Step 1: Create the IAM Role for Lambda

1. Go to **IAM** → **Roles** → **Create Role**

2. **Trusted entity:** Choose **AWS Service** → **Lambda**

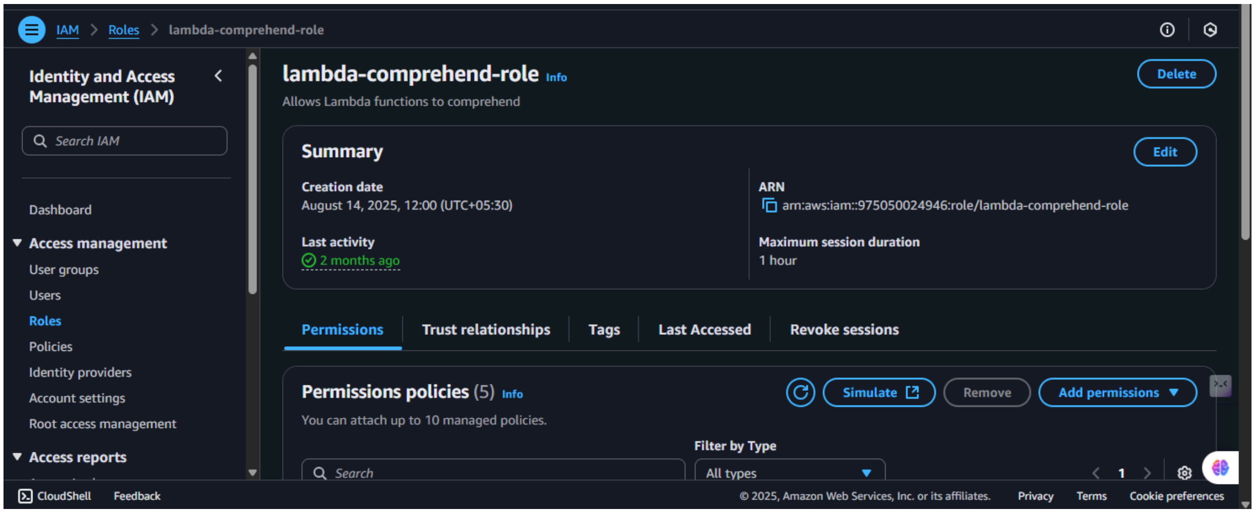
3. **Permissions policies:** Attach:

   ○ `ComprehendFullAccess` *(for simplicity; for production, prefer least privilege — e.g., `comprehend:DetectSentiment` only)*

   ○ `CloudWatchLogsFullAccess` *(so Lambda can log output)*

4. **Name:** `LambdaComprehendRole`

5. Create the role and **note the ARN** — you'll attach it to the Lambda function.



lambda-comprehend-role Info

Allows Lambda functions to comprehend

Summary

Creation date: August 14, 2025, 12:00 (UTC+05:30)

ARN: arn:aws:iam::975050024946:role/lambda-comprehend-role

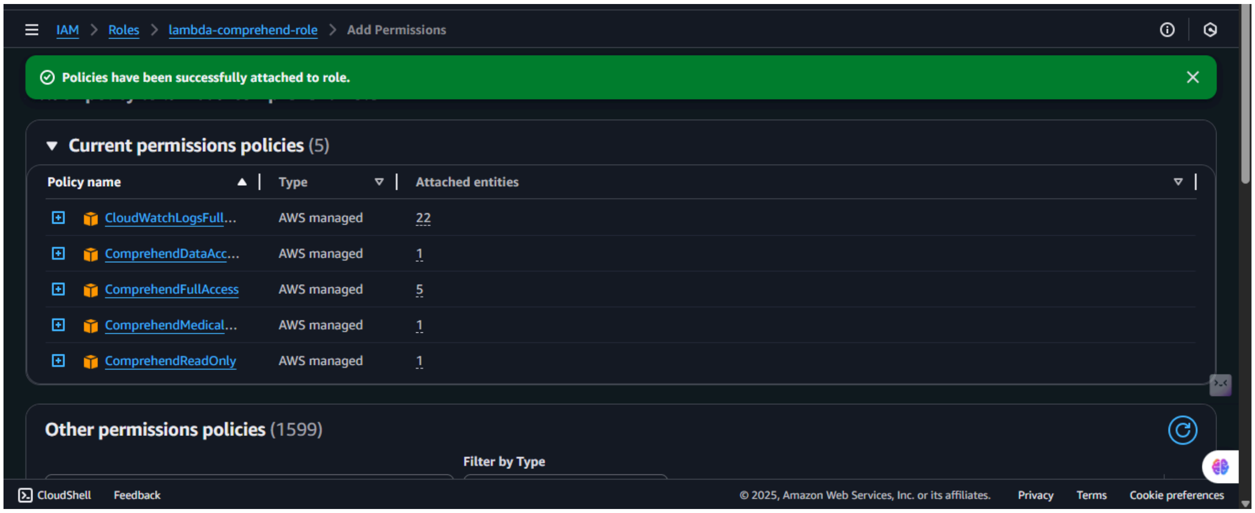
Last activity: 2 months ago

Maximum session duration: 1 hour

Permissions policies (5) Info

You can attach up to 10 managed policies.

Filter by Type: All types



IAM > Roles > lambda-comprehend-role > Add Permissions

Policies have been successfully attached to role.

Current permissions policies (5)

Policy name	Type	Attached entities
CloudWatchLogsFull...	AWS managed	22
ComprehendDataAcc...	AWS managed	1
ComprehendFullAccess	AWS managed	5
ComprehendMedical...	AWS managed	1
ComprehendReadOnly	AWS managed	1

Other permissions policies (1599)

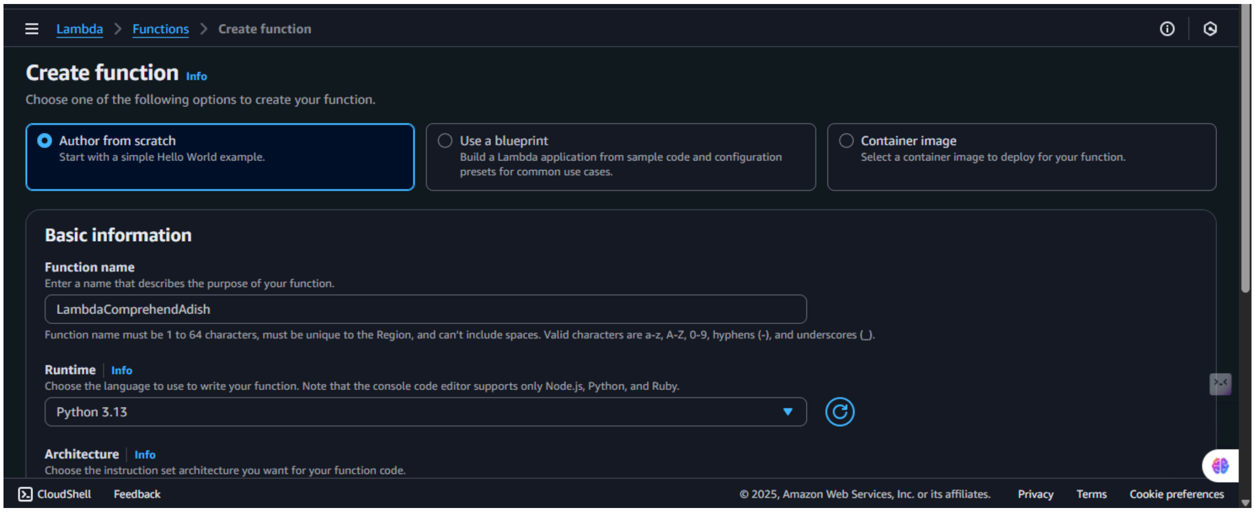
Filter by Type

CloudShell Feedback © 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

2: Create the Lambda Function

1. Go to **AWS Lambda** → **Create function**
   - Runtime: **Python 3.13** (or 3.11)
   - Execution role: **Use existing role** → **LambdaComprehend**
2. In the **Function Code** section, paste the following:

```
import json

import boto3

import logging

# Set up logging

logger = logging.getLogger()

logger.setLevel(logging.INFO)

# Create Comprehend client

comprehend = boto3.client('comprehend')

def lambda_handler(event, context):
    """
    Lambda function to analyze sentiment of user reviews using Amazon Comprehend
    """
    try:
        # Extract review text from the incoming event
        review_text = event.get('review')
        if not review_text:
            logger.error("No review text found in event.")
            return {
                'statusCode': 400,
                'body': json.dumps({'error': 'Missing review text'})
            }

        logger.info(f"Received review: {review_text}")

        # Call Amazon Comprehend for sentiment analysis
        response = comprehend.detect_sentiment(
            Text=review_text,
            LanguageCode='en'
        )

        sentiment = response['Sentiment']
        sentiment_scores = response['SentimentScore']
        logger.info(f"Detected Sentiment: {sentiment}")
        logger.info(f"Sentiment Scores: {json.dumps(sentiment_scores, indent=2)}")

        # Return result
        return {
            'statusCode': 200,
            'body': json.dumps({
                'review': review_text,
                'sentiment': sentiment,
                'scores': sentiment_scores
            })
        }
    except Exception as e:
        logger.error(f"Error processing review: {str(e)}")
        return {
            'statusCode': 500,
            'body': json.dumps({'error': str(e)})
        }
```


Lambda > Functions > Create function
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

Create function Info

Choose one of the following options to create your function.

● Author from scratch
Start with a simple Hello World example.

○ Use a blueprint
Build a Lambda application from sample code and configuration presets for common use cases.

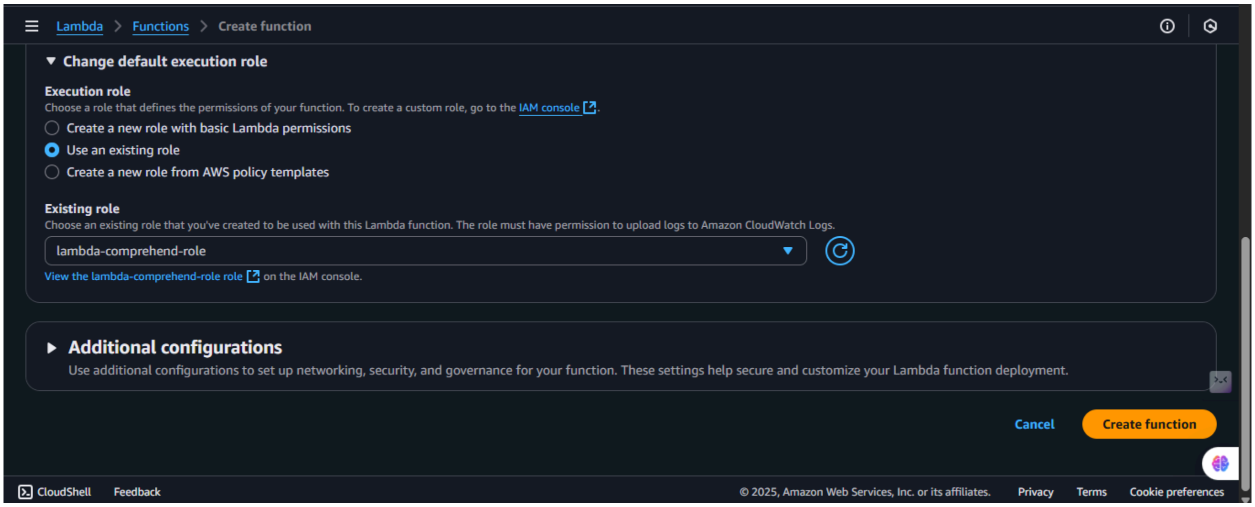
○ Container image
Select a container image to deploy for your function.

Basic information

Function name
Enter a name that describes the purpose of your function.
LambdaComprehendAdish
Function name must be 1 to 64 characters, must be unique to the Region, and can't include spaces. Valid characters are a-z, A-Z, 0-9, hyphens (-), and underscores (_).

Runtime | Info
Choose the language to use to write your function. Note that the console code editor supports only Node.js, Python, and Ruby.
Python 3.13

Architecture | Info
Choose the instruction set architecture you want for your function code.

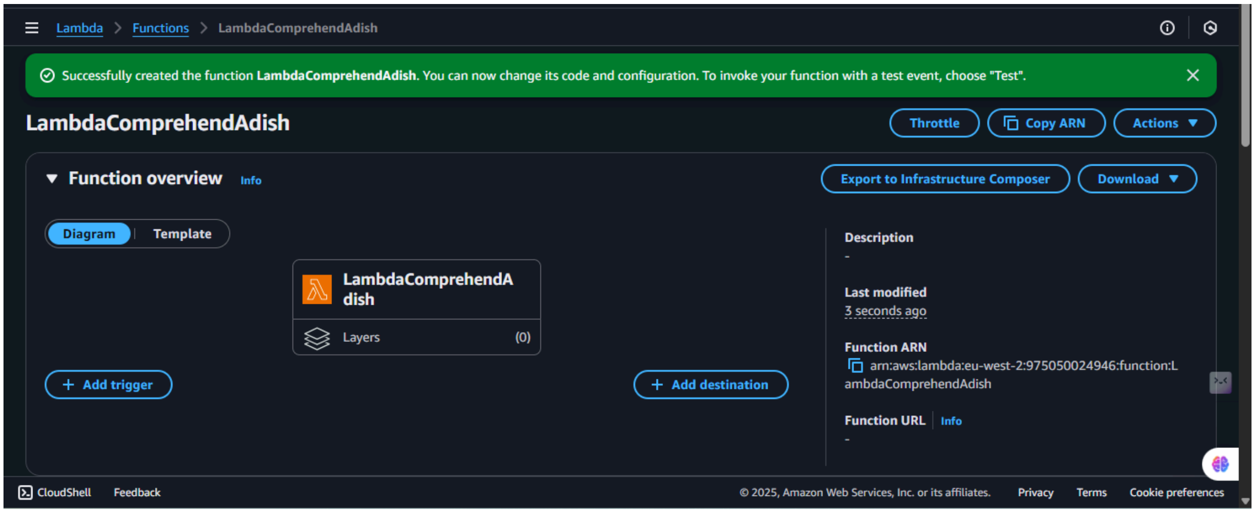

Lambda > Functions > Create function
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

▼ Change default execution role

Execution role
Choose a role that defines the permissions of your function. To create a custom role, go to the IAM console.
○ Create a new role with basic Lambda permissions
● Use an existing role
○ Create a new role from AWS policy templates

Existing role
Choose an existing role that you've created to be used with this Lambda function. The role must have permission to upload logs to Amazon CloudWatch Logs.
lambda-comprehend-role
View the lambda-comprehend-role role on the IAM console.

▶ Additional configurations
Use additional configurations to set up networking, security, and governance for your function. These settings help secure and customize your Lambda function deployment.



Successfully created the function **LambdaComprehendAdish**. You can now change its code and configuration. To invoke your function with a test event, choose "Test".

LambdaComprehendAdish

▼ Function overview Info

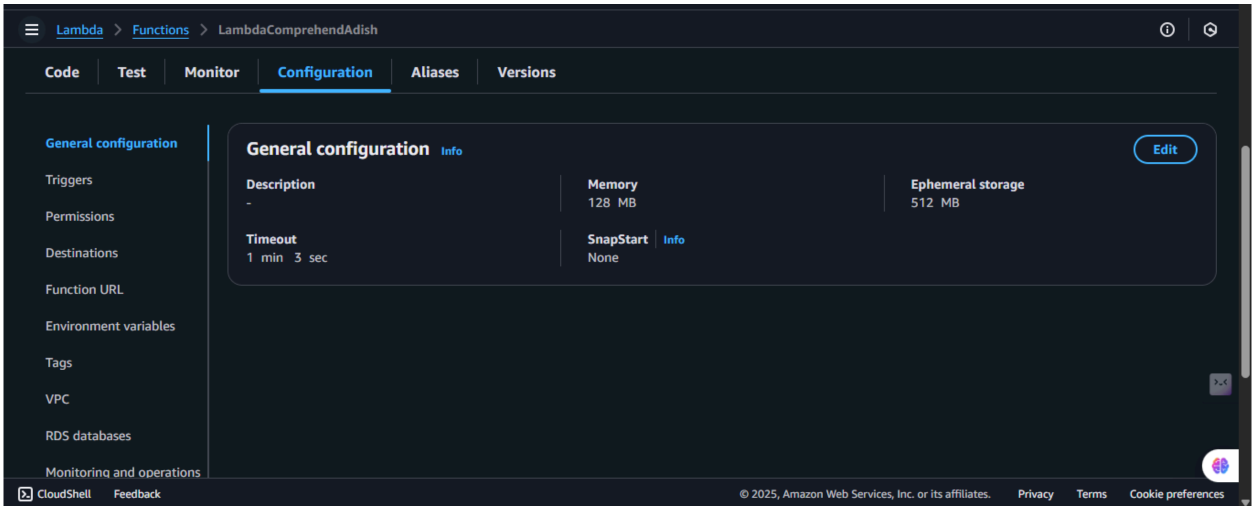
LambdaComprehendAdish

Description
-

Last modified
3 seconds ago

Function ARN
arn:aws:lambda:eu-west-2:975050024946:function:LambdaComprehendAdish

Function URL | Info
-



General configuration

Description: -

Memory: 128 MB

Ephemeral storage: 512 MB

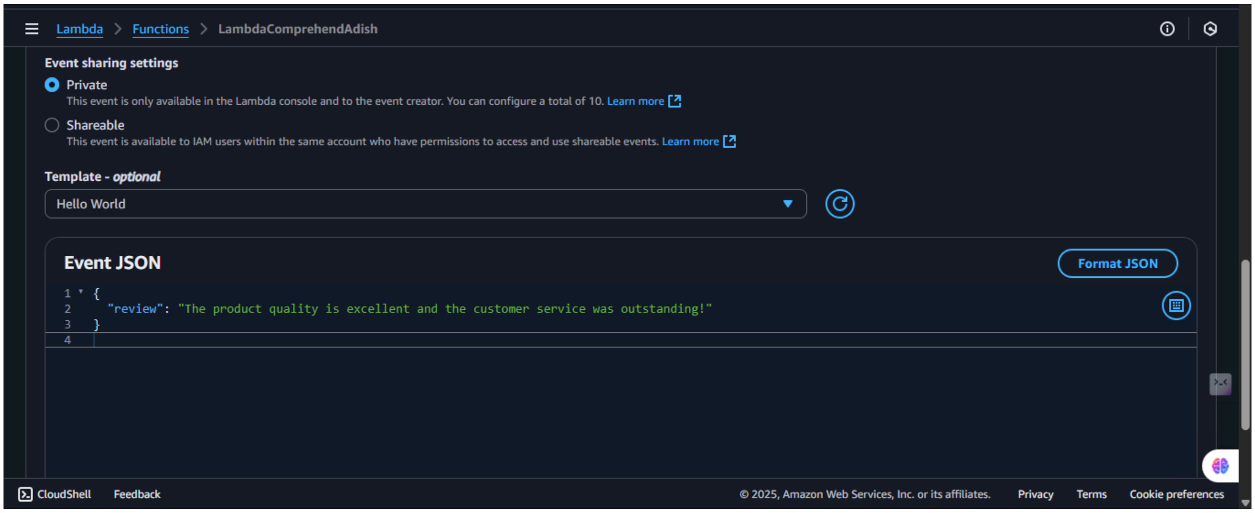
Timeout: 1 min 3 sec

SnapStart: None

Testing the Lambda Function

1. Go to **Test** → **Configure test event**
2. Choose **“Create new test event”**
3. Name it: `PositiveReview`
4. Add this sample input:

```
{
  "review": "The product quality is excellent and the customer service was outstanding!"
}
```



Event sharing settings

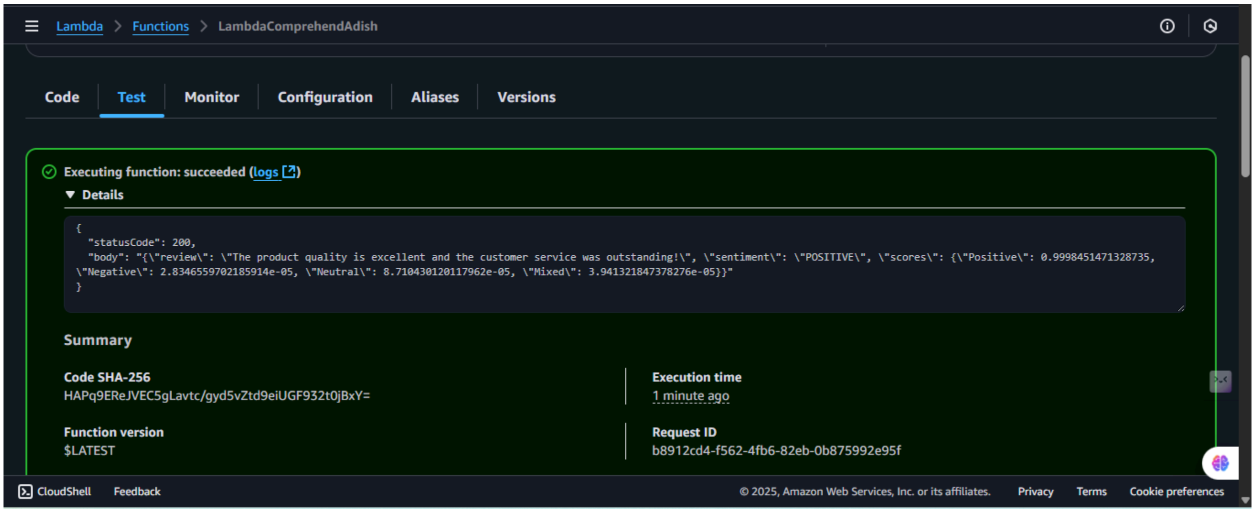
● Private
This event is only available in the Lambda console and to the event creator. You can configure a total of 10. Learn more

○ Shareable
This event is available to IAM users within the same account who have permissions to access and use shareable events. Learn more

Template - *optional*
Hello World

Event JSON

```
{
  "review": "The product quality is excellent and the customer service was outstanding!"
}
```



Executing function: succeeded (logs)

Details

```
{
  "statusCode": 200,
  "body": "{\"review\": \"The product quality is excellent and the customer service was outstanding!\", \"sentiment\": \"POSITIVE\", \"scores\": {\"Positive\": 0.9998451471328735,
\"Negative\": 2.8346559702185914e-05, \"Neutral\": 8.710430120117962e-05, \"Mixed\": 3.941321847378276e-05}}"
}
```

Summary

Code SHA-256
HAPq9EReJVEC5gLavtc/gyd5vZtd9eiUGF932t0jBxY=

Execution time
1 minute ago

Function version
$LATEST

Request ID
b8912cd4-f562-4fb6-82eb-0b875992e95f

In the **Execution logs**, you should see output like

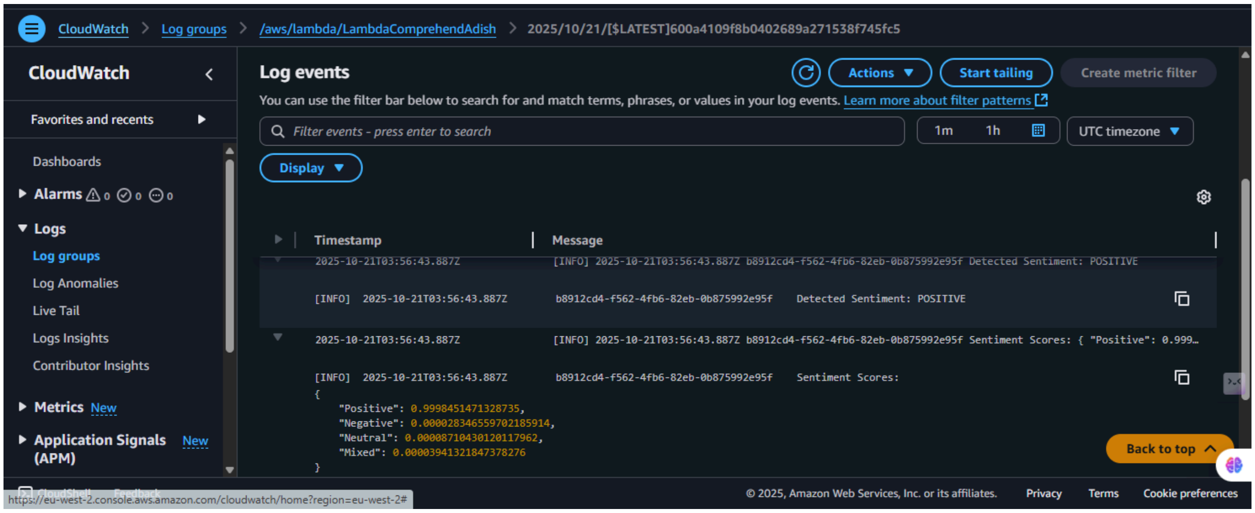
```
INFO  Detected Sentiment: POSITIVE

INFO  Sentiment Scores: {

  "Positive": 0.98,

  "Negative": 0.01,

  "Neutral": 0.01,

  "Mixed": 0.00

}
```

View Logs in CloudWatch

1. Go to **CloudWatch** → **Logs** → **Log groups**

2. Find `/aws/lambda/<YourLambdaName>`

3. View detailed logs for sentiment analysis results.


CloudWatch > Log groups > /aws/lambda/LambdaComprehendAdish > 2025/10/21/[$LATEST]600a4109f8b0402689a271538f745fc5
CloudWatch
Favorites and recents
Dashboards
Alarms 0 0 0
Logs
Log groups
Log Anomalies
Live Tail
Logs Insights
Contributor Insights
Metrics New
Application Signals New (APM)
Log events
Actions
Start tailing
Create metric filter
Filter events - press enter to search
1m
1h
UTC timezone
Display
Timestamp
Message
Back to top
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences
https://eu-west-2.console.aws.amazon.com/cloudwatch/home?region=eu-west-2#

You can use the filter bar below to search for and match terms, phrases, or values in your log events. Learn more about filter patterns

```
2025-10-21T03:56:43.887Z    [INFO] 2025-10-21T03:56:43.887Z b8912cd4-f562-4fb6-82eb-0b875992e95f Detected Sentiment: POSITIVE

[INFO]  2025-10-21T03:56:43.887Z      b8912cd4-f562-4fb6-82eb-0b875992e95f    Detected Sentiment: POSITIVE

2025-10-21T03:56:43.887Z    [INFO] 2025-10-21T03:56:43.887Z b8912cd4-f562-4fb6-82eb-0b875992e95f Sentiment Scores: { "Positive": 0.999…

[INFO]  2025-10-21T03:56:43.887Z      b8912cd4-f562-4fb6-82eb-0b875992e95f    Sentiment Scores:
{
    "Positive": 0.9998451471328735,
    "Negative": 0.000028346559702185914,
    "Neutral": 0.00008710430120117962,
    "Mixed": 0.000039413218473782 76
}
```