

Vol.155 2021年6月

# ECS 簡単デプロイツール

# **AWS Copilot CLI**

山本 悠介

DX 技術本部

#### 4 はじめに

AWS Copilot CLI は Amazon ECS CLI の後継にあたるもので、コンテナアプリケーションのビルド、リリース、運用を簡単に実現出来るツールになります。

本稿では、AWS Copliot CLI を利用し ECS クラスターを作成し、コンテナアプリケーションを動作させます。その簡単さが伝わればと思います。



シングルリージョン、マルチ AZ 構成でパブリックサブネット×2、プライベートサブネット×2の一般的な構成の上で ECS クラスターを構築し、コンテナを Service として起動させます。

#### 4 AWS Copilot CLI をインストールする

AWS Copilot CLI は Linux、macOS、Windows システムをサポートしていますが、今回は Linux システムにて進めます。

```
$ sudo curl -Lo /usr/local/bin/copilot
https://github.com/aws/copilot-cli/releases/latest/download/copilot-
linux \
    && sudo chmod +x /usr/local/bin/copilot \
    && copilot -help
```



#### **↓**AWS 認証情報を設定する

aws configure を実行して AWS の認証情報を設定します。

\$ aws configure

## ↓ デプロイするアプリケーションを準備する

Nginx サーバをデモサービスとしてデプロイしてみます。利用する Dockerfile を記載します。

Dockerfile 内容)

FROM nginx:alpine EXPOSE 80 COPY index.html /usr/share/nginx/html

表示する index.html を準備します。 index.html 内容例)

<html> <head><title>copilot test</title></head> <body>copilot test</body> </html>

適当なディレクトリ(例は nginx)に配置します。

nginx Dockerfile index.html



## 4 アプリケーションをセットアップする

AWS Copilot には Application という Service と Environment、Pipeline を取りまとめ る概念が存在します。上記で作成したディレクトリ内にて copilot init を実行しま す。

\$ copilot init

実行すると、

Application の名前:適当につけます。今回は copilot-demo とします ワークロードタイプ:今回は Load Balanced Web Service を選択します Service name: ECS ないに起動する Service 名。今回は front-end とします image:./Dockerfileを選択します port:公開するポート。今回は 80 とします

などそれぞれ聞かれるので、回答して進めます。回答し終わると、AWS Copilot がそれ ぞれ AWS リソースの作成を開始するので暫し待ちます。

## 4 テスト環境にデプロイする

アプリケーションのセットアップ終了後、test 環境へのデプロイ許可を問われるので、 y を 選択し、デプロイを行います。

リソースの作成状況は CloudFormation でも確認できます。暫し待ちましょう。

□ スタック (13) C	StackSet-copi	lot-demo-infra	structure-d5f0302	5-4f62-46e2-866f-046007	c8d4
Q スタック名によるフィルター	削除更新	スタックアクション 🔻	スタックの作成 ▼		
アクティブ▼	スタックの情報 イ	ג-עע אלא	はカ パラメータ テンプレ	マート 変更セット	
StackSet-copilot-demo-infrastructure- d5f03025-4f62-46e2-866f-046007c8d 4a5	<b>イベント</b> (12)				[
2021-06-03 15:12:48 UTC+0900 CREATE_IN_PROGRESS	Q 検索イベント				
copilot-demo-infrastructure-roles	タイムスタンプ	▼ 論理 ID	ステータス	状況の理由	
2021-06-03 15:10:43 UTC+0900 CREATE_COMPLETE	2021-06-03 15:13:19 UT	C+0900 PipelineB ucketPoli	uiltArtifactB O CREATE_COM	MPLETE -	
	2021-06-03 15:13:19 UT	C+0900 PipelineB ucketPoli	uiltArtifactB (i) cy CREATE_IN_PRC	OGRESS Resource creation Initiated	
	2021-06-03 15:13:18 UT	C+0900 PipelineB ucketPoli	uiltArtifactB () cy CREATE_IN_PRC		
	2021-06-03 15:13:16 UT	C+0900 PipelineB ucket	uiltArtifactB OCREATE_COM	MPLETE -	
	2021-06-03 15:12:55 UT	C+0900 KMSKey	CREATE_IN_PRC	OGRESS Resource creation Initiated	
	2021-06-03 15:12:55 UT	C+0900 ECRRepot	frontDASHen	MPLETE -	
	2021-06-03 15:12:54 UT	C+0900 ECRRepot	frontDASHen ③ CREATE_IN_PRC	OGRESS Resource creation Initiated	
	2021-06-03 15:12:53 UT	C+0900 PipelineB	uiltArtifactB	Resource creation Initiated	
	2021.06.03 15:12:52 11	C+0000 KMSKey	3		

#### デプロイが終了すると、エンドポイントが表示されます。

3.112.14.117 - ec2-user@ip-172-31-36-181:~/nginx VT	_		×
<u>File Edit Setup Control Window K</u> anjiCode <u>H</u> elp			1
- Creating the infrastructure for stack copilot-demo-test-front-end	[create	complet	tel 🔨
- Service discovery for your services to communicate within the VPC [2 7e]	[create	comple	te]
- Update your environment's shared resources	[update	complet	te]
[124.1s] - A security group for your load balancer allowing HTTP and HTTPS traffic	[create	comple	te]
- An Application Load Balancer to distribute public traffic to your services	[create	complet	te]
[94.0s] - An IAM Role for the Fargate agent to make AWS API calls on your behalf	[create	complet	te]
[2].4sj - A CloudWatch log group to hold your service logs	[create	comple	te]
[2,rs] - Creating the infrastructure for stack copilot-demo-test-front-end [200 a.]	[create	complet	te]
2203.35) - Service discovery for your services to communicate within the VPC - 2 7-1	[create	complet	te]
- Update your environment's shared resources	[update	complet	te]
- A security group for your load balancer allowing HTTP and HTTPS traffic	[create	comple	te]
- An Application Load Balancer to distribute public traffic to your services	[create	complet	te]
- An IAM Role for the Fargate agent to make AWS API calls on your behalf [21.4e]	[create	complet	te]
- A CloudWatch log group to hold your service logs	[create	complet	te]
[2.7s] - Creating the infrastructure for stack copilot-demo-test-front-end [200 a.]	[create	complet	te]
- Service discovery for your services to communicate within the VPC	[create	complet	te]
- Update your environment's shared resources	[update	comple	te]
- A security group for your load balancer allowing HTTP and HTTPS traffic	[create	complet	te]
- An Application Load Balancer to distribute public traffic to your services	[create	complet	te]
- An IAM Role for the Fargate agent to make AWS API calls on your behalf	[create	complet	te]
- A CloudWatch log group to hold your service logs	[create	complet	te]
- An ECS service to run and maintain your tasks in the environment cluster	[create	comple	te]
Deployments			
Revision Rollout Desired Running Failed Pending			
PRIMARY 1 [completed] 1 1 0 0			
- A target group to connect the load balancer to your service	[create	comple	te]
- An ECS task definition to group your containers and run them on ECS	[create	comple	te]
- An IAM role to control permissions for the containers in your tasks	[create	complet	te]
Peployed front-end, you can access it at <u>http://copil-Publi-PYXZISKN9F8I-588777</u> ; alb anazonaws.com	572.ap-n	ortheast	t-1.
[ec2-user8ip-172-31-36-181 nginx]\$			~



アクセスしてみると...

	_	^
0	* 4	) E
	9 📀	9 🛛 🖈 重

# 表示されています!

Services	ECR 〉 リポジトリ Private Public					
Amazon ECS						
Clusters Task definitions	プライベートリポジ	ト <b>リ</b> (5)	C	プッシュコマンドの表示	削除 編集 リオ	ポジトリを作成
	Q リボジトリを検索					(1>⊚
Amazon EKS		URI	作成時刻 🚽	タグのイミュータビリテ	ィー プッシュ時にスキャン	暗畳化タイプ
Clusters		- dkrecrar				
Amazon ECR	copilot- demo/front- end	northeast- 1.amazonaws.com/copilo demo/front-end	2021年6月03 t- 日 15:12:54	無効	無効	AES-256
Repositories	-		and a feature	fm // h	ám.ch	455.256
Public gallery 🔼				mxu	mxo	AE3-230
		*		無効	無効	AES-256
				無効	無効	AES-256
				無効	有効	KMS
UIDス ご開ルが開かせください Amazon ECS クラスター クラ	ラスター > copilot-demo-test-Cluste フラスター:copilot- ラスター上のリソースの詳細を取得し	r-qimy17R8LT6 demo-test-Cluster ≢す。	-qlrnYf7R8LT6		クラスターの更新	クラスターの削除
タスク定義						
Account Cottingo	クラスター ARN	arn:aws:ecs:ap-northeast-1:75809/	4741038-cluster/conilot-demo-te			
Account Settings			11-11000.Gubicinoopiloi-demo-te	st-Cluster-qim 17 R8L16		
Amazon EKS	ステータス	ACTIVE		st-Cluster-qim Y17R8L16		
Amazon EKS Clusters	ステータス 登録済みコンテナインスタンス	ACTIVE 0		st-Cluster-qim 11/ K8L 16		
Amazon EKS Clusters Amazon ECR	ステータス 登録済みコンテナインスタンス 保留中のタスクの数	ACTIVE 0 0 個の Fargate、0 個の EC2、0 個	の External	st-Cluster-qim tt/ K8L16		
Account Settings Amazon EKS Clusters Amazon ECR リポジトリ AWS Markethiace	ステータス 登録済みコンテナインスタンス 保留中のタスクの数 実行中のタスクの数	ACTIVE 0 0 個の Fargate、0 個の EC2、0 個 1 個の Fargate、0 個の EC2、0 個	の External の External	st-Cluster-qim Y1/ K8L i 6		
Account Settings Amazon EKS Clusters Amazon ECR リポジトリ AWS Marketplace ソフトウェアを探 す	ステータス 登録演みコンテナインスタンス 保留中のタスクの数 実行中のタスクの数 アクティブサービス数 ドレイニングサービス数	ACTIVE 0 0 個の Fargate、0 個の EC2、0 個 1 個の Fargate、0 個の EC2、0 個 1 個の Fargate、0 個の EC2、0 個 0 個の Fargate、0 個の EC2、0 個	() External () External () External () External	st-Ciluster-qim y1/ Kol i b		
Account Settings Amazon EKS Clusters Jポジトリ AWS Marketplace ソフトウェアを探 す サブスクリプショ ンマ	ステータス 整時済みコンテナインスタンス 保留中のタスクの数 実行中のタスクの数 アクティブサービス数 ドレイニングサービス数 ナービス タスク ECS インスタ	ACTIVE 0 0 個の Fargate、0 個の EC2、0 個 1 個の Fargate、0 個の EC2、0 個 1 個の Fargate、0 個の EC2、0 個 0 個の Fargate、0 個の EC2、0 個 <b>&gt;ス メトリクス タスクのス</b>	の External の External の External の External の External	キレCluster-qim Y/ KoLi 6 キャパシティープロバイダー		
Account Settings Amazon EKS Clusters Amazon ECR リポジトリ AWS Marketplace ソフトウェアを探 す サブスクリプショ ン C	ステータス 登録済みコンテナインスタンス 保留中のタスクの数 実行中のタスクの数 アクティブサービス数 ドレイニングサービス数 サービス タスク ECS インスタ 作成 更新 削除 ア	ACTIVE 0 0 個の Fargate、0 個の EC2.0 個 1 個の Fargate、0 個の EC2.0 個 1 個の Fargate、0 個の EC2.0 個 0 個の Fargate、0 個の EC2.0 個 ンス メトリクス タスクのス クション・	の External の External の External の External たたシュージング Tags =	<b>st-Cluster-qim 11/ KoL i 6</b> キヤパシティープロパイダー 	日: 2021年6月03日 3:31:24 午後	) 分前) 😂 😧
Account Settings Amazon EKS Clusters Amazon ECR リポジトリ AWS Marketplace ソフトウェアを探 す サブスクリプショ ン C	ステータス 登録演みコンテナインスタンス 保留中のタスクの数 実行中のタスクの数 アクティブサービス数 ドレイニングサービス数 サービス タスク ECS インスタ 作成 更新 削除 ア マこのページのフィルター j	ACTIVE 0 0 個の Fargate、 0 個の EC2、 0 幅 1 個の Fargate、 0 個の EC2、 0 個 1 個の Fargate、 0 個の EC2、 0 個 0 個の Fargate、 0 個の EC2、 0 個 ンス メトリクス タスクのス クション・ 型動タイプ ALL ・ サート	(C External C E	st-Cluster-qim 11/ KoL 16 キャパシティープロバイダー 風終史新	∃: 2021年6月03日 3.31:24 午後 (	)分前) 2 0
Account Settings Amazon EKS Clusters Amazon ECR リポジトリ AWS Marketplace ソフトウェアを探 す サブスクリプショ ン C	ステータス 登録演みコンテナインスタンス 保留中のタスクの数 実行中のタスクの数 アクティブサービス数 ドレイニングサービス数 サービス タスク ECS インスタ 作成 更新 削除 ア てこのページのフィルター 1 サービス名	ACTIVE 0 0 個の Fargate、 0 個の EC2. 0 個 1 個の Fargate、 0 個の EC2. 0 個 1 個の Fargate、 0 個の EC2. 0 個 0 個の Fargate、 0 個の EC2. 0 個 ンス メトリクス タスクのス クション・ 記動タイプ ALL ・ サーt	の External の External の External の External の External の External センジュージング Tags = セスタイプ ALL ータス… サービスタ… タ	st-Cluster-qim 11/ KoL 16 キャパシティープロバイダー 最終更新 スク定義 必要なタス	日: 2021年6月03日 3.31:24 午後 ( 実行中の今 起動タイプ	1分前) <b>2 0</b> く 1-1 > ブラットフ

#### ECR や ECS クラスター、Service が作成されているのも確認できます。

パブリックサブネット×2、プライベートサブネット×2 が作成されていることも確認で きます。

copilot-{アプリケーション名}-{環境名}-pub0の様に名前が付くようですね。

New VPC Experience     Tell us what you think	サブネット (4) 情報			С アクション ▼ サブネットを作成		
VPC ダッシュボード New	Q サブネットをフィルター				< 1 > 🕲	
VPC でフィルタリング: Q, VPC の選択	search: copilot-demo × フィルター	をクリア				
VIRTUAL PRIVATE	□ Name ▽	サプネット ID マ 状態	ag	VPC		
CLOUD	copilot-copilot-demo-test-priv0	subnet-0b7602383fad71ebf	Available	vpc-0164a3b0dbf4a3c60   copilot-copilot-demo-tes	t 10.0.2.0/24	
VPC New	copilot-copilot-demo-test-pub0	subnet-009f3481e9b716654	Available	vpc-0164a3b0dbf4a3c60   copilot-copilot-demo-tes	t 10.0.0/24	
サプネット New	conilot-conilot-demo-test-publ	subnet-0f5784c7fcaaf5c00	Available	voc-0164a3b0dbf4a3c60   copilot-copilot-demo-tes	t 10.0.1.0/24	
ルートテーブル New			transitione.	The offers of the second	10.0.7.0/24	
インターネットゲートウ	Copilot-copilot-demo-test-privi	subnet-0deabd2dc0501f416	waitable	vpc-0164a5b0dbf4a5c60 [ copilot-copilot-demo-tes	t 10.0.3.0/24	

#### 後片付け

copilot app deleteを実行して作成したリソースを全て削除します。

```
$ copilot app delete
```

#### \rm まとめ

以上、AWS Copilot CLI にてコンテナアプリケーションのビルド、リリースを行いました。簡単さが伝わりましたでしょうか。

今回はワークロードとして Load Balanced Web Service を選びましたが、内部通信用の Backend Service、バッチなどに使用できる Scheduled job なども用意されています。

既存のサブネットを指定してコンテナをデプロイすることも可能です。

CI/CD パイプライン(CodePipeline)もコマンドで作成することが出来ます。

AWSのアカウントさえあればすぐ試せるので興味がある方はやってみてください!



AWS Copilot CLI - AWS Copilot CLI https://aws.github.io/copilot-cli/

Amazon ECS アプリケーションのデプロイによる AWS Copilot の開始方法 - Amazon Elastic Container Service https://docs.aws.amazon.com/ja\_jp/AmazonECS/latest/developerguide/getting-tarted-aws-copilot-cli.html



**GSLetterNeo** Vol.155 2021年6月20日発行 発行者 株式会社 SRA 先端技術研究所

編集者 土屋正人 熊澤努 方 学芬 バックナンバー http://www.sra.co.jp/gsletter お問い合わせ gsneo@sra.co.jp

## SLetterNeo Vol 155

