



Chat Lambda IAM Permissions

AWS Energy Data Insights Platform | Agent Router & Orchestrator

Overview

Service Role: ChatLambdaRole

Function Name: chat

Purpose: Routes user queries to specialized agents, manages conversation state, invokes tool Lambdas, and generates AI responses

Timeout: 300 seconds | **Memory:** 1024 MB

DynamoDB Permissions

dynamodb:PutItem

REQUIRED

Resources:

- arn:aws:dynamodb:*:*:table/ChatMessage-*
- arn:aws:dynamodb:*:*:table/ChatSession-*
- arn:aws:dynamodb:*:*:table/SessionContext-*

Save chat messages, create/update sessions, and persist session context. Critical for conversation history and state management.

dynamodb:GetItem

REQUIRED

Resources: Same as PutItem

Retrieve individual messages and session data. Used for context retrieval and message updates.

dynamodb:Query

REQUIRED

Resources:

- arn:aws:dynamodb:*:*:table/ChatMessage-*/index/*
- arn:aws:dynamodb:*:*:table/ChatSession-*/index/*

Query conversation history using GSI (chatSessionId-createdAt-index). Essential for retrieving message history for context.

dynamodb:UpdateItem

REQUIRED

Resources: Same as PutItem

Update existing messages (e.g., marking as complete) and session metadata (e.g., lastMessageAt timestamp).

S3 Permissions

s3:GetObject

REQUIRED

Resource: arn:aws:s3:::storage-bucket/*

Read LAS files for petrophysical analysis and retrieve generated artifacts. Required for well data discovery and analysis workflows.

s3:PutObject

REQUIRED

Resource: arn:aws:s3:::storage-bucket/*

Store generated artifacts (visualizations, reports, analysis results). Critical for artifact persistence and retrieval.

Bedrock Permissions

bedrock:InvokeModel

REQUIRED

Resource: arn:aws:bedrock:*:*:foundation-model/anthropic.claude-*

Invoke Claude 3.5 Sonnet for AI response generation. Used by general knowledge agent and other agents requiring LLM capabilities.

bedrock:InvokeModelWithResponseStream

OPTIONAL

Resource: arn:aws:bedrock:*:*:foundation-model/anthropic.claude-*

Enable streaming responses for real-time user experience. Optional but recommended for improved UX.

Lambda Invocation Permissions

lambda:InvokeFunction

REQUIRED

Resources:

- arn:aws:lambda:*:*:function:petrophysics-calculator
- arn:aws:lambda:*:*:function:renewable-orchestrator
- arn:aws:lambda:*:*:function:-chat (self-invocation)

Invoke specialized tool Lambdas for domain-specific processing. Self-invocation enables async processing pattern for long-running queries.

CloudWatch Logs Permissions

logs:CreateLogGroup

REQUIRED

Resource: arn:aws:logs:*:log-group:/aws/lambda/chat:*

Standard Lambda logging permission. Required for CloudWatch log group creation.

logs:CreateLogStream

REQUIRED

Resource: arn:aws:logs:*log-group:/aws/lambda/chat:*

Create log streams for each Lambda invocation. Essential for debugging and monitoring.

logs:PutLogEvents

REQUIRED

Resource: arn:aws:logs:log-group:/aws/lambda/chat:*

Write log events to CloudWatch. Critical for troubleshooting agent routing and processing issues.

 **Complete IAM Policy JSON**

```
{ "Version": "2012-10-17", "Statement": [ { "Effect": "Allow", "Action": [ "dynamodb:PutItem", "dynamodb:GetItem", "dynamodb:Query", "dynamodb:UpdateItem" ], "Resource": [ "arn:aws:dynamodb:*:table/ChatMessage-*", "arn:aws:dynamodb:*:table/ChatMessage-*/index/*", "arn:aws:dynamodb:*:table/ChatSession-*", "arn:aws:dynamodb:*:table/SessionContext-*" ] }, { "Effect": "Allow", "Action": [ "s3:GetObject", "s3:PutObject" ], "Resource": "arn:aws:s3:::storage-bucket/*" }, { "Effect": "Allow", "Action": [ "bedrock:InvokeModel", "bedrock:InvokeModelWithResponseStream" ], "Resource": "arn:aws:bedrock:*:foundation-model/anthropic.claude-*" }, { "Effect": "Allow", "Action": "lambda:InvokeFunction", "Resource": [ "arn:aws:lambda:*:function:petrophysics-calculator", "arn:aws:lambda:*:function:renewable-orchestrator", "arn:aws:lambda:*:function:-chat" ] }, { "Effect": "Allow", "Action": [ "logs:CreateLogGroup", "logs:CreateLogStream", "logs:PutLogEvents" ], "Resource": "arn:aws:logs:*log-group:/aws/lambda/chat:*
```

⚠ Self-Invocation Pattern: The chat Lambda can invoke itself asynchronously to handle long-running queries that exceed API Gateway's 29-second timeout. This requires the `lambda:InvokeFunction` permission on its own function ARN.

💡 Security Note: This role has broad permissions due to its central orchestration role. In production, consider splitting into separate roles for different agent types or implementing resource-based policies for finer-grained control.