` and the unique identifier is the function name with a function version or alias name suffix that is not `$LATEST`. Example: `function:my-function:prod` or `function:my-function:1`.

- Amazon Keyspaces table - The resource type is `table` and the unique identifier is the table name. Example: `keyspace/mykeyspace/table/mytable`.
- Amazon MSK cluster - The resource type and unique identifier are specified using the cluster ARN. Example: `arn:aws:kafka:us-east-1:123456789012:cluster/demo-cluster-1/6357e0b2-0e6a-4b86-a0b4-70df934c2e31-5`.
- Amazon ElastiCache replication group - The resource type is `replication-group` and the unique identifier is the replication group name. Example: `replication-group/mycluster`.
- Neptune cluster - The resource type is `cluster` and the unique identifier is the cluster name. Example: `cluster:mycluster`.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1600.

Pattern: `[\u0020-\uD7FF\uE000-\uFFFD\uD800\uDC00-\uDBFF\uDFFF\r\n\t]*`

Required: Yes

ScalableDimension

The scalable dimension. This string consists of the service namespace, resource type, and scaling property.

- `ecs:service:DesiredCount` - The desired task count of an ECS service.
- `elasticmapreduce:instancegroup:InstanceCount` - The instance count of an EMR Instance Group.
- `ec2:spot-fleet-request:TargetCapacity` - The target capacity of a Spot Fleet.
- `appstream:fleet:DesiredCapacity` - The desired capacity of an AppStream 2.0 fleet.
- `dynamodb:table:ReadCapacityUnits` - The provisioned read capacity for a DynamoDB table.
- `dynamodb:table:WriteCapacityUnits` - The provisioned write capacity for a DynamoDB table.
- `dynamodb:index:ReadCapacityUnits` - The provisioned read capacity for a DynamoDB global secondary index.
- `dynamodb:index:WriteCapacityUnits` - The provisioned write capacity for a DynamoDB global secondary index.
- `rds:cluster:ReadReplicaCount` - The count of Aurora Replicas in an Aurora DB cluster. Available for Aurora MySQL-compatible edition and Aurora PostgreSQL-compatible edition.
- `sagemaker:variant:DesiredInstanceCount` - The number of EC2 instances for an SageMaker model endpoint variant.
- `custom-resource:ResourceType:Property` - The scalable dimension for a custom resource provided by your own application or service.
- `comprehend:document-classifier-endpoint:DesiredInferenceUnits` - The number of inference units for an Amazon Comprehend document classification endpoint.
- `comprehend:entity-recognizer-endpoint:DesiredInferenceUnits` - The number of inference units for an Amazon Comprehend entity recognizer endpoint.
- `lambda:function:ProvisionedConcurrency` - The provisioned concurrency for a Lambda function.
- `cassandra:table:ReadCapacityUnits` - The provisioned read capacity for an Amazon Keyspaces table.
- `cassandra:table:WriteCapacityUnits` - The provisioned write capacity for an Amazon Keyspaces table.
- `kafka:broker-storage:VolumeSize` - The provisioned volume size (in GiB) for brokers in an Amazon MSK cluster.
- `elasticache:replication-group:NodeGroups` - The number of node groups for an Amazon ElastiCache replication group.

- `elasticache:replication-group:Replicas` - The number of replicas per node group for an Amazon ElastiCache replication group.
- `neptune:cluster:ReadReplicaCount` - The count of read replicas in an Amazon Neptune DB cluster.

Type: String

Valid Values: `ecs:service:DesiredCount | ec2:spot-fleet-request:TargetCapacity | elasticmapreduce:instancegroup:InstanceCount | appstream:fleet:DesiredCapacity | dynamodb:table:ReadCapacityUnits | dynamodb:table:WriteCapacityUnits | dynamodb:index:ReadCapacityUnits | dynamodb:index:WriteCapacityUnits | rds:cluster:ReadReplicaCount | sagemaker:variant:DesiredInstanceCount | custom-resource:ResourceType:Property | comprehend:document-classifier-endpoint:DesiredInferenceUnits | comprehend:entity-recognizer-endpoint:DesiredInferenceUnits | lambda:function:ProvisionedConcurrency | cassandra:table:ReadCapacityUnits | cassandra:table:WriteCapacityUnits | kafka:broker-storage:VolumeSize | elasticache:replication-group:NodeGroups | elasticache:replication-group:Replicas | neptune:cluster:ReadReplicaCount`

Required: No

ScalableTargetAction

The new minimum and maximum capacity. You can set both values or just one. At the scheduled time, if the current capacity is below the minimum capacity, Application Auto Scaling scales out to the minimum capacity. If the current capacity is above the maximum capacity, Application Auto Scaling scales in to the maximum capacity.

Type: ScalableTargetAction (p. 77) object

Required: No

Schedule

The schedule for this action. The following formats are supported:

- At expressions - "`at(`*`yyyy-mm-dd`*`T`*`hh:mm:ss`*`)`"
- Rate expressions - "`rate(`*`value unit`*`)`"
- Cron expressions - "`cron(`*`fields`*`)`"

At expressions are useful for one-time schedules. Cron expressions are useful for scheduled actions that run periodically at a specified date and time, and rate expressions are useful for scheduled actions that run at a regular interval.

At and cron expressions use Universal Coordinated Time (UTC) by default.

The cron format consists of six fields separated by white spaces: [Minutes] [Hours] [Day_of_Month] [Month] [Day_of_Week] [Year].

For rate expressions, *value* is a positive integer and *unit* is `minute` | `minutes` | `hour` | `hours` | `day` | `days`.

For more information and examples, see Example scheduled actions for Application Auto Scaling in the *Application Auto Scaling User Guide*.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1600.

Pattern: `[\u0020-\uD7FF\uE000-\uFFFD\uD800\uDC00-\uDBFF\uDFFF\r\n\t]*`

Required: Yes

ScheduledActionARN

The Amazon Resource Name (ARN) of the scheduled action.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1600.

Pattern: `[\u0020-\uD7FF\uE000-\uFFFD\uD800\uDC00-\uDBFF\uDFFF\r\n\t]*`

Required: Yes

ScheduledActionName

The name of the scheduled action.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

Pattern: `(?!((^[ ]+.*)|(.*([\u0000-\u001f]|[\u007f-\u009f]|[:/|])+.*)|(.*[ ]+$))).+`

Required: Yes

ServiceNamespace

The namespace of the AWS service that provides the resource, or a `custom-resource`.

Type: String

Valid Values: `ecs | elasticmapreduce | ec2 | appstream | dynamodb | rds | sagemaker | custom-resource | comprehend | lambda | cassandra | kafka | elasticache | neptune`

Required: Yes

StartTime

The date and time that the action is scheduled to begin, in UTC.

Type: Timestamp

Required: No

Timezone

The time zone used when referring to the date and time of a scheduled action, when the scheduled action uses an at or cron expression.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1600.

Pattern: `[\u0020-\uD7FF\uE000-\uFFFD\uD800\uDC00-\uDBFF\uDFFF\r\n\t]*`

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 Go
- AWS SDK for Java V2
- AWS SDK for Ruby V3

StepAdjustment

Represents a step adjustment for a StepScalingPolicyConfiguration. Describes an adjustment based on the difference between the value of the aggregated CloudWatch metric and the breach threshold that you've defined for the alarm.

For the following examples, suppose that you have an alarm with a breach threshold of 50:

- To trigger the adjustment when the metric is greater than or equal to 50 and less than 60, specify a lower bound of 0 and an upper bound of 10.
- To trigger the adjustment when the metric is greater than 40 and less than or equal to 50, specify a lower bound of -10 and an upper bound of 0.

There are a few rules for the step adjustments for your step policy:

- The ranges of your step adjustments can't overlap or have a gap.
- At most one step adjustment can have a null lower bound. If one step adjustment has a negative lower bound, then there must be a step adjustment with a null lower bound.
- At most one step adjustment can have a null upper bound. If one step adjustment has a positive upper bound, then there must be a step adjustment with a null upper bound.
- The upper and lower bound can't be null in the same step adjustment.

Contents

MetricIntervalLowerBound

The lower bound for the difference between the alarm threshold and the CloudWatch metric. If the metric value is above the breach threshold, the lower bound is inclusive (the metric must be greater than or equal to the threshold plus the lower bound). Otherwise, it is exclusive (the metric must be greater than the threshold plus the lower bound). A null value indicates negative infinity.

Type: Double

Required: No

MetricIntervalUpperBound

The upper bound for the difference between the alarm threshold and the CloudWatch metric. If the metric value is above the breach threshold, the upper bound is exclusive (the metric must be less than the threshold plus the upper bound). Otherwise, it is inclusive (the metric must be less than or equal to the threshold plus the upper bound). A null value indicates positive infinity.

The upper bound must be greater than the lower bound.

Type: Double

Required: No

ScalingAdjustment

The amount by which to scale, based on the specified adjustment type. A positive value adds to the current capacity while a negative number removes from the current capacity. For exact capacity, you must specify a positive value.

Type: Integer

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 Go
- AWS SDK for Java V2
- AWS SDK for Ruby V3

StepScalingPolicyConfiguration

Represents a step scaling policy configuration to use with Application Auto Scaling.

Contents

AdjustmentType

Specifies how the `ScalingAdjustment` value in a StepAdjustment is interpreted (for example, an absolute number or a percentage). The valid values are `ChangeInCapacity`, `ExactCapacity`, and `PercentChangeInCapacity`.

`AdjustmentType` is required if you are adding a new step scaling policy configuration.

Type: String

Valid Values: `ChangeInCapacity | PercentChangeInCapacity | ExactCapacity`

Required: No

Cooldown

The amount of time, in seconds, to wait for a previous scaling activity to take effect.

With scale-out policies, the intention is to continuously (but not excessively) scale out. After Application Auto Scaling successfully scales out using a step scaling policy, it starts to calculate the cooldown time. The scaling policy won't increase the desired capacity again unless either a larger scale out is triggered or the cooldown period ends. While the cooldown period is in effect, capacity added by the initiating scale-out activity is calculated as part of the desired capacity for the next scale-out activity. For example, when an alarm triggers a step scaling policy to increase the capacity by 2, the scaling activity completes successfully, and a cooldown period starts. If the alarm triggers again during the cooldown period but at a more aggressive step adjustment of 3, the previous increase of 2 is considered part of the current capacity. Therefore, only 1 is added to the capacity.

With scale-in policies, the intention is to scale in conservatively to protect your application's availability, so scale-in activities are blocked until the cooldown period has expired. However, if another alarm triggers a scale-out activity during the cooldown period after a scale-in activity, Application Auto Scaling scales out the target immediately. In this case, the cooldown period for the scale-in activity stops and doesn't complete.

Application Auto Scaling provides a default value of 600 for Amazon ElastiCache replication groups and a default value of 300 for the following scalable targets:

- AppStream 2.0 fleets
- Aurora DB clusters
- ECS services
- EMR clusters
- Neptune clusters
- SageMaker endpoint variants
- Spot Fleets
- Custom resources

For all other scalable targets, the default value is 0:

- Amazon Comprehend document classification and entity recognizer endpoints
- DynamoDB tables and global secondary indexes
- Amazon Keyspaces tables

- Lambda provisioned concurrency
- Amazon MSK broker storage

Type: Integer

Required: No

MetricAggregationType

The aggregation type for the CloudWatch metrics. Valid values are `Minimum`, `Maximum`, and `Average`. If the aggregation type is null, the value is treated as `Average`.

Type: String

Valid Values: `Average | Minimum | Maximum`

Required: No

MinAdjustmentMagnitude

The minimum value to scale by when the adjustment type is `PercentChangeInCapacity`. For example, suppose that you create a step scaling policy to scale out an Amazon ECS service by 25 percent and you specify a `MinAdjustmentMagnitude` of 2. If the service has 4 tasks and the scaling policy is performed, 25 percent of 4 is 1. However, because you specified a `MinAdjustmentMagnitude` of 2, Application Auto Scaling scales out the service by 2 tasks.

Type: Integer

Required: No

StepAdjustments

A set of adjustments that enable you to scale based on the size of the alarm breach.

At least one step adjustment is required if you are adding a new step scaling policy configuration.

Type: Array of StepAdjustment (p. 91) 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 Go
- AWS SDK for Java V2
- AWS SDK for Ruby V3

SuspendedState

Specifies whether the scaling activities for a scalable target are in a suspended state.

Contents

DynamicScalingInSuspended

Whether scale in by a target tracking scaling policy or a step scaling policy is suspended. Set the value to `true` if you don't want Application Auto Scaling to remove capacity when a scaling policy is triggered. The default is `false`.

Type: Boolean

Required: No

DynamicScalingOutSuspended

Whether scale out by a target tracking scaling policy or a step scaling policy is suspended. Set the value to `true` if you don't want Application Auto Scaling to add capacity when a scaling policy is triggered. The default is `false`.

Type: Boolean

Required: No

ScheduledScalingSuspended

Whether scheduled scaling is suspended. Set the value to `true` if you don't want Application Auto Scaling to add or remove capacity by initiating scheduled actions. The default is `false`.

Type: Boolean

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 Go
- AWS SDK for Java V2
- AWS SDK for Ruby V3

TargetTrackingScalingPolicyConfiguration

Represents a target tracking scaling policy configuration to use with Application Auto Scaling.

Contents

CustomizedMetricSpecification

A customized metric. You can specify either a predefined metric or a customized metric.

Type: CustomizedMetricSpecification (p. 68) object

Required: No

DisableScaleIn

Indicates whether scale in by the target tracking scaling policy is disabled. If the value is `true`, scale in is disabled and the target tracking scaling policy won't remove capacity from the scalable target. Otherwise, scale in is enabled and the target tracking scaling policy can remove capacity from the scalable target. The default value is `false`.

Type: Boolean

Required: No

PredefinedMetricSpecification

A predefined metric. You can specify either a predefined metric or a customized metric.

Type: PredefinedMetricSpecification (p. 71) object

Required: No

ScaleInCooldown

The amount of time, in seconds, after a scale-in activity completes before another scale-in activity can start.

With the *scale-in cooldown period*, the intention is to scale in conservatively to protect your application’s availability, so scale-in activities are blocked until the cooldown period has expired. However, if another alarm triggers a scale-out activity during the scale-in cooldown period, Application Auto Scaling scales out the target immediately. In this case, the scale-in cooldown period stops and doesn't complete.

Application Auto Scaling provides a default value of 600 for Amazon ElastiCache replication groups and a default value of 300 for the following scalable targets:

- AppStream 2.0 fleets
- Aurora DB clusters
- ECS services
- EMR clusters
- Neptune clusters
- SageMaker endpoint variants
- Spot Fleets
- Custom resources

For all other scalable targets, the default value is 0:

- Amazon Comprehend document classification and entity recognizer endpoints
- DynamoDB tables and global secondary indexes

- Amazon Keyspaces tables
- Lambda provisioned concurrency
- Amazon MSK broker storage

Type: Integer

Required: No

ScaleOutCooldown

The amount of time, in seconds, to wait for a previous scale-out activity to take effect.

With the *scale-out cooldown period*, the intention is to continuously (but not excessively) scale out. After Application Auto Scaling successfully scales out using a target tracking scaling policy, it starts to calculate the cooldown time. The scaling policy won't increase the desired capacity again unless either a larger scale out is triggered or the cooldown period ends. While the cooldown period is in effect, the capacity added by the initiating scale-out activity is calculated as part of the desired capacity for the next scale-out activity.

Application Auto Scaling provides a default value of 600 for Amazon ElastiCache replication groups and a default value of 300 for the following scalable targets:

- AppStream 2.0 fleets
- Aurora DB clusters
- ECS services
- EMR clusters
- Neptune clusters
- SageMaker endpoint variants
- Spot Fleets
- Custom resources

For all other scalable targets, the default value is 0:

- Amazon Comprehend document classification and entity recognizer endpoints
- DynamoDB tables and global secondary indexes
- Amazon Keyspaces tables
- Lambda provisioned concurrency
- Amazon MSK broker storage

Type: Integer

Required: No

TargetValue

The target value for the metric. Although this property accepts numbers of type Double, it won't accept values that are either too small or too large. Values must be in the range of -2^360 to 2^360. The value must be a valid number based on the choice of metric. For example, if the metric is CPU utilization, then the target value is a percent value that represents how much of the CPU can be used before scaling out.

Type: Double

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 Go
- 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 Signature Version 4 Signing Process in the *Amazon Web Services General Reference*.

Action

The action to be performed.

Type: string

Required: Yes

Version

The API version that the request is written for, expressed in the format YYYY-MM-DD.

Type: string

Required: Yes

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 Task 2: Create a String to Sign for Signature Version 4 in the *Amazon Web Services General Reference*.

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 Handling Dates in Signature Version 4 in the *Amazon Web Services General Reference*.

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 Security Token Service, go to AWS Services That Work with IAM in the *IAM User Guide*.

Condition: If you're using temporary security credentials from the AWS Security Token Service, 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 Task 1: Create a Canonical Request For Signature Version 4 in the *Amazon Web Services General Reference*.

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 Errors

This section lists the errors common to the API actions of all AWS services. For errors specific to an API action for this service, see the topic for that API action.

AccessDeniedException

You do not have sufficient access to perform this action.

HTTP Status Code: 400

IncompleteSignature

The request signature does not conform to AWS standards.

HTTP Status Code: 400

InternalFailure

The request processing has failed because of an unknown error, exception or failure.

HTTP Status Code: 500

InvalidAction

The action or operation requested is invalid. Verify that the action is typed correctly.

HTTP Status Code: 400

InvalidClientTokenId

The X.509 certificate or AWS access key ID provided does not exist in our records.

HTTP Status Code: 403

InvalidParameterCombination

Parameters that must not be used together were used together.

HTTP Status Code: 400

InvalidParameterValue

An invalid or out-of-range value was supplied for the input parameter.

HTTP Status Code: 400

InvalidQueryParameter

The AWS query string is malformed or does not adhere to AWS standards.

HTTP Status Code: 400

MalformedQueryString

The query string contains a syntax error.

HTTP Status Code: 404

MissingAction

The request is missing an action or a required parameter.

HTTP Status Code: 400

MissingAuthenticationToken

The request must contain either a valid (registered) AWS access key ID or X.509 certificate.

HTTP Status Code: 403

MissingParameter

A required parameter for the specified action is not supplied.

HTTP Status Code: 400

NotAuthorized

You do not have permission to perform this action.

HTTP Status Code: 400

OptInRequired

The AWS access key ID needs a subscription for the service.

HTTP Status Code: 403

RequestExpired

The request reached the service more than 15 minutes after the date stamp on the request or more than 15 minutes after the request expiration date (such as for pre-signed URLs), or the date stamp on the request is more than 15 minutes in the future.

HTTP Status Code: 400

ServiceUnavailable

The request has failed due to a temporary failure of the server.

HTTP Status Code: 503

ThrottlingException

The request was denied due to request throttling.

HTTP Status Code: 400

ValidationError

The input fails to satisfy the constraints specified by an AWS service.

HTTP Status Code: 400

Logging Application Auto Scaling API Calls with AWS CloudTrail

Application Auto Scaling is integrated with AWS CloudTrail, a service that provides a record of actions taken by a user, role, or an AWS service using the Application Auto Scaling API. CloudTrail captures all API calls for Application Auto Scaling as events. The calls captured include calls from the AWS Management Console and code calls to the Application Auto Scaling API. If you create a trail, you can enable continuous delivery of CloudTrail events to an Amazon S3 bucket, including events for Application Auto Scaling. If you don't configure a trail, you can still view the most recent events in the CloudTrail console in **Event history**. Using the information collected by CloudTrail, you can determine the request that was made to Application Auto Scaling, the IP address from which the request was made, who made the request, when it was made, and additional details.

To learn more about CloudTrail, see the AWS CloudTrail User Guide.

Application Auto Scaling Information in CloudTrail

CloudTrail is enabled on your AWS account when you create the account. When Application Auto Scaling activity occurs, that activity is recorded in a CloudTrail event along with other AWS service events in **Event history**. You can view, search, and download recent events in your AWS account. For more information, see Viewing Events with CloudTrail Event History.

For an ongoing record of events in your AWS account, including events for Application Auto Scaling, create a trail. A *trail* enables CloudTrail to deliver log files to an Amazon S3 bucket. By default, when you create a trail in the console, the trail applies to all AWS Regions. The trail logs events from all Regions in the AWS partition and delivers the log files to the Amazon S3 bucket that you specify. Additionally, you can configure other Amazon Web Services to further analyze and act upon the event data collected in CloudTrail logs. For more information, see the following:

- Overview for Creating a Trail
- CloudTrail Supported Services and Integrations
- Configuring Amazon SNS Notifications for CloudTrail
- Receiving CloudTrail Log Files from Multiple Regions and Receiving CloudTrail Log Files from Multiple Accounts

All Application Auto Scaling actions are logged by CloudTrail and are documented in the Application Auto Scaling API Reference. For example, calls to the `PutScalingPolicy`, `DeleteScalingPolicy`, and `DescribeScalingPolicies` actions generate entries in the CloudTrail log files.

Every event or log entry contains information about who generated the request. The identity information helps you determine the following:

- Whether the request was made with root or AWS Identity and Access Management (IAM) user credentials.
- Whether the request was made with temporary security credentials for a role or federated user.
- Whether the request was made by another AWS service.

For more information, see the CloudTrail userIdentity Element.

Understanding Application Auto Scaling Log File Entries

A trail is a configuration that enables delivery of events as log files to an Amazon S3 bucket that you specify. CloudTrail log files contain one or more log entries. An event represents a single request from any source and includes information about the requested action, the date and time of the action, request parameters, and so on. CloudTrail log files aren't an ordered stack trace of the public API calls, so they don't appear in any specific order.

The following example shows a CloudTrail log entry that demonstrates the `DescribeScalableTargets` action.

```
{
    "eventVersion": "1.05",
    "userIdentity": {
        "type": "Root",
        "principalId": "123456789012",
        "arn": "arn:aws:iam::123456789012:root",
        "accountId": "123456789012",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "sessionContext": {
            "attributes": {
                "mfaAuthenticated": "false",
                "creationDate": "2018-08-21T17:05:42Z"
            }
        }
    },
    "eventTime": "2018-08-16T23:20:32Z",
    "eventSource": "autoscaling.amazonaws.com",
    "eventName": "DescribeScalableTargets",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "72.21.196.68",
    "userAgent": "EC2 Spot Console",
    "requestParameters": {
        "serviceNamespace": "ec2",
        "scalableDimension": "ec2:spot-fleet-request:TargetCapacity",
        "resourceIds": [
            "spot-fleet-request/sfr-05ceaf79-3ba2-405d-e87b-612857f1357a"
        ]
    },
    "responseElements": null,
    "additionalEventData": {
        "service": "application-autoscaling"
    },
    "requestID": "0737e2ea-fb2d-11e3-bfd8-99133058e7bb",
    "eventID": "3fcfb182-98f8-4744-bd45-b38835ab61cb",
    "eventType": "AwsApiCall",
    "recipientAccountId": "123456789012"
}
```