

AWSハンズオン

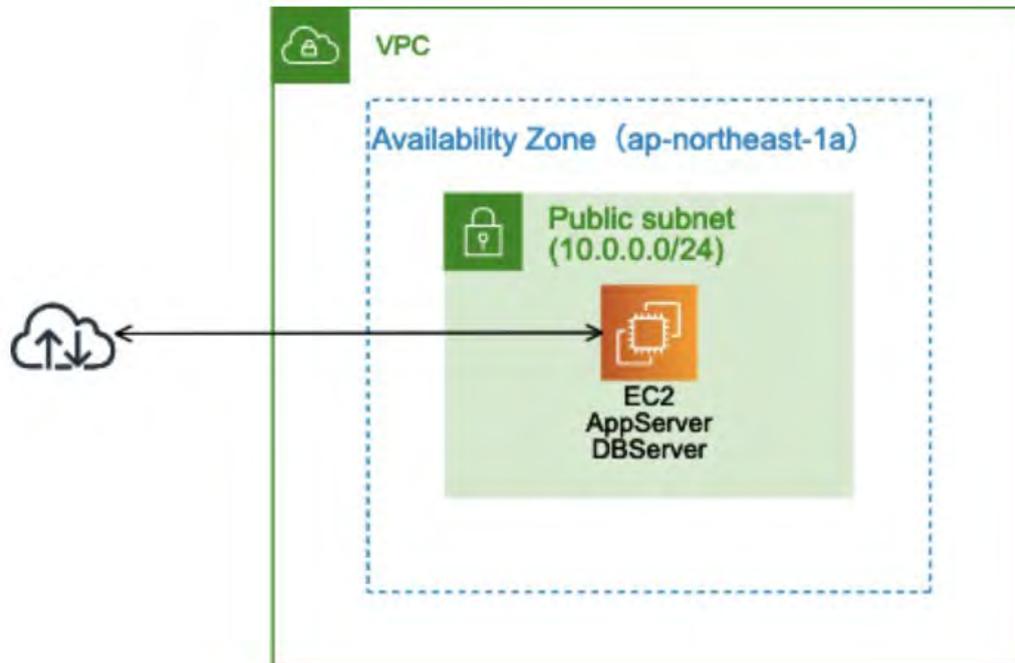
[フェーズ1]	
~サーバー 1 台構成で Redmine 環境を構築~	4
▼フェーズ 1-1: コンソールへのログインと VPC(ネットワーク)の作成	4
▼フェーズ 1-2: サブネットを追加作成	15
▼フェーズ 1-3: Amazon EC2 インスタンスの作成	21
▼フェーズ 1-4: Elastic IP(固定 IP)の割り当て	35
[フェーズ 2]	
~拡張性を向上しつつDB 運用負荷を軽減する構成を構築~	40
▼フェーズ 2-1: Amazon RDS のセキュリティグループを作成	40
▼フェーズ 2-2: DB サブネットグループを作成	43
▼フェーズ 2-3: Amazon RDS インスタンスを作成	47
▼フェーズ 2-4: RDSに接続	55
▼フェーズ 2-5: Redmine S3対応	60
[フェーズ 3]	
~ロードバランサーを使った負荷分散環境を構築~	77
▼フェーズ 3-1: Web サーバーの AMI(パッケージ)を作成	77
▼フェーズ 3-2: 2 個目の Amazon EC2 インスタンスを作成	80
▼フェーズ 3-3: Elastic Load Balancing(ロードバランサー)を作成	86
▼フェーズ 3-4: Elastic Load Balancing 経由でアクセス	96
▼フェーズ 3-5: セキュリティグループ設定変更	97
[フェーズ 4]	
~ Amazon RDS を Multi-AZ 構成に変更 ~	100
▼フェーズ 4: Amazon RDS を Multi-AZ 構成に変更	100
~ 構築した環境の後片付け ~	108

参考サイト

<https://aws.amazon.com/jp/getting-started/projects/scalable-wordpress-website/>

[フェーズ1]

~サーバー 1 台構成で *Redmine* 環境を構築~



▼フェーズ 1-1: コンソールへのログインと VPC(ネットワーク)の作成

ステップ 1-1-1: AWS マネジメントコンソールにログインする



アカウント:

ユーザー名:

パスワード:

サインイン

1

ルートアカウント認証情報を使用してサインイン
パスワードをお忘れですか?

AWSでのワークロードの 起動に役立つリソースセンター

どなたでも簡単にAWSを開始できるチュートリアルや
中・上級者向けのユースケース別ガイド、トレーニング等
をご活用ください

詳細はこちら »

1. アカウント、ユーザー名、パスワード等を入力して、AWSマネジメントコンソールにログインします。

ログイン方法は利用するアカウント種類によって異なります。

IAM アカウントを御利用の場合

- 会社等で、IAM アカウントをあらかじめ準備されているケース
- 事前にログイン情報が記載された csv ファイル (user1.csv等) を確認してください。
- そのファイルに、ログイン用の URL、User Name、パスワードが記載されていますので、それに従ってログインしてください。

AWS のルートアカウント(個人アカウント)をご利用の場合

- <https://console.aws.amazon.com> にブラウザでアクセスしてください。
- アカウントの E メールアドレスとパスワードでログインしてください。
 1. 左上部の「ホームに戻るボタン」をクリックします。
 2. すべてのサービスを表示 をクリックします。

ステップ 1-1-2: リージョンを変更する

The screenshot shows the AWS Management Console interface. At the top right, the current region is set to '東京' (Tokyo), which is highlighted with a red box and a circled '1'. Below this, a list of available regions is displayed. The 'アジアパシフィック (東京) ap-northeast-1' region is highlighted with a red box and a circled '2'. The interface also shows various AWS services, recent accessed services, and solution architectures.

1 「リージョン」をクリックします。

2 「アジアパシフィック (東京)」を選択します。

1. 「リージョン」をクリックします。
2. 「アジアパシフィック (東京)」を選択します。

ステップ 1-1-3: VPC 管理ページを開く



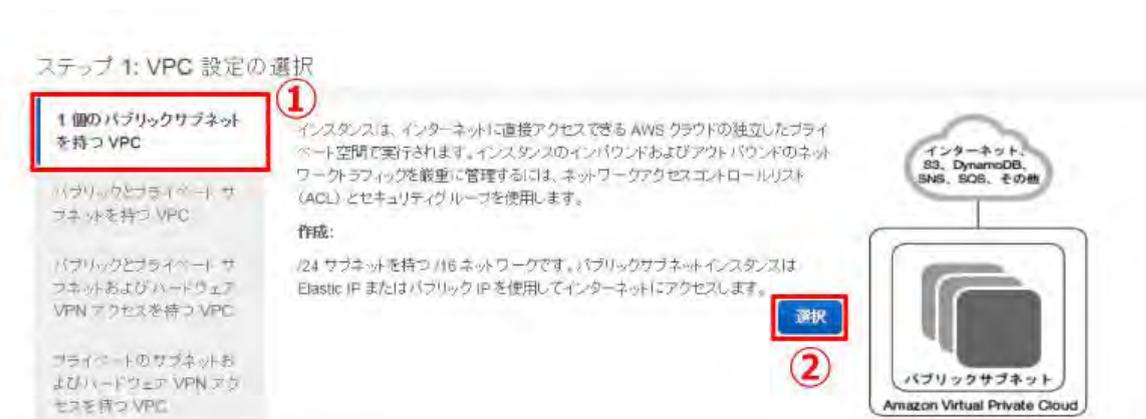
1. クリック画面上部の検索窓から「VPC」と入力します。
2. 「VPC」をクリックします。

ステップ 1-1-4: VPC の作成ウィザードを開始する



1. 「VPC ウィザードの起動」をクリックします。

ステップ 1-1-5: VPC 作成ウィザード



1. 「1 個のパブリックサブネットを持つ VPC」をクリックします。
2. 「選択」をクリックします。

ステップ 2: 1 個のパブリックサブネットを持つ VPC

IPv4 CIDR ブロック: (1) (65531 利用可能な IP アドレス)

IPv6 CIDR ブロック: IPv6 CIDR ブロックなし
 Amazon が提供した IPv6 CIDR ブロック

VPC 名: (2)

パブリックサブネットの IPv4 CIDR: (3) (251 利用可能な IP アドレス)

アベイラビリティゾーン: (4)

サブネット名:

VPC を作成した後は、より多くのサブネットを追加できます。

サービスエンドポイント

DNS ホスト名を有効化: はい いいえ

ハードウェアのテナンシー:

(5)

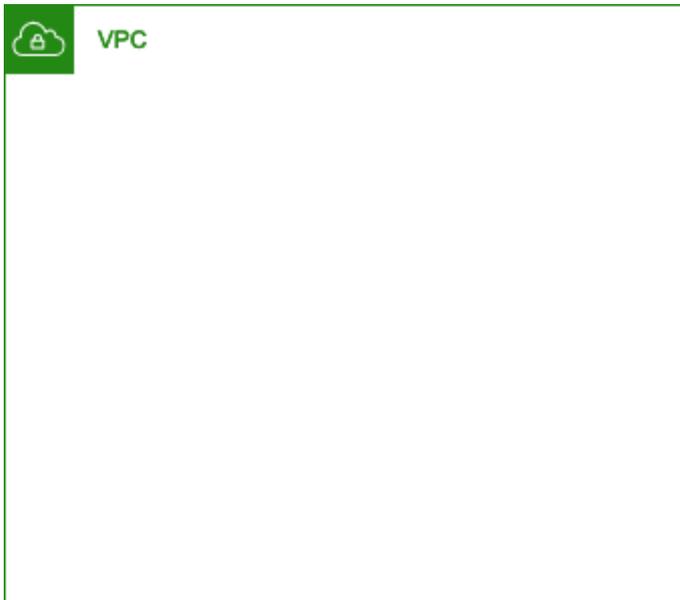
1. 「10.0.0.0/16」であることを確認します。
2. 「handson-自分のユーザー名」と入力します。
例) handson-user1
3. 「10.0.0.0/24」であることを確認します。
4. 「ap-northeast-1a」であることを確認します。



VPC が作成されました。

1. 「OK」をクリックします。

以下の図の緑枠である「VPC」を作成しました。
これでサーバーを配置できるネットワークを作ったことになります。



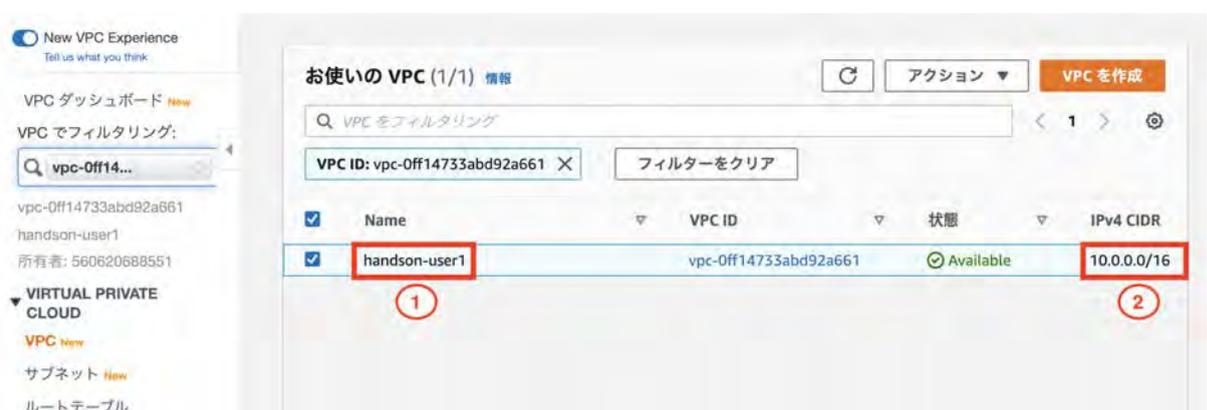
ステップ 1-1-6: VPC のフィルタリング設定



VPCでフィルタリングします。先ほど作成したVPCはすぐにはプルダウンメニューに含まれないため、一度画面をリロードする必要があります。

1. 画面をリロードする
2. 画面左上の「VPC でフィルタリング」のプルダウンメニューから先ほど作成した VPC を選択してください。
※他VPC と間違わないように注意してください。

ステップ 1-1-7: 作成された VPC の確認



1. 「VPC」をクリックします。
2. 先ほど作成した VPC が存在するか(正しく絞り込めているか)を確認します。

3. 「10.0.0.0/16」であることを確認します。

ステップ 1-1-8: ウィザードで作成されたサブネットを確認

The screenshot displays the AWS Management Console interface for managing subnets. On the left sidebar, the 'サブネット' (Subnets) option is highlighted with a red box and a circled '1'. The main content area shows a table of subnets with the following columns: Name, Subnet ID, Status, VPC, and IPv4 CIDR. The first row, 'パブリックサブネット' (Public Subnet), is selected with a checkmark and highlighted with a red box and a circled '2'. The 'IPv4 CIDR' for this subnet is '10.0.0.0/24', highlighted with a red box and a circled '3'. Below the table, the details for 'subnet-00914e4d6dc9cd3d4 / パブリックサブネット' are shown, with the 'Availability Zone' field set to 'ap-northeast-1a', highlighted with a red box and a circled '4'.

1. 「サブネット」をクリックします。
2. サブネットを選択します。
3. 「10.0.0.0/24」であることを確認します。
4. 「ap-northeast-1a」であることを確認します。

ステップ 1-1-9: 作成されたサブネットの Route Table を確認

VPC ダッシュボード

VPC でフィルタリング: vpc-075ecb...

Virtual Private Cloud

VPC

サブネット

ルートテーブル

インターネットゲートウェイ

Egress Only インターネットゲートウェイ

DHCP オプションセット

Elastic IP

エンドポイント

エンドポイントのサービス

サブネットの作成 アクション

タグや属性によるフィルター, またはキーワードによる検索

4 中の 1 ~ 4

Name	サブネット ID	状態	VPC	IPv4 CIDR	利用可能な IPv4	IPv6 CIDR
パブリック...	subnet-02ad147fa93cbe2c1	available	vpc-075ecbe4fa77857ea ...	10.0.0.0/24	251	-
パブリック...	subnet-0f0fc03e797c4941b	available	vpc-075ecbe4fa77857ea ...	10.0.1.0/24	251	-
プライベート...	subnet-0c0a6430b5136cf34	available	vpc-075ecbe4fa77857ea ...	10.0.2.0/24	251	-
プライベート...	subnet-0fcd360d06af1849f	available	vpc-075ecbe4fa77857ea ...	10.0.3.0/24	251	-

サブネット: subnet-02ad147fa93cbe2c1

説明 フローログ **ルートテーブル** ネットワーク ACL タグ 共有

ルートテーブルの関連付けの編集

ルートテーブル: rtb-03c1b7f45c3d081e8

2 中の 1 ~ 2

送信先	ターゲット
10.0.0.0/16	local
0.0.0.0/0	igw-002a4a076f98873ff

VPC のネットワークアドレス 10.0.0.0/16 のターゲットが local に、デフォルトルートの 0.0.0.0/0 のターゲットがインターネットゲートウェイ (igw-XXXX)になっており、インターネットと通信できる設定になっています。

1. 「ルートテーブル」をクリックします。
2. 内容を確認します。

確認したサブネットは図の緑色の領域のことで...



▼ フェーズ 1-2: サブネットを追加作成



ステップ 1-2-1: サブネットを 3 つ追加作成

The screenshot shows the AWS Management Console interface for managing subnets. The 'Create Subnet' button is highlighted with a red box and a circled '1'. The table below shows one existing public subnet.

Name	サブネット ID	状態	VPC
パブリックサブネット	subnet-00914e4d6dc9cd3d4	Available	vpc-0ed...

VPC > サブネット > サブネットを作成

サブネットを作成 情報

VPC

VPC ID
この VPC にサブネットを作成します。

vpc-0edcc812ddf7e67e (handson-user1) 2

関連付けられた VPC CIDR

IPv4 CIDR
10.0.0.0/16

サブネットの設定

サブネットの CIDR ブロックとアベイラビリティゾーンを指定します。

1. 「サブネットの作成」をクリックします。
2. VPC IDはフェーズ1-1-5で作成したものを選択してください。

サブネットの設定

サブネットの CIDR ブロックとアベイラビリティゾーンを指定します。

サブネット 1 (1 個中)

サブネット名

「Name」というキーと、指定した値を使用してタグを作成します。

パブリック サブネットc 1

名前の長さは最大 256 文字です。

アベイラビリティゾーン 情報

サブネットが存在するゾーンを選択するが、Amazon が選択するゾーンを受け入れます。

アジアパシフィック (東京) / ap-northeast-1c 2

IPv4 CIDR ブロック 情報

10.0.1.0/24 3

▼ タグ - オプション

キー

Name

値 - オプション

パブリック サブネットc

削除

新しいタグを追加

さらに 49 個の タグ を追加できます。

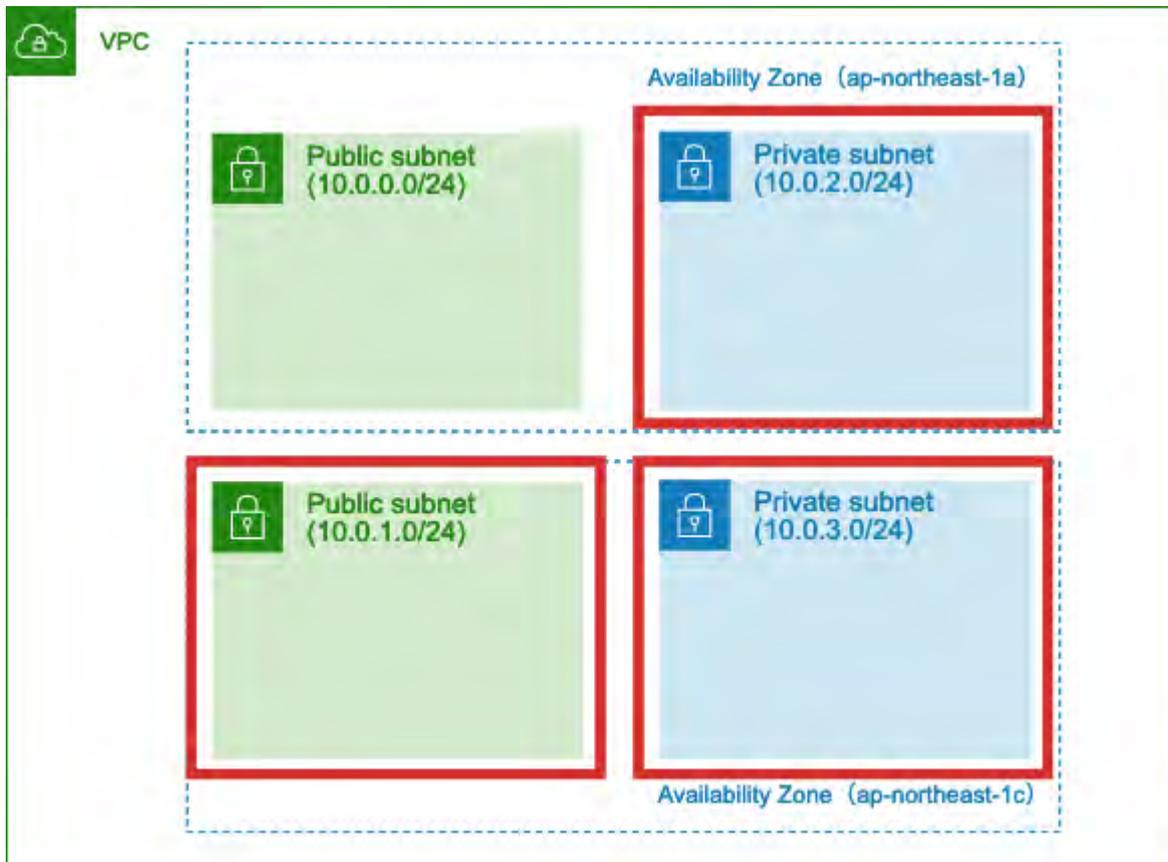
削除

新しいサブネットを追加 4

- 以下の表と図を参考にサブネットを3つ作成してください。(上スクリーンショットは表の1つ目のサブネット作成例)
- 「新しいサブネットを追加」をクリックすることで2つ目、3つ目のサブネットも同時に作成することができます。

	サブネット名 (①)	アベイラビリティゾーン(②)	CIDRブロック (③)
1つ目	パブリックサブネットc	ap-northeast-1c	10.0.1.0/24
2つ目	プライベートサブネットa	ap-northeast-1a	10.0.2.0/24

3つ目	プライベートサブネットc	ap-northeast-1c	10.0.3.0/24
-----	--------------	-----------------	-------------



サブネット 3 (3 個中)

サブネット名
「Name」というキーと、指定した値を使用してタグを作成します。

プライベート サブネットc

名前の長さは最大 256 文字です。

アベイラビリティゾーン [情報](#)
サブネットが存在するゾーンを選択するか、Amazon が選択するゾーンを受け入れます。

アジアパシフィック (東京) / ap-northeast-1c

IPv4 CIDR ブロック [情報](#)

10.0.3.0/24

▼ タグ - オプション

キー	値 - オプション	
Name	プライベート サブネットc	削除

[新しいタグを追加](#)

さらに 49 個の タグ、を追加できます。

[削除](#)

[新しいサブネットを追加](#)

1

キャンセル [サブネットを作成](#)

1. 3つとも入力後「サブネットを作成」ボタンをクリックします。

図の赤枠の部分を作成しました。

ステップ 1-2-2: 全てのサブネットを確認

3 件のサブネットが正常に作成されました。 subnet-023010195ad625cd0, subnet-0500dc2cc5e7a7ba9, subnet-0aac33f339a943358

サブネット (3) 情報

フィルターをクリア

VPC ID	名前タグ	所有者	サブネット ID	状態	VPC
vpc-0ff14733abd92a661	handson-user1	560620688551	subnet-0aac33f339a943358	Available	vpc-0
			subnet-0500dc2cc5e7a7ba9	Available	vpc-0
			subnet-023010195ad625cd0	Available	vpc-0

サブネットが作成できたら、作成した3つのサブネットだけが表示されるフィルターがかかっているため、一度クリアしてVPCでフィルタリングをします。

1. 「フィルターをクリア」ボタンをクリックする。
2. 「vpcでフィルタリング」からフェーズ1-1-5で作成したvpcを選択してください。

サブネットの作成 アクション

タグや属性によるフィルター、またはキーワードによる検索

4 中の 1 ~ 4

Name	サブネット ID	状態	VPC	IPv4 CIDR	利用可能な IPv4	IPv6 CI
パブリック...	subnet-0bdeae1e233de5ee2	available	vpc-047f4caec8f8c5613 ...	10.0.0.0/24	251	-
パブリック...	subnet-0d33bb82d8787124b	available	vpc-047f4caec8f8c5613 ...	10.0.1.0/24	251	-
プライベート...	subnet-09dd92e84b07d72bd	available	vpc-047f4caec8f8c5613 ...	10.0.2.0/24	251	-
プライベート...	subnet-03a01402d0edbc1cf	available	vpc-047f4caec8f8c5613 ...	10.0.3.0/24	251	-

ウィザードで作成したサブネットと追加したサブネットを確認します。
 パブリックサブネットが2、プライベートサブネットが2、
ap-northeast-1aアベイラビリティゾーンが2、**ap-northeast-1c**アベイラビリティゾーンが2
 作成していることを確認します。

ステップ 1-2-3: パブリックサブネットのルートテーブルを変更



1. サブネットの検索窓で「10.0.1.0」と入力し、フィルタをかけます。



追加したサブネット「10.0.1.0」を実際にインターネットと通信できるように、ルートテーブルの割り当てを変更します。

1. 「10.0.1.0/24」のサブネットをクリックします。
2. 「ルートテーブル」をクリックします。
3. 「ルートテーブルの関連付けの編集」をクリックします。

サブネット > ルートテーブルの関連付けの編集

ルートテーブルの関連付けの編集

サブネット ID subnet-0d33bb82d8787124b

ルートテーブル ID* 1

送信先 ターゲット

10.0.0.0/16	local
0.0.0.0/0	igw-061c06c43d4f95f87

2

* 必須

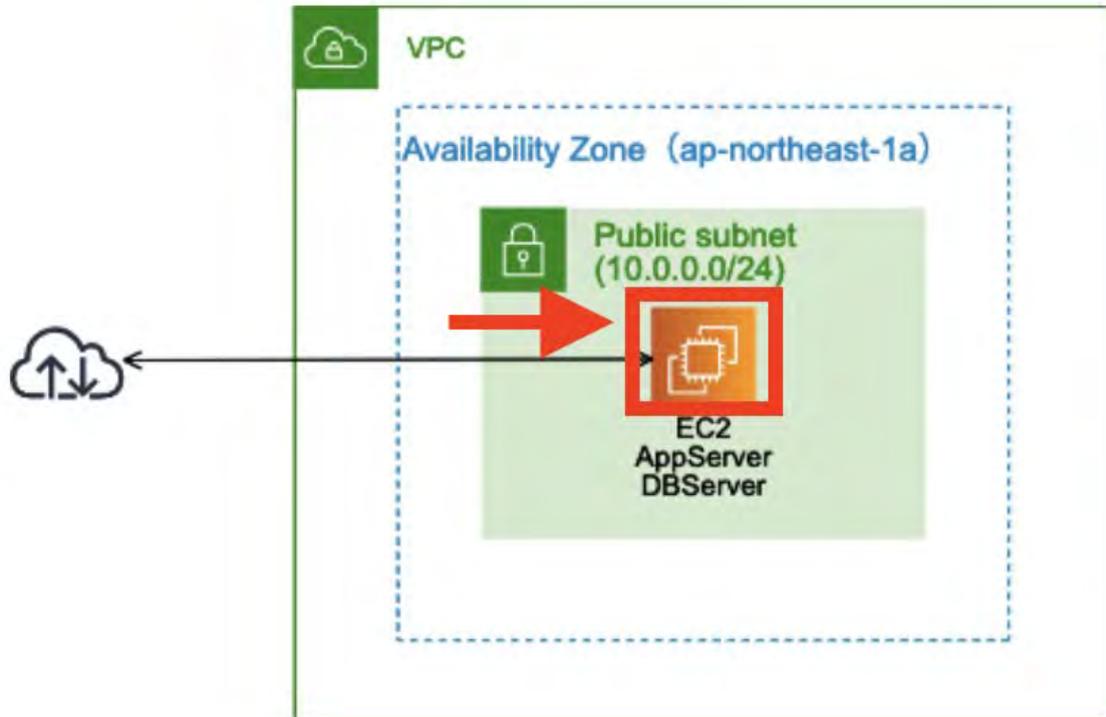
キャンセル 3

1. これまでと異なるものを選択してください。(メインルートテーブルではない方)
※この VPC にはルートテーブルが 2 つしかありません
2. 「0.0.0.0/0」が表示されていることを確認します。
3. 「保存」をクリックします。

▼フェーズ 1-3: Amazon EC2 インスタンスの作成

図のオレンジの部分を作成します。

インスタンスとはAWSクラウドにある仮想サーバーのことです。



ステップ 1-3-1: EC2 管理ページを開く



1. 「EC2」を入力します。
2. 「EC2」をクリックします。

ステップ 1-3-2: EC2 インスタンスの作成



Web サーバーの作成を行います。

1. 「インスタンス」をクリックします。
2. 「インスタンスを起動」をクリックします。

1. AMI の選択 2. インスタンスタイプを選択 3. インスタンスの設定 4. ストレージの追加 5. タグの追加 6. セキュリティグループの設定 7. 確認

ステップ 1: Amazon マシンイメージ (AMI) キャンセルして終了

AMI は、インスタンスの作成に必要なソフトウェア構成 (OS、アプリケーションサーバー、アプリケーション) を含むテンプレートです。AMI は、AWS が提供するもの、ユーザーコミュニティが提供するもの、または AWS Marketplace に掲載されているものを選択できます。独自の AMI のいずれかを選択することもできます。

redmine

クイックスタート (0)

マイ AMI (0)

AWS Marketplace (12)

コミュニティ AMI (426)

Categories

All Categories

Infrastructure Software (1)

DevOps (10)

Business Applications (8)

Operating System

All Linux/Unix

Redmine Certified by Bitnami

★★★★★ (19) | 4.1.0-0 on Ubuntu 16.04 | 以前のバージョン | 提供: Bitnami

Linux/Unix, Ubuntu 16.04 | 64 ビット Amazon マシンイメージ (AMI) x86 | 更新済み: 2019/12/26

Up-to-date and secure image. Redmine is an open source management application. It includes a tracking issue system, Gantt charts for a visual view of projects and deadlines, and supports SCM integration for version control.

詳細

Redmine Powered by Intuz

★★★★★ (0) | 4.0.3 | 提供: Intuz

Linux/Unix, Ubuntu 16.04 | 64 ビット Amazon マシンイメージ (AMI) x86 | 更新済み: 2019/10/25

Intuz Redmine has apache, mysql, ruby, php, phpryadmin, webmin, redmine and scripts which make it easy for you to use Redmine. We have integrated phpryadmin, webmin and scripts for backup, update and password recovery.

詳細

無料利用枠の対象

無料利用枠の対象

選択

選択

1. 「AWS Marketplace」をクリックします。
2. 「redmine」と入力しエンターを押す。
3. 「Redmine Certified by Bitnami」を選択します。

Redmine Certified by Bitnami



無料利用枠の対象

Redmine Certified by Bitnami

Redmine is a project management and issue-tracking platform. It enables teams to manage multiple projects from a single user interface. This solution provides enterprise-grade features such as LDAP user access management, multiple database support, and bug tracking tools. It is fully integrated with Git and Mercurial.

This image is configured ...

[AWS Marketplace での詳細の表示](#)

詳細

料金に関する詳細情報

時間料金	インスタンスタイプ	ソフトウェア	EC2	合計
	t2.micro	\$0.00	\$0.015	\$0.015/時間
	t2.small	\$0.00	\$0.03	\$0.03/時間
	t2.medium	\$0.00	\$0.061	\$0.061/時間
	t2.large	\$0.00	\$0.122	\$0.122/時間
	t2.xlarge	\$0.00	\$0.243	\$0.243/時間
	t2.2xlarge	\$0.00	\$0.486	\$0.486/時間
	t3a.micro	\$0.00	\$0.012	\$0.012/時間
	t3a.small	\$0.00	\$0.025	\$0.025/時間
	t3a.medium	\$0.00	\$0.049	\$0.049/時間
	t3a.large	\$0.00	\$0.098	\$0.098/時間
	t3a.xlarge	\$0.00	\$0.196	\$0.196/時間

製品の詳細

担当: Bitnami

お客様による評価: ★★★★★ (19)

最新バージョン: 4.1.0-0 on Ubuntu 16.04

基本オペレーティングシステム: Linux/Unix, Ubuntu 16.04

実施形式: 64 ビット Amazon マシンイメージ (AMI) x86

ライセンス契約: エンドユーザーライセンス契約

Marketplace での使用開始日: 2016/10/28

ハイライト

- Manage and track multiple projects, with a separate document manager, wiki, calendar, Gantt charts, forums, and time tracking for each one. Create custom fields by project for bugs, time tracking, and users.

キャンセル **Continue**

1. 「Continue」をクリックします。

1. AMI の選択 2. インスタンスタイプの選択 3. インスタンスの設定 4. ストレージの追加 5. タグの追加 6. セキュリティグループの設定 7. 確認

ステップ 2: インスタンスタイプの選択

Amazon EC2 では、異なるユースケースに合わせて最適化されたさまざまなインスタンスタイプが用意されています。インスタンスとは、アプリケーションを実行できる仮想サーバーです。インスタンスタイプはさまざまな CPU、メモリ、ストレージ、ネットワークキャパシティの組み合わせによって構成されているため、使用するアプリケーションに合わせて適切なリソースの組み合わせを柔軟に選択できます。インスタンスタイプおよびそれをコンピューティングのニーズに適用する方法に関する詳細はこちら。

フィルター条件: すべてのインスタンスファミリー 現行世代 列の表示/非表示

現在選択中: t3a.small (- ECU, 2 vCPU, 2.2 GHz, -, 2 GiB メモリ, EBS のみ)

注: ベンダーは、この製品を使用して最良の結果を得るために t3a.small 個のインスタンス (またはそれ以上) を使用することを推奨しています。

	ファミリー	タイプ	vCPU	メモリ (GiB)	インスタンスストレージ (GB)	EBS 最適化利用	ネットワークパフォーマンス	IPv6 サポート
	t2	t2.nano	1	0.5	EBS のみ	-	低から中	はい
①	t3	t3.nano	2	0.5	EBS のみ	はい	最大 5 ギガビット	はい
	t3	t3.micro	2	1	EBS のみ	はい	最大 5 ギガビット	はい
	t3	t3.small	2	2	EBS のみ	はい	最大 5 ギガビット ②	はい

キャンセル 戻る 確認と作成 次のステップ: インスタンスの詳細の設定

1. 「t3.micro」を選択します。
2. 「次のステップ: インスタンスの詳細の設定」をクリックします。

1. AMI の選択 2. インスタンスタイプの選択 3. インスタンスの設定 4. ストレージの追加 5. タグの追加 6. セキュリティグループの設定 7. 確認

ステップ 3: インスタンスの詳細の設定

要件に合わせてインスタンスを設定します。同じ AMI からの複数インスタンス作成や、より低料金を実現するためのスポットインスタンスのリクエスト、インスタンスへのアクセス管理ロール割り当てなどを行うことができます。

インスタンス数 ① 1 Auto Scaling グループに作成する ①

購入のオプション ① スポットインスタンスのリクエスト

ネットワーク ① vpc-03308f52025f13277 | handson-user1 ① 新しい VPC の作成

サブネット ① subnet-05d68f524017631b | パブリックサブネット ② 新しいサブネットの作成
251 個の IP アドレスが利用可能

自動割り当てパブリック IP ① 有効 ③

配置グループ ① インスタンスをプレイズメントグループに追加します。

キャパシティの予約 ① 開く

ドメイン結合ディレクトリ ① ディレクトリなし ④ 新しいディレクトリの作成

IAM ロール ① なし ④ 新しい IAM ロールの作成

CPU オプション ① CPU オプションを指定

シャットダウン動作 ① 停止

停止 - 休止動作 ① 停止動作に休止動作を追加する

終了保護の有効化 ① 誤った終了を防止します

モニタリング ① CloudWatch 詳細モニタリングを有効化

キャンセル 戻る 確認と作成 次のステップ: ストレージの追加

インスタンスの詳細設定を行います。VPC を選択するところでは、フェーズ1-1-5で作成した VPC を選択してください。

1. フェーズ1-1-5で作成した VPC を選択します。
2. 「10.0.0.0/24 | パブリックサブネット | ap-northeast-1a」を選択します。
※ プライベートサブネットと間違えないこと
3. 「有効」を選択します。
4. 「次のステップ: ストレージの追加」をクリックします。

1. AMI の選択 2. インスタンスタイプの選択 3. インスタンスの設定 4. **ストレージの追加** 5. タグの追加 6. セキュリティグループの設定 7. 確認

ステップ 4: ストレージの追加

インスタンスは次のストレージデバイス設定を使用して作成されます。インスタンスに追加の EBS ボリュームやインスタンスストアボリュームをアタッチするか、ルートボリュームの設定を編集することができます。また、インスタンスを作成してから追加の EBS ボリュームをアタッチすることもできますが、インスタンスストアボリュームはアタッチできません。Amazon EC2 のストレージオプションに関する [詳細](#) はこちらをご覧ください。

ボリュームタイプ	デバイス	スナップショット	サイズ (GiB)	ボリュームタイプ	IOPS	スループット (MB/秒)	終了時に削除	暗号化
ルート	/dev/xvda	snap-0a3a21e764482ce15	8	汎用 SSD (gp2)	100 / 3000	該当なし	<input checked="" type="checkbox"/>	暗号化

新しいボリュームの追加

キャンセル 戻る **確認と作成** **次のステップ: タグの追加**

ストレージは変更せずに、次に進みます。

1. 「次のステップ: タグの追加」をクリックします。

1. AMI の選択 2. インスタンスタイプの選択 3. インスタンスの設定 4. ストレージの追加 5. **タグの追加** 6. セキュリティグループの設定 7. 確認

ステップ 5: タグの追加

タグは、大文字と小文字が区別されるキーと値のペアから構成されます。たとえば、キーに「Name」、値に「Webserver」を使用してタグを定義することができます。タグのコピーは、ボリューム、インスタンス、またはその両方に適用できます。タグは、すべてのインスタンスとボリュームに適用されます。Amazon EC2 リソースのタグ付けに関する [詳細](#) はこちら。

キー (最大 128 文字)	値 (最大 256 文字)	インスタンス	ボリューム
Name	webserver#1-user1	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

タグの追加

キャンセル 戻る **確認と作成** **次のステップ: セキュリティグループの設定**

インスタンスを区別できるようにタグに名前を設定します。(webserver#1-user1 等ユーザー名を付けます。)

1. 「タグの追加」をクリックします。

- キーに「Name」と入力します。
- 「webserver#1- ユーザー名」とします。
例) [webserver#1-user1]
- 「次のステップ: セキュリティグループの設定」をクリックします。

1. AMI の選択 2. インスタンスタイプの選択 3. インスタンスの設定 4. ストレージの追加 5. タグの追加 6. セキュリティグループの設定 7. 確認

ステップ 6: セキュリティグループの設定

セキュリティグループは、インスタンスのトラフィックを制御するファイアウォールのルールセットです。このページで、特定のトラフィックに対してインスタンスへの到達を許可するルールを追加できます。たとえば、ウェブサーバーをセットアップして、インターネットトラフィックにインスタンスへの到達を許可する場合、HTTP および HTTPS ポートに無制限のアクセス権限を与えます。新しいセキュリティグループを作成するか、次の既存のセキュリティグループから選択することができます。Amazon EC2 セキュリティグループに関する [詳細はこちら](#)。

セキュリティグループの割り当て: 新しいセキュリティグループを作成する **1**
 既存のセキュリティグループを選択する

セキュリティグループ名: **2**
 説明:

タイプ	プロトコル	ポート範囲	ソース	説明
SSH	TCP	22	カスタム 0.0.0.0/0	例: SSH for Admin Desktop
HTTP	TCP	80	カスタム 0.0.0.0/0	例: SSH for Admin Desktop
HTTPS	TCP	443	カスタム 0.0.0.0/0	例: SSH for Admin Desktop

ルールの追加 **3**

「新しいセキュリティグループを作成する」を選択します。

- 「新しいセキュリティグループを作成する」を選択します。
- セキュリティグループ名は **web-ユーザー名**としてください。説明にも同じ値を入力します。
例) web-user1
- タイプが「HTTPS」のルールを削除する。

1. AMI の選択 2. インスタンスタイプの選択 3. インスタンスの設定 4. ストレージの追加 5. タグの追加 6. セキュリティグループの設定 7. 確認

ステップ 6: セキュリティグループの設定

セキュリティグループは、インスタンスのトラフィックを制御するファイアウォールのルールセットです。このページで、特定のトラフィックに対してインスタンスへの到達を許可するルールを追加できます。たとえば、ウェブサーバーをセットアップして、インターネットトラフィックにインスタンスへの到達を許可する場合、HTTP および HTTPS ポートに無制限のアクセス権限を与えます。新しいセキュリティグループを作成するか、次の既存のセキュリティグループから選択することができます。Amazon EC2 セキュリティグループに関する [詳細はこちら](#)。

セキュリティグループの割り当て: 新しいセキュリティグループを作成する
 既存のセキュリティグループを選択する

セキュリティグループ名:
 説明:

タイプ	プロトコル	ポート範囲	ソース	説明
SSH	TCP	22	カスタム 0.0.0.0/0	例: SSH for Admin Desktop
HTTP	TCP	80	カスタム 0.0.0.0/0	例: SSH for Admin Desktop

ルールの追加



警告

送信元が 0.0.0.0/0 のルールを指定すると、すべての IP アドレスからインスタンスにアクセスすることが許可されます。セキュリティグループのルールを設定して、既知の IP アドレスからのみアクセスできるようにすることをお勧めします。

キャンセル 戻る **確認と作成** **1**

ルールタイプがsshとhttpの2つ設定されていることを確認し、確認と作成ボタンをクリックします。「新しいセキュリティグループを作成する」を選択します。

1. 「確認と作成」をクリックする。

1. AMI の選択 2. インスタンスタイプの選択 3. インスタンスの設定 4. ストレージの追加 5. タグの追加 6. セキュリティグループの設定 7. 確認

ステップ 7: インスタンス作成の確認

インスタンスの作成に関する詳細を確認してください。各セクションの変更に戻ることができます。[作成] をクリックして、インスタンスにキーペアを割り当て、作成処理を完了します。

⚠ インスタンスのセキュリティを強化してください。セキュリティグループ web-user1 は世界に向けて開かれています。
このインスタンスには、どの IP アドレスからもアクセスできる可能性があります。セキュリティグループのルールを更新して、既知の IP アドレスからのみアクセスできるようにすることをお勧めします。
また、セキュリティグループの追加ポートを覆って、実行中のアプリケーションやサービスへのアクセスを容易にすることもできます。たとえば、ウェブサーバー用に HTTP (80) を開きます。 [セキュリティグループの編集](#)

⚠ お客様のインスタンス設定は無料利用枠の対象ではありません
無料利用枠の対象であるインスタンスを起動するには、選択している AMI、インスタンスタイプ、設定オプション、ストレージデバイスをチェックします。 [無料利用枠の利用枠と使用制限に関する詳細情報をご覧ください](#)

当時はこの表示をしない

▼ AMI の詳細 [AMI の編集](#)



Redmine Certified by Bitnami

This image may not be the latest version available and might include security vulnerabilities. Please check the latest, up-to-date, available version at <https://bitnami.com/stacks>.

無料利用
枠の対象

ルートデバイスタイプ: ebs 仮想化タイプ: hvm

ソフトウェアの時間料金: \$0.00 1 時間あたり 上 t3.small インスタンス 追加の税金または料金が適用される場合があります。
この AMI から作成するとソフトウェアの課金が始まり、インスタンスを終了するまで続きます。

この製品を起動することで、このソフトウェアにサブスクリプションし、このソフトウェアの使用により料金表条件と販売者の条件に従うことに同意するものとします。
[エンドユーザーライセンス契約](#)

▼ インスタンスタイプ [インスタンスタイプの編集](#)

[キャンセル](#)
[戻る](#)
起動

画面を下にスクロールさせて設定内容を確認してから作成します。

1. 「起動」をクリックします。

ステップ 1-3-3: キーペアを選択する

既存のキーペアを選択するか、新しいキーペアを作成します。 ×

キーペアは、AWS が保存するパブリックキーとユーザーが保存するプライベートキーファイルで構成されます。組み合わせて使用することで、インスタンスに安全に接続できます。Windows AMI の場合、プライベートキーファイルは、インスタンスへのログインに使用されるパスワードを取得するために必要です。Linux AMI の場合、プライベートキーファイルを使用してインスタンスに SSH で安全に接続できます。

注: 選択したキーペアは、このインスタンスに対して権限がある一連のキーに追加されます。「パブリック AMI から既存のキーペアを削除する」の詳細情報をご覧ください。

1 新しいキーペアの作成

2 キーペア名
handson-20201221

3 キーペアのダウンロード

4

続行するには、事前にプライベートキーファイル (*.pem ファイル) をダウンロードする必要があります。それを、安全でアクセス可能な場所に保存します。一度作成されたファイルは再度ダウンロードすることはできなくなります。

キャンセル インスタンスの作成

新しいキーペアを作成します。

1. 「新しいキーペアの作成」を選択します。
2. 「handson-今日の日付」と入力します。
例) handson-2021xxxx
3. 「キーペアのダウンロード」をクリックして、ファイルをダウンロードします。ダウンロードしたファイルは今後の作業で使うため大事に保管します。
4. 「インスタンスの作成」をクリックします。

作成ステータス

✔ インスタンスは現在作成中です
次のインスタンスの作成が開始されました。 ID: i-0ffcb0385028bc3a9 作成ログの表示

! 予想請求額の通知を受け取る
請求アラートの作成。AWS 請求書の予想請求額が設定した金額を超えた場合(つまり、無料利用枠を超えた場合)、メール通知を受け取ります。

インスタンスへの接続方法

インスタンスは作成中です。実行中状態になり、使用する準備ができるまでに数分かかります。新しいインスタンスの使用時間は、すぐに始まります。インスタンスを停止または削除するまで続きます。
 [インスタンスの表示] をクリックして、インスタンスのステータスを監視します。インスタンスが一度実行中状態になれば、[インスタンス] 画面から接続できます。インスタンスへの接続方法の [詳細はこちら]。

- ▼ ここには、作業を始めるのに役立つリソースがあります
- Linux インスタンスへの接続方法 • Amazon EC2 ユーザーガイド
 - AWS 無料利用枠の詳細 • Amazon EC2 チェックアウトオーバーレイ

インスタンスの作成中、次のことも行うことができます

- ステータスチェックシステムの作成。これらのインスタンスがステータスチェックに合格しなかったときは通知が送信されます。(追加料金が適用される場合があります)
- 追加の EBS ボリュームを作成してアタッチする。(追加料金が適用される場合があります)
- セキュリティグループの管理

①

インスタンスの表示

EC2 インスタンスが作成されました。

1. 「インスタンスの表示」をクリックします。

ステップ 1-3-4: 作成した EC2 インスタンスを確認

The screenshot shows the AWS Management Console interface for EC2 instances. The main content area displays a table with the following data:

Name	インスタンス ID	インスタンスタイプ	アベイラビリティゾーン	インスタンスのステータス	ステータスチェック
webserv1-user1	i-0ffcb0385028bc3a9	t2.micro	ap-northeast-1a	pending	2/2 のチェックに合格しました

Annotations in the image include:

- A red box around the 'インスタンスの表示' button in the top navigation bar, labeled with a circled '1'.
- A red box around the 'pending' status and the 'ステータスチェック' column, which shows '2/2 のチェックに合格しました'.
- A red arrow pointing from the 'pending' status to the '作成完了' (Completed) label.
- A red box around the '作成中' (In Progress) label at the top right.

ユーザー名等で絞込を行うと便利です。インスタンス作成完了には数分かかります。

1. ユーザー名を入れてリターンを押すことで表示を絞り込むことができます。

例)user1

▼フェーズ 1-4: Elastic IP (固定 IP) の割り当て

「サービス」→「ec2」の画面を表示します。

ステップ 1-4-1: Elastic IP (EIP) を取得



1. 「Elastic IP」をクリックします。
2. 「Elastic IP アドレスの割り当て」をクリックします。

Elastic IP アドレスの割り当て

パブリック IPv4 アドレスプールから Elastic IP アドレスを割り当てるか、AWS Global Accelerator からのグローバル IP アドレスを使用します。実行中のインスタンスに 1 つの Elastic IP を無料で関連付けることができます。インスタンスに関連付けられている 2 つ目以上の Elastic IP、停止したインスタンスやアタッチされていないネットワークインターフェイスに関連付けられている Elastic IP、および関連付けられていない Elastic IP に対して料金が発生します。 [詳細はこちら](#)

Elastic IP アドレスの設定

ネットワークボーダーグループ

ネットワークボーダーグループは、パブリック IPv4 アドレスがアドバタイズされるゾーンの論理グループです。このパラメータを設定して、IPv4 アドレスを Network Border Group のゾーンに制限します。



パブリック IPv4 アドレスプール

パブリック IP アドレスは、Amazon のパブリック IP アドレスのプール、顧客が所有してアカウントに持ち込むプール、または顧客が所有して引き続きアドバタイズするプールから割り当てられます。

- Amazon の IPv4 アドレスプール
- AWS アカウントに持ち込むパブリック IPv4 アドレス (プールが見つからなかったためにオプションが無効化されています) [詳細はこちら](#)
- 顧客所有の IPv4 アドレスのプール (顧客所有のプールが見つからないため、オプションは無効です) [詳細はこちら](#)

グローバル静的 IP アドレス

AWS Global Accelerator は、AWS エッジロケーションからのエニーキャストを使用して世界中で発表されたグローバル静的 IP アドレスを提供するため、Amazon のグローバルネットワークを使用することで、ユーザートラフィックの可用性を向上しレイテンシーを低減させることができます。 [詳細はこちら](#)

1

1. 「割り当て」をクリックします。

ステップ 1-4-2: Elastic IP (EIP) をインスタンスに紐付け



EC2 > Elastic IP アドレス > 18.176.78.173

Elastic IP アドレス (1) アクション ▼ Elastic IP アドレスの割り当て

Elastic IP アドレスは、ユーザーが AWS アカウントに割り当てる静的なパブリック IPv4 アドレスで、インターネットからアクセスできます。 [詳細はこちら](#)

パブリック IPv4 アドレス: 18.176.78.173 フィルターをクリアする

<input checked="" type="checkbox"/>	Name	パブリック IPv4 アドレス	割り当て ID	関連付けられたインスタンス	フリ
<input checked="" type="checkbox"/>		18.176.78.173	eipalloc-0e910a976edc5e106	-	-

先ほど割り当てられたEIPをインスタンスに関連付けます。

1. 「このElastic IPアドレスを関連付ける」をクリックします。

EC2 > Elastic IP アドレス > Elastic IP アドレスの関連付け

Elastic IP アドレスの関連付け

この Elastic IP アドレスに関連付けるインスタンスまたはネットワークインターフェイスを選択します (18.178.232.102)

Elastic IP アドレス: 18.178.232.102

リソースタイプ
Elastic IP アドレスに関連付けるリソースのタイプを選択します。

インスタンス
 ネットワークインターフェイス

⚠ すでに Elastic IP アドレスが関連付けられているインスタンスに Elastic IP アドレスを関連付けると、前に関連付けられていた Elastic IP アドレスの関連付けが解除されますが、アカウントへの割り当ては維持されます。詳細はこちら。

インスタンス 1

Q user1 X [refresh]

i-0e57c47d806af0edd (webserver#1-user1) - running

Elastic IP アドレスに関連付けるプライベート IP アドレスです。

Q プライベート IP アドレスを選択します

再関連付け
Elastic IP アドレスがすでにリソースに関連付けられている場合に、そのアドレスを別のリソースに再度関連付けることができるかどうかを指定します。

Elastic IP アドレスの再関連付けを許可する

キャンセル 2 関連付ける

取得した EIP を EC2 インスタンスに紐付けます。フェーズ1-3 で作成した EC2 インスタンスを選択してください。

1. クリックすると候補が表示されます 自分の名前(例. user1) 等を入力しフェーズ1-3 で作成した EC2 インスタンスを選択してください。
例) [webserver#1-user1]等
2. 「関連付ける」をクリックします。

Elastic IP アドレス (1/1)

検索: Elastic IP アドレスをフィルタリング

パブリック IPv4 アドレス: 13.112.104.143

Name	割り当てられた IPv...	タイプ	割り当て ID
-	13.112.104.143	パブリック IP	eipalloc-06596884b5257d5bb

13.112.104.143

概要 | タグ

概要

割り当てられた IPv4 アドレス	タイプ	割り当て ID	アソシエーション ID
13.112.104.143	パブリック IP	eipalloc-06596884b5257d5bb	eipassoc-07f9eca91dc7190c9
スコープ	関連付けられたインスタンス ID	プライベート IP アドレス	ネットワークインターフェイス ID
VPC	i-08684383ac0a66b83	10.0.0.243	eni-0db437026c3a30661
ネットワークインターフェイス所有者のアカウント ID	パブリック DNS	NAT ゲートウェイ ID	アドレスプール
560620688551	ec2-13-112-104-143.ap-northeast-1.compute.amazonaws.com	-	Amazon
ネットワークボーダーグループ			
ap-northeast-1			

正しくEC2 インスタンス紐付けられていることを確認します。EIP は後で使用するため、メモしておきます。

1. 関連付けられたインスタンスIDをクリックし、フェーズ1-3 で作成した EC2 インスタンス画面に移動することを確認します。

ステップ 1-4-3: Elastic IPアドレスをメモ

The screenshot shows the AWS Management Console interface for an EC2 instance. The instance name is 'webserver#1-user1' and its ID is 'i-08684383ac0a66b83'. The instance is in the '実行中' (Running) state. The 'Elastic IP Address' field is highlighted with a red box and a circled '1', indicating the address to be noted: 13.112.104.143 [パブリック IP].

Name	インスタンス ID	インスタンス...	インスタン...	ステータスチェック
webserver#1-user1	i-08684383ac0a66b83	実行中	t3.micro	2/2 のチェックに合格し...

インスタンス: i-08684383ac0a66b83 (webserver#1-user1)

詳細 | セキュリティ | ネットワーキング | ストレージ | ステータスチェック | モニタリング | タグ

▼ インスタンス概要 情報

インスタンス ID	パブリック IPv4 アドレス	プライベート IPv4 アドレス
i-08684383ac0a66b83 (webserver#1-user1)	13.112.104.143 オープンアドレス	10.0.0.243
インスタンスの状態	パブリック IPv4 DNS	プライベート IPv4 DNS
実行中	ec2-13-112-104-143.ap-northeast-1.compute.amazonaws.com オープンアドレス	ip-10-0-0-243.ap-northeast-1.compute.internal
インスタンスタイプ	Elastic IP アドレス	VPC ID
t3.micro	13.112.104.143 [パブリック IP]	vpc-009c18aca5dc628b1 (handson-user1)

1. 作成したインスタンスを選択して、「Elastic IPアドレス」をメモする。

ステップ 1-4-4: Redmineにアクセス

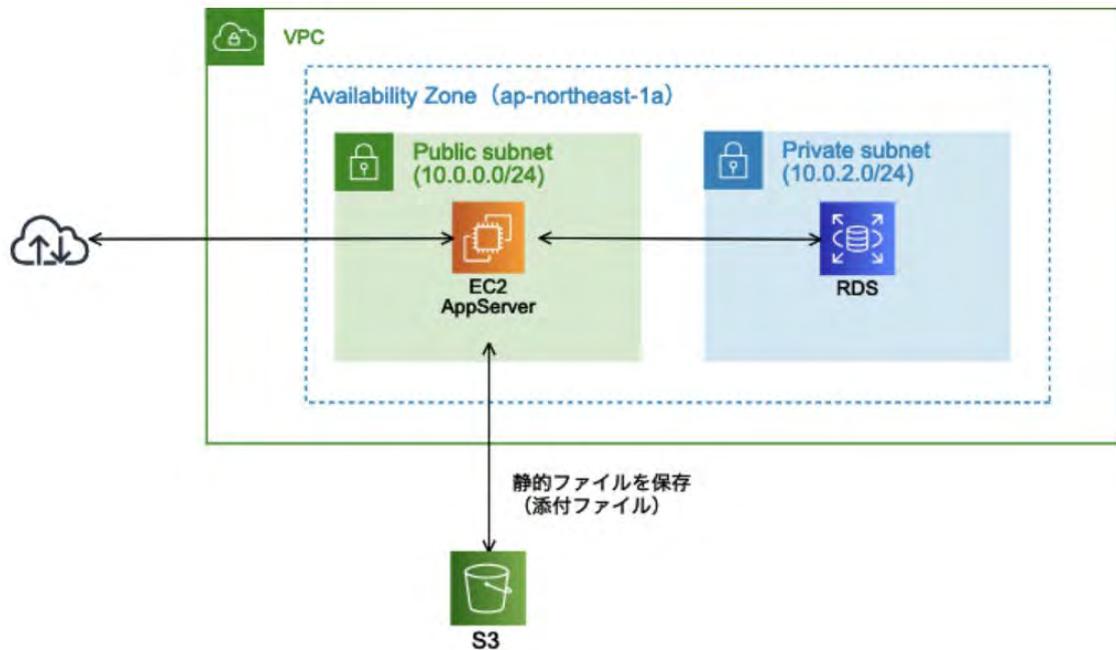


先ほどメモした EIP にアクセスし、redmineが表示されることを確認します。

1. ブラウザで<http://<Elastic IPアドレス>/> にアクセスします。
2. redmineが表示されることを確認します。

[フェーズ 2]

~拡張性を向上しつつDB 運用負荷を軽減する構成を構築~



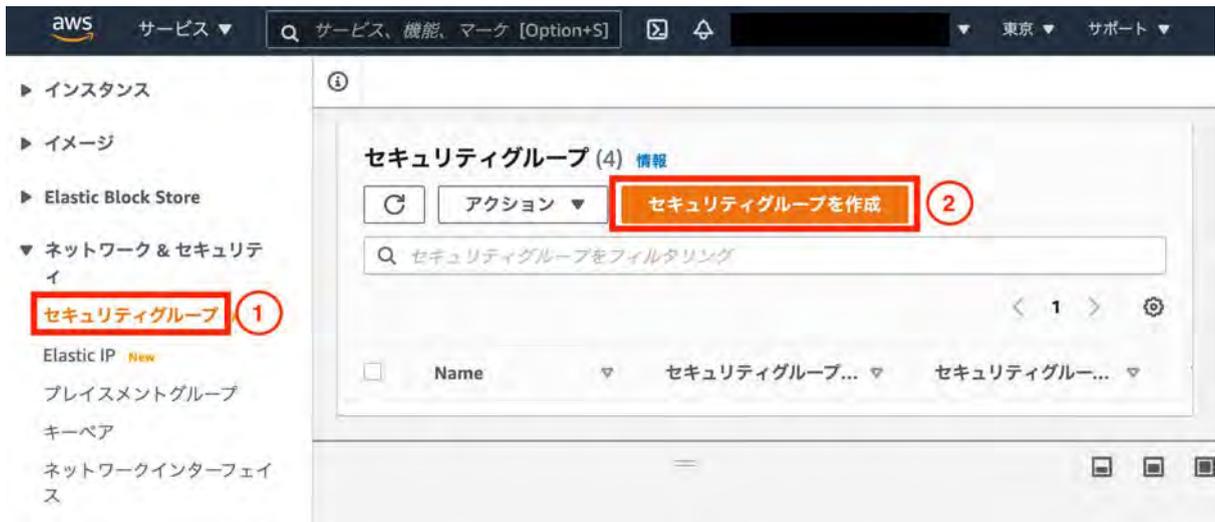
▼フェーズ 2-1: Amazon RDS のセキュリティグループを作成

ステップ 2-1-1: DB 用セキュリティグループを作成



1. 「ec2」を入力します。

2. 「EC2」をクリックします。



1. 「セキュリティグループ」をクリックします。
2. 「セキュリティグループの作成」をクリックします。



1. 「db-ユーザー名」を入力します。例)db-user1

2. 「RDS for MySQL」など説明を入力します。
3. フェーズ1-1-5 で作成したVPC を選択してください。 例)handson-user1 を選択
4. 「ルールを追加」をクリックします。
※ インバウンドルールであることを確認

1. 「MySQL/Aurora」を選択します。
2. 「カスタム」を選択します。
3. 「Web」と入力して候補を表示させます。
Web と入力しても補完されない場合には、該当するセキュリティグループの ID (sg-xxxxxx) を入力します。
4. 「候補」をクリックします。

5. 「セキュリティグループを作成」をクリックします。

▼フェーズ 2-2: DB サブネットグループを作成

ステップ2-2-1: Amazon RDS 管理ページを開く



1. 「rds」を入力します。
2. 「RDS」をクリックします。

ステップ 2-2-2: DB サブネットグループを作成



プライベートサブネット内に DB サブネットグループを作成します。

1. 「サブネットグループ」をクリックします。
2. 「DB サブネットグループの作成」をクリックします。

RDS > サブネットグループ > Create DB subnet group

DB サブネットグループを作成

新しいサブネットグループを作成するには、名前と説明を入力し、既存の VPC を選択します。その後、その VPC に関連するサブネットを追加できます。

サブネットグループの詳細

名前
サブネットグループの作成後に名前を変更することはできません。

db subnet user1 **1**

1~255 文字にする必要があります。英数字、スペース、ハイフン、アンダースコア、ピリオドを使用できます。

説明

RDS for MySQL **2**

VPC
DB サブネットグループに使用するサブネットに対応する VPC 識別子を選択します。サブネットグループが作成された後、別の VPC 識別子を選択することはできません。

handson-user1 (vpc-0edcc812ddfb7e67e) **3**

1. 「db subnet ユーザー名」を入力します。 例) db subnet user1
2. 「RDS for MySQL」などを入力します。
3. フェーズ1-1-5 で作成した VPCを選択します。 例)[handson-user1]

サブネットを追加

アベイラビリティゾーン
追加するサブネットを含むアベイラビリティゾーンを選択します。

1

アベイラビリティゾーンを選択

ap-northeast-1a × ap-northeast-1c ×

サブネット
追加するサブネットを選択します。リストには、選択したアベイラビリティゾーンのサブネットが含まれます。

2

サブネットを選択

subnet-059a79fa8c3ccbd5b (10.0.2.0/24) ×
subnet-0c2f8a0ca84f2f452 (10.0.3.0/24) ×

選択したサブネット (2)

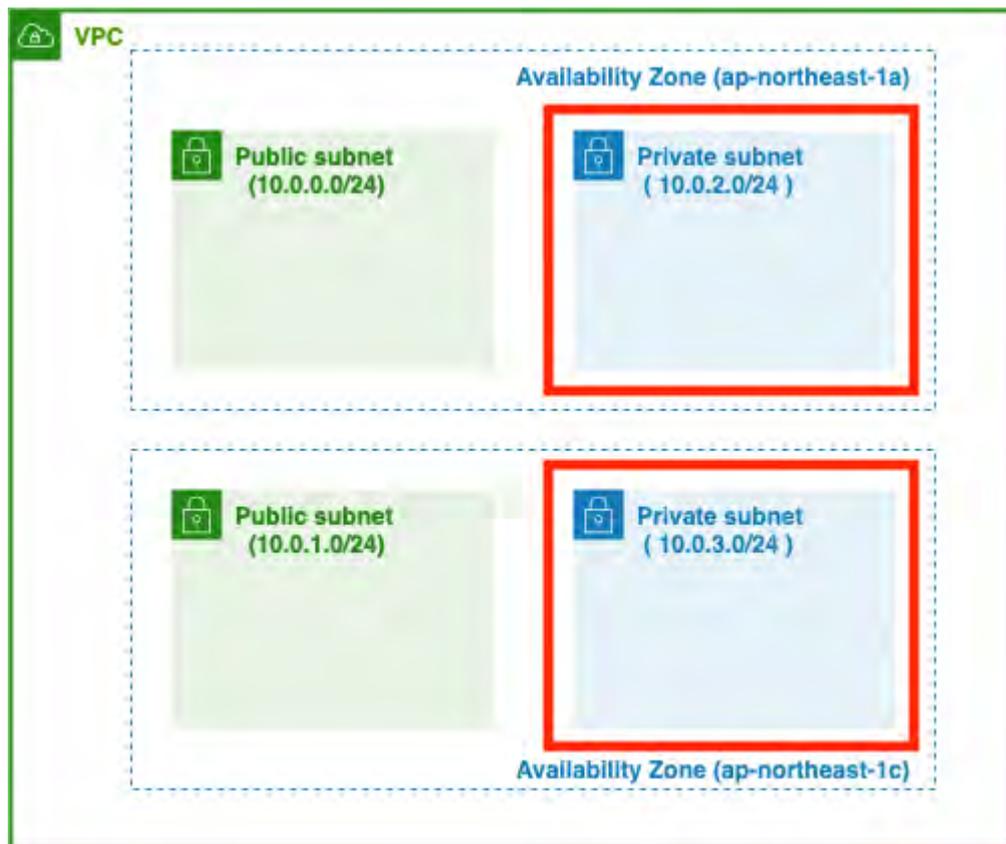
アベイラビリティゾーン	サブネット ID	CIDR ブロック
ap-northeast-1a	subnet-059a79fa8c3ccbd5b	10.0.2.0/24
ap-northeast-1c	subnet-0c2f8a0ca84f2f452	10.0.3.0/24

3

キャンセル 作成

ap-northeast-1a のプライベートサブネット (10.0.2.0/24) と ap-northeast-1c のプライベートサブネット (10.0.3.0/24) を追加します。

1. 「ap-northeast-1a」、「ap-northeast-1c」を選択します。
2. 「プライベートサブネット(10.0.2.0/24)」、「プライベートサブネット(10.0.3.0/24)」を選択します。
3. 「作成」をクリックします。



RDS > サブネットグループ

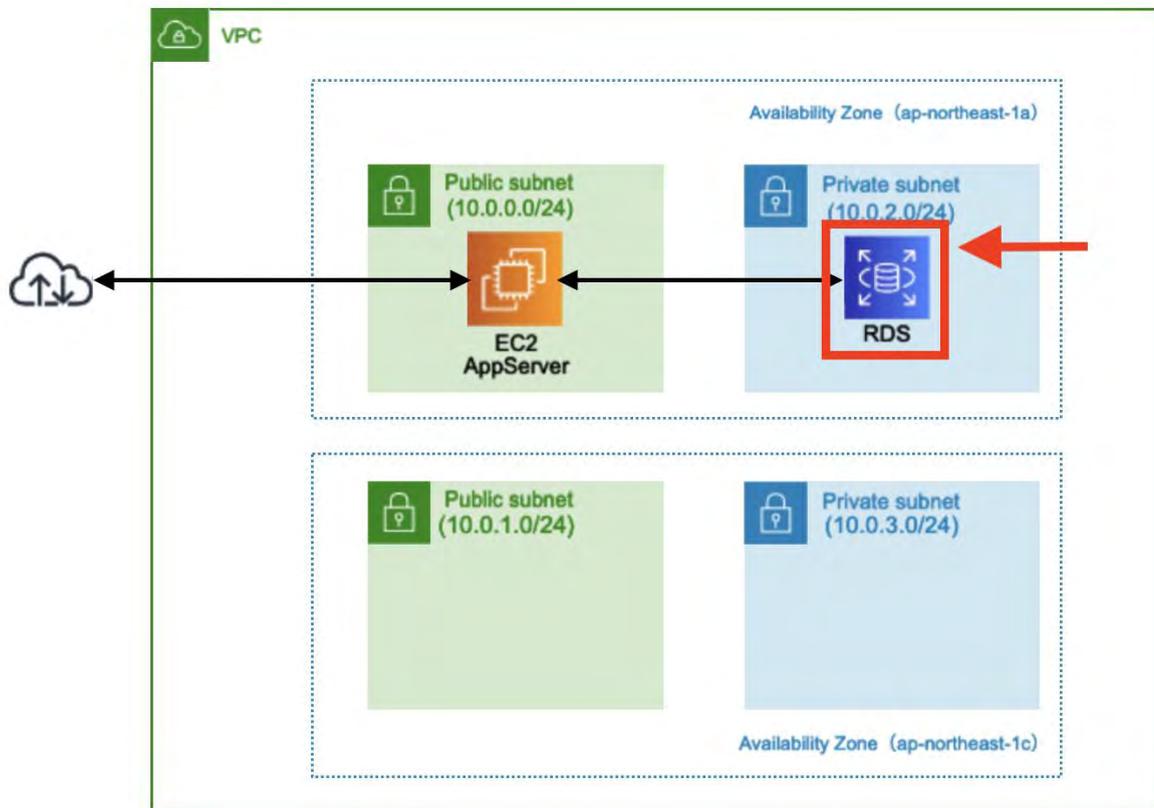
サブネットグループ (2) リフレッシュ 編集 削除 DB サブネットグループの作成

フィルタ サブネットグループ < 1 > 設定

<input type="checkbox"/>	名前	▲ 説明	▼ ステータス ▼
<input type="checkbox"/>	db subnet user1	db subnet user1	完了

DBサブネットが作成されました。

▼フェーズ 2-3: Amazon RDS インスタンスを作成



ステップ 2-3-1: データベースの作成

The screenshot shows the AWS Management Console interface for Amazon RDS. On the left, a navigation menu lists various options, with 'ダッシュボード' (Dashboard) highlighted in a red box and a circled '1'. The main content area is titled 'リソース' (Resources) and includes a '更新' (Refresh) button. Below this, it states 'Asia Pacific (Tokyo) リージョンで、以下の Amazon RDS リソースを使用します (使用した量/クォータ)'. A table lists various resources and their counts: DB インスタンス (1/40), ストレージ割り当て (0 パイ ト/100.00 TB), DB インスタンス上限を引き上げるには、こちらをクリックしてください, リザーブドインスタンス (0/40), スナップショット (60), 手動 (7/100), 自動 (0), 最近のイベント (5), イベントサブスクリプション (0/20), パラメータグループ (9), デフォルト (6), カスタム (3/100), オプショングループ (5), デフォルト (5), カスタム (0/20), サブネットグループ (2/50), サポートされているプラットフォーム VPC, デフォルトネットワーク vpc-775b0510. Below the resource list, there is a section titled 'データベースの作成' (Create Database) with a brief description of Amazon RDS. At the bottom of this section, there are two buttons: 'S3 から復元' (Restore from S3) and 'データベースの作成' (Create Database), with the latter highlighted in a red box and a circled '2'. A note at the bottom states '注: DB インスタンスは以下で作成されます Asia Pacific (Tokyo) リージョン'.

1. 「ダッシュボード」をクリックします。
2. 「データベースの作成」をクリックします。

データベースの作成

データベース作成方法を選択 情報

標準作成

可用性、セキュリティ、バックアップ、メンテナンスといったすべての設定オプションを設定します。

簡単作成

推奨されるベストプラクティス設定を使用します。一部の設定オプションは、データベースの作成後に変更できません。

エンジンのオプション

エンジンのタイプ 情報

Amazon Aurora



MySQL



MariaDB



PostgreSQL



Oracle



Microsoft SQL Server



エディション

MySQL Community

バージョン 情報

MySQL 5.7.22

[エンジンのオプション]

1. 「MySQL」を選択します。

テンプレート

お客様のユースケースに合わせてサンプルテンプレートを選択します。

1

<p><input type="radio"/> 本番稼働用 高い可用性と、高速で安定したパフォーマンスのためには、デフォルト値を使用します。</p>	<p><input checked="" type="radio"/> 開発/テスト このインスタンスは本番稼働環境ではない開発で使用します。</p>	<p><input type="radio"/> 無料利用枠 RDS 無料利用枠を利用すると、新しいアプリケーションの開発、既存のアプリケーションのテスト、Amazon RDS の実践経験の蓄積が可能です。 情報</p>
--	--	---

設定

DB インスタンス識別子 [情報](#)

DB インスタンスの名前を入力します。この名前は、AWS アカウントが現在の AWS リージョンで所有しているすべての DB インスタンスにおいて一意である必要があります。

redmine-user1

2

DB インスタンス識別子は、大文字と小文字を区別しませんが、すべて小文字で保存されます (例: "mydbinstance")。制約として、使用できるのは 1~60 文字以内で英数字またはハイフンのみです (SQL Server は 1~15 文字)。1 文字目は英文字でなければなりません。また、ハイフンを連続で 2 つ使ったり、最後の文字をハイフンにしたりすることはできません。

▼ 認証情報の設定

マスターユーザー名 [情報](#)

DB インスタンスのマスターユーザーのログイン ID を入力します。

admin

3

1~16 文字の英数字。1 文字目は文字である必要があります

パスワードの自動生成

Amazon RDS がパスワードを生成するか、お客様がご自身でパスワードを指定することができます

マスターパスワード [情報](#)

.....

4

制約事項: 表示可能な ASCII 文字で 8 文字以上で入力してください。次の文字を含めることはできません: / (スラッシュ)、" (二重引用符)、および @ (アットマーク)。

パスワードを確認 [情報](#)

.....

5

[テンプレート]

1. 「開発/テスト」を選択します。

[設定]

DB インスタンス識別子とパスワードは、**redmine-自分の名前**とします。

2. 「redmine-自分の名前」と入力します。例) redmine-user1

3. 「admin」と入力します。
4. admin のパスワード「redmine-xxxx」(xxxxはユーザー名など任意の文字列)を入力します。
例)redmine-user1
5. 再度パスワードを入力します。

DB インスタンスサイズ

DB インスタンスクラス [情報](#)

処理能力とメモリの要件に合った DB インスタンスクラスを選択します。以下の DB インスタンスクラスオプションは、上記で選択したエンジンでサポートされているものに制限されます。

標準クラス (m クラスを含む)
 メモリ最適化クラス (r クラスと x クラスを含む)
 バースト可能クラス (t クラスを含む) 1

db.t3.micro ▼ 2

2 vCPUs 1 GiB RAM Network: 2,085 Mbps

i New instance classes are available for specific engine versions. [情報](#)

以前の世代のクラスを含める

[DBインスタンスサイズ]

1. 「バースト可能クラス(tクラスを含む)」をクリックします。
2. 「db.t3.micro」を選択します。

可用性と耐久性

マルチ AZ 配置 [情報](#)

スタンバイインスタンスを作成する (本稼働環境向けに推奨)
データの冗長性を提供し、I/O のフリーズを防ぎ、システムバックアップの間のレイテンシーの急上昇を最小限に抑えるために、別のアベイラビリティゾーン (AZ) にスタンバイを作成します。

スタンバイインスタンスを作成しないでください 1

[可用性と耐久性]

3. 「スタンバイインスタンスを作成しないでください」を選択します。

接続 🔄

Virtual Private Cloud (VPC) 情報
この DB インスタンスの仮想ネットワーク環境を定義する VPC。

handson-user1 (vpc-0edcc812ddfb7e67e) 1

対応する DB サブネットグループがある VPC のみが表示されます。

i データベースの作成後に、VPC の選択を変更することはできません。

サブネットグループ 情報
選択した VPC で DB インスタンスが使用できるサブネットと IP 範囲を定義する DB サブネットグループ。

db subnet user1 2

パブリックアクセス可能 情報

あり
VPC 外部の Amazon EC2 インスタンスとデバイスがお客様のデータベースに接続できます。データベースに接続できる VPC 内の EC2 インスタンスおよびデバイスを指定する 1 つ以上の VPC セキュリティグループを選択します。

なし 3
RDS はパブリック IP アドレスをデータベースに割り当てません。VPC 内部の Amazon EC2 インスタンスとデバイスのみをお客様のデータベースに接続できます。

[接続]

1. フェーズ1-1-5で作成したVPCを選択します。例)handson-user1
2. 自動的に RDS サブネットグループが選択されます。
3. 「なし」を選択します。

VPC セキュリティグループ
 RDS セキュリティグループを 1 つ以上選択し、データベースへのアクセスを許可します。セキュリティグループのルールで EC2 インスタンスと VPC 外のデバイスからの着信トラフィックが許可されていることを確認します (セキュリティグループはパブリックにアクセス可能なデータベースに必要です)。

既存の選択
 既存の VPC セキュリティグループの選択

新規作成
 新しい VPC セキュリティグループの作成

既存の VPC セキュリティグループ
 VPC セ **4** ティグループを選択します

default **X**

アベイラビリティーゾーン [情報](#)
 ap-northeast-1a **6**

▶ 追加の接続設定

4. 既存のVPCセキュリティグループでdefaultが選択されている場合は、「×」で外します。
5. 「ステップ 1: DB 用セキュリティグループを作成」で作成したセキュリティグループを選択します。例) db-user1
6. 「ap-northeast-1a」を選択します。

データベース認証

データベース認証オプション [情報](#)

パスワード認証
 データベースのパスワードを使用して認証します。

パスワードと IAM データベース認証
 AWS IAM ユーザーとロールを介して、データベースパスワードとユーザー認証情報を使用して認証します。

パスワードと Kerberos 認証 (このバージョンでは使用できません)
 承認されたユーザーに、Kerberos 認証を使ってこの DB インスタンスで認証を行うことを許可するディレクトリを選択します。

1

▶ **追加設定**
 データベースオプション、暗号化が有効、バックアップが有効、バックトラックが無効、拡張モニタリングが有効、メンテナンス、CloudWatch Logs、削除保護が無効

▼ **追加設定** ①

データベースオプション, 暗号化が有効, バックアップが有効, バックトラックが無効, 拡張モニタリングが有効, メンテナンス, CloudWatch Logs, 削除保護が無効

データベースの選択肢

最初のデータベース名 [情報](#)

データベース名を指定しないと、Amazon RDS はデータベースを作成しません。

DB パラメータグループ [情報](#)

default.mysql8.0 ▼

オプショングループ [情報](#)

default:mysql-8-0 ▼

バックアップ

データベースのポイントインタイムスナップショットを作成します

自動バックアップの有効化
バックアップを有効にすると、特定の時間枠でデータベースのバックアップが自動的に作成されます。

⚠ 自動バックアップは現在 InnoDB ストレージエンジンでのみサポートされていることに注意してください。MyISAM を使用している場合、詳細については [こちら](#) を参照してください。

バックアップ保持期間 [情報](#)

このインスタンスの自動バックアップを RDS が保存する日数を選択します。

0日間 ▼ ②

キャンセル **データベースの作成** ③

[追加設定]

1. 「追加設定」はデフォルトでは折りたたまれているため、クリックして設定を表示します。
2. バックアップ保持期間で「0日間」を選択します。
3. 「データベースの作成」をクリックします。

▼フェーズ 2-4: RDSに接続

ステップ 2-4-1: 作成した RDS インスタンスを確認

「サービス」→「RDS」画面を表示します。



1. 「データベース」をクリックします。
2. フェーズ2-3-1で作成した RDS インスタンスをクリックします。

The screenshot shows the Amazon RDS console for an instance named 'redmine-user1'. The left sidebar contains navigation options like 'ダッシュボード', 'データベース', 'Query Editor', etc. The main content area shows the instance details under the '接続とセキュリティ' (Connections and Security) tab. A red circle with the number '1' highlights the 'Endpoint' field, which contains the URL 'redmine-user1.cizpucnnhfj8.ap-northeast-1.rds.amazonaws.com'.

概要			
DB 識別子 redmine-user1	CPU 1.67%	情報 利用可能	クラス db.t2.micro
ロール インスタンス	現在のアクティビティ 0 接続	エンジン MySQL Community	リージョンと AZ ap-northeast-1a

接続とセキュリティ		
エンドポイントとポート エンドポイント redmine-user1.cizpucnnhfj8.ap-northeast-1.rds.amazonaws.com ポート 3306	ネットワーク アベイラビリティゾーン ap-northeast-1a VPC handson-user1 (vpc-0a941f74723ca26f2) サブネットグループ db subnet user1 サブネット subnet-05dae220b74f2a8c1 subnet-082bafb5ee8ec3a86	セキュリティ VPC セキュリティグループ db-user1 (sg-099eb01756122c893) (アクティブ) パブリックアクセスセキュリティ なし 認証機関 rds-ca-2019 証明機関の日付 Aug 23rd, 2024

RDS の各インスタンスにはエンドポイント (Endpoint) と呼ばれるホスト名が設定されます。エンドポイントをメモします。

表示されない場合は画面をリロードしてください。

※ 作成されるまで時間がかかります

1. エンドポイントをメモします。

ステップ 2-4-2: database.yml をバックアップ

作成したインスタンスに接続します。

今回はsshで接続するため手元のターミナルを開きます。

1. EC2インスタンス作成時にダウンロードした「プライベートキーファイル(handson-2021xxxx.pem)」があるか確認する

```
ls ~/Downloads/handson-2021xxxx.pem
```

2. プライベートキーファイル(handson-2021xxxx.pem)をわかりやすいところに移動させる

```
mv ~/Downloads/handson-2021xxxx.pem [ディレクトリ名]/handson-2021xxxx.pem
```

3. 以下のコマンドを実行して権限を変更する

```
cd [ディレクトリ]  
chmod 400 handson-2021xxxx.pem
```

4. インスタンス作成後にメモしたEIPを使用してインスタンスに接続する
(yes と入力)

```
ssh -i "handson-2021xxxx.pem" bitnami@[ webserver#1のElastic IPアドレス ]
```

接続ができれば、以下のコマンドを実行してMySQLのdatabase.ymlをバックアップする。

```
# root  
sudo su  
# redmineのディレクトリに移動  
cd /opt/bitnami/apps/redmine/htdocs/  
# database.ymlをバックアップ  
cp config/database.yml config/database_bk.yml
```

ステップ 2-4-3: RDSに接続

引き続きターミナルで作業します。

以下のコマンドを実行してdatabase.ymlの以下の箇所を編集します。

```
# database.ymlを編集
vi config/database.yml
```

```
production:
  adapter: mysql2
  database: rds_redmine
  host: [メモしたRDS Endpoint]
  username: admin
  password: [redmine-自分の名前(例:redmine-user1)]
  encoding: utf8
```

以下のコマンドを実行します。
databaseを作成、マイグレーションをします。

```
#databaseを作成
bundle exec rake db:create RAILS_ENV=production
#マイグレーション
bundle exec rake db:migrate RAILS_ENV=production
```

設定が終了したらApacheを再起動して設定を反映させます。

```
#apacheの停止
/opt/bitnami/apache2/scripts/ctl.sh stop
#apacheの起動
```

```
/opt/bitnami/apache2/scripts/ctl.sh start  
#apacheのステータス確認  
/opt/bitnami/apache2/scripts/ctl.sh status
```

以下のコマンドを実行し、mysqlを停止します。
mysql停止後もredmineにアクセスできることを確認してください。

```
/opt/bitnami/mysql/scripts/ctl.sh stop
```

▼フェーズ 2-5: Redmine S3対応

ステップ 2-5-1: Redmineにダミーデータを登録

引き続きインスタンスに接続してコマンドを実行し、ダミーデータを登録します。
以下のコマンドを実行する事でredmineにダミーのデータが登録され、動作検証がスムーズに行えます。

1. 以下のコマンドを実行します。

```
RAILS_ENV=production bundle exec rake db:fixtures:load
```

ステップ 2-5-2: Redmineにファイルをアップロード

ブラウザで<http://<Elastic IPアドレス>/> にアクセスしてredmineを表示し、ファイルをアップロードします。



1. ブラウザで<http://<Elastic IPアドレス>> にアクセスしてredmineを表示します。
2. Redmine画面右上の「ログイン」をクリックします。



ログインID
admin 1

パスワード パスワードの再設定
..... 2

ログイン 3

先ほどfixtures:loadを実行しダミーのユーザが作成されているため、adminでログインします。

1. ログインIDに「admin」を入力します。
2. パスワードに「admin」を入力します。
3. ログインをクリックします。



Homeが表示されたらログイン成功です。

Home My page **Projects** Administration Help Logged in as admin My account Sign out

Redmine 1 Search: Jump to a project...

Projects Activity Issues Spent time Gantt Calendar News

Projects New project Administration

Filters

Status is active Add filter

Options

Apply Clear Save

eCookbook 2
Recipes management application
Private child of eCookbook
This is a private subproject of a public project
Child of private child
This is a public subproject of a private project
eCookbook Subproject 1
eCookbook Subproject 1
eCookbook Subproject 2
eCookbook Subproject 2

OnlineStore
E-commerce web site

My projects
Also available in: Atom

Redmineにログイン後、以下の手順でファイルをアップロードします。

1. 「projects」をクリックします。
2. 「eCookbook」をクリックします。

Home My page Projects Administration Help Logged in as admin My account Sign out

eCookbook Search: eCookbook

+ Overview Activity Roadmap Issues Spent time Gantt Calendar News Documents Wiki Forums **Files** Repository Settings

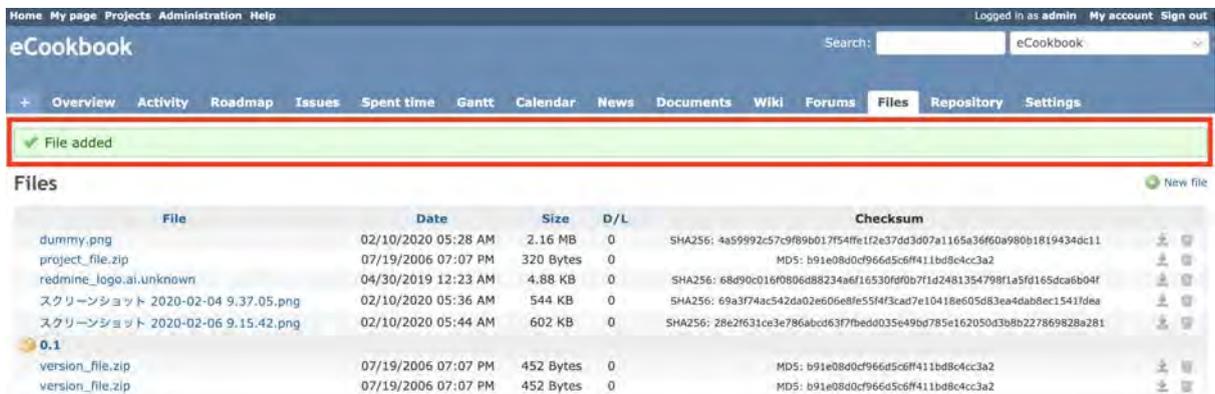
Files New file

File	Date	Size	D/L	Checksum
dummy.png	02/10/2020 05:28 AM	2.16 MB	0	SHA256: 4a59992c57c9f89b017f54ffe1f2e37dd3d07a1165a36f60a980b1819434dc11
project_file.zip	07/19/2006 07:07 PM	320 Bytes	0	MDS: b91e08d0cf966d5c6ff411bd804cc3a2
redmine_logo.ai.unknown	04/30/2019 12:23 AM	4.88 KB	0	SHA256: 68d90c0160806d88234a6f16530fd0b7f1d2481354798f1a5fd165dca6b04f
スクリーンショット 2020-02-04 9:37:05.png	02/10/2020 05:36 AM	544 KB	0	SHA256: 69a3f74ac542da02e606e8fe55f4f3cad7e10418e605d83ea4dab8ec1541fdea
0.1				
version_file.zip	07/19/2006 07:07 PM	452 Bytes	0	MDS: b91e08d0cf966d5c6ff411bd804cc3a2
version_file.zip	07/19/2006 07:07 PM	452 Bytes	0	MDS: b91e08d0cf966d5c6ff411bd804cc3a2

1. 「Files」をクリックします。
2. 「New file」をクリックします。



1. 「ファイルを選択」をクリックして、アップロードするファイルを選択します。
2. 「Add」をクリックします。



redmineの画面に「File added」というアラートが表示されたら、ファイルのアップロードができています。

ステップ 2-5-3: ファイルのアップロード確認

ファイルがアップロードされているかをec2の中で確認します。

以下のコマンドを実行して、先ほどアップロードしたファイルがあることを確認します。

```
cd /opt/bitnami/apps/redmine/htdocs
# ファイルがあることを確認
ls files/20xx(年)/xx(月)/
→ xxxxxxxx.png などと表示されればOK
```

ステップ 2-5-4: S3アクセス用のユーザーの作成



サービス検索窓でIAMを検索し選択します。

1. 「iam」を入力します。
2. 「IAM」をクリックします。



「ユーザーを追加」画面へ移動します。

1. 「ユーザー」をクリックします。

2. 「ユーザーを追加」をクリックします。

ユーザーを追加

1 2 3 4 5

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* 1
[別のユーザーの追加](#)

AWS アクセスの種類を選択

これらのユーザーから AWS にアクセスする方法を選択します。アクセスキーと自動生成パスワードは前のステップで提供されています。 [詳細はこちら](#)

- アクセスの種類* **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。 2
- 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。

3

キャンセル

次のステップ: アクセス権限

1. ユーザ名に「**s3access-2021xxxx**」と入力します。
2. 「プログラムによるアクセス」にチェックを入れます。
3. 「次のステップ: アクセス権限」をクリックします。

ユーザーを追加

1 2 3 4 5

▼ アクセス許可の設定

1

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 Q AmazonS3FullAccess 2 1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"/> AmazonS3FullAccess	AWS による管理	Permissions policy (9)

3

キャンセル 戻る 4 次のステップ: タグ

1. 「既存のポリシーを直接アタッチ」を選択します。
2. ポリシーのフィルタで「AmazonS3FullAccess」と入力して検索します。
3. 表示された「AmazonS3FullAccess」ポリシーにチェックを入れます。
4. 「次のステップ: タグ」をクリックします。

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
Name	iam-user1	✕
新しいキーを追加		

さらに 49 個のタグを追加できます。

3

キャンセル 戻る 次のステップ: 確認

1. キーに「Name」を入力します。
2. 値に「iam-ユーザー名」を入力します。例) iam-user1
3. 「次のステップ: 確認」をクリックします。

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	s3access-20200228
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	AmazonS3FullAccess

タグ

新しいユーザーは次のタグを受け取ります

キー	値
Name	iam-user1

キャンセル 戻る **ユーザーの作成**

設定確認 & ユーザーの作成をします。

1. 設定内容を確認し「ユーザーの作成」をクリックします。

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://533384410763.signin.aws.amazon.com/console>」でサインインできます

↓ .csv のダウンロード

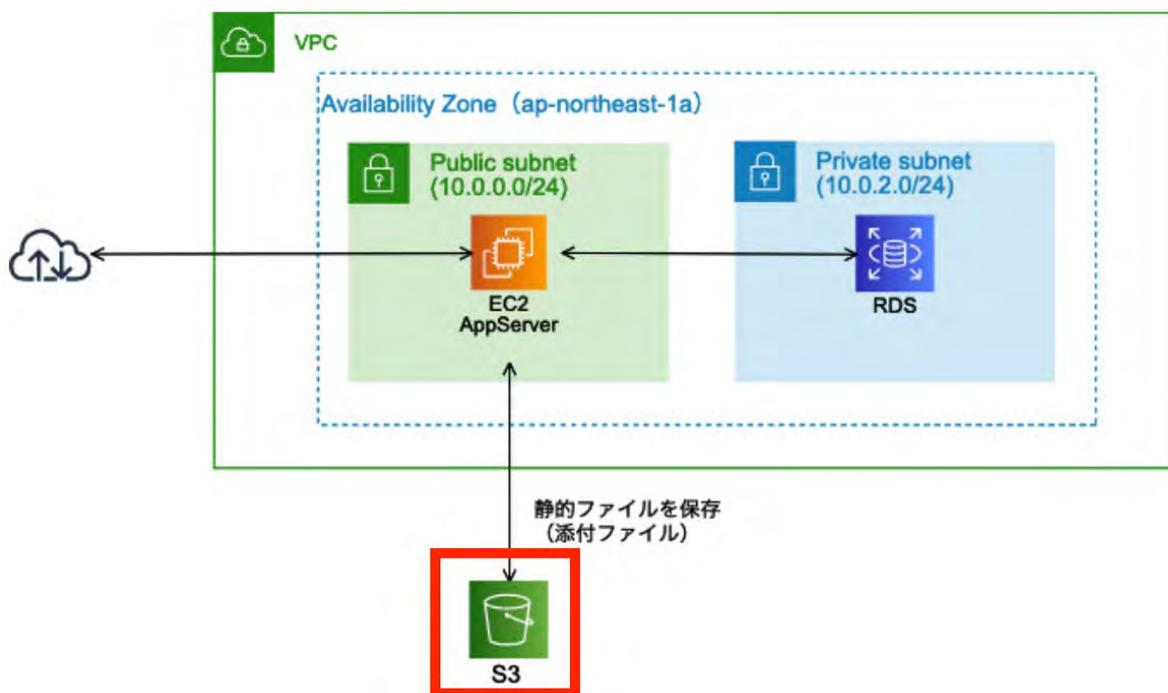
1

ユーザー	アクセスキー ID	シークレットアクセスキー
▶ s3access-20200228	AKIAXYMBQG2FV5OYN2XD	***** 表示

アクセスキーやシークレットアクセスキーが書かれているcsvをダウンロードする。
※ 後ほど使用するので大切に保管すること

1. 「.csvのダウンロード」をクリックします。
2. ダウンロードしたcsvは後ほど使用するため、大切に保管する。

ステップ 2-5-5: S3 バケット作成





サービス検索窓でS3を検索し選択します。

1. 「S3」を入力します。
2. 「S3」をクリックします。



1. 「バケットを作成」をクリックします。

Amazon S3 > バケットを作成

バケットを作成

バケットは S3 に保存されたデータのためのコンテナです。 [詳細](#)

一般的な設定

バケット名
 ①

バケット名は一意である必要があり、スペース、または大文字を含めることはできません。 [バケットの命名規則をご参照ください](#)

リージョン
 ②

既存のバケットから設定をコピー - オプション
次の設定のバケット設定のみがコピーされます。

ブロックパブリックアクセスのバケット設定

パブリックアクセスは、アクセスコントロールリスト (ACL、Access Control List)、バケットポリシー、アクセスポイントポリシー、またはそのすべてを介してバケットとオブジェクトに許可されます。このバケットとそのオブジェクトへの公開アクセスが確実にブロックされるようにするには、[パブリックアクセスをすべてブロック] を有効にします。これらの設定はこのバケットとそのアクセスポイントにのみ適用されます。AWS では [パブリックアクセスをすべてブロック] を有効にすることをお勧めしますが、これらの設定を適用する前に、アプリケーション

③ バケットを作成したら、バケットにファイルとフォルダをアップロードし、追加のバケット設定を行うことができます。 ③

バケットを作成します。バケット名はグローバルで一意である必要があります。
xxxxは自分の名前等を入力し、他と被らないバケット名にしてください。

1. 「redmine-xxxx(自分の名前)-2021xxxx」と入力します。例) redmine-user1-20210210
2. 「アジアパシフィック(東京)」を選択します。
3. 「作成」をクリックします。

ステップ 2-5-6: S3プラグインの導入

ssh接続した状態で以下のコマンドを実行します。

```
$ sudo su
# redmineのディレクトリに移動
cd /opt/bitnami/apps/redmine/htdocs/
# Pluginのダウンロード
git clone https://github.com/redmica/redmica_s3.git plugins/redmica_s3
```

```
# Pluginの設定ファイルの作成
cp plugins/redmica_s3/config/s3.yml.example config/s3.yml
vi config/s3.yml
```

s3.ymlファイルを開いて、以下のように設定します。
「**access_key_id**」「**secret_access_key**」は先ほどダウンロードしたCSVの情報を入力します。
また、**bucket**には先ほど作成したバケット名を入力します。例) redmine-user1-2021xxxx

```
production:
  access_key_id: CSVの情報を入力
  secret_access_key: CSVの情報を入力
  bucket: redmine-user1(自分の名前)-2021xxxx
  folder: files
  thumb_folder: tmp/thumbnails
  import_folder: tmp/imports
  region: ap-northeast-1
```

```
# 所有者の変更
chown -R bitnami:daemon plugins/redmica_s3
chown -R bitnami:daemon config/s3.yml
```

```
# Gemfileに以下の1行を追記
echo gem 'zip', '2.0.2' >> Gemfile.local
# 必要ライブラリーのインストール
bundle install --no-deployment
bundle exec rake redmine:plugins RAILS_ENV=production
```

ステップ 2-5-7: Apacheの再起動&設定を反映

2. 以下のコマンドでapacheを再起動し、設定を反映させます。

```
# apacheの停止
/opt/bitnami/apache2/scripts/ctl.sh stop

# apacheの起動
/opt/bitnami/apache2/scripts/ctl.sh start
```

3. 以下のコマンドでapacheのstatusを確認します。
「**apache already running**」と表示されることを確認します。

```
# apacheのステータスを確認
/opt/bitnami/apache2/scripts/ctl.sh status
```

ステップ 2-5-8: 再度Redmineにファイルをアップロード

redmineに再度ファイルをアップロードします。
今回アップロードしたファイルはS3にも保存されます。

Home My page **Projects** Administration Help

Logged in as admin My account Sign out

Redmine 1 Search: Jump to a project...

Projects Activity Issues Spent time Gantt Calendar News

Projects New project Administration

Filters

Status is active Add filter

Options

Apply Clear Save

eCookbook 2

Recipes management application

Private child of eCookbook

This is a private subproject of a public project

Child of private child

This is a public subproject of a private project

eCookbook Subproject 1

eCookbook Subproject 1

eCookbook Subproject 2

eCookbook Subproject 2

OnlineStore

E-commerce web site

My projects

Also available in: Atom

Redmineにアクセスして以下の手順でファイルをアップロードします。

3. 「projects」をクリックします。
4. 「eCookbook」をクリックします。

Home My page Projects Administration Help

Logged in as admin My account Sign out

eCookbook Search: eCookbook

+ Overview Activity Roadmap Issues Spent time Gantt Calendar News Documents Wiki Forums **Files** Repository Settings

Files New file

File	Date	Size	D/L	Checksum	
dummy.png	02/10/2020 05:28 AM	2.16 MB	0	SHA256: 4a59992c57c9f89b017f54ffe1f2e37dd3d07a1165a36f60a980b1819434dc11	📄 🗑️
project_file.zip	07/19/2006 07:07 PM	320 Bytes	0	MDS: b91e08d0cf966d5c6ff411bd804cc3a2	📄 🗑️
redmine_logo.ai.unknown	04/30/2019 12:23 AM	4.88 KB	0	SHA256: 68d90c016f0806d88234a6f16530fd0b7f1d2481354798f1a5fd165dca6b04f	📄 🗑️
スクリーンショット 2020-02-04 9:37:05.png	02/10/2020 05:36 AM	544 KB	0	SHA256: 69a3f74ac542da02e606e8fe55f4f3cad7e10418e605d83ea4dab8ec1541fdea	📄 🗑️
0.1					
version_file.zip	07/19/2006 07:07 PM	452 Bytes	0	MDS: b91e08d0cf966d5c6ff411bd804cc3a2	📄 🗑️
version_file.zip	07/19/2006 07:07 PM	452 Bytes	0	MDS: b91e08d0cf966d5c6ff411bd804cc3a2	📄 🗑️

3. 「Files」をクリックします。
4. 「New file」をクリックします。



3. 「ファイルを選択」をクリックして、アップロードするファイルを選択します。
4. 「Add」をクリックします。

ステップ 2-5-9: S3アップロード確認

redmineにアップロードしたファイルがS3に保存されているか確認します。

AWSコンソールを開きます。



1. 「s3」を入力します。
2. 「S3」をクリックします。



先ほどフェーズ2-5-5で作成したバケット名をクリックします。自分の名前などで検索をかけると見つけやすいです。

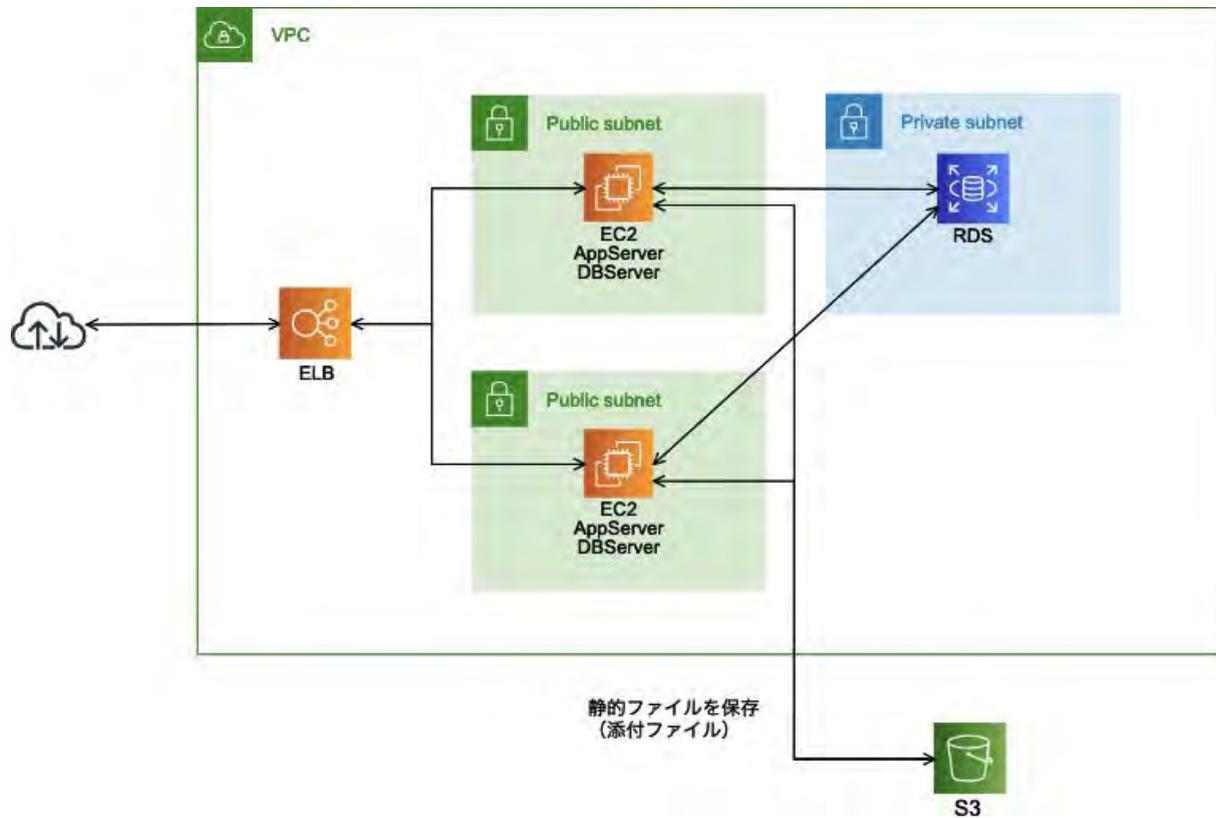
1. 自分の名前などで検索します。(日付でも)
2. フェーズ2-5-5で作成したバケット名をクリックします。



1. 「files」→「2021(年)」→「02(月)」をクリックして、先ほどredmineにアップロードしたファイルが保存されていることを確認します

[フェーズ 3]

～ロードバランサーを使った負荷分散環境を構築～



▼フェーズ 3-1: Web サーバーの AMI (パッケージ) を作成

ステップ 3-1-1: Amazon EC2 管理ページを開く



1. 「ec2」を入力します。
2. 「EC2」をクリックします。

ステップ 3-1-2: Web サーバーの AMI を作成



1. 「インスタンス」をクリックします。
2. 「webservers#1-ユーザー名」をクリックします。
3. 「アクション」をクリックします。
4. 「イメージとテンプレート」-「イメージを作成」をクリックします。

EC2 > インスタンス > i-0cb374513b86bcfe3 > イメージを作成

イメージを作成 情報

イメージ (AMI と呼ばれます) は、EC2 インスタンスの起動時に適用されるプログラムと設定を定義します。既存のインスタンスの版定からイメージを作成できます。

インスタンス ID
 i-0cb374513b86bcfe3 (webserver#1-user1)

イメージ名
 1

最大 127 文字。作成後に変更することはできません。

イメージの説明 - オプション

最大 255 文字

再起動しない
 有効化

インスタンスボリューム

ボリュームタイプ	デバイス	スナップショット	サイズ	ボリュームタイプ	IOPS	Throughput	終了時に削除	暗号化済み
EBS	/dev/x...	ボリュームから新しい...	10	EBS 汎用 SSD + gp2	100		<input checked="" type="checkbox"/> 有効化	<input type="checkbox"/>

リソースにタグが関連付けられていません。

さらに 50 のタグを追加できます

2

1. “redmine ユーザー名”などのイメージ名を入力します。
例) [redmine user1]
2. 「イメージを作成」をクリックします。

リザーブドインスタンス
 専有ホスト New
 キャパシティの予約

▼ イメージ

AMI 1

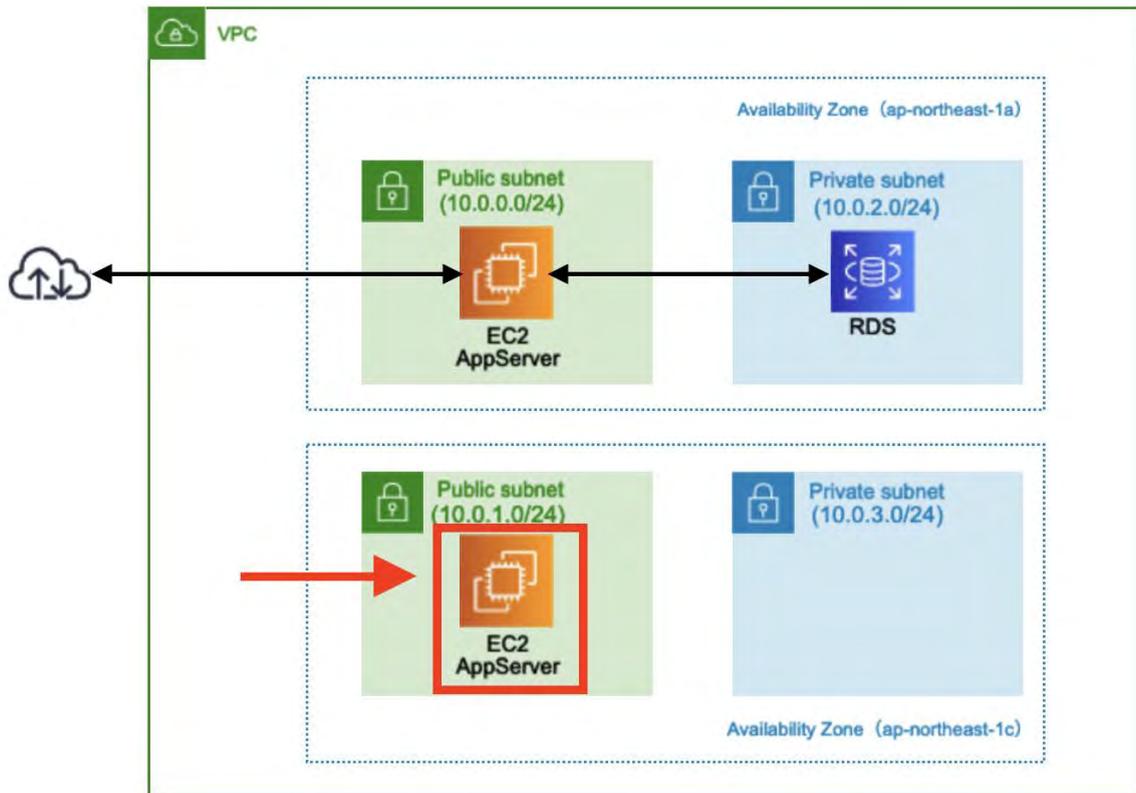
Name	AMI 名	AMI ID	ソース	所有者	可視性	ステータス	作成
	redmine user1	ami-0bae2ab4ed60cce7e	560620688551/r...	560620688551	プライベート	pending	2021

AMI の画面で AMI 作成を待ちます。完了するまで数分かかります。

「状態」欄が「available」となれば作成完了です。

「available」が表示されない場合は画面をリロードしてください。

▼フェーズ 3-2: 2 個目の Amazon EC2 インスタンスを作成



ステップ 3-2-1: 2 個目の Amazon EC2 インスタンス作成



作成した AMI からインスタンスを作成します。

1. フェーズ3-1-2で作成した AMI を右クリックします。
例)redmine user1
2. 「起動」をクリックします。

1. AMI の選択 2. インスタンスタイプの選択 3. インスタンスの設定 4. ストレージの追加 5. タグの追加 6. セキュリティグループの設定 7. 確認

ステップ 2: インスタンスタイプの選択

Amazon EC2 では、異なるユースケースに合わせて最適化されたさまざまなインスタンスタイプが用意されています。インスタンスとは、アプリケーションを実行できる仮想サーバーです。インスタンスタイプはさまざまな CPU、メモリ、ストレージ、ネットワークキャパシティの組み合わせによって構成されているため、使用するアプリケーションに合わせて適切なリソースの組み合わせを柔軟に選択できます。インスタンスタイプおよびそれをコンピューティングのニーズに適用する方法に関する 詳細はこちら。

フィルター条件:

現在選択中: t3.micro (- ECU, 2 vCPU, 2.5 GHz, -, 1 GiB メモリ, EBS のみ)

	ファミリー	タイプ	vCPU	メモリ (GiB)	インスタンスストレージ (GB)	EBS 最適化利用	ネットワークパフォーマンス	IPv6 サポート
<input type="checkbox"/>	t2	t2.nano	1	0.5	EBS のみ	-	低から中	はい
<input type="checkbox"/>	t2	t2.xlarge	8	32	EBS のみ	-	中	はい
<input type="checkbox"/>	t3	t3.nano	2	0.5	EBS のみ	はい	最大 5 ギガビット	はい
<input checked="" type="checkbox"/>	t3	t3.micro	2	1	EBS のみ	はい	最大 5 ギガビット	はい

キャンセル 戻る 次のステップ: インスタンスの詳細の設定

3. 「t3.micro」を選択します。
4. 「次のステップ: インスタンスの詳細の設定」をクリックします。

1. AMI の選択 2. インスタンスタイプの選択 3. インスタンスの設定 4. ストレージの追加 5. タグの追加 6. セキュリティグループの設定 7. 確認

ステップ 3: インスタンスの詳細の設定

要件に合わせてインスタンスを設定します。同じ AMI からの複数インスタンス作成や、より低料金を実現するためのスポットインスタンスのリクエスト、インスタンスへのアクセス管理ロール割り当てなどを行うことができます。

インスタンス数 Auto Scaling グループに作成する

購入のオプション スポットインスタンスのリクエスト

ネットワーク

サブネット
251 個の IP アドレスが利用可能

自動割り当てパブリック IP

配置グループ インスタンスをプレイズメントグループに追加します。

キャパシティの予約

ドメイン結合ディレクトリ

IAM ロール

CPU オプション CPU オプションを指定

シャットダウン動作

キャンセル 戻る 次のステップ: ストレージの追加

インスタンスは 1 個目と異なるアベイラビリティゾーンに作成します。

VPC とサブネットの選択に注意してください。

1. フェーズ1-1-5で作成した VPC を選択します。例)handson-user1
2. 「パブリックサブネット[ap-northeast-1c]」を選択します。
3. 「有効」を選択します。
4. 「次のステップ: ストレージの追加」をクリックします。

1. AMI の選択 2. インスタンスタイプの選択 3. インスタンスの設定 4. ストレージの追加 5. タグの追加 6. セキュリティグループの設定 7. 確認

ステップ 4: ストレージの追加

インスタンスは次のストレージデバイス設定を使用して作成されます。インスタンスに追加の EBS ボリュームやインスタンスストアボリュームをアタッチするか、ルートボリュームの設定を編集することができます。また、インスタンスを作成してから追加の EBS ボリュームをアタッチすることもできますが、インスタンスストアボリュームはアタッチできません。Amazon EC2 のストレージオプションに関する [詳細](#) はこちらをご覧ください。

ボリュームタイプ	デバイス	スナップショット	サイズ (GiB)	ボリュームタイプ	IOPS	スループット (MB/秒)	終了時に削除	暗号化
ルート	/dev/xvda	snap-00e37af63af76d77c	8	汎用 SSD (gp2)	100 / 3000	該当なし	<input checked="" type="checkbox"/>	暗号化

新しいボリュームの追加

キャンセル 戻る 確認と作成 **次のステップ: タグの追加**

ストレージは変更せずに、次に進みます。

1. 「次のステップ: タグの追加」をクリックします。

1. AMI の選択 2. インスタンスタイプの選択 3. インスタンスの設定 4. ストレージの追加 5. タグの追加 6. セキュリティグループの設定 7. 確認

ステップ 5: タグの追加

タグは、大文字と小文字が区別されるキーと値のペアから構成されます。たとえば、キーに「Name」、値に「Webserver」を使用してタグを定義することができます。タグのコピーは、ボリューム、インスタンス、またはその両方に適用できます。タグは、すべてのインスタンスとボリュームに適用されます。Amazon EC2 リソースのタグ付けに関する [詳細](#) はこちら。

キー (最大 128 文字)	値 (最大 256 文字)	インスタンス	ボリューム
Name	webserver#2-user1	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

タグの追加 (最大 50 個のタグ)

キャンセル 戻る 確認と作成 **次のステップ: セキュリティグループの設定**

インスタンスを区別できるようにタグに名前を設定します。

1. 「タグの追加」をクリックします。
2. キーに「Name」と入力します。

- 「webserver#2- ユーザー名」とします。
例)[webserver#2-user1]
- 「次のステップ: セキュリティグループの設定」をクリックします。

1. AMI の選択 2. インスタンスタイプを選択 3. インスタンスの設定 4. ストレージの追加 5. タグの追加 6. セキュリティグループの設定 7. 確認

ステップ 6: セキュリティグループの設定

セキュリティグループは、インスタンスのトラフィックを制御するファイアウォールのルールセットです。このページで、特定のトラフィックに対してインスタンスへの到達を許可するルールを追加できます。たとえば、ウェブサーバーをセットアップして、インターネットトラフィックにインスタンスへの到達を許可する場合、HTTP および HTTPS ポートに無制限のアクセス権限を与えます。新しいセキュリティグループを作成するか、次の既存のセキュリティグループから選択することができます。Amazon EC2 セキュリティグループに関する [詳細はこちら](#)。

セキュリティグループの割り当て: 新しいセキュリティグループを作成する 既存のセキュリティグループを選択する **1**

セキュリティグループ ID	名前	説明	アクション
sg-0aabd2572a015ff88	db-user1	RDS for MySQL	コピーして新規作成
sg-024a52f3067fadd71	default	default VPC security group	コピーして新規作成
<input checked="" type="checkbox"/> sg-00f3d0e5d3fc12c43	web-user1	web-user1	コピーして新規作成 2

sg-00f3d0e5d3fc12c43 に関するインバウンドのルール (選択したセキュリティグループ: sg-00f3d0e5d3fc12c43)

タイプ	プロトコル	ポート範囲	ソース	説明
HTTP	TCP	80	0.0.0.0/0	

キャンセル 戻る **確認と作成** **3**

既に行ったセキュリティグループを使用します。

- 「既存のセキュリティグループを選択する」をクリックします。
- フェーズ **1-3-4** で作成したセキュリティグループ (**web-user1** 等) をクリックします。
- 「確認と作成」をクリックします。

1. AMI の選択 2. インスタンスタイプを選択 3. インスタンスの設定 4. ストレージの追加 5. タグの追加 6. セキュリティグループの設定 7. 確認

ステップ 7: インスタンス作成の確認

インスタンスの作成に関する詳細を確認してください。各セクションの変更に戻ることができます。[作成] をクリックして、インスタンスにキーペアを割り当て、作成処理を完了します。

⚠ インスタンスのセキュリティを強化してください。セキュリティグループ web-user1 は世界に向けて開かれています。
このインスタンスには、どの IP アドレスからもアクセスできる可能性があります。セキュリティグループのルールを更新して、既知の IP アドレスからのみアクセスできるようにすることをお勧めします。
また、セキュリティグループの追加ポートを開いて、実行中のアプリケーションやサービスへのアクセスを容易にすることもできます。たとえば、ウェブサーバー用に HTTP (80) を開きます。 [セキュリティグループの編集](#)

AMI の詳細 AMI の編集

wordpress user1 - ami-09120f42858f0b0d7
ルートデバイスタイプ: ebs 仮想化タイプ: hvm

インスタンスタイプ インスタンスタイプの編集

インスタンスタイプ	ECU	vCPU	メモリ (GiB)	インスタンス ストレージ (GB)	EBS 最適化利用	ネットワークパフォーマンス
t2.micro	可変	1	1	EBS のみ	-	Low to Moderate 1

キャンセル 戻る **起動**

設定内容を確認してから作成します。

1. 「起動」をクリックします。

ステップ 3-2-2: キーペアを選択する

既存のキーペアを選択するか、新しいキーペアを作成します。 ×

キーペアは、AWS が保存するパブリックキーとユーザーが保存するプライベートキーファイルで構成されます。組み合わせて使用することで、インスタンスに安全に接続できます。Windows AMI の場合、プライベートキーファイルは、インスタンスへのログインに使用されるパスワードを取得するために必要です。Linux AMI の場合、プライベートキーファイルを使用してインスタンスに SSH で安全に接続できます。

注: 選択したキーペアは、このインスタンスに対して権限がある一連のキーに追加されます。「パブリック AMI から既存のキーペアを削除する」の詳細情報をご覧ください。

1. 「既存のキーペアの選択」を選択します。

2. 先ほど作成した「handson-20201221」を選択します。

3. チェックを入れます。

4. 「インスタンスの作成」をクリックします。

既存のキーペアを選択します。

1. 「既存のキーペアの選択」を選択します。
2. 先ほど作成した「handson-2021xxxx」を選択します。
3. チェックを入れます。
4. 「インスタンスの作成」をクリックします。

ステップ 3-2-3: 作成した 2 個目の EC2 インスタンスを確認

The screenshot shows the AWS Management Console interface for EC2 instances. On the left, there is a navigation menu with options like 'EC2 ダッシュボード', 'イベント', 'タグ', '制限', 'インスタンス', 'イメージ', 'Elastic Block Store', and 'ネットワーク & セキュリティ'. The main area displays a list of instances under the heading 'インスタンス (1/2)'. The instance 'webservers#2-user1' is selected, and its details are shown below. The 'Networking' tab is active, displaying network configuration details. Three red boxes with numbers 1, 2, and 3 highlight specific elements: 1. The instance name 'webservers#2-user1' in the list. 2. The 'Networking' tab in the instance details view. 3. The availability zone 'ap-northeast-1c' in the 'Availability Zones' section of the networking details.

Name	インスタンス ID	インスタンス...	インスタン...	ステータスチェック
webservers#1-user1	i-08684383ac0a66b83	実行中	t3.micro	2/2 のチェックに
webservers#2-user1	i-052231e4a1715ffa2	実行中	t3.small	-

インスタンス: i-052231e4a1715ffa2 (webservers#2-user1)

詳細 | セキュリティ | **ネットワーキング** | ストレージ | ステータスチェック | モニタリング | タグ

▼ ネットワーキングの詳細 情報

パブリック IPv4 アドレス	プライベート IPv4 アドレス	VPC ID
18.183.2.230 オープンアドレス	10.0.1.172	vpc-009c18aca5dc628b1 (handson-user1)
パブリック IPv4 DNS	プライベート IPv4 DNS	サブネット ID
ec2-18-183-2-230.ap-northeast-1.compute.amazonaws.com オープンアドレス	ip-10-0-1-172.ap-northeast-1.compute.internal	subnet-03561a3c4065809a8 (パブリックサブネットc)
IPv6 アドレス	セカンダリプライベート IPv4 アドレス	アベイラビリティゾーン
-	-	ap-northeast-1c

インスタンスの作成が完了するのに数分間かかります。

他ユーザのインスタンスが表示されている場合は上部の検索ボックスにユーザー名を入れて絞り込んでください。

webservers#2-ユーザー名、ap-northeast-1c に作成されていることを確認してください。

1. 2個目のインスタンスが作成されていることを確認する。
2. 2個目のインスタンスを選択し、「ネットワーキング」をクリックします。
3. アベイラビリティゾーンが「ap-northeast-1c」であることを確認する。

ステップ 3-2-4: 2 個目の EC2 インスタンスのパブリックIPv4アドレスをメモ

The screenshot shows the AWS Management Console interface for EC2 instances. On the left, there is a navigation menu with categories like 'インスタンス', 'イメージ', 'Elastic Block Store', and 'ネットワーク & セキュリティ'. The main area displays a list of instances. The second instance, 'webservers#2-user1', is selected, indicated by a red box and a circled '1'. Below the list, the details for this instance are shown. The 'パブリック IPv4 アドレス' (Public IPv4 address) is highlighted with a red box and a circled '2', showing the value '18.183.2.230 | オープンアドレス'.

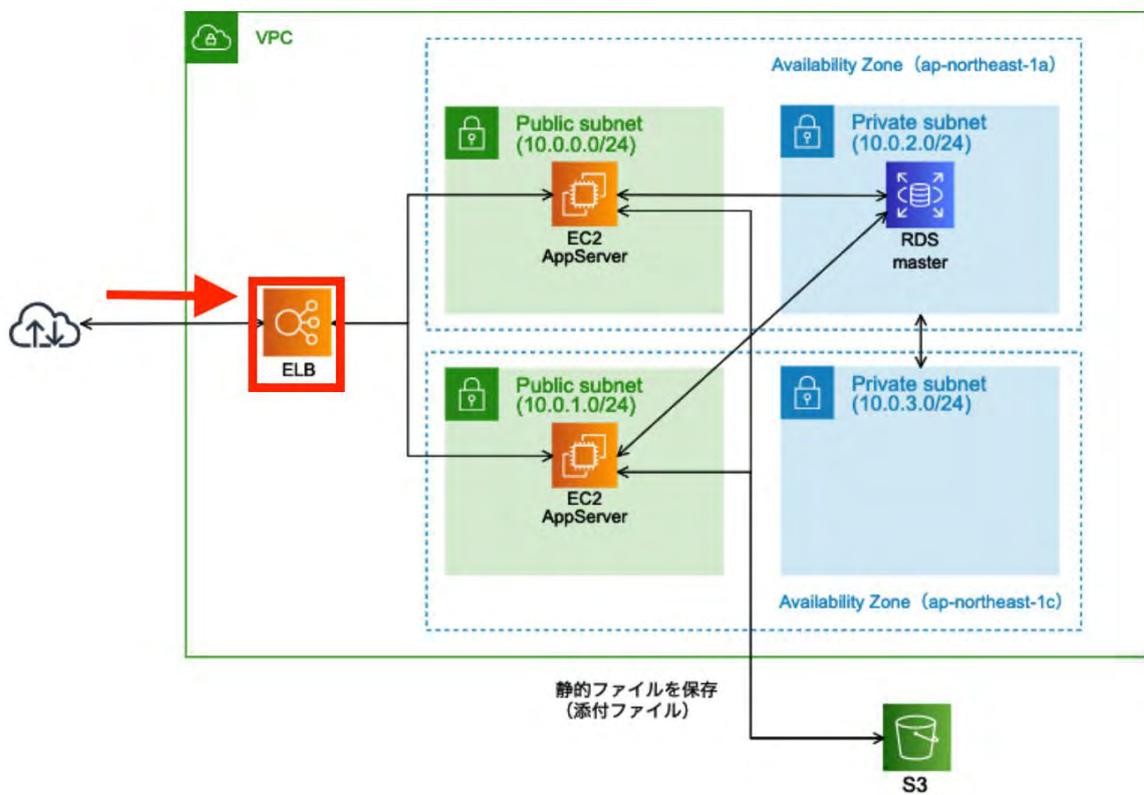
Name	インスタンス ID	インスタンス...	インスタン...	ステータスチェック
webservers#1-user1	i-08684383ac0a66b85	実行中	t3.micro	2/2 のチェックに
webservers#2-user1	i-052231e4a1715ffa2	実行中	t3.small	-

インスタンス ID	パブリック IPv4 アドレス	プライベート IPv4 アドレス
i-052231e4a1715ffa2 (webservers#2-user1)	18.183.2.230 オープンアドレス	10.0.1.172

1. 2個目のインスタンス「webservers#2- ユーザー名」を選択する
例) [webservers#2-user1]
2. 「パブリックIPv4アドレス」をメモする。

▼フェーズ 3-3: Elastic Load Balancing (ロードバランサー)を作成

ステップ 3-3-1: ELB を作成





2 台の Web サーバーへのアクセスを振り分ける ELB を作成します。

1. 「ロードバランサー」を選択します。(EC2のサイドメニュー)
2. 「ロードバランサーの作成」をクリックします。

ステップ 3-3-2: ALBを選択

ロードバランサーの種類の選択

Elastic Load Balancing は 3 種類のロードバランサー (Application Load Balancer、Network Load Balancer (新規)、および Classic Load Balancer) をサポートします。お客様のニーズに合うロードバランサーの種類を選択してください。

お客様に最適なロードバランサーの詳細

Application Load Balancer

HTTP
HTTPS

1
作成

HTTP および HTTPS トラフィックを使用するウェブアプリケーション用に柔軟性の高い機能セットが必要な場合は、Application Load Balancer を選択します。Application Load Balancer はリクエストレベルで動作し、マイクロサービスとコンテナを含む、アプリケーションアーキテクチャを対象とした高度なルーティングおよび可視性機能を提供します。

[詳細はこちら >](#)

Network Load Balancer

TCP
TLS
UDP

作成

非常に高いパフォーマンス、大規模な TLS のオフロード、証明書のデプロイの一元管理、UDP のサポート、およびアプリケーションの静的 IP アドレスが必要な場合は、Network Load Balancer を選択します。Network Load Balancer は接続レベルで動作し、非常に低いレイテンシーを維持しながら、1 秒あたり数百万のリクエストを確実に処理することができます。

[詳細はこちら >](#)

Classic Load Balancer

以前の世代
HTTP、HTTPS、および TCP

作成

EC2-Classical ネットワークで既存のアプリケーションを実行している場合は、Classic Load Balancer を選択します。

[詳細はこちら >](#)

今回は「Application Load Balancer」を選択します。

ステップ 3-3-3: ELB を作成 (1)

1. ロードバランサーの設定 2. セキュリティ設定の構成 3. セキュリティグループの設定 4. ルーティングの設定 5. ターゲットの登録 6. 確認

手順 1: ロードバランサーの設定

基本的な設定

ロードバランサーを設定するには、名前を指定し、スキームを選択して、1 つ以上のリスナーを指定し、ネットワークを選択します。デフォルトの設定は、ポート 80 で HTTP トラフィックを受信するリスナーを持つ、選択したネットワークのインターネット接続ロードバランサーです。

名前 (i)	elb-user1	1
スキーム (i)	<input checked="" type="radio"/> インターネット向け <input type="radio"/> 内部	
IP アドレスタイプ (i)	ipv4	

リスナー

リスナーとは、設定したプロトコルとポートを使用して接続リクエストをチェックするプロセスです。

ロードバランサーのプロトコル	ロードバランサーのポート
HTTP	80
リスナーの追加	

1. 「**elb**-ユーザ名」と入力します。
例) elb-user1

アベイラビリティゾーン

ロードバランサーのアベイラビリティゾーンを指定します。ロードバランサーは、指定されたアベイラビリティゾーンにのみトラフィックをルーティングします。アベイラビリティゾーンごとに1つだけサブネットを指定できます。ロードバランサーの可用性を高めるには、2つ以上のアベイラビリティゾーンからサブネットを指定する必要があります。

ELB を 2 つのパブリックサブネットに配置します。

利用可能なサブネット一覧からパブリックサブネット 2 つを「+」をクリックして選択してください。

1. VPCは「**handson-user1**」をクリックを選択します。
2. 「**ap-northeast-1a**」,「**ap-northeast-1c**」にチェックを入れ、それぞれ「パブリックサブネット」を選択します。
3. 「次の手順: セキュリティ設定の構成」をクリックします。

手順 2: セキュリティ設定の構成

特に設定しないため、そのまま次の手順へ進みます。

4. 「次の手順: セキュリティグループの設定」をクリックします。

ステップ 3-3-4: ELB を作成 (2)

1. ロードバランサーの設定 2. セキュリティ設定の構成 3. セキュリティグループの設定 4. ルーティングの設定 5. ターゲットの登録 6. 確認

手順 3: セキュリティグループの設定

セキュリティグループは、ロードバランサーへのトラフィックを制御するファイアウォールのルールセットです。このページで、特定のトラフィックに対してロードバランサーへの到達を許可するルールを追加できます。最初に、新しいセキュリティグループを作成するか、既存のセキュリティグループから選択するかを決定します。

セキュリティグループの割り当て 新しいセキュリティグループを作成する **1**
 既存のセキュリティグループを選択する

セキュリティグループ名 **2**
 説明

タイプ	プロトコル	ポート範囲	ソース
<input type="text" value="HTTP"/> 3	TCP	80	<input type="text" value="任意の場所"/> 4 0.0.0.0/0, ::0

ルールの追加

キャンセル 戻る **5** 次の手順: ルーティングの設定

1. 「新しいセキュリティグループを作成する」を選択します。
2. セキュリティグループ名と説明に「elb-ユーザ名」名前を入力します。
例) elb-user1
3. 「HTTP」を選択します。
4. 「任意の場所」を選択します。
5. 「次の手順: ルーティングの設定」をクリックします。

ステップ 3-3-5: ELB を作成 (3)

1. ロードバランサーの設定 2. セキュリティ設定の構成 3. セキュリティグループの設定 4. ルーティングの設定 5. ターゲットの登録 6. 確認

手順 4: ルーティングの設定

Your load balancer routes requests to the targets in this target group using the protocol and port that you specify, and performs health checks on the targets using these health check settings. The target group you specify in this step will apply to all of the listeners configured on this load balancer; you can edit the listeners and add listeners after the load balancer is created.

ターゲットグループ

ターゲットグループ

名前 ①

ターゲットの種類

- インスタンス
- IP
- Lambda 関数

プロトコル

ポート

プロトコルバージョン

- HTTP1
HTTP/1.1 を使用してターゲットにリクエストを送信します。これはリクエストプロトコルが HTTP/1.1 または HTTP/2 の場合にサポートされます。
- HTTP2
HTTP/2 を使用してターゲットにリクエストを送信します。これはリクエストプロトコルが HTTP/2 または gRPC の場合にサポートされますが、gRPC 固有の機能は使用できません。
- gRPC
gRPC を使用してターゲットにリクエストを送信します。これはリクエストプロトコルが gRPC の場合にサポートされます。

新しいターゲットグループを作成します。

- 「elb-ユーザ名」と名前を入力します。
例) elb-user1

ステップ 3-3-6: ELB を作成 (4)

ヘルスチェック

プロトコル

パス ①

ヘルスチェックの詳細設定

ポート トラフィックポート
 上書き

正常のしきい値

非正常のしきい値

タイムアウト 秒

間隔 秒

成功コード

キャンセル 戻る ②

ヘルスチェックの条件を変更します。

1. 「/login」に変更します。
2. 「次の手順: ターゲットの登録」をクリックします。

ステップ 3-3-7: ELB を作成 (5)

1. ロードバランサーの設定 2. セキュリティ設定の構成 3. セキュリティグループの設定 4. ルーティングの設定 5. **ターゲットの登録** 6. 確認

手順 5: ターゲットの登録

ターゲットグループにターゲットを登録します。有効にしたアベイラビリティゾーンでターゲットを登録する場合、登録処理が完了し、ターゲットが最初のヘルスチェックに合格するとすぐに、ロードバランサーはターゲットへのリクエストのルーティングを開始します。

登録済みターゲット

インスタンスを登録解除するには、1つ以上の登録インスタンスを選択し、[削除] をクリックします。

インスタンス	名前	ポート	状態	セキュリティグループ	ゾーン
利用可能なインスタンスがありません。					

インスタンス

追加のインスタンスを登録するには、1つ以上の実行中のインスタンスを選択し、ポートを指定して、[追加] をクリックします。デフォルトのポートは、ターゲットグループに対して指定されたポートです。指定されたポートすでにインスタンスが登録されている場合は、別のポートを指定する必要があります。

登録済みに追加 ポート

Q インスタンスを検索 X

1	インスタンス	名前	状態	セキュリティ	ゾーン	サブネット ID	サブネット CIDR
<input checked="" type="checkbox"/>	i-0b2df657a8fd03...	webserver#2-user1	running	web-user1	ap-northeast-1c	subnet-00601428dba9dee5f	10.0.1.0/24
<input checked="" type="checkbox"/>	i-0cb374513b86b...	webserver#1-user1	running	web-user1	ap-northeast-1a	subnet-08a4d9a7da978501b	10.0.0.0/24

1. 「インスタンス」に表示されている2つのインスタンスを選択する。
2. 「登録済みに追加」をクリックする。

1. ロードバランサーの設定 2. セキュリティ設定の確認 3. セキュリティグループの設定 4. ルーティングの設定 5. ターゲットの登録 6. 確認

手順 5: ターゲットの登録

ターゲットグループにターゲットを登録します。有効にした Availability Zones でターゲットを登録する場合、登録処理が完了し、ターゲットが最初のヘルスチェックに合格するとすぐに、ロードバランサーはターゲットへのリクエストのルーティングを開始します。

登録済みターゲット

インスタンスを登録解除するには、1つ以上の登録インスタンスを選択し、[削除] をクリックします。

①

インスタンス	名前	ポート	状態	セキュリティグループ	ゾーン
i-0b2df657a8fd0348a	webserver#2-user1	80	running	web-user1	ap-northeast-1c
i-0cb374513b86bcte3	webserver#1-user1	80	running	web-user1	ap-northeast-1a

インスタンス

追加のインスタンスを登録するには、1つ以上の実行中のインスタンスを選択し、ポートを指定して、[追加] をクリックします。デフォルトのポートは、ターゲットグループに対して指定されたポートです。指定されたポートですでにインスタンスが登録されている場合は、別のポートを指定する必要があります。

登録済みを追加 ポート 80

Q インスタンスを検索 X

インスタンス	名前	状態	セキュリティ	ゾーン	サブネット ID	サブネット CIDR
i-0b2df657a8fd03...	webserver#2-user1	running	web-user1	ap-northeast-1c	subnet-00601428dba9dee5f	10.0.1.0/24
i-0cb374513b86b...	webserver#1-user1	running	web-user1	ap-northeast-1a	subnet-08a4d9a7da978501b	10.0.0.0/24

キャンセル 戻る **次の手順: 確認** ②

登録済みターゲットに2つのインスタンスが追加されていることを確認する。

1. 「登録済みターゲット」に2つのインスタンスが追加されていることを確認する。
2. 「次の手順: 確認」をクリックする。

ステップ 3-3-8: ELB を作成 (6)

1. ロードバランサーの設定 2. セキュリティ設定の構成 3. セキュリティグループの設定 4. ルーティングの設定 5. ターゲットの登録 6. 確認

手順 6: 確認
 実行する前にロードバランサーの詳細を確認してください。

▼ ロードバランサー 編集

名前 elb-user1
 スキーム internet-facing
 リスナー ポート:80 - プロトコル:HTTP
 IP アドレスタイプ ipv4
 VPC vpc-0a9c06e1a7e78a21a (handson-user1)
 サブネット subnet-08a4d9a7da978501b (パブリックサブネット), subnet-00601428dba9dee5f (パブリック サブネット-c)
 タグ

▼ セキュリティグループ 編集

セキュリティグループ elb-user1

▼ ルーティング 編集

ターゲットグループ 新しいターゲットグループ
 ターゲットグループ名 elb-user1
 ポート 80
 ターゲットの種類 instance
 プロトコル HTTP
 プロトコルバージョン HTTP1
 ヘルスチェックプロトコル HTTP
 パス /login
 ヘルスチェックポート traffic port
 正常のしきい値 5
 非正常のしきい値 2
 タイムアウト 5
 間隔 30
 health コード 200

キャンセル 戻る **1** 作成

1. 「作成」をクリックします。

ステップ 3-3-10: 作成されたELBを確認

ロードバランサー作成状況

✔ ロードバランサーを正常に作成しました。
 ロードバランサー elb-user1 が正常に作成されました。
 注 新しいロードバランサーでインスタンスがアクティブになるまで自分がかかることがあります。

1 閉じる

ELB が作成されました。

1. 「閉じる」をクリックします。

ロードバランサーの作成

検索 : user1 1 一の追加

名前	DNS 名	状態	VPC ID	アベイラビリティゾーン
elb-user1	elb-user1-236872948.ap-nor...		vpc-0219c5e2bc2073785	ap-northea

ロードバランサー: elb-user1

説明 インスタンス ヘルスチェック リスナー モニタリング タグ 移行

基本的な設定

名前	elb-user1	作成時刻	2020年1月28日 14:04:58 UTC+9
DNS 名	elb-user1-236872948.ap-northeast-1.elb.amazonaws.com (A レコード) 3	ホストゾーン	Z14GRHDCWA56QT
種類	Classic (今すぐ移行)	ステータス	2個のうち 0個のインスタンスが実行中です
スキーム	internet-facing	VPC	vpc-0219c5e2bc2073785
アベイラビリティゾーン	subnet-018727f21dfd31004 - ap-northeast-1c. subnet-024c450a811ced4fd - ap-northeast-1a		

作成された ELB の DNS 名 (ホスト名) をメモします。

(Aレコード) は省きます。

1. ユーザー名で絞りこみます。
2. 先ほど作成した **ELB** を選択します。
3. **DNS名** をメモします。

※ ロードバランサーが作成されるまで少し時間がかかります。「状態」が active になったのを確認してからアクセスしてください。

▼ フェーズ 3-4: Elastic Load Balancing 経由でアクセス

ステップ 3-4-1: ELB 経由でアクセス

`http://<ELB の DNS 名>/` を開いて `redmine` が表示されることを確認します。

ステップ 3-4-2: 両方のサーバにアクセスがされているか確認

`webserver#1`, `webserver#2` それぞれに `ssh` でログインし、以下のコマンドを実行してアクセスログを表示させることが可能です。

ELB の定期的なヘルスチェックが実行されたり、`redmine` でページをリロードするたびに双方の EC2 へアクセスされている状況を確認できます。

[`webserver#1`]

以下のコマンドを実行します。

```
ssh -i "handson-2021xxxx.pem" bitnami@[ webserver#1のElastic IPアドレス ]
```

```
$ sudo su
```

```
# redmineのディレクトリに移動
$ cd /opt/bitnami/apps/redmine/htdocs/
# アクセスログを表示
$ tail -f log/production.log
```

[**webserver#2**]

別ターミナルを開き、**webserver#1**と同様にssh接続してアクセスログを表示します。

```
ssh -i "handson-2021xxxx.pem" bitnami@[ webserver#2のパブリックIPv4アドレス
]
```

```
$ sudo su
# redmineのディレクトリに移動
$ cd /opt/bitnami/apps/redmine/htdocs/
# アクセスログを表示
$ tail -f log/production.log
```

redmineをリロード等してログがそれぞれに流れることを確認してください。

ログ表示はCtrl + C で終了できます。

▼フェーズ 3-5: セキュリティグループ設定変更

ステップ 3-5-1: セキュリティグループ設定変更

The screenshot shows the AWS Management Console interface for managing security groups. The left sidebar contains navigation options, with 'セキュリティグループ' (Security Groups) highlighted. The main content area displays a list of security groups with a search filter 'user1' applied. The 'web-user1' group is selected. Below the list, the 'インバウンドルール' (Inbound Rules) tab is active, and the 'インバウンドルールを編集' (Edit Inbound Rules) button is highlighted.

Name	セキュリティグループ...	セキュリティグルー...	VPC ID
-	sg-08dc5e471ddb376de	db-user1	vpc-06c666d0963d91afc
<input checked="" type="checkbox"/>	sg-0c559b3a6ec816b9a	web-user1	vpc-06c666d0963d91afc
<input type="checkbox"/>	sg-0e28b98ce24779657	elb-user1	vpc-06c666d0963d91afc

タイプ	プロトコル	ポート範囲	ソース	説明 - オプション
HTTP	TCP	80	0.0.0.0/0	-
SSH	TCP	22	0.0.0.0/0	-

セキュリティグループの設定を変更し、Web サーバーへの HTTP アクセスは ELB からに限定するようにします。

1. 「セキュリティグループ」をクリックします。(EC2のサイドメニュー)
2. ユーザー名で絞り込みます。
3. グループ名「web-ユーザー名」を選択します。
4. 「インバウンドルール」をクリックします。
5. 「インバウンドルールを編集」をクリックします。

EC2 > セキュリティグループ > sg-0c559b3a6ec816b9a - web-user1 > インバウンドルールを編集

インバウンドルールを編集 情報

インバウンドルールは、インスタンスに到達できる着信トラフィックをコントロールします。

インバウンドルール 情報

タイプ <small>情報</small>	プロトコル <small>情報</small>	ポート範囲 <small>情報</small>	ソース <small>情報</small>	説明 - オプション <small>情報</small>	
HTTP	TCP	80	カスタム <input type="text" value="0.0.0.0/0"/> <input type="button" value="X"/>	<input type="text"/>	<input type="button" value="削除"/>
SSH	TCP	22	カスタム <input type="text" value="セキュリティグループ"/>	<input type="text"/>	<input type="button" value="削除"/>

注意: 既存のルールを編集すると、編集したルールが削除されて、新しい詳細を含む新しいルールが作成されます。これにより、そのルールに依存するトラフィックは、新しいルールが作成されるまで非常に短時間切断されます。

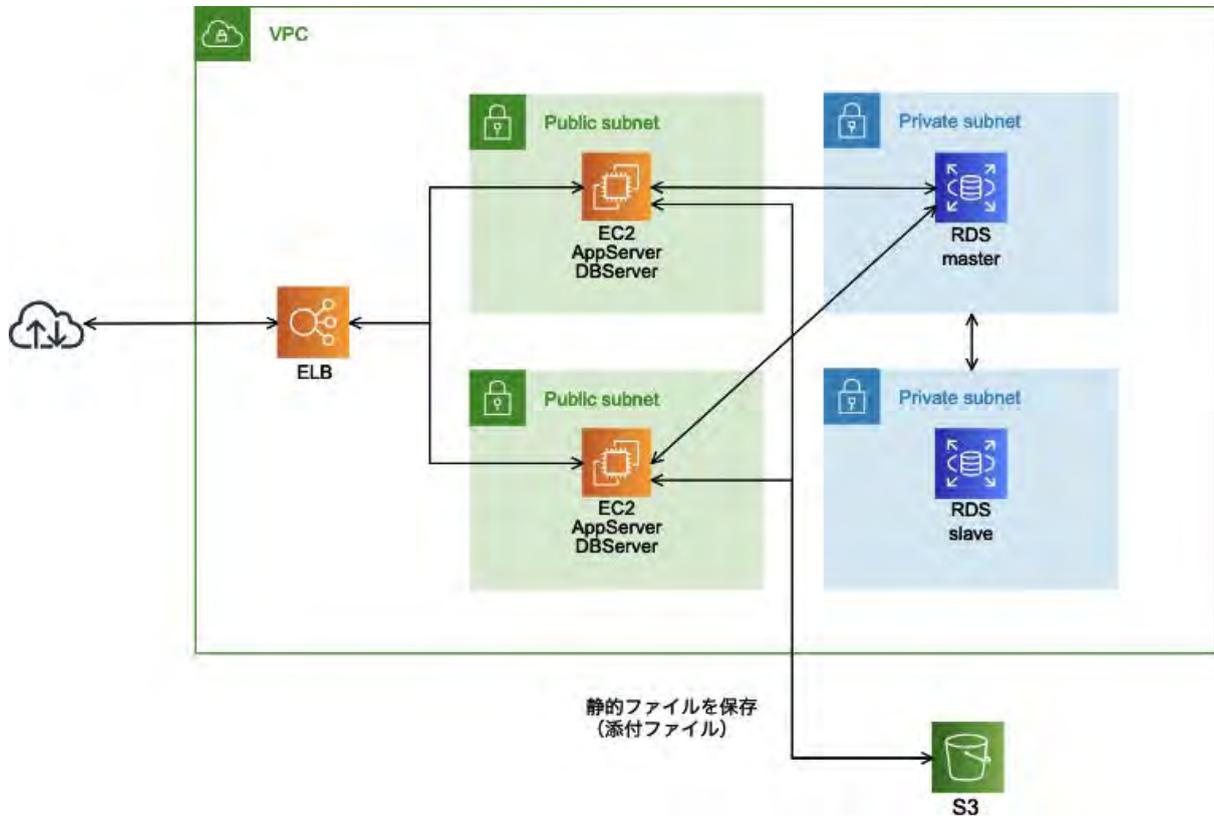
キャンセル

1. タイプ httpのリソースの「0.0.0.0/0」を x をクリックして削除します。
2. 入力欄に「elb」と入力して候補を表示させます。
3. 表示された候補から「elb-ユーザー名」を選択します。
4. 「ルールを保存」をクリックします。

保存後、ロードバランサーのDNSからはアクセス可能で、EC2のパブリックDNSからアクセスできないことを確認してください。

[フェーズ 4]

~ Amazon RDS を Multi-AZ 構成に変更 ~



▼フェーズ 4: Amazon RDS を Multi-AZ 構成に変更

ステップ 4-1: RDS 管理ページを開く



1. 「rds」を入力します。

2. 「RDS」をクリックします。

ステップ 4-2: RDS インスタンスの設定変更



1. 「データベース」を選択します。
2. フェーズ2-3-1で作成したRDSインスタンスを選択します。
3. 「変更」をクリックします。

ステップ 4-3: Multi-AZ を有効にする



マルチAZ配置設定

1. 「可用性と耐久性」の項目で「スタンバイインスタンスを作成する」を選択します。
2. 選択後、ページ下部の「続行」をクリックします。

RDS > データベース > 変更

DB インスタンスの変更: wp-user1

変更の概要

以下の変更を送信しようとしています。変更される値のみが表示されます。変更をよく確認してから、[DB インスタンスの変更] をクリックしてください。

属性	現在の値	新しい値
マルチ AZ 配置	いいえ	はい



Potential performance impact

You may experience a significant performance impact when converting this database instance to Multi-AZ configuration. This impact will be more noticeable on database instances with large amounts of storage and write-intensive workloads.

変更のスケジュール

変更を適用する時間

 次に予定されるメンテナンスウィンドウ中に適用します

最新のメンテナンスウィンドウ: thu:18:34-thu:19:04

1

 すぐに適用

このデータベースインスタンスのメンテナンスウィンドウ設定に関わらず、このリクエストの変更とすべての保留中の変更はできるだけ早く非同期に適用されます。



予期されないダウンタイムの可能性

変更の即時適用を選択した場合、保留中の変更キューにあるすべての変更も同様に適用されます。ダウンタイムを必要とする保留中の変更がある場合、即時適用を選択すると予想外のダウンタイムが発生することがあります。

2

キャンセル

戻る

DB インスタンスの変更

「すぐに適用」をオンにしなければ、サーバーの停止や負荷が伴う変更は次のメンテナンスウィンドウのタイミングで適用されますが、今回は「すぐに適用」を行います。

1. 「すぐに適用」にチェックを入れます。
2. 「DB インスタンスの変更」をクリックします。

ステップ 4-4: Multi-AZ 化の完了を確認



1. フェーズ2-3-1で作成したDBインスタンス(例: redmine-user1)をクリックします。

The screenshot displays the Amazon RDS console for the instance 'redmine-user1'. The left sidebar contains navigation options like 'ダッシュボード', 'データベース', 'Query Editor', etc. The main content area shows the instance details under the '概要' tab. A red box highlights the '情報' (Info) section, which indicates the instance is '利用可能' (Available), marked with a circled '1'. Below this, the '接続とセキュリティ' (Connections and Security) section is visible, showing details for endpoints, ports, network, and security groups.

概要			
DB 識別子 redmine-user1	CPU 2.67%	情報 利用可能	クラス db.t2.micro
ロール インスタンス	現在のアクティビティ 2 接続	エンジン MySQL Community	リージョンと AZ ap-northeast-1a

接続とセキュリティ		
エンドポイントとポート	ネットワーク	セキュリティ
エンドポイント redmine-user1.cizpucnnhfj8.ap-northeast-1.rds.amazonaws.com	アベイラビリティゾーン ap-northeast-1a	VPC セキュリティグループ db-user1 (sg-099eb01756122c893) (アクティブ)
ポート 3306	VPC handson-user1 (vpc-0a941f74723ca26f2)	パブリックアクセスセキュリティ なし
	サブネットグループ db subnet user1	認証機関 rds-ca-2019
	サブネット subnet-05dae220b74f2a8c1 subnet-082bafb5ee8ec3a86	証明機関の日付 Aug 23rd, 2024

変更完了を待ちます(約 10 分間かかります)。

ステータスが[利用可能]にならない場合は、画面を更新して再描画します。

ステップ 4-5: 設定変更内容を確認

The screenshot shows the Amazon RDS console for a MySQL instance named 'redmine-user1'. The left sidebar contains navigation options like 'ダッシュボード', 'データベース', 'Query Editor', etc. The main content area shows the instance details and configuration options. The '設定' (Settings) tab is highlighted with a red box and a circled '1'. In the '可用性' (Availability) section, the 'マルチAZ' (Multi-AZ) option is checked and highlighted with a red box and a circled '2'.

概要			
DB 識別子	CPU	情報	クラス
redmine-user1	2.67%	利用可能	db.t2.micro
ロール	現在のアクティビティ	エンジン	リージョンと AZ
インスタンス	2 接続	MySQL Community	ap-northeast-1a

インスタンス			
設定	インスタンスクラス	ストレージ	Performance Insights
DB インスタンス ID	インスタンスクラス	暗号化	Performance Insights が有効
redmine-user1	db.t2.micro	有効でない	なし
エンジンバージョン	vCPU	ストレージタイプ	
5.7.22	1	汎用 (SSD)	
DB 名	RAM	IOPS	
rds_redmine	1 GB	-	
ライセンスモデル	可用性	ストレージ	
General Public License	マルチAZ	20 GiB	
オプショングループ	マスターユーザー名	ストレージの自動スケールリング	
default:mysql-5-7	admin	有効	
ARN	IAM db 認証	最大ストレージしきい値	
arn:aws:rds:ap-northeast-1:533384410763:db:redmine-user1	有効でない	1000 GiB	

Multu-AZ 配置への設定がすぐに適用されることを確認します。

1. 「設定」をクリックします。
2. 「マルチAZ」がありであることを確認します。

ステップ 4-6: RDS インスタンスをフェイルオーバーさせる



RDS をスタンバイ側に切り替え、挙動を確認します。

1. 「データベース」をクリックします。
2. フェーズ2-3-1で作成したインスタンスを選択します。
3. 「アクション」をクリックします。
4. 「再起動」をクリックします。



フェイルオーバーを選択して再起動させます。(再起動が完了するまでは redmine にアクセスできなくなります。再起動が完了すると元通りアクセスできるようになります。)

1. 「フェイルオーバーし再起動します」にチェックを入れます。
2. 「再起動」をクリックします。

The screenshot shows the Amazon RDS console interface. On the left is a navigation menu with options like 'ダッシュボード', 'データベース', 'Query Editor', etc. The main content area displays the details for the database instance 'redmine-user1'. The '概要' (Summary) section contains a table of instance properties:

DB 識別子	CPU	ステータス	クラス
redmine-user1	4.00%	利用可能	db.t3.micro
ロール	現在のアクティビティ	エンジン	リージョンと AZ
インスタンス	0 接続	MySQL Community	ap-northeast-1c

The 'リージョンと AZ' (Region and Availability Zone) field is highlighted with a red rectangular box.

起動後、「リージョンとAZ」が **ap-northeast-1c**に変更されていることを確認してください。

～ 構築した環境の後片付け ～

今回構築した環境は、そのままにしておくとも費用が発生するものがあります。

フェーズ 4 までの作業終了・または途中で作業を終了される場合は、以下の手順で構築した環境の後片付けをお願いします。

以下の手順で構築した環境の後片付けをしてください。

[RDS]

* データベース

DB識別子が「redmine-自分の名前(user1)」を削除

1. 選択->アクション->削除
2. 「最終スナップショットを作成しますか？」のチェックを外す。
3. 「インスタンスの削除後、システムスナップショットとポイントインタイムの復元を含む自動バックアップが利用不可となることを了承しました。」にチェックをいれる
4. 「delete me」を入力後、削除する

削除するのに時間がかかるため、RDS以外を先に削除する

[ec2]

* インスタンス

webserver#1-自分の名前(**user1**)と**webserver#2**-自分の名前(**user1**)それぞれ削除

1. 選択 -> アクション -> インスタンスの状態 -> 終了
2. インスタンスの状態が「terminated」となれば OK

* Elastic IP アドレス

自分が作成したインスタンスと関連付けている**Elastic IP** アドレスを削除

1. 選択 -> Actions -> Elastic IPアドレスの関連付けの解除
2. その後、もう一度選択して Elastic IPアドレスの関連付けの開放をする

* AMI

「**redmine** 自分の名前(**user1**)」を登録解除

1. 選択 -> アクション -> 登録解除

* ロードバランサー

「**elb-自分の名前(user1)**」を削除

1. 選択 -> アクション -> 削除

[s3]

* バケット

「**redmine-自分の名前(user1)-2021xxxx**」を削除

1. バケット名をクリックし、「files/」を選択 -> 削除
2. 「完全に削除」と入力し、オブジェクトを削除
3. バケット一覧画面に戻り、選択 -> 削除
4. バケット名を入力後、削除

[IAM]

* ユーザー

「**s3access-2021xxxx**」を削除

1. 選択 -> ユーザーの削除
2. チェックボックスをオンにしたあと、削除

[ec2]

* キーペア

本日利用した「**handson-2021xxxx**」キーペアを削除

1. 選択 -> アクション -> 削除
2. 削除を入力後、削除ボタンをクリック

* セキュリティグループ

「**db-自分の名前(user1)**」「**web-自分の名前(user1)**」「**elb-自分の名前(user1)**」の順でそれぞれ削除する

※ 削除されない場合は、時間を置いてください。

1. 選択 -> アクション -> セキュリティグループの削除

* ターゲットグループ

「**elb-自分の名前(user1)**」を削除

1. 選択 -> アクション -> 削除

[VPC]

* VPC

「**handson**-自分の名前(**user1**)」を削除

1. 選択 -> アクション -> 削除

[RDS]

* サブネットグループ

「**db subnet** 自分の名前(**user1**)」を削除

1. データベースが削除されるまで待ちます
2. 選択->削除