aws

API Reference

Direct Connect

API Version 2012-10-25

Direct Connect: API Reference

Table of Contents

Welcome

Direct Connect links your internal network to an Direct Connect location over a standard Ethernet fiber-optic cable. One end of the cable is connected to your router, the other to an Direct Connect router. With this connection in place, you can create virtual interfaces directly to the AWS Cloud (for example, to Amazon EC2 and Amazon S3) and to Amazon VPC, bypassing Internet service providers in your network path. A connection provides access to all AWS Regions except the China (Beijing) and (China) Ningxia Regions. AWS resources in the China Regions can only be accessed through locations associated with those Regions.

This document was last published on April 10, 2026.

Actions

The following actions are supported:

- AcceptDirectConnectGatewayAssociationProposal
- AllocateConnectionOnInterconnect
- AllocateHostedConnection
- AllocatePrivateVirtualInterface
- AllocatePublicVirtualInterface
- AllocateTransitVirtualInterface
- AssociateConnectionWithLag
- AssociateHostedConnection
- AssociateMacSecKey
- AssociateVirtualInterface
- ConfirmConnection
- ConfirmCustomerAgreement
- ConfirmPrivateVirtualInterface
- ConfirmPublicVirtualInterface
- ConfirmTransitVirtualInterface
- CreateBGPPeer
- CreateConnection
- CreateDirectConnectGateway
- CreateDirectConnectGatewayAssociation
- CreateDirectConnectGatewayAssociationProposal
- CreateInterconnect
- CreateLag
- CreatePrivateVirtualInterface
- CreatePublicVirtualInterface
- CreateTransitVirtualInterface
- DeleteBGPPeer
- DeleteConnection

- DeleteDirectConnectGateway
- DeleteDirectConnectGatewayAssociation
- DeleteDirectConnectGatewayAssociationProposal
- DeleteInterconnect
- DeleteLag
- DeleteVirtualInterface
- DescribeConnectionLoa
- DescribeConnections
- DescribeConnectionsOnInterconnect
- DescribeCustomerMetadata
- DescribeDirectConnectGatewayAssociationProposals
- DescribeDirectConnectGatewayAssociations
- DescribeDirectConnectGatewayAttachments
- DescribeDirectConnectGateways
- DescribeHostedConnections
- DescribeInterconnectLoa
- DescribeInterconnects
- DescribeLags
- DescribeLoa
- DescribeLocations
- DescribeRouterConfiguration
- DescribeTags
- DescribeVirtualGateways
- DescribeVirtualInterfaces
- DisassociateConnectionFromLag
- DisassociateMacSecKey
- ListVirtualInterfaceTestHistory
- StartBgpFailoverTest
- StopBgpFailoverTest
- TagResource

- UntagResource
- UpdateConnection
- UpdateDirectConnectGateway
- UpdateDirectConnectGatewayAssociation
- UpdateLag
- UpdateVirtualInterfaceAttributes

AcceptDirectConnectGatewayAssociationProposal

Accepts a proposal request to attach a virtual private gateway or transit gateway to a Direct Connect gateway.

Request Syntax

```
{
   "associatedGatewayOwnerAccount": "string",
   "directConnectGatewayId": "string",
   "overrideAllowedPrefixesToDirectConnectGateway": [
      {
         "cidr": "string"
      }
   ],
   "proposalId": "string"
}
```

Request Parameters

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

The request accepts the following data in JSON format.

associatedGatewayOwnerAccount

The ID of the AWS account that owns the virtual private gateway or transit gateway.

Type: String

Required: Yes

directConnectGatewayId

The ID of the Direct Connect gateway.

Type: String

Required: Yes

overrideAllowedPrefixesToDirectConnectGateway

Overrides the Amazon VPC prefixes advertised to the Direct Connect gateway.

For information about how to set the prefixes, see Allowed Prefixes in the *Direct Connect User Guide*.

Type: Array of RouteFilterPrefix objects

Required: No

proposalId

The ID of the request proposal.

Type: String

Required: Yes

Response Syntax

```
{
   "directConnectGatewayAssociation": {
      "allowedPrefixesToDirectConnectGateway": [
         {
            "cidr": "string"
         }
      ],
      "associatedCoreNetwork": {
         "attachmentId": "string",
         "id": "string",
         "ownerAccount": "string"
      },
      "associatedGateway": {
         "id": "string",
         "ownerAccount": "string",
         "region": "string",
         "type": "string"
      },
      "associationId": "string",
      "associationState": "string",
      "directConnectGatewayId": "string",
      "directConnectGatewayOwnerAccount": "string",
      "stateChangeError": "string",
      "virtualGatewayId": "string",
      "virtualGatewayOwnerAccount": "string",
      "virtualGatewayRegion": "string"
```

```
    }
}
```

Response Elements

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

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

directConnectGatewayAssociation

Information about an association between a Direct Connect gateway and a virtual gateway or transit gateway.

Type: DirectConnectGatewayAssociation object

Errors

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

DirectConnectClientException

One or more parameters are not valid.

HTTP Status Code: 400

DirectConnectServerException

A server-side error occurred.

HTTP Status Code: 400

See Also

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

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

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

AllocateConnectionOnInterconnect

> **Note**
>
> Deprecated. Use AllocateHostedConnection instead.

Creates a hosted connection on an interconnect.

Allocates a VLAN number and a specified amount of bandwidth for use by a hosted connection on the specified interconnect.

> **Note**
>
> Intended for use by Direct Connect Partners only.

Request Syntax

```
{
   "bandwidth": "string",
   "connectionName": "string",
   "interconnectId": "string",
   "ownerAccount": "string",
   "vlan": number
}
```

Request Parameters

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

The request accepts the following data in JSON format.

bandwidth

The bandwidth of the connection. The possible values are 50Mbps, 100Mbps, 200Mbps, 300Mbps, 400Mbps, 500Mbps, 1Gbps, 2Gbps, 5Gbps, and 10Gbps. Note that only those Direct Connect Partners who have met specific requirements are allowed to create a 1Gbps, 2Gbps, 5Gbps or 10Gbps hosted connection.

Type: String

Required: Yes

connectionName

The name of the provisioned connection.

Type: String

Required: Yes

interconnectId

The ID of the interconnect on which the connection will be provisioned.

Type: String

Required: Yes

ownerAccount

The ID of the AWS account of the customer for whom the connection will be provisioned.

Type: String

Required: Yes

vlan

The dedicated VLAN provisioned to the connection.

Type: Integer

Required: Yes

Response Syntax

```
{
   "awsDevice": "string",
   "awsDeviceV2": "string",
   "awsLogicalDeviceId": "string",
   "bandwidth": "string",
   "connectionId": "string",
```

```
   "connectionName": "string",
   "connectionState": "string",
   "encryptionMode": "string",
   "hasLogicalRedundancy": "string",
   "jumboFrameCapable": boolean,
   "lagId": "string",
   "loaIssueTime": number,
   "location": "string",
   "macSecCapable": boolean,
   "macSecKeys": [
      {
         "ckn": "string",
         "secretARN": "string",
         "startOn": "string",
         "state": "string"
      }
   ],
   "ownerAccount": "string",
   "partnerInterconnectMacSecCapable": boolean,
   "partnerName": "string",
   "portEncryptionStatus": "string",
   "providerName": "string",
   "region": "string",
   "tags": [
      {
         "key": "string",
         "value": "string"
      }
   ],
   "vlan": number
}
```

Response Elements

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

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

awsDevice

This parameter has been deprecated.

The Direct Connect endpoint on which the physical connection terminates.

Type: String

awsDeviceV2

The Direct Connect endpoint that terminates the physical connection.

Type: String

awsLogicalDeviceId

The Direct Connect endpoint that terminates the logical connection. This device might be different than the device that terminates the physical connection.

Type: String

bandwidth

The bandwidth of the connection.

Type: String

connectionId

The ID of the connection.

Type: String

connectionName

The name of the connection.

Type: String

connectionState

The state of the connection. The following are the possible values:

- `ordering`: The initial state of a hosted connection provisioned on an interconnect. The connection stays in the ordering state until the owner of the hosted connection confirms or declines the connection order.
- `requested`: The initial state of a standard connection. The connection stays in the requested state until the Letter of Authorization (LOA) is sent to the customer.
- `pending`: The connection has been approved and is being initialized.
- `available`: The network link is up and the connection is ready for use.
- down: The network link is down.

- `deleting`: The connection is being deleted.
- `deleted`: The connection has been deleted.
- `rejected`: A hosted connection in the `ordering` state enters the `rejected` state if it is deleted by the customer.
- unknown: The state of the connection is not available.

Type: String

Valid Values: `ordering | requested | pending | available | down | deleting | deleted | rejected | unknown`

encryptionMode

The MAC Security (MACsec) connection encryption mode.

The valid values are `no_encrypt`, `should_encrypt`, and `must_encrypt`.

Type: String

hasLogicalRedundancy

Indicates whether the connection supports a secondary BGP peer in the same address family (IPv4/IPv6).

Type: String

Valid Values: `unknown | yes | no`

jumboFrameCapable

Indicates whether jumbo frames are supported.

Type: Boolean

lagId

The ID of the LAG.

Type: String

loaIssueTime

The time of the most recent call to DescribeLoa for this connection.

Type: Timestamp

location

The location of the connection.

Type: String

macSecCapable

Indicates whether the connection supports MAC Security (MACsec).

Type: Boolean

macSecKeys

The MAC Security (MACsec) security keys associated with the connection.

Type: Array of MacSecKey objects

ownerAccount

The ID of the AWS account that owns the connection.

Type: String

partnerInterconnectMacSecCapable

Indicates whether the interconnect hosting this connection supports MAC Security (MACsec).

Type: Boolean

partnerName

The name of the Direct Connect service provider associated with the connection.

Type: String

portEncryptionStatus

The MAC Security (MACsec) port link status of the connection.

The valid values are `Encryption Up`, which means that there is an active Connection Key Name, or `Encryption Down`.

Type: String

providerName

The name of the service provider associated with the connection.

Type: String

region

The AWS Region where the connection is located.

Type: String

tags

The tags associated with the connection.

Type: Array of Tag objects

Array Members: Minimum number of 1 item.

vlan

The ID of the VLAN.

Type: Integer

Errors

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

DirectConnectClientException

One or more parameters are not valid.

HTTP Status Code: 400

DirectConnectServerException

A server-side error occurred.

HTTP Status Code: 400

See Also

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

- AWS Command Line Interface V2

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

AllocateHostedConnection

Creates a hosted connection on the specified interconnect or a link aggregation group (LAG) of interconnects.

Allocates a VLAN number and a specified amount of capacity (bandwidth) for use by a hosted connection on the specified interconnect or LAG of interconnects. AWS polices the hosted connection for the specified capacity and the Direct Connect Partner must also police the hosted connection for the specified capacity.

> **Note**
>
> Intended for use by Direct Connect Partners only.

Request Syntax

```
{
   "bandwidth": "string",
   "connectionId": "string",
   "connectionName": "string",
   "ownerAccount": "string",
   "tags": [
      {
         "key": "string",
         "value": "string"
      }
   ],
   "vlan": number
}
```

Request Parameters

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

The request accepts the following data in JSON format.

bandwidth

The bandwidth of the connection. The possible values are 50Mbps, 100Mbps, 200Mbps, 300Mbps, 400Mbps, 500Mbps, 1Gbps, 2Gbps, 5Gbps, 10Gbps, and 25Gbps. Note that only

those Direct Connect Partners who have met specific requirements are allowed to create a 1Gbps, 2Gbps, 5Gbps, 10Gbps, or 25Gbps hosted connection.

Type: String

Required: Yes

connectionId

The ID of the interconnect or LAG.

Type: String

Required: Yes

connectionName

The name of the hosted connection.

Type: String

Required: Yes

ownerAccount

The ID of the AWS account ID of the customer for the connection.

Type: String

Required: Yes

tags

The tags associated with the connection.

Type: Array of Tag objects

Array Members: Minimum number of 1 item.

Required: No

vlan

The dedicated VLAN provisioned to the hosted connection.

Type: Integer

Required: Yes

Response Syntax

```
{
   "awsDevice": "string",
   "awsDeviceV2": "string",
   "awsLogicalDeviceId": "string",
   "bandwidth": "string",
   "connectionId": "string",
   "connectionName": "string",
   "connectionState": "string",
   "encryptionMode": "string",
   "hasLogicalRedundancy": "string",
   "jumboFrameCapable": boolean,
   "lagId": "string",
   "loaIssueTime": number,
   "location": "string",
   "macSecCapable": boolean,
   "macSecKeys": [
      {
         "ckn": "string",
         "secretARN": "string",
         "startOn": "string",
         "state": "string"
      }
   ],
   "ownerAccount": "string",
   "partnerInterconnectMacSecCapable": boolean,
   "partnerName": "string",
   "portEncryptionStatus": "string",
   "providerName": "string",
   "region": "string",
   "tags": [
      {
         "key": "string",
         "value": "string"
      }
   ],
   "vlan": number
}
```

Response Elements

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

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

awsDevice

This parameter has been deprecated.

The Direct Connect endpoint on which the physical connection terminates.

Type: String

awsDeviceV2

The Direct Connect endpoint that terminates the physical connection.

Type: String

awsLogicalDeviceId

The Direct Connect endpoint that terminates the logical connection. This device might be different than the device that terminates the physical connection.

Type: String

bandwidth

The bandwidth of the connection.

Type: String

connectionId

The ID of the connection.

Type: String

connectionName

The name of the connection.

Type: String

connectionState

The state of the connection. The following are the possible values:

- `ordering`: The initial state of a hosted connection provisioned on an interconnect. The connection stays in the ordering state until the owner of the hosted connection confirms or declines the connection order.

- `requested`: The initial state of a standard connection. The connection stays in the requested state until the Letter of Authorization (LOA) is sent to the customer.
- `pending`: The connection has been approved and is being initialized.
- `available`: The network link is up and the connection is ready for use.
- down: The network link is down.
- `deleting`: The connection is being deleted.
- `deleted`: The connection has been deleted.
- `rejected`: A hosted connection in the `ordering` state enters the `rejected` state if it is deleted by the customer.
- unknown: The state of the connection is not available.

Type: String

Valid Values: `ordering | requested | pending | available | down | deleting | deleted | rejected | unknown`

encryptionMode

The MAC Security (MACsec) connection encryption mode.

The valid values are `no_encrypt`, `should_encrypt`, and `must_encrypt`.

Type: String

hasLogicalRedundancy

Indicates whether the connection supports a secondary BGP peer in the same address family (IPv4/IPv6).

Type: String

Valid Values: `unknown | yes | no`

jumboFrameCapable

Indicates whether jumbo frames are supported.

Type: Boolean

lagId

The ID of the LAG.

Type: String

loaIssueTime

The time of the most recent call to DescribeLoa for this connection.

Type: Timestamp

location

The location of the connection.

Type: String

macSecCapable

Indicates whether the connection supports MAC Security (MACsec).

Type: Boolean

macSecKeys

The MAC Security (MACsec) security keys associated with the connection.

Type: Array of MacSecKey objects

ownerAccount

The ID of the AWS account that owns the connection.

Type: String

partnerInterconnectMacSecCapable

Indicates whether the interconnect hosting this connection supports MAC Security (MACsec).

Type: Boolean

partnerName

The name of the Direct Connect service provider associated with the connection.

Type: String

portEncryptionStatus

The MAC Security (MACsec) port link status of the connection.

The valid values are `Encryption Up`, which means that there is an active Connection Key Name, or `Encryption Down`.

Type: String

providerName

The name of the service provider associated with the connection.

Type: String

region

The AWS Region where the connection is located.

Type: String

tags

The tags associated with the connection.

Type: Array of Tag objects

Array Members: Minimum number of 1 item.

vlan

The ID of the VLAN.

Type: Integer

Errors

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

DirectConnectClientException

One or more parameters are not valid.

HTTP Status Code: 400

DirectConnectServerException

A server-side error occurred.

HTTP Status Code: 400

DuplicateTagKeysException

A tag key was specified more than once.

HTTP Status Code: 400

TooManyTagsException

You have reached the limit on the number of tags that can be assigned.

HTTP Status Code: 400

See Also

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

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

AllocatePrivateVirtualInterface

Provisions a private virtual interface to be owned by the specified AWS account.

Virtual interfaces created using this action must be confirmed by the owner using ConfirmPrivateVirtualInterface. Until then, the virtual interface is in the `Confirming` state and is not available to handle traffic.

Request Syntax

```
{
   "connectionId": "string",
   "newPrivateVirtualInterfaceAllocation": {
      "addressFamily": "string",
      "amazonAddress": "string",
      "asn": number,
      "asnLong": number,
      "authKey": "string",
      "customerAddress": "string",
      "mtu": number,
      "tags": [
         {
            "key": "string",
            "value": "string"
         }
      ],
      "virtualInterfaceName": "string",
      "vlan": number
   },
   "ownerAccount": "string"
}
```

Request Parameters

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

The request accepts the following data in JSON format.

connectionId

The ID of the connection on which the private virtual interface is provisioned.

Type: String

Required: Yes

newPrivateVirtualInterfaceAllocation

Information about the private virtual interface.

Type: NewPrivateVirtualInterfaceAllocation object

Required: Yes

ownerAccount

The ID of the AWS account that owns the virtual private interface.

Type: String

Required: Yes

Response Syntax

```
{
   "addressFamily": "string",
   "amazonAddress": "string",
   "amazonSideAsn": number,
   "asn": number,
   "asnLong": number,
   "authKey": "string",
   "awsDeviceV2": "string",
   "awsLogicalDeviceId": "string",
   "bgpPeers": [
      {
         "addressFamily": "string",
         "amazonAddress": "string",
         "asn": number,
         "asnLong": number,
         "authKey": "string",
         "awsDeviceV2": "string",
         "awsLogicalDeviceId": "string",
         "bgpPeerId": "string",
         "bgpPeerState": "string",
         "bgpStatus": "string",
         "customerAddress": "string"
```

```
      }
   ],
   "connectionId": "string",
   "customerAddress": "string",
   "customerRouterConfig": "string",
   "directConnectGatewayId": "string",
   "jumboFrameCapable": boolean,
   "location": "string",
   "mtu": number,
   "ownerAccount": "string",
   "region": "string",
   "routeFilterPrefixes": [
      {
         "cidr": "string"
      }
   ],
   "siteLinkEnabled": boolean,
   "tags": [
      {
         "key": "string",
         "value": "string"
      }
   ],
   "virtualGatewayId": "string",
   "virtualInterfaceId": "string",
   "virtualInterfaceName": "string",
   "virtualInterfaceState": "string",
   "virtualInterfaceType": "string",
   "vlan": number
}
```

Response Elements

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

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

addressFamily

The address family for the BGP peer.

Type: String

Valid Values: `ipv4 | ipv6`

amazonAddress

The IP address assigned to the Amazon interface.

Type: String

amazonSideAsn

The autonomous system number (AS) for the Amazon side of the connection.

Type: Long

asn

The autonomous system number (ASN). The valid range is from 1 to 2147483646 for Border Gateway Protocol (BGP) configuration. If you provide a number greater than the maximum, an error is returned. Use `asnLong` instead.

- You can use `asnLong` or `asn`, but not both. We recommend using `asnLong` as it supports a greater pool of numbers.
- If you provide a value in the same API call for both `asn` and `asnLong`, the API will only accept the value for `asnLong`.
- If you enter a 4-byte ASN for the `asn` parameter, the API returns an error.
- If you are using a 2-byte ASN, the API response will include the 2-byte value for both the `asn` and `asnLong` fields.

Type: Integer

asnLong

The long ASN for the virtual interface. The valid range is from 1 to 4294967294 for BGP configuration.

Note the following limitations when using `asnLong`:

- You can use `asnLong` or `asn`, but not both. We recommend using `asnLong` as it supports a greater pool of numbers.
- `asnLong` accepts any valid ASN value, regardless if it's 2-byte or 4-byte.
- When using a 4-byte `asnLong`, the API response returns `0` for the legacy `asn` attribute since 4-byte ASN values exceed the maximum supported value of 2,147,483,647.
- If you are using a 2-byte ASN, the API response will include the 2-byte value for both the `asn` and `asnLong` fields.

- If you provide a value in the same API call for both `asn` and `asnLong`, the API will only accept the value for `asnLong`.

Type: Long

authKey

The authentication key for BGP configuration. This string has a minimum length of 6 characters and and a maximun lenth of 80 characters.

Type: String

awsDeviceV2

The Direct Connect endpoint that terminates the physical connection.

Type: String

awsLogicalDeviceId

The Direct Connect endpoint that terminates the logical connection. This device might be different than the device that terminates the physical connection.

Type: String

bgpPeers

The BGP peers configured on this virtual interface.

Type: Array of BGPPeer objects

connectionId

The ID of the connection.

Type: String

customerAddress

The IP address assigned to the customer interface.

Type: String

customerRouterConfig

The customer router configuration.

Type: String

directConnectGatewayId

The ID of the Direct Connect gateway.

Type: String

jumboFrameCapable

Indicates whether jumbo frames are supported.

Type: Boolean

location

The location of the connection.

Type: String

mtu

The maximum transmission unit (MTU), in bytes. The supported values are 1500 and 8500. The default value is 1500

Type: Integer

ownerAccount

The ID of the AWS account that owns the virtual interface.

Type: String

region

The AWS Region where the virtual interface is located.

Type: String

routeFilterPrefixes

The routes to be advertised to the AWS network in this Region. Applies to public virtual interfaces.

Type: Array of RouteFilterPrefix objects

siteLinkEnabled

Indicates whether SiteLink is enabled.

Type: Boolean

tags

The tags associated with the virtual interface.

Type: Array of Tag objects

Array Members: Minimum number of 1 item.

virtualGatewayId

The ID of the virtual private gateway. Applies only to private virtual interfaces.

Type: String

virtualInterfaceId

The ID of the virtual interface.

Type: String

virtualInterfaceName

The name of the virtual interface assigned by the customer network. The name has a maximum of 100 characters. The following are valid characters: a-z, 0-9 and a hyphen (-).

Type: String

virtualInterfaceState

The state of the virtual interface. The following are the possible values:

- `confirming`: The creation of the virtual interface is pending confirmation from the virtual interface owner. If the owner of the virtual interface is different from the owner of the connection on which it is provisioned, then the virtual interface will remain in this state until it is confirmed by the virtual interface owner.
- `verifying`: This state only applies to public virtual interfaces. Each public virtual interface needs validation before the virtual interface can be created.
- `pending`: A virtual interface is in this state from the time that it is created until the virtual interface is ready to forward traffic.
- `available`: A virtual interface that is able to forward traffic.
- `down`: A virtual interface that is BGP down.

- `testing`: A virtual interface is in this state immediately after calling StartBgpFailoverTest and remains in this state during the duration of the test.
- `deleting`: A virtual interface is in this state immediately after calling DeleteVirtualInterface until it can no longer forward traffic.
- `deleted`: A virtual interface that cannot forward traffic.
- `rejected`: The virtual interface owner has declined creation of the virtual interface. If a virtual interface in the `Confirming` state is deleted by the virtual interface owner, the virtual interface enters the `Rejected` state.
- unknown: The state of the virtual interface is not available.

Type: String

Valid Values: `confirming | verifying | pending | available | down | testing | deleting | deleted | rejected | unknown`

virtualInterfaceType

The type of virtual interface. The possible values are `private`, `public` and `transit`.

Type: String

vlan

The ID of the VLAN.

Type: Integer

Errors

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

DirectConnectClientException

One or more parameters are not valid.

HTTP Status Code: 400

DirectConnectServerException

A server-side error occurred.

HTTP Status Code: 400

DuplicateTagKeysException

A tag key was specified more than once.

HTTP Status Code: 400

TooManyTagsException

You have reached the limit on the number of tags that can be assigned.

HTTP Status Code: 400

See Also

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

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

AllocatePublicVirtualInterface

Provisions a public virtual interface to be owned by the specified AWS account.

The owner of a connection calls this function to provision a public virtual interface to be owned by the specified AWS account.

Virtual interfaces created using this function must be confirmed by the owner using ConfirmPublicVirtualInterface. Until this step has been completed, the virtual interface is in the `confirming` state and is not available to handle traffic.

When creating an IPv6 public virtual interface, omit the Amazon address and customer address. IPv6 addresses are automatically assigned from the Amazon pool of IPv6 addresses; you cannot specify custom IPv6 addresses.

Request Syntax

```
{
   "connectionId": "string",
   "newPublicVirtualInterfaceAllocation": {
      "addressFamily": "string",
      "amazonAddress": "string",
      "asn": number,
      "asnLong": number,
      "authKey": "string",
      "customerAddress": "string",
      "routeFilterPrefixes": [
         {
            "cidr": "string"
         }
      ],
      "tags": [
         {
            "key": "string",
            "value": "string"
         }
      ],
      "virtualInterfaceName": "string",
      "vlan": number
   },
   "ownerAccount": "string"
}
```

Request Parameters

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

The request accepts the following data in JSON format.

connectionId

The ID of the connection on which the public virtual interface is provisioned.

Type: String

Required: Yes

newPublicVirtualInterfaceAllocation

Information about the public virtual interface.

Type: NewPublicVirtualInterfaceAllocation object

Required: Yes

ownerAccount

The ID of the AWS account that owns the public virtual interface.

Type: String

Required: Yes

Response Syntax

```
{
   "addressFamily": "string",
   "amazonAddress": "string",
   "amazonSideAsn": number,
   "asn": number,
   "asnLong": number,
   "authKey": "string",
   "awsDeviceV2": "string",
   "awsLogicalDeviceId": "string",
   "bgpPeers": [
      {
         "addressFamily": "string",
         "amazonAddress": "string",
```

```
            "asn": number,
            "asnLong": number,
            "authKey": "string",
            "awsDeviceV2": "string",
            "awsLogicalDeviceId": "string",
            "bgpPeerId": "string",
            "bgpPeerState": "string",
            "bgpStatus": "string",
            "customerAddress": "string"
         }
      ],
      "connectionId": "string",
      "customerAddress": "string",
      "customerRouterConfig": "string",
      "directConnectGatewayId": "string",
      "jumboFrameCapable": boolean,
      "location": "string",
      "mtu": number,
      "ownerAccount": "string",
      "region": "string",
      "routeFilterPrefixes": [
         {
            "cidr": "string"
         }
      ],
      "siteLinkEnabled": boolean,
      "tags": [
         {
            "key": "string",
            "value": "string"
         }
      ],
      "virtualGatewayId": "string",
      "virtualInterfaceId": "string",
      "virtualInterfaceName": "string",
      "virtualInterfaceState": "string",
      "virtualInterfaceType": "string",
      "vlan": number
}
```

Response Elements

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

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

addressFamily

The address family for the BGP peer.

Type: String

Valid Values: `ipv4 | ipv6`

amazonAddress

The IP address assigned to the Amazon interface.

Type: String

amazonSideAsn

The autonomous system number (AS) for the Amazon side of the connection.

Type: Long

asn

The autonomous system number (ASN). The valid range is from 1 to 2147483646 for Border Gateway Protocol (BGP) configuration. If you provide a number greater than the maximum, an error is returned. Use `asnLong` instead.

- You can use `asnLong` or `asn`, but not both. We recommend using `asnLong` as it supports a greater pool of numbers.
- If you provide a value in the same API call for both `asn` and `asnLong`, the API will only accept the value for `asnLong`.
- If you enter a 4-byte ASN for the `asn` parameter, the API returns an error.
- If you are using a 2-byte ASN, the API response will include the 2-byte value for both the `asn` and `asnLong` fields.

Type: Integer

asnLong

The long ASN for the virtual interface. The valid range is from 1 to 4294967294 for BGP configuration.

Note the following limitations when using `asnLong`:

- You can use asnLong or asn, but not both. We recommend using asnLong as it supports a greater pool of numbers.
- asnLong accepts any valid ASN value, regardless if it's 2-byte or 4-byte.
- When using a 4-byte asnLong, the API response returns 0 for the legacy asn attribute since 4-byte ASN values exceed the maximum supported value of 2,147,483,647.
- If you are using a 2-byte ASN, the API response will include the 2-byte value for both the asn and asnLong fields.
- If you provide a value in the same API call for both asn and asnLong, the API will only accept the value for asnLong.

Type: Long

authKey

The authentication key for BGP configuration. This string has a minimum length of 6 characters and and a maximun lenth of 80 characters.

Type: String

awsDeviceV2

The Direct Connect endpoint that terminates the physical connection.

Type: String

awsLogicalDeviceId

The Direct Connect endpoint that terminates the logical connection. This device might be different than the device that terminates the physical connection.

Type: String

bgpPeers

The BGP peers configured on this virtual interface.

Type: Array of BGPPeer objects

connectionId

The ID of the connection.

Type: String

customerAddress

The IP address assigned to the customer interface.

Type: String

customerRouterConfig

The customer router configuration.

Type: String

directConnectGatewayId

The ID of the Direct Connect gateway.

Type: String

jumboFrameCapable

Indicates whether jumbo frames are supported.

Type: Boolean

location

The location of the connection.

Type: String

mtu

The maximum transmission unit (MTU), in bytes. The supported values are 1500 and 8500. The default value is 1500

Type: Integer

ownerAccount

The ID of the AWS account that owns the virtual interface.

Type: String

region

The AWS Region where the virtual interface is located.

Type: String

routeFilterPrefixes

The routes to be advertised to the AWS network in this Region. Applies to public virtual interfaces.

Type: Array of RouteFilterPrefix objects

siteLinkEnabled

Indicates whether SiteLink is enabled.

Type: Boolean

tags

The tags associated with the virtual interface.

Type: Array of Tag objects

Array Members: Minimum number of 1 item.

virtualGatewayId

The ID of the virtual private gateway. Applies only to private virtual interfaces.

Type: String

virtualInterfaceId

The ID of the virtual interface.

Type: String

virtualInterfaceName

The name of the virtual interface assigned by the customer network. The name has a maximum of 100 characters. The following are valid characters: a-z, 0-9 and a hyphen (-).

Type: String

virtualInterfaceState

The state of the virtual interface. The following are the possible values:

- `confirming`: The creation of the virtual interface is pending confirmation from the virtual interface owner. If the owner of the virtual interface is different from the owner of the

connection on which it is provisioned, then the virtual interface will remain in this state until it is confirmed by the virtual interface owner.

- `verifying`: This state only applies to public virtual interfaces. Each public virtual interface needs validation before the virtual interface can be created.
- `pending`: A virtual interface is in this state from the time that it is created until the virtual interface is ready to forward traffic.
- `available`: A virtual interface that is able to forward traffic.
- down: A virtual interface that is BGP down.
- `testing`: A virtual interface is in this state immediately after calling StartBgpFailoverTest and remains in this state during the duration of the test.
- `deleting`: A virtual interface is in this state immediately after calling DeleteVirtualInterface until it can no longer forward traffic.
- `deleted`: A virtual interface that cannot forward traffic.
- `rejected`: The virtual interface owner has declined creation of the virtual interface. If a virtual interface in the `Confirming` state is deleted by the virtual interface owner, the virtual interface enters the `Rejected` state.
- unknown: The state of the virtual interface is not available.

Type: String

Valid Values: `confirming | verifying | pending | available | down | testing | deleting | deleted | rejected | unknown`

virtualInterfaceType

The type of virtual interface. The possible values are `private`, `public` and `transit`.

Type: String

vlan

The ID of the VLAN.

Type: Integer

Errors

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

DirectConnectClientException

One or more parameters are not valid.

HTTP Status Code: 400

DirectConnectServerException

A server-side error occurred.

HTTP Status Code: 400

DuplicateTagKeysException

A tag key was specified more than once.

HTTP Status Code: 400

TooManyTagsException

You have reached the limit on the number of tags that can be assigned.

HTTP Status Code: 400

See Also

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

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

AllocateTransitVirtualInterface

Provisions a transit virtual interface to be owned by the specified AWS account. Use this type of interface to connect a transit gateway to your Direct Connect gateway.

The owner of a connection provisions a transit virtual interface to be owned by the specified AWS account.

After you create a transit virtual interface, it must be confirmed by the owner using ConfirmTransitVirtualInterface. Until this step has been completed, the transit virtual interface is in the `requested` state and is not available to handle traffic.

Request Syntax

```
{
   "connectionId": "string",
   "newTransitVirtualInterfaceAllocation": {
      "addressFamily": "string",
      "amazonAddress": "string",
      "asn": number,
      "asnLong": number,
      "authKey": "string",
      "customerAddress": "string",
      "mtu": number,
      "tags": [
         {
            "key": "string",
            "value": "string"
         }
      ],
      "virtualInterfaceName": "string",
      "vlan": number
   },
   "ownerAccount": "string"
}
```

Request Parameters

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

The request accepts the following data in JSON format.

connectionId

The ID of the connection on which the transit virtual interface is provisioned.

Type: String

Required: Yes

newTransitVirtualInterfaceAllocation

Information about the transit virtual interface.

Type: NewTransitVirtualInterfaceAllocation object

Required: Yes

ownerAccount

The ID of the AWS account that owns the transit virtual interface.

Type: String

Required: Yes

Response Syntax

```
{
   "virtualInterface": {
      "addressFamily": "string",
      "amazonAddress": "string",
      "amazonSideAsn": number,
      "asn": number,
      "asnLong": number,
      "authKey": "string",
      "awsDeviceV2": "string",
      "awsLogicalDeviceId": "string",
      "bgpPeers": [
         {
            "addressFamily": "string",
            "amazonAddress": "string",
            "asn": number,
            "asnLong": number,
            "authKey": "string",
```

```
            "awsDeviceV2": "string",
            "awsLogicalDeviceId": "string",
            "bgpPeerId": "string",
            "bgpPeerState": "string",
            "bgpStatus": "string",
            "customerAddress": "string"
         }
      ],
      "connectionId": "string",
      "customerAddress": "string",
      "customerRouterConfig": "string",
      "directConnectGatewayId": "string",
      "jumboFrameCapable": boolean,
      "location": "string",
      "mtu": number,
      "ownerAccount": "string",
      "region": "string",
      "routeFilterPrefixes": [
         {
            "cidr": "string"
         }
      ],
      "siteLinkEnabled": boolean,
      "tags": [
         {
            "key": "string",
            "value": "string"
         }
      ],
      "virtualGatewayId": "string",
      "virtualInterfaceId": "string",
      "virtualInterfaceName": "string",
      "virtualInterfaceState": "string",
      "virtualInterfaceType": "string",
      "vlan": number
   }
}
```

Response Elements

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

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

virtualInterface

Information about the transit virtual interface.

Type: VirtualInterface object

Errors

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

DirectConnectClientException

One or more parameters are not valid.

HTTP Status Code: 400

DirectConnectServerException

A server-side error occurred.

HTTP Status Code: 400

DuplicateTagKeysException

A tag key was specified more than once.

HTTP Status Code: 400

TooManyTagsException

You have reached the limit on the number of tags that can be assigned.

HTTP Status Code: 400

See Also

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

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

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

AssociateConnectionWithLag

Associates an existing connection with a link aggregation group (LAG). The connection is interrupted and re-established as a member of the LAG (connectivity to AWS is interrupted). The connection must be hosted on the same Direct Connect endpoint as the LAG, and its bandwidth must match the bandwidth for the LAG. You can re-associate a connection that's currently associated with a different LAG; however, if removing the connection would cause the original LAG to fall below its setting for minimum number of operational connections, the request fails.

Any virtual interfaces that are directly associated with the connection are automatically re-associated with the LAG. If the connection was originally associated with a different LAG, the virtual interfaces remain associated with the original LAG.

For interconnects, any hosted connections are automatically re-associated with the LAG. If the interconnect was originally associated with a different LAG, the hosted connections remain associated with the original LAG.

Request Syntax

```
{
   "connectionId": "string",
   "lagId": "string"
}
```

Request Parameters

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

The request accepts the following data in JSON format.

connectionId

The ID of the connection.

Type: String

Required: Yes

lagId

The ID of the LAG with which to associate the connection.

Type: String

Required: Yes

Response Syntax

```
{
   "awsDevice": "string",
   "awsDeviceV2": "string",
   "awsLogicalDeviceId": "string",
   "bandwidth": "string",
   "connectionId": "string",
   "connectionName": "string",
   "connectionState": "string",
   "encryptionMode": "string",
   "hasLogicalRedundancy": "string",
   "jumboFrameCapable": boolean,
   "lagId": "string",
   "loaIssueTime": number,
   "location": "string",
   "macSecCapable": boolean,
   "macSecKeys": [
      {
         "ckn": "string",
         "secretARN": "string",
         "startOn": "string",
         "state": "string"
      }
   ],
   "ownerAccount": "string",
   "partnerInterconnectMacSecCapable": boolean,
   "partnerName": "string",
   "portEncryptionStatus": "string",
   "providerName": "string",
   "region": "string",
   "tags": [
      {
         "key": "string",
         "value": "string"
      }
   ],
   "vlan": number
```

```
}
```

Response Elements

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

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

awsDevice

This parameter has been deprecated.

The Direct Connect endpoint on which the physical connection terminates.

Type: String

awsDeviceV2

The Direct Connect endpoint that terminates the physical connection.

Type: String

awsLogicalDeviceId

The Direct Connect endpoint that terminates the logical connection. This device might be different than the device that terminates the physical connection.

Type: String

bandwidth

The bandwidth of the connection.

Type: String

connectionId

The ID of the connection.

Type: String

connectionName

The name of the connection.

Type: String

connectionState

The state of the connection. The following are the possible values:

- `ordering`: The initial state of a hosted connection provisioned on an interconnect. The connection stays in the ordering state until the owner of the hosted connection confirms or declines the connection order.
- `requested`: The initial state of a standard connection. The connection stays in the requested state until the Letter of Authorization (LOA) is sent to the customer.
- `pending`: The connection has been approved and is being initialized.
- `available`: The network link is up and the connection is ready for use.
- down: The network link is down.
- `deleting`: The connection is being deleted.
- `deleted`: The connection has been deleted.
- `rejected`: A hosted connection in the `ordering` state enters the `rejected` state if it is deleted by the customer.
- unknown: The state of the connection is not available.

Type: String

Valid Values: `ordering | requested | pending | available | down | deleting | deleted | rejected | unknown`

encryptionMode

The MAC Security (MACsec) connection encryption mode.

The valid values are `no_encrypt`, `should_encrypt`, and `must_encrypt`.

Type: String

hasLogicalRedundancy

Indicates whether the connection supports a secondary BGP peer in the same address family (IPv4/IPv6).

Type: String

Valid Values: unknown | `yes` | no

jumboFrameCapable

Indicates whether jumbo frames are supported.

Type: Boolean

lagId

The ID of the LAG.

Type: String

loaIssueTime

The time of the most recent call to DescribeLoa for this connection.

Type: Timestamp

location

The location of the connection.

Type: String

macSecCapable

Indicates whether the connection supports MAC Security (MACsec).

Type: Boolean

macSecKeys

The MAC Security (MACsec) security keys associated with the connection.

Type: Array of MacSecKey objects

ownerAccount

The ID of the AWS account that owns the connection.

Type: String

partnerInterconnectMacSecCapable

Indicates whether the interconnect hosting this connection supports MAC Security (MACsec).

Type: Boolean

partnerName

The name of the Direct Connect service provider associated with the connection.

Type: String

portEncryptionStatus

The MAC Security (MACsec) port link status of the connection.

The valid values are `Encryption Up`, which means that there is an active Connection Key Name, or `Encryption Down`.

Type: String

providerName

The name of the service provider associated with the connection.

Type: String

region

The AWS Region where the connection is located.

Type: String

tags

The tags associated with the connection.

Type: Array of Tag objects

Array Members: Minimum number of 1 item.

vlan

The ID of the VLAN.

Type: Integer

Errors

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

DirectConnectClientException

One or more parameters are not valid.

HTTP Status Code: 400

DirectConnectServerException

A server-side error occurred.

HTTP Status Code: 400

See Also

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

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

AssociateHostedConnection

Associates a hosted connection and its virtual interfaces with a link aggregation group (LAG) or interconnect. If the target interconnect or LAG has an existing hosted connection with a conflicting VLAN number or IP address, the operation fails. This action temporarily interrupts the hosted connection's connectivity to AWS as it is being migrated.

> **Note**
>
> Intended for use by Direct Connect Partners only.

Request Syntax

```
{
   "connectionId": "string",
   "parentConnectionId": "string"
}
```

Request Parameters

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

The request accepts the following data in JSON format.

connectionId

The ID of the hosted connection.

Type: String

Required: Yes

parentConnectionId

The ID of the interconnect or the LAG.

Type: String

Required: Yes

Response Syntax

```
{
   "awsDevice": "string",
   "awsDeviceV2": "string",
   "awsLogicalDeviceId": "string",
   "bandwidth": "string",
   "connectionId": "string",
   "connectionName": "string",
   "connectionState": "string",
   "encryptionMode": "string",
   "hasLogicalRedundancy": "string",
   "jumboFrameCapable": boolean,
   "lagId": "string",
   "loaIssueTime": number,
   "location": "string",
   "macSecCapable": boolean,
   "macSecKeys": [
      {
         "ckn": "string",
         "secretARN": "string",
         "startOn": "string",
         "state": "string"
      }
   ],
   "ownerAccount": "string",
   "partnerInterconnectMacSecCapable": boolean,
   "partnerName": "string",
   "portEncryptionStatus": "string",
   "providerName": "string",
   "region": "string",
   "tags": [
      {
         "key": "string",
         "value": "string"
      }
   ],
   "vlan": number
}
```

Response Elements

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

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

awsDevice

This parameter has been deprecated.

The Direct Connect endpoint on which the physical connection terminates.

Type: String

awsDeviceV2

The Direct Connect endpoint that terminates the physical connection.

Type: String

awsLogicalDeviceId

The Direct Connect endpoint that terminates the logical connection. This device might be different than the device that terminates the physical connection.

Type: String

bandwidth

The bandwidth of the connection.

Type: String

connectionId

The ID of the connection.

Type: String

connectionName

The name of the connection.

Type: String

connectionState

The state of the connection. The following are the possible values:

- `ordering`: The initial state of a hosted connection provisioned on an interconnect. The connection stays in the ordering state until the owner of the hosted connection confirms or declines the connection order.

- `requested`: The initial state of a standard connection. The connection stays in the requested state until the Letter of Authorization (LOA) is sent to the customer.
- `pending`: The connection has been approved and is being initialized.
- `available`: The network link is up and the connection is ready for use.
- down: The network link is down.
- `deleting`: The connection is being deleted.
- `deleted`: The connection has been deleted.
- `rejected`: A hosted connection in the `ordering` state enters the `rejected` state if it is deleted by the customer.
- unknown: The state of the connection is not available.

Type: String

Valid Values: `ordering | requested | pending | available | down | deleting | deleted | rejected | unknown`

encryptionMode

The MAC Security (MACsec) connection encryption mode.

The valid values are `no_encrypt`, `should_encrypt`, and `must_encrypt`.

Type: String

hasLogicalRedundancy

Indicates whether the connection supports a secondary BGP peer in the same address family (IPv4/IPv6).

Type: String

Valid Values: `unknown | yes | no`

jumboFrameCapable

Indicates whether jumbo frames are supported.

Type: Boolean

lagId

The ID of the LAG.

Type: String

loaIssueTime

The time of the most recent call to DescribeLoa for this connection.

Type: Timestamp

location

The location of the connection.

Type: String

macSecCapable

Indicates whether the connection supports MAC Security (MACsec).

Type: Boolean

macSecKeys

The MAC Security (MACsec) security keys associated with the connection.

Type: Array of MacSecKey objects

ownerAccount

The ID of the AWS account that owns the connection.

Type: String

partnerInterconnectMacSecCapable

Indicates whether the interconnect hosting this connection supports MAC Security (MACsec).

Type: Boolean

partnerName

The name of the Direct Connect service provider associated with the connection.

Type: String

portEncryptionStatus

The MAC Security (MACsec) port link status of the connection.

The valid values are `Encryption Up`, which means that there is an active Connection Key Name, or `Encryption Down`.

Type: String

providerName

The name of the service provider associated with the connection.

Type: String

region

The AWS Region where the connection is located.

Type: String

tags

The tags associated with the connection.

Type: Array of Tag objects

Array Members: Minimum number of 1 item.

vlan

The ID of the VLAN.

Type: Integer

Errors

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

DirectConnectClientException

One or more parameters are not valid.

HTTP Status Code: 400

DirectConnectServerException

A server-side error occurred.

HTTP Status Code: 400

See Also

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

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

AssociateMacSecKey

Associates a MAC Security (MACsec) Connection Key Name (CKN)/ Connectivity Association Key (CAK) pair with a Direct Connect connection.

You must supply either the `secretARN`, or the CKN/CAK (`ckn` and `cak`) pair in the request.

For information about MAC Security (MACsec) key considerations, see MACsec pre-shared CKN/CAK key considerations in the *Direct Connect User Guide*.

Request Syntax

```
{
   "cak": "string",
   "ckn": "string",
   "connectionId": "string",
   "secretARN": "string"
}
```

Request Parameters

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

The request accepts the following data in JSON format.

cak

The MAC Security (MACsec) CAK to associate with the connection.

You can create the CKN/CAK pair using an industry standard tool.

The valid values are 64 hexadecimal characters (0-9, A-E).

If you use this request parameter, you must use the `ckn` request parameter and not use the `secretARN` request parameter.

Type: String

Required: No

ckn

The MAC Security (MACsec) CKN to associate with the connection.

You can create the CKN/CAK pair using an industry standard tool.

The valid values are 64 hexadecimal characters (0-9, A-E).

If you use this request parameter, you must use the `cak` request parameter and not use the `secretARN` request parameter.

Type: String

Required: No

connectionId

The ID of the dedicated connection (dxcon-xxxx), interconnect (dxcon-xxxx), or LAG (dxlag-xxxx).

You can use DescribeConnections, DescribeInterconnects, or DescribeLags to retrieve connection ID.

Type: String

Required: Yes

secretARN

The Amazon Resource Name (ARN) of the MAC Security (MACsec) secret key to associate with the connection.

You can use DescribeConnections or DescribeLags to retrieve the MAC Security (MACsec) secret key.

If you use this request parameter, you do not use the `ckn` and `cak` request parameters.

Type: String

Required: No

Response Syntax

```
{
   "connectionId": "string",
   "macSecKeys": [
```

```
      {
         "ckn": "string",
         "secretARN": "string",
         "startOn": "string",
         "state": "string"
      }
   ]
}
```

Response Elements

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

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

connectionId

The ID of the dedicated connection (dxcon-xxxx), interconnect (dxcon-xxxx), or LAG (dxlag-xxxx).

Type: String

macSecKeys

The MAC Security (MACsec) security keys associated with the connection.

Type: Array of MacSecKey objects

Errors

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

DirectConnectClientException

One or more parameters are not valid.

HTTP Status Code: 400

DirectConnectServerException

A server-side error occurred.

HTTP Status Code: 400

See Also

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

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

AssociateVirtualInterface

Associates a virtual interface with a specified link aggregation group (LAG) or connection. Connectivity to AWS is temporarily interrupted as the virtual interface is being migrated. If the target connection or LAG has an associated virtual interface with a conflicting VLAN number or a conflicting IP address, the operation fails.

Virtual interfaces associated with a hosted connection cannot be associated with a LAG; hosted connections must be migrated along with their virtual interfaces using AssociateHostedConnection.

To reassociate a virtual interface to a new connection or LAG, the requester must own either the virtual interface itself or the connection to which the virtual interface is currently associated. Additionally, the requester must own the connection or LAG for the association.

Request Syntax

```
{
   "connectionId": "string",
   "virtualInterfaceId": "string"
}
```

Request Parameters

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

The request accepts the following data in JSON format.

connectionId

The ID of the LAG or connection.

Type: String

Required: Yes

virtualInterfaceId

The ID of the virtual interface.

Type: String

Required: Yes

Response Syntax

```
{
   "addressFamily": "string",
   "amazonAddress": "string",
   "amazonSideAsn": number,
   "asn": number,
   "asnLong": number,
   "authKey": "string",
   "awsDeviceV2": "string",
   "awsLogicalDeviceId": "string",
   "bgpPeers": [
      {
         "addressFamily": "string",
         "amazonAddress": "string",
         "asn": number,
         "asnLong": number,
         "authKey": "string",
         "awsDeviceV2": "string",
         "awsLogicalDeviceId": "string",
         "bgpPeerId": "string",
         "bgpPeerState": "string",
         "bgpStatus": "string",
         "customerAddress": "string"
      }
   ],
   "connectionId": "string",
   "customerAddress": "string",
   "customerRouterConfig": "string",
   "directConnectGatewayId": "string",
   "jumboFrameCapable": boolean,
   "location": "string",
   "mtu": number,
   "ownerAccount": "string",
   "region": "string",
   "routeFilterPrefixes": [
      {
         "cidr": "string"
      }
   ],
   "siteLinkEnabled": boolean,
   "tags": [
      {
```

```
            "key": "string",
            "value": "string"
        }
    ],
    "virtualGatewayId": "string",
    "virtualInterfaceId": "string",
    "virtualInterfaceName": "string",
    "virtualInterfaceState": "string",
    "virtualInterfaceType": "string",
    "vlan": number
}
```

Response Elements

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

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

addressFamily

The address family for the BGP peer.

Type: String

Valid Values: `ipv4 | ipv6`

amazonAddress

The IP address assigned to the Amazon interface.

Type: String

amazonSideAsn

The autonomous system number (AS) for the Amazon side of the connection.

Type: Long

asn

The autonomous system number (ASN). The valid range is from 1 to 2147483646 for Border Gateway Protocol (BGP) configuration. If you provide a number greater than the maximum, an error is returned. Use `asnLong` instead.

- You can use asnLong or asn, but not both. We recommend using asnLong as it supports a greater pool of numbers.
- If you provide a value in the same API call for both asn and asnLong, the API will only accept the value for asnLong.
- If you enter a 4-byte ASN for the asn parameter, the API returns an error.
- If you are using a 2-byte ASN, the API response will include the 2-byte value for both the asn and asnLong fields.

Type: Integer

asnLong

The long ASN for the virtual interface. The valid range is from 1 to 4294967294 for BGP configuration.

Note the following limitations when using asnLong:

- You can use asnLong or asn, but not both. We recommend using asnLong as it supports a greater pool of numbers.
- asnLong accepts any valid ASN value, regardless if it's 2-byte or 4-byte.
- When using a 4-byte asnLong, the API response returns 0 for the legacy asn attribute since 4-byte ASN values exceed the maximum supported value of 2,147,483,647.
- If you are using a 2-byte ASN, the API response will include the 2-byte value for both the asn and asnLong fields.
- If you provide a value in the same API call for both asn and asnLong, the API will only accept the value for asnLong.

Type: Long

authKey

The authentication key for BGP configuration. This string has a minimum length of 6 characters and and a maximun lenth of 80 characters.

Type: String

awsDeviceV2

The Direct Connect endpoint that terminates the physical connection.

Type: String

awsLogicalDeviceId

The Direct Connect endpoint that terminates the logical connection. This device might be different than the device that terminates the physical connection.

Type: String

bgpPeers

The BGP peers configured on this virtual interface.

Type: Array of BGPPeer objects

connectionId

The ID of the connection.

Type: String

customerAddress

The IP address assigned to the customer interface.

Type: String

customerRouterConfig

The customer router configuration.

Type: String

directConnectGatewayId

The ID of the Direct Connect gateway.

Type: String

jumboFrameCapable

Indicates whether jumbo frames are supported.

Type: Boolean

location

The location of the connection.

Type: String

mtu

The maximum transmission unit (MTU), in bytes. The supported values are 1500 and 8500. The default value is 1500

Type: Integer

ownerAccount

The ID of the AWS account that owns the virtual interface.

Type: String

region

The AWS Region where the virtual interface is located.

Type: String

routeFilterPrefixes

The routes to be advertised to the AWS network in this Region. Applies to public virtual interfaces.

Type: Array of RouteFilterPrefix objects

siteLinkEnabled

Indicates whether SiteLink is enabled.

Type: Boolean

tags

The tags associated with the virtual interface.

Type: Array of Tag objects

Array Members: Minimum number of 1 item.

virtualGatewayId

The ID of the virtual private gateway. Applies only to private virtual interfaces.

Type: String

virtualInterfaceId

The ID of the virtual interface.

Type: String

virtualInterfaceName

The name of the virtual interface assigned by the customer network. The name has a maximum of 100 characters. The following are valid characters: a-z, 0-9 and a hyphen (-).

Type: String

virtualInterfaceState

The state of the virtual interface. The following are the possible values:

- `confirming`: The creation of the virtual interface is pending confirmation from the virtual interface owner. If the owner of the virtual interface is different from the owner of the connection on which it is provisioned, then the virtual interface will remain in this state until it is confirmed by the virtual interface owner.
- `verifying`: This state only applies to public virtual interfaces. Each public virtual interface needs validation before the virtual interface can be created.
- `pending`: A virtual interface is in this state from the time that it is created until the virtual interface is ready to forward traffic.
- `available`: A virtual interface that is able to forward traffic.
- down: A virtual interface that is BGP down.
- `testing`: A virtual interface is in this state immediately after calling StartBgpFailoverTest and remains in this state during the duration of the test.
- `deleting`: A virtual interface is in this state immediately after calling DeleteVirtualInterface until it can no longer forward traffic.
- `deleted`: A virtual interface that cannot forward traffic.
- `rejected`: The virtual interface owner has declined creation of the virtual interface. If a virtual interface in the `Confirming` state is deleted by the virtual interface owner, the virtual interface enters the `Rejected` state.
- unknown: The state of the virtual interface is not available.

Type: String

Valid Values: `confirming | verifying | pending | available | down | testing | deleting | deleted | rejected | unknown`

virtualInterfaceType

The type of virtual interface. The possible values are `private`, `public` and `transit`.

Type: String

vlan

The ID of the VLAN.

Type: Integer

Errors

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

DirectConnectClientException

One or more parameters are not valid.

HTTP Status Code: 400

DirectConnectServerException

A server-side error occurred.

HTTP Status Code: 400

See Also

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

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

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

ConfirmConnection

Confirms the creation of the specified hosted connection on an interconnect.

Upon creation, the hosted connection is initially in the `Ordering` state, and remains in this state until the owner confirms creation of the hosted connection.

Request Syntax

```
{
   "connectionId": "string"
}
```

Request Parameters

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

The request accepts the following data in JSON format.

connectionId

The ID of the hosted connection.

Type: String

Required: Yes

Response Syntax

```
{
   "connectionState": "string"
}
```

Response Elements

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

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

connectionState

The state of the connection. The following are the possible values:

- `ordering`: The initial state of a hosted connection provisioned on an interconnect. The connection stays in the ordering state until the owner of the hosted connection confirms or declines the connection order.
- `requested`: The initial state of a standard connection. The connection stays in the requested state until the Letter of Authorization (LOA) is sent to the customer.
- `pending`: The connection has been approved and is being initialized.
- `available`: The network link is up and the connection is ready for use.
- down: The network link is down.
- `deleting`: The connection is being deleted.
- `deleted`: The connection has been deleted.
- `rejected`: A hosted connection in the `ordering` state enters the `rejected` state if it is deleted by the customer.
- unknown: The state of the connection is not available.

Type: String

Valid Values: `ordering | requested | pending | available | down | deleting | deleted | rejected | unknown`

Errors

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

DirectConnectClientException

One or more parameters are not valid.

HTTP Status Code: 400

DirectConnectServerException

A server-side error occurred.

HTTP Status Code: 400

See Also

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

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

ConfirmCustomerAgreement

The confirmation of the terms of agreement when creating the connection/link aggregation group (LAG).

Request Syntax

```
{
   "agreementName": "string"
}
```

Request Parameters

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

The request accepts the following data in JSON format.

agreementName

The name of the customer agreement.

Type: String

Length Constraints: Maximum length of 100.

Required: No

Response Syntax

```
{
   "status": "string"
}
```

Response Elements

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

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

status

The status of the customer agreement when the connection was created. This will be either `signed` or `unsigned`.

Type: String

Length Constraints: Maximum length of 30.

Errors

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

DirectConnectClientException

One or more parameters are not valid.

HTTP Status Code: 400

DirectConnectServerException

A server-side error occurred.

HTTP Status Code: 400

See Also

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

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

- AWS SDK for Python
- AWS SDK for Ruby V3

ConfirmPrivateVirtualInterface

Accepts ownership of a private virtual interface created by another AWS account.

After the virtual interface owner makes this call, the virtual interface is created and attached to the specified virtual private gateway or Direct Connect gateway, and is made available to handle traffic.

Request Syntax

```
{
   "directConnectGatewayId": "string",
   "virtualGatewayId": "string",
   "virtualInterfaceId": "string"
}
```

Request Parameters

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

The request accepts the following data in JSON format.

directConnectGatewayId

The ID of the Direct Connect gateway.

Type: String

Required: No

virtualGatewayId

The ID of the virtual private gateway.

Type: String

Required: No

virtualInterfaceId

The ID of the virtual interface.

Type: String

Required: Yes

Response Syntax

```
{
   "virtualInterfaceState": "string"
}
```

Response Elements

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

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

virtualInterfaceState

The state of the virtual interface. The following are the possible values:

- `confirming`: The creation of the virtual interface is pending confirmation from the virtual interface owner. If the owner of the virtual interface is different from the owner of the connection on which it is provisioned, then the virtual interface will remain in this state until it is confirmed by the virtual interface owner.
- `verifying`: This state only applies to public virtual interfaces. Each public virtual interface needs validation before the virtual interface can be created.
- `pending`: A virtual interface is in this state from the time that it is created until the virtual interface is ready to forward traffic.
- `available`: A virtual interface that is able to forward traffic.
- down: A virtual interface that is BGP down.
- `testing`: A virtual interface is in this state immediately after calling StartBgpFailoverTest and remains in this state during the duration of the test.
- `deleting`: A virtual interface is in this state immediately after calling DeleteVirtualInterface until it can no longer forward traffic.
- `deleted`: A virtual interface that cannot forward traffic.
- `rejected`: The virtual interface owner has declined creation of the virtual interface. If a virtual interface in the `Confirming` state is deleted by the virtual interface owner, the virtual interface enters the `Rejected` state.

- unknown: The state of the virtual interface is not available.

Type: String

Valid Values: `confirming | verifying | pending | available | down | testing | deleting | deleted | rejected | unknown`

Errors

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

DirectConnectClientException

One or more parameters are not valid.

HTTP Status Code: 400

DirectConnectServerException

A server-side error occurred.

HTTP Status Code: 400

See Also

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

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

ConfirmPublicVirtualInterface

Accepts ownership of a public virtual interface created by another AWS account.

After the virtual interface owner makes this call, the specified virtual interface is created and made available to handle traffic.

Request Syntax

```
{
   "virtualInterfaceId": "string"
}
```

Request Parameters

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

The request accepts the following data in JSON format.

virtualInterfaceId

The ID of the virtual interface.

Type: String

Required: Yes

Response Syntax

```
{
   "virtualInterfaceState": "string"
}
```

Response Elements

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

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

virtualInterfaceState

The state of the virtual interface. The following are the possible values:

- `confirming`: The creation of the virtual interface is pending confirmation from the virtual interface owner. If the owner of the virtual interface is different from the owner of the connection on which it is provisioned, then the virtual interface will remain in this state until it is confirmed by the virtual interface owner.
- `verifying`: This state only applies to public virtual interfaces. Each public virtual interface needs validation before the virtual interface can be created.
- `pending`: A virtual interface is in this state from the time that it is created until the virtual interface is ready to forward traffic.
- `available`: A virtual interface that is able to forward traffic.
- down: A virtual interface that is BGP down.
- `testing`: A virtual interface is in this state immediately after calling StartBgpFailoverTest and remains in this state during the duration of the test.
- `deleting`: A virtual interface is in this state immediately after calling DeleteVirtualInterface until it can no longer forward traffic.
- `deleted`: A virtual interface that cannot forward traffic.
- `rejected`: The virtual interface owner has declined creation of the virtual interface. If a virtual interface in the `Confirming` state is deleted by the virtual interface owner, the virtual interface enters the `Rejected` state.
- unknown: The state of the virtual interface is not available.

Type: String

Valid Values: `confirming | verifying | pending | available | down | testing | deleting | deleted | rejected | unknown`

Errors

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

DirectConnectClientException

One or more parameters are not valid.

HTTP Status Code: 400

DirectConnectServerException

A server-side error occurred.

HTTP Status Code: 400

See Also

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

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

ConfirmTransitVirtualInterface

Accepts ownership of a transit virtual interface created by another AWS account.

After the owner of the transit virtual interface makes this call, the specified transit virtual interface is created and made available to handle traffic.

Request Syntax

```
{
   "directConnectGatewayId": "string",
   "virtualInterfaceId": "string"
}
```

Request Parameters

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

The request accepts the following data in JSON format.

directConnectGatewayId

The ID of the Direct Connect gateway.

Type: String

Required: Yes

virtualInterfaceId

The ID of the virtual interface.

Type: String

Required: Yes

Response Syntax

```
{
   "virtualInterfaceState": "string"
```

```
}
```

Response Elements

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

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

virtualInterfaceState

The state of the virtual interface. The following are the possible values:

- `confirming`: The creation of the virtual interface is pending confirmation from the virtual interface owner. If the owner of the virtual interface is different from the owner of the connection on which it is provisioned, then the virtual interface will remain in this state until it is confirmed by the virtual interface owner.
- `verifying`: This state only applies to public virtual interfaces. Each public virtual interface needs validation before the virtual interface can be created.
- `pending`: A virtual interface is in this state from the time that it is created until the virtual interface is ready to forward traffic.
- `available`: A virtual interface that is able to forward traffic.
- down: A virtual interface that is BGP down.
- `testing`: A virtual interface is in this state immediately after calling StartBgpFailoverTest and remains in this state during the duration of the test.
- `deleting`: A virtual interface is in this state immediately after calling DeleteVirtualInterface until it can no longer forward traffic.
- `deleted`: A virtual interface that cannot forward traffic.
- `rejected`: The virtual interface owner has declined creation of the virtual interface. If a virtual interface in the `Confirming` state is deleted by the virtual interface owner, the virtual interface enters the `Rejected` state.
- unknown: The state of the virtual interface is not available.

Type: String

Valid Values: `confirming | verifying | pending | available | down | testing | deleting | deleted | rejected | unknown`

Errors

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

DirectConnectClientException

One or more parameters are not valid.

HTTP Status Code: 400

DirectConnectServerException

A server-side error occurred.

HTTP Status Code: 400

See Also

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

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

CreateBGPPeer

Creates a BGP peer on the specified virtual interface.

You must create a BGP peer for the corresponding address family (IPv4/IPv6) in order to access AWS resources that also use that address family.

If logical redundancy is not supported by the connection, interconnect, or LAG, the BGP peer cannot be in the same address family as an existing BGP peer on the virtual interface.

When creating a IPv6 BGP peer, omit the Amazon address and customer address. IPv6 addresses are automatically assigned from the Amazon pool of IPv6 addresses; you cannot specify custom IPv6 addresses.

> **Important**
>
> If you let AWS auto-assign IPv4 addresses, a /30 CIDR will be allocated from 169.254.0.0/16. AWS does not recommend this option if you intend to use the customer router peer IP address as the source and destination for traffic. Instead you should use RFC 1918 or other addressing, and specify the address yourself. For more information about RFC 1918 see Address Allocation for Private Internets.

For a public virtual interface, the Autonomous System Number (ASN) must be private or already on the allow list for the virtual interface.

Request Syntax

```
{
   "newBGPPeer": {
      "addressFamily": "string",
      "amazonAddress": "string",
      "asn": number,
      "asnLong": number,
      "authKey": "string",
      "customerAddress": "string"
   },
   "virtualInterfaceId": "string"
}
```

Request Parameters

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

The request accepts the following data in JSON format.

newBGPPeer

Information about the BGP peer.

Type: NewBGPPeer object

Required: No

virtualInterfaceId

The ID of the virtual interface.

Type: String

Required: No

Response Syntax

```
{
   "virtualInterface": {
      "addressFamily": "string",
      "amazonAddress": "string",
      "amazonSideAsn": number,
      "asn": number,
      "asnLong": number,
      "authKey": "string",
      "awsDeviceV2": "string",
      "awsLogicalDeviceId": "string",
      "bgpPeers": [
         {
            "addressFamily": "string",
            "amazonAddress": "string",
            "asn": number,
            "asnLong": number,
            "authKey": "string",
            "awsDeviceV2": "string",
            "awsLogicalDeviceId": "string",
            "bgpPeerId": "string",
```

```
            "bgpPeerState": "string",
            "bgpStatus": "string",
            "customerAddress": "string"
         }
      ],
      "connectionId": "string",
      "customerAddress": "string",
      "customerRouterConfig": "string",
      "directConnectGatewayId": "string",
      "jumboFrameCapable": boolean,
      "location": "string",
      "mtu": number,
      "ownerAccount": "string",
      "region": "string",
      "routeFilterPrefixes": [
         {
            "cidr": "string"
         }
      ],
      "siteLinkEnabled": boolean,
      "tags": [
         {
            "key": "string",
            "value": "string"
         }
      ],
      "virtualGatewayId": "string",
      "virtualInterfaceId": "string",
      "virtualInterfaceName": "string",
      "virtualInterfaceState": "string",
      "virtualInterfaceType": "string",
      "vlan": number
   }
}
```

Response Elements

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

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

virtualInterface

The virtual interface.

Type: VirtualInterface object

Errors

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

DirectConnectClientException

One or more parameters are not valid.

HTTP Status Code: 400

DirectConnectServerException

A server-side error occurred.

HTTP Status Code: 400

See Also

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

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

CreateConnection

Creates a connection between a customer network and a specific Direct Connect location.

A connection links your internal network to an Direct Connect location over a standard Ethernet fiber-optic cable. One end of the cable is connected to your router, the other to an Direct Connect router.

To find the locations for your Region, use DescribeLocations.

You can automatically add the new connection to a link aggregation group (LAG) by specifying a LAG ID in the request. This ensures that the new connection is allocated on the same Direct Connect endpoint that hosts the specified LAG. If there are no available ports on the endpoint, the request fails and no connection is created.

Request Syntax

```
{
   "bandwidth": "string",
   "connectionName": "string",
   "lagId": "string",
   "location": "string",
   "providerName": "string",
   "requestMACSec": boolean,
   "tags": [
      {
         "key": "string",
         "value": "string"
      }
   ]
}
```

Request Parameters

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

The request accepts the following data in JSON format.

bandwidth

The bandwidth of the connection.

Type: String

Required: Yes

connectionName

The name of the connection.

Type: String

Required: Yes

lagId

The ID of the LAG.

Type: String

Required: No

location

The location of the connection.

Type: String

Required: Yes

providerName

The name of the service provider associated with the requested connection.

Type: String

Required: No

requestMACSec

Indicates whether you want the connection to support MAC Security (MACsec).

MAC Security (MACsec) is unavailable on hosted connections. For information about MAC Security (MACsec) prerequisites, see MAC Security in Direct Connect in the *Direct Connect User Guide*.

Type: Boolean

Required: No

tags

The tags to associate with the lag.

Type: Array of Tag objects

Array Members: Minimum number of 1 item.

Required: No

Response Syntax

```
{
   "awsDevice": "string",
   "awsDeviceV2": "string",
   "awsLogicalDeviceId": "string",
   "bandwidth": "string",
   "connectionId": "string",
   "connectionName": "string",
   "connectionState": "string",
   "encryptionMode": "string",
   "hasLogicalRedundancy": "string",
   "jumboFrameCapable": boolean,
   "lagId": "string",
   "loaIssueTime": number,
   "location": "string",
   "macSecCapable": boolean,
   "macSecKeys": [
      {
         "ckn": "string",
         "secretARN": "string",
         "startOn": "string",
         "state": "string"
      }
   ],
   "ownerAccount": "string",
   "partnerInterconnectMacSecCapable": boolean,
   "partnerName": "string",
   "portEncryptionStatus": "string",
   "providerName": "string",
   "region": "string",
   "tags": [
      {
```

```
            "key": "string",
            "value": "string"
        }
    ],
    "vlan": number
}
```

Response Elements

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

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

awsDevice

This parameter has been deprecated.

The Direct Connect endpoint on which the physical connection terminates.

Type: String

awsDeviceV2

The Direct Connect endpoint that terminates the physical connection.

Type: String

awsLogicalDeviceId

The Direct Connect endpoint that terminates the logical connection. This device might be different than the device that terminates the physical connection.

Type: String

bandwidth

The bandwidth of the connection.

Type: String

connectionId

The ID of the connection.

Type: String

connectionName

The name of the connection.

Type: String

connectionState

The state of the connection. The following are the possible values:

- `ordering`: The initial state of a hosted connection provisioned on an interconnect. The connection stays in the ordering state until the owner of the hosted connection confirms or declines the connection order.
- `requested`: The initial state of a standard connection. The connection stays in the requested state until the Letter of Authorization (LOA) is sent to the customer.
- `pending`: The connection has been approved and is being initialized.
- `available`: The network link is up and the connection is ready for use.
- down: The network link is down.
- `deleting`: The connection is being deleted.
- `deleted`: The connection has been deleted.
- `rejected`: A hosted connection in the `ordering` state enters the `rejected` state if it is deleted by the customer.
- unknown: The state of the connection is not available.

Type: String

Valid Values: `ordering | requested | pending | available | down | deleting | deleted | rejected | unknown`

encryptionMode

The MAC Security (MACsec) connection encryption mode.

The valid values are `no_encrypt`, `should_encrypt`, and `must_encrypt`.

Type: String

hasLogicalRedundancy

Indicates whether the connection supports a secondary BGP peer in the same address family (IPv4/IPv6).

Type: String

Valid Values: `unknown | yes | no`

jumboFrameCapable

Indicates whether jumbo frames are supported.

Type: Boolean

lagId

The ID of the LAG.

Type: String

loaIssueTime

The time of the most recent call to DescribeLoa for this connection.

Type: Timestamp

location

The location of the connection.

Type: String

macSecCapable

Indicates whether the connection supports MAC Security (MACsec).

Type: Boolean

macSecKeys

The MAC Security (MACsec) security keys associated with the connection.

Type: Array of MacSecKey objects

ownerAccount

The ID of the AWS account that owns the connection.

Type: String

partnerInterconnectMacSecCapable

Indicates whether the interconnect hosting this connection supports MAC Security (MACsec).

Type: Boolean

partnerName

The name of the Direct Connect service provider associated with the connection.

Type: String

portEncryptionStatus

The MAC Security (MACsec) port link status of the connection.

The valid values are `Encryption Up`, which means that there is an active Connection Key Name, or `Encryption Down`.

Type: String

providerName

The name of the service provider associated with the connection.

Type: String

region

The AWS Region where the connection is located.

Type: String

tags

The tags associated with the connection.

Type: Array of Tag objects

Array Members: Minimum number of 1 item.

vlan

The ID of the VLAN.

Type: Integer

Errors

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

DirectConnectClientException

One or more parameters are not valid.

HTTP Status Code: 400

DirectConnectServerException

A server-side error occurred.

HTTP Status Code: 400

DuplicateTagKeysException

A tag key was specified more than once.

HTTP Status Code: 400

TooManyTagsException

You have reached the limit on the number of tags that can be assigned.

HTTP Status Code: 400

See Also

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

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

CreateDirectConnectGateway

Creates a Direct Connect gateway, which is an intermediate object that enables you to connect a set of virtual interfaces and virtual private gateways. A Direct Connect gateway is global and visible in any AWS Region after it is created. The virtual interfaces and virtual private gateways that are connected through a Direct Connect gateway can be in different AWS Regions. This enables you to connect to a VPC in any Region, regardless of the Region in which the virtual interfaces are located, and pass traffic between them.

Request Syntax

```
{
   "amazonSideAsn": number,
   "directConnectGatewayName": "string",
   "tags": [
      {
         "key": "string",
         "value": "string"
      }
   ]
}
```

Request Parameters

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

The request accepts the following data in JSON format.

amazonSideAsn

The autonomous system number (ASN) for Border Gateway Protocol (BGP) to be configured on the Amazon side of the connection. The ASN must be in the private range of 64,512 to 65,534 or 4,200,000,000 to 4,294,967,294. The default is 64512.

Type: Long

Required: No

directConnectGatewayName

The name of the Direct Connect gateway.

Type: String

Required: Yes

tags

The key-value pair tags associated with the request.

Type: Array of Tag objects

Array Members: Minimum number of 1 item.

Required: No

Response Syntax

```
{
   "directConnectGateway": {
      "amazonSideAsn": number,
      "directConnectGatewayId": "string",
      "directConnectGatewayName": "string",
      "directConnectGatewayState": "string",
      "ownerAccount": "string",
      "stateChangeError": "string",
      "tags": [
         {
            "key": "string",
            "value": "string"
         }
      ]
   }
}
```

Response Elements

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

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

directConnectGateway

The Direct Connect gateway.

Type: DirectConnectGateway object

Errors

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

DirectConnectClientException

One or more parameters are not valid.

HTTP Status Code: 400

DirectConnectServerException

A server-side error occurred.

HTTP Status Code: 400

See Also

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

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

CreateDirectConnectGatewayAssociation

Creates an association between a Direct Connect gateway and a virtual private gateway. The virtual private gateway must be attached to a VPC and must not be associated with another Direct Connect gateway.

Request Syntax

```
{
   "addAllowedPrefixesToDirectConnectGateway": [
      {
         "cidr": "string"
      }
   ],
   "directConnectGatewayId": "string",
   "gatewayId": "string",
   "virtualGatewayId": "string"
}
```

Request Parameters

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

The request accepts the following data in JSON format.

addAllowedPrefixesToDirectConnectGateway

The Amazon VPC prefixes to advertise to the Direct Connect gateway

This parameter is required when you create an association to a transit gateway.

For information about how to set the prefixes, see Allowed Prefixes in the *Direct Connect User Guide*.

Type: Array of RouteFilterPrefix objects

Required: No

directConnectGatewayId

The ID of the Direct Connect gateway.

Type: String

Required: Yes

gatewayId

The ID of the virtual private gateway or transit gateway.

Type: String

Required: No

virtualGatewayId

The ID of the virtual private gateway.

Type: String

Required: No

Response Syntax

```
{
   "directConnectGatewayAssociation": {
      "allowedPrefixesToDirectConnectGateway": [
         {
            "cidr": "string"
         }
      ],
      "associatedCoreNetwork": {
         "attachmentId": "string",
         "id": "string",
         "ownerAccount": "string"
      },
      "associatedGateway": {
         "id": "string",
         "ownerAccount": "string",
         "region": "string",
         "type": "string"
      },
      "associationId": "string",
      "associationState": "string",
      "directConnectGatewayId": "string",
      "directConnectGatewayOwnerAccount": "string",
```

```
        "stateChangeError": "string",
        "virtualGatewayId": "string",
        "virtualGatewayOwnerAccount": "string",
        "virtualGatewayRegion": "string"
    }
}
```

Response Elements

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

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

directConnectGatewayAssociation

The association to be created.

Type: DirectConnectGatewayAssociation object

Errors

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

DirectConnectClientException

One or more parameters are not valid.

HTTP Status Code: 400

DirectConnectServerException

A server-side error occurred.

HTTP Status Code: 400

See Also

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

- AWS Command Line Interface V2

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

CreateDirectConnectGatewayAssociationProposal

Creates a proposal to associate the specified virtual private gateway or transit gateway with the specified Direct Connect gateway.

You can associate a Direct Connect gateway and virtual private gateway or transit gateway that is owned by any AWS account.

Request Syntax

```
{
   "addAllowedPrefixesToDirectConnectGateway": [
      {
         "cidr": "string"
      }
   ],
   "directConnectGatewayId": "string",
   "directConnectGatewayOwnerAccount": "string",
   "gatewayId": "string",
   "removeAllowedPrefixesToDirectConnectGateway": [
      {
         "cidr": "string"
      }
   ]
}
```

Request Parameters

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

The request accepts the following data in JSON format.

addAllowedPrefixesToDirectConnectGateway

The Amazon VPC prefixes to advertise to the Direct Connect gateway.

Type: Array of RouteFilterPrefix objects

Required: No

directConnectGatewayId

The ID of the Direct Connect gateway.

Type: String

Required: Yes

directConnectGatewayOwnerAccount

The ID of the AWS account that owns the Direct Connect gateway.

Type: String

Required: Yes

gatewayId

The ID of the virtual private gateway or transit gateway.

Type: String

Required: Yes

removeAllowedPrefixesToDirectConnectGateway

The Amazon VPC prefixes to no longer advertise to the Direct Connect gateway.

Type: Array of RouteFilterPrefix objects

Required: No

Response Syntax

```
{
   "directConnectGatewayAssociationProposal": {
      "associatedGateway": {
         "id": "string",
         "ownerAccount": "string",
         "region": "string",
         "type": "string"
      },
      "directConnectGatewayId": "string",
      "directConnectGatewayOwnerAccount": "string",
      "existingAllowedPrefixesToDirectConnectGateway": [
         {
            "cidr": "string"
         }
      ],
```

```
      "proposalId": "string",
      "proposalState": "string",
      "requestedAllowedPrefixesToDirectConnectGateway": [
         {
            "cidr": "string"
         }
      ]
   }
}
```

Response Elements

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

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

directConnectGatewayAssociationProposal

Information about the Direct Connect gateway proposal.

Type: DirectConnectGatewayAssociationProposal object

Errors

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

DirectConnectClientException

One or more parameters are not valid.

HTTP Status Code: 400

DirectConnectServerException

A server-side error occurred.

HTTP Status Code: 400

See Also

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

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

CreateInterconnect

Creates an interconnect between an Direct Connect Partner's network and a specific Direct Connect location.

An interconnect is a connection that is capable of hosting other connections. The Direct Connect Partner can use an interconnect to provide Direct Connect hosted connections to customers through their own network services. Like a standard connection, an interconnect links the partner's network to an Direct Connect location over a standard Ethernet fiber-optic cable. One end is connected to the partner's router, the other to an Direct Connect router.

You can automatically add the new interconnect to a link aggregation group (LAG) by specifying a LAG ID in the request. This ensures that the new interconnect is allocated on the same Direct Connect endpoint that hosts the specified LAG. If there are no available ports on the endpoint, the request fails and no interconnect is created.

For each end customer, the Direct Connect Partner provisions a connection on their interconnect by calling AllocateHostedConnection. The end customer can then connect to AWS resources by creating a virtual interface on their connection, using the VLAN assigned to them by the Direct Connect Partner.

> **Note**
>
> Intended for use by Direct Connect Partners only.

Request Syntax

```
{
   "bandwidth": "string",
   "interconnectName": "string",
   "lagId": "string",
   "location": "string",
   "providerName": "string",
   "requestMACSec": boolean,
   "tags": [
      {
         "key": "string",
         "value": "string"
      }
```

```
    ]
}
```

Request Parameters

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

The request accepts the following data in JSON format.

bandwidth

The port bandwidth, in Gbps. The possible values are 1, 10, and 100.

Type: String

Required: Yes

interconnectName

The name of the interconnect.

Type: String

Required: Yes

lagId

The ID of the LAG.

Type: String

Required: No

location

The location of the interconnect.

Type: String

Required: Yes

providerName

The name of the service provider associated with the interconnect.

Type: String

Required: No

requestMACSec

Indicates whether you want the interconnect to support MAC Security (MACsec).

Type: Boolean

Required: No

tags

The tags to associate with the interconnect.

Type: Array of Tag objects

Array Members: Minimum number of 1 item.

Required: No

Response Syntax

```
{
   "awsDevice": "string",
   "awsDeviceV2": "string",
   "awsLogicalDeviceId": "string",
   "bandwidth": "string",
   "encryptionMode": "string",
   "hasLogicalRedundancy": "string",
   "interconnectId": "string",
   "interconnectName": "string",
   "interconnectState": "string",
   "jumboFrameCapable": boolean,
   "lagId": "string",
   "loaIssueTime": number,
   "location": "string",
   "macSecCapable": boolean,
   "macSecKeys": [
      {
         "ckn": "string",
         "secretARN": "string",
         "startOn": "string",
         "state": "string"
      }
```

```
    ],
    "portEncryptionStatus": "string",
    "providerName": "string",
    "region": "string",
    "tags": [
       {
          "key": "string",
          "value": "string"
       }
    ]
}
```

Response Elements

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

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

awsDevice

This parameter has been deprecated.

The Direct Connect endpoint on which the physical connection terminates.

Type: String

awsDeviceV2

The Direct Connect endpoint that terminates the physical connection.

Type: String

awsLogicalDeviceId

The Direct Connect endpoint that terminates the logical connection. This device might be different than the device that terminates the physical connection.

Type: String

bandwidth

The bandwidth of the connection.

Type: String

encryptionMode

The MAC Security (MACsec) encryption mode.

The valid values are `no_encrypt`, `should_encrypt`, and `must_encrypt`.

Type: String

hasLogicalRedundancy

Indicates whether the interconnect supports a secondary BGP in the same address family (IPv4/IPv6).

Type: String

Valid Values: `unknown | yes | no`

interconnectId

The ID of the interconnect.

Type: String

interconnectName

The name of the interconnect.

Type: String

interconnectState

The state of the interconnect. The following are the possible values:

- `requested`: The initial state of an interconnect. The interconnect stays in the requested state until the Letter of Authorization (LOA) is sent to the customer.
- `pending`: The interconnect is approved, and is being initialized.
- `available`: The network link is up, and the interconnect is ready for use.
- `down`: The network link is down.
- `deleting`: The interconnect is being deleted.
- `deleted`: The interconnect is deleted.
- `unknown`: The state of the interconnect is not available.

Type: String

Valid Values: `requested | pending | available | down | deleting | deleted | unknown`

jumboFrameCapable

Indicates whether jumbo frames are supported.

Type: Boolean

lagId

The ID of the LAG.

Type: String

loaIssueTime

The time of the most recent call to DescribeLoa for this connection.

Type: Timestamp

location

The location of the connection.

Type: String

macSecCapable

Indicates whether the interconnect supports MAC Security (MACsec).

Type: Boolean

macSecKeys

The MAC Security (MACsec) security keys.

Type: Array of MacSecKey objects

portEncryptionStatus

The MAC Security (MACsec) port link status.

The valid values are `Encryption Up`, which means that there is an active Connection Key Name, or `Encryption Down`.

Type: String

providerName

The name of the service provider associated with the interconnect.

Type: String

region

The AWS Region where the connection is located.

Type: String

tags

The tags associated with the interconnect.

Type: Array of Tag objects

Array Members: Minimum number of 1 item.

Errors

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

DirectConnectClientException

One or more parameters are not valid.

HTTP Status Code: 400

DirectConnectServerException

A server-side error occurred.

HTTP Status Code: 400

DuplicateTagKeysException

A tag key was specified more than once.

HTTP Status Code: 400

TooManyTagsException

You have reached the limit on the number of tags that can be assigned.

HTTP Status Code: 400

See Also

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

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

CreateLag

Creates a link aggregation group (LAG) with the specified number of bundled physical dedicated connections between the customer network and a specific Direct Connect location. A LAG is a logical interface that uses the Link Aggregation Control Protocol (LACP) to aggregate multiple interfaces, enabling you to treat them as a single interface.

All connections in a LAG must use the same bandwidth (either 1Gbps, 10Gbps, 100Gbps, or 400Gbps) and must terminate at the same Direct Connect endpoint.

You can have up to 10 dedicated connections per location. Regardless of this limit, if you request more connections for the LAG than Direct Connect can allocate on a single endpoint, no LAG is created..

You can specify an existing physical dedicated connection or interconnect to include in the LAG (which counts towards the total number of connections). Doing so interrupts the current physical dedicated connection, and re-establishes them as a member of the LAG. The LAG will be created on the same Direct Connect endpoint to which the dedicated connection terminates. Any virtual interfaces associated with the dedicated connection are automatically disassociated and re-associated with the LAG. The connection ID does not change.

If the AWS account used to create a LAG is a registered Direct Connect Partner, the LAG is automatically enabled to host sub-connections. For a LAG owned by a partner, any associated virtual interfaces cannot be directly configured.

Request Syntax

```
{
   "childConnectionTags": [
      {
         "key": "string",
         "value": "string"
      }
   ],
   "connectionId": "string",
   "connectionsBandwidth": "string",
   "lagName": "string",
   "location": "string",
   "numberOfConnections": number,
   "providerName": "string",
   "requestMACSec": boolean,
```

```
   "tags": [
      {
         "key": "string",
         "value": "string"
      }
   ]
}
```

Request Parameters

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

The request accepts the following data in JSON format.

childConnectionTags

The tags to associate with the automtically created LAGs.

Type: Array of Tag objects

Array Members: Minimum number of 1 item.

Required: No

connectionId

The ID of an existing dedicated connection to migrate to the LAG.

Type: String

Required: No

connectionsBandwidth

The bandwidth of the individual physical dedicated connections bundled by the LAG. The possible values are 1Gbps,10Gbps, 100Gbps, and 400Gbps.

Type: String

Required: Yes

lagName

The name of the LAG.

Type: String

Required: Yes

location

The location for the LAG.

Type: String

Required: Yes

numberOfConnections

The number of physical dedicated connections initially provisioned and bundled by the LAG. You can have a maximum of four connections when the port speed is 1Gbps or 10Gbps, or two when the port speed is 100Gbps or 400Gbps.

Type: Integer

Required: Yes

providerName

The name of the service provider associated with the LAG.

Type: String

Required: No

requestMACSec

Indicates whether the connection will support MAC Security (MACsec).

> **Note**
>
> All connections in the LAG must be capable of supporting MAC Security (MACsec). For information about MAC Security (MACsec) prerequisties, see MACsec prerequisties in the *Direct Connect User Guide*.

Type: Boolean

Required: No

tags

The tags to associate with the LAG.

Type: Array of Tag objects

Array Members: Minimum number of 1 item.

Required: No

Response Syntax

```
{
   "allowsHostedConnections": boolean,
   "awsDevice": "string",
   "awsDeviceV2": "string",
   "awsLogicalDeviceId": "string",
   "connections": [
      {
         "awsDevice": "string",
         "awsDeviceV2": "string",
         "awsLogicalDeviceId": "string",
         "bandwidth": "string",
         "connectionId": "string",
         "connectionName": "string",
         "connectionState": "string",
         "encryptionMode": "string",
         "hasLogicalRedundancy": "string",
         "jumboFrameCapable": boolean,
         "lagId": "string",
         "loaIssueTime": number,
         "location": "string",
         "macSecCapable": boolean,
         "macSecKeys": [
            {
               "ckn": "string",
               "secretARN": "string",
               "startOn": "string",
               "state": "string"
            }
         ],
         "ownerAccount": "string",
         "partnerInterconnectMacSecCapable": boolean,
         "partnerName": "string",
         "portEncryptionStatus": "string",
         "providerName": "string",
         "region": "string",
```

```
            "tags": [
               {
                  "key": "string",
                  "value": "string"
               }
            ],
            "vlan": number
         }
      ],
      "connectionsBandwidth": "string",
      "encryptionMode": "string",
      "hasLogicalRedundancy": "string",
      "jumboFrameCapable": boolean,
      "lagId": "string",
      "lagName": "string",
      "lagState": "string",
      "location": "string",
      "macSecCapable": boolean,
      "macSecKeys": [
         {
            "ckn": "string",
            "secretARN": "string",
            "startOn": "string",
            "state": "string"
         }
      ],
      "minimumLinks": number,
      "numberOfConnections": number,
      "ownerAccount": "string",
      "providerName": "string",
      "region": "string",
      "tags": [
         {
            "key": "string",
            "value": "string"
         }
      ]
}
```

Response Elements

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

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

allowsHostedConnections

Indicates whether the LAG can host other connections.

Type: Boolean

awsDevice

This parameter has been deprecated.

The Direct Connect endpoint that hosts the LAG.

Type: String

awsDeviceV2

The Direct Connect endpoint that hosts the LAG.

Type: String

awsLogicalDeviceId

The Direct Connect endpoint that terminates the logical connection. This device might be different than the device that terminates the physical connection.

Type: String

connections

The connections bundled by the LAG.

Type: Array of Connection objects

connectionsBandwidth

The individual bandwidth of the physical connections bundled by the LAG. The possible values are 1Gbps, 10Gbps, 100Gbps, or 400 Gbps..

Type: String

encryptionMode

The LAG MAC Security (MACsec) encryption mode.

The valid values are `no_encrypt`, `should_encrypt`, and `must_encrypt`.

Type: String

hasLogicalRedundancy

Indicates whether the LAG supports a secondary BGP peer in the same address family (IPv4/IPv6).

Type: String

Valid Values: unknown | yes | no

jumboFrameCapable

Indicates whether jumbo frames are supported.

Type: Boolean

lagId

The ID of the LAG.

Type: String

lagName

The name of the LAG.

Type: String

lagState

The state of the LAG. The following are the possible values:

- requested: The initial state of a LAG. The LAG stays in the requested state until the Letter of Authorization (LOA) is available.
- pending: The LAG has been approved and is being initialized.
- available: The network link is established and the LAG is ready for use.
- down: The network link is down.
- deleting: The LAG is being deleted.
- deleted: The LAG is deleted.
- unknown: The state of the LAG is not available.

Type: String

Valid Values: requested | pending | available | down | deleting | deleted | unknown

location

The location of the LAG.

Type: String

macSecCapable

Indicates whether the LAG supports MAC Security (MACsec).

Type: Boolean

macSecKeys

The MAC Security (MACsec) security keys associated with the LAG.

Type: Array of MacSecKey objects

minimumLinks

The minimum number of physical dedicated connections that must be operational for the LAG itself to be operational.

Type: Integer

numberOfConnections

The number of physical dedicated connections initially provisioned and bundled by the LAG. You can have a maximum of four connections when the port speed is 1 Gbps or 10 Gbps, or two when the port speed is 100 Gbps or 400 Gbps.

Type: Integer

ownerAccount

The ID of the AWS account that owns the LAG.

Type: String

providerName

The name of the service provider associated with the LAG.

Type: String

region

The AWS Region where the connection is located.

Type: String

tags

The tags associated with the LAG.

Type: Array of Tag objects

Array Members: Minimum number of 1 item.

Errors

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

DirectConnectClientException

One or more parameters are not valid.

HTTP Status Code: 400

DirectConnectServerException

A server-side error occurred.

HTTP Status Code: 400

DuplicateTagKeysException

A tag key was specified more than once.

HTTP Status Code: 400

TooManyTagsException

You have reached the limit on the number of tags that can be assigned.

HTTP Status Code: 400

See Also

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

- AWS Command Line Interface V2

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

CreatePrivateVirtualInterface

Creates a private virtual interface. A virtual interface is the VLAN that transports Direct Connect traffic. A private virtual interface can be connected to either a Direct Connect gateway or a Virtual Private Gateway (VGW). Connecting the private virtual interface to a Direct Connect gateway enables the possibility for connecting to multiple VPCs, including VPCs in different AWS Regions. Connecting the private virtual interface to a VGW only provides access to a single VPC within the same Region.

Setting the MTU of a virtual interface to 8500 (jumbo frames) can cause an update to the underlying physical connection if it wasn't updated to support jumbo frames. Updating the connection disrupts network connectivity for all virtual interfaces associated with the connection for up to 30 seconds. To check whether your connection supports jumbo frames, call DescribeConnections. To check whether your virtual interface supports jumbo frames, call DescribeVirtualInterfaces.

Request Syntax

```
{
   "connectionId": "string",
   "newPrivateVirtualInterface": {
      "addressFamily": "string",
      "amazonAddress": "string",
      "asn": number,
      "asnLong": number,
      "authKey": "string",
      "customerAddress": "string",
      "directConnectGatewayId": "string",
      "enableSiteLink": boolean,
      "mtu": number,
      "tags": [
         {
            "key": "string",
            "value": "string"
         }
      ],
      "virtualGatewayId": "string",
      "virtualInterfaceName": "string",
      "vlan": number
   }
}
```

Request Parameters

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

The request accepts the following data in JSON format.

connectionId

The ID of the connection.

Type: String

Required: Yes

newPrivateVirtualInterface

Information about the private virtual interface.

Type: NewPrivateVirtualInterface object

Required: Yes

Response Syntax

```
{
   "addressFamily": "string",
   "amazonAddress": "string",
   "amazonSideAsn": number,
   "asn": number,
   "asnLong": number,
   "authKey": "string",
   "awsDeviceV2": "string",
   "awsLogicalDeviceId": "string",
   "bgpPeers": [
      {
         "addressFamily": "string",
         "amazonAddress": "string",
         "asn": number,
         "asnLong": number,
         "authKey": "string",
         "awsDeviceV2": "string",
         "awsLogicalDeviceId": "string",
         "bgpPeerId": "string",
```

```
            "bgpPeerState": "string",
            "bgpStatus": "string",
            "customerAddress": "string"
         }
      ],
      "connectionId": "string",
      "customerAddress": "string",
      "customerRouterConfig": "string",
      "directConnectGatewayId": "string",
      "jumboFrameCapable": boolean,
      "location": "string",
      "mtu": number,
      "ownerAccount": "string",
      "region": "string",
      "routeFilterPrefixes": [
         {
            "cidr": "string"
         }
      ],
      "siteLinkEnabled": boolean,
      "tags": [
         {
            "key": "string",
            "value": "string"
         }
      ],
      "virtualGatewayId": "string",
      "virtualInterfaceId": "string",
      "virtualInterfaceName": "string",
      "virtualInterfaceState": "string",
      "virtualInterfaceType": "string",
      "vlan": number
}
```

Response Elements

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

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

addressFamily

The address family for the BGP peer.

Type: String

Valid Values: `ipv4 | ipv6`

amazonAddress

The IP address assigned to the Amazon interface.

Type: String

amazonSideAsn

The autonomous system number (AS) for the Amazon side of the connection.

Type: Long

asn

The autonomous system number (ASN). The valid range is from 1 to 2147483646 for Border Gateway Protocol (BGP) configuration. If you provide a number greater than the maximum, an error is returned. Use `asnLong` instead.

- You can use `asnLong` or `asn`, but not both. We recommend using `asnLong` as it supports a greater pool of numbers.
- If you provide a value in the same API call for both `asn` and `asnLong`, the API will only accept the value for `asnLong`.
- If you enter a 4-byte ASN for the `asn` parameter, the API returns an error.
- If you are using a 2-byte ASN, the API response will include the 2-byte value for both the `asn` and `asnLong` fields.

Type: Integer

asnLong

The long ASN for the virtual interface. The valid range is from 1 to 4294967294 for BGP configuration.

Note the following limitations when using `asnLong`:

- You can use `asnLong` or `asn`, but not both. We recommend using `asnLong` as it supports a greater pool of numbers.
- `asnLong` accepts any valid ASN value, regardless if it's 2-byte or 4-byte.
- When using a 4-byte `asnLong`, the API response returns `0` for the legacy `asn` attribute since 4-byte ASN values exceed the maximum supported value of 2,147,483,647.

- If you are using a 2-byte ASN, the API response will include the 2-byte value for both the `asn` and `asnLong` fields.
- If you provide a value in the same API call for both `asn` and `asnLong`, the API will only accept the value for `asnLong`.

Type: Long

authKey

The authentication key for BGP configuration. This string has a minimum length of 6 characters and and a maximun lenth of 80 characters.

Type: String

awsDeviceV2

The Direct Connect endpoint that terminates the physical connection.

Type: String

awsLogicalDeviceId

The Direct Connect endpoint that terminates the logical connection. This device might be different than the device that terminates the physical connection.

Type: String

bgpPeers

The BGP peers configured on this virtual interface.

Type: Array of BGPPeer objects

connectionId

The ID of the connection.

Type: String

customerAddress

The IP address assigned to the customer interface.

Type: String

customerRouterConfig

The customer router configuration.

Type: String

directConnectGatewayId

The ID of the Direct Connect gateway.

Type: String

jumboFrameCapable

Indicates whether jumbo frames are supported.

Type: Boolean

location

The location of the connection.

Type: String

mtu

The maximum transmission unit (MTU), in bytes. The supported values are 1500 and 8500. The default value is 1500

Type: Integer

ownerAccount

The ID of the AWS account that owns the virtual interface.

Type: String

region

The AWS Region where the virtual interface is located.

Type: String

routeFilterPrefixes

The routes to be advertised to the AWS network in this Region. Applies to public virtual interfaces.

Type: Array of RouteFilterPrefix objects

siteLinkEnabled

Indicates whether SiteLink is enabled.

Type: Boolean

tags

The tags associated with the virtual interface.

Type: Array of Tag objects

Array Members: Minimum number of 1 item.

virtualGatewayId

The ID of the virtual private gateway. Applies only to private virtual interfaces.

Type: String

virtualInterfaceId

The ID of the virtual interface.

Type: String

virtualInterfaceName

The name of the virtual interface assigned by the customer network. The name has a maximum of 100 characters. The following are valid characters: a-z, 0-9 and a hyphen (-).

Type: String

virtualInterfaceState

The state of the virtual interface. The following are the possible values:

- `confirming`: The creation of the virtual interface is pending confirmation from the virtual interface owner. If the owner of the virtual interface is different from the owner of the connection on which it is provisioned, then the virtual interface will remain in this state until it is confirmed by the virtual interface owner.
- `verifying`: This state only applies to public virtual interfaces. Each public virtual interface needs validation before the virtual interface can be created.
- `pending`: A virtual interface is in this state from the time that it is created until the virtual interface is ready to forward traffic.
- `available`: A virtual interface that is able to forward traffic.
- `down`: A virtual interface that is BGP down.

- `testing`: A virtual interface is in this state immediately after calling StartBgpFailoverTest and remains in this state during the duration of the test.
- `deleting`: A virtual interface is in this state immediately after calling DeleteVirtualInterface until it can no longer forward traffic.
- `deleted`: A virtual interface that cannot forward traffic.
- `rejected`: The virtual interface owner has declined creation of the virtual interface. If a virtual interface in the `Confirming` state is deleted by the virtual interface owner, the virtual interface enters the `Rejected` state.
- unknown: The state of the virtual interface is not available.

Type: String

Valid Values: `confirming | verifying | pending | available | down | testing | deleting | deleted | rejected | unknown`

virtualInterfaceType

The type of virtual interface. The possible values are `private`, `public` and `transit`.

Type: String

vlan

The ID of the VLAN.

Type: Integer

Errors

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

DirectConnectClientException

One or more parameters are not valid.

HTTP Status Code: 400

DirectConnectServerException

A server-side error occurred.

HTTP Status Code: 400

DuplicateTagKeysException

A tag key was specified more than once.

HTTP Status Code: 400

TooManyTagsException

You have reached the limit on the number of tags that can be assigned.

HTTP Status Code: 400

See Also

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

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

CreatePublicVirtualInterface

Creates a public virtual interface. A virtual interface is the VLAN that transports Direct Connect traffic. A public virtual interface supports sending traffic to public services of AWS such as Amazon S3.

When creating an IPv6 public virtual interface (`addressFamily` is `ipv6`), leave the `customer` and `amazon` address fields blank to use auto-assigned IPv6 space. Custom IPv6 addresses are not supported.

Request Syntax

```
{
   "connectionId": "string",
   "newPublicVirtualInterface": {
      "addressFamily": "string",
      "amazonAddress": "string",
      "asn": number,
      "asnLong": number,
      "authKey": "string",
      "customerAddress": "string",
      "routeFilterPrefixes": [
         {
            "cidr": "string"
         }
      ],
      "tags": [
         {
            "key": "string",
            "value": "string"
         }
      ],
      "virtualInterfaceName": "string",
      "vlan": number
   }
}
```

Request Parameters

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

The request accepts the following data in JSON format.

connectionId

The ID of the connection.

Type: String

Required: Yes

newPublicVirtualInterface

Information about the public virtual interface.

Type: NewPublicVirtualInterface object

Required: Yes

Response Syntax

```
{
   "addressFamily": "string",
   "amazonAddress": "string",
   "amazonSideAsn": number,
   "asn": number,
   "asnLong": number,
   "authKey": "string",
   "awsDeviceV2": "string",
   "awsLogicalDeviceId": "string",
   "bgpPeers": [
      {
         "addressFamily": "string",
         "amazonAddress": "string",
         "asn": number,
         "asnLong": number,
         "authKey": "string",
         "awsDeviceV2": "string",
         "awsLogicalDeviceId": "string",
         "bgpPeerId": "string",
         "bgpPeerState": "string",
         "bgpStatus": "string",
         "customerAddress": "string"
      }
   ],
   "connectionId": "string",
   "customerAddress": "string",
```

```
    "customerRouterConfig": "string",
    "directConnectGatewayId": "string",
    "jumboFrameCapable": boolean,
    "location": "string",
    "mtu": number,
    "ownerAccount": "string",
    "region": "string",
    "routeFilterPrefixes": [
       {
          "cidr": "string"
       }
    ],
    "siteLinkEnabled": boolean,
    "tags": [
       {
          "key": "string",
          "value": "string"
       }
    ],
    "virtualGatewayId": "string",
    "virtualInterfaceId": "string",
    "virtualInterfaceName": "string",
    "virtualInterfaceState": "string",
    "virtualInterfaceType": "string",
    "vlan": number
}
```

Response Elements

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

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

addressFamily

The address family for the BGP peer.

Type: String

Valid Values: `ipv4 | ipv6`

amazonAddress

The IP address assigned to the Amazon interface.

Type: String

amazonSideAsn

The autonomous system number (AS) for the Amazon side of the connection.

Type: Long

asn

The autonomous system number (ASN). The valid range is from 1 to 2147483646 for Border Gateway Protocol (BGP) configuration. If you provide a number greater than the maximum, an error is returned. Use asnLong instead.

- You can use asnLong or asn, but not both. We recommend using asnLong as it supports a greater pool of numbers.
- If you provide a value in the same API call for both asn and asnLong, the API will only accept the value for asnLong.
- If you enter a 4-byte ASN for the asn parameter, the API returns an error.
- If you are using a 2-byte ASN, the API response will include the 2-byte value for both the asn and asnLong fields.

Type: Integer

asnLong

The long ASN for the virtual interface. The valid range is from 1 to 4294967294 for BGP configuration.

Note the following limitations when using asnLong:

- You can use asnLong or asn, but not both. We recommend using asnLong as it supports a greater pool of numbers.
- asnLong accepts any valid ASN value, regardless if it's 2-byte or 4-byte.
- When using a 4-byte asnLong, the API response returns 0 for the legacy asn attribute since 4-byte ASN values exceed the maximum supported value of 2,147,483,647.
- If you are using a 2-byte ASN, the API response will include the 2-byte value for both the asn and asnLong fields.
- If you provide a value in the same API call for both asn and asnLong, the API will only accept the value for asnLong.

Type: Long

authKey

The authentication key for BGP configuration. This string has a minimum length of 6 characters and and a maximun lenth of 80 characters.

Type: String

awsDeviceV2

The Direct Connect endpoint that terminates the physical connection.

Type: String

awsLogicalDeviceId

The Direct Connect endpoint that terminates the logical connection. This device might be different than the device that terminates the physical connection.

Type: String

bgpPeers

The BGP peers configured on this virtual interface.

Type: Array of BGPPeer objects

connectionId

The ID of the connection.

Type: String

customerAddress

The IP address assigned to the customer interface.

Type: String

customerRouterConfig

The customer router configuration.

Type: String

directConnectGatewayId

The ID of the Direct Connect gateway.

Type: String

jumboFrameCapable

Indicates whether jumbo frames are supported.

Type: Boolean

location

The location of the connection.

Type: String

mtu

The maximum transmission unit (MTU), in bytes. The supported values are 1500 and 8500. The default value is 1500

Type: Integer

ownerAccount

The ID of the AWS account that owns the virtual interface.

Type: String

region

The AWS Region where the virtual interface is located.

Type: String

routeFilterPrefixes

The routes to be advertised to the AWS network in this Region. Applies to public virtual interfaces.

Type: Array of RouteFilterPrefix objects

siteLinkEnabled

Indicates whether SiteLink is enabled.

Type: Boolean

tags

The tags associated with the virtual interface.

Type: Array of Tag objects

Array Members: Minimum number of 1 item.

virtualGatewayId

The ID of the virtual private gateway. Applies only to private virtual interfaces.

Type: String

virtualInterfaceId

The ID of the virtual interface.

Type: String

virtualInterfaceName

The name of the virtual interface assigned by the customer network. The name has a maximum of 100 characters. The following are valid characters: a-z, 0-9 and a hyphen (-).

Type: String

virtualInterfaceState

The state of the virtual interface. The following are the possible values:

- `confirming`: The creation of the virtual interface is pending confirmation from the virtual interface owner. If the owner of the virtual interface is different from the owner of the connection on which it is provisioned, then the virtual interface will remain in this state until it is confirmed by the virtual interface owner.
- `verifying`: This state only applies to public virtual interfaces. Each public virtual interface needs validation before the virtual interface can be created.
- `pending`: A virtual interface is in this state from the time that it is created until the virtual interface is ready to forward traffic.
- `available`: A virtual interface that is able to forward traffic.
- down: A virtual interface that is BGP down.
- `testing`: A virtual interface is in this state immediately after calling StartBgpFailoverTest and remains in this state during the duration of the test.
- `deleting`: A virtual interface is in this state immediately after calling DeleteVirtualInterface until it can no longer forward traffic.
- `deleted`: A virtual interface that cannot forward traffic.

- `rejected`: The virtual interface owner has declined creation of the virtual interface. If a virtual interface in the `Confirming` state is deleted by the virtual interface owner, the virtual interface enters the `Rejected` state.
- unknown: The state of the virtual interface is not available.

Type: String

Valid Values: `confirming | verifying | pending | available | down | testing | deleting | deleted | rejected | unknown`

virtualInterfaceType

The type of virtual interface. The possible values are `private`, `public` and `transit`.

Type: String

vlan

The ID of the VLAN.

Type: Integer

Errors

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

DirectConnectClientException

One or more parameters are not valid.

HTTP Status Code: 400

DirectConnectServerException

A server-side error occurred.

HTTP Status Code: 400

DuplicateTagKeysException

A tag key was specified more than once.

HTTP Status Code: 400

TooManyTagsException

You have reached the limit on the number of tags that can be assigned.

HTTP Status Code: 400

See Also

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

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

CreateTransitVirtualInterface

Creates a transit virtual interface. A transit virtual interface should be used to access one or more transit gateways associated with Direct Connect gateways. A transit virtual interface enables the connection of multiple VPCs attached to a transit gateway to a Direct Connect gateway.

> **Important**
>
> If you associate your transit gateway with one or more Direct Connect gateways, the Autonomous System Number (ASN) used by the transit gateway and the Direct Connect gateway must be different. For example, if you use the default ASN 64512 for both your the transit gateway and Direct Connect gateway, the association request fails.

A jumbo MTU value must be either 1500 or 8500. No other values will be accepted. Setting the MTU of a virtual interface to 8500 (jumbo frames) can cause an update to the underlying physical connection if it wasn't updated to support jumbo frames. Updating the connection disrupts network connectivity for all virtual interfaces associated with the connection for up to 30 seconds. To check whether your connection supports jumbo frames, call DescribeConnections. To check whether your virtual interface supports jumbo frames, call DescribeVirtualInterfaces.

Request Syntax

```
{
   "connectionId": "string",
   "newTransitVirtualInterface": {
      "addressFamily": "string",
      "amazonAddress": "string",
      "asn": number,
      "asnLong": number,
      "authKey": "string",
      "customerAddress": "string",
      "directConnectGatewayId": "string",
      "enableSiteLink": boolean,
      "mtu": number,
      "tags": [
         {
            "key": "string",
            "value": "string"
         }
```

```
        ],
        "virtualInterfaceName": "string",
        "vlan": number
    }
}
```

Request Parameters

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

The request accepts the following data in JSON format.

connectionId

The ID of the connection.

Type: String

Required: Yes

newTransitVirtualInterface

Information about the transit virtual interface.

Type: NewTransitVirtualInterface object

Required: Yes

Response Syntax

```
{
   "virtualInterface": {
      "addressFamily": "string",
      "amazonAddress": "string",
      "amazonSideAsn": number,
      "asn": number,
      "asnLong": number,
      "authKey": "string",
      "awsDeviceV2": "string",
      "awsLogicalDeviceId": "string",
      "bgpPeers": [
         {
            "addressFamily": "string",
            "amazonAddress": "string",
```

```
            "asn": number,
            "asnLong": number,
            "authKey": "string",
            "awsDeviceV2": "string",
            "awsLogicalDeviceId": "string",
            "bgpPeerId": "string",
            "bgpPeerState": "string",
            "bgpStatus": "string",
            "customerAddress": "string"
         }
      ],
      "connectionId": "string",
      "customerAddress": "string",
      "customerRouterConfig": "string",
      "directConnectGatewayId": "string",
      "jumboFrameCapable": boolean,
      "location": "string",
      "mtu": number,
      "ownerAccount": "string",
      "region": "string",
      "routeFilterPrefixes": [
         {
            "cidr": "string"
         }
      ],
      "siteLinkEnabled": boolean,
      "tags": [
         {
            "key": "string",
            "value": "string"
         }
      ],
      "virtualGatewayId": "string",
      "virtualInterfaceId": "string",
      "virtualInterfaceName": "string",
      "virtualInterfaceState": "string",
      "virtualInterfaceType": "string",
      "vlan": number
   }
}
```

Response Elements

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

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

virtualInterface

Information about a virtual interface.

Type: VirtualInterface object

Errors

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

DirectConnectClientException

One or more parameters are not valid.

HTTP Status Code: 400

DirectConnectServerException

A server-side error occurred.

HTTP Status Code: 400

DuplicateTagKeysException

A tag key was specified more than once.

HTTP Status Code: 400

TooManyTagsException

You have reached the limit on the number of tags that can be assigned.

HTTP Status Code: 400

See Also

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

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

DeleteBGPPeer

Deletes the specified BGP peer on the specified virtual interface with the specified customer address and ASN.

You cannot delete the last BGP peer from a virtual interface.

Request Syntax

```
{
   "asn": number,
   "asnLong": number,
   "bgpPeerId": "string",
   "customerAddress": "string",
   "virtualInterfaceId": "string"
}
```

Request Parameters

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

The request accepts the following data in JSON format.

asn

The autonomous system number (ASN). The valid range is from 1 to 2147483646 for Border Gateway Protocol (BGP) configuration. If you provide a number greater than the maximum, an error is returned. Use asnLong instead.

- You can use asnLong or asn, but not both. We recommend using asnLong as it supports a greater pool of numbers.
- If you provide a value in the same API call for both asn and asnLong, the API will only accept the value for asnLong.
- If you enter a 4-byte ASN for the asn parameter, the API returns an error.
- If you are using a 2-byte ASN, the API response will include the 2-byte value for both the asn and asnLong fields.

Type: Integer

Required: No

asnLong

The long ASN for the BGP peer to be deleted from a Direct Connect virtual interface. The valid range is from 1 to 4294967294 for BGP configuration.

Note the following limitations when using `asnLong`:

- You can use `asnLong` or `asn`, but not both. We recommend using `asnLong` as it supports a greater pool of numbers.
- `asnLong` accepts any valid ASN value, regardless if it's 2-byte or 4-byte.
- When using a 4-byte `asnLong`, the API response returns `0` for the legacy `asn` attribute since 4-byte ASN values exceed the maximum supported value of 2,147,483,647.
- If you are using a 2-byte ASN, the API response will include the 2-byte value for both the `asn` and `asnLong` fields.
- If you provide a value in the same API call for both `asn` and `asnLong`, the API will only accept the value for `asnLong`.

Type: Long

Required: No

bgpPeerId

The ID of the BGP peer.

Type: String

Required: No

customerAddress

The IP address assigned to the customer interface.

Type: String

Required: No

virtualInterfaceId

The ID of the virtual interface.

Type: String

Required: No

Response Syntax

```
{
   "virtualInterface": {
      "addressFamily": "string",
      "amazonAddress": "string",
      "amazonSideAsn": number,
      "asn": number,
      "asnLong": number,
      "authKey": "string",
      "awsDeviceV2": "string",
      "awsLogicalDeviceId": "string",
      "bgpPeers": [
         {
            "addressFamily": "string",
            "amazonAddress": "string",
            "asn": number,
            "asnLong": number,
            "authKey": "string",
            "awsDeviceV2": "string",
            "awsLogicalDeviceId": "string",
            "bgpPeerId": "string",
            "bgpPeerState": "string",
            "bgpStatus": "string",
            "customerAddress": "string"
         }
      ],
      "connectionId": "string",
      "customerAddress": "string",
      "customerRouterConfig": "string",
      "directConnectGatewayId": "string",
      "jumboFrameCapable": boolean,
      "location": "string",
      "mtu": number,
      "ownerAccount": "string",
      "region": "string",
      "routeFilterPrefixes": [
         {
            "cidr": "string"
```

```
            }
         ],
         "siteLinkEnabled": boolean,
         "tags": [
            {
               "key": "string",
               "value": "string"
            }
         ],
         "virtualGatewayId": "string",
         "virtualInterfaceId": "string",
         "virtualInterfaceName": "string",
         "virtualInterfaceState": "string",
         "virtualInterfaceType": "string",
         "vlan": number
      }
}
```

Response Elements

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

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

virtualInterface

The virtual interface.

Type: VirtualInterface object

Errors

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

DirectConnectClientException

One or more parameters are not valid.

HTTP Status Code: 400

DirectConnectServerException

A server-side error occurred.

HTTP Status Code: 400

See Also

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

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

DeleteConnection

Deletes the specified connection.

Deleting a connection only stops the Direct Connect port hour and data transfer charges. If you are partnering with any third parties to connect with the Direct Connect location, you must cancel your service with them separately.

Request Syntax

```
{
   "connectionId": "string"
}
```

Request Parameters

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

The request accepts the following data in JSON format.

connectionId

The ID of the connection.

Type: String

Required: Yes

Response Syntax

```
{
   "awsDevice": "string",
   "awsDeviceV2": "string",
   "awsLogicalDeviceId": "string",
   "bandwidth": "string",
   "connectionId": "string",
   "connectionName": "string",
   "connectionState": "string",
   "encryptionMode": "string",
   "hasLogicalRedundancy": "string",
   "jumboFrameCapable": boolean,
```

```
   "lagId": "string",
   "loaIssueTime": number,
   "location": "string",
   "macSecCapable": boolean,
   "macSecKeys": [
      {
         "ckn": "string",
         "secretARN": "string",
         "startOn": "string",
         "state": "string"
      }
   ],
   "ownerAccount": "string",
   "partnerInterconnectMacSecCapable": boolean,
   "partnerName": "string",
   "portEncryptionStatus": "string",
   "providerName": "string",
   "region": "string",
   "tags": [
      {
         "key": "string",
         "value": "string"
      }
   ],
   "vlan": number
}
```

Response Elements

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

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

awsDevice

This parameter has been deprecated.

The Direct Connect endpoint on which the physical connection terminates.

Type: String

awsDeviceV2

The Direct Connect endpoint that terminates the physical connection.

Type: String

awsLogicalDeviceId

The Direct Connect endpoint that terminates the logical connection. This device might be different than the device that terminates the physical connection.

Type: String

bandwidth

The bandwidth of the connection.

Type: String

connectionId

The ID of the connection.

Type: String

connectionName

The name of the connection.

Type: String

connectionState

The state of the connection. The following are the possible values:

- `ordering`: The initial state of a hosted connection provisioned on an interconnect. The connection stays in the ordering state until the owner of the hosted connection confirms or declines the connection order.
- `requested`: The initial state of a standard connection. The connection stays in the requested state until the Letter of Authorization (LOA) is sent to the customer.
- `pending`: The connection has been approved and is being initialized.
- `available`: The network link is up and the connection is ready for use.
- down: The network link is down.
- `deleting`: The connection is being deleted.
- `deleted`: The connection has been deleted.
- `rejected`: A hosted connection in the `ordering` state enters the `rejected` state if it is deleted by the customer.

- unknown: The state of the connection is not available.

Type: String

Valid Values: `ordering | requested | pending | available | down | deleting | deleted | rejected | unknown`

encryptionMode

The MAC Security (MACsec) connection encryption mode.

The valid values are `no_encrypt`, `should_encrypt`, and `must_encrypt`.

Type: String

hasLogicalRedundancy

Indicates whether the connection supports a secondary BGP peer in the same address family (IPv4/IPv6).

Type: String

Valid Values: `unknown | yes | no`

jumboFrameCapable

Indicates whether jumbo frames are supported.

Type: Boolean

lagId

The ID of the LAG.

Type: String

loaIssueTime

The time of the most recent call to DescribeLoa for this connection.

Type: Timestamp

location

The location of the connection.

Type: String

macSecCapable

Indicates whether the connection supports MAC Security (MACsec).

Type: Boolean

macSecKeys

The MAC Security (MACsec) security keys associated with the connection.

Type: Array of MacSecKey objects

ownerAccount

The ID of the AWS account that owns the connection.

Type: String

partnerInterconnectMacSecCapable

Indicates whether the interconnect hosting this connection supports MAC Security (MACsec).

Type: Boolean

partnerName

The name of the Direct Connect service provider associated with the connection.

Type: String

portEncryptionStatus

The MAC Security (MACsec) port link status of the connection.

The valid values are `Encryption Up`, which means that there is an active Connection Key Name, or `Encryption Down`.

Type: String

providerName

The name of the service provider associated with the connection.

Type: String

region

The AWS Region where the connection is located.

Type: String

tags

The tags associated with the connection.

Type: Array of Tag objects

Array Members: Minimum number of 1 item.

vlan

The ID of the VLAN.

Type: Integer

Errors

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

DirectConnectClientException

One or more parameters are not valid.

HTTP Status Code: 400

DirectConnectServerException

A server-side error occurred.

HTTP Status Code: 400

See Also

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

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

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

DeleteDirectConnectGateway

Deletes the specified Direct Connect gateway. You must first delete all virtual interfaces that are attached to the Direct Connect gateway and disassociate all virtual private gateways associated with the Direct Connect gateway.

Request Syntax

```
{
   "directConnectGatewayId": "string"
}
```

Request Parameters

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

The request accepts the following data in JSON format.

directConnectGatewayId

The ID of the Direct Connect gateway.

Type: String

Required: Yes

Response Syntax

```
{
   "directConnectGateway": {
      "amazonSideAsn": number,
      "directConnectGatewayId": "string",
      "directConnectGatewayName": "string",
      "directConnectGatewayState": "string",
      "ownerAccount": "string",
      "stateChangeError": "string",
      "tags": [
         {
            "key": "string",
            "value": "string"
```

```
            }
        ]
    }
}
```

Response Elements

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

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

directConnectGateway

The Direct Connect gateway.

Type: DirectConnectGateway object

Errors

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

DirectConnectClientException

One or more parameters are not valid.

HTTP Status Code: 400

DirectConnectServerException

A server-side error occurred.

HTTP Status Code: 400

See Also

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

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

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

DeleteDirectConnectGatewayAssociation

Deletes the association between the specified Direct Connect gateway and virtual private gateway.

We recommend that you specify the `associationID` to delete the association. Alternatively, if you own virtual gateway and a Direct Connect gateway association, you can specify the `virtualGatewayId` and `directConnectGatewayId` to delete an association.

Request Syntax

```
{
   "associationId": "string",
   "directConnectGatewayId": "string",
   "virtualGatewayId": "string"
}
```

Request Parameters

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

The request accepts the following data in JSON format.

associationId

The ID of the Direct Connect gateway association.

Type: String

Required: No

directConnectGatewayId

The ID of the Direct Connect gateway.

Type: String

Required: No

virtualGatewayId

The ID of the virtual private gateway.

Type: String

Required: No

Response Syntax

```
{
   "directConnectGatewayAssociation": {
      "allowedPrefixesToDirectConnectGateway": [
         {
            "cidr": "string"
         }
      ],
      "associatedCoreNetwork": {
         "attachmentId": "string",
         "id": "string",
         "ownerAccount": "string"
      },
      "associatedGateway": {
         "id": "string",
         "ownerAccount": "string",
         "region": "string",
         "type": "string"
      },
      "associationId": "string",
      "associationState": "string",
      "directConnectGatewayId": "string",
      "directConnectGatewayOwnerAccount": "string",
      "stateChangeError": "string",
      "virtualGatewayId": "string",
      "virtualGatewayOwnerAccount": "string",
      "virtualGatewayRegion": "string"
   }
}
```

Response Elements

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

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

directConnectGatewayAssociation

Information about the deleted association.

Type: DirectConnectGatewayAssociation object

Errors

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

DirectConnectClientException

One or more parameters are not valid.

HTTP Status Code: 400

DirectConnectServerException

A server-side error occurred.

HTTP Status Code: 400

See Also

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

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

DeleteDirectConnectGatewayAssociationProposal

Deletes the association proposal request between the specified Direct Connect gateway and virtual private gateway or transit gateway.

Request Syntax

```
{
   "proposalId": "string"
}
```

Request Parameters

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

The request accepts the following data in JSON format.

proposalId

The ID of the proposal.

Type: String

Required: Yes

Response Syntax

```
{
   "directConnectGatewayAssociationProposal": {
      "associatedGateway": {
         "id": "string",
         "ownerAccount": "string",
         "region": "string",
         "type": "string"
      },
      "directConnectGatewayId": "string",
      "directConnectGatewayOwnerAccount": "string",
      "existingAllowedPrefixesToDirectConnectGateway": [
         {
            "cidr": "string"
         }
```

```
      ],
      "proposalId": "string",
      "proposalState": "string",
      "requestedAllowedPrefixesToDirectConnectGateway": [
         {
            "cidr": "string"
         }
      ]
   }
}
```

Response Elements

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

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

directConnectGatewayAssociationProposal

The ID of the associated gateway.

Type: DirectConnectGatewayAssociationProposal object

Errors

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

DirectConnectClientException

One or more parameters are not valid.

HTTP Status Code: 400

DirectConnectServerException

A server-side error occurred.

HTTP Status Code: 400

See Also

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

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

DeleteInterconnect

Deletes the specified interconnect.

> **Note**
>
> Intended for use by Direct Connect Partners only.

Request Syntax

```
{
   "interconnectId": "string"
}
```

Request Parameters

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

The request accepts the following data in JSON format.

interconnectId

The ID of the interconnect.

Type: String

Required: Yes

Response Syntax

```
{
   "interconnectState": "string"
}
```

Response Elements

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

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

interconnectState

The state of the interconnect. The following are the possible values:

- `requested`: The initial state of an interconnect. The interconnect stays in the requested state until the Letter of Authorization (LOA) is sent to the customer.
- `pending`: The interconnect is approved, and is being initialized.
- `available`: The network link is up, and the interconnect is ready for use.
- `down`: The network link is down.
- `deleting`: The interconnect is being deleted.
- `deleted`: The interconnect is deleted.
- `unknown`: The state of the interconnect is not available.

Type: String

Valid Values: `requested | pending | available | down | deleting | deleted | unknown`

Errors

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

DirectConnectClientException

One or more parameters are not valid.

HTTP Status Code: 400

DirectConnectServerException

A server-side error occurred.

HTTP Status Code: 400

See Also

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

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

DeleteLag

Deletes the specified link aggregation group (LAG). You cannot delete a LAG if it has active virtual interfaces or hosted connections.

Request Syntax

```
{
   "lagId": "string"
}
```

Request Parameters

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

The request accepts the following data in JSON format.

lagId

The ID of the LAG.

Type: String

Required: Yes

Response Syntax

```
{
   "allowsHostedConnections": boolean,
   "awsDevice": "string",
   "awsDeviceV2": "string",
   "awsLogicalDeviceId": "string",
   "connections": [
      {
         "awsDevice": "string",
         "awsDeviceV2": "string",
         "awsLogicalDeviceId": "string",
         "bandwidth": "string",
         "connectionId": "string",
         "connectionName": "string",
         "connectionState": "string",
```

```
         "encryptionMode": "string",
         "hasLogicalRedundancy": "string",
         "jumboFrameCapable": boolean,
         "lagId": "string",
         "loaIssueTime": number,
         "location": "string",
         "macSecCapable": boolean,
         "macSecKeys": [
            {
               "ckn": "string",
               "secretARN": "string",
               "startOn": "string",
               "state": "string"
            }
         ],
         "ownerAccount": "string",
         "partnerInterconnectMacSecCapable": boolean,
         "partnerName": "string",
         "portEncryptionStatus": "string",
         "providerName": "string",
         "region": "string",
         "tags": [
            {
               "key": "string",
               "value": "string"
            }
         ],
         "vlan": number
      }
   ],
   "connectionsBandwidth": "string",
   "encryptionMode": "string",
   "hasLogicalRedundancy": "string",
   "jumboFrameCapable": boolean,
   "lagId": "string",
   "lagName": "string",
   "lagState": "string",
   "location": "string",
   "macSecCapable": boolean,
   "macSecKeys": [
      {
         "ckn": "string",
         "secretARN": "string",
         "startOn": "string",
```

```
         "state": "string"
      }
   ],
   "minimumLinks": number,
   "numberOfConnections": number,
   "ownerAccount": "string",
   "providerName": "string",
   "region": "string",
   "tags": [
      {
         "key": "string",
         "value": "string"
      }
   ]
}
```

Response Elements

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

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

allowsHostedConnections

Indicates whether the LAG can host other connections.

Type: Boolean

awsDevice

This parameter has been deprecated.

The Direct Connect endpoint that hosts the LAG.

Type: String

awsDeviceV2

The Direct Connect endpoint that hosts the LAG.

Type: String

awsLogicalDeviceId

The Direct Connect endpoint that terminates the logical connection. This device might be different than the device that terminates the physical connection.

Type: String

connections

The connections bundled by the LAG.

Type: Array of Connection objects

connectionsBandwidth

The individual bandwidth of the physical connections bundled by the LAG. The possible values are 1Gbps, 10Gbps, 100Gbps, or 400 Gbps..

Type: String

encryptionMode

The LAG MAC Security (MACsec) encryption mode.

The valid values are `no_encrypt`, `should_encrypt`, and `must_encrypt`.

Type: String

hasLogicalRedundancy

Indicates whether the LAG supports a secondary BGP peer in the same address family (IPv4/IPv6).

Type: String

Valid Values: `unknown | yes | no`

jumboFrameCapable

Indicates whether jumbo frames are supported.

Type: Boolean

lagId

The ID of the LAG.

Type: String

lagName

The name of the LAG.

Type: String

lagState

The state of the LAG. The following are the possible values:

- `requested`: The initial state of a LAG. The LAG stays in the requested state until the Letter of Authorization (LOA) is available.
- `pending`: The LAG has been approved and is being initialized.
- `available`: The network link is established and the LAG is ready for use.
- `down`: The network link is down.
- `deleting`: The LAG is being deleted.
- `deleted`: The LAG is deleted.
- `unknown`: The state of the LAG is not available.

Type: String

Valid Values: `requested | pending | available | down | deleting | deleted | unknown`

location

The location of the LAG.

Type: String

macSecCapable

Indicates whether the LAG supports MAC Security (MACsec).

Type: Boolean

macSecKeys

The MAC Security (MACsec) security keys associated with the LAG.

Type: Array of MacSecKey objects

minimumLinks

The minimum number of physical dedicated connections that must be operational for the LAG itself to be operational.

Type: Integer

numberOfConnections

The number of physical dedicated connections initially provisioned and bundled by the LAG. You can have a maximum of four connections when the port speed is 1 Gbps or 10 Gbps, or two when the port speed is 100 Gbps or 400 Gbps.

Type: Integer

ownerAccount

The ID of the AWS account that owns the LAG.

Type: String

providerName

The name of the service provider associated with the LAG.

Type: String

region

The AWS Region where the connection is located.

Type: String

tags

The tags associated with the LAG.

Type: Array of Tag objects

Array Members: Minimum number of 1 item.

Errors

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

DirectConnectClientException

One or more parameters are not valid.

HTTP Status Code: 400

DirectConnectServerException

A server-side error occurred.

HTTP Status Code: 400

See Also

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

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

DeleteVirtualInterface

Deletes a virtual interface.

Request Syntax

```
{
   "virtualInterfaceId": "string"
}
```

Request Parameters

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

The request accepts the following data in JSON format.

virtualInterfaceId

The ID of the virtual interface.

Type: String

Required: Yes

Response Syntax

```
{
   "virtualInterfaceState": "string"
}
```

Response Elements

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

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

virtualInterfaceState

The state of the virtual interface. The following are the possible values:

- `confirming`: The creation of the virtual interface is pending confirmation from the virtual interface owner. If the owner of the virtual interface is different from the owner of the connection on which it is provisioned, then the virtual interface will remain in this state until it is confirmed by the virtual interface owner.
- `verifying`: This state only applies to public virtual interfaces. Each public virtual interface needs validation before the virtual interface can be created.
- `pending`: A virtual interface is in this state from the time that it is created until the virtual interface is ready to forward traffic.
- `available`: A virtual interface that is able to forward traffic.
- down: A virtual interface that is BGP down.
- `testing`: A virtual interface is in this state immediately after calling StartBgpFailoverTest and remains in this state during the duration of the test.
- `deleting`: A virtual interface is in this state immediately after calling DeleteVirtualInterface until it can no longer forward traffic.
- `deleted`: A virtual interface that cannot forward traffic.
- `rejected`: The virtual interface owner has declined creation of the virtual interface. If a virtual interface in the `Confirming` state is deleted by the virtual interface owner, the virtual interface enters the `Rejected` state.
- unknown: The state of the virtual interface is not available.

Type: String

Valid Values: `confirming | verifying | pending | available | down | testing | deleting | deleted | rejected | unknown`

Errors

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

DirectConnectClientException

One or more parameters are not valid.

HTTP Status Code: 400

DirectConnectServerException

A server-side error occurred.

HTTP Status Code: 400

See Also

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

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

DescribeConnectionLoa

> **Note**
>
> Deprecated. Use DescribeLoa instead.

Gets the LOA-CFA for a connection.

The Letter of Authorization - Connecting Facility Assignment (LOA-CFA) is a document that your APN partner or service provider uses when establishing your cross connect to AWS at the colocation facility. For more information, see Requesting Cross Connects at Direct Connect Locations in the *Direct Connect User Guide*.

Request Syntax

```
{
   "connectionId": "string",
   "loaContentType": "string",
   "providerName": "string"
}
```

Request Parameters

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

The request accepts the following data in JSON format.

connectionId

The ID of the connection.

Type: String

Required: Yes

loaContentType

The standard media type for the LOA-CFA document. The only supported value is application/pdf.

Type: String

Valid Values: `application/pdf`

Required: No

providerName

The name of the APN partner or service provider who establishes connectivity on your behalf. If you specify this parameter, the LOA-CFA lists the provider name alongside your company name as the requester of the cross connect.

Type: String

Required: No

Response Syntax

```
{
   "loa": {
      "loaContent": blob,
      "loaContentType": "string"
   }
}
```

Response Elements

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

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

loa

The Letter of Authorization - Connecting Facility Assignment (LOA-CFA).

Type: Loa object

Errors

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

DirectConnectClientException

One or more parameters are not valid.

HTTP Status Code: 400

DirectConnectServerException

A server-side error occurred.

HTTP Status Code: 400

See Also

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

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

DescribeConnections

Displays the specified connection or all connections in this Region.

Request Syntax

```
{
   "connectionId": "string",
   "maxResults": number,
   "nextToken": "string"
}
```

Request Parameters

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

The request accepts the following data in JSON format.

connectionId

The ID of the connection.

Type: String

Required: No

maxResults

The maximum number of results to return with a single call. To retrieve the remaining results, make another call with the returned `nextToken` value.

If `MaxResults` is given a value larger than 100, only 100 results are returned.

Type: Integer

Required: No

nextToken

The token for the next page of results.

Type: String

Required: No

Response Syntax

```
{
   "connections": [
      {
         "awsDevice": "string",
         "awsDeviceV2": "string",
         "awsLogicalDeviceId": "string",
         "bandwidth": "string",
         "connectionId": "string",
         "connectionName": "string",
         "connectionState": "string",
         "encryptionMode": "string",
         "hasLogicalRedundancy": "string",
         "jumboFrameCapable": boolean,
         "lagId": "string",
         "loaIssueTime": number,
         "location": "string",
         "macSecCapable": boolean,
         "macSecKeys": [
            {
               "ckn": "string",
               "secretARN": "string",
               "startOn": "string",
               "state": "string"
            }
         ],
         "ownerAccount": "string",
         "partnerInterconnectMacSecCapable": boolean,
         "partnerName": "string",
         "portEncryptionStatus": "string",
         "providerName": "string",
         "region": "string",
         "tags": [
            {
               "key": "string",
               "value": "string"
            }
         ],
         "vlan": number
      }
   ],
   "nextToken": "string"
```

```
}
```

Response Elements

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

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

connections

The connections.

Type: Array of Connection objects

nextToken

The token to use to retrieve the next page of results. This value is `null` when there are no more results to return.

Type: String

Errors

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

DirectConnectClientException

One or more parameters are not valid.

HTTP Status Code: 400

DirectConnectServerException

A server-side error occurred.

HTTP Status Code: 400

See Also

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

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

DescribeConnectionsOnInterconnect

> **Note**
>
> Deprecated. Use DescribeHostedConnections instead.

Lists the connections that have been provisioned on the specified interconnect.

> **Note**
>
> Intended for use by Direct Connect Partners only.

Request Syntax

```
{
   "interconnectId": "string"
}
```

Request Parameters

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

The request accepts the following data in JSON format.

interconnectId

The ID of the interconnect.

Type: String

Required: Yes

Response Syntax

```
{
   "connections": [
      {
```

```
            "awsDevice": "string",
            "awsDeviceV2": "string",
            "awsLogicalDeviceId": "string",
            "bandwidth": "string",
            "connectionId": "string",
            "connectionName": "string",
            "connectionState": "string",
            "encryptionMode": "string",
            "hasLogicalRedundancy": "string",
            "jumboFrameCapable": boolean,
            "lagId": "string",
            "loaIssueTime": number,
            "location": "string",
            "macSecCapable": boolean,
            "macSecKeys": [
               {
                  "ckn": "string",
                  "secretARN": "string",
                  "startOn": "string",
                  "state": "string"
               }
            ],
            "ownerAccount": "string",
            "partnerInterconnectMacSecCapable": boolean,
            "partnerName": "string",
            "portEncryptionStatus": "string",
            "providerName": "string",
            "region": "string",
            "tags": [
               {
                  "key": "string",
                  "value": "string"
               }
            ],
            "vlan": number
         }
      ],
      "nextToken": "string"
   }
```

Response Elements

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

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

connections

The connections.

Type: Array of Connection objects

nextToken

The token to use to retrieve the next page of results. This value is `null` when there are no more results to return.

Type: String

Errors

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

DirectConnectClientException

One or more parameters are not valid.

HTTP Status Code: 400

DirectConnectServerException

A server-side error occurred.

HTTP Status Code: 400

See Also

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

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

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

DescribeCustomerMetadata

Get and view a list of customer agreements, along with their signed status and whether the customer is an NNIPartner, NNIPartnerV2, or a nonPartner.

Response Syntax

```
{
   "agreements": [
      {
         "agreementName": "string",
         "status": "string"
      }
   ],
   "nniPartnerType": "string"
}
```

Response Elements

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

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

agreements

The list of customer agreements.

Type: Array of CustomerAgreement objects

nniPartnerType

The type of network-to-network interface (NNI) partner. The partner type will be one of the following:

- V1: This partner can only allocate 50Mbps, 100Mbps, 200Mbps, 300Mbps, 400Mbps, or 500Mbps subgigabit connections.
- V2: This partner can only allocate 1GB, 2GB, 5GB, or 10GB hosted connections.
- nonPartner: The customer is not a partner.

Type: String

Valid Values: `v1 | v2 | nonPartner`

Errors

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

DirectConnectClientException

One or more parameters are not valid.

HTTP Status Code: 400

DirectConnectServerException

A server-side error occurred.

HTTP Status Code: 400

See Also

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

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

DescribeDirectConnectGatewayAssociationProposals

Describes one or more association proposals for connection between a virtual private gateway or transit gateway and a Direct Connect gateway.

Request Syntax

```
{
   "associatedGatewayId": "string",
   "directConnectGatewayId": "string",
   "maxResults": number,
   "nextToken": "string",
   "proposalId": "string"
}
```

Request Parameters

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

The request accepts the following data in JSON format.

associatedGatewayId

The ID of the associated gateway.

Type: String

Required: No

directConnectGatewayId

The ID of the Direct Connect gateway.

Type: String

Required: No

maxResults

The maximum number of results to return with a single call. To retrieve the remaining results, make another call with the returned `nextToken` value.

If `MaxResults` is given a value larger than 100, only 100 results are returned.

Type: Integer

Required: No

nextToken

The token for the next page of results.

Type: String

Required: No

proposalId

The ID of the proposal.

Type: String

Required: No

Response Syntax

```
{
   "directConnectGatewayAssociationProposals": [
      {
         "associatedGateway": {
            "id": "string",
            "ownerAccount": "string",
            "region": "string",
            "type": "string"
         },
         "directConnectGatewayId": "string",
         "directConnectGatewayOwnerAccount": "string",
         "existingAllowedPrefixesToDirectConnectGateway": [
            {
               "cidr": "string"
            }
         ],
         "proposalId": "string",
         "proposalState": "string",
         "requestedAllowedPrefixesToDirectConnectGateway": [
```

```
            {
               "cidr": "string"
            }
         ]
      }
   ],
   "nextToken": "string"
}
```

Response Elements

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

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

directConnectGatewayAssociationProposals

Describes the Direct Connect gateway association proposals.

Type: Array of DirectConnectGatewayAssociationProposal objects

nextToken

The token to use to retrieve the next page of results. This value is `null` when there are no more results to return.

Type: String

Errors

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

DirectConnectClientException

One or more parameters are not valid.

HTTP Status Code: 400

DirectConnectServerException

A server-side error occurred.

HTTP Status Code: 400

See Also

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

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

DescribeDirectConnectGatewayAssociations

Lists the associations between your Direct Connect gateways and virtual private gateways and transit gateways. You must specify one of the following:

- A Direct Connect gateway

  The response contains all virtual private gateways and transit gateways associated with the Direct Connect gateway.
- A virtual private gateway

  The response contains the Direct Connect gateway.
- A transit gateway

  The response contains the Direct Connect gateway.
- A Direct Connect gateway and a virtual private gateway

  The response contains the association between the Direct Connect gateway and virtual private gateway.
- A Direct Connect gateway and a transit gateway

  The response contains the association between the Direct Connect gateway and transit gateway.
- A Direct Connect gateway and a virtual private gateway

  The response contains the association between the Direct Connect gateway and virtual private gateway.
- A Direct Connect gateway association to a Cloud WAN core network

  The response contains the Cloud WAN core network ID that the Direct Connect gateway is associated to.

Request Syntax

```
{
   "associatedGatewayId": "string",
   "associationId": "string",
   "directConnectGatewayId": "string",
   "maxResults": number,
```

```
    "nextToken": "string",
    "virtualGatewayId": "string"
}
```

Request Parameters

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

The request accepts the following data in JSON format.

associatedGatewayId

The ID of the associated gateway.

Type: String

Required: No

associationId

The ID of the Direct Connect gateway association.

Type: String

Required: No

directConnectGatewayId

The ID of the Direct Connect gateway.

Type: String

Required: No

maxResults

The maximum number of results to return with a single call. To retrieve the remaining results, make another call with the returned `nextToken` value.

If `MaxResults` is given a value larger than 100, only 100 results are returned.

Type: Integer

Required: No

nextToken

The token provided in the previous call to retrieve the next page.

Type: String

Required: No

virtualGatewayId

The ID of the virtual private gateway or transit gateway.

Type: String

Required: No

Response Syntax

```
{
   "directConnectGatewayAssociations": [
      {
         "allowedPrefixesToDirectConnectGateway": [
            {
               "cidr": "string"
            }
         ],
         "associatedCoreNetwork": {
            "attachmentId": "string",
            "id": "string",
            "ownerAccount": "string"
         },
         "associatedGateway": {
            "id": "string",
            "ownerAccount": "string",
            "region": "string",
            "type": "string"
         },
         "associationId": "string",
         "associationState": "string",
         "directConnectGatewayId": "string",
         "directConnectGatewayOwnerAccount": "string",
         "stateChangeError": "string",
         "virtualGatewayId": "string",
         "virtualGatewayOwnerAccount": "string",
         "virtualGatewayRegion": "string"
      }
   ],
```

```
   "nextToken": "string"
}
```

Response Elements

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

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

directConnectGatewayAssociations

Information about the associations.

Type: Array of DirectConnectGatewayAssociation objects

nextToken

The token to retrieve the next page.

Type: String

Errors

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

DirectConnectClientException

One or more parameters are not valid.

HTTP Status Code: 400

DirectConnectServerException

A server-side error occurred.

HTTP Status Code: 400

See Also

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

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

DescribeDirectConnectGatewayAttachments

Lists the attachments between your Direct Connect gateways and virtual interfaces. You must specify a Direct Connect gateway, a virtual interface, or both. If you specify a Direct Connect gateway, the response contains all virtual interfaces attached to the Direct Connect gateway. If you specify a virtual interface, the response contains all Direct Connect gateways attached to the virtual interface. If you specify both, the response contains the attachment between the Direct Connect gateway and the virtual interface.

Request Syntax

```
{
   "directConnectGatewayId": "string",
   "maxResults": number,
   "nextToken": "string",
   "virtualInterfaceId": "string"
}
```

Request Parameters

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

The request accepts the following data in JSON format.

directConnectGatewayId

The ID of the Direct Connect gateway.

Type: String

Required: No

maxResults

The maximum number of results to return with a single call. To retrieve the remaining results, make another call with the returned `nextToken` value.

If `MaxResults` is given a value larger than 100, only 100 results are returned.

Type: Integer

Required: No

nextToken

The token provided in the previous call to retrieve the next page.

Type: String

Required: No

virtualInterfaceId

The ID of the virtual interface.

Type: String

Required: No

Response Syntax

```
{
   "directConnectGatewayAttachments": [
      {
         "attachmentState": "string",
         "attachmentType": "string",
         "directConnectGatewayId": "string",
         "stateChangeError": "string",
         "virtualInterfaceId": "string",
         "virtualInterfaceOwnerAccount": "string",
         "virtualInterfaceRegion": "string"
      }
   ],
   "nextToken": "string"
}
```

Response Elements

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

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

directConnectGatewayAttachments

The attachments.

Type: Array of DirectConnectGatewayAttachment objects

nextToken

The token to retrieve the next page.

Type: String

Errors

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

DirectConnectClientException

One or more parameters are not valid.

HTTP Status Code: 400

DirectConnectServerException

A server-side error occurred.

HTTP Status Code: 400

See Also

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

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

DescribeDirectConnectGateways

Lists all your Direct Connect gateways or only the specified Direct Connect gateway. Deleted Direct Connect gateways are not returned.

Request Syntax

```
{
   "directConnectGatewayId": "string",
   "maxResults": number,
   "nextToken": "string"
}
```

Request Parameters

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

The request accepts the following data in JSON format.

directConnectGatewayId

The ID of the Direct Connect gateway.

Type: String

Required: No

maxResults

The maximum number of results to return with a single call. To retrieve the remaining results, make another call with the returned `nextToken` value.

If `MaxResults` is given a value larger than 100, only 100 results are returned.

Type: Integer

Required: No

nextToken

The token provided in the previous call to retrieve the next page.

Type: String

Required: No

Response Syntax

```
{
   "directConnectGateways": [
      {
         "amazonSideAsn": number,
         "directConnectGatewayId": "string",
         "directConnectGatewayName": "string",
         "directConnectGatewayState": "string",
         "ownerAccount": "string",
         "stateChangeError": "string",
         "tags": [
            {
               "key": "string",
               "value": "string"
            }
         ]
      }
   ],
   "nextToken": "string"
}
```

Response Elements

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

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

directConnectGateways

The Direct Connect gateways.

Type: Array of DirectConnectGateway objects

nextToken

The token to retrieve the next page.

Type: String

Errors

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

DirectConnectClientException

One or more parameters are not valid.

HTTP Status Code: 400

DirectConnectServerException

A server-side error occurred.

HTTP Status Code: 400

See Also

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

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

DescribeHostedConnections

Lists the hosted connections that have been provisioned on the specified interconnect or link aggregation group (LAG).

> **Note**
>
> Intended for use by Direct Connect Partners only.

Request Syntax

```
{
   "connectionId": "string",
   "maxResults": number,
   "nextToken": "string"
}
```

Request Parameters

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

The request accepts the following data in JSON format.

connectionId

The ID of the interconnect or LAG.

Type: String

Required: Yes

maxResults

The maximum number of results to return with a single call. To retrieve the remaining results, make another call with the returned `nextToken` value.

If `MaxResults` is given a value larger than 100, only 100 results are returned.

Type: Integer

Required: No

nextToken

The token for the next page of results.

Type: String

Required: No

Response Syntax

```
{
   "connections": [
      {
         "awsDevice": "string",
         "awsDeviceV2": "string",
         "awsLogicalDeviceId": "string",
         "bandwidth": "string",
         "connectionId": "string",
         "connectionName": "string",
         "connectionState": "string",
         "encryptionMode": "string",
         "hasLogicalRedundancy": "string",
         "jumboFrameCapable": boolean,
         "lagId": "string",
         "loaIssueTime": number,
         "location": "string",
         "macSecCapable": boolean,
         "macSecKeys": [
            {
               "ckn": "string",
               "secretARN": "string",
               "startOn": "string",
               "state": "string"
            }
         ],
         "ownerAccount": "string",
         "partnerInterconnectMacSecCapable": boolean,
         "partnerName": "string",
         "portEncryptionStatus": "string",
         "providerName": "string",
         "region": "string",
         "tags": [
            {
```

```
                "key": "string",
                "value": "string"
            }
        ],
        "vlan": number
    }
  ],
  "nextToken": "string"
}
```

Response Elements

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

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

connections

The connections.

Type: Array of Connection objects

nextToken

The token to use to retrieve the next page of results. This value is `null` when there are no more results to return.

Type: String

Errors

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

DirectConnectClientException

One or more parameters are not valid.

HTTP Status Code: 400

DirectConnectServerException

A server-side error occurred.

HTTP Status Code: 400

See Also

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

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

DescribeInterconnectLoa

> **Note**
>
> Deprecated. Use DescribeLoa instead.

Gets the LOA-CFA for the specified interconnect.

The Letter of Authorization - Connecting Facility Assignment (LOA-CFA) is a document that is used when establishing your cross connect to AWS at the colocation facility. For more information, see Requesting Cross Connects at Direct Connect Locations in the *Direct Connect User Guide*.

Request Syntax

```
{
   "interconnectId": "string",
   "loaContentType": "string",
   "providerName": "string"
}
```

Request Parameters

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

The request accepts the following data in JSON format.

interconnectId

The ID of the interconnect.

Type: String

Required: Yes

loaContentType

The standard media type for the LOA-CFA document. The only supported value is application/pdf.

Type: String

Valid Values: `application/pdf`

Required: No

providerName

The name of the service provider who establishes connectivity on your behalf. If you supply this parameter, the LOA-CFA lists the provider name alongside your company name as the requester of the cross connect.

Type: String

Required: No

Response Syntax

```
{
   "loa": {
      "loaContent": blob,
      "loaContentType": "string"
   }
}
```

Response Elements

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

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

loa

The Letter of Authorization - Connecting Facility Assignment (LOA-CFA).

Type: Loa object

Errors

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

DirectConnectClientException

One or more parameters are not valid.

HTTP Status Code: 400

DirectConnectServerException

A server-side error occurred.

HTTP Status Code: 400

See Also

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

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

DescribeInterconnects

Lists the interconnects owned by the AWS account or only the specified interconnect.

Request Syntax

```
{
   "interconnectId": "string",
   "maxResults": number,
   "nextToken": "string"
}
```

Request Parameters

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

The request accepts the following data in JSON format.

interconnectId

The ID of the interconnect.

Type: String

Required: No

maxResults

The maximum number of results to return with a single call. To retrieve the remaining results, make another call with the returned `nextToken` value.

If `MaxResults` is given a value larger than 100, only 100 results are returned.

Type: Integer

Required: No

nextToken

The token for the next page of results.

Type: String

Required: No

Response Syntax

```
{
   "interconnects": [
      {
         "awsDevice": "string",
         "awsDeviceV2": "string",
         "awsLogicalDeviceId": "string",
         "bandwidth": "string",
         "encryptionMode": "string",
         "hasLogicalRedundancy": "string",
         "interconnectId": "string",
         "interconnectName": "string",
         "interconnectState": "string",
         "jumboFrameCapable": boolean,
         "lagId": "string",
         "loaIssueTime": number,
         "location": "string",
         "macSecCapable": boolean,
         "macSecKeys": [
            {
               "ckn": "string",
               "secretARN": "string",
               "startOn": "string",
               "state": "string"
            }
         ],
         "portEncryptionStatus": "string",
         "providerName": "string",
         "region": "string",
         "tags": [
            {
               "key": "string",
               "value": "string"
            }
         ]
      }
   ],
   "nextToken": "string"
}
```

Response Elements

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

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

interconnects

The interconnects.

Type: Array of Interconnect objects

nextToken

The token to use to retrieve the next page of results. This value is `null` when there are no more results to return.

Type: String

Errors

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

DirectConnectClientException

One or more parameters are not valid.

HTTP Status Code: 400

DirectConnectServerException

A server-side error occurred.

HTTP Status Code: 400

See Also

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

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

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

DescribeLags

Describes all your link aggregation groups (LAG) or the specified LAG.

Request Syntax

```
{
   "lagId": "string",
   "maxResults": number,
   "nextToken": "string"
}
```

Request Parameters

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

The request accepts the following data in JSON format.

lagId

The ID of the LAG.

Type: String

Required: No

maxResults

The maximum number of results to return with a single call. To retrieve the remaining results, make another call with the returned `nextToken` value.

If `MaxResults` is given a value larger than 100, only 100 results are returned.

Type: Integer

Required: No

nextToken

The token for the next page of results.

Type: String

Required: No

Response Syntax

```
{
   "lags": [
      {
         "allowsHostedConnections": boolean,
         "awsDevice": "string",
         "awsDeviceV2": "string",
         "awsLogicalDeviceId": "string",
         "connections": [
            {
               "awsDevice": "string",
               "awsDeviceV2": "string",
               "awsLogicalDeviceId": "string",
               "bandwidth": "string",
               "connectionId": "string",
               "connectionName": "string",
               "connectionState": "string",
               "encryptionMode": "string",
               "hasLogicalRedundancy": "string",
               "jumboFrameCapable": boolean,
               "lagId": "string",
               "loaIssueTime": number,
               "location": "string",
               "macSecCapable": boolean,
               "macSecKeys": [
                  {
                     "ckn": "string",
                     "secretARN": "string",
                     "startOn": "string",
                     "state": "string"
                  }
               ],
               "ownerAccount": "string",
               "partnerInterconnectMacSecCapable": boolean,
               "partnerName": "string",
               "portEncryptionStatus": "string",
               "providerName": "string",
               "region": "string",
               "tags": [
                  {
                     "key": "string",
                     "value": "string"
```

```
                    }
                ],
                "vlan": number
            }
        ],
        "connectionsBandwidth": "string",
        "encryptionMode": "string",
        "hasLogicalRedundancy": "string",
        "jumboFrameCapable": boolean,
        "lagId": "string",
        "lagName": "string",
        "lagState": "string",
        "location": "string",
        "macSecCapable": boolean,
        "macSecKeys": [
            {
                "ckn": "string",
                "secretARN": "string",
                "startOn": "string",
                "state": "string"
            }
        ],
        "minimumLinks": number,
        "numberOfConnections": number,
        "ownerAccount": "string",
        "providerName": "string",
        "region": "string",
        "tags": [
            {
                "key": "string",
                "value": "string"
            }
        ]
    }
],
"nextToken": "string"
}
```

Response Elements

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

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

lags

The LAGs.

Type: Array of Lag objects

nextToken

The token to use to retrieve the next page of results. This value is `null` when there are no more results to return.

Type: String

Errors

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

DirectConnectClientException

One or more parameters are not valid.

HTTP Status Code: 400

DirectConnectServerException

A server-side error occurred.

HTTP Status Code: 400

See Also

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

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

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

DescribeLoa

Gets the LOA-CFA for a connection, interconnect, or link aggregation group (LAG).

The Letter of Authorization - Connecting Facility Assignment (LOA-CFA) is a document that is used when establishing your cross connect to AWS at the colocation facility. For more information, see Requesting Cross Connects at Direct Connect Locations in the *Direct Connect User Guide*.

Request Syntax

```
{
   "connectionId": "string",
   "loaContentType": "string",
   "providerName": "string"
}
```

Request Parameters

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

The request accepts the following data in JSON format.

connectionId

The ID of a connection, LAG, or interconnect.

Type: String

Required: Yes

loaContentType

The standard media type for the LOA-CFA document. The only supported value is application/pdf.

Type: String

Valid Values: `application/pdf`

Required: No

providerName

The name of the service provider who establishes connectivity on your behalf. If you specify this parameter, the LOA-CFA lists the provider name alongside your company name as the requester of the cross connect.

Type: String

Required: No

Response Syntax

```
{
   "loaContent": blob,
   "loaContentType": "string"
}
```

Response Elements

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

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

loaContent

The binary contents of the LOA-CFA document.

Type: Base64-encoded binary data object

loaContentType

The standard media type for the LOA-CFA document. The only supported value is application/pdf.

Type: String

Valid Values: `application/pdf`

Errors

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

DirectConnectClientException

One or more parameters are not valid.

HTTP Status Code: 400

DirectConnectServerException

A server-side error occurred.

HTTP Status Code: 400

See Also

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

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

DescribeLocations

Lists the Direct Connect locations in the current AWS Region. These are the locations that can be selected when calling CreateConnection or CreateInterconnect.

Response Syntax

```
{
   "locations": [
      {
         "availableMacSecPortSpeeds": [ "string" ],
         "availablePortSpeeds": [ "string" ],
         "availableProviders": [ "string" ],
         "locationCode": "string",
         "locationName": "string",
         "region": "string"
      }
   ]
}
```

Response Elements

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

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

locations

The locations.

Type: Array of Location objects

Errors

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

DirectConnectClientException

One or more parameters are not valid.

HTTP Status Code: 400

DirectConnectServerException

A server-side error occurred.

HTTP Status Code: 400

See Also

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

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

DescribeRouterConfiguration

Details about the router.

Request Syntax

```
{
   "routerTypeIdentifier": "string",
   "virtualInterfaceId": "string"
}
```

Request Parameters

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

The request accepts the following data in JSON format.

routerTypeIdentifier

Identifies the router by a combination of vendor, platform, and software version. For example, `CiscoSystemsInc-2900SeriesRouters-IOS124`.

Type: String

Required: No

virtualInterfaceId

The ID of the virtual interface.

Type: String

Required: Yes

Response Syntax

```
{
   "customerRouterConfig": "string",
   "router": {
      "platform": "string",
```

```
        "routerTypeIdentifier": "string",
        "software": "string",
        "vendor": "string",
        "xsltTemplateName": "string",
        "xsltTemplateNameForMacSec": "string"
    },
    "virtualInterfaceId": "string",
    "virtualInterfaceName": "string"
}
```

Response Elements

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

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

customerRouterConfig

The customer router configuration.

Type: String

router

The details about the router.

Type: RouterType object

virtualInterfaceId

The ID assigned to the virtual interface.

Type: String

virtualInterfaceName

Provides the details about a virtual interface's router.

Type: String

Errors

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

DirectConnectClientException

One or more parameters are not valid.

HTTP Status Code: 400

DirectConnectServerException

A server-side error occurred.

HTTP Status Code: 400

See Also

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

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

DescribeTags

Describes the tags associated with the specified Direct Connect resources.

Request Syntax

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

Request Parameters

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

The request accepts the following data in JSON format.

resourceArns

The Amazon Resource Names (ARNs) of the resources.

Type: Array of strings

Required: Yes

Response Syntax

```
{
   "resourceTags": [
      {
         "resourceArn": "string",
         "tags": [
            {
               "key": "string",
               "value": "string"
            }
         ]
      }
   ]
}
```

Response Elements

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

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

resourceTags

Information about the tags.

Type: Array of ResourceTag objects

Errors

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

DirectConnectClientException

One or more parameters are not valid.

HTTP Status Code: 400

DirectConnectServerException

A server-side error occurred.

HTTP Status Code: 400

See Also

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

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

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

DescribeVirtualGateways

> **Note**
>
> Deprecated. Use `DescribeVpnGateways` instead. See DescribeVPNGateways in the *Amazon Elastic Compute Cloud API Reference*.

Lists the virtual private gateways owned by the AWS account.

You can create one or more Direct Connect private virtual interfaces linked to a virtual private gateway.

Response Syntax

```
{
   "virtualGateways": [
      {
         "virtualGatewayId": "string",
         "virtualGatewayState": "string"
      }
   ]
}
```

Response Elements

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

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

virtualGateways

The virtual private gateways.

Type: Array of VirtualGateway objects

Errors

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

DirectConnectClientException

One or more parameters are not valid.

HTTP Status Code: 400

DirectConnectServerException

A server-side error occurred.

HTTP Status Code: 400

See Also

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

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

DescribeVirtualInterfaces

Displays all virtual interfaces for an AWS account. Virtual interfaces deleted fewer than 15 minutes before you make the request are also returned. If you specify a connection ID, only the virtual interfaces associated with the connection are returned. If you specify a virtual interface ID, then only a single virtual interface is returned.

A virtual interface (VLAN) transmits the traffic between the Direct Connect location and the customer network.

- If you're using an `asn`, the response includes the ASN value in both the `asn` and `asnLong` fields.
- If you're using `asnLong`, the response returns a value of `0` (zero) for the `asn` attribute because it exceeds the highest ASN value of 2,147,483,647 that it can support

Request Syntax

```
{
   "connectionId": "string",
   "maxResults": number,
   "nextToken": "string",
   "virtualInterfaceId": "string"
}
```

Request Parameters

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

The request accepts the following data in JSON format.

connectionId

The ID of the connection.

Type: String

Required: No

maxResults

The maximum number of results to return with a single call. To retrieve the remaining results, make another call with the returned `nextToken` value.

If `MaxResults` is given a value larger than 100, only 100 results are returned.

Type: Integer

Required: No

nextToken

The token for the next page of results.

Type: String

Required: No

virtualInterfaceId

The ID of the virtual interface.

Type: String

Required: No

Response Syntax

```
{
   "nextToken": "string",
   "virtualInterfaces": [
      {
         "addressFamily": "string",
         "amazonAddress": "string",
         "amazonSideAsn": number,
         "asn": number,
         "asnLong": number,
         "authKey": "string",
         "awsDeviceV2": "string",
         "awsLogicalDeviceId": "string",
         "bgpPeers": [
            {
               "addressFamily": "string",
               "amazonAddress": "string",
               "asn": number,
               "asnLong": number,
               "authKey": "string",
               "awsDeviceV2": "string",
```

```
                "awsLogicalDeviceId": "string",
                "bgpPeerId": "string",
                "bgpPeerState": "string",
                "bgpStatus": "string",
                "customerAddress": "string"
             }
          ],
          "connectionId": "string",
          "customerAddress": "string",
          "customerRouterConfig": "string",
          "directConnectGatewayId": "string",
          "jumboFrameCapable": boolean,
          "location": "string",
          "mtu": number,
          "ownerAccount": "string",
          "region": "string",
          "routeFilterPrefixes": [
             {
                "cidr": "string"
             }
          ],
          "siteLinkEnabled": boolean,
          "tags": [
             {
                "key": "string",
                "value": "string"
             }
          ],
          "virtualGatewayId": "string",
          "virtualInterfaceId": "string",
          "virtualInterfaceName": "string",
          "virtualInterfaceState": "string",
          "virtualInterfaceType": "string",
          "vlan": number
       }
    ]
}
```

Response Elements

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

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

nextToken

The token to use to retrieve the next page of results. This value is `null` when there are no more results to return.

Type: String

virtualInterfaces

The virtual interfaces

Type: Array of VirtualInterface objects

Errors

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

DirectConnectClientException

One or more parameters are not valid.

HTTP Status Code: 400

DirectConnectServerException

A server-side error occurred.

HTTP Status Code: 400

See Also

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

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

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

DisassociateConnectionFromLag

Disassociates a connection from a link aggregation group (LAG). The connection is interrupted and re-established as a standalone connection (the connection is not deleted; to delete the connection, use the DeleteConnection request). If the LAG has associated virtual interfaces or hosted connections, they remain associated with the LAG. A disassociated connection owned by an Direct Connect Partner is automatically converted to an interconnect.

If disassociating the connection would cause the LAG to fall below its setting for minimum number of operational connections, the request fails, except when it's the last member of the LAG. If all connections are disassociated, the LAG continues to exist as an empty LAG with no physical connections.

Request Syntax

```
{
   "connectionId": "string",
   "lagId": "string"
}
```

Request Parameters

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

The request accepts the following data in JSON format.

connectionId

The ID of the connection.

Type: String

Required: Yes

lagId

The ID of the LAG.

Type: String

Required: Yes

Response Syntax

```
{
   "awsDevice": "string",
   "awsDeviceV2": "string",
   "awsLogicalDeviceId": "string",
   "bandwidth": "string",
   "connectionId": "string",
   "connectionName": "string",
   "connectionState": "string",
   "encryptionMode": "string",
   "hasLogicalRedundancy": "string",
   "jumboFrameCapable": boolean,
   "lagId": "string",
   "loaIssueTime": number,
   "location": "string",
   "macSecCapable": boolean,
   "macSecKeys": [
      {
         "ckn": "string",
         "secretARN": "string",
         "startOn": "string",
         "state": "string"
      }
   ],
   "ownerAccount": "string",
   "partnerInterconnectMacSecCapable": boolean,
   "partnerName": "string",
   "portEncryptionStatus": "string",
   "providerName": "string",
   "region": "string",
   "tags": [
      {
         "key": "string",
         "value": "string"
      }
   ],
   "vlan": number
}
```

Response Elements

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

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

awsDevice

This parameter has been deprecated.

The Direct Connect endpoint on which the physical connection terminates.

Type: String

awsDeviceV2

The Direct Connect endpoint that terminates the physical connection.

Type: String

awsLogicalDeviceId

The Direct Connect endpoint that terminates the logical connection. This device might be different than the device that terminates the physical connection.

Type: String

bandwidth

The bandwidth of the connection.

Type: String

connectionId

The ID of the connection.

Type: String

connectionName

The name of the connection.

Type: String

connectionState

The state of the connection. The following are the possible values:

- `ordering`: The initial state of a hosted connection provisioned on an interconnect. The connection stays in the ordering state until the owner of the hosted connection confirms or declines the connection order.

- `requested`: The initial state of a standard connection. The connection stays in the requested state until the Letter of Authorization (LOA) is sent to the customer.
- `pending`: The connection has been approved and is being initialized.
- `available`: The network link is up and the connection is ready for use.
- down: The network link is down.
- `deleting`: The connection is being deleted.
- `deleted`: The connection has been deleted.
- `rejected`: A hosted connection in the `ordering` state enters the `rejected` state if it is deleted by the customer.
- unknown: The state of the connection is not available.

Type: String

Valid Values: `ordering | requested | pending | available | down | deleting | deleted | rejected | unknown`

encryptionMode

The MAC Security (MACsec) connection encryption mode.

The valid values are `no_encrypt`, `should_encrypt`, and `must_encrypt`.

Type: String

hasLogicalRedundancy

Indicates whether the connection supports a secondary BGP peer in the same address family (IPv4/IPv6).

Type: String

Valid Values: `unknown | yes | no`

jumboFrameCapable

Indicates whether jumbo frames are supported.

Type: Boolean

lagId

The ID of the LAG.

Type: String

loaIssueTime

The time of the most recent call to DescribeLoa for this connection.

Type: Timestamp

location

The location of the connection.

Type: String

macSecCapable

Indicates whether the connection supports MAC Security (MACsec).

Type: Boolean

macSecKeys

The MAC Security (MACsec) security keys associated with the connection.

Type: Array of MacSecKey objects

ownerAccount

The ID of the AWS account that owns the connection.

Type: String

partnerInterconnectMacSecCapable

Indicates whether the interconnect hosting this connection supports MAC Security (MACsec).

Type: Boolean

partnerName

The name of the Direct Connect service provider associated with the connection.

Type: String

portEncryptionStatus

The MAC Security (MACsec) port link status of the connection.

The valid values are `Encryption Up`, which means that there is an active Connection Key Name, or `Encryption Down`.

Type: String

providerName

The name of the service provider associated with the connection.

Type: String

region

The AWS Region where the connection is located.

Type: String

tags

The tags associated with the connection.

Type: Array of Tag objects

Array Members: Minimum number of 1 item.

vlan

The ID of the VLAN.

Type: Integer

Errors

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

DirectConnectClientException

One or more parameters are not valid.

HTTP Status Code: 400

DirectConnectServerException

A server-side error occurred.

HTTP Status Code: 400

See Also

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

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

DisassociateMacSecKey

Removes the association between a MAC Security (MACsec) security key and a Direct Connect connection.

Request Syntax

```
{
   "connectionId": "string",
   "secretARN": "string"
}
```

Request Parameters

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

The request accepts the following data in JSON format.

connectionId

The ID of the dedicated connection (dxcon-xxxx), interconnect (dxcon-xxxx), or LAG (dxlag-xxxx).

You can use DescribeConnections, DescribeInterconnects, or DescribeLags to retrieve connection ID.

Type: String

Required: Yes

secretARN

The Amazon Resource Name (ARN) of the MAC Security (MACsec) secret key.

You can use DescribeConnections to retrieve the ARN of the MAC Security (MACsec) secret key.

Type: String

Required: Yes

Response Syntax

```
{
```

```
   "connectionId": "string",
   "macSecKeys": [
      {
         "ckn": "string",
         "secretARN": "string",
         "startOn": "string",
         "state": "string"
      }
   ]
}
```

Response Elements

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

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

connectionId

The ID of the dedicated connection (dxcon-xxxx), interconnect (dxcon-xxxx), or LAG (dxlag-xxxx).

Type: String

macSecKeys

The MAC Security (MACsec) security keys no longer associated with the connection.

Type: Array of MacSecKey objects

Errors

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

DirectConnectClientException

One or more parameters are not valid.

HTTP Status Code: 400

DirectConnectServerException

A server-side error occurred.

HTTP Status Code: 400

See Also

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

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

ListVirtualInterfaceTestHistory

Lists the virtual interface failover test history.

Request Syntax

```
{
   "bgpPeers": [ "string" ],
   "maxResults": number,
   "nextToken": "string",
   "status": "string",
   "testId": "string",
   "virtualInterfaceId": "string"
}
```

Request Parameters

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

The request accepts the following data in JSON format.

bgpPeers

The BGP peers that were placed in the DOWN state during the virtual interface failover test.

Type: Array of strings

Required: No

maxResults

The maximum number of results to return with a single call. To retrieve the remaining results, make another call with the returned `nextToken` value.

If `MaxResults` is given a value larger than 100, only 100 results are returned.

Type: Integer

Required: No

nextToken

The token for the next page of results.

Type: String

Required: No

status

The status of the virtual interface failover test.

Type: String

Required: No

testId

The ID of the virtual interface failover test.

Type: String

Required: No

virtualInterfaceId

The ID of the virtual interface that was tested.

Type: String

Required: No

Response Syntax

```
{
   "nextToken": "string",
   "virtualInterfaceTestHistory": [
      {
         "bgpPeers": [ "string" ],
         "endTime": number,
         "ownerAccount": "string",
         "startTime": number,
         "status": "string",
         "testDurationInMinutes": number,
         "testId": "string",
         "virtualInterfaceId": "string"
      }
   ]
```

```
}
```

Response Elements

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

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

nextToken

The token to use to retrieve the next page of results. This value is `null` when there are no more results to return.

Type: String

virtualInterfaceTestHistory

The ID of the tested virtual interface.

Type: Array of VirtualInterfaceTestHistory objects

Errors

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

DirectConnectClientException

One or more parameters are not valid.

HTTP Status Code: 400

DirectConnectServerException

A server-side error occurred.

HTTP Status Code: 400

See Also

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

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

StartBgpFailoverTest

Starts the virtual interface failover test that verifies your configuration meets your resiliency requirements by placing the BGP peering session in the DOWN state. You can then send traffic to verify that there are no outages.

You can run the test on public, private, transit, and hosted virtual interfaces.

You can use ListVirtualInterfaceTestHistory to view the virtual interface test history.

If you need to stop the test before the test interval completes, use StopBgpFailoverTest.

Request Syntax

```
{
   "bgpPeers": [ "string" ],
   "testDurationInMinutes": number,
   "virtualInterfaceId": "string"
}
```

Request Parameters

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

The request accepts the following data in JSON format.

bgpPeers

The BGP peers to place in the DOWN state.

Type: Array of strings

Required: No

testDurationInMinutes

The time in minutes that the virtual interface failover test will last.

Maximum value: 4,320 minutes (72 hours).

Default: 180 minutes (3 hours).

Type: Integer

Required: No

virtualInterfaceId

The ID of the virtual interface you want to test.

Type: String

Required: Yes

Response Syntax

```
{
   "virtualInterfaceTest": {
      "bgpPeers": [ "string" ],
      "endTime": number,
      "ownerAccount": "string",
      "startTime": number,
      "status": "string",
      "testDurationInMinutes": number,
      "testId": "string",
      "virtualInterfaceId": "string"
   }
}
```

Response Elements

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

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

virtualInterfaceTest

Information about the virtual interface failover test.

Type: VirtualInterfaceTestHistory object

Errors

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

DirectConnectClientException

One or more parameters are not valid.

HTTP Status Code: 400

DirectConnectServerException

A server-side error occurred.

HTTP Status Code: 400

See Also

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

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

StopBgpFailoverTest

Stops the virtual interface failover test.

Request Syntax

```
{
   "virtualInterfaceId": "string"
}
```

Request Parameters

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

The request accepts the following data in JSON format.

virtualInterfaceId

The ID of the virtual interface you no longer want to test.

Type: String

Required: Yes

Response Syntax

```
{
   "virtualInterfaceTest": {
      "bgpPeers": [ "string" ],
      "endTime": number,
      "ownerAccount": "string",
      "startTime": number,
      "status": "string",
      "testDurationInMinutes": number,
      "testId": "string",
      "virtualInterfaceId": "string"
   }
}
```

Response Elements

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

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

virtualInterfaceTest

Information about the virtual interface failover test.

Type: VirtualInterfaceTestHistory object

Errors

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

DirectConnectClientException

One or more parameters are not valid.

HTTP Status Code: 400

DirectConnectServerException

A server-side error occurred.

HTTP Status Code: 400

See Also

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

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

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

TagResource

Adds the specified tags to the specified Direct Connect resource. Each resource can have a maximum of 50 tags.

Each tag consists of a key and an optional value. If a tag with the same key is already associated with the resource, this action updates its value.

Request Syntax

```
{
   "resourceArn": "string",
   "tags": [
      {
         "key": "string",
         "value": "string"
      }
   ]
}
```

Request Parameters

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

The request accepts the following data in JSON format.

resourceArn

The Amazon Resource Name (ARN) of the resource.

Type: String

Required: Yes

tags

The tags to add.

Type: Array of Tag objects

Array Members: Minimum number of 1 item.

Required: Yes

Response Elements

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

Errors

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

DirectConnectClientException

One or more parameters are not valid.

HTTP Status Code: 400

DirectConnectServerException

A server-side error occurred.

HTTP Status Code: 400

DuplicateTagKeysException

A tag key was specified more than once.

HTTP Status Code: 400

TooManyTagsException

You have reached the limit on the number of tags that can be assigned.

HTTP Status Code: 400

See Also

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

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

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

UntagResource

Removes one or more tags from the specified Direct Connect resource.

Request Syntax

```
{
   "resourceArn": "string",
   "tagKeys": [ "string" ]
}
```

Request Parameters

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

The request accepts the following data in JSON format.

resourceArn

The Amazon Resource Name (ARN) of the resource.

Type: String

Required: Yes

tagKeys

The tag keys of the tags to remove.

Type: Array of strings

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

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

Required: Yes

Response Elements

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

Errors

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

DirectConnectClientException

One or more parameters are not valid.

HTTP Status Code: 400

DirectConnectServerException

A server-side error occurred.

HTTP Status Code: 400

See Also

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

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

UpdateConnection

Updates the Direct Connect connection configuration.

You can update the following parameters for a connection:

- The connection name
- The connection's MAC Security (MACsec) encryption mode.

Request Syntax

```
{
   "connectionId": "string",
   "connectionName": "string",
   "encryptionMode": "string"
}
```

Request Parameters

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

The request accepts the following data in JSON format.

connectionId

The ID of the connection.

You can use DescribeConnections to retrieve the connection ID.

Type: String

Required: Yes

connectionName

The name of the connection.

Type: String

Required: No

encryptionMode

The connection MAC Security (MACsec) encryption mode.

The valid values are `no_encrypt`, `should_encrypt`, and `must_encrypt`.

Type: String

Required: No

Response Syntax

```
{
   "awsDevice": "string",
   "awsDeviceV2": "string",
   "awsLogicalDeviceId": "string",
   "bandwidth": "string",
   "connectionId": "string",
   "connectionName": "string",
   "connectionState": "string",
   "encryptionMode": "string",
   "hasLogicalRedundancy": "string",
   "jumboFrameCapable": boolean,
   "lagId": "string",
   "loaIssueTime": number,
   "location": "string",
   "macSecCapable": boolean,
   "macSecKeys": [
      {
         "ckn": "string",
         "secretARN": "string",
         "startOn": "string",
         "state": "string"
      }
   ],
   "ownerAccount": "string",
   "partnerInterconnectMacSecCapable": boolean,
   "partnerName": "string",
   "portEncryptionStatus": "string",
   "providerName": "string",
   "region": "string",
   "tags": [
      {
```

```
            "key": "string",
            "value": "string"
        }
    ],
    "vlan": number
}
```

Response Elements

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

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

awsDevice

This parameter has been deprecated.

The Direct Connect endpoint on which the physical connection terminates.

Type: String

awsDeviceV2

The Direct Connect endpoint that terminates the physical connection.

Type: String

awsLogicalDeviceId

The Direct Connect endpoint that terminates the logical connection. This device might be different than the device that terminates the physical connection.

Type: String

bandwidth

The bandwidth of the connection.

Type: String

connectionId

The ID of the connection.

Type: String

connectionName

The name of the connection.

Type: String

connectionState

The state of the connection. The following are the possible values:

- `ordering`: The initial state of a hosted connection provisioned on an interconnect. The connection stays in the ordering state until the owner of the hosted connection confirms or declines the connection order.
- `requested`: The initial state of a standard connection. The connection stays in the requested state until the Letter of Authorization (LOA) is sent to the customer.
- `pending`: The connection has been approved and is being initialized.
- `available`: The network link is up and the connection is ready for use.
- down: The network link is down.
- `deleting`: The connection is being deleted.
- `deleted`: The connection has been deleted.
- `rejected`: A hosted connection in the `ordering` state enters the `rejected` state if it is deleted by the customer.
- unknown: The state of the connection is not available.

Type: String

Valid Values: `ordering | requested | pending | available | down | deleting | deleted | rejected | unknown`

encryptionMode

The MAC Security (MACsec) connection encryption mode.

The valid values are `no_encrypt`, `should_encrypt`, and `must_encrypt`.

Type: String

hasLogicalRedundancy

Indicates whether the connection supports a secondary BGP peer in the same address family (IPv4/IPv6).

Type: String

Valid Values: `unknown | yes | no`

jumboFrameCapable

Indicates whether jumbo frames are supported.

Type: Boolean

lagId

The ID of the LAG.

Type: String

loaIssueTime

The time of the most recent call to DescribeLoa for this connection.

Type: Timestamp

location

The location of the connection.

Type: String

macSecCapable

Indicates whether the connection supports MAC Security (MACsec).

Type: Boolean

macSecKeys

The MAC Security (MACsec) security keys associated with the connection.

Type: Array of MacSecKey objects

ownerAccount

The ID of the AWS account that owns the connection.

Type: String

partnerInterconnectMacSecCapable

Indicates whether the interconnect hosting this connection supports MAC Security (MACsec).

Type: Boolean

partnerName

The name of the Direct Connect service provider associated with the connection.

Type: String

portEncryptionStatus

The MAC Security (MACsec) port link status of the connection.

The valid values are `Encryption Up`, which means that there is an active Connection Key Name, or `Encryption Down`.

Type: String

providerName

The name of the service provider associated with the connection.

Type: String

region

The AWS Region where the connection is located.

Type: String

tags

The tags associated with the connection.

Type: Array of Tag objects

Array Members: Minimum number of 1 item.

vlan

The ID of the VLAN.

Type: Integer

Errors

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

DirectConnectClientException

One or more parameters are not valid.

HTTP Status Code: 400

DirectConnectServerException

A server-side error occurred.

HTTP Status Code: 400

See Also

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

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

UpdateDirectConnectGateway

Updates the name of a current Direct Connect gateway.

Request Syntax

```
{
   "directConnectGatewayId": "string",
   "newDirectConnectGatewayName": "string"
}
```

Request Parameters

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

The request accepts the following data in JSON format.

directConnectGatewayId

The ID of the Direct Connect gateway to update.

Type: String

Required: Yes

newDirectConnectGatewayName

The new name for the Direct Connect gateway.

Type: String

Required: Yes

Response Syntax

```
{
   "directConnectGateway": {
      "amazonSideAsn": number,
      "directConnectGatewayId": "string",
      "directConnectGatewayName": "string",
      "directConnectGatewayState": "string",
      "ownerAccount": "string",
      "stateChangeError": "string",
```

```
      "tags": [
         {
            "key": "string",
            "value": "string"
         }
      ]
   }
}
```

Response Elements

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

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

directConnectGateway

Informaiton about a Direct Connect gateway, which enables you to connect virtual interfaces and virtual private gateways or transit gateways.

Type: DirectConnectGateway object

Errors

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

DirectConnectClientException

One or more parameters are not valid.

HTTP Status Code: 400

DirectConnectServerException

A server-side error occurred.

HTTP Status Code: 400

See Also

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

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

UpdateDirectConnectGatewayAssociation

Updates the specified attributes of the Direct Connect gateway association.

Add or remove prefixes from the association.

Request Syntax

```
{
   "addAllowedPrefixesToDirectConnectGateway": [
      {
         "cidr": "string"
      }
   ],
   "associationId": "string",
   "removeAllowedPrefixesToDirectConnectGateway": [
      {
         "cidr": "string"
      }
   ]
}
```

Request Parameters

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

The request accepts the following data in JSON format.

addAllowedPrefixesToDirectConnectGateway

The Amazon VPC prefixes to advertise to the Direct Connect gateway.

Type: Array of RouteFilterPrefix objects

Required: No

associationId

The ID of the Direct Connect gateway association.

Type: String

Required: No

removeAllowedPrefixesToDirectConnectGateway

The Amazon VPC prefixes to no longer advertise to the Direct Connect gateway.

Type: Array of RouteFilterPrefix objects

Required: No

Response Syntax

```
{
   "directConnectGatewayAssociation": {
      "allowedPrefixesToDirectConnectGateway": [
         {
            "cidr": "string"
         }
      ],
      "associatedCoreNetwork": {
         "attachmentId": "string",
         "id": "string",
         "ownerAccount": "string"
      },
      "associatedGateway": {
         "id": "string",
         "ownerAccount": "string",
         "region": "string",
         "type": "string"
      },
      "associationId": "string",
      "associationState": "string",
      "directConnectGatewayId": "string",
      "directConnectGatewayOwnerAccount": "string",
      "stateChangeError": "string",
      "virtualGatewayId": "string",
      "virtualGatewayOwnerAccount": "string",
      "virtualGatewayRegion": "string"
   }
}
```

Response Elements

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

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

directConnectGatewayAssociation

Information about an association between a Direct Connect gateway and a virtual private gateway or transit gateway.

Type: DirectConnectGatewayAssociation object

Errors

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

DirectConnectClientException

One or more parameters are not valid.

HTTP Status Code: 400

DirectConnectServerException

A server-side error occurred.

HTTP Status Code: 400

See Also

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

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

- AWS SDK for Python
- AWS SDK for Ruby V3

UpdateLag

Updates the attributes of the specified link aggregation group (LAG).

You can update the following LAG attributes:

- The name of the LAG.
- The value for the minimum number of connections that must be operational for the LAG itself to be operational.
- The LAG's MACsec encryption mode.

  AWS assigns this value to each connection which is part of the LAG.
- The tags

> **Note**
>
> If you adjust the threshold value for the minimum number of operational connections, ensure that the new value does not cause the LAG to fall below the threshold and become non-operational.

Request Syntax

```
{
   "encryptionMode": "string",
   "lagId": "string",
   "lagName": "string",
   "minimumLinks": number
}
```

Request Parameters

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

The request accepts the following data in JSON format.

encryptionMode

The LAG MAC Security (MACsec) encryption mode.

AWS applies the value to all connections which are part of the LAG.

Type: String

Required: No

lagId

The ID of the LAG.

Type: String

Required: Yes

lagName

The name of the LAG.

Type: String

Required: No

minimumLinks

The minimum number of physical connections that must be operational for the LAG itself to be operational.

Type: Integer

Required: No

Response Syntax

```
{
   "allowsHostedConnections": boolean,
   "awsDevice": "string",
   "awsDeviceV2": "string",
   "awsLogicalDeviceId": "string",
   "connections": [
      {
         "awsDevice": "string",
         "awsDeviceV2": "string",
         "awsLogicalDeviceId": "string",
         "bandwidth": "string",
         "connectionId": "string",
         "connectionName": "string",
```

```
         "connectionState": "string",
         "encryptionMode": "string",
         "hasLogicalRedundancy": "string",
         "jumboFrameCapable": boolean,
         "lagId": "string",
         "loaIssueTime": number,
         "location": "string",
         "macSecCapable": boolean,
         "macSecKeys": [
            {
               "ckn": "string",
               "secretARN": "string",
               "startOn": "string",
               "state": "string"
            }
         ],
         "ownerAccount": "string",
         "partnerInterconnectMacSecCapable": boolean,
         "partnerName": "string",
         "portEncryptionStatus": "string",
         "providerName": "string",
         "region": "string",
         "tags": [
            {
               "key": "string",
               "value": "string"
            }
         ],
         "vlan": number
      }
   ],
   "connectionsBandwidth": "string",
   "encryptionMode": "string",
   "hasLogicalRedundancy": "string",
   "jumboFrameCapable": boolean,
   "lagId": "string",
   "lagName": "string",
   "lagState": "string",
   "location": "string",
   "macSecCapable": boolean,
   "macSecKeys": [
      {
         "ckn": "string",
         "secretARN": "string",
```

```
            "startOn": "string",
            "state": "string"
         }
      ],
      "minimumLinks": number,
      "numberOfConnections": number,
      "ownerAccount": "string",
      "providerName": "string",
      "region": "string",
      "tags": [
         {
            "key": "string",
            "value": "string"
         }
      ]
}
```

Response Elements

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

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

allowsHostedConnections

Indicates whether the LAG can host other connections.

Type: Boolean

awsDevice

This parameter has been deprecated.

The Direct Connect endpoint that hosts the LAG.

Type: String

awsDeviceV2

The Direct Connect endpoint that hosts the LAG.

Type: String

awsLogicalDeviceId

The Direct Connect endpoint that terminates the logical connection. This device might be different than the device that terminates the physical connection.

Type: String

connections

The connections bundled by the LAG.

Type: Array of Connection objects

connectionsBandwidth

The individual bandwidth of the physical connections bundled by the LAG. The possible values are 1Gbps, 10Gbps, 100Gbps, or 400 Gbps..

Type: String

encryptionMode

The LAG MAC Security (MACsec) encryption mode.

The valid values are `no_encrypt`, `should_encrypt`, and `must_encrypt`.

Type: String

hasLogicalRedundancy

Indicates whether the LAG supports a secondary BGP peer in the same address family (IPv4/IPv6).

Type: String

Valid Values: `unknown | yes | no`

jumboFrameCapable

Indicates whether jumbo frames are supported.

Type: Boolean

lagId

The ID of the LAG.

Type: String

lagName

The name of the LAG.

Type: String

lagState

The state of the LAG. The following are the possible values:

- `requested`: The initial state of a LAG. The LAG stays in the requested state until the Letter of Authorization (LOA) is available.
- `pending`: The LAG has been approved and is being initialized.
- `available`: The network link is established and the LAG is ready for use.
- down: The network link is down.
- `deleting`: The LAG is being deleted.
- `deleted`: The LAG is deleted.
- unknown: The state of the LAG is not available.

Type: String

Valid Values: `requested | pending | available | down | deleting | deleted |` unknown

location

The location of the LAG.

Type: String

macSecCapable

Indicates whether the LAG supports MAC Security (MACsec).

Type: Boolean

macSecKeys

The MAC Security (MACsec) security keys associated with the LAG.

Type: Array of MacSecKey objects

minimumLinks

The minimum number of physical dedicated connections that must be operational for the LAG itself to be operational.

Type: Integer

numberOfConnections

The number of physical dedicated connections initially provisioned and bundled by the LAG. You can have a maximum of four connections when the port speed is 1 Gbps or 10 Gbps, or two when the port speed is 100 Gbps or 400 Gbps.

Type: Integer

ownerAccount

The ID of the AWS account that owns the LAG.

Type: String

providerName

The name of the service provider associated with the LAG.

Type: String

region

The AWS Region where the connection is located.

Type: String

tags

The tags associated with the LAG.

Type: Array of Tag objects

Array Members: Minimum number of 1 item.

Errors

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

DirectConnectClientException

One or more parameters are not valid.

HTTP Status Code: 400

DirectConnectServerException

A server-side error occurred.

HTTP Status Code: 400

See Also

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

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

UpdateVirtualInterfaceAttributes

Updates the specified attributes of the specified virtual private interface.

Setting the MTU of a virtual interface to 8500 (jumbo frames) can cause an update to the underlying physical connection if it wasn't updated to support jumbo frames. Updating the connection disrupts network connectivity for all virtual interfaces associated with the connection for up to 30 seconds. To check whether your connection supports jumbo frames, call DescribeConnections. To check whether your virtual interface supports jumbo frames, call DescribeVirtualInterfaces.

Request Syntax

```
{
   "enableSiteLink": boolean,
   "mtu": number,
   "virtualInterfaceId": "string",
   "virtualInterfaceName": "string"
}
```

Request Parameters

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

The request accepts the following data in JSON format.

enableSiteLink

Indicates whether to enable or disable SiteLink.

Type: Boolean

Required: No

mtu

The maximum transmission unit (MTU), in bytes. The supported values are 1500 and 8500. The default value is 1500.

Type: Integer

Required: No

virtualInterfaceId

The ID of the virtual private interface.

Type: String

Required: Yes

virtualInterfaceName

The name of the virtual private interface.

Type: String

Required: No

Response Syntax

```
{
   "addressFamily": "string",
   "amazonAddress": "string",
   "amazonSideAsn": number,
   "asn": number,
   "asnLong": number,
   "authKey": "string",
   "awsDeviceV2": "string",
   "awsLogicalDeviceId": "string",
   "bgpPeers": [
      {
         "addressFamily": "string",
         "amazonAddress": "string",
         "asn": number,
         "asnLong": number,
         "authKey": "string",
         "awsDeviceV2": "string",
         "awsLogicalDeviceId": "string",
         "bgpPeerId": "string",
         "bgpPeerState": "string",
         "bgpStatus": "string",
         "customerAddress": "string"
      }
   ],
   "connectionId": "string",
   "customerAddress": "string",
```

```
    "customerRouterConfig": "string",
    "directConnectGatewayId": "string",
    "jumboFrameCapable": boolean,
    "location": "string",
    "mtu": number,
    "ownerAccount": "string",
    "region": "string",
    "routeFilterPrefixes": [
       {
          "cidr": "string"
       }
    ],
    "siteLinkEnabled": boolean,
    "tags": [
       {
          "key": "string",
          "value": "string"
       }
    ],
    "virtualGatewayId": "string",
    "virtualInterfaceId": "string",
    "virtualInterfaceName": "string",
    "virtualInterfaceState": "string",
    "virtualInterfaceType": "string",
    "vlan": number
}
```

Response Elements

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

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

addressFamily

The address family for the BGP peer.

Type: String

Valid Values: `ipv4 | ipv6`

amazonAddress

The IP address assigned to the Amazon interface.

Type: String

amazonSideAsn

The autonomous system number (AS) for the Amazon side of the connection.

Type: Long

asn

The autonomous system number (ASN). The valid range is from 1 to 2147483646 for Border Gateway Protocol (BGP) configuration. If you provide a number greater than the maximum, an error is returned. Use asnLong instead.

- You can use asnLong or asn, but not both. We recommend using asnLong as it supports a greater pool of numbers.
- If you provide a value in the same API call for both asn and asnLong, the API will only accept the value for asnLong.
- If you enter a 4-byte ASN for the asn parameter, the API returns an error.
- If you are using a 2-byte ASN, the API response will include the 2-byte value for both the asn and asnLong fields.

Type: Integer

asnLong

The long ASN for the virtual interface. The valid range is from 1 to 4294967294 for BGP configuration.

Note the following limitations when using asnLong:

- You can use asnLong or asn, but not both. We recommend using asnLong as it supports a greater pool of numbers.
- asnLong accepts any valid ASN value, regardless if it's 2-byte or 4-byte.
- When using a 4-byte asnLong, the API response returns 0 for the legacy asn attribute since 4-byte ASN values exceed the maximum supported value of 2,147,483,647.
- If you are using a 2-byte ASN, the API response will include the 2-byte value for both the asn and asnLong fields.
- If you provide a value in the same API call for both asn and asnLong, the API will only accept the value for asnLong.

Type: Long

authKey

The authentication key for BGP configuration. This string has a minimum length of 6 characters and and a maximun lenth of 80 characters.

Type: String

awsDeviceV2

The Direct Connect endpoint that terminates the physical connection.

Type: String

awsLogicalDeviceId

The Direct Connect endpoint that terminates the logical connection. This device might be different than the device that terminates the physical connection.

Type: String

bgpPeers

The BGP peers configured on this virtual interface.

Type: Array of BGPPeer objects

connectionId

The ID of the connection.

Type: String

customerAddress

The IP address assigned to the customer interface.

Type: String

customerRouterConfig

The customer router configuration.

Type: String

directConnectGatewayId

The ID of the Direct Connect gateway.

Type: String

jumboFrameCapable

Indicates whether jumbo frames are supported.

Type: Boolean

location

The location of the connection.

Type: String

mtu

The maximum transmission unit (MTU), in bytes. The supported values are 1500 and 8500. The default value is 1500

Type: Integer

ownerAccount

The ID of the AWS account that owns the virtual interface.

Type: String

region

The AWS Region where the virtual interface is located.

Type: String

routeFilterPrefixes

The routes to be advertised to the AWS network in this Region. Applies to public virtual interfaces.

Type: Array of RouteFilterPrefix objects

siteLinkEnabled

Indicates whether SiteLink is enabled.

Type: Boolean

tags

The tags associated with the virtual interface.

Type: Array of Tag objects

Array Members: Minimum number of 1 item.

virtualGatewayId

The ID of the virtual private gateway. Applies only to private virtual interfaces.

Type: String

virtualInterfaceId

The ID of the virtual interface.

Type: String

virtualInterfaceName

The name of the virtual interface assigned by the customer network. The name has a maximum of 100 characters. The following are valid characters: a-z, 0-9 and a hyphen (-).

Type: String

virtualInterfaceState

The state of the virtual interface. The following are the possible values:

- `confirming`: The creation of the virtual interface is pending confirmation from the virtual interface owner. If the owner of the virtual interface is different from the owner of the connection on which it is provisioned, then the virtual interface will remain in this state until it is confirmed by the virtual interface owner.
- `verifying`: This state only applies to public virtual interfaces. Each public virtual interface needs validation before the virtual interface can be created.
- `pending`: A virtual interface is in this state from the time that it is created until the virtual interface is ready to forward traffic.
- `available`: A virtual interface that is able to forward traffic.
- down: A virtual interface that is BGP down.
- `testing`: A virtual interface is in this state immediately after calling StartBgpFailoverTest and remains in this state during the duration of the test.
- `deleting`: A virtual interface is in this state immediately after calling DeleteVirtualInterface until it can no longer forward traffic.
- `deleted`: A virtual interface that cannot forward traffic.

- `rejected`: The virtual interface owner has declined creation of the virtual interface. If a virtual interface in the `Confirming` state is deleted by the virtual interface owner, the virtual interface enters the `Rejected` state.
- unknown: The state of the virtual interface is not available.

Type: String

Valid Values: `confirming | verifying | pending | available | down | testing | deleting | deleted | rejected | unknown`

virtualInterfaceType

The type of virtual interface. The possible values are `private`, `public` and `transit`.

Type: String

vlan

The ID of the VLAN.

Type: Integer

Errors

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

DirectConnectClientException

One or more parameters are not valid.

HTTP Status Code: 400

DirectConnectServerException

A server-side error occurred.

HTTP Status Code: 400

See Also

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

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

Data Types

The Direct Connect API contains several data types that various actions use. This section describes each data type in detail.

> **Note**
>
> The order of each element in a data type structure is not guaranteed. Applications should not assume a particular order.

The following data types are supported:

- AssociatedCoreNetwork
- AssociatedGateway
- BGPPeer
- Connection
- CustomerAgreement
- DirectConnectGateway
- DirectConnectGatewayAssociation
- DirectConnectGatewayAssociationProposal
- DirectConnectGatewayAttachment
- Interconnect
- Lag
- Loa
- Location
- MacSecKey
- NewBGPPeer
- NewPrivateVirtualInterface
- NewPrivateVirtualInterfaceAllocation
- NewPublicVirtualInterface
- NewPublicVirtualInterfaceAllocation
- NewTransitVirtualInterface

- NewTransitVirtualInterfaceAllocation
- ResourceTag
- RouteFilterPrefix
- RouterType
- Tag
- VirtualGateway
- VirtualInterface
- VirtualInterfaceTestHistory

AssociatedCoreNetwork

The AWS Cloud WAN core network that the Direct Connect gateway is associated to. This is only returned when a Direct Connect gateway is associated to a Cloud WAN core network.

Contents

attachmentId

the ID of the Direct Connect gateway attachment.

Type: String

Length Constraints: Minimum length of 12. Maximum length of 28.

Pattern: `^attachment-([0-9a-f]{1,17})$`

Required: No

id

The ID of the Cloud WAN core network that the Direct Connect gateway is associated to.

Type: String

Length Constraints: Minimum length of 14. Maximum length of 30.

Pattern: `^core-network-([0-9a-f]{1,17})$`

Required: No

ownerAccount

The account owner of the Cloud WAN core network.

Type: String

Required: No

See Also

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

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

AssociatedGateway

Information about the associated gateway.

Contents

id

The ID of the associated gateway.

Type: String

Required: No

ownerAccount

The ID of the AWS account that owns the associated virtual private gateway or transit gateway.

Type: String

Required: No

region

The Region where the associated gateway is located.

Type: String

Required: No

type

The type of associated gateway.

Type: String

Valid Values: `virtualPrivateGateway | transitGateway`

Required: No

See Also

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

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

BGPPeer

Information about a BGP peer.

Contents

addressFamily

The address family for the BGP peer.

Type: String

Valid Values: `ipv4 | ipv6`

Required: No

amazonAddress

The IP address assigned to the Amazon interface.

Type: String

Required: No

asn

The autonomous system number (ASN). The valid range is from 1 to 2147483646 for Border Gateway Protocol (BGP) configuration. If you provide a number greater than the maximum, an error is returned. Use `asnLong` instead.

- You can use `asnLong` or `asn`, but not both. We recommend using `asnLong` as it supports a greater pool of numbers.
- If you provide a value in the same API call for both `asn` and `asnLong`, the API will only accept the value for `asnLong`.
- If you enter a 4-byte ASN for the `asn` parameter, the API returns an error.
- If you are using a 2-byte ASN, the API response will include the 2-byte value for both the `asn` and `asnLong` fields.

Type: Integer

Required: No

asnLong

The long ASN for the BGP peer. The valid range is from 1 to 4294967294 for BGP configuration.

Note the following limitations when using `asnLong`:

- You can use `asnLong` or `asn`, but not both. We recommend using `asnLong` as it supports a greater pool of numbers.
- `asnLong` accepts any valid ASN value, regardless if it's 2-byte or 4-byte.
- When using a 4-byte `asnLong`, the API response returns `0` for the legacy `asn` attribute since 4-byte ASN values exceed the maximum supported value of 2,147,483,647.
- If you are using a 2-byte ASN, the API response will include the 2-byte value for both the `asn` and `asnLong` fields.
- If you provide a value in the same API call for both `asn` and `asnLong`, the API will only accept the value for `asnLong`.

Type: Long

Required: No

authKey

The authentication key for BGP configuration. This string has a minimum length of 6 characters and and a maximun lenth of 80 characters.

Type: String

Required: No

awsDeviceV2

The Direct Connect endpoint that terminates the BGP peer.

Type: String

Required: No

awsLogicalDeviceId

The Direct Connect endpoint that terminates the logical connection. This device might be different than the device that terminates the physical connection.

Type: String

Required: No

bgpPeerId

The ID of the BGP peer.

Type: String

Required: No

bgpPeerState

The state of the BGP peer. The following are the possible values:

- `verifying`: The BGP peering addresses or ASN require validation before the BGP peer can be created. This state applies only to public virtual interfaces.
- `pending`: The BGP peer is created, and remains in this state until it is ready to be established.
- `available`: The BGP peer is ready to be established.
- `deleting`: The BGP peer is being deleted.
- `deleted`: The BGP peer is deleted and cannot be established.

Type: String

Valid Values: `verifying | pending | available | deleting | deleted`

Required: No

bgpStatus

The status of the BGP peer. The following are the possible values:

- up: The BGP peer is established. This state does not indicate the state of the routing function. Ensure that you are receiving routes over the BGP session.
- down: The BGP peer is down.
- unknown: The BGP peer status is not available.

Type: String

Valid Values: up | down | unknown

Required: No

customerAddress

The IP address assigned to the customer interface.

Type: String

Required: No

See Also

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

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

Connection

Information about an Direct Connect connection.

Contents

awsDevice

This member has been deprecated.

The Direct Connect endpoint on which the physical connection terminates.

Type: String

Required: No

awsDeviceV2

The Direct Connect endpoint that terminates the physical connection.

Type: String

Required: No

awsLogicalDeviceId

The Direct Connect endpoint that terminates the logical connection. This device might be different than the device that terminates the physical connection.

Type: String

Required: No

bandwidth

The bandwidth of the connection.

Type: String

Required: No

connectionId

The ID of the connection.

Type: String

Required: No

connectionName

The name of the connection.

Type: String

Required: No

connectionState

The state of the connection. The following are the possible values:

- `ordering`: The initial state of a hosted connection provisioned on an interconnect. The connection stays in the ordering state until the owner of the hosted connection confirms or declines the connection order.
- `requested`: The initial state of a standard connection. The connection stays in the requested state until the Letter of Authorization (LOA) is sent to the customer.
- `pending`: The connection has been approved and is being initialized.
- `available`: The network link is up and the connection is ready for use.
- down: The network link is down.
- `deleting`: The connection is being deleted.
- `deleted`: The connection has been deleted.
- `rejected`: A hosted connection in the `ordering` state enters the `rejected` state if it is deleted by the customer.
- unknown: The state of the connection is not available.

Type: String

Valid Values: `ordering | requested | pending | available | down | deleting | deleted | rejected | unknown`

Required: No

encryptionMode

The MAC Security (MACsec) connection encryption mode.

The valid values are `no_encrypt`, `should_encrypt`, and `must_encrypt`.

Type: String

Required: No

hasLogicalRedundancy

Indicates whether the connection supports a secondary BGP peer in the same address family (IPv4/IPv6).

Type: String

Valid Values: `unknown | yes | no`

Required: No

jumboFrameCapable

Indicates whether jumbo frames are supported.

Type: Boolean

Required: No

lagId

The ID of the LAG.

Type: String

Required: No

loaIssueTime

The time of the most recent call to DescribeLoa for this connection.

Type: Timestamp

Required: No

location

The location of the connection.

Type: String

Required: No

macSecCapable

Indicates whether the connection supports MAC Security (MACsec).

Type: Boolean

Required: No

macSecKeys

The MAC Security (MACsec) security keys associated with the connection.

Type: Array of MacSecKey objects

Required: No

ownerAccount

The ID of the AWS account that owns the connection.

Type: String

Required: No

partnerInterconnectMacSecCapable

Indicates whether the interconnect hosting this connection supports MAC Security (MACsec).

Type: Boolean

Required: No

partnerName

The name of the Direct Connect service provider associated with the connection.

Type: String

Required: No

portEncryptionStatus

The MAC Security (MACsec) port link status of the connection.

The valid values are `Encryption Up`, which means that there is an active Connection Key Name, or `Encryption Down`.

Type: String

Required: No

providerName

The name of the service provider associated with the connection.

Type: String

Required: No

region

The AWS Region where the connection is located.

Type: String

Required: No

tags

The tags associated with the connection.

Type: Array of Tag objects

Array Members: Minimum number of 1 item.

Required: No

vlan

The ID of the VLAN.

Type: Integer

Required: No

See Also

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

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

CustomerAgreement

The name and status of a customer agreement.

Contents

agreementName

The name of the agreement.

Type: String

Length Constraints: Maximum length of 100.

Required: No

status

The status of the customer agreement. This will be either `signed` or `unsigned`

Type: String

Length Constraints: Maximum length of 30.

Required: No

See Also

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

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

DirectConnectGateway

Information about a Direct Connect gateway, which enables you to connect virtual interfaces and virtual private gateway or transit gateways.

Contents

amazonSideAsn

The autonomous system number (AS) for the Amazon side of the connection.

Type: Long

Required: No

directConnectGatewayId

The ID of the Direct Connect gateway.

Type: String

Required: No

directConnectGatewayName

The name of the Direct Connect gateway.

Type: String

Required: No

directConnectGatewayState

The state of the Direct Connect gateway. The following are the possible values:

- `pending`: The initial state after calling CreateDirectConnectGateway.
- `available`: The Direct Connect gateway is ready for use.
- `deleting`: The initial state after calling DeleteDirectConnectGateway.
- `deleted`: The Direct Connect gateway is deleted and cannot pass traffic.

Type: String

Valid Values: `pending | available | deleting | deleted`

Required: No

ownerAccount

The ID of the AWS account that owns the Direct Connect gateway.

Type: String

Required: No

stateChangeError

The error message if the state of an object failed to advance.

Type: String

Required: No

tags

Information about a tag.

Type: Array of Tag objects

Array Members: Minimum number of 1 item.

Required: No

See Also

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

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

DirectConnectGatewayAssociation

Information about an association between a Direct Connect gateway and a virtual private gateway or transit gateway.

Contents

allowedPrefixesToDirectConnectGateway

The Amazon VPC prefixes to advertise to the Direct Connect gateway.

Type: Array of RouteFilterPrefix objects

Required: No

associatedCoreNetwork

The ID of the Cloud WAN core network associated with the Direct Connect gateway attachment.

Type: AssociatedCoreNetwork object

Required: No

associatedGateway

Information about the associated gateway.

Type: AssociatedGateway object

Required: No

associationId

The ID of the Direct Connect gateway association.

Type: String

Required: No

associationState

The state of the association. The following are the possible values:

- `associating`: The initial state after calling CreateDirectConnectGatewayAssociation.
- `associated`: The Direct Connect gateway and virtual private gateway or transit gateway are successfully associated and ready to pass traffic.

- `disassociating`: The initial state after calling DeleteDirectConnectGatewayAssociation.
- `disassociated`: The virtual private gateway or transit gateway is disassociated from the Direct Connect gateway. Traffic flow between the Direct Connect gateway and virtual private gateway or transit gateway is stopped.
- `updating`: The CIDR blocks for the virtual private gateway or transit gateway are currently being updated. This could be new CIDR blocks added or current CIDR blocks removed.

Type: String

Valid Values: `associating | associated | disassociating | disassociated | updating`

Required: No

directConnectGatewayId

The ID of the Direct Connect gateway.

Type: String

Required: No

directConnectGatewayOwnerAccount

The ID of the AWS account that owns the associated gateway.

Type: String

Required: No

stateChangeError

The error message if the state of an object failed to advance.

Type: String

Required: No

virtualGatewayId

The ID of the virtual private gateway. Applies only to private virtual interfaces.

Type: String

Required: No

virtualGatewayOwnerAccount

The ID of the AWS account that owns the virtual private gateway.

Type: String

Required: No

virtualGatewayRegion

This member has been deprecated.

The AWS Region where the virtual private gateway is located.

Type: String

Required: No

See Also

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

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

DirectConnectGatewayAssociationProposal

Information about the proposal request to attach a virtual private gateway to a Direct Connect gateway.

Contents

associatedGateway

Information about the associated gateway.

Type: AssociatedGateway object

Required: No

directConnectGatewayId

The ID of the Direct Connect gateway.

Type: String

Required: No

directConnectGatewayOwnerAccount

The ID of the AWS account that owns the Direct Connect gateway.

Type: String

Required: No

existingAllowedPrefixesToDirectConnectGateway

The existing Amazon VPC prefixes advertised to the Direct Connect gateway.

Type: Array of RouteFilterPrefix objects

Required: No

proposalId

The ID of the association proposal.

Type: String

Required: No

proposalState

The state of the proposal. The following are possible values:

- `accepted`: The proposal has been accepted. The Direct Connect gateway association is available to use in this state.
- `deleted`: The proposal has been deleted by the owner that made the proposal. The Direct Connect gateway association cannot be used in this state.
- `requested`: The proposal has been requested. The Direct Connect gateway association cannot be used in this state.

Type: String

Valid Values: `requested | accepted | deleted`

Required: No

requestedAllowedPrefixesToDirectConnectGateway

The Amazon VPC prefixes to advertise to the Direct Connect gateway.

Type: Array of RouteFilterPrefix objects

Required: No

See Also

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

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

DirectConnectGatewayAttachment

Information about an attachment between a Direct Connect gateway and a virtual interface.

Contents

attachmentState

The state of the attachment. The following are the possible values:

- `attaching`: The initial state after a virtual interface is created using the Direct Connect gateway.
- `attached`: The Direct Connect gateway and virtual interface are attached and ready to pass traffic.
- `detaching`: The initial state after calling DeleteVirtualInterface.
- `detached`: The virtual interface is detached from the Direct Connect gateway. Traffic flow between the Direct Connect gateway and virtual interface is stopped.

Type: String

Valid Values: `attaching | attached | detaching | detached`

Required: No

attachmentType

The type of attachment.

Type: String

Valid Values: `TransitVirtualInterface | PrivateVirtualInterface`

Required: No

directConnectGatewayId

The ID of the Direct Connect gateway.

Type: String

Required: No

stateChangeError

The error message if the state of an object failed to advance.

Type: String

Required: No

virtualInterfaceId

The ID of the virtual interface.

Type: String

Required: No

virtualInterfaceOwnerAccount

The ID of the AWS account that owns the virtual interface.

Type: String

Required: No

virtualInterfaceRegion

The AWS Region where the virtual interface is located.

Type: String

Required: No

See Also

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

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

Interconnect

Information about an interconnect.

Contents

awsDevice

This member has been deprecated.

The Direct Connect endpoint on which the physical connection terminates.

Type: String

Required: No

awsDeviceV2

The Direct Connect endpoint that terminates the physical connection.

Type: String

Required: No

awsLogicalDeviceId

The Direct Connect endpoint that terminates the logical connection. This device might be different than the device that terminates the physical connection.

Type: String

Required: No

bandwidth

The bandwidth of the connection.

Type: String

Required: No

encryptionMode

The MAC Security (MACsec) encryption mode.

The valid values are `no_encrypt`, `should_encrypt`, and `must_encrypt`.

Type: String

Required: No

hasLogicalRedundancy

Indicates whether the interconnect supports a secondary BGP in the same address family (IPv4/IPv6).

Type: String

Valid Values: `unknown | yes | no`

Required: No

interconnectId

The ID of the interconnect.

Type: String

Required: No

interconnectName

The name of the interconnect.

Type: String

Required: No

interconnectState

The state of the interconnect. The following are the possible values:

- `requested`: The initial state of an interconnect. The interconnect stays in the requested state until the Letter of Authorization (LOA) is sent to the customer.
- `pending`: The interconnect is approved, and is being initialized.
- `available`: The network link is up, and the interconnect is ready for use.
- `down`: The network link is down.
- `deleting`: The interconnect is being deleted.
- `deleted`: The interconnect is deleted.

- unknown: The state of the interconnect is not available.

Type: String

Valid Values: `requested | pending | available | down | deleting | deleted | unknown`

Required: No

jumboFrameCapable

Indicates whether jumbo frames are supported.

Type: Boolean

Required: No

lagId

The ID of the LAG.

Type: String

Required: No

loaIssueTime

The time of the most recent call to DescribeLoa for this connection.

Type: Timestamp

Required: No

location

The location of the connection.

Type: String

Required: No

macSecCapable

Indicates whether the interconnect supports MAC Security (MACsec).

Type: Boolean

Required: No

macSecKeys

The MAC Security (MACsec) security keys.

Type: Array of MacSecKey objects

Required: No

portEncryptionStatus

The MAC Security (MACsec) port link status.

The valid values are `Encryption Up`, which means that there is an active Connection Key Name, or `Encryption Down`.

Type: String

Required: No

providerName

The name of the service provider associated with the interconnect.

Type: String

Required: No

region

The AWS Region where the connection is located.

Type: String

Required: No

tags

The tags associated with the interconnect.

Type: Array of Tag objects

Array Members: Minimum number of 1 item.

Required: No

See Also

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

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

Lag

Information about a link aggregation group (LAG).

Contents

allowsHostedConnections

Indicates whether the LAG can host other connections.

Type: Boolean

Required: No

awsDevice

This member has been deprecated.

The Direct Connect endpoint that hosts the LAG.

Type: String

Required: No

awsDeviceV2

The Direct Connect endpoint that hosts the LAG.

Type: String

Required: No

awsLogicalDeviceId

The Direct Connect endpoint that terminates the logical connection. This device might be different than the device that terminates the physical connection.

Type: String

Required: No

connections

The connections bundled by the LAG.

Type: Array of Connection objects

Required: No

connectionsBandwidth

The individual bandwidth of the physical connections bundled by the LAG. The possible values are 1Gbps, 10Gbps, 100Gbps, or 400 Gbps..

Type: String

Required: No

encryptionMode

The LAG MAC Security (MACsec) encryption mode.

The valid values are `no_encrypt`, `should_encrypt`, and `must_encrypt`.

Type: String

Required: No

hasLogicalRedundancy

Indicates whether the LAG supports a secondary BGP peer in the same address family (IPv4/IPv6).

Type: String

Valid Values: `unknown | yes | no`

Required: No

jumboFrameCapable

Indicates whether jumbo frames are supported.

Type: Boolean

Required: No

lagId

The ID of the LAG.

Type: String

Required: No

lagName

The name of the LAG.

Type: String

Required: No

lagState

The state of the LAG. The following are the possible values:

- `requested`: The initial state of a LAG. The LAG stays in the requested state until the Letter of Authorization (LOA) is available.
- `pending`: The LAG has been approved and is being initialized.
- `available`: The network link is established and the LAG is ready for use.
- down: The network link is down.
- `deleting`: The LAG is being deleted.
- `deleted`: The LAG is deleted.
- unknown: The state of the LAG is not available.

Type: String

Valid Values: `requested | pending | available | down | deleting | deleted |` unknown

Required: No

location

The location of the LAG.

Type: String

Required: No

macSecCapable

Indicates whether the LAG supports MAC Security (MACsec).

Type: Boolean

Required: No

macSecKeys

The MAC Security (MACsec) security keys associated with the LAG.

Type: Array of MacSecKey objects

Required: No

minimumLinks

The minimum number of physical dedicated connections that must be operational for the LAG itself to be operational.

Type: Integer

Required: No

numberOfConnections

The number of physical dedicated connections initially provisioned and bundled by the LAG. You can have a maximum of four connections when the port speed is 1 Gbps or 10 Gbps, or two when the port speed is 100 Gbps or 400 Gbps.

Type: Integer

Required: No

ownerAccount

The ID of the AWS account that owns the LAG.

Type: String

Required: No

providerName

The name of the service provider associated with the LAG.

Type: String

Required: No

region

The AWS Region where the connection is located.

Type: String

Required: No

tags

The tags associated with the LAG.

Type: Array of Tag objects

Array Members: Minimum number of 1 item.

Required: No

See Also

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

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

Loa

Information about a Letter of Authorization - Connecting Facility Assignment (LOA-CFA) for a connection.

Contents

loaContent

The binary contents of the LOA-CFA document.

Type: Base64-encoded binary data object

Required: No

loaContentType

The standard media type for the LOA-CFA document. The only supported value is application/pdf.

Type: String

Valid Values: `application/pdf`

Required: No

See Also

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

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

Location

Information about an Direct Connect location.

Contents

availableMacSecPortSpeeds

The available MAC Security (MACsec) port speeds for the location.

Type: Array of strings

Required: No

availablePortSpeeds

The available port speeds for the location.

Type: Array of strings

Required: No

availableProviders

The name of the service provider for the location.

Type: Array of strings

Required: No

locationCode

The code for the location.

Type: String

Required: No

locationName

The name of the location. This includes the name of the colocation partner and the physical site of the building.

Type: String

Required: No

region

The AWS Region for the location.

Type: String

Required: No

See Also

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

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

MacSecKey

Information about the MAC Security (MACsec) secret key.

Contents

ckn

The Connection Key Name (CKN) for the MAC Security secret key.

Type: String

Required: No

secretARN

The Amazon Resource Name (ARN) of the MAC Security (MACsec) secret key.

Type: String

Required: No

startOn

The date that the MAC Security (MACsec) secret key takes effect. The value is displayed in UTC format.

Type: String

Required: No

state

The state of the MAC Security (MACsec) secret key.

The possible values are:

- `associating`: The MAC Security (MACsec) secret key is being validated and not yet associated with the connection or LAG.
- `associated`: The MAC Security (MACsec) secret key is validated and associated with the connection or LAG.
- `disassociating`: The MAC Security (MACsec) secret key is being disassociated from the connection or LAG

- `disassociated`: The MAC Security (MACsec) secret key is no longer associated with the connection or LAG.

Type: String

Required: No

See Also

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

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

NewBGPPeer

Information about a new BGP peer.

Contents

addressFamily

The address family for the BGP peer.

Type: String

Valid Values: `ipv4 | ipv6`

Required: No

amazonAddress

The IP address assigned to the Amazon interface.

Type: String

Required: No

asn

The autonomous system number (ASN). The valid range is from 1 to 2147483646 for Border Gateway Protocol (BGP) configuration. If you provide a number greater than the maximum, an error is returned. Use `asnLong` instead.

Type: Integer

Required: No

asnLong

The long ASN for a new BGP peer. The valid range is from 1 to 4294967294.

Type: Long

Required: No

authKey

The authentication key for BGP configuration. This string has a minimum length of 6 characters and and a maximun lenth of 80 characters.

Type: String

Required: No

customerAddress

The IP address assigned to the customer interface.

Type: String

Required: No

See Also

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

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

NewPrivateVirtualInterface

Information about a private virtual interface.

Contents

virtualInterfaceName

The name of the virtual interface assigned by the customer network. The name has a maximum of 100 characters. The following are valid characters: a-z, 0-9 and a hyphen (-).

Type: String

Required: Yes

vlan

The ID of the VLAN.

Type: Integer

Required: Yes

addressFamily

The address family for the BGP peer.

Type: String

Valid Values: `ipv4 | ipv6`

Required: No

amazonAddress

The IP address assigned to the Amazon interface.

Type: String

Required: No

asn

The autonomous system number (ASN). The valid range is from 1 to 2147483646 for Border Gateway Protocol (BGP) configuration. If you provide a number greater than the maximum, an error is returned. Use `asnLong` instead.

- You can use asnLong or asn, but not both. We recommend using asnLong as it supports a greater pool of numbers.
- If you provide a value in the same API call for both asn and asnLong, the API will only accept the value for asnLong.
- If you enter a 4-byte ASN for the asn parameter, the API returns an error.
- If you are using a 2-byte ASN, the API response will include the 2-byte value for both the asn and asnLong fields.

The valid values are 1-2147483646.

Type: Integer

Required: No

asnLong

The long ASN for a new private virtual interface. The valid range is from 1 to 4294967294 for BGP configuration.

Note the following limitations when using asnLong:

- You can use asnLong or asn, but not both. We recommend using asnLong as it supports a greater pool of numbers.
- asnLong accepts any valid ASN value, regardless if it's 2-byte or 4-byte.
- When using a 4-byte asnLong, the API response returns 0 for the legacy asn attribute since 4-byte ASN values exceed the maximum supported value of 2,147,483,647.
- If you are using a 2-byte ASN, the API response will include the 2-byte value for both the asn and asnLong fields.
- If you provide a value in the same API call for both asn and asnLong, the API will only accept the value for asnLong.

Type: Long

Required: No

authKey

The authentication key for BGP configuration. This string has a minimum length of 6 characters and and a maximun lenth of 80 characters.

Type: String

Required: No

customerAddress

The IP address assigned to the customer interface.

Type: String

Required: No

directConnectGatewayId

The ID of the Direct Connect gateway.

Type: String

Required: No

enableSiteLink

Indicates whether to enable or disable SiteLink.

Type: Boolean

Required: No

mtu

The maximum transmission unit (MTU), in bytes. The supported values are 1500 and 8500. The default value is 1500.

Type: Integer

Required: No

tags

The tags associated with the private virtual interface.

Type: Array of Tag objects

Array Members: Minimum number of 1 item.

Required: No

virtualGatewayId

The ID of the virtual private gateway.

Type: String

Required: No

See Also

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

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

NewPrivateVirtualInterfaceAllocation

Information about a private virtual interface to be provisioned on a connection.

Contents

virtualInterfaceName

The name of the virtual interface assigned by the customer network. The name has a maximum of 100 characters. The following are valid characters: a-z, 0-9 and a hyphen (-).

Type: String

Required: Yes

vlan

The ID of the VLAN.

Type: Integer

Required: Yes

addressFamily

The address family for the BGP peer.

Type: String

Valid Values: `ipv4 | ipv6`

Required: No

amazonAddress

The IP address assigned to the Amazon interface.

Type: String

Required: No

asn

The autonomous system number (ASN). The valid range is from 1 to 2147483646 for Border Gateway Protocol (BGP) configuration. If you provide a number greater than the maximum, an error is returned. Use `asnLong` instead.

- You can use asnLong or asn, but not both. We recommend using asnLong as it supports a greater pool of numbers.
- If you provide a value in the same API call for both asn and asnLong, the API will only accept the value for asnLong.
- If you enter a 4-byte ASN for the asn parameter, the API returns an error.
- If you are using a 2-byte ASN, the API response will include the 2-byte value for both the asn and asnLong fields.

The valid values are 1-2147483646.

Type: Integer

Required: No

asnLong

The ASN when allocating a new private virtual interface. The valid range is from 1 to 4294967294 for BGP configuration.

Note the following limitations when using asnLong:

- You can use asnLong or asn, but not both. We recommend using asnLong as it supports a greater pool of numbers.
- asnLong accepts any valid ASN value, regardless if it's 2-byte or 4-byte.
- When using a 4-byte asnLong, the API response returns 0 for the legacy asn attribute since 4-byte ASN values exceed the maximum supported value of 2,147,483,647.
- If you are using a 2-byte ASN, the API response will include the 2-byte value for both the asn and asnLong fields.
- If you provide a value in the same API call for both asn and asnLong, the API will only accept the value for asnLong.

Type: Long

Required: No

authKey

The authentication key for BGP configuration. This string has a minimum length of 6 characters and and a maximun lenth of 80 characters.

Type: String

Required: No

customerAddress

The IP address assigned to the customer interface.

Type: String

Required: No

mtu

The maximum transmission unit (MTU), in bytes. The supported values are 1500 and 8500. The default value is 1500.

Type: Integer

Required: No

tags

The tags associated with the private virtual interface.

Type: Array of Tag objects

Array Members: Minimum number of 1 item.

Required: No

See Also

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

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

NewPublicVirtualInterface

Information about a public virtual interface.

Contents

virtualInterfaceName

The name of the virtual interface assigned by the customer network. The name has a maximum of 100 characters. The following are valid characters: a-z, 0-9 and a hyphen (-).

Type: String

Required: Yes

vlan

The ID of the VLAN.

Type: Integer

Required: Yes

addressFamily

The address family for the BGP peer.

Type: String

Valid Values: `ipv4 | ipv6`

Required: No

amazonAddress

The IP address assigned to the Amazon interface.

Type: String

Required: No

asn

The autonomous system number (ASN). The valid range is from 1 to 2147483646 for Border Gateway Protocol (BGP) configuration. If you provide a number greater than the maximum, an error is returned. Use `asnLong` instead.

- You can use asnLong or asn, but not both. We recommend using asnLong as it supports a greater pool of numbers.
- If you provide a value in the same API call for both asn and asnLong, the API will only accept the value for asnLong.
- If you enter a 4-byte ASN for the asn parameter, the API returns an error.
- If you are using a 2-byte ASN, the API response will include the 2-byte value for both the asn and asnLong fields.

Type: Integer

Required: No

asnLong

The long ASN for a new public virtual interface. The valid range is from 1 to 4294967294 for BGP configuration.

Note the following limitations when using asnLong:

- You can use asnLong or asn, but not both. We recommend using asnLong as it supports a greater pool of numbers.
- asnLong accepts any valid ASN value, regardless if it's 2-byte or 4-byte.
- When using a 4-byte asnLong, the API response returns 0 for the legacy asn attribute since 4-byte ASN values exceed the maximum supported value of 2,147,483,647.
- If you are using a 2-byte ASN, the API response will include the 2-byte value for both the asn and asnLong fields.
- If you provide a value in the same API call for both asn and asnLong, the API will only accept the value for asnLong.

Type: Long

Required: No

authKey

The authentication key for BGP configuration. This string has a minimum length of 6 characters and and a maximun lenth of 80 characters.

Type: String

Required: No

customerAddress

The IP address assigned to the customer interface.

Type: String

Required: No

routeFilterPrefixes

The routes to be advertised to the AWS network in this Region. Applies to public virtual interfaces.

Type: Array of RouteFilterPrefix objects

Required: No

tags

The tags associated with the public virtual interface.

Type: Array of Tag objects

Array Members: Minimum number of 1 item.

Required: No

See Also

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

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

NewPublicVirtualInterfaceAllocation

Information about a public virtual interface to be provisioned on a connection.

Contents

virtualInterfaceName

The name of the virtual interface assigned by the customer network. The name has a maximum of 100 characters. The following are valid characters: a-z, 0-9 and a hyphen (-).

Type: String

Required: Yes

vlan

The ID of the VLAN.

Type: Integer

Required: Yes

addressFamily

The address family for the BGP peer.

Type: String

Valid Values: `ipv4 | ipv6`

Required: No

amazonAddress

The IP address assigned to the Amazon interface.

Type: String

Required: No

asn

The autonomous system number (ASN). The valid range is from 1 to 2147483646 for Border Gateway Protocol (BGP) configuration. If you provide a number greater than the maximum, an error is returned. Use `asnLong` instead.

- You can use asnLong or asn, but not both. We recommend using asnLong as it supports a greater pool of numbers.
- If you provide a value in the same API call for both asn and asnLong, the API will only accept the value for asnLong.
- If you enter a 4-byte ASN for the asn parameter, the API returns an error.
- If you are using a 2-byte ASN, the API response will include the 2-byte value for both the asn and asnLong fields.

The valid values are 1-2147483646.

Type: Integer

Required: No

asnLong

The ASN when allocating a new public virtual interface. The valid range is from 1 to 4294967294 for BGP configuration.

Note the following limitations when using asnLong:

- You can use asnLong or asn, but not both. We recommend using asnLong as it supports a greater pool of numbers.
- asnLong accepts any valid ASN value, regardless if it's 2-byte or 4-byte.
- When using a 4-byte asnLong, the API response returns 0 for the legacy asn attribute since 4-byte ASN values exceed the maximum supported value of 2,147,483,647.
- If you are using a 2-byte ASN, the API response will include the 2-byte value for both the asn and asnLong fields.
- If you provide a value in the same API call for both asn and asnLong, the API will only accept the value for asnLong.

Type: Long

Required: No

authKey

The authentication key for BGP configuration. This string has a minimum length of 6 characters and and a maximun lenth of 80 characters.

Type: String

Required: No

customerAddress

The IP address assigned to the customer interface.

Type: String

Required: No

routeFilterPrefixes

The routes to be advertised to the AWS network in this Region. Applies to public virtual interfaces.

Type: Array of RouteFilterPrefix objects

Required: No

tags

The tags associated with the public virtual interface.

Type: Array of Tag objects

Array Members: Minimum number of 1 item.

Required: No

See Also

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

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

NewTransitVirtualInterface

Information about a transit virtual interface.

Contents

addressFamily

The address family for the BGP peer.

Type: String

Valid Values: `ipv4 | ipv6`

Required: No

amazonAddress

The IP address assigned to the Amazon interface.

Type: String

Required: No

asn

The autonomous system number (ASN). The valid range is from 1 to 2147483646 for Border Gateway Protocol (BGP) configuration. If you provide a number greater than the maximum, an error is returned. Use `asnLong` instead.

- You can use `asnLong` or `asn`, but not both. We recommend using `asnLong` as it supports a greater pool of numbers.
- If you provide a value in the same API call for both `asn` and `asnLong`, the API will only accept the value for `asnLong`.
- If you enter a 4-byte ASN for the `asn` parameter, the API returns an error.
- If you are using a 2-byte ASN, the API response will include the 2-byte value for both the `asn` and `asnLong` fields.

Type: Integer

Required: No

asnLong

The long ASN for a new transit virtual interface.The valid range is from 1 to 4294967294 for BGP configuration.

Note the following limitations when using `asnLong`:

- You can use `asnLong` or `asn`, but not both. We recommend using `asnLong` as it supports a greater pool of numbers.
- `asnLong` accepts any valid ASN value, regardless if it's 2-byte or 4-byte.
- When using a 4-byte `asnLong`, the API response returns `0` for the legacy `asn` attribute since 4-byte ASN values exceed the maximum supported value of 2,147,483,647.
- If you are using a 2-byte ASN, the API response will include the 2-byte value for both the `asn` and `asnLong` fields.
- If you provide a value in the same API call for both `asn` and `asnLong`, the API will only accept the value for `asnLong`.

Type: Long

Required: No

authKey

The authentication key for BGP configuration. This string has a minimum length of 6 characters and and a maximun lenth of 80 characters.

Type: String

Required: No

customerAddress

The IP address assigned to the customer interface.

Type: String

Required: No

directConnectGatewayId

The ID of the Direct Connect gateway.

Type: String

Required: No

enableSiteLink

Indicates whether to enable or disable SiteLink.

Type: Boolean

Required: No

mtu

The maximum transmission unit (MTU), in bytes. The supported values are 1500 and 8500. The default value is 1500.

Type: Integer

Required: No

tags

The tags associated with the transitive virtual interface.

Type: Array of Tag objects

Array Members: Minimum number of 1 item.

Required: No

virtualInterfaceName

The name of the virtual interface assigned by the customer network. The name has a maximum of 100 characters. The following are valid characters: a-z, 0-9 and a hyphen (-).

Type: String

Required: No

vlan

The ID of the VLAN.

Type: Integer

Required: No

See Also

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

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

NewTransitVirtualInterfaceAllocation

Information about a transit virtual interface to be provisioned on a connection.

Contents

addressFamily

The address family for the BGP peer.

Type: String

Valid Values: `ipv4 | ipv6`

Required: No

amazonAddress

The IP address assigned to the Amazon interface.

Type: String

Required: No

asn

The autonomous system number (ASN). The valid range is from 1 to 2147483646 for Border Gateway Protocol (BGP) configuration. If you provide a number greater than the maximum, an error is returned. Use `asnLong` instead.

- You can use `asnLong` or `asn`, but not both. We recommend using `asnLong` as it supports a greater pool of numbers.
- If you provide a value in the same API call for both `asn` and `asnLong`, the API will only accept the value for `asnLong`.
- If you enter a 4-byte ASN for the `asn` parameter, the API returns an error.
- If you are using a 2-byte ASN, the API response will include the 2-byte value for both the `asn` and `asnLong` fields.

The valid values are 1-2147483646.

Type: Integer

Required: No

asnLong

The ASN when allocating a new transit virtual interface. The valid range is from 1 to 4294967294 for BGP configuration.

Note the following limitations when using asnLong:

- You can use asnLong or asn, but not both. We recommend using asnLong as it supports a greater pool of numbers.
- asnLong accepts any valid ASN value, regardless if it's 2-byte or 4-byte.
- When using a 4-byte asnLong, the API response returns 0 for the legacy asn attribute since 4-byte ASN values exceed the maximum supported value of 2,147,483,647.
- If you are using a 2-byte ASN, the API response will include the 2-byte value for both the asn and asnLong fields.
- If you provide a value in the same API call for both asn and asnLong, the API will only accept the value for asnLong.

Type: Long

Required: No

authKey

The authentication key for BGP configuration. This string has a minimum length of 6 characters and and a maximun lenth of 80 characters.

Type: String

Required: No

customerAddress

The IP address assigned to the customer interface.

Type: String

Required: No

mtu

The maximum transmission unit (MTU), in bytes. The supported values are 1500 and 8500. The default value is 1500

Type: Integer

Required: No

tags

The tags associated with the transitive virtual interface.

Type: Array of Tag objects

Array Members: Minimum number of 1 item.

Required: No

virtualInterfaceName

The name of the virtual interface assigned by the customer network. The name has a maximum of 100 characters. The following are valid characters: a-z, 0-9 and a hyphen (-).

Type: String

Required: No

vlan

The ID of the VLAN.

Type: Integer

Required: No

See Also

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

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

ResourceTag

Information about a tag associated with an Direct Connect resource.

Contents

resourceArn

The Amazon Resource Name (ARN) of the resource.

Type: String

Required: No

tags

The tags.

Type: Array of Tag objects

Array Members: Minimum number of 1 item.

Required: No

See Also

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

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

RouteFilterPrefix

Information about a route filter prefix that a customer can advertise through Border Gateway Protocol (BGP) over a public virtual interface.

Contents

cidr

The CIDR block for the advertised route. Separate multiple routes using commas. An IPv6 CIDR must use /64 or shorter.

Type: String

Required: No

See Also

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

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

RouterType

Information about the virtual router.

Contents

platform

The virtual interface router platform.

Type: String

Required: No

routerTypeIdentifier

Identifies the router by a combination of vendor, platform, and software version. For example, `CiscoSystemsInc-2900SeriesRouters-IOS124`.

Type: String

Required: No

software

The router software.

Type: String

Required: No

vendor

The vendor for the virtual interface's router.

Type: String

Required: No

xsltTemplateName

The template for the virtual interface's router.

Type: String

Required: No

xsltTemplateNameForMacSec

The MAC Security (MACsec) template for the virtual interface's router.

Type: String

Required: No

See Also

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

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

Tag

Information about a tag.

Contents

key

The key.

Type: String

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

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

Required: Yes

value

The value.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 256.

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

Required: No

See Also

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

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

VirtualGateway

Information about a virtual private gateway for a private virtual interface.

Contents

virtualGatewayId

The ID of the virtual private gateway.

Type: String

Required: No

virtualGatewayState

The state of the virtual private gateway. The following are the possible values:

- `pending`: Initial state after creating the virtual private gateway.
- `available`: Ready for use by a private virtual interface.
- `deleting`: Initial state after deleting the virtual private gateway.
- `deleted`: The virtual private gateway is deleted. The private virtual interface is unable to send traffic over this gateway.

Type: String

Required: No

See Also

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

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

VirtualInterface

Information about a virtual interface.

Contents

addressFamily

The address family for the BGP peer.

Type: String

Valid Values: `ipv4 | ipv6`

Required: No

amazonAddress

The IP address assigned to the Amazon interface.

Type: String

Required: No

amazonSideAsn

The autonomous system number (AS) for the Amazon side of the connection.

Type: Long

Required: No

asn

The autonomous system number (ASN). The valid range is from 1 to 2147483646 for Border Gateway Protocol (BGP) configuration. If you provide a number greater than the maximum, an error is returned. Use `asnLong` instead.

- You can use `asnLong` or `asn`, but not both. We recommend using `asnLong` as it supports a greater pool of numbers.
- If you provide a value in the same API call for both `asn` and `asnLong`, the API will only accept the value for `asnLong`.
- If you enter a 4-byte ASN for the `asn` parameter, the API returns an error.

- If you are using a 2-byte ASN, the API response will include the 2-byte value for both the asn and asnLong fields.

Type: Integer

Required: No

asnLong

The long ASN for the virtual interface. The valid range is from 1 to 4294967294 for BGP configuration.

Note the following limitations when using asnLong:

- You can use asnLong or asn, but not both. We recommend using asnLong as it supports a greater pool of numbers.
- asnLong accepts any valid ASN value, regardless if it's 2-byte or 4-byte.
- When using a 4-byte asnLong, the API response returns 0 for the legacy asn attribute since 4-byte ASN values exceed the maximum supported value of 2,147,483,647.
- If you are using a 2-byte ASN, the API response will include the 2-byte value for both the asn and asnLong fields.
- If you provide a value in the same API call for both asn and asnLong, the API will only accept the value for asnLong.

Type: Long

Required: No

authKey

The authentication key for BGP configuration. This string has a minimum length of 6 characters and and a maximun lenth of 80 characters.

Type: String

Required: No

awsDeviceV2

The Direct Connect endpoint that terminates the physical connection.

Type: String

Required: No

awsLogicalDeviceId

The Direct Connect endpoint that terminates the logical connection. This device might be different than the device that terminates the physical connection.

Type: String

Required: No

bgpPeers

The BGP peers configured on this virtual interface.

Type: Array of BGPPeer objects

Required: No

connectionId

The ID of the connection.

Type: String

Required: No

customerAddress

The IP address assigned to the customer interface.

Type: String

Required: No

customerRouterConfig

The customer router configuration.

Type: String

Required: No

directConnectGatewayId

The ID of the Direct Connect gateway.

Type: String

Required: No

jumboFrameCapable

Indicates whether jumbo frames are supported.

Type: Boolean

Required: No

location

The location of the connection.

Type: String

Required: No

mtu

The maximum transmission unit (MTU), in bytes. The supported values are 1500 and 8500. The default value is 1500

Type: Integer

Required: No

ownerAccount

The ID of the AWS account that owns the virtual interface.

Type: String

Required: No

region

The AWS Region where the virtual interface is located.

Type: String

Required: No

routeFilterPrefixes

The routes to be advertised to the AWS network in this Region. Applies to public virtual interfaces.

Type: Array of RouteFilterPrefix objects

Required: No

siteLinkEnabled

Indicates whether SiteLink is enabled.

Type: Boolean

Required: No

tags

The tags associated with the virtual interface.

Type: Array of Tag objects

Array Members: Minimum number of 1 item.

Required: No

virtualGatewayId

The ID of the virtual private gateway. Applies only to private virtual interfaces.

Type: String

Required: No

virtualInterfaceId

The ID of the virtual interface.

Type: String

Required: No

virtualInterfaceName

The name of the virtual interface assigned by the customer network. The name has a maximum of 100 characters. The following are valid characters: a-z, 0-9 and a hyphen (-).

Type: String

Required: No

virtualInterfaceState

The state of the virtual interface. The following are the possible values:

- `confirming`: The creation of the virtual interface is pending confirmation from the virtual interface owner. If the owner of the virtual interface is different from the owner of the connection on which it is provisioned, then the virtual interface will remain in this state until it is confirmed by the virtual interface owner.
- `verifying`: This state only applies to public virtual interfaces. Each public virtual interface needs validation before the virtual interface can be created.
- `pending`: A virtual interface is in this state from the time that it is created until the virtual interface is ready to forward traffic.
- `available`: A virtual interface that is able to forward traffic.
- down: A virtual interface that is BGP down.
- `testing`: A virtual interface is in this state immediately after calling StartBgpFailoverTest and remains in this state during the duration of the test.
- `deleting`: A virtual interface is in this state immediately after calling DeleteVirtualInterface until it can no longer forward traffic.
- `deleted`: A virtual interface that cannot forward traffic.
- `rejected`: The virtual interface owner has declined creation of the virtual interface. If a virtual interface in the `Confirming` state is deleted by the virtual interface owner, the virtual interface enters the `Rejected` state.
- unknown: The state of the virtual interface is not available.

Type: String

Valid Values: `confirming | verifying | pending | available | down | testing | deleting | deleted | rejected | unknown`

Required: No

virtualInterfaceType

The type of virtual interface. The possible values are `private`, `public` and `transit`.

Type: String

Required: No

vlan

The ID of the VLAN.

Type: Integer

Required: No

See Also

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

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

VirtualInterfaceTestHistory

Information about the virtual interface failover test.

Contents

bgpPeers

The BGP peers that were put in the DOWN state as part of the virtual interface failover test.

Type: Array of strings

Required: No

endTime

The time that the virtual interface moves out of the DOWN state.

Type: Timestamp

Required: No

ownerAccount

The owner ID of the tested virtual interface.

Type: String

Required: No

startTime

The time that the virtual interface moves to the DOWN state.

Type: Timestamp

Required: No

status

The status of the virtual interface failover test.

Type: String

Required: No

testDurationInMinutes

The time that the virtual interface failover test ran in minutes.

Type: Integer

Required: No

testId

The ID of the virtual interface failover test.

Type: String

Required: No

virtualInterfaceId

The ID of the tested virtual interface.

Type: String

Required: No

See Also

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

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

Common Parameters

The following list contains the parameters that all actions use for signing Signature Version 4 requests with a query string. Any action-specific parameters are listed in the topic for that action. For more information about Signature Version 4, see Signing AWS API requests in the *IAM User Guide*.

X-Amz-Algorithm

The hash algorithm that you used to create the request signature.

Condition: Specify this parameter when you include authentication information in a query string instead of in the HTTP authorization header.

Type: string

Valid Values: `AWS4-HMAC-SHA256`

Required: Conditional

X-Amz-Credential

The credential scope value, which is a string that includes your access key, the date, the region you are targeting, the service you are requesting, and a termination string ("aws4_request"). The value is expressed in the following format: *access_key*/*YYYYMMDD*/*region*/*service*/aws4_request.

For more information, see Create a signed AWS API request in the *IAM User Guide*.

Condition: Specify this parameter when you include authentication information in a query string instead of in the HTTP authorization header.

Type: string

Required: Conditional

X-Amz-Date

The date that is used to create the signature. The format must be ISO 8601 basic format (YYYYMMDD'T'HHMMSS'Z'). For example, the following date time is a valid X-Amz-Date value: `20120325T120000Z`.

Condition: X-Amz-Date is optional for all requests; it can be used to override the date used for signing requests. If the Date header is specified in the ISO 8601 basic format, X-Amz-Date is not required. When X-Amz-Date is used, it always overrides the value of the Date header. For more information, see Elements of an AWS API request signature in the *IAM User Guide*.

Type: string

Required: Conditional

X-Amz-Security-Token

The temporary security token that was obtained through a call to AWS Security Token Service (AWS STS). For a list of services that support temporary security credentials from AWS STS, see AWS services that work with IAM in the *IAM User Guide*.

Condition: If you're using temporary security credentials from AWS STS, you must include the security token.

Type: string

Required: Conditional

X-Amz-Signature

Specifies the hex-encoded signature that was calculated from the string to sign and the derived signing key.

Condition: Specify this parameter when you include authentication information in a query string instead of in the HTTP authorization header.

Type: string

Required: Conditional

X-Amz-SignedHeaders

Specifies all the HTTP headers that were included as part of the canonical request. For more information about specifying signed headers, see Create a signed AWS API request in the *IAM User Guide*.

Condition: Specify this parameter when you include authentication information in a query string instead of in the HTTP authorization header.

Type: string

Required: Conditional

Common Error Types

This section lists common error types that this AWS service may return. Not all services return all error types listed here. For errors specific to an API action for this service, see the topic for that API action.

AccessDeniedException

You don't have permission to perform this action. Verify that your IAM policy includes the required permissions.

HTTP Status Code: 403

ExpiredTokenException

The security token included in the request has expired. Request a new security token and try again.

HTTP Status Code: 403

IncompleteSignature

The request signature doesn't conform to AWS standards. Verify that you're using valid AWS credentials and that your request is properly formatted. If you're using an SDK, ensure it's up to date.

HTTP Status Code: 403

InternalFailure

The request can't be processed right now because of an internal server issue. Try again later. If the problem persists, contact AWS Support.

HTTP Status Code: 500

MalformedHttpRequestException

The request body can't be processed. This typically happens when the request body can't be decompressed using the specified content encoding algorithm. Verify that the content encoding header matches the compression format used.

HTTP Status Code: 400

NotAuthorized

You don't have permissions to perform this action. Verify that your IAM policy includes the required permissions.

HTTP Status Code: 401

OptInRequired

Your AWS account needs a subscription for this service. Verify that you've enabled the service in your account.

HTTP Status Code: 403

RequestAbortedException

The request was aborted before a response could be returned. This typically happens when the client closes the connection.

HTTP Status Code: 400

RequestEntityTooLargeException

The request entity is too large. Reduce the size of the request body and try again.

HTTP Status Code: 413

RequestTimeoutException

The request timed out. The server didn't receive the complete request within the expected time frame. Try again.

HTTP Status Code: 408

ServiceUnavailable

The service is temporarily unavailable. Try again later.

HTTP Status Code: 503

ThrottlingException

Your request rate is too high. The AWS SDKs automatically retry requests that receive this exception. Reduce the frequency of requests.

HTTP Status Code: 400

UnknownOperationException

The action or operation isn't recognized. Verify that the action name is spelled correctly and that it's supported by the API version you're using.

HTTP Status Code: 404

UnrecognizedClientException

The X.509 certificate or AWS access key ID you provided doesn't exist in our records. Verify that you're using valid credentials and that they haven't expired.

HTTP Status Code: 403

ValidationError

The input doesn't meet the required format or constraints. Check that all required parameters are included and that values are valid.

HTTP Status Code: 400