

# AmazonConnectによる自動電話通知（4.Lambdaによる起動）

AWS



×

ZABBIX

×

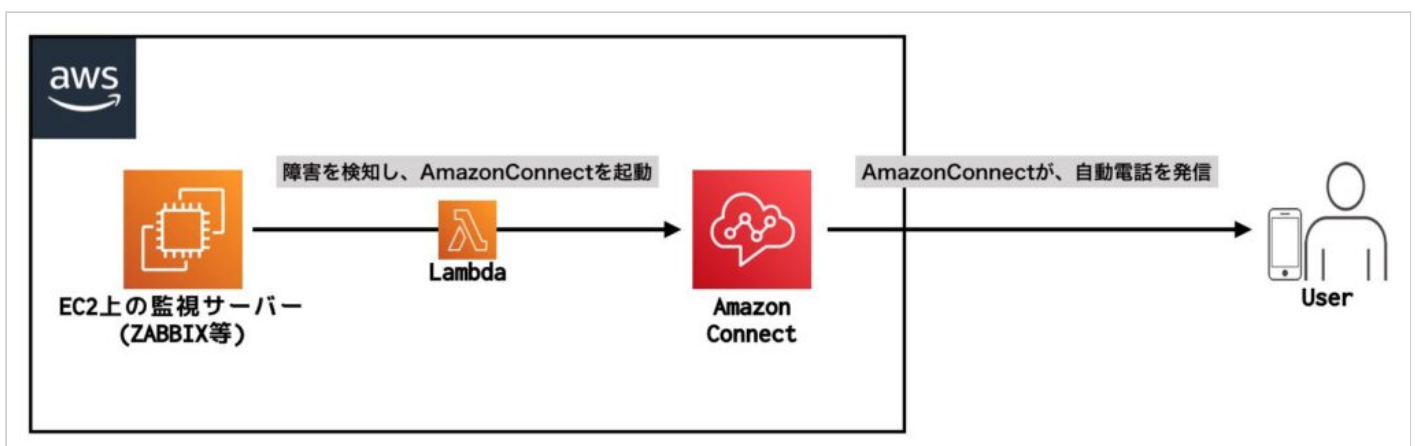


## AmazonConnectによる自動電話通知 （4.Lambdaによる起動）

2021.11.12 2021.10.14

[【前回】 AmazonConnectによる自動電話通知（3.問い合わせフローの作成）](#)[【次回】 AmazonConnectによる自動電話通知（5.EC2との連携）](#)[【簡易版】 AmazonConnectによる自動電話通知（まとめ）](#)

監視サーバーで障害を検知した際に、自動で電話通知できるようにしていきます。ネットワークエンジニアも利用することの多い監視サーバー(Zabbix)で障害検知し、AWS上のAmazonConnectを利用し自動電話を発信します。

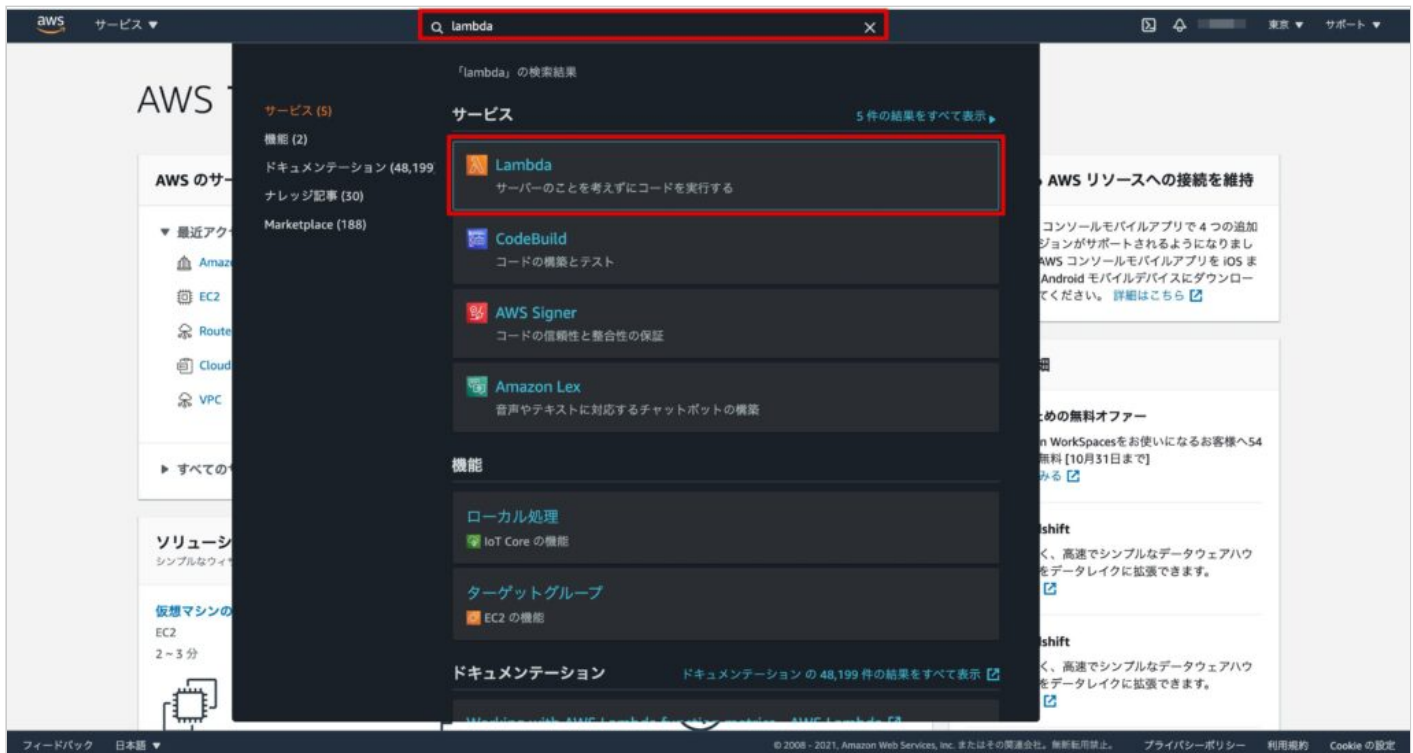


自動電話通知イメージ

## Lambda関数の作成

## 関数の作成

AWSマネジメントコンソール上で、「lambda」を検索します。



「関数の作成」をクリックします。



下記の通り入力し、「関数の作成」をクリックします。

オプション：一から作成を選択

関数名：任意の名前を入力 ※ここでは、“AmazonConnectTest-001”としています。

ランタイム : Pythonを選択 ※ここでは、最新版の“Python 3.9”を選択しています。

Lambda > 関数 > 関数の作成

## 関数の作成 Info

以下のいずれかのオプションを選択して、関数を作成します。

**一から作成** Info  
シンプルな Hello World の例で開始します。

設計図の使用  
一般的ユースケース用のサンプルコードと設定プリセットから Lambda アプリケーションを構築します。

コンテナイメージ  
関数にデプロイするコンテナイメージを選択します。

Serverless Application Repository の参照  
AWS Serverless Application Repository からサンプル Lambda アプリケーションをデプロイします。

### 基本的な情報

関数名  
関数の目的を名前として入力します。  
**AmazonConnectTest-001**  
半角英数字、ハイフン、アンダースコアのみを使用でき、スペースは使用できません。

ランタイム Info  
関数の記述に使用する言語を選択します。コンソールコードエディタは Node.js、Python、および Ruby のみをサポートすることに注意してください。  
**Python 3.9**

アーキテクチャ Info  
関数コードに必要な命令セットアーキテクチャを選択します。  
☒ x86\_64  
☐ arm64

アクセス権限 Info  
デフォルトでは、Lambda は Amazon CloudWatch Logs にログをアップロードするアクセス許可を持つ実行ロールを作成します。このデフォルトのロールは、後でトリガーを追加するときにカスタマイズできます。  
[▶ デフォルトの実行ロールの変更](#)

[▶ 詳細設定](#)

キャンセル **関数の作成**



関数が作成されたことを確認します。

Lambda > 関数 > AmazonConnectTest-001

## AmazonConnectTest-001

スロットリング ARN をコピー アクション ▼


**▼ 関数の概要 Info**

 **AmazonConnectTest-001**  
 Layers (0)

[+ トリガーを追加](#) [+ 送信先を追加](#)

説明  
-

最終更新  
4 秒前

関数の ARN  
 `arn:aws:lambda:ap-northeast-1: :function:AmazonConnectTest-001`

コード テスト モニタリング 設定 エイリアス バージョン

### コードソース Info

[アップロード元 ▼](#)

File Edit Find View Go Tools Window **Test** Deploy **Changes deployed**

Go to Anything (35 P)

Environment  
AmazonConnectTest  
lambda\_function.py

1 `report.json`  
2  
3 `def lambda_handler(event, context):`  
4 `# TODO`  
5 `return`  
6

Code change

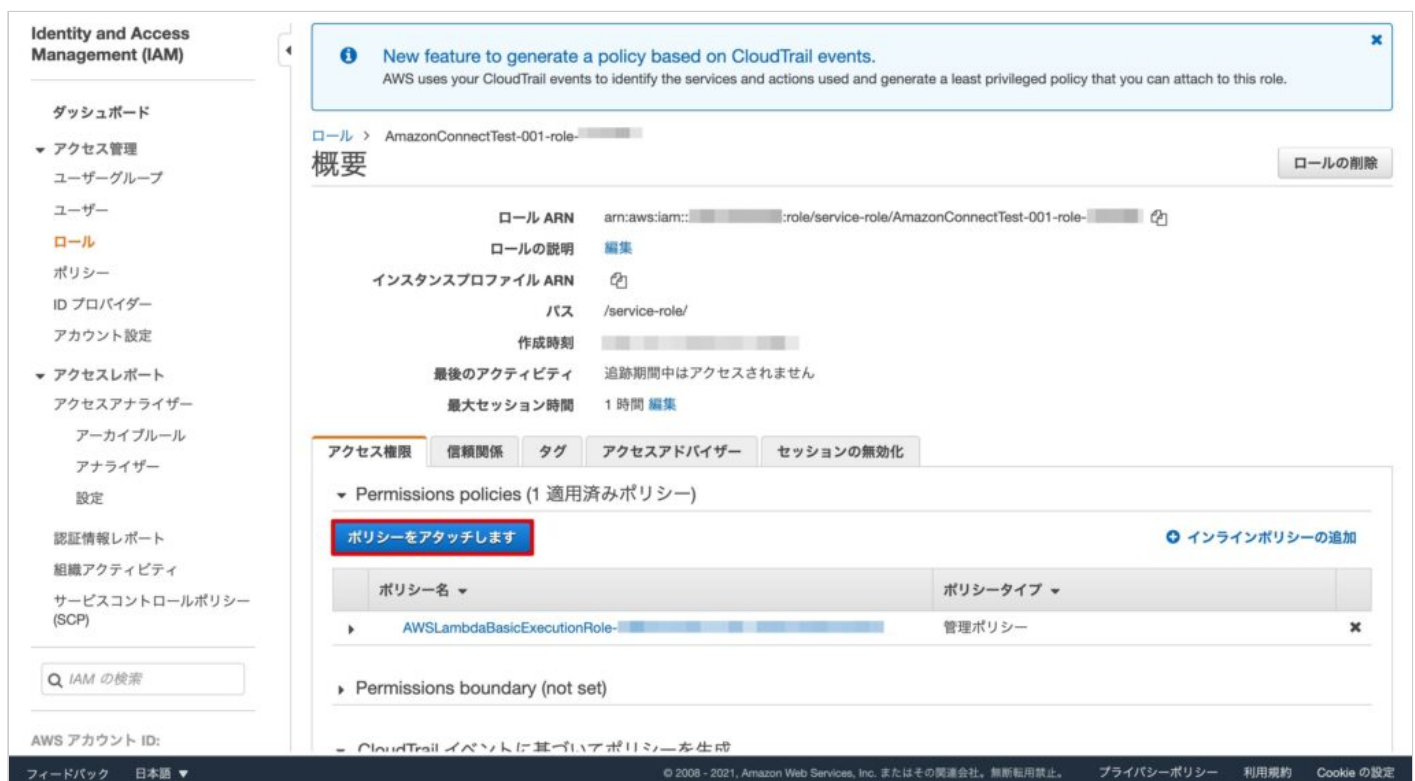
フィードバック 日本語 ▼ © 2008 - 2021, Amazon Web Services, Inc. またはその関連会社。無断転用禁止。 プライバシーポリシー 利用規約 Cookie の設定

## アクセス権の追加（ロールの設定）

作成された関数の設定タブに移動し、実行ロールをクリックします。



「ポリシーをアタッチします」をクリックします。



“connect”で検索し、「AmazonConnect\_FullAccess」にチェックを入れ、「ポリシーのアタッチ」をクリックします。

### AmazonConnectTest-001-role- にアクセス権限を追加する

#### アクセス権限をアタッチする

ポリシーの作成

ポリシーのフィルタ  10 件の結果を表示中

ポリシー名	タイプ	次として使用
<input checked="" type="checkbox"/> AmazonConnect_FullAccess	AWS による管理	Permissions policy (1)
<input type="checkbox"/> AmazonConnectReadOnlyAccess	AWS による管理	なし
<input type="checkbox"/> AmazonConnectVoiceIDFullAccess	AWS による管理	なし
<input type="checkbox"/> AmazonMSKConnectReadOnlyAccess	AWS による管理	なし
<input type="checkbox"/> AWSConnector	AWS による管理	なし
<input type="checkbox"/> AWSDirectConnectFullAccess	AWS による管理	なし
<input type="checkbox"/> AWSDirectConnectReadOnlyAccess	AWS による管理	なし
<input type="checkbox"/> EC2InstanceConnect	AWS による管理	なし
<input type="checkbox"/> ServerMigrationConnector	AWS による管理	なし
<input type="checkbox"/> VMImportExportRoleForAWSConnector	AWS による管理	なし

キャンセル **ポリシーのアタッチ**

フィードバック 日本語 ▼ © 2008 - 2021, Amazon Web Services, Inc. またはその関連会社。無断転用禁止。 プライバシーポリシー 利用規約 Cookie の設定

ポリシーがアタッチされたことを確認します。

### Identity and Access Management (IAM)

ダッシュボード

- アクセス管理
- ユーザーグループ
- ユーザー
- ロール**
- ポリシー
- ID プロバイダー
- アカウント設定

アクセスレポート

- アクセスアナライザー
- アーカイブルール
- アナライザー
- 設定

認証情報レポート

組織アクティビティ

サービスコントロールポリシー (SCP)

Q IAM の検索

AWS アカウント ID:

**概要**

ロール ARN: `arn:aws:iam:::role/service-role/AmazonConnectTest-001-role-`

ロールの説明: [編集](#)

インスタンスプロフィール ARN: [編集](#)

パス: `/service-role/`

作成時刻: [編集](#)

最後のアクティビティ: 追跡期間中はアクセスされません

最大セッション時間: 1 時間 [編集](#)

ロールの削除

アクセス権限

Permissions policies (2 適用済みポリシー)

**ポリシーをアタッチします** [インラインポリシーの追加](#)

ポリシー名	ポリシータイプ
<input type="checkbox"/> AWSLambdaBasicExecutionRole-	管理ポリシー
<input checked="" type="checkbox"/> AmazonConnect_FullAccess	AWS 管理ポリシー

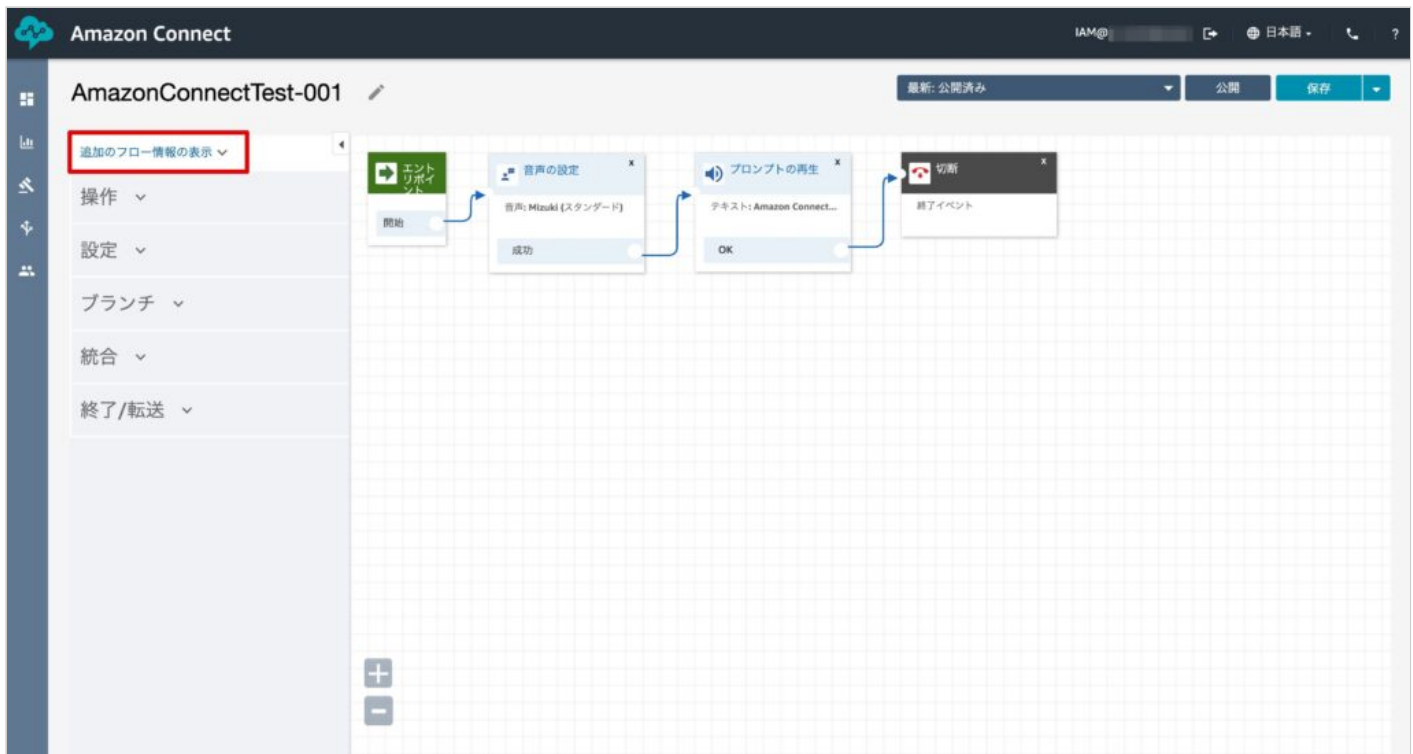
Permissions boundary (not set)

フィードバック 日本語 ▼ © 2008 - 2021, Amazon Web Services, Inc. またはその関連会社。無断転用禁止。 プライバシーポリシー 利用規約 Cookie の設定

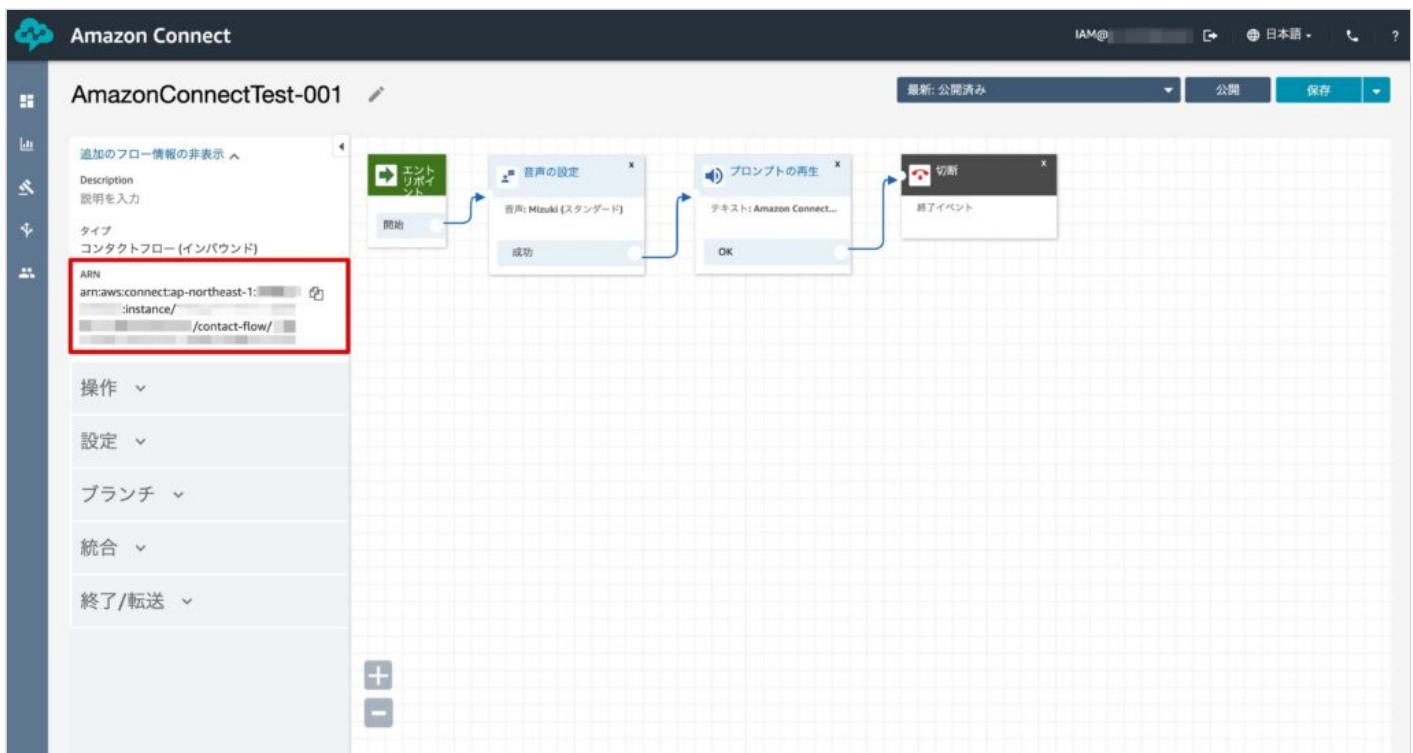
## AmazonConnectのインスタンスIDと問い合わせフローIDを確認

AmazonConnectの問い合わせフロー画面で、「通知のフロー情報の表示」を展開します。





ARNが表示されます。instanceの後ろの文字列がインスタンスID、contact-flowの後ろの文字列が問い合わせフローIDとなりますので、メモしておきます。



## コードの記述

Lambdaのコードを記述します。

```
import boto3
```

```
def lambda_handler(event, context):

    connect = boto3.client('connect')

    connect.start_outbound_voice_contact(
        DestinationPhoneNumber='+81*****',
        ContactFlowId='*****',
        InstanceId='*****',
        SourcePhoneNumber='+81*****',
    )
```

*DestinationPhoneNumber* : 発信先の電話番号

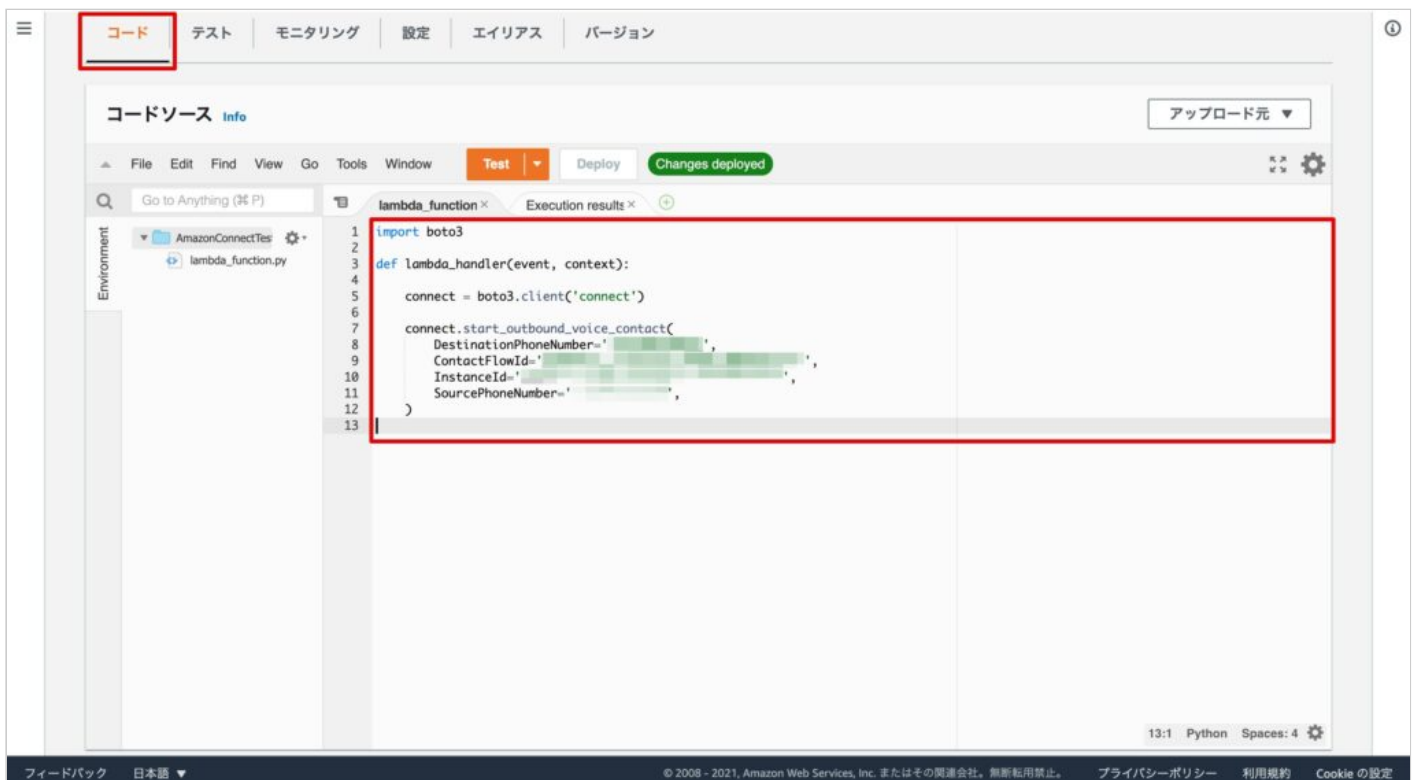
※国番号をつけて記述(日本の090の番号の場合、+8190\*\*\*\*\*)

*ContactFlowId*: 問い合わせフローID

*InstanceId*: インスタンスID

*SourcePhoneNumber*: 発信元の電話番号

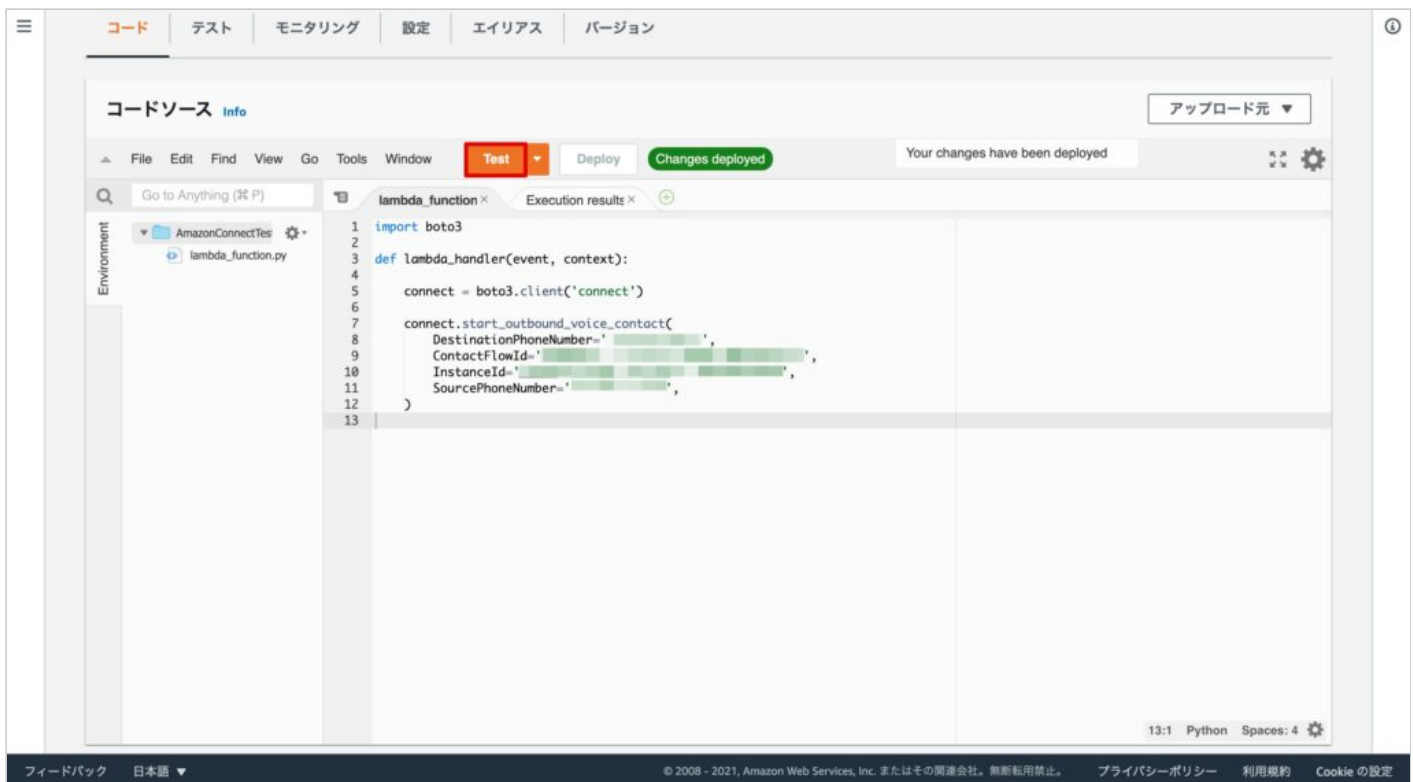
※国番号をつけて記述(日本の050の番号の場合、+8150\*\*\*\*\*)



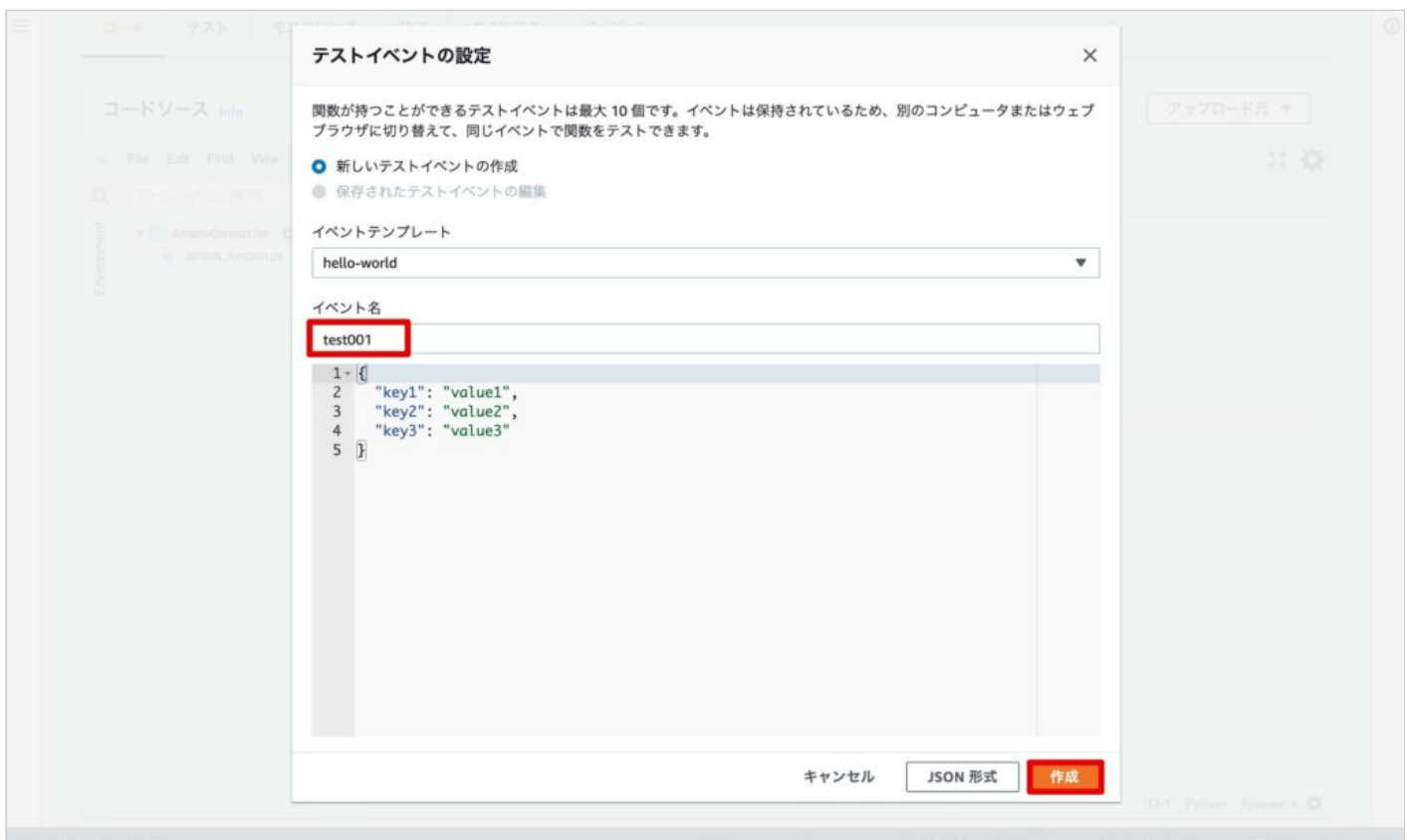
## Lambda関数のテスト

### テスト設定

「Test」をクリックします。



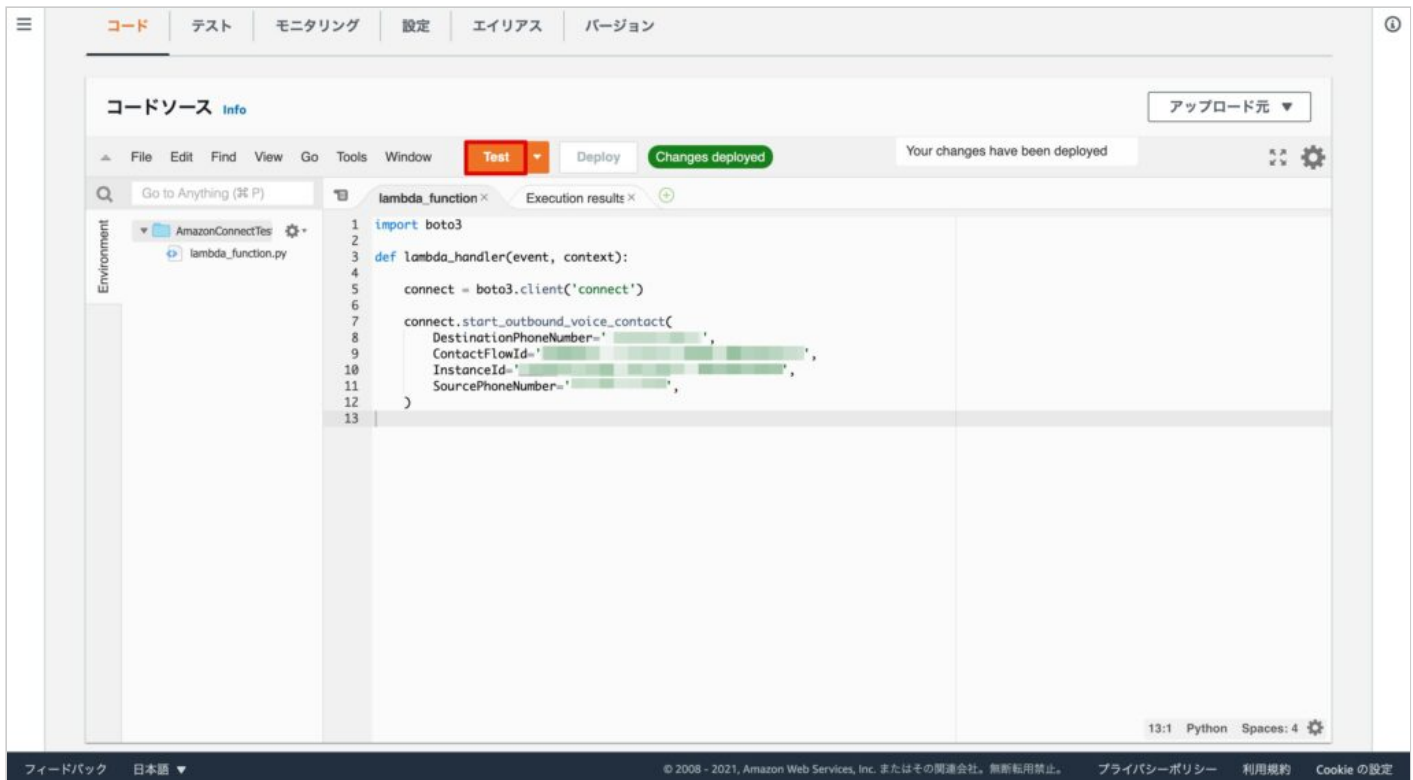
テストイベントの設定画面が表示されます。今回は引数は不要となりますが、“hello-world”のテンプレートのままで任意のイベント名を入力し、「作成」をクリックします。



## テスト実施

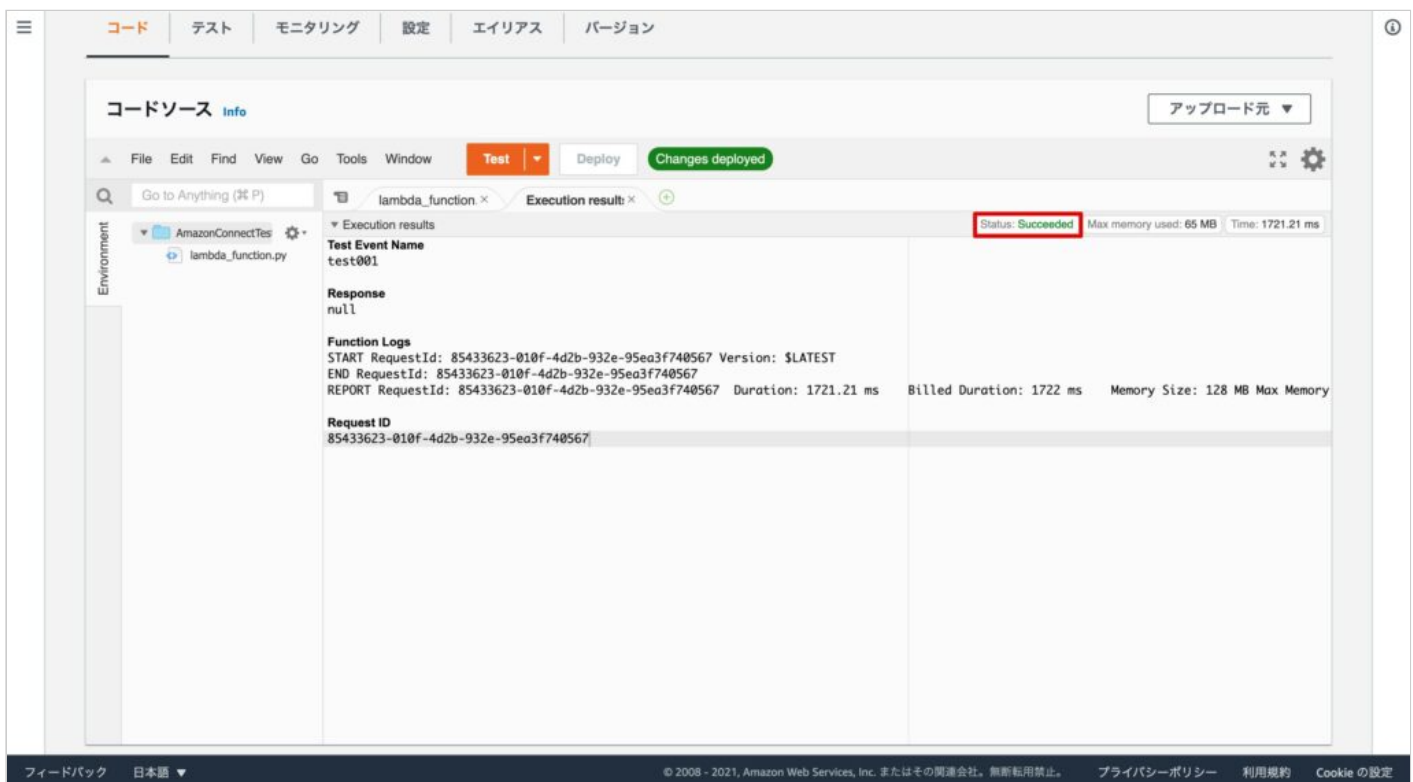
再度、「Test」をクリックします。





## テスト成功

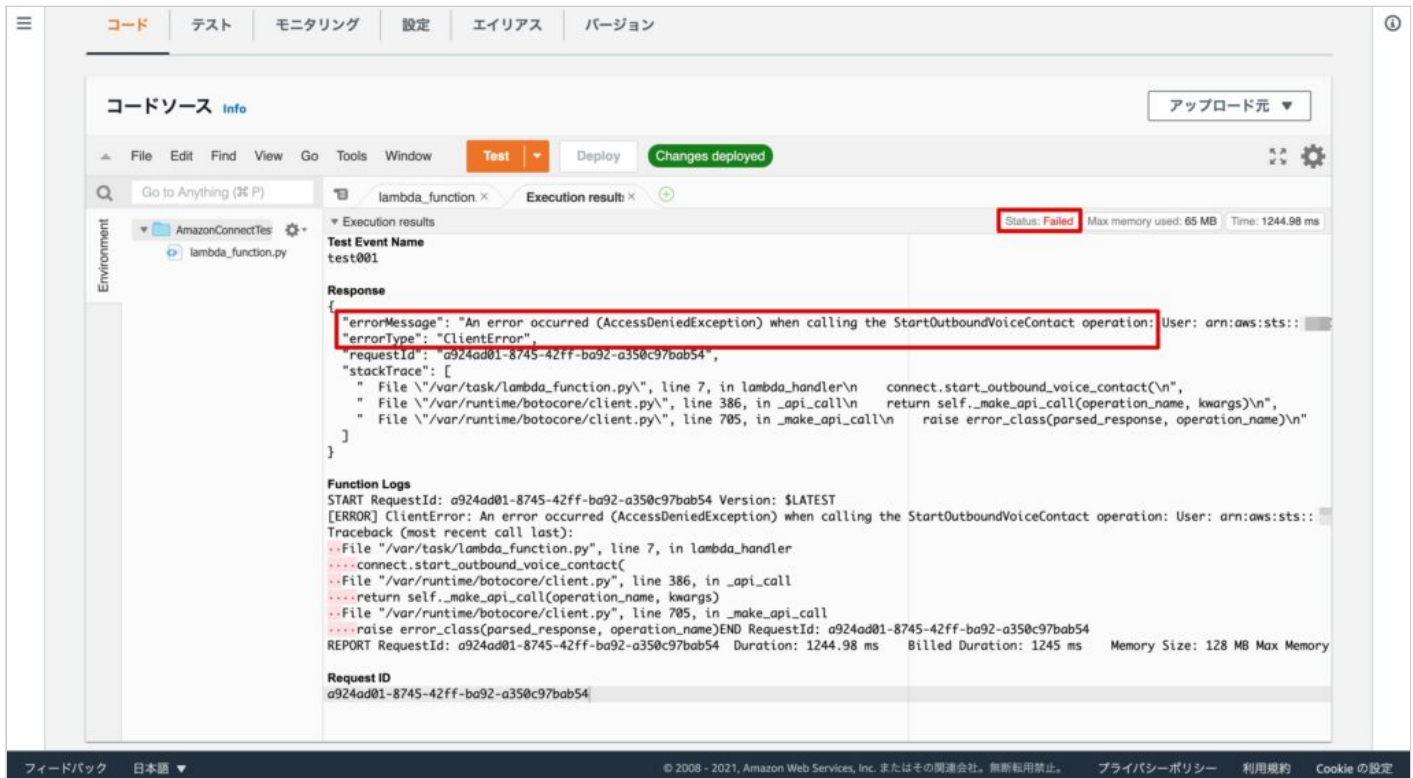
Statusが“Succeeded”となれば、テスト成功です。発信先の番号に着信があり、問い合わせフローで設定した音声再生されることを確認します。



## テスト失敗ケース1

Statusが“Failed”となり以下のメッセージが表示される場合は、アクセス権の設定を確認してください。

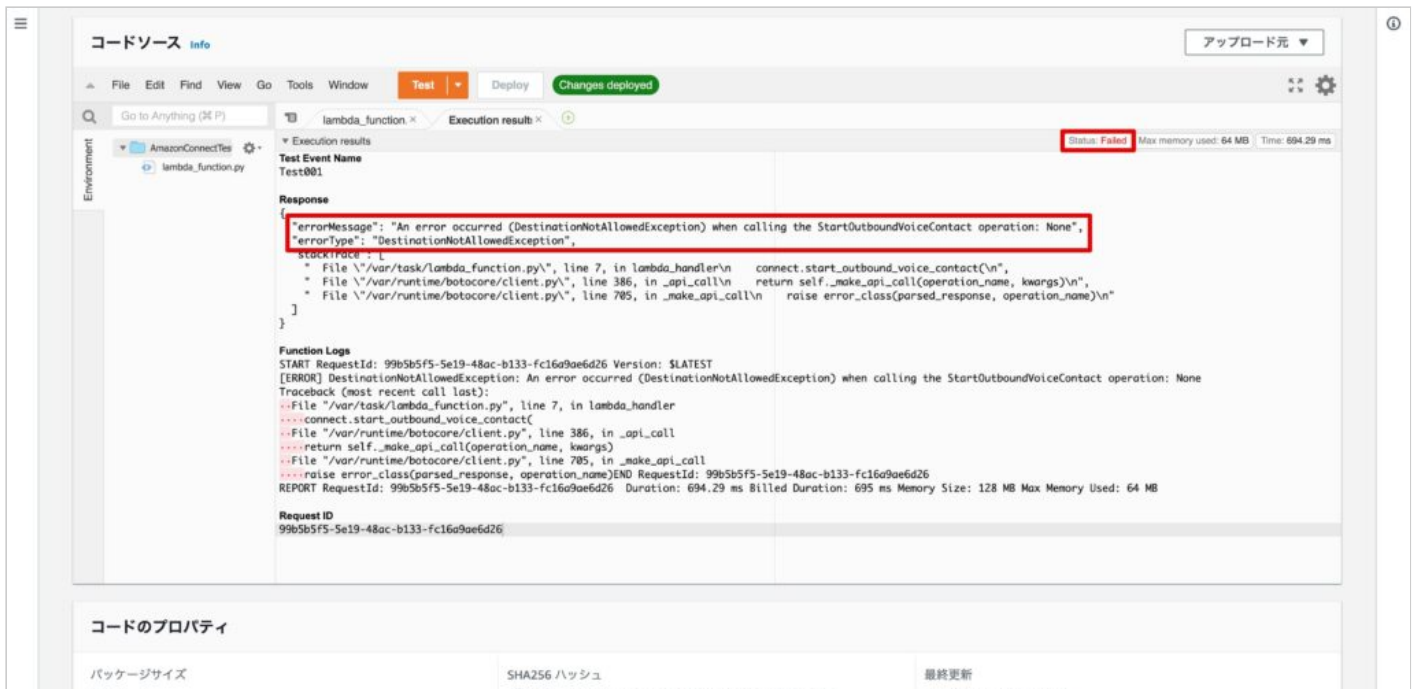
```
"errorMessage": "An error occurred (AccessDeniedException) when calling the  
StartOutboundVoiceContact operation"  
"errorType": "ClientError"
```



## テスト失敗ケース2

Statusが"Failed"となり以下のメッセージが表示される場合は、AmazonConnectのサービスクォータにより発信通話が制限されています。

```
"errorMessage": "An error occurred (DestinationNotAllowedException) when calling the  
StartOutboundVoiceContact operation"  
"errorType": "DestinationNotAllowedException"
```



## [Amazon Connect サービスクォータ](#)

### Amazon Connect サービスクォータ

PDF

特に明記しない限り、すべてのサービスクォータは調整/増加できます。

- インスタンスを作成し（存在している必要があります）、サービスクォータの増加を送信します。の使用[Amazon Connect サービスのクォータ増加フォーム](#)。この申請にアクセスするには、AWS アカウントにサインインしている必要があります。
- サービスクォータを増やすには、最大数週間かかることがあります。大規模なプロジェクトの一部としてクォータを増やす場合は、この時間をプランに追加してください。
- 同じフォームを使用して、現在のキャリアから Amazon Connect への米国の電話番号を移植するリクエストを送信します。電話番号の移植の詳細については、「[現在の電話番号を移植する](#)」を参照してください。

### デフォルトで許可されていないプレフィックス

次のプレフィックスを持つ日本の携帯電話番号は、デフォルトでは許可されません。

- +8170, 8180, 8190

このような日本の携帯電話番号に発信するには、サービスクォータ引き上げリクエストを送信する必要があります。

この場合、AWSサポートに問い合わせを行い、サービス制限を解除する必要があります。サービス制限解除の方法を下記を参照してください。

## [Amazon Connect からの発信通話を許可する方法](#)

以上で、AmazonConnectによる自動電話通知（4.Lambdaによる起動）の説明は完了です！

