

Integrating CogniSphere with Amazon Bedrock

Introduction

Amazon Bedrock is a fully managed service that provides access to a variety of high-performing foundation models (FMs) from leading AI companies such as Anthropic, AI21 Labs, Cohere, Meta, Mistral AI, Stability AI, and Amazon. These models are accessible via a single API, allowing organizations to develop advanced AI solutions efficiently.

This guide will walk you through the process of integrating the **CogniSphere** application with Amazon Bedrock. You will learn how to:

- Set up the required models on Bedrock
- Configure Identity and Access Management (IAM) permissions
- Test the application

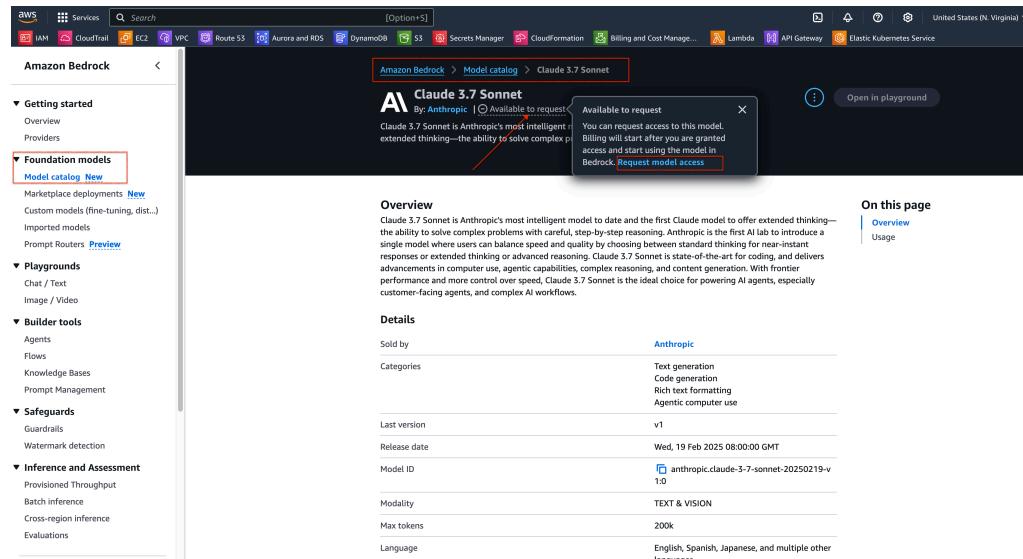
Prerequisites

Before proceeding, ensure that you have the following:

- ✓ An **AWS account** with the required permissions to complete the integration steps.
- ✓ **CogniSphere application** installed on your server.

1. Select Foundation Models (FMs)

1. Log in to the **AWS Management Console**.
2. Navigate to **Amazon Bedrock**.
3. Request access to one or more foundation models (FMs) you want to integrate with.



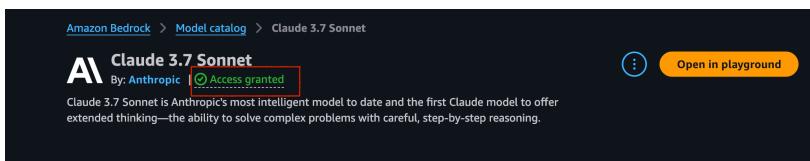
The screenshot shows the AWS Management Console with the Amazon Bedrock service selected. In the left sidebar, under 'Foundation models', the 'Model catalog' tab is active. The main content area displays the 'Claude 3.7 Sonnet' model card. A tooltip over the 'Available to request' button indicates that you can request access to this model. The 'Available to request' button is highlighted with a red arrow. The 'Overview' section describes Claude 3.7 Sonnet as Anthropic's most intelligent model, capable of solving complex problems with step-by-step reasoning. The 'Details' section lists the model's provider (Anthropic), categories (Text generation, Code generation, Rich text formatting, Agentic computer use), and various technical specifications like last version (v1), release date (Wed, 19 Feb 2025 08:00:00 GMT), and supported languages (English, Spanish, Japanese, and multiple other languages).

Note: This guide uses **Anthropic Claude 3.7 Sonnet** as an example. Not all models are available in all AWS regions.

4. Fill in the required details and submit the request.

The screenshot shows the 'Add use case details for Anthropic' step in the Amazon Bedrock Model access Request model access process. The form includes fields for Company name, Company website URL, What industry do you operate in?, Who are your intended users? (with options for Internal employees and External users), and a large text area for Describe your use cases (Do not share any PII or IP information). A note at the bottom states: 'Aggregated metrics (not including Your Content) arising from your use of Anthropic models on Amazon Bedrock will be shared with Anthropic for purposes of trust and safety.' Buttons for 'Cancel', 'Previous', and 'Next' are at the bottom.

5. Wait approximately **2 minutes** for access to be granted. You should see a confirmation once access is approved.



Overview
Claude 3.7 Sonnet is Anthropic's most intelligent model to date and the first Claude model to offer extended thinking—the ability to solve complex problems with careful, step-by-step reasoning. Anthropic is the first AI lab to introduce a single model where users can balance speed and quality by choosing between standard thinking for near-instant responses or extended thinking or advanced reasoning. Claude 3.7 Sonnet is state-of-the-art for coding, and delivers advancements in computer use, agentic capabilities, complex reasoning, and content generation. With frontier performance and more control over speed, Claude 3.7 Sonnet is the ideal choice for powering AI agents, especially customer-facing agents, and complex AI workflows.

On this page
[Overview](#)
[Usage](#)

2. Create IAM Access Keys

Step 1: Create a New IAM Policy

1. Go to **AWS Identity and Access Management (IAM)**.
2. Create a **new policy** with the required permissions for integration.

The screenshot shows the AWS IAM Policies list. The left sidebar shows 'Identity and Access Management (IAM)' and 'Policies'. The main area lists 1339 policies, with one named 'WAFV2LoggingServiceRolePolicy' highlighted. The table columns include 'Policy name', 'Type', 'Used as', and 'Description'. The 'Actions' and 'Create policy' buttons are at the top right.

Policy name	Type	Used as	Description
WorkLinkServiceRolePolicy	AWS managed	None	Enables access to AWS Services and Re...
WellArchitectedConsoleReadOnlyAccess	AWS managed	None	Provides read-only access to AWS Well-...
WellArchitectedConsoleFullAccess	AWS managed	None	Provides full access to AWS Well-Archit...
WAFV2LoggingServiceRolePolicy	AWS managed	None	This policy creates a service-linked ro...

Screenshot of the AWS IAM 'Create policy' wizard, Step 1: Specify permissions.

The Policy editor shows the following JSON code:

```

1 * {
2     "Version": "2012-10-17",
3     "Statement": [
4         {
5             "Action": [
6                 "bedrock:InvokeModel",
7                 "bedrock:InvokeModelWithResponseStream"
8             ],
9             "Effect": "Allow",
10            "Resource": "*",
11            "Sid": ""
12        }
13    ]
14 }

```

UI elements include tabs for Visual, JSON, Actions, and a 'Select a statement' dropdown with a '+ Add new statement' button.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": [
        "bedrock:InvokeModel",
        "bedrock:InvokeModelWithResponseStream"
      ],
      "Effect": "Allow",
      "Resource": "*",
      "Sid": ""
    }
  ]
}
```

3. Give the policy a meaningful name.

Screenshot of the AWS IAM 'Create policy' wizard, Step 2: Review and create.

Policy details

Policy name: cognisphere_policy

Description - optional: A short explanation for this policy.

Permissions defined in this policy

Allow (1 of 439 services)

Service	Access level	Resource	Request condition
Bedrock	Limited: Read	All resources	None

Add tags - optional

Tags are key-value pairs that you can add to AWS resources to help identify, organize, or search for resources.

You can add up to 50 more tags.

Buttons at the bottom: Cancel, Previous, Create policy (highlighted).

Step 2: Create an IAM User

1. Navigate to **IAM > Users** and create a new user for integration.
2. Assign the **previously created policy** to this user.

If you're creating programmatic access through access keys or service-specific credentials for AWS CodeCommit or Amazon Keyspaces, you can generate them after you create this IAM user.

Permissions policies (1/1339)
Choose one or more policies to attach to your new user.

Name	Type	Used as
cognisphere-policy	Customer managed	Permissions policy

Review and create
Review your choices. After you create the user, you can view and download the autogenerated password, if enabled.

User details	Console password type	Require password reset
User name cognisphere_user	None	No

Permissions summary

Name	Type	Used as
cognisphere-policy	Customer managed	Permissions policy

Tags - optional
Tags are key-value pairs you can add to AWS resources to help identify, organize, or search for resources. Choose any tags you want to associate with this user.
No tags associated with the resource.

Add new tag
You can add up to 50 more tags.

Step 3: Generate Access Keys

1. Open the **Security Credentials** tab for the IAM user.
2. Click **Create Access Key**.
3. Choose **Application running outside AWS** as the use case.
4. Complete the credential creation process.
5. **Save the Access Key ID and Secret Access Key securely.**

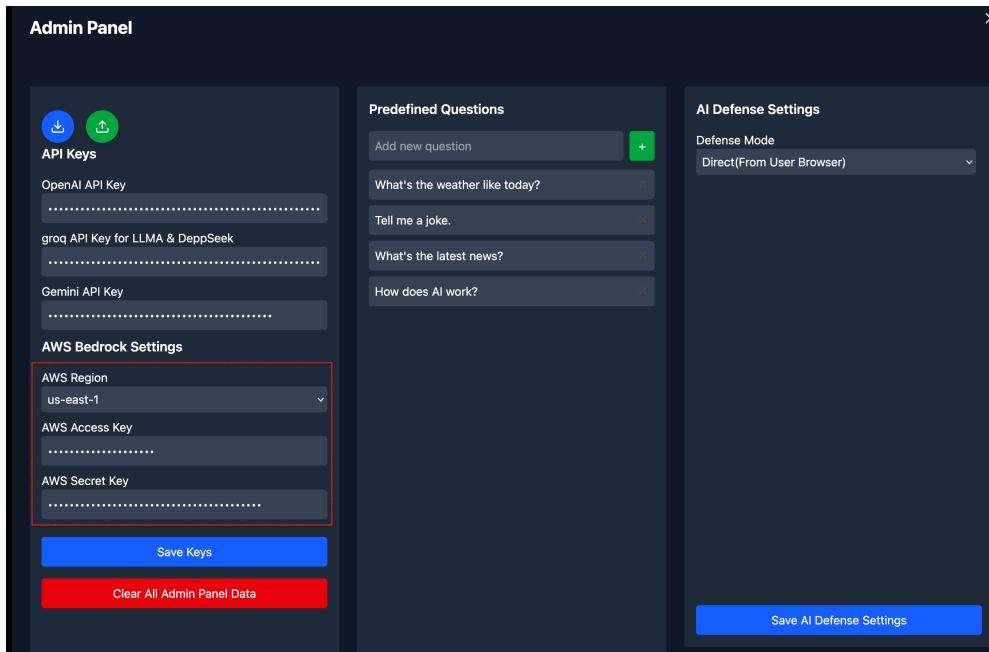
The screenshot shows the 'Security credentials' tab selected in the IAM console. Under the 'Access keys' section, there is a button labeled 'Create access key'. This button is highlighted with a red box.

This is the first step of the 'Create access key' wizard. It asks for a use case. The option 'Application running outside AWS' is selected and highlighted with a blue box. Other options include 'Command Line Interface (CLI)', 'Local code', 'Application running on an AWS compute service', 'Third-party service', and 'Other'.

This is the second step of the wizard. It shows the generated access key and secret access key. The access key is 'AKIA...', and the secret access key is partially visible as '***** Show'. Both fields are highlighted with red boxes. A note at the bottom says, 'This is the only time that the secret access key can be viewed or downloaded. You cannot recover it later. However, you can create a new access key any time.'

3. Configure CogniSphere with AWS Credentials

1. Open the **CogniSphere** application and navigate to **Admin Panel**.
2. Enter the following details in the configuration settings:
 - o **AWS Region**
 - o **AWS Access Key ID**
 - o **AWS Secret Access Key**
3. Save the configuration.



Your **CogniSphere application** is now integrated with Amazon Bedrock! We can test

