aws

API Reference

Amazon EC2 Instance Connect

API Version 2018-04-02

Amazon EC2 Instance Connect: API Reference

Table of Contents

Welcome

This is the *Amazon EC2 Instance Connect API Reference*. It provides descriptions, syntax, and usage examples for each of the actions for Amazon EC2 Instance Connect. Amazon EC2 Instance Connect enables system administrators to publish one-time use SSH public keys to EC2, providing users a simple and secure way to connect to their instances.

To view the Amazon EC2 Instance Connect content in the *Amazon EC2 User Guide*, see Connect to your Linux instance using EC2 Instance Connect.

For Amazon EC2 APIs, see the Amazon EC2 API Reference.

This document was last published on April 10, 2026.

Actions

The following actions are supported:

- SendSerialConsoleSSHPublicKey
- SendSSHPublicKey

SendSerialConsoleSSHPublicKey

Pushes an SSH public key to the specified EC2 instance. The key remains for 60 seconds, which gives you 60 seconds to establish a serial console connection to the instance using SSH. For more information, see EC2 Serial Console in the *Amazon EC2 User Guide*.

Request Syntax

```
{
   "InstanceId": "string",
   "SerialPort": number,
   "SSHPublicKey": "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.

InstanceId

The ID of the EC2 instance.

Type: String

Length Constraints: Minimum length of 10. Maximum length of 32.

Pattern: `^i-[a-f0-9]+$`

Required: Yes

SerialPort

The serial port of the EC2 instance. Currently only port 0 is supported.

Default: 0

Type: Integer

Valid Range: Fixed value of 0.

Required: No

SSHPublicKey

The public key material. To use the public key, you must have the matching private key. For information about the supported key formats and lengths, see Requirements for key pairs in the *Amazon EC2 User Guide*.

Type: String

Length Constraints: Minimum length of 80. Maximum length of 4096.

Required: Yes

Response Syntax

```
{
   "RequestId": "string",
   "Success": boolean
}
```

Response Elements

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

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

RequestId

The ID of the request. Please provide this ID when contacting AWS Support for assistance.

Type: String

Success

Is true if the request succeeds and an error otherwise.

Type: Boolean

Errors

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

AuthException

Either your AWS credentials are not valid or you do not have access to the EC2 instance.

HTTP Status Code: 400

EC2InstanceNotFoundException

The specified instance was not found.

HTTP Status Code: 400

EC2InstanceStateInvalidException

Unable to connect because the instance is not in a valid state. Connecting to a stopped or terminated instance is not supported. If the instance is stopped, start your instance, and try to connect again.

HTTP Status Code: 400

EC2InstanceTypeInvalidException

The instance type is not supported for connecting via the serial console. Only Nitro instance types are currently supported.

HTTP Status Code: 400

EC2InstanceUnavailableException

The instance is currently unavailable. Wait a few minutes and try again.

HTTP Status Code: 400

InvalidArgsException

One of the parameters is not valid.

HTTP Status Code: 400

SerialConsoleAccessDisabledException

Your account is not authorized to use the EC2 Serial Console. To authorize your account, run the EnableSerialConsoleAccess API. For more information, see EnableSerialConsoleAccess in the *Amazon EC2 API Reference*.

HTTP Status Code: 400

SerialConsoleSessionLimitExceededException

The instance currently has 1 active serial console session. Only 1 session is supported at a time.

HTTP Status Code: 400

SerialConsoleSessionUnavailableException

Unable to start a serial console session. Please try again.

HTTP Status Code: 500

ServiceException

The service encountered an error. Follow the instructions in the error message and try again.

HTTP Status Code: 500

ThrottlingException

The requests were made too frequently and have been throttled. Wait a while and try again. To increase the limit on your request frequency, contact AWS Support.

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

SendSSHPublicKey

Pushes an SSH public key to the specified EC2 instance for use by the specified user. The key remains for 60 seconds. For more information, see Connect to your Linux instance using EC2 Instance Connect in the *Amazon EC2 User Guide*.

Request Syntax

```
{
   "AvailabilityZone": "string",
   "InstanceId": "string",
   "InstanceOSUser": "string",
   "SSHPublicKey": "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.

AvailabilityZone

The Availability Zone in which the EC2 instance was launched.

Type: String

Length Constraints: Minimum length of 6. Maximum length of 32.

Pattern: `^(\w+-){2,3}\d+\w+$`

Required: No

InstanceId

The ID of the EC2 instance.

Type: String

Length Constraints: Minimum length of 10. Maximum length of 32.

Pattern: `^i-[a-f0-9]+$`

Required: Yes

InstanceOSUser

The OS user on the EC2 instance for whom the key can be used to authenticate.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 32.

Pattern: `^[A-Za-z_][A-Za-z0-9\@\._-]{0,30}[A-Za-z0-9\$_-]?$`

Required: Yes

SSHPublicKey

The public key material. To use the public key, you must have the matching private key.

Type: String

Length Constraints: Minimum length of 80. Maximum length of 4096.

Required: Yes

Response Syntax

```
{
   "RequestId": "string",
   "Success": boolean
}
```

Response Elements

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

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

RequestId

The ID of the request. Please provide this ID when contacting AWS Support for assistance.

Type: String

Success

Is true if the request succeeds and an error otherwise.

Type: Boolean

Errors

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

AuthException

Either your AWS credentials are not valid or you do not have access to the EC2 instance.

HTTP Status Code: 400

EC2InstanceNotFoundException

The specified instance was not found.

HTTP Status Code: 400

EC2InstanceStateInvalidException

Unable to connect because the instance is not in a valid state. Connecting to a stopped or terminated instance is not supported. If the instance is stopped, start your instance, and try to connect again.

HTTP Status Code: 400

EC2InstanceUnavailableException

The instance is currently unavailable. Wait a few minutes and try again.

HTTP Status Code: 400

InvalidArgsException

One of the parameters is not valid.

HTTP Status Code: 400

ServiceException

The service encountered an error. Follow the instructions in the error message and try again.

HTTP Status Code: 500

ThrottlingException

The requests were made too frequently and have been throttled. Wait a while and try again. To increase the limit on your request frequency, contact AWS Support.

HTTP Status Code: 400

Examples

Push an SSH public key to an instance

This example sends the specified SSH public key to the specified instance in the specified Availability Zone. The key is used to authenticate the specified user.

Sample Request

```
POST / HTTP/1.1
Content-Type: application/x-amz-json-1.1
X-Amz-Target: AWSEC2InstanceConnectService.SendSSHPublicKey

{
   "AvailabilityZone": "us-east-2b",
   "InstanceId": "i-1234567890abcdef0",
   "InstanceOSUser": "ec2-user",
   "SSHPublicKey": "<ssh-public-key-material>"
}
```

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 AWS EC2 Instance Connect API has no separate data types.

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