

# AWSサイト間VPNの構築（7.AWS CLI によるVPN接続の作成）

AWS



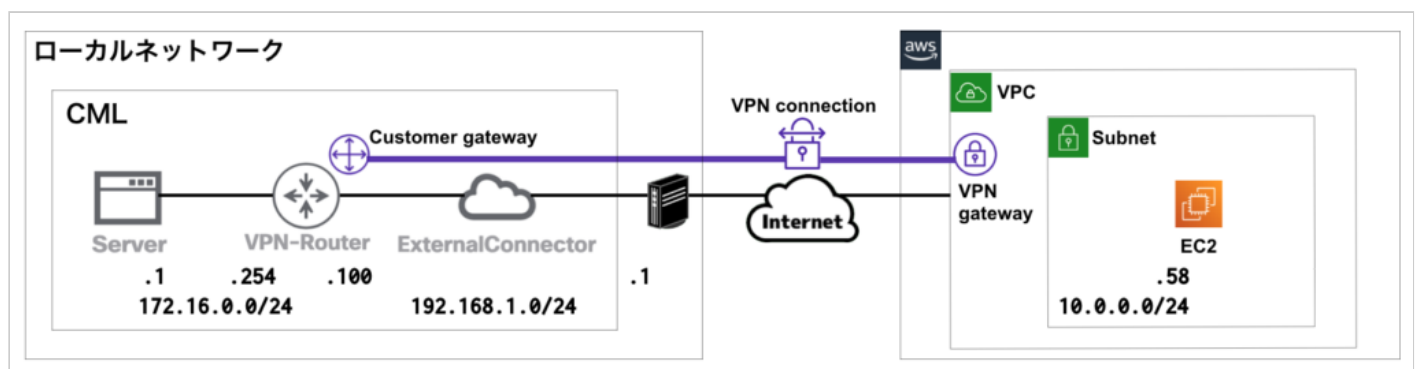
## AWSサイト間VPNの構築 （7.AWS CLI によるVPN接続の作成）

2021.09.20

[【前回】AWSサイト間VPNの構築（6.IKEv2の設定）](#)

## ネットワーク構成

下記の構成でAWSのサイト間VPNを構築しました。



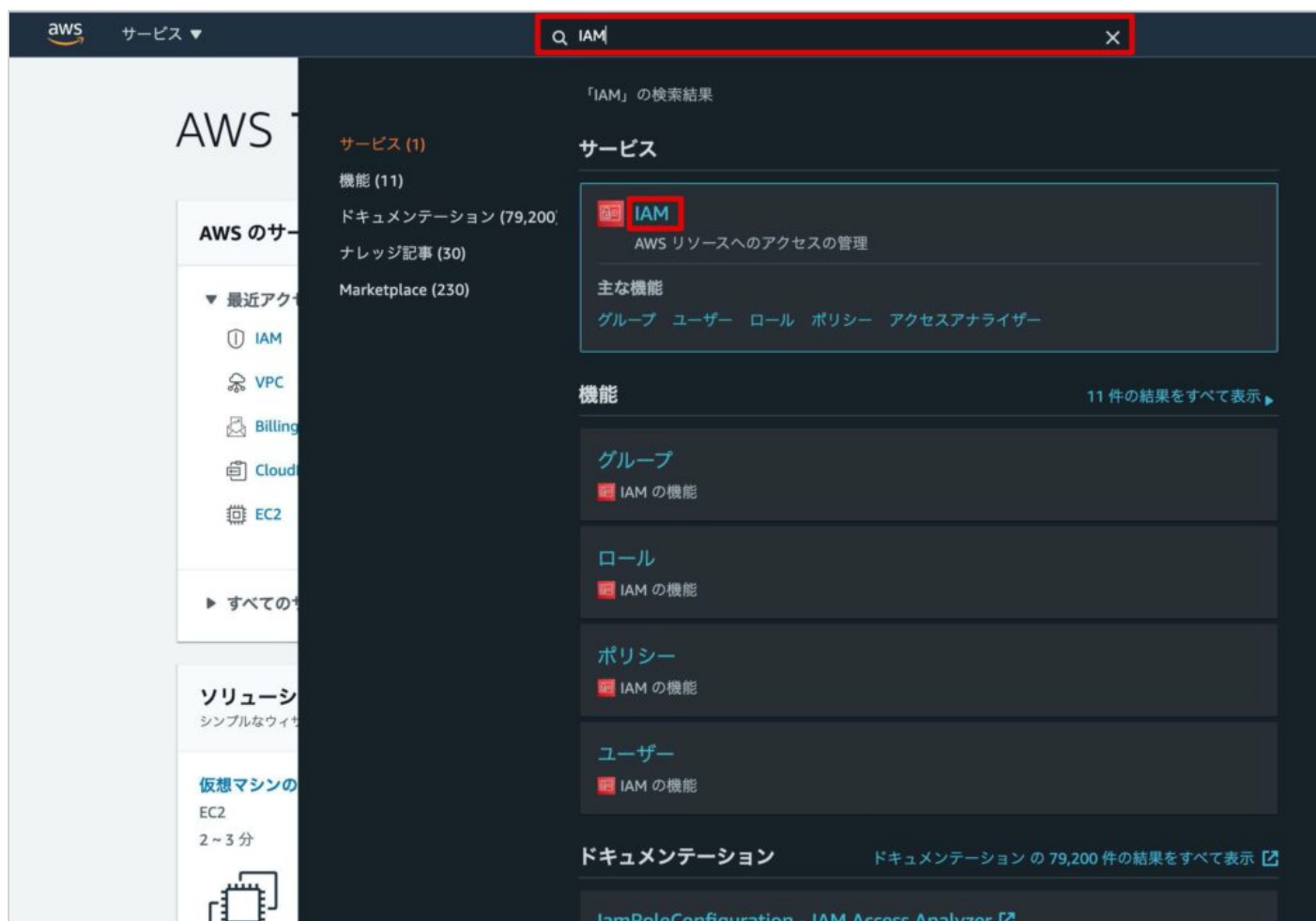
## AWS CLI によるVPN接続の作成と削除

VPN接続を作成すると、利用時間により課金されます。費用を抑えるためには、検証時のみVPN接続を作成する必要があります。都度、AWSコンソール画面からの作成は手間がかかるため、AWS CLI によるコマンドラインからの作成と削除方法を説明します。

## ユーザーの作成

AWS CLI を利用するためのAWSユーザーを作成します。

検索窓に“IAM”と入力し、表示される「IAM」をクリックします。



左のメニューで「ユーザー」を選択し、「ユーザーを追加」をクリックします。



The screenshot shows the AWS IAM console interface. On the left, the 'Identity and Access Management (IAM)' menu is open, and 'Users' is selected under the 'Access Management' section. The main content area displays the 'Users' page, which is currently empty (0 users). A red box highlights the 'Add User' button in the top right corner. Below the button is a search bar and a table with columns for user management. The table is empty, and a message at the bottom states 'No resources to display'.

ユーザー名に任意の名前を入力します。（ここでは、“cliadmin”としています。）

認証情報タイプに関しては、CLI専用のユーザーとするため、「アクセスキー – プログラムによるアクセス」にチェックを入れます。

## ユーザーを追加

12345

### ユーザー詳細の設定

同じアクセスの種類とアクセス権限を使用して複数のユーザーを一度に追加できます。 [詳細はこちら](#)

ユーザー名\*

cliadmin

別のユーザーの追加

### AWS アクセスの種類を選択

これらのユーザーが主に AWS にアクセスする方法を選択します。プログラムによるアクセスのみを選択しても、ユーザーは引き受けたロールを使用してコンソールにアクセスすることはできます。アクセスキーと自動生成されたパスワードは、最後のステップで提供されます。 [詳細はこちら](#)

AWS 認証情報タイプの選択\*

☒ **アクセスキー - プログラムによるアクセス**  
AWS API、CLI、SDK などの開発ツールの **アクセスキー ID** と **シークレットアクセスキー** を有効にします。

☐ **パスワード - AWS マネジメントコンソールへのアクセス**  
ユーザーに AWS マネジメントコンソールへのサインインを許可するための **パスワード** を有効にします。

\* 必須

キャンセル

次のステップ: アクセス権限

© 2008 - 2021, Amazon Web Services, Inc. またはその関連会社。無断転用禁止。

アクセス権限に関しては、「既存のポリシーを直接アタッチ」を選択し、必要なアクセス権限を追加します。ここでは、管理者として全権限を付与するため、「AdministratorAccess」を追加しています。

## ユーザーを追加

1


2


3


4


5

### ▼ アクセス許可の設定

 ユーザーをグループに追加













 アクセス権限を既存のユーザーからコピー

 既存のポリシーを直接アタッチ

ポリシーの作成 

ポリシーのフィルタ ▼

682 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"/>	 AdministratorAccess	ジョブ機能	なし
<input type="checkbox"/>	 AdministratorAccess-Amplify	AWS による管理	なし
<input type="checkbox"/>	 AdministratorAccess-AWSElasticBeanstalk	AWS による管理	なし
<input type="checkbox"/>	 AlexaForBusinessDeviceSetup	AWS による管理	なし
<input type="checkbox"/>	 AlexaForBusinessFullAccess	AWS による管理	なし
<input type="checkbox"/>	 AlexaForBusinessGatewayExecution	AWS による管理	なし
<input type="checkbox"/>	 AlexaForBusinessLifesizeDelegatedAccessPolicy	AWS による管理	なし
<input type="checkbox"/>	 AlexaForBusinessPolyDelegatedAccessPolicy	AWS による管理	なし
<input type="checkbox"/>	 AlexaForBusinessReadOnlyAccess	AWS による管理	なし
<input type="checkbox"/>	 AmazonAPIGatewayAdministrator	AWS による管理	なし
<input type="checkbox"/>	 AmazonAPIGatewayInvokeFullAccess	AWS による管理	なし
<input type="checkbox"/>	 AmazonAPIGatewayPushToCloudWatchLogs	AWS による管理	なし

キャンセル

戻る

次のステップ: タグ

© 2008 - 2021, Amazon Web Services, Inc. またはその関連会社。無断転用禁止。

必要に応じてタグを追加します。（必須ではありません。）

## ユーザーを追加

1

2

3

4

5

### タグの追加 (オプション)

IAM タグは、ユーザー に追加できるキーと値のペアです。タグには、E メールアドレスなどのユーザー情報を含めるか、役職などの説明文とすることができます。タグを使用して、このユーザー のアクセスを整理、追跡、制御できます。[詳細はこちら](#)

キー

値 (オプション)

削除

さらに 50 個のタグを追加できます。

キャンセル

戻る

次のステップ: 確認

© 2008 - 2021, Amazon Web Services, Inc. またはその関連会社。無断転用禁止。

設定内容を確認し、「ユーザーの作成」をクリックします。

2021/11/17 9:17

AWSサイト間VPNの構築（7.AWS CLI によるVPN接続の作成） | ネットワークエンジニアのガイドマップ

ユーザーを追加

1

2

3

4

5

確認

選択内容を確認します。ユーザーを作成した後で、自動生成パスワードとアクセスキーを確認してダウンロードできます。

ユーザー詳細

ユーザー名

cliadmin

AWS アクセスの種類

プログラムによるアクセス - アクセスキーを使用

アクセス権限の境界

アクセス権限の境界が設定されていません

アクセス権限の概要

次のポリシー例は、上記のユーザーにアタッチされます。

タイプ	名前
管理ポリシー	<a href="#">AdministratorAccess</a>

タグ

追加されたタグはありません。

キャンセル

戻る

ユーザーの作成

© 2008 - 2021, Amazon Web Services, Inc. またはその関連会社。無断転用禁止。

「シークレットアクセスキー」を表示します。

## ユーザーを追加

1 2 3 4 5

## 成功

以下に示すユーザーを正常に作成しました。ユーザーのセキュリティ認証情報を確認してダウンロードできます。AWS マネジメントコンソールへのサインイン手順を E メールでユーザーに送信することもできます。今回が、これらの認証情報をダウンロードできる最後の機会です。ただし、新しい認証情報はいつでも作成できます。

AWS マネジメントコンソールへのアクセス権を持つユーザーは「[https://\[redacted\].signin.aws.amazon.com/console](https://[redacted].signin.aws.amazon.com/console)」でサインインできます

[.csv のダウンロード](#)

	ユーザー	アクセスキー ID	シークレットアクセスキー
▶	✓ cliadmin	[redacted]	***** <a href="#">表示</a>

閉じる

© 2008 - 2021, Amazon Web Services, Inc. またはその関連会社。無断転用禁止。

“アクセスキーID”と“シークレットアクセスキー”をコピーするか、CSVをダウンロードして保存します。



## ユーザーを追加

1 2 3 4 5

## 成功

以下に示すユーザーを正常に作成しました。ユーザーのセキュリティ認証情報を確認してダウンロードできます。AWS マネジメントコンソールへのサインイン手順を E メールでユーザーに送信することもできます。今回が、これらの認証情報をダウンロードできる最後の機会です。ただし、新しい認証情報はいつでも作成できます。

AWS マネジメントコンソールへのアクセス権を持つユーザーは「[https://\[redacted\].signin.aws.amazon.com/console](https://[redacted].signin.aws.amazon.com/console)」でサインインできます

## .csv のダウンロード

	ユーザー	アクセスキー ID	シークレットアクセスキー
▶	✓ cliadmin	[redacted]	[redacted] 非表示

閉じる

© 2008 - 2021, Amazon Web Services, Inc. またはその関連会社。無断転用禁止。

## AWS CLI の設定

EC2のマシンイメージで「Amazon Linux」を選択している場合、AWS CLIのインストールは不要なため、設定を行えばすぐに利用ができます。

初期設定を確認するために、EC2上で下記のコマンドを実施します。

[EC2]

```
aws configure list
```

何も設定されていないことがわかります。

```
[ec2-user@ip-10-0-0-100 ~]$ aws configure list
```

Name

Value

Type

Location

-----

-----

-----

-----

```

-
      profile                <not set>                None                None
access_key                  <not set>                  None                None
secret_key                  <not set>                  None                None
      region                <not set>                  None                None

```

作成したユーザーのプロファイルを設定します。

```
[EC2]
aws configure --profile cliadmin
(下記を設定)
AWS Access Key ID [None]:アクセスキーIDを入力
AWS Secret Access Key [None]: シークレットアクセスキーを入力
Default region name [None]: ap-northeast-1を入力 (EC2を配置しているリージョン)
Default output format [None]: 入力不要(デフォルトでjson形式)
```

設定したユーザーをデフォルトとして永続的に利用するために、環境変数を変更します。

```
[EC2]
sudo vi /etc/environment
(下記を追記)
AWS_DEFAULT_PROFILE=cliadmin
```

設定を反映するために、一度ログアウトして、再度ログインします。

再ログイン後、設定が反映されていることを確認します。

```
[EC2]
aws configure list
```

```
[ec2-user@ip-10-0-0-100 ~]$ aws configure list

      Name                                Value
Type    Location
-----  -----
      profile                            cliadmin
```

```
manual --profile
access_key ***** shared-credentials-file
secret_key ***** shared-credentials-file
region ap-northeast-1 config-file
~/.aws/config
```

## カスタマーゲートウェイIDの確認

カスタマーゲートウェイのIDを確認します。(CustomerGatewayIdの値をメモします。)

[EC2]

```
aws ec2 describe-customer-gateways
```

```
[ec2-user@ip-10-0-0-100 ~]$ aws ec2 describe-customer-gateways
{
  "CustomerGateways": [
    {
      "Tags": [
        {
          "Value": "aws-vpn-test-cgw",
          "Key": "Name"
        }
      ],
      "IpAddress": "xxx.xxx.xxx.xxx",
      "State": "available",
      "CustomerGatewayId": "cgw-xxxxxxxxxxxxxxxxxx",
      "BgpAsn": "65000",
      "Type": "ipsec.1"
    }
  ]
}
```

## 仮想プライベートゲートウェイIDの確認

仮想プライベートゲートウェイのIDを確認します。(VpnGatewayIdの値をメモします。)

[EC2]

```
aws ec2 describe-vpn-gateways
```

```
[ec2-user@ip-10-0-0-100 ~]$ aws ec2 describe-vpn-gateways
{
  "VpnGateways": [
    {
      "Tags": [
        {
          "Value": "aws-vpn-test-vgw",
          "Key": "Name"
        }
      ],
      "VpcAttachments": [
        {
          "State": "attached",
          "VpcId": "vpc-xxxxxxxxxxxxxxxxxx"
        }
      ],
      "AmazonSideAsn": 64512,
      "State": "available",
      "VpnGatewayId": "vgw-xxxxxxxxxxxxxxxxxx",
      "Type": "ipsec.1"
    }
  ]
}
```

## クラウドフォーメーションのテンプレート作成

テンプレート作成用のディレクトリを作成します。

```
[EC2]
mkdir CloudFormation
cd CloudFormation
```

テンプレートファイルを作成します。(赤アンダーライン部分は、適宜修正します。)

```
[EC2]
vi VPNConnection.yaml
```

（下記を記述）

```
AWSTemplateFormatVersion: '2010-09-09'
```

```
Resources:
```

```
# VPNConnection
```

```
VPNConnection1:
```

```
  Type: AWS::EC2::VPNConnection
```

```
  Properties:
```

```
    Type: ipsec.1
```

```
    StaticRoutesOnly: true
```

```
    VpnTunnelOptionsSpecifications:
```

```
      - TunnelInsideCidr: 169.254.173.148/30 ※トンネル1の内部 IPv4 CIDR
```

```
        PreSharedKey: ***** ※トンネル1の事前共有キー
```

```
    CustomerGatewayId: cgw-xxxxxxxxxxxxxxxxx ※カスタマーゲートウェイID
```

```
    VpnGatewayId: vgw-xxxxxxxxxxxxxxxxx ※仮想プライベートゲートウェイID
```

```
    Tags:
```

```
      - Key: Name
```

```
        Value: aws-vpn-test
```

```
VPNConnectionRoute1:
```

```
  Type: AWS::EC2::VPNConnectionRoute
```

```
  Properties:
```

```
    DestinationCidrBlock: 192.168.1.0/24
```

```
    VpnConnectionId:
```

```
      !Ref VPNConnection1
```

```
VPNConnectionRoute2:
```

```
  Type: AWS::EC2::VPNConnectionRoute
```

```
  Properties:
```

```
    DestinationCidrBlock: 172.16.0.0/24
```

```
    VpnConnectionId:
```

```
      !Ref VPNConnection1
```

## クラウドフォーメーションのスタック作成

クラウドフォーメーションのスタックを作成し、テンプレートに記述したサイト間VPN接続を作成します。

```
[EC2]
```

```
aws cloudformation create-stack --stack-name vpnteststack --template-body
```

```
file://VPNConnection.yaml
```

```
[ec2-user@ip-10-0-0-100 CloudFormation]$ aws cloudformation create-stack --
stack-name vpnteststack --template-body file://VPNConnection.yaml
{
  "StackId": "arn:aws:cloudformation:ap-northeast-
1:xxxxxxxxxxxx:stack/vpnteststack/xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx"
}
```

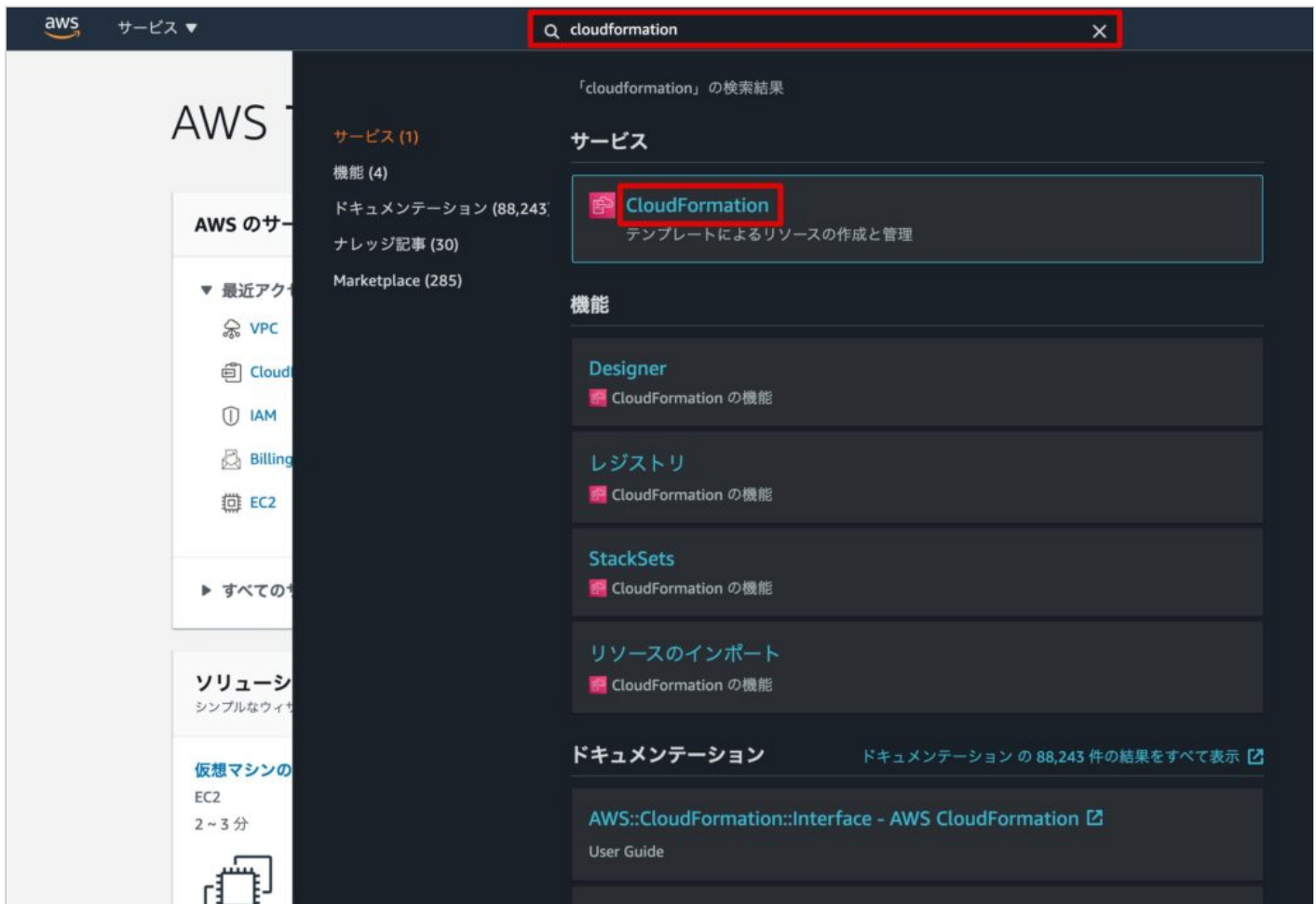
作成したスタックを確認します。

ステータスが"CREATE\_IN\_PROGRESS"となっていて、作成中であることがわかります。

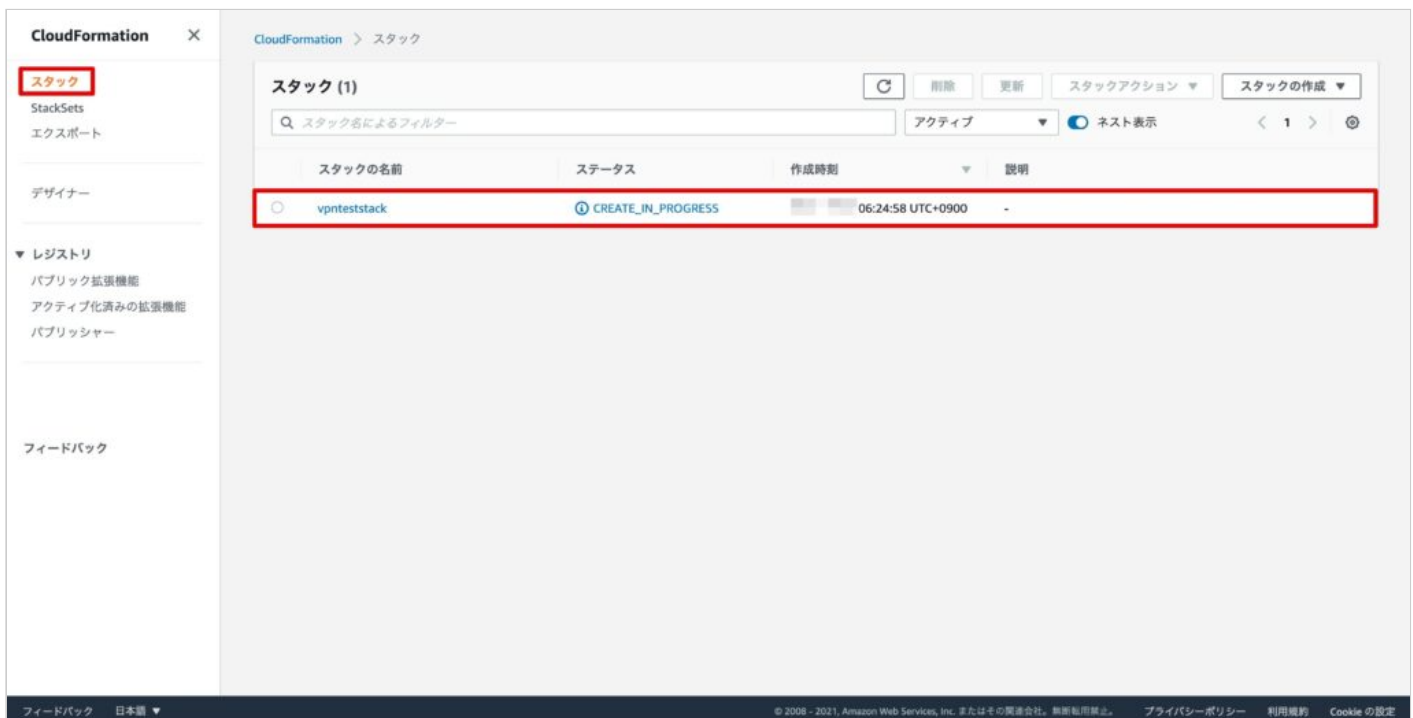
```
[EC2]
aws cloudformation describe-stacks
```

```
[ec2-user@ip-10-0-0-100 CloudFormation]$ aws cloudformation describe-stacks
{
  "Stacks": [
    {
      "StackId": "arn:aws:cloudformation:ap-northeast-
1:xxxxxxxxxxxx:stack/vpnteststack/xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx",
      "DriftInformation": {
        "StackDriftStatus": "NOT_CHECKED"
      },
      "Tags": [],
      "CreationTime": "xxxx-xx-xxT22:23:16.497Z",
      "StackName": "vpnteststack",
      "NotificationARNs": [],
      "StackStatus": "CREATE_IN_PROGRESS",
      "DisableRollback": false,
      "RollbackConfiguration": {}
    }
  ]
}
```

AWSの画面でも確認します。検索窓に"cloudformation"と入力し、表示される「CloudFormation」をクリックします。



左のメニューで「スタック」を選択し、作成したスタックが表示されていることを確認します。



対象のスタックをクリックすると詳細が表示されます。

The screenshot shows the AWS CloudFormation console for the 'vpnteststack' stack. The stack is in the 'CREATE\_IN\_PROGRESS' state, as indicated by the status icon and the 'ステータス' (Status) section. The '概要' (Summary) section provides details about the stack, including its ID, creation time, and current status.

項目	値
スタック ID	arn:aws:cloudformation:ap-northeast-1:stack/vpnteststack/
ステータス	CREATE_IN_PROGRESS
ルートスタック	-
作成時刻	06:24:58 UTC+0900
更新時刻	-
ドリフトステータス	NOT_CHECKED
削除保護	無効

「イベント」タブでスタック内のリソースの作成状況が確認できます。

The screenshot shows the 'vpnteststack' stack with the 'イベント' (Events) tab selected. The 'イベント (3)' section displays a list of events, including the creation of 'VPNConnection1' and the stack itself.

タイムスタンプ	論理 ID	ステータス	状況の理由
06:25:05 UTC+0900	VPNConnection1	CREATE_IN_PROGRESS	Resource creation Initiated
06:25:02 UTC+0900	VPNConnection1	CREATE_IN_PROGRESS	-
06:24:58 UTC+0900	vpnteststack	CREATE_IN_PROGRESS	User Initiated

数分後にもう一度確認すると、ステータスが"CREATE\_COMPLETE"となり、スタックの作成が完了していることがわかります。

[E02]

```
aws cloudformation describe-stacks
```



```
[ec2-user@ip-10-0-0-100 CloudFormation]$ aws cloudformation describe-stacks
{
  "Stacks": [
    {
      "StackId": "arn:aws:cloudformation:ap-northeast-1:xxxxxxxxxxxx:stack/vpnteststack/xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx",
      "DriftInformation": {
        "StackDriftStatus": "NOT_CHECKED"
      },
      "Tags": [],
      "CreationTime": "xxxx-xx-xxT22:23:16.497Z",
      "StackName": "vpnteststack",
      "NotificationARNs": [],
      "StackStatus": "CREATE_COMPLETE",
      "DisableRollback": false,
      "RollbackConfiguration": {}
    }
  ]
}
```

The screenshot shows the AWS CloudFormation console interface. On the left, the 'CloudFormation' sidebar is visible with options like 'スタック', 'ドリフト', 'StackSets', 'エクスポート', 'デザイナー', 'レジストリ', and 'フィードバック'. The main area displays the 'vpnteststack' stack details. The 'イベント' (Events) tab is selected, showing a list of events. The first event, 'vpnteststack' at 06:30:44 UTC+0900, is highlighted with a red box, indicating a 'CREATE\_COMPLETE' status. Other events include 'VPNConnectionRoute 2' and 'VPNConnectionRoute 1' in 'CREATE\_COMPLETE' status, and 'VPNConnectionRoute 2' and 'VPNConnectionRoute 1' in 'CREATE\_IN\_PROGRESS' status.

タイムスタンプ	論理 ID	ステータス	状況の理由
06:30:44 UTC+0900	vpnteststack	CREATE_COMPLETE	-
06:30:43 UTC+0900	VPNConnectionRoute 2	CREATE_COMPLETE	-
06:30:43 UTC+0900	VPNConnectionRoute 1	CREATE_COMPLETE	-
06:30:28 UTC+0900	VPNConnectionRoute 2	CREATE_IN_PROGRESS	Resource creation Initiated
06:30:27 UTC+0900	VPNConnectionRoute 1	CREATE_IN_PROGRESS	Resource creation Initiated
06:30:27 UTC+0900	VPNConnectionRoute 2	CREATE_IN_PROGRESS	-
06:30:27 UTC+0900	VPNConnectionRoute 1	CREATE_IN_PROGRESS	-
06:30:25 UTC+0900	VPNConnection1	CREATE_COMPLETE	-
06:25:05 UTC+0900	VPNConnection1	CREATE_IN_PROGRESS	Resource creation Initiated

## サイト間VPN接続の確認

クラウドフォーメーションにより作成されたサイト間VPN接続を確認します。

[EC2]

aws ec2 describe-vpn-connections

[ec2-user@ip-10-0-0-100 CloudFormation]\$ aws ec2 describe-vpn-connections

```

{
  "VpnConnections": [
    {
      "Category": "VPN",
      "VpnConnectionId": "vpn-xxxxxxxxxxxxxxxx",
      "Tags": [
        {
          "Value": "vpnteststack",
          "Key": "aws:cloudformation:stack-name"
        },
        {
          "Value": "VPNConnection1",
          "Key": "aws:cloudformation:logical-id"
        },
        {
          "Value": "arn:aws:cloudformation:ap-northeast-1:xxxxxxxxxxxx:stack/vpnteststack/xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx",
          "Key": "aws:cloudformation:stack-id"
        },
        {
          "Value": "aws-vpn-test",
          "Key": "Name"
        }
      ],
      ~ . ~ . ~ . ~ . ~ 中略 ~ . ~ . ~ . ~ . ~
    },
    {
      "DestinationCidrBlock": "172.16.0.0/24",
      "State": "available"
    },
    {
      "DestinationCidrBlock": "192.168.1.0/24",
      "State": "available"
    }
  ],

```

```

"State": "available",
"VpnGatewayId": "vgw-xxxxxxxxxxxxxxxx",
"CustomerGatewayId": "cgw-xxxxxxxxxxxxxxxx",
"Type": "ipsec.1",
"Options": {
  "RemoteIpv4NetworkCidr": "0.0.0.0/0",
  "EnableAcceleration": false,
  "TunnelInsideIpVersion": "ipv4",
  "TunnelOptions": [
    {
      "TunnelInsideCidr": "169.254.173.148/30",
      "PreSharedKey": "xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx",
      "OutsideIpAddress": "xxx.xxx.xxx.xxx"
    },
    {
      "TunnelInsideCidr": "169.254.209.216/30",
      "PreSharedKey": "xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx",
      "OutsideIpAddress": "xxx.xxx.xxx.xxx"
    }
  ],
  "StaticRoutesOnly": true,
  "LocalIpv4NetworkCidr": "0.0.0.0/0"
},
]
}

```

AWSの画面でも確認します。

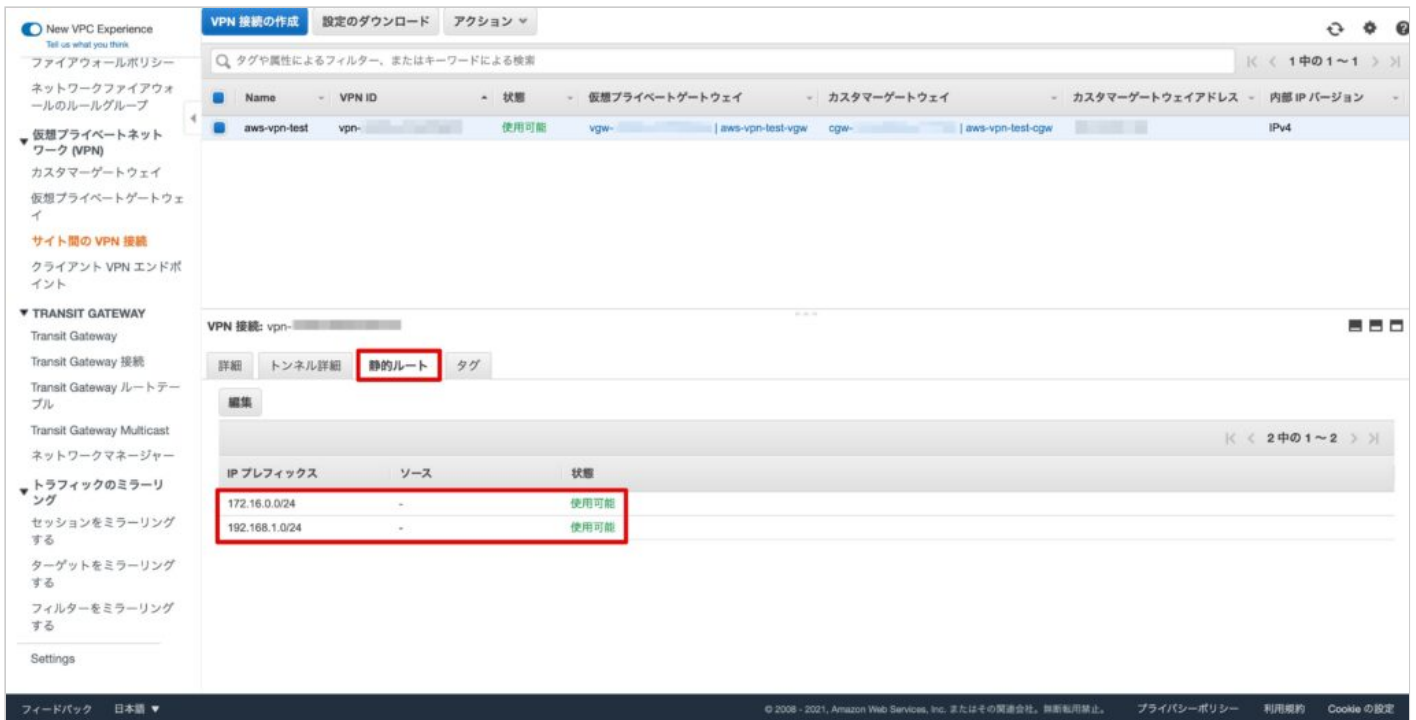
“aws-vpn-test”が作成され、使用可能となっています。



指定した内部 IPv4 CIDR でトンネル1が作成されています。



指定した静的ルートも追加され、使用可能となっています。



## サイト間VPN接続の削除

対象のスタックを削除することにより、自動的にサイト間VPN接続を削除することができます。  
AWS CLIを利用して、スタックを削除するには下記のコマンドを実行します。

[EC2]

```
aws cloudformation delete-stack --stack-name vpnteststack
```

コマンド実行直後にスタックのステータスを確認すると、“DELETE\_IN\_PROGRESS”となっています。

[EC2]

```
aws cloudformation describe-stacks
```

```
[ec2-user@ip-10-0-0-100 CloudFormation]$ aws cloudformation describe-stacks
{
  "Stacks": [
    {
      "StackId": "arn:aws:cloudformation:ap-northeast-1:xxxxxxxxxxxx:stack/vpnteststack/xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx",
      "DriftInformation": {
        "StackDriftStatus": "NOT_CHECKED"
      },
      "DeletionTime": "xxxx-xx-xxT22:53:08.707Z",
```

```

"Tags": [],
"CreationTime": "xxxx-xx-xxT22:23:16.497Z",
"StackName": "vpnteststack",
"NotificationARNs": [],
"StackStatus": "DELETE_IN_PROGRESS",
"DisableRollback": false,
"RollbackConfiguration": {}
}
]
}

```

AWSの画面上で確認すると、対象のスタックが削除されています。

The screenshot shows the AWS CloudFormation console for the stack 'vpnteststack'. The 'Events' tab is selected, displaying a list of 19 events. The first event, timestamped '06:59:16 UTC+0900', shows the stack 'vpnteststack' with a status of 'DELETE\_COMPLETE'. This row is highlighted with a red box. Other events show the deletion of associated resources like 'VPNConnection1' and 'VPNConnectionRoute'.

タイムスタンプ	論理 ID	ステータス	状況の理由
06:59:16 UTC+0900	vpnteststack	DELETE_COMPLETE	-
06:59:15 UTC+0900	VPNConnection1	DELETE_COMPLETE	-
06:58:59 UTC+0900	VPNConnection1	DELETE_IN_PROGRESS	-
06:58:58 UTC+0900	VPNConnectionRoute 2	DELETE_COMPLETE	-
06:58:58 UTC+0900	VPNConnectionRoute 1	DELETE_COMPLETE	-
06:58:42 UTC+0900	VPNConnectionRoute 1	DELETE_IN_PROGRESS	-
06:58:42 UTC+0900	VPNConnectionRoute 2	DELETE_IN_PROGRESS	-
06:58:40 UTC+0900	vpnteststack	DELETE_IN_PROGRESS	User Initiated
06:30:44 UTC+0900	vpnteststack	CREATE_COMPLETE	-
06:30:43 UTC+0900	VPNConnectionRoute 3	CREATE_COMPLETE	-

サイト間VPN接続に関しても、削除されていることがわかります。



これで、AWSサイト間VPNの構築（7.AWS CLI によるVPN接続の作成）の説明は完了です！