

# AWS Cloudでリソース配置スクリプトを追加する方法

## 内容

[概要](#)

[前提条件](#)

[要件](#)

[使用するコンポーネント](#)

[AWSクラウドでリソース配置スクリプトを追加する方法](#)

## 概要

このドキュメントでは、CloudCenter 4.8.2用のAmazon Web Services (AWS)クラウドにユーザーリソース配置スクリプトを追加する方法について説明します。

## 前提条件

## 要件

次の項目に関する知識があることが推奨されます。

- Linux
- AWS EC2クラウド
- シェルスクリプト

## 使用するコンポーネント

このドキュメントの情報は、次のソフトウェアのバージョンに基づいています。

- CloudCenter バージョン 4.8.2
- CCO(CiscoCloud Orchestrator)
- CCM(CiscoCloud Manager)
- Redhat7またはCentos7
- コールアウトスクリプトを格納するリポジトリ

このドキュメントの情報は、特定のラボ環境にあるデバイスに基づいて作成されました。このドキュメントで使用するすべてのデバイスは、初期（デフォルト）設定の状態から起動しています。本稼働中のネットワークでは、各コマンドによって起こる可能性がある影響を十分確認してください。

## AWSクラウドでリソース配置スクリプトを追加する方法

CloudCenterがユーザ設定に応じてリソースを配置するためにサポートするクラウドは異なります。このドキュメントでは、AWSクラウドでのリソース配置コールアウトスクリプトの設定について説明します。

リソースを配置するためのコールアウトスクリプトを配置するには、次の手順に従ってください

ステップ1: リポジトリ/Linux VMにログインし、リソースを配置するためのbashスクリプトを作成します。任意のLinux VMでスクリプトを作成する場合は、アプリケーションを展開するためにアクセスできるように、リポジトリ内でこれらのファイルをコピーしてください。

ステップ2: **vi callout.sh**を実行し、スクリプトに内容を入力します。

```
#!/bin/bash

. /utils.sh

print_log "$eNV_imageName"

print_log "$Cloud_Setting_CloudFamily"

print_log "$eNV_parentJobName"

content="{\"vpcId\": \"vpc-31e88948\",
\"subnetId\": \"subnet-44f8bb0c\",
\"securityGroupList\": \"sg-0f05b97b\",
\"vmTagsList\": \"Name:RP_001, PayProfile:Dev, User:AdminUser\",
\"assignPublicIp\": \"true\",
\"nodeInfo\": \"VpcID:vpc-31ee2948, subnetId: subnet-44ftb40c, securityGroupList:sg-0f04b97b \"}"

print_ext_service_result "$content"
```

**注:** VPC ID、サブネット、セキュリティグループ、vmTagList、node-infoを自分の情報に置き換えます。

すべての情報は**AWS EC2**インスタンスから取得できます

ステップ3: bashスクリプトをリポジトリに保存し、権限を755に変更します。

ステップ4. **#chmod 755 callout.sh**

ステップ5: スクリプトを作成したら、CloudCenter Managerからリソース配置スクリプトを有効にする必要があります。

a. CloudCenter Manager GUI > Environment > New Environmentにログインします。

General Settings

\* NAME

Resource\_pl

resource Placement

1

SERVICENOW EXTENSION ⓘ

None

APPROVAL REQUIRED TO DEPLOY TO THIS ENVIRONMENT ⓘ

NO

Cloud Selection

\* CLOUD REGION / 1 SELECTED

\* CLOUD ACCOUNT

AWS RTP  
US East (Virginia)

AWS RTP

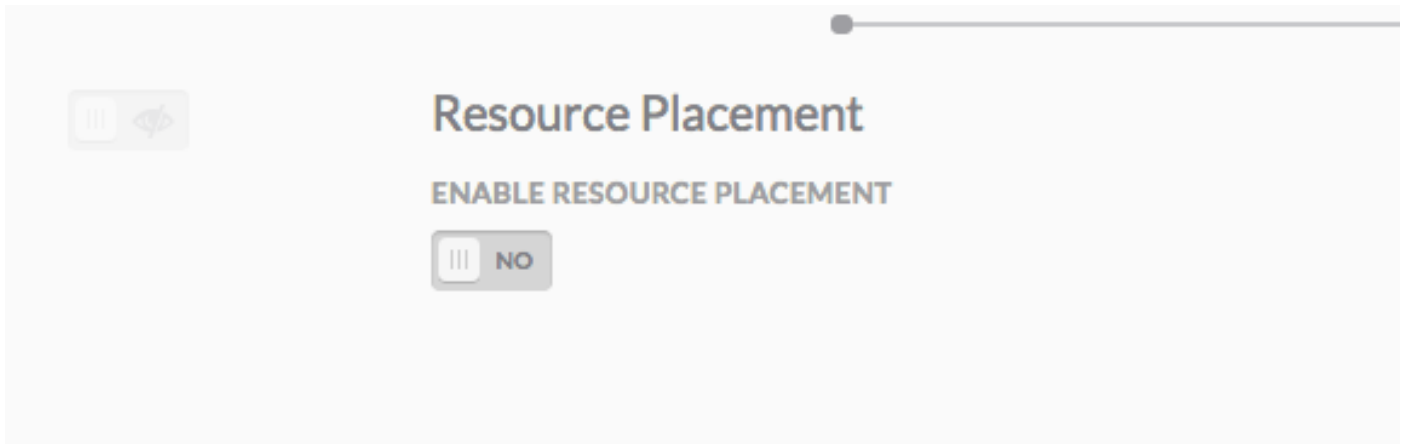
Simplified Networks

USE SIMPLIFIED NETWORKS ⓘ

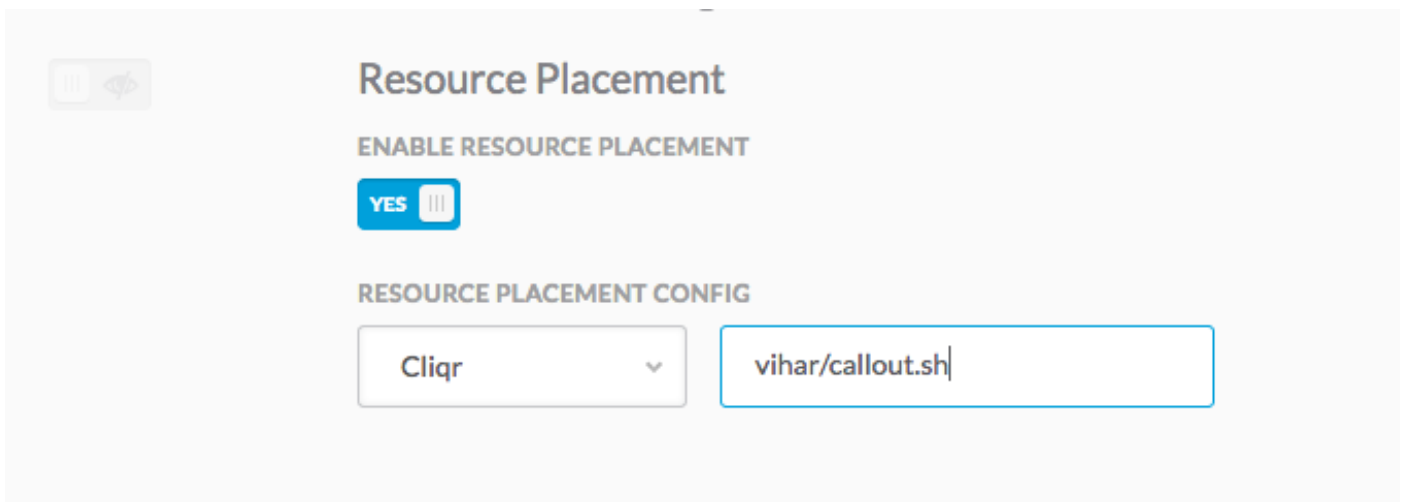
NO

b. [既定のクラウド設定を定義]タブをクリックします。

c. 図に示すように、導入環境で使用するインスタンス・タイプを選択し、リソース配置を有効にします。



d.[リソース配置を有効にする]をクリックすると、図に示すように、リソース配置設定ファイルを設定するオプションが表示されます。



注：リポジトリにアップロードしたファイルcallout.shの場所を指定して、[完了]タブを2回クリックして、[配置環境]ページを終了します。

ステップ6：新しく作成した導入環境を使用して新しいインスタンスを導入します。作成したリソース配置スクリプトを使用して、VMが正常に配置されます。

ヒント：CCOからgateway.logファイルを確認して、スクリプトが正常に実行されているかどうかを確認できます。

これは、ログファイルに表示される出力です。

```
2018-01-09 15:16:14,917 INFO service.LifecycleClusterStartAction [RxCachedThreadScheduler-6] - DeploymentJobID=37 requestNodeCount = 1 , minNodeCount=1 , createdCount = 1
```

```
2018-01-09 15:16:16,121 INFO strategy.DockerBaseCallout [threadPoolExecutor-21] - Output from Callout
```

```
CLIQR_EXTERNAL_SERVICE_LOG_MSG_START^M
```

```
Executing service resourcePlacement action with command:  
"/opt/remoteFiles/cliqr_local_file/callout.sh" from directory: ^M
```

```
CLIQR_EXTERNAL_SERVICE_LOG_MSG_END^M
```

```
CLOUD_CENTER_SCRIPT_OUTPUT^M
CLIQR_EXTERNAL_SERVICE_LOG_MSG_START^M
Executing script/command: /opt/remoteFiles/cliqr_local_file/callout.sh.^M
CLIQR_EXTERNAL_SERVICE_LOG_MSG_END^M
CLIQR_EXTERNAL_SERVICE_LOG_MSG_START^M
Ubuntu 14.04^M
CLIQR_EXTERNAL_SERVICE_LOG_MSG_END^M
CLIQR_EXTERNAL_SERVICE_LOG_MSG_START^M
^M
CLIQR_EXTERNAL_SERVICE_LOG_MSG_END^M
CLIQR_EXTERNAL_SERVICE_LOG_MSG_START^M
docker_cluster^M
CLIQR_EXTERNAL_SERVICE_LOG_MSG_END^M
CLIQR_EXTERNAL_SERVICE_RESULT_START^M
{"vpcId":" vpc-31e88948",^M
"subnetId":" subnet-44f8bb0c ",^M
"securityGroupList":" sg-0f05b97b ",^M
"vmTagsList":"Name:RP_001,PayProfile:Dev,User:AdminUser",^M
"assignPublicIp":"true",^M
"nodeInfo":"VpcID: vpc-31e88948, subnetId: subnet-44f8bb0c,securityGroupList: sg-0f05b97b "}^M
CLIQR_EXTERNAL_SERVICE_RESULT_END^M
```