

AmazonConnectによる自動電話通知（7.複数連絡先への電話通知〈構築⑨〉）

AWS



×

ZABBIX

×



AmazonConnectによる自動電話通知 （7.複数連絡先への電話通知〈構築⑨〉）

2021.11.12 2021.11.07

[【前回】 AmazonConnectによる自動電話通知（7.複数連絡先への電話通知〈構築⑧〉）](#)[【次回】 AmazonConnectによる自動電話通知（8.周回数の指定）](#)[【簡易版】 AmazonConnectによる自動電話通知（まとめ）](#)

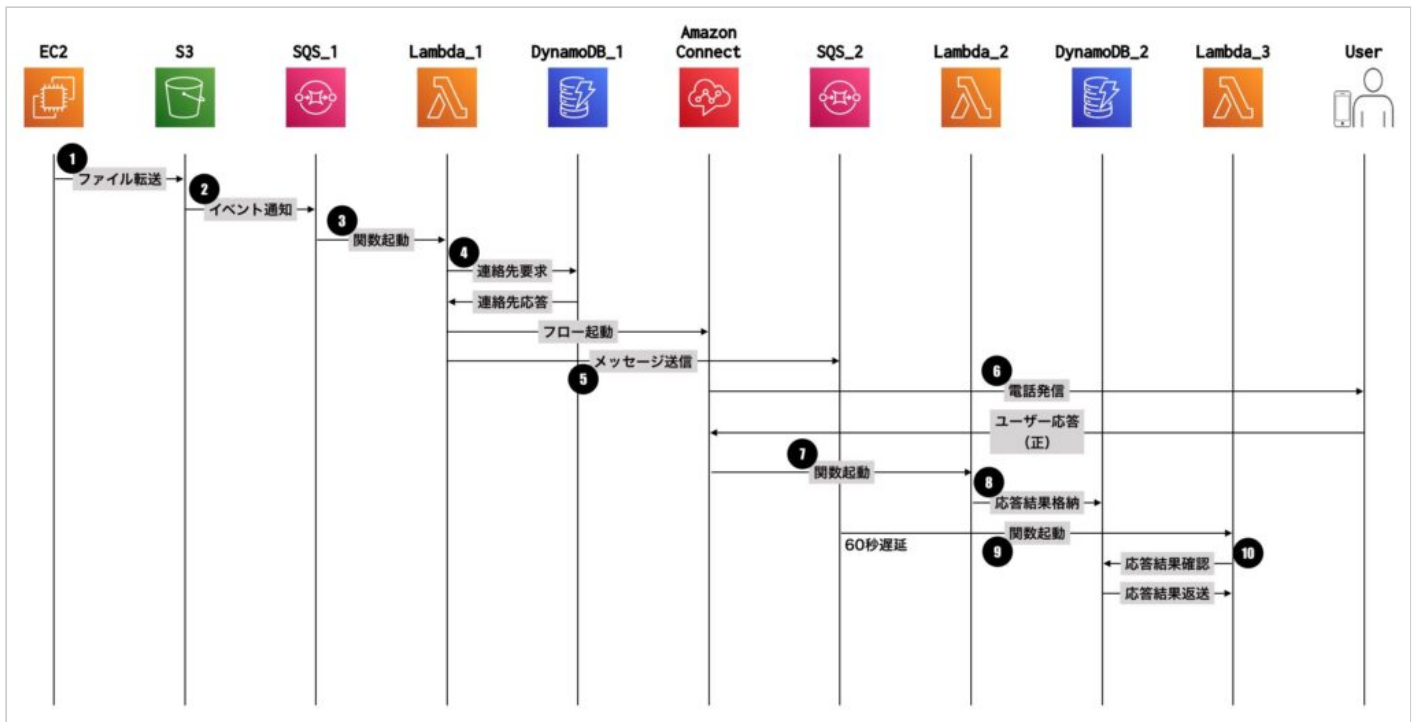
監視サーバーで障害を検知した際に、自動で電話通知できるようにしていきます。ネットワークエンジニアも利用することの多い監視サーバー(Zabbix)で障害検知し、AWS上のAmazonConnectを利用し自動電話を発信します。

今回は下記の条件を満たせるようにAWSの各サービスを利用して自動電話通知の仕組みを導入します。

- 複数の通知先を登録した連絡先リストを持たせる。
- 連絡先リストに優先度(通知順)を設定する。
- 優先度が高い人に最初に電話する。
- 応答が無かった場合、次の優先度の人に順番に電話する。
- 連絡先リストの最後まで電話しても応答が無かった場合、最初に戻って継続する。

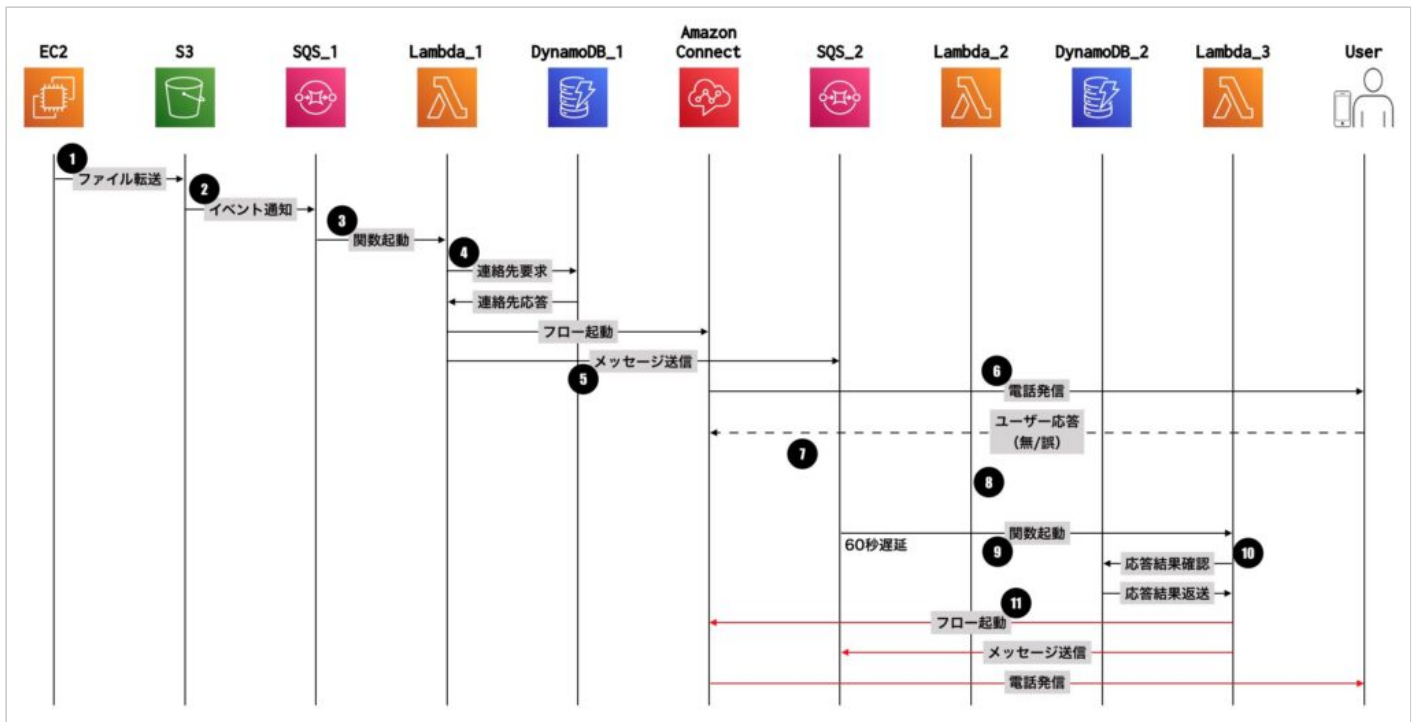
自動電話通知フロー

電話に応答した場合のフロー



1. EC2上の監視サーバーで障害を検知し、S3へトリガーファイルを格納
2. S3のイベント通知機能で、SQS_1にメッセージを送信
3. SQS_1をトリガーとして、Lambda_1を起動
4. Lambda_1がDynamoDB_1から連絡先を取得し、AmazonConnectを起動
5. Lambda_1がAmazonConnectを起動すると同時に、SQS_2へメッセージを送信
6. AmazonConnectがユーザーへ自動電話通知を実施
7. ユーザーが正常応答し、AmazonConnectがLambda_2を起動
8. Lambda_2が応答結果をDynamoDB_2に保存(応答OK)
9. 60秒後にSQS_2をトリガーとしてLambda_3を起動
10. Lambda_3がDynamoDB_2の応答結果を確認(正常応答しているため、何もせずに処理完了)

電話に応答しなかった場合のフロー



1. EC2上の監視サーバーで障害を検知し、S3へトリガーファイルを格納
2. S3のイベント通知機能で、SQS_1にメッセージを送信
3. SQS_1をトリガーとして、Lambda_1を起動
4. Lambda_1がDynamoDB_1から連絡先を取得し、AmazonConnectを起動
5. Lambda_1がAmazonConnectを起動すると同時に、SQS_2へメッセージを送信
6. AmazonConnectがユーザーへ自動電話通知を実施
7. ユーザーが正常応答せず、AmazonConnectがLambda_2を起動
8. Lambda_2が応答結果をDynamoDB_2に保存(応答NG)
9. 60秒後にSQS_2をトリガーとしてLambda_3を起動
10. Lambda_3がDynamoDB_2の応答結果を確認
11. 正常応答していないため、再度AmazonConnectを起動(以降、5から繰り返す)

テスト

構築①～⑧でAWSサービスの準備は完了となるため、テストを行います。

EC2から下記のコマンドを実施し、S3へファイルをアップロードします。※黄色アンダーライン箇所は、作成したS3バケット名を指定してください。

```
touch /tmp/test.txt
aws s3 cp /tmp/test.txt s3://amazonconnect-alert-notification-bucket
```

```
[ec2-user@ip-10-0-0-100 ~]$ touch /tmp/test.txt
[ec2-user@ip-10-0-0-100 ~]$
[ec2-user@ip-10-0-0-100 ~]$ aws s3 cp /tmp/test.txt s3://amazonconnect-alert-
notification-bucket
upload: ../../tmp/test.txt to s3://amazonconnect-alert-notification-
bucket/test.txt
[ec2-user@ip-10-0-0-100 ~]$
```

以下の観点で確認を行います。

- 初回の電話通知はプライオリティ"1"の通知先に連絡されること
- 正常応答しなかった場合、次のプライオリティの通知先に連絡されること
- 連絡先リストの最後まで連絡しても正常応答しなかった場合、プライオリティ"1"の通知先に戻って連絡されること

1回目、2回目の通知には正常応答せず、3回目に正常応答した場合のCloudWatchのログです。

CloudWatch > Log groups > /aws/lambda/amazonconnect-status-save > [breadcrumb]

ログイベント
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](#)

☐ View as text アクション

通知 [X] Clear 1m 30m 1h 12h Custom [icon]

タイムスタンプ	メッセージ
T05:47:44.960+09:00	+81 [redacted] への通知に失敗しました。 ← プライオリティ1の連絡先
T05:48:56.165+09:00	+81 [redacted] への通知に失敗しました。 ← プライオリティ2の連絡先
T05:49:46.267+09:00	+81 [redacted] への通知に成功しました。 ← プライオリティ1の連絡先

【Zabbix】スクリプトの修正

メディアタイプから実行するスクリプトを修正します。Zabbixの通知設定については下記を参照してください。

[AmazonConnectによる自動電話通知（5.EC2/ZABBIXとの連携）](#)

ZabbixをインストールしたEC2にSSHログインし、下記を実行します。

```
cd /usr/lib/zabbix/alertscripts/
sudo vi amazonconnect_notification.sh
（下記の通り修正）
```

```
-----
#!/bin/sh
```

```
echo "`date` Amazon Connect Notification" >> /tmp/zabbix-amazonconnect.log
```

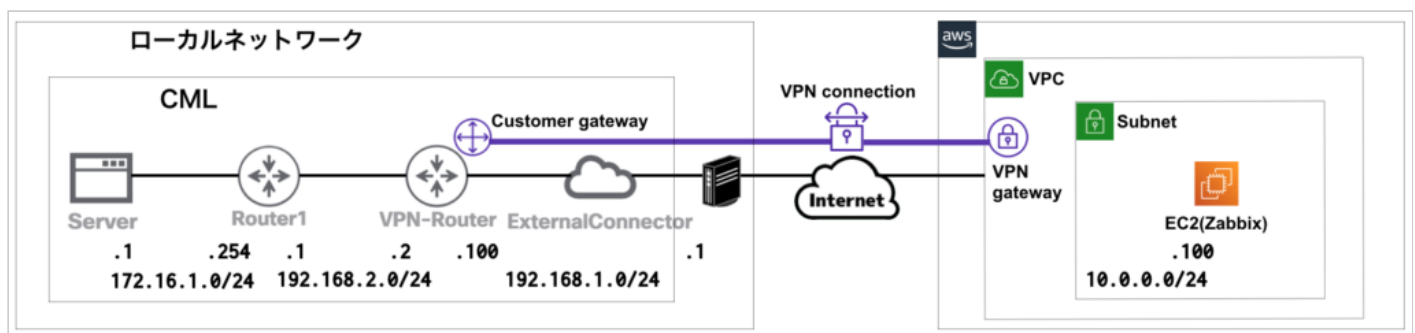
```
aws s3 cp /tmp/test.txt s3://amazonconnect-alert-notification-bucket
```

```
-----
```

障害検知の確認

ICMPノード監視の登録方法は下記を参照してください。

[AWS上での監視サーバー\(Zabbix\)構築【6.監視登録\(ICMPノード監視\)】](#)



Router1のGi0/1をshutdownし、Serverへの疎通が通らないようにします。

```
[Router1]
int Gi0/1
shut
```

Zabbix上で、Serverの障害が検知されることを確認します。

アクションにマウスカーソルを当てると、アクションが実行されていることが分かります。



フロー通りに、自動電話通知が行われることを確認します。

以上で、AmazonConnectによる自動電話通知（7.複数連絡先への電話通知〈構築⑨〉）の説明は完了です！