

Amazon API Gateway



Amazon API Gateway: API Reference

Table of Contents

Amazon API Gateway WebSocket and HTTP API Reference	1
Resources	2
AccessLogSettings	3
URI	3
HTTP methods	3
Schemas	4
Properties	5
Api	6
URI	6
HTTP methods	6
Schemas	10
Properties	13
ApiMapping	23
URI	23
HTTP methods	23
Schemas	26
Properties	27
ApiMappings	30
URI	30
HTTP methods	30
Schemas	32
Properties	34
Apis	37
URI	37
HTTP methods	37
Schemas	41
Properties	44
Authorizer	55
URI	55
HTTP methods	55
Schemas	57
Properties	59
Authorizers	68
URI	68

HTTP methods	68
Schemas	70
Properties	72
AuthorizersCache	81
URI	81
HTTP methods	82
Schemas	83
Properties	83
Cors	84
URI	84
HTTP methods	84
Schemas	85
Properties	86
Deployment	87
URI	87
HTTP methods	87
Schemas	89
Properties	90
Deployments	93
URI	94
HTTP methods	94
Schemas	96
Properties	97
DomainName	101
URI	101
HTTP methods	102
Schemas	104
Properties	106
DomainNames	114
URI	114
HTTP methods	114
Schemas	116
Properties	119
ExportedAPI	128
URI	128
HTTP methods	129

Schemas	130
Properties	131
Integration	132
URI	133
HTTP methods	133
Schemas	135
Properties	137
IntegrationResponse	153
URI	153
HTTP methods	153
Schemas	156
Properties	157
IntegrationResponses	163
URI	163
HTTP methods	163
Schemas	165
Properties	167
Integrations	174
URI	174
HTTP methods	174
Schemas	176
Properties	179
Model	195
URI	195
HTTP methods	195
Schemas	198
Properties	199
Models	203
URI	203
HTTP methods	203
Schemas	205
Properties	207
ModelTemplate	211
URI	211
HTTP methods	211
Schemas	212

Properties	212
Route	214
URI	214
HTTP methods	214
Schemas	216
Properties	218
RouteRequestParameter	225
URI	226
HTTP methods	226
Schemas	227
Properties	227
RouteResponse	228
URI	228
HTTP methods	228
Schemas	231
Properties	232
RouteResponses	236
URI	236
HTTP methods	237
Schemas	239
Properties	241
Routes	245
URI	245
HTTP methods	245
Schemas	247
Properties	250
RouteSettings	258
URI	258
HTTP methods	258
Schemas	259
Properties	260
Stage	261
URI	261
HTTP methods	261
Schemas	264
Properties	266

Stages	274
URI	274
HTTP methods	275
Schemas	277
Properties	280
Tags	289
URI	289
HTTP methods	289
Schemas	292
Properties	293
VPCLink	295
URI	295
HTTP methods	295
Schemas	298
Properties	299
VPCLinks	303
URI	303
HTTP methods	303
Schemas	305
Properties	306
Operations	312
Document History	318
Amazon Glossary	319

Amazon API Gateway WebSocket and HTTP API Reference

Welcome to the Amazon API Gateway WebSocket and HTTP API Reference. For a description of Amazon API Gateway and how it works, see the [API Gateway Developer Guide](#).

Resources

The REST API includes the following resources.

Topics

- [AccessLogSettings](#)
- [Api](#)
- [ApiMapping](#)
- [ApiMappings](#)
- [Apis](#)
- [Authorizer](#)
- [Authorizers](#)
- [AuthorizersCache](#)
- [Cors](#)
- [Deployment](#)
- [Deployments](#)
- [DomainName](#)
- [DomainNames](#)
- [ExportedAPI](#)
- [Integration](#)
- [IntegrationResponse](#)
- [IntegrationResponses](#)
- [Integrations](#)
- [Model](#)
- [Models](#)
- [ModelTemplate](#)
- [Route](#)
- [RouteRequestParameter](#)
- [RouteResponse](#)

- [RouteResponses](#)
- [Routes](#)
- [RouteSettings](#)
- [Stage](#)
- [Stages](#)
- [Tags](#)
- [VPCLink](#)
- [VPCLinks](#)

AccessLogSettings

Represents the access log settings for a stage.

URI

`/v2/apis/apiId/stages/stageName/accesslogsettings`

HTTP methods

DELETE

Operation ID: DeleteAccessLogSettings

Deletes the AccessLogSettings for a Stage. To disable access logging for a Stage, delete its AccessLogSettings.

Path parameters

Name	Type	Required	Description
<i>stageName</i>	String	True	The stage name. Stage names can contain only alphanumeric characters, hyphens, and underscores, or be \$default.

Name	Type	Required	Description
<i>apiId</i>	String	True	The API identifier.

Responses

Status code	Response model	Description
204	None	The request has succeeded , and there is no additional content to send in the response payload body.
404	NotFoundException	The resource specified in the request was not found.
429	LimitExceededException	The client is sending more than the allowed number of requests per unit of time.

Schemas

Response bodies

NotFoundException schema

```
{  
  "message": "string",  
  "resourceType": "string"  
}
```

LimitExceededException schema

```
{  
  "message": "string",
```

```
"limitType": "string"  
}
```

Properties

LimitExceededException

A limit has been exceeded. See the accompanying error message for details.

message

Describes the error encountered.

Type: string

Required: False

limitType

The limit type.

Type: string

Required: False

NotFoundException

The resource specified in the request was not found. See the message field for more information.

message

Describes the error encountered.

Type: string

Required: False

resourceType

The resource type.

Type: string

Required: False

Api

Represents an API.

URI

/v2/apis/*apiId*

HTTP methods

GET

Operation ID: GetApi

Gets an Api resource.

Path parameters

Name	Type	Required	Description
<i>apiId</i>	String	True	The API identifier.

Responses

Status code	Response model	Description
200	Api	Success
404	NotFoundException	The resource specified in the request was not found.
429	LimitExceededException	The client is sending more than the allowed number of requests per unit of time.

PUT

Operation ID: ReimportApi

Puts an Api resource.

Path parameters

Name	Type	Required	Description
<i>apiId</i>	String	True	The API identifier.

Query parameters

Name	Type	Required	Description
failOnWarnings	String	False	Specifies whether to rollback the API creation when a warning is encountered. By default, API creation continues if a warning is encountered.
basepath	String	False	Specifies how to interpret the base path of the API during import. Valid values are ignore, prepend, and split. The default value is ignore. To learn more, see Set the OpenAPI basePath Property . Supported only for HTTP APIs.

Responses

Status code	Response model	Description
201	Api	The request has succeeded and has resulted in the creation of a resource.
400	BadRequestException	One of the parameters in the request is invalid.
404	NotFoundException	The resource specified in the request was not found.
409	ConflictException	The resource already exists.
429	LimitExceededException	The client is sending more than the allowed number of requests per unit of time.

DELETE

Operation ID: DeleteApi

Deletes an Api resource.

Path parameters

Name	Type	Required	Description
<i>apiId</i>	String	True	The API identifier.

Responses

Status code	Response model	Description
204	None	The request has succeeded , and there is no additiona

Status code	Response model	Description
404	NotFoundException	The resource specified in the request was not found.
429	LimitExceededException	The client is sending more than the allowed number of requests per unit of time.

PATCH

Operation ID: UpdateApi

Updates an Api resource.

Path parameters

Name	Type	Required	Description
<i>apiId</i>	String	True	The API identifier.

Responses

Status code	Response model	Description
200	Api	Success
400	BadRequestException	One of the parameters in the request is invalid.
404	NotFoundException	The resource specified in the request was not found.
409	ConflictException	The resource already exists.

Status code	Response model	Description
429	LimitExceededException	The client is sending more than the allowed number of requests per unit of time.

Schemas

Request bodies

PUT schema

```
{  
  "body": "string"  
}
```

PATCH schema

```
{  
  "name": "string",  
  "description": "string",  
  "version": "string",  
  "routeSelectionExpression": "string",  
  "apiKeySelectionExpression": "string",  
  "disableSchemaValidation": boolean,  
  "ipAddressType": enum,  
  "target": "string",  
  "credentialsArn": "string",  
  "routeKey": "string",  
  "corsConfiguration": {  
    "allowOrigins": [  
      "string"  
    ],  
    "allowCredentials": boolean,  
    "exposeHeaders": [  
      "string"  
    ],  
    "maxAge": integer,  
    "allowMethods": [  
      "string"  
    ]  
  }  
}
```

```
    ],
    "allowHeaders": [
      "string"
    ]
  },
  "disableExecuteApiEndpoint": boolean
}
```

Response bodies

Api schema

```
{
  "apiId": "string",
  "name": "string",
  "description": "string",
  "version": "string",
  "protocolType": enum,
  "ipAddressType": enum,
  "routeSelectionExpression": "string",
  "apiKeySelectionExpression": "string",
  "disableSchemaValidation": boolean,
  "warnings": [
    "string"
  ],
  "importInfo": [
    "string"
  ],
  "apiEndpoint": "string",
  "apiGatewayManaged": boolean,
  "createdDate": "string",
  "tags": {
  },
  "corsConfiguration": {
    "allowOrigins": [
      "string"
    ],
    "allowCredentials": boolean,
    "exposeHeaders": [
      "string"
    ],
    "maxAge": integer,
    "allowMethods": [
```

```
    "string"  
  ],  
  "allowHeaders": [  
    "string"  
  ]  
},  
"disableExecuteApiEndpoint": boolean  
}
```

BadRequestException schema

```
{  
  "message": "string"  
}
```

NotFoundException schema

```
{  
  "message": "string",  
  "resourceType": "string"  
}
```

ConflictException schema

```
{  
  "message": "string"  
}
```

LimitExceededException schema

```
{  
  "message": "string",  
  "limitType": "string"  
}
```

Properties

Api

Represents an API.

apiId

The API ID.

Type: string

Required: False

name

The name of the API.

Type: string

Required: True

description

The description of the API.

Type: string

Required: False

version

A version identifier for the API.

Type: string

Required: False

protocolType

The API protocol.

Type: [ProtocolType](#)

Required: True

ipAddressType

The IP address types that can invoke the API.

Type: [IpAddressType](#)

Required: False

routeSelectionExpression

The route selection expression for the API. For HTTP APIs, the `routeSelectionExpression` must be `${request.method} ${request.path}`. If not provided, this will be the default for HTTP APIs. This property is required for WebSocket APIs.

Type: string

Required: True

apiKeySelectionExpression

An API key selection expression. Supported only for WebSocket APIs. See [API Key Selection Expressions](#).

Type: string

Required: False

disableSchemaValidation

Avoid validating models when creating a deployment. Supported only for WebSocket APIs.

Type: boolean

Required: False

warnings

The warning messages reported when `failonwarnings` is turned on during API import.

Type: Array of type string

Required: False

importInfo

The validation information during API import. This may include particular properties of your OpenAPI definition which are ignored during import. Supported only for HTTP APIs.

Type: Array of type string

Required: False

apiEndpoint

The URI of the API, of the form {api-id}.execute-api.{region}.amazonaws.com. The stage name is typically appended to this URI to form a complete path to a deployed API stage.

Type: string

Required: False

Format: uri

apiGatewayManaged

Specifies whether an API is managed by API Gateway. You can't update or delete a managed API by using API Gateway. A managed API can be deleted only through the tooling or service that created it.

Type: boolean

Required: False

createdDate

The timestamp when the API was created.

Type: string

Required: False

Format: date-time

tags

A collection of tags associated with the API.

Type: [Tags](#)

Required: False

corsConfiguration

A CORS configuration. Supported only for HTTP APIs.

Type: [Cors](#)

Required: False

disableExecuteApiEndpoint

Specifies whether clients can invoke your API by using the default `execute-api` endpoint. By default, clients can invoke your API with the default `https://{api_id}.execute-api.{region}.amazonaws.com` endpoint. To require that clients use a custom domain name to invoke your API, disable the default endpoint.

Type: boolean

Required: False

BadRequestException

The request is not valid, for example, the input is incomplete or incorrect. See the accompanying error message for details.

message

Describes the error encountered.

Type: string

Required: False

ConflictException

The requested operation would cause a conflict with the current state of a service resource associated with the request. Resolve the conflict before retrying this request. See the accompanying error message for details.

message

Describes the error encountered.

Type: string

Required: False

Cors

Represents a CORS configuration. Supported only for HTTP APIs. See [Configuring CORS](#) for more information.

allowOrigins

Represents a collection of allowed origins. Supported only for HTTP APIs.

Type: Array of type string

Required: False

allowCredentials

Specifies whether credentials are included in the CORS request. Supported only for HTTP APIs.

Type: boolean

Required: False

exposeHeaders

Represents a collection of exposed headers. Supported only for HTTP APIs.

Type: Array of type string

Required: False

maxAge

The number of seconds that the browser should cache preflight request results. Supported only for HTTP APIs.

Type: integer

Required: False

allowMethods

Represents a collection of allowed HTTP methods. Supported only for HTTP APIs.

Type: Array of type string

Required: False

allowHeaders

Represents a collection of allowed headers. Supported only for HTTP APIs.

Type: Array of type string

Required: False

IpAddressType

The IP address types that can invoke your API or domain name.

ipv4

dualstack

LimitExceededException

A limit has been exceeded. See the accompanying error message for details.

message

Describes the error encountered.

Type: string

Required: False

limitType

The limit type.

Type: string

Required: False

NotFoundException

The resource specified in the request was not found. See the message field for more information.

message

Describes the error encountered.

Type: string

Required: False

resourceType

The resource type.

Type: string

Required: False

ProtocolType

Represents a protocol type.

WEBSOCKET

HTTP

ReimportApiInput

Overwrites the configuration of an existing API using the provided definition. Supported only for HTTP APIs.

body

The OpenAPI definition. Supported only for HTTP APIs.

Type: string

Required: True

Tags

Represents a collection of tags associated with the resource.

key-value pairs

Type: string

UpdateApiInput

Represents the input parameters for an UpdateApi request.

name

The name of the API.

Type: string

Required: False

description

The description of the API.

Type: string

Required: False

version

A version identifier for the API.

Type: string

Required: False

routeSelectionExpression

The route selection expression for the API. For HTTP APIs, the `routeSelectionExpression` must be `${request.method} ${request.path}`. If not provided, this will be the default for HTTP APIs. This property is required for WebSocket APIs.

Type: string

Required: False

apiKeySelectionExpression

An API key selection expression. Supported only for WebSocket APIs. See [API Key Selection Expressions](#).

Type: string

Required: False

disableSchemaValidation

Avoid validating models when creating a deployment. Supported only for WebSocket APIs.

Type: boolean

Required: False

ipAddressType

The IP address types that can invoke the API. Use `ipv4` to allow only IPv4 addresses to invoke your API, or use `dualstack` to allow both IPv4 and IPv6 addresses to invoke your API.

Type: [IpAddressType](#)

Required: False

target

This property is part of quick create. For HTTP integrations, specify a fully qualified URL. For Lambda integrations, specify a function ARN. The type of the integration will be `HTTP_PROXY` or `AWS_PROXY`, respectively. The value provided updates the integration URI and integration type. You can update a quick-created target, but you can't remove it from an API. Supported only for HTTP APIs.

Type: string

Required: False

credentialsArn

This property is part of quick create. It specifies the credentials required for the integration, if any. For a Lambda integration, three options are available. To specify an IAM Role for API Gateway to

assume, use the role's Amazon Resource Name (ARN). To require that the caller's identity be passed through from the request, specify `arn:aws:iam::*:user/*`. To use resource-based permissions on supported Amazon services, don't specify this parameter. Currently, this property is not used for HTTP integrations. If provided, this value replaces the credentials associated with the quick create integration. Supported only for HTTP APIs.

Type: string

Required: False

routeKey

This property is part of quick create. If not specified, the route created using quick create is kept. Otherwise, this value replaces the route key of the quick create route. Additional routes may still be added after the API is updated. Supported only for HTTP APIs.

Type: string

Required: False

corsConfiguration

A CORS configuration. Supported only for HTTP APIs.

Type: [Cors](#)

Required: False

disableExecuteApiEndpoint

Specifies whether clients can invoke your API by using the default `execute-api` endpoint. By default, clients can invoke your API with the default `https://{api_id}.execute-api.{region}.amazonaws.com` endpoint. To require that clients use a custom domain name to invoke your API, disable the default `execute-api` endpoint.

Type: boolean

Required: False

ApiMapping

Represent an API mapping. An API mapping relates a path of your custom domain name to a stage of your API. A custom domain name can have multiple API mappings, but the paths can't overlap. To learn more, see [Set Up a Custom Domain Name for an API in API Gateway](#).

URI

`/v2/domainnames/domainName/apimappings/apiMappingId`

HTTP methods

GET

Operation ID: GetApiMapping

Gets an API mapping.

Path parameters

Name	Type	Required	Description
<i>apiMappingId</i>	String	True	The API mapping identifier.
<i>domainName</i>	String	True	The domain name.

Responses

Status code	Response model	Description
200	ApiMapping	Success
400	BadRequestException	One of the parameters in the request is invalid.
404	NotFoundException	The resource specified in the request was not found.

Status code	Response model	Description
429	LimitExceededException	The client is sending more than the allowed number of requests per unit of time.

DELETE

Operation ID: DeleteApiMapping

Deletes an API mapping.

Path parameters

Name	Type	Required	Description
<i>apiMappingId</i>	String	True	The API mapping identifier.
<i>domainName</i>	String	True	The domain name.

Responses

Status code	Response model	Description
204	None	The request has succeeded , and there is no additional content to send in the response payload body.
400	BadRequestException	One of the parameters in the request is invalid.
404	NotFoundException	The resource specified in the request was not found.
429	LimitExceededException	The client is sending more than the allowed number of requests per unit of time.

PATCH

Operation ID: UpdateApiMapping

The API mapping.

Path parameters

Name	Type	Required	Description
<i>apiMappingId</i>	String	True	The API mapping identifier.
<i>domainName</i>	String	True	The domain name.

Query parameters

Name	Type	Required	Description
apiId	String	True	The identifier of the API.

Responses

Status code	Response model	Description
200	ApiMapping	Success
400	BadRequestException	One of the parameters in the request is invalid.
404	NotFoundException	The resource specified in the request was not found.
409	ConflictException	The resource already exists.
429	LimitExceededException	The client is sending more than the allowed number of requests per unit of time.

Schemas

Request bodies

PATCH schema

```
{
  "apiMappingKey": "string",
  "apiId": "string",
  "stage": "string"
}
```

Response bodies

ApiMapping schema

```
{
  "apiMappingId": "string",
  "apiMappingKey": "string",
  "apiId": "string",
  "stage": "string"
}
```

BadRequestException schema

```
{
  "message": "string"
}
```

NotFoundException schema

```
{
  "message": "string",
  "resourceType": "string"
}
```

ConflictException schema

```
{
```

```
"message": "string"
}
```

LimitExceededException schema

```
{
  "message": "string",
  "limitType": "string"
}
```

Properties

ApiMapping

Represents an API mapping.

apiMappingId

The API mapping identifier.

Type: string

Required: False

apiMappingKey

The API mapping key.

Type: string

Required: False

apiId

The API identifier.

Type: string

Required: True

stage

The API stage.

Type: string

Required: True

BadRequestException

The request is not valid, for example, the input is incomplete or incorrect. See the accompanying error message for details.

message

Describes the error encountered.

Type: string

Required: False

ConflictException

The requested operation would cause a conflict with the current state of a service resource associated with the request. Resolve the conflict before retrying this request. See the accompanying error message for details.

message

Describes the error encountered.

Type: string

Required: False

LimitExceededException

A limit has been exceeded. See the accompanying error message for details.

message

Describes the error encountered.

Type: string

Required: False

limitType

The limit type.

Type: string

Required: False

NotFoundException

The resource specified in the request was not found. See the message field for more information.

message

Describes the error encountered.

Type: string

Required: False

resourceType

The resource type.

Type: string

Required: False

UpdateApiMappingInput

Represents the input parameters for an UpdateApiMapping request.

apiMappingKey

The API mapping key.

Type: string

Required: False

apiId

The API identifier.

Type: string

Required: False

stage

The API stage.

Type: string

Required: False

ApiMappings

Represents a collection of API mappings.

URI

/v2/domainnames/*domainName*/apimappings

HTTP methods

GET

Operation ID: GetApiMappings

Gets API mappings.

Path parameters

Name	Type	Required	Description
<i>domainName</i>	String	True	The domain name.

Query parameters

Name	Type	Required	Description
nextToken	String	False	The next page of elements from this

Name	Type	Required	Description
maxResults	String	False	collection. Not valid for the last element of the collection. The maximum number of elements to be returned for this resource.

Responses

Status code	Response model	Description
200	ApiMappings	Success
400	BadRequestException	One of the parameters in the request is invalid.
404	NotFoundException	The resource specified in the request was not found.
429	LimitExceededException	The client is sending more than the allowed number of requests per unit of time.

POST

Operation ID: CreateApiMapping

Creates an API mapping.

Path parameters

Name	Type	Required	Description
<i>domainName</i>	String	True	The domain name.

Responses

Status code	Response model	Description
201	ApiMapping	The request has succeeded and has resulted in the creation of a resource.
400	BadRequestException	One of the parameters in the request is invalid.
404	NotFoundException	The resource specified in the request was not found.
409	ConflictException	The resource already exists.
429	LimitExceededException	The client is sending more than the allowed number of requests per unit of time.

Schemas

Request bodies

POST schema

```
{
  "apiMappingKey": "string",
  "apiId": "string",
  "stage": "string"
}
```

Response bodies

ApiMappings schema

```
{
  "items": [
    {
```

```
    "apiMappingId": "string",
    "apiMappingKey": "string",
    "apiId": "string",
    "stage": "string"
  }
],
"nextToken": "string"
}
```

ApiMapping schema

```
{
  "apiMappingId": "string",
  "apiMappingKey": "string",
  "apiId": "string",
  "stage": "string"
}
```

BadRequestException schema

```
{
  "message": "string"
}
```

NotFoundException schema

```
{
  "message": "string",
  "resourceType": "string"
}
```

ConflictException schema

```
{
  "message": "string"
}
```


LimitExceededException schema

```
{
  "message": "string",
  "limitType": "string"
}
```

Properties

ApiMapping

Represents an API mapping.

apiMappingId

The API mapping identifier.

Type: string

Required: False

apiMappingKey

The API mapping key.

Type: string

Required: False

apiId

The API identifier.

Type: string

Required: True

stage

The API stage.

Type: string

Required: True

ApiMappings

Represents a collection of ApiMappings resources.

items

The elements from this collection.

Type: Array of type [ApiMapping](#)

Required: False

nextToken

The next page of elements from this collection. Not valid for the last element of the collection.

Type: string

Required: False

BadRequestException

The request is not valid, for example, the input is incomplete or incorrect. See the accompanying error message for details.

message

Describes the error encountered.

Type: string

Required: False

ConflictException

The requested operation would cause a conflict with the current state of a service resource associated with the request. Resolve the conflict before retrying this request. See the accompanying error message for details.

message

Describes the error encountered.

Type: string

Required: False

CreateApiMappingInput

Represents the input parameters for a `CreateApiMapping` request.

apiMappingKey

The API mapping key.

Type: string

Required: False

apiId

The API identifier.

Type: string

Required: True

stage

The API stage.

Type: string

Required: True

LimitExceededException

A limit has been exceeded. See the accompanying error message for details.

message

Describes the error encountered.

Type: string

Required: False

limitType

The limit type.

Type: string

Required: False

NotFoundException

The resource specified in the request was not found. See the message field for more information.

message

Describes the error encountered.

Type: string

Required: False

resourceType

The resource type.

Type: string

Required: False

Apis

Represents your APIs as a collection. A collection offers a paginated view of your APIs.

URI

/v2/apis

HTTP methods

GET

Operation ID: GetApis

Gets a collection of `Api` resources.

Query parameters

Name	Type	Required	Description
<code>nextToken</code>	String	False	The next page of elements from this collection. Not valid for the last element of the collection.
<code>maxResults</code>	String	False	The maximum number of elements to be returned for this resource.

Responses

Status code	Response model	Description
200	Apis	Success
400	BadRequestException	One of the parameters in the request is invalid.
404	NotFoundException	The resource specified in the request was not found.
429	LimitExceededException	The client is sending more than the allowed number of requests per unit of time.

POST

Operation ID: `CreateApi`

Creates an `Api` resource.

Responses

Status code	Response model	Description
201	Api	The request has succeeded and has resulted in the creation of a resource.
400	BadRequestException	One of the parameters in the request is invalid.
404	NotFoundException	The resource specified in the request was not found.
409	ConflictException	The resource already exists.
429	LimitExceededException	The client is sending more than the allowed number of requests per unit of time.

PUT

Operation ID: ImportApi

Imports an API.

Query parameters

Name	Type	Required	Description
failOnWarnings	String	False	Specifies whether to rollback the API creation when a warning is encountered. By default, API creation continues if a warning is encountered.

Name	Type	Required	Description
basepath	String	False	Specifies how to interpret the base path of the API during import. Valid values are ignore, prepend, and split. The default value is ignore. To learn more, see Set the OpenAPI basePath Property . Supported only for HTTP APIs.

Responses

Status code	Response model	Description
201	Api	The request has succeeded and has resulted in the creation of a resource.
400	BadRequestException	One of the parameters in the request is invalid.
404	NotFoundException	The resource specified in the request was not found.
409	ConflictException	The resource already exists.
429	LimitExceededException	The client is sending more than the allowed number of requests per unit of time.

Schemas

Request bodies

POST schema

```
{
  "name": "string",
  "description": "string",
  "version": "string",
  "protocolType": enum,
  "routeSelectionExpression": "string",
  "apiKeySelectionExpression": "string",
  "disableSchemaValidation": boolean,
  "tags": {
  },
  "target": "string",
  "credentialsArn": "string",
  "routeKey": "string",
  "corsConfiguration": {
    "allowOrigins": [
      "string"
    ],
    "allowCredentials": boolean,
    "exposeHeaders": [
      "string"
    ],
    "maxAge": integer,
    "allowMethods": [
      "string"
    ],
    "allowHeaders": [
      "string"
    ]
  },
  "disableExecuteApiEndpoint": boolean
}
```

PUT schema

```
{
  "body": "string"
}
```



```
}
```

Response bodies

Apis schema

```
{
  "items": [
    {
      "apiId": "string",
      "name": "string",
      "description": "string",
      "version": "string",
      "protocolType": enum,
      "ipAddressType": enum,
      "routeSelectionExpression": "string",
      "apiKeySelectionExpression": "string",
      "disableSchemaValidation": boolean,
      "warnings": [
        "string"
      ],
      "importInfo": [
        "string"
      ],
      "apiEndpoint": "string",
      "apiGatewayManaged": boolean,
      "createdDate": "string",
      "tags": {
      },
      "corsConfiguration": {
        "allowOrigins": [
          "string"
        ],
        "allowCredentials": boolean,
        "exposeHeaders": [
          "string"
        ],
        "maxAge": integer,
        "allowMethods": [
          "string"
        ],
        "allowHeaders": [
          "string"
        ]
      }
    }
  ]
}
```

```

    ]
  },
  "disableExecuteApiEndpoint": boolean
}
],
"nextToken": "string"
}

```

Api schema

```

{
  "apiId": "string",
  "name": "string",
  "description": "string",
  "version": "string",
  "protocolType": enum,
  "ipAddressType": enum,
  "routeSelectionExpression": "string",
  "apiKeySelectionExpression": "string",
  "disableSchemaValidation": boolean,
  "warnings": [
    "string"
  ],
  "importInfo": [
    "string"
  ],
  "apiEndpoint": "string",
  "apiGatewayManaged": boolean,
  "createdDate": "string",
  "tags": {
  },
  "corsConfiguration": {
    "allowOrigins": [
      "string"
    ],
    "allowCredentials": boolean,
    "exposeHeaders": [
      "string"
    ],
    "maxAge": integer,
    "allowMethods": [
      "string"
    ],
  },
}

```

```
    "allowHeaders": [  
      "string"  
    ],  
    "disableExecuteApiEndpoint": boolean  
  }  
}
```

BadRequestException schema

```
{  
  "message": "string"  
}
```

NotFoundException schema

```
{  
  "message": "string",  
  "resourceType": "string"  
}
```

ConflictException schema

```
{  
  "message": "string"  
}
```

LimitExceededException schema

```
{  
  "message": "string",  
  "limitType": "string"  
}
```

Properties

Api

Represents an API.

apId

The API ID.

Type: string

Required: False

name

The name of the API.

Type: string

Required: True

description

The description of the API.

Type: string

Required: False

version

A version identifier for the API.

Type: string

Required: False

protocolType

The API protocol.

Type: [ProtocolType](#)

Required: True

ipAddressType

The IP address types that can invoke the API.

Type: [IpAddressType](#)

Required: False

routeSelectionExpression

The route selection expression for the API. For HTTP APIs, the `routeSelectionExpression` must be `${request.method} ${request.path}`. If not provided, this will be the default for HTTP APIs. This property is required for WebSocket APIs.

Type: string

Required: True

apiKeySelectionExpression

An API key selection expression. Supported only for WebSocket APIs. See [API Key Selection Expressions](#).

Type: string

Required: False

disableSchemaValidation

Avoid validating models when creating a deployment. Supported only for WebSocket APIs.

Type: boolean

Required: False

warnings

The warning messages reported when `failonwarnings` is turned on during API import.

Type: Array of type string

Required: False

importInfo

The validation information during API import. This may include particular properties of your OpenAPI definition which are ignored during import. Supported only for HTTP APIs.

Type: Array of type string

Required: False

apiEndpoint

The URI of the API, of the form {api-id}.execute-api.{region}.amazonaws.com. The stage name is typically appended to this URI to form a complete path to a deployed API stage.

Type: string

Required: False

Format: uri

apiGatewayManaged

Specifies whether an API is managed by API Gateway. You can't update or delete a managed API by using API Gateway. A managed API can be deleted only through the tooling or service that created it.

Type: boolean

Required: False

createdDate

The timestamp when the API was created.

Type: string

Required: False

Format: date-time

tags

A collection of tags associated with the API.

Type: [Tags](#)

Required: False

corsConfiguration

A CORS configuration. Supported only for HTTP APIs.

Type: [Cors](#)

Required: False

disableExecuteApiEndpoint

Specifies whether clients can invoke your API by using the default `execute-api` endpoint. By default, clients can invoke your API with the default `https://{api_id}.execute-api.{region}.amazonaws.com` endpoint. To require that clients use a custom domain name to invoke your API, disable the default endpoint.

Type: boolean

Required: False

Apis

Represents a collection of APIs.

items

The elements from this collection.

Type: Array of type [Api](#)

Required: False

nextToken

The next page of elements from this collection. Not valid for the last element of the collection.

Type: string

Required: False

BadRequestException

The request is not valid, for example, the input is incomplete or incorrect. See the accompanying error message for details.

message

Describes the error encountered.

Type: string

Required: False

ConflictException

The requested operation would cause a conflict with the current state of a service resource associated with the request. Resolve the conflict before retrying this request. See the accompanying error message for details.

message

Describes the error encountered.

Type: string

Required: False

Cors

Represents a CORS configuration. Supported only for HTTP APIs. See [Configuring CORS](#) for more information.

allowOrigins

Represents a collection of allowed origins. Supported only for HTTP APIs.

Type: Array of type string

Required: False

allowCredentials

Specifies whether credentials are included in the CORS request. Supported only for HTTP APIs.

Type: boolean

Required: False

exposeHeaders

Represents a collection of exposed headers. Supported only for HTTP APIs.

Type: Array of type string

Required: False

maxAge

The number of seconds that the browser should cache preflight request results. Supported only for HTTP APIs.

Type: integer

Required: False

allowMethods

Represents a collection of allowed HTTP methods. Supported only for HTTP APIs.

Type: Array of type string

Required: False

allowHeaders

Represents a collection of allowed headers. Supported only for HTTP APIs.

Type: Array of type string

Required: False

CreateApiInput

Represents the input parameters for a `CreateApi` request.

name

The name of the API.

Type: string

Required: True

description

The description of the API.

Type: string

Required: False

version

A version identifier for the API.

Type: string

Required: False

protocolType

The API protocol.

Type: [ProtocolType](#)

Required: True

routeSelectionExpression

The route selection expression for the API. For HTTP APIs, the `routeSelectionExpression` must be `${request.method} ${request.path}`. If not provided, this will be the default for HTTP APIs. This property is required for WebSocket APIs.

Type: string

Required: False

apiKeySelectionExpression

An API key selection expression. Supported only for WebSocket APIs. See [API Key Selection Expressions](#).

Type: string

Required: False

disableSchemaValidation

Avoid validating models when creating a deployment. Supported only for WebSocket APIs.

Type: boolean

Required: False

tags

The collection of tags. Each tag element is associated with a given resource.

Type: [Tags](#)

Required: False

target

This property is part of quick create. Quick create produces an API with an integration, a default catch-all route, and a default stage which is configured to automatically deploy changes. For HTTP integrations, specify a fully qualified URL. For Lambda integrations, specify a function ARN. The type of the integration will be HTTP_PROXY or AWS_PROXY, respectively. Supported only for HTTP APIs.

Type: string

Required: False

credentialsArn

This property is part of quick create. It specifies the credentials required for the integration, if any. For a Lambda integration, three options are available. To specify an IAM Role for API Gateway to assume, use the role's Amazon Resource Name (ARN). To require that the caller's identity be passed through from the request, specify `arn:aws:iam::*:user/*`. To use resource-based permissions on supported Amazon services, specify `null`. Currently, this property is not used for HTTP integrations. Supported only for HTTP APIs.

Type: string

Required: False

routeKey

This property is part of quick create. If you don't specify a routeKey, a default route of `$default` is created. The `$default` route acts as a catch-all for any request made to your API, for a particular stage. The `$default` route key can't be modified. You can add routes after creating the API, and you can update the route keys of additional routes. Supported only for HTTP APIs.

Type: string

Required: False

corsConfiguration

A CORS configuration. Supported only for HTTP APIs. See [Configuring CORS](#) for more information.

Type: [Cors](#)

Required: False

disableExecuteApiEndpoint

Specifies whether clients can invoke your API by using the default `execute-api` endpoint. By default, clients can invoke your API with the default `https://{api_id}.execute-api.{region}.amazonaws.com` endpoint. To require that clients use a custom domain name to invoke your API, disable the default endpoint.

Type: boolean

Required: False

ImportApiInput

Represents the input to `ImportAPI`. Supported only for HTTP APIs.

body

The OpenAPI definition. Supported only for HTTP APIs.

Type: string

Required: True

IpAddressType

The IP address types that can invoke your API or domain name.

`ipv4`

`dualstack`

LimitExceededException

A limit has been exceeded. See the accompanying error message for details.

message

Describes the error encountered.

Type: string

Required: False

limitType

The limit type.

Type: string

Required: False

NotFoundException

The resource specified in the request was not found. See the message field for more information.

message

Describes the error encountered.

Type: string

Required: False

resourceType

The resource type.

Type: string

Required: False

ProtocolType

Represents a protocol type.

WEBSOCKET

HTTP

Tags

Represents a collection of tags associated with the resource.

key-value pairs

Type: string

Authorizer

Represents an authorizer. JWT authorizers are supported only for HTTP APIs.

URI

`/v2/apis/apiId/authorizers/authorizerId`

HTTP methods

GET

Operation ID: GetAuthorizer

Gets an Authorizer.

Path parameters

Name	Type	Required	Description
<i>authorizerId</i>	String	True	The authorizer identifier.
<i>apiId</i>	String	True	The API identifier.

Responses

Status code	Response model	Description
200	Authorizer	Success

Status code	Response model	Description
404	NotFoundException	The resource specified in the request was not found.
429	LimitExceededException	The client is sending more than the allowed number of requests per unit of time.

DELETE

Operation ID: DeleteAuthorizer

Deletes an Authorizer.

Path parameters

Name	Type	Required	Description
<i>authorizerId</i>	String	True	The authorizer identifier.
<i>apiId</i>	String	True	The API identifier.

Responses

Status code	Response model	Description
204	None	The request has succeeded , and there is no additional content to send in the response payload body.
404	NotFoundException	The resource specified in the request was not found.
429	LimitExceededException	The client is sending more than the allowed number of requests per unit of time.

PATCH

Operation ID: UpdateAuthorizer

Updates an Authorizer.

Path parameters

Name	Type	Required	Description
<i>authorizerId</i>	String	True	The authorizer identifier.
<i>apiId</i>	String	True	The API identifier.

Responses

Status code	Response model	Description
200	Authorizer	Success
400	BadRequestException	One of the parameters in the request is invalid.
404	NotFoundException	The resource specified in the request was not found.
409	ConflictException	The resource already exists.
429	LimitExceededException	The client is sending more than the allowed number of requests per unit of time.

Schemas

Request bodies

PATCH schema

```
{
```



```

"authorizerId": "string",
"authorizerType": enum,
"authorizerUri": "string",
"authorizerCredentialsArn": "string",
"identitySource": [
  "string"
],
"identityValidationExpression": "string",
"authorizerResultTtlInSeconds": integer,
"jwtConfiguration": {
  "issuer": "string",
  "audience": [
    "string"
  ]
},
"authorizerPayloadFormatVersion": "string",
"enableSimpleResponses": boolean
}

```

Response bodies

Authorizer schema

```

{
  "authorizerId": "string",
  "name": "string",
  "authorizerType": enum,
  "authorizerUri": "string",
  "authorizerCredentialsArn": "string",
  "identitySource": [
    "string"
  ],
  "identityValidationExpression": "string",
  "authorizerResultTtlInSeconds": integer,
  "jwtConfiguration": {
    "issuer": "string",
    "audience": [
      "string"
    ]
  },
  "authorizerPayloadFormatVersion": "string",
  "enableSimpleResponses": boolean
}

```

BadRequestException schema

```
{
  "message": "string"
}
```

NotFoundException schema

```
{
  "message": "string",
  "resourceType": "string"
}
```

ConflictException schema

```
{
  "message": "string"
}
```

LimitExceededException schema

```
{
  "message": "string",
  "limitType": "string"
}
```

Properties

Authorizer

Represents an authorizer.

authorizerId

The authorizer identifier.

Type: string

Required: False

name

The name of the authorizer.

Type: string

Required: True

authorizerType

The authorizer type. Specify REQUEST for a Lambda function using incoming request parameters. Specify JWT to use JSON Web Tokens (supported only for HTTP APIs).

Type: [AuthorizerType](#)

Required: False

authorizerUri

The authorizer's Uniform Resource Identifier (URI). For REQUEST authorizers, this must be a well-formed Lambda function URI, for example, `arn:aws:apigateway:us-west-2:lambda:path/2015-03-31/functions/arn:aws:lambda:us-west-2:{account_id}:function:{lambda_function_name}/invocations`. In general, the URI has this form: `arn:aws:apigateway:{region}:lambda:path/{service_api}` , where `{region}` is the same as the region hosting the Lambda function, `path` indicates that the remaining substring in the URI should be treated as the path to the resource, including the initial `/`. For Lambda functions, this is usually of the form `/2015-03-31/functions/[FunctionARN]/invocations`. Supported only for REQUEST authorizers.

Type: string

Required: False

authorizerCredentialsArn

Specifies the required credentials as an IAM role for API Gateway to invoke the authorizer. To specify an IAM role for API Gateway to assume, use the role's Amazon Resource Name (ARN). To use resource-based permissions on the Lambda function, don't specify this parameter. Supported only for REQUEST authorizers.

Type: string

Required: False

identitySource

The identity source for which authorization is requested.

For a REQUEST authorizer, this is optional. The value is a set of one or more mapping expressions of the specified request parameters. The identity source can be headers, query string parameters, stage variables, and context parameters. For example, if an Auth header and a Name query string parameter are defined as identity sources, this value is `route.request.header.Auth`, `route.request.querystring.Name` for WebSocket APIs. For HTTP APIs, use selection expressions prefixed with `$`, for example, `$request.header.Auth`, `$request.querystring.Name`. These parameters are used to perform runtime validation for Lambda-based authorizers by verifying all of the identity-related request parameters are present in the request, not null, and non-empty. Only when this is true does the authorizer invoke the authorizer Lambda function. Otherwise, it returns a 401 Unauthorized response without calling the Lambda function. For HTTP APIs, identity sources are also used as the cache key when caching is enabled. To learn more, see [Working with Amazon Lambda authorizers for HTTP APIs](#).

For JWT, a single entry that specifies where to extract the JSON Web Token (JWT) from inbound requests. Currently only header-based and query parameter-based selections are supported, for example `$request.header.Authorization`.

Type: Array of type string

Required: False

identityValidationExpression

The validation expression does not apply to the REQUEST authorizer.

Type: string

Required: False

authorizerResultTtlInSeconds

The time to live (TTL) for cached authorizer results, in seconds. If it equals 0, authorization caching is disabled. If it is greater than 0, API Gateway caches authorizer responses. The maximum value is 3600, or 1 hour. Supported only for HTTP API Lambda authorizers.

Type: integer

Required: False

jwtConfiguration

Represents the configuration of a JWT authorizer. Required for the JWT authorizer type. Supported only for HTTP APIs.

Type: [JWTConfiguration](#)

Required: False

authorizerPayloadFormatVersion

Specifies the format of the payload sent to an HTTP API Lambda authorizer. Required for HTTP API Lambda authorizers. Supported values are 1.0 and 2.0. To learn more, see [Working with Amazon Lambda authorizers for HTTP APIs](#).

Type: string

Required: False

enableSimpleResponses

Specifies whether a Lambda authorizer returns a response in a simple format. If enabled, the Lambda authorizer can return a boolean value instead of an IAM policy. Supported only for HTTP APIs. To learn more, see [Working with Amazon Lambda authorizers for HTTP APIs](#).

Type: boolean

Required: False

AuthorizerType

The authorizer type. Specify REQUEST for a Lambda function using incoming request parameters. Specify JWT to use JSON Web Tokens (supported only for HTTP APIs).

REQUEST

JWT

BadRequestException

The request is not valid, for example, the input is incomplete or incorrect. See the accompanying error message for details.

message

Describes the error encountered.

Type: string

Required: False

ConflictException

The requested operation would cause a conflict with the current state of a service resource associated with the request. Resolve the conflict before retrying this request. See the accompanying error message for details.

message

Describes the error encountered.

Type: string

Required: False

JWTConfiguration

Represents the configuration of a JWT authorizer. Required for the JWT authorizer type. Supported only for HTTP APIs.

issuer

The base domain of the identity provider that issues JSON Web Tokens. For example, an Amazon Cognito user pool has the following format: `https://cognito-idp.{region}.amazonaws.com/{userPoolId}` . Required for the JWT authorizer type. Supported only for HTTP APIs.

Type: string

Required: False

audience

A list of the intended recipients of the JWT. A valid JWT must provide an aud that matches at least one entry in this list. See [RFC 7519](#). Required for the JWT authorizer type. Supported only for HTTP APIs.

Type: Array of type string

Required: False

LimitExceededException

A limit has been exceeded. See the accompanying error message for details.

message

Describes the error encountered.

Type: string

Required: False

limitType

The limit type.

Type: string

Required: False

NotFoundException

The resource specified in the request was not found. See the message field for more information.

message

Describes the error encountered.

Type: string

Required: False

resourceType

The resource type.

Type: string

Required: False

UpdateAuthorizerInput

The input parameters for an `UpdateAuthorizer` request.

name

The name of the authorizer.

Type: string

Required: False

authorizerType

The authorizer type. Specify `REQUEST` for a Lambda function using incoming request parameters. Specify `JWT` to use JSON Web Tokens (supported only for HTTP APIs).

Type: [AuthorizerType](#)

Required: False

authorizerUri

The authorizer's Uniform Resource Identifier (URI). For `REQUEST` authorizers, this must be a well-formed Lambda function URI, for example, `arn:aws:apigateway:us-west-2:lambda:path/2015-03-31/functions/arn:aws:lambda:us-west-2:{account_id}:function:{lambda_function_name}/invocations`. In general, the URI has this form: `arn:aws:apigateway:{region}:lambda:path/{service_api}` , where `{region}` is the same as the region hosting the Lambda function, `path` indicates that the remaining substring in the URI should be treated as the path to the resource, including the initial `/`. For Lambda functions, this is usually of the form `/2015-03-31/functions/[FunctionARN]/invocations`. Supported only for `REQUEST` authorizers.

Type: string

Required: False

authorizerCredentialsArn

Specifies the required credentials as an IAM role for API Gateway to invoke the authorizer. To specify an IAM role for API Gateway to assume, use the role's Amazon Resource Name (ARN). To use resource-based permissions on the Lambda function, don't specify this parameter.

Type: string

Required: False

identitySource

The identity source for which authorization is requested.

For a REQUEST authorizer, this is optional. The value is a set of one or more mapping expressions of the specified request parameters. The identity source can be headers, query string parameters, stage variables, and context parameters. For example, if an Auth header and a Name query string parameter are defined as identity sources, this value is `route.request.header.Auth`, `route.request.querystring.Name` for WebSocket APIs. For HTTP APIs, use selection expressions prefixed with `$`, for example, `$request.header.Auth`, `$request.querystring.Name`. These parameters are used to perform runtime validation for Lambda-based authorizers by verifying all of the identity-related request parameters are present in the request, not null, and non-empty. Only when this is true does the authorizer invoke the authorizer Lambda function. Otherwise, it returns a 401 Unauthorized response without calling the Lambda function. For HTTP APIs, identity sources are also used as the cache key when caching is enabled. To learn more, see [Working with Amazon Lambda authorizers for HTTP APIs](#).

For JWT, a single entry that specifies where to extract the JSON Web Token (JWT) from inbound requests. Currently only header-based and query parameter-based selections are supported, for example `$request.header.Authorization`.

Type: Array of type string

Required: False

identityValidationExpression

This parameter is not used.

Type: string

Required: False

authorizerResultTtlInSeconds

The time to live (TTL) for cached authorizer results, in seconds. If it equals 0, authorization caching is disabled. If it is greater than 0, API Gateway caches authorizer responses. The maximum value is 3600, or 1 hour. Supported only for HTTP API Lambda authorizers.

Type: integer

Required: False

jwtConfiguration

Represents the configuration of a JWT authorizer. Required for the JWT authorizer type. Supported only for HTTP APIs.

Type: [JWTConfiguration](#)

Required: False

authorizerPayloadFormatVersion

Specifies the format of the payload sent to an HTTP API Lambda authorizer. Required for HTTP API Lambda authorizers. Supported values are 1.0 and 2.0. To learn more, see [Working with Amazon Lambda authorizers for HTTP APIs](#).

Type: string

Required: False

enableSimpleResponses

Specifies whether a Lambda authorizer returns a response in a simple format. By default, a Lambda authorizer must return an IAM policy. If enabled, the Lambda authorizer can return a boolean value instead of an IAM policy. Supported only for HTTP APIs. To learn more, see [Working with Amazon Lambda authorizers for HTTP APIs](#).

Type: boolean

Required: False

Authorizers

Represents the collection of authorizers for an API.

URI

/v2/apis/*apiId*/authorizers

HTTP methods

GET

Operation ID: GetAuthorizers

Gets the Authorizers for an API.

Path parameters

Name	Type	Required	Description
<i>apiId</i>	String	True	The API identifier.

Query parameters

Name	Type	Required	Description
nextToken	String	False	The next page of elements from this collection. Not valid for the last element of the collection.
maxResults	String	False	The maximum number of elements to be returned for this resource.

Responses

Status code	Response model	Description
200	Authorizers	Success
400	BadRequestException	One of the parameters in the request is invalid.
404	NotFoundException	The resource specified in the request was not found.
429	LimitExceededException	The client is sending more than the allowed number of requests per unit of time.

POST

Operation ID: CreateAuthorizer

Creates an Authorizer for an API.

Path parameters

Name	Type	Required	Description
<i>apiId</i>	String	True	The API identifier.

Responses

Status code	Response model	Description
201	Authorizer	The request has succeeded and has resulted in the creation of a resource.
400	BadRequestException	One of the parameters in the request is invalid.

Status code	Response model	Description
404	NotFoundException	The resource specified in the request was not found.
409	ConflictException	The resource already exists.
429	LimitExceededException	The client is sending more than the allowed number of requests per unit of time.

Schemas

Request bodies

POST schema

```
{
  "name": "string",
  "authorizerType": enum,
  "authorizerUri": "string",
  "authorizerCredentialsArn": "string",
  "identitySource": [
    "string"
  ],
  "identityValidationExpression": "string",
  "authorizerResultTtlInSeconds": integer,
  "jwtConfiguration": {
    "issuer": "string",
    "audience": [
      "string"
    ]
  },
  "authorizerPayloadFormatVersion": "string",
  "enableSimpleResponses": boolean
}
```

Response bodies

Authorizers schema

```
{
  "items": [
    {
      "authorizerId": "string",
      "name": "string",
      "authorizerType": enum,
      "authorizerUri": "string",
      "authorizerCredentialsArn": "string",
      "identitySource": [
        "string"
      ],
      "identityValidationExpression": "string",
      "authorizerResultTtlInSeconds": integer,
      "jwtConfiguration": {
        "issuer": "string",
        "audience": [
          "string"
        ]
      },
      "authorizerPayloadFormatVersion": "string",
      "enableSimpleResponses": boolean
    }
  ],
  "nextToken": "string"
}
```

Authorizer schema

```
{
  "authorizerId": "string",
  "name": "string",
  "authorizerType": enum,
  "authorizerUri": "string",
  "authorizerCredentialsArn": "string",
  "identitySource": [
    "string"
  ],
  "identityValidationExpression": "string",
  "authorizerResultTtlInSeconds": integer,
  "jwtConfiguration": {
    "issuer": "string",
```

```
  "audience": [  
    "string"  
  ],  
  "authorizerPayloadFormatVersion": "string",  
  "enableSimpleResponses": boolean  
}
```

BadRequestException schema

```
{  
  "message": "string"  
}
```

NotFoundException schema

```
{  
  "message": "string",  
  "resourceType": "string"  
}
```

ConflictException schema

```
{  
  "message": "string"  
}
```

LimitExceededException schema

```
{  
  "message": "string",  
  "limitType": "string"  
}
```

Properties

Authorizer

Represents an authorizer.

authorizerId

The authorizer identifier.

Type: string

Required: False

name

The name of the authorizer.

Type: string

Required: True

authorizerType

The authorizer type. Specify REQUEST for a Lambda function using incoming request parameters. Specify JWT to use JSON Web Tokens (supported only for HTTP APIs).

Type: [AuthorizerType](#)

Required: False

authorizerUri

The authorizer's Uniform Resource Identifier (URI). For REQUEST authorizers, this must be a well-formed Lambda function URI, for example, `arn:aws:apigateway:us-west-2:lambda:path/2015-03-31/functions/arn:aws:lambda:us-west-2:{account_id}:function:{lambda_function_name}/invocations`. In general, the URI has this form: `arn:aws:apigateway:{region}:lambda:path/{service_api}`, where `{region}` is the same as the region hosting the Lambda function, `path` indicates that the remaining substring in the URI should be treated as the path to the resource, including the initial `/`. For Lambda functions, this is usually of the form `/2015-03-31/functions/[FunctionARN]/invocations`. Supported only for REQUEST authorizers.

Type: string

Required: False

authorizerCredentialsArn

Specifies the required credentials as an IAM role for API Gateway to invoke the authorizer. To specify an IAM role for API Gateway to assume, use the role's Amazon Resource Name (ARN). To use resource-based permissions on the Lambda function, don't specify this parameter. Supported only for REQUEST authorizers.

Type: string

Required: False

identitySource

The identity source for which authorization is requested.

For a REQUEST authorizer, this is optional. The value is a set of one or more mapping expressions of the specified request parameters. The identity source can be headers, query string parameters, stage variables, and context parameters. For example, if an Auth header and a Name query string parameter are defined as identity sources, this value is `route.request.header.Auth`, `route.request.querystring.Name` for WebSocket APIs. For HTTP APIs, use selection expressions prefixed with `$`, for example, `$request.header.Auth`, `$request.querystring.Name`. These parameters are used to perform runtime validation for Lambda-based authorizers by verifying all of the identity-related request parameters are present in the request, not null, and non-empty. Only when this is true does the authorizer invoke the authorizer Lambda function. Otherwise, it returns a 401 Unauthorized response without calling the Lambda function. For HTTP APIs, identity sources are also used as the cache key when caching is enabled. To learn more, see [Working with Amazon Lambda authorizers for HTTP APIs](#).

For JWT, a single entry that specifies where to extract the JSON Web Token (JWT) from inbound requests. Currently only header-based and query parameter-based selections are supported, for example `$request.header.Authorization`.

Type: Array of type string

Required: False

identityValidationExpression

The validation expression does not apply to the REQUEST authorizer.

Type: string

Required: False

authorizerResultTtlInSeconds

The time to live (TTL) for cached authorizer results, in seconds. If it equals 0, authorization caching is disabled. If it is greater than 0, API Gateway caches authorizer responses. The maximum value is 3600, or 1 hour. Supported only for HTTP API Lambda authorizers.

Type: integer

Required: False

jwtConfiguration

Represents the configuration of a JWT authorizer. Required for the JWT authorizer type. Supported only for HTTP APIs.

Type: [JWTConfiguration](#)

Required: False

authorizerPayloadFormatVersion

Specifies the format of the payload sent to an HTTP API Lambda authorizer. Required for HTTP API Lambda authorizers. Supported values are 1.0 and 2.0. To learn more, see [Working with Amazon Lambda authorizers for HTTP APIs](#).

Type: string

Required: False

enableSimpleResponses

Specifies whether a Lambda authorizer returns a response in a simple format. If enabled, the Lambda authorizer can return a boolean value instead of an IAM policy. Supported only for HTTP APIs. To learn more, see [Working with Amazon Lambda authorizers for HTTP APIs](#).

Type: boolean

Required: False

AuthorizerType

The authorizer type. Specify REQUEST for a Lambda function using incoming request parameters. Specify JWT to use JSON Web Tokens (supported only for HTTP APIs).

REQUEST

JWT

Authorizers

Represents a collection of authorizers.

items

The elements from this collection.

Type: Array of type [Authorizer](#)

Required: False

nextToken

The next page of elements from this collection. Not valid for the last element of the collection.

Type: string

Required: False

BadRequestException

The request is not valid, for example, the input is incomplete or incorrect. See the accompanying error message for details.

message

Describes the error encountered.

Type: string

Required: False

ConflictException

The requested operation would cause a conflict with the current state of a service resource associated with the request. Resolve the conflict before retrying this request. See the accompanying error message for details.

message

Describes the error encountered.

Type: string

Required: False

CreateAuthorizerInput

Represents the input parameters for a `CreateAuthorizer` request.

name

The name of the authorizer.

Type: string

Required: True

authorizerType

The authorizer type. Specify `REQUEST` for a Lambda function using incoming request parameters. Specify `JWT` to use JSON Web Tokens (supported only for HTTP APIs).

Type: [AuthorizerType](#)

Required: True

authorizerUri

The authorizer's Uniform Resource Identifier (URI). For `REQUEST` authorizers, this must be a well-formed Lambda function URI, for example, `arn:aws:apigateway:us-west-2:lambda:path/2015-03-31/functions/arn:aws:lambda:us-west-2:{account_id}:function:{lambda_function_name}/invocations`. In general, the URI has this form: `arn:aws:apigateway:{region}:lambda:path/{service_api}` ,

where {region} is the same as the region hosting the Lambda function, path indicates that the remaining substring in the URI should be treated as the path to the resource, including the initial /. For Lambda functions, this is usually of the form /2015-03-31/functions/[FunctionARN]/invocations. Supported only for REQUEST authorizers.

Type: string

Required: False

authorizerCredentialsArn

Specifies the required credentials as an IAM role for API Gateway to invoke the authorizer. To specify an IAM role for API Gateway to assume, use the role's Amazon Resource Name (ARN). To use resource-based permissions on the Lambda function, don't specify this parameter. Supported only for REQUEST authorizers.

Type: string

Required: False

identitySource

The identity source for which authorization is requested.

For a REQUEST authorizer, this is optional. The value is a set of one or more mapping expressions of the specified request parameters. The identity source can be headers, query string parameters, stage variables, and context parameters. For example, if an Auth header and a Name query string parameter are defined as identity sources, this value is `route.request.header.Auth`, `route.request.querystring.Name` for WebSocket APIs. For HTTP APIs, use selection expressions prefixed with `$`, for example, `$request.header.Auth`, `$request.querystring.Name`. These parameters are used to perform runtime validation for Lambda-based authorizers by verifying all of the identity-related request parameters are present in the request, not null, and non-empty. Only when this is true does the authorizer invoke the authorizer Lambda function. Otherwise, it returns a 401 Unauthorized response without calling the Lambda function. For HTTP APIs, identity sources are also used as the cache key when caching is enabled. To learn more, see [Working with Amazon Lambda authorizers for HTTP APIs](#).

For JWT, a single entry that specifies where to extract the JSON Web Token (JWT) from inbound requests. Currently only header-based and query parameter-based selections are supported, for example `$request.header.Authorization`.

Type: Array of type string

Required: True

identityValidationExpression

This parameter is not used.

Type: string

Required: False

authorizerResultTtlInSeconds

The time to live (TTL) for cached authorizer results, in seconds. If it equals 0, authorization caching is disabled. If it is greater than 0, API Gateway caches authorizer responses. The maximum value is 3600, or 1 hour. Supported only for HTTP API Lambda authorizers.

Type: integer

Required: False

jwtConfiguration

Represents the configuration of a JWT authorizer. Required for the JWT authorizer type. Supported only for HTTP APIs.

Type: [JWTConfiguration](#)

Required: False

authorizerPayloadFormatVersion

Specifies the format of the payload sent to an HTTP API Lambda authorizer. Required for HTTP API Lambda authorizers. Supported values are 1.0 and 2.0. To learn more, see [Working with Amazon Lambda authorizers for HTTP APIs](#).

Type: string

Required: False

enableSimpleResponses

Specifies whether a Lambda authorizer returns a response in a simple format. By default, a Lambda authorizer must return an IAM policy. If enabled, the Lambda authorizer can return a boolean value instead of an IAM policy. Supported only for HTTP APIs. To learn more, see [Working with Amazon Lambda authorizers for HTTP APIs](#).

Type: boolean

Required: False

JWTConfiguration

Represents the configuration of a JWT authorizer. Required for the JWT authorizer type. Supported only for HTTP APIs.

issuer

The base domain of the identity provider that issues JSON Web Tokens. For example, an Amazon Cognito user pool has the following format: `https://cognito-idp.{region}.amazonaws.com/{userPoolId}`. Required for the JWT authorizer type. Supported only for HTTP APIs.

Type: string

Required: False

audience

A list of the intended recipients of the JWT. A valid JWT must provide an aud that matches at least one entry in this list. See [RFC 7519](#). Required for the JWT authorizer type. Supported only for HTTP APIs.

Type: Array of type string

Required: False

LimitExceededException

A limit has been exceeded. See the accompanying error message for details.

message

Describes the error encountered.

Type: string

Required: False

limitType

The limit type.

Type: string

Required: False

NotFoundException

The resource specified in the request was not found. See the message field for more information.

message

Describes the error encountered.

Type: string

Required: False

resourceType

The resource type.

Type: string

Required: False

AuthorizersCache

Represents the authorizers cache for a stage. Supported only for HTTP API Lambda authorizers.

URI

`/v2/apis/apiId/stages/stageName/cache/authorizers`

HTTP methods

DELETE

Operation ID: ResetAuthorizersCache

Resets all authorizer cache entries for the specified stage. Supported only for HTTP API Lambda authorizers.

Path parameters

Name	Type	Required	Description
<i>stageName</i>	String	True	The stage name. Stage names can contain only alphanumeric characters, hyphens, and underscores, or be <code>\$default</code> . Maximum length is 128 characters.
<i>apiId</i>	String	True	The API identifier.

Responses

Status code	Response model	Description
204	None	The request has succeeded , and there is no additional content to send in the response payload body.
404	NotFoundException	The resource specified in the request was not found.

Status code	Response model	Description
429	LimitExceededException	The client is sending more than the allowed number of requests per unit of time.

Schemas

Response bodies

NotFoundExpection schema

```
{
  "message": "string",
  "resourceType": "string"
}
```

LimitExceededException schema

```
{
  "message": "string",
  "limitType": "string"
}
```

Properties

LimitExceededException

A limit has been exceeded. See the accompanying error message for details.

message

Describes the error encountered.

Type: string

Required: False

limitType

The limit type.

Type: string

Required: False

NotFoundException

The resource specified in the request was not found. See the message field for more information.

message

Describes the error encountered.

Type: string

Required: False

resourceType

The resource type.

Type: string

Required: False

Cors

Represents a CORS configuration. Supported only for HTTP APIs.

URI

`/v2/apis/apiId/cors`

HTTP methods

DELETE

Operation ID: DeleteCorsConfiguration

Deletes a CORS configuration.

Path parameters

Name	Type	Required	Description
<i>apiId</i>	String	True	The API identifier.

Responses

Status code	Response model	Description
204	None	The request has succeeded , and there is no additional content to send in the response payload body.
404	NotFoundException	The resource specified in the request was not found.
429	LimitExceededException	The client is sending more than the allowed number of requests per unit of time.

Schemas

Response bodies

NotFoundException schema

```
{
  "message": "string",
  "resourceType": "string"
}
```

LimitExceededException schema

```
{
  "message": "string",
  "limitType": "string"
}
```

```
}
```

Properties

LimitExceededException

A limit has been exceeded. See the accompanying error message for details.

message

Describes the error encountered.

Type: string

Required: False

limitType

The limit type.

Type: string

Required: False

NotFoundException

The resource specified in the request was not found. See the message field for more information.

message

Describes the error encountered.

Type: string

Required: False

resourceType

The resource type.

Type: string

Required: False

Deployment

Represents an API deployment.

URI

`/v2/apis/apiId/deployments/deploymentId`

HTTP methods

GET

Operation ID: GetDeployment

Gets a Deployment.

Path parameters

Name	Type	Required	Description
<i>apiId</i>	String	True	The API identifier.
<i>deploymentId</i>	String	True	The deployment ID.

Responses

Status code	Response model	Description
200	Deployment	Success
404	NotFoundException	The resource specified in the request was not found.
429	LimitExceededException	The client is sending more than the allowed number of requests per unit of time.

DELETE

Operation ID: DeleteDeployment

Deletes a Deployment.

Path parameters

Name	Type	Required	Description
<i>apiId</i>	String	True	The API identifier.
<i>deploymentId</i>	String	True	The deployment ID.

Responses

Status code	Response model	Description
204	None	The request has succeeded , and there is no additional content to send in the response payload body.
404	NotFoundException	The resource specified in the request was not found.
429	LimitExceededException	The client is sending more than the allowed number of requests per unit of time.

PATCH

Operation ID: UpdateDeployment

Updates a Deployment.

Path parameters

Name	Type	Required	Description
<i>apiId</i>	String	True	The API identifier.
<i>deploymentId</i>	String	True	The deployment ID.

Responses

Status code	Response model	Description
200	Deployment	Success
400	BadRequestException	One of the parameters in the request is invalid.
404	NotFoundException	The resource specified in the request was not found.
409	ConflictException	The resource already exists.
429	LimitExceededException	The client is sending more than the allowed number of requests per unit of time.

Schemas

Request bodies

PATCH schema

```
{
  "description": "string"
}
```

Response bodies

Deployment schema

```
{
  "deploymentId": "string",
  "description": "string",
  "createdDate": "string",
  "deploymentStatus": enum,
  "deploymentStatusMessage": "string",
  "autoDeployed": boolean
}
```


BadRequestException schema

```
{  
  "message": "string"  
}
```

NotFoundException schema

```
{  
  "message": "string",  
  "resourceType": "string"  
}
```

ConflictException schema

```
{  
  "message": "string"  
}
```

LimitExceededException schema

```
{  
  "message": "string",  
  "limitType": "string"  
}
```

Properties

BadRequestException

The request is not valid, for example, the input is incomplete or incorrect. See the accompanying error message for details.

message

Describes the error encountered.

Type: string

Required: False

ConflictException

The requested operation would cause a conflict with the current state of a service resource associated with the request. Resolve the conflict before retrying this request. See the accompanying error message for details.

message

Describes the error encountered.

Type: string

Required: False

Deployment

An immutable representation of an API that can be called by users. A Deployment must be associated with a Stage for it to be callable over the internet.

deploymentId

The identifier for the deployment.

Type: string

Required: False

description

The description for the deployment.

Type: string

Required: False

createdDate

The date and time when the Deployment resource was created.

Type: string

Required: False

Format: date-time

deploymentStatus

The status of the deployment: PENDING, FAILED, or SUCCEEDED.

Type: [DeploymentStatus](#)

Required: False

deploymentStatusMessage

May contain additional feedback on the status of an API deployment.

Type: string

Required: False

autoDeployed

Specifies whether a deployment was automatically released.

Type: boolean

Required: False

DeploymentStatus

Represents a deployment status.

PENDING

FAILED

DEPLOYED

LimitExceededException

A limit has been exceeded. See the accompanying error message for details.

message

Describes the error encountered.

Type: string

Required: False

limitType

The limit type.

Type: string

Required: False

NotFoundException

The resource specified in the request was not found. See the message field for more information.

message

Describes the error encountered.

Type: string

Required: False

resourceType

The resource type.

Type: string

Required: False

UpdateDeploymentInput

Represents the input parameters for an UpdateDeployment request.

description

The description for the deployment resource.

Type: string

Required: False

Deployments

Represents the collection of deployments for an API.

URI

/v2/apis/*apiId*/deployments

HTTP methods

GET

Operation ID: GetDeployments

Gets the Deployments for an API.

Path parameters

Name	Type	Required	Description
<i>apiId</i>	String	True	The API identifier.

Query parameters

Name	Type	Required	Description
nextToken	String	False	The next page of elements from this collection. Not valid for the last element of the collection.
maxResults	String	False	The maximum number of elements to be returned for this resource.

Responses

Status code	Response model	Description
200	Deployments	Success

Status code	Response model	Description
400	BadRequestException	One of the parameters in the request is invalid.
404	NotFoundException	The resource specified in the request was not found.
429	LimitExceededException	The client is sending more than the allowed number of requests per unit of time.

POST

Operation ID: CreateDeployment

Creates a Deployment for an API.

Path parameters

Name	Type	Required	Description
<i>apiId</i>	String	True	The API identifier.

Responses

Status code	Response model	Description
201	Deployment	The request has succeeded and has resulted in the creation of a resource.
400	BadRequestException	One of the parameters in the request is invalid.
404	NotFoundException	The resource specified in the request was not found.
409	ConflictException	The resource already exists.

Status code	Response model	Description
429	LimitExceededException	The client is sending more than the allowed number of requests per unit of time.

Schemas

Request bodies

POST schema

```
{
  "description": "string",
  "stageName": "string"
}
```

Response bodies

Deployments schema

```
{
  "items": [
    {
      "deploymentId": "string",
      "description": "string",
      "createdDate": "string",
      "deploymentStatus": enum,
      "deploymentStatusMessage": "string",
      "autoDeployed": boolean
    }
  ],
  "nextToken": "string"
}
```

Deployment schema

```
{
  "deploymentId": "string",
}
```

```
"description": "string",
"createdDate": "string",
"deploymentStatus": enum,
"deploymentStatusMessage": "string",
"autoDeployed": boolean
}
```

BadRequestException schema

```
{
  "message": "string"
}
```

NotFoundException schema

```
{
  "message": "string",
  "resourceType": "string"
}
```

ConflictException schema

```
{
  "message": "string"
}
```

LimitExceededException schema

```
{
  "message": "string",
  "limitType": "string"
}
```

Properties

BadRequestException

The request is not valid, for example, the input is incomplete or incorrect. See the accompanying error message for details.

message

Describes the error encountered.

Type: string

Required: False

ConflictException

The requested operation would cause a conflict with the current state of a service resource associated with the request. Resolve the conflict before retrying this request. See the accompanying error message for details.

message

Describes the error encountered.

Type: string

Required: False

CreateDeploymentInput

Represents the input parameters for a `CreateDeployment` request.

description

The description for the deployment resource.

Type: string

Required: False

stageName

The name of an existing stage to associate with the deployment.

Type: string

Required: False

Deployment

An immutable representation of an API that can be called by users. A Deployment must be associated with a Stage for it to be callable over the internet.

deploymentId

The identifier for the deployment.

Type: string

Required: False

description

The description for the deployment.

Type: string

Required: False

createdDate

The date and time when the Deployment resource was created.

Type: string

Required: False

Format: date-time

deploymentStatus

The status of the deployment: PENDING, FAILED, or SUCCEEDED.

Type: [DeploymentStatus](#)

Required: False

deploymentStatusMessage

May contain additional feedback on the status of an API deployment.

Type: string

Required: False

autoDeployed

Specifies whether a deployment was automatically released.

Type: boolean

Required: False

DeploymentStatus

Represents a deployment status.

PENDING

FAILED

DEPLOYED

Deployments

A collection resource that contains zero or more references to your existing deployments, and links that guide you on how to interact with your collection. The collection offers a paginated view of the contained deployments.

items

The elements from this collection.

Type: Array of type [Deployment](#)

Required: False

nextToken

The next page of elements from this collection. Not valid for the last element of the collection.

Type: string

Required: False

LimitExceededException

A limit has been exceeded. See the accompanying error message for details.

message

Describes the error encountered.

Type: string

Required: False

limitType

The limit type.

Type: string

Required: False

NotFoundException

The resource specified in the request was not found. See the message field for more information.

message

Describes the error encountered.

Type: string

Required: False

resourceType

The resource type.

Type: string

Required: False

DomainName

Represents a domain name. See [Set up Custom Domain Name for an API in API Gateway](#).

URI

`/v2/domainnames/domainName`

HTTP methods

GET

Operation ID: GetDomainName

Gets a domain name.

Path parameters

Name	Type	Required	Description
<i>domainName</i>	String	True	The domain name.

Responses

Status code	Response model	Description
200	DomainName	Success
404	NotFoundException	The resource specified in the request was not found.
429	LimitExceededException	The client is sending more than the allowed number of requests per unit of time.

DELETE

Operation ID: DeleteDomainName

Deletes a domain name.

Path parameters

Name	Type	Required	Description
<i>domainName</i>	String	True	The domain name.

Responses

Status code	Response model	Description
204	None	The request has succeeded , and there is no additional content to send in the response payload body.
404	NotFoundException	The resource specified in the request was not found.
429	LimitExceededException	The client is sending more than the allowed number of requests per unit of time.

PATCH

Operation ID: UpdateDomainName

Updates a domain name.

Path parameters

Name	Type	Required	Description
<i>domainName</i>	String	True	The domain name.

Responses

Status code	Response model	Description
200	DomainName	Success
400	BadRequestException	One of the parameters in the request is invalid.
404	NotFoundException	The resource specified in the request was not found.

Status code	Response model	Description
409	ConflictException	The resource already exists.
429	LimitExceededException	The client is sending more than the allowed number of requests per unit of time.

Schemas

Request bodies

PATCH schema

```
{
  "domainNameConfigurations": [
    {
      "apiGatewayDomainName": "string",
      "certificateArn": "string",
      "certificateName": "string",
      "certificateUploadDate": "string",
      "domainNameStatus": enum,
      "domainNameStatusMessage": "string",
      "endpointType": enum,
      "hostedZoneId": "string",
      "ipAddressType": enum,
      "ownershipVerificationCertificateArn": "string",
      "securityPolicy": enum
    }
  ],
  "mutualTlsAuthentication": {
    "truststoreUri": "string",
    "truststoreVersion": "string"
  }
}
```

Response bodies

DomainName schema

```
{
```

```
"domainName": "string",
"domainNameConfigurations": [
  {
    "apiGatewayDomainName": "string",
    "certificateArn": "string",
    "certificateName": "string",
    "certificateUploadDate": "string",
    "domainNameStatus": enum,
    "domainNameStatusMessage": "string",
    "endpointType": enum,
    "hostedZoneId": "string",
    "ipAddressType": enum,
    "ownershipVerificationCertificateArn": "string",
    "securityPolicy": enum
  }
],
"apiMappingSelectionExpression": "string",
"tags": {
},
"mutualTlsAuthentication": {
  "truststoreUri": "string",
  "truststoreVersion": "string",
  "truststoreWarnings": [
    "string"
  ]
}
}
```

BadRequestException schema

```
{
  "message": "string"
}
```

NotFoundException schema

```
{
  "message": "string",
  "resourceType": "string"
}
```


ConflictException schema

```
{  
  "message": "string"  
}
```

LimitExceededException schema

```
{  
  "message": "string",  
  "limitType": "string"  
}
```

Properties

BadRequestException

The request is not valid, for example, the input is incomplete or incorrect. See the accompanying error message for details.

message

Describes the error encountered.

Type: string

Required: False

ConflictException

The requested operation would cause a conflict with the current state of a service resource associated with the request. Resolve the conflict before retrying this request. See the accompanying error message for details.

message

Describes the error encountered.

Type: string

Required: False

DomainName

Represents a domain name.

domainName

The name of the DomainName resource.

Type: string

Required: True

domainNameConfigurations

The domain name configurations.

Type: Array of type [DomainNameConfiguration](#)

Required: False

apiMappingSelectionExpression

The API mapping selection expression.

Type: string

Required: False

tags

The collection of tags associated with a domain name.

Type: [Tags](#)

Required: False

mutualTlsAuthentication

The mutual TLS authentication configuration for a custom domain name.

Type: [MutualTlsAuthentication](#)

Required: False

DomainNameConfiguration

The domain name configuration.

apiGatewayDomainName

A domain name for the API.

Type: string

Required: False

certificateArn

An Amazon-managed certificate that will be used by the edge-optimized endpoint for this domain name. Amazon Certificate Manager is the only supported source.

Type: string

Required: False

certificateName

The user-friendly name of the certificate that will be used by the edge-optimized endpoint for this domain name.

Type: string

Required: False

certificateUploadDate

The timestamp when the certificate that was used by edge-optimized endpoint for this domain name was uploaded.

Type: string

Required: False

Format: date-time

domainNameStatus

Identifies the status of a domain name migration. Statuses can have a value of AVAILABLE, UPDATING, PENDING_CERTIFICATE_REIMPORT, or PENDING_OWNERSHIP_VERIFICATION. A domain

can be modified if its status is AVAILABLE. If the domain's status is UPDATING, you must wait until the current operation is complete.

Type: [DomainNameStatus](#)

Required: False

domainNameStatusMessage

An optional text message containing detailed information about status of the domain name migration.

Type: string

Required: False

endpointType

The endpoint type.

Type: [EndpointType](#)

Required: False

hostedZoneId

The Amazon Route 53 Hosted Zone ID of the endpoint.

Type: string

Required: False

ipAddressType

The IP address types that can invoke the domain name. Use `ipv4` to allow only IPv4 addresses to invoke your domain name, or use `dualstack` to allow both IPv4 and IPv6 addresses to invoke your domain name.

Type: [IpAddressType](#)

Required: False

ownershipVerificationCertificateArn

The Amazon resource name (ARN) for the public certificate issued by Amazon Certificate Manager. This ARN is used to validate custom domain ownership. It's required only if you configure mutual TLS and use either an ACM-imported or a private CA certificate ARN as the regionalCertificateArn.

Type: string

Required: False

securityPolicy

The Transport Layer Security (TLS) version of the security policy for this domain name. The valid values are TLS_1_0 and TLS_1_2.

Type: [SecurityPolicy](#)

Required: False

DomainNameStatus

Identifies the status of a domain name migration. Statuses can have a value of AVAILABLE, UPDATING, PENDING_CERTIFICATE_REIMPORT, or PENDING_OWNERSHIP_VERIFICATION. A domain can be modified if its status is AVAILABLE. If the domain's status is UPDATING, you must wait until the current operation is complete.

AVAILABLE

UPDATING

PENDING_CERTIFICATE_REIMPORT

PENDING_OWNERSHIP_VERIFICATION

EndpointType

Represents an endpoint type.

REGIONAL

EDGE

IpAddressType

The IP address types that can invoke your API or domain name.

ipv4
dualstack

LimitExceededException

A limit has been exceeded. See the accompanying error message for details.

message

Describes the error encountered.

Type: string
Required: False

limitType

The limit type.

Type: string
Required: False

MutualTlsAuthentication

If specified, API Gateway performs two-way authentication between the client and the server. Clients must present a trusted certificate to access your API.

truststoreUri

An Amazon S3 URL that specifies the truststore for mutual TLS authentication, for example, `s3://bucket-name/key-name` . The truststore can contain certificates from public or private certificate authorities. To update the truststore, upload a new version to S3, and then update your custom domain name to use the new version. To update the truststore, you must have permissions to access the S3 object.

Type: string
Required: False

truststoreVersion

The version of the S3 object that contains your truststore. To specify a version, you must have versioning enabled for the S3 bucket.

Type: string

Required: False

truststoreWarnings

A list of warnings that API Gateway returns while processing your truststore. Invalid certificates produce warnings. Mutual TLS is still enabled, but some clients might not be able to access your API. To resolve warnings, upload a new truststore to S3, and then update your domain name to use the new version.

Type: Array of type string

Required: False

MutualTlsAuthenticationInput

If specified, API Gateway performs two-way authentication between the client and the server. Clients must present a trusted certificate to access your API.

truststoreUri

An Amazon S3 URL that specifies the truststore for mutual TLS authentication, for example, `s3://bucket-name/key-name` . The truststore can contain certificates from public or private certificate authorities. To update the truststore, upload a new version to S3, and then update your custom domain name to use the new version. To update the truststore, you must have permissions to access the S3 object.

Type: string

Required: False

truststoreVersion

The version of the S3 object that contains your truststore. To specify a version, you must have versioning enabled for the S3 bucket.

Type: string

Required: False

NotFoundException

The resource specified in the request was not found. See the message field for more information.

message

Describes the error encountered.

Type: string

Required: False

resourceType

The resource type.

Type: string

Required: False

SecurityPolicy

The Transport Layer Security (TLS) version of the security policy for this domain name. The valid values are TLS_1_0 and TLS_1_2.

TLS_1_0

TLS_1_2

Tags

Represents a collection of tags associated with the resource.

key-value pairs

Type: string

UpdateDomainNameInput

Represents the input parameters for an UpdateDomainName request.

domainNameConfigurations

The domain name configurations.

Type: Array of type [DomainNameConfiguration](#)

Required: False

mutualTlsAuthentication

The mutual TLS authentication configuration for a custom domain name.

Type: [MutualTlsAuthenticationInput](#)

Required: False

DomainNames

Represents a collection of domain names. See [Set up Custom Domain Name for an API in API Gateway](#).

URI

/v2/domainnames

HTTP methods

GET

Operation ID: GetDomainNames

Gets the domain names for an Amazon Web Services account.

Query parameters

Name	Type	Required	Description
nextToken	String	False	The next page of elements from this

Name	Type	Required	Description
maxResults	String	False	collection. Not valid for the last element of the collection. The maximum number of elements to be returned for this resource.

Responses

Status code	Response model	Description
200	DomainNames	Success
400	BadRequestException	One of the parameters in the request is invalid.
404	NotFoundException	The resource specified in the request was not found.
429	LimitExceededException	The client is sending more than the allowed number of requests per unit of time.

POST

Operation ID: CreateDomainName

Creates a domain name.

Responses

Status code	Response model	Description
201	DomainName	The request has succeeded and has resulted in the creation of a resource.
400	BadRequestException	One of the parameters in the request is invalid.
403	AccessDeniedException	403 response
404	NotFoundException	The resource specified in the request was not found.
409	ConflictException	The resource already exists.
429	LimitExceededException	The client is sending more than the allowed number of requests per unit of time.

Schemas

Request bodies

POST schema

```
{
  "domainName": "string",
  "domainNameConfigurations": [
    {
      "apiGatewayDomainName": "string",
      "certificateArn": "string",
      "certificateName": "string",
      "certificateUploadDate": "string",
      "domainNameStatus": enum,
      "domainNameStatusMessage": "string",
      "endpointType": enum,
      "hostedZoneId": "string",
    }
  ]
}
```

```

    "ipAddressType": enum,
    "ownershipVerificationCertificateArn": "string",
    "securityPolicy": enum
  }
],
"tags": {
},
"mutualTlsAuthentication": {
  "truststoreUri": "string",
  "truststoreVersion": "string"
}
}

```

Response bodies

DomainNames schema

```

{
  "items": [
    {
      "domainName": "string",
      "domainNameConfigurations": [
        {
          "apiGatewayDomainName": "string",
          "certificateArn": "string",
          "certificateName": "string",
          "certificateUploadDate": "string",
          "domainNameStatus": enum,
          "domainNameStatusMessage": "string",
          "endpointType": enum,
          "hostedZoneId": "string",
          "ipAddressType": enum,
          "ownershipVerificationCertificateArn": "string",
          "securityPolicy": enum
        }
      ],
      "apiMappingSelectionExpression": "string",
      "tags": {
    },
    "mutualTlsAuthentication": {
      "truststoreUri": "string",
      "truststoreVersion": "string",
      "truststoreWarnings": [

```

```

        "string"
      ]
    }
  },
  "nextToken": "string"
}

```

DomainName schema

```

{
  "domainName": "string",
  "domainNameConfigurations": [
    {
      "apiGatewayDomainName": "string",
      "certificateArn": "string",
      "certificateName": "string",
      "certificateUploadDate": "string",
      "domainNameStatus": enum,
      "domainNameStatusMessage": "string",
      "endpointType": enum,
      "hostedZoneId": "string",
      "ipAddressType": enum,
      "ownershipVerificationCertificateArn": "string",
      "securityPolicy": enum
    }
  ],
  "apiMappingSelectionExpression": "string",
  "tags": {
  },
  "mutualTlsAuthentication": {
    "truststoreUri": "string",
    "truststoreVersion": "string",
    "truststoreWarnings": [
      "string"
    ]
  }
}

```

BadRequestException schema

```

{

```

```
"message": "string"
}
```

AccessDeniedException schema

```
{
  "message": "string"
}
```

NotFoundException schema

```
{
  "message": "string",
  "resourceType": "string"
}
```

ConflictException schema

```
{
  "message": "string"
}
```

LimitExceededException schema

```
{
  "message": "string",
  "limitType": "string"
}
```

Properties

AccessDeniedException

message

Type: string

Required: False

BadRequestException

The request is not valid, for example, the input is incomplete or incorrect. See the accompanying error message for details.

message

Describes the error encountered.

Type: string

Required: False

ConflictException

The requested operation would cause a conflict with the current state of a service resource associated with the request. Resolve the conflict before retrying this request. See the accompanying error message for details.

message

Describes the error encountered.

Type: string

Required: False

CreateDomainNameInput

Represents the input parameters for a `CreateDomainName` request.

domainName

The domain name.

Type: string

Required: True

domainNameConfigurations

The domain name configurations.

Type: Array of type [DomainNameConfiguration](#)

Required: False

tags

The collection of tags associated with a domain name.

Type: [Tags](#)

Required: False

mutualTlsAuthentication

The mutual TLS authentication configuration for a custom domain name.

Type: [MutualTlsAuthenticationInput](#)

Required: False

DomainName

Represents a domain name.

domainName

The name of the DomainName resource.

Type: string

Required: True

domainNameConfigurations

The domain name configurations.

Type: Array of type [DomainNameConfiguration](#)

Required: False

apiMappingSelectionExpression

The API mapping selection expression.

Type: string

Required: False

tags

The collection of tags associated with a domain name.

Type: [Tags](#)

Required: False

mutualTlsAuthentication

The mutual TLS authentication configuration for a custom domain name.

Type: [MutualTlsAuthentication](#)

Required: False

DomainNameConfiguration

The domain name configuration.

apiGatewayDomainName

A domain name for the API.

Type: string

Required: False

certificateArn

An Amazon-managed certificate that will be used by the edge-optimized endpoint for this domain name. Amazon Certificate Manager is the only supported source.

Type: string

Required: False

certificateName

The user-friendly name of the certificate that will be used by the edge-optimized endpoint for this domain name.

Type: string

Required: False

certificateUploadDate

The timestamp when the certificate that was used by edge-optimized endpoint for this domain name was uploaded.

Type: string

Required: False

Format: date-time

domainNameStatus

Identifies the status of a domain name migration. Statuses can have a value of AVAILABLE, UPDATING, PENDING_CERTIFICATE_REIMPORT, or PENDING_OWNERSHIP_VERIFICATION. A domain can be modified if its status is AVAILABLE. If the domain's status is UPDATING, you must wait until the current operation is complete.

Type: [DomainNameStatus](#)

Required: False

domainNameStatusMessage

An optional text message containing detailed information about status of the domain name migration.

Type: string

Required: False

endpointType

The endpoint type.

Type: [EndpointType](#)

Required: False

hostedZoneId

The Amazon Route 53 Hosted Zone ID of the endpoint.

Type: string

Required: False

ipAddressType

The IP address types that can invoke the domain name. Use `ipv4` to allow only IPv4 addresses to invoke your domain name, or use `dualstack` to allow both IPv4 and IPv6 addresses to invoke your domain name.

Type: [IpAddressType](#)

Required: False

ownershipVerificationCertificateArn

The Amazon resource name (ARN) for the public certificate issued by Amazon Certificate Manager. This ARN is used to validate custom domain ownership. It's required only if you configure mutual TLS and use either an ACM-imported or a private CA certificate ARN as the `regionalCertificateArn`.

Type: string

Required: False

securityPolicy

The Transport Layer Security (TLS) version of the security policy for this domain name. The valid values are `TLS_1_0` and `TLS_1_2`.

Type: [SecurityPolicy](#)

Required: False

DomainNameStatus

Identifies the status of a domain name migration. Statuses can have a value of `AVAILABLE`, `UPDATING`, `PENDING_CERTIFICATE_REIMPORT`, or `PENDING_OWNERSHIP_VERIFICATION`. A domain can be modified if its status is `AVAILABLE`. If the domain's status is `UPDATING`, you must wait until the current operation is complete.

`AVAILABLE`

`UPDATING`

`PENDING_CERTIFICATE_REIMPORT`

PENDING_OWNERSHIP_VERIFICATION

DomainNames

Represents a collection of domain names.

items

The elements from this collection.

Type: Array of type [DomainName](#)

Required: False

nextToken

The next page of elements from this collection. Not valid for the last element of the collection.

Type: string

Required: False

EndpointType

Represents an endpoint type.

REGIONAL

EDGE

IpAddressType

The IP address types that can invoke your API or domain name.

ipv4

dualstack

LimitExceededException

A limit has been exceeded. See the accompanying error message for details.

message

Describes the error encountered.

Type: string

Required: False

limitType

The limit type.

Type: string

Required: False

MutualTlsAuthentication

If specified, API Gateway performs two-way authentication between the client and the server. Clients must present a trusted certificate to access your API.

truststoreUri

An Amazon S3 URL that specifies the truststore for mutual TLS authentication, for example, `s3://bucket-name/key-name` . The truststore can contain certificates from public or private certificate authorities. To update the truststore, upload a new version to S3, and then update your custom domain name to use the new version. To update the truststore, you must have permissions to access the S3 object.

Type: string

Required: False

truststoreVersion

The version of the S3 object that contains your truststore. To specify a version, you must have versioning enabled for the S3 bucket.

Type: string

Required: False

truststoreWarnings

A list of warnings that API Gateway returns while processing your truststore. Invalid certificates produce warnings. Mutual TLS is still enabled, but some clients might not be able to access your API. To resolve warnings, upload a new truststore to S3, and then update your domain name to use the new version.

Type: Array of type string

Required: False

MutualTlsAuthenticationInput

If specified, API Gateway performs two-way authentication between the client and the server. Clients must present a trusted certificate to access your API.

truststoreUri

An Amazon S3 URL that specifies the truststore for mutual TLS authentication, for example, `s3://bucket-name/key-name` . The truststore can contain certificates from public or private certificate authorities. To update the truststore, upload a new version to S3, and then update your custom domain name to use the new version. To update the truststore, you must have permissions to access the S3 object.

Type: string

Required: False

truststoreVersion

The version of the S3 object that contains your truststore. To specify a version, you must have versioning enabled for the S3 bucket.

Type: string

Required: False

NotFoundException

The resource specified in the request was not found. See the message field for more information.

message

Describes the error encountered.

Type: string

Required: False

resourceType

The resource type.

Type: string

Required: False

SecurityPolicy

The Transport Layer Security (TLS) version of the security policy for this domain name. The valid values are TLS_1_0 and TLS_1_2.

TLS_1_0

TLS_1_2

Tags

Represents a collection of tags associated with the resource.

key-value pairs

Type: string

ExportedAPI

Represents an exported definition of an API in a particular output format, for example, YAML. The API is serialized to the requested specification, for example, OpenAPI 3.0.

URI

`/v2/apis/apiId/exports/specification`

HTTP methods

GET

Operation ID: ExportApi

Exports a definition of an API in a particular output format and specification.

Path parameters

Name	Type	Required	Description
<i>specification</i>	String	True	The version of the API specification to use. 0AS30, for OpenAPI 3.0, is the only supported value.
<i>apiId</i>	String	True	The API identifier.

Query parameters

Name	Type	Required	Description
includeExtensions	String	False	Specifies whether to include API Gateway extension s in the exported API definition. API Gateway extension s are included by default.
stageName	String	False	The name of the API stage to export. If you don't specify this property, a representation of the latest

Name	Type	Required	Description
			API configuration is exported.
exportVersion	String	False	The version of the API Gateway export algorithm. API Gateway uses the latest version by default. Currently, the only supported version is 1.0.
outputType	String	True	The output type of the exported definition file. Valid values are JSON and YAML.

Responses

Status code	Response model	Description
200	ExportedApi	Success
400	BadRequestException	One of the parameters in the request is invalid.
404	NotFoundException	The resource specified in the request was not found.
429	LimitExceededException	The client is sending more than the allowed number of requests per unit of time.

Schemas

Response bodies

ExportedApi schema

```
"string"
```

BadRequestException schema

```
{  
  "message": "string"  
}
```

NotFoundException schema

```
{  
  "message": "string",  
  "resourceType": "string"  
}
```

LimitExceededException schema

```
{  
  "message": "string",  
  "limitType": "string"  
}
```

Properties

BadRequestException

The request is not valid, for example, the input is incomplete or incorrect. See the accompanying error message for details.

message

Describes the error encountered.

Type: string

Required: False

LimitExceededException

A limit has been exceeded. See the accompanying error message for details.

message

Describes the error encountered.

Type: string

Required: False

limitType

The limit type.

Type: string

Required: False

NotFoundException

The resource specified in the request was not found. See the message field for more information.

message

Describes the error encountered.

Type: string

Required: False

resourceType

The resource type.

Type: string

Required: False

Integration

Represents an API integration.

URI

/v2/apis/*apiId*/integrations/*integrationId*

HTTP methods

GET

Operation ID: GetIntegration

Gets an Integration.

Path parameters

Name	Type	Required	Description
<i>apiId</i>	String	True	The API identifier.
<i>integrationId</i>	String	True	The integration ID.

Responses

Status code	Response model	Description
200	Integration	Success
404	NotFoundException	The resource specified in the request was not found.
429	LimitExceededException	The client is sending more than the allowed number of requests per unit of time.

DELETE

Operation ID: DeleteIntegration

Deletes an Integration.

Path parameters

Name	Type	Required	Description
<i>apiId</i>	String	True	The API identifier.
<i>integrationId</i>	String	True	The integration ID.

Responses

Status code	Response model	Description
204	None	The request has succeeded , and there is no additional content to send in the response payload body.
404	NotFoundException	The resource specified in the request was not found.
429	LimitExceededException	The client is sending more than the allowed number of requests per unit of time.

PATCH

Operation ID: UpdateIntegration

Updates an Integration.

Path parameters

Name	Type	Required	Description
<i>apiId</i>	String	True	The API identifier.
<i>integrationId</i>	String	True	The integration ID.

Responses

Status code	Response model	Description
200	Integration	Success
400	BadRequestException	One of the parameters in the request is invalid.
404	NotFoundException	The resource specified in the request was not found.
409	ConflictException	The resource already exists.
429	LimitExceededException	The client is sending more than the allowed number of requests per unit of time.

Schemas

Request bodies

PATCH schema

```
{
  "description": "string",
  "integrationType": enum,
  "integrationSubtype": "string",
  "integrationMethod": "string",
  "integrationUri": "string",
  "tlsConfig": {
    "serverNameToVerify": "string"
  },
  "credentialsArn": "string",
  "connectionType": enum,
  "connectionId": "string",
  "requestParameters": {
  },
  "responseParameters": {
  },
  "requestTemplates": {
```

```
},
  "templateSelectionExpression": "string",
  "passthroughBehavior": enum,
  "contentHandlingStrategy": enum,
  "timeoutInMillis": integer,
  "payloadFormatVersion": "string"
}
```

Response bodies

Integration schema

```
{
  "integrationId": "string",
  "description": "string",
  "integrationType": enum,
  "integrationSubtype": "string",
  "integrationMethod": "string",
  "integrationUri": "string",
  "tlsConfig": {
    "serverNameToVerify": "string"
  },
  "credentialsArn": "string",
  "connectionType": enum,
  "connectionId": "string",
  "requestParameters": {
  },
  "responseParameters": {
  },
  "requestTemplates": {
  },
  "templateSelectionExpression": "string",
  "passthroughBehavior": enum,
  "contentHandlingStrategy": enum,
  "timeoutInMillis": integer,
  "integrationResponseSelectionExpression": "string",
  "payloadFormatVersion": "string",
  "apiGatewayManaged": boolean
}
```

BadRequestException schema

```
{
  "message": "string"
}
```

NotFoundException schema

```
{
  "message": "string",
  "resourceType": "string"
}
```

ConflictException schema

```
{
  "message": "string"
}
```

LimitExceededException schema

```
{
  "message": "string",
  "limitType": "string"
}
```

Properties

BadRequestException

The request is not valid, for example, the input is incomplete or incorrect. See the accompanying error message for details.

message

Describes the error encountered.

Type: string

Required: False

ConflictException

The requested operation would cause a conflict with the current state of a service resource associated with the request. Resolve the conflict before retrying this request. See the accompanying error message for details.

message

Describes the error encountered.

Type: string

Required: False

ConnectionType

Represents a connection type.

INTERNET

VPC_LINK

ContentHandlingStrategy

Specifies how to handle response payload content type conversions. Supported only for WebSocket APIs.

CONVERT_TO_BINARY

CONVERT_TO_TEXT

Integration

Represents an integration.

integrationId

Represents the identifier of an integration.

Type: string

Required: False

description

Represents the description of an integration.

Type: string

Required: False

integrationType

The integration type of an integration. One of the following:

AWS: for integrating the route or method request with an Amazon service action, including the Lambda function-invoking action. With the Lambda function-invoking action, this is referred to as the Lambda custom integration. With any other Amazon service action, this is known as Amazon integration. Supported only for WebSocket APIs.

AWS_PROXY: for integrating the route or method request with a Lambda function or other Amazon service action. This integration is also referred to as a Lambda proxy integration.

HTTP: for integrating the route or method request with an HTTP endpoint. This integration is also referred to as the HTTP custom integration. Supported only for WebSocket APIs.

HTTP_PROXY: for integrating the route or method request with an HTTP endpoint, with the client request passed through as-is. This is also referred to as HTTP proxy integration.

MOCK: for integrating the route or method request with API Gateway as a "loopback" endpoint without invoking any backend. Supported only for WebSocket APIs.

Type: [IntegrationType](#)

Required: False

integrationSubtype

Supported only for HTTP API AWS_PROXY integrations. Specifies the Amazon service action to invoke. To learn more, see [Integration subtype reference](#).

Type: string

Required: False

integrationMethod

Specifies the integration's HTTP method type.

Type: string

Required: False

integrationUri

For a Lambda integration, specify the URI of a Lambda function.

For an HTTP integration, specify a fully-qualified URL.

For an HTTP API private integration, specify the ARN of an Application Load Balancer listener, Network Load Balancer listener, or Amazon Cloud Map service. If you specify the ARN of an Amazon Cloud Map service, API Gateway uses `DiscoverInstances` to identify resources. You can use query parameters to target specific resources. To learn more, see [DiscoverInstances](#). For private integrations, all resources must be owned by the same Amazon Web Services account.

Type: string

Required: False

tlsConfig

The TLS configuration for a private integration. If you specify a TLS configuration, private integration traffic uses the HTTPS protocol. Supported only for HTTP APIs.

Type: [TlsConfig](#)

Required: False

credentialsArn

Specifies the credentials required for the integration, if any. For Amazon integrations, three options are available. To specify an IAM Role for API Gateway to assume, use the role's Amazon Resource Name (ARN). To require that the caller's identity be passed through from the request, specify the string `arn:aws:iam::*:user/*`. To use resource-based permissions on supported Amazon services, don't specify this parameter.

Type: string

Required: False

connectionType

The type of the network connection to the integration endpoint. Specify `INTERNET` for connections through the public routable internet or `VPC_LINK` for private connections between API Gateway and resources in a VPC. The default value is `INTERNET`.

Type: [ConnectionType](#)

Required: False

connectionId

The ID of the VPC link for a private integration. Supported only for HTTP APIs.

Type: string

Required: False

requestParameters

For WebSocket APIs, a key-value map specifying request parameters that are passed from the method request to the backend. The key is an integration request parameter name and the associated value is a method request parameter value or static value that must be enclosed within single quotes and pre-encoded as required by the backend. The method request parameter value must match the pattern of `method.request.{location}.{name}`, where `{location}` is `queryString`, `path`, or `header`; and `{name}` must be a valid and unique method request parameter name.

For HTTP API integrations with a specified `integrationSubtype`, request parameters are a key-value map specifying parameters that are passed to `AWS_PROXY` integrations. You can provide static values, or map request data, stage variables, or context variables that are evaluated at runtime. To learn more, see [Working with Amazon service integrations for HTTP APIs](#).

For HTTP API integrations without a specified `integrationSubtype` request parameters are a key-value map specifying how to transform HTTP requests before sending them to backend integrations. The key should follow the pattern `<action>:<header|queryString|path>.<location>`. The action can be `append`, `overwrite` or `remove`. For values, you can provide static values, or map request data, stage variables, or context variables that are evaluated at runtime. To learn more, see [Transforming API requests and responses](#).

Type: [IntegrationParameters](#)

Required: False

responseParameters

Supported only for HTTP APIs. You use response parameters to transform the HTTP response from a backend integration before returning the response to clients. Specify a key-value map from a selection key to response parameters. The selection key must be a valid HTTP status code within the range of 200-599. Response parameters are a key-value map. The key must match the pattern `<action>:<header>.<location>` or `overwrite.statuscode`. The action can be `append`, `overwrite` or `remove`. The value can be a static value, or map to response data, stage variables, or context variables that are evaluated at runtime. To learn more, see [Transforming API requests and responses](#).

Type: [ResponseParameters](#)

Required: False

requestTemplates

Represents a map of Velocity templates that are applied on the request payload based on the value of the Content-Type header sent by the client. The content type value is the key in this map, and the template (as a String) is the value. Supported only for WebSocket APIs.

Type: [TemplateMap](#)

Required: False

templateSelectionExpression

The template selection expression for the integration. Supported only for WebSocket APIs.

Type: string

Required: False

passthroughBehavior

Specifies the pass-through behavior for incoming requests based on the Content-Type header in the request, and the available mapping templates specified as the `requestTemplates` property on the Integration resource. There are three valid values: `WHEN_NO_MATCH`, `WHEN_NO_TEMPLATES`, and `NEVER`. Supported only for WebSocket APIs.

WHEN_NO_MATCH passes the request body for unmapped content types through to the integration backend without transformation.

NEVER rejects unmapped content types with an HTTP 415 Unsupported Media Type response.

WHEN_NO_TEMPLATES allows pass-through when the integration has no content types mapped to templates. However, if there is at least one content type defined, unmapped content types will be rejected with the same HTTP 415 Unsupported Media Type response.

Type: [PassthroughBehavior](#)

Required: False

contentHandlingStrategy

Supported only for WebSocket APIs. Specifies how to handle response payload content type conversions. Supported values are CONVERT_TO_BINARY and CONVERT_TO_TEXT, with the following behaviors:

CONVERT_TO_BINARY: Converts a response payload from a Base64-encoded string to the corresponding binary blob.

CONVERT_TO_TEXT: Converts a response payload from a binary blob to a Base64-encoded string.

If this property is not defined, the response payload will be passed through from the integration response to the route response or method response without modification.

Type: [ContentHandlingStrategy](#)

Required: False

timeoutInMillis

Custom timeout between 50 and 29,000 milliseconds for WebSocket APIs and between 50 and 30,000 milliseconds for HTTP APIs. The default timeout is 29 seconds for WebSocket APIs and 30 seconds for HTTP APIs.

Type: integer

Required: False

integrationResponseSelectionExpression

The integration response selection expression for the integration. Supported only for WebSocket APIs. See [Integration Response Selection Expressions](#).

Type: string

Required: False

payloadFormatVersion

Specifies the format of the payload sent to an integration. Required for HTTP APIs. For HTTP APIs, supported values for Lambda proxy integrations are 1.0 and 2.0. For all other integrations, 1.0 is the only supported value. To learn more, see [Working with Amazon Lambda proxy integrations for HTTP APIs](#).

Type: string

Required: False

apiGatewayManaged

Specifies whether an integration is managed by API Gateway. If you created an API using using quick create, the resulting integration is managed by API Gateway. You can update a managed integration, but you can't delete it.

Type: boolean

Required: False

IntegrationParameters

For WebSocket APIs, a key-value map specifying request parameters that are passed from the method request to the backend. The key is an integration request parameter name and the associated value is a method request parameter value or static value that must be enclosed within single quotes and pre-encoded as required by the backend. The method request parameter value must match the pattern of `method.request.{location}.{name}`, where `{location}` is `queryString`, `path`, or `header`; and `{name}` must be a valid and unique method request parameter name.

For HTTP API integrations with a specified `integrationSubtype`, request parameters are a key-value map specifying parameters that are passed to `AWS_PROXY` integrations. You can provide

static values, or map request data, stage variables, or context variables that are evaluated at runtime. To learn more, see [Working with Amazon service integrations for HTTP APIs](#).

For HTTP API integrations without a specified `integrationSubtype` request parameters are a key-value map specifying how to transform HTTP requests before sending them to the backend. The key should follow the pattern `<action>:<header|querystring|path>.<location>` where action can be `append`, `overwrite` or `remove`. For values, you can provide static values, or map request data, stage variables, or context variables that are evaluated at runtime. To learn more, see [Transforming API requests and responses](#).

key-value pairs

Type: string

IntegrationType

Represents an API method integration type.

AWS

HTTP

MOCK

HTTP_PROXY

AWS_PROXY

LimitExceededException

A limit has been exceeded. See the accompanying error message for details.

message

Describes the error encountered.

Type: string

Required: False

limitType

The limit type.

Type: string

Required: False

NotFoundException

The resource specified in the request was not found. See the message field for more information.

message

Describes the error encountered.

Type: string

Required: False

resourceType

The resource type.

Type: string

Required: False

PassthroughBehavior

Represents passthrough behavior for an integration response. Supported only for WebSocket APIs.

WHEN_NO_MATCH

NEVER

WHEN_NO_TEMPLATES

ResponseParameters

Supported only for HTTP APIs. You use response parameters to transform the HTTP response from a backend integration before returning the response to clients.

key-value pairs

Type: object

TemplateMap

A mapping of identifier keys to templates. The value is an actual template script. The key is typically a `SelectionKey` which is chosen based on evaluating a selection expression.

key-value pairs

Type: string

TlsConfig

The TLS configuration for a private integration. If you specify a TLS configuration, private integration traffic uses the HTTPS protocol. Supported only for HTTP APIs.

serverNameToVerify

If you specify a server name, API Gateway uses it to verify the hostname on the integration's certificate. The server name is also included in the TLS handshake to support Server Name Indication (SNI) or virtual hosting.

Type: string

Required: False

TlsConfigInput

The TLS configuration for a private integration. If you specify a TLS configuration, private integration traffic uses the HTTPS protocol. Supported only for HTTP APIs.

serverNameToVerify

If you specify a server name, API Gateway uses it to verify the hostname on the integration's certificate. The server name is also included in the TLS handshake to support Server Name Indication (SNI) or virtual hosting.

Type: string

Required: False

UpdateIntegrationInput

Represents the input parameters for an `UpdateIntegration` request.

description

The description of the integration

Type: string

Required: False

integrationType

The integration type of an integration. One of the following:

AWS: for integrating the route or method request with an Amazon service action, including the Lambda function-invoking action. With the Lambda function-invoking action, this is referred to as the Lambda custom integration. With any other Amazon service action, this is known as Amazon integration. Supported only for WebSocket APIs.

AWS_PROXY: for integrating the route or method request with a Lambda function or other Amazon service action. This integration is also referred to as a Lambda proxy integration.

HTTP: for integrating the route or method request with an HTTP endpoint. This integration is also referred to as the HTTP custom integration. Supported only for WebSocket APIs.

HTTP_PROXY: for integrating the route or method request with an HTTP endpoint, with the client request passed through as-is. This is also referred to as HTTP proxy integration. For HTTP API private integrations, use an HTTP_PROXY integration.

MOCK: for integrating the route or method request with API Gateway as a "loopback" endpoint without invoking any backend. Supported only for WebSocket APIs.

Type: [IntegrationType](#)

Required: False

integrationSubtype

Supported only for HTTP API AWS_PROXY integrations. Specifies the Amazon service action to invoke. To learn more, see [Integration subtype reference](#).

Type: string

Required: False

integrationMethod

Specifies the integration's HTTP method type. For WebSocket APIs, if you use a Lambda integration, you must set the integration method to POST.

Type: string

Required: False

integrationUri

For a Lambda integration, specify the URI of a Lambda function.

For an HTTP integration, specify a fully-qualified URL.

For an HTTP API private integration, specify the ARN of an Application Load Balancer listener, Network Load Balancer listener, or Amazon Cloud Map service. If you specify the ARN of an Amazon Cloud Map service, API Gateway uses `DiscoverInstances` to identify resources. You can use query parameters to target specific resources. To learn more, see [DiscoverInstances](#). For private integrations, all resources must be owned by the same Amazon Web Services account.

Type: string

Required: False

tlsConfig

The TLS configuration for a private integration. If you specify a TLS configuration, private integration traffic uses the HTTPS protocol. Supported only for HTTP APIs.

Type: [TlsConfigInput](#)

Required: False

credentialsArn

Specifies the credentials required for the integration, if any. For Amazon integrations, three options are available. To specify an IAM Role for API Gateway to assume, use the role's Amazon Resource Name (ARN). To require that the caller's identity be passed through from the request, specify the string `arn:aws:iam::*:user/*`. To use resource-based permissions on supported Amazon services, don't specify this parameter.

Type: string

Required: False

connectionType

The type of the network connection to the integration endpoint. Specify `INTERNET` for connections through the public routable internet or `VPC_LINK` for private connections between API Gateway and resources in a VPC. The default value is `INTERNET`.

Type: [ConnectionType](#)

Required: False

connectionId

The ID of the VPC link for a private integration. Supported only for HTTP APIs.

Type: string

Required: False

requestParameters

For WebSocket APIs, a key-value map specifying request parameters that are passed from the method request to the backend. The key is an integration request parameter name and the associated value is a method request parameter value or static value that must be enclosed within single quotes and pre-encoded as required by the backend. The method request parameter value must match the pattern of `method.request.{location}.{name}`, where `{location}` is `queryString`, `path`, or `header`; and `{name}` must be a valid and unique method request parameter name.

For HTTP API integrations with a specified `integrationSubtype`, request parameters are a key-value map specifying parameters that are passed to `AWS_PROXY` integrations. You can provide static values, or map request data, stage variables, or context variables that are evaluated at runtime. To learn more, see [Working with Amazon service integrations for HTTP APIs](#).

For HTTP API integrations without a specified `integrationSubtype` request parameters are a key-value map specifying how to transform HTTP requests before sending them to the backend. The key should follow the pattern `<action>:<header|queryString|path>.<location>` where `action` can be `append`, `overwrite` or `remove`. For values, you can provide static values, or map request

data, stage variables, or context variables that are evaluated at runtime. To learn more, see [Transforming API requests and responses](#).

Type: [IntegrationParameters](#)

Required: False

responseParameters

Supported only for HTTP APIs. You use response parameters to transform the HTTP response from a backend integration before returning the response to clients. Specify a key-value map from a selection key to response parameters. The selection key must be a valid HTTP status code within the range of 200-599. Response parameters are a key-value map. The key must match the pattern `<action>:<header>.<location>` or `overwrite.statuscode`. The action can be `append`, `overwrite` or `remove`. The value can be a static value, or map to response data, stage variables, or context variables that are evaluated at runtime. To learn more, see [Transforming API requests and responses](#).

Type: [ResponseParameters](#)

Required: False

requestTemplates

Represents a map of Velocity templates that are applied on the request payload based on the value of the Content-Type header sent by the client. The content type value is the key in this map, and the template (as a String) is the value. Supported only for WebSocket APIs.

Type: [TemplateMap](#)

Required: False

templateSelectionExpression

The template selection expression for the integration.

Type: string

Required: False

passthroughBehavior

Specifies the pass-through behavior for incoming requests based on the Content-Type header in the request, and the available mapping templates specified as the `requestTemplates` property on the Integration resource. There are three valid values: `WHEN_NO_MATCH`, `WHEN_NO_TEMPLATES`, and `NEVER`. Supported only for WebSocket APIs.

`WHEN_NO_MATCH` passes the request body for unmapped content types through to the integration backend without transformation.

`NEVER` rejects unmapped content types with an HTTP 415 Unsupported Media Type response.

`WHEN_NO_TEMPLATES` allows pass-through when the integration has no content types mapped to templates. However, if there is at least one content type defined, unmapped content types will be rejected with the same HTTP 415 Unsupported Media Type response.

Type: [PassthroughBehavior](#)

Required: False

contentHandlingStrategy

Supported only for WebSocket APIs. Specifies how to handle response payload content type conversions. Supported values are `CONVERT_TO_BINARY` and `CONVERT_TO_TEXT`, with the following behaviors:

`CONVERT_TO_BINARY`: Converts a response payload from a Base64-encoded string to the corresponding binary blob.

`CONVERT_TO_TEXT`: Converts a response payload from a binary blob to a Base64-encoded string.

If this property is not defined, the response payload will be passed through from the integration response to the route response or method response without modification.

Type: [ContentHandlingStrategy](#)

Required: False

timeoutInMillis

Custom timeout between 50 and 29,000 milliseconds for WebSocket APIs and between 50 and 30,000 milliseconds for HTTP APIs. The default timeout is 29 seconds for WebSocket APIs and 30 seconds for HTTP APIs.

Type: integer
Required: False

payloadFormatVersion

Specifies the format of the payload sent to an integration. Required for HTTP APIs. For HTTP APIs, supported values for Lambda proxy integrations are 1.0 and 2.0. For all other integrations, 1.0 is the only supported value. To learn more, see [Working with Amazon Lambda proxy integrations for HTTP APIs](#).

Type: string
Required: False

IntegrationResponse

Represents an integration response. Supported only for WebSocket APIs.

URI

/v2/apis/*apiId*/integrations/*integrationId*/
integrationresponses/*integrationResponseId*

HTTP methods

GET

Operation ID: GetIntegrationResponse

Gets an IntegrationResponses.

Path parameters

Name	Type	Required	Description
<i>apiId</i>	String	True	The API identifier.
<i>integrationResponseId</i>	String	True	The integration response ID.
<i>integrationId</i>	String	True	The integration ID.

Responses

Status code	Response model	Description
200	IntegrationResponse	Success
404	NotFoundException	The resource specified in the request was not found.
429	LimitExceededException	The client is sending more than the allowed number of requests per unit of time.

DELETE

Operation ID: DeleteIntegrationResponse

Deletes an IntegrationResponses.

Path parameters

Name	Type	Required	Description
<i>apiId</i>	String	True	The API identifier.
<i>integrationResponseId</i>	String	True	The integration response ID.
<i>integrationId</i>	String	True	The integration ID.

Responses

Status code	Response model	Description
204	None	The request has succeeded , and there is no additional content to send in the response payload body.

Status code	Response model	Description
404	NotFoundException	The resource specified in the request was not found.
429	LimitExceededException	The client is sending more than the allowed number of requests per unit of time.

PATCH

Operation ID: UpdateIntegrationResponse

Updates an IntegrationResponses.

Path parameters

Name	Type	Required	Description
<i>apiId</i>	String	True	The API identifier.
<i>integrationResponseId</i>	String	True	The integration response ID.
<i>integrationId</i>	String	True	The integration ID.

Responses

Status code	Response model	Description
200	IntegrationResponse	Success
400	BadRequestException	One of the parameters in the request is invalid.
404	NotFoundException	The resource specified in the request was not found.
409	ConflictException	The resource already exists.

Status code	Response model	Description
429	LimitExceededException	The client is sending more than the allowed number of requests per unit of time.

Schemas

Request bodies

PATCH schema

```
{
  "integrationResponseKey": "string",
  "responseParameters": {
  },
  "responseTemplates": {
  },
  "templateSelectionExpression": "string",
  "contentHandlingStrategy": enum
}
```

Response bodies

IntegrationResponse schema

```
{
  "integrationResponseId": "string",
  "integrationResponseKey": "string",
  "responseParameters": {
  },
  "responseTemplates": {
  },
  "templateSelectionExpression": "string",
  "contentHandlingStrategy": enum
}
```

BadRequestException schema

```
{
```

```
"message": "string"
}
```

NotFoundException schema

```
{
  "message": "string",
  "resourceType": "string"
}
```

ConflictException schema

```
{
  "message": "string"
}
```

LimitExceededException schema

```
{
  "message": "string",
  "limitType": "string"
}
```

Properties

BadRequestException

The request is not valid, for example, the input is incomplete or incorrect. See the accompanying error message for details.

message

Describes the error encountered.

Type: string

Required: False

ConflictException

The requested operation would cause a conflict with the current state of a service resource associated with the request. Resolve the conflict before retrying this request. See the accompanying error message for details.

message

Describes the error encountered.

Type: string

Required: False

ContentHandlingStrategy

Specifies how to handle response payload content type conversions. Supported only for WebSocket APIs.

CONVERT_TO_BINARY

CONVERT_TO_TEXT

IntegrationParameters

For WebSocket APIs, a key-value map specifying request parameters that are passed from the method request to the backend. The key is an integration request parameter name and the associated value is a method request parameter value or static value that must be enclosed within single quotes and pre-encoded as required by the backend. The method request parameter value must match the pattern of `method.request.{location}.{name}`, where `{location}` is `queryString`, `path`, or `header`; and `{name}` must be a valid and unique method request parameter name.

For HTTP API integrations with a specified `integrationSubtype`, request parameters are a key-value map specifying parameters that are passed to `AWS_PROXY` integrations. You can provide static values, or map request data, stage variables, or context variables that are evaluated at runtime. To learn more, see [Working with Amazon service integrations for HTTP APIs](#).

For HTTP API integrations without a specified `integrationSubtype` request parameters are a key-value map specifying how to transform HTTP requests before sending them to the backend.

The key should follow the pattern `<action>:<header|querystring|path>.<location>` where action can be `append`, `overwrite` or `remove`. For values, you can provide static values, or map request data, stage variables, or context variables that are evaluated at runtime. To learn more, see [Transforming API requests and responses](#).

key-value pairs

Type: string

IntegrationResponse

Represents an integration response.

integrationResponseId

The integration response ID.

Type: string

Required: False

integrationResponseKey

The integration response key.

Type: string

Required: True

responseParameters

A key-value map specifying response parameters that are passed to the method response from the backend. The key is a method response header parameter name and the mapped value is an integration response header value, a static value enclosed within a pair of single quotes, or a JSON expression from the integration response body. The mapping key must match the pattern of `method.response.header.{name}`, where name is a valid and unique header name. The mapped non-static value must match the pattern of `integration.response.header.{name}` or `integration.response.body.{JSON-expression}`, where name is a valid and unique response header name and JSON-expression is a valid JSON expression without the \$ prefix.

Type: [IntegrationParameters](#)

Required: False

responseTemplates

The collection of response templates for the integration response as a string-to-string map of key-value pairs. Response templates are represented as a key/value map, with a content-type as the key and a template as the value.

Type: [TemplateMap](#)

Required: False

templateSelectionExpression

The template selection expressions for the integration response.

Type: string

Required: False

contentHandlingStrategy

Supported only for WebSocket APIs. Specifies how to handle response payload content type conversions. Supported values are `CONVERT_TO_BINARY` and `CONVERT_TO_TEXT`, with the following behaviors:

`CONVERT_TO_BINARY`: Converts a response payload from a Base64-encoded string to the corresponding binary blob.

`CONVERT_TO_TEXT`: Converts a response payload from a binary blob to a Base64-encoded string.

If this property is not defined, the response payload will be passed through from the integration response to the route response or method response without modification.

Type: [ContentHandlingStrategy](#)

Required: False

LimitExceededException

A limit has been exceeded. See the accompanying error message for details.

message

Describes the error encountered.

Type: string

Required: False

limitType

The limit type.

Type: string

Required: False

NotFoundException

The resource specified in the request was not found. See the message field for more information.

message

Describes the error encountered.

Type: string

Required: False

resourceType

The resource type.

Type: string

Required: False

TemplateMap

A mapping of identifier keys to templates. The value is an actual template script. The key is typically a `SelectionKey` which is chosen based on evaluating a selection expression.

key-value pairs

Type: string

UpdateIntegrationResponseInput

Represents the input parameters for an UpdateIntegrationResponse request.

integrationResponseKey

The integration response key.

Type: string

Required: False

responseParameters

A key-value map specifying response parameters that are passed to the method response from the backend. The key is a method response header parameter name and the mapped value is an integration response header value, a static value enclosed within a pair of single quotes, or a JSON expression from the integration response body. The mapping key must match the pattern of `method.response.header.{name}`, where `name` is a valid and unique header name. The mapped non-static value must match the pattern of `integration.response.header.{name}` or `integration.response.body.{JSON-expression}`, where `{name}` is a valid and unique response header name and `{JSON-expression}` is a valid JSON expression without the `$` prefix.

Type: [IntegrationParameters](#)

Required: False

responseTemplates

The collection of response templates for the integration response as a string-to-string map of key-value pairs. Response templates are represented as a key/value map, with a content-type as the key and a template as the value.

Type: [TemplateMap](#)

Required: False

templateSelectionExpression

The template selection expression for the integration response. Supported only for WebSocket APIs.

Type: string

Required: False

contentHandlingStrategy

Supported only for WebSocket APIs. Specifies how to handle response payload content type conversions. Supported values are CONVERT_TO_BINARY and CONVERT_TO_TEXT, with the following behaviors:

CONVERT_TO_BINARY: Converts a response payload from a Base64-encoded string to the corresponding binary blob.

CONVERT_TO_TEXT: Converts a response payload from a binary blob to a Base64-encoded string.

If this property is not defined, the response payload will be passed through from the integration response to the route response or method response without modification.

Type: [ContentHandlingStrategy](#)

Required: False

IntegrationResponses

Represents the collection of responses for an integration. Supported only for WebSocket APIs.

URI

/v2/apis/*apiId*/integrations/*integrationId*/integrationresponses

HTTP methods

GET

Operation ID: GetIntegrationResponses

Gets the IntegrationResponses for an Integration.

Path parameters

Name	Type	Required	Description
<i>integrationId</i>	String	True	The integration ID.

Name	Type	Required	Description
<i>apiId</i>	String	True	The API identifier.

Query parameters

Name	Type	Required	Description
nextToken	String	False	The next page of elements from this collection. Not valid for the last element of the collection.
maxResults	String	False	The maximum number of elements to be returned for this resource.

Responses

Status code	Response model	Description
200	IntegrationResponses	Success
400	BadRequestException	One of the parameters in the request is invalid.
404	NotFoundException	The resource specified in the request was not found.
429	LimitExceededException	The client is sending more than the allowed number of requests per unit of time.

POST

Operation ID: CreateIntegrationResponse

Creates an IntegrationResponses.

Path parameters

Name	Type	Required	Description
<i>apiId</i>	String	True	The API identifier.
<i>integrationId</i>	String	True	The integration ID.

Responses

Status code	Response model	Description
201	IntegrationResponse	The request has succeeded and has resulted in the creation of a resource.
400	BadRequestException	One of the parameters in the request is invalid.
404	NotFoundException	The resource specified in the request was not found.
409	ConflictException	The resource already exists.
429	LimitExceededException	The client is sending more than the allowed number of requests per unit of time.

Schemas

Request bodies

POST schema

```
{
  "integrationResponseKey": "string",
  "responseParameters": {
  },
  "responseTemplates": {
  },
  "templateSelectionExpression": "string",
  "contentHandlingStrategy": enum
}
```

Response bodies

IntegrationResponses schema

```
{
  "items": [
    {
      "integrationResponseId": "string",
      "integrationResponseKey": "string",
      "responseParameters": {
      },
      "responseTemplates": {
      },
      "templateSelectionExpression": "string",
      "contentHandlingStrategy": enum
    }
  ],
  "nextToken": "string"
}
```

IntegrationResponse schema

```
{
  "integrationResponseId": "string",
  "integrationResponseKey": "string",
  "responseParameters": {
  },
  "responseTemplates": {
  },
  "templateSelectionExpression": "string",
}
```

```
"contentHandlingStrategy": enum
}
```

BadRequestException schema

```
{
  "message": "string"
}
```

NotFoundException schema

```
{
  "message": "string",
  "resourceType": "string"
}
```

ConflictException schema

```
{
  "message": "string"
}
```

LimitExceededException schema

```
{
  "message": "string",
  "limitType": "string"
}
```

Properties

BadRequestException

The request is not valid, for example, the input is incomplete or incorrect. See the accompanying error message for details.

message

Describes the error encountered.

Type: string

Required: False

ConflictException

The requested operation would cause a conflict with the current state of a service resource associated with the request. Resolve the conflict before retrying this request. See the accompanying error message for details.

message

Describes the error encountered.

Type: string

Required: False

ContentHandlingStrategy

Specifies how to handle response payload content type conversions. Supported only for WebSocket APIs.

CONVERT_TO_BINARY

CONVERT_TO_TEXT

CreateIntegrationResponseInput

Represents the input parameters for a CreateIntegrationResponse request.

integrationResponseKey

The integration response key.

Type: string

Required: True

responseParameters

A key-value map specifying response parameters that are passed to the method response from the backend. The key is a method response header parameter name and the mapped value is

an integration response header value, a static value enclosed within a pair of single quotes, or a JSON expression from the integration response body. The mapping key must match the pattern of `method.response.header.{name}`, where `{name}` is a valid and unique header name. The mapped non-static value must match the pattern of `integration.response.header.{name}` or `integration.response.body.{JSON-expression}`, where `{name}` is a valid and unique response header name and `{JSON-expression}` is a valid JSON expression without the `$` prefix.

Type: [IntegrationParameters](#)

Required: False

responseTemplates

The collection of response templates for the integration response as a string-to-string map of key-value pairs. Response templates are represented as a key/value map, with a content-type as the key and a template as the value.

Type: [TemplateMap](#)

Required: False

templateSelectionExpression

The template selection expression for the integration response. Supported only for WebSocket APIs.

Type: string

Required: False

contentHandlingStrategy

Specifies how to handle response payload content type conversions. Supported values are `CONVERT_TO_BINARY` and `CONVERT_TO_TEXT`, with the following behaviors:

`CONVERT_TO_BINARY`: Converts a response payload from a Base64-encoded string to the corresponding binary blob.

`CONVERT_TO_TEXT`: Converts a response payload from a binary blob to a Base64-encoded string.

If this property is not defined, the response payload will be passed through from the integration response to the route response or method response without modification.

Type: [ContentHandlingStrategy](#)

Required: False

IntegrationParameters

For WebSocket APIs, a key-value map specifying request parameters that are passed from the method request to the backend. The key is an integration request parameter name and the associated value is a method request parameter value or static value that must be enclosed within single quotes and pre-encoded as required by the backend. The method request parameter value must match the pattern of `method.request.{location}.{name}`, where `{location}` is `queryString`, `path`, or `header`; and `{name}` must be a valid and unique method request parameter name.

For HTTP API integrations with a specified `integrationSubtype`, request parameters are a key-value map specifying parameters that are passed to `AWS_PROXY` integrations. You can provide static values, or map request data, stage variables, or context variables that are evaluated at runtime. To learn more, see [Working with Amazon service integrations for HTTP APIs](#).

For HTTP API integrations without a specified `integrationSubtype` request parameters are a key-value map specifying how to transform HTTP requests before sending them to the backend. The key should follow the pattern `<action>:<header|queryString|path>.<location>` where `action` can be `append`, `overwrite` or `remove`. For values, you can provide static values, or map request data, stage variables, or context variables that are evaluated at runtime. To learn more, see [Transforming API requests and responses](#).

key-value pairs

Type: string

IntegrationResponse

Represents an integration response.

integrationResponseId

The integration response ID.

Type: string

Required: False

integrationResponseKey

The integration response key.

Type: string

Required: True

responseParameters

A key-value map specifying response parameters that are passed to the method response from the backend. The key is a method response header parameter name and the mapped value is an integration response header value, a static value enclosed within a pair of single quotes, or a JSON expression from the integration response body. The mapping key must match the pattern of `method.response.header.{name}`, where `name` is a valid and unique header name. The mapped non-static value must match the pattern of `integration.response.header.{name}` or `integration.response.body.{JSON-expression}`, where `name` is a valid and unique response header name and `JSON-expression` is a valid JSON expression without the `$` prefix.

Type: [IntegrationParameters](#)

Required: False

responseTemplates

The collection of response templates for the integration response as a string-to-string map of key-value pairs. Response templates are represented as a key/value map, with a content-type as the key and a template as the value.

Type: [TemplateMap](#)

Required: False

templateSelectionExpression

The template selection expressions for the integration response.

Type: string

Required: False

contentHandlingStrategy

Supported only for WebSocket APIs. Specifies how to handle response payload content type conversions. Supported values are `CONVERT_TO_BINARY` and `CONVERT_TO_TEXT`, with the following behaviors:

`CONVERT_TO_BINARY`: Converts a response payload from a Base64-encoded string to the corresponding binary blob.

`CONVERT_TO_TEXT`: Converts a response payload from a binary blob to a Base64-encoded string.

If this property is not defined, the response payload will be passed through from the integration response to the route response or method response without modification.

Type: [ContentHandlingStrategy](#)

Required: False

IntegrationResponses

Represents a collection of integration responses.

items

The elements from this collection.

Type: Array of type [IntegrationResponse](#)

Required: False

nextToken

The next page of elements from this collection. Not valid for the last element of the collection.

Type: string

Required: False

LimitExceededException

A limit has been exceeded. See the accompanying error message for details.

message

Describes the error encountered.

Type: string

Required: False

limitType

The limit type.

Type: string

Required: False

NotFoundException

The resource specified in the request was not found. See the message field for more information.

message

Describes the error encountered.

Type: string

Required: False

resourceType

The resource type.

Type: string

Required: False

TemplateMap

A mapping of identifier keys to templates. The value is an actual template script. The key is typically a `SelectionKey` which is chosen based on evaluating a selection expression.

key-value pairs

Type: string

Integrations

Represents the collection of integrations for an API.

URI

/v2/apis/*apiId*/integrations

HTTP methods

GET

Operation ID: GetIntegrations

Gets the Integrations for an API.

Path parameters

Name	Type	Required	Description
<i>apiId</i>	String	True	The API identifier.

Query parameters

Name	Type	Required	Description
nextToken	String	False	The next page of elements from this collection. Not valid for the last element of the collection.
maxResults	String	False	The maximum number of elements to be returned for this resource.

Responses

Status code	Response model	Description
200	Integrations	Success
400	BadRequestException	One of the parameters in the request is invalid.
404	NotFoundException	The resource specified in the request was not found.
429	LimitExceededException	The client is sending more than the allowed number of requests per unit of time.

POST

Operation ID: CreateIntegration

Creates an Integration.

Path parameters

Name	Type	Required	Description
<i>apiId</i>	String	True	The API identifier.

Responses

Status code	Response model	Description
201	Integration	The request has succeeded and has resulted in the creation of a resource.
400	BadRequestException	One of the parameters in the request is invalid.

Status code	Response model	Description
404	NotFoundException	The resource specified in the request was not found.
409	ConflictException	The resource already exists.
429	LimitExceededException	The client is sending more than the allowed number of requests per unit of time.

Schemas

Request bodies

POST schema

```
{
  "description": "string",
  "integrationType": enum,
  "integrationSubtype": "string",
  "integrationMethod": "string",
  "integrationUri": "string",
  "tlsConfig": {
    "serverNameToVerify": "string"
  },
  "credentialsArn": "string",
  "connectionType": enum,
  "connectionId": "string",
  "requestParameters": {
  },
  "responseParameters": {
  },
  "requestTemplates": {
  },
  "templateSelectionExpression": "string",
  "passthroughBehavior": enum,
  "contentHandlingStrategy": enum,
  "timeoutInMillis": integer,
  "payloadFormatVersion": "string"
```

```
}
```

Response bodies

Integrations schema

```
{
  "items": [
    {
      "integrationId": "string",
      "description": "string",
      "integrationType": enum,
      "integrationSubtype": "string",
      "integrationMethod": "string",
      "integrationUri": "string",
      "tlsConfig": {
        "serverNameToVerify": "string"
      },
      "credentialsArn": "string",
      "connectionType": enum,
      "connectionId": "string",
      "requestParameters": {
      },
      "responseParameters": {
      },
      "requestTemplates": {
      },
      "templateSelectionExpression": "string",
      "passthroughBehavior": enum,
      "contentHandlingStrategy": enum,
      "timeoutInMillis": integer,
      "integrationResponseSelectionExpression": "string",
      "payloadFormatVersion": "string",
      "apiGatewayManaged": boolean
    }
  ],
  "nextToken": "string"
}
```

Integration schema

```
{
```



```
"integrationId": "string",
"description": "string",
"integrationType": enum,
"integrationSubtype": "string",
"integrationMethod": "string",
"integrationUri": "string",
"tlsConfig": {
  "serverNameToVerify": "string"
},
"credentialsArn": "string",
"connectionType": enum,
"connectionId": "string",
"requestParameters": {
},
"responseParameters": {
},
"requestTemplates": {
},
"templateSelectionExpression": "string",
"passthroughBehavior": enum,
"contentHandlingStrategy": enum,
"timeoutInMillis": integer,
"integrationResponseSelectionExpression": "string",
"payloadFormatVersion": "string",
"apiGatewayManaged": boolean
}
```

BadRequestException schema

```
{
  "message": "string"
}
```

NotFoundException schema

```
{
  "message": "string",
  "resourceType": "string"
}
```

ConflictException schema

```
{  
  "message": "string"  
}
```

LimitExceededException schema

```
{  
  "message": "string",  
  "limitType": "string"  
}
```

Properties

BadRequestException

The request is not valid, for example, the input is incomplete or incorrect. See the accompanying error message for details.

message

Describes the error encountered.

Type: string

Required: False

ConflictException

The requested operation would cause a conflict with the current state of a service resource associated with the request. Resolve the conflict before retrying this request. See the accompanying error message for details.

message

Describes the error encountered.

Type: string

Required: False

ConnectionType

Represents a connection type.

INTERNET
VPC_LINK

ContentHandlingStrategy

Specifies how to handle response payload content type conversions. Supported only for WebSocket APIs.

CONVERT_TO_BINARY
CONVERT_TO_TEXT

CreateIntegrationInput

Represents the input parameters for a CreateIntegration request.

description

The description of the integration.

Type: string

Required: False

integrationType

The integration type of an integration. One of the following:

AWS: for integrating the route or method request with an Amazon service action, including the Lambda function-invoking action. With the Lambda function-invoking action, this is referred to as the Lambda custom integration. With any other Amazon service action, this is known as Amazon integration. Supported only for WebSocket APIs.

AWS_PROXY: for integrating the route or method request with a Lambda function or other Amazon service action. This integration is also referred to as a Lambda proxy integration.

HTTP: for integrating the route or method request with an HTTP endpoint. This integration is also referred to as the HTTP custom integration. Supported only for WebSocket APIs.

HTTP_PROXY: for integrating the route or method request with an HTTP endpoint, with the client request passed through as-is. This is also referred to as HTTP proxy integration. For HTTP API private integrations, use an HTTP_PROXY integration.

MOCK: for integrating the route or method request with API Gateway as a "loopback" endpoint without invoking any backend. Supported only for WebSocket APIs.

Type: [IntegrationType](#)

Required: True

integrationSubtype

Supported only for HTTP API AWS_PROXY integrations. Specifies the Amazon service action to invoke. To learn more, see [Integration subtype reference](#).

Type: string

Required: False

integrationMethod

Specifies the integration's HTTP method type. For WebSocket APIs, if you use a Lambda integration, you must set the integration method to POST.

Type: string

Required: False

integrationUri

For a Lambda integration, specify the URI of a Lambda function.

For an HTTP integration, specify a fully-qualified URL.

For an HTTP API private integration, specify the ARN of an Application Load Balancer listener, Network Load Balancer listener, or Amazon Cloud Map service. If you specify the ARN of an Amazon Cloud Map service, API Gateway uses `DiscoverInstances` to identify resources. You can

use query parameters to target specific resources. To learn more, see [DiscoverInstances](#). For private integrations, all resources must be owned by the same Amazon Web Services account.

Type: string

Required: False

tlsConfig

The TLS configuration for a private integration. If you specify a TLS configuration, private integration traffic uses the HTTPS protocol. Supported only for HTTP APIs.

Type: [TlsConfigInput](#)

Required: False

credentialsArn

Specifies the credentials required for the integration, if any. For Amazon integrations, three options are available. To specify an IAM Role for API Gateway to assume, use the role's Amazon Resource Name (ARN). To require that the caller's identity be passed through from the request, specify the string `arn:aws:iam::*:user/*`. To use resource-based permissions on supported Amazon services, don't specify this parameter.

Type: string

Required: False

connectionType

The type of the network connection to the integration endpoint. Specify `INTERNET` for connections through the public routable internet or `VPC_LINK` for private connections between API Gateway and resources in a VPC. The default value is `INTERNET`.

Type: [ConnectionType](#)

Required: False

connectionId

The ID of the VPC link for a private integration. Supported only for HTTP APIs.

Type: string

Required: False

requestParameters

For WebSocket APIs, a key-value map specifying request parameters that are passed from the method request to the backend. The key is an integration request parameter name and the associated value is a method request parameter value or static value that must be enclosed within single quotes and pre-encoded as required by the backend. The method request parameter value must match the pattern of `method.request.{location}.{name}`, where `{location}` is `queryString`, `path`, or `header`; and `{name}` must be a valid and unique method request parameter name.

For HTTP API integrations with a specified `integrationSubtype`, request parameters are a key-value map specifying parameters that are passed to `AWS_PROXY` integrations. You can provide static values, or map request data, stage variables, or context variables that are evaluated at runtime. To learn more, see [Working with Amazon service integrations for HTTP APIs](#).

For HTTP API integrations without a specified `integrationSubtype` request parameters are a key-value map specifying how to transform HTTP requests before sending them to the backend. The key should follow the pattern `<action>:<header|queryString|path>.<location>` where `action` can be `append`, `overwrite` or `remove`. For values, you can provide static values, or map request data, stage variables, or context variables that are evaluated at runtime. To learn more, see [Transforming API requests and responses](#).

Type: [IntegrationParameters](#)

Required: False

responseParameters

Supported only for HTTP APIs. You use response parameters to transform the HTTP response from a backend integration before returning the response to clients. Specify a key-value map from a selection key to response parameters. The selection key must be a valid HTTP status code within the range of 200-599. Response parameters are a key-value map. The key must match the pattern `<action>:<header>.<location>` or `overwrite.statuscode`. The action can be `append`, `overwrite` or `remove`. The value can be a static value, or map to response data, stage variables, or context variables that are evaluated at runtime. To learn more, see [Transforming API requests and responses](#).

Type: [ResponseParameters](#)

Required: False

requestTemplates

Represents a map of Velocity templates that are applied on the request payload based on the value of the Content-Type header sent by the client. The content type value is the key in this map, and the template (as a String) is the value. Supported only for WebSocket APIs.

Type: [TemplateMap](#)

Required: False

templateSelectionExpression

The template selection expression for the integration.

Type: string

Required: False

passthroughBehavior

Specifies the pass-through behavior for incoming requests based on the Content-Type header in the request, and the available mapping templates specified as the `requestTemplates` property on the Integration resource. There are three valid values: `WHEN_NO_MATCH`, `WHEN_NO_TEMPLATES`, and `NEVER`. Supported only for WebSocket APIs.

`WHEN_NO_MATCH` passes the request body for unmapped content types through to the integration backend without transformation.

`NEVER` rejects unmapped content types with an HTTP 415 Unsupported Media Type response.

`WHEN_NO_TEMPLATES` allows pass-through when the integration has no content types mapped to templates. However, if there is at least one content type defined, unmapped content types will be rejected with the same HTTP 415 Unsupported Media Type response.

Type: [PassthroughBehavior](#)

Required: False

contentHandlingStrategy

Supported only for WebSocket APIs. Specifies how to handle response payload content type conversions. Supported values are `CONVERT_TO_BINARY` and `CONVERT_TO_TEXT`, with the following behaviors:

`CONVERT_TO_BINARY`: Converts a response payload from a Base64-encoded string to the corresponding binary blob.

`CONVERT_TO_TEXT`: Converts a response payload from a binary blob to a Base64-encoded string.

If this property is not defined, the response payload will be passed through from the integration response to the route response or method response without modification.

Type: [ContentHandlingStrategy](#)

Required: False

timeoutInMillis

Custom timeout between 50 and 29,000 milliseconds for WebSocket APIs and between 50 and 30,000 milliseconds for HTTP APIs. The default timeout is 29 seconds for WebSocket APIs and 30 seconds for HTTP APIs.

Type: integer

Required: False

payloadFormatVersion

Specifies the format of the payload sent to an integration. Required for HTTP APIs. For HTTP APIs, supported values for Lambda proxy integrations are `1.0` and `2.0`. For all other integrations, `1.0` is the only supported value. To learn more, see [Working with Amazon Lambda proxy integrations for HTTP APIs](#).

Type: string

Required: False

Integration

Represents an integration.

integrationId

Represents the identifier of an integration.

Type: string

Required: False

description

Represents the description of an integration.

Type: string

Required: False

integrationType

The integration type of an integration. One of the following:

AWS: for integrating the route or method request with an Amazon service action, including the Lambda function-invoking action. With the Lambda function-invoking action, this is referred to as the Lambda custom integration. With any other Amazon service action, this is known as Amazon integration. Supported only for WebSocket APIs.

AWS_PROXY: for integrating the route or method request with a Lambda function or other Amazon service action. This integration is also referred to as a Lambda proxy integration.

HTTP: for integrating the route or method request with an HTTP endpoint. This integration is also referred to as the HTTP custom integration. Supported only for WebSocket APIs.

HTTP_PROXY: for integrating the route or method request with an HTTP endpoint, with the client request passed through as-is. This is also referred to as HTTP proxy integration.

MOCK: for integrating the route or method request with API Gateway as a "loopback" endpoint without invoking any backend. Supported only for WebSocket APIs.

Type: [IntegrationType](#)

Required: False

integrationSubtype

Supported only for HTTP API AWS_PROXY integrations. Specifies the Amazon service action to invoke. To learn more, see [Integration subtype reference](#).

Type: string

Required: False

integrationMethod

Specifies the integration's HTTP method type.

Type: string

Required: False

integrationUri

For a Lambda integration, specify the URI of a Lambda function.

For an HTTP integration, specify a fully-qualified URL.

For an HTTP API private integration, specify the ARN of an Application Load Balancer listener, Network Load Balancer listener, or Amazon Cloud Map service. If you specify the ARN of an Amazon Cloud Map service, API Gateway uses `DiscoverInstances` to identify resources. You can use query parameters to target specific resources. To learn more, see [DiscoverInstances](#). For private integrations, all resources must be owned by the same Amazon Web Services account.

Type: string

Required: False

tlsConfig

The TLS configuration for a private integration. If you specify a TLS configuration, private integration traffic uses the HTTPS protocol. Supported only for HTTP APIs.

Type: [TlsConfig](#)

Required: False

credentialsArn

Specifies the credentials required for the integration, if any. For Amazon integrations, three options are available. To specify an IAM Role for API Gateway to assume, use the role's Amazon Resource Name (ARN). To require that the caller's identity be passed through from the request, specify the string `arn:aws:iam::*:user/*`. To use resource-based permissions on supported Amazon services, don't specify this parameter.

Type: string

Required: False

connectionType

The type of the network connection to the integration endpoint. Specify `INTERNET` for connections through the public routable internet or `VPC_LINK` for private connections between API Gateway and resources in a VPC. The default value is `INTERNET`.

Type: [ConnectionType](#)

Required: False

connectionId

The ID of the VPC link for a private integration. Supported only for HTTP APIs.

Type: string

Required: False

requestParameters

For WebSocket APIs, a key-value map specifying request parameters that are passed from the method request to the backend. The key is an integration request parameter name and the associated value is a method request parameter value or static value that must be enclosed within single quotes and pre-encoded as required by the backend. The method request parameter value must match the pattern of `method.request.{location}.{name}`, where `{location}` is `queryString`, `path`, or `header`; and `{name}` must be a valid and unique method request parameter name.

For HTTP API integrations with a specified `integrationSubtype`, request parameters are a key-value map specifying parameters that are passed to `AWS_PROXY` integrations. You can provide

static values, or map request data, stage variables, or context variables that are evaluated at runtime. To learn more, see [Working with Amazon service integrations for HTTP APIs](#).

For HTTP API integrations without a specified `integrationSubtype` request parameters are a key-value map specifying how to transform HTTP requests before sending them to backend integrations. The key should follow the pattern `<action>:<header|querystring|path>.<location>`. The action can be `append`, `overwrite` or `remove`. For values, you can provide static values, or map request data, stage variables, or context variables that are evaluated at runtime. To learn more, see [Transforming API requests and responses](#).

Type: [IntegrationParameters](#)

Required: False

responseParameters

Supported only for HTTP APIs. You use response parameters to transform the HTTP response from a backend integration before returning the response to clients. Specify a key-value map from a selection key to response parameters. The selection key must be a valid HTTP status code within the range of 200-599. Response parameters are a key-value map. The key must match the pattern `<action>:<header>.<location>` or `overwrite.statuscode`. The action can be `append`, `overwrite` or `remove`. The value can be a static value, or map to response data, stage variables, or context variables that are evaluated at runtime. To learn more, see [Transforming API requests and responses](#).

Type: [ResponseParameters](#)

Required: False

requestTemplates

Represents a map of Velocity templates that are applied on the request payload based on the value of the Content-Type header sent by the client. The content type value is the key in this map, and the template (as a String) is the value. Supported only for WebSocket APIs.

Type: [TemplateMap](#)

Required: False

templateSelectionExpression

The template selection expression for the integration. Supported only for WebSocket APIs.

Type: string

Required: False

passthroughBehavior

Specifies the pass-through behavior for incoming requests based on the Content-Type header in the request, and the available mapping templates specified as the `requestTemplates` property on the Integration resource. There are three valid values: `WHEN_NO_MATCH`, `WHEN_NO_TEMPLATES`, and `NEVER`. Supported only for WebSocket APIs.

`WHEN_NO_MATCH` passes the request body for unmapped content types through to the integration backend without transformation.

`NEVER` rejects unmapped content types with an HTTP 415 Unsupported Media Type response.

`WHEN_NO_TEMPLATES` allows pass-through when the integration has no content types mapped to templates. However, if there is at least one content type defined, unmapped content types will be rejected with the same HTTP 415 Unsupported Media Type response.

Type: [PassthroughBehavior](#)

Required: False

contentHandlingStrategy

Supported only for WebSocket APIs. Specifies how to handle response payload content type conversions. Supported values are `CONVERT_TO_BINARY` and `CONVERT_TO_TEXT`, with the following behaviors:

`CONVERT_TO_BINARY`: Converts a response payload from a Base64-encoded string to the corresponding binary blob.

`CONVERT_TO_TEXT`: Converts a response payload from a binary blob to a Base64-encoded string.

If this property is not defined, the response payload will be passed through from the integration response to the route response or method response without modification.

Type: [ContentHandlingStrategy](#)

Required: False

timeoutInMillis

Custom timeout between 50 and 29,000 milliseconds for WebSocket APIs and between 50 and 30,000 milliseconds for HTTP APIs. The default timeout is 29 seconds for WebSocket APIs and 30 seconds for HTTP APIs.

Type: integer

Required: False

integrationResponseSelectionExpression

The integration response selection expression for the integration. Supported only for WebSocket APIs. See [Integration Response Selection Expressions](#).

Type: string

Required: False

payloadFormatVersion

Specifies the format of the payload sent to an integration. Required for HTTP APIs. For HTTP APIs, supported values for Lambda proxy integrations are 1.0 and 2.0. For all other integrations, 1.0 is the only supported value. To learn more, see [Working with Amazon Lambda proxy integrations for HTTP APIs](#).

Type: string

Required: False

apiGatewayManaged

Specifies whether an integration is managed by API Gateway. If you created an API using using quick create, the resulting integration is managed by API Gateway. You can update a managed integration, but you can't delete it.

Type: boolean

Required: False

IntegrationParameters

For WebSocket APIs, a key-value map specifying request parameters that are passed from the method request to the backend. The key is an integration request parameter name and the associated value is a method request parameter value or static value that must be enclosed within single quotes and pre-encoded as required by the backend. The method request parameter value must match the pattern of `method.request.{location}.{name}`, where `{location}` is `queryString`, `path`, or `header`; and `{name}` must be a valid and unique method request parameter name.

For HTTP API integrations with a specified `integrationSubtype`, request parameters are a key-value map specifying parameters that are passed to `AWS_PROXY` integrations. You can provide static values, or map request data, stage variables, or context variables that are evaluated at runtime. To learn more, see [Working with Amazon service integrations for HTTP APIs](#).

For HTTP API integrations without a specified `integrationSubtype` request parameters are a key-value map specifying how to transform HTTP requests before sending them to the backend. The key should follow the pattern `<action>:<header|queryString|path>.<location>` where `action` can be `append`, `overwrite` or `remove`. For values, you can provide static values, or map request data, stage variables, or context variables that are evaluated at runtime. To learn more, see [Transforming API requests and responses](#).

key-value pairs

Type: string

IntegrationType

Represents an API method integration type.

AWS
HTTP
MOCK
HTTP_PROXY
AWS_PROXY

Integrations

Represents a collection of integrations.

items

The elements from this collection.

Type: Array of type [Integration](#)

Required: False

nextToken

The next page of elements from this collection. Not valid for the last element of the collection.

Type: string

Required: False

LimitExceededException

A limit has been exceeded. See the accompanying error message for details.

message

Describes the error encountered.

Type: string

Required: False

limitType

The limit type.

Type: string

Required: False

NotFoundException

The resource specified in the request was not found. See the message field for more information.

message

Describes the error encountered.

Type: string

Required: False

resourceType

The resource type.

Type: string

Required: False

PassthroughBehavior

Represents passthrough behavior for an integration response. Supported only for WebSocket APIs.

WHEN_NO_MATCH

NEVER

WHEN_NO_TEMPLATES

ResponseParameters

Supported only for HTTP APIs. You use response parameters to transform the HTTP response from a backend integration before returning the response to clients.

key-value pairs

Type: object

TemplateMap

A mapping of identifier keys to templates. The value is an actual template script. The key is typically a `SelectionKey` which is chosen based on evaluating a selection expression.

key-value pairs

Type: string

TlsConfig

The TLS configuration for a private integration. If you specify a TLS configuration, private integration traffic uses the HTTPS protocol. Supported only for HTTP APIs.

serverNameToVerify

If you specify a server name, API Gateway uses it to verify the hostname on the integration's certificate. The server name is also included in the TLS handshake to support Server Name Indication (SNI) or virtual hosting.

Type: string

Required: False

TlsConfigInput

The TLS configuration for a private integration. If you specify a TLS configuration, private integration traffic uses the HTTPS protocol. Supported only for HTTP APIs.

serverNameToVerify

If you specify a server name, API Gateway uses it to verify the hostname on the integration's certificate. The server name is also included in the TLS handshake to support Server Name Indication (SNI) or virtual hosting.

Type: string

Required: False

Model

Represents a data model for an API. Supported only for WebSocket APIs. See [Create Models and Mapping Templates for Request and Response Mappings](#).

URI

`/v2/apis/apiId/models/modelId`

HTTP methods

GET

Operation ID: GetModel

Gets a Model.

Path parameters

Name	Type	Required	Description
<i>modelId</i>	String	True	The model ID.
<i>apiId</i>	String	True	The API identifier.

Responses

Status code	Response model	Description
200	Model	Success
404	NotFoundException	The resource specified in the request was not found.
429	LimitExceededException	The client is sending more than the allowed number of requests per unit of time.

DELETE

Operation ID: DeleteModel

Deletes a Model.

Path parameters

Name	Type	Required	Description
<i>modelId</i>	String	True	The model ID.
<i>apiId</i>	String	True	The API identifier.

Responses

Status code	Response model	Description
204	None	The request has succeeded , and there is no additional content to send in the response payload body.
404	NotFoundException	The resource specified in the request was not found.
429	LimitExceededException	The client is sending more than the allowed number of requests per unit of time.

PATCH

Operation ID: UpdateModel

Updates a Model.

Path parameters

Name	Type	Required	Description
<i>modelId</i>	String	True	The model ID.
<i>apiId</i>	String	True	The API identifier.

Responses

Status code	Response model	Description
200	Model	Success
400	BadRequestException	One of the parameters in the request is invalid.

Status code	Response model	Description
404	NotFoundException	The resource specified in the request was not found.
409	ConflictException	The resource already exists.
429	LimitExceededException	The client is sending more than the allowed number of requests per unit of time.

Schemas

Request bodies

PATCH schema

```
{
  "name": "string",
  "description": "string",
  "schema": "string",
  "contentType": "string"
}
```

Response bodies

Model schema

```
{
  "modelId": "string",
  "name": "string",
  "description": "string",
  "schema": "string",
  "contentType": "string"
}
```

BadRequestException schema

```
{
```

```
"message": "string"
}
```

NotFoundException schema

```
{
  "message": "string",
  "resourceType": "string"
}
```

ConflictException schema

```
{
  "message": "string"
}
```

LimitExceededException schema

```
{
  "message": "string",
  "limitType": "string"
}
```

Properties

BadRequestException

The request is not valid, for example, the input is incomplete or incorrect. See the accompanying error message for details.

message

Describes the error encountered.

Type: string

Required: False

ConflictException

The requested operation would cause a conflict with the current state of a service resource associated with the request. Resolve the conflict before retrying this request. See the accompanying error message for details.

message

Describes the error encountered.

Type: string

Required: False

LimitExceededException

A limit has been exceeded. See the accompanying error message for details.

message

Describes the error encountered.

Type: string

Required: False

limitType

The limit type.

Type: string

Required: False

Model

Represents a data model for an API. Supported only for WebSocket APIs. See [Create Models and Mapping Templates for Request and Response Mappings](#).

modelId

The model identifier.

Type: string

Required: False

name

The name of the model. Must be alphanumeric.

Type: string

Required: True

description

The description of the model.

Type: string

Required: False

schema

The schema for the model. For application/json models, this should be JSON schema draft 4 model.

Type: string

Required: False

contentType

The content-type for the model, for example, "application/json".

Type: string

Required: False

NotFoundException

The resource specified in the request was not found. See the message field for more information.

message

Describes the error encountered.

Type: string

Required: False

resourceType

The resource type.

Type: string

Required: False

UpdateModelInput

Represents the input parameters for an `UpdateModel` request. Supported only for WebSocket APIs.

name

The name of the model.

Type: string

Required: False

description

The description of the model.

Type: string

Required: False

schema

The schema for the model. For `application/json` models, this should be JSON schema draft 4 model.

Type: string

Required: False

contentType

The content-type for the model, for example, `"application/json"`.

Type: string

Required: False

Models

Represents the collection of data models for an API. Supported only for WebSocket APIs. See [Create Models and Mapping Templates for Request and Response Mappings](#).

URI

/v2/apis/*apiId*/models

HTTP methods

GET

Operation ID: GetModels

Gets the Models for an API.

Path parameters

Name	Type	Required	Description
<i>apiId</i>	String	True	The API identifier.

Query parameters

Name	Type	Required	Description
nextToken	String	False	The next page of elements from this collection. Not valid for the last element of the collection.
maxResults	String	False	The maximum number of elements

Name	Type	Required	Description
			to be returned for this resource.

Responses

Status code	Response model	Description
200	Models	Success
400	BadRequestException	One of the parameters in the request is invalid.
404	NotFoundException	The resource specified in the request was not found.
429	LimitExceededException	The client is sending more than the allowed number of requests per unit of time.

POST

Operation ID: CreateModel

Creates a Model for an API.

Path parameters

Name	Type	Required	Description
<i>apiId</i>	String	True	The API identifier.

Responses

Status code	Response model	Description
201	Model	The request has succeeded and has resulted in the creation of a resource.
400	BadRequestException	One of the parameters in the request is invalid.
404	NotFoundException	The resource specified in the request was not found.
409	ConflictException	The resource already exists.
429	LimitExceededException	The client is sending more than the allowed number of requests per unit of time.

Schemas

Request bodies

POST schema

```
{
  "name": "string",
  "description": "string",
  "schema": "string",
  "contentType": "string"
}
```

Response bodies

Models schema

```
{
  "items": [
```

```
{
  "modelId": "string",
  "name": "string",
  "description": "string",
  "schema": "string",
  "contentType": "string"
},
"nextToken": "string"
}
```

Model schema

```
{
  "modelId": "string",
  "name": "string",
  "description": "string",
  "schema": "string",
  "contentType": "string"
}
```

BadRequestException schema

```
{
  "message": "string"
}
```

NotFoundException schema

```
{
  "message": "string",
  "resourceType": "string"
}
```

ConflictException schema

```
{
  "message": "string"
}
```

```
}
```

LimitExceededException schema

```
{  
  "message": "string",  
  "limitType": "string"  
}
```

Properties

BadRequestException

The request is not valid, for example, the input is incomplete or incorrect. See the accompanying error message for details.

message

Describes the error encountered.

Type: string

Required: False

ConflictException

The requested operation would cause a conflict with the current state of a service resource associated with the request. Resolve the conflict before retrying this request. See the accompanying error message for details.

message

Describes the error encountered.

Type: string

Required: False

CreateModelInput

Represents the input parameters for a `CreateModel` request.

name

The name of the model. Must be alphanumeric.

Type: string

Required: True

description

The description of the model.

Type: string

Required: False

schema

The schema for the model. For application/json models, this should be JSON schema draft 4 model.

Type: string

Required: True

contentType

The content-type for the model, for example, "application/json".

Type: string

Required: False

LimitExceededException

A limit has been exceeded. See the accompanying error message for details.

message

Describes the error encountered.

Type: string

Required: False

limitType

The limit type.

Type: string

Required: False

Model

Represents a data model for an API. Supported only for WebSocket APIs. See [Create Models and Mapping Templates for Request and Response Mappings](#).

modelId

The model identifier.

Type: string

Required: False

name

The name of the model. Must be alphanumeric.

Type: string

Required: True

description

The description of the model.

Type: string

Required: False

schema

The schema for the model. For application/json models, this should be JSON schema draft 4 model.

Type: string

Required: False

contentType

The content-type for the model, for example, "application/json".

Type: string

Required: False

Models

Represents a collection of data models. See [Create Models and Mapping Templates for Request and Response Mappings](#).

items

The elements from this collection.

Type: Array of type [Model](#)

Required: False

nextToken

The next page of elements from this collection. Not valid for the last element of the collection.

Type: string

Required: False

NotFoundException

The resource specified in the request was not found. See the message field for more information.

message

Describes the error encountered.

Type: string

Required: False

resourceType

The resource type.

Type: string

Required: False

ModelTemplate

Represents a model template. Supported only for WebSocket APIs.

URI

`/v2/apis/apiId/models/modelId/template`

HTTP methods

GET

Operation ID: GetModelTemplate

Gets a model template.

Path parameters

Name	Type	Required	Description
<i>modelId</i>	String	True	The model ID.
<i>apiId</i>	String	True	The API identifier.

Responses

Status code	Response model	Description
200	Template	Success
404	NotFoundException	The resource specified in the request was not found.

Status code	Response model	Description
429	LimitExceededException	The client is sending more than the allowed number of requests per unit of time.

Schemas

Response bodies

Template schema

```
{  
  "value": "string"  
}
```

NotFoundExpection schema

```
{  
  "message": "string",  
  "resourceType": "string"  
}
```

LimitExceededException schema

```
{  
  "message": "string",  
  "limitType": "string"  
}
```

Properties

LimitExceededException

A limit has been exceeded. See the accompanying error message for details.

message

Describes the error encountered.

Type: string

Required: False

limitType

The limit type.

Type: string

Required: False

NotFoundException

The resource specified in the request was not found. See the message field for more information.

message

Describes the error encountered.

Type: string

Required: False

resourceType

The resource type.

Type: string

Required: False

Template

Represents a template.

value

The template value.

Type: string

Required: False

Route

Represents a route for an API.

URI

`/v2/apis/apiId/routes/routeId`

HTTP methods

GET

Operation ID: GetRoute

Gets a Route.

Path parameters

Name	Type	Required	Description
<i>apiId</i>	String	True	The API identifier.
<i>routeId</i>	String	True	The route ID.

Responses

Status code	Response model	Description
200	Route	Success
404	NotFoundException	The resource specified in the request was not found.
429	LimitExceededException	The client is sending more than the allowed number of requests per unit of time.

DELETE

Operation ID: DeleteRoute

Deletes a Route.

Path parameters

Name	Type	Required	Description
<i>apiId</i>	String	True	The API identifier.
<i>routeId</i>	String	True	The route ID.

Responses

Status code	Response model	Description
204	None	The request has succeeded , and there is no additional content to send in the response payload body.
404	NotFoundException	The resource specified in the request was not found.
429	LimitExceededException	The client is sending more than the allowed number of requests per unit of time.

PATCH

Operation ID: UpdateRoute

Updates a Route.

Path parameters

Name	Type	Required	Description
<i>apiId</i>	String	True	The API identifier.
<i>routeId</i>	String	True	The route ID.

Responses

Status code	Response model	Description
200	Route	Success
400	BadRequestException	One of the parameters in the request is invalid.
404	NotFoundException	The resource specified in the request was not found.
409	ConflictException	The resource already exists.
429	LimitExceededException	The client is sending more than the allowed number of requests per unit of time.

Schemas

Request bodies

PATCH schema

```
{
  "routeKey": "string",
  "authorizationType": enum,
  "authorizerId": "string",
  "authorizationScopes": [
    "string"
  ],
  "apiKeyRequired": boolean,
  "requestParameters": {
  },
  "requestModels": {
  },
  "modelSelectionExpression": "string",
  "target": "string",
  "operationName": "string",
  "routeResponseSelectionExpression": "string"
}
```

Response bodies

Route schema

```
{
  "routeId": "string",
  "routeKey": "string",
  "authorizationType": enum,
  "authorizerId": "string",
  "authorizationScopes": [
    "string"
  ],
  "apiKeyRequired": boolean,
  "requestParameters": {
  },
  "requestModels": {
  },
  "modelSelectionExpression": "string",
  "target": "string",
  "operationName": "string",
  "routeResponseSelectionExpression": "string",
  "apiGatewayManaged": boolean
}
```

BadRequestException schema

```
{
  "message": "string"
}
```

NotFoundException schema

```
{
  "message": "string",
  "resourceType": "string"
}
```

ConflictException schema

```
{
```



```
"message": "string"
}
```

LimitExceededException schema

```
{
  "message": "string",
  "limitType": "string"
}
```

Properties

AuthorizationType

The authorization type. For WebSocket APIs, valid values are NONE for open access, AWS_IAM for using Amazon IAM permissions, and CUSTOM for using a Lambda authorizer. For HTTP APIs, valid values are NONE for open access, JWT for using JSON Web Tokens, AWS_IAM for using Amazon IAM permissions, and CUSTOM for using a Lambda authorizer.

NONE

AWS_IAM

CUSTOM

JWT

BadRequestException

The request is not valid, for example, the input is incomplete or incorrect. See the accompanying error message for details.

message

Describes the error encountered.

Type: string

Required: False

ConflictException

The requested operation would cause a conflict with the current state of a service resource associated with the request. Resolve the conflict before retrying this request. See the accompanying error message for details.

message

Describes the error encountered.

Type: string

Required: False

LimitExceededException

A limit has been exceeded. See the accompanying error message for details.

message

Describes the error encountered.

Type: string

Required: False

limitType

The limit type.

Type: string

Required: False

NotFoundException

The resource specified in the request was not found. See the message field for more information.

message

Describes the error encountered.

Type: string

Required: False

resourceType

The resource type.

Type: string

Required: False

ParameterConstraints

Validation constraints imposed on parameters of a request (path, query string, headers).

required

Whether or not the parameter is required.

Type: boolean

Required: False

Route

Represents a route.

routeId

The route ID.

Type: string

Required: False

routeKey

The route key for the route. For HTTP APIs, the route key can be either `$default`, or a combination of an HTTP method and resource path, for example, `GET /pets`.

Type: string

Required: True

authorizationType

The authorization type for the route. For WebSocket APIs, valid values are NONE for open access, AWS_IAM for using Amazon IAM permissions, and CUSTOM for using a Lambda authorizer. For HTTP APIs, valid values are NONE for open access, JWT for using JSON Web Tokens, AWS_IAM for using Amazon IAM permissions, and CUSTOM for using a Lambda authorizer.

Type: [AuthorizationType](#)

Required: False

authorizerId

The identifier of the Authorizer resource to be associated with this route. The authorizer identifier is generated by API Gateway when you created the authorizer.

Type: string

Required: False

authorizationScopes

A list of authorization scopes configured on a route. The scopes are used with a JWT authorizer to authorize the method invocation. The authorization works by matching the route scopes against the scopes parsed from the access token in the incoming request. The method invocation is authorized if any route scope matches a claimed scope in the access token. Otherwise, the invocation is not authorized. When the route scope is configured, the client must provide an access token instead of an identity token for authorization purposes.

Type: Array of type string

Required: False

apiKeyRequired

Specifies whether an API key is required for this route. Supported only for WebSocket APIs.

Type: boolean

Required: False

requestParameters

The request parameters for the route. Supported only for WebSocket APIs.

Type: [RouteParameters](#)

Required: False

requestModels

The request models for the route. Supported only for WebSocket APIs.

Type: [RouteModels](#)

Required: False

modelSelectionExpression

The model selection expression for the route. Supported only for WebSocket APIs.

Type: string

Required: False

target

The target for the route.

Type: string

Required: False

operationName

The operation name for the route.

Type: string

Required: False

routeResponseSelectionExpression

The route response selection expression for the route. Supported only for WebSocket APIs.

Type: string

Required: False

apiGatewayManaged

Specifies whether a route is managed by API Gateway. If you created an API using quick create, the `$default` route is managed by API Gateway. You can't modify the `$default` route key.

Type: boolean

Required: False

RouteModels

The route models.

key-value pairs

Type: string

RouteParameters

The route parameters.

key-value pairs

Type: object

UpdateRouteInput

Represents the input parameters for an `UpdateRoute` request.

routeKey

The route key for the route. For HTTP APIs, the route key can be either `$default`, or a combination of an HTTP method and resource path, for example, `GET /pets`.

Type: string

Required: False

authorizationType

The authorization type for the route. For WebSocket APIs, valid values are NONE for open access, AWS_IAM for using Amazon IAM permissions, and CUSTOM for using a Lambda authorizer. For HTTP APIs, valid values are NONE for open access, JWT for using JSON Web Tokens, AWS_IAM for using Amazon IAM permissions, and CUSTOM for using a Lambda authorizer.

Type: [AuthorizationType](#)

Required: False

authorizerId

The identifier of the Authorizer resource to be associated with this route. The authorizer identifier is generated by API Gateway when you created the authorizer.

Type: string

Required: False

authorizationScopes

The authorization scopes supported by this route.

Type: Array of type string

Required: False

apiKeyRequired

Specifies whether an API key is required for the route. Supported only for WebSocket APIs.

Type: boolean

Required: False

requestParameters

The request parameters for the route. Supported only for WebSocket APIs.

Type: [RouteParameters](#)

Required: False

requestModels

The request models for the route. Supported only for WebSocket APIs.

Type: [RouteModels](#)

Required: False

modelSelectionExpression

The model selection expression for the route. Supported only for WebSocket APIs.

Type: string

Required: False

target

The target for the route.

Type: string

Required: False

operationName

The operation name for the route.

Type: string

Required: False

routeResponseSelectionExpression

The route response selection expression for the route. Supported only for WebSocket APIs.

Type: string

Required: False

RouteRequestParameter

Represents a route request parameter.

URI

/v2/apis/*apiId*/routes/*routeId*/requestparameters/*requestParameterKey*

HTTP methods

DELETE

Operation ID: DeleteRouteRequestParameter

Deletes a route request parameter. Supported only for WebSocket APIs.

Path parameters

Name	Type	Required	Description
<i>requestParameterKey</i>	String	True	The route request parameter key.
<i>apiId</i>	String	True	The API identifier.
<i>routeId</i>	String	True	The route ID.

Responses

Status code	Response model	Description
204	None	The request has succeeded , and there is no additional content to send in the response payload body.
404	NotFoundException	The resource specified in the request was not found.
429	LimitExceededException	The client is sending more than the allowed number of requests per unit of time.

Schemas

Response bodies

NotFoundException schema

```
{
  "message": "string",
  "resourceType": "string"
}
```

LimitExceededException schema

```
{
  "message": "string",
  "limitType": "string"
}
```

Properties

LimitExceededException

A limit has been exceeded. See the accompanying error message for details.

message

Describes the error encountered.

Type: string

Required: False

limitType

The limit type.

Type: string

Required: False

NotFoundException

The resource specified in the request was not found. See the message field for more information.

message

Describes the error encountered.

Type: string

Required: False

resourceType

The resource type.

Type: string

Required: False

RouteResponse

Represents a route response. Supported only for WebSocket APIs.

URI

`/v2/apis/apiId/routes/routeId/routeresponses/routeResponseId`

HTTP methods

GET

Operation ID: GetRouteResponse

Gets a RouteResponse.

Path parameters

Name	Type	Required	Description
<i>routeResponseId</i>	String	True	The route response ID.

Name	Type	Required	Description
<i>apiId</i>	String	True	The API identifier.
<i>routeId</i>	String	True	The route ID.

Responses

Status code	Response model	Description
200	RouteResponse	Success
404	NotFoundException	The resource specified in the request was not found.
429	LimitExceededException	The client is sending more than the allowed number of requests per unit of time.

DELETE

Operation ID: DeleteRouteResponse

Deletes a RouteResponse.

Path parameters

Name	Type	Required	Description
<i>routeResponseId</i>	String	True	The route response ID.
<i>apiId</i>	String	True	The API identifier.
<i>routeId</i>	String	True	The route ID.

Responses

Status code	Response model	Description
204	None	The request has succeeded , and there is no additional content to send in the response payload body.
404	NotFoundException	The resource specified in the request was not found.
429	LimitExceededException	The client is sending more than the allowed number of requests per unit of time.

PATCH

Operation ID: UpdateRouteResponse

Updates a RouteResponse.

Path parameters

Name	Type	Required	Description
<i>routeResponseId</i>	String	True	The route response ID.
<i>apiId</i>	String	True	The API identifier.
<i>routeId</i>	String	True	The route ID.

Responses

Status code	Response model	Description
200	RouteResponse	Success

Status code	Response model	Description
400	BadRequestException	One of the parameters in the request is invalid.
404	NotFoundException	The resource specified in the request was not found.
409	ConflictException	The resource already exists.
429	LimitExceededException	The client is sending more than the allowed number of requests per unit of time.

Schemas

Request bodies

PATCH schema

```
{
  "routeResponseKey": "string",
  "responseParameters": {
  },
  "responseModels": {
  },
  "modelSelectionExpression": "string"
}
```

Response bodies

RouteResponse schema

```
{
  "routeResponseId": "string",
  "routeResponseKey": "string",
  "responseParameters": {
  },
  "responseModels": {
  },
}
```

```
"modelSelectionExpression": "string"
}
```

BadRequestException schema

```
{
  "message": "string"
}
```

NotFoundException schema

```
{
  "message": "string",
  "resourceType": "string"
}
```

ConflictException schema

```
{
  "message": "string"
}
```

LimitExceededException schema

```
{
  "message": "string",
  "limitType": "string"
}
```

Properties

BadRequestException

The request is not valid, for example, the input is incomplete or incorrect. See the accompanying error message for details.

message

Describes the error encountered.

Type: string

Required: False

ConflictException

The requested operation would cause a conflict with the current state of a service resource associated with the request. Resolve the conflict before retrying this request. See the accompanying error message for details.

message

Describes the error encountered.

Type: string

Required: False

LimitExceededException

A limit has been exceeded. See the accompanying error message for details.

message

Describes the error encountered.

Type: string

Required: False

limitType

The limit type.

Type: string

Required: False

NotFoundException

The resource specified in the request was not found. See the message field for more information.

message

Describes the error encountered.

Type: string

Required: False

resourceType

The resource type.

Type: string

Required: False

ParameterConstraints

Validation constraints imposed on parameters of a request (path, query string, headers).

required

Whether or not the parameter is required.

Type: boolean

Required: False

RouteModels

The route models.

key-value pairs

Type: string

RouteParameters

The route parameters.

key-value pairs

Type: object

RouteResponse

Represents a route response.

routeResponseId

Represents the identifier of a route response.

Type: string

Required: False

routeResponseKey

Represents the route response key of a route response.

Type: string

Required: True

responseParameters

Represents the response parameters of a route response.

Type: [RouteParameters](#)

Required: False

responseModels

Represents the response models of a route response.

Type: [RouteModels](#)

Required: False

modelSelectionExpression

Represents the model selection expression of a route response. Supported only for WebSocket APIs.

Type: string

Required: False

UpdateRouteResponseInput

Represents the input parameters for an UpdateRouteResponse request.

routeResponseKey

The route response key.

Type: string

Required: False

responseParameters

The route response parameters.

Type: [RouteParameters](#)

Required: False

responseModels

The response models for the route response.

Type: [RouteModels](#)

Required: False

modelSelectionExpression

The model selection expression for the route response. Supported only for WebSocket APIs.

Type: string

Required: False

RouteResponses

Represents the collection of responses for a route. Supported only for WebSocket APIs.

URI

`/v2/apis/apiId/routes/routeId/routeresponses`

HTTP methods

GET

Operation ID: GetRouteResponses

Gets the RouteResponses for a Route.

Path parameters

Name	Type	Required	Description
<i>routeId</i>	String	True	The route ID.
<i>apiId</i>	String	True	The API identifier.

Query parameters

Name	Type	Required	Description
nextToken	String	False	The next page of elements from this collection. Not valid for the last element of the collection.
maxResults	String	False	The maximum number of elements to be returned for this resource.

Responses

Status code	Response model	Description
200	RouteResponses	Success
400	BadRequestException	One of the parameters in the request is invalid.

Status code	Response model	Description
404	NotFoundException	The resource specified in the request was not found.
429	LimitExceededException	The client is sending more than the allowed number of requests per unit of time.

POST

Operation ID: CreateRouteResponse

Creates a RouteResponse for a Route.

Path parameters

Name	Type	Required	Description
<i>apiId</i>	String	True	The API identifier.
<i>routeId</i>	String	True	The route ID.

Responses

Status code	Response model	Description
201	RouteResponse	The request has succeeded and has resulted in the creation of a resource.
400	BadRequestException	One of the parameters in the request is invalid.
404	NotFoundException	The resource specified in the request was not found.
409	ConflictException	The resource already exists.

Status code	Response model	Description
429	LimitExceededException	The client is sending more than the allowed number of requests per unit of time.

Schemas

Request bodies

POST schema

```
{
  "routeResponseKey": "string",
  "responseParameters": {
  },
  "responseModels": {
  },
  "modelSelectionExpression": "string"
}
```

Response bodies

RouteResponses schema

```
{
  "items": [
    {
      "routeResponseId": "string",
      "routeResponseKey": "string",
      "responseParameters": {
      },
      "responseModels": {
      },
      "modelSelectionExpression": "string"
    }
  ],
  "nextToken": "string"
}
```

RouteResponse schema

```
{
  "routeResponseId": "string",
  "routeResponseKey": "string",
  "responseParameters": {
  },
  "responseModels": {
  },
  "modelSelectionExpression": "string"
}
```

BadRequestException schema

```
{
  "message": "string"
}
```

NotFoundException schema

```
{
  "message": "string",
  "resourceType": "string"
}
```

ConflictException schema

```
{
  "message": "string"
}
```

LimitExceededException schema

```
{
  "message": "string",
  "limitType": "string"
}
```

Properties

BadRequestException

The request is not valid, for example, the input is incomplete or incorrect. See the accompanying error message for details.

message

Describes the error encountered.

Type: string

Required: False

ConflictException

The requested operation would cause a conflict with the current state of a service resource associated with the request. Resolve the conflict before retrying this request. See the accompanying error message for details.

message

Describes the error encountered.

Type: string

Required: False

CreateRouteResponseInput

Represents the input parameters for an CreateRouteResponse request.

routeResponseKey

The route response key.

Type: string

Required: True

responseParameters

The route response parameters.

Type: [RouteParameters](#)

Required: False

responseModels

The response models for the route response.

Type: [RouteModels](#)

Required: False

modelSelectionExpression

The model selection expression for the route response. Supported only for WebSocket APIs.

Type: string

Required: False

LimitExceededException

A limit has been exceeded. See the accompanying error message for details.

message

Describes the error encountered.

Type: string

Required: False

limitType

The limit type.

Type: string

Required: False

NotFoundException

The resource specified in the request was not found. See the message field for more information.

message

Describes the error encountered.

Type: string

Required: False

resourceType

The resource type.

Type: string

Required: False

ParameterConstraints

Validation constraints imposed on parameters of a request (path, query string, headers).

required

Whether or not the parameter is required.

Type: boolean

Required: False

RouteModels

The route models.

key-value pairs

Type: string

RouteParameters

The route parameters.

key-value pairs

Type: object

RouteResponse

Represents a route response.

routeResponseId

Represents the identifier of a route response.

Type: string

Required: False

routeResponseKey

Represents the route response key of a route response.

Type: string

Required: True

responseParameters

Represents the response parameters of a route response.

Type: [RouteParameters](#)

Required: False

responseModels

Represents the response models of a route response.

Type: [RouteModels](#)

Required: False

modelSelectionExpression

Represents the model selection expression of a route response. Supported only for WebSocket APIs.

Type: string

Required: False

RouteResponses

Represents a collection of route responses.

items

The elements from this collection.

Type: Array of type [RouteResponse](#)

Required: False

nextToken

The next page of elements from this collection. Not valid for the last element of the collection.

Type: string

Required: False

Routes

Represents the collection of routes for an API.

URI

`/v2/apis/apiId/routes`

HTTP methods

GET

Operation ID: GetRoutes

Gets the Routes for an API.

Path parameters

Name	Type	Required	Description
<i>apiId</i>	String	True	The API identifier.

Query parameters

Name	Type	Required	Description
nextToken	String	False	The next page of elements from this collection. Not valid for the last element of the collection.
maxResults	String	False	The maximum number of elements to be returned for this resource.

Responses

Status code	Response model	Description
200	Routes	Success
400	BadRequestException	One of the parameters in the request is invalid.
404	NotFoundException	The resource specified in the request was not found.
429	LimitExceededException	The client is sending more than the allowed number of requests per unit of time.

POST

Operation ID: CreateRoute

Creates a Route for an API.

Path parameters

Name	Type	Required	Description
<i>apiId</i>	String	True	The API identifier.

Responses

Status code	Response model	Description
201	Route	The request has succeeded and has resulted in the creation of a resource.
400	BadRequestException	One of the parameters in the request is invalid.
404	NotFoundException	The resource specified in the request was not found.
409	ConflictException	The resource already exists.
429	LimitExceededException	The client is sending more than the allowed number of requests per unit of time.

Schemas

Request bodies

POST schema

```
{
  "routeKey": "string",
  "authorizationType": enum,
  "authorizerId": "string",
  "authorizationScopes": [
    "string"
  ],
  "apiKeyRequired": boolean,
```

```

  "requestParameters": {
  },
  "requestModels": {
  },
  "modelSelectionExpression": "string",
  "target": "string",
  "operationName": "string",
  "routeResponseSelectionExpression": "string"
}

```

Response bodies

Routes schema

```

{
  "items": [
    {
      "routeId": "string",
      "routeKey": "string",
      "authorizationType": enum,
      "authorizerId": "string",
      "authorizationScopes": [
        "string"
      ],
      "apiKeyRequired": boolean,
      "requestParameters": {
      },
      "requestModels": {
      },
      "modelSelectionExpression": "string",
      "target": "string",
      "operationName": "string",
      "routeResponseSelectionExpression": "string",
      "apiGatewayManaged": boolean
    }
  ],
  "nextToken": "string"
}

```

Route schema

```

{

```

```
"routeId": "string",
"routeKey": "string",
"authorizationType": enum,
"authorizerId": "string",
"authorizationScopes": [
  "string"
],
"apiKeyRequired": boolean,
"requestParameters": {
},
"requestModels": {
},
"modelSelectionExpression": "string",
"target": "string",
"operationName": "string",
"routeResponseSelectionExpression": "string",
"apiGatewayManaged": boolean
}
```

BadRequestException schema

```
{
  "message": "string"
}
```

NotFoundException schema

```
{
  "message": "string",
  "resourceType": "string"
}
```

ConflictException schema

```
{
  "message": "string"
}
```


LimitExceededException schema

```
{
  "message": "string",
  "limitType": "string"
}
```

Properties

AuthorizationType

The authorization type. For WebSocket APIs, valid values are NONE for open access, AWS_IAM for using Amazon IAM permissions, and CUSTOM for using a Lambda authorizer. For HTTP APIs, valid values are NONE for open access, JWT for using JSON Web Tokens, AWS_IAM for using Amazon IAM permissions, and CUSTOM for using a Lambda authorizer.

NONE

AWS_IAM

CUSTOM

JWT

BadRequestException

The request is not valid, for example, the input is incomplete or incorrect. See the accompanying error message for details.

message

Describes the error encountered.

Type: string

Required: False

ConflictException

The requested operation would cause a conflict with the current state of a service resource associated with the request. Resolve the conflict before retrying this request. See the accompanying error message for details.

message

Describes the error encountered.

Type: string

Required: False

CreateRouteInput

Represents the input parameters for a `CreateRoute` request.

routeKey

The route key for the route. For HTTP APIs, the route key can be either `$default`, or a combination of an HTTP method and resource path, for example, `GET /pets`.

Type: string

Required: True

authorizationType

The authorization type for the route. For WebSocket APIs, valid values are `NONE` for open access, `AWS_IAM` for using Amazon IAM permissions, and `CUSTOM` for using a Lambda authorizer. For HTTP APIs, valid values are `NONE` for open access, `JWT` for using JSON Web Tokens, `AWS_IAM` for using Amazon IAM permissions, and `CUSTOM` for using a Lambda authorizer.

Type: [AuthorizationType](#)

Required: False

authorizerId

The identifier of the `Authorizer` resource to be associated with this route. The authorizer identifier is generated by API Gateway when you created the authorizer.

Type: string

Required: False

authorizationScopes

The authorization scopes supported by this route.

Type: Array of type string

Required: False

apiKeyRequired

Specifies whether an API key is required for the route. Supported only for WebSocket APIs.

Type: boolean

Required: False

requestParameters

The request parameters for the route. Supported only for WebSocket APIs.

Type: [RouteParameters](#)

Required: False

requestModels

The request models for the route. Supported only for WebSocket APIs.

Type: [RouteModels](#)

Required: False

modelSelectionExpression

The model selection expression for the route. Supported only for WebSocket APIs.

Type: string

Required: False

target

The target for the route.

Type: string

Required: False

operationName

The operation name for the route.

Type: string

Required: False

routeResponseSelectionExpression

The route response selection expression for the route. Supported only for WebSocket APIs.

Type: string

Required: False

LimitExceededException

A limit has been exceeded. See the accompanying error message for details.

message

Describes the error encountered.

Type: string

Required: False

limitType

The limit type.

Type: string

Required: False

NotFoundException

The resource specified in the request was not found. See the message field for more information.

message

Describes the error encountered.

Type: string

Required: False

resourceType

The resource type.

Type: string

Required: False

ParameterConstraints

Validation constraints imposed on parameters of a request (path, query string, headers).

required

Whether or not the parameter is required.

Type: boolean

Required: False

Route

Represents a route.

routeId

The route ID.

Type: string

Required: False

routeKey

The route key for the route. For HTTP APIs, the route key can be either `$default`, or a combination of an HTTP method and resource path, for example, `GET /pets`.

Type: string

Required: True

authorizationType

The authorization type for the route. For WebSocket APIs, valid values are NONE for open access, AWS_IAM for using Amazon IAM permissions, and CUSTOM for using a Lambda authorizer. For HTTP APIs, valid values are NONE for open access, JWT for using JSON Web Tokens, AWS_IAM for using Amazon IAM permissions, and CUSTOM for using a Lambda authorizer.

Type: [AuthorizationType](#)

Required: False

authorizerId

The identifier of the `Authorizer` resource to be associated with this route. The authorizer identifier is generated by API Gateway when you created the authorizer.

Type: string

Required: False

authorizationScopes

A list of authorization scopes configured on a route. The scopes are used with a JWT authorizer to authorize the method invocation. The authorization works by matching the route scopes against the scopes parsed from the access token in the incoming request. The method invocation is authorized if any route scope matches a claimed scope in the access token. Otherwise, the invocation is not authorized. When the route scope is configured, the client must provide an access token instead of an identity token for authorization purposes.

Type: Array of type string

Required: False

apiKeyRequired

Specifies whether an API key is required for this route. Supported only for WebSocket APIs.

Type: boolean

Required: False

requestParameters

The request parameters for the route. Supported only for WebSocket APIs.

Type: [RouteParameters](#)

Required: False

requestModels

The request models for the route. Supported only for WebSocket APIs.

Type: [RouteModels](#)

Required: False

modelSelectionExpression

The model selection expression for the route. Supported only for WebSocket APIs.

Type: string

Required: False

target

The target for the route.

Type: string

Required: False

operationName

The operation name for the route.

Type: string

Required: False

routeResponseSelectionExpression

The route response selection expression for the route. Supported only for WebSocket APIs.

Type: string

Required: False

apiGatewayManaged

Specifies whether a route is managed by API Gateway. If you created an API using quick create, the `$default` route is managed by API Gateway. You can't modify the `$default` route key.

Type: boolean

Required: False

RouteModels

The route models.

key-value pairs

Type: string

RouteParameters

The route parameters.

key-value pairs

Type: object

Routes

Represents a collection of routes.

items

The elements from this collection.

Type: Array of type [Route](#)

Required: False

nextToken

The next page of elements from this collection. Not valid for the last element of the collection.

Type: string

Required: False

RouteSettings

Represents the route settings for a stage.

URI

`/v2/apis/apiId/stages/stageName/routesettings/routeKey`

HTTP methods

DELETE

Operation ID: DeleteRouteSettings

Deletes the RouteSettings for a stage.

Path parameters

Name	Type	Required	Description
<i>stageName</i>	String	True	The stage name. Stage names can contain only alphanumeric characters, hyphens, and underscores, or be \$default. Maximum length is 128 characters.

Name	Type	Required	Description
<i>routeKey</i>	String	True	The route key.
<i>apiId</i>	String	True	The API identifier.

Responses

Status code	Response model	Description
204	None	The request has succeeded , and there is no additional content to send in the response payload body.
404	NotFoundException	The resource specified in the request was not found.
429	LimitExceededException	The client is sending more than the allowed number of requests per unit of time.

Schemas

Response bodies

NotFoundException schema

```
{
  "message": "string",
  "resourceType": "string"
}
```

LimitExceededException schema

```
{
  "message": "string",
```

```
"limitType": "string"  
}
```

Properties

LimitExceededException

A limit has been exceeded. See the accompanying error message for details.

message

Describes the error encountered.

Type: string

Required: False

limitType

The limit type.

Type: string

Required: False

NotFoundException

The resource specified in the request was not found. See the message field for more information.

message

Describes the error encountered.

Type: string

Required: False

resourceType

The resource type.

Type: string

Required: False

Stage

Represents an API stage.

URI

`/v2/apis/apiId/stages/stageName`

HTTP methods

GET

Operation ID: GetStage

Gets a Stage.

Path parameters

Name	Type	Required	Description
<i>stageName</i>	String	True	The stage name. Stage names can contain only alphanumeric characters, hyphens, and underscores, or be \$default. Maximum length is 128 characters.
<i>apiId</i>	String	True	The API identifier.

Responses

Status code	Response model	Description
200	Stage	Success

Status code	Response model	Description
404	NotFoundException	The resource specified in the request was not found.
429	LimitExceededException	The client is sending more than the allowed number of requests per unit of time.

DELETE

Operation ID: DeleteStage

Deletes a Stage.

Path parameters

Name	Type	Required	Description
<i>stageName</i>	String	True	The stage name. Stage names can contain only alphanumeric characters, hyphens, and underscores, or be \$default. Maximum length is 128 characters.
<i>apiId</i>	String	True	The API identifier.

Responses

Status code	Response model	Description
204	None	The request has succeeded , and there is no additiona

Status code	Response model	Description
404	NotFoundException	The resource specified in the request was not found.
429	LimitExceededException	The client is sending more than the allowed number of requests per unit of time.

PATCH

Operation ID: UpdateStage

Updates a Stage.

Path parameters

Name	Type	Required	Description
<i>stageName</i>	String	True	The stage name. Stage names can contain only alphanumeric characters, hyphens, and underscores, or be <code>\$default</code> . Maximum length is 128 characters.
<i>apiId</i>	String	True	The API identifier.

Responses

Status code	Response model	Description
200	Stage	Success

Status code	Response model	Description
400	BadRequestException	One of the parameters in the request is invalid.
404	NotFoundException	The resource specified in the request was not found.
409	ConflictException	The resource already exists.
429	LimitExceededException	The client is sending more than the allowed number of requests per unit of time.

Schemas

Request bodies

PATCH schema

```
{
  "description": "string",
  "deploymentId": "string",
  "clientCertificateId": "string",
  "defaultRouteSettings": {
    "detailedMetricsEnabled": boolean,
    "loggingLevel": enum,
    "dataTraceEnabled": boolean,
    "throttlingBurstLimit": integer,
    "throttlingRateLimit": number
  },
  "routeSettings": {
  },
  "stageVariables": {
  },
  "accessLogSettings": {
    "format": "string",
    "destinationArn": "string"
  },
  "autoDeploy": boolean
}
```

```
}
```

Response bodies

Stage schema

```
{
  "stageName": "string",
  "description": "string",
  "deploymentId": "string",
  "clientCertificateId": "string",
  "defaultRouteSettings": {
    "detailedMetricsEnabled": boolean,
    "loggingLevel": enum,
    "dataTraceEnabled": boolean,
    "throttlingBurstLimit": integer,
    "throttlingRateLimit": number
  },
  "routeSettings": {
  },
  "stageVariables": {
  },
  "accessLogSettings": {
    "format": "string",
    "destinationArn": "string"
  },
  "autoDeploy": boolean,
  "lastDeploymentStatusMessage": "string",
  "createdDate": "string",
  "lastUpdatedDate": "string",
  "tags": {
  },
  "apiGatewayManaged": boolean
}
```

BadRequestException schema

```
{
  "message": "string"
}
```


NotFoundException schema

```
{
  "message": "string",
  "resourceType": "string"
}
```

ConflictException schema

```
{
  "message": "string"
}
```

LimitExceededException schema

```
{
  "message": "string",
  "limitType": "string"
}
```

Properties

AccessLogSettings

Settings for logging access in a stage.

format

A single line format of the access logs of data, as specified by selected `$context` variables. The format must include at least `$context.requestId`.

Type: string

Required: False

destinationArn

The ARN of the CloudWatch Logs log group to receive access logs.

Type: string

Required: False

BadRequestException

The request is not valid, for example, the input is incomplete or incorrect. See the accompanying error message for details.

message

Describes the error encountered.

Type: string

Required: False

ConflictException

The requested operation would cause a conflict with the current state of a service resource associated with the request. Resolve the conflict before retrying this request. See the accompanying error message for details.

message

Describes the error encountered.

Type: string

Required: False

LimitExceededException

A limit has been exceeded. See the accompanying error message for details.

message

Describes the error encountered.

Type: string

Required: False

limitType

The limit type.

Type: string

Required: False

LoggingLevel

The logging level.

ERROR

INFO

false

NotFoundException

The resource specified in the request was not found. See the message field for more information.

message

Describes the error encountered.

Type: string

Required: False

resourceType

The resource type.

Type: string

Required: False

RouteSettings

Represents a collection of route settings.

detailedMetricsEnabled

Specifies whether detailed metrics are enabled.

Type: boolean

Required: False

loggingLevel

Specifies the logging level for this route: INFO, ERROR, or OFF. This property affects the log entries pushed to Amazon CloudWatch Logs. Supported only for WebSocket APIs.

Type: [LoggingLevel](#)

Required: False

dataTraceEnabled

Specifies whether (`true`) or not (`false`) data trace logging is enabled for this route. This property affects the log entries pushed to Amazon CloudWatch Logs. Supported only for WebSocket APIs.

Type: boolean

Required: False

throttlingBurstLimit

Specifies the throttling burst limit.

Type: integer

Required: False

Format: int32

throttlingRateLimit

Specifies the throttling rate limit.

Type: number

Required: False

Format: double

RouteSettingsMap

The route settings map.

key-value pairs

Type: object

Stage

Represents an API stage.

stageName

The name of the stage.

Type: string

Required: True

description

The description of the stage.

Type: string

Required: False

deploymentId

The identifier of the Deployment that the Stage is associated with. Can't be updated if autoDeploy is enabled.

Type: string

Required: False

clientCertificateId

The identifier of a client certificate for a Stage. Supported only for WebSocket APIs.

Type: string

Required: False

defaultRouteSettings

Default route settings for the stage.

Type: [RouteSettings](#)

Required: False

routeSettings

Route settings for the stage, by routeKey.

Type: [RouteSettingsMap](#)

Required: False

stageVariables

A map that defines the stage variables for a stage resource. Variable names can have alphanumeric and underscore characters, and the values must match `[A-Za-z0-9-._~:/?#&=,]+`.

Type: [StageVariablesMap](#)

Required: False

accessLogSettings

Settings for logging access in this stage.

Type: [AccessLogSettings](#)

Required: False

autoDeploy

Specifies whether updates to an API automatically trigger a new deployment. The default value is `false`.

Type: boolean

Required: False

lastDeploymentStatusMessage

Describes the status of the last deployment of a stage. Supported only for stages with `autoDeploy` enabled.

Type: string

Required: False

createdDate

The timestamp when the stage was created.

Type: string

Required: False

Format: date-time

lastUpdatedDate

The timestamp when the stage was last updated.

Type: string

Required: False

Format: date-time

tags

The collection of tags. Each tag element is associated with a given resource.

Type: [Tags](#)

Required: False

apiGatewayManaged

Specifies whether a stage is managed by API Gateway. If you created an API using quick create, the `$default` stage is managed by API Gateway. You can't modify the `$default` stage.

Type: boolean

Required: False

StageVariablesMap

The stage variable map.

key-value pairs

Type: string

Tags

Represents a collection of tags associated with the resource.

key-value pairs

Type: string

UpdateStageInput

Represents the input parameters for an UpdateStage request.

description

The description for the API stage.

Type: string

Required: False

deploymentId

The deployment identifier for the API stage. Can't be updated if autoDeploy is enabled.

Type: string

Required: False

clientCertificateId

The identifier of a client certificate for a Stage.

Type: string

Required: False

defaultRouteSettings

The default route settings for the stage.

Type: [RouteSettings](#)

Required: False

routeSettings

Route settings for the stage.

Type: [RouteSettingsMap](#)

Required: False

stageVariables

A map that defines the stage variables for a Stage. Variable names can have alphanumeric and underscore characters, and the values must match [A-Za-z0-9-._~:/?#&=,]+.

Type: [StageVariablesMap](#)

Required: False

accessLogSettings

Settings for logging access in this stage.

Type: [AccessLogSettings](#)

Required: False

autoDeploy

Specifies whether updates to an API automatically trigger a new deployment. The default value is false.

Type: boolean

Required: False

Stages

Represents the collection of stages for an API.

URI

/v2/apis/*apiId*/stages

HTTP methods

GET

Operation ID: GetStages

Gets the Stages for an API.

Path parameters

Name	Type	Required	Description
<i>apiId</i>	String	True	The API identifier.

Query parameters

Name	Type	Required	Description
nextToken	String	False	The next page of elements from this collection. Not valid for the last element of the collection.
maxResults	String	False	The maximum number of elements to be returned for this resource.

Responses

Status code	Response model	Description
200	Stages	Success
400	BadRequestException	One of the parameters in the request is invalid.

Status code	Response model	Description
404	NotFoundException	The resource specified in the request was not found.
429	LimitExceededException	The client is sending more than the allowed number of requests per unit of time.

POST

Operation ID: CreateStage

Creates a Stage for an API.

Path parameters

Name	Type	Required	Description
<i>apiId</i>	String	True	The API identifier.

Responses

Status code	Response model	Description
201	Stage	The request has succeeded and has resulted in the creation of a resource.
400	BadRequestException	One of the parameters in the request is invalid.
404	NotFoundException	The resource specified in the request was not found.
409	ConflictException	The resource already exists.

Status code	Response model	Description
429	LimitExceededException	The client is sending more than the allowed number of requests per unit of time.

Schemas

Request bodies

POST schema

```
{
  "stageName": "string",
  "description": "string",
  "deploymentId": "string",
  "clientCertificateId": "string",
  "defaultRouteSettings": {
    "detailedMetricsEnabled": boolean,
    "loggingLevel": enum,
    "dataTraceEnabled": boolean,
    "throttlingBurstLimit": integer,
    "throttlingRateLimit": number
  },
  "routeSettings": {
  },
  "stageVariables": {
  },
  "accessLogSettings": {
    "format": "string",
    "destinationArn": "string"
  },
  "tags": {
  },
  "autoDeploy": boolean
}
```

Response bodies

Stages schema

```
{
  "items": [
    {
      "stageName": "string",
      "description": "string",
      "deploymentId": "string",
      "clientCertificateId": "string",
      "defaultRouteSettings": {
        "detailedMetricsEnabled": boolean,
        "loggingLevel": enum,
        "dataTraceEnabled": boolean,
        "throttlingBurstLimit": integer,
        "throttlingRateLimit": number
      },
      "routeSettings": {
      },
      "stageVariables": {
      },
      "accessLogSettings": {
        "format": "string",
        "destinationArn": "string"
      },
      "autoDeploy": boolean,
      "lastDeploymentStatusMessage": "string",
      "createdDate": "string",
      "lastUpdatedDate": "string",
      "tags": {
      },
      "apiGatewayManaged": boolean
    }
  ],
  "nextToken": "string"
}
```

Stage schema

```
{
  "stageName": "string",
  "description": "string",
  "deploymentId": "string",
  "clientCertificateId": "string",
```

```
"defaultRouteSettings": {
  "detailedMetricsEnabled": boolean,
  "loggingLevel": enum,
  "dataTraceEnabled": boolean,
  "throttlingBurstLimit": integer,
  "throttlingRateLimit": number
},
"routeSettings": {
},
"stageVariables": {
},
"accessLogSettings": {
  "format": "string",
  "destinationArn": "string"
},
"autoDeploy": boolean,
"lastDeploymentStatusMessage": "string",
"createdDate": "string",
"lastUpdatedDate": "string",
"tags": {
},
"apiGatewayManaged": boolean
}
```

BadRequestException schema

```
{
  "message": "string"
}
```

NotFoundException schema

```
{
  "message": "string",
  "resourceType": "string"
}
```

ConflictException schema

```
{
```

```
"message": "string"
}
```

LimitExceededException schema

```
{
  "message": "string",
  "limitType": "string"
}
```

Properties

AccessLogSettings

Settings for logging access in a stage.

format

A single line format of the access logs of data, as specified by selected `$context` variables. The format must include at least `$context.requestId`.

Type: string

Required: False

destinationArn

The ARN of the CloudWatch Logs log group to receive access logs.

Type: string

Required: False

BadRequestException

The request is not valid, for example, the input is incomplete or incorrect. See the accompanying error message for details.

message

Describes the error encountered.

Type: string

Required: False

ConflictException

The requested operation would cause a conflict with the current state of a service resource associated with the request. Resolve the conflict before retrying this request. See the accompanying error message for details.

message

Describes the error encountered.

Type: string

Required: False

CreateStageInput

Represents the input parameters for a `CreateStage` request.

stageName

The name of the stage.

Type: string

Required: True

description

The description for the API stage.

Type: string

Required: False

deploymentId

The deployment identifier of the API stage.

Type: string

Required: False

clientCertificateId

The identifier of a client certificate for a Stage. Supported only for WebSocket APIs.

Type: string

Required: False

defaultRouteSettings

The default route settings for the stage.

Type: [RouteSettings](#)

Required: False

routeSettings

Route settings for the stage, by routeKey.

Type: [RouteSettingsMap](#)

Required: False

stageVariables

A map that defines the stage variables for a Stage. Variable names can have alphanumeric and underscore characters, and the values must match `[A-Za-z0-9-._~:/?#&=,]+`.

Type: [StageVariablesMap](#)

Required: False

accessLogSettings

Settings for logging access in this stage.

Type: [AccessLogSettings](#)

Required: False

tags

The collection of tags. Each tag element is associated with a given resource.

Type: [Tags](#)

Required: False

autoDeploy

Specifies whether updates to an API automatically trigger a new deployment. The default value is false.

Type: boolean

Required: False

LimitExceededException

A limit has been exceeded. See the accompanying error message for details.

message

Describes the error encountered.

Type: string

Required: False

limitType

The limit type.

Type: string

Required: False

LoggingLevel

The logging level.

ERROR

INFO

false

NotFoundException

The resource specified in the request was not found. See the message field for more information.

message

Describes the error encountered.

Type: string

Required: False

resourceType

The resource type.

Type: string

Required: False

RouteSettings

Represents a collection of route settings.

detailedMetricsEnabled

Specifies whether detailed metrics are enabled.

Type: boolean

Required: False

loggingLevel

Specifies the logging level for this route: INFO, ERROR, or OFF. This property affects the log entries pushed to Amazon CloudWatch Logs. Supported only for WebSocket APIs.

Type: [LoggingLevel](#)

Required: False

dataTraceEnabled

Specifies whether (`true`) or not (`false`) data trace logging is enabled for this route. This property affects the log entries pushed to Amazon CloudWatch Logs. Supported only for WebSocket APIs.

Type: boolean

Required: False

throttlingBurstLimit

Specifies the throttling burst limit.

Type: integer

Required: False

Format: int32

throttlingRateLimit

Specifies the throttling rate limit.

Type: number

Required: False

Format: double

RouteSettingsMap

The route settings map.

key-value pairs

Type: object

Stage

Represents an API stage.

stageName

The name of the stage.

Type: string
Required: True

description

The description of the stage.

Type: string
Required: False

deploymentId

The identifier of the Deployment that the Stage is associated with. Can't be updated if autoDeploy is enabled.

Type: string
Required: False

clientCertificateId

The identifier of a client certificate for a Stage. Supported only for WebSocket APIs.

Type: string
Required: False

defaultRouteSettings

Default route settings for the stage.

Type: [RouteSettings](#)
Required: False

routeSettings

Route settings for the stage, by routeKey.

Type: [RouteSettingsMap](#)
Required: False

stageVariables

A map that defines the stage variables for a stage resource. Variable names can have alphanumeric and underscore characters, and the values must match [A-Za-z0-9-._~:/?#&=,]+.

Type: [StageVariablesMap](#)

Required: False

accessLogSettings

Settings for logging access in this stage.

Type: [AccessLogSettings](#)

Required: False

autoDeploy

Specifies whether updates to an API automatically trigger a new deployment. The default value is false.

Type: boolean

Required: False

lastDeploymentStatusMessage

Describes the status of the last deployment of a stage. Supported only for stages with autoDeploy enabled.

Type: string

Required: False

createdDate

The timestamp when the stage was created.

Type: string

Required: False

Format: date-time

lastUpdatedDate

The timestamp when the stage was last updated.

Type: string

Required: False

Format: date-time

tags

The collection of tags. Each tag element is associated with a given resource.

Type: [Tags](#)

Required: False

apiGatewayManaged

Specifies whether a stage is managed by API Gateway. If you created an API using quick create, the `$default` stage is managed by API Gateway. You can't modify the `$default` stage.

Type: boolean

Required: False

StageVariablesMap

The stage variable map.

key-value pairs

Type: string

Stages

A collection of Stage resources that are associated with the ApiKey resource.

items

The elements from this collection.

Type: Array of type [Stage](#)

Required: False

nextToken

The next page of elements from this collection. Not valid for the last element of the collection.

Type: string

Required: False

Tags

Represents a collection of tags associated with the resource.

key-value pairs

Type: string

Tags

Represents a collection of tags associated with the resource.

URI

`/v2/tags/resource-arn`

HTTP methods

GET

Operation ID: GetTags

Gets a collection of Tag resources.

Path parameters

Name	Type	Required	Description
<i>resource-arn</i>	String	True	The resource ARN for the tag.

Responses

Status code	Response model	Description
200	Tags	Success
400	BadRequestException	One of the parameters in the request is invalid.
404	NotFoundException	The resource specified in the request was not found.
409	ConflictException	The resource already exists.
429	LimitExceededException	The client is sending more than the allowed number of requests per unit of time.

POST

Operation ID: TagResource

Creates a new Tag resource to represent a tag.

Path parameters

Name	Type	Required	Description
<i>resource-arn</i>	String	True	The resource ARN for the tag.

Responses

Status code	Response model	Description
201	None	The request has succeeded and has resulted in the creation of a resource.

Status code	Response model	Description
400	BadRequestException	One of the parameters in the request is invalid.
404	NotFoundException	The resource specified in the request was not found.
409	ConflictException	The resource already exists.
429	LimitExceededException	The client is sending more than the allowed number of requests per unit of time.

DELETE

Operation ID: UntagResource

Deletes a Tag.

Path parameters

Name	Type	Required	Description
<i>resource-arn</i>	String	True	The resource ARN for the tag.

Query parameters

Name	Type	Required	Description
tagKeys	String	True	The tag keys.

Responses

Status code	Response model	Description
204	None	The request has succeeded , and there is no additiona

Status code	Response model	Description
400	BadRequestException	Content to send in the response payload body. One of the parameters in the request is invalid.
404	NotFoundException	The resource specified in the request was not found.
409	ConflictException	The resource already exists.
429	LimitExceededException	The client is sending more than the allowed number of requests per unit of time.

Schemas

Request bodies

POST schema

```
{  
  "tags": {  
  }  
}
```

Response bodies

Tags schema

```
{  
}
```

BadRequestException schema

```
{
```

```
"message": "string"
}
```

NotFoundException schema

```
{
  "message": "string",
  "resourceType": "string"
}
```

ConflictException schema

```
{
  "message": "string"
}
```

LimitExceededException schema

```
{
  "message": "string",
  "limitType": "string"
}
```

Properties

BadRequestException

The request is not valid, for example, the input is incomplete or incorrect. See the accompanying error message for details.

message

Describes the error encountered.

Type: string

Required: False

ConflictException

The requested operation would cause a conflict with the current state of a service resource associated with the request. Resolve the conflict before retrying this request. See the accompanying error message for details.

message

Describes the error encountered.

Type: string

Required: False

LimitExceededException

A limit has been exceeded. See the accompanying error message for details.

message

Describes the error encountered.

Type: string

Required: False

limitType

The limit type.

Type: string

Required: False

NotFoundException

The resource specified in the request was not found. See the message field for more information.

message

Describes the error encountered.

Type: string

Required: False

resourceType

The resource type.

Type: string

Required: False

TagResourceInput

Represents the input parameters for a TagResource request.

tags

The collection of tags. Each tag element is associated with a given resource.

Type: [Tags](#)

Required: False

Tags

Represents a collection of tags associated with the resource.

key-value pairs

Type: string

VPCLink

Represents a VPC link. A VPC link enables you to create integrations with private resources in a VPC.

URI

`/v2/vpcLinks/vpcLinkId`

HTTP methods

GET

Operation ID: GetVpcLink

Gets a VPC link.

Path parameters

Name	Type	Required	Description
<i>vpcLinkId</i>	String	True	The ID of the VPC link.

Responses

Status code	Response model	Description
200	VpcLink	Success
404	NotFoundException	The resource specified in the request was not found.
429	LimitExceededException	The client is sending more than the allowed number of requests per unit of time.

DELETE

Operation ID: DeleteVpcLink

Deletes a VPC link.

Path parameters

Name	Type	Required	Description
<i>vpcLinkId</i>	String	True	The ID of the VPC link.

Responses

Status code	Response model	Description
202	None	202 response
404	NotFoundException	The resource specified in the request was not found.
429	LimitExceededException	The client is sending more than the allowed number of requests per unit of time.

PATCH

Operation ID: UpdateVpcLink

Updates a VPC link.

Path parameters

Name	Type	Required	Description
<i>vpcLinkId</i>	String	True	The ID of the VPC link.

Responses

Status code	Response model	Description
200	VpcLink	200 response
400	BadRequestException	One of the parameters in the request is invalid.
404	NotFoundException	The resource specified in the request was not found.

Status code	Response model	Description
429	LimitExceededException	The client is sending more than the allowed number of requests per unit of time.

Schemas

Request bodies

PATCH schema

```
{  
  "name": "string"  
}
```

Response bodies

VpcLink schema

```
{  
  "vpcLinkId": "string",  
  "name": "string",  
  "subnetIds": [  
    "string"  
  ],  
  "securityGroupIds": [  
    "string"  
  ],  
  "tags": {  
  },  
  "createdDate": "string",  
  "vpcLinkStatus": enum,  
  "vpcLinkStatusMessage": "string",  
  "vpcLinkVersion": enum  
}
```

BadRequestException schema

```
{
```

```
"message": "string"
}
```

NotFoundException schema

```
{
  "message": "string",
  "resourceType": "string"
}
```

LimitExceededException schema

```
{
  "message": "string",
  "limitType": "string"
}
```

Properties

BadRequestException

The request is not valid, for example, the input is incomplete or incorrect. See the accompanying error message for details.

message

Describes the error encountered.

Type: string

Required: False

LimitExceededException

A limit has been exceeded. See the accompanying error message for details.

message

Describes the error encountered.

Type: string

Required: False

limitType

The limit type.

Type: string

Required: False

NotFoundException

The resource specified in the request was not found. See the message field for more information.

message

Describes the error encountered.

Type: string

Required: False

resourceType

The resource type.

Type: string

Required: False

Tags

Represents a collection of tags associated with the resource.

key-value pairs

Type: string

UpdateVpcLinkInput

Represents the input parameters for an UpdateVpcLink request.

name

The name of the VPC link.

Type: string

Required: False

VpcLink

Represents a VPC link.

vpcLinkId

The ID of the VPC link.

Type: string

Required: True

name

The name of the VPC link.

Type: string

Required: True

subnetIds

A list of subnet IDs to include in the VPC link.

Type: Array of type string

Required: True

securityGroupIds

A list of security group IDs for the VPC link.

Type: Array of type string

Required: True

tags

Tags for the VPC link.

Type: [Tags](#)

Required: False

createdDate

The timestamp when the VPC link was created.

Type: string

Required: False

Format: date-time

vpLinkStatus

The status of the VPC link.

Type: [VpcLinkStatus](#)

Required: False

vpLinkStatusMessage

A message summarizing the cause of the status of the VPC link.

Type: string

Required: False

vpLinkVersion

The version of the VPC link.

Type: [VpcLinkVersion](#)

Required: False

VpcLinkStatus

The status of the VPC link.

PENDING
AVAILABLE
DELETING
FAILED
INACTIVE

VpcLinkVersion

The version of the VPC link.

V2

VPCLinks

Represents your VPC links as a collection. A collection offers a paginated view of your VPC links.

URI

/v2/vpclinks

HTTP methods

GET

Operation ID: GetVpcLinks

Gets a collection of VPC links.

Query parameters

Name	Type	Required	Description
nextToken	String	False	The next page of elements from this collection. Not valid for the last element of the collection.
maxResults	String	False	The maximum number of elements

Name	Type	Required	Description
			to be returned for this resource.

Responses

Status code	Response model	Description
200	VpcLinks	Success
400	BadRequestException	One of the parameters in the request is invalid.
429	LimitExceededException	The client is sending more than the allowed number of requests per unit of time.

POST

Operation ID: CreateVpcLink

Creates a VPC link.

Responses

Status code	Response model	Description
201	VpcLink	The request has succeeded and has resulted in the creation of a resource.
400	BadRequestException	One of the parameters in the request is invalid.
429	LimitExceededException	The client is sending more than the allowed number of requests per unit of time.

Schemas

Request bodies

POST schema

```
{
  "name": "string",
  "subnetIds": [
    "string"
  ],
  "securityGroupIds": [
    "string"
  ],
  "tags": {
  }
}
```

Response bodies

VpcLinks schema

```
{
  "items": [
    {
      "vpcLinkId": "string",
      "name": "string",
      "subnetIds": [
        "string"
      ],
      "securityGroupIds": [
        "string"
      ],
      "tags": {
      },
      "createdDate": "string",
      "vpcLinkStatus": enum,
      "vpcLinkStatusMessage": "string",
      "vpcLinkVersion": enum
    }
  ],
  "nextToken": "string"
}
```



```
}
```

VpcLink schema

```
{
  "vpcLinkId": "string",
  "name": "string",
  "subnetIds": [
    "string"
  ],
  "securityGroupIds": [
    "string"
  ],
  "tags": {
  },
  "createdDate": "string",
  "vpcLinkStatus": enum,
  "vpcLinkStatusMessage": "string",
  "vpcLinkVersion": enum
}
```

BadRequestException schema

```
{
  "message": "string"
}
```

LimitExceededException schema

```
{
  "message": "string",
  "limitType": "string"
}
```

Properties

BadRequestException

The request is not valid, for example, the input is incomplete or incorrect. See the accompanying error message for details.

message

Describes the error encountered.

Type: string

Required: False

CreateVpcLinkInput

Represents the input parameters for a CreateVpcLink request.

name

The name of the VPC link.

Type: string

Required: True

subnetIds

A list of subnet IDs to include in the VPC link.

Type: Array of type string

Required: True

securityGroupIds

A list of security group IDs for the VPC link.

Type: Array of type string

Required: False

tags

A list of tags.

Type: [Tags](#)

Required: False

LimitExceededException

A limit has been exceeded. See the accompanying error message for details.

message

Describes the error encountered.

Type: string

Required: False

limitType

The limit type.

Type: string

Required: False

Tags

Represents a collection of tags associated with the resource.

key-value pairs

Type: string

VpcLink

Represents a VPC link.

vpcLinkId

The ID of the VPC link.

Type: string

Required: True

name

The name of the VPC link.

Type: string

Required: True

subnetIds

A list of subnet IDs to include in the VPC link.

Type: Array of type string

Required: True

securityGroupIds

A list of security group IDs for the VPC link.

Type: Array of type string

Required: True

tags

Tags for the VPC link.

Type: [Tags](#)

Required: False

createdDate

The timestamp when the VPC link was created.

Type: string

Required: False

Format: date-time

vpclinkStatus

The status of the VPC link.

Type: [VpcLinkStatus](#)

Required: False

vpcLinkStatusMessage

A message summarizing the cause of the status of the VPC link.

Type: string

Required: False

vpcLinkVersion

The version of the VPC link.

Type: [VpcLinkVersion](#)

Required: False

VpcLinkStatus

The status of the VPC link.

PENDING

AVAILABLE

DELETING

FAILED

INACTIVE

VpcLinkVersion

The version of the VPC link.

V2

VpcLinks

Represents a collection of VPCLinks.

items

A collection of VPC links.

Type: Array of type [VpcLink](#)

Required: False

nextToken

The next page of elements from this collection. Not valid for the last element of the collection.

Type: string

Required: False

Operations

The REST API includes the following operations

- [CreateApi](#)
Creates an `Api` resource.
- [CreateApiMapping](#)
Creates an API mapping.
- [CreateAuthorizer](#)
Creates an `Authorizer` for an API.
- [CreateDeployment](#)
Creates a `Deployment` for an API.
- [CreateDomainName](#)
Creates a domain name.
- [CreateIntegration](#)
Creates an `Integration`.
- [CreateIntegrationResponse](#)
Creates an `IntegrationResponses`.
- [CreateModel](#)
Creates a `Model` for an API.
- [CreateRoute](#)
Creates a `Route` for an API.
- [CreateRouteResponse](#)
Creates a `RouteResponse` for a `Route`.
- [CreateStage](#)
Creates a `Stage` for an API.

- [CreateVpcLink](#)

Creates a VPC link.

- [DeleteAccessLogSettings](#)

Deletes the AccessLogSettings for a Stage. To disable access logging for a Stage, delete its AccessLogSettings.

- [DeleteApi](#)

Deletes an Api resource.

- [DeleteApiMapping](#)

Deletes an API mapping.

- [DeleteAuthorizer](#)

Deletes an Authorizer.

- [DeleteCorsConfiguration](#)

Deletes a CORS configuration.

- [DeleteDeployment](#)

Deletes a Deployment.

- [DeleteDomainName](#)

Deletes a domain name.

- [DeleteIntegration](#)

Deletes an Integration.

- [DeleteIntegrationResponse](#)

Deletes an IntegrationResponses.

- [DeleteModel](#)

Deletes a Model.

- [DeleteRoute](#)

Deletes a Route.

- [DeleteRouteRequestParameter](#)

Deletes a route request parameter. Supported only for WebSocket APIs.

- [DeleteRouteResponse](#)

Deletes a RouteResponse.

- [DeleteRouteSettings](#)

Deletes the RouteSettings for a stage.

- [DeleteStage](#)

Deletes a Stage.

- [DeleteVpcLink](#)

Deletes a VPC link.

- [ExportApi](#)

Exports a definition of an API in a particular output format and specification.

- [GetApi](#)

Gets an Api resource.

- [GetApiMapping](#)

Gets an API mapping.

- [GetApiMappings](#)

Gets API mappings.

- [GetApis](#)

Gets a collection of Api resources.

- [GetAuthorizer](#)

Gets an Authorizer.

- [GetAuthorizers](#)

Gets the Authorizers for an API.

- [GetDeployment](#)

Gets a Deployment.

- [GetDeployments](#)

Gets the Deployments for an API.

- [GetDomainName](#)

Gets a domain name.

- [GetDomainNames](#)

Gets the domain names for an Amazon Web Services account.

- [GetIntegration](#)

Gets an Integration.

- [GetIntegrationResponse](#)

Gets an IntegrationResponses.

- [GetIntegrationResponses](#)

Gets the IntegrationResponses for an Integration.

- [GetIntegrations](#)

Gets the Integrations for an API.

- [GetModel](#)

Gets a Model.

- [GetModels](#)

Gets the Models for an API.

- [GetModelTemplate](#)

Gets a model template.

- [GetRoute](#)

Gets a Route.

- [GetRouteResponse](#)

Gets a RouteResponse.

- [GetRouteResponses](#)

Gets the RouteResponses for a Route.

- [GetRoutes](#)

Gets the Routes for an API.

- [GetStage](#)

Gets a Stage.

- [GetStages](#)

Gets the Stages for an API.

- [GetTags](#)

Gets a collection of Tag resources.

- [GetVpcLink](#)

Gets a VPC link.

- [GetVpcLinks](#)

Gets a collection of VPC links.

- [ImportApi](#)

Imports an API.

- [ReimportApi](#)

Puts an Api resource.

- [ResetAuthorizersCache](#)

Resets all authorizer cache entries for the specified stage. Supported only for HTTP API Lambda authorizers.

- [TagResource](#)

Creates a new Tag resource to represent a tag.

- [UntagResource](#)

Deletes a Tag.

- [UpdateApi](#)

Updates an Api resource.

- [UpdateApiMapping](#)

The API mapping.

- [UpdateAuthorizer](#)

Updates an Authorizer.

- [UpdateDeployment](#)

Updates a Deployment.

- [UpdateDomainName](#)

Updates a domain name.

- [UpdateIntegration](#)

Updates an Integration.

- [UpdateIntegrationResponse](#)

Updates an IntegrationResponses.

- [UpdateModel](#)

Updates a Model.

- [UpdateRoute](#)

Updates a Route.

- [UpdateRouteResponse](#)

Updates a RouteResponse.

- [UpdateStage](#)

Updates a Stage.

- [UpdateVpcLink](#)

Updates a VPC link.

Document History for the API Gateway WebSocket and HTTP API Reference

The following table describes major updates to the Amazon API Gateway WebSocket and HTTP API Reference.

- **Latest documentation update:** March 28, 2025

Change	Description	Date
IP address types	Added support for IP address types for APIs and domain names.	March 28, 2025
Mutual TLS authentication	Added support for mutual TLS authentication for HTTP APIs.	September 17, 2020
HTTP API Lambda and IAM authorizers	Added support for Lambda and IAM authorizers for HTTP APIs.	September 9, 2020
HTTP API Amazon service integrations	Added support for Amazon service integrations for HTTP APIs. To learn more, see Integration .	August 20, 2020
HTTP API export	Added support for exporting definitions of HTTP APIs. To learn more, see ExportedAPI .	April 20, 2020
HTTP APIs	General availability release of HTTP APIs.	March 12, 2020
HTTP APIs	Beta release of HTTP APIs.	December 4, 2019
Major release	Initial documentation release.	November 29, 2018

Amazon Glossary

For the latest Amazon terminology, see the [Amazon glossary](#) in the *Amazon Web Services Glossary Reference*.