

Troubleshooting Amazon Bedrock API Error Codes

This section provides detailed information about the common errors you might encounter when using Amazon Bedrock APIs, the cause of the error, and the solution for resolving the error.

AccessDeniedException

HTTP Status Code: 403

Cause: You do not have sufficient permissions to perform the requested action.

Solution:

- Verify that your IAM user or role has the necessary permissions for the action you are attempting.
- If you are using temporary security credentials, ensure they haven't expired.

FTUFormNotFilled

HTTP Status Code: 404

Cause: Model use case details have not been submitted for this account

Solution:

- Fill out the Anthropic use case details form before using the model

IncompleteSignature

HTTP Status Code: 400

Cause: The request signature does not conform to AWS standards.

Solution:

- Ensure you are using an AWS SDK version that supports Amazon Bedrock.
- Verify that your AWS access key ID and secret key are correctly configured.

- If you are manually signing requests, we suggest double-checking your signature calculation process.

InternalFailure

HTTP Status Code: 500

Cause: The request processing has failed due to a server error

Solution:

- We suggest employing AWS recommended approach of using [retries with exponential backoff](#) and random [jitter](#) for improved reliability.
- If the issue persists, please contact [AWS Support Center](#) and provide details about your request and the error you are encountering.

InvalidAction

HTTP Status Code: 400

Cause: The action or operation requested is invalid

Solution:

- We suggest double-checking the spelling and formatting of the action name in your request.
- Verify that the action calling is supported by Amazon Bedrock and is correctly documented as shown in [Amazon Bedrock API Reference](#).
- Ensure you are using the most up-to-date version of the AWS SDK or CLI.

InvalidClientId

HTTP Status Code: 403

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

Solution:

- Verify that you are using the correct AWS access key ID.

- If you recently created new access keys, ensure you are using the new credentials and not the old ones.

AWS Marketplace Agreement Failed within 15 minutes

HTTP Status Code: 403

Cause: The AWS Marketplace Agreement failed due to an underlying issue.

Solution:

- Review the error message and remediate the underlying issue. Common underlying issues are invalid payment error and restricted geo-location.
- For invalid payment error, please review [Restriction on credit and debit card purchases for AISPL customers using AWS Marketplace](#) and [INVALID_PAYMENT_INSTRUMENT after requesting model access in Amazon Bedrock..](#)

AWS Marketplace Agreement Pending after 15 minutes

HTTP Status Code: 403

Cause: The AWS Marketplace Agreement has not succeeded and it has been 15 minutes since the request was made.

Solution:

- Try the request again every 15 minutes. If the issue persists, please contact [AWS Support Center](#) and provide details about your request and the error you are encountering.

MPAgreementBeingCreated

HTTP Status Code: 403

Cause: Your account is not authorized to access this model. Your AWS Marketplace subscription for this model is still being processed

Solution:

- Try again after 15 minutes

NotAuthorized

HTTP Status Code: 400

Cause: You do not have permission to perform this action.

Solution:

- Review your IAM permissions and ensure you have the necessary rights to perform the requested action on Amazon Bedrock resources.
- If you are using an IAM role, verify that the role has the appropriate permissions and trust relationships.
- Check for any organizational policies or service control policies that might be restricting your access.

RequestExpired

HTTP Status Code: 400

Cause: The request is no longer valid due to expired timestamps.

Solution:

- Ensure your system clock is correctly synchronized with a reliable time source.
- If you are making requests from different time zones, be aware of potential timestamp discrepancies.

ServiceUnavailable

HTTP Status Code: 503

Cause: The service is temporarily unable to handle the request. 503 errors are used for regular throttling.

Solution:

- We suggest employing AWS recommended approach of using [retries with exponential backoff](#) and random [jitter](#) for improved reliability.
- Consider switching to a different AWS Region if the issue persists in your current Region. Different Regions may have varying levels of load and availability.
- [Use Cross-Region inference](#) to seamlessly manage unplanned traffic bursts by utilizing compute across different AWS Regions.
- If you have high throughput requirements, we suggest exploring [Provisioned Throughput](#) for your use case.

Best practices

- Ensure your application can handle 503 status codes appropriately in your error handling and retry logic.
- Check the AWS Service Health Dashboard for any announced issues or scheduled maintenance that might affect the service.

If you experience frequent 503 errors or if they significantly impact your operations, please contact [AWS Support](#) for further assistance and guidance tailored to your specific use case.

ThrottlingException

HTTP Status Code: 429

Cause: The request was denied due to exceeding the account quotas for Amazon Bedrock.

Solution:

- Check the Amazon Bedrock service quotas in the [Amazon Bedrock service quotas](#) console to learn about the limits allotted to your account.
- We suggest employing AWS recommended approach of using [retries with exponential backoff](#) and random [jitter](#) for improved reliability.
- If you have high throughput requirements, we suggest exploring [Provisioned Throughput](#) for your use case.
- Request for quota increase by contacting your account manager or [AWS Support](#) if your workload traffic exceeds your account quotas.

ValidationError

HTTP Status Code: 400

Cause: The input fails to satisfy the constraints specified by Amazon Bedrock.

Solution:

- Review the API documentation to ensure all required parameters are included and formatted correctly.
- Check that your input values are within the allowed ranges or conform to the expected patterns.
- We suggest paying attention to any specific validation rules mentioned in the API reference for the action you are using.

ResourceNotFound

HTTP Status Code: 404

Cause: The requested resource could not be found.

Solution:

- Verify the correctness of model ID, endpoint name, or other resource identifiers in your request.
- Please implement a fallback mechanism to use alternative models or endpoints when a primary resource is not found.

Best practices

- Use [ListFoundationModels](#) to learn about the available Amazon Bedrock foundation models that you can use.
- We suggest implementing a periodic synchronization process to update your local resource catalog.

If you continue to experience issues after trying these solutions, contact [AWS Support](#) for further assistance and guidance tailored to your specific use case.