aws

ユーザーガイド

Amazon Lightsail

Amazon Lightsail: ユーザーガイド

Table of Contents

Amazon Lightsail とは何ですか?

Amazon Lightsail は、ウェブサイトやウェブアプリケーションを構築する必要があるユーザーにとって、Amazon Web Services (AWS) を使い始める最も簡単な方法です。インスタンス (仮想プライベートサーバー)、コンテナサービス、マネージドデータベース、コンテンツ配信ネットワーク (CDN) ディストリビューション、ロードバランサー、SSD ベースのブロックストレージ、静的 IP アドレス、登録済みドメインの DNS 管理、リソーススナップショット (バックアップ) など、プロジェクトを迅速に開始するために必要なものがすべて含まれており、予測可能な低価格の月額料金で利用できます。

Lightsail ではリサーチ用の Amazon Lightsail も提供しています。Lightsail for Research を使用すると、学者や研究者はこの地域で強力な仮想コンピューターを作成できます。 AWS クラウドこれらの仮想コンピュータには、RStudio や Scilab などの研究用アプリケーションがプリインストールされています。詳細については、『研究用 Amazon Lightsail ユーザーガイド』を参照してください。

トピック

Lightsail の特徴

Lightsail には以下の高レベル機能が備わっています。

インスタンス

Lightsail は、セットアップが簡単で、の性能と信頼性に裏打ちされた仮想プライベートサーバー (インスタンス) を提供しています。 AWSウェブサイト、ウェブアプリケーション、またはプロジェクトを数分で起動し、直感的な Lightsail コンソールまたは API からインスタンスを管理できます。

インスタンスを作成するときは、シンプルなオペレーティングシステム (OS)、事前設定されたアプリケーション、または開発スタック (Windows、Plesk、LAMP WordPress、Nginx など) を使用

します。 click-to-launch すべての Lightsail インスタンスにはファイアウォールが組み込まれており、ソース IP、ポート、プロトコルに基づいてインスタンスへのトラフィックを許可または制限できます。詳細はこちら

コンテナ

クラウド内のコンテナ化されたアプリケーションを実行し、安全にアクセスします。コンテナは、コードとその依存関係をパッケージ化するソフトウェアのスタンダード単位で、アプリケーションが 1 つのコンピューティング環境から別のコンピューティング環境に迅速かつ確実に実行します。詳細はこちら

ロードバランサー

インスタンス間でウェブトラフィックをルーティングすることで、ウェブサイトやアプリケーションがトラフィックの変動に対応し、障害から保護され、シームレスな訪問者体験を提供できるようになります。詳細はこちら

マネージド型データベース

Lightsail では、メモリ、処理、ストレージ、転送の許容量を含むフル設定の MySQL または PostgreSQL データベースプランを提供しています。Lightsail マネージドデータベースを使用すると、データベースを仮想サーバーから独立して簡単にスケーリングしたり、アプリケーションの可用性を向上させたり、クラウドでスタンドアロンデータベースを実行したりできます。詳細はこちら

ブロックストレージとオブジェクトストレージ

Lightsail はブロックストレージとオブジェクトストレージの両方を提供しています。Linux または Windows 仮想サーバー用の可用性の高い SSD ベースのストレージを使用すると、ストレージを迅速かつ簡単に拡張できます。詳細はこちら

Lightsail オブジェクトストレージバケットを使用すると、インターネット上のどこからでも、いつでもオブジェクトを保存および取得できます。静的コンテンツをクラウド上でホストすることもできます。詳細はこちら

CDN ディストリビューション

Lightsail は、Amazon と同じインフラストラクチャ上に構築されたコンテンツ配信ネットワーク (CDN) 配信を可能にします。 CloudFront世界中にプロキシサーバーを設定することで、コンテンツを世界中の視聴者に簡単に配信できます。これにより、ユーザーは地理的に近いウェブサイトにアクセスできるようになり、レイテンシーが減少します。詳細はこちら

AWS サービスへのアクセス

> Lightsail は、インスタンス、マネージドデータベース、ロードバランサーなどの機能を集中的に使用して、簡単に始められるようにしています。しかし、だからといってこれらのオプションに限定されるわけではありません。Amazon VPC AWS ピアリングを通じて、Lightsail プロジェクトを 90 を超える他のサービスの一部と統合できます。詳細はこちら

ライトセイルの詳細については、「Amazon Lightsail」を参照してください。

Lightsail は誰のためのものですか？

Lightsail はすべての人のためのものです。Lightsail インスタンス用のイメージを選択してプロジェクトをすぐに開始できるので、ソフトウェアやフレームワークのインストールにそれほど時間をかける必要はありません。

個人のデベロッパーや趣味で個人的なプロジェクトに取り組んでいる方には、Lightsail が基本的なクラウドリソースのデプロイと管理をお手伝いします。仮想マシンやネットワーキングなどのクラウドサービスの学習または試用に興味がある場合もあるでしょう。Lightsail を使用すると、すぐに使い始めることができます。

Lightsail には、基本オペレーティングシステム、LAMP、LEMP (Nginx)、SQL Server Express などの開発スタック、および Drupal や Magento WordPress などのアプリケーションを含むイメージがあります。各イメージにインストールされているソフトウェアの詳細については、「Lightsail インスタンスイメージの選択」を参照してください。

プロジェクトが大きくなるにつれて、ブロックストレージディスクを追加して Lightsail インスタンスにアタッチできます。これらのインスタンスとディスクのスナップショットを作成すると、それらのスナップショットから新しいインスタンスを簡単に作成できます。VPC をピアリングして、Lightsail インスタンスが Lightsail AWS 以外の他のリソースを使用できるようにすることもできます。

Lightsail ロードバランサーを作成し、ターゲットインスタンスをアタッチして可用性の高いアプリケーションを作成することもできます。暗号化された (HTTPS) トラフィック、セッション永続性、ヘルスチェックなどを処理できるように、ロードバランサーを設定することもできます。

アクセスLightsail

以下のインターフェースで Lightsail リソースを作成および管理できます。

Amazon Lightsail コンソール

Lightsail インスタンスとリソースを作成、管理するためのシンプルなウェブインターフェイス。AWS アカウントにサインアップした場合は、にサインインしてコンソールのホームページから Lightsail を選択すると、Lightsail コンソールにアクセスできます。 AWS Management Console

AWS Command Line Interface

AWS コマンドラインシェルのコマンドを使用してサービスとやり取りできます。Windows、Mac、Linux でサポートされています。 AWS CLI の詳細については、「AWS Command Line Interface ユーザーガイド」を参照してください。Lightsail コマンドは Amazon Lightsail API リファレンスをご覧ください。

AWS Tools for PowerShell

PowerShell によって公開されている機能に基づいて構築されたモジュールセット。 AWS SDK for .NET Tools PowerShell を使用すると、 AWS PowerShell コマンドラインからリソースに対する操作をスクリプト化できます。使用を開始する方法については、『AWS Tools for Windows PowerShell ユーザーガイド』を参照してください。Lightsail のコマンドレットは、「コマンドレットリファレンス」にあります。AWS Tools for PowerShell

Query API

Lightsail にはクエリ API が用意されています。このリクエストは、HTTP 動詞である GET または POST と、`Action` という名前の Query パラメータを使用する HTTP または HTTPS リクエストです。Lightsail の API アクションの詳細については、Amazon Lightsail API リファレンスの「アクション」を参照してください。

AWS SDK

HTTP や HTTPS 経由でリクエストを送信する代わりに、言語固有の API を使用してアプリケーションを構築したい場合は、ライブラリ、サンプルコード、チュートリアル、 AWS その他のリソースをソフトウェア開発者向けに用意しています。これらのライブラリには、リクエストの暗号化署名、リクエストの再試行、エラーレスポンスの処理などのタスクを自動化する基本機能が用意されているので、開発を簡単に始められます。詳細については、「構築に役立つツール」を参照してください。 AWS

Lightsail を使い始めましょう

Lightsail を使用するように設定したら、インスタンスを起動、接続、クリーンアップできます。チュートリアル: Amazon Lightsail インスタンスの使用を開始する

関連サービス

インスタンスやディスクなどの Lightsail リソースは、Lightsail を使用して直接プロビジョニングできます。さらに、 AWS 次のような他のサービスを使用してリソースをプロビジョニングできます。

- 「Amazon EC2」

  ソフトウェアシステムの構築とホストに使用する、サイズ変更可能なコンピューティング容量 (文字通り Amazon のデータセンターのサーバー) を提供します。Lightsail と Amazon EC2 を比較するには、Amazon Lightsail または Amazon EC2 を参照してください。

- Amazon EC2 Auto Scaling

  アプリケーションの負荷を処理するために適切な数の Amazon EC2 インスタンスがあることを確認できます。

- Elastic Load Balancing

  アプリケーションの着信トラフィックを複数の インスタンスに自動的に分散できます。

- Amazon Relational Database Service (Amazon RDS)

  クラウド内でマネージドリレーショナルデータベースを簡単に設定、運用、およびスケールできます。

- Amazon Elastic Container Service (Amazon ECS)

  Amazon EC2 インスタンスのクラスターにコンテナ化されたアプリケーションをデプロイ、管理、スケーリングします。

見積もり、請求、コストの最適化

ユースケースの見積もりを作成するには、を使用してください。 AWS AWS Pricing Calculator

請求を表示するには、AWS Billing and Cost Management コンソールで請求およびコスト管理ダッシュボードに移動します。請求書には、料金の明細が記載された使用状況レポートへのリンクが記載されています。 AWS アカウント請求の詳細については、「AWS Billing and Cost Management ユーザーガイド」を参照してください。

AWS 請求、アカウント、イベントに関して質問がある場合は、AWS Support にお問い合わせください。

を使用して、 AWS 環境のコスト、セキュリティ、パフォーマンスを最適化できますAWS Trusted Advisor。

Amazon Lightsail を使用するために AWS アカウントをセットアップします。

初めてAWS を利用する方は、Amazon Lightsail の使用を始める前に、このページに記載されているセットアップの前提条件を完了させてください。これらのセットアップ手順には、AWS Identity and Access Management (IAM) サービスを使用します。IAM の詳細については、「IAM ユーザーガイド」を参照してください。

トピック

AWS へのサインアップ

AWS アカウントをお持ちでない場合は、以下の手順を実行してアカウントを作成してください。

AWS アカウント にサインアップするには

1. https://portal.aws.amazon.com/billing/signup を開きます。
2. オンラインの手順に従います。

   サインアップ手順の一環として、通話呼び出しを受け取り、電話のキーパッドを用いて検証コードを入力するように求められます。

   AWS アカウント にサインアップすると、AWS アカウントのルートユーザー が作成されます。ルートユーザーには、アカウントのすべての AWS のサービス とリソースへのアクセス権があります。セキュリティのベストプラクティスとして、管理ユーザーに管理アクセスを割り当て、ルートユーザーのみを使用してルートユーザーアクセスが必要なタスクを実行してください。

IAM ユーザーを作成する

管理者ユーザーを作成するには、以下のいずれかのオプションを選択します。

管理者を管理する方法を 1 つ選択します	To	By	以下の操作も可能
IAM Identity Center 内 (推奨)	短期の認証情報を使用して AWS にアクセスします。 これはセキュリティのベストプラクティスと一致しています。ベストプラクティスの詳細については、IAM ユーザーガイドの「IAM でのセキュリティのベストプラクティス」を参照してください。	AWS IAM Identity Center ユーザーガイドの「開始方法」の手順に従います。	AWS Command Line Interface ユーザーガイドの「AWS IAM Identity Center を使用するための AWS CLI の設定」に従って、プログラムによるアクセスを設定します。
IAM 内 (非推奨)	長期認証情報を使用して AWS にアクセスする。	IAM ユーザーガイドの「最初の IAM 管理者のユーザーおよびグループの作成」の手順に従います。	IAM ユーザーガイドの「IAM ユーザーのアクセスキーの管理」に従って、プログラムによるアクセスを設定します。

チュートリアル: Amazon Lightsail インスタンスの使用を開始する

このチュートリアルでは、Amazon Lightsail インスタンスを作成、接続、使用する方法について説明します。Lightsail では、インスタンスは仮想プライベートサーバー (仮想マシンとも呼ばれます) です。で Lightsail インスタンスを作成および管理しますAWS クラウド。インスタンスの作成時に、そこのオペレーティングシステム (OS) が含まれているイメージを選択します。基本 OS だけでなくアプリケーションまたは開発スタックが含まれているインスタンスのイメージを選択することもできます。

このチュートリアルで作成したインスタンスには、インスタンスを作成してから削除するまでの間、使用料がかかります。削除はこのチュートリアルの最後に行う手順です。料金の詳細については、「Lightsail の料金」を参照してください。

トピック

ステップ 1: 前提条件を満たす

を初めてご利用になる場合はAWS、Amazon Lightsail の使用を開始する前に、セットアップの前提条件を完了してください。詳細については、「Amazon Lightsail を使用するために AWS アカウントをセットアップします。」を参照してください。

ステップ 2: インスタンスを作成する

次の手順で説明するように、Lightsail コンソールを使用してインスタンスを作成できます。このチュートリアルは、最初のインスタンスをすばやく起動できるようにすることを目的としています。また、利用可能なアプリケーションとハードウェア プランを調べることをお勧めします。詳細については、「Amazon Lightsail インスタンスイメージを選択してください」を参照してください。

1. Lightsail コンソールにサインインします。
2. ホームページで [インスタンスの作成] を選択します。
3. インスタンスの場所 (AWS リージョン および アベイラビリティゾーン) を選択します。待ち時間を短縮するには、お客様が実際にいる場所に最も近い AWS リージョン を選択してください。

   別の場所でインスタンスを作成するには、[AWS リージョン とアベイラビリティーゾーンの変更] を選択します。
4. アプリケーション ([アプリ + OS]) またはオペレーティングシステム ([OS のみ]) を選択できます。

   Lightsail インスタンスイメージの詳細については、「」を参照してくださいAmazon Lightsail インスタンスイメージを選択してください。
5. インスタンスプランを選択します。

   インスタンスがデュアルスタック (IPv4 および IPv6) を使用するかIPv6-onlyネットワークを使用するかを選択します。Lightsail ブループリントの中には、現時点では IPv6-onlyネットワークをサポートしていないものがあります。IPv6-onlyネットワークをサポートするブループリントについては、「」を参照してくださいAmazon Lightsail インスタンスイメージを選択してください。

   3.50 USD Lightsail プランは 1 か月間 (最大 750 時間) 無料で試すことができます。1 か月の無料期間分はアカウントに返金されます。詳細については、Lightsail の料金ページを参照してください。
6. インスタンスの名前を入力します。

   リソース名:

   - AWS リージョン Lightsail アカウントの各 内で一意である必要があります。
   - 2〜255 文字を使用する必要があります。

- 先頭と末尾は英数字または数字を使用する必要があります。
- 英数字、数字、ピリオド、ダッシュ、アンダースコアを使用することができます。

7. [インスタンスの作成] を選択します。

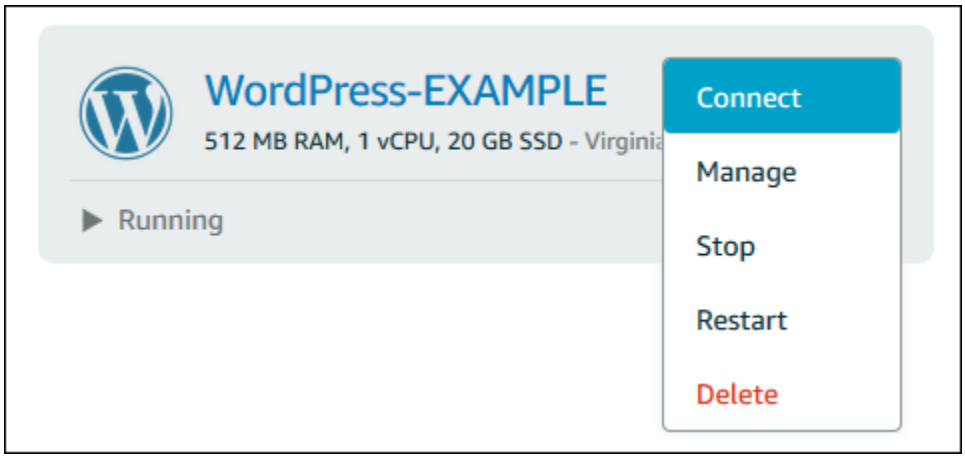
数分以内に Lightsail インスタンスの準備が完了し、接続できます。

ステップ 3: インスタンスに接続する

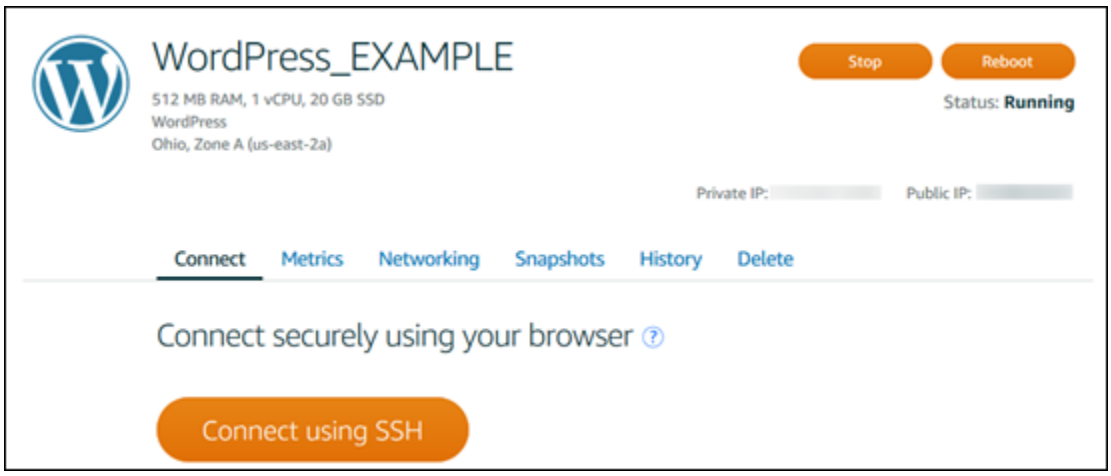
1.

> **Note**
>
> Lightsail ブラウザベースの SSH/RDP クライアントは IPv4 トラフィックのみを受け入れます。サードパーティーのクライアントを使用して、IPv6 経由でインスタンスに SSH または RDP 接続します。詳細については、「インスタンスに接続します」を参照してください。

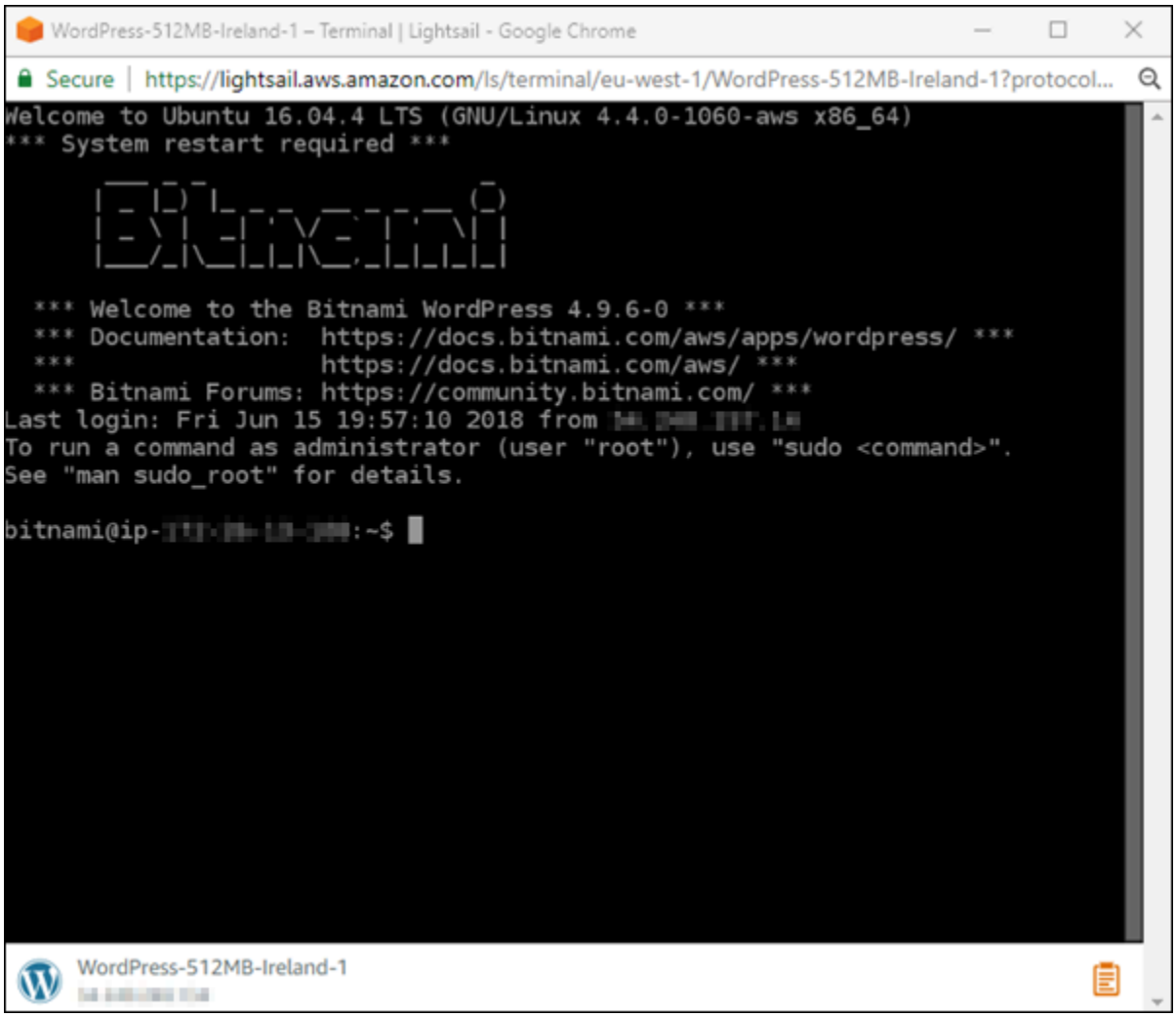
Lightsail ホームページで、インスタンス名の右側にあるメニューを選択し、接続を選択します。



または、インスタンス管理ページを開き、[接続] タブを選択することもできます。



2. SSH クライアントを設定せずに、ターミナルにコマンドを入力して Lightsail インスタンスを管理できるようになりました。



仮想コンピュータに接続してストレージを追加する方法については、このチュートリアルの次のステップに進みます。

ステップ 4: インスタンスにストレージを追加する

Lightsail には、インスタンスにアタッチできるブロックレベルのストレージボリューム (ディスク) が用意されています。インスタンスにはシステムディスクが付属していますが、ニーズの変化に応じて追加のストレージディスクをアタッチできます。インスタンスからディスクをデタッチし、別のインスタンスにアタッチすることもできます。

追加のディスクを作成したら、Lightsail インスタンスに接続してディスクをフォーマットしてマウントする必要があります。

ディスクの作成、アタッチおよび管理の詳細については、「Lightsail ブロックストレージディスクを作成して Linux ベースのインスタンスにアタッチする」を参照してください。

このチュートリアルの次の手順で、仮想コンピューターのバックアップについて説明します。

ステップ 5: スナップショットを作成する

スナップショットはデータ point-in-time のコピーです。インスタンスのスナップショットを作成し、新しいインスタンスを作成したり、データをバックアップしたりするためのベースラインとして使用できます。スナップショットには、インスタンスの復元に必要なすべてのデータ (スナップショットが作成された時点からのデータ) が含まれます。

スナップショットの作成と管理に関する詳細は、「Linux または Unix Lightsail インスタンスのスナップショットを作成する」を参照してください。

このチュートリアルの次の手順で、仮想コンピューターリソースのクリーンアップについて説明します。

ステップ 6: クリーンアップする

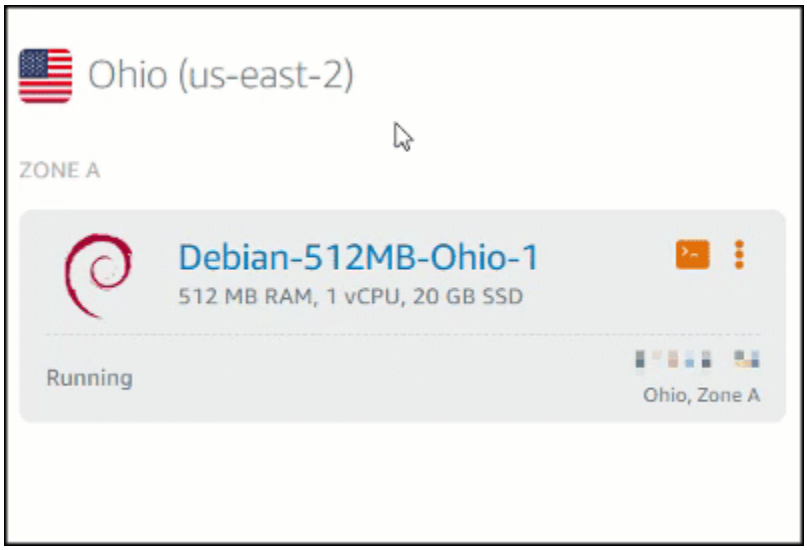
このチュートリアル用に作成したインスタンスを使用して操作した後に、削除することができます。これにより、不要になったインスタンスに対する料金は発生しなくなります。

インスタンスを削除しても、それに関連するスナップショットやアタッチされたディスクは削除されません。このチュートリアル用にスナップショットとディスクを作成した場合は、それらも削除する必要があります。

後のためにインスタンスを保存したいが料金を発生させたくない場合は、そのインスタンスを削除する代わりに停止することができます。後でそのインスタンスを再起動できます。料金の詳細については、「Lightsail 料金表」を参照してください。

> **Important**
>
> Lightsail リソースの削除は永続的なアクションです。削除されたデータは復元できません。後でデータが必要になるかもしれない場合は、削除する前に仮想コンピュータのスナップショットを作成してください。詳細については、「Linux または Unix Lightsail インスタンスのスナップショットを作成する」を参照してください。

1. Lightsail コンソールにサインインします。
2. ナビゲーションペインで、[Instances (インスタンス)] を選択します。
3. 削除するインスタンスのアクションメニューアイコン (⋮) を選択し、[削除] を選択します。



4. [はい、削除します] を選択して削除を確定します。

次のステップ

以下のトピックを使用して、Amazon Lightsail Linux および Windows ベースのインスタンスの使用を開始します。

- Amazon Lightsail で Linux/Unix ベースのインスタンスの使用を開始する
- Amazon Lightsail で Windows Server ベースのインスタンスの使用を開始する

Amazon Lightsail で Linux/Unix ベースのインスタンスの使用を開始する

Linux/Unix ベースの Lightsail インスタンス (仮想プライベートサーバー) は、 などのアプリケーション WordPress や LAMP などの開発スタックを数秒で実行できます。インスタンスの実行が開始されると、Lightsail を離れることなく SSH 経由でインスタンスに接続できます。その方法は次のとおりです。

Windows ベースのインスタンスを作成するには、「Amazon Lightsail での Windows ベースのインスタンスの開始方法」を参照してください。

Linux ベースのインスタンスを作成する

1. ホームページで [インスタンスの作成] を選択します。
2. インスタンスの場所 (AWS リージョン およびアベイラビリティーゾーン) を選択します。

   変更 AWS リージョン とアベイラビリティーゾーンを選択して、別の場所にインスタンスを作成します。
3. 必要に応じて、アベイラビリティーゾーンを変更できます。

   [アベイラビリティーゾーンの変更] を選択する。
4. Linux プラットフォームを選択します。
5. アプリケーション ([アプリ + OS]) またはオペレーティングシステム ([OS のみ]) を選択します。

   Lightsail インスタンスイメージの詳細については、「Amazon Lightsail インスタンスイメージの選択」を参照してください。
6. インスタンスプランを選択します。

   インスタンスがデュアルスタック (IPv4 および IPv6) を使用するかIPv6-onlyネットワークを使用するかを選択します。現時点では、一部の Lightsail ブループリントは IPv6-onlyネットワークをサポートしていません。IPv6-onlyネットワークをサポートするブループリントについては、「」を参照してくださいAmazon Lightsail インスタンスイメージを選択してください。

   3.50 USD Lightsail プランは 1 か月間 (最大 750 時間) 無料で試すことができます。1 か月の無料期間分はアカウントに返金されます。詳細については、Lightsail の料金ページを参照してください。

> **Note**
>
> AWS 無料利用枠の一部として、一部のインスタンスバンドルで Amazon Lightsail を無料で使い始めることができます。詳細については、「Amazon Lightsail 料金表」ページのAWS 「 無料利用枠」を参照してください。

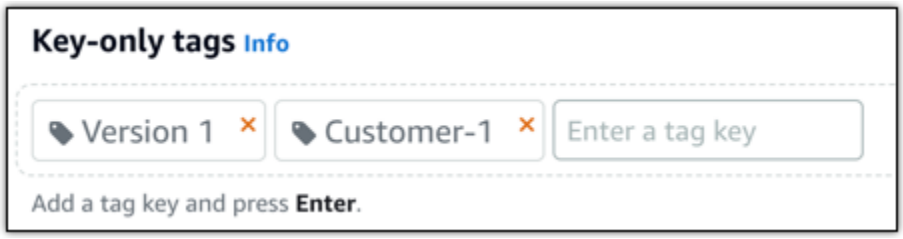
7. インスタンスの名前を入力します。

   リソース名:

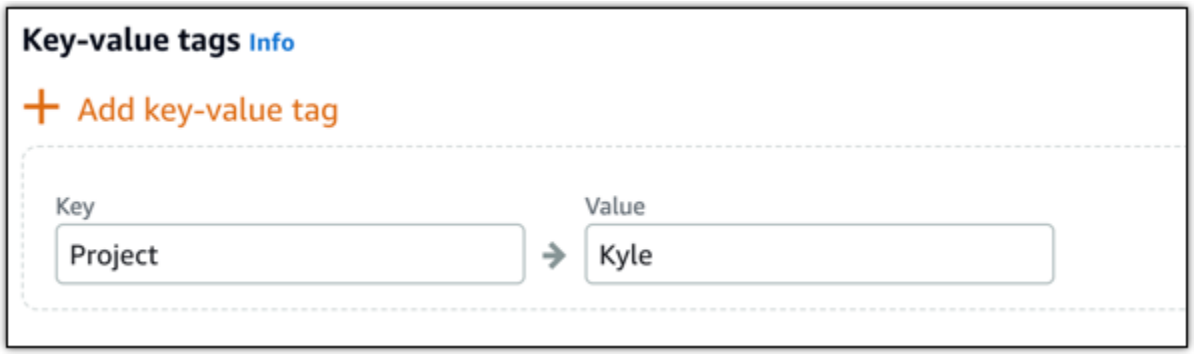
   - AWS リージョン Lightsail アカウントの各 内で一意である必要があります。
   - 2〜255 文字を使用する必要があります。
   - 先頭と末尾は英数字または数字を使用する必要があります。
   - 英数字、数字、ピリオド、ダッシュ、アンダースコアを使用することができます。

8. 以下のいずれかのオプションを選択して、インスタンスにタグを追加します。

   - [キーオンリータグの追加]。タグキーのテキストボックスに新しいタグを入力し、Enter キーを押します。X を選択して、残したくないタグをすべて削除します。

   - [key-value タグの作成] から [キー] テキストボックスにキーを入力し、[値] テキストボックスに値を入力します。キーと値のタグは、一度に 1 つのみ追加できます。キー値タグを追加するには [キー値タグの追加] を選択し、残したくないタグを削除するには [X] を選択します。

> **Note**
>
> 「キーのみ」のタグと「キーバリュー」のタグの詳細については、「タグ」を参照してください。

9. [インスタンスの作成] を選択します。

   高度な作成オプションについては、「起動スクリプトを使用して起動時に Amazon Lightsail インスタンスを設定する」または「Linux/Unix ベースの Lightsail インスタンスの SSH を設定する」を参照してください。

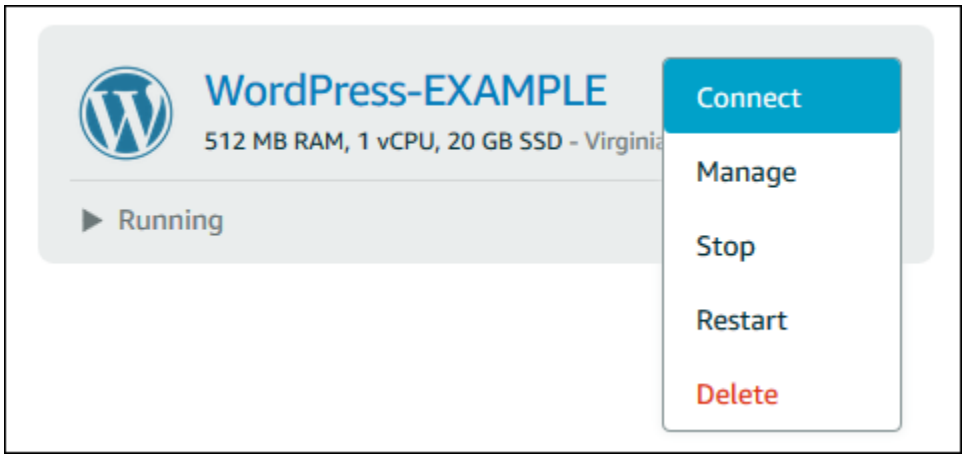
数分以内に Lightsail インスタンスの準備が完了し、Lightsail を離れることなく SSH 経由で接続できます。

インスタンスへの接続

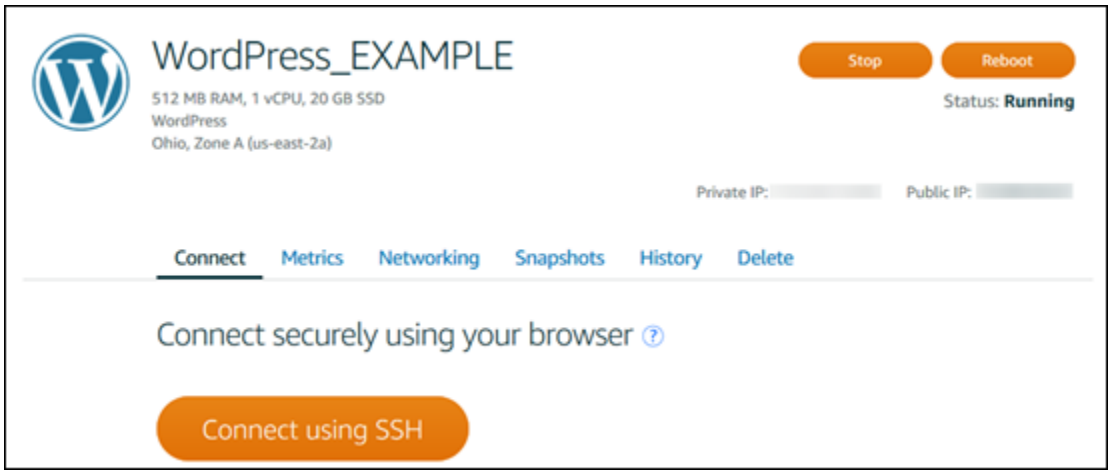
1.

> **Note**
>
> Lightsail ブラウザベースの SSH/RDP クライアントは IPv4 トラフィックのみを受け入れます。サードパーティーのクライアントを使用して、IPv6 経由でインスタンスに SSH または RDP 接続します。詳細については、「インスタンスに接続します」を参照してください。

   Lightsail ホームページで、インスタンス名の右側にあるメニューを選択し、接続を選択します。

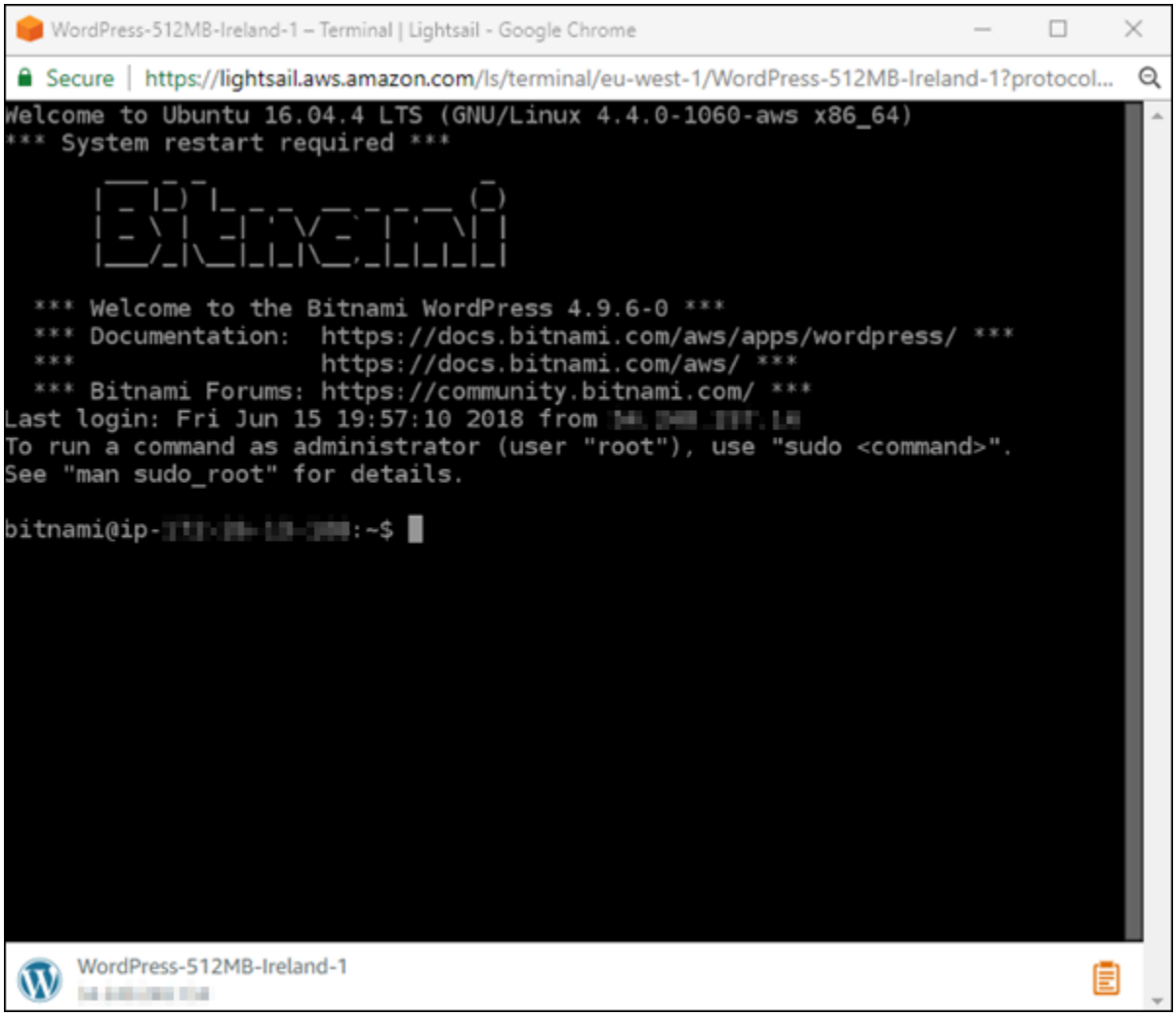


または、インスタンス管理ページを開き、[接続] タブを選択することもできます。



> **Note**
>
> PuTTY などの SSH クライアントを使用してインスタンスに接続するには、「PuTTY をセットアップして Lightsail インスタンスに接続する」の手順に従います。

2. これで、SSH クライアントを設定せずに、ターミナルにコマンドを入力して Lightsail インスタンスを管理できます。



次のステップ

インスタンスに接続できるようになった後、次に行う作業はお客様の使用計画によって異なります。例:

- the section called “WordPress” ブログを作成する場合は 。
- Lightsail インスタンスを再起動するたびに同じ IP アドレスを保持するには、インスタンスの静的 IP アドレスを作成します。
- バックアップとしてインスタンスのスナップショットを作成します。

Amazon Lightsail で Windows Server ベースのインスタンスの使用を開始する

Windows Server オペレーティングシステム (OS) を実行する Lightsail インスタンスを作成できます。3 つの OS ブループリント (Windows Server 2022、Windows Server 2019、Windows Server 2016) を利用できます。また、SQL Server Express 2022、2019、2016 では、あらかじめ設定されたブループリントが用意されています。

このトピックでは、ソフトウェアの選択、Windows Server ベースのインスタンスの作成、それに接続する方法について説明します。

AWS での Windows Server の詳細

Windows Server ベースのインスタンスを選択する

Lightsail で Windows Server ベースのインスタンスを作成するには、3 つのオプションがあります。

[Windows Server 2022]

Windows Server を実行する Lightsail は、Microsoft Web Platform を使用してアプリケーションをデプロイするための高速で信頼性の高い環境です。Lightsail を使用すると、高性能で信頼性が高く、費用対効果の高い AWS クラウド コンピューティングプラットフォームで、互換性のある任意の Windows ベースのソリューションを実行できます。一般的な Windows ユースケースには、Enterprise Windows ベースのアプリケーションホスティング、ウェブサイトおよびウェブサービスホスティング、データ処理、分散テスト、ASP.NET アプリケーションホスティング、そして Windows ソフトウェアを必要とする、他のすべてのアプリケーションが含まれます。

Windows Server 2022 イメージの詳細について説明します

Windows Server 2019

何らかの理由で Windows Server 2012 R2 および Windows Server 2016 を実行する必要がある場合を除き、Windows Server 2019 の最新バージョンを使用することをお勧めします。

Windows Server を実行する Lightsail は、Microsoft Web Platform を使用してアプリケーションをデプロイするための高速で信頼性の高い環境です。Lightsail を使用すると、AWS の高性能で信頼性が高く、費用対効果の高いクラウドコンピューティングプラットフォーム上で、互換性のある任意の Windows ベースのソリューションを実行できます。一般的な Windows ユースケースには、Enterprise Windows ベースのアプリケーションホスティング、ウェブサイトおよびウェ

ブサービスホスティング、データ処理、分散テスト、ASP.NET アプリケーションホスティング、Windows ソフトウェアが必要な他のあらゆるアプリケーションが含まれます。

Windows Server 2019 イメージの詳細

Windows Server 2016

Windows Server を実行する Lightsail は、Microsoft Web Platform を使用してアプリケーションをデプロイするための高速で信頼性の高い環境です。Lightsail を使用すると、AWS の高性能で信頼性が高く、費用対効果の高いクラウドコンピューティングプラットフォーム上で、互換性のある任意の Windows ベースのソリューションを実行できます。一般的な Windows ユースケースには、Enterprise Windows ベースのアプリケーションホスティング、ウェブサイトおよびウェブサービスホスティング、データ処理、分散テスト、ASP.NET アプリケーションホスティング、Windows ソフトウェアが必要な他のあらゆるアプリケーションが含まれます。

Windows Server 2016 イメージの詳細

[SQL Server Express 2022]

SQL Server Express は、無料でダウンロード、配布、使用できるリレーショナルデータベース管理システムです。小規模な埋め込みアプリケーションを特にターゲットとするデータベースを構成しています。この Lightsail イメージは、Windows Server 2022 のベース OS で実行されます。

SQL Server Express 2022 イメージの詳細

[SQL Server Express 2019]

SQL Server Express は、無料でダウンロード、配布、使用できるリレーショナルデータベース管理システムです。小規模な埋め込みアプリケーションを特にターゲットとするデータベースを構成しています。この Lightsail イメージは、Windows Server 2022 のベース OS で実行されます。

SQL Server Express 2019 イメージの詳細

[SQL Server Express 2016]

SQL Server Express は、無料でダウンロード、配布、使用できるリレーショナルデータベース管理システムです。小規模な埋め込みアプリケーションを特にターゲットとするデータベースを構成しています。この Lightsail イメージは、Windows Server 2016 のベース OS で実行されます。

SQL Server Express イメージの詳細

Windows Server ベースの インスタンスを作成する

Windows Server ベースのインスタンスは、Lightsail コンソールまたは AWS Command Line Interface () を使用して作成できますAWS CLI。

コンソールを使用してインスタンスを作成するには

1. Lightsail にサインインし、ホームページに移動します。
2. [インスタンスの作成] を選択します。
3. Windows Server ベースの Lightsail インスタンスを作成する AWS リージョン を選択します。

   例えば `Ohio (us-east-2)` です。
4. [Microsoft Windows] プラットフォームを選択します。
5. Windows Server 2022、Windows Server 2019、Windows Server 2016 のブループリントを選択するには、[OS のみ] を選択します。

   SQL Server Express 設計図を選択するには、[アプリ + OS] を選択します。
6. インスタンスプランを選択します。

   インスタンスがデュアルスタック (IPv4 および IPv6) を使用するかIPv6-onlyネットワークを使用するかを選択します。現時点では、一部の Lightsail ブループリントは IPv6-onlyネットワークをサポートしていません。IPv6-onlyネットワークをサポートするブループリントについては、「」を参照してくださいAmazon Lightsail インスタンスイメージを選択してください。

   プランには、低コストで予測可能なコストとマシン設定 (RAM、SSD、vCPU)、データ転送も含まれます。

   > **Note**
   >
   > 設計図によっては、一部のインスタンスプランを使用できません。たとえば、SQL Server Express 設計図には 2 つの最小プランを使用できません。少なくとも、2 GB RAM および 50 GB SSD を含むプランを使用するか、より大きなプランのいずれかを選択する必要があります。
7. インスタンスの名前を入力します。

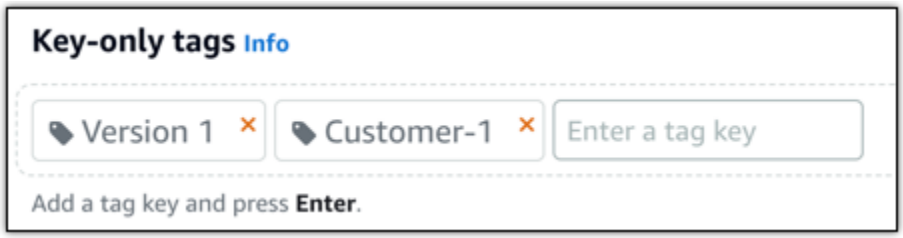
   リソース名:

   - AWS リージョン Lightsail アカウントの各 内で一意である必要があります。

- 2〜255 文字を使用する必要があります。
- 先頭と末尾は英数字または数字を使用する必要があります。
- 英数字、数字、ピリオド、ダッシュ、アンダースコアを使用することができます。

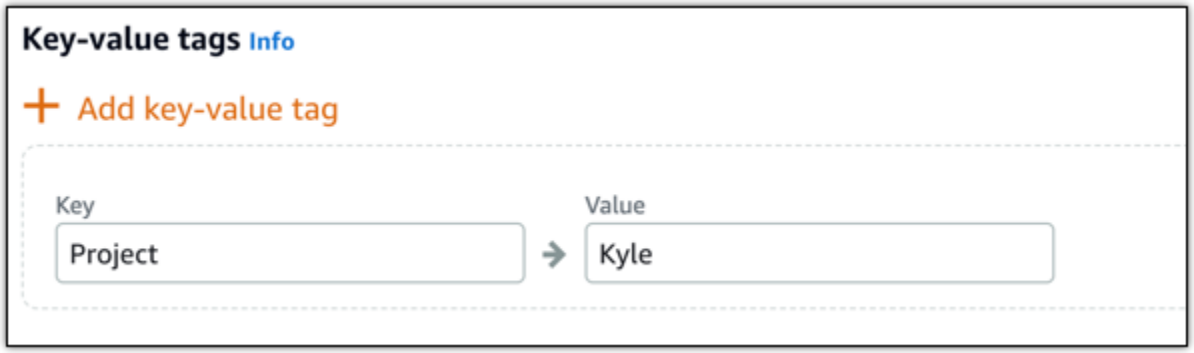
8. 以下のいずれかのオプションを選択して、インスタンスにタグを追加します。

   - [Add key-only tags] (キーのみのタグを追加) または [Edit key-only tags] (キーのみのタグを編集) (タグが追加済みの場合)を追加。タグキーのテキストボックスに新しいタグを入力し、Enter キーを押します。タグの入力を完了したら、[保存] を選択してタグを追加し、追加しない場合は、[キャンセル] を選択します。

   - [key-value タグの作成] から [キー] テキストボックスにキーを入入力し、[値] テキストボックスに値を入力します。タグの入力を完了したら、[保存] を選択し、追加しない場合は、[キャンセル] を選択します。

     キーバリューのタグは、保存する際に一つずつ追加することができます。さらに key-value タグを追加するには、以上のステップを繰り返します。

> **Note**
>
> 「キーのみ」のタグと「キーバリュー」のタグの詳細については、「タグ」を参照してください。

9. [インスタンスの作成] を選択します。

を使用してインスタンスを作成するには AWS CLI

1. まだ AWS CLIをインストールして設定していない場合は、インストールして設定します。

   詳細については、「Amazon Lightsail と連携 AWS Command Line Interface するように を設定する」を参照してください。
2. コマンドプロンプトまたはターミナルウィンドウを開きます。
3. まだ設定していない場合は、 AWS CLI を使用して を設定しaws configure、Lightsail リソースを作成する AWS リージョン を選択します。
4. オハイオリージョンで実行 AWS CLI されている Windows Server 2016 インスタンスを作成するには、次のコマンドを入力します。

   ```
   aws lightsail create-instances --instance-names InstanceName --availability-
   zone us-east-2a --blueprint-id windows_server_2016_2017_09_13 --bundle-id
    medium_win_1_0
   ```

   コマンドで、 を新しいインスタンスの名前*InstanceName*に置き換えます。

   成功すると、 AWS CLIから次の出力が表示されます。

   ```
   {
       "operations": [
           {
               "status": "Started",
               "resourceType": "Instance",
               "isTerminal": false,
               "statusChangedAt": 1508086226.4,
               "location": {
                   "availabilityZone": "us-east-2a",
                   "regionName": "us-east-2"
               },
               "operationType": "CreateInstance",
               "resourceName": "my-windows-instance",
               "id": "344acdc8-f9c4-4eda-8232-12345EXAMPLE",
               "createdAt": 1508086225.467
           }
       ]
   }
   ```

> **Note**
>
> 使用可能な設計図の一覧を取得するには、get-blueprints コマンドを使用します。使用可能なバンドルの一覧を取得するには、get-bundles コマンドを使用します。get-instance-access-details コマンドを使用したインスタンスのパスワードの取得の詳細について説明します。

インスタンスへの接続

Windows Server ベースの Lightsail インスタンスを作成したら、ブラウザベースの RDP クライアントまたは任意のリモートデスクトップクライアントを使用してインスタンスに接続できます。

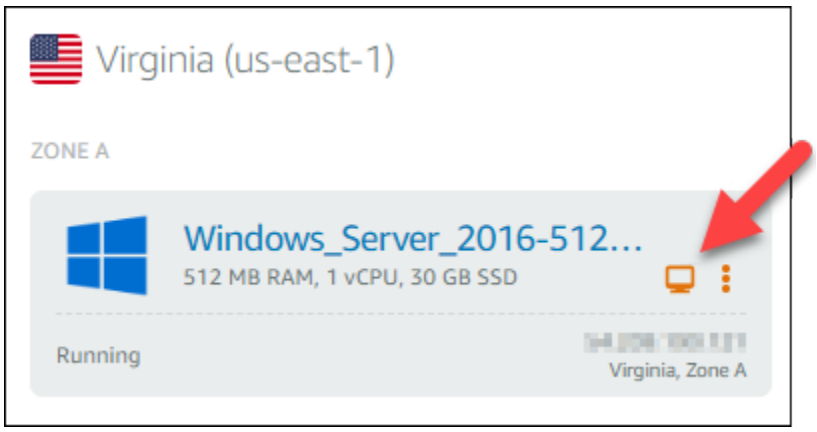
> **Note**
>
> インスタンスを作成した後、インスタンスに接続できるようになるまで最大 15 分かかります。

Lightsail ブラウザベースの RDP クライアントを使用して接続するには

1.

> **Note**
>
> Lightsail ブラウザベースの SSH/RDP クライアントは IPv4 トラフィックのみを受け入れます。サードパーティーのクライアントを使用して、IPv6 経由でインスタンスに SSH または RDP 接続します。詳細については、「インスタンスに接続します」を参照してください。

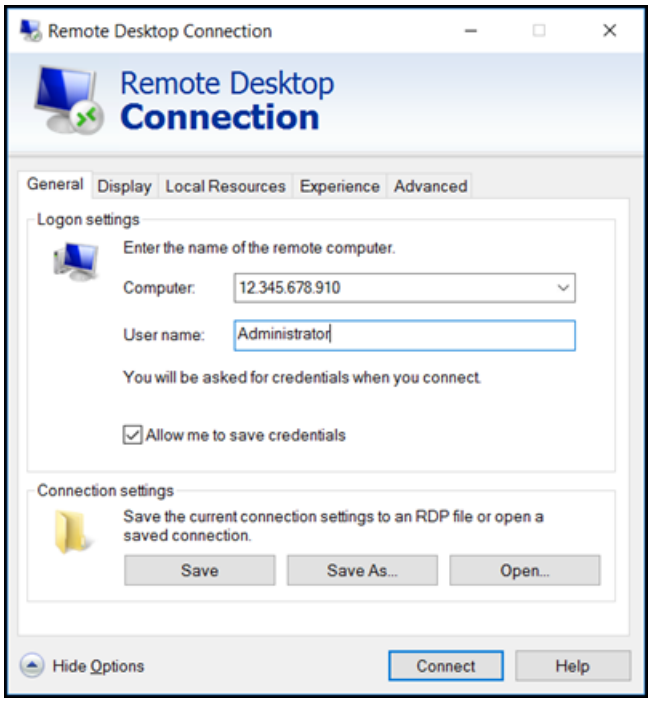
ホームページで、インスタンスの横の [RDP を使用して接続] アイコンを選択します。



2. または、ショートカットメニューやインスタンス管理ページからインスタンスに接続することもできます。

独自の RDP クライアントを使用して接続するには

1. IP アドレスを取得するには、Lightsail のホームページに移動します。
2. IP アドレスをクリップボードにコピーします。
3. Windows のリモートデスクトップ接続などの RDP クライアントを開きます。
4. IP アドレスを [コンピューター] フィールドに貼り付けます。
5. [オプションの表示] を選択し、[Administratorユーザー名] に「」と入力します。

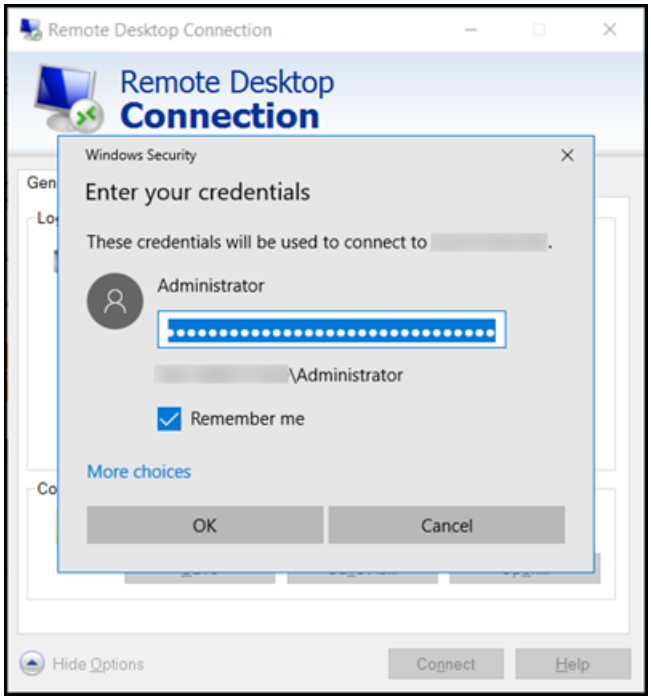


6. [接続]を選択します。

7. パスワードを取得するには、Lightsail のインスタンス管理ページに移動します。

   Lightsail ホームページでインスタンスの名前を選択 (またはショートカットメニューから管理を選択) することで、インスタンス管理ページにアクセスできます。

8. [デフォルトのパスワードを表示] を選択します。
9. デフォルトパスワードをクリップボードにコピーします。
10. パスワードを [リモートデスクトップ接続] に貼り付け、[このアカウントを記憶する] を選択して今後はこのダイアログボックスが表示されないようにします。



11. [OK] をクリックします。
12. [Don't ask me again for connections to this computer (このコンピューターでは接続の確認をしない)] を選択し、[Yes (はい)] を選択します。

Amazon Lightsail のインスタンス (仮想プライベートサーバー)

Lightsail インスタンスは仮想プライベートサーバー (仮想マシン とも呼ばれます) です。インスタンスの作成時に、オペレーティングシステム (OS) が含まれているイメージを選択します。基本 OS だけでなくアプリケーションまたは開発スタックが含まれているインスタンスのイメージを選択することもできます。

オペレーティングシステム、アプリケーション、および開発フレームワークの完全なリストについては、「Lightsail インスタンスイメージの選択」を参照してください。

インスタンスの詳細については、次のトピックを参照してください。

トピック

- Lightsail インスタンスを作成する
- Lightsail インスタンスを削除する
- Amazon Lightsail インスタンスイメージを選択してください
- Lightsail IPv6-onlyインスタンスプラン
- Lightsail の SSH キーペア
- Linux または Unix Lightsail インスタンスのスナップショットを作成する
- Lightsail インスタンスを管理する
- Lightsail ファイアウォールルールリファレンス
- Lightsail におけるインスタンスメタデータサービス (IMDS) とユーザーデータ

Lightsail インスタンスを作成する

Lightsail インスタンスは、仮想プライベートサーバー (VPS) とも呼ばれ、 のようなアプリケーション WordPress や LAMP のような開発スタックを数秒で実行できます。インスタンスの実行が開始されると、Lightsail を離れることなく SSH 経由でインスタンスに接続できます。その方法は次のとおりです。

1. ホームページで [インスタンスの作成] を選択します。
2. インスタンスの場所 (AWS リージョン および アベイラビリティゾーン) を選択します。

別の場所でインスタンスを作成するには、[AWS リージョン とアベイラビリティーゾーンの変更] を選択します。

3. 必要に応じて、アベイラビリティーゾーンを変更できます。

   ドロップダウンリストからアベイラビリティーゾーンを選択します。

4. アプリケーション ([アプリ + OS]) またはオペレーティングシステム ([OS のみ]) を選択します。

   Lightsail インスタンスイメージの詳細については、「Amazon Lightsail インスタンスイメージの選択」を参照してください。

5. インスタンスプランを選択します。

   インスタンスがデュアルスタック (IPv4 および IPv6) を使用するかIPv6-onlyネットワークを使用するかを選択します。現時点では、一部の Lightsail ブループリントは IPv6-onlyネットワークをサポートしていません。IPv6-onlyネットワークをサポートするブループリントについては、「」を参照してくださいAmazon Lightsail インスタンスイメージを選択してください。

   3.50 USD Lightsail プランは 1 か月間 (最大 750 時間) 無料で試すことができます。1 か月の無料期間分はアカウントに返金されます。詳細については、Lightsail の料金ページを参照してください。

   > **Note**
   >
   > AWS 無料利用枠の一部として、選択したインスタンスバンドルで Amazon Lightsail を無料で使い始めることができます。詳細については、「Amazon Lightsail 料金表」ページのAWS「 無料利用枠」を参照してください。

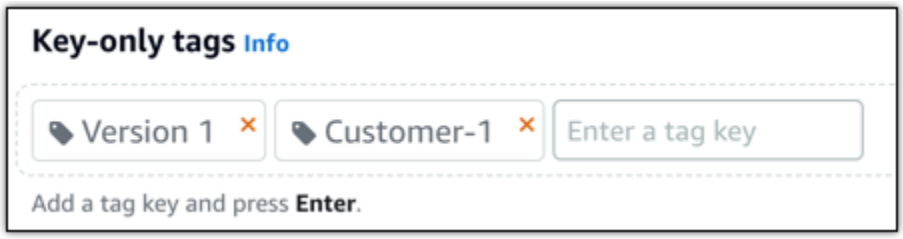
6. インスタンスの名前を入力します。

   リソース名:

   - AWS リージョン Lightsail アカウントの各 内で一意である必要があります。
   - 2〜255 文字を使用する必要があります。
   - 先頭と末尾は英数字または数字を使用する必要があります。
   - 英数字、数字、ピリオド、ダッシュ、アンダースコアを使用することができます。

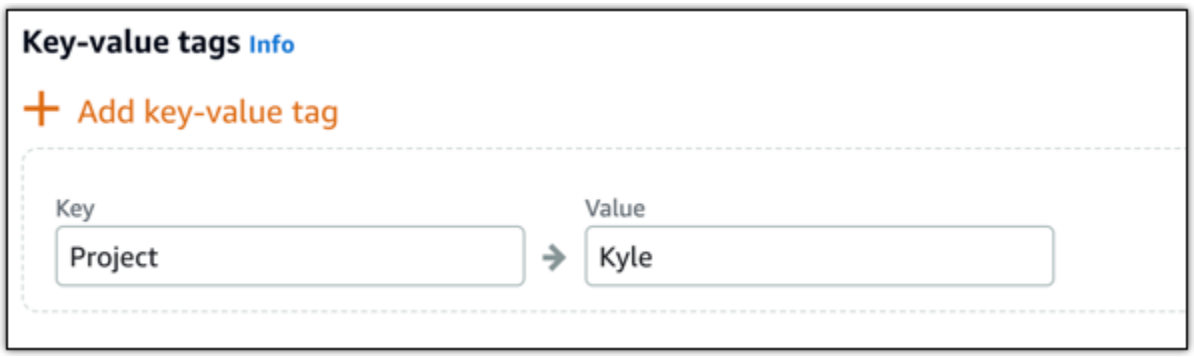
7. 以下のいずれかのオプションを選択して、インスタンスにタグを追加します。

- [Add key-only tags] (キーのみのタグを追加) または [Edit key-only tags] (キーのみのタグを編集) (タグが追加済みの場合)を追加。タグキーのテキストボックスに新しいタグを入力し、Enter キーを押します。タグの入力を完了したら、[保存] を選択してタグを追加し、追加しない場合は、[キャンセル] を選択します。



- [key-value タグの作成] から [キー] テキストボックスにキーを入力し、[値] テキストボックスに値を入力します。タグの入力を完了したら、[保存] を選択し、追加しない場合は、[キャンセル] を選択します。

  キーバリューのタグは、保存する際に一つずつ追加することができます。さらに key-value タグを追加するには、以上のステップを繰り返します。



> **Note**
>
> 「キーのみ」のタグと「キーバリュー」のタグの詳細については、「タグ」を参照してください。

8. [インスタンスの作成] を選択します。

   高度な作成オプションについては、「起動スクリプトを使用して起動時に Amazon Lightsail インスタンスを設定する」または「Linux/Unix ベースのインスタンスの SSH を設定する」を参照してください。

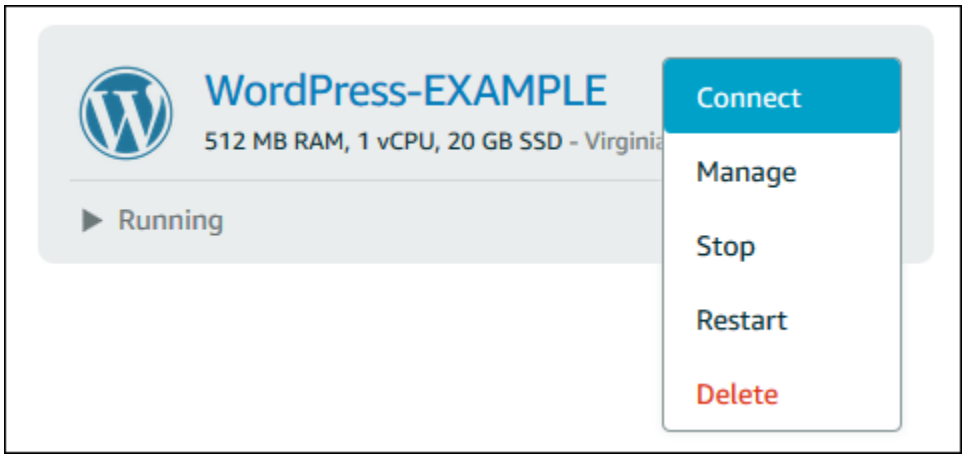
数分以内に Lightsail インスタンスの準備が完了し、Lightsail を離れることなく SSH 経由で接続できます。

インスタンスへの接続方法

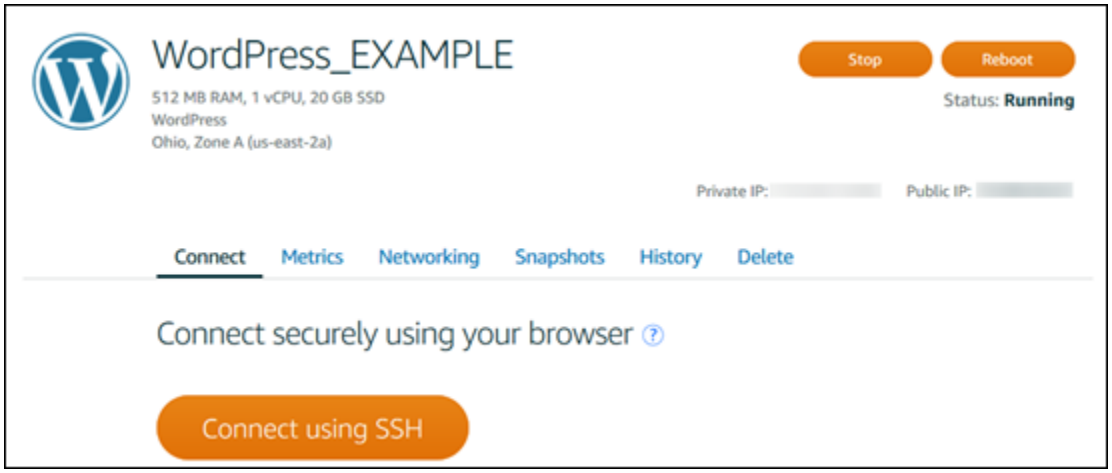
1.

> **Note**
>
> Lightsail ブラウザベースの SSH/RDP クライアントは、IPv4 トラフィックのみを受け入れます。サードパーティーのクライアントを使用して、IPv6 経由でインスタンスに SSH または RDP 接続します。詳細については、「インスタンスに接続します」を参照してください。

Lightsail ホームページで、インスタンス名の右側にあるメニューを選択し、接続を選択します。

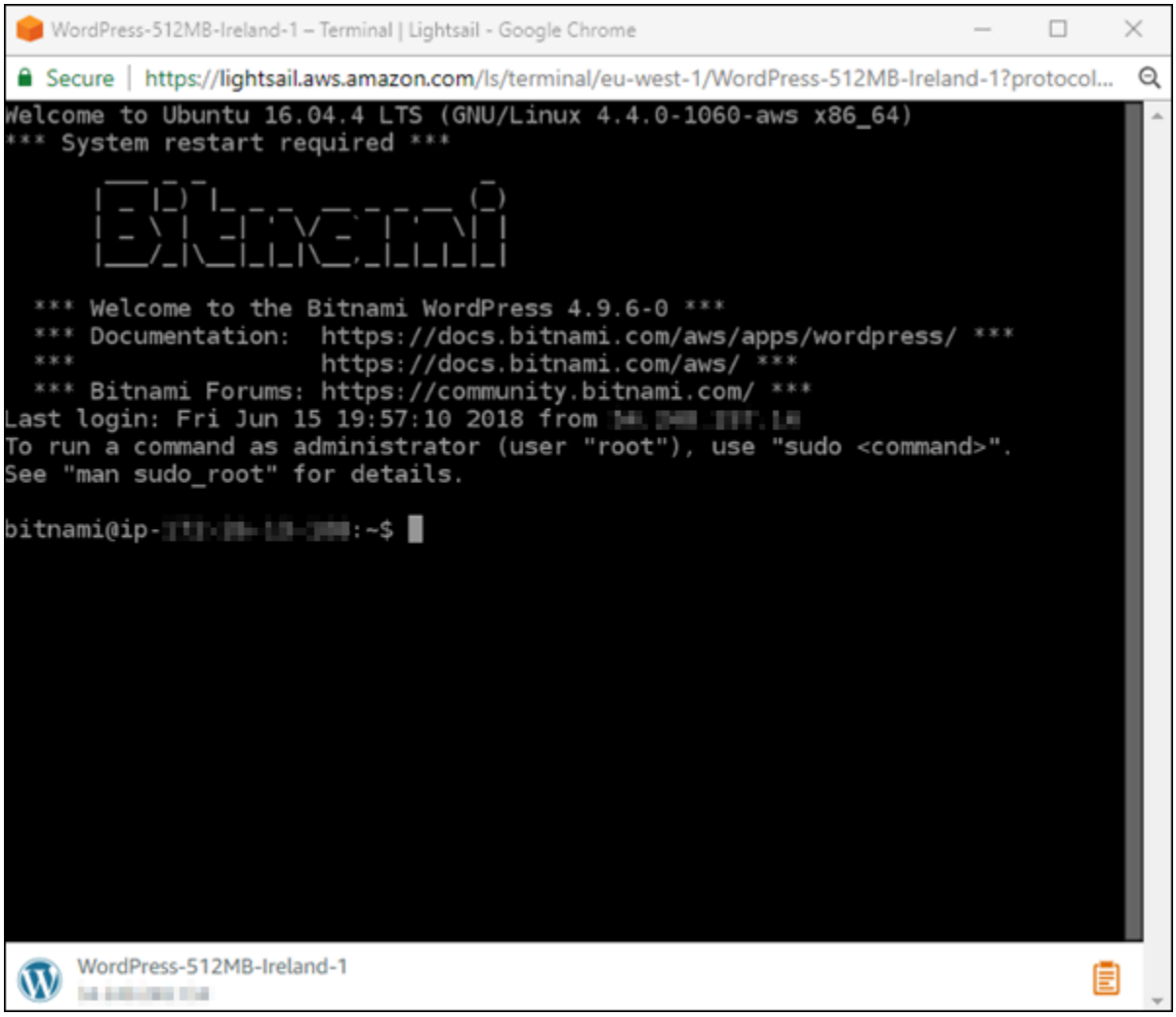


または、インスタンス管理ページを開き、[接続] タブを選択することもできます。



> **Note**
>
> PuTTY などの SSH クライアントを使用してインスタンスに接続する場合は、「PuTTY をセットアップして Lightsail インスタンスに接続する」の手順に従います。

2. SSH クライアントを設定せずに、ターミナルにコマンドを入力して Lightsail インスタンスを管理できるようになりました。



次のステップ

インスタンスに接続できるようになった後、次に行う作業はお客様の使用計画によって異なります。例:

- the section called “WordPress” ブログを作成する場合は 。

- Lightsail インスタンスを再起動するたびに同じ IP アドレスを保持するには、インスタンスの静的 IP アドレスを作成します。
- バックアップとしてインスタンスのスナップショットを作成します。

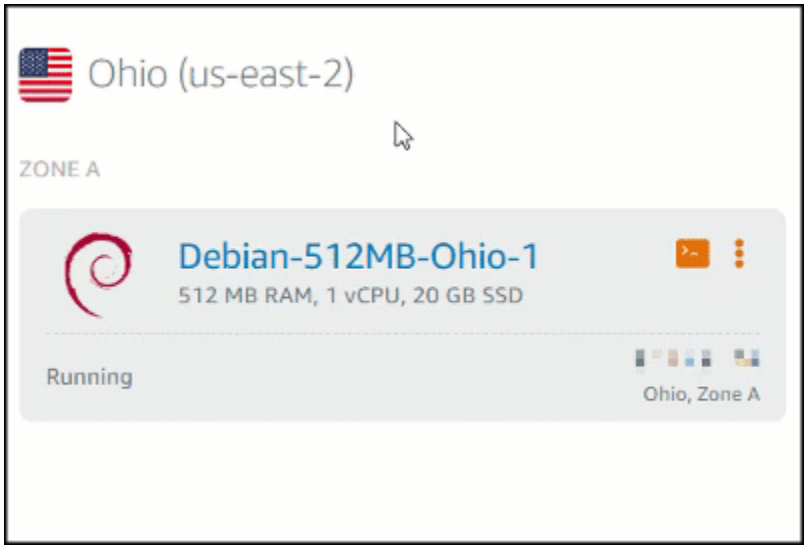
Lightsail インスタンスを削除する

インスタンスが不要になった場合は、Amazon Lightsail コンソールまたは AWS Command Line Interface (AWS CLI) を使用して削除できます。インスタンスを削除すると、インスタンスに対する課金も停止します。ただし、削除したインスタンスにアタッチされていたリソース (静的 IP やスナップショットなど) に対しては、これらを削除するまで料金が発生します。

> **Note**
>
> 削除したインスタンスは復旧できません。インスタンスのデータが後で必要になった場合に備えて、削除する前にインスタンスのスナップショットを作成します。詳細については、「Linux または Unix インスタンスのスナップショットを作成する」または「Windows Server インスタンスのスナップショットを作成する」を参照してください。

Lightsail コンソールのホームページでインスタンスを削除する

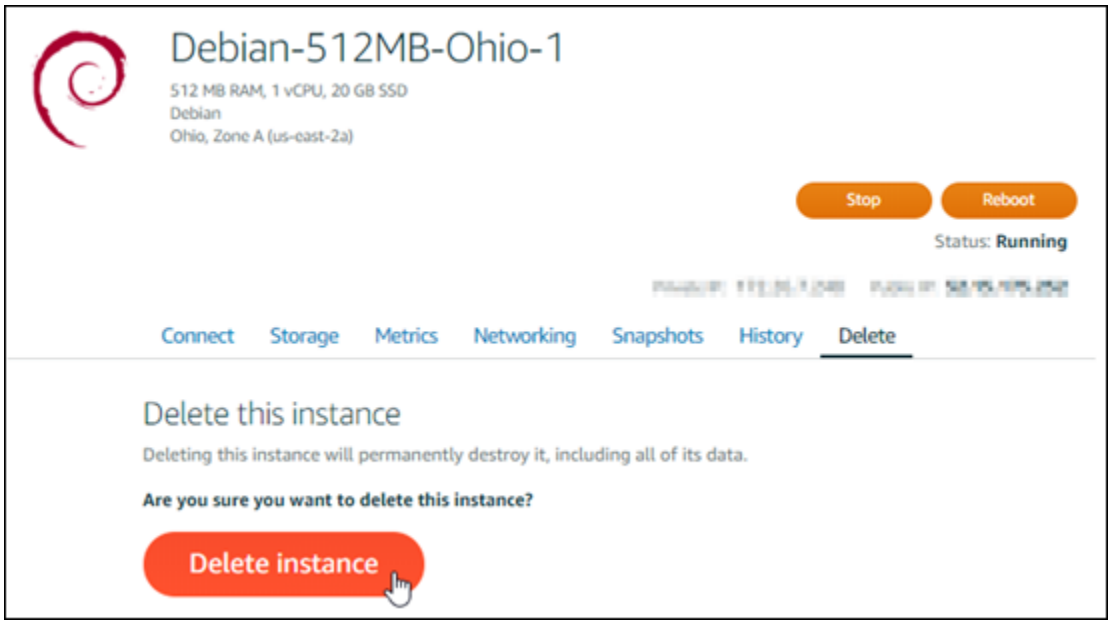
1. Lightsail コンソールにサインインします。
2. 削除するインスタンスのアクションメニューアイコン (⋮) を選択し、[削除] を選択します。



3. [はい] を選択して削除を確定します。

Lightsail コンソールのインスタンス管理ページでインスタンスを削除する

1. Lightsail コンソールのホームページで、削除するインスタンスを選択します。
2. [削除]、[インスタンスの削除] の順に選択します。



3. [はい] を選択して削除を確定します。

AWS CLI を使用してインスタンスを削除する

1. 以下の前提条件を完了します (まだの場合)。
   a. AWS CLI のインストール。詳細については、「AWS CLI をインストールする」を参照してください。
   b. AWS CLI を設定します。詳細については、「AWS CLI の設定」を参照してください。
2. ターミナルまたはコマンドプロンプトウィンドウを開き、次のコマンドを入力して、削除するインスタンスの名前を取得します。

```
aws lightsail get-instances
```

   次のような結果が表示されます。

3. 削除するインスタンスの名前を選択してコピーします。この名前は次のステップで使用します。

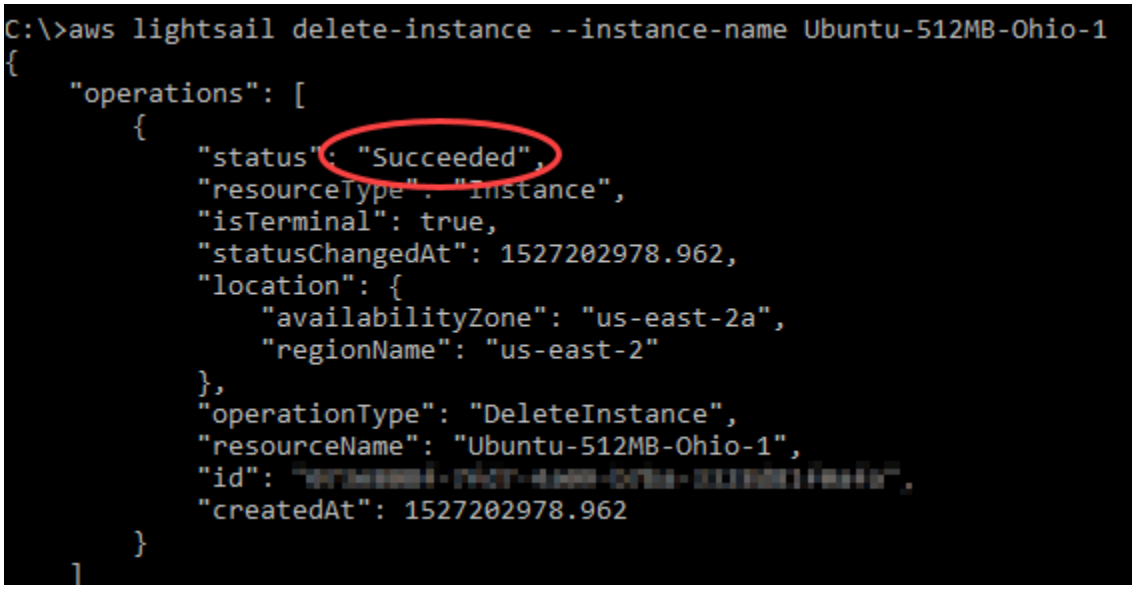
> **Note**
>
> 削除するインスタンスが表示されない場合は、インスタンスがある AWS リージョン に AWS CLI が設定されていることを確認します。詳細については、「AWS CLI の設定」を参照してください。

4. 以下のコマンドを入力して、インスタンスを削除します。

```
aws lightsail delete-instance --instance-name InstanceName
```

コマンドで、*InstanceName* をインスタンスの名前に置き換えます。

削除が成功した場合は、次のような確認メッセージが表示されます。



> **Note**
>
> 削除が失敗した場合は、エラーメッセージが表示されます。インスタンス名を正確にコピーして貼り付けたことを確認し、もう一度試します。

次のステップ

インスタンスの削除後に、インスタンスに関連付けられていた静的 IP、スナップショット、ブロックストレージディスク、およびロードバランサーは Lightsail に残るため、引き続き料金が発生します。これらのリソースの削除方法については、以下の記事を参照してください。

- 静的 IP の削除
- スナップショットを削除する
- ブロックストレージディスクをデタッチおよび削除する
- ロードバランサーの削除

Amazon Lightsail インスタンスイメージを選択してください

Lightsail には、仮想プライベートサーバーを作成するためのオプションがいくつか用意されています。このトピックは、自分のプロジェクトに適したオペレーティングシステム (OS)、アプリケーションスタック、または開発スタックを決定するのに役立ちます。アプリケーションは、機能領域 (CMS や e コマースなど) 別に整理されています。

プラットフォームを比較する

Lightsail には、Linux/UNIX ベースのプラットフォームと Windows ベースのプラットフォームの 2 つのプラットフォームから選択できます。既にアプリケーションがわかっている場合、既に OS プラットフォームを選んでいることと思います。開始するには、以下のいずれかのオプションを選択します。

- Linux/Unix ベースのインスタンスの使用を開始する
- Windows ベースのインスタンスの使用を開始する

オペレーティングシステムの比較

Lightsail には複数のオペレーティングシステムから選択できます。

[Windows Server 2022]

Windows Server を実行している Lightsail は、Microsoft ウェブプラットフォームを使用してアプリケーションをデプロイするための高速で信頼性の高い環境です。Lightsail を使用すると、互換性のある Windows ベースのソリューションを、高性能で信頼性が高く、費用対効果の高いコンピューティングプラットフォームで実行できます。 AWS クラウド 一般的な Windows ユースケースには、Enterprise Windows ベースのアプリケーションホスティング、ウェブサイトおよびウェブサービスホスティング、データ処理、分散テスト、ASP.NET アプリケーションホスティング、そして Windows ソフトウェアを必要とする、他のすべてのアプリケーションが含まれます。サポート終了情報については、「Microsoft の ウェブサイト」を参照してください。

このブループリントは Lightsail IPv6 専用インスタンスプランと互換性があります。

Windows Server 2022 イメージの詳細について説明します

Windows Server 2019

Windows Server を実行している Lightsail は、Microsoft ウェブプラットフォームを使用してアプリケーションをデプロイするための高速で信頼性の高い環境です。Lightsail を使用すると、高性能で信頼性が高く、費用対効果の高い AWS クラウドコンピューティングプラットフォームで、互換性のある Windows ベースのソリューションを実行できます。一般的な Windows ユース

ケースには、Enterprise Windows ベースのアプリケーションホスティング、ウェブサイトおよびウェブサービスホスティング、データ処理、分散テスト、ASP.NET アプリケーションホスティング、そして Windows ソフトウェアを必要とする、他のすべてのアプリケーションが含まれます。サポート終了情報については、「Microsoft の ウェブサイト」を参照してください。

このブループリントは Lightsail IPv6 専用インスタンスプランと互換性があります。

Windows Server 2019 イメージの詳細について説明します

Windows Server 2016

Windows Server を実行している Lightsail は、Microsoft ウェブプラットフォームを使用してアプリケーションをデプロイするための高速で信頼性の高い環境です。Lightsail を使用すると、高性能で信頼性が高く、費用対効果の高い AWS クラウドコンピューティングプラットフォームで、互換性のある Windows ベースのソリューションを実行できます。一般的な Windows ユースケースには、Enterprise Windows ベースのアプリケーションホスティング、ウェブサイトおよびウェブサービスホスティング、データ処理、分散テスト、ASP.NET アプリケーションホスティング、そして Windows ソフトウェアを必要とする、他のすべてのアプリケーションが含まれます。サポート終了情報については、「Microsoft の ウェブサイト」を参照してください。

このブループリントは Lightsail IPv6 専用インスタンスプランと互換性があります。

Windows Server 2016 イメージの詳細

Amazon Linux 2023

Amazon Linux 2023 (AL2023) は、 AWSでの汎用ワークロード向けとして理想的な次世代型 Amazon Linux です。AL2023 には、一般公開の開始後 5 年間サポートが提供されます。AL2023 は Amazon Linux パッケージリポジトリの特定のバージョンにロックされるため、更新を適用するタイミングと方法を管理できます。また、AL2023 では、頻繁に行われる更新や、コンプライアンスのニーズを満たすのに役立つ機能を取得することもできます。

AL2023 から起動された Lightsail インスタンスには、デフォルトでインスタンスメタデータサービスバージョン 2 (IMDSv2) が適用されます。詳細については、「インスタンスメタデータサービスバージョン 2 の仕組み」を参照してください。

このブループリントは Lightsail IPv6 専用インスタンスプランと互換性があります。

Amazon Linux 2023 の詳細はこちら。

Amazon Linux 2

Amazon Linux 2 は、 AWSの Linux サーバーオペレーティングシステムでは、前世代の Amazon Linux です。これはクラウドおよびエンタープライズアプリケーションの開発と実行のために設

計された、安定した安全で高性能な実行環境を提供します。Amazon Linux 2 では、Linux での最新のイノベーションへのアクセスを含む長期的なサポートを提供する、アプリケーション環境を取得できます。Amazon Linux 2 には追加料金はかかりません。サポート終了情報については、「Amazon Linux 2 に関するよくある質問」を参照してください。

このブループリントは Lightsail IPv6 専用インスタンスプランと互換性があります。

Amazon Linux 2 の詳細をご覧ください。

AlmaLinux OS 9

AlmaLinux OS 9 はオープンソースで、コミュニティが所有、管理する、永遠にフリーなエンタープライズ Linux ディストリビューションです。長期的な安定性を重視し、堅牢なプロダクショングレードのプラットフォームを提供します。 AlmaLinux RHEL® およびプレストリーム CentOS と互換性があります。サポート終了情報については、AlmaLinux OS Foundation の Web サイトを参照してください。

このブループリントは Lightsail IPv6 専用インスタンスプランと互換性があります。

OS 9 について詳しくはこちらをご覧ください。 AlmaLinux

CentOS 7

> **Important**
>
> CentOS 7 は、2024 年 6 月 30 日にサポート終了 (EOL) を迎えます。2024 年 6 月 30 日以降は、この設計図を使用して新しい Lightsail インスタンスを作成できなくなります。詳細については、「CentOS のウェブサイト」を参照してください。

CentOS は、コミュニティに支援されたエンタープライズクラスのコンピューティングプラットフォームを無料で提供する Linux ディストリビューションであり、そのアップストリームソースである Red Hat Enterprise Linux と機能的に互換性があります。サポート終了情報については、Red Hat のウェブサイトを参照してください。

CentOS 7 の詳細について説明します。

CentOS Stream 9

CentOS Stream 9 は、CentOS Stream ディストリビューションでの次のメジャーリリースです。CentOS Stream 9 は、Red Hat Enterprise Linux (RHEL) の開発の直前に一歩先んじて、継続的に配信されるディストリビューションであり、Fedora Linux と RHEL の中間的な位置付け

となっています。これは、RHEL と機能的な互換性を持つように設計されており、提供される Linux 環境は安定で、予測性、管理性、再現性を備えています。サポート終了情報については、「CentOS ウェブサイト」を参照してください。

このブループリントは Lightsail IPv6 専用インスタンスプランと互換性があります。

CentOS Stream の詳細はこちら。

Debian 10、11、12

> **Important**
>
> Debian 10 は 2024 年 6 月 30 日に長期サポートが終了します。2024 年 6 月 30 日以降は、この設計図を使用して新しい Lightsail インスタンスを作成できなくなります。

Debian は無料のオペレーティングシステムであり、インターネット上で協力しあう、世界中の何千人ものボランティアによって開発されました。Debian プロジェクトの主な強みは、ボランティアの基盤、Debian 社会契約およびフリーソフトウェアへの献身、可能な限り最高のオペレーティングシステムを提供するというコミットメントです。この新しいリリースは、その方向へ向かうためにもう 1 つの重要なステップです。サポート終了情報については、「Debian ウェブサイト」を参照してください。

このブループリントは Lightsail IPv6 専用インスタンスプランと互換性があります。

Debian の詳細を確認してください。

FreeBSD 13

FreeBSD は、パワーサーバー、デスクトップ、および組み込みシステムに使用されるオペレーティングシステムです。FreeBSD は、カリフォルニア大学バークレー校で開発された UNIX のバージョンである BSD から派生し、大規模なコミュニティによって 30 年間以上継続的に開発されています。アクセス量が最大級のウェブサイトや、非常に広範な組み込みネットワーキングシステムおよびストレージシステムの多くで、この OS のネットワーキング、セキュリティ、ストレージ、およびモニタリング機能 (pf ファイアウォール、Capsicum や CloudABI 機能のフレームワーク、ZFS ファイルシステム、DTrace 動的トレースフレームワークなど) を理由に、FreeBSD がプラットフォームとして選ばれています。サポート終了情報については、「FreeBSD の ウェブサイト」を参照してください。

このブループリントは Lightsail IPv6 専用インスタンスプランと互換性があります。

FreeBSD の詳細を確認してください。

[openSUSE 15]

openSUSE ディストリビューションは、安定していて使いやすい、包括的な汎用 Linux ディストリビューションです。openSUSE は、デスクトップやサーバーで作業するユーザーおよび開発者を対象としています。openSUSE は、初心者、経験豊富なユーザー、およびマニアックなユーザーなどに最適であり、つまり誰にとっても申し分ありません。サポート終了の情報については、「openSUSE の ウェブサイト」を参照してください。

このブループリントは Lightsail IPv6 専用インスタンスプランと互換性があります。

openSUSE の詳細を確認してください。

[Ubuntu 18、20 および 22]

> **Important**
>
> Ubuntu 18.04は、2023年5月31日にスタンダードSupport が終了しました。2024 年 5 月 31 日以降は、この設計図を使用して新しい Lightsail インスタンスを作成できなくなります。詳細については、Ubuntu のウェブサイトを参照してください。

Ubuntu Server は Debian ベースの Linux オペレーティングシステムであり、仮想サーバーに使用されます。Ubuntu のデフォルトインストールには、Firefox、Thunderbird、Transmission LibreOffice など、さまざまなソフトウェアが含まれています。APT ベースのパッケージ管理ツール (`apt-get`) を使用して、Evolution、GIMP、Pidgin、Synaptic など、多数の追加ソフトウェアパッケージをインストールできます。サポート終了情報については、「Ubuntu ウェブサイト」を参照してください。

このブループリントは Lightsail IPv6 専用インスタンスプランと互換性があります。

Ubuntu の詳細を確認してください。

データベースアプリケーションの比較

Lightsail では以下のデータベースアプリケーションを使用できます。

[SQL Server 2022 Express]

SQL Server Express は、無料でダウンロード、配布、使用できるリレーショナルデータベース管理システムです。小規模な埋め込みアプリケーションを特にターゲットとするデータベースを

構成しています。この Lightsail イメージは Windows サーバー 2022 のベース OS 上で動作します。

このブループリントは Lightsail IPv6 専用インスタンスプランと互換性があります。

SQL Server 2022 Express イメージの詳細

[SQL Server 2019 Express]

SQL Server Express は、無料でダウンロード、配布、使用できるリレーショナルデータベース管理システムです。小規模な埋め込みアプリケーションを特にターゲットとするデータベースを構成しています。この Lightsail イメージは Windows サーバー 2022 のベース OS 上で動作します。

このブループリントは Lightsail IPv6 専用インスタンスプランと互換性があります。

SQL Server 2019 Express イメージの詳細

SQL Server 2016 Express

SQL Server Express は、無料でダウンロード、配布、使用できるリレーショナルデータベース管理システムです。小規模な埋め込みアプリケーションを特にターゲットとするデータベースを構成しています。この Lightsail イメージは Windows サーバー 2016 のベース OS 上で動作します。

このブループリントは Lightsail IPv6 専用インスタンスプランと互換性があります。

SQL Server 2016 Express イメージの詳細

CMS アプリケーションの比較

Lightsail では以下のコンテンツ管理システム (CMS) アプリケーションを使用できます。

WordPress Bitnami による認定を受けています。

Bitnami WordPress は Lightsail ready-to-use WordPress 上で実行するための事前設定済みのイメージです。 WordPress は、ブログや Web サイトを構築するための人気のウェブパブリッシングプラットフォームです。提供されているさまざまなテーマ、拡張機能、プラグイン、およびウィジェットを使用して WordPress をカスタマイズできます。

WordPress フルテーマシステムを備えているため、数回クリックするだけでサイトの見た目を変えることができます。また、 WordPress 既存の無料または商用テーマを使用することもできます。 WordPress W3C の標準に完全に準拠しています。

WordPress Bitnami アプリケーションの詳細をご覧ください。

WordPress Bitnami によるマルチサイト認定を受けています。

WordPress マルチサイトを使用すると、管理者は同じインスタンスから複数のウェブサイトをホストおよび管理できます。 WordPress これらのウェブサイトは、すべてが一意のドメイン名を持ち、所有者がカスタマイズできます。また、サーバー管理者が利用可能にするテーマやプラグインなどのアセットを共有できます。すべてのサイトに対する更新を同時にプッシュできるため、常に安全で保護された状態に保つことができます。

WordPress マルチサイトは、中央管理者に全体的な制御を委ねながら、多数のユーザーが自分の Web サイトをホストできるようにする必要がある大学、企業、機関などの組織に最適です。

Bitnami WordPress マルチサイトアプリケーションの詳細をご覧ください。

cPanel & WebHost マネージャー (WHM)

cPanel & WHM は、Linux OS 用に構築されたツールのスイートであり、シンプルなグラフィカルユーザーインターフェイスを介してウェブホスティングタスクを自動化する機能を提供します。お客様のサーバー管理、および顧客によるウェブサイト管理を簡素化することを目的としています。

cPanel & WHM についての詳細はこちら。

PrestaShop Bitnami によるパッケージ化

PrestaShop 世界で最も多用されている e コマースソリューションの 1 つです。これは、100 万人以上のアクティブなメンバーのコミュニティを持つ、フリーでオープンソースソフトウェアです。オンラインストアをすぐに立ち上げて運営できるように設計されており、すぐに販売を開始できるようにあらかじめ設定されたテーマと、サイトの外観を簡単にカスタマイズできるライブコンフィグレーターを備えています。 PrestaShop マルチストアサポート、カスタマイズ可能なURL、複数の支払いゲートウェイオプション（Stripeを含むPayPal ）、Amazon、eBay、Facebookなどとのマーケットプレイス統合を備えています。

について詳しくは、こちらをご覧ください。 PrestaShop

Ghost (Bitnami によってパッケージ化)

Ghostは、個人的なブログから主要なニュースサイトまで、あらゆるものに適したパブリッシングプラットフォームです。Node.js 上に構築された最新のテクノロジースタックは、コンテンツ作成者の使いやすさを維持しながら、他のアプリケーションやツールとの統合を求める開発者に汎用性と柔軟性を提供します。

Bitnami Ghost アプリケーションの詳細を確認する。

Joomla! (Bitnami によってパッケージ化)

ビタミ・ジュモラ！ Joomla! を実行するための事前設定済みのイメージです。 ready-to-use Lightsail で。Joomla! は、さまざまなウェブサイトおよびポータルの構築に使用できる CMS です。個人、企業、中小企業、非営利団体、およびその他の組織のウェブサイトで利用できます。

Joomla! では、登録システム機能によってユーザーが個人用オプションを設定することもできます。認証はユーザー管理の重要な部分であり、Joomla! は LDAP や OpenID などの複数のプロトコルをサポートしています。Joomla! は多言語をサポートし、ウェブサイトや管理パネルで多言語を利用するためのガイダンスを提供しています。また、Banner Manager により、サイトのバナーを簡単にセットアップして管理できます。インプレッション数の設定や特別な URL などのメトリクスを追跡できます。

Bitnami Joomla! アプリケーションの詳細を確認してください。

Drupal (Bitnami によってパッケージ化)

Bitnami Drupal は Lightsail で Drupal ready-to-use を実行するための事前設定済みのイメージです。Drupal は、ユーザーがコンテンツを簡単に公開、管理、および整理できるようにする、コンテンツ管理プラットフォームです。Drupal は、コミュニティのウェブポータル、ディスカッションサイト、企業のウェブサイトなどに使用されています。Drupal は、モジュールを接続することによって容易に拡張できます。Drupal は、高パフォーマンス用に構築されていて、多数のサーバーにスケーリング可能であり、REST、JSON、SOAP、およびその他の形式と簡単に統合できます。

Drupal では何千ものアドオンモジュールとデザインを無償で使用できます。Drupal は複数の言語でも使用できます。

Bitnami Drupal アプリケーションの詳細を確認してください。

アプリケーションスタックおよびサーバーの比較

Lightsail には、さまざまな開発プロジェクトに対応する 5 つのアプリケーションスタックとサーバーがあります。各イメージでは、Linux/Unix (Ubuntu) が基本オペレーティングシステムとして使用されます。

[LAMP スタック (PHP 8) (Bitnami によってパッケージ化)]

Bitnami LAMP スタックでは、PHP アプリケーションの開発とデプロイが簡略化されます。これには、Apache、MySQL、PHP、 ready-to-run およびのバージョンと phpMyAdmin、これらの各コンポーネントの実行に必要なその他のソフトウェアが含まれています。Bitnami LAMP スタッ

クは完全に統合および設定されているため、Lightsail でインスタンスを作成するとすぐにアプリケーションの開発を開始できます。Bitnami LAMP スタックは定期的に更新されているため、バンドルされている各コンポーネントの最新の安定版リリースを常に利用できます。

このブループリントは Lightsail IPv6 専用インスタンスプランと互換性があります。

Bitnami LAMP スタックの詳細を確認してください。

Django (Bitnami によってパッケージ化)

Django は、迅速な開発とクリーンで実用的な設計を奨励する高レベルの Python ウェブフレームワークです。Python は、ソフトウェア開発の多くの種類のために使用することができる動的なオブジェクト指向プログラミング言語です。Bitnami Django スタックは Django とそのランタイム依存関係のデプロイを大幅に簡素化します。また、Python、Django、MySQL、 ready-to-run Apache のバージョンも含まれています。

Bitnami Django スタックの詳細を確認してください。

Node.js (Bitnami によってパッケージ化)

Bitnami Node.js は Lightsail で Node.js ready-to-use を実行するための事前設定済みのイメージです。Node.js は Chrome JavaScript のランタイム上に構築されたプラットフォームで、高速でスケーラブルなネットワークアプリケーションを簡単に作成できます。イベント駆動型のノンブロッキング I/O モデルが使用されているため、軽量かつ効率的です。Node.js はデータ集約型のリアルタイムアプリケーションに適しています。

Bitnami Node.js スタックの詳細を確認してください。

MEAN stack (Bitnami によってパッケージ化)

Bitnami MEAN スタックでは、ワンクリックでデプロイできる、MongoDB および Node.js 用の完全な開発環境が提供されています。MongoDB、Express、Angular、Node.js、Git、PHP、およびの最新の安定版リリースが含まれています。 RockMongo

このブループリントは Lightsail IPv6 専用インスタンスプランと互換性があります。

Bitnami MEAN スタックの詳細を確認してください。

GitLab Bitnami によって CE パッケージ化されています。

Bitnami GitLab コミュニティエディション (CE) は、Lightsail ready-to-use GitLab 上で実行するための事前設定済みのイメージです。 GitLab は、高速で安全で、Ruby on Railsをベースにしたセルフホスト型のGit管理ソフトウェアです。 GitLab CI (これも含まれています) は、Git と緊密に統合されたオープンソースの継続的インテグレーション (CI) GitLabサーバーです。

GitLab 独自のサーバー上でコードを安全に保ち、リポジトリ、ユーザー、アクセス権限を管理できます。これは自己完結型であるため、インストールした GitLab を別のサーバーに簡単にコピーまたは移動できます。

GitLabBitnami スタックの詳細をご覧ください。

Nginx (LEMP スタック) (Bitnami によってパッケージ化)

Bitnami NGINX スタックでは、ワンクリックで起動できる、PHP、MySQL、および NGINX の完全な開発環境が提供されます。また phpMyAdmin、SQLite、FastCGI、Memcache ImageMagick、GD、CURL、PEAR、PECL、およびその他のコンポーネントもバンドルしています。

NGINX は非同期サーバーであり、スケーラビリティが主な長所となっています。NGINX スタックは LEMP (Linux、NGINX、MySQL、および PHP) とも呼ばれます。

Bitnami Nginx (LEMP) スタックの詳細を確認してください。

Ubuntu の Plesk ホスティングスタック

Plesk が提供するホスティングスタックを使用して、Lightsail と AWS でウェブサイトとアプリケーションを構築、保護、実行します。これには、ウェブベースのサーバー管理ツールやセキュリティツールがすべて含まれているほか、 WordPress グラフィカルユーザーインターフェイスによる自動化も含まれています。ウェブの専門家の作業を簡易化し、顧客が必要とするスケーラビリティ、セキュリティ、パフォーマンスを提供します。

Plesk をセットアップおよび設定する。

Plesk スタックの詳細を確認してください。

e コマース アプリケーション

Lightsail には現在、Magento という 1 つの電子商取引アプリケーションイメージがあります。この Magento イメージでは、Linux/Unix (Ubuntu) が基本オペレーティングシステムとして使用されます。

Magento (Bitnami によってパッケージ化)

ビットナミ Magento は、Lightsail で Magento ready-to-use を実行するための事前設定済みのイメージです。Magento を使用して、魅力的で応答性が高くて安全なサイトを構築できます。Magento は、機能が豊富で柔軟性に優れた e コマースソリューションであり、トランザク

ションオプション、マルチストア機能、ロイヤルティプログラム、製品のカテゴリ化、買い物客のフィルタリング、プロモーションルールなどの機能を備えます。

Magento を使用することによって、自社のブランドを反映しつつ高度にカスタマイズした、e コマース用サイトを作成できます。Magento はユーザーのビジネスオペレーションと統合されるため、ご自分のビジネスニーズに合わせて e コマースサイトを管理できます。

Bitnami Magento スタックの詳細を確認してください。

プロジェクト管理アプリケーション

Lightsail には現在、Redmine というプロジェクト管理アプリケーションイメージが 1 つあります。このイメージでは、Linux/Unix (Ubuntu) が基本オペレーティングシステムとして使用されます。

Redmine (Bitnami によってパッケージ化)

Bitnami Redmine は Lightsail で Redmine ready-to-use を実行するための事前設定済みのイメージです。Redmine は、柔軟性に優れたプロジェクト管理ウェブアプリケーションです。複数のプロジェクト、ロールベースのアクセスコントロール、ガントチャートとカレンダー、ニュース/ドキュメント/ファイルの管理、プロジェクトごとの Wiki とフォーラム、SCM 統合などがサポートされています。

このブループリントは Lightsail IPv6 専用インスタンスプランと互換性があります。

Bitnami Redmine スタックの詳細を確認してください。

Lightsail IPv6-onlyインスタンスプラン

パブリックで到達可能な IPv4 アドレスは、広く使用されているため、供給が短く、グローバル需要が絶えず増加しています。新しい IP バージョン 4 (IPv4) アドレスの最後に利用可能なブロックは、2011 年に割り当てられました。その時点から、すべてのユーザーが利用可能なアドレスの有限セットを再利用しています。IP バージョン 6 (IPv6) は次世代の IP アドレス標準です。IPv6 の追加 - 最終的には IPv4 を置き換え、IP アドレスの枯渇を解決しようとします。

IPv6-onlyインスタンスプランとは

Lightsail インスタンスプランは、選択したオペレーティングシステム (OS) とアプリケーションをバンドルします。また、IPv4 と IPv6 (デュアルスタック) の両方、または IPv6-onlyネットワーキン

グのサポートも含まれています。デュアルスタックプランは、インスタンスにパブリック IPv4 アドレスとパブリック IPv6 アドレスを割り当てます。このプランでは、必要に応じて IPv6 を有効または無効にできます。IPv6-onlyインスタンスプランでは、インスタンスはパブリック IPv6 アドレスを受け取り、パブリック IPv4 トラフィックをサポートしません。IPv6-onlyプランをサポートする Lightsail プラットフォームとブループリントについては、「」を参照してくださいAmazon Lightsail インスタンスイメージを選択してください。

パブリック IPv6-onlyインスタンスを作成します。 IPv4 IPv6-onlyインスタンスを作成する前に、IPv6 経由で通信できることを確認してください。詳細については、「 の IPv6 到達可能性」を参照してくださいLightsail で IPv6 到達可能性を検証する。既存のインスタンスをデュアルスタックから IPv6-only、または IPv6-onlyからデュアルスタックに移行するには、「」を参照してくださいスナップショットから Lightsail インスタンスを作成する。

IPv6 に関する考慮事項

IPv6-onlyインスタンスを作成する前に、次の考慮事項を確認してください。

- ネットワークインフラストラクチャとインターネットサービスプロバイダー (ISP) の両方が IPv6-compatibleであることを確認します。詳細については、「Lightsail で IPv6 到達可能性を検証する」を参照してください。
- アプリケーションとユーザーが IPv6 経由で通信できることを確認します。詳細については、「Lightsail で IPv6 到達可能性を検証する」を参照してください。
- インスタンスは IPv6 経由でのみパブリックに通信します。Lightsail アカウントの他のリソースと通信するためのプライベート IPv4 アドレスも受け取ります。IPv6-onlyインスタンスは、受信または送信パブリック IPv4 トラフィックをサポートしていません。詳細については、「Amazon Lightsail の IP アドレス」を参照してください。
- Lightsail ブラウザベースの SSH および RDP クライアントは、IPv4 トラフィックのみを受け入れます。サードパーティーのクライアントを使用して、IPv6 経由でインスタンスに SSH または RDP 接続します。詳細については、「インスタンスに接続します」を参照してください。
- 現時点では、IPv6-onlyインスタンスを Lightsail コンテンツ配信ネットワーク (CDN) ディストリビューションのオリジンとして設定することはできません。

IPv6-onlyインスタンスに移行する

既存のデュアルスタックインスタンスを IPv6-onlyプランに移行できます。開始する前に、前のIPv6 に関する考慮事項セクションを確認することをお勧めします。

移行するには、デュアルスタックインスタンスのスナップショットを作成し、スナップショットから新しいインスタンスを作成します。インスタンス作成ワークフロー中に IPv6-onlyネットワークプランを選択します。この手順の詳細については、「」を参照してくださいスナップショットから Lightsail インスタンスを作成する。

IPv6-onlyインスタンスプランからデュアルスタックプランに移行するには、代わりにデュアルスタックプランを選択します。

Lightsail の SSH キーペア

キーペアは、Amazon Lightsail インスタンスに接続するときにアイデンティティを証明するために使用する一連のセキュリティ認証情報です。キーペアはパブリックキーとプライベートキーで構成されます。Lightsail はパブリックキーをインスタンスに保存し、プライベートキーを保存します。

キーペアファイルには、以下のテキストが含まれています。

Linux および Unix インスタンスでは、プライベートキーを使用することでインスタンスへのセキュアな SSH 接続を確立できます。Windows インスタンスでは、インスタンスへのセキュアな RDP 接続を確立するために使用されるデフォルトの管理者パスワードを、プライベートキーが復号化します。

プライベートキーにアクセスできるユーザーならば誰でもインスタンスに接続できるため、プライベートキーは安全な場所に保存することが重要です。

目次

- インスタンスへの接続
- インスタンスに保存されているキーの管理

キーペアオプションの選択

Lightsail インスタンスを作成するときに、次のいずれかのキーペアオプションを選択できます。Windows インスタンスは常にデフォルトキーを使用します。このため、Windows インスタンスの作成時にキーペアを作成したり、キーをアップロードしたりすることはできません。

- デフォルトキーペア – Lightsail は、インスタンスを作成する各 AWS リージョン にデフォルトキーペアを自動的に作成します。インスタンスでデフォルトのキーペアを使用すると、Lightsail はパブリックキーをインスタンスに保存します。デフォルトキーペアのプライベートキーは、Lightsail コンソールのアカウントページからいつでもダウンロードできます。AWS リージョンごとに最大 1 つのデフォルトキーペアを設定できます。
- キーペアの作成 (Linux および Unix インスタンス) — Lightsail コンソールを使用して、インスタンスで使用する新しいカスタムキーペアを作成できます。カスタムキーペアを作成するときは、一意の名前を付けると、Lightsail はパブリックキーをインスタンスに保存します。カスタムキーペアのプライベートキーをダウンロードできるのは、最初の作成時のみです。
- キーをアップロードする (Linux および Unix インスタンス) – 既存のキーペアを独自に使用するには、パブリックキーを Lightsail にアップロードできます。インスタンスで使用するパブリックキーをアップロードするときは、一意の名前を付けると、Lightsail はそれをインスタンスに保存します。キーペアのプライベートキーは、ユーザーが保持して保存します。

複数のインスタンスに単一の公開キーを設定する場合、これらのインスタンスへの接続には同じキーペアのプライベートキーを使用できます。キーペアの管理の詳細については、「Amazon Lightsail でのキーペアの管理」を参照してください。

インスタンスに接続します

Lightsail インスタンスに接続するには、次のいずれかのオプションを使用します。

Lightsail ブラウザベースの SSH および RDP クライアント

Lightsail コンソールでは、ブラウザベースの SSH クライアントを使用して Linux および Unix インスタンスに瞬時に接続し、ブラウザベースの RDP クライアントを使用して Windows インスタンスに接続できます。Lightsail ブラウザベースの SSH および RDP クライアントは、IPv4 トラフィックのみを受け入れます。デュアルスタックインスタンスを作成するか、サードパーティーのクライア

ントを使用して IPv6 経由でインスタンスに SSH または RDP 接続します。ブラウザベースのクライアントを使用してインスタンスに接続するときは、コンピュータに SSH クライアントをインストール、キーペアを設定、または管理者パスワードを指定する必要はありません。これは、インスタンスに接続するための最も迅速な方法です。詳細については、「Connecting to your Linux or Unix instance in Amazon Lightsail」(Amazon Lightsail の Linux または Unix インスタンスに接続する) および「Connecting to your Windows instance in Amazon Lightsail」(Amazon Lightsail の Windows インスタンスに接続する) を参照してください。

ブラウザベースのクライアントは、インスタンスの作成時に設定するキーペア (デフォルトキーや、ユーザーが作成またはアップロードするキーなど) とは異なるキーペアを使用します。このため、当初設定したキーのいずれかを削除したり紛失したりした場合でも、ブラウザベースのクライアントを使用してインスタンスへの接続を継続することができます。

サードパーティーの SSH および RDP クライアント

サードパーティーの SSH クライアントを使用して Linux および Unix インスタンスに接続し、サードパーティーの RDP クライアントを使用して Windows インスタンスに接続することができます。SSH クライアントを使用するときは、インスタンスで設定したキーペアのプライベートキーを使用するようにクライアントを設定する必要があります。RDP クライアントを使用するときは、Windows インスタンスの管理者パスワードを指定する必要があります。

Windows コンピュータをローカルで使用する場合は、次のクライアントを使用して Lightsail インスタンスに接続できます。

- PuTTY – Use PuTTY SSH を使用した Linux または Unix インスタンスへの接続には、PuTTY を使用します。詳細については、「PuTTY を設定してインスタンスに接続する」を参照してください。
- リモートデスクトップ接続 – RDP を使用した Windows インスタンスへの接続には、リモートデスクトップ接続クライアントを使用します。詳細については、「Windows コンピュータでリモートデスクトップ接続クライアントを使用して Windows インスタンスに接続する」を参照してください。

Mac コンピュータをローカルで使用する場合は、次のクライアントを使用して Lightsail インスタンスに接続します。

- ターミナル内のネイティブ SSH クライアント – Linux および Unix インスタンスへの接続には、ターミナル内のネイティブ SSH クライアントを使用します。詳細については、「ターミナルで SSH を使用して Linux または Unix インスタンスに接続する」を参照してください。

- Microsoft リモートデスクトップ – RDP を使用した Windows インスタンスへの接続には、macOS 用の Microsoft リモートデスクトップクライアントを使用します。詳細については、「Mac で Microsoft リモートデスクトップクライアントを使用して Windows インスタンスへ接続する」を参照してください。

インスタンスに保存されているキーの管理

インスタンスが実行状態になったら、インスタンスに新しいキーを追加したり、最初に割り当てたキーを交換したりすることができます。例えば、組織内のユーザーが個別のキーを使用してインスタンスにアクセスする必要がある場合は、そのキーをインスタンスに追加できます。もう 1 つの例として、誰かが組織から脱退し、その人物がプライベートキー (.PEM) ファイルのコピーを持っているという場合があります。そのキーを新しいキーと交換する、または完全に削除することによって、この人物がインスタンスに接続できないようにすることが可能です。詳細については、「Amazon Lightsail のインスタンスに保存されているキーの管理」を参照してください。

トピック

- Lightsail Linux または Unix インスタンスに接続する
- Lightsail Windows インスタンスに接続する

Lightsail Linux または Unix インスタンスに接続する

Amazon Lightsail にはブラウザベースの SSH クライアントが用意されています。これは Linux または Unix インスタンスに接続する最も速い方法です。独自の SSH クライアントを使用してインスタンスに接続することもできます。詳細については、「PuTTY をダウンロードしてセットアップする」を参照してください。

SSH を使用してインスタンスに接続し、ソフトウェアパッケージのインストールやウェブアプリケーションの設定などの管理タスクをサーバーで実行します。ブラウザベースの SSH クライアントは、ソフトウェアのインストールを必要とせず、インスタンスの作成後、ほぼ即座に使用できます。

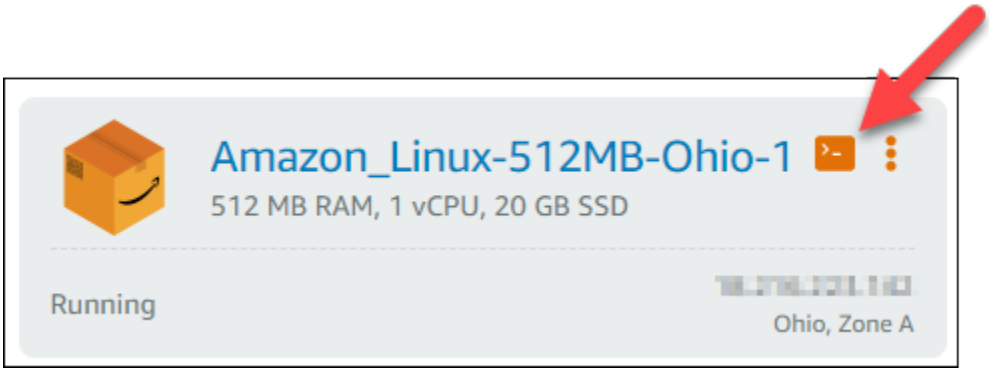
> **Note**
>
> Lightsail ブラウザベースの SSH/RDP クライアントは IPv4 トラフィックのみを受け入れます。サードパーティーのクライアントを使用して、IPv6 経由でインスタンスに SSH または RDP 接続します。詳細については、「インスタンスに接続します」を参照してください。

Lightsail で Windows Server インスタンスに接続するには、「Windows ベースのインスタンスに接続する」を参照してください。

Linux または Unix インスタンスに接続するには

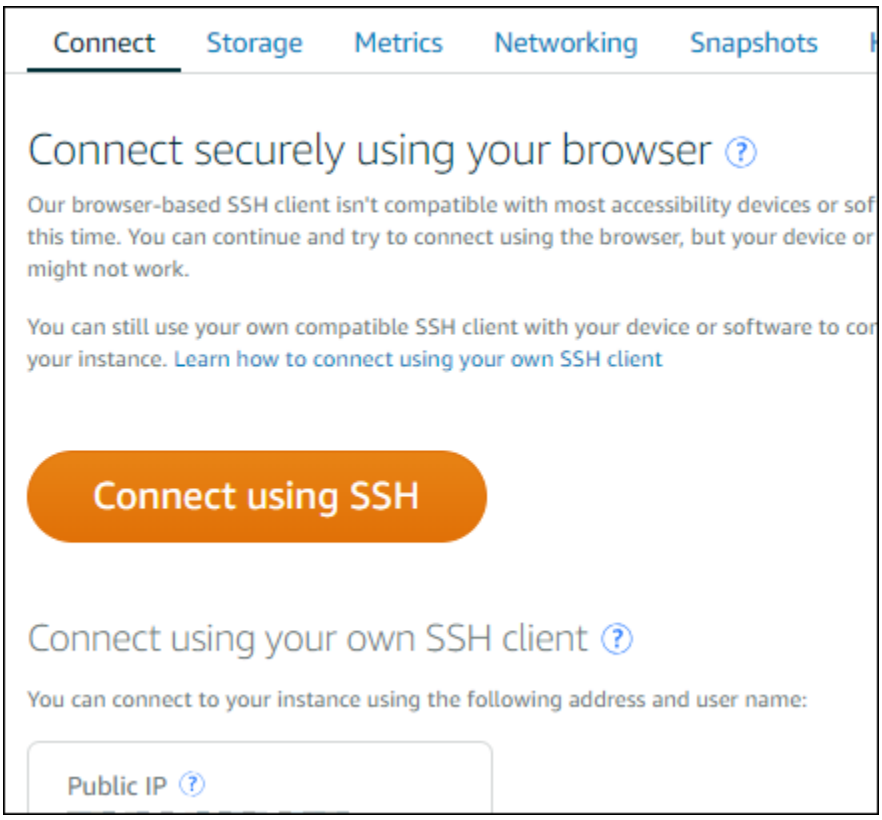
1. Lightsail コンソールにサインインします。
2. 接続先のインスタンスのブラウザベースの SSH クライアントにアクセスするには、以下のいずれかの操作を行います。

   - 次の例に示すように、クリック接続アイコンを選択します。

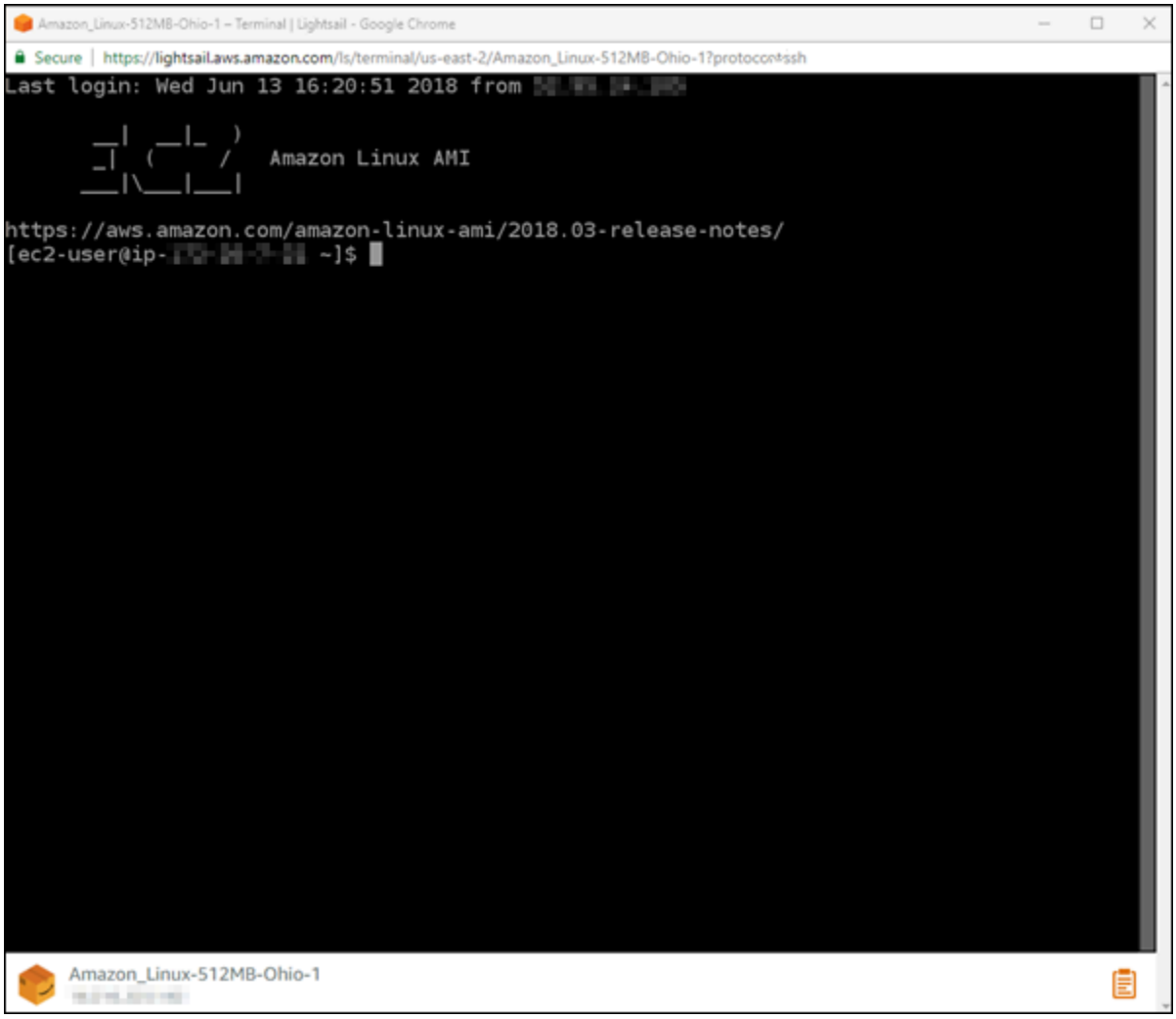


   - アクションメニューアイコン (⋮) を選択し、次に [接続] を選択します。

   - インスタンスの名前を選択し、[接続] タブの [SSH を使用して接続] を選択します。



ブラウザベースの SSH クライアントを開いて、次の例に示すようなターミナル画面が表示されると、インスタンスとのやり取りを開始できます。



> **Note**
>
> [接続] タブには、独自の SSH クライアントを使用して接続するために必要な情報も表示されます。詳細については、「PuTTY をダウンロードしてセットアップする」を参照してください。

ブラウザベースの SSH クライアントを使用して Linux または Unix インスタンスとやり取りする

ブラウザベースの SSH クライアントのターミナル画面に直接 Linux または UNIX コマンドを入力したり、ターミナル画面からテキストをコピーしたりします。以下のセクションでは、SSH でクリップボードに (またはクリップボードから) テキストをコピーして貼り付ける方法を示します。

ブラウザベースの SSH クライアントにテキストを貼り付けるには

1. ローカルデスクトップのテキストを強調表示し、Ctrl+C または Cmd+C を押してテキストをローカルクリップボードにコピーします。
2. ブラウザベースの SSH クライアントの右下で、クリップボードアイコンを選択します。ブラウザベースの SSH クライアントのクリップボードテキストボックスが表示されます。
3. テキストボックス内をクリックし、Ctrl+V または Cmd+V を押して、ローカルクリップボードの内容をブラウザベースの SSH クライアントのクリップボードに貼り付けます。
4. SSH ターミナル画面の任意の領域を右クリックし、ブラウザベースの SSH クライアントのクリップボードからターミナル画面にテキストを貼り付けます。

ブラウザベースの SSH クライアントからテキストをコピーするには

1. ターミナル画面でテキストを強調表示します。
2. ブラウザベースの SSH クライアントの右下で、クリップボードアイコンを選択します。ブラウザベースの SSH クライアントのクリップボードテキストボックスが表示されます。
3. コピーするテキストを強調表示し、Ctrl+C または Cmd+C を押してテキストをローカルクリップボードにコピーします。これで、コピーしたテキストをローカルデスクトップの任意の場所に貼り付けることができます。

Lightsail の SSH キーの設定

Secure SHell (SSH) は、仮想プライベートサーバー (または Lightsail インスタンス) に安全に接続するためのプロトコルです。SSH は、リモートサーバーを承認されたユーザーに一致させるパブリックキーとプライベートキーを作成することによって動作します。このキーペアを使用し、ブラウザベースの SSH ターミナルを使用して Lightsail インスタンスに接続できます。

SSH の詳細については、「SSH について」を参照してください。

Lightsail インスタンスの作成時のデフォルトオプションでは、Lightsail がユーザーの SSH キーを管理するようになっています。Lightsail では、Linux ベースのインスタンスに安全に接続するための、ブラウザベースの SSH クライアントが提供されています。このクライアントは完全に機能するターミナルであり、そこでコマンドを入力したりインスタンスへの変更を行ったりできます。

Windows ベースのインスタンスでは、SSH の代わりにリモートデスクトップ (RDP) プロトコルを使用します。Lightsail における Windows ベースのインスタンスの詳細については、「Lightsail で Windows ベースのインスタンスの使用を開始する」を参照してください。

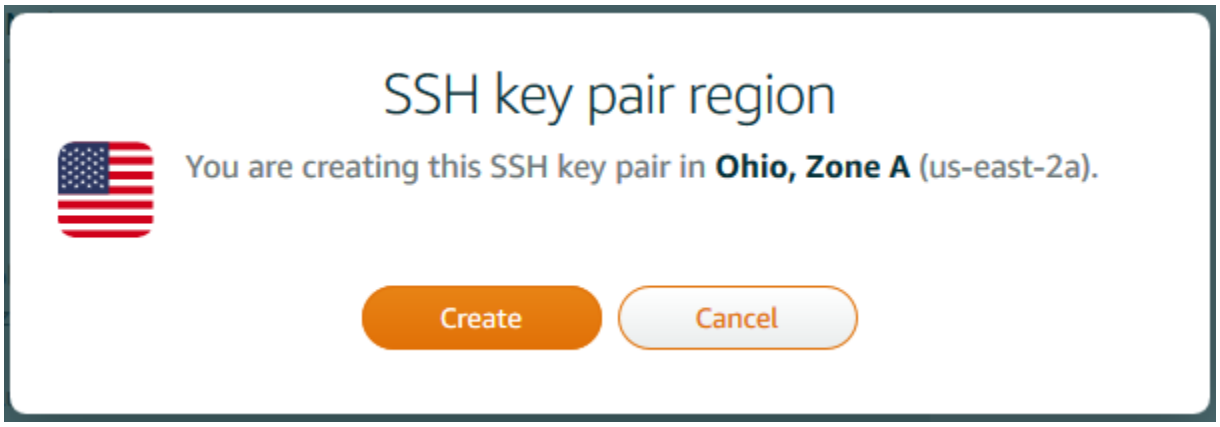
> **Important**
>
> SSH キーはリージョンごとに管理されます。新しい AWS リージョン でインスタンスを作成する際に、そのリージョンのデフォルトのキーペアを使用するオプションが提示されます。そのリージョンでカスタムキーを使用することもできます。独自のキーをアップロードする場合は、Lightsail インスタンスがあるリージョンごとにアップロードする必要があることに注意してください。

デフォルトのキーを使用している場合でも、保管用にプライベートキーをダウンロードできます。キーのダウンロードは、インスタンスの作成時または作成後に行うことができます。インスタンスを作成した後にキーをダウンロードすることを選択した場合、[アカウント] ページの [SSH キー] の下でダウンロードできます。

新規キーの作成

デフォルトのキーを使用することを選択しない場合は、Lightsail インスタンスの作成時に新規キーペアを作成できます。

1. まだ作成していない場合は [インスタンスの作成] を選択します。
2. [インスタンスの作成] ページで [SSH キーペアの変更] を選択します。
3. 新規作成を選択します。
4. 新規キーを作成しているリージョンが Lightsail に表示されます。



   [Create] (作成) を選択します。
5. キーペアの名前を入力します。

   リソース名:

- Lightsail アカウントの各 AWS リージョン 内で一意である必要があります。
- 2〜255 文字を使用する必要があります。
- 先頭と末尾は英数字または数字を使用する必要があります。
- 英数字、数字、ピリオド、ダッシュ、アンダースコアを使用することができます。

6. [キーペアの生成] を選択します。

> **Important**
>
> 見つけやすい場所にキーを保存します。また、他のユーザーがそのキーを読み取りできないようにアクセス許可を設定することをお勧めします。

7. インスタンスの作成を続行します。

既存のキーのアップロード

Lightsail インスタンスの作成時に、既存のキーをアップロードすることもできます。

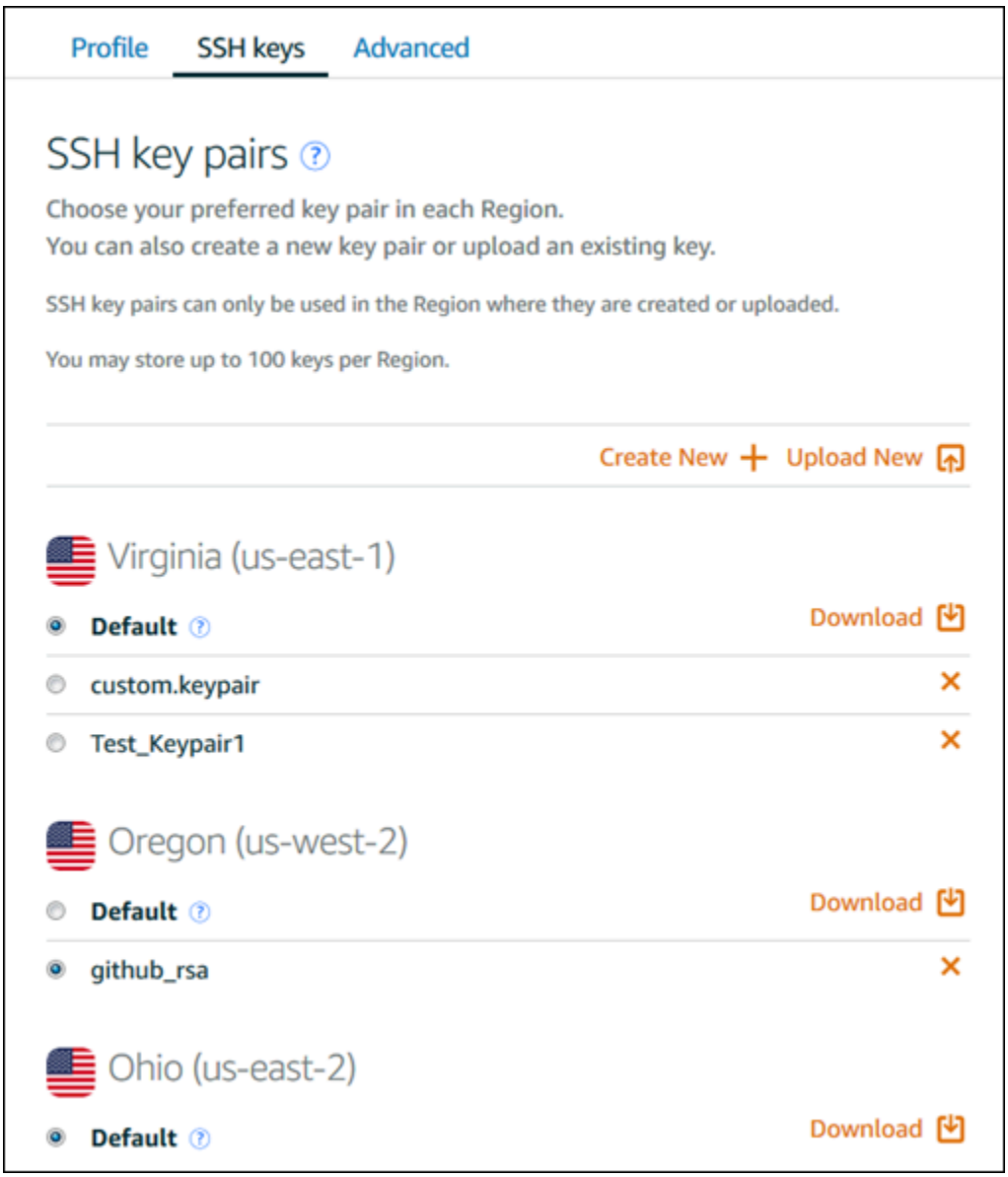
1. まだ作成していない場合は [インスタンスの作成] を選択します。
2. [インスタンスの作成] ページで [SSH キーペアの変更] を選択します。
3. [今すぐアップロード] を選択します。
4. 新規キーをアップロードしているリージョンが Lightsail に表示されます。

   [アップロード] を選択します。
5. [参照] を選択して、ローカルマシンでキーを見つけます。

   プライベートキーではなくパブリックキーをアップロードしていることを確認します。例えば、`github_rsa.pub` です。
6. [Upload key] (キーのアップロード) をクリックします。
7. インスタンスの作成を続行します。

キーを管理する

キーは、[アカウント] ページの [SSH キー] タブで管理できます。各リージョンで使用中の各キーペアが表示されます。



このページで、新しい Lightsail インスタンスを作成する際に、デフォルトで使用するキーを変更できます。新規キーの作成、既存のキーのアップロード、およびプライベートキーのダウンロードを行うこともできます。PuTTY と同様に SSH クライアントを使用して接続できますが、プライベートキーを持っている必要があります。プライベートキーは [アカウント] ページでダウンロードできます。Lightsail インスタンスに接続するように PuTTY をセットアップする方法の詳細を確認してください。

SSH コマンドを使用して Lightsail Linux/UNIX ベースのインスタンスConnect

ローカルマシンが macOS を含む Linux または Unix オペレーティングシステムを使用している場合は、ターミナルウィンドウから SSH クライアントを使用して Amazon Lightsail の Linux または Unix インスタンスに接続できます。

このガイドで説明するインスタンスへの接続方法は、多数あるうちの 1 つです。他の方法に関する詳細は、「SSH のキーペア」を参照してください。

Lightsail で Linux または Unix インスタンスに接続する最も簡単な方法は、Lightsail コンソールで利用できるブラウザベースの SSH クライアントを使用することです。詳細については、「Linux または Unix インスタンスに接続する」を参照してください。

> **Important**
>
> Lightsail ブラウザーベースの SSH/RDP クライアントは IPv4 トラフィックのみを受け入れます。サードパーティのクライアントを使用して IPv6 経由でインスタンスに SSH または RDP で接続します。詳細については、「インスタンスに接続します」を参照してください。

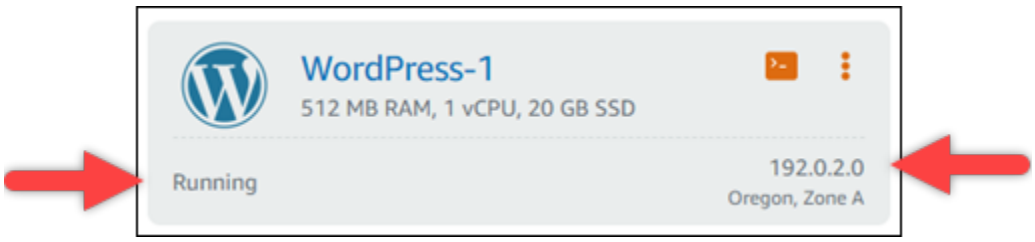
コンテンツ

ステップ 1: インスタンスが実行されていることを確認し、パブリック IP アドレスを取得する

以下の手順では、Lightsail コンソールにサインインして、インスタンスが実行状態であることを確認し、インスタンスのパブリック IP アドレスを取得します。SSH 接続を確立するには、インスタンスが実行中になっている必要があります。またこのガイドの後半で SSH に接続する際、インスタンスのパブリック IP アドレスが必要になります。

1. Lightsail コンソールにサインインします。
2. Lightsail ホームページのインスタンスタブで、接続するインスタンスを見つけます。
3. インスタンスが実行中であることを確認し、インスタンスのパブリック IP アドレスを書き留めます。

   次の例に示すように、インスタンスの状態とパブリック IP アドレスはインスタンス名の横に表示されます。



ステップ 2: インスタンスで使用されている SSH キーペアを確認する

次の手順では、インスタンスで使用されている SSH キーペアを確認します。インスタンスに対して認証し、SSH 接続を確立するには、キーペアのプライベートキーが必要になります。

1. Lightsail ホームページのインスタンスタブで、接続するインスタンスの名前を選択します。

   インスタンス管理ページが表示され、インスタンスを管理するためのさまざまなタブオプションが表示されます。

2. [接続] タブで、下にスクロールして、インスタンスで使用されているキーペアを確認します。考えられる可能性は 2 つあります。

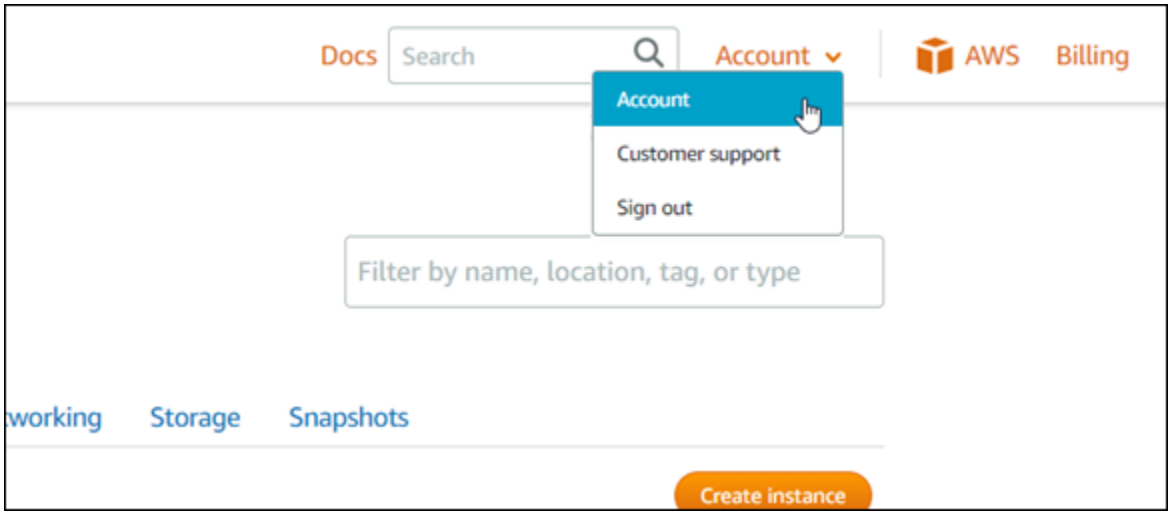
1. 次の例は、インスタンスが作成された AWS リージョン用のデフォルトのキーペア を使用しているインスタンスを示しています。インスタンスがデフォルトのキーペアを使用している場合は、この手順のステップ 3 に進み、キーペアのプライベートキーをダウンロードします。Lightsail は、各 AWS リージョンのデフォルトkey pair プライベートキーのみを保存します。

   > You configured this instance to use **default (us-west-2)** key pair.
   > You can download your default private key from the Account page.

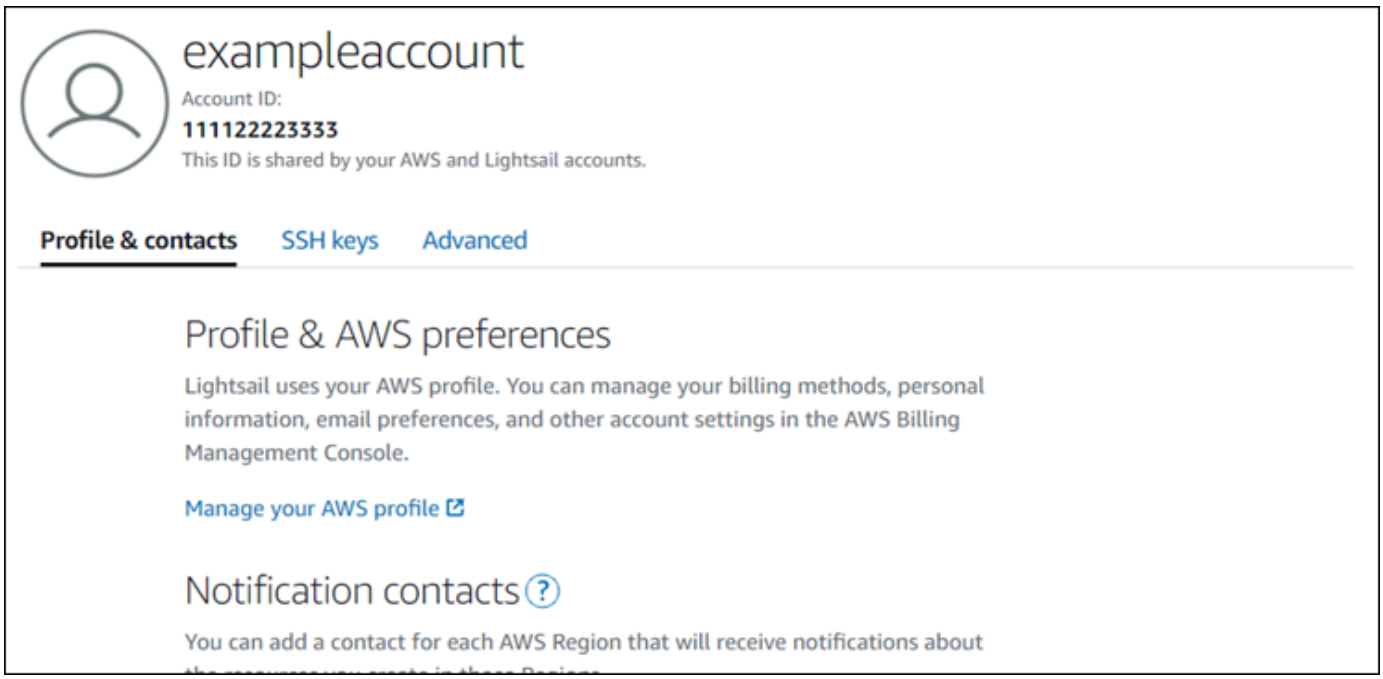
2. 次の例は、ユーザによってアップロードまたは作成されたカスタムキーペアを使用しているインスタンスを示しています。インスタンスがカスタムキーペアを使用している場合は、キーを保存している、カスタムキーペアのプライベートキーの位置を特定する必要があります。カスタムキーペアのプライベートキーを無くした場合、インスタンスへの SSH 接続を、独自のクライアントを使って確立することができなくなります。ただし、Lightsail コンソールで利用できるブラウザベースの SSH クライアントは引き続き使用できます。カスタムキーペアのプライベートキーの位置を特定したら、このガイドの次の「ステップ 3: プライベートキーの権限を変更し、SSH を使ってインスタンスに接続する」のセクションに進んでください。

   > You configured this instance to use **MyKeyPair (us-west-2)** key pair.

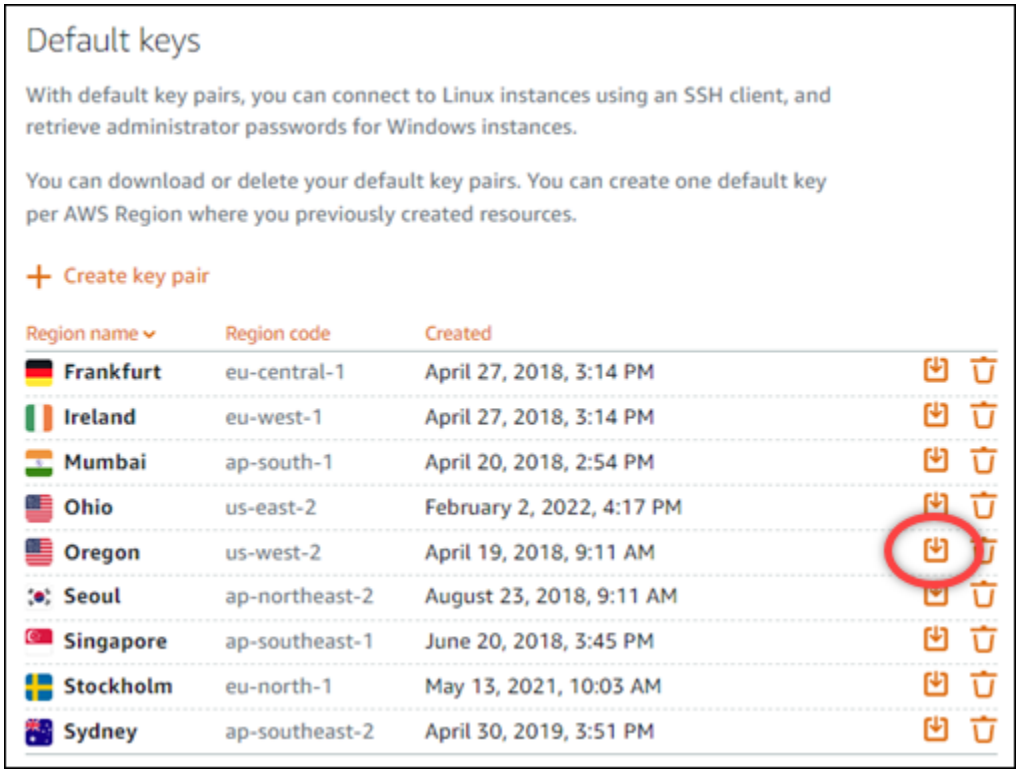
3. トップナビゲーションメニューで [コンソール] を選択し、[アカウント] を選択します。



アカウント管理ページが表示され、アカウント設定を管理するためのさまざまなタブオプションが表示されます。



4. [SSH キー] タブを選択します。
5. 下にスクロールし、接続先のインスタンスの AWS リージョンのデフォルトキーの横にあるダウンロードアイコンを選択します。



プライベートキーはユーザーのローカルマシンにダウンロードされます。ユーザーのホームディレクトリにある「Keys」フォルダなど、SSH キーが保存されているディレクトリに、ダウン

ロードしたキーを移動することも可能です。このガイドの次のセクションで、プライベートキーが保存されるディレクトリを参照する必要があります。プライベートキーが .pem 以外の形式で保存しようとした場合、保存する前に手動で形式を .pem に変更する必要があります。

> **Note**
>
> Lightsail には、.pemファイルやその他の証明書形式を操作するためのユーティリティは用意されていません。プライベートキーファイルの形式変換する必要がある場合、OpenSSL などのフリーのオープンソースツールを容易に利用できます。

このガイドの次の「ステップ 3: プライベートキーのアクセス権限を変更して、SSH を使用してインスタンスに接続する」のセクションに進んで、ダウンロードしたプライベートキーを使ってインスタンスへの SSH 接続を確立します。

ステップ 3: プライベートキーのアクセス許可を変更し、SSH を使ってインスタンスに接続する

次の手順では、プライベートキーファイルの権限を変更して、お客様以外のユーザーが読み書きできないように変更します。次に、ローカルマシンでターミナルウィンドウを開き、SSH コマンドを実行して Lightsail のインスタンスとの接続を確立します。

1. ローカルマシンでターミナルウィンドウを開きます。
2. 次のコマンドを入力して、キーペアのプライベートキーが本人にしか読み書きできないようにします。これは、一部のオペレーティングシステムで要求される、セキュリティのベストプラクティスです。

```
sudo chmod 400 /path/to/private-key.pem
```

コマンドで */path/to/private-key.pem* を、インスタンスで使われるキーペアのプライベートキーが保存されている場所を向いたディレクトリパスに、置き換えます。

例:

```
sudo chmod 400 /Users/user/Keys/LightsailDefaultKey-us-west-2.pem
```

3. SSH を使用して Lightsail のインスタンスに接続するには、次のコマンドを入力します。

```
ssh -i /path/to/private-key.pem username@public-ip-address
```

コマンドを、以下のように置き換えます。

- */path/to/private-key.pem* を、インスタンスで使われるキーペアのプライベートキーが保存されている場所を向いたディレクトリパスに置き換えます。
- *username* をインスタンスのユーザーネームに置き換えます。インスタンスで使用されるブループリントに応じて、以下のいずれかのユーザー名を指定できます。
  - AlmaLinux OS 9、Amazon Linux 2、Amazon Linux 2023、CentOS Stream 9、FreeBSD、および openSUSE インスタンス:`ec2-user`
  - CentOS 7 インスタンス:`centos`
  - Debian インスタンス: `admin`
  - Ubuntu インスタンス: `ubuntu`
  - Bitnami インスタンス: `bitnami`
  - Plesk インスタンス: `ubuntu`
  - cPanel & WHM インスタンス : `centos`
- *public-ip-address*このガイドの前半で Lightsail コンソールでメモしたインスタンスのパブリック IP アドレスに置き換えてください。

絶対パスの例:

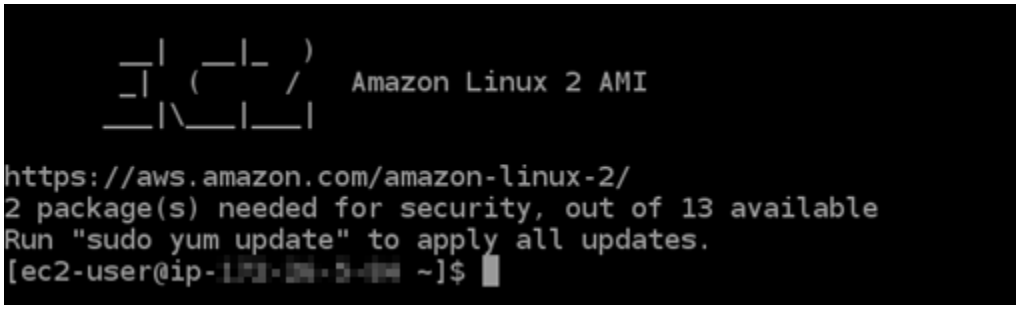
```
ssh -i /Users/user/Keys/LightsailDefaultKey-us-west-2.pem ec2-user@192.0.1.0
```

相対パスの例:

`./` が `.pem` ファイルをプレフィックスすることに気を付けてください。`./` を省略して単に `LightsailDefaultKey-us-west-2.pem` を書くだけでは動作しません。

```
ssh -i ./LightsailDefaultKey-us-west-2.pem ec2-user@192.0.1.0
```

インスタンスにようこそ、のメッセージが表示されたら、インスタンスには正常に接続された状態です。次の例は、Amazon Linux 2 インスタンスのウェルカムメッセージを示しています。他のインスタンスのブループリントでも、同様のウェルカムメッセージがあります。接続したら、Lightsail のインスタンスでコマンドを実行できます。接続を解除するには、`exit` を入力して Enter を押します。



PuTTY を使用して Lightsail Linux/Unix ベースのインスタンスに接続する

Lightsail のブラウザベースの SSH ターミナルに加えて、PuTTY などの SSH クライアントを使用して Linux ベースのインスタンスに接続することもできます。PuTTY をセットアップする方法については、Lightsail の「SSH を使用して接続するように PuTTY をダウンロードしてセットアップする」を参照してください。

> **Note**
>
> RDP を使用して Windows ベースのインスタンスに接続するには、「Windows ベースの Lightsail インスタンスに接続する」を参照してください。

Lightsail が提供するデフォルトのプライベートキー、Lightsail の新しいプライベートキー、または別のサービスで使用する別のプライベートキーを使用できます。

1. PuTTY を起動します (たとえば、[スタート] メニューで [すべてのプログラム]、[PuTTY]、[PuTTY] の順に選択します)。
2. [ロード] を選択し、保存済みセッションを見つけます。

   保存済みセッションがない場合は、「ステップ 4: プライベートキーとインスタンスの情報を使用して PuTTY の設定を完了する」を参照してください。
3. インスタンスのオペレーティングシステムに応じて、次のいずれかのデフォルトのユーザー名を使用してログインします。

   - AlmaLinux、Amazon Linux 2、Amazon Linux 2023、CentOS Stream 9、FreeBSD、openSUSE インスタンス: `ec2-user`
   - CentOS 7 インスタンス: `centos`
   - Debian インスタンス: `admin`

- Ubuntu インスタンス: `ubuntu`
- Bitnami インスタンス: `bitnami`
- Plesk インスタンス: `ubuntu`
- cPanel & WHM インスタンス : `centos`

インスタンスオペレーティングシステムの詳細については、「Lightsail でのイメージの選択」を参照してください。

SSH の詳細については、「SSH と Amazon Lightsail インスタンスへの接続」を参照してください。

SFTP を使用して Lightsail Linux インスタンスに接続する

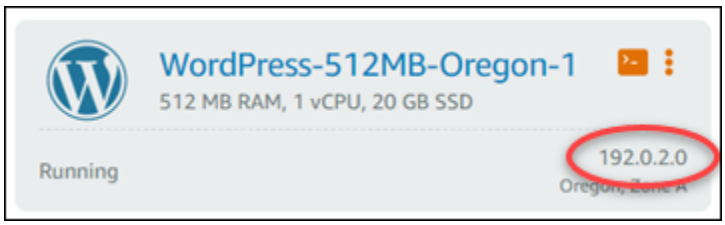
SFTP (SSH File Transfer Protocol) を使用してインスタンスに接続することで、ローカルコンピュータと Amazon Lightsail の Linux または Unix インスタンス間でファイルを転送できます。これを行うには、インスタンスのプライベートキーを取得する必要があり、これを使用して FTP クライアントを設定します。このチュートリアルでは、インスタンスに接続するように FileZilla FTP クライアントを設定する方法を示します。これらのステップは、他の FTP クライアントにも適用されます。

コンテンツ

前提条件

以下の前提条件を完了します (まだの場合)。

- ローカルコンピュータ FileZilla に をダウンロードしてインストールします。詳細については、次のダウンロードオプションを参照してください。
  - Windows 用ダウンロード FileZilla クライアント
  - Mac OS X 用 FileZilla クライアントをダウンロードする
  - Linux 用 FileZilla クライアントをダウンロードする
- インスタンスのパブリック IP アドレスを取得します。Lightsail コンソール にサインインし、次の例に示すように、インスタンスの横に表示されるパブリック IP アドレスをコピーします。

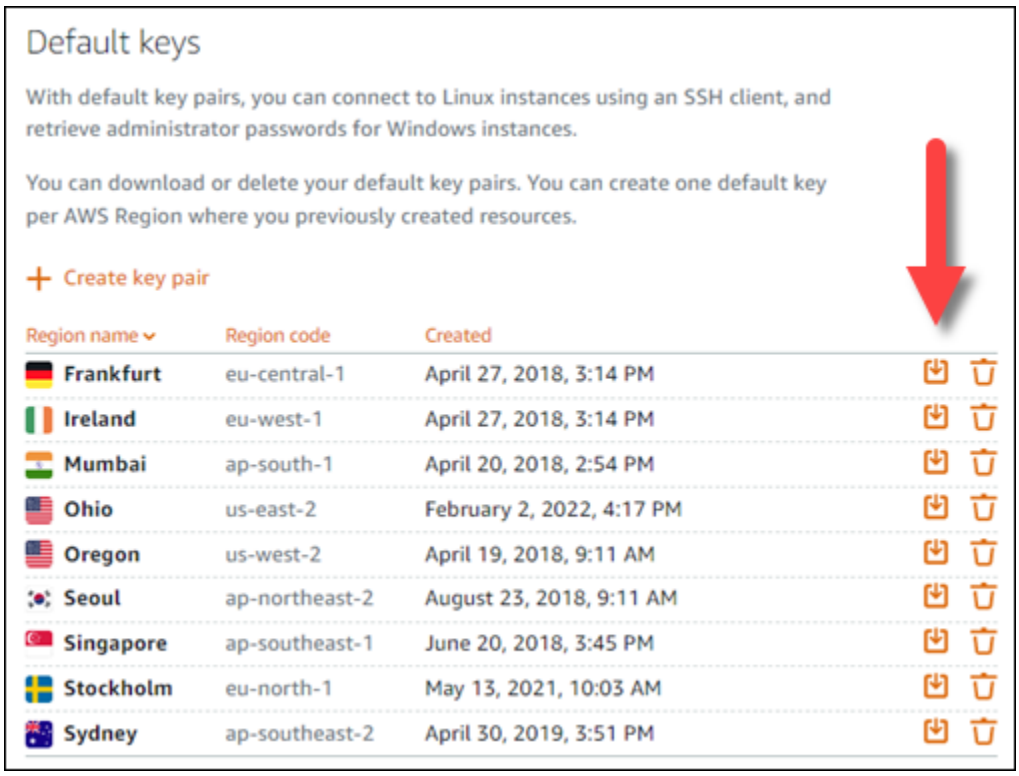


インスタンスの SSH キーを取得する

を使用してインスタンスに接続するために必要な、インスタンスの AWS リージョンのデフォルトのプライベートキーを取得するには、次のステップを実行します FileZilla。

> **Note**
>
> 独自のキーペアを使用している場合、または Lightsail コンソールを使用してキーペアを作成した場合は、独自のプライベートキーを検索し、それを使用してインスタンスに接続します。Lightsail コンソールを使用して独自のキーをアップロードしたり、キーペアを作成したりしても、Lightsail はプライベートキーを保存しません。プライベートキーなしで SFTP を使用してインスタンスに接続することはできません。

1. Lightsail コンソールにサインインします。
2. 上部のナビゲーションバーで [アカウント] を選択し、ドロップダウンから [アカウント] を選択します。
3. [SSH キー] タブを選択します。
4. ページの [Default keys] (デフォルトキー) セクションまで下にスクロールします。
5. インスタンスが配置されているリージョンのデフォルトのプライベートキーの横にある [ダウンロード] を選択します。

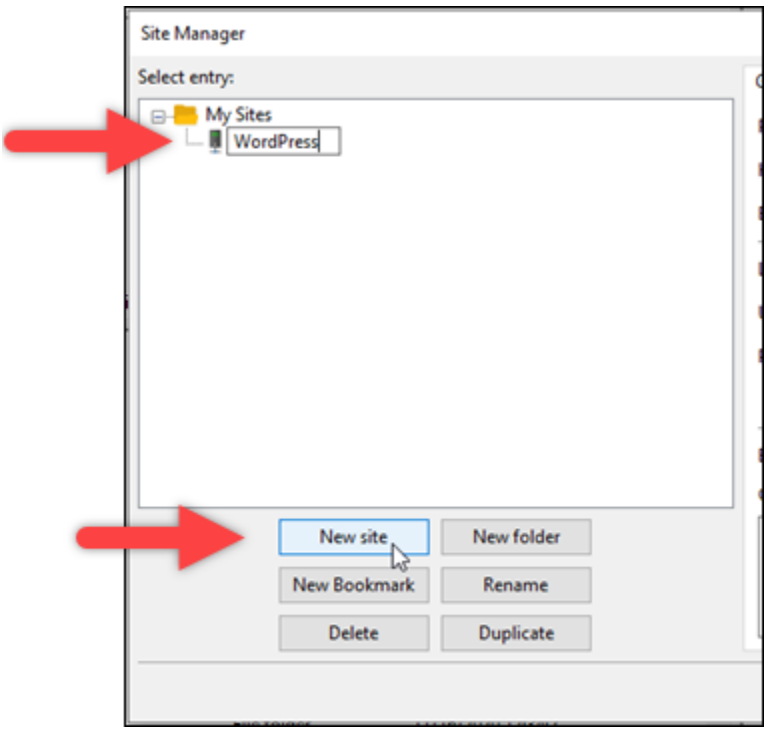


6. ローカルドライブのセキュリティが確保された場所にプライベートキーを保存します。

インスタンス FileZilla を設定して接続する

インスタンスに接続する FileZilla ように を設定するには、次のステップを実行します。

1. を開きます FileZilla。
2. [ファイルFile]、[サイトマネージャー] を選択します。
3. [新しいサイト] を選択してサイトに名前を付けます。

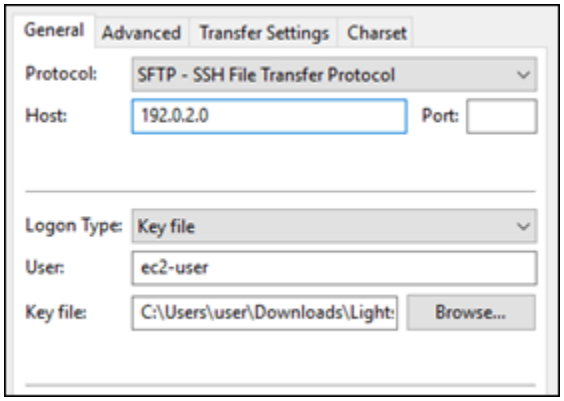


4. [プロトコル] ドロップダウンで、[SFTP – SSH File Transfer プロトコル] を選択します。
5. [ホスト] テキストボックスに、インスタンスのパブリック IP アドレスを入力するか、貼り付けます。
6. [ログオンタイプ] ドロップダウンで、[キーファイル] を選択します。
7. [ユーザー] テキストボックスに、インスタンスのオペレーティングシステムに応じて、次のいずれかのデフォルトのユーザー名を入力します。

   - AlmaLinux、Amazon Linux 2、Amazon Linux 2023、CentOS Stream 9、FreeBSD、openSUSE インスタンス: `ec2-user`
   - CentOS 7 インスタンス: `centos`
   - Debian インスタンス: `admin`
   - Ubuntu インスタンス: `ubuntu`
   - Bitnami インスタンス: `bitnami`
   - Plesk インスタンス: `ubuntu`
   - cPanel & WHM インスタンス : `centos`

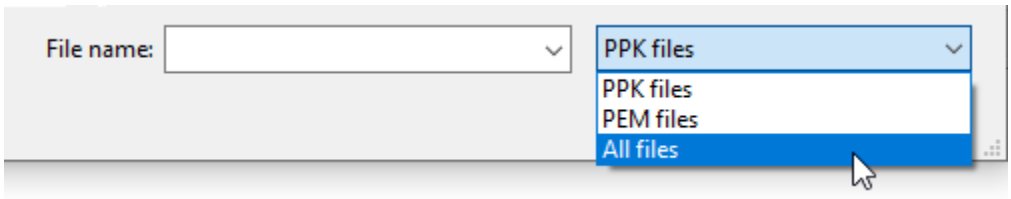
> **Important**
>
> ここにリストされているデフォルトのユーザー名とは異なるユーザー名を使用している場合は、ユーザーにインスタンスへの書き込み許可を付与します。

8. [キーファイル] テキストボックスの横で、[参照] を選択します。

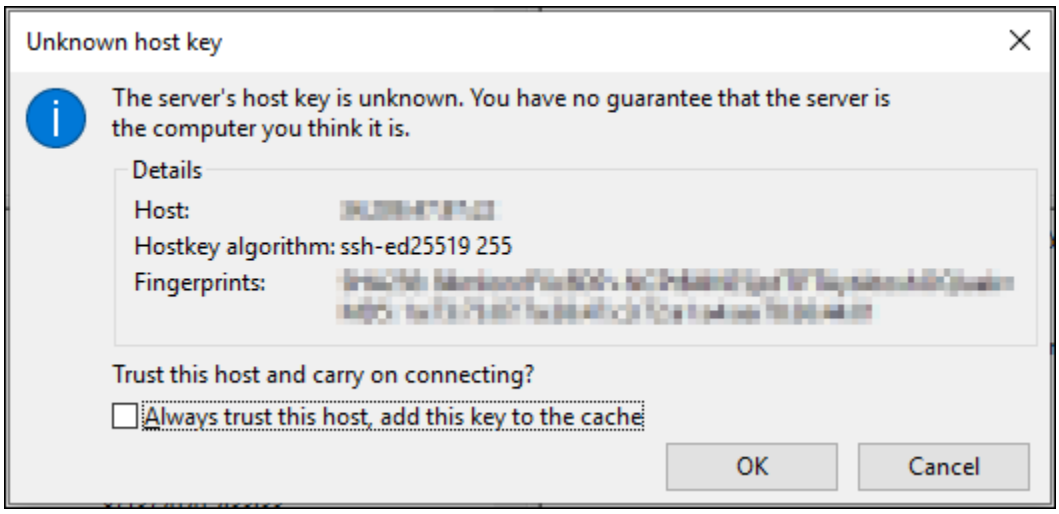


9. この手順の前半で Lightsail コンソールからダウンロードしたプライベートキーファイルを見つけ、 を開くを選択します。

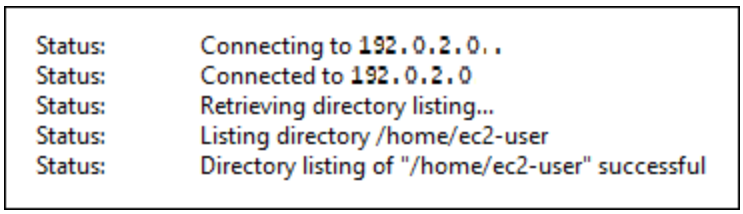
> **Note**
>
> Windows を使用している場合は、pem ファイルを検索する際に既定のファイルの種類を [すべてのファイル] に変更します。



10. [接続]を選択します。
11. 以下の例のようなプロンプトが表示され、ホストキーが不明であることが分かります。[OK] を選択してプロンプトを確認し、インスタンスに接続します。



次の例のようなステータスメッセージが表示されている場合は、正常に接続されています。



ローカルコンピュータとインスタンス間でファイルを転送する方法など FileZilla、 の使用の詳細については、FileZilla Wiki ページ「」を参照してください。

Amazon Lightsail で SSH キーを管理

キーペアを使用することで、Amazon Lightsail インスタンスへのセキュアな接続を確立することができます。Amazon Lightsail インスタンスを新規作成するときに、Lightsail が作成するキーペア (Lightsail デフォルトキーペア) の使用、またはユーザーが作成するカスタムキーペアの使用を選択できます。詳細については、「キーペアと Amazon Lightsail 内のインスタンスへの接続」を参照してください。

Linux および Unix インスタンスでは、プライベートキーを使用することでインスタンスへのセキュアな SSH 接続を確立できます。Windows インスタンスでは、インスタンスへのセキュアな RDP 接続を確立するために使用されるデフォルトの管理者パスワードを、プライベートキーが復号化します。

このガイドでは、Lightsail インスタンスで使用できるキーの管理方法について説明します。キーの表示、既存キーの削除、および新しいキーの作成やアップロードを実行できます。

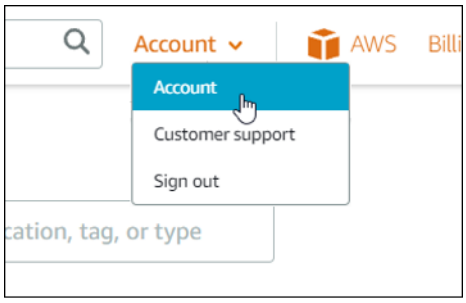
目次

- デフォルトキーとカスタムキーを表示する
- Lightsail コンソールからデフォルトキーのプライベートキーをダウンロードする
- Lightsail コンソールでカスタムキーを削除する
- Lightsail コンソールでデフォルトキーを削除して新しいキーを作成する
- Lightsail コンソールを使用してカスタムキーを作成する
- ssh-keygen を使用してカスタムキーを作成し、Lightsail にアップロードする

デフォルトキーとカスタムキーを表示する

Lightsail コンソールでデフォルトキーとカスタムキーを表示するには、以下の手順を実行します。

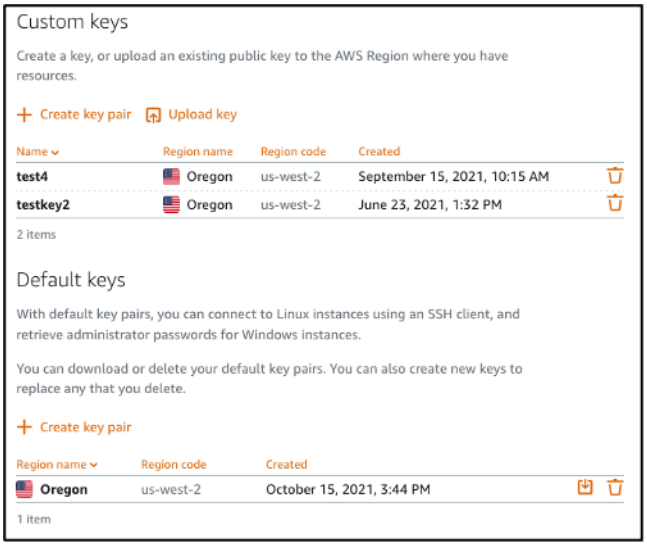
1. Lightsail コンソールにサインインします。
2. Lightsail ホームページの上部にあるナビゲーションメニューで [Account (アカウント)] を選択します。
3. ドロップダウンメニューで [Account (アカウント)] を選択します。



4. [SSH キー] タブを選択します。

   SSH キーページには、以下がリストされます。

   - カスタムキー – これらは、Lightsail コンソール、または ssh-keygen などのサードパーティーツールを使用してユーザーが作成するキーです。AWS リージョンごとに多数のカスタムキーを設定できます。
   - デフォルトキー – これらは、Lightsail が作成するキーです。デフォルトキーは、AWS リージョンごとに 1 つしか設定できません。



カスタムキーとデフォルトキーはリージョン別です。例えば、米国西部 (オレゴン) AWS リージョン内のキーを設定できるのは、そのリージョンで作成されたインスタンスだけです。キーの詳細については、「キーペアと Amazon Lightsail 内のインスタンスへの接続」を参照してください。

SSH キーページでは、キーペアの作成、キーのアップロード、キーの削除、および Lightsail デフォルトキーペアのプライベートキーのダウンロードを実行できます。

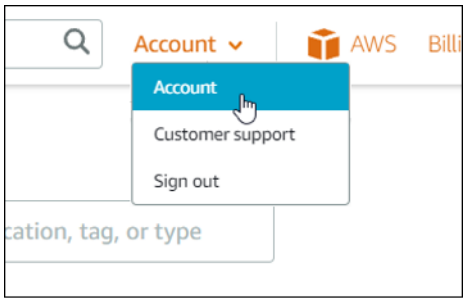
> **Note**
>
> Lightsail はカスタムキーペアのプライベートキーを保存しないため、プライベートキーをダウンロードすることはできません。カスタムキーペアのプライベートキーを紛失した場合は、新しいキーを作成して、それをインスタンスで設定する必要があります。その後、紛失したキーを削除します。詳細については、本ガイド後出の「Create a custom key using the Lightsail console」(Lightsail コンソールを使用してカスタムキーを作成する) または「Create a custom key using ssh-keygen and upload to 」(ssh-keygen を使用してカスタムキーを作成し、 にアップロードする) を参照してください。

Lightsail コンソールからデフォルトキーのプライベートキーをダウンロードする

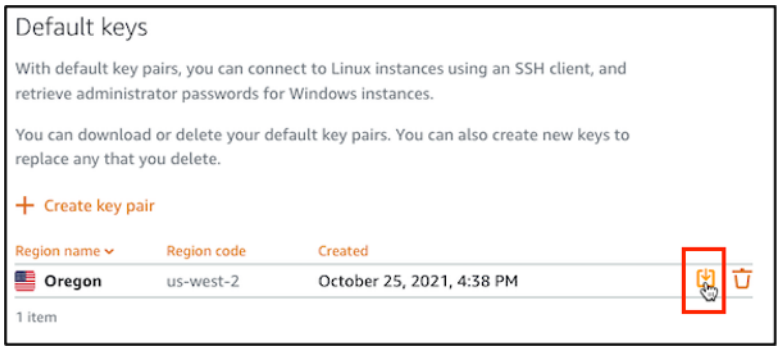
Lightsail コンソールからデフォルトキーペアのプライベートキーをダウンロードするには、以下の手順を実行します。

1. Lightsail コンソールにサインインします。

2. Lightsail ホームページの上部にあるナビゲーションペインで [Account] (アカウント) をクリックします。
3. ドロップダウンメニューで [Account (アカウント)] を選択します。



4. [SSH キー] タブを選択します。
5. そのページの [Default keys] (デフォルトキー) セクションで、ダウンロードするキーのダウンロードアイコンを選択します。



> **Important**
>
> プライベートキーは安全な場所に保存してください。このキーはインスタンスへの接続に使用できるため、公開しないでください。

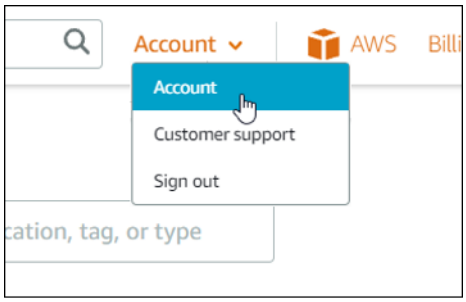
SSH クライアントは、プライベートキーを使用してインスタンスに接続するように設定できます。詳細については、「インスタンスへの接続」を参照してください。

Lightsail コンソールでカスタムキーを削除する

Lightsail コンソールでカスタムキーを削除するには、以下の手順を実行します。これによって、Lightsail で作成される新しいインスタンスにカスタムキーが設定されないようになります。

1. Lightsail コンソールにサインインします。

2. Lightsail ホームページの上部にあるナビゲーションペインで [Account] (アカウント) をクリックします。
3. ドロップダウンメニューで [Account (アカウント)] を選択します。



4. [SSH キー] タブを選択します。
5. そのページの [Custom keys] (カスタムキー) セクションで、削除するキーの削除アイコンを選択します。

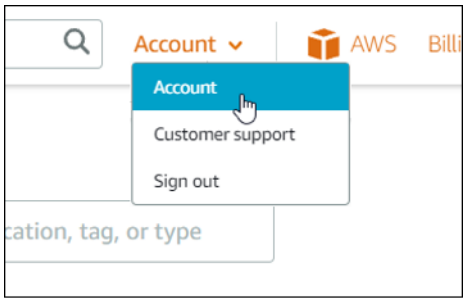
カスタムキーを削除しても、以前に作成された現在実行中のインスタンスからカスタムキーペアの公開キーが削除されることはありません。実行中のインスタンスに保存されている以前に設定された公開キーを削除するには、「Amazon Lightsail 内のインスタンスに保存されているキーの管理」を参照してください。

Lightsail コンソールでデフォルトキーを削除して新しいキーを作成する

Lightsail コンソールでデフォルトキーを削除するには、以下の手順を実行してください。削除することによって、Lightsail で作成される新しいインスタンスにデフォルトキーが設定されないようになります。削除後、削除したキーを置き換えるための新しいデフォルトキーを作成できます。Lightsail で作成する新しいインスタンスには、新しいデフォルトキーを設定することができます。

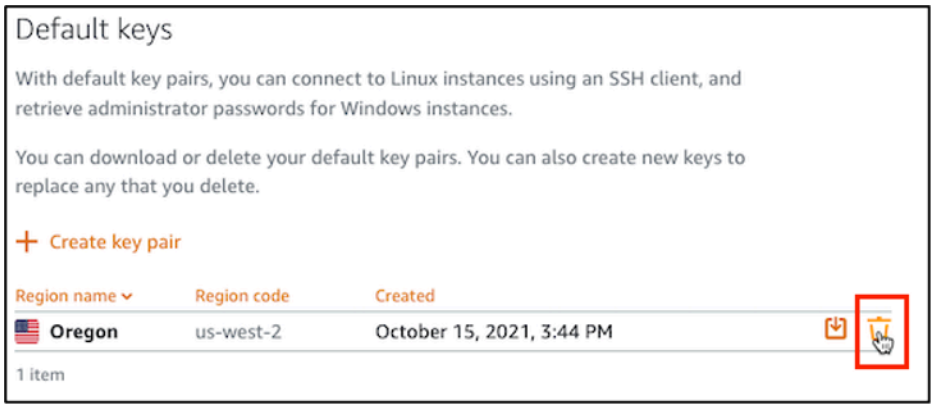
1. Lightsail コンソールにサインインします。
2. Lightsail ホームページの上部にあるナビゲーションペインで [Account] (アカウント) をクリックします。

3. ドロップダウンメニューで [Account (アカウント)] を選択します。



4. [SSH キー] タブを選択します。

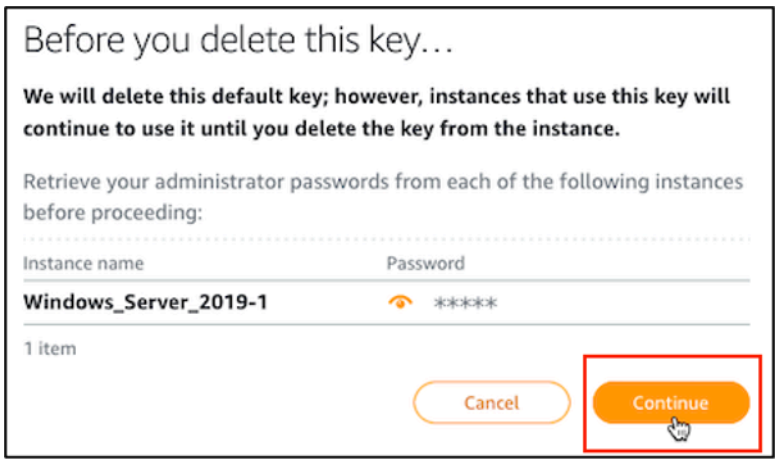
5. そのページの [Default keys] (デフォルトキー) セクションで、削除するデフォルトキーの削除アイコンを選択します。



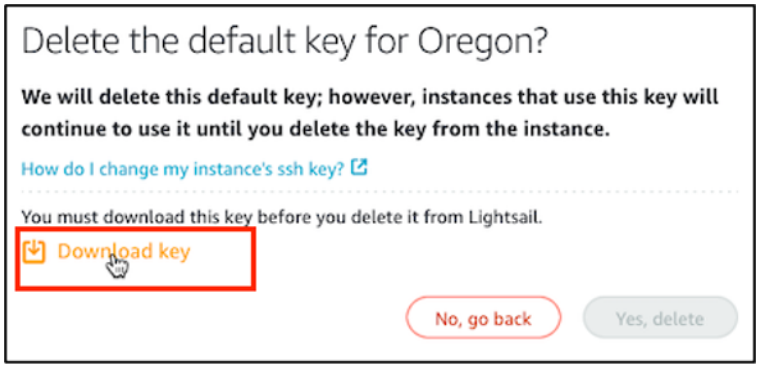
> **Important**
>
> デフォルトキーを削除しても、以前に作成された現在実行中のインスタンスからカスタムキーペアの公開キーが削除されることはありません。詳細については、「Amazon Lightsail 内のインスタンスに保存されているキーの管理」を参照してください。

6. デフォルトキーは、Windows インスタンスの管理者パスワードを生成するために使用されます。デフォルトキーを削除する前に、削除するデフォルトキーを使用するすべての Windows インスタンスから管理者パスワードを取得して保存する必要があります。

7. [Continue] (続行) を選択して、デフォルトキーを削除します。



8. デフォルトキーは、削除する前にダウンロードする必要があります。デフォルトキーをダウンロードしたら、[Yes, delete] (はい、削除します) を選択して、デフォルトキーを完全に削除することができるようになります。



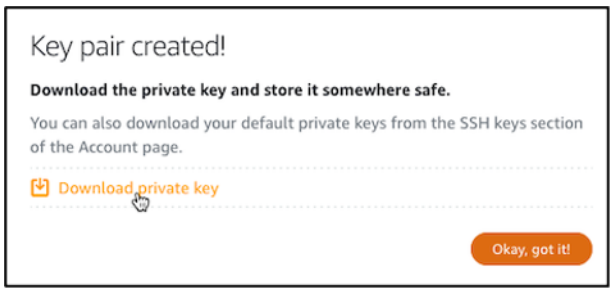
9. デフォルトキーが削除されました。[OK] を選択します。

以下の手順はオプションで、削除したデフォルトキーペアを置き換える場合にのみ実行するようにしてください。

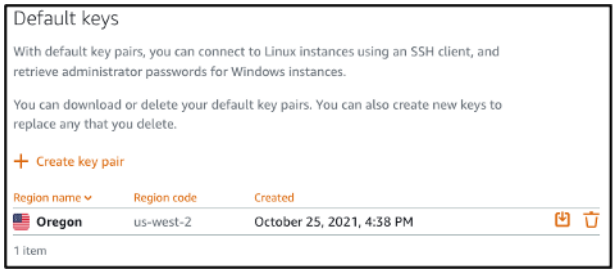
10. このページの [Default keys] (デフォルトキー) セクションで、[Create key pair] (キーペアを作成) を選択します。
11. 表示される [Select a region] (リージョンの選択) プロンプトで、新しいデフォルトキーを作成する AWS リージョンを選択します。同じ AWS リージョン内の新しいインスタンスには、新しいデフォルトキーを設定することができます。

> **Note**
>
> これらの手順を使用して、デフォルトキーペアを作成できるリージョンは、Lightsail リソースを作成した AWS リージョン のみです。新しいリージョンでデフォルトキーペアを作成するには、そのリージョンで Lightsail リソースを作成する必要があります。リソースを作成すると、デフォルトキーペアも作成されます。

12. プライベートキーをダウンロードして、安全な場所に保存します。
13. [Ok, got it!] (わかりました!) を選択して続行します。



14. Lightsail コンソールの SSH キーページで、新しいデフォルトキーを確認します。

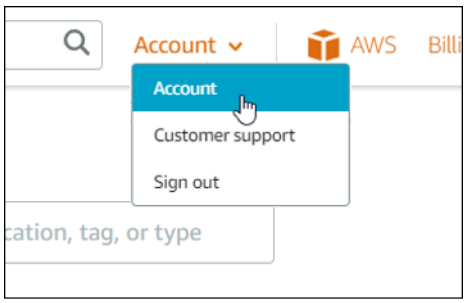


Lightsail で作成する新しいインスタンスには、新しいデフォルトキーを設定することができます。以前に作成されて、現在実行中のインスタンスで新しいデフォルトキーを設定するには、「Amazon Lightsail 内のインスタンスに保存されているキーを管理する」を参照してください。

Lightsail コンソールを使用してカスタムキーを作成する

Lightsail コンソールを使用してカスタムキーペアを作成するには、以下の手順を実行します。Lightsail で作成する新しいインスタンスには、新しいカスタムキーを設定することができます。

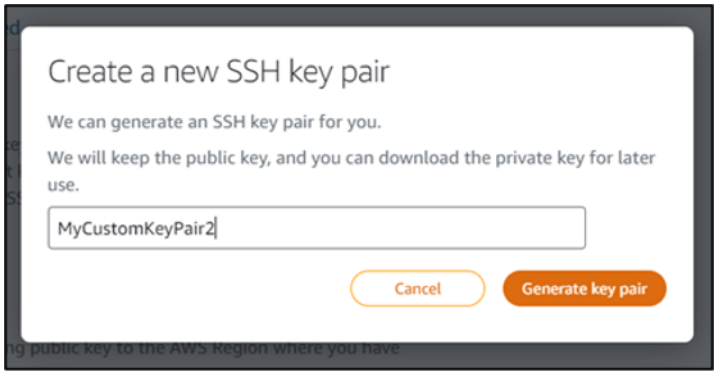
1. Lightsail コンソールにサインインします。
2. Lightsail ホームページの上部にあるナビゲーションペインで [Account] (アカウント) をクリックします。
3. ドロップダウンメニューで [Account (アカウント)] を選択します。



4. [SSH キー] タブを選択します。
5. そのページの [Custom keys] (カスタムキー) セクションで、[Create key pair] (キーペアを作成) をクリックします。

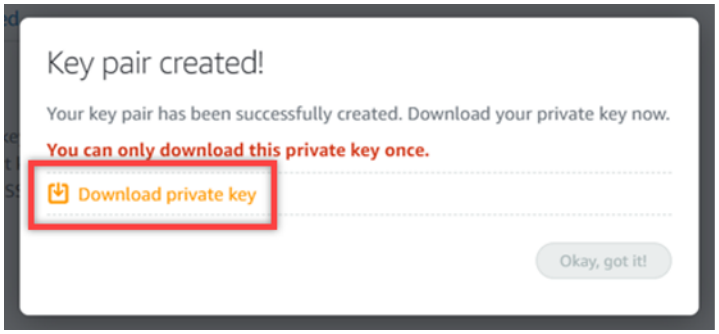
6. 表示される [Select a region] (リージョンの選択) プロンプトで、新しいカスタムキーを作成する AWS リージョンを選択します。同じ AWS リージョン内の新しいインスタンスには、新しいカスタムキーを設定することができます。

7. 表示される [Create a new SSH key pair] (新しい SSH キーペアの作成) プロンプトでカスタムキーに名前を付け、[Generate key pair] (キーペアの生成) を選択します。

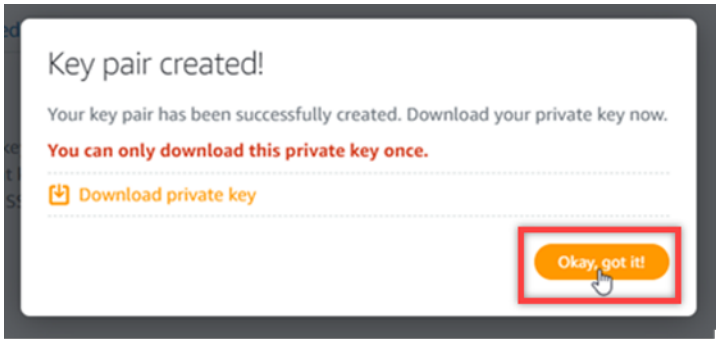


8. 表示される [Key pair created!] (キーペアが作成されました!) プロンプトで [Download private key] (プライベートキーのダウンロード) を選択して、プライベートキーをローカルコンピュータに保存します。

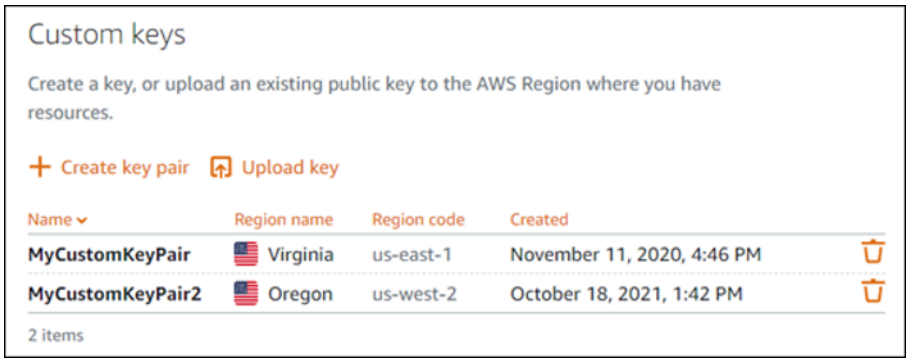
> **Important**
>
> プライベートキーは安全な場所に保存してください。このキーはインスタンスへの接続に使用できるため、公開しないでください。
> カスタムキーのプライベートキーをダウンロードできるのは、この時だけです。カスタムキーペアのプライベートキーは、Lightsail では保存されません。このプロンプトを閉じてしまうと、再度ダウンロードすることはできません。



9. [Ok, got it!] (わかりました!) を選択して、プロンプトを閉じます。



10. 新しいカスタムキーは、このページのカスタムキーセクションにリストされます。



Lightsail で作成する新しいインスタンスには、新しいカスタムキーを設定することができます。以前に作成された現在実行中のインスタンスで新しいカスタムキーを設定するには、「Amazon Lightsail 内のインスタンスに保存されているキーを管理する」を参照してください。

ssh-keygen を使用してカスタムキーを作成し、Lightsail にアップロードする

ssh-keygen などのサードパーティーツールを使用してローカルコンピュータでカスタムキーペアを作成するには、以下の手順を実行します。キーを作成したら、そのキーを Lightsail コンソールにアップロードできます。Lightsail で作成する新しいインスタンスには、新しいカスタムキーを設定することができます。

1. ローカルコンピュータで、コマンドプロンプトまたはターミナルを開きます。
2. 次のコマンドを入力して、新しいキーペアを作成します。

```
ssh-keygen -t rsa
```

3. キーペアを保存するコンピュータのディレクトリの場所を指定します。

   例えば、以下のディレクトリのいずれかを指定できます。

   a. Windows の場合: `C:\Users\<UserName>\.ssh\<KeyPairName>`

b. macOS、Linux、または Unix の場合: /home/*<UserName>*/.ssh/*<KeyPairName>*

<UserName> を現在サインインしているユーザーの名前に置き換えて、*<KeyPairName>* を新しいキーペアの名前に置き換えます。

以下の例では、Windows コンピュータの C:\Keys ディレクトリを指定し、新しいキーに MyNewLightsailCustomKey という名前を付けました。

```
C:\Users\[redacted]>ssh-keygen -t rsa
Generating public/private rsa key pair.
Enter file in which to save the key (C:\Users\[redacted]/.ssh/id_rsa): C:\Keys\MyNewLighstailCustomKey
```

4. キーのパスフレーズを入力して、Enter を押します。パスフレーズの入力中にパスフレーズは表示されません。

   このパスフレーズは後ほど、キーペアの公開キーが設定されているインスタンスに接続するために SSH クライアントでキーペアのプライベートキーを設定するときに必要になります。

```
Enter passphrase (empty for no passphrase):
```

5. 確認のためパスフレーズをもう一度入力して、Enter を押します。パスフレーズの入力中にパスフレーズは表示されません。

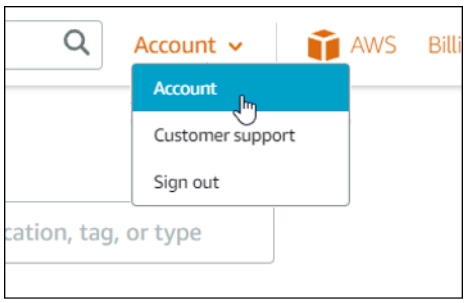
```
Enter same passphrase again:
```

6. 指定されたディレクトリにプライベートキーと公開キーが保存されたことを示すプロンプトが表示されます。

```
Your identification has been saved in C:\Keys\MyNewLighstailCustomKey.
Your public key has been saved in C:\Keys\MyNewLighstailCustomKey.pub.
```

   次に、キーペアの公開キーを Lightsail コンソールにアップロードします。

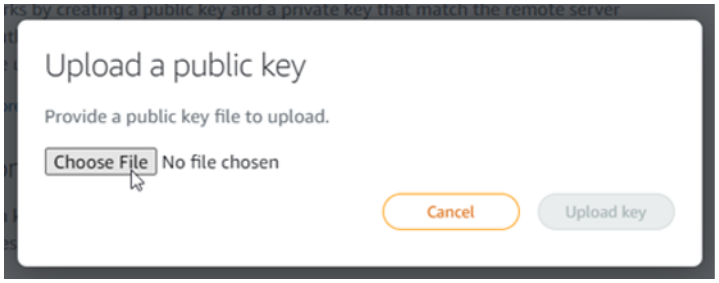
7. Lightsail コンソールにサインインします。
8. Lightsail ホームページの上部にあるナビゲーションペインで [Account] (アカウント) をクリックします。
9. ドロップダウンメニューで [Account (アカウント)] を選択します。



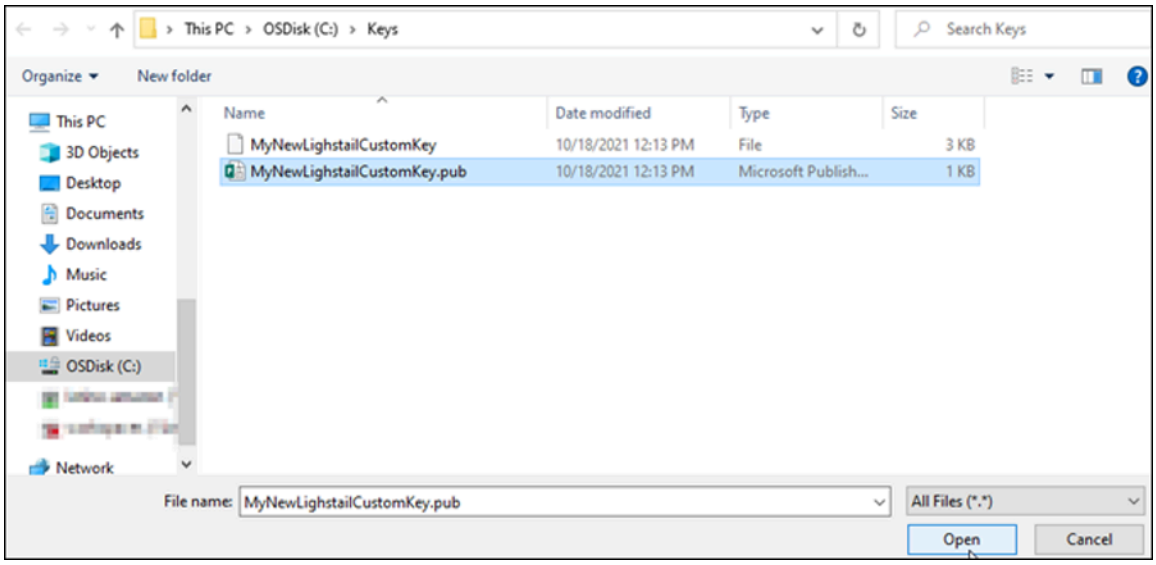
10. [SSH キー] タブを選択します。
11. そのページの [Custom keys] (カスタムキー) セクションで、[Upload key] (キーのアップロード) を選択します。

12. 表示される [Select a region] (リージョンの選択) プロンプトで、新しいカスタムキーをアップロードする AWS リージョンを選択します。同じ AWS リージョン内の新しいインスタンスには、新しいカスタムキーを設定することができます。

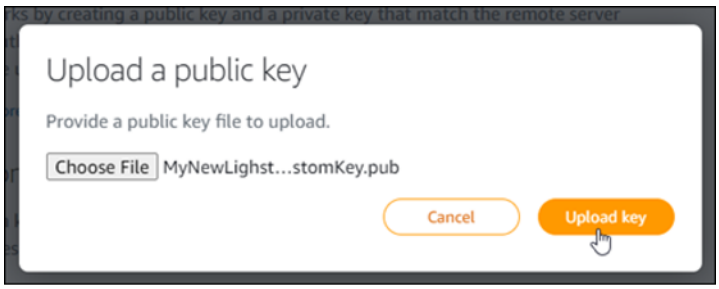
13. [アップロード] を選択します。
14. 表示される [Upload a public key] (公開キーのアップロード) プロンプトで、[Choose File] (ファイルを選択) をクリックします。



15. ローカルコンピュータで、この手順で先ほど作成したキーペアの公開キーを検索し、[Open] (開く) を選択します。キーペアの公開キーは、.PUB ファイル拡張子を持つファイルです。



16. [Upload key] (キーのアップロード) をクリックします。



17. 新しいカスタムキーは、このページの [Custom keys] (カスタムキー) セクションにリストされます。

キーをアップロードした AWS リージョンで作成する新しいインスタンスには、新しいカスタムキーを設定することができます。以前に作成された現在実行中のインスタンスで新しいカスタムキーを設定するには、「Amazon Lightsail 内のインスタンスに保存されているキーを管理する」を参照してください。

Lightsail インスタンスに保存されている SSH キーの管理

キーペアを使用することで、Amazon Lightsail インスタンスへのセキュアな接続を確立することができます。Lightsail では、Linux または Unix インスタンスの新規作成時に、インスタンスでキーペアの公開キーを設定します。インスタンスに対する SSH 接続を確立するときは、キーペアのプライベートキーを使用してインスタンスへの認証を行います。キーの詳細については、「キーペアとインスタンスへの接続」を参照してください。

インスタンスが実行状態になったら、インスタンスに新しい公開キーを追加する、またはインスタンスの公開キーを交換 (既存の公開キーを削除して新しいものを追加) することで、インスタンスへの接続に使用されるキーペアを変更できます。次の理由から、これが必要になる場合があります。

- 組織内のユーザーが個別のキーペアを使用してインスタンスにアクセスする必要がある場合は、インスタンスに公開キーを追加できます。
- 漏洩したキーを使用していたインスタンスのスナップショットから作成された新しいインスタンスをセキュア化する必要がある場合。
- 誰かがプライベートキーのコピーを持っており、その人物がインスタンスに接続できないようにしたい場合 (例えば、その人物が組織から脱退した場合など)、インスタンスの公開キーを削除して、新しいものに交換することができます。

インスタンスのキーペアを追加または交換するには、インスタンスに接続できる必要があります。既存のプライベートキーを紛失した場合は、Lightsail のブラウザベースの SSH クライアントを使用してインスタンスに接続できます。詳細については、「Linux または Unix インスタンスへの接続」を参照してください。

目次

ステップ 1: プロセスについて学ぶ

以下は、インスタンスでキーを追加および削除するためのおおまかな手順です。新しいキーを追加せずにインスタンスからキーを削除する場合は、本ガイド後述の「ステップ 5: Delete an existing public key from your instance」(インスタンスから既存の公開キーを削除する) を参照してください。

1. キーペアを作成する – インスタンスに新しいキーを追加するには、まず新しいキーペアを作成する必要があります。Lightsail コンソールを使用してカスタムもしくはデフォルトのキーペアを作成する、または ssh-keygen などのサードパーティーツールを使用してローカルコンピュータでそれらを作成できます。どちらの方法でも、公開キーとプライベートキーで構成される新しいキーペアが生成されます。詳細については、本ガイド後述の「ステップ 2: キーペアを作成する」を参照してください。
2. インスタンスに公開キーを追加する – キーペアを作成したら、SSH を使用してインスタンスに接続し、キーペアの公開キーをインスタンスに追加します。詳細については、本ガイド後述の「ステップ 3:Add a public key to your instance」(インスタンスに公開キーを追加する) を参照してください。
3. 新しいキーペアを使用してインスタンスに接続できることをテストする – キーペアの公開キーがインスタンスに保存されたら、SSH を使用したインスタンスへの接続にキーペアのプライベートキーを使用できることをテストする必要があります。詳細については、本ガイド後述の「ステップ 4: Connect to your instance using the new key pair」(新しいキーペアを使用してインスタンスに接続する) を参照してください。
4. インスタンスから古い公開キーを削除する – 新しいキーを使用したインスタンスへの接続が正常に行われたら、インスタンスから古い公開キーを削除できます。このステップを実行して、ユーザーが古いキーペアを使用してインスタンスに接続できないようにします。詳細については、本ガイド後述の「ステップ 5:インスタンスから既存の公開キーを削除する」を参照してください。

ステップ 2: キーペアを作成する

ssh-keygen を使用してローカルコンピュータでキーペアを作成するには、以下の手順を実行します。

1. ローカルコンピュータで、コマンドプロンプトまたはターミナルを開きます。
2. 次のコマンドを入力して、新しいキーペアを作成します。

```
ssh-keygen -t rsa
```

3. キーペアを保存するコンピュータのディレクトリの場所を指定します。

   以下はその例です。

   - Windows の場合: `C:\Users\<UserName>\.ssh\<KeyPairName>`
   - macOS、Linux、または Unix の場合: `/home/<UserName>/.ssh/<KeyPairName>`

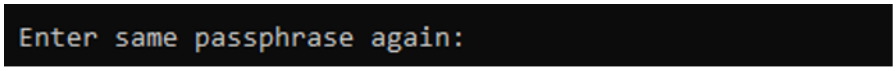
   <UserName> を現在サインインしているユーザーの名前に置き換えて、*<KeyPairName>* を新しいキーペアの名前に置き換えます。

   以下の例では、Windows コンピュータの `C:\Keys` ディレクトリを指定し、新しいキーに `MyNewLightsailCustomKey` という名前を付けました。

4. キーのパスフレーズを入力して、Enter を押します。パスフレーズの入力中にパスフレーズは表示されません。

   このパスフレーズは後ほど、公開キーが設定されているインスタンスに接続するために SSH クライアントでプライベートキーを設定するときに必要になります。

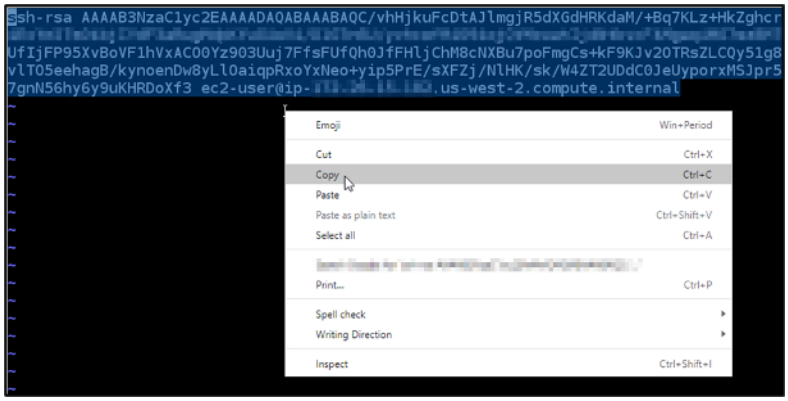
5. 確認のためパスフレーズをもう一度入力して、Enter を押します。パスフレーズの入力中にパスフレーズは表示されません。



6. 指定されたディレクトリにプライベートキーと公開キーが保存されたことを示すプロンプトが表示されます。

```
Your identification has been saved in C:\Keys\MyNewLighstailCustomKey.
Your public key has been saved in C:\Keys\MyNewLighstailCustomKey.pub.
```

7. 公開キー (.PUB) ファイルを開いて、ファイル内のテキストをコピーします。

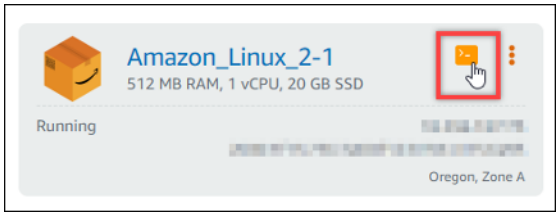


本ガイドの次のセクションに進み、Lightsail インスタンスに新しい公開キーを追加します。

ステップ 3: インスタンスに公開キーを追加する

インスタンスに公開キーを追加するには、次のステップを実行します。公開キーの内容は、Linux および Unix インスタンスの ~/.ssh/authorized_keys ファイルに保存されています。

1. Lightsail コンソールにサインインします。
2. Lightsail ホームページで [Instances] (インスタンス) タブを選択します。
3. 接続するインスタンスのブラウザベースの SSH クライアントアイコンをクリックします。



4. 接続されたら、任意のテキストエディタを使用して、authorized_keys ファイルを編集するための以下のコマンドを入力します。以下の手順では、デモ用に Vim を使用します。

```
sudo vim ~/.ssh/authorized_keys
```

インスタンス上で設定されている現在の公開キーは、次の例のような結果で表示されます。今回の場合、インスタンスが作成された AWS リージョンの Lightsail デフォルトキーが、インスタンスで設定されている唯一の公開キーになります。

5. [I] キーを押して、Vim エディタを挿入モードにします。
6. ファイルの最後の公開キーの後に改行を入力します。
7. このガイドの前のセクションで (新しいキーペアを作成した後) コピーした公開キーテキストを貼り付けます。以下の例のような結果が表示されるはずです。

8. ESC キーを押します。次に、:wq! を入力し、Enter キーを押して編集内容を保存して、Vim エディタを終了します。

これで、インスタンスに新しい公開キーが追加されました。本ガイドの次のセクションに進み、新しいキーペアを使用してインスタンスに接続します。

ステップ 4: 新しいキーペアを使用してインスタンスに接続する

新しいキーペアをテストするには、インスタンスとの接続を切断してから、このガイドで先ほど作成したプライベートキーを使用してインスタンスに再接続します。詳細については、「キーペアと Amazon Lightsail 内のインスタンスへの接続」を参照してください。新しいキーを使用してインスタンスに正常に接続したら、インスタンスから古いキーを削除できます。次のステップに進んで、インスタンスから公開キーを削除する方法を学びます。

ステップ 5: インスタンスから既存の公開キーを削除する

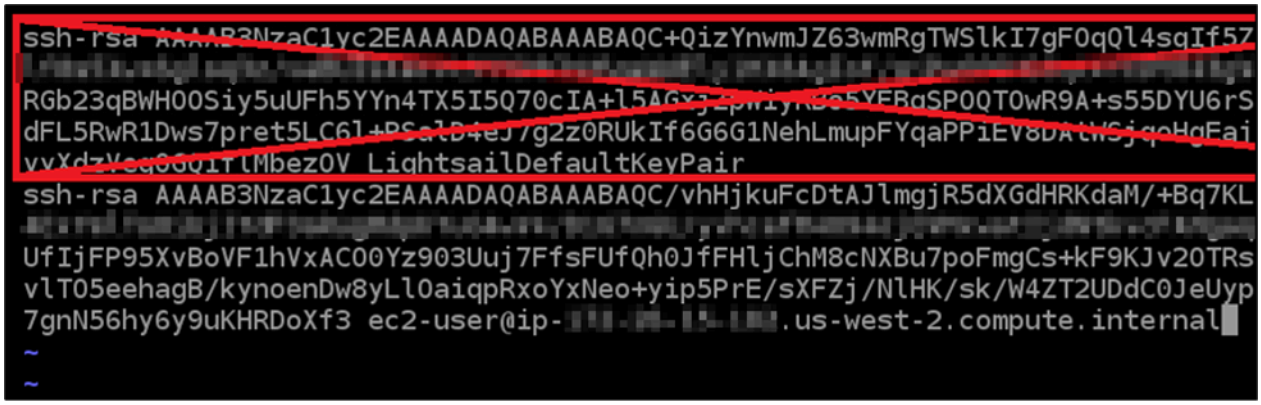
インスタンスから公開キーを削除するには、以下の手順を実行します。これは、ユーザーが古いキーペアを使用してインスタンスに接続できないようにします。この手順は、新しいキーペアを使用したインスタンスへの接続が正常に行われた後で実行してください。

1. SSH を使用してインスタンスに接続します。

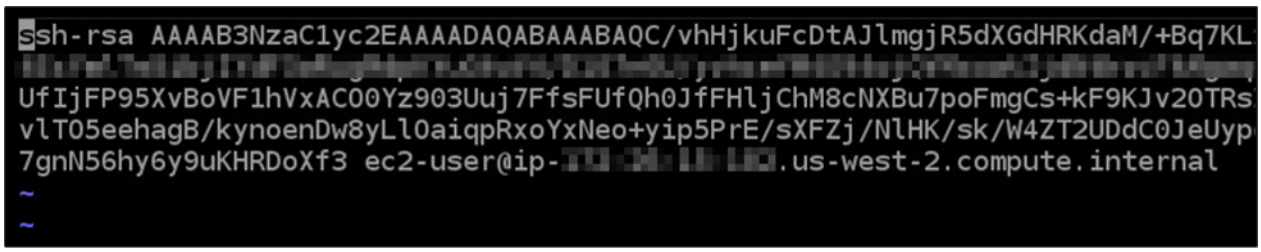
2. 任意のテキストエディアを使用して、authorized_keys ファイルを編集するための以下のコマンドを入力します。以下の手順では、デモ用に Vim を使用します。

```
sudo vim ~/.ssh/authorized_keys
```

3. I の文字キーを押して、Vim エディタを挿入モードにします。
4. インスタンスから削除したい公開キーを含んでいるテキストの行を削除します。



結果は以下の例のようになります。ここでは、新しい公開キーが、表示されている唯一のキーです。



5. ESC キーを押します。次に、:wq! を入力し、Enter キーを押して編集内容を保存して、Vim エディタを終了します。

削除された公開キーは、インスタンスから消去された状態です。インスタンスは、そのキーペアのプライベートキーを使用する接続を拒否します。

Lightsail 用の PuTTY をダウンロードしてセットアップする

PuTTY などの SSH クライアントを使用して Lightsail インスタンスに接続できます。PuTTY ではプライベート SSH キーのコピーが必要です。キーがすでにある場合や、Lightsail が作成するキーペアを使用したい場合があります。どちらの方法についても説明しています。SSH の詳細については、「SSH キーペア」を参照してください。このトピックでは、キーペアをダウンロードし、インスタンスに接続するように PuTTY をセットアップするステップについて順を追って説明します。

このガイドで説明するインスタンスへの接続方法は、多数あるうちの 1 つです。他の方法に関する詳細は、「SSH のキーペア」を参照してください。

Lightsail で Linux または Unix インスタンスに接続する最も簡単な方法は、Lightsail コンソールで使用できるブラウザベースの SSH クライアントを使用することです。詳細については、「Amazon Lightsail での Linux または Unix インスタンスへの接続」を参照してください。

前提条件

- Lightsail で実行中のインスタンスが必要です。詳細については、「Amazon Lightsail でインスタンスを作成する」を参照してください。
- 静的 IP アドレスを作成してインスタンスにアタッチすることをお勧めします。これにより、後でパブリック IP アドレスが変わった場合に、PuTTY を再設定する必要がなくなります。詳細については、「静的 IP を作成してインスタンスにアタッチする」を参照してください。

ステップ 1: PuTTY をダウンロードしてインストールする

PuTTY は、Windows 用の SSH の無料の実装です。暗号化が許可されていない国に関連する制限など、PuTTY の詳細については、PuTTY ウェブサイトの「」を参照してください。すでに PuTTY を持っている場合は、この手順をスキップしてステップ 2 に進むことができます。

1. PuTTY のインストーラまたは実行可能ファイルを、「PuTTY のダウンロード」からダウンロードします。

   どのダウンロードを選択すればいいか判断できない場合は、PuTTY のドキュメントを参照してください。最新バージョンをダウンロードすることをお勧めします。
2. ステップ 2 に進み、PuTTY を設定する前にプライベートキーを取得します。

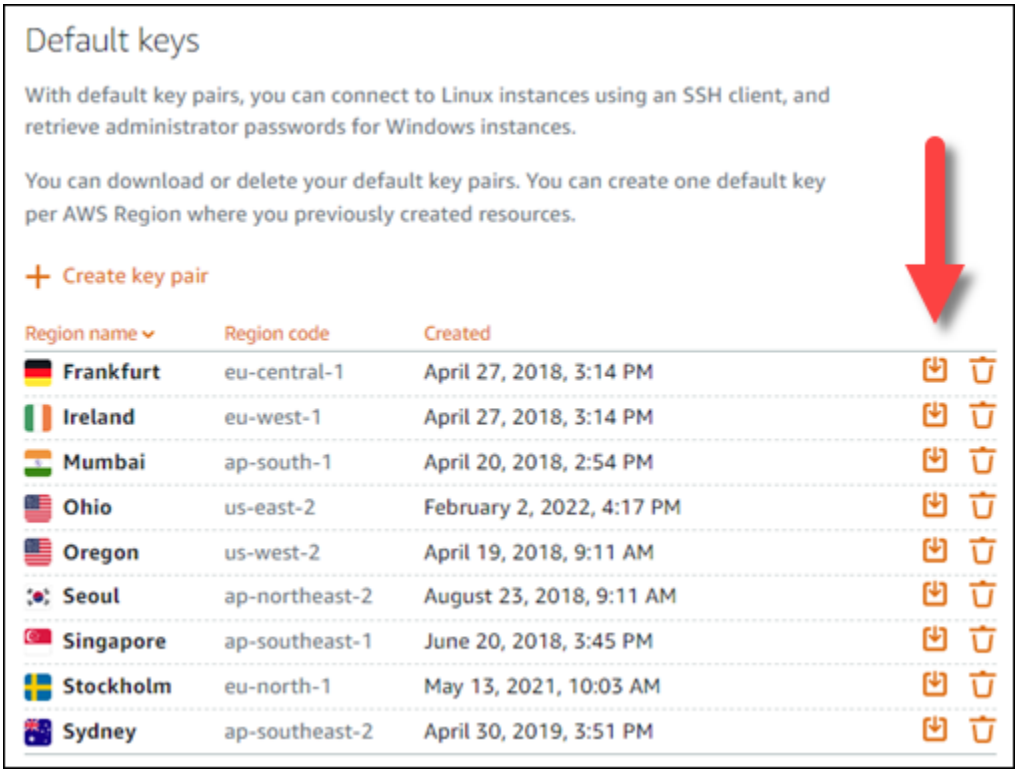
ステップ 2: プライベートキーの準備を整える

プライベートキーを取得する方法にはいくつかの選択肢があります。Lightsail が生成するデフォルトのプライベートキーを使用したり、Lightsail に新しいプライベートキーを作成させたり、別のサービスのプライベートキーを既に持っていたりすることができます。各オプションの手順については、以下に概要を示します。

1. Lightsail コンソールにサインインします。
2. 上部のナビゲーションバーで [アカウント] を選択し、ドロップダウンから [アカウント] を選択します。

3. [SSH キー] タブを選択します。
4. 使用するプライベートキーに応じて、次のいずれかのオプションを選択します。

   - Lightsail が生成するデフォルトのプライベートキーを使用するには、ページの「デフォルトキー」セクションで、インスタンス AWS リージョン が配置されている のデフォルトのプライベートキーの横にあるダウンロードアイコンを選択します。



   - Lightsail で新しいキーペアを作成するには、ページのカスタムキーセクションでキーペアの作成を選択します。インスタンス AWS リージョン が配置されている を選択し、 の作成 を選択します。名前を入力し、[Generate key pair] (キーペアの生成) を選択します。プライベートキーをダウンロードするためのオプションが表示されます。

     > **Important**
     >
     > プライベートキーは一度だけダウンロードすることができます。セキュリティで保護されている場所に保存します。

   - 独自のキーペアを使用するには、[今すぐアップロード] を選択します。インスタンス AWS リージョン が配置されている を選択し、 をアップロード を選択します。[Upload file (ファイルのアップロード)] を選択し、ローカルドライブのファイルを見つけます。パブリッ

クキーファイルを Lightsail にアップロードする準備ができたら、キーのアップロードを選択します。

5. プライベートキーをダウンロードした場合、または Lightsail で新しいプライベートキーを作成した場合は、簡単に見つけることができる場所に.pemキーファイルを保存してください。

   他のユーザーが読み取ることができないようにファイルのアクセス許可も設定することをお勧めします。

ステップ 3: Lightsail プライベートキーを使用して PuTTYgen を設定する

.pem キーファイルのコピーを作成したら、PuTTY キージェネレータ (PuTTYgen) を使用して PuTTY をセットアップできます。

1. PuTTYgen を起動します (例: [スタート] メニューで [すべてのプログラム]、[PuTTY]、[PuTTYgen] の順に選択します)。
2. ロード] を選択します。

   PuTTYgen では、デフォルトでは .ppk 拡張子を持つファイルだけが表示されます。.pem ファイルの場所を特定するには、すべてのタイプのファイルを表示するオプションを選択します。
3. lightsailDefaultKey.pem を選択して [Open (開く)] を選択します。

   キーの正常なインポートが PuTTYgen で確認されたら、[OK] を選択できます。
4. [Save private key (プライベートキーを保存)] を選択し、パスフレーズ付きで保存しないことを確認します。

   追加のセキュリティ対策としてパスフレーズを作成することを選択した場合は、PuTTY を使用してインスタンスに接続するたびにそのパスフレーズを入力する必要があります。
5. プライベートキーを保存する名前と場所を指定し、[Save (保存)] を選択します。
6. PuTTYgen を閉じます。

ステップ 4: プライベートキーとインスタンスの情報を使用して PuTTY の設定を完了する

もう少しです。これから最後の変更を行います。

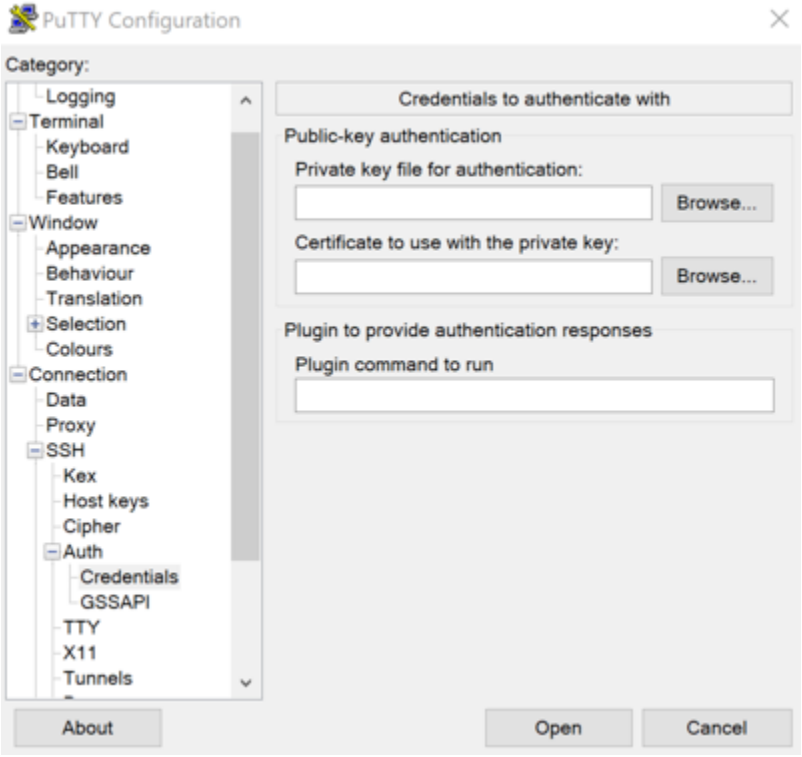
1. PuTTY を開きます。
2. Lightsail から、インスタンス管理ページからパブリック IP アドレスを取得します (通常は静的 IP アドレス を使用します) 。

Lightsail のホームページからパブリック IP アドレスを取得するか、インスタンスを選択して詳細を表示できます。

3. パブリック IP アドレスを [Host Name (or IP address) (ホスト名 (または IP アドレス))] フィールドに入力するか、貼り付けます。

> **Note**
>
> Lightsail インスタンスの SSH 用にポート 22 が既に開いているため、デフォルトのポートを受け入れます。

4. [接続] の下の [SSH] および [認証] を展開して、[認証情報] を選択します。



5. [Browse (参照)] を選択し、前のステップで作成した `.ppk` ファイルの場所に移動して選択し、[Open (開く)] を選択します。
6. [開く] を再度選択し、[承諾] を選択して、今後はこの接続を信頼することを示します。
7. インスタンスのオペレーティングシステムに応じて、次のいずれかのデフォルトのユーザー名を使用してログインします。

   - AlmaLinux、Amazon Linux 2、Amazon Linux 2023、CentOS Stream 9、FreeBSD、openSUSE インスタンス: `ec2-user`

- CentOS 7 インスタンス: `centos`
- Debian インスタンス: `admin`
- Ubuntu インスタンス: `ubuntu`
- Bitnami インスタンス: `bitnami`
- Plesk インスタンス: `ubuntu`
- cPanel & WHM インスタンス : `centos`

インスタンスのオペレーティングシステムの詳細については、「イメージの選択」を参照してください。

8. 後で使用できるように接続を保存します。

次のステップ

再度接続する必要がある場合は、「PuTTY を使用して Linux/Unix ベースのインスタンスに接続する」を参照してください。

Lightsail Windows インスタンスに接続する

Lightsail コンソールで使用可能なブラウザベースの RDP クライアントを使用して、Amazon Lightsail の Windows Server インスタンスに接続できます。ブラウザベースの RDP クライアントはソフトウェアのインストールを必要とせず、Windows Server インスタンスにその作成後すぐに接続でき、そのインスタンスを使用できます。インスタンスに接続し、ソフトウェアのインストールやウェブアプリケーションの設定などの管理タスクをサーバーで実行します。

> **Important**
>
> Lightsail ブラウザベースの SSH/RDP クライアントは IPv4 トラフィックのみを受け入れます。サードパーティーのクライアントを使用して、IPv6 経由でインスタンスに SSH または RDP 接続します。詳細については、「インスタンスに接続します」を参照してください。

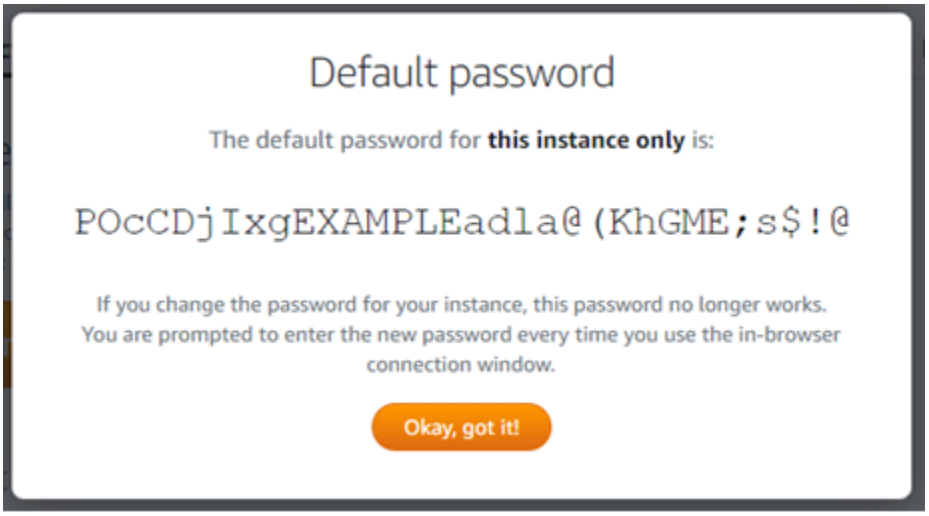
独自の RDP クライアントを使用して、Windows にバンドルされているリモートデスクトップ接続などのインスタンスに接続することもできます。独自の RDP クライアントの設定の詳細については、「リモートデスクトップ接続クライアントを使用しての Windows インスタンスへの接続」を参照してください。Lightsail で Linux または Unix インスタンスに接続するには、「Linux または Unix インスタンスに接続する」を参照してください。

Windows Server インスタンスのデフォルトの管理者パスワード

ランダムに生成されたデフォルトの管理者パスワードは、Windows Server インスタンスにその作成時に割り当てられます。Lightsail コンソールのブラウザベースの RDP クライアントは、デフォルトの管理者パスワードを使用してインスタンスにサインインします。インスタンスの管理者パスワードを変更すると、ブラウザベースの RDP クライアントを使用してインスタンスに接続しようとするたびに、新しいパスワードを手動で入力するように指示が出ます。Lightsail は新しい管理者パスワードを保存せず、インスタンスから取得することもできません。

> **Important**
>
> 管理者パスワードを紛失した場合、インスタンスにサインインできなくなり、パスワードをリセットできなくなります。新しい管理者パスワードは、紛失した場合に後で取得できる安全な場所 (AWS Secrets Manager など) に保管してください。詳細は、「AWS Secrets Manager ユーザーガイド」を参照してください。

管理者パスワードを元のデフォルトの管理者パスワードに変更することで、ブラウザベースの RDP クライアントを使用してインスタンスにアクセスするたびに、パスワードの入力を求められないようにできます。Lightsail ホームページのインスタンスタブを選択すると、元のデフォルトの管理者パスワードを見つけることができます。以下の例に示すように、Windows Server インスタンスの名前を選択し、[Connect (接続)] タブを選択し、[デフォルトのパスワードを表示] を選択して、元のデフォルトの管理者パスワードを表示します。

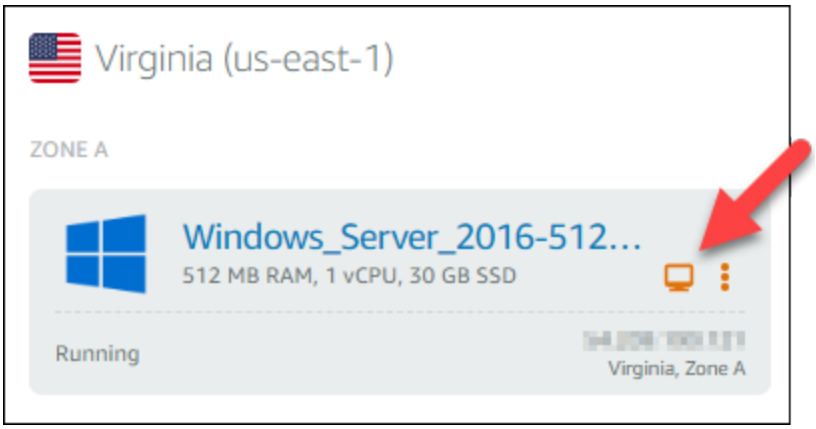


ブラウザベースの RDP クライアントを使用して Windows Server インスタンスに接続する

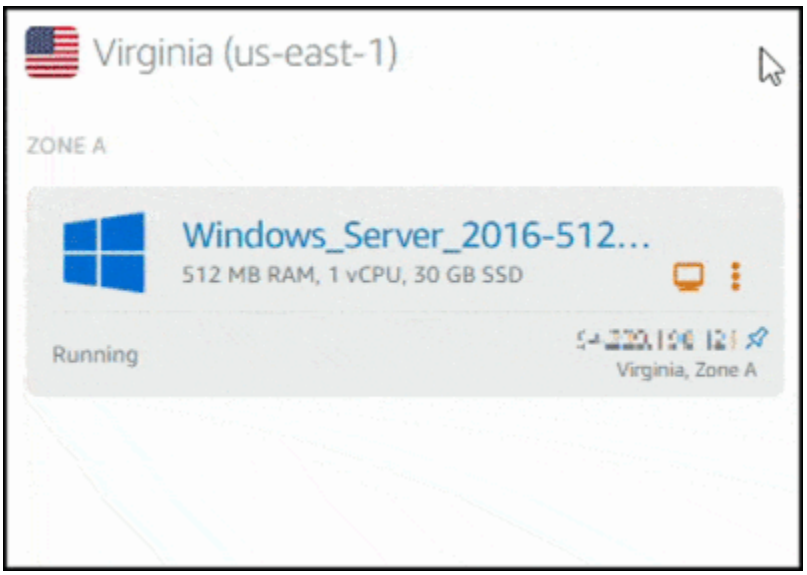
Lightsail コンソールでブラウザベースの RDP クライアントを使用して Windows Server インスタンスに接続するには、次の手順に従います。

1. Lightsail コンソールにサインインします。
2. 接続先のインスタンスのブラウザベースの RDP クライアントにアクセスするには、以下のいずれかのステップを実行します。

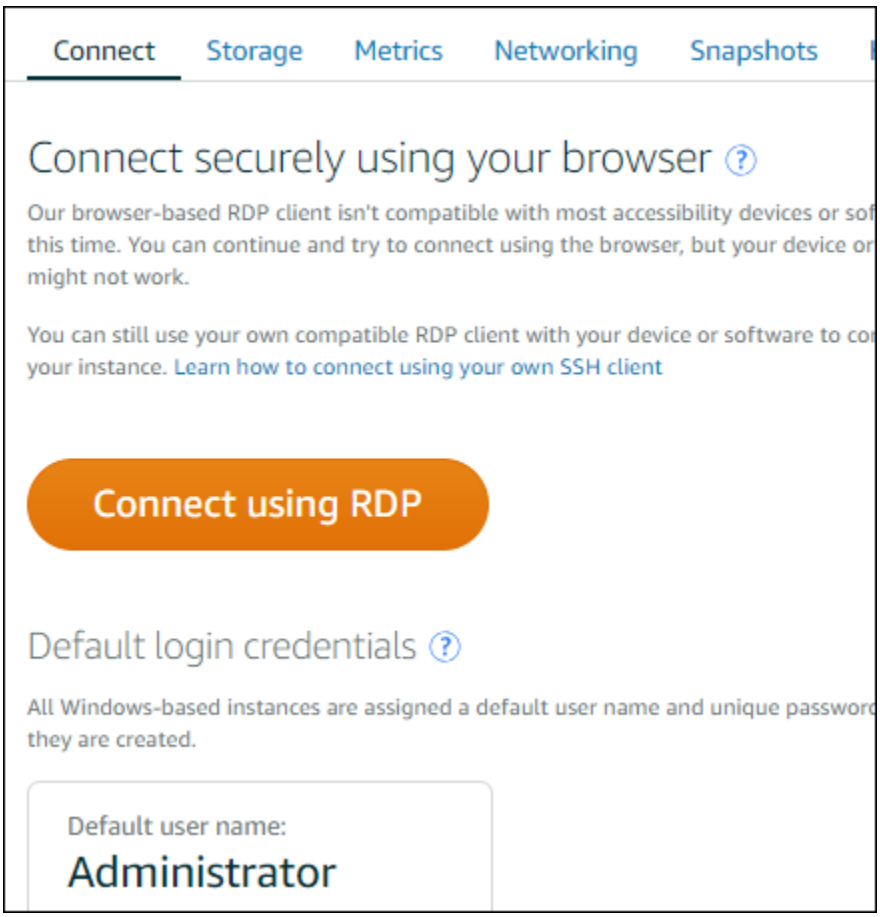
   - 以下の例に示すように、ブラウザベースの RDP クライアントアイコンを選択します。



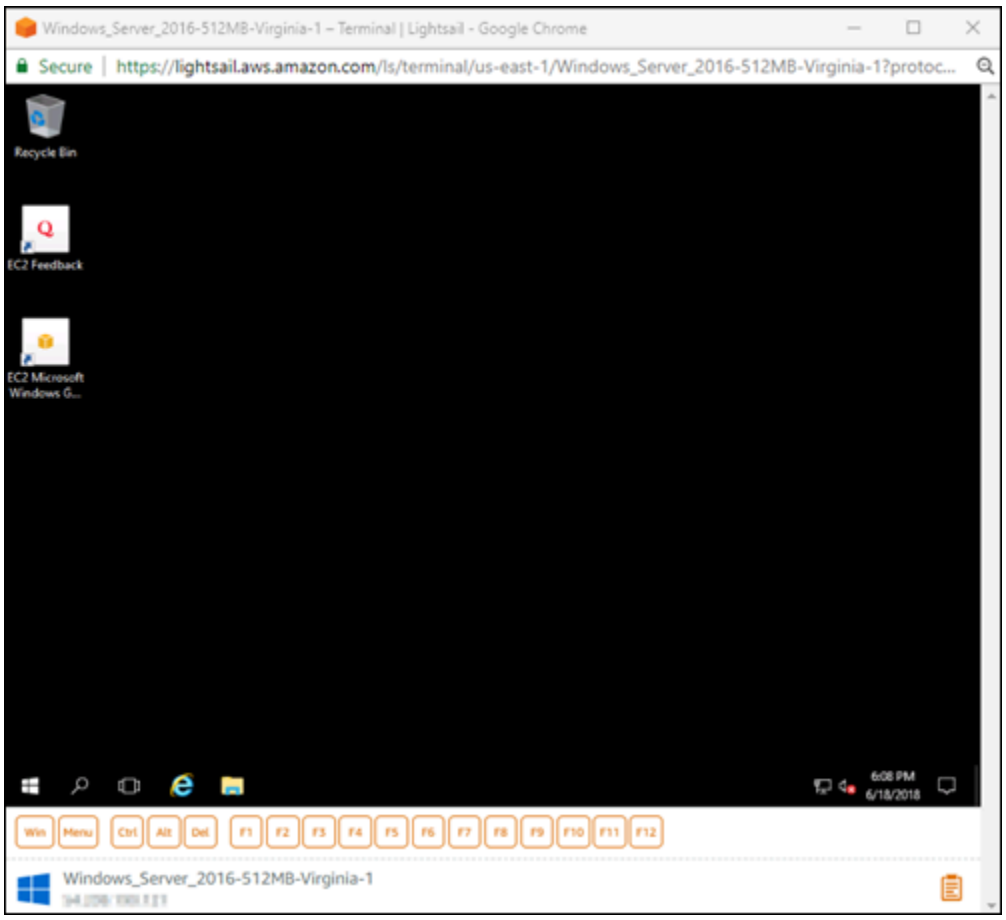
   - 以下の例に示すように、アクションメニューアイコン (⋮) を選択してから、[接続] を選択します。



   - インスタンスの名前を選択し、[接続] タブの [RDP を使用して接続] を選択します。



ブラウザベースの RDP クライアントが開いて、以下の例に示すような Windows デスクトップが表示されると、インスタンスとのやり取りを開始できます。



> **Note**
>
> [接続] タブには、独自の RDP クライアントを使用して接続するために必要な情報 (Windows インスタンスのデフォルトのユーザー名やパスワードなど) も表示されます。独自の RDP クライアントの設定の詳細については、「リモートデスクトップ接続クライアントを使用した Amazon Lightsail での Windows インスタンスへの接続」を参照してください。

ブラウザベースの RDP クライアントを使用して Windows インスタンスとやり取りする

ブラウザベースの RDP クライアントは、独自のローカル Windows デスクトップと同じように使用します。RDP には、インスタンスとのやり取りに役立つ Windows 固有のキー (ファンクションキーなど) が含まれています。以下のセクションでは、RDP でクリップボードに (またはクリップボードから) テキストをコピーして貼り付ける方法を示します。

ブラウザベースの RDP クライアントにテキストを貼り付けるには

1. ローカルデスクトップのテキストを強調表示し、Ctrl+C または Cmd+C を押してテキストをローカルクリップボードにコピーします。
2. ブラウザベースの RDP クライアントの右下で、クリップボードアイコンを選択します。ブラウザベースの RDP クライアントのクリップボードテキストボックスが表示されます。
3. テキストボックス内をクリックし、Ctrl+V または Cmd+V を押して、ローカルクリップボードの内容をブラウザベースの RDP クライアントのクリップボードに貼り付けます。
4. リモートデスクトップ画面の任意の領域を右クリックし、ブラウザベースの RDP クライアントのクリップボードからリモートデスクトップ画面にテキストを貼り付けます。

ブラウザベースの RDP クライアントからテキストをコピーするには

1. リモートデスクトップ画面でテキストを強調表示します。
2. ブラウザベースの RDP クライアントの右下で、クリップボードアイコンを選択します。ブラウザベースの RDP クライアントのクリップボードテキストボックスが表示されます。

3. コピーするテキストを強調表示し、Ctrl+C または Cmd+C を押してテキストをローカルクリップボードにコピーします。これで、コピーしたテキストをローカルデスクトップの任意の場所に貼り付けることができます。

Lightsail Windows インスタンスの管理者パスワードの変更

Windows Server ベースの Lightsail インスタンスを作成するときは、インスタンスを作成する AWS リージョン のデフォルトのパスワードを使用します。これにより、ブラウザベースのリモートデスクトップ (RDP) クライアントと、リモートデスクトップ接続などのクライアントを使用して接続しやすくなります。

> **Important**
>
> Lightsail がインスタンスのパスワードを生成できるようにすることを強くお勧めします。カスタムパスワードは保存されないため、管理者パスワードを変更した場合、Lightsail インスタンスへのアクセスが失われるリスクがあります。

Windows Server を使用して管理者パスワードを変更する

Windows Server の [パスワードの変更] ツールを使用して管理者パスワードを変更できます。Windows Server ベースの Lightsail インスタンスで `Ctrl` + `Alt` + `Del` を押し、[パスワードの変更] を選択します。

キーを復号する

Windows Server ベースの Lightsail インスタンスでパスワードを変更した場合、AWS Command Line Interface (AWS CLI) を使用してパスワードの復号に役立つ情報を取得できます。

AWS CLI を使用して暗号化テキストを取得する

1. まだ AWS CLI をインストールして設定していない場合は、インストールして設定します。

   詳細については、「Amazon Lightsail で使用するために AWS Command Line Interface を設定する」を参照してください。
2. コマンドプロントまたはターミナルを開きます。
3. 次のコマンドを入力します。

   ```
   aws lightsail get-instance-access-details --instance-name my-instance
   ```

   ここで、*my-instance* は情報を取得するインスタンスの名前です。

   以下のような出力結果が表示されるはずです。

   ```
   {
       "accessDetails": {
           "username": "Administrator",
           "protocol": "rdp",
           "ipAddress": "12.345.678.910",
           "passwordData": {
               "ciphertext": "cipher",
               "keyPairName": "my-ohio-key"
           },
           "password": "",
           "instanceName": "2016-ohio-windows"
       }
   }
   ```
4. 暗号化テキストは、使用可能な任意のアプリケーションでパスワードの復号化に使用できます。

リモートデスクトップ接続を使用して、Window から Lightsail Windows インスタンスに接続するには

Windows オペレーティングシステムに含まれているリモートデスクトップ接続 (RDC) クライアントを使用して、Amazon Lightsail の Windows インスタンスに接続できます。RDC では、Windows インスタンスの管理者ユーザー名とパスワードを使用する必要があります。この場合のパスワードは、インスタンスの作成時にインスタンスに割り当てられるデフォルトのパスワード、またはデフォルトのパスワードを変更した場合は自分のパスワードです。

このトピックでは、Lightsail コンソールからデフォルトの管理者パスワードを取得し、Windows インスタンスに接続するように RDC を設定するステップについて説明します。ブラウザを使用して、Lightsail コンソールからインスタンスに接続することもできます。詳細については、「ウェブベースの RDP クライアントを使用して Windows インスタンスに接続する」を参照してください。

Windows インスタンスのデフォルトの管理者パスワードを取得する

次のステップに従って、RDC を使用してインスタンスに接続するために必要な Windows インスタンスのデフォルトの管理者パスワードを取得します。

> **Note**
>
> デフォルトの管理者パスワードを変更した場合、インスタンスの Lightsail コンソールに表示されるパスワードは機能しません。パスワードは覚えておく必要があります。RDC で管理者パスワードを使用せずにインスタンスに接続することはできません。

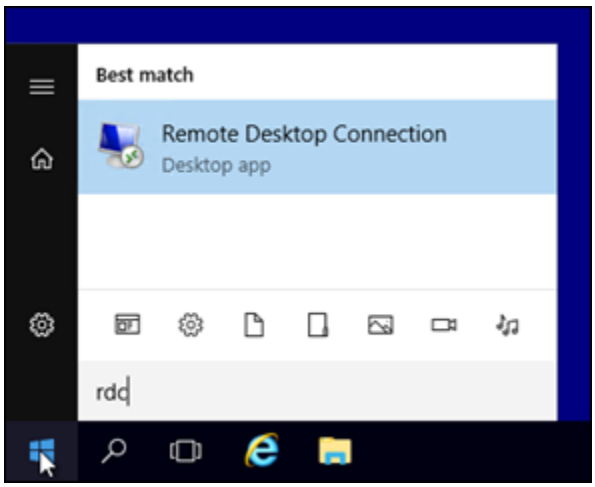
1. Lightsail コンソールにサインインします。
2. 接続先の Windows インスタンスを選択します。
3. インスタンス管理ページの [接続] タブで、[デフォルトのパスワードを表示] を選択します。
4. 表示されるデフォルトのパスワードをハイライト表示し、[Ctrl+C] または [Cmd+C] を押してコピーします。これで、パスワードがクリップボードにコピーされます。

   このガイドの次のセクションに進んで RDC を設定し、パスワードをクライアントに貼り付けます。

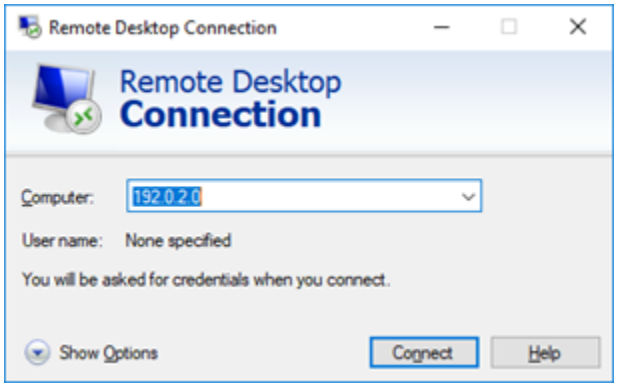
RDC を設定し、Windows インスタンスに接続する

以下のステップに従って、RDC を設定し、Windows インスタンスに接続します。

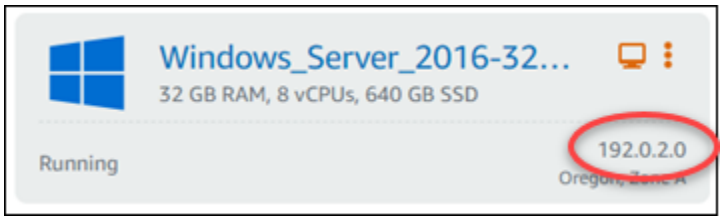
1. Windows メニューを開き、`Remote Desktop Connection` または `RDC` を検索します。
2. 検索結果の [リモートデスクトップ接続] を選択します。



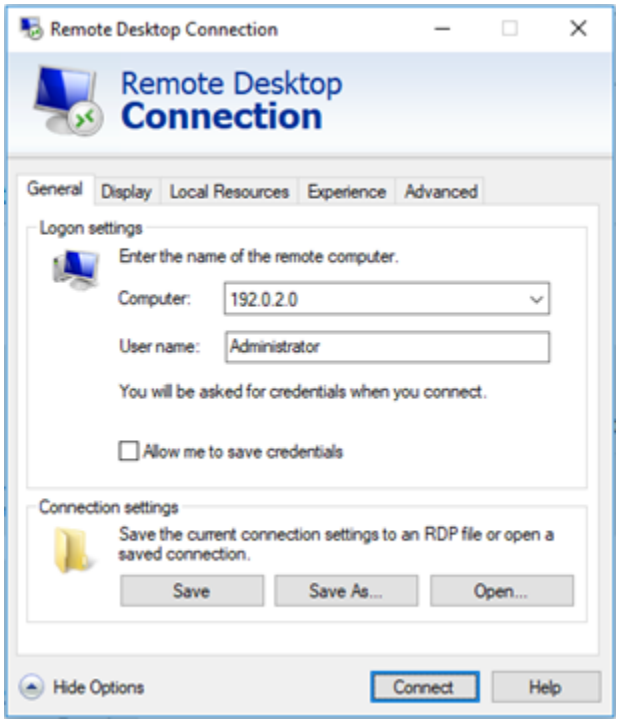
3. [コンピュータ] テキストボックスに、Windows インスタンスのパブリック IP アドレスを入力します。



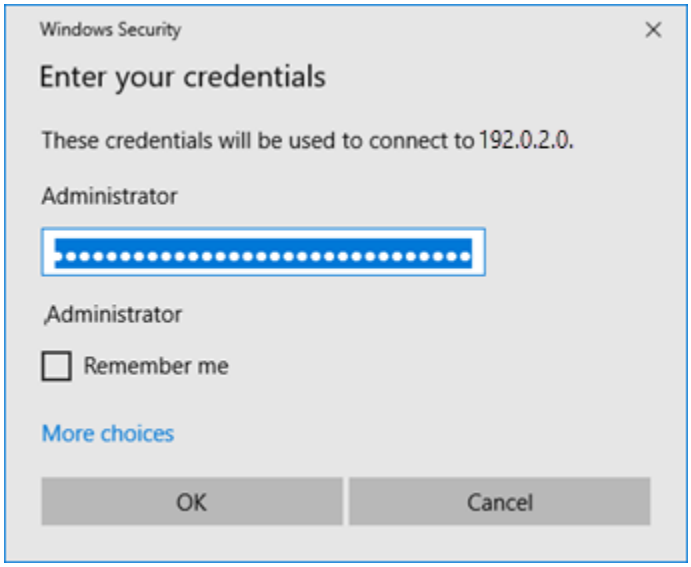
次の例に示すように、パブリック IP は、Lightsail コンソールのインスタンスの横に表示されます。



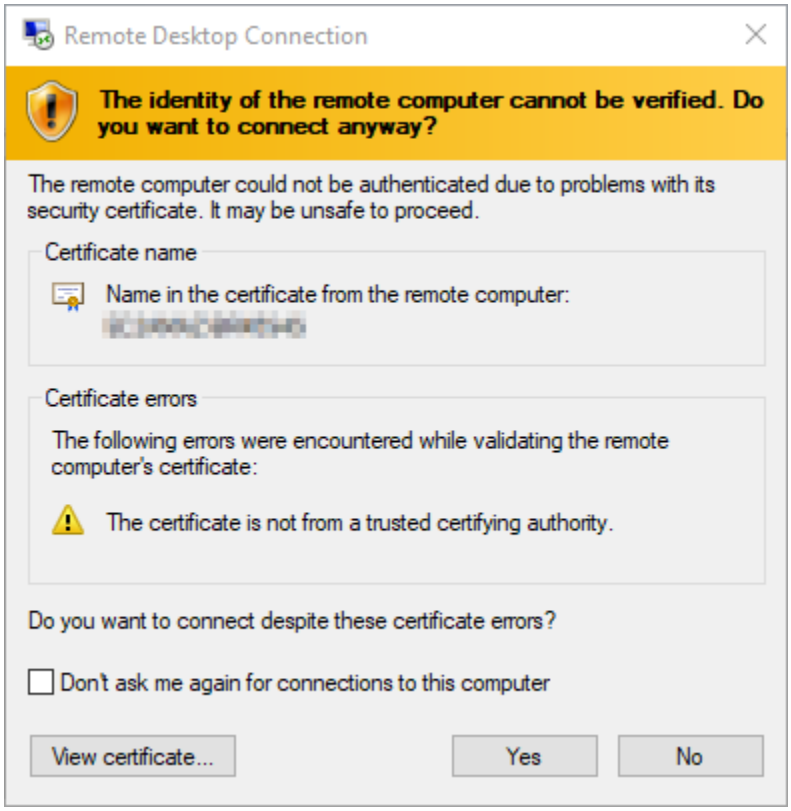
4. [オプションの表示] を選択して追加の接続オプションを表示します。
5. [User Name] (ユーザー名) テキストボックスに `Administrator` を入力します。これは、Lightsail 内のすべての Windows インスタンスのデフォルトユーザー名です。



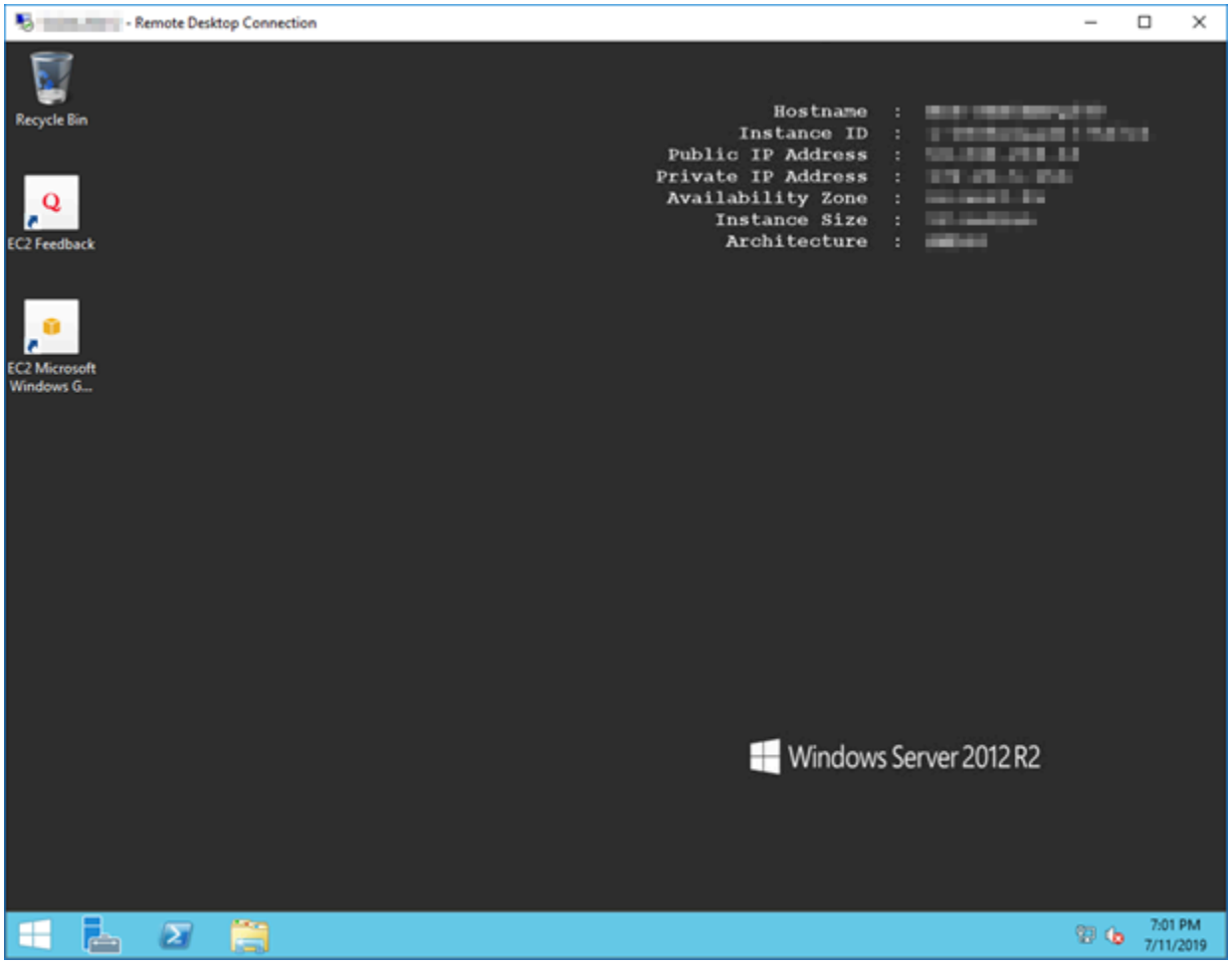
6. [Connect] (接続) を選択します。
7. 表示されるプロンプトで、この手順の前半で Lightsail コンソールからコピーしたデフォルトの管理者パスワードを入力するか貼り付け、[OK] を選択します。



8. 表示されるプロンプトで、[はい] を選択して Windows インスタンスに接続します。証明書エラーが出ても無視します。



インスタンスに接続されると、次の例のような画面が表示されます。



リモートデスクトップ接続を使用して macOS から Lightsail Windows インスタンスに接続する

Microsoft リモートデスクトップクライアントを使用して、macOS コンピュータから Windows インスタンスに接続することができます。Microsoft リモートデスクトップでは、Lightsail Windows インスタンスに管理者ユーザー名とパスワードを使用する必要があります。パスワードは、インスタンスの作成時に割り当てられたデフォルトのパスワード、またはデフォルトのパスワードを変更した場合は独自のパスワードを使用できます。

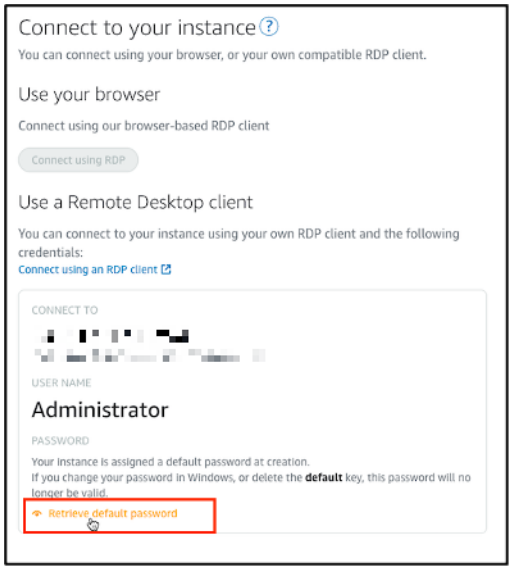
このトピックでは、Lightsail コンソールからデフォルトの管理者パスワードを取得し、Windows インスタンスに接続するように Microsoft リモートデスクトップを設定する手順について説明します。ブラウザを使用して Lightsail コンソール内からインスタンスに接続することもできます。詳細については、「Microsoft リモートデスクトップクライアントを使用して Windows インスタンスに接続する」を参照してください。

Windows インスタンスについて必要な接続情報を取得する

Microsoft リモートデスクトップクライアントを使用して Windows インスタンスに接続するには、そのインスタンスのパブリック IP アドレス、ユーザー名、および管理者パスワードが必要になります。

必要な情報を取得するには、以下の手順を実行します。

1. Lightsail コンソールにサインインします。
2. Lightsail ホームページで [Instances] (インスタンス) タブを選択します。
3. 接続するインスタンスのパブリック IP アドレスをメモします。
4. 接続するインスタンスの名前を選択します。
5. [Connect] (接続) タブを選択します。
6. [Show default password] (デフォルトのパスワードを表示) をクリックして、インスタンスの Windows 管理者パスワードを取得します。



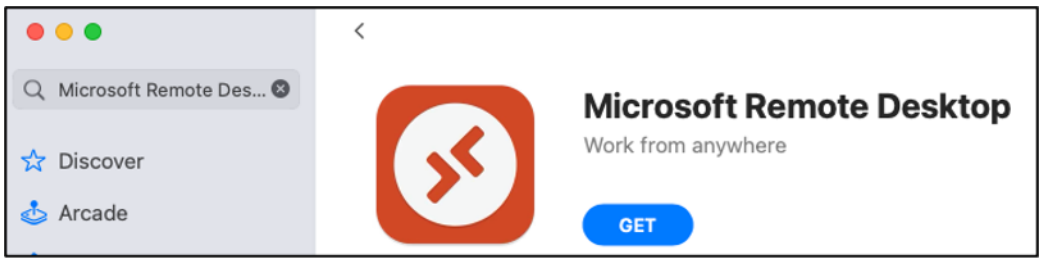
プロンプトに Windows インスタンスのデフォルト管理者パスワードが表示されます。

7. 管理者パスワードをコピーします。これは、本ガイドの後半で Microsoft リモートデスクトップクライアントを使用してインスタンスにサインインするために使用します。

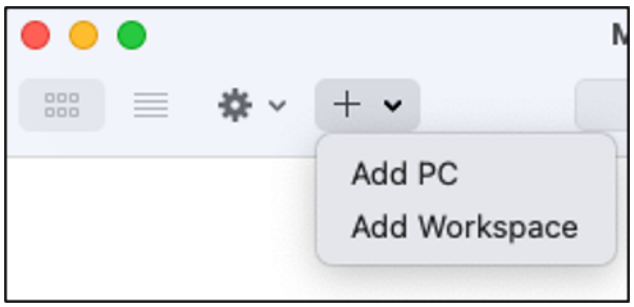
Microsoft リモートデスクトップを設定してインスタンスに接続する

Mac に Microsoft リモートデスクトップクライアントをインストールして、インスタンスに接続するようにクライアントを設定するには、以下の手順を実行します。

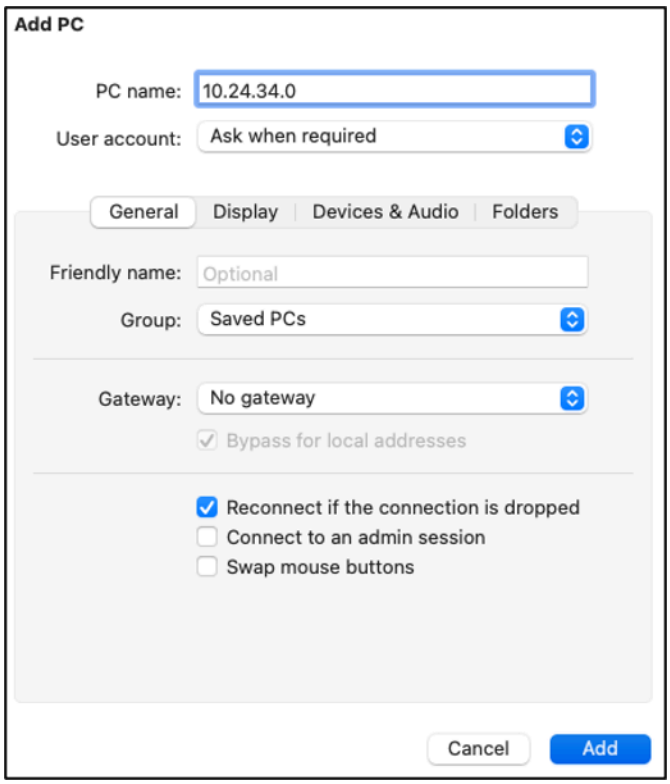
1. Mac で App Store を開き、[Microsoft Remote Desktop] (Microsoft リモートデスクトップ) を検索します。
2. 検索結果で [Microsoft Remote Desktop] (Microsoft リモートデスクトップ) を見つけ、[GET] (入手) をクリックしてアプリケーションをインストールします。



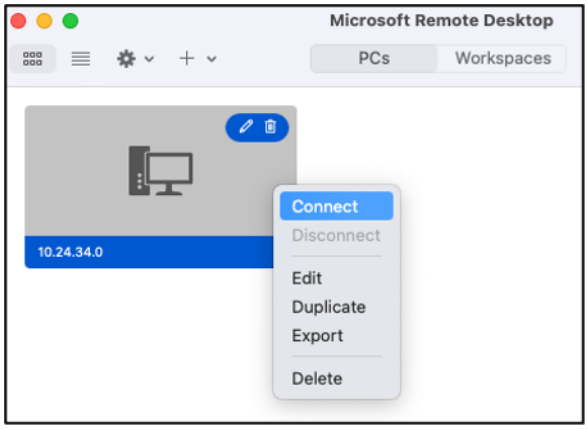
3. インストールが完了したら、[Microsoft Remote Desktop] (Microsoft リモートデスクトップ) を開きます。
4. 上部で [+] アイコンを選択し、[PC の追加] を選択します。



5. [PC name] (PC 名) テキストボックスに、インスタンスのパブリック IP アドレスを貼り付けます。
6. [追加] を選択します。

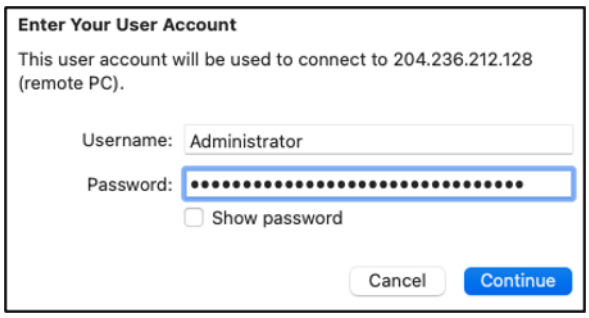


7. インスタンスのアイコンを右クリックし、[Connect] (接続) を選択します。

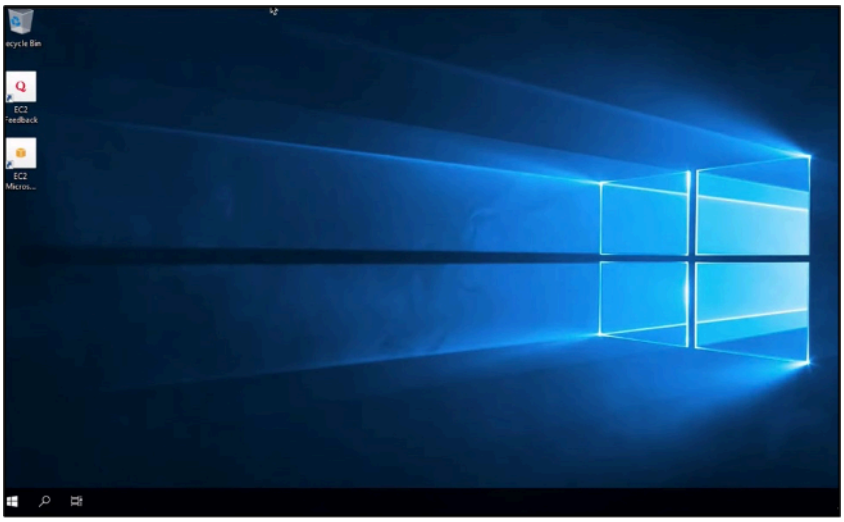


8. [Username]] (ユーザーネーム) テキストボックスに [Administrator (管理者) と入力し、[Password] (パスワード) テキストボックスに本ガイドで先ほど取得したデフォルトの管理者パスワードを入力します。

9. [Connect] (接続) をクリックしてインスタンスに接続します。



これで Lightsail Windows インスタンスに接続されました。



Linux または Unix Lightsail インスタンスのスナップショットを作成する

Linux/Unix ベースの Lightsail インスタンスのスナップショットを作成できます。インスタンススナップショットはシステムディスクのコピーであり、元のマシン構成 (メモリ、CPU、ディスクサイズ、およびデータ転送レート) を複製したものです。ブロックストレージディスクをインスタンスにアタッチした場合、Lightsail はスナップショットの一部としてそれらの追加のディスクをコピーします。詳細については、「スナップショット」を参照してください。

> **Note**
>
> Windows Server ベースの Lightsail インスタンスのスナップショットを作成する手順は異なります。詳細については、「Windows Server インスタンスのスナップショットを作成する」を参照してください。

スナップショットを作成するには、Lightsail でインスタンスが作成されている必要があります。インスタンスの準備が整ったら、次の手順に従ってスナップショットを作成します。

1. Lightsail のホームページで、スナップショットを作成するインスタンスの名前を選択します。
2. [スナップショット] タブを選択します。
3. このページの [手動スナップショット] セクションで、[スナップショットの作成] を選択し、スナップショットの名前を入力します。

   リソース名:

   - Lightsail アカウントの各 AWS リージョン 内で一意である必要があります。
   - 2〜255 文字を使用する必要があります。
   - 先頭と末尾は英数字または数字を使用する必要があります。
   - 英数字、数字、ピリオド、ダッシュ、アンダースコアを使用することができます。
4. [Create] (作成) を選択します。

   作成したスナップショットのステータスが [Snapshotting... (スナップショットの作成中)] と表示されることがあります。

   スナップショットが完了したら、スナップショットから別のインスタンスを作成することができます。たとえば、以前のものより大きいサイズのバンドルを選択できます。

> **Important**
>
> Lightsail では、スナップショットから新しいインスタンスを作成するときに、同じサイズまたはより大きいサイズのインスタンスバンドルを作成できます。現在は、スナップショットより小さいサイズのインスタンスの作成はサポートされていません。スナップショットから新しいインスタンスを作成する際、より小さいオプションは灰色で表示されます。

スナップショットを使用して、より大きいインスタンスサイズを作成するには、Lightsail コンソールで、CLI の [create-instances-from-snapshot] CLI コマンドを使用するか、[CreateInstancesFromSnapshot] API オペレーションを使用します。詳細については、「スナップショットからインスタンスを作成する」を参照してください。

Lightsail バンドルについて詳しくは、「Lightsail 料金表」を参照してください。

トピック

- Amazon Lightsail スナップショットから作成された Amazon EC2 の Linux または Unix インスタンスに接続する
- Lightsail スナップショットから作成された Amazon EC2 の Windows Server インスタンスに接続する
- Lightsail Windows Server インスタンスのスナップショットを作成する
- Lightsail スナップショットから作成された Amazon EC2 の Windows サーバーインスタンスを保護する
- Lightsail スナップショットから作成した Amazon EC2 の Linux または Unix インスタンスを保護する方法について説明します。

Amazon Lightsail スナップショットから作成された Amazon EC2 の Linux または Unix インスタンスに接続する

Amazon Lightsail スナップショットから Amazon Elastic Compute Cloud (Amazon EC2) に Linux または Unix インスタンスを作成したら、ソース Lightsail インスタンスへの接続方法と同様に、SSH 経由でインスタンスに接続できます。インスタンスを認証するには、ソースインスタンスの にデフォルトの Lightsail キーペアを使用するか AWS リージョン、独自のキーペアを使用します。このガイドでは、PuTTY を使用して EC2 の Linux または Unix インスタンスに接続する方法を示します。

> **Note**
>
> Windows Server インスタンスへの接続の詳細については、「Lightsail スナップショットから作成された Amazon EC2 Windows Server インスタンスに接続する」を参照してください。

目次

- 次のステップ

インスタンスのキーを取得する

新しい Amazon EC2 インスタンスに接続するために必要な適切なキーを取得します。必要なキーは、ソース Lightsail インスタンスへの接続方法によって異なります。ソースの Lightsail インスタンスへの接続方法としては以下が挙げられます。

- ソースインスタンスのリージョンでデフォルトの Lightsail キーペアを使用する — Lightsail アカウントページの SSH キータブからデフォルトのプライベートキーをダウンロードします。デフォルトの Lightsail キーの詳細については、「SSH キーペア」を参照してください。

  > **Note**
  >
  > EC2 インスタンスに接続したら、インスタンスからデフォルトの Lightsail キーを削除し、独自のキーペアに置き換えることをお勧めします。詳細については、「Lightsail スナップショットから作成された Amazon EC2 の Linux または Unix インスタンスを保護する」を参照してください。

- 独自のキーペアを使用する – プライベートキーを見つけて Amazon EC2 インスタンスへの接続に使用します。Lightsail は、独自のキーペアを使用する場合、プライベートキーを保存しません。プライベートキーをなくすと、Amazon EC2 インスタンスに接続できなくなります。

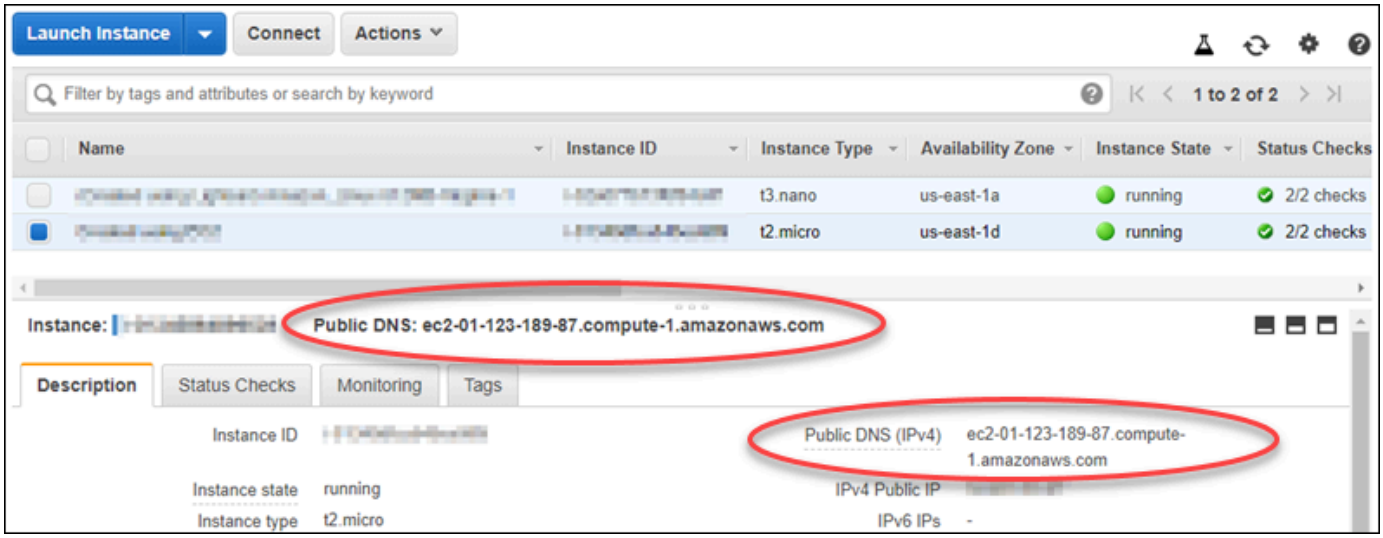
インスタンスのパブリック DNS アドレスを取得する

Amazon EC2 インスタンスのパブリック DNS アドレスを取得し、これを SSH クライアント (PuTTY など) の設定時に使用してインスタンスに接続します。

インスタンスのパブリック DNS アドレスを取得するには

1. Amazon EC2 コンソールにサインインします。
2. 左側のナビゲーションペインから、[インスタンス] を選択します。
3. 接続先である実行中の Linux または Unix をインスタンスを選択します。
4. 下部のペインで、インスタンスのパブリック DNS アドレスを見つけます。

   このアドレスを SSH クライアントの設定時に使用してインスタンスに接続します。このガイドの「PuTTY をダウンロードしてインストールする」セクションに進み、PuTTY SSH クライアントをダウンロードしてインストールします。



PuTTY をダウンロードしてインストールする

PuTTY は Windows 用の無料の SSH クライアントです。詳細については、「PuTTY: a free SSH and Telnet client」を参照してください。このウェブサイトでは、暗号化を許可しない諸国での制限についても説明しています。すでに PuTTY がある場合は、このガイドの後述の「PuTTYgen を使用してキーを設定する」セクションに進むことができます。

PuTTY インストーラまたは実行可能ファイルをダウンロードします。最新バージョンをダウンロードすることをお勧めします。ただし、どのダウンロードを選択するかの詳細については、PuTTY のドキュメントを参照してください。

このガイドの「PuTTYgen を使用してキーを設定する」セクションに進み、PuTTYgen でキーを設定します。

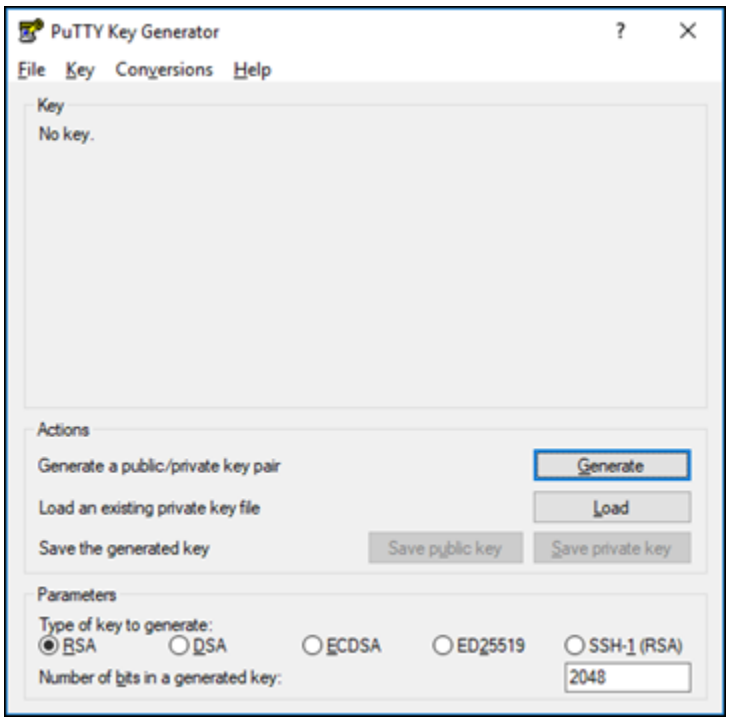
PuTTYgen を使用してキーを設定する

PuTTYgen は、PuTTY で使用するパブリックキーとプライベートキーのペアを生成します。このステップは、PuTTY が受け入れるキーファイルタイプ (.PPK) を使用するために必要です。

PuTTYgen を使用してキーを設定するには

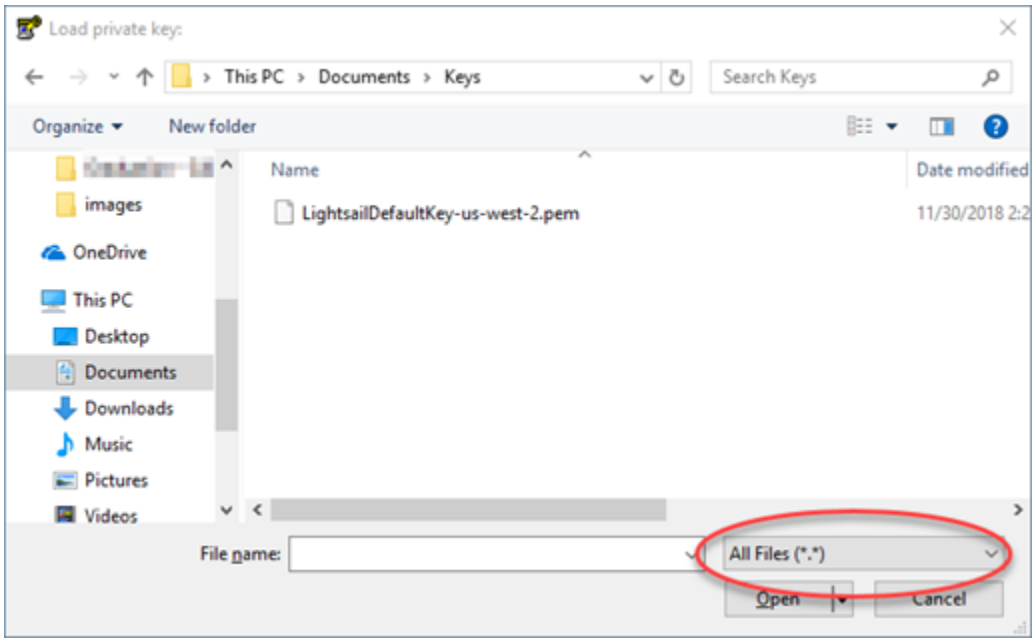
1. PuTTYgen を起動します。

   たとえば、Windows のスタートメニューで、[すべてのプログラム]、[PuTTY]、[PuTTYgen] の順に選択します。



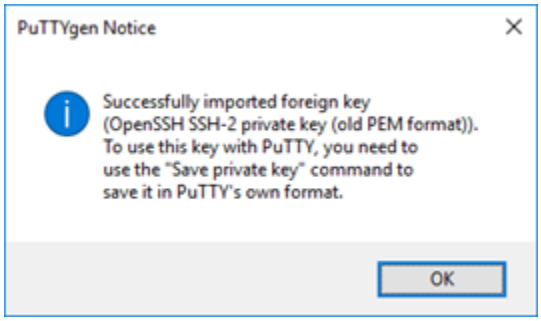
2. ロード] を選択します。

   デフォルトでは、PuTTYgen には拡張子が .PPK のファイルだけが表示されます。.PEM ファイルを見つけるには、すべてのファイルの種類を表示するオプションを選択します。



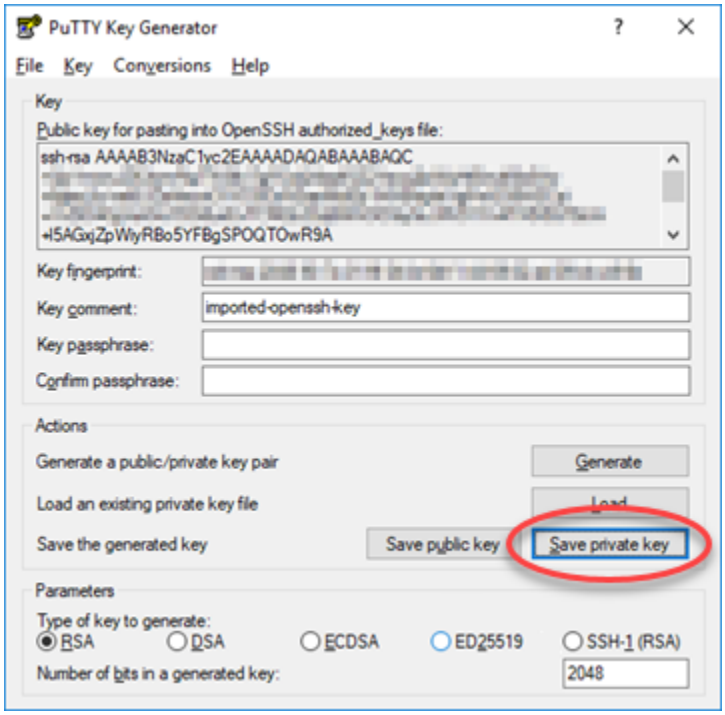
3. このガイドの前半でダウンロードしたデフォルトの Lightsail キーファイル (.PEM) を選択し、を開くを選択します。

4. キーが正常にインポートされたことが PuTTYgen で確認されたら、[OK] を選択します。



5. [Save private key (プライベートキーの保存)] を選択し、パスフレーズ付きで保存しないことを確認します。

   追加のセキュリティ対策としてパスフレーズを作成すると、PuTTY を使用してインスタンスに接続するたびにパスフレーズを入力する必要があります。



6. プライベートキーを保存する名前と場所を指定し、[Save (保存)] を選択します。

   PuTTYgen に、新しいキーファイルが .PPK ファイルタイプとして保存されます。

7. PuTTYgen を閉じます。

   このガイドの「インスタンスに接続するように PuTTY を設定する」セクションに進み、生成した新しい.PPK ファイルを使用して PuTTY を設定し、Amazon EC2 の Linux または Unix インスタンスに接続します。

インスタンスに接続するように PuTTY を設定する

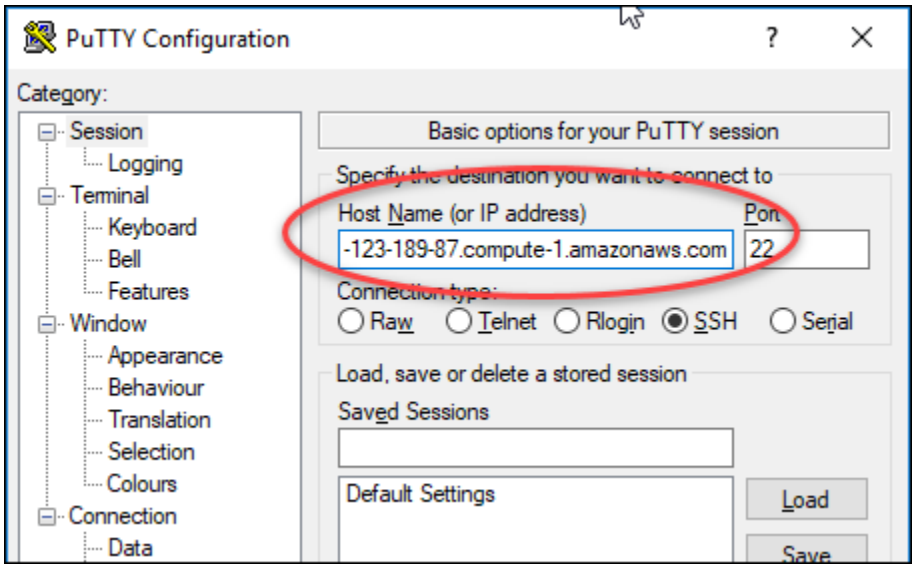
SSH を使用して Linux または Unix インスタンスに接続するためのすべての要件を満たしたので、次に PuTTY を設定します。

Linux または Unix インスタンスに接続するように PuTTY を設定するには

1. PuTTY を開きます。

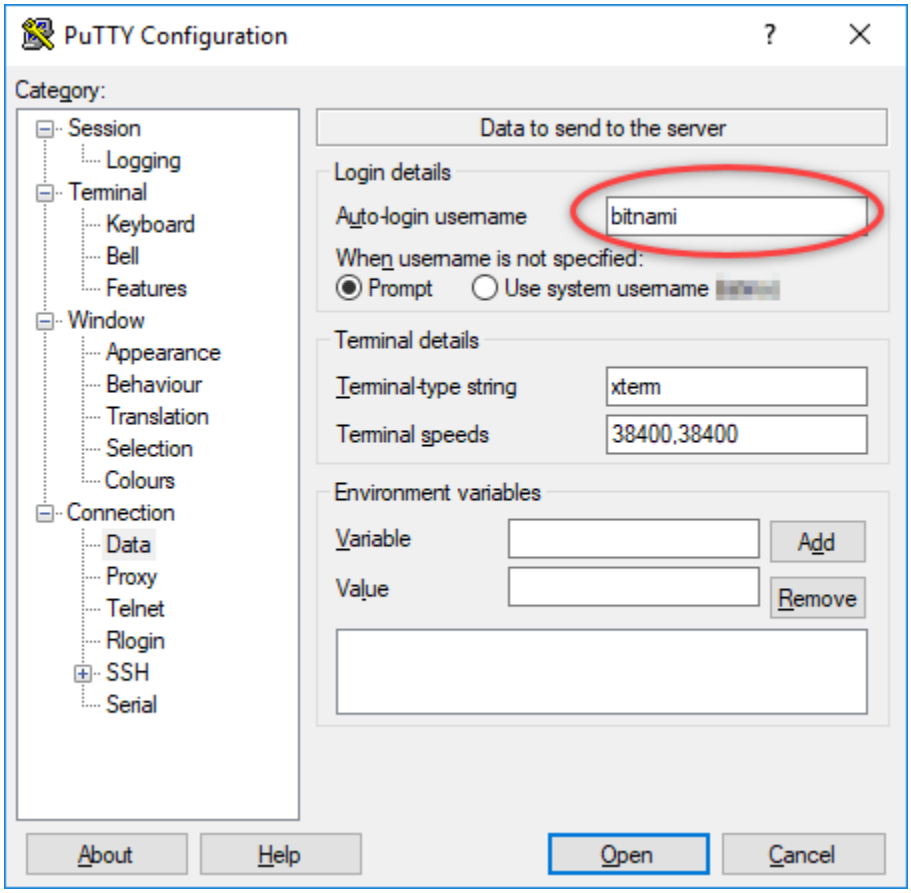
   たとえば、Windows のスタートメニューで、[すべてのプログラム]、[PuTTY]、[PuTTY] の順に選択します。

2. [ホスト名] テキストボックスに、このガイドで前に Amazon EC2 コンソールから取得した、インスタンスのパブリック DNS アドレスを入力します。



3. 左のナビゲーションペインの [Connection (接続)] セクションで、[Data (データ)] を選択します。

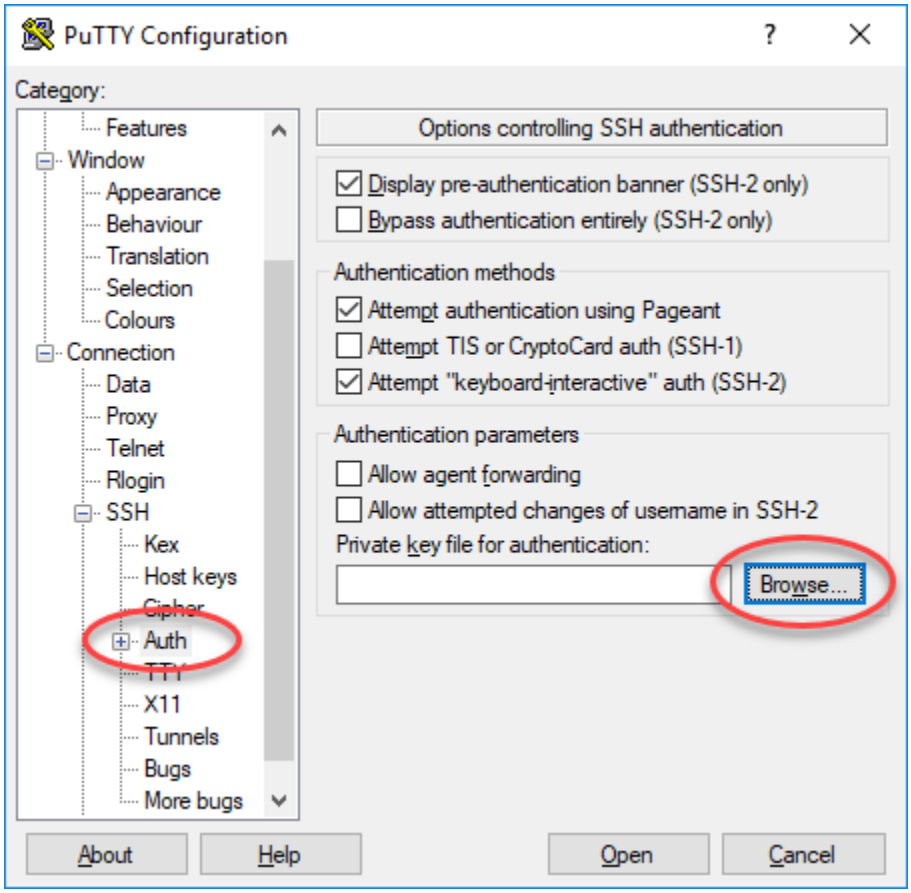
4. [Auto-login username (自動ログインのユーザー名)] テキストボックスに、インスタンスにログインするときに使用するユーザー名を入力します。



ソース Lightsail インスタンスの設計図に応じて、次のいずれかのデフォルトのユーザー名を入力します。

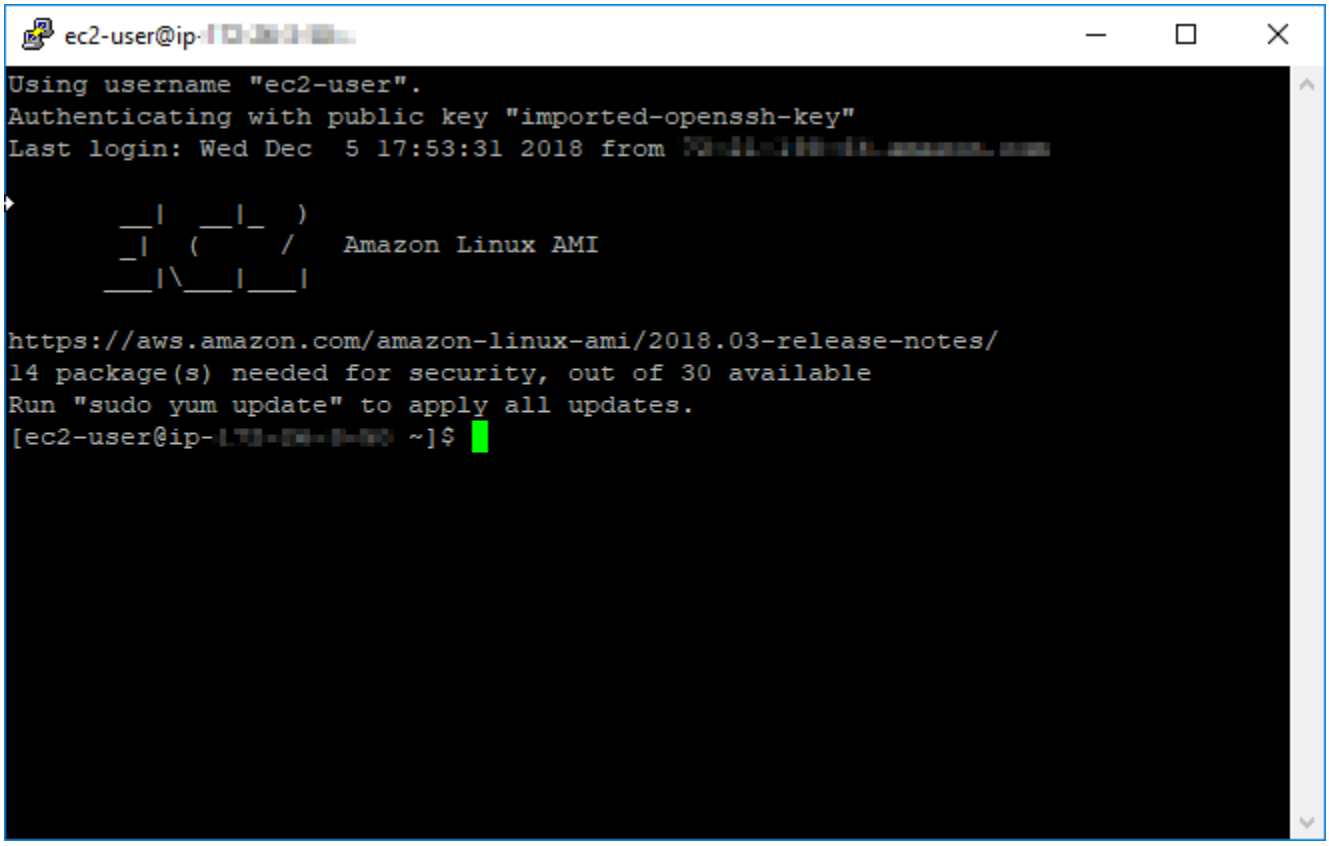
- AlmaLinux、Amazon Linux 2、Amazon Linux 2023、CentOS Stream 9、FreeBSD、openSUSE インスタンス: `ec2-user`
- CentOS 7 インスタンス: `centos`
- Debian インスタンス: `admin`
- Ubuntu インスタンス: `ubuntu`
- Bitnami インスタンス: `bitnami`
- Plesk インスタンス: `ubuntu`
- cPanel と WHM インスタンス : `centos`

5. 左のナビゲーションペインの [Connection (接続)] セクションで、[SSH] を展開して [Auth (認証)] を選択します。
6. [Browse (参照)] を選択し、このガイドの前のセクションで作成した .PPK ファイルに移動して、[Open (開く)] を選択します。



7. [Open (開く)] を選択してインスタンスに接続し、今後はこの接続を信頼するために [Yes (はい)] を選択します。

   インスタンスに正常に接続されると、次のような画面が表示されます。



次のステップ

Amazon EC2 の新しい Linux または Unix インスタンスには、エクスポートしたスナップショットから新しいインスタンスを作成するために Amazon EC2 を使用する場合、Lightsail サービスからの残差キーが含まれています。新しい Amazon EC2 インスタンスのセキュリティを強化するには、これらのキーを削除することをお勧めします。詳細については、「Lightsail スナップショットから作成された Amazon EC2 の Linux または Unix インスタンスを保護する」を参照してください。

Lightsail スナップショットから作成された Amazon EC2 の Windows Server インスタンスに接続する

(Amazon Elastic Compute Cloud (Amazon EC2) で新しい Windows Server インスタンスを作成すると、Remote Desktop Protocol (RDP) を使用して、このインスタンスに接続できます。接続方法はソースの Amazon Lightsail インスタンスに接続したときと同様です。ソースインスタンスの AWS リージョン における Lightsail のデフォルトキーペアを使用して EC2 インスタンスに接続します。このガイドでは、Microsoft リモートデスクトップ接続を使用して Windows Server インスタンスに接続する方法について説明します。

> **Note**
>
> Linux または Unix インスタンスへの接続の詳細については、「Lightsail スナップショットから作成した Amazon EC2 の Linux または Unix インスタンスに接続する」を参照してください。

目次

インスタンスのキーを取得する

Amazon EC2 の Windows Server インスタンスは、ソースインスタンスのリージョンの Lightsail のデフォルトキーペアを使用して、デフォルトの管理者パスワードを取得します。

Lightsail アカウントページの SSH キータブからデフォルトのプライベートキーをダウンロードします。デフォルトの Lightsail SSH キーの詳細については、「SSH のキーペア」を参照してください。

> **Note**
>
> EC2 インスタンスに接続したら、Amazon EC2 で Windows Server インスタンスの管理者パスワードを変更することをお勧めします。これにより、Lightsail のデフォルトキーペアと Amazon EC2 の Windows Server インスタンスの関連付けが解除されます。詳細については、「Lightsail スナップショットから作成した Amazon EC2 の Windows Server インスタンスを保護する」を参照してください。

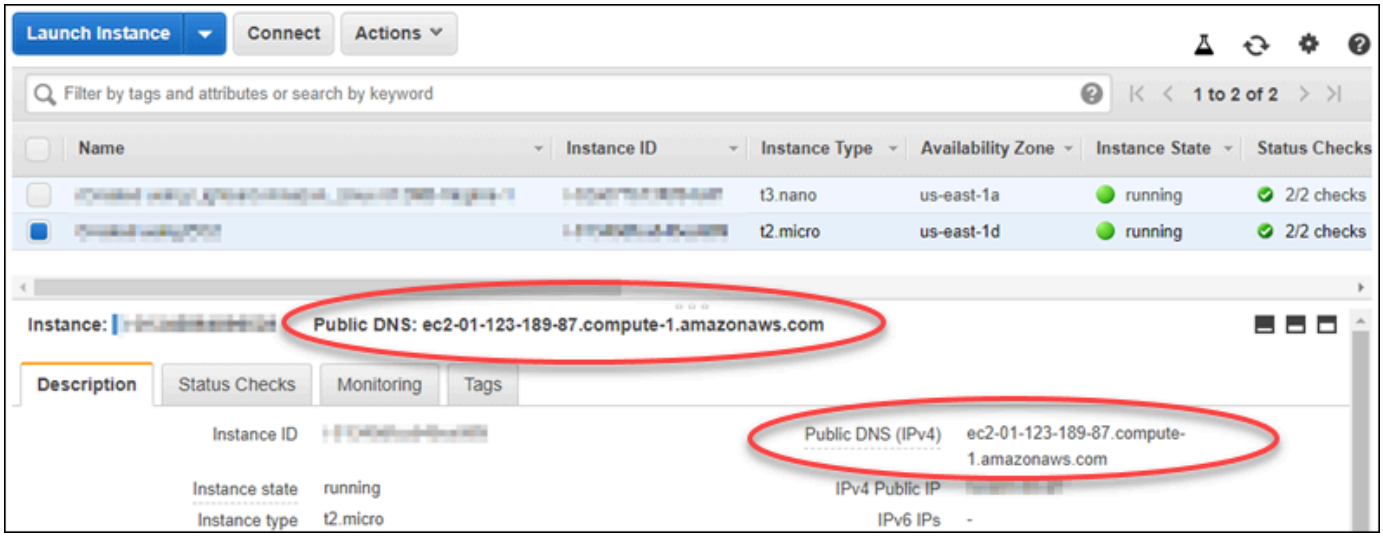
インスタンスのパブリック DNS アドレスを取得する

Amazon EC2 インスタンスのパブリック DNS アドレスを取得し、これを RDP クライアント (Microsoft リモートデスクトップ接続など) の設定時に使用してインスタンスに接続します。

インスタンスのパブリック DNS アドレスを取得するには

1. Amazon EC2 コンソールにサインインします。
2. 左側のナビゲーションペインから、[インスタンス] を選択します。
3. 接続先である実行中の Windows Server インスタンスを選択します。
4. 下部のペインで、インスタンスのパブリック DNS アドレスを見つけます。

   このアドレスを RDP クライアントの設定時に使用してインスタンスに接続します。「Windows Server インスタンスのパスワードを取得する」セクションに進み、Amazon EC2 で Windows Server インスタンスのデフォルトの管理者パスワードを取得する方法を確認します。

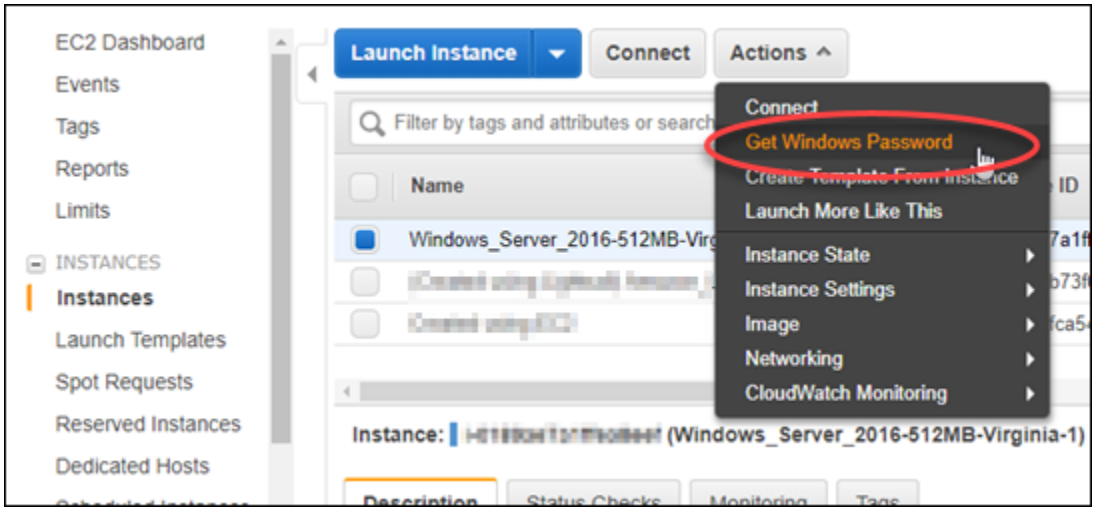


Windows Server インスタンスのパスワードを取得する

Amazon EC2 コンソールから Windows Server インスタンスのパスワードを取得します。このパスワードは、RDP を通じて Windows Server インスタンスに接続するときに、このインスタンスにサインインするために使用します。

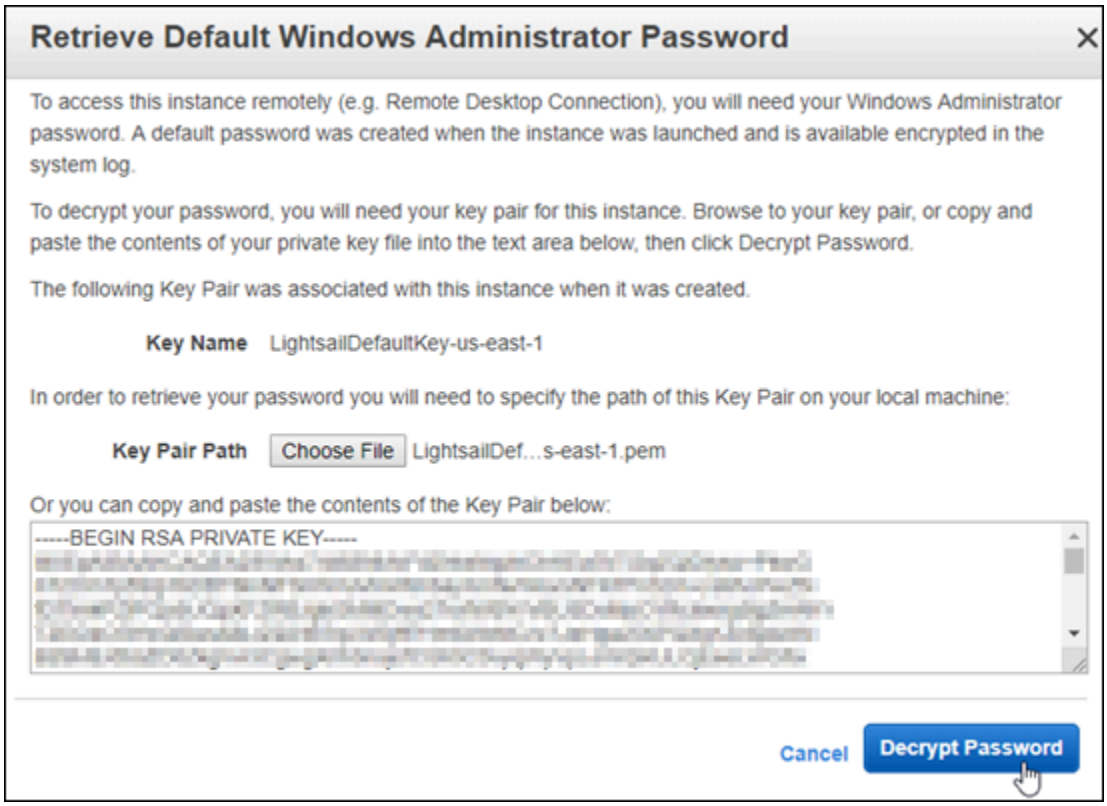
Windows Server インスタンスのパスワードを取得するには

1. Amazon EC2 コンソールにサインインします。
2. 左のナビゲーションペインの [インスタンス] を選択します。
3. 接続先の Windows Server インスタンスを選択します。
4. [アクション]、[Windows パスワードの取得] の順に選択します。

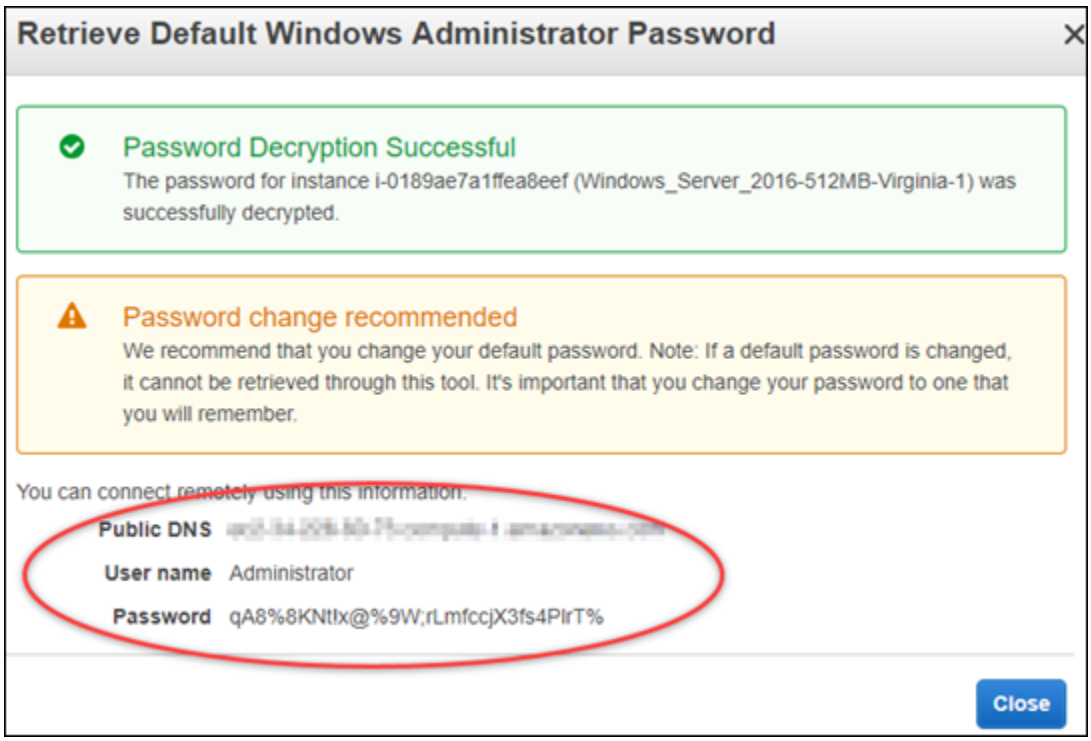


5. プロンプトに応じて [参照] を選択し、このガイドで前に Lightsail からダウンロードしたデフォルトのプライベートキーファイルを開きます。

6. [Decrypt Password] (パスワードを復号化) を選択します。



パブリック DNS およびユーザー名と共に、パスワードが画面に表示されます。パスワードをクリップボードにコピーします。このパスワードは、次の「Windows Server インスタンスに接続するようにリモートデスクトップ接続を設定する」セクションで使用します。パスワードを強調表示し、Ctrl+C (Windows) または Cmd+C (macOS) を押します。



このガイドの「Windows Server インスタンスに接続するようにリモートデスクトップ接続を設定する」セクションに進み、Amazon EC2 で Windows Server インスタンスに接続するようにリモートデスクトップ接続を設定する方法を確認します。

Windows Server インスタンスに接続するようにリモートデスクトップ接続を設定する

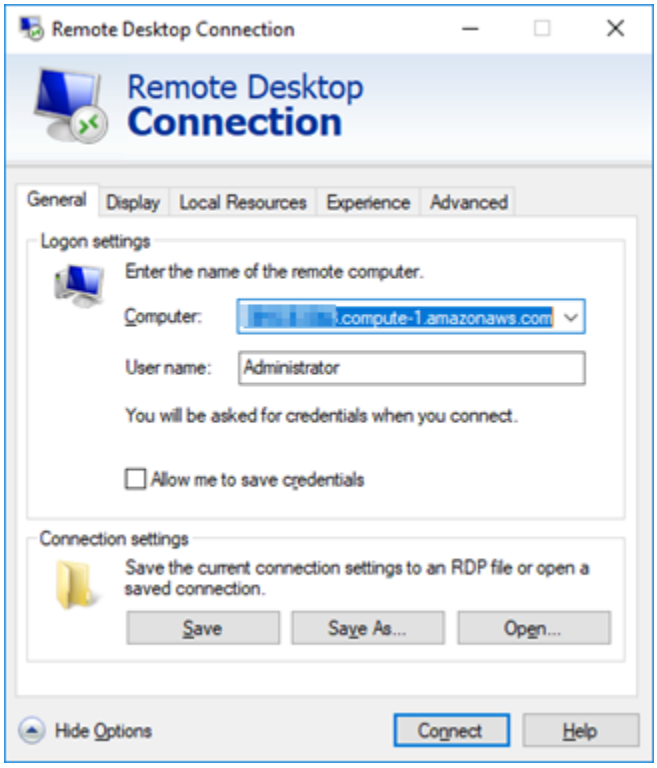
リモートデスクトップ接続は、ほとんどの Windows オペレーティングシステムにプリインストールされている RDP クライアントです。これを使用して、Amazon EC2 の Windows Server インスタンスにグラフィカルに接続します。

Windows Server インスタンスに接続するようにリモートデスクトップ接続を設定するには

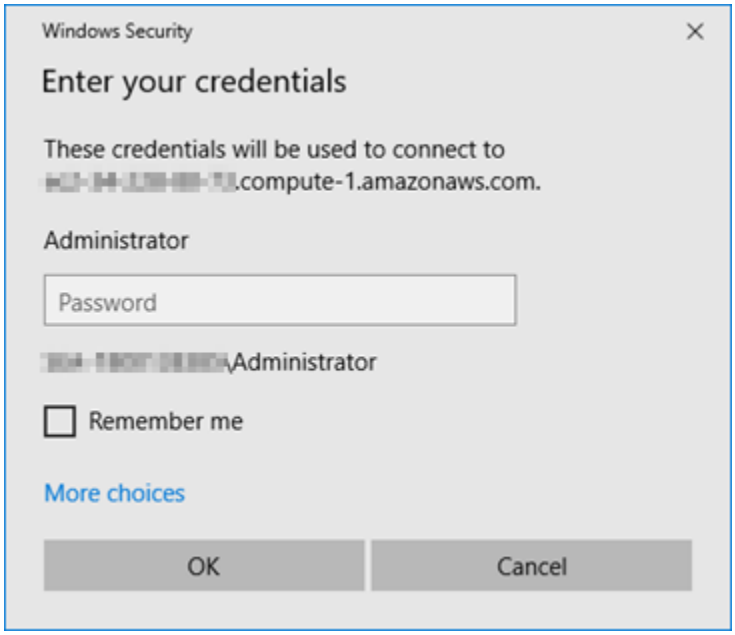
1. リモートデスクトップ接続を開きます。

   たとえば、Windows のスタート メニューを選択し、[リモートデスクトップ接続] を見つけます。

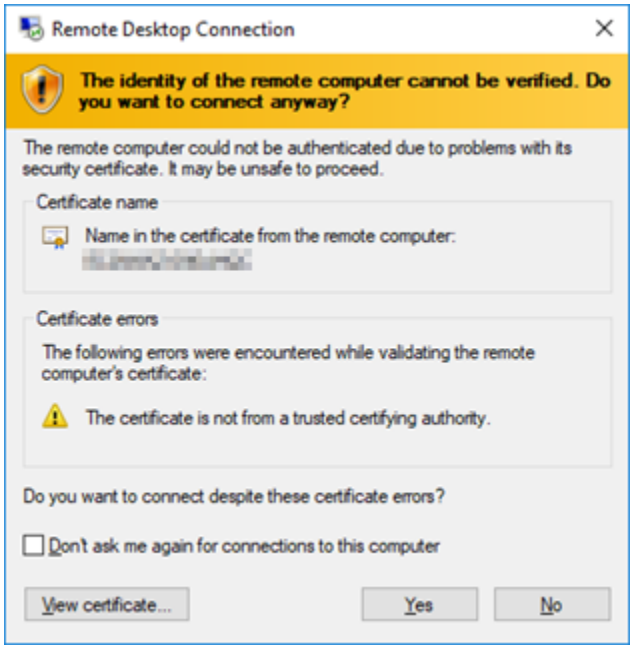
2. [コンピューター] テキストボックスに、このガイドで前に取得した Amazon EC2 の Windows Server インスタンスのパブリック DNS アドレスを入力します。
3. [オプションの表示] を選択して追加のオプションを表示します。
4. `Administrator` をユーザー名テキストボックスに入力します。



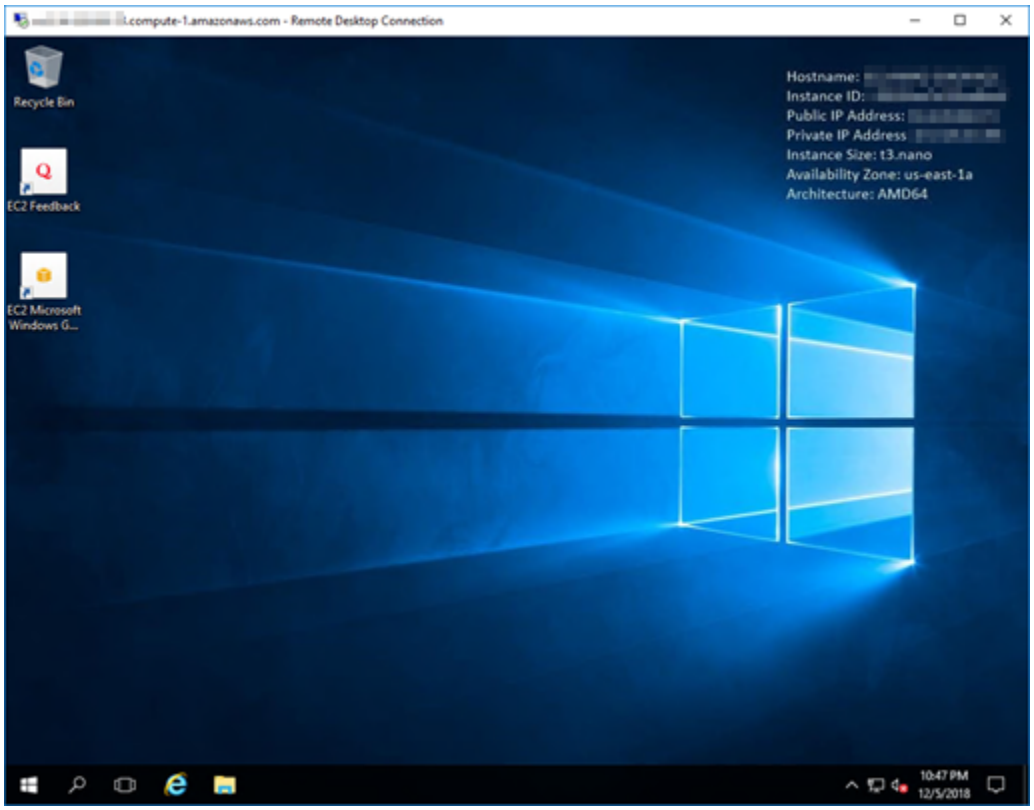
5. [接続] を選択して Windows Server インスタンスに接続します。
6. Windows セキュリティのプロンプトで、[パスワード] テキストボックスに Windows Server インスタンスのパスワードを入力し、[OK] を選択します。



7. リモートデスクトップ接続のプロンプトで、[はい] を選択して接続します。



インスタンスに正常に接続されると、次のような画面が表示されます。



次のステップ

Amazon EC2 で Windows Server インスタンスの管理者パスワードを変更することをお勧めします。これにより、Lightsail のデフォルトキーペアと Amazon EC2 の Windows Server インスタンスの関連付けが解除されます。詳細については、「Lightsail スナップショットから作成した Amazon EC2 の Windows Server インスタンスを保護する」を参照してください。

Lightsail Windows Server インスタンスのスナップショットを作成する

スナップショットは、インスタンスのシステムディスクおよびオリジナル設定のコピーです。スナップショットには、メモリ、CPU、ディスクサイズ、データ転送レートなどの情報が含まれています。詳細については、「スナップショット」を参照してください。

Lightsail で Windows Server インスタンスのスナップショットを作成するには、最初にバックアップスナップショットを作成します。次に、システム準備 (Sysprep) という特別なユーティリティを使用して、2 つ目のスナップショットを作成します。Sysprep は Windows Server のインストールを一般化するため、インスタンスをスナップショットとしてバックアップできます。次に、そのスナップショットからインスタンスを作成すると、Windows インスタンスの最初の実行エクスペリエンスであるように感じます。

Linux または UNIX インスタンスのスナップショットを作成するには、「Linux または Unix インスタンスのスナップショットを作成する」を参照してください。

目次

ステップ 1: Sysprep を実行する前にバックアップスナップショットを作成する

Sysprep を実行してスナップショットを作成すると、システム固有の情報はインスタンスから削除されます。これは、インスタンスで実行されているアプリケーションに意図しない結果をもたらす場合があります。したがって、Sysprep を実行する前にバックアップスナップショットを必ず作成し、何か異常が発生した場合に備えて別のスナップショットを用意します。

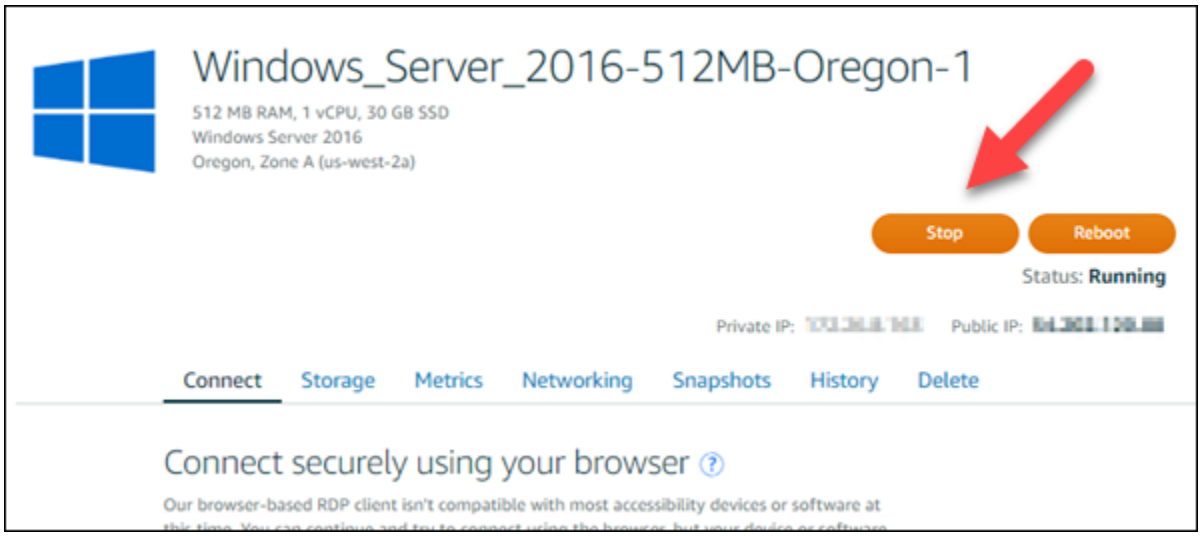
Sysprep を実行する前にスナップショットを作成すると、このバックアップスナップショットを使用して作成するインスタンスでは、元のインスタンスと同じ管理者パスワードが使用されます。これらのインスタンスには、Lightsail コンソールのブラウザベースの RDP クライアントを使用して接続で

きません。ただし、独自の RDP クライアントおよび元のインスタンスと同じ管理者パスワードを使用して接続できます。詳細については、「Windows コンピュータでリモートデスクトップ接続クライアントを使用して Amazon Lightsail の Windows インスタンスに接続する」を参照してください。

> **Important**
>
> 元の Windows インスタンスの管理者パスワードを保存して、安全な場所に保管します。後に問題が発生した場合に管理者パスワードが必要になります。その場合は、Sysprep を実行する前に作成したスナップショットからインスタンスを作成します。

Sysprep を実行する前にバックアップスナップショットを作成するには

1. Lightsail コンソールにサインインします。
2. Lightsail のホームページで、スナップショットを作成する Windows Server インスタンスの名前を選択します。
3. インスタンス管理ページの上部にある [停止] を選択してインスタンスを停止します。



> **Note**
>
> インスタンスを停止すると、再開するまでインスタンスのウェブサイトやサービスは使用できなくなります。

4. [スナップショット] タブを選択します。
5. このページの [手動スナップショット] セクションで、[スナップショットの作成] を選択し、スナップショットの名前を入力します。

リソース名:

- Lightsail アカウントの各 AWS リージョン 内で一意である必要があります。
- 2〜255 文字を使用する必要があります。
- 先頭と末尾は英数字または数字を使用する必要があります。
- 英数字、数字、ピリオド、ダッシュ、アンダースコアを使用することができます。

6. [Create] (作成) を選択します。
7. 確認のために、プロンプトで [スナップショットの作成] をもう一度選択します。

   スナップショットプロセスの完了までには数分かかります。
8. スナップショットの作成後、インスタンス管理ページの上部にある [開始] を選択し、インスタンスを再開します。

ステップ 2: Sysprep を使用してインスタンスに接続し、シャットダウンする

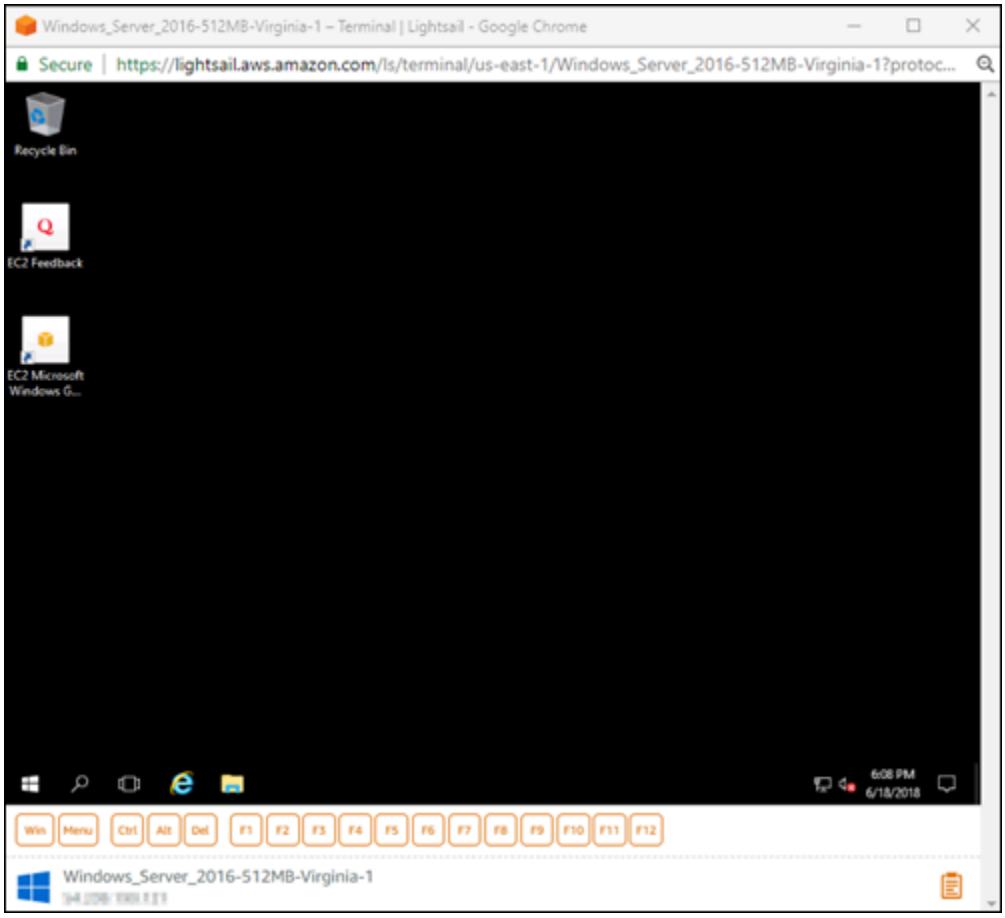
バックアップスナップショットを作成したので、次は Windows Server インスタンスで Sysprep を実行します。これに伴ってインスタンスがシャットダウンされ、スナップショットを作成できるようになります。Sysprep の詳細については、Microsoft のドキュメントで「Sysprep Overview」を参照してください。

このステップでは、プリインストール済みのアプリケーションを通じてインスタンスに接続し、Sysprep を実行します。アプリケーションは Windows Server 2019 および Windows Server 2016 のインスタンスでは「EC2LaunchSettings」で、Windows Server 2012 インスタンスでは「Ec2ConfigService の設定」です。

インスタンスに接続して Sysprep を実行するには

1. インスタンス管理ページで、[接続] タブ、[RDP を使用して接続] の順に選択します。

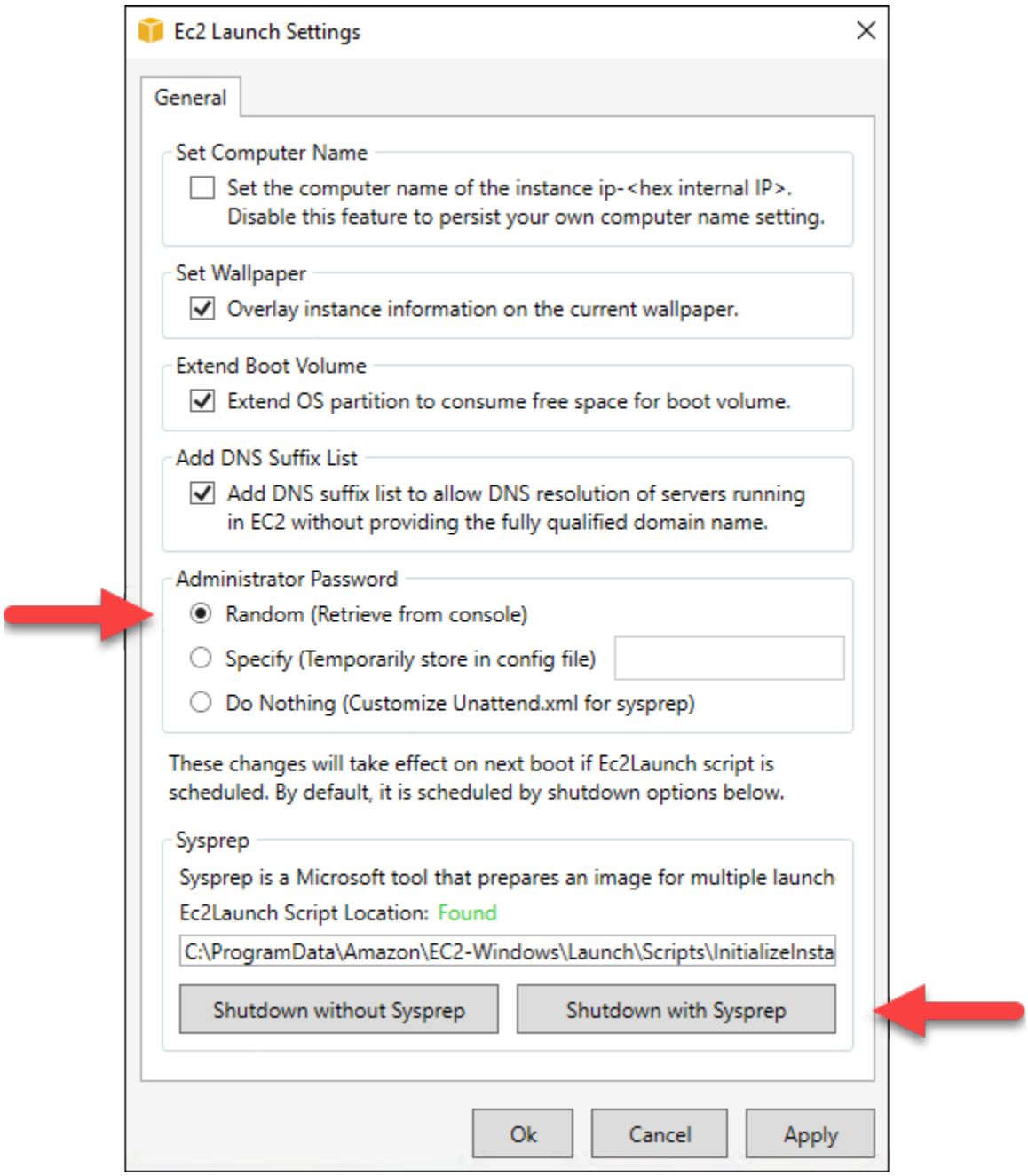
   以下の例に示すように、ブラウザベースの RDP ウィンドウが開きます。



2. タスクバーで、Windows アイコンを選択するか、[Win] を選択してスタートメニューを開きます。
3. 以下のいずれかのオプションを選択します。

   - Windows Server 2019 および Windows Server 2016 インスタンスの場合、[スタート]、[Ec2LaunchSettings] の順に選択します。
   - Windows Server 2012 インスタンスの場合、[開始]、[Ec2ConfigService Settings] の順に選択します。
4. [Administrator Password (管理者パスワード)] セクションで、[Random (Retrieve from console) (ランダム (コンソールから取得))]、[Shutdown with Sysprep (Sysprep でシャットダウン)] の順に選択します。

   Windows Server 2012 インスタンスの Ec2ConfigService Settings アプリケーションの場合、[Random (Retrieve from console) (ランダム (コンソールから取得))] オプションと [Shutdown with Sysprep (Sysprep でシャットダウン)] オプションは [起動] タブの下に表示されます。



5. [Yes (はい)] をクリックし、Sysprep を実行してインスタンスをシャットダウンすることを確認します。

   インスタンスで Sysprep の実行が始まり、RDP 接続がシャットダウンし、Lightsail インスタンスが数分後に実行を停止します。

ステップ 3: Sysprep の実行後にスナップショットを作成する

インスタンスが停止状態になったら、Lightsail コンソールでスナップショットを作成します。Sysprep の実行後に Windows Server インスタンスのスナップショットを作成すると、このスナップショットから作成するすべてのインスタンスで一意の管理者パスワードが使用されます。これらのインスタンスには、Lightsail コンソールのブラウザベースの RDP クライアントを使用して接続できます。

Lightsail コンソールでスナップショットを作成するには

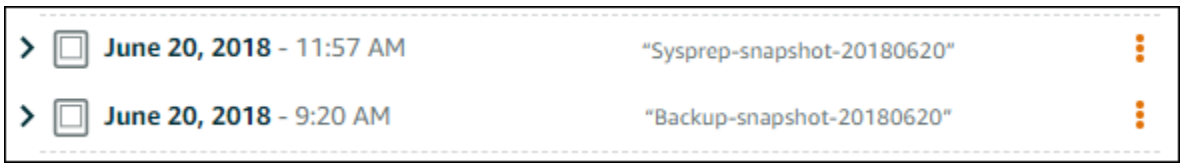
1. Lightsail コンソールに戻ります。
2. Windows Server インスタンスのインスタンス管理ページで、[Snapshots (スナップショット)] タブを選択します
3. このページの [手動スナップショット] セクションで、[スナップショットの作成] を選択し、スナップショットの名前を入力します。

   リソース名:

   - Lightsail アカウントの各 AWS リージョン 内で一意である必要があります。
   - 2〜255 文字を使用する必要があります。
   - 先頭と末尾は英数字または数字を使用する必要があります。
   - 英数字、数字、ピリオド、ダッシュ、アンダースコアを使用することができます。
4. [Create] (作成) を選択します。
5. スナップショットのインスタンスを作成する準備ができたら、プロンプトで [スナップショットの作成] を選択します。

   スナップショットプロセスの完了までには数分かかります。
6. スナップショットの作成後、インスタンス管理ページの上部にある [開始] を選択し、インスタンスを再開します。

   この時点で、次の例に示すように、Windows Server インスタンスの 2 つの作成済みスナップショットが表示されます。



Sysprep スナップショットを使用して、新しいインスタンスを作成します。バックアップスナップショットは、Sysprep の実行後に元のインスタンスが予期どおりに機能しない場合にのみ使用します。

次のステップ

Sysprep およびバックアップスナップショットの作成が完了したので、次のステップを実行できます。

- 元のインスタンスに接続し、インスタンスのアプリケーションが Sysprep の実行後に予期どおりに機能することを確認します。詳細については、「Amazon Lightsail を使用して Windows Server インスタンスに接続する」を参照してください。
- Sysprep を使用して新しいインスタンスを作成し、これに接続して、新しいインスタンスのアプリケーションが予期どおりに機能することを確認します。詳細については、「スナップショットからインスタンスを作成する」を参照してください。
- Sysprep の実行後に元のインスタンスが想定どおりに機能することを確認した後、バックアップスナップショットを削除します。詳細については、「スナップショットを削除する」を参照してください。
- Sysprep の実行後にインスタンスが予期したとおりに機能しない場合は、「スナップショットからインスタンスを作成する」のステップに従って、バックアップスナップショットから新しいインスタンスを作成します。

Lightsail スナップショットから作成された Amazon EC2 の Windows サーバーインスタンスを保護する

Amazon Lightsail スナップショットから作成した Amazon Elastic Compute Cloud (Amazon EC2) の Windows Server インスタンスのセキュリティを向上させるために、デフォルトの管理者パスワードを変更することをお勧めします。これにより、Lightsail のキーペアと Amazon EC2 の Windows Server インスタンスの関連付けが解除されます。

> **Note**
>
> Lightsail のスナップショットから Amazon EC2 の Linux または Unix インスタンスを作成した場合は、これらのインスタンスを保護するためにいくつかのステップを実行する必要があ

ります。詳細については、「Lightsail スナップショットから作成した Amazon EC2 の Linux または Unix インスタンスを保護する」を参照してください。

目次

Amazon EC2 の Windows Server インスタンスに接続する

Windows Server の管理者パスワードを変更するには、リモートデスクトッププロトコル (RDP) を使用して Amazon EC2 の Windows Service インスタンスに接続します。インスタンスに接続する方法については、「Lightsail スナップショットから作成した Amazon EC2 の Windows Server インスタンスに接続する」を参照してください。

Amazon EC2 でインスタンスに接続したら、このガイドの「Amazon EC2 でWindows サーバーインスタンスのデフォルト管理者パスワードを変更する」セクションに進んでください。

Amazon EC2 の Windows Server インスタンスのデフォルト管理者パスワードを変更する

Windows Server インスタンスのデフォルトパスワードを変更し、Lightsail のキーペアと Amazon EC2 の新しい Windows Server インスタンスとの関連付けを解除します。

Amazon EC2 の Windows Server インスタンスのデフォルト管理者パスワードを変更する

1. インスタンスへの RDP 接続を確立したら、コマンドプロンプトを開いて次のコマンドを入力します。

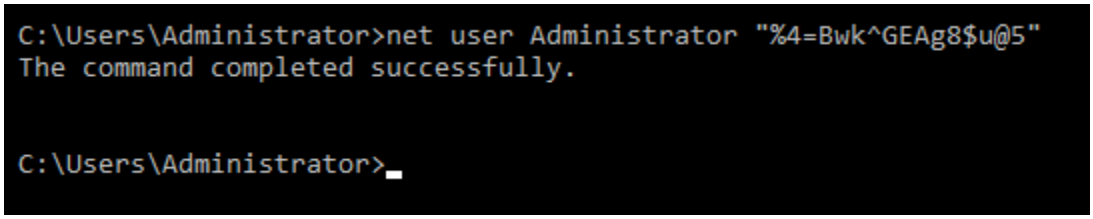
   ```
   net user Administrator "Password"
   ```

   コマンドで、*Password* を新しいパスワードに置き換えます。

   例:

   ```
   net user Administrator "%4=Bwk^GEAg8$u@5"
   ```

   次のような結果が表示されます。



2. 新しいパスワードを安全な場所に保存します。Amazon EC2 コンソールを使用して新しいパスワードを取得することはできません。コンソールで取得できるのはデフォルトのパスワードのみです。パスワードの変更後に、デフォルトのパスワードを使用してインスタンスに接続しようとすると、認証情報が無効であるというエラーが表示されます。

   パスワードを忘れた場合、またはパスワードの有効期限が切れた場合は、新しいパスワードを生成できます。パスワードをリセットする手順については、Amazon EC2 ドキュメントの「紛失または期限切れの Windows 管理者パスワードのリセット」を参照してください。

Lightsail スナップショットから作成した Amazon EC2 の Linux または Unix インスタンスを保護する方法について説明します。

Amazon Lightsail および Amazon Elastic Compute Cloud (Amazon EC2) は公開キー暗号化を使用し、ログイン情報の暗号化と復号を行います。パブリックキー暗号はパブリックキーを使用してデータを暗号化し (パスワードなど)、受信者はプライベートキーを使用してデータを復号します。パブリックキーとプライベートキーは、キーペアと呼ばれます。

Linux または Unix インスタンスを Lightsail から EC2 にエクスポートすると、新しい EC2 インスタンスには Lightsail サービスからのキーが残ります。セキュリティ上のベストプラクティスとして、未使用のキーはインスタンスから削除してください。

Lightsail スナップショットから作成した EC2 の Linux または Unix インスタンスのセキュリティを向上させるには、インスタンスの作成後に以下のアクションを実行することをお勧めします。

- Lightsail のソースインスタンスに接続するために使用した Lightsail のデフォルトキーを削除して置き換えます。独自のキーを使用してインスタンスに接続した場合や、Lightsail コンソールでインスタンスのキーを作成した場合には、Lightsail のデフォルトキーは Amazon EC2 インスタンス内に存在しません。
- Lightsail のシステムキー (`lightsail_instance_ca.pub` キーとも呼ばれます) を削除します。Linux および Unix インスタンスのこのキーにより、Lightsail のブラウザベースの SSH クライアントはインスタンスに接続できます。Lightsail コンソールまたは Lightsail API

の [Amazon EC2 インスタンスを作成する] ページを使用して EC2 インスタンスが作成すると、`lightsail_instance_ca.pub` キーは自動的に削除されます。

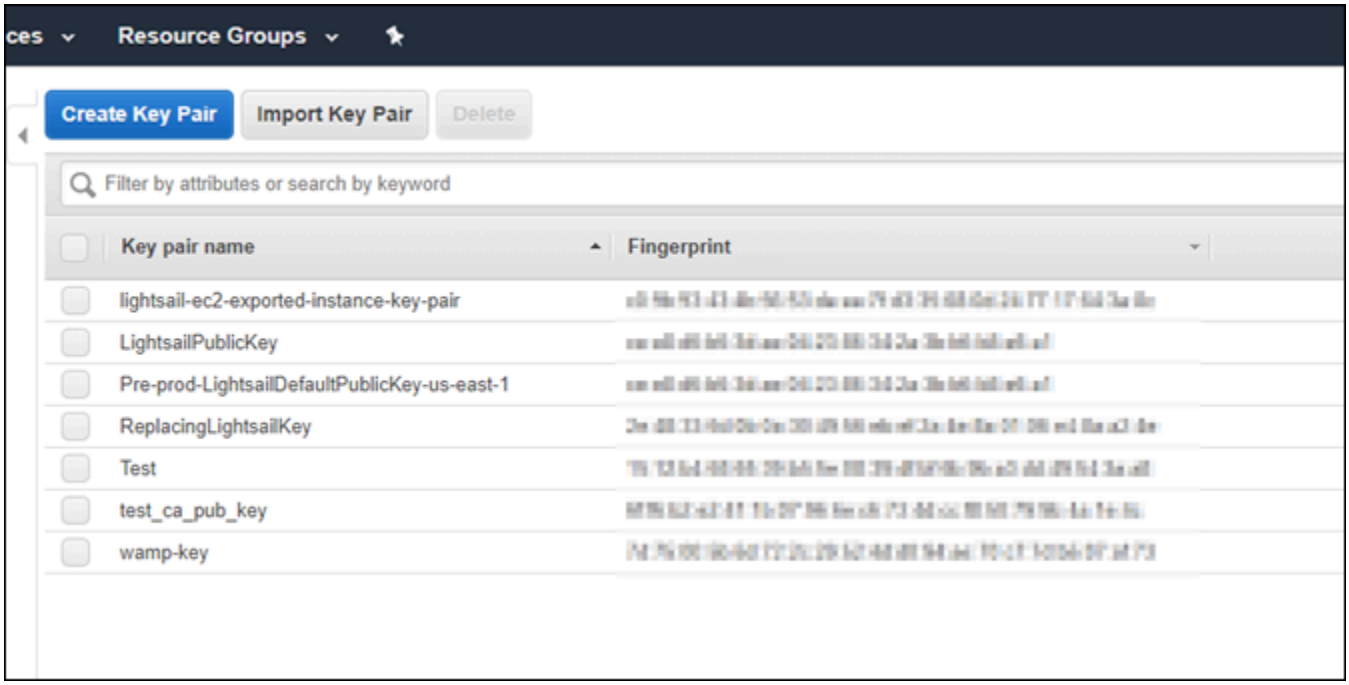
目次

Amazon EC2 を使用してプライベートキーを作成する

Amazon EC2 コンソールを使用して、Lightsail のデフォルトキーを置き換えるために使用する新しいキーペアを作成します。

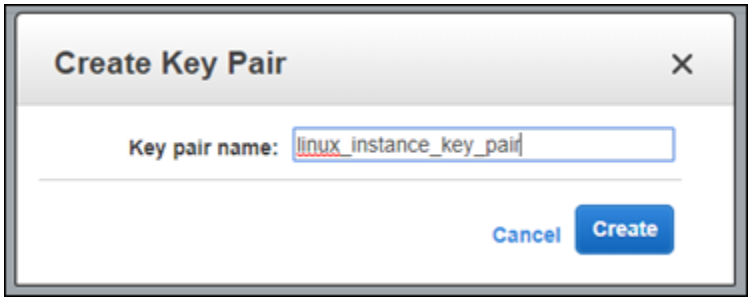
Amazon EC2 を使用してプライベート キーを作成するには

1. Amazon EC2 コンソールにサインインします。
2. 左のナビゲーションペインから、[キーペア] を選択します。
3. [キーペアの作成] を選択します。



4. [キーペア名] テキストボックスにキー名を入力し、[作成] を選択します。

   新しいプライベートキーが自動的にダウンロードされます。プライベートキーの保存先を書き留めておきます。次の「PuTTYgen を使用してパブリックキーを作成する」セクションでパブリックキーを作成するときに必要になります。



PuTTYgen を使用してパブリックキーを作成する

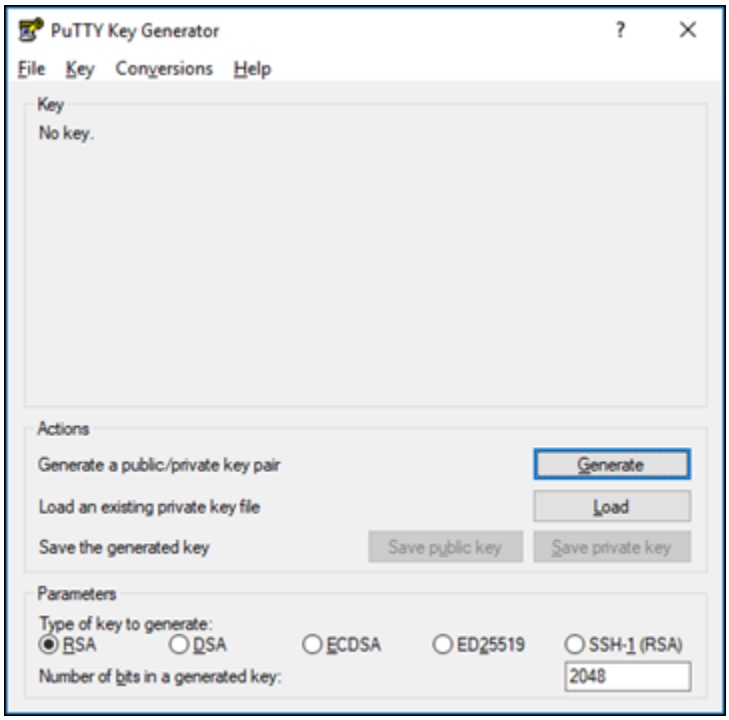
PuTTYgen は PuTTY に含まれているツールです。PuTTYgen では、このガイドで後ほどインスタンスに追加するパブリックキーテキストを生成します。

> **Note**
>
> Linux または Unix インスタンスに接続するように PuTTY を設定する方法の詳細については、「Lightsail スナップショットから作成した Amazon EC2 の Linux または Unix インスタンスに接続する」を参照してください。

PuTTYgen を使用してパブリックキーを作成するには

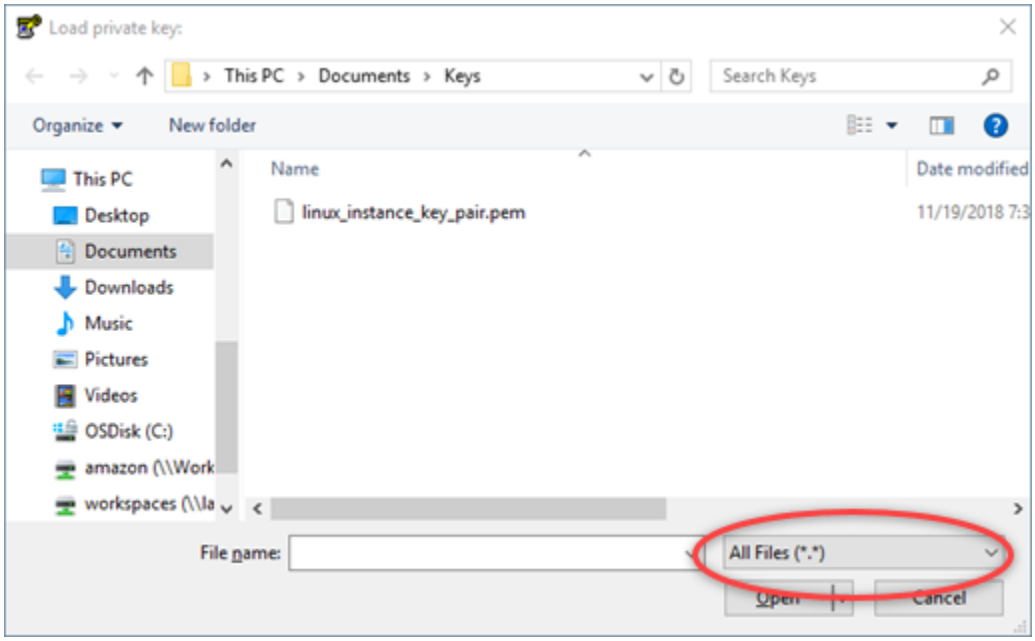
1. PuTTYgen を起動します。

   たとえば、Windows のスタートメニューで、[すべてのプログラム]、[PuTTY]、[PuTTYgen] の順に選択します。



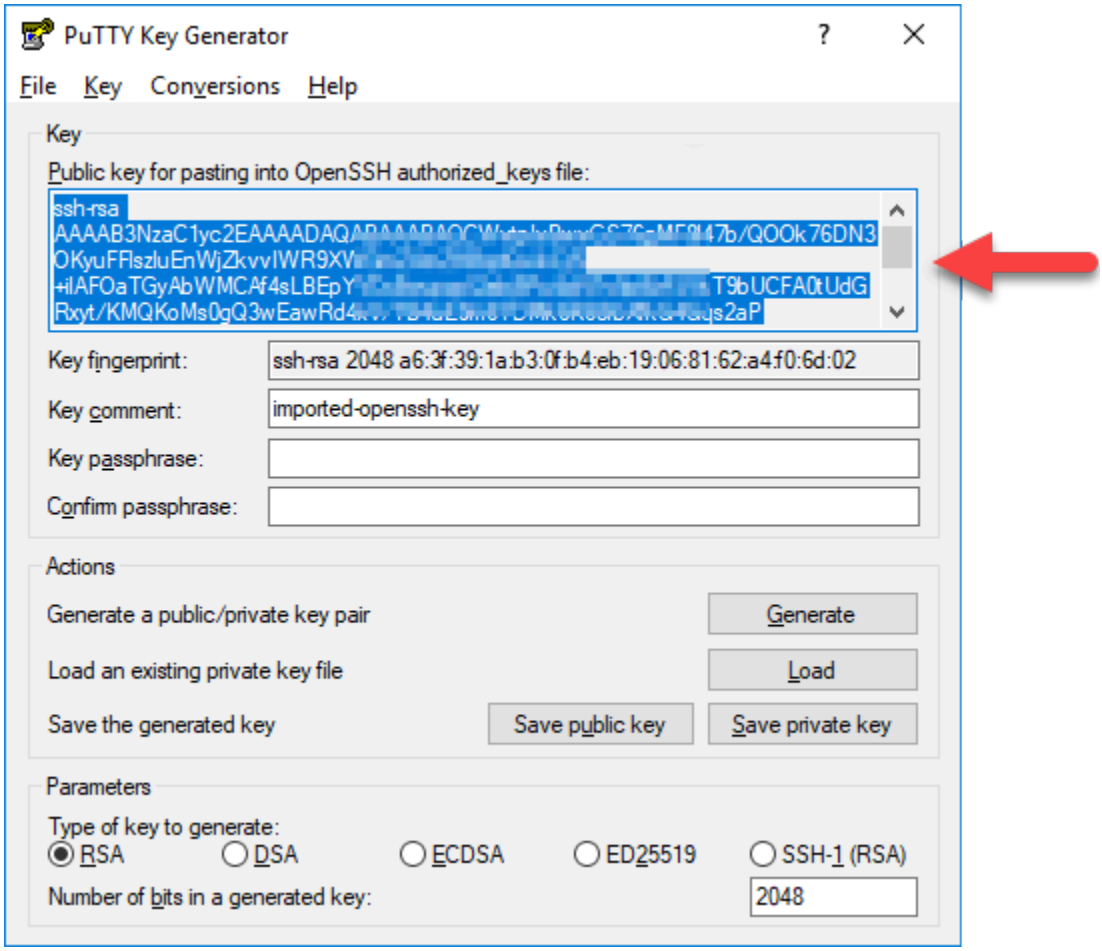
2. [Load] (ロード) を選択します。

   デフォルトでは、PuTTYgen には拡張子が .PPK のファイルだけが表示されます。.PEM ファイルを見つけるには、すべてのファイルの種類を表示するオプションを選択します。



3. このガイドで先ほど作成したプライベートキーの場所に移動します。プライベートキーを選択し、[開く] を選択します。
4. キーが正常にインポートされたことが PuTTYgen で確認されたら、[OK] を選択します。
5. [パブリックキー] テキストボックスの内容を強調表示し、Ctrl+C (Windows) または Cmd+C (macOS) を押してクリップボードにコピーします。

   メモ帳や TextEdit などのテキストエディタを開き、Ctrl+V (Windows) または Cmd+V (macOS) を押してパブリックキーテキストを貼り付けます。パブリックキーテキストのファイルを保存します。このガイドで後ほど必要になります。



6. 「Amazon EC2 の Linux または Unix インスタンスに接続する」セクションに進み、EC2 インスタンスに接続してパブリックキーを追加します。

Amazon EC2 の Linux または Unix インスタンスに接続する

SSH を使用して Amazon EC2 で Linux または Unix インスタンスに接続し、Lightsail のデフォルトキーとシステムキーを削除します。詳細については、「Amazon Lightsail スナップショットから作成した Amazon EC2 の Linux または Unix インスタンスに接続する」を参照してください。

Amazon EC2 でインスタンスに接続したら、このガイドの「公開キーをインスタンスに追加して接続テストをする」のセクションに進んでください。

インスタンスにパブリックキーを追加して接続をテストする

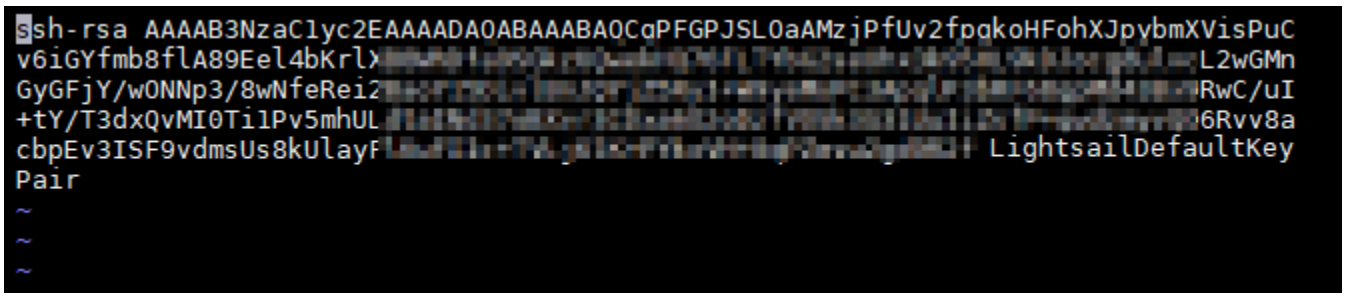
公開キーの内容は、Linux および Unix インスタンスの `~/.ssh/authorized_keys` ファイルに保存されています。このファイルを編集し、Lightsail のデフォルトキーを Amazon EC2 の Linux または Unix インスタンスから削除して置き換えます。

インスタンスにパブリックキーを追加して接続をテストするには

1. インスタンスへの SSH 接続を確立したら、次のコマンドを入力し、Vim テキストエディタを使用して authorized_keys ファイルを編集します。

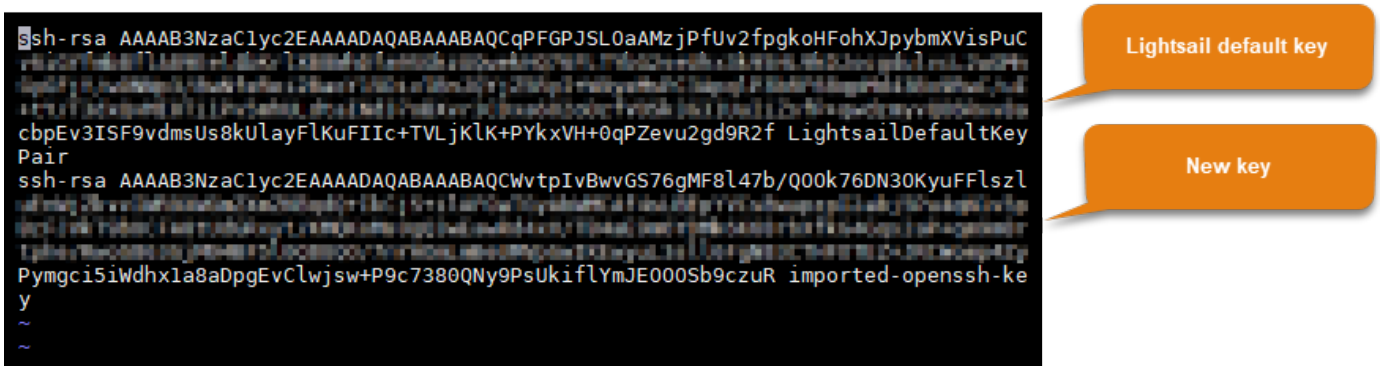
```
sudo vim ~/.ssh/authorized_keys
```

> **Note**
>
> 以下のステップでは、デモの目的で Vim を使用します。ただし、以下のステップでは任意のテキストエディタを使用できます。



2. I キーを押して Vim エディタを挿入モードにします。
3. Lightsail のデフォルトキーの後に追加の行を入力します。
4. このガイドで先ほど保存したパブリックキーテキストをコピーして貼り付けます。

   結果は次のようになります。



5. ESC キーを押して :wq! を入力すると、編集が保存され Vim が終了します。
6. 次のコマンドを入力して Open SSH サーバーを再起動します。

```
sudo /etc/init.d/sshd restart
```

次のような結果が表示されます。

```
[ec2-user@ip-172-26-11-173 ~]$ sudo /etc/init.d/sshd restart
Stopping sshd:                                             [  OK  ]
Starting sshd:                                             [  OK  ]
[ec2-user@ip-172-26-11-173 ~]$
```

新しいパブリックキーがインスタンスに追加されました。新しいキーペアをテストするには、インスタンスから切断します。Lightsail デフォルトキーの代わりに新しいプライベートキーを使用するように PuTTY を設定します。新しいキーペアを使用してインスタンスに正常に接続できる場合は、「Lightsail のデフォルトキーを削除する」セクションに進み、Lightsail のデフォルトキーを削除します。

Lightsail のデフォルトキーを削除する

インスタンスに新しいパブリックキーを追加し、新しいキーペアを使用してインスタンスに正常に接続したら、Lightsail のデフォルトキーを削除します。

Lightsail のデフォルトキーを削除するには

1. インスタンスへの SSH 接続を確立したら、次のコマンドを入力し、Vim テキストエディタを使用して `authorized_keys file` を編集します。

```
sudo vim ~/.ssh/authorized_keys
```

2. I キーを押して Vim エディタを挿入モードにします。
3. `LightsailDefaultKeyPair` で終わる行を削除します。これが Lightsail のデフォルトキーです。

4. ESC キーを押して :wq! を入力すると、編集が保存され Vim が終了します。
5. 次のコマンドを入力して Open SSH サーバーを再起動します。

```
sudo /etc/init.d/sshd restart
```

次のような結果が表示されます。

```
[ec2-user@ip-172-26-11-173 ~]$ sudo /etc/init.d/sshd restart
Stopping sshd:                                             [  OK  ]
Starting sshd:                                             [  OK  ]
[ec2-user@ip-172-26-11-173 ~]$
```

Lightsail のデフォルトキーがインスタンスから削除されました。これで Lightsail のデフォルトキーを使用する接続はインスタンスで拒否されます。「Lightsail のシステムキーを削除する」セクションに進み、Lightsail のシステムキーを削除します。

Lightsail のシステムキーを削除する

Linux および Unix インスタンスにある Lightsail のシステムキー (lightsail_instance_ca.pub キーとも呼ばれます) により、Lightsail のブラウザベースの SSH クライアントはインスタンスに接続できます。以下のステップを実行して、Amazon EC2 の Linux または Unix インスタンスから lightsail_instance_ca.pub キーを削除し、/etc/ssh/sshd_config ファイルを編集します。/etc/ssh/sshd_config ファイルは、インスタンスへの SSH 接続のパラメータを定義します。

Lightsail のシステムキーを削除するには

1. インスタンスに接続されている SSH のターミナルウィンドウで、次のコマンドを入力して lightsail_instance_ca.pub キーを削除します。

```
sudo rm -r /etc/ssh/lightsail_instance_ca.pub
```

2. 次のコマンドを入力し、Vim テキストエディタを使用して sshd_config ファイルを編集します。

```
sudo vim /etc/ssh/sshd_config
```

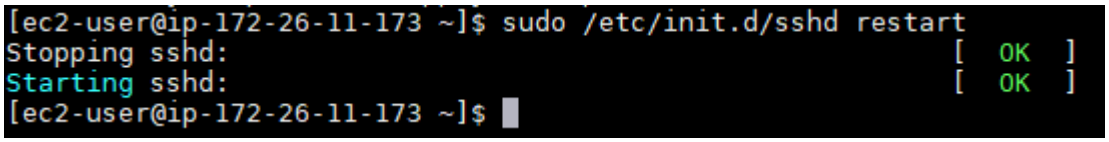
3. I キーを押して Vim エディタを挿入モードにします。
4. 次のテキストをファイルから削除します (ある場合)。

```
TrustedUserCAKeys /etc/ssh/lightsail_instance_ca.pub
```

5. ESC キーを押して :wq! を入力すると、編集が保存され Vim が終了します。
6. 次のコマンドを入力して Open SSH サーバーを再起動します。

```
sudo /etc/init.d/sshd restart
```

次のような結果が表示されます。



lightsail_instance_ca.pub キーがインスタンスから削除されました。関連する sshd_config ファイルが更新されて、このキーが除外されます。

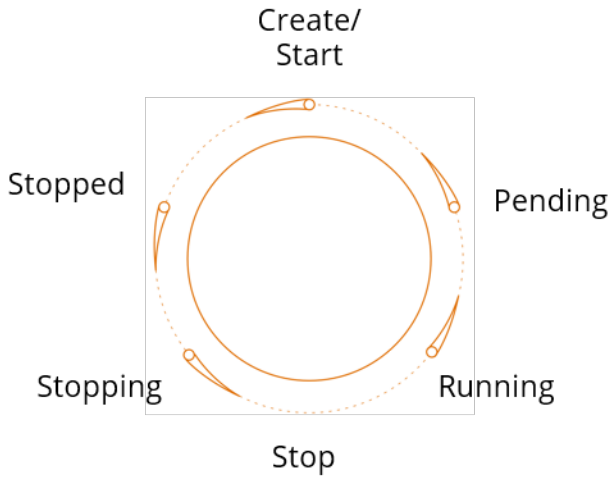
Lightsail インスタンスを管理する

Lightsail では、仮想プライベートサーバーはインスタンスと呼ばれています。インスタンスへの接続、ポートとファイアウォールの設定の管理、メトリクスの表示、静的 IP のインスタンスとの関連付けなどを行うことができます。タスクを選択して、インスタンスを最大限に活用する方法について学習します。

- Linux または Unix インスタンスに接続する
- メトリクスを表示する
- 静的 IP アドレスを作成してインスタンスにアタッチする
- ファイアウォールとポート
- Linux または Unix インスタンスのスナップショットの作成
- インスタンスを開始、停止、または再起動する
- インスタンスを強制停止する

Lightsail インスタンスの開始、停止、または再起動

Lightsail でインスタンスを作成すると、ユーザーのマシンは [保留中] 状態になった後で [実行中] 状態に移行します。インスタンスが実行中になると、そのインスタンスを再起動するか、停止して再起動できます。そのサイクルは次のようになっています。

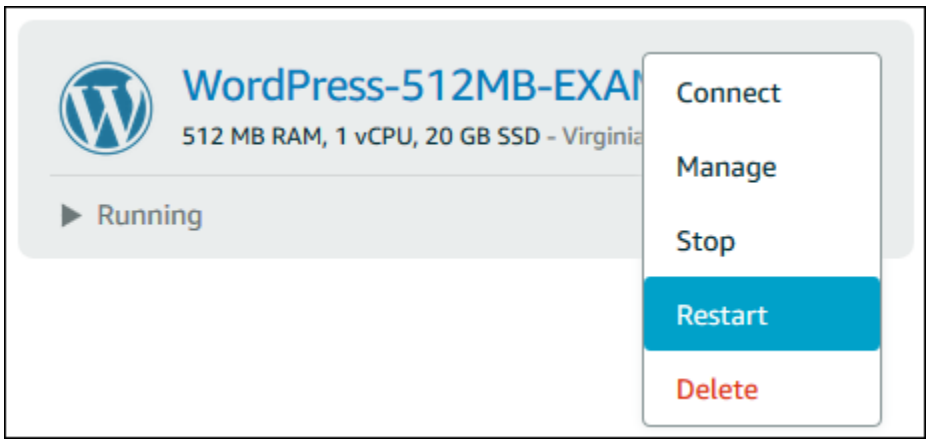


インスタンスの状態は、インスタンスを管理する場合やホームページでインスタンスを表示する場合に確認できます。

> **Important**
>
> インスタンスの作成時にインスタンスに割り当てられるデフォルトの公開 IPv4 アドレスは、インスタンスを停止してまた開始すると変更されます。必要に応じて、静的 IPv4 アドレスをインスタンスに作成し、アタッチできます。静的 IPv4 アドレスは、インスタンスのデフォルトの公開 IPv4 アドレスに置き換えられ、インスタンスを停止して起動しても変わりません。詳細については、「静的 IP を作成してインスタンスにアタッチする」を参照してください。

実行中のインスタンスの再起動

- ホームページで再起動するインスタンスを選択するか、インスタンス管理メニューで [再起動] を選択します。

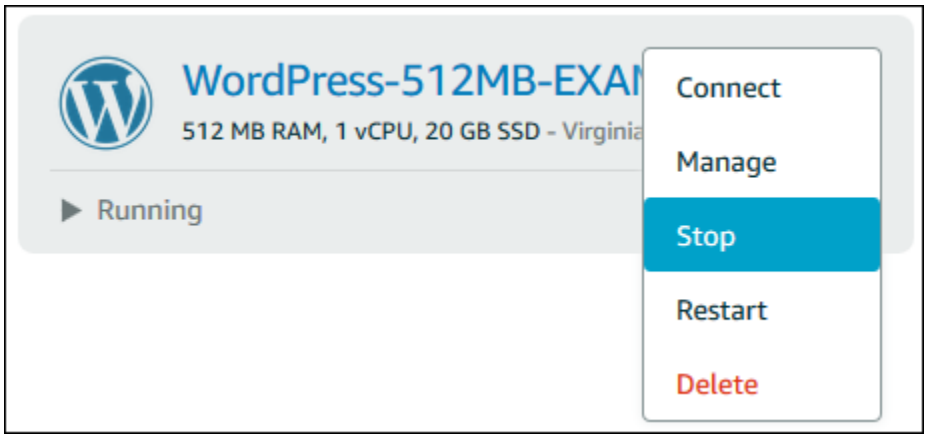


インスタンス管理ページでインスタンスを表示している場合は、[再起動] を選択し、プロンプトが表示されたら [確認] を選択します。

> **Note**
>
> インスタンスを再起動するには、そのインスタンスが [実行中] 状態である必要があります。

実行中のインスタンスの停止

- ホームページで、停止するインスタンスを選択し、インスタンス管理メニューで [停止] を選択します。

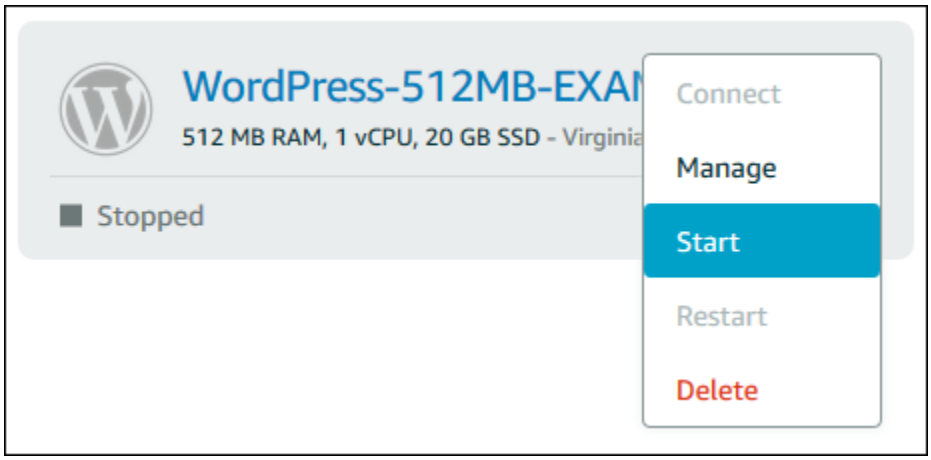


インスタンス管理ページでインスタンスを表示している場合は、[停止] を選択し、プロンプトが表示されたら [確認] を選択します。

> **Note**
>
> インスタンスを停止するには、そのインスタンスが [実行中] 状態である必要があります。

停止した後のインスタンスの開始

- ホームページで、開始するインスタンスを選択し、インスタンス管理メニューで [開始] を選択します。



インスタンス管理ページでインスタンスを表示している場合は、[開始] を選択します。

> **Note**
>
> インスタンスを開始するには、そのインスタンスが [停止] 状態である必要があります。

拡張ネットワーキングについて Amazon EC2 インスタンスを更新する

一部の Lightsail インスタンスは、拡張ネットワーキングに対応していないため、現行世代の EC2 インスタンスタイプ (T3、M5、C5、または R5) と互換性がありません。ソースの Lightsail インスタンスに互換性がない場合は、エクスポートしたスナップショットから EC2 インスタンスを作成するときに、以前の世代のインスタンスタイプ (T2、M4、C4、または R4) から選択する必要があります。これらのインスタンスタイプオプションは、Lightsail コンソールの[Amazon EC2 インスタンス作成] ページを使用して EC2 インスタンスを作成するときに表示されます。

> **Note**
>
> 拡張ネットワークの詳細については、「Amazon EC2 ドキュメント」の「Linux での拡張ネットワーク」または「Windows での拡張ネットワーク」を参照してください。

ソースの Lightsail インスタンスに互換性がない場合に最新世代の EC2 インスタンスタイプを使用するには、まず以前の世代のインスタンスタイプ (T2、M4、C4、または R4) を使用して新しい EC2 インスタンスを作成し、インスタンスのネットワーキングドライバーを更新します。次に、インスタンスを目的の最新世代のインスタンスタイプに更新します。

前提条件

エクスポートした Lightsail スナップショットから Amazon EC2 インスタンスを作成する必要があります。Lightsail インスタンスに互換性がない場合は、Amazon EC2 インスタンスの作成時に以前の世代のインスタンスタイプ (T2、M4、C4、または R4) を選択します。詳細については、「Lightsail でエクスポートしたスナップショットから Amazon EC2 インスタンスを作成する」を参照してください。

新しい EC2 インスタンスが起動して実行中になったら、このガイドの「Elastic Network Adapter で拡張ネットワーキングを有効にする」セクションに進み、拡張ネットワーキングを有効にする方法を確認します。

Elastic Network Adapter で拡張ネットワーキングを有効にする

新しいインスタンスが起動して実行中になったら、以下のいずれかの「Amazon EC2 ドキュメント」のガイドを参照して Elastic Network Adapter (ENA) で拡張ネットワーキングを有効にします。

- Linux インスタンスにおける Elastic Network Adapter (ENA) を使用した拡張ネットワーキングの有効化
- Windows インスタンスにおける Elastic Network Adapter (ENA) を使用した拡張ネットワーキングの有効化

インスタンスタイプをアップグレードする

拡張ネットワーキングを有効にしたら、以下のいずれかのガイドの手順に従ってインスタンスタイプをアップグレードできます。

- Windows Server インスタンスの場合 — 最新世代のインスタンスタイプへの移行

- Linux または Unix インスタンスの場合 — インスタンスタイプを変更する

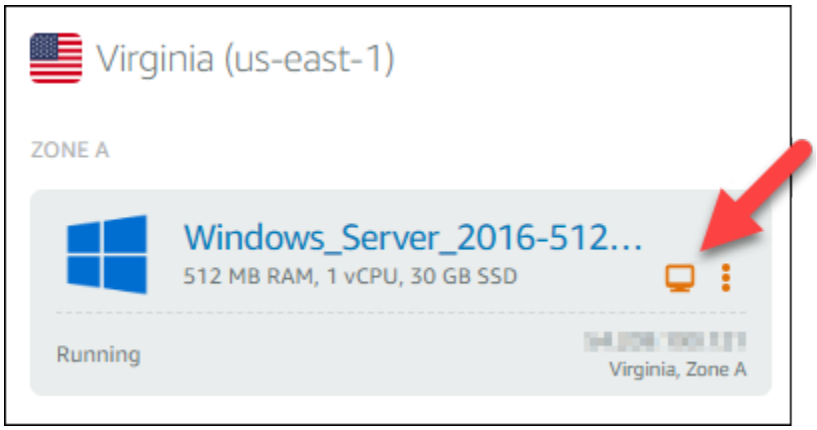
Lightsail Windows Server インスタンスのストレージ領域を拡張する

スナップショットを使用してより大きなプランの新しい Windows Server インスタンスを作成すると、使用可能なストレージ領域がプランに指定されている領域より小さいことに気づく場合があります。通常、より大きなプランに指定されている追加分のストレージ領域は未割り当てのため、アクティブボリュームで使用されないことが原因です。このトピックの手順では、Windows Server インスタンスのファイルシステムを拡張し、使用可能なストレージ領域を最大限に利用する方法を示します。

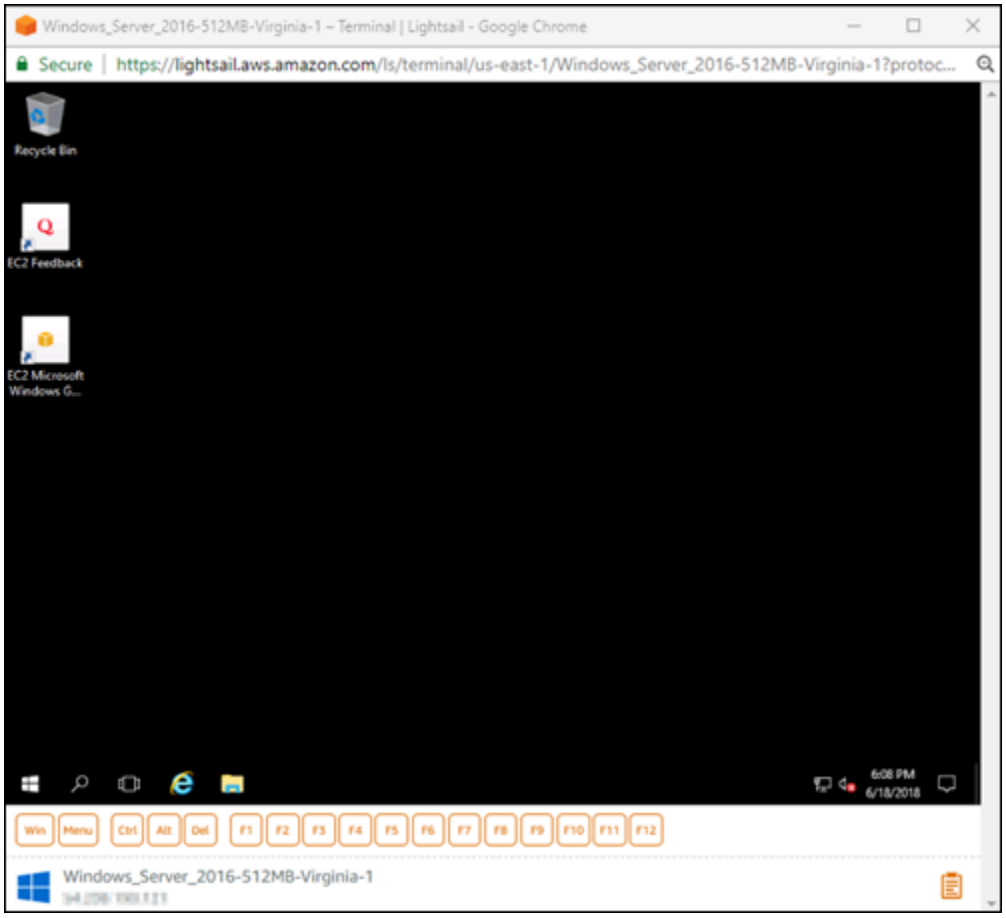
> **Note**
>
> この状況が発生するのは、システム準備 (Sysprep) ユーティリティの実行前に作成したスナップショットを使用して Windows Server インスタンスを作成した場合に限ります。詳細については、「Windows Server インスタンスのスナップショットを作成する」を参照してください。

Windows Server インスタンスのファイルシステムを拡張するには

1. Lightsail コンソールにサインインします。
2. Lightsail のホームページで、接続先のインスタンスの RDP クライアントアイコンを選択します。



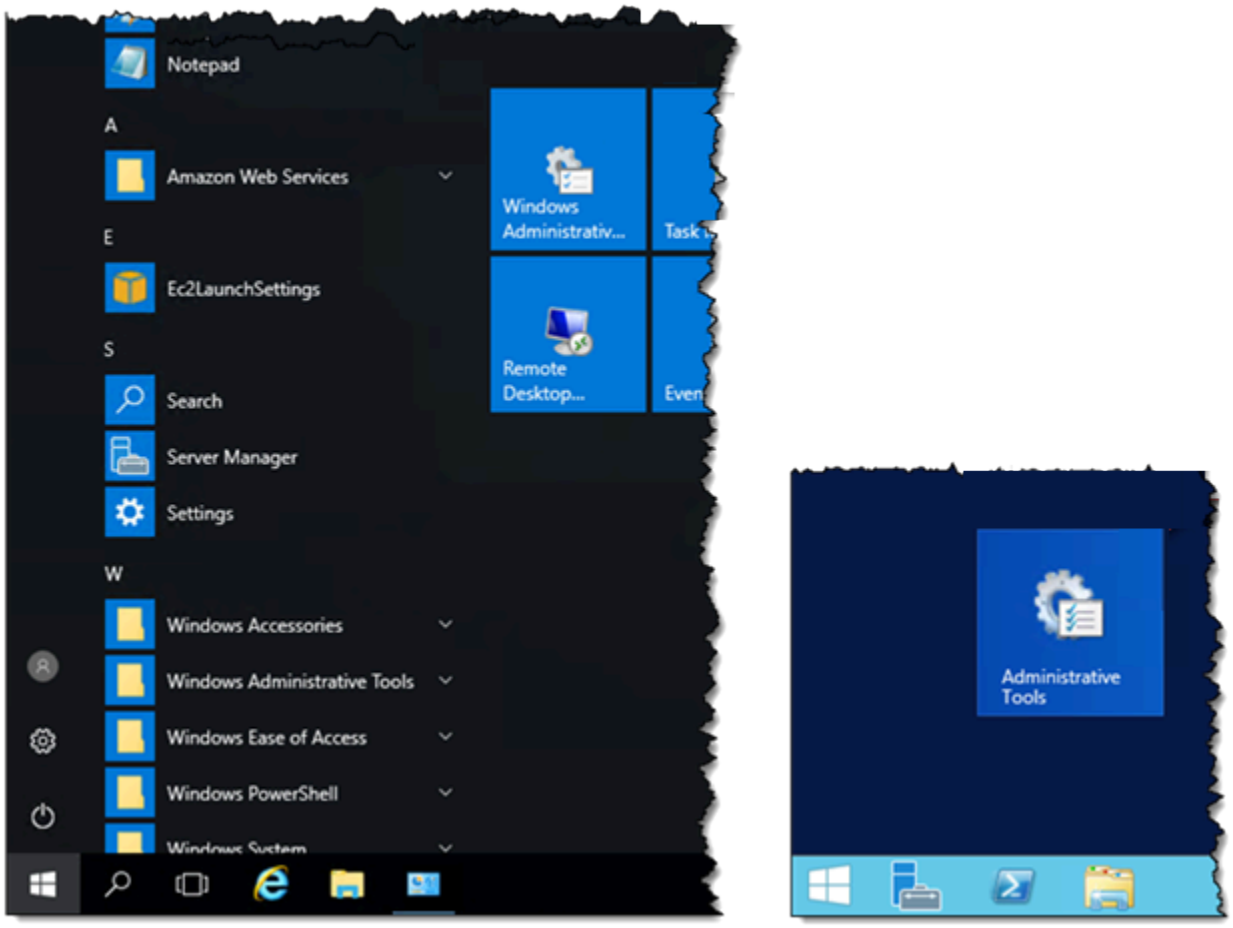
次の例に示すように、ブラウザベースの RDP クライアントウィンドウが開きます。



3. タスクバーで Windows アイコンを選択し、以下のいずれかのオプションを選択します。

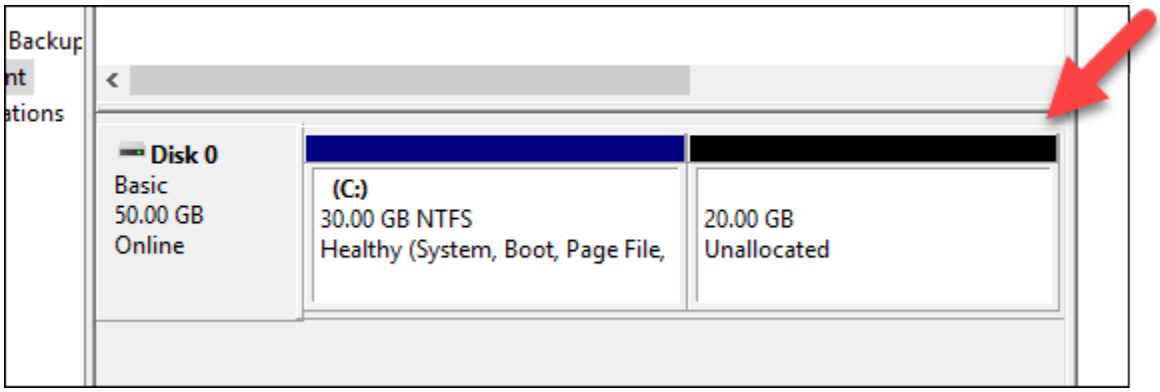
   a. Windows Server 2019 または Windows Server 2016 インスタンスの場合は、[スタート]、[Windows Administrative Tools] (Windows 管理ツール) の順に選択します。

   b. Windows Server 2012 インスタンスの場合は、[スタート]、[管理ツール] の順に選択します。

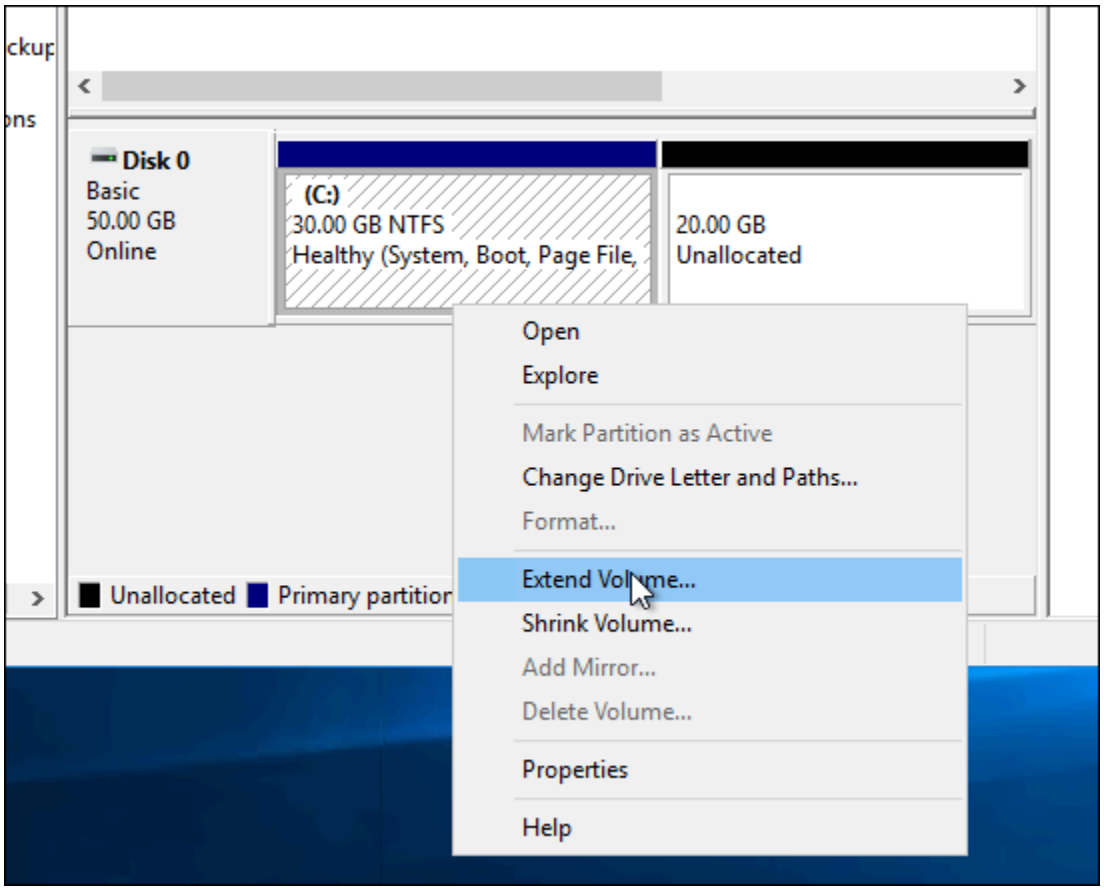


4. [コンピューターの管理] を選択します。
5. [コンピューターの管理] コンソールの左側のペインで、[ディスクの管理] を選択します。
6. [操作] メニューの [ディスクの再スキャン] を選択します。

   ディスクに関連付けられている未割り当て領域が表示される場合があります。未割り当て領域を利用するようにディスクのアクティブボリュームを拡張します。

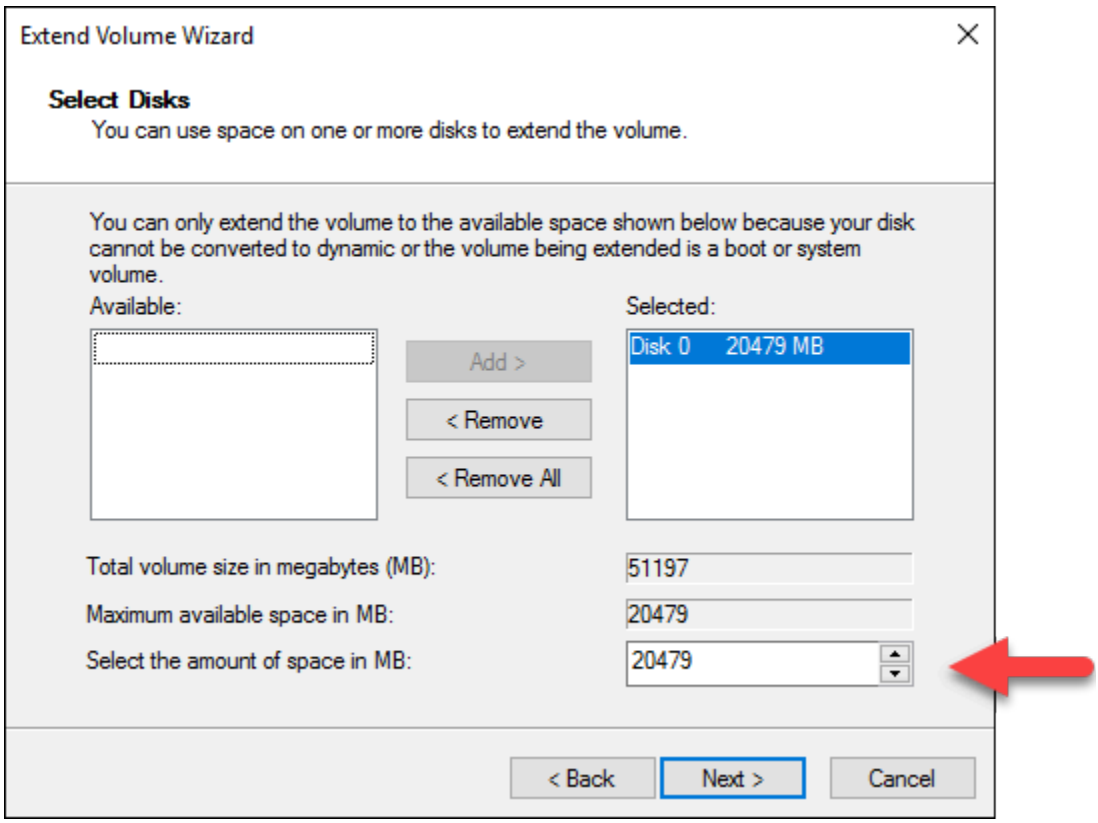


7. 未割り当て領域と同じディスクのアクティブボリュームを右クリックし、[ボリュームの拡張] を選択します。



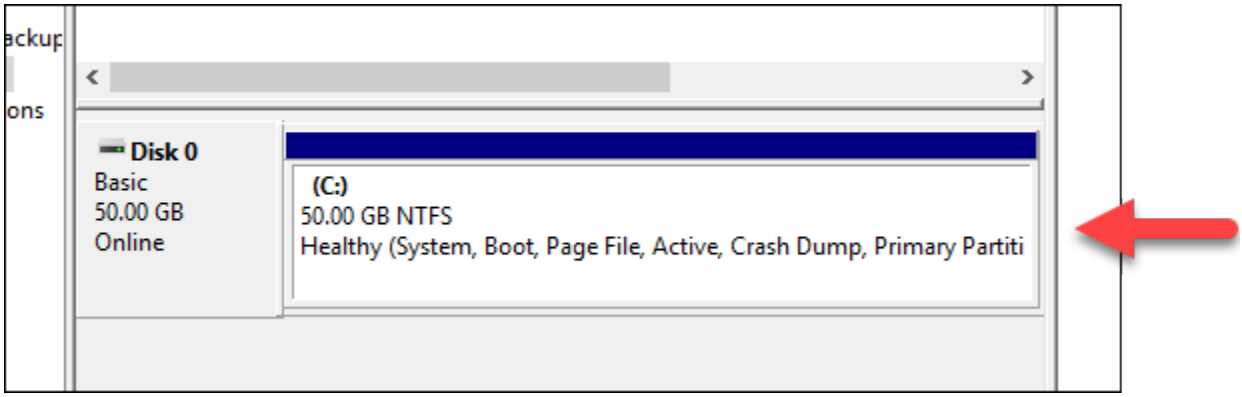
8. ボリュームの拡張ウィザードが開いたら、[次へ] を選択します。

9. [ディスク領域 (MB) を選択] フィールドで、ボリュームを拡張するメガバイト数を入力します。通常、これは最大の未割り当て領域に設定します。入力する値は、ボリュームの最終的なサイズではなく、追加する領域のサイズです。



10. ボリュームの拡張ウィザードを完了します。

    指定した未割り当て領域を利用するようにアクティブボリュームが拡張されます。次の例では、すべての未割り当て領域が選択されています。



Lightsail インスタンスの起動時に起動スクリプトを使用してインスタンスを設定する

Linux/Unix ベースのインスタンスの作成時に、ソフトウェアの追加、ソフトウェアの更新、インスタンスの設定などを他の方法で行う起動スクリプトを追加できます。追加データで Windows ベースの

インスタンスを設定するには、「Windows PowerShell を使用して新しい Lightsail インスタンスを設定する」を参照してください。

> **Note**
>
> インスタンスでソフトウェアを取得するためのコマンドは、選択したマシンイメージに応じて異なります。Amazon Linux では yum が使用され、Debian および Ubuntu の両方では apt-get が使用されます。WordPress や他のアプリケーションイメージでは、オペレーティングシステムとして Ubuntu が使用されているため、apt-get が使用されます。FreeBSD および openSUSE では、freebsd-update や zypper (openSUSE) などのカスタムツールを使用するための追加のユーザー設定が必要です。

例: Node.js をインストールするように Ubuntu サーバーを設定する

次の例では、apt-get コマンドを使用して、パッケージリストを更新し、Node.js をインストールしています。

1. [インスタンスを作成する] ページの [OS のみ] タブで [Ubuntu] を選択します。
2. 下にスクロールして [起動スクリプトの追加] を選択します。
3. 次の内容を入力します。

```
# update package list
apt-get -y update
# install some of my favorite tools
apt-get install -y nodejs
```

> **Note**
>
> サーバーを設定するために送信するコマンドは root として実行されるため、コマンドの前に sudo を付ける必要はありません。

4. [インスタンスの作成] を選択します。

例: プラグインをダウンロードしてインストールするように WordPress サーバーを設定する

次の例では、パッケージリストを更新し、WordPress 用の BuddyPress プラグインをダウンロードしてインストールしています。

1. [インスタンスを作成する] ページで [WordPress] を選択します。
2. [起動スクリプトの追加] を選択します。
3. 次の内容を入力します。

```
# update package list
apt-get -y update
# download wordpress plugin
wget "https://downloads.wordpress.org/plugin/buddypress.2.7.0.zip"
apt-get -y install unzip
# unzip into wordpress plugin directory
unzip buddypress.2.7.0.zip -d /var/wordpress/plugins
```

4. [インスタンスの作成] を選択します。

Windows PowerShell またはバッチスクリプトを使用して Lightsail インスタンスを設定する

Windows ベースのインスタンスを作成するとき、Windows PowerShell スクリプトまたは他のバッチスクリプトを使用して設定することができます。これは、インスタンスの起動直後に実行されるワンタイムスクリプトです。このトピックでは、スクリプトの構文と使用を開始するための例を示します。スクリプトが正常に実行されたかどうかをテストする方法も示します。

PowerShell スクリプトを起動して実行するインスタンスを作成する

次の手順では、インスタンスの起動直後に chocolatey というツールを新しいインスタンスにインストールします。

1. Lightsail のホームページで [インスタンスの作成] を選択します。
2. インスタンスを作成する AWS リージョン およびアベイラビリティーゾーンを選択します。
3. [プラットフォームの選択] で [Microsoft Windows] を選択します。
4. [OS のみ] を選択してから、[Windows Server 2019]、[Windows Server 2016]、[Windows Server 2012 R2] を選択します。

5. [起動スクリプトの追加] を選択します。
6. 次の内容を入力します。

```
<powershell>
iex ((New-Object System.Net.WebClient).DownloadString('https://chocolatey.org/
install.ps1'))
</powershell>
```

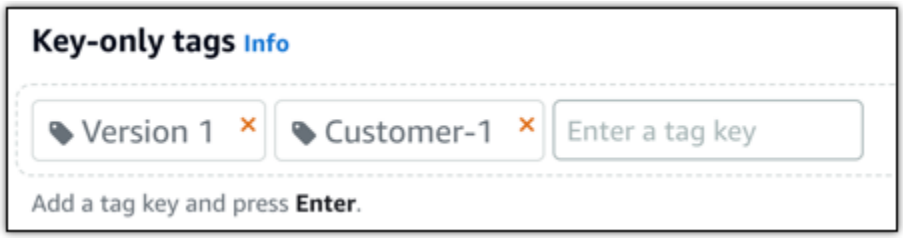
> **Note**
>
> PowerShell スクリプトは常に <powershell></powershell> タグで囲む必要があります。PowerShell コマンドまたはバッチスクリプトは、<script></script> タグを使用するか、タグをまったく使用せずに入力できます。

7. インスタンスの名前を入力します。

   リソース名:

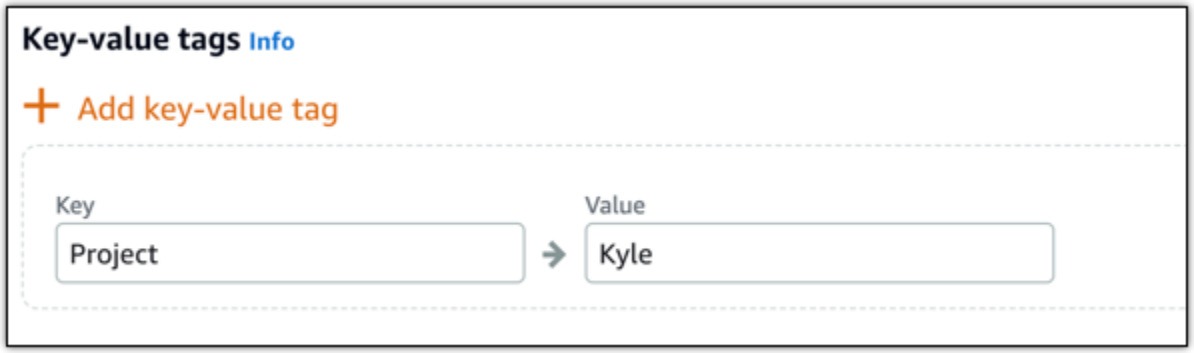
   - Lightsail アカウントの各 AWS リージョン 内で一意である必要があります。
   - 2〜255 文字を使用する必要があります。
   - 先頭と末尾は英数字または数字を使用する必要があります。
   - 英数字、数字、ピリオド、ダッシュ、アンダースコアを使用することができます。
8. 以下のいずれかのオプションを選択して、インスタンスにタグを追加します。

   - [key-only タグの追加] または [key-only タグの編集] (タグが追加済みの場合)。タグキーのテキストボックスに新しいタグを入力し、Enter キーを押します。タグの入力を完了したら、[保存] を選択してタグを追加し、追加しない場合は、[キャンセル] を選択します。

   - [key-value タグの作成] から [キー] テキストボックスにキーを入力し、[値] テキストボックスに値を入力します。タグの入力を完了したら、[保存] を選択し、追加しない場合は、[キャンセル] を選択します。

キーバリューのタグは、保存する際に一つずつ追加することができます。さらに key-value タグを追加するには、以上のステップを繰り返します。



> **Note**
>
> 「キーのみ」のタグと「キーバリュー」のタグの詳細については、「タグ」を参照してください。

9. [インスタンスの作成] を選択します。

スクリプトが正常に実行されたことを確認する

インスタンスにログインして、スクリプトが正常に実行されたことを確認できます。Windows ベースのインスタンスが RDP 接続を受け入れるようになるまで最大で 15 分かかります。準備ができたら、ブラウザベースの RDP クライアントを使用してログインするか、独自の RDP クライアントを設定します。詳細については、「Windows ベースのインスタンスに接続する」を参照してください。

1. Lightsail インスタンスに接続できるようになったら、コマンドプロントを開きます (または Windows エクスプローラーを開きます)。
2. 次のように入力して Log ディレクトリに移動します。

```
cd C:\ProgramData\Amazon\EC2-Windows\Launch\Log
```

> **Note**
>
> Windows Server 2012 では、コマンドは `cd C:\Program Files\Amazon\Ec2ConfigService\Logs` です。

3. テキストエディターで `UserdataExecution.log` を開くか、`type UserdataExecution.log` と入力します。

   ログファイルには次のように表示されます。

```
2017/10/11 20:32:12Z: <powershell> tag was provided.. running powershell content
2017/10/11 20:32:13Z: Message: The output from user scripts: iex ((New-Object
 System.Net.WebClient).DownloadString('https://chocolatey.org/install.ps1'))

2017/10/11 20:32:13Z: Userdata execution done
```

Lightsail における Windows Server ベースのインスタンスを保護するためのベストプラクティス

この記事では、Windows Server を実行する Lightsail インスタンスを使用する際にセキュリティリスクを回避するために役立つヒントやテクニックについて説明します。

Lightsail のパスワードについて

Windows Server ベースのインスタンスを作成するとき、Lightsail は推測しにくい長いパスワードをランダムに生成します。新しいインスタンスではこのパスワードを一意に使用します。デフォルトのパスワードを使用すると、リモートデスクトップ (RDP) を使用してインスタンスにすばやく接続できます。 インスタンスには常に AdministratorLightsail としてログインします。

パスワードの管理

Windows Server ベースのインスタンスのパスワードを覚えやすいものに変更できます。これは、リモートデスクトップクライアントを使用して Lightsail インスタンスにアクセスする場合に役立つ場合があります。生成したパスワードが Lightsail に保存されることはありません。

> **Note**
>
> Lightsail では、ブラウザベースの RDP クライアントに、Lightsail で生成されたパスワードまたは独自のカスタムパスワードを使用できます。カスタムパスワードを使用する場合、ログインするたびにパスワードの入力を求められます。インスタンスにすばやくアクセスする場合、ブラウザベースの RDP クライアントで Lightsail によって生成されたデフォルトパスワードを使用した方が簡単です。

管理者パスワードを安全に変更するには、Windows Server のパスワードマネージャーを使用します。`Ctrl + Alt + Del` を押し、[パスワードの変更] を選択します。Lightsail にはパスワードが保存されないため、必ずパスワードを記録しておいてください。パスワードを取得する必要がある場合は、以下の「Windows ベースのインスタンスの管理者パスワードを変更する」を参照してください。

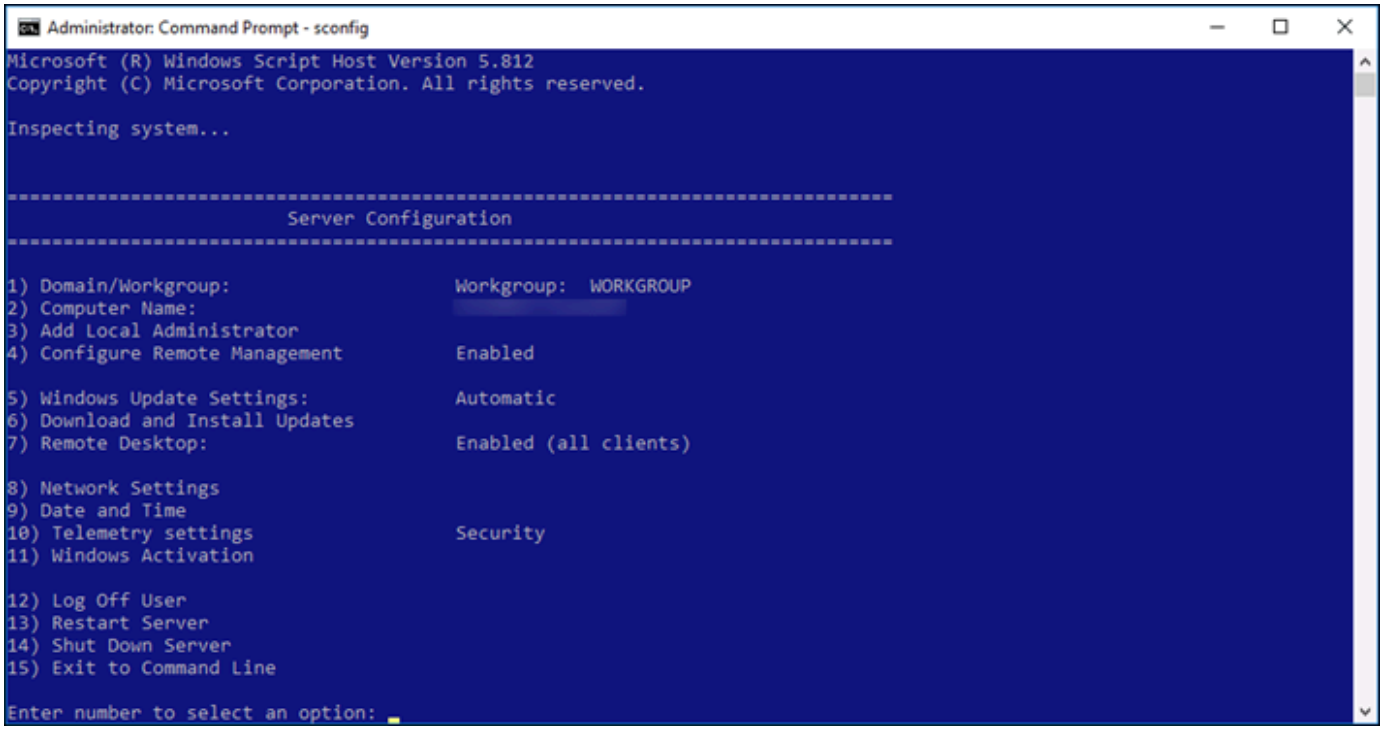
パスワードを一意のデフォルトパスワードから変更する場合、必ず強力なパスワードを使用してください。名前や辞書に載っている単語をベースとしたパスワードや、一連の文字の繰り返しは避けてください。

セキュリティパッチ

Windows Server ベースの Lightsail インスタンスを最新のセキュリティパッチで更新された状態に保つことをお勧めします。サーバーが更新をダウンロードおよびインストールするよう設定されていることを確認してください。次の手順では、Windows Server を実行する Lightsail インスタンスで直接これを行う方法について説明します。

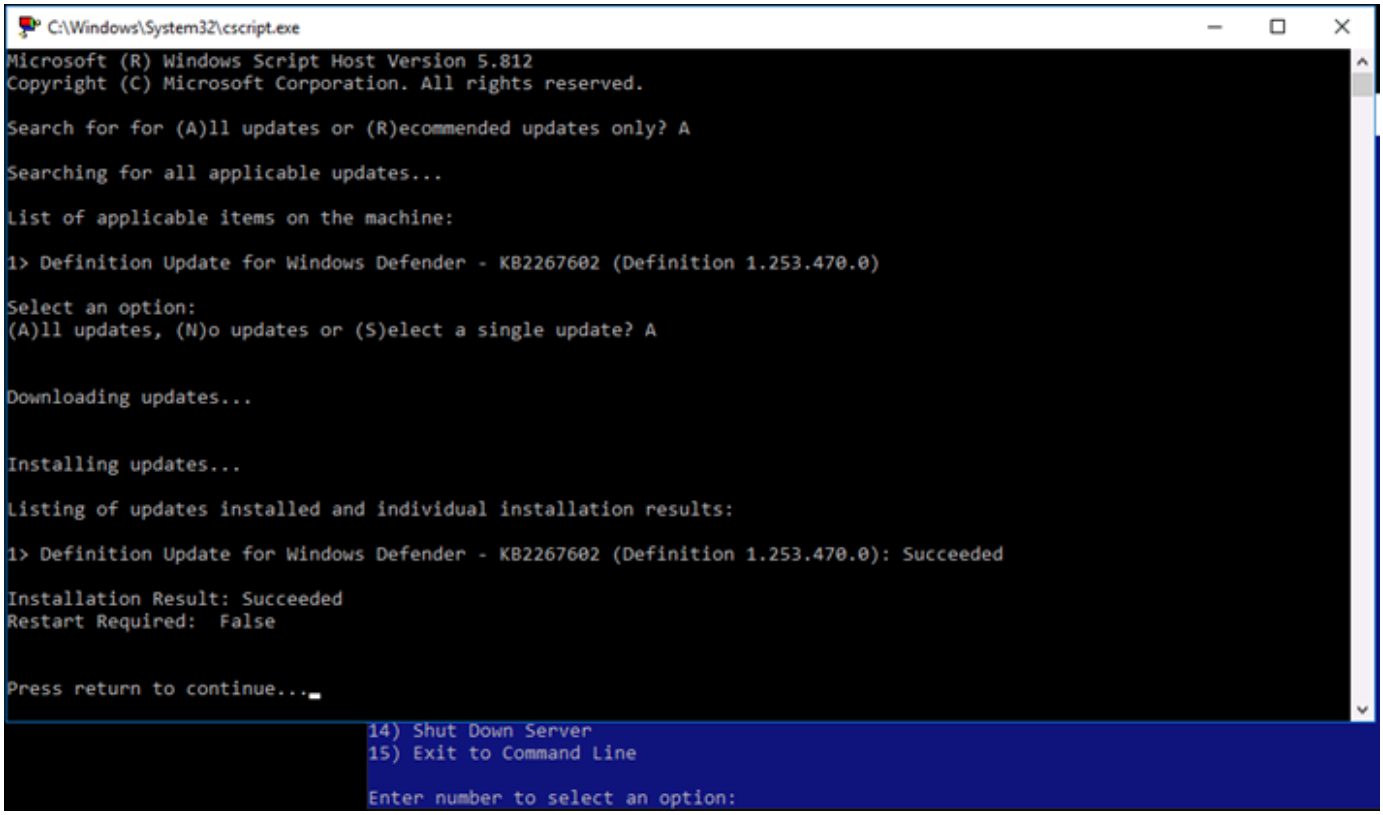
1. Windows Server ベースのインスタンスで、コマンドプロンプトを開きます。
2. 「`sconfig`」と入力し、`Enter` を押します。

   Windows Update Settings (5 番) はデフォルトで `Automatic` になっています。



3. 新しい更新プログラムをダウンロードしてインストールするには、6 と入力して Enter キーを押します。
4. 新しいコマンドウィンドウで「A」と入力して [(A)ll updates (すべての更新)] を検索し、Enter キーを押します。
5. もう一度「A」と入力して [(A)ll updates (すべての更新)] をインストールし、Enter キーを押します。

   完了したら、インストール結果と詳しい手順が記載されたメッセージが表示されます (該当する場合)。



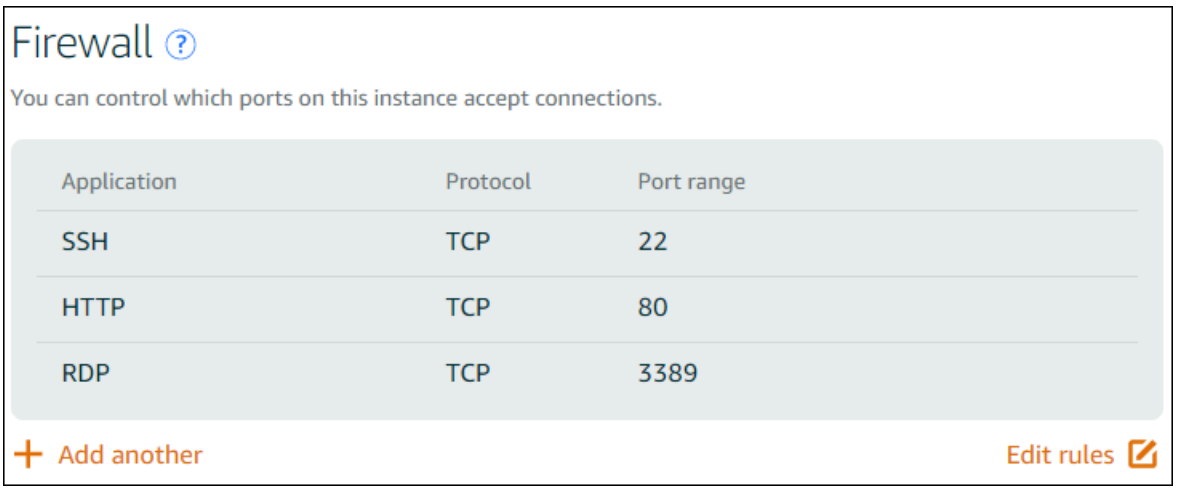
Windows Server でアカウントロックアウトポリシーを有効にする

ログイン試行に一定回数失敗した場合にアカウントを一時的または永続的に無効にするよう Windows Server を設定できます。たとえば、間違ったパスワードを 3 つ使用してインスタンスにログインしようとした場合にロックアウトすることができます。

詳細については、『Windows Server documentation』の「Account Lockout Policy」を参照してください。

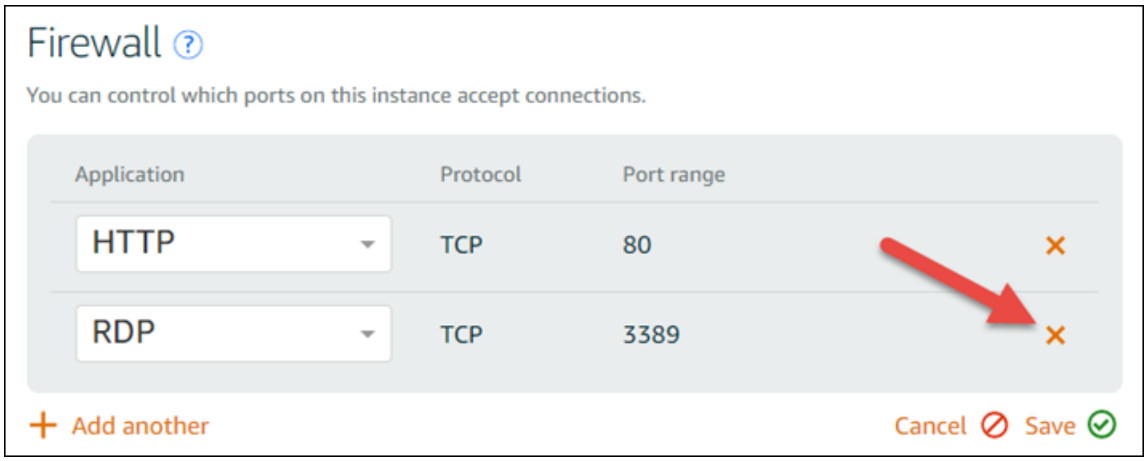
ポートとファイアウォールの設定

デフォルトでは、Windows Server ベースのインスタンスで次のポートを開きます。



有効にしたポートは世界中に公開され、ソース IP によって制限することはできません。インスタンスへのアクセスを制限するには、これらのポートを無効にし、インスタンスへのアクセスが必要なときにのみ有効にすることができます。その方法は次のとおりです。

1. Lightsail で管理するインスタンスを検索し、[管理] を選択します。
2. [ネットワーキング] を選択します。
3. インスタンスの [ネットワーキング] ページで、[ルールの編集] を選択します。
4. ルールの横にあるオレンジ色の [x] を選択することで、RDP/TCP/3389 ルールを削除します。



5. [保存] を選択します。

Lightsail ファイアウォールルールリファレンス

Amazon Lightsail インスタンスのファイアウォールには、インスタンスの役割を反映するルールを追加できます。たとえば、ウェブサーバーとして設定するインスタンスには、インバウンドの HTTP および HTTPS アクセスを許可するファイアウォールルールが必要です。データベースのインスタ

ンスには、データベースタイプへのアクセス (MySQL のポート 3306 を介したアクセスなど) を許可するルールが必要です。ファイアウォールの詳細については、「Lightsail のインスタンスファイアウォール」を参照してください。

このガイドでは、特定の種類のアクセスを対象として、インスタンスのファイアウォールに追加できるルールの種類を例として示します。ルールは、特に明記しない限り、アプリケーション、プロトコル、ポート、および送信元 IP アドレスのリスト (アプリケーション - プロトコル - ポート - 送信元 IP アドレスなど) として示します。

目次

ウェブサーバールール

次のインバウンドルールは、HTTP および HTTPS アクセスを許可します。

> **Note**
>
> 一部の Lightsail インスタンスには、デフォルトで以下のファイアウォールルールが設定されています。詳細については、「ファイアウォールとポート」を参照してください。

HTTP

HTTP - TCP - 80 - すべての IP アドレス

HTTPS

HTTPS - TCP - 443 - すべての IP アドレス

コンピュータからインスタンスに接続するためのルール

インスタンスに接続するには、SSH アクセス (Linux インスタンスの場合) または RDP アクセス (Windows インスタンスの場合) を許可するルールを追加します。

> **Note**
>
> すべての Lightsail インスタンスには、デフォルトで以下のファイアウォールルールのいずれかが設定されています。詳細については、「ファイアウォールとポート」を参照してください。

SSH

SSH - TCP - 22 - コンピュータのパブリック IP アドレス、またはローカルネットワークの IP アドレス範囲 (CIDR ブロック表記)。

RDP

RDP - TCP - 3389 - コンピュータのパブリック IP アドレス、またはローカルネットワークの IP アドレス範囲 (CIDR ブロック表記)。

データベースサーバールール

次のインバウンドルールは、インスタンスで実行中のデータベースのタイプに応じて、データベースアクセス用に追加できるルールの例です。

SQL Server

カスタム - TCP - 1433 - コンピュータのパブリック IP アドレス、またはローカルネットワークの IP アドレス範囲 (CIDR ブロック表記)。

MySQL/Aurora

MySQL/Aurora - TCP - 3306 - コンピュータのパブリック IP アドレス、またはローカルネットワークの IP アドレス範囲 (CIDR ブロック表記)。

PostgreSQL

PostgreSQL - TCP - 5432 - コンピュータのパブリック IP アドレス、またはローカルネットワークの IP アドレス範囲 (CIDR ブロック表記)。

Oracle-RDS

Oracle - RDS - TCP - 1521 - コンピュータのパブリック IP アドレス、またはローカルネットワークの IP アドレス範囲 (CIDR ブロック表記)。

Amazon Redshift

カスタム - TCP - 5439 - コンピュータのパブリック IP アドレス、またはローカルネットワークの IP アドレス範囲 (CIDR ブロック表記)。

DNS サーバールール

インスタンスを DNS サーバーとして設定した場合、TCP および UDP のトラフィックは、ポート 53 経由で DNS サーバーに到達できる必要があります。

DNS (TCP)

DNS (TCP) - 53 - コンピュータの IP アドレス、またはローカルネットワークの IP アドレス範囲 (CIDR ブロック表記)。

DNS (UDP)

DNS (UDP) - UDP - 53 - コンピュータの IP アドレス、またはローカルネットワークの IP アドレス範囲 (CIDR ブロック表記)。

SMTP メール

インスタンスで SMTP を有効にするには、次のファイアウォールルールを設定する必要があります。

> **Important**
>
> 次のルールを設定したら、インスタンスの逆引き DNS も設定する必要があります。そうしないと、E メールは TCP ポート 25 経由に制限される場合があります。詳細については、「E メールサーバーの逆引き DNS を設定する」を参照してください。

SMTP

カスタム - TCP - 25 - インスタンスと通信するホストの IP アドレス。

Amazon Lightsail のインスタンスファイアウォール

Amazon Lightsail コンソールのファイアウォールは、パブリック IP アドレスを介してインスタンスに接続できるトラフィックを制御する仮想ファイアウォールとして機能します。Lightsail で作成する各インスタンスには、IPv4 アドレス用と IPv6 アドレス用の 2 つのファイアウォールがあります。各ファイアウォールには、インスタンスに着信するトラフィックをフィルタリングする一連のルールが含まれています。各ファイアウォールは、互いに独立しています。IPv4 と IPv6 のファイアウォールルールを別個に設定する必要があります。インスタンスのファイアウォールは、トラフィックを許可または制限するルールを追加および削除することで、いつでも編集できます。

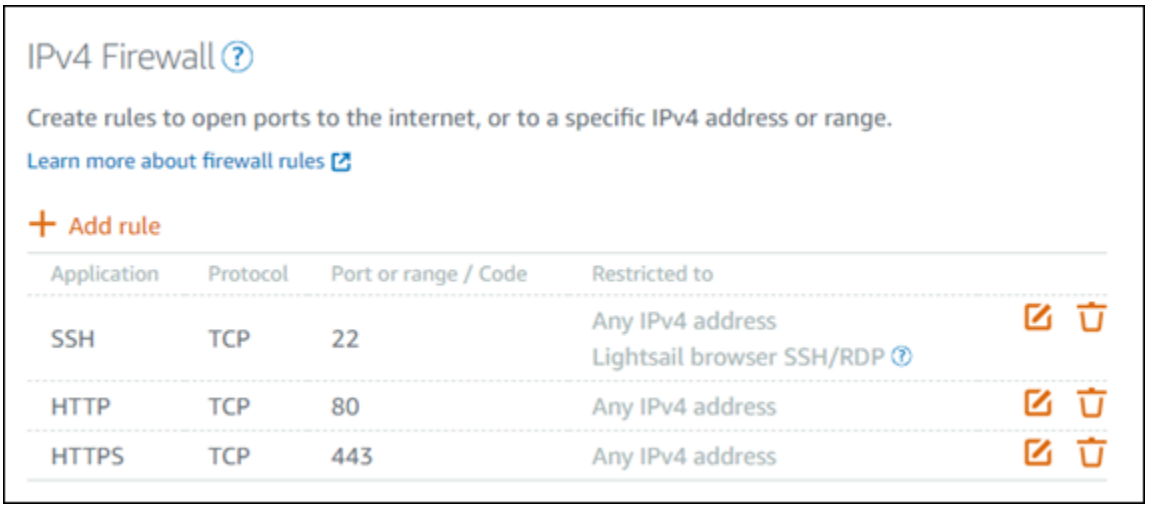
目次

Lightsail ファイアウォール

各 Lightsail インスタンスには、IPv4 アドレス用と IPv6 アドレス用の 2 つのファイアウォールがあります。Lightsail インスタンスに出入りするすべてのインターネットトラフィックは、そのファイアウォールを通過します。インスタンスのファイアウォールは、インスタンスへの流入を許可されたインターネットトラフィックを制御します。ただし、送信トラフィックは制御しません。 ファイアウォールは、すべてのアウトバウンドトラフィックを許可します。インスタンスのファイアウォールは、受信ラフィックを許可または制限するルールを追加および削除することで、いつでも編集できます。各ファイアウォールは、互いに独立しています。IPv4 と IPv6 のファイアウォールルールを別個に設定する必要があります。

ファイアウォールルールは常にアクセスを許可します。アクセスを拒否するルールを作成することはできません。インスタンスへの着信トラフィックを許可するルールをインスタンスのファイアウォールに追加します。インスタンスのファイアウォールにルールを追加するときは、以下の例 (IPv4) に示すように、使用するプロトコル、開くポート、インスタンスに接続できる IPv4 および IPv6 アド

レスを指定します。アプリケーションレイヤーのプロトコルタイプも指定できます。これは、インスタンスで使用するサービスに応じて、プロトコルとポート範囲を自動的に指定するプリセットです。



> **Important**
>
> ファイアウォールルールが影響するのは、インスタンスのパブリック IP アドレス経由で受信されるトラフィックのみです。同じ のアカウント内の Lightsail リソース、または同じ のピアリングされた仮想プライベートクラウド (VPC) 内のリソースから発信される可能性のある AWS リージョン、インスタンスのプライベート IP アドレスを通過するトラフィックには影響しません AWS リージョン。

ファイアウォールルールおよび設定可能なパラメータについては、このガイドの以降のセクションで説明します。

ファイアウォールルールを作成する

ファイアウォールルールを作成して、クライアントがインスタンスまたはインスタンスで実行されているアプリケーションとの接続を確立できるようにします。例えば、すべてのウェブブラウザがインスタンス上の WordPress アプリケーションに接続できるようにするには、任意の IP アドレスからポート 80 経由で Transmission Control Protocol (TCP) を有効にするファイアウォールルールを設定します。インスタンスのファイアウォールでこのルールがすでに設定されている場合は、ルールを削除して、ウェブブラウザがインスタンス上の WordPress アプリケーションに接続できないようにすることができます。

> **Important**
>
> Lightsail コンソールを使用して、一度に最大 30 個の送信元 IP アドレスを追加できます。一度に最大 60 個の IP アドレスを追加するには、Lightsail API、 AWS Command Line Interface（AWS CLI）、または AWS SDK を使用します。このクォータは、IPv4 ルールと IPv6 ルールに対して個別に適用されます。例えば、ファイアウォールで、IPv4 トラフィックと IPv6 トラフィックのそれぞれに 60 個のインバウンドルールを設定できます。個々の IP アドレスを CIDR 範囲に統合することをお勧めします。詳細については、このガイドの「送信元 IP アドレスの指定」セクションを参照してください。

また、SSH クライアントからインスタンスへの接続を許可してサーバーで管理タスクを実行できます。そのためには、接続を確立する必要があるコンピュータの IP アドレスに限り、ポート 22 を介した TCP を許可するファイアウォールルールを設定します。この場合、インスタンスへの SSH 接続の確立をすべての IP アドレスに許可しないようにします。許可すると、インスタンスのセキュリティリスクが生じる可能性があります。

> **Note**
>
> このセクションで説明するファイアウォールルールの例は、インスタンスのファイアウォールにデフォルトで設定済みである場合があります。詳細については、このガイドの後半の「デフォルトのファイアウォールルール」を参照してください。

特定のポートに複数のルールがある場合、最も許容度の大きいルールを適用します。たとえば、IP アドレス 192.0.2.1 からの TCP ポート 22 (SSH) へのアクセスを許可するルールを追加したとします。次に、すべてのユーザーからの TCP ポート 22 へのアクセスを許可する別のルールを追加します。これにより、すべてのユーザーが TCP ポート 22 にアクセスできるようになります。

プロトコルを指定する

プロトコルは 2 台のコンピューター間でデータを送信する形式です。Lightsail では、ファイアウォールルールで次のプロトコルを指定できます。

- Transmission Control Protocol (TCP) は主に、クライアントとインスタンスで実行されているアプリケーション間の接続を確立して、データの交換が完了するまで接続を維持するために使用されます。これは広く使用されており、ファイアウォールルールで指定することが多いプロトコルです。TCP は、送信されたデータに欠落がないこと、および送信されたすべてのデータが目的の受

信者に到達することを保証します。ウェブブラウジング、金融取引、テキストメッセージングなど、高い信頼性が要求されるが、転送時間の重要度が低いネットワークアプリケーションに最適です。これらのユースケースでは、データの一部が失われると、重大な価値の損失となります。

- UDP (User Datagram Protocol) は、インスタンスで実行されているアプリケーションとクライアントとの間で低レイテンシーおよび損失許容接続を確立するために主に使用します。ゲーム、音声、ビデオ通信など、体感レイテンシーの重要度が高いネットワークアプリケーションに最適です。これらのユースケースでは、多少のデータ損失が生じる場合がありますが、体感品質を損なうほどではありません。
- ICMP (Internet Control Message Protocol) は、ネットワーク通信の問題を診断するために主に使用します。たとえば、データが送信先にタイムリーに到着しているかどうかを確認します。このプロトコルは Ping ユーティリティに最適です。このユーティリティでは、ローカルコンピュータとインスタンス間の接続速度をテストできます。データがインスタンスに到着してローカルコンピュータに戻ってくるまでの所要時間をレポートします。

  > **Note**
  >
  > Lightsail コンソールを使用してインスタンスの IPv6 ファイアウォールに ICMP ルールを追加すると、ICMPv6 を使用するようにルールが自動的に設定されます。詳細については、Wikipedia の「IPv6 のインターネット制御メッセージプロトコル」を参照してください。

- すべてでは、インスタンスへのすべてのプロトコルトラフィックの流入を許可します。どのプロトコルを指定すればよいかわからない場合は、このプロトコルを指定します。これには、上に示したプロトコルだけでなく、すべてのインターネットプロトコルが含まれます。詳細については、「Protocol Numbers」(Internet Assigned Numbers Authority ウェブサイト) を参照してください。

ポートの指定

コンピュータがキーボードやマウスなどの周辺機器と通信するためのコンピュータの物理ポートと同様に、ネットワークポートはインスタンスのインターネット通信エンドポイントとして機能します。コンピュータは、インスタンスと接続するときに、通信を確立するためのポートを公開します。

ファイアウォールルールで指定できるポートの範囲は 0 ~ 65535 です。インスタンスへの接続をクライアントに許可するファイアウォールを作成する場合、使用するプロトコル (このガイドの前半で説明) と、接続の確立に使用できるポート番号を指定します。プロトコルとポートを使用して接続を確立できる IP アドレスを指定することもできます。これについては、このガイドの次のセクションで説明します。

よく使用されるポートと、これらのポートを使用するサービスは以下のとおりです。

- FTP (File Transfer プロトコル) を介したデータ転送では、ポート 20 を使用します。
- FTP に介したコマンド制御では、ポート 21 を使用します。
- Secure Shell (SSH) では、ポート 22 を使用します。
- Telnet リモートログインサービス、および暗号化されていないテキストメッセージでは、ポート 23 を使用します。
- SMTP (Simple Mail Transfer Protocol) では、E メールの送信にポート 25 を使用します。

> **Important**
>
> インスタンスで SMTP を有効にするには、インスタンスにリバース DNS も設定する必要があります。そうしないと、E メールが TCP ポート 25 経由に制限される可能性があります。詳細については、「Amazon Lightsail インスタンスでの E メールサーバーの逆引き DNS の設定」を参照してください。

- ドメインネームシステム (DNS) サービスでは、ポート 53 を使用します。
- ウェブブラウザがウェブサイトに接続するために使用する HTTP (ハイパーテキスト転送プロトコル) では、ポート 80 を使用します。
- E メールクライアントがサーバーから E メールを取得するために使用する POP3 (Post Office Protocol) では、ポート 110 を使用します。
- NNTP (Network News Transfer Protocol) では、ポート 119 を使用します。
- NTP (Network Time Protocol) では、ポート 123 を使用します。
- デジタルメールを管理するために使用する IMAP (インターネットメッセージアクセスコントロール) では、ポート 143 を使用します。
- SNMP (簡易ネットワーク管理プロトコル) では、ポート 161 を使用します。
- ウェブブラウザがウェブサイトへの暗号化された接続を確立するために使用する TLS/SSL を介した HTTP Secure (HTTPS) HTTP では、ポート 443 を使用します。

詳細については、「Service Name and Transport Protocol Port Number Registry」(Internet Assigned Numbers Authority ウェブサイト) を参照してください。

アプリケーションレイヤーのプロトコルタイプを指定する

ファイアウォールルールの作成時に、アプリケーションレイヤーのプロトコルタイプを指定できます。プロトコルタイプは、インスタンスで有効にしたサービスに応じてルールのプロトコルとポー

ト範囲を指定するプリセットです。これにより、SSH、RDP、HTTP などのサービスで使用する一般的なプロトコルやポートを検索する必要がなくなります。これらのアプリケーションレイヤーのプロトコルタイプを選択するだけで、プロトコルとポートが自動的に指定されます。独自のプロトコルとポートを指定する場合は、アプリケーションレイヤーのプロトコルタイプとして [カスタムルール] を選択できます。これにより、該当するパラメータを制御できます。

> **Note**
>
> アプリケーションレイヤープロトコルタイプは、Lightsail コンソールを使用してのみ指定できます。Lightsail API、 AWS Command Line Interface （ AWS CLI ）、または SDKsを使用してアプリケーションレイヤープロトコルタイプを指定することはできません。

Lightsail コンソールでは、次のアプリケーションレイヤープロトコルタイプを使用できます。

- カスタム - 独自のプロトコルとポートを指定する場合は、このオプションを選択します。
- すべてのプロトコル – すべてのプロトコルを指定して、独自のポートを指定する場合は、このオプションを選択します。
- すべての TCP – TCP プロトコルを使用するが、どのポートを開けばよいかわからない場合は、このオプションを選択します。これにより、すべてのポート (0 ~ 65535) を介した TCP が有効になります。
- すべての UDP – UDP プロトコルを使用するが、どのポートを開けばよいかわからない場合は、このオプションを選択します。これにより、すべてのポート (0 ~ 65535) で UDP が有効になります。
- すべての ICMP — すべての ICMP タイプとコードを指定するには、このオプションを選択します。
- カスタム ICMP - ICMP プロトコルを使用し、ICMP のタイプとコードを定義する場合は、このオプションを選択します。ICMP タイプとコードの詳細については、Wikipedia の「制御メッセージ」を参照してください。
- DNS – インスタンスで DNS を有効にする場合は、このオプションを選択します。これにより、ポート 53 を介した TCP および UDP が有効になります。
- HTTP – インスタンスでホストされているウェブサイトへの接続をウェブブラウザに許可する場合は、このオプションを選択します。これにより、ポート 80 を介した TCP が有効になります。
- HTTPS – インスタンスでホストされているウェブサイトへの暗号化された接続の確立をウェブブラウザに許可する場合は、このオプションを選択します。これにより、ポート 443 を介した TCP が有効になります。

- MySQL/Aurora – インスタンスでホストされている MySQL または Aurora データベースへの接続をクライアントに許可する場合は、このオプションを選択します。これにより、ポート 3306 を介した TCP が有効になります。
- Oracle-RDS – インスタンスでホストされている Oracle または RDS データベースへの接続をクライアントに許可する場合は、このオプションを選択します。これにより、ポート 1521 を介した TCP が有効になります。
- Ping (ICMP) – Ping ユーティリティを使用してリクエストに応答することをインスタンスに許可する場合は、このオプションを選択します。IPv4 ファイアウォールでは、ICMP タイプ 8 (エコー) とコード -1 (すべてのコード) が有効になります。IPv6 ファイアウォールでは、ICMP タイプ 129 (エコー応答) とコード 0 が有効になります。
- RDP – インスタンスへの接続を RDP クライアントに許可する場合に、このオプションを選択します。これにより、ポート 3389 を介した TCP が有効になります。
- SSH – インスタンスへの接続を SSH クライアントに許可する場合に、このオプションを選択します。これにより、ポート 22 を介した TCP が有効になります。

送信元 IP アドレスを指定する

ファイアウォールルールは、デフォルトですべての IP アドレスに対して、指定したプロトコルとポートを介してインスタンスに接続することを許可します。これは、HTTP や HTTPS を経由するウェブブラウザなどのトラフィックに最適です。ただし、SSH や RDP などのトラフィックの場合、これらのアプリケーションを使用してインスタンスに接続することをすべての IP アドレスに許可することは、セキュリティ上のリスクとなります。そのため、ファイアウォールルールを IPv4 または IPv6 アドレス、あるいは IP アドレス範囲に制限できます。

- IPv4 アドレスについて - 単一の IPv4 アドレス(203.0.113.1 など) または IPv4 アドレス範囲を指定できます。Lightsail コンソールでは、範囲はダッシュ (192.0.2.0-192.0.2.255 など) または CIDR ブロック表記 (192.0.2.0/24 など) を使用して指定できます。CIDR ブロックの表記の詳細については、Wikipedia の「Classless Inter-Domain Routing」記事を参照してください。
- IPv6ファイアウォールについて - 単一の IPv6 アドレス (2001:0db8:85a3:0000:0000:8a2e:0370:7334 など) または IPv6 アドレス範囲を指定できます。Lightsail コンソールでは、IPv6 の範囲は CIDR ブロック表記 (2001:db8:: /32 など) で指定できます。IPv6 CIDR ブロック表記の詳細については、Wikipedia の「IPv6 CIDR ブロック」を参照してください。

デフォルトの Lightsail ファイアウォールルール

新しいインスタンスを作成すると、そのインスタンスへの基本的なアクセスを許可する以下のデフォルトのルールが、IPv4 および IPv6 のファイアウォールに事前設定されます。デフォルトのルールは、作成するインスタンスのタイプに応じて異なります。これらのルールは、アプリケーション、プロトコル、ポート、および送信元 IP アドレスのリスト (アプリケーション - プロトコル - ポート - 送信元 IP アドレスなど) として示してあります。

AlmaLinux、Amazon Linux 2、Amazon Linux 2023、CentOS、Debian、FreeBSD、openSUSE、Ubuntu (ベースオペレーティングシステム)

- SSH - TCP - 22 - すべての IP アドレス
- HTTP - TCP - 80 - すべての IP アドレス

WordPress、Ghost、Joomla! PrestaShop、Drupal (CMS アプリケーション)

- SSH - TCP - 22 - すべての IP アドレス
- HTTP - TCP - 80 - すべての IP アドレス
- HTTPS - TCP - 443 - すべての IP アドレス

cPanel & WHM (CMS アプリケーション)

- SSH - TCP - 22 - すべての IP アドレス
- DNS (UDP) - UDP - 53 - すべての IP アドレス
- DNS (TCP) - TCP - 53 - すべての IP アドレス
- HTTP - TCP - 80 - すべての IP アドレス
- HTTPS - TCP - 443 - すべての IP アドレス
- カスタム - TCP - 2078 - すべての IP アドレス
- カスタム - TCP - 2083 - すべての IP アドレス
- カスタム - TCP - 2087 - すべての IP アドレス
- カスタム - TCP - 2089 - すべての IP アドレス

LAMP、Django、Node.js、MEAN GitLab、Nginx (開発スタック)

- SSH - TCP - 22 - すべての IP アドレス

HTTP - TCP - 80 - すべての IP アドレス

HTTPS - TCP - 443 - すべての IP アドレス

Magento (e コマースアプリケーション)

SSH - TCP - 22 - すべての IP アドレス

HTTP - TCP - 80 - すべての IP アドレス

HTTPS - TCP - 443 - すべての IP アドレス

Redmine (プロジェクト管理アプリケーション)

SSH - TCP - 22 - すべての IP アドレス

HTTP - TCP - 80 - すべての IP アドレス

HTTPS - TCP - 443 - すべての IP アドレス

Plesk (ホスティングスタック)

SSH - TCP - 22 - すべての IP アドレス

HTTP - TCP - 80 - すべての IP アドレス

HTTPS - TCP - 443 - すべての IP アドレス

カスタム - TCP - 53 - すべての IP アドレス

カスタム - UDP - 53 - すべての IP アドレス

カスタム - TCP -8443 - すべての IP アドレス

カスタム - TCP - 8447 - すべての IP アドレス

Windows Server 2022、Windows Server 2019、および Windows Server 2016

SSH - TCP - 22 - すべての IP アドレス

HTTP - TCP - 80 - すべての IP アドレス

RDP - TCP - 3389 - すべての IP アドレス

SQL Server Express 2022、SQL Server Express 2019、SQL Server Express 2016

SSH - TCP - 22 - すべての IP アドレス

HTTP - TCP - 80 - すべての IP アドレス

RDP - TCP - 3389 - すべての IP アドレス

ファイアウォールに関する他の参照情報

Lightsail でファイアウォールを管理するのに役立つ記事を以下に示します。

- インスタンスのファイアウォールルールを追加および編集する
- ファイアウォールルールのリファレンス

Amazon Lightsail のインスタンスのファイアウォールルールを追加および編集する

Amazon Lightsail インスタンスのファイアウォールにルールを追加して、インスタンスに接続できるトラフィックを制御できます。ファイアウォールルールを追加する場合、インスタンスの接続が許可されているアプリケーションレイヤーのプロトコルタイプ、プロトコル、ポート、および送信元 IPv4 or IPv6 アドレスを指定できます、ファイアウォールの詳細については、「ファイアウォールとポート」を参照してください。

目次

インスタンスのファイアウォールルールを追加および編集する

Lightsail コンソールでファイアウォールルールを追加または編集するには、次の手順を実行します。

1. Lightsail コンソールにサインインします。
2. Lightsail ホームページで、[Instances (インスタンス)] タブを選択します。
3. ファイアウォールルールを追加または編集する対象のインスタンスの名前を選択します。
4. インスタンスの管理ページで [ネットワーキング] タブを選択します。

   [ネットワーキング] タブには、インスタンスのパブリック IP アドレスとプライベート IP アドレス、インスタンスに設定された IPv4 や IPv6 ファイアウォールルールが表示されます。

> **Note**
>
> IPv6 ファイアウォールは、インスタンスで IPv6 を有効にしている場合にのみ表示されます。詳細については、「IPv6 を有効化または無効化する」を参照してください。

5. ルールの送信元 IP が IPv4 アドレスか IPv6 アドレスかに応じて、次のいずれかの手順を実行します。

   - ページの [ファイアウォール] セクションまでスクロールし、[ルールの追加] を選択します。
   - ページの [ファイアウォール] セクションまでスクロールし、[ルールの追加] を選択します。

   既存のルールの横にある [編集] (鉛筆アイコン) を選択して編集することもできます。

6. [アプリケーション] ドロップダウンメニューからアプリケーションレイヤーのプロトコルタイプを選択します。

   アプリケーションレイヤーのプロトコルタイプを選択すると、プロトコルとポートのプリセットが自動的に指定されます。値の例は [Custom (カスタム)]、[All TCP (すべての TCP)]、[All UDP (すべての UDP)]、[Custom ICMP (カスタム ICMP)]、[SSH]、[RDP] です。

   選択したアプリケーションレイヤープロトコルタイプに応じて、以下のオプション設定を定義できます。

   - (オプション) [Custom (カスタム)] オプションを選択すると、[プロトコル] ドロップダウンメニューから値を選択できます。使用可能なプロトコル値は [TCP] および [UDP] です。

     [Port (ポート)] フィールドに 1 つのポート番号またはポート番号の範囲 (7000 ~ 8000 など) を入力することもできます。
   - (オプション) [Custom ICMP (カスタム ICMP)] オプションを選択した場合、[Type (タイプ)] フィールドで ICMP タイプを、[Code (コード)] フィールドで ICMP コードを指定できます。ICMP タイプとコードに関する詳細については、ウィキペディアの「コントロールメッセージ」を参照してください。

     > **Note**
     >
     > Lightsail コンソールを使用してインスタンスの IPv6 ファイアウォールに ICMP ルールを追加すると、このルールは ICMPv6 を使用するように自動的に設定されます。詳

細については、ウィキペディアの「IPv6 のインターネットコントロールメッセージプロトコル」を参照してください。

- (オプション) 指定したプロトコルとポートへのアクセスを特定の IP アドレスや IP アドレス範囲に制限するには、[IP アドレスに制限する] を選択します。指定したプロトコルとポートに対してすべての IP アドレスを許可する場合は、このオプションをオフのままにします。

  1 つの IPv4 アドレス (`203.0.113.1` など) または IPv4 アドレスの範囲を入力できます。その範囲は、ダッシュ表記 (`192.0.2.0-192.0.2.255` など) または CIDR ブロック表記 (`192.0.2.0/24` など) で指定できます。CIDR ブロック表記の詳細については、ウィキペディアの「Classless Inter-Domain Routing」記事を参照してください。

- (オプション) [SSH] または [RDP] アプリケーションレイヤープロトコルタイプを選択してから [IP アドレスに制限する] を選択する場合は、[Lightsail ブラウザの SSH/RDP を許可する] を選択して、Lightsail コンソールで利用できるブラウザベースの SSH および RDP クライアントを使用したインスタンスへの接続を許可することができます。これらのブラウザベースのクライアントからのアクセスをブロックするには、このオプションをオフのままにします。

7. ルールをファイアウォールに追加するには、[作成] を選択します。

   しばらくすると、ファイアウォールルールが追加されます。

インスタンスのファイアウォールルールを削除する

Lightsail コンソールでインスタンスのファイアウォールルールを削除するには、次の手順を実行します。

1. Lightsail コンソールにサインインします。
2. Lightsail ホームページで、[Instances (インスタンス)] タブを選択します。
3. ファイアウォールルールを削除する対象のインスタンスの名前を選択します。
4. インスタンスの管理ページで [ネットワーキング] タブを選択します。
5. ルールの送信元 IP が IPv4 アドレスか IPv6 アドレスかに応じて、次のいずれかの手順を実行します。

   - IPv4 ページの [ファイアウォール] セクションまでスクロールし、既存のルールの横にある [削除] (ゴミ箱アイコン) を選択して削除します。
   - IPv6 ページの [ファイアウォール] セクションまでスクロールし、既存のルールの横にある [削除] (ゴミ箱アイコン) を選択して削除します。

> **Important**
>
> ファイアウォールルールが影響するのは、インスタンスのパブリック IP アドレス経由で受信されるトラフィックのみです。このルールは、同じ AWS リージョン 内のアカウントにある Lightsail リソース、または同じ AWS リージョン 内のピアリングされた仮想プライベートクラウド (VPC) にあるリソースを送信元とする、インスタンスのプライベート IP アドレス経由で受信されるトラフィックには影響しません。例えば、インスタンスのファイアウォールから SSH ルール (TCP ポート 22) を削除する場合、同じ AWS リージョン 内の同じ Lightsail アカウントにある他のインスタンスは、インスタンスのプライベート IP アドレスを指定することによって引き続き SSH を使用して接続することができます。

しばらくすると、ファイアウォールルールが削除されます。

ファイアウォールに関する他の参照情報

Lightsail でのファイアウォールの管理に役立つ記事をいくつか以下に示します。

- ファイアウォールとポート
- ファイアウォールルールのリファレンス

Lightsail におけるインスタンスメタデータサービス (IMDS) とユーザーデータ

インスタンスメタデータは、インスタンスに関するデータで、実行中のインスタンスを設定または管理するために使用します。インスタンスメタデータは、ホスト名、イベント、およびセキュリティグループなどでカテゴリ分けされます。インスタンスメタデータを使用して、インスタンスの起動時に指定したユーザーデータにアクセスすることもできます。例えば、インスタンスを設定するためにパラメータを指定したり、単純なスクリプトを含めたりすることができます。インスタンスには、インスタンスの起動時に生成されるインスタンスアイデンティティドキュメントなどの動的データも含まれます。

> **Important**
>
> インスタンスメタデータおよびユーザーデータにはそのインスタンス自体内からのみアクセスできるものの、データは認証または暗号化手法によって保護されていません。インスタンス、そしてインスタンス上で実行される任意のソフトウェアに対して直接アクセス権がある可能性がある人は、メタデータを表示できます。そのため、パスワードまたは存続期間の長い暗号化キーなどの機密データは、ユーザーデータとして保管しないようにしてください。

Instance Metadata Service を使う

次のいずれかのメソッドを使って、Lightsail 内の実行中のインスタンスからインスタンスメタデータにアクセスできます。

- インスタンスメタデータサービスバージョン 1 (IMDSv1) – リクエスト/レスポンスメソッド
- インスタンスメタデータサービスバージョン 2 (IMDSv2) – セッション指向メソッド

> **Important**
>
> Lightsail のすべてのインスタンスブループリントが IMDSv2 をサポートされているわけではありません。`MetadataNoToken` インスタンスメトリクスは、IMDSv1 を使用しているインスタンスメタデータサービスへの呼び出しの数を追跡します。詳細については、「インスタンスのメトリクスを表示する」を参照してください。

IDMS の詳細については、「インスタンスメタデータサービス (IMDS) の設定」を参照してください。

IMDS 関連の追加のドキュメント

次の IMDS ドキュメントは、「Linux インスタンス用 Amazon Elastic Compute Cloud ユーザーガイド」と「Windows インスタンス用 Amazon Elastic Compute クラウドユーザーガイド」で利用できます。

> **Note**
>
> Amazon EC2 では、インスタンスのブループリントは Amazon マシンイメージ (AMIs) と呼ばれます。

- Linux インスタンスの場合:
  - インスタンスメタデータオプションの設定
  - インスタンスメタデータの取得
  - インスタンスユーザーデータの使用
  - 動的データの取得
  - インスタンスメタデータのカテゴリ
  - 例: AMI 作成インデックス値
  - インスタンスアイデンティティドキュメント
- Windows インスタンスの場合:
  - インスタンスメタデータオプションの設定
  - インスタンスメタデータの取得
  - インスタンスユーザーデータの使用
  - 動的データの取得
  - インスタンスメタデータのカテゴリ
  - 例: AMI 作成インデックス値
  - インスタンスアイデンティティドキュメント

Lightsail でインスタンスメタデータサービス (IMDS) を設定する

次のいずれかのメソッドを使用して、実行中のインスタンスからインスタンスメタデータにアクセスできます。

- インスタンスメタデータサービスバージョン 1 (IMDSv1) – リクエスト/レスポンスメソッド
- インスタンスメタデータサービスバージョン 2 (IMDSv2) – セッション指向メソッド

> **Important**
>
> Lightsail のすべてのインスタンスブループリントが IMDSv2 をサポートされているわけではありません。`MetadataNoToken` インスタンスメトリクスは、IMDSv1 を使用しているインスタンスメタデータサービスへの呼び出しの数を追跡します。詳細については、「インスタンスのメトリクスを表示する」を参照してください。

デフォルトでは、IMDSv1またはIMDSv2のいずれか、あるいは両方を使用できます。インスタンスメタデータサービスは、IMDSv2に固有の PUT ヘッダーまたは GET ヘッダーがリクエストに存在するかどうかに基づいて、IMDSv1 リクエストと IMDSv2 リクエ ストを区別します。詳細については、「EC2 Instance Metadata Service の拡張により、オープンファイアウォール、リバースプロキシ、および SSRF の脆弱性に対して多層防御を追加する」を参照してください。

ローカルコードまたはユーザーに IMDSv2を使用させるように、各インスタンスのインスタンスメタデータサービスを構成することができます。IMDSv2を使用しなければならないように指定すると、IMDSv1はもう機能しなくなります。詳細については、「Linux インスタンス用 Amazon Elastic Compute Cloud ユーザーガイド」の「インスタンスのメタデータオプションを設定する」を参照してください。

インスタンスのメタデータを取得するには、「Linux インスタンス用 Amazon Elastic Compute Cloud ユーザーガイド」の「インスタンスのメタデータを取得する」を参照してください。

> **Note**
>
> このセクションの例では、インスタンスメタデータサービスの IPv4 アドレスを使用します (`169.254.169.254`)。IPv6 アドレスを使用してインスタンスのインスタンスメタデータを取得する場合は、IPv6 アドレスを有効にして使用してください (`fd00:ec2::254`)。インスタンスメタデータサービスの IPv6 アドレスは、IMDSv2 コマンドと互換性があります。

インスタンスメタデータサービスバージョン 2 の仕組み

IMDSv2 は、セッション指向リクエストを使用します。セッション指向リクエストを使用して、セッション期間 (1 秒 ~ 6 時間) を定義するセッショントークンを作成します。指定した期間中、それ以降のリクエストに同じセッショントークンを使用できます。指定した期間が期限切れになった後、将来のリクエストに使用する新しいセッショントークンを作成する必要があります。

> **Important**
>
> Amazon Linux 2023 から起動された Lightsail インスタンスでは、デフォルトで IMDSv2 が設定されます。

次の例では、LinuxPowerShell シェルスクリプトと IMDSv2 を使用して、最上位インスタンスメタデータアイテムを取得しています。これらの例では、以下のことを行います。

- PUT リクエストを使用して、6 時間 (21,600 秒) のセッショントークンを作成する
- セッショントークンヘッダーを TOKEN (Linux の場合) または token (Windows の場合) という名前の変数に保管する
- トークンを使用して最上位メタデータアイテムをリクエストする

次のコマンドを使用してインストールして起動します。

- Linux の場合:
  - 最初に、次のコマンドを使用してトークンを生成します。

```
[ec2-user ~]$ TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-
aws-ec2-metadata-token-ttl-seconds: 21600"`
```

  - その後、次のコマンドを使用して、トークンを使用して上位レベルのメタデータアイテムを生成します。

```
[ec2-user ~]$ curl -H "X-aws-ec2-metadata-token: $TOKEN" -v http://169.254.169.254/
latest/meta-data/
```

- Windows の場合:
  - 最初に、次のコマンドを使用してトークンを生成します。

```
PS C:\> [string]$token = Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token-
ttl-seconds" = "21600"} -Method PUT -Uri http://169.254.169.254/latest/api/token
```

  - その後、次のコマンドを使用して、トークンを使用して上位レベルのメタデータアイテムを生成します。

```
PS C:\> Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token" = $token} -Method
 GET -Uri http://169.254.169.254/latest/meta-data/
```

トークンを作成した後、期限切れになるまで再使用することができます。次の例では、各コマンドはインスタンスの起動に使用されるブループリント (Amazon マシンイメージ (AMI)) の ID を取得します。前の例のトークンは再利用されます。$TOKEN (Linux) または $token (Windows) に保管されます。

- Linux の場合:

```
[ec2-user ~]$ curl -H "X-aws-ec2-metadata-token: $TOKEN" -v http://169.254.169.254/
latest/meta-data/ami-id
```

- Windows の場合:

```
PS C:\> Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token" = $token} `
-Method GET -uri http://169.254.169.254/latest/meta-data/ami-id
```

IMDSv2 を使用してインスタンスメタデータをリクエストする際は、リクエストに次の項目が含まれている必要があります。

- **PUT** リクエスト – PUT リクエストを使用して、インスタンスメタデータサービスに対してセッションを開始します。PUT リクエストは、インスタンスメタデータサービスに対する後続の GET リクエストに含まれている必要のあるトークンを返します。このトークンは、IMDSv2 を使用する際、メタデータにアクセスするのに必要です。
- トークン – トークンを、インスタンスメタデータサービスに対するすべての GET リクエストに含めます。トークンの使用が `required` に設定されている場合、有効なトークンがないリクエスト、または有効期限切れのトークンを持つリクエストで `401 - Unauthorized` HTTP エラーコードが発生します。トークン使用要件の変更に関する情報については、「AWS CLI コマンドリファレンス」の「インスタンスのメタデータのオプションの更新」を参照してください。
  - トークンはインスタンス固有のキーです。トークンは他のインスタンスで有効ではなく、生成されたインスタンスの外で使用しようとすると拒否されます。
  - PUT リクエストには、トークンの有効期限 (TTL) を秒単位で指定するヘッダーが含まれている必要があります。TTL は最大 6 時間 (21,600 秒) まで指定できます。トークンは論理的セッションを表します。TTL は、トークンが有効な時間の長さ、つまりセッションの期間を指定します。
  - トークンの期限が切れた後、インスタンスメタデータにアクセスし続けるためには、別の PUT リクエストを使用して新しいセッションを作成する必要があります。
  - 各リクエストについてトークンを再使用するか、あるいは新しいトークンを作成することを選択できます。少数のリクエストでは、インスタンスメタデータサービスにアクセスする必要があるたびに、トークンを生成してすぐに使用するほうが簡単かもしれません。ただし、効率を重視するなら、インスタンスメタデータをリクエストする必要があるたびに PUT リクエストを書くのではなく、トークン期間を長く指定して再使用することができます。それぞれが独自のセッションを表すトークンを同時に使用できる数については、実質的な制限はありません。ただし、IMDSv2 では、通常のインスタンスメタデータサービス接続とスロットリングの制限に

よって制約を受けます。詳細については、「Linux インスタンス向け Amazon Elastic Compute Cloud ユーザーガイド」の「クエリスロットリング」を参照してください。

HTTP GET および HEAD メソッドは IMDSv2 インスタンスメタデータリクエストで許可されています。PUT リクエストは、X-Forwarded-For ヘッダーが含まれている場合、拒否されます。

デフォルトで、PUT リクエストに対するレスポンスには IP プロトコルレベルで 1 のレスポンスホップリミット (有効期限) があります。より大きなホップリミットが必要な場合は、update-instance-metadata-options コマンドを使用して調整できます。例えば、インスタンスで実行されているコンテナサービスとの下位互換性のためにホップリミットを拡大する必要があるかもしれません。詳細については、「AWS CLI コマンドリファレンス」の「インスタンスのメタデータのオプションの更新」を参照してください。

インスタンスメタデータサービスバージョン 2 の使用への移行

インスタンスメタデータサービスバージョン 2 (IMDSv2) の使用は任意です。インスタンスメタデータサービスバージョン 1 (IMDSv1) は、終了の期限なく引き続きサポートされます。IMDSv2 の使用に移行する場合、次のツールと移行パスを使用することが推奨されます。

IMDSv2 への移行に役立つツール

お使いのソフトウェアで IMDSv1 が使用されている場合、次のツールを使用して IMDSv2 を使用するようソフトウェアを再構成することができます。

- AWS ソフトウェア: 最新バージョンの AWS SDK および AWS CLI がIMDSv2 をサポートしています。IMDSv2 を使用するには、インスタンスの AWS SDK および AWS CLI のバージョンが最新であることを確認する必要があります。AWS CLI の更新の詳細については、「AWS Command Line Interface ユーザーガイド」の、「AWS CLI のインストール、更新、およびアンインストール」を参照してください。すべての Amazon Linux 2 ソフトウェアパッケージが IMDSv2 をサポートしています。
- インスタンスのメトリクス: IMDSv2 はトークンベースのセッションを使用しますが、IMDSv1 は使用しません。MetadataNoToken インスタンスのメトリクスは、IMDSv1 を使用しているインスタンスメタデータサービスへの呼び出しの数を追跡します。このメトリクスをゼロまでトラッキングすることにより、すべてのソフトウェアが IMDSv2 を使用するようアップグレードされたかどうか、およびいつアップデートが行われたかを測定できます。詳細については、「Amazon Lightsail でインスタンスのメトリクスを表示する」を参照してください。
- Lightsail API オペレーションと AWS CLI コマンドへの更新: 既存のインスタンスについては、インスタンスのメタデータのオプションの更新 の AWS CLI コマンド (または

UpdateInstanceMetadataOptions API オペレーション) を使用して IMDSv2 の使用を要求できます。コマンドの例を次に示します。*InstanceName* はインスタンスの名前に、*RegionName* はインスタンスが存在する AWS リージョン に置き換えてください。

```
aws lightsail update-instance-metadata-options --region RegionName --instance-
name InstanceName --http-tokens required
```

IMDSv2 アクセスを必要とする推奨パス

前述のツールを使用し、IMDSv2 への移行にこのパスに従うことを推奨します。

ステップ 1: 開始時

AWS SDK、AWS CLI、およびインスタンスでロール資格情報を使用するソフトウェアを、IMDSv2 対応バージョンに更新します。AWS CLI の更新に関する情報については、AWS Command Line Interface ユーザーガイドの「AWS CLI の最新バージョンへのアップグレード」を参照してください。

次に、IMDSv2 リクエストを使用してインスタンスメタデータに直接アクセスする (つまり、AWS SDK を使用しない) ソフトウェアを変更します。

ステップ 2: 移行中

MetadataNoToken のインスタンスメトリクスを使用して、移行の進行状況を追跡します。このメトリクスは、インスタンスで IMDSv1 を使用しているインスタンスメタデータサービスへの呼び出しの数を示します。詳細については、「インスタンスのメトリクスを表示する」を参照してください。

ステップ 3: すべてのインスタンスですべての準備が完了した時点

インスタンスのメトリクス MetadataNoToken が IMDSv1 の使用ゼロを記録した時点で、すべてのインスタンスにおいてすべての準備が完了します。この段階で、インスタンスのメタデータのオプションの更新 コマンドから IMDSv2 の使用をリクエストできます。実行中のインスタンスでこれらの変更を行うことができます。インスタンスを再起動する必要はありません。

既存のインスタンスのインスタンスメタデータオプションの更新は、Lightsail API または AWS CLI を介してのみ使用できます。現在のところ、Lightsail コンソールでは使用できません。詳細については、「update-instance-metadata-options」を参照してください。

IMDS 関連の追加のドキュメント

次の IMDS ドキュメントは、「Linux インスタンス用 Amazon Elastic Compute Cloud ユーザーガイド」と「Windows インスタンス用 Amazon Elastic Compute クラウドユーザーガイド」で利用できます。

> **Note**
>
> Amazon EC2 では、インスタンスのブループリントは Amazon マシンイメージ (AMIs) と呼ばれます。

- Linux インスタンスの場合:
  - インスタンスメタデータオプションの設定
  - インスタンスメタデータの取得
  - インスタンスユーザーデータの使用
  - 動的データの取得
  - インスタンスメタデータのカテゴリ
  - 例: AMI 作成インデックス値
  - インスタンスアイデンティティドキュメント
- Windows インスタンスの場合:
  - インスタンスメタデータオプションの設定
  - インスタンスメタデータの取得
  - インスタンスユーザーデータの使用
  - 動的データの取得
  - インスタンスメタデータのカテゴリ
  - 例: AMI 作成インデックス値
  - インスタンスアイデンティティドキュメント

Amazon Lightsail におけるブロックストレージディスク

システムディスクでは、ワークロードの実行に必要な安定していて低レイテンシーのパフォーマンスが提供されます。Lightsail ディスクを使用すると、使用量の拡張または縮小を分単位で行うことができます。プロビジョニングしているサイズに合わせて、低料金でご利用いただけます。

Linux/Unix ベースまたは Windows Server ベースのインスタンスでは、最大 80 GB のシステムディスクを選択できます。「Lightsail で Linux ベースのインスタンスの使用を開始する」または「Windows Server ベースのインスタンスの使用を開始する」を参照してください。

追加のブロックストレージディスクを作成することで、仮想プライベートサーバーにストレージをさらに追加することもできます。「ブロックストレージディスクの作成と Linux ベースのインスタンスへのアタッチ」または「ブロックストレージディスクの作成と Windows Server インスタンスへのアタッチ」を参照してください。

ブロックストレージディスク

ブロックストレージは、データを「ブロック」として管理するストレージアーキテクチャです。各ストレージブロック (Lightsail では「ディスク」とも呼ばれます) は、サーバーにアタッチ可能な個々のハードディスクと同様に機能します。通常、特定のデータをコアサービスから分離し、インスタンスやブートストレージディスクで障害や他の問題が発生した場合にアプリケーションデータを保護する必要があるアプリケーションまたはソフトウェアに追加のブロックストレージを使用できます。

Lightsail では、ブロックストレージにソリッドステートドライブ (SSD) が使用されます。このタイプのブロックストレージは、リーズナブルな料金と良好なパフォーマンスのバランスが取れています。Lightsail で実行される大半のワークロードをサポートすることを目的としています。Lightsail の追加のブロックストレージディスクにより、保存されたデータに頻繁にアクセスするアプリケーションやソフトウェアに必要な一貫したパフォーマンスと低レイテンシーが実現します。

> **Note**
>
> 一定の IOPS パフォーマンスまたはディスクあたりの高いスループットを必要とするアプリケーションや、MongoDB、Cassandra などの大規模データベースを実行するお客様は、Lightsail の代わりに GP2 やプロビジョンド IOPS SSD ストレージを搭載する Amazon EC2 を使用することをお勧めします。

「Amazon EBS ボリューム」の詳細は、「Amazon EC2 ユーザーガイド」でご覧いただけます。

ディスククォータ

- リージョンあたり 20,000 GB。
- ディスクあたり最大 16 TB、またはディスクあたり最小 8 GB。
- インスタンスあたり最大で 15 個までのアタッチされたディスクおよび 1 個のブートボリュームディスクを保持できます。

Lightsail ブロックストレージディスクを作成して Linux ベースのインスタンスにアタッチする

Lightsail インスタンス用に追加のブロックストレージディスクを作成してアタッチすることができます。追加ディスクを作成したら、Linux/Unix ベースの Lightsail インスタンスに接続してディスクをフォーマットおよびマウントする必要があります。

このトピックでは、Lightsail を使用して新しいディスクを作成し、アタッチする方法について説明します。さらに、アタッチされたディスクをフォーマットしてマウントできるように、SSH を使用して Linux/Unix ベースのインスタンスに接続する方法についても説明します。

Windows Server ベースのインスタンスを使用している場合は、代わりに「ブロックストレージディスクを作成して Windows Server インスタンスにアタッチする」を参照してください。

ステップ 1: 新しいディスクを作成してインスタンスにアタッチする

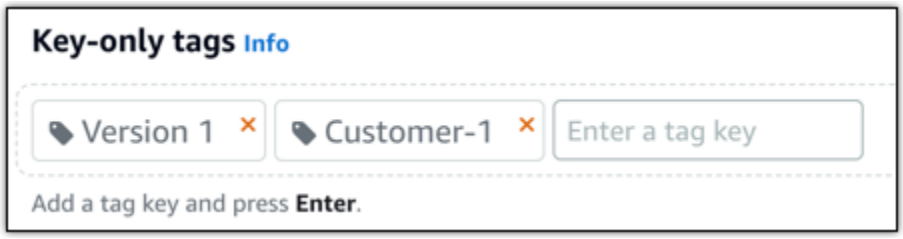
1. Lightsail のホームページで [ストレージ] を選択します。
2. [ディスクの作成] を選択します。
3. Lightsail インスタンスが配置されている AWS リージョン およびアベイラビリティーゾーンを選択します。
4. サイズを選択します。
5. ディスクの名前を入力します。

   リソース名:

   - Lightsail アカウントの各 AWS リージョン 内で一意である必要があります。
   - 2〜255 文字を使用する必要があります。
   - 先頭と末尾は英数字または数字を使用する必要があります。
   - 英数字、数字、ピリオド、ダッシュ、アンダースコアを使用することができます。

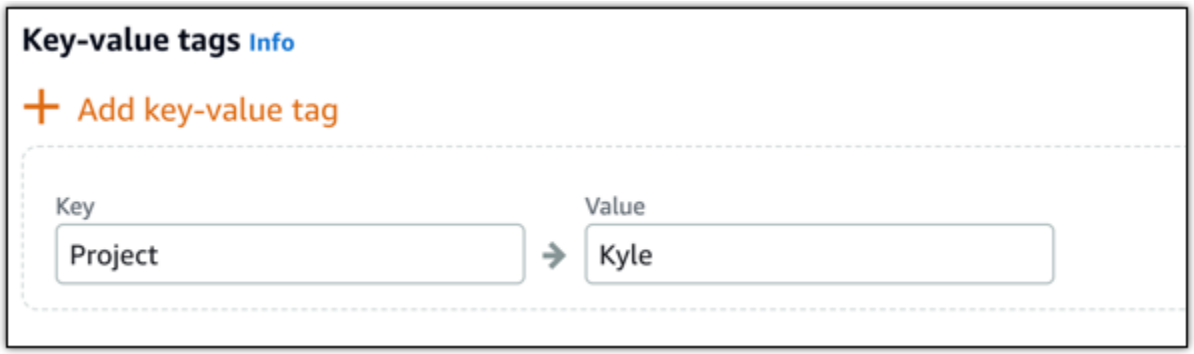
6. 以下のいずれかのオプションを選択して、ディスクにタグを追加します。

   - [Add key-only tags (キーのみのタグを追加)] または [Edit key-only tags (キーのみのタグを編集)] (タグが追加済みの場合)を追加。タグキーのテキストボックスに新しいタグを入力し、Enter キーを押します。タグの入力を完了したら、[保存] を選択してタグを追加し、追加しない場合は、[キャンセル] を選択します。

   - [key-value タグの作成] から [キー] テキストボックスにキーを入力し、[値] テキストボックスに値を入力します。タグの入力を完了したら、[保存] を選択し、追加しない場合は、[キャンセル] を選択します。

     キーバリューのタグは、保存する際に一つずつ追加することができます。さらに key-value タグを追加するには、以上のステップを繰り返します。

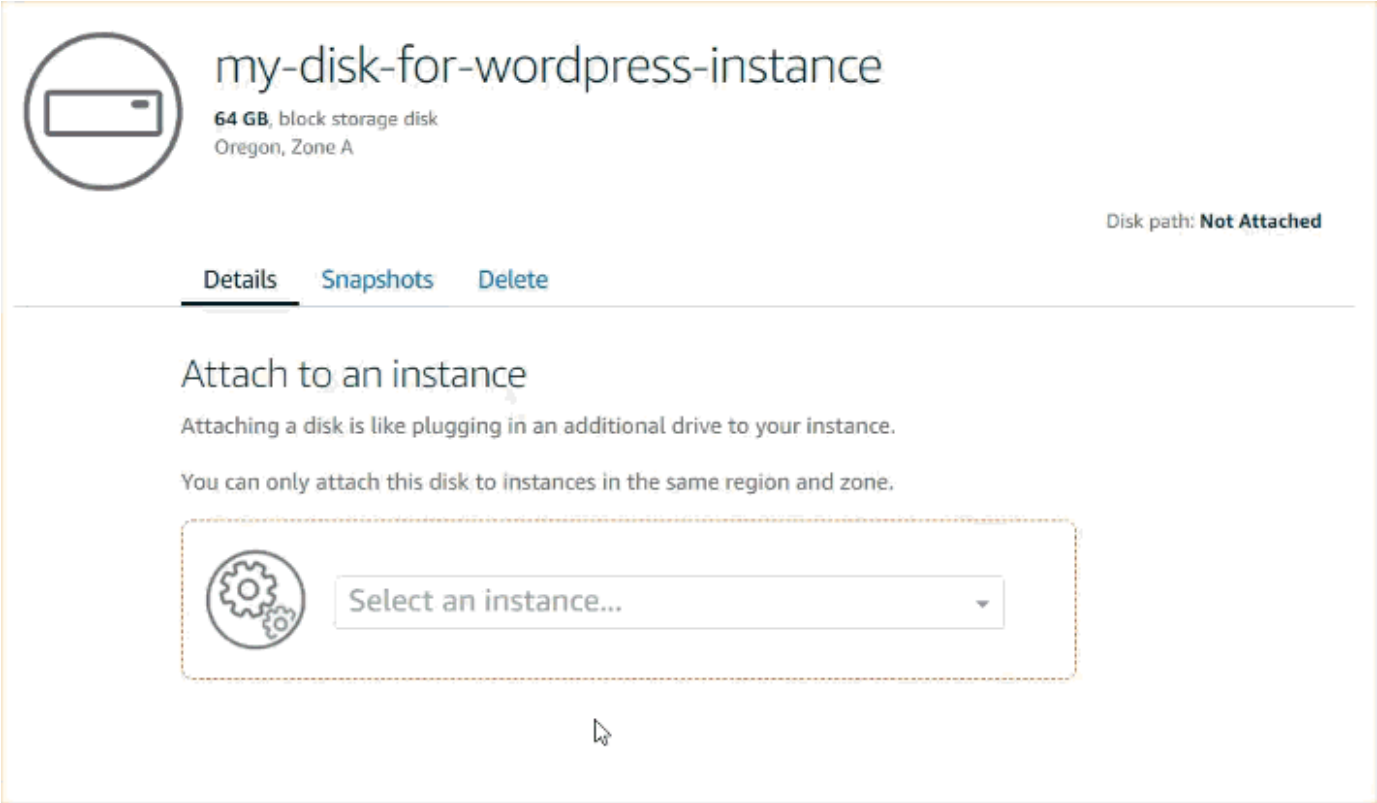
     

   > **Note**
   >
   > 「キーのみ」のタグと「キーバリュー」のタグの詳細については、「タグ」を参照してください。

7. [ディスクの作成] を選択します。

   数秒後、ディスクが作成され、新しいディスク管理ページが表示されます。

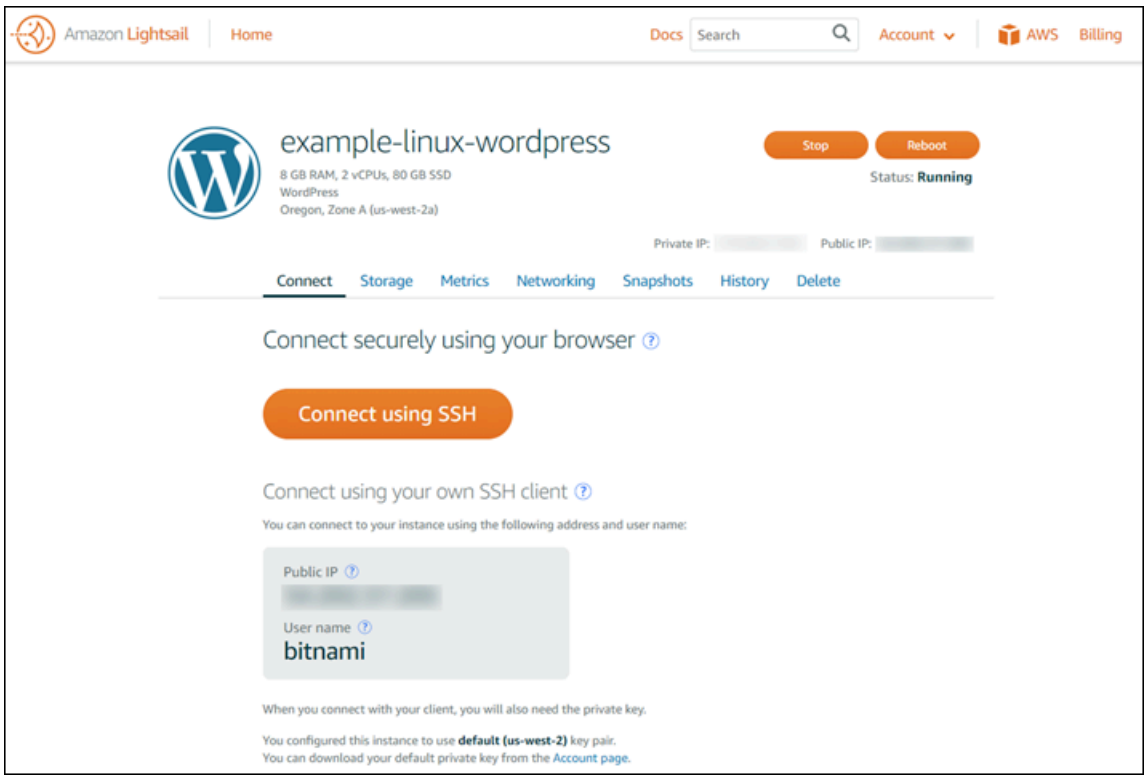
8. リストからインスタンスを選択し、[アタッチ] を選択して、新しいディスクをインスタンスにアタッチします。



ステップ 2: インスタンスに接続し、ディスクをフォーマットしてマウントする

1. ディスクを作成してアタッチしたら、Lightsail でインスタンス管理ページに戻ります。

   デフォルトでは、[接続] タブが表示されます。



2. [SSH を使用して接続] を選択してインスタンスに接続します。
3. 次の内容を入力します。

```
lsblk
```

次のような出力が表示されます。

```
NAME    MAJ:MIN RM SIZE RO TYPE MOUNTPOINT
xvda    202:0    0  80G  0 disk
##xvda1 202:1    0  80G  0 part /
xvdf    202:80   0  64G  0 disk
```

`lsblk` の出力は、ディスクパスからプレフィックス `/dev/` を削除します。

4. ディスクにファイルシステムを作成する必要があるかどうかを確認します。新しいディスクは未加工のブロックデバイスであるため、マウントして使用する前に、ボリュームにファイルシステムを作成する必要があります。スナップショットから復元されたディスクには、多くの場合既にファイルシステムがあります。既存のファイルシステムの上に新しいファイルシステムを作成した場合、データが上書きされます。特殊な情報 (ファイルシステムの種類など) を一覧表示するには、次のコマンドを使用します。

```
sudo file -s /dev/xvdf
```

新しいディスクでは、次のような出力が表示されます。

```
/dev/xvdf: data
```

次のような出力が表示される場合、ディスクに既にファイルシステムがあることを意味します。

```
/dev/xvda1: Linux rev 1.0 ext4 filesystem data, UUID=1701d228-e1bd-4094-
a14c-12345EXAMPLE (needs journal recovery) (extents) (large files) (huge files)
```

5. 次のコマンドを使用して、ディスクに ext4 ファイルシステムを作成します。*device_name* をデバイス名 (例: /dev/xvdf) に置き換えます。アプリケーションの要件またはオペレーティングシステムの制限に応じて、ext3、XFS など、異なるファイルシステムの種類を選択できます。

> **Important**
>
> この手順では、空のディスクをマウントすることを前提としています。既にデータが含まれるディスク (スナップショットから復元したディスクなど) をマウントする場合は、ディスクのマウント前に mkfs を使用しないでください。代わりに、この手順のステップ 6 に進んでマウントポイントを作成します。ステップ 1 を実行した場合、ディスクがフォーマットされ、既存のデータが削除されます。

```
sudo mkfs -t ext4 device_name
```

次のような出力が表示されます。

```
mke2fs 1.42.9 (4-Feb-2014)
Filesystem label=
OS type: Linux
Block size=4096 (log=2)
Fragment size=4096 (log=2)
Stride=0 blocks, Stripe width=0 blocks
4194304 inodes, 16777216 blocks
838860 blocks (5.00%) reserved for the super user
First data block=0
```

```
Maximum filesystem blocks=4294967296
512 block groups
32768 blocks per group, 32768 fragments per group
8192 inodes per group
Superblock backups stored on blocks:
32768, 98304, 163840, 229376, 294912, 819200, 884736, 1605632, 2654208,
4096000, 7962624, 11239424

Allocating group tables: done
Writing inode tables: done
Creating journal (32768 blocks): done
Writing superblocks and filesystem accounting information: done
```

6. 次のコマンドを使用して、ディスクのマウントポイントディレクトリを作成します。マウントポイントとは、ディスクをマウントした後、ファイルシステムツリー内でディスクが配置され、ファイルの読み書きが実行される場所です。*mount_point* を場所 (例: /data) に置き換えます。

```
sudo mkdir mount_point
```

7. ディスクでファイルシステムが作成されたことを確認するには、次のコマンドを入力します。

```
sudo file -s /dev/xvdf
```

/dev/xvdf: data の代わりに、以下のような出力結果が表示されるはずです。

```
/dev/xvdf: Linux rev 1.0 ext4 filesystem data, UUID=0ee83fdf-e370-442e-
ae38-12345EXAMPLE (extents) (large files) (huge files)
```

8. 最後に、次のコマンドを入力して、ディスクをマウントします。

```
sudo mount device_name mount_point
```

新しいディスクマウントのファイルのアクセス許可をプレビューして、ユーザーとアプリケーションがディスクに書き込みできることを確認します。ファイルのアクセス許可の詳細については、「Amazon EC2 ユーザーガイド」の「Amazon EBS ボリュームを使用できるようにする」を参照してください。

ステップ 3: インスタンスを再起動するたびにディスクをマウントする

通常は、Lightsail インスタンスを再起動するたびにこのディスクをマウントしてください。マウントしない場合、このステップは省略可能です。

1. システムブート時に常に、このディスクをマウントするには、/etc/fstab ファイルにデバイス用のエントリを追加します。

   /etc/fstab ファイルのバックアップコピーを作成すると、編集中に誤って破壊/削除してしまった場合にこのコピーを使用できます。

   ```
   sudo cp /etc/fstab /etc/fstab.orig
   ```

2. 任意のテキストエディタ (例: vim など) を使って /etc/fstab ファイルを開きます。

   変更を保存するには、ファイルを開く前に sudo と入力する必要があります。

3. 次のフォーマットを使って、ディスクのファイルの最後に新しい行を追加します。

   ```
   device_name  mount_point  file_system_type  fs_mntops  fs_freq  fs_passno
   ```

   たとえば、新しい行は以下のようになります。

   ```
   /dev/xvdf /data ext4 defaults,nofail 0 2
   ```

4. ファイルを保存し、テキストエディタを終了します。

Lightsail ブロックストレージディスクを作成して Windows Server インスタンスにアタッチする

ストレージ容量を追加する必要がある場合は、Amazon Lightsail でブロックストレージディスクを作成して Windows Server インスタンスにアタッチできます。ブロックストレージディスクの詳細については、「ブロックストレージディスク」を参照してください。

このガイドでは、Lightsail コンソールを使用して新しいブロックストレージディスクを作成し、Windows Server インスタンスにアタッチする方法について説明します。さらに、RDP を使用して Windows Server インスタンスに接続し、ディスクをオンラインにして初期化する方法についても説明します。

この手順は、Windows Server 2016 と Windows Server 2012 R2 で同じです。

> **Note**
>
> Linux または Unix インスタンスを使用している場合は、「ディスクを作成して Linux または Unix インスタンスにアタッチする」を参照してください。

ステップ 1: 新しいブロックストレージディスクを作成してインスタンスにアタッチする

Amazon Lightsail コンソールを使用して新しいブロックストレージディスクを作成し、インスタンスにアタッチします。

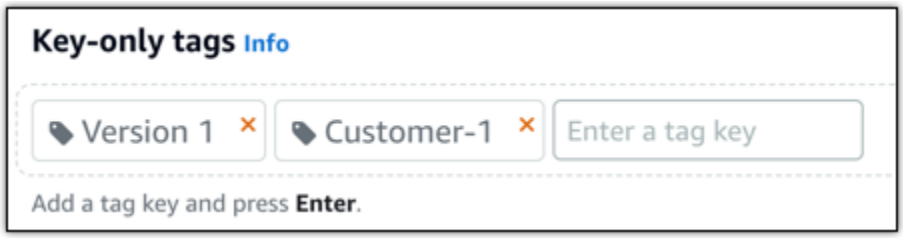
新しいブロックストレージディスクを作成してインスタンスにアタッチするには

1. Lightsail コンソールにサインインします。
2. [ストレージ] タブ、[ディスクの作成] の順に選択します。
3. Lightsail インスタンスが配置されている AWS リージョン およびアベイラビリティーゾーンを選択します。
4. ディスクサイズを選択します。
5. ストレージディスクの名前を入力します。

   リソース名:

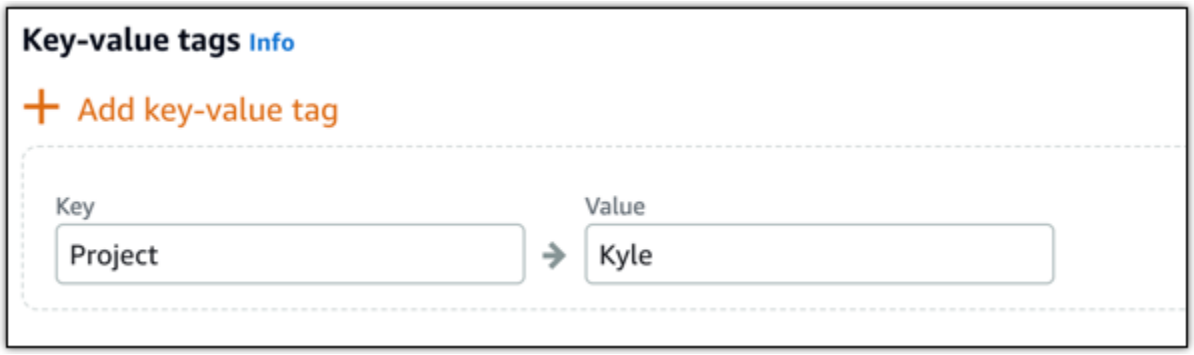
   - Lightsail アカウントの各 AWS リージョン 内で一意である必要があります。
   - 2〜255 文字を使用する必要があります。
   - 先頭と末尾は英数字または数字を使用する必要があります。
   - 英数字、数字、ピリオド、ダッシュ、アンダースコアを使用することができます。
6. 以下のいずれかのオプションを選択して、ディスクにタグを追加します。

   - [Add key-only tags (キーのみのタグを追加)] または [Edit key-only tags (キーのみのタグを編集)] (タグが追加済みの場合)を追加。タグキーのテキストボックスに新しいタグを入力し、Enter キーを押します。タグの入力を完了したら、[保存] を選択してタグを追加し、追加しない場合は、[キャンセル] を選択します。



- [key-value タグの作成] から [キー] テキストボックスにキーを入力し、[値] テキストボックスに値を入力します。タグの入力を完了したら、[保存] を選択し、追加しない場合は、[キャンセル] を選択します。

  キーバリューのタグは、保存する際に一つずつ追加することができます。さらに key-value タグを追加するには、以上のステップを繰り返します。



> **Note**
>
> 「キーのみ」のタグと「キーバリュー」のタグの詳細については、「タグ」を参照してください。

7. [ディスクの作成] を選択します。

   数秒後にディスクが作成され、新しいディスク管理ページでディスクの情報を確認できます。

8. リストからインスタンスを選択し、[アタッチ] を選択して、新しいディスクをインスタンスにアタッチします。

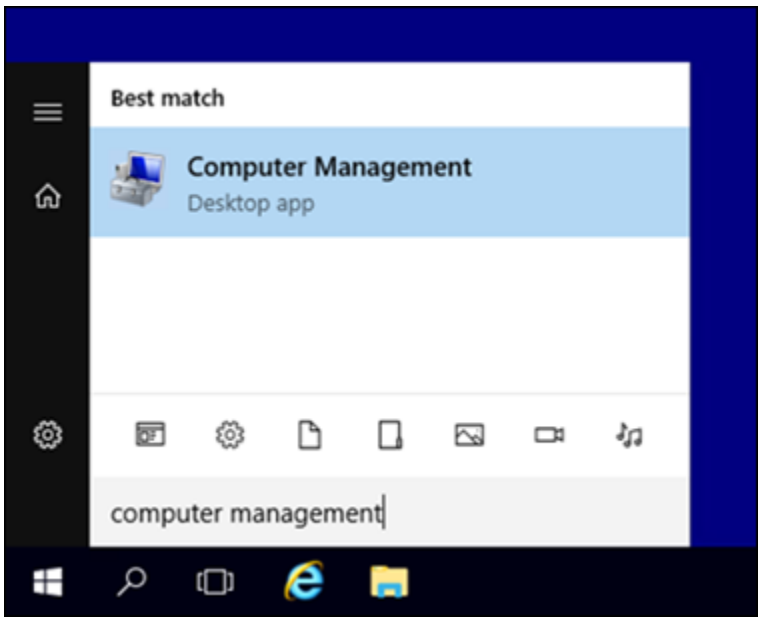
ブロックストレージディスクをオンラインにするには、このガイドの「ステップ 2: インスタンスに接続し、ブロックストレージディスクをオンラインにする」セクションに進みます。

ステップ 2: インスタンスに接続し、ブロックストレージディスクをオンラインにする

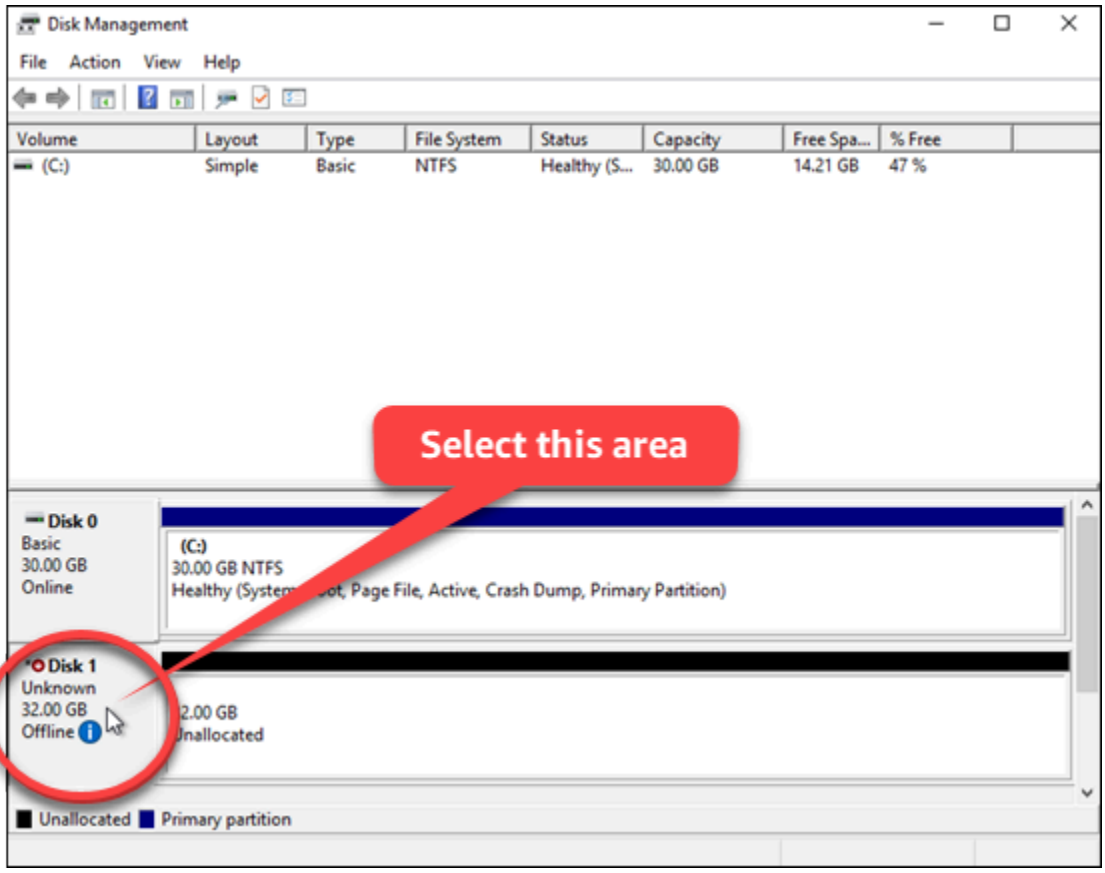
Windows Server インスタンスに接続し、ディスクの管理ユーティリティを使用して、前にアタッチしたブロックストレージディスクをオンラインにします。

インスタンスに接続してブロックストレージディスクをオンラインにするには

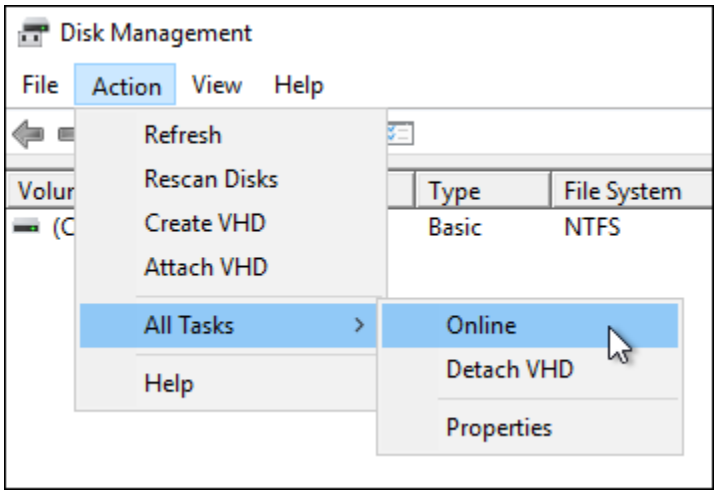
1. Lightsail コンソールのホームページに移動します。
2. このガイドで前に追加のストレージディスクをアタッチしたインスタンスの名前を選択します。
3. [接続] タブで、[RDP を使用して接続] を選択します。
4. Windows のスタートメニューで、コンピューターの管理を検索し、検索結果から [コンピューターの管理] を選択します。



5. [コンピューターの管理] の左側のペインで、[ディスクの管理] を選択します。
6. [ディスクの管理] ユーティリティの下部のペインで、[不明 / オフライン] というラベルが付いているディスクを選択します。これが、このガイドで前にインスタンスにアタッチしたブロックストレージディスクです。



7. ディスクを選択した状態で、[操作] メニューの [すべてのタスク] をポイントし、[オンライン] を選択します。



ブロックストレージディスクのステータスが [初期化されていません] に更新されるのがわかります。ブロックストレージディスクはまだオンラインになっていません。ブロックストレージディスクを初期化するには、このガイドの「ステップ 3: ブロックストレージディスクを初期化する」セクションに進みます。

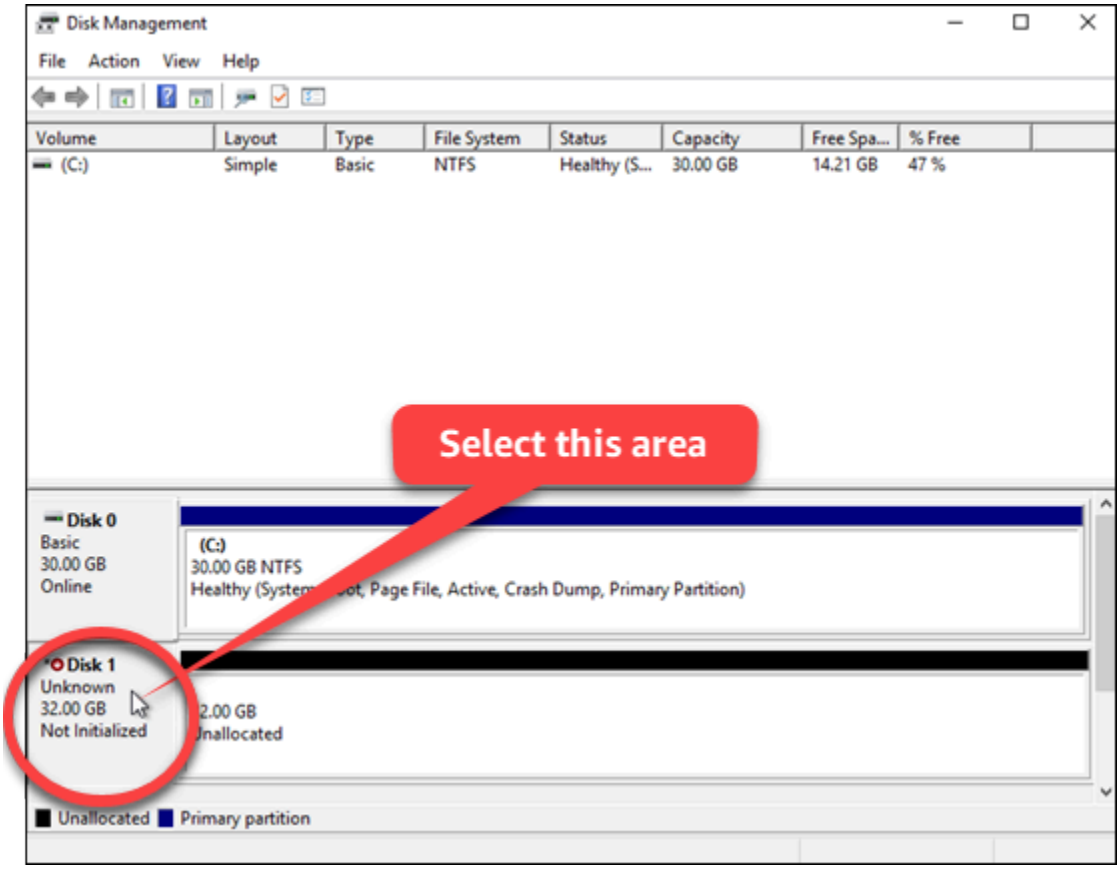
ステップ 3: ブロックストレージディスクを初期化する

フォーマットできるように、ブロックストレージディスクを初期化します。

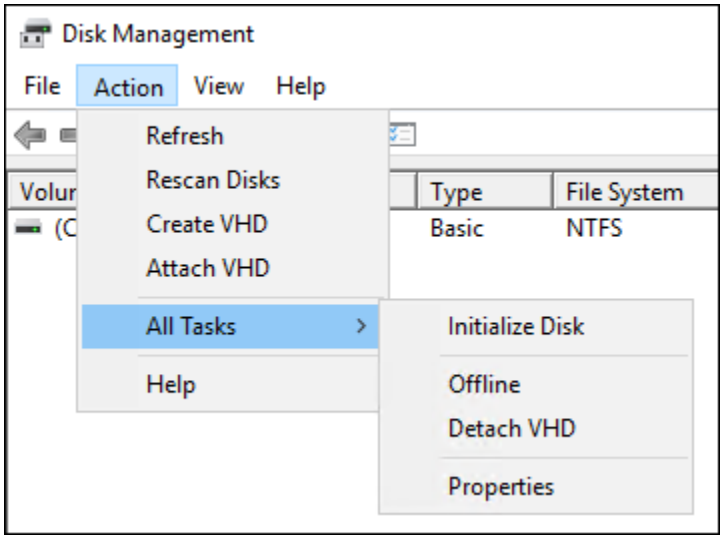
> **Important**
>
> すでにデータが含まれているディスク (スナップショットから作成したディスクなど) をマウントする場合は、ディスクを再フォーマットしないように注意してください。再フォーマットすると、既存のデータが削除されます。

ブロックストレージディスクを初期化するには

1. ディスクの管理ユーティリティの下部のペインで、[不明 / 初期化されていません] というラベルが付いているディスクを選択します。



2. ディスクを選択した状態で、[操作] メニューの [すべてのタスク] をポイントし、[ディスクの初期化] を選択します。



3. 新しいディスクのパーティションスタイルを選択し、[OK] を選択します。

> **Note**
>
> パーティションスタイルの詳細については、Microsoft の記事「About partition styles - GPT and MBR」を参照してください。

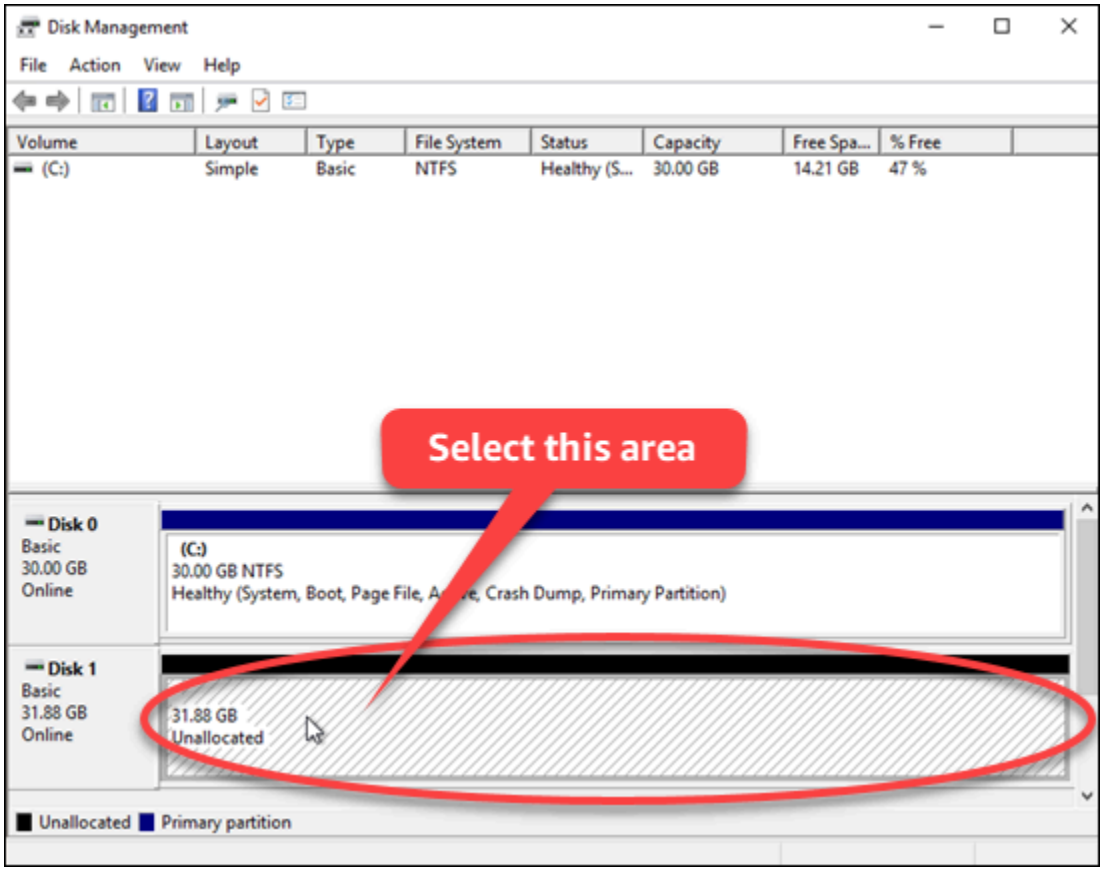
ブロックストレージディスクのステータスが [オンライン] に更新されるのがわかります。ファイルシステムでブロックストレージディスクをフォーマットにするには、このガイドの「ステップ 4: ディスクをファイルシステムでフォーマットする」セクションに進みます。

ステップ 4: ディスクをファイルシステムでフォーマットする

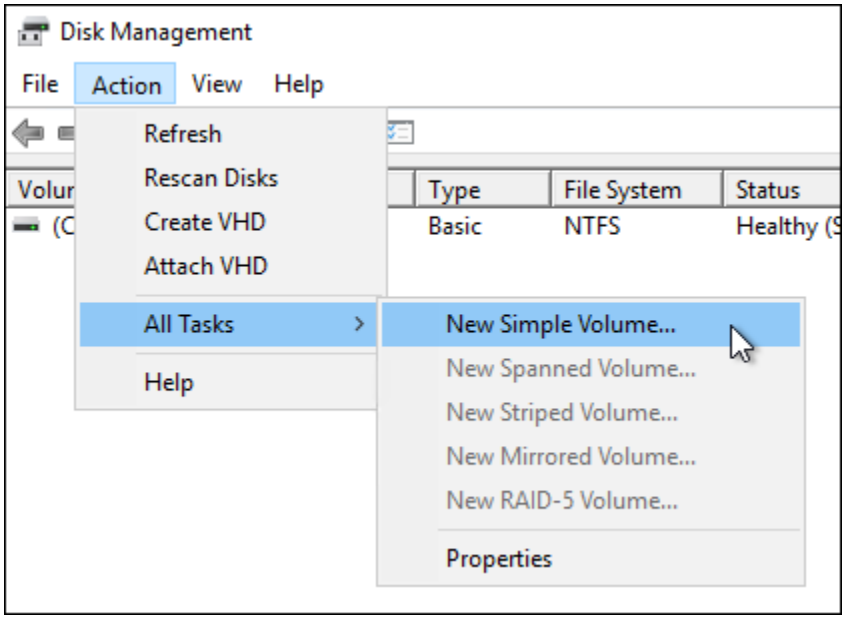
Windows Server の新しいシンプルボリュームウィザードを使用して、ドライブ文字を割り当て、ディスクをファイルシステムでフォーマットします。

ディスクをファイルシステムでフォーマットするには

1. ディスクの管理ユーティリティの下部のペインで、[未割り当て] というラベルが付いているブロックストレージディスクのパーティションを選択します。



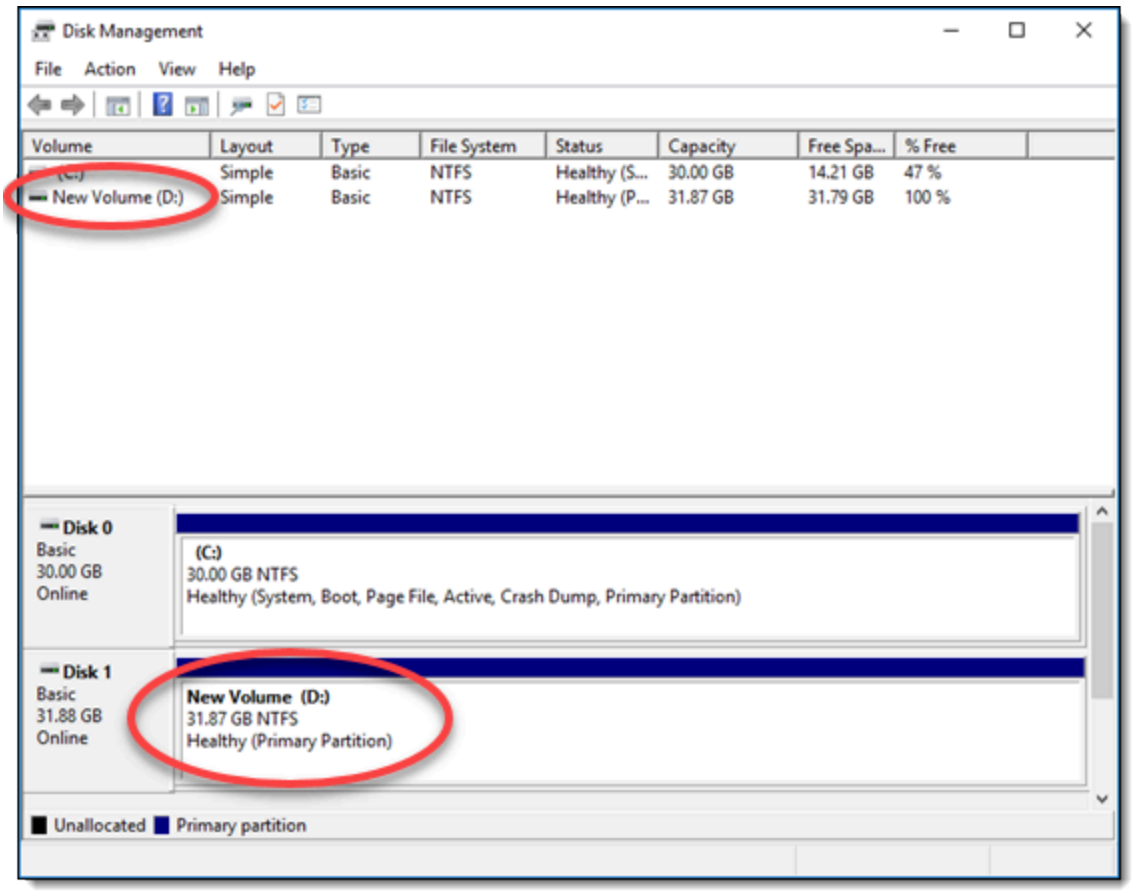
2. パーティションを選択した状態で、[アクション] メニューの [すべてのタスク] をポイントし、[新しいシンプルボリューム] を選択します。



3. 新しいシンプルボリュームウィザードの指示に従って、NTFS、FAT32、または ReFS ファイルシステムタイプを選択し、ディスクをフォーマットします。

> **Note**
>
> これらの各ファイルシステムの詳細については、Microsoft の記事「NTFS の概要」、「Resilient File System (ReFS) の概要」、および「FAT32 ファイル システムについて」を参照してください。

完了すると、ドライブ文字と次のメッセージがディスクの管理ユーティリティに表示されます。



Lightsail ブロックストレージディスクをデタッチおよび削除する

ブロックストレージディスクが不要になった場合、停止した Lightsail インスタンスからデタッチして削除できます。このトピックでは、データをバックアップしてディスクを安全に削除する方法について説明します。

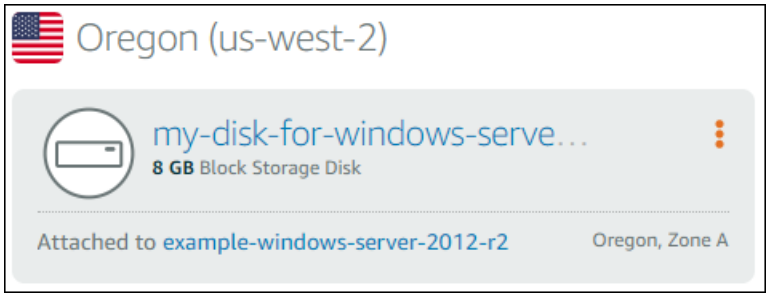
前提条件

- インスタンスの実行を停止します。これは、ディスクをデタッチして削除する前に実行する必要があります。インスタンスを停止する方法の詳細
- (オプション) ディスクのスナップショットを作成することをお勧めします。このようにして、状況が変わった場合もバックアップを利用できます。詳細については、「データベースのスナップショットを作成する」を参照してください。

ディスクをデタッチおよび削除する

Lightsail インスタンスを停止した後、ディスクを安全にデタッチして削除できます。

1. ホームページで [ストレージ] を選択します。
2. アタッチされたディスクの名前を選択して管理します。



3. ディスク管理ページで、[デタッチ] を選択します。

   数秒後、ディスクがデタッチされ、削除または再アタッチする準備ができます。
4. [削除] タブを選択します。
5. [ディスクの削除] を選択し、[はい、削除します] を選択して削除を確定します。

> **Important**
>
> これは永続的オペレーションで、取消すことはできません。削除するとディスク上のすべてのデータが失われます。

Amazon Lightsail のスナップショット

Amazon Lightsail でインスタンス、データベース、ブロックストレージディスクの point-in-time スナップショットを作成し、それらをベースラインとして使用して新しいリソースを作成したり、データをバックアップしたりできます。スナップショットには、リソースの復元に必要なすべてのデータ (スナップショットが作成された時点のデータ) が含まれます。スナップショットからリソースを作成して復元すると、その新しいリソースはスナップショットの作成に使用された元のリソースの正確なレプリカとして始まります。Lightsail アカウントのスナップショットには、手動スナップショット、自動スナップショット、コピーされたスナップショット、またはシステムディスクスナップショットのいずれであるかにかかわらず、スナップショットストレージ料金が請求されます。データの破損やディスクの障害が発生した場合は、作成したスナップショットからディスクを作成し、古いディスクを置き換えることができます。スナップショットを使用して新しいディスクをプロビジョニングし、新しいインスタンスの起動時にアタッチすることもできます。

目次

手動スナップショット

インスタンス、マネージドデータベース、ブロックストレージディスクのスナップショットをいつでも手動で作成します。手動スナップショットは、削除するまで無期限に保存されます。

手動スナップショットの作成の詳細については、以下のガイドを参照してください。

- Linux または Unix インスタンスのスナップショットを作成する
- Windows Server インスタンスのスナップショットを作成する
- データベースのスナップショットを作成する
- ブロックストレージディスクのスナップショットを作成する

自動スナップショット

Lightsail インスタンスまたはブロックストレージディスクで重要な情報をホストしている場合は、手動スナップショットを作成して頻繁にバックアップする必要があります。ただし、管理タスクを頻繁に実行する時間を見つけるのが難しい場合があります。その場合は、自動スナップショットを使用して、手動操作なしで Lightsail がユーザーに代わってインスタンスまたはブロックストレージディスクの日次バックアップを作成するようにします。毎日 7 つの最新の自動スナップショットが保存されたあと、最も古いものから最新のものに置き換えられます。

自動スナップショットの詳細については、以下のガイドを参照してください。

- インスタンスの自動スナップショットを有効または無効にする
- インスタンスまたはディスクの自動スナップショット時間の変更
- 自動スナップショットを削除する

> **Important**
>
> ソースリソースを削除すると、リソースに関連付けられたすべての自動スナップショットが削除されます。この動作は、ソースリソースを削除した後でも Lightsail アカウントに保持される手動スナップショットとは異なります。ソースリソースを削除するときに自動スナップショットを保持するには、「自動スナップショットの保持」を参照してください。

システムディスクのスナップショット

インスタンスが応答しなくなり、システムディスク上のファイルにアクセスする必要がある場合は、インスタンスルートボリュームをバックアップするためにそのボリュームのスナップショットを作成できます。次に、スナップショットから新しいブロックストレージディスクを作成し、別のインスタンスにアタッチすることで、システムディスク内のファイルにアクセスできます。詳細については、「インスタンスルートボリュームのスナップショットを作成する」を参照してください。

スナップショットからの新しいリソースの作成

スナップショットを使用して、元のリソースと同じプラン、またはそれ以上のプランを使用して、新しい Lightsail リソースを作成します。スナップショットに基づいてリソースを作成すると、新しいリソースは、スナップショットの作成に使用された元のリソースのレプリカとなります。スナップ

ショットは、小さい Lightsail プランを使用して新しいリソースを作成するために使用することはできません。

詳細については、以下のガイドを参照してください。

- スナップショットからのインスタンスの作成
- スナップショットからデータベースを作成する
- スナップショットから新しいブロックストレージディスクを作成する
- スナップショットからより大きなインスタンス、ブロックストレージディスク、またはデータベースを作成する

スナップショットをコピーする

インスタンスとブロックストレージディスクのスナップショットは、同じ Lightsail アカウント内の 1 つの Amazon Web Services (AWS) リージョンから別のリージョンにコピーできます。データベーススナップショットをリージョン間でコピーすることはできません。詳細については、「1 つの から別の AWS リージョン にスナップショットをコピーする」を参照してください。

のスナップショットを Amazon EC2 にエクスポートする

Lightsail は、 の使用を開始する最も簡単な方法です AWS。ただし、Lightsail には、Amazon EC2 や他の AWS サービスに存在しない制限があります。Lightsail インスタンスとブロックストレージディスクのスナップショットを Amazon EC2 にエクスポートして、利用可能な幅広いインスタンスタイプを活用し、 の幅広いサービスを使用します AWS。詳細については、「スナップショットを Amazon EC2 にエクスポートする」を参照してください。

> **Note**
>
> 現時点では、cPanel & WHM、Django、および Ghost インスタンスのスナップショットを Amazon EC2 にエクスポートすることはできません。

スナップショットを削除する

不要になった Lightsail スナップショットは、毎月のスナップショットストレージ料金が発生しないように削除します。詳細については、「スナップショットを削除する」を参照してください。

Lightsail ブロックストレージディスクのスナップショットを作成する

Lightsail では、ディスクスナップショットを追加のブロックストレージディスクのバックアップとして作成することができます。

ディスクのスナップショットを、新しいディスクまたはデータバックアップのためのベースラインとして使用できます。ディスクのスナップショットを定期的に作成する場合、スナップショットは差分になります。最後にスナップショットを作成した時点から、デバイス上で変更があったブロックだけが、新しいスナップショットに保存されます。スナップショットの保存は増分ベースで行われるものの、最新のスナップショットさえあればディスク全体を復元できるようにスナップショット削除プロセスは設計されています。

詳細については、「スナップショット」を参照してください。

1. Lightsail のホームページで [Storage] (ストレージ) タブを選択します。
2. スナップショットを作成するブロックストレージディスクの名前を選択します。
3. [スナップショット] タブを選択します。
4. このページの [手動スナップショット] セクションで、[スナップショットの作成] を選択し、スナップショットの名前を入力します。

   リソース名:

   - Lightsail アカウントの各 AWS リージョン 内で一意である必要があります。
   - 2〜255 文字を使用する必要があります。
   - 先頭と末尾は英数字または数字を使用する必要があります。
   - 英数字、数字、ピリオド、ダッシュ、アンダースコアを使用することができます。
5. [Create] (作成) を選択します。

   作成したスナップショットのステータスが [Snapshotting... (スナップショットの作成中)] と表示されることがあります。

   スナップショットが完了したら、スナップショットから別のディスクを作成することができます。

スナップショットから新しい Lightsail ブロックストレージディスクを作成する

ディスクスナップショットから新しいブロックストレージディスクを作成できます。完全に新規のディスクを作成する場合は、代わりに「追加のブロックストレージディスクを作成する (Linux/Unix)」または「ブロックストレージディスクを作成して Windows Server インスタンスにアタッチする」を参照してください。

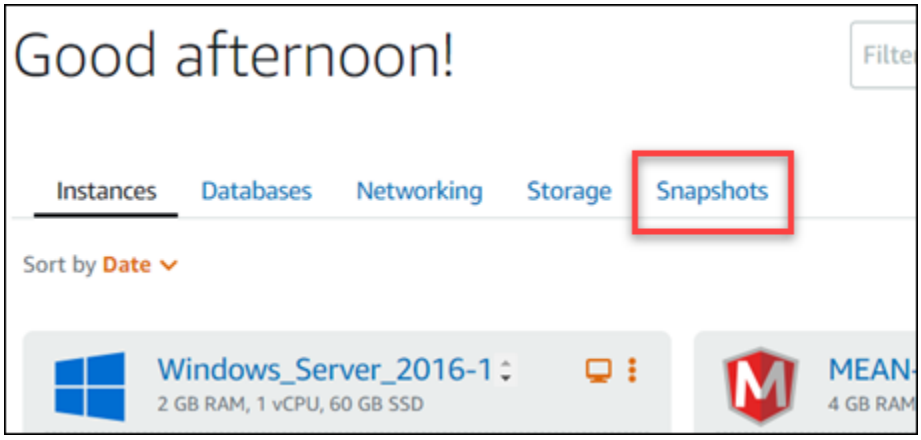
ブロックストレージディスクのスナップショットを、新しいディスクまたはデータバックアップのためのベースラインとして使用できます。ディスクのスナップショットを定期的に作成する場合、スナップショットは差分になります。最後にスナップショットを作成した時点から、ディスク上で変更があったブロックだけが、新しいスナップショットに保存されます。スナップショットの保存は増分ベースで行われるものの、最新のスナップショットさえあればディスク全体を復元できるようにスナップショット削除プロセスは設計されています。ブロックストレージディスクのスナップショットを作成するには、「ブロックストレージディスクのスナップショットを作成する」を参照してください。

ステップ 1: ディスクスナップショットを検索して新しいディスクの作成を選択する

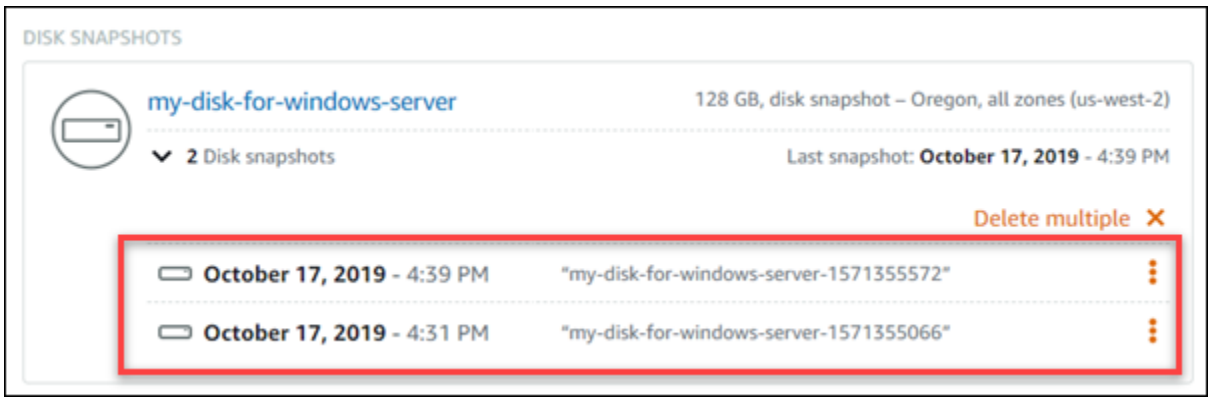
Lightsail の 2 つの場所 (Lightsail ホームページの [スナップショット] タブ、またはディスク管理ページの [スナップショット] タブ) のいずれかで、ディスクスナップショットから新しいインスタンスを作成できます。

Lightsail ホームページから

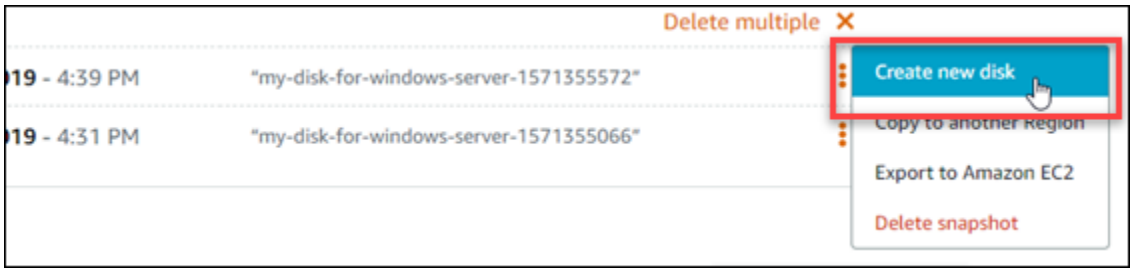
1. Lightsail のホームページで [スナップショット] タブを選択します。



2. ディスクの名前を見つけ、その下のノードを展開して、そのディスクの利用可能なすべてのスナップショットを表示します。

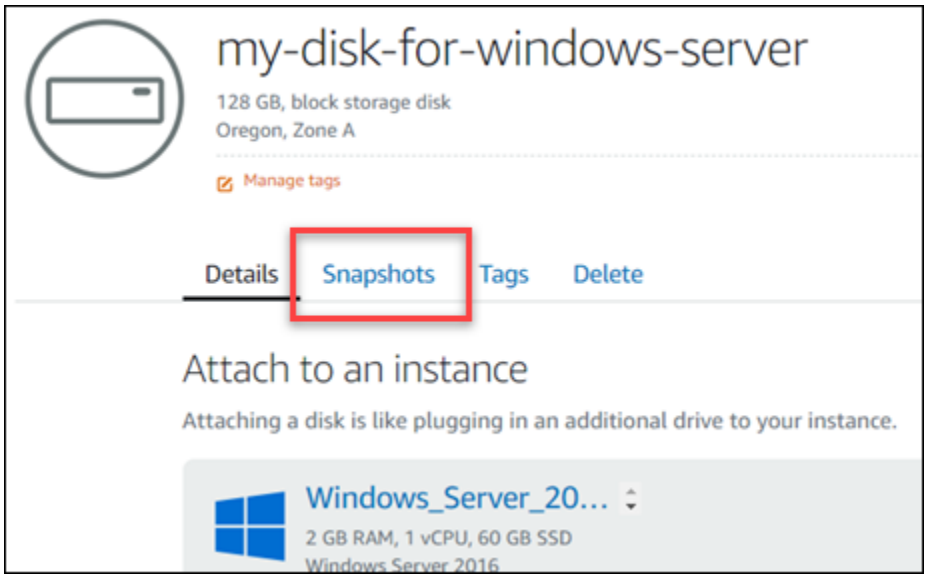


3. 新しいディスクを作成するスナップショットの横にあるアクションメニューアイコン (:) を選択し、[新しいディスクの作成] を選択します。

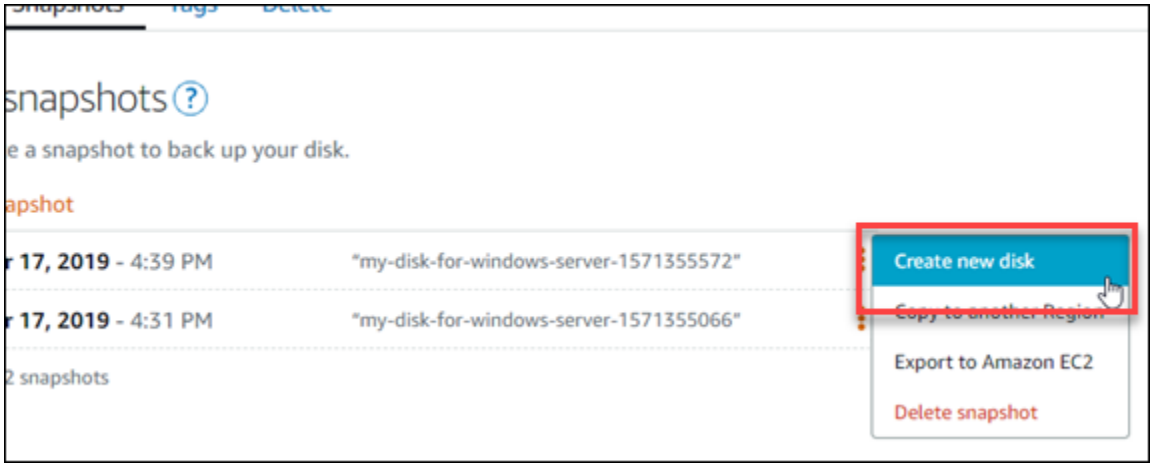


Lightsail のディスク管理ページから

1. Lightsail のホームページで [ストレージ] タブを選択します。
2. スナップショットを表示するディスクの名前を選択します。
3. [スナップショット] タブを選択します。



4. このページの [手動スナップショット] セクションで、新しいディスクを作成するスナップショットの横にあるアクションメニューアイコン (:) を選択してから、[新しいディスクの作成] を選択します。



ステップ 2: ディスクスナップショットから新しいディスクを作成する

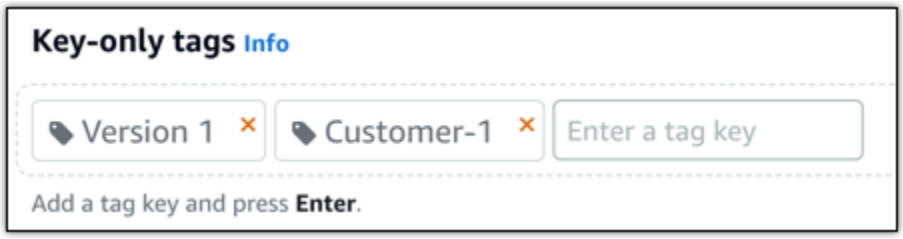
1. 新しいディスクのアベイラビリティーゾーンを選択するか、デフォルトのままにします (例: `us-east-2a`)。

   新しいディスクは、ソースディスクと同じ AWS リージョン に作成する必要があります。
2. 新しいディスクには、ソーススナップショット以上のサイズを選択してください。
3. ディスクの名前を入力します。

   リソース名:

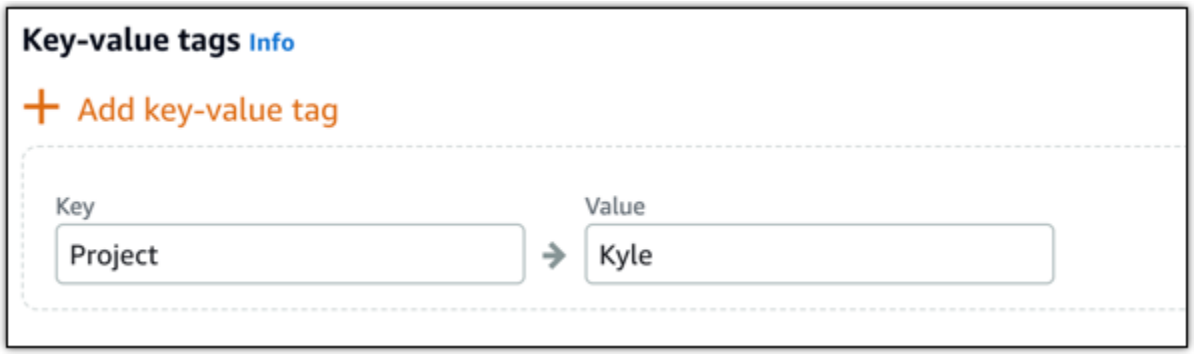
   - Lightsail アカウントの各 AWS リージョン 内で一意である必要があります。
   - 2〜255 文字を使用する必要があります。
   - 先頭と末尾は英数字または数字を使用する必要があります。
   - 英数字、数字、ピリオド、ダッシュ、アンダースコアを使用することができます。
4. 以下のいずれかのオプションを選択して、ディスクにタグを追加します。

   - [key-only タグの追加] または [key-only タグの編集] (タグが追加済みの場合)。タグキーのテキストボックスに新しいタグを入力し、Enter キーを押します。タグの入力を完了したら、[保存] を選択してタグを追加し、追加しない場合は、[キャンセル] を選択します。



- [key-value タグの作成] から [キー] テキストボックスにキーを入力し、[値] テキストボックスに値を入力します。タグの入力を完了したら、[保存] を選択し、追加しない場合は、[キャンセル] を選択します。

  キーバリューのタグは、保存する際に一つずつ追加することができます。さらに key-value タグを追加するには、以上のステップを繰り返します。



> **Note**
>
> 「キーのみ」のタグと「キーバリュー」のタグの詳細については、「タグ」を参照してください。

5. [ディスクの作成] を選択します。

Lightsail インスタンスのルートボリュームのスナップショットを作成する

システムディスクのスナップショットを作成して、Amazon Lightsail インスタンスのルートボリュームをバックアップします。そして、スナップショットから新しいブロックストレージディスクを作成し、別のインスタンスにアタッチすることによって、バックアップされたファイルにアクセスします。必要な場合は、以下のステップを実行します。

- 失敗したインスタンスのルートボリュームからデータを復旧します。

- ブロックストレージディスクに対して行うように、インスタンスのルートボリュームのバックアップを作成します。

AWS Command Line Interface (AWS CLI) を使用してインスタンスルートボリュームを作成します。スナップショットを作成した後、Lightsail コンソールを使用してスナップショットからブロックストレージディスクを作成します。次に、それを実行中のインスタンスにアタッチし、そのインスタンスからアクセスします。

目次

ステップ 1: 前提条件を満たす

まだ AWS CLI をインストールして設定していない場合は、インストールして設定します。詳細については、「Lightsail で使用するために AWS CLI を設定する」を参照してください。

ステップ 2: インスタンスのルートボリュームスナップショットを作成する

ターミナルまたはコマンドプロンプトウィンドウを開き、次のコマンドを入力して、ルートボリュームのスナップショットのインスタンスを作成します。

```
aws lightsail create-disk-snapshot --region AWSRegion --instance-name InstanceName --
disk-snapshot-name DiskSnapshotName
```

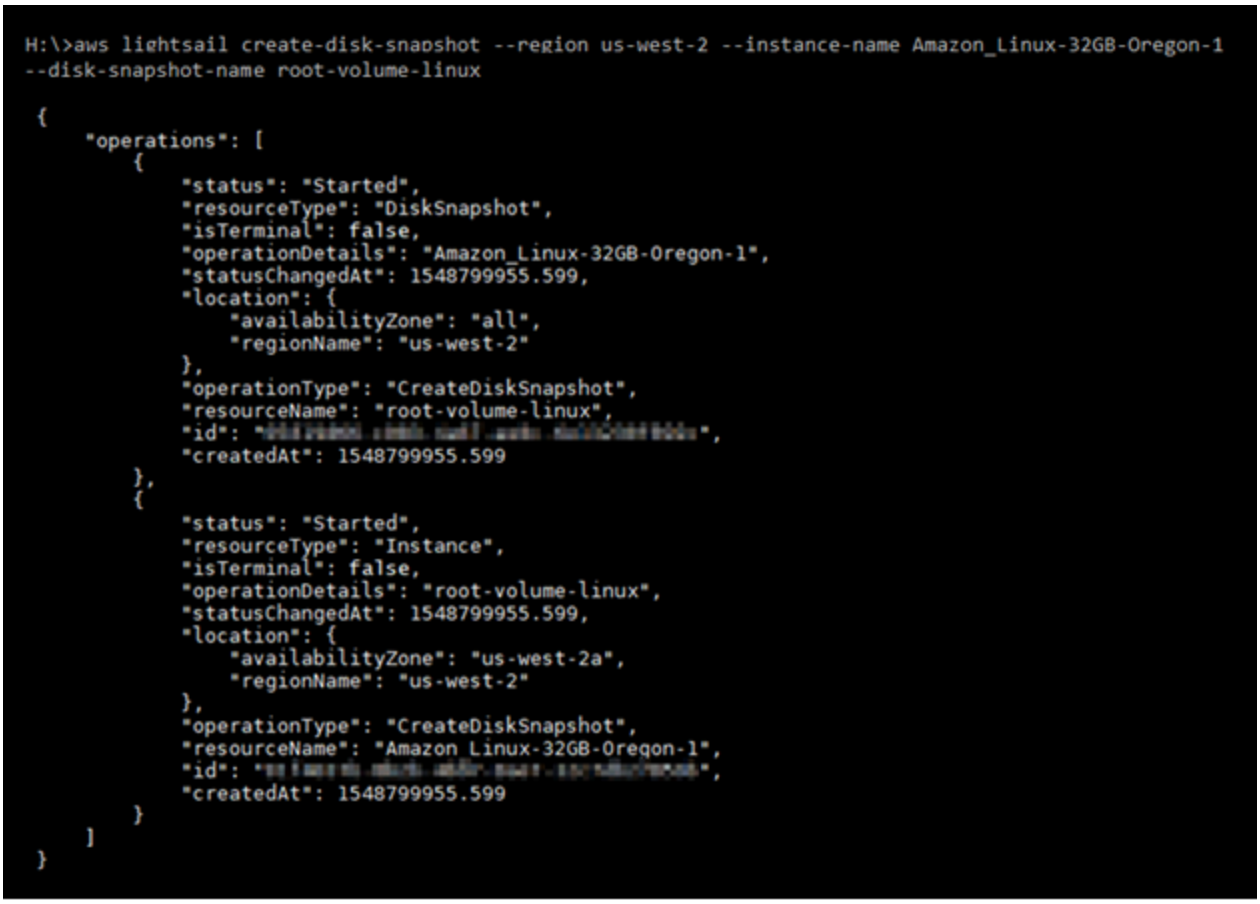
コマンドを、以下のように置き換えます。

- *AWSRegion* をインスタンスの AWS リージョン に置き換えます。
- *InstanceName* は、ルートボリュームをバックアップするインスタンスの名前に置き換えます。
- *DiskSnapshotName* は、作成される新しいディスクスナップショットの名前に置き換えます。

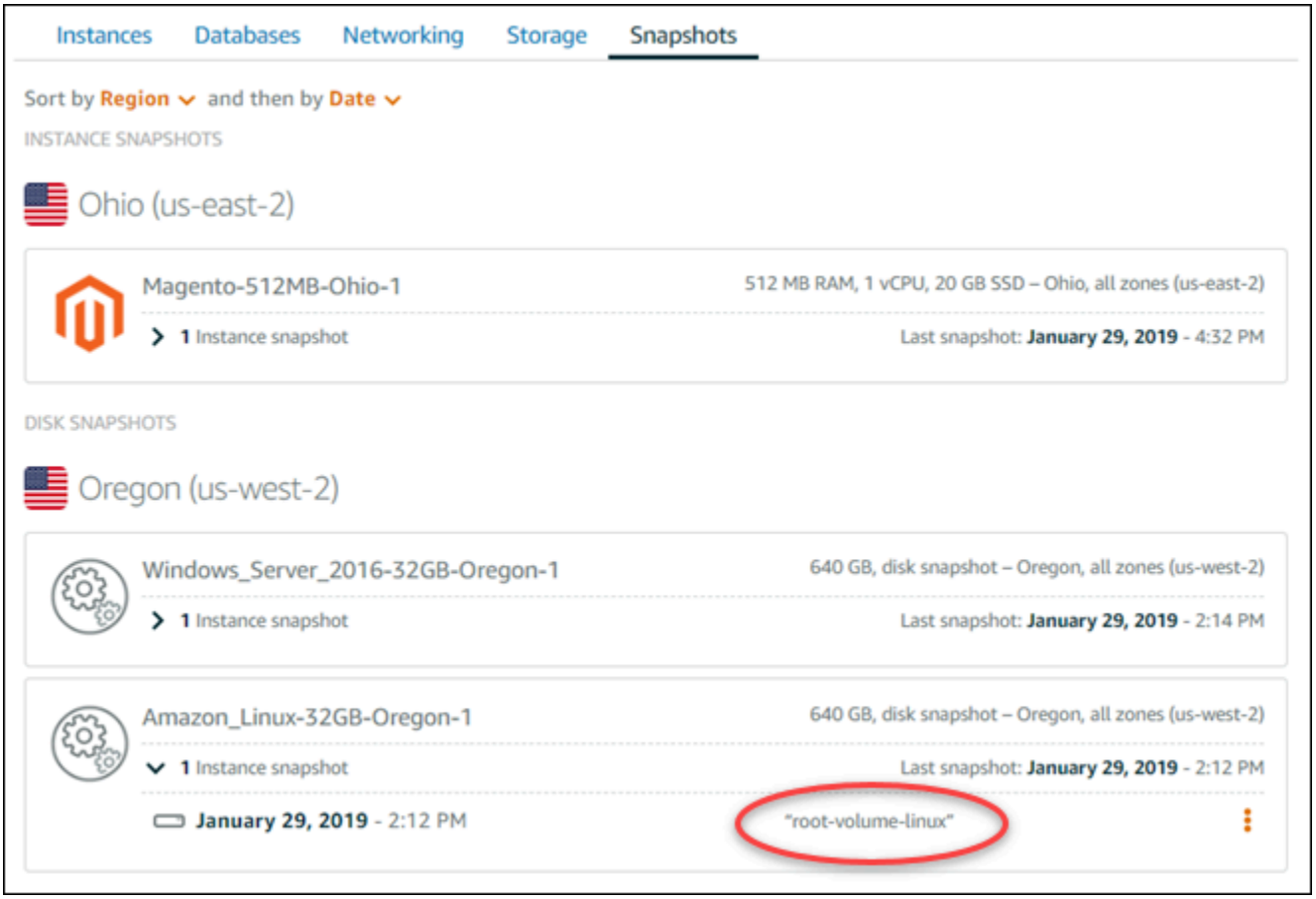
例:

```
aws lightsail create-disk-snapshot --region us-west-2 --instance-
name Amazon_Linux-32MB-Oregon-1 --disk-snapshot-name root-volume-linux
```

成功すると、以下のような結果が表示されます。



スナップショットが作成されるまで数分待ちます。作成後、Lightsail ホームページのスナップショットタブをクリックし、次の例に示すように、[ディスクスナップショット] セクションまでスクロールして確認することができます。



ステップ 3: スナップショットからブロックストレージディスクを作成し、インスタンスへアタッチする

インスタンスのルートボリュームのスナップショットから新しいブロックストレージディスクを作成し、そのコンテンツにアクセスするためには、別のインスタンスにアタッチします。失敗したインスタンスのルートボリュームからデータを復旧する必要がある場合は、以下を実行します。

> **Note**
>
> この新しいブロックストレージディスクは、ソースのスナップショットと同じ AWS リージョン に作成されます。別のリージョンにブロックストレージディスクを作成するためには、目的のリージョンにスナップショットをコピーし、コピーしたスナップショットから新しいディスクを作成します。詳細については、「1 つの AWS リージョン から別のリージョンにスナップショットをコピーする」を参照してください。

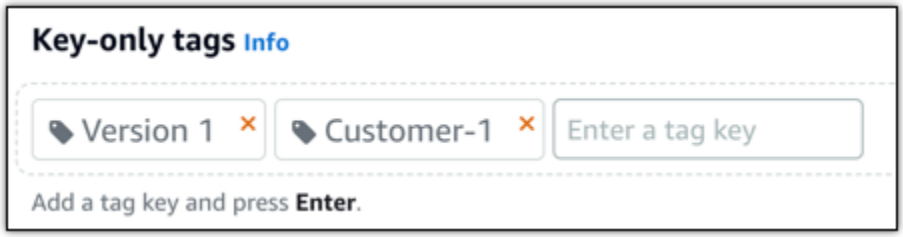
1. Lightsail コンソールにサインインします。

2. Lightsail のホームページで [スナップショット] タブを選択します。
3. 使用するルートボリュームディスクスナップショットの横に表示されるアクションメニューアイコン (⋮) を選択し、[Create new disk (新しいディスクの作成)] を選択します。
4. ディスクのアベイラビリティーゾーンを選択するか、デフォルトのままにします。
5. ソースディスクと同等、もしくはそれ以上のサイズのディスクを選択してください。
6. ディスクの名前を入力します。

   リソース名:

   - Lightsail アカウントの各 AWS リージョン 内で一意である必要があります。
   - 2〜255 文字を使用する必要があります。
   - 先頭と末尾は英数字または数字を使用する必要があります。
   - 英数字、数字、ピリオド、ダッシュ、アンダースコアを使用することができます。
7. 以下のいずれかのオプションを選択して、ディスクにタグを追加します。

   - [key-only タグの追加] または [key-only タグの編集] (タグが追加済みの場合)。タグキーのテキストボックスに新しいタグを入力し、Enter キーを押します。タグの入力を完了したら、[保存] を選択してタグを追加し、追加しない場合は、[キャンセル] を選択します。

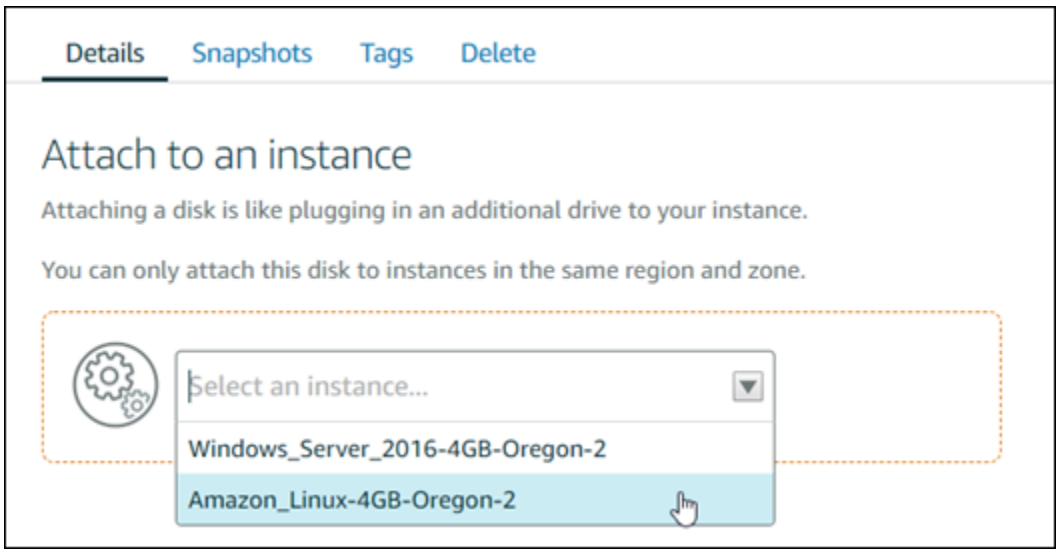
     

   - [key-value タグの作成] から [キー] テキストボックスにキーを入力し、[値] テキストボックスに値を入力します。タグの入力を完了したら、[保存] を選択し、追加しない場合は、[キャンセル] を選択します。

     キーバリューのタグは、保存する際に一つずつ追加することができます。さらに key-value タグを追加するには、以上のステップを繰り返します。

> **Note**
>
> 「キーのみ」のタグと「キーバリュー」のタグの詳細については、「タグ」を参照してください。

8. [ディスクの作成] を選択します。
9. ディスクが作成されたら、ディスクをアタッチするインスタンスを [Select an instance (インスタンスの選択)] ドロップダウンメニューで選択します。これは次の例で示されます。



10. [アタッチ] を選択して、選択したインスタンスにディスクをアタッチします。

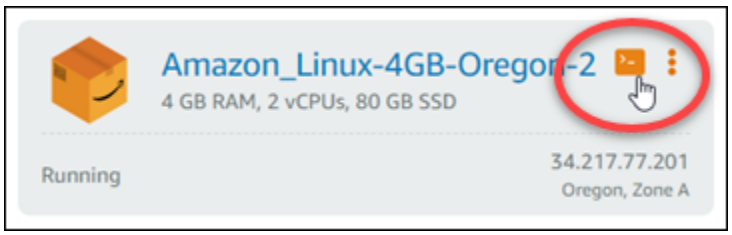
    ディスクがインスタンスにアタッチされます。次に、Linux にマウントするか、Windows でオンラインにすることによって、該当するオペレーティングシステムにアクセスできる状態にします。詳細については、このガイドの次の [インスタンスからブロックストレージにアクセスする] セクションを参照してください。

ステップ 4: インスタンスからブロックストレージディスクにアクセスする

インスタンスにアタッチした後でブロックストレージディスクにアクセスするには、Linux または Unix でマウントするか、Windows でオンラインにする必要があります。

Linux または Unix インスタンスにブロックストレージディスクをマウントしてアクセスする

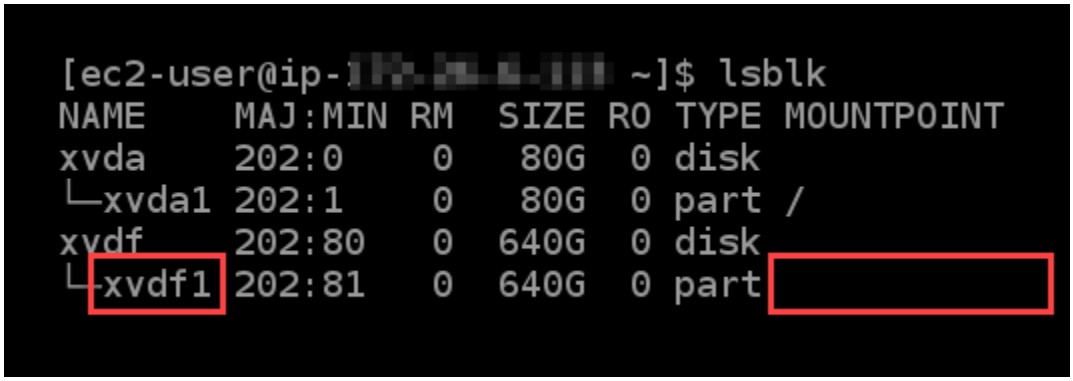
1. Lightsail ホームページで、ブロックストレージディスクをアタッチした Linux または Unix インスタンスのブラウザベースの SSH クライアントのアイコンを選択します。



2. ブラウザベースの SSH クライアントを接続したら、次のコマンドを入力して、インスタンスにアタッチしたブロックストレージディスクデバイスを表示します。

```
lsblk
```

次の例のような結果が表示されます。この例でxvdf1 は、マウントポイントがないため、マウントされていないインスタンスにアタッチされたブロックストレージディスクです。また、結果では、デバイス名から /dev/ が除外されているため、実際のデバイス名は /dev/xvdf1 となります。



3. 次のコマンドを入力して、ブロックストレージディスクのマウントポイントを作成します。

```
sudo mkdir MountPoint
```

コマンドで *MountPoint*を、ブロックストレージディスクをマウントし、アクセス可能にするディレクトリの名前に置き換えます。

例:

```
sudo mkdir xvdf
```

4. 次のコマンドを入力し、前のステップで作成したマウントポイントにブロックストレージディスクをマウントします。

```
sudo mount /dev/DeviceName MountPoint
```

コマンドを、以下のように置き換えます。

- *DeviceName* は、ブロックストレージディスクデバイスの名前に置き換えます。
- *MountPoint* は、前のステップで作成したマウントポイントディレクトリに置き換えます。

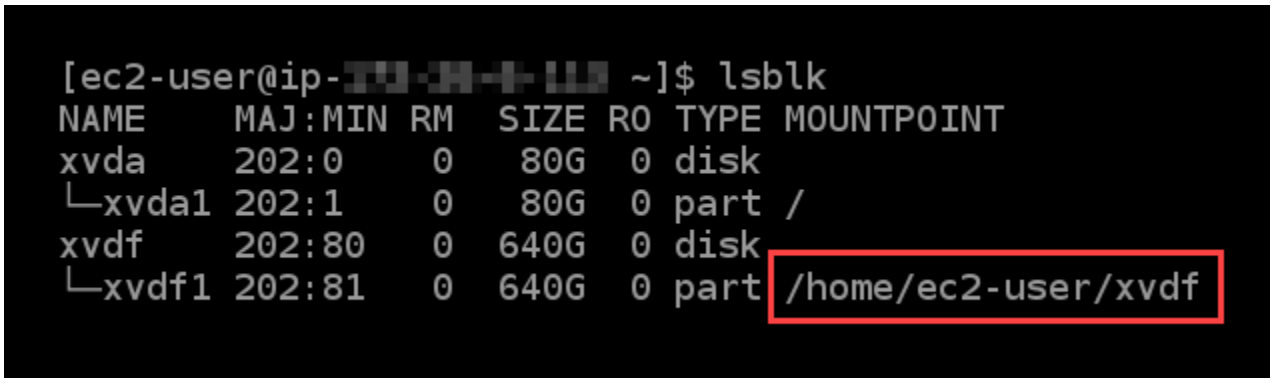
例:

```
sudo mount /dev/xvdf1 xvdf
```

5. 次のコマンドを入力して、インスタンスにアタッチしたブロックストレージディスクデバイスを表示します。

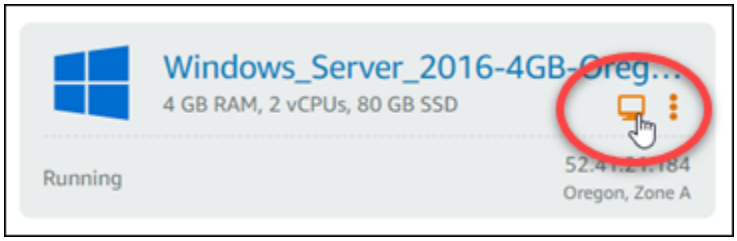
```
lsblk
```

次の例のような結果が表示されます。この例では、*xvdf1* デバイスは */home/ec2-user/xvdf* ディレクトリにマウントされ、アクセス可能になっています。マウントポイントのディレクトリで、ブロックストレージディスクとそのコンテンツにアクセスできるようになりました。

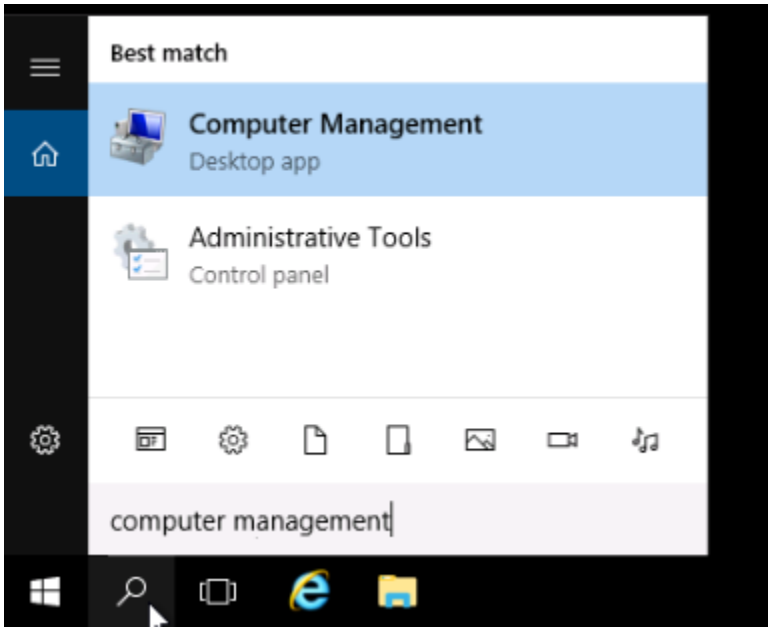


Windows インスタンスでブロックストレージディスクをオンラインにしてアクセスします。

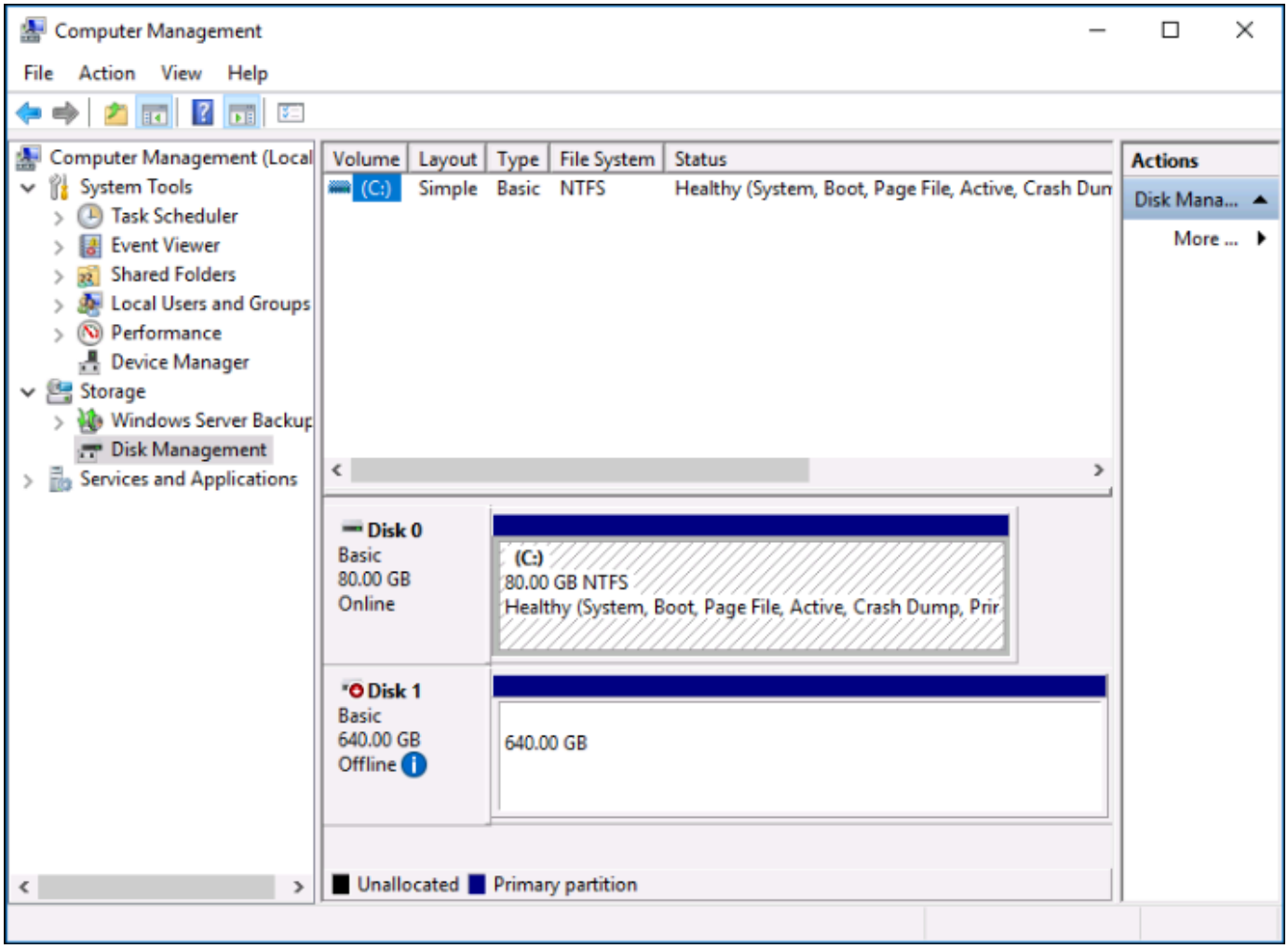
1. Lightsail ホームページで、ブロックストレージディスクをアタッチした Windows インスタンスのブラウザベースの RDP クライアントアイコンを選択します。



2. ブラウザベースの SSH クライアントが接続されたら、Windows タスクバーで [コンピュータの管理] を選択し、その結果から [コンピューターの管理] を選択します。

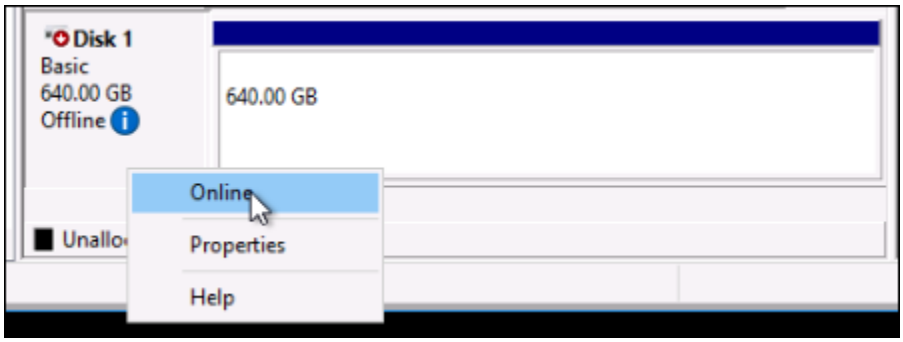


3. [コンピュータの管理] コンソールの左側のナビゲーションメニューで、以下の例のように [ディスク管理] を選択します。

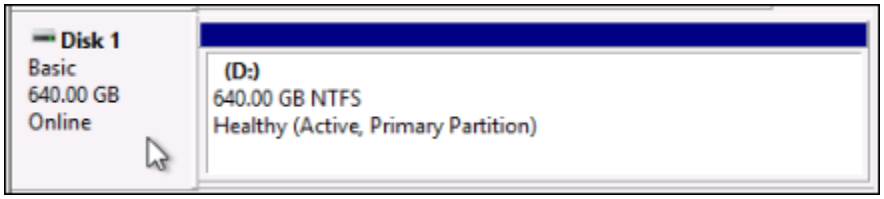


4. 最近インスタンスにアタッチしたディスクを見つけます。[オフライン] とラベル付けされているはずです。

5. [オフライン] ラベルを右クリックし、[オンライン] を選択します。



ディスクが [オンライン] として表示され、ドライブ文字が関連付けられているはずです。File Explorer を開いて指定したドライブ文字を参照することにより、ブロックストレージディスクとそのコンテンツにアクセスできるようになりました。

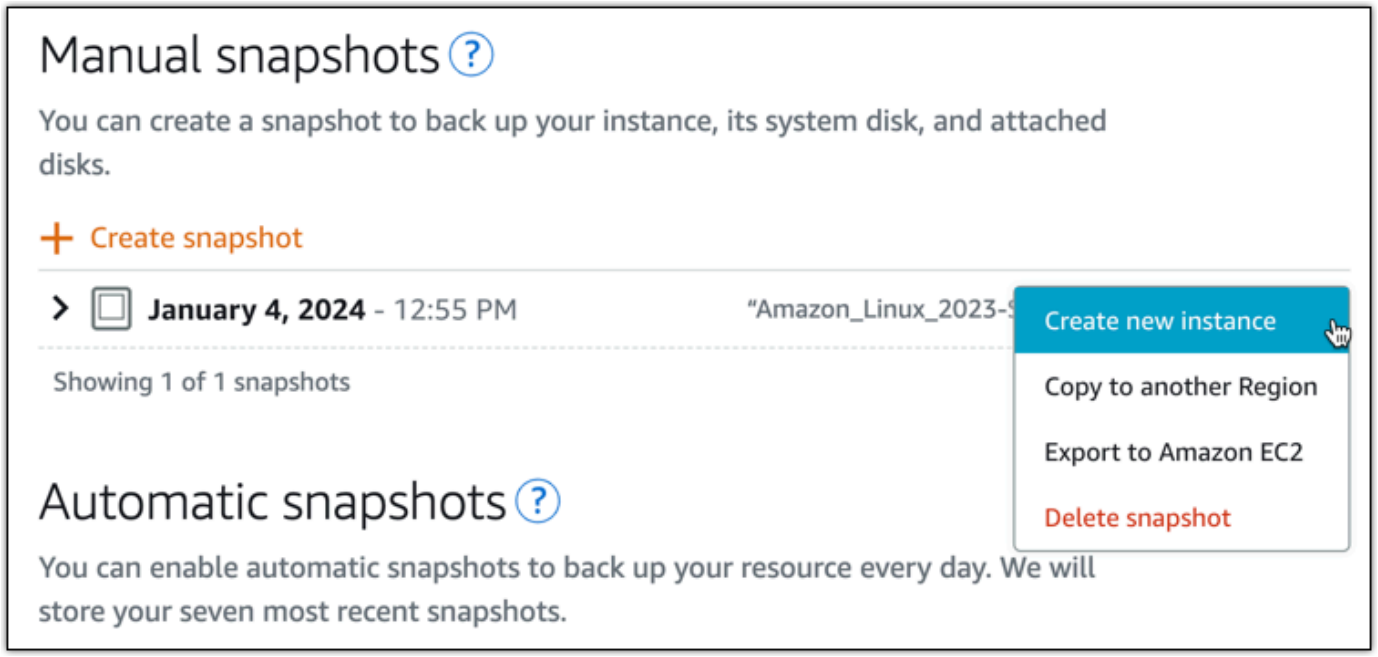


スナップショットから Lightsail インスタンスを作成する

Lightsail でスナップショットを作成したら、そのスナップショットから新しいインスタンスを作成できます。インスタンスサイズやネットワークタイプ - デュアルスタックまたは IPv6-onlyなど、新しいインスタンスの属性を変更できます。新しいインスタンスには、システムディスクと、追加したアタッチされたブロックストレージディスクが含まれます。

そのスナップショットから別のインスタンスを作成する前に、インスタンスのスナップショットが必要です。詳細については、Linux または Unix Lightsail インスタンスのスナップショットを作成するまたはLightsail Windows Server インスタンスのスナップショットを作成するを参照してください。

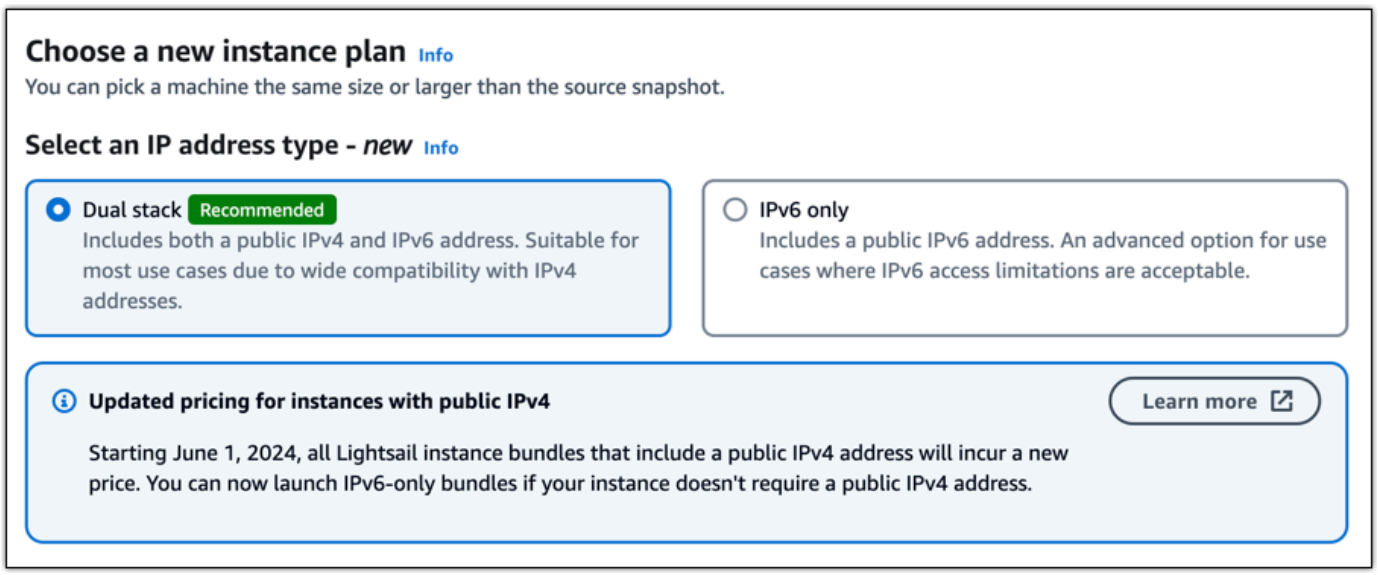
1. Lightsail コンソールで、スナップショットを作成するインスタンスを選択して新しいインスタンスを作成します。
2. [スナップショット] タブを選択します。
3. 手動スナップショット セクションで、スナップショットの横にあるアクションメニューアイコン (⋮) を選択し、新しいインスタンスの作成 を選択します。



4. 「スナップショットからインスタンスを作成する」ページが開きます。使用するオプションの設定を選択します。アベイラビリティーゾーンの変更、起動スクリプトの追加、インスタンスへの接続方法の変更などを行うことができます。

5. 新しいインスタンスのプラン (またはバンドル) を選択します。デュアルスタック (IPv4 および IPv6) インスタンスプランを使用するインスタンスを作成するか、IPv6-onlyプランを使用するインスタンスを作成するかを選択できます。元のインスタンスよりも大きなバンドルサイズを選択することもできます。IPv6-onlyインスタンスプランの詳細については、「」を参照してくださいLightsail IPv6-onlyインスタンスプラン。

> **Note**
>
> 元のインスタンスよりも小さなバンドルサイズを使用するインスタンスを作成することはできません。



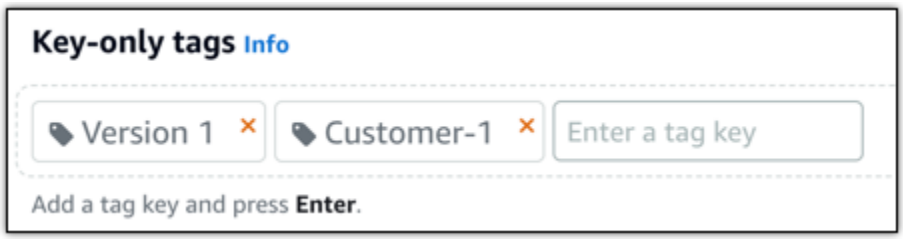
6. インスタンスの名前を入力します。

   リソース名:

   - Lightsail AWS リージョンの各アカウント内で一意である必要があります。
   - 2～255 文字を使用する必要があります。
   - 先頭と末尾は英数字にする必要があります。
   - 英数字、ピリオド、ダッシュ、アンダースコアを含めることができます。

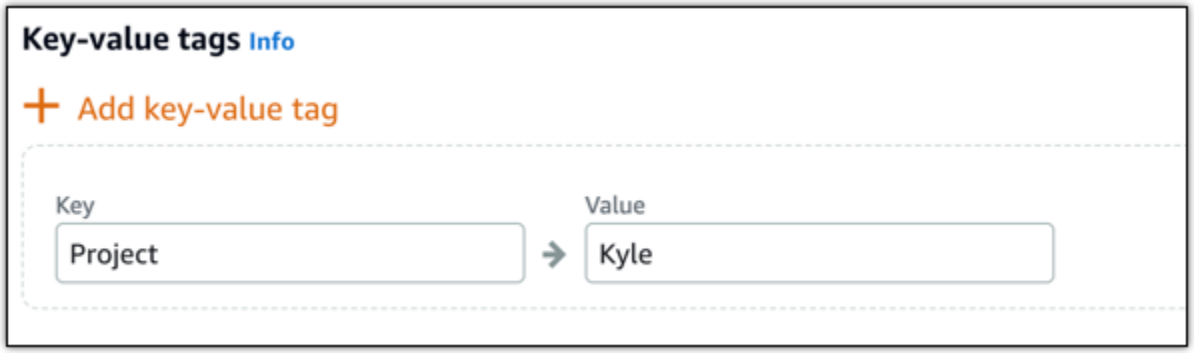
7. 以下のいずれかのオプションを選択して、インスタンスにタグを追加します。

- [Add key-only tags] (キーのみのタグを追加) または [Edit key-only tags] (キーのみのタグを編集) (タグが追加済みの場合)を追加。テキストボックスに新しいタグを入力し、Enter キーを押します。を保存またはキャンセルを選択します。



- キーバリュータグ を作成し、キーテキストボックスにキーを入力し、値テキストボックスに値を入力します。を保存またはキャンセルを選択します。

  キーバリューのタグは、保存する際に一つずつ追加することができます。さらに key-value タグを追加するには、以上のステップを繰り返します。



> **Note**
>
> 「キーのみ」のタグと「キーバリュー」のタグの詳細については、「タグ」を参照してください。

8. [インスタンスの作成] を選択します。

   Lightsail が管理ページを開き、新しいインスタンスを管理できます。

> **Important**
>
> 元のインスタンスのカスタムファイアウォールルールは、スナップショットから作成した新しいインスタンスにはコピーされません。デフォルトのルールのみが新しいインス

タンスにコピーオーバーします。詳細については、このガイドの後半の「デフォルトのインスタンスのファイアウォールルール」を参照してください。

Lightsail のスナップショットからより大きなインスタンス、ブロックストレージディスク、またはデータベースを作成する

これは、お客様のクラウドプロジェクトが増大して、より多くの処理能力がすぐに必要になった場合に必要です。その場合に、当社ではお客様を支援できます。Lightsail インスタンス、ブロックストレージディスク、またはデータベースのサイズを大きくするには、リソースのスナップショットを作成し、スナップショットを使用してそのリソースの新しく大きなバージョンを作成します。

> **Note**
>
> 元のリソースよりも小さいプランサイズを使用して、スナップショットからリソースを作成することはできません。たとえば、8 GB のインスタンスから 2 GB のインスタンスに移行することはできません。
> インスタンスの作成時にインスタンスに割り当てられるデフォルトの公開 IPv4 アドレスは、インスタンスを停止してまた開始すると変更されます。必要に応じて、静的 IPv4 アドレスをインスタンスに作成し、アタッチできます。静的 IP アドレスを使用すると、アドレスをアカウント内の別のインスタンスに迅速に再マッピングすることで、インスタンスやソフトウェアの障害をマスクできます。または、ドメインがインスタンスを参照するように、ドメインの DNS レコードに静的 IP アドレスを指定することもできます。詳細については、「IP アドレス」を参照してください。

前提条件

Lightsail インスタンス、ブロックストレージディスク、またはデータベースのスナップショットが必要になります。詳細については、「スナップショット」を参照してください。

リソースを作成する

1. Lightsail コンソールにサインインします。
2. [スナップショット] タブを選択します。
3. 新しく大きなリソースを作成するためにスナップショットを使用する Lightsail リソースを見つけ、右矢印を選択してスナップショットのリストを展開します。

4. 使用するスナップショットの横にある省略記号アイコンを選択し、[Create new] (新規作成) を選択します。

5. [作成] ページには、選択可能なオプション設定がいくつかあります。たとえば、アベイラビリティーゾーンを変更できます。インスタンスの場合は、起動スクリプトを追加したり、接続に使用する SSH キーを変更したりできます。

   すべてのデフォルト値をそのまま使用して、次のステップに進むことができます。

6. 新しいリソースのプラン (またはバンドル) を選択します。この時点で、必要に応じて、元のリソースよりも大きなバンドルサイズを選択できます。

> **Note**
>
> 元のリソースよりも小さなプランサイズを使用してリソースを作成することはできません。元のリソースよりも小さなバンドルオプションは使用できません。

7. インスタンスの名前を入力します。

   リソース名:

   - Lightsail アカウントの各 AWS リージョン 内で一意である必要があります。
   - 2〜255 文字を使用する必要があります。
   - 先頭と末尾は英数字または数字を使用する必要があります。
   - 英数字、数字、ピリオド、ダッシュ、アンダースコアを使用することができます。

8. [作成] を選択します。

   Lightsail では新しいリソースの管理ページに移動し、管理を開始できます。

AWS CLI を使用して Lightsail スナップショットからより大きなインスタンス、ブロックストレージディスク、またはデータベースを作成する

これは、お客様のクラウドプロジェクトが増大して、より多くの処理能力がすぐに必要になった場合に必要です。その場合に、当社ではお客様を支援できます。すべての操作は、Lightsail コンソール内で行うか、AWS Command Line Interface (AWS CLI) を使用して行うことができます。

現在の インスタンスのスナップショットLightsailを作成し、そのスナップショットに基づいてお客様が必要とする処理能力を持つ、より大きいサイズの新規インスタンスを作成する方法を提示します。

> **Note**
>
> 現時点では、スナップショットより小さいサイズ (またはバンドル) のインスタンスの作成はサポートされていません。作成できるのは、同じサイズまたはより大きいサイズのインスタンスのみです。

前提条件

1. まず、まだ AWS CLI をインストールしていない場合はインストールする必要があります。詳細については、「AWS Command Line Interface のインストール」を参照してください。AWS CLI を設定する必要があります。
2. 作業するインスタンスのスナップショットも必要です。詳細については、「Linux または Unix インスタンスのスナップショットを作成する」を参照してください 。

ステップ 1: スナップショット名を取得する

明らかだと思われるかもしれませんが、この AWS CLI コマンドを実行してより大きいインスタンスを作成する前に、スナップショット名を知っている必要があります。幸いなことに、その名前は簡単に取得できます。

1. AWS CLI で次のように入力します。

   ```
   aws lightsail get-instance-snapshots
   ```

   次のような出力が表示されます。

```
{
    "instanceSnapshots": [
        {
            "fromInstanceName": "WordPress-512MB-EXAMPLE",
            "name": "WordPress-512MB-EXAMPLE-system-1234567891011",
            "sizeInGb": 20,
            "resourceType": "InstanceSnapshot",
            "fromInstanceArn":
            "arn:aws:lightsail:us-
east-1:123456789101:Instance/86f49ee4-26cc-4802-9b0d-12345EXAMPLE",
            "state": "available",
            "arn": "arn:aws:lightsail:us-east-1:123456789101:InstanceSnapshot/
c87acb5f-851e-4fbc-94f1-12345EXAMPLE",
            "fromBundleId": "nano_1_0",
            "fromBlueprintId": "wordpress_4_6_1",
            "createdAt": 1480898073.653,
            "location": {
                "availabilityZone": "all",
                "regionName": "us-east-2"
            }
        }
    ]
}
```


2. [name] の値を、後で見つけやすい場所にコピーします。この名前は、AWS CLI コマンドで `--instance-snapshot-name` の値として使用します。

ステップ 2: バンドルを選択する

バンドルは、実際にはインスタンスの料金プランおよび設定です。たとえば、Linux ベースの Medium バンドルは、1 か月あたり 20 USD の料金で、4.0 GB の RAM、80 GB の SSD ストレージなどが設定されています。

小さいバンドルから始めて、処理能力の追加が必要になった場合は、より大きなバンドルにアップグレードできます。詳細については、「スナップショットからより大きなインスタンス、ブロックストレージディスク、またはデータベースを作成する」を参照してください。

> **Important**
>
> スナップショットからより小さいサイズのバンドルに変更することはできません。より小さいバンドルを作成する場合は、最初からやり直す必要があります。

1. 次の AWS CLI コマンドを入力します。

```
aws lightsail get-bundles
```

出力は次のようになります。

```
{
    "bundles": [
        {
            "name": "Nano",
            "power": 300,
            "price": 5.0,
            "ramSizeInGb": 0.5,
            "diskSizeInGb": 20,
            "transferPerMonthInGb": 1024,
            "cpuCount": 1,
            "instanceType": "t2.nano",
            "isActive": true,
            "bundleId": "nano_1_0"
        },
        {
            "name": "Micro",
            "power": 500,
            "price": 10.0,
            "ramSizeInGb": 1.0,
            "diskSizeInGb": 30,
            "transferPerMonthInGb": 2048,
            "cpuCount": 1,
            "instanceType": "t2.micro",
            "isActive": true,
            "bundleId": "micro_1_0"
        },
        {
            "name": "Small",
            "power": 1000,
            "price": 20.0,
```

```
            "ramSizeInGb": 2.0,
            "diskSizeInGb": 40,
            "transferPerMonthInGb": 3072,
            "cpuCount": 1,
            "instanceType": "t2.small",
            "isActive": true,
            "bundleId": "small_1_0"
        },
        {
            "name": "Medium",
            "power": 2000,
            "price": 40.0,
            "ramSizeInGb": 4.0,
            "diskSizeInGb": 60,
            "transferPerMonthInGb": 4096,
            "cpuCount": 2,
            "instanceType": "t2.medium",
            "isActive": true,
            "bundleId": "medium_1_0"
        },
        {
            "name": "Large",
            "power": 3000,
            "price": 80.0,
            "ramSizeInGb": 8.0,
            "diskSizeInGb": 80,
            "transferPerMonthInGb": 5120,
            "cpuCount": 2,
            "instanceType": "t2.large",
            "isActive": true,
            "bundleId": "large_1_0"
        }
    ]
}
```


2. 目的のバンドルの [bundleId] の値を見つけます。詳細については、「Lightsail 料金表」を参照してください。

ステップ 3: AWS CLI コマンドを記述して新規インスタンスを作成する

これで、パラメーター値が取得済みであるため、インスタンスを作成するためのコマンドを記述して実行する準備ができました。

1. 次の内容を入力します。

```
aws lightsail create-instances-from-snapshot --instance-names
 MyNewInstanceFromSnapshot --availability-zone us-east-1a --instance-snapshot-name
 WordPress-512MB-EXAMPLE-system-1234567891011 --bundle-id medium_1_0
```

出力は次のようになります。

```
{
    "operations": [
        {
            "status": "Started",
            "resourceType": "Instance",
            "isTerminal": false,
            "statusChangedAt": 1486863990.961,
            "location": {
                "availabilityZone": "us-east-2a",
                "regionName": "us-east-2"
            },
            "operationType": "CreateInstance",
            "resourceName": "MyNewInstanceFromSnapshot",
            "id": "30fec45e-e7d7-4e18-96c8-12345EXAMPLE",
            "createdAt": 1486863989.784
        }
    ]
}
```

> **Note**
>
> AWS CLI を使用して、リージョンとアベイラビリティーゾーンのリストを返すこともできます。`aws lightsail get-regions --include-availability-zones` リクエストでアベイラビリティーゾーンのリストを返すには、`get-regions` と入力します。

2. Lightsail コンソールで新しいインスタンスを開き、インスタンスの変更を開始します。

次のステップ

スナップショットから新しいインスタンスを作成した後に、次のことを行うことができます。

- 以前のインスタンスを使用しない場合は削除できます。これを行うには、Lightsail コンソールまたは CLI コマンド delete-instance を使用します。
- 以前のスナップショットが必要ない場合は削除できます。これを行うには、Lightsail コンソールまたは CLI コマンド delete-instance-snapshot を使用します。
- 以前のインスタンスに静的 IP アドレスがアタッチされていた場合は、そのまま新しいインスタンスにアタッチできます。これを行うには、コンソールを使用します。「静的 IP を作成してインスタンスにアタッチする」を参照してください。

Lightsail スナップショットを削除する

月額料金が発生しないように、不要になった Amazon Lightsail のインスタンス、データベース、ディスクスナップショットは削除します。

個々のスナップショットを削除する

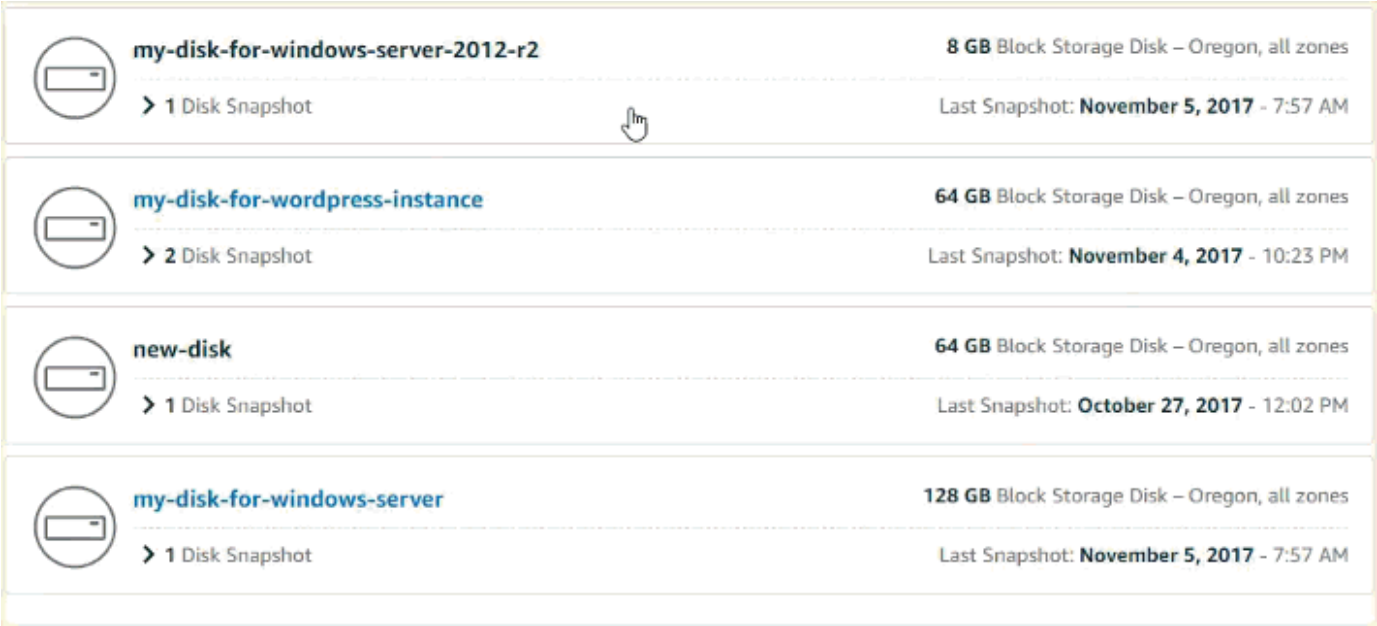
1. Lightsail コンソールで、[スナップショット] タブを選択します。
2. スナップショットを削除する Lightsail リソースを見つけ、右矢印を選択して、そのリソースで使用可能なスナップショットのリストを展開します。
3. 削除するスナップショットの横にあるアクションメニューアイコン (⋮) を選択してから、[スナップショットの削除] を選択します。

4. [はい] を選択して、スナップショットを削除することを確定します。

> **Important**
>
> これは永続的オペレーションで、取消すことはできません。削除するとスナップショット上のすべてのデータが失われます。

複数のスナップショットを削除する

1. Lightsail のホームページで [スナップショット] を選択します。
2. 削除するスナップショットの Lightsail リソースを見つけ、右矢印を選択してスナップショットのリストを展開します。



3. [複数を削除] を選択します。
4. 削除するスナップショットを選択し、[削除] を選択します。
5. [はい] を選択して、スナップショットを削除することを確定します。

> **Important**
>
> これは永続的オペレーションで、取消すことはできません。削除するとスナップショット上のすべてのデータが失われます。

Lightsail インスタンスとディスクの自動スナップショットを有効または無効にする

インスタンスまたはブロックストレージディスクの自動スナップショット機能を有効化すると、Amazon Lightsail は、デフォルトの自動スナップショット時間、または指定した時刻にリソースのスナップショットを作成します。手動スナップショットと同様に、自動スナップショットをベースラインとして、新しいリソースを作成したり、データをバックアップできます。

自動スナップショットが作成されると、Lightsail アカウントに保存されている自動スナップショットはスナップショットストレージ料金が課金されます。

目次

自動スナップショットの制限

自動スナップショットには、以下の制限が適用されます。

- Lightsail コンソールを使用して、ブロックストレージディスクの自動スナップショットを有効または無効にすることはできません。ブロックストレージディスクの自動スナップショットを有効または無効にするには、Lightsail API、AWS Command Line Interface (AWS CLI)、または SDK を使用する必要があります。詳細については、「AWS CLI を使用した自動スナップショットを有効または無効にする」を参照してください。
- 自動スナップショットは現在、Windows インスタンスまたはマネージドデータベースではサポートされていません。代わりに、Windows インスタンスまたはマネージドデータベースの手動スナップショットを作成して、それらをバックアップする必要があります。詳細については、「Windows Server インスタンスのスナップショットを作成する」および「データベースのスナップショットを作成する」を参照してください。マネージドデータベースには、デフォルトで有効になっているポイントインタイムバックアップ機能もあり、これを使用してデータを新しいデータベースに復元できます。詳細については、「ポイントインタイムバックアップからデータベースを作成する」を参照してください。
- 自動スナップショットでは、ソースリソースのタグは保持されません。自動スナップショットから作成される新しいリソースでソースリソースのタグを保持するには、自動スナップショットから新しいリソースを作成するときにタグを手動で追加する必要があります。詳細については、「リソースにタグを追加する」を参照してください。

自動スナップショット保持

毎日 7 つの最新の自動スナップショットが保存されたあと、最も古いものから最新のものに置き換えられます。さらに、ソースリソースを削除した場合、リソースに関連付けられたすべての自動スナップショットは削除されます。この動作は、ソースリソースを削除した後でも Lightsail アカウントに保存される手動スナップショットとは異なります。自動のスナップショットを置き換えられないようにしたり、ソースリソースを削除した際に削除されないようにしたい場合、自動スナップショットを手動スナップショットとしてコピーすることができます。

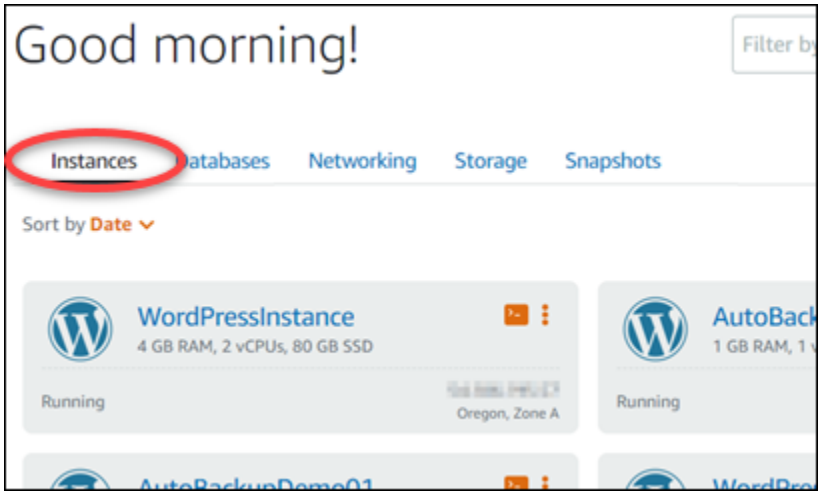
リソースの自動スナップショット機能を無効化した時、リソースの既存の自動スナップショットは、以下のいずれかの操作を行うまで、ソースリソースとともに保持されます。

- 自動スナップショットを再度有効化して、既存の自動スナップショットが新しいスナップショットに置き換える。
- 既存の自動スナップショットを手動で削除する。
- ソースリソースを削除して関連した自動スナップショットを削除する。

Lightsail コンソールを使用してインスタンスの自動スナップショットを有効または無効にする

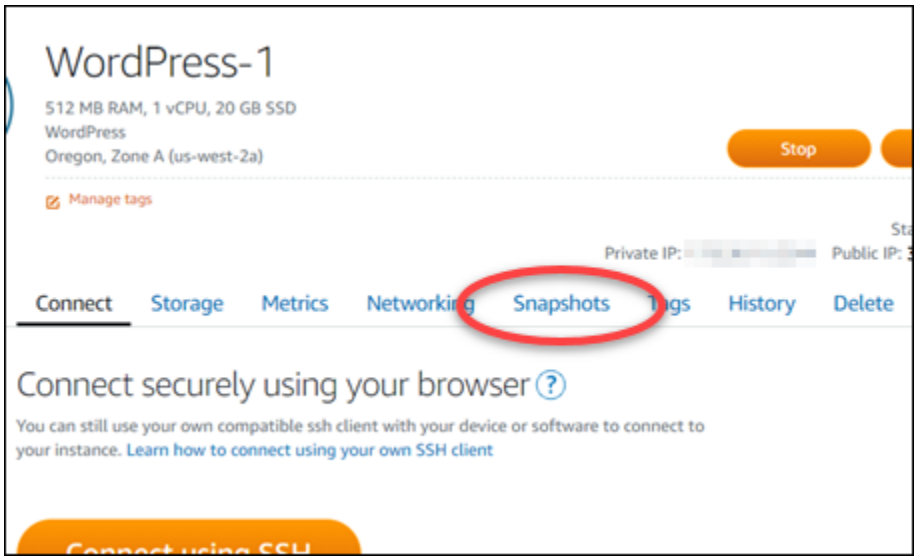
Lightsail コンソールを使用してインスタンスの自動スナップショットを有効または無効にするには、以下の手順を実行します。

1. Lightsail コンソールにサインインします。
2. Lightsail ホームページで、[Instances (インスタンス)] タブを選択します。



3. 自動スナップショットを有効または無効にするインスタンスの名前を選択します。

4. インスタンス管理ページで、[Snapshots (スナップショット)] タブを選択します。



5. [自動スナップショット] セクションで、トグルを選択して有効にします。同様に、有効になっている場合は、トグルを選択して無効にします。
6. プロンプトで、[Yes, enable (はい、有効にする)] を選択して自動スナップショットを有効にするか、[Yes, disable (はい、無効にする)] を選択してこの機能を無効にします。

   しばらくすると、自動スナップショットが有効または無効になります。

   - 自動スナップショット機能を有効にした場合は、自動スナップショット時間の変更も必要になることがあります。詳細については、「インスタンスまたはブロックストレージディスクの自動スナップショット時間を変更する」を参照してください。
   - 自動スナップショット機能を無効にする場合、リソースの既存の自動スナップショットは、この機能を再度有効にして新しいスナップショットに置き換えられるか、お客様が削除するまで、保持されます。Lightsail アカウントに保存されている自動スナップショットは、スナップショットストレージ料金が課金されます。自動スナップショットの削除の詳細については、「インスタンスの自動スナップショットを削除する」を参照してください。

AWS CLI を使用したインスタンスまたはブロックストレージディスクの自動スナップショットを有効または無効にする

AWS CLI を使用してインスタンスまたはブロックストレージディスクの自動スナップショットを有効または無効にするには、以下の手順を実行します。

1. ターミナルまたはコマンドプロントウィンドウを開きます。

まだインストールしていない場合は、AWS CLI をインストールし、Lightsail と連携するように設定します。

2. 自動スナップショットを有効にするか無効にするかに応じて、この手順で説明するコマンドのいずれかを入力します。

> **Note**
>
> これらのコマンドで は、autoSnapshotAddOnRequest={snapshotTimeOfDay=HH:00} パラメータは オプションです。自動スナップショットを有効にするときに毎日の自動スナップショット時間を指定しない場合、Lightsail によってリソースにデフォルトのスナップショット時間が割り当てられます。詳細については、「インスタンスまたはブロックストレージディスクの自動スナップショット時間を変更する」を参照してください。

- 以下のコマンドを入力して、既存のリソースの自動スナップショットを有効にします。

```
aws lightsail enable-add-on --region Region --resource-name ResourceName --add-
on-request
 addOnType=AutoSnapshot,autoSnapshotAddOnRequest={snapshotTimeOfDay=HH:00}
```

コマンドを、以下のように置き換えます。

- *Region* は、リソースが存在する AWS リージョン に置き換えます。
- *ResourceName* は、リソースの名前に置き換えます。
- *HH:00* は、協定世界時 (UTC) での毎日の自動スナップショット時間 (1 時間単位) に置き換えます。

例:

```
aws lightsail enable-add-on --region us-west-2 --resource-name WordPress-1 --add-
on-request
 addOnType=AutoSnapshot,autoSnapshotAddOnRequest={snapshotTimeOfDay=18:00}
```

- 以下のコマンドを入力して、新しいインスタンスを作成するときに自動スナップショットを有効にします。

```
aws lightsail create-instances --region Region --availability-
zone AvailabilityZone --blueprint-id BlueprintID --
bundle-id BundleID --instance-name InstanceName --add-ons
 addOnType=AutoSnapshot,autoSnapshotAddOnRequest={snapshotTimeOfDay=HH:00}
```

コマンドを、以下のように置き換えます。

- インスタンスが作成される AWS リージョン の *#####*。
- *AvailabilityZone* は、インスタンスが作成されるアベイラビリティーゾーンに置き換えます。
- *BlueprintID* は、インスタンスに使用する設計図 ID に置き換えます。
- *BundleID* は、インスタンスに使用するバンドル ID に置き換えます。
- *InstanceName* は、インスタンスに使用する名前に置き換えます。
- *HH:00* は、協定世界時 (UTC) での毎日の自動スナップショット時間 (1 時間単位) に置き換えます。

例:

```
aws lightsail create-instances --region us-west-2 --availability-
zone us-west-2a --blueprint-id wordpress_5_1_1_2 --bundle-
id medium_2_0 --instance-name WordPressInstance --add-ons
 addOnType=AutoSnapshot,autoSnapshotAddOnRequest={snapshotTimeOfDay=20:00}
```

- 以下のコマンドを入力して、新しいディスクを作成するときに自動スナップショットを有効にします。

```
aws lightsail create-disk --region Region --availability-
zone AvailabilityZone --size-in-gb Size --disk-name DiskName --add-ons
 addOnType=AutoSnapshot,autoSnapshotAddOnRequest={snapshotTimeOfDay=HH:00}
```

コマンドを、以下のように置き換えます。

- ディスクが作成される AWS リージョン の *#####*。
- *AvailabilityZone* は、ディスクが作成されるアベイラビリティーゾーンに置き換えます。
- *Size* は、ディスクの希望サイズ (GB 単位) に置き換えます。
- *DiskName* は、ディスクに使用する名前に置き換えます。

- *HH:00* は、協定世界時 (UTC) での毎日の自動スナップショット時間 (1 時間単位) に置き換えます。

例:

```
aws lightsail create-disk --region us-west-2 --availability-
zone us-west-2a --size-in-gb 32 --disk-name Disk01 --add-ons
 addOnType=AutoSnapshot,autoSnapshotAddOnRequest={snapshotTimeOfDay=18:59}
```

- 以下のコマンドを入力して、リソースの自動スナップショットを無効にします。

```
aws lightsail disable-add-on --region Region --resource-name ResourceName --add-
on-type AutoSnapshot
```

コマンドを、以下のように置き換えます。

- *Region* は、リソースが存在する AWS リージョン に置き換えます。
- *ResourceName* は、リソースの名前に置き換えます。

例:

```
aws lightsail disable-add-on --region us-west-1 --resource-
name MyFirstWordPressWebsite01 --add-on-type AutoSnapshot
```

以下の例のような結果が表示されるはずです。

```
{
    "operations": [
        {
            "id": "2610213c-d68f-488e-9124-245913a2a22a",
            "resourceName": "WordPressInstance",
            "resourceType": "Instance",
            "createdAt": 1566431564.323,
            "location": {
                "availabilityZone": "us-west-2a",
                "regionName": "us-west-2"
            },
            "isTerminal": false,
            "operationType": "CreateInstance",
            "status": "Started",
            "statusChangedAt": 1566431564.323
        },
        {
            "id": "fd04446d-8106-4c7e-8d69-f42be811453a",
            "resourceName": "WordPressInstance",
            "resourceType": "Instance",
            "createdAt": 1566431566.368,
            "location": {
                "availabilityZone": "us-west-2",
                "regionName": "us-west-2"
            },
            "isTerminal": false,
            "operationDetails": "EnableAddOn - AutoBackup",
            "operationType": "EnableAddOn",
            "status": "Started"
        }
    ]
}
```


しばらくすると、自動スナップショットが有効または無効になります。

- 自動スナップショット機能を有効にした場合は、自動スナップショット時間の変更も必要になることがあります。詳細については、「インスタンスまたはブロックストレージディスクの自動スナップショット時間を変更する」を参照してください。
- 自動スナップショット機能を無効にする場合、既存の自動スナップショットは、この機能を再度有効にして新しいスナップショットに置き換えられるか、お客様が削除するまで、保持されます。Lightsail アカウントに保存されている自動スナップショットは、スナップショットストレージ料金が課金されます。自動スナップショットの削除の詳細については、「インスタンスの自動スナップショットを削除する」を参照してください。

> **Note**
>
> EnableAddOn と DisableAddOn API オペレーションの詳細については、「Lightsail API ドキュメントの EnableAddOn および DisableAddOn」 を参照してください。

Lightsail での自動スナップショットの時間を変更する

インスタンスまたはブロックストレージディスクの自動スナップショット特徴を有効化する際、Lightsail は、リソースのスナップショットを 1 日 1 回デフォルトの自動スナップショット時間、または指定した時刻に作成します。このガイドの手順に従って、リソースの自動スナップショット時間を変更します。

目次

自動スナップショット時間の制限

自動スナップショット時間には、以下の制限が適用されます。

- Lightsail コンソールを使用して、ブロックストレージディスクの自動スナップショット時間を変更することはできません。ブロックストレージディスクの自動スナップショット時間を変更するには、Lightsail API、AWS Command Line Interface (AWS CLI)、または SDK を使用する必要があります。詳細については、「AWS CLI を使用して自動スナップショット時間を変更する」を参照してください。
- 自動スナップショット時間は 1 時間単位でのみ指定できます。また、現在の時刻から 30 分後よりも後の時間である必要もあります。Lightsail は指定した時刻から最大 45 分後までの間に自動スナップショットを作成します。

> **Important**
>
> 自動スナップショットの作成中は、手動スナップショットを作成できません。

- リソースの自動スナップショット時間を変更すると、以下の条件下でなければ、その時間は通常すぐに有効になります。
  - 現在の日に自動スナップショットがすでに作成されていて、スナップショット時間を後の時刻に変更した場合、新しいスナップショット時間は翌日に有効になります。その結果、現在の日に 2 つのスナップショットが作成されることはありません。

- 現在の日の自動スナップショットがまだ作成されておらず、スナップショット時間をその日の過去の時刻に変更した場合、新しいスナップショット時間は翌日に有効になります。また、スナップショットは、現在の日の以前に設定した時刻に自動的に作成されます。その結果、現在の日のスナップショットが作成されます。
- 現在の日の自動スナップショットがまだ作成されておらず、スナップショット時間を現在の時刻から 30 分以内の時刻に変更した場合、新しいスナップショット時間は翌日に有効になります。また、スナップショットは、現在の日の以前に設定した時刻に自動的に作成されます。現在の時間と指定した新しいスナップショット時間の間に 30 分が必要であるため、現在の日にスナップショットが作成されます。
- 現在の時間から 30 分以内に自動スナップショットが作成されるようにスケジュールされている場合、スナップショット時間を変更すると、新しいスナップショット時間は翌日に有効になります。また、スナップショットは、現在の日の以前に設定した時刻に自動的に作成されます。現在の時間と指定した新しいスナップショット時間の間に 30 分が必要であるため、現在の日にスナップショットが作成されます。

これらの条件のいずれかに当てはまると、Lightsail コンソールにメッセージが表示されて、新しいスナップショットが有効になるまでに最大 24 時間かかる場合があることが通知されます。

AWS リージョン のデフォルトの自動スナップショット時間

自動スナップショットを有効にするときに自動スナップショット時間を指定しない場合は、Lightsail によって以下のデフォルトの自動スナップショット時間のいずれかが割り当てられます。この時間は、インスタンスまたはブロックストレージディスクが存在する AWS リージョン によって異なります。

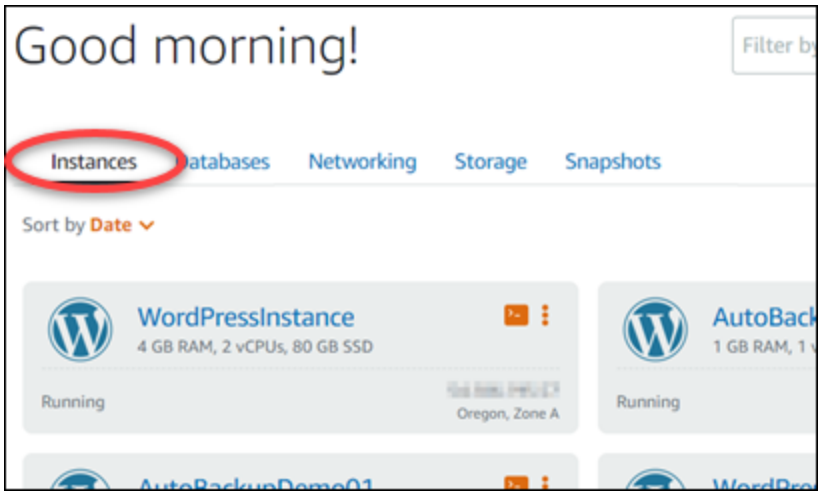
- 米国東部 (オハイオ) (us-east-2): 03:00 UTC
- 米国東部 (バージニア北部) (us-east-1): 06:00 UTC
- 米国西部 (オレゴン) (us-west-2): 06:00 UTC
- アジアパシフィック (ムンバイ) (ap-south-1): 17:00 UTC
- アジアパシフィック (ソウル) (ap-northeast-2): 13:00 UTC
- アジアパシフィック (シンガポール) (ap-southeast-1): 14:00 UTC
- アジアパシフィック (シドニー) (ap-southeast-2)): 12:00 UTC
- アジアパシフィック (東京) (ap-northeast-1): 13:00 UTC
- カナダ (中部) (ca-central-1): 06:00 UTC
- 欧州 (フランクフルト) (eu-central-1): 20:00 UTC

- 欧州 (アイルランド) (eu-west-1): 22:00 UTC
- 欧州 (ロンドン) (eu-west-2): 06:00 UTC
- 欧州 (パリ) (eu-west-3): 07:00 UTC
- 欧州 (ストックホルム) (eu-north-1): 08:00 UTC

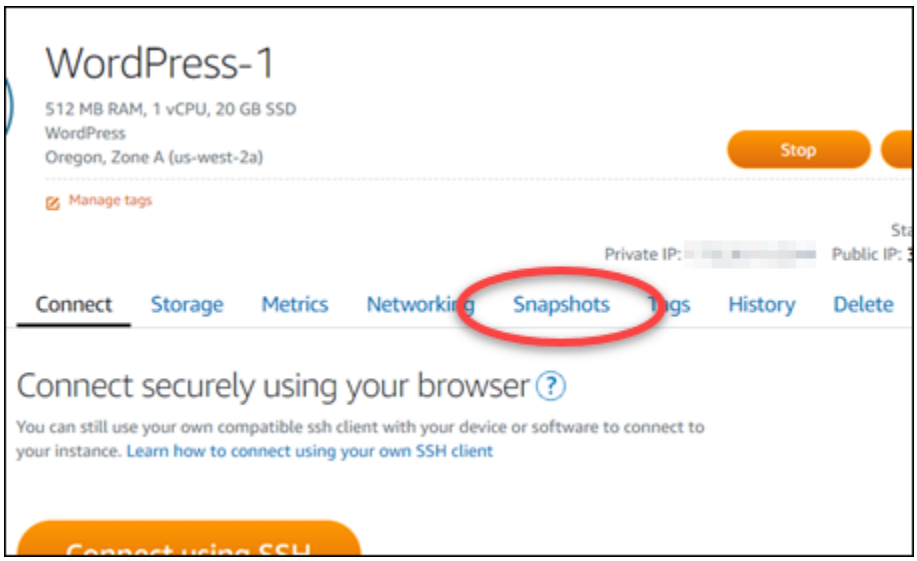
Lightsail コンソールを使用して自動スナップショット時間を変更する

Lightsail コンソールを使用してインスタンスの自動スナップショット時間を変更するには、以下の手順を実行します。

1. Lightsail コンソールにサインインします。
2. Lightsail ホームページで、[インスタンス] タブを選択します。



3. 自動スナップショット時間を変更するインスタンスの名前を選択します。
4. インスタンス管理ページで、[Snapshots (スナップショット)] タブを選択します。



5. [Automatic snapshots (自動スナップショット)] セクションで、[Change snapshot time (スナップショット時間の変更)] を選択します。
6. Lightsail によって自動スナップショットが作成される時刻を選択します。選択する時間は、協定世界時 (UTC) であることが必要です。
7. [変更] を選択して、新しいスナップショット時間を保存します。

   しばらくすると、自動スナップショット時間が更新されます。制限は新しい自動スナップショット時間の発効日に適用されるものとします。詳細については、「自動スナップショット時間の制限」を参照してください。

AWS CLI を使用してインスタンスおよびブロックストレージディスクの自動スナップショット時間を変更する

AWS CLI を使用してインスタンスまたはブロックストレージディスクの自動スナップショット時間を変更するには、以下の手順を実行します。

1. ターミナルまたはコマンドプロントウィンドウを開きます。

   まだインストールしていない場合は、AWS CLI をインストールし、Lightsail と連携するように設定します。
2. 以下のコマンドを入力して、リソースの自動スナップショット時間を変更します。

```
aws lightsail enable-add-on --region Region --resource-name ResourceName --add-on-request addOnType=AutoSnapshot,autoSnapshotAddOnRequest={snapshotTimeOfDay=HH:00}
```

コマンドを、以下のように置き換えます。

- *Region* は、リソースが存在する AWS リージョン に置き換えます。
- *ResourceName* は、リソースの名前に置き換えます。
- *HH:00* は、協定世界時 (UTC) での毎日の自動スナップショット時間 (1 時間単位) に置き換えます。

例:

```
aws lightsail enable-add-on --region us-west-1 --resource-
name MyFirstWordPressWebsite01 --add-on-request
 addOnType=AutoSnapshot,autoSnapshotAddOnRequest={snapshotTimeOfDay=12:00}
```

以下の例のような結果が表示されるはずです。

しばらくすると、自動スナップショット時間が更新されます。制限は新しい自動スナップショット時間の発効日に適用されるものとします。詳細については、「自動スナップショット時間の制限」を参照してください。

> **Note**
>
> EnableAddOn API オペレーションの詳細については、Lightsail API ドキュメントの EnableAddOn を参照してください。

Lightsail で自動スナップショットを削除する

Amazon Lightsail でインスタンスまたはブロックストレージディスクの自動スナップショットはいつでも削除できます。これは、自動スナップショットを有効にしていても、有効にしていた後に無効にしていても同じです。Lightsail アカウントに自動的に保存されているスナップショットに対してスナップショットストレージ料金が請求されます。自動スナップショットが不要になった場合は、このガイドの手順に従って削除します。たとえば、自動スナップショットを手動スナップショットにコピーして元のスナップショットが不要になった場合や、リソースの自動スナップショット機能を無効にしたため、保持している既存の自動スナップショットが不要になった場合です。

目次

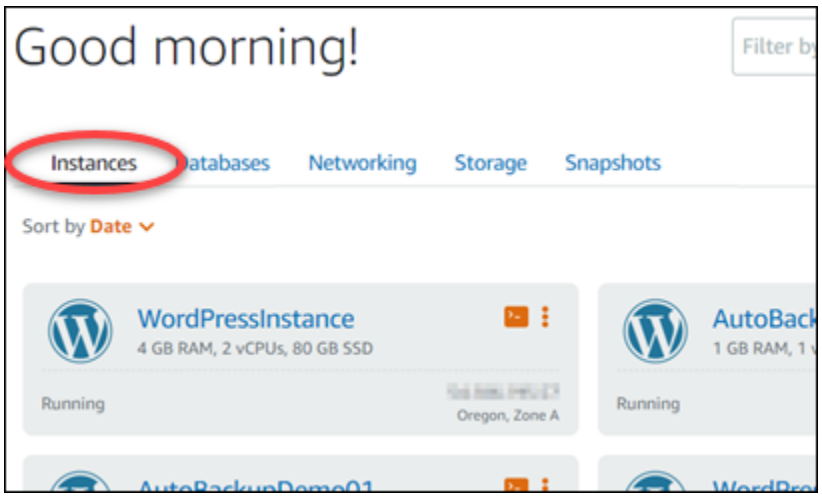
自動スナップショットの削除に関する制限

ブロックストレージディスクの自動スナップショットは、Lightsail コンソールを使用して削除できません。ブロックストレージディスクの自動スナップショットを削除するには、Lightsail API、AWS Command Line Interface (AWS CLI)、または SDK を使用する必要があります。詳細については、「AWS CLI を使用してインスタンスまたはブロックストレージディスクの自動スナップショットを削除する」を参照してください。

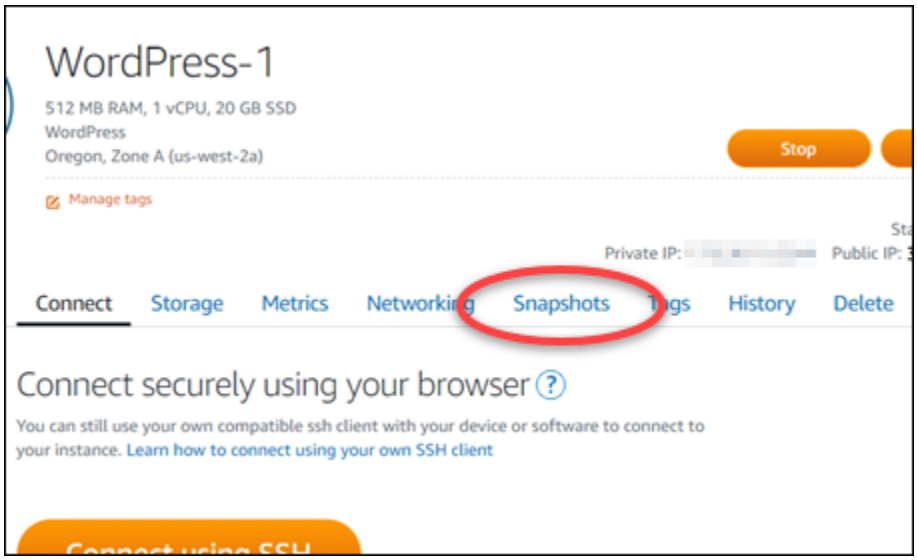
Lightsail コンソールを使用してインスタンスの自動スナップショットを削除する

Lightsail コンソールを使用してインスタンスの自動スナップショットを削除するには、以下の手順を実行します。

1. Lightsail コンソールにサインインします。
2. Lightsail ホームページで、[インスタンス] タブを選択します。



3. 自動スナップショットを削除するインスタンスの名前を選択します。
4. インスタンス管理ページで、[Snapshots (スナップショット)] タブを選択します。



5. [Automatic snapshots (自動スナップショット)] セクションで、削除する自動スナップショットの横にある省略記号アイコンを選択し、[スナップショットの削除] を選択します。
6. プロンプトで、[はい] を選択して、スナップショットを削除することを確定します。

   しばらくすると、自動スナップショットが削除されます。

AWS CLI を使用してインスタンスまたはブロックストレージディスクの自動スナップショットを削除する

AWS CLI を使用してインスタンスまたはブロックストレージディスクの自動スナップショットを削除するには、以下の手順を実行します。

1. ターミナルまたはコマンドプロントウィンドウを開きます。

   まだインストールしていない場合は、AWS CLI をインストールし、Lightsail と連携するように設定します。

2. 以下のコマンドを入力して、特定のリソースの使用可能な自動スナップショットの日付を取得します。自動スナップショットの日付は、後続のコマンドで date パラメータとして指定するために必要です。

   ```
   aws lightsail --region Region get-auto-snapshots --resource-name ResourceName
   ```

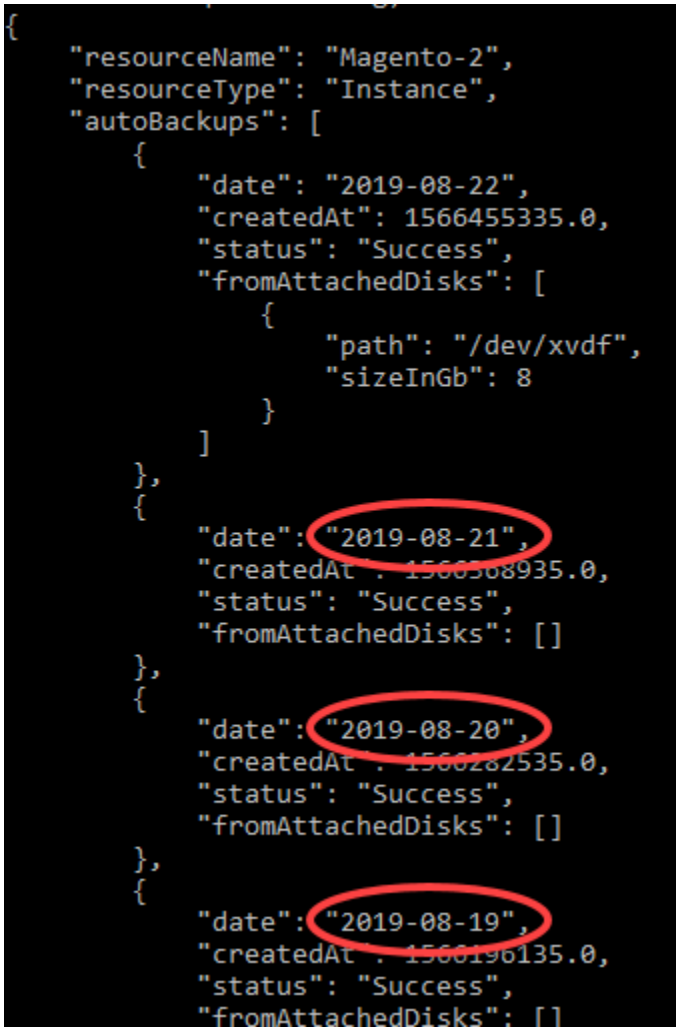
   コマンドを、以下のように置き換えます。

   - *Region* は、リソースが存在する AWS リージョン に置き換えます。
   - *ResourceName* は、リソースの名前に置き換えます。

   例:

   ```
   aws lightsail --region us-west-2 get-auto-snapshots --resource-
   name MyFirstWordPressWebsite01
   ```

   以下のような結果が表示され、使用可能な自動スナップショットが一覧表示されます。



3. 以下のコマンドを入力して、自動スナップショットを削除します。

```
aws lightsail --region Region delete-auto-snapshot --resource-name ResourceName --
date YYYY-MM-DD
```

コマンドを、以下のように置き換えます。

- *Region* は、リソースが存在する AWS リージョン に置き換えます。
- *ResourceName* は、リソースの名前に置き換えます。
- *YYYY-MM-DD* は、前のコマンドで取得した使用可能な自動スナップショットの日付に置き換えます。

例:

```
aws lightsail --region us-west-2 delete-auto-snapshot --resource-
name MyFirstWordPressWebsite01 --date 2019-09-16
```

以下の例のような結果が表示されるはずです。

しばらくすると、自動スナップショットが削除されます。

> **Note**
>
> これらのコマンドのスナップショットの GetAutoSnapshot および DeleteAutoSnapshot API オペレーションの詳細については、Lightsail API ドキュメント内の「スナップショットの取得」および「自動スナップショットの削除」を参照してください。

Lightsail で自動スナップショットを保持する

Amazon Lightsail でインスタンスまたはブロックストレージディスクに対して自動スナップショット機能を有効にする場合、リソースの自動スナップショットは最新の 7 日分のみが保存されます。その後、最も古いものが最新のものに置き換えられます。さらに、出典リソースを削除した場合、リソースに関連付けられたすべての自動スナップショットは削除されます。

特定の自動スナップショットを置き換えられないようにしたり、ソースリソースを削除した際にリソースも削除されないようにしたい場合は、手動スナップショットとしてコピーすることができます。手動スナップショットは、ユーザーがマニュアルで削除しない限り保持されます。

このガイドの手順に従って、自動スナップショットを手動スナップショットとしてコピーして保存します。Lightsailアカウントに保存されている自動スナップショットは、スナップショットストレージ料金が課金されます。

> **Note**
>
> リソースの自動スナップショット機能を無効にする場合、リソースの既存の自動スナップショットは、この機能を再度有効にして新しいスナップショットに置き換えられるか、お客様が自動スナップショットを削除するまで、保持されます。

目次

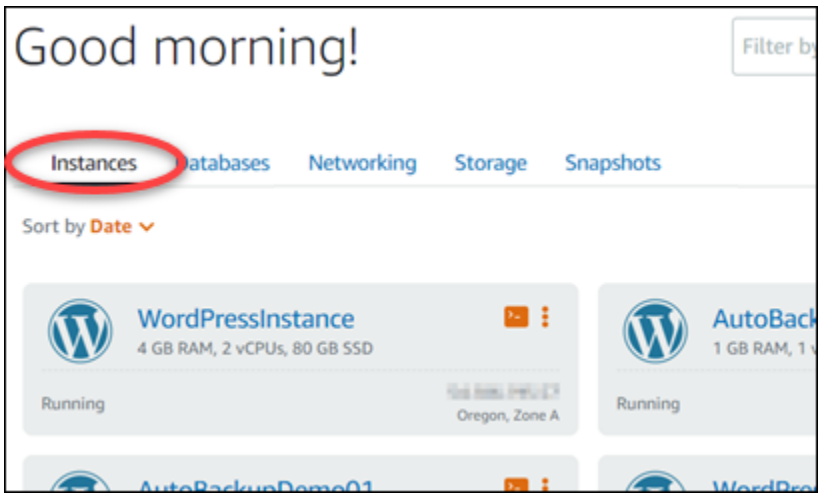
自動スナップショットの制限を保持する

Lightsail コンソールを使用して、ブロックストレージディスクの自動スナップショットを手動スナップショットにコピーすることはできません。ブロックストレージディスクの自動スナップショットをコピーするには、Lightsail API、AWS Command Line Interface (AWS CLI)、または SDK を使用する必要があります。詳細については、「AWS CLI を使用したインスタンスおよびブロックストレージディスクの自動スナップショットの保持」を参照してください。

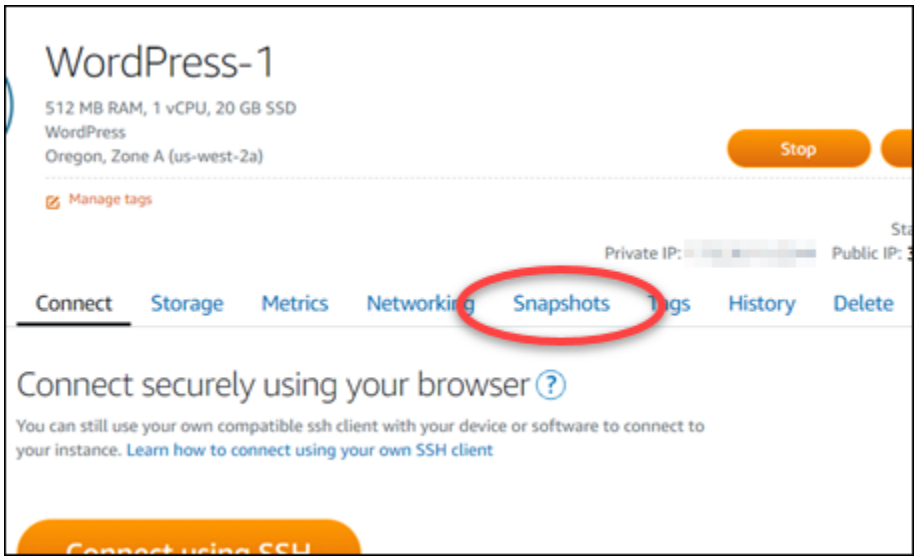
Lightsail コンソールを使用したインスタンスの自動スナップショットの保持

Lightsail コンソールを使用してインスタンスの自動スナップショットを保持するには、以下の手順を実行します。

1. Lightsail コンソールにサインインします。
2. Lightsail ホームページで、[インスタンス] タブを選択します。



3. 自動スナップショットを保持するインスタンスの名前を選択します。
4. インスタンス管理ページで、[Snapshots (スナップショット)] タブを選択します。



5. [Automatic snapshots (自動スナップショット)] セクションで、保持する自動スナップショットの横にある省略記号アイコンを選択し、[Keep snapshot (スナップショットの保持)] を選択します。
6. プロンプトで「Yes, save (はい、保存する)」を選択して、自動スナップショットを保持することを確定します。

   しばらくすると、自動スナップショットが手動スナップショットとしてコピーされます。手動スナップショットは、お客様が削除するまで保持されます。

> **Important**
>
> 自動スナップショットが不要になった場合は、削除することをお勧めします。そうでないと、Lightsailアカウントに保存される自動スナップショットと重複する手動スナップショットに対してスナップショットストレージ料金が課金されます。詳細については、「自動インスタンスのスナップショットを削除する」を参照してください。

AWS CLI を使用したインスタンスとブロックストレージディスクの自動スナップショットの保持

AWS CLI を使用してインスタンスまたはブロックストレージディスクの自動スナップショットを保持するには、以下の手順を実行します。

1. ターミナルまたはコマンドプロントウィンドウを開きます。

   まだインストールしていない場合は、AWS CLI をインストールし、Lightsail と連携するように設定します。

2. 以下のコマンドを入力して、特定のリソースの使用可能な自動スナップショットの日付を取得します。自動スナップショットの日付は、後続のコマンドで `restore date` パラメータとして指定するために必要です。

   ```
   aws lightsail get-auto-snapshots --region Region --resource-name ResourceName
   ```

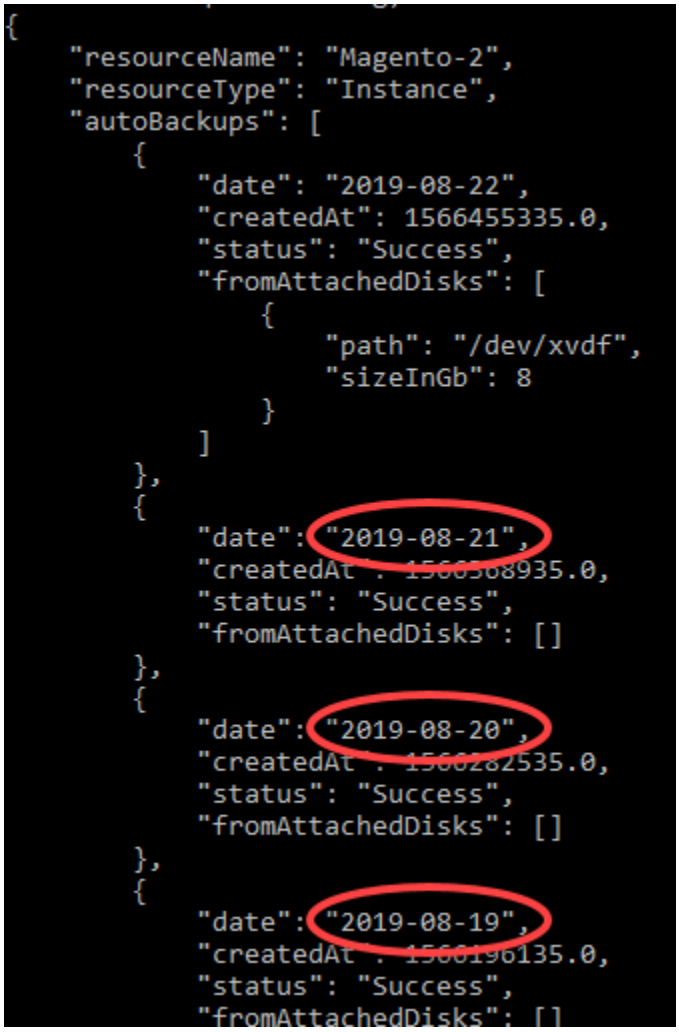
   コマンドを、以下のように置き換えます。

   - *Region* は、リソースが存在する AWS リージョン に置き換えます。
   - *ResourceName* は、リソースの名前に置き換えます。

   例:

   ```
   aws lightsail get-auto-snapshots --region us-west-2 --resource-
   name MyFirstWordPressWebsite01
   ```

   以下のような結果が表示され、使用可能な自動スナップショットが一覧表示されます。



3. 以下のコマンドを入力して、特定のリソースの自動スナップショットを保持します。

```
aws lightsail copy-snapshot --region TargetRegion --source-resource-
name ResourceName --restore-date YYYY-MM-DD --source-region SourceRegion --target-
snapshot-name SnapshotName
```

コマンドを、以下のように置き換えます。

- *TargetRegion* は、スナップショットのコピー先の AWS リージョン に置き換えます。
- *ResourceName* は、リソースの名前に置き換えます。
- *YYYY-MM-DD* は、前のコマンドで取得した使用可能な自動スナップショットの日付に置き換えます。
- *SourceRegion* は、自動スナップショットが現在存在する AWS リージョン に置き換えます。

- *SnapshotName* は、作成される新しいスナップショットの名前に置き換えます。

例:

```
aws lightsail copy-snapshot --region us-west-2 --source-resource-
name MyFirstWordPressWebsite01 --restore-date 2019-09-16 --source-region us-west-2
 --target-snapshot-name Snapshot-Copied-From-Auto-Snapshot
```

以下の例のような結果が表示されるはずです。

しばらくすると、自動スナップショットが手動スナップショットとしてコピーされます。手動スナップショットは、お客様が削除するまで保持されます。

> **Important**
>
> 自動スナップショットが不要になった場合は、削除することをお勧めします。そうしないと、Lightsailアカウントに保存された自動スナップショットと複製の手動スナップショットに対してスナップショットストレージの料金が課金されます。詳細については、「自動インスタンスのスナップショットを削除する」を参照してください。

> **Note**
>
> これらのコマンドでの GetAutoSnapshots と CopySnapshot API オペレーションに関する詳細については、LightsailAPI ドキュメントのGetAutoSnapshots および CopySnapshot を参照してください。

1 つの AWS リージョン から別のリージョンに Lightsail スナップショットをコピーする

Amazon Lightsail では、インスタンスのスナップショットやブロックストレージディスクのスナップショットを、2 つの AWS リージョン 間または同じリージョン内でコピーできます。例えば、もしあるリージョンで作成して設定したリソースが別のリージョンにより適していることが判明した場合、リージョン間でスナップショットをコピーすることができます。または、複数のリージョンを跨いでリソースを複製することもできます。このガイドでは、Lightsail スナップショットをコピーするプロセスについて説明します。

前提条件

コピーする Lightsail インスタンスまたはブロックストレージディスクのスナップショットを作成します。詳細については、以下のいずれかのガイドを参照してください。

- Linux または Unix インスタンスのスナップショットを作成する
- Windows Server インスタンスのスナップショットを作成する
- ブロックストレージディスクのスナップショットを作成する

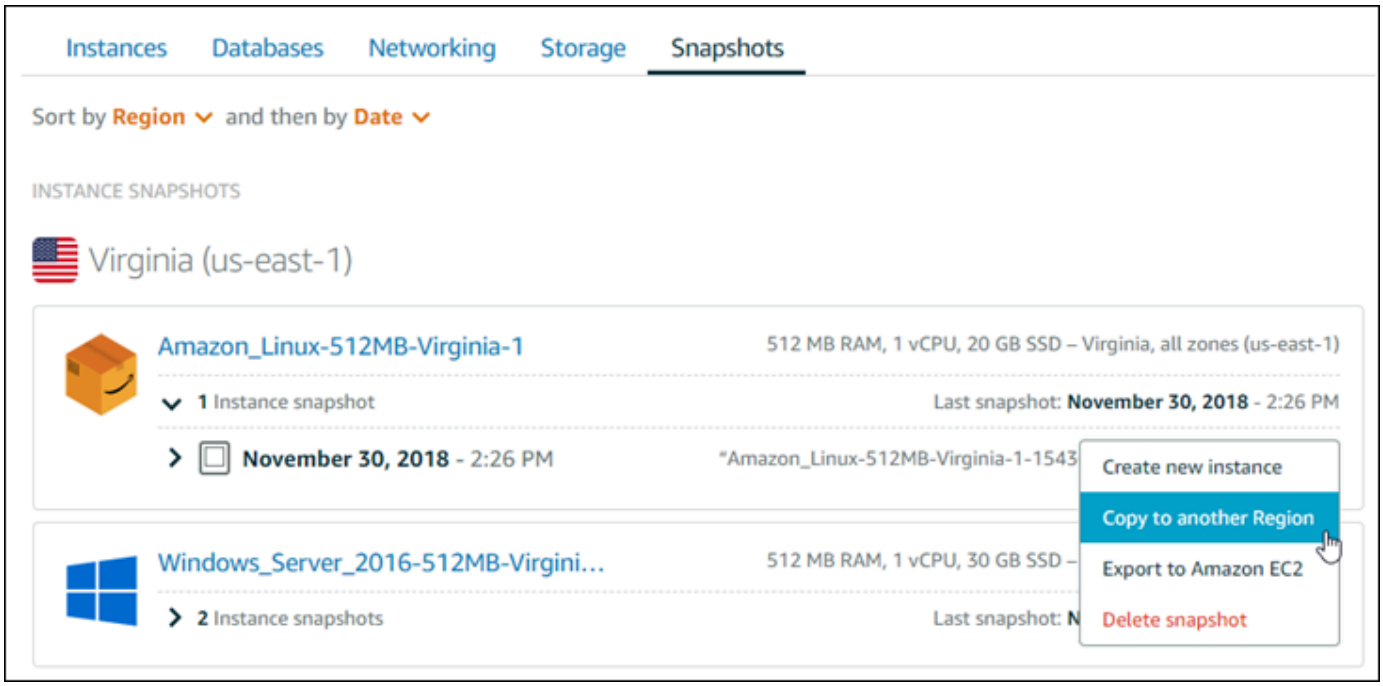
のスナップショットをコピーする

Lightsail インスタンスのスナップショットやブロックストレージディスクのスナップショットを、2 つの AWS リージョン 間または同じリージョン内でコピーできます。

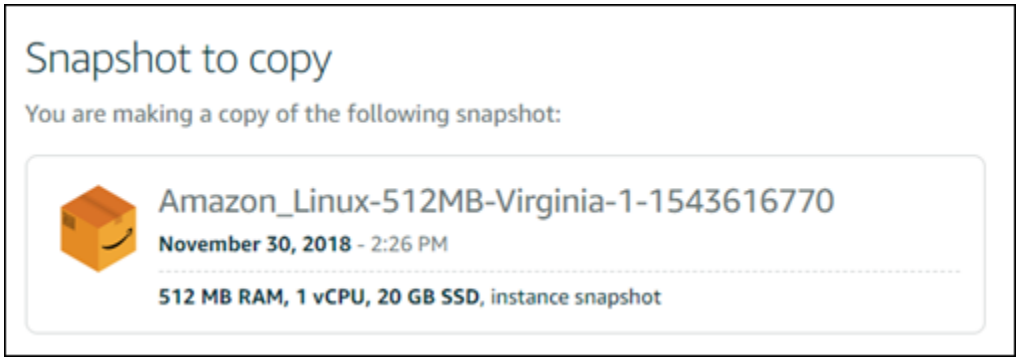
Lightsail スナップショットをコピーするには

1. Lightsail コンソールにサインインします。
2. Lightsail のホームページで [スナップショット] タブを選択します。

3. コピーするインスタンスまたはブロックストレージディスクを見つけてノードを展開し、そのリソースで使用可能なスナップショットを表示します。
4. 目的のスナップショットのアクションメニューアイコン (:) を選択し、[Copy to another Region (別のリージョンにコピー)] を選択します。



5. [Copy a snapshot (スナップショットをコピーする)] ページの [Snapshot to copy (コピーするスナップショット)] 部分で、表示されているスナップショットの詳細がコピー元のインスタンスやブロックストレージディスクの仕様と一致していることを確認します。



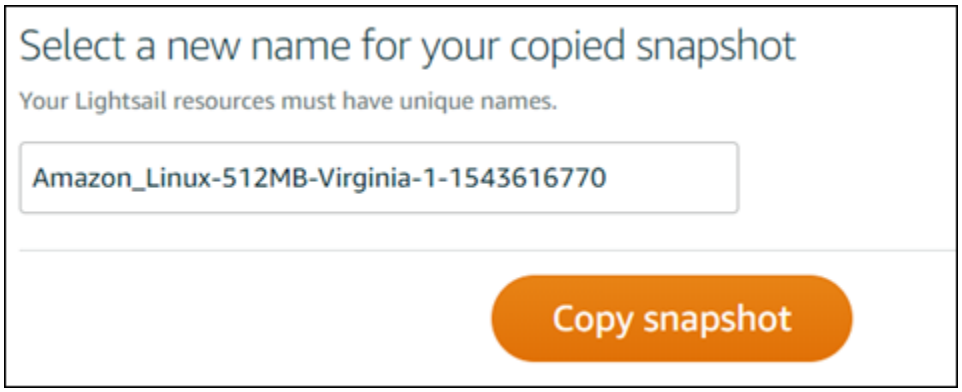
6. このページの [リージョンの選択] セクションで、スナップショットのコピー先のリージョンを選択します。
7. スナップショットコピーの名前を入力します。

   リソース名:

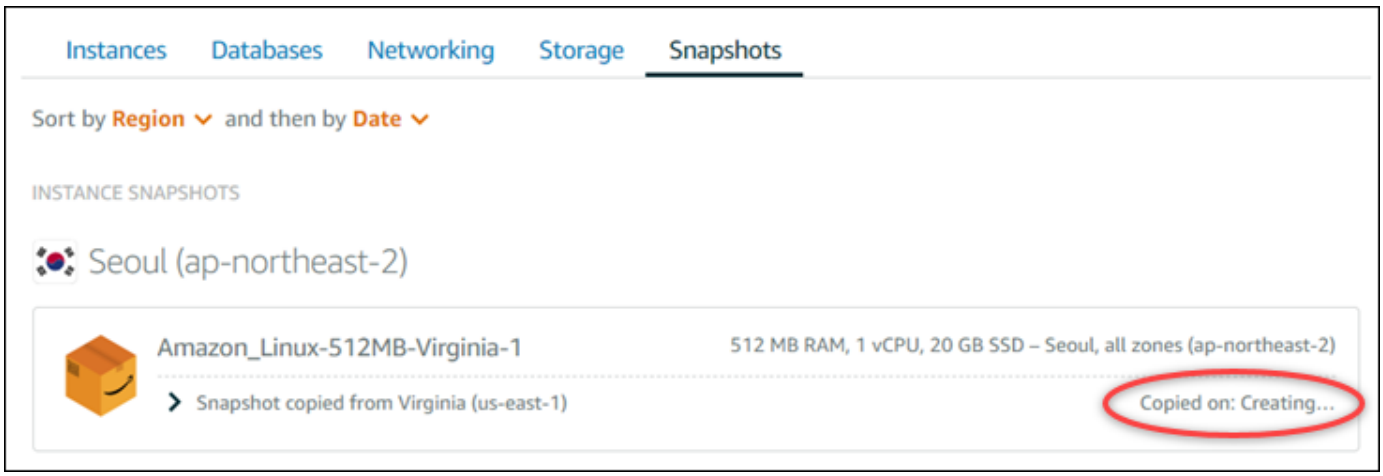
   - Lightsail アカウントの各 AWS リージョン 内で一意である必要があります。

- 2〜255 文字を使用する必要があります。
- 先頭と末尾は英数字または数字を使用する必要があります。
- 英数字、数字、ピリオド、ダッシュ、アンダースコアを使用できます。

8. [スナップショットのコピー] を選択します。



スナップショットのコピーはまもなく利用可能になります。所要時間は、ソースインスタンスのサイズと設定によります。スナップショットコピーのステータスは、次のスクリーンショットに示すように、Lightsail のホームページのスナップショットタブを開き、作成中のステータスのスナップショットを探すことで確認できます。スナップショットの準備が整うと、ステータスも応じてアップデートされます。



次のステップ

Lightsail の別のリージョンにスナップショットをコピーしたあと、追加で実行できるステップがいくつかあります。

- コピーしたスナップショットが利用可能になったら、このスナップショットから新しいインスタンスを作成します。詳細については、「スナップショットからインスタンスを作成する」を参照してください。
- コピー元のスナップショットが不要になった場合は、削除します。さもなければ、スナップショットの保存料金が発生します。

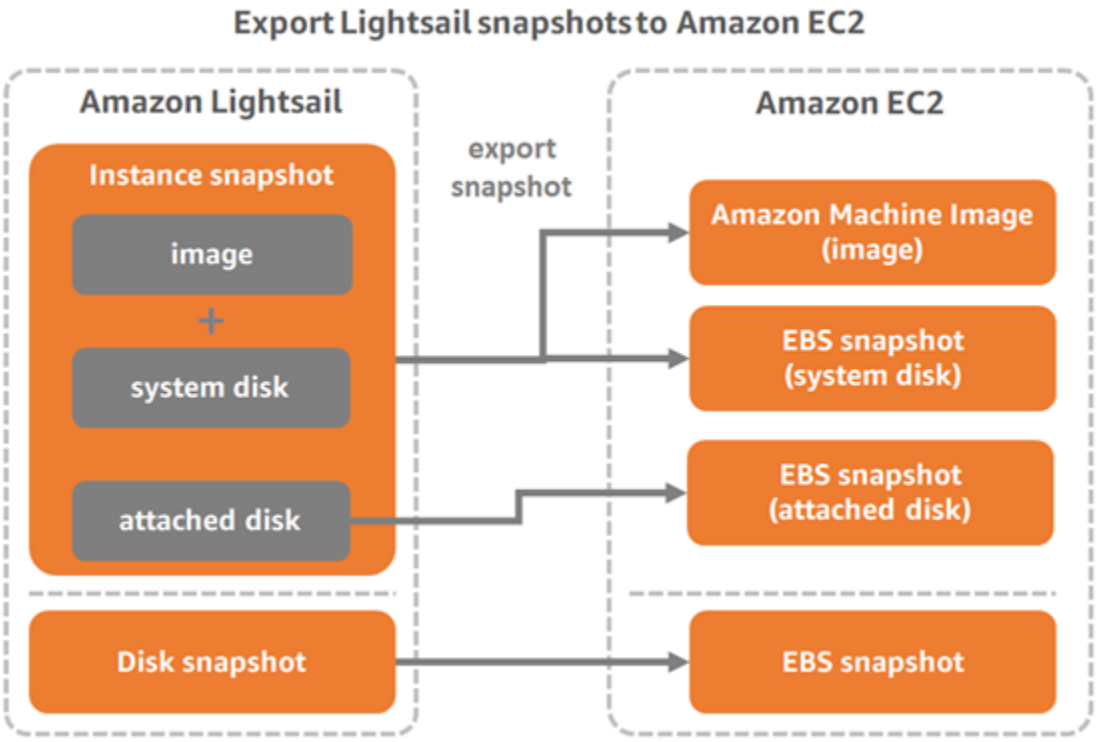
Lightsail スナップショットを Amazon EC2 にエクスポートする

Lightsail のインスタンスとブロックストレージディスクのスナップショットは、以下のいずれかの方法を使用して Amazon EC2 にエクスポートできます。

- Lightsail コンソール。詳細については、「スナップショットを Amazon EC2 にエクスポートする」を参照してください。
- Lightsail API、AWS Command Line Interface (AWS CLI)、または SDK。詳細については、「Lightsail API ドキュメント」の「ExportSnapshot オペレーション」または AWS CLI ドキュメントの「export-snapshot コマンド」を参照してください。

インスタンスおよびブロックストレージディスクのスナップショットをエクスポートできます。ただし、現時点では Django、Ghost、cPanel & WHM インスタンスのスナップショットはエクスポートできません。スナップショットは、同じ Lightsail の AWS リージョン から Amazon EC2 へとエクスポートされます。スナップショットを別のリージョンにエクスポートするには、まず Lightsail でスナップショットを別のリージョンにコピーしてからエクスポートを実行します。詳細については、「1 つの AWS リージョン から別のリージョンにスナップショットをコピーする」を参照してください。

Lightsail インスタンスのスナップショットをエクスポートすると、Amazon マシンイメージ (AMI) および Amazon Elastic Block Store (Amazon EBS) のスナップショットが Amazon EC2 に作成されます。これは、Lightsail インスタンスはイメージとシステムディスクで構成されていますが、Lightsail コンソールでは、これらが 1 つのインスタンスエンティティとしてまとめられ、管理が効率化されるためです。スナップショットの作成時に、ソースの Lightsail インスタンスに 1 つ以上のブロックストレージディスクがアタッチされている場合、アタッチされているディスクごとに追加の EBS スナップショットが Amazon EC2 に作成されます。Lightsail のブロックストレージディスクのスナップショットをエクスポートすると、1 つの EBS スナップショットが Amazon EC2 に作成されます。Amazon EC2 のすべてのエクスポートしたリソースには、Lightsail の対応するリソースとは異なる独自の一意な識別子が割り当てられます。



> **Note**
>
> Lightsail では、AWS Identity and Access Management (IAM) サービスにリンクされたロール (SLR) を使用してスナップショットを Amazon EC2 にエクスポートします。SLR の詳細については、「サービスにリンクされたロール」を参照してください。

エクスポートプロセスは時間がかかる場合があります。所要時間は、ソースのインスタンスやブロックストレージディスクのサイズと設定に応じて異なります。Lightsail コンソールのタスクモニターを使用して、エクスポートのステータスを追跡します。詳細については、「タスクモニター」を参照してください。

エクスポートした Lightsail スナップショットから Amazon EC2 リソースを作成する

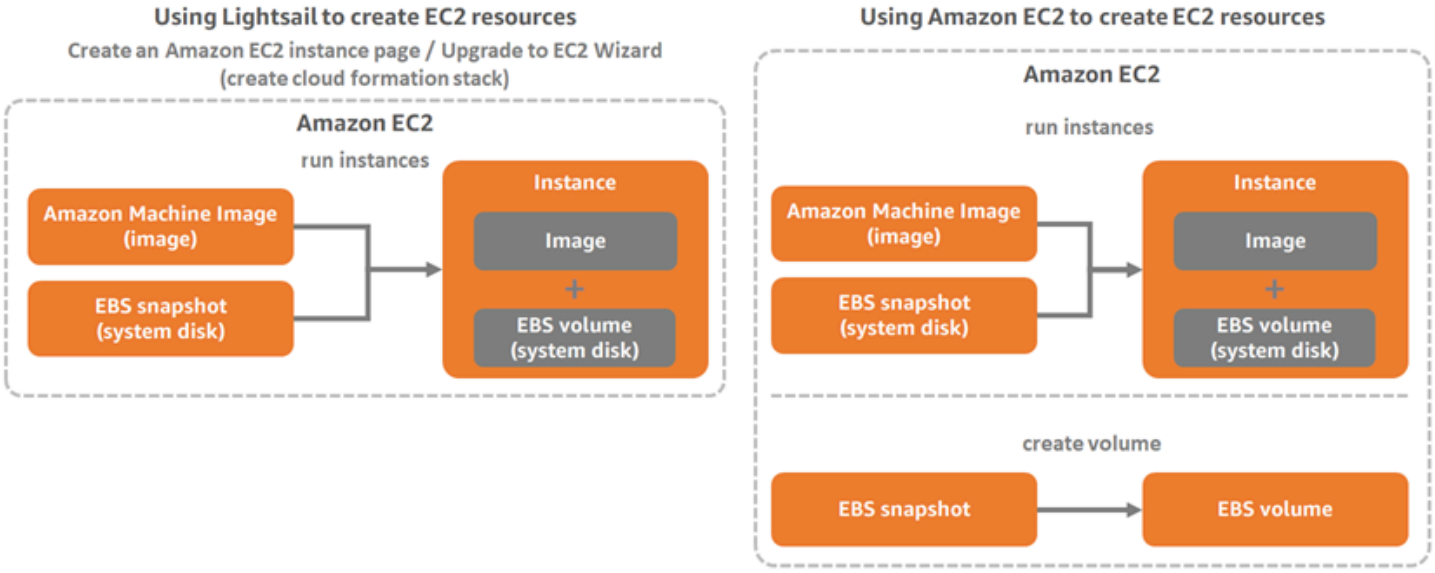
Lightsail スナップショットをエクスポートして、(AMI、EBS スナップショット、または両方として) Amazon EC2 で使用可能にした後は、以下のいずれかの方法を使用して、スナップショットから Amazon EC2 リソースを作成できます。

- Lightsail コンソールの「Amazon EC2 インスタンスを作成する」ページは、「Amazon EC2 へのアップグレードウィザード」とも呼ばれます。詳細については、「エクスポートしたスナップショットから Amazon EC2 インスタンスを作成する」を参照してください。
- Lightsail API、AWS CLI、または SDK。詳細については、「Lightsail API ドキュメント」の「CreateCloudFormationStack オペレーション」、または AWS CLI ドキュメントの「create-cloud-formation-stack コマンド」を参照してください。

> **Note**
>
> Lightsail では、エクスポートしたインスタンスから Amazon EC2 インスタンスを作成できますが、エクスポートしたブロックストレージディスクスナップショットから EBS ボリュームを作成することはできません。この目的には、Amazon EC2 コンソール、API、または AWS CLI を使用する必要があります。詳細については、「エクスポートしたディスクスナップショットから Amazon EBS ボリュームを作成する」を参照してください。

- Amazon EC2 コンソール、Amazon EC2 API、AWS CLI、または SDK。詳細については、Amazon EC2 ドキュメントの「インスタンス起動ウィザードを使用したインスタンスの起動」または「スナップショットからの Amazon EBS ボリュームの復元」を参照してください。

エクスポートしたインスタンススナップショット (AMI および EBS スナップショット) から Amazon EC2 インスタンスを作成すると、1 つの EC2 インスタンスが起動されます。Lightsail インスタンススナップショットのエクスポートで生成される AMI および EBS スナップショットは、相互に自動的にリンクされて EC2 インスタンスを形成します。Amazon EC2 で EBS ボリュームを作成するには、エクスポートした Lightsail ブロックストレージディスクのスナップショット (EBS スナップショット) を使用できます。



> **Note**
>
> Lightsail では、CloudFormation スタックを使用してインスタンスおよび関連リソースを EC2 に作成します。詳細については、「Lightsail の AWS CloudFormation スタック」を参照してください。

エクスポートしたスナップショットから Amazon EC2 リソースを作成するプロセスは時間がかかる場合があります。所要時間は、ソースインスタンスのサイズと設定によります。このタスクのステータスを追跡するには、Lightsail コンソールのタスクモニターを使用します。詳細については、「タスクモニター」を参照してください。

Amazon EC2 インスタンスタイプを選択する

Amazon EC2 が提供するインスタンスオプションは、Lightsail で使用可能なオプションよりも充実しています。Amazon EC2 では、コンピューティング (C5)、メモリ (R5)、または両方のバランス (T3 と M5) に最適化されたインスタンスタイプを選択できます。Lightsail では、[Amazon EC2 インスタンスの作成] ページにこれらのオプションがありますが、Amazon EC2 を使用して、エクスポートしたスナップショットから新しいインスタンスを作成する場合は、さらに多くのインスタンスタイプのオプションを利用できます。EC2 インスタンスタイプの詳細については、Amazon EC2 ドキュメントの「インスタンスタイプ」を参照してください。

エクスポートしたスナップショットから EC2 インスタンスを作成する前に、Lightsail と Amazon EC2 のインスタンス料金の違いを確認することが重要です。インスタンスの料金の詳細については、「Lightsail の料金」と「Amazon EC2 の料金」のページを参照してください。

Lightsail と Amazon EC2 のインスタンスタイプの互換性

一部の Lightsail インスタンスは、拡張ネットワーキングに対応していないため、現行世代の EC2 インスタンスタイプ (T3、M5、C5、または R5) と互換性がありません。ソースの Lightsail インスタンスに互換性がない場合は、エクスポートしたスナップショットから EC2 インスタンスを作成するときに、以前の世代のインスタンスタイプ (T2、M4、C4、または R4) から選択する必要があります。これらのオプションは、Lightsail コンソールの「Amazon EC2 インスタンスを作成する」ページを使用して、EC2 インスタンスを作成するときに表示されます。

ソースの Lightsail インスタンスに互換性がない場合に最新世代の EC2 インスタンスタイプを使用するには、まず以前の世代のインスタンスタイプ (T2、M4、C4、または R4) を使用して新しい EC2 インスタンスを作成し、ネットワーキングドライバーを更新します。次に、インスタンスを目的の現行世代のインスタンスタイプに更新します。詳細については、「Amazon EC2 インスタンスの拡張ネットワーキング」を参照してください。

Amazon EC2 インスタンスに接続する

Lightsail インスタンスに接続する場合と同様の方法で Amazon EC2 インスタンスに接続できます。つまり、Linux および Unix インスタンスには SSH を使用し、Windows Server インスタンスには RDP を使用します。ただし、Lightsail コンソールでブラウザベースの SSH/RDP クライアントを使用した場合、使用するブラウザのバージョンによっては、このクライアントを Amazon EC2 で使用できない場合があります。この場合は、独自の SSH/RDP クライアントを設定して EC2 インスタンスに接続する必要があります。詳細については、以下のガイドを参照してください。

- Lightsail スナップショットから作成された Amazon EC2 の Linux または Unix インスタンスに接続する
- Lightsail スナップショットから作成された Amazon EC2 の Windows Server インスタンスに接続する

Amazon EC2 インスタンスを保護する

エクスポートした Lightsail スナップショットから EC2 インスタンスを作成した後で、必要に応じて、いくつかのアクションを実行して新しいインスタンスのセキュリティを強化します。それらのアクションは、EC2 インスタンスのオペレーティングシステムによって異なります。

Amazon EC2 での Linux および Unix インスタンスの保護

EC2 (EC2 コンソール、EC2 API、EC2 用 AWS CLI、または EC2 用 SDK) を使用して、エクスポートしたスナップショットから Amazon EC2 の Linux または Unix インスタンスを作成すると、新しい

EC2 インスタンスには Lightsail サービスからの SSH キーが残る場合があります。これらのキーを削除して新しいインスタンスのセキュリティを強化することをお勧めします。

詳細については、「Lightsail スナップショットから作成した Amazon EC2 の Linux または Unix インスタンスを保護する」を参照してください。

Amazon EC2 の Windows Server インスタンスの保護

エクスポートしたスナップショットから Amazon EC2 の Windows Server インスタンスを作成すると、Lightsail と EC2 へのアクセス権がある AWS アカウントのすべてのユーザーは、ソースインスタンスに最初に割り当てられたデフォルトの管理者パスワードを取得できます。これは新しい EC2 インスタンスのパスワードにもなります。セキュリティを強化するために、Amazon EC2 インスタンスのデフォルトの管理者パスワードを変更することをお勧めします (まだ変更していない場合)。

詳細については、「Lightsail スナップショットから作成した Amazon EC2 の Windows Server インスタンスを保護する」を参照してください。

Lightsail スナップショットをエクスポートして Amazon EC2 でリソースを作成する

スナップショットのエクスポートを開始し、スナップショットから Amazon EC2 リソースを作成するには、以下のガイドを参照してください。

- タスクモニター
- Lightsail 用の AWS CloudFormation スタック
- スナップショットを Amazon EC2 にエクスポートする
- エクスポートしたスナップショットから Amazon EC2 インスタンスを作成する
- エクスポートしたディスクスナップショットから Amazon EBS ボリュームを作成する
- Amazon EC2 インスタンスの拡張ネットワーキング
- Lightsail スナップショットから作成された Amazon EC2 の Linux または Unix インスタンスに接続する
- Lightsail スナップショットから作成された Amazon EC2 の Windows Server インスタンスに接続する
- Lightsail スナップショットから作成された Amazon EC2 の Linux または Unix インスタンスを保護する
- Lightsail スナップショットから作成された Amazon EC2 の Windows Server インスタンスを保護する

- 1 つの AWS リージョン から別のリージョンにスナップショットをコピーする
- サービスにリンクされたロール

Lightsail のスナップショットを Amazon EC2 にエクスポートする方法

Amazon Lightsail インスタンスおよびブロックストレージディスクのスナップショットを Amazon Elastic Compute Cloud (Amazon EC2) にエクスポートできます。Lightsail インスタンスのスナップショットをエクスポートすると、Amazon マシンイメージ (AMI) および Amazon Elastic Block Store (Amazon EBS) のスナップショットが Amazon EC2 に作成されます。これは、Lightsail インスタンスはイメージとシステムディスクで構成されていますが、Lightsail コンソールでは、これらが 1 つのインスタンスエンティティとしてまとめられ、管理が効率化されるためです。スナップショットの作成時に、ソースの Lightsail インスタンスに 1 つ以上のブロックストレージディスクがアタッチされている場合、アタッチされているディスクごとに追加の EBS スナップショットが Amazon EC2 に作成されます。

Lightsail のブロックストレージディスクのスナップショットをエクスポートすると、1 つの EBS スナップショットが Amazon EC2 に作成されます。Amazon EC2 のすべてのエクスポートしたリソースには、Lightsail の対応するリソースとは異なる独自の一意な識別子が割り当てられます。

このガイドでは、Lightsail スナップショットをエクスポートする方法、エクスポートのステータスを追跡する方法、およびエクスポートしたスナップショットを Amazon EC2 で (AMI、EBS スナップショット、またはその両方として) 利用可能にした後のステップについて説明します。

> **Important**
>
> このガイドの手順を実行する前に、Lightsail のエクスポートプロセスを再確認することをお勧めします。詳細については、「スナップショットを Amazon EC2 にエクスポートする」を参照してください。

目次

Lightsail スナップショットをエクスポートするために必要なサービスにリンクされたロールおよび IAM アクセス許可

Lightsail では、AWS Identity and Access Management (IAM) サービスにリンクされたロール (SLR) を使用してスナップショットを Amazon EC2 にエクスポートします。SLR の詳細については、「サービスにリンクされたロール」を参照してください。

スナップショットのエクスポートを実行するユーザーによっては、必要に応じて以下の追加のアクセス許可を IAM で設定します。

- Amazon アカウントのルートユーザーがエクスポートを実行する場合は、このガイドの「前提条件」セクションに進みます。アカウントルートユーザーは、スナップショットのエクスポートを実行するために必要なアクセス許可をすでに持っています。
- IAM ユーザーがエクスポートを実行する場合は、AWS アカウント管理者が以下のポリシーをユーザーに追加する必要があります。ユーザーのアクセス権限を変更する方法については、IAM ドキュメント内の「IAM ユーザーのアクセス権限の変更」を参照してください。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "iam:CreateServiceLinkedRole",
            "Resource": "arn:aws:iam::*:role/aws-service-role/
lightsail.amazonaws.com/AWSServiceRoleForLightsail*",
            "Condition": {"StringLike": {"iam:AWSServiceName":
 "lightsail.amazonaws.com"}}
        },
        {
            "Effect": "Allow",
            "Action": "iam:PutRolePolicy",
            "Resource": "arn:aws:iam::*:role/aws-service-role/
lightsail.amazonaws.com/AWSServiceRoleForLightsail*"
        }
    ]
}
```

前提条件

Amazon EC2 にエクスポートする Lightsail インスタンスまたはブロックストレージディスクのスナップショットを作成します。詳細については、以下のいずれかのガイドを参照してください。

- Linux または Unix インスタンスのスナップショットを作成する
- Windows Server インスタンスのスナップショットを作成する
- ブロックストレージディスクのスナップショットを作成する

Amazon EC2 に Lightsail スナップショットをエクスポートする

スナップショットを Amazon EC2 にエクスポートする最も効率的な方法は、Lightsail コンソールを使用することです。スナップショットをエクスポートするには、Lightsail API、AWS Command Line Interface (AWS CLI)、または SDK を使用することもできます。詳細については、Lightsail API ドキュメントの「ExportSnapshot オペレーション」または AWS CLI ドキュメントの「export-snapshot コマンド」を参照してください。

> **Note**
>
> スナップショットは、同じ Lightsail の AWS リージョン から Amazon EC2 へとエクスポートされます。スナップショットを別のリージョンにエクスポートするには、まず Lightsail でスナップショットを別のリージョンにコピーしてからエクスポートを実行します。詳細については、「1 つの AWS リージョン から別のリージョンにスナップショットをコピーする」を参照してください。

Amazon EC2 に Lightsail スナップショットをエクスポートする

1. Lightsail コンソールにサインインします。
2. Lightsail のホームページで [スナップショット] タブを選択します。
3. エクスポートするインスタンスまたはブロックストレージディスクを見つけてノードを展開し、そのリソースの使用可能なスナップショットを表示します。
4. 目的のスナップショットの [アクション] メニューを選択し、[Amazon EC2 にエクスポート] を選択します。

> **Note**
>
> 現時点では、cPanel & WHM、Django、および Ghost インスタンスのスナップショットを Amazon EC2 にエクスポートすることはできません。

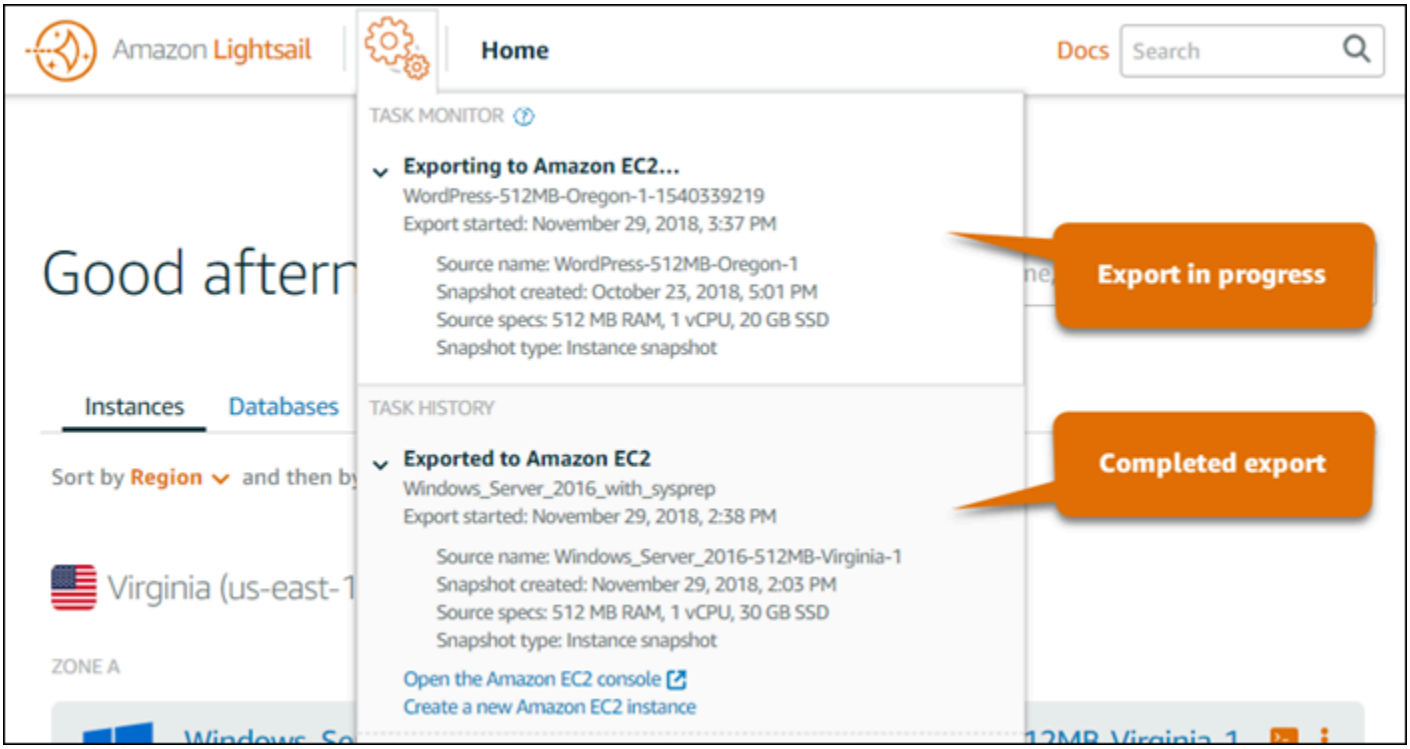
5. プロンプトに表示される重要な詳細情報を確認します。
6. Amazon EC2 にエクスポートすることに同意する場合は、[はい、続行します] を選択してプロセスを開始します。

   エクスポートプロセスは時間がかかる場合があります。所要時間は、ソースのインスタンスやブロックストレージディスクのサイズと設定に応じて異なります。このガイドの「エクスポートのステータスを追跡する」セクションに進み、エクスポートのステータスを追跡します。

タスクのステータスを追跡する

Lightsail コンソールのタスクモニターを使用して、エクスポートのステータスを追跡します。タスクモニターには、Lightsail コンソールの各ページで上部のナビゲーションペインからアクセスできます。詳細については、「タスクモニター」を参照してください。

タスクモニターには、スナップショットのエクスポートに関する以下の情報が表示されます。



- Snapshot name (スナップショット名) – ソースの Lightsail スナップショットの名前。
- Export started (エクスポートの開始日時) – スナップショットのエクスポートが開始された日付と時刻。
- Snapshot created (スナップショットの作成日時) – ソースの Lightsail スナップショットが作成された日付と時刻。
- Source specs (ソースの仕様) – ソースの Lightsail インスタンスの仕様 (メモリ、処理、ストレージなど)。
- Snapshot type (スナップショットのタイプ) – Lightsail スナップショットのタイプ。インスタンススナップショットまたはディスクスナップショットのいずれかになります。

タスクモニターには、完了したスナップショットのエクスポートに関する以下の情報が表示されます。

- スナップショットが Amazon EC2 に正常にエクスポートされた場合は、[エクスポート済み] と表示されます。
- 失敗 – スナップショットのエクスポート中に問題が発生した場合に表示されます。

スナップショットが正常にエクスポートされた場合は、完了したエクスポートに関する以下のオプションがタスクモニターに表示されます。

- [新しい Amazon EC2 インスタンスの作成] – Lightsail コンソールを使用して Amazon EC2 で新しいインスタンスを作成する場合は、このオプションを選択します。詳細については、「エクスポートしたスナップショットから Amazon EC2 インスタンスを作成する」を参照してください。
- [Amazon EC2 コンソールを開く] – Amazon EC2 コンソールを使用して、エクスポートしたスナップショットから新しい EC2 リソースを作成する場合に、このオプションを選択します。Lightsail ブロックストレージディスクのスナップショットをエクスポートした場合は、Amazon EC2 を使用してスナップショット (EBS スナップショット) から EBS ボリュームを作成する必要があります。詳細については、Amazon EC2 ドキュメントの「インスタンス起動ウィザードを使用したインスタンスの起動」または「スナップショットからの Amazon EBS ボリュームの復元」を参照してください。

> **Note**
>
> ソースの Lightsail スナップショットが不要になった場合は、これを削除します。そうしないと、スナップショットの保存料金が発生します。

エクスポートした Lightsail ディスクスナップショットから Amazon EBS ボリュームを作成する

Lightsail ブロックストレージディスクのスナップショットをエクスポートして Amazon EC2 で (EBS スナップショットとして) 使用可能にすると、Amazon EC2 コンソールを使用してスナップショットから EBS ボリュームを作成できます。

> **Note**
>
> エクスポートされたインスタンススナップショットから EC2 インスタンスを作成するには、「Lightsail でエクスポートしたスナップショットから Amazon EC2 インスタンスを作成する」を参照してください。

新しい EBS ボリュームは、Amazon EC2 API、AWS CLI、または SDK を使用して作成することもできます。詳細については、Amazon EC2 ドキュメントの「インスタンス起動ウィザードを使用してインスタンスを起動する」または「スナップショットからの Amazon EBS ボリュームの復元」を参照してください。

> **Important**
>
> このガイドの手順を実行する前に、Lightsail のエクスポートプロセスを再確認することをお勧めします。詳細については、「スナップショットを Amazon EC2 にエクスポートする」を参照してください。

前提条件

Lightsail ブロックストレージディスクのスナップショットを Amazon EC2 にエクスポートします。詳細については、「スナップショットを Amazon EC2 にエクスポートする」を参照してください。

エクスポートした Lightsail ブロックストレージディスクスナップショットから EBS ボリュームを作成する

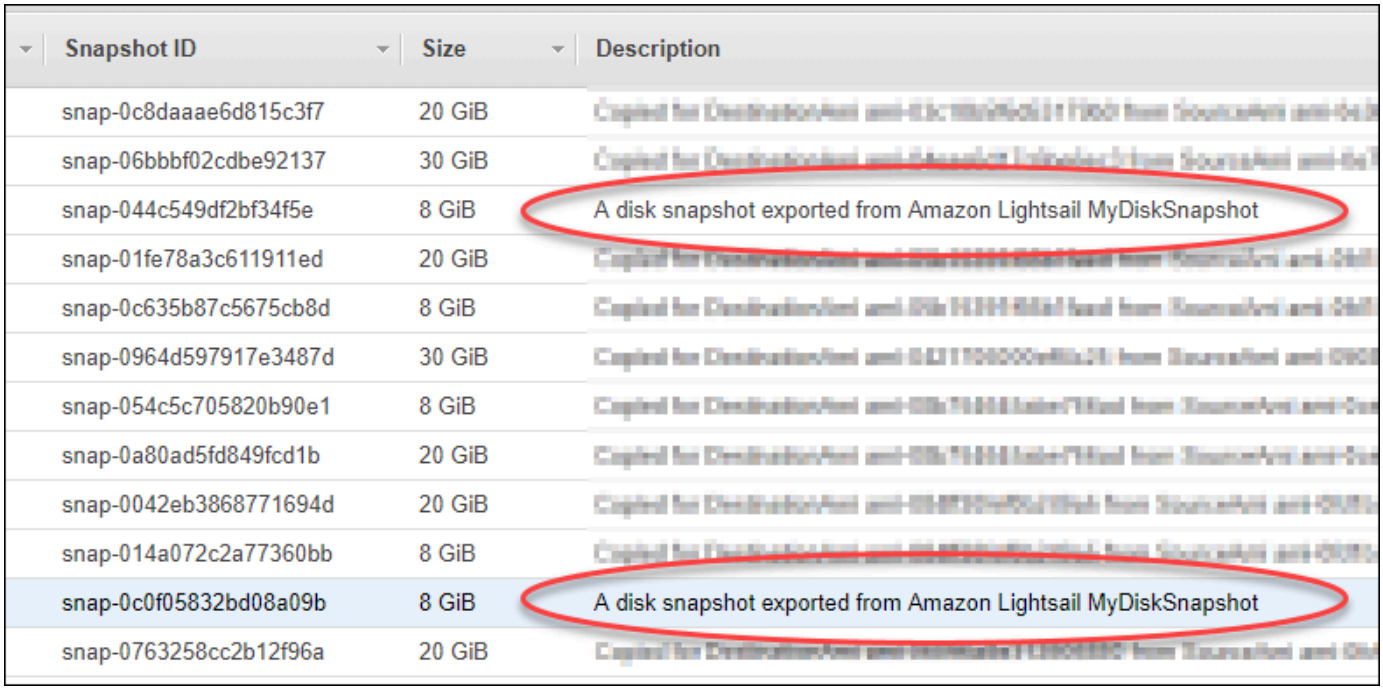
Amazon EC2 コンソールを使用して、エクスポートした Lightsail ブロックストレージディスクのスナップショットから新しい EBS ボリュームを作成します。

> **Note**
>
> これらの手順は Amazon EC2 のドキュメントにも記載されています。詳細については、Amazon EC2 のドキュメントで「スナップショットからの Amazon EBS ボリュームの復元」を参照してください。

エクスポートされた Lightsail ブロックストレージディスクスナップショットから EBS ボリュームを作成するには

1. Amazon EC2 コンソールにサインインします。
2. ナビゲーションバーから、スナップショットが存在するリージョンを選択します。
3. ナビゲーションペインで [Elastic Block Store (EBS)]、そして[Snapshots] の順に選択します。
4. エクスポートした Lightsail ブロックストレージディスクスナップショットを見つけて選択します。

   エクスポートしたディスクスナップショットは、次のスクリーンショットに示すように、EBS スナップショットの説明「A disk snapshot exported from Amazon Lightsail (からエクスポートされたディスクスナップショット)」で識別できます。



Snapshot ID	Size	Description
snap-0c8daaae6d815c3f7	20 GiB	
snap-06bbbf02cdbe92137	30 GiB	
snap-044c549df2bf34f5e	8 GiB	A disk snapshot exported from Amazon Lightsail MyDiskSnapshot
snap-01fe78a3c611911ed	20 GiB	
snap-0c635b87c5675cb8d	8 GiB	
snap-0964d597917e3487d	30 GiB	
snap-054c5c705820b90e1	8 GiB	
snap-0a80ad5fd849fcd1b	20 GiB	
snap-0042eb3868771694d	20 GiB	
snap-014a072c2a77360bb	8 GiB	
snap-0c0f05832bd08a09b	8 GiB	A disk snapshot exported from Amazon Lightsail MyDiskSnapshot
snap-0763258cc2b12f96a	20 GiB	

5. [アクション]、そして[ボリュームの作成] の順に選択します。
6. [ボリュームタイプ] ドロップダウンメニューからボリュームタイプを選択します。詳細については、Amazon EC2ドキュメントの「Amazon EBS ボリュームタイプ」を参照してください。
7. [Size (GiB)] に、ボリュームのサイズを入力するか、スナップショットのデフォルトサイズが適切であることを実証します。
8. プロビジョンド IOPS SSD ボリュームの場合は、[IOPS] に、ボリュームがサポートすべき 1 秒あたりの入力/出力オペレーション (IOPS) の最大数を入力します。
9. [Availability Zone] では、ボリュームを作成するアベイラビリティーゾーンを選択します。EBS ボリュームは、 EC2 インスタンスと同じアベイラビリティーゾーンに限りアタッチできます。
10. (オプション) [Create additional tags (追加のタグを作成)] を選択してボリュームにタグを追加します。タグごとに、タグキーとタグの値を指定します。
11. [Create Volume (ボリュームの作成)] を選択します。ボリュームが作成されると、そのボリュームは Amazon EC2 コンソールの [Elastic Block Store (EBS) > ボリューム] セクションにリストされます。

次のステップ

新しい Amazon EC2 インスタンスの作成後には以下の追加のステップを実行できます。

- スナップショットからボリュームを復元すると、インスタンスにアタッチして使用を開始できます。詳細については、Amazon EC2 ドキュメントの「インスタンスへの Amazon EBS ボリュームのアタッチ」を参照してください。
- スナップショットを、そのスナップショットのデフォルトよりも大きなボリュームに復元する場合、追加容量の利点を活用できるように、ボリュームのファイルシステムを拡張する必要があります。詳細については、Amazon EC2 ドキュメントの、「Linux での EBS ボリュームのサイズ、IOPS、またはタイプの変更」を参照してください。

エクスポートした Lightsail スナップショットから Amazon EC2 インスタンスを作成する

Lightsail インスタンスのスナップショットがエクスポートされ、(AMI および EBS スナップショットとして) Amazon EC2 で利用可能になると、Amazon Lightsail コンソールの [Amazon EC2 インスタンスの作成] ページ (「Amazon EC2 へのアップグレードウィザード」とも呼ばれます) を使用して、スナップショットから Amazon EC2 インスタンスを作成することができます。このウィザードでは、要件に一致する EC2 インスタンスタイプの選択、セキュリティグループのポートの設定、起動スクリプトの追加など、EC2 インスタンスの設定を行うことができます。Lightsail コンソールのウィザードにより、新しい EC2 インスタンスおよび関連リソースの作成プロセスが簡素化されます。

> **Note**
>
> エクスポートしたブロックストレージディスクのスナップショットから Amazon Elastic Block Store (Amazon EBS) ボリュームを作成する場合は、「エクスポートされたディスクのスナップショットから Amazon EBS ボリュームを作成する」を参照してください。

新しい EC2 インスタンスは、Lightsail API、AWS CLI、または SDK を使用して作成することもできます。詳細については、Lightsail API ドキュメントの「CreateCloudFormationStack オペレーション」、または AWS CLI ドキュメントの「Create-cloud-formation-stack コマンド」を参照してください。または、Amazon EC2 を使い慣れている場合は、EC2 コンソール、Amazon EC2 API、AWS CLI、または SDK を使用できます。詳細については、Amazon EC2 ドキュメントの「インスタンス起動ウィザードを使用したインスタンスの起動」または「スナップショットからの Amazon EBS ボリュームの復元」を参照してください。

> **Important**
>
> このガイドの手順を実行する前に、Lightsail のエクスポートプロセスを再確認することをお勧めします。詳細については、「スナップショットを Amazon EC2 にエクスポートする」を参照してください。

目次

Lightsail の AWS CloudFormationスタック

Lightsail では、AWS CloudFormation スタックを使用して EC2 インスタンスおよび関連リソースを作成します。Lightsail の CloudFormation スタックの詳細については、「Lightsail の AWS CloudFormation スタック」を参照してください。

「Amazon EC2 インスタンスの作成」ページで EC2 インスタンスを作成するユーザーによっては、以下の追加のアクセス許可を IAM で設定する必要があります。

- Amazon アカウントのルートユーザーが EC2 インスタンスを作成する場合は、このガイドの「前提条件」セクションに進みます。ルートユーザーは、Lightsail を使用して EC2 インスタンスを作成するために必要なアクセス許可をすでに持っています。
- IAM ユーザーが EC2 インスタンスを作成する場合は、AWS アカウント管理者が以下のアクセス許可をユーザーに追加する必要があります。ユーザーのアクセス権限を変更する方法については、IAM ドキュメント内の「IAM ユーザーのアクセス権限の変更」を参照してください。
  - ユーザーが Lightsail を使用して Amazon EC2 インスタンスを作成するには、以下のアクセス許可が必要です。

> **Note**
>
> 以下のアクセス許可により、CloudFormation スタックを作成できます。ただし、作成が失敗した場合は、ロールバックプロセスで追加のアクセス許可が必要になることがあります。アクセス許可が不足すると、残りのリソースは Amazon EC2 でロールバックされない可能性があります。このような場合は、AWS CloudFormation コンソールに移動して手動で EC2 リソースを削除できます。詳細については、「Lightsail の AWS CloudFormation スタック」を参照してください。

- ec2:DescribeAvailabilityZones
- ec2:DescribeSubnets
- ec2:DescribeRouteTables
- ec2:DescribeInternetGateways
- ec2:DescribeVpcs
- cloudformation:CreateStack
- cloudformation:ValidateTemplate
- iam:CreateServiceLinkedRole
- iam:PutRolePolicy

- ユーザーが EC2 インスタンスのセキュリティグループでポートを設定する場合は、以下のアクセス許可が必要になります。
  - ec2:DescribeSecurityGroups
  - ec2:CreateSecurityGroup
  - ec2:AuthorizeSecurityGroupIngress
- ユーザーが Amazon EC2 で Windows Server インスタンスを作成する場合は、以下のアクセス許可が必要です。
  - ec2:DescribeKeyPairs
  - ec2:ImportKeyPair
- ユーザーが Amazon EC2 インスタンスを初めて作成する場合や、仮想プライベートクラウド (VPC) の設定が完全に失敗した場合は、以下のアクセス許可が必要です。
  - ec2:AssociateRouteTable
  - ec2:AttachInternetGateway

- ec2:CreateInternetGateway
- ec2:CreateRoute
- ec2:CreateRouteTable
- ec2:CreateSubnet
- ec2:CreateVpc
- ec2:ModifySubnetAttribute
- ec2:ModifyVpcAttribute

前提条件

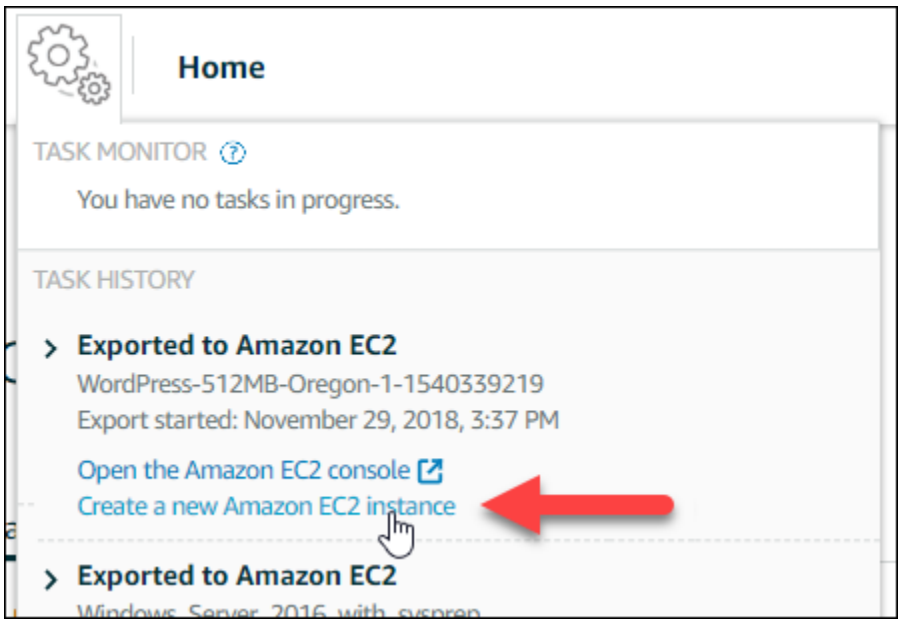
Lightsail インスタンスのスナップショットを Amazon EC2 にエクスポートします。詳細については、「スナップショットを Amazon EC2 にエクスポートする」を参照してください。

Lightsail コンソールの「Amazon EC2 インスタンスの作成」ページにアクセスする

Lightsail コンソールの「Amazon EC2 インスタンスの作成」ページは、インスタンスのスナップショットが EC2 に正常にエクスポートされた後にのみ、タスクモニターからアクセスできます。

Lightsail コンソールの「Amazon EC2 インスタンスの作成」ページにアクセスするには

1. Lightsail コンソールにサインインします。
2. 上部のナビゲーションペインでタスクモニターアイコンを選択します。
3. [タスク履歴] セクションでエクスポート完了済みのインスタンスのスナップショットを見つけて、[Amazon EC2 インスタンスの作成] を選択します。



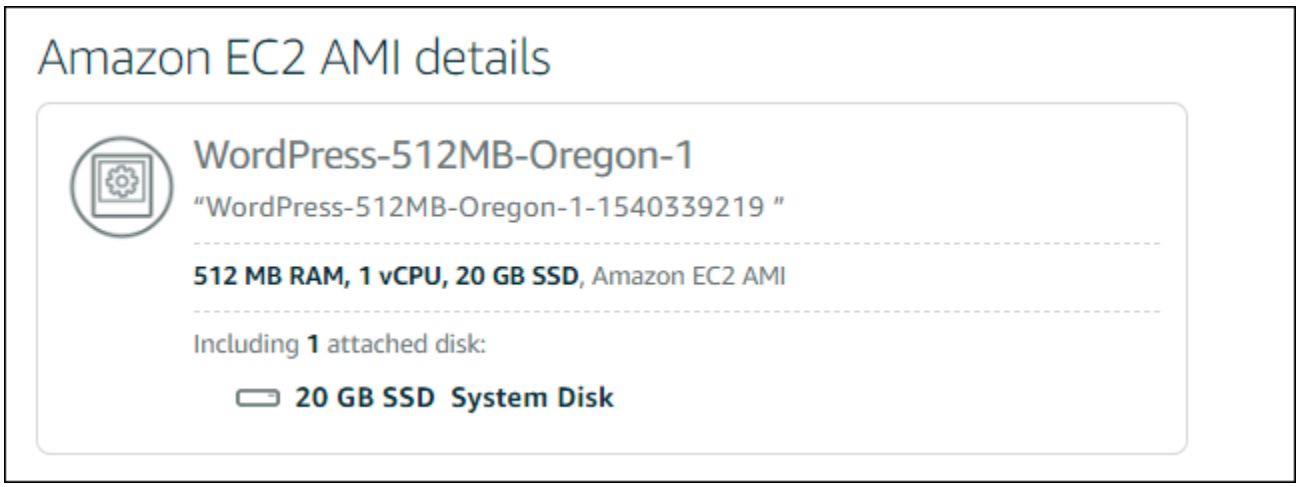
[Amazon EC2 インスタンスの作成] ページが表示されます。このガイドの次の「Amazon EC2 インスタンスの作成」セクションに進み、このページを使用して EC2 インスタンスを設定して作成する方法を確認します。

Amazon EC2 インスタンスを作成する

[Amazon EC2 インスタンスの作成] ページを使用して EC2 インスタンスを作成します。エクスポートされた Lightsail スナップショットから複数の EC2 インスタンスを作成するには、以下の手順を複数回繰り返します。ただし、各インスタンスが作成されるまで待ってから、次のインスタンスを作成します。

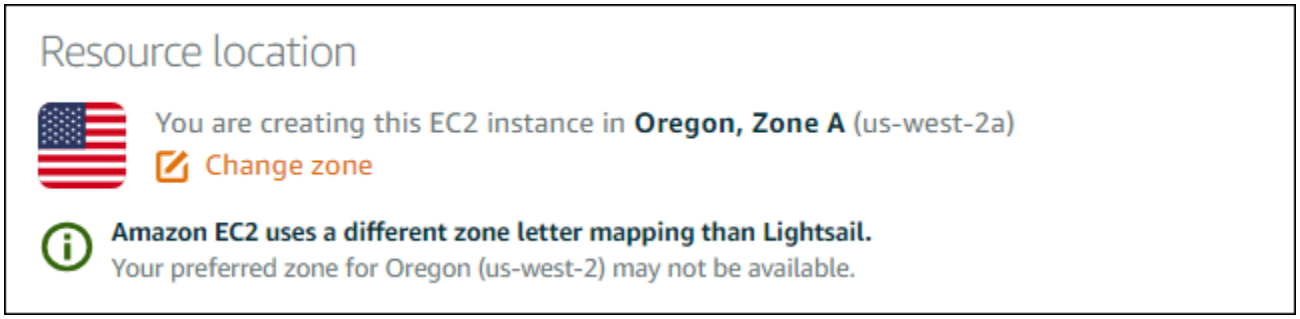
Amazon EC2 インスタンスを作成するには

1. ページの [Amazon EC2 AMI の詳細] セクションで、表示されている Amazon マシンイメージ (AMI) の詳細がソースの Lightsail インスタンスの仕様と一致していることを確認します。

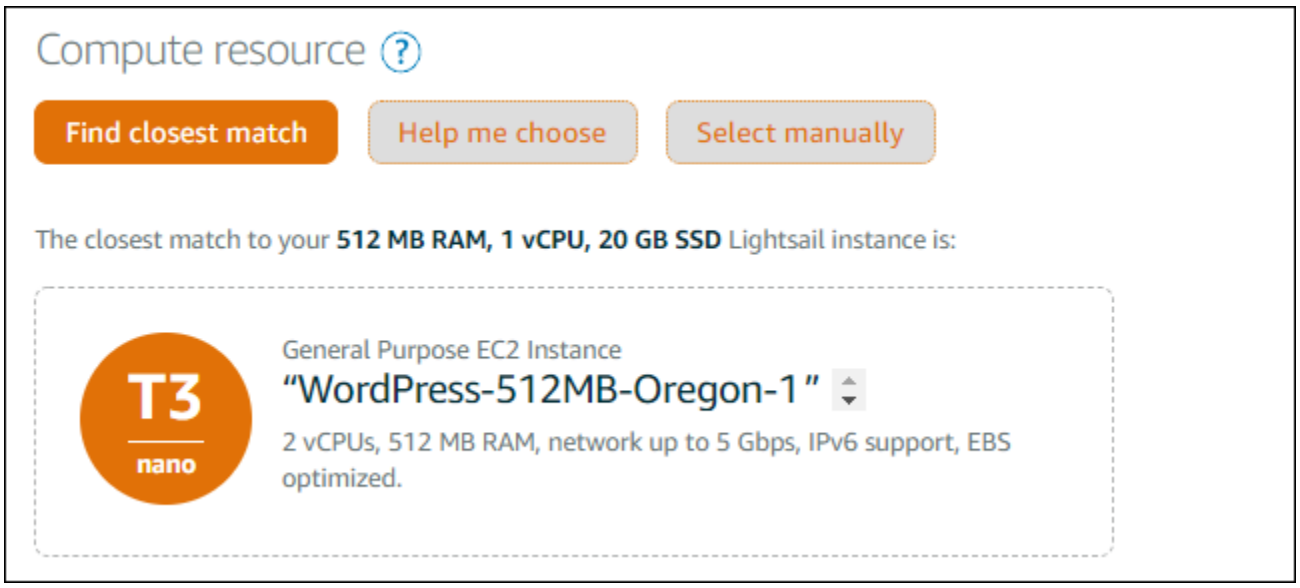


2. ページの [Resource location (リソースの場所)] セクションで、必要に応じてインスタンスのアベイラビリティーゾーンを変更します。ソースの Lightsail スナップショットと同じ AWS リージョン に Amazon EC2 リソースが作成されます。

> **Note**
>
> すべてのユーザーがすべてのアベイラビリティーゾーンを使用できるとは限りません。使用できないアベイラビリティーゾーンを選択すると、EC2 インスタンスの作成時にエラーが発生します。



3. ページの [Compute resource (コンピューティングリソース)] セクションで、次のいずれかのオプションを選択します。



a. [最も適切なものを選択する] は、ソースの Lightsail インスタンスの仕様と厳密に一致する Amazon EC2 インスタンスタイプが自動的に選択されます。

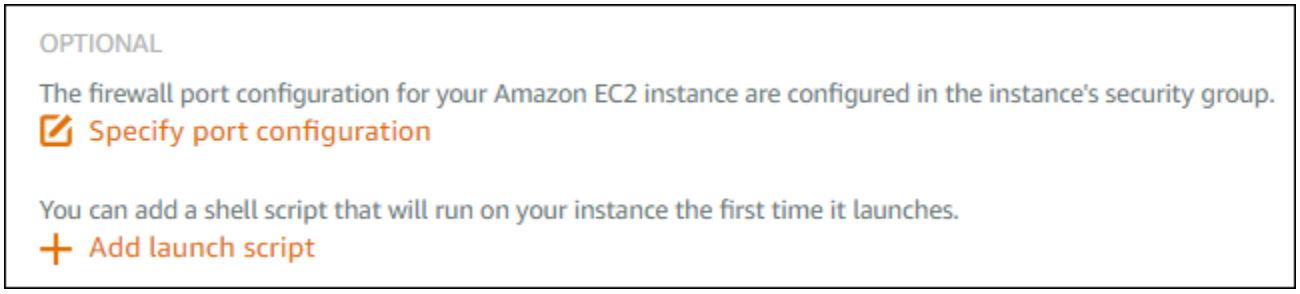
b. [選択のヘルプ] では、新しい Amazon EC2 インスタンスの仕様に関する簡単なアンケートに回答します。コンピューティングを最適化したインスタンスタイプ、メモリを最適化したインスタンスタイプ、または 2 つの間でバランスを取ったインスタンスタイプから選択できます。

c. [手動で選択] では、[Amazon EC2 インスタンスの作成] ページから利用可能なインスタンスタイプが一覧表示されます。

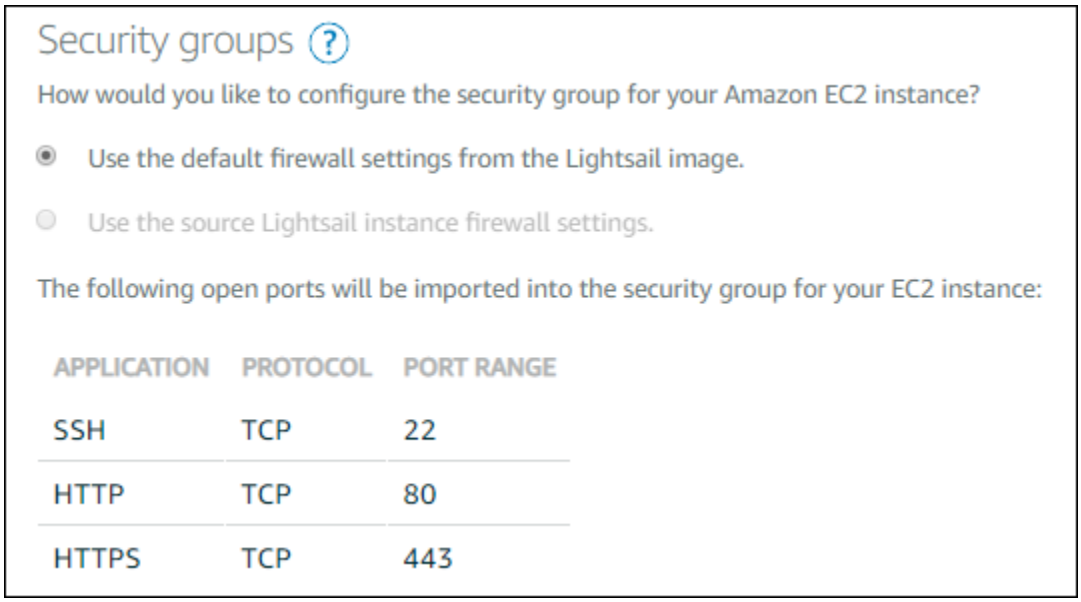
> **Note**
>
> 一部の Lightsail インスタンスは、拡張ネットワーキングに対応していないため、現行世代の EC2 インスタンスタイプ (T3、M5、C5、または R5) と互換性がありません。ソースの Lightsail インスタンスに互換性がない場合は、エクスポートしたスナップショットから EC2 インスタンスを作成するときに、以前の世代のインスタンスタイプ (T2、M4、C4、または R4) から選択する必要があります。これらのインスタンスタイプのオプションは、Lightsail コンソールの [Amazon EC2 インスタンスの作成] ページで表示されます。
>
> ソースの Lightsail インスタンスに互換性がない場合に最新世代の EC2 インスタンスタイプを使用するには、まず以前の世代のインスタンスタイプ (T2、M4、C4、または R4) を使用して新しい EC2 インスタンスを作成し、ネットワーキングドライバーを更新します。次に、インスタンスを目的の現行世代のインスタンスタイプに

更新します。詳細については、「拡張ネットワーキング用に Amazon EC2 インスタンスを更新する」を参照してください。

4. ページの [Optional (オプション)] セクションで以下の操作を行います。



a. [ポート設定の指定を] を選択して Amazon EC2 インスタンスのファイアウォール設定を選択し、次のいずれかのオプションを選択します。



i. Lightsail イメージのデフォルトのファイアウォール設定を使用し、新しい EC2 インスタンスでソースの Lightsail 設計図のデフォルトポートを設定します。Lightsail ブループリントのデフォルトポートの詳細については、「ファイアウォールとポート」を参照してください。

ii. ソースの Lightsail インスタンスのファイアウォール設定を使用し、新しい EC2 インスタンスでソースの Lightsail インスタンスのポートを設定します。このオプションは、ソースの Lightsail インスタンスがアクティブな場合にのみ使用できます。

b. ページの [起動スクリプト] セクションで、起動時に EC2 インスタンスを設定するスクリプトを追加する場合は、[起動スクリプトの追加] を選択します。

5. ページの [Connection security (接続セキュリティ)] セクションで、ソースの Lightsail インスタンスに接続した方法を確認します。これにより、適切な SSH キーを取得して、新しい EC2 インスタンスに接続します。ソースの Lightsail インスタンスへの接続方法としては以下が挙げられます。

   a. ソースインスタンスのリージョンにおけるデフォルトの Lightsail キーペアを使用する – EC2 インスタンスに接続するために、該当する AWS リージョン ンの一意なデフォルトの Lightsail キーをダウンロードして使用します。

      > **Note**
      >
      > Lightsail の Windows Server インスタンスでは、常にデフォルトの Lightsail キーペアが使用されます。

   b. 独自のキーペアを使用する – プライベートキーを見つけて EC2 インスタンスへの接続に使用します。

      > **Note**
      >
      > 個人のプライベートキーは Lightsail に保存されません。したがって、プライベートキーをダウンロードするオプションは提供されていません。プライベートキーが見つからない場合は、EC2 インスタンスに接続できません。

6. ページの [ストレージリソース] セクションで、作成する EBS ボリュームがソースの Lightsail インスタンスのシステムディスクおよびアタッチ済みブロックストレージディスクと一致することを確認します。

7. Lightsail の外部におけるリソースの作成に関する重要な詳細を確認します。
8. Amazon EC2 でインスタンスを作成することに同意する場合は、[EC2 でリソースを作成する] を選択します。

   インスタンスが作成されていることを Lightsail が確認し、AWS CloudFormation スタックに関する情報が表示されます。Lightsail では、EC2 インスタンスおよび関連リソースの作成に CloudFormation スタックが使用されます。詳細については、「Lightsail の AWS CloudFormation スタック」を参照してください。

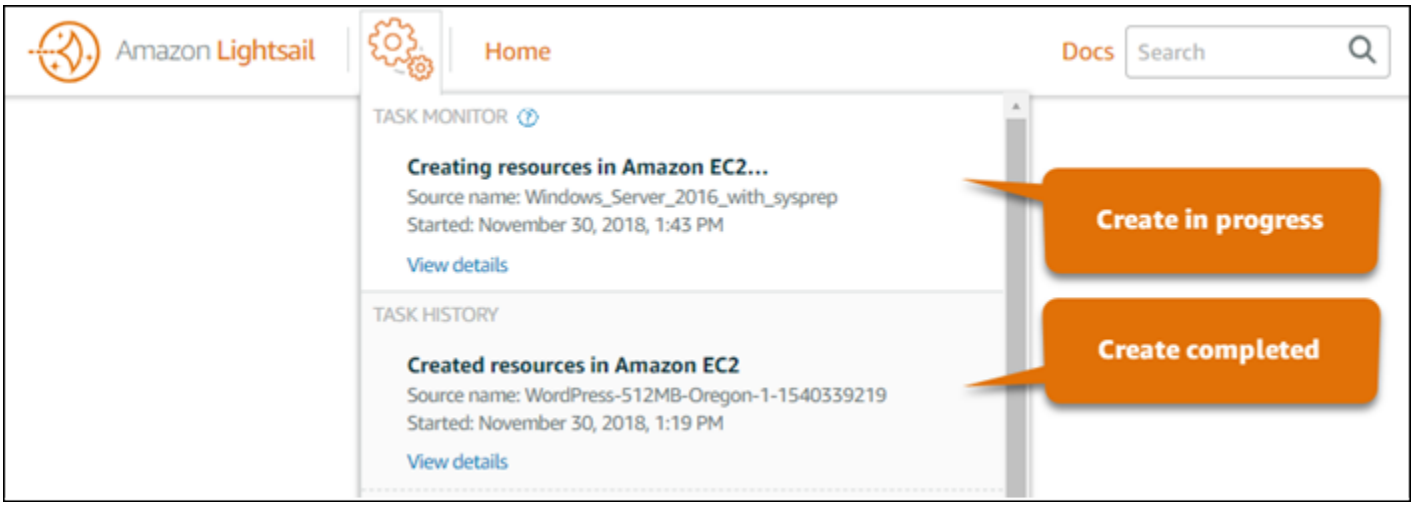
   このガイドの「新しい Amazon EC2 インスタンスのステータスを追跡する」セクションに進んで、新しい EC2 インスタンスのステータスを追跡します。

   > **Important**
   >
   > 新しい EC2 インスタンスが作成されるまで待ってから、同じエクスポートしたスナップショットから別の EC2 インスタンスを作成します。

新しい Amazon EC2 インスタンスのステータスを追跡する

Lightsail コンソールのタスクモニターを使用して、新しい EC2 インスタンスのステータスを追跡します。タスクモニターには、Lightsail コンソールの各ページで上部のナビゲーションペインからアクセスできます。詳細については、「タスクモニター」を参照してください。

タスクモニターには、作成中の EC2 インスタンスに関する以下の情報が表示されます。



- Source name (ソース名) – ソースの Lightsail スナップショットの名前。
- Started (開始日時) – 作成リクエストが開始された日付と時刻。

タスクモニターには、作成済みの EC2 インスタンスに関する以下の情報が表示されます。

- 作成済み – Amazon EC2 リソースが正常に作成された場合に表示されます。新しい EC2 インスタンスの準備が完了したら、このガイドの「次のステップ」セクションに進み、以降のステップを確認します。
- 失敗 – EC2 インスタンスの作成中に問題が発生した場合に表示されます。

次のステップ

Amazon EC2 インスタンスの作成後に、以下の追加のステップを実行できます。

- Lightsail インスタンスに接続する場合と同様の方法で Amazon EC2 インスタンスに接続できます。つまり、Linux および Unix インスタンスには SSH を使用し、Windows Server インスタンスには RDP を使用します。ただし、Lightsail コンソールでブラウザベースの SSH/RDP クライアントを使用した場合、使用するブラウザのバージョンによっては、このクライアントを Amazon EC2 で使用できない場合があります。この場合は、独自の SSH/RDP クライアントを設定して

EC2 インスタンスに接続する必要があります。詳細については、以下のガイドを参照してください。

- Lightsail スナップショットから作成された Amazon EC2 の Linux または Unix インスタンスに接続する
- Lightsail スナップショットから作成された Amazon EC2 の Windows Server インスタンスに接続する

- Lightsail スナップショットから作成した Amazon EC2 の Linux または Unix インスタンスには、Lightsail からの SSH キーが残っている場合があります。これらのキーを削除して EC2 インスタンスのセキュリティを強化することをお勧めします。詳細については、「Lightsail スナップショットから作成した Amazon EC2 の Linux または Unix インスタンスを保護する」を参照してください。

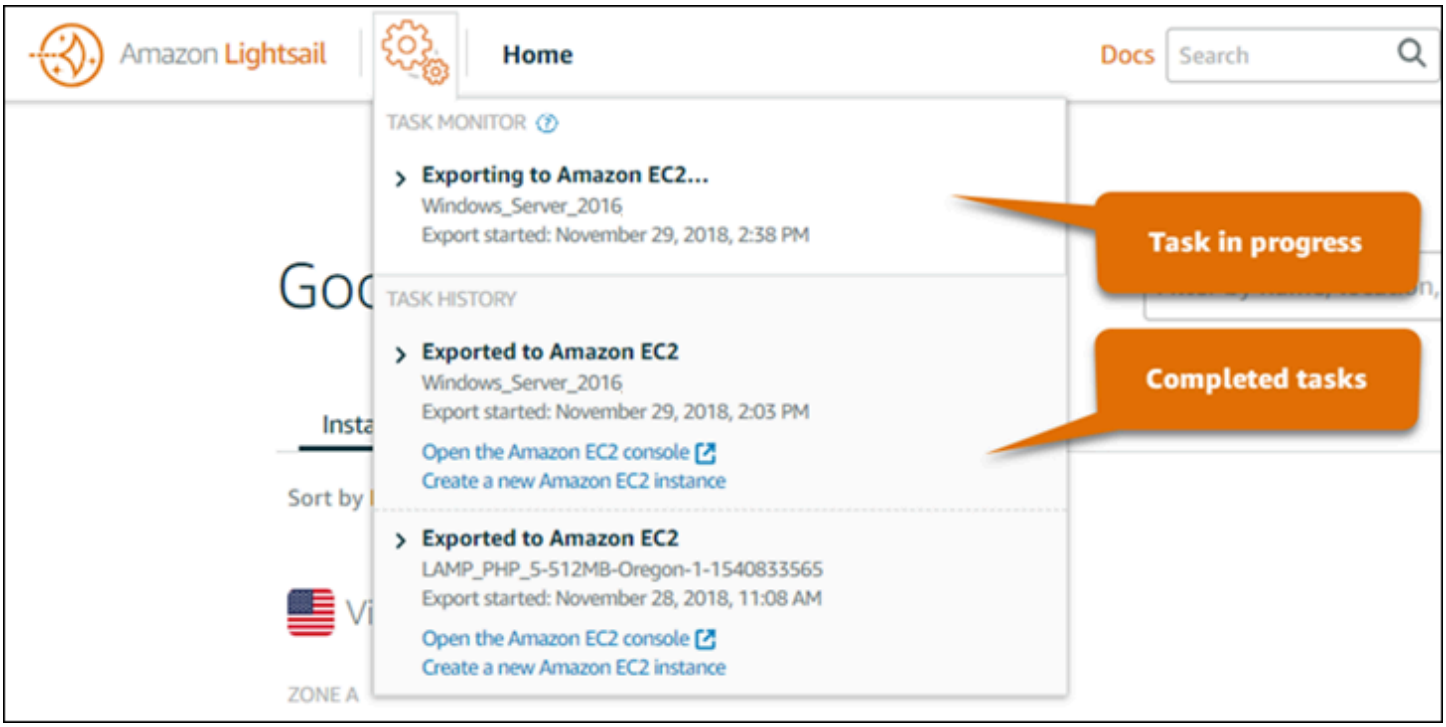
EC2 インスタンスの作成後に、ソースの Lightsail インスタンスと同じ設定にするために、必要に応じて、さらにいくつかのステップを実行する場合があります。EC2 インスタンスを設定する追加のステップは以下のとおりです。

- Amazon EC2 インスタンスのセキュリティグループを編集してファイアウォール設定を構成します。詳細については、 の「Linux インスタンス用の Amazon EC2 セキュリティグループ」または Amazon EC2 ドキュメントの「Windows インスタンス用の Amazon EC2 セキュリティグループ」を参照してください。
- Lightsail の静的 IP を作成して Lightsail インスタンスにアタッチした場合は、Elastic IP を作成して Amazon EC2 インスタンスにアタッチする必要があります。詳細については、Amazon EC2 ドキュメント内の「Elastic IP アドレス」を参照してください。
- Lightsail の DNS ゾーンを作成して Lightsail インスタンスのドメインを設定した場合は、Amazon Route 53 の DNS ゾーンを作成してドメインの DNS を管理し、新しい Amazon EC2 インスタンスをポイントするようにドメインを設定する必要があります。詳細については、Amazon Route 53 ドキュメント内の「Amazon Route 53 をDNSサービスとして設定し、Amazon Route 53 を既存ドメインの DNS サービスにする」を参照してください。
- Lightsail ロードバランサーを作成して Lightsail インスタンス用に設定した場合は、Amazon EC2 インスタンスの Application Load Balancer を設定する必要があります。詳細については、Elastic Load Balancing ドキュメント内の「Application Load Balancer の使用開始」を参照してください。
- Lightsail データベースは Amazon EC2 インスタンスからアクセスできません。Amazon EC2 にエクスポートした Lightsail インスタンスが Lightsail データベースに接続されている場合、このデータベースのデータに新しい Amazon EC2 インスタンスからアクセスするには、このデータベース

を手動で Amazon Relational Database Service (Amazon RDS) に移行する必要があります。詳細については、「わずかなダウンタイムでの Amazon RDS MySQL または MariaDB DB インスタンスへのデータのインポート」および「Amazon RDS DB インスタンスへの接続」を参照してください。

Lightsail コンソールタスクモニター

Amazon Lightsail コンソールのタスクモニターでは、Amazon EC2 への Lightsail スナップショットのエクスポートや、エクスポートしたインタンススナップショットからの新しい EC2 インスタンスの作成のステータスを追跡します。これらのタスクの所要時間は、ソースインスタンスやブロックストレージディスクのサイズと設定に応じて異なります。タスクモニターには、最新の進行中または完了済みの 20 個のタスクが表示されます。タスクモニターには、Lightsail コンソールの各ページで上部のナビゲーションペインからアクセスできます。タスクモニターのアイコンは、タスクの進行中はオレンジ色になり、すべてのタスクが完了済みになると灰色になります。



Amazon EC2 への Lightsail スナップショットのエクスポートや、エクスポートしたスナップショットからの EC2 インスタンスの作成の詳細については、以下のガイドを参照してください。

- スナップショットを Amazon EC2 にエクスポートする
- エクスポートしたスナップショットから Amazon EC2 インスタンスを作成する

Amazon Lightsail でのドメイン登録

ウェブサイトには example.com などの名前が必要です。Amazon Lightsail では、ウェブサイトの名前 (ドメイン名) を登録できます。ウェブサイトにアクセスするには、ウェブブラウザにドメイン名を入力します。

Amazon Lightsail コンソールの [ドメインと DNS] タブを使用して、ドメイン名を登録および管理します。Lightsail は、可用性と拡張性に優れたドメインネームシステム (DNS) ウェブサービスである Amazon Route 53 を使用し、ドメインを登録します。ドメインが登録されたら、Lightsail リソースに割り当てたり、DNS レコードを管理したりできます。DNS の一般的な情報については、「DNS」を参照してください。

Amazon Lightsail でのドメイン登録の詳細については、この先を参照してください。

目次

ドメイン登録の仕組み

Amazon Lightsail でドメイン名を登録する方法の概要を次に示します。

1. 目的のドメイン名がインターネットで使用できることを確認します。希望するドメイン名が使用できない場合は、他の名前を試したり、.com などの最上位ドメインのみを .org や .net などの別の最上位ドメインに変更したりできます。Lightsail でサポートされている最上位ドメイン (TLD) の一覧については、「Amazon Lightsail で登録できるドメイン」を参照してください。
2. Lightsail を使用してドメイン名を登録します。ドメインを登録するときは、ドメインの所有者の名前と連絡先情報、その他の連絡先の名前とその情報を提供します。

登録プロセスの最後に、お客様から提供された情報がドメインのレジストラに送信されます。ドメインレジストラは、特定の TLD のドメイン登録を処理する ICANN (Internet Corporation for Assigned Names and Numbers) から認定を受けている会社です。ドメインのレジストラは、Amazon Registrar か、当社のレジストラアソシエイトである Gandi のいずれかです。

Amazon Registrar と Gandi では、デフォルトで非表示になる情報が異なります。Amazon Registrar, Inc. はお客様の連絡先情報をすべて非表示にし、Gandi は組織名を除くすべての連絡先情報を非表示にします。

- ドメインのレジストラを特定するには、「Amazon Lightsail に登録できるドメイン」を参照してください。
- レジストラはお客様の情報をドメインのレジストリに送信します。レジストリとは、.com などの 1 つまたは複数の最上位ドメインのドメイン登録を販売する会社です。
- レジストリは、お客様のドメインに関する情報を自社のデータベースに保存し、その情報の一部をパブリック WHOIS データベースにも保存します。

ドメイン名を登録する方法の詳細については、「新しいドメインを登録する」を参照してください。

Lightsail を使用してドメインを登録すると、Route 53 はネームサーバーのセットをドメインに割り当てることにより、ドメインの DNS サービスになります。ネームサーバーとは、ドメイン名を IP アドレスに変換するのに役立つサーバーです。

Lightsail は自動的に次の処理を実行して、ドメインの DNS サービスになります。

- ドメインと同じ名前の Lightsail DNS ゾーンを作成します。
- 4 つの一連のネームサーバーを Lightsail DNS ゾーンに割り当てます。
- ドメインの Route 53 ネームサーバーを Lightsail DNS ゾーンのネームサーバーに置き換えます。

別のレジストラにドメイン名を既に登録している場合は、ドメインの DNS の管理を Lightsail に移管するように選択できます。この操作は、Lightsail の他の機能は使用する場合は不要です。詳細については、「DNS ゾーンを作成してドメインの DNS レコードを管理する」を参照してください。

Lightsail で登録できるドメイン

Lightsail は、Route 53 と同じ汎用最上位ドメイン (TLD) を使用します。Lightsail でドメインの登録に使用できる汎用 TLD の一覧については、Amazon Route 53 デベロッパーガイドの「Amazon Route 53 に登録できるドメイン」を参照してください。

TLD がリストに含まれていない場合、または地理的ドメインを登録する場合は、Route 53 コンソールを使用することをお勧めします。地理的ドメインは、Route 53 を使用して登録してから Lightsail コンソールで使用できるようになります。詳細については、「Amazon Route 53 デベロッパーガイド」の「地理的最上位ドメイン」を参照してください。

ドメイン登録の料金

Lightsail では、ドメイン登録には Route 53 が使用されます。したがって、Route 53 の料金は Lightsail 登録にも適用されます。

ドメイン登録のコストの詳細については、Amazon Route 53 デベロッパーガイドの「Amazon Route 53 に登録できるドメイン」を参照してください。

ドメインに関する追加情報

次の記事は、Lightsail でのドメインの管理に役立ちます。

- DNS
- ドメイン名をフォーマットする
- Amazon Route 53 で Lightsail ドメインを管理する
- ドメインの DNS レコードを管理する DNS ゾーンを作成する
- ドメイン登録更新
- DNS ゾーンを編集または削除する
- ドメインをロードバランサーにポイントする
- ドメインをディストリビューションにポイントする
- ドメインをインスタンスにポイントする
- ドメインへのトラフィックをコンテナサービスにルーティングする

Amazon Lightsail の DNS

ユーザーは、インスタンスのパブリックインターネットプロトコル (IP) アドレス (IPv4 または IP6 アドレス) にアクセスすることで Lightsail インスタンス上のウェブアプリケーションにアクセスできます。ただし、IP アドレスは複雑で覚えにくいという欠点があります。そのため、インスタンス上のウェブアプリケーションにアクセスするためなど`example.com`、 easy-to-remember ユーザーにドメイン名を参照してもらう必要があります。そのためには、ドメインネームシステム (DNS) を使用します。DNS は、登録されたドメイン名を IP アドレスにマッピングするディレクトリとして機能します。

ドメイン名のトラフィックを Lightsail インスタンスにルーティングするには、ドメイン名をインスタンスの静的 IPv4 アドレスを指すアドレス (A) レコード、またはインスタンスの IPv6 アドレスを指

す AAAA レコードを追加します。Lightsail を使用してドメイン名を登録した場合、ドメイン名を登録したときに作成された DNS ゾーンから DNS レコードを管理できます。ドメインが別のレジストラを通じて登録された場合は、そのレジストラで DNS レコードを管理するか、ドメインの DNS の管理を Lightsail に移管できます。

ドメイン名を Lightsail インスタンスに簡単にマッピングできるように、DNS ゾーンを作成してドメインの DNS レコードの管理を Lightsail に移管することをお勧めします。詳細については、「DNS ゾーンを作成してドメインの DNS レコードを管理する」を参照してください。Lightsail では最大 6 つの DNS ゾーンを作成できます。6 つ以上の DNS ゾーンが必要な場合は、すべてのドメインでの DNS 管理に、 Route 53 を使用することをお勧めします。Route 53 を使用して、ドメイン名を Lightsail インスタンスに向けることができます。Route 53 による DNS の管理の詳細については、「Amazon Route 53 を使用してドメインをインスタンスに参照させる」を参照してください。

DNS の用語

ドメインの DNS を管理するには、以下の用語を理解しておく必要があります。

Apex ドメイン/ルートドメイン

apex ドメイン (ルートドメインとも呼ばれます) は、サブドメインパートを含まないドメインです。apex ドメインの例は `example.com` です。サブドメインの例は `www.example.com` や `blog.example.com`です。これらがサブドメインであるのは、それぞれサブドメインパートとして `www` と `blog` を含んでいるためです。

ドメインネームシステム (DNS)

DNS は`example.com`、 easy-to-remember などのドメイン名をウェブサーバーの IP アドレスにルーティングします。

詳細については、Wikipedia の「Domain Name System」を参照してください。

DNS レコード

DNS レコードはマッピングパラメータです。DNS サーバーに対して、ドメインやサブドメインに対応する IP アドレスやホスト名を示します。

詳細については、Wikipedia の「DNS レコードタイプの一覧」を参照してください。

DNS ゾーン

DNS ゾーンは、特定のドメイン (`example.com` など) やそのサブドメイン (`blog.example.com` など) のトラフィックをインターネットでルーティングする方法に関する情報を保持するコンテナです。

詳細については、Wikipedia の「DNS ゾーン」を参照してください。

ドメイン名レジストラ

ドメイン名レジストラ (ドメイン名プロバイダーとも呼ばれます) は、ドメイン名の割り当てを管理する企業または組織です。Lightsail、Amazon Route 53、またはその他のドメイン名レジストラを使用して、ドメインを購入したり、既存のドメインを管理したりできます。

詳細については、Wikipedia の「Domain name registrar」を参照してください。

ネームサーバー

ネームサーバーは、トラフィックをドメインにルーティングします。Lightsail では、 easy-to-remember ネームサーバーはドメイン名を IP AWS アドレスに変換するのに役立つネットワークサービスを実行するインスタンスです。Lightsail には、 AWS トラフィックをドメインにルーティングするためのネームサーバーオプション (例:`ns-NN.awsdns-NN.com`) がいくつか用意されています。ドメインレジストラを使用してドメインを変更する場合、 AWS これらのネームサーバーの中から選択できます。

詳細については、Wikipedia の「Name server」を参照してください。

サブドメイン

サブドメインは、ドメイン階層内で、上位のドメインに属するドメインのことです (ルートドメインを除く)。たとえば、`blog` は `blog.example.com` サブドメインのサブドメインパートです。

詳細については、Wikipedia の「Subdomain」を参照してください。

有効期限 (TTL)

TTL は、ローカル解決ネームサーバーでの DNS レコードの有効期限を表します。たとえば、期限が短いほど変更が有効になるまでに待機する時間が短くなります。TTL は Lightsail DNS ゾーンでは設定できません。代わりに、すべての Lightsail DNS レコードは 60 秒の TTL にデフォルト設定されています。

詳細については、Wikipedia の「Time to live」を参照してください。

ワイルドカード DNS レコード

ワイルドカード DNS レコードは、存在しないドメイン名に対するリクエストのマッチングを行います。ワイルドカード DNS レコードを指定するには、ドメイン名の左端にアスタリスク記号 (*) を使用します (例: `*.example.com` または `*example.com`)。

> **Note**
>
> Lightsail DNS ゾーンは、ネームサーバー (NS) レコードで定義されたネームサーバードメイン (*awsdns.com) のワイルドカードレコードをサポートします。

Lightsail DNS ゾーンでサポートされている DNS レコードタイプ

アドレス (A) レコード

A レコードは、ドメイン (example.com など) やサブドメイン (blog.example.com など) をウェブサーバーの IP アドレスにマッピングします。

たとえば、Lightsail DNS ゾーンでは、example.com (ドメインの頂点) のウェブトラフィックをインスタンスに送りたいとします。A レコードを作成し、@ シンボルを [サブドメイン] のテキストボックスに入力し、 ウェブサーバーの IP アドレスを [Resolves to address] (解決するアドレス) テキストボックスに入力します。

A レコードの詳細については、Wikipedia の「DNS レコードタイプの一覧」を参照してください。

AAAA レコード

AAAA レコードは、ドメイン (example.com など) やサブドメイン (blog.example.com など) をウェブサーバーの IPv6 アドレスにマッピングします。

たとえば、 Lightsail DNS ゾーンで、IPv6 プロトコルで example.com (ドメインの apex) のウェブトラフィックをインスタンスにダイレクトするとします。AAAA レコードを作成し、@ シンボルを [サブドメイン] のテキストボックスに入力し、 ウェブサーバーの IP アドレスを [Resolves to address] (解決するアドレス) テキストボックスに入力します。

AAAA レコードの詳細については、Wikipedia の「IPv6 のドメインネームシステム」を参照してください。

> **Note**
>
> Lightsail は静的 IPv6 アドレスをサポートしていません。Lightsail リソースを削除して新しいリソースを作成した場合、または同じリソースで IPv6 を無効にしてから再度有効にする場合は、リソースの最新の IPv6 アドレスを反映するように AAAA レコードを更新する必要がある場合があります。

正規名 (CNAME) レコード

CNAME レコードは、エイリアスまたはサブドメイン (`blog.example.com` など) を別のドメインまたはサブドメインにマッピングします。

たとえば、Lightsail DNS ゾーンで、のウェブトラフィックをに転送したいとします。`www.example.com` `example.com`この場合、`www` のエイリアス CNAME レコードを作成し、「解決先」アドレスとして `example.com` を使用します。

詳細については、Wikipedia の「CNAME Record」を参照してください。

メールエクスチェンジャ (MX) レコード

MX レコードは、サブドメイン (`mail.example.com` など) を E メールサーバーアドレスにマッピングします。複数のサーバーを定義する場合は、優先度の値を設定します。

たとえば、Lightsail DNS ゾーンで `10 inbound-smtp.us-west-2.amazonaws.com` Amazon WorkMail サーバー宛のメールをダイレクトしたいとします。`mail.example.com`この場合に作成する MX レコードでは、サブドメインとして `example.com`、優先度として `10`、「解決先」アドレスとして `inbound-smtp.us-west-2.amazonaws.com` を設定します。

詳細については、Wikipedia の「MX レコード」を参照してください。

ネームサーバー (NS) レコード

NS レコードは、サブドメイン (`test.example.com` など) をネームサーバー (`ns-NN.awsdns-NN.com` など) に委任します。

詳細については、Wikipedia の「Name server」を参照してください。

サービスロケーター (SRV) レコード

SRV レコードは、サブドメイン (`service.example.com` など) をサービスアドレスにマッピングします。優先度、重み、およびポート番号の値を設定します。通常 SRV レコードに関連付けられるサービスとして、テレフォニーやインスタントメッセージングなどがあります。

たとえば、Lightsail DNS ゾーンでは、へのトラフィックをに転送したいとします。`service.example.com` `1 10 5269 xmpp-server.example.com`この場合に作成する SRV レコードでは、優先度として `1`、重みとして `10`、ポート番号として `5269`、「マッピング先」アドレスとして `xmpp-server.example.com` を設定します。

詳細については、Wikipedia の「SRV Record」を参照してください。

テキスト (TXT) レコード

TXT レコードは、サブドメインをプレーンテキストにマッピングします。TXT レコードを作成し、サービスプロバイダーに対してドメインの所有権を確認します。

たとえば、Lightsail DNS ゾーンで は、23223a30-7f1d-4sx7-84fb-31bdes7csdbb_amazonchime.example.comホスト名がクエリされたときに応答したいとします。この場合に作成する TXT レコードでは、サブドメイン値として _amazonchime、「応答先」値として 23223a30-7f1d-4sx7-84fb-31bdes7csdbb を設定します。

詳細については、Wikipedia の「TXT Record」を参照してください。

トピック

- Lightsail DNS ゾーンを作成してドメインの DNS レコードを管理する
- Lightsail DNS ゾーンを編集または削除する
- Lightsail でウェブサイトにインターネットトラフィックがルーティングされる方法
- Lightsail ドメインをインスタンスにポイントする
- Lightsail ドメインをロードバランサーにポイントする
- 別の DNS サービスを使用するために Lightsail ドメインネームサーバーを更新する
- Amazon Route 53を使用してドメインを Lightsail インスタンスにポイントする

Lightsail DNS ゾーンを作成してドメインの DNS レコードを管理する

などのドメイン名のトラフィックを Amazon Lightsail インスタンスexample.comにルーティングするには、ドメインのドメインネームシステム (DNS) にレコードを追加します。ドメインの DNS レコードは、ドメインを登録したレジストラを使用して管理することも、Lightsail を使用して管理することもできます。

ドメインの DNS レコードの管理を Lightsail に転送することをお勧めします。これにより、ドメインとコンピューティングリソースを 1 か所のLightsail で効率的に管理できます。Lightsail を使用してドメインの DNS レコードを管理するには、Lightsail DNS ゾーンを作成します。最大 6 つの Lightsail DNS ゾーンを作成できます。6 つを超えるドメイン名を管理しているために、6 つ以上の DNS ゾーンが必要な場合は、すべてのドメインでの DNS 管理に、Amazon Route 53 を使用することをお勧めします。Route 53 を使用して、ドメインのトラフィックを Lightsail リソースにルーティングできま

す。Route 53 による DNS の管理の詳細については、「Amazon Route 53 を使用してドメインをインスタンスに参照させる」を参照してください。

このガイドでは、ドメインの Lightsail DNS ゾーンを作成する方法と、ドメインの DNS レコードの管理を Lightsail に転送する方法を示します。ドメインの DNS レコードの管理を Lightsail に移管した後も、ドメインの更新と請求はドメインのレジストラで引き続き管理されます。

> **Important**
>
> ドメインの DNS に対して行った変更は、インターネットの DNS を通じて伝播されるまで数時間かかる場合があります。このため、Lightsail への管理の転送が伝達される間は、ドメインの DNS レコードをドメインの現在の DNS ホスティングプロバイダーで保持する必要があります。これにより、転送の実行中も、ドメインのトラフィックが途切れることなくリソースにルーティングされます。

目次

ステップ 1: 前提条件を満たす

以下の前提条件を満たします (まだ満たしていない場合)。

1. ドメイン名を登録します。次に、ドメインのネームサーバーを編集するための管理者アクセス権があることを確認します。

   登録済みドメイン名が必要な場合は、Lightsail を使用してドメインを登録できます。詳細については、「ドメインの登録」を参照してください。
2. ドメインに必要な DNS レコードタイプが Lightsail DNS ゾーンでサポートされていることを確認します。Lightsail DNS ゾーンは現在、アドレス (A および AAAA)、正規名 (CNAME)、メールエクスチェンジャ (MX)、ネームサーバー (NS)、サービスロケーター (SRV)、テキスト (TXT) レコードタイプをサポートしています。NS レコードには、ワイルドカード DNS レコードエントリを使用できます。

ドメインに必要な DNS レコードタイプが Lightsail DNS ゾーンでサポートされていない場合は、より多くのレコードタイプをサポートしているため、ドメインの DNS ホスティングプロバイダーとして Route 53 を使用できます。詳細については、「Amazon Route 53 デベロッパーガイド」の「サポートされる DNS レコードタイプ」と「Amazon Route 53 を既存ドメインの DNS サービスとして使用する」を参照してください。

3. ドメインをポイントする Lightsail インスタンスを作成します。詳細については、「インスタンスを作成する」を参照してください。
4. 静的 IP を作成し、Lightsail インスタンスにアタッチします。詳細については、「静的 IP を作成してインスタンスにアタッチする」を参照してください。

ステップ 2: Lightsail コンソールで DNS ゾーンを作成する

Lightsail で DNS ゾーンを作成するには、次のステップを実行します。DNS ゾーンを作成するとき、DNS ゾーンが適用されるドメイン名を指定する必要があります。

1. Lightsail コンソールにサインインします。
2. [Domains & DNS] (ドメイン & DNS) タブを開き、次に [Create DNS zone] (DNS ゾーンの作成) を選択します。
3. 以下のオプションのいずれかを選択します。

   - [Amazon Route 53 に登録されているドメインを使用]して、Amazon Route 53 に登録されたドメインを指定します。
   - [Use a domain from another registrar] (別のレジストラのドメインを使用) で、別のレジストラを使用して登録されたドメインを指定します。
4. 登録済みドメイン名 (`example.com` など) を選択または入力します。

   ドメイン名を入力するときに `www` を含める必要はありません。このガイドの「ステップ 3: DNS ゾーンにレコードを追加する」セクションで、アドレス (A) レコードを使用して `www` を追加できます。

   > **Note**
   >
   > Lightsail DNS ゾーンは、バージニア (`us-east-1`) に作成されますAWS リージョン。作成する Lightsail DNS ゾーン () と同じリージョンのリソースに名前を付けると、リソース名の競合エラー (「一部の名前は既に使用されていますexample.com」) が発生します。

このエラーを解決するには、リソースのスナップショットを作成します。スナップショットから新しいリソースを作成して、新しい一意の名前を付けます。次に、Lightsail DNS ゾーンを作成するドメインと同じ名前の元のリソースを削除します。

5. [DNS ゾーンの作成] を選択します。

   ユーザーは DNS ゾーンの [Assignments] (割り当て) ページにリダイレクトされ、ドメインリソースの割り当てを管理することが可能になります。割り当てを使用して、ロードバランサーやインスタンスなどの Lightsail リソースにドメインをポイントします。

ステップ 3: DNS ゾーンにレコードを追加する

ドメインの DNS ゾーンにレコードを追加するには、以下のステップを実行します。DNS レコードは、そのドメインにインターネットトラフィックがルーティングされる方法を指定します。たとえば、ドメインの apex (`example.com`) のトラフィックを 1 つのインスタンスにルーティングし、サブドメイン (`blog.example.com` など) のトラフィックを異なるインスタンスにルーティングできます。

1. DNS ゾーンの割り当てページで、[DNS records] (DNS レコード) タブを選択します。

   DNS ゾーンは Lightsail コンソールのドメインと DNS タブに表示されます。 https://lightsail.aws.amazon.com/

   > **Note**
   >
   > DNS ゾーンの [Assignments] (割り当て) ページでは、ドメインが指す Lightsail リソースを追加、削除、または変更できます。Lightsail インスタンス、ディストリビューション、コンテナサービス、ロードバランサー、静的 IP アドレスなどをドメインに指定できます。[DNS records] (DNS レコード) のページでは、ドメインの DNS レコードを追加、編集、または削除できます。

2. 以下のいずれかのレコードタイプを選択します。

   アドレス (A) レコード

   A レコードは、 などのドメイン`example.com`、または などのサブドメイン`blog.example.com`、 などのウェブサーバーまたはインスタンスの IPv4 アドレスにマッピングします`192.0.2.255`。

1. [Record name] (レコード名) テキストボックスに、レコードのターゲットサブドメインを入力するか、@ 記号を入力してドメインの最上位を定義します。
2. [Resolves to (解決先)] テキストボックスに、レコードのターゲット IP アドレスを入力し、実行中のインスタンスまたは設定済みのロードバランサーを選択します。実行中のインスタンスを選択すると、そのインスタンスのパブリック IP アドレスが自動的に追加されます。
3. AWS リソースエイリアスを選択して、トラフィックを Lightsail にルーティングし、ディストリビューションやコンテナサービスなどのAWSリソースを選択します。DNS ゾーン内のあるレコードから別のレコードにトラフィックをルーティングできます。

> **Note**
>
> Lightsail インスタンスに静的 IP をアタッチし、レコードが解決される値として静的 IP を選択することをお勧めします。詳細については、「静的 IP を作成する」を参照してください。

AAAA レコード

AAAA レコードは、ドメイン (`example.com` など) やサブドメイン (`blog.example.com` など) をウェブサーバーまたはインスタンスの IPv6 アドレス (`2001:0db8:85a3:0000:0000:8a2e:0370:7334` など) にマッピングします。

> **Note**
>
> Lightsail は静的 IPv6 アドレスをサポートしていません。Lightsail リソースを削除して新しいリソースを作成する場合、または同じリソースで IPv6 を無効にして再度有効にする場合は、リソースの最新の IPv6 アドレスを反映するように AAAA レコードを更新する必要がある場合があります。

1. [Record name] (レコード名) テキストボックスに、レコードのターゲットサブドメインを入力するか、@ 記号を入力してドメインの最上位を定義します。
2. [Resolves to (解決先)] テキストボックスに、レコードのターゲット IPv6 アドレスを入力し、実行中のインスタンスまたは設定済みのロードバランサーを選択します。実行中のインスタンスを選択すると、そのインスタンスのパブリック IPv6 アドレスが自動的に追加されます。

3. AWS リソースエイリアスを選択して、トラフィックを Lightsail にルーティングし、ディストリビューションやコンテナサービスなどのAWSリソースを選択します。DNS ゾーン内のあるレコードから別のレコードにトラフィックをルーティングできます。

正規名 (CNAME) レコード

CNAME レコードは、`www.example.com` などのエイリアスまたはサブドメインを `example.com` などの別のドメイン、または `blog.example.com` などの別のサブドメインにマップします。

1. [Record name] (レコード名) テキストボックスに、レコードのサブドメインを入力します。
2. [Route traffic to] (トラフィックのルーティング先) テキストボックスに、レコードのターゲットドメインまたはサブドメインを入力します。

メールエクスチェンジャ (MX) レコード

MX レコードは、サブドメイン (`mail.example.com` など) を E メールサーバーアドレスにマッピングします。複数のサーバーを定義する場合は、優先度の値を設定します。

1. [Record name] (レコード名) テキストボックスに、レコードのサブドメインを入力します。
2. [優先度] テキストボックスに、レコードの優先度を入力します。これは、複数のサーバーにレコードを追加する場合に重要です。
3. [Route traffic to] (トラフィックのルーティング先) テキストボックスに、レコードのターゲットドメインまたはサブドメインを入力します。

サービスロケーター (SRV) レコード

SRV レコードは、サブドメイン (`service.example.com` など) をサービスアドレスにマッピングします。優先度、重み、およびポート番号の値を設定します。通常 SRV レコードに関連付けられるサービスとして、テレフォニーやインスタントメッセージングなどがあります。

1. [Record name] (レコード名) テキストボックスに、レコードのサブドメインを入力します。
2. [優先度] テキストボックスに、レコードの優先度を入力します。
3. [Weight (重み)] テキストボックスに、同じ優先度を持つ SRV レコードの相対的な重みを入力します。
4. [Route traffic to] (トラフィックのルーティング先) テキストボックスに、レコードのターゲットドメインまたはサブドメインを入力します。

5. [Port (ポート)] テキストボックスに、サービスへの接続を確立できるポート番号を入力します。

テキスト (TXT) レコード

TXT レコードは、サブドメインをプレーンテキストにマッピングします。TXT レコードを作成し、サービスプロバイダーに対してドメインの所有権を確認します。

1. [Record name] (レコード名) テキストボックスに、レコードのサブドメインを入力します。
2. [Responds with (応答内容)] テキストボックスに、サブドメインに対してクエリが実行されたときに返すテキストレスポンスを入力します。

> **Note**
>
> 入力テキストは、引用符で囲む必要はありません。

3. レコードの追加が終了したら、保存アイコンを選択して変更を保存します。

レコードが DNS ゾーンに追加されます。ドメインの DNS ゾーンに複数のレコードを追加する場合は、上の手順を繰り返します。

> **Note**
>
> DNS レコードの有効期限 (TTL) を Lightsail DNS ゾーンで設定することはできません。代わりに、すべての Lightsail DNS レコードはデフォルトで 60 秒の TTL になります。詳細については、Wikipedia の「Time to live」を参照してください。

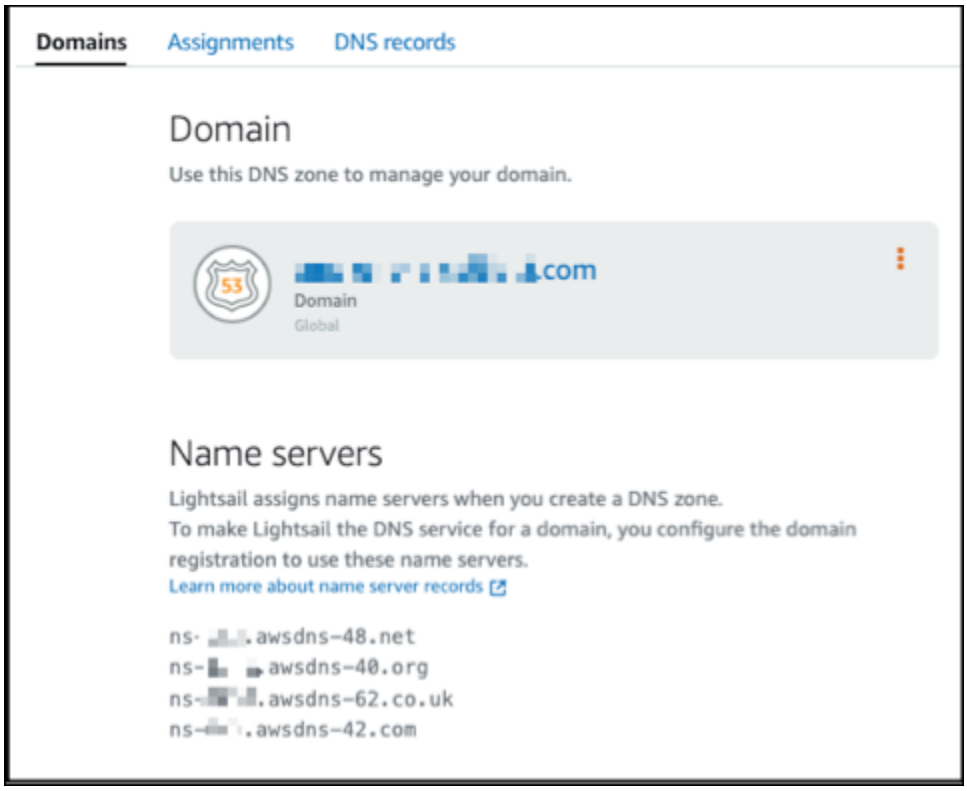
ステップ 4: ドメインの現在の DNS ホスティングプロバイダーでネームサーバーを変更する

ドメインの DNS レコードの管理を Lightsail に転送するには、次のステップを実行します。これを行うには、ドメインの現在の DNS ホスティングプロバイダーのウェブサイトにサインインし、ドメインのネームサーバーを Lightsail ネームサーバーに変更します。

> **Important**
>
> ウェブトラフィックが現在ドメインにルーティングされている場合は、ドメインの現在の DNS ホスティングプロバイダーでネームサーバーを変更する前に、既存の DNS レコード

がすべて Lightsail DNS ゾーンに存在することを確認してください。これにより、Lightsail DNS ゾーンへの転送後、トラフィックは中断することなく継続的に流れます。

1. ドメインの DNS ゾーン管理ページに記載されている Lightsail ネームサーバーを書き留めます。ネームサーバーは Lightsail DNS ゾーンのドメインタブにあります。



2. ドメインの現在の DNS ホスティングプロバイダーのウェブサイトにサインインします。
3. ドメインのネームサーバーを編集できるページを見つけます。

   このページを見つける詳しい方法については、ドメインの現在の DNS ホスティングプロバイダーのドキュメントを参照してください。
4. Lightsail ネームサーバーを入力し、一覧表示されている他のネームサーバーを削除します。
5. 変更を保存します。

   ネームサーバーの変更がインターネットの DNS を通じて伝播されるまで数時間かかる場合があります。完了すると、ドメインのインターネットトラフィックが Lightsail DNS ゾーンを経由してルーティングされ始めます。

次のステップ

- DNS ゾーンを編集または削除する
- ロードバランサーを作成してインスタンスをアタッチする

Lightsail DNS ゾーンを編集または削除する

ドメインの DNS ゾーンの DNS レコードを追加、編集、または削除できます。ドメインの DNS レコードの管理を別の DNS ホスティングプロバイダーに引き渡したり、ドメインを登録したレジストラに戻したりする場合は、ドメインの DNS ゾーンを Amazon Lightsail で削除することもできます。

> **Note**
>
> Lightsail コンソールで DNS エディタを使用してレコードを編集するには、事前にドメインの DNS レコードの管理を Lightsail に引き渡す必要があります。詳細については、「DNS ゾーンを作成してドメインの DNS レコードを管理する」を参照してください。

DNS レコードを編集する

ドメインの DNS ゾーンの DNS レコードは、Lightsail コンソールを使用していつでも編集できます。

DNS ゾーンを編集するには

1. Lightsail コンソールにサインインします。
2. [Domains & DNS] (ドメインと DNS) タブを選択し、編集する DNS ゾーンの名前を選択します。
3. DNS ゾーンの [DNS records] (DNS レコード) ページで、次のいずれかのオプションを選択します。
   - 新しいレコードを追加するには、[レコードの追加] を選択します。
   - 既存のレコードを編集するには、編集するレコードの横にある編集アイコンを選択します。
   - 既存のレコードを削除するには、削除するレコードの横にある削除アイコンを選択します。
4. 終了したら、保存アイコンを選択して変更を保存します。

> **Note**
>
> DNS レコードの変更がインターネットの DNS を通じて伝播されるまで数時間かかる場合があります。

DNS ゾーンの削除

Lightsail でドメインの DNS ゾーンを削除できます。

> **Important**
>
> 引き続きドメインを通じてトラフィックをルーティングする場合は、Lightsail でドメインの DNS ゾーンを削除する前に、別の DNS ホスティングプロバイダーを準備します。そうしないと、Lightsail の DNS ゾーンを削除したときに、ウェブサイトへのすべてのトラフィックが停止します。

DNS ゾーンを削除するには

1. Lightsail コンソールのホームページで、[Domains & DNS] (ドメインと DNS) タブを選択します。
2. 削除する DNS ゾーンの名前を選択します。
3. 縦三点リーダーメニュー (⋮) を選択します。次に、[Delete] (削除) オプションを選択します。
4. [Delete DNS zone] (DNS ゾーンを削除) を選択して削除を確定します。

   DNS ゾーンが Lightsail から削除されます。

Lightsail でウェブサイトにインターネットトラフィックがルーティングされる方法

スマートフォン、ラップトップ、ウェブサイトサーバーなど、インターネット上のすべてのコンピュータは、一意の文字列を使用して相互に通信します。これらの文字列は、IP アドレスと呼ばれ、次のいずれかの形式になります。

- インターネットプロトコルバージョン 4 (IPv4) 形式 (192.0.2.44 など)

- インターネットプロトコルバージョン 6 (IPv6) 形式 (2001:DB8::/32 など)

ブラウザを開いてウェブサイトにアクセスするときは、このような長い文字列を覚えて入力する必要はありません。代わりに、example.com のようなドメイン名を入力しても、正しい場所にアクセスできます。そのためには、ドメインネームシステム (DNS) を使用します。DNS は、登録されたドメイン名を IP アドレスにマッピングするディレクトリとして機能します。

目次

ドメインのインターネットトラフィックをルーティングするように Lightsail を設定する方法の概要

ここでは、Lightsail を使用して、ウェブサイトやウェブアプリケーションにインターネットトラフィックをルーティングするようにドメインを登録および設定する方法の概要を示します。

1. ドメイン名を登録する。概要については、「ドメイン登録」を参照してください。
2. ドメイン名を登録すると、Lightsail によってドメインと同じ名前の DNS ゾーンが自動的に作成されます。
3. Lightsail コンソールでは、インスタンスやロードバランサーなどの Lightsail リソースにドメインを簡単に割り当てることができます。DNS ゾーンに DNS レコードを作成して、リソースにトラフィックをルーティングすることもできます。各レコードには、ドメインのトラフィックをどのようにルーティングするかについて、以下のような情報が含まれます。

   名前

   レコードの名前は、ドメイン名 (example.com) またはサブドメイン名 (www.example.com、retail.example.com) に対応します。DNS ゾーン内の各レコードの名前は、DNS ゾーンの名前で終わる必要があります。例えば、DNS ゾーンの名前が example.com の場合、すべてのレコード名は example.com で終わる必要があります。

   タイプ

レコードタイプは、通常、トラフィックをルーティングする先のリソースのタイプによって決まります。例えば、トラフィックを E メールサーバーにルーティングするには、[Type] (タイプ) で [MX] を指定します。ドメイン名のトラフィックを Lightsail インスタンスにルーティングするには、ドメイン名がインスタンスの静的 IPv4 アドレスを指す A レコードを追加するか、インスタンスの IPv6 アドレスを指す AAAA レコードを追加します。

4. [Target] (ターゲット)

   ターゲットとは、トラフィックのルーティング先です。トラフィックを Lightsail インスタンス、Lightsail コンテナサービス、その他の Lightsail リソースにルーティングするエイリアスレコードを作成できます。詳細については、「DNS」を参照してください。

ドメインにインターネットトラフィックがルーティングされる方法

インスタンス、ロードバランサー、ディストリビューション、コンテナサービスなどのリソースにインターネットトラフィックをルーティングするように Lightsail を設定した後に、www.example.com のコンテンツへのリクエストがあると、何が起こるかを次に示します。

1. ユーザーがウェブブラウザを開き、アドレスバーに「www.example.com」と入力して、Enter キーを押したとします。
2. www.example.com のリクエストは DNS リゾルバーにルーティングされます。DNS リゾルバーは通常、ユーザーのインターネットサービスプロバイダー (ISP) によって管理されています。ISP には、ケーブルインターネットプロバイダー、DSL ブロードバンドプロバイダー、企業ネットワークなどがあります。
3. ISP の DNS リゾルバーは、www.example.com のリクエストを DNS ルートネームサーバーに転送します。
4. DNS リゾルバーは www.example.com のリクエストが再びあると、今度は .com ドメインのいずれかの TLD ネームサーバーに転送します。.com ドメインのネームサーバーは、example.com ドメインに関連付けられている 4 つのネームサーバーの名前でリクエストに応答します。

   DNS リソルバーは、4 つのネームサーバーをキャッシュ (保存) します。次回に誰かが example.com を参照すると、既に example.com のネームサーバーがあるため、ステップ 3 およびステップ 4 はスキップされます。通常、ネームサーバーは 2 日間キャッシュされます。
5. DNS リゾルバーは、ネームサーバーを選択し、www.example.com のリクエストをそのネームサーバーに転送します。

6. ネームサーバーは、example.com DNS ゾーンで www.example.com レコードを検索し、関連付けられた値 (ウェブサーバーの IP アドレス 192.0.2.44 など) を取得します。次に、ネームサーバーは IP アドレスを DNS リゾルバーに返します。
7. DNS リゾルバーには最終的に、ユーザーが必要とする IP アドレスがあります。リゾルバーは、その値をウェブブラウザに返します。
8. ウェブブラウザは、DNS リゾルバーから取得した IP アドレスに www.example.com のリクエストを送信します。これは、Lightsail インスタンスで実行されているウェブサーバー、ウェブサイトエンドポイントとして設定されているコンテナサービスなど、コンテンツが置かれている場所です。
9. 192.0.2.44 にあるウェブサーバーなどのリソースは、www.example.com のウェブページをウェブブラウザに返し、ウェブブラウザはそのページを表示します。

次のステップ

- DNS
- ドメインをインスタンスにポイントする
- ドメインをロードバランサーにポイントする
- ドメインをディストリビューションにポイントする

Lightsail ドメインをインスタンスにポイントする

Amazon Lightsail の DNS ゾーンを使用して、example.com などの登録済みドメイン名を、Lightsail インスタンス (別名、仮想プライベートサーバー (VPS)) 上で実行されているウェブサイトに向けてポイントすることができます。Lightsail アカウントには最大 6 つの DNS ゾーンを作成できます。DNS レコードタイプは、全種類サポートされているわけではありません。Lightsail DNS ゾーンの詳細については、「DNS」を参照してください。

6 つ以上の DNS ゾーンを作成する場合や、Lightsail でサポートされていない DNS レコードタイプを使用する場合は、Amazon Route 53 の ホストゾーンの使用をお勧めします。Route 53 を使用すると、DNS で、最大 500 個のドメインに対応させることができます。また、より多様な DNS レコードタイプをサポートするようになります。詳細については、Amazon Route 53 デベロッパーガイドの「ホストゾーンの使用」を参照してください。

このガイドでは、Lightsail で管理されているドメインの DNS レコードを編集して、Lightsail インスタンスに向けてポイントさせる方法を説明します。DNS の変更がインターネットの DNS を通じて伝播されるまで、最大 48 時間待機します。

前提条件

以下の前提条件を満たします (まだ満たしていない場合)。

- Lightsail を使用してドメイン名を登録する 詳細については、「新しいドメインを登録する」を参照してください。
- すでにドメインを登録しているけれども、そのレコードの管理に Lightsail を使用していない場合は、ドメインの DNS レコードの管理を Lightsail に転送する必要があります。詳細については、「DNS ゾーンを作成してドメインの DNS レコードを管理する」を参照してください。
- Lightsail にアタッチされているデフォルトの動的なパブリック IP アドレスは、インスタンスを停止して再起動するたびに変わります。パブリック IP アドレスが変わらないようにするには、静的 IP アドレスを作成して、それをインスタンスにアタッチします。このガイドでは、ドメイン内で静的 IP アドレスを解決する DNS ゾーンに DNS レコードを作成します。これにより、インスタンスの停止および再開のたびに、ドメインの DNS レコードを更新する必要がなくなります。詳細については、「静的 IP を作成してインスタンスにアタッチする」を参照してください。

  オプション – Lightsail インスタンスでは IPv6 を有効のままにしておくことができます。この IPv6 アドレスは、インスタンスを停止および再開しても保持されます。詳細については、「IPv6 の有効化と無効化」を参照してください。

Lightsail インスタンスにドメインを割り当てる

Lightsail 内のインスタンスにドメインを割り当てるには、以下のいずれかの方法を使用します。

- インスタンスドメインタブ
- 静的 IP ドメインタブ
- DNS ゾーン割り当てタブ

インスタンスドメインタブ

Lightsail コンソールにあるインスタンスの [Domains] (ドメイン) タブで、以下の手順を実行して、ドメインを Lightsail インスタンスに割り当てます。

インスタンスの [Domains] (ドメイン) タブを使用してドメインを割り当てるには

1. Lightsail コンソールにサインインします。
2. ドメインの割り当て先となるインスタンス名を選択します。
3. [Domains] (ドメイン) タブで [Assign domain] (ドメインの割り当て) を選択します。

4. Lightsail インスタンスに割り当てるドメインを選択します。
5. ルーティング情報が正しいことを確認し、[Assign] (割り当て) を選択します。

オプション

インスタンスへのドメイン割り当てを編集また削除するには、ドメイン名の横にある編集アイコン、または、ごみ箱アイコンを選択します。

静的 IP ドメインタブ

Lightsail コンソールにある静的 IP の [Domains] (ドメイン) タブで、以下の手順を実行して、ドメインを Lightsail インスタンスに割り当てます。

静的 IP の [Domains] (ドメイン) タブを使用してドメインを割り当てるには

1. Lightsail コンソールにサインインします。
2. [ネットワーキング] タブを選択します。
3. ドメインの割り当て先となる静的 IP を選択します。
4. [Domains] (ドメイン) タブで [Assign domain] (ドメインの割り当て) を選択します。
5. 静的 IP に割り当てるドメインを選択します。
6. ルーティング情報が正しいことを確認し、[Assign] (割り当て) を選択します。

オプション

静的 IP アドレスへのドメイン割り当てを編集または削除するには、ドメイン名の横にある編集アイコン、または、ごみ箱アイコンを選択します。

DNS ゾーン割り当てタブ

DNS ゾーンの [Assignments] (割り当て) タブで、以下の手順を実行して、Lightsail インスタンスにドメインを割り当てます。

[Assignments] (割り当て) タブを使用してドメインを割り当てるには

1. Lightsail コンソールにサインインします。
2. [ドメインと DNS] タブを選択します。
3. 対象のドメイン名に対して使用する DNS ゾーンを選択します。
4. [Assignments] (割り当て) タブで [Add assignment] (割り当てを追加) を選択します。

5. Lightsail インスタンスに割り当てるドメイン名を選択します。静的 IP がインスタンスにまだアタッチされていない場合は、アタッチするよう求められます。
6. ルーティング情報が正しいことを確認し、[Assign] (割り当て) を選択します。

オプション

このリソースでのドメイン割り当てを編集または削除するには、ドメイン名の横にある編集アイコン、または、ごみ箱アイコンを選択します。

Lightsail ドメインをロードバランサーにポイントする

トラフィックを暗号化 (HTTPS) するドメインを自分が管理していることを確認したら、アドレス (A) レコードを、ドメインを Lightsail ロードバランサーにポイントするドメインの DNS ホスティングプロバイダーに追加する必要があります。このガイドでは、Lightsail DNSゾーン、および Amazon Route 53 ホストゾーンに A レコードを追加する方法について説明します。

DNS ゾーン - アサインメントページを使用して A レコードを追加する

1. Lightsail のホームページで [Domains & DNS] (ドメインと DNS) を選択します。
2. 管理する DNS ゾーンを選択します。
3. [Assignments] (割り当て) タブを選択します。
4. [Add assignment] (割り当てを追加) を選択します。
5. [Select a domain name] (ドメイン名を選択) フィールドで、ドメイン名を使用するか、ドメインのサブドメインを使用するかを選択します。
6. [Select a resource] (リソースの選択) ドロップダウンで、ドメインを割り当てるロードバランサーを選択します。
7. [Assign (割り当てる)] を選択します。

変更がインターネットの DNS を通じて伝播されるまで待ちます。これには数分から数時間かかる場合があります。

DNS ゾーン - DNS レコードページを使用して A レコードを追加する

1. Lightsail のホームページで [Domains & DNS] (ドメインと DNS) を選択します。
2. 管理する DNS ゾーンを選択します。
3. [DNS records] (DNS レコード) タブを選択します。

4. 現在の DNS ゾーンの状態に応じて、次のいずれかの手順を実行します。

   - A レコードが追加されていない場合、レコードを追加を選択します。
   - A レコードが既に追加されている場合、ページにリストされている既存のレコードの横にある編集アイコンを選択し、ステップ 5 に進みます。

5. レコードタイプのドロップダウンメニューで A レコードを選択します。
6. [Record name] (レコード名) テキストボックスに、次のいずれかのオプションを入力します。

   - @を入力して、ドメインの頂点のトラフィックをロードバランサーにルーティングします。(例：`example.com`)
   - wwwを入力して、www サブドメインのトラフィックをロードバランサーにルーティングします。 (例:`www.example.com`)

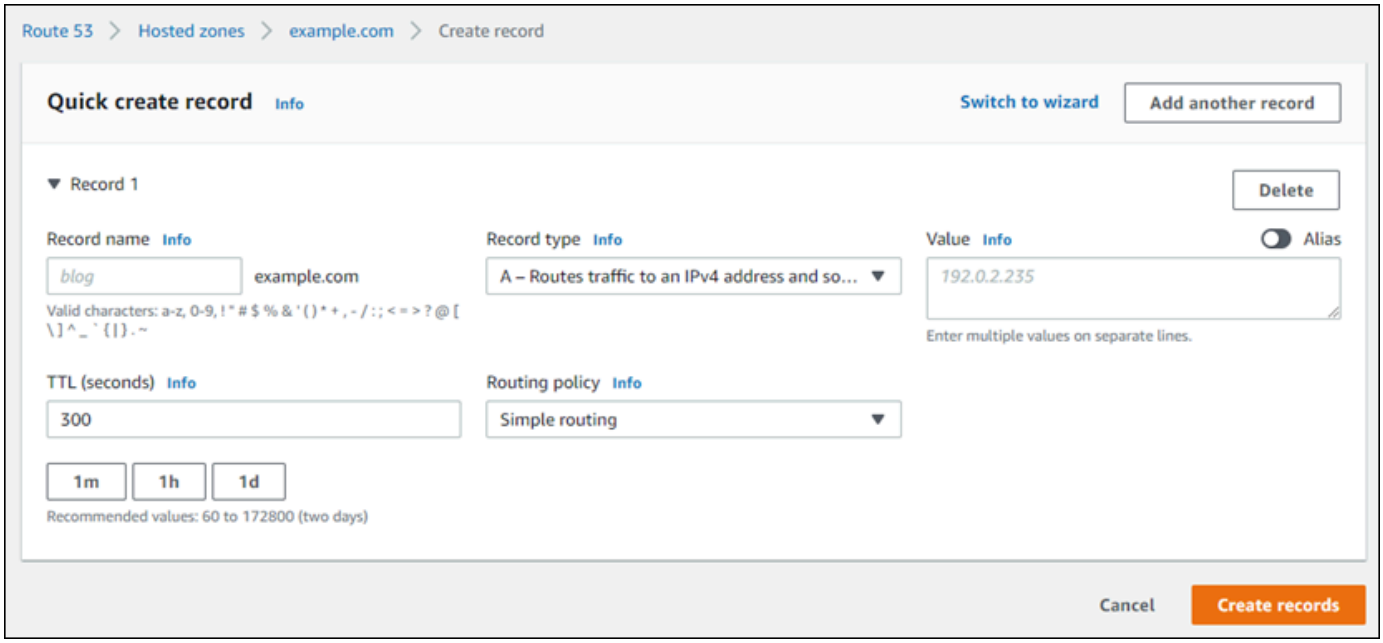
7. 解決テキストボックスで、Lightsailロードバランサーの名前を選択します。
8. 保存アイコンを選択します。

変更がインターネットの DNS を通じて伝播されるまで待ちます。これには数分から数時間かかる場合があります。

Route 53 に A レコードを追加する

1. Route 53 コンソールにサインインします。
2. ナビゲーションペインで [Hosted zones] を選択します。
3. ロードバランサーへのトラフィックのルートに使用するドメイン名のホストゾーンを選択します。
4. [Create record] (レコードを作成) を選択します。

   「レコードのクイック作成」ページが表示されます。

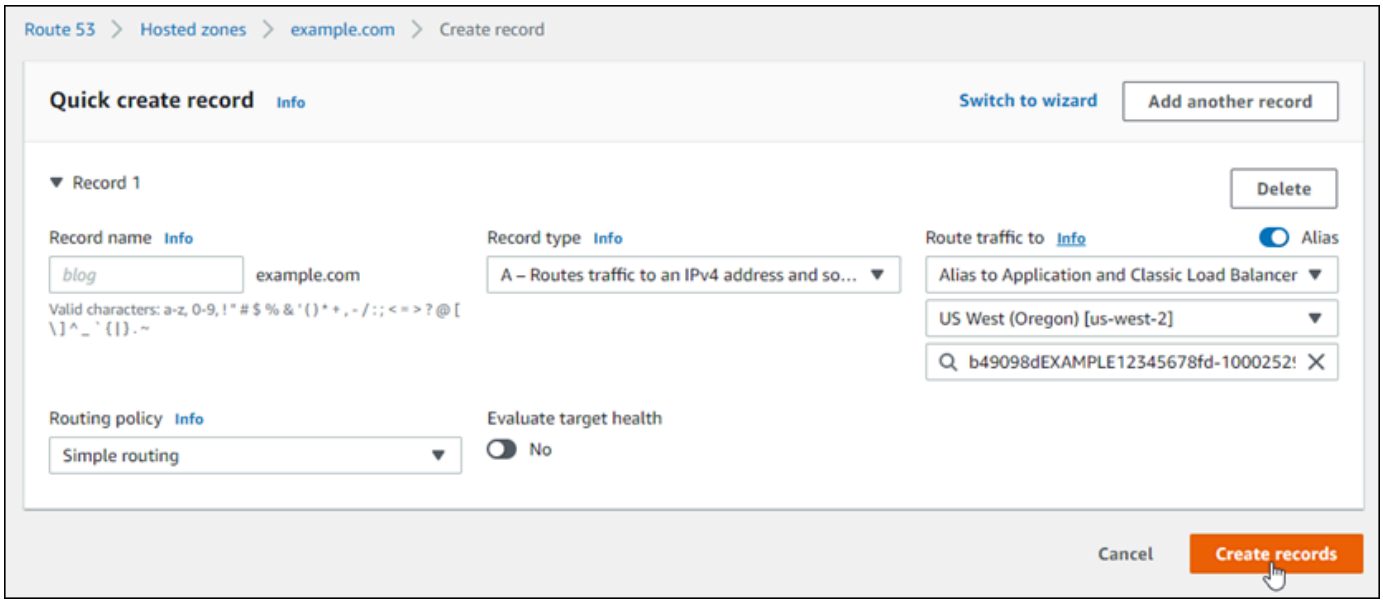


> **Note**
>
> ルーティングポリシーの選択ページが表示されている場合、クイック作成に切り替えるを選択し、次の手順を続行する前に、クイック作成ウィザードに切り替えます。

5. レコード名は、wwwサブドメイン (例:`www.example.com`) を使用する場合wwwを入力するか、ドメインの頂点を使用する場合は空白のままにします。 (例:`example.com`)
6. レコードタイプは、A – IPv4アドレスにトラフィックをルートしていくつかのAWSリソースを選択します。
7. エイリアス切り替えを選択して、エイリアスレコードを有効にします。
8. トラフィックのルーティング先は以下を選択します。
   a. エンドポイントの選択は、アプリケーションにエイリアスおよびClassic Load Balancerを選択します。
   b. リージョンの選択は、Lightsail ロードバランサーを作成したAWSリージョンを選択します。
   c. ロードバランサーの選択は、Lightsailロードバランサーのエンドポイント URL (例:DNS名) を入力、あるいはペーストします。
9. ルーティングポリシーは、シンプルルーティングを選択し、ターゲットの正常性の評価切り替えを無効化します。

Lightsail は既にロードバランサーでヘルスチェックを実行しています。詳細については、ロードバランサーのヘルスチェック を参照してください。

レコードは以下の例のようになります。



10. [レコード作成] を選択してホストゾーンにレコードを追加します。

> **Note**
>
> 変更がインターネットの DNS を通じて伝播されるまで待ちます。これには数分から数時間かかる場合があります。

別の DNS サービスを使用するために Lightsail ドメインネームサーバーを更新する

Lightsail で登録したドメインの DNS レコードは、Amazon Lightsail DNS ゾーンを使用して管理できます。または、必要に応じて、ドメインの DNS レコードの管理を別の DNS ホストプロバイダーに移管することもできます。このガイドでは、Lightsail に登録したドメインの DNS レコードの管理を別の DNS ホストプロバイダーに移管する方法を説明します。

> **Important**
>
> ドメインの DNS に対して行った変更は、インターネットの DNS を通じて伝播されるまで数時間かかる場合があります。このため、管理の移管が完了するまでは、ドメインの DNS レ

コードを現在の DNS ホストプロバイダーで保管しておく必要があります。これにより、転送の実行中も、ドメインのトラフィックが途切れることなくリソースにルーティングされます。

目次

前提条件を満たす

以下の前提条件を満たします (まだ満たしていない場合)。

1. ドメイン名を登録します。Lightsail を使用してドメイン名を登録できます。詳細については、「新しいドメインを登録する」を参照してください。
2. DNS サービスから提供されるプロセスを使用して、ドメインのネームサーバーを取得します。

DNS ゾーンにレコードを追加する

次の手順を実行して、別の DNS ホストプロバイダーのネームサーバーを Lightsail の登録ドメインに追加します。

1. Lightsail コンソールにサインインします。
2. [ドメインと DNS] タブを選択します。
3. 他の DNS サービスを使用するように設定するドメインの名前を選択します。
4. [Edit Name Servers] (ネームサーバーを編集) を選択します。
5. ネームサーバーの名前を、前提条件を完了したときに DNS サービスから取得したネームサーバーに変更します。
6. [保存] を選択します。

Amazon Route 53を使用してドメインを Lightsail インスタンスにポイントする

Amazon Lightsail の DNS ゾーンを使用すると、`example.com` などの登録されたドメイン名を Lightsail インスタンスで実行されているウェブサイトに簡単にポイントできます。Lightsail DNS

ゾーンは最大 6 つまで作成できます。また、 DNS レコードタイプが全種類サポートされているわけではありません。Lightsail DNS ゾーンの詳細については、「DNS」を参照してください。

Lightsail DNS ゾーンが過度に制限されている場合、Amazon Route 53 ホストゾーンを使用して、ドメインの DNS レコードを管理することをお勧めします。DNS は、Route 53 を使用して、最大 500 のドメインを管理でき、非常に多様な DNS レコードタイプをサポートします。また、ドメインの DNS レコードを管理するためにすでに Route 53 を使用していて、引き続き使用することを希望する場合があるかも知れません。このガイドでは、Route 53 で管理されているドメインの DNS レコードを編集して、Lightsail インスタンスをポイントする方法を示します。

前提条件

以下の前提条件を満たします (まだ満たしていない場合)。

- Route 53 を使用してドメイン名を登録する。詳細については、「Route 53 ドキュメント」の「新しいドメインの登録」を参照してください。
- すでにドメインを登録しているけれども、そのレコードの管理に Route 53 を使用していない場合は、ドメインの DNS レコードの管理を Route 53 に転送する必要があります。詳細については、「Route 53 ドキュメント」の「Amazon Route 53 を既存ドメインの DNS サービスにする」を参照してください。
- Route 53 にドメインのパブリックホストゾーンを作成します。詳細については、「Route 53 ドキュメント」の「公開ホストゾーンの作成」を参照してください。
- 静的 IP を作成して Lightsail インスタンスにアタッチします。このガイドでは、インスタンスの静的 IP アドレス (パブリック IP アドレス) に解決される、ドメイン の Route 53 ホストゾーンで DNS レコードを作成します。詳細については、「静的 IP を作成してインスタンスにアタッチする」を参照してください。

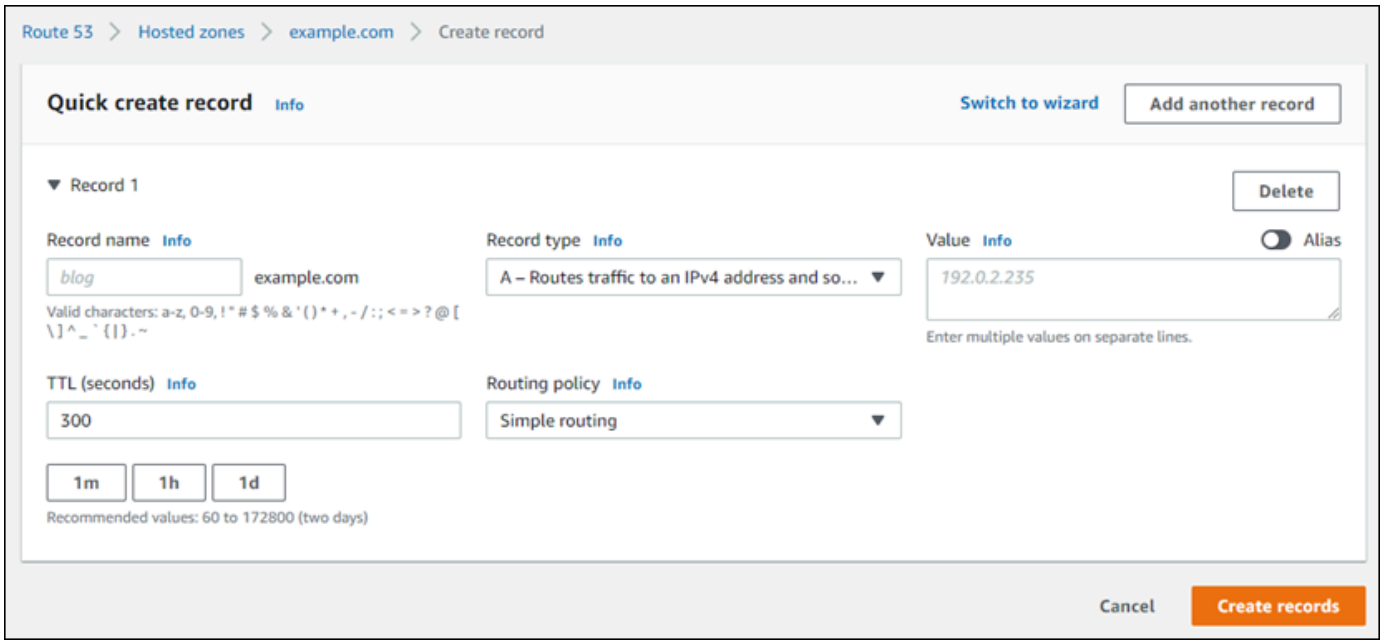
Route 53 を使用してドメインを Lightsail インスタンスにポイントする

Route 53 で 2 つの代表的な DNS レコード、アドレス、および正規名を設定して、ドメインを Lightsail インスタンスにポイントするには、次の手順を実行します。

> **Note**
>
> この手順については Route 53 デベロッパーガイドでも説明しています。詳細については、「Route 53 ドキュメント」の「Amazon Route 53 コンソールを使用したレコードの作成」を参照してください。

1. Route 53 コンソールにサインインします。
2. ナビゲーションペインで [Hosted zones] を選択します。
3. ロードバランサーへのトラフィックのルートに使用するドメイン名のホストゾーンを選択します。
4. [Create record] (レコードを作成) を選択します。

   「レコードのクイック作成」ページが表示されます。



> **Note**
>
> 「ルーティングポリシーの選択」ページが表示された場合は、[クイック作成に切り替える] を選択し、クイック作成ウィザードに切り替えてから、次のステップを続行します。

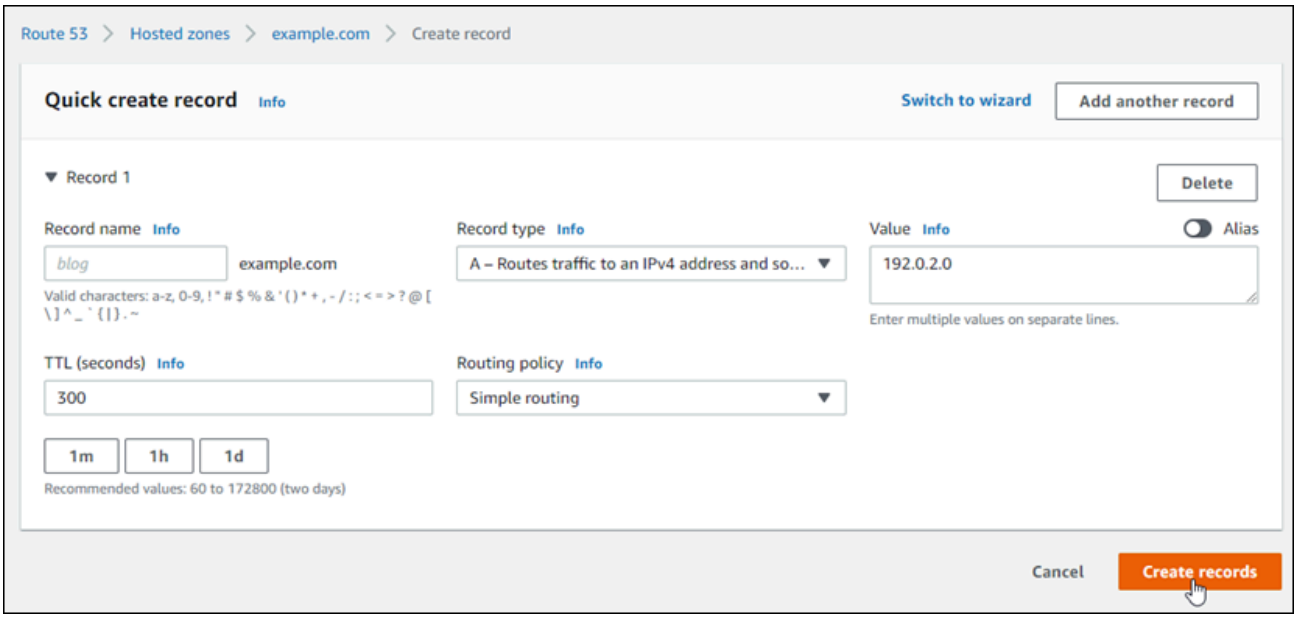
5. [レコードのタイプ] で、以下のいずれかのオプションを選択します。

   A - トラフィックを IPv4 アドレスと一部の AWS リソースにルーティング

   アドレス (A) レコードは、ドメイン (`example.com` など) やサブドメイン (`blog.example.com` など) をウェブサーバーの IP アドレス (`192.0.2.255` など) にマッピングします。

   1. [レコード名] テキストボックスを空のまま、`example.com` などのドメインの頂点が IP アドレスをポイントするようにするか、サブドメインを入力します。

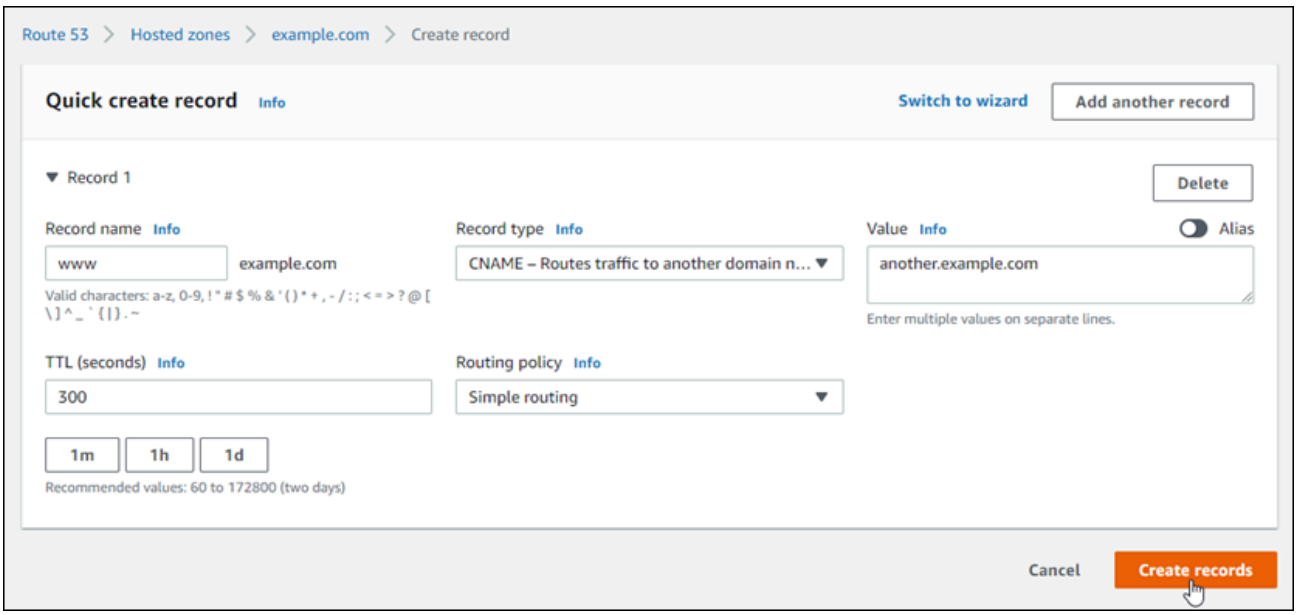
2. [レコードタイプ] ドロップダウンメニューで、[A – トラフィックをIPv4 アドレスと一部の AWS リソース にルーティング] を選択します。
3. Lightsail インスタンスの静的 IP アドレス (パブリック IP アドレス) を [値] テキストボックスに入力します。
4. TTL を 300 に保ち、ルーティングポリシーを [シンプルルーティング] のままにしておきます。



CNAME - トラフィックを別のドメイン名および一部の AWS リソースにルーティング

正規名 (CNAME) レコードは、`www.example.com` などのエイリアスまたはサブドメインを `example.com` などのドメイン、または `www2.example.com` などのサブドメインにマップします。CNAME レコードは、あるドメインを別のドメインにリダイレクトします。

1. [レコード名] テキストボックスにサブドメインを入力します。
2. [レコードタイプ] ドロップダウンメニューで [CNAME - トラフィックを別のドメイン名および一部の AWS リソースにルーティング] を選択します。
3. [Value] (値) テキストボックスにドメイン (例：`example.com`) またはサブドメイン (例：`another.example.com`) を入力します。
4. TTL を 300 に保ち、ルーティングポリシーを [シンプルルーティング] のままにしておきます。



6. [レコード作成] を選択してホストゾーンにレコードを追加します。

> **Note**
>
> 変更がインターネットの DNS を通じて伝播されるまで待ちます。これには数分から数時間かかる場合があります。

Route 53 ホストゾーンで既存のレコードセットを編集するには、編集するレコードを選択し、変更内容を入力して、[保存] を選択します。

Lightsail で新しいドメインを登録する

Amazon Lightsail を使用して新しいドメインを登録できます。可用性が高くスケーラブルな DNS ウェブサービスである Amazon Route 53 を通じて Lightsail ドメインを登録します。使用しているドメインの中に、他のプロバイダーに登録されたものがある場合は、それらのドメインの DNS 管理を Lightsail に移管できます。これらのドメインを Lightsail リソースにポイントさせることもできます。

Lightsail で新しいドメインを登録するには、以下の手順のいずれかを実行します。

- 新しいドメインの登録については、「Lightsail を使用して新しいドメインを登録する」を参照してください。

- 既存のドメインについては、「DNS ゾーンを作成してドメインの DNS レコードを管理する」を参照してください。
- ドメインの別のレジストラへの移管については、「Amazon Route 53 での Lightsail ドメインの管理」を参照してください。

ドメイン登録を開始する前に、以下の考慮事項に留意してください。

ドメイン登録の料金

ドメイン登録にかかるコストについては、「Amazon Route 53 料金表」を参照してください。

ドメインでのサービスクォータ

ユーザーが登録できるドメイン数には上限があります。詳細については、「Amazon Route 53 デベロッパーガイド」の「サービス クォータ」を参照してください。上限の引き上げについては、Route 53 にお問い合わせください。

サポートされるドメイン

Lightsail では、すべての汎用最上位ドメイン (TLD) の登録をサポートしています。サポートされている TLD の一覧については、「Amazon Route 53 デベロッパーガイド」の「Amazon Route 53 に登録できるドメイン」を参照してください。

地理的最上位ドメインを登録するには Route 53 を使用する必要があります。詳細については、「Amazon Route 53 デベロッパーガイド」の「地理的最上位ドメイン」を参照してください。

登録の完了後、ドメイン名を変更することはできません。

誤って正しくないドメイン名を登録した場合でも、そのドメイン名を変更することはできません。代わりに、正しい名前を指定しながら、新たにドメイン名を登録する必要があります。誤って登録したドメイン名について、料金の払い戻しはありません。

DNS ゾーンの料金

ドメインを Lightsail に登録すると、ドメインの DNS サービスが自動的に作成されます。Lightsail DNS ゾーンには料金はかかりません。

Lightsail を使用して新しいドメインを登録する

目次

- 新しいドメインを登録する
- ドメインの連絡先情報を検証する

前提条件を満たす

以下の前提条件を満たします (まだ満たしていない場合)。

1. ドメインの必要な DNS レコードタイプが Lightsail の DNS ゾーンでサポートされていることを確認します。Lightsail DNS ゾーンでは現在、アドレス (A)、正規名 (CNAME)、メールエクスチェンジャ (MX)、ネームサーバー (NS)、サービスロケーター (SRV)、およびテキスト (TXT) のレコードタイプがサポートされています。NS レコードには、ワイルドカード DNS レコードエントリを使用できます。

   ドメインに必要な DNS レコードタイプが Lightsail の DNS ゾーンでサポートされていない場合は、ドメインの DNS ホスティングプロバイダとして、Route 53 を使用するとよいでしょう。Route 53 はより多くのレコードタイプをサポートします。詳細については、「Amazon Route 53 デベロッパーガイド」の「サポートされる DNS レコードタイプ」と「Amazon Route 53 を既存ドメインの DNS サービスとして使用する」を参照してください。

新しいドメインを登録する

新しいドメインを登録するには

1. Lightsail コンソールにサインインします。
2. [ドメインと DNS] タブを選択します。
3. [Register Domain] (ドメインの登録) を選択し、登録するドメインを指定します。
   a. 登録するドメイン名を入力し、[Check availability] (使用可能かチェック) を選択してそのドメイン名が使用できるかどうか確認します。そのドメインが使用可能な場合は、Automatic domain renewal (ドメインの自動更新) をそのまま続けます。
   b. ドメイン名が使用できない場合には、最初の選択肢の代わりとして、あるいはそれに加えて登録できる、他のドメインが一覧表示されます。登録に使用するドメインで、[Select] (選択) を選択します。
4. 有効期限の前に、ドメイン登録の自動更新を行うかどうかを選択します。自分のために登録したドメイン名は、デフォルトで 1 年間その所有権を維持できます。ドメイン名登録の更新を行わないと、有効期限が切れた後、そのドメイン名を他の誰かが登録に使用できるようになります。ド

メイン名を確実に維持するためには、毎年自動更新を行うように設定するか、より長い所有期間を設定します。

5. [Domain contact information] (ドメインの連絡先情報) セクションで、ドメインの登録者、管理者、技術担当者の連絡先情報を入力します。詳細については、「Values that you specify when you register or transfer a domain」(ドメインを登録または移管するときに指定する値) を参照してください。

   以下の考慮事項に注意してください。

   姓と名前

   [First Name] (名) と [Last Name] (姓) には、ご自身の本名と同じ名前を指定することをお勧めします。ドメイン設定の変更に際しては、一部のドメインレジストリで、身分証明書の提供が求められる場合があります。お客様の ID の名前は、ドメイン登録者の連絡先の名前と完全に一致する必要があります。

   他の連絡先

   デフォルトでは、3 種類の連絡先すべてについて同じ情報が使用されます。連絡先として 1 つ以上の異なる情報を入力する場合は、[Same as registrant] (登録者と同じ) チェックボックスをオフにした後に、新しい連絡先情報を入力します。

6. [Privacy Protection] (プライバシー保護) セクションで、WHOIS クエリに対し連絡先情報を隠蔽するかどうかを選択します。

   詳細については、次のトピックを参照してください。

   - プライバシー保護
   - Amazon Route 53 に登録できるドメイン

7. [Register domain] (ドメインの登録) を選択して続行します。[DNS zones] (DNSゾーン) と [Summary] (概要) セクションに、ドメインのDNSゾーン、料金、および更新のスケジュールに関する情報が表示されます。
8. ドメインを登録する前に、「Amazon Route 53 ドメイン名登録規約」に同意する必要があります。

ドメインの連絡先情報を検証する

ドメイン登録の完了後は、登録者の連絡先として有効な E メールアドレスが指定されていることを確認する必要があります。

以下のいずれかの E メールアドレスを使用して、自動的に確認の E メールが送信されます。

noreply@registrar.amazon.com

Amazon Registrar をレジストラとして使用するドメインの場合

noreply@domainnameverification.net

当社のレジストラアソシエイトである Gandi をレジストラとして使用するドメインの場合 自分の TLD のレジストラを特定するには、「Amazon Route 53 デベロッパーガイド」の「Amazon Route 53 に登録できるドメイン」を参照してください。

以下の手順により、ドメイン検証のプロセスを完了します。

ドメインの検証を完了するには

1. 確認 E メールを受け取ったら、E メール内のリンクを選択し、指定した E メールアドレスが有効であることを確認します。E メールがすぐに届かない場合は、迷惑メールフォルダーを確認します。
2. Lightsail コンソールに戻ります。ステータスが自動的に [Verified] (検証済み) に更新されない場合は、[Refresh status] (ステータスの更新) を選択します。

> **Important**
>
> 連絡先となっている登録者は、E メールの指示に従って、そのメールの受信を確認する必要があります。これを行わない場合、対象のドメインは ICANN の規定に従い停止されます。ドメインが停止されると、インターネットでそのドメインにアクセスできなくなります。

3. ドメイン登録が完了したら、DNS サービスとして Lightsail を使用するか、他の DNS サービスを使用するかを選択します。
   - Lightsail

     ドメインおよびサブドメインへのトラフィックのルーティング方法を Lightsail に指示するためのレコードを、ドメイン登録時に Lightsail が作成した DNS ゾーン内に作成します。

     例えば、誰かがブラウザにドメイン名を入力し、そのクエリが Lightsail に転送された際、Lightsail がそのクエリに対して IP アドレスで応答するのか、ロードバランサーの名前で応答するのかを指定します。詳細については、「DNS ゾーンの編集または削除」を参照してください。
   - 別の DNS サービスの使用

Lightsail 以外の DNS サービスに DNS クエリをルーティングするように新しいドメインを設定します。詳細については、「他の DNS サービスを使用するときにドメインのネームサーバーを更新するには」を参照してください。

Amazon Registrar に登録されているドメインに関する情報を表示する

Amazon Lightsail と Amazon Route 53 を使用して登録した .com、.net、および .org ドメインであり、かつ Amazon Registrar がレジストラであるドメインに関する情報を表示できます。これには、ドメインの初回登録時に提供した情報、ドメイン所有者、技術担当者、管理者の連絡先情報が含まれます。

次の点に注意してください。

プライバシー保護がアクティブになっている場合にドメインの連絡先にメールを送信

ドメインに対してプライバシー保護がアクティブになっている場合、登録者、技術担当者、管理者の連絡先情報は、Amazon Registrar プライバシーサービスの連絡先情報に置き換えられます。例えば、example.com ドメインが Amazon Registrar に登録されており、プライバシー保護がアクティブになっている場合、WHOIS クエリに対するレスポンスでは、[Registrant Email] (登録者 E メール) の値は `owner1234@example.com.whoisprivacyservice.org` のようになります。

プライバシー保護がアクティブになっている場合にドメインの連絡先に問い合わせるには、対応するメールアドレスにメールを送信します。E メールは、該当する連絡先に自動的に転送されます。

不正使用を報告

不適切なコンテンツ、フィッシング、マルウェア、スパムなど、違法行為や「Acceptable Use Policy」(利用規定ポリシー) への違反を報告するには、abuse@amazon.com 宛てに E メールを送信してください。

Amazon Registrar に登録されているドメインに関する情報を表示するには

1. ウェブブラウザで、以下のウェブサイトのいずれかに移動します。両方のウェブサイトには同じ情報が表示されます。ただし、使用されるプロトコル、情報の表示形式は異なります。
   - WHOIS: https://registrar.amazon.com/whois
   - RDAP: https://registrar.amazon.com/rdap

2. 情報を表示するドメインの名前を入力し、[Search (検索)] を選択します。検索したドメインが Amazon Lightsail または Route 53 を使用して登録されていない場合は、そのドメインがレジストラデータベースにないことを示すメッセージが表示されます。

Lightsail のドメイン名をフォーマットする

ユーザーがウェブサイトやアプリケーションにアクセスしやすいように、覚えやすいドメイン名を選択します。ドメイン名 (および DNS ゾーンの名前、レコード名) は、ピリオド (.) で区切られた一連のラベルから構成されます。命名要件は、ドメイン名を登録するのか、DNS ゾーンまたはレコードの名前を指定するのかによって異なります。

ドメイン名は、次のガイドラインに従ってフォーマットします。

目次

ドメイン名登録用のドメイン名をフォーマットする

ドメイン名登録では、ドメイン名は 1～255 文字でなければなりません。ドメイン名に使用できるのは、a - z、A - Z、0～9、ハイフン (-)、ピリオド (.) です。

ドメイン名の先頭または末尾にスペースまたはハイフンを使用することはできません。Lightsail は、有効な汎用最上位ドメイン (TLD) 名をすべてサポートします。詳細については、Amazon Route 53 デベロッパーガイドの「汎用最上位ドメイン」を参照してください。

DNS ゾーンとレコード用のドメイン名をフォーマットする

DNS ゾーンとレコードの場合、ドメイン名は 1～255 文字でなければなりません。ドメイン名に使用できるのは、a - z、A - Z、0～9、ハイフン (-)、ピリオド (.) です。スペースは使用できません。

Lightsail では、大文字 (A - Z) で指定しても、英字は小文字 (a - z) として格納されます。

Lightsail は、汎用 TLD と地理的 TLD の両方の DNS ゾーンをサポートします。地理的 TLD のその他の例については、Amazon Route 53 デベロッパーガイドの「地理的最上位ドメイン」を参照してください。

DNS ゾーンとレコードの名前でのアスタリスク (*) の使用

DNS では、名前の中の位置によっては、アスタリスク (*) 文字はワイルドカード文字と見なされます。ワイルドカード DNS レコードとは、未定義のサブドメインの DNS リクエストに応答するレコードです。Lightsail では、次の条件で、名前にアスタリスク (*) を含む DNS ゾーンとレコードを作成できます。

DNS ゾーン

- アスタリスク (*) をドメイン名の左端のラベルに含めることはできません。例えば、subdomain.*.example.com は使用できません。
- アスタリスク (*) が他の位置に含まれる場合、DNS はこれをワイルドカードとしてではなく、ASCII 42 文字として扱います。ASCII 文字の詳細については、ウィキペディアの「ASCII」を参照してください。

DNS レコード

DNS レコード名の中でアスタリスク (*) をワイルドカードとして使用する際は、次の制約に注意してください。

- ワイルドカードとして、アスタリスクはドメイン名の左端のラベルを置き換えるものである必要があります。例えば、*.example.com、*.acme.example.com などです。prod.*.example.com のようにアスタリスクを他の位置に含めると、DNS はこれをワイルドカードとしてではなく、ASCII 42 文字として扱います。
- アスタリスクで、ラベル全体を置き換える必要があります。例えば、*prod.example.com や prod*.example.com と指定することはできません。
- 特定のドメイン名が優先されます。例えば、*.example.com と acme.example.com のレコードを作成すると、acme.example.com の DNS クエリは、acme.example.com レコードの値で応答します。
- アスタリスクは、アスタリスクが含まれたサブドメインレベル、およびそのサブドメインのすべてのサブドメインの DNS クエリに適用されます。例えば、*.example.com という名前のレコードを作成すると、*.example.com の DNS クエリは次の名前に応答します。

  zenith.example.com

  acme.zenith.example.com

  pinnacle.acme.zenith.example.com (その DNS ゾーンにどのタイプのレコードもない場合)

*.example.com という名前のレコードを作成し、example.com レコードがない場合、Lightsail は example.com の DNS クエリに NXDOMAIN (存在しないドメイン) と応答します。

同じレベルのすべてのサブドメインの DNS クエリに対して、さらにドメイン名の DNS クエリに対しても、同じレスポンスを返すように Lightsail を設定できます。例えば、example.com レコードを使用して、acme.example.com や zenith.example.com などの DNS クエリに応答するように Lightsail を設定できます。サブドメインのトラフィックを example.com の最上位ドメインにルーティングするには、次の手順を実行します。

1. ドメインのレコードを作成します (example.com など)。
2. サブドメインのエイリアスレコードを作成します (*.example.com など)。前のステップで作成したレコードの名前を、エイリアスレコードのターゲットとして指定します。

次のステップ

詳細については、次のトピックを参照してください。

- ドメインの DNS レコードを管理する DNS ゾーンを作成する
- DNS

Amazon Route 53 で Lightsail ドメインを管理する

Amazon Lightsail では、可用性が高くスケーラブルな DNS ウェブサービスである Amazon Route 53 を通じてドメインを登録します。Lightsail を使用してドメインを登録することで、Lightsail と Route 53 の両方でドメインの管理が行えます。

ドメインの登録や、ドメインのトラフィックを Lightsail リソースにルーティングするなどのタスクは、Lightsail コンソールで実行します。詳細については、「Amazon Lightsail でのドメイン登録」を参照してください。

ドメインの移管や登録の削除などの高度なタスクは、Amazon Route 53 コンソールで実行する必要があります。

このガイドでは、Route 53 コンソールを使用して完了できる高度な管理タスクの一部について説明します。Route 53 の概要については、「Amazon Route 53 デベロッパーガイド」の「Amazon Route 53 とは」を参照してください。

目次

- ドメイン登録のステータスを表示する
- 別の登録への許可のない移管を防ぐためにドメインをロックする
- 失効した、または削除されたドメインを復元する
- ドメインを移管する
- ドメイン名の登録を削除する

ドメイン登録のステータスを表示する

ドメイン名には、拡張プロビジョニングプロトコル (EPP) ステータスコードとも呼ばれるステータスがあります。EPP ステータスコードは、ICANN (ドメイン名に関する中心的なデータベースを管理する組織) により開発されました。EPP ステータスコードは、各種オペレーションに関するステータスを表します。これらのステータスの例としては、ドメイン名の登録、ドメイン名の登録の更新などに関するものが挙げられます。すべてのレジストラは、この同じステータスコードを使用します。ドメインのステータスコードを確認するには、「Amazon Route 53 デベロッパーガイド」の「ドメイン登録のステータスの表示」を参照してください。

別の登録への許可のない移管を防ぐためにドメインをロックする

すべての汎用最上位ドメイン (TLD) のドメインレジストリでは、許可なく他者がドメインを別のレジストラに移管することを防止するために、ユーザーが自分のドメインをロックする手段を提供しています。詳細については、「Amazon Route 53 デベロッパーガイド」の「別の登録への許可のない移管を防ぐためにドメインをロックする」を参照してください。

失効した、または削除されたドメインを復元する

後期更新期間が終了する前にドメインを更新しないか、ドメインを誤って削除した場合、最上位ドメイン (TLD) のいくつかのレジストリにより、他のユーザーが登録できるようになる前に、ドメインを復元することができます。ドメイン登録の復元を試すには、以下からリンク先にある手順を使用してください。詳細については、「Amazon Route 53 デベロッパーガイド」の「失効した、または削除されたドメインの復元」を参照してください。

ドメイン登録を移管する

別の登録から Amazon Route 53 に、AWS アカウントから別のアカウントに、または Route 53 から別の登録に、ドメインの登録を移管できます。詳細については、「Amazon Route 53 デベロッパーガイド」の「ドメインを移管する」を参照してください。

ドメイン名の登録を削除する

最上位ドメイン (TLD) では、必要がなくなった登録を削除できます。レジストリで登録を削除できる場合、このトピックの手順を実行します。詳細については、Amazon Route 53 デベロッパーガイドの「ドメイン名の登録を削除する」を参照してください。

Lightsail でドメインを登録または移管するときのドメイン情報を提供する

Amazon Lightsail を使用してドメインを登録する場合、登録期間 (契約期間) やドメインの連絡先情報などのドメイン情報を提供します。また、ドメインの自動更新とプライバシー保護も設定します。

Lightsail に現在登録されているドメインの情報を変更することもできます。次の点に注意してください。

- ドメインの連絡先情報を変更した場合は、登録者の連絡先に変更について通知メールが送信されます。この E メールの送信元は noreply@amazon.com です。ほとんどの変更について、登録者は応答する必要はありません。
- 連絡先情報の変更が所有権の変更も含む場合は、登録者の連絡先に追加のメールが送信されます。ドメイン名の中央データベースを管理する組織である ICANN の規則では、メールを受け取ったことについて登録者の連絡先による確認が必要です。詳細については、このセクションの後半にある姓名および組織の項目を参照してください。

既存のドメインの連絡先情報の変更については、「ドメインの連絡先情報を更新する」を参照してください。

お客様が提供するドメイン情報

- 用語
- ドメインの自動更新
- 登録者、管理者、および技術担当者の連絡先
- 登録者と同じ
- 連絡先のタイプ
- 姓名
- 組織
- Email(メール)

- 電話
- 住所 1
- 住所 2
- 国
- 状態
- 市
- 郵便番号
- プライバシー保護

用語

ドメインの登録期間。通常、期間は 1 年ですが、ドメインの登録時に最大 10 年まで延長できます。

ドメインの自動更新

ドメインを Lightsail に登録すると、ドメインが自動的に更新されるようになります。自動更新期間は通常 1 年間です。有効期限切れになる前に Lightsail で自動的にドメイン登録を更新するかどうかを選択します。登録料はお客様の AWS アカウントに課金されます。詳細については、「ドメイン登録の更新」を参照してください。

> **Important**
>
> ドメイン自動更新を非アクティブにした場合、有効期限が過ぎるとドメイン登録は更新されません。その結果、ドメイン名のコントロールを失う可能性があります。

登録者、管理者、および技術担当者の連絡先

デフォルトでは、3 種類の連絡先すべてについて同じ情報が使用されます。連絡先として異なる情報を入力する場合は、それぞれの連絡先について [Same as registrant] (登録者と同じ) の横にあるボックスのチェックを外します。

登録者と同じ

ドメインの登録者、管理者、技術担当者の連絡先として同じ連絡先情報を使用するかどうかを指定します。

連絡先のタイプ

この連絡先のカテゴリ。次の点に注意してください。

- [Company] (会社) または [Association] (協会) オプションを選択した場合は、組織名を入力する必要があります。
- 一部の最上位ドメイン (TLD) の場合、使用可能なプライバシー保護は、[Contact type] (連絡先のタイプ) で選択した値によって異なります。TLD のプライバシー保護設定については、「Amazon Route 53 に登録できるドメイン」を参照してください
-

姓名

連絡先の姓名。[First name] (名) と [Last name] には、公的な身分証明書の名前を使用することをお勧めします。一部のドメイン設定の変更については、身分証明書の提示が必要です。その場合、身分証明書の名前がドメイン登録者の連絡先の名前と一致する必要があります。

登録者の連絡先メールアドレスを変更した場合、このメールは以前のメールアドレスと新しいメールアドレスの両方に送信されます。

組織

連絡先と関連付けられている組織 (存在する場合)。登録者と管理者の連絡先の場合、これは通常、ドメインを登録する組織です。技術担当者の連絡先の場合、これはドメインを管理する組織のこともあります。

連絡先のタイプが [Person] (個人) 以外のときに、登録者の連絡先の [Organization] (組織) フィールドを変更すると、ドメインの所有者が変更されます。ICANN の規則では、登録者の連絡先にメールを送付して承認を得る必要があります。メールは次のメールアドレスの 1 つから送信されます。

- noreply@registrar.amazon.com - Amazon Registrar によって登録された TLD の場合
- noreply@domainnameverification.net – レジストラアソシエイトである Gandi によって登録された TLD の場合

お客様の TLD のレジストラを特定するには、「Amazon Route 53 に登録できるドメイン」を参照してください。

登録者の連絡先 E メールアドレスを変更した場合、この E メールは以前の E メールアドレスと新しい E メールアドレスの両方に送信されます。

メール

連絡先のメールアドレス。次の点に注意してください。

登録者の連絡先のメールアドレスを変更すると、以前のメールアドレスと新しいメールアドレスの両方に通知メールが送信されます。この E メールの送信元は noreply@amazon.com です。

電話

連絡先の電話番号です。

- 米国またはカナダの電話番号を入力する場合は、「1」の後に市外局番を含む 10 桁の電話番号を入力します。
- その他の場所の電話番号を入力する場合は、国コードの後に残りの電話番号を入力します。電話の国コードの一覧については、ウィキペディアの「国際電話番号の一覧」を参照してください。

住所 1

連絡先の住所または私書箱。

住所 2

アパート、スイート、ユニット、ビル、フロア、配達先コードなど、連絡先の追加住所情報。

国

連絡先の国。

状態

連絡先の都道府県。

市町村

連絡先の市町村。

郵便番号

連絡先の郵便番号。

プライバシー保護

WHOIS クエリに対して連絡先情報を隠すかどうかを選択します。ドメインの連絡先情報のプライバシー保護をアクティブにすると、WHOIS (「who is」) クエリは、個人情報の代わりにドメインレジストラの連絡先情報を返します。ドメインレジストラは、ドメイン名登録を管理する会社です。

> **Note**
>
> 管理者、登録者、および技術担当者の連絡先に同じプライバシー設定が適用されます。

ドメインの連絡先情報のプライバシー保護を非アクティブにすると、指定したメールアドレスに送られてくるスパムメールの数が増えます。

だれでもドメインの WHOIS クエリを送信して、そのドメインのすべての連絡先情報を取得することができます。WHOIS コマンドは多くのオペレーティングシステムで利用でき、多くのウェブサイトでウェブアプリケーションとしても利用できます。

> **Important**
>
> ドメイン連絡先情報の正当なユーザーもいますが、最も一般的なユーザーは、迷惑メールや詐欺メールをドメインの連絡先に送りつけるスパム業者です。一般的に、[Contact information] (連絡先情報) については [Privacy protection] (プライバシー保護) を有効のままにしておくことをお勧めします。

プライバシー保護の詳細については、以下のトピックを参照してください。

- ドメインのプライバシー保護を管理する
- Amazon Route 53 に登録できるドメイン

Lightsail でドメイン登録更新を管理する

Amazon Lightsail にドメインを登録すると、ドメインがデフォルトで自動的に更新されるようになります。デフォルトの自動更新期間は 1 年間ですが、一部の最上位ドメイン (TLD) のレジストリの更

新期間はこれより長くなっています。すべての汎用 TLD では、ドメイン登録期間を、通常 1 年単位で最大 10 年まで延長することができます。

> **Note**
>
> AWS アカウント を閉鎖する場合は、必ず自動更新を非アクティブにしてください。そうしないと、アカウントを閉鎖した後でもドメイン登録が更新されます。

目次

自動更新

自動更新がアクティブな場合のタイムラインは次のとおりです。

有効期限の 45 日前

当社から登録者の連絡先に E メールを送信し、自動更新がアクティブになっていることを伝えます。E メールには、自動更新を非アクティブにする方法についての説明も記載されています。登録者の連絡先メールアドレスを最新状態にして、メールが届くようにしておきます。

有効期限の 35 日前または 30 日前

.com.ar、.com.br、.jp ドメインを除くすべてドメインでは、有効期限の 35 日前にドメイン登録の更新が行われます。これにより、ドメイン名の有効期限が切れる前に、更新に関する問題を解決する時間を確保できます。

.com.ar、.com.br、.jp のドメインのレジストリでは、有効期限の 30 日前にならないとドメインの更新ができません。当社のレジストラアソシエイトである Gandi から、有効期限の 30 日前に更新についての E メールが送信されます。自動更新がアクティブな場合、この E メールはドメインを更新したのと同じ日に送信されます。

自動更新が非アクティブの場合、ドメイン名の有効期限が近づいたときのタイムラインは次のとおりです。

有効期限の 45 日前

登録者の連絡先に E メールを送信し、自動更新が現在非アクティブになっていることを伝えます。E メールには、自動更新をアクティブにする方法についての説明も記載されています。登録者の連絡先メールアドレスを最新状態にして、メールが届くようにしておきます。

有効期限切れの 35 日前 または 7 日前

ドメインの自動更新が非アクティブの場合、ドメイン登録の運営組織である ICANN は、レジストラが登録者の連絡先に E メールを送信することを義務付けています。メールは次のメールアドレスの 1 つから送信されます。

noreply@registrar.amazon.com - Amazon Registrar がレジストラになっているドメインの場合

noreply@domainnameverification.net - 当社のレジストラアソシエイトである Gandi がレジストラになっているドメインの場合

有効期限まで 30 日未満の期間に自動更新をアクティブにすると、ドメインは 24 時間以内に更新されます。

更新期間の詳細については、Amazon Route 53 デベロッパーガイドの「Amazon Route 53 に登録できるドメイン」の中の該当する TLD についての「ドメインの更新と復元の期限」セクションを参照してください。

有効期限経過後

ほとんどのドメインは有効期限が切れても短期間保持されるため、有効期限後に失効したドメインを更新できる場合もありますが、ドメインの維持を希望する場合は、自動更新をアクティブにしておくことを強くお勧めします。有効期限後にドメインを更新しようとする場合についての詳細は、Amazon Route 53 デベロッパーガイドの「失効した、または削除されたドメインを復元する」を参照してください。

ドメインの有効期限が切れたが、ドメインで後期更新が許可されている場合は、標準更新価格でドメインを更新できます。ドメインがまだ期限切れ後の更新期間内であるかどうかを確認するには、Amazon Route 53 デベロッパーガイドの「ドメインの登録期間を延長する」に記載されている手順を実行します。ドメインがまだリストされる場合は、後期更新期間内です。

ドメイン登録中のドメイン自動更新の設定

新しいドメイン名を Lightsail に登録すると、ドメインが自動的に更新されるようになります。ドメイン登録手続き中に、自動ドメイン更新を非アクティブにすることもできます。

1. Lightsail コンソールにサインインします。
2. [ドメインと DNS] タブを選択します。
3. [Register domain] (ドメインの登録) ボタンを選択します。
4. Lightsail に登録するドメイン名を指定し、[空き状況の確認] を選択します。
5. ドメイン名が利用可能な場合は、ドメイン登録ページが表示されます。[Automatic domain renewal] (自動ドメイン更新) セクションで、トグルスイッチをオンまたはオフにして、自動ドメイン更新をアクティブまたは非アクティブにします。

登録済みのドメイン自動更新の設定

Lightsail により、有効期限の少し前にドメインの登録を自動更新するかどうかの変更を行う場合、または自動更新の現在の設定を表示する場合には、次の手順を実行します。

1. Lightsail コンソールにサインインします。
2. [ドメインと DNS] タブを選択します。
3. 表示または更新するドメインを選択します。
4. [Contact info] (連絡先情報) タブを選択します
5. 5. [Automatic domain renewal] (ドメインの自動更新) セクションで、トグルスイッチをオンまたはオフにして、ドメインの登録期間中の自動更新をアクティブまたは非アクティブにします。

Lightsail のドメイン連絡先のプライバシー保護を管理する

ドメインを Lightsail に登録すると、ドメインのすべての連絡先について、デフォルトでプライバシー保護がアクティブになります。これによって一般的に、WHOIS ("Who is") クエリから返される連絡先情報の大部分が非表示になり、送られてくるスパムの数が減少します。お客様の連絡先情報は、レジストラの連絡先情報または "REDACTED FOR PRIVACY" という文言に置き換えられます。プライバシー保護の利用には料金はかかりません。

プライバシー保護を非アクティブにすると、誰でもドメインに関する WHOIS クエリを送信でき、ほとんどの最上位ドメイン (TLD) について、ドメインの登録時に指定したすべての連絡先情報を取得できる可能性があります。この情報には、名前、住所、電話番号、E メールアドレスが含まれています。WHOIS コマンドは広く利用可能です。このコマンドは、多くのオペレーティングシステムに含まれ、多くのウェブサイトでウェブアプリケーションとしても利用できます。

Lightsail を使用して登録したドメインのプライバシー保護を管理するには、次の手順を実行します。

目次

前提条件を満たす

ドメインを Lightsail に登録します。詳細については、「新しいドメインを登録する」を参照してください。

ドメインのプライバシー保護を管理する

1. Lightsail コンソールにサインインします。
2. [ドメインと DNS] タブを選択します。
3. プライバシー保護を変更するドメインの名前を選択します。
4. [Contact info] (連絡先情報) を選択します。
5. [Privacy protection] (プライバシー保護) トグルスイッチをオンまたはオフにすることで、連絡先情報のプライバシー保護を管理できます。

Lightsail でドメインの連絡先情報を更新

Amazon Lightsail にドメインを登録するときは、ドメインの連絡先情報を指定します。連絡先情報には、次の 3 つのタイプがあります。

- 登録者: ドメインの所有者
- 管理者: ドメインの管理責任者
- 技術担当者: ドメインに技術的な変更を加える責任者

ドメインの連絡先情報は、ドメインの所有権を確認し、ドメイン名に関連する情報を最新の状態に保つために使用されます。

トピック

ドメインの所有者は誰ですか。

連絡先のタイプが [Person] で、登録者の連絡先の [First Name] または [Last Name] フィールドを変更すると、ドメインの所有者を変更したことになります。

連絡先のタイプが [Person] 以外のときに [Organization] を変更すると、ドメインの所有者が変更されます。

Lightsail に現在登録されているドメインの連絡先情報を変更すると、次のアクションが発生します。

- ドメインの連絡先情報を変更した場合は、登録者の連絡先に変更について通知メールが送信されます。この E メールの送信元は noreply@amazon.com です。ほとんどの変更について、登録者は応答する必要はありません。
- 連絡先情報の変更が所有権の変更も含む場合は、登録者の連絡先に追加のメールが送信されます。ドメイン名の中央データベースを管理する組織である ICANN の規則では、メールを受け取ったことについて登録者の連絡先による確認が必要です。

ドメインの連絡先情報を更新

ドメインの連絡先情報を更新するには、以下の手順を実行します。

1. Lightsail コンソールにサインインします。
2. [ドメインと DNS] タブを選択します。
3. 更新するドメインの名前を選択します。
4. [Contact info] (連絡先情報) タブを選択します。次に、[Edit contact] (連絡先を編集) を選択します。
5. 目的の値を更新します。詳細については、「Amazon Route 53 デベロッパーガイド」の「ドメインを登録または移管するときに指定する値」を参照してください。
6. [Save (保存)] を選択します。

Amazon Lightsail のデータベース

Amazon Lightsail で MySQL または PostgreSQL マネージドデータベースは、いくつかのステップで作成できます。Lightsail は、一般的なメンテナンスとセキュリティタスクを管理することで、データベース管理をより効率的にします。Lightsail コンソールを使用すると、次のことができます。

- データベースをスナップショットにバックアップする。
- スナップショットから新しいより大きなデータベースを作成する。
- ブラウザベースのログやメトリクスを使用して、一般的な問題のトラブルシューティングを行います。
- point-in-time バックアップおよび復元オペレーションを使用してデータを復元します。

Lightsail インスタンスでアプリケーションを構築し、Lightsail マネージドデータベースに接続できます。スタンドアロンデータベースを作成して、貴社の分析ツールやクエリツールを接続することもできます。スタンダートデータベースプランと高可用性データベースプランから選択できます。これらのプランでは、事前設定されたデータベース、SSD ベースのストレージ、およびデータ転送の割り当てが月額の固定料金で提供されます。Lightsail データベースは、 AWS Command Line Interface（AWS CLI）、API、または SDK を使用して管理することもできます。

Lightsail データベースを選択する

Amazon Lightsail は、MySQL および PostgreSQL データベースの最新のメジャーバージョンを提供します。このガイドでは、プロジェクトに適したデータベースの選択に役立つ情報を提供します。

Lightsail は、SQL Server を搭載した Windows Server 2022 インスタンスも提供します。詳細については、「Amazon Lightsail インスタンスイメージの選択」を参照してください。

Lightsail のマネージドデータベースを比較する

MySQL

MySQL 5.7 および 8.0 は Lightsail で使用できます。MySQL は最も広く採用されているオープンソースのリレーショナルデータベースです。多くの一般的なウェブサイト、アプリケーション、および商用製品でプライマリのリレーショナルデータストアとして使用されています。MySQL は、高信頼性の安定した安全な SQL ベースのデータベース管理システムとして、20 年以上にわたってコミュニティからの開発の支援とサポートを受けています。MySQL データベースは、ミッションクリティカルなアプリケーションや動的なウェブサイトなど、さまざまなユースケースに適しています。ま

た、ソフトウェア、ハードウェア、およびアプライアンスの埋め込みデータベースとしても機能します。

> **Important**
>
> 2024 年 6 月 30 日以降、Lightsail は MySQL 5.7 をサポートしなくなり、このブループリントを使用して新しいデータベースを作成できなくなります。データベースインスタンスのメジャーバージョンをアップグレードする方法については、「Lightsail データベースのメジャーバージョンのアップグレード」を参照してください。

詳細については、次の MySQL ドキュメントを参照してください。

- MySQL 5.7 のドキュメント
- MySQL 8.0 のドキュメント

PostgreSQL

PostgreSQL 11、12、13、14、15、および 16 は Lightsail で使用できます。PostgreSQL は、30 年以上もの間開発されてきた強力なオープンソースのオブジェクトリレーショナルデータベースシステムであり、信頼性、機能の堅牢性、およびパフォーマンスで高い評価を得ています。

公式ドキュメント には PostgreSQL のインストール方法と使用方法を説明する豊富な情報があります。PostgreSQL コミュニティ では、テクノロジーに精通し、その仕組みを理解して、そしてキャリアの機会を見つけるために役立つ多くの場所が提供されています。

> **Important**
>
> 2024 年 6 月 30 日以降、Lightsail は PostgreSQL 11 をサポートしなくなり、このブループリントを使用して新しいデータベースを作成できなくなります。データベースインスタンスのメジャーバージョンをアップグレードする方法については、「Lightsail データベースのメジャーバージョンのアップグレード」を参照してください。

詳細については、次の PostgreSQL ドキュメントを参照してください。

- PostgreSQL 11 ドキュメント
- PostgreSQL 12 ドキュメント

- PostgreSQL 13 ドキュメント
- PostgreSQL 14 ドキュメント
- PostgreSQL 15 ドキュメント
- PostgreSQL 16 ドキュメント

データのインポートを最適化する

Lightsail では、複数のデータベースプランを使用できます。各データベースプランには、特定のメモリ、vCPU、ストレージ、データ転送許容量の仕様があります。各データベースプランにはこれらの仕様があるため、新しい Lightsail データベースにインポートするデータ量に適したサイズのデータベースプランを選択することが重要です。サイズの要件に満たないプランを選択すると、データのインポートが遅くなることがあります。以下のガイドラインに従って、データのインポート要件に応じた適切なデータベースプランを選択してください。

- Micro $15 USD/月データベースプラン – データの転送量が 10 GB を超えると、データのインポートが遅くなる場合があります。
- Small $30 USD/月データベースプラン – データの転送量が 20 GB を超えると、データのインポートが遅くなる場合があります。
- Medium $60 USD/月データベースプラン – データの転送量が 85 GB を超えると、データのインポートが遅くなる場合があります。
- Large $115 USD/月データベースプラン – データの転送量が 156 GB を超えると、データのインポートが遅くなる場合があります。

> **Note**
>
> データベースへのデータのインポートの詳細については、「MySQL データベースへのデータのインポート」または「PostgreSQL データベースへのデータへのインポート」を参照してください。

Lightsail の高可用性データベース

Lightsail の高可用性マネージドデータベースは、フェイルオーバー サポートを提供するために、1 つのアベイラビリティーゾーンでプライマリデータベースを保持し、別のアベイラビリティーゾーンでスタンバイ用のセカンダリデータベースを保持します。高可用性データベースは、使用負荷の高

い、データの冗長性を必要とする本番稼働用のワークロードにお勧めします。開発およびテストの目的には、高可用性ではないスタンダードデータベースを使用できます。

高可用性データベースを作成するには、マネージドデータベースの作成時に Lightsail の高可用性データベースプランのいずれかを選択します。詳細については、「データベースを作成する」を参照してください。また、スタンダードデータベースを高可用性データベースに変更することもできます。スタンダードデータベースのスナップショットを作成し、そのスナップショットから新しいデータベースを作成して、高可用性プランを選択します。詳細については、「スナップショットからデータベースを作成する」を参照してください。

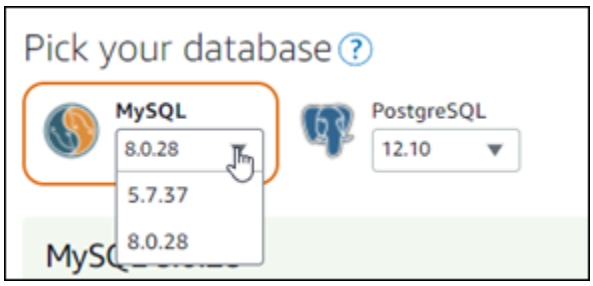
Lightsail データベースを作成する

Amazon Lightsail でマネージドデータベースを作成します。MySQL または PostgreSQL の最新メジャーバージョンから選択し、データベースをスタンダードプランまたは高可用性プランで設定できます。

> **Note**
>
> Lightsail のマネージドデータベースの詳細については、「データベースを選択する」を参照してください。

データベースを作成する

1. Lightsail コンソールにサインインします。
2. Lightsail のホームページで [データベース] タブを選択します。
3. [データベースの作成] を選択します。
4. データベースの AWS リージョン およびアベイラビリティーゾーンを選択します。
    1. [AWS リージョン とアベイラビリティーゾーンの変更] を選択し、リージョンを選択します。
    2. [アベイラビリティーゾーンの変更] を選択し、アベイラビリティーゾーンを選択します。
5. データベースのタイプを選択します。使用可能なエンジンオプションのいずれかから、ドロップダウンメニューを選択し、Lightsail でサポートされている最新のメジャーデータベースバージョンのどちらかを選択します。



6. 必要に応じて、以下のいずれかのオプションを選択します。

   - ログイン認証情報の指定 – 独自のデータベースユーザー名とパスワードを指定します。それ以外の場合は、Lightsail が代わりにユーザー名を指定し、強力なパスワードを作成します。
     - 独自のユーザー名を指定するには、 [Specify login credentials (ログイン認証情報の指定)] を選択し、テキストボックスにユーザー名を入力します。選択したデータベースエンジンに応じて、次の制約が適用されます。

       MySQL

       - MySQL に必要です。
       - 1 ~ 16 文字の英字または数字を使用することができます。
       - 1 字目は文字である必要があります。
       - 選択したデータベースエンジンの予約語は使用できません。MySQL の予約語の詳細については、MySQL 5.6、MySQL 5.7、または MySQL 8.0 のキーワードと予約語の記事を参照してください。

       PostgreSQL

       - PostgreSQL には必須です。
       - 1 ~ 63 文字の英字または数字が使用できます。
       - 1 字目は文字である必要があります。
       - 選択したデータベースエンジンの予約語は使用できません。PostgreSQL の予約語の詳細については、PostgreSQL 9.6、PostgreSQL 10、PostgreSQL 11、または PostgreSQL 12 の SQL キーワードの記事を参照してください。
     - 独自のパスワードを指定するには、[Create a strong password for me (強力なパスワードを作成する)] チェックボックスをオンにし、パスワードをテキストボックスに入力します。パスワードには「/」「"」または「@」を除く表示可能な任意の ASCII 文字を使用することができます。MySQL データベースの場合、パスワードには 8 ~ 41 文字の英数字を使用できます。PostgreSQL データベースの場合、パスワードには 8 ~ 128 文字の英数字を使用できます。

- マスターデータベース名の指定 – 独自のプライマリデータベース名を指定します。指定しないと、Lightsail で自動的に名前が指定されます。独自のプライマリデータベース名を指定するには、 [Specify the master database name (マスターデータベース名の指定)] を選択し、テキストボックスに名前を入力します。選択したデータベースエンジンに応じて、次の制約が適用されます。

  MySQL

  - 1～64 の文字または数字を使用する必要があります。
  - 先頭は文字を使用する必要があります。後続の文字には、英字、アンダースコア、または数字 (0～9) を使用することができます。
  - 選択したデータベースエンジンの予約語は使用できません。MySQL の予約語の詳細については、MySQL 5.6、MySQL 5.7、または MySQL 8.0 のキーワードと予約語の記事を参照してください。

  PostgreSQL

  - 1～63 文字の英字、数字、またはアンダースコアを使用する必要があります。
  - 先頭は文字を使用する必要があります。後続の文字には、英字、アンダースコア、または数字 (0～9) を使用することができます。
  - 選択したデータベースエンジンの予約語は使用できません。PostgreSQL の予約語の詳細については、PostgreSQL 9.6、PostgreSQL 10、PostgreSQL 11、または PostgreSQL 12 の SQL キーワードの記事を参照してください。

7. 高可用性データベースプランまたはスタンダードデータベースプランを選択します。

   高可用性プランでデータベースを作成すると、プライマリデータベースのほかに、フェイルオーバーのサポートとしてスタンバイ用のセカンダリデータベースが別のアベイラビリティーゾーンに作成されます。詳細については、「高可用性データベース」を参照してください。複数の異なる価格のデータベースバンドルオプションを利用できます。オプションごとにメモリ、処理、ストレージ容量、および転送レートのレベルが異なります。

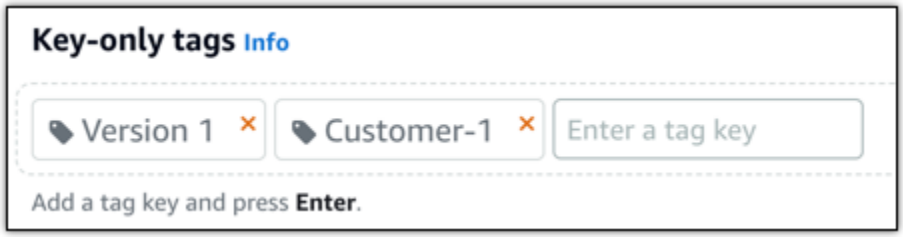
8. データベースの名前を入力します。

   リソース名:

   - Lightsail アカウントの各 AWS リージョン 内で一意である必要があります。
   - 2～255 文字を使用する必要があります。
   - 先頭と末尾は英数字または数字を使用する必要があります。
   - 英数字、数字、ピリオド、ダッシュ、アンダースコアを使用することができます。

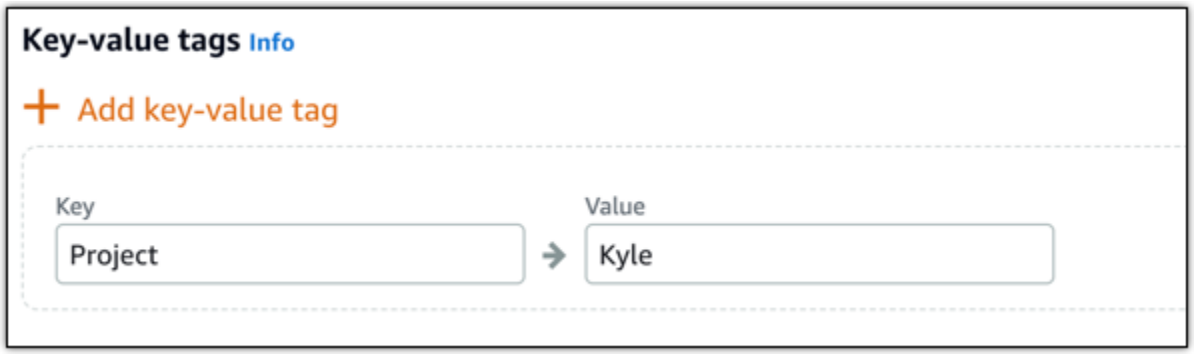
9. 以下のいずれかのオプションを選択して、データベースにタグを追加します。

   - [key-only タグの追加] または [key-only タグの編集] (タグが追加済みの場合)。タグキーのテキストボックスに新しいタグを入力し、Enter キーを押します。タグの入力を完了したら、[保存] を選択してタグを追加し、追加しない場合は、[キャンセル] を選択します。

   - [key-value タグの作成] から [キー] テキストボックスにキーを入力し、[値] テキストボックスに値を入力します。タグの入力を完了したら、[保存] を選択し、追加しない場合は、[キャンセル] を選択します。

     キーバリューのタグは、保存する際に一つずつ追加することができます。さらに key-value タグを追加するには、以上のステップを繰り返します。

   > **Note**
   >
   > 「キーのみ」のタグと「キーバリュー」のタグの詳細については、「タグ」を参照してください。

10. [データベースの作成] を選択します。

    数分以内に、Lightsail データベースの準備が完了します。データのインポートに関する設定を開始するか、データベースクライアントを使用して接続することができます。

次のステップ

Lightsail で新しいデータベースの使用を開始したら、データベースの管理に役立つ以下のガイドを参照してください。

- データベースのデータのインポートモードを設定する
- Amazon Lightsail のデータベースのパブリックモードを設定する
- データベースのパスワードを管理する
- MySQL データベースに接続する
- PostgreSQL データベースに接続する
- MySQL データベースにデータをインポートする
- データを PostgreSQL データベースにインポートする
- データベースのスナップショットを作成する

Lightsail MySQL データベースに接続する

Amazon Lightsail で MySQL マネージドデータベースを作成すると、標準の MySQL クライアントアプリケーションまたはユーティリティを使用して、このインスタンスに接続できます。Lightsail コンソールのデータベース管理ページから、データベースのエンドポイント、ポート、ユーザー名、およびパスワードを取得する必要があります。これらの値は、クライアントやウェブアプリケーションでデータベース接続を設定するときに指定します。

このガイドでは、必要な接続情報を取得する方法、およびマネージドデータベースに接続するように MySQL Workbench を設定する方法について説明します。

> **Note**
>
> PostgreSQL データベースへの接続の詳細については、「PostgreSQL データベースに接続する」を参照してください。

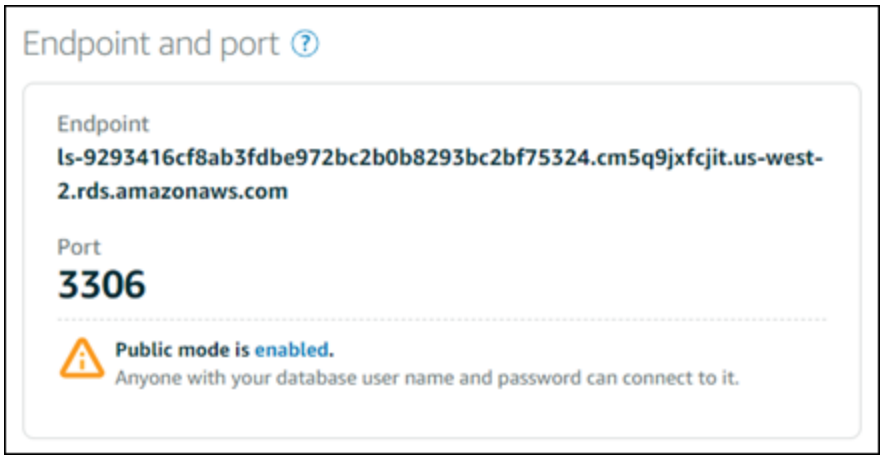
ステップ 1: MySQL データベース接続の詳細を取得する

Lightsail コンソールからデータベースのエンドポイントとポート情報を取得します。これらの情報は、データベースに接続するようにクライアントを設定するときに使用します。

データベース接続の詳細を取得するには

1. Lightsail コンソールにサインインします。
2. Lightsail のホームページで [データベース] タブを選択します。
3. 接続先のデータベースの名前を選択します。
4. [接続] タブの [Endpoint and port (エンドポイントとポート)] セクションで、エンドポイントとポートの情報を書き留めます。

   間違えて入力しないように、エンドポイントをクリップボードにコピーすることをお勧めします。そのためには、エンドポイントを強調表示し、Ctrl+C (Windows) または Cmd+C (macOS) を押してクリップボードにコピーします。次に、Ctrl+V または Cmd+V を押して貼り付けます。



5. [接続] タブの [ユーザー名とパスワード] セクションで、ユーザー名を確認して、[パスワード] セクションの [表示] を選択して現在のデータベースパスワードを表示します。

   マネージドパスワードは複雑であるため、間違って入力しないように、これもコピーして貼り付けることをお勧めします。マネージドパスワードを強調表示し、Ctrl+C (Windows) または Cmd+C (macOS) を押してクリップボードにコピーします。次に、Ctrl+V または Cmd+V を押して貼り付けます。

ステップ 2: MySQL データベースのパブリック可用性を設定する

データベースに外部から接続したり、データベースとは異なる AWS リージョン の Lightsail インスタンスから接続したりするには、データベースのパブリックモードを有効にする必要があります。パブリックモードを有効にすると、誰でもデータベースのユーザー名とパスワードを使用してデータベースに接続できます。データベースのパブリックでの可用性を設定するには、ガイドの「データベースのパブリックモードの設定」の手順に従います。

> **Note**
>
> データベースと同じリージョンにある Lightsail インスタンスのいずれかからデータベースに接続する場合は、ステップ 3 に進みます。

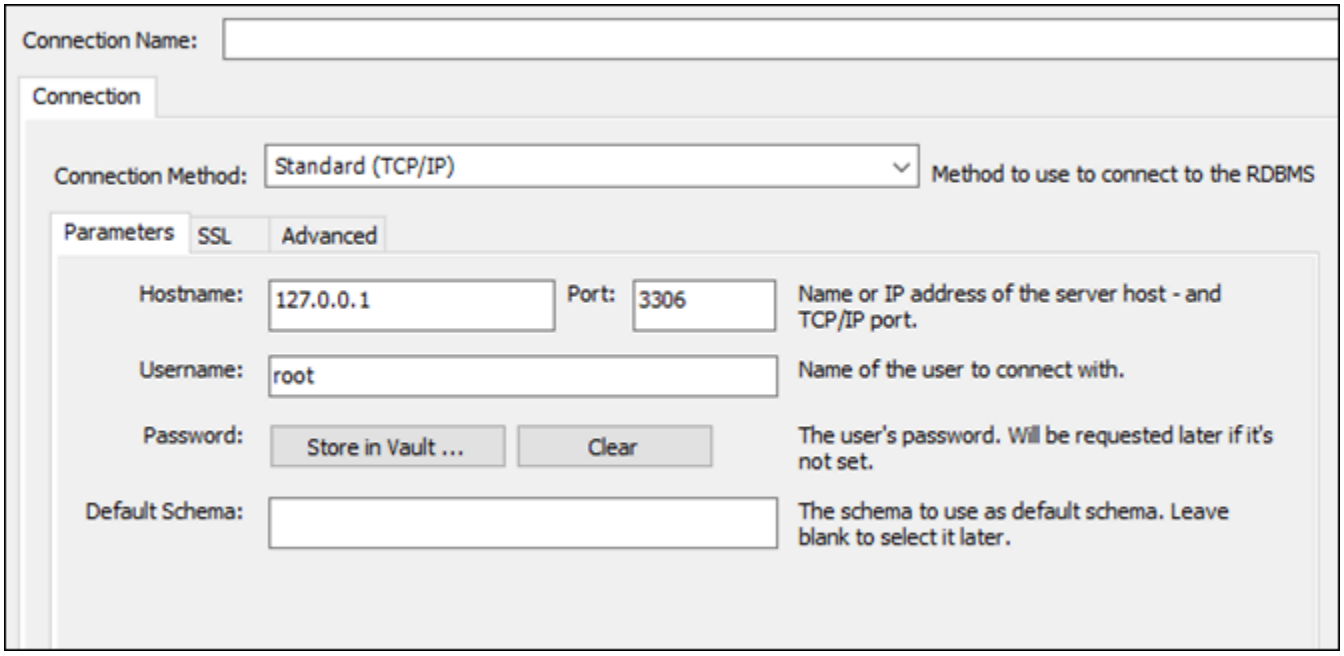
ステップ 3: MySQL データベースに接続するようにデータベースクライアントを設定する

MySQL データベースに接続するには、前に取得したエンドポイントとポートを使用するようにデータベースクライアントを設定します。以下のステップは、MySQL Workbench を設定する方法を示していますが、これらのステップは他のクライアントと同様の場合があります。

> **Note**
>
> MySQL Workbench の使用方法の詳細については、MySQL Workbench のマニュアルを参照してください。

データベースに接続するように MySQL Workbench を設定するには

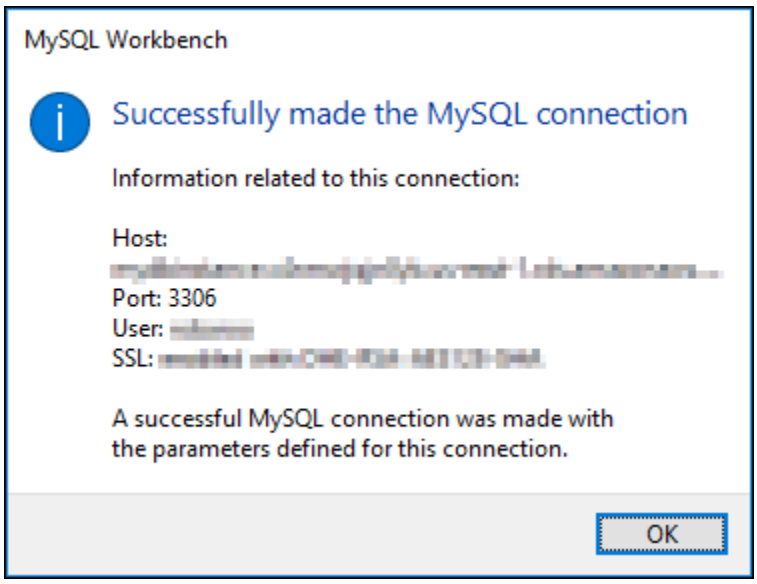
1. MySQL Workbench を開きます。
2. [Database (データベース)] メニュー、[Manage connections (接続の管理)] の順に選択します。
3. 表示されるフォームに以下の情報を入力します。



- 接続名: データベースに似た接続名を使用することをお勧めします。後で接続を識別しやすくなります。
- 接続方法: [標準 (TCP/IP)] を選択します。
- ポート — 先に取得したデータベースのポートを入力します。MySQL のデフォルトポートは 3306 です。
- ホスト名 — 先に取得したデータベースエンドポイントを入力します。データベースエンドポイントを Lightsail コンソールからコピーした場合は、まだクリップボード内に残っているため、Ctrl+V (Windows) または Cmd+V (macOS) を押して貼り付けます。
- ユーザー名: 前に取得したデータベースユーザー名を入力します。
- パスワード: [Store in Vault (ボールトに保存)] を選択します。表示されるウィンドウで、前に取得したデータベースパスワードを入力します。パスワードを Lightsail コンソールからコピーした場合は、まだクリップボード内に残っているため、Ctrl+V (Windows) または Cmd+V (macOS) を押して貼り付けます。[OK] をクリックしてパスワードを保存します。
- デフォルトスキーマ: このテキストボックスは空白のままにします。

4. [Test connection (テスト接続)] を選択し、クライアントからデータベースに接続できるかどうかを確認します。

   接続に成功すると、次の例に示すようなプロンプトが表示されます。情報を確認したら [OK] をクリックして閉じます。

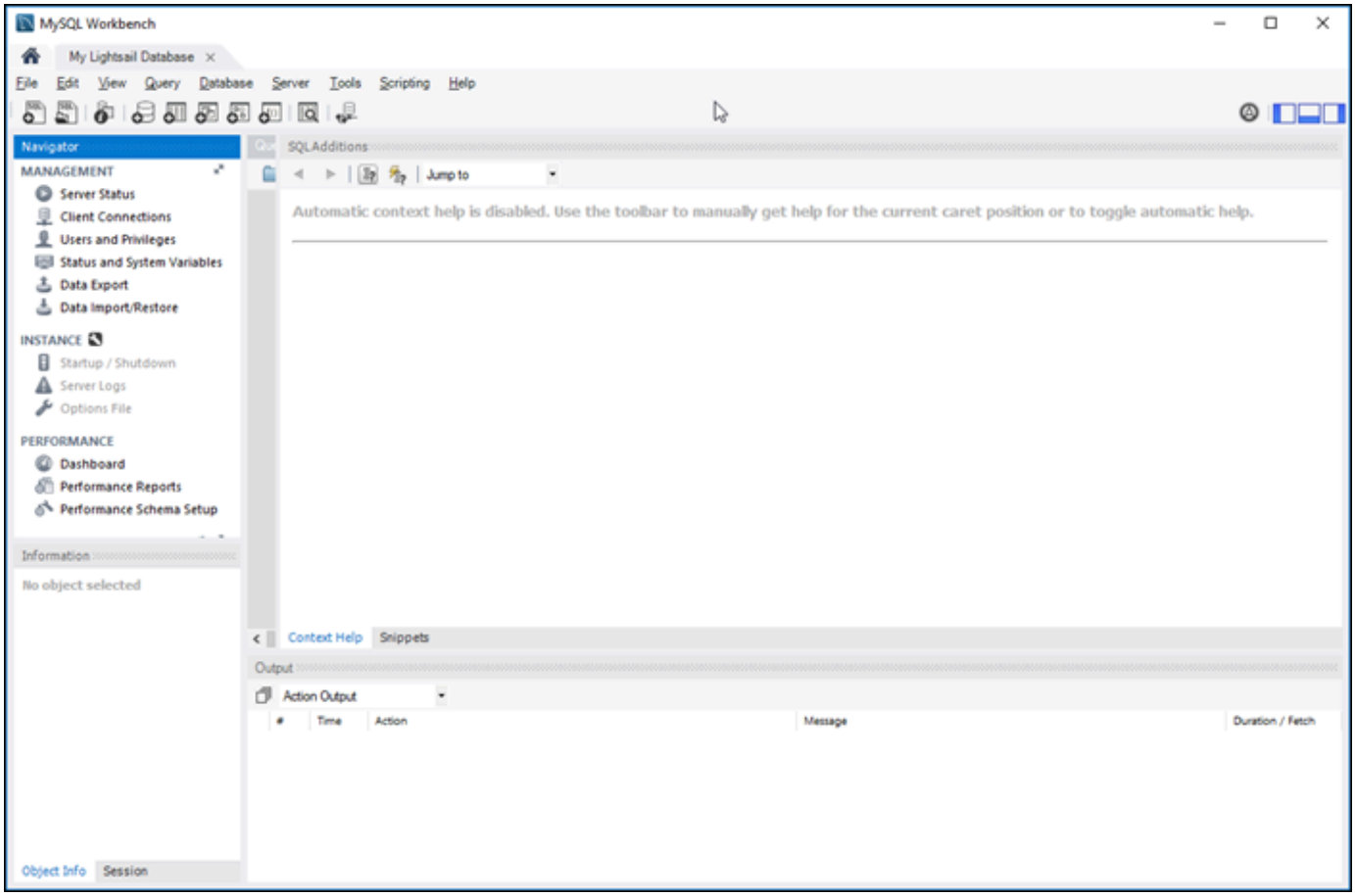


5. [New (新規)] を選択して新しい接続の詳細を保存し、[Close (閉じる)] を選択して接続管理ウィンドウを閉じます。

   新しいデータベース接続が、MySQL Workbench アプリケーションのホームページの [MySQL Connections (MySQL 接続)] セクションに表示されます。

6. データベースに接続するには、新しいデータベース接続を選択します。

   接続に成功すると、次の例に示すようなウィンドウが表示されます。



次のステップ

次のガイドは、Lightsail のデータベースにデータをインポートする際に役立ちます。

- MySQL データベースにデータをインポートする

SSL を使用して Lightsail MySQL データベースに接続する

Amazon Lightsail によって SSL 証明書が作成され、プロビジョニング時に MySQL マネージド型データベースにインストールされます。証明書は認証機関 (CA) によって署名され、なりすまし攻撃から保護するために、SSL 証明書の共通名 (CN) としてデータベースエンドポイントが含まれます。

Lightsail によって作成された SSL 証明書は信頼されたルートエンティティであり、ほとんどの場合は使用できますが、アプリケーションが証明書チェーンを受け入れていない場合は使用できない可能性があります。アプリケーションが証明書チェーンを受け入れていない場合は、AWS リージョン に接続している中間証明書の使用が必要になる場合があります。

マネージドデータベースの CA 証明書、サポートされる AWS リージョン 、アプリケーションの中間証明書のダウンロード方法の詳細については、「マネージドデータベースの SSL 証明書をダウンロード」を参照してください。

サポートされている接続

MySQL は以下のバージョンで安全な接続のため yaSSL を使用します。

- MySQL バージョン 5.7.19 および 5.7 以前のバージョン
- MySQL バージョン 5.6.37 および 5.6 以前のバージョン
- MySQL バージョン 5.5.57 および 5.5 以前のバージョン

MySQL は以下のバージョンで安全な接続のため OpenSSL を使用します。

- MySQL バージョン 8.0
- MySQL バージョン 5.7.21 以降の 5.7 バージョン
- MySQL バージョン 5.6.39 以降の 5.6 バージョン
- MySQL バージョン 5.5.59 以降の 5.5 バージョン

MySQL マネージド型データベースは、Transport Layer Security (TLS) バージョン 1.0、1.1、1.2 をサポートしています。以下のリストでは、MySQL バージョンがサポートする TLS を示しています。

- MySQL 8.0 - TLS 1.0、TLS 1.1、および TLS 1.2
- MySQL 5.7 - TLS 1.0 と TLS 1.1 TLS 1.2 は、MySQL 5.7.21 以降でのみサポートされています。
- MySQL 5.6 - TLS1.0
- MySQL 5.5 - TLS1.0

前提条件

- データベースへの接続に使用するコンピュータに MySQL サーバーをインストールします。詳細については、MySQL ウェブサイトの「MySQL Community Server download」を参照してください。
- データベースの該当する証明書をダウンロードします。詳細については、「マネージドデータベースの SSL 証明書をダウンロード」を参照してください。

SSL を使用して MySQL データベースに接続する

SSL を使用して MySQL データベースに接続するには、以下の手順を実行します。

1. ターミナルまたはコマンドプロントウィンドウを開きます。
2. MySQL データベースのバージョンに応じて、以下のコマンドのいずれかを入力します。

   - MySQL 5.7 以降のデータベースに接続するには、以下のコマンドを入力します。

     ```
     mysql -h DatabaseEndpoint --ssl-ca=/path/to/certificate/rds-combined-ca-bundle.pem --ssl-mode=VERIFY_IDENTITY -u UserName -p
     ```

     コマンドを、以下のように置き換えます。

     - *DatabaseEndpoint* は、データベースのエンドポイントに置き換えます。
     - */path/to/certificate/rds-combined-ca-bundle.pem* は、データベースの証明書をダウンロードして保存したローカルパスに置き換えます。
     - *UserName* は、データベースのユーザー名に置き換えます。

     例:

     ```
     mysql -h ls-1c51a7c70a4fb55e542829a4e4e0d735ba42.czowadgeezqi.us-west-2.rds.amazonaws.com --ssl-ca=/home/ec2-user/rds-combined-ca-bundle.pem --ssl-mode=VERIFY_IDENTITY -u dbmasteruser -p
     ```

   - MySQL 6.7 以降のデータベースに接続するには、以下のコマンドを入力します。

     ```
     mysql -h DatabaseEndpoint --ssl-ca=/path/to/certificate/rds-combined-ca-bundle.pem --ssl-verify-server-cert -u UserName -p
     ```

     コマンドを、以下のように置き換えます。

     - *DatabaseEndpoint* は、データベースのエンドポイントに置き換えます。
     - */path/to/certificate/rds-combined-ca-bundle.pem* は、データベースの証明書をダウンロードして保存したローカルパスに置き換えます。
     - *UserName* は、データベースのユーザー名に置き換えます。

     例:

```
mysql -h ls-1c51a7c70a4fb55e542829a4e4e0d735ba42.czowadgeezqi.us-
west-2.rds.amazonaws.com --ssl-ca=/home/ec2-user/rds-combined-ca-bundle.pem --
ssl-verify-server-cert -u dbmasteruser -p
```

3. プロンプトが表示されたら、前のコマンドで指定したデータベースユーザーのパスワードを入力し、Enter キーを押します。

   以下の例のような結果が表示されるはずです。

```
[ec2-user@ip-172-26-5-44 ~]$ mysql -h ls-1c51a[redacted]829a4e4e0d735ba42.czowadgeezqi.us-west-2.rds.amazonaws
.com --ssl-ca=/home/ec2-user/rds-ca-2015-root.pem --ssl-verify-server-cert -u dbmasteruser -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 2727
Server version: 8.0.16 Source distribution

Copyright (c) 2000, 2018, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql>
```

4. 「**status**」と入力し、Enter キーを押して、接続のステータスを表示します。

   [SSL] の横で [Cipher in use is (使用中の暗号)] に値が表示されている場合、接続は暗号化されています。

```
mysql> status
--------------
mysql  Ver 14.14 Distrib 5.5.62, for Linux (x86_64) using readline 5.1

Connection id:          2727
Current database:
Current user:           dbmasteruser@172.26.5.44
SSL:                    Cipher in use is DHE-RSA-AES256-SHA
Current pager:          stdout
Using outfile:          ''
Using delimiter:        ;
Server version:         8.0.16 Source distribution
Protocol version:       10
Connection:             ls-1c51a7beedc70a4fb55e542829a4e4e0d735ba42.czowadgeezqi.us-west-2.rds.amazonaws.com via TCP/I
P
Server characterset:    utf8mb4
Db     characterset:    utf8mb4
Client characterset:    utf8
Conn.  characterset:    utf8
TCP port:               3306
Uptime:                 9 days 16 hours 24 min 33 sec

Threads: 3  Questions: 557480  Slow queries: 0  Opens: 242  Flush tables: 3  Open tables: 146  Queries per second avg:
 0.666
--------------
```

Lightsail PostgreSQL データベースに接続する

Amazon Lightsail で PostgreSQL マネージドデータベースを作成すると、標準の PostgreSQL クライアントアプリケーションまたはユーティリティを使用して、このインスタンスに接続できます。Lightsail コンソールのデータベース管理ページから、データベースのエンドポイント、ポート、ユーザー名、およびパスワードを取得する必要があります。これらの値は、クライアントやウェブアプリケーションでデータベース接続を設定するときに指定します。

このガイドでは、必要な接続情報を取得する方法、およびマネージドデータベースに接続するように pgAdmin クライアントを設定する方法について説明します。

> **Note**
>
> MySQL データベースへの接続の詳細については、「MySQL データベースに接続する」を参照してください。

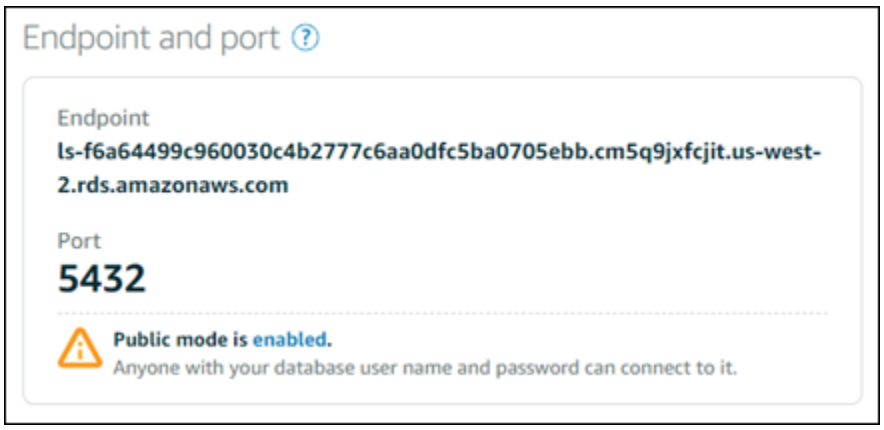
ステップ 1: PostgreSQL データベース接続の詳細を取得する

Lightsail コンソールからデータベースのエンドポイントとポート情報を取得します。これらの情報は、データベースに接続するようにクライアントを設定するときに使用します。

データベース接続の詳細を取得するには

1. Lightsail コンソールにサインインします。
2. Lightsail のホームページで [データベース] タブを選択します。
3. 接続先のデータベースの名前を選択します。
4. [接続] タブの [Endpoint and port (エンドポイントとポート)] セクションで、エンドポイントとポートの情報を書き留めます。

   間違えて入力しないように、エンドポイントをクリップボードにコピーすることをお勧めします。そのためには、エンドポイントを強調表示し、Ctrl+C (Windows) または Cmd+C (macOS) を押してクリップボードにコピーします。次に、Ctrl+V または Cmd+V を押して貼り付けます。



5. [接続] タブの [ユーザー名とパスワード] セクションで、ユーザー名を確認して、[パスワード] セクションの [表示] を選択して現在のデータベースパスワードを表示します。

   マネージドパスワードは複雑であるため、間違って入力しないように、これもコピーして貼り付けることをお勧めします。マネージドパスワードを強調表示し、Ctrl+C (Windows) または Cmd+C (macOS) を押してクリップボードにコピーします。次に、Ctrl+V または Cmd+V を押して貼り付けます。

ステップ 2: PostgreSQL データベースのパブリック可用性を設定する

データベースに外部から接続したり、データベースとは異なるリージョンの Lightsail インスタンスから接続するには、データベースのパブリックモードを有効にする必要があります。パブリックモードを有効にすると、誰でもデータベースのユーザー名とパスワードを使用してデータベースに接続できます。データベースのパブリックでの可用性を設定するには、ガイドの「データベースのパブリックモードの設定」の手順に従います。

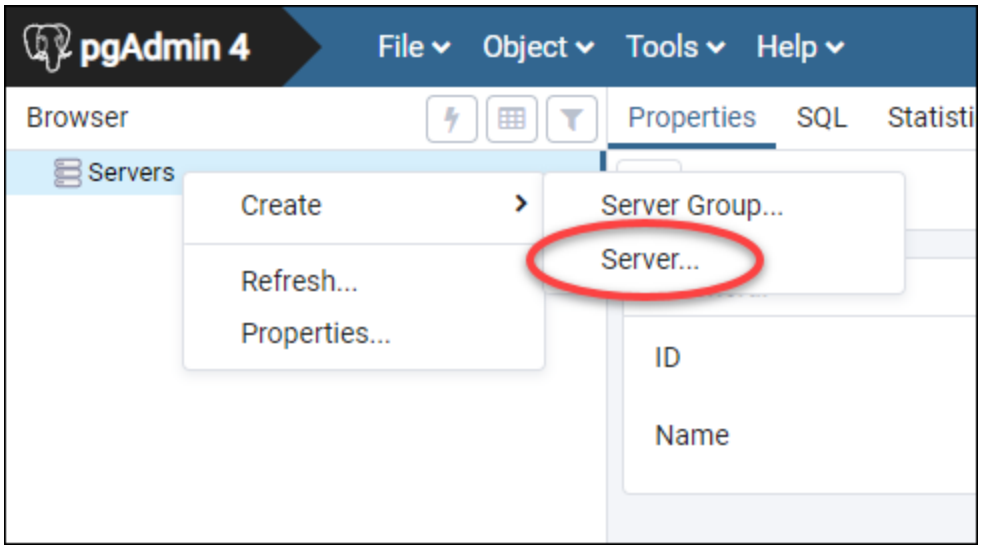
> **Note**
>
> データベースと同じリージョンにある Lightsail インスタンスのいずれかからデータベースに接続する場合は、ステップ 3 に進みます。

ステップ 3: PostgreSQL データベースに接続するようにデータベースクライアントを設定する

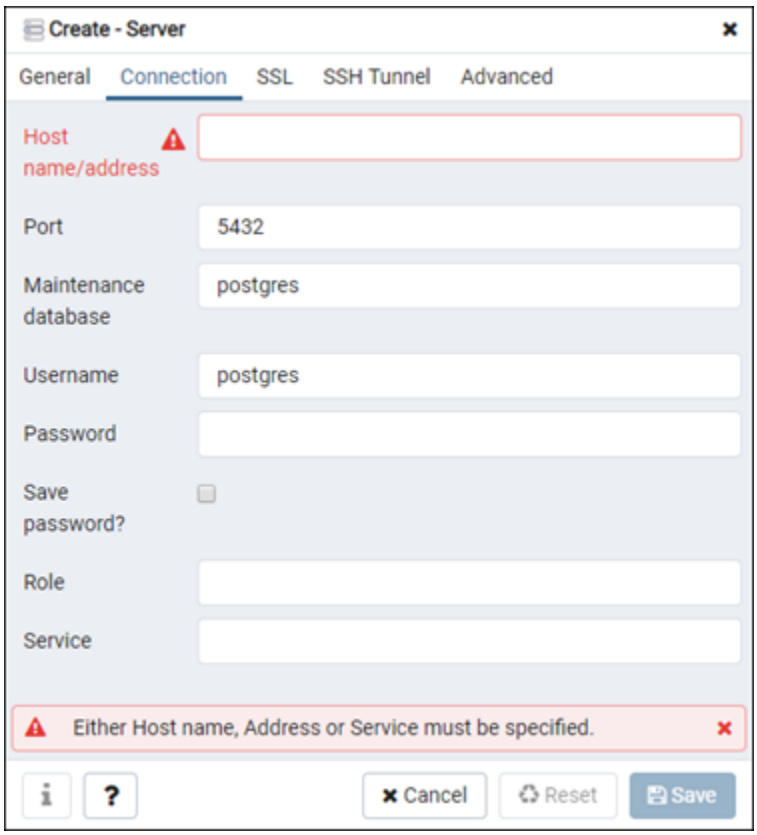
PostgreSQL データベースに接続するには、前に取得したエンドポイントとポートを使用するようにデータベースクライアントを設定します。以下のステップは、pgAdmin を設定する方法を示していますが、これらのステップは他のクライアントと同様の場合があります。

> **Note**
>
> pgAdmin を使用する方法の詳細については、「pgAdmin ドキュメント」を参照してください。

データベースに接続するように pgAdmin を設定するには

1. [pgAdmin] を開きます。
2. 左側のナビゲーションメニュー から [サーバー] を右クリックします。
3. [作成]、[サーバー] の順に選択します。
4. 

5. [Create-Server] フォームに、サーバー名を入力します。データベースに似た接続名を使用することをお勧めします。後で接続を識別しやすくなります。
6. [接続] タブを選択し、表示されるフォームに、次の情報を入力します。



- ホスト名/アドレス — 先に取得したデータベースエンドポイントを入力します。データベースエンドポイントを Lightsail コンソールからコピーした場合は、まだクリップボード内に残っているため、Ctrl+V (Windows) または Cmd+V (macOS) を押して貼り付けます。
- ポート — 先に取得したデータベースのポートを入力します。PostgreSQL のデフォルトポートは 5432 です。
- メンテナンスデータベース — クライアントが接続する初期データベースの名前を指定します。これは、Lightsail に PostgreSQL データベースを作成したときに指定したプライマリデータベースの名前です。

  プライマリデータベースの名前を覚えていない場合は、`postgres` を入力します。すべての PostgreSQL のマネージド型データベースには接続可能な `postgres` データベースがあり、後で PostgreSQL マネージド型データベースの他のすべてのデータベースにアクセスできるようになります。
- ユーザー名: 前に取得したデータベースユーザー名を入力します。
- パスワード — 先に取得したデータベースのパスワードを入力します。パスワードを Lightsail コンソールからコピーした場合は、まだクリップボード内に残っているため、Ctrl+V (Windows) または Cmd+V (macOS) を押して貼り付けます。[パスワードを保存] を選択してパスワードを保存します。

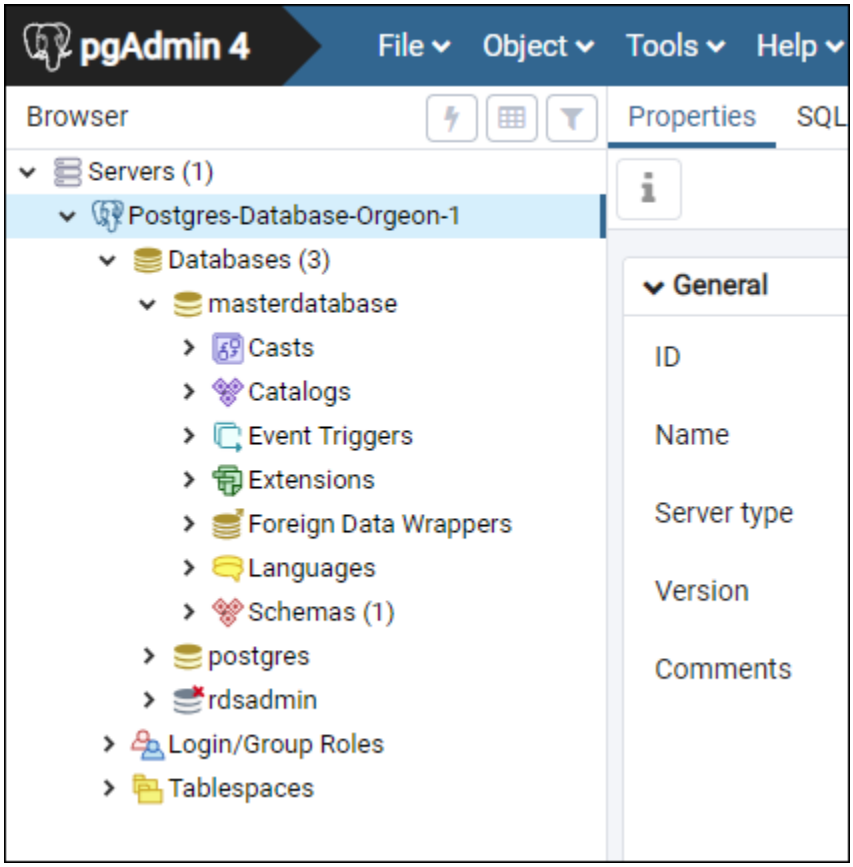
- ロールとサービス — これらのフィールドは空白のままにしておきます。

7. [保存] を選択して新しいサーバーの詳細を保存します。

   サーバーセクションの pgAdmin アプリケーションの左側のナビゲーションメニューに新しいデータベース接続が表示されます。

8. データベースに接続するには、新しいデータベース接続をダブルクリックします。

   接続に成功すると、そのデータベースの使用可能なリソースのリストが表示されます。



次のステップ

次のガイドは、Lightsail のデータベースにデータをインポートする際に役立ちます。

- データを PostgreSQL データベースにインポートする

SSL を使用して Lightsail PostgreSQL データベースに接続する

Amazon Lightsail によって SSL 証明書が作成され、プロビジョニング時に PostgreSQL (Postgres) マネージド型データベースにインストールされます。証明書は認証機関 (CA) によって署名され、なりすまし攻撃から保護するために、SSL 証明書の共通名 (CN) としてデータベースエンドポイントが含まれます。

Lightsail によって作成された SSL 証明書は信頼されたルートエンティティであり、ほとんどの場合は使用できますが、アプリケーションが証明書チェーンを受け入れていない場合は使用できない可能性があります。アプリケーションが証明書チェーンを受け入れていない場合は、AWS リージョン に接続している中間証明書の使用が必要になる場合があります。

マネージドデータベースの CA 証明書、サポートされる AWS リージョン 、アプリケーションの中間証明書のダウンロード方法の詳細については、「マネージドデータベースの SSL 証明書をダウンロード」を参照してください。

前提条件

- データベースへの接続に使用するコンピュータに PostgreSQL サーバーをインストールします。詳細については、Postgres ウェブサイトの「PostgreSQL Downloads」を参照してください。
- データベースの該当する証明書をダウンロードします。詳細については、「マネージドデータベースの SSL 証明書をダウンロード」を参照してください。

SSL を使用して Postgres データベースに接続する

SSL を使用して Postgres データベースに接続するには、以下の手順を実行します。

1. ターミナルまたはコマンドプロントウィンドウを開きます。
2. PostgreSQL データベースに接続するには、以下のコマンドを入力します。

```
psql -h DatabaseEndpoint -p 5432 "dbname=DatabaseName user=UserName sslrootcert=/
path/to/certificate/rds-combined-ca-bundle.pem sslmode=verify-full"
```

コマンドを、以下のように置き換えます。

- *DatabaseEndpoint* は、データベースのエンドポイントに置き換えます。
- *DatabaseName* は、接続するデータベースの名前に置き換えます。

- *UserName* は、データベースのユーザー名に置き換えます。
- */path/to/certificate/rds-combined-ca-bundle.pem* は、データベースの証明書をダウンロードして保存したローカルパスに置き換えます。

例:

```
psql -h ls-8e81e07f8b821917b11e1c6a0e26cb73c203.czowadgeezqi.us-
west-2.rds.amazonaws.com -p 5432 "dbname=dbmaster user=dbmasteruser sslrootcert=/
home/ec2-user/rds-combined-ca-bundle.pem sslmode=verify-full"
```

3. プロンプトが表示されたら、前のコマンドで指定したデータベースユーザーのパスワードを入力し、Enter キーを押します。

   次の例のような結果が表示されます。[SSL connection (SSL 接続)] に値が表示されている場合、接続は暗号化されています。

Lightsail データベースを削除する

Amazon Lightsail でマネージドデータベースが不要になった場合は、これを削除します。データベースを削除すると、データベースに対する課金も停止します。

> **Note**
>
> 削除したデータベースは復元できません。このガイドで示す手順の一環としてデータベースの最終スナップショットを作成できます。または、削除プロセスとは関係なしにスナップショットを作成することもできます。詳細については、「データベースのスナップショットを作成する」を参照してください。

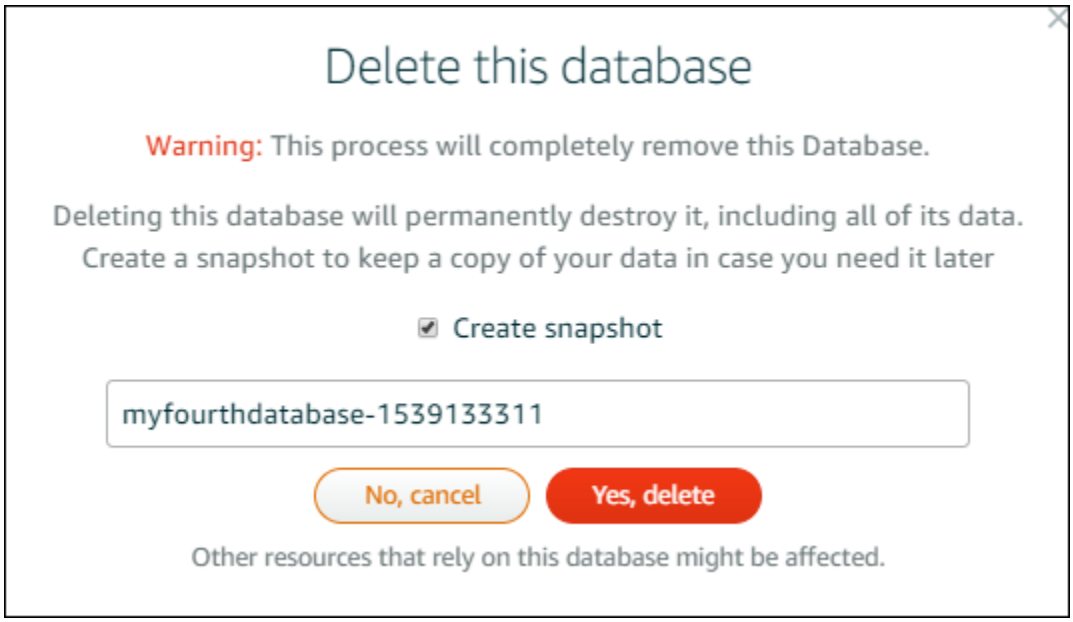
データベースを削除するには

1. Lightsail コンソールにサインインします。

2. Lightsail のホームページで [データベース] タブを選択します。
3. 削除するデータベースの名前を選択します。
4. [削除] タブを選択します。
5. データベースを削除する前に最終スナップショットを作成するには、[削除前にスナップショットを作成する] をオンにします。次に、スナップショットの名前を入力します。

   リソース名:

   - Lightsail アカウントの各 AWS リージョン 内で一意である必要があります。
   - 2〜255 文字を使用する必要があります。
   - 先頭と末尾は英数字または数字を使用する必要があります。
   - 英数字、数字、ピリオド、ダッシュ、アンダースコアを使用することができます。
6. [データベースの削除] を選択します。
7. [はい、削除します] を選択して削除を確定します。



削除する前にスナップショットを作成することを選択した場合は、Lightsail ホームページのスナップショットタブで表示されます。

Lightsail データベースのデータのインポートモードを設定する

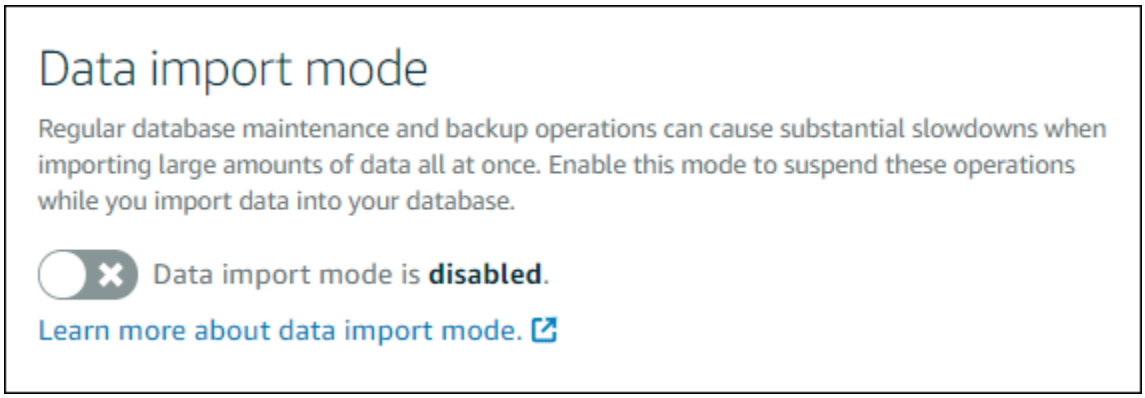
大量のデータを一度にすべてインポートする場合、定期的なデータベースのバックアップオペレーションのせいで大幅な遅延や速度の低下が生じることがあります。大量のデータをインポートすると

きは、バックアップオペレーションを停止するように Amazon Lightsail マネージドデータベースのデータのインポートモードを設定できます。

> **Important**
>
> データのインポートモードが有効になるとすべての緊急復元バックアップが削除されます。データのインポートモードが有効になる前にバックアップする場合は、データベースのスナップショットを作成します。詳細については、「データベースのスナップショットを作成する」を参照してください。

データベースのデータのインポートモードを設定するには

1. Lightsail コンソールにサインインします。
2. Lightsail のホームページで [データベース] タブを選択します。
3. データのインポートモードを設定するデータベースの名前を選択します。
4. [接続] タブの [Data import mode (データのインポートモード)] セクションで、トグルを使用してデータのインポートモードをオンにします。同様に、インポートの完了後は、トグルを使用してオフにします。



これでデータのインポートモードが有効になり、データベースのバックアップオペレーションが停止されます。データのインポートモードは一時的に有効にすることをお勧めします。大量のデータをデータベース内にインポートする必要がある場合に限り、使用してください。インポートが完了したらすぐにデータのインポートモードを無効にして、バックアップオペレーションを回復します。

> **Note**
>
> インポートするデータの量によっては、インポートが遅くなることがあります。詳細については、「データのインポートの最適化」を参照してください。

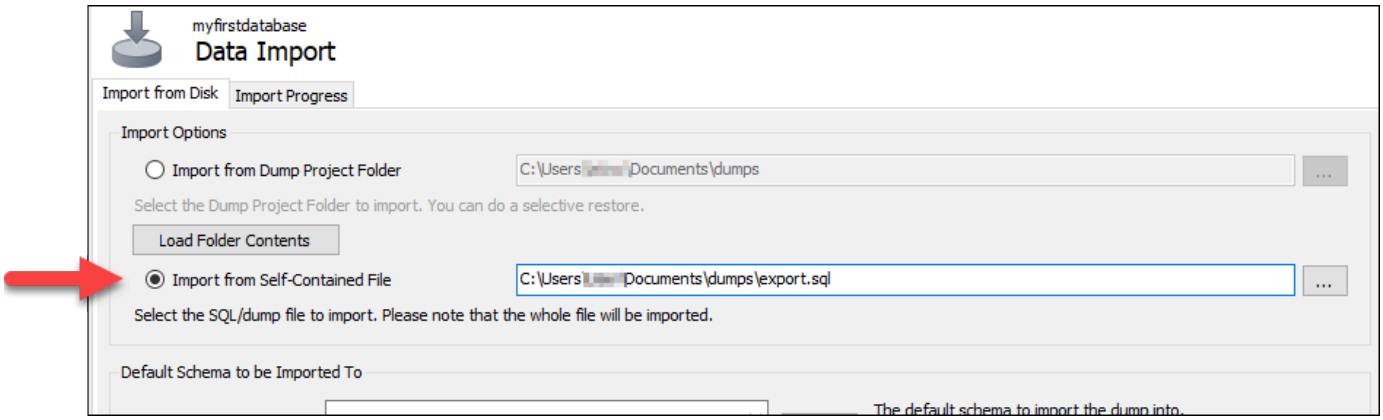
Lightsail で MySQL データベースにデータをインポートする

MySQL Workbench を使用して、Amazon Lightsail の MySQL マネージドデータベースに SQL ファイル (.SQL) をインポートできます。

> **Note**
>
> MySQL Workbench をデータベースに接続する方法については、「MySQL データベースに接続する」を参照してください。

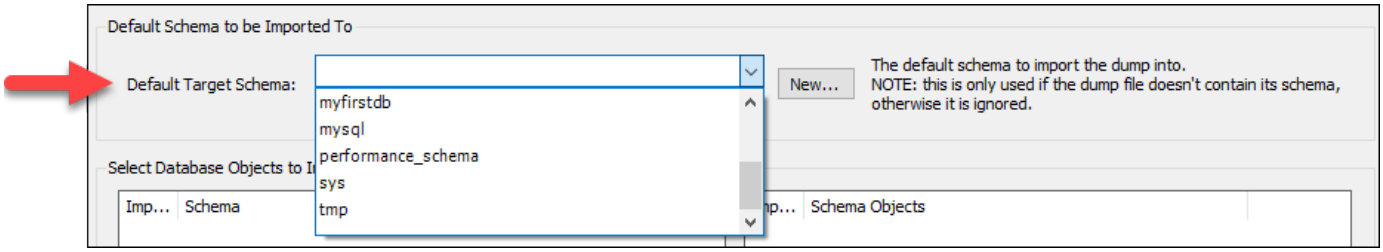
データベースにデータをインポートするには

1. MySQL Workbench を開きます。
2. MySQL 接続のリストで、MySQL マネージドデータベースを選択します。
3. 左のナビゲーションメニューから [Data Import/Restore (データのインポート/復元)] を選択します。
4. [Data Import] ペインで、[Import Option] セクションにある [Import from Self-Contained File] (自己完結型ファイルからインポート) を選択します。



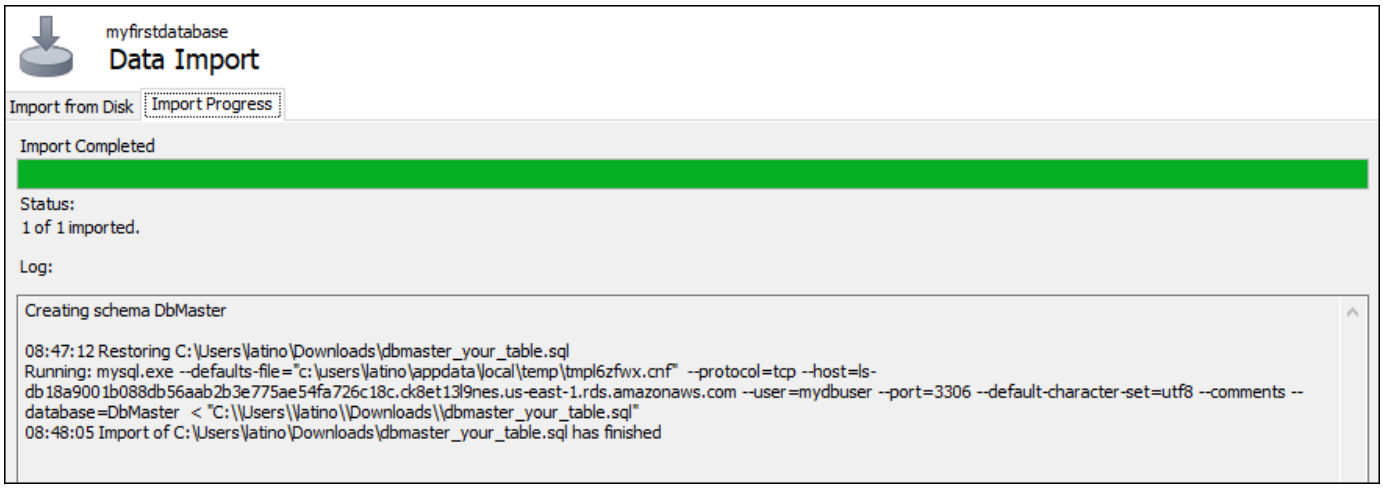
5. 省略記号ボタンを選択し、インポートする .SQL ファイルのローカルドライブを参照します。
6. インポートする .SQL ファイルを選択し、[Open (開く)] を選択します。

7. [Default Target Schema (デフォルトのターゲットスキーマ)] ドロップダウンメニューを選択し、ファイルをインポートする先の既存のデータベースを選択します。[New (新規)] を選択して新しいデータベースを作成することもできます。



8. [Start Import (インポートの開始)] を選択してインポートを開始します。

   .SQL ファイルのサイズに応じて、完了するまで数分かかる場合があります。インポートが完了した後、次のようなメッセージが表示されます。



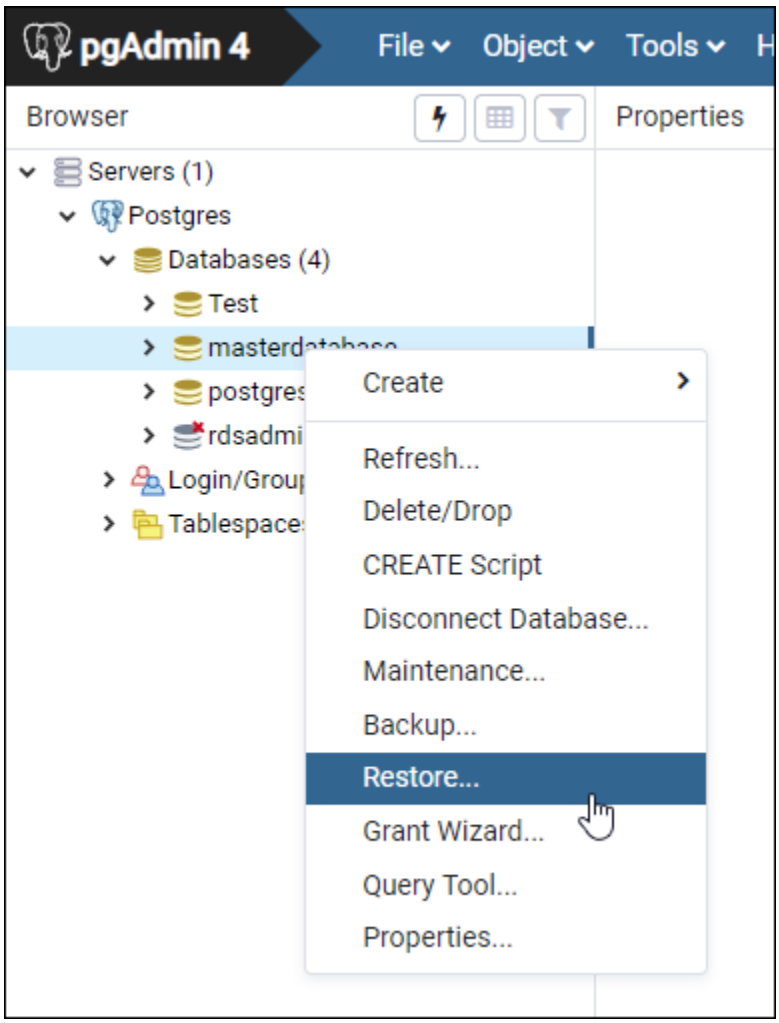
Lightsail の PostgreSQL データベースにデータをインポートする

pgAdmin を使用して、Amazon Lightsail の PostgreSQL マネージドデータベースにデータベースバックアップファイルをインポートできます。

> **Note**
>
> pgAdmin をデータベースに接続する方法については、「PostgreSQL データベースに接続する」を参照してください。他のデータベースにインポートできる PostgreSQL データベースバックアップを作成する方法について詳細は、pgAdmin ドキュメントの「バックアップダイアログ」を参照してください。

データベースにバックアップファイルをインポートするには

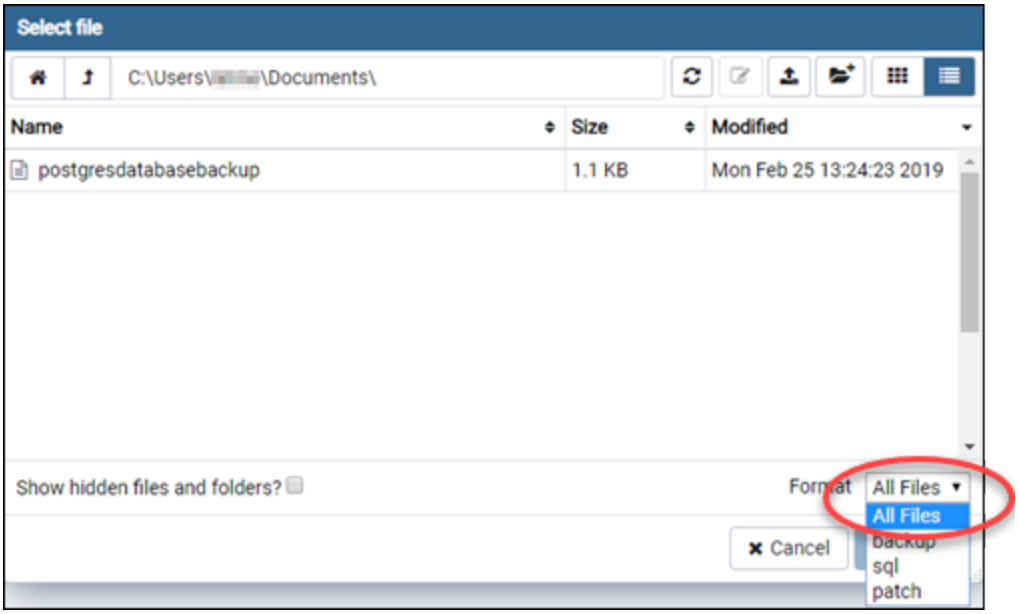
1. [pgAdmin] を開きます。
2. サーバー接続のリストで、Amazon Lightsail の PostgreSQL マネージドデータベースをダブルクリックして接続します。
3. Databases ノードを展開します。
4. データベースバックアップファイルからのデータをインポートするデータベースを右クリックして、その後 [Restore (復元)] を選択します。



5. [Restore (復元)] フォームで、次のフィールドに入力します。

   - Format (形式) - バックアップファイルの形式を選択します。
   - Filename (ファイル名) - 省略記号アイコンを選択し、ローカルドライブのデータベースバックアップファイルを見つけて選択します。ファイルがハイライトされたら、[Select (選択)] を選択して、[Restore (復元)] プロンプトに戻ります。

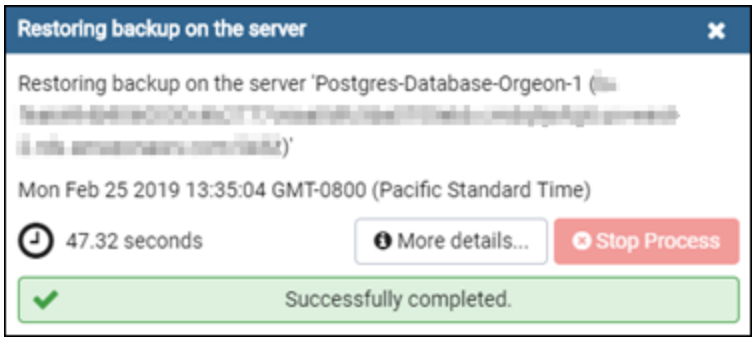
> **Note**
>
> [Format (形式)] ドロップダウンメニューを選択し、[All files (すべてのファイル)] を選択してローカルドライブにあるすべてのファイル形式を表示します。バックアップファイルは、デフォルトで選択されているファイル形式 (sql) とは異なる形式で保存されている場合があります。



- Number of jobs (ジョブの数) および Role name (ロール名) - これらのフィールドは空白のままにしておきます。

6. インポートを開始するには [Restore (復元)] を選択します。

   データベースバックアップファイルのサイズに応じて、完了するまで数分かかる場合があります。インポートが完了した後、次のようなメッセージが表示されます。



Lightsail データベースのログと履歴を表示する

Amazon Lightsail コンソールでデータベースのログと変更履歴を表示します。データベースのログは、データベースの問題の診断に役立つ情報を提供します。同様に、データベースの履歴は、データベースに加えられた変更を示します。これにより、最近の変更と問題の関連性を確認できます。

データベースのログを表示するには

1. Lightsail コンソールにサインインします。
2. Lightsail のホームページで [データベース] タブを選択します。
3. ログを表示するデータベースの名前を選択します。
4. [Logs and history (ログと履歴)] タブを選択します。

   このページには、データベースのログとデータベースに加えられた変更の履歴が表示されます。
5. データベースのログを選択します。以下のデータベースのログを利用できます。

   MySQL データベースのログ

   - エラーログ — mysqld の起動時間およびシャットダウン時間の記録。これには、サーバーの起動とシャットダウン時、およびサーバーの実行中に発生するエラー、警告、注意などの診断メッセージも含まれます。詳細については、MySQL 5.6、MySQL 5.7、または MySQL 8.0 のドキュメントでエラーログに関する記事を参照してください。
   - 全般ログ — mysqld の動作に関する全般ログ。サーバーは、クライアントが接続または切断したときにこのログに情報を書き込みます。また、クライアントから受信した各 SQL ステートメントをログに記録します 詳細については、MySQL 5.6、MySQL 5.7、または MySQL 8.0 のドキュメントで全般クエリログに関する記事を参照してください。
   - スロークエリログ — 実行に long_query_time 秒を超える時間がかかり、検証に min_examined_row_limit 行以上を要した SQL ステートメントのレコード。詳細については、MySQL 5.6、MySQL 5.7、または MySQL 8.0 のドキュメントでスロークエリログに関する記事を参照してください。

   > **Note**
   >
   > MySQL データベースの一般ログとスロークエリログはデフォルトで無効になっています。これらのログを有効にして、データの収集を開始するには、いくつかのデータベー

スパラメータを更新します。詳細については、「Amazon Lightsail で MySQL データベースの一般ログとスロークエリログを有効にする」を参照してください。

PostgreSQL データベースのログ

- Postgres ログ - データベースの起動時間およびシャットダウン時間の記録。これには、データベースの起動とシャットダウン時、およびデータベースが実行中に発生するエラー、警告、通知、デバッグなどの診断メッセージも含まれます。詳細については、PostgreSQL 9.6、PostgreSQL 10、PostgreSQL 11、または PostgreSQL 12 ドキュメントのエラーレポートとログ記録の記事を参照してください。

トピック

- Lightsail MySQL データベースの一般ログとスロークエリログを有効にする

Lightsail MySQL データベースの一般ログとスロークエリログを有効にする

Amazon Lightsail の MySQL データベースでは、一般ログとスロークエリログはデフォルトで無効になっています。これらのログを有効にして、データの収集を開始するには、いくつかのデータベースパラメータを更新します。Lightsail API、 AWS Command Line Interface (AWS CLI) 、または SDKs を使用してデータベースパラメータを更新します。このガイドでは AWS CLI を使用して、データベースのパラメータを更新し一般ログとスロークエリログを有効にする方法を説明します。 また、一般ログとスロークエリログを制御するいくつかの追加オプションと、ログデータの保持期間がどのように処理されるかについても説明します。

前提条件

まだ AWS CLI をインストールして設定していない場合は、インストールして設定します。詳細については、「Amazon Lightsail と連携AWS Command Line Interfaceするように を設定する」を参照してください。

Lightsail コンソールで一般ログとスロークエリログを有効にする

Lightsail コンソールで一般ログとスロークエリログを有効にするには、 `general_log` および `slow_query_log` データベースパラメータを の値で更新し`1`、 `log_output`パラメータを の値で更新する必要がありま`FILE`す。

Lightsail コンソールで一般ログとスロークエリログを有効にするには

1. ターミナルまたはコマンドプロントウィンドウを開きます。
2. 次のコマンドを入力して、general_log パラメータの値を 1 に更新します。これは true または有効です。

```
aws lightsail update-relational-database-parameters --
region Region --relational-database-name DatabaseName --parameters
 "parameterName=general_log,parameterValue=1,applyMethod=pending-reboot"
```

   コマンドを、以下のように置き換えます。

   - *DatabaseName* をデータベースの名前に置き換えます。
   - *Region* は、データベースの AWS リージョン に置き換えます。
3. 次のコマンドを入力して、slow_query_log パラメータの値を 1 に更新します。これは true または有効です。

```
aws lightsail update-relational-database-parameters --
region Region --relational-database-name DatabaseName --parameters
 "parameterName=slow_query_log,parameterValue=1,applyMethod=pending-reboot"
```

   コマンドを、以下のように置き換えます。

   - *DatabaseName* をデータベースの名前に置き換えます。
   - *Region* は、データベースの AWS リージョン に置き換えます。
4. 次のコマンドを入力して、 log_outputパラメータを の値に更新します。これによりFILE、ログデータがシステムファイルに書き込まれ、Lightsail コンソールに表示されるようになります。

```
aws lightsail update-relational-database-parameters --
region Region --relational-database-name DatabaseName --parameters
 "parameterName=log_output,parameterValue=FILE,applyMethod=pending-reboot"
```

   コマンドを、以下のように置き換えます。

   - *DatabaseName* をデータベースの名前に置き換えます。
   - *Region* は、データベースの AWS リージョン に置き換えます。
5. 次のコマンドを入力してデータベースを再起動し、変更を反映させます。

```
aws lightsail reboot-relational-database --region Region --relational-database-
name DatabaseName
```

コマンドを、以下のように置き換えます。

- *DatabaseName* をデータベースの名前に置き換えます。
- *Region* は、データベースの AWS リージョン に置き換えます。

この時点で、データベースは再起動中使用できなくなります。数分待ってから Lightsail コンソールにサインインして、データベースの一般ログとスロークエリログを表示します。詳細については、「Amazon Lightsail でのデータベースログと履歴の表示」を参照してください。

> **Note**
>
> データベースパラメータの更新の詳細については、「Amazon Lightsail でのデータベースパラメータの更新」を参照してください。

データベースログのその他のオプションの制御

MySQL の一般ログとスロークエリログのその他のオプションを制御するには、次のパラメータを更新します。

- `log_output` — このパラメータは `TABLE` に設定します。一般クエリが `mysql.general_log` テーブルに書き込まれ、スロークエリは `mysql.slow_log` テーブルに書き込まれます。`log_output` パラメータを `NONE` に設定して、ログ記録を無効にすることもできます。

  > **Note**
  >
  > `log_output` パラメータを に設定すると`TABLE`、一般クエリログデータとスロークエリログデータが Lightsail コンソールに表示されなくなります。ログデータを表示するには、代わりにデータベースの `mysql.general_log` および `mysql.slow_log` テーブルを参照する必要があります。

- `long_query_time` — ファストクエリがスロークエリログに記録されないようにするために、ログに記録されるクエリの最短実行時間の値を秒単位で指定します。デフォルトは 10 秒であり、最小値は 0 です。`log_output` パラメータが `FILE` に設定されている場合は、マイク

口秒の精度になるように、浮動小数点値を指定できます。log_output パラメータが TABLE に設定されている場合は、秒の精度になるように、整数値を指定する必要があります。実行時間が long_query_time パラメータの値を超えたクエリのみが口グに記録されます。例えば、long_query_time を 0.1 に設定すると、実行時間が 100 ミリ秒未満のすべてのクエリは口グに記録されなくなります。

- log_queries_not_using_indexes — インデックスを使用しないすべてのクエリをスロークエリログに記録するには、1 に設定します。デフォルトは 0 です。インデックスを使用しないクエリは、その実行時間が long_query_time パラメーターの値未満であっても口グに記録されます。

ログデータの保持

ログ記録が有効になっている場合、テーブルログのローテーションまたは口グファイルの削除が定期的に実行されます。これは、口グファイルが大きくなることでデータベースが使用できなくなったりパフォーマンスに影響する可能性を低く抑えるための予防措置です。log_output パラメータが FILE または TABLE に設定されている場合、口グ記録は次のように処理されます。

- FILE 口グ記録が有効になっている場合、口グファイルの検査が 1 時間ごとに実行され、作成後 24 時間を超えた古い口グファイルは削除されます。場合によっては、削除後の残りの口グファイルの合計サイズが、データベースに割り当てられた領域のしきい値である 2% を超えることがあります。この場合、口グファイルのサイズがしきい値以下になるまで、最も大きい口グファイルから順に削除されます。
- TABLE 口グ記録を有効化すると、24 時間ごとに口グテーブルのローテーションが実行される場合があります。

  このログテーブルのローテーションは、テーブルログに使用されている領域が、割り当てられたストレージ領域の 20 % を超えるか、すべての口グの合計サイズが 10 GB を超えると、実行されます。

  データベースに使用されている領域が、データベースに割り当てられたストレージ領域の 90% を超えている場合は、口グのローテーションを実行するためのしきい値が小さくなります。

  テーブルログに使用されている領域が、割り当てられたストレージ領域の 10% を超えるか、すべての口グの合計サイズが 5 GB を超えると、口グテーブルのローテーションが実行されます。

  low_free_storage にサブスクライブして、口グテーブルのローテーションが実行されて領域が解放されたときに通知を受け取ることができます。

- ログテーブルのローテーションが実行されると、現在のログテーブルがバックアップのログテーブルにコピーされ、現在のログテーブル内にあるエントリは削除されます。バックアップのログテーブルが既に存在する場合は、現在のログテーブルをバックアップにコピーする前に、削除されます。バックアップのログテーブルは、照会することができます。mysql.general_log テーブルに対するバックアップのログテーブルは、mysql.general_log_backup という名前になります。mysql.slow_log テーブルに対するバックアップのログテーブルは、mysql.slow_log_backup という名前になります。
- mysql.general_log テーブルのローテーションは、mysql.rds_rotate_general_logprocedure を呼び出すことで実行できます。mysql.slow_log テーブルのローテーションは、mysql.rds_rotate_slow_logprocedure を呼び出すことで実行できます。
- データベースバージョンのアップグレード時にも、テーブルログのローテーションが実行されます。

Lightsail データベースのスナップショットを作成する

Amazon Lightsail でマネージドデータベースのスナップショットを作成できます。スナップショットは、問題が発生した場合にデータベースの復元に使用できるデータベースのコピーです。また、スナップショットを使用して、高可用性プランまたはスタンダードプランなどの別のプラン を使用する新しいデータベースを作成することもできます。

スタンダードデータベースのスナップショットを作成する場合、データベースのサイズに応じて、数秒から数分、データベースが使用不可になります。高可用性データベースの場合、スナップショットはスタンバイデータベースを使用して作成されるため、スナップショットオペレーションによる影響はありません。

データベースのスナップショットを作成するには

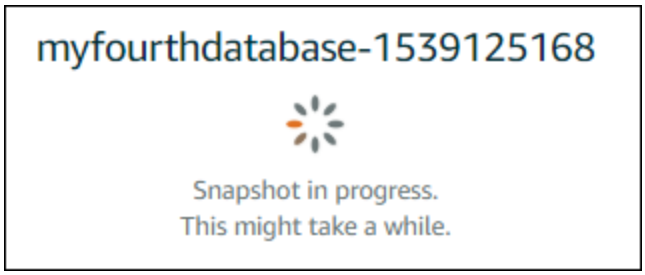
1. Lightsail コンソールにサインインします。
2. Lightsail のホームページで [データベース] タブを選択します。
3. スナップショットを作成するデータベースの名前を選択します。
4. [スナップショットと復元] タブを選択します。
5. このページの [手動スナップショット] セクションで、[スナップショットの作成] を選択し、スナップショットの名前を入力します。

   リソース名:

- Lightsail アカウントの各 AWS リージョン 内で一意である必要があります。
- 2〜255 文字を使用する必要があります。
- 先頭と末尾は英数字または数字を使用する必要があります。
- 英数字、数字、ピリオド、ダッシュ、アンダースコアを使用することができます。

6. [Create(作成)] を選択します。

スナップショットの作成プロセスが開始され、ステータスとして [スナップショットを作成中です] と表示されます。



スナップショットの作成プロセスが完了すると、新しいスナップショットが [最近のスナップショット] セクションに表示されます。アカウントのすべてのスナップショットは、Lightsail のホームページの [スナップショット] タブでも確認できます。

次のステップ

スナップショットの準備が完了したら、スナップショットから新しいデータベース (元のデータベースの複製) を作成できます。詳細については、「スナップショットからデータベースを作成する」を参照してください。

トピック

- Amazon Lightsail で特定の時点のバックアップからデータベースを作成する
- Lightsail でスナップショットからデータベースを作成する

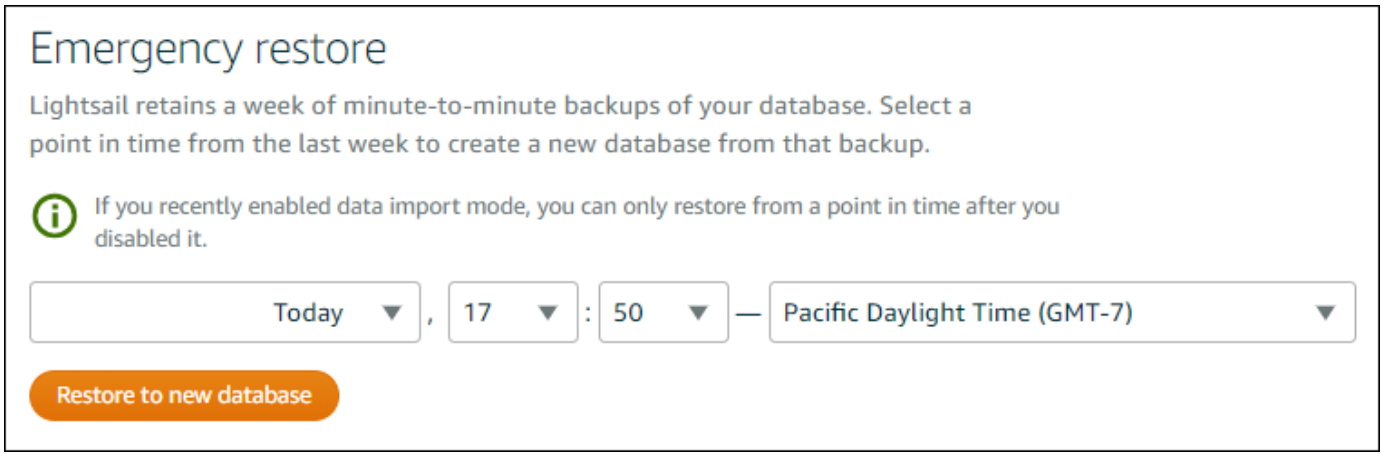
Amazon Lightsail で特定の時点のバックアップからデータベースを作成する

Amazon Lightsail で特定の時点のバックアップを使用して、新しいマネージドデータベースを作成できます。特定の時点のデータベースのバックアップは、5 分間単位で、過去を遡って 7 日分、利用できます。これにより、障害が発生したデータベースを過去 1 週間前までの特定の時点に復旧できます。

スナップショットから新しいデータベースを作成することもできます。詳細については、Amazon Lightsail でスナップショットからデータベースを作成するを参照してください。

特定の時点のバックアップからデータベースを作成するには

1. Lightsail コンソールにサインインします。
2. Lightsail のホームページで [データベース] タブを選択します。
3. プランを変更するデータベースの名前を選択します。
4. [Snapshots and restore (スナップショットおよび復元)] タブを選択します。
5. [Emergency restore (緊急復元)] セクションで、新しいデータベースに使用するバックアップの日時を選択します。



6. [Restore to new database (新しいデータベースに復元)] を選択します。
7. [新しいデータベースを作成] ページで、[ゾーンの変更] を選択して別のアベイラビリティーゾーンを選択します。前に選択したスナップショットと同じ AWS リージョンに新しいデータベースが作成されます。
8. 新しいデータベースプランを選択します。

   高可用性データベースプランまたはスタンダードデータベースプランを選択します。高可用性プランでデータベースを作成すると、プライマリデータベースのほかに、フェイルオーバーのサ

ポートとしてスタンバイ用のセカンダリデータベースが別のアベイラビリティーゾーンに作成されます。詳細については、「高可用性データベース」を参照してください。

> **Note**
>
> 元のデータベースプランより小さいデータベースプランを選択することはできません。

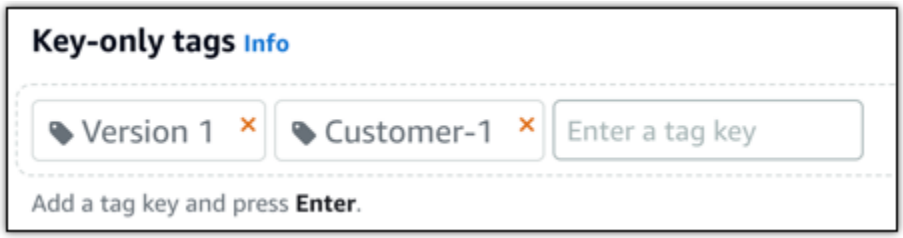
9. データベースの名前を入力します。

   リソース名:

   - Lightsail アカウントの各 AWS リージョン 内で一意である必要があります。
   - 2〜255 文字を使用する必要があります。
   - 先頭と末尾は英数字または数字を使用する必要があります。
   - 英数字、数字、ピリオド、ダッシュ、アンダースコアを使用することができます。

10. 以下のいずれかのオプションを選択して、データベースにタグを追加します。

    - [key-only タグの追加] または [key-only タグの編集] (タグが追加済みの場合)。タグキーのテキストボックスに新しいタグを入力し、Enter キーを押します。タグの入力を完了したら、[保存] を選択してタグを追加し、追加しない場合は、[キャンセル] を選択します。

    - [key-value タグの作成] から [キー] テキストボックスにキーを入力し、[値] テキストボックスに値を入力します。タグの入力を完了したら、[保存] を選択し、追加しない場合は、[キャンセル] を選択します。

      キーバリューのタグは、保存する際に一つずつ追加することができます。さらに key-value タグを追加するには、以上のステップを繰り返します。

> **Note**
>
> 「キーのみ」のタグと「キーバリュー」のタグの詳細については、「タグ」を参照してください。

11. [データベースの作成] を選択します。

    数分以内に、新しい Lightsail のデータベースが新しいデータベースプランまたはバンドルとして使用可能になります。

次のステップ

新しいデータベースが使用可能になったら、次のアクションを実行します。

- 元のデータベースが不要な場合は、削除できます。詳細については、「データベースを削除する」を参照してください。
- 特定の時点のバックアップから作成したデータベースは、Lightsail で作成された強力なパスワードを使用するように設定します。詳細については、「データベースのパスワードを管理する」を参照してください。

Lightsail でスナップショットからデータベースを作成する

元のデータベースで障害が発生した場合は、Amazon Lightsail でスナップショットから新しいマネージドデータベースを作成できます。また、データベースを別のプラン (高可用性プランまたはスタンダードプラン) に変更することもできます。さらに、元のデータベースの特定時点のバックアップから新しいデータベースを作成することもできます。詳細については、「Amazon Lightsail で特定の時点のバックアップからデータベースを作成する」を参照してください。

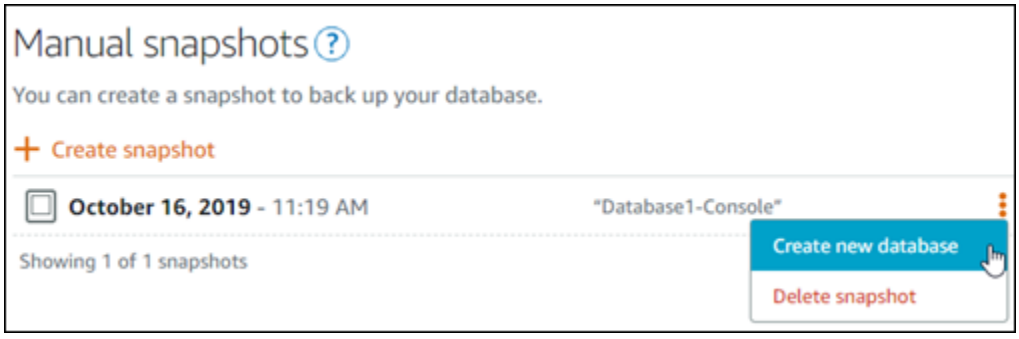
データベースを複製する際は、元のデータベースとは異なるプランやよりサイズの大きなプランを選択できます。ただし、元のデータベースよりサイズの小さいプランを選択することはできません。

> **Note**
>
> 高可用性プランでデータベースを作成すると、プライマリデータベースのほかに、フェイルオーバーのサポートとしてスタンバイ用のセカンダリデータベースが別のアベイラビリティーゾーンに作成されます。詳細については、「高可用性データベース」を参照してください。

スナップショットからデータベースを作成するには

1. Lightsail コンソールにサインインします。
2. Lightsail のホームページで [データベース] タブを選択します。
3. スナップショットから新しいデータベースを作成することによって複製する元のデータベースの名前を選択します。
4. [スナップショットと復元] タブを選択します。
5. このページの [手動スナップショット] セクションで、新しいデータベースを作成するスナップショットの横にあるアクションメニューアイコン (:) を選択してから、[Create new database (新しいデータベースの作成)] を選択します。

> **Note**
>
> データベースの既存のスナップショットが必要となります。スナップショットをまだ作成していない場合は、「データベースのスナップショットを作成する」を参照してください。



6. [Create new database (新しいデータベースの作成)] を選択します。
7. [新しいデータベースを作成] ページで、[ゾーンの変更] を選択して別のアベイラビリティーゾーンを選択します。前に選択したスナップショットと同じ AWS リージョンに新しいデータベースが作成されます。
8. 新しいデータベースプランを選択します。

   高可用性データベースプランまたはスタンダードデータベースプランを選択します。高可用性プランでデータベースを作成すると、プライマリデータベースのほかに、フェイルオーバーのサポートとしてスタンバイ用のセカンダリデータベースが別のアベイラビリティーゾーンに作成されます。詳細については、「高可用性データベース」を参照してください。

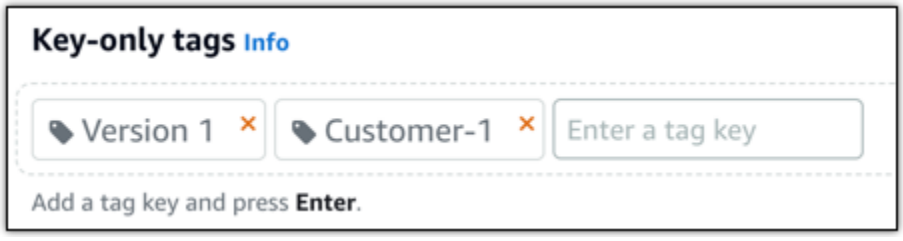
   > **Note**
   >
   > スナップショットの作成に使用した元のデータベースのプランよりサイズの小さいデータベースプランを選択することはできません。

9. データベースの名前を入力します。

   リソース名:

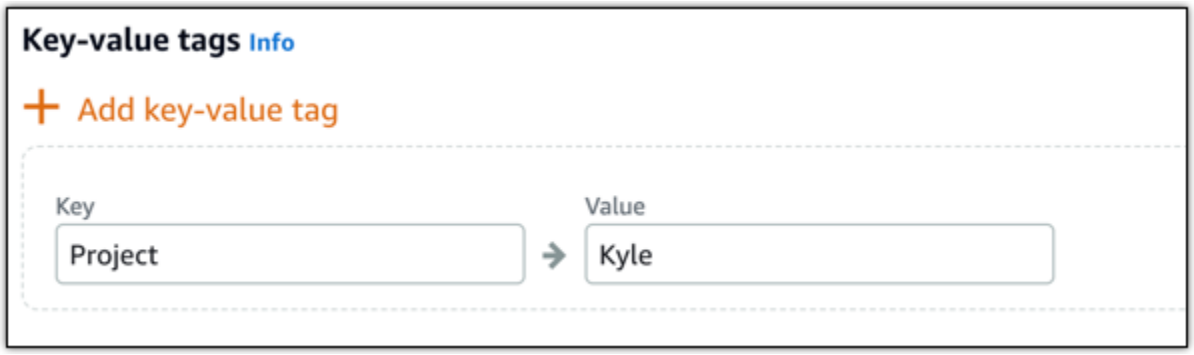
   - Lightsail アカウントの各 AWS リージョン 内で一意である必要があります。
   - 2〜255 文字を使用する必要があります。
   - 先頭と末尾は英数字または数字を使用する必要があります。
   - 英数字、数字、ピリオド、ダッシュ、アンダースコアを使用することができます。
10. 以下のいずれかのオプションを選択して、データベースにタグを追加します。

    - [key-only タグの追加] または [key-only タグの編集] (タグが追加済みの場合)。タグキーのテキストボックスに新しいタグを入力し、Enter キーを押します。タグの入力を完了したら、[保存] を選択してタグを追加し、追加しない場合は、[キャンセル] を選択します。



- [key-value タグの作成] から [キー] テキストボックスにキーを入力し、[値] テキストボックスに値を入力します。タグの入力を完了したら、[保存] を選択し、追加しない場合は、[キャンセル] を選択します。

  キーバリューのタグは、保存する際に一つずつ追加することができます。さらに key-value タグを追加するには、以上のステップを繰り返します。



> **Note**
>
> 「キーのみ」のタグと「キーバリュー」のタグの詳細については、「タグ」を参照してください。

11. [データベースの作成] を選択します。

    数分以内に、新しい Lightsail のデータベースが新しいデータベースプランまたはバンドルとして使用可能になります。

次のステップ

新しいデータベースが使用可能になったら、次のアクションを実行します。

- 新しいデータベースを作成して既存のデータベースを置き換える場合、既存のデータベースに依存するアプリケーションがあるときは、アプリケーションの依存関係を新しいデータベースに必ず更新します。
- 元のデータベースが不要な場合は、削除できます。詳細については、「データベースを削除する」を参照してください。
- スナップショットから作成したデータベースは、Lightsail で作成された強力なパスワードを使用するように設定します。詳細については、「データベースのパスワードを管理する」を参照してください。

Lightsail マネージド型データベースの SSL 証明書をダウンロードする

アプリケーションの Secure Socket Layer (SSL) または Transport Layer Security (TLS) を使用して、MySQL または PostgreSQL を実行している Amazon Lightsail のマネージド型データベースへの接続を暗号化できます。各 DB エンジンには SSL/TLS を実装する独自のプロセスがあります。詳細については、「SSL を使用した MySQL データベースの接続」または「SSL を使用した PostgreSQL データベースの接続」を参照してください。

> **Note**
>
> ダウンロード可能な証明書には Amazon Relational Database Service (Amazon RDS) のラベルが付いていますが、Lightsail のマネージド型データベースにも使用できます。

すべての AWS リージョン の証明書バンドル

すべての AWS リージョン の中間証明書とルート証明書の両方を含む証明書バンドルを取得する場合、またはアプリケーションが Microsoft Windows にあり、PKCS7 ファイルを必要とする場合、「Amazon Relational Database Service ユーザーガイド」の「すべての AWS リージョン の証明書バンドル」を参照してください。

このルート証明書は信頼されたルートエンティティであり、ほとんどの場合は使用することができます。ただし、アプリケーションが証明書チェーンを受け入れていない場合は使用できない場合があります。アプリケーションが証明書チェーンを受け入れていない場合、このドキュメントの次のセクションに進みます。

特定の AWS リージョン の証明書バンドル

特定の AWS リージョン の中間証明書とルート証明書の両方を含む証明書バンドルを取得するには、「Amazon Relational Database Service ユーザーガイド」の「特定の AWS リージョン の証明書バンドル」を参照してください。

Lightsail データベースの CA 証明書バージョンを更新する

Amazon Lightsail は、SSL/TLS を使用してマネージドデータベースに接続するための新しい認証局 (CA) 証明書を公開しました。このガイドでは、新しい CA 証明書にアップグレードする方法について説明します。証明書は、 update-relational-database API アクションを使用してのみアップグレー

ドできます。新しい証明書は、rds-ca-rsa2048-g1、、rds-ca-rsa4096-g1および と呼ばれますrds-ca-ecc384-g1。古い証明書は と呼ばれますrds-ca-2019。 AWS セキュリティのベストプラクティスとして CA 証明書を提供しています。マネージドデータベースの CA 証明書とサポートされている の詳細については、「マネージドデータベースの SSL 証明書のダウンロード AWS リージョン」を参照してください。

古い CA 証明書 (rds-ca-2019) は 2024 年 8 月 22 日に有効期限が切れます。したがって、このガイドの手順をできる限り早く完了して、新しい証明書を使用するようにマネージド型データベースを変更することを強くお勧めします。アプリケーションが SSL/TLS を使用して Lightsail マネージドデータベースに接続しない場合、アクションは必要ありません。これらのステップが完了しない場合、アプリケーションは 2024 年 8 月 22 日以降、SSL/TLS を使用してマネージドデータベースに接続できません。

2024 年 1 月 26 日以降に作成された新しいマネージドデータベースは、デフォルトでrds-ca-rsa2048-g1証明書を使用します。古い証明書 (rds-ca-2019) を使用するように新しいマネージドデータベースを一時的に変更する場合は、 AWS Command Line Interface () を使用できますAWS CLI。2024 年 1 月 26 日より前に作成されたマネージドデータベースは、、rds-ca-rsa2048-g1、rds-ca-rsa4096-g1および rds-ca-2019証明書に更新されるまでrds-ca-ecc384-g1証明書を使用します。

> **Note**
>
> このガイドの手順は、本番稼働用環境で使用する前に、開発環境またはステージング環境でテストしてください。

前提条件

- このガイドでは、 AWS CloudShell を使用してアップグレードを実行します。 CloudShell は、Lightsail コンソールから直接起動できるブラウザベースの事前認証済みシェルです。では CloudShell、Bash、 PowerShellZ シェルなどの任意のシェルを使用して AWS Command Line Interface （ AWS CLI) コマンドを実行できます。この手順は、コマンドラインツールのダウンロードもインストールも不要です。のセットアップと使用方法の詳細については CloudShell、AWS CloudShell Lightsail の「」を参照してください。
- 以下の手順を実行する前に、新しい SSL/TLS 証明書を使用するようにデータベースアプリケーションを更新してください。新しい SSL/TLS 証明書のアプリケーションを更新する方法は、特定のアプリケーションにより異なります。アプリケーション開発者と協力して、アプリケーションの

SSL/TLS 証明書を更新します。新しい SSL/TLS 証明書のためのアプリケーションの更新について詳しくは、「Amazon Relational Database Service ユーザーガイド」の「新しい SSL/TLS 証明書を使用して MySQL DB インスタンスに接続するためのアプリケーションの更新」または「新しい SSL/TLS 証明書を使用して PostgreSQL DB インスタンスに接続するためのアプリケーションの更新」を参照してください。

マネージドデータベースのアクティブな CA 証明書を特定する

Lightsail データベースインスタンスのアクティブな CA 証明書を特定するには、次のステップを実行します。

1. ターミナル、AWS CloudShell、またはコマンドプロンプトウィンドウを開きます。
2. 次のコマンドを入力して、マネージドデータベースのアクティブな CA 証明書を特定します。

```
aws lightsail get-relational-database --relational-database-name DatabaseName --
region DatabaseRegion | grep "caCertificateIdentifier"
```

コマンドで、 を、変更するデータベースの名前*DatabaseName*に置き換え、 を、データベースインスタンスが存在する に置き換え*DatabaseRegion* AWS リージョン ます。

例

```
aws lightsail get-relational-database --relational-database-name Database-1 --
region us-east-1 | grep "caCertificateIdentifier"
```

コマンドは、データベースのアクティブな CA 証明書の ID を返します。

例

```
"caCertificateIdentifier": "rds-ca-rsa2048-g1"
```

新しい CA 証明書を使用するためにマネージド型データベースを変更する

Lightsail でマネージドデータベースを変更して新しい CA 証明書 (`rds-ca-rsa2048-g1`、、) のいずれかを使用するには`rds-ca-rsa4096-g1`、次のステップを実行します`rds-ca-ecc384-g1`。

1. ターミナル、AWS CloudShell、またはコマンドプロンプトウィンドウを開きます。

2. 次のコマンドを入力して、マネージドデータベースで新しい証明書を使用します。

```
aws lightsail update-relational-database --relational-database-name DatabaseName --
ca-certificate-identifier rds-ca-rsa2048-g1
```

コマンドで、 を、変更するデータベースの名前*DatabaseName*に置き換えます。

例

```
aws lightsail update-relational-database --relational-database-name Database-1 --
ca-certificate-identifier rds-ca-rsa2048-g1
```

マネージドデータベースで使用される CA 証明書は、データベースの次のメンテナンスウィンドウ中に更新されます。または、 コマンドの最後に `--apply-immediately`パラメータを追加した場合はすぐに更新されます。

古い CA 証明書を使用するためにマネージド型データベースを変更する

Lightsail のマネージドデータベースを変更して古い CA 証明書 () を使用するには、以下の手順を実行しますrds-ca-2019。これは、新しい証明書 (rds-ca-rsa2048-g1、、および rds-ca-ecc384-g1) のいずれかで重大な問題が発生しrds-ca-rsa4096-g1、古い証明書を一時的に元に戻す必要がある場合にのみ実行してください。

1. ターミナル、AWS CloudShell、またはコマンドプロンプトウィンドウを開きます。
2. マネージド型データベースで rds-ca-2019 を使用するには、以下のコマンドを入力します。

```
aws lightsail update-relational-database --relational-database-name DatabaseName --
ca-certificate-identifier rds-ca-2019
```

コマンドで、 を、変更するデータベースの名前*DatabaseName*に置き換えます。

例

```
aws lightsail update-relational-database --relational-database-name Database-1 --
ca-certificate-identifier rds-ca-2019
```

マネージドデータベースで使用される CA 証明書は、データベースの次のメンテナンスウィンドウ中に更新されます。または、 コマンドの最後に `--apply-immediately`パラメータを追加した場合はすぐに更新されます。

Lightsail データベースの優先メンテナンスおよびバックアップ期間を変更する

Amazon Lightsail で新バージョンのデータベースのサポートを開始した場合、既存のマネージドデータベースを新バージョンにアップグレードできます。アップグレードには、メジャーバージョンのアップグレードとマイナーバージョンのアップグレードの 2 種類があります。現在、Lightsail はマイナーバージョンのアップグレードのみサポートしています。

マイナーバージョンアップグレードおよび他のデータベースメンテナンスタスクは、データベースのメンテナンスウィンドウ中に自動的に実行されます。優先メンテナンス期間は 30 分のウィンドウで、各 AWS リージョン ごとに 8 時間の時間ブロックからランダムに選択されます。このウィンドウはランダムな曜日に発生します。データベースのバックアップは、バックアップウィンドウ中に実行されます。優先バックアップ期間は 30 分の期間で、各 AWS リージョン ごとに 8 時間の時間ブロックからランダムに選択されます。このウィンドウもランダムな曜日に発生します。

> **Note**
>
> リージョン別のデフォルトメンテナンスウィンドウの時間ブロックの詳細については、Amazon Relational Database Service (Amazon RDS) ドキュメントの「DB インスタンスのメンテナンス」ガイドを参照してください。リージョン別のデフォルトバックアップウィンドウの時間ブロックの詳細については、Amazon RDS ドキュメントの「バックアップの使用」ガイドを参照してください。

このガイドでは、メンテナンスおよびバックアップウィンドウを、データベースの負荷が最も低い時間帯に変更する方法を示します。

前提条件

データベースのメンテナンスおよびバックアップウィンドウを変更するには、AWS Command Line Interface (AWS CLI) を使用する必要があります。

以下の前提条件を満たしてください。

- AWS CLI のインストール — 詳細については、「AWS CLII のインストール」を参照してください。
- AWS CLI を設定する — 詳細については、「AWS CLI の設定」を参照してください。

データベースのメンテナンスウィンドウを変更する

データベースは、メンテナンスまたはバックアップオペレーション中は利用できない場合があります。したがって、メンテナンスまたはバックアップウィンドウを、データベースの負荷が最も低い時間に変更する必要が生じる場合があります。

データベースのメンテナンスウィンドウを変更するには

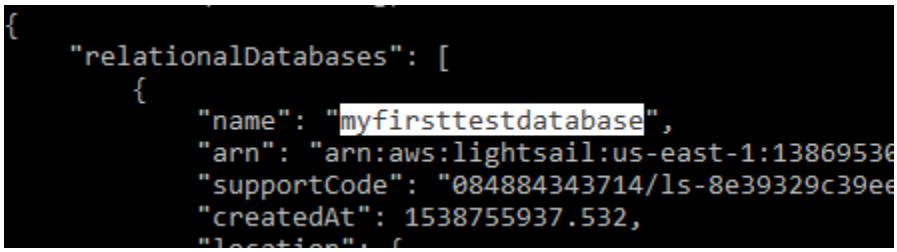
1. ターミナルまたはコマンドプロントウィンドウを開きます。
2. 次のコマンドを入力して、メンテナンスウィンドウを変更するデータベースの名前を取得します。

   ```
   aws lightsail get-relational-databases
   ```

   以下の例のような結果が表示されるはずです。

> **Note**
>
> 変更するデータベースが表示されない場合は、AWS CLI に設定されている AWS リージョン にデータベースが存在することを確認します。詳細については、「AWS CLI の設定」を参照してください。

3. 変更するデータベースの名前を強調表示し、Ctrl+C (Windows) または Cmd+C (macOS) を押して、クリップボードにコピーします。これを次のステップで使用します。



4. 変更するウィンドウに応じて、以下のいずれかのコマンドを入力します。

   - 次のコマンドを入力し、データベースのメンテナンスウィンドウを変更します。

```
aws lightsail update-relational-database --relational-database-name DatabaseName
 --preferred-maintenance-window MaintenanceWindow
```

コマンドを、以下のように置き換えます。

- *DatabaseName* は、データベースの名前に置き換えます。
- *MaintenanceWindow* は、新しいメンテナンスウィンドウの時間枠に置き換えます。

  メンテナンスウィンドウの時間を ddd:hh24:mi-ddd:hh24:mi 形式で定義します。また、協定世界時 (UTC) 形式で指定し、最低 30 分のウィンドウとして定義する必要があります。メンテナンスウィンドウは、バックアップウィンドウと重複できません。

例:

```
aws lightsail update-relational-database --relational-database-
name myproductiondb --preferred-maintenance-window Tue:16:00-Tue:16:30
```

- 次のコマンドを入力し、データベースのバックアップウィンドウを変更します。

```
aws lightsail update-relational-database --relational-database-name DatabaseName
 --preferred-backup-window BackupWindow
```

コマンドを、以下のように置き換えます。

- *DatabaseName* は、データベースの名前に置き換えます。
- *BackupWindow* は、新しいバックアップウィンドウの時間枠に置き換えます。

  バックアップウィンドウの時間を hh24:mi-hh24:mi 形式で定義します。また、協定世界時 (UTC) 形式で指定し、最低 30 分のウィンドウとして定義する必要があります。バックアップウィンドウは、メンテナンスウィンドウと重複できません。

例:

```
aws lightsail update-relational-database --relational-database-
name myproductiondb --preferred-backup-window 14:00-14:30
```

以下の例のような結果が表示されるはずです。

次のステップ

データベースの管理に役立つ以下のガイドを参照してください。

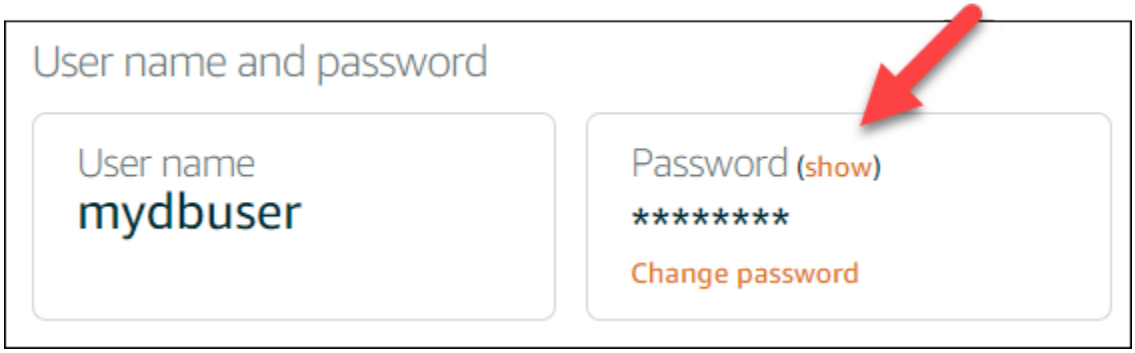
- データベースのデータのインポートモードを設定する
- データベースのパブリックモードを設定する
- データベースのパスワードを管理する
- MySQL データベースに接続する
- PostgreSQL データベースに接続する
- MySQL データベースにデータをインポートする
- データを PostgreSQL データベースにインポートする
- データベースのスナップショットを作成する

Lightsail データベースのパスワードを管理する

Amazon Lightsail で新しいデータベースを作成する場合、Lightsail で自動的に強力なパスワードを作成するか、独自にパスワードを指定できます。現在のデータベースのパスワードは、Lightsail コンソールでいつでも表示または変更できます。

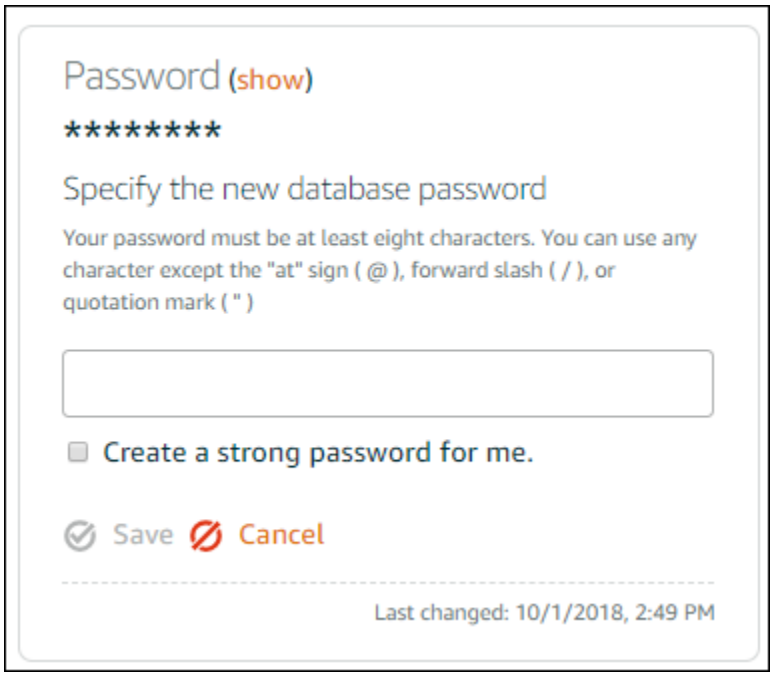
データベースのパスワードを管理するには

1. Lightsail コンソールにサインインします。
2. Lightsail のホームページで [データベース] タブを選択します。
3. パスワードを管理するデータベースの名前を選択します。
4. [接続] タブの [User name and passwords (ユーザー名とパスワード)] セクションで、[表示] を選択して現在のデータベースパスワードを表示します。



5. データベースのパスワードを変更するには、[パスワードの変更] を選択します。

   Lightsail で自動的に強力なパスワードを作成するか、独自のパスワードをテキストボックスに入力するかを選択できます。パスワードには「/」「"」または「@」を除く表示可能な任意の ASCII 文字を使用することができます。MySQL データベースの場合、パスワードには 8 ~ 41 文字の英数字を使用する必要があります。PostgreSQL の場合、パスワードには 8 ~ 128 文字の英数字を使用する必要があります。



6. 完了したら、[保存] を選択します。

   データベースのパスワードの変更はすぐに適用されます。独自のパスワードを入力した場合、パスワードはすぐに保存されます。Lightsail で自動的にパスワードを作成した場合は、数秒以内に生成されます。新しいパスワードを表示するには、[表示] を選択します。

次のステップ

Lightsail でのデータベースの管理に役立つ以下のガイドを参照してください。

- MySQL データベースに接続する
- PostgreSQL データベースに接続する
- データベースのスナップショットを作成する

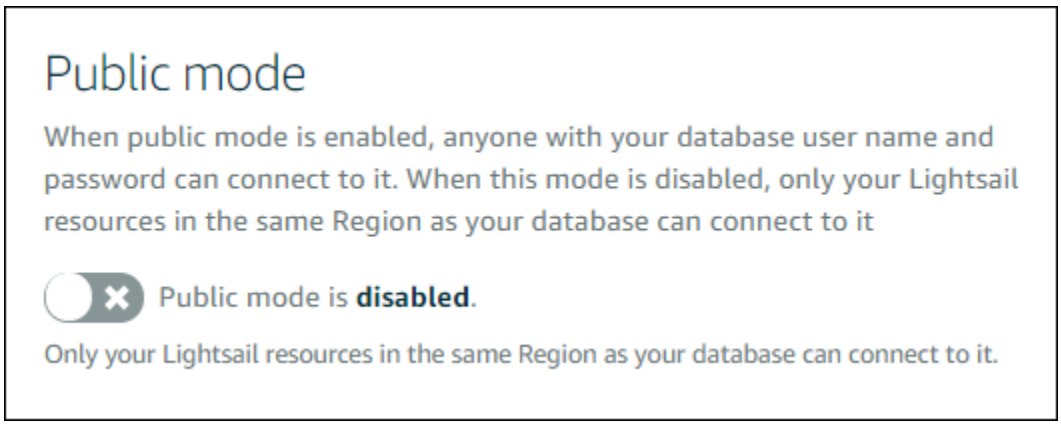
Lightsail データベースのパブリックモードを設定する

Amazon Lightsail のマネージドデータベースにアクセスできるのは、同じ Lightsail アカウント内にある Lightsail リソース (インスタンス、ロードバランサーなど) のみです。1 つの一般的なシナリオでは、一般向けのウェブアプリケーションを持つ Lightsail インスタンスと、パブリックアクセス可能ではない Lightsail データベースの両方を作成し、この 2 つを接続します。

データベースをパブリックアクセス可能にするには、パブリックモード機能を有効にします。これにより、すべてのユーザーがデータベースエンドポイント、ポート、ユーザー名、およびパスワードを使用してデータベースに接続できます。詳細については、「MySQL データベースに接続する」または「PostgreSQL データベースに接続する」を参照してください。

データベースのパブリックモードを設定するには

1. Lightsail コンソールにサインインします。
2. Lightsail のホームページで [データベース] タブを選択します。
3. パブリックモードを設定するデータベースの名前を選択します。
4. [ネットワーキング] タブを選択します。
5. [Public mode (パブリックモード)] セクションで、トグルを使用してパブリックモードをオンにします。これをオフにする場合も、トグルを使用します。



パブリックアクセシビリティの設定の適用が即座に開始されますが、完了するまでに数分かかることがあります。この間に、データベースのステータスは [変更中] に変わります。パブリックアクセシビリティの設定が適用されると、データベースのステータスは [利用可能] に変わります。

次のステップ

データベースの管理に役立つ以下のガイドを参照してください。

- データベースのデータのインポートモードを設定する
- データベースのパスワードを管理する
- MySQL データベースに接続する
- PostgreSQL データベースに接続する
- MySQL データベースにデータをインポートする
- データを PostgreSQL データベースにインポートする
- データベースのスナップショットを作成する

Lightsail データベースのパラメータの更新

データベースのパラメータ (データベースのシステム変数とも呼ばれます) は、Amazon Lightsail のマネージドインスタンスの基本的なプロパティを定義します。たとえば、データベース接続の数を制限するデータベースのパラメータを定義したり、データベースのバッファプールサイズを制限する別のパラメータを定義したりできます。このガイドでは、AWS Command Line Interface (AWS CLI) を使用してマネージドデータベースのパラメータリストを取得し、パラメータを更新する方法を示します。

> **Note**
>
> MySQL のシステム変数の詳細については、MySQL 5.6、MySQL 5.7、または MySQL 8.0 のドキュメントをご覧ください。PostgreSQL システム変数の詳細については、PostgreSQL 9.6、PostgreSQL 10、PostgreSQL 11、または PostgreSQL 12 のドキュメントを参照してください。

前提条件

- まだ AWS CLI をインストールして設定していない場合は、インストールして設定します。詳細については、「Lightsail で使用するために AWS CLI を設定する」を参照してください。

使用可能なデータベースのパラメータのリストを取得します。

データベースのパラメータは、データベースエンジンによって異なります。そのため、使用しているマネージドデータベースに応じたパラメータのリストを取得する必要があります。これにより、どのパラメータを変更し、どのような方法でパラメータを有効にするかを決定できます。

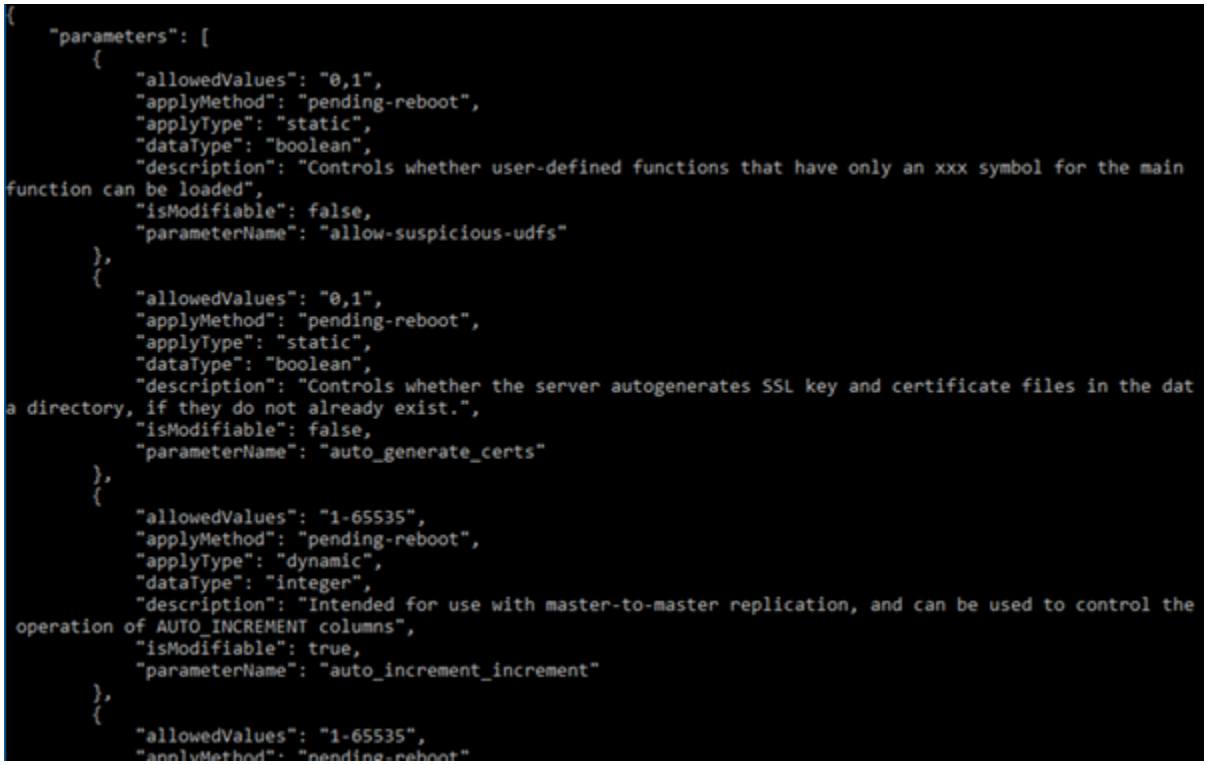
使用可能なデータベースのパラメータのリストを取得するには

1. ターミナルまたはコマンドプロントウィンドウを開きます。
2. 以下のコマンドを入力して、データベースのパラメータのリストを取得します。

```
aws lightsail get-relational-database-parameters --relational-database-
name DatabaseName
```

コマンドで、*DatabaseName* をデータベースの名前に置き換えます。

以下の例のような結果が表示されるはずです。



> **Note**
>
> パラメータの結果がページ分割される場合は、「次ページトークン ID」が表示されます。この次ページトークン ID を書き留め、表示されたとおりに次のステップで使用して、パラメータ結果の次のページを表示します。

3. 結果がページ分割されている場合は、次のコマンドを使用して追加のパラメータセットを表示します。それ以外の場合は、次のステップに進みます。

```
aws lightsail get-relational-database-parameters --relational-database-
name DatabaseName --page-token NextPageTokenID
```

コマンドを、以下のように置き換えます。

- *DatabaseName* は、データベースの名前に置き換えます。
- *NextPageTokenID* は、次のページトークン ID に置き換えます。

結果には、データベースのパラメータごとに以下の情報が表示されます。

- 使用できる値 — パラメータの有効な値の範囲を指定します。

- 適用方法 — パラメータの変更を適用するタイミングを指定します。使用できるオプションは、immediate または pending-reboot です。適用方法の定義の詳細については、次の「適用タイプ」を参照してください。
- 適用タイプ — エンジン固有の送信タイプを指定します。dynamic が表示された場合は、適用方法として immediate を使用してパラメータを適用できます。データベースは新しいパラメータ値の使用を即座に開始します。static が表示された場合は、パラメータの適用方法として pending-reboot のみ使用できます。データベースは新しいパラメータ値の使用を再起動後にのみ開始します。
- データ型 — パラメータの有効なデータ型を指定します。
- 説明 — パラメータの説明です。
- 変更可能 — パラメータが変更可能であるかどうかを示すブール値。true が表示された場合、パラメータは変更可能です。
- パラメータ名 — パラメータの名前を指定します。この値は update relational database オペレーションおよび parameter name パラメータと組み合わせて使用します。

4. 変更するパラメータを検索し、パラメータ名、使用できる値、および適用方法を書き留めます。間違えて入力しないように、パラメータ名をクリップボードにコピーすることをお勧めします。これを行うには、パラメータ名を強調表示し、Ctrl+C (Windows) または Cmd+C (macOS) を押してクリップボードにコピーします。次に、Ctrl+V または Cmd+V を押して貼り付けます。

   変更するパラメータの名前を確認したら、このガイドの次のセクションに進み、パラメータを目的の値に変更します。

データベースのパラメータを更新する

変更するパラメータの名前を確認したら、次のステップを実行して Lightsail のマネージドデータベースのパラメータを変更し ます。

データベースのパラメータを更新するには

- 次のコマンドをターミナルまたはコマンドプロンプトウィンドウに入力し、マネージドデータベースのパラメータを更新します。

```
aws lightsail update-relational-database-parameters
 --relational-database-name DatabaseName --parameters
 "parameterName=ParameterName,parameterValue=NewParameterValue,applyMethod=ApplyMethod"
```

コマンドを、以下のように置き換えます。

- *DatabaseName* は、データベースの名前に置き換えます。
- *ParameterName* は、変更するパラメータの名前に置き換えます。
- *NewParameterValue* は、パラメータの新しい値に置き換えます。
- *ApplyMethod* は、パラメータの適用方法に置き換えます。

  パラメータの適用タイプが dynamic である場合は、適用方法として immediate を使用してパラメータを適用できます。データベースは新しいパラメータ値の使用を即座に開始します。ただし、パラメータの適用タイプが static である場合は、パラメータの適用方法として pending-reboot のみ使用できます。データベースは新しいパラメータ値の使用を再起動後にのみ開始します。

以下の例のような結果が表示されるはずです。


```
{
    "operations": [
        {
            "id": "2c650987-11e8-463f-94d5-0c15aacaf12b",
            "resourceName": "myfirsttestdatabase",
            "resourceType": "RelationalDatabase",
            "createdAt": 1539204831.214,
            "location": {
                "availabilityZone": "us-east-1a",
                "regionName": "us-east-1"
            },
            "isTerminal": true,
            "operationType": "UpdateRelationalDatabaseParameters",
            "status": "Succeeded",
            "statusChangedAt": 1539204831.214
        }
    ]
}
```


データベースのパラメータは、使用される適用方法に応じて更新されます。

Lightsail データベースのメジャーバージョンのアップグレード

Amazon Lightsail がデータベースエンジンの新しいバージョンをサポートしている場合は、データベースを新しいバージョンにアップグレードできます。Lightsail には、MySQL と PostgreSQL の 2 つのデータベースブループリントがあります。このガイドでは、MySQL または PostgreSQL データ

ベースインスタンスのメジャーバージョンをアップグレードする方法について説明します。データベースのメジャーバージョンをアップグレードするには、 update-relational-database API アクションを使用します。

AWS CloudShell を使用してアップグレードを実行します。 CloudShell は、Lightsail コンソールから直接起動できるブラウザベースの事前認証済みシェルです。では CloudShell、Bash、 PowerShellZ シェルなどの任意のシェルを使用して AWS Command Line Interface （ AWS CLI) コマンドを実行できます。この手順は、コマンドラインツールのダウンロードもインストールも不要です。のセットアップおよび使用方法の詳細については CloudShell、AWS CloudShell Lightsail の「」を参照してください。

変更点を理解する

メジャーバージョンをアップグレードすると、以前のバージョンと多くの非互換性が生じる可能性があります。これらの非互換性により、アップグレード中に問題が発生する可能性があります。アップグレードを成功させるには、データベースの準備が必要になる場合があります。データベースのメジャーバージョンのアップグレードについては、MySQL および PostgreSQL ウェブサイトの以下のトピックを参照してください。

- アップグレードのためのインストールの準備
- MySQL アップグレードチェッカーユーティリティ
- PostgreSQL クラスターのアップグレード

前提条件

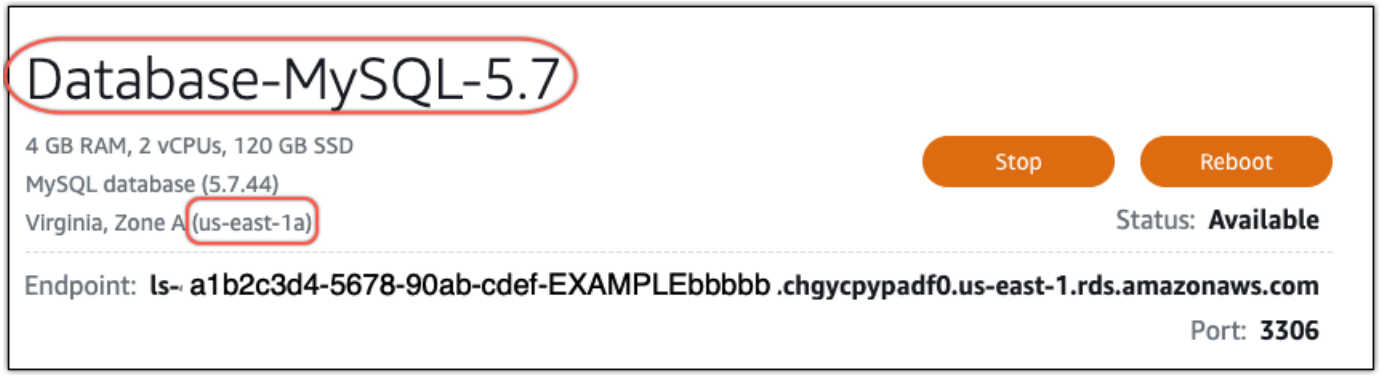
1. アプリケーションがデータベースの両方のメジャーバージョンをサポートしていることを確認します。
2. 変更を加える前に、データベースインスタンスのスナップショットを作成することをお勧めします。詳細については、「Lightsail データベースのスナップショットを作成する」を参照してください。
3. （ オプション) 先ほど作成したスナップショットから新しいデータベースインスタンスを作成します。データベースの更新にはダウンタイムが必要なため、現在アクティブなデータベースをアップグレードする前に、新しいデータベースでアップグレードをテストできます。データベースのコピー作成の詳細については、「Lightsail データベースのスナップショットを作成する」を参照してください。

データベースのメジャーバージョンを更新する

Lightsail は、MySQL および PostgreSQL データベースインスタンスのメジャーバージョンアップグレードをサポートしています。MySQL データベースは、次の手順の例として使用されます。ただし、PostgreSQL データベースの場合、プロセスとコマンドは同じです。

Lightsail データベースのデータベースメジャーバージョンをアップグレードするには、以下の手順を実行します。

1. Lightsail コンソールにサインインします。
2. 左のナビゲーションペインの [データベース] を選択します。
3. アップグレードするデータベースインスタンスの名前と AWS リージョン を書き留めます。



4. Lightsail コンソールの左下隅で、 を選択しますCloudShell。同じブラウザタブで CloudShell ターミナルが開きます。コマンドプロンプトが表示されたら、シェルは対話的な操作の準備ができています。
5. CloudShell プロンプトで次のコマンドを入力して、使用可能なデータベースブループリント IDs のリストを取得します。

```
aws lightsail get-relational-database-blueprints
```

6. アップグレード先のメジャーバージョンのブループリント ID を書き留めます。例えば `mysql_8_0` です。

7. 次のコマンドを入力して、データベースのメジャーバージョンをアップグレードします。アップグレードは、データベースの次のメンテナンスウィンドウ中に行われます。コマンドで、 をデータベースの名前*DatabaseName*に、*blueprintId* をアップグレード先のメジャーバージョンのブループリント ID *DatabaseRegion*に、 をデータベース AWS リージョン がある に置き換えます。

```
aws lightsail update-relational-database \
 --relational-database-name DatabaseName \
 --relational-database-blueprint-id blueprintId \
 --region DatabaseRegion
```

(オプション) アップグレードをすぐに適用するには、 コマンドに `--apply-immediately`パラメータを含めます。次の例のような応答が表示され、アップグレードの適用中はデータベースが使用できなくなります。詳細については、Lightsail API リファレンスupdate-relational-databaseの「」を参照してください。

```
% aws lightsail update-relational-database \
--relational-database-name "Database-Mysql-5.7" \
--relational-database-blueprint-id "mysql_8_0" \
--apply-immediately \
[--region us-east-1
{
    "operations": [
        {
            "id": "a1b2c3d4-5678-90ab-cdef-EXAMPLEbbbbb",
            "resourceName": "Database-Mysql-5.7",
            "resourceType": "RelationalDatabase",
            "createdAt": 2024-01-01T00:00:00.00000+00:00",
            "location": {
                "availabilityZone": "us-east-1a",
                "regionName": "us-east-1"
            },
            "isTerminal": true,
            "operationDetails": "",
            "operationType": "UpdateRelationalDatabase",
            "status": "Succeeded",
            "statusChangedAt": 2024-01-01T00:00:00.00000+00:00",
        }
```

8. 次のコマンドを入力して、メジャーバージョンアップグレードが次のデータベースメンテナンスウィンドウにスケジュールされていることを確認します。コマンドで、 をデータベースの名前*DatabaseName*に置き換え、 をデータベース*DatabaseRegion* AWS リージョン がある に置き換えます。

```
aws lightsail get-relational-database \
 --relational-database-name DatabaseName \
 --region DatabaseRegion
```

get-relational-database レスポンスでは、データベースは次のメンテナンスウィンドウ中に保留中のメジャーバージョンアップグレードstateを通知します。次のメンテナンスウィンドウの日時は、レスポンスの preferredMaintenanceWindowセクションで確認できます。

データベースインスタンスの状態

```
"state": "upgrading",
  "backupRetentionEnabled": true,
  "pendingModifiedValues": {
  "engineVersion": "8.0.36"
```

メンテナンスウィンドウ

```
"preferredMaintenanceWindow": "wed: 09:22-wed: 09:52"
```

次のステップ

テストデータベースを作成した場合は、アップグレードされたデータベースでアプリケーションが動作することを確認した後に削除できます。前のデータベースに復元する必要がある場合に備えて、前のデータベースで作成したスナップショットを保持します。また、アップグレードしたデータベースのスナップショットを作成して、新しい point-in-time コピーを作成する必要があります。

Amazon Lightsail のロードバランサー

Lightsail ロードバランサーは、着信ウェブトラフィックを複数のアベイラビリティーゾーンの複数の Lightsail インスタンス間で分散します。ロードバランシングを使用すると、インスタンスでのアプリケーションの可用性と耐障害性が向上します。アプリケーションへのリクエストの流れを中断することなく、ニーズの変化に応じて Lightsail ロードバランサーに対して インスタンスの追加と削除を行うことができます。

Lightsail のロードバランシングを使用して、DNS ホスト名を作成し、このホスト名に送信されるすべてのリクエストを、ターゲットの Lightsail インスタンスのプールにルーティングします。インスタンスの合計数に関する Lightsail アカウントのクォータを超えない限り、任意の数のターゲットインスタンスをロードバランサーに追加できます。

ロードバランサーの機能

Lightsail ロードバランサーは、次の機能を備えています。

- HTTPS 暗号化 — デフォルトでは、Lightsail ロードバランサーは、ポート 80 を介して暗号化されていない (HTTP) トラフィックリクエストを処理します。検証済みの Lightsail SSL/TLS 証明書をロードバランサーにアタッチして HTTPS 暗号化をアクティブ化します。これにより、ロードバランサーは、ポート 443 を介して暗号化された (HTTPS) トラフィックリクエストを処理することができます。詳細については、「SSL/TLS 証明書」を参照してください。

  ロードバランサーで HTTPS 暗号化をアクティブ化すると、次の機能を使用できます。

  - HTTP から HTTPS へのリダイレクト — HTTP から HTTPS へのリダイレクトをアクティブ化して、HTTP リクエストを HTTPS 暗号化接続に自動的にリダイレクトします。詳細については、「ロードバランサーの HTTP から HTTPS へのリダイレクトの設定」を参照してください。
  - TLS セキュリティポリシー — ロードバランサーで TLS セキュリティポリシーを設定します。詳細については、「Amazon Lightsail ロードバランサーでの TLS セキュリティポリシーの設定」を参照してください。
- ヘルスチェック — デフォルトでは、ヘルスチェックは、アタッチされたインスタンスにおいて、それらのインスタンスで実行されているウェブアプリケーションのルートで実行されます。ヘルスチェックは、ロードバランサーから正常なインスタンスにのみリクエストを送信できるように、インスタンスのヘルス状態をモニタリングします。詳細については、「Lightsail ロードバランサーのヘルスチェック」を参照してください。

- セッション永続性 — ウェブサイトの訪問者のブラウザでセッション情報をローカルに保存する場合は、セッション永続性を設定します。例えば、ロードバランシング済みの Lightsail インスタンスで、ショッピングカート機能を備えた Magento e コマースアプリケーションを実行しているとします。ウェブサイトの訪問者がショッピングカートに商品を追加してセッションを終了した後に戻ってくると、セッション永続性を設定した場合はショッピングカートの商品が残っています。詳細については、「ロードバランサーのセッション永続性を有効にする」を参照してください。

ロードバランサーを使用するタイミング

ロードバランサーは、トラフィックがときどき急上昇するウェブサイトがある場合や、多くのユーザーが一度に利用したときにインスタンスで大きな負荷が発生するコストコンテンツがある場合に使用してください。たとえば、画像が多いウェブサイトがある場合、他のページリクエストを使ってイメージリクエストをロードバランシングすることができます。このようにすると、ページのロード時間が短縮され、ユーザーの満足度が向上します。

ロードバランサーを使用すると、可用性の高いウェブサイトを作成できます。高可用性とは、一定期間内にウェブサイトやアプリケーションが稼働している時間の長さを指します。これまでサイトが停止したことがある場合、ロードバランサーはアップタイムの増加に役立つ場合があります。Lightsail ロードバランサーを使用すると、複数のアベイラビリティーゾーンに分散されるターゲットインスタンスを追加することで、アプリケーションの可用性を高めることができます。

耐障害性は、関連する概念です。いずれかのインスタンスまたはデータベースで障害が発生した後もサイトが動作し続ける場合、耐障害性があると見なされます。ロードバランサーは、耐障害性を備えたアプリケーションまたはウェブサイトを作成するのに役立ちます。

ロードバランシングが推奨される アプリケーション

すべての Lightsail アプリケーションにロードバランサーが必要なわけではありません。ロードバランシングされたアプリケーションを作成することに決定した場合、最初にアプリケーションを設定する必要があります。例えば、ロードバランシング用の LAMP スタックアプリケーションを準備するには、最初にすべてのターゲットインスタンスが読み取り/書き込みを行うための一元的な専用データベースを作成する必要があります。Lightsail オブジェクトストレージバケットなどの一元化されたメディアストレージの作成を検討することも考えられます。詳細については、「ロードバランシング用のインスタンスを設定する」を参照してください。

ロードバランサーの使用を開始する

ロードバランサーを作成するには、Lightsail コンソール、AWS Command Line Interface (AWS CLI)、または Lightsail API を使用します。ロードバランシング用のインスタンスも設定する必要があります。

ロードバランサーを作成して設定したインスタンスをアタッチすると、次のトピックを使用して HTTPS を有効にすることができます。詳細については、「ロードバランサー用の SSL/TLS 証明書を作成する」を参照してください。

Lightsail ロードバランサーを作成してインスタンスをアタッチする

アプリケーションの冗長性を高め、より多くのウェブトラフィックを処理するには、 ロードバランサーを作成します。ロードバランサーが作成されたら、負荷分散する Lightsail インスタンスをアタッチできます。詳細については、「ロードバランサー」を参照してください。

前提条件

開始する前に、ロードバランシング用の Lightsail インスタンスを準備したことを確認します。詳細については、「ロードバランシング用の インスタンスを設定する」を参照してください。

ロードバランサーの作成

1. Lightsail コンソールにサインインします。
2. [ネットワーキング] タブを選択します。
3. [ロードバランサーの作成] を選択します。
4. ロードバランサーが作成される AWS リージョン を確認するか、[リージョンの変更] を選択して、別のリージョンを選択します。

   > **Note**
   >
   > デフォルトでは、HTTP リクエストを受け入れるため、ポート 80 が開いたロードバランサーが作成されます。ロードバランサーを作成した後は、SSL/TLS 証明書を作成して HTTPS を設定できます。詳細については、「ロードバランサー用の SSL/TLS 証明書を作成する」を参照してください

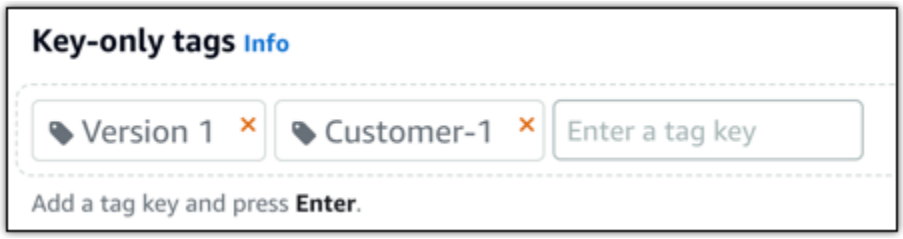
5. ロードバランサーの名前を入力します。

リソース名:

- Lightsail アカウントの各 AWS リージョン 内で一意である必要があります。
- 2〜255 文字を使用する必要があります。
- 先頭と末尾は英数字または数字を使用する必要があります。
- 英数字、数字、ピリオド、ダッシュ、アンダースコアを使用することができます。

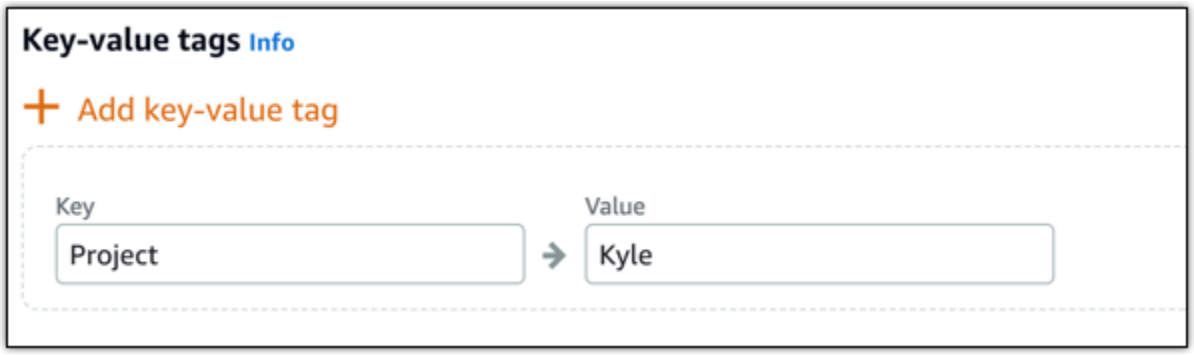
6. 以下のいずれかのオプションを選択して、ロードバランサーにタグを追加します。

- [key-only タグの追加] または [key-only タグの編集] (タグが追加済みの場合)。タグキーのテキストボックスに新しいタグを入力し、Enter キーを押します。タグの入力を完了したら、[保存] を選択してタグを追加し、追加しない場合は、[キャンセル] を選択します。



- [key-value タグの作成] から [キー] テキストボックスにキーを入力し、[値] テキストボックスに値を入力します。タグの入力を完了したら、[保存] を選択し、追加しない場合は、[キャンセル] を選択します。

  キーバリューのタグは、保存する際に一つずつ追加することができます。さらに key-value タグを追加するには、以上のステップを繰り返します。



> **Note**
>
> 「キーのみ」のタグと「キーバリュー」のタグの詳細については、「タグ」を参照してください。

7. [ロードバランサーの作成] を選択します。

インスタンスをロードバランサーにアタッチする

ロードバランサーを作成した後、Lightsail のロードバランサー管理ページに移動します。そのページを再表示する必要がある場合は、Lightsail ホーム画面の[ネットワーク]タブを開き、Lightsailロードバランサーの名前を選択し、管理します。

> **Note**
>
> Lightsail インスタンスは、ロードバランサーにアタッチする前に実行されている必要があります。

1. ロードバランサー管理ページで、[ターゲットインスタンス] を選択します。
2. [ターゲットインスタンス] ドロップダウンリストでインスタンスを選択します。
3. [Attach] (添付) を選択します。アタッチには数分かかる場合があります。

   [Attach another] を選択し、前述のステップを繰り返して、別のインスタンスをロードバランサーにアタッチします。

次のステップ

ロードバランサーが作成され、インスタンスがアタッチされたら、続く次のステップを完了して、ロードバランサーを設定します。

- ロードバランサー用の SSL/TLS 証明書を作成する
- ロードバランサー用のヘルスチェックをカスタマイズする

ロードバランサーに関する問題が発生した場合は、「ロードバランサーに関するトラブルシューティング」を参照してください。

Amazon Lightsail ロードバランサー用の SSL/TLS 証明書を作成する

Lightsail ロードバランサーを作成したら、Transport Layer Security (TLS) 証明書をアタッチして HTTPS を有効にできます。SSL/TLS 証明書を使用すると、ロードバランサーが暗号化されたウェブトラフィックを処理できるようになるため、ユーザーに安全な体験を提供できます。詳細については、「SSL/TLS 証明書」を参照してください。

前提条件

開始する前に、以下のものが必要です。

- Lightsail ロードバランサー。詳細については、「ロードバランサーを作成する」を参照してください。

証明書リクエストを作成する

1. Lightsail コンソールにサインインします。
2. Lightsail のホームページで [ネットワーキング] を選択します。
3. SSL/TLS 証明書を設定するロードバランサーの名前を選択します。
4. [Custom domains] (カスタムドメイン) タブを選択します。
5. [証明書の作成] を選択します。
6. 証明書の名前を入力するか、デフォルトをそのまま使用します。

   リソース名:

   - Lightsail アカウントの各 AWS リージョン 内で一意である必要があります。
   - 2〜255 文字を使用する必要があります。
   - 先頭と末尾は英数字または数字を使用する必要があります。
   - 英数字、数字、ピリオド、ダッシュ、アンダースコアを使用することができます。
7. プライマリドメイン (`www.example.com`) と、最大 9 つの代替ドメインまたはサブドメインを入力します。

   詳細については、「代替ドメインとサブドメインを SSL/TLS 証明書に追加する」を参照してください。

8. [証明書の作成] を選択します。

   Lightsail により検証プロセスが開始されます。72 時間以内にドメインを所有していることを検証してください。

   証明書を作成すると、ドメイン名とすべての代替ドメインおよびサブドメインと共に証明書が表示されます。ドメインおよびサブドメインごとに DNS レコードを作成する必要があります。

次のステップ

- 自分がドメインを所有していることを確認する

トピック

- 代替ドメインとサブドメインを Lightsail の SSL/TLS 証明書に追加する
- Amazon Lightsail で SSL/TLS 証明書を確認する
- 検証された SSL/TLS 証明書を Amazon Lightsail ロードバランサーにアタッチする
- Amazon Lightsail で SSL/TLS 証明書を削除する

代替ドメインとサブドメインを Lightsail の SSL/TLS 証明書に追加する

Lightsail ロードバランサーの SSL/TLS 証明書を作成するとき、代替ドメインとサブドメインを追加できます。これらの代替名を使用すると、ロードバランサーへのすべてのトラフィックが確実に暗号化されます。

プライマリドメインを指定する場合は、`www.example.com` などの完全修飾ドメイン名か、`example.com` などの apex ドメイン名を使用することができます。

ドメインおよびサブドメインの合計数が 10 を超えることはできないため、代替ドメインおよびサブドメインは証明書に最大 9 個追加できます。次のリストのようなエントリを追加することができます。

- example.com
- example.net
- blog.example.com
- myexamples.com

代替ドメインとサブドメインを含む証明書を作成するには

1. ロードバランサーを作成します (まだ作成していない場合)。
2. Lightsail のホームページで、[ネットワーキング] タブを選択します。
3. Lightsail ロードバランサーを選択します。
4. [Custom domains] (カスタムドメイン) タブを選択します。
5. [証明書の作成] を選択します。
6. 証明書の名前を入力するか、デフォルトの名前をそのまま使用します。

   リソース名:

   - Lightsail アカウントの各 AWS リージョン 内で一意である必要があります。
   - 2〜255 文字を使用する必要があります。
   - 先頭と末尾は英数字または数字を使用する必要があります。
   - 英数字、数字、ピリオド、ダッシュ、アンダースコアを使用することができます。
7. プライマリドメイン (`www.example.com`) と、最大 9 つの代替ドメインまたはサブドメインを入力します。
8. [証明書の作成] を選択します。

   作成後、72 時間以内にドメインを所有していることを検証してください。

次のステップ

- DNS を使用してドメインの所有権を検証する

  検証後、検証された証明書を選択して Lightsail ロードバランサーに関連付けることができます。
- セッション永続性を有効にする

Amazon Lightsail で SSL/TLS 証明書を確認する

Lightsail で SSL/TLS 証明書を作成した後は、その証明書に追加したすべてのドメインとサブドメインを、管理できていることを確認する必要があります。

目次

ステップ 1: ドメインの Lightsail DNS ゾーンを作成する

まだ行っていない場合は、ドメインの Lightsail DNS ゾーンを作成します。詳細については、「DNS ゾーンを作成してドメインの DNS レコードを管理する」を参照してください。

ステップ 2: ドメインの DNS ゾーンにレコードを追加する

作成した証明書により、一連の正規名 (CNAME) レコードが提供されます。これらのレコードをドメインの DNS ゾーンに追加して、そのドメインを所有し管理できていることを検証します。

> **Important**
>
> Lightsail は、ユーザーが証明書の作成時に指定したドメインまたはサブドメインが、そのユーザーにより管理されていることを自動的に検証しようとします。[Create certificate] (証明書を作成) を選択すると、CNAME レコードがドメインの DNS ゾーンに追加されます。自動検証が成功すると、証明書のステータスが [Attempting to validate your certificate] (証明書の検証を試行しています) から、[Valid, in use] (有効、使用中) に変わります。
> 自動検証が失敗した場合は、次の手順に進んでください。

以下のステップでは、Lightsail コンソールで CNAME レコードを取得して、ドメインの DNS ゾーンに追加する方法を説明します。

1. Lightsail コンソールにサインインします。
2. Lightsail ホームページの上部にあるナビゲーションメニューで [Account (アカウント)] を選択します。
3. ドロップダウンメニューで [Account (アカウント)] を選択します。
4. [証明書] タブを選択します。
5. 検証する証明書を見つけ、ドメインごとに追加する必要がある CNAME レコードの [Name] (名前) と [Value] (値) をメモします。

   Ctrl+C (Windows) または Cmd+C (Mac) を押してクリップボードにコピーします。

6. Windows を使用している場合はメモ帳、Mac を使用している場合は TextEdit などのテキストエディタを開きます。テキストファイルで、Windows を使用している場合は Ctrl+V、Mac を使用している場合は Cmd+V を押して、テキストファイルに値を貼り付けます。

   このテキストファイルは開いたままにしておきます。このガイドの後半でドメインの DNS ゾーンにレコードを追加するときに、これらの CNAME 値が必要になります。

7. Lightsail コンソールのトップナビゲーションバーの [ホーム] を選択します。
8. Lightsail のホームページで [Domains & DNS] (ドメイン & DNS) を選択します。
9. 証明書を使用するドメインの DNS ゾーンを選択します。
10. [DNS records] (DNS レコード) タブで [Add record] (レコードを追加) を選択します。
11. レコードタイプに、[CNAME] を選択します。
12. 証明書の CNAME レコードを含むテキストファイルに切り替えます。

    CNAME レコードの [名前] をコピーします。例えば、`_1bfb0b9ef15a50f9041e559d2c67b760` です。
13. DNS レコードページに切り替え、[Name] (名前) の情報を [Record name] (レコード名) フィールドに貼り付けます。

> **Important**
>
> ドメイン名が含まれている CNAME レコード (`.example.com` など) を追加すると、ドメイン名が重複したレコード (`.example.com.example.com` など) になります。重複

を避けるために、必要な CNAME の一部だけが追加されるようにエントリを編集してください。_1bfb0b9ef15a50f9041e559d2c67b760 となります。

14. CNAME レコードの [値] をコピーします。例えば、_c9a0c385eda13283350e35f297469a13.hkvuiqjoua.acm-validations.aws. です。
15. DNS レコードページに切り替え、[Value] (値) の情報を [Route traffic to] (トラフィックのルーティング先) フィールドに貼り付けます。
16. [Save] (保存) を選択して、レコードを追加します。
17. 代替サブドメインがある場合、[レコードの追加] を選択して別のレコードを追加します。

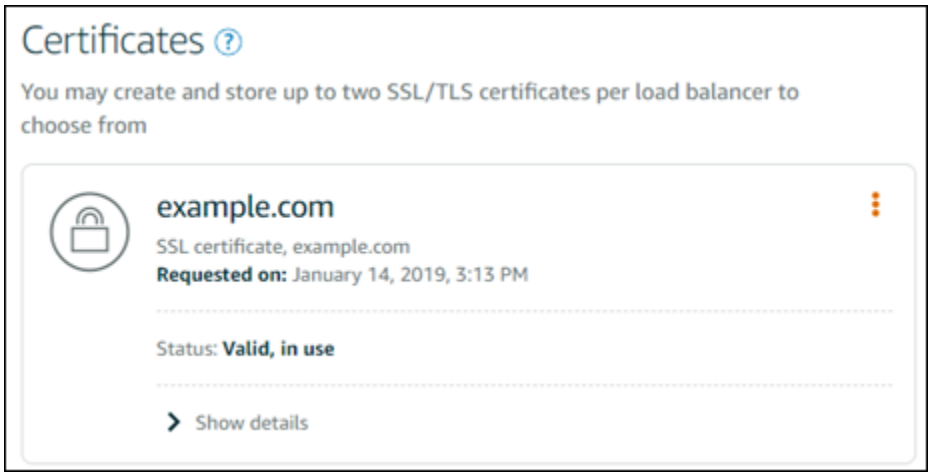
> **Note**
>
> 詳細については、「代替ドメインとサブドメインを Amazon Lightsail の SSL/TLS 証明書に追加する」を参照してください。

18. 代替サブドメインの CNAME レコードを (1 つ以上) 追加するには、ステップ 11〜17 を繰り返します。

DNS ゾーン管理ページからは、エイリアス (A) レコードを追加してロードバランサーをポイントしたり、他の Lightsail リソースを追加することもできます。

完了すると、DNS ゾーンは以下のスクリーンショットのようになります。

しばらくすると、ドメインの検証が完了し、証明書に次のメッセージが表示されます。

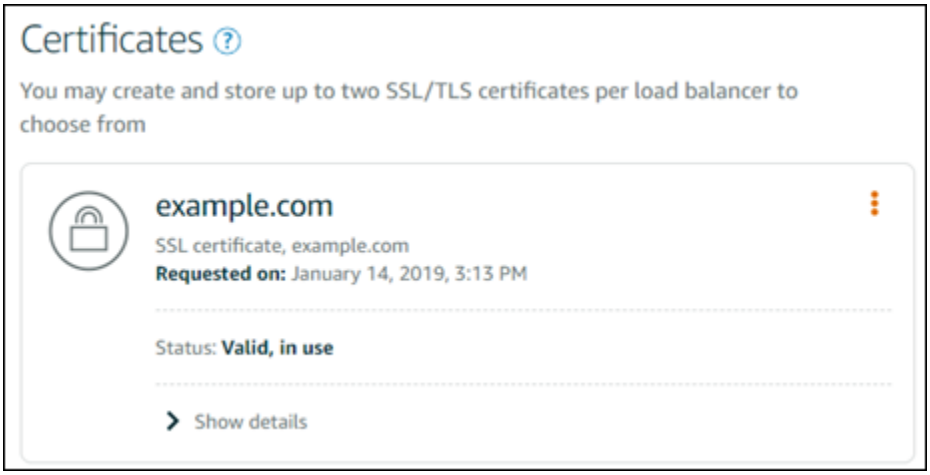


次のステップ

ドメインが確認されたら、「検証された SSL/TLS 証明書をロードバランサーにアタッチする」準備が整います。

検証された SSL/TLS 証明書を Amazon Lightsail ロードバランサーにアタッチする

ドメインを管理していることが確認されると、証明書のステータスが [Valid] (有効) に変わります。



次のステップでは、Lightsail ロードバランサーに証明書をアタッチします。

1. Lightsail のホームページで [ネットワーキング] を選択します。
2. ロードバランサーを選択します。
3. [Custom domains] (カスタムドメイン) タブを選択します。
4. [Certificates] (証明書) セクションで、[Attach certificate] (証明書のアタッチ) を選択します。
5. ドロップダウンリストから証明書を選択します。
6. [Attach] (アタッチ) を選択して証明書をアタッチします。

Amazon Lightsail で SSL/TLS 証明書を削除する

使用しなくなった SSL/TLS 証明書を削除できます。たとえば、証明書の有効期限が切れており、検証済みの更新された証明書を既にアタッチしている場合などです。証明書を削除する前に複製する場合、以下のステップ 5 と同じショートカットメニューから [重複] を選択します。

> **Important**
>
> 削除する証明書が有効であり、使用中の場合、ロードバランサーは暗号化 (HTTPS) トラフィックを処理できなくなります。Lightsail ロードバランサーでは、引き続き非暗号化 (HTTP) トラフィックがサポートされます。

SSL/TLS 証明書の削除は元に戻すことができません。365 日間に作成できる証明書の数にはクォータがあります。詳細については、AWS Certificate Manager ユーザーガイドの「クォータ」を参照してください。

1. Lightsail のホームページで [ネットワーキング] を選択します。
2. SSL/TLS 証明書がアタッチされているロードバランサーを選択します。
3. ロードバランサーの管理ページで [インバウンドトラフィック] タブを選択します。
4. ページの [証明書] セクションで、削除する証明書の省略記号アイコン (:) を選択し、[削除] を選択します。

   [削除] オプションは、削除する証明書が使用中の場合は使用できません。使用中の証明書を削除するには、まず証明書を使用しているロードバランサーの証明書を変更するか、証明書を使用しているロードバランサーで HTTPS を無効にする必要があります。

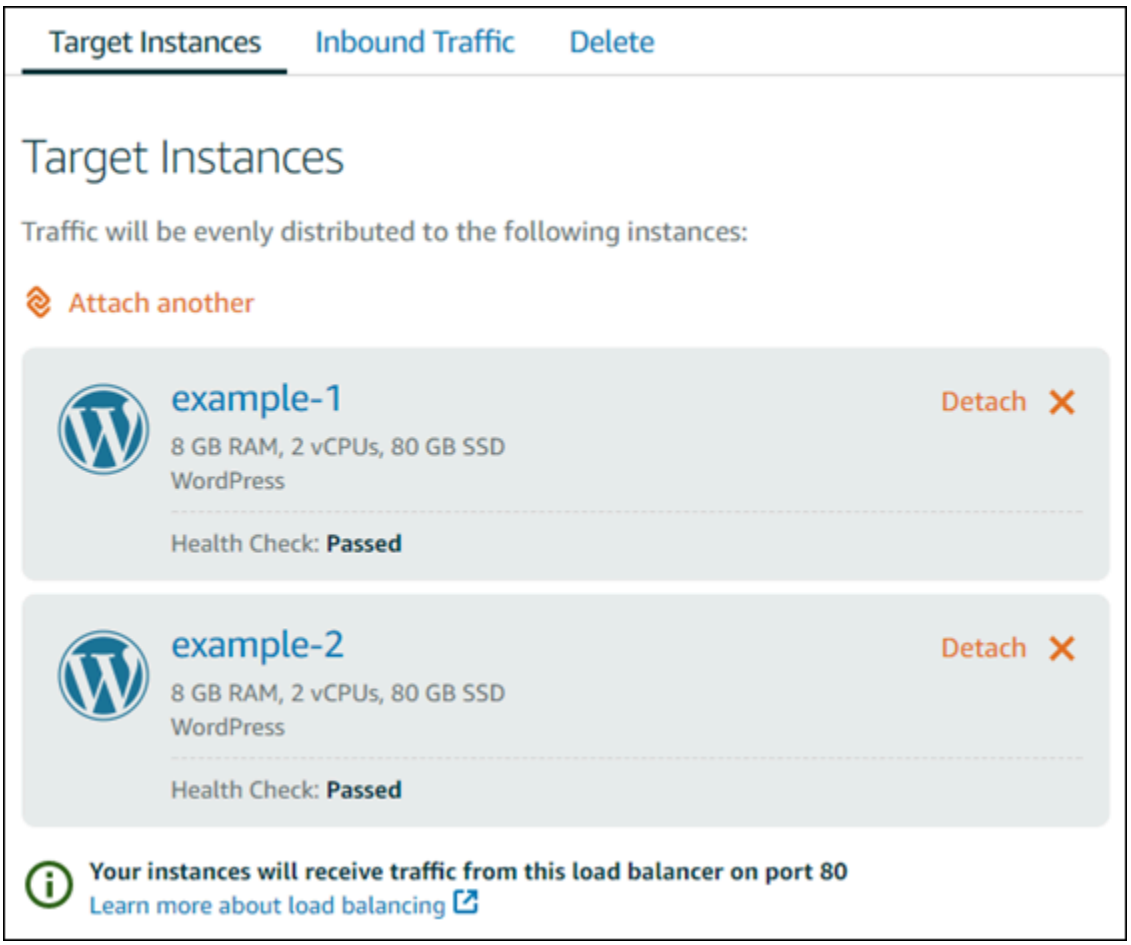
Amazon Lightsail ロードバランサーの設定を更新する

Lightsail ロードバランサーは、AWS リージョン と名前を選択するだけで作成できます。このトピックでは、ロードバランサーを更新して他のオプションを有効にする方法について説明します。

ロードバランサーをまだ作成していない場合は、作成する必要があります。ロードバランサーの作成

ヘルスチェック

まず、ロードバランシング用のインスタンスを設定 をします。完了したら、インスタンスをロードバランサーにアタッチできます。インスタンスをアタッチすると、ヘルスチェックプロセスが開始され、ロードバランサー管理ページに成功または失敗のメッセージが表示されます。



ヘルスチェックのパスをカスタマイズすることもできます。たとえば、ホームページのロードに時間がかかる場合や多くの画像が使用されている場合、すぐにロードされる別のページをチェックするように Lightsail を設定します。ロードバランサーのヘルスチェックパスのカスタマイズ

暗号化されたトラフィック (HTTPS)

ウェブサイトでより安全なユーザー体験を実現するため、HTTPS を設定できます。ロードバランサーを設定したら SSL/TLS 証明書を作成して検証する 3 ステップのプロセスです。

HTTPS の詳細

セッション永続性

セッション永続性は、ユーザーのブラウザでセッション情報をローカルに保存する場合に役立ちます。たとえば、Lightsail にショッピングカートのある Magento e コマースアプリケーションを実行しているとします。セッション永続性を有効にした場合、ユーザーがショッピングカートに商品を追加してセッションを終了しても、戻ってくるとカートに商品が残っています。

また、永続的なセッションの Cookie の有効期間を調整することもできます。これは、特に長い有効期間や短い有効期間が必要な場合に役立ちます。詳細については、「ロードバランサーのセッション永続性を有効にする」を参照してください。

ロードバランシング用の Lightsail インスタンスを設定する

インスタンスを Lightsail ロードバランサーにアタッチする前に、アプリケーションの設定を評価する必要があります。たとえば、ロードバランサーは多くの場合、データ層がアプリケーションの残りの部分と分離されている方が適切に動作します。このトピックでは、各 Lightsail インスタンスについて説明し、ロードバランシング (水平スケーリング) を行うかどうかとアプリケーションの最適な設定方法に関する推奨事項を示します。

一般的なガイドライン: データベースを使用するアプリケーション

データベースを使用する Lightsail アプリケーションの場合、データベースインスタンスが 1 つだけになるように、データベースインスタンスをアプリケーションの残りの部分から分離することをお勧めします。主な理由は、複数のデータベースにデータが書き込まれないようにすることです。1 つのデータベースインスタンスを作成しない場合、ユーザーがたまたまヒットしたインスタンス上のデータベースにデータが書き込まれます。

WordPress

水平スケーリングを行いますか? はい、WordPress ブログまたはウェブサイトで行います。

Lightsail ロードバランサーを使用する前の設定の推奨事項

- ロードバランサーの背後で実行されているあらゆる WordPress インスタンスが同じ場所に情報を保存および取得するように、データベースを分離します。データベースのより高いパフォーマンスが必要な場合、ウェブサーバーとは別個に処理能力やメモリをレプリケートまたは変更することができます。
- ファイルと静的コンテンツを Lightsail バケットにオフロードします。これを実行するには、WordPress のウェブサイトに WP Offload Media Lite プラグインをインストールし、Lightsail バケットに接続するように設定する必要があります。詳細については、「チュートリアル: WordPress インスタンスをストレージ バケットに接続する」を参照してください。

Node.js

水平スケーリングを行いますか? はい。ただし、いくつかの考慮事項があります。

Lightsail ロードバランサーを使用する前の設定の推奨事項

- Lightsail では、Bitnami によってパッケージ化されている Node.js スタックには、Node.js、Apache、Redis (メモリ内データベース)、Python が含まれています。デプロイするアプリケーションに応じて、いくつかのサーバー間でロードバランシングすることができます。ただし、すべてのウェブサーバー間でトラフィックが分散され、Redis が別のサーバーに移動されるようにロードバランサーを設定する必要があります。
- Redis サーバーを別のサーバーに分割して、すべてのインスタンスと通信します。必要に応じて、データベースサーバーを追加します。
- Redis の主なユースケースの 1 つは、データをローカルにキャッシュするため、中央のデータベースに継続的にヒットする必要はありません。Redis によるパフォーマンス向上を活用するには、セッション永続性を有効にすることをお勧めします。詳細については、「ロードバランサーのセッション永続性を有効にする」を参照してください。
- 共有 Redis ノードを作成することもできるため、セッション永続性を使用する各マシンでノードを共有したり、ローカルキャッシュを使用したりすることもできます。
- Apache を使用してロードバランサーをデプロイする場合は、`mod_proxy_balancer` を Apache サーバーに含めることを検討してください。

詳細については、「Scaling Node.js applications」を参照してください。

Magento

水平スケーリングを行いますか? はい。

Lightsail ロードバランサーを使用する前の設定の推奨事項

- Amazon RDS データベースなど、追加のコンポーネントを使用する Magento の AWS リファレンスデプロイを使用できます。「AWS のTerraform Magento Adobe Commerce」を参照してください。
- 必ず、セッション永続性を有効にしてください。Magento はショッピングカートを使用しているため、セッションをまたいで複数回訪問するお客様が、新しいセッションで戻ってきたときもショッピングカート内に商品を保持することができます。詳細については、「ロードバランサーのセッション永続性を有効にする」を参照してください。

GitLab

水平スケーリングを行いますか? はい。ただし、考慮事項があります。

Lightsail ロードバランサーを使用する前の設定の推奨事項

以下を準備する必要があります。

- 実行されており、使用準備ができた Redis ノード
- 共有ネットワークストレージサーバー (NFS)
- アプリケーション用の一元化されたデータベース (MySQL または PostgreSQL)。上記のデータベースに関する一般的なガイドラインを参照してください。

詳細については、GitLab ウェブサイトの「高可用性」を参照してください。

> **Note**
>
> 上記で言及されている共有ネットワークストレージサーバー (NFS) は、GitLab ブループリントでは現在利用できません。

Drupal

水平スケーリングを行いますか? はい。Drupal には、アプリケーションを水平スケーリングする方法を説明する公式ドキュメント「Server Scaling」があります。

Lightsail ロードバランサーを使用する前の設定の推奨事項

異なるインスタンス間でファイルが同期されるように Drupal モジュールを設定する必要があります。Drupal ウェブサイトにはいくつかのモジュールがありますが、本稼働使用ではなくプロトタイプ作成の方に適している可能性があります。

ファイルを Amazon S3 に保存できるモジュールを使用します。これにより、ターゲットインスタンスごとに別個のコピーを保持するのではなく、ファイルを一元化された場所に保存できます。このようにして、ファイルを編集した場合、ヒットしたインスタンスに関係なく、一元化されたストアから更新を取得してユーザーに同じファイルを表示できます。

- Amazon S3 ファイルシステム
- コンテンツの同期

詳細については、「Scaling Drupal horizontally and in cloud」(Drupal を水平方向とクラウドでスケーリングする) を参照してください。

LAMP スタック

水平スケーリングを行いますか? はい。

Lightsail ロードバランサーを使用する前の設定の推奨事項

- 別のインスタンスにデータベースを作成する必要があります。ロードバランサーの背後にあるすべてのインスタンスは、この別個のデータベースインスタンスをポイントするため、同じ場所に情報を保存および取得できます。
- デプロイするアプリケーションに応じて、ファイルシステムを共有する方法を検討します (NFS、Lightsail ブロックストレージディスク、または Amazon S3 ストレージ)。

MEAN スタック

水平スケーリングを行いますか? はい。

Lightsail ロードバランサーを使用する前の設定の推奨事項

MongoDB を別のマシンに移動し、Lightsail インスタンス間でルートドキュメントを共有するメカニズムを設定します。

Redmine

水平スケーリングを行いますか? はい。

Lightsail ロードバランサーを使用する前の設定の推奨事項

- Redmine_s3 プラグインを取得し、添付ファイルをローカルファイルシステムではなく Amazon S3 に格納します。
- 別のインスタンスにデータベースを分離します。

Nginx

水平スケーリングを行いますか? はい。

Nginx を実行する Lightsail インスタンスを 1 つ以上、Lightsail ロードバランサーにアタッチできます。詳細については、「Scaling Web Applications with NGINX, Part 1: Load Balancing」を参照してください。

Joomla!

水平スケーリングを行いますか? はい。ただし、考慮事項があります。

Lightsail ロードバランサーを使用する前の設定の推奨事項

Joomla ウェブサイトに公式ドキュメントはありませんが、コミュニティフォーラムで議論されています。一部のユーザーは、次の設定のクラスターを作成して Joomla インスタンスを水平スケーリングしています。

- セッション永続性を有効にするように設定された Lightsail ロードバランサー。詳細については、「ロードバランサーのセッション永続性を有効にする」を参照してください。
- Joomla! のドキュメントルートが同期された状態でロードバランサーにアタッチされた Joomla を実行する複数の Lightsail。これを行うには、Rsync などのツールを使用する、すべての Lightsail インスタンス間でコンテンツを同期できるNFS サーバーを持つ、または AWS を使用してファイルを共有します。
- レプリケーションクラスターで設定されたいくつかのデータベースサーバー。
- Lightsail インスタンスごとに設定されている同じキャッシュシステム。JotCache など、役に立つ拡張機能がいくつかあります。

Amazon Lightsail ロードバランサーに TLS セキュリティポリシーを設定します。

Amazon Lightsail ロードバランサーで HTTPS を有効にすると、暗号化された接続の TLS セキュリティポリシーを設定できます。このガイドでは、Lightsail ロードバランサーで設定できるセキュリティポリシーと、ロードバランサーのセキュリティポリシーを更新する手順について説明します。ロードバランサーの詳細については、「ロードバランサー」を参照してください。

セキュリティポリシーの概要

Lightsail 負荷分散は、セキュリティポリシーと呼ばれる Secure Socket Layer (SSL) ネゴシエーション設定を使用して、クライアントとロードバランサーの間の SSL 接続をネゴシエーションします。セキュリティポリシーはプロトコルと暗号の組み合わせです。プロトコルは、クライアントとサーバーの間の安全な接続を確立し、クライアントとロードバランサーの間で受け渡しされるすべてのデータのプライバシーを保証します。暗号とは、暗号化キーを使用してコード化されたメッセージを作成する暗号化アルゴリズムです。プロトコルは、複数の暗号を使用し、インターネットを介し

てデータを暗号化します。接続ネゴシエーションのプロセスで、クライアントとロードバランサーでは、それぞれサポートされる暗号とプロトコルのリストが優先される順に表示されます。デフォルトでは、サーバーのリストで最初にクライアントの暗号と一致した暗号が安全な接続用に選択されます。Lightsail ロードバランサーは、クライアント接続またはターゲット接続の SSL 再ネゴシエーションをサポートしていません。

Lightsail ロードバランサーで HTTPS を有効にすると、TLS-2016-08セキュリティポリシーがデフォルトで設定されます。このガイドで後述するように、必要に応じて別のセキュリティポリシーを設定できます。フロントエンド接続のみに使用するセキュリティポリシーを選択できます。バックエンド接続には、常に TLS-2016-08 セキュリティポリシーが使用されます。Lightsail ロードバランサーはカスタムセキュリティポリシーをサポートしていません。

サポートされているセキュリティポリシーとプロトコル

Lightsail ロードバランサーは、以下のセキュリティポリシーとプロトコルで構成できます。

Security policies	TLS-2016-08 (default)	TLS-FS-1-2-Res-2019-08
TLS Protocols		
Protocol-TLSv1	✓	
Protocol-TLSv1.1	✓	
Protocol-TLSv1.2	✓	✓
TLS Ciphers		
ECDHE-ECDSA-AES128-GCM-SHA256	✓	✓
ECDHE-RSA-AES128-GCM-SHA256	✓	✓
ECDHE-ECDSA-AES128-SHA256	✓	✓
ECDHE-RSA-AES128-SHA256	✓	✓
ECDHE-ECDSA-AES128-SHA	✓	
ECDHE-RSA-AES128-SHA	✓	
ECDHE-ECDSA-AES256-GCM-SHA384	✓	✓
ECDHE-RSA-AES256-GCM-SHA384	✓	✓
ECDHE-ECDSA-AES256-SHA384	✓	✓
ECDHE-RSA-AES256-SHA384	✓	✓
ECDHE-RSA-AES256-SHA	✓	
ECDHE-ECDSA-AES256-SHA	✓	
AES128-GCM-SHA256	✓	
AES128-SHA256	✓	
AES128-SHA	✓	

前提条件を満たす

以下の前提条件を完了します (まだの場合)。

- ロードバランサーを作成してインスタンスをアタッチする。詳細については、「ロードバランサーを作成してインスタンスをアタッチする」を参照してください。
- SSL/TLS 証明書を作成し、ロードバランサーにアタッチして HTTPS を有効にします。詳細については、「Create an SSL/TLS certificate for your Lightsail load balancer」(Lightsail ロードバランサーの SSL/TLS 証明書を作成する) を参照してください。証明書の詳細については、「SSL/TLS 証明書」を参照してください。

Lightsail コンソールを使用してセキュリティポリシーを設定する

Lightsail コンソールを使用してセキュリティポリシーを設定するには、以下の手順を実行します。

1. Lightsail コンソールにサインインします。
2. lightsail のホームページで [Networking] (ネットワーク) タブを選択します。
3. TLS セキュリティポリシーを設定するロードバランサーの名前を選択します。
4. [インバウンドトラフィック] タブを選択します。
5. ページの [TLS security protocols] (TLS セキュリティプロトコル) セクションで [Change protocols] (プロトコルを変更) を選択します。
6. [Supported protocols] (サポートされているプロトコル) ドロップダウンメニューで、次のいずれかのオプションを選択します。

   - [TLS バージョン 1.2] — このオプションは最も安全ですが、古いブラウザは接続できない可能性があります。
   - [TLS バージョン 1.0、1.1、および 1.2] — このオプションは、ブラウザとの互換性が最も高くなります。
7. [Save] (保存) を選択して、選択したプロトコルをロードバランサーに適用します。

   変更が有効になるまで、少し時間がかかります。

を使用してセキュリティポリシーを設定します。 AWS CLI

AWS Command Line Interface (AWS CLI)を使用してセキュリティポリシーを設定するには、次の手順を実行します。これは、`update-load-balancer-attribute` コマンドを使用して行います。

詳細については、『AWS CLI コマンドリファレンス』update-load-balancer-attributeのを参照してください。

> **Note**
>
> この手順を続行する前に AWS CLI 、をインストールして Lightsail 用に設定する必要があります。詳細については、「Lightsail AWS CLI と連携するようにを設定する」を参照してください。

1. ターミナルまたはコマンドプロントウィンドウを開きます。
2. 次のコマンドを入力して、ロードバランサーの TLS セキュリティポリシーを変更します。

```
aws lightsail update-load-balancer-attribute --load-balancer-name LoadBalancerName
 --attribute-name TlsPolicyName --attribute-value AttributeValue
```

コマンドで、次のサンプルテキストを独自のテキストに置き換えます。

- *LoadBalancerName*TLS セキュリティポリシーを変更したいロードバランサーの名前を入力します。
- *AttributeValue*TLS-2016-08TLS-FS-1-2-Res-2019-08またはのセキュリティポリシーを使用します。

> **Note**
>
> コマンドの TlsPolicyName 属性は、ロードバランサーで設定されている TLS セキュリティポリシーを編集することを指定します。

例：

```
aws lightsail update-load-balancer-attribute --load-balancer-name MyLoadBalancer --
attribute-name TlsPolicyName --attribute-value TLS-2016-08
```

変更が有効になるまで、少し時間がかかります。

Lightsail ロードバランサーに HTTP から HTTPS へのリダイレクトを設定する

Amazon Lightsail ロードバランサーで HTTPS を設定した後、HTTP から HTTPS へのリダイレクトを設定して、HTTP 接続を使用してウェブサイトまたはウェブアプリケーションを閲覧するユーザーが暗号化された HTTPS 接続に自動的にリダイレクトされるようにすることができます。ロードバランサーの詳細については、「ロードバランサー」を参照してください。

前提条件を満たす

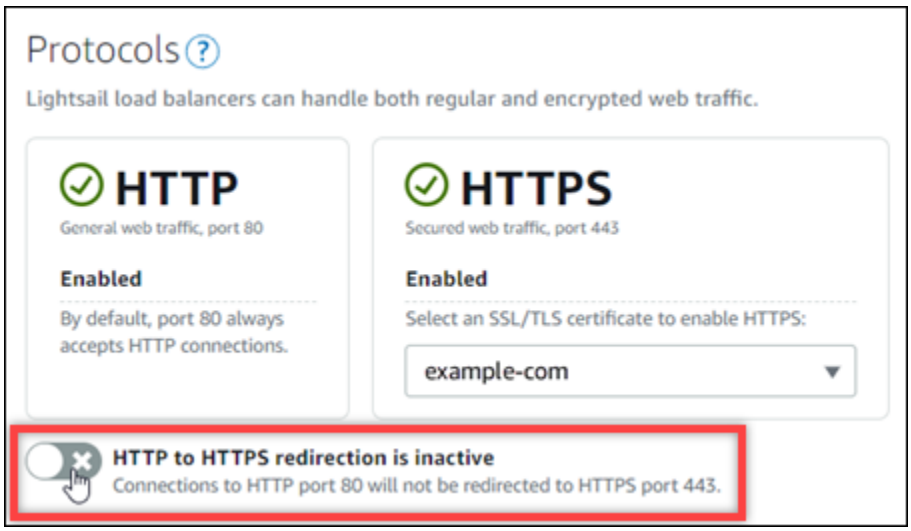
以下の前提条件を完了します (まだの場合)。

- ロードバランサーを作成してインスタンスをアタッチする。詳細については、「ロードバランサーを作成してインスタンスをアタッチする」を参照してください。
- SSL/TLS 証明書を作成し、ロードバランサーにアタッチして HTTPS を有効にします。詳細については、「Lightsail ロードバランサー用の SSL/TLS 証明書を作成する」を参照してください。証明書の詳細については、「SSL/TLS 証明書」を参照してください。

Lightsail コンソールを使用してロードバランサーでの HTTPS リダイレクトを設定する

Lightsail コンソールを使用してロードバランサーで HTTPS リダイレクトを設定するには、次の手順を実行します。

1. Lightsail コンソールにサインインします。
2. Lightsail のホームページで、[ネットワーキング] タブを選択します。
3. HTTPS リダイレクトを設定するロードバランサーの名前を選択します。
4. [インバウンドトラフィック] タブを選択します。
5. ページの [Protocols] (プロトコル) セクションでは、次のいずれかのアクションを実行できます。



- HTTP から HTTPS へのリダイレクトをオンにするには、方向オプションをアクティブに切り替えます。
- HTTP から HTTPS へのリダイレクトをオフにするには、方向オプションを非アクティブに切り替えます。

変更が有効になるまで、少し時間がかかります。

AWS CLI を使用して、ロードバランサーに HTTP から HTTPS へのリダイレクトを設定する

AWS Command Line Interface (AWS CLI)を使用してロードバランサーで HTTPS リダイレクトを設定するには、次の手順を実行します。これは、`update-load-balancer-attribute` コマンドを使用して行います。詳細については、「AWS CLI コマンドリファレンス」の「update-load-balancer-attribute」を参照してください。

> **Note**
>
> この手順を続行する前に、AWS CLI をインストールして Lightsail 用に設定する必要があります。　詳細については、「Lightsail で使用するために AWS CLI を設定する」を参照してください。

1. ターミナルまたはコマンドプロントウィンドウを開きます。
2. 次のコマンドを入力して、ロードバランサーで HTTPS リダイレクトを設定します。

```
aws lightsail update-load-balancer-attribute --load-balancer-name LoadBalancerName
 --attribute-name HttpsRedirectionEnabled --attribute-value AttributeValue
```

コマンドで、次のサンプルテキストを独自のテキストに置き換えます。

- *LoadBalancerName* を、HTTP から HTTPS へのリダイレクトをアクティブ化または非アクティブ化するロードバランサーの名前に置き換えます。
- *AttributeValue* を、リダイレクトをアクティブ化する `true`、またはリダイレクトを非アクティブ化する `false` に置き換えます。

> **Note**
>
> コマンドの `HttpsRedirectionEnabled` 属性は、指定されたロードバランサーについて HTTPS リダイレクトが有効か無効かを編集することを指定します。

例:

- ロードバランサーで HTTP から HTTPS へのリダイレクトをアクティブ化するには、次の手順を実行します。

```
aws lightsail update-load-balancer-attribute --load-balancer-name MyLoadBalancer
 --attribute-name HttpsRedirectionEnabled --attribute-value true
```

- ロードバランサーで HTTP から HTTPS へのリダイレクトを非アクティブ化するには、次の手順を実行します。

```
aws lightsail update-load-balancer-attribute --load-balancer-name MyLoadBalancer
 --attribute-name HttpsRedirectionEnabled --attribute-value false
```

変更が有効になるまで、少し時間がかかります。

Lightsail ロードバランサーのセッション維持機能を有効にする

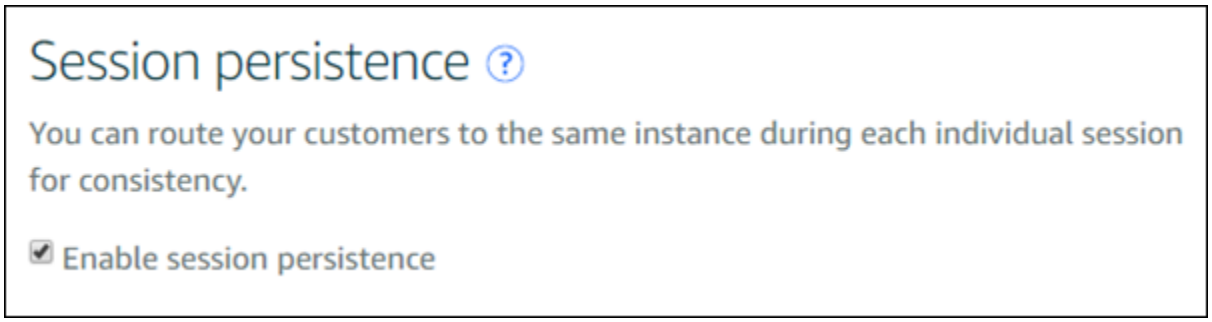
ユーザーのセッション永続性を有効にすることができます。これは、ユーザーのブラウザでセッション情報をローカルに保存する場合に役立ちます。たとえば、Lightsail にショッピングカートのある

Magento e コマースアプリケーションを実行しているとします。セッション永続性を有効にした場合、ユーザーがショッピングカートに商品を追加してサイトから離れても、戻ってくるとカートに商品が残っています。

AWS Command Line Interface (AWS CLI) または Lightsail API を使用して、Cookie の維持期間を調整することもできます。

セッション永続性を有効にする

1. Lightsail のホームページで [ネットワーキング] を選択します。
2. ロードバランサーを選択して管理します。
3. [インバウンドトラフィック] タブを選択します。
4. [セッション永続性を有効にする] を選択します。



Cookie の有効期間を調整する

また、永続的なセッションの Cookie の有効期間を調整することもできます。これは、特に長い有効期間や短い有効期間が必要な場合に役立ちます。たとえば、多くの e コマースサイト期間では有効期間が非常に長くなっています。これにより、顧客がサイトを離れて戻ってきても、ショッピングカート内の商品が失われません。

まだの場合は AWS CLI をセットアップして設定します。

AWS Command Line Interface と連携するための Amazon Lightsail の設定

1. コマンドプロンプトまたはターミナルウィンドウを開きます。
2. 次の AWS CLI コマンドを入力し、Cookie の有効期間を 3 日に延長します (259,200 秒)。

```
aws lightsail update-load-balancer-attribute --load-balancer-name LoadBalancerName
 --attribute-name SessionStickiness_LB_CookieDurationSeconds --attribute-value
 259200
```

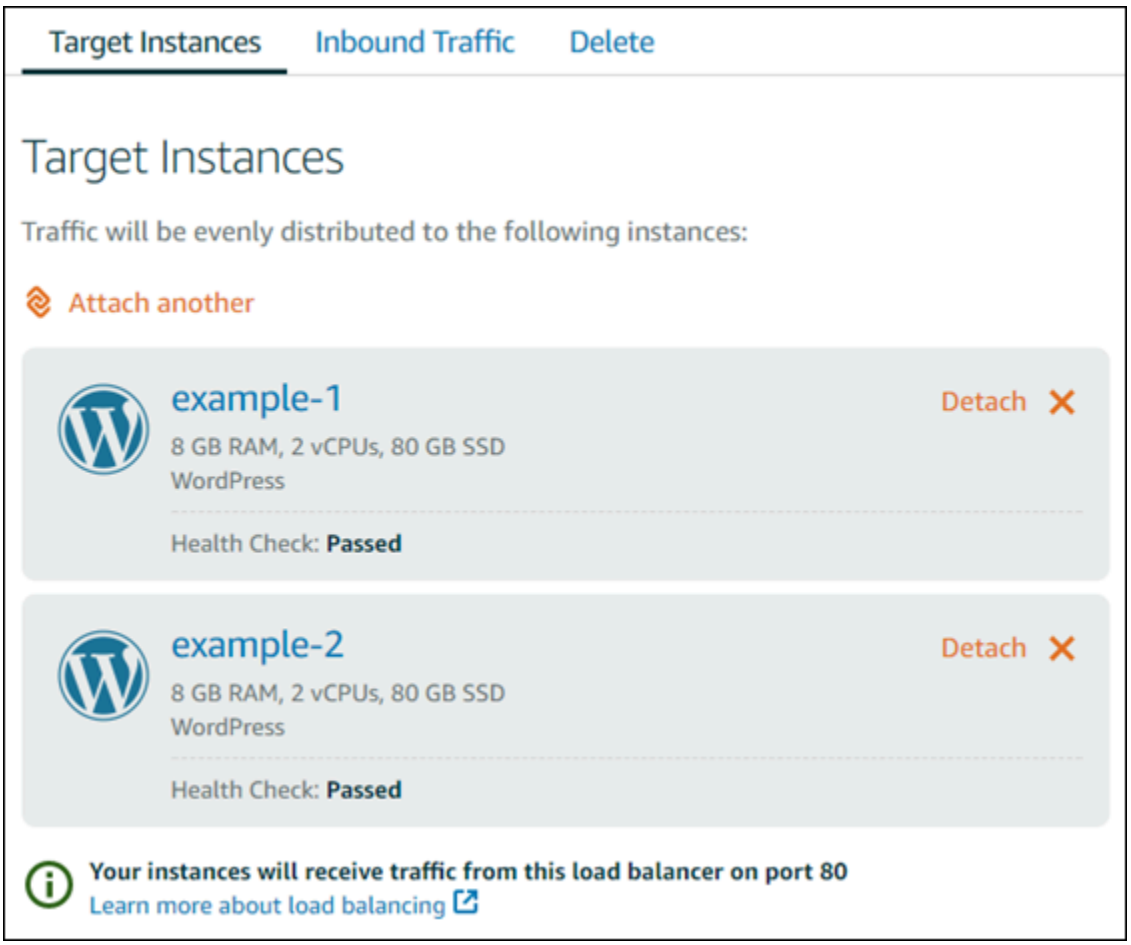
コマンドで、*LoadBalancerName* をロードバランサーの名前に置き換えます。

成功すると、次のような応答が表示されます。


```
{
    "operations": [
        {
            "status": "Succeeded",
            "resourceType": "LoadBalancer",
            "isTerminal": true,
            "operationDetails": "SessionStickiness_LB_CookieDurationSeconds",
            "statusChangedAt": 1511758936.174,
            "location": {
                "availabilityZone": "all",
                "regionName": "us-west-2"
            },
            "operationType": "UpdateLoadBalancerAttribute",
            "resourceName": "example-load-balancer",
            "id": "681c2bd9-9a51-402b-8ad2-12345EXAMPLE",
            "createdAt": 1511758936.174
        }
    ]
}
```


Amazon Lightsail ロードバランサーのヘルスチェック

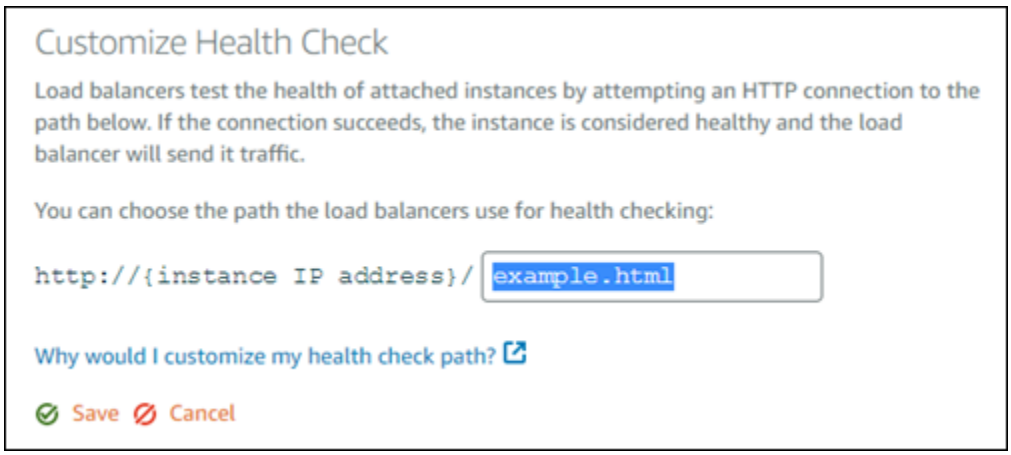
ヘルスチェックは、Lightsail インスタンスをロードバランサーにアタッチするとすぐに開始され、その後 30 秒ごとに実行されます。ヘルスチェックのステータスを表示するには、ロードバランサーの管理ページを参照してください。



ヘルスチェックのパスをカスタマイズする

ヘルスチェックのパスをカスタマイズすることが必要な場合があります。たとえば、ホームページのロードに時間がかかる場合や多くの画像が使用されている場合、すぐにロードされる別のページをチェックするように Lightsail を設定します。

1. Lightsail のホームページで [ネットワーキング] を選択します。
2. ロードバランサーを選択して管理します。
3. [ターゲットインスタンス] タブで [ヘルスチェックのカスタマイズ] を選択します。
4. ヘルスチェックの有効なパスを入力し、[保存] を選択します。



ヘルスチェックメトリクス

次のメトリックスはヘルスチェックの問題を診断するのに役立ちます。AWS Command Line Interface または Lightsail API を使用して、特定のヘルスチェックメトリクスに関する情報を返します。

- **ClientTLSNegotiationErrorCount** – クライアントで開始された TLS 接続のうち、ロードバランサーとのセッションの確立に失敗したものの数。暗号化またはプロトコルの不一致が原因である場合があります。

  Statistics: 最も有用な統計は Sum です。
- **HealthyHostCount** – 正常と見なされるターゲットインスタンスの数。

  Statistics: 最も有用な統計は Average、Minimum、Maximum です。
- **UnhealthyHostCount** – 異常と見なされるターゲットインスタンスの数。

  Statistics: 最も有用な統計は Average、Minimum、Maximum です。
- **HTTPCode_LB_4XX_Count** – ロードバランサーから送信される HTTP 4XX クライアントエラーコードの数。リクエストの形式が不正な場合、または不完全な場合は、クライアントエラーが生成されます。それらのリクエストはターゲットインスタンスで受信されません。この数には、ターゲットインスタンスによって生成される応答コードは含まれません。

  Statistics: 最も有用な統計は Sum です。Minimum、Maximum、および Average はすべて 1 を返すことに注意してください。
- **HTTPCode_LB_5XX_Count** – ロードバランサーから送信される HTTP 5XX サーバーエラーコードの数。この数には、ターゲットインスタンスによって生成される応答コードは含まれません。

Statistics: 最も有用な統計は Sum です。Minimum、Maximum、および Average はすべて 1 を返すことに注意してください。Minimum、Maximum、および Average はすべて 1 を返すことに注意してください。

- **HTTPCode_Instance_2XX_Count** – ターゲットインスタンスで生成された HTTP 応答コードの数。これには、ロードバランサーによって生成される応答コードは含まれません。

  Statistics: 最も有用な統計は Sum です。Minimum、Maximum、および Average はすべて 1 を返すことに注意してください。

- **HTTPCode_Instance_3XX_Count** – ターゲットインスタンスで生成された HTTP 応答コードの数。これには、ロードバランサーによって生成される応答コードは含まれません。

  Statistics: 最も有用な統計は Sum です。Minimum、Maximum、および Average はすべて 1 を返すことに注意してください。

- **HTTPCode_Instance_4XX_Count** – ターゲットインスタンスで生成された HTTP 応答コードの数。これには、ロードバランサーによって生成される応答コードは含まれません。

  Statistics: 最も有用な統計は Sum です。Minimum、Maximum、および Average はすべて 1 を返すことに注意してください。

- **HTTPCode_Instance_5XX_Count** – ターゲットインスタンスで生成された HTTP 応答コードの数。これには、ロードバランサーによって生成される応答コードは含まれません。

  Statistics: 最も有用な統計は Sum です。Minimum、Maximum、および Average はすべて 1 を返すことに注意してください。

- **InstanceResponseTime** – ロードバランサーからリクエストを送信してから、ターゲットインスタンスからの応答を受信するまでの経過時間 (秒)。

  Statistics: 最も有用な統計は Average です。

- **RejectedConnectionCount** – ロードバランサーが接続の最大数に達したため、拒否された接続の数。

  Statistics: 最も有用な統計は Sum です。

- **RequestCount** – IPv4 経由で処理されたリクエストの数。この数には、ロードバランサーのターゲットインスタンスによって生成されたレスポンスを含むリクエストのみが含まれます。

  Statistics: 最も有用な統計は Sum です。Minimum、Maximum、および Average はすべて 1 を返すことに注意してください。

トピック

Lightsail ロードバランサーヘルスチェックのステータス

デフォルトでは、Lightsail はウェブアプリケーションのルート ("/") でインスタンスに対するヘルスチェックを実行します。ヘルスチェックは、ロードバランサーから正常なインスタンスにのみリクエストを送信できるように、登録されたインスタンスのヘルス状態をモニタリングするために使用されます。ヘルスチェックは、インスタンスをロードバランサーにアタッチするとすぐに開始します。

以下のいずれかのステータスが返されます。

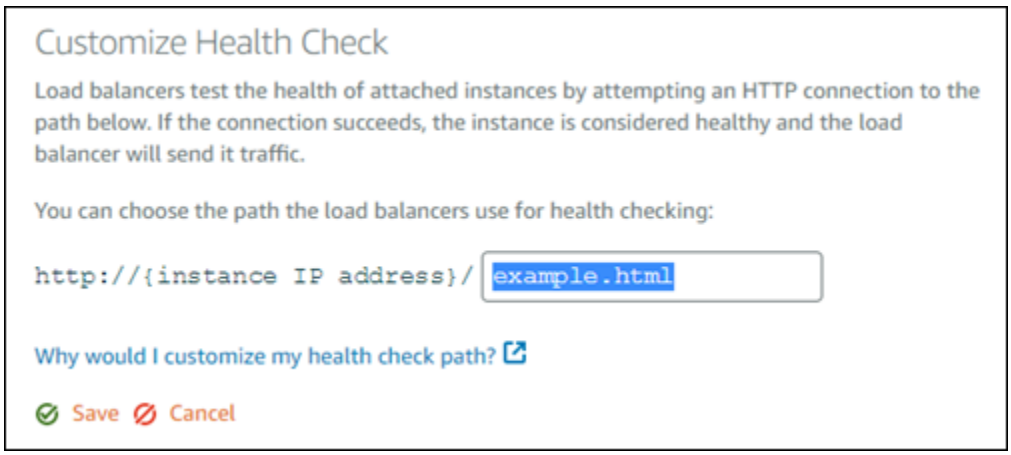
- [成功]
- [Failed] (失敗)

ヘルスチェックが失敗した場合、AWS Command Line Interface または Lightsail API を使用することで問題の原因を調べることができます。詳細については、トラブルシューティングガイドを参照してください。

ヘルスチェックのパスをカスタマイズする

ヘルスチェックのパスをカスタマイズすることが必要な場合があります。たとえば、ホームページのロードに時間がかかる場合や多くの画像が使用されている場合、すぐにロードされる別のページをチェックするように Lightsail を設定します。

1. Lightsail のホームページで [ネットワーキング] を選択します。
2. ロードバランサーを選択して管理します。
3. [ターゲットインスタンス] タブで [ヘルスチェックのカスタマイズ] を選択します。
4. ヘルスチェックの有効なパスを入力し、[保存] を選択します。



Lightsail ロードバランサーからインスタンスをデタッチする

インスタンスを Lightsail ロードバランサーにアタッチする必要がなくなった場合、デタッチすることができます。ロードバランサーから Lightsail インスタンスをデタッチときは、デタッチする前に指定したインスタンスがこれ以上必要なくなるまで待機します。

1. Lightsail のホームページで [ネットワーキング] を選択します。
2. 管理するロードバランサーを選択します。
3. [ターゲットインスタンス] タブで、デタッチするロードバランサーの横にある [デタッチ] を選択します。

Lightsail ロードバランサーを削除する

不要になった場合は、Lightsail ロードバランサーを削除できます。ロードバランサーを削除すると、それにアタッチされている Lightsail インスタンスもすべてデタッチされますが、Lightsail インスタンスは削除されません。SSL/TLS 証明書を使用して暗号化された (HTTPS) トラフィックを有効にした場合、ロードバランサーを削除するとロードバランサーに関連付けられた SSL/TLS 証明書も完全に削除されます。

> **Important**
>
> Lightsail ロードバランサーとそれに関連する証明書を削除すると、元に戻すことができません。

1. Lightsail のホームページで [ネットワーキング] を選択します。

2. 削除するロードバランサーを選択します。
3. [Delete] (削除) をクリックします。
4. [ロードバランサーの削除] を選択します。
5. [Yes, delete] (はい、削除します) を選択します。

Amazon Lightsail コンテンツ配信ネットワークディストリビューション

Lightsail ディストリビューションは、グローバルに分散したサーバーのネットワーク、エッジロケーションを使用して、コンテンツをすばやくユーザーに配信します。ディストリビューションを使用するには、まず 1 つの Lightsail インスタンスまたはコンテナサービス上、または Lightsail ロードバランサーにアタッチされた複数のインスタンス上でウェブサイトかウェブアプリケーションを作成してホストするか、Lightsail バケットに静的コンテンツを保存します。次に、Lightsail ディストリビューションを作成して、インスタンス、コンテナサービス、ロードバランサー、またはバケットからコンテンツをプルし、キャッシュし、提供するように設定します。インスタンス、コンテナサービス、ロードバランサー、またはバケットといったディストリビューションのオリジンは、コンテンツの決定的なソースです。

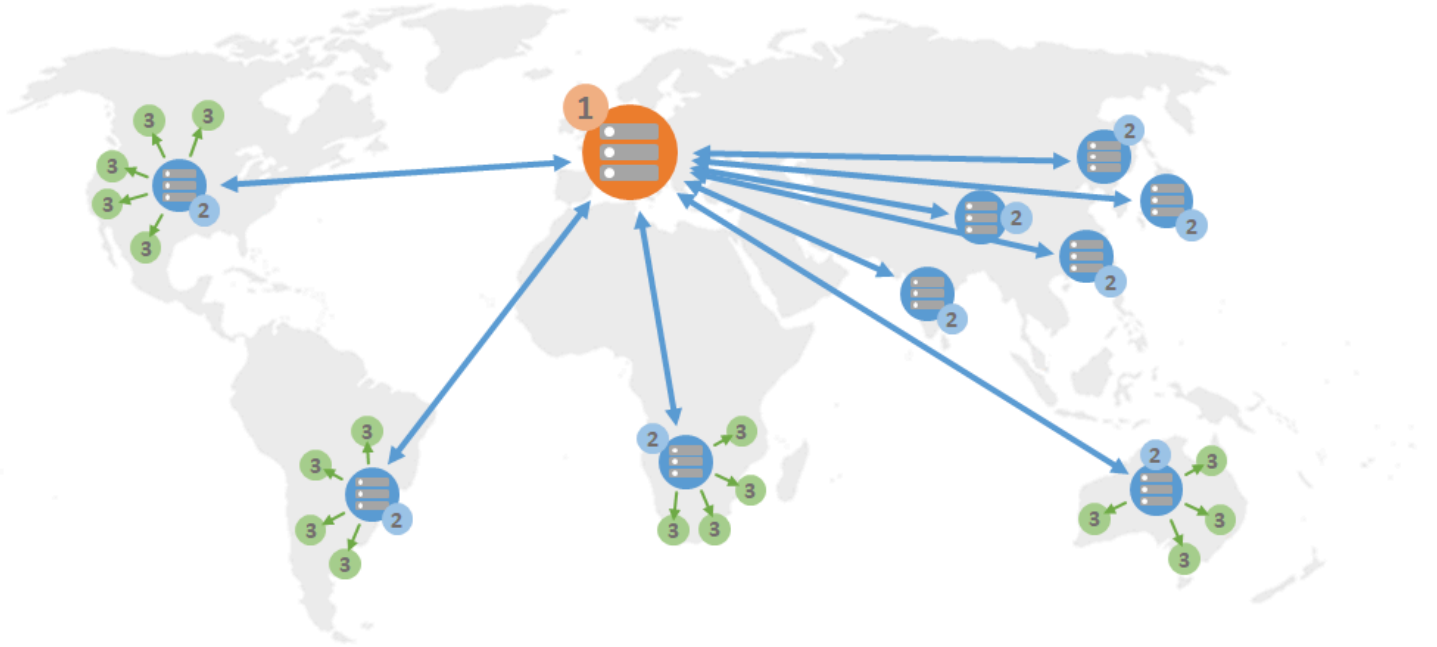
ユーザーがディストリビューションを通じて提供されているウェブサイトにアクセスしてコンテンツをリクエストすると、リクエストはレイテンシーの点から最も近い場所にルーティングされます。次に、ディストリビューションは以下のいずれかのアクションを実行します。

- すでにコンテンツがエッジロケーション内にキャッシュされている場合、ディストリビューションはそのコンテンツをユーザーに提供します。
- コンテンツがそのエッジロケーションにまだキャッシュされていない場合、ディストリビューションは指定されたオリジンからコンテンツを取得し、キャッシュし、ユーザーに提供します。

コンテンツは、ディストリビューションに指定するキャッシュのライフスパン (存続時間) の間、エッジロケーションにキャッシュされるため、同じロケーションにある他のリクエストは直ちに満たされます。キャッシュされたコンテンツは、キャッシュのライフスパンに達すると、エッジロケーションから削除されます。次回、コンテンツリクエストがエッジロケーションにルーティングされる際に、ディストリビューションがコンテンツを取得し、キャッシュ、および配信します。

以下の図表では、

- 1 は、ウェブサイトをホストしている Lightsail インスタンスまたはコンテナサービス、インスタンスがアタッチされているロードバランサー、静的コンテンツをホストしているバケットなど、ディストリビューションのオリジンを示します。
- 2 は、ディストリビューション、またはオリジンからコンテンツをプル、キャッシュし、配信するエッジロケーションを示します。
- 3 は、エッジロケーションからコンテンツを提供するユーザーを示します。

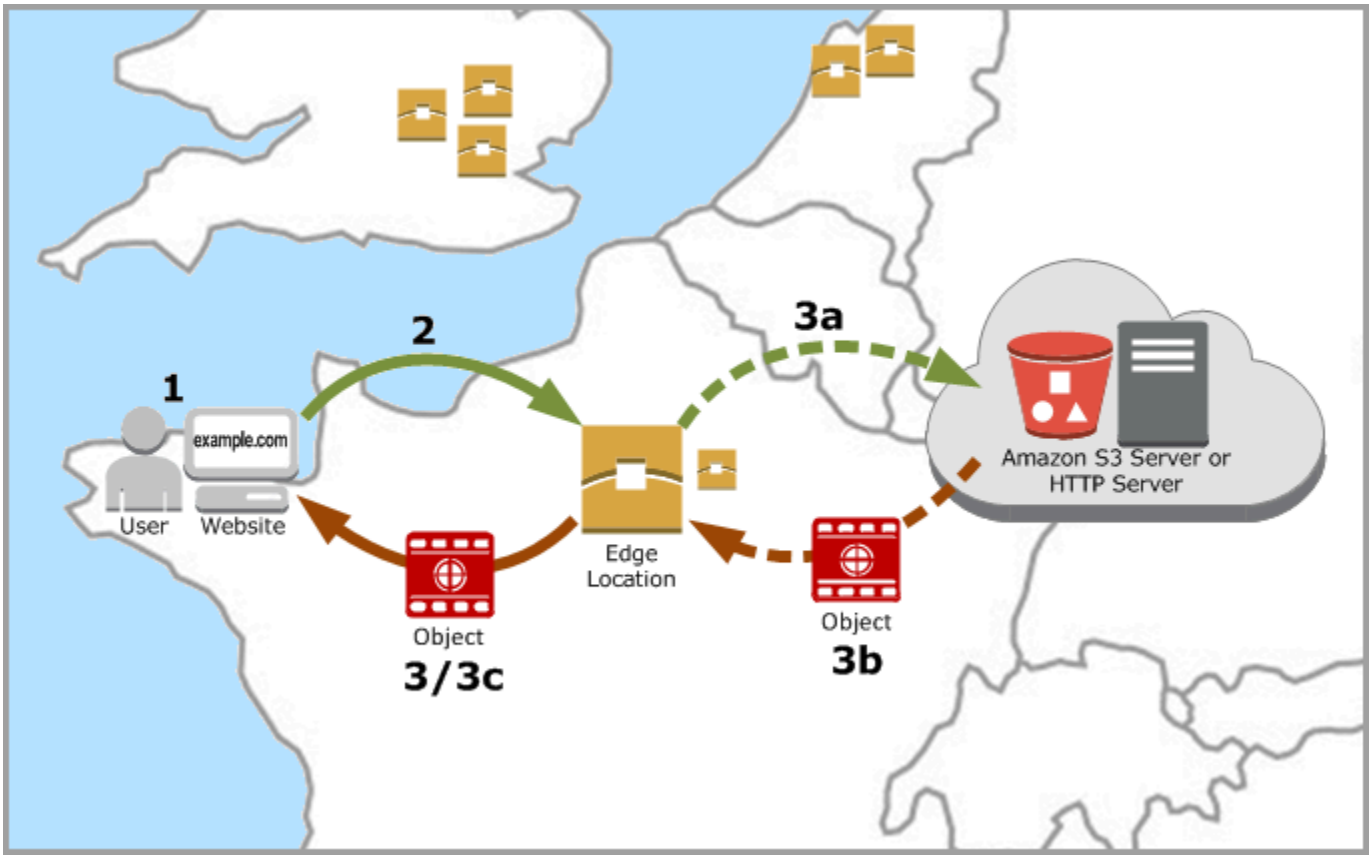


> **Note**
>
> この図表は図示のみを目的とし、実際のエッジロケーションは表示していません。エッジロケーションの詳細については、このガイドの後半のエッジロケーションと IP アドレスの範囲を参照してください。

例えば、ウェブサイトがフランスでホストされており、フランスの他の地域のユーザーがコンテンツを表示したい場合、ページはミリ秒単位の時間でロードされます。

訪問者が近くにいない場合は少し複雑です。

オーストラリアのユーザーがコンテンツを表示したい場合、ブラウザはフランスにあるサーバーからそれを取得し、数千マイル離れた場所にいるそのユーザーに表示する必要があります。異なる国のユーザーが同時に同じコンテンツをリクエストすると、それらのリクエストによってサーバーに負荷がかかり、コンテンツのロードと提供に時間がかかります。これは、エンドユーザーのためにコンテンツがロードされる速度に影響します。



CDN は、エッジロケーションでウェブサイトのコンテンツをキャッシュすることで、この状況を解決します。このコンテンツ提供方法は、1 つの中央リソースからコンテンツを提供する従来の方法よりも高速でかつ効率的です。ビューワーがウェブサイトで、またはアプリケーション経由でリクエストを実行すると、DNS はユーザーのリクエストに対応できる最適なロケーションにリクエストをルーティングします。すべてのユーザーが遠くにある同じ中央リソースにアクセスするのとは対照的に、ユーザーは近くの場所からコンテンツにアクセスします。

ユースケース

高速で安全なウェブサイトを配信する

Lightsail 配信は、世界中のビューワーへのコンテンツ (例えば、ウェブサイトページ、イメージ、スタイルシート、JavaScript など) の配信を高速化します。ディストリビューションを使用することで、AWS バックボーンネットワークおよびエッジサーバーを活用でき、ウェブサイトを閲覧するビューワーに、高速で、安全で、信頼性の高い体験を提供できます。

サイトのセキュリティを改善する

TLS ターミネーションを利用して、ウェブサイトを強化し、パフォーマンスを改善します。これは、暗号化処理をディストリビューションにオフロードすることで、オリジンのロードを軽減します。登録済みドメイン名と Lightsail SSL/TLS 証明書を使用して、ディストリビューションの Hypertext Transfer Protocol Secure (HTTPS) を有効化できます。ユーザーはディストリビューションへの暗号化された HTTPS 接続を確立し、HTTP を使用してオリジンからコンテンツを取得します。

アプリケーションの最適化

WordPress や静的なウェブサイトなど、さまざまなアプリケーション向けにディストリビューションを簡単に最適化できます。ディストリビューションを使用してコンテンツをキャッシュし、配信すると、ほとんどのリクエストがインスタンス、コンテナサービス、ロードバランサー、またはバケットではなくディストリビューションによって処理されるため、オリジンへの負荷も軽減されます。

ディストリビューションを設定する

Lightsail インスタンスとディストリビューションを使用してウェブサイトやウェブアプリケーションを提供するためにの具体的なステップです。

1. ディストリビューションでインスタンス、コンテナサービス、バケットのどれを使用するか応じて、次のいずれかを実行します。

   - コンテンツをホストする Lightsail インスタンスを作成します。インスタンスは、ディストリビューションのオリジンとして機能します。オリジンサーバーには、コンテンツのオリジナルの最終バージョンが保存されます。詳細については、「インスタンスを作成する」を参照してください。

     Lightsail 静的 IP をインスタンスにアタッチします。インスタンスを停止して起動すると、インスタンスのデフォルトのパブリック IP アドレスが変更されるため、ディストリビューションとオリジンインスタンスの接続が切断されます。インスタンスを停止して開始しても、静的 IP は変更されません。詳細については、「静的 IP を作成してインスタンスにアタッチする」を参照してください。

     インスタンスにコンテンツとファイルをアップロードします。ファイル (オブジェクト) には、ウェブページ、イメージ、メディアファイルに限らず、HTTP 経由で提供できるもの全てが含まれます。

- ウェブサイトやウェブアプリケーションをホストする Lightsail コンテナサービスを作成します。コンテナサービスは、ディストリビューションのオリジンとして機能します。オリジンサーバーには、コンテンツのオリジナルの最終バージョンが保存されます。詳細については、「Amazon Lightsail コンテナサービスを作成する」を参照してください。
- Lightsail バケットを作成して、静的コンテンツを保存します。バケットは、ディストリビューションのオリジンとして機能します。オリジンは、オリジナル、最終バージョンコンテンツを保存します。詳細については、「バケットの作成」を参照してください。

  Lightsail コンソール、AWS Command Line Interface (AWS CLI)、AWS API を使用して、ファイルをバケットにアップロードします。ファイルのアップロードに関する詳細については、「バケットにファイルをアップロードする」を参照してください。

2. (オプション) インスタンスでホストされているウェブサイトが耐障害性を必要とする場合、Lightsail ロードバランサーを作成します。次に、インスタンスの複数のコピーをロードバランサーにアタッチします。インスタンスをオリジンとして設定する代わりに、ロードバランサー（複数のインスタンスが添付されている）をディストリビューションのオリジンとして設定することができます。詳細については、「ロードバランサーを作成してインスタンスをアタッチする」を参照してください。
3. Lightsail ディストリビューションを作成し、インスタンス、コンテナサービス、ロードバランサー、またはバケットをオリジンとして設定します。同時に、コンテンツのキャッシュライフスパン、ウェブサイトまたは ウェブアプリケーションのどの要素をキャッシュするかなどの、詳細を指定します。詳細については、「ディストリビューションを作成する」を参照してください。
4. (オプション) ディストリビューションのオリジンが WordPress インスタンスである場合は、インスタンスの WordPress 設定ファイルを編集して、WordPress ウェブサイトをディストリビューションと連携させる必要があります。詳細については、「ディストリビューションと動作するように WordPress インスタンスを設定する」を参照してください。
5. (オプション) Lightsail コンソールでドメインの DNS を管理するために Lightsail DNS ゾーンを作成します。これにより、Lightsail リソースにドメインを簡単にマッピングすることができます。詳細については、「DNS ゾーンを作成してドメインの DNS レコードを管理する」を参照してください。または、現在ホストされているドメインの DNS をホストし続けることもできます。
6. ディストリビューションで使用するために、ドメインの Lightsail SSL/TLS 証明書を作成します。Lightsail ディストリビューションには HTTPS が必要なため、ディストリビューションで使用する前に、ドメインの SSL/TLS 証明書をリクエストする必要があります。詳細については、「ディストリビューションの SSL/TLS 証明書を作成する」を参照してください。
7. ディストリビューションのカスタムドメインを有効化して、登録済みドメイン名をディストリビューションで使用できるようにします。カスタムドメインを有効化するには、ドメイン用に作

成したLightsail SSL/TLS 証明書を指定する必要があります。これにより、ドメインがディストリビューションに追加され、HTTPS が有効になります。詳細については、「ディストリビューション用のカスタムドメインを有効にする」を参照してください。

8. ドメインの DNS にエイリアスレコードを追加して、ドメインのトラフィックをディストリビューションにルーティングします。エイリアスレコードを追加したら、ドメインにアクセスしたユーザーはディストリビューションを通じてルーティングされます。詳細については、「ドメインをディストリビューションにポイントする」を参照してください。
9. ディストリビューションがコンテンツをキャッシュしていることをテストします。詳細については、「ディストリビューションをテストする」を参照してください。

エッジロケーションと IP アドレス範囲

Lightsail ディストリビューションは、Amazon CloudFront と同じエッジサーバーと IP アドレス範囲を使用します。CloudFront エッジサーバーの場所の一覧については、「Amazon CloudFront 製品の詳細情報ページ」をご覧ください。CloudFront の IP 範囲のリストについては、「CloudFront のグローバル IP リスト」を参照してください。

Lightsail コンテンツ配信ネットワークディストリビューションを作成する

このガイドでは、Lightsail コンソールを使用して Amazon Lightsail ディストリビューションを作成する方法と、設定できるディストリビューション設定について説明します。ディストリビューションの詳細については、「コンテンツ配信ネットワークディストリビューション」を参照してください。

目次

前提条件

ディストリビューションの作成のスタート前に、前提条件として次の作業を完了してください。

1. ディストリビューションでインスタンス、コンテナサービス、バケットのどれを使用するか応じて、次のいずれかを実行します。
   - Lightsail インスタンスを作成して、コンテンツをホストします。インスタンスは、ディストリビューションのオリジンとして機能します。オリジンサーバーには、コンテンツのオリジナルの最終バージョンが保存されます。詳細については、「インスタンスを作成する」を参照してください。

     Lightsail 静的 IP をインスタンスにアタッチします。インスタンスを停止して起動すると、インスタンスのデフォルトのパブリック IP アドレスが変更されるため、ディストリビューションとオリジンインスタンスの接続が切断されます。インスタンスを停止して開始しても、静的 IP は変更されません。詳細については、「静的 IP を作成してインスタンスにアタッチする」を参照してください。

     インスタンスにコンテンツとファイルをアップロードします。ファイル (オブジェクト) には、ウェブページ、イメージ、メディアファイルに限らず、HTTP 経由で提供できるもの全てが含まれます。
   - Lightsail コンテナサービスを作成して、ウェブサイトまたはウェブアプリケーションをホストします。コンテナサービスは、ディストリビューションのオリジンとして機能します。オリジンサーバーには、コンテンツのオリジナルの最終バージョンが保存されます。詳細については、「Amazon Lightsail コンテナサービスの作成」を参照してください。
   - Lightsail バケットを作成して、静的コンテンツを保存します。バケットは、ディストリビューションのオリジンとして機能します。オリジンは、オリジナル、最終バージョンコンテンツを保存します。詳細については、「バケットの作成」を参照してください。

     Lightsail コンソール、 AWS Command Line Interface (AWS CLI) 、および AWS APIs。ファイルのアップロードに関する詳細については、「バケットにファイルをアップロードする」を参照してください。
2. (オプション) ウェブサイトで耐障害性が必要な場合は、Lightsail ロードバランサーを作成します。次に、インスタンスの複数のコピーをロードバランサーにアタッチします。インスタンスをオリジンとして設定する代わりに、ロードバランサー (複数のインスタンスが添付されている)

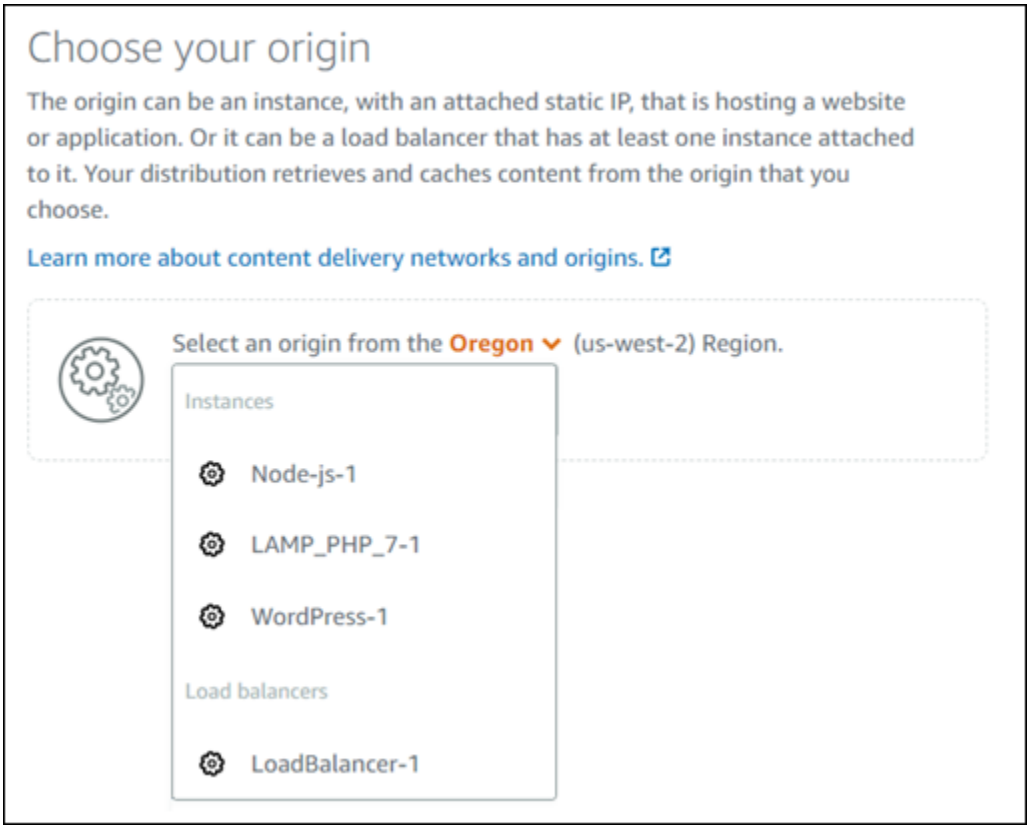
をディストリビューションのオリジンとして設定することができます。詳細については、「ロードバランサーを作成してインスタンスをアタッチする」を参照してください。

オリジンリソース

オリジンとは、あなたのディストリビューションのためのコンテンツの決定的なソースです。ディストリビューションを作成するときは、ウェブサイトまたはウェブアプリケーションのコンテンツをホストする Lightsail インスタンス、コンテナサービス、バケット、またはロードバランサー (1 つ以上のインスタンスがアタッチされている) を選択します。

> **Note**
>
> 現時点では、IPv6-onlyインスタンスを Lightsail コンテンツ配信ネットワーク (CDN) ディストリビューションのオリジンとして設定することはできません。

1 つのディストリビューションにつき 1 つのオリジンのみ選択できます。ディストリビューションを作成後、いつでもオリジンを変更できます。詳細については、「ディストリビューションのオリジンを変更する」を参照してください。



オリジンプロトコルポリシー

オリジンプロトコルポリシーは、オリジンからコンテンツを引き出す時にディストリビューションが使用するプロトコルポリシーです。ディストリビューションのオリジンを選択した後、オリジンからコンテンツを引き出す際に、Hypertext Transfer Protocol (HTTP) か、Hypertext Transfer Protocol Secure (HTTPS) どちらを使用すべきか決めます。オリジンが HTTPS 用に設定されていない場合は、HTTP を使用する必要があります。

ディストリビューションに対して、次のいずれかのオリジンプロトコルポリシーを選択できます。

- HTTP Only - オリジンへのアクセスに HTTP のみを使用します。これはデフォルトの設定です。
- HTTPS Only (HTTPS のみ) : オリジンへのアクセスに HTTPS のみを使用します。

オリジンプロトコルポリシーを編集するステップは、このガイドで後述する「ディストリビューションを作成する」セクションを参照してください。

> **Note**
>
> Lightsail バケットをディストリビューションのオリジンとして選択すると、オリジンプロトコルポリシーはデフォルトで HTTPS のみに設定されます。バケットがディストリビューションのオリジンである場合、オリジンプロトコルポリシーを変更することはできません。

キャッシュ動作とキャッシュプリセット

キャッシュプリセットは、オリジンでホストするコンテンツの種類に応じて、ディストリビューションの設定を自動的に設定します。例えば、静的コンテンツに最適を選択すると、プリセットは、静的ウェブサイトで最適に動作する設定にディストリビューションを自動的に設定します。ウェブサイトが WordPress インスタンスでホストされている場合は、プリセットに最適な WordPress を選択して、ディストリビューションが WordPress ウェブサイトで動作するように自動的に設定されるようにします。

> **Note**
>
> Lightsail バケットをディストリビューションのオリジンとして選択すると、キャッシュプリセットオプションは使用できません。バケットに保存されている静的コンテンツに最適なディストリビューション設定が自動的に適用されます。

ディストリビューション用に、以下のいずれかのキャッシュプリセットを選択できます。

- 静的コンテンツに最適 - このプリセットは、ディストリビューションをすべてをキャッシュするように設定されます。このプリセットは、オリジンで静的コンテンツ (静的 HTML ページなど) をホストする場合や、ウェブサイトにアクセスするユーザーごとに変更されないコンテンツをホストする場合に最適です。このプリセットを選択すると、ディストリビューション上のすべてのコンテンツがキャッシュされます。
- 動的コンテンツに最適 - このプリセットは、ディストリビューションをディストリビューションを作成するページのセクションにあるディレクトリとファイルの上書きのキャッシュで指定されていないもの以外をキャッシュしないように設定されます。詳細については、ディレクトリとファイルの上書きを後ほど参照してください。このプリセットは、オリジンで動的なコンテンツや、ウェブサイトやウェブアプリケーションにアクセスするユーザーごとに変化するコンテンツをホストする場合に最適です。
- 最適な WordPress- このプリセットは、インスタンスの WordPress `wp-includes/`および `wp-content/` ディレクトリ内のファイル以外をキャッシュしないようにディストリビューションを設定します。このプリセットは、オリジンが、WordPress Certified by Bitnami および Automattic ブループリント (マルチサイトブループリントを除く) を使用するインスタンスである場合に最適です。このプリセットの詳細については、 WordPress 「プリセットのキャッシュに最適」を参照してください。

> **Note**
>
> カスタム設定プリセットは選択できません。プリセットはプリセットを選択後、自動で選択されますが、ディストリビューションの設定を手動で変更します。

キャッシュプリセットは Lightsail コンソールでのみ指定できます。Lightsail API、、AWS CLIおよび SDKs を使用して指定することはできません。

WordPress キャッシュプリセットに最適

ディストリビューションのオリジンとして、WordPress Certified by Bitnami と Automattic ブループリントを使用するインスタンスを選択すると、Lightsail はディストリビューションに Best for WordPress caching プリセットを適用するかどうかを尋ねます。現在の を適用すると、ディストリビューションは WordPress ウェブサイトで最適に動作するように自動的に設定されます。他に適用しなければいけないディストリビューション設定はありません。 WordPress ウェブサイトの `wp-includes/`および `wp-content/` ディレクトリにあるファイル以外は、プリセットに最適な

キャッシュなし。 WordPressまた、毎日キャッシュをクリアするようにディストリビューションを設定し（キャッシュ寿命は 1 日）、すべての HTTP メソッドを許可し、Host ヘッダーのみを転送し、Cookieを転送せず、すべてのクエリ文字列を転送します。

> **Important**
>
> ウェブサイトをディストリビューションと連携させる WordPressには、インスタンスの設定 WordPress ファイルを編集する必要があります。詳細については、「ディストリビューション と連携するように WordPress インスタンスを設定する」を参照してください。

デフォルトの動作

デフォルトの動作は、ディストリビューションがコンテンツキャッシュをどのように処理するかを指定します。ディストリビューションのデフォルトの動作は、選択したキャッシュプリセットによって自動的に決定されます。別のデフォルト動作を選択した場合、キャッシュプリセットは自動的にカスタム設定にされます。

> **Note**
>
> Lightsail バケットをディストリビューションのオリジンとして選択した場合、デフォルトの動作オプションは使用できません。バケットに保存されている静的コンテンツに最適ディストリビューション設定が自動的に適用されます。

ディストリビューションでは、以下のいずれかのデフォルト動作を選択できます。

- すべてをキャッシュする - この動作は、ウェブサイト全てを静的コンテンツとしてキャッシュ、対応するようにディストリビューションを設定します。このオプションは、閲覧者によって変更されないコンテンツをオリジンがホストしている場合、または ウェブサイトが cookie、ヘッダー、またはクエリ文字列を使用してコンテンツをパーソナライズしない場合に最適です。
- 何もキャッシュしない - この動作は、指定したオリジンファイルとフォルダーパスのみをキャッシュするようにディストリビューションを設定します。このオプションは、ウェブサイトやウェブアプリケーションが cookie、ヘッダー、クエリ文字列を使用して、個々のユーザー向けにコンテンツをパーソナライズする場合に最適です。このオプションを選択すると、キャッシュするには、ディレクトリとファイルパスの上書きを指定する必要があります。

ディレクトリとファイルの上書き

ディレクトリとファイルの上書きを使用して、選択したデフォルトの動作を上書きしたり、例外を追加することが可能です。例えば、すべてをキャッシュするを選択した場合、上書きを使用して、ディストリビューションがキャッシュしないディレクトリ、ファイル、またはファイルの種類を指定します。代わりに、何もキャッシュしないを選択した場合、上書きを使用して、ディストリビューションがキャッシュするディレクトリ、ファイル、またはファイルの種類を指定します。

ディレクトリとファイルの上書きセクションで、キャッシュするディレクトリまたはファイルへのパスを指定するか、キャッシュしないかを指定できます。アスタリスク記号を使用して、ワイルドカードディレクトリ (path/to/assets/*)、ファイルタイプ (*.html,*jpg,*js)を指定する。ディレクトリとファイルのパスでは、大文字と小文字が区別されます。

> **Note**
>
> ディストリビューションのオリジンとして Lightsail バケットを選択すると、ディレクトリとファイルの上書きオプションは使用できません。選択したバケットに保存されているものすべてがキャッシュされます。

以下は、ディレクトリとファイルの上書きを指定する方法の例です。

- Lightsail インスタンスで実行されている Apache ウェブサーバーのドキュメントルート内のすべてのファイルをキャッシュするには、以下を指定します。

  ```
  var/www/html/
  ```

- Apache ウェブサーバーのドキュメントルートのインデックスページのみをキャッシュするには、次のファイルを指定します。

  ```
  var/www/html/index.html
  ```

- Apache ウェブサーバーのドキュメントルートの .html ファイルのみをキャッシュするには、次のように指定します。

  ```
  var/www/html/*.html
  ```

- 以下を指定して、 Apache ウェブサーバーのドキュメントルートにある images サブディレクトリの .jpg、.png、および.gif ファイルのみをキャッシュします。

```
var/www/html/images/*.jpg
```

```
var/www/html/images/*.png
```

```
var/www/html/images/*.gif
```

以下を指定して、 Apache ウェブサーバーのドキュメントルートにある images サブディレクトリのすべてのファイルをキャッシュするします。

```
var/www/html/images/
```

キャッシュの詳細設定

詳細設定を使用して、ディストリビューション上のコンテンツのキャッシュのライフスパン、許可されている HTTP メソッド、 HTTP ヘッダー転送、 cookie 転送、およびクエリ文字列転送を指定できます。指定したアドバンスト設定は、ディストリビューションがキャッシュするディレクトリとファイルにのみ適用されます。これには、キャッシュとして指定したディレクトリとファイルの上書きも含まれます。

> **Note**
>
> Lightsail バケットをディストリビューションのオリジンとして選択すると、高度なキャッシュ設定はディストリビューションの作成ページでは使用できません。バケットに保存される静的コンテンツに最適なディストリビューション設定が自動的に適用されます。ただし、ディストリビューションの作成後に、ディストリビューション管理ページでアドバンストキャッシュ設定を変更できます。

次のアドバンスト設定を編集できます。

キャッシュ寿命 (TTL)

コンテンツが更新されたかどうかを確認するためにディストリビューションからオリジンに別のリクエストを送るまで、コンテンツをディストリビューションのキャッシュに保持する期間を制御します。デフォルト値は 1 日です。この期間を短くすると、動的なコンテンツを供給できます。この期間を長くすると、ユーザー側のパフォーマンスは向上します。ファイルがエッジロケーションから直

接返される可能性が高くなるためです。期間を長くすると、ディストリビューションがコンテンツを引き出す頻度が低くなるため、オリジンの負荷も軽減されます。

> **Note**
>
> 指定するキャッシュのライフスパン値は、オリジンが `Cache-Control max-age`、`Cache-Control s-maxage`、`Expires` などの HTTP ヘッダーをコンテンツに追加しないときにのみ適用されます。

許可される HTTP メソッド

ディストリビューションが処理してオリジンに転送する HTTP メソッドをコントロールします。HTTP メソッドは、オリジンで実行されるべきパフォーマンスを示します。例えば、GET メソッドはオリジンからデータを取得し、PUT メソッドは、囲まれたエンティティをオリジンに保存することを要求します。

以下のディストリビューションの HTTP メソッドのオプションのいずれかを選択できます。

- 許可された GET、 HEAD、 OPTIONS、 PUT,、PATCH、 POST と DELETE メソッド
- GET、HEAD、OPTIONS メソッドを許可する
- GET と HEAD メソッドを許可する

ディストリビューションは、常に GET および HEAD の応答をキャッシュします。OPTIONS リクエストを許可するように選択した場合、ディストリビューションは OPTIONS の応答もキャッシュします。ディストリビューションは他の HTTP メソッドへのレスポンスをキャッシュしません。詳細については、「HTTP メソッド」を参照してください。

> **Important**
>
> サポートされているすべての HTTP メソッドを許可するようにディストリビューションを構成する場合、オリジンインスタンスにすべてのメソッドを処理させるように設定する必要があります。例えば POST を使用したいので、上記のメソッドを受け入れて転送するようにディストリビューションを構成する場合は、削除すべきでないリソースをビューワーが削除できないようにするために、DELETE リクエストを適切に処理するようオリジンサーバーを構成する必要があります。詳細については、ウェブサイトまたはウェブアプリケーションのドキュメントを検索してください。

HTTP ヘッダーの転送

ディストリビューションが、指定されたヘッダーの値に基づいてコンテンツをキャッシュするか否か、そしてどのヘッダーに基づくのかをコントロールします。HTTP ヘッダーは、クライアントブラウザ、要求されたページ、オリジンなどの情報を保持します。たとえば、Accept-Languageヘッダーはクライアントの言語を送信します (例えば、英語なら en-US) 。これにより、オリジンはクライアントの言語でコンテンツに対応できます (利用可能な場合) 。

ディストリビューションでは、次の HTTP ヘッダーのオプションのいずれかを選択できます。

- ヘッダーを転送しない
- 指定するヘッダーのみを転送する

ヘッダーを転送しないを選択すると、ディストリビューションはヘッダー値に基づいたコンテンツのキャッシュを行いません。選択したオプションにかかわらず、ディストリビューションは特定のヘッダーをオリジンに転送し、転送したヘッダーに基づいて特定のアクションを実行します。ディストリビューションがヘッダーの転送を処理する方法の詳細については、「HTTP リクエストヘッダーとディストリビューション動作」を参照してください。

Cookie の転送

ディストリビューションがオリジンに Cookie を転送するかどうか、および転送する場合はどれを転送するかをコントロールします。Cookieには、オリジンの ウェブページでの訪問者の行動に関する情報や、訪問者が提供した名前や関心事などの情報など、オリジンに送信される小さなデータが含まれます。

ディストリビューションでは、以下の Cookie 転送オプションのいずれかを選択できます。

- cookie を転送しない
- すべての Cookie を転送する
- 指定した Cookie を転送する

すべての cookie を転送するを選択した場合、ディストリビューションは、アプリケーションで使用されている Cookie の数に関係なく、すべての Cookie を転送します。指定した Cookie を転送するを選択した場合、ディストリビューションに転送して欲しい cookies 名を表示されるテキストボックスに入力します。以下のワイルドカード文字を使用して Cookie 名を指定することができます。

- * は、Cookie 名に含まれる 0 個以上の文字と一致します。

- ? は、Cookie 名に含まれる 1 文字と一致します。

例えば、オブジェクトに対するビューワーリクエストに `userid_`*`member-number`*Cookie 名が含まれているとします。各ユーザーに割り当てられた一意の値 `member-number` (`userid_123`,`userid_124`,`userid_125` など) 。ディストリビューションが、各メンバーについて個別バージョンのコンテンツでキャッシュするものとします。これは Cookie をオリジンに転送することで実行できますが、ビューワーのリクエストには、 ディストリビューションにキャッシュして欲しくない cookies が含まれます。これに代わる方法として、Cookie 名に以下の値を指定できます。その場合、ディストリビューションは `userid_` から始まるすべての Cookie をオリジン `userid_*` に転送します。

クエリ文字列の転送

ディストリビューションがオリジンにクエリ文字列を転送するかどうか、および転送する場合にどれを転送するかをコントロールします。クエリ文字列は、指定されたパラメータに値を割り当てる URL の一部です。例えば、`https://example.com/over/there?name=ferret` URL は `name=ferret` クエリ文字列を含みます。サーバーは、そのようなページのリクエストを受信すると、プログラムを実行し、`name=ferret` クエリ文字列を変更せずにプログラムに渡します。疑問符はセパレータとして使用され、クエリ文字列の一部ではありません。

ディストリビューションがクエリ文字列を転送しないようにするか、指定したクエリ文字列のみを転送するかを選択できます。オリジンがクエリ文字列パラメータの値に関係なくコンテンツの同じバージョンを返す場合、クエリ文字列を転送しないように選択します。これにより、 ディストリビューションがキャッシュからリクエストを処理できる可能性が高くなり、パフォーマンスが向上し、オリジンの負荷が軽減されます。オリジンサーバーが 1 つ以上のクエリ文字列パラメータに基づいてコンテンツの異なるバージョンを返す場合、選択したクエリ文字列のみを転送します。

ディストリビューションプラン

ディストリビューションプランは、毎月のデータ転送クォータとディストリビューションのコストを指定します。プランの月次データ転送クォータよりも多くのデータが配信される場合、超過分が課金されます。詳細については、Lightsail の料金 ページを参照してください。

超過料金が発生しないようにするには、クォータを超える前に、現在のディストリビューションのプランを毎月のデータ転送量が多い別のプランに変更します。ディストリビューションのプランは、各 AWS 請求サイクルで 1 回だけ変更できます。作成後にディストリビューションプランを変更する方法の詳細については、「ディストリビューションのプランを変更する」を参照してください。

ディストリビューションを作成する

ディストリビューションを作成する手順は以下のとおりです。

1. Lightsail コンソールにサインインします。
2. lightsail のホームページで [Networking] (ネットワーク) タブを選択します。
3. [ディストリビューションの作成] を選択します。
4. ページの [オリジンを選択する] セクションで、オリジンリソースが作成された AWS リージョンを選択します。

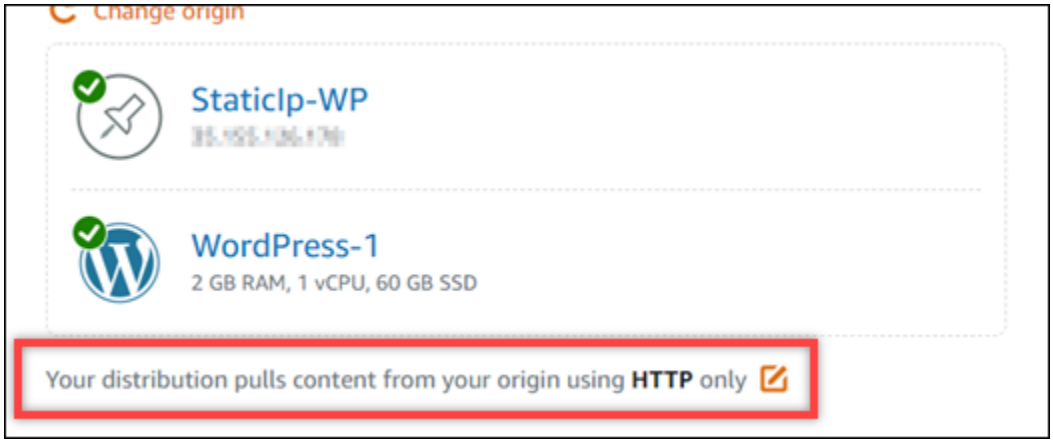
   ディストリビューションはグローバルリソースです。どの AWS リージョン でもオリジンをリファレンスでき、グローバルにそのコンテンツを配信することができます。
5. オリジンの選択。オリジンは、Lightsail インスタンス、コンテナサービス、バケット、またはロードバランサー (1 つ以上のインスタンスがアタッチされている) です。詳細については、オリジンリソースを参照してください。

   > **Important**
   >
   > ディストリビューションのオリジンとして Lightsail コンテナサービスを選択すると、Lightsail はディストリビューションのデフォルトドメイン名をコンテナサービスのカスタムドメインとして自動的に追加します。これにより、ディストリビューションとコンテナサービスの間でトラフィックをルーティングできます。ただし、場合によっては、ディストリビューションのデフォルトドメイン名をコンテナサービスに手動で追加する必要があります。詳細については、「ディストリビューションのデフォルトドメインをコンテナサービスに追加する」を参照してください。
6. (オプション) オリジンプロトコルポリシーを変更するには、ディストリビューションが使用する現在のオリジンプロトコルポリシーの横に表示される鉛筆アイコンを選択します。詳細については、オリジンプロトコルポリシーを参照してください。

   このオプションは、オリジンの選択セクションにあり、選択したディストリビューションのオリジンリソース下にあります。

   > **Note**
   >
   > Lightsail バケットをディストリビューションのオリジンとして選択すると、オリジンプロトコルポリシーはデフォルトで HTTPS のみに設定されます。バケットがディストリ

ビューションのオリジンである場合、オリジンプロトコルポリシーを変更することはできません。



7. ディストリビューションのキャッシュ動作 (キャッシングプリセットとも呼ばれます) を選択します。詳細については、「キャッシュ動作とキャッシングプリセット」を参照してください。

> **Note**
>
> Lightsail バケットをディストリビューションのオリジンとして選択すると、キャッシュプリセットオプションは使用できません。バケットに保存されている静的コンテンツに最適なディストリビューション設定が自動的に適用されます。

8. (オプション) [すべての設定を表示] を選択して、ディストリビューションの追加のキャッシュ動作設定を表示させます。

> **Note**
>
> Lightsail バケットをディストリビューションのオリジンとして選択すると、キャッシュ動作設定は使用できません。バケットに保存されている静的コンテンツに最適なディストリビューション設定が自動的に適用されます。

9. (オプション) ディストリビューションのデフォルトの動作を選択します。詳細については、デフォルト動作を参照してください。

> **Note**
>
> Lightsail バケットをディストリビューションのオリジンとして選択した場合、デフォルトの動作オプションは使用できません。バケットに保存されている静的コンテンツに最適なディストリビューション設定が自動的に適用されます。

10. (オプション) [パスの追加] を選択して、ディストリビューションのキャッシュ動作を上書きするディレクトリとファイルを追加します。詳細については、「ディレクトリとファイルの上書き」を参照してください。

> **Note**
>
> ディストリビューションのオリジンとして Lightsail バケットを選択すると、ディレクトリとファイルの上書きオプションは使用できません。バケットに保存されている静的コンテンツに最適なディストリビューション設定が自動的に適用されます。

11. (オプション) 編集するディストリビューションの横に表示されるアドバンスト設定用の鉛筆アイコンを選択します。詳細については、「アドバンストキャッシュ動作設定」を参照してください。

> **Note**
>
> Lightsail バケットをディストリビューションのオリジンとして選択すると、高度なキャッシュ設定はディストリビューションの作成ページでは使用できません。バケットに保存される静的コンテンツに最適なディストリビューション設定が自動的に適用されます。ただし、ディストリビューションの作成後に、ディストリビューション管理ページでアドバンストキャッシュ設定を変更できます。

12. ディストリビューションプランを選択します。詳細については、「ディストリビューションプラン」を参照してください。
13. ディストリビューションの名前を入力します。

    リソース名:

    - AWS リージョン Lightsail アカウントの各 内で一意である必要があります。
    - 2〜255 文字を使用する必要があります。
    - 先頭と末尾は英数字または数字を使用する必要があります。

- 英数字、数字、ピリオド、ダッシュ、アンダースコアを使用することができます。

14. ディストリビューションのコストを確認します。
15. [ディストリビューションの作成] を選択します。

    しばらくすると、ディストリビューションが作成されます。

次のステップ

ディストリビューションを起動して実行したら、次の手順を実行することをお勧めします。

1. ディストリビューションのオリジンが WordPress インスタンスの場合は、インスタンスの設定ファイルを編集して、 WordPress ウェブサイトを WordPressディストリビューションと連携させる必要があります。詳細については、「ディストリビューション と連携するように WordPress インスタンスを設定する」を参照してください。
2. (オプション) Lightsail コンソールでドメインの DNS を管理する Lightsail DNS ゾーンを作成します。これにより、ドメインを Lightsail リソースに簡単にマッピングできます。詳細については、「DNS ゾーンを作成してドメインの DNS レコードを管理する」を参照してください。または、現在ホストされているドメインの DNS をホストし続けることもできます。
3. ディストリビューションで使用するドメインの Lightsail SSL/TLS 証明書を作成します。Lightsail ディストリビューションには HTTPS が必要なため、ディストリビューションで使用する前に、ドメインの SSL/TLS 証明書をリクエストする必要があります。詳細については、「ディストリビューションの SSL/TLS 証明書を作成する」を参照してください。
4. ディストリビューションでカスタムドメインを有効にして、ディストリビューションでドメインを使用できるようにします。カスタムドメインを有効にするには、ドメイン用に作成した Lightsail SSL/TLS 証明書を指定する必要があります。これにより、ドメインがディストリビューションに追加され、HTTPS が有効になります。詳細については、「ディストリビューション用のカスタムドメインを有効にする」を参照してください。
5. ドメインの DNS にエイリアスレコードを追加して、ドメインのトラフィックをディストリビューションにルーティングします。エイリアスレコードを追加したら、ドメインにアクセスしたユーザーはディストリビューションを通じてルーティングされます。詳細については、「ドメインをディストリビューションにポイントする」を参照してください。
6. ディストリビューションがコンテンツをキャッシュしていることをテストします。詳細については、「ディストリビューションをテストする」を参照してください。

Lightsail ディストリビューションを削除する

使用していない Amazon Lightsail ディストリビューションはいつでも削除できます。

ディストリビューションを削除する

ディストリビューションを削除するためには、次の手順を実行します。

1. Lightsail コンソールにサインインします。
2. Lightsail のホームページで、[ネットワーキング] タブを選択します。
3. 削除するディストリビューションの名前を選択します。
4. ディストリビューション管理ページで [Delete] (削除) タブを選択します。
5. ディストリビューションを削除するためには、[ディストリビューションの削除] を選択します。
6. [はい、削除します] を選択して削除を確定します。

Lightsail ディストリビューションのキャッシュ動作を変更する。

キャッシュ動作は Amazon Lightsail ディストリビューションによってキャッシュされているもの、されていないもののオリジンを設定可能にします。例えば、オリジンから個々のディレクトリ、ファイル、またはファイルタイプをキャッシュするように指定できます。オリジンに転送される HTML メソッドとヘッダーを指定することもできます。このガイドでは、ディストリビューションのキャッシュ動作を変更する方法について説明します。ディストリビューションの詳細については、「コンテンツ配信ネットワークディストリビューション」を参照してください。

目次

キャッシュプリセット

キャッシュプリセットは、オリジンでホストするコンテンツの種類に応じて、ディストリビューションの設定を自動的に設定します。例えば、静的コンテンツに最適を選択すると、プリセットは、静的ウェブサイトで最適に動作する設定にディストリビューションを自動的に設定します。ウェブサイトが WordPress インスタンスでホストされている場合は、 WordPress に最適プリセットを使用して、WordPress ウェブサイトで最適に動作するようにディストリビューションが自動的に設定されます。

ディストリビューションでは、以下のいずれかのキャッシュプリセットを選択できます。

- 静的コンテンツに最適 - このプリセットは、ディストリビューションをすべてをキャッシュするように設定されます。このプリセットは、オリジンで静的コンテンツ (静的 HTML ページなど) をホストする場合や、ウェブサイトにアクセスするユーザーごとに変更されないコンテンツをホストする場合に最適です。このプリセットを選択すると、ディストリビューション上のすべてのコンテンツがキャッシュされます。
- 動的コンテンツに最適 - このプリセットは、ディストリビューションをディストリビューションを作成するページのセクションにあるディレクトリとファイルの上書きのキャッシュで指定されていないもの以外をキャッシュしないように設定されます。詳細については、ディレクトリとファイルの上書きを後ほど参照してください。このプリセットは、オリジンで動的なコンテンツや、ウェブサイトやウェブアプリケーションにアクセスするユーザーごとに変化するコンテンツをホストする場合に最適です。
- WordPress に最適 - このプリセットは、ディストリビューションを`wp-includes/`と`wp-content/` WordPress インスタンスのディレクトリーにあるファイル以外をキャッシュしないように設定されます。このプリセットは、オリジンのインスタンスが WordPress Certified by Bitnami and Automatic ブループリントの場合に最適です (マルチサイトブループリントを除く) 。このプリセットの詳細については、 WordPress の最適キャッシュプリセット

> **Note**
>
> カスタム設定プリセットは選択できません。プリセットはプリセットを選択後、自動で選択されますが、ディストリビューションの設定を手動で変更します。

Lightsail コンソールのみでキャッシュのプリセットを指定できます。これは、 Lightsail API、AWS CLI、および SDK を使用して指定することはできません。

WordPress の最適キャッシュプリセット

選択したインスタンスが、 WordPress Certified by Bitnami and Automattic 設計図をディストリビューションのオリジンとして使用している場合、Lightsail はディストリビューションのキャッシュプリセットに [WordPress に最適] を適用するかを確認するプロンプトが表示されます。プリセットを適用すると、ディストリビューションは WordPress のウェブサイトに最適した動作に自動的に設定されます。他に適用しなければいけないディストリビューション設定はありません。WordPress に最適は、`wp-includes/`内のファイルおよび`wp-content/` WordPress ウェブサイトのディレクトリ以外をキャッシュしないようにプリセットされます。また、毎日キャッシュをクリアするようにディストリビューションを設定し (キャッシュ寿命は 1 日) 、すべての HTTP メソッドを許可し、`Host` ヘッダーのみを転送し、Cookieを転送せず、すべてのクエリ文字列を転送します。

> **Important**
>
> WordPress ウェブサイトをディストリビューションで使用できるようにするには、インスタンスの WordPress 設定ファイルを編集する必要があります。詳細については、「ディストリビューションと動作するように WordPress インスタンスを設定する」を参照してください。

デフォルトの動作

デフォルトの動作は、ディストリビューションがコンテンツキャッシュをどのように処理するかを指定します。ディストリビューションのデフォルトの動作は、選択したキャッシュプリセットによって自動的に決定されます。別のデフォルト動作を選択した場合、キャッシュプリセットは自動的にカスタム設定にされます。

ディストリビューションは、以下のデフォルトの動作のいずれかから選択できます。

- すべてをキャッシュする - この動作は、ウェブサイト全てを静的コンテンツとしてキャッシュ、対応するようにディストリビューションを設定します。このオプションは、閲覧者によって変更されないコンテンツをオリジンがホストしている場合、または ウェブサイトが cookie、ヘッダー、またはクエリ文字列を使用してコンテンツをパーソナライズしない場合に最適です。
- 何もキャッシュしない - この動作は、指定したオリジンファイルとフォルダーパスのみをキャッシュするようにディストリビューションを設定します。このオプションは、ウェブサイトやウェブアプリケーションが cookie、ヘッダー、クエリ文字列を使用して、個々のユーザー向けにコンテンツをパーソナライズする場合に最適です。このオプションを選択すると、キャッシュするには、ディレクトリとファイルパスの上書きを指定する必要があります。

ディレクトリとファイルの上書き

ディレクトリとファイルの上書きを使用して、選択したデフォルトの動作を上書きしたり、例外を追加することが可能です。例えば、すべてをキャッシュするを選択した場合、上書きを使用して、ディストリビューションがキャッシュしないディレクトリ、ファイル、またはファイルの種類を指定します。代わりに、何もキャッシュしないを選択した場合、上書きを使用して、ディストリビューションがキャッシュするディレクトリ、ファイル、またはファイルの種類を指定します。

ディレクトリとファイルの上書きセクションで、キャッシュするディレクトリまたはファイルへのパスを指定するか、キャッシュしないかを指定できます。アスタリスク記号を使用して、ワイルドカードディレクトリ (`path/to/assets/*`)、ファイルタイプ (`*.html`,`*jpg`,`*js`)を指定する。ディレクトリとファイルのパスでは、大文字と小文字が区別されます。

ディレクトリとファイルの上書きを指定する方法の例をいくつか紹介します。

- 以下を指定して、Lightsail インスタンスで動作している Apache ウェブサーバーのドキュメントルート内のすべてのファイルをキャッシュします。

  ```
  var/www/html/
  ```

- 以下を指定して、 Apache ウェブサーバーのドキュメントルートにあるインデックスページのみをキャッシュします。

  ```
  var/www/html/index.html
  ```

- 以下を指定して、 Apache ウェブサーバーのドキュメントルートにある. html ファイルのみをキャッシュします。

  ```
  var/www/html/*.html
  ```

- 以下を指定して、 Apache ウェブサーバーのドキュメントルートにある images サブディレクトリの .jpg、.png、および.gif ファイルのみをキャッシュします。

  ```
  var/www/html/images/*.jpg
  ```

  ```
  var/www/html/images/*.png
  ```

  ```
  var/www/html/images/*.gif
  ```

以下を指定して、 Apache ウェブサーバーのドキュメントルートにある images サブディレクトリのすべてのファイルをキャッシュするします。

```
var/www/html/images/
```

キャッシュの詳細設定

詳細設定を使用して、ディストリビューション上のコンテンツのキャッシュのライフスパン、許可されている HTTP メソッド、 HTTP ヘッダー転送、 cookie 転送、およびクエリ文字列転送を指定できます。指定したアドバンスト設定は、ディストリビューションがキャッシュするディレクトリとファイルにのみ適用されます。これには、キャッシュとして指定したディレクトリとファイルの上書きも含まれます。

次のアドバンスト設定を編集できます。

キャッシュ寿命 (TTL)

コンテンツが更新されたかどうかを確認するためにディストリビューションからオリジンに別のリクエストを送るまで、コンテンツをディストリビューションのキャッシュに保持する期間を制御します。デフォルト値は 1 日です。この期間を短くすると、動的なコンテンツを供給できます。この期間を長くすると、ユーザー側のパフォーマンスは向上します。ファイルがエッジロケーションから直接返される可能性が高くなるためです。期間を長くすると、ディストリビューションがコンテンツを引き出す頻度が低くなるため、オリジンの負荷も軽減されます。

> **Note**
>
> 指定するキャッシュのライフスパン値は、オリジンが `Cache-Control max-age`、`Cache-Control s-maxage`、`Expires` などの HTTP ヘッダーをコンテンツに追加しないときにのみ適用されます。

許可される HTTP メソッド

ディストリビューションが処理してオリジンに転送する HTTP メソッドをコントロールします。HTTP メソッドは、オリジンで実行されるべきパフォーマンスを示します。例えば、GET メソッドはオリジンからデータを取得し、PUT メソッドは、囲まれたエンティティをオリジンに保存することを要求します。

以下のディストリビューションの HTTP メソッドのオプションのいずれかを選択できます。

- 許可された GET、 HEAD、 OPTIONS、 PUT,、PATCH、 POST と DELETE メソッド
- GET、HEAD、OPTIONS メソッドを許可する
- GET と HEAD メソッドを許可する

ディストリビューションは、常に GET および HEAD の応答をキャッシュします。OPTIONS リクエストを許可するように選択した場合、ディストリビューションは OPTIONS の応答もキャッシュします。ディストリビューションは他の HTTP メソッドの応答をキャッシュしません。

> **Important**
>
> サポートされているすべての HTTP メソッドを許可するようにディストリビューションを設定した場合、オリジンインスタンスがすべてのメソッドを処理できるように設定する必要があります。例えば POST を使用したいので、上記のメソッドを受け入れて転送するようにディストリビューションを構成する場合は、削除すべきでないリソースをビューワーが削除できないようにするために、DELETE リクエストを適切に処理するようオリジンサーバーを構成する必要があります。詳細については、ウェブサイトまたはウェブアプリケーションのドキュメントを検索してください。

HTTP ヘッダーの転送

ディストリビューションが、指定されたヘッダーの値に基づいてコンテンツをキャッシュするか否か、そしてどのヘッダーに基づくのかをコントロールします。HTTP ヘッダーは、クライアントブラウザ、要求されたページ、オリジンなどの情報を保持します。たとえば、Accept-Languageヘッダーはクライアントの言語を送信します (例えば、英語なら en-US) 。これにより、オリジンはクライアントの言語でコンテンツに対応できます (利用可能な場合) 。

ディストリビューションでは、次の HTTP ヘッダーのオプションのいずれかを選択できます。

- ヘッダーを転送しない
- 指定するヘッダーのみを転送する

ヘッダーを転送しないを選択すると、ディストリビューションはヘッダー値に基づいたコンテンツのキャッシュを行いません。選択したオプションにかかわらず、ディストリビューションは特定のヘッダーをオリジンに転送し、転送したヘッダーに基づいて特定のアクションを実行します。

Cookie の転送

ディストリビューションがオリジンに Cookie を転送するかどうか、および転送する場合はどれを転送するかをコントロールします。Cookieには、オリジンの ウェブページでの訪問者の行動に関する情報や、訪問者が提供した名前や関心事などの情報など、オリジンに送信される小さなデータが含まれます。

ディストリビューションでは、以下の Cookie 転送オプションのいずれかを選択できます。

- cookie を転送しない
- すべての Cookie を転送する
- 指定した Cookie を転送する

すべての cookie を転送するを選択した場合、ディストリビューションは、アプリケーションで使用されている Cookie の数に関係なく、すべての Cookie を転送します。指定した Cookie を転送するを選択した場合、ディストリビューションに転送して欲しい cookies 名を表示されるテキストボックスに入力します。以下のワイルドカード文字を使用して Cookie 名を指定することができます。

- * は、Cookie 名に含まれる 0 個以上の文字と一致します。
- ? は、Cookie 名に含まれる 1 文字と一致します。

例えば、オブジェクトに対するビューワーリクエストに `userid_`*`member-number`*Cookie 名が含まれているとします。各ユーザーに割り当てられた一意の値 `member-number` (`userid_123`,`userid_124`,`userid_125` など) 。ディストリビューションが、各メンバーについて個別バージョンのコンテンツでキャッシュするものとします。これは Cookie をオリジンに転送することで実行できますが、ビューワーのリクエストには、 ディストリビューションにキャッシュして欲しくない cookies が含まれます。これに代わる方法として、Cookie 名に以下の値を指定できます。その場合、ディストリビューションは `userid_` から始まるすべての Cookie をオリジン `userid_*` に転送します。

クエリ文字列の転送

ディストリビューションがオリジンにクエリ文字列を転送するかどうか、および転送する場合にどれを転送するかをコントロールします。クエリ文字列は、指定されたパラメータに値を割り当てる URL の一部です。例えば、`https://example.com/over/there?name=ferret` URL は `name=ferret` クエリ文字列を含みます。サーバーは、そのようなページのリクエストを受信すると、プログラムを実行し、`name=ferret` クエリ文字列を変更せずにプログラムに渡します。疑問符はセパレータとして使用され、クエリ文字列の一部ではありません。

ディストリビューションがクエリ文字列を転送しないようにするか、指定したクエリ文字列のみを転送するかを選択できます。オリジンがクエリ文字列パラメータの値に関係なくコンテンツの同じバージョンを返す場合、クエリ文字列を転送しないように選択します。これにより、 ディストリビューションがキャッシュからリクエストを処理できる可能性が高くなり、パフォーマンスが向上し、オリジンの負荷が軽減されます。オリジンサーバーが 1 つ以上のクエリ文字列パラメータに基づいて、異なるバージョンのコンテンツを返す場合、指定したクエリ文字列のみを転送するように選択します。

ディストリビューションのキャッシュ動作を変更する

ディストリビューションのデフォルトのキャッシュ動作を変更するには、以下の手順を実行します。

1. Lightsail コンソールにサインインします。
2. Lightsail のホームページで、[ネットワーキング] タブを選択します。
3. デフォルトのキャッシュ動作を変更するディストリビューションの名前を選択します。
4. ディストリビューション管理ページのキャッシュタブを開きます。
5. キャッシュを設定するセクションで、ディストリビューションのキャッシュプリセットを選択します。詳細については、キャッシュプリセットを参照してください。
6. デフォルトのキャッシュ動作を変更するを選択して、ディストリビューションのデフォルト動作を変更します。次に、ディストリビューションのデフォルト動作を選択します。詳細については、デフォルト動作を参照してください。
7. パスの追加を選択して、ディストリビューションのキャッシュ動作にディレクトリとファイルの上書きを追加します。詳細については、ディレクトリとファイルの上書きを参照してください。
8. 編集したいディストリビューションの詳細設定の横に表示される鉛筆アイコンを選択します。詳細については、キャッシュ動作詳細設定を参照してください。

ディストリビューションの設定を保存したら、すべてのエッジロケーションに伝達し始めます。エッジロケーションで設定が更新されるまでは、 以前の設定に基づいて、そののロケーションからコンテンツを引き続き供給します。エッジロケーションで設定がが更新されると、 新しい設定に基づいて、そのロケーションからコンテンツを直ちに供給し始めます。

変更は、すべてのエッジロケーションにすぐに伝達されるわけではありません。伝達が完了すると、ディストリビューションのステータスが進行中からデプロイ済みに変わります。ディストリビューションがが変更を伝達している間、特定のエッジロケーションでコンテンツが以前の設定または新しい設定のどちらに基づいて供給されるかを判別することはできません。

トピック

Lightsail ディストリビューションのキャッシュをリセット

キャッシュの寿命 (有効期限 [TTL]) 設定は、コンテンツが Amazon Lightsail ディストリビューションに残る期間をコントロールします。キャッシュの有効期限より前にキャッシュをクリアにする必要がある場合は、ディストリビューションのキャッシュを手動でリセットすることもできます。キャッシュをクリアにすると、次回ユーザーがコンテンツをリクエストした際、ディストリビューションはコンテンツの最新バージョンをオリジンから取り出し、キャッシュします。このガイドでは、ディストリビューションでキャッシュを手動でリセットする方法を説明します。ディストリビューションの詳細については、「コンテンツ配信ネットワークディストリビューション」を参照してください。

ディストリビューションのキャッシュをリセット

ディストリビューションのキャッシュをリセットするには、以下の手順を実行します。

1. Lightsail コンソールにサインインします。
2. Lightsail のホームページで、[ネットワーキング] タブを選択します。
3. キャッシュをリセットしたいディストリビューションの名前を選択します。
4. ディストリビューションの管理ページで [Cache] タブを選択します。
5. ページの [キャッシュのリセット] セクションまでスクロールして、[キャッシュのリセット] を選択します。
6. 確認プロンプトで [はい、リセットします] を選択してディストリビューションのキャッシュのリセットを確定します。または [いいえ、キャンセル] を選択して、ディストリビューションのキャッシュをリセットしないようにします。

Lightsail ディストリビューションのオリジンを変更する

このガイドでは、Amazon Lightsail ディストリビューションを作成後、オリジンの変更方法を紹介します。オリジンとは、あなたのディストリビューションのためのコンテンツの決定的なソースです。ディストリビューションを作成するときに、Lightsail インスタンス、Lightsail バケット、Lightsail ウェブサイトまたはウェブアプリケーションのコンテンツをホストするロードバランサー (1 つ以上のインスタンスが添付されている) を選択します。詳細については、「コンテンツ配信ネットワークディストリビューション」を参照してください。

ディストリビューションを作成後、いつでもオリジンを変更できます。オリジンを変更する時、直ちにディストリビューションはエッジロケーションに変更を適用します。ディストリビューションがエッジロケーションの新しいオリジンに更新されるまで、古いオリジンにリクエストが転送されます。

オリジンを変更しても、 ディストリビューションは新しいオリジンからのコンテンツでエッジキャッシュを生成し直す必要はありません。ウェブサイトまたはウェブアプリケーション内でユーザーのリクエストが変更されていない限り、コンテンツのキャッシュのライフスパンが切れるまで、ディストリビューションは、エッジキャッシュに既存するコンテンツを供給します。

オリジンプロトコルポリシー

オリジンプロトコルポリシーは、オリジンからコンテンツを引き出す時にディストリビューションが使用するプロトコルポリシーです。ディストリビューションのオリジンを選択した後、オリジンからコンテンツを引き出す際に、Hypertext Transfer Protocol (HTTP) か、Hypertext Transfer Protocol Secure (HTTPS) どちらを使用すべきか決めます。オリジンが HTTPS 用に設定されていない場合は、HTTP を使用する必要があります。

ディストリビューションに対して、次のいずれかのオリジンプロトコルポリシーを選択できます。

- HTTP Only - オリジンへのアクセスに HTTP のみを使用します。これはデフォルトの設定です。
- HTTPS Only - オリジンへのアクセスに HTTPS のみを使用します。

オリジンプロトコルポリシーを編集する手順は、このガイドのディストリビューションのオリジンを変更するセクションを参照してください。

ディストリビューションのオリジンを変更する

ディストリビューションのオリジンの変更を行うには、以下の手順を実行します。

1. Lightsail コンソールにサインインします。
2. Lightsail のホームページで、[ネットワーキング] タブを選択します。
3. オリジンを変更したいディストリビューションの名前を選択します。
4. ディストリビューション管理ページの詳細タブを選択して、オリジンの選択までスクロールします。

   オリジンの選択セクションには、ディストリビューションの現在のオリジンが表示されます。

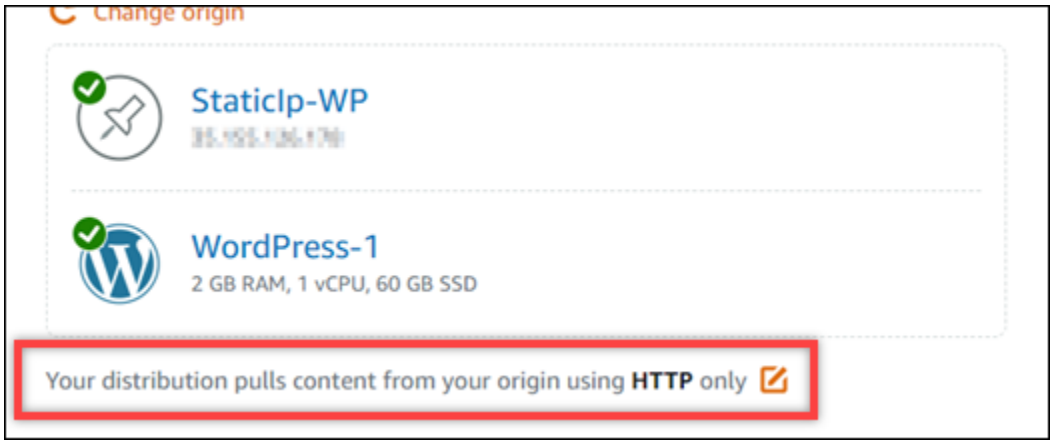
5. オリジンを変更を選択します。
6. オリジンのリソースが作成された AWS リージョンを選択します。

   ディストリビューションはグローバルリソースです。どの AWS リージョンでもオリジンをリファレンスでき、グローバルにそのコンテンツを配信することができます。
7. オリジンの選択。オリジンは、インスタンス、バケット、またはロードバランサー (1 つ以上のインスタンスが添付されている) にすることが可能です。
8. 保存を選択して、新しいオリジンでディストリビューションを更新します。

   ディストリビューションのオリジンを選択したら、オリジンからコンテンツを引き出す際に、Hypertext Transfer Protocol (HTTP) かHypertext Transfer Protocol Secure (HTTPS) のどちらを使用するかを決める必要があります。
9. (オプション) オリジンプロトコルポリシーを変更するには、ディストリビューションが使用する現在のオリジンプロトコルポリシーの横に表示される鉛筆アイコンを選択します。詳細については、オリジンプロトコルポリシーを参照してください。

   このオプションは、オリジンの選択セクションにあり、選択したディストリビューションのオリジンリソース下にあります。

> **Note**
>
> Lightsail バケットをディストリビューションのオリジンとして選択すると、[Origin protocol policy] (オリジンプロトコルポリシー) はデフォルトで [HTTPS only] (HTTPS のみ) に設定されます。バケットがディストリビューションのオリジンである場合、オリジンプロトコルポリシーを変更することはできません。



10. HTTP Only または HTTPS Only を選択して、保存を選択してオリジンプロトコルポリシーを保存します。

ディストリビューション設定に変更を保存すると、 変更をすべてのエッジロケーションに伝達し始めます。エッジロケーションで設定が更新されるまでは、 以前の設定に基づいて、そののロケーションからコンテンツを引き続き供給します。エッジロケーションで設定がが更新されると、 新しい設定に基づいて、そのロケーションからコンテンツを直ちに供給し始めます。

変更は、すべてのエッジロケーションにすぐに伝達されるわけではありません。伝達が完了すると、ディストリビューションのステータスが進行中からデプロイ済みに変わります。ディストリビューションがが変更を伝達している間、特定のエッジロケーションでコンテンツが以前の設定または新しい設定のどちらに基づいて供給されるかを判別することはできません。

Lightsail ディストリビューションのプランを変更する

Amazon Lightsail ディストリビューションを作成する際は、毎月のデータ転送クォータとディストリビューションのコストに適したディストリビューションプランを選択します。プランの月次データ転送クォータよりも多くのデータが配信される場合、超過分が課金されます。超過料金の詳細については、Lightsail料金表を参照ください。

超過料金が発生しないようにするには、クォータを超える前に、現在のディストリビューションのプランを毎月のデータ転送量が多い別のプランに変更します。ディストリビューションのプランは、各 AWS 請求サイクルで 1 回だけ変更できます。このガイドでは、ディストリビューションのプランの変更方法を説明します。

ディストリビューションの詳細については、「コンテンツ配信ネットワークディストリビューション」を参照してください。

ディストリビューションプランを変更する

ディストリビューションのプランを変更するには、以下の手順を行います。

1. Lightsail コンソールにサインインします。
2. Lightsail のホームページで、[ネットワーキング] タブを選択します。
3. 参照したい現在の月間データ転送のディストリビューション名を選択します。
4. ディストリビューション管理ページにある詳細タブを選択します。
5. データ転送セクションのページで、ディストリビューションプラン変更を選択します。
6. 確認プロンプトではい、変更しますを選択し、確認します。

7. 次のプロンプトで、新しいディストリビューションプランを選択しプランの選択を選択します。
8. 次のプロンプトで、はい、適用しますを選択して、新しいディストリビューションプランを適用することを確認します。いいえ、戻るを選択すると、新しいプランは適用されません。

Lightsail ディストリビューションのカスタムドメイン

登録済みドメイン名をディストリビューションで使用するために、カスタムドメインを Amazon Lightsail ディストリビューションと動作するように設定します。カスタムドメインを有効にする前は、ディストリビューションを最初に作成したときに関連付けられたデフォルトドメイン (例: `123456abcdef.cloudfront.net`) に対してのみ、ディストリビューションはトラフィックを受け入れます。カスタムドメインを有効にする際、ディストリビューションと動作するように作成したドメインの Lightsail SSL/TLS 証明書を選択する必要があります。カスタムドメインを有効にすると、選択した証明書に関連付けられているすべてのドメインのトラフィックがディストリビューションで受け入れられます。

> **Important**
>
> ディストリビューション事に、一度に 1 つの証明書のみを使用することができます。ディストリビューションでカスタムドメインを無効にすると、カスタムドメインを再度有効にするまで、登録したドメインの HTTPS トラフィックをディストリビューションで処理できなくなります。
>
> SSL/TLS 証明書に関連付けられたドメイン名は、Amazon CloudFront サービスのディストリビューションを含む、すべての Amazon Web Services (AWS) アカウントのディストリビューションで使用することはできません。ドメインの証明書を作成することはできますが、ディストリビューションと使用することはできません。

ディストリビューションの詳細については、「コンテンツ配信ネットワークディストリビューション」を参照してください。

前提条件

開始する前に、Lightsail ディストリビューションを作成する必要があります。詳細については、「ディストリビューションを作成する」を参照してください。

ディストリビューション用の SSL/TLS 証明書の作成と検証が必要です。詳細については、「ディストリビューションの SSL/TLS 証明書を作成する」および「ディストリビューションの SSL/TLS 証明書を検証する」を参照してください。

ディストリビューションのカスタムドメインを有効にする

ディストリビューションのカスタムドメインを有効にするには、以下の手順を実行します。

1. Lightsail コンソールにサインインします。
2. Lightsail のホームページで、[ネットワーキング] タブを選択します。
3. カスタムドメインを有効にするディストリビューションの名前を選択します。
4. ディストリビューション管理ページで [カスタムドメイン] タブを選択します。
5. [証明書のアタッチ] を選択します。

   証明書がない場合は、ディストリビューションにアタッチする前に、ドメインの SSL/TLS 証明書を作成してから検証する必要があります。詳細については、「ディストリビューションの SSL/TLS 証明書を作成する」を参照してください。
6. 表示されるドロップダウンメニューで、ディストリビューションと使用するドメインの有効な証明書を選択します。
7. 証明書情報が正しいことを確認し、[アタッチ] を選択します。
8. ディストリビューションの [Status] (ステータス) が [Updating] (更新中) に変わります。ステータスが [Enabled] (使用可能) に変わると、証明書のドメインが [Custom domains] (カスタムドメイン) セクションに表示されます。
9. [Add domain assignment] (ドメイン割り当てを追加) を選択して、ドメインがディストリビューションを指すようにします。
10. 証明書と DNS 情報が正しいことを確認し、[Add assignment] (割り当てを追加) を選択します。しばらくすると、選択したドメインのトラフィックがディストリビューションによって受け入れられ始めます。

トピック

- ドメインを Lightsail ディストリビューションにポイントする
- Lightsail ディストリビューションのカスタムドメインを変更する
- Lightsail ディストリビューションのカスタムドメインを無効にする
- ディストリビューションのデフォルトドメインを Lightsail コンテナサービスに追加する

ドメインを Lightsail ディストリビューションにポイントする

ディストリビューションのカスタムドメインを有効にしたら、メンバードメイン名を Amazon Lightsail ディストリビューションに向ける必要があります。ディストリビューションで使用中の証明書に指定されている、各ドメインの DNS ゾーンにエイリアスレコードを追加して行います。追加するレコードはすべて、ディストリビューションのデフォルトのドメイン (例: `123456abcdef.cloudfront.net`) に向ける必要があります。

このガイドでは、Lightsail DNS ゾーンを使用してディストリビューションにドメインを向ける手順について説明しています。Domain.com や GoDaddy などの別の DNS ホスティングプロバイダーを使用して、ドメインをディストリビューションに向ける手順と類似しているかもしれません。Lightsail DNS ゾーンの詳細については、「DNS」を参照してください。

ディストリビューションの詳細は、「ディストリビューションを作成する」を参照してください。

目次

ステップ 1: 前提条件を満たす

開始するには、Lightsail ディストリビューション用のカスタムドメインを有効にします。詳細については、「ディストリビューション用のカスタムドメインを有効にする」を参照してください。

ステップ 2: ディストリビューションのデフォルトドメインを取得する

以下の手順を実行して、ディストリビューションのデフォルトドメイン名を取得します。このドメイン名は、ドメインの DNS にエイリアスレコードを追加するときに指定します。

1. Lightsail コンソールにサインインします。
2. Lightsail のホームページで、[ネットワーキング] タブを選択します。
3. デフォルトのドメイン名を取得したいディストリビューション名を選択します。
4. ディストリビューションの管理ページのヘッダーセクションにある、ディストリビューションのデフォルトドメイン名を書き留めます。ディストリビューションのデフォルトドメイン名は `123456abcdef.cloudfront.net` と類似しています。

この値は、ドメイン DNS のエイリアスレコードのパートとして、追加する必要があります。この値はテキストファイルにコピー、ペーストして、後で参照できるようにしておくことをお勧めします。このチュートリアルの次の「ステップ 3: ドメインの DNS ゾーンにレコードを追加する」セクションに進みます。

ステップ 3: ドメインの DNS ゾーンにレコードを追加する

ドメインの DNS ゾーンにレコードを追加するには、次のステップを実行します。

1. Lightsail のホームページで [Domains & DNS] (ドメインと DNS) タブを選択します。
2. ページの [DNS zones] (DNS ゾーン) セクションで、レコードを追加したいドメイン名を選択します。そのレコードがユーザーのドメインへのトラフィックをディストリビューションに送信します。
3. [DNS records] (DNS レコード) タブを選択します。次に、[Add record] (レコードの追加) を選択します。
4. ディストリビューションにポイントするドメインのタイプに応じて、以下のいずれかの手順を実行します。

   - アドレス (A) レコードを選択して、頂点ドメイン (例: `example.com`) を ディストリビューション)にポイントします。

     ドメインの頂点の A レコードが DNS ゾーンにすでに存在する場合は、別の A レコードを追加するのではなく、既存のレコードを編集する必要があります。
   - 正規名 (CNAME) を選択して、`website.example.com` などのサブドメインをディストリビューションに対しポイントします。
5. A レコードを追加する場合は、[Resolves to] (解決先) テキストボックスでディストリビューション名を選択します。CNAME レコードを追加する場合は、[Maps to] (マッピング先) テキストボックスにディストリビューションのデフォルトドメイン名を入力します。

> **Note**
>
> DNS ゾーンに A レコードを追加してディストリビューション名を選択すると、アドレスレコードとは異なるエイリアスレコードが追加されます。Lightsail により、他の DNS ホスティングプロバイダーで通常必要とされる追加の手順を行わずに、エイリアスレコードを簡単に追加できます。

6. 保存アイコンを選択して、レコードを DNS ゾーンに保存します。

   これらの手順を繰り返すと、ディストリビューションで使用している証明書と紐づくドメイン用に、他の DNS レコードも追加できます。変更がインターネットの DNS を通じて伝達されるまで待ちます。数分後に、ドメインがディストリビューションをポイントしているか確認してください。なお、ディストリビューションもテストする必要があります。詳細については、次の「ディストリビューションのテスト」を参照してください。

Lightsail ディストリビューションのカスタムドメインを変更する

Amazon Lightsail ディストリビューションが使用しているカスタムドメインを他のドメインあるいはドメインのセットに変更することができます。変更するには、ディストリビューションで使用するドメイン用の新しい SSL/TLS 証明書を作成する必要があります。詳細については、「ディストリビューションの SSL/TLS 証明書を作成する」を参照してください。　新しい証明書が検証されたら、古い証明書を新しい証明書とスワップします。これにより、ディストリビューションのカスタムドメインが変更されます。

ディストリビューションの詳細は、「ディストリビューションを作成する」を参照してください。

ディストリビューションのカスタムドメインを変更する

ディストリビューションのカスタムドメインを変更するには、以下の手順を実行します。

1. Lightsail コンソールにサインインします。
2. Lightsail のホームページで、[ネットワーキング] タブを選択します。
3. カスタムドメインを変更するディストリビューションの名前を選択します。
4. ディストリビューション管理ページのカスタムドメインタブを選択します。
5. ディストリビューションに現在アタッチされている SSL/TLS 証明書のアタッチを解除します。

   ディストリビューションのステータスが [In progress] (進行中) に変化します。
6. ディストリビューションのステータスが [Enabled] (有効) に戻ったら、[Attach certificate] (証明書をアタッチ) を選択します。
7. 表示されるドロップダウンメニューで、ディストリビューションと使用するドメインの有効な証明書を選択します。
8. 証明書情報が正しいことを確認し、[アタッチ] を選択します。
9. ドメインの DNS にドメイン割り当てを追加して、ドメインがディストリビューションを指すようにします。

ディストリビューションの [Status] (ステータス) が [Updating] (更新中) に変わります。ステータスが [Ready] (準備完了) に変わると、証明書のドメインが [Custom domains] (カスタムドメイン) セクションに表示されます。[Add domain assignment] (ドメイン割り当てを追加) を選択して、ドメインがディストリビューションを指すようにします。

10. [Add assignment] (割り当てを追加) を選択します。しばらくすると、選択したドメインのトラフィックがディストリビューションによって受け入れられ始めます。
11. [保存] を選択します。

Lightsail ディストリビューションのカスタムドメインを無効にする

Amazon Lightsail ディストリビューションのカスタムドメインを無効にし、登録済みドメイン名のディストリビューションとの使用を停止します。カスタムドメインを無効にすると、ディストリビューションは、最初に作成したときにディストリビューションに関連付けられたデフォルトドメイン (例:`123456abcdef.cloudfront.net`) のみを受け入れます。以前に関連付けられたカスタムドメインのトラフィックには 403 エラーが表示されます。

ディストリビューションの詳細については、「コンテンツ配信ネットワークディストリビューション」を参照してください。

ディストリビューションのカスタムドメインを無効にする

ディストリビューションのカスタムドメインを無効にするには、以下の手順を行います。

1. Lightsail コンソールにサインインします。
2. Lightsail のホームページで、[ネットワーキング] タブを選択します。
3. カスタムドメインを無効にするディストリビューションの名前を選択します。
4. ディストリビューション管理ページで [カスタムドメイン] タブを選択します。

   [Custom domains] (カスタムドメイン) ページには、ディストリビューションに現在アタッチされている SSL/TLS 証明書があれば表示されます。
5. 以下のオプションのいずれかを選択します。

   1. [Configure distribution domains] (ディストリビューションドメインを設定する) を選択して、以前に選択したドメインの選択を解除するか、ディストリビューションに関連付けられているドメインをさらに選択します。

2. [デタッチ] を選択してディストリビューションから証明書をデタッチし、関連付けられているすべてのドメインを削除します。

6. カスタムドメインを無効にするリクエストが送信され、ディストリビューションのステータスが [In progress] (進行中) へ変更されます。しばらくすると、ディストリビューションのステータスが[Enabled] (有効) に変更されます。

カスタムドメインを無効にすると、ディストリビューションは、最初に作成したときにディストリビューションに関連付けられたデフォルトドメイン (例:123456abcdef.cloudfront.net) のみを受け入れます。以前に関連付けられたカスタムドメインのトラフィックには 403 エラーが表示されます。ドメインの DNS レコードを更新して、それらのドメインのトラフィックが別のリソースに送信されるようにする必要があります。

ディストリビューションのデフォルトドメインを Lightsail コンテナサービスに追加する

Amazon Lightsail コンテナサービスをコンテンツ配信ネットワーク (CDN) ディストリビューションのオリジンとして選択できます。そうすると、ディストリビューションでは、コンテナサービスでホストされているウェブサイトまたはウェブアプリケーションがキャッシュされ提供されます。Lightsail コンテナサービスを使用した Lightsail ディストリビューションを使用している場合、Lightsail によって、ディストリビューションのデフォルトドメイン名がカスタムドメインとしてコンテナサービスに自動的に追加されます。これにより、ディストリビューションとコンテナサービスの間でトラフィックをルーティングできます。しかし、以下の状況においては、このガイドで説明されている手順を実行して、ディストリビューションのデフォルトドメイン名をコンテナサービスに手動で追加する必要があります。

- 何らかの不具合により、ディストリビューションのデフォルトドメイン名がコンテナサービスに自動的に追加されない場合。
- Lightsail ディストリビューション以外のディストリビューションをコンテナサービスで使用している場合。

AWS Command Line Interface (AWS CLI)を使用する場合のみ、ディストリビューションのデフォルトドメイン名をコンテナサービスに手動で追加できます。コンテナサービスの詳細については、「コンテナサービス」を参照してください。ディストリビューションの詳細については、「オブジェクトストレージ」を参照してください。

ディストリビューションのデフォルトドメインを コンテナサービスに追加する

AWS Command Line Interface (AWS CLI) を使用して、次の手順を実行し、Lightsail でディストリビューションのデフォルトドメインをコンテナサービスに追加します。これは、`update-container-service` コマンドを使用して実行できます。詳細については、AWS CLI コマンドリファレンスの「update-container-service」を参照してください。

> **Note**
>
> この手順を続行する前に、AWS CLI をインストールして Lightsail 用に設定する必要があります。 詳細については、「Lightsail で使用するために AWS CLI を設定する」を参照してください。

1. ターミナルまたはコマンドプロントウィンドウを開きます。
2. 以下のコマンドのいずれかを入力して、ディストリビューションのデフォルトドメインをコンテナサービスに追加します。

   > **Note**
   >
   > コンテナサービスにカスタムドメインを追加した場合は、カスタムドメインとディストリビューションのデフォルトドメインの両方を指定する必要があります。

   コンテナサービスにカスタムドメインが設定されていない場合:

   ```
   aws lightsail update-container-service --service-name ContainerServiceName --
   public-domain-names '{"_": ["DistributionDefaultDomain"]}'
   ```

   コンテナサービスに 1 つまたは複数のカスタムドメインが設定されている場合:

   ```
   aws lightsail update-container-service --service-name ContainerServiceName
    --public-domain-names '{"CertificateName": ["ExistingCustomDomain"],"_":
    ["DistributionDefaultDomain"]}'
   ```

   コマンドで、次のサンプルテキストを独自のテキストに置き換えます。

- *ContainerServiceName* - ディストリビューションのオリジンとして指定された Lightsail コンテナサービスの名前。
- *DistributionDefaultDomain* - コンテナサービスをオリジンとして使用しているディストリビューションのデフォルトドメイン。例えば、example123.cloudfront.net です。
- *CertificateName* - コンテナサービスに現在アタッチされているカスタムドメインがある場合、その Lightsail 証明書の名前。コンテナサービスにアタッチされたカスタムドメインがない場合は、コンテナサービスでカスタムドメインが設定されていないというラベルの付いたコマンドを使用します。
- *DistributionDefaultDomain* - コンテナサービスに現在アタッチされているカスタムドメイン。

例:

- コンテナサービスにカスタムドメインが設定されていない場合:

```
aws lightsail update-container-service --service-name ContainerServiceName --
public-domain-names '{"_": ["example123.cloudfront.net"]}'
```

- コンテナサービスに 1 つまたは複数のカスタムドメインが設定されている場合:

```
aws lightsail update-container-service --service-name ContainerServiceName
 --public-domain-names '{"example-com": ["example.com"],"_":
 ["example123.cloudfront.net"]}'
```

Lightsail ディストリビューションのリクエストとレスポンスの動作

このガイドでは、リクエストを処理してオリジンに転送したり、オリジンからのレスポンスを処理したりする場合の Amazon Lightsail ディストリビューションの動作について説明します。ディストリビューションの詳細については、「コンテンツ配信ネットワークディストリビューション」を参照してください。

トピック

- ディストリビューションがリクエストを処理してオリジンに転送する方法
- ディストリビューションがオリジンからの応答を処理する方法

ディストリビューションがリクエストを処理してオリジンに転送する方法

このトピックには、ディストリビューションがビューワーリクエストを処理してオリジンに転送する方法に関する情報が含まれています。

目次

認証

DELETE、GET、HEAD、PATCH、POST、PUT リクエストの場合、Authorization ヘッダーをオリジンに転送するように ディストリビューションを設定すると、クライアント認証を要求するようにオリジンサーバーを設定できます。

OPTIONS リクエストの場合、次のディストリビューション設定を使用した場合のみ、クライアント認証を要求するようにオリジンサーバーを設定することができます。

- Authorization ヘッダーをオリジンに転送するようにディストリビューションを設定する。
- OPTIONS リクエストへの応答をキャッシュしないようにディストリビューションを設定する。

HTTP または HTTPS のいずれかを使用してオリジンにリクエストを転送するようにディストリビューションを構成することができます。

キャッシュ期間

ディストリビューションが別のリクエストをオリジンに転送するまでにオブジェクトをキャッシュに保持する時間をコントロールするには :

- Cache-Control または Expires ヘッダーフィールドを各オブジェクトに追加するようにオリジンを構成します。
- キャッシュ寿命 (TTL) には、デフォルト値の 1 日を使用します。

ディストリビューション設定の詳細については、「ディストリビューションアドバンスト設定」を参照してください。

クライアント IP アドレス

ビューワーがリクエストをディストリビューションに送信し、X-Forwarded-For リクエストヘッダーを含めない場合、ディストリビューションは TCP 接続からビューワーの IP アドレスを取得して、IP アドレスが含まれた X-Forwarded-For ヘッダーを追加し、リクエストをオリジンに転送します。たとえば、ディストリビューションが TCP 接続から IP アドレス 192.0.2.2 を取得する場合、以下のヘッダーをオリジンに転送します。

```
X-Forwarded-For: 192.0.2.2
```

ビューワーがリクエストをディストリビューションに転送して X-Forwarded-For リクエストヘッダーを含める場合、 ビューワーの IP アドレスを TCP 接続から取得してそれを X-Forwarded-For

ヘッダーの末尾に追加し、リクエストをオリジンに転送します。たとえば、ビューワーのリクエストに X-Forwarded-For: 192.0.2.4,192.0.2.3 が含まれ、ディストリビューションが TCP 接続から IP アドレス 192.0.2.2 を取得する場合、以下のヘッダーをオリジンに転送します。

```
X-Forwarded-For: 192.0.2.4,192.0.2.3,192.0.2.2
```

ロードバランサー (Elastic Load Balancing を含む)、ウェブアプリケーションファイアウォール、リバースプロキシ、侵入防御システム、API Gateway などの一部のアプリケーションでは、リクエストを転送したディストリビューションエッジサーバーの IP アドレスを X-Forwarded-For ヘッダーの末尾に付加します。たとえば、ディストリビューションから ELB に転送するリクエストに X-Forwarded-For: 192.0.2.2 が含まれていて、 エッジサーバーの IP アドレスが 192.0.2.199 である場合、インスタンスで受け取るリクエストのヘッダーは次のようになります。

```
X-Forwarded-For: 192.0.2.2,192.0.2.199
```

> **Note**
>
> X-Forwarded-For ヘッダーには、IPv4 アドレス (192.0.2.44 など) および IPv6 アドレス (2001:0db8:85a3:0000:0000:8a2e:0370:7334 など) が含まれます。

クライアント側の SSL 認証

Lightsail ディストリビューションは、クライアント側 SSL 証明書によるクライアント認証をサポートしていません。オリジンがクライアント側証明書をリクエストした場合、 ディストリビューションはリクエストを削除します。

圧縮

Lightsail ディストリビューションは、Accept-Encodingというフィールド値を持つリクエストを転送します。"identity" "gzip"

条件付きリクエスト

ディストリビューションは、エッジキャッシュで有効期限切れになっているオブジェクトに対するリクエストを受け取ると、リクエストをオリジンに転送し、オブジェクトの最新バージョンを取得するか、 エッジキャッシュに最新バージョンが既に存在することをオリジンに確認します。通常、オリジンはオブジェクトをディストリビューションに最後に送信するときに、ETag 値または LastModified 値、あるいはその両方の値をレスポンスに含めます。ディストリビューションがオリジンに転送する新しいリクエストには、次のどちらかまたは両方を追加します。

- オブジェクトの有効期限切れバージョンの `If-Match` 値が含まれる `If-None-Match` または `ETag` ヘッダー。
- オブジェクトの有効期限切れバージョンの `If-Modified-Since` 値が含まれる `LastModified` ヘッダー。

オリジンは、この情報を使用して、オブジェクトが更新されているかどうかを判別します。つまり、オブジェクト全体をディストリビューションに返すか、または HTTP 304 ステータスコード (変更なし) のみを返すかを判別します。

cookie

Cookie をオリジンに転送するようにディストリビューションを構成できます。詳細については、「ディストリビューションアドバンスト設定」を参照してください。

クロスオリジンリソース共有 (CORS)

ディストリビューションで Cross-Origin Resource Sharing 設定を尊重する場合は、 `Origin` ヘッダーをオリジンに転送するように設定します。

暗号化

ビューワーに HTTPS を使用してディストリビューションに接続するように要求し、HTTP または HTTPS を使用してリクエストをオリジンに転送するようにディストリビューションに要求することができます。

ディストリビューションは、SSLv3、TLSv1.0、TLSv1.1、および TLSv1.2 プロトコルを使用して、HTTPS リクエストをオリジンに転送します。SSL と TLS のその他のバージョンはサポートされていません。

本文を含む GET リクエスト

ビューワーの `GET` リクエストの本文が含まれている場合、 ディストリビューションはビューワーに HTTP ステータスコード 403 (禁止) を返します。

HTTP メソッド

サポートするすべての HTTP メソッドを許可するようディストリビューションを構成すると、ディストリビューションはビューワーからの以下のリクエストを受け入れてオリジンに転送します。

- DELETE
- GET
- HEAD
- OPTIONS
- PATCH
- POST
- PUT

ディストリビューションは、GET リクエストと HEAD リクエストへの応答を常にキャッシュします。OPTIONS リクエストへの応答をキャッシュするように設定することもできます。ディストリビューションはその他のメソッドを使用するリクエストへのレスポンスをキャッシュしません。

オリジンが上記のメソッドを処理するかどうかを構成する方法の詳細については、オリジンのドキュメントを参照してください。

> **Important**
>
> ディストリビューションがサポートするすべての HTTP メソッドを受け入れてオリジンに転送するように設定する場合、オリジンサーバーがすべてのメソッドを処理するように構成します。たとえば、POST を使用するために、上記のメソッドを受け入れて転送するようにディストリビューションを構成する場合は、DELETE リクエストを適切に処理するようオリジンサーバーを設定して、削除すべきでないリソースをビューワーが削除できないようにする必要があります。詳細については、HTTP サーバーのドキュメントを参照してください。

HTTP リクエストヘッダーとディストリビューション動作

次の表は、オリジンに転送できる HTTP リクエストヘッダーを示しています (例外も注記されています)。この表には、各ヘッダーについて以下に関する情報も含まれています。

- サポート - そのヘッダーの値に基づいてオブジェクトをキャッシュするようにディストリビューションを設定できるかどうか。

  Date および User-Agent ヘッダーの値に基づいてオブジェクトをキャッシュするようにディストリビューションを設定できますが、これはお勧めできません。これらのヘッダーには可能な値が多数あり、その値に基づいてキャッシュすると、ディストリビューションがオリジンに転送するリクエストの数が大幅に増加します。

- 設定していない場合の動作 - ヘッダーをオリジンに転送するように設定していない場合、ディストリビューションはヘッダー値に基づいてオブジェクトをキャッシュします。

- ヘッダー - 他の定義されたヘッダー

  サポート - あり

  設定されていない場合の動作 - ディストリビューションはヘッダーをオリジンに転送します。
- ヘッダー - Accept

  サポート - あり

  設定されていない場合の動作 - ディストリビューションはヘッダーを削除します。
- ヘッダー - Accept-Charset

  サポート - あり

  設定されていない場合の動作 - ディストリビューションはヘッダーを削除します。
- ヘッダー - Accept-Encoding

  サポート - あり

  設定されていない場合の動作 - 値に gzip が含まれる場合、ディストリビューションは Accept-Encoding: gzip をオリジンに転送します。値に gzip が含まれない場合、ディストリビューションはリクエストをオリジンに転送する前に Accept-Encoding ヘッダーフィールドを削除します。
- ヘッダー - Accept-Language

  サポート - あり

  設定されていない場合の動作 - ディストリビューションはヘッダーを削除します。
- ヘッダー - Authorization

  サポート - あり

  設定されていない場合の動作:

  - GET、HEAD の各リクエスト – ディストリビューションは、リクエストをオリジンに転送する前に Authorization ヘッダーフィールドを削除します。

  - OPTIONS リクエスト – OPTIONS リクエストへの応答をキャッシュするようにディストリビューションを設定した場合、ディストリビューションは、リクエストをオリジンに転送する前に、Authorization ヘッダーフィールドを削除します。

    OPTIONS リクエストへの応答をキャッシュするようにディストリビューションを設定しなかった場合、ディストリビューションは Authorization ヘッダーフィールドをオリジンに転送します。

  - DELETE、PATCH、POST、PUT の各リクエスト – ディストリビューションは、リクエストをオリジンに転送する前にヘッダーフィールドを削除しません。

- ヘッダー - Cache-Control

  サポート - なし

  設定されていない場合の動作-ディストリビューションはヘッダーをオリジンに転送します。

- ヘッダー - CloudFront-Forwarded-Proto

  サポート - あり

  設定されていない場合の動作 - ディストリビューションは、リクエストをオリジンに転送する前にヘッダーを追加しません。

- ヘッダー - CloudFront-Is-Desktop-Viewer

  サポート - あり

  設定されていない場合の動作 - ディストリビューションは、リクエストをオリジンに転送する前にヘッダーを追加しません。

- ヘッダー - CloudFront-Is-Mobile-Viewer

  サポート - あり

  設定されていない場合の動作 - ディストリビューションは、リクエストをオリジンに転送する前にヘッダーを追加しません。

- ヘッダー - CloudFront-Is-Tablet-Viewer

  サポート - あり

  設定されていない場合の動作 - ディストリビューションは、リクエストをオリジンに転送する前にヘッダーを追加しません。

- ヘッダー - CloudFront-Viewer-Country

  サポート - あり

  設定されていない場合の動作 - ディストリビューションは、リクエストをオリジンに転送する前にヘッダーを追加しません。
- ヘッダー - Connection

  サポート - なし

  設定されていない場合の動作 - ディストリビューションは、オリジンに転送する前に、このヘッダーをConnection: Keep-Aliveで置き換えます。
- ヘッダー - Content-Length

  サポート - なし

  設定されていない場合の動作-ディストリビューションはヘッダーをオリジンに転送します。
- ヘッダー - Content-MD5

  サポート - あり

  設定されていない場合の動作-ディストリビューションはヘッダーをオリジンに転送します。
- ヘッダー - Content-Type

  サポート - あり

  設定されていない場合の動作-ディストリビューションはヘッダーをオリジンに転送します。
- ヘッダー - Cookie

  サポート - なし

  設定されていない場合の動作 - Cookie を転送するようにディストリビューションを設定している場合、Cookieヘッダーフィールドをオリジンに転送します。そうでない場合、ディストリビューションは Cookie ヘッダーフィールドを削除します。
- ヘッダー - Date

  サポート対象 - あり、ただし推奨されません

  設定されていない場合の動作 - ディストリビューションはヘッダーをオリジンに転送します。
- ヘッダー - Expect

サポート - あり

設定されていない場合の動作 - ディストリビューションはヘッダーを削除します。

- ヘッダー - From

  サポート - あり

  設定されていない場合の動作-ディストリビューションはヘッダーをオリジンに転送します。

- ヘッダー - Host

  サポート - あり

  設定されていない場合の動作 - ディストリビューションは、リクエストされたオブジェクトに関連付けられたオリジンのドメイン名に値を設定します。

- ヘッダー - If-Match

  サポート - あり

  設定されていない場合の動作-ディストリビューションはヘッダーをオリジンに転送します。

- ヘッダー - If-Modified-Since

  サポート - あり

  設定されていない場合の動作-ディストリビューションはヘッダーをオリジンに転送します。

- ヘッダー - If-None-Match

  サポート - あり

  設定されていない場合の動作-ディストリビューションはヘッダーをオリジンに転送します。

- ヘッダー - If-Range

  サポート - あり

  設定されていない場合の動作-ディストリビューションはヘッダーをオリジンに転送します。

- ヘッダー - If-Unmodified-Since

  サポート - あり

  設定されていない場合の動作-ディストリビューションはヘッダーをオリジンに転送します。

- ヘッダー - Max-Forwards

  サポート - なし

  設定されていない場合の動作-ディストリビューションはヘッダーをオリジンに転送します。
- ヘッダー - Origin

  サポート - あり

  設定されていない場合の動作-ディストリビューションはヘッダーをオリジンに転送します。
- ヘッダー - Pragma

  サポート - なし

  設定されていない場合の動作-ディストリビューションはヘッダーをオリジンに転送します。
- ヘッダー - Proxy-Authenticate

  サポート - なし

  設定されていない場合の動作 - ディストリビューションはヘッダーを削除します。
- ヘッダー - Proxy-Authorization

  サポート - なし

  設定されていない場合の動作 - ディストリビューションはヘッダーを削除します。
- ヘッダー - Proxy-Connection

  サポート - なし

  設定されていない場合の動作 - ディストリビューションはヘッダーを削除します。
- ヘッダー - Range

  サポート対象 - あり (デフォルト)

  設定されていない場合の動作 - ディストリビューションはヘッダーをオリジンに転送します。
- ヘッダー - Referer

  サポート - あり

  設定されていない場合の動作 - ディストリビューションはヘッダーを削除します。

- ヘッダー - Request-Range

  サポート - なし

  設定されていない場合の動作 - > ディストリビューションはヘッダーをオリジンに転送します。
- ヘッダー - TE

  サポート - なし

  設定されていない場合の動作 - ディストリビューションはヘッダーを削除します。
- ヘッダー - Trailer

  サポート - なし

  設定されていない場合の動作 - ディストリビューションはヘッダーを削除します。
- ヘッダー - Transfer-Encoding

  サポート - なし

  設定されていない場合の動作-ディストリビューションはヘッダーをオリジンに転送します。
- ヘッダー - Upgrade

  サポート-いいえ (接続を除く) WebSocket

  設定されていない場合の動作- WebSocket 接続を確立していない限り、ディストリビューションによってヘッダーが削除されます。
- ヘッダー - User-Agent

  サポート - あり、ただし推奨されません

  設定されていない場合の動作 - ディストリビューションはこのヘッダーフィールドの値をAmazon CloudFrontで置き換えます。
- ヘッダー - Via

  サポート - あり

  設定されていない場合の動作-ディストリビューションはヘッダーをオリジンに転送します。
- ヘッダー - Warning

  サポート - あり

設定されていない場合の動作-ディストリビューションはヘッダーをオリジンに転送します。

- ヘッダー - X-Amz-Cf-Id

  サポート - なし

  設定されていない場合の動作 - ディストリビューションは、リクエストをオリジンに転送する前に、ビューワーリクエストにヘッダーを追加します。ヘッダー値には、リクエストを一意に識別する暗号化された文字列が含められます。

- ヘッダー - X-Edge-*

  サポート - なし

  設定されていない場合の動作 - あなたのディストリビューションは、すべてのX-Edge-*ヘッダー。

- ヘッダー - X-Forwarded-For

  サポート - あり

  設定されていない場合の動作-ディストリビューションはヘッダーをオリジンに転送します。

- ヘッダー - X-Forwarded-Proto

  サポート - なし

  設定されていない場合の動作 - ディストリビューションはヘッダーを削除します。

- ヘッダー - X-Real-IP

  サポート - なし

  設定されていない場合の動作 - ディストリビューションはヘッダーを削除します。

HTTP バージョン

ディストリビューションは HTTP/1.1 を使用してオリジンにリクエストを転送します。

リクエストの最大長と URL の最大長

パス、クエリ文字列 (ある場合)、ヘッダーを含め、リクエストの最大長は 20480 バイトです。

ディストリビューションはリクエストから URL を構築します。この URL の最大長は 8192 文字です。

リクエストまたは URL がこの最大制限を超えると、ディストリビューションは、リクエストエンティティが長すぎることを示す HTTP ステータスコード 413 (Request Entity Too Large) をビューワーに返してから、ビューワーへの TCP 接続を終了します。

OCSP Stapling

オブジェクトに対する HTTPS リクエストをビューワーが送信する際には、ドメインの SSL 証明書が無効になっていないことをディストリビューションまたはビューワーが認証機関 (CA) に対して確認する必要があります。OCSP Stapling を使用すると、ディストリビューションで証明書を検証して CA からの応答をキャッシュできるため、クライアントが直接 CA に対して証明書を検証する必要がなくなり、証明書の検証速度が向上します。

同一ドメイン内のオブジェクトに対する多数の HTTPS リクエストをディストリビューションが受信した場合は、OCSP Stapling によるパフォーマンス向上がさらに顕著になります。エッジロケーション内の各サーバーは、別々の検証リクエストを送信する必要があります。同一ドメインに対する多数の HTTPS リクエストを が受信するとすぐに、エッジロケーション内のすべてのサーバーが、SSL ハンドシェイクでパケットに "ステープリング" できるという CA からの応答を受信します。証明書が有効であることをビューワーが確認すると、ディストリビューションはリクエストされたオブジェクトを提供できます。エッジロケーション内でディストリビューションが十分なトラフィックを確保できない場合、新しいリクエストは、CA に対して証明書がまだ検証されていないサーバーに誘導される可能性が高くなります。この場合は、ビューワーが検証ステップを別途実行し、ディストリビューションサーバーがオブジェクトを提供します。このディストリビューションサーバーも CA に検証リクエストを送信するため、同じドメイン名が含まれるリクエストを次に受信したときには、CA からの検証応答が既に存在しているということになります。

永続的接続

ディストリビューションがオリジンからレスポンスを取得すると、その期間中に別のリクエストが届くのに備え、数秒間、接続を維持しようとします。持続的接続を維持すると、TCP 接続の再構築に必要な時間と後続のリクエストに対する別の TLS ハンドシェイクの実行に必要な時間を節約できます。

プロトコル

ディストリビューションは、Lightsail コンソールの Origin プロトコルポリシーフィールドの値に基づいて HTTP または HTTPS リクエストをオリジンサーバーに転送します。Lightsail コンソールでは、オプションは HTTP のみ、HTTPS のみとなっています。

[HTTP のみ] または [HTTPS のみ] を指定すると、ディストリビューションは、ビューワーリクエストのプロトコルに関係なく、指定されたプロトコルのみを使用してリクエストをオリジンに転送します。

> **Important**
>
> ディストリビューションが HTTPS プロトコルを使用してリクエストをオリジンに転送し、オリジンサーバーから無効な証明書または自己署名証明書が返された場合、ディストリビューションは TCP 接続を中断します。

クエリ文字列

ディストリビューションがクエリ文字列パラメータをオリジンに転送するかどうかを設定できます。

オリジン接続のタイムアウトと試行

デフォルトでは、ディストリビューションはセカンダリオリジンへの接続を試行したり、エラーレスポンスを返したりする前に 30 秒 (それぞれ 10 秒間の試行が 3 回) 待機します。

オリジン応答タイムアウト

オリジン応答タイムアウト (オリジンの読み取りタイムアウトまたはオリジンリクエストタイムアウトとも呼ばれます) は、次の両方に適用されます。

- ディストリビューションがリクエストをオリジンに転送してからレスポンスを受け取るまでの待機時間 (秒)
- ディストリビューションがオリジンからレスポンスのパケットを受け取ってから次のパケットを受け取るまでの待機時間 (秒)

ディストリビューションの動作は、ビューワーリクエストの HTTP メソッドによって決まります。

- GET および HEAD リクエスト – 応答タイムアウトの期間内にオリジンが応答しない場合、または応答を停止した場合、ディストリビューションは接続を中断します。指定されたオリジン接続の試行回数が 1 回を超える場合、ディストリビューションは完全な応答の取得を再試行します。オリジン接続の試行回数設定の値で決められているように、ディストリビューションは最大 3 回試行します。最後の試行でもオリジンが応答しない場合、ディストリビューションは同じオリジンのコンテンツに対する別のリクエストを受け取るまで接続を試みません。

- DELETE、OPTIONS、PATCH、PUT、POST の各リクエスト – オリジンが 30 秒以内に応答しない場合、ディストリビューションは接続を中断し、オリジンへの接続を再試行しません。クライアントは、必要に応じてリクエストを再送信できます。

同じオブジェクト (トラフィックスパイク) の同時リクエスト

ディストリビューションエッジロケーションがオブジェクトのリクエストを受け取り、オブジェクトが現在キャッシュにないか、有効期限が切れている場合、ディストリビューションはすぐにオリジンにリクエストを送信します。トラフィックスパイクがある場合 (同じオブジェクトへの追加のリクエストが、オリジンが最初のリクエストに応答する前にエッジロケーションに届く場合)、ディストリビューションは短時間一時停止してから、オブジェクトへの追加のリクエストをオリジンに転送します。通常、最初のリクエストへのレスポンスは、それ以降のリクエストに対するレスポンスの前に、ディストリビューションエッジロケーションに届きます。この短い停止により、オリジンサーバーでの不要な負荷が減ります。リクエストヘッダーや Cookie に基づいてキャッシュするようにディストリビューションを設定した場合など、追加のリクエストが同じでない場合、ディストリビューションはすべての一意のリクエストをオリジンに転送します。

ユーザーエージェントヘッダー

ユーザーがコンテンツの表示に使用しているデバイスに基づいて、オブジェクトの異なるバージョンを ディストリビューションでキャッシュするには、次の 1 つ以上のヘッダーをオリジンに転送するように ディストリビューションを設定することをお勧めします。

- `CloudFront-Is-Desktop-Viewer`
- `CloudFront-Is-Mobile-Viewer`
- `CloudFront-Is-SmartTV-Viewer`
- `CloudFront-Is-Tablet-Viewer`

ディストリビューションは、`User-Agent` ヘッダーの値に基づいて、これらのヘッダーの値を `true` または `false` に設定した後、リクエストをオリジンに転送します。デバイスが複数のカテゴリに属する場合は、複数の値が `true` になることがあります。たとえば、あるタブレットデバイスについて、ディストリビューションが`CloudFront-Is-Mobile-Viewer` と `CloudFront-Is-Tablet-Viewer` の両方を `true` に設定する場合があります。

`User-Agent` ヘッダーの値に基づいてオブジェクトをキャッシュするようにディストリビューションを設定できますが、これはお勧めできません。`User-Agent` ヘッダーには可能な値が多数あり、

その値に基づいてキャッシュすると、ディストリビューションがオリジンに転送するリクエストの数が大幅に増加します。

ディストリビューションが User-Agent ヘッダーの値に基づいてオブジェクトをキャッシュするように設定しない場合、ディストリビューションは以下の値を指定した User-Agent ヘッダーを追加して、リクエストをオリジンに転送します。

```
User-Agent = Amazon CloudFront
```

ディストリビューションは、ビューワーからのリクエストに User-Agent ヘッダーが含まれているかどうかに関係なく、このヘッダーを追加します。ビューワーからのリクエストに User-Agent ヘッダーが含まれる場合、ディストリビューションはそのヘッダーを削除します。

ディストリビューションがオリジンからの応答を処理する仕組み

このトピックには、 オリジンからのレスポンスを ディストリビューショが処理する方法に関する情報が含まれています。

目次

100-continue レスポンス

オリジンは複数の 100-continue レスポンスをディストリビューションに送信することはできません。最初の 100-continue レスポンスの後で、ディストリビューションは HTTP 200 OK レスポンス

を予期します。オリジンが最初のレスポンスの後に別の 100-continue レスポンスを送信すると、ディストリビューションはエラーを返します。

キャッシュ

- オリジンが Date および Last-Modified ヘッダーフィールドに有効かつ正確な値を設定していることを確認します。
- ビューワーからのリクエストに If-Match または If-None-Match リクエストヘッダーフィールドが含まれる場合、ETag レスポンスヘッダーフィールドを設定します。ETag の値が指定されていない場合、ディストリビューションは以降の If-Match または If-None-Match ヘッダーを無視します。
- 通常、ディストリビューションはオリジンからのレスポンスの Cache-Control: no-cache ヘッダーを優先します。例外については、「同じオブジェクトに対する同時要求 (トラフィックの急増) 」を参照してください。

取り消されたリクエスト

オブジェクトがエッジキャッシュになく、ディストリビューションがオブジェクトをオリジンから取得したものの、リクエストされたそのオブジェクトを配信する前にビューワーがセッションを終了すると (ブラウザを閉じるなど)、ディストリビューションはそのオブジェクトをエッジロケーションにキャッシュしません。

コンテンツネゴシエーション

オリジンが応答で Vary:* を返し、対応するキャッシュ動作の [最小 TTL] の値が [0] の場合、ディストリビューションはオブジェクトをキャッシュしますが、そのオブジェクトの後続のすべてのリクエストをオリジンに転送して、キャッシュにオブジェクトの最新バージョンが含まれていることを確認します。ディストリビューションには、If-None-Match や If-Modified-Since などの条件付きヘッダーは含まれません。その結果、オリジンはすべてのリクエストに応じてディストリビューションにオブジェクトを返します。

Vary:*オリジンがレスポンスで返され、対応するキャッシュ動作の Minimum TTL の値がそれ以外の値の場合は、ディストリビューションが削除または置換する HTTP CloudFront レスポンスヘッダーの説明に従ってヘッダーを処理します。Vary

cookie

キャッシュ動作の Cookie を有効にしており、オリジンが Cookie とオブジェクトを返す場合、ディストリビューションはオブジェクトと Cookie の両方をキャッシュします。これによってオブジェクトのキャッシュ性が低下することに注意してください。

切断された TCP 接続

オリジンがオブジェクトをディストリビューションに返している間にディストリビューションとオリジン間の TCP 接続が中断した場合、ディストリビューションの動作は、オリジンが `Content-Length` ヘッダーをレスポンスに含めたかどうかによって異なります。

- Content-Length ヘッダー – ディストリビューションは、オブジェクトをオリジンから取得すると、ビューワーにオブジェクトを返します。ただし、`Content-Length` ヘッダーの値がオブジェクトのサイズに一致しない場合、ディストリビューションはオブジェクトをキャッシュしません。
- Transfer-Encoding: Chunked – ディストリビューションは、オブジェクトをオリジンから取得すると、ビューワーにオブジェクトを返します。ただし、チャンクレスポンスが完了していない場合、ディストリビューションはオブジェクトをキャッシュしません。
- Content-Length ヘッダーなし – ディストリビューションはオブジェクトをビューワーに返して、オブジェクトをキャッシュしますが、オブジェクトが完全でない可能性があります。`Content-Length` ヘッダーがない場合、ディストリビューションは、TCP 接続が誤って中断されたか、または故意に中断されたかを判断できません。

`Content-Length` ヘッダーを追加して、ディストリビューションが不完全なオブジェクトをキャッシュしないように HTTP サーバーを設定することをお勧めします。

ディストリビューションが削除または置き換える HTTP レスポンスヘッダー

ディストリビューションは、 オリジンからのレスポンスをビューワーに転送する前に、以下のヘッダーフィールドを削除または更新します。

- `Set-Cookie` - Cookie を転送するようにディストリビューションを設定している場合、`Set-Cookie` ヘッダーフィールドがクライアントに転送されます。
- `Trailer`
- `Transfer-Encoding` - オリジンがこのヘッダーフィールドを返す場合、ディストリビューションは値を `chunked` に設定してからビューワーにレスポンスを返します。
- `Upgrade`

- Vary – 次の点に注意してください。
  - デバイス固有のヘッダーのいずれかをオリジン (CloudFront-Is-Desktop-Viewer、CloudFront-Is-Mobile-Viewer、CloudFront-Is-SmartTV-Viewer、CloudFront-Is-Tablet-Viewer) に転送するようにディストリビューションを設定しており、オリジンが Vary:User-Agent をディストリビューションに返すように設定している場合、ディストリビューションは Vary:User-Agent をビューワーに返します。
  - Varyヘッダーに、Accept-Encoding または Cookie のいずれかを含めるよう設定した場合、ディストリビューションはビューワーへの応答にその値を含めます。
  - ヘッダーの許可リストをオリジンに転送するようにディストリビューションを設定し、ヘッダー内のディストリビューションにヘッダー名を返すようにオリジンを設定した場合 (例:Vary:Accept-Charset,Accept-Language)、Varyディストリビューションはそれらの値を含むヘッダーをビューアーに返します。Vary
  - ディストリビューションが、* の Vary ヘッダーの値を処理するかについて詳しくは「コンテンツネゴシエーション」を参照してください。
  - Vary ヘッダーで他の値を返すようにオリジンを設定している場合、ディストリビューションは応答をビューワーに返す前にその値を削除します。
- Via – ディストリビューションは、ビューワーへの応答で値を次のように設定します。

  Via: *http-version alphanumeric-string*.cloudfront.net (CloudFront)

  たとえば、クライアントが HTTP/1.1 を介してリクエストを行った場合、値は次のようになります。

  Via: 1.1 1026589cc7887e7a0dc7827b4example.cloudfront.net (CloudFront)

最大ファイルサイズ

ディストリビューションがビューワーに返すレスポンス本文の最大サイズは 20 GB です。これには、Content-Length ヘッダーの値を指定しないチャンク転送レスポンスが含まれます。

使用できないオリジン

オリジンサーバーが使用できないときに、ディストリビューションがエッジキャッシュに存在するオブジェクトのリクエストを受け取り、そのオブジェクトが (たとえば Cache-Control max-age ディレクティブに指定された期間が経過しているために) 有効期限切れになっている場合、ディストリビューションは有効期限切れバージョンのオブジェクトを供給するか、またはカスタムエラーページを供給します。

場合によって、要求頻度の低いオブジェクトは削除されてエッジキャッシュで使用できなくなることがあります。ディストリビューションは、削除されたオブジェクトを提供することはできません。

リダイレクト

オリジンサーバーでオブジェクトの場所を変更した場合、リクエストを新しい場所にリダイレクトするようにウェブサーバーを構成できます。リダイレクトが構成された後、ビューワーがオブジェクトのリクエストを最初に送信したときに、ディストリビューションはリクエストをオリジンに送信し、オリジンはリダイレクトで応答します (例: `302 Moved Temporarily`)。ディストリビューションはリダイレクトをキャッシュし、ビューワーにリダイレクトを返します。ディストリビューションはリダイレクトに従いません。

リクエストを以下のどちらかの場所にリダイレクトするようにウェブサーバーを構成できます。

- オリジンサーバーのオブジェクトの新しい URL。ビューワーが新しい URL へのリダイレクトに従う場合、ビューワーはディストリビューションをバイパスし、オリジンに直接アクセスします。つまり、オリジンにあるオブジェクトの新しい URL にリクエストをリダイレクトしないことをお勧めします。
- オブジェクトの新しいディストリビューション URL。新しいディストリビューション URL を含むリクエストがビューワーから送信されると、ディストリビューションは、オリジンの新しい場所からオブジェクトを取得し、エッジロケーションにキャッシュした後、ビューワーにオブジェクトを返します。オブジェクトに対する以降のリクエストはエッジロケーションによって処理されます。これにより、オリジンのオブジェクトを要求するビューワーに関連するレイテンシーと負荷が回避されます。ただし、オブジェクトに対する新しいすべてのリクエストに、ディストリビューションへの 2 つのリクエストに対する料金がかかります。

転送エンコード

Lightsail chunked ディストリビューションはヘッダーの値のみをサポートします。`Transfer-Encoding`オリジンが `Transfer-Encoding: chunked` を返した場合、ディストリビューションは、エッジロケーションで受け取ったオブジェクトをクライアントに返し、そのオブジェクトをチャンク形式でキャッシュして以降のリクエストに備えます。

ビューワーが `Range GET` をリクエストし、オリジンは `Transfer-Encoding: chunked` を返した場合、ディストリビューションはリクエストされた範囲ではなくオブジェクト全体をビューワーに返します。

レスポンスのコンテンツ長を事前に決定できない場合は、チャンクエンコーディングを使用することをお勧めします。詳細については、「中断された TCP 接続」を参照してください。

Lightsail ディストリビューションをテストする

このガイドでは、Amazon Lightsail ディストリビューションがオリジンからコンテンツをキャッシュして提供しているかをテストする方法を紹介します。このテストは、登録したドメイン名をディストリビューションに追加した後に行う必要があります。ディストリビューションの詳細については、「コンテンツ配信ネットワークディストリビューション」を参照してください。

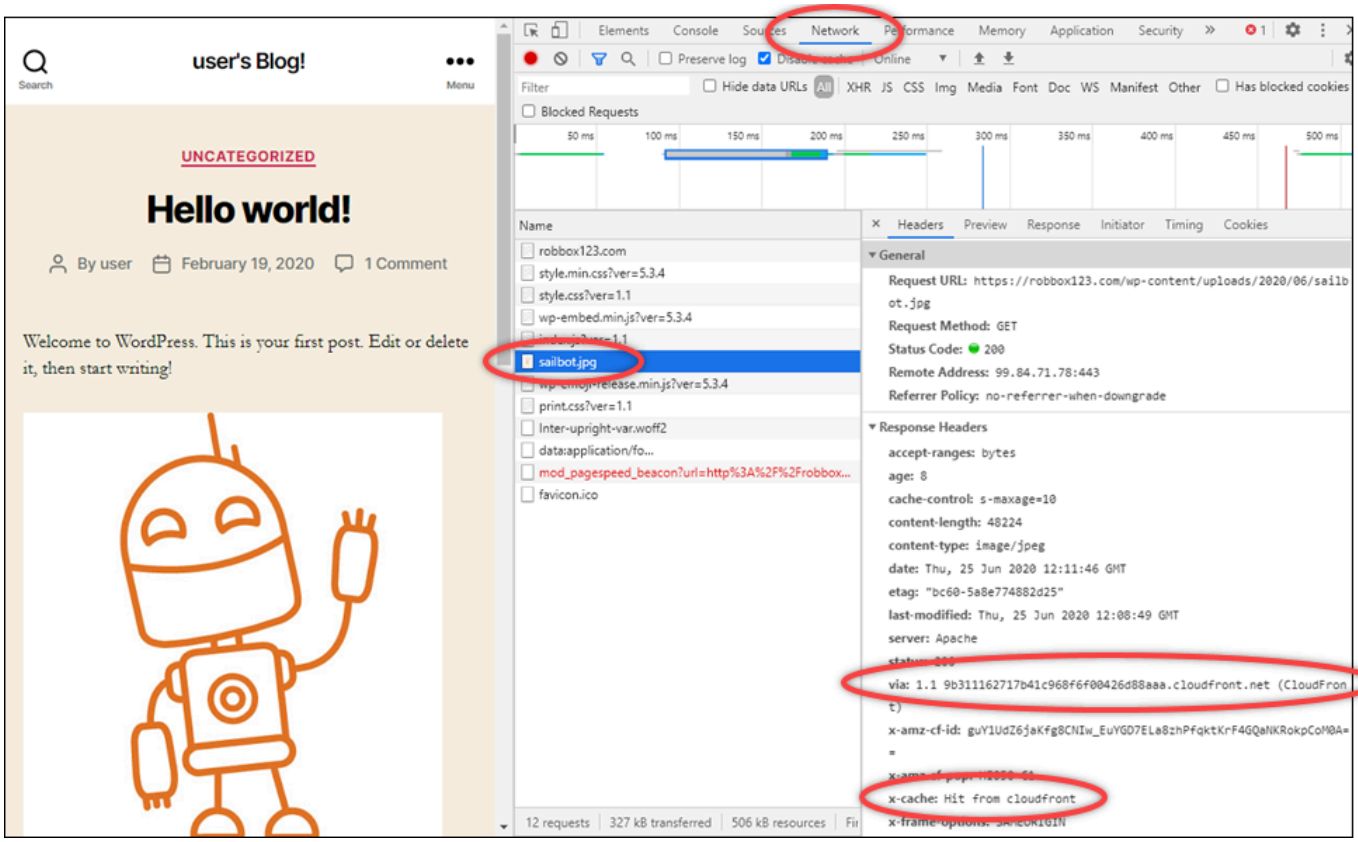
ディストリビューションをテスト

ディストリビューションをテストするには、以下の手順を行います。この手順では Chrome ウェブブラウザを使用していますが、他のブラウザでも同様の手順を使用することができます。

1. Chrome ウェブブラウザを開きます。
2. ブラウザウィンドウの右上にある Chrome メニュー を開き、[その他のツール]の [デベロッパーツール]を選択します。

   ショートカット Option + ⌘ + J (macOS の場合) 、Shift + Ctrl + J (Windows/Linux の場合) を使用することもできます。
3. デベロッパー ツールのペインで、[ネットワーク]タブを選択します。
4. ディストリビューションのドメインを参照します (例 : `https://www.example.com`)。

   Chrome のデベロッパーツールの[ネットワーク]タブには、ウェブサイトのオブジェクトのリストが表示されます。
5. イメージファイル (.jpg, .png, .gif) などの静的オブジェクトを選択します。
6. 表示される[ヘッダー] パネルで、`via` と `x-cache` のヘッダーの両方に CloudFront が記載されていることがわかります。これにより、ディストリビューションがオリジンからコンテンツをキャッシュして提供していることが確認されます。


user's Blog!
Search
Menu
UNCATEGORIZED
Hello world!
By user
February 19, 2020
1 Comment
Welcome to WordPress. This is your first post. Edit or delete it, then start writing!
Elements
Console
Network
Performance
Memory
Application
Security
Preserve log
Online
Filter
Hide data URLs
XHR JS CSS Img Media Font Doc WS Manifest Other
Has blocked cookies
Blocked Requests
Name
robbox123.com
style.min.css?ver=5.3.4
style.css?ver=1.1
wp-embed.min.js?ver=5.3.4
sailbot.jpg
print.css?ver=1.1
Inter-upright-var.woff2
data:application/fo...
favicon.ico
Headers
Preview
Response
Initiator
Timing
Cookies
General
Request URL: https://robbox123.com/wp-content/uploads/2020/06/sailbot.jpg
Request Method: GET
Status Code: 200
Remote Address: 99.84.71.78:443
Referrer Policy: no-referrer-when-downgrade
Response Headers
accept-ranges: bytes
age: 8
cache-control: s-maxage=10
content-length: 48224
content-type: image/jpeg
date: Thu, 25 Jun 2020 12:11:46 GMT
etag: "bc60-5a8e774882d25"
last-modified: Thu, 25 Jun 2020 12:08:49 GMT
server: Apache
via: 1.1 9b311162717b41c968f6f00426d88aaa.cloudfront.net (CloudFront)
x-amz-cf-id: guY1UdZ6jaKfg8CNIw_EuYGD7ELa8zhPfqktKrF4GQaNKRokpCoM0A==
x-cache: Hit from cloudfront
12 requests
327 kB transferred
506 kB resources

Amazon Lightsail のネットワーキングリソース

Lightsail ネットワーキングリソースは、ユーザーや外部サービスが Lightsail インスタンスに接続する方法を改善します。

ロードバランサー

ロードバランサーを作成すると、冗長性を追加したり、処理するトラフィック量を増やしたりできます。詳細については、「ロードバランサー」を参照してください。

静的 IP アドレス

静的 IP アドレスを作成すると、インスタンスを再起動しても同じ IP アドレスを保持できます。詳細については、「静的 IP アドレス」を参照してください。

Amazon Lightsail のリージョンとアベイラビリティーゾーン

Amazon Lightsail でリソースを作成するときは、ユーザーに最も近い AWS リージョン に作成します。たとえば、ブログのトラフィックが主にスイスで発生する場合は [フランクフルト] または [パリ] を選択します。

> **Note**
>
> DNS ゾーンはグローバルリソースです。それらのリソースは、米国東部 (バージニア北部) (us-east-1) リージョンにのみ作成されますが、任意の AWS リージョン のインスタンスを参照できます。

Lightsail は以下の AWS リージョン で利用できます。

- 米国東部 (オハイオ) (us-east-2)
- 米国東部 (バージニア北部) (us-east-1)
- 米国西部 (オレゴン) (us-west-2)
- アジアパシフィック (ムンバイ) (ap-south-1)
- アジアパシフィック (ソウル) (ap-northeast-2)
- アジアパシフィック (シンガポール) (ap-southeast-1)

- アジアパシフィック (シドニー) (ap-southeast-2)
- アジアパシフィック (東京) (ap-northeast-1)
- カナダ (中部) (ca-central-1)
- 欧州 (フランクフルト) (eu-central-1)
- 欧州 (アイルランド) (eu-west-1)
- 欧州 (ロンドン) (eu-west-2)
- 欧州 (パリ) (eu-west-3)
- 欧州 (ストックホルム) (eu-north-1)

SSH キーと Lightsail リージョン

Lightsail では、AWS リージョン でインスタンスを作成すると同時に、そのリージョンで[デフォルト]の SSH キーが作成されます。このデフォルトキーは、その特定のリージョンのインスタンスに接続するためにのみ使用できます。インスタンスがあるすべてのリージョンで同じキーを使用するには、独自のキーペアを作成し、それらのリージョンにアップロードします。または、既存のキーペアを各リージョンにアップロードします。

詳細については、「SSH のキーペア」を参照してください。

Lightsail リージョンを使用するためのヒント

各 AWS リージョン は、他の AWS リージョン と完全に分離されるように設計されています。これにより、最大限の耐障害性と安定性が達成されます。

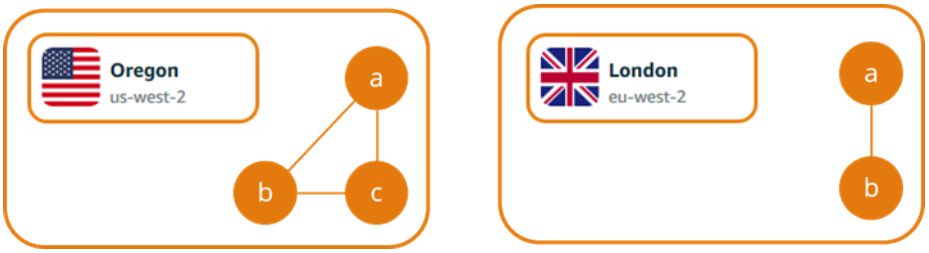
リージョン間のすべての通信は、パブリックインターネットを通して行われます。したがって、適切な暗号方式を使用してデータを保護する必要があります。リージョン間のデータ転送には料金がかか

ることに注意してください。詳細については、「Amazon EC2 料金表 - データ転送」を参照してください。

AWS Command Line Interface (AWS CLI) または API オペレーションを使用して Lightsail インスタンスを操作するときは、そのリージョンエンドポイントを指定する必要があります。AWS CLI コマンドで `--region` オプションを使用して `us-east-1` を指定し、DNS ゾーンおよびネットワークリソースに関する情報を取得します。AWS CLI の `--region` オプションの使用方法の詳細については、『リファレンス』の「AWS CLI汎用オプション」を参照してください。

Lightsail アベイラビリティーゾーン

アベイラビリティーゾーンは、物理的に独立した独自のインフラストラクチャで実行されているデータセンターの集合です。アベイラビリティーゾーンは、高度な信頼性を実現できるよう設計されています。発電機や冷却装置などの一般的な障害発生点は、アベイラビリティーゾーン間では共有されていません。アベイラビリティーゾーンは物理的にも離れているため、火災、竜巻、洪水などの極度の災害であっても、影響を受けるのはその発生場所にある単一のアベイラビリティーゾーンのみです。



各 AWS リージョン には 複数の孤立したアベイラビリティーゾーンがあり、そこはリージョン名に続く文字で示されます (例: `us-east-2`*a*)。一度に Lightsail インスタンスを作成できるのは 1 つのアベイラビリティーゾーンだけです。インスタンスを作成した時点では、一部のアベイラビリティーゾーンが表示されないことがあります。アベイラビリティーゾーンのリストが全く表示されていない場合は、前のステップで選択したリージョンを確認してください。

アベイラビリティーゾーンと Lightsail アプリケーション

別のアベイラビリティーゾーンでもインスタンスを起動することにより、1 つの場所で障害が発生してもアプリケーションを保護できます。

複数のアベイラビリティーゾーンで利用できるインスタンスを作成するには、最初にインスタンスのスナップショットを作成します。次に、作成したスナップショットから新しいインスタンスを作成する際に、別のアベイラビリティーゾーンを選択します。

詳細については、「Amazon EC2 ユーザーガイド」の「AWS リージョン とアベイラビリティゾーン」を参照してください。

Amazon Lightsail インスタンスで E メールサーバーの逆引き DNS を設定する

E メールサーバーでは、ドメインネームシステム (DNS) 逆引き参照を使用して、メッセージの発信元を追跡し、それがスパムや悪意のあるメッセージではないことを確認します。DNS 逆引き参照は、IP アドレスのドメイン名を返します。これは、ドメインの IP アドレスを返す DNS 前方参照の反対です。

たとえば、IP アドレス `192.168.1.2` の DNS 逆引き参照がサブドメイン `mail.example.com` を返し、サブドメイン `mail.example.com` の DNS 前方参照が IP アドレス `192.168.1.2` を返すと、IP アドレス `192.168.1.2` の逆引き DNS は前方確認されます。詳細については、Wikipedia の「Forward-confirmed reverse DNS」を参照してください。

Amazon Lightsail インスタンスの逆引き DNS を設定するには、前提条件を満たした上で、アウトバウンドメッセージングのクォータの削除リクエストを AWS Support に送信します。これらのステップを以下のセクションで示します。

前提条件

逆引き DNS を設定するには、次の前提条件を以下の順に実行します。

1. E メールサーバーとして使用する Lightsail インスタンスを作成します。詳細については、「インスタンスを作成する」を参照してください。
2. 逆引き DNS レコードとして使用する静的 IP を作成し、これを実行中のインスタンスにアタッチします。詳細については、「静的 IP を作成してインスタンスにアタッチする」を参照してください。

   > **Important**
   >
   > インスタンスの初回作成時に割り当てられるデフォルトのパブリック IP を逆引き DNS として使用することはできません。インスタンスのデフォルトのパブリック IP は、インスタンスの停止や開始に伴って変わるためです。

3. ドメインの DNS ゾーンで、サブドメイン (`mail.example.com` など) をポイントするエイリアスレコード (A レコード) を、実行中のインスタンスの静的 IP アドレスに追加します。このサブドメインは、静的 IP アドレスの DNS 逆引き参照を実行したときに返されます。詳細については、「DNS ゾーンを作成してドメインの DNS レコードを管理する」を参照してください。

> **Note**
>
> ドメインの DNS レコードの管理を Lightsail に引き渡すことをお勧めします。これにより、ドメインなどのすべてのリソースを 1 つの場所 (Lightsail コンソール) で管理できます。詳細については、「DNS ゾーンを作成してドメインの DNS レコードを管理する」を参照してください。

4. 変更がインターネットの DNS を通じて伝播されるまで待ちます。次に、AWS Support に対して逆引き DNS の設定リクエストを送信します。

AWS Support に逆引き DNS の設定リクエストを送信する

セキュリティ上の理由から、Lightsail では、ポート 25 を介したアウトバウンドメッセージがデフォルトで制限されます。ただし、このクォータをアカウントから削除するリクエストを AWS Support に送信し、静的 IP の逆引き DNS を設定できます。

AWS Support にリクエストを送信するには

1. Lightsail コンソールに AWS アカウントのルートユーザーとしてサインインします。

> **Important**
>
> リクエストは AWS アカウントのルートユーザーを使用して送信する必要があります。AWS アカウントのルートユーザーの詳細については、「AWS アカウントのルートユーザー」を参照してください。

2. E メール送信制限解除リクエストフォームに移動し、以下の必須情報を入力します。

> **Note**
>
> このフォームは、Elastic IP (EIP) や EC2 インスタンスなどの Amazon Elastic Compute (EC2) リソースを参照します。ただし、静的 IP や Lightsail インスタンスなどの Lightsail リソース用のフォームを使用することもできます。

- E メールアドレス – リクエストに関するメッセージを受信できる E メールアドレスを入力します。このテキストボックスには、アカウントの E メールアドレスが事前設定されます。

- ユースケースの説明 - E メールのクォータの削除をリクエストする理由を入力します。
- Elastic IP アドレス – このガイドの前提条件のステップ 2 でインスタンスにアタッチした静的 IP アドレスを入力します。静的 IP アドレスを 2 つまで入力できます。
- EIP の逆引き DNS レコード – このガイドの前提条件のステップ 3 で定義したサブドメインを入力します。このドメインは、DNS 逆引き参照を実行したときに返されます。

3. 終了したら、[送信] を選択します。

   AWS Support でリクエストが受理されると、静的 IP アドレスが DNS 逆引き参照で前方確認されます。

   後で Lightsail アカウントから静的 IP アドレスを削除する場合は、逆引き DNS 設定の削除リクエストを AWS Support に送信する必要があります。逆引き DNS 設定が削除されると、Lightsail コンソールを使用して Lightsail アカウントから静的 IP アドレスを削除できます。詳細については、「静的 IP を削除する」を参照してください。

Amazon Lightsail の外部の AWS リソースを使用するために Amazon VPC ピア接続をセットアップする

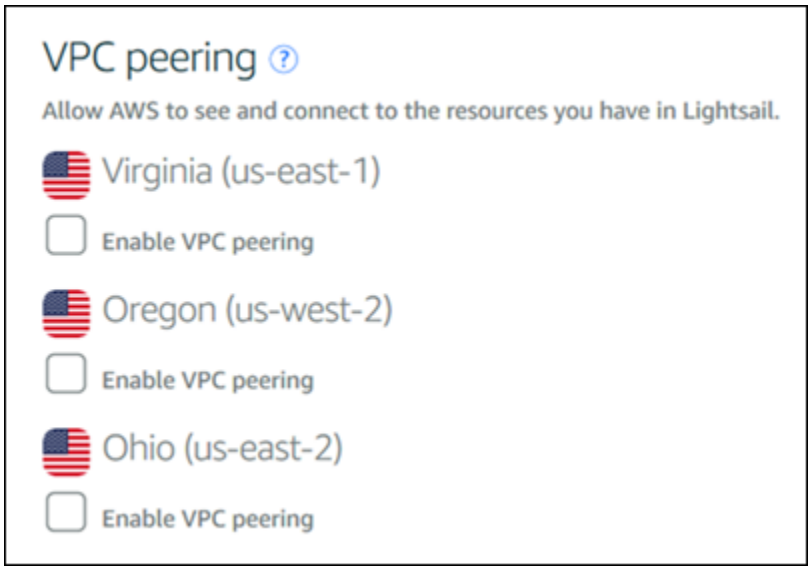
Lightsail では、仮想プライベートクラウド (VPC) ピア接続により、Amazon RDS データベースなどの AWS リソースに接続できます。VPC は、自分の AWS アカウント専用の仮想ネットワークです。Lightsail 内で作成するすべてのものは VPC 内に存在し、Lightsail VPC を Amazon VPC に接続できます。

Amazon S3、Amazon CloudFront、Amazon DynamoDB などの一部の AWS リソースでは、VPC ピアリングを有効にする必要はありません。

> **Note**
>
> Lightsail で VPC ピア接続を有効にするには、デフォルト Amazon VPC が必要です。デフォルトの Amazon VPC がない場合は作成できます。詳細については、「Amazon VPC ユーザーガイド」の「デフォルトの VPC を作成する」を参照してください。
> AWS リージョン は相互に隔離されているため、VPC も VPC を作成したリージョンで隔離されています。Lightsail リソースが存在する各リージョンで VPC ピア接続を有効にする必要があります。

デフォルトの Amazon VPC の準備が整ったら、以下の手順に従って、Lightsail VPC と Amazon VPC をピア接続します。

1. Lightsail コンソールの上部にあるナビゲーションメニューで [アカウント] を選択します。
2. ドロップダウンから [アカウント] を選択します。
3. [Advanced] (アドバンス) タブを選択します。
4. 有効にする AWS リージョン リージョンで [VPC ピアリングの有効化] を選択します。



ピア接続に失敗した場合は、VPC ピア接続を再度有効にしてみてください。それでも失敗する場合は、AWS カスタマーサポートにお問い合わせください。

ピアリング要求が正常に終了すると、ピア接続が AWS アカウントに作成されます。ナビゲーションペインで、[Amazon VPC ダッシュボード] を選択し、[ピア接続] を選択して、作成されたピア接続を表示します。

Amazon VPC の詳細については、「Amazon VPC ユーザーガイド」の「VPC とサブネット」を参照してください。

Amazon Lightsail の IP アドレス

IP アドレスを使用して、Lightsail インスタンスおよびその他の Lightsail リソースと通信できます。たとえば、インスタンスのパブリック IP アドレスを使用して、インスタンスのネットワークステータスを確認し (PING を使用)、インスタンスへの SSH 接続を確立し、カスタムドメイン名からインスタンスにトラフィックをルーティングできます。Lightsail リソースの IP アドレスでできることは他にも多数あります。

Lightsail インスタンス、コンテナサービス、ロードバランサーは、IPv4 と IPv6 の両方のアドレス指定プロトコルをサポートします。こういったリソースでは、デフォルトで IPv4 アドレス設定プロトコルを使用します。この動作を無効にすることはできません。オプションで、インスタンス、コンテナサービス、ロードバランサーの IPv6 を有効にできます。

このガイドでは、Lightsail の IP アドレスについて知っておくべきことについて説明します。

目次

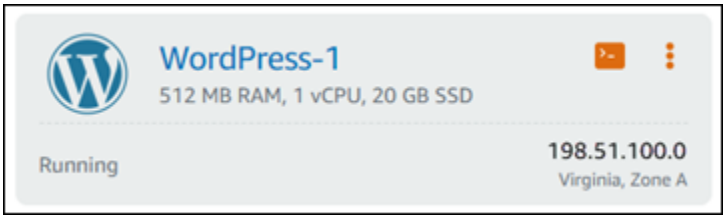
インスタンスのプライベートとパブリックの IPv4 アドレス

Lightsail インスタンスを作成すると、パブリック IPv4 アドレスとプライベート IPv4 アドレスが割り当てられます。パブリック IP アドレスはインターネットにアクセスでき、プライベート IP アドレスは同じ の Lightsail アカウントのリソースにのみアクセスできますAWS リージョン。

> **Note**
>
> インスタンスのプライベート IP アドレスは、同じ AWS リージョン内の他の AWS リソースにアクセスできますが、VPC ピアリングを有効にする場合は Lightsail アカウント外にあります。詳細については、「Lightsail の外部で AWS リソースと連携するように Amazon VPC ピアリングをセットアップする」を参照してください。

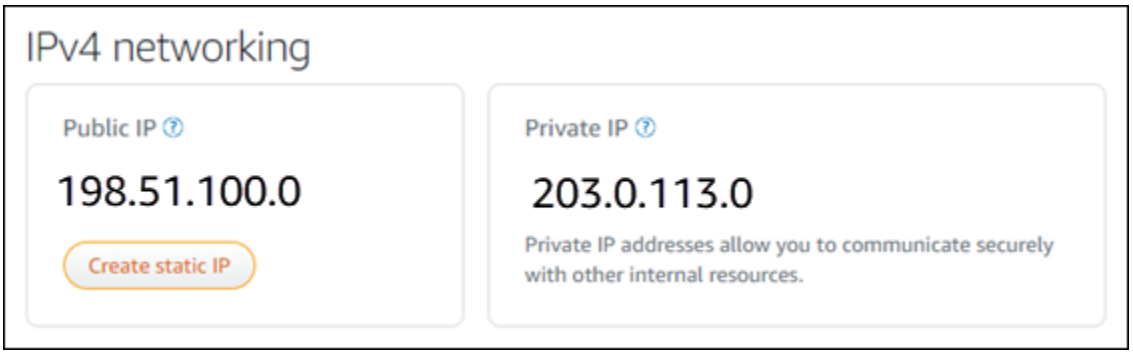
インスタンスの IP アドレスは、Lightsail コンソールの次の領域に表示されます。

- 次の例は、Lightsail ホームページ上のインスタンスのパブリック IP アドレスを示しています。



- 次の例は、インスタンス管理ページのヘッダー領域にあるインスタンスのパブリックとプライベート IP アドレスを示しています。

- 次の例は、インスタンス管理ページの [ネットワーク] タブにあるインスタンスのパブリック IP アドレスとプライベート IP アドレスを示しています。



インスタンスの IPv4 アドレスを使用する場合は、以下の点に注意してください。

- インスタンスのパブリック IP アドレスは変わることがあります。インスタンスに静的 IP を添付することで、変更されない IP アドレスを割り当てます。詳細については、このガイドのインスタンスの静的 IP アドレスセクションを参照してください。
- Lightsail はデフォルトで IPv4 アドレスを使用します。ただし、2021 年 1 月 12 日より前に作成された一部の Lightsail リソースでは、オプションで IPv6 を有効にできます。2021 年 1 月 12 日以降に作成されたリソースでは、IPv6 がデフォルトで有効になっています。詳細については、このガイドのインスタンス、コンテナーサービス、CDN ディストリビューション、およびロードバランサー用の IPv6 セクションを参照してください。
- インスタンスのファイアウォールにルールを追加して、インスタンスに接続できるトラフィックを制御できます。詳細については、「インスタンスのファイアウォール」を参照してください。

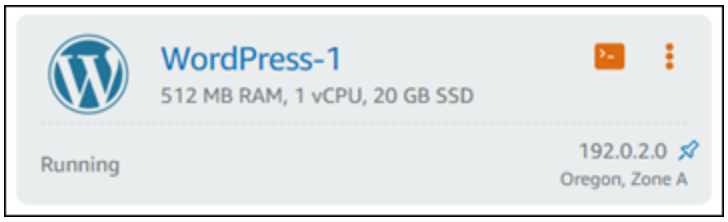
インスタンスの静的 IPv4 アドレス

インスタンスの作成時にインスタンスに割り当てられるデフォルトの公開 IPv4 アドレスは、インスタンスを停止してまた開始すると変更されます。必要に応じて、静的 IPv4 アドレスをインスタンスに作成し、アタッチできます。静的 IPv4 アドレスは、インスタンスのデフォルトの公開 IPv4 アド

レスに置き換えられ、インスタンスを停止して起動しても変わりません。1 つの静的 IP を 1 つのインスタンスにアタッチできます。詳細については、「静的 IP を作成してインスタンスにアタッチする」を参照してください。

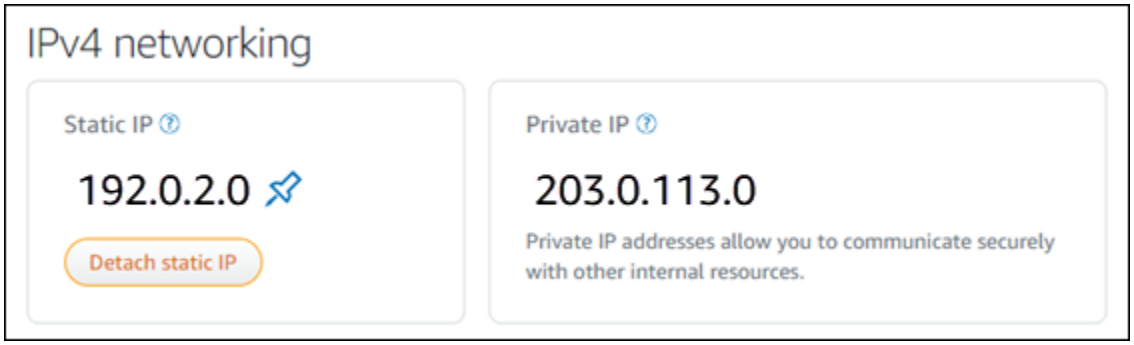
静的 IP を作成してインスタンスにアタッチすると、Lightsail コンソールの次の領域に表示されます。

- 次の例は、Lightsail ホームページ上のインスタンスの静的 IP アドレスを示しています。画鋲アイコンは、パブリック IP アドレスが静的であることを示します。

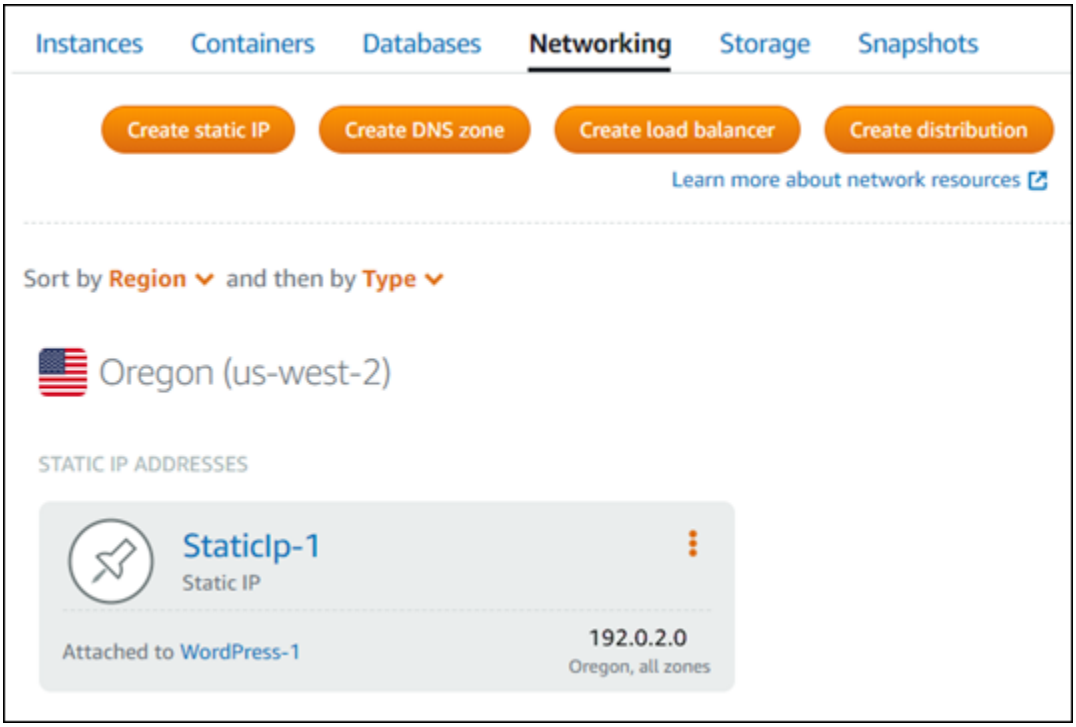


- 次の例は、インスタンス管理ページのヘッダー領域にあるインスタンスの静的 IP アドレスを示しています。画鋲アイコンは、パブリック IP アドレスが静的であることを示します。

- 次の例は、インスタンス管理ページの [ネットワーク] タブにあるインスタンスの静的 IP アドレスを示しています。デフォルトのパブリック IP アドレスは表示されなくなり、静的 IP アドレスに置き換えられました。画鋲アイコンは、パブリック IP アドレスが静的であることを示します。



- 次の例に示すように、Lightsail ホームページのネットワークタブに移動することで、作成したすべての静的 IPs を表示できます。



インスタンス、コンテナーサービス、CDN ディストリビューション、およびロードバランサー用の IPv6

IPv6 は、2021 年 1 月 12 日以降に作成された Lightsail インスタンス、コンテナサービス、CDN ディストリビューション、およびロードバランサーに対してデフォルトで有効になっています。オプションとして、2021 年 1 月 12 日より前に作成されたリソースの IPv6 を有効にすることもできます。特定のリソースに対して IPv6 を有効にすると、Lightsail はそのリソースに IPv6 アドレスを自動的に割り当てます。IPv6 アドレスを自分で選択または指定することはできません。詳細については、「IPv6 を有効化または無効化する」を参照してください。

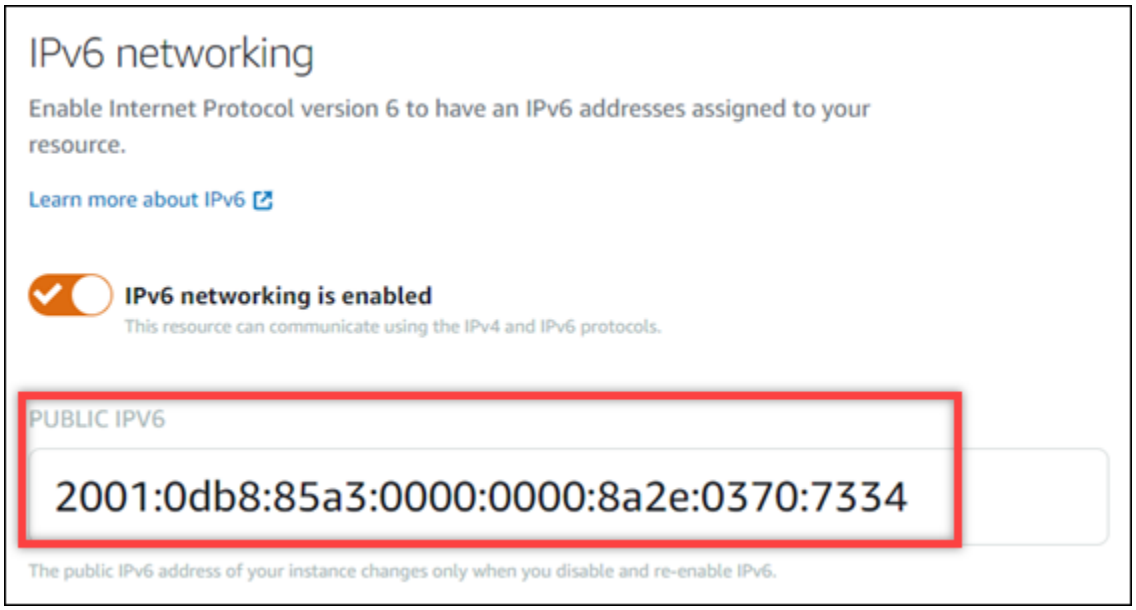
IPv6-onlyインスタンスを作成することもできます。IPv6-onlyインスタンスは IPv6 経由でのみパブリックに通信でき、パブリック IPv4 アドレスはありません。詳細については、「Lightsail IPv6-only インスタンスプラン」を参照してください。

インスタンスの IPv6 アドレスは、Lightsail コンソールの次の領域に表示されます。

- 次の例は、Lightsail ホームページ上のインスタンスの IPv6 アドレスを示しています。

- 次の例は、リソースの管理ページのヘッダー領域にあるリソースの IPv6 アドレスを示しています。

- 次の例は、リソース管理ページの [ネットワーク] タブにあるリソースの IPv6 アドレスを示しています。



リソースで IPv6 を有効にして使用するときは、次の点に注意してください。

- リソースに対して IPv6 を有効にすると、リソースは IPv4 および IPv6 (デュアルスタックモード) 経由で、または IPv4 経由でのみ通信できます。 IPv6

- リソースに対して IPv6 を有効にすると、Lightsail はそのリソースに IPv6 アドレスを自動的に割り当てます。IPv6 アドレスを自分で選択または指定することはできません。リソースに対して IPv6 を有効にすると、IPv6 プロトコル経由のネットワークトラフィックの受け入れが開始されます。
- インスタンスの IPv6 アドレスは、インスタンスを停止してまた開始しても保持されます。インスタンスを削除するか、インスタンスに対して IPv6 を無効にした場合にのみ解除されます。これらのアクションのいずれかを実行した後は、同じ IPv6 アドレスを取得することはできません。
- インスタンスに割り当てられるすべての IPv6 アドレスは公開されているため、インターネット経由で接続することができます。インスタンスに割り当てられるプライベート IPv6 アドレスは存在しません。
- インスタンスの IPv4 アドレスと IPv6 アドレスは互いに独立しています。従って、 IPv4 と IPv6 のインスタンスファイアウォールのルールは個別に設定する必要があります。詳細については、「インスタンスのファイアウォール」を参照してください。
- Lightsail で使用可能なすべてのインスタンスブループリントが、IPv6 が有効になっているときに IPv6 に自動的に設定されるわけではありません。次の設計図を使用するインスタンスでは、IPv6 を有効にした後で、追加の設定手順が必要になります。
  - cPanel — 詳細については、「cPanel インスタンスに IPv6 を設定する」を参照してください。
  - Debian 8 — 詳細については、「Debian 8 インスタンスに IPv6 を設定する」を参照してください。
  - GitLab – 詳細については、 GitLab 「インスタンスの IPv6 を設定する」を参照してください。
  - Nginx — 詳細については、「Nginx インスタンスに IPv6 を設定する」を参照してください。
  - Plesk — 詳細については、「Plesk インスタンスに IPv6 を設定する」を参照してください。
  - Ubuntu 16 — 詳細については、「Ubuntu 16 インスタンスに IPv6 を設定する」を参照してください。

> **Note**
>
> PrestaShop は現在、IPv6 アドレスをサポートしていません。インスタンスの IPv6 を有効にすることはできますが、 PrestaShop ソフトウェアは IPv6 ネットワークを介したリクエストに応答しません。

Amazon Lightsail の静的 IP アドレス

静的 IP アドレスは固定のパブリック IP アドレスであり、インスタンスまたはその他リソースに割り当ておよび再割り当てできます。静的 IP アドレスを未設定の場合、インスタンスを停止または再起動するたびに、Lightsail で新しいパブリック IP アドレスが割り当てられます。

> **Important**
>
> 最初に静的 IP アドレスを作成してインスタンスに添付せずにインスタンスを停止または再起動すると、インスタンスの再起動時に IP アドレスが失われます。インスタンスが常に同じパブリック IP アドレスを持つように、静的 IP アドレスを作成してインスタンスにアタッチする必要があります。詳細については、「静的 IP アドレスを作成する」を参照してください。

目次

静的 IP アドレスを作成して Lightsail インスタンスにアタッチする

Amazon Lightsail インスタンスにアタッチされたデフォルトの動的パブリック IP アドレスは、インスタンスを停止して再起動するたびに変更されます。パブリック IP アドレスが変わらないようにするには、静的 IP アドレスを作成してインスタンスに接続します。その後、登録したドメイン名がインスタンスをポイントするように設定すると、インスタンスを停止して再起動するたびに、ドメインの DNS レコードを更新する必要がなくなります。1 つの静的 IP を 1 つのインスタンスにアタッチできます。詳細については、「静的 IP アドレス」を参照してください。

前提条件

Lightsail で実行されているデュアルスタックインスタンスが少なくとも 1 つ必要です。インスタンスを作成するには、「インスタンスを作成する」を参照してください。

静的 IP アドレスの作成およびインスタンスへの割り当て

以下の手順に従って、新しい静的 IP アドレスを作成し、Lightsail のインスタンスにアタッチします。

1. https://lightsail.aws.amazon.com/ で Lightsail コンソールにサインインします。

2. Lightsail ホームページで、ネットワーク を選択します。
3. [静的 IP の作成] を選択します。
4. 静的 IP を作成する AWS リージョン を選択します。

> **Note**
>
> 静的 IP アドレスは、同じリージョンのインスタンスにのみアタッチできます。

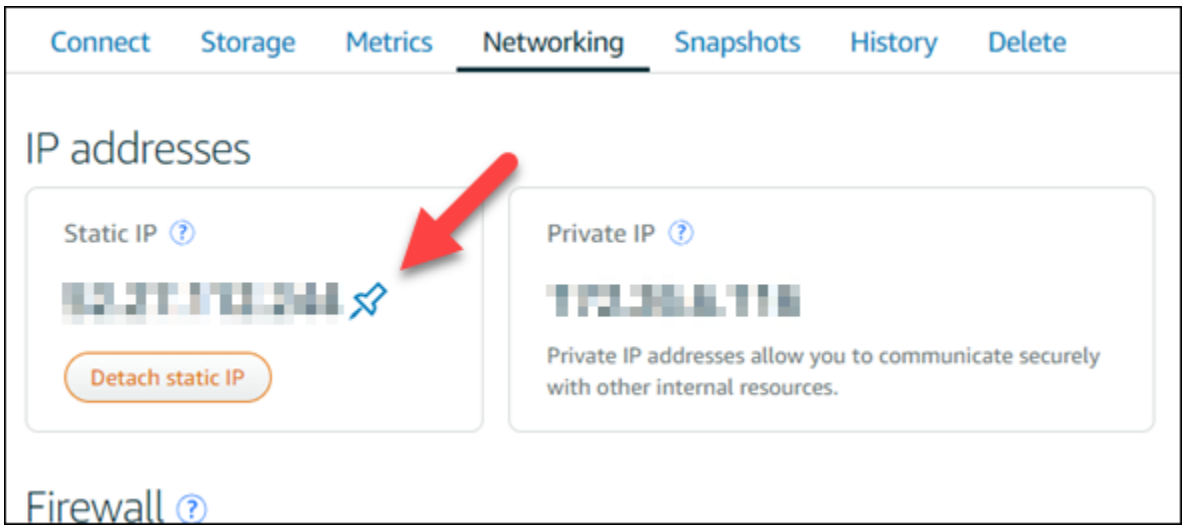
5. 静的 IP をアタッチする Lightsail リソースを選択します。
6. 静的 IP の名前を入力します。

   リソース名:

   - AWS リージョン Lightsail アカウントの各 内で一意である必要があります。
   - 2〜255 文字を使用する必要があります。
   - 先頭と末尾は英数字または数字を使用する必要があります。
   - 英数字、数字、ピリオド、ダッシュ、アンダースコアを使用することができます。
7. [Create(作成)] を選択します。

   ホームページに移動すると、管理できる静的 IP アドレスを確認できます。

   また、インスタンス管理ページでは、[ネットワーキング] タブのパブリック IP アドレスの横に青い押しピンが表示されます。これは、現在の IP アドレスが静的であることを示します。



詳細については、「パブリック IP アドレスとプライベート IP アドレス」を参照してください。

Lightsail で静的 IP アドレスを削除する

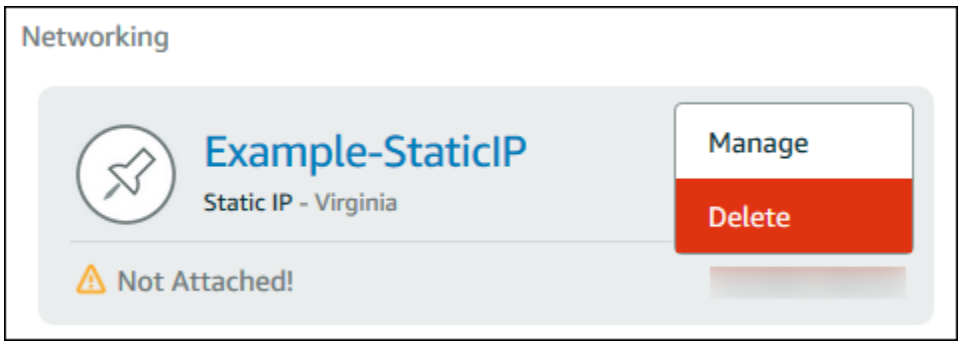
Amazon Lightsail アカウントAWS リージョンでは、 ごとに最大 5 つの静的 IPs を作成できます。静的 IP アドレスが割り当てられているインスタンスを削除しても、静的 IP アドレスはアカウントに残ります。静的 IP アドレスが不要になった場合は、Lightsail コンソールまたは AWS Command Line Interface () を使用して削除できますAWS CLI。このガイドでは、Lightsail アカウントから静的 IP アドレスを削除する方法について説明します。静的 IP の詳細については、「IP アドレス」を参照してください。

> **Important**
>
> 静的 IP を削除すると、その静的 IP が Lightsail アカウントから完全に削除されます。インスタンスなど、その静的 IP を使用するリソースは影響を受けます。静的 IP を削除すると、元に戻すことはできません。

Lightsail コンソールを使用して静的 IP を削除する

Lightsail コンソールを使用して静的 IP を削除するには、以下の手順を実行します。

1. Lightsail コンソールにサインインします。
2. Lightsail ホームページで、ネットワーク を選択します。
3. ネットワークページで、削除する静的 IP アドレスの横にある縦楕円 (⋮) アイコンを選択し、削除を選択します。



AWS CLI を使用した静的 IP の削除

AWS CLI を使用して静的 IP を削除するには、以下の手順を完了してください。Lightsail アカウントから静的 IP を削除するコマンドは ですrelease-static-ip。静的 IP を作成する場合、実際は、静的 IP を割り当てています。したがって、実際には静的 IP を削除するのではなく、解放することになります。

前提条件

まず、まだ AWS CLI をインストールしていない場合はインストールする必要があります。詳細については、「AWS Command Line Interface のインストール」を参照してください。AWS CLI を設定する必要があります。

解放する静的 IP の名前が必要になります。この名前は、AWS CLI コマンド `get-static-ips` を使用して取得できます。

1. 次のコマンドを入力します。

```
aws lightsail get-static-ips
```

次のような出力が表示されます。

```
{
    "staticIps": [
        {
            "name": "Example-StaticIP",
            "resourceType": "StaticIp",
            "attachedTo": "MyInstance",
            "arn": "arn:aws:lightsail:us-east-2:123456789101:StaticIp/5282f35e-
c720-4e5a-1234-12345EXAMPLE",
            "isAttached": true,
```

```
            "ipAddress": "192.0.2.0",
            "createdAt": 1489750629.026,
            "location": {
                "availabilityZone": "all",
                "regionName": "us-east-2"
            }
        },
        {
            "name": "my-other-static-ip",
            "resourceType": "StaticIp",
            "arn": "arn:aws:lightsail:us-east-2:123456789101:StaticIp/
f5885e14-8984-49e5-1234-12345EXAMPLE",
            "isAttached": false,
            "ipAddress": "192.0.2.2",
            "createdAt": 1483653597.815,
            "location": {
                "availabilityZone": "all",
                "regionName": "us-east-2"
            }
        }
    ]
}
```

2. 解放する静的 IP の [名前] フィールドの値を選択し、次のステップで使用できるようにその名前をメモします。

   たとえば、その値をクリップボードにコピーできます。

3. 次のコマンドを入力します。

   ```
   aws lightsail release-static-ip --static-ip-name StaticIpName
   ```

   コマンドで、 を静的 IP の名前*StaticIpName*に置き換えます。

   成功すると、次のような出力が表示されます。

   ```
   {
       "operations": [
           {
               "status": "Succeeded",
               "resourceType": "StaticIp",
               "isTerminal": true,
               "statusChangedAt": 1489860944.19,
   ```

```
            "location": {
                "availabilityZone": "all",
                "regionName": "us-east-2"
            },
            "operationType": "ReleaseStaticIp",
            "resourceName": "Example-StaticIP",
            "id": "92a2f0d2-eef2-4e6f-1234-12345EXAMPLE",
            "createdAt": 1489860944.19
        }
    ]
}
```

Amazon Lightsail で IPv6 を有効または無効にする

IPv6 は、2021 年 1 月 12 日以降に作成された Lightsail インスタンス、コンテナサービス、CDN ディストリビューション、およびロードバランサーに対してデフォルトで有効になっています。オプションとして、2021 年 1 月 12 日より前に作成されたリソースの IPv6 を有効にすることもできます。このガイドでは IPv6 を有効または無効にする方法について説明します。IPv6 アドレスの詳細については、「IP アドレス」を参照してください。

目次

IPv6 に関する考慮事項

IPv6 は 2021 年 1 月 12 日Lightsailに実装されました。そのため、次のガイドラインに従って、一部のリソースに対して IPv6 を手動で有効または無効にする必要がある場合があります。

- 1 月 12 日より前に作成されたインスタンス、CDN ディストリビューション、およびロードバランサーは、有効にするまで IPv6 は無効になっています。ただし、1 月 12 日より後に作成されたインスタンス、CDN ディストリビューション、およびロードバランサーは作成時に IPv6 が有効になります。
- 1 月 12 日より前または後に作成されたコンテナサービスでは IPv6 が有効になっています。
- IPv6 は、インスタンス、CDN ディストリビューション、およびロードバランサーに対していつでも手動で有効または無効にできます。コンテナサービスに対して無効にすることはできません。

IPv6 を有効にして使用する場合、以下の点に注意してください。

- リソースに対して IPv6 を有効にすると、リソースは IPv4 のみ、または IPv4 と IPv6 (デュアルスタックモード) で通信できます。
- インスタンスに対して IPv6 を有効にすると、Lightsail はそのインスタンスに IPv6 アドレスを自動的に割り当てます。IPv6 アドレスを自分で選択したり指定することはできません。コンテナサービス、CDN ディストリビューション、またはロードバランサーに対して IPv6 を有効にすると、そのリソースは IPv6 経由でインターネットトラフィックを受け入れ始めます。
- インスタンスの IPv6 アドレスは、インスタンスを停止してまた開始しても保持されます。インスタンスを削除するか、インスタンスに対して IPv6 を無効にした場合にのみ解除されます。これらのアクションのいずれかを実行した後は、同じ IPv6 アドレスを取得することはできません。
- インスタンスに割り当てられるすべての IPv6 アドレスは公開されているため、インターネット経由で接続することができます。インスタンスに割り当てられるプライベート IPv6 アドレスは存在しません。
- インスタンスの IPv4 アドレスと IPv6 アドレスは互いに独立しています。従って、 IPv4 と IPv6 のインスタンスファイアウォールのルールは個別に設定する必要があります。詳細については、「インスタンスのファイアウォール」を参照してください。
- Lightsail のすべてのインスタンスの設計図は、IPv6 が有効なときに IPv6 用に自動的に構成されます。次の設計図を使用するインスタンスでは、IPv6 を有効にした後で、追加の設定手順が必要になります。
  - cPanel — 詳細については、「cPanel インスタンスに IPv6 を設定する」を参照してください。
  - Debian 8 — 詳細については、「Debian 8 インスタンスに IPv6 を設定する」を参照してください。
  - GitLab — 詳細については、「GitLab インスタンスに IPv6 を設定する」を参照してください。
  - Nginx — 詳細については、「Nginx インスタンスに IPv6 を設定する」を参照してください。
  - Plesk — 詳細については、「Plesk インスタンスに IPv6 を設定する」を参照してください。
  - Ubuntu 16 — 詳細については、「Ubuntu 16 インスタンスに IPv6 を設定する」を参照してください。

IPv6 を有効にする

インスタンス、CDN ディストリビューション、およびロードバランサーに対して IPv6 を有効にするには、以下の手順を実行してください。

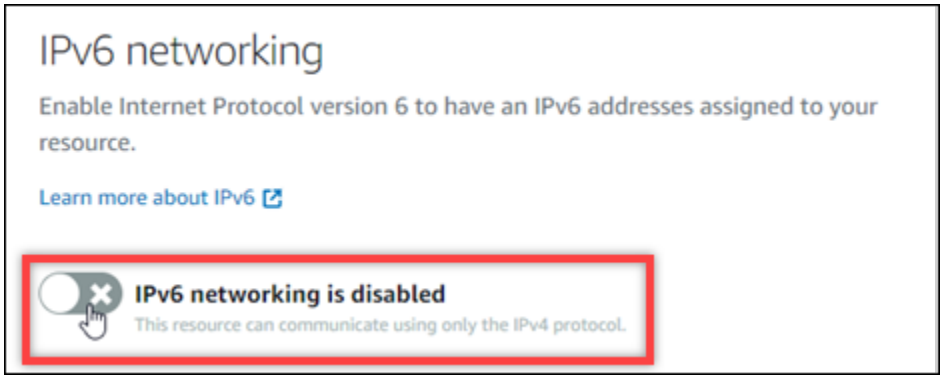
1. Lightsail コンソールにサインインします。

2. IPv6 を有効にするリソースに応じて、次のいずれかの手順を実行してください。

   - インスタンスで IPv6 を有効にするには、Lightsail ホームページの [インスタンス] タブをクリックし、IPv6 を有効にするインスタンスの名前を選択します。
   - CDN ディストリビューションまたはロードバランサーで IPv6 を有効にするには、Lightsail ホームページの [ネットワーキング] タブをクリックし、IPv6 を有効にする CDN ディストリビューションまたはロードバランサーの名前を選択します。

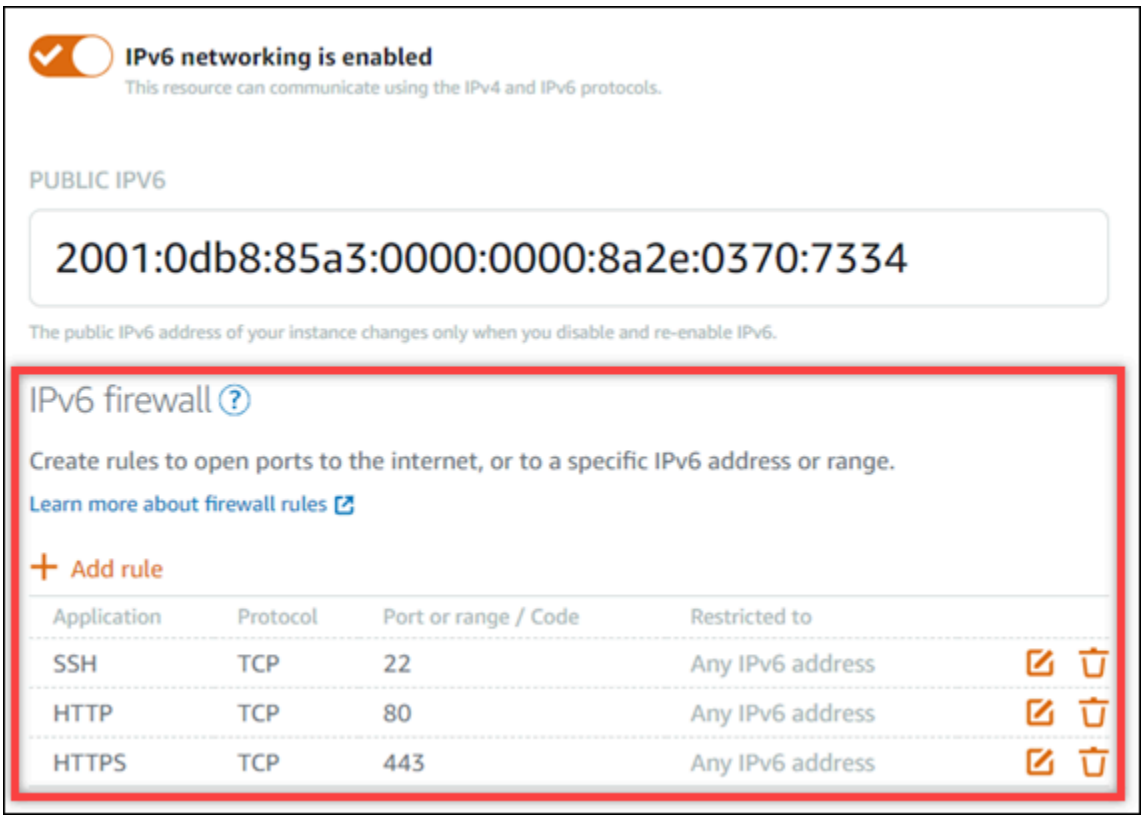
3. リソースの管理ページで [Networking] タブを選択します。

4. ページのIPv6 ネットワーキングセクションで、 IPv6 のリソースを有効にするトグルを選択します。



   リソースに対して IPv6 を有効にした後は、次の点に注意してください。

   - CDN ディストリビューションまたはロードバランサーに対して IPv6 を有効にすると、そのリソースは IPv6 トラフィックを受け入れ始めます。インスタンスに対して IPv6 を有効にすると、IPv6 アドレスが割り当てられ、次の例に示すように、 IPv6 ファイアウォールが使用可能になります。

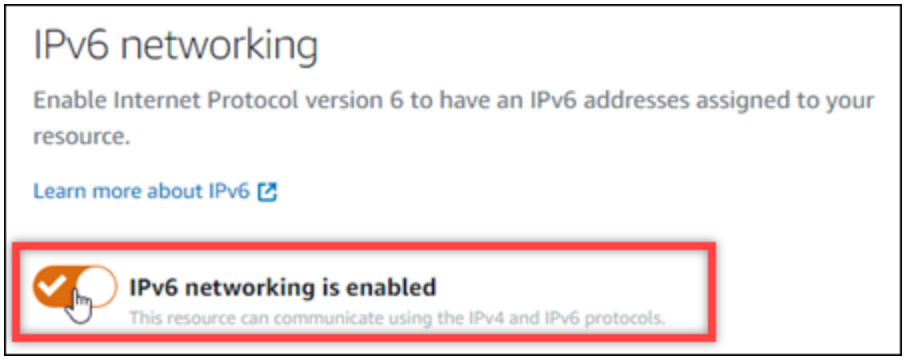


- 次のブループリントを使用するインスタンスでは、IPv6 を有効にした後、インスタンスが新しい IPv6 アドレスを認識できるように追加の手順が必要です。
  - cPanel — 詳細については、「cPanel インスタンスに IPv6 を設定する」を参照してください。
  - Debian 8 — 詳細については、「Debian 8 インスタンスに IPv6 を設定する」を参照してください。
  - GitLab — 詳細については、「GitLab インスタンスに IPv6 を設定する」を参照してください。
  - Nginx — 詳細については、「Nginx インスタンスに IPv6 を設定する」を参照してください。
  - Plesk — 詳細については、「Plesk インスタンスに IPv6 を設定する」を参照してください。
  - Ubuntu 16 — 詳細については、「Ubuntu 16 インスタンスに IPv6 を設定する」を参照してください。
- インスタンス、コンテナサービス、CDN ディストリビューション、またはロードバランサーのトラフィックを転送する登録済みのドメイン名を持つ場合、 IPv6 トラフィックをリソースにルーティングするために、ドメインの DNS に IPv6 アドレスレコード (AAAA) を作成するようにしてください。

IPv6 を無効にする

インスタンス、CDN ディストリビューション、およびロードバランサーの IPv6 を無効にするには、以下の手順を実行します。

1. Lightsail コンソールにサインインします。
2. IPv6 を無効にするリソースに応じて、次のいずれかの手順を実行します。
   - インスタンスの IPv6 を無効にするには、Lightsail ホームページの [インスタンス] タブをクリックし、IPv6 を無効にするインスタンスの名前を選択します。
   - CDN ディストリビューションまたはロードバランサーの IPv6 を無効にするには、Lightsail ホームページの [ネットワーキング] タブをクリックし、IPv6 を無効にする CDN ディストリビューションまたはロードバランサーの名前を選択します。
3. リソースの管理ページで [Networking] タブを選択します。
4. ページの IPv6 ネットワーキングセクションで、 IPv6 を無効にするトグルを選択してください。



Amazon Lightsail の SSL/TLS 証明書

Amazon Lightsail は SSL/TLS 証明書を使用して、Lightsail ロードバランサー、コンテンツ配信ネットワーク (CDN) ディストリビューション、およびコンテナサービスで使用できるカスタム (登録済み) ドメインを検証します。検証済みの証明書がこれらの Lightsail リソースのいずれかに添付されると、ドメインを介してそのリソースにルーティングされるトラフィックは、ハイパーテキスト転送プロトコルセキュア (HTTPS) を使用して暗号化されます。

Amazon Lightsail でトランスポート層セキュリティ (TLS) 証明書を作成して、Lightsail ロードバランサー、コンテンツデリバリー、ネットワークディストリビューション、コンテナサービスで使用するカスタム (登録済み) ドメインの暗号化されたウェブトラフィックを有効にすることができま

す。TLS は、Secure Socket Layer (SSL) の更新された、より安全なバージョンです。Lightsail のドキュメントとコンソールでは、これを SSL/TLS と呼んでいるのがわかります。

> **Note**
>
> ロードバランサー、CDN ディストリビューション、コンテナサービスにアタッチできる Lightsail 証明書は (ACM) サービスによって発行されます。 AWS Certificate Manager 2022 年 10 月 11 日以降、ロードバランサー、CDN ディストリビューション、コンテナサービス用に Lightsail を通じて取得したパブリック証明書はすべて、ACM が管理する複数の中間認証局 (ICA) または下位 CA のいずれかから発行されます。詳細については、「AWS セキュリティブログ」の「Amazon introduces dynamic intermediate certificate authorities」(Amazon が動的中間認証局を導入) を参照してください。

HTTPS を使用する理由

何よりも優先されるのはセキュリティです。HTTPS では、TLS を使用してデータが移動される、セキュリティの追加レイヤーが実現します。HTTPS 暗号化では、ウェブサーバーとクライアントのブラウザだけがトラフィックを復号化できるエンティティであるため、それらの間では内容が秘密に保たれます。HTTPS 接続では、クライアントがサーバーと交換するデータを別の当事者が変更できないため、HTTPS も安全です。

前述のセキュリティ上の利点を除き、HTTP に加えて HTTPS を使用する他の理由があります。たとえば、2014 年に Google は検索結果において安全なウェブサイトを上位にランク付けし始めました。つまり、HTTPS を使用するサイトは、HTTP しか使用していないサイト (他のすべては同じ) と比較して検索結果の上位にランク付けされます。

ランキングシグナルとしての HTTPS の詳細

プロセスの概要

Lightsail 証明書を使用するプロセスは簡単です。これには、次のステップが含まれます。

1. ロードバランサー、CDN ディストリビューション、コンテナサービスなどの Lightsail 証明書を使用できる Lightsail リソースを作成します。
2. Lightsail を使用してドメインの証明書を作成します。
3. ドメインの DNS に正規名 (CNME) レコードを追加して証明書を検証します
4. 検証済みの証明書を Lightsail リソースに添付します。

5. ドメインの DNS を変更して、トラフィックを Lightsail リソースにルーティングします。

証明書がリソースに添付された後、ドメインを介してそのリソースにルーティングされるトラフィックは、HTTPS を使用して暗号化されます。

ディストリビューションおよびコンテナサービスを使用した SSL/TLS 証明書を使用する

Lightsail ディストリビューションとコンテナサービスには HTTPS が必要です。これらのリソースのいずれかを作成すると、HTTPS はデフォルトで、リソースのデフォルトドメインに有効化されます (例: `https://123456abcdef.cloudfront.net/` はディストリビューション用、`https://container-service-1.123456abcdef.us-west-2.cs.amazonlightsail.com/` はコンテナサービス用)。登録したドメイン名 (例:`example.com`) をディストリビューションまたはコンテナサービスで使用する場合は、Lightsail SSL/TLS 証明書を作成し、ドメイン名で検証し、リソースでカスタムドメインを有効にする必要があります。ディストリビューションまたはコンテナサービスでカスタムドメインを有効にすると、ドメインの検証済み証明書もリソースに添付されます。

ディストリビューションでカスタムドメインと HTTPS を有効化するには、次のリンクをクリックします。

- ディストリビューションの SSL/TLS 証明書を作成する
- ディストリビューションの SSL/TLS 証明書を表示する
- ディストリビューションの SSL/TLS 証明書を表示する
- ディストリビューションのカスタムドメインを有効にする
- ドメインをディストリビューションにポイントする

ディストリビューションの詳細については、「コンテンツ配信ネットワークディストリビューション」を参照してください。

コンテナサービスでカスタムドメインと HTTPS を有効化するには、次のリンクをクリックします。

- コンテナサービス用の SSL/TLS 証明書を作成する
- コンテナサービスの SSL/TLS 証明書を検証する
- カスタムドメインを有効にして管理する

コンテナサービスの詳細については、「コンテナサービス」を参照してください。

ロードバランサーでの SSL/TLS 証明書の使用

Lightsail ロードバランサーを作成すると、ポート 80 がデフォルトで開かれ、通常の HTTP トラフィックを処理できます。ポート 443 で HTTPS トラフィックを有効にするには、SSL/TLS 証明書を作成し、ドメイン名で検証し、ロードバランサーにアタッチする必要があります。

ロードバランサーあたり最大 2 つの SSL/TLS 証明書を作成できます。ロードバランサーごとに、一度に 1 つの証明書のみ使用できます。有効な使用中の証明書をロードバランサーから削除した場合、そのロードバランサーは、別の有効な証明書をアタッチするまで、指定されたドメインの HTTPS トラフィックを処理できなくなります。

ロードバランサーで HTTPS の使用を開始するには、次のリンクをクリックします。

- ロードバランサーを作成してインスタンスをアタッチする
- SSL/TLS 証明書を作成する
- ドメインの所有権を検証する
- 検証された証明書をアタッチして HTTPS を有効にする

ロードバランサーの詳細については、「ロードバランサー」を参照してください。

Lightsail コンテナサービスの SSL/TLS 証明書

Lightsail コンテナサービス用の Amazon Lightsail TLS/SSL 証明書を作成できます。証明書を作成するときは、証明書のプライマリおよび代替ドメイン名を指定します。コンテナサービスのカスタムドメインを有効にして証明書を選択すると、コンテナサービスのカスタムドメインとして追加する証明書から最大 4 つのドメインを選択できます。ドメインの DNS レコードを更新してトラフィックをコンテナサービスに誘導すると、サービスはトラフィックを受け入れ、HTTPS を使用してコンテンツを提供します。アカウントに作成できる証明書の数にはクォータがあります。詳細については、「Lightsail のサービスクォータ」を参照してください。

SSL/TLS 証明書の詳細については、「コンテナサービス証明書」を参照してください。

前提条件

開始する前に、Lightsail コンテナサービスを作成する必要があります。詳細については、「コンテナサービスを作成する」および「コンテナサービス」を参照してください。

コンテナサービス用の SSL/TLS 証明書を作成する

コンテナサービス用の SSL/TLS 証明書を作成するには、以下の手順を実行します。

1. Lightsail コンソールにサインインします。
2. Lightsail コンソールのホームページで、[コンテナ] タブを選択します。
3. 証明書を作成するコンテナサービスの名前を選択します。
4. [Custom domains] (カスタムドメイン) のタブで、コンテナサービス管理ページを選択します。
5. ページ下部の [アタッチされた証明書] セクションまでスクロールします。

   他の Lightsail リソース用に作成された証明書や、使用中あるいは使用されていない証明書など、すべての証明書がページの [Attached certificates] (アタッチされた証明書) セクションに表示されます。
6. [証明書の作成] を選択します。
7. 証明書を識別する一意の名前を [Certificate name] (証明書の名前) テキストボックスに入力します。次に、[Continue] (続行する) を選択します。
8. 証明書とともに使用するプライマリドメイン名 (例: `example.com`) を、[Specify up to 10 domains or subdomains] (最大 10 個のドメインまたはサブドメインを指定) フィールドに入力します。
9. (オプション) 別のドメイン名 (例: www.example.com) を [Specify up to 10 domains or subdomains] (最大 10 個のドメインまたはサブドメインを指定) フィールドに入力します。

   証明書には最大 9 個の代替ドメインを追加できます。カスタムドメインを有効にし、サービスの証明書を選択した後、コンテナサービスでは最大 4 つの証明書ドメインを使用できます。
10. [証明書の作成] を選択します。

    証明書のリクエストが送信されると、新しい証明書のステータスは [証明書の検証試行中] に変更されます。この間、Lightsail は証明書の検証レコードをプライマリドメインの DNS に追加しようとします。しばらくすると、ステータスは [有効] に変化します。

自動検証に失敗した場合、コンテナサービスとともに使用する前に、ドメインで証明書を検証する必要があります。詳細については、「コンテナサービスの SSL/TLS 証明書を検証する」を参照してください。

トピック

- Lightsail コンテナサービスの SSL/TLS 証明書を検証する
- Lightsail コンテナサービスの SSL/TLS 証明書を表示する

Lightsail コンテナサービスの SSL/TLS 証明書を検証する

Amazon Lightsail SSL/TLS 証明書は、それを作成した後 Lightsail コンテナサービスで使用する前に、検証される必要があります。証明書に対するリクエストが送信されると、新しい証明書のステータスが[Attempting to validate your certificate] (証明書の検証を試みています) に変更されます。この間に Lightsail は、証明書に指定したドメイン名の DNS に証明書の検証レコードを追加することを試みます。しばらくすると、ステータスが [Valid] (有効) または [Validation timed out] (検証がタイムアウトしました) に変わります。

自動検証に失敗した場合は、証明書の作成時に指定したすべてのドメイン名を管理していることを確認します。そのためには、証明書で指定された各ドメインの DNS ゾーンに正規名 (CNAME) レコードを追加します。追加する必要のあるレコードが、[Validation details] (検証の詳細) セクションに一覧表示されます。

このガイドでは、Lightsail DNS ゾーンを使用して、証明書を手動で検証するための手順を説明します。Domain.com や GoDaddy などの別の DNS ホスティングプロバイダーを使用して、証明書を検証する手順と類似しているかもしれません。Lightsail DNS ゾーンの詳細については、「DNS」を参照してください。

SSL/TLS 証明書の詳細については、「SSL/TLS 証明書」を参照してください。

前提条件

開始する前に、コンテナサービス用の SSL/TLS 証明書を作成する必要があります。詳細については、「コンテナサービスの SSL/TLS 証明書の作成」を参照してください。

CNAME レコードの値を取得して証明書を検証する

次の手順を実行して、証明書を検証するためにドメインに追加する必要があるCNAMEレコードを取得します。

1. Lightsail コンソール にサインインします。
2. Lightsail コンソールのホームページで、[コンテナ] タブを選択します。
3. 証明書を作成するコンテナサービスの名前を選択します。
4. [Custom domains] (カスタムドメイン) のタブで、コンテナサービス管理ページを選択します。
5. ページ下部の [アタッチされた証明書] セクションまでスクロールします。

   他の Lightsail リソース用に作成された証明書や、検証が保留中の証明書など、すべての証明書が、このページの [Attached certificates] (アタッチされた証明書) セクションに一覧表示されます。
6. 検証する証明書を見つけて、[Validation details] (検証の詳細) を展開し、リストされているドメインごとに追加する必要がある CNAME レコードの [Name] (名前) と [Value] (値) をメモします。

   これらのレコードは、リストされているとおりに正確に追加する必要があります。この値はコピーしてテキストファイルに貼り付け、後で参照できるようにしておくことをお勧めします。詳細については、このガイドの「ドメインの DNS ゾーンに CNAME レコードを追加する」セクションを参照してください。

ドメインの DNS ゾーンに CNAME レコードを追加する

ドメインの DNS ゾーンにCNAMEレコードを追加するには、次のステップを実行します。

1. Lightsail のホームページで [Domains & DNS] (ドメインと DNS) タブを選択します。
2. ページの [DNS ゾーン] セクションで、証明書を検証するために CNAME レコードを追加するドメイン名を選択します。
3. [DNS records] (DNS レコード) タブを選択します。
4. DNS レコードの管理ページで、[Add record] (レコードの追加) を選択します。
5. [Record type] (レコードタイプ) のドロップダウンメニューから、[CNAME] を選択します。
6. [Record name] (レコード名) テキストボックスに、証明書から取得した値を使用して、CNAME レコードの [Name] (名前) を入力します。

   Lightsail コンソールには、ドメインの頂点部分があらかじめ入力されています。たとえば、`www.example.com` サブドメインを追加する場合は、`www` をテキストボックスに入力するだけで、レコードを保存するときに Lightsail が `.example.com` の部分を追加します。
7. [Route traffic to] (トラフィックのルーティング先) テキストボックスに、証明書から取得した CNAME レコード内にある [Value] (値) の部分を入力します。

8. 入力した値が、検証する証明書に記載されている値とまったく同じであることを確認します。
9. 保存アイコンを選択して、レコードを DNS ゾーンに保存します。

これらのステップを繰り返して、検証が必要な証明書のドメインに CNAME レコードを追加します。変更がインターネットの DNS を通じて伝播されるまで待ちます。数分後に、証明書のステータスが [有効] に変わるはずです。詳細については、本ガイドの「証明書のステータスの検証」セクションを参照してください。

証明書のステータスを表示する

SSL/TLS 証明書のステータスを表示するには、次の手順を実行します。

1. Lightsail コンソールのホームページで、[コンテナ] タブを選択します。
2. 証明書のステータスを表示するコンテナサービスの名前を選択します。
3. コンテナサービス 管理ページで [カスタムドメイン] タブを選択します。
4. ページ下部の [アタッチされた証明書] セクションまでスクロールします。

ステータスが [Pending validation] (検証の保留中) や [Valid] (有効) の証明書を含む、すべての証明書が、このページの [Attached certificates] (アタッチされた証明書) セクションに一覧表示されます。

> **Note**
>
> 証明書の検証中、[Custom domains] (カスタムドメイン) ページを開けたままにしている場合は、証明書の更新ステータスを確認するためにページの更新が必要となる場合があります。

[有効] なステータスは、ドメインに追加した CNAME レコードで証明書が正常に検証されたことを確認します。証明書に関する重要な日付、暗号化の詳細、ID、検証レコードを表示するには、[Details] (詳細) を選択します。証明書は、有効にした日から 13 か月間有効で、その後、Lightsail は自動的に再検証を試みます。ドメインに追加した CNAME レコードは、リストされている有効期限日に証明書が再検証されるときに必要になるため、削除しないでください。

SSL/TLS 証明書を検証した後、コンテナサービスのカスタムドメインを有効にして、サービスで証明書のドメイン名を使用できるようにする必要があります。詳細については、「コンテナサービスでカスタムドメインを有効にして管理する」を参照してください。

Lightsail コンテナサービスの SSL/TLS 証明書を表示する

Lightsail コンテナサービス用に作成した Amazon Lightsail SSL/TLS 証明書を表示できます。これを行うには、Lightsail コンソールから、対象となるコンテナサービスの管理ページにアクセスします。

SSL/TLS 証明書の詳細については、「SSL/TLS 証明書」を参照してください。

前提条件

開始する前に、Lightsail コンテナサービスを作成する必要があります。詳細については、「Amazon Lightsail コンテナサービスの作成」および「コンテナサービス」を参照してください。

コンテナサービス用の SSL/TLS 証明書も作成しておく必要があります。詳細については、「コンテナサービスの SSL/TLS 証明書を作成する」を参照してください。

コンテナサービスの SSL/TLS 証明書を表示するには

以下の手順を実行して、コンテナサービスの SSL/TLS 証明書を表示します。

1. Lightsail コンソールにサインインします。
2. Lightsail コンソールのホームページで、[コンテナ] タブを選択します。
3. コンテナサービスの名前を選択します。

   選択したコンテナサービスに関係なく、すべての証明書を表示できます。
4. コンテナサービス 管理ページで [Custom domains] (カスタムドメイン) タブを選択します。
5. ページ下部の [アタッチされた証明書] セクションまでスクロールします。

   すべての証明書は、このページの [Attached certificates] (アタッチされた証明書) セクションに一覧表示されます。証明書に関する重要な日付、暗号化の詳細、ID、およびドメインを表示するには、[Details] (詳細) を選択します。証明書の検証レコードを表示するには、[Validation details] (検証の詳細) を選択します。証明書は、作成日から 13 か月間有効です。その後、Lightsail は自動的に証明書の再認証を試みます。ドメインに追加した CNAME レコードは、リストされている有効期限日に証明書が再検証されるときに必要になるため、削除しないでください。

   コンテナサービスで使用する有効な SSL/TLS 証明書を取得したら、サービスで証明書のドメイン名を使用できるようにカスタムドメインを有効にする必要があります。詳細については、「カスタムドメインの有効化と管理」を参照してください。

Lightsail ディストリビューション SSL/TLS 証明書

ライトセイルディストリビューション用の Amazon Lightsail TLS/SSL 証明書を作成できます。証明書を作成するときは、証明書のプライマリおよび代替ドメイン名を指定します。ディストリビューションのカスタムドメインを有効にして証明書を選択すると、それらのドメインはディストリビューションのカスタムドメインとして追加されます。ディストリビューションを指すようにドメインの DNS レコードを更新すると、ディストリビューションはトラフィックを受け入れ、HTTPS を使用してコンテンツを提供します。作成できる証明書の数にはクォータがあります。詳細については、Lightsail Service Quotas を参照してください。

SSL/TLS 証明書の詳細については、「SSL/TLS 証明書」を参照してください。

> **Important**
>
> ディストリビューションの SSL/TLS 証明書を作成するときに指定するドメイン名は、Amazon サービスのディストリビューションを含め、すべての Amazon Web Services (AWS) アカウントの別のディストリビューションで使用することはできません。 CloudFront ドメインの証明書を作成することはできますが、その証明書をディストリビューションで使用することはできません。

前提条件

始める前に、Lightsail ディストリビューションを作成する必要があります。詳細については、「ディストリビューションを作成する」および「コンテンツ配信ネットワークディストリビューション」を参照してください。

ディストリビューション用の SSL/TLS 証明書を作成する

ディストリビューション用の SSL/TLS 証明書を作成するには、以下の手順を実行します。

1. Lightsail コンソールにサインインします。
2. lightsail のホームページで [Networking] (ネットワーク) タブを選択します。
3. 証明書を作成する対象のディストリビューションの名前を選択します。
4. ディストリビューションの管理ページのカスタムドメインタブを選択します。
5. ページ下部の [アタッチされた証明書] セクションまでスクロールします。

ページの [Attached certificates] (アタッチされた証明書) セクションには、他のディストリビューション用に作成された証明書や、使用中の証明書も使用中でない証明書も含む、すべてのディストリビューション証明書が含まれます。

6. [証明書の作成] を選択します。
7. 証明書を識別する一意の名前を [Certificate name] (証明書の名前) テキストボックスに入力します。次に、[Continue] (続行する) を選択します。
8. 証明書とともに使用するプライマリドメイン名 (例: `example.com`) を、[Specify up to 10 domains or subdomains] (最大 10 個のドメインまたはサブドメインを指定) フィールドに入力します。
9. (オプション) 代替ドメイン名 (例: `www.example.com`) を、残りの [Specify up to 10 domains or subdomains] (最大 10 個のドメインまたはサブドメインを指定) フィールドに入力します。

   証明書には最大 9 個の代替ドメインを追加できます。カスタムドメインを有効にし、ディストリビューションの証明書を選択すると、すべての証明書ドメインをディストリビューションで使用できるようになります。
10. [作成] を選択します。

    証明書のリクエストが送信されると、新しい証明書のステータスは [証明書の検証試行中] に変更されます。この間、Lightsail は証明書の検証レコードをプライマリドメインの DNS に追加しようとします。しばらくすると、ステータスは [有効] に変化します。

    自動検証に失敗した場合、ディストリビューションとともに使用する前に、ドメインで証明書を検証する必要があります。詳細については、「ディストリビューションの SSL/TLS 証明書を検証する」を参照してください。

トピック

- Lightsail ディストリビューションの SSL/TLS 証明書を表示する
- Lightsail ディストリビューションの SSL/TLS 証明書の検証
- Lightsail ディストリビューション証明書の TLS プロトコルの最小バージョンを設定します。
- Lightsail ディストリビューションの SSL/TLS 証明書を削除する

Lightsail ディストリビューションの SSL/TLS 証明書を表示する

Lightsail ディストリビューション用に作成した Amazon Lightsail SSL/TLS 証明書を表示できます。これを行うには、Lightsail コンソールの任意のディストリビューションの管理ページにアクセスします。

SSL/TLS 証明書の詳細については、「SSL/TLS 証明書」を参照してください。

前提条件

始める前に、Lightsail ディストリビューションを作成する必要があります。詳細については、「ディストリビューションを作成する」および「コンテンツ配信ネットワークディストリビューション」を参照してください。

ディストリビューションの SSL/TLS 証明書も作成しておく必要があります。詳細については、「ディストリビューションの SSL/TLS 証明書を作成する」を参照してください。

ディストリビューションの SSL/TLS 証明書を表示

以下の手順を実行して、ディストリビューションの SSL/TLS 証明書を表示します。

1. Lightsail コンソールにサインインします。
2. lightsail のホームページで [Networking] (ネットワーク) タブを選択します。
3. ディストリビューションの名前を選択します。

   選択したディストリビューションに関係なく、すべての証明書を表示できます。
4. ディストリビューション管理ページで [カスタムドメイン] タブを選択します。
5. ページの下部にある [Attached certificates] (アタッチされた証明書) セクションまで下にスクロールします。

   すべてのディストリビューション証明書は、このページの [Attached certificates] (アタッチされた証明書) セクションに一覧表示されます。証明書に関する重要な日付、暗号化の詳細、ID、検証レコードを表示するには、[Validation details] (検証の詳細) を展開します。証明書は、作成日から 13 か月間有効です。その後、Lightsail は自動的に証明書の再認証を試みます。ドメインに追加した CNAME レコードは、リストされている有効期限日に証明書が再検証されるときに必要になるため、削除しないでください。

   ディストリビューションで使用する有効な SSL/TLS 証明書を取得したら、カスタムドメインを有効にして、ディストリビューションの証明書のドメイン名を使用できるようにする必要があり

ます。詳細については、「ディストリビューション用のカスタムドメインを有効にする」を参照してください。

Lightsail ディストリビューションの SSL/TLS 証明書の検証

Amazon Lightsail SSL / TLS 証明書は、作成した後 Lightsail ディストリビューションで使用する前に、検証を行う必要があります。証明書に対するリクエストが送信されると、新しい証明書のステータスが[Attempting to validate your certificate] (証明書の検証を試みています) に変更されます。この間に Lightsail は、証明書に指定したドメイン名の DNS に証明書の検証レコードを追加することを試みます。しばらくすると、ステータスが [Valid] (有効) または [Validation timed out] (検証がタイムアウトしました) に変わります。

自動検証に失敗した場合は、証明書の作成時に指定したすべてのドメイン名を管理していることを確認します。そのためには、証明書で指定された各ドメインの DNS ゾーンに正規名 (CNAME) レコードを追加します。追加する必要のあるレコードが、[Validation details] (検証の詳細) セクションに一覧表示されます。

このガイドでは、Lightsail DNS ゾーンを使用して、証明書を手動で検証するための手順を説明します。Domain.com や GoDaddy などの別の DNS ホスティングプロバイダーを使用して、証明書を検証する手順と類似しているかもしれません。Lightsail DNS ゾーンの詳細については、「DNS」を参照してください。

SSL/TLS 証明書の詳細については、「SSL/TLS 証明書」を参照してください。

目次

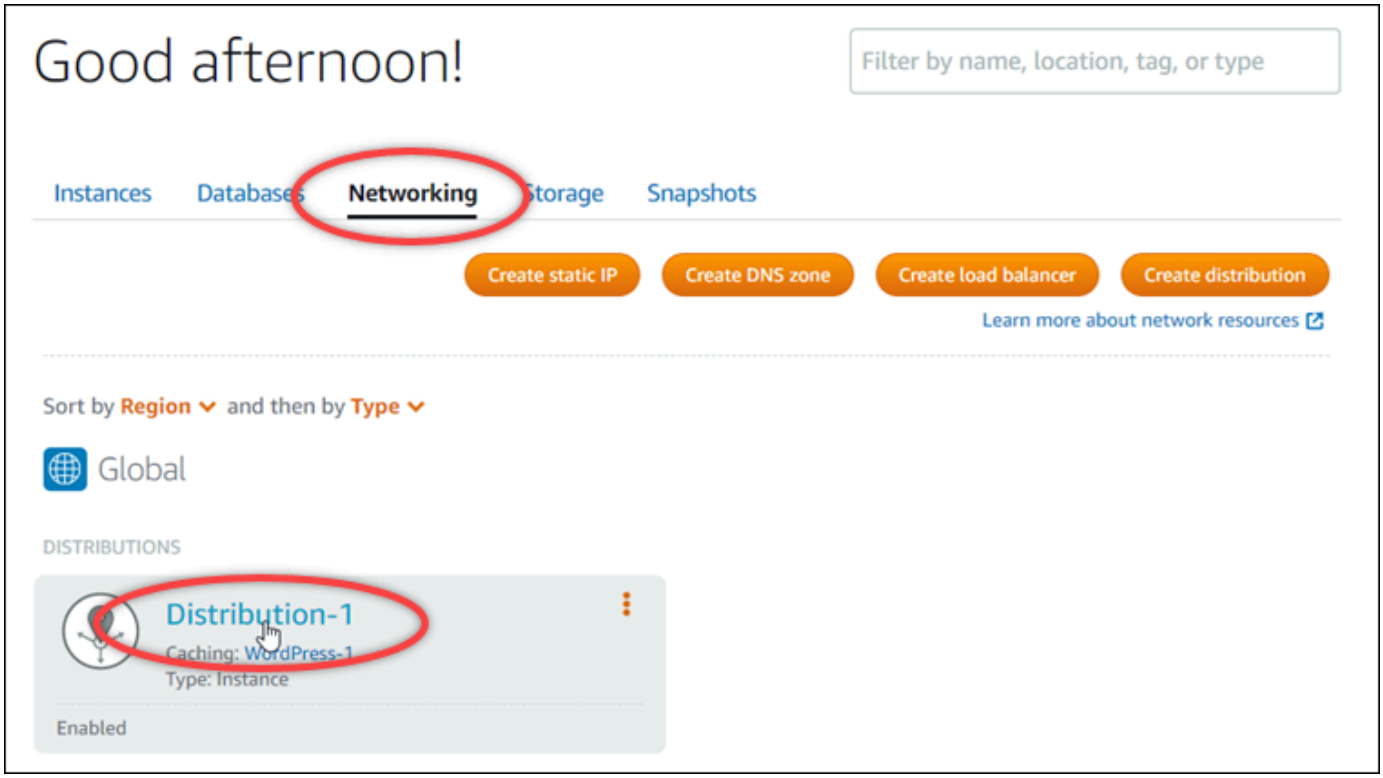
前提条件

開始する前に、ディストリビューション用の SSL/TLS 証明書を作成する必要があります。詳細については、「ディストリビューションの SSL/TLS 証明書を作成する」を参照してください。

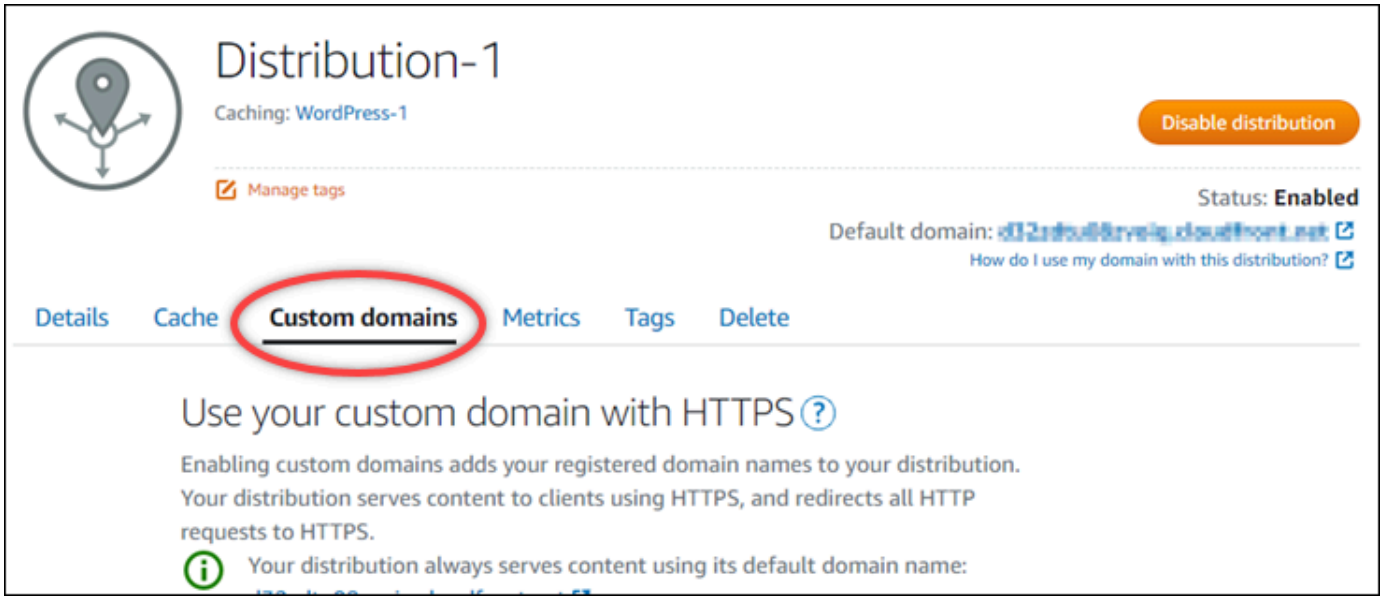
CNAME レコードの値を取得して証明書を検証する

次の手順を実行して、証明書を検証するためにドメインに追加する必要があるCNAMEレコードを取得します。

1. Lightsail コンソール にサインインします。
2. Lightsail のホームページで、[ネットワーキング] タブを選択します。
3. 証明書の CNAME レコード値を取得するディストリビューションの名前を選択します。



4. ディストリビューションの管理ページのカスタムドメインタブを選択します。



5. ページ下部の [アタッチされた証明書] セクションまでスクロールします。

   他の Lightsail リソース用に作成された証明書や、検証が保留中の証明書など、すべてのディストリビューション証明書が、このページの [Attached certificates] (アタッチされた証明書) セクションに一覧表示されます。

6. 検証する証明書を見つけて、[Validation details] (検証の詳細) を展開し、リストされているドメインごとに追加する必要がある CNAME レコードの [Name] (名前) と [Value] (値) をメモします。

   これらのレコードは、リストされているとおりに正確に追加する必要があります。この値はコピーしてテキストファイルに貼り付け、後で参照できるようにしておくことをお勧めします。詳細については、このガイドの「ドメインの DNS ゾーンに CNAME レコードを追加する」セクションを参照してください。

ドメインの DNS ゾーンに CNAME レコードを追加する

ドメインの DNS ゾーンにCNAMEレコードを追加するには、次のステップを実行します。

1. Lightsail のホームページで [Domains & DNS] (ドメインと DNS) タブを選択します。
2. ページの [DNS ゾーン] セクションで、証明書を検証するために CNAME レコードを追加するドメイン名を選択します。
3. [DNS records] (DNS レコード) タブを選択します。
4. DNS レコードの管理ページで、[Add record] (レコードの追加) を選択します。
5. [Record type] (レコードタイプ) のドロップダウンメニューから、[CNAME] を選択します。

6. [Record name] (レコード名) テキストボックスに、証明書から取得した値を使用して、CNAME レコードの [Name] (名前) を入力します。

   Lightsail コンソールには、ドメインの頂点部分があらかじめ入力されています。たとえば、`www.example.com` サブドメインを追加する場合は、`www` をテキストボックスに入力するだけで、レコードを保存するときに Lightsail が `.example.com` の部分を追加します。

7. [Route traffic to] (トラフィックのルーティング先) テキストボックスに、証明書から取得した CNAME レコード内にある [Value] (値) の部分を入力します。

8. 入力した値が、検証する証明書に記載されている値とまったく同じであることを確認します。

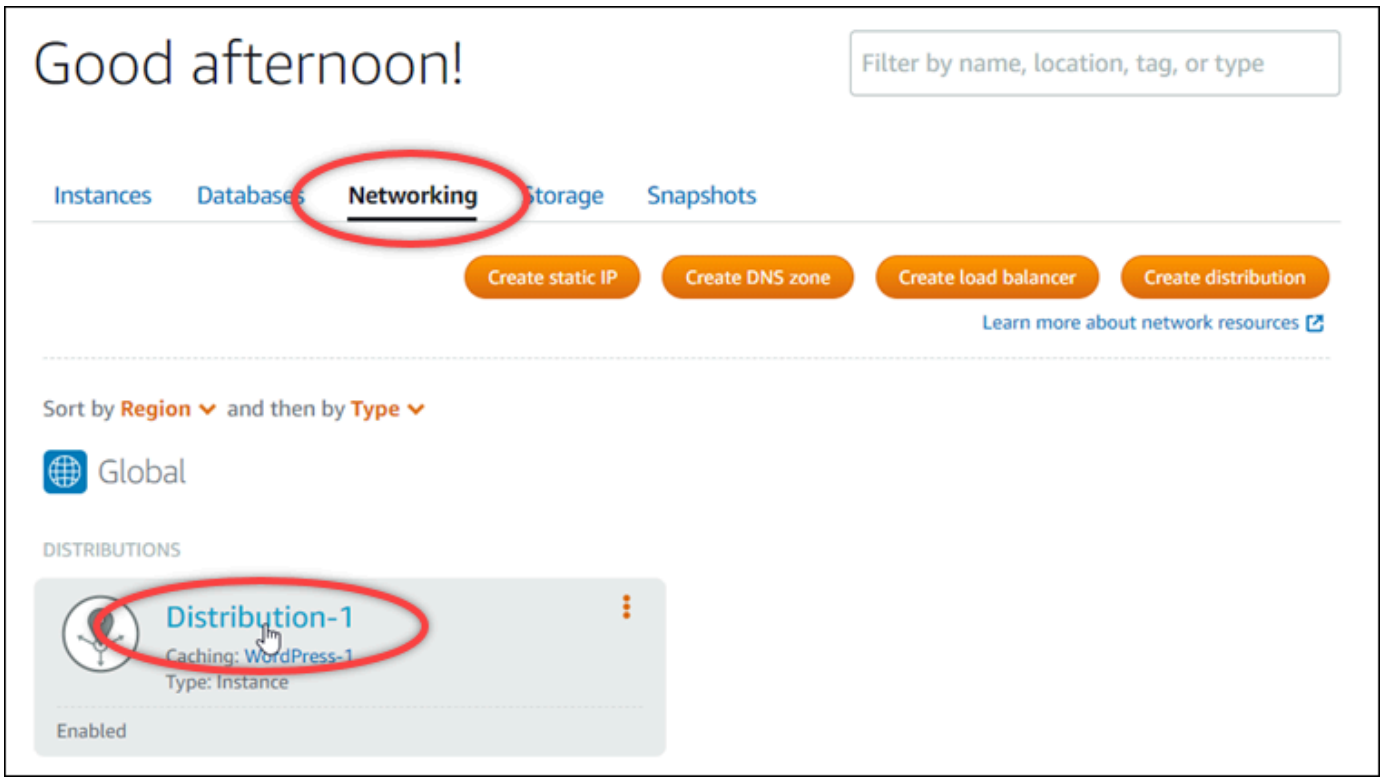
9. 保存アイコンを選択して、レコードを DNS ゾーンに保存します。

   これらのステップを繰り返して、検証が必要な証明書のドメインに CNAME レコードを追加します。変更がインターネットの DNS を通じて伝播されるまで待ちます。数分後に、ディストリビューション証明書のステータスが [有効] に変わるはずです。詳細については、本ガイドの以下の「ディストリビューション証明書のステータスを表示する」セクションを参照してください。

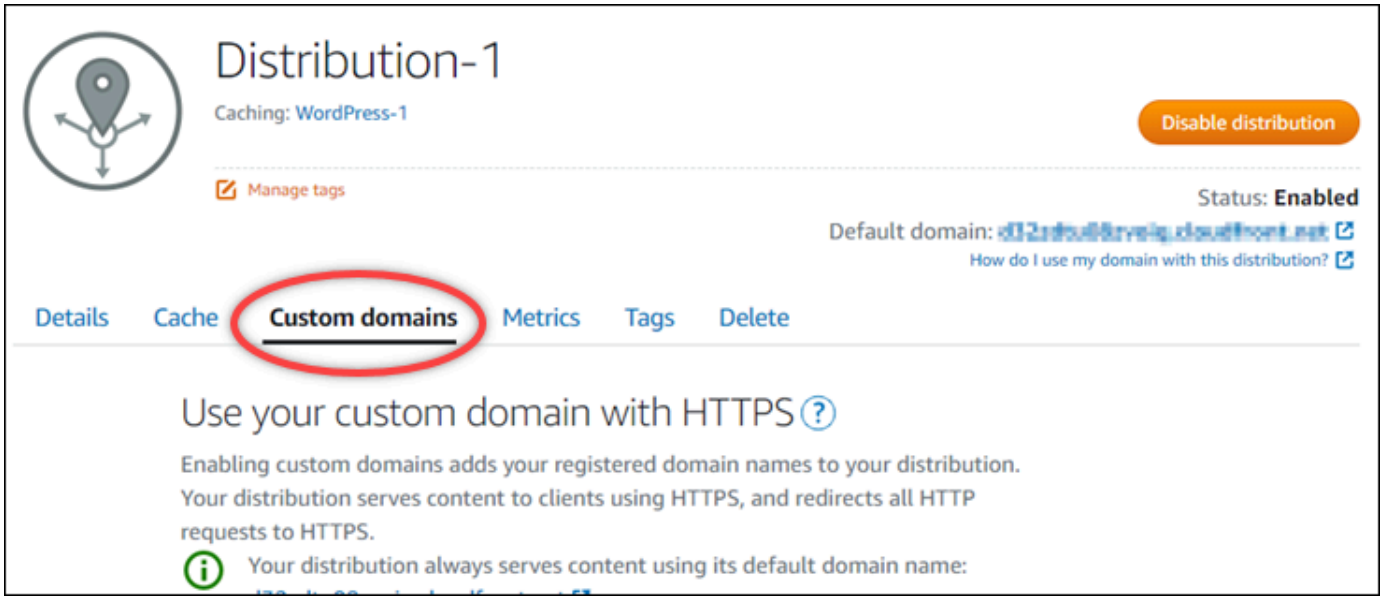
ディストリビューション証明書のステータスを表示する

以下の手順を実行して、ディストリビューションの SSL/TLS 証明書を表示します。

1. Lightsail のホームページで、[ネットワーキング] タブを選択します。
2. 証明書のステータスを表示するディストリビューションの名前を選択します。

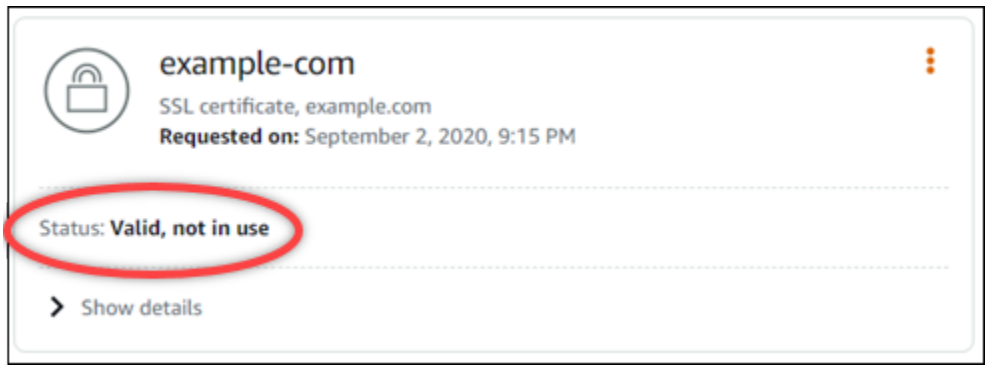


3. ディストリビューションの管理ページのカスタムドメインタブを選択します。



4. ページ下部の [アタッチされた証明書] セクションまでスクロールします。

   ステータスが [Pending validation] (検証の保留中) および [Valid] (有効) の証明書を含む、すべてのディストリビューション証明書が、このページの [Attached certificates] (アタッチされた証明書) セクションに一覧表示されます。



[有効] なステータスは、ドメインに追加した CNAME レコードで証明書が正常に検証されたことを確認します。証明書に関する重要な日付、暗号化の詳細、ID、検証レコードを表示するには、[Details] (詳細) を選択します。証明書は、有効にした日から 13 か月間有効で、その後、Lightsail は自動的に再検証を試みます。ドメインに追加した CNAME レコードは、リストされている [有効期限] 日に証明書が再検証されるときに必要になるため、削除しないでください。

SSL/TLS 証明書を検証したら、ディストリビューションのカスタムドメインを有効にして、ディストリビューションの証明書のドメイン名を使用する必要があります。詳細については、「ディストリビューション用のカスタムドメインを有効にする」を参照してください。

Lightsail ディストリビューション証明書の TLS プロトコルの最小バージョンを設定します。

Amazon Lightsail は SSL/TLS 証明書を使用して、Lightsail ディストリビューションで使用できるカスタム (登録済み) ドメインを検証します。このガイドでは、SSL/TLS 証明書に設定できる Viewer の最小 TLS プロトコルバージョン (プロトコルバージョン) について説明します。SSL/TLS 証明書の詳細については、「Lightsail の SSL/TLS 証明書」を参照してください。ビューアーは、Lightsail ディストリビューションに関連付けられているエッジロケーションに HTTP リクエストを行うアプリケーションです。ディストリビューションの詳細については、「Lightsail のコンテンツ配信ネットワークディストリビューション」を参照してください。

`TLSv1.2_2021`プロトコルバージョンは、ディストリビューションのカスタムドメインを有効にするときにデフォルトで設定されます。このガイドの後半で説明するように、別のプロトコルバージョンを設定できます。Lightsail ディストリビューションはカスタム TLS プロトコルバージョンをサポートしていません。

サポートされるプロトコル

Lightsail ディストリビューションは以下の TLS プロトコルで設定できます。

- (推奨) TLSv1.2_2021
- TLSv1.2_2019
- TLSv1.2_2018
- TLSv1.1_2016

前提条件

以下の前提条件を完了します (まだの場合)。

- Lightsail コンテンツ配信ネットワークディストリビューションの作成
- ディストリビューションの SSL/TLS 証明書を作成する
- ディストリビューションの SSL/TLS 証明書を表示する
- ディストリビューションのカスタムドメインを有効にする
- ドメインをディストリビューションに向けてください。

ディストリビューションの TLS プロトコルの最小バージョンを特定してください。

Lightsail ディストリビューションの TLS プロトコルの最小 TLS プロトコルバージョンを確認するには、次の手順を実行します。

> **Note**
>
> このガイドでは、 AWS CloudShell を使用してアップグレードを実行します。 CloudShell は、Lightsail コンソールから直接起動できるブラウザベースの事前認証済みシェルです。では CloudShell、Bash や Z AWS CLI シェルなどの任意のシェルを使用してコマンドを実行できます。 PowerShellこの手順は、コマンドラインツールのダウンロードもインストールも不要です。設定方法および使用方法の詳細については CloudShell、「Lightsail」AWS CloudShell の「」を参照してください。

1. ターミナル、AWS CloudShell、またはコマンドプロンプトウィンドウを開きます。
2. 次のコマンドを入力して、Lightsail ディストリビューションの TLS プロトコルの最小バージョンを特定します。

```
aws lightsail get-distributions --distribution-name DistributionName --region us-
east-1 | grep "viewerMinimumTlsProtocolVersion"
```

コマンドで、*DistributionName*変更するディストリビューションの名前に置き換えます。

例

```
aws lightsail get-distributions --distribution-name Distribution-1 --region us-
east-1 | grep "viewerMinimumTlsProtocolVersion"
```

このコマンドは、ディストリビューションの TLS プロトコルの最小バージョンの ID を返します。

例

```
"viewerMinimumTlsProtocolVersion": "TLSv1.2_2021"
```

を使用して TLS プロトコルの最小バージョンを設定します。 AWS CLI

以下の手順を実行し、 AWS Command Line Interface ()AWS CLIを使用して TLS プロトコルバージョンを設定します。これは、update-distribution コマンドを使用して実行できます。詳細については、『コマンドリファレンス』の update-distribution 属性を参照してください。AWS CLI

1. ターミナル、AWS CloudShell、またはコマンドプロンプトウィンドウを開きます。
2. 次のコマンドを入力して、ディストリビューションの TLS プロトコルの最小バージョンを変更します。

```
aws lightsail update-distribution --distribution-name DistributionName --viewer-
minimum-tls-protocol-version ProtocolVersion
```

コマンドで、次のサンプルテキストを独自のテキストに置き換えます。

- *DistributionName*更新するディストリビューションの名前を入力します。
- *ProtocolVersion*有効な TLS プロトコルバージョンで。たとえば、TLSv1.2_2021、TLSv1.2_2019 などです。

例：

```
aws lightsail update-distribution --distribution-name  MyDistribution --viewer-
minimum-tls-protocol-version TLSv1.2_2021
```

変更が有効になるまで、少し時間がかかります。

Lightsail ディストリビューションの SSL/TLS 証明書を削除する

ディストリビューションで使用しなくなった Amazon Lightsail SSL/TLS 証明書を削除できます。たとえば、証明書の有効期限が切れており、検証済みの更新された証明書を既にアタッチしている場合などです。証明書の詳細については、「SSL/TLS 証明書」を参照してください。ディストリビューションの詳細については、「コンテンツ配信ネットワークディストリビューション」を参照してください。

SSL/TLS 証明書の削除は元に戻すことができません。365 日間に作成できる証明書の数にはクォータがあります。詳細については、AWS 全般のリファレンス の「Lightsail サービスクォータ」を参照してください。

ディストリビューション用の SSL/TLS 証明書を削除する

ディストリビューション用の SSL/TLS 証明書を削除するには、以下の手順を実行します。

1. Lightsail コンソールにサインインします。
2. Lightsail のホームページで、[ネットワーキング] タブを選択します。
3. SSL/TLS 証明書を削除するディストリビューションの名前を選択します。証明書が現在使用中でない場合は、すべてのディストリビューションにすべての証明書がリストされるため、どのディストリビューションでも選択できます。
4. ディストリビューション管理ページで [カスタムドメイン] タブを選択します。
5. ページの [証明書] セクションで、削除する証明書の省略記号アイコン (:) を選択し、[削除] を選択します。

   [削除] オプションは、削除する証明書が使用中の場合は使用できません。使用中の証明書を削除するには、まず証明書を使用しているディストリビューションのカスタムドメインを変更するか、証明書を使用しているディストリビューションのカスタムドメインを無効にする必要があります。詳細については、「ディストリビューションのカスタムドメインを変更する」および「ディストリビューションのカスタムドメインを有効化する」を参照してください。

6. [はい、削除します] を選択して削除を確定します。

Amazon Lightsail でのオブジェクトストレージ

Amazon Lightsail オブジェクトストレージサービスを使用して、インターネット上のどこからでも、いつでもオブジェクトの格納と取得ができます。また、ウェブスケールのコンピューティングを開発者が簡単に利用できるよう設計されています。また、Amazon Simple Storage Service (Amazon S3) を使用して構築されています。Lightsail オブジェクト ストレージにより、スケーラブルで信頼性が高く、かつ高速で安価なデータストレージインフラストラクチャを利用できるようになります。このインフラストラクチャは、Amazon が独自にウェブサイトのグローバルネットワークを運営するために使用しているものと同じです。このサービスの目的は、規模の拡大や縮小のメリットを最大限に活かし、その利益をお客様に提供することです。

オブジェクトストレージの概念

次の概念と用語は Lightsail オブジェクトストレージに適用されます。

バケット

バケットとは、オブジェクトのコンテナで、Lightsail オブジェクトストレージサービスに保存されています。すべてのオブジェクトは、バケットに格納され、それぞれのバケットには、独自の URL があります。たとえば、`media/sailbot.jpg` という名前のオブジェクトが米国東部 (バージニア北部) リージョン (`us-east-1`) の `DOC-EXAMPLE-BUCKET` バケットに保存される場合、`https://DOC-EXAMPLE-BUCKET.us-east-1.amazonaws.com/media/sailbot.jpg` に類似した URL を使用してアドレスを指定できます。

Lightsail が利用可能な AWS リージョン でバケットを作成できます。Lightsail が利用可能な AWS リージョン の詳細については、「AWS 全般のリファレンス」の「リージョンとエンドポイント」を参照してください。

バケットストレージプラン

AWS API 内のバンドルとされるストレージ プランは、バケットの月額コスト、ストレージ容量、データ転送クォータを決めます。最初にバケットを作成するときに、ストレージプランを選択する必要があります。バケットの起動後に変更することもできます。

バケットのプランは、月次の AWS 請求サイクルの中で 1 回だけ変更できます。バケットがストレージ領域またはデータ転送クォータを一貫して上回っている場合、またはバケットの使用量がストレージ領域またはデータ転送クォータより低い範囲にある場合、バケットのプランを変更します。バケットの使用量の変動が予測できない場合があるため、バケットのプランの変更は、短期的な毎月のコス

ト削減策ではなく、長期的な戦略としてのみ行うことを強くお勧めします。今後長期にわたりバケットに十分なストレージ容量とデータ転送クォータを提供するストレージプランを選択します。

オブジェクト

オブジェクトは、バケットに格納される基本エンティティです。バケットにアップロードしたファイルは、格納されている間、オブジェクトと呼ばれます。オブジェクトは、データとメタデータで構成されます。データ部分は、Lightsail オブジェクトストレージサービスの不透明体です。メタデータは、オブジェクトを表現する名前と値のペアのセットです。これには最終更新日などのデフォルトのメタデータや、標準 HTTP メタデータ (Content-Type など) が含まれます。

オブジェクトは、キー名とバージョン ID によってバケット内で一意に識別されます。

オブジェクトキー名

キー名とは、バケット内のオブジェクトの固有の識別子です。バケット内のすべてのオブジェクトは、厳密に 1 個のキーを持ちます。バケット、キー、バージョン ID の組み合わせで、各オブジェクトを一意に識別します。そのため、Lightsail オブジェクトストレージを「バケット + キー + バージョン」とオブジェクト自体の間の基本データマップと考えることができます。Lightsail オブジェクトストレージ内の各オブジェクトは、ウェブサービスエンドポイント、バケット名、キー、およびオプションでバージョンを組み合わせることで一意にアクセスすることができます。たとえば、`https://DOC-EXAMPLE-BUCKET.us-east-1.amazonaws.com/media/sailbot.jpg` という URL では、「`DOC-EXAMPLE-BUCKET`」がバケットの名前で、「`media/sailbot.jpg`」がオブジェクトキー名になります。

オブジェクトのバージョニング

バージョニングとは、同じバケット内でオブジェクトの複数のバリアントを保持する機能です。バージョニングを有効にすると、 バケットに保存されたあらゆるオブジェクトのあらゆるバージョンを保存、取得、復元します。バージョニングを使用すれば、意図しないユーザーアクションからもアプリケーション障害からも、より簡単に復旧できます。

バケットの作成時、デフォルトでは、バージョニングは無効になっています。バージョニングを有効にすると、バケットに格納するすべてのオブジェクトのすべてのバージョンは、保存されたバージョンを手動で削除するまで保持されます。たとえば、`media/sailbot.jpg` オブジェクトを格納した後で、同じオブジェクトキー名を持つより大きなファイルを格納すると、元の小さいオブジェクトが以前のバージョンとして保持されます。新しい大きなオブジェクトは現行のバージョンになります。以前のバージョンのオブジェクトが必要ないと判断した場合、オブジェクトを削除できます。オブジェクトの最新バージョンを削除すると、そのオブジェクトの以前のバージョンはすべて削除されます。

格納されたオブジェクトバージョンは、格納されたオブジェクトの現在のバージョンと同じ方法で、バケットのストレージ領域を消費します。バージョニングを有効にした後は、そのバージョニングを中断して、オブジェクトバージョンの保存を停止できます。これにより、新しいオブジェクトバージョンをアップロードするときに、バケットのストレージ領域が消費されることも少なくなります。バージョニングを一時停止すると、保存されたオブジェクトバージョンは保持されますが、バージョニングを一時停止している間にアップロードした新しいオブジェクトバージョンは保持されません。

バケットとオブジェクトのアクセス

デフォルトでは、すべてのオブジェクトストレージリソース (バケットとオブジェクト) はプライベートです。バケットを作成した Lightsail アカウント (バケット所有者) のみが、バケットとそれに含まれるオブジェクトにアクセスできます。バケット所有者は、他のユーザーにアクセス許可を付与することもできます。これは、すべてのオブジェクトまたは個々のオブジェクトを公開に設定することで実行できます。これにより、世界中の誰でも読みやすくなります。また、プログラムによる完全なアクセスを付与するには、Lightsail インスタンスをバケットに追加するか、バケットのアクセスキーを作成します。最後に、他の AWS アカウントを使用して、バケットへのプログラムによる読み取り専用アクセスを許可します。

AWS リージョン

また、Lightsail が利用可能なすべての AWS リージョン で、Lightsail オブジェクト ストレージ バケットを作成できます。レイテンシーを最適化し、コストを最小限に抑えて規制用件に対応できるリージョンを選ぶとよいでしょう。明示的に別のリージョンに移動する場合を除き、AWS リージョン に格納されたオブジェクトは、そのリージョンから移動されることはありません。たとえば、米国西部 (オレゴン) リージョンに格納されたオブジェクトがそこから移動することはありません。

バケットとオブジェクトを管理する

Lightsail オブジェクトストレージは、シンプルさと堅牢性を重視し、必要な機能に絞って提供しています。バケットとオブジェクトを管理する要素の一部を次に示します。

- バケットの作成 – データを格納するバケットを作成します。バケットとは、Lightsail オブジェクトストレージにおける基本的なコンテナです。詳細については、「バケットの作成」を参照してください。
- データの保存 — Lightsail コンソール、AWS Command Line Interface (AWS CLI)、および AWS API を使用してバケットにファイルをアップロードします。ファイルのアップロードに関する詳細については、「バケットにファイルをアップロードする」を参照してください。

- データのダウンロード — 保存したオブジェクトをいつでもダウンロードできます。詳細については、「バケットからオブジェクトをダウンロードする」を参照してください。
- アクセス権の付与 — 外部 (ソフトウェアや個人など) からの、バケット内のデータのアップロードまたはデータのダウンロードアクセスを許可または拒否します。認証メカニズムによって、データソースを不正アクセスから保護することができます。詳細については、「バケットのアクセス許可」を参照してください。
- バージョニング管理 — バケットに保存されたあらゆるオブジェクトのあらゆるバージョンを保持するために、バージョニングを有効にします。詳細については、「バケットでのオブジェクトのバージョニングの有効化と一時停止」を参照してください。
- 使用状況のモニタリング — バケットに格納されているオブジェクトの数と、使用されているストレージ領域の量を監視します。詳細については、「バケットメトリクスを表示」を参照してください。
- ストレージプランを変更する — バケットが過剰に使用されている場合はアップサイズを、使用されていない場合はダウンサイズします。詳細については、「バケットのプランの変更」を参照してください。
- バケットを接続する — Lightsail バケットを WordPress ウェブサイトに接続して、ウェブサイトの画像と添付ファイルを保存します。バケットを、Lightsail コンテンツ配信ネットワーク (CDN) ディストリビューションのオリジンとして指定します。これにより、世界中のユーザーへのバケット内のオブジェクトの配信が高速化されます。詳細については、「チュートリアル: バケットを WordPress インスタンスに接続する」および「チュートリアル: バケットをコンテンツ配信ネットワークディストリビューションと使用する」を参照してください。
- バケットの削除 — 使用しなくなったバケットを削除します。詳細については、「バケットの削除」を参照してください。

Lightsail バケットを作成する

クラウドへのファイルのアップロードを開始する準備が整ったら、Amazon Lightsailオブジェクトストレージサービスでバケットを作成します。Lightsail オブジェクトストレージサービスにアップロードしたすべてのファイルは Lightsail バケットに保存されます。バケットについての詳細は、「オブジェクトストレージ」を参照してください。

バケットを作成する

Lightsail バケットを作成する手順は以下のとおりです。

1. Lightsail コンソールにサインインします。

2. Lightsail のホームページで [Storage] (ストレージ) タブを選択します。
3. [バケットを作成] を選択します。
4. [AWS リージョン の変更] を選択して、バケットを作成するリージョンを選択します。

   バケットで使用する予定のリソースと同じ AWS リージョン でバケットを作成することをお勧めします。作成後にバケットのリージョンを変更することはできません。
5. バケットのストレージプランを選択します。

   ストレージプランでは、バケットの月額コスト、ストレージ領域のクォータ、データ転送クォータを指定します。

   バケットのプランは、月次の AWS 請求サイクルの中で 1 回だけ変更できます。バケットがストレージ領域またはデータ転送クォータを一貫して上回っている場合、またはバケットの使用量がストレージ領域またはデータ転送クォータより低い範囲にある場合、バケットのプランを変更します。詳細については、「バケットのプランを変更する」を参照してください。
6. バケットの名前を入力します。

   バケット名の詳細については、「Amazon Lightsail でのバケット命名規則」を参照してください。
7. [Create bucket] (バケットの作成) を選択します。

   新しいバケットの管理ページにリダイレクトされます。バケットを使用および管理するための追加ドキュメントについては、このガイドの「次のステップ」セクションに進んでください。

バケットとオブジェクトを管理する

これらは、Lightsail オブジェクトストレージバケットを管理する一般的な手順です。

1. Amazon Lightsail オブジェクトストレージサービスでのオブジェクトとバケットについて説明します。詳細については、「Amazon Lightsail のオブジェクトストレージ」を参照してください。
2. Amazon Lightsail でバケットに付けることができる名前について説明します。詳細については、「Amazon Lightsail でのバケットの命名規則」をご参照ください。
3. バケットを作成して、Lightsail オブジェクトストレージサービスの使用を開始します。詳細については、「Amazon Lightsail におけるバケットの作成 」を参照してください。
4. バケットのセキュリティのベストプラクティスと、バケットに設定できるアクセス許可について説明します。バケット内のすべてのオブジェクトをパブリックまたはプライベートにすることも、オブジェクトを個別に選択してパブリックにすることもできます。また、アクセスキーを

作成し、インスタンスをバケットに追加し、他の AWS アカウントにアクセス権を付与することで、バケットへのアクセスを許可することもできます。詳細については、「Amazon Lightsail オブジェクトストレージのセキュリティベストプラクティス」と「Amazon Lightsail でのバケットのアクセス許可を理解する」を参照してください。

バケットのアクセス許可について理解したら、以下のガイドを参照してバケットへのアクセスを許可してください。

- Amazon Lightsail でのバケットへのパブリックアクセスをブロックする
- Amazon Lightsail でのバケットのアクセス許可の設定
- Amazon Lightsail でのバケット内の個々のオブジェクトに対するアクセス許可の設定
- Amazon Lightsail でのバケットのアクセスキーの作成
- Amazon Lightsail でのバケットのリソースアクセスの設定
- Amazon Lightsail でのバケットのクロスアカウントアクセスの設定

5. バケットのアクセスログの記録を有効にする方法と、アクセスログを使用してバケットのセキュリティを監査する方法について説明します。詳細については、以下のガイドを参照してください。
   - Amazon Lightsail オブジェクトストレージサービスのバケットへのアクセスのログ記録
   - Amazon Lightsail オブジェクトストレージサービスのバケットのアクセスログの形式
   - Amazon Lightsail オブジェクトストレージサービスのバケットへのアクセスのログ記録を有効にする
   - Amazon Lightsailでのバケットのアクセスログを使用するリクエストの特定
6. Lightsail でバケットを管理する機能をユーザーに付与する IAM ポリシーを作成します。詳細については、「Amazon Lightsail でバケットを管理する IAM ポリシー」を参照してください。
7. バケット内のオブジェクトにラベルを付けて識別する方法について説明します。詳細については、「Amazon Lightsail でのオブジェクトキー名を理解する」を参照してください。
8. ファイルをアップロードしてバケット内のオブジェクトを管理する方法について説明します。詳細については、以下のガイドを参照してください。
   - Amazon Lightsail のバケットにファイルをアップロードする
   - マルチパートアップロードを使用した Amazon Lightsail のバケットへのファイルのアップロード
   - Amazon Lightsail のバケット内のオブジェクトの表示
   - Amazon Lightsail のバケット内のオブジェクトのコピーまたは移動
   - Amazon Lightsail のバケットからのオブジェクトのダウンロード

- Amazon Lightsail のバケット内のオブジェクトのフィルタリング
- Amazon Lightsail のバケット内のオブジェクトのタグ付け
- Amazon Lightsail のバケット内のオブジェクトの削除

9. オブジェクトのバージョニングを有効にすると、バケットに保存されたあらゆるオブジェクトのあらゆるバージョンを保存、取得、復元します。詳細については、「Amazon Lightsail のバケットでのオブジェクトのバージョニングの有効化と一時停止」を参照してください。
10. オブジェクトのバージョニングを有効にすると、バケット内のオブジェクトの以前のバージョンを復元できます。詳細については、「Amazon Lightsail のバケット内のオブジェクトの以前のバージョンの復元」を参照してください。
11. バケットの使用率を監視します。詳細については、「Amazon Lightsail でのバケットのメトリクスの表示」を参照してください。
12. バケットの使用率がしきい値を超えたときにバケットメトリクスが通知されるよう、アラームを設定します。詳細については、「Amazon Lightsail でのバケットメトリクスアラームの作成」を参照してください。
13. ストレージとネットワーク転送量が不足している場合は、バケットのストレージプランを変更します。詳細については、「Amazon Lightsail のバケットのプランの変更」を参照してください。
14. バケットを他のリソースに接続する方法について説明します。詳細については、以下のチュートリアルを参照してください。
    - チュートリアル: WordPress インスタンスの Amazon Lightsail バケットへの接続
    - チュートリアル: Lightsail コンテンツ配信ネットワークディストリビューションでの Amazon Lightsail バケットの使用
15. 使用しなくなったバケットを削除します。詳細については、「Amazon Lightsail でのバケットの削除」を参照してください。

Lightsail バケットを削除する

Amazon Lightsail オブジェクトストレージサービスで使用していないバケットを削除します。バケットを削除すると、保存されたバージョンのオブジェクトやアクセスキーなど、バケット内のすべてのオブジェクトが完全に削除されます。

バケットについての詳細は、「オブジェクトストレージ」を参照してください。

バケットの強制削除

次のいずれかの条件を持つバケットは、削除を承認しない限り、削除できません。

- ディストリビューションのオリジンのバケット。
- インスタンスが添付されたバケット。
- オブジェクトがあるバケット。
- アクセスキーがあるバケット。

バケットに依存する既存のワークフローが中断されないように、削除を承認する必要があります。たとえば、バケットにメディアを保存している WordPress ウェブサイトや、バケット内のオブジェクトをキャッシュして提供しているディストリビューションなどが該当します。

前述の条件のいずれかを持つバケットの削除を承認するには、バケットを強制的に削除する必要があります。バケットを削除する前に、Lightsail サービスはこれらのどの条件が該当するか、プロンプトで表示します。Lightsail コンソールを使用してバケットを削除すると、バケットを強制的に削除するオプションが表示されます。AWS CLI を使用する場合、`delete-bucket` リクエストを作成するときに `--force-delete` フラグを指定する必要があります。これらの手順については、このガイドの「Lightsail コンソールを使用してバケットを削除する」および「AWS CLI を使用してバケットを削除する」のセクションを参照してください。

Lightsail コンソールを使用してバケットを削除する

Lightsail コンソールを使用してバケットを削除するには、以下の手順を実行します。

1. Lightsail コンソールにサインインします。
2. Lightsail のホームページで [Storage] (ストレージ) タブを選択します。
3. 削除するバケットの名前を選択します。
4. タブメニューの省略記号 (:) アイコンを選択し、そして [削除] を選択します。
5. [Delete Bucket (バケットを削除)] を選択します。
6. 表示されるプロンプトで、バケットが次の条件のいずれかを満たしているかどうかを確認します。
   - オブジェクトを含む
   - アクセスキーを含む
   - インスタンスに添付されている
   - ディストリビューションのオリジンである

これらの条件のいずれかが該当する場合は、バケットを強制的に削除するように選択する必要があります。

7. 以下のオプションのいずれかを選択します。

   - [強制削除] を選択することで、この手順のステップ 6 の条件を有していてもバケットを削除することができます。
   - ステップ 6 に記された条件を有していない場合は、「はい、削除します」を選択してバケットを削除します。
   - 「いいえ、キャンセルします」を選択して削除をキャンセルします。

AWS CLI を使用してバケットを削除する

AWS Command Line Interface (AWS CLI) を使用してバケットを削除するには、以下の手順を実行します。これは、`delete-bucket` コマンドを使用して実行できます。詳細については、「AWS CLI コマンドリファレンス」の「delete-bucket」を参照してください。

> **Note**
>
> この手順を続行する前に、AWS CLI をインストールし、Lightsail と Amazon S3 用に設定する必要があります。詳細については、「Lightsail で使用するために AWS CLI を設定する」を参照してください。

1. ターミナルまたはコマンドプロントウィンドウを開きます。
2. コマンドプロンプトまたはターミナルウィンドウで、次のいずれかのコマンドを入力します。

   - 本ガイドの[「バケットの強制削除」セクションに記されている条件が当てはまらないバケットを削除するためには、以下のコマンドを入力します。

     ```
     aws lightsail delete-bucket --bucket-name BucketName
     ```

   - 本ガイドの「バケットの強制削除」セクションに記されている条件が当てはまるバケットを削除するためには、以下のコマンドを入力します。

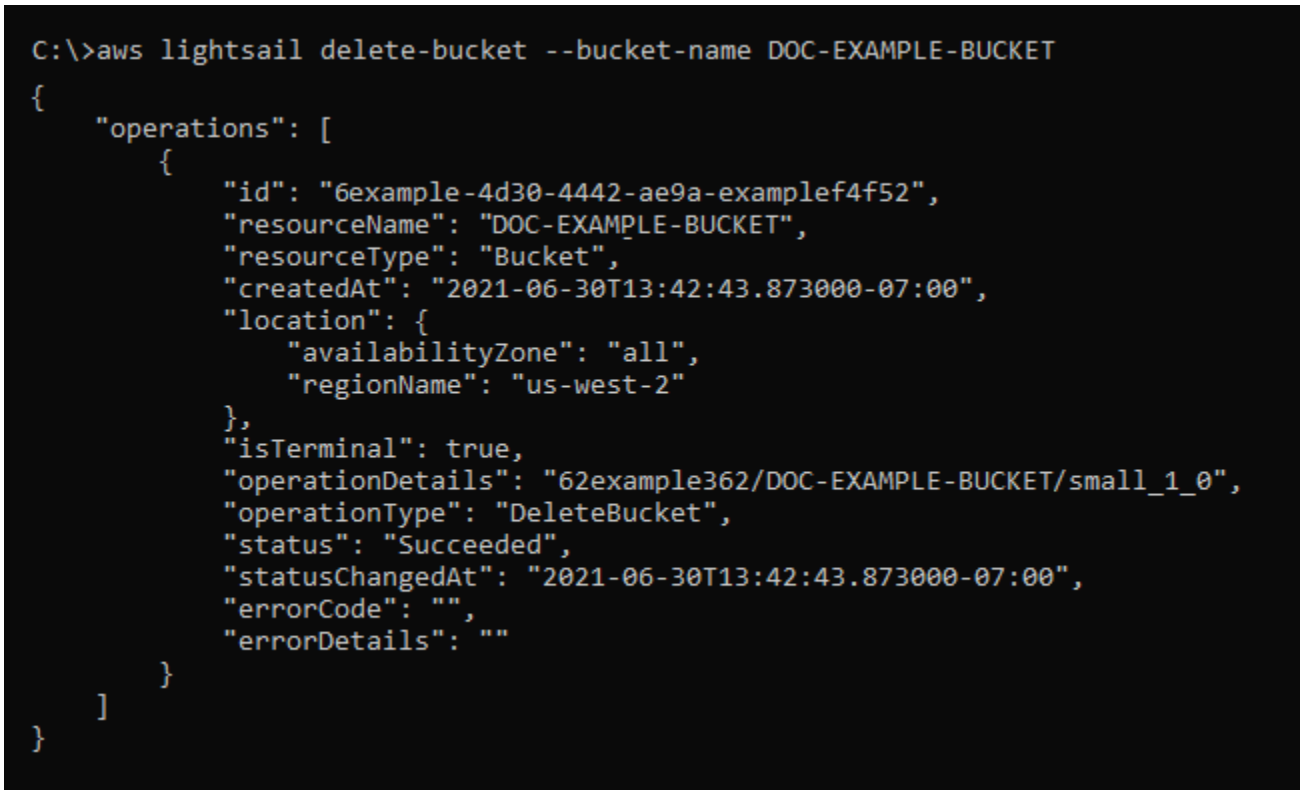
     ```
     aws lightsail delete-bucket --bucket-name BucketName --force-delete
     ```

コマンドで、*BucketName* を削除するバケットの名前に置き換えます。

例:

```
aws lightsail delete-bucket --bucket-name DOC-EXAMPLE-BUCKET
```

以下の例のような結果が表示されるはずです。



バケットとオブジェクトを管理する

これらは、Lightsail オブジェクトストレージバケットを管理する一般的な手順です。

1. Amazon Lightsail オブジェクトストレージサービスでのオブジェクトとバケットについて説明します。詳細については、「Amazon Lightsail のオブジェクトストレージ」を参照してください。
2. Amazon Lightsail でバケットに付けることができる名前について説明します。詳細については、「Amazon Lightsail でのバケットの命名規則」をご参照ください。
3. バケットを作成して、Lightsail オブジェクトストレージサービスの使用を開始します。詳細については、「Amazon Lightsail におけるバケットの作成 」を参照してください。

4. バケットのセキュリティのベストプラクティスと、バケットに設定できるアクセス許可について説明します。バケット内のすべてのオブジェクトをパブリックまたはプライベートにすることも、オブジェクトを個別に選択してパブリックにすることもできます。また、アクセスキーを作成し、インスタンスをバケットに追加し、他の AWS アカウントにアクセス権を付与することで、バケットへのアクセスを許可することもできます。詳細については、「Amazon Lightsail オブジェクトストレージのセキュリティベストプラクティス」と「Amazon Lightsail でのバケットのアクセス許可を理解する」を参照してください。

   バケットのアクセス許可について理解したら、以下のガイドを参照してバケットへのアクセスを許可してください。

   - Amazon Lightsail でのバケットへのパブリックアクセスをブロックする
   - Amazon Lightsail でのバケットのアクセス許可の設定
   - Amazon Lightsail でのバケット内の個々のオブジェクトに対するアクセス許可の設定
   - Amazon Lightsail でのバケットのアクセスキーの作成
   - Amazon Lightsail でのバケットのリソースアクセスの設定
   - Amazon Lightsail でのバケットのクロスアカウントアクセスの設定
5. バケットのアクセスログの記録を有効にする方法と、アクセスログを使用してバケットのセキュリティを監査する方法について説明します。詳細については、以下のガイドを参照してください。

   - Amazon Lightsail オブジェクトストレージサービスのバケットへのアクセスのログ記録
   - Amazon Lightsail オブジェクトストレージサービスのバケットのアクセスログの形式
   - Amazon Lightsail オブジェクトストレージサービスのバケットへのアクセスのログ記録を有効にする
   - Amazon Lightsailでのバケットのアクセスログを使用するリクエストの特定
6. Lightsail でバケットを管理する機能をユーザーに付与する IAM ポリシーを作成します。詳細については、「Amazon Lightsail でバケットを管理する IAM ポリシー」を参照してください。
7. バケット内のオブジェクトにラベルを付けて識別する方法について説明します。詳細については、「Amazon Lightsail でのオブジェクトキー名を理解する」を参照してください。
8. ファイルをアップロードしてバケット内のオブジェクトを管理する方法について説明します。詳細については、以下のガイドを参照してください。

   - Amazon Lightsail のバケットにファイルをアップロードする
   - マルチパートアップロードを使用した Amazon Lightsail のバケットへのファイルのアップロード

- Amazon Lightsail のバケット内のオブジェクトの表示
- Amazon Lightsail のバケット内のオブジェクトのコピーまたは移動
- Amazon Lightsail のバケットからのオブジェクトのダウンロード
- Amazon Lightsail のバケット内のオブジェクトのフィルタリング
- Amazon Lightsail のバケット内のオブジェクトのタグ付け
- Amazon Lightsail のバケット内のオブジェクトの削除

9. オブジェクトのバージョニングを有効にすると、バケットに保存されたあらゆるオブジェクトのあらゆるバージョンを保存、取得、復元します。詳細については、「Amazon Lightsail のバケットでのオブジェクトのバージョニングの有効化と一時停止」を参照してください。
10. オブジェクトのバージョニングを有効にすると、バケット内のオブジェクトの以前のバージョンを復元できます。詳細については、「Amazon Lightsail のバケット内のオブジェクトの以前のバージョンの復元」を参照してください。
11. バケットの使用率を監視します。詳細については、「Amazon Lightsail でのバケットのメトリクスの表示」を参照してください。
12. バケットの使用率がしきい値を超えたときにバケットメトリクスが通知されるよう、アラームを設定します。詳細については、「Amazon Lightsail でのバケットメトリクスアラームの作成」を参照してください。
13. ストレージとネットワーク転送量が不足している場合は、バケットのストレージプランを変更します。詳細については、「Amazon Lightsail のバケットのプランの変更」を参照してください。
14. バケットを他のリソースに接続する方法について説明します。詳細については、以下のチュートリアルを参照してください。
    - チュートリアル: WordPress インスタンスの Amazon Lightsail バケットへの接続
    - チュートリアル: Lightsail コンテンツ配信ネットワークディストリビューションでの Amazon Lightsail バケットの使用
15. 使用しなくなったバケットを削除します。詳細については、「Amazon Lightsail でのバケットの削除」を参照してください。

Lightsail バケットのアクセスキーを作成する

アクセスキーを使用して、バケットとそのオブジェクトへのフルアクセスを許可する認証情報セットを作成します。ソフトウェアまたはプラグインでアクセスキーを設定して、AWS API、および AWS SDK を使用してバケットへの完全な読み取りおよび書き込みアクセスを許可できます。AWS CLI でアクセスキーを設定することもできます。

アクセスキーは、アクセスキー ID とシークレットアクセスキーとのセットで構成されます。シークレットアクセスキーは、作成時にのみ使用できます。シークレットアクセスキーがコピーされた場合、紛失した場合、または危険にさらされた場合は、アクセスキーを削除し、新しいキーを作成する必要があります。1 つのバケットにつき、最大 2 つのアクセスキーを持つことができます。バケットのアクセスキーは 2 つ持つことができますが、キーをローテーションする必要がある場合、 1 つのキーが便利です。アクセスキーをローテーションするには、新しいキーを作成し、ソフトウェアで設定してテストしてから、以前のキーを削除します。アクセスキーを削除すると、永久に削除されるため、再度取得することはできません。新しいアクセスキーでのみ置き換えることができます。

許可のオプションの詳細については、「バケットのアクセス許可」を参照してください。セキュリティのベストプラクティスの詳細については、「オブジェクトストレージのセキュリティのベストプラクティス」を参照してください。バケットについての詳細は、「オブジェクトストレージ」を参照してください。

バケットのアクセスキーを作成する

バケットのアクセスキーを作成するには、以下の手順を実行します。

1. Lightsail コンソールにサインインします。
2. Lightsail のホームページで [Storage] (ストレージ) タブを選択します。
3. アクセス権限を設定するバケットの名前を選択します。
4. [Permissions (許可)] タブを選択します。

   ページのアクセスキーセクションには、バケットの既存のアクセスキー (存在する場合) が表示されます。
5. バケットの新しいキーを作成するには、[Create access key (アクセスキーの作成)] を選択します。

   > **Note**
   >
   > 削除するキーのごみ箱アイコンを選択して、既存のアクセスキーを削除することもできます。
6. 表示されるプロンプトで、「はい、作成します」を選択し、新しいアクセスキーの作成を確定します。それ以外の場合は、[キャンセル] を選択します。
7. 表示される成功プロンプトで、アクセスキー ID を書き留めます。

8. [Show secret access key (シークレットアクセスキーを表示)] を選択してシークレットアクセスキーを表示し、それをメモします。シークレットアクセスキーは再度表示されることはありません。

> **Important**
>
> アクセスキー ID とシークレットアクセスキーは安全な場所に保存します。これらが漏洩された場合は、削除し、新しいキーを作成する必要があります。

9. [Continue (続行)] を選択して終了します。

新しいアクセスキーはページのアクセスキーのセクションで操作します。アクセスキーが漏洩された場合、または紛失した場合は、キーを削除し、新しいキーを作成します。

> **Note**
>
> 各アクセスキーの隣に表示される [最終使用日] の列は、キーが最後に使用されたのがいつかを示します。キーが使用されていない場合は、ダッシュが表示されます。アクセスキーノードを展開して、キーが最後に使用されたサービスと AWS リージョン を表示します。

Lightsail バケットに対するパブリックアクセスをブロックする

Amazon Simple Storage Service (Amazon S3) は、お客様がデータを保存して保護することができるオブジェクトストレージサービスです。Amazon Lightsail オブジェクトストレージサービスは Amazon S3 テクノロジーに基づいて構築されています。Amazon S3 はアカウントレベルのブロックパブリックアクセスを提供しており、これを使用してAWS アカウント 内のすべての S3 バケットへのパブリックアクセスを制限できます。アカウントレベルのパブリックアクセスのブロックでは、既存の個別のバケットおよびオブジェクトの許可にかかわらず、AWS アカウント のすべての S3 バケットをプライベートにすることができます。

パブリックアクセスを許可または拒否する場合、Lightsail オブジェクトストレージバケットは次のことを考慮します。

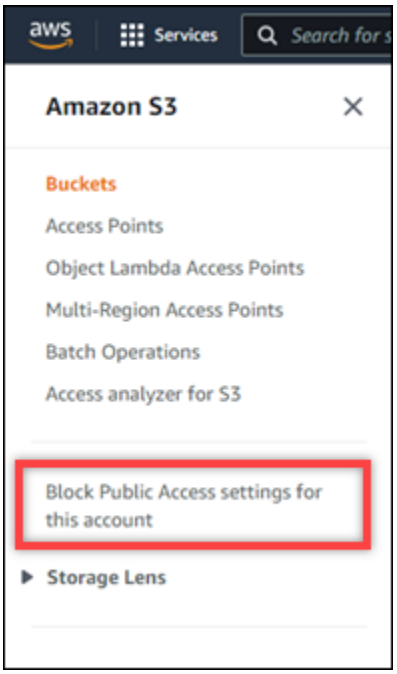
- Lightsail バケットアクセス許可。詳細については、「バケットのアクセス許可」を参照してください。

- Amazon S3 アカウントレベルのパブリックアクセスのブロック設定。Lightsail バケットアクセス許可を上書きします。

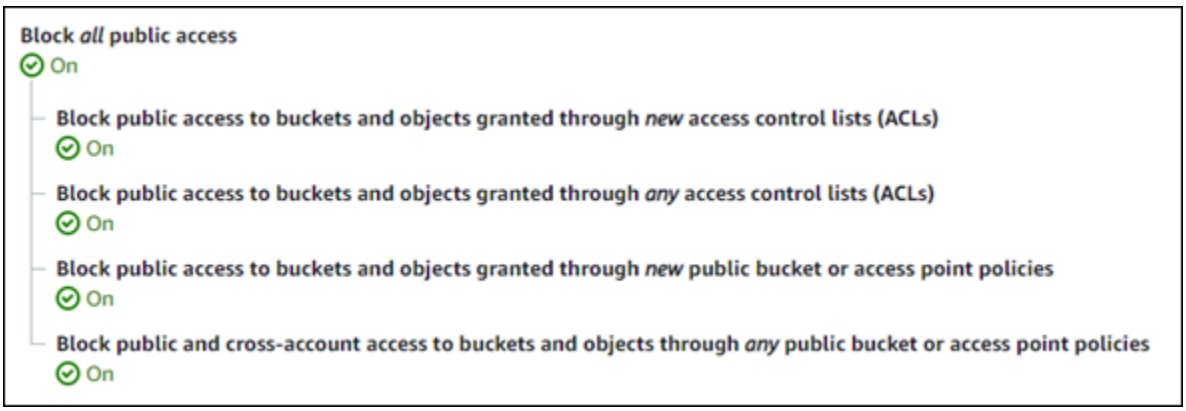
Amazon S3 でアカウントレベルの [すべてのパブリックアクセスをブロックする] をオンにすると、パブリック Lightsail バケットとオブジェクトがプライベートになり、公開アクセスが無効になります。

アカウントのブロックパブリックアクセス設定の構成

パブリックアクセスのブロックを設定するには、Amazon S3 コンソール、AWS Command Line Interface、(AWS CLI)、AWS SDK、および REST API を使用することができます。次の例に示すように、Amazon S3 コンソールのナビゲーションペインでアカウントレベルのパブリックアクセスのブロック機能にアクセスできます。



Amazon S3 コンソールには、すべてのパブリックアクセスのブロック、新しいまたは任意のアクセスコントロールリストを通じて付与されたパブリックアクセスのブロック、新しいまたは任意のパブリックバケットまたはアクセスポイントポリシーを通じて付与されたバケットおよびオブジェクトへのパブリックアクセスのブロックの設定があります。



Amazon S3 コンソールで各設定を [オン] または [オフ] にできます。API では、対応する設定は TRUE (オン) または FALSE (オフ) です。次のセクションでは、S3 バケットと Lightsail バケットに対する各設定の影響について説明します。

> **Note**
>
> 次のセクションでは、アクセスコントロールリスト (ACL) について説明します。ACL は、バケットまたは個々のオブジェクトを所有している、またはそれらにアクセスできるユーザーを定義します。詳細については、「Amazon S3 ユーザーガイド」の「アクセスコントロールリストの概要」を参照してください。

- [すべてのパブリックアクセスをブロックする] - この設定をオンにすると、S3 バケット、Lightsail バケット、およびそれらに対応するオブジェクトへのすべてのパブリックアクセスがブロックされます。この設定には、次の設定がすべて組み込まれています。この設定をオンにすると、あなた (バケット所有者) と許可されたユーザーのみが、バケットとそのオブジェクトにアクセスできます。この設定は、Amazon S3 コンソールでのみオンにできます。AWS CLI、Amazon S3 API、または AWS SDK では使用できません。
  - 新しいアクセスコントロールリスト (ACL) を通じて付与されたバケットおよびオブジェクトへのパブリックアクセスをブロック — この設定をオンにすると、バケットおよびオブジェクトに対するパブリック ACL の配置がブロックされます。この設定は、既存の ACL には影響しません。したがって、既にパブリック ACL を持つオブジェクトはパブリックのままとなります。この設定は、バケットアクセス許可が [All objects are public and read-only] (すべてのオブジェクトがパブリックかつ読み取り専用) に設定されているため、パブリックであるオブジェクトに影響を与えることもありません。この設定は、Amazon S3 API で `BlockPublicAcls` としてラベル付けされています。

> **Note**
>
> Offload Media Light プラグインなど、メディアを Lightsail バケットに配置する WordPress プラグインは、この設定をオンにすると動作を停止する場合があります。これは、ほとんどの WordPress プラグインがオブジェクトでパブリック読み取り ACL を設定するためです。オブジェクト ACL を切り替える WordPress プラグインも動作を停止する可能性があります。

- すべてのアクセスコントロールリスト (ACL) を通じて付与されたバケットおよびオブジェクトへのパブリックアクセスをブロック — この設定をオンにすると、パブリック ACL が無視され、バケットおよびオブジェクトへのパブリックアクセスがブロックされます。この設定では、パブリック ACL をバケットとオブジェクトに配置できますが、アクセス権を付与するときは無視されます。Lightsail バケットの場合、バケットのアクセス許可を [All objects are public and read-only] (すべてのオブジェクトがパブリックかつ読み取り専用) に設定するか、または個別のオブジェクトの許可を [Public (read-only)] (パブリック (読み取り専用)) に設定することは、パブリック ACL をいずれかに配置することと同等です。この設定は、Amazon S3 API で `IgnorePublicAcls` としてラベル付けされています。
- 新しいパブリックバケットまたはアクセスポイントポリシーを通じて付与されたバケットおよびオブジェクトへのパブリックアクセスをブロック — この設定をオンにすると、[すべてのオブジェクトがパブリックかつ読み取り専用] のバケットアクセス許可が、Lightsail バケットで設定されないようにブロックします。この設定は、[All objects are public and read-only] (すべてのオブジェクトがパブリックかつ読み取り専用) のバケットアクセス許可で既に設定されているバケットには影響しません。この設定は、Amazon S3 API で `BlockPublicPolicy` としてラベル付けされています。
- 任意のパブリックバケットまたはアクセスポイントポリシーを通じたバケットおよびオブジェクトへのパブリックおよびクロスアカウントアクセスをブロック — この設定をオンにすると、すべての Lightsail バケットがプライベートになります。これにより、[All objects are public and read-only] (すべてのオブジェクトがパブリックかつ読み取り専用) のバケットアクセス許可で設定されている場合でも、すべての Lightsail バケットがプライベートになります。この設定は、Amazon S3 API で `RestrictPublicBuckets` としてラベル付けされています。

> **Important**
>
> この設定は、Lightsail バケットで設定されているクロスアカウントアクセス (Lightsail で [All objects are public and read-only] (すべてのオブジェクトがパブリックかつ読み取り専用) のバケットアクセス許可でも設定されているもの) もブロックします。クロスアカ

ウントアクセスを引き続き許可するには、Amazon S3 で [任意のパブリックバケットまたはアクセスポイントポリシーを通じたバケットおよびオブジェクトへのパブリックおよびクロスアカウントアクセスをブロック] をオンにする前に、Lightsail で [すべてのオブジェクトをプライベートにする] のバケットアクセス許可で Lightsail バケットを設定してください。

ブロックパブリックアクセスとその設定方法の詳細については、「Amazon S3 ユーザーガイド」で以下のリソースを参照してください。

- Amazon S3 ストレージへのパブリックアクセスのブロック
- アカウントのブロックパブリックアクセス設定の構成

Lightsail コンソール、AWS CLI、AWS SDK、および REST API を使用して、Lightsail バケットのアクセス許可を設定します。詳細については、「バケットのアクセス許可」を参照してください。

> **Note**
>
> Lightsail は、Amazon S3 から現在のアカウントレベルのブロックパブリックアクセス設定を取得し、それを Lightsail オブジェクトストレージリソースに適用するためにサービスリンクロールを使用します。Amazon S3 でパブリックアクセスのブロックを設定した後、Lightsail で有効になるまで少なくとも 1 時間待機します。詳細については、「サービスにリンクされたロール」を参照してください。

バケットとオブジェクトを管理する

これらは、Lightsail オブジェクトストレージバケットを管理する一般的な手順です。

1. Amazon Lightsail オブジェクトストレージサービスでのオブジェクトとバケットについて説明します。詳細については、「Amazon Lightsail のオブジェクトストレージ」を参照してください。
2. Amazon Lightsail でバケットに付けることができる名前について説明します。詳細については、「Amazon Lightsail でのバケットの命名規則」をご参照ください。
3. バケットを作成して、Lightsail オブジェクトストレージサービスの使用を開始します。詳細については、「Amazon Lightsail におけるバケットの作成 」を参照してください。
4. バケットのセキュリティのベストプラクティスと、バケットに設定できるアクセス許可について説明します。バケット内のすべてのオブジェクトをパブリックまたはプライベートにすること

も、オブジェクトを個別に選択してパブリックにすることもできます。また、アクセスキーを作成し、インスタンスをバケットに追加し、他の AWS アカウントにアクセス権を付与することで、バケットへのアクセスを許可することもできます。詳細については、「Amazon Lightsail オブジェクトストレージのセキュリティベストプラクティス」と「Amazon Lightsail でのバケットのアクセス許可を理解する」を参照してください。

バケットのアクセス許可について理解したら、以下のガイドを参照してバケットへのアクセスを許可してください。

- Amazon Lightsail でのバケットへのパブリックアクセスをブロックする
- Amazon Lightsail でのバケットのアクセス許可の設定
- Amazon Lightsail でのバケット内の個々のオブジェクトに対するアクセス許可の設定
- Amazon Lightsail でのバケットのアクセスキーの作成
- Amazon Lightsail でのバケットのリソースアクセスの設定
- Amazon Lightsail でのバケットのクロスアカウントアクセスの設定

5. バケットのアクセスログの記録を有効にする方法と、アクセスログを使用してバケットのセキュリティを監査する方法について説明します。詳細については、以下のガイドを参照してください。
   - Amazon Lightsail オブジェクトストレージサービスのバケットへのアクセスのログ記録
   - Amazon Lightsail オブジェクトストレージサービスのバケットのアクセスログの形式
   - Amazon Lightsail オブジェクトストレージサービスのバケットへのアクセスのログ記録を有効にする
   - Amazon Lightsailでのバケットのアクセスログを使用するリクエストの特定
6. Lightsail でバケットを管理する機能をユーザーに付与する IAM ポリシーを作成します。詳細については、「Amazon Lightsail でバケットを管理する IAM ポリシー」を参照してください。
7. バケット内のオブジェクトにラベルを付けて識別する方法について説明します。詳細については、「Amazon Lightsail でのオブジェクトキー名を理解する」を参照してください。
8. ファイルをアップロードしてバケット内のオブジェクトを管理する方法について説明します。詳細については、以下のガイドを参照してください。
   - Amazon Lightsail のバケットにファイルをアップロードする
   - マルチパートアップロードを使用した Amazon Lightsail のバケットへのファイルのアップロード
   - Amazon Lightsail のバケット内のオブジェクトの表示
   - Amazon Lightsail のバケット内のオブジェクトのコピーまたは移動

- Amazon Lightsail のバケットからのオブジェクトのダウンロード
- Amazon Lightsail のバケット内のオブジェクトのフィルタリング
- Amazon Lightsail のバケット内のオブジェクトのタグ付け
- Amazon Lightsail のバケット内のオブジェクトの削除

9. オブジェクトのバージョニングを有効にすると、バケットに保存されたあらゆるオブジェクトのあらゆるバージョンを保存、取得、復元します。詳細については、「Amazon Lightsail のバケットでのオブジェクトのバージョニングの有効化と一時停止」を参照してください。
10. オブジェクトのバージョニングを有効にすると、バケット内のオブジェクトの以前のバージョンを復元できます。詳細については、「Amazon Lightsail のバケット内のオブジェクトの以前のバージョンの復元」を参照してください。
11. バケットの使用率を監視します。詳細については、「Amazon Lightsail でのバケットのメトリクスの表示」を参照してください。
12. バケットの使用率がしきい値を超えたときにバケットメトリクスが通知されるよう、アラームを設定します。詳細については、「Amazon Lightsail でのバケットメトリクスアラームの作成」を参照してください。
13. ストレージとネットワーク転送量が不足している場合は、バケットのストレージプランを変更します。詳細については、「Amazon Lightsail のバケットのプランの変更」を参照してください。
14. バケットを他のリソースに接続する方法について説明します。詳細については、以下のチュートリアルを参照してください。
    - チュートリアル: WordPress インスタンスの Amazon Lightsail バケットへの接続
    - チュートリアル: Lightsail コンテンツ配信ネットワークディストリビューションでの Amazon Lightsail バケットの使用
15. 使用しなくなったバケットを削除します。詳細については、「Amazon Lightsail でのバケットの削除」を参照してください。

Amazon Lightsail でのバケットのアクセスログ

アクセスログは、Amazon Lightsail オブジェクトストレージサービスのバケットに対して行われたリクエストの詳細なレコードを提供します。この情報には、リクエストタイプ、リクエストで指定したリソース、リクエストを処理した日時などが含まれます。アクセスのログは、多くのアプリケーションに役立ちます。例えば、アクセスのログ情報は、セキュリティやアクセスの監査に役立ちます。また、顧客基盤について知るうえでも役立ちます。

目次

- ログ配信を有効にするには何が必要ですか
- ログオブジェクトのキーフォーマット
- ログを配信する方法
- ベストエフォート型のアクセスログ配信
- バケットのログ記録ステータスの変更が有効になるまでには時間がかかる

ログ配信を有効にするには何が必要ですか

ログ配信を有効にする前に、次の点を考慮してください。詳細は、「バケットアクセスのログ記録を有効にする」を参照してください。

1. ログのターゲットバケットを特定します。このバケットは、Lightsail がアクセスログをオブジェクトとして保存する場所です。ソースバケットとターゲットバケットの両方が同じ AWS リージョンにあり、同じアカウントによって所有されている必要があります。

   ログの保存先のバケットとして、ソースバケットと同じリージョンにあるユーザー所有のバケットを指定できます。これにはソースバケット自体も含まれます。ただし、ログを管理しやすくするため、アクセスログは別のバケットに保存することをお勧めします。

   ソースバケットとターゲットバケットが同じである場合、バケットに書き込まれるログに関する追加のログが作成されます。これは、ストレージの消費がいくらか増える可能性があるため、望ましくない場合があります。また、ログに関する追加のログのために、必要なログを見つけにくくなります。アクセスログの保存先をソースバケットにする場合は、ログオブジェクトを簡単に区別できるように、すべてのログオブジェクトキーにプレフィックスを指定し、オブジェクト名を共通の文字列で始めてください。キープレフィックスは、複数のバケットが同じターゲットバケットにログを記録する場合に、ソースバケットを区別するためにも役立ちます。
2. (オプション) ログオブジェクトキーのプレフィックスを特定します。プレフィックスを使用すると、ログオブジェクトを見つけやすくなります。例えば、プレフィックスの値として `logs/` を指定すると、Lightsail で作成する各ログオブジェクトのキーの先頭に `logs/` というプレフィックスが付きます。プレフィックスの末尾であることを示すには、末尾のスラッシュ `/` が必要です。`logs/` プレフィックス付きのログオブジェクトキーの例を次に示します。

   ```
   logs/2021-11-31-21-32-16-E568B2907131C0C0
   ```

ログオブジェクトのキーフォーマット

Lightsail では、ターゲットバケットにログオブジェクトをアップロードする際に、以下のオブジェクトキーフォーマットを使用します。

```
TargetPrefix/YYYY-mm-DD-HH-MM-SS-UniqueString
```

このキーで、YYYY、mm、DD、HH、MM、SS は、ログファイルを配信した年、月、日、時、分、秒をそれぞれ表す数字です。これらの日付と時刻は協定世界時 (UTC) です。

ある時点で配信されたログファイルには、その時点より前に書き込まれたレコードが含まれます。特定の期間のすべてのログレコードが配信されたかどうかを知る方法はありません。

キーの UniqueString コンポーネントは、ファイルの上書きを防止するためのものです。意味はないため、ログ処理ソフトウェアでは無視されます。

ログを配信する方法

Lightsail は、アクセスログレコードを定期的に収集してログファイルにまとめ、そのログファイルをログオブジェクトとしてターゲットバケットにアップロードします。複数のソースバケットでログ記録の配信先が同じターゲットバケットである場合、これらのすべてのソースバケットのアクセスログがターゲットバケットに収容されます。ただし、各ログオブジェクトは、ソースバケット別にアクセスログレコードをレポートします。

ベストエフォート型のアクセスログ配信

アクセスログレコードの配信は、ベストエフォートで行われます。ログ記録用に適切にバケットを設定した場合、そのバケットへのほとんどのリクエストについてログレコードが配信されます。ほとんどのログレコードは、記録された時間から数時間以内に配信されますが、配信間隔は短くなる場合もあります。

アクセスのログ記録の完全性や適時性は保証されません。リクエストのログレコードが、リクエストが実際に処理されてからかなり後に配信されたり、配信すらされなかったりすることもあり得ます。アクセスログの目的は、バケットに対するトラフィックの特性を理解することです。ログレコードが失われることはまれですが、すべてのリクエストが完全に報告されるとは限りません。

バケットのログ記録ステータスの変更が有効になるまでには時間がかかる

バケットのログ記録ステータスの変更がログファイルの配信に反映されるまでには時間がかかります。例えば、バケットのログを有効にする場合、その後数時間に行われるリクエストは記録される

こともあれば、されないこともあります。ログ記録のターゲットバケットをバケット A からバケット B に変更すると、その後 1 時間は一部のログがバケット A に引き続き配信されたり、新しいターゲットバケット B に配信されたりします。いずれにしても、最終的に新しい設定が有効になるため、ユーザー側の操作は一切不要です。

トピック

- Amazon Lightsail のバケットアクセスログのフォーマット
- Amazon Lightsail のバケットアクセスログ記録の有効化
- Amazon Lightsail でバケットアクセスログを使用してリクエストを識別する

Amazon Lightsail のバケットアクセスログのフォーマット

アクセスログは、Amazon Lightsail オブジェクトストレージサービスのバケットに対して行われたリクエストの詳細なレコードを提供します。アクセスのログ記録を使用して、セキュリティとアクセスを監査し、顧客ベースを確認することができます。このセクションでは、アクセスログファイルの形式およびその他の詳細について説明します。ログ記録の基本の詳細については、「バケットのアクセスログ」を参照してください。

アクセスのログファイルは、一連のログレコードを改行で区切って構成します。各ログレコードは 1 個のリクエストを表し、各フィールドをスペースで区切って構成します。

次に示すのは、5 個のログレコードで構成されるログの例です。

```
79a59df900b949e55d96a1e698fbacedfd6e09d98eacf8f8d5218e7cd47ef2be
 awsexamplebucket1 [06/Feb/2019:00:00:38 +0000] 192.0.2.3
 79a59df900b949e55d96a1e698fbacedfd6e09d98eacf8f8d5218e7cd47ef2be 3E57427F3EXAMPLE
 REST.GET.VERSIONING - "GET /awsexamplebucket1?versioning HTTP/1.1" 200 - 113 - 7 -
 "-" "S3Console/0.4" - s9lzHYrFp76ZVxRcpX9+5cjAnEH2ROuNkd2BHfIa6UkFVdtjf5mKR3/eTPFvsiP/
XV/VLi31234= SigV2 ECDHE-RSA-AES128-GCM-SHA256 AuthHeader awsexamplebucket1.s3.us-
west-1.amazonaws.com TLSV1.1
```

```
79a59df900b949e55d96a1e698fbacedfd6e09d98eacf8f8d5218e7cd47ef2be
 awsexamplebucket1 [06/Feb/2019:00:00:38 +0000] 192.0.2.3
 79a59df900b949e55d96a1e698fbacedfd6e09d98eacf8f8d5218e7cd47ef2be 891CE47D2EXAMPLE
 REST.GET.LOGGING_STATUS - "GET /awsexamplebucket1?logging HTTP/1.1" 200 - 242
 - 11 - "-" "S3Console/0.4" - 9vKBE6vMhrNiWHZmb2L0mXOcqPGzQOI5XLnCtZNPxev+Hf
+7tpT6sxDwDty4LHBUOZJG96N1234= SigV2 ECDHE-RSA-AES128-GCM-SHA256 AuthHeader
 awsexamplebucket1.s3.us-west-1.amazonaws.com TLSV1.1
```

```
79a59df900b949e55d96a1e698fbacedfd6e09d98eacf8f8d5218e7cd47ef2be
 awsexamplebucket1 [06/Feb/2019:00:00:38 +0000] 192.0.2.3
 79a59df900b949e55d96a1e698fbacedfd6e09d98eacf8f8d5218e7cd47ef2be A1206F460EXAMPLE
 REST.GET.BUCKETPOLICY - "GET /awsexamplebucket1?policy HTTP/1.1" 404
 NoSuchBucketPolicy 297 - 38 - "-" "S3Console/0.4" - BNaBsXZQQDbssi6xMBdBU2sLt
+Yf5kZDmeBUP35sFoKa3sLLeMC78iwEIWxs99CRUrbS4n11234= SigV2 ECDHE-RSA-AES128-GCM-SHA256
 AuthHeader awsexamplebucket1.s3.us-west-1.amazonaws.com TLSV1.1
```

```
79a59df900b949e55d96a1e698fbacedfd6e09d98eacf8f8d5218e7cd47ef2be
 awsexamplebucket1 [06/Feb/2019:00:01:00 +0000] 192.0.2.3
 79a59df900b949e55d96a1e698fbacedfd6e09d98eacf8f8d5218e7cd47ef2be 7B4A0FABBEXAMPLE
 REST.GET.VERSIONING - "GET /awsexamplebucket1?versioning HTTP/1.1" 200 - 113
 - 33 - "-" "S3Console/0.4" - Ke1bUcazaN1jWuUlPJaxF64cQVpUEhoZKEG/hmy/gijN/
I1DeWqDfFvnpybfEseEME/u7ME1234= SigV2 ECDHE-RSA-AES128-GCM-SHA256 AuthHeader
 awsexamplebucket1.s3.us-west-1.amazonaws.com TLSV1.1
```

```
79a59df900b949e55d96a1e698fbacedfd6e09d98eacf8f8d5218e7cd47ef2be
 awsexamplebucket1 [06/Feb/2019:00:01:57 +0000] 192.0.2.3
 79a59df900b949e55d96a1e698fbacedfd6e09d98eacf8f8d5218e7cd47ef2be
 DD6CC733AEXAMPLE REST.PUT.OBJECT s3-dg.pdf "PUT /awsexamplebucket1/
s3-dg.pdf HTTP/1.1" 200 - - 4406583 41754 28 "-" "S3Console/0.4" -
 10S62Zv81kBW7BB6SX4XJ48o6kpcl6LPwEoizZQQxJd5qDSCTLX0TgS37kYUBKQW3+bPdrg1234= SigV4
 ECDHE-RSA-AES128-SHA AuthHeader awsexamplebucket1.s3.us-west-1.amazonaws.com TLSV1.1
```

> **Note**
>
> 任意のログレコードフィールドを – (ダッシュ) に設定して、データが不明または使用不可であること、またはフィールドがこのリクエストに適用されなかったことを示すことができます。

目次

ログレコードフィールド

次のリストには、ログレコードのフィールドが説明されています。

アクセスポイントの ARN (Amazon リソースネーム)

リクエストのアクセスポイントの Amazon リソースネーム (ARN) です。アクセスポイントの ARN の形式が不正、または使用されていない場合、このフィールドには「-」が含まれます。アクセスポイントの詳細については、「アクセスポイントを使用する」を参照してください。ARN の詳細については、AWS 全般リファレンスで「Amazon リソースネーム (ARN)」に関するトピックを参照してください。

エントリ例

```
arn:aws:s3:us-east-1:123456789012:accesspoint/example-AP
```

バケット所有者

ソースバケット所有者の正規ユーザー ID。正規ユーザー ID は、別の形式の AWS アカウント ID です。正規ユーザー ID の詳細については、「AWS 全般のリファレンス」の「AWS アカウント ID」を参照してください。アカウントの正規ユーザー ID を検索する方法については、「AWS アカウントの正規ユーザー ID の検索」を参照してください。

エントリ例

```
79a59df900b949e55d96a1e698fbacedfd6e09d98eacf8f8d5218e7cd47ef2be
```

バケット

リクエストの処理ターゲットのバケットの名前。システムで受け取ったリクエストの形式に誤りがあり、バケットを特定できない場合、そのリクエストはアクセスログに表示されません。

エントリ例

```
awsexamplebucket1
```

Time (時間)

リクエストが受信された時間。これらの日付と時刻は協定世界時 (UTC) です。*strftime()* 用語を使用すると、形式は次のようになります: *[%d/%b/%Y:%H:%M:%S %z]*

エントリ例

```
[06/Feb/2019:00:00:38 +0000]
```

リモート IP

リクエスタの表面上のインターネットアドレス。中間プロキシやファイアウォールにより、リクエストを作成したマシンの実際のアドレスが不明確になる場合があります。

エントリ例

```
192.0.2.3
```

リクエスタ

リクエスタの正規ユーザー ID。認証されていないリクエストの場合は - です。リクエスタが IAM ユーザーの場合、このフィールドは、リクエスタの IAM ユーザー名と IAM ユーザーが所属する AWS ルートアカウントを返します。この識別子は、アクセスコントロールに使用されるものと同じです。

エントリ例

```
79a59df900b949e55d96a1e698fbacedfd6e09d98eacf8f8d5218e7cd47ef2be
```

リクエスト ID

各リクエストを一意に識別するために Lightsail で生成される文字列。

エントリ例

```
3E57427F33A59F07
```

操作

ここに表示されているオペレーション は、SOAP.*operation*、REST.*HTTP_method.resource_type*、WEBSITE.*HTTP_method.resource_t* または BATCH.DELETE.OBJECT と表示されます。

エントリ例

```
REST.PUT.OBJECT
```

キー

リクエストの URL エンコードされた「key」部分、オペレーションがキーパラメータを取らない場合は「-」。

エントリ例

```
/photos/2019/08/puppy.jpg
```

Request-URI

HTTP リクエストメッセージの Request-URI の部分。

エントリ例

```
"GET /awsexamplebucket1/photos/2019/08/puppy.jpg?x-foo=bar HTTP/1.1"
```

HTTP ステータス

レスポンスの HTTP ステータスの数値。

エントリ例

```
200
```

エラーコード

Amazon S3 エラーコード、またはエラーが発生しなかった場合は「-」。

エントリ例

```
NoSuchBucket
```

送信バイト数

送信されたレスポンスのバイト数 (HTTP プロトコルオーバーヘッドを除きます)。ゼロの場合は「-」。

エントリ例

```
2662992
```

オブジェクトのサイズ

該当するオブジェクトの合計サイズ。

エントリ例

```
3462992
```

合計時間

バケットから見た、リクエストの転送の時間数 (ミリ秒単位)。これは、リクエストが受信されてから、レスポンスの最終バイトが送信されるまでの時間を計測した値です。クライアント側での計測値は、ネットワーク遅延により長くなる場合があります。

エントリ例

```
70
```

ターンアラウンド時間

Lightsail でリクエストの処理に要した時間数 (ミリ秒単位)。これは、リクエストの最終バイトが受信されてから、レスポンスの先頭バイトが送信されるまでの時間を計測した値です。

エントリ例

```
10
```

リファラー

HTTP Referer ヘッダーの値 (存在する場合)。一般に、HTTP ユーザーエージェント (ブラウザなど) は、このヘッダーをリクエスト作成時のリンクページや埋め込みページの URL に設定します。

エントリ例

```
"http://www.amazon.com/webservices"
```

ユーザーエージェント

HTTP User-Agent ヘッダーの値。

エントリ例

```
"curl/7.15.1"
```

バージョン ID

リクエストのバージョン ID、または オペレーションが versionId パラメータを取らない場合は「-」。

エントリ例

```
3HL4kqtJvjVBH40Nrjfkd
```

ホスト ID

x-amz-id-2 または Lightsail で拡張されたリクエスト ID。

エントリ例

```
s9lzHYrFp76ZVxRcpX9+5cjAnEH2ROuNkd2BHfIa6UkFVdtjf5mKR3/eTPFvsiP/XV/VLi31234=
```

署名バージョン

署名バージョン SigV2 か SigV4 (リクエストの認証に使用)、または - (認証されていないリクエストの場合)。

エントリ例

```
SigV2
```

暗号スイート

HTTPS リクエストに対してネゴシエートされた Secure Sockets Layer (SSL) 暗号、または HTTP リクエストに対してネゴシエートされた -。

エントリ例

```
ECDHE-RSA-AES128-GCM-SHA256
```

認証タイプ

使用されるリクエスト認証のタイプ。認証ヘッダーは AuthHeader、クエリ文字列 (署名付き URL) は QueryString、認証されていないリクエストには -。

エントリ例

```
AuthHeader
```

ホストヘッダー

Lightsail への接続に使用するエンドポイント。

エントリ例

```
s3.us-west-2.amazonaws.com
```

TLS のバージョン

クライアントによってネゴシエートされた Transport Layer Security (TLS) バージョン。値は TLSv1、TLSv1.1、TLSv1.2、- のいずれかです (TLS を使用しなかった場合)。

エントリ例

```
TLSv1.2
```

コピーオペレーションの追加ログ記録

コピーオペレーションには GET と PUT が含まれます。このため、コピーオペレーションの実行時には 2 つのログレコードが記録されます。前述のセクションでは、コピーオペレーションの PUT 部分に関連するフィールドを説明しています。次のリストでは、コピーオペレーションの GET 部分に関連するフィールドを説明します。

バケット所有者

コピーされたオブジェクトを格納するバケットの正規ユーザー ID。正規ユーザー ID は、別の形式の AWS アカウント ID です。正規ユーザー ID の詳細については、「AWS 全般のリファレンス」の「AWS アカウント ID」を参照してください。アカウントの正規ユーザー ID を検索する方法については、「AWS アカウントの正規ユーザー ID の検索」を参照してください。

エントリ例

```
79a59df900b949e55d96a1e698fbacedfd6e09d98eacf8f8d5218e7cd47ef2be
```

バケット

コピーターゲットのオブジェクトを格納するバケットの名前。

エントリ例

```
awsexamplebucket1
```

Time (時間)

リクエストが受信された時間。これらの日付と時刻は協定世界時 (UTC) です。strftime() terminology を使用した形式は次のようになります: [%d/%B/%Y:%H:%M:%S %z]

エントリ例

```
[06/Feb/2019:00:00:38 +0000]
```

リモート IP

リクエスタの表面上のインターネットアドレス。中間プロキシやファイアウォールにより、リクエストを作成したマシンの実際のアドレスが不明確になる場合があります。

エントリ例

```
192.0.2.3
```

リクエスタ

リクエスタの正規ユーザー ID。認証されていないリクエストの場合は - です。リクエスタが IAM ユーザーの場合、このフィールドは、リクエスタの IAM ユーザー名と IAM ユーザーが所属する AWS ルートアカウントと共に表示します。この識別子は、アクセスコントロールに使用されるものと同じです。

エントリ例

```
79a59df900b949e55d96a1e698fbacedfd6e09d98eacf8f8d5218e7cd47ef2be
```

リクエスト ID

各リクエストを一意に識別するために Lightsail で生成される文字列。

エントリ例

```
3E57427F33A59F07
```

操作

ここに表示されているオペレーション
は、SOAP.*operation*、REST.*HTTP_method.resource_type*、WEBSITE.*HTTP_method.resource_t*
または BATCH.DELETE.OBJECT と表示されます。

エントリ例

```
REST.COPY.OBJECT_GET
```

キー

コピーターゲットのオブジェクトのkey」部分。オペレーションがキーパラメータを取らない場合は「-」。

エントリ例

```
/photos/2019/08/puppy.jpg
```

Request-URI

HTTP リクエストメッセージの Request-URI の部分。

エントリ例

```
"GET /awsexamplebucket1/photos/2019/08/puppy.jpg?x-foo=bar"
```

HTTP ステータス

コピーオペレーションの GET 部分の HTTP ステータスの数値。

エントリ例

```
200
```

エラーコード

コピーオペレーションの GET 部分の Amazon S3 エラーコード、またはエラーがない場合は「-」。

エントリ例

```
NoSuchBucket
```

送信バイト数

送信されたレスポンスのバイト数 (HTTP プロトコルオーバーヘッドを除きます)。ゼロの場合は「-」。

エントリ例

```
2662992
```

オブジェクトのサイズ

該当するオブジェクトの合計サイズ。

エントリ例

```
3462992
```

合計時間

バケットから見た、リクエストの転送の時間数 (ミリ秒単位)。これは、リクエストが受信されてから、レスポンスの最終バイトが送信されるまでの時間を計測した値です。クライアント側での計測値は、ネットワーク遅延により長くなる場合があります。

エントリ例

```
70
```

ターンアラウンド時間

Lightsail でリクエストの処理に要した時間数 (ミリ秒単位)。これは、リクエストの最終バイトが受信されてから、レスポンスの先頭バイトが送信されるまでの時間を計測した値です。

エントリ例

```
10
```

リファラー

HTTP Referer ヘッダーの値 (存在する場合)。一般に、HTTP ユーザーエージェント (ブラウザなど) は、このヘッダーをリクエスト作成時のリンクページや埋め込みページの URL に設定します。

エントリ例

```
"http://www.amazon.com/webservices"
```

ユーザーエージェント

HTTP User-Agent ヘッダーの値。

エントリ例

```
"curl/7.15.1"
```

バージョン ID

コピー対象のオブジェクトのバージョン ID、または x-amz-copy-source ヘッダーでコピー元の一部として versionId パラメータを指定しなかった場合は「-」。

エントリ例

```
3HL4kqtJvjVBH40Nrjfkd
```

ホスト ID

x-amz-id-2 または Lightsail で拡張されたリクエスト ID。

エントリ例

```
s9lzHYrFp76ZVxRcpX9+5cjAnEH2ROuNkd2BHfIa6UkFVdtjf5mKR3/eTPFvsiP/XV/VLi31234=
```

署名バージョン

署名バージョン SigV2 か SigV4 (リクエストの認証に使用)、または - (認証されていないリクエストの場合)。

エントリ例

```
SigV2
```

暗号スイート

HTTPS リクエストに対してネゴシエートされた Secure Sockets Layer (SSL) 暗号、または HTTP リクエストに対してネゴシエートされた -。

エントリ例

```
ECDHE-RSA-AES128-GCM-SHA256
```

認証タイプ

使用されるリクエスト認証のタイプ。認証ヘッダーは AuthHeader、クエリ文字列 (署名付き URL) は QueryString、認証されていないリクエストには -。

エントリ例

```
AuthHeader
```

ホストヘッダー

Lightsail への接続に使用するエンドポイント。

エントリ例

```
s3.us-west-2.amazonaws.com
```

TLS のバージョン

クライアントによってネゴシエートされた Transport Layer Security (TLS) バージョン。値は TLSv1、TLSv1.1、TLSv1.2、- のいずれかです (TLS を使用しなかった場合)。

エントリ例

```
TLSv1.2
```

カスタムアクセスログ情報

リクエストのアクセスログレコードに保存するカスタム情報を含めることができます。これを行うには、リクエストの URL にカスタムクエリ文字列パラメータを追加します。Lightsail では、「x-」で始まるクエリ文字列パラメータは無視されますが、これらのパラメータはログレコードの Request-URI フィールドの一部として、リクエストのアクセスログレコードに追加されます。

例えば、GET の "s3.amazonaws.com/awsexamplebucket1/photos/2019/08/puppy.jpg?x-user=johndoe" リクエストは、"s3.amazonaws.com/awsexamplebucket1/

photos/2019/08/puppy.jpg" のリクエストと同じように動作します。ただし 、"x-user=johndoe" 文字列は関連付けられたログレコードの Request-URI フィールドに含まれている点が異なります。この機能は REST インターフェイスでのみ利用できます。

拡張可能なアクセスログの形式のプログラミングに関する考慮事項

場合によっては、新しいフィールドを各行末に追加することで、アクセスログレコードの形式を拡張することがあります。したがって、アクセスログを解析するコードは、理解できない可能性のある後続フィールドを処理するよう作成する必要があります。

Amazon Lightsail のバケットアクセスログ記録の有効化

アクセスログは、Amazon Lightsail オブジェクトストレージサービスのバケットに対して行われたリクエストの詳細なレコードを提供します。アクセスのログは、多くのアプリケーションに役立ちます。例えば、アクセスのログ情報は、セキュリティやアクセスの監査に役立ちます。また、顧客基盤について知るうえでも役立ちます。

デフォルトでは、Lightsail によってバケットへのアクセスのログは収集されません。ログ記録を有効にすると、Lightsail は、ソースバケットのアクセスログを選択されたターゲットバケットに配信します。ソースバケットとターゲットバケットの両方が同じ AWS リージョンにあり、同じアカウントによって所有されている必要があります。

アクセスログのレコードには、バケットに対するリクエストの詳細が取り込まれます。この情報には、リクエストタイプ、リクエストで指定したリソース、リクエストを処理した日時などが含まれます。このガイドでは、Lightsail API、AWS Command Line Interface (AWS CLI)、または AWS SDK を使用してバケットへのアクセスログ記録を有効または無効にする方法について説明します。

ログ記録の基本の詳細については、「バケットのアクセスログ」を参照してください。

目次

アクセスログ記録のコスト

バケットに対してアクセスのログ記録を有効にしても追加料金はかかりません。ただし、システムがバケットに配信するログファイルのストレージ領域は消費されます。ログはいつでも削除できます。

ログバケットのデータ転送が設定された月額許容範囲内にある場合、ログファイルの配信に対してデータ転送料金はかかりません。

ターゲットバケットでアクセスのログ記録が有効になっていない必要があります。ログの保存先のバケットとして、ソースバケットと同じリージョンにあるユーザー所有のバケットを指定できます。これにはソースバケット自体も含まれます。ただし、ログを管理しやすくするため、アクセスログは別のバケットに保存することをお勧めします。

AWS CLI を使用してアクセスログ記録を有効にする

バケットのアクセスログ記録を有効にするには、バケットがある各 AWS リージョン に専用のロギングバケットを作成することをお勧めします。その後、 アクセスログをその専用のロギングバケットに配信します。

AWS CLI を使用してアクセスログ記録を有効にするには、次の手順を実行します。

> **Note**
>
> この手順を続行する前に、AWS CLI をインストールして Lightsail 用に設定する必要があります。 詳細については、「Lightsail で使用するために AWS CLI を設定する」を参照してください。

1. ローカルコンピュータでコマンドプロンプトまたはターミナルウィンドウを開きます。
2. 次のコマンドを入力して、アクセスのログ記録を有効にします。

```
aws lightsail update-bucket --bucket-name SourceBucketName --access-log-config
 "{\"enabled\": true, \"destination\": \"TargetBucketName\", \"prefix\":
 \"ObjectKeyNamePrefix/\"}"
```

コマンドで、次のサンプルテキストを独自のテキストに置き換えます。

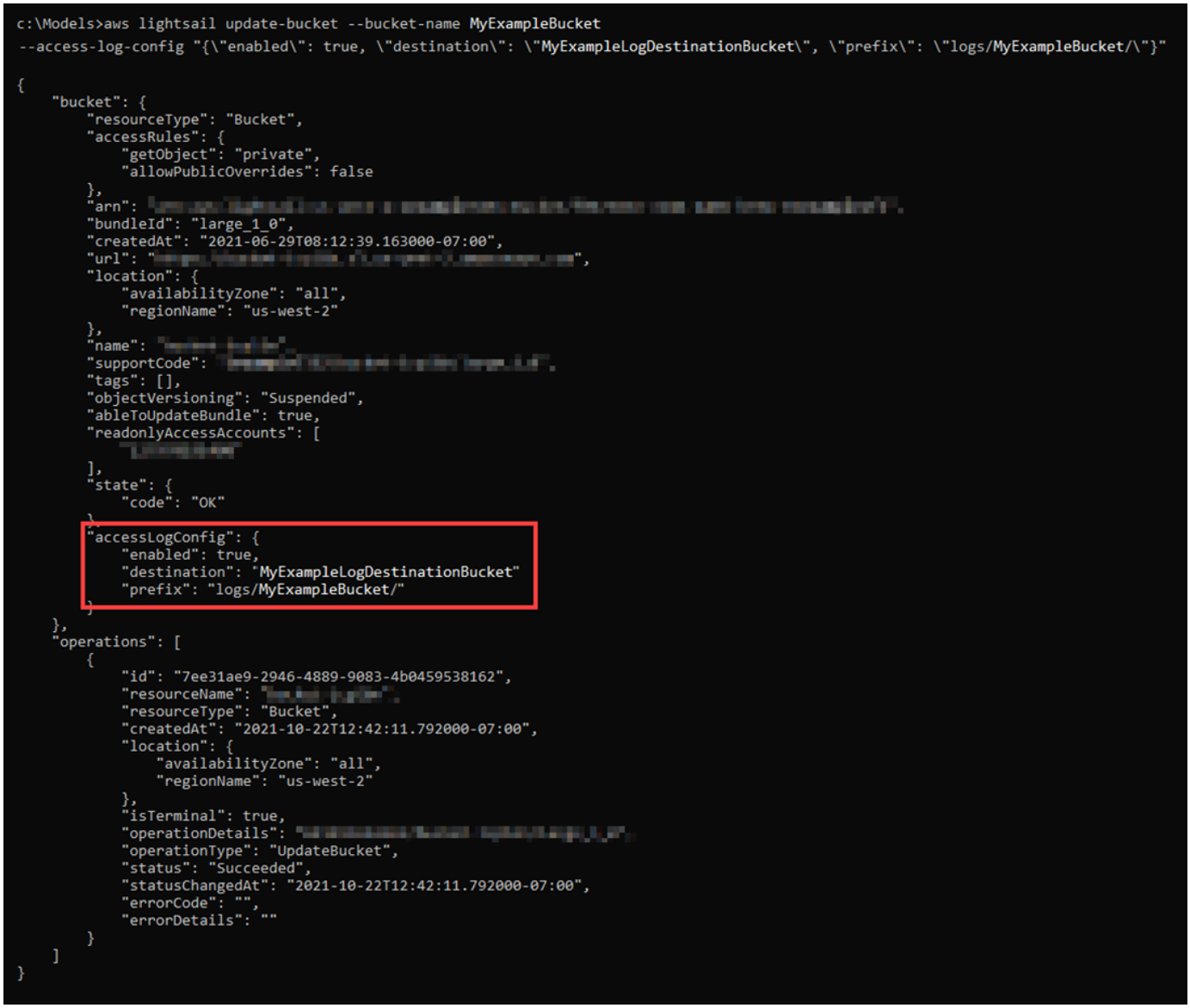
- *sourceBucketName* - アクセスログが作成されるソースバケットの名前。
- *TargetBucketName* - アクセスログを保存するターゲットバケットの名前。
- *ObjectKeyNamePrefix/* - アクセスログのオプションのオブジェクトキー名のプレフィックス。このプレフィックスは、スラッシュ (/) で終わる必要があります。

例

```
aws lightsail update-bucket --bucket-name MyExampleBucket --access-log-config
 "{\"enabled\": true, \"destination\": \"MyExampleLogDestinationBucket\", \"prefix
\": \"logs/MyExampleBucket/\"}"
```

この例では、*MyExampleBucket* はアクセスログが作成されるソースバケット、*MyExampleLogDestinationBucket* はアクセスログが保存される保存先バケット、*logs/MyExampleBucket/* はアクセスログのオブジェクトキー名のプレフィックスです。

コマンドを実行すると、次の例のような結果が表示されます。ソースバケットが更新され、アクセスログの生成が開始され、保存先バケットに保存されます。



AWS CLI を使用してのアクセスログ記録の無効化

AWS CLI を使用してアクセスログ記録を無効にするには、次の手順を実行します。

> **Note**
>
> この手順を続行する前に、AWS CLI をインストールして Lightsail 用に設定する必要があります。　詳細については、「Lightsail で使用するために AWS CLI を設定する」を参照してください。

1. ローカルコンピュータでコマンドプロンプトまたはターミナルウィンドウを開きます。
2. 次のコマンドを入力して、アクセスのログ記録を無効にします。

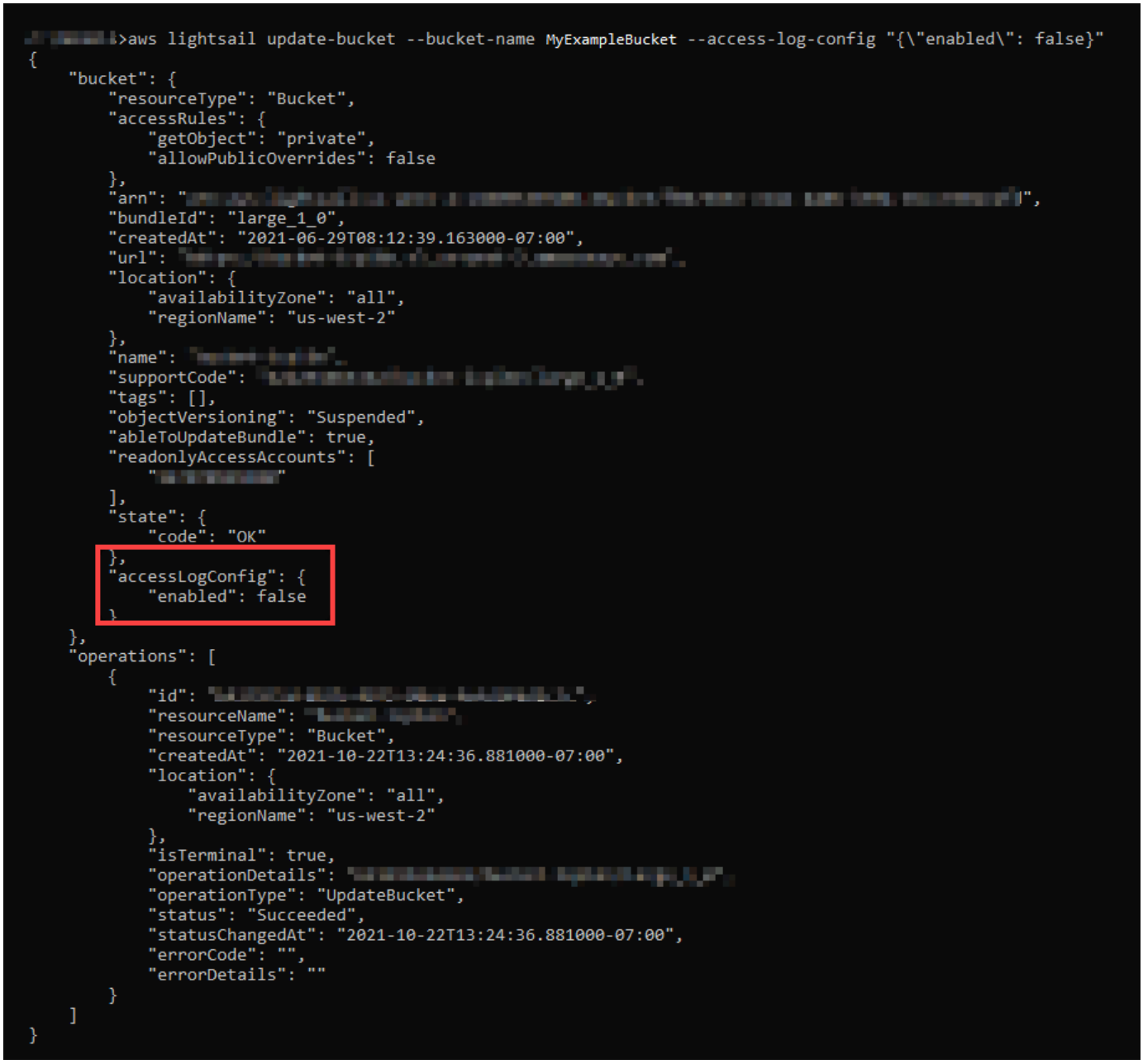
```
aws lightsail update-bucket --bucket-name SourceBucketName --access-log-config
 "{\"enabled\": false}"
```

コマンドで、*SourceBucketName* をアクセスログを無効にするソースバケットの名前に置き換えます。

例

```
aws lightsail update-bucket --bucket-name MyExampleBucket --access-log-config
 "{\"enabled\": false}"
```

コマンドを実行すると、次の例のような結果が表示されます。



Amazon Lightsail でバケットアクセスログを使用してリクエストを識別する

このガイドでは、アクセスログを使用したバケットへのリクエストの識別方法を説明します。詳細については、「バケットアクセスのログ」を参照してください。

目次

- Amazon S3 アクセスログを使用してオブジェクトアクセスリクエストの識別する

Amazon Athena を使用してリクエストのアクセスログをクエリする

Amazon Athena を使用して、アクセスログのバケットへのリクエストをクエリ、識別することができます。

Lightsail は、アクセスログをオブジェクトとして Lightsail バケットに保存します。多くの場合、ログを分析できるツールを使用する方が簡単です。Athena はオブジェクトの分析をサポートしてしるため、アクセスログに対してクエリを実行するのに使用できます。

例

次の例は、Amazon Athena でバケットサーバーアクセスログをクエリする方法を示しています。

> **Note**
>
> Athena クエリでバケットの場所を指定するには、ログが配信されるターゲットバケット名とターゲットプレフィックスを、次のような S3 URI でフォーマットする必要があります: `s3://DOC-EXAMPLE-BUCKET1-logs/prefix/`

1. https://console.aws.amazon.com/athena/ で Athena コンソールを開きます。
2. クエリエディタで、次のようなコマンドを実行します。

```
create database bucket_access_logs_db
```

> **Note**
>
> ベストプラクティスとして、データベースは、S3 バケットと同じ AWS リージョンで作成することをお勧めします。

3. クエリエディタで、次のようなコマンドを実行して、ステップ 2 で作成したデータベースでテーブルスキーマを作成します。STRING および BIGINT データ型の値はアクセスログのプロパティです。これらのプロパティは Athena でクエリできます。LOCATION の場合は、前述のようにバケットとプレフィックスパスを入力します。

```
CREATE EXTERNAL TABLE `s3_access_logs_db.mybucket_logs`(
  `bucketowner` STRING,
```

```
  `bucket_name` STRING,
  `requestdatetime` STRING,
  `remoteip` STRING,
  `requester` STRING,
  `requestid` STRING,
  `operation` STRING,
  `key` STRING,
  `request_uri` STRING,
  `httpstatus` STRING,
  `errorcode` STRING,
  `bytessent` BIGINT,
  `objectsize` BIGINT,
  `totaltime` STRING,
  `turnaroundtime` STRING,
  `referrer` STRING,
  `useragent` STRING,
  `versionid` STRING,
  `hostid` STRING,
  `sigv` STRING,
  `ciphersuite` STRING,
  `authtype` STRING,
  `endpoint` STRING,
  `tlsversion` STRING)
ROW FORMAT SERDE
  'org.apache.hadoop.hive.serde2.RegexSerDe'
WITH SERDEPROPERTIES (
  'input.regex'='([^ ]*) ([^ ]*) \\[(.*?)\\] ([^ ]*) ([^ ]*) ([^ ]*) ([^ ]*)
 ([^ ]*) (\"[^\"]*\"|-) (-|[0-9]*) ([^ ]*) ([^ ]*) ([^ ]*) ([^ ]*) ([^ ]*) ([^ ]*)
 (\"[^\"]*\"|-) ([^ ]*)(?: ([^ ]*) ([^ ]*) ([^ ]*) ([^ ]*) ([^ ]*) ([^ ]*))?.*$')
STORED AS INPUTFORMAT
  'org.apache.hadoop.mapred.TextInputFormat'
OUTPUTFORMAT
  'org.apache.hadoop.hive.ql.io.HiveIgnoreKeyTextOutputFormat'
LOCATION
  's3://doc-example-bucket1-logs/prefix/'
```

4. ナビゲーションペインにある、[データベース] で、データベースを選択します。
5. [テーブル] で、テーブル名の横にある、[Preview table (テーブルのプレビュー)] を選択します。

   [結果] ペインに、サーバーアクセスログのデータ (bucketowner、bucket、requestdatetime など) が表示されます。これは、Athena テーブルが正常に作成されたことを意味します。これで、バケットサーバーアクセスログのクエリを実行できます。

例 - 誰がいつオブジェクトを削除したか (タイムスタンプ、IP アドレス、および IAM ユーザー) を表示する

```
SELECT RequestDateTime, RemoteIP, Requester, Key
FROM s3_access_logs_db.mybucket_logs
WHERE key = 'images/picture.jpg' AND operation like '%DELETE%';
```

例 - IAM ユーザーによって実行されたすべてのオペレーションを表示する

```
SELECT *
FROM s3_access_logs_db.mybucket_logs
WHERE requester='arn:aws:iam::123456789123:user/user_name';
```

例 - 特定の期間にオブジェクトに対して実行されたすべてのオペレーションを表示する

```
SELECT *
FROM s3_access_logs_db.mybucket_logs
WHERE Key='prefix/images/picture.jpg'
    AND parse_datetime(RequestDateTime,'dd/MMM/yyyy:HH:mm:ss Z')
    BETWEEN parse_datetime('2017-02-18:07:00:00','yyyy-MM-dd:HH:mm:ss')
    AND parse_datetime('2017-02-18:08:00:00','yyyy-MM-dd:HH:mm:ss');
```

例 - 特定の期間に特定の IP アドレスによって送信されたデータの量を表示する

```
SELECT SUM(bytessent) AS uploadTotal,
      SUM(objectsize) AS downloadTotal,
      SUM(bytessent + objectsize) AS Total
FROM s3_access_logs_db.mybucket_logs
WHERE RemoteIP='1.2.3.4'
AND parse_datetime(RequestDateTime,'dd/MMM/yyyy:HH:mm:ss Z')
BETWEEN parse_datetime('2017-06-01','yyyy-MM-dd')
AND parse_datetime('2017-07-01','yyyy-MM-dd');
```

Amazon S3 アクセスログを使用してオブジェクトアクセスリクエストの識別する

アクセスログに対するクエリを使用して、GET、PUT、DELETE などのオペレーションに対するオブジェクトアクセスリクエストを識別し、それらのリクエストに関する詳細情報を確認することができます。

次の Amazon Athena クエリの例は、サーバーアクセスログからバケットに対するすべての PUT オブジェクトリクエストを取得する方法を示しています。

例 - 一定期間内に PUT オブジェクトリクエストを送信しているすべてのリクエスタを表示する

```
SELECT Bucket, Requester, RemoteIP, Key, HTTPStatus, ErrorCode, RequestDateTime
FROM s3_access_logs_db
WHERE Operation='REST.PUT.OBJECT' AND
parse_datetime(RequestDateTime,'dd/MMM/yyyy:HH:mm:ss Z')
BETWEEN parse_datetime('2019-07-01:00:42:42','yyyy-MM-dd:HH:mm:ss')
AND
parse_datetime('2019-07-02:00:42:42','yyyy-MM-dd:HH:mm:ss')
```

次の Amazon Athena クエリの例は、サーバーアクセスログから Amazon S3 に対するすべての GET オブジェクトリクエストを取得する方法を示しています。

例 - 一定期間内に GET オブジェクトリクエストを送信しているすべてのリクエスタを表示する

```
SELECT Bucket, Requester, RemoteIP, Key, HTTPStatus, ErrorCode, RequestDateTime
FROM s3_access_logs_db
WHERE Operation='REST.GET.OBJECT' AND
parse_datetime(RequestDateTime,'dd/MMM/yyyy:HH:mm:ss Z')
BETWEEN parse_datetime('2019-07-01:00:42:42','yyyy-MM-dd:HH:mm:ss')
AND
parse_datetime('2019-07-02:00:42:42','yyyy-MM-dd:HH:mm:ss')
```

次の Amazon Athena のクエリの例は、S3 バケットへのすべての匿名リクエストをサーバーアクセスログから取得する方法を示しています。

例 - 特定の期間にバケットにリクエストを行っているすべての匿名リクエスタを表示する

```
SELECT Bucket, Requester, RemoteIP, Key, HTTPStatus, ErrorCode, RequestDateTime
FROM s3_access_logs_db.mybucket_logs
WHERE Requester IS NULL AND
parse_datetime(RequestDateTime,'dd/MMM/yyyy:HH:mm:ss Z')
BETWEEN parse_datetime('2019-07-01:00:42:42','yyyy-MM-dd:HH:mm:ss')
AND
parse_datetime('2019-07-02:00:42:42','yyyy-MM-dd:HH:mm:ss')
```

> **Note**
>
> - ニーズに合わせられるように、データ範囲を変更することができます。

- このクエリの例は、セキュリティのモニタリングにも役立つ場合があります。予期しないまたは不正な IP アドレス/リクエスタからの `PutObject` または `GetObject` コールの結果を確認し、バケットへの匿名リクエストを特定できます。
- このクエリでは、ログ記録が有効になった時間以降の情報のみ取得されます。

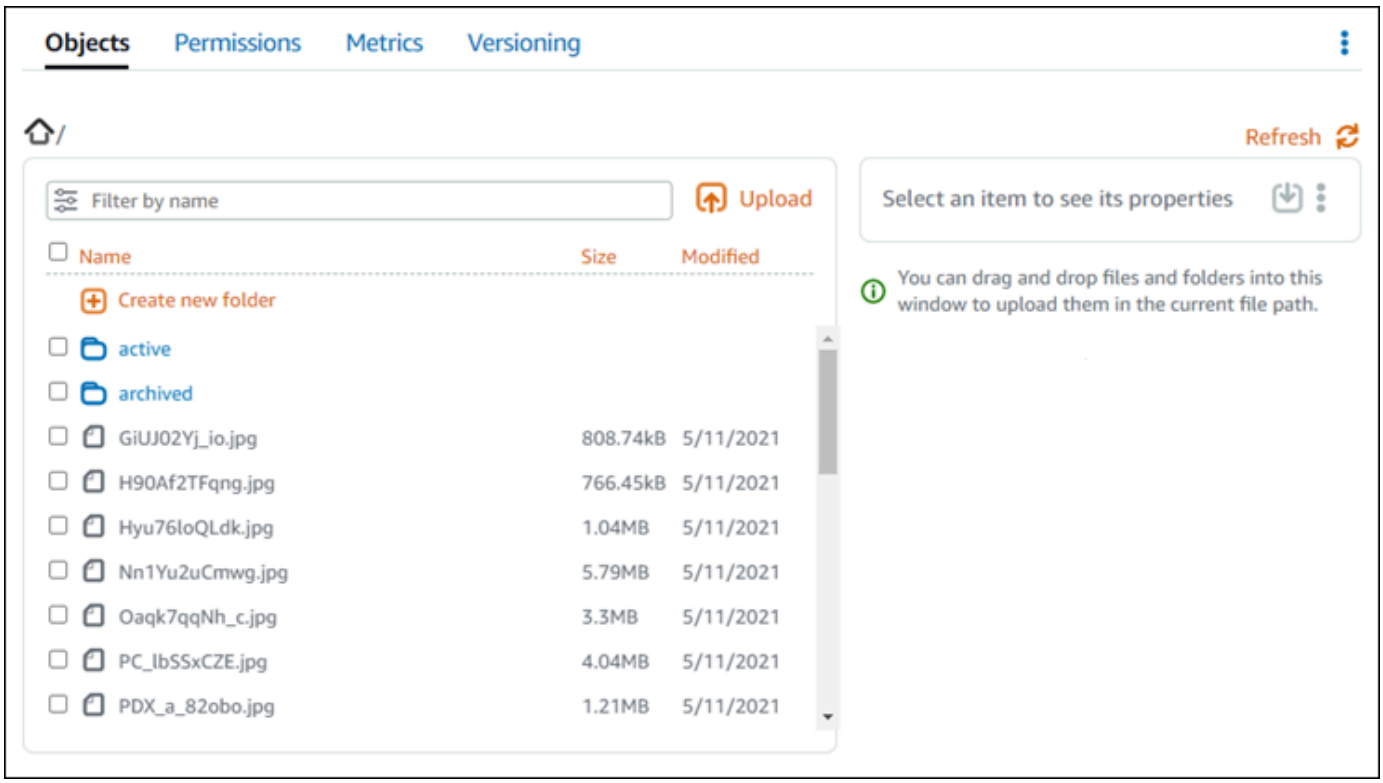
Amazon Lightsail のバケットオブジェクト

Lightsail コンソールを使用して、Amazon Lightsail オブジェクトストレージサービスのバケットに保存されているすべてのオブジェクトを表示できます。AWS Command Line Interface (AWS CLI) および AWS SDK を使用して、バケット内のオブジェクトキーをリスト化することもできます。バケットについての詳細は、「オブジェクトストレージ」を参照してください。

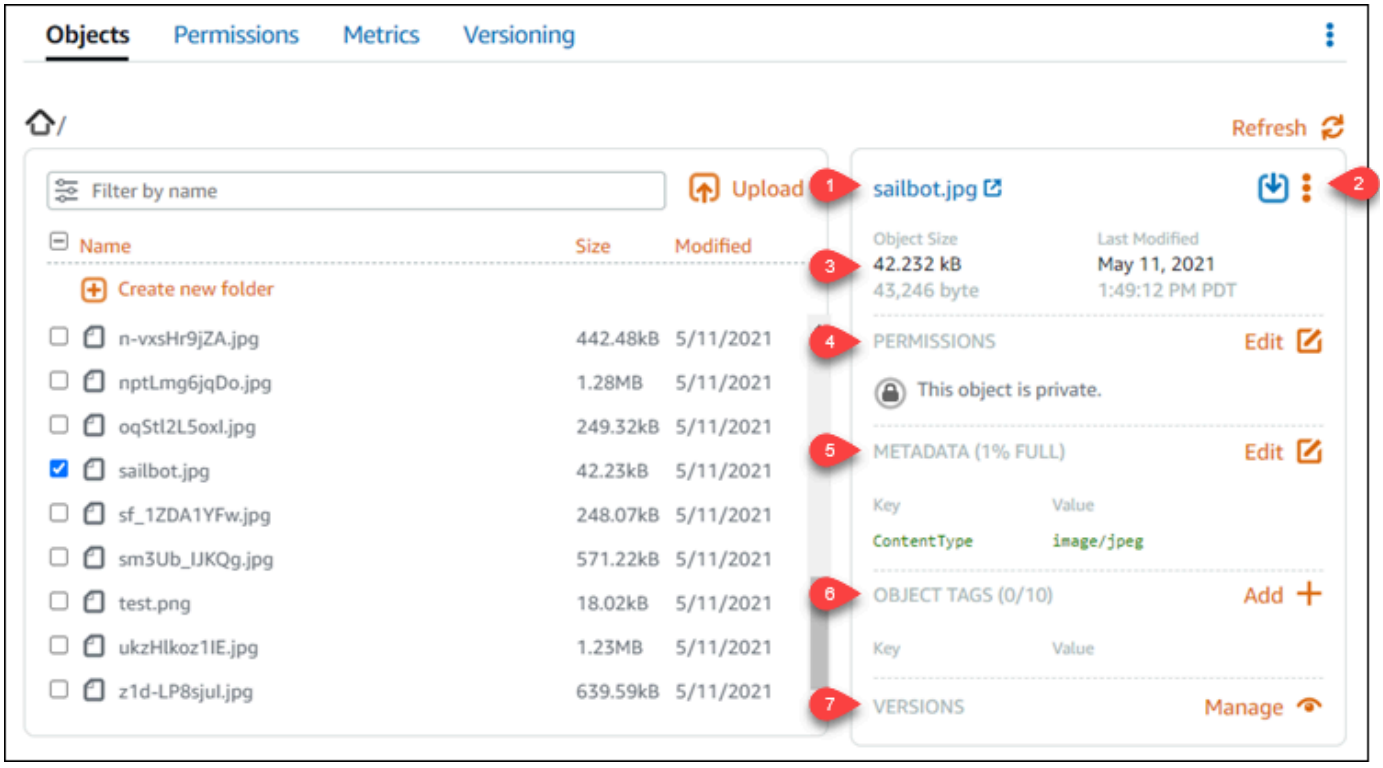
Lightsail コンソールを使用してオブジェクトをフィルターする

Lightsail コンソールを使用してバケットに格納されたオオブジェクトストレージを表示するには、以下の手順を実行します。

1. Lightsail コンソールにサインインします。
2. Lightsail のホームページで [Storage] (ストレージ) タブを選択します。
3. オブジェクトを表示するバケットの名前を選択します。
4. [オブジェクト] タブの [オブジェクトブラウザ] ペインには、バケットに保存されているオブジェクトとフォルダが表示されます。



5. プロパティを表示するオブジェクトのロケーションを見つけます。
6. プロパティを表示するオブジェクトの横にチェックマークを追加します。
7. ページの右側にある [オブジェクトのプロパティ] ペインに、オブジェクトに関する情報が表示されます。



表示される情報には、次の情報が含まれます。

1. オブジェクトを表示およびダウンロードするリンク。
2. アクションメニュー (:) を使用して、オブジェクトをコピーまたは削除します。オブジェクトのコピーと削除の詳細については、「Amazon Lightsail バケット内のオブジェクトのコピーまたは移動」および「バケットのオブジェクトの削除」を参照してください。
3. オブジェクトのサイズ、および最終更新タイムスタンプ。
4. 個々のオブジェクトのアクセス許可は、プライベートまたは公開 (読み取り専用) です。バケットのアクセス許可の詳細については、「バケットのアクセス許可」を参照してください。
5. オブジェクトのメタデータ。コンテンツタイプ (ContentType) キーは、現時点で Lightsail オブジェクトストレージサービスがサポートする唯一のメタデータです。
6. オブジェクトキーバリュータグ 詳細については、「バケットオブジェクトにタグを付ける」を参照してください。
7. オブジェクトの保存されたバージョンを管理するオプション。詳細については、「バケットでのオブジェクトのバージョニングの有効化と一時停止」を参照してください。

> **Note**
>
> 複数のオブジェクトを選択すると、[オブジェクトのプロパティ] ペインには、選択したオブジェクトの合計サイズのみが表示されます。

AWS CLI を使用してオブジェクトを表示するには、

AWS Command Line Interface (AWS CLI) を使用して、バケットのオブジェクトのキーをリスト化するには、次の手順を実行します。これは、`list-objects-v2` コマンドを使用して実行できます。詳細については、「AWS CLI コマンドリファレンス」の「list-objects-v2」を参照してください。

> **Note**
>
> この手順を続行する前に、AWS CLI をインストールし、Lightsail と Amazon S3 用に設定する必要があります。詳細については、「Amazon Lightsail で使用するために AWS Command Line Interface を設定する」を参照してください。

1. コマンドプロンプトまたはターミナルウィンドウを開きます。
2. 以下のいずれかのコマンドを入力します。

   - 次のコマンドを入力して、バケット内のすべてのオブジェクトキーをリスト化します。

     ```
     aws s3api list-objects-v2 --bucket BucketName --query "Contents[].{Key: Key,
      Size: Size}"
     ```

     コマンドで、*BucketName* をすべてのオブジェクトをリスト化するバケットの名前に置き換えます。

   - 次のコマンドを入力して、特定のオブジェクトキー名のプレフィックスで始まるオブジェクトをリスト化します。

     ```
     aws s3api list-objects-v2 --bucket BucketName --prefix ObjectKeyNamePrefix --
     query "Contents[].{Key: Key, Size: Size}"
     ```

     コマンドで、次のサンプルテキストを独自のテキストに置き換えます。

- *BucketName* - すべてのオブジェクトをリストするバケット名。
- *ObjectKeyNamePrefix* - オブジェクトキー名のプレフィックスで、指定されたプレフィックスで始まるキーへのレスポンスを制限します。

> **Note**
>
> これらのコマンドは、--query パラメータを使用して、list-objects-v2 リクエストのレスポンスを各オブジェクトのキーバリューとサイズにフィルタリングします。

例:

バケット内のすべてのオブジェクトキーをリスト化

```
aws s3api list-objects-v2 --bucket DOC-EXAMPLE-BUCKET --query "Contents[].{Key:
 Key, Size: Size}"
```

前述のコマンドでは、次の例に示すような結果が表示されます。

```
C:\>aws s3api list-objects-v2 --bucket DOC-EXAMPLE-BUCKET --query "Contents[].{Key: Key, Size: Size}"
[
    {
        "Key": "GiUJ02Yj_io.jpg",
        "Size": 828150
    },
    {
        "Key": "H90Af2TFqng.jpg",
        "Size": 784846
    },
    {
        "Key": "Hyu76loQLdk.jpg",
        "Size": 1086363
    },
    {
        "Key": "Nn1Yu2uCmwg.jpg",
        "Size": 6075006
    },
    {
        "Key": "Oaqk7qqNh_c.jpg",
        "Size": 3458557
    },
    {
        "Key": "PC_lbSSxCZE.jpg",
        "Size": 4239636
    },
    {
        "Key": "PDX_a_82obo.jpg"
```

オブジェクトキーのリストは、archived/オブジェクトキー名のプレフィックスで始まります:

```
aws s3api list-objects-v2 --bucket DOC-EXAMPLE-BUCKET --prefix archived/ --query
 "Contents[].{Key: Key, Size: Size}"
```

前述のコマンドでは、次の例に示すような結果が表示されます。

```
C:\>aws s3api list-objects-v2  --bucket DOC-EXAMPLE-BUCKET --prefix archived/ --query "Contents[].{Key: Key, Size: Size}"
[
    {
        "Key": "archived/",
        "Size": 0
    },
    {
        "Key": "archived/1_CMoFsPfso.jpg",
        "Size": 2561865
    },
    {
        "Key": "archived/3y1zF4hIPCg.jpg",
        "Size": 6404907
    },
    {
        "Key": "archived/5IHz5WhosQE.jpg",
        "Size": 2377975
    },
    {
        "Key": "archived/sailbot.jpg",
        "Size": 43246
    }
]
```

バケットとオブジェクトを管理する

これらは、Lightsail オブジェクトストレージバケットを管理する一般的な手順です。

1. Amazon Lightsail オブジェクトストレージサービスでのオブジェクトとバケットについて説明します。詳細については、「Amazon Lightsail のオブジェクトストレージ」を参照してください。
2. Amazon Lightsail でバケットに付けることができる名前について説明します。詳細については、「Amazon Lightsail でのバケットの命名規則」をご参照ください。
3. バケットを作成して、Lightsail オブジェクトストレージサービスの使用を開始します。詳細については、「Amazon Lightsail におけるバケットの作成 」を参照してください。
4. バケットのセキュリティのベストプラクティスと、バケットに設定できるアクセス許可について説明します。バケット内のすべてのオブジェクトをパブリックまたはプライベートにすることも、オブジェクトを個別に選択してパブリックにすることもできます。また、アクセスキーを作成し、インスタンスをバケットに追加し、他の AWS アカウントにアクセス権を付与することで、バケットへのアクセスを許可することもできます。詳細については、「Amazon Lightsail オブジェクトストレージのセキュリティベストプラクティス」と「Amazon Lightsail でのバケットのアクセス許可を理解する」を参照してください。

バケットのアクセス許可について理解したら、以下のガイドを参照してバケットへのアクセスを許可してください。

- Amazon Lightsail でのバケットへのパブリックアクセスをブロックする
- Amazon Lightsail でのバケットのアクセス許可の設定
- Amazon Lightsail でのバケット内の個々のオブジェクトに対するアクセス許可の設定
- Amazon Lightsail でのバケットのアクセスキーの作成
- Amazon Lightsail でのバケットのリソースアクセスの設定
- Amazon Lightsail でのバケットのクロスアカウントアクセスの設定

5. バケットのアクセスログの記録を有効にする方法と、アクセスログを使用してバケットのセキュリティを監査する方法について説明します。詳細については、以下のガイドを参照してください。
   - Amazon Lightsail オブジェクトストレージサービスのバケットへのアクセスのログ記録
   - Amazon Lightsail オブジェクトストレージサービスのバケットのアクセスログの形式
   - Amazon Lightsail オブジェクトストレージサービスのバケットへのアクセスのログ記録を有効にする
   - Amazon Lightsailでのバケットのアクセスログを使用するリクエストの特定
6. Lightsail でバケットを管理する機能をユーザーに付与する IAM ポリシーを作成します。詳細については、「Amazon Lightsail でバケットを管理する IAM ポリシー」を参照してください。
7. バケット内のオブジェクトにラベルを付けて識別する方法について説明します。詳細については、「Amazon Lightsail でのオブジェクトキー名を理解する」を参照してください。
8. ファイルをアップロードしてバケット内のオブジェクトを管理する方法について説明します。詳細については、以下のガイドを参照してください。
   - Amazon Lightsail のバケットにファイルをアップロードする
   - マルチパートアップロードを使用した Amazon Lightsail のバケットへのファイルのアップロード
   - Amazon Lightsail のバケット内のオブジェクトの表示
   - Amazon Lightsail のバケット内のオブジェクトのコピーまたは移動
   - Amazon Lightsail のバケットからのオブジェクトのダウンロード
   - Amazon Lightsail のバケット内のオブジェクトのフィルタリング
   - Amazon Lightsail のバケット内のオブジェクトのタグ付け
   - Amazon Lightsail のバケット内のオブジェクトの削除

9. オブジェクトのバージョニングを有効にすると、バケットに保存されたあらゆるオブジェクトのあらゆるバージョンを保存、取得、復元します。詳細については、「Amazon Lightsail のバケットでのオブジェクトのバージョニングの有効化と一時停止」を参照してください。

10.オブジェクトのバージョニングを有効にすると、バケット内のオブジェクトの以前のバージョンを復元できます。詳細については、「Amazon Lightsail のバケット内のオブジェクトの以前のバージョンの復元」を参照してください。

11.バケットの使用率を監視します。詳細については、「Amazon Lightsail でのバケットのメトリクスの表示」を参照してください。

12.バケットの使用率がしきい値を超えたときにバケットメトリクスが通知されるよう、アラームを設定します。詳細については、「Amazon Lightsail でのバケットメトリクスアラームの作成」を参照してください。

13.ストレージとネットワーク転送量が不足している場合は、バケットのストレージプランを変更します。詳細については、「Amazon Lightsail のバケットのプランの変更」を参照してください。

14.バケットを他のリソースに接続する方法について説明します。詳細については、以下のチュートリアルを参照してください。

- チュートリアル: WordPress インスタンスの Amazon Lightsail バケットへの接続
- チュートリアル: Lightsail コンテンツ配信ネットワークディストリビューションでの Amazon Lightsail バケットの使用

15.使用しなくなったバケットを削除します。詳細については、「Amazon Lightsail でのバケットの削除」を参照してください。

トピック

- Amazon Lightsail のバケットオブジェクトをコピーまたは移動する
- Amazon Lightsail のバケットオブジェクトを削除する
- Amazon Lightsail のバケットからオブジェクトをダウンロードする
- Amazon Lightsail のバケットオブジェクトをフィルタリングする
- Amazon Lightsail のオブジェクトのバージョニングを有効化および一時停止する
- Amazon Lightsail でバケットオブジェクトの以前のバージョンを復元する
- Amazon Lightsail のバケットオブジェクトをタグ付けする

Amazon Lightsail のバケットオブジェクトをコピーまたは移動する

Amazon Lightsail オブジェクトストレージサービスのバケットに既に保存されているオブジェクトをコピーできます。このガイドでは、Lightsail コンソールで AWS Command Line Interface (AWS CLI) を使用しオブジェクトをコピーする方法を紹介します。バケット内のオブジェクトをコピーして、複製コピーを作成したり、オブジェクトの名前を変更したり、Lightsail のロケーションをまたいで移動させることができます (例えば、ある AWS リージョン から Lightsail が利用可能な別のリージョンへのオブジェクトの移動など)。ロケーション間のオブジェクトのコピーは、AWS API、AWS SDK、AWS Command Line Interface (AWS CLI) を使用してのみ行えます。

バケットについての詳細は、「オブジェクトストレージ」を参照してください。

オブジェクトのコピーに関する制約事項

Lightsail コンソールでは、2 GB までのサイズのオブジェクトのコピーを作成することができます。AWS Command Line Interface (AWS CLI)、AWS API、AWS SDK を使用する場合、1 回のオブジェクトアクションでコピーを作成できるオブジェクトのサイズは最大 5 GB です。5 GB を超えるオブジェクトをコピーするには、AWS CLI、AWS API、AWS SDK のマルチパートアップロードアクションを使用する必要があります。詳細については、「マルチパートアップロードを使用してバケットにファイルをアップロードする」を参照してください。

Lightsailコンソールを使用してオブジェクトをコピーする

Lightsailコンソールを使用し、バケットに格納されたオブジェクトをコピーするには、以下の手順を実行します。バケット内のオブジェクトを移動するには、そのオブジェクトを新しい場所にコピーし、元のオブジェクトを削除する必要があります。

1. Lightsail コンソールにサインインします。
2. Lightsailのホームページで [ストレージ] のタブを選択します。
3. オブジェクトのコピー先のバケットの名前を選択します。
4. オブジェクトのタブで、オブジェクトブラウザペインを使用し、オブジェクトのコピー先のロケーションを参照します。
5. コピーするオブジェクトの隣のチェックマークを入れます。
6. オブジェクト情報のウィンドウで、アクション (:) メニューを選択し、にコピーします。
7. 送信先の選択ペインで、選択したオブジェクトをコピーするバケット内のロケーションを参照します。送信先パステキストボックスにフォルダ名を入力して、新しいパスを作成することもできます。

8. 選択したコピー先または指定したコピー先にオブジェクトをコピーするためには、コピーを選択します。それ以外の場合は、[いいえ、キャンセル] を選択します。

   オブジェクトが正常にコピーされると、コピー完了のメッセージが表示されます。オブジェクトの移動を目的としていた場合は、元のオブジェクトを削除する必要があります。詳細については、「バケットオブジェクトを削除する」を参照してください。

AWS CLIを使用してオブジェクトをコピー

AWS Command Line Interface (AWS CLI) を使用してバケットのオブジェクトをコピーするには、以下の手順を実行します。これは、copy-object コマンドを使用して実行できます。詳細については、「AWS CLI コマンドリファレンス」の「copy-object」を参照してください。

> **Note**
>
> この手順を続行する前に、AWS CLI をインストールし、Lightsail と Amazon S3 用に設定する必要があります。詳細については、「Lightsail で使用するために AWS CLI を設定する」を参照してください。

1. ターミナルまたはコマンドプロントウィンドウを開きます。
2. バケット内のオブジェクトをコピーするには、次のコマンドを入力します。

   ```
   aws s3api copy-object --copy-source SourceBucketNameAndObjectKey --
   key DestinationObjectKey --bucket DestinationBucketName --acl bucket-owner-full-
   control
   ```

   コマンドで、次のサンプルテキストを独自のテキストに置き換えます。

   - *SourceBucketNameAndObjectKey* - ソースオブジェクトが現在存在するバケットの名前と、コピーされるオブジェクトのフルオブジェクトキー。たとえば、DOC-EXAMPLE-BUCKETバケットからオブジェクトimages/sailbot.jpgをコピーするには、DOC-EXAMPLE-BUCKET/images/sailbot.jpgを指定します。
   - *DestinationObjectKey* - 新しいオブジェクトのコピーのフルオブジェクトキー。
   - *DestinationBucket* - 送信先バケットの名前。

   例:

- バケット内のオブジェクトを同じバケット内にコピーする:

```
aws s3api copy-object --copy-source DOC-EXAMPLE-BUCKET/images/sailbot.jpg --
key media/sailbot.jpg --bucket DOC-EXAMPLE-BUCKET --acl bucket-owner-full-control
```

- バケットから別のバケットへオブジェクトをコピーする:

```
aws s3api copy-object --copy-source DOC-EXAMPLE-BUCKET-1/images/sailbot.jpg --
key images/sailbot.jpg --bucket DOC-EXAMPLE-BUCKET-2 --acl bucket-owner-full-
control
```

以下の例のような結果が表示されるはずです。

```
C:\>aws s3api copy-object --copy-source DOC-EXAMPLE-BUCKET/images/sailbot.jpg --key images/archived/sailbot.jpg --bucket DOC-EXAMPLE-BUCKET
{
    "ServerSideEncryption": "AES256",
    "CopyObjectResult": {
        "ETag": "\"694d34example91d92d64f342aa234c3\"",
        "LastModified": "2021-05-10T05:35:42+00:00"
    }
}
```

バケットとオブジェクトを管理する

これらは、Lightsail オブジェクトストレージバケットを管理する一般的な手順です。

1. Amazon Lightsail オブジェクトストレージサービスでのオブジェクトとバケットについて説明します。詳細については、「Amazon Lightsail のオブジェクトストレージ」を参照してください。
2. Amazon Lightsail でバケットに付けることができる名前について説明します。詳細については、「Amazon Lightsail でのバケットの命名規則」をご参照ください。
3. バケットを作成して、Lightsail オブジェクトストレージサービスの使用を開始します。詳細については、「Amazon Lightsail におけるバケットの作成 」を参照してください。
4. バケットのセキュリティのベストプラクティスと、バケットに設定できるアクセス許可について説明します。バケット内のすべてのオブジェクトをパブリックまたはプライベートにすることも、オブジェクトを個別に選択してパブリックにすることもできます。また、アクセスキーを作成し、インスタンスをバケットに追加し、他の AWS アカウントにアクセス権を付与することで、バケットへのアクセスを許可することもできます。詳細については、「Amazon Lightsail オブジェクトストレージのセキュリティベストプラクティス」と「Amazon Lightsail でのバケットのアクセス許可を理解する」を参照してください。

バケットのアクセス許可について理解したら、以下のガイドを参照してバケットへのアクセスを許可してください。

- Amazon Lightsail でのバケットへのパブリックアクセスをブロックする
- Amazon Lightsail でのバケットのアクセス許可の設定
- Amazon Lightsail でのバケット内の個々のオブジェクトに対するアクセス許可の設定
- Amazon Lightsail でのバケットのアクセスキーの作成
- Amazon Lightsail でのバケットのリソースアクセスの設定
- Amazon Lightsail でのバケットのクロスアカウントアクセスの設定

5. バケットのアクセスログの記録を有効にする方法と、アクセスログを使用してバケットのセキュリティを監査する方法について説明します。詳細については、以下のガイドを参照してください。
   - Amazon Lightsail オブジェクトストレージサービスのバケットへのアクセスのログ記録
   - Amazon Lightsail オブジェクトストレージサービスのバケットのアクセスログの形式
   - Amazon Lightsail オブジェクトストレージサービスのバケットへのアクセスのログ記録を有効にする
   - Amazon Lightsailでのバケットのアクセスログを使用するリクエストの特定
6. Lightsail でバケットを管理する機能をユーザーに付与する IAM ポリシーを作成します。詳細については、「Amazon Lightsail でバケットを管理する IAM ポリシー」を参照してください。
7. バケット内のオブジェクトにラベルを付けて識別する方法について説明します。詳細については、「Amazon Lightsail でのオブジェクトキー名を理解する」を参照してください。
8. ファイルをアップロードしてバケット内のオブジェクトを管理する方法について説明します。詳細については、以下のガイドを参照してください。
   - Amazon Lightsail のバケットにファイルをアップロードする
   - マルチパートアップロードを使用した Amazon Lightsail のバケットへのファイルのアップロード
   - Amazon Lightsail のバケット内のオブジェクトの表示
   - Amazon Lightsail のバケット内のオブジェクトのコピーまたは移動
   - Amazon Lightsail のバケットからのオブジェクトのダウンロード
   - Amazon Lightsail のバケット内のオブジェクトのフィルタリング
   - Amazon Lightsail のバケット内のオブジェクトのタグ付け
   - Amazon Lightsail のバケット内のオブジェクトの削除

9. オブジェクトのバージョニングを有効にすると、バケットに保存されたあらゆるオブジェクトのあらゆるバージョンを保存、取得、復元します。詳細については、「Amazon Lightsail のバケットでのオブジェクトのバージョニングの有効化と一時停止」を参照してください。
10. オブジェクトのバージョニングを有効にすると、バケット内のオブジェクトの以前のバージョンを復元できます。詳細については、「Amazon Lightsail のバケット内のオブジェクトの以前のバージョンの復元」を参照してください。
11. バケットの使用率を監視します。詳細については、「Amazon Lightsail でのバケットのメトリクスの表示」を参照してください。
12. バケットの使用率がしきい値を超えたときにバケットメトリクスが通知されるよう、アラームを設定します。詳細については、「Amazon Lightsail でのバケットメトリクスアラームの作成」を参照してください。
13. ストレージとネットワーク転送量が不足している場合は、バケットのストレージプランを変更します。詳細については、「Amazon Lightsail のバケットのプランの変更」を参照してください。
14. バケットを他のリソースに接続する方法について説明します。詳細については、以下のチュートリアルを参照してください。
    - チュートリアル: WordPress インスタンスの Amazon Lightsail バケットへの接続
    - チュートリアル: Lightsail コンテンツ配信ネットワークディストリビューションでの Amazon Lightsail バケットの使用
15. 使用しなくなったバケットを削除します。詳細については、「Amazon Lightsail でのバケットの削除」を参照してください。

Amazon Lightsail のバケットオブジェクトを削除する

Amazon Lightsail オブジェクトストレージサービス内のバケットからオブジェクトを削除することができます。ストレージ領域を解放するには、不要になったオブジェクトを削除します。たとえば、ログファイルを収集している場合は、不要になったファイルを削除することをお勧めします。

バケットについての詳細は、「オブジェクトストレージ」を参照してください。

目次

- AWS CLI を使用して複数のオブジェクトまたはオブジェクトバージョンを削除する

バージョニングが有効なバケットからオブジェクトを削除する

バージョニングがバケットで有効化されている場合、複数のバージョンのオブジェクトがバケット内に存在する可能性があります。Lightsail コンソール、AWS CLI、AWS API、または AWS SDKS を使用して、オブジェクトのどのバージョンも削除することができます。ただし、次のオプションを検討する必要があります。

Lightsail コンソールを使用して、オブジェクトおよびオブジェクトバージョンを削除する

Lightsail コンソールの [オブジェクト] タブのオブジェクトブラウザペインでオブジェクトの最新バージョンを削除すると、以前のバージョンのオブジェクトもすべて削除されます。オブジェクトの特定のバージョンを削除するには、バージョンの管理ペインから実行してください。バージョン管理ペインを使用してオブジェクトの現在のバージョンを削除すると、以前の最新のバージョンが現在のバージョンとして復元されます。詳細については、このガイドで後述する「Lightsail コンソールを使用してオブジェクトバージョンを削除する」を参照してください。

Lightsail API、AWS CLI、または AWS SDK を使用して、オブジェクトおよびオブジェクトバージョンを削除する

単一のオブジェクトとその保存されているすべてのバージョンを削除するには、削除リクエストでオブジェクトのキーのみを指定します。オブジェクトの特定のバージョンを削除するためには、オブジェクトのキー名とバージョン ID の両方を指定します。詳細については、このガイドで後述する「AWS CLI で単一のオブジェクトまたはオブジェクトバージョンを削除するには」を参照してください。

Lightsail コンソールを使用してオブジェクトを削除します。

Lightsail コンソールを使い、保存された以前のバージョンを含めオブジェクトを削除するには、次の手順に従います。Lightsail コンソールでは、一度に 1 つずつしかオブジェクトを削除できません。複数オブジェクトの一括削除には、AWS CLI を使用してください。詳細については、このガイドで後述する「AWS CLI を使用して複数のオブジェクトまたはオブジェクトバージョンを削除する」を参照してください。

1. Lightsail コンソールにサインインします。
2. Lightsail のホームページで [Storage] (ストレージ) タブを選択します。
3. オブジェクトを削除するバケットの名前を選択します。

4. [オブジェクト] タブのオブジェクトブラウザペインを使用して、削除するオブジェクトの場所を参照します。
5. 削除するオブジェクトの横にあるチェックマークを追加します。
6. オブジェクト情報ペインで、アクション (:) メニューを選択し、[削除] を選択します。
7. 表示される確認ペインで[はい、削除します]を選択し、オブジェクトを完全に削除することを確認します。

   フォルダ内の唯一のオブジェクトを削除すると、そのフォルダも削除されます。これは、フォルダがオブジェクトキー名の一部であり、バケット内の他のオブジェクトが同じオブジェクトプレフィックスを共有していない場合、オブジェクトを削除すると、先行するフォルダも削除されるために発生します。詳細については、「オブジェクトストレージバケットのキー名」を参照してください。

Lightsail コンソールを使用してオブジェクトバージョンを削除する。

オブジェクトの保存されたバージョンを削除するには、次の手順を実行します。これは、バージョニングが有効なバケットでのみ可能です。詳細については、「バケットでのオブジェクトのバージョニングの有効化と一時停止」を参照してください。

1. Lightsail コンソールにサインインします。
2. Lightsail のホームページで [Storage] (ストレージ) タブを選択します。
3. オブジェクトを削除するバケットの名前を選択します。
4. オブジェクトブラウザペインを使用して、削除するオブジェクトの場所を参照します。
5. 削除するオブジェクトの保存された旧バージョンの横にチェックマークを追加します。
6. オブジェクト情報ペインのバージョンのセクションで[Manage] (管理) を選択します。
7. 保存されたオブジェクトのバージョンを管理するペインで、削除するオブジェクトのバージョンの横にチェックマークを追加します。

   オブジェクトの現在のバージョンを削除するように選択することもできます。
8. [選択済みを削除] をクリックして、選択したバージョンを削除します。

   削除した場合:

   - オブジェクトの現在のバージョン-オブジェクトの以前の最新のバージョンが現在のバージョンとして復元されます。

- オブジェクトの唯一のバージョン-オブジェクトがバケットから削除されます。削除したバージョンが現在のフォルダ内の唯一のオブジェクトである場合、フォルダも削除されます。これは、フォルダがオブジェクトキー名の一部であり、バケット内の他のオブジェクトが同じオブジェクトキープレフィックス共有していない場合、オブジェクトを削除すると、先行するフォルダも削除されるため発生します。詳細については、「バケットでのオブジェクトのバージョニングの有効化と一時停止」を参照してください。

AWS CLI を使用し単一のオブジェクトまたはオブジェクトバージョンを削除する

AWS Command Line Interface (AWS CLI) を使用して、バケット内の 1 つのオブジェクトまたはオブジェクトバージョンを削除するには、以下の手順を実行します。これは、`delete-object` コマンドを使用して実行できます。詳細については、「AWS CLI コマンドリファレンス」の「delete-object」を参照してください。

> **Note**
>
> この手順を続行する前に、AWS CLI をインストールし、Lightsail と Amazon S3 用に設定する必要があります。詳細については、「 Amazon Lightsail で使用するために AWS Command Line Interface を設定する」を参照してください。

1. ターミナルまたはコマンドプロントウィンドウを開きます。
2. バケット内のオブジェクトまたはオブジェクトバージョンを削除するには、次のコマンドを入力します。

   オブジェクトを削除するには:

   ```
   aws s3api delete-object --bucket BucketName --key ObjectKey
   ```

   オブジェクトバージョンを削除するには

   > **Note**
   >
   > オブジェクトバージョンの削除は、バージョニングが有効なバケットでのみ可能です。詳細については、「バケットでのオブジェクトのバージョニングの有効化と一時停止」を参照してください。

```
aws s3api delete-object --bucket BucketName --key ObjectKey --version-id VersionID
```

コマンドで、次のサンプルテキストを独自のテキストに置き換えます。

- *BucketName* - リストで、オブジェクトを削除するバケットの名前を選択します。
- *ObjectKey* - 削除するオブジェクトの完全なオブジェクトキー。
- *VersionId* - 削除するオブジェクトバージョンの ID。

例:

オブジェクトの削除：

```
aws s3api delete-object --bucket DOC-EXAMPLE-BUCKET --key images/sailbot.jpg
```

オブジェクトバージョンの削除：

```
aws s3api delete-object --bucket DOC-EXAMPLE-BUCKET --key images/sailbot.jpg --
version-id YF0YMBlUvexampleO07l2vJi9hRz4ujX
```

以下の例のような結果が表示されるはずです。

```
C:\Users\latino>aws s3api delete-object --bucket DOC-EXAMPLE-BUCKET  --key  images/sailbot.jpg  --version-id YF0YMBlUvexampleO07l2vJi9hRz4ujX
{
    "VersionId": "YF0YMBexampleY7PO07l2vJi9hRz4ujX"
}
```

AWS CLI を使用して複数のオブジェクトまたはオブジェクトバージョンを削除する

AWS Command Line Interface (AWS CLI) を使用してバケット内の複数のオブジェクトを削除するには、以下の手順を実行します。これは、delete-objects コマンドを使用して実行できます。詳細については、「AWS CLI コマンドリファレンス」の「delete-objects」を参照してください。

> **Note**
>
> この手順を続行する前に、AWS CLI をインストールし、Lightsail と Amazon S3 用に設定する必要があります。詳細については、「Amazon Lightsail で使用するために AWS Command Line Interface を設定する」を参照してください。

1. ターミナルまたはコマンドプロントウィンドウを開きます。
2. バケット内の複数のオブジェクトまたは複数のオブジェクトバージョンを削除するには、次のコマンドを入力します。

```
aws s3api delete-objects --bucket BucketName --delete file://LocalDirectory
```

以下のコマンド例を使用するには、以下のテキストを独自のものに置き換えてください。

- *BucketName* - 複数のオブジェクトまたは複数のオブジェクトバージョンを削除するバケットの名前。
- *LocalDirectory* - 削除するオブジェクトまたはバージョンを指定する .json ドキュメントのコンピュータ上のディレクトリパス。.json ドキュメントは以下のようにフォーマットできます。

  オブジェクトを削除するには、.json ファイルに次のテキストを入力し、*ObjectKey* を削除するオブジェクトのオブジェクトキーに置き換えます。

```
{
  "Objects": [
    {
      "Key": "ObjectKey1"
    },
    {
      "Key": "ObjectKey2"
    }
  ],
  "Quiet": false
}
```

  オブジェクトのバージョンを削除するには、.json ファイルに次のテキストを入力します。*ObjectKey* および *VersionID* を削除するオブジェクトバージョンのオブジェクトキーとオブジェクト ID に置き換えます。

> **Note**
>
> オブジェクトバージョンの削除は、バージョンが有効なバケットでのみ可能です。詳細については、「バケットでのオブジェクトのバージョニングの有効化と一時停止」を参照してください。

```
{
  "Objects": [
    {
      "Key": "ObjectKey1",
      "VersionId": "VersionID1"
    },
    {
      "Key": "ObjectKey2",
      "VersionId": "VersionID2"
    }
  ],
  "Quiet": false
}
```

例:

- Linux または Unix コンピュータの場合は、次の操作を行います。

```
aws s3api delete-objects --bucket DOC-EXAMPLE-BUCKET --delete file://home/user/Documents/delete-objects.json
```

- Windows コンピュータの場合:

```
aws s3api delete-objects --bucket DOC-EXAMPLE-BUCKET --delete file://C:\Users\user\Documents\delete-objects.json
```

以下の例のような結果が表示されるはずです。

```
C:\>aws s3api delete-objects --bucket DOC-EXAMPLE-BUCKET  --delete file://C:\Users\user\Documents\delete-objects.json
{
    "Deleted": [
        {
            "Key": "images/sailbot.jpg",
            "VersionId": "26sqexampleztRiT6TsGhMMz0FxQAEw."
        },
        {
            "Key": "images/sailbot.jpg",
            "VersionId": "QwDrexampleDJxJtZC1CrExbpNlEC5O4"
        }
    ]
}
```

バケットとオブジェクトを管理する

これらは、Lightsail オブジェクトストレージバケットを管理する一般的な手順です。

1. Amazon Lightsail オブジェクトストレージサービスでのオブジェクトとバケットについて説明します。詳細については、「Amazon Lightsail のオブジェクトストレージ」を参照してください。
2. Amazon Lightsail でバケットに付けることができる名前について説明します。詳細については、「Amazon Lightsail でのバケットの命名規則」をご参照ください。
3. バケットを作成して、Lightsail オブジェクトストレージサービスの使用を開始します。詳細については、「Amazon Lightsail におけるバケットの作成 」を参照してください。
4. バケットのセキュリティのベストプラクティスと、バケットに設定できるアクセス許可について説明します。バケット内のすべてのオブジェクトをパブリックまたはプライベートにすることも、オブジェクトを個別に選択してパブリックにすることもできます。また、アクセスキーを作成し、インスタンスをバケットに追加し、他の AWS アカウントにアクセス権を付与することで、バケットへのアクセスを許可することもできます。詳細については、「Amazon Lightsail オブジェクトストレージのセキュリティベストプラクティス」と「Amazon Lightsail でのバケットのアクセス許可を理解する」を参照してください。

   バケットのアクセス許可について理解したら、以下のガイドを参照してバケットへのアクセスを許可してください。

   - Amazon Lightsail でのバケットへのパブリックアクセスをブロックする
   - Amazon Lightsail でのバケットのアクセス許可の設定
   - Amazon Lightsail でのバケット内の個々のオブジェクトに対するアクセス許可の設定
   - Amazon Lightsail でのバケットのアクセスキーの作成
   - Amazon Lightsail でのバケットのリソースアクセスの設定
   - Amazon Lightsail でのバケットのクロスアカウントアクセスの設定
5. バケットのアクセスログの記録を有効にする方法と、アクセスログを使用してバケットのセキュリティを監査する方法について説明します。詳細については、以下のガイドを参照してください。

   - Amazon Lightsail オブジェクトストレージサービスのバケットへのアクセスのログ記録
   - Amazon Lightsail オブジェクトストレージサービスのバケットのアクセスログの形式
   - Amazon Lightsail オブジェクトストレージサービスのバケットへのアクセスのログ記録を有効にする
   - Amazon Lightsailでのバケットのアクセスログを使用するリクエストの特定

6. Lightsail でバケットを管理する機能をユーザーに付与する IAM ポリシーを作成します。詳細については、「Amazon Lightsail でバケットを管理する IAM ポリシー」を参照してください。
7. バケット内のオブジェクトにラベルを付けて識別する方法について説明します。詳細については、「Amazon Lightsail でのオブジェクトキー名を理解する」を参照してください。
8. ファイルをアップロードしてバケット内のオブジェクトを管理する方法について説明します。詳細については、以下のガイドを参照してください。
   - Amazon Lightsail のバケットにファイルをアップロードする
   - マルチパートアップロードを使用した Amazon Lightsail のバケットへのファイルのアップロード
   - Amazon Lightsail のバケット内のオブジェクトの表示
   - Amazon Lightsail のバケット内のオブジェクトのコピーまたは移動
   - Amazon Lightsail のバケットからのオブジェクトのダウンロード
   - Amazon Lightsail のバケット内のオブジェクトのフィルタリング
   - Amazon Lightsail のバケット内のオブジェクトのタグ付け
   - Amazon Lightsail のバケット内のオブジェクトの削除
9. オブジェクトのバージョニングを有効にすると、バケットに保存されたあらゆるオブジェクトのあらゆるバージョンを保存、取得、復元します。詳細については、「Amazon Lightsail のバケットでのオブジェクトのバージョニングの有効化と一時停止」を参照してください。
10. オブジェクトのバージョニングを有効にすると、バケット内のオブジェクトの以前のバージョンを復元できます。詳細については、「Amazon Lightsail のバケット内のオブジェクトの以前のバージョンの復元」を参照してください。
11. バケットの使用率を監視します。詳細については、「Amazon Lightsail でのバケットのメトリクスの表示」を参照してください。
12. バケットの使用率がしきい値を超えたときにバケットメトリクスが通知されるよう、アラームを設定します。詳細については、「Amazon Lightsail でのバケットメトリクスアラームの作成」を参照してください。
13. ストレージとネットワーク転送量が不足している場合は、バケットのストレージプランを変更します。詳細については、「Amazon Lightsail のバケットのプランの変更」を参照してください。
14. バケットを他のリソースに接続する方法について説明します。詳細については、以下のチュートリアルを参照してください。
   - チュートリアル: WordPress インスタンスの Amazon Lightsail バケットへの接続
   - チュートリアル: Lightsail コンテンツ配信ネットワークディストリビューションでの Amazon Lightsail バケットの使用

15.使用しなくなったバケットを削除します。詳細については、「Amazon Lightsail でのバケットの削除」を参照してください。

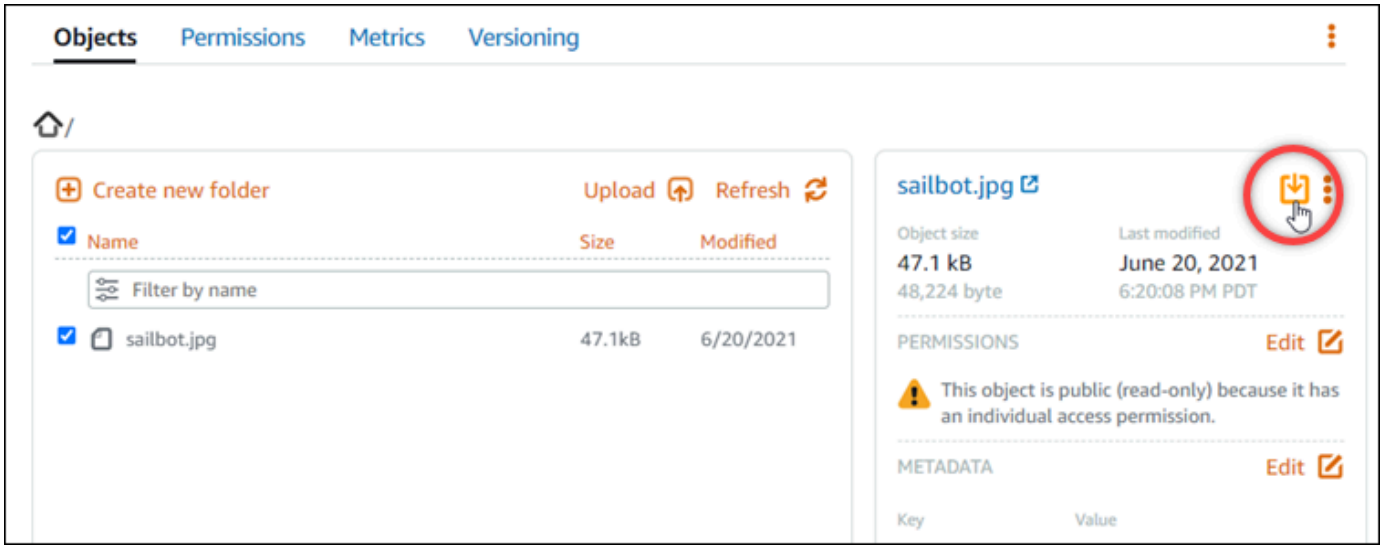
Amazon Lightsail のバケットからオブジェクトをダウンロードする

Amazon Lightsail オブジェクトストレージサービスでは、アクセス可能なバケットまたはパブリック（読み取り専用）のバケットからオブジェクトをダウンロードすることができます。Lightsail コンソールを使用すれば、オブジェクトを一つずつダウンロードできます。1 回のリクエストで複数のオブジェクトをダウンロードするには、AWS Command Line Interface (AWS CLI)、AWS SDK、または REST API を使用します。このガイドでは、Lightsail コンソールと AWS CLI を使用してオブジェクトをダウンロードする方法を紹介します。バケットについての詳細は、「オブジェクトストレージ」を参照してください。

Lightsail コンソールを使用してオブジェクトをダウンロードする

Lightsail コンソールを使用してバケットからオブジェクトをダウンロードするには、次の手順を実行します。

1. Lightsail コンソールにサインインします。
2. Lightsail のホームページで [Storage] (ストレージ) タブを選択します。
3. ファイルをダウンロードしたいバケットの名前を選択します。
4. オブジェクトタブのオブジェクトブラウザペインでダウンロードするオブジェクトの場所を参照します。
5. ダウンロードするオブジェクトの横にチェックマークを追加します。
6. 左オブジェクト情報ペインで、ダウンロードアイコンを選択します。



ブラウザの設定に応じて、選択したファイルはページに表示されるか、コンピュータにダウンロードされます。ファイルがページに表示されている場合は、ファイルを右クリックして、[Save as] を選択すると、コンピュータに保存されます。

AWS CLI を使用してオブジェクトをダウンロードするには

AWS Command Line Interface (AWS CLI) を使用してバケットからオブジェクトをダウンロードするには、次の手順を実行します。これは、`get-object` コマンドを使用して実行できます。詳細については、「AWS CLI コマンドリファレンス」の「get-object」を参照してください。

> **Note**
>
> この手順を続行する前に、AWS CLI をインストールし、Lightsail と Amazon S3 用に設定する必要があります。詳細については、「Amazon Lightsail で使用するために AWS Command Line Interface を設定する」を参照してください。

1. ターミナルまたはコマンドプロントウィンドウを開きます。
2. バケットからオブジェクトをダウンロードするには、次のコマンドを入力します。

```
aws s3api get-object --bucket BucketName --key ObjectKey LocalFilePath
```

コマンドで、次のサンプルテキストを独自のテキストに置き換えます。

- *BucketName* - オブジェクトのダウンロード元となるバケット名。
- *ObjectKey* - ダウンロードするオブジェクトの完全なオブジェクトキー。
- *LocalFilePath* - ダウンロードしたファイルを保存するコンピュータ上の完全なファイルパス。

例:

```
aws s3api get-object --bucket DOC-EXAMPLE-BUCKET --key images/sailbot.jpg C:\Users
\user\Pictures\sailbot.jpg
```

以下の例のような結果が表示されるはずです。

```
C:\>aws s3api get-object --bucket DOC-EXAMPLE-BUCKET --key images/sailbot.jpg C:\Users\user\Pictures\sailbot.jpg
{
    "AcceptRanges": "bytes",
    "LastModified": "2021-05-10T05:09:31+00:00",
    "ContentLength": 48224,
    "ETag": "\"694d34example91d92d64f342aa234c3\"",
    "ContentType": "binary/octet-stream",
    "ServerSideEncryption": "AES256",
    "Metadata": {}
}
```

バケットとオブジェクトを管理する

これらは、Lightsail オブジェクトストレージバケットを管理する一般的な手順です。

1. Amazon Lightsail オブジェクトストレージサービスでのオブジェクトとバケットについて説明します。詳細については、「Amazon Lightsail のオブジェクトストレージ」を参照してください。
2. Amazon Lightsail でバケットに付けることができる名前について説明します。詳細については、「Amazon Lightsail でのバケットの命名規則」をご参照ください。
3. バケットを作成して、Lightsail オブジェクトストレージサービスの使用を開始します。詳細については、「Amazon Lightsail におけるバケットの作成 」を参照してください。
4. バケットのセキュリティのベストプラクティスと、バケットに設定できるアクセス許可について説明します。バケット内のすべてのオブジェクトをパブリックまたはプライベートにすることも、オブジェクトを個別に選択してパブリックにすることもできます。また、アクセスキーを作成し、インスタンスをバケットに追加し、他の AWS アカウントにアクセス権を付与することで、バケットへのアクセスを許可することもできます。詳細については、「Amazon Lightsail オブジェクトストレージのセキュリティベストプラクティス」と「Amazon Lightsail でのバケットのアクセス許可を理解する」を参照してください。

   バケットのアクセス許可について理解したら、以下のガイドを参照してバケットへのアクセスを許可してください。

   - Amazon Lightsail でのバケットへのパブリックアクセスをブロックする
   - Amazon Lightsail でのバケットのアクセス許可の設定
   - Amazon Lightsail でのバケット内の個々のオブジェクトに対するアクセス許可の設定
   - Amazon Lightsail でのバケットのアクセスキーの作成
   - Amazon Lightsail でのバケットのリソースアクセスの設定
   - Amazon Lightsail でのバケットのクロスアカウントアクセスの設定

5. バケットのアクセスログの記録を有効にする方法と、アクセスログを使用してバケットのセキュリティを監査する方法について説明します。詳細については、以下のガイドを参照してください。
   - Amazon Lightsail オブジェクトストレージサービスのバケットへのアクセスのログ記録
   - Amazon Lightsail オブジェクトストレージサービスのバケットのアクセスログの形式
   - Amazon Lightsail オブジェクトストレージサービスのバケットへのアクセスのログ記録を有効にする
   - Amazon Lightsailでのバケットのアクセスログを使用するリクエストの特定
6. Lightsail でバケットを管理する機能をユーザーに付与する IAM ポリシーを作成します。詳細については、「Amazon Lightsail でバケットを管理する IAM ポリシー」を参照してください。
7. バケット内のオブジェクトにラベルを付けて識別する方法について説明します。詳細については、「Amazon Lightsail でのオブジェクトキー名を理解する」を参照してください。
8. ファイルをアップロードしてバケット内のオブジェクトを管理する方法について説明します。詳細については、以下のガイドを参照してください。
   - Amazon Lightsail のバケットにファイルをアップロードする
   - マルチパートアップロードを使用した Amazon Lightsail のバケットへのファイルのアップロード
   - Amazon Lightsail のバケット内のオブジェクトの表示
   - Amazon Lightsail のバケット内のオブジェクトのコピーまたは移動
   - Amazon Lightsail のバケットからのオブジェクトのダウンロード
   - Amazon Lightsail のバケット内のオブジェクトのフィルタリング
   - Amazon Lightsail のバケット内のオブジェクトのタグ付け
   - Amazon Lightsail のバケット内のオブジェクトの削除
9. オブジェクトのバージョニングを有効にすると、バケットに保存されたあらゆるオブジェクトのあらゆるバージョンを保存、取得、復元します。詳細については、「Amazon Lightsail のバケットでのオブジェクトのバージョニングの有効化と一時停止」を参照してください。
10. オブジェクトのバージョニングを有効にすると、バケット内のオブジェクトの以前のバージョンを復元できます。詳細については、「Amazon Lightsail のバケット内のオブジェクトの以前のバージョンの復元」を参照してください。
11. バケットの使用率を監視します。詳細については、「Amazon Lightsail でのバケットのメトリクスの表示」を参照してください。

12バケットの使用率がしきい値を超えたときにバケットメトリクスが通知されるよう、アラームを設定します。詳細については、「Amazon Lightsail でのバケットメトリクスアラームの作成」を参照してください。

13ストレージとネットワーク転送量が不足している場合は、バケットのストレージプランを変更します。詳細については、「Amazon Lightsail のバケットのプランの変更」を参照してください。

14バケットを他のリソースに接続する方法について説明します。詳細については、以下のチュートリアルを参照してください。

- チュートリアル: WordPress インスタンスの Amazon Lightsail バケットへの接続
- チュートリアル: Lightsail コンテンツ配信ネットワークディストリビューションでの Amazon Lightsail バケットの使用

15使用しなくなったバケットを削除します。詳細については、「Amazon Lightsail でのバケットの削除」を参照してください。

Amazon Lightsail のバケットオブジェクトをフィルタリングする

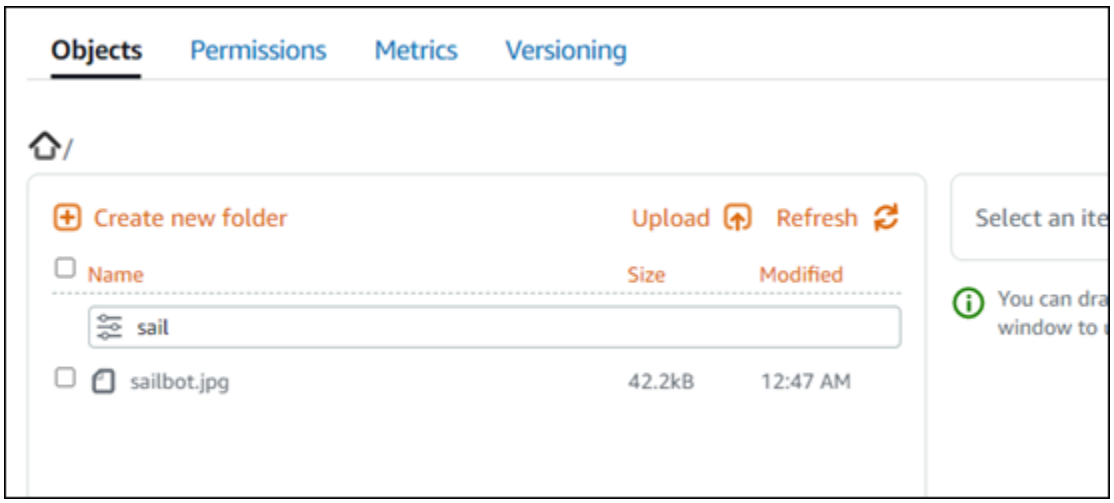
Amazon Lightsail オブジェクトストレージサービスでは、バケット内のオブジェクトをフィルタリングで検索できます。このガイドでは、Lightsail コンソールおよび AWS Command Line Interface (AWS CLI) を使用してオブジェクトをフィルタリングする方法を紹介します。バケットについての詳細は、「オブジェクトストレージ」を参照してください。

Lightsail コンソールを使用してオブジェクトをフィルターする

Lightsail コンソールでバケットのオブジェクトをフィルタリングするには、以下の手順を実行します。

1. Lightsail コンソールにサインインします。
2. Lightsail のホームページで [Storage] (ストレージ) タブを選択します。
3. オブジェクトを検索するバケットの名前を選択します。
4. オブジェクトタブで、オブジェクトの接頭辞を [名前でフィルタリングする] テキストボックスに入力します。

   現在表示中のフォルダのオブジェクトのリストは、入力したテキストに合わせてフィルタされます。次の例は、`sail` と入力した場合、ページ上のオブジェクトのリストが `sail` で始まるものに限定してフィルタリングされることを示しています。



別のフォルダでオブジェクトのリストをフィルタするには、そのフォルダへ移動します。次に、オブジェクトの接頭辞を [名前でフィルタリングする] テキストボックスに入力します。

AWS CLI を使用してオブジェクトをフィルタリングする

AWS Command Line Interface (AWS CLI) を使用してバケットのオブジェクトをフィルタリングするには、以下の手順を実行します。これは、`list-objects-v2` コマンドを使用して実行できます。詳細については、「AWS CLI コマンドリファレンス」の「list-objects-v2」を参照してください。

> **Note**
>
> この手順を続行する前に、AWS CLI をインストールし、Lightsail と Amazon S3 用に設定する必要があります。詳細については、「Amazon Lightsail で使用するために AWS Command Line Interface を設定する」を参照してください。

1. ターミナルまたはコマンドプロントウィンドウを開きます。
2. 特定のオブジェクトキー名のプレフィックスで始まるオブジェクトをリストするには、次のコマンドを入力します。

```
aws s3api list-objects-v2 --bucket BucketName --prefix ObjectKeyNamePrefix --query
 "Contents[].{Key: Key, Size: Size}"
```

コマンドで、次のサンプルテキストを独自のテキストに置き換えます。

- *BucketName* - すべてのオブジェクトをリストするバケット名。

- *ObjectKeyNamePrefix* - 指定されたプレフィックスで始まるキーに応答を制限するオブジェクトキー名プレフィックス。

> **Note**
>
> このコマンドは、--query パラメーターを利用し、list-objects-v2 リクエストへの応答を各オブジェクトのキー値とサイズにフィルタリングします。

例:

```
aws s3api list-objects-v2 --bucket DOC-EXAMPLE-BUCKET --prefix archived/ --query
 "Contents[].{Key: Key, Size: Size}"
```

次の例のような結果が表示されます。

```
C:\>aws s3api list-objects-v2  --bucket DOC-EXAMPLE-BUCKET --prefix archived/ --query "Contents[].{Key: Key, Size: Size}"
[
    {
        "Key": "archived/",
        "Size": 0
    },
    {
        "Key": "archived/1_CMoFsPfso.jpg",
        "Size": 2561865
    },
    {
        "Key": "archived/3y1zF4hIPCg.jpg",
        "Size": 6404907
    },
    {
        "Key": "archived/5IHz5WhosQE.jpg",
        "Size": 2377975
    },
    {
        "Key": "archived/sailbot.jpg",
        "Size": 43246
    }
]
```

バケットとオブジェクトを管理する

これらは、Lightsail オブジェクトストレージバケットを管理する一般的な手順です。

1. Amazon Lightsail オブジェクトストレージサービスでのオブジェクトとバケットについて説明します。詳細については、「Amazon Lightsail のオブジェクトストレージ」を参照してください。
2. Amazon Lightsail でバケットに付けることができる名前について説明します。詳細については、「Amazon Lightsail でのバケットの命名規則」をご参照ください。

3. バケットを作成して、Lightsail オブジェクトストレージサービスの使用を開始します。詳細については、「Amazon Lightsail におけるバケットの作成 」を参照してください。
4. バケットのセキュリティのベストプラクティスと、バケットに設定できるアクセス許可について説明します。バケット内のすべてのオブジェクトをパブリックまたはプライベートにすることも、オブジェクトを個別に選択してパブリックにすることもできます。また、アクセスキーを作成し、インスタンスをバケットに追加し、他の AWS アカウントにアクセス権を付与することで、バケットへのアクセスを許可することもできます。詳細については、「Amazon Lightsail オブジェクトストレージのセキュリティベストプラクティス」と「Amazon Lightsail でのバケットのアクセス許可を理解する」を参照してください。

   バケットのアクセス許可について理解したら、以下のガイドを参照してバケットへのアクセスを許可してください。

   - Amazon Lightsail でのバケットへのパブリックアクセスをブロックする
   - Amazon Lightsail でのバケットのアクセス許可の設定
   - Amazon Lightsail でのバケット内の個々のオブジェクトに対するアクセス許可の設定
   - Amazon Lightsail でのバケットのアクセスキーの作成
   - Amazon Lightsail でのバケットのリソースアクセスの設定
   - Amazon Lightsail でのバケットのクロスアカウントアクセスの設定
5. バケットのアクセスログの記録を有効にする方法と、アクセスログを使用してバケットのセキュリティを監査する方法について説明します。詳細については、以下のガイドを参照してください。

   - Amazon Lightsail オブジェクトストレージサービスのバケットへのアクセスのログ記録
   - Amazon Lightsail オブジェクトストレージサービスのバケットのアクセスログの形式
   - Amazon Lightsail オブジェクトストレージサービスのバケットへのアクセスのログ記録を有効にする
   - Amazon Lightsailでのバケットのアクセスログを使用するリクエストの特定
6. Lightsail でバケットを管理する機能をユーザーに付与する IAM ポリシーを作成します。詳細については、「Amazon Lightsail でバケットを管理する IAM ポリシー」を参照してください。
7. バケット内のオブジェクトにラベルを付けて識別する方法について説明します。詳細については、「Amazon Lightsail でのオブジェクトキー名を理解する」を参照してください。
8. ファイルをアップロードしてバケット内のオブジェクトを管理する方法について説明します。詳細については、以下のガイドを参照してください。

   - Amazon Lightsail のバケットにファイルをアップロードする

- マルチパートアップロードを使用した Amazon Lightsail のバケットへのファイルのアップロード
- Amazon Lightsail のバケット内のオブジェクトの表示
- Amazon Lightsail のバケット内のオブジェクトのコピーまたは移動
- Amazon Lightsail のバケットからのオブジェクトのダウンロード
- Amazon Lightsail のバケット内のオブジェクトのフィルタリング
- Amazon Lightsail のバケット内のオブジェクトのタグ付け
- Amazon Lightsail のバケット内のオブジェクトの削除

9. オブジェクトのバージョニングを有効にすると、バケットに保存されたあらゆるオブジェクトのあらゆるバージョンを保存、取得、復元します。詳細については、「Amazon Lightsail のバケットでのオブジェクトのバージョニングの有効化と一時停止」を参照してください。
10. オブジェクトのバージョニングを有効にすると、バケット内のオブジェクトの以前のバージョンを復元できます。詳細については、「Amazon Lightsail のバケット内のオブジェクトの以前のバージョンの復元」を参照してください。
11. バケットの使用率を監視します。詳細については、「Amazon Lightsail でのバケットのメトリクスの表示」を参照してください。
12. バケットの使用率がしきい値を超えたときにバケットメトリクスが通知されるよう、アラームを設定します。詳細については、「Amazon Lightsail でのバケットメトリクスアラームの作成」を参照してください。
13. ストレージとネットワーク転送量が不足している場合は、バケットのストレージプランを変更します。詳細については、「Amazon Lightsail のバケットのプランの変更」を参照してください。
14. バケットを他のリソースに接続する方法について説明します。詳細については、以下のチュートリアルを参照してください。
    - チュートリアル: WordPress インスタンスの Amazon Lightsail バケットへの接続
    - チュートリアル: Lightsail コンテンツ配信ネットワークディストリビューションでの Amazon Lightsail バケットの使用
15. 使用しなくなったバケットを削除します。詳細については、「Amazon Lightsail でのバケットの削除」を参照してください。

Amazon Lightsail のオブジェクトのバージョニングを有効化および一時停止する

Amazon Lightsail オブジェクトストレージサービスでのバージョニングとは、同じバケット内でオブジェクトの複数のバリアントを保持する手段です。バージョニング機能を使用すると、バケットに保存されたすべてのオブジェクトのすべてのバージョンを、保存、取得、復元することができます。バージョニングを使用すれば、誤ったユーザーアクションやアプリケーション障害からより簡単に回復することができます。バケットのバージョニングが有効な場合に、Lightsail オブジェクトストレージサービスが同じオブジェクトに対する複数の書き込みリクエストを同時に受信すると、すべてのオブジェクトが保存されます。バージョニングは Lightsail オブジェクトストレージサービスのバケットではデフォルトで無効になっているため、明示的に有効にする必要があります。バケットについての詳細は、「オブジェクトストレージ」を参照してください。

> **Important**
>
> 「個々のオブジェクトを公開可能 (読み取り専用)」のアクセス権が設定されているバケットでバージョニングを有効または一時停止にすると、アクセス権は「すべてのオブジェクトはプライベートです」にリセットされます。個々のオブジェクトをパブリックにするオプションを引き続き使用する場合は、バケットのアクセス権限を手動で「個々のオブジェクトを公開可能 (読み取り専用)」に変更する必要があります。詳細については、「バケットのアクセス許可を設定する」を参照してください。

バージョンが無効化、有効化、一時停止されたバケット

Lightsail コンソールのバケットバージョニングには、3 つの状態があります。

- 無効 (API と SDK では `NeverEnabled`)
- 有効 (API と SDK では `Enabled`)
- 一時停止 (API と SDK では `Suspended`)

一度バケットでバージョニングを有効にすると、無効状態に戻すことはできません。ただし、バージョニングを一時停止することは可能です。バージョニングは、バケットレベルで有効化および停止します。

バージョニングの状態は、バケット内のすべてのオブジェクト (一部ではない) に適用されます。バケットでバージョニングを有効にすると、すべての新しいオブジェクトがバージョニングされ、一意

のバージョン ID が割り当てられます。バージョニングが有効されたときにバケット内にすでに存在したオブジェクトは、それ以降は常にバージョニングされます。将来のリクエストによってオブジェクトが修正された場合、固有のバージョン ID が割り当てられます。

バージョン ID

バケットのバージョニングを有効にすると、Lightsail オブジェクトストレージサービスは保存されてるオブジェクトに対して固有のバージョン ID を自動的に生成します。例えば、1 つのバケット内に、`photo.gif` (バージョン 111111) と `photo.gif` (バージョン 121212) のように、キーは同じだがバージョン ID が異なる 2 つのオブジェクトを保持することができます。

バージョン ID を編集することはできません。バージョン ID は、Unicode、UTF-8 エンコード、URL 対応の不透明型文字列で、長さは 1,024 バイト以下です。以下はバージョン ID の例です。

```
3sL4kqtJlcpXroDTDmJ+rmSpXd3dIbrHY+MTRCxf3vjVBH40Nr8X8gdRQBpUMLUo
```

Lightsail コンソールを使用してオブジェクトのバージョニングを有効化または一時停止する

Lightsail コンソールを使用してオブジェクトのバージョニングを有効または一時停止するには、次の手順を実行します。

1. Lightsail コンソールにサインインします。
2. Lightsail のホームページで [Storage] (ストレージ) タブを選択します。
3. バージョニングを有効または一時停止するバケットの名前を選択します。
4. [Versioning] (バージョニング) タブを選択します。
5. バケットの現在のバージョニング状態に応じて、次のいずれかのアクションを実行します。

   - バージョニングが現在停止されているか、有効になっていない場合は、 ページの [Object versioning] (オブジェクトのバージョニング) セクションにあるトグルを選択してバージョニングを有効にします。

- バージョニングが現在有効になっている場合は、 ページの [Object versioning] セクションにあるトグルを選択してバージョニングを一時停止にします。

AWS CLI を使用してオブジェクトのバージョニングを有効または一時停止にする

AWS Command Line Interface (AWS CLI) を使用してオブジェクトのバージョニングを有効化または一時停止するには、次の手順を実行します。これは、update-bucket コマンドを使用して実行できます。詳細については、「AWS CLI コマンドリファレンス」の「update-bucket」を参照してください。

> **Note**
>
> この手順を続行する前に、AWS CLI をインストールし、Lightsail と Amazon S3 用に設定する必要があります。詳細については、「 Lightsail で使用するために AWS CLI を設定する」を参照してください。

1. ターミナルまたはコマンドプロントウィンドウを開きます。
2. 次のコマンドを入力して、オブジェクトのバージョニングを有効または一時停止にします。

```
aws lightsail update-bucket --bucket-name BucketName --versioning VersioningState
```

コマンド内で、次のサンプルテキストを独自のテキストに置き換えます。

- *BucketName* - オブジェクトのバージョニングを有効にしたいバケットの名前。
- *VersioningState* - 以下のいずれかを指します。
  - Enabled - オブジェクトのバージョニングを有効にする。
  - Suspended - 有効になっているオブジェクトのバージョニングを一時停止する。

例:

```
aws lightsail update-bucket --bucket-name DOC-EXAMPLE-BUCKET --versioning Enabled
```

以下の例のような結果が表示されるはずです。

```
C:\>aws lightsail update-bucket --bucket-name DOC-EXAMPLE-BUCKET --versioning Enabled
{
    "bucket": {
        "resourceType": "Bucket",
        "accessRules": {
            "getObject": "private",
            "allowPublicOverrides": false
        },
        "arn": "arn:aws:lightsail:us-west-2:1example7491:Bucket/f067383e-ee41-4485-b934-example2e2fd",
        "bundleId": "small_1_0",
        "createdAt": "2021-06-29T08:12:39.163000-07:00",
        "url": "https://DOC-EXAMPLE-BUCKET.s3.us-west-2.amazonaws.com",
        "location": {
            "availabilityZone": "all",
            "regionName": "us-west-2"
        },
        "name": "DOC-EXAMPLE-BUCKET",
        "supportCode": "621291663362/DOC-EXAMPLE-BUCKET/small_1_0",
        "tags": [],
        "objectVersioning": "Enabled",
        "ableToUpdateBundle": true
    },
    "operations": [
        {
            "id": "0d53d290-f4b2-43f0-89d2-example43448",
            "resourceName": "DOC-EXAMPLE-BUCKET",
            "resourceType": "Bucket",
            "createdAt": "2021-06-29T08:29:56.241000-07:00",
            "location": {
                "availabilityZone": "all",
                "regionName": "us-west-2"
            },
            "isTerminal": true,
            "operationDetails": "6example3362/DOC-EXAMPLE-BUCKET/small_1_0",
            "operationType": "UpdateBucket",
            "status": "Succeeded",
            "statusChangedAt": "2021-06-29T08:29:56.241000-07:00",
            "errorCode": "",
            "errorDetails": ""
        }
    ]
}
```

バケットとオブジェクトを管理する

これらは、Lightsail オブジェクトストレージバケットを管理する一般的な手順です。

1. Amazon Lightsail オブジェクトストレージサービスでのオブジェクトとバケットについて説明します。詳細については、「Amazon Lightsail のオブジェクトストレージ」を参照してください。
2. Amazon Lightsail でバケットに付けることができる名前について説明します。詳細については、「Amazon Lightsail でのバケットの命名規則」をご参照ください。
3. バケットを作成して、Lightsail オブジェクトストレージサービスの使用を開始します。詳細については、「Amazon Lightsail におけるバケットの作成 」を参照してください。

4. バケットのセキュリティのベストプラクティスと、バケットに設定できるアクセス許可について説明します。バケット内のすべてのオブジェクトをパブリックまたはプライベートにすることも、オブジェクトを個別に選択してパブリックにすることもできます。また、アクセスキーを作成し、インスタンスをバケットに追加し、他の AWS アカウントにアクセス権を付与することで、バケットへのアクセスを許可することもできます。詳細については、「Amazon Lightsail オブジェクトストレージのセキュリティベストプラクティス」と「Amazon Lightsail でのバケットのアクセス許可を理解する」を参照してください。

   バケットのアクセス許可について理解したら、以下のガイドを参照してバケットへのアクセスを許可してください。

   - Amazon Lightsail でのバケットへのパブリックアクセスをブロックする
   - Amazon Lightsail でのバケットのアクセス許可の設定
   - Amazon Lightsail でのバケット内の個々のオブジェクトに対するアクセス許可の設定
   - Amazon Lightsail でのバケットのアクセスキーの作成
   - Amazon Lightsail でのバケットのリソースアクセスの設定
   - Amazon Lightsail でのバケットのクロスアカウントアクセスの設定
5. バケットのアクセスログの記録を有効にする方法と、アクセスログを使用してバケットのセキュリティを監査する方法について説明します。詳細については、以下のガイドを参照してください。

   - Amazon Lightsail オブジェクトストレージサービスのバケットへのアクセスのログ記録
   - Amazon Lightsail オブジェクトストレージサービスのバケットのアクセスログの形式
   - Amazon Lightsail オブジェクトストレージサービスのバケットへのアクセスのログ記録を有効にする
   - Amazon Lightsailでのバケットのアクセスログを使用するリクエストの特定
6. Lightsail でバケットを管理する機能をユーザーに付与する IAM ポリシーを作成します。詳細については、「Amazon Lightsail でバケットを管理する IAM ポリシー」を参照してください。
7. バケット内のオブジェクトにラベルを付けて識別する方法について説明します。詳細については、「Amazon Lightsail でのオブジェクトキー名を理解する」を参照してください。
8. ファイルをアップロードしてバケット内のオブジェクトを管理する方法について説明します。詳細については、以下のガイドを参照してください。

   - Amazon Lightsail のバケットにファイルをアップロードする
   - マルチパートアップロードを使用した Amazon Lightsail のバケットへのファイルのアップロード

- Amazon Lightsail のバケット内のオブジェクトの表示
- Amazon Lightsail のバケット内のオブジェクトのコピーまたは移動
- Amazon Lightsail のバケットからのオブジェクトのダウンロード
- Amazon Lightsail のバケット内のオブジェクトのフィルタリング
- Amazon Lightsail のバケット内のオブジェクトのタグ付け
- Amazon Lightsail のバケット内のオブジェクトの削除

9. オブジェクトのバージョニングを有効にすると、バケットに保存されたあらゆるオブジェクトのあらゆるバージョンを保存、取得、復元します。詳細については、「Amazon Lightsail のバケットでのオブジェクトのバージョニングの有効化と一時停止」を参照してください。
10. オブジェクトのバージョニングを有効にすると、バケット内のオブジェクトの以前のバージョンを復元できます。詳細については、「Amazon Lightsail のバケット内のオブジェクトの以前のバージョンの復元」を参照してください。
11. バケットの使用率を監視します。詳細については、「Amazon Lightsail でのバケットのメトリクスの表示」を参照してください。
12. バケットの使用率がしきい値を超えたときにバケットメトリクスが通知されるよう、アラームを設定します。詳細については、「Amazon Lightsail でのバケットメトリクスアラームの作成」を参照してください。
13. ストレージとネットワーク転送量が不足している場合は、バケットのストレージプランを変更します。詳細については、「Amazon Lightsail のバケットのプランの変更」を参照してください。
14. バケットを他のリソースに接続する方法について説明します。詳細については、以下のチュートリアルを参照してください。
    - チュートリアル: WordPress インスタンスの Amazon Lightsail バケットへの接続
    - チュートリアル: Lightsail コンテンツ配信ネットワークディストリビューションでの Amazon Lightsail バケットの使用
15. 使用しなくなったバケットを削除します。詳細については、「Amazon Lightsail でのバケットの削除」を参照してください。

Amazon Lightsail でバケットオブジェクトの以前のバージョンを復元する

Amazon Lightsail オブジェクトストレージサービスのバケットがバージョニング対応の場合、オブジェクトの前のバージョンを復元できます。オブジェクトの前のバージョンを復元して、意図せぬユーザーアクションやアプリケーションの障害から回復します。

Lightsail コンソールを使ってオブジェクトの前のバージョンを復元できます。AWS Command Line Interface (AWS CLI) や AWS SDKs を使用してオブジェクトの前のバージョンを復元することもできます。これを行うには、オブジェクトの特定のバージョンをバケットにコピーし、同じオブジェクトキー名を使用します。これにより、現在のバージョンが前のバージョンに置き換えられ、前のバージョンが現在のバージョンになります。バージョニングの詳細については、「バケット内のオブジェクトのバージョニングの有効化と一時停止」を参照してください。バケットについての詳細は、「オブジェクトストレージ」を参照してください。

Lightsail コンソールを使ってオブジェクトの前のバージョンを復元する

Lightsail コンソールを使用してオブジェクトのバージョニングを有効または一時停止するには、次の手順を実行します。

1. Lightsail コンソールにサインインします。
2. Lightsail のホームページで [Storage] (ストレージ) タブを選択します。
3. 前のバージョンを復元したいオブジェクトが入っているバケットの名前を選択します。
4. [オブジェクト] タブにある [オブジェクトブラウザ] ペインを使用して、オブジェクトの場所を参照します。
5. 前のバージョンを復元したいオブジェクトの横にチェックマークを追加します。
6. [オブジェクト情報] ペインの [バージョン] セクションにある [管理] を選択します。
7. [復元] を選択します。
8. 表示される [保存されたバージョン] ペインの [オブジェクトの復元] で、復元したいオブジェクトのバージョンを選択します。
9. [Continue] (続行) をクリックします。
10. 確認プロンプトが表示されたら、[はい、復元します] を選択して、オブジェクトのバージョンを復元します。復元しない場合は、[いいえ。キャンセルする] を選択します。

AWS CLI コンソールを使ってオブジェクトの前のバージョンを復元する

オブジェクト AWS Command Line Interface (AWS CLI) の前のバージョンを復元するには、次の手順を実行します。これは、`copy-object` コマンドを使用して行います。同じオブジェクトキーを使用して、オブジェクトの前のバージョンを同じバケットにコピーする必要があります。詳細については、「AWS CLI コマンドリファレンス」の「copy-object」を参照してください。

> **Note**
>
> この手順を続行する前に、AWS CLI をインストールし、Lightsail と Amazon S3 用に設定する必要があります。詳細については、「Amazon Lightsail で使用するために AWS Command Line Interface を設定する」を参照してください。

1. ターミナルまたはコマンドプロントウィンドウを開きます。
2. オブジェクトの前のバージョンを復元するには、次のコマンドを入力します。

```
aws s3api copy-object --copy-source "BucketName/ObjectKey?versionId=VersionId" --
key ObjectKey --bucket BucketName
```

コマンド内で、次のサンプルテキストを独自のテキストに置き換えます。

- *BucketName* - 前のバージョンを復元したいオブジェクトが入っているバケットの名前。同じバケット名を --copy-source および --bucket パラメータに指定する必要があります。
- *ObjectKey* - 復元するオブジェクトの名前。同じオブジェクトキーの名前を --copy-source および --key パラメータに指定する必要があります。
- *VersionId* - 現在のバージョンとして復元したい、前のオブジェクトのバージョンの ID。list-object-versions コマンドを使用して、バケット内のオブジェクトのバージョン ID 一覧を取得します。

例:

```
aws s3api copy-object --copy-source "DOC-EXAMPLE-BUCKET/sailbot.jpg?
versionId=GQWEexample87Mdl8Q_DKdVTiVMi_VyU" -key sailbot.jpg --bucket DOC-EXAMPLE-
BUCKET
```

以下の例のような結果が表示されるはずです。

```
C:\>aws s3api copy-object --copy-source "DOC-EXAMPLE-BUCKET/sailbot.jpg?versionId=GQWEexample87Mdl8Q_DKdVTiVMi_VyU"
--key sailbot.jpg --bucket DOC-EXAMPLE-BUCKET
{
    "CopySourceVersionId": "GQWEcouyrfexampleQ_DKdVTiVMi_VyU",
    "VersionId": "hjL8anKzI1xcXYyexampleDvvqMXSLoi",
    "ServerSideEncryption": "AES256",
    "CopyObjectResult": {
        "ETag": "\"dc5afd388fb3example20cda3fe41c54\"",
        "LastModified": "2021-05-16T06:45:35+00:00"
    }
}
```

バケットとオブジェクトを管理する

これらは、Lightsail オブジェクトストレージバケットを管理する一般的な手順です。

1. Amazon Lightsail オブジェクトストレージサービスでのオブジェクトとバケットについて説明します。詳細については、「Amazon Lightsail のオブジェクトストレージ」を参照してください。
2. Amazon Lightsail でバケットに付けることができる名前について説明します。詳細については、「Amazon Lightsail でのバケットの命名規則」をご参照ください。
3. バケットを作成して、Lightsail オブジェクトストレージサービスの使用を開始します。詳細については、「Amazon Lightsail におけるバケットの作成 」を参照してください。
4. バケットのセキュリティのベストプラクティスと、バケットに設定できるアクセス許可について説明します。バケット内のすべてのオブジェクトをパブリックまたはプライベートにすることも、オブジェクトを個別に選択してパブリックにすることもできます。また、アクセスキーを作成し、インスタンスをバケットに追加し、他の AWS アカウントにアクセス権を付与することで、バケットへのアクセスを許可することもできます。詳細については、「Amazon Lightsail オブジェクトストレージのセキュリティベストプラクティス」と「Amazon Lightsail でのバケットのアクセス許可を理解する」を参照してください。

   バケットのアクセス許可について理解したら、以下のガイドを参照してバケットへのアクセスを許可してください。

   - Amazon Lightsail でのバケットへのパブリックアクセスをブロックする
   - Amazon Lightsail でのバケットのアクセス許可の設定
   - Amazon Lightsail でのバケット内の個々のオブジェクトに対するアクセス許可の設定
   - Amazon Lightsail でのバケットのアクセスキーの作成
   - Amazon Lightsail でのバケットのリソースアクセスの設定
   - Amazon Lightsail でのバケットのクロスアカウントアクセスの設定
5. バケットのアクセスログの記録を有効にする方法と、アクセスログを使用してバケットのセキュリティを監査する方法について説明します。詳細については、以下のガイドを参照してください。

   - Amazon Lightsail オブジェクトストレージサービスのバケットへのアクセスのログ記録
   - Amazon Lightsail オブジェクトストレージサービスのバケットのアクセスログの形式
   - Amazon Lightsail オブジェクトストレージサービスのバケットへのアクセスのログ記録を有効にする
   - Amazon Lightsailでのバケットのアクセスログを使用するリクエストの特定

6. Lightsail でバケットを管理する機能をユーザーに付与する IAM ポリシーを作成します。詳細については、「Amazon Lightsail でバケットを管理する IAM ポリシー」を参照してください。
7. バケット内のオブジェクトにラベルを付けて識別する方法について説明します。詳細については、「Amazon Lightsail でのオブジェクトキー名を理解する」を参照してください。
8. ファイルをアップロードしてバケット内のオブジェクトを管理する方法について説明します。詳細については、以下のガイドを参照してください。
   - Amazon Lightsail のバケットにファイルをアップロードする
   - マルチパートアップロードを使用した Amazon Lightsail のバケットへのファイルのアップロード
   - Amazon Lightsail のバケット内のオブジェクトの表示
   - Amazon Lightsail のバケット内のオブジェクトのコピーまたは移動
   - Amazon Lightsail のバケットからのオブジェクトのダウンロード
   - Amazon Lightsail のバケット内のオブジェクトのフィルタリング
   - Amazon Lightsail のバケット内のオブジェクトのタグ付け
   - Amazon Lightsail のバケット内のオブジェクトの削除
9. オブジェクトのバージョニングを有効にすると、バケットに保存されたあらゆるオブジェクトのあらゆるバージョンを保存、取得、復元します。詳細については、「Amazon Lightsail のバケットでのオブジェクトのバージョニングの有効化と一時停止」を参照してください。
10. オブジェクトのバージョニングを有効にすると、バケット内のオブジェクトの以前のバージョンを復元できます。詳細については、「Amazon Lightsail のバケット内のオブジェクトの以前のバージョンの復元」を参照してください。
11. バケットの使用率を監視します。詳細については、「Amazon Lightsail でのバケットのメトリクスの表示」を参照してください。
12. バケットの使用率がしきい値を超えたときにバケットメトリクスが通知されるよう、アラームを設定します。詳細については、「Amazon Lightsail でのバケットメトリクスアラームの作成」を参照してください。
13. ストレージとネットワーク転送量が不足している場合は、バケットのストレージプランを変更します。詳細については、「Amazon Lightsail のバケットのプランの変更」を参照してください。
14. バケットを他のリソースに接続する方法について説明します。詳細については、以下のチュートリアルを参照してください。
   - チュートリアル: WordPress インスタンスの Amazon Lightsail バケットへの接続
   - チュートリアル: Lightsail コンテンツ配信ネットワークディストリビューションでの Amazon Lightsail バケットの使用

15.使用しなくなったバケットを削除します。詳細については、「Amazon Lightsail でのバケットの削除」を参照してください。

Amazon Lightsail のバケットオブジェクトをタグ付けする

バケット内のオブジェクトにタグ付けして、目的、所有者、環境、またはその他の基準で分類します。タグは、アップロード時またはアップロード後にオブジェクトに追加できます。バケットについての詳細は、「オブジェクトストレージ」を参照してください。

Lightsail コンソールを使用してオブジェクトのタグを追加および削除

以下の手順で、Lightsail コンソールを使ってバケット内のオブジェクトにタグを追加･削除します。

1. Lightsail コンソールにサインインします。
2. Lightsail のホームページで [Storage] (ストレージ) タブを選択します。
3. オブジェクトにタグ付けするバケットの名前を選択します。
4. [オブジェクト]タブの[オブジェクト ブラウザ] ペインを使って、オブジェクトの場所を参照します。
5. タグを追加または削除するオブジェクトの横に、チェックマークを付けます。
6. [オブジェクト情報]ペインの[オブジェクトタグ] セクションで、次のいずれかのオプションを選択します。

   - [追加] または[編集] (タグが追加済みの場合)。[キー] テキストボックスにキーを入力し、[値] テキストボックスに値を入力します。次に、保存 を選択して、タグを追加します。それ以外の場合は、[キャンセル] を選択します。
   - [編集] し、削除するキーバリュー タグの横にある [X] を選択します。タグの削除が完了したら [保存] を選択し、タグを削除しない場合は[キャンセル] を選択します。

AWS CLI を使用してオブジェクトのタグを追加および削除

AWS Command Line Interface (AWS CLI) を使用して、オブジェクトにタグを追加したり、オブジェクトからタグを削除したりするには、次の手順を実行します。これを行うには、`put-object-tagging` と`delete-object-tagging` コマンドを使用します。詳細については、「AWS CLI コマンドリファレンス」の「put-object-tagging」 および 「delete-object-tagging」 を参照してください。

> **Note**
>
> この手順を続行する前に、AWS CLI をインストールし、Lightsail と Amazon S3 用に設定する必要があります。詳細については、「Lightsail で使用するために AWS CLI を設定する」を参照してください。

1. ターミナルまたはコマンドプロントウィンドウを開きます。
2. 次のいずれかのコマンドを入力します。

   - オブジェクトにタグを追加するには

     ```
     aws s3api put-object-tagging --bucket BucketName --key ObjectKey --tagging
      "{\"TagSet\":[{ \"Key\": \"KeyTag\", \"Value\": \"ValueTag\" }]}"
     ```

     コマンドで、次のサンプルテキストを独自のテキストに置き換えます。

     - *BucketName* - タグ付けするオブジェクトを含むバケットの名前。
     - *ObjectKey* - タグ付けするオブジェクトの完全なオブジェクトキー。
     - *KeyTag* - タグのキー値。
     - *ValueTag* - タグの値。

   - オブジェクトにタグを追加するには

     ```
     aws s3api put-object-tagging --bucket BucketName --key ObjectKey --tagging
      "{\"TagSet\":[{ \"Key\": \"KeyTag1\", \"Value\": \"ValueTag1\" }, { \"Key\":
      \"KeyTag2\", \"Value\": \"ValueTag2\" }]}"
     ```

     コマンドで、次のサンプルテキストを独自のテキストに置き換えます。

     - *BucketName* - タグ付けするオブジェクトを含むバケットの名前。
     - *ObjectKey* - タグ付けするオブジェクトの完全なオブジェクトキー。
     - *KeyTag1* - 最初のタグのキーバリュー。
     - *ValueTag1* - 最初のタグの値。
     - *KeyTag2* - 2番目のタグのキーバリュー。
     - *ValueTag2* - 2番目のタグの値。

   - オブジェクトからタグを削除します。

```
aws s3api delete-object-tagging --bucket BucketName --key ObjectKey
```

コマンドで、次のサンプルテキストを独自のテキストに置き換えます。

- *BucketName* - すべてのタグを削除するオブジェクトが含まれているバケットの名前。。
- *ObjectKey* - タグ付けするオブジェクトの完全なオブジェクトキー。

例:

```
aws s3api delete-object --bucket DOC-EXAMPLE-BUCKET --key nptLmg6jqDo.jpg --tagging
 "{\"TagSet\":[{ \"Key\": \"Importance\", \"Value\": \"High\" }]}"
```

以下の例のような結果が表示されるはずです。

```
C:\>aws s3api put-object-tagging --bucket DOC-EXAMPLE-BUCKET  --key nptLmg6jqDo.jpg
--tagging "{\"TagSet\":[{ \"Key\": \"Importance\", \"Value\": \"High\" }]}"
{
    "VersionId": "9nL2d41NuZdhdk4HS3kZIwOxJeS1kCkm"
}
```

バケットとオブジェクトを管理する

これらは、Lightsail オブジェクトストレージバケットを管理する一般的な手順です。

1. Amazon Lightsail オブジェクトストレージサービスでのオブジェクトとバケットについて説明します。詳細については、「Amazon Lightsail のオブジェクトストレージ」を参照してください。
2. Amazon Lightsail でバケットに付けることができる名前について説明します。詳細については、「Amazon Lightsail でのバケットの命名規則」をご参照ください。
3. バケットを作成して、Lightsail オブジェクトストレージサービスの使用を開始します。詳細については、「Amazon Lightsail におけるバケットの作成 」を参照してください。
4. バケットのセキュリティのベストプラクティスと、バケットに設定できるアクセス許可について説明します。バケット内のすべてのオブジェクトをパブリックまたはプライベートにすることも、オブジェクトを個別に選択してパブリックにすることもできます。また、アクセスキーを作成し、インスタンスをバケットに追加し、他の AWS アカウントにアクセス権を付与することで、バケットへのアクセスを許可することもできます。詳細については、「Amazon Lightsail オ

ブジェクトストレージのセキュリティベストプラクティス」と「Amazon Lightsail でのバケットのアクセス許可を理解する」を参照してください。

バケットのアクセス許可について理解したら、以下のガイドを参照してバケットへのアクセスを許可してください。

- Amazon Lightsail でのバケットへのパブリックアクセスをブロックする
- Amazon Lightsail でのバケットのアクセス許可の設定
- Amazon Lightsail でのバケット内の個々のオブジェクトに対するアクセス許可の設定
- Amazon Lightsail でのバケットのアクセスキーの作成
- Amazon Lightsail でのバケットのリソースアクセスの設定
- Amazon Lightsail でのバケットのクロスアカウントアクセスの設定

5. バケットのアクセスログの記録を有効にする方法と、アクセスログを使用してバケットのセキュリティを監査する方法について説明します。詳細については、以下のガイドを参照してください。
   - Amazon Lightsail オブジェクトストレージサービスのバケットへのアクセスのログ記録
   - Amazon Lightsail オブジェクトストレージサービスのバケットのアクセスログの形式
   - Amazon Lightsail オブジェクトストレージサービスのバケットへのアクセスのログ記録を有効にする
   - Amazon Lightsailでのバケットのアクセスログを使用するリクエストの特定
6. Lightsail でバケットを管理する機能をユーザーに付与する IAM ポリシーを作成します。詳細については、「Amazon Lightsail でバケットを管理する IAM ポリシー」を参照してください。
7. バケット内のオブジェクトにラベルを付けて識別する方法について説明します。詳細については、「Amazon Lightsail でのオブジェクトキー名を理解する」を参照してください。
8. ファイルをアップロードしてバケット内のオブジェクトを管理する方法について説明します。詳細については、以下のガイドを参照してください。
   - Amazon Lightsail のバケットにファイルをアップロードする
   - マルチパートアップロードを使用した Amazon Lightsail のバケットへのファイルのアップロード
   - Amazon Lightsail のバケット内のオブジェクトの表示
   - Amazon Lightsail のバケット内のオブジェクトのコピーまたは移動
   - Amazon Lightsail のバケットからのオブジェクトのダウンロード
   - Amazon Lightsail のバケット内のオブジェクトのフィルタリング
   - Amazon Lightsail のバケット内のオブジェクトのタグ付け

- Amazon Lightsail のバケット内のオブジェクトの削除

9. オブジェクトのバージョニングを有効にすると、バケットに保存されたあらゆるオブジェクトのあらゆるバージョンを保存、取得、復元します。詳細については、「Amazon Lightsail のバケットでのオブジェクトのバージョニングの有効化と一時停止」を参照してください。
10. オブジェクトのバージョニングを有効にすると、バケット内のオブジェクトの以前のバージョンを復元できます。詳細については、「Amazon Lightsail のバケット内のオブジェクトの以前のバージョンの復元」を参照してください。
11. バケットの使用率を監視します。詳細については、「Amazon Lightsail でのバケットのメトリクスの表示」を参照してください。
12. バケットの使用率がしきい値を超えたときにバケットメトリクスが通知されるよう、アラームを設定します。詳細については、「Amazon Lightsail でのバケットメトリクスアラームの作成」を参照してください。
13. ストレージとネットワーク転送量が不足している場合は、バケットのストレージプランを変更します。詳細については、「Amazon Lightsail のバケットのプランの変更」を参照してください。
14. バケットを他のリソースに接続する方法について説明します。詳細については、以下のチュートリアルを参照してください。
    - チュートリアル: WordPress インスタンスの Amazon Lightsail バケットへの接続
    - チュートリアル: Lightsail コンテンツ配信ネットワークディストリビューションでの Amazon Lightsail バケットの使用
15. 使用しなくなったバケットを削除します。詳細については、「Amazon Lightsail でのバケットの削除」を参照してください。

Lightsail バケットのリソースアクセスを設定する

Amazon Lightsail インスタンスを Lightsail バケットに添付して、バケットとそのオブジェクトに全プログラムアクセスを許可します。インスタンスをバケットにアタッチする場合、アクセスキーなどの認証情報を管理する必要はありません。アタッチするインスタンスおよびバケットは同じ AWS リージョン に存在する必要があります。インスタンスを別のリージョンにあるバケットにアタッチすることはできません。

リソースアクセスは、バケットに直接ファイルをアップロードするようにインスタンスでソフトウェアまたはプラグインが設定されている場合に適しています。例えば、バケットにメディアファイルを保存するように WordPress インスタンスを設定する場合。詳細については、「チュートリアル: バケットを WordPress インスタンスに接続する」を参照してください。

許可のオプションの詳細については、「バケットのアクセス許可」を参照してください。セキュリティのベストプラクティスの詳細については、「オブジェクトストレージのセキュリティのベストプラクティス」を参照してください。バケットについての詳細は、「オブジェクトストレージ」を参照してください。

バケットのリソースアクセスの設定

バケットのリソースアクセスを設定するには、以下の手順を実行します。

1. Lightsail コンソールにサインインします。
2. Lightsail のホームページで [Storage] (ストレージ) タブを選択します。
3. リソースアクセスを設定するバケット名を選択します。
4. [Permissions] (許可) タブを選択します。

   リソースアクセスセクションには、バケットに現在添付されているインスタンスが表示されます。(存在する場合)
5. [Attach instance] (インスタンスをアタッチ) を選択してインスタンスをバケットにアタッチします。
6. [Select an instance] (インスタンスを選択) ドロップダウンメニューで、バケットにアタッチするインスタンスを選択します。

   > **Note**
   >
   > 実行中または停止状態のインスタンスのみをアタッチできます。さらに、同じ AWS リージョン にあるインスタンスのみをバケットとしてアタッチすることが可能です。
7. [Attach] (アタッチ) を選択してインスタンスをアタッチします。それ以外の場合は、[キャンセル] を選択します。

   インスタンスは添付された後、バケットとそのオブジェクトへの全アクセス許可があります。インスタンスでソフトウェアまたはプラグインを設定して、プログラムでバケット上のファイルにアクセスしたりアップロードをすることが可能です。例えば、バケットにメディアファイルを保存するように WordPress インスタンスを設定する場合。詳細については、「チュートリアル: バケットを WordPress インスタンスに接続する」を参照してください。

Lightsail バケットのプランを変更する

Amazon Lightsail オブジェクトストレージサービスでは、バケットのストレージプランによって、月額コスト、ストレージクォータ、およびデータ転送クォータが変わります。バケットのストレージプランは、毎月の AWS 請求サイクル中で 1 回のみ更新可能です。バケットのストレージプランを変更すると、ストレージおよびネットワーク転送クォータがリセットされます。ただし、以前のストレージプランを使用して発生したストレージ容量とデータ転送超過料金は対象外となります。

バケットのストレージプランがストレージまたはデータ転送クォータを一貫して上回っている場合、またはバケットの使用量が一貫してクォータを下回っている場合は、バケットのストレージプランを変更しましょう。バケットの使用量の変動が予測できない場合があるため、バケットのストレージプランは、短期的な月々のコスト削減策としてではなく、長期的な戦略としてのみ変更をすることをお勧めします。長期間バケットに十分なストレージ容量とデータ転送クォータを提供するストレージプランを選択しましょう。

バケットについての詳細は、「オブジェクトストレージ」を参照してください。

Lightsail コンソールを使用してバケットのストレージプランを変更する

Lightsail コンソールを使用してバケットのストレージプランを変更するには、以下の手順を実行します。

1. Lightsail コンソールにサインインします。
2. Lightsail のホームページで [Storage] (ストレージ) タブを選択します。
3. プランを変更するバケットの名前を選択します。
4. バケット管理ページで メトリクス タブを選択します。
5. ストレージプランを変更するを選択します。
6. 表示される確認プロンプトで、はい、変更しますを選択して、バケットのストレージプランの変更を続行します。それ以外の場合は、[キャンセル] を選択します。
7. 使用したいプランを選択し、プラン選択を選択します。
8. 表示される確認プロンプトで、はい、適用しますを選択してバケットの変更を適用するか、いいえ、戻るを選択して適用をしないようにします。

AWS CLI を使用してバケットのストレージプランを変更する

AWS Command Line Interface (AWS CLI) を使用してバケットのプランを変更するには、以下の手順を実行します。これを行うには、`update-bucket-bundle` コマンドを使用します。API において

「バケットストレージプラン」は「バケットバンドル」と表記されるのでご注意ください。詳細については、「AWS CLI コマンドリファレンス」の「update-bucket-bundle」を参照してください。

> **Note**
>
> この手順を続行する前に、AWS CLI をインストールし、Lightsail と Amazon S3 用に設定する必要があります。詳細については、「Lightsail で使用するために AWS CLI を設定する」を参照してください。

1. ターミナルまたはコマンドプロントウィンドウを開きます。
2. バケットのプランを変更するには、以下のコマンドを入力します。

```
aws lightsail update-bucket-bundle --bucket-name BucketName --bundle-id BundleID
```

コマンドで、以下のサンプルテキストを独自のテキストに置き換えます。

- *BucketName* - 変更したいストレージプランのバケットの名前
- *BundleId* - 適用したいバケットの新しいバケットバンドルの ID。get-bucket-bundles コマンドを使用して、使用可能なバケットバンドルとその ID を一覧表示します。詳細については、「AWS CLI コマンドリファレンス」の「get-bucket-bundles」を参照してください。

例:

```
aws lightsail update-bucket-bundle --bucket-name DOC-EXAMPLE-BUCKET --bundle-
id medium_1_0
```

以下の例のような結果が表示されるはずです。

```
C:\>aws lightsail update-bucket-bundle --bucket-name DOC-EXAMPLE-BUCKET --bundle-id medium_1_0

{
    "operations": [
        {
            "id": "8example-8176-48bd-b1da-exampleb8404",
            "resourceName": "DOC-EXAMPLE-BUCKET",
            "resourceType": "Bucket",
            "createdAt": "2021-06-30T12:05:57.362000-07:00",
            "location": {
                "availabilityZone": "all",
                "regionName": "us-west-2"
            },
            "isTerminal": true,
            "operationDetails": "62example362/DOC-EXAMPLE-BUCKET/medium_1_0",
            "operationType": "UpdateBucketBundle",
            "status": "Succeeded",
            "statusChangedAt": "2021-06-30T12:05:57.362000-07:00",
            "errorCode": "",
            "errorDetails": ""
        }
    ]
}
```

Lightsail バケットのアクセス許可を設定する

バケットのアクセス許可を使用して、バケット内のオブジェクトへのパブリック (認証されていない) 読み取り専用アクセスを制御します。バケットはプライベートまたはパブリック (読み取り専用) にすることができます。また、バケットをプライベートにしつつ、個々のオブジェクトをパブリック (読み取り専用) にするオプションもあります。

> **Important**
>
> バケットをパブリック (読み取り専用) にすると、バケット内のすべてのオブジェクトが、バケットの URL を介してインターネット上の誰でも読み取り可能になります。(例えば、`https://DOC-EXAMPLE-BUCKET.us-east-1.amazonaws.com/media/sailbot.jpg`) インターネット上の誰にもオブジェクトへのアクセスを許可したくない場合は、バケットをパブリック (読み取り専用) にしないでください。

許可のオプションの詳細については、「バケットのアクセス許可」を参照してください。セキュリティのベストプラクティスの詳細については、「オブジェクトストレージのセキュリティのベストプラクティス」を参照してください。バケットについての詳細は、「オブジェクトストレージ」を参照してください。

> **Important**
>
> Lightsail オブジェクトストレージリソースは、パブリックアクセスを許可または拒否するときに、Lightsail バケットのアクセス許可と Amazon S3 アカウントレベルのブロックパブリックアクセス設定の両方を考慮するようになります。詳細については、「バケットに対するブロックパブリックアクセス」を参照してください。

バケットのアクセス許可設定

バケットのアクセス許可を設定する手順は以下を実行します。

1. Lightsail コンソールにサインインします。
2. Lightsail のホームページで [Storage] (ストレージ) タブを選択します。
3. アクセス権限を設定するバケットの名前を選択します。
4. [Permissions (許可)] タブを選択します。

   ページの [Bucket access permissions] (バケットアクセス許可) セクションに、バケットの現在設定されているアクセス権限が表示されます。
5. [Change permission] (権限の変更) を選択して、バケットのアクセス権限を変更します。
6. 以下のオプションのいずれかを選択します。

   - すべてのオブジェクトはプライベートです – バケット内のすべてのオブジェクトは、ご自身またはアクセスを許可したユーザーのみが読み取ることができます。
   - 個々のオブジェクトを公開可能にする (読み取り専用) — バケット内のオブジェクトは、パブリックにする個々のオブジェクト (読み取り専用) を指定しない限り、ご自身またはアクセスを許可したユーザーのみが読み取ることができます。個々のオブジェクトのアクセス許可の詳細については「バケット内の個々のオブジェクトに対するアクセス許可の設定」を参照してください。

     [個々のオブジェクトを公開可能にする (読み取り専用)] オプションは、バケット内のオブジェクトの一部を公開する必要があり、その他全てをプライベートにする場合にお勧めします。例えば、一部の WordPress プラグインでは、バケットの個々のオブジェクトを公開する必要があります。詳細については、「チュートリアル: バケットを WordPress インスタンスに接続する」および「チュートリアル: バケットをコンテンツ配信ネットワークディストリビューションと使用する」を参照してください。

- すべてのオブジェクトを公開 (読み取り専用) — バケット内のすべてのオブジェクトは、インターネット上の誰でも読み取り可能です。

> **Important**
>
> バケットをパブリック (読み取り専用) にすると、バケット内のすべてのオブジェクトが、バケットの URL を介してインターネット上の誰でも読み取り可能になります。 (例えば、`https://DOC-EXAMPLE-BUCKET.us-east-1.amazonaws.com/media/sailbot.jpg`) インターネット上の誰にもオブジェクトへのアクセスを許可したくない場合は、バケットをパブリック (読み取り専用) にしないでください。

7. [保存] を選択して変更を保存します。それ以外の場合は、[キャンセル] を選択します。

   変更したバケットアクセス許可に応じて、以下の変更が適用されます。

   - すべてのオブジェクトはプライベート - バケット内のすべてのオブジェクトは、個々のオブジェクトのアクセス許可がパブリック (読み取り専用)に以前設定されていても、プライベートに設定されます。
   - 個々のオブジェクトを公開可能にする (読み取り専用) - 個々のオブジェクトのアクセス許可がパブリック (読み取り専用)に以前設定されていたのがパブリックに設定されます。オブジェクトに対して個々のオブジェクトにアクセス許可を設定することが可能になります。
   - オブジェクトを全て公開 (読み取り専用) - バケット内のすべてのオブジェクトは、個々のオブジェクトのアクセス許可がプライベートに以前設定されていても、パブリックに設定されます。

   個々のオブジェクトのアクセス許可の詳細については「バケット内の個々のオブジェクトに対するアクセス許可の設定」を参照してください。

Lightsail バケットのクロスアカウントアクセスを設定する

クロスアカウントアクセスを使用して、他の AWS アカウントとそのユーザーに対してバケット内のすべてのオブジェクトに読み取り専用アクセスを許可します。クロスアカウント アクセスは、他の AWS アカウントとオブジェクトを共有したい場合に最適です。他の AWS アカウントにクロスアカウントアクセスを許可すると、そのアカウントのユーザーは、バケットとオブジェクトの URL を通じてバケット内のオブジェクトに読み取り専用のアクセスが可能になります (例えば、`https://DOC-EXAMPLE-BUCKET.us-east-1.amazonaws.com/media/sailbot.jpg`) 。最大 10 個の AWS アカウントにバケットのアクセス権を与えることができます。

許可のオプションの詳細については、「バケットのアクセス許可」を参照してください。セキュリティのベストプラクティスの詳細については、「オブジェクトストレージのセキュリティのベストプラクティス」を参照してください。バケットについての詳細は、「オブジェクトストレージ」を参照してください。

バケットのクロスアカウントアクセスの設定

バケットのクロスアカウントアクセスを設定するには、以下の手順を実行します。

1. Lightsail コンソールにサインインします。
2. Lightsail のホームページで [Storage] (ストレージ) タブを選択します。
3. クロスアカウントアクセスを設定するバケット名を選択します。
4. [アクセス許可] タブを選択します。

   ページの [クロスアカウントアクセス] セクションに、バケットにアクセスするように現在設定されている AWS アカウント ID が表示されています (存在する場合)。
5. [クロスアカウントアクセスを追加する] を選択して、別の AWS アカウントにバケットへのアクセスを許可します。
6. アクセスを許可したい AWS アカウントの ID を [アカウント ID] テキストボックスに入力します。
7. 保存を選択してアクセスを許可します。それ以外の場合は、[キャンセル] を選択します。

   追加した AWS アカウント ID は、このページの [クロスアカウントアクセス] セクションにリストされます。AWS アカウントのクロスアカウントアクセスを削除するには、削除する AWS アカウント ID の隣にある、削除 (ゴミ箱) アイコンを選択します。

Lightsail で個々のバケットオブジェクトに対するアクセス許可を設定する

個々のオブジェクトアクセス許可を使用して、認証なしで公開されたバケット内の個々のオブジェクトの読み取り専用アクセスを制御します。バケットはプライベートまたはパブリック (読み取り専用) に設定することができます。

> **Important**
>
> 個々のオブジェクトアクセス許可は、バケットのアクセス許可が個々のオブジェクトを公開可能にする (読み取り専用) に設定されている場合のみ設定が可能です。バケット許可のオプ

ションの詳細については、「バケットのアクセス許可」を参照してください。バケットについての詳細は、「オブジェクトストレージ」を参照してください。

個々のオブジェクトへのアクセス許可の設定は、バケット内のオブジェクトの一部を公開し、その他全てをプライベートにする必要があるなどの、特殊な場合にのみ行うことをお勧めします。例えば、一部の WordPress プラグインでは、バケットの個々のオブジェクトを公開する必要があります。詳細については、「チュートリアル: バケットを WordPress インスタンスに接続する」および「チュートリアル: バケットをコンテンツ配信ネットワークディストリビューションと使用する」を参照してください。

許可のオプションの詳細については、「バケットのアクセス許可」を参照してください。セキュリティのベストプラクティスの詳細については、「オブジェクトストレージのセキュリティのベストプラクティス」を参照してください。バケットについての詳細は、「オブジェクトストレージ」を参照してください。

個々のオブジェクトのアクセス許可の設定

バケットの個々のオブジェクトのアクセス許可を設定するには、以下の手順を実行します。Lightsail でバケットを管理する権限をユーザーに付与する IAM ポリシーの例については、「バケットを管理する IAM ポリシー」を参照してください。

1. Lightsail コンソールにサインインします。
2. Lightsail のホームページで [Storage] (ストレージ) タブを選択します。
3. 個々のオブジェクトのアクセス許可を設定するバケット名を選択します。
4. [オブジェクト] タブを選択します。
5. アクセス許可を設定するオブジェクトの横にチェックマークを追加します。

   オブジェク情報ペインがオブジェクトの現在のアクセス許可の状態を表示します。
6. オブジェクト情報ペインの [アクセス許可] セクションで [編集] を選択して、オブジェクトのアクセス許可を変更します。

> **Note**
>
> 編集オプションが使用できない場合は、そのバケットのアクセス許可では、個々のオブジェクトアクセス許可を設定することができないことを意味します。個々のオブジェクトアクセス許可を設定するには、バケットアクセス許可を「個々のオブジェクトを公開

可能にする (読み取り専用) 」に設定する必要があります。詳細については、「バケットのアクセス許可を設定する」を参照してください。

7. [アクセス許可の選択] のドロップダウンメニューから以下のいずれかのオプションを選択します。

   - プライベート – オブジェクトはご自身とアクセスを許可したユーザーのみが読み取ることができます。
   - パブリック (読み取り専用) – オブジェクトは世界中の誰もが読み取ることができます。

8. [保存] を選択して変更を保存します。それ以外の場合は、[キャンセル] を選択します。

   バケットのバケットアクセス許可設定は、個々のオブジェクトのアクセス許可に以下の影響を与えます。

   - バケットのアクセス許可が「すべてのオブジェクトはプライベートです」に設定されている場合、個々のオブジェクトのアクセス許可がパブリック (読み取り専用)に以前設定されていても、バケット内のすべてのオブジェクトはプライベートに設定されます。ただし、以前設定されていた個々のオブジェクトのアクセス許可は保持されます。例えば、バケットのアクセス許可を「個々のオブジェクトを公開可能にする (読み取り専用) 」に戻すと、すべてのオブジェクトで個々のアクセス許可がパブリック (読み取り専用) 再びパブリックになります。
   - バケットのアクセス許可が「すべてのオブジェクトをパブリック (読み取り専用) にする」に設定されている場合、個々のオブジェクトのアクセス許可がプライベートに以前設定されていても、バケット内の全てのオブジェクトはパブリックに設定されます。

     バケットアクセス許可の詳細については、「バケットアクセス許可を設定する」を参照してください。

Lightsail マルチパートアップロードによるバケットへのファイルのアップロード

マルチパートアップロードを使用すると、単一のファイルをパートのセットとしてバケットにアップロードできます。各パートは、ファイルのデータの連続する部分です。これらのファイルパートは、任意の順序で個別にアップロードできます。いずれかのパートの送信が失敗すると、他のパートに影響を与えることなくそのパートを再送することができます。ファイルのすべてのパートがアップロードされると、Amazon S3 はこれらのパートをアセンブルし、Amazon Lightsail のバケットにオブジェクトを作成します。通常、オブジェクトサイズが 100 MB 以上の場合は、単一のオペレーショ

ンでオブジェクトをアップロードする代わりに、マルチパートアップロードを使用することを考慮してください。バケットについての詳細は、「オブジェクトストレージ」を参照してください。

マルチパートアップロードの使用には、次の利点があります。

- スループットの向上 – パートを並列にアップロードすることで、スループットを向上させることができます。
- ネットワークの問題からの素早い回復 – パートサイズは比較的小さいため、ネットワークエラーにより失敗したアップロードを再開する際の影響を最小限に抑えることができます。
- 時間をおいてアップロード - ファイルのパートを時間をおいてアップロードできます。マルチパートアップロードを開始してから、24 時間以内にマルチパートアップロードを完了します。
- ファイルの最終的なサイズが不明な状態でアップロードを開始 – ファイルの作成中でもアップロードを開始できます。

次の方法でマルチパートアップロードを使用することをお勧めします。

- 安定した高帯域幅ネットワーク経由でラージファイルをアップロードする場合は、複数スレッドのパフォーマンスのために並行してファイルパートをアップロードする「マルチパートアップロード」を使用すると、可能な帯域幅の使用を最大化します。
- むらがあるネットワークでアップロードを実行する場合は、マルチパートアップロードを使用して、アップロードの再開を回避することで、ネットワークエラーに対する弾力性を高めます。マルチパートアップロードを使用している場合、中断されたパートのみを対象にアップロードを再試行します。最初からやり直したり、ファイル全体を再度アップロードする必要はありません。

目次

- AWS CLI を使用したマルチパートアップロードを完了する
- AWS CLI を使用した、バケットのマルチパートアップロードのリスト表示
- AWS CLI を使用したマルチパートアップロードの停止

マルチパートアップロードのプロセス

マルチパートアップロードには、Amazon S3 アクションを使用して、Lightsail のバケットにファイルをアップロードする 3 つのステップがあります。

1. CreateMultipartUploadアクションを使って、マルチパートアップロードを開始します。
2. UploadPartアクションを使ってファイルパートをアップロードします。
3. CompleteMultipartUploadアクションを使ってマルチパートアップロードを実行します。

> **Note**
>
> AbortMultipartUploadアクションを使用して、マルチパートアップロードを開始した後に停止できます。

マルチパートアップロードのリクエストが完了すると、Amazon シンプルストレージサービスはアップロードされたパートからオブジェクトを構築します。その後、バケット内の他のオブジェクトにアクセスするのと同じ方法で、オブジェクトにアクセスできます。

進行中のすべてのマルチパートアップロードをリストしたり、特定のマルチパートアップロードにおいてアップロードが完了したパートのリスト表示を取得したりできます。このようなオペレーションのそれぞれについて、このセクションで説明します。

マルチパートアップロードの開始

マルチパートアップロードを開始するリクエストを送信すると、Amazon シンプルストレージサービスはアップロード ID を含むレスポンスを返します。これは、マルチパートアップロードの一意の識別子です。パートのアップロード、パートのリスト、アップロードの完了、アップロードの停止を行うときは常に、アップロード ID を指定する必要があります。アップロードされるオブジェクトを説明するメタデータを提供したい場合は、マルチパートアップロードを開始するリクエストでメタデータを指定する必要があります。

パートのアップロード

パートをアップロードするときは、アップロード ID に加えて、パート番号を指定する必要があります。1 ~ 10,000 の範囲で任意のパート番号を選択できます。パート番号によって、アップロードするオブジェクトに含まれるパートとその位置が一意に識別されます。選択するパート番号は、連続している必要はありません (例えば、1、5、14など)。以前にアップロードしたパートと同じパート番号を使って新しいパートをアップロードした場合、以前のパートは上書きされます。

パートをアップロードするたびに Amazon シンプルストレージサービスはレスポンスに ETag ヘッダーを返します。パートのアップロードごとに、パート番号と ETag 値を記録する必要があります。マルチパートアップロードを完了するためには、残りのリクエストにこれらの値を含める必要があります。

> **Note**
>
> マルチパートアップロードのすべてのアップロードされたパートは、バケットに保存されます。アップロードを完了するか、アップロードを停止するか、アップロードがタイムアウトするまで、バケットのストレージ容量を消費します。詳細については、このガイドで後述する「マルチパートアップロードの保持期間」を参照してください。

マルチパートアップロードの完了

マルチパートアップロードを完了すると、パート番号に基づいて昇順に連結されたオブジェクトが Amazon シンプルストレージサービスによって作成されます。マルチパートアップロードの開始リクエストにオブジェクトメタデータが提供されている場合、Amazon シンプルストレージサービスによってそのメタデータはオブジェクトに関連付けられます。完了リクエストが正常に処理されると、個々のパートはなくなります。

マルチパートアップロードの完了リクエストには、アップロード ID と、パート番号およびそれに対応する ETag 値の両方のリストが含まれている必要があります。Amazon シンプルストレージサービスからのレスポンスには、結合されるオブジェクトデータを一意に識別する ETag が含まれます。この ETag が、オブジェクトデータの MD5 ハッシュになるとは限りません。

マルチパートアップロードは停止することもできます。マルチパートアップロードを停止した後は、再度同じアップロード ID を使ってパートをアップロードすることはできません。キャンセルされたマルチパートアップロードの任意の部分のすべてのストレージが解放されます。パートのアップロードが進行しているときにマルチパートアップロードを停止した場合は、停止後もそのパートのアップロードは成功または失敗する可能性があります。すべてのパートによって使用されているストレージを全部解放するには、すべてのパートのアップロードが完了した後で初めてマルチパートアップロードを停止する必要があります。

マルチパートアップロードのリスト化

特定のマルチパートアップロードのパートや、進行中のすべてのマルチパートアップロードをリスト表示できます。パートのリストオペレーションでは、特定のマルチパートアップロードについて既にアップロードしたパートの情報が返されます。パートのリストリクエストを送信するたびに、指定したマルチパートアップロードのパート情報 (最大で 1,000 個のパート) が Amazon シンプルストレージサービスから返されます。マルチパートアップロードに 1,000 個を超えるパートが含まれる場合、すべてのパートを取得するにはパートのリストリクエストを追加で送信する必要があります。返されるパートのリストには、アップロード中のパートは含まれていないことに注意してください。マルチパートアップロードのリストオペレーションを使用すると、進行中のマルチパートアップロードのリストを取得できます。

進行中のマルチパートアップロードとは、開始されているものの、まだ完了または停止されていないアップロードを意味します。各リクエストに最大 1,000 個のマルチパートアップロードが返されます。進行中のマルチパートアップロードが 1,000 個を超える場合、残りのマルチパートアップロードを取得するには、リクエストを追加で送信する必要があります。返されるリストは確認の目的でのみ使用します。マルチパートアップロードの完了リクエストを送信するときに、リストの結果を使用しないでください。その代わりに、パートのアップロード時に指定したパート番号と、それに対応する、Amazon シンプルストレージサービスから返される ETag 値で構成される独自のリストを維持しておいてください。

マルチパートアップロードの同時オペレーション

分散開発環境においては、アプリケーションから同じオブジェクトに対して複数の更新が同時に開始されることもありえます。同じオブジェクトキーを使ってアプリケーションから複数のマルチパートアップロードが開始される可能性もあります。そのようなアップロードごとに、アプリケーションからパートのアップロードが行われ、アップロードの完了リクエストが Amazon シンプルストレージサービスに送信されて、オブジェクトが作成されます。バケットでバージョニングが有効になっている場合は、マルチパートアップロードを完了するたびに新しいバージョンが作成されます。バージョニングが有効になっていないバケットの場合は、マルチパートアップロードの開始から完了までの間に受信されたリクエストなど、他のリクエストが優先される可能性もあります。

> **Note**
>
> マルチパートアップロードを開始してから完了する前に受信したリクエストなど、他のリクエストが優先される可能性があります。たとえば、あるキーを使ってマルチパートアップロードを開始した後、マルチパートアップロードが完了しないうちに別のオペレーションによってそのキーが削除されることがあります。この場合、マルチパートアップロードの完了

レスポンスによって、オブジェクトを確認できなくても、オブジェクト作成の成功が示される可能性があります。

マルチパートアップロードの保持期間

マルチパートアップロードのすべてのアップロードパートは、バケットに保存されます。アップロードを完了するか、アップロードを停止するか、またはアップロードがタイムアウトするまで、バケットのストレージ容量を消費します。マルチパートアップロードはタイムアウトになり、マルチパートアップロードが作成されてから 24 時間後に削除されます。マルチパートアップロードを停止するか、タイムアウトすると、アップロードされたすべてのパートが削除され、バケットで使用するために使用したストレージ領域が解放されます。

Amazon シンプルストレージサービスのマルチパートアップロード制限

次の表は、マルチパートアップロードの主な仕様をまとめたものです。

- 最大オブジェクトサイズ：5 TB
- アップロードあたりの最大パート数：10,000
- パート番号：1 ~ 10,000（両端を含む）
- パートサイズ：5 MB（最小）- 5 GB（最大）マルチパートアップロードの最後のパートには、サイズの制限はありません。
- パートのリストリクエストで返されるパートの最大数：1,000
- マルチパートアップロードのリストリクエストで返されるマルチパートアップロードの最大数：1,000

アップロードするファイルを分割します。

Linux または Unix オペレーティングシステムで`split`コマンドを使用して、ファイルを複数のパートに分割し、バケットにアップロードします。Windows オペレーティングシステムでファイルを分割するために使用できる同様のフリーウェア アプリケーションがあります。ファイルを複数のパートに分割した後、本ガイドの「マルチパートアップロードの開始」セクションに進んでください。

AWS CLI を使用したマルチパートアップロードの開始

AWS Command Line Interface (AWS CLI) を使用してマルチパートアップロードを開始するには、以下の手順を実行してください。これは、`create-multipart-upload` コマンドを使用して行いま

す。詳細については、「AWS CLIコマンドリファレンス」の「create-multipart-upload」を参照してください。

> **Note**
>
> この手順を続行する前に、AWS CLI をインストールし、Lightsail と Amazon S3 用に設定する必要があります。詳細については、「Lightsail で使用するために AWS CLI を設定する」を参照してください。

1. ターミナルまたはコマンドプロントウィンドウを開きます。
2. 次のコマンドを入力して、バケットのマルチパートアップロードを作成します。

```
aws s3api create-multipart-upload --bucket BucketName --key ObjectKey --acl bucket-owner-full-control
```

コマンドで、次のサンプルテキストを独自のテキストに置き換えます。

- *BucketName*#- マルチパートアップロードを作成するバケットの名前。
- *ObjectKey*#- アップロードするファイルに使用するオブジェクトキー。

例:

```
aws s3api create-multipart-upload --bucket DOC-EXAMPLE-BUCKET --key sailbot.mp4 --acl bucket-owner-full-control
```

次の例のような結果が表示されます。レスポンスにはUploadIDが含まれており、以降のコマンドでパーツをアップロードしたり、このオブジェクトのマルチパートアップロードを完了させるために指定する必要があります。

```
C:\ >aws s3api create-multipart-upload --bucket DOC-EXAMPLE-BUCKET --key sailbot.mp4
{
    "AbortDate": "2021-05-20T00:00:00+00:00",
    "AbortRuleId": "ExpireMultiPart",
    "ServerSideEncryption": "AES256",
    "Bucket": "DOC-EXAMPLE-BUCKET",
    "Key": "sailbot.mp4",
    "UploadId": "R4QU.mO.exampleiHWiLOeNw7JtXX7OotRhTLsXXCzF21CZdYlfj5lfjtiMnpzVw2WPj.exampleBTmL_N_.42.DlHYOTsITFsX.tO3XOUTTAHiCxY5VR8jWRGdkVkUG"
}
```

マルチパートアップロード用のUploadID ができたら、このガイドの「AWS CLI を使用してパートをアップロードする」のセクションに進み、パートのアップロードを開始します。

AWS CLI を使用してパートをアップロードする

AWS Command Line Interface (AWS CLI)を使用して、マルチパートアップロードのパートをアップロードするには、以下の手順を実行してください。これは、upload-part コマンドを使用して行います。詳細については、「AWS CLI コマンドリファレンス」の「upload-part」を参照してください。

> **Note**
>
> この手順を続行する前に、AWS CLI をインストールし、Lightsail と Amazon S3 用に設定する必要があります。詳細については、「Lightsail で使用するために AWS CLI を設定する」を参照してください。

1. ターミナルまたはコマンドプロントウィンドウを開きます。
2. 以下のコマンドを入力して、パートをバケットにアップロードします。

```
aws s3api upload-part --bucket BucketName --key ObjectKey --part-number Number --
body FilePart --upload-id "UploadID"  --acl bucket-owner-full-control
```

コマンドで、次のサンプルテキストを独自のテキストに置き換えます。

- *BucketName*#- マルチパートアップロードを作成するバケットの名前。
- *ObjectKey*#- アップロードするファイルに使用するオブジェクトキー。
- *Number* - アップロードするパートのパート番号。パート番号によって、アップロードするオブジェクトに含まれるパートとその位置が一意に識別されます。アップロードするパートごとに、--part-number パラメータを段階的に増やしてください。そのためには、マルチパートアップロードの完了時に Amazon Simple Storage Service がオブジェクトをアセンブルする順序で番号を付けてください。
- *FilePart* - コンピュータからアップロードするパートファイル。
- *UploadID* - このガイドで前半で作成したマルチパートアップロードのアップロード IDです。

例:

```
aws s3api upload-part --bucket DOC-EXAMPLE-BUCKET --
key sailbot.mp4 --part-number 1 --body sailbot.mp4.001 --upload-id
```

```
"R4QU.mO.exampleiHWiLOeNw7JtXX7OotRhTLsXXCzF21CZdYlfj5lfjtiMnpzVw2WPj.exampleBTmL_N_.42.Dl
--acl bucket-owner-full-control
```

次の例のような結果が表示されます。アップロードするパートごとに、upload-part コマンドを繰り返します。パーツのアップロードリクエストの応答には、アップロードしたパートの ETag 値が含まれます。アップロードした各パーツの ETag 値を記録する。このガイドで後述するマルチパートアップロードを完了するには、すべての ETag 値が必要です。

```
C:\ >aws s3api upload-part --bucket DOC-EXAMPLE-BUCKET  --key sailbot.mp4  --part-number 1 --body sailbot.mp4.001
 --upload-id "R4QU.mO.exampleiHWiLOeNw7JtXX7OotRhTLsXXCzF21CZdYlfj5lfjtiMnpzVw2WPj.exampleBTmL_N_.42.DlHYOTsITFsX.tO3XOUTTAHiCxY5VR8jWRGdkVkUG"
{
    "ServerSideEncryption": "AES256",
    "ETag": "\"4example7530246113e837a860a38bbb\""
}
```

AWS CLI を使用したマルチパートアップロードのパートのリスト化

AWS Command Line Interface (AWS CLI)を使用して、マルチパートアップロードのパートをリストにするには、以下の手順を実行してください。これは、list-parts コマンドを使用して行います。詳細については、「AWS CLI コマンドリファレンス」の「list-parts」を参照してください。

マルチパートアップロードでアップロードされたすべてのパーツのETag 値を取得するには、この手順を実行します。これらの値は、このガイドの後半でマルチパートアップロードを完了するために必要となります。ただし、パートのアップロードの応答からすべてのETag 値を記録した場合は、この手順をスキップして、このガイドの「マルチパートアップロード .json ファイルの作成」セクションに進むことができます。

> **Note**
>
> この手順を続行する前に、AWS CLI をインストールし、Lightsail と Amazon S3 用に設定する必要があります。詳細については、「Lightsail で使用するために AWS CLI を設定する」を参照してください。

1. ターミナルまたはコマンドプロントウィンドウを開きます。
2. バケットのマルチパートアップロードのパートをリストにするには、次のコマンドを入力します。

```
aws s3api list-parts --bucket BucketName --key ObjectKey --upload-id "UploadID"
```

コマンドで、次のサンプルテキストを独自のテキストに置き換えます。

- *BucketName*#- マルチパートアップロードのパートをリストするバケットの名前。
- *ObjectKey*#- マルチパートアップロードのオブジェクトキー。
- *UploadID* - このガイドの前半で作成したマルチパートアップロードのアップロード ID です。

例:

```
aws s3api list-parts --bucket DOC-EXAMPLE-BUCKET --key sailbot.mp4 --upload-id
 "R4QU.mO.exampleiHWiLOeNw7JtXX7OotRhTLsXXCzF21CZdYlfj5lfjtiMnpzVw2WPj.exampleBTmL_N_.42.Dl
```

次の例のような結果が表示されます。レスポンスには、マルチパートアップロードでアップロードしたパーツのすべてのパート番号とETag 値がリスト表示されます。これらの値をクリップボードにコピーして、このガイドの「マルチパートアップロード .json の作成」セクションに進みます。

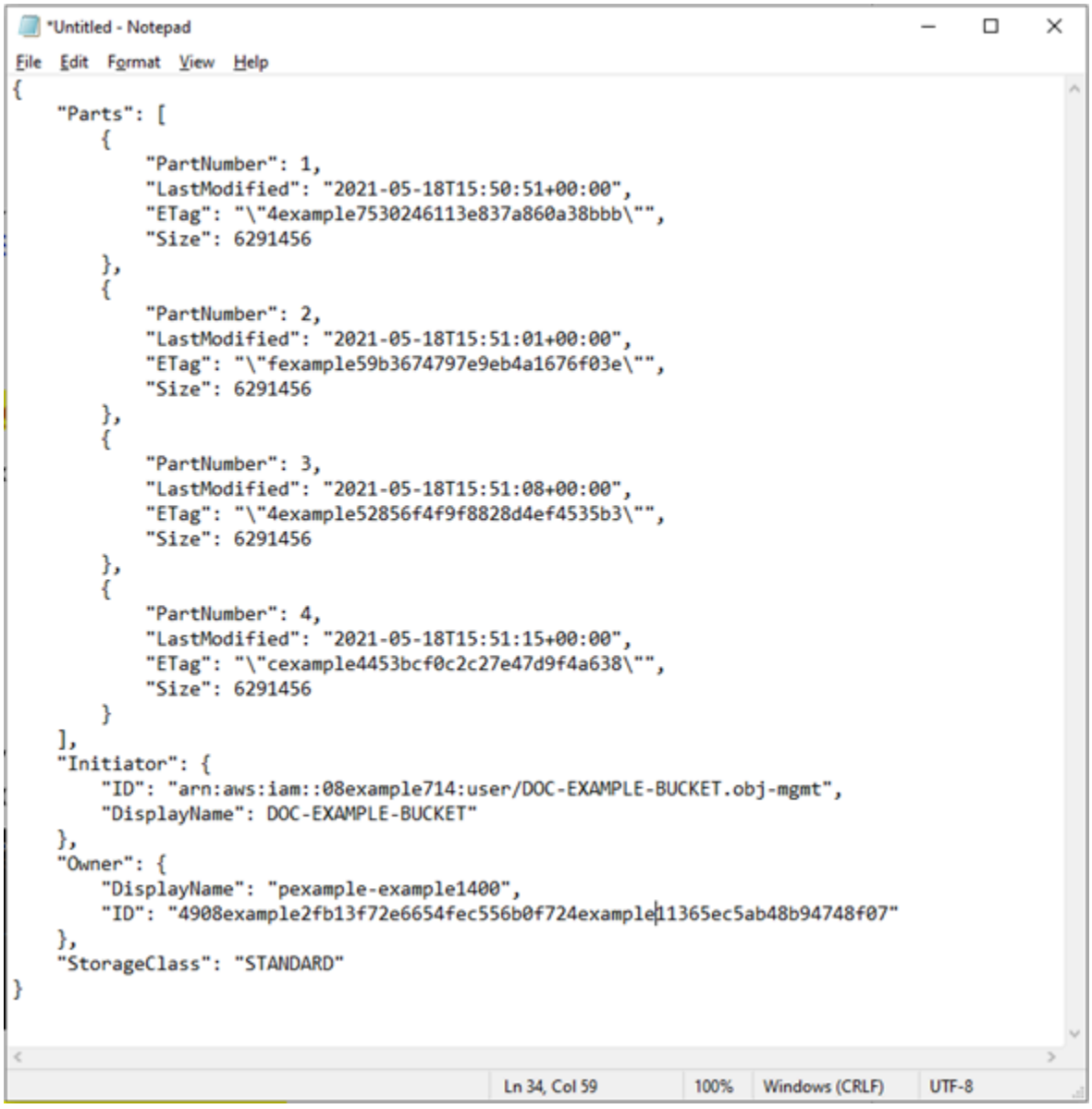
```
C:\>aws s3api list-parts --bucket DOC-EXAMPLE-BUCKET --key sailbot.mp4 --upload-id "R4QU.mO.exampleiHWiLOeNw7JtXX7OotR
hTLsXXCzF21CZdYlfj5lfjtiMnpzVw2WPj.exampleBTmL_N_.42.DlHYOTsITFsX.tO3XOUTTAHiCxY5VR8jWRGdkVkUG"
{
    "Parts": [
        {
            "PartNumber": 1,
            "LastModified": "2021-05-18T15:50:51+00:00",
            "ETag": "\"4example7530246113e837a860a38bbb\"",
            "Size": 6291456
        },
        {
            "PartNumber": 2,
            "LastModified": "2021-05-18T15:51:01+00:00",
            "ETag": "\"fexample59b3674797e9eb4a1676f03e\"",
            "Size": 6291456
        },
        {
            "PartNumber": 3,
            "LastModified": "2021-05-18T15:51:08+00:00",
            "ETag": "\"4example52856f4f9f8828d4ef4535b3\"",
            "Size": 6291456
        },
        {
            "PartNumber": 4,
            "LastModified": "2021-05-18T15:51:15+00:00",
            "ETag": "\"cexample4453bcf0c2c27e47d9f4a638\"",
            "Size": 6291456
        }
    ],
    "Initiator": {
        "ID": "arn:aws:iam::08example714:user/DOC-EXAMPLE-BUCKET.obj-mgmt",
        "DisplayName": "DOC-EXAMPLE-BUCKET"
    },
    "Owner": {
        "DisplayName": "pexample-example1400",
        "ID": "4908example2fb13f72e6654fec556b0f724example11365ec5ab48b94748f07"
    },
    "StorageClass": "STANDARD"
}
```

マルチパートアップロード .json ファイルの作成

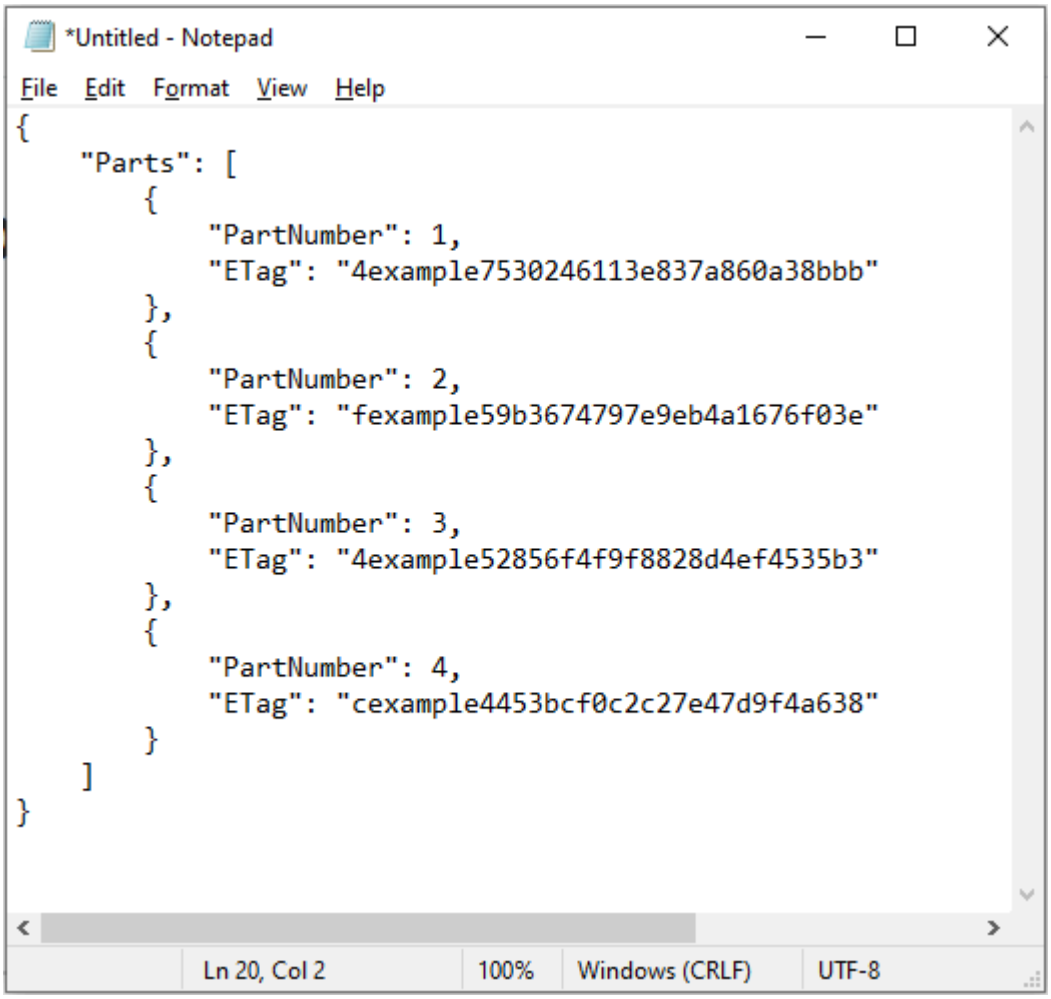
以下の手順で、アップロードしたすべてのパーツとそのETag 値を定義したマルチパートアップロード .json ファイルを作成します。これは、このガイドの後半で、マルチパートアップロードを完了するために必要です。

1. テキストエディターを開き、このガイドの前のセクションでリクエストしたlist-parts コマンドからのレスポンスを貼り付けます。

   結果は次の例のようになります。



2. 次の例に示すように、テキスト ファイルを再フォーマットします。



3. テキスト ファイルを、コンピュータに`mpstructure.json` のように保存し、本ガイドの「AWS CLI を使用したマルチパートアップロードを完了」に進みます。

AWS CLI を使用したマルチパートアップロードの完了

AWS Command Line Interface (AWS CLI) を使用してマルチパートアップロードを完了するには、以下の手順を実行してください。これは、`complete-multipart-upload` コマンドを使用して行います。詳細については、「AWS CLIコマンドリファレンス」の「complete-multipart-upload」を参照してください。

> **Note**
>
> この手順を続行する前に、AWS CLI をインストールし、Lightsail と Amazon S3 用に設定する必要があります。詳細については、「Lightsail で使用するために AWS CLI を設定する」を参照してください。

1. ターミナルまたはコマンドプロントウィンドウを開きます。
2. 以下のコマンドを入力して、パートをバケットにアップロードします。

```
aws s3api complete-multipart-upload --multipart-upload file://JSONFileName --
bucket BucketName --key ObjectKey --upload-id "UploadID"  --acl bucket-owner-full-
control
```

コマンドで、次のサンプルテキストを独自のテキストに置き換えます。

- *JSONFileName*#- このガイドの前半で作成した .json ファイルの名前 (例：mpstructure.json)
- *BucketName*#- マルチパートアップロードを完了するバケットの名前
- *ObjectKey*#- マルチパートアップロードのオブジェクトキー
- *UploadID* - このガイドの前半で作成したマルチパートアップロードのアップロード ID

Example:

```
aws s3api complete-multipart-upload --multipart-upload file://mpstructure.json
 --bucket DOC-EXAMPLE-BUCKET --key sailbot.mp4 --upload-id
 "R4QU.mO.exampleiHWiLOeNw7JtXX7OotRhTLsXXCzF21CZdYlfj5lfjtiMnpzVw2WPj.exampleBTmL_N_.42.Dl
 --acl bucket-owner-full-control
```

次の例に示すようなレスポンスが表示されます。これにより、マルチパートアップロードが完了したことを確認します。これで、オブジェクトがアセンブルされ、バケットで使用可能になります。

```
C:\>aws s3api complete-multipart-upload --multipart-upload file://mpstructure.json --bucket DOC-EXAMPLE-BUCKET  --key sailbot.mp4
--upload-id "R4QU.mO.exampleiHWiLOeNw7JtXX7OotRhTLsXXCzF21CZdYlfj5lfjtiMnpzVw2WPj.exampleBTmL_N_.42.DlHYOTsITFsX.tO3XOUTTAHiCxY5VR8jWRGdkVkUG"
{
    "ServerSideEncryption": "AES256",
    "VersionId": "MexampleKMdfPQb.2YZHqOvE_T.vSDtY",
    "Location": "https://DOC-EXAMPLE-BUCKET.s3.us-west-2.amazonaws.com/sailbot.mp4",
    "Bucket": "DOC-EXAMPLE-BUCKET",
    "Key": "sailbot.mp4",
    "ETag": "\"1example5964e3115e5d3f3c9a731585-4\""
}
```

AWS CLI を使用した、バケットのマルチパートアップロードのリスト化

AWS Command Line Interface (AWS CLI) を使用してバケットのマルチパートアップロードをリストにするには、以下の手順を実行します。これは、list-multipart-uploads コマンドを使用して

行います。詳細については、「AWS CLI コマンドリファレンス」の「list-multipart-uploads」を参照してください。

> **Note**
>
> この手順を続行する前に、AWS CLI をインストールし、Lightsail と Amazon S3 用に設定する必要があります。詳細については、「Lightsail で使用するために AWS CLI を設定する」を参照してください。

1. ターミナルまたはコマンドプロントウィンドウを開きます。
2. 以下のコマンドを入力して、パートをバケットにアップロードします。

```
aws s3api list-multipart-uploads --bucket BucketName
```

コマンドの中で、*BucketName*を、すべてのマルチパートアップロードをリストアップしたいバケットの名前に置き換えます。

例:

```
aws s3api list-multipart-uploads --bucket DOC-EXAMPLE-BUCKET
```

次の例に示すようなレスポンスが表示されます。

```
C:\>aws s3api list-multipart-uploads --bucket DOC-EXAMPLE-BUCKET
{
    "Uploads": [
        {
            "UploadId": "R4QU.mO.exampleiHWiLOeNw7JtXX7OotRhTLsXXCzF21CZdYlfj5lfjtiMnpzVw2WPj.exampleBTmL_N_.42.DlHYOTsITFsX.tO3XOUTTAHiCxY5VR8jWRGdkVkUG",
            "Key": "sailbot.mp4",
            "Initiated": "2021-05-18T15:49:11+00:00",
            "StorageClass": "STANDARD",
            "Owner": {
                "DisplayName": "pexample-example1400",
                "ID": "4908example2fb13f72e6654fec556b0f724example11365ec5ab48b94748f07"
            },
            "Initiator": {
                "ID": "arn:aws:iam::08example714:user/DOC-EXAMPLE-BUCKET.obj-mgmt",
                "DisplayName": "DOC-EXAMPLE-BUCKET"
            }
        }
    ]
}
```

AWS CLI を使用したマルチパートアップロードの停止

AWS Command Line Interface (AWS CLI) を使用してマルチパートアップロードを完了するには、以下の手順を実行します。マルチパートアップロード開始したものの、それを続行したくない場合に、

これを行います。これは、abort-multipart-upload コマンドを使用して行います。詳細については、「AWS CLI コマンドリファレンス」の「abort-multipart-upload」を参照してください。

> **Note**
>
> この手順を続行する前に、AWS CLI をインストールし、Lightsail と Amazon S3 用に設定する必要があります。詳細については、「Lightsail で使用するために AWS CLI を設定する」を参照してください。

1. ターミナルまたはコマンドプロントウィンドウを開きます。
2. 以下のコマンドを入力して、パートをバケットにアップロードします。

```
aws s3api abort-multipart-upload --bucket BucketName --key ObjectKey --upload-id
 "UploadID" --acl bucket-owner-full-control
```

コマンドで、次のサンプルテキストを独自のテキストに置き換えます。

- *BucketName*#- マルチパートアップロードを停止するバケットの名前。
- *ObjectKey*#- マルチパートアップロードのオブジェクトキー。
- *UploadID* - 停止するマルチパートアップロードのアップロード ID。

例:

```
aws s3api abort-multipart-upload --bucket DOC-EXAMPLE-BUCKET --key sailbot.mp4 --
upload-id
 "R4QU.mO.exampleiHWiLOeNw7JtXX7OotRhTLsXXCzF21CZdYlfj5lfjtiMnpzVw2WPj.exampleBTmL_N_.42.Dl
 --acl bucket-owner-full-control
```

このコマンドはレスポンスを返しません。以下のコマンドを実行するにはlist-multipart-uploadsコマンドを実行して、マルチパートアップロードが停止したことを確認します。

Amazon Lightsail でのバケットの名前付け

バケットを作成する場合、Amazon Lightsail オブジェクトストレージサービスで名前を指定する必要があります。バケットの名前は、バケットに保存されているオブジェクトにアクセスするときにカスタマーが使用する URL のパートです。たとえば、us-east-1 AWS リージョン のバケットに

DOC-EXAMPLE-BUCKET と名前を付けた場合、バケットの URL は DOC-EXAMPLE-BUCKET.s3.us-east-1.amazonaws.com となります。作成後にバケットの名前を変更することはできません。指定したバケット名をカスタマーが確認できることに留意してください。Lightsail オブジェクトストレージサービスの詳細については、「オブジェクトストレージ」を参照してください。バケットの作成の詳細については、「バケットの作成」を参照してください。

バケット名は、DNS に準拠している必要があります。このため、Lightsail でのバケットの命名には、次のルールが適用されます。

- バケット名は 3～56 文字の長さにする必要があります。
- バケット名は、小文字、数字、およびハイフン (-) のみで構成できます。
- バケット名は、文字または数字で開始および終了する必要があります。
- ハイフン (-) は単語を区切ることができますが、連続して指定することはできません。たとえば、doc-example-bucket は許可されますが、doc--example--bucket は許可されません。
- バケット名は、Amazon Simple Storage Service (Amazon S3) のバケットを含む、aws (スタンダード リージョン) パーティション内で固有でなくてはなりません。

バケット名の例

次の例に示すバケット名は有効であり、命名の推奨ガイドラインに従っています。

- docexamplebucket1
- log-delivery-march-2020
- my-hosted-content

次の例に示すバケット名は許可されません。

- doc.example.bucket
- doc--example--bucket
- doc-example-bucket-

Lightsail オブジェクトストレージバケットのキー名

バケットにアップロードしたファイルは、Amazon Lightsail オブジェクトストレージサービスにオブジェクトとして保存されます。オブジェクトキー (またはキー名) によって、バケットに保存されて

いるオブジェクトを一意に識別します。このガイドでは、Lightsail コンソールに表示されるバケットのフォルダー構造を構成するキー名とキー名プレフィックスの概念について説明します。バケットについての詳細は、「オブジェクトストレージ」を参照してください。

キー名

Lightsail オブジェクトストレージサービスのデータモデルは、ファイルシステムのような階層構造ではなく、フラット構造を使用しています。フォルダとサブフォルダの階層はありません。ただし、キー名のプレフィックスや区切り記号を使用して論理的な階層を暗示できます。Lightsail コンソールは、キー名のプレフィックスを使用してオブジェクトをフォルダー構造に表示します。

バケットに、次のようなオブジェクトキーを持つ 4 つのオブジェクトがあるとします。

- `Development/Projects.xls`
- `Finance/statement1.pdf`
- `Private/taxdocument.pdf`
- `to-dos.doc`

Lightsail コンソールは、キー名のプレフィックス (`Development/`、`Finance/`、および`Private/`) と区切り文字 (`/`) を使用してフォルダー構造を表します。`to-dos.doc` キーにはプレフィックスがないため、そのオブジェクトはバケットのルートレベルに直接表示されます。Lightsail `Development/` コンソールのフォルダーを参照すると、オブジェクトが表示されます。`Projects.xls`Finance/ フォルダに `statement1.pdf` オブジェクト、および `Private/` フォルダに `taxdocument.pdf` オブジェクトが表示されます。

Lightsail コンソールでは、キー名のプレフィックスとデリミター値をキー名として 0 バイトのオブジェクトを作成することでフォルダーを作成できます。これらのフォルダオブジェクトはコンソールに表示されません。ただし、他のオブジェクトと同様に動作します。Amazon S3 API、 AWS Command Line Interface (AWS CLI)、または AWS SDK を使用してそれらを表示および操作できます。

オブジェクトキーの命名のガイドライン

オブジェクトキー名には UTF-8 文字を使用できます。ただし、キー名に特定の文字を使用すると、一部のアプリケーションやプロトコルで問題が発生することがあります。以下のガイドラインは、DNS、ウェブセーフ文字、XML パーサー、その他の API とのコンプライアンスを最大化するのに役立ちます。

セーフ文字

以下の文字セットは、一般的にキー名で使用しても安全です。

- アルファベットの文字
  - 0-9
  - a ~ z
  - A ~ Z
- 特殊文字
  - スラッシュ (/)
  - 感嘆符 (!)
  - ハイフン (-)
  - 下線 (_)
  - ピリオド (.)
  - アスタリスク (*)
  - 一重引用符 (')
  - 丸かっこ開き (()
  - 丸かっこ閉じ ())

有効なオブジェクトキー名の例を次に示します。

- `4my-organization`
- `my.great_photos-2014/jan/myvacation.jpg`
- `videos/2014/birthday/video1.wmv`

> **Important**
>
> オブジェクトキー名が 1 つのピリオド (.) または 2 つのピリオド (..) で終わる場合、Lightsail コンソールを使用してオブジェクトをダウンロードすることはできません。キー名が 1 つまたは 2 つのピリオドで終わるオブジェクトをダウンロードするには、Amazon S3 API、AWS CLI、および AWS SDK を使用する必要があります。詳細については、「バケットオブジェクトをダウンロードする」を参照してください。

特殊な処理を必要とする可能性がある文字

キー名で以下の文字を使用すると、追加のコード処理が必要になる場合があり、16 進数として URL エンコードまたは参照することが必要になる可能性があります。これらの文字の一部は表示不可能な文字であり、ブラウザで処理されない場合があります。この場合も、特殊な処理が必要です。

- アンパサンド ("&")
- ドル記号 ("$")
- 16 進数の 00 ~ 1F (10 進数の 0 ~ 31) の範囲および 7F (10 進数の 127) の ASCII 文字
- アットマーク ("@")
- 等号 ("=")
- セミコロン (";")
- コロン (":")
- プラス記号 ("+")
- スペース – いくつかの用途 (特に複数のスペース) では、スペースの重要なシーケンスが失われる可能性があります。
- カンマ (",")
- 疑問符 ("?")

使用しない方がよい文字

すべてのアプリケーションで一貫性を維持するには相当な量の特殊な処理が必要になるため、キー名には以下の文字を使用しないでください。

- バックスラッシュ ("\")
- 左中括弧 ("{")
- 表示不可能な ASCII 文字 (10 進数の 128 ~ 255 の文字)
- カレット ("^")
- 右中括弧 ("}")
- パーセント記号 ("%")
- アクサングラーブ/バックティック ("`")
- 右角括弧 ("]")

- 引用符
- 大なり記号 (">")
- 左角括弧 ("[")
- チルダ ("~")
- 小なり記号 ("<")
- シャープ記号 ("#")
- 縦棒/パイプ ("|")

XML 関連のオブジェクトキーの制約

end-of-line 処理に関する XML 標準で規定されているように、すべての XML テキストは正規化され、1 つのキャリッジリターン (ASCII コード 13) とキャリッジリターンの直後に改行 (ASCII コード 10) が続くものは 1 つの改行文字に置き換えられます。XML リクエストでオブジェクトキーを正しく解析するには、キャリッジリターンやその他の特殊文字を XML タグ内に挿入するときに、同等の XML エンティティコードに置き換える必要があります。以下では、当該特殊文字とそれに相当するエンティティコードのリストを示しています。

- `'` としての '
- `"` としての ”
- `&` としての &
- `<` としての <
- `>` としての <
- `
` または `
` としての `\r`
- `
` または `
` としての `\n`

次の例では、キャリッジリターンの代わりに XML エンティティコードを使用する方法を示しています。この `DeleteObjects` リクエストにより、 キーパラメータ `/some/prefix/objectwith\rcarriagereturn` (\r はキャリッジリターン) を持つオブジェクトが削除されます。

```
<Delete xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
      <Object>
        <Key>/some/prefix/objectwith&#13;carriagereturn</Key>
      </Object>
    </Delete>
```

Lightsail におけるオブジェクトストレージのセキュリティのベストプラクティス

Amazon Lightsail オブジェクトストレージには、独自のセキュリティポリシーを開発および実装する際に考慮する必要のあるいくつかのセキュリティ機能が用意されています。以下のベストプラクティスは一般的なガイドラインであり、完全なセキュリティソリューションを説明するものではありません。これらのベストプラクティスは顧客の環境に必ずしも適切または十分でない可能性があるので、処方箋ではなく、あくまで有用な検討事項とお考えください。

目次

予防的セキュリティのベストプラクティス

以下のベストプラクティスは、Lightsail バケットでセキュリティ問題を防ぐのに役立ちます。

最小特権アクセスの実装

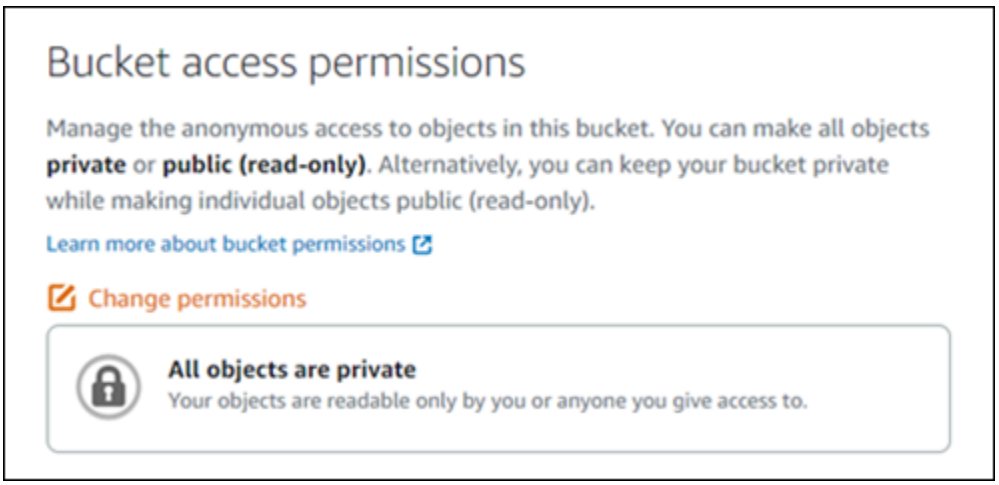
アクセス許可を付与する場合、どのユーザーにどの Lightsail リソースに対するアクセス許可を付与するかは、お客様が決定します。これらのリソースで許可したい特定のアクションを有効にするの

も、お客様になります。このため、タスクの実行に必要な許可のみを付与する必要があります。最小特権アクセスの実装は、セキュリティリスクと、エラーや悪意によってもたらされる可能性のある影響の低減における基本になります。

バケットを管理するための IAM ポリシーの作成の詳細については、「バケットを管理する IAM ポリシー」を参照してください。Lightsail バケットでサポートされている Amazon S3 アクションの詳細については、Amazon Lightsail API リファレンスの「オブジェクトストレージのアクション」を参照してください。

Lightsail バケットがパブリックにアクセス可能ではないことを確認する

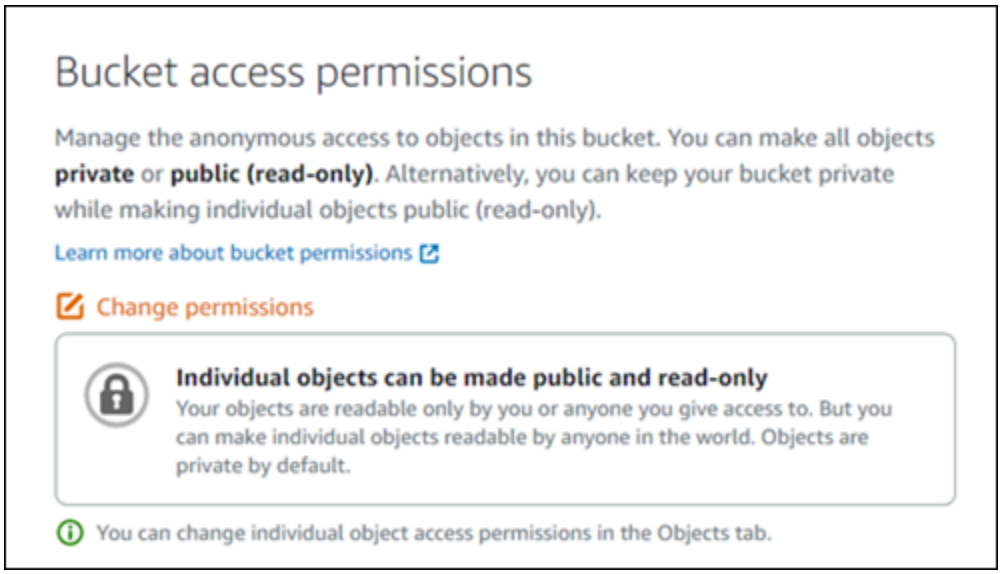
デフォルトでは、 バケットとオブジェクトはプライベートです。バケットのアクセス許可セットをすべてのオブジェクトはプライベートに設定して、バケットをプライベートに保ちます。大部分のユースケースでは、バケットや個々のオブジェクトをパブリックにする必要はありません。詳細については「バケット内の個々のオブジェクトに対するアクセス許可の設定」を参照してください。



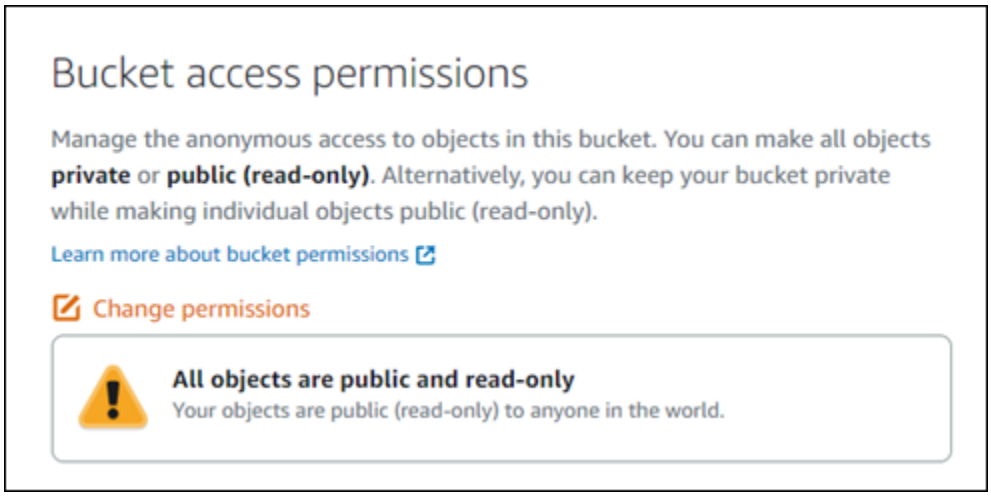
ただし、バケットを使用してウェブサイトやアプリケーションのメディアをホストしている場合は、特定のシナリオでは、バケットまたは個々のオブジェクトをパブリックにする必要があります。次のいずれかのオプションを設定して、バケットまたは個々のオブジェクトをパブリックにすることができます。

- バケット内のオブジェクトの一部のみをインターネット上の誰にでもパブリック (読み取り専用) する必要がある場合は、バケットのアクセス許可を個々のオブジェクトをパブリックにして読み取り専用にするに変更し、パブリックにする必要があるオブジェクトのみをパブリック (読み取り専用)に変更します。このオプションはバケットをプライベートにしますが、個々のオブジェクトをパブリックにするオプションも提供します。パブリックにアクセスしたくない機密情報または秘密情報が含まれている場合は、個々のオブジェクトを公開しないでください。個々のオブジェクトを

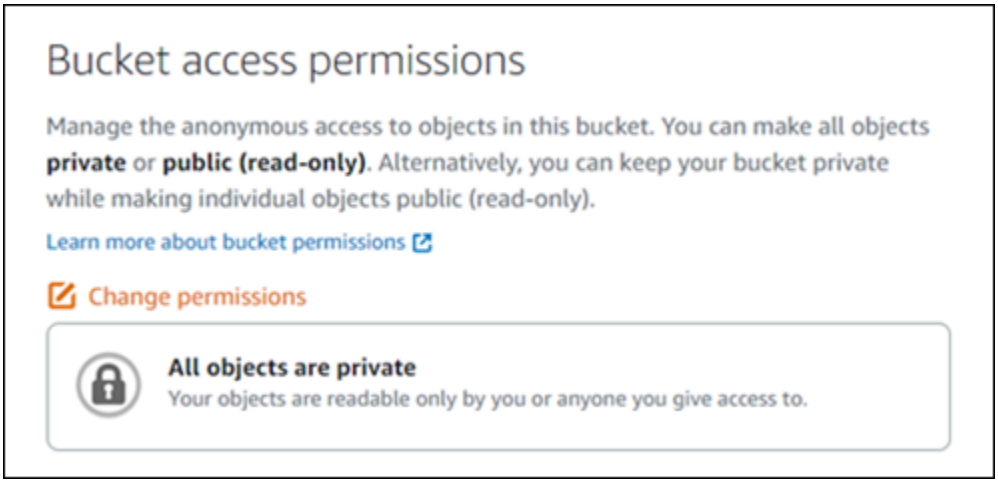
パブリックにする場合は、個々のオブジェクトのパブリックアクセシビリティを定期的に検証する必要があります。



- バケット内のすべてのオブジェクトをインターネット上の誰にでもパブリック（読み取り専用）する必要がある場合は、バケットのアクセス許可をすべてのオブジェクトはパブリックで読み取り専用に変更します。バケット内のいずれかのオブジェクトに機密情報または秘密情報が含まれている場合は、このオプションを使用しないでください。



- 以前にバケットをパブリックに変更した場合、または個々のオブジェクトをパブリックに変更した場合は、バケットのアクセス許可をすべてのオブジェクトはプライベートに変更することで、バケットとそのすべてのオブジェクトをプライベートにすばやく変更できます。

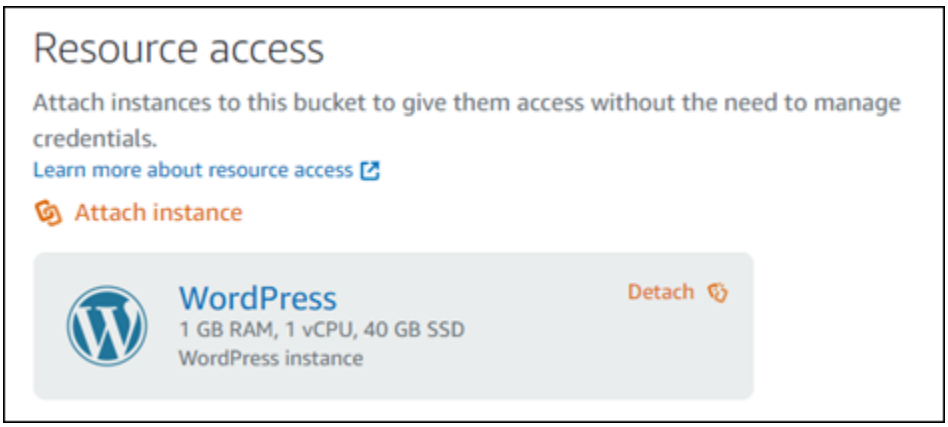


Amazon S3 でパブリックアクセスのブロックを有効にする

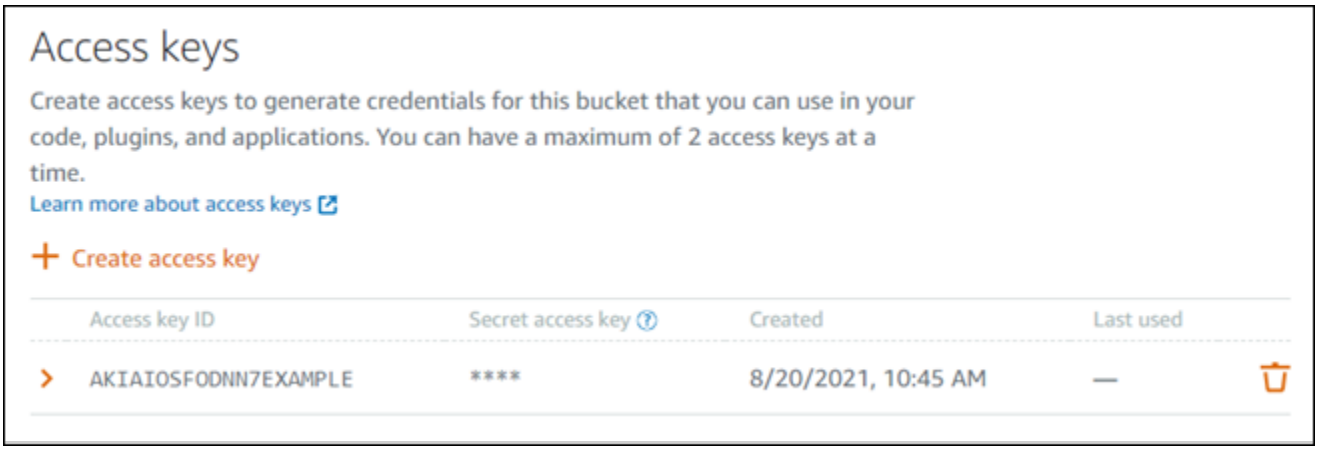
Lightsail オブジェクトストレージリソースは、パブリックアクセスを許可または拒否するときに、Lightsail バケットのアクセス許可と Amazon S3 アカウントレベルのブロックパブリックアクセス設定の両方を考慮するようになります。Amazon S3 アカウントレベルのブロックパブリックアクセスにより、アカウント管理者およびバケット所有者は、Amazon S3 および Lightsail バケットへのパブリックアクセスを一元的に制限できます。ブロックパブリックアクセスは、リソースがどのように作成され設定されているか、また設定されている可能性のある個々のバケットとオブジェクト権限にかかわらず、すべての Amazon S3 および Lightsail のバケットをプライベートにすることができます。詳細については、「バケットに対するブロックパブリックアクセス」を参照してください。

バケットにインスタンスをアタッチして、プログラムによる完全なアクセスを付与する

Lightsail オブジェクトストレージバケットにインスタンスを添付するのは、バケットへのアクセスを提供する最も安全な方法です。リソースアクセス機能は、インスタンスをバケットにアタッチする方法であり、インスタンスにバケットへの完全なプログラムによるアクセスを付与します。この方法では、バケット認証情報をインスタンスまたはアプリケーションに直接保存する必要はなく、定期的に認証情報をローテーションする必要もありません。例えば、一部の WordPress プラグインは、インスタンスがアクセスできるバケットにアクセスできます。詳細については、「バケットのリソースアクセスを設定する」および「チュートリアル: バケットを WordPress インスタンスに接続する」を参照してください。



ただし、アプリケーションが Lightsail インスタンスにない場合は、バケットアクセスキーを作成して設定することができます。バケットアクセスキーは、自動的にローテーションされない長期的な認証情報です。



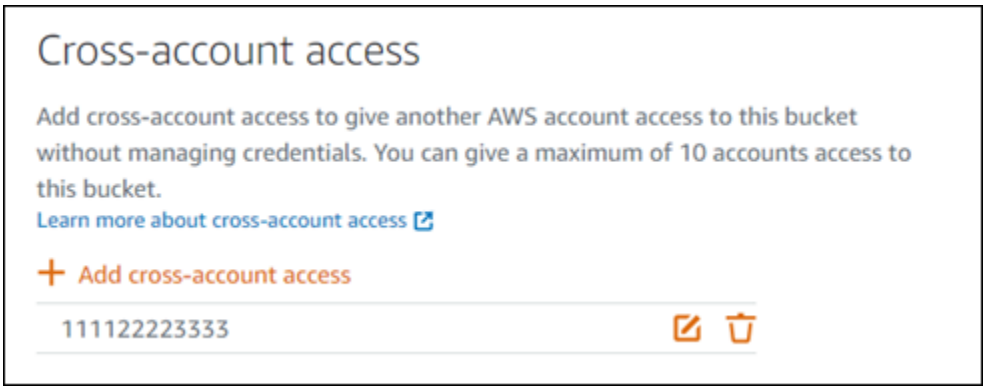
アクセスキーを作成して使用し、アプリケーションまたはプラグインにバケット内のオブジェクトへの完全なプログラムによるアクセスを付与できます。バケットでアクセスキーを使用する場合は、キーを定期的にローテーションし、既存のキーのインベントリを作成する必要があります。アクセスキーが最後に使用された日付と、そのキーが使用された AWS リージョン が、キーの使用方法に関する期待に沿っていることを確認します。アクセスキーが最後に使用された日付は、バケットの管理ページの [許可] タブのアクセスキーセクションの Lightsail コンソールに表示されます。使用されていないアクセスキーを削除します。

シークレットアクセスキーを誤ってパブリックと共有した場合は、削除して新しいシークレットアクセスキーを作成する必要があります。バケットごとに最大 2 つのアクセスキーを持つことができます。同時に 2 つの異なるアクセスキーを使用できますが、バケットで 1 つのアクセスキーを使用しないことは、最小限のダウンタイムでキーをローテーションする必要がある場合に役立ちます。アクセスキーをローテーションするには、新しいキーを作成し、ソフトウェアで設定してテストしてから、以前のキーを削除します。アクセスキーを削除すると、永久に削除されるため、再度取得するこ

とはできません。新しいアクセスキーでのみ置き換えることができます。詳細については、「バケットのアクセスキーの作成」を参照してください。

クロスアカウントアクセスを使用して、バケット内のオブジェクトへのアクセスを他の AWS アカウントに付与する

クロスアカウントアクセスを使用すると、バケットとそのオブジェクトをパブリックにすることなく、AWS アカウントを持つ特定の個人がバケット内のオブジェクトにアクセスできるようになります。クロスアカウントアクセスを設定している場合は、リストされているアカウント ID が、バケット内のオブジェクトへのアクセスを許可する正しいアカウントであることを確認してください。詳細については、「バケットのクロスアカウントアクセスの設定」を参照してください。



データの暗号化

Lightsail は、Amazon マネージドキーを使用したサーバー側の暗号化と HTTPS (TLS) の強制による転送中のデータの暗号化を実行します。サーバー側の暗号化は、別のサービスに保存されているキーを使用してデータを暗号化することで、データへのリスクを軽減するのに役立ちます。さらに、送信中のデータの暗号化は、潜在的な攻撃者が中間者攻撃または同様の攻撃を使用してネットワークトラフィックを盗聴または操作することを防止するのに役立ちます。

バージョニングの有効化

バージョニングとは、同じバケット内でオブジェクトの複数のバリアントを保持する手段です。バージョニングを使用して、Lightsail バケットに保存されたあらゆるオブジェクトのあらゆるバージョンを保存、取得、復元することができます。バージョニングを使用すれば、意図しないユーザーアクションからもアプリケーション障害からも、簡単に復旧できます。詳細については、「バケットのオブジェクトのバージョニングを有効化または一時停止する」を参照してください。

モニタリングと監査のベストプラクティス

以下のベストプラクティスは、Lightsail バケットの潜在的なセキュリティ上の弱点とインシデントを検出するのに役立ちます。

アクセスログ記録を有効にし、セキュリティとアクセス監査を定期的に実行する

アクセスのログ記録には、バケットに対するリクエストの詳細が記録されます。この情報には、リクエストタイプ (GET、PUT)、リクエストで指定したリソース、リクエストを処理した日時などが含まれます。バケットのアクセスログ記録を有効にし、セキュリティとアクセス監査を定期的に実行して、バケットにアクセスしているエンティティを特定します。デフォルトでは、Lightsail によってバケットへのアクセスのログは収集されません。アクセスログ記録を手動で有効にする必要があります。詳細については、「バケットのアクセスログ」と「バケットのアクセスログの記録を有効にする」を参照してください。

Lightsail バケットの特定、タグ付け、および監査

IT アセットの特定はガバナンスとセキュリティの重要な側面です。セキュリティ体制を評価し、潜在的な弱点に対処するには、すべての Lightsail バケットを可視化する必要があります。

タグ付けを使用してセキュリティまたは監査で注意を要するリソースを識別してから、それらのタグを、リソースを検索する必要があるときに使用します。詳細については、「タグ」を参照してください。

AWS モニタリングツールによるモニタリングの実装

モニタリングは、Lightsail バケットおよびその他のリソースの信頼性、セキュリティ、可用性、パフォーマンスを維持する上で重要です。バケットサイズ (`BucketSizeBytes`) と Lightsail での `Number of objects` (NumberOfObjects) バケットメトリクスの通知アラームを作成してモニタリングすることができます。例えば、バケットのサイズが特定のサイズに増減したとき、またはバケット内のオブジェクト数が特定の数に増減したときに通知を受け取ることができます。詳細については、「バケットメトリクスアラームの作成」を参照してください。

AWS CloudTrailを使用

AWS CloudTrail は、Lightsail のユーザー、ロール、または AWS のサービスによって実行されたアクションのレコードを提供します。CloudTrail で収集された情報を使用して、Lightsail に対するリクエスト、リクエスト元の IP アドレス、リクエストの実行者、リクエスト

日時などの詳細を把握できます。例えば、データアクセスに影響するアクション (特に CreateBucketAccessKey、GetBucketAccessKeys、DeleteBucketAccessKey、SetResourceAcc および UpdateBucket) の CloudTrail エントリを特定できます。AWS アカウントをセットアップすると、CloudTrail はデフォルトで有効になっています。CloudTrail コンソールで最近のイベントを確認できます。Lightsail バケットのアクティビティとイベントの継続的なレコードを作成するには、CloudTrail コンソールで追跡作成できます。詳細については、https://docs.aws.amazon.com/awscloudtrail/latest/userguide/logging-data-events-with-cloudtrail.html ユーザーガイドのAWS CloudTrail証跡へのデータイベントのログ記録 を参照してください。

AWS セキュリティアドバイザリを監視する

AWS アカウントに登録されているメインの E メールアドレスを注意してモニタリングしてください。AWS は、この E メールアドレスを使用して、お客様に影響を与える可能性のあるセキュリティ問題が新たに発生した場合に連絡します。

広範な影響を与える AWS の運用上の問題は AWS Service Health Dashboard に投稿されます。運用上の問題は Personal Health Dashboard を介して個々のアカウントにも投稿されます。詳細については、AWS の正常性に関するドキュメントを参照してください。

Amazon Lightsail でのバケットのアクセス許可

デフォルトでは、すべての Amazon Lightsail オブジェクトストレージリソース (バケットとオブジェクト) はプライベートです。これは、バケット所有者、つまりバケットを作成した Lightsail アカウントのみがバケットとそのオブジェクトにアクセスできることを意味します。バケット所有者は、オプションで他のユーザーにアクセス許可を付与できます。バケットとそのオブジェクトへのアクセスを許可するには、以下の方法があります。

- 読み取り専用アクセス — 以下のオプションは、バケットの URL (たとえば、https://DOC-EXAMPLE-BUCKET.us-east-1.amazonaws.com/media/sailbot.jpg) を介してバケットとそのオブジェクトへの読み取り専用アクセスを制御します。
  - バケットアクセス許可 — バケットのアクセス許可を使用して、インターネット上のすべてのオブジェクトへのアクセスを許可します。詳細については、このガイドで後述する「バケットのアクセス許可」を参照してください。
  - 個々のオブジェクトのアクセス許可 — 個々のオブジェクトアクセス許可を使用して、インターネット上のすべてのユーザーに、バケット内の個々のオブジェクトへのアクセスを許可します。詳細については、このガイドで後述する「個々のオブジェクトへのアクセス許可」を参照してください。

  - クロスアカウントアクセス — クロスアカウントアクセスを使用して、他の AWS アカウントのバケット内のすべてのオブジェクトへのアクセスを許可します。詳細については、このガイドで後述する「クロスアカウント アクセス」を参照してください。
- 読み取りおよび書き込みアクセス — バケットとそのオブジェクトへの完全な読み取りおよび書き込みアクセスを制御します。これらのオプションは AWS Command Line Interface (AWS CLI)、AWS API、AWS SDK とともに使用します。
  - アクセスキー — アクセスキーを使用して、アプリケーションやプラグインへのアクセスを許可します。詳細については、このガイドで後述する「アクセスキー」を参照してください。
  - リソースアクセス — リソースアクセスを使用して、Lightsail インスタンスへのアクセスを許可します。詳細については、このガイドで後述する。[リソースアクセス]]を参照してください。
- Amazon Simple Storage Service ブロックパブリックアクセス — Amazon Simple Storage Service (Amazon S3) アカウントレベルのブロックパブリックアクセス機能を使用して、Amazon S3 および Lightsail でバケットへのパブリックアクセスを一元的に制限します。ブロックパブリックアクセスは、設定された個々のバケットとオブジェクトの許可にかかわらず、すべての Amazon S3 および Lightsail のバケットをプライベートにすることができます。詳細については、このガイドで後述する「Amazon S3 パブリックアクセスブロック」を参照してください。

バケットについての詳細は、「オブジェクトストレージ」を参照してください。セキュリティのベストプラクティスの詳細については、「オブジェクトストレージのセキュリティのベストプラクティス」を参照してください。

バケットのアクセス許可

バケットのアクセス許可を使用して、バケット内のオブジェクトへの公開 (認証されていない) 読み取り専用アクセスを制御します。バケットのアクセス許可を設定する場合、以下のいずれかのオプションを選択します。

- すべてのオブジェクトがプライベート — バケット内のすべてのオブジェクトは、ご自身またはアクセスを許可したユーザーのみが読み取ることができます。このオプションでは、個々のオブジェクトを公開 (読み取り専用) にすることはできません。
- 個々のオブジェクトが公開可能 (読み取り専用) — バケット内のオブジェクトは、個々のオブジェクトを公開 (読み取り専用) として指定しない限り、自身またはアクセスを許可したユーザーのみが読み取ることができます。このオプションを使用すると、個々のオブジェクトを公開 (読み取り専用) にできます。詳細については、このガイドで後述する「個々のオブジェクトへのアクセス許可」を参照してください。

- すべてのオブジェクトが公開 (読み取り専用) — バケット内のすべてのオブジェクトは、インターネット上の誰でも読み取り可能です。このオプションを選択すると、バケット内のすべてのオブジェクトは、バケットのURL (たとえば、https://DOC-EXAMPLE-BUCKET.us-east-1.amazonaws.com/media/sailbot.jpg) を介してインターネット上の誰でも読み取り可能になります。

バケットアクセス許可の設定に関する詳細については、「バケットアクセス許可の設定」を参照してください。

個々のオブジェクトのアクセス許可

個々のオブジェクトアクセス許可を使用して、認証なしで公開されたバケット内の個々のオブジェクトの読み取り専用アクセスを制御します。個々のオブジェクトのアクセス権は、バケットのアクセス許可が、個々のオブジェクトの公開 (読み取り専用) を許可している場合にのみ設定できます。個々のオブジェクトのアクセス許可を設定する場合は、以下のいずれかのオプションを選択します。

- プライベート — このオブジェクトはご自身とアクセスを許可したユーザーのみが読み取ることができます。
- 公開 (読み取り専用) — このオブジェクトは、インターネット上の誰でも読み取り可能です。個々のオブジェクトは、インターネット上の誰でもバケットの URL を通じて読み取れるようになります (たとえば、https://DOC-EXAMPLE-BUCKET.us-east-1.amazonaws.com/media/sailbot.jpg)。

個々のオブジェクトのアクセス許可の設定に関する詳細については、「バケット内の個々のオブジェクトに対するアクセス許可の設定」を参照してください。

クロスアカウントアクセス

クロスアカウントアクセスを使用すると、他の AWS アカウントとそのユーザーに対して、バケット内のすべてのオブジェクトに対する認証された読み取り専用のアクセスが付与されます。クロスアカウント アクセスは、他の AWS アカウントとオブジェクトを共有したい場合に最適です。別の AWS アカウントにクロスアカウントアクセスを付与すると、そのアカウントのユーザーは、バケットの URL (例えば、https://DOC-EXAMPLE-BUCKET.us-east-1.amazonaws.com/media/sailbot.jpg) を通じてバケット内のオブジェクトに読み取り専用のアクセスが可能になります。最大 10 個の AWS アカウントにアクセス権を与えることができます。

クロスアカウントアクセスの設定に関する詳細については、「バケットのクロスアカウント アクセスの設定」を参照してください。

アクセスキー

アクセスキーを使用して、バケットとそのオブジェクトへの完全な読み取りおよび書き込みアクセスを付与する認証情報セットを作成します。アクセスキーは、アクセスキー ID とシークレットアクセスキーがセットです。バケットごとに最大 2 つのアクセスキーを持つことができます。アプリケーションにアクセスキーを設定することで、アプリケーションが AWS API や AWS SDKs を使用してバケットやそのオブジェクトにアクセスできるようになります。AWS CLI でアクセスキーを設定することもできます。

アクセスキーの作成に関する詳細については、「バケットのアクセスキーの作成」を参照してください。

リソースアクセス

リソースアクセスを使用し、Lightsail インスタンスのバケットとそのオブジェクトへの完全な読み取りおよび書き込みアクセスを許可します。リソースアクセスでは、アクセスキーなどの認証情報を管理する必要はありません。インスタンスへのアクセスを許可するには、インスタンスを同じAWS リージョン のバケットにアタッチします。アクセスを拒否するには、インスタンスをバケットからデタッチします。リソースアクセスは、インスタンス上のアプリケーションで、バケット上のファイルをプログラムでアップロードしたりアクセスしたりするように設定する場合に最適です。このようなユースケースの 1 つに、メディアファイルをバケットに保存するように WordPress インスタンスを設定するものがあります。詳細については、「チュートリアル: バケットを WordPress インスタンスに接続する」および「チュートリアル: バケットをコンテンツ配信ネットワークディストリビューションと使用する」を参照してください。

リソースアクセスの設定に関する詳細については、「バケットのリソースアクセスの設定」を参照してください。

Amazon S3 パブリックアクセスブロック

Amazon S3 パブリックアクセスブロック機能を使用して、Amazon S3 および Lightsail でバケットへのパブリックアクセスを一元的に制限します。ブロックパブリックアクセスは、設定された個々のバケットとオブジェクトの許可にかかわらず、すべての Amazon S3 および Lightsail のバケットをプライベートにすることができます。Amazon S3 コンソール、AWS CLI、AWS SDK、および REST API を使用して、Lightsail オブジェクトストレージサービス内のバケットを含めたアカウント内のすべてのバケットに対し、ブロックパブリックアクセス設定を構成します。詳細については、「バケットに対するブロックパブリックアクセス」を参照してください。

Amazon Lightsail バケットにファイルをアップロードする

Amazon Lightsail オブジェクトストレージサービスのバケットにファイルをアップロードすると、そのファイルはオブジェクトとして保存されます。オブジェクトは、オブジェクトを記述するファイルデータとメタデータから構成されます。バケットには、オブジェクトをいくつでも保存できます。

ファイルタイプ (イメージ、バックアップ、データ、ムービーなど) を問わず、各種のファイルをバケットにアップロードできます。Lightsail コンソールを使用してアップロードできる最大ファイルサイズは 2 GB です。大きなファイルをアップロードするには、Lightsail API、 AWS Command Line Interface (AWS CLI) 、または AWS SDKsを使用します。

Lightsail には、アップロードするファイルのサイズに応じて次のオプションがあります。

- Lightsail コンソールを使用して最大 2 GB のサイズのオブジェクトをアップロードする — Lightsail コンソールでは、最大 2 GB のサイズの単一のオブジェクトをアップロードできます。詳細については、このガイドで後述する「Lightsail コンソールを使用してバケットにファイルをアップロードする」を参照してください。
- AWS SDK、REST API または AWS CLI を使用した単一のオペレーションで最大 5 GB のサイズのオブジェクトをアップロードします — 単一の PUT オペレーションで、最大 5 GB のサイズの単一のオブジェクトをアップロードできます。詳細については、このガイドで後述する「AWS CLI を使用したバケットへのファイルのアップロード」を参照してください。
- AWS SDK、REST API または AWS CLI を使用してオブジェクトをパート別にアップロードする — マルチパートアップロード API を使用して、サイズが 5 MB から 5 TB の単一のラージ オブジェクトをアップロードできます。マルチパートアップロード API は大容量オブジェクトのアップロードを効率よく行えるように設計されています。1 つのオブジェクトをいくつかに分けてアップロードできます。オブジェクトのパートは、単独で、任意の順序で、または並行してアップロードできます。詳細については、「マルチパートアップロードを使用してバケットにファイルをアップロードする」を参照してください。

バケットについての詳細は、「オブジェクトストレージ」を参照してください。

オブジェクトキーの名前とバージョニング

Lightsail コンソールを使用してファイルをアップロードすると、ファイル名がオブジェクトキー名として使用されます。オブジェクトキー (またはキー名) によって、バケットに保存されているオブジェクトを一意に識別します。ファイルがアップロードされたフォルダが存在する場合、キー名のプレフィックスとして使用されます。たとえば、`sailbot.jpg` という名前のファイルを `images` と

いう名前のバケット内のフォルダーにアップロードすると、完全なオブジェクトキー名とプレフィックスは images/sailbot.jpg になります。ただし、オブジェクトはコンソールの sailbot.jpg フォルダ内で images として表示されます。オブジェクトキー名の詳細については、「オブジェクトストレージバケットのキー名」を参照してください。

Lightsail コンソールを使用してディレクトリをアップロードすると、ディレクトリ内のすべてのファイルとサブフォルダがバケットにアップロードされます。Lightsail は、アップロードされた各ファイル名とフォルダ名を組み合わせたオブジェクトキー名を割り当てます。例えば、 sample1.jpgと の 2 つのファイルimagesを含む という名前のフォルダをアップロードするとsample2.jpg、Lightsail はファイルをアップロードし、対応するキー名 images/sample1.jpgと を割り当てますimages/sample2.jpg。コンソールには、オブジェクトが sample1.jpg および sample2.jpg の images フォルダーとして表示されます。

すでに存在するキー名のファイルをアップロードし、バケットのバージョニングが有効になっていない場合、新しくアップロードされたオブジェクトが前のオブジェクトに置き換えられます。ただし、バケットでバージョニングが有効になっている場合、Lightsail は既存のオブジェクトを置き換える代わりに、オブジェクトの新しいバージョンを作成します。詳細については、「バケットのオブジェクトのバージョニングを有効化または一時停止する」を参照してください。

Lightsail コンソールを使用してバケットにファイルをアップロードする

Lightsail コンソールを使用してファイルとディレクトリをアップロードするには、以下の手順を実行します。

1. Lightsail コンソールにサインインします。
2. Lightsail ホームページで、ストレージタブを選択します。
3. ファイルとフォルダをアップロードするバケットの名前を選択します。
4. [Objects] (オブジェクト) タブで、次のいずれかのアクションを実行します。
   - ファイルとフォルダを [Objects] ページにドラッグアンドドロップします。
   - [Upload] (アップロード) を選択し、[File] (ファイル) を選択して個々のファイルをアップロードするか、[Directory] (ディレクトリ) を選択して、フォルダとそのすべてのコンテンツをアップロードします。

> **Note**
>
> [Create new folder] (新しいフォルダの作成) を選択してフォルダを作成することもできます。その後、新しいフォルダを参照して、そのフォルダにファイルをアップロードできます。

アップロードが完了すると、正常にアップロードしましたというメッセージが表示されます。

AWS CLI を使用して、バケットにファイルをアップロードするには

AWS Command Line Interface (AWS CLI) を使用してファイルやフォルダーをバケットにアップロードを完了するには、以下の手順を実行します。これは、`put-object` コマンドを使用して実行できます。詳細については、「AWS CLI コマンドリファレンス」の「put-object」を参照してください。

> **Note**
>
> この手順を続行する前に、 をインストールAWS CLIし、Lightsail と Amazon S3 用に設定する必要があります。詳細については、「Lightsail と連携AWS CLIするように を設定する」を参照してください。

1. ターミナルまたはコマンドプロントウィンドウを開きます。
2. 以下のコマンドを入力して、ファイルをバケットにアップロードします。

```
aws s3api put-object --bucket BucketName --key ObjectKey --body LocalDirectory --
acl bucket-owner-full-control
```

コマンドで、次のサンプルテキストを独自のテキストに置き換えます。

- *BucketName* を、ファイルをアップロードするバケットの名前に置き換えます。
- *ObjectKey* バケット内のオブジェクトの完全なオブジェクトキーを入力します。
- *LocalDirectoryFire* を、アップロードするファイルのコンピュータ上のローカルディレクトリフォルダパスに置き換えます。

例：

- Linux または Unix コンピュータの場合：

```
aws s3api put-object --bucket DOC-EXAMPLE-BUCKET --key images/sailbot.jpg --
body home/user/Pictures/sailbot.jpg --acl bucket-owner-full-control
```

- Windows コンピュータの場合:

```
aws s3api put-object --bucket DOC-EXAMPLE-BUCKET --key images/sailbot.jpg --
body "C:\Users\user\Pictures\sailbot.jpg" --acl bucket-owner-full-control
```

以下の例のような結果が表示されるはずです。

```
C:\>aws s3api put-object --bucket DOC-EXAMPLE-BUCKET --key images/sailbot.jpg --body "C:\Users\user\Pictures\sailbot.jpg"
{
    "ETag": "\"694d34edexampled92d64f342aa234c3\""
}
```

IPv6-onlyリクエスト用に AWS CLI を設定する

Amazon S3 は IPv6 経由のバケットアクセスをサポートしています。IPv6 での Amazon S3 API コールを使用したリクエストは、デュアルスタックのエンドポイントを使用して行います。このセクションでは、IPv6 経由でデュアルスタックのエンドポイントにリクエストを行う方法の例を示します。詳細については、「Amazon S3 ユーザーガイド」の「Amazon S3 デュアルスタックエンドポイントの使用」を参照してください。 Amazon S3 のセットアップ手順についてはAWS CLI、「Amazon Lightsail と連携AWS Command Line Interfaceするように を設定する」を参照してください。

> **Important**
>
> バケットにアクセスするクライアントやネットワークは、IPv6 の使用を有効にする必要があります。詳細については、「IPv6 到達可能性」を参照してください。

IPv6-onlyインスタンスから S3 リクエストを行うには、2 つの方法があります。すべての Amazon S3 リクエストAWS CLIを、指定した のデュアルスタックのエンドポイントに転送するように を設定できますAWS リージョン。または、指定したAWS CLIコマンド (すべてのコマンドではない) に対してのみデュアルスタックのエンドポイントを使用する場合は、すべてのコマンドに S3 デュアルスタックのエンドポイントを追加できます。

AWS CLI を設定する

AWS Config ファイルのプロファイルtrueで設定値を use_dualstack_endpointに設定して、Amazon S3 および s3api AWS CLI コマンドによって行われたすべての Amazon S3 リクエストを、指定されたリージョンのデュアルスタックエンドポイントに転送します。 AWS Config リージョンは、AWS CLI設定ファイルで指定するか、--region オプションを使用して コマンドで指定します。

次のコマンドを入力して、 を設定しますAWS CLI。

```
aws configure set default.s3.use_dualstack_endpoint true
```

```
aws configure set default.s3.addressing_style virtual
```

特定のコマンドにデュアルスタックのエンドポイントを追加する

コマンドごとにデュアルスタックのエンドポイントを使用するには、任意の s3 https://s3.dualstack.*aws-region*.amazonaws.comまたは s3api コマンドhttp://s3.dualstack.*aws-region*.amazonaws.comに対して --endpoint-urlパラメータを または に設定します。以下の例では、*bucketname* と *aws-region* をバケット名と に置き換えますAWS リージョン。

```
aws s3api list-objects --bucket bucketname --endpoint-url https://s3.dualstack.aws-region.amazonaws.com
```

Lightsail でのバケットとオブジェクトの管理

Lightsail オブジェクトストレージバケットを管理する一般的な手順は次のとおりです。

1. Amazon Lightsail オブジェクトストレージサービスのオブジェクトとバケットについて説明します。詳細については、Amazon Lightsail のオブジェクトストレージ を参照してください。
2. Amazon Lightsail でバケットに付けることができる名前について説明します。詳細については、「Amazon Lightsail のバケット命名規則」を参照してください。
3. バケットを作成して Lightsail オブジェクトストレージサービスの使用を開始します。詳細については、「Amazon Lightsail でのバケットの作成」を参照してください。
4. バケットのセキュリティのベストプラクティスと、バケットに設定できるアクセス許可について説明します。バケット内のすべてのオブジェクトをパブリックまたはプライベートにすること

も、オブジェクトを個別に選択してパブリックにすることもできます。また、アクセスキーを作成し、インスタンスをバケットに追加し、他の AWS アカウントにアクセス権を付与することで、バケットへのアクセスを許可することもできます。詳細については、「Amazon Lightsail オブジェクトストレージのセキュリティのベストプラクティス」および「Amazon Lightsail でのバケットのアクセス許可について」を参照してください。

バケットのアクセス許可について理解したら、以下のガイドを参照してバケットへのアクセスを許可してください。

- Amazon Lightsail のバケットへのパブリックアクセスをブロックする
- Amazon Lightsail でのバケットアクセス許可の設定
- Amazon Lightsail のバケット内の個々のオブジェクトに対するアクセス許可の設定
- Amazon Lightsail でのバケットのアクセスキーの作成
- Amazon Lightsail でのバケットのリソースアクセスの設定
- Amazon Lightsail でバケットのクロスアカウントアクセスを設定する

5. バケットのアクセスログの記録を有効にする方法と、アクセスログを使用してバケットのセキュリティを監査する方法について説明します。詳細については、以下のガイドを参照してください。
   - Amazon Lightsail オブジェクトストレージサービスでのバケットのアクセスログ記録
   - Amazon Lightsail オブジェクトストレージサービスのバケットのアクセスログ形式
   - Amazon Lightsail オブジェクトストレージサービスでのバケットのアクセスログ記録の有効化
   - Amazon Lightsail でバケットのアクセスログを使用してリクエストを識別する
6. Lightsail でバケットを管理する権限をユーザーに付与する IAM ポリシーを作成します。詳細については、「Amazon Lightsail でバケットを管理する IAM ポリシー」を参照してください。
7. バケット内のオブジェクトにラベルを付けて識別する方法について説明します。詳細については、「Amazon Lightsail でのオブジェクトキー名の理解」を参照してください。
8. ファイルをアップロードしてバケット内のオブジェクトを管理する方法について説明します。詳細については、以下のガイドを参照してください。
   - Amazon Lightsail のバケットにファイルをアップロードする
   - マルチパートアップロードを使用した Amazon Lightsail のバケットへのファイルのアップロード
   - Amazon Lightsail でバケット内のオブジェクトを表示する
   - Amazon Lightsail のバケット内のオブジェクトのコピーまたは移動
   - Amazon Lightsail のバケットからオブジェクトをダウンロードする

- Amazon Lightsail のバケット内のオブジェクトのフィルタリング
- Amazon Lightsail のバケット内のオブジェクトのタグ付け
- Amazon Lightsail でバケット内のオブジェクトを削除する

9. オブジェクトのバージョニングを有効にすると、バケットに保存されたあらゆるオブジェクトのあらゆるバージョンを保存、取得、復元します。詳細については、「Amazon Lightsail のバケットでのオブジェクトのバージョニングの有効化と一時停止」を参照してください。
10. オブジェクトのバージョニングを有効にすると、バケット内のオブジェクトの以前のバージョンを復元できます。詳細については、「Amazon Lightsail のバケット内のオブジェクトの以前のバージョンの復元」を参照してください。
11. バケットの使用率を監視します。詳細については、「Amazon Lightsail でのバケットのメトリクスの表示」を参照してください。
12. バケットの使用率がしきい値を超えたときにバケットメトリクスが通知されるよう、アラームを設定します。詳細については、「Amazon Lightsail でのバケットメトリクスアラームの作成」を参照してください。
13. ストレージとネットワーク転送量が不足している場合は、バケットのストレージプランを変更します。詳細については、「Amazon Lightsail でのバケットのプランの変更」を参照してください。
14. バケットを他のリソースに接続する方法について説明します。詳細については、以下のチュートリアルを参照してください。
    - チュートリアル: WordPress インスタンスを Amazon Lightsail バケットに接続する
    - チュートリアル: Lightsail コンテンツ配信ネットワークディストリビューションで Amazon Lightsail バケットを使用する
15. 使用しなくなったバケットを削除します。詳細については、「Amazon Lightsail でのバケットの削除」を参照してください。

Amazon Lightsail のコンテナサービス

Amazon Lightsail コンテナサービスは、高度スケーラブルコンピューティングとネットワークリソースで、コンテナをデプロイ、実行、管理することができます。コンテナは、コードとその依存関係をパッケージ化するソフトウェアのスタンダード単位で、アプリケーションが 1 つのコンピューティング環境から別のコンピューティング環境に迅速かつ確実に実行します。

Lightsail コンテナサービスをコンピューティング環境として提供し、AWS インフラストラクチャで、ローカルマシーンで作成したイメージ、あるいはオンラインリポジトリ、Amazon ECR Public Gallery からのイメージを使用してサービスにプッシュすることで、コンテナを実行するものであると考えてください。

Docker などのソフトウェアをインストールすることで、ローカルマシン上でコンテナをローカルで実行することもできます。Amazon Elastic Container Service (Amazon ECS) と Amazon Elastic Compute Cloud (Amazon EC2) は、コンテナを実行できる AWS インフラストラクチャ内の別のリソースです。詳細については、Amazon ECS 開発者ガイドを参照してください。

目次

コンテナ

コンテナは、コードと依存関係をパッケージ化するソフトウェアのスタンダード単位で、1 つのコンピューティング環境から別のコンピューティング環境へアプリケーションを迅速かつ確実に実行します。開発環境でコンテナを実行し、本番稼働前環境にデプロイしてから、本稼働環境にデプロイできます。開発環境がローカルマシンであるか、本番稼働前環境がデータセンターの物理サーバーであるか、運用環境がクラウドのバーチャルプライベートサーバーであるかにかかわらず、コンテナは確実に実行されます。

コンテナイメージは軽量で、スタンドアローンで実行可能な、アプリケーションの実行に必要なもの (コード、ランタイム、システムツール、システムライブラリ、設定) が全て含まれるソフトウェアのパッケージです。コンテナイメージは、ランタイム時にコンテナになります。アプリケーションとその依存関係をコンテナ化することで、ソフトウェアをデプロイするオペレーティングシステムとインフラストラクチャ上で正しく動作するかどうかを心配する必要がなくなり、コードに集中する時間を増やすことができます。

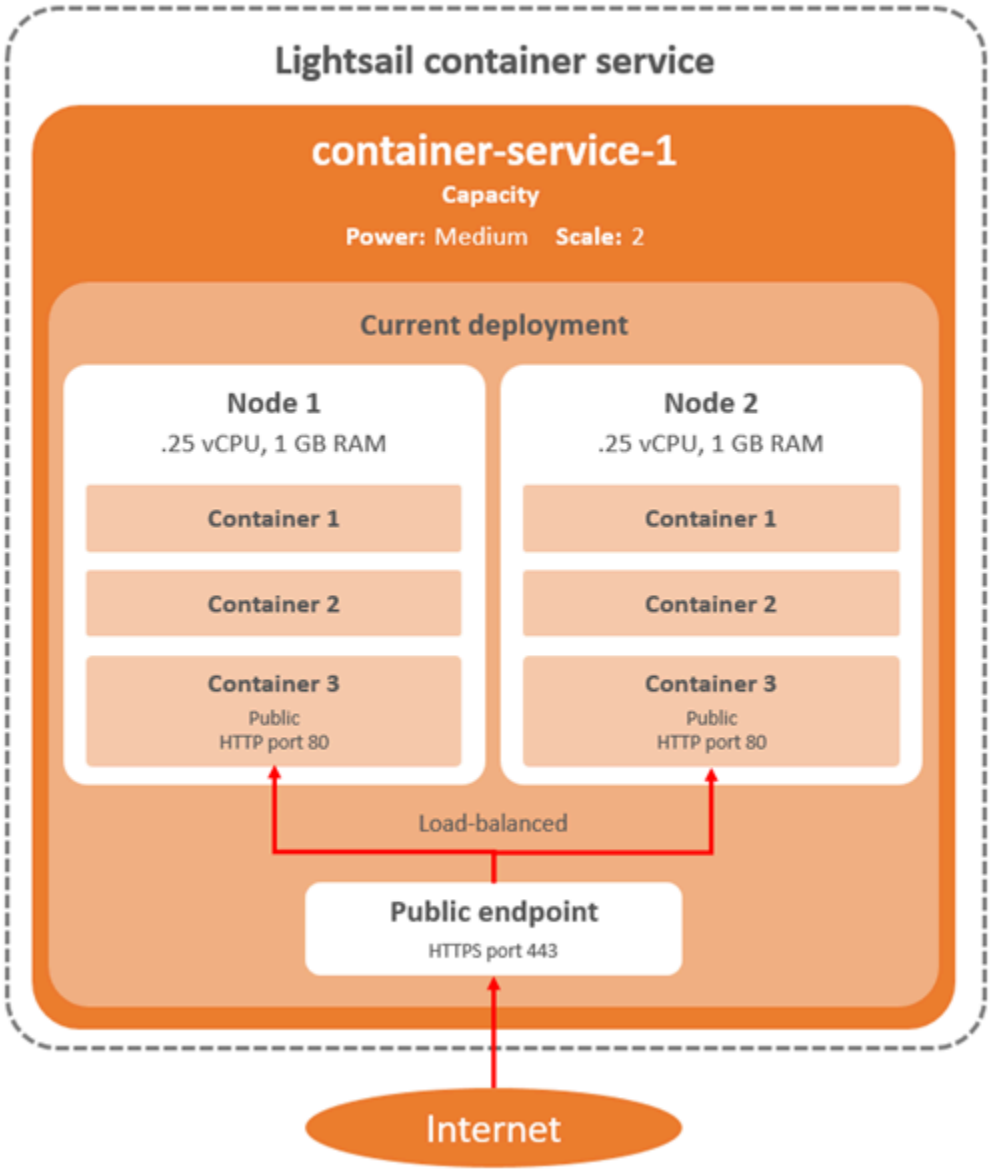
コンテナとコンテナイメージの詳細については、Docker ドキュメントの「コンテナとは」を参照してください。

Lightsail コンテナサービスの要素

Lightsail コンテナサービスの使用を開始する前に、以下の主な要素を理解しておく必要があります。

Lightsail コンテナサービス

コンテナサービスは、Lightsail コンピューティングリソースで、Lightsail が使用可能な AWS リージョン で作成できます。コンテナサービスは、いつでも作成および削除できます。詳細については、「Lightsail コンテナサービスを作成する」および「Lightsail コンテナサービスを削除する」を参照してください。



コンテナサービス容量 (スケールとパワー)

コンテナサービスを最初に作成するときは、以下の容量パラメータを選択する必要があります。

- スケール — コンテナのワークロードを実行するコンピューティングノードの数です。コンテナのワークロードは、サービスのコンピューティングノード間でコピーされます。コンテナサービスには最大 20 のコンピューティングノードを指定できます。可用性と容量を向上させるために、サービスを強化させるノードの数に応じてスケールを選択します。コンテナへのトラフィックは、ノード全体にロードバランスされます。

- パワー — コンテナサービス内の各ノードのメモリと vCPUs です。選択できるパワーは、Nano (Na)、Micro (Mi)、Small (Sm)、Medium (Md)、Large (Lg)、Xlarge (Xl) で、それぞれメモリと vCPUs の容量が徐々に大きくなっています。

コンテナサービスのスケールを 1 より大きく指定すると、コンテナワークロードはサービスの複数のコンピューティングノードにコピーされます。例えば、サービスのスケールが 3 で、パワーが Nano の場合、コンテナワークロードのコピーが 3 つあり、それぞれ 512 MB の RAM と 0.25 の vCPUs を持つ 3 つのコンピューティングリソースで実行されています。受信トラフィックは、3 つのリソース間でロードバランスされます。コンテナサービスに指定する容量が大きいほど、処理できるトラフィックが増えます。

プロビジョニングが不十分であることがわかった場合、コンテナサービスのパワーとスケールはダウンタイムなしでいつでも動的に増加でき、過剰な場合は減少することもできます。Lightsail は、現在のデプロイメントに合わせて容量の変更を自動的に管理します。詳細については、「コンテナサービスの容量を変更する」を参照してください。

料金

コンテナサービスの月額料金は、そのパワーの価格にコンピューティングノード数 (サービスのスケール) を乗算して計算されます。例えば、ミディアムパワーのサービスの価格が 40 USD、コンピューティングノードが 3 である場合、1 か月あたり 120 USD になります。コンテナサービスが有効か無効か、デプロイがあるかどうかに関わらず、コンテナサービスに対して課金されます。コンテナサービスの課金を停止するには、コンテナサービスを削除する必要があります。

各コンテナサービスには、設定された容量に関係なく、500 GB の月次データ転送クォータが含まれます。データ転送クォータは、選択したサービスのパワーとスケールに関わらず変更されることはありません。クォータを超えるデータをインターネットに転送すると、AWS リージョン によって異なる超過料金が、1 GB あたり 0.09 USD から発生します。クォータを超過したインターネットからのデータ転送には、超過料金が発生しません。詳細については、「Lightsail 料金表ページ」を参照してください。

デプロイ

Lightsail コンテナサービスでデプロイを作成できます。デプロイは、サービスで起動するコンテナワークロードの仕様のセットです。

デプロイ内の各コンテナエントリに対して、以下のパラメータを指定できます。

- 起動されるコンテナ名

- コンテナで使用するソースコンテナイメージ
- コンテナの起動時に実行するコマンド
- コンテナに適用する環境可変
- コンテナで開くネットワークポート
- コンテナサービスのデフォルトドメインを介してパブリックにアクセスできるようにする、デプロイ内のコンテナ

> **Note**
>
> デプロイ内の 1 つのコンテナのみが、各コンテナサービスに対してパブリックにアクセス可能にすることができます。

次のヘルスチェックパラメータは、デプロイの起動後にデプロイのパブリックエンドポイントに適用されます。

- ヘルスチェックを実行するディレクトリパス。
- 間隔秒、タイムアウト秒、成功コード、正常しきい値、異常しきい値など、ヘルスチェックの高度な設定。

コンテナサービスは、一度に 1 つのアクティブなデプロイを持つことができ、デプロイは最大 10 個のコンテナエントリを持つことができます。デプロイは、コンテナサービスの作成と同時に作成する、あるいはサービスが起動され実行されてから作成することができます。詳細については、「コンテナサービスのデプロイの作成と管理」を参照してください。

デプロイバージョン

コンテナサービスで作成するすべてのデプロイは、デプロイバージョンとして保存されます。既存のデプロイのパラメータを変更すると、コンテナがサービスに再デプロイされ、デプロイが変更された場合は新しいデプロイバージョンが作成されます。各コンテナサービスの最新の 50 のデプロイバージョンが保存されます。50 のデプロイバージョンのいずれかを使用して、新しいデプロイを同じコンテナに作成できます。詳細については、「コンテナサービスのデプロイの作成と管理」を参照してください。

コンテナイメージソース

デプロイを作成するときは、デプロイ内の各コンテナエントリにソースコンテナイメージを指定する必要があります。デプロイを作成した直後に、コンテナサービスは指定したソースからイメージを取り出し、それらを使用してコンテナを作成します。

以下のソースから指定してイメージを取り出せます。

- Amazon ECR Public Gallery、その他のパブリックコンテナイメージレジストリなどのパブリックレジストリ。Amazon ECR Public の詳細については、「Amazon ECR Public ユーザーガイド」の「Amazon Elastic Container Registry Public とは?」を参照してください。
- ローカルマシンからプッシュされたイメージをコンテナサービスに追加します。ローカルマシンでコンテナイメージを作成する場合は、コンテナサービスにプッシュして、デプロイの作成時に使用できます。詳細については、「コンテナサービスイメージを作成する」および「コンテナイメージをプッシュして管理する」を参照してください。

Lightsail コンテナサービスは、Linux ベースのコンテナイメージをサポートしています。Windows ベースのコンテナイメージは現在サポートされていませんが、Docker、AWS Command Line Interface (AWS CLI)、および Lightsail Control (lightsailctl) プラグインを Windows で実行することで、Linux ベースのイメージを構築して Lightsail コンテナサービスにプッシュできます。

パブリックエンドポイントとデフォルトドメイン

デプロイを作成するときに、コンテナサービスのパブリックエンドポイントとして機能するデプロイ内のコンテナエントリを指定できます。パブリックエンドポイントコンテナ上のアプリケーションは、コンテナサービスのランダムに生成されたデフォルトドメインを介して、インターネット上でパブリックにアクセスできます。デフォルトのドメインは `https://<ServiceName>.<RandomGUID>.<AWSRegion>.cs.amazonlightsail.com` 形式です。*<ServiceName>* はコンテナサービスの名前、*<RandomGUID>* は Lightsail アカウント用に AWS リージョン でランダムに生成されたグローバルに一意のコンテナサービスの識別子、*<AWSRegion>* はコンテナサービスが作成された AWS リージョン です。Lightsail コンテナサービスのパブリックエンドポイントは HTTPS のみをサポートし、TCP または UDP トラフィックはサポートされていません。サービスのパブリックエンドポイントにできるコンテナは 1 つだけです。したがって、アプリケーションのフロントエンドをホストしているコンテナをパブリックエンドポイントとして選択し、残りのコンテナは内部的にアクセス可能であることを確認してください。

コンテナサービスのデフォルトドメインか、独自のカスタムドメイン (登録されたドメイン名) を使用できます。コンテナサービスでのカスタムドメインの使用の詳細については、「コンテナサービスでカスタムドメインを有効にして管理する」を参照してください。

プライベートドメイン

また、すべてのコンテナサービスは、プライベートドメインを保持していて、*<ServiceName>*.service.local にフォーマットされており*<ServiceName>*はコンテナサービスの名前です。プライベートドメインを使用して、サービスと同じ AWS リージョンにある別の Lightsail リソースからコンテナサービスにアクセスします。プライベートドメインは、サービスのデプロイメントでパブリックエンドポイントを指定しない場合、コンテナサービスにアクセスする唯一の方法です。パブリックエンドポイントを指定しなくても、コンテナサービスに対してデフォルトのドメインが生成されますが、観覧しようとすると、404 No Such Service エラーメッセージが表示されます。

コンテナサービスのプライベートドメインを使用して特定のコンテナにアクセスするには、接続要求を受け入れるコンテナのオープンポートを指定する必要があります。これを実行するには、リクエストのドメインを*<ServiceName>*.service.local:*<PortNumber>* にフォーマットし、この中で*<ServiceName>*はコンテナサービスの名前、*<PortNumber>*は、接続したいコンテナのオープンポートです。例えば、コンテナサービスにデプロイcontainer-service-1 を作成し、Redis コンテナを指定してポート 6379 が開いている場合は、リクエストのドメインを*container-service-1*.service.local:*6379* にフォーマットします。

カスタムドメインと SSL/TLS 証明書

デフォルトドメインを使用する代わりに、コンテナサービスでカスタムドメインを最大 4 つ使用できます。例えば、カスタムドメインのトラフィックを、example.com のようにパブリックエンドポイントとしてラベル付けされたデプロイ内のコンテナにルーティングできます。

カスタムドメインをサービスで使用するには、まず、使用するドメインの SSL/TLS 証明書をリクエストする必要があります。その後、ドメインの DNS に CNAME レコードのセットを追加して、SSL/TLS 証明書を検証する必要があります。SSL/TLS 証明書の検証した後、有効な SSL/TLS 証明書をサービスに添付して、コンテナサービスでカスタムドメインを有効化します。詳細については、「Lightsail コンテナサービスの SSL/TLS 証明書を作成する」、「Lightsail コンテナサービスの SSL/TLS 証明書を検証する」、および「Lightsail コンテナサービスでカスタムドメインを有効にして管理する」を参照してください。

コンテナログ

コンテナサービスのすべてのコンテナは、コンテナのオペレーションを診断するためにアクセスできるログを生成します。ログは、コンテナ内で実行されている stdout および stderr にプロセスの流れを提供します。詳細については、「コンテナサービスログを表示する」を参照してください。

メトリクス

コンテナサービスのメトリックスをモニタリングして、過剰使用が原因の可能性とする問題を診断します。メトリックスをモニタリングして、サービスのプロビジョニングが不足していないか、あるいは過剰にプロビジョニングされているかを判断することもできます。詳細については、「コンテナサービスメトリクスを表示する」を参照してください。

Lightsail コンテナサービスを使用する

ローカルマシーンからサービスにイメージをプッシュしてデプロイで使用する場合の一般的な Lightsail コンテナサービスの管理手順は以下になります。

1. Lightsail アカウントにコンテナサービスを作成する。詳細については、「Lightsail コンテナサービスを作成する」を参照してください。
2. ローカルマシンにソフトウェアをインストールし、独自のコンテナイメージを作成し、Lightsail コンテナサービスにプッシュします。詳細については、詳細を参照して、以下のガイドラインを参照してください。
   - Lightsail コンテナサービスのコンテナイメージを管理するためのソフトウェアをインストールする
   - Lightsail コンテナサービスのコンテナイメージを作成する
   - Lightsail コンテナサービスのコンテナイメージをプッシュして管理する
3. コンテナを設定して起動するデプロイをコンテナサービス内に作成します。詳細については、「Lightsail コンテナサービスのデプロイを作成して管理する」を参照してください。
4. コンテナサービスの以前のデプロイを表示します。以前のデプロイ バージョンを使用して、新しいデプロイを作成できます。詳細については、「Lightsail コンテナサービスのデプロイバージョンを表示して管理する」を参照してください。
5. コンテナサービスのコンテナのログを表示します。詳細については、「Lightsail コンテナサービスのコンテナログを表示する」を参照してください。
6. コンテナで使用するドメイン用の SSL/TLS 証明書を作成します。詳細については、「Lightsail コンテナサービスの SSL/TLS 証明書を作成する」を参照してください。

7. ドメインの DNS にレコードを追加して、SSL/TLS 証明書を検証します。詳細については、「Lightsail コンテナサービスの SSL/TLS 証明書を検証する」を参照してください。
8. 有効な SSL/TLS 証明書をコンテナサービスに添付して、カスタムドメインを有効にします。詳細については、「Lightsail コンテナサービスのカスタムドメインを有効にして管理する」を参照してください。
9. コンテナサービスの使用状況メトリクスをモニタリングします。詳細については、「コンテナサービスメトリクスを表示する」を参照してください。
10. (オプション) パワースペックを垂直方向に増やし、スケールスペックを水平方向に増やして、コンテナサービスの容量をスケールします。詳細については、「Lightsail コンテナサービスの容量を変更する」を参照してください。
11. コンテナサービスを使用していない場合は、月額料金が発生しないようにコンテナサービスを削除します。詳細については、「Lightsail コンテナサービスを削除する」を参照してください。

パブリックレジストリからのコンテナイメージをデプロイで使用する場合、Lightsail コンテナサービスの一般的な管理手順は以下になります。

1. Lightsail アカウントにコンテナサービスを作成する。詳細については、「Lightsail コンテナサービスを作成する」を参照してください。
2. 公開レジストリからのコンテナイメージを使用する場合は、Amazon ECR Public Gallery などの公開レジストリから使用するコンテナイメージを探します。Amazon ECR Public の詳細については、「Amazon ECR Public ユーザーガイド」の「Amazon Elastic Container Registry Public とは?」を参照してください。
3. コンテナを設定して起動するデプロイをコンテナサービス内に作成します。詳細については、「Lightsail コンテナサービスのデプロイを作成して管理する」を参照してください。
4. コンテナサービスの以前のデプロイを表示します。以前のデプロイ バージョンを使用して、新しいデプロイを作成できます。詳細については、「Lightsail コンテナサービスのデプロイバージョンを表示して管理する」を参照してください。
5. コンテナサービスのコンテナのログを表示します。詳細については、「Lightsail コンテナサービスのコンテナログを表示する」を参照してください。
6. コンテナで使用するドメイン用の SSL/TLS 証明書を作成します。詳細については、「Lightsail コンテナサービスの SSL/TLS 証明書を作成する」を参照してください。
7. ドメインの DNS にレコードを追加して、SSL/TLS 証明書を検証します。詳細については、「Lightsail コンテナサービスの SSL/TLS 証明書を検証する」を参照してください。

8. 有効な SSL/TLS 証明書をコンテナサービスに添付して、カスタムドメインを有効にします。詳細については、「Lightsail コンテナサービスのカスタムドメインを有効にして管理する」を参照してください。
9. コンテナサービスの使用状況メトリクスをモニタリングします。詳細については、「コンテナサービスメトリクスを表示する」を参照してください。
10. (オプション) パワースペックを垂直方向に増やし、スケールスペックを水平方向に増やして、コンテナサービスの容量をスケールします。詳細については、「Lightsail コンテナサービスの容量を変更する」を参照してください。
11. コンテナサービスを使用していない場合は、月額料金が発生しないようにコンテナサービスを削除します。詳細については、「Lightsail コンテナサービスを削除する」を参照してください。

Lightsail コンテナサービスを作成する

このガイドでは、Lightsail コンソールを使用して作成可能な Amazon Lightsail コンテナサービスとその設定について説明します。

開始する前に、Lightsail コンテナサービスの基礎を学んでおくことをお勧めします。詳細については、「コンテナサービス」を参照してください。

コンテナサービス容量 (スケールとパワー)

コンテナサービスの容量は、最初に作成するときに選択する必要があります。容量は、次に示すパラメータの組み合わせで構成されます。

- スケール: コンテナのワークロードを実行するコンピューティングノードの数。コンテナのワークロードは、サービスのコンピューティングノード間でコピーされます。コンテナサービスには最大 20 のコンピューティングノードを指定できます。可用性と容量を向上させるために、サービスを強化させるノードの数に応じてスケールを選択します。コンテナへのトラフィックは、すべてのノードでロードバランスされます。
- パワー: コンテナサービス内の各ノードのメモリと vCPUs。選択できるパワーは、ナノ (Na)、マイクロ (Mi)、スモール (Sm)、ミディアム (Md)、ラージ (Lg)、エクストララージ (Xl) で、それぞれメモリと vCPUs の容量が徐々に大きくなります。

着信トラフィックは、コンテナサービスのスケール (コンピューティングノードの数) 全体にわたってロードバランスされます。たとえば、Nano パワーでスケールが 3 のサービスの場合、コンテナワークロードのコピーが 3 つ実行されます。各ノードには 512 MB の RAM と 0.25 の vCPUs 容量

があります。受信トラフィックは、3 つのノード間でロードバランシングされます。選択したコンテナサービス容量が大きいほど、処理できるトラフィックが増えます。

プロビジョニングが不十分であることがわかった場合、コンテナサービスのパワーとスケールはダウンタイムなしでいつでも動的に増加でき、過剰な場合は減少することもできます。Lightsail は、現在のデプロイメントに合わせて容量の変更を自動的に管理します。詳細については、「Lightsail コンテナサービスの容量を変更する」を参照してください。

料金

コンテナサービスの月額料金は、そのパワーの基本価格にスケール (コンピューティングノード数) を乗算して計算されます。たとえば、ミディアムレベルのパワーが 40 USD でスケールが 3 のサービスでは、月額 120 USD の費用がかかります。

各コンテナサービスには、構成された容量に関わらず、500 GB の月次データ転送クォータが含まれます。データ転送クォータは、選択したサービスのパワーとスケールに関わらず変更されることはありません。クォータを超えるデータをインターネットに転送すると、AWS リージョンによって異なる超過料金が、1 GB あたり 0.09 USD から発生します。クォータを超過したインターネットからのデータ転送には、超過料金が発生しません。詳細については、「Lightsail 料金表ページ」を参照してください。

コンテナサービスが有効か無効か、デプロイがあるかどうかに関係なく、コンテナサービスに対して課金されます。コンテナサービスの課金を停止するには、サービスを削除する必要があります。詳細については、「Lightsail コンテナサービスを削除する」を参照してください。

コンテナサービスステータス

コンテナサービスは、次に示す状態のいずれかになります。

- 保留中 – コンテナサービスを作成しています。
- 準備完了 – コンテナサービスは実行中ですが、アクティブなコンテナデプロイがありません。
- デプロイ – デプロイがコンテナサービスに対して起動されます。
- 実行中 – コンテナサービスが実行中で、アクティブなデプロイがあります。
- 更新中 – コンテナサービスの容量またはそのカスタムドメインが更新されています。
- 削除中 – コンテナサービスが削除されています。削除を選択した後、コンテナサービスはしばらくの間この状態を表示します。
- 無効 – コンテナサービスが無効になり、アクティブなデプロイとコンテナ (もし存在すれば) がシャットダウンされます。

コンテナサービスのサブステータス

コンテナサービスがデプロイまたは更新中の状態の場合、コンテナサービスの状態の下に、追加で次のサブ状態のいずれかが表示されます。

- システムリソースの作成 – コンテナサービスのシステムリソースが作成されています。
- ネットワークインフラストラクチャの作成 – コンテナサービスのネットワークインフラストラクチャが作成されています。
- プロビジョニング証明書 – コンテナサービス用の SSL/TLS 証明書が作成されています。
- プロビジョニングサービス – コンテナサービスがプロビジョニングされています。
- デプロイの作成 – デプロイがコンテナサービス上に作成されています。
- ヘルスチェック評価 – デプロイの正常性が評価されています。
- デプロイのアクティベーション – デプロイがアクティベーションされています。

コンテナサービスが保留中状態の場合、コンテナサービスの状態の下に、次の追加のサブ状態のいずれかが表示されます。

- 証明書の制限を超えました – コンテナサービスに必要な SSL/TLS 証明書の数はアカウントで許可されている証明書の最大数を超えています。
- 未知のエラー – コンテナサービスの作成中にエラーが発生しました。

コンテナサービスの作成

以下の手順に従って、Lightsail コンテナサービスを作成します。

1. Lightsail コンソールにサインインします。
2. Lightsail コンソールのホームページで、[Containers] (コンテナ) タブを選択します。
3. コンテナサービスの作成を選択します。
4. [コンテナサービスの作成]のページで、[AWS リージョン の変更] を選択し、コンテナサービスの AWS リージョン を選択します。
5. コンテナサービスの容量を選択します。詳細については、本ガイドの「コンテナサービス容量 (スケールとパワー)」セクションを参照してください。
6. 以下のステップを実行して、コンテナサービスの作成と同時に起動されるデプロイを作成します。それ以外の場合は、手順 7 に進み、デプロイなしでコンテナサービスを作成します。

公開レジストリーからコンテナイメージを使用する予定の場合は、デプロイでコンテナサービスを作成します。ローカルマシン上のコンテナイメージを使用する予定の場合は、デプロイなしでサービスを作成します。サービスの起動と実行後に、ローカルマシンからコンテナサービスにコンテナイメージをプッシュできます。コンテナサービスに登録されているプッシュされたコンテナイメージを使用してデプロイを作成できます。

a. [Create a deployment] (デプロイを作成する) を選択します。

b. 以下のオプションのいずれかを選択します。

- デプロイ例の選択 – Lightsail チームによって事前設定されたデプロイパラメータのセットを用いてキュレートされたコンテナイメージを使用し、デプロイを作成するためには、このオプションを選択します。このオプションは、一般的なコンテナをコンテナサービス上で起動して実行するための最速かつ最も簡単な方法を提供します。
- カスタムデプロイを指定 – 選択したコンテナを指定してデプロイを作成するには、このオプションを選択します。

デプロイフォームビューが開き、新しいデプロイパラメータを入力することができます。

c. デプロイのパラメータを入力します。指定できるデプロイパラメータの詳細については、「Lightsail コンテナサービスのデプロイを作成して管理する」ガイドの「デプロイパラメータ」セクションを参照してください。

d. [コンテナエントリの追加] を選択することによって、デプロイに複数のコンテナエントリを追加できます。デプロイには最大 10 のコンテナエントリを追加することができます。

e. デプロイのパラメータを入力し終わったら、[保存してデプロイ] を選択して、コンテナサービス上にデプロイを作成します。

7. コンテナサービスの名前を入力します。

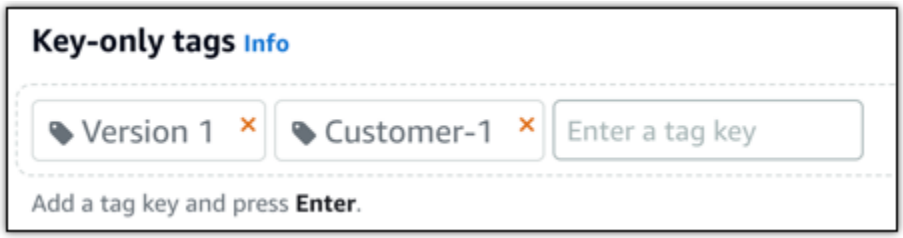
コンテナサービス名は、次のものである必要があります。

- Lightsail アカウントの各 AWS リージョン 内で一意である必要があります。
- 2〜63 文字を使用する必要があります。
- 英数字またはハイフンのみを使用する必要があります。
- ハイフン (-) で単語を区切ることができますが、名前の先頭または末尾に付けることはできません。

> **Note**
>
> 指定する名前は、コンテナサービスのデフォルトのドメイン名の一部となり、一般ユーザーに表示されます。

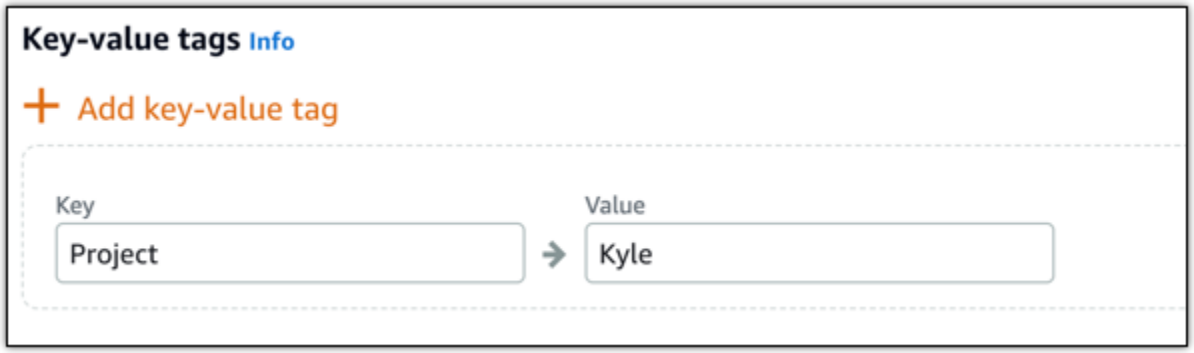
8. 以下のいずれかのオプションを選択して、コンテナサービスにタグを追加します。

   - [key-only タグの追加] または [key-only タグの編集] (タグが追加済みの場合)。タグキーのテキストボックスに新しいタグを入力し、Enter キーを押します。タグの入力を完了したら、[保存] を選択してタグを追加し、追加しない場合は、[キャンセル] を選択します。

   - [key-value タグの作成] から [キー] テキストボックスにキーを入入力し、[値] テキストボックスに値を入力します。タグの入力を完了したら、[保存] を選択し、追加しない場合は、[キャンセル] を選択します。

     キーバリューのタグは、保存する際に一つずつ追加することができます。さらに key-value タグを追加するには、以上のステップを繰り返します。

> **Note**
>
> 「キーのみ」のタグと「キーバリュー」のタグの詳細については、「タグ」を参照してください。

9. [コンテナサービスの作成] を選択します。

新しいコンテナサービスの管理ページにリダイレクトされます。作成している間、新しいコンテナサービスのステータスは「保留中」となります。しばらくすると、現在のデプロイがない場合、サービスの状態は「準備完了」を表示し、デプロイが作成された場合、「実行中」が表示されます。

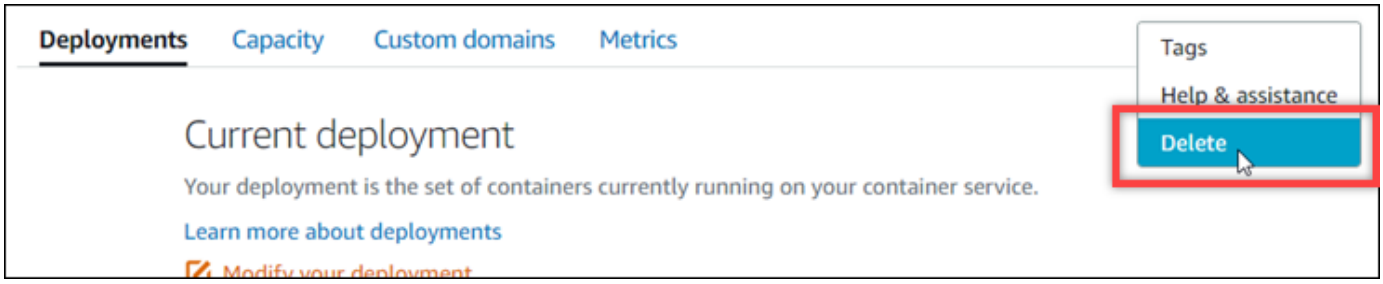
Lightsail コンテナサービスを削除する

Amazon Lightsail コンテナサービスを使用しなくなった場合、いつでも削除することができます。コンテナサービスを削除すると、そのサービスに関連付けられているすべてのデプロイと登録済みコンテナイメージが完全に破棄されます。ただし、作成した SSL/TLS 証明書やドメインは Lightsail アカウントに残るので、別のリソースで使用することができます。コンテナサービスの詳細については、「Amazon Lightsail のコンテナサービス」を参照してください。

コンテナサービスを削除

以下の手順を実行して、コンテナサービスを削除します。

1. Lightsail コンソールにサインインします。
2. Lightsail コンソールのホームページで、[Containers] (コンテナ) タブを選択します。
3. 削除するコンテナサービスの名前を選択します。
4. タブメニューで省略記号アイコンを選択し、[削除] を選択します。



5. [コンテナサービスを削除する] をクリックしてサービスを削除します。
6. 表示されるプロンプトで、「はい、削除します」を選択して、削除が永続的であることを確認します。

   しばらくすると、コンテナサービスが削除されます。

Lightsail コンテナサービスのイメージ

Docker を使用して、コンテナに基づいた分散アプリケーションの構築、実行、テスト、デプロイを行えます。Amazon Lightsail コンテナサービスは、デプロイで Docker コンテナイメージを使用してコンテナを起動します。

このガイドでは、Dockerfile を使用してローカルマシン上にコンテナイメージを作成する方法を説明します。イメージが作成されたら、そのイメージを Lightsail コンテナサービスにプッシュしてデプロイできます。

このガイドの手順を完了するには、Docker の概要と機能についての基本的な理解が必要です。Docker の詳細については、「Docker とは」、「Docker の概要」を参照してください。

目次

ステップ 1: 前提条件を満たす

作業を開始する前に、コンテナの作成に必要なソフトウェアをインストールし、Lightsail コンテナサービスにプッシュする必要があります。たとえば、Docker をインストールして使用して、Lightsail コンテナサービスで使用できるコンテナイメージを作成してビルドする必要があります。詳細については、「Amazon Lightsail コンテナサービス用のコンテナイメージを管理するソフトウェアのインストール」を参照してください。

ステップ 2: Dockerfile を作成してコンテナイメージを構築する

以下のステップを実行して Dockerfile を作成し、`mystaticwebsite` Docker コンテナイメージを構築します。コンテナイメージは、Ubuntu の Apache ウェブサーバーでホストされている単純な静的ウェブサイト用です。

1. `mystaticwebsite` の作成フォルダを作成し、Dockerfile を保存するローカルマシン上に配置します。

2. 先ほど作成したフォルダに Dockerfile を作成します。

   Dockerfile は、.TXT のようなファイル拡張子を使用しません。完全なファイル名は Dockerfile です。

3. コンテナイメージの設定方法に応じて次のコードブロックのいずれかをコピーし、Dockerfile に貼り付けます。

   - Hello World メッセージを含む単純な静的なウェブサイトコンテナイメージを作成する場合、次のコードブロックをコピーして Dockerfile に貼り付けます。このコードサンプルは Ubuntu 18.04 イメージを使用します。RUN の手順により、パッケージキャッシュが更新され、Apache がインストールされて設定されてから、Hello World のメッセージがウェブサーバーのドキュメントルートに出力されます。EXPOSE の命令はコンテナ上のポート 80 を公開し、CMD の命令はウェブサーバーを起動します。

     ```
     FROM ubuntu:18.04

     # Install dependencies
     RUN apt-get update && \
      apt-get -y install apache2

     # Write hello world message
     RUN echo 'Hello World!' > /var/www/html/index.html

     # Open port 80
     EXPOSE 80

     # Start Apache service
     CMD ["/usr/sbin/apache2ctl", "-D", "FOREGROUND"]
     ```

   - 静的ウェブサイトコンテナイメージに独自の HTML ファイルセットを使用する場合には、html フォルダを Dockerfile の保存先と同じフォルダに配置します。次に、HTML ファイルをそのフォルダに入れます。

     HTML ファイルを html フォルダに保存したら、次のコードブロックをコピーして Dockerfile に貼り付けます。このコードサンプルは Ubuntu 18.04 イメージを使用します。RUN の命令はパッケージキャッシュを更新し、Apache をインストールして設定します。COPY の命令は html フォルダの内容をウェブサーバーのドキュメントルートにコピーします。EXPOSE の命令はコンテナ上のポート 80 を公開し、CMDの命令はウェブサーバーを起動します。

     ```
     FROM ubuntu:18.04
     ```

```
# Install dependencies
RUN apt-get update && \
 apt-get -y install apache2

# Copy html directory files
COPY html /var/www/html/

# Open port 80
EXPOSE 80

CMD ["/usr/sbin/apache2ctl", "-D", "FOREGROUND"]
```

4. コマンドプロンプトまたはターミナルウィンドウを開き、 Dockerfile を格納しているフォルダにディレクトリを変更します。
5. 次のコマンドを入力して、フォルダ内の Dockerfile を使用してコンテナイメージを構築します。このコマンドは、 mystaticwebsite という新しい Docker コンテナイメージをビルドします。

   ```
   docker build -t mystaticwebsite .
   ```

   イメージが正常に構築されたことを確認するメッセージが表示されます。
6. 次のコマンドを入力して、ローカルマシン上のコンテナイメージを表示します。

   ```
   docker images --filter reference=mystaticwebsite
   ```

   次の例に示すような結果が表示され、作成された新しいコンテナイメージが示されます。

   ```
   C:\Users\[redacted]\Documents\Docker\Dockerfiles\mystaticwebsite>docker images --filter reference=mystaticwebsite
   REPOSITORY          TAG                 IMAGE ID            CREATED             SIZE
   mystaticwebsite     latest              8f7ffd1013e0        8 minutes ago       199MB
   ```

   新しく構築したコンテナイメージは、ローカルマシン上で新しいコンテナを実行させることによってテストすることができます。次の手順は、本ガイドの「ステップ 3: 新しいコンテナイメージを実行する」セクションを参照してください。

ステップ 3: 新しいコンテナイメージを実行する

作成した新しいコンテナイメージを実行するには、以下の手順に従います。

1. コマンドプロンプトまたはターミナルウィンドウに次のコマンドを入力して、本ガイドの「ステップ 2: Dockerfile を作成してコンテナイメージを構築する」のセクションで構築したコンテナーイメージを実行します。-p 8080:80 オプションは、コンテナ上の公開されたポート 80 をホストシステム上のポート 8080 にマッピングします。-d オプションは、コンテナをデタッチモードで実行するように指定します。

```
docker container run -d -p 8080:80 --name mystaticwebsite mystaticwebsite:latest
```

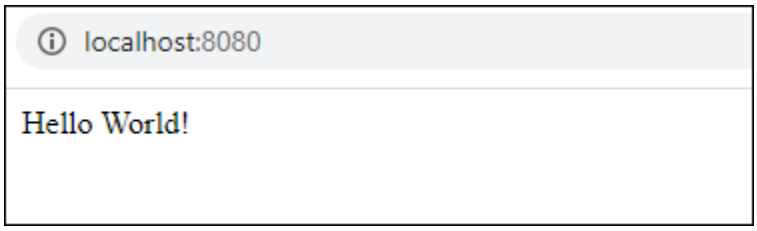
2. 次のコマンドを入力して、実行中のコンテナを表示します。

```
docker container ls -a
```

次の例に示すような結果が表示され、新しい実行中のコンテナが示されます。

```
C:\Users\[redacted]\Documents\Docker\Dockerfiles\mystaticwebsite>docker container ls -a
CONTAINER ID        IMAGE                       COMMAND                  CREATED             STATUS              PORTS                    NAMES
62382081e06b        mystaticwebsite:latest      "/bin/sh -c /root/ru…"   6 minutes ago       Up 6 minutes        0.0.0.0:8080->80/tcp     mystaticwebsite
```

3. コンテナが起動して実行されていることを確認するには、新しいブラウザウィンドウで http://localhost:8080 を開きます。次の例に示すようなメッセージが表示されます。これにより、コンテナがローカルマシン上で稼働していることが確認されます。



新しく構築されたコンテナイメージを Lightsail アカウントにプッシュする準備が整いました。これにより、Lightsail コンテナサービスにデプロイできるようになります。詳細については、Amazon Lightsail コンテナサービスのコンテナイメージのプッシュと管理 を参照してください。

（オプション）ステップ 4: ローカルマシンで実行されているコンテナをクリーンアップする

Lightsail コンテナサービスにプッシュできるコンテナイメージを作成したので、このガイドの手順に従って、ローカルマシンで実行されているコンテナをクリーンアップします。

ローカルマシンで実行されているコンテナをクリーンアップするには、以下の手順にを実行します。

1. ローカルマシンで実行されているコンテナを表示するには、次のコマンドを実行します。

```
docker container ls -a
```

次のような結果が表示され、ローカルマシンで実行されているコンテナの名前が一覧表示されます。

```
C:\Users\[redacted]\Documents\Docker\Dockerfiles\mystaticwebsite>docker container ls -a
CONTAINER ID   IMAGE                    COMMAND                  CREATED          STATUS          PORTS                    NAMES
62382081e06b   mystaticwebsite:latest   "/bin/sh -c /root/ru…"   6 minutes ago    Up 6 minutes    0.0.0.0:8080->80/tcp     mystaticwebsite
```

2. 次のコマンドを実行して、このガイドの前述の部分で作成した実行中のコンテナを削除します。これにより、コンテナは強制的に停止され、完全に削除されます。

```
docker container rm <ContainerName> --force
```

コマンドで、<ContainerName> (コンテナ名) を、削除する設定セットの名前で置き換えます。

例:

```
docker container rm mystaticwebsite --force
```

このガイドを元に作成されたコンテナは削除されます。

コンテナイメージの作成後の次のステップ

コンテナイメージを作成した後、デプロイの準備が整ったらそれらを Lightsail コンテナサービスにプッシュします。詳細については、「Lightsail コンテナサービスイメージを管理する」を参照してください。

トピック

- Lightsail コンテナサービスイメージの管理
- Lightsail コンテナーサービスプラグインをインストールする
- Lightsail の Amazon ECR プライベートリポジトリへのアクセスを管理する

Lightsail コンテナサービスイメージの管理

Amazon Lightsail コンテナサービスでデプロイを作成する場合は、コンテナエントリごとに出典コンテナイメージを指定する必要があります。Amazon ECR Public Gallery などの公開レジストリのイ

メージを使用することができます。または、ローカルマシンで作成したイメージを使用できます。このガイドでは、コンテナイメージをローカルマシンから Lightsail コンテナサービスにプッシュする方法を説明しています。コンテナイメージの作成に関する詳細については、「コンテナサービスイメージの作成」を参照してください。

目次

前提条件

コンテナサービスへのコンテナイメージのプッシュを開始する前に、次の必要条件を完了します。

- Lightsail アカウントにコンテナサービスを作成する。詳細については、「Amazon Lightsail コンテナサービスの作成」を参照してください。
- ローカルマシンにソフトウェアをインストールし、独自のコンテナイメージを作成し、Lightsail コンテナサービスにプッシュします。詳細については、「Amazon Lightsail コンテナサービス用のコンテナイメージを管理するソフトウェアのインストール」を参照してください。
- Lightsail コンテナサービスにプッシュしたい独自のコンテナイメージを、ローカルマシンに作成する。詳細については、「Amazon Lightsailコンテナサービスでのコンテナイメージの作成」を参照してください。

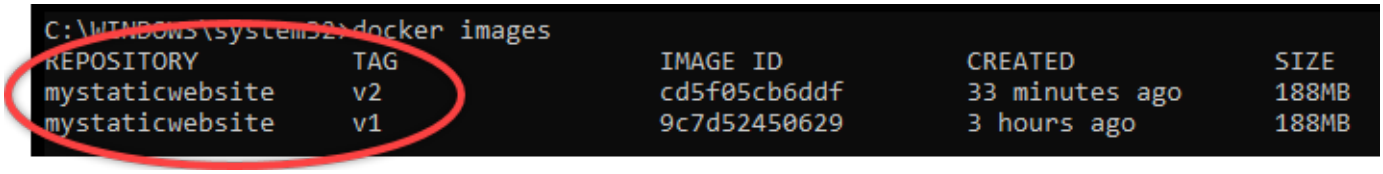
コンテナイメージをローカルマシンからコンテナサービスにプッシュする。

コンテナイメージをコンテナサービスにプッシュするには、以下の手順を実行します。

1. コマンドプロンプトまたはターミナルウィンドウを開きます。
2. コマンドプロンプトまたはターミナルウィンドウで、次のコマンドを入力して、現在ローカルマシン上にある Docker イメージ を表示します。

   ```
   docker images
   ```

3. その結果、コンテナサービスにプッシュしたいコンテナイメージ名 (リポジトリ名) とそのタグが見つけられます。これは次のステップで必要になるため、書きとめておきます。



4. 次のコマンドを入力して、ローカルマシン上のコンテナイメージをコンテナサービスにプッシュします。

```
aws lightsail push-container-image --region <Region> --service-
name <ContainerServiceName> --label <ContainerImageLabel> --
image <LocalContainerImageName>:<ImageTag>
```

コマンドを、以下のように置き換えます。

- *<Region>* をコンテナサービスが作成された AWS リージョンに置き換えます。
- *<ContainerServiceName>* をコンテナサービス名に置き換えます。
- *<ContainerImageLabel>* を、コンテナサービスに保存される際にコンテナイメージに与えたいラベルに置き換えます。登録しているコンテナイメージの異なるバージョンを追跡する際に使用できる記述的ラベルを指定します。

  このラベルは、コンテナサービスによって生成されたコンテナイメージ名の一部になります。例えば、コンテナサービス名が container-service-1 の場合には、コンテナイメージラベルは mystaticsite になり、これがユーザーがプッシュするコンテナイメージの最初のバージョンになります。そしてコンテナサービスによって生成されたイメージ名は :container-service-1.mystaticsite.1 になります。
- *<LocalContainerImageName>* を、コンテナサービスにプッシュしたいコンテナイメージ名に置き換えます。この手順の前のステップで、コンテナイメージ名は取得しています。
- *<Image Tag>* を、コンテナサービスにプッシュしたいコンテナイメージのタグに置き換えます。この手順の前のステップで、コンテナイメージのタグは取得しています。

例:

```
aws lightsail push-container-image --region us-west-2 --service-name myservice --
label mystaticwebsite --image mystaticwebsite:v2
```

次の例のような結果が表示されていれば、コンテナイメージがコンテナサービスにプッシュされたことを確認できます。

```
C:\WINDOWS\system32>aws lightsail push-container-image --service-name myservice --label mystaticwebsite
--image mystaticwebsite:v2

[1B5a355b95: Preparing
[1B0994b087: Preparing
[1B0c904ff3: Preparing
[1B370aa736: Preparing
[1Bf192bbc8: Preparing
[1Bbc0bd923: Preparing
[7BDigest: sha256:3a585ca39bba342e390b39f2fea00bbc20f492c0cda7b923dd766abe31918f3bB/1.96kB
Image "mystaticwebsite:v2" registered.
Refer to this image as ":myservice.mystaticwebsite.2" in deployments.
```

このガイドの以下の「コンテナサービスに保存されているコンテナイメージを表示する」セクションを参照して、Lightsail コンソールでコンテナサービスにプッシュされたコンテナイメージを確認してください。

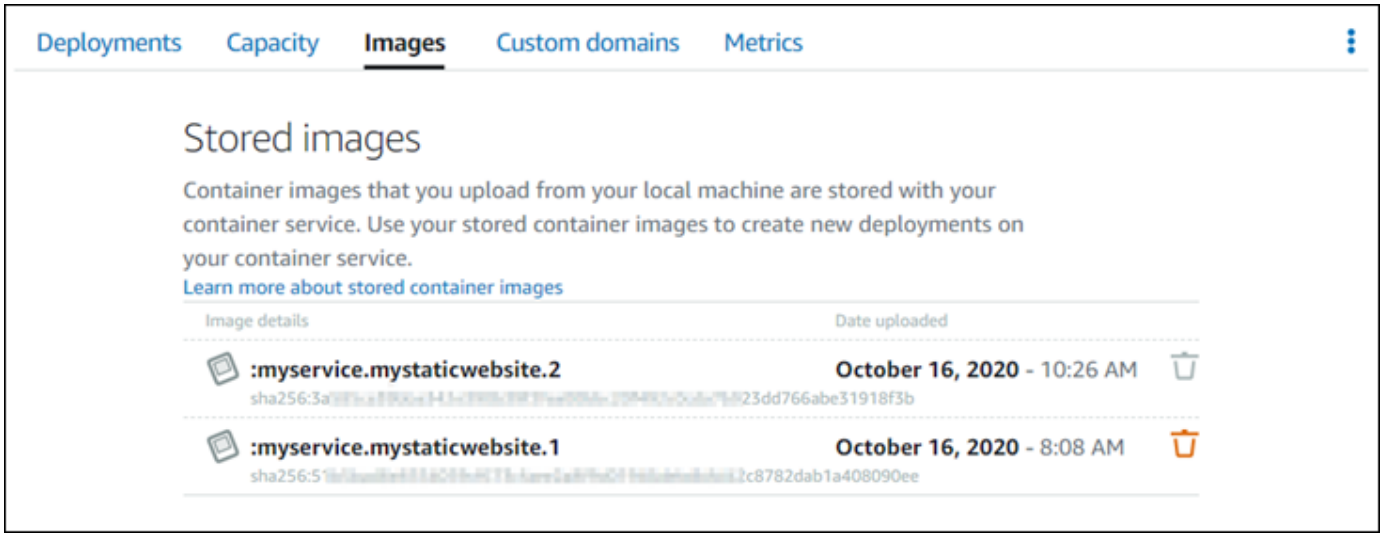
コンテナサービスに保存されているコンテナイメージを表示する

コンテナサービスにプッシュ、保存されているコンテナイメージを表示するには、以下の手順を実行します。

1. Lightsail コンソールにサインインします。
2. Lightsail コンソールのホームページで、[コンテナ] タブを選択します。
3. 表示したい保存されたコンテナイメージのコンテナサービス名を選択します。
4. コンテナサービス管理ページで、[イメージ] タブを選択します。

> **Note**
>
> コンテナサービスにイメージをプッシュしていない場合、[イメージ] タブは表示されません。コンテナサービスのイメージタブを表示するには、まずコンテナイメージをサービスにプッシュする必要があります。

[イメージ] ページには、コンテナサービスにプッシュされ、現在ユーザーのサービス内に保存されているコンテナイメージの一覧が表示されます。現在のデプロイで使用されているコンテナイメージは削除できないため、削除アイコンは灰色に表示されます。



サービスに保存されているコンテナイメージを使用して、デプロイが作成できます。詳細については、「Amazon Lightsail コンテナサービスのデプロイの作成と管理」を参照してください。

コンテナサービスに保存されているコンテナイメージを削除する

コンテナサービスにプッシュ、保存されているコンテナイメージを削除するには、以下の手順を実行します。

1. Lightsail コンソールにサインインします。
2. Lightsail コンソールのホームページで、[コンテナ] タブを選択します。
3. 現在のデプロイを表示したいコンテナサービス名を選択します。
4. コンテナサービス管理ページで、[イメージ] タブを選択します。

> **Note**
>
> コンテナサービスにイメージをプッシュしていない場合、[イメージ] タブは表示されません。コンテナサービスのイメージタブを表示するには、まずコンテナイメージをサービスにプッシュする必要があります。

5. 削除したいコンテナイメージを見つけ、削除アイコン (ごみ箱) を選択します。

> **Note**
>
> 現在のデプロイで使用されているコンテナイメージは削除できないため、削除アイコンは灰色に表示されます。

6. 確認プロンプトが表示されたら、[はい、削除します] を選択して保存されたイメージの完全な削除を確定します。

   保存されたコンテナイメージは、コンテナサービスからただちに削除されます。

Lightsail コンテナーサービスプラグインをインストールする

Amazon Lightsail コンソールを使用して Lightsail コンテナサービスを作成すると、Amazon ECR Public Gallery などのオンライン公開レジストリからコンテナイメージを使ったデプロイを作成できます。独自のコンテナイメージを作成してコンテナサービスにプッシュするには、コンテナイメージを作成する予定のコンピューター上に、以下の追加ソフトウェアをインストールする必要があります。

- Docker – 独自のコンテナイメージを実行、テスト、作成することが可能で、イメージは Lightsail コンテナサービスで使用することができます。
- AWS Command Line Interface (AWS CLI) – 作成したコンテナイメージのパラメータを指定し、それらを Lightsail コンテナサービスにプッシュすることができます。バージョン 2.1.1 とそれ以降のバージョンで、Lightsail コントロールプラグインは機能します。
- Lightsail Control (lightsailctl) plugin – ローカルマシン上にあるコンテナイメージへの AWS CLI のアクセスを可能にします。

このガイドの次のセクションでは、これらのソフトウェアパッケージをダウンロードする場所と、インストール方法について説明しています。コンテナサービスの詳細については、「コンテナサービス」を参照してください。

目次

- macOS に lightsailctl プラグインをインストールする
- Linux で lightsail ctl プラグインをインストールする

Docker をインストールする

Docker は、Linux コンテナをベースにしている配信されたアプリケーションの構築、実行、テスト、そしてデプロイを可能にするテクノロジーです。Lightsail コンテナサービスで使用することができる独自のコンテナイメージを作成したい場合は、Docker ソフトウェアをインストールして使用する必要があります。詳細については、「Lightsail コンテナサービスでのコンテナイメージの作成」を参照してください。

Docker はさまざまなオペレーティングシステムで使用できます。Ubuntu のような最新の Linux ディストリビューションや、 macOS や Windows でも使用できます。特定のオペレーティングシステムに Docker をインストールする方法の詳細については、Docker インストールガイド を参照してください。

> **Note**
>
> Docker の最新バージョンがインストールされている必要があります。旧バージョンの Docker は、このガイドで後述される AWS CLI や Lightsail コントロール (lightsailctl) プラグインで動作する保証はありません。

AWS CLI をインストールする

AWS CLI は、コマンドラインシェルでコマンドを使用して、Lightsail などの AWS サービスとやり取りするためのオープンソースツールです。ローカルマシンで作成されたコンテナイメージを Lightsail コンテナサービスにプッシュするには、AWS CLI をインストールして使用する必要があります。

AWS CLI は以下のバージョンで利用可能です。

- バージョン 2.x — 現在一般的にご利用いただける AWS CLI のリリース。こちらは AWS CLI の最新のメジャーバージョンです。最新の機能をすべてサポートしており、コンテナイメージを Lightsail コンテナサービスにプッシュする機能などを含みます。バージョン 2.1.1 とそれ以降のバージョンで、Lightsail コントロールプラグインは機能します。

- バージョン 1.x – 下位互換性のために使用できる AWS CLI の以前のバージョン。このバージョンでは、コンテナイメージを Lightsail コンテナサービスにプッシュする機能がサポートされていません。そのため、代わりに AWS CLI のバージョン 2 をインストールする必要があります。

AWS CLI のバージョン 2 は、Linux、macOS、Windows のオペレーティングシステムで使用できます。これらのオペレーティングシステムに AWS CLI をインストールする方法については、AWS CLI ユーザーガイドの「AWS CLI バージョン 2 のインストール」をご確認ください。

Lightsail コントローラー プラグインをインストールする

Lightsail コントローラー (lightsailctl) プラグインは、ローカルマシーンに作成されたコンテナイメージに AWS CLI がアクセスできるようにする軽量アプリケーションです。このプラグインは、コンテナイメージを Lightsail コンテナサービスにプッシュすることで、それらをご自身のサービスにデプロイできるようにします。

システム要件

- 64 ビット対応の Windows、macOS、および Linux オペレーティングシステム。
- lightsailctl プラグインを使用するには、AWS CLI バージョン 2 をローカルマシンにインストールする必要があります。詳細については、このガイドの前のセクションにあった「AWS CLI をインストールする」を参照してください。

最新バージョンの lightsailctl プラグインを使用する

lightsailctl プラグインは、機能強化のために随時更新されます。lightsailctl プラグインを使用する際は、最新バージョンを使用していることを確認するためのチェックが毎回実行されます。新しいバージョンが利用可能であることが判明した場合は、最新バージョンに更新して新しい機能を利用するように求められます。最新バージョンが利用可能な場合は、インストレーションのプロセスを繰り返して lightsailctl プラグインの最新バージョンを取得する必要があります。

以下の一覧は、ctl プラグインのすべてのリリースと、各バージョンに含まれる機能と強化の一覧です。

- v1.0.0 (2020 年 11 月 12 日リリース) — 初期リリースでは、AWS CLI バージョン 2 の機能が追加され、コンテナイメージを Lightsail コンテナサービスにプッシュすることが可能になりました。

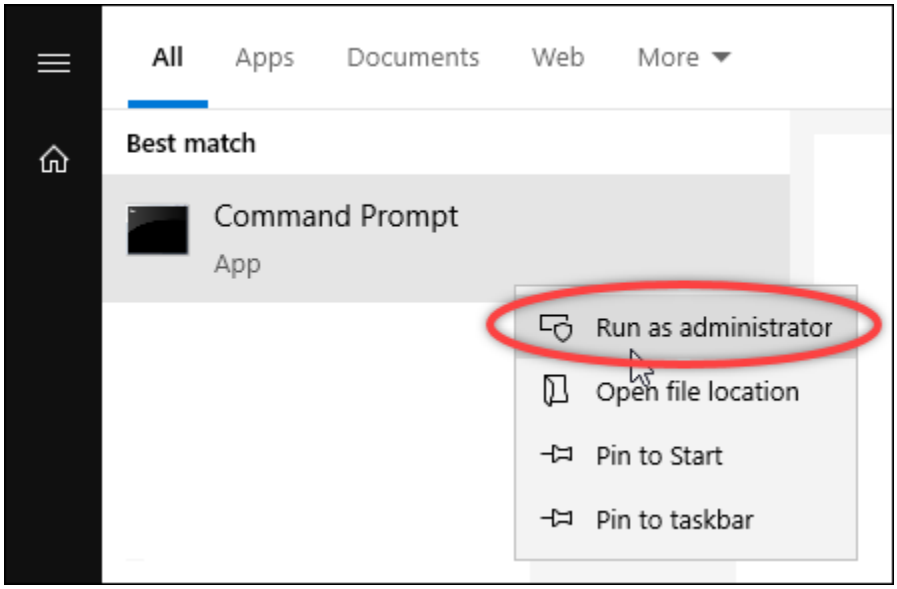
Windows に lightsailctl プラグインをインストールする

Windows に lightsailctl プラグインをインストールするには、次の手順を実行します。

1. 次の URL から実行可能ファイルをダウンロードして、C:\Temp\lightsailctl\ ディレクトリに保存します。

```
https://s3.us-west-2.amazonaws.com/lightsailctl/latest/windows-amd64/
lightsailctl.exe
```

2. Windows Start ボタンを選択して、cmd を検索します。
3. 検索結果から Command Prompt アプリケーションを右クリックし、[Run as administrator] を選択します。



> **Note**
>
> デバイスに変更を加えることを Command Prompt に許可するかの確認プロンプトが表示される場合があります。はい を選択してインストールを続行します。

4. 次のコマンドを入力してパス環境可変を設定すると、lightsailctl プラグインを保存している C:\Temp\lightsailctl\ ディレクトリが指定されます。

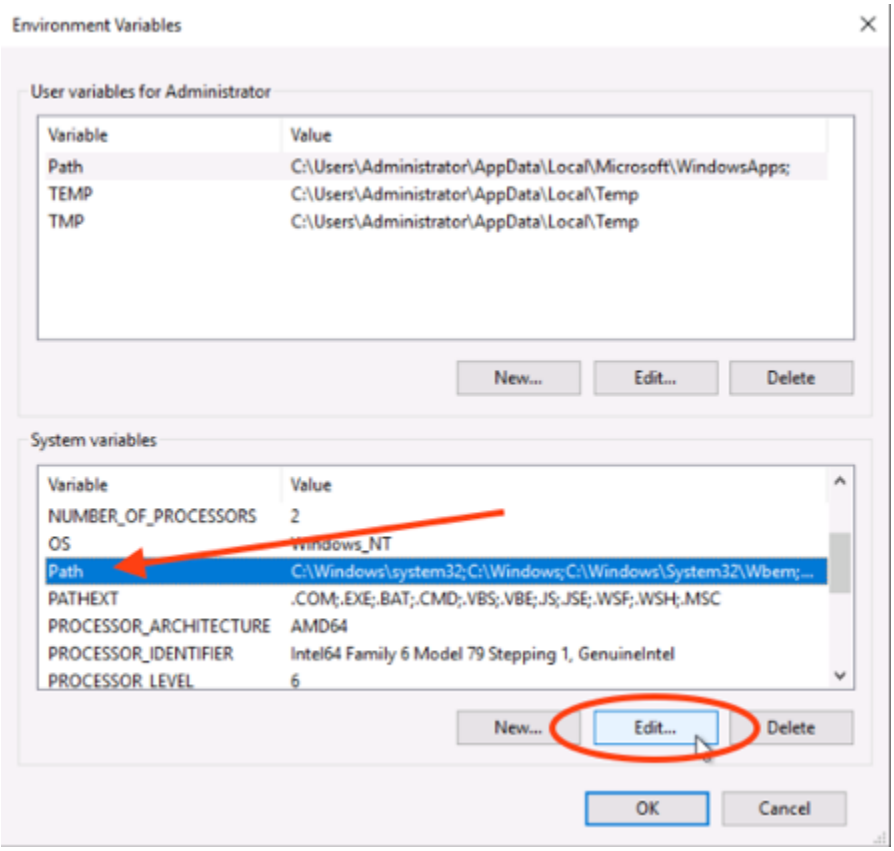
```
setx PATH "%PATH%;C:\Temp\lightsailctl" /M
```

次の例のような結果が表示されます。

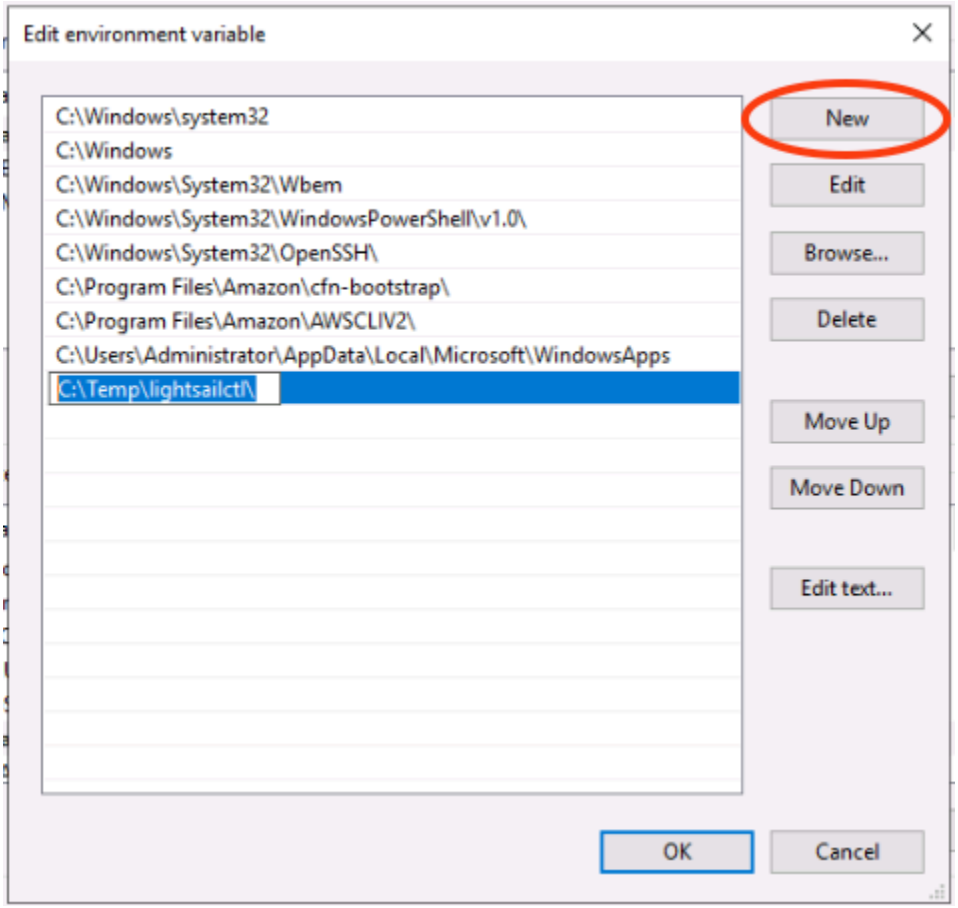
```
C:\WINDOWS\system32>setx PATH "%PATH%;C:\Temp\lightsailctl\" /M

SUCCESS: Specified value was saved.
```

`setx` コマンドは 1,024 文字を超えると切り捨てられます。PATH に複数の変数がすでに設定されている場合は、以下の手順を使用して PATH 環境変数を手動で設定します。

1. [Start] (スタート) メニューから [Control Panel] (コントロールパネル) を開きます。
2. [System and Security] (システムとセキュリティ) を選択し、[System] (システム) を選択します。
3. [システムの詳細設定] を選択します。
4. [System Properties] (システムのプロパティ) ダイアログボックスで、[Advanced] (詳細設定) タブを開き、[Environment Variables] (環境変数) を選択します。
5. [Environment Variables] (環境変数) ダイアログボックスの [System Variables] (システム変数) ボックスで、[Path] (パス) を選択します。
6. [System Variables] (システム変数) ボックスの下にある [Edit] (編集) ボタンを選択します。



7. [New] (新規) を選択し、次のパスを入力します。`C:\Temp\lightsailctl\`



8. 3 つの連続したダイアログボックスで [OK] を選択し、[System] (システム) ダイアログボックスを閉じます。

これで AWS Command Line Interface (AWS CLI) を使用してコンテナイメージを Lightsail コンテナサービスにプッシュする準備が整いました。詳しくは、「コンテナイメージをプッシュして管理する」を参照してください。

macOS に lightsailctl プラグインをインストールする

macOS に lightsailctl プラグインをダウンロードしてインストールするには、以下のいずれかの手順を実行してください。

ホームブリューダウンロードとインストール

1. ターミナルウィンドウを開きます。
2. 次のコマンドを入力して、lightsail ctl プラグインのダウンロードとインストールを行います。

```
brew install aws/tap/lightsailctl
```

> **Note**
>
> Homebrew の詳細については、Homebrew ウェブサイトを参照してください。

手動のダウンロードとインストール

1. ターミナルウィンドウを開きます。
2. 次のコマンドを入力して、lightsail ctl プラグインをダウンロードし、bin フォルダにコピーします。

```
curl "https://s3.us-west-2.amazonaws.com/lightsailctl/latest/darwin-amd64/
lightsailctl" -o "/usr/local/bin/lightsailctl"
```

3. 次のコマンドを入力して、実行可能なプラグインを作成します。

```
chmod +x /usr/local/bin/lightsailctl
```

4. 次のコマンドを入力して、プラグインの拡張属性をクリアにします。

```
xattr -c /usr/local/bin/lightsailctl
```

これで AWS CLI を使用してコンテナイメージを Lightsail コンテナサービスにプッシュする準備が整いました。詳しくは、「コンテナイメージをプッシュして管理する」を参照してください。

Linux で lightsail ctl プラグインをインストールする

Linux に Lightsail コンテナサービスプラグインをインストールするには、次の手順を実行します。

1. ターミナルウィンドウを開きます。
2. 次のコマンドを入力して、lightsail ctl プラグインをダウンロードします。
   - AMD 64 ビットのアーキテクチャーバージョンのプラグインの場合：

```
curl "https://s3.us-west-2.amazonaws.com/lightsailctl/latest/linux-amd64/
lightsailctl" -o "/usr/local/bin/lightsailctl"
```

   - ARM 64 ビットのアーキテクチャーバージョンのプラグインの場合：

```
curl "https://s3.us-west-2.amazonaws.com/lightsailctl/latest/linux-arm64/
lightsailctl" -o "/usr/local/bin/lightsailctl"
```

3. 次のコマンドを入力して、実行可能なプラグインを作成します。

```
sudo chmod +x /usr/local/bin/lightsailctl
```

これで AWS CLI を使用してコンテナイメージを Lightsail コンテナサービスにプッシュする準備が整いました。詳しくは、「コンテナイメージをプッシュして管理する」を参照してください。

Lightsail の Amazon ECR プライベートリポジトリへのアクセスを管理する

Amazon Elastic Container Registry (Amazon ECR) は AWS Identity and Access Management (IAM) を使用したリソースベースのアクセス権限によるプライベートリポジトリをサポートする AWS マネージドコンテナイメージレジストリサービスです。Amazon Lightsail コンテナサービスに対して、Amazon ECR プライベートリポジトリへのアクセスを許可することができます。その後、プライベートリポジトリからコンテナサービスにイメージをデプロイすることができます。

Lightsail コンソールまたは AWS Command Line Interface (AWS CLI) を使用して、Lightsail コンテナサービスと Amazon ECR プライベートリポジトリのアクセスを管理できます。ただし、プロセスを簡素化するため Lightsail コンソールの使用をお勧めします。

コンテナサービスの詳細については、「コンテナサービス」を参照してください。Amazon ECR の詳細については、「Amazon ECR ユーザーガイド」を参照してください。

目次

必要なアクセス許可

Lightsail コンテナサービスの Amazon ECR プライベートリポジトリへのアクセスを管理するユーザーは、IAM で以下のいずれかのアクセス許可ポリシーを持つ必要があります。詳細については、[AWS Identity and Access Management ユーザーガイド]の「IAM ID アクセス許可の追加と削除」を参照してください。

任意の Amazon ECR プライベートリポジトリにアクセス権を付与する

以下のアクセス許可ポリシーは、任意の Amazon ECR プライベートリポジトリへのアクセスを設定する権限をユーザーに付与します。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "ManageEcrPrivateRepositoriesAccess",
            "Effect": "Allow",
            "Action": [
                "ecr:SetRepositoryPolicy",
                "ecr:DescribeRepositories",
                "ecr:DeleteRepositoryPolicy",
                "ecr:GetRepositoryPolicy"
            ],
            "Resource": "arn:aws:ecr:*:AwsAccountId:repository/*"
        }
    ]
}
```

ポリシー内で、*AwsAccountId* をお使いの AWS アカウント ID ナンバーに置き換えます。

特定の Amazon ECR プライベートリポジトリにアクセス権を付与する

以下のアクセス許可ポリシーは、特定の AWS リージョン 内の特定の Amazon ECR プライベートリポジトリへのアクセスを設定する権限をユーザーに付与します。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "ManageEcrPrivateRepositoriesAccess",
            "Effect": "Allow",
            "Action": [
```

```
                "ecr:SetRepositoryPolicy",
                "ecr:DescribeRepositories",
                "ecr:DeleteRepositoryPolicy",
                "ecr:GetRepositoryPolicy"
            ],
            "Resource": "arn:aws:ecr:AwsRegion:AwsAccountId:repository/RepositoryName"
        }
    ]
}
```

ポリシー内で、次のサンプルテキストを独自のテキストに置き換えます。

- *AwsRegion* — プライベートリポジトリの AWS リージョン コード (例: us-east-1)。Lightsail コンテナサービスが、アクセスするプライベートリポジトリと同じ AWS リージョン に存在する必要があります。
- *AwsAccountId* — AWS アカウント ID ナンバー。
- *RepositoryName* — アクセスを管理するプライベートリポジトリの名前。

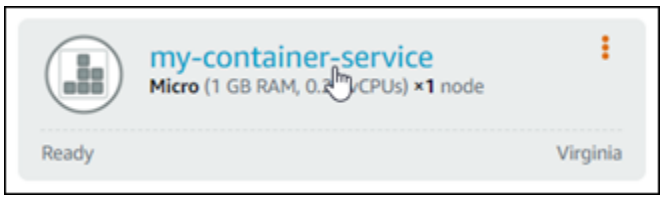
以下は、アクセス許可ポリシーに例の値を入力した一例です。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "ManageEcrPrivateRepositoriesAccess",
            "Effect": "Allow",
            "Action": [
                "ecr:SetRepositoryPolicy",
                "ecr:DescribeRepositories",
                "ecr:DeleteRepositoryPolicy",
                "ecr:GetRepositoryPolicy"
            ],
            "Resource": "arn:aws:ecr:us-east-1:111122223333:repository/my-private-repo"
        }
    ]
}
```

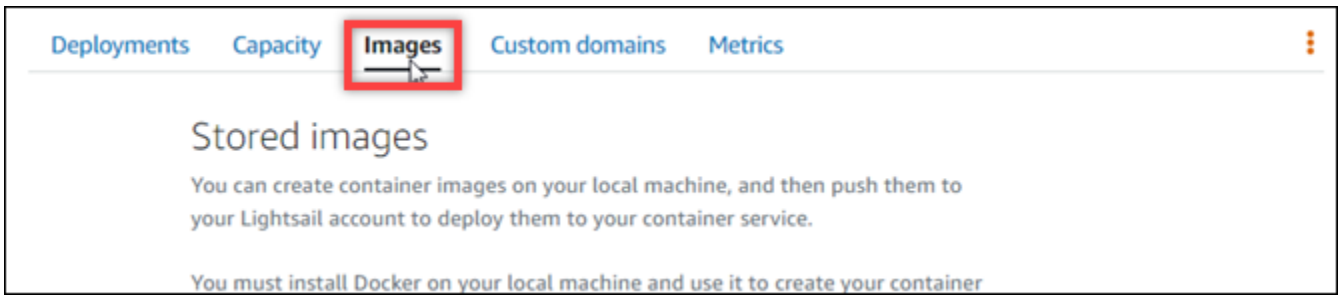
Lightsail コンソールを使用してプライベートリポジトリへのアクセスを管理する

Lightsail コンソールを使用して Lightsail コンテナサービスの Amazon ECR プライベートリポジトリへのアクセスを管理するには、次の手順を実行します。

1. Lightsail コンソールにサインインします。
2. Lightsail コンソールのホームページで、[Containers] (コンテナ) タブを選択します。
3. Amazon ECR プライベートリポジトリへのアクセスを設定したいコンテナサービスの名前を選択します。

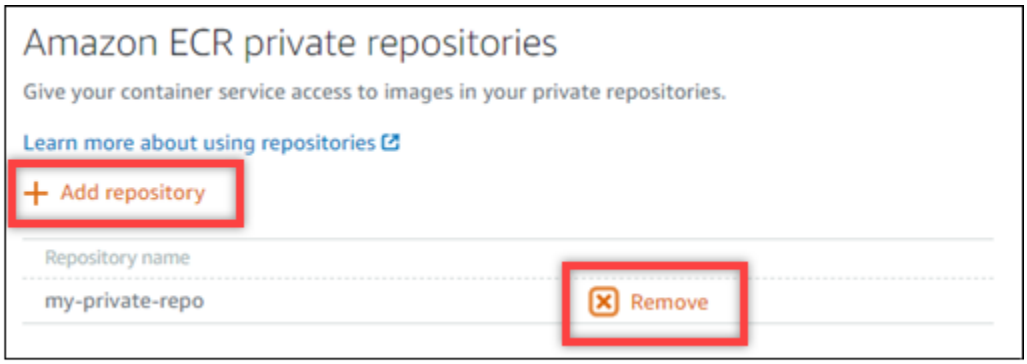


4. [Images] (イメージ) タブを選択します。

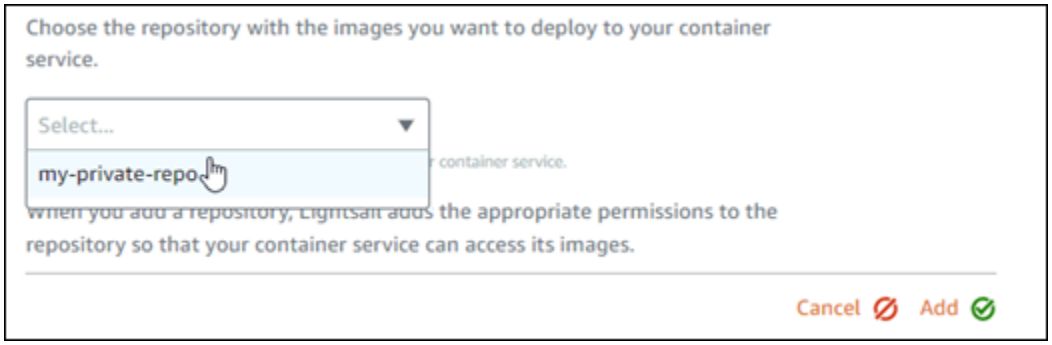


5. [リポジトリの追加] を選択すると、コンテナサービスの Amazon ECR プライベートリポジトリへのアクセス権が付与されます。

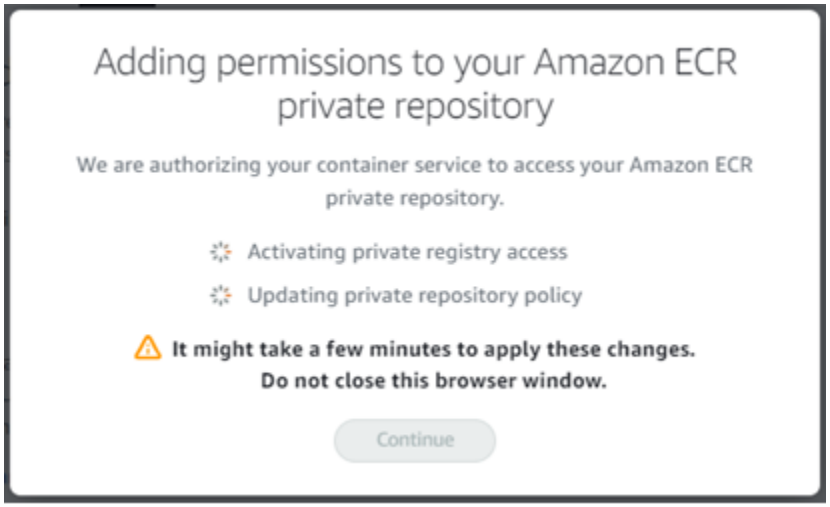
> **Note**
>
> [削除] を選択すると、以前に追加した Amazon ECR プライベートリポジトリからコンテナサービスのアクセスが削除されます。



6. 表示されるドロップダウンから、アクセスするプライベートリポジトリを選択し、[Add] (追加) を選択します。

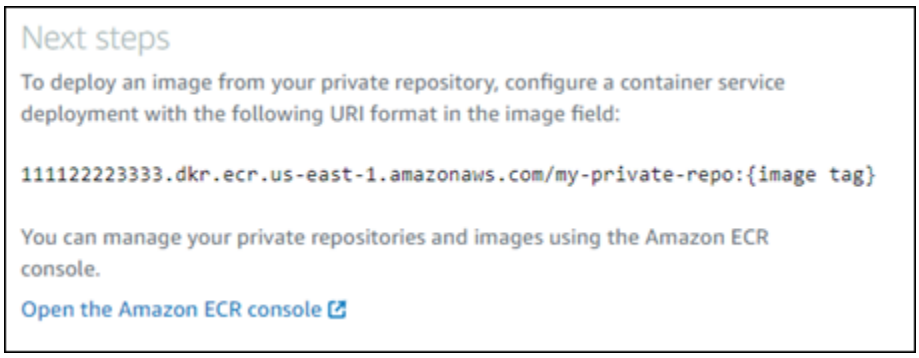


Lightsail プリンシパル Amazon リソースネーム (ARN) を含むコンテナサービスの Amazon ECR イメージプーラー IAM ロールを有効にするまでに時間がかかります。Lightsail は選択した Amazon ECR プライベートリポジトリのアクセス権限ポリシーに IAM ロールプリンシパル ARN を自動的に追加します。これにより、コンテナサービスはプライベートリポジトリとそのイメージにアクセスできるようになります。プロセスが完了し、[Continue] (続行) を選択できることを示すモーダルが表示されるまで、ブラウザウィンドウは閉じないでください。



7. アクティベーションが完了したら、[Continue] (続行) を選択します。

選択した Amazon ECR プライベートリポジトリが追加されると、このページの [Amazon ECR プライベートリポジトリ] セクションに表示されます。このページには、プライベートリポジトリから Lightsail コンテナサービスにイメージをデプロイする方法の手順が含まれています。リポジトリにあるイメージを使用するには、コンテナサービスのデプロイの作成時に、ページに [Image] (イメージ) の値として表示される URI 形式を指定します。指定する URI では、`:#####` の例を、デプロイしたいイメージのタグに置き換えます。詳細については、「コンテナサービスのデプロイの作成と管理」を参照してください。



AWS CLI を使用してプライベートリポジトリへのアクセスを管理する

AWS Command Line Interface (AWS CLI) を使用して、Amazon ECR プライベートリポジトリへの Lightsail コンテナサービスのアクセスを管理するには、以下の手順が必要です。

> **Important**
>
> Lightsail コンテナサービスの Amazon ECR プライベートリポジトリへのアクセスを管理する際には、プロセスが簡潔になるため Lightsail コンソールを使用することを推奨します。詳細については、本ガイドの前半の「Lightsail コンソールを使用してプライベートリポジトリへのアクセスを管理する」を参照してください。

1. Amazon ECR イメージプーラー IAM ロールを有効または無効化する — Lightsail の AWS CLI `update-container-service` コマンドを使用して Amazon ECR イメージプーラー IAM ロールを有効または無効にします。有効にすると、Amazon ECR イメージプーラー IAM ロールにプリンシパル Amazon リソースネーム (ARN) が作成されます。詳細については、このガイドの「Amazon ECR イメージプーラー IAM ロールを有効または無効にする」セクションを参照してください。
2. Amazon ECR プライベートリポジトリにポリシーステートメントがあるかどうかを判断する — Amazon ECR イメージプーラー IAM ロールを有効にした後、コンテナサービスでアクセスしたい Amazon ECR プライベートリポジトリに既存のポリシーステートメントがあるかどうかを判断する必要があります。詳細については、このガイドの後半にある「Amazon ECR プライベートリポジトリにポリシーステートメントがあるかどうかを判断する」を参照してください。

   リポジトリに既存のポリシーステートメントがあるかどうかに応じて、次のいずれかの方法を使用して IAM ロールプリンシパル ARN をリポジトリに追加します。

a. ポリシーステートメントを持たないプライベートリポジトリにポリシーを追加する — Amazon ECR の AWS CLI `set-repository-policy` コマンドを使用して、コンテナサービスの Amazon ECR イメージプーラーロールプリンシパル ARN を既存のポリシーを有するプライベートリポジトリに追加することができます。詳細については、本ガイドの後半にある「ポリシーステートメントを持たないプライベートリポジトリにポリシーを追加する」を参照してください。

b. ポリシーステートメントを持っているプライベートリポジトリにポリシーを追加する — Amazon ECR の AWS CLI `set-repository-policy` コマンドを使用して、コンテナサービスの Amazon ECR イメージプーラーロールを既存のポリシーを持たないプライベートリポジトリに追加することができます。詳細については、本ガイドの後半にある「ポリシーステートメントを有するプライベートリポジトリにポリシーを追加する」を参照してください。

Amazon ECR イメージプラー IAM ロールを有効または無効にする

以下の手順を完了し、Lightsail コンテナサービスの Amazon ECR イメージプーラー IAM ロールを有効または無効にします。Lightsail の AWS CLI `update-container-service` コマンドを使用することで、Amazon ECR イメージプーラー IAM ロールを有効または無効にすることができます。詳細については、「AWS CLI コマンドリファレンス」の「update-container-service」を参照してください。

> **Note**
>
> この手順を続行する前に、AWS CLI をインストールし、Lightsail 用に設定する必要があります。詳細については、「Lightsail で使用するために AWS CLI を設定する」を参照してください。

1. ターミナルまたはコマンドプロントウィンドウを開きます。
2. 次のコマンドを入力して、コンテナサービスを更新し、Amazon ECR イメージプーラー IAM ロールを有効または無効にします。

```
aws lightsail update-container-service --service-name ContainerServiceName --
private-registry-access ecrImagePullerRole={isActive=RoleActivationState} --
region AwsRegionCode
```

コマンドで、次のサンプルテキストを独自のテキストに置き換えます。

- *ContainerServiceName* — Amazon ECR イメージプーラー IAM ロールを有効または無効にするコンテナサービスの名前です。
- *RoleActivationState* — Amazon ECR イメージプーラー IAM ロールのアクティブ化状態です。ロールを有効にするには `true` を指定し、無効にするには `false` を指定します。
- *AwsRegionCode* — コンテナサービスの AWS リージョンコード (例えば、`us-east-1`)。

例:

- Amazon ECR イメージプーラー IAM ロールを有効にするには:

```
aws lightsail update-container-service --service-name my-container-service --
private-registry-access ecrImagePullerRole={isActive=true} --region us-east-1
```

- Amazon ECR イメージプーラー IAM ロールを無効にするには:

```
aws lightsail update-container-service --service-name my-container-service --
private-registry-access ecrImagePullerRole={isActive=false} --region us-east-1
```

3. オプション:

- Amazon ECR イメージプーラーロールを有効にした場合 — 前のレスポンスを取得後は、少なくとも 30 秒待機します。その後、次のステップに進んで、コンテナサービスの Amazon ECR イメージプーラー IAM ロールのプリンシパル ARN を取得します。
- Amazon ECR イメージプーラーロールを無効にした場合 — 以前に Amazon ECR イメージプーラー IAM ロールのプリンシパル ARN を Amazon ECR プライベートリポジトリのアクセス許可ポリシーに追加している場合、リポジトリからこのアクセス許可ポリシーを削除する必要があります。詳細については、「Amazon ECR ユーザーガイド」の「プライベートリポジトリポリシーステートメントを削除する」を参照してください。

4. 次のコマンドを入力して、コンテナサービスの Amazon ECR イメージプーラー IAM ロールのプリンシパル ARN を取得します。

```
aws lightsail get-container-services --service-name ContainerServiceName --
region AwsRegionCode
```

コマンドで、次のサンプルテキストを独自のテキストに置き換えます。

- *ContainerServiceName* — Amazon ECR イメージプーラー IAM ロールのプリンシパル ARN を取得するためのコンテナサービスの名前です。
- *AwsRegionCode* — コンテナサービスの AWS リージョンコード (例えば、us-east-1)。

例:

```
aws lightsail get-container-services --service-name my-container-service --
region us-east-1
```

レスポンスに ECR イメージプーラー IAM ロールのプリンシパル ARN がないか探します。ロールがリストにある場合は、コピーして書き留めます。本ガイドの次のセクションで必要になります。次に、コンテナサービスでアクセスしたい Amazon ECR プライベートリポジトリに、既存のポリシーステートメントがあるかどうかを見極める必要があります。本ガイドの「Amazon ECR プライベートリポジトリにポリシーステートメントがあるかどうかを見極める」のセクションに進んでください。

Amazon ECR プライベートリポジトリにポリシーステートメントがあるかどうかを見極める

以下の手順で、Amazon ECR プライベートリポジトリにポリシーステートメントがあるかどうかを見極めます。Amazon ECR 用の AWS CLI get-repository-policy コマンドを使用できます。詳細については、「AWS CLI コマンドリファレンス」の「update-container-service」を参照してください。

> **Note**
>
> この手順を続行する前に、AWS CLI をインストールし、Amazon ECR 用に設定する必要があります。詳細については、「Amazon ECR ユーザーガイド」の「Amazon ECR でのセットアップ」を参照してください。

1. ターミナルまたはコマンドプロントウィンドウを開きます。
2. 以下のコマンドを入力して、特定のプライベートリポジトリのポリシーステートメントを取得します。

```
aws ecr get-repository-policy --repository-name RepositoryName --
region AwsRegionCode
```

コマンドで、次のサンプルテキストを独自のテキストに置き換えます。

- *RepositoryName* — Lightsail コンテナサービスへのアクセスを設定するプライベートリポジトリの名前です。
- *AwsRegion* — プライベートリポジトリの AWS リージョン コード (例: us-east-1) です。

例:

```
aws ecr get-repository-policy --repository-name my-private-repo --region us-east-1
```

以下のレスポンスのいずれかが表示されます。

- RepositoryPolicyNotFoundException — プライベートリポジトリにポリシーステートメントがありません。リポジトリにポリシーステートメントがない場合は、本ガイドの後半にある「ポリシーステートメントを持たないプライベートリポジトリにポリシーを追加する」のセクションにある手順に従います。

```
C:\>aws ecr get-repository-policy --repository-name my-private-repo

An error occurred (RepositoryPolicyNotFoundException) when calling the GetRepositoryPolicy operation: Repository policy
does not exist for the repository with name 'my-private-repo' in the registry with id '[redacted]'
```

- リポジトリポリシーが見つかった場合 - プライベートリポジトリにはポリシーステートメントがあり、リクエストに対するレスポンスに表示されます。リポジトリにポリシーステートメントがある場合は、既存のポリシーをコピーして、本ガイドの後半にある「ポリシーステートメントを有するプライベートリポジトリにポリシーを追加する」のセクションにある手順に従います。

```
C:\>aws ecr get-repository-policy --repository-name my-private-repo
{
    "registryId": "[redacted]",
    "repositoryName": "my-private-repo",
    "policyText": "{\n  \"Version\" : \"2012-10-17\",\n  \"Statement\" : [ {\n    \"Sid\" : \"AllowUserPushPull\",\n
\"Effect\" : \"Allow\",\n    \"Principal\" : {\n      \"AWS\" : \"arn:aws:iam::[redacted]:user/example-user\"\n    },\
n    \"Action\" : [ \"ecr:BatchGetImage\", \"ecr:BatchCheckLayerAvailability\", \"ecr:CompleteLayerUpload\", \"ecr:GetDo
wnloadUrlForLayer\", \"ecr:InitiateLayerUpload\", \"ecr:PutImage\", \"ecr:UploadLayerPart\" ]\n  } ]\n}"
}
```

ポリシーステートメントを持たないプライベートリポジトリにポリシーを追加する

以下の手順に従って、ポリシーステートメントを持たない Amazon ECR プライベートリポジトリにポリシーを追加します。追加するポリシーには、Lightsail コンテナサービスの Amazon ECR イメー

ジプーラー IAM ロール プリンシパル ARN を含める必要があります。これにより、コンテナサービスにアクセス権が付与され、プライベートリポジトリからイメージをデプロイできるようになります。

> **Important**
>
> Lightsail コンソールを使用してアクセスを設定すると、Lightsail は Amazon ECR イメージプーラーロールを Amazon ECR プライベートリポジトリに自動的に追加します。この場合、このセクションの手順を使用して、プライベートリポジトリに Amazon ECR イメージプーラーロールを手動で追加する必要はありません。詳細については、本ガイドの前半の「Lightsail コンソールを使用してプライベートリポジトリへのアクセスを管理する」を参照してください。

AWS CLI を使用して、プライベートリポジトリにポリシーを追加できます。これを行うには、ポリシーを含む JSON ファイルを作成し、Amazon ECR の `set-repository-policy` コマンドでそのファイルを参照します。詳細については、「AWS CLI コマンドリファレンス」の「set-repository-policy」を参照してください。

> **Note**
>
> この手順を続行する前に、AWS CLI をインストールし、Amazon ECR 用に設定する必要があります。詳細については、「Amazon ECR ユーザーガイド」の「Amazon ECR でのセットアップ」を参照してください。

1. テキストエディタを開き、次のポリシーステートメントを新しいテキストファイルに貼り付けます。

```
{
  "Version": "2008-10-17",
  "Statement": [
  {
      "Sid": "AllowLightsailPull-ecr-private-repo-demo",
      "Effect": "Allow",
      "Principal": {
        "AWS": "IamRolePrincipalArn"
      },
      "Action": [
```

```
        "ecr:BatchGetImage",
        "ecr:GetDownloadUrlForLayer"
      ]
    }
  ]
}
```

テキストファイル内では、*IamRolePrincipalArn* を、本ガイドの前半で取得したコンテナサービスの Amazon ECR イメージプーラー IAM ロールのプリンシパル ARN に置き換えてください。

2. ファイルを ecr-policy.json という名前で、コンピュータ上のアクセス可能な場所 (例: Windows では C:\Temp\ecr-policy.json、macOS や Linux では /tmp/ecr-policy.json) に保存します。
3. 作成された ecr-policy.json ファイルのファイルパスの場所を書き留めます。この手順の後半に出てくるコマンドで、これを指定します。
4. ターミナルまたはコマンドプロントウィンドウを開きます。
5. 以下のコマンドを入力して、コンテナサービスを使ってアクセスしたいプライベートリポジトリのポリシーステートメントを設定します。

```
aws ecr set-repository-policy --repository-name RepositoryName --policy-text
 file://path/to/ecr-policy.json --region AwsRegionCode
```

コマンドで、次のサンプルテキストを独自のテキストに置き換えます。

- *RepositoryName* — ポリシーを追加するプライベートリポジトリの名前です。
- *path/to/* — 本ガイドの前半部分で作成した、コンピュータ上の ecr-policy.json ファイルへのパスです。
- *AwsRegion* — プライベートリポジトリの AWS リージョン コード (例: us-east-1) です。

例:

- Windows の場合:

```
aws ecr set-repository-policy --repository-name my-private-repo --policy-text
 file://C:\Temp\ecr-policy.json --region us-east-1
```

- macOS または Linux の場合:

```
aws ecr set-repository-policy --repository-name my-private-repo --policy-text
 file:///tmp/ecr-policy.json --region us-east-1
```

これで、コンテナサービスはプライベートリポジトリとそのイメージにアクセスできるようになります。リポジトリにあるイメージを使用するには、コンテナサービスのデプロイのイメージ値として以下の URI を指定します。URI 内の##例を、デプロイしたいイメージのタグに置き換えます。詳細については、「コンテナサービスのデプロイの作成と管理」を参照してください。

```
AwsAccountId.dkr.ecr.AwsRegionCode.amazonaws.com/RepositoryName:ImageTag
```

URI 内の次のサンプルテキストを自身が使用するテキストに置き換えます。

- *AwsAccountId* — AWS アカウント ID ナンバー。
- *AwsRegion* — プライベートリポジトリの AWS リージョン コード (例: us-east-1) です。

- *RepositoryName* — コンテナイメージをデプロイするプライベートリポジトリの名前です。
- *ImageTag* — コンテナサービスにデプロイするプライベートリポジトリのコンテナイメージのタグです。

例:

```
111122223333.dkr.ecr.us-east-1.amazonaws.com/my-private-repo:myappimage
```

ポリシーステートメントを有するプライベートリポジトリにポリシーを追加する

以下の手順に従って、ポリシーステートメントを有する Amazon ECR プライベートリポジトリにポリシーを追加します。追加するポリシーには、既存のポリシーと、Lightsail コンテナサービスの Amazon ECR イメージプーラー IAM ロール プリンシパル ARN が含まれる新しいポリシーを含める必要があります。これにより、プライベートリポジトリ上にある既存のアクセス許可が維持されながら、同時にプライベートリポジトリからイメージをデプロイするためのコンテナサービスへのアクセス権も付与されます。

> **Important**
>
> Lightsail コンソールを使用してアクセスを設定すると、Lightsail は Amazon ECR イメージプーラーロールを Amazon ECR プライベートリポジトリに自動的に追加します。この場合、このセクションの手順を使用して、プライベートリポジトリに Amazon ECR イメージプーラーロールを手動で追加する必要はありません。詳細については、本ガイドの前半の「Lightsail コンソールを使用してプライベートリポジトリへのアクセスを管理する」を参照してください。

AWS CLI を使用して、プライベートリポジトリにポリシーを追加できます。これを行うには、既存のポリシーと新しいポリシーが含まれる JSON ファイルを作成します。その後、そのファイルを Amazon ECR の `set-repository-policy` コマンドで参照します。詳細については、「AWS CLI コマンドリファレンス」の「set-repository-policy」を参照してください。

> **Note**
>
> この手順を続行する前に、AWS CLI をインストールし、Amazon ECR 用に設定する必要があります。詳細については、「Amazon ECR ユーザーガイド」の「Amazon ECR でのセットアップ」を参照してください。

1. ターミナルまたはコマンドプロントウィンドウを開きます。
2. 以下のコマンドを入力して、特定のプライベートリポジトリのポリシーステートメントを取得します。

   ```
   aws ecr get-repository-policy --repository-name RepositoryName --
   region AwsRegionCode
   ```

   コマンドで、次のサンプルテキストを独自のテキストに置き換えます。

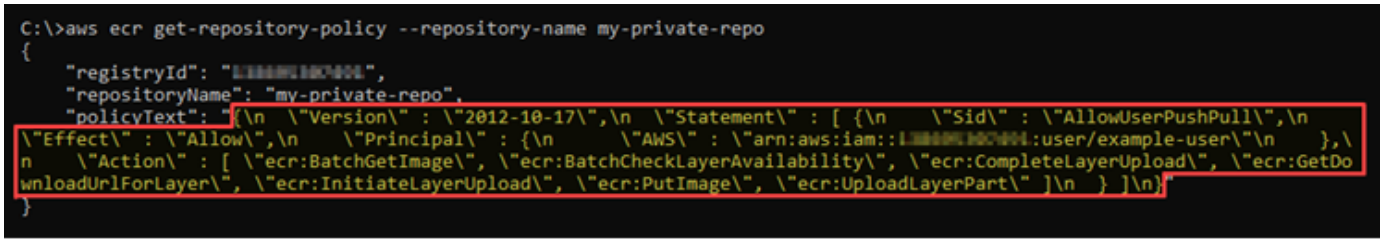
   - *RepositoryName* — Lightsail コンテナサービスへのアクセスを設定するプライベートリポジトリの名前です。
   - *AwsRegion* — プライベートリポジトリの AWS リージョン コード (例: `us-east-1`) です。

   例:

```
aws ecr get-repository-policy --repository-name my-private-repo --region us-east-1
```

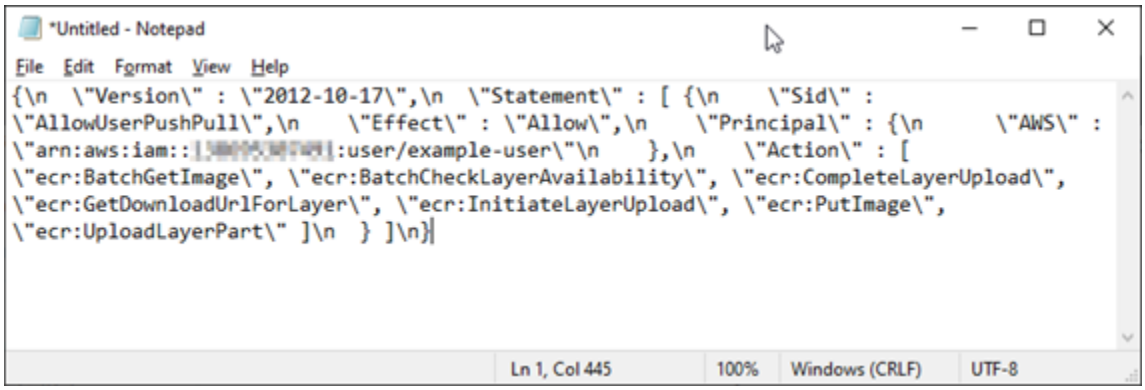
3. レスポンスに、既存のポリシーをコピーし、次のステップに進みます。

   次の例でハイライトされている部分のように、二重引用符で囲まれた policyText の内容のみをコピーする必要があります。



4. テキストエディタを開き、前の手順でコピーしたプライベートリポジトリの既存のポリシーを貼り付けます。

   結果は次の例のようになります。



5. 貼り付けたテキスト内の \n を改行に置き換え、残りの \ は削除します。

   結果は次の例のようになります。

6. テキストファイルの末尾に、次のポリシーステートメントを貼り付けます。

```
,
{
  "Version": "2008-10-17",
  "Statement": [
  {
      "Sid": "AllowLightsailPull-ecr-private-repo-demo",
      "Effect": "Allow",
      "Principal": {
        "AWS": "IamRolePrincipalArn"
      },
      "Action": [
        "ecr:BatchGetImage",
        "ecr:GetDownloadUrlForLayer"
      ]
    }
  ]
}
```

7. テキストファイル内では、*IamRolePrincipalArn* を、本ガイドの前半で取得したコンテナサービスの Amazon ECR イメージプーラー IAM ロールのプリンシパル ARN に置き換えてください。

   結果は次の例のようになります。

8. ファイルを `ecr-policy.json` という名前で、コンピュータ上のアクセス可能な場所 (例: Windows では `C:\Temp\ecr-policy.json`、macOS や Linux では `/tmp/ecr-policy.json`) に保存します。

9. `ecr-policy.json` ファイルのファイルパスの場所を書き留めます。この手順の後半に出てくるコマンドで、これを指定します。

10. ターミナルまたはコマンドプロントウィンドウを開きます。
11. 以下のコマンドを入力して、コンテナサービスを使ってアクセスしたいプライベートリポジトリのポリシーステートメントを設定します。

```
aws ecr set-repository-policy --repository-name RepositoryName --policy-text
 file://path/to/ecr-policy.json --region AwsRegionCode
```

コマンドで、次のサンプルテキストを独自のテキストに置き換えます。

- *RepositoryName* — ポリシーを追加するプライベートリポジトリの名前です。
- *path/to/* — 本ガイドの前半部分で作成した、コンピュータ上の ecr-policy.json ファイルへのパスです。
- *AwsRegion* — プライベートリポジトリの AWS リージョン コード (例: us-east-1) です。

例:

- Windows の場合:

```
aws ecr set-repository-policy --repository-name my-private-repo --policy-text
 file://C:\Temp\ecr-policy.json --region us-east-1
```

- macOS または Linux の場合:

```
aws ecr set-repository-policy --repository-name my-private-repo --policy-text
 file:///tmp/ecr-policy.json --region us-east-1
```

次の例に示すようなレスポンスが表示されます。

```
C:\>aws ecr set-repository-policy --repository-name my-private-repo --policy-text file://C:\Temp\ecr-policy.json --regio
n us-west-2
{
    "registryId": "[redacted]",
    "repositoryName": "my-private-repo",
    "policyText": "{\n  \"Version\" : \"2012-10-17\",\n  \"Statement\" : [ {\n    \"Sid\" : \"AllowLightsailPull-my-cont
ainer-service\",\n    \"Effect\" : \"Allow\",\n    \"Principal\" : {\n      \"AWS\" : \"arn:aws:iam::[redacted]:role/a
mazon/lightsail/[redacted]/containers/my-container-service/private-repo-access/[redacted]\"\n    },\n    \"Action\" : [ \"ecr:BatchGetImage\", \"ecr:GetDownloadUrlForLayer\" ]\n  }, {\n    \"Sid\" :
\"AllowUserPushPull\",\n    \"Effect\" : \"Allow\",\n    \"Principal\" : {\n      \"AWS\" : \"arn:aws:iam::[redacted]:
user/example-user\"\n    },\n    \"Action\" : [ \"ecr:BatchCheckLayerAvailability\", \"ecr:BatchGetImage\", \"ecr:Comple
teLayerUpload\", \"ecr:GetDownloadUrlForLayer\", \"ecr:InitiateLayerUpload\", \"ecr:PutImage\", \"ecr:UploadLayerPart\"
]\n  } ]\n}"
}
```

get-repository-policy コマンドをもう一度実行すると、プライベートリポジトリに新しく追加されたポリシーステートメントが表示されます。これで、コンテナサービスはプライベートリポジトリとそのイメージにアクセスできるようになります。リポジトリにあるイメージを使用するには、コンテナサービスのデプロイのイメージ値として以下の URI を指定します。URI 内の##例を、デプロイしたいイメージのタグに置き換えます。詳細については、「コンテナサービスのデプロイの作成と管理」を参照してください。

```
AwsAccountId.dkr.ecr.AwsRegionCode.amazonaws.com/RepositoryName:ImageTag
```

URI 内の次のサンプルテキストを自身が使用するテキストに置き換えます。

- *AwsAccountId* — AWS アカウント ID ナンバー。
- *AwsRegion* — プライベートリポジトリの AWS リージョン コード (例: us-east-1) です。
- *RepositoryName* — コンテナイメージをデプロイするプライベートリポジトリの名前です。
- *ImageTag* — コンテナサービスにデプロイするプライベートリポジトリのコンテナイメージのタグです。

例:

```
111122223333.dkr.ecr.us-east-1.amazonaws.com/my-private-repo:myappimage
```

Lightsail でコンテナサービスのデプロイを作成して管理する

Amazon Lightsail コンテナサービスでコンテナを起動する準備ができたら、デプロイを作成します。デプロイは、サービスに起動させたいコンテナの仕様セットです。コンテナサービスは、一度に 1 つのデプロイを実行することが可能で、デプロイは最大 10 個のコンテナエントリを持つことができます。デプロイは、コンテナサービスと同時に作成でき、あるいはサービスの起動と実行後にも作成できます。

> **Note**
>
> 新しいデプロイを作成すると、コンテナサービスの既存の使用率メトリクスが消え、新しい現在のデプロイのメトリクスだけが表示されます。

コンテナサービスの詳細については、「Amazon Lightsail のコンテナサービス」を参照してください。

目次

前提条件

コンテナサービスにデプロイの作成を開始する前に、前提条件として以下を完了します。

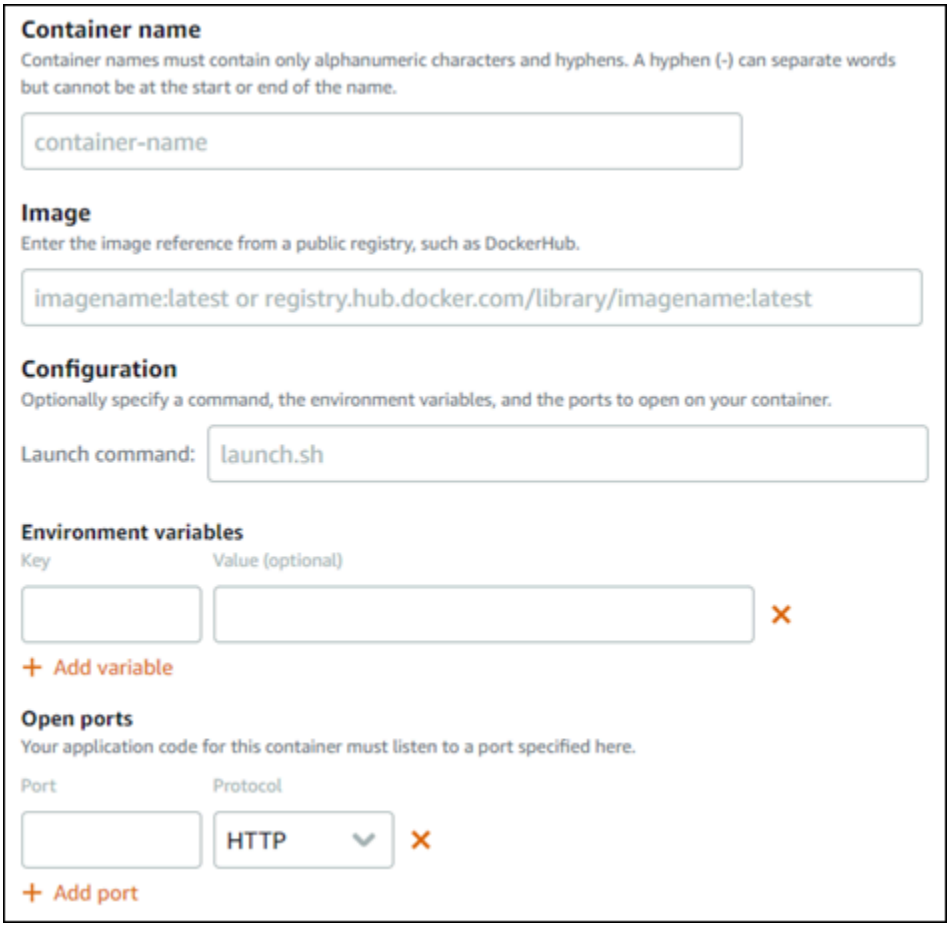
- Lightsail アカウントにコンテナサービスを作成する。詳細については、「Amazon Lightsail コンテナサービスの作成」を参照してください。
- コンテナサービスでコンテナを起動する際に使用するコンテナイメージを特定します。
  - Amazon ECR Public Gallery などのパブリックレジストリでコンテナイメージを検索します。詳細については、「Amazon ECR Public ユーザーガイド」の「Amazon ECR Public Gallery」を参照してください。
  - ローカルマシンでコンテナイメージを作成し、Lightsail コンテナサービスにプッシュします。詳細については、以下のガイドを参照してください。
    - Amazon Lightsail コンテナサービスのコンテナイメージを管理するためのソフトウェアのインストール
    - コンテナサービスのイメージを作成する
    - コンテナイメージをプッシュして管理する

デプロイパラメータ

このセクションでは、コンテナエントリと、デプロイのパブリックエンドポイントに指定できるパラメータについて説明します。

コンテナエントリパラメータ

デプロイには最大 10 個のコンテナエントリを追加できます。各コンテナエントリには、指定できる以下のパラメータがあります。



- コンテナ名 – コンテナ名を入力します。デプロイ内のすべてのコンテナは一意の名前を持つ必要があり、英数字とハイフンのみが使用可能です。ハイフンは単語を区別するために使用することができますが、名前の先頭または末尾には使用できません。
- ソースイメージ – コンテナのソースコンテナイメージを指定します。以下のソースからコンテナイメージを指定することができます。
  - Amazon ECR Public Gallery、その他のパブリックコンテナイメージレジストリなどのパブリックレジストリ。

Amazon ECR Public の詳細については、「Amazon ECR Public ユーザーガイド」の「Amazon Elastic Container Registry Public とは?」を参照してください。

- ローカルマシンからコンテナサービスにプッシュされたイメージ。保存されたイメージを指定するには、[保存されたイメージを選択] を選択し、使用するイメージを選択します。

  ローカルマシンでコンテナイメージを作成する場合は、コンテナサービスにプッシュして、デプロイを作成する時に使用することができます。詳細については、Amazon Lightsail コンテナサービスのコンテナイメージの作成およびAmazon Lightsail コンテナサービスでコンテナイメージをプッシュして管理するを参照してください。

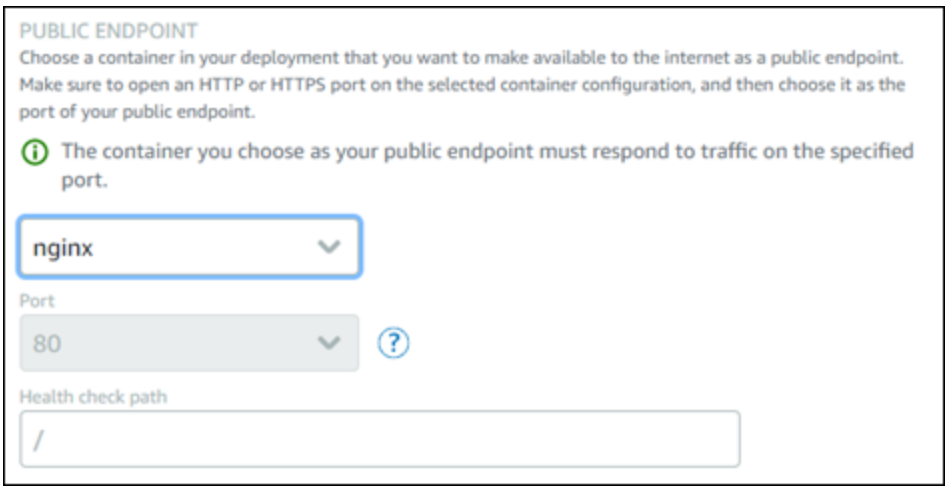
- 起動コマンド – シェルスクリプト、または コンテナの作成時にコンテナを設定する bash スクリプトを実行するための起動コマンドを指定します。起動コマンドでは、ソフトウェアの追加、ソフトウェアの更新、あるいはコンテナの設定などを他の方法で行うことができます。
- 環境可変 – 環境可変を指定します。環境可変は、コンテナによって実行されるアプリケーションまたはスクリプトの動的設定を提供するキーバリューパラメーターです。
- オープンポート – コンテナで開くポートとプロトコルを指定します。HTTP、HTTPS、TCP、および UDP 経由でポートが開くように指定できます。コンテナサービスのパブリックエンドポイントとして使用する予定のコンテナの HTTP ポートまたは HTTPS ポートを開く必要があります。詳細については、このガイドの以下のセクションを参照してください。

パブリックエンドポイントパラメータ

コンテナサービスのパブリックエンドポイントとして機能するデプロイ内のコンテナエントリを指定できます。パブリックエンドポイントコンテナ上のアプリケーションは、コンテナサービスのランダムに生成されたデフォルトドメインを介して、インターネット上でパブリックにアクセスできます。デフォルトのドメインは `https://<ServiceName>.<RandomGUID>.<AWSRegion>.cs.amazonlightsail.com` 形式です。*<ServiceName>* はコンテナサービスの名前、*<RandomGUID>* は Lightsail アカウント用に AWS リージョンでランダムに生成されたグローバルに一意のコンテナサービスの識別子、*<AWSRegion>* はコンテナサービスが作成された AWS リージョンです。Lightsail コンテナサービスのパブリックエンドポイントは HTTPS のみをサポートし、TCP または UDP トラフィックはサポートされていません。サービスのパブリックエンドポイントにできるコンテナは 1 つだけです。したがって、アプリケーションのフロントエンドをホストしているコンテナをパブリックエンドポイントとして選択し、残りのコンテナは内部的にアクセス可能であることを確認してください。

> **Note**
>
> コンテナサービスでは、独自のカスタムドメイン名を使用できます。詳細については、Amazon Lightsail コンテナサービスでのカスタムドメインの有効化と管理 を参照してください。

デプロイのパブリックエンドポイントとコンテナサービスには、以下のパラメータを指定できます。



- エンドポイントコンテナ – コンテナサービスのパブリックエンドポイントとして機能するデプロイ内のコンテナ名を選択します。デプロイで HTTP ポートまたは HTTPS ポートが開いているコンテナのみがドロップダウンメニューに表示されます。
- ポート – パブリックエンドポイントに使用する HTTP ポートまたは HTTPS ポートを選択します。選択したコンテナで開かれている HTTP ポートと HTTPS ポートのみがドロップダウンメニューに表示されます。選択したコンテナが最初に起動したときに HTTPS 接続をサポートするように設定されていない場合は、HTTP ポートを選択します。

> **Note**
>
> パブリックエンドポイントポートとして HTTP ポートを選択した場合でも、コンテナサービスのデフォルトドメインはデフォルトで HTTPS が使用されます。これは、コンテナサービスのロードバランサーがデフォルトで HTTPS に設定されていますが、HTTP を使用してコンテナとの接続を確立するためです。
> コンテナサービスのロードバランサーは HTTP を使用してコンテナに接続しますが、HTTPS を使用してユーザーにコンテンツを提供します。

- ヘルスチェックパス – コンテナサービスのロードバランサーが定期的にチェックして正常であることを確認するために選択された、パブリックエンドポイントコンテナのパスを指定します。
- ヘルスチェックの詳細設定 - 選択したパブリックエンドポイントコンテナに対して、次のヘルスチェック設定を設定できます。
  - ヘルスチェックのタイムアウト (秒) - ヘルスチェックのレスポンスを待つ時間 (秒単位)。この間にレスポンスが受信されない場合、ヘルスチェックは失敗します。2～60 秒を指定できます。
  - ヘルスチェックの間隔 (秒)-コンテナのヘルスチェックのおおよその間隔 (秒単位)。5～300 秒を指定できます。
  - ヘルスチェックの成功コード-コンテナからの正常なレスポンスを確認するために使用する HTTP コード。200 から 499 までの値を指定できます。複数の値 (例: 200,202) または値の範囲 (例: 200-299) を指定できます。
  - ヘルスチェックの健全性しきい値 - コンテナをヘルス状態に移行するために必要な連続したヘルスチェックの成功数。
  - ヘルスチェックの異常しきい値 - コンテナを異常状態に移行するために必要な連続したヘルスチェックの成功数。

プライベートドメイン

また、すべてのコンテナサービスは、プライベートドメインを保持していて、*<ServiceName>*.service.local にフォーマットされており*<ServiceName>*はコンテナサービスの名前です。プライベートドメインを使用して、サービスと同じ AWS リージョンにある別の Lightsail リソースからコンテナサービスにアクセスします。プライベートドメインは、サービスのデプロイメントでパブリックエンドポイントを指定しない場合、コンテナサービスにアクセスする唯一の方法です。パブリックエンドポイントを指定しなくても、コンテナサービスに対してデフォルトのドメインが生成されますが、観覧しようとすると、`404 No Such Service` エラーメッセージが表示されます。

コンテナサービスのプライベートドメインを使用して特定のコンテナにアクセスするには、接続要求を受け入れるコンテナのオープンポートを指定する必要があります。これを実行するには、リクエストのドメインを*<ServiceName>*.service.local:*<PortNumber>* にフォーマットし、この中で*<ServiceName>*はコンテナサービスの名前、*<PortNumber>*は、接続したいコンテナのオープンポートです。例えば、コンテナサービスにデプロイcontainer-service-1 を作成し、Redis コンテナを指定してポート 6379 が開いている場合は、リクエストのドメインを*container-service-1*.service.local:*6379* にフォーマットします。

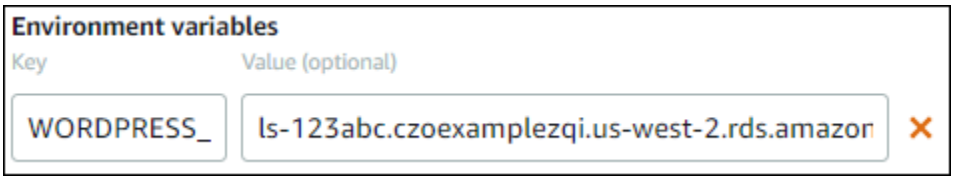
コンテナ間の通信

環境変数を使用すると、同じコンテナサービス内のコンテナ間、異なるコンテナサービス内のコンテナ、またはコンテナと他のリソース間 (コンテナとマネージドデータベース間など) の通信を開くことができます。

同じコンテナサービス内のコンテナ間の通信を開くには、次の例のように、`localhost` を参照する環境変数をコンテナのデプロイに追加します。

異なるコンテナサービスにあるコンテナ間の通信を開くには、次の例のように、プライベートドメイン (`container-service-1.service.local` など) を参照する環境変数をコンテナのデプロイに追加します。

コンテナと他のリソース間の通信を開くには、リソースのパブリックエンドポイント URL を参照する環境変数をコンテナのデプロイに追加します。例えば、Lightsail マネージドデータベースのパブリックエンドポイントは通常 `ls-123abc.czoexamplezqi.us-west-2.rds.amazonaws.com` です。したがって、次の例に示すように、環境変数でそのことを参照する必要があります。



コンテナログ

デプロイ内のすべてのコンテナがログを生成します。コンテナログは、コンテナ内で実行されている stdout および stderr にプロセスの流れを提供します。コンテナのログに定期的にアクセスして、オペレーションを診断します。詳細については、「Amazon Lightsail コンテナサービスのコンテナのログを表示」を参照してください。

デプロイバージョン

コンテナサービスで作成するすべてのデプロイは、デプロイバージョンとして保存されます。既存のデプロイのパラメータを変更すると、コンテナがサービスに再デプロイされ、デプロイが変更された場合は新しいデプロイバージョンが作成されます。各コンテナサービスの最新の 50 のデプロイバージョンが保存されます。50 のデプロイバージョンのいずれかを使用して、新しいデプロイを同じコンテナに作成できます。詳細については、「Amazon Lightsail コンテナサービスのデプロイバージョンの表示と管理」を参照してください。

デプロイのステータス

デプロイの作成後、デプロイは以下のいずれかの状態になります。

- アクティブ化中 – デプロイがアクティブ化されており、コンテナが作成されています。
- アクティブ – デプロイは正常に作成され、コンテナサービスで現在実行されています。
- 非アクティブ – 以前に正常に作成されたデプロイは、コンテナ上で実行されていません。
- 失敗 – デプロイで指定された 1 つ以上のコンテナが起動できなかったため、デプロイが失敗しました。

デプロイエラー

デプロイ内の 1 つ以上のコンテナの起動に失敗すると、デプロイは失敗します。デプロイが失敗し、コンテナサービスで以前のデプロイが実行されていた場合、コンテナサービスは以前のデプロイをアクティブなデプロイとして維持します。以前のデプロイがない場合、コンテナサービスは準備完了状態のままになり、現在アクティブなデプロイはありません。

失敗したデプロイのコンテナログを表示して、問題の診断とトラブルシューティングを行います。詳細については、「Amazon Lightsail コンテナサービスのコンテナのログを表示」を参照してください。

現在のコンテナサービスのデプロイの表示

以下の手順を実行して、Lightsail コンテナサービスの現在のデプロイを表示します。

1. Lightsail コンソールにサインインします。
2. Lightsail コンソールのホームページで、[Containers] (コンテナ) タブを選択します。
3. 現在のデプロイを表示したいコンテナサービス名を選択します。

4. コンテナサービス管理ページで、[デプロイ] タブを選択します。

   デプロイページには、現在のデプロイとデプロイバージョンが一覧表示されます。コンテナサービスでデプロイをまだ作成していない場合、ページの両方のセクションは空です。

コンテナサービスのデプロイを作成または変更

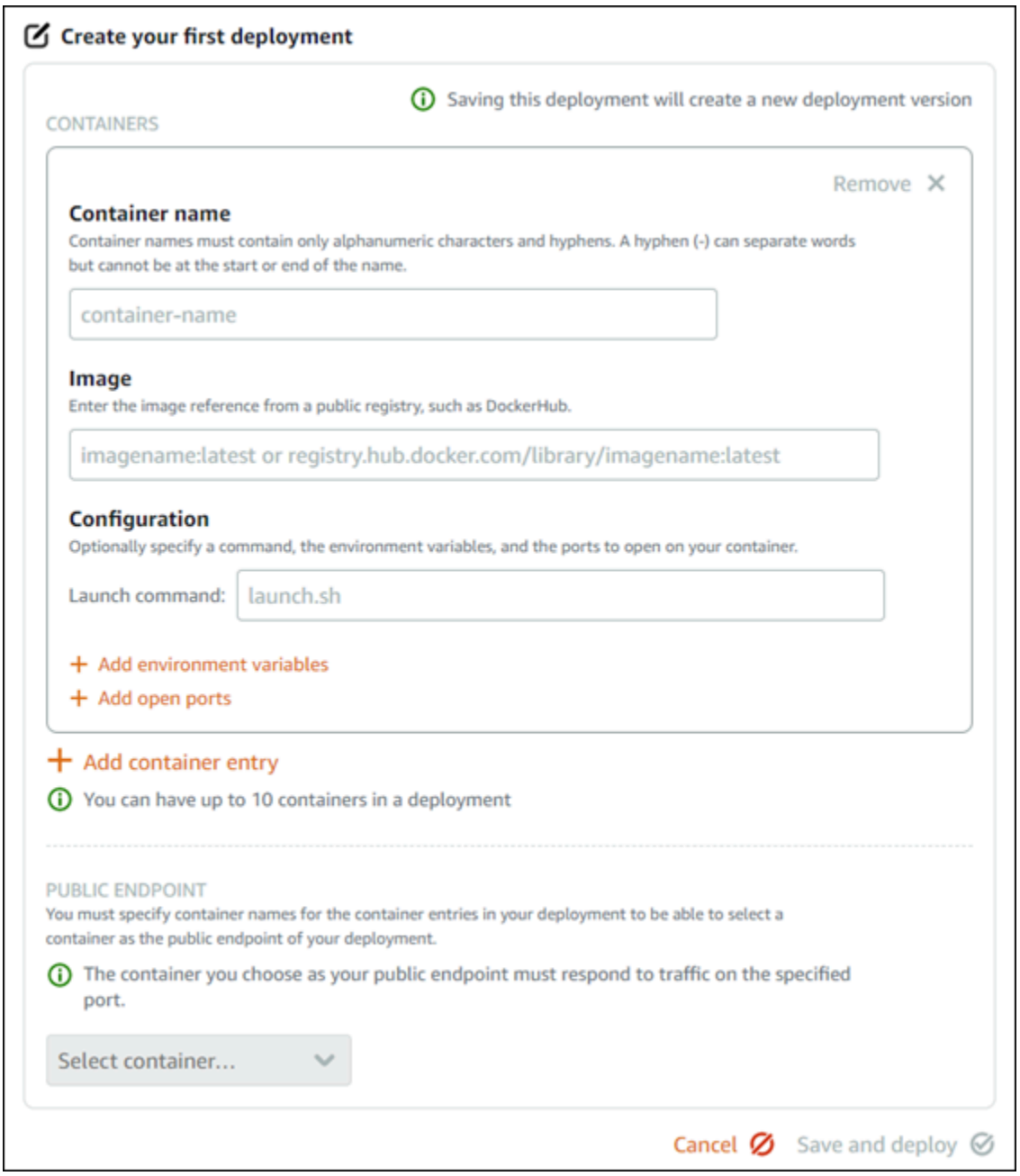
Lightsail コンテナサービスのデプロイを作成または変更するには、以下の手順を実行します。新しいデプロイを作成する場合も、既存のデプロイを変更する場合も、コンテナサービスでは、すべてのデプロイが新しいデプロイバージョンとして保存されます。詳細については、「Amazon Lightsail コンテナサービスのデプロイ バージョンの表示と管理」を参照してください。

1. Lightsail コンソールにサインインします。
2. Lightsail コンソールのホームページで、[Containers] (コンテナ) タブを選択します。
3. コンテナサービスのデプロイを作成または変更するコンテナサービス名を選択します。
4. コンテナサービスの管理ページで、デプロイタブを選択します。

   デプロイページには、現在のデプロイとデプロイのバージョンが一覧表示されます。(存在する場合)
5. 以下のオプションのいずれかを選択します。

   - コンテナサービスに既存のデプロイがある場合は、デプロイの変更を選択します。
   - コンテナサービスにデプロイがない場合は、デプロイの作成を選択します。

     デプロイフォームが開き、既存のデプロイパラメータを編集したり、新しいデプロイパラメータを入力することができます。



6. デプロイのパラメータを入力します。指定できるデプロイパラメータの詳細については、このガイドの前半のデプロイパラメータセクションを参照してください。
7. [コンテナエントリを追加] を選択して、デプロイに複数のコンテナエントリを追加します。デプロイには最大 10 のコンテナエントリを追加することができます。
8. コンテナサービスのパブリックエンドポイントとして機能するデプロイ内のコンテナエントリを指定します。これには、HTTP または HTTPS ポート、選択したコンテナエントリのヘルスチェックパス、および詳細なヘルスチェック設定の指定が含まれます。詳細については、このガイドの前半にある「パブリックエンドポイントパラメータ」を参照してください。

9. デプロイのパラメータの入力が終了したら [保存してデプロイ] を選択して、コンテナサービス上にデプロイを作成します。

   コンテナサービスのステータスが [デプロイ中] に変わり、デプロイが作成されます。しばらくすると、デプロイのステータスに応じて、コンテナサービスのステータスが以下のいずれかに変わります。

   - デプロイが成功すると、コンテナサービスのステータスが [実行中] に変わり、デプロイのステータスが [アクティブ] に変わります。デプロイメントでパブリックエンドポイントを設定した場合、パブリックエンドポイントとして選択されたコンテナは、コンテナサービスのデフォルトドメインを介して使用できます。
   - デプロイが失敗し、コンテナサービスで以前のデプロイが実行されている場合、コンテナサービスのステータスが [実行中] に変わり、コンテナサービスは、以前のデプロイをアクティブデプロイとして維持します。以前のデプロイがない場合、コンテナサービスのステータスが [準備完了] に変わり、現在アクティブなデプロイはありません。失敗したデプロイのコンテナログを表示して、問題の診断とトラブルシューティングを行います。詳細については、「Amazon Lightsail コンテナサービスのコンテナログの表示」を参照してください。

トピック

Lightsail コンテナサービスの容量を変更する

Amazon Lightsail コンテナサービスの容量は、そのスケールと能力で構成されています。スケールはコンテナーサービス内のコンピューティングノードの数を指定し、能力はサービス内の各ノードのメモリと vCPUs を指定します。スケールは高可用性と大きな容量のために供給するノードの量を基に選択します。

このガイドの手順に従うことで、ダウンタイムを発生させることなくいつでもコンテナサービスの能力とスケールが不足している場合は動的に増やすことができ、過剰になっている場合は減らすことができます。Lightsail は、現在のデプロイメントに合わせて容量の変更を自動的に管理します。

> **Note**
>
> 新しいデプロイを作成すると、コンテナサービスの既存の使用率メトリクスが消え、新しい現在のデプロイのメトリクスだけが表示されます。

コンテナサービスの詳細については、「コンテナサービス」を参照してください。

コンテナサービスの容量を変更する

以下の手順を実行して Lightsail コンテナサービスの容量を変更します。

1. Lightsail コンソールにサインインします。
2. Lightsail コンソールのホームページで、[コンテナ] タブを選択します。
3. 容量を変更するコンテナサービスの名前を選択します。
4. コンテナサービス管理ページで、容量タブを選択します。

   現在のコンテナサービスの能力、スケール、月額料金は容量ページにあります。
5. 能力とスケールを変更するには、容量の変更を選択します。
6. 表示される確認プロンプトで、はい、続行しますを選択して、コンテナサービスの容量が変更されると現在のデプロイが再デプロイされることを確認します。
7. コンテナサービスの新しい能力とスケールを選択します。
8. はい、適用しますを選択して、新しい容量をコンテナサービスに適用します。

   コンテナサービスのステータスが更新中に変わります。数秒後、サービスのステータスが有効に変わり、新しい容量でのオペレーションが開始されます。

Lightsail コンテナサービスのデプロイバージョンの表示と管理

Amazon Lightsail コンテナサービスで作成するすべてのデプロイは、デプロイバージョンとして保存されます。既存のデプロイのパラメーターを変更すると、コンテナがサービスに再デプロイされ、変更されたデプロイは新しいデプロイバージョンとされます。各コンテナサービスの最新の 50 のデプロイバージョンが保存されます。50 のデプロイバージョンのいずれかを使用して、新しいデプロイを同じコンテナに作成できます。このガイドでは、コンテナサービスのデプロイバージョンの表示および管理の方法を説明します。

コンテナサービスの詳細については、「コンテナサービス」を参照してください。

デプロイバージョンのステータス

各デプロイバージョンは、作成後、以下のいずれかのスタータスになります。

- デプロイ (アクティベート中) – デプロイは起動中です。
- アクティブ – デプロイは正常に作成され、コンテナサービスで現在実行されています。コンテナサービスでは、1 回につき 1 つのデプロイのみを実行することができます。
- 非アクティブ – 以前正常に作成されたデプロイは、コンテナ上で実行されなくなりました。
- 失敗 – デプロイで指定された 1 つ以上のコンテナが起動されなかったため、デプロイが失敗しました。

前提条件

開始する前に、Lightsail コンテナサービスを作成する必要があります。詳細については、「コンテナサービスを作成する」を参照してください。

コンテナを設定して起動するデプロイをコンテナサービス内に作成します。詳細については、「Amazon Lightsail コンテナサービスのデプロイの作成と管理」を参照してください。

コンテナサービスのデプロイバージョンを表示する

Lightsail コンテナサービスのデプロイバージョンを表示するには、以下の手順を実行します。

1. Lightsail コンソールにサインインします。
2. Lightsail コンソールのホームページで、[Containers] (コンテナ) タブを選択します。
3. デプロイバージョンを表示するコンテナサービスの名前を選択します。
4. コンテナサービス管理ページで、[デプロイ] タブを選択します。

   デプロイページには、現在のデプロイとデプロイバージョンが一覧表示されます。(存在する場合)
5. コンテナサービスのデプロイバージョンは、デプロイバージョンセクションにリストされています。

   各デプロイには、作成日、ステータス、アクションメニューがあります。
6. デプロイバージョンのアクションメニューで、以下のいずれかのオプションを選択します。

- 新しいデプロイを作成 – 選択したデプロイバージョンから新しいデプロイを作成するには、以下のオプションを選択します。デプロイの作成の詳細については、コンテナサービスのデプロイを作成または変更するを参照してください。

  > **Note**
  >
  > 失敗ステータスがあるバージョンから新しくデプロイを作成する場合、不具合を修正してからデプロイを作成する必要があります。修正されていない場合、デプロイは再び失敗する可能性が高いです。

- 詳細を表示 – このオプションを選択して、選択したデプロイバージョンのコンテナエントリとパブリックエンドポイントパラメーターを表示します。失敗したデプロイを診断する必要がある場合は、デプロイのコンテナログを表示することも可能です。詳細については、「コンテナサービスログを表示する」を参照してください。

Lightsail コンテナのサービスログを表示する

Amazon Lightsail コンテナサービスのデプロイのすべてのコンテナはログを生成します。コンテナログは、コンテナ内で実行されるプロセスの stdout ストリームと stderr ストリームを提供します。コンテナのログに定期的にアクセスして、オペレーションを診断します。最新の 3 日間のログエントリが保存され、最も古いエントリが最新のエントリに置き換えられます。

コンテナログのフィルタ処理

コンテナログには、1 日に数百のエントリを含めることができます。フィルタリングオプションを使用すると、ログウィンドウに表示されるエントリ数が減り、探しているものを見つけやすくなります。コンテナログは、開始日と終了日 (現地時間)、および特定の期間でフィルタリングできます。期間でフィルタリングする場合、指定した期間のログエントリを含めるか除外するかを選択できます。

[include] (含む) または [exclude] (除く) のフィルター用語は、大文字と小文字を区別する完全一致を検索します。たとえば、「HTTP がメッセージに含まれるログイベントのみを含める」と指定した場合、HTTP がメッセージに含まれるログイベントはすべて表示されますが、http がメッセージに含まれないログイベントは表示されません。Error を除くと指定した場合、Error がメッセージに含まれていないすべてのログイベントが表示され、ERROR がメッセージに含まれているログイベントも表示されます。

前提条件

開始する前に、Lightsail コンテナサービスを作成する必要があります。詳細については、「Amazon Lightsail コンテナサービスの作成」を参照してください。

コンテナを設定して起動するデプロイをコンテナサービス内に作成します。詳細については、「Amazon Lightsail コンテナサービスのデプロイの作成と管理」を参照してください。

コンテナのログの表示

Lightsail コンテナ サービスのコンテナログを表示するには、以下の手順を実行します。

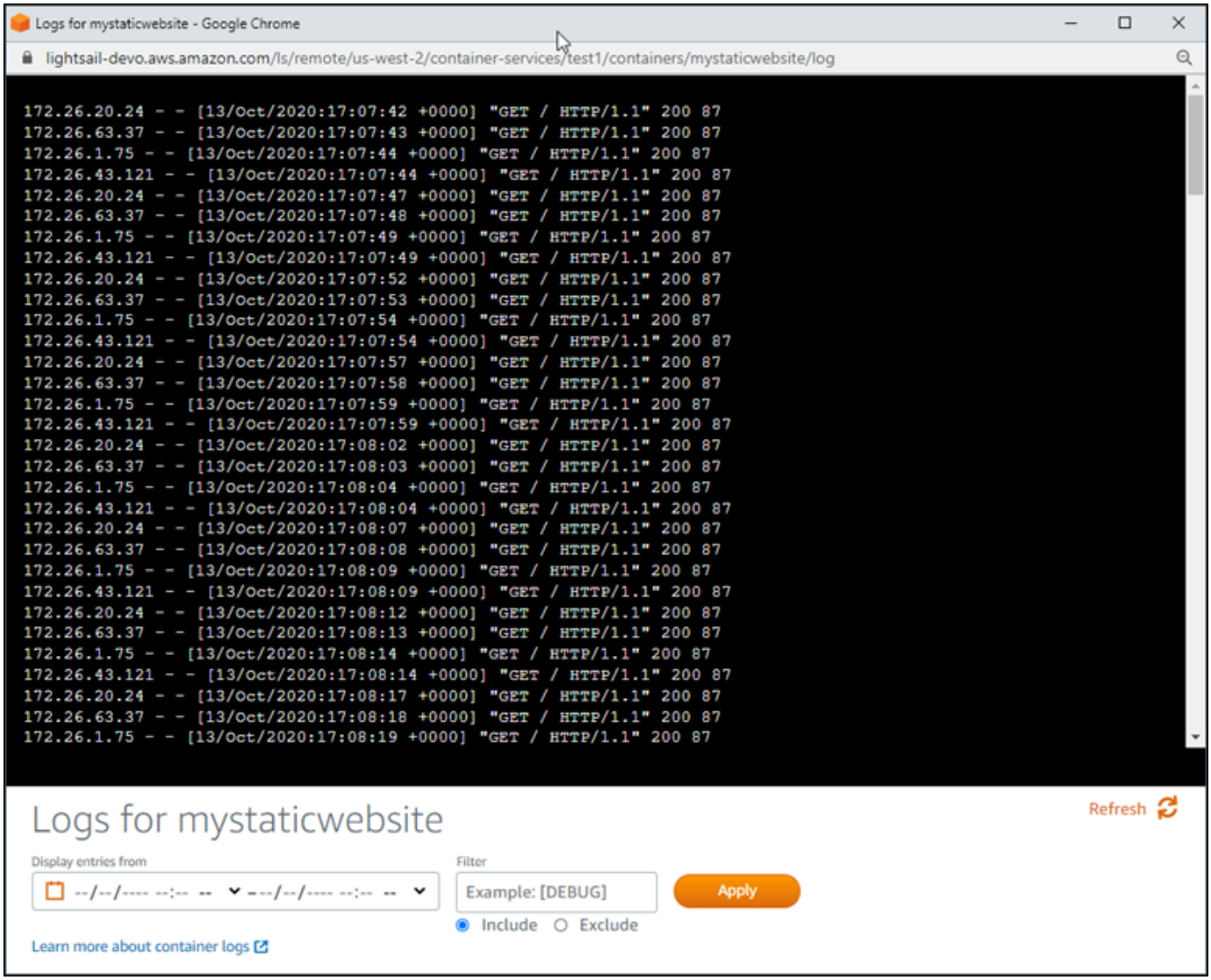
1. Lightsail コンソールにサインインします。
2. Lightsail コンソールのホームページで、[コンテナ] タブを選択します。
3. コンテナログを表示するコンテナサービスの名前を選択します。
4. コンテナサービス管理ページで、[デプロイ] タブを選択します。

   [デプロイ] ページには、現在のデプロイとデプロイのバージョンが一覧表示されます。(存在する場合)
5. 次のいずれかのオプションを選択して、コンテナログを表示します。

   - 現在のデプロイのコンテナログにアクセスするには、ページの[Current deployment] (現在のデプロイ) セクションでコンテナエントリの[ログを開く] を選択します。
   - 以前のデプロイのコンテナログにアクセスするには、以前のデプロイのアクションメニューアイコン (⋮) を選択し、[デプロイバージョン] セクションを選択し、[詳細を表示] を選択します。表示される [バージョンの詳細] ページで、一覧表示されているコンテナエントリの [ログを開く] を選択します。

   ブラウザの新規ウィンドウでコンテナログが開きます。下にスクロールしてさらに多くのログエントリを表示したり、ページを更新して最新のエントリセットをロードしたりできます。フィルタオプションが、ページの下部に表示されます。

> **Note**
>
> ログエントリは昇順で、協定世界時 (UTC) で表示されます。つまり、最も古いログエントリが一番上に表示され、新しいログエントリを表示するには、下にスクロールする必要があります。



Lightsail のカスタムドメインを有効にして管理する

登録済みドメイン名をサービスで使用するためには、Amazon Lightsail コンテナサービスのカスタムドメインを有効化します。カスタムドメインを有効にする前に、コンテナサービスは、最初

の作成時にサービスに関連付けられたデフォルトドメインのトラフィックのみ受け入れます (例: `containerservicename.123456abcdef.us-west-2.cs.amazonlightsail.com`)。カスタムドメインを有効にする場合は、コンテナサービスで使用するドメイン用に作成した Lightsail SSL/TLS 証明書を選択し、その証明書から使用するドメインを選択します。カスタムドメインを有効にすると、コンテナサービスは、選択された証明書に関連付けられているすべてのドメインのトラフィックを受け入れます。

> **Important**
>
> ディストリビューションのオリジンとして Lightsail コンテナサービスを選択すると、Lightsail はディストリビューションのデフォルトドメイン名をカスタムドメインとしてコンテナサービスに自動的に追加します。これにより、ディストリビューションとコンテナサービスの間でトラフィックをルーティングできます。ただし、場合によっては、ディストリビューションのデフォルトドメイン名をコンテナサービスに手動で追加する必要があります。詳細については、「ディストリビューションのデフォルトドメインをコンテナサービスに追加する」を参照してください。

目次

コンテナサービスのカスタムドメインの制限

コンテナサービスのカスタムドメインには、以下の制限が当てはまります。

- Lightsail コンテナサービスそれぞれに最大 4 つのカスタムドメインを使用できます。複数のサービスで同じドメインを使用することはできません。
- Lightsail DNS ゾーンを使用してドメインの DNS を管理する場合、ドメインの apex (例: `example.com`) とサブドメイン (例: `www.example.com`) のトラフィックをコンテナサービスにルーティングできます。

前提条件

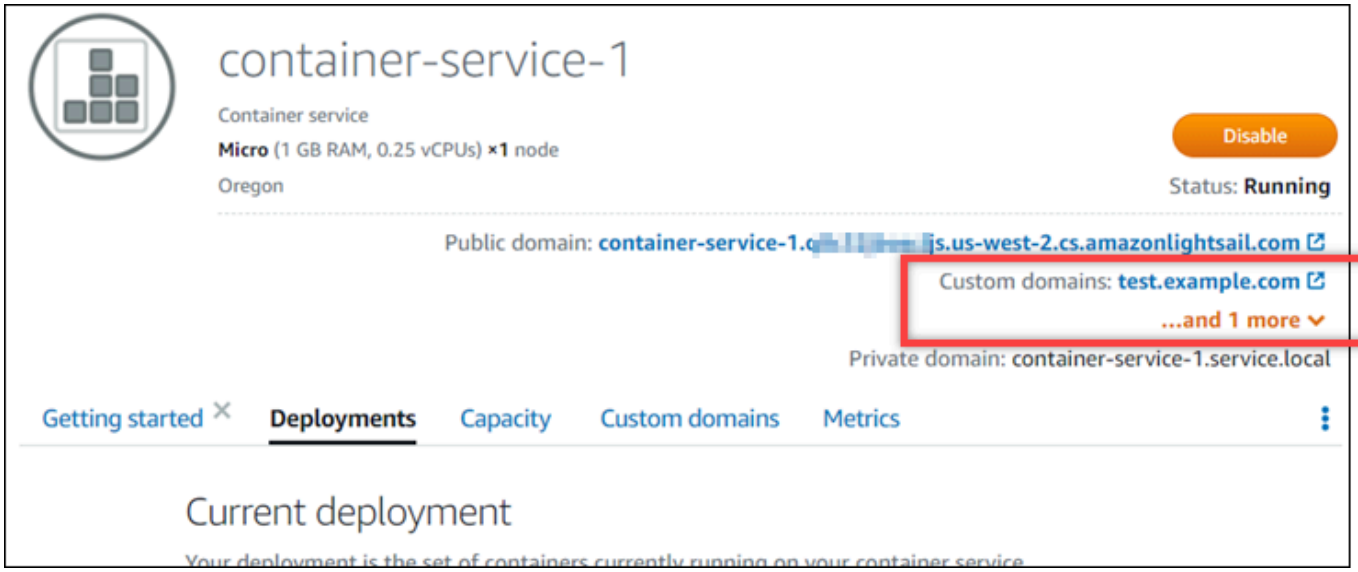
開始する前に、Lightsail コンテナサービスを作成する必要があります。詳細については、「Amazon Lightsail コンテナサービスの作成」を参照してください。

コンテナサービス用の SSL/TLS 証明書が作成され、検証されている必要があります。詳細については、「コンテナサービスの SSL/TLS 証明書を作成する」および「コンテナサービスの SSL/TLS 証明書を検証する」を参照してください。

コンテナサービスのカスタムドメインの表示

コンテナサービスで現在有効になっているカスタムドメインを表示するには、以下の手順を実行します。

1. Lightsail コンソールにサインインします。
2. Lightsail コンソールのホームページで、[Containers] (コンテナ) タブを選択します。
3. 有効にしたカスタムドメインを表示させたいコンテナサービスの名前を選択します。
4. 次の例に示すように、コンテナサービス管理ページの見出しでカスタムドメインの値を見つけます。これらは、コンテナサービスで現在有効になっているカスタムドメインです。



5. コンテナサービス管理ページで、[カスタムドメイン] タブを選択します。

   アタッチされた各証明書で使用されているカスタムドメインは、このページの [Custom domain SSL/TLS certificates] (カスタムドメイン SSL/TLS 証明書) セクションに一覧表示されています。コンテナサービスに現在アタッチされている証明書は、[Attached certificates] (アタッチされた証明書) セクションに一覧表示されています。

コンテナサービスのカスタムドメインを有効にする

Lightsail コンテナサービスのカスタムドメインを有効にするには、以下の手順を実行してサービスに証明書をアタッチします。

1. Lightsail コンソールにサインインします。
2. Lightsail コンソールのホームページで、[Containers] (コンテナ) タブを選択します。
3. カスタムドメインを有効にするコンテナサービスの名前を選択します。
4. コンテナサービス管理ページで [カスタムドメイン] タブを選択します。

   [カスタムドメイン] ページは、コンテナサービスに現在添付されている SSL/TLS 証明書 (存在する場合) を表示します。
5. [証明書のアタッチ] を選択します。

   証明書がない場合は、コンテナサービスにアタッチする前に、ドメインの SSL/TLS 証明書を作成してから検証する必要があります。詳細については、「コンテナサービスの SSL/TLS 証明書を作成する」を参照してください。
6. 表示されるドロップダウンメニューで、コンテナサービスとともに使用するドメインの有効な証明書を選択します。
7. 証明書情報が正しいことを確認し、[Attach] (アタッチ) を選択します。
8. コンテナサービスの [Status] (ステータス) が [Updating] (更新中) に変わります。ステータスが [Ready] (準備完了) に変わると、証明書のドメインが [Custom domains] (カスタムドメイン) セクションに表示されます。
9. [Add domain assignment] (ドメイン割り当ての追加) を選択して、ドメインがコンテナサービスを指すようにします
10. 証明書と DNS 情報が正しいことを確認し、[Add assignment] (割り当てを追加) を選択します。しばらくすると、選択したドメインのトラフィックがコンテナサービスによって受け入れられ始めます。
11. ドメイン割り当てを追加したら、新しいブラウザウィンドウを開き、コンテナサービスに対して有効にしたカスタムドメインを参照します。コンテナサービスで実行されているアプリケーション (存在する場合) がロードされます。

コンテナサービスのカスタムドメインを無効化する

Lightsail コンテナサービスのカスタムドメインを無効にするには、次の手順を実行します。サービスから証明書をデタッチするか、以前に選択したドメインの選択を解除します。

1. Lightsail コンソールにサインインします。
2. Lightsail コンソールのホームページで、[Containers] (コンテナ) タブを選択します。
3. カスタムドメインを無効にするコンテナサービスの名前を選択します。
4. コンテナサービス管理ページで、[カスタムドメイン] タブを選択します。

   [カスタムドメイン] ページは、コンテナサービスに現在添付されている SSL/TLS 証明書 (存在する場合) を表示します。
5. 以下のオプションのいずれかを選択します。

   1. [Configure container service domains] (コンテナサービスドメインを設定する) を選択して、以前に選択したドメインの選択を解除するか、コンテナサービスに関連付けられているドメインをさらに選択します。
   2. [デタッチ] を選択してコンテナサービスから証明書をデタッチし、関連付けられているすべてのドメインをサービスから削除します。

> **Important**
>
> トラフィックルートがコンテナサービスへのルーティングを停止し、代わりに別のリソースにルーティングするように、まだ行っていない場合は、ドメインの DNS レコードを変更します。

トピック

- ドメインへのトラフィックを Lightsail コンテナサービスにルーティングする
- Route 53 のドメインへのトラフィックを Lightsail コンテナサービスにルーティングする

ドメインへのトラフィックを Lightsail コンテナサービスにルーティングする

使用しているサービスでカスタムドメインを有効にした場合は、登録したドメイン名を Amazon Lightsail コンテナサービスに向けてポイントする必要があります。これを行うには、コンテナサービスで使用している証明書に指定されている各ドメインの DNS ゾーンに、エイリアスレコードを追加します。追加するレコードはすべて、コンテナサービスのデフォルトのドメイン (例:`https://<ServiceName>.<RandomGUID>.<AWSRegion>.cs.amazonlightsail.com`) に紐づけする必要があります。

このガイドでは、Lightsail DNS ゾーンを使用してコンテナサービスにドメインを指定する手順について説明します。Lightsail DNS ゾーンの詳細については、「Amazon Lightsail での DNS」を参照してください。

コンテナサービスの詳細については、「コンテナサービス」を参照してください。

> **Note**
>
> Route 53 を使用してドメインの DNS をホストする場合は、Route 53 のドメインのホストゾーンにエイリアスレコードを追加する必要があります。詳細については、「Route 53 のドメインへのトラフィックを Amazon Lightsail コンテナサービスにルーティング」を参照してください。

前提条件

開始する前に、Lightsail コンテナサービスのカスタムドメインを有効にする必要があります。詳細については、Amazon Lightsail コンテナサービスでのカスタムドメインの有効化と管理 を参照してください。

コンテナサービスのデフォルトドメインを取得する

以下の手順を実行して、コンテナサービスのデフォルトのドメイン名を取得します。このドメイン名は、ドメインの DNS にエイリアスレコードを追加するときに指定します。

1. Lightsail コンソールにサインインします。
2. Lightsail コンソールのホームページで、[コンテナ] タブを選択します。
3. デフォルトのドメイン名を取得するコンテナサービスの名前を選択します。

4. コンテナサービス管理ページのヘッダー部分にある、デフォルトのドメイン名を書き留めます。コンテナサービスのデフォルトのドメイン名は、*<ServiceName>*.*<RandomGUID>*.*<AWSRegion>*.cs.amazonlightsail.com と似ています。

   この値は、ドメイン DNS の正規名 (CNAME) レコードの一部として、追加する必要があります。この値はテキストファイルにコピー、ペーストして、後で参照できるようにしておくことをお勧めします。詳細については、このガイドの「ドメインの DNS ゾーンに CNAME レコードを追加する」セクションを参照してください。

ドメインの DNS ゾーンにレコードを追加する

アドレス (IPv4 の場合は A、IPv6 の場合は AAAA) および正規 (CNAME) レコードをドメインの DNS ゾーンに追加するには、次の手順を実行します。

1. Lightsail のホームページで [Domains & DNS] (ドメインと DNS) タブを選択します。
2. ページの [DNS zones] (DNS ゾーン) セクションで、レコードを追加したいドメイン名を選択します。そのレコードがユーザーのドメインへのトラフィックをコンテナサービスに送信します。
3. [DNS records] (DNS レコード) タブを選択します。
4. DNS ゾーンの現在の状態に応じて、次のいずれかのステップを実行します。

   - A、AAAA、ないし CNAME レコードを追加していない場合は、[Add record] (レコードの追加) を選択します。
   - 以前に A、AAAA、または CNAME レコードを追加している場合は、ページに記載されている既存の A、AAAA、ないし CNAME レコードの横にある編集アイコンを選択し、この手順のステップ 5 まで進んでください。

5. [Record type] (レコードタイプ) のドロップダウンメニューにある [A record]、[AAAA record]、ないし [CNAME record] を選択します。

   - A レコードを追加して、ドメインの頂点 (例: `example.com`) をマッピングします。または、IPv4 ネットワーク下のコンテナサービスにサブドメイン (例: `www.example.com`) をマッピングします。
   - AAA レコードを追加して、ドメインの頂点 (例: `example.com`) をマッピングします。または、IPv6 ネットワーク下のコンテナサービスにサブドメイン (例: `www.example.com`) をマッピングします。

- CNAME レコードを追加して、コンテナサービスのパブリックドメイン (デフォルト DNS) にサブドメイン (例: `www.example.com`) をマッピングします。

6. [Record name] (レコード名) テキストボックスに、次のいずれかのオプションを入力します。

   - A レコードないし AAAA レコードの場合は、`@` を入力してドメインの頂点 (例: `example.com`) へのトラフィックをコンテナサービスに送信します。またはサブドメイン (例: `www`) を入力して、サブドメイン (例: `www.example.com`) へのトラフィックをコンテナサービスにルーティングします。
   - CNAME レコードの場合は、サブドメイン (例: `www`) を入力してサブドメイン (例: `www.example.com`)へのトラフィックをコンテナサービスに送信します。

7. 追加するレコードに応じて、次のいずれかの手順を実行します。

   - A レコードまたは AAAA レコードの場合は、[Resolves to] (解決先) テキストボックスにあるコンテナサービス名を選択します。
   - CNAME レコードの場合は、コンテナサービスのデフォルトのドメイン名を [Maps to] (マッピング先) テキストボックスに入力します。

8. 保存アイコンを選択して、レコードを DNS ゾーンに保存します。

   これらの手順を繰り返して、コンテナサービスで使用している証明書が紐づくドメイン用の DNS レコードを追加します。変更がインターネットの DNS を通じて伝達されるまで待ちます。数分後に、ドメインがコンテナサービスを指しているかどうか確認してください。

Route 53 のドメインへのトラフィックを Lightsail コンテナサービスにルーティングする

`example.com` などの登録済みドメインのトラフィックを、Lightsail コンテナサービスで実行されているアプリケーションにルーティングできます。これを行うには、Lightsail コンテナサービスのデフォルトドメインを指すドメインのホストゾーンに、エイリアスレコードを追加します。

このチュートリアルでは、Lightsail コンテナサービスのエイリアスレコードを Route 53 のホストゾーンに追加する方法を説明ます。これは、AWS Command Line Interface (AWS CLI) を使用してのみ実行できます。Route 53 コンソールを使用しても実行されません。

> **Note**
>
> Lightsail を使用してドメインの DNS をホストする場合は、Lightsail のドメインのホストゾーンにエイリアスレコードを追加する必要があります。詳細については、「Amazon

Lightsail のドメインへのトラフィックをLightsail コンテナサービスにルーティング」を参照してください。

目次

ステップ 1: 前提条件を満たす

以下の前提条件を完了します (まだの場合)。

- Route 53 でドメイン名を登録するか、Route 53 を登録された (既存の) ドメイン名の DNS サービスにします。詳細については、「Amazon Route 53 デベロッパーガイド」の「Amazon Route 53 を使用したドメイン名の登録」か「Amazon Route 53 を既存のドメインの DNS サービスにする」を参照してください。
- アプリケーションを Lightsail コンテナサービスにデプロイします。詳細については、「コンテナサービスのデプロイの作成と管理」を参照してください。
- Lightsail コンテナサービスで、登録済みのドメイン名を有効にします。詳細については、「カスタムドメインの有効化と管理」を参照してください。
- アカウントを使用して AWS CLI を設定します。詳細については、「 Lightsail で使用するために AWS CLI を設定する」を参照してください。

ステップ 2: Lightsail コンテナサービスのホストゾーン ID を取得する

Route 53 のホストゾーンにエイリアスレコードを追加するときは、Lightsail コンテナサービスのホストゾーン ID を指定する必要があります。例えば、Lightsail コンテナサービスが米国西部 (オレゴン) (us-west-2) AWS リージョン にある場合、Lightsail コンテナサービスのエイリアスレコードを Route 53 のホストゾーンに追加するなら、ホストゾーン ID `Z0959753D43BBB908BAV` を指定する必要があります。

Lightsail コンテナサービスを作成できる各 AWS リージョンのホストゾーン ID を以下に示します。

欧州 (ロンドン) (eu-west-2): Z0624918ZXDYQZLOXA66

米国東部 (バージニア北部) (us-east-1): Z06246771KYU0IRHI74W4

アジアパシフィック (シンガポール) (ap-southeast-1): Z0625921354DRJH4EY9V0

欧州 (アイルランド) (eu-west-1): Z0624732FELAMMKW3Y21

アジアパシフィック (東京) (ap-northeast-1): Z0626125UAU4JWQ9JSKN

アジアパシフィック (ソウル) (ap-northeast-2): Z06260262XZM84B2WPLHH

アジアパシフィック (ムンバイ) (ap-south-1): Z10460781IQMISS0I0VVY

アジアパシフィック (シドニー) (ap-southeast-2): Z09597943PQQZATPFE96E

カナダ (中部) (ca-central-1): Z10450993RIRIJJUUMA5W

ヨーロッパ (フランクフルト) (eu-Central-1): Z06137433FV04OY4EC6L0

欧州 (ストックホルム) (eu-north-1): Z016970523TDG2TZMUXKK

欧州 (パリ) (eu-west-3): Z09594631DSW2QUR7CFGO

米国東部 (オハイオ) (us-east-2): Z10362273VJ548563IY84

米国西部 (オレゴン) (us-west-2): Z0959753D43BBB908BAV

ステップ 3: レコードセット JSON ファイルを作成する

AWS CLI を使用して、 DNS レコードを Route 53 にあるドメインのホストゾーンに追加する場合、レコードの一連の設定パラメータを指定する必要があります。これを行う最も簡単な方法は、すべてのパラメータを含む JSON (.json) ファイルを作成し、AWS CLI のリクエストで JSON ファイルを参照することです。

次の手順を完了させ、エイリアスレコードのレコードセットパラメータを持つ JSON ファイルを作成します。

1. Windows の場合は Notepad、Linux の場合は Nano などのテキストエディタを開きます。
2. 次のテキストをコピーし、テキストエディターに貼り付けます。

```
{
  "Comment": "Comment",
  "Changes": [
    {
      "Action": "CREATE",
```

```
      "ResourceRecordSet": {
        "Name": "Domain.",
        "Type": "A",
        "AliasTarget": {
          "HostedZoneId": "LightsailContainerServiceHostedZoneID",
          "DNSName": " LightsailContainerServiceAddress.",
          "EvaluateTargetHealth": true
        }
      }
    }
  ]
}
```

ファイルで、次のサンプルテキストを独自のテキストに置き換えます。

- *Comment* を、レコードセットに関する個人的なメモまたはコメントで。
- *Domain* を、Lightsail コンテナサービスで使用したい登録済みのドメイン名で (例: `example.com` または `www.example.com`)。ドメインのルートを Lightsail コンテナサービスで使用する場合は、ドメインのサブドメイン空間で記号 `@` を指定する必要があります (例えば、`@.example.com`)。
- *Lightsail ContainerServiceHostedZoneID* を、Lightsail コンテナサービスを作成した AWS リージョンのホストゾーン ID で置き換えます。詳細については、このガイドの前半にある「ステップ 2: Lightsail コンテナサービスのホストゾーン ID を取得する」を参照してください。
- *LightsailContainerServiceAddress* を、Lightsail コンテナサービスのパブリックドメイン名で置き換えます。これを取得するには、Lightsail コンソールにサインインし、コンテナーサービスを参照し、コンテナーサービス管理ページのヘッダーセクションにリストされているパブリックドメインをコピーします (例えば、`container-service-1.q8cexampleljs.us-west-2.cs.amazonlightsail.com`)。

例:

```
{
  "Comment": "Alias record for Lightsail container service",
  "Changes": [
    {
      "Action": "CREATE",
      "ResourceRecordSet": {
        "Name": "@.example.com.",
```

```
        "Type": "A",
        "AliasTarget": {
          "HostedZoneId": "Z0959753D43BBB908BAV",
          "DNSName": "container-service-1.q8cexampleljs.us-
west-2.cs.amazonlightsail.com.",
          "EvaluateTargetHealth": true
        }
      }
    }
  ]
}
```

3. ローカルディレクトリに change-resource-record-sets.json としてファイルを保存します。

ステップ 4: Route 53 で、ドメインのホストゾーンにレコードを追加する

次の手順を完了させ、AWS CLI を使用して、Route 53 のドメインのホストゾーンにレコードを追加します。これは、change-resource-record-sets コマンドを使用して行います。詳細については、「AWS CLI コマンドリファレンス」の「change-resource-record-sets」を参照してください。

> **Note**
>
> この手順を続行する前に、AWS CLI をインストールし、Lightsail と Route 53 用に設定する必要があります。詳細については、「Lightsail で使用するために AWS CLI を設定する」を参照してください。

1. ターミナルまたはコマンドプロントウィンドウを開きます。
2. 次のコマンドを入力して、Route 53 のドメインのホストゾーンにレコードを追加します。

```
aws route53 change-resource-record-sets --hosted-zone-id HostedZoneID --change-
batch PathToJsonFile
```

コマンドで、次のサンプルテキストを独自のテキストに置き換えます。

- *HostedZoneId* を、Route 53 で登録済みドメインのホストゾーンの ID で置き換えます。list-hosted-zones コマンドを使用して、Route 53 アカウントのホストゾーンの ID のリストを取得します。

- *PathToJsonFile* を、レコードパラメータを含む .json ファイルのコンピュータ上のローカルディレクトリフォルダパスで。詳細については、このガイドの前半にある「ステップ 3: レコードセット JSON ファイルを作成する」セクションを参照してください。

例:

Linux または Unix コンピュータの場合は、次の操作を行います。

```
aws route53 change-resource-record-sets --hosted-zone-id Z123456789ABCDEFGHIJ --
change-batch home/user/awscli/route53/change-resource-record-sets.json
```

Windows コンピュータの場合:

```
aws route53 change-resource-record-sets --hosted-zone-id Z123456789ABCDEFGHIJ --
change-batch file://C:\awscli\route53\change-resource-record-sets.json
```

以下の例のような結果が表示されるはずです。

```
H:\>aws route53 change-resource-record-sets --hosted-zone-id Z123456789ABCDEFGHIJ
--change-batch file://C:\awscli\route53\change-resource-record-sets.json
{
    "ChangeInfo": {
        "Id": "/change/C05953EXAMPLEZ4V4LOAC",
        "Status": "PENDING",
        "SubmittedAt": "2021-08-11T20:58:30.960000+00:00",
        "Comment": "Alias record for Lightsail container service"
    }
}
```

変更がインターネットの DNS を通じて伝播されるまで数時間かかる場合があります。完了すると、Route 53 の登録ドメインのインターネットトラフィックが Lightsail コンテナサービスへルーティング開始されるはずです。

Amazon Lightsail のセキュリティ

AWS では、クラウドのセキュリティが最優先事項です。AWS の顧客は、セキュリティを最も重視する組織の要件を満たすように構築されたデータセンターとネットワークアーキテクチャから利点を得られます。

セキュリティは、AWS と顧客の間の責任共有です。責任共有モデルでは、この責任がクラウドのセキュリティおよびクラウド内のセキュリティとして説明されています。

- クラウドのセキュリティ - AWS は、AWS クラウドで AWS のサービスを実行するインフラストラクチャを保護する責任を負います。また、AWS は、使用するサービスを安全に提供します。コンプライアンスプログラム、およびそれらが適用されるサービスについては、「コンプライアンスプログラムによる AWS 対象範囲内のサービス」を参照してください。
- クラウド内のセキュリティ - ユーザーの責任は、使用する AWS のサービスに応じて異なります。またお客様は、データの機密性、企業要件、適用法令と規制などのその他の要因に対しても責任を担います。

このドキュメントは、Amazon Lightsail を使用する際に責任共有モデルを適用する方法を理解するのに役立ちます。次のトピックでは、セキュリティおよびコンプライアンスの目的を達成するために Amazon Lightsail を設定する方法を示します。また、Amazon Lightsail リソースのモニタリングや保護に役立つ他の AWS サービスの使用方法についても説明します。

Amazon Lightsail でのインフラストラクチャセキュリティ

マネージドサービスである Amazon Lightsail は AWS グローバルネットワークセキュリティで保護されています。AWSセキュリティサービスと AWS がインフラストラクチャを保護する方法については、「AWS クラウドセキュリティ」を参照してください。インフラストラクチャセキュリティのベストプラクティスを使用して AWS 環境を設計するには、「セキュリティの柱 - AWS Well-Architected Framework」の「インフラストラクチャ保護」を参照してください。

AWS の発行済み API コールを使用して、ネットワーク経由で Lightsail にアクセスします。クライアントは以下をサポートする必要があります。

- Transport Layer Security (TLS) TLS 1.2 および TLS 1.3 をお勧めします。
- DHE (Ephemeral Diffie-Hellman) や ECDHE (Elliptic Curve Ephemeral Diffie-Hellman) などの Perfect Forward Secrecy (PFS) を使用した暗号スイートです。これらのモードは、Java 7 以降など、最近のほとんどのシステムでサポートされています。

また、リクエストは、アクセスキー ID と、IAM プリンシパルに関連付けられているシークレットアクセスキーを使用して署名する必要があります。または、AWS Security Token Service (AWS STS) を使用して、一時的なセキュリティ認証情報を生成し、リクエストに署名することもできます。

Amazon Lightsail での耐障害性

AWS グローバルインフラストラクチャは AWS リージョン およびアベイラビリティーゾーンを中心に構築されています。AWS リージョン には、低レイテンシー、高いスループット、そして高度の冗長ネットワークで接続されている複数の物理的に独立・隔離されたアベイラビリティーゾーンがあります。アベイラビリティーゾーンでは、ゾーン間で中断することなく自動的にフェイルオーバーするアプリケーションとデータベースを設計および運用することができます。アベイラビリティーゾーンは、従来の単一または複数のデータセンターインフラストラクチャよりも可用性が高く、フォールトトレラントで、スケーラブルです。

AWS リージョン とアベイラビリティーゾーンの詳細については、「AWS グローバルインフラストラクチャ」を参照してください。

AWS では、Amazon Lightsail グローバルインフラストラクチャに加えて、データの耐障害性とバックアップのニーズに対応できるように複数の機能を提供しています。

- リージョン間でインスタンスとディスクのスナップショットをコピーする。詳細については、「スナップショット」を参照してください。
- インスタンスおよびディスクのスナップショットを自動化します。詳細については、「スナップショット」を参照してください。
- ロードバランサーを使用して、単一のアベイラビリティーゾーンまたは複数のアベイラビリティーゾーンにある複数のインスタンスの間で受信トラフィックを分散する。詳細については、「ロードバランサー」を参照してください。

Amazon Lightsail のためのアイデンティティおよびアクセス管理

対象者

AWS Identity and Access Management (IAM) の使い方は、Amazon Lightsail での作業によって異なります。

サービスユーザー - Amazon Lightsail サービスを使用してジョブを実行する場合は、必要な権限と認証情報を管理者が用意します。作業を実行するためにさらに多くの Amazon Lightsail 機能を使用するとき、追加の権限が必要になる場合があります。アクセスの管理方法を理解すると、管理者から適

切な権限をリクエストするのに役に立ちます。Amazon Lightsail の機能にアクセスできない場合は、「Identity and Access Management (IAM)のトラブルシューティング」を参照してください。

サービス管理者 - 社内の Amazon Lightsail リソースを担当している場合は、通常、Amazon Lightsail への完全なアクセスがあります。従業員がどの Amazon Lightsail特徴とリソースアクセスする必要があるかを決定するのは管理者のジョブです。その後、IAM 管理者にリクエストを送信して、サービスユーザーの権限を変更する必要があります。このページの情報を点検して、IAM の基本概念を理解してください。会社で Amazon Lightsail ともに IAM を使用する方法の詳細については、「Amazon Lightsail と IAM を連携する方法」を参照してください。

IAM 管理者 - 管理者は、Amazon Lightsail へのアクセスを管理するポリシーの書き込み方法の詳細について確認する場合があります。IAM で使用できる Amazon Lightsail アイデンティティベースのポリシーの例については、「Amazon Lightsail アイデンティティベースのポリシーの例」を参照してください。

アイデンティティを使用した認証

認証は、アイデンティティ認証情報を使用して AWS にサインインする方法です。AWS Management Console によるサインインの詳細については、「IAM User Guide」(IAM ユーザーガイド) の「The IAM Console and Sign-in Page」(IAM コンソールとサインインページ) を参照してください。

AWS アカウント のルートユーザーもしくは IAM ユーザーとして、または IAM ロールを引き受けることによって認証される (AWS にサインインする) 必要があります。会社のシングルサインオン認証を使用することも、Google や Facebook を使用してサインインすることもできます。このような場合、管理者は以前に IAM ロールを使用して ID フェデレーションを設定しました。他の会社の認証情報を使用して AWS にアクセスした場合、ロールは間接的に割り当てられています。

AWS Management Console に直接サインインするには、パスワード、およびルートユーザーの E メールまたは IAM ユーザー名を使用します。ルートユーザーまたは IAM ユーザーのアクセスキーを使用して AWS にプログラム的にアクセスできます。AWS は、ユーザーの認証情報を使用してリクエストに暗号署名するための SDK とコマンドラインツールを提供します。AWS ツールを使用しない場合は、リクエストに自分で署名する必要があります。これには、インバウンド API リクエストを認証するためのプロトコルである署名バージョン 4 を使用します。リクエストの認証の詳細については、『』の「署名バージョン 4 の署名プロセスAWS 全般のリファレンス」を参照してください。

使用する認証方法を問わず、追加のセキュリティ情報の提供を要求される場合もあります。例えば、AWS は、アカウントのセキュリティを強化するために多要素認証 (MFA) を使用することをお勧

めします。詳細については、「IAM ユーザーガイド」の「AWS での多要素認証 (MFA) の使用」を参照してください。

AWS アカウントAWS アカウントのルートユーザー

AWS アカウントを作成する場合は、そのアカウントのすべての AWS のサービスとリソースに対して完全なアクセス権を持つ 1 つのサインインアイデンティティから始めます。このアイデンティティは AWS アカウントのルートユーザーと呼ばれ、アカウントの作成に使用した E メールアドレスとパスワードでサインインすることによってアクセスできます。日常的なタスクには、ルートユーザーを使用しないことを強くお勧めします。ルートユーザーの認証情報は保護し、ルートユーザーでしか実行できないタスクを実行するときに使用します。ルートユーザーとしてサインインする必要があるタスクの完全なリストについては、「IAM ユーザーガイド」の「ルートユーザー認証情報が必要なタスク」を参照してください。

IAM ユーザーとグループ

IAM ユーザーは、1 人のユーザーまたは 1 つのアプリケーションに対して特定の権限を持つ AWS アカウント 内のアイデンティティです。可能であれば、パスワードやアクセスキーなどの長期的な認証情報を保有する IAM ユーザーを作成する代わりに、一時的な認証情報を使用することをお勧めします。ただし、IAM ユーザーでの長期的な認証情報が必要な特定のユースケースがある場合は、アクセスキーをローテーションすることをお勧めします。詳細については、「IAM ユーザーガイド」の「長期的な認証情報を必要とするユースケースのためにアクセスキーを定期的にローテーションする」を参照してください。

IAM グループは、IAM ユーザーの集団を指定するアイデンティティです。グループとしてサインインすることはできません。グループを使用して、複数のユーザーに対して一度に権限を指定できます。多数のユーザーグループがある場合、グループを使用することで権限の管理が容易になります。例えば、IAMAdmins という名前のグループを設定して、そのグループに IAM リソースを管理する権限を与えることができます。

ユーザーは、ロールとは異なります。ユーザーは 1 人の人または 1 つのアプリケーションに一意に関連付けられますが、ロールはそれを必要とする任意の人が引き受けるようになっています。ユーザーには永続的な長期の認証情報がありますが、ロールでは一時的な認証情報が提供されます。詳細については、「IAM ユーザーガイド」の「IAM ユーザーの作成が適している場合 (ロールではなく)」を参照してください。

IAM ロール

IAM ロールは、特定の権限を持つ、AWS アカウント 内のアイデンティティです。これは IAM ユーザーに似ていますが、特定のユーザーには関連付けられていません。ロールを切り替えることによっ

て、AWS Management Console で IAM ロールを一時的に引き受けることができます。ロールを引き受けるには、AWS CLI または AWS API オペレーションを呼び出すか、カスタム URL を使用します。ロールを使用する方法の詳細については、「IAM ユーザーガイド」の「IAM ロールを使用する」を参照してください。

一時的な認証情報を持った IAM ロールは、以下の状況で役立ちます。

- フェデレーティッドユーザーアクセス – フェデレーティッドアイデンティティに権限を割り当てるには、ロールを作成してそのロールの権限を定義します。フェデレーティッドアイデンティティが認証されると、そのアイデンティティはロールに関連付けられ、ロールで定義されている権限が付与されます。フェデレーションの詳細については、「IAM ユーザーガイド」の「サードパーティーアイデンティティプロバイダー向けロールの作成」を参照してください。IAM アイデンティティセンターを使用する場合、権限セットを設定します。アイデンティティが認証後にアクセスできるものを制御するため、IAM Identity Center は、権限セットを IAM のロールに関連付けます。権限セットの詳細については、「AWS IAM Identity Center ユーザーガイド」の「権限セット」を参照してください。
- 一時的な IAM ユーザー権限 - IAM ユーザーまたはロールは、特定のタスクに対して複数の異なる権限を一時的に IAM ロールで引き受けることができます。
- クロスアカウントアクセス - IAM ロールを使用して、自分のアカウントのリソースにアクセスすることを、別のアカウントの人物 (信頼済みプリンシパル) に許可できます。クロスアカウントアクセス権を付与する主な方法は、ロールを使用することです。ただし、一部の AWS のサービスでは、(ロールをプロキシとして使用する代わりに) リソースにポリシーを直接アタッチできます。クロスアカウントアクセスにおけるロールとリソースベースのポリシーの違いについては、『IAM ユーザーガイド』の「IAM ロールとリソースベースのポリシーとの相違点」を参照してください。
- クロスサービスアクセス - 一部の AWS のサービスでは、他の AWS のサービスの機能を使用します。例えば、あるサービスで呼び出しを行うと、通常そのサービスによって Amazon EC2 でアプリケーションが実行されたり、Amazon S3 にオブジェクトが保存されたりします。サービスでは、呼び出し元プリンシパルの権限、サービスロール、またはサービスリンクロールを使用してこれを行う場合があります。
  - 転送アクセスセッション (FAS、Forward Access Session) – IAM ユーザーまたはロールを使用して AWS でアクションを実行するユーザーは、プリンシパルと見なされます。一部のサービスを使用する際に、あるアクションを実行することで、別のサービスの別のアクションが開始されることがあります。FAS は、AWS のサービスを呼び出すプリンシパルのアクセス許可を使用し、リクエスト元の AWS のサービスと組み合わせて、ダウンストリームサービスにリクエストを行います。FAS リクエストは、完了するために他の AWS のサービスまたはリソースとのやり取

りを必要とするリクエストをサービスが受信した場合にのみ作成されます。この場合、両方のアクションを実行するための権限が必要です。FAS リクエストを行う際のポリシーの詳細については、「転送アクセスセッション」を参照してください。

- サービスロール - サービスがユーザーに代わってアクションを実行するために引き受ける IAM ロールです。IAM 管理者は、IAM 内からサービスロールを作成、変更、削除できます。詳細については、「IAM ユーザーガイド」の「AWS のサービス に権限を委任するロールの作成」を参照してください。
- サービスリンクロール - サービスリンクロールは、AWS のサービス にリンクされたサービスロールの一種です。サービスがロールを引き受け、ユーザーに代わってアクションを実行できるようになります。サービスリンクロールは、AWS アカウント に表示され、サービスによって所有されます。IAM 管理者は、サービスリンクロールの権限を表示できますが、編集することはできません。

- Amazon EC2 で実行されているアプリケーション - EC2 インスタンスで実行され、AWS CLI または AWS API 要求を行っているアプリケーションの一時的な認証情報を管理するには、IAM ロールを使用できます。これは、EC2 インスタンス内でのアクセスキーの保存に推奨されます。AWS ロールを EC2 インスタンスに割り当て、そのすべてのアプリケーションで使用できるようにするには、インスタンスに添付されたインスタンスプロファイルを作成します。インスタンスプロファイルにはロールが含まれ、EC2 インスタンスで実行されるプログラムは一時的な認証情報を取得できます。詳細については、「IAM ユーザーガイド」の「Amazon EC2 インスタンスで実行されるアプリケーションに IAM ロールを使用して権限を付与する」を参照してください。

IAM ロールと IAM ユーザーのどちらを使用するかについては、「IAM ユーザーガイド」の「IAM ロールの作成が適している場合 (ユーザーではなく)」を参照してください。

一時的な認証情報を持った IAM ロールは、以下の状況で役立ちます。

- 一時的な IAM ユーザー許可 - IAM ユーザーは、特定のタスクに対して複数の異なる許可を一時的に IAM ロールで引き受けることができます。
- フェデレーティッドユーザーアクセス – フェデレーティッドアイデンティティに権限を割り当てるには、ロールを作成してそのロールの権限を定義します。フェデレーティッドアイデンティティが認証されると、そのアイデンティティはロールに関連付けられ、ロールで定義されている権限が付与されます。フェデレーションの詳細については、「IAM ユーザーガイド」の「サードパーティーアイデンティティプロバイダー向けロールの作成」を参照してください。IAM アイデンティティセンターを使用する場合、権限セットを設定します。アイデンティティが認証後にアクセスできるものを制御するため、IAM Identity Center は、権限セットを IAM のロールに関連付け

ます。権限セットの詳細については、「AWS IAM Identity Center ユーザーガイド」の「権限セット」を参照してください。

- クロスアカウントアクセス - IAM ロールを使用して、自分のアカウントのリソースにアクセスすることを、別のアカウントの人物 (信頼済みプリンシパル) に許可できます。クロスアカウントアクセス権を付与する主な方法は、ロールを使用することです。ただし、一部の AWS のサービスでは、(ロールをプロキシとして使用する代わりに) リソースにポリシーを直接アタッチできます。クロスアカウントアクセスにおけるロールとリソースベースのポリシーの違いについては、『IAM ユーザーガイド』の「IAM ロールとリソースベースのポリシーとの相違点」を参照してください。
- クロスサービスアクセス - 一部の AWS のサービスでは、他の AWS のサービスの機能を使用します。例えば、あるサービスで呼び出しを行うと、通常そのサービスによって Amazon EC2 でアプリケーションが実行されたり、Amazon S3 にオブジェクトが保存されたりします。サービスでは、呼び出し元プリンシパルの権限、サービスロール、またはサービスリンクロールを使用してこれを行う場合があります。
  - プリンシパル権限 - IAM ユーザーまたはロールを使用して AWS でアクションを実行する場合、そのユーザーはプリンシパルと見なされます。ポリシーによって、プリンシパルに権限が付与されます。一部のサービスを使用する際に、アクションを実行することで別のサービスの別のアクションがトリガーされることがあります。この場合、両方のアクションを実行するための権限が必要です。アクションがポリシーで追加の依存アクションを必要とするかどうかを確認するには、「サービス認可リファレンス」の「Amazon Lightsail のアクション、リソース、および条件キー」をご参照ください。
  - サービスロール - サービスがユーザーに代わってアクションを実行するために引き受ける IAM ロールです。IAM 管理者は、IAM 内からサービスロールを作成、変更、削除できます。詳細については、「IAM ユーザーガイド」の「AWS のサービス に権限を委任するロールの作成」を参照してください。
  - サービスリンクロール - サービスリンクロールは、AWS のサービス にリンクされたサービスロールの一種です。サービスがロールを引き受け、ユーザーに代わってアクションを実行できるようになります。サービスリンクロールは、AWS アカウント に表示され、サービスによって所有されます。IAM 管理者は、サービスリンクロールの権限を表示できますが、編集することはできません。
- Amazon EC2 で実行されているアプリケーション - EC2 インスタンスで実行され、AWS CLI または AWS API 要求を行っているアプリケーションの一時的な認証情報を管理するには、IAM ロールを使用できます。これは、EC2 インスタンス内でのアクセスキーの保存に推奨されます。AWS ロールを EC2 インスタンスに割り当て、そのすべてのアプリケーションで使用できるようにするには、インスタンスに添付されたインスタンスプロファイルを作成します。インスタンスプロファ

イルにはロールが含まれ、EC2 インスタンスで実行されるプログラムは一時的な認証情報を取得できます。詳細については、「IAM ユーザーガイド」の「Amazon EC2 インスタンスで実行されるアプリケーションに IAM ロールを使用して権限を付与する」を参照してください。

IAM ロールと IAM ユーザーのどちらを使用するかについては、「IAM ユーザーガイド」の「IAM ロールの作成が適している場合 (ユーザーではなく)」を参照してください。

ポリシーを使用したアクセスの管理

AWS でアクセス権を管理するには、ポリシーを作成して AWS アイデンティティまたはリソースにアタッチします。ポリシーは AWS のオブジェクトであり、アイデンティティやリソースに関連付けて、これらの権限を定義します。AWS は、プリンシパル (ユーザー、ルートユーザー、またはロールセッション) がリクエストを行うと、これらのポリシーを評価します。ポリシーでの権限により、リクエストが許可されるか拒否されるかが決まります。大半のポリシーは JSON ドキュメントとして AWSに保存されます。JSON ポリシードキュメントの構造と内容の詳細については、『IAM ユーザーガイド』の「JSON ポリシー概要」を参照してください。

管理者は AWSJSON ポリシーを使用して、だれが何にアクセスできるかを指定できます。つまり、どのプリンシパルがどのリソースに対してどのような条件下でアクションを実行できるかということです。

デフォルトでは、ユーザーやロールに権限はありません。IAM 管理者は、リソースで必要なアクションを実行するための権限をユーザーに付与する IAM ポリシーを作成できます。その後、管理者はロールに IAM ポリシーを追加し、ユーザーはロールを引き継ぐことができます。

IAM ポリシーは、オペレーションの実行方法を問わず、アクションの権限を定義します。例えば、`iam:GetRole` アクションを許可するポリシーがあるとします。このポリシーがあるユーザーは、AWS Management Console、AWS CLI、または AWS API からロール情報を取得できます。

管理者は AWS JSON ポリシーを使用して、誰が何にアクセスできるかを指定できます。つまり、どのプリンシパルがどのリソースに対してどのような条件下でアクションを実行できるかということです。

すべての IAM エンティティ (ユーザーまたはロール) は、許可のない状態からスタートします。言い換えると、デフォルト設定では、ユーザーは何もできず、自分のパスワードを変更することすらできません。何かを実行する許可をユーザーに付与するには、管理者がユーザーに許可ポリシーをアタッチする必要があります。また、管理者は、必要な許可があるグループにユーザーを追加できます。管理者がグループに許可を付与すると、そのグループ内のすべてのユーザーにこれらの許可が付与されます。

IAM ポリシーは、オペレーションの実行方法を問わず、アクションの許可を定義します。例えば、`iam:GetRole` アクションを許可するポリシーがあるとします。このポリシーがあるユーザーは、AWS Management Console、AWS CLI、または AWS API からロール情報を取得できます。

アイデンティティベースのポリシー

アイデンティティベースポリシーは、IAM ユーザー、ユーザーのグループ、ロールなど、アイデンティティにアタッチできる JSON 権限ポリシードキュメントです。これらのポリシーは、ユーザーとロールが実行できるアクション、リソース、および条件を制御します。アイデンティティベースのポリシーを作成する方法については、『IAM ユーザーガイド』の「IAM ポリシーの作成」を参照してください。

アイデンティティベースポリシーは、さらにインラインポリシーまたはマネージドポリシーに分類できます。インラインポリシーは、単一のユーザー、グループ、またはロールに直接埋め込まれます。管理ポリシーは、AWS アカウント 内の複数のユーザー、グループ、およびロールにアタッチできるスタンドアロンポリシーです。マネージドポリシーには、AWS マネージドポリシーとカスタマー管理ポリシーがあります。マネージドポリシーまたはインラインポリシーのいずれかを選択する方法については、『IAM ユーザーガイド』の「マネージドポリシーとインラインポリシーの比較」を参照してください。

アイデンティティベースポリシーは、IAM ユーザー、ユーザーグループ、ロールなど、アイデンティティにアタッチできる JSON 権限ポリシードキュメントです。これらのポリシーは、ユーザーとロールが実行できるアクション、リソース、および条件を制御します。アイデンティティベースのポリシーを作成する方法については、『IAM ユーザーガイド』の「IAM ポリシーの作成」を参照してください。

リソースベースのポリシー

リソースベースのポリシーは、リソースに添付する JSON ポリシードキュメントです。リソースベースのポリシーの例には、IAM ロールの信頼ポリシーや Amazon S3 バケットポリシーがあります。リソースベースのポリシーをサポートするサービスでは、サービス管理者はポリシーを使用して特定のリソースへのアクセスを制御できます。ポリシーがアタッチされているリソースの場合、指定されたプリンシパルがそのリソースに対して実行できるアクションと条件は、ポリシーによって定義されます。リソースベースのポリシーでは、プリンシパルを指定する必要があります。プリンシパルには、アカウント、ユーザー、ロール、フェデレーションユーザー、または AWS のサービス を含めることができます。

リソースベースのポリシーは、そのサービス内にあるインラインポリシーです。リソースベースのポリシーでは IAM の AWSマネージドポリシーは使用できません。

リソースベースのポリシーは、リソースに添付する JSON ポリシードキュメントです。リソースベースのポリシーの例には、IAM ロールの信頼ポリシーや Amazon S3 バケットポリシーがあります。リソースベースのポリシーをサポートするサービスでは、サービス管理者はポリシーを使用して特定のリソースへのアクセスを制御できます。ポリシーがアタッチされているリソースの場合、指定されたプリンシパルがそのリソースに対して実行できるアクションと条件は、ポリシーによって定義されます。リソースベースのポリシーでは、プリンシパルを指定する必要があります。プリンシパルには、アカウント、ユーザー、ロール、フェデレーションユーザー、または AWS のサービス を含めることができます。

アクセスコントロールリスト (ACL)

アクセスコントロールリスト (ACL) は、どのプリンシパル (アカウントメンバー、ユーザー、またはロール) がリソースにアクセスするための権限を持つかをコントロールします。ACL はリソースベースのポリシーに似ていますが、JSON ポリシードキュメント形式は使用しません。

Simple Storage Service (Amazon S3)、AWS WAF、および Amazon VPC は、ACL をサポートするサービスの例です。ACL の詳細については、「Amazon Simple Storage Service 開発者ガイド」の「アクセスコントロールリスト (ACL) の概要」を参照してください。

アクセスコントロールリスト (ACL) は、どのプリンシパル (アカウントメンバー、ユーザー、またはロール) がリソースにアクセスするための権限を持つかをコントロールします。ACL はリソースベースのポリシーに似ていますが、JSON ポリシードキュメント形式は使用しません。

その他のポリシータイプ

AWS では、他の一般的ではないポリシータイプをサポートしています。これらのポリシータイプでは、より一般的なポリシータイプで付与された最大の権限を設定できます。

- 権限の境界 - 権限の境界は、アイデンティティベースのポリシーによって IAM エンティティ (IAM ユーザーまたはロール) に付与できる権限の上限を設定する高度な機能です。エンティティに権限の境界を設定できます。結果として得られる権限は、エンティティのアイデンティティベースポリシーとその権限の境界の共通部分になります。`Principal` フィールドでユーザーまたはロールを指定するリソースベースのポリシーでは、権限の境界は制限されません。これらのポリシーのいずれかを明示的に拒否した場合、権限は無効になります。権限の境界の詳細については、「IAM ユーザーガイド」の「IAM エンティティの権限の境界」を参照してください。
- サービスコントロールポリシー (SCP) - SCP は、AWS Organizations で組織や組織単位 (OU) の最大権限を指定する JSON ポリシーです。AWS Organizations は、顧客のビジネスが所有する複数の AWS アカウント をグループ化し、一元的に管理するサービスです。組織内のすべての機能を有効にすると、サービスコントロールポリシー (SCP) を一部またはすべてのアカウントに適用

できます。SCP はメンバーアカウントのエンティティに対する権限を制限します (各 AWS アカウントのルートユーザー など)。組織 と SCP の詳細については、AWS Organizations ユーザーガイドの「SCP の仕組み」を参照してください。

- セッションポリシー - セッションポリシーは、ロールまたはフェデレーションユーザーの一時的なセッションをプログラムで作成する際にパラメータとして渡す高度なポリシーです。結果としてセッションの権限の範囲は、ユーザーまたはロールのアイデンティティベースポリシーとセッションポリシーの共通部分になります。また、リソースベースのポリシーから権限が派生する場合もあります。これらのポリシーのいずれかを明示的に拒否した場合、権限は無効になります。詳細については、「IAM ユーザーガイド」の「セッションポリシー」を参照してください。

- 権限の境界 - 権限の境界は、アイデンティティベースのポリシーによって IAM エンティティ (IAM ユーザーまたはロール) に付与できる権限の上限を設定する高度な機能です。エンティティに許可の境界を設定できます。結果として許可される範囲は、エンティティのアイデンティティベースポリシーとその許可の境界の共通部分になります。`Principal` フィールドでユーザーまたはロールを指定するリソースベースのポリシーでは、許可の境界は制限されません。これらのポリシーのいずれかを明示的に拒否した場合、権限は無効になります。権限の境界の詳細については、「IAM ユーザーガイド」の「IAM エンティティの権限の境界」を参照してください。
- サービスコントロールポリシー (SCP) - SCP は、AWS Organizations で組織や組織単位 (OU) の最大権限を指定する JSON ポリシーです。AWS Organizations は、顧客のビジネスが所有する複数の AWS アカウント をグループ化し、一元的に管理するサービスです。組織内のすべての機能を有効にすると、サービスコントロールポリシー (SCP) を一部またはすべてのアカウントに適用できます。SCP ではメンバーアカウントのエンティティ (各 AWS アカウント ルートユーザーなど) に対する許可が制限されます。Organizations と SCP の詳細については、AWS Organizations ユーザーガイドの「SCP の仕組み」を参照してください。
- セッションポリシー - セッションポリシーは、ロールまたはフェデレーションユーザーの一時的なセッションをプログラムで作成する際にパラメータとして渡す高度なポリシーです。結果としてセッションの権限の範囲は、ユーザーまたはロールのアイデンティティベースポリシーとセッションポリシーの共通部分になります。また、リソースベースのポリシーから権限が派生する場合もあります。これらのポリシーのいずれかを明示的に拒否した場合、権限は無効になります。詳細については、「IAM ユーザーガイド」の「セッションポリシー」を参照してください。

複数のポリシータイプ

1 つのリクエストに複数のタイプのポリシーが適用されると、結果として作成される権限を理解するのがさらに難しくなります。複数のポリシータイプが関連するとき、リクエストを許可するかどうか

を AWS が決定する方法の詳細については、IAM ユーザーガイドのポリシーの評価ロジックを参照してください。

トピック

- AWS の Amazon Lightsail 管理ポリシー
- Amazon Lightsail と IAM の連携について
- IAM ユーザーの Amazon Lightsail へのアクセスを管理します。

AWS の Amazon Lightsail 管理ポリシー

ユーザー、グループ、ロールに許可を追加するには、自分でポリシーを作成するよりも、AWS マネージドポリシーを使用する方が簡単です。チームに必要な許可のみを提供する IAM カスタマーマネージドポリシーを作成するには、時間と専門知識が必要です。すぐに使用を開始するために、AWS マネージドポリシーを使用できます。これらのポリシーは、一般的なユースケースをターゲット範囲に含めており、AWS アカウント で利用できます。AWS マネージドポリシーの詳細については、「IAM ユーザーガイド」の「AWS マネージドポリシー」を参照してください。

AWS のサービスは、AWS マネージドポリシーを維持および更新します。AWS マネージドポリシーの許可を変更することはできません。サービスでは、新しい機能を利用できるようにするために、AWS マネージドポリシーに許可が追加されることがあります。この種類の更新は、ポリシーがアタッチされている、すべてのアイデンティティ (ユーザー、グループおよびロール) に影響を与えます。新しい機能が立ち上げられた場合や、新しいオペレーションが使用可能になった場合に、各サービスが AWS マネージドポリシーを更新する可能性が最も高くなります。サービスは、AWS マネージドポリシーから許可を削除しないため、ポリシーの更新によって既存の許可が破棄されることはありません。

さらに、AWS は、複数のサービスにまたがるジョブ機能の特徴に対するマネージドポリシーもサポートしています。例えば、ReadOnlyAccess AWS マネージドポリシーでは、すべての AWS のサービスおよびリソースへの読み取り専用アクセスを許可します。サービスが新しい機能を起動する場合、AWS は、新たなオペレーションとリソース用に、読み取り専用の許可を追加します。ジョブ機能ポリシーのリストと説明については、IAM ユーザーガイドの「AWSジョブ関数のマネージドポリシー」を参照してください。

AWS 管理ポリシー: Lightsail ExportAccess

IAM エンティティに Lightsail ExportAccess をアタッチすることはできません。このポリシーは、ユーザーに代わって Lightsail がアクションを実行することを許可する、サービスにリンクされた

ロールにアタッチされます。詳細については、「サービスにリンクされたロール」を参照してください。

このポリシーは、Lightsail がインスタンスおよびディスクスナップショットを Amazon Elastic Compute Cloud にエクスポートし、現在のアカウントレベルで Block Public Access 設定を Amazon Simple Storage Service (Amazon S3) から取得できるようにするアクセス許可を付与します。

許可の詳細

このポリシーには、以下の許可が含まれています。

- ec2 – インスタンスイメージとディスクスナップショットの一覧表示とコピーをするためのアクセスを許可します。
- iam – サービスにリンクされたロールの削除と、サービスにリンクされたロールの削除のステータスを取得するためのアクセスを許可します。
- s3 – AWS アカウントの PublicAccessBlock 設定を取得するためのアクセスを許可します。

```
{
 "Version": "2012-10-17",
 "Statement": [
  {
   "Effect": "Allow",
   "Action": [
    "iam:DeleteServiceLinkedRole",
    "iam:GetServiceLinkedRoleDeletionStatus"
   ],
   "Resource": "arn:aws:iam::*:role/aws-service-role/lightsail.amazonaws.com/
AWSServiceRoleForLightsail*"
  },
  {
   "Effect": "Allow",
   "Action": [
    "ec2:CopySnapshot",
    "ec2:DescribeSnapshots",
    "ec2:CopyImage",
    "ec2:DescribeImages"
   ],
   "Resource": "*"
  },
  {
   "Effect": "Allow",
```

```
    "Action": [
     "s3:GetAccountPublicAccessBlock"
    ],
    "Resource": "*"
   }
  ]
}
```

Lightsail マネージドポリシーの AWS 更新

- `LightsailExportAccess` 管理ポリシーの編集

  `LightsailExportAccess` 管理ポリシーに `s3:GetAccountPublicAccessBlock` アクションが追加されました。現在のアカウントレベルの Block Public Access 設定を Amazon S3 から取得する許可を Lightsail に付与します。

  2022 年 1 月 14 日

- Lightsail は変更の追跡を開始しました

  Lightsail が AWS マネージドポリシーの変更の追跡を開始しました。

  2022 年 1 月 14 日

Amazon Lightsail と IAM の連携について

Lightsail へのアクセスを管理するために IAM を使用する前に、Lightsail でどの IAM特徴が使用できるかを理解しておく必要があります。Lightsail およびその他の AWS のサービスが IAM と連携する方法の概要を把握するには、IAM ユーザーガイドの「IAM と連携する AWS のサービス」を参照してください。

Lightsail アイデンティティベースのポリシー

IAM のアイデンティティベースポリシーでは、許可または拒否するアクションとリソース、またアクションを許可または拒否する条件を指定できます。Lightsail は、特定の操作、リソース、および条件キーをサポートしています。JSON ポリシーで使用するすべての要素については、「IAM ユーザーガイド」の「IAM JSON ポリシーの要素のリファレンス」を参照してください。

アクション

管理者は AWS JSON ポリシーを使用して、誰が何にアクセスできるかを指定できます。つまり、どの プリンシパル がどの リソース に対してどのような 条件 下で アクション を実行できるかということです。

JSON ポリシーの Action 要素には、ポリシー内のアクセスを許可または拒否するために使用できるアクションが記述されます。ポリシーアクションの名前は通常、関連する AWS API オペレーションと同じです。一致する API オペレーションのない許可のみのアクションなど、いくつかの例外があります。また、ポリシーに複数のアクションが必要なオペレーションもあります。これらの追加アクションは、依存アクションと呼ばれます。

このアクションは、関連付けられたオペレーションを実行するための許可を付与するポリシーで使用されます。

Lightsailのポリシーアクションは、アクションの前に次のプレフィックス を使用します: lightsail:。たとえば、Lightsail CreateInstances API オペレーションで Lightsail インスタンスを実行するためのアクセス許可をユーザーに付与するには、ポリシーに lightsail:CreateInstances アクションを含めます。ポリシーステートメントには、Action または NotAction 要素を含める必要があります。Lightsail は、このサービスで実行できるタスクを記述する独自のアクションのセットを定義します。

単一のステートメントに複数のアクションを指定するには、次のようにカンマで区切ります。

```
"Action": [
      "lightsail:action1",
      "lightsail:action2"
```

ワイルドカード (*) を使用して複数のアクションを指定することができます。たとえば、Create という単語で始まるすべてのアクションを指定するには、次のアクションを含めます。

```
"Action": "lightsail:Create*"
```

Lightsail アクションのリストを表示するには、IAM ユーザーガイドのAmazon Lightsail によって定義されたアクションを参照してください。

リソース

管理者は AWS JSON ポリシーを使用して、誰が何にアクセスできるかを指定できます。つまり、どのプリンシパルがどのリソースに対してどのような条件下でアクションを実行できるかということです。

Resource JSON ポリシー要素は、オブジェクトあるいはアクションが適用されるオブジェクトを指定します。ステートメントには、Resource または NotResource 要素を含める必要があります。ベストプラクティスとして、Amazon リソースネーム (ARN) を使用してリソースを指定します。これは、リソースレベルの許可と呼ばれる特定のリソースタイプをサポートするアクションに対して実行できます。

オペレーションのリスト化など、リソースレベルの許可をサポートしないアクションの場合は、ステートメントがすべてのリソースに適用されることを示すために、ワイルドカード (*) を使用します。

```
"Resource": "*"
```

> **Important**
>
> Lightsail は API アクションによっては、リソースレベルのアクセス許可をサポートしません。詳細については、「リソースレベルのアクセス許可およびタグに基づく承認のサポート」を参照してください。

Lightsail インスタンスリソースには次のような ARN があります。

```
arn:${Partition}:lightsail:${Region}:${Account}:Instance/${InstanceId}
```

ARN の形式の詳細については、「Amazon リソースネーム (ARN) と AWS サービスの名前空間」を参照してください。

例えば、ステートメントで ea123456-e6b9-4f1d-b518-3ad1234567e6 インスタンスを指定するには、次の ARN を使用します。

```
"Resource": "arn:aws:lightsail:us-east-1:123456789012:Instance/ea123456-e6b9-4f1d-
b518-3ad1234567e6"
```

特定のアカウントに属するすべてのインスタンスを指定するには、ワイルドカード (*) を使用します。

```
"Resource": "arn:aws:lightsail:us-east-1:123456789012:Instance/*"
```

リソースを作成するためのアクションなど、一部の Lightsail アクションは、特定のリソースでは実行できません。このような場合は、ワイルドカード (*) を使用する必要があります。

```
"Resource": "*"
```

Lightsail API アクションの多くが複数のリソースと関連します。たとえば、AttachDisk では Lightsail ブロックストレージディスクをインスタンスにアタッチするため、IAM ユーザーにはディスクおよびインスタンスを使用するアクセス許可が必要になります。複数のリソースを単一のステートメントで指定するには、ARN をカンマで区切ります。

```
"Resource": [
      "resource1",
      "resource2"
```

Lightsail リソースタイプとその ARN のリストを表示するには、IAM ユーザーガイドのAmazon Lightsail で定義されるリソースを参照してください。どのアクションで各リソースの ARN を指定できるかについては、Amazon Lightsail で定義されるアクションを参照してください。

条件キー

管理者は AWS JSON ポリシーを使用して、誰が何にアクセスできるかを指定できます。つまり、どのプリンシパルがどのリソースに対してどのような条件下でアクションを実行できるかということです。

Condition 要素 (または Condition ブロック) を使用すると、ステートメントが有効な条件を指定できます。Condition 要素はオプションです。イコールや未満などの条件演算子を使用して条件式を作成することで、ポリシーの条件とリクエスト内の値を一致させることができます。

1 つのステートメントに複数の Condition 要素を指定する場合、または 1 つの Condition 要素に複数のキーを指定する場合、AWS では AND 論理演算子を使用してそれらを評価します。単一の条件キーに複数の値を指定する場合、AWS では OR 論理演算子を使用して条件を評価します。ステートメントの許可が付与される前にすべての条件が満たされる必要があります。

条件を指定する際にプレースホルダー変数も使用できます。例えば IAM ユーザーに、IAM ユーザー名がタグ付けされている場合のみリソースにアクセスできる許可を付与することができます。詳細については、IAM ユーザーガイドの「IAM ポリシーの要素: 変数およびタグ」を参照してください。

AWS はグローバル条件キーとサービス固有の条件キーをサポートしています。すべての AWS グローバル条件キーを確認するには、IAM ユーザーガイドの「AWS グローバル条件コンテキストキー」を参照してください。

Lightsail にはサービス固有条件キーがありませんが、いくつかのグローバル条件キーの使用がサポートされています。すべてのAWSグローバル条件キーを確認するには、IAM ユーザーガイドの「AWS グローバル条件コンテキストキー」を参照してください。

Lightsail 条件キーのリストを確認するには、IAM ユーザーガイドの「Amazon Lightsail の条件キー」をご参照ください。どのアクションおよびリソースと条件キーを使用できるかについては、「Amazon Lightsail で定義されるアクション」を参照してください。

例

Lightsail のアイデンティティベースのポリシーの例については、「Amazon Lightsail アイデンティティベースのポリシーの例」を参照してください。

Lightsail リソースベースのポリシー

Lightsail は、リソースベースのポリシーをサポートしません。

アクセスコントロールリスト (ACL)

Lightsail はアクセスコントロールリスト (ACL) をサポートしません。

Lightsail タグに基づいた承認

タグを Lightsail リソースにアタッチすることも、Lightsail へのリクエストでタグを渡すこともできます。タグに基づいてアクセスを管理するには、`lightsail:ResourceTag/key-name`、`aws:RequestTag/key-name`、または `aws:TagKeys` の条件キーを使用して、ポリシーの [Condition element] (条件要素) でタグ情報を提供します。

> **Important**
>
> Lightsail では、一部の API アクションのタグに基づく認証はサポートされていません。詳細については、「リソースレベルのアクセス許可およびタグに基づく承認のサポート」を参照してください。

Lightsail リソースのタグ付けの詳細については、「タグ」を参照してください。

リソース上のタグに基づいてリソースへのアクセスを制限するためのアイデンティティベースのポリシーの例については、「タグに基づく Lightsail リソースの作成と削除の許可」を参照してください。

Lightsail IAM ロール

IAM ロールは AWS アカウント内のエンティティで、特定の許可を持っています。

Lightsail を使用した一時的な認証情報の使用

一時的な認証情報を使用して、フェデレーションでサインインする、IAM ロールを引き受ける、またはクロスアカウントロールを引き受けることができます。一時的なセキュリティ認証情報を取得するには、AssumeRole または GetFederationToken などの AWS STS API オペレーションを呼び出します。

Lightsail では、一時認証情報の使用をサポートしています。

サービスリンクロール

サービスにリンクされたロールは、AWS サービスが他のサービスのリソースにアクセスして自動的にアクションを完了することを許可します。サービスにリンクされたロールは IAM アカウント内に表示され、サービスによって所有されます。IAM 管理者は、サービスにリンクされたロールの許可を表示できますが、編集することはできません。

Lightsail はサービスにリンクされたロールをサポートします。Lightsail サービスにリンクされたロールの作成または管理の詳細については、「サービスにリンクされたロール」を参照してください。

サービスロール

Lightsail はサービスロールをサポートしていません。

トピック

- Amazon Lightsail アイデンティティベースポリシーの例
- Amazon Lightsail リソースレベルのアクセス許可ポリシーの例
- Amazon Lightsail のサービスにリンクされたロールの使用
- Amazon Lightsail でバケットを管理するための IAM ポリシー

Amazon Lightsail アイデンティティベースポリシーの例

デフォルトでは、IAM ユーザーおよびロールには、Lightsail リソースを作成または変更するアクセス許可はありません。AWS Management Console、AWS CLI、または AWS API を使用してタスクを実行することもできません。IAM 管理者は、ユーザーとロールに必要な、指定されたリソースで

特定の API オペレーションを実行する許可をユーザーとロールに付与する IAM ポリシーを作成する必要があります。続いて、管理者はそれらの許可が必要な IAM ユーザーまたはグループにそのポリシーをアタッチします。

JSON ポリシードキュメントのこれらの例を使用して、IAM アイデンティティベースのポリシーを作成する方法については、「IAM ユーザーガイド」の「JSON タブでのポリシーの作成」を参照してください。

ポリシーのベストプラクティス

ID ベースのポリシーは、ユーザーのアカウントで誰かが Amazon Lightsail リソースを作成、アクセス、または削除できるどうかを決定します。これらのアクションを実行すると、AWS アカウント に追加料金が発生する可能性があります。アイデンティティベースポリシーを作成したり編集したりする際には、以下のガイドラインと推奨事項に従ってください。

- AWS マネージドポリシーを使用して開始し、最小特権の許可に移行する – ユーザーとワークロードへの許可の付与を開始するには、多くの一般的なユースケースのために許可を付与する AWS マネージドポリシーを使用します。これらは AWS アカウント で使用できます。ユースケースに応じた AWS カスタマーマネージドポリシーを定義することで、許可をさらに減らすことをお勧めします。詳細については、「IAM ユーザーガイド」の「AWS マネージドポリシー」または「AWS ジョブ機能の管理ポリシー」を参照してください。
- 最小特権を適用する – IAM ポリシーで許可を設定するときは、タスクの実行に必要な許可のみを付与します。これを行うには、特定の条件下で特定のリソースに対して実行できるアクションを定義します。これは、最小特権アクセス許可とも呼ばれています。IAM を使用して許可を適用する方法の詳細については、「IAM ユーザーガイド」の「IAM でのポリシーとアクセス許可」を参照してください。
- IAM ポリシーで条件を使用してアクセスをさらに制限する – ポリシーに条件を追加して、アクションやリソースへのアクセスを制限できます。例えば、ポリシー条件を記述して、すべてのリクエストを SSL を使用して送信するように指定することができます。また、AWS のサービス などの特定の AWS CloudFormation を介して使用する場合、条件を使用してサービスアクションへのアクセスを許可することもできます。詳細については、「IAM ユーザーガイド」の [IAM JSON policy elements: Condition] (IAM JSON ポリシー要素：条件) を参照してください。
- IAM Access Analyzer を使用して IAM ポリシーを検証し、安全で機能的な許可を確保する - IAM Access Analyzer は、新規および既存のポリシーを検証して、ポリシーが IAM ポリシー言語 (JSON) および IAM のベストプラクティスに準拠するようにします。IAM Access Analyzer は 100 を超えるポリシーチェックと実用的な推奨事項を提供し、安全で機能的なポリシーを作成できるようサポートします。詳細については、「IAM ユーザーガイド」の「IAM Access Analyzer ポリシーの検証」を参照してください。

- 多要素認証 (MFA) を要求する – AWS アカウント で IAM ユーザーまたはルートユーザーを要求するシナリオがある場合は、セキュリティを強化するために MFA をオンにします。API オペレーションが呼び出されるときに MFA を必須にするには、ポリシーに MFA 条件を追加します。詳細については、「IAM ユーザーガイド」の「MFA 保護 API アクセスの設定」を参照してください。

IAM でのベストプラクティスの詳細については、「IAM ユーザーガイド」の「IAM でのセキュリティのベストプラクティス」を参照してください。

Lightsail コンソールの使用

Amazon Lightsail コンソールにアクセスするには、すべての Lightsail アクションとリソースに対するフルアクセス許可が必要です。これらの許可により、AWS アカウントの Lightsail リソースに関する詳細を一覧表示および表示できるようにする必要があります。最小限必要なアクセス許可よりも制限されたアイデンティティベースのポリシーを作成すると (つまり、フルアクセスではない)、そのポリシーをもつエンティティ (IAM ユーザーまたはロール) に対してはコンソールが意図したとおりに機能しません。

これらのエンティティが Lightsail コンソールを使用できるように、以下の 次のポリシーもそれらのエンティティにアタッチします。詳細については、IAM ユーザーガイドの「ユーザーへのアクセス許可の追加」を参照してください。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "lightsail:*"
            ],
            "Resource": "*"
        }
    ]
}
```

AWS CLI または AWS API のみを呼び出すユーザーには、最小限のコンソール許可を付与する必要はありません。代わりに、実行しようとしている API オペレーションに一致するアクションのみへのアクセスが許可されます。

ユーザーが自分のアクセス許可を表示できるようにする

この例では、ユーザーアイデンティティに添付されたインラインおよびマネージドポリシーの表示を IAM ユーザーに許可するポリシーを作成する方法を示します。このポリシーには、コンソールで、または AWS CLI か AWS API を使用してプログラム的に、このアクションを完了するアクセス許可が含まれています。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "ViewOwnUserInfo",
            "Effect": "Allow",
            "Action": [
                "iam:GetUserPolicy",
                "iam:ListGroupsForUser",
                "iam:ListAttachedUserPolicies",
                "iam:ListUserPolicies",
                "iam:GetUser"
            ],
            "Resource": ["arn:aws:iam::*:user/${aws:username}"]
        },
        {
            "Sid": "NavigateInConsole",
            "Effect": "Allow",
            "Action": [
                "iam:GetGroupPolicy",
                "iam:GetPolicyVersion",
                "iam:GetPolicy",
                "iam:ListAttachedGroupPolicies",
                "iam:ListGroupPolicies",
                "iam:ListPolicyVersions",
                "iam:ListPolicies",
                "iam:ListUsers"
            ],
            "Resource": "*"
        }
    ]
}
```

タグに基づく Lightsail リソースの作成と削除の許可

アイデンティティベースのポリシーの条件を使用して、タグに基づいて Lightsail リソースへのアクセスをコントロールできます。この例では、作成リクエストで `allow` のキータグと `true` の値が定義されていない限り、ユーザーが新しい Lightsail リソースを作成できないようにするポリシーを作成する方法を示します。このポリシーは、 `allow/true` のキーバリューのタグが定義されていない限り、ユーザーによるリソースの削除も禁止します。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "lightsail:Create*",
                "lightsail:TagResource",
                "lightsail:UntagResource"
            ],
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "aws:RequestTag/allow": "true"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "lightsail:Delete*",
                "lightsail:TagResource",
                "lightsail:UntagResource"
            ],
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "aws:ResourceTag/allow": "true"
                }
            }
        }
    ]
}
```

次のポリシーでは、キーと値のタグが allow/false ではないリソースのタグの変更をユーザーに禁止します。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Deny",
            "Action": [
                "lightsail:TagResource"
            ],
            "Resource": "*",
            "Condition": {
                "StringNotEquals": {
                    "aws:ResourceTag/allow": "false"
                }
            }
        }
    ]
}
```

これらのポリシーは、アカウントの IAM ユーザーにアタッチできます。詳細については、「IAM ユーザーガイド」の「IAM JSON Policy Elements: Condition」(IAM JSON ポリシー要素: 条件) を参照してください。

Amazon Lightsail リソースレベルのアクセス許可ポリシーの例

リソースレベルのアクセス許可とは、ユーザーがアクションを実行できるリソースを指定できる機能を意味します。Amazon Lightsail では、リソースレベルのアクセス許可がサポートされます。これは、特定の Lightsail アクションでは、満たす必要がある条件、またはユーザーが使用したり、編集したりできる特定のリソースに基づいて、ユーザーがそれらのアクションをいつ使用できるかを制御できることを意味します。たとえば、特定の Amazon リソースネーム (ARN) を使用して、インスタンスまたはデータベースを管理するアクセス許可をユーザーに付与することができます。

> **Important**
>
> Lightsail は、一部の API アクションにはリソースレベルのアクセス許可をサポートしません。詳細については、「リソースレベルのアクセス許可およびタグに基づく承認のサポート」を参照してください。

Lightsail アクション、ARN および IAM ポリシーステートメントで使用できる Lightsail 条件キーによって作成または変更されるリソースの詳細については、「IAM ユーザーガイド」の 「Amazon Lightsail の操作、リソース、条件キー」 を参照してください。

特定のインスタンスの管理を許可する

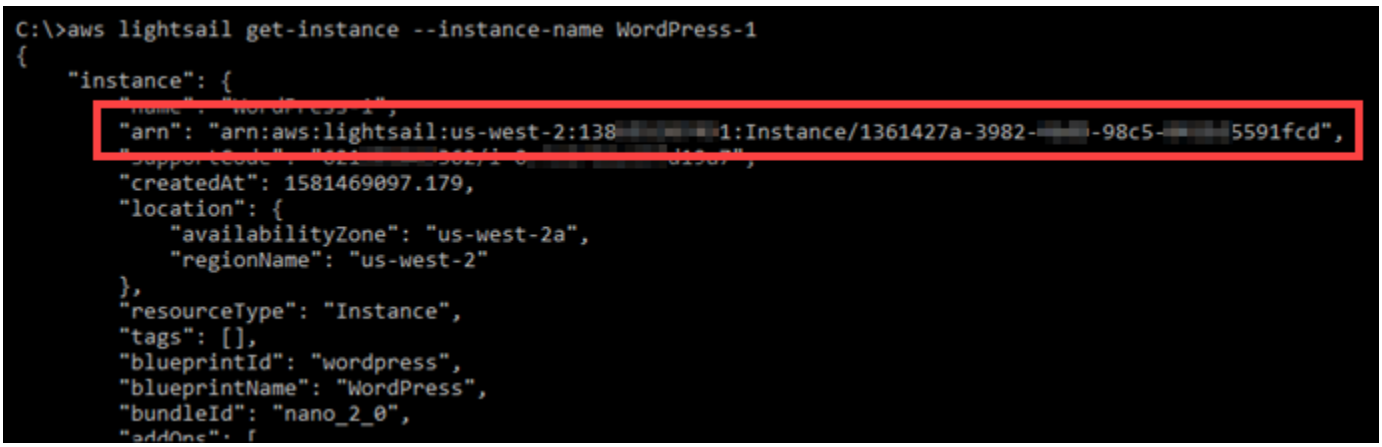
次のポリシーでは、インスタンスの再起動/開始/停止、インスタンスポートの管理、特定のインスタンスのインスタンススナップショットの作成へのアクセス権を付与します。また、Lightsail アカウント内の他のインスタンス関連の情報やリソースへの読み取り専用アクセスも提供します。ポリシーで、 *InstanceARN* をインスタンスの Amazon リソースネーム ((ARN) に置き換えます。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "VisualEditor0",
            "Effect": "Allow",
            "Action": [
                "lightsail:GetActiveNames",
                "lightsail:GetAlarms",
                "lightsail:GetAutoSnapshots",
                "lightsail:GetBlueprints",
                "lightsail:GetBundles",
                "lightsail:GetCertificates",
                "lightsail:GetCloudFormationStackRecords",
                "lightsail:GetContactMethods",
                "lightsail:GetDisk",
                "lightsail:GetDisks",
                "lightsail:GetDiskSnapshot",
                "lightsail:GetDiskSnapshots",
                "lightsail:GetDistributionBundles",
                "lightsail:GetDistributionLatestCacheReset",
                "lightsail:GetDistributionMetricData",
                "lightsail:GetDistributions",
                "lightsail:GetDomain",
                "lightsail:GetDomains",
                "lightsail:GetExportSnapshotRecords",
                "lightsail:GetInstance",
                "lightsail:GetInstanceAccessDetails",
                "lightsail:GetInstanceMetricData",
                "lightsail:GetInstancePortStates",
                "lightsail:GetInstances",
                "lightsail:GetInstanceSnapshot",
```

```
                "lightsail:GetInstanceSnapshots",
                "lightsail:GetInstanceState",
                "lightsail:GetKeyPair",
                "lightsail:GetKeyPairs",
                "lightsail:GetLoadBalancer",
                "lightsail:GetLoadBalancerMetricData",
                "lightsail:GetLoadBalancers",
                "lightsail:GetLoadBalancerTlsCertificates",
                "lightsail:GetOperation",
                "lightsail:GetOperations",
                "lightsail:GetOperationsForResource",
                "lightsail:GetRegions",
                "lightsail:GetRelationalDatabase",
                "lightsail:GetRelationalDatabaseBlueprints",
                "lightsail:GetRelationalDatabaseBundles",
                "lightsail:GetRelationalDatabaseEvents",
                "lightsail:GetRelationalDatabaseLogEvents",
                "lightsail:GetRelationalDatabaseLogStreams",
                "lightsail:GetRelationalDatabaseMetricData",
                "lightsail:GetRelationalDatabaseParameters",
                "lightsail:GetRelationalDatabases",
                "lightsail:GetRelationalDatabaseSnapshot",
                "lightsail:GetRelationalDatabaseSnapshots",
                "lightsail:GetStaticIp",
                "lightsail:GetStaticIps",
                "lightsail:IsVpcPeered"
            ],
            "Resource": "*"
        },
        {
            "Sid": "VisualEditor2",
            "Effect": "Allow",
            "Action": [
                "lightsail:CloseInstancePublicPorts",
                "lightsail:CreateInstanceSnapshot",
                "lightsail:OpenInstancePublicPorts",
                "lightsail:PutInstancePublicPorts",
                "lightsail:RebootInstance",
                "lightsail:StartInstance",
                "lightsail:StopInstance"
            ],
            "Resource": "InstanceARN"
        }
    ]
```

```
}
```

インスタンスの ARN を取得するには、GetInstance Lightsail API アクションを使用し、instanceName パラメータを使用してインスタンスの名前を指定します。インスタンス ARN は、次の例に示すように、そのアクションの結果に一覧表示されます。詳細については、Amazon Lightsail API リファレンスの GetInstance を参照してください。



特定のデータベースの管理を許可する

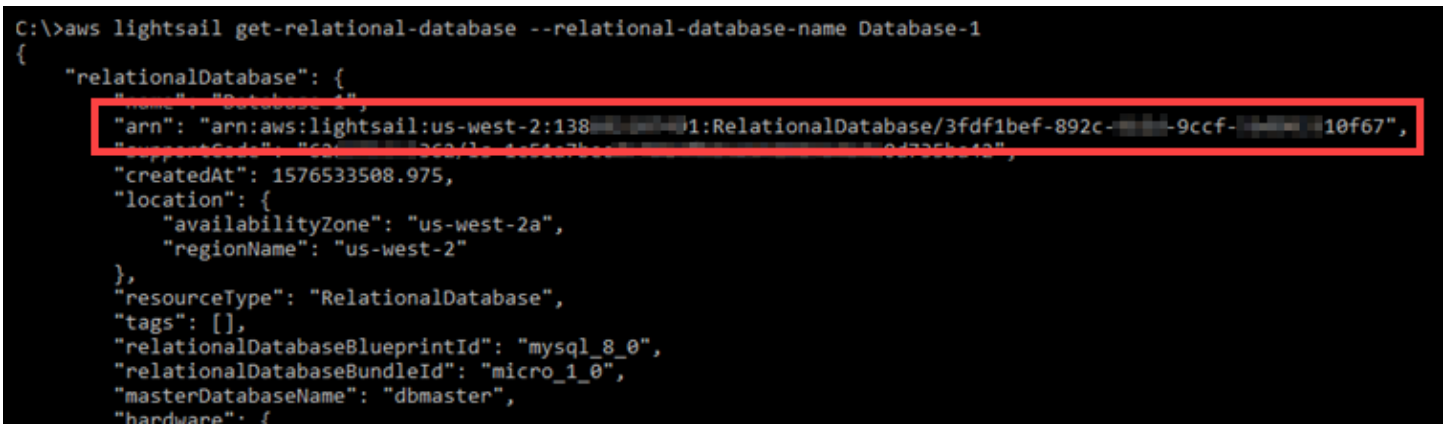
次のポリシーは、特定のデータベースの再起動/開始/停止および更新へのアクセス権を付与します。また、Lightsail アカウント内の他のデータベース関連の情報やリソースへの読み取り専用アクセスも提供します。ポリシーで、*DatabaseARN* をデータベースの Amazon リソースネーム (ARN) に置き換えます。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "VisualEditor0",
            "Effect": "Allow",
            "Action": [
                "lightsail:GetActiveNames",
                "lightsail:GetAlarms",
                "lightsail:GetAutoSnapshots",
                "lightsail:GetBlueprints",
                "lightsail:GetBundles",
                "lightsail:GetCertificates",
                "lightsail:GetCloudFormationStackRecords",
                "lightsail:GetContactMethods",
                "lightsail:GetDisk",
                "lightsail:GetDisks",
```

```
                "lightsail:GetDiskSnapshot",
                "lightsail:GetDiskSnapshots",
                "lightsail:GetDistributionBundles",
                "lightsail:GetDistributionLatestCacheReset",
                "lightsail:GetDistributionMetricData",
                "lightsail:GetDistributions",
                "lightsail:GetDomain",
                "lightsail:GetDomains",
                "lightsail:GetExportSnapshotRecords",
                "lightsail:GetInstance",
                "lightsail:GetInstanceAccessDetails",
                "lightsail:GetInstanceMetricData",
                "lightsail:GetInstancePortStates",
                "lightsail:GetInstances",
                "lightsail:GetInstanceSnapshot",
                "lightsail:GetInstanceSnapshots",
                "lightsail:GetInstanceState",
                "lightsail:GetKeyPair",
                "lightsail:GetKeyPairs",
                "lightsail:GetLoadBalancer",
                "lightsail:GetLoadBalancerMetricData",
                "lightsail:GetLoadBalancers",
                "lightsail:GetLoadBalancerTlsCertificates",
                "lightsail:GetOperation",
                "lightsail:GetOperations",
                "lightsail:GetOperationsForResource",
                "lightsail:GetRegions",
                "lightsail:GetRelationalDatabase",
                "lightsail:GetRelationalDatabaseBlueprints",
                "lightsail:GetRelationalDatabaseBundles",
                "lightsail:GetRelationalDatabaseEvents",
                "lightsail:GetRelationalDatabaseLogEvents",
                "lightsail:GetRelationalDatabaseLogStreams",
                "lightsail:GetRelationalDatabaseMetricData",
                "lightsail:GetRelationalDatabaseParameters",
                "lightsail:GetRelationalDatabases",
                "lightsail:GetRelationalDatabaseSnapshot",
                "lightsail:GetRelationalDatabaseSnapshots",
                "lightsail:GetStaticIp",
                "lightsail:GetStaticIps",
                "lightsail:IsVpcPeered"
            ],
            "Resource": "*"
        },
```

```
        {
            "Sid": "VisualEditor2",
            "Effect": "Allow",
            "Action": [
                "lightsail:RebootRelationalDatabase",
                "lightsail:StartRelationalDatabase",
                "lightsail:StopRelationalDatabase",
                "lightsail:UpdateRelationalDatabase"
            ],
            "Resource": "DatabaseARN"
        }
    ]
}
```

データベースの ARN を取得するには、`GetRelationalDatabase` Lightsail API アクションを使用し、`relationalDatabaseName` パラメータを使用してデータベースの名前を指定します。データベース ARN は、次の例に示すように、そのアクションの結果に一覧表示されます。詳細については、Amazon Lightsail API リファレンス の GetBucketEncryption を参照してください。



Amazon Lightsail のサービスにリンクされたロールの使用

Amazon Lightsail は AWS Identity and Access Management (IAM) サービスにリンクされたロールを使用します。サービスにリンクされたロールは、Amazon Lightsail に直接リンクされた一意のタイプの IAM ロールです。サービスにリンクされたロールは、Amazon Lightsail による事前定義済みのロールであり、ユーザーに代わって Lightsail から AWS の他のサービスを呼び出すために必要なすべてのアクセス権限を備えています。

サービスにリンクされたロールを使用することで、必要なアクセス権限を手動で追加する必要がなくなるため、Amazon Lightsail の設定が簡単になります。Amazon Lightsail は、サービスにリンクされたロールのアクセス許可を定義します。特に定義されている場合を除き、Amazon Lightsail のみがそ

のロールを引き受けることができます。定義されたアクセス許可には、他の IAM エンティティにアタッチできない信頼ポリシーとアクセス許可ポリシーが含まれます。

サービスにリンクされたロールは、まずその関連リソースを削除しなければ削除できません。これにより、リソースへの意図しないアクセスによるアクセス許可の削除が防止され、Amazon Lightsail リソースは保護されます。

サービスリンクロールをサポートする他のサービスについては、「IAM と連携する AWS のサービス」でサービスリンクロール列がはいになっているサービスを検索してください。サービスリンクロールに関するドキュメントをサービスで表示するには、[Yes] (はい) リンクを選択します。

Amazon Lightsail のサービスにリンクされたロールのアクセス許可

Amazon Lightsail は、[AWSServiceRoleForLightsail] という名前のサービスにリンクされたロールを使用して、Amazon Elastic Compute Cloud (Amazon EC2)に Lightsail インスタンスとブロックストレージのディスクのスナップショットをエクスポートしたり、Amazon Simple Storage Service (Amazon S3) から現在のアカウントレベルのブロックパブリックアクセス設定を取得したりします。

Lightsail サービスにリンクされたロール AWSServiceRoleForECS は、次のサービスを信頼してロールを引き受けます。

- `lightsail.amazonaws.com`

ロールのアクセス許可ポリシーは、指定したリソースに対して以下のアクションを実行することを Amazon Lightsail に許可します。

- アクション: すべての AWS リソースに対する `ec2:CopySnapshot`。
- アクション: すべての AWS リソースに対する `ec2:DescribeSnapshots`。
- アクション: すべての AWS リソースに対する `ec2:CopyImage`。
- アクション: すべての AWS リソースに対する `ec2:DescribeImages`。
- アクション: すべての AWS AWS CloudFormation スタック上の `cloudformation:DescribeStacks`。
- アクション: すべての AWS リソースに対する `s3:GetAccountPublicAccessBlock`。

サービスにリンクされたロールの権限

IAM; エンティティ (ユーザー、グループ、ロールなど) がサービスにリンクされたロールの説明を作成または編集できるようにするには、アクセス許可を設定する必要があります。

特定のサービスにリンクされたロールの作成を IAM エンティティに許可するには

サービスにリンクされたロールを作成する必要のある IAM エンティティに、次のポリシーを追加します。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "iam:CreateServiceLinkedRole",
            "Resource": "arn:aws:iam::*:role/aws-service-role/lightsail.amazonaws.com/
AWSServiceRoleForLightsail*",
            "Condition": {"StringLike": {"iam:AWSServiceName":
 "lightsail.amazonaws.com"}}
        },
        {
            "Effect": "Allow",
            "Action": "iam:PutRolePolicy",
            "Resource": "arn:aws:iam::*:role/aws-service-role/lightsail.amazonaws.com/
AWSServiceRoleForLightsail*"
        }
    ]
}
```

IAM エンティティがサービスにリンクされた任意のロールを作成することを許可するには

サービスにリンクされたロール、または必要なポリシーを含む任意のサービスロールを作成する必要のある IAM エンティティのアクセス許可ポリシーに、次のステートメントを追加します。このポリシーにより、ロールにポリシーがアタッチされます。

```
{
    "Effect": "Allow",
    "Action": "iam:CreateServiceLinkedRole",
    "Resource": "arn:aws:iam::*:role/aws-service-role/*"
```

```
}
```

IAM エンティティが任意のサービスロールの説明を編集することを許可するには

サービスにリンクされたロール、または任意のサービスロールの説明を編集する必要のある IAM エンティティのアクセス許可ポリシーに、次のステートメントを追加します。

```
{
    "Effect": "Allow",
    "Action": "iam:UpdateRoleDescription",
    "Resource": "arn:aws:iam::*:role/aws-service-role/*"
}
```

IAM エンティティがサービスにリンクされた特定のロールを削除することを許可するには

サービスにリンクされたロールを削除する必要のある IAM エンティティのアクセス許可ポリシーに、次のステートメントを追加します。

```
{
    "Effect": "Allow",
    "Action": [
        "iam:DeleteServiceLinkedRole",
        "iam:GetServiceLinkedRoleDeletionStatus"
    ],
    "Resource": "arn:aws:iam::*:role/aws-service-role/lightsail.amazonaws.com/
AWSServiceRoleForLightsail*"
}
```

IAM エンティティがサービスロールを削除することを許可するには

サービスにリンクされたロール、または任意のサービスロールを削除する必要のある IAM; エンティティのアクセス許可ポリシーに、次のステートメントを追加します。

```
{
    "Effect": "Allow",
    "Action": [
```

```
        "iam:DeleteServiceLinkedRole",
        "iam:GetServiceLinkedRoleDeletionStatus"
    ],
    "Resource": "arn:aws:iam::*:role/aws-service-role/*"
}
```

または、AWS 管理ポリシーを使用して、サービスへのフルアクセスを付与することもできます。

Amazon Lightsail のサービスにリンクされたロールの作成

サービスにリンクされたロールを手動で作成する必要はありません。Lightsail インスタンスまたはブロックストレージのディスクスナップショットを Amazon EC2 にエクスポートするか、AWS AWS Management Console、AWS CLI または AWS API で Lightsail バケットを作成または更新すると、Amazon Lightsail には、サービスにリンクされたロールが作成されます。

このサービスにリンクされたロールを削除した後に、そのロールを再作成する必要がある場合は、同じプロセスを使用してアカウントでロールを再作成することができます。Lightsail インスタンスまたはブロックストレージのディスクスナップショットを Amazon EC2 にエクスポートしたり、Lightsail バケットを作成または更新したりすると、Amazon Lightsail には、サービスにリンクされたロールが作成されます。

> **Important**
>
> サービスにリンクされたロールの作成を Amazon Lightsail に許可するように IAM アクセス許可を設定する必要があります。これを行うには、次の「サービスにリンクされたロールのアクセス許可」セクションのステップを実行します。

Amazon Lightsail のサービスにリンクされたロールの編集

Amazon Lightsail では、AWSServiceRoleForFIS Lightsail のサービスにリンクされたロールを編集することはできません。サービスにリンクされたロールを作成すると、多くのエンティティによってロールが参照される可能性があるため、ロール名を変更することはできません。ただし、IAM を使用したロールの説明の編集はできます。詳細については、「IAM ユーザーガイド」の「サービスにリンクされたロールの編集」を参照してください。

Amazon Lightsail のサービスにリンクされたロールの削除

サービスにリンクされたロールが必要な機能またはサービスが不要になった場合には、そのロールを削除することをお勧めします。そうすることで、積極的にモニタリングまたは保守されていない未使

用のエンティティを排除できます。ただし、AWSServiceRoleFor Lightsail サービスにリンクされたロールを削除する前に、コピーが保留中の状態になっている Amazon Lightsail のインスタンスまたはディスクスナップショットがないことを確認する必要があります。詳細については、「スナップショットを Amazon EC2 にエクスポートする」を参照してください。

IAM を使用してサービスリンクロールを手動で削除するには

IAM コンソール、AWS CLI、または AWS API を使用して、Lightsail サービスにリンクされたロールである AWSServiceRoleForECS を削除します。詳細については、「IAM ユーザーガイド」の「サービスにリンクされたロールの削除」を参照してください。

Amazon Lightsail のサービスにリンクされたロールをサポートするリージョン

Amazon Lightsail は、サービスを利用できるすべてのリージョンで、サービスにリンクされたロールの使用をサポートします。Lightsail が使用できるリージョンの詳細については、「Amazon Lightsail リージョン」 を参照してください。

Amazon Lightsail でバケットを管理するための IAM ポリシー

以下のポリシーは、Amazon Lightsail オブジェクトストレージサービスにて特定のバケットを管理するためのアクセス権限をユーザーに付与します。このポリシーは、Lightsail コンソール、AWS Command Line Interface (AWS CLI)、AWS API、および AWS SDK を通じてバケットにアクセスを許可します。ポリシーで、*<BucketName>*を管理するバケット名に置き換えます。IAM ポリシーの詳細については、AWS Identity and Access Management ユーザーガイドの「IAM ポリシーの作成」を参照してください。IAM ユーザーとユーザーグループの作成の詳細は、AWS Identity and Access Managementユーザーガイド の「最初の IAM 委任ユーザーとユーザー グループの作成」を参照してください。

> **Important**
>
> このポリシーを持たないユーザーは、Lightsail コンソールのバケット管理ページにあるオブジェクトタブを表示する際にエラーが発生します。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "LightsailAccess",
            "Effect": "Allow",
```

```
            "Action": "lightsail:*",
            "Resource": "*"
        },
        {
            "Sid": "S3BucketAccess",
            "Effect": "Allow",
            "Action": "s3:*",
            "Resource": [
                "arn:aws:s3:::<BucketName>/*",
                "arn:aws:s3:::<BucketName>"
            ]
        }
    ]
}
```

バケットとオブジェクトを管理する

これらは、Lightsail オブジェクトストレージバケットを管理する一般的な手順です。

1. Amazon Lightsail オブジェクトストレージサービスでのオブジェクトとバケットについて説明します。詳細については、「Amazon Lightsail のオブジェクトストレージ」を参照してください。
2. Amazon Lightsail でバケットに付けることができる名前について説明します。詳細については、「Amazon Lightsail でのバケットの命名規則」をご参照ください。
3. バケットを作成して、Lightsail オブジェクトストレージサービスの使用を開始します。詳細については、「Amazon Lightsail におけるバケットの作成 」を参照してください。
4. バケットのセキュリティのベストプラクティスと、バケットに設定できるアクセス許可について説明します。バケット内のすべてのオブジェクトをパブリックまたはプライベートにすることも、オブジェクトを個別に選択してパブリックにすることもできます。また、アクセスキーを作成し、インスタンスをバケットに追加し、他の AWS アカウントにアクセス権を付与することで、バケットへのアクセスを許可することもできます。詳細については、「Amazon Lightsail オブジェクトストレージのセキュリティベストプラクティス」と「Amazon Lightsail でのバケットのアクセス許可を理解する」を参照してください。

   バケットのアクセス許可について理解したら、以下のガイドを参照してバケットへのアクセスを許可してください。

   - Amazon Lightsail でのバケットへのパブリックアクセスをブロックする
   - Amazon Lightsail でのバケットのアクセス許可の設定
   - Amazon Lightsail でのバケット内の個々のオブジェクトに対するアクセス許可の設定
   - Amazon Lightsail でのバケットのアクセスキーの作成

- Amazon Lightsail でのバケットのリソースアクセスの設定
- Amazon Lightsail でのバケットのクロスアカウントアクセスの設定

5. バケットのアクセスログ記録を有効にする方法と、アクセスログを使用してバケットのセキュリティを監査する方法について説明します。詳細については、以下のガイドを参照してください。
   - Amazon Lightsail オブジェクトストレージサービスのバケットへのアクセスのログ記録
   - Amazon Lightsail オブジェクトストレージサービスのバケットのアクセスログの形式
   - Amazon Lightsail オブジェクトストレージサービスのバケットへのアクセスのログ記録を有効にする
   - Amazon Lightsailでのバケットのアクセスログを使用するリクエストの特定
6. Lightsail でバケットを管理する機能をユーザーに付与する IAM ポリシーを作成します。詳細については、「Amazon Lightsail でバケットを管理する IAM ポリシー」を参照してください。
7. バケット内のオブジェクトにラベルを付けて識別する方法について説明します。詳細については、「Amazon Lightsail でのオブジェクトキー名を理解する」を参照してください。
8. ファイルをアップロードしてバケット内のオブジェクトを管理する方法について説明します。詳細については、以下のガイドを参照してください。
   - Amazon Lightsail のバケットにファイルをアップロードする
   - マルチパートアップロードを使用した Amazon Lightsail のバケットへのファイルのアップロード
   - Amazon Lightsail のバケット内のオブジェクトの表示
   - Amazon Lightsail のバケット内のオブジェクトのコピーまたは移動
   - Amazon Lightsail のバケットからのオブジェクトのダウンロード
   - Amazon Lightsail のバケット内のオブジェクトのフィルタリング
   - Amazon Lightsail のバケット内のオブジェクトのタグ付け
   - Amazon Lightsail のバケット内のオブジェクトの削除
9. オブジェクトのバージョニングを有効にすると、バケットに保存されたあらゆるオブジェクトのあらゆるバージョンを保存、取得、復元します。詳細については、「Amazon Lightsail のバケットでのオブジェクトのバージョニングの有効化と一時停止」を参照してください。
10. オブジェクトのバージョニングを有効にすると、バケット内のオブジェクトの以前のバージョンを復元できます。詳細については、「Amazon Lightsail のバケット内のオブジェクトの以前のバージョンの復元」を参照してください。
11. バケットの使用率を監視します。詳細については、「Amazon Lightsail でのバケットのメトリクスの表示」を参照してください。

12バケットの使用率がしきい値を超えたときにバケットメトリクスが通知されるよう、アラームを設定します。詳細については、「Amazon Lightsail でのバケットメトリクスアラームの作成」を参照してください。

13ストレージとネットワーク転送量が不足している場合は、バケットのストレージプランを変更します。詳細については、「Amazon Lightsail のバケットのプランの変更」を参照してください。

14バケットを他のリソースに接続する方法について説明します。詳細については、以下のチュートリアルを参照してください。

- チュートリアル: WordPress インスタンスの Amazon Lightsail バケットへの接続
- チュートリアル: Lightsail コンテンツ配信ネットワークディストリビューションでの Amazon Lightsail バケットの使用

15使用しなくなったバケットを削除します。詳細については、「Amazon Lightsail でのバケットの削除」を参照してください。

IAM ユーザーの Amazon Lightsail へのアクセスを管理します。

AWS アカウントのルートユーザー、管理者アクセス権限を持つ AWS Identity and Access Management (IAM) ユーザーは、AWS アカウントに 1 つ以上の IAM ユーザーを作成することができ、これらのユーザーは、AWS が提供するサービスへのアクセス権限をさまざまなレベルで設定されます。

Amazon Lightsail では、Lightsail にのみアクセスできる IAM ユーザーを作成するとよいでしょう。Lightsail リソースを表示、作成、編集、または削除するためのアクセス許可を必要としているが AWS が提供する他のサービスへのアクセス許可を必要としない誰かがチームに加わるときに、これを行います。これを設定するにはまず、Lightsail へのアクセス許可を付与する IAM ポリシーを作成する必要があります。その後、IAM グループを作成し、ポリシーをそのグループにアタッチします。その後、IAM ユーザーを作成してグループのメンバーにします。これにより、Lightsail へのアクセス許可が付与されます。

誰かがチームを去るとき、たとえばその人物が、チームを離れるが引き続き会社に勤める場合、Lightsail アクセスグループからそのユーザーを削除して、Lightsail へのアクセスを取り消すことができます。あるいは、たとえばユーザーが退社して今後アクセス権限を必要としない場合、IAM からそのユーザーを削除できます。

目次

- IAM ユーザーを作成して、そのユーザーを Lightsail アクセスグループに追加する

Lightsail アクセス用の IAM ポリシーを作成します

Lightsail アクセス権のための IAM ポリシーを作成するには、以下の手順に従います。詳細については、IAM ドキュメントの IAM ポリシーの作成を参照してください。

1. IAM コンソールにサインインします。
2. 左のナビゲーションペインの [ポリシー] を選択します。
3. [Create Policy] (ポリシーの作成) を選択します。
4. [Create Policy (ポリシーの作成)] ページで、[JSON] タブを選択します。

5. テキストボックスの内容をハイライトしてから、次のポリシー構成テキストをコピーして貼り付けます。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "lightsail:*"
            ],
            "Resource": "*"
        }
    ]
}
```

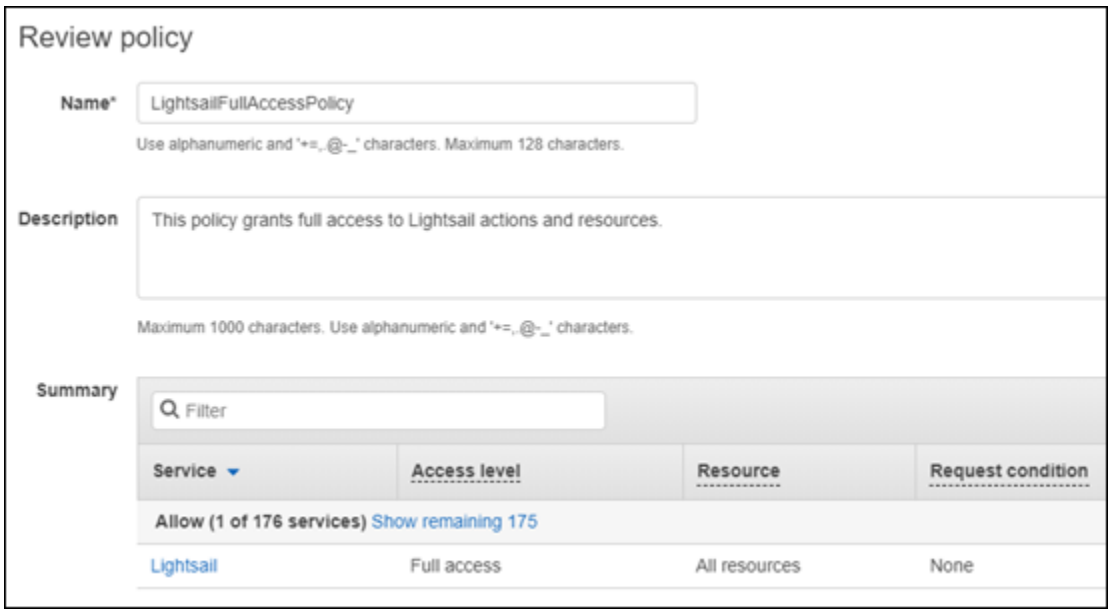
結果は次の例のようになります。

これにより、すべての Lightsail アクションおよびリソースへのアクセス許可を付与します。VPC ピアリングの有効化、Lightsail スナップショットの Amazon EC2 へのエクスポート、Lightsail を使用した Amazon EC2 リソースの作成など、AWS が提供する他のサービスへのアクセスを必要とするアクションには、このポリシーに含まれない追加の権限が必要です。詳細については、以下のガイドを参照してください。

- Amazon Lightsail の外部の AWS リソースを使用するために Amazon VPC ピア接続をセットアップする
- Amazon Lightsail のスナップショットを Amazon EC2 にエクスポートする
- Lightsail でエクスポートしたスナップショットから Amazon EC2 インスタンスを作成する

アクション固有およびリソース固有の許可が与えられる例については、Amazon Lightsail リソースレベルの許可ポリシー例 を参照してください。

6. [Review Policy (ポリシーの確認)] を選択します。
7. [ポリシーの確認] ページで、ポリシー名を選択します。分かりやすい名前 (例: `LightsailFullAccessPolicy`) をつけます。
8. 説明を追加し、ポリシー設定を確認します。変更が必要な場合は、[戻る] を選択してポリシーを変更します。



9. ポリシーの設定が正しいことを確認したら、[Create Policy (ポリシーの作成)] を選択します。

   これでポリシーが作成され、既存の IAM グループに追加することも、このガイドの次のセクションの手順に従って、新しい IAM グループを作成することもできます。

Lightsail アクセス許可の IAM グループを作成し、Lightsail アクセスポリシーをアタッチする

Lightsail アクセス許可のために IAM グループを作成し、本ガイドの前出のセクションで作成した Lightsail アクセスポリシーをアタッチするには、この手順に従います。詳細については、IAM ドキュメント内にある「IAM グループの作成」および「IAM グループへのポリシーのアタッチ」を参照してください。

1. IAM コンソールの左側のナビゲーションペインで [グループ] を選択します。
2. [Create New Group (新しいグループの作成)] を選択します。
3. [Set Group Name (グループ名の設定)] ページで、グループを選択します。分かりやすい名前 (例: `LightsailFullAccessGroup`) をつけます。
4. [Attach Policy (ポリシーのアタッチ)] ページで、本ガイドで作成した Lightsail ポリシー (例: `LightsailFullAccessPolicy`) を検索します。
5. ポリシーの横にチェックマークを追加し、[Next step (次のステップ)] を選択します。
6. グループの設定を確認します。変更が必要な場合は、[戻る] を選択してグループのポリシーを変更します。

7. グループの設定が正しいことを確認したら、[グループの作成] を選択します。

   これでグループが作成され、グループに追加されたユーザーは Lightsail のアクションとリソースにアクセスできるようになります。本ガイドの次のセクションのステップに従って、既存の IAM ユーザーをグループに追加するか、新しい IAM ユーザーを作成することができます。

IAM ユーザーを作成して、そのユーザーを Lightsail アクセスグループに追加する

IAM ユーザーを作成して、そのユーザーを Lightsail アクセスグループに追加するには、次のステップに従います。詳細については、IAM ドキュメントの「AWS アカウントで IAM ユーザーを作成する」および「IAM グループでユーザーを追加または削除する」を参照してください。

1. IAM コンソールの左側のナビゲーションペインで、[ユーザー] を選択します。
2. [Add user] (ユーザーを追加) を選択します。
3. ページの [Set user details (ユーザー詳細の設定)] セクションで、ユーザー名をつけます。
4. ページの [AWS のアクセスタイプの選択] セクションで、次のオプションから選択します。

   a. [Programmatic Access (プログラムによるアクセス)] を選択して、AWS API、CLI、SDK、その他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、Lightsail アクションまたはリソースで使用できるようにします。詳細については、「Lightsail で使用するために AWS CLI を設定する」を参照してください。

   b. [AWS マネジメントコンソールへのアクセス] を選択してパスワードを有効にすると、ユーザーは AWS マネジメントコンソールにサインインできるようになり、それに従い Lightsail コンソールにもサインインできるようになります。このオプションが選択されたとき、次のパスワードオプションが表示されます。

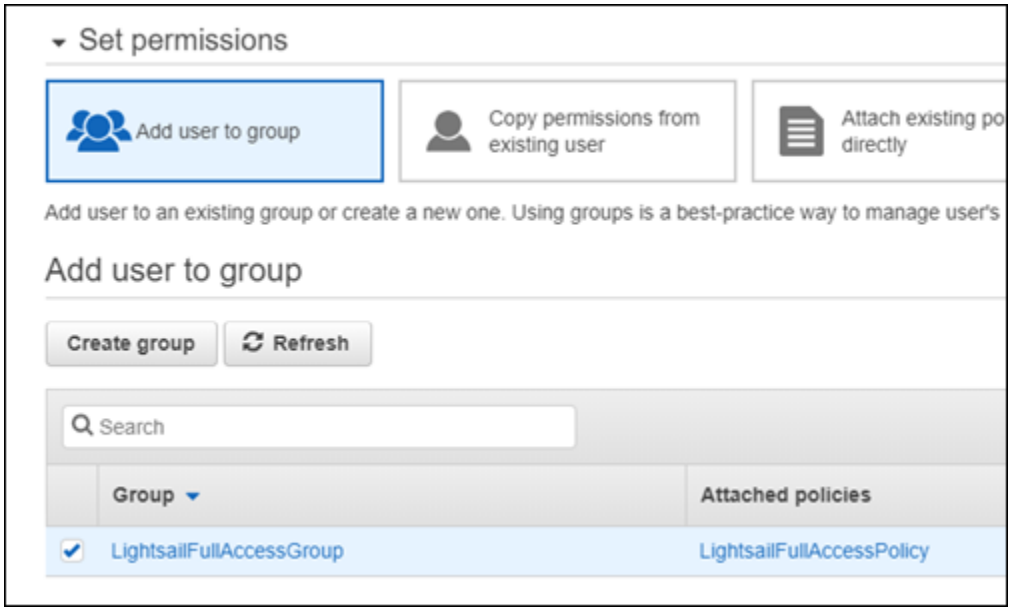
      i. [自動生成パスワード] を選択すると IAM がパスワードを生成し、または、[カスタムパスワード] を選択すると独自のパスワードを入力できます。

      ii. [Require password reset (パスワードのリセットが必要)] を選択すると、次回のログイン時にユーザーが新しいパスワードを作成します (パスワードをリセットする)。

   > **Note**
   >
   > [Programmatic Access (プログラムを使用したアクセス)] のみを選択した場合、ユーザーは AWS コンソール、および Lightsail コンソールにサインインできません。

5. [Next: Permissions] (次へ: 許可) を選択します。

6. ページの [Set permissions (許可を設定)] セクションで [ユーザーをグループに追加] を選択して、本ガイドの前半で作成した Lightsail アクセスグループ (例: `LightsailFullAccessGroup`) を選択します。



7. [Next: Tags] (次へ: タグ) を選択します。
8. (オプション) タグをキーバリューペアとしてアタッチして、メタデータをユーザーに追加します。IAM でのタグの仕様について詳細は、「IAM エンティティのタグ付け」を参照してください。
9. [Next: Review] (次へ: レビュー) を選択します。
10. ユーザー設定を確認します。変更が必要な場合は、[戻る] を選択してユーザーのグループまたはポリシーを変更します。
11. ユーザーの設定が正しいことを確認したら、[ユーザーの作成] を選択します。

    ユーザーが作成され、作成されたユーザーに Lightsail へのアクセスが付与されます。ユーザーの Lightsail アクセスを取り消すには、Lightsail アクセスグループからユーザーを削除します。詳細については、IAM ドキュメントの「IAM グループへのユーザーの追加と削除」を参照してください。
12. ユーザーの認証情報を取得するには、以下のオプションを選択します。

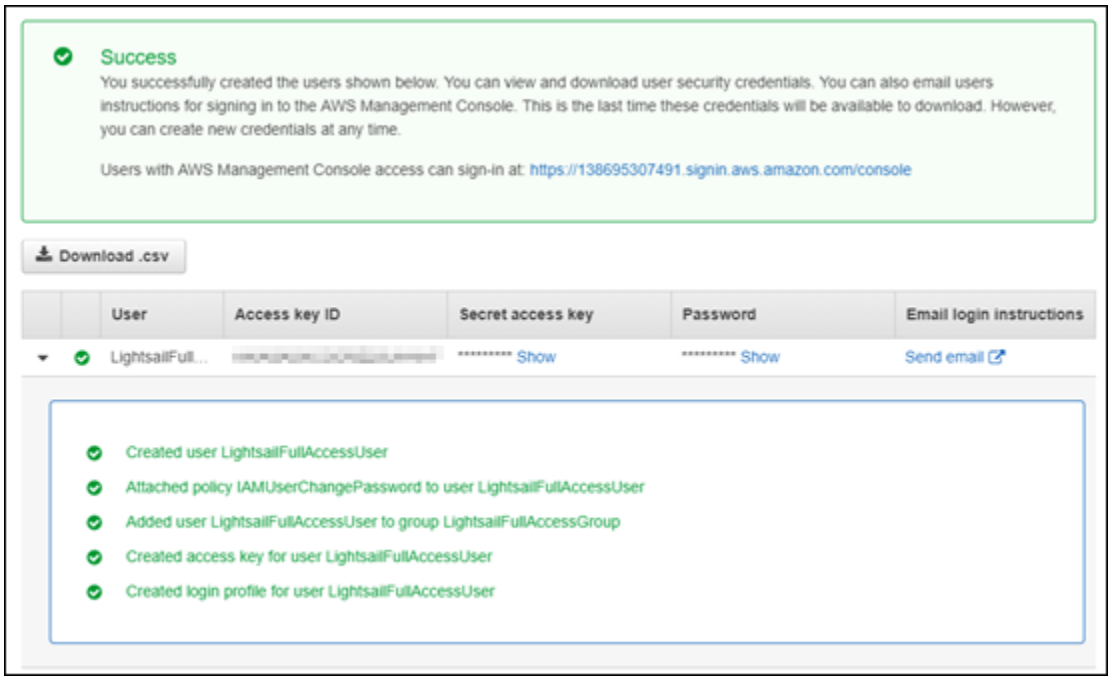
    a. [.csv のダウンロード] を選択して、ユーザー名、パスワード、アクセスキー ID、シークレットアクセスキー、および AWS コンソールのユーザーアカウントへのログインリンクを含んだファイルをダウンロードします。

b. [シークレットアクセスキー] で [表示] を選択すると、プログラム (AWS API、CLI、SDK、その他の開発ツール) を使用して Lightsail にアクセスするのに使用できるアクセスキーが表示されます。

> **Important**
>
> シークレットアクセスキーを表示またはダウンロードできるのはこのときだけです。AWS API を使用する前に、ユーザーにこの情報を提供する必要があります。ユーザーの新しいアクセスキー ID とシークレットアクセスキーは、安全な場所に保存してください。このステップを行った後に、シークレットキーに再度アクセスすることはできません。

c. ユーザーのパスワードが IAM によって生成されている場合、[パスワード] で [表示] を選択するとユーザーのパスワードが表示されます。ユーザーが初回サインインできるように、ユーザーにパスワードを提供する必要があります。

d. [E メールを送信する] を選択すると、Lightsail へのアクセス許可が付与されたことを知らせる E メールがユーザーに送られます。



Amazon Lightsail での更新管理

Amazon Web Services (AWS) Amazon Lightsail、およびサードパーティアプリケーションベンダーは、Lightsail AWS で利用可能なインスタンスイメージ (ブループリント とも呼ばれます) を定期的に

更新し、パッチを適用します。また、インスタンスを作成した後は、Lightsail ではインスタンス上のオペレーティングシステムやアプリケーションの更新やパッチ適用を行わないでください。Lightsail は、Lightsail コンテナサービスで設定したオペレーティングシステムやソフトウェアの更新や修正を行いません。したがって、Amazon Lightsail インスタンスのオペレーティングシステムやアプリケーションに対するパッチ適用やそれらの更新およびセキュリティ確保を定期的に行うことが推奨されます。詳細については、AWS 責任共有モデルを参照してください。

インスタンスブループリントソフトウェアのサポート

以下の Amazon Lightsail プラットフォームとブループリントのリストは、各ベンダーのサポートページにリンクしています。そこで、ハウツーガイド、オペレーティングシステムとアプリケーションを最新の状態に保つなどの情報を表示できます。アプリケーションベンダーが提供している、自動更新サービスまたは推奨更新インストールプロセスを使用することもできます。

Windows

- Windows Server 2022、Windows Server 2019、Windows Server 2016、Windows Server 2012 R2
- Microsoft SQL Server

Linux および Unix - オペレーティングシステムのみ

- Amazon Linux 2023
- Amazon Linux 2
- Ubuntu
- Debian
- FreeBSD
- openSUSE
- CentOS

Linux および Unix - オペレーティングシステムとアプリケーション

- Ubuntu の Plesk ホスティングスタック
- cPanel & WHM for Linux
- WordPress
- WordPress マルチサイト

- LAMP (PHP 8)
- Node.js
- Joomla!
- Magento
- MEAN
- Drupal
- GitLab CE
- Redmine
- Nginx
- Ghost
- Django
- PrestaShop

Amazon Lightsail のコンプライアンス検証

AWS では、コンプライアンスに役立つ、次のリソースを提供しています。

- セキュリティとコンプライアンスのクイックスタートガイド - これらのデプロイガイドでは、アーキテクチャ上の考慮事項について説明し、機密性とコンプライアンスに焦点を当てたベースライン環境を AWS にデプロイするためのステップが示されています。
- AWS コンプライアンスのリソース - ワークブックとお客様の業界や所在地に適用される場合があるガイドのコレクション。
- AWS Config デベロッパーガイドのルールでのリソースの評価 - AWS Config サービスでは、自社のプラクティス、業界ガイドライン、および規制に対するリソースの設定の準拠状態を評価します。
- AWS Security Hub - この AWS サービスは、AWS 内でのユーザーのセキュリティ状態に関する包括的な見解を提供し、業界のセキュリティスタンダード、およびベストプラクティスに対するコンプライアンスを確認するために役立ちます。

Amazon Lightsail リソースのモニタリング

Amazon Lightsail はインスタンス、データベース、ディストリビューション、ロードバランサー、コンテナサービス、およびバケットのメトリクスデータをチェック、収集し、それぞれのパフォーマンスをモニタリングします。時間の経過とともにベースラインを確立し、リソースのパフォーマンスに関する異常や問題をより簡単に検出できるようにアラームを設定できます。

Amazon Lightsail は、インスタンス、データベース、コンテンツ配信ネットワーク (CDN) ディストリビューション、ロードバランサー、コンテナサービス、バケットのメトリクスデータをレポートします。このデータは、Lightsail コンソールで表示およびモニタリングできます。モニタリングは、リソースの信頼性、可用性、パフォーマンスを維持する上で重要な部分です。リソースから定期的にメトリクスデータをモニタリングして収集し、マルチポイント障害が発生した場合に、より簡単にデバッグできるようにします。

目次

リソースを効果的にモニタリングする

環境内の通常のリソースパフォーマンスのベースラインを確立する必要があります。さまざまな時間帯に、さまざまな負荷条件でパフォーマンスを測定します。リソースをモニタリングするときは、時間の経過に伴うリソースのパフォーマンスの履歴を書き留めて記録する必要があります。収集した履歴データに対して、リソースの現在のパフォーマンスを比較します。これにより、通常のパフォーマンスパターンとパフォーマンスの異常を特定し、それらに対処するための方法を考案することができます。

たとえば、インスタンスの CPU 使用率、ネットワーク使用率、ステータスチェックをモニタリングできます。確立したベースラインからパフォーマンスが外れた場合は、インスタンスの再設定または最適化を行って CPU 使用率の抑制、またはネットワークトラフィックの低減を行うことが必要な場合があります。インスタンスが CPU 使用率のしきい値を超えて動作し続ける場合は、インスタンスのより大きなプランに切り替えることができます (たとえば、$3.50 USD/月プランではなく $5 USD/月プランを使用します)。インスタンスの新しいスナップショットを作成し、大きなプランを使用してスナップショットから新しいインスタンスを作成することで、より大きなプランに切り替えることができます。

ベースラインを確立したら、リソースが指定のしきい値を超えたときに通知するように Lightsail コンソールでアラームを設定できます。詳細については、「通知」および「アラーム」を参照してください。

メトリクスの概念と用語

次の用語と概念は、Lightsail でのメトリクスの使用をよりよく理解するのに役立ちます。

メトリクス

メトリクスは、時間順に並んだ一連のデータポイントを表します。メトリクスはモニタリング対象の変数と考え、データポイントは時間の経過と共に変数の値を表します。メトリクスは、名前によって一意に定義されます。たとえば、Lightsail によって提供される一部のインスタンスメトリクスには、CPU 使用率 (`CPUUtilization`)、着信ネットワークトラフィック (`NetworkIn`)、発信ネットワークトラフィック (`NetworkOut`) などがあります。Lightsail で使用可能なすべてのリソースメトリクスの詳細については、「Lightsail でのメトリクス」を参照してください。

メトリクスの保持

期間が 60 秒 (1 分の解像度) のデータポイントは、15 日間使用できます。期間が 300 秒 (5 分の解像度) のデータポイントは、63 日間使用できます。期間が 3600 秒 (1 時間の解像度) のデータポイントは、455 日 (15 か月) 間使用できます。

最初は短い期間で発行されるデータポイントは、長期的なストレージのため一緒に集計されます。たとえば、1 分の精度を持つデータポイントは、1 分の解像度で 15 日間使用できます。15 日を過ぎてもこのデータはまだ利用できますが、集計され、5 分の解像度のみで取得可能になります。63 日を過ぎるとこのデータはさらに集計され、1 時間の解像度のみで利用できます。これらの期間より長くメトリクスを利用する必要がある場合は、Lightsail API、AWS Command Line Interface (AWS CLI)、SDK を使用して、データポイントをオフラインまたは異なるストレージに取得できます。

詳細については、「Lightsail API リファレンス」の「GetInstanceMetricData」、「GetBucketMetricData」、「GetLoadBalancerMetricData」、「GetDistributionMetricData」および「GetRelationalDatabaseMetricData」を参照してください。

統計

メトリクス統計は、一定期間にわたってデータを集計する手段です。統計情報の例としては、`Average`、`Sum`、`Maximum` などがあります。たとえば、`Average` 統計を使用してインスタン

スの CPU 使用率メトリクスデータを平均化し、Sum 統計を使用してデータベース接続を追加したり、Maximum 統計を使用してロードバランサーの最大応答時間を取得したりできます。

利用可能なメトリクスの統計一覧は、「Lightsail API リファレンス」の「GetInstanceMetricData の統計」、「GetBucketMetricData の統計」、「GetLoadBalancerMetricData の統計」、「GetDistributionMetricData の統計」および「GetRelationalDatabaseMetricData の統計」を参照してください。

単位

各統計には、測定単位があります。単位の例は、Bytes、Seconds、Count、Percent などです。ユニットの全一覧は、Lightsail API リファレンスの GetInstanceMetricData のユニット、GetLoadBalancerMetricData のユニット、GetDistributionMetricData のユニットおよび GetRelationalDatabaseMetricData のユニットを参照してください。

期間

期間とは、返されたデータポイントの粒度を示す特定のデータポイントに関連付けられた時間の長さです。各データポイントは、指定された期間に収集されたメトリクスデータの集約を表しています。期間は秒単位で定義され、期間の有効値は 60 秒 (1 分) と 300 秒 (5 分) の倍数です。

Lightsail API を使用してデータポイントを取得する場合、期間、開始時刻、終了時刻を指定できます。これらのパラメータでは、データポイントに関連する全体の時間長を決定します。Lightsail は、1 分単位または 5 分単位でメトリックデータを報告します。したがって、期間は 60 秒と 300 秒の倍数で指定する必要があります。開始時刻と終了時刻に指定した値により、Lightsail により返される期間が決まります。10 分区切りで集約された統計を取得する場合は、期間を 600 に指定します。1 時間分の集約された統計の場合は、期間を 3600 などに設定します。

期間は、Lightsail アラームにとっても重要です。Lightsail はアラームのデータポイントを 5 分ごとに評価し、アラームの各データポイントは 5 分間の集約データを表します。特定のメトリクスをモニタリングするアラームを作成したら、そのメトリクスと指定したしきい値を比較するよう Lightsail に依頼していることになります。ユーザーは、Lightsail がその比較を行う方法を広範囲に制御できます。比較を行う期間を指定し、結論に達するために使用する評価期間の数を指定することもできます。詳細については、「アラーム」を参照してください。

Alarms

アラームは、指定した期間に 1 つのメトリクスをモニタリングし、メトリクスが指定したしきい値を超えたときに通知します。通知は、Lightsail コンソールに表示されるバナー、指定したメールアド

レスに送信された メール、指定した携帯電話番号に送信された SMS テキストメッセージです。詳細については、「アラーム」を参照してください。

Lightsail で使用可能なメトリクス

インスタンスメトリクス

次のインスタンスメトリクスを使用できます。詳細については、「Amazon Lightsail でインスタンスのメトリクスを表示する」を参照してください。

- CPU 使用率 (**CPUUtilization**) — 割り当てられたコンピューティングユニットのうち、現在インスタンス上で使用されているものの割合。このメトリクスは、インスタンスでアプリケーションを実行するための処理能力を識別します。インスタンスがフルプロセッサコアに割り当てられていない場合に、オペレーティングシステムのツールが Lightsail よりも低い割合を示す場合があります。

  Lightsail コンソールでインスタンスの CPU 使用率メトリクスグラフを表示すると、持続可能なゾーンとバースト可能なゾーンが表示されます。これらのゾーンの意味の詳細については、「CPU 使用率の持続可能なゾーンとバースト可能なゾーン」を参照してください。
- バーストキャパシティ (**BurstCapacityTime**) および割合 (**BurstCapacityPercentage**) — バーストキャパシティ分数は、インスタンスが CPU 使用率 100% でバーストできる時間を表します。バーストキャパシティの割合は、インスタンスで利用できる CPU パフォーマンスの割合です。インスタンスはバースト容量を継続的に消費し、蓄積します。インスタンスが 100% の CPU 使用率で動作しているときにのみ、バーストキャパシティの分数がフルレートで消費されます。インスタンスのバースト容量の詳細については、「Amazon Lightsail でのインスタンスのバースト容量の表示」を参照してください。
- 受信ネットワークトラフィック (**NetworkIn**) — すべてのネットワークインターフェイスでの、このインスタンスによって受信されたバイト数。このメトリクスは、1 つのインスタンスへの着信ネットワークトラフィックの量を表しています。報告された数は、期間中に受信されたバイト数です。このメトリクスは 5 分間隔でレポートされるため、レポートされた数を 300 で割ると、バイト/秒を算出できます。
- 送信ネットワークトラフィック (**NetworkOut**) — すべてのネットワークインターフェイスでの、このインスタンスから送信されたバイト数。このメトリクスは、1 つのインスタンスからの発信ネットワークトラフィックの量を表しています。報告された数は、期間中に送信されたバイト数です。このメトリクスは 5 分間隔でレポートされるため、レポートされた数を 300 で割ると、バイト/秒を算出できます。

- ステータスチェックの失敗 (**StatusCheckFailed**) — インスタンスが、インスタンスステータスチェックとシステムステータスチェックの両方に合格したか失敗したかをレポートします。このメトリクスは 0 (合格) または 1 (失敗) となります。このメトリクスは、1 分間の頻度で利用できます。
- インスタンスステータスチェックの失敗 (**StatusCheckFailed_Instance**) — インスタンスがインスタンスステータスチェックに合格したか、失敗したかをレポートします。このメトリクスは 0 (合格) または 1 (失敗) となります。このメトリクスは、1 分間の頻度で利用できます。
- ステータスチェックの失敗 (**StatusCheckFailed_System**) — インスタンスが、システムステータスチェックに合格したか失敗したかをレポートします。このメトリクスは 0 (合格) または 1 (失敗) となります。このメトリクスは、1 分間の頻度で利用できます。
- トークンメタデータなしのリクエスト (**MetadataNoToken**) — トークンなしでインスタンスのメタデータサービスに正常にアクセスした回数。このメトリクスにより、トークンを使用しない Instance Metadata Service バージョン 1 を使用してインスタンスメタデータにアクセスするプロセスがあるかどうかがわかります。すべてのリクエストがトークン支援のセッション (Instance Metadata Service バージョン 2 など) を使用している場合、値は 0 になります。詳細については、「Amazon Lightsail のインスタンスメタデータとユーザーデータ」を参照してください。

データベースメトリクス

次のデータベースメトリクスを使用できます。詳細については、「Amazon Lightsail でのデータベースメトリクスの表示」を参照してください。

- CPU 使用率 (**CPUUtilization**) — データベースで現在使用されている CPU 使用率の割合。
- データベース接続 (**DatabaseConnections**) — 使用中のデータベース接続の数。
- ディスクのキューの深度 (**DiskQueueDepth**) — ディスクへのアクセスを待機している未処理の IO (読み取り/書き込みリクエスト) の数。
- 空きストレージ容量 (**FreeStorageSpace**) — 使用可能なストレージの容量。
- ネットワーク受信スループット (**NetworkReceiveThroughput**) — モニタリングとレプリケーションに使用する顧客データベーストラフィックと AWS トラフィックの両方を含む、データベースの受信ネットワークトラフィック。
- ネットワーク送信スループット (**NetworkTransmitThroughput**) — モニタリングとレプリケーションに使用する顧客データベーストラフィックと AWS トラフィックの両方を含む、データベースの送信ネットワークトラフィック。

ディストリビューションメトリクス

以下のディストリビューションメトリクスが利用可能です。詳細については、「Amazon Lightsail のディストリビューションメトリクスの表示」を参照してください。

- リクエスト (**Requests**) — すべての HTTP メソッド、および HTTP と HTTPS 両方のリクエストについて、ディストリビューションが受信したビューワーリクエストの総数。
- アップロードされたバイト数 (**BytesUploaded**) — POST リクエストと PUT リクエストを使用して、ディストリビューションによってオリジンにアップロードされたバイト数。
- ダウンロードされたバイト数 (**BytesDownloaded**) — GET リクエスト、HEAD リクエスト、および OPTIONS リクエストに対してビューワーがダウンロードしたバイト数。
- トータルエラー率 (**TotalErrorRate**) — レスポンスの HTTP ステータスコードが 4xx または 5xx であったすべてのビューワーリクエストの割合 (%)。
- HTTP 4xx トータルエラー率 (**4xxErrorRate**) — レスポンスの HTTP ステータスコードが 4xx であったすべてのビューワーリクエストの割合 (%)。このような場合、クライアントまたはクライアントビューワーでエラーが発生した可能性があります。たとえば、ステータスコード 404 (Not Found) は、クライアントが、検出できないオブジェクトをリクエストしたことを意味します。
- HTTP 5xx トータルエラー率 (**5xxErrorRate**) — レスポンスの HTTP ステータスコードが 5xx であったすべてのビューワーリクエストの割合 (%)。このような場合、オリジンサーバーはリクエストを満たしませんでした。たとえば、ステータスコード 503 (Service Unavailable) は、オリジンサーバーが現在利用できないことを意味します。

ロードバランサーのメトリクス

次のロードバランサーメトリクスを使用できます。詳細については、「Amazon Lightsail でのロードバランサーメトリクスの表示」を参照してください。

- 正常ホスト数 (**HealthyHostCount**) — 正常と見なされるターゲットインスタンスの数。
- 異常ホスト数 (**UnhealthyHostCount**) — 異常と見なされるターゲットインスタンスの数。
- ロードバランサー HTTP 4XX (HTTPCode_LB_4XX_Count) — ロードバランサーから発生した HTTP 4XX クライアントエラーコードの数。リクエストの形式が不正な場合、または不完全な場合は、クライアントエラーが生成されます。これらのリクエストは、ターゲットインスタンスによって受信されませんでした。この数には、ターゲットインスタンスによって生成される応答コードは含まれません。

- ロードバランサー HTTP 5XX (HTTPCode_LB_5XX_Count) — ロードバランサーから発生した HTTP 5XX サーバーのエラーコードの数。これには、ターゲットインスタンスによって生成される応答コードは含まれません。ロードバランサーにアタッチされている正常なインスタンスがない場合、またはリクエストレートがインスタンスやロードバランサーの容量を超える場合 (スピルオーバー)、このメトリクスが報告されます。
- インスタンス HTTP 2XX (**HTTPCode_Instance_2XX_Count**) — ターゲットインスタンスによって生成された HTTP 2XX 応答コードの数。これには、ロードバランサーによって生成される応答コードは含まれません。
- インスタンス HTTP 3XX (**HTTPCode_Instance_3XX_Count**) — ターゲットインスタンスによって生成された HTTP 3XX 応答コードの数。これには、ロードバランサーによって生成される応答コードは含まれません。
- インスタンス HTTP 4XX (**HTTPCode_Instance_4XX_Count**) — ターゲットインスタンスによって生成された HTTP 4XX 応答コードの数。これには、ロードバランサーによって生成される応答コードは含まれません。
- インスタンス HTTP 5XX (**HTTPCode_Instance_5XX_Count**) — ターゲットインスタンスによって生成された HTTP 5XX 応答コードの数。これには、ロードバランサーによって生成される応答コードは含まれません。
- インスタンスからの応答時間 (**InstanceResponseTime**) — ロードバランサーがリクエストを送信してから、ターゲットインスタンスからの応答を受信するまでの経過時間 (秒)。
- クライアント TLS ネゴシエーションエラー数 (**ClientTLSNegotiationErrorCount**) — クライアントにより開始され、ロードバランサーによって生成された TLS エラーのためにロードバランサーとのセッションを確立しなかった、TLS 接続の数。暗号化またはプロトコルの不一致が原因である場合があります。
- リクエストの数 (**RequestCount**) — IPv4 経由で処理されたリクエストの数。この数には、ロードバランサーのターゲットインスタンスによって生成されたレスポンスを含むリクエストのみが含まれます。
- 拒否された接続数 (**RejectedConnectionCount**) — ロードバランサーが接続の最大数に達したため、拒否された接続の数。

コンテナサービスのメトリクス

以下のコンテナサービスメトリクスが利用可能です。詳細については、「コンテナサービスメトリクスを表示する」を参照してください。

- CPU 使用率 (**CPUUtilization**) — コンテナサービスの全ノードで現在使用されているコンピューティングユニットの平均比率。このメトリクスは、コンテナサービス上のコンテナを実行するのに必要な処理能力を特定します。
- メモリ使用率 (**MemoryUtilization**) — コンテナサービスの全ノードで現在使用されているメモリの平均比率。このメトリクスは、コンテナサービス上のコンテナを実行するのに必要なメモリを特定します。

バケットメトリクス

次のバケットメトリクスが利用可能です。詳細については、「Amazon Lightsail のバケットメトリクスを表示」を参照してください。

- [バケットサイズ (**BucketSizeBytes**)] — バケットに保存されたデータの量。この値を計算するには、バケット内のすべてのオブジェクト (最新のオブジェクトと最新でないオブジェクトの両方) のサイズを合計します。これには、バケットに対するすべての不完全なマルチパートアップロードのすべてのパートのサイズも含めます。
- [オブジェクトの数 (**NumberOfObjects**)] — バケットに保存されたオブジェクトの総数。この値を計算するには、バケット内のすべてのオブジェクト (最新のオブジェクトと最新でないオブジェクトの両方) と、バケットに対するすべての不完全なマルチパートアップロードの合計パート数をカウントします。

> **Note**
>
> バケットが空の場合、バケットメトリックスデータはレポートされません。

Lightsail リソースヘルスのメトリック

さまざまな期間で以下の Amazon Lightsail のリソースメトリクスを表示できます。Lightsail のリソースメトリクスの詳細については、「リソースメトリクス」を参照してください。

インスタンスメトリクス

次のインスタンスメトリクスを使用できます。詳細については、「Amazon Lightsail でインスタンスのメトリクスを表示する」を参照してください。

- CPU 使用率 (**CPUUtilization**) — 割り当てられたコンピューティングユニットのうち、現在インスタンス上で使用されているものの割合。このメトリクスは、インスタンスでアプリケーションを実行するための処理能力を識別します。インスタンスがフルプロセッサコアに割り当てられていない場合に、オペレーティングシステムのツールが Lightsail よりも低い割合を示す場合があります。

  Lightsail コンソールでインスタンスの CPU 使用率メトリクスグラフを表示すると、持続可能なゾーンとバースト可能なゾーンが表示されます。これらのゾーンの意味の詳細については、「CPU 使用率の持続可能なゾーンとバースト可能なゾーン」を参照してください。
- バーストキャパシティ (**BurstCapacityTime**) および割合 (**BurstCapacityPercentage**) — バーストキャパシティ分数は、インスタンスが CPU 使用率 100% でバーストできる時間を表します。バーストキャパシティの割合は、インスタンスで利用できる CPU パフォーマンスの割合です。インスタンスはバースト容量を継続的に消費し、蓄積します。インスタンスが 100% の CPU 使用率で動作しているときにのみ、バーストキャパシティの分数がフルレートで消費されます。インスタンスのバーストキャパシティの詳細については、「インスタンスのバーストキャパシティの表示」を参照してください。
- 受信ネットワークトラフィック (**NetworkIn**) — すべてのネットワークインターフェイスでの、このインスタンスによって受信されたバイト数。このメトリクスは、1 つのインスタンスへの着信ネットワークトラフィックの量を表しています。報告された数は、期間中に受信されたバイト数です。このメトリクスは 5 分間隔でレポートされるため、レポートされた数を 300 で割ると、バイト/秒を算出できます。
- 送信ネットワークトラフィック (**NetworkOut**) — すべてのネットワークインターフェイスでの、このインスタンスから送信されたバイト数。このメトリクスは、1 つのインスタンスからの発信ネットワークトラフィックの量を表しています。報告された数は、期間中に送信されたバイト数です。このメトリクスは 5 分間隔でレポートされるため、レポートされた数を 300 で割ると、バイト/秒を算出できます。
- ステータスチェックの失敗 (**StatusCheckFailed**) — インスタンスが、インスタンスステータスチェックとシステムステータスチェックの両方に合格したか失敗したかをレポートします。このメトリクスは 0 (合格) または 1 (失敗) となります。このメトリクスは、1 分間の頻度で利用できます。
- インスタンスステータスチェックの失敗 (**StatusCheckFailed_Instance**) — インスタンスがインスタンスステータスチェックに合格したか、失敗したかをレポートします。このメトリクスは 0 (合格) または 1 (失敗) となります。このメトリクスは、1 分間の頻度で利用できます。
- ステータスチェックの失敗 (**StatusCheckFailed_System**) — インスタンスが、システムステータスチェックに合格したか失敗したかをレポートします。このメトリクスは 0 (合格) または 1 (失敗) となります。このメトリクスは、1 分間の頻度で利用できます。

- ステータスチェックの失敗 (**StatusCheckFailed_System**) — インスタンスが、システムステータスチェックに合格したか失敗したかをレポートします。このメトリクスは 0 (合格) または 1 (失敗) となります。このメトリクスは、1 分間の頻度で利用できます。
- トークンメタデータなしのリクエスト (**MetadataNoToken**) — トークンなしでインスタンスのメタデータサービスに正常にアクセスした回数。このメトリクスにより、トークンを使用しない Instance Metadata Service バージョン 1 を使用してインスタンスメタデータにアクセスするプロセスがあるかどうかがわかります。すべてのリクエストが、Instance Metadata Service バージョン 2 あどのトークン支援のセッションを使用している場合、値は 0 になります。詳細については、「インスタンスメタデータとユーザーデータ」を参照してください。

データベースメトリクス

次のデータベースメトリクスを使用できます。詳細については、「データベースメトリクスの表示」を参照してください。

- CPU 使用率 (**CPUUtilization**) — データベースで現在使用されている CPU 使用率の割合。
- データベース接続 (**DatabaseConnections**) — 使用中のデータベース接続の数。
- ディスクのキューの深度 (**DiskQueueDepth**) — ディスクへのアクセスを待機している未処理の IO (読み取り/書き込みリクエスト) の数。
- 空きストレージ容量 (**FreeStorageSpace**) — 使用可能なストレージの容量。
- ネットワーク受信スループット (**NetworkReceiveThroughput**) — モニタリングとレプリケーションに使用する顧客データベーストラフィックと AWS トラフィックの両方を含む、データベースの受信ネットワークトラフィック。
- ネットワーク送信スループット (**NetworkTransmitThroughput**) — モニタリングとレプリケーションに使用する顧客データベーストラフィックと AWS トラフィックの両方を含む、データベースの送信ネットワークトラフィック。

ディストリビューションメトリクス

以下のディストリビューションメトリクスが利用可能です。詳細については、「Amazon Lightsail のディストリビューションメトリクスの表示」を参照してください。

- リクエスト — すべての HTTP メソッド、および HTTP と HTTPS 両方のリクエストについて、ディストリビューションが受信したビューワーリクエストの総数。

- アップロードされたバイト数 — POST リクエストと PUT リクエストを使用して、ディストリビューションによってオリジンにアップロードされたバイト数。
- ダウンロードされたバイト数 — GET リクエスト、HEAD リクエスト、および OPTIONS リクエストに対してビューワーがダウンロードしたバイト数。
- トータルエラー率 — レスポンスの HTTP ステータスコードが 4xx または 5xx であったすべてのビューワーリクエストの割合 (%)。
- HTTP 4xx トータルエラー率 — レスポンスの HTTP ステータスコードが 4xx であったすべてのビューワーリクエストの割合 (%)。このような場合、クライアントまたはクライアントビューワーでエラーが発生した可能性があります。たとえば、ステータスコード 404 (Not Found) は、クライアントが、検出できないオブジェクトをリクエストしたことを意味します。
- HTTP 5xx トータルエラー率 — レスポンスの HTTP ステータスコードが 5xx であったすべてのビューワー リクエストの割合 (%)。このような場合、オリジンサーバーはリクエストを満たしませんでした。たとえば、ステータスコード 503 (Service Unavailable) は、オリジンサーバーが現在利用できないことを意味します。

ロードバランサーのメトリクス

次のロードバランサーメトリクスを使用できます。詳細については、「ロードバランサーメトリクスの表示」を参照してください。

- 正常ホスト数 (**HealthyHostCount**) — 正常と見なされるターゲットインスタンスの数。
- 異常ホスト数 (**UnhealthyHostCount**) — 異常と見なされるターゲットインスタンスの数。
- ロードバランサー HTTP 4XX (HTTPCode_LB_4XX_Count) — ロードバランサーから発生した HTTP 4XX クライアントエラーコードの数。リクエストの形式が不正な場合、または不完全な場合は、クライアントエラーが生成されます。これらのリクエストは、ターゲットインスタンスによって受信されませんでした。この数には、ターゲットインスタンスによって生成される応答コードは含まれません。
- ロードバランサー HTTP 5XX (HTTPCode_LB_5XX_Count) — ロードバランサーから発生した HTTP 5XX サーバーのエラーコードの数。これには、ターゲットインスタンスによって生成される応答コードは含まれません。ロードバランサーにアタッチされている正常なインスタンスがない場合、またはリクエストレートがインスタンスやロードバランサーの容量を超える場合 (スピルオーバー)、このメトリクスが報告されます。
- インスタンス HTTP 2XX (**HTTPCode_Instance_2XX_Count**) — ターゲットインスタンスによって生成された HTTP 2XX 応答コードの数。これには、ロードバランサーによって生成される応答コードは含まれません。

- インスタンス HTTP 3XX (**HTTPCode_Instance_3XX_Count**) — ターゲットインスタンスによって生成された HTTP 3XX 応答コードの数。これには、ロードバランサーによって生成される応答コードは含まれません。
- インスタンス HTTP 4XX (**HTTPCode_Instance_4XX_Count**) — ターゲットインスタンスによって生成された HTTP 4XX 応答コードの数。これには、ロードバランサーによって生成される応答コードは含まれません。
- インスタンス HTTP 5XX (**HTTPCode_Instance_5XX_Count**) — ターゲットインスタンスによって生成された HTTP 5XX 応答コードの数。これには、ロードバランサーによって生成される応答コードは含まれません。
- インスタンスからの応答時間 (**InstanceResponseTime**) — ロードバランサーがリクエストを送信してから、ターゲットインスタンスからの応答を受信するまでの経過時間 (秒)。
- リクエストの数 (**RequestCount**) — IPv4 経由で処理されたリクエストの数。この数には、ロードバランサーのターゲットインスタンスによって生成されたレスポンスを含むリクエストのみが含まれます。
- クライアント TLS ネゴシエーションエラー数 (**ClientTLSNegotiationErrorCount**) — クライアントにより開始され、ロードバランサーによって生成された TLS エラーのためにロードバランサーとのセッションを確立しなかった、TLS 接続の数。暗号化またはプロトコルの不一致が原因である場合があります。
- 拒否された接続数 (**RejectedConnectionCount**) — ロードバランサーが接続の最大数に達したため、拒否された接続の数。

コンテナサービスのメトリクス

以下のコンテナサービスメトリクスが利用可能です。詳細については、「コンテナサービスメトリクスを表示する」を参照してください。

- CPU 使用率 — コンテナサービスの全ノードで現在使用されているコンピューティングユニットの平均比率。このメトリクスは、コンテナサービス上のコンテナを実行するのに必要な処理能力を特定します。
- メモリ使用率 — コンテナサービスの全ノードで現在使用されているメモリの平均比率。このメトリクスは、コンテナサービス上のコンテナを実行するのに必要なメモリを特定します。

バケットメトリクス

次のバケットメトリクスが利用可能です。詳細については、「バケットメトリクスを表示」を参照してください。

- [バケットサイズ] — バケットに保存されたデータの量。この値を計算するには、バケット内のすべてのオブジェクト (最新のオブジェクトと最新でないオブジェクトの両方) のサイズを合計します。これには、バケットに対するすべての不完全なマルチパートアップロードのすべてのパートのサイズも含めます。
- オブジェクトの数 — バケットに保存されたオブジェクトの総数。この値を計算するには、バケット内のすべてのオブジェクト (最新のオブジェクトと最新でないオブジェクトの両方) と、バケットに対するすべての不完全なマルチパートアップロードの合計パート数をカウントします。

> **Note**
>
> バケットが空の場合、バケットメトリックスデータはレポートされません。

トピック

Lightsail のメトリクスの通知

インスタンス、データベース、ロードバランサーないしコンテンツ配信ネットワーク (CDN) ディストリビューションのメトリクスが指定した値を超えた場合に通知を受けられるよう、Lightsail を設定することが可能す。通知は、Lightsail コンソールに表示されるバナー、指定したアドレスに送信される メール、指定した携帯電話番号に送信される SMS テキストメッセージの形式にすることができます。

通知を取得するには、リソースの 1 つのメトリクスを監視するアラームを設定する必要があります。たとえば、指定した時間内にインスタンスの発信ネットワークトラフィックが 500 KB を超えた

場合に通知するアラームを設定できます。詳細については、「メトリクスのアラーム」を参照してください。

アラームが作動すると、通知バナーが Lightsail コンソールに表示されます。メールおよび SMS テキストメッセージで通知を受けるには、リソースを監視する各 AWS リージョン で、メールアドレスと携帯電話番号を通知連絡先として追加する必要があります。詳細については、「通知連絡先を追加する」を参照してください。

> **Note**
>
> SMS テキストメッセージングは、Lightsail リソースを作成できるすべての AWS リージョンでサポートされているわけではありません。また、一部の国や地域にテキストメッセージを送信することはできません。詳細については、「通知連絡先を追加する」を参照してください。

通知が予定されているときに通知を受け取らない場合は、通知の連絡先が正しく設定されていることを確認するためにいくつかの点を確認してください。詳細については、「通知のトラブルシューティング」を参照してください。

通知の受信を停止するには、Lightsail から メールと携帯電話を削除します。詳細については、「メトリクスアラームを削除または無効化する」を参照してください。また、アラームを無効化または削除して、特定のアラームの通知の受信を停止することもできます。詳細については、「メトリクスアラームを削除または無効化する」を参照してください。

Lightsail インスタンスのバーストキャパシティを表示する

Amazon Lightsail は、ベースラインの CPU パフォーマンスを提供するインスタンスを提供しますが、必要に応じてベースラインを超えて追加の CPU パフォーマンスを一時的に提供することもできます。これをバーストといいます。ベースラインパフォーマンスとバースト機能は以下のインスタンスメトリクスによって制御されます。

- CPU 使用率 - 割り当てられたコンピューティングユニットのうち、現在インスタンスで使用されているものの割合。このメトリクスは、インスタンスでアプリケーションを実行するために使用される処理能力を表します。
- CPU バースト容量の割合 - インスタンスで利用できる CPU パフォーマンスの割合。
- CPU バースト容量の分数 - インスタンスが 100% の CPU 使用率でバーストできる時間長。

このガイドでは、これらのメトリクスをモニタリングしてインスタンスの可用性を最大化する方法を示します。

目次

ベースライン CPU パフォーマンスとバーストキャパシティの増加を理解する

Lightsail インスタンスは、1 時間あたりの CPU バーストキャパシティーの一定のレートを継続的に獲得 (ミリ秒レベルの解像度) します。これは、インスタンスの CPU 使用率が 0% を超えた場合にも消費されます。バースト容量が蓄積されるか消費されるかの会計処理もミリ秒レベルの細かさで行われるため、CPU バースト容量の過剰消費について心配する必要はありません。CPU の短期バーストでは、バースト容量のごく一部が使用されます。

インスタンスが使用している CPU リソースがベースラインパフォーマンスに必要な数よりも少ない場合 (アイドル時など)、未使用の CPU バースト容量が CPU バースト容量の割合と分数という形で蓄積されます。インスタンスがベースラインパフォーマンスレベルを超えてバーストする必要がある場合、蓄積された CPU バースト容量を消費します。インスタンスが蓄積した CPU バースト容量が多いほど、より高いパフォーマンスが必要なときに、ベースラインを超えてバーストできる時間が長くなります。

ベースライン CPU パフォーマンス

次のリストは、各 Lightsail インスタンスプランのパフォーマンスベースラインの概要を示しています。

- Linux または Unix 3.50 USD/月および Windows 8 USD/月 (2 vCPU、512 MB メモリ、30 GB のストレージ) のインスタンスプランには、5% の CPU 使用率パフォーマンス ベースラインが含まれています。
- Linux または Unix 5 USD/月および Windows 12 USD/月 (2 vCPUs、1 GB メモリ、40 GB ストレージ) のインスタンスプランには、10% の CPU 使用率パフォーマンス ベースラインが含まれています。

- Linux または Unix 10 USD/月および Windows 20 USD/月 (2 vCPUs、2 GB メモリ、60 GB ストレージ) のインスタンスプランには、20% の CPU 使用率パフォーマンス ベースラインが含まれています。
- Linux または Unix 20 USD/月および Windows 40 USD/月 (2 vCPUs、4 GB のメモリ、80 GB のストレージ) のインスタンスプランには、20% の CPU 使用率パフォーマンス ベースラインが含まれています。
- Linux または Unix 40 USD/月および Windows 70 USD/月 (2 vCPUs、8 GB のメモリ、160 GB のストレージ) のインスタンスプランには、30% の CPU 使用率パフォーマンス ベースラインが含まれています。
- Linux または Unix 80 USD/月および Windows 120 USD/月 (4 vCPUs、16 GB メモリ、320 GB ストレージ) のインスタンスプランには、40% の CPU 使用率パフォーマンス ベースラインが含まれています。
- Linux または Unix 160 USD/月および Windows 240 USD/月 (8 vCPUs、32 GB メモリ、640 GB ストレージ) のインスタンスプランには、40% の CPU 使用率パフォーマンス ベースラインが含まれています。

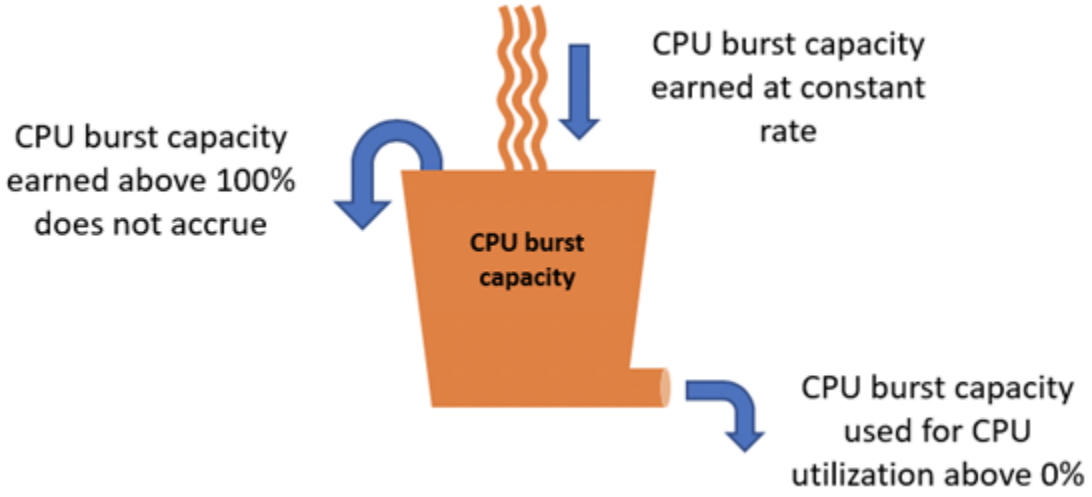
これらのパフォーマンス ベースラインは vCPU 単位です。Lightsail コンソールの CPU 使用率メトリクスグラフは、複数の vCPU を持つインスタンスの CPU 使用率とベースラインを平均化します。たとえば、40 USD/月のインスタンスには 2 つの vCPU が割り当てられ、平均の CPU 使用率ベースラインは 30% です。したがって、以下の場合が考えられます。

- 1 つの vCPU が 50% で動作し、もう 1 つの vCPU が 0% で動作すると、25% の平均 CPU 使用率がグラフに表示されます。これにより、インスタンスの CPU 使用率が 30% のベースラインを下回り、持続可能なゾーンに入ります。
- 1 つの vCPU が 30% で動作し、もう 1 つの vCPU が 20% で動作すると、25% の平均 CPU 使用率がグラフに表示されます。これにより、インスタンスの CPU 使用率が 30% のベースラインを下回り、持続可能なゾーンに入ります。
- 1 つの vCPU が 35% で動作し、もう 1 つの vCPU が 25% で動作すると、30% の平均 CPU 使用率がグラフに表示されます。これにより、インスタンスの CPU 使用率が 30% のベースラインになります。
- 1 つの vCPU が 100% で動作し、もう 1 つの vCPU が 90% で動作すると、95% の平均 CPU 使用率がグラフに表示されます。これにより、インスタンスの CPU 使用率が 30% のベースラインを超え、バースト可能なゾーンに入ります。

> **Note**
>
> 持続可能なゾーンとバースト可能なゾーンの詳細については、このガイドで後述される「インスタンスがバーストする時期の特定」を参照してください。

CPU バースト容量の蓄積

Lightsail インスタンスプランはすべて、1 時間あたりの CPU バーストキャパシティの 4.17% を蓄積します。蓄積できる 最大 CPU バーストキャパシティの割合は、24 時間で蓄積できる CPU バーストキャパシティの割合と同じです。CPU バーストキャパシティのパーセンテージが 100% に達すると、インスタンスは CPU バーストキャパシティの蓄積を停止します。



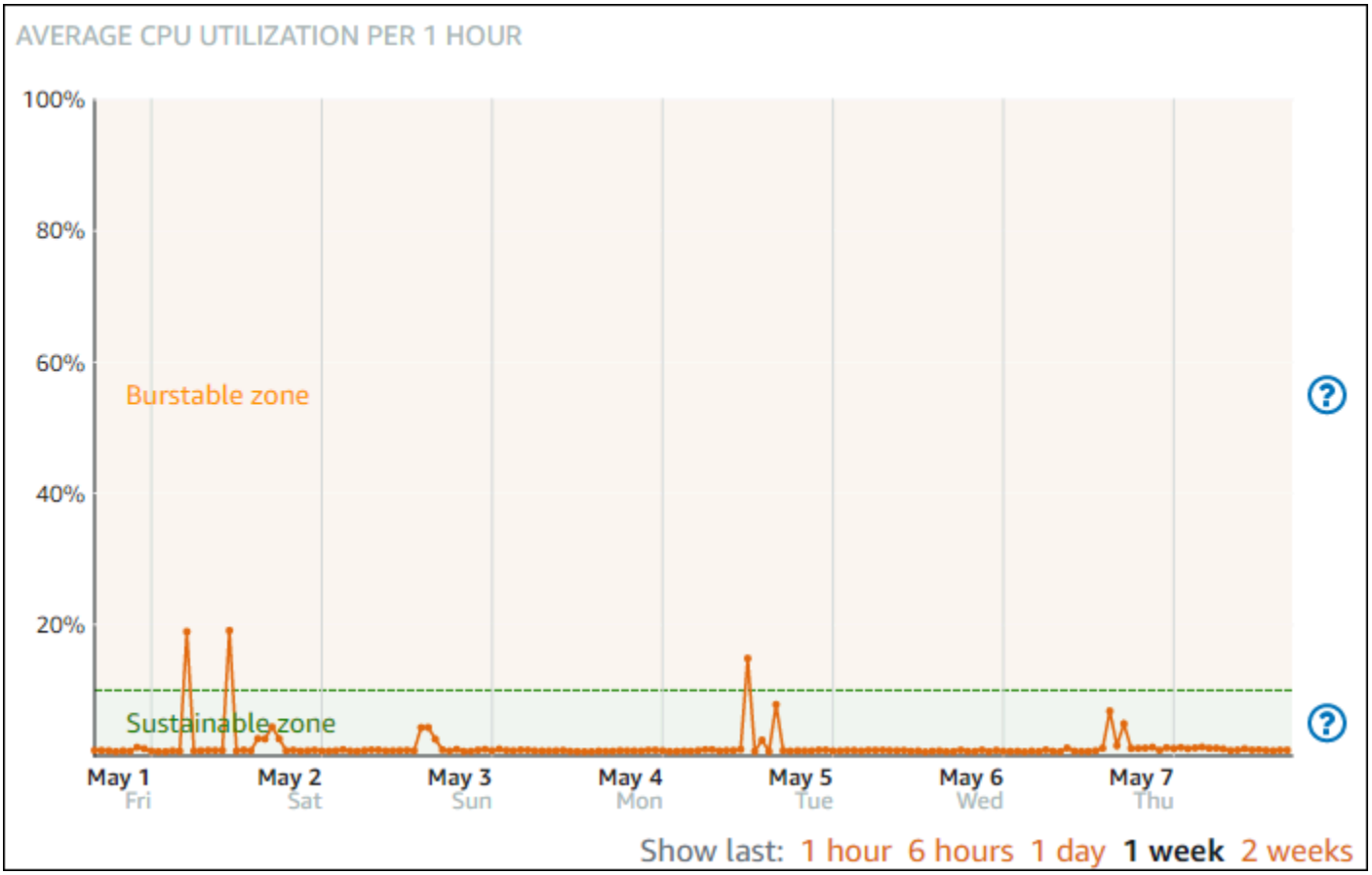
> **Important**
>
> 蓄積された CPU バーストキャパシティ
>
> - 2023 年 6 月 29 日より前に作成されたインスタンス - CPU バーストキャパシティは、インスタンスが停止しても保持されません。インスタンスを停止すると、蓄積されたバーストキャパシティーはすべて失われます。
> - 2023 年 6 月 29 日以降に作成されたインスタンス - CPU バーストキャパシティは、インスタンスの停止と起動の間に 7 日間保持されます。
> - 実行中のインスタンスで蓄積された CPU バースト容量に有効期限はありません。

Lightsail インスタンスは、起動時に追加の CPU バーストキャパシティを受け取ります。これは、起動 CPU バーストキャパシティと呼ばれます。起動 CPU バースト容量を使用すると、インスタンスは起動直後にバーストしてから、追加のバースト容量を蓄積します。起動 CPU バースト容量はバースト容量の制限にカウントされません。インスタンスが起動 CPU バースト容量を消費せず、バースト容量が蓄積されている間に 24 時間以上アイドル状態のままである場合、その CPU バースト容量の割合メトリクスグラフは 100% 以上として表示されます。

さらに、一部の Lightsail インスタンスは起動モードで起動します。これにより、バースト可能なインスタンスに通常存在するパフォーマンス制限の一部が一時的に削除されます。起動モードでは、インスタンスの全体的なパフォーマンスに影響を与えずに、リソースを大量に消費するスクリプトを起動時に実行できます。

インスタンスがバーストする時期を特定する

インスタンスの CPU 使用率メトリクスグラフに、持続可能なゾーンとバースト可能なゾーンが表示されます。次のCPU使用率メトリクス グラフの例では、インスタンスがLinuxまたはUnixベースの 5 USD/月のインスタンスプランを使用しているため、パフォーマンス ベースラインは10%です。

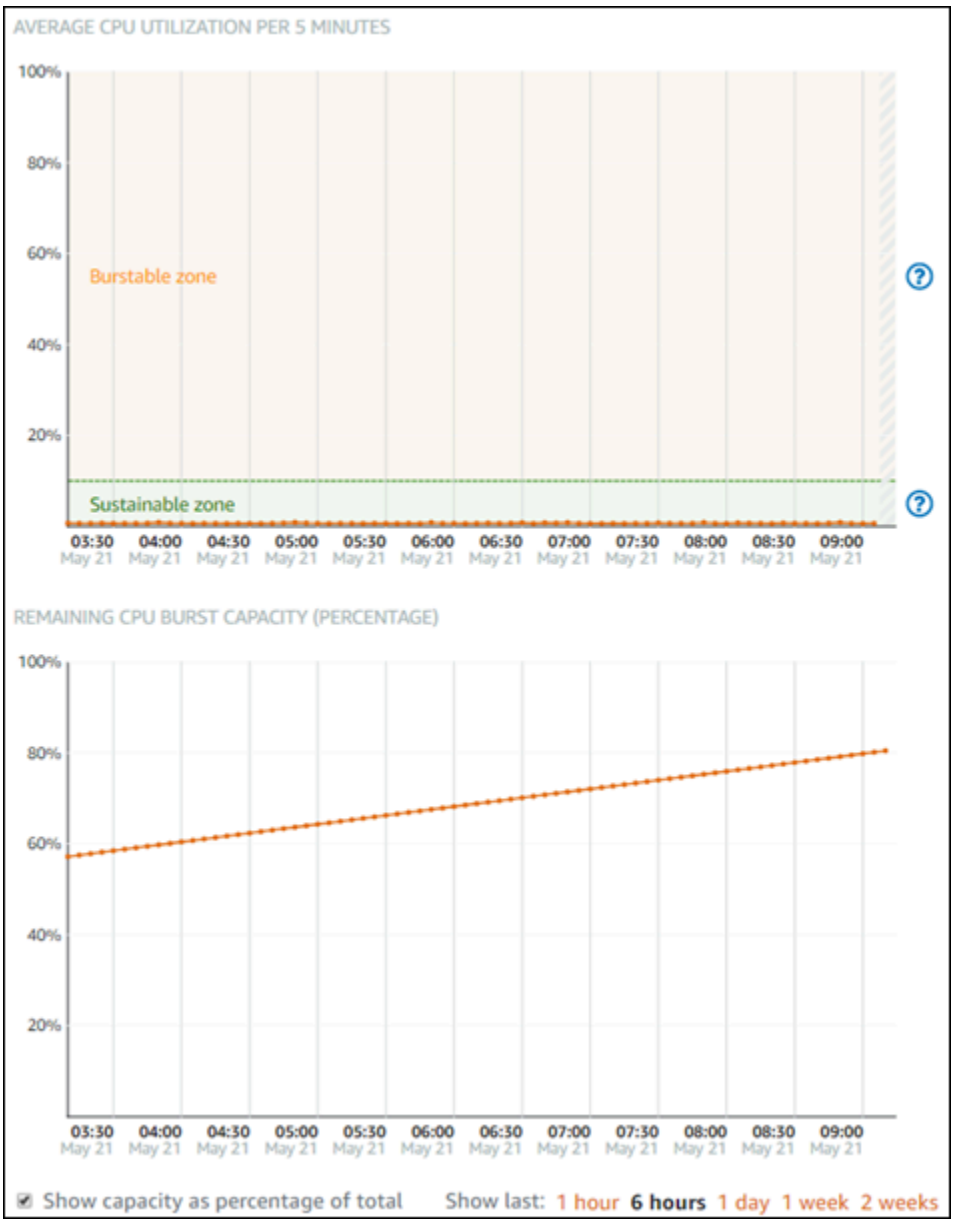


Lightsail インスタンスは、システムの動作に影響を与えずに、持続可能ゾーンで無期限に運用できます。コードのコンパイル、新しいソフトウェアのインストール、バッチジョブの実行、ピークの負荷リクエストの処理など、負荷が高い場合、インスタンスがバースト可能なゾーンで動作し始めることがあります。バーストな可能ゾーンで動作している間、インスタンスは大量の CPU サイクルを消費します。したがって、この領域では限られた期間しか作動できません。

インスタンスがバースト可能なゾーンで動作できる期間は、バースト可能なゾーンにどの程度入っているかによって異なります。バースト可能なゾーンの下限近くで動作しているインスタンスは、バースト可能なゾーンの上限近くで動作しているインスタンスよりも長い時間バーストできます。ただし、一定期間バースト可能なゾーンにあるインスタンスは、持続可能なゾーンで再び動作するまで、最終的にすべての CPU 容量を使い果たすことになります。したがって、このガイドの次のセクションで説明する残り CPU バースト容量もモニタリングすることが重要です。

CPU バーストキャパシティのモニタリング

Lightsail コンソールの CPU 概要ページには、使用可能な CPU バーストキャパシティと比較したインスタンスの CPU 使用率が表示されます。以下の CPU 概要の例では、インスタンスが持続可能なゾーンでベースラインを下回って継続的に動作しているため、CPU バースト容量の割合が増加しています。



残り CPU バースト容量のグラフビューは、CPU バースト容量の割合と分数で切り替えることができます。バースト可能なゾーンで動作しているとき、インスタンスはより多くの CPU バースト容量を消費します。CPU バースト容量の分数メトリクスは、インスタンスが 100% の CPU 使用率でバーストできる時間長です。インスタンスがバースト可能なゾーンで動作しているとき、CPU バースト容量 (割合) がインスタンスの現在の CPU 使用率と同じレートで消費されます。例えば、Linux または Unix ベース 5 USD/月のインスタンスの CPU 使用状況のベースラインは 10% で、1 時間あたり 6 分の CPU バーストキャパシティ (分数) が累積されます。したがって、以下の場合が考えられます。

- 60 分間、バースト可能なゾーンでの CPU 使用率が 100% のとき、その期間中、CPU バースト容量 (分数) が 100% のレートで消費されます。インスタンスは 60 分の CPU バーストキャパシティを消費し、6 分を累積するため、正味 54 分が消費されます。
- 60 分間、バースト可能なゾーンでの CPU 使用率が 50% のとき、その期間中、CPU バースト容量 (分数) が 50% のレートで消費されます。インスタンスは 30 分の CPU バーストキャパシティを消費し、6 分を累積するため、正味 24 分が消費されます。
- 60 分間、インスタンスのベースラインでの CPU 使用率が 10% のとき、その期間中、CPU バースト容量 (分数) が 10% のレートで消費されます。インスタンスは 6 分の CPU バースト容量を消費し、6 分を蓄積します。インスタンスがベースラインで動作しているとき、CPU バースト容量の分数は増減しません。
- 60 分間、持続可能なゾーンでの CPU 使用率が 5% のとき、その期間中、CPU バースト容量 (分数) が 5% のレートで消費されます。インスタンスは 3 分の CPU バーストキャパシティを消費し、6 分を累積するため、正味 3 分が累積されます。

あるいは、インスタンスは 60 分の CPU バースト容量を蓄積した場合、CPU 使用率 100% で 60 分間、50% で 120 分間、または 25% で 150 分間動作できます。

CPU 使用率が高い場合のトラブルシューティング

インスタンスがバースト可能なゾーンで頻繁にまたは長期間にわたって動作する場合、インスタンスはすべてのバースト容量を使用します。これは、インスタンスがプロビジョニング不足であることを示している可能性があります。また、サービスの実行頻度が高すぎるか、インスタンスで不要なソフトウェアが実行されていることを示している可能性もあります。

Linux/Unix インスタンスの top や Windows Server インスタンスのタスクマネージャーなどのツールを使用して、インスタンスのバーストの原因を調査します。これらのツールでは、インスタンスでリソースを消費しているサービスが表示されます。最も多くのリソースを消費しているサービスを特定し、インスタンスのワークロードに影響を与えずにそれらのサービスを無効にできるかどうかを決定します。サービスを無効にするか、ソフトウェアをアンインストールすることで、インスタンスのバーストを減らすことができ、インスタンスのサイズを大きくする必要がなくなる場合があります。

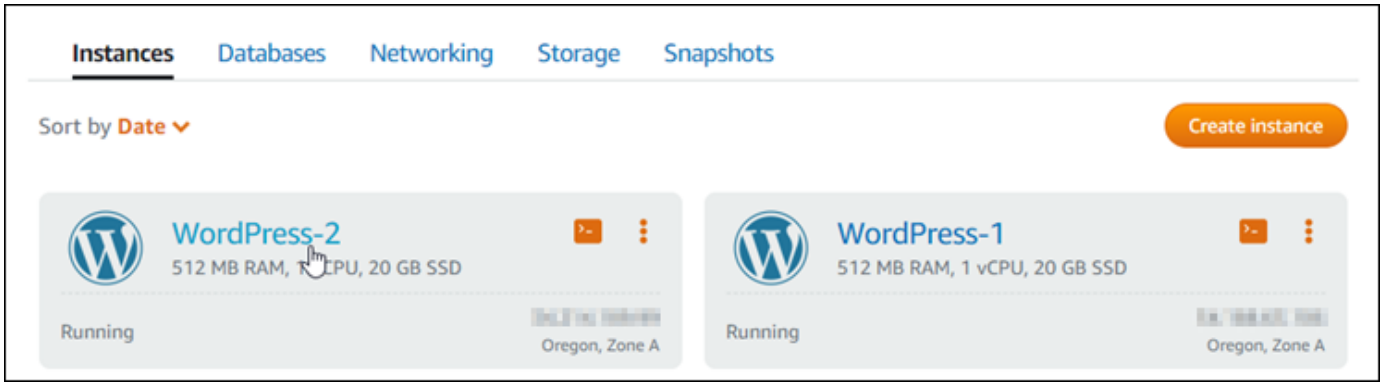
インスタンスが実際にプロビジョニング不足で、CPU 使用率を下げることができない場合は、処理能力を増やすことでバースト容量の消費を減らすことができます。これを行うには、インスタンスのスナップショットを作成し、より大きな Lightsail インスタンスプランを使用してスナップショットから新しいインスタンスを作成します。たとえば、以前のインスタンスで使用されていた Linux または Unix ベースの月額 10 USD プランの代わりに、新しいインスタンスで Linux または Unix ベースの月額 20 USD プランを使用します。新しいインスタンスが稼働中になったら、必要に応じてワー

クロードの DNS を変更して、古いインスタンスを新しいインスタンスと交換します。トラフィックが新しいインスタンスへルーティングされ始めたら、プロビジョニング不足の古いインスタンスを削除します。詳細については、「スナップショット」を参照してください。

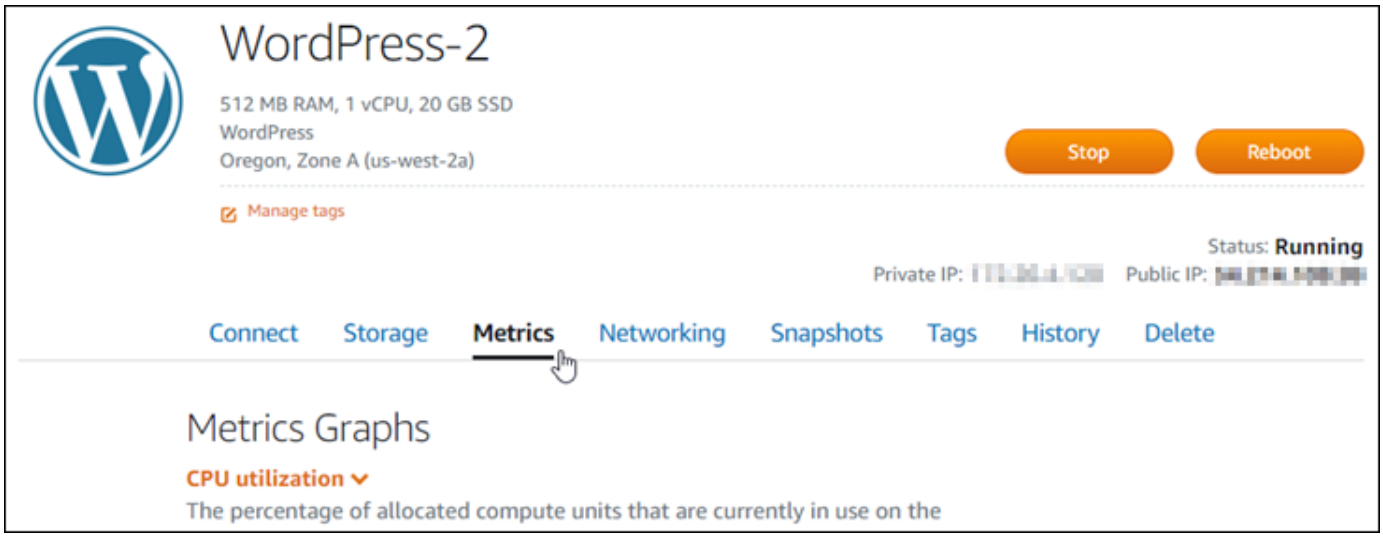
インスタンスのバーストキャパシティを表示する

CPU 概要ページにアクセスし、インスタンスの CPU 使用率と残り CPU バースト容量を表示するには、以下の手順を実行します。

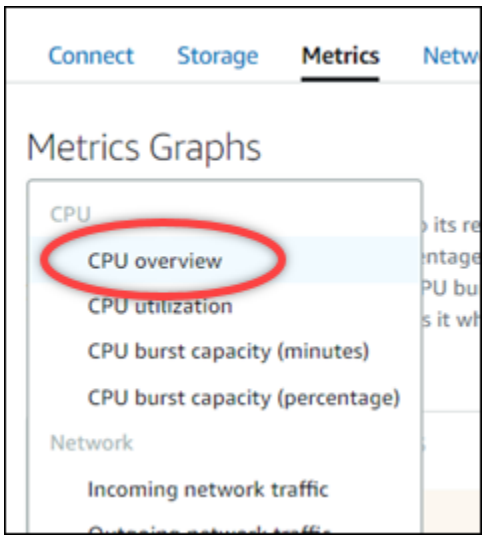
1. Lightsail コンソールにサインインします。
2. Lightsail ホームページで、インスタンスタブを選択します。
3. CPU 使用率とバースト容量を表示するインスタンスの名前を選択します。



4. インスタンス管理ページで [Metrics (メトリクス)] タブを選択します。



5. [Metrics graphs (メトリクスグラフ)] 見出しの下のドロップダウンメニューで [CPU overview (CPU 概要)] を選択します。



このページには、[5分間の平均 CPU 使用率] と [残りの CPU バーストキャパシティ] のグラフが表示されます。

> **Note**
>
> インスタンスを作成した後、しばらくの間、[残りの CPU バーストキャパシティ] のグラフに [起動モード] ゾーンが表示されることがあります。Lightsail インスタンスの中には、起動モードで起動するものがあります。これにより、バースト可能なインスタンスに通常存在するパフォーマンス制限の一部が一時的に削除されます。起動モードでは、インスタンスの全体的なパフォーマンスに影響を与えずに、リソースを大量に消費するスクリプトを起動時に実行できます。

6. メトリクスグラフでは、以下のアクションを実行できます。

   - バースト容量グラフで、[Show capacity as percentage of total (合計容量の割合として容量を表示)] を選択して、ビューを使用可能なバースト容量の分数から使用可能なバースト容量の割合に変更します。
   - グラフの表示を変更して、1 時間、6 時間、1 日、1 週間、2 週間のデータを表示します。
   - データポイント上にカーソルを置くと、そのデータポイントに関する詳細情報が表示されます。

- CPU 使用率とバースト容量が指定したしきい値を超えたときに通知するアラームを追加します。CPU 概要ページでアラームを追加することはできません。それらのアラームは、個々の CPU 使用率、CPU バースト容量の割合、CPU バースト容量の分数メトリクスグラフのページで追加する必要があります。詳細については、「アラーム」および「インスタンスのメトリクスアラームを作成する」を参照してください。

Lightsail のインスタンスメトリクスの表示

Amazon Lightsail でインスタンスを起動すると、インスタンスの管理ページの [Metrics (メトリクス)] タブでメトリクスグラフを表示できるようになります。メトリクスのモニタリングは、リソースの信頼性、可用性、パフォーマンスを維持する上で重要な要素です。リソースから定期的にメトリクスデータをモニタリングして収集し、マルチポイント障害が発生した場合に、より簡単にデバッグできるようにします。メトリクスの詳細については、「Amazon Lightsail のメトリクス」を参照してください。

リソースを監視するときは、環境内の通常のリソースパフォーマンスのベースラインを確立する必要があります。その後、リソースのパフォーマンスが指定のしきい値を超えたときに通知するように、Lightsail コンソールでアラームを設定できます。詳細については、「通知」および「アラーム」を参照してください。

目次

利用可能なインスタンスメトリクス

次のインスタンスメトリクスを使用できます。

- CPU 使用率 (**CPUUtilization**) — 割り当てられたコンピューティングユニットのうち、現在インスタンス上で使用されているものの割合。このメトリクスは、インスタンスでアプリケーションを実行するための処理能力を識別します。インスタンスがフルプロセッサコアに割り当てられていない場合に、オペレーティングシステムのツールが Lightsail よりも低い割合を示す場合があります。

Lightsail コンソールでインスタンスの CPU 使用率メトリクスグラフを表示すると、持続可能なゾーンとバースト可能なゾーンが表示されます。これらのゾーンの意味の詳細については、「CPU 使用率の持続可能なゾーンとバースト可能なゾーン」を参照してください。

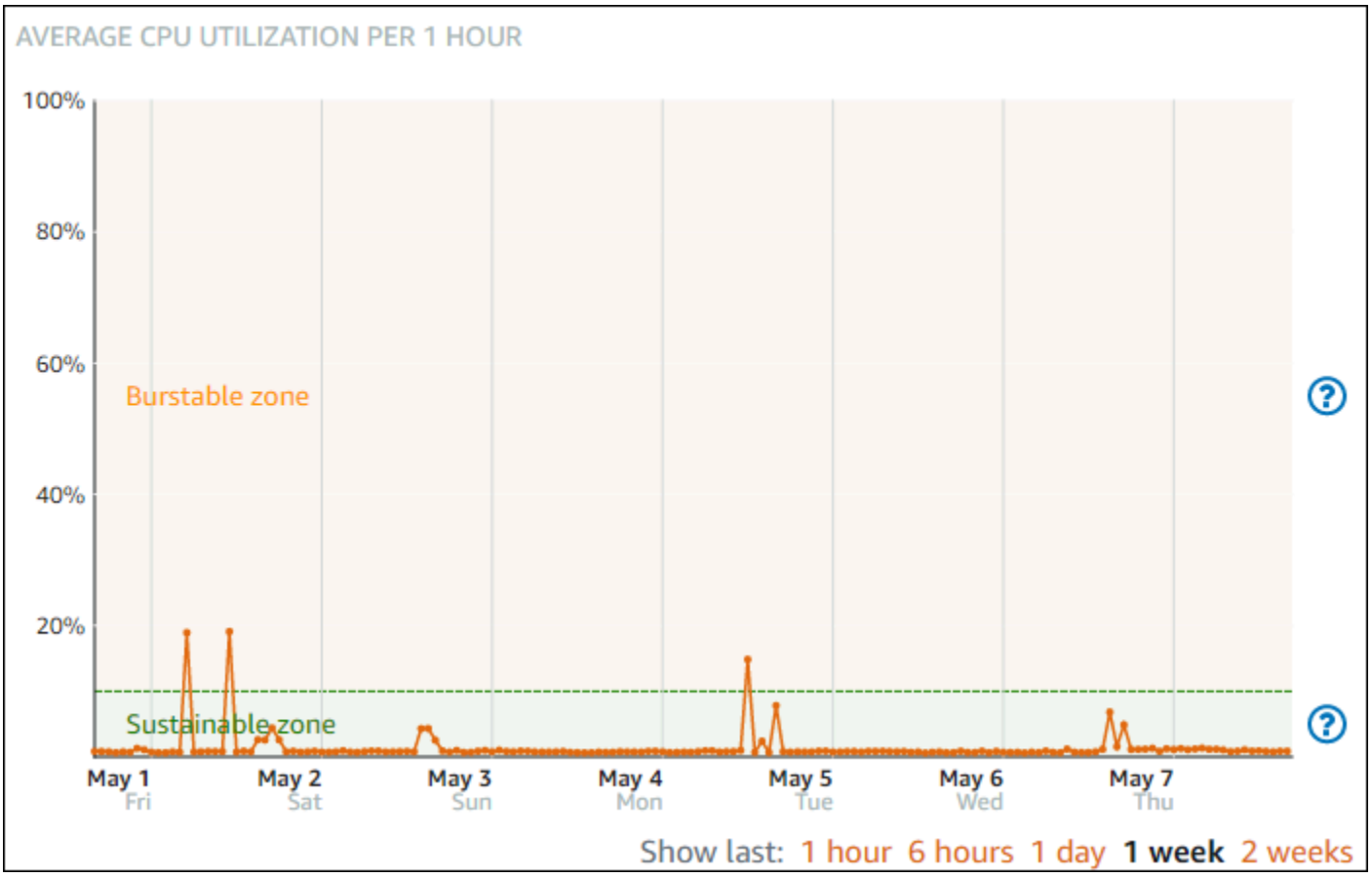
- バーストキャパシティ (**BurstCapacityTime**) および割合 (**BurstCapacityPercentage**) — バーストキャパシティ分数は、インスタンスが CPU 使用率 100% でバーストできる時間を表します。バーストキャパシティの割合は、インスタンスで利用できる CPU パフォーマンスの割合です。インスタンスはバースト容量を継続的に消費し、蓄積します。インスタンスが 100% の CPU 使用率で動作しているときにのみ、バーストキャパシティの分数がフルレートで消費されます。インスタンスのバーストキャパシティの詳細については、「インスタンスのバーストキャパシティの表示」を参照してください。
- 受信ネットワークトラフィック (**NetworkIn**) — すべてのネットワークインターフェイスでの、このインスタンスによって受信されたバイト数。このメトリクスは、1 つのインスタンスへの着信ネットワークトラフィックの量を表しています。報告された数は、期間中に受信されたバイト数です。このメトリクスは 5 分間隔でレポートされるため、レポートされた数を 300 で割ると、バイト/秒を算出できます。
- 送信ネットワークトラフィック (**NetworkOut**) — すべてのネットワークインターフェイスでの、このインスタンスから送信されたバイト数。このメトリクスは、1 つのインスタンスからの発信ネットワークトラフィックの量を表しています。報告された数は、期間中に送信されたバイト数です。このメトリクスは 5 分間隔でレポートされるため、レポートされた数を 300 で割ると、バイト/秒を算出できます。
- ステータスチェックの失敗 (**StatusCheckFailed**) — インスタンスが、インスタンスステータスチェックとシステムステータスチェックの両方に合格したか失敗したかをレポートします。このメトリクスは 0 (合格) または 1 (失敗) となります。このメトリクスは、1 分間の頻度で利用できます。
- インスタンスステータスチェックの失敗 (**StatusCheckFailed_Instance**) — インスタンスがインスタンスステータスチェックに合格したか、失敗したかをレポートします。このメトリクスは 0 (合格) または 1 (失敗) となります。このメトリクスは、1 分間の頻度で利用できます。
- ステータスチェックの失敗 (**StatusCheckFailed_System**) — インスタンスが、システムステータスチェックに合格したか失敗したかをレポートします。このメトリクスは 0 (合格) または 1 (失敗) となります。このメトリクスは、1 分間の頻度で利用できます。
- トークンメタデータなしのリクエスト (**MetadataNoToken**) — トークンなしでインスタンスのメタデータサービスに正常にアクセスした回数。このメトリクスにより、トークンを使用しない Instance Metadata Service バージョン 1 を使用してインスタンスメタデータにアクセスするプロセスがあるかどうかがわかります。すべてのリクエストがトークン支援のセッション (Instance

Metadata Service バージョン 2 など) を使用している場合、値は 0 になります。詳細については、「インスタンスメタデータとユーザーデータ」を参照してください。

CPU 使用率の持続可能なゾーンとバースト可能なゾーン

Lightsail は、バースト可能なインスタンスを使用します。これは CPU パフォーマンスのベースライン量を提供しますが、必要に応じてベースラインを上回る CPU パフォーマンスを一時的に提供することもできます。これをバーストといいます。バースト可能なインスタンスを使用すると、時々発生するパフォーマンスの急上昇 (スパイク) に対応するためにインスタンスを過剰にプロビジョンする必要がありません。つまり、使用しない容量に対して料金を支払う必要がありません。

インスタンスの CPU 使用率メトリクスグラフに、持続可能なゾーンとバースト可能なゾーンが表示されます。Lightsail 持続可能領域内だと、インスタンスはシステムのオペレーションに影響を与えることなく無制限で作動することができます。



コードのコンパイル、新しいソフトウェアのインストール、バッチジョブの実行、ピークの負荷リクエストの処理など、負荷が高い場合、インスタンスがバースト可能なゾーンで動作し始めることがあ

ります。バーストな可能ゾーンで動作している間、インスタンスは大量の CPU サイクルを消費します。したがって、この領域では限られた期間しか作動できません。

インスタンスがバースト可能なゾーンで動作できる期間は、バースト可能なゾーンにどの程度入っているかによって異なります。バースト可能なゾーンの下限近くで動作しているインスタンスは、バースト可能なゾーンの上限近くで動作しているインスタンスよりも長い時間バーストできます。ただし、一定期間バースト可能なゾーンにあるインスタンスは、持続可能なゾーンで再び動作するまで、最終的にすべての CPU 容量を使い果たすことになります。

インスタンスの CPU 使用率メトリクスを監視して、持続可能なゾーンとバーストな可能ゾーン間でパフォーマンスがどのように分散されているかを確認してください。システムが時折バースト可能なゾーンに移動するだけの場合は、実行中のインスタンスを引き続き使用しても問題ありません。ただし、インスタンスがバーストな可能ゾーンでかなりの時間を費やしている場合は、インスタンスのより大きなプランに切り替えることを検討してください (たとえば、$3.50 USD/月プランではなく $10 USD/月プランを使用します)。インスタンスの新しいスナップショットを作成し、スナップショットから新しいインスタンスを作成することで、より大きなプランに切り替えることができます。

Lightsail コンソールでインスタンスメトリクスを表示する

Lightsail コンソールでインスタンスメトリクスを表示するには、次の手順を実行します。

1. Lightsail コンソールにサインインします。
2. Lightsail ホームページで、[Instances (インスタンス)] タブを選択します。
3. メトリクスを表示するインスタンスの名前を選択します。
4. インスタンス管理ページで [Metrics (メトリクス)] タブを選択します。
5. [Metrics graph (メトリクスグラフ)] 見出しの下のドロップダウンメニューで、表示するメトリクスを選択します。

   グラフには、選択したメトリクスのデータポイントが視覚的に表示されます。

   > **Note**
   >
   > Lightsail コンソールでインスタンスの CPU 使用率メトリクスグラフを表示すると、持続可能なゾーンとバースト可能なゾーンが表示されます。これらのゾーンの詳細については、「CPU 使用率の持続可能なゾーンとバースト可能なゾーン」を参照してください。

6. メトリクスグラフでは、次のアクションを実行できます。

- グラフの表示を変更して、1 時間、6 時間、1 日、1 週間、2 週間のデータを表示します。
- データポイント上にカーソルを置くと、そのデータポイントに関する詳細情報が表示されます。
- 指定したしきい値をメトリクスが超えたときに通知される、選択したメトリクスのアラームを追加します。詳細については、「アラーム」および「インスタンスのメトリクスアラームを作成する」を参照してください。

次のステップ

インスタンスメトリクスに対して実行できる追加のタスクがいくつかあります。

- 指定したしきい値をメトリクスが超えたときに通知される、選択したメトリクスのアラームを追加します。詳細については、「メトリクスのアラーム」および「インスタンスのメトリクスアラームを作成する」を参照してください。
- アラームが作動すると、通知バナーが Lightsail コンソールに表示されます。メールおよび SMS テキストメッセージで通知を受けるには、リソースを監視する各 AWS リージョン で、メールアドレスと携帯電話番号を通知連絡先として追加する必要があります。詳細については、「通知連絡先を追加する」を参照してください。
- 通知の受信を停止するには、Lightsail から メールと携帯電話を削除します。詳細については、「メトリクスアラームを削除または無効化する」を参照してください。また、アラームを無効化または削除して、特定のアラームの通知の受信を停止することもできます。詳細については、「メトリクスアラームを削除または無効化する」を参照してください。

Lightsail のメトリクスアラーム

Amazon Lightsail では、インスタンス、データベース、ロードバランサー、コンテンツ配信ネットワーク (CDN) ディストリビューションでの 1 つのメトリクスを監視するアラームを作成できます。アラームは、指定したしきい値を基準にしたメトリクスの値に基づいて通知するように設定できます。通知は、Lightsail コンソールに表示されるバナー、メールアドレスに送信される メール、携帯電話番号に送信される SMS テキストメッセージです。このガイドでは、アラームの条件と設定について説明します。

目次

- アラームの例
- アラームによる欠落データの処理方法の設定
- データが欠落した場合のアラーム状態の評価方法
- グラフ化された例の欠落データ
- アラームの詳細

アラームの設定

Lightsail コンソールでアラームを追加するには、インスタンス、データベース、ロードバランサーあるいは CDN ディストリビューションの [メトリクス] タブを参照します。次に、モニタリングするメトリクスを選択し、[Add alarm (アラームの追加)] を選択します。メトリクスごとに 2 つのアラームを追加できます。メトリクスの詳細については、「リソースのメトリクス」を参照してください。

アラームを設定するには、まずしきい値を特定します。しきい値は、アラームが状態を変更する時点のメトリクス値です (OK 状態から ALARM 状態への変更、またはその逆の変更など)。詳細については、「 アラームの状態」を参照してください。次に、メトリクスとしきい値の比較に使用する比較演算子を選択します。使用できる演算子は、greater than or equal to、greater than、less than、less than or equal to です。

次に、アラームの状態を変更するまでに、しきい値を超える必要がある回数と、メトリクスを評価する期間を指定します。Lightsail はアラームのデータポイントを 5 分ごとに評価し、各データポイントは 5 分間の集約データを表します。たとえば、しきい値が 2 回を超えたときにトリガーするアラームを指定した場合、評価期間は過去 10 分以上 (最大 24 時間) である必要があります。しきい値を 10 回超えたときにトリガーするアラームを指定した場合、評価期間は過去 50 分以上 (最大 24 時間) である必要があります。

アラームの条件を設定したら、通知方法を設定できます。アラームが OK 状態から ALARM 状態に変化すると、通知バナーは常に Lightsail コンソールに表示されます。メールおよび SMS テキストメッセージによる通知を選択することもできますが、それらの通知連絡先を設定する必要があります。詳細については、「メトリクスの通知」を参照してください。メール、または SMS テキストメッセージによる通知を選択する場合、アラームの状態が ALARM 状態から OK 状態に変化したときに通知を受けるように選択することもできます。これは、すべてクリアな通知と見なされます。

アラームの [Advanced settings (詳細設定)] では、Lightsail が欠落しているメトリクスデータを処理する方法を選択できます。詳細については、「 アラームが欠落データを処理する方法の設定」を参照してください。

アラームの状態

アラームは、常に次の状態のいずれかになります。

- ALARM — メトリクスの値が、定義したしきい値の範囲外にあります。

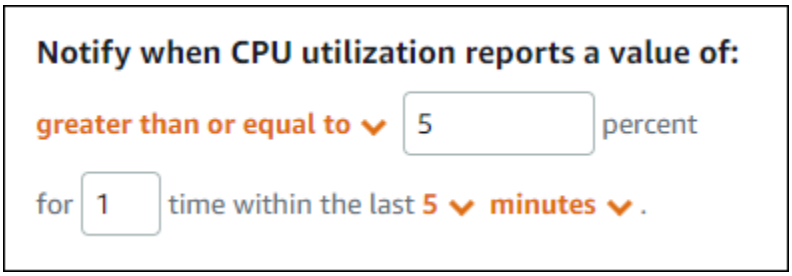
  たとえば、greater than 比較演算子を選択した場合、メトリクスが指定したしきい値を超えると、アラームは ALARM 状態になります。less than 比較演算子を選択した場合、メトリクスが指定したしきい値を下回ると、アラームは ALARM 状態になります。
- OK: — メトリクスの値が、定義されたしきい値の範囲内にあります。

  たとえば、greater than 比較演算子を選択した場合、メトリクスが指定したしきい値を下回ると、アラームは OK 状態になります。less than 比較演算子を選択した場合、メトリクスが指定したしきい値を超えると、アラームは OK 状態になります。
- INSUFFICIENT_DATA — アラームが開始されたか、メトリクスが利用可能でないか、またはメトリクスがアラームの状態を決定するためのデータが不足しています。

アラームは、状態変更に対してのみトリガーされます。アラームは単に、特定の状態にあるだけでは作動しません — 状態が変更されていることが条件です。アラームがトリガーされると、Lightsail コンソールにバナーが表示されます。E メールや SMS テキストメッセージで通知するようにアラームを設定することもできます。

アラームの例

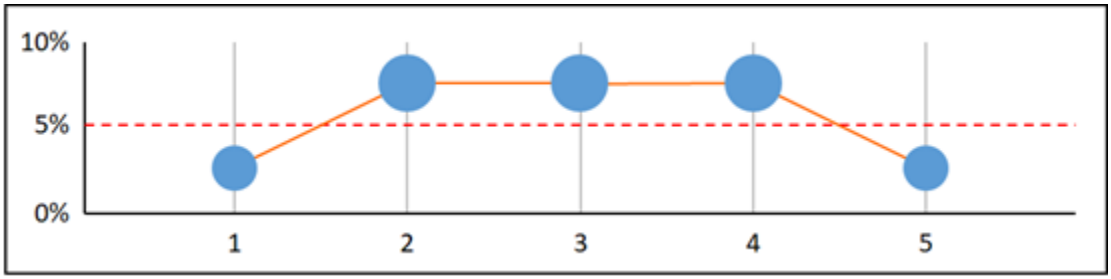
前述のアラーム条件を考慮して、インスタンスの CPU 使用率が 5 分間隔の 1 回で 5% 以上になったときに ALARM 状態になるアラームを設定できます。次の例は、Lightsail コンソールでのこのアラームの設定を示しています。



この例では、インスタンスの CPU 使用率メトリクスが 1 つのデータポイントで 5% 以上の使用率を報告した場合、アラームは OK 状態から ALARM 状態に変化します。使用率が 5% 以上と報告された後続の各データポイントは、アラームをある ALARM 状態で維持します。インスタンスの CPU 使用

率メトリクスが、1 つのデータポイントでの使用率を 4.9% 以下と報告すると、アラームは ALARM 状態から OK 状態に変わります。

次のグラフは、このアラームをさらに示しています。赤い点線は 5% の CPU 使用率のしきい値を表し、青い点はメトリクスデータポイントを表します。アラームは、最初のデータポイントの OK 状態です。2 番目のデータポイントは、データポイントがしきい値を超えているため、アラームを ALARM 状態に変更します。データポイントはしきい値よりも大きくなるため、3 番目と 4 番目のデータポイントは ALARM 状態を維持します。5 番目のデータポイントは、データポイントがしきい値を下回っているため、アラームを OK 状態に変更します。



アラームによる欠落データの処理方法の設定

場合によっては、アラームのあるメトリクスのデータポイントがレポートされないことがあります。たとえば、接続が失われたり、サーバーがダウンしたりした場合に発生します。

Lightsail では、アラームの設定時に欠落したデータポイントを処理する方法を指定できます。これは、監視しているデータの種類に応じて適切な場合、ALARM 状態に遷移するようにアラームを設定する場合に便利です。欠落データが問題を示すものではない場合の誤検出を避けることができます。

各アラームが常に 3 つの状態のいずれかであるように、データポイントはそれぞれ、次の 3 つのカテゴリのいずれかの状態に該当します。

- Not breaching — データポイントがしきい値の範囲内です。

  たとえば、greater than 比較演算子を選択した場合、指定したしきい値を下回ったときにデータポイントが `Not breaching` になります。less than 比較演算子を選択した場合、指定したしきい値を超えたときにデータポイントは `Not breaching` になります。

- Breaching — データポイントがしきい値の範囲外です。

  たとえば、greater than 比較演算子を選択した場合、指定したしきい値を超えたときにデータポイントが `Breaching` になります。less than 比較演算子を選択すると、指定したしきい値を下回ったときにデータポイントは `Breaching` になります。

- Missing — 欠落しているデータポイントに対する動作は、`treat missing data` パラメータによって指定されます。

アラームごとに、Lightsail が欠落データポイントを次のいずれかとして処理するように指定できます。

- Not breaching — 欠落データポイントは「正常」とされ、しきい値内として扱われます。
- Breaching — 欠落データポイントは「不良」とされ、しきい値超過として扱われます。
- Ignore — 現在のアラーム状態が維持されます。
- Missing — 状態を変更するかどうかを評価する際に、アラームは欠落データポイントを考慮に入れません。これは、アラームのデフォルトの動作です。

最適な選択は、メトリクスの種類によって異なります。インスタンスの CPU 使用率などのメトリクスでは、欠落しているデータポイントをしきい値を超過として扱うことができます。これは、欠落しているデータポイントが、何かが間違っていることを示している可能性があるためです。ただし、ロードバランサーの HTTP 500 サーバーエラー数など、エラーが発生したときにのみデータポイントを生成するメトリクスでは、欠落したデータをしきい値内として扱うことができます。

アラームに最適なオプションを選択すると、不必要で誤解を招くアラームの状態の変更を防ぐことができます。また、システムの正常性をより正確に示します。

データが欠落した場合のアラーム状態の評価方法

欠落データの処理方法に設定した値にかかわらず、アラームが状態を変更するかどうかを評価する際に、Lightsail は評価期間の指定よりも多くのデータポイントを取得しようとします。取得しようとするデータポイントの正確な数は、アラーム期間の長さによって異なります。取得を試みるデータポイントのタイムフレームは評価範囲です。

Lightsail がこれらのデータポイントを取得すると、次の処理が実行されます。

- 評価範囲内のデータポイントが欠落していない場合、Lightsail は収集された最新のデータポイントに基づいてアラームを評価します。
- 評価範囲のデータポイントの一部が欠落しているが、取得された既存のデータポイントの数がアラームの評価期間以上である場合、正常に取得された最新の既存のデータポイントに基づいてアラームの状態が Lightsail で評価されます。この場合、欠落データを処理する方法に設定した値は不要であり、無視されます。

- 評価範囲のデータポイントの一部が欠落しており、取得された既存のデータポイントの数がアラームの評価期間の数を下回る場合、Lightsail によって、欠落データ部分に欠落データの処理方法に指定された結果が入力され、アラームが評価されます。ただし、評価範囲内の実際のデータポイントが、報告されたタイミングにかかわらず、評価に含まれます。 Lightsail は、欠落データポイントの使用を最小限に抑えます。

これらのすべての状況で、評価されるデータポイントの数は、評価期間の値と同じです。超過している数がアラームを発生させるデータポイント数の値よりも少ない場合、アラームの状態は OK に設定されます。それ以外の場合、状態は ALARM に設定されます。

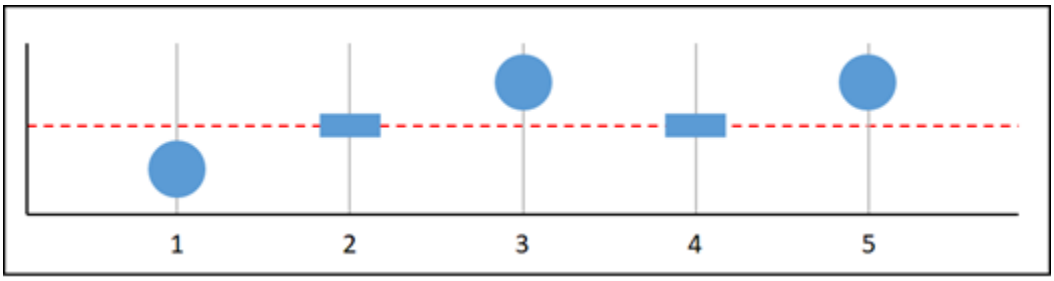
> **Note**
>
> この動作の特殊なケースは、メトリクスのフローが停止した後の一定期間、Lightsail アラームが最後のデータポイントのセットを繰り返し再評価する可能性があることです。この再評価により、メトリクスのストリームが停止する直前にアラームの状態が変更されていた場合に、アクションが再実行される可能性があります。この動作を軽減するには、より短い期間を使用します。

グラフ化された例の欠落データ

このセクションの次のグラフは、アラーム評価動作の例を示しています。グラフ A、B、C、D、E では、確実にアラームに違反しているデータポイントの数と評価期間は両方とも 3 になります。赤い点線はしきい値、青い点は有効なデータポイントを表し、ダッシュは欠落データを表します。しきい値ラインより上のデータポイントはしきい値を超過しており、しきい値を下回るデータポイントはしきい値内です。最新の 3 つのデータポイントの一部が欠落している場合、Lightsail は追加の有効なデータポイントを取得しようとします。

> **Note**
>
> アラームの作成直後にデータポイントが欠落し、アラームの作成前にメトリクスが Lightsail に報告されている場合、Lightsail はアラームを評価する際にアラーム作成前の直近のデータポイントを取得します。

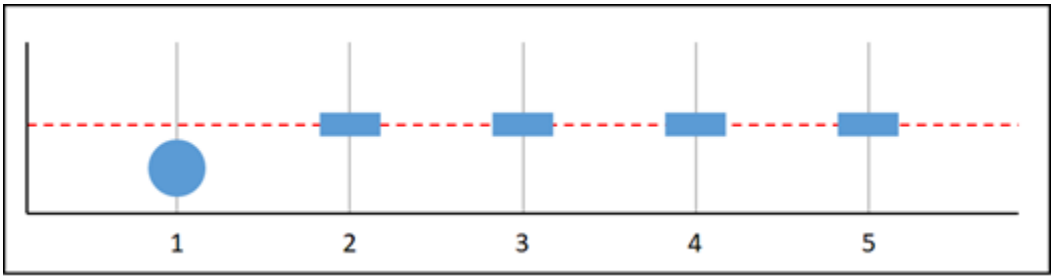
グラフ A



前述のグラフ化メトリクスでは、データポイント 1 がしきい値内、データポイント 2 が欠落し、データポイント 3 がしきい値を超過し、データポイント 4 が欠落し、データポイント 5 がしきい値を超過しています。評価範囲内に有効なデータポイントが 3 つあるので、このメトリクスの欠落しているデータポイントはゼロになります。欠落しているデータポイントを次のように扱うようにアラームを設定した場合:

- Not breaching — アラームは OK 状態になります。
- Breaching — アラームは OK 状態になります。
- Ignore — アラームは OK 状態になります。
- Missing — アラームは OK 状態になります。

グラフ B

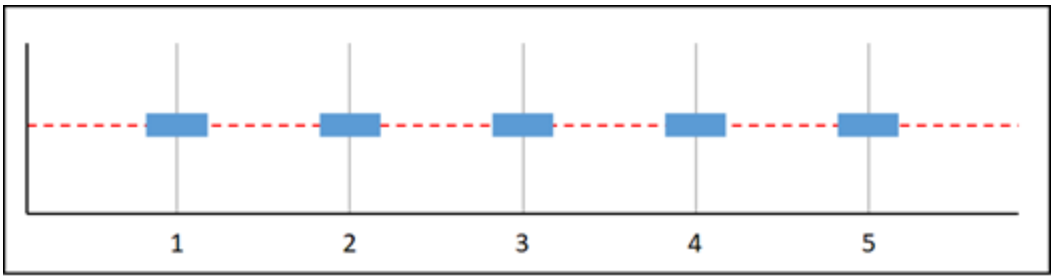


前述のグラフ化メトリクスでは、データポイント 1 がしきい値内にあり、データポイント 2 ~ 5 が欠落しています。評価範囲内にデータポイントが 1 つしかないので、このメトリクスには 2 つの欠落データポイントがあります。欠落しているデータポイントを次のように扱うようにアラームを設定した場合:

- Not breaching — アラームは OK 状態になります。
- Breaching — アラームは OK 状態になります。
- Ignore — アラームは OK 状態になります。
- Missing — アラームは OK 状態になります。

このシナリオでは、失われたデータがしきい値を超過として扱われる場合でも、アラームは OK 状態のままになります。これは、1 つの既存のデータポイントがしきい値内であるため、しきい値を超過として扱われる 2 つの欠落データポイントとともに評価されるためです。次回このアラームが評価されるときに、データがまだ欠落している場合は、ALARM に送られます。これは、しきい値内のデータポイントが取得された 5 つの最新のデータポイントに含まれることがなくなったためです。

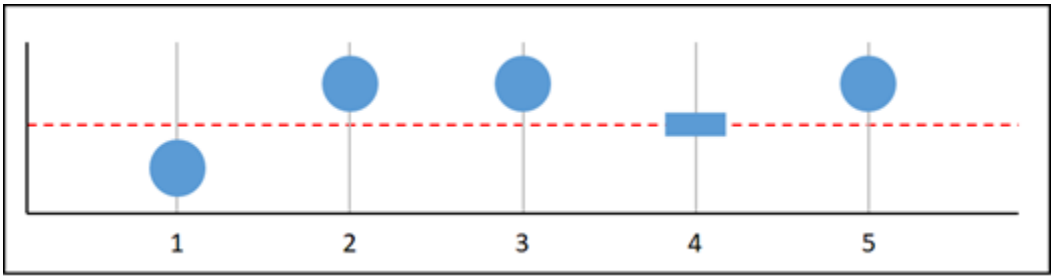
グラフ C



前述のグラフ化メトリクスでは、すべてのデータポイントが欠落しています。評価範囲内のすべてのデータポイントが欠落している場合、このメトリクスには 3 つの欠落データポイントがあります。欠落しているデータポイントを次のように扱うようにアラームを設定した場合:

- Not breaching — アラームは OK 状態になります。
- Breaching — アラームは ALARM 状態になります。
- Ignore — アラームは現在の状態を維持します。
- Missing — アラームは INSUFFICIENT_DATA 状態になります。

グラフ D

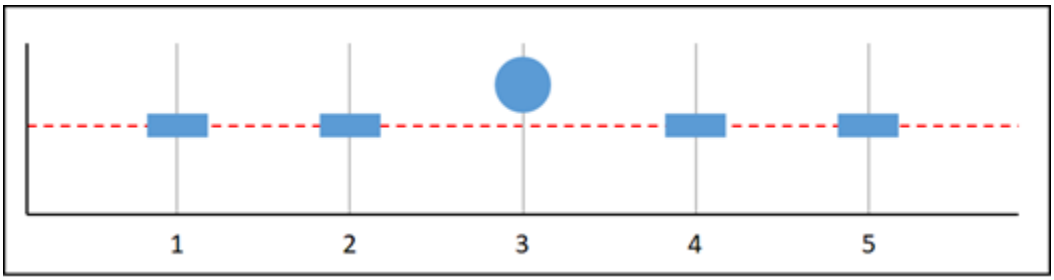


前述のグラフ化メトリクスでは、データポイント 1 がしきい値内、データポイント 2 がしきい値を超過し、データポイント 3 がしきい値を超過し、データポイント 4 が欠落し、データポイント 5 がしきい値を超過しています。評価範囲内に有効なデータポイントが 4 つあるので、このメトリクスの欠落しているデータポイントはゼロになります。欠落しているデータポイントを次のように扱うようにアラームを設定した場合:

- Not breaching — アラームは ALARM 状態になります。
- Breaching — アラームは ALARM 状態になります。
- Ignore — アラームは ALARM 状態になります。
- Missing — アラームは ALARM 状態になります。

このシナリオでは、アラームはすべてのケースで ALARM 状態になります。これは、欠落したデータの処理方法の設定が不要で、無視される十分な実際のデータポイントがあるためです。

グラフ E

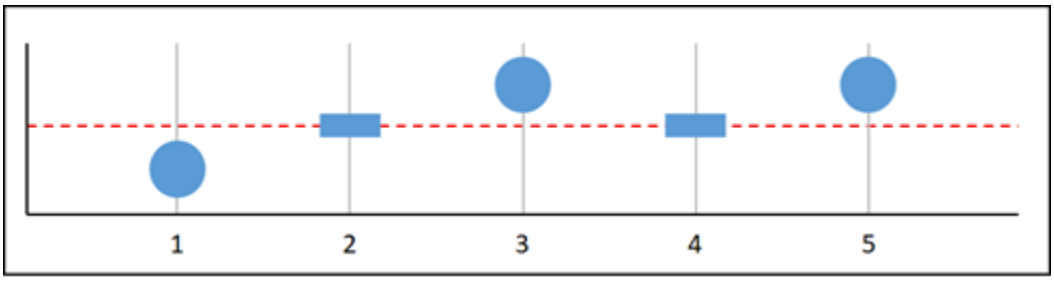


前述のグラフ化メトリクスでは、データポイント 1 と 2 が欠落し、データポイント 3 がしきい値を超過し、データポイント 4 と 5 が欠落しています。評価範囲内にデータポイントが 1 つしかないので、このメトリクスには 2 つの欠落データポイントがあります。欠落しているデータポイントを次のように扱うようにアラームを設定した場合:

- Not breaching — アラームは OK 状態になります。
- Breaching — アラームは ALARM 状態になります。
- Ignore — アラームは現在の状態を維持します。
- Missing — アラームは ALARM 状態になります。

グラフF、G、H、I、J では、 アラームへのデータポイントは 2 で、評価期間は 3 です。3 中 2 、N 中 M のアラームです。5 はアラームの評価範囲です。

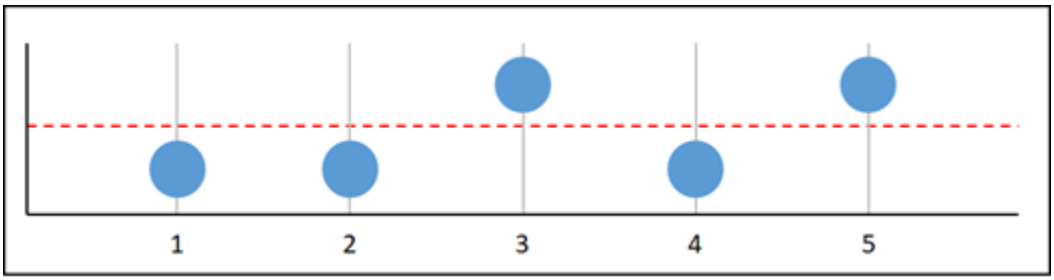
グラフ F



前述のグラフ化メトリクスでは、データポイント 1 がしきい値内で、データポイント 2 が欠落し、データポイント 3 がしきい値を超過し、データポイント 4 が欠落し、データポイント 5 がしきい値を超過しています。評価範囲に 3 つのデータポイントがあるので、このメトリクスの欠落したデータポイントはゼロになります。欠落しているデータポイントを次のように扱うようにアラームを設定した場合:

- Not breaching — アラームは ALARM 状態になります。
- Breaching — アラームは ALARM 状態になります。
- Ignore — アラームは ALARM 状態になります。
- Missing — アラームは ALARM 状態になります。

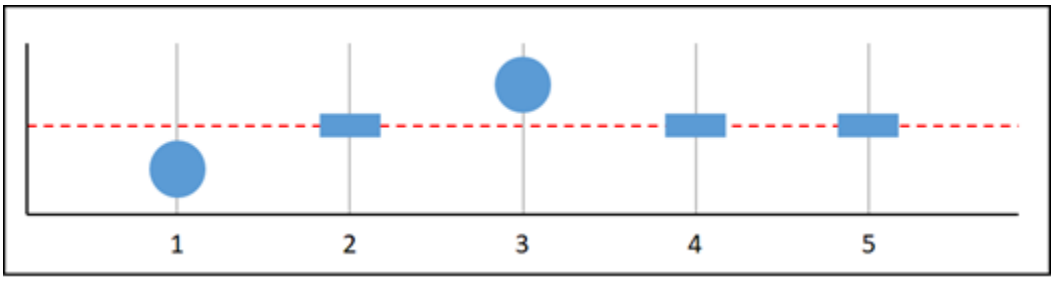
グラフ G



前述のグラフ化メトリクスでは、データポイント 1 と 2 がしきい値内で、データポイント 3 がしきい値を超過し、データポイント 4 がしきい値内で、データポイント 5 がしきい値を超過しています。評価範囲に 5 つのデータポイントがあるので、このメトリクスの欠落データポイントはゼロになります。欠落しているデータポイントを次のように扱うようにアラームを設定した場合:

- Not breaching — アラームは ALARM 状態になります。
- Breaching — アラームは ALARM 状態になります。
- Ignore — アラームは ALARM 状態になります。
- Missing — アラームは ALARM 状態になります。

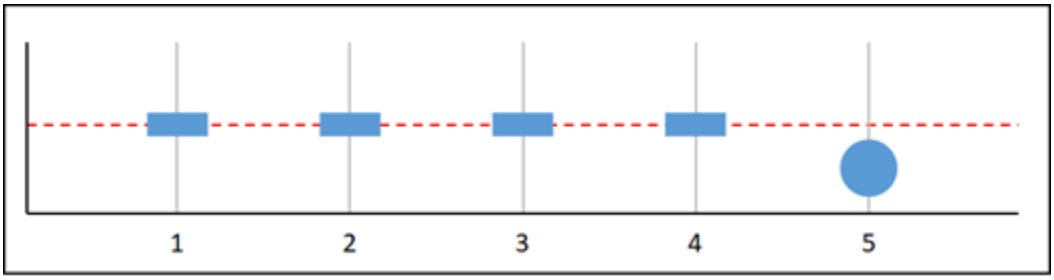
グラフ H



前述のグラフ化メトリクスでは、データポイント 1 がしきい値内で、データポイント 2 が欠落し、データポイント 3 がしきい値を超過し、データポイント 4 と 5 が欠落しています。評価範囲に 2 つのデータポイントがある場合ので、このメトリクスには欠落したデータポイントが 1 つあります。欠落しているデータポイントを次のように扱うようにアラームを設定した場合:

- Not breaching — アラームは OK 状態になります。
- Breaching — アラームは ALARM 状態になります。
- Ignore — アラームは OK 状態になります。
- Missing — アラームは OK 状態になります。

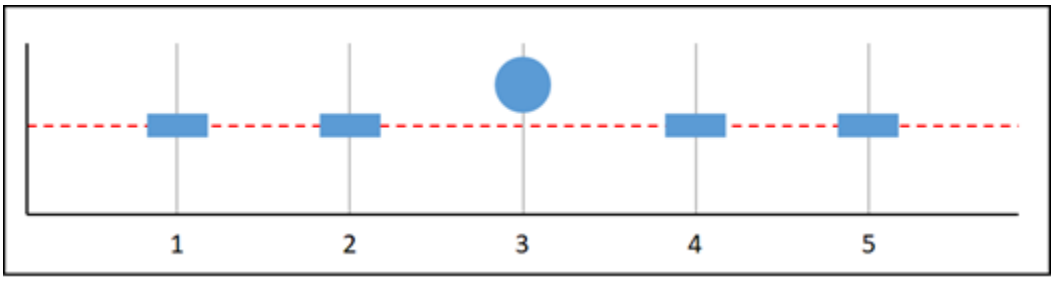
グラフ I



前述のグラフ化メトリクスでは、データポイント 1~4 が欠落し、データポイント 5 がしきい値内です。評価範囲に 1 つのデータポイントがあるので、このメトリクスには 2 つの欠落データポイントがあります。欠落しているデータポイントを次のように扱うようにアラームを設定した場合:

- Not breaching — アラームは OK 状態になります。
- Breaching — アラームは ALARM 状態になります。
- Ignore — アラームは OK 状態になります。
- Missing — アラームは OK 状態になります。

グラフ J



前述のグラフ化メトリクスでは、データポイント 1 と 2 が欠落し、データポイント 3 がしきい値を超過し、データポイント 4 と 5 が欠落しています。評価範囲に 1 つのデータポイントがあるので、このメトリクスには 2 つの欠落データポイントがあります。欠落しているデータポイントを次のように扱うようにアラームを設定した場合:

- Not breaching — アラームは OK 状態になります。
- Breaching — アラームは ALARM 状態になります。
- Ignore — アラームは現在の状態を維持します。
- Missing — アラームは ALARM 状態になります。

アラームの詳細

ここでは、Lightsail でアラームを管理するのに役立つ記事をいくつか紹介します。

- インスタンスのメトリクスアラームを作成する
- データベースのメトリクスにアラームを作成する
- ロードバランサーのメトリックスアラームを作成する
- ディストリビューションのメトリクスにアラームを作成する
- メトリクスアラームの削除または無効化

Lightsail インスタンスのメトリクスアラームを作成する

単一のインスタンスメトリクスを監視する Amazon Lightsail アラームを作成できます。アラームは、指定したしきい値を基準にしたメトリクスの値に基づいて通知するように設定できます。通知は、Lightsail コンソールに表示されるバナー、メールアドレスに送信される メール、携帯電話番号に送信される SMS テキストメッセージです。アラームの詳細については、「アラーム」を参照してください。

目次

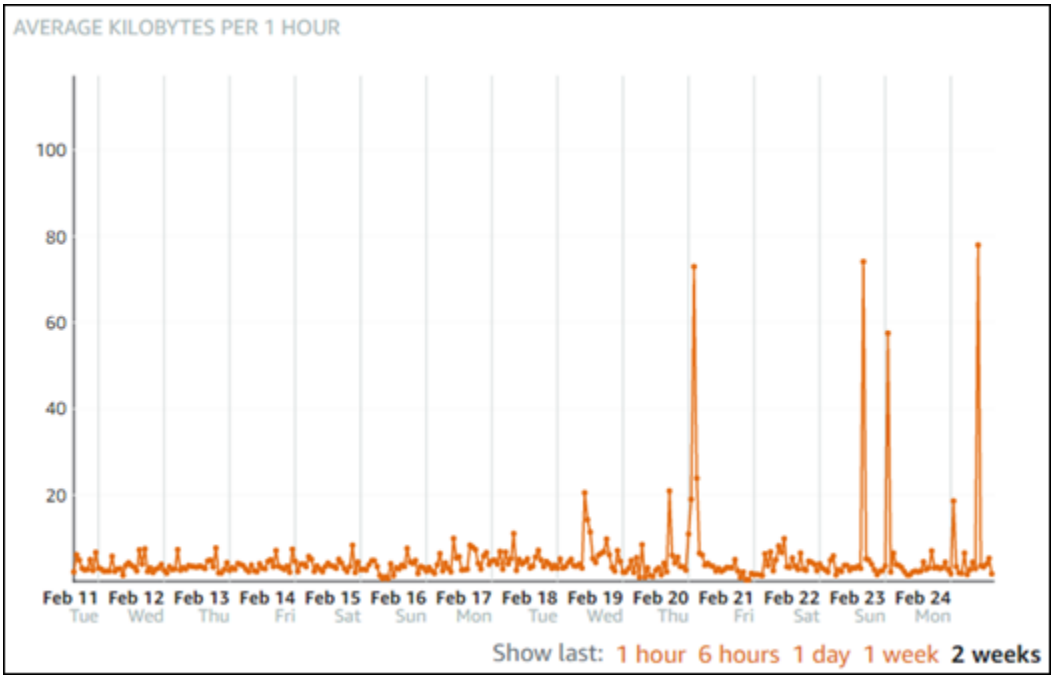
インスタンスアラームの制限

アラームには、以下の制限が適用されます。

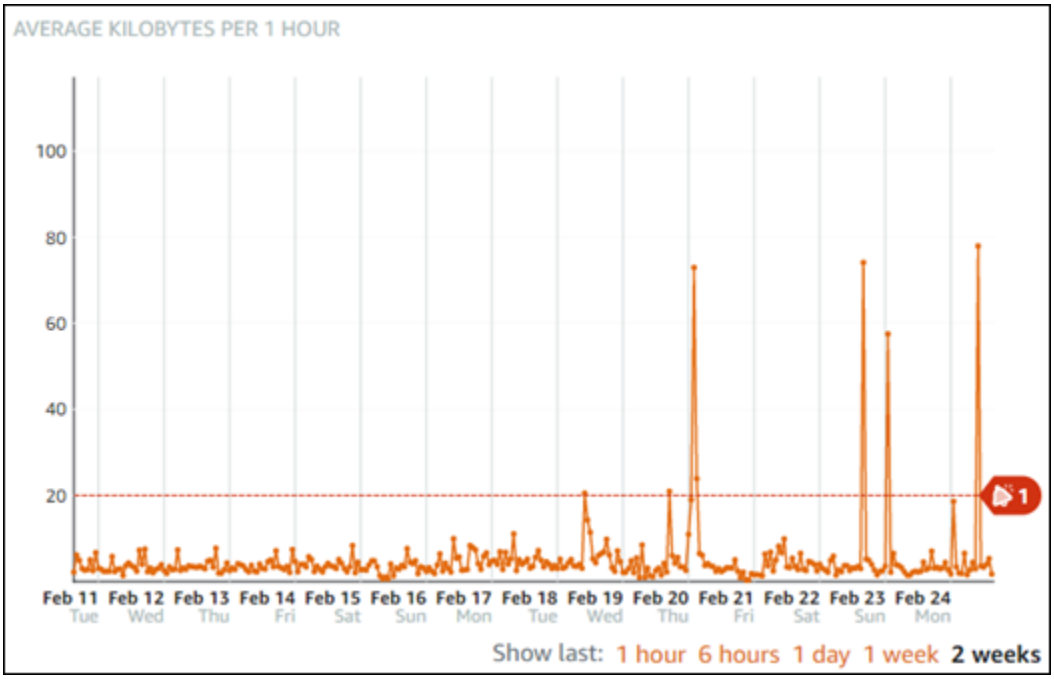
- メトリクスごとに 2 つのアラームを設定できます。
- アラームは 5 分間隔で評価され、アラームの各データポイントは、集計されたメトリクスデータの 5 分間隔を表します。
- アラームの状態が OK に変わったときに通知するようにアラームを設定できるのは、メールや SMS テキストメッセージで通知するようにアラームを設定した場合だけです。
- OK アラーム通知をテストできるのは、メールや SMS テキストメッセージで通知するようにアラームを設定した場合だけです。
- アラームの状態が `INSUFFICIENT_DATA` に変わったときに通知するようにアラームを設定できるのは、メールや SMS テキストメッセージで通知するようにアラームを設定した場合と、欠落しているデータポイントに対して [欠落しているデータを評価しない] オプションを選択した場合だけです。
- 通知をテストできるのは、アラームが OK 状態の場合だけです。

インスタンスアラームの設定に関するベストプラクティス

インスタンスのメトリクスアラームを設定する前に、メトリクスの履歴データを表示する必要があります。過去 2 週間のメトリクスの低レベル、中間レベル、高レベルを識別します。次の発信ネットワークトラフィック (`NetworkOut`) メトリクスグラフの例では、低レベルは 1 時間あたり 0 ~ 10 KB、中間レベルは 1 時間あたり 10 ~ 20 KB、高レベルは 1 時間あたり 20 ~ 80 KB です。



アラームしきい値を低レベル範囲 (1 時間あたり 5 KB など) 以上に設定すると、アラーム通知が頻繁に送信され、不要なアラーム通知が発生する可能性があります。アラームしきい値を高レベルの範囲 (たとえば、1 時間あたり 20 KB) 以上に設定した場合、アラーム通知の頻度は低くなりますが、調査がさらに意義のあるものになる場合があります。アラームを設定して有効にすると、次の例に示すように、しきい値を表すアラームラインがグラフに表示されます。1 というラベルの付いたアラームラインはアラーム 1 のしきい値を表し、2 というラベルのアラームラインはアラーム 2 のしきい値を表します。



デフォルトのアラーム設定

Lightsail コンソールで新しいアラームを追加すると、デフォルトのアラーム設定があらかじめ入力されています。これは、選択したメトリクスの推奨アラーム設定です。ただし、デフォルトのアラーム設定がリソースに適していることを確認する必要があります。たとえば、インスタンスの発信ネットワークトラフィック (NetworkOut) メトリクスのデフォルトのアラームしきい値は、過去 10 分以内に 2 回 0 バイト[以下]です。ただし、トラフィックの多いイベントの通知を受ける場合は、アラームのしきい値を過去 10 分以内に 2 回 50 KB [以上]に変更するか、これらの設定に 2 番目のアラームを追加して、トラフィックがない場合および、トラフィックが多い場合に通知を受け取るようにします。　指定するしきい値は、このガイドの「インスタンスアラームの設定に関するベストプラクティス」セクションで説明されているように、メトリクスの高レベルと低レベルと一致するように調整する必要があります。

Lightsail コンソールを使用してインスタンスのメトリクスアラームを作成する

Lightsail コンソールを使用してインスタンスメトリクスアラームを作成するには、以下のステップを実行します。

1. Lightsail コンソールにサインインします。
2. Lightsail ホームページで、[Instances (インスタンス)] タブを選択します。
3. アラームを作成するインスタンスの名前を選択します。
4. インスタンス管理ページで [Metrics (メトリクス)] タブを選択します。
5. [Metrics Graphs (メトリクスグラフ)] 見出しの下のドロップダウンメニューで、アラームを作成するメトリクスを選択します。詳細については、「リソースのメトリクス」を参照してください。
6. アラームセクションのページでアラームの追加を選択します。
7. ドロップダウンメニューから比較演算子の値を選択します。例の値は、greater than or equal to、greater than、less than、less than or equal to です。
8. アラームのしきい値を入力します。
9. アラームへのデータポイントを入力します。
10. 評価期間を選択します。期間は、5 分から 24 時間まで 5 分単位で指定できます。
11. 次のいずれかの通知方法を選択します。

    - メール — アラームの状態が ALARM に変わると、メールで通知されます。
    - SMS テキストメッセージ — アラームの状態が ALARM に変わると、SMS テキストメッセージによって通知されます。SMS メッセージングは、Lightsail リソースを作成できるすべての

AWS リージョンでサポートされているわけではなく、また、一部の国/地域には SMS テキストメッセージを送信できません。詳細については、「SMS テキストメッセージングのサポート」を参照してください。

> **Note**
>
> メールまたは SMS による通知を選択したが、リソースの AWSリージョンで通知の連絡先をまだ設定していない場合は、メールアドレスまたは携帯電話番号を追加する必要があります。詳細については、「メトリクスの通知」を参照してください。

12. (オプション) [Send me a notification when the alarm state change to OK (アラームの状態が OK に変わったときに通知を送信する)] を選択して、アラームの状態が OK に変わったときに通知を受け取ります。このオプションは、メールまたは SMS テキストメッセージで通知されるように選択した場合にのみ使用できます。
13. (オプション) [Advanced settings (詳細設定)] を選択し、次のいずれかのオプションを選択します。

    - アラームによる欠落データの処理方法を選択します。次のオプションを使用できます。
      - しきい値の範囲内にないと仮定する(しきい値を超過) — 欠落しているデータポイントは「不良」として扱われ、しきい値を超えています。
      - しきい値内にあると仮定する (しきい値を超過していない) — 欠落しているデータポイントは、「良い」と見なされ、しきい値の範囲内で処理されます。
      - 最後の正常なデータポイントの値を使用する (無視して現在のアラーム状態を維持する) — 現在のアラーム状態が維持されます。
      - 評価しない (欠落しているデータを欠落しているデータとして扱う) — 状態を変更するかどうかを評価する際に、欠落データポイントを考慮に入れません。
    - [Send a notification if there is insufficient data (データが不足している場合に通知を送信)] を選択して、アラームの状態が INSUFFICIENT_DATA に変わったときに通知を受け取ります。このオプションは、メールまたは SMS テキストメッセージで通知されるように選択した場合にのみ使用できます。
14. [Create (作成)] を選択してアラームを追加します。

    後でアラームを編集するには、編集するアラームの横にある省略記号アイコン (⋮) を選択し、[アラームの編集] を選択します。

Lightsail コンソールを使用してインスタンスのメトリクスアラームをテストする

Lightsail コンソールを使用してアラームをテストするには、次の手順を実行します。アラームをテストして、アラームがトリガーされたときに メールや SMS テキストメッセージを受信するなど、設定された通知オプションが機能していることを確認します。

1. Lightsail コンソールにサインインします。
2. Lightsail ホームページで、[Instances (インスタンス)] タブを選択します。
3. アラームをテストするインスタンスの名前を選択します。
4. インスタンス管理ページで [Metrics (メトリクス)] タブを選択します。
5. [Metrics Graphs (メトリクスグラフ)] 見出しの下のドロップダウンメニューで、アラームをテストするメトリクスを選択します。
6. ページの [アラーム] セクションまでスクロールし、テストするアラームの横にある省略記号アイコン (⋮) を選択します。
7. 以下のオプションのいずれかを選択します。

   - [アラーム通知のテスト] - このオプションを選択すると、アラームの状態が ALARM に変わったときの通知をテストできます。
   - [OK 通知のテスト] — このオプションを選択すると、アラームの状態が OK に変わったときの通知をテストできます。

   > **Note**
   >
   > これらのオプションのいずれかが使用できない場合は、アラームの通知オプションが設定されていないか、アラームが現在 ALARM 状態になっている可能性があります。詳細については、「インスタンスのアラーム制限」を参照してください。

   選択したテストオプションに応じて、アラームが一時的に ALARM または OK 状態に変化し、アラームの通知方法として設定した内容に応じて、メールまたは SMS テキストメッセージが送信されます。通知バナーは、ALARM 通知のテストを選択した場合にのみ Lightsail コンソールに表示されます。OK 通知のテストを選択した場合、通知バナーは表示されません。アラームは、通常、数秒後に実際の状態に戻ります。

次のステップ

インスタンスアラームに対して実行できる追加のタスクがいくつかあります。

- 通知の受信を停止するには、Lightsail から メールと携帯電話を削除します。詳細については、「通知連絡先の削除」を参照してください。また、アラームを無効化または削除して、特定のアラームの通知の受信を停止することもできます。詳細については、「メトリクスアラームを削除または無効化する」を参照してください。

Lightsail メトリクスアラームの削除または無効化

アラームを削除して、Amazon Lightsail アラームによってモニタリングされているメトリクスがしきい値を超えたときの通知を停止できます。アラームを無効にして、通知の受信を停止することもできます。詳細については、「 アラーム」を参照してください。

目次

Lightsail コンソールを使用してメトリクスアラームを削除する

Lightsail コンソールを使用してメトリクスアラームを削除するには、次の手順を実行します。

1. Lightsail コンソールにサインインします。
2. Lightsail ホームページで、[インスタンス]、[データベース]、または [ネットワーク] タブを選択します。
3. アラームを削除するリソース (インスタンス、データベース、ロードバランサー) の名前を選択します。
4. リソースの管理ページで [メトリクス] タブを選択します。
5. [Metrics Graphs (メトリクスグラフ)] 見出しの下のドロップダウンで、アラームを削除するメトリクスを選択します。
6. ページの [アラーム] セクションまで下にスクロールし、削除するアラームの横にある省略記号アイコン (⋮) を選択します。
7. [Delete] (削除) をクリックします。
8. プロンプトで、[Delete (削除)] を選択して、アラームを削除することを確定します。

Lightsail コンソールを使用するメトリクスアラームの無効化および有効化

Lightsail コンソールを使用してメトリクスアラームを無効にするには、次の手順を実行します。

1. Lightsail コンソールにサインインします。
2. Lightsail ホームページで、[インスタンス]、[データベース]、または [ネットワーク] タブを選択します。
3. アラームを無効にするリソース (インスタンス、データベース、ロードバランサー) の名前を選択します。
4. リソースの管理ページで [メトリクス] タブを選択します。
5. [Metrics Graphs (メトリクスグラフ)] 見出しの下のドロップダウンで、アラームを無効にするメトリクスを選択します。
6. ページの [Alarms (アラーム)] セクションまで下にスクロールし、無効にするアラームを探し、トグルを選択して無効にします。同様に、無効になっている場合は、トグルを選択して有効にします。

Lightsail バケットのメトリクスの表示

Amazon Lightsail オブジェクトストレージサービスでバケットを作成した後、バケットの管理ページの [メトリクス] タブでそのメトリクスグラフを表示できます。メトリクスのモニタリングは、バケットの可用性、パフォーマンスを維持する上で重要なパートです。バケットから定期的にメトリクスデータをモニタリングして収集し、必要に応じてバケットのストレージスペースとネットワーク転送クォータをアップサイズまたはダウンサイズできるようにします。メトリクスの詳細については、「リソースのメトリクス」を参照してください。

リソースを監視するときは、環境内の通常のリソースパフォーマンスのベースラインを確立する必要があります。その後、リソースのパフォーマンスが指定のしきい値を超えたときに通知するように、Lightsail コンソールでアラームを設定できます。詳細については、「通知」および「アラーム」を参照してください。

バケットメトリクス

次のバケットメトリクスが利用可能です。

- [バケットサイズ] — バケットに保存されたデータの量。この値を計算するには、バケット内のすべてのオブジェクト (最新のオブジェクトと最新でないオブジェクトの両方) のサイズを合計しま

す。これには、バケットに対するすべての不完全なマルチパートアップロードのすべてのパートのサイズも含めます。

- オブジェクトの数 — バケットに保存されたオブジェクトの総数。この値を計算するには、バケット内のすべてのオブジェクト (最新のオブジェクトと最新でないオブジェクトの両方) と、バケットに対するすべての不完全なマルチパートアップロードの合計パート数をカウントします。

> **Note**
>
> バケットが空の場合、バケット メトリクス データはレポートされません。

Lightsail コンソールでのバケットメトリクスの表示

Lightsail コンソールでバケットメトリクスを表示するには、以下の手順を実行します。

1. Lightsail コンソールにサインインします。
2. Lightsail のホームページで [Storage] (ストレージ) タブを選択します。
3. メトリクスを表示するバケットの名前を選択します。
4. バケット管理ページで [Metrics] (メトリクス) タブを選択します。
5. [Metrics graphs] (メトリクスグラフ) の見出しの下のドロップダウンメニューで、表示するメトリクスを選択します。

   グラフには、選択したメトリクスのデータポイントが視覚的に表示されます。

   ScreenshotTBD

   メトリクスグラフでは、次のアクションを実行できます。

   - グラフの表示を変更して、1 時間、6 時間、1 日、1 週間、2 週間のデータを表示します。
   - データポイント上にカーソルを置くと、そのデータポイントに関する詳細情報が表示されます。
   - 指定したしきい値をメトリクスが超えたときに通知される、選択したメトリクスのアラームを追加します。詳細については、「アラーム」および「バケットメトリクスアラームの作成」を参照してください。

バケットとオブジェクトを管理する

これらは、Lightsail オブジェクトストレージバケットを管理する一般的な手順です。

1. Amazon Lightsail オブジェクトストレージサービスでのオブジェクトとバケットについて説明します。詳細については、「Amazon Lightsail のオブジェクトストレージ」を参照してください。
2. Amazon Lightsail でバケットに付けることができる名前について説明します。詳細については、「Amazon Lightsail でのバケットの命名規則」をご参照ください。
3. バケットを作成して、Lightsail オブジェクトストレージサービスの使用を開始します。詳細については、「Amazon Lightsail におけるバケットの作成 」を参照してください。
4. バケットのセキュリティのベストプラクティスと、バケットに設定できるアクセス許可について説明します。バケット内のすべてのオブジェクトをパブリックまたはプライベートにすることも、オブジェクトを個別に選択してパブリックにすることもできます。また、アクセスキーを作成し、インスタンスをバケットに追加し、他の AWS アカウントにアクセス権を付与することで、バケットへのアクセスを許可することもできます。詳細については、「Amazon Lightsail オブジェクトストレージのセキュリティベストプラクティス」と「Amazon Lightsail でのバケットのアクセス許可を理解する」を参照してください。

   バケットのアクセス許可について理解したら、以下のガイドを参照してバケットへのアクセスを許可してください。

   - Amazon Lightsail でのバケットへのパブリックアクセスをブロックする
   - Amazon Lightsail でのバケットのアクセス許可の設定
   - Amazon Lightsail でのバケット内の個々のオブジェクトに対するアクセス許可の設定
   - Amazon Lightsail でのバケットのアクセスキーの作成
   - Amazon Lightsail でのバケットのリソースアクセスの設定
   - Amazon Lightsail でのバケットのクロスアカウントアクセスの設定
5. バケットのアクセスログの記録を有効にする方法と、アクセスログを使用してバケットのセキュリティを監査する方法について説明します。詳細については、以下のガイドを参照してください。

   - Amazon Lightsail オブジェクトストレージサービスのバケットへのアクセスのログ記録
   - Amazon Lightsail オブジェクトストレージサービスのバケットのアクセスログの形式
   - Amazon Lightsail オブジェクトストレージサービスのバケットへのアクセスのログ記録を有効にする
   - Amazon Lightsailでのバケットのアクセスログを使用するリクエストの特定

6. Lightsail でバケットを管理する機能をユーザーに付与する IAM ポリシーを作成します。詳細については、「Amazon Lightsail でバケットを管理する IAM ポリシー」を参照してください。
7. バケット内のオブジェクトにラベルを付けて識別する方法について説明します。詳細については、「Amazon Lightsail でのオブジェクトキー名を理解する」を参照してください。
8. ファイルをアップロードしてバケット内のオブジェクトを管理する方法について説明します。詳細については、以下のガイドを参照してください。
   - Amazon Lightsail のバケットにファイルをアップロードする
   - マルチパートアップロードを使用した Amazon Lightsail のバケットへのファイルのアップロード
   - Amazon Lightsail のバケット内のオブジェクトの表示
   - Amazon Lightsail のバケット内のオブジェクトのコピーまたは移動
   - Amazon Lightsail のバケットからのオブジェクトのダウンロード
   - Amazon Lightsail のバケット内のオブジェクトのフィルタリング
   - Amazon Lightsail のバケット内のオブジェクトのタグ付け
   - Amazon Lightsail のバケット内のオブジェクトの削除
9. オブジェクトのバージョニングを有効にすると、バケットに保存されたあらゆるオブジェクトのあらゆるバージョンを保存、取得、復元します。詳細については、「Amazon Lightsail のバケットでのオブジェクトのバージョニングの有効化と一時停止」を参照してください。
10. オブジェクトのバージョニングを有効にすると、バケット内のオブジェクトの以前のバージョンを復元できます。詳細については、「Amazon Lightsail のバケット内のオブジェクトの以前のバージョンの復元」を参照してください。
11. バケットの使用率を監視します。詳細については、「Amazon Lightsail でのバケットのメトリクスの表示」を参照してください。
12. バケットの使用率がしきい値を超えたときにバケットメトリクスが通知されるよう、アラームを設定します。詳細については、「Amazon Lightsail でのバケットメトリクスアラームの作成」を参照してください。
13. ストレージとネットワーク転送量が不足している場合は、バケットのストレージプランを変更します。詳細については、「Amazon Lightsail のバケットのプランの変更」を参照してください。
14. バケットを他のリソースに接続する方法について説明します。詳細については、以下のチュートリアルを参照してください。
   - チュートリアル: WordPress インスタンスの Amazon Lightsail バケットへの接続
   - チュートリアル: Lightsail コンテンツ配信ネットワークディストリビューションでの Amazon Lightsail バケットの使用

15.使用しなくなったバケットを削除します。詳細については、「Amazon Lightsail でのバケットの削除」を参照してください。

トピック

- Lightsail バケットのメトリクスアラームを作成する

Lightsail バケットのメトリクスアラームを作成する

単一のバケットメトリクスを監視するAmazon Lightsail アラームを作成できます。アラームは、指定したしきい値を基準にしたメトリクスの値に基づいて通知するように設定できます。通知は、Lightsail コンソールに表示されるバナー、メールアドレスに送信される メール、携帯電話番号に送信される SMS テキストメッセージです。アラームの詳細については、「アラーム」を参照してください。

目次

バケットアラームの制限

アラームには、以下の制限が適用されます。

- メトリクスごとに 2 つのアラームを設定できます。
- アラームは 5 分間隔で評価され、アラームの各データポイントは、集計されたメトリクスデータの 5 分間隔を表します。
- アラームの状態が OK に変わったときに通知するようにアラームを設定できるのは、メールや SMS テキストメッセージで通知するようにアラームを設定した場合だけです。
- OK アラーム通知をテストできるのは、メールや SMS テキストメッセージで通知するようにアラームを設定した場合だけです。

- アラームの状態が INSUFFICIENT_DATA に変わったときに通知するようにアラームを設定できるのは、メールや SMS テキストメッセージで通知するようにアラームを設定した場合と、欠落しているデータポイントに対して [欠落しているデータを評価しない] オプションを選択した場合だけです。
- 通知をテストできるのは、アラームが OK 状態の場合だけです。

バケットアラームの設定に関するベストプラクティス

バケットのメトリクスアラームを設定する前に、受けたい通知を決めておきます。例えば、バケットサイズメトリクスを念頭に置くと、バケットがほぼ満杯になったときに通知を受けることが可能です。バケットの現在のプランに 5 GB のストレージスペースが含まれている場合は、バケットサイズメトリクスが 4.5 GB に達すると通知されます。バケットプランを増量しなければいけなくなる前に通知されます。

デフォルトのアラーム設定

Lightsail コンソールで新しいアラームを追加すると、デフォルトのアラーム設定があらかじめ入力されています。　これは、選択したメトリクスの推奨アラーム設定です。ただし、デフォルトのアラーム設定がリソースに適していることを確認する必要があります。例えば、バケットサイズのバイトのメトリクスに対するデフォルトのアラームしきい値は 75 GB [以上]です。ただし、ストレージスペースが 5 GB しかないように設定されている場合、リクエストのしきい値はバケットに対して高すぎる可能性があります。アラームのしきい値を、4.5 GB以上にするとよいでしょう。

Lightsail コンソールを使用してバケットメトリクスアラームを作成する

Lightsail コンソールを使用してバケットメトリクスアラームを作成するには、次のステップを実行します。

1. Lightsail コンソールにサインインします。
2. Lightsail のホームページで [Storage] (ストレージ) タブを選択します。
3. 作成するアラームのバケットの名前を選択します。
4. バケット管理ページで メトリクスタブを選択します。
5. [Metrics Graphs (メトリクスグラフ)] 見出しの下のドロップダウンメニューで、アラームを作成するメトリクスを選択します。詳細については、「リソースのメトリクス」を参照してください。
6. アラームセクションのページでアラームの追加を選択します。

7. ドロップダウンメニューから比較演算子の値を選択します。例の値は、greater than or equal to、greater than、less than、less than or equal to です。
8. アラームのしきい値を入力します。
9. アラームへのデータポイントを入力します。
10. 評価期間を選択します。期間は、5 分から 24 時間まで 5 分単位で指定できます。
11. 次のいずれかの通知方法を選択します。

    - メール — アラームの状態が ALARM に変わると、メールで通知されます。
    - SMS テキストメッセージ — アラームの状態が ALARM に変わると、SMS テキストメッセージによって通知されます。SMS メッセージングは、すべての AWS リージョン でサポートされているわけではなく、また、一部の国/地域には SMS テキストメッセージを送信できません。詳細については、「SMS テキストメッセージングのサポート」を参照してください。

    > **Note**
    >
    > メールまたは SMS による通知を選択したが、リソースの AWS リージョン で通知の連絡先をまだ設定していない場合は、メールアドレスまたは携帯電話番号を追加する必要があります。詳細については、「通知」を参照してください。

12. (オプション) [Send me a notification when the alarm state change to OK (アラームの状態が OK に変わったときに通知を送信する)] を選択して、アラームの状態が OK に変わったときに通知を受け取ります。このオプションは、メールまたは SMS テキストメッセージで通知されるように選択した場合にのみ使用できます。
13. (オプション) [Advanced settings (詳細設定)] を選択し、次のいずれかのオプションを選択します。

    - アラームによる欠落データの処理方法を選択します。次のオプションを使用できます。
      - しきい値の範囲内にないと仮定する(しきい値を超過) — 欠落しているデータポイントは「不良」として扱われ、しきい値を超えています。
      - しきい値内にあると仮定する (しきい値を超過していない) — 欠落しているデータポイントは、「良い」と見なされ、しきい値の範囲内で処理されます。
      - 最後の正常なデータポイントの値を使用する (無視して現在のアラーム状態を維持する) — 現在のアラーム状態が維持されます。
      - 評価しない (欠落しているデータを欠落しているデータとして扱う) — 状態を変更するかどうかを評価する際に、欠落データポイントを考慮に入れません。

- [Send a notification if there is insufficient data (データが不足している場合に通知を送信)] を選択して、アラームの状態が INSUFFICIENT_DATA に変わったときに通知を受け取ります。このオプションは、メールまたは SMS テキストメッセージで通知されるように選択した場合にのみ使用できます。

14. [Create (作成)] を選択してアラームを追加します。

    後でアラームを編集するには、編集するアラームの横にある省略記号アイコン (⋮) を選択し、[アラームの編集] を選択します。

Lightsail コンソールを使用してバケットメトリクスアラームをテストする

Lightsail コンソールを使用してアラームをテストするには、次の手順を実行します。アラームをテストして、アラームがトリガーされたときに メールや SMS テキストメッセージを受信するなど、設定された通知オプションが機能していることを確認します。

1. Lightsail コンソールにサインインします。
2. Lightsail のホームページで [Storage] (ストレージ) タブを選択します。
3. アラームをテストするバケットの名前を選択します。
4. バケット管理ページで メトリクスタブを選択します。
5. [Metrics Graphs (メトリクスグラフ)] 見出しの下のドロップダウンメニューで、アラームをテストするメトリクスを選択します。
6. ページの [アラーム] セクションまでスクロールし、テストするアラームの横にある省略記号アイコン (⋮) を選択します。
7. 以下のオプションのいずれかを選択します。

    - [アラーム通知のテスト] - このオプションを選択すると、アラームの状態が ALARM に変わったときの通知をテストできます。
    - [OK 通知のテスト] — このオプションを選択すると、アラームの状態が OK に変わったときの通知をテストできます。

> **Note**
>
> これらのオプションのいずれかが使用できない場合は、アラームの通知オプションが設定されていないか、アラームが現在 ALARM 状態になっている可能性があります。詳細については、バケットアラームの制限を参照してください。

選択したテストオプションに応じて、アラームが一時的に ALARM または OK 状態に変化し、アラームの通知方法として設定した内容に応じて、メールまたは SMS テキストメッセージが送信されます。通知バナーは、ALARM 通知のテストを選択した場合にのみ Lightsail コンソールに表示されます。OK 通知のテストを選択した場合、通知バナーは表示されません。アラームは、通常、数秒後に実際の状態に戻ります。

バケットアラームの作成後の次のステップ

バケットアラームに対して実行できる追加のタスクがいくつかあります。

- 通知の受信を停止するには、Lightsail から メールと携帯電話を削除します。詳細については、「通知連絡先の削除」を参照してください。また、アラームを無効化または削除して、特定のアラームの通知の受信を停止することもできます。詳細については、「メトリクスアラームを削除または無効化する」を参照してください。

Lightsail コンテナサービスメトリクスの表示

Amazon Lightsail コンテナサービスの作成後、サービスの管理ページのメトリクスタブでメトリクスグラフを表示できます。メトリクスのモニタリングは、リソースの信頼性、可用性、パフォーマンスを維持する上で重要な要素です。リソースから定期的にメトリクスデータをモニタリングして収集し、マルチポイント障害が発生した場合に、より簡単にデバッグできるようにします。メトリクスの詳細については、「Amazon Lightsail のメトリクス」を参照してください。

リソースを監視するときは、環境内の通常のリソースパフォーマンスのベースラインを確立する必要があります。

> **Note**
>
> アラームと通知は、現在、コンテナサービスメトリクスではサポートされていません。

コンテナサービスのメトリクス

以下のコンテナサービスメトリクスを使用できます。

- CPU 使用率 — コンテナサービスの全ノードで現在使用されている、コンピュータ単位の平均比率。このメトリクスは、コンテナサービス上のコンテナを実行するのに必要な処理能力を特定します。
- メモリ使用率 — コンテナサービスの全ノードで現在使用されているメモリの平均比率。このメトリクスは、コンテナサービスでコンテナを実行するのに必要なメモリを表します。

> **Note**
>
> 新しいデプロイを作成すると、コンテナサービスの既存の使用率メトリクスが消え、新しい現在のデプロイのメトリクスだけが表示されます。

Lightsail コンソールでコンテナサービスのメトリクスを表示する

Lightsail コンソールでコンテナサービスメトリクスを表示するには、以下の手順を実行します。

1. Lightsail コンソールにサインインします。
2. Lightsail コンソールのホームページで、[Containers] (コンテナ) タブを選択します。
3. メトリクスを表示するコンテナの名前を選択します。
4. コンテナ サービス 管理ページで [Metrics] (メトリクス) タブを選択します。
5. [メトリクス] グラフの見出しの下のドロップダウンメニューで、表示するメトリクスを選択します。

   グラフには、選択したメトリクスのデータポイントが視覚的に表示されます。
6. メトリクスグラフでは、次のアクションを実行できます。
   - グラフの表示を変更して、1 時間、6 時間、1 日、1 週間、2 週間のデータを表示します。
   - データポイント上にカーソルを置くと、そのデータポイントに関する詳細情報が表示されます。

   > **Note**
   >
   > アラームと通知は、現在、コンテナサービスメトリクスではサポートされていません。

Lightsail データベースメトリクスを表示する

Amazon Lightsail でデータベースを起動すると、データベースの管理ページの [Metrics (メトリクス)] タブでメトリクスグラフを表示できるようになります。メトリクスのモニタリングは、リソースの信頼性、可用性、パフォーマンスを維持する上で重要な要素です。リソースから定期的にメトリクスデータをモニタリングして収集し、マルチポイント障害が発生した場合に、より簡単にデバッグできるようにします。メトリクスの詳細については、「メトリクス」を参照してください。

リソースを監視するときは、環境内の通常のリソースパフォーマンスのベースラインを確立する必要があります。ベースラインを確立したら、リソースのパフォーマンスが指定のしきい値を超えたときに通知するように Lightsail コンソールでアラームを設定できます。詳細については、「通知」および「アラーム」を参照してください。

目次

データベースメトリクス

次のデータベースメトリクスを使用できます。

- CPU 使用率 (**CPUUtilization**) — データベースで現在使用されている CPU 使用率の割合。
- データベース接続 (**DatabaseConnections**) — 使用中のデータベース接続の数。
- ディスクのキューの深度 (**DiskQueueDepth**) — ディスクへのアクセスを待機している未処理の IO (読み取り/書き込みリクエスト) の数。
- 空きストレージ容量 (**FreeStorageSpace**) — 使用可能なストレージの容量。
- ネットワーク受信スループット (**NetworkReceiveThroughput**) — モニタリングとレプリケーションに使用する顧客データベーストラフィックと AWS トラフィックの両方を含む、データベースの受信ネットワークトラフィック。
- ネットワーク送信スループット (**NetworkTransmitThroughput**) — モニタリングとレプリケーションに使用する顧客データベーストラフィックと AWS トラフィックの両方を含む、データベースの送信ネットワークトラフィック。

Lightsail コンソールでのデータベースメトリクスの表示

Lightsail コンソールでデータベースメトリクスを表示するには、次の手順を実行します。

1. Lightsail コンソールにサインインします。
2. Lightsail のホームページで [データベース] タブを選択します。
3. メトリクスを表示するデータベースの名前を選択します。
4. データベース管理ページで [Metrics (メトリクス)] タブを選択します。
5. [Metrics graph (メトリクスグラフ)] 見出しの下のドロップダウンメニューで、表示するメトリクスを選択します。

   グラフには、選択したメトリクスのデータポイントが視覚的に表示されます。
6. メトリクスグラフでは、次のアクションを実行できます。

   - グラフの表示を変更して、1 時間、6 時間、1 日、1 週間、2 週間のデータを表示します。
   - データポイント上にカーソルを置くと、そのデータポイントに関する詳細情報が表示されます。
   - 指定したしきい値をメトリクスが超えたときに通知される、選択したメトリクスのアラームを追加します。詳細については、「アラーム」および「データベースメトリクスアラームの作成」を参照してください。

データベースメトリクスの表示後の次のステップ

データベースメトリクスに対して実行できる追加のタスクがいくつかあります。

- 指定したしきい値をメトリクスが超えたときに通知される、選択したメトリクスのアラームを追加します。詳細については、「アラーム」および「データベースメトリクスアラームの作成」を参照してください。
- アラームが作動すると、通知バナーが Lightsail コンソールに表示されます。メールおよび SMS テキストメッセージで通知を受けるには、リソースを監視する各 AWS リージョン で、メールアドレスと携帯電話番号を通知連絡先として追加する必要があります。詳細については、「通知連絡先の追加」を参照してください 。
- 通知の受信を停止するには、Lightsail から メールと携帯電話を削除します。詳細については、「メトリクスアラームを削除または無効化する」を参照してください。また、アラームを無効化または削除して、特定のアラームの通知の受信を停止することもできます。詳細については、「メトリクスアラームを削除または無効化する」を参照してください。

トピック

- Lightsail データベースのメトリクスにアラームを作成する

Lightsail データベースのメトリクスにアラームを作成する

1 つのデータベースメトリクスを監視する Amazon Lightsail アラームを作成できます。アラームは、指定したしきい値を基準にしたメトリクスの値に基づいて通知するように設定できます。通知は、Lightsail コンソールに表示されるバナー、メールアドレスに送信される メール、携帯電話番号に送信される SMS テキストメッセージです。アラームの詳細については、「アラーム」を参照してください。

目次

データベースアラームの制限

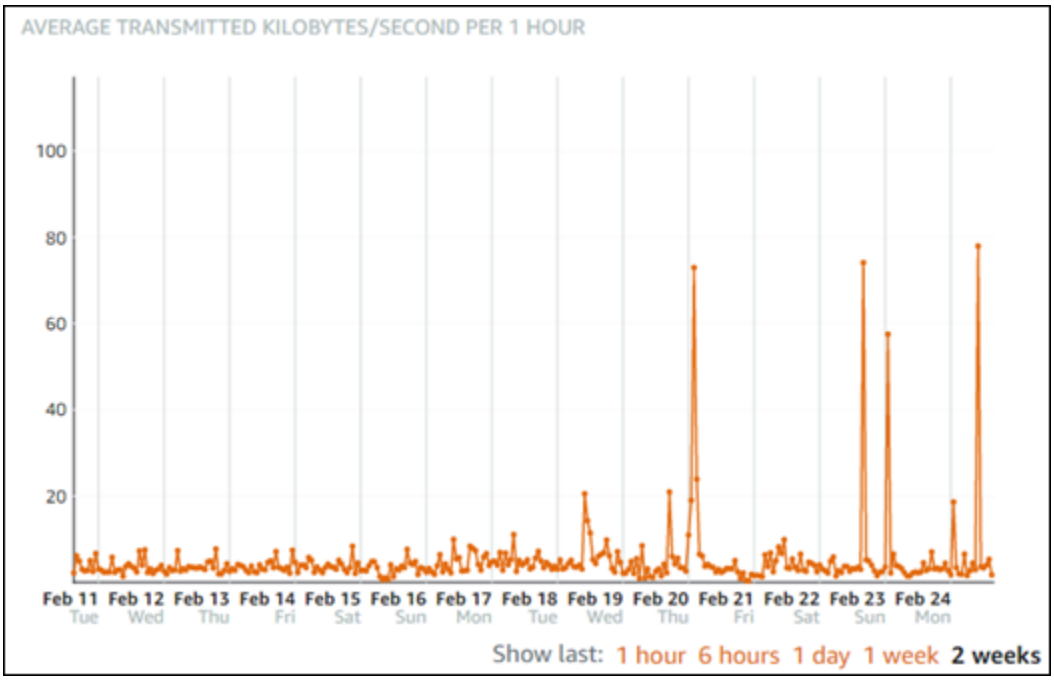
アラームには、以下の制限が適用されます。

- メトリクスごとに 2 つのアラームを設定できます。
- アラームは 5 分間隔で評価され、アラームの各データポイントは、集計されたメトリクスデータの 5 分間隔を表します。
- アラームの状態が OK に変わったときに通知するようにアラームを設定できるのは、メールや SMS テキストメッセージで通知するようにアラームを設定した場合だけです。
- OK アラーム通知をテストできるのは、メールや SMS テキストメッセージで通知するようにアラームを設定した場合だけです。
- アラームの状態が INSUFFICIENT_DATA に変わったときに通知するようにアラームを設定できるのは、メールや SMS テキストメッセージで通知するようにアラームを設定した場合と、欠落しているデータポイントに対して [欠落しているデータを評価しない] オプションを選択した場合だけです。

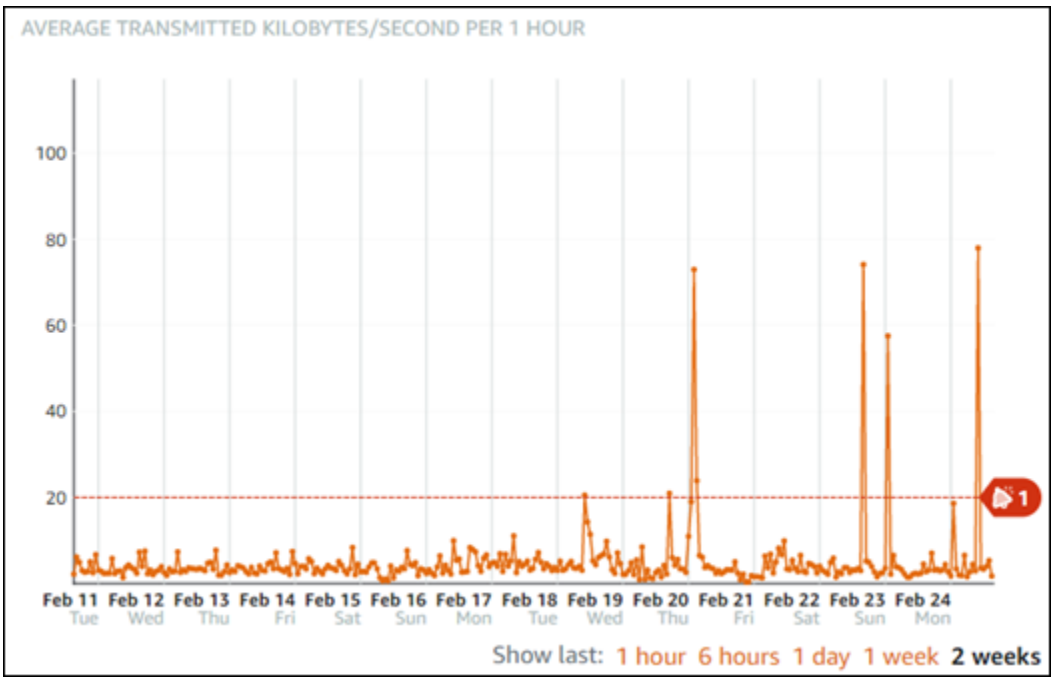
- 通知をテストできるのは、アラームが OK 状態の場合だけです。

データベースアラームの設定に関するベストプラクティス

データベースのメトリクスアラームを設定する前に、メトリクスの履歴データを表示する必要があります。過去 2 週間のメトリクスの低レベル、中間レベル、高レベルを識別します。次のネットワーク送信スループット (NetworkTransmitThroughput) メトリクスグラフの例では、低レベルは 0 ~ 10 KB/秒、中間レベルは 1 時間あたり 10 ~ 20 KB/秒、高レベルは 1 時間あたり 20 ~ 80 KB/秒の間です。



アラームしきい値を低レベル範囲 (1 時間あたり 5 KB/秒など) 以上に設定すると、より頻繁に、不要なアラーム通知が送信されます。アラームしきい値を高レベルの範囲 (たとえば、1 時間あたり 20 KB) 以上に設定した場合、アラーム通知の頻度は低くなりますが、調査がさらに意義のあるものになる場合があります。アラームを設定して有効にすると、次の例に示すように、しきい値を表すアラームラインがグラフに表示されます。1 というラベルの付いたアラームラインはアラーム 1 のしきい値を表し、2 というラベルのアラームラインはアラーム 2 のしきい値を表します。



デフォルトのアラーム設定

Lightsail コンソールで新しいアラームを追加すると、デフォルトのアラーム設定があらかじめ入力されています。これは、選択したメトリクスの推奨アラーム設定です。ただし、デフォルトのアラーム設定がリソースに適していることを確認する必要があります。たとえば、空きストレージ容量 (`FreeStorageSpace`) メトリクスのデフォルトのアラームしきい値は、過去 5 分間 1 回で 5 バイト未満です。ただし、その空きストレージ容量のしきい値は、データベースに対して低すぎる可能性があります。アラームのしきい値を、過去 5 分以内に 1 回で 4 GB 未満に変更することもできます。

Lightsail コンソールを使用してデータベースのメトリクスアラームを作成する

Lightsail コンソールを使用してデータベースメトリクスアラームを作成するには、次の手順を実行します。

1. Lightsail コンソールにサインインします。
2. Lightsail のホームページで [データベース] タブを選択します。
3. アラームを作成するデータベースの名前を選択します。
4. データベース管理ページで [Metrics (メトリクス)] タブを選択します。
5. [Metrics Graphs (メトリクスグラフ)] 見出しの下のドロップダウンメニューで、アラームを作成するメトリクスを選択します。詳細については、「リソースのメトリクス」を参照してください。

6. アラームセクションのページでアラームの追加を選択します。
7. ドロップダウンメニューから比較演算子の値を選択します。例の値は、greater than or equal to、greater than、less than、less than or equal to です。
8. アラームのしきい値を入力します。
9. アラームへのデータポイントを入力します。
10. 評価期間を選択します。期間は、5 分から 24 時間まで 5 分単位で指定できます。
11. 次のいずれかの通知方法を選択します。

    - メール — アラームの状態が ALARM に変わると、メールで通知されます。
    - SMS テキストメッセージ — アラームの状態が ALARM に変わると、SMS テキストメッセージによって通知されます。SMS メッセージングは、Lightsail リソースを作成できるすべての AWS リージョンでサポートされているわけではなく、また、一部の国/地域には SMS テキストメッセージを送信できません。詳細については、「SMS テキストメッセージングのサポート」を参照してください。

    > **Note**
    >
    > メールまたは SMS による通知を選択したが、リソースの AWSリージョンで通知の連絡先をまだ設定していない場合は、メールアドレスまたは携帯電話番号を追加する必要があります。詳細については、「通知」を参照してください。

12. (オプション) [Send me a notification when the alarm state change to OK (アラームの状態が OK に変わったときに通知を送信する)] を選択して、アラームの状態が OK に変わったときに通知を受け取ります。このオプションは、メールまたは SMS テキストメッセージで通知されるように選択した場合にのみ使用できます。
13. (オプション) [Advanced settings (詳細設定)] を選択し、次のいずれかのオプションを選択します。

    - アラームによる欠落データの処理方法を選択します。次のオプションを使用できます。
      - しきい値の範囲内にないと仮定する(しきい値を超過) — 欠落しているデータポイントは「不良」として扱われ、しきい値を超えています。
      - しきい値内にあると仮定する (しきい値を超過していない) — 欠落しているデータポイントは、「良い」と見なされ、しきい値の範囲内で処理されます。
      - 最後の正常なデータポイントの値を使用する (無視して現在のアラーム状態を維持する) — 現在のアラーム状態が維持されます。

- 評価しない (欠落しているデータを欠落しているデータとして扱う) — 状態を変更するかどうかを評価する際に、欠落データポイントを考慮に入れません。

- [Send a notification if there is insufficient data (データが不足している場合に通知を送信)] を選択して、アラームの状態が INSUFFICIENT_DATA に変わったときに通知を受け取ります。このオプションは、メールまたは SMS テキストメッセージで通知されるように選択した場合にのみ使用できます。

14. [Create (作成)] を選択してアラームを追加します。

    後でアラームを編集するには、編集するアラームの横にある省略記号アイコン (⋮) を選択し、[アラームの編集] を選択します。

Lightsail コンソールを使用したデータベースメトリクスアラームのテスト

Lightsail コンソールを使用してアラームをテストするには、次の手順を実行します。アラームをテストして、アラームがトリガーされたときに メールや SMS テキストメッセージを受信するなど、設定された通知オプションが機能していることを確認します。

1. Lightsail コンソールにサインインします。
2. Lightsail のホームページで [データベース] タブを選択します。
3. アラームを表示するデータベースの名前を選択します。
4. データベース管理ページで [Metrics (メトリクス)] タブを選択します。
5. [Metrics Graphs (メトリクスグラフ)] 見出しの下のドロップダウンメニューで、アラームをテストするメトリクスを選択します。
6. ページの [アラーム] セクションまでスクロールし、テストするアラームの横にある省略記号アイコン (⋮) を選択します。
7. 以下のオプションのいずれかを選択します。

    - [アラーム通知のテスト] - このオプションを選択すると、アラームの状態が ALARM に変わったときの通知をテストできます。
    - [OK 通知のテスト] — このオプションを選択すると、アラームの状態が OK に変わったときの通知をテストできます。

> **Note**
>
> これらのオプションのいずれかが使用できない場合は、アラームの通知オプションが設定されていないか、アラームが現在 ALARM 状態になっている可能性があります。詳細については、「データベースアラームの制限」を参照してください。

選択したテストオプションに応じて、アラームが一時的に ALARM または OK 状態に変化し、アラームの通知方法として設定した内容に応じて、メールまたは SMS テキストメッセージが送信されます。通知バナーは、ALARM 通知のテストを選択した場合にのみ Lightsail コンソールに表示されます。OK 通知のテストを選択した場合、通知バナーは表示されません。アラームは、通常、数秒後に実際の状態に戻ります。

データベースアラームの作成後の次の手順

データベースアラームに対して実行できる追加のタスクがいくつかあります。

- 通知の受信を停止するには、Lightsail から メールと携帯電話を削除します。詳細については、「通知連絡先の削除」を参照してください。また、アラームを無効化または削除して、特定のアラームの通知の受信を停止することもできます。詳細については、「メトリクスアラームを削除または無効化する」を参照してください。

Lightsail ディストリビューションのメトリクスを表示

Amazon Lightsail でディストリビューションを作成した後、 ディストリビューションの管理ページの [Metrics] (メトリクス) タブで、そのメトリクスのグラフを見ることができます。メトリクスのモニタリングは、リソースの信頼性、可用性、パフォーマンスを維持する上で重要な要素です。リソースから定期的にメトリクスデータをモニタリングして収集し、マルチポイント障害が発生した場合に、より簡単にデバッグできるようにします。メトリクスの詳細については、「メトリクス」を参照してください。

リソースを監視するときは、環境内の通常のリソースパフォーマンスのベースラインを確立する必要があります。その後、リソースのパフォーマンスが指定のしきい値を超えたときに通知するように、Lightsail コンソールでアラームを設定できます。詳細については、「通知」および「アラーム」を参照してください。

目次

ディストリビューションメトリクス

次のディストリビューションメトリクスが利用可能です。

- リクエスト — すべての HTTP メソッド、および HTTP と HTTPS 両方のリクエストについて、ディストリビューションが受信したビューワーリクエストの総数。
- アップロードされたバイト数 — POST リクエストと PUT リクエストを使用して、ディストリビューションによってオリジンにアップロードされたバイト数。
- ダウンロードされたバイト数 — GET リクエスト、HEAD リクエスト、および OPTIONS リクエストに対してビューワーがダウンロードしたバイト数。
- トータルエラー率 — レスポンスの HTTP ステータスコードが 4xx または 5xx であったすべてのビューワーリクエストの割合 (%)。
- HTTP 4xx トータルエラー率 — レスポンスの HTTP ステータスコードが 4xx であったすべてのビューワーリクエストの割合 (%)。このような場合、クライアントまたはクライアントビューワーでエラーが発生した可能性があります。たとえば、ステータスコード 404 (Not Found) は、クライアントが、検出できないオブジェクトをリクエストしたことを意味します。
- HTTP 5xx トータルエラー率 — レスポンスの HTTP ステータスコードが 5xx であったすべてのビューワー リクエストの割合 (%)。このような場合、オリジンサーバーはリクエストを満たしませんでした。たとえば、ステータスコード 503 (Service Unavailable) は、オリジンサーバーが現在利用できないことを意味します。

ディストリビューションメトリクスを Lightsail コンソールに表示する

Lightsail コンソールでディストリビューションメトリクスを表示するには、次の手順を実行します。

1. Lightsail コンソールにサインインします。
2. Lightsail のホームページで、[ネットワーキング] タブを選択します。
3. メトリクスを表示するディストリビューションの名前を選択します。
4. ディストリビューション管理ページで [Metrics] (メトリクス) タブを選択します。

5. [Metrics graph (メトリクスグラフ)] 見出しの下のドロップダウンメニューで、表示するメトリクスを選択します。

   グラフには、選択したメトリクスのデータポイントが視覚的に表示されます。
6. メトリクスグラフでは、次のアクションを実行できます。

   - グラフの表示を変更して、1 時間、6 時間、1 日、1 週間、2 週間のデータを表示します。
   - データポイント上にカーソルを置くと、そのデータポイントに関する詳細情報が表示されます。
   - 指定したしきい値をメトリクスが超えたときに通知される、選択したメトリクスのアラームを追加します。詳細については、「アラーム」および「インスタンスのメトリクスアラームを作成する」を参照してください。

ディストリビューションメトリクスの表示後の次のステップ

ディストリビューションメトリクスに対して実行できる追加のタスクがいくつかあります。

- 指定したしきい値をメトリクスが超えたときに通知される、選択したメトリクスのアラームを追加します。詳細については、「アラーム」および「ディストリビューションメトリクスアラームの作成」を参照してください。
- アラームが作動すると、通知バナーが Lightsail コンソールに表示されます。メールおよび SMS テキストメッセージで通知を受けるには、リソースを監視する各 AWS リージョン で、メールアドレスと携帯電話番号を通知連絡先として追加する必要があります。詳細については、「通知連絡先を追加する」を参照してください。
- 通知の受信を停止するには、Lightsail から メールと携帯電話を削除します。詳細については、「メトリクスアラームを削除または無効化する」を参照してください。また、アラームを無効化または削除して、特定のアラームの通知の受信を停止することもできます。詳細については、「メトリクスアラームを削除または無効化する」を参照してください。

トピック

- Lightsail ディストリビューションのメトリクスアラームを作成する

Lightsail ディストリビューションのメトリクスアラームを作成する

単一のディストリビューションメトリクスをモニタリングする Amazon Lightsail アラームを作成できます。アラームは、指定したしきい値を基準にしたメトリクスの値に基づいて通知するように設

定できます。通知は、Lightsail コンソールに表示されるバナー、メールアドレスに送信される メール、携帯電話番号に送信される SMS テキストメッセージです。アラームの詳細については、「アラーム」を参照してください。

目次

ディストリビューションアラームの制限

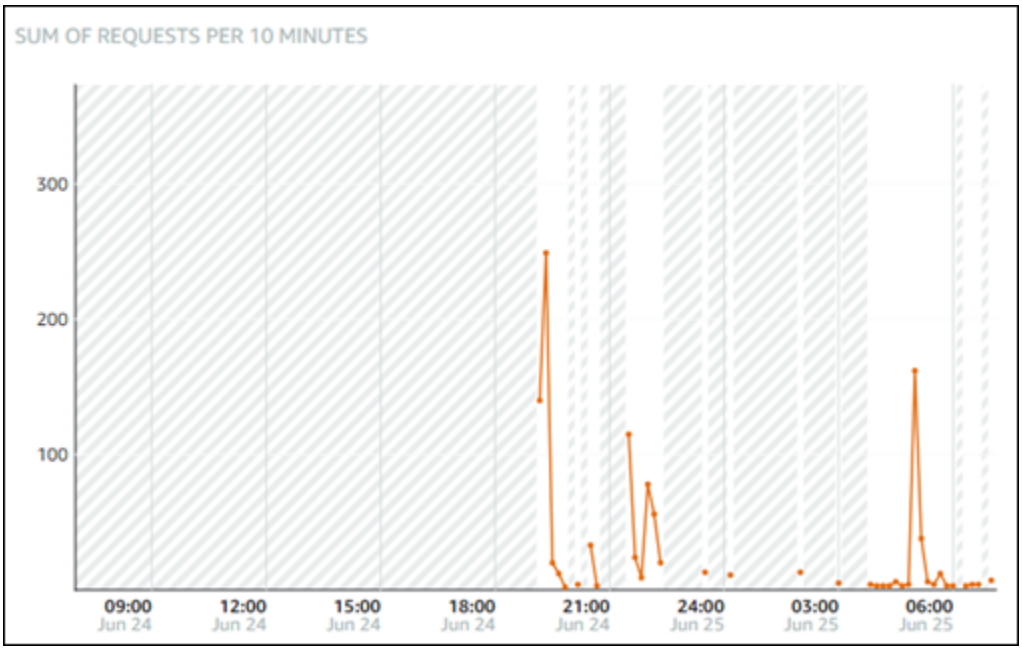
アラームには、以下の制限が適用されます。

- メトリクスごとに 2 つのアラームを設定できます。
- アラームは 5 分間隔で評価され、アラームの各データポイントは、集計されたメトリクスデータの 5 分間隔を表します。
- アラームの状態が OK に変わったときに通知するようにアラームを設定できるのは、メールや SMS テキストメッセージで通知するようにアラームを設定した場合だけです。
- OK アラーム通知をテストできるのは、メールや SMS テキストメッセージで通知するようにアラームを設定した場合だけです。
- アラームの状態が `INSUFFICIENT_DATA` に変わったときに通知するようにアラームを設定できるのは、メールや SMS テキストメッセージで通知するようにアラームを設定した場合と、欠落しているデータポイントに対して [欠落しているデータを評価しない] オプションを選択した場合だけです。
- 通知をテストできるのは、アラームが OK 状態の場合だけです。

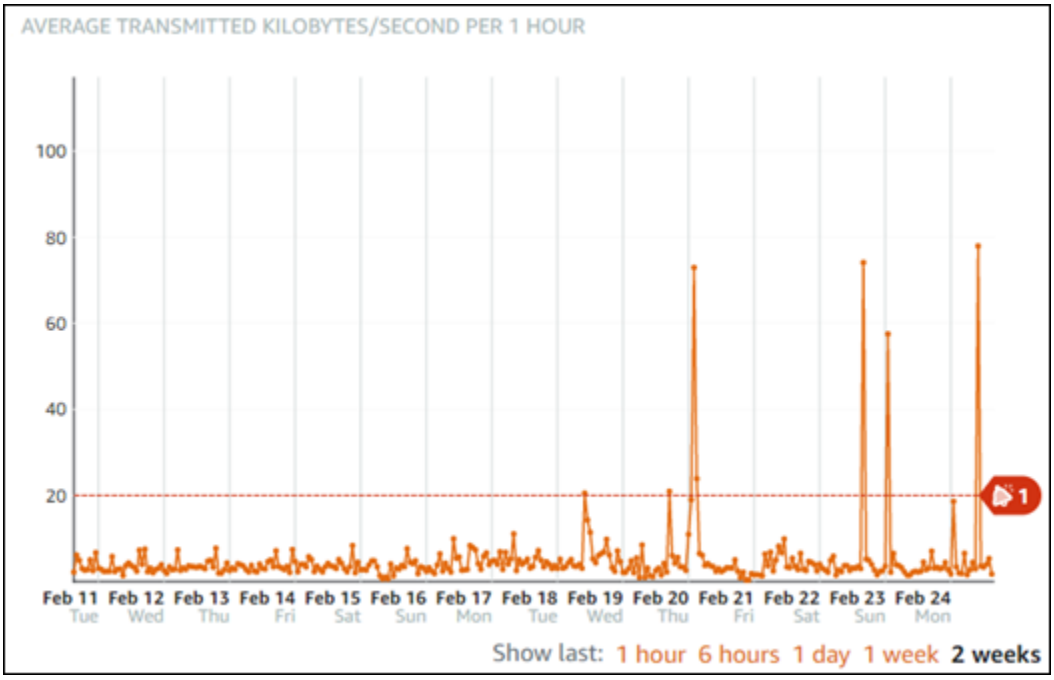
ディストリビューションアラームの設定に関するベストプラクティス

ディストリビューションのメトリクスアラームを設定する前に、メトリクスのデータ履歴を表示する必要があります。過去 2 週間のメトリクスの低レベル、中間レベル、高レベルを識別します。以下

のリクエストでメトリクスグラフの例では、低レベルは 0～10 のリクエスト、中間レベルは 10～50 リクエスト、高レベルは 50～250 リクエストとなります。



アラームを低レベル範囲 (例えば 5 リクエスト) 以上に設定すると、アラーム通知が頻繁に送信され、不要なアラーム通知が発生する可能性があります。アラームしきい値を高レベルの範囲 (例えば 150 リクエスト) 以上に設定した場合、アラーム通知の頻度は低くなりますが、さらに調査することが重要になります。アラームを設定して有効にすると、次の例に示すように、しきい値を表すアラームラインがグラフに表示されます。1 というラベルの付いたアラームラインはアラーム 1 のしきい値を表し、2 というラベルのアラームラインはアラーム 2 のしきい値を表します。



デフォルトのアラーム設定

Lightsail コンソールで新しいアラームを追加すると、デフォルトのアラーム設定があらかじめ入力されています。これは、選択したメトリクスの推奨アラーム設定です。ただし、デフォルトのアラーム設定がリソースに適していることを確認する必要があります。リクエストのメトリクスのデフォルトのアラームしきい値が、過去 15 分以内に 3 回で 45 リクエスト[以上]の場合。リクエストのしきい値は、ディストリビューションに対して低すぎる可能性があります。アラームのしきい値を、過去 15 分以内に 3 回で 150 リクエスト以上に変更することが望ましいです。

Lightsail コンソールを使用してディストリビューションメトリクスアラームを作成する

Lightsail コンソールを使用してディストリビューションメトリクスアラームを作成するには、以下のステップを実行します。

1. Lightsail コンソールにサインインします。
2. Lightsail のホームページで、[ネットワーキング] タブを選択します。
3. アラームを作成する対象のディストリビューションを選択します。
4. ディストリビューション管理ページでメトリクスタブを選択します。
5. [Metrics Graphs (メトリクスグラフ)] 見出しの下のドロップダウンメニューで、アラームを作成するメトリクスを選択します。詳細については、「リソースのメトリクス」を参照してください。
6. アラームセクションのページでアラームの追加を選択します。
7. ドロップダウンメニューから比較演算子の値を選択します。例の値は、greater than or equal to、greater than、less than、less than or equal to です。
8. アラームのしきい値を入力します。
9. アラームへのデータポイントを入力します。
10. 評価期間を選択します。期間は、5 分から 24 時間まで 5 分単位で指定できます。
11. 次のいずれかの通知方法を選択します。

    - メール — アラームの状態が ALARM に変わると、メールで通知されます。
    - SMS テキストメッセージ — アラームの状態が ALARM に変わると、SMS テキストメッセージによって通知されます。SMS メッセージングは、Lightsail リソースを作成できるすべての AWS リージョンでサポートされているわけではなく、また、一部の国/地域には SMS テキストメッセージを送信できません。詳細については、「SMS テキストメッセージングのサポート」を参照してください。

> **Note**
>
> メールまたは SMS による通知を選択したが、リソースの AWS リージョン で通知の連絡先をまだ設定していない場合は、メールアドレスまたは携帯電話番号を追加する必要があります。詳細については、「通知」を参照してください。

12. (オプション) [Send me a notification when the alarm state change to OK (アラームの状態が OK に変わったときに通知を送信する)] を選択して、アラームの状態が OK に変わったときに通知を受け取ります。このオプションは、メールまたは SMS テキストメッセージで通知されるように選択した場合にのみ使用できます。
13. (オプション) [Advanced settings (詳細設定)] を選択し、次のいずれかのオプションを選択します。
    - アラームによる欠落データの処理方法を選択します。次のオプションを使用できます。
      - しきい値の範囲内にないと仮定する(しきい値を超過) — 欠落しているデータポイントは「不良」として扱われ、しきい値を超えています。
      - しきい値内にあると仮定する (しきい値を超過していない) — 欠落しているデータポイントは、「良い」と見なされ、しきい値の範囲内で処理されます。
      - 最後の正常なデータポイントの値を使用する (無視して現在のアラーム状態を維持する) — 現在のアラーム状態が維持されます。
      - 評価しない (欠落しているデータを欠落しているデータとして扱う) — 状態を変更するかどうかを評価する際に、欠落データポイントを考慮に入れません。
    - [Send a notification if there is insufficient data (データが不足している場合に通知を送信)] を選択して、アラームの状態が INSUFFICIENT_DATA に変わったときに通知を受け取ります。このオプションは、メールまたは SMS テキストメッセージで通知されるように選択した場合にのみ使用できます。
14. [Create (作成)] を選択してアラームを追加します。

    後でアラームを編集するには、編集するアラームの横にある省略記号アイコン (⋮) を選択し、[アラームの編集] を選択します。

ディストリビューションメトリクスアラームをテストする

Lightsail コンソールを使用してアラームをテストするには、次の手順を実行します。アラームをテストして、アラームがトリガーされたときに メールや SMS テキストメッセージを受信するなど、設定された通知オプションが機能していることを確認します。

1. Lightsail コンソールにサインインします。
2. Lightsail のホームページで、[ネットワーキング] タブを選択します。
3. アラームをテストしたいディストリビューションの名前を選択します。
4. ディストリビューション管理ページでメトリクスタブを選択します。
5. [Metrics Graphs (メトリクスグラフ)] 見出しの下のドロップダウンメニューで、アラームをテストするメトリクスを選択します。
6. ページの [アラーム] セクションまでスクロールし、テストするアラームの横にある省略記号アイコン (⋮) を選択します。
7. 以下のオプションのいずれかを選択します。

   - [アラーム通知のテスト] - このオプションを選択すると、アラームの状態が ALARM に変わったときの通知をテストできます。
   - [OK 通知のテスト] — このオプションを選択すると、アラームの状態が OK に変わったときの通知をテストできます。

   > **Note**
   >
   > これらのオプションのいずれかが使用できない場合は、アラームの通知オプションが設定されていないか、アラームが現在 ALARM 状態になっている可能性があります。詳細については、ディストリビューションアラーム制限を参照してください。

   選択したテストオプションに応じて、アラームが一時的に ALARM または OK 状態に変化し、アラームの通知方法として設定した内容に応じて、メールまたは SMS テキストメッセージが送信されます。通知バナーは、ALARM 通知のテストを選択した場合にのみ Lightsail コンソールに表示されます。OK 通知のテストを選択した場合、通知バナーは表示されません。アラームは、通常、数秒後に実際の状態に戻ります。

ディストリビューションアラームの作成後の次のステップ

ディストリビューションアラームに対して実行できる追加のタスクがいくつかあります。

- 通知の受信を停止するには、Lightsail から メールと携帯電話を削除します。詳細については、「通知連絡先の削除」を参照してください。また、アラームを無効化または削除して、特定のアラームの通知の受信を停止することもできます。詳細については、「メトリクスアラームを削除または無効化する」を参照してください。

Lightsail ロードバランサーのヘルスメトリクスを表示する

Amazon Lightsail でロードバランサーを作成し、インスタンスにアタッチすると、ロードバランサーの管理ページの [Metrics (メトリクス)] タブでメトリクスグラフを表示できます。メトリクスのモニタリングは、リソースの信頼性、可用性、パフォーマンスを維持する上で重要な要素です。リソースから定期的にメトリクスデータをモニタリングして収集し、マルチポイント障害が発生した場合に、より簡単にデバッグできるようにします。メトリクスの詳細については、「メトリクス」を参照してください。

リソースを監視するときは、環境内の通常のリソースパフォーマンスのベースラインを確立する必要があります。ベースラインを確立したら、リソースのパフォーマンスが指定のしきい値を超えたときに通知するように Lightsail コンソールでアラームを設定できます。詳細については、「通知」および「アラーム」を参照してください。

目次

ロードバランサーのメトリクス

次のロードバランサーメトリクスを使用できます。

- 正常ホスト数 (**HealthyHostCount**) — 正常と見なされるターゲットインスタンスの数。
- 異常ホスト数 (**UnhealthyHostCount**) — 異常と見なされるターゲットインスタンスの数。
- ロードバランサー HTTP 4XX (HTTPCode_LB_4XX_Count) — ロードバランサーから発生した HTTP 4XX クライアントエラーコードの数。リクエストの形式が不正な場合、または不完全な場

合は、クライアントエラーが生成されます。これらのリクエストは、ターゲットインスタンスによって受信されませんでした。この数には、ターゲットインスタンスによって生成される応答コードは含まれません。

- ロードバランサー HTTP 5XX (HTTPCode_LB_5XX_Count) — ロードバランサーから発生した HTTP 5XX サーバーのエラーコードの数。これには、ターゲットインスタンスによって生成される応答コードは含まれません。ロードバランサーにアタッチされている正常なインスタンスがない場合、またはリクエストレートがインスタンスやロードバランサーの容量を超える場合 (スピルオーバー)、このメトリクスが報告されます。
- インスタンス HTTP 2XX (**HTTPCode_Instance_2XX_Count**) — ターゲットインスタンスによって生成された HTTP 2XX 応答コードの数。これには、ロードバランサーによって生成される応答コードは含まれません。
- インスタンス HTTP 3XX (**HTTPCode_Instance_3XX_Count**) — ターゲットインスタンスによって生成された HTTP 3XX 応答コードの数。これには、ロードバランサーによって生成される応答コードは含まれません。
- インスタンス HTTP 4XX (**HTTPCode_Instance_4XX_Count**) — ターゲットインスタンスによって生成された HTTP 4XX 応答コードの数。これには、ロードバランサーによって生成される応答コードは含まれません。
- インスタンス HTTP 5XX (**HTTPCode_Instance_5XX_Count**) — ターゲットインスタンスによって生成された HTTP 5XX 応答コードの数。これには、ロードバランサーによって生成される応答コードは含まれません。
- インスタンスからの応答時間 (**InstanceResponseTime**) — ロードバランサーがリクエストを送信してから、ターゲットインスタンスからの応答を受信するまでの経過時間 (秒)。
- クライアント TLS ネゴシエーションエラー数 (**ClientTLSNegotiationErrorCount**) — クライアントにより開始され、ロードバランサーによって生成された TLS エラーのためにロードバランサーとのセッションを確立しなかった、TLS 接続の数。暗号化またはプロトコルの不一致が原因である場合があります。
- リクエストの数 (**RequestCount**) — IPv4 経由で処理されたリクエストの数。この数には、ロードバランサーのターゲットインスタンスによって生成されたレスポンスを含むリクエストのみが含まれます。
- 拒否された接続数 (**RejectedConnectionCount**) — ロードバランサーが接続の最大数に達したため、拒否された接続の数。

ロードバランサーメトリクスの表示

Lightsail コンソールでロードバランサーのメトリクスを表示するには、次の手順を実行します。

1. Lightsail コンソールにサインインします。
2. Lightsail のホームページで、[ネットワーキング] タブを選択します。
3. メトリクスを表示するロードバランサーの名前を選択します。
4. ロードバランサーの管理ページで [Metrics (メトリクス)] タブを選択します。
5. [Metrics graph (メトリクスグラフ)] 見出しの下のドロップダウンメニューで、表示するメトリクスを選択します。

   グラフには、選択したメトリクスのデータポイントが視覚的に表示されます。
6. メトリクスグラフでは、次のアクションを実行できます。

   - グラフの表示を変更して、1 時間、6 時間、1 日、1 週間、2 週間のデータを表示します。
   - データポイント上にカーソルを置くと、そのデータポイントに関する詳細情報が表示されます。
   - 指定したしきい値をメトリクスが超えたときに通知される、選択したメトリクスのアラームを追加します。詳細については、「アラーム」および「ロードバランサーのメトリクスアラームの作成」を参照してください。

次のステップ

ロードバランサーのメトリクスに対して実行できる追加のタスクがいくつかあります。

- 指定したしきい値をメトリクスが超えたときに通知される、選択したメトリクスのアラームを追加します。詳細については、「アラーム」および「ロードバランサーのメトリクスアラームの作成」を参照してください。
- アラームが作動すると、通知バナーが Lightsail コンソールに表示されます。メールおよび SMS テキストメッセージで通知を受けるには、リソースを監視する各 AWS リージョン で、メールアドレスと携帯電話番号を通知連絡先として追加する必要があります。詳細については、「通知連絡先を追加する」を参照してください。
- 通知の受信を停止するには、Lightsail から メールと携帯電話を削除します。詳細については、「メトリクスアラームを削除または無効化する」を参照してください。また、アラームを無効化または削除して、特定のアラームの通知の受信を停止することもできます。詳細については、「メトリクスアラームを削除または無効化する」を参照してください。

トピック

- Lightsail ロードバランサーのメトリックスアラームを作成する

Lightsail ロードバランサーのメトリックスアラームを作成する

単一のロードバランサーメトリクスを監視する Amazon Lightsail アラームを作成できます。アラームは、指定したしきい値を基準にしたメトリクスの値に基づいて通知するように設定できます。通知は、Lightsail コンソールに表示されるバナー、メールアドレスに送信される メール、携帯電話番号に送信される SMS テキストメッセージです。アラームの詳細については、「アラーム」を参照してください。

目次

ロードバランサーのアラーム制限

アラームには、以下の制限が適用されます。

- メトリクスごとに 2 つのアラームを設定できます。
- アラームは 5 分間隔で評価され、アラームの各データポイントは、集計されたメトリクスデータの 5 分間隔を表します。
- アラームの状態が OK に変わったときに通知するようにアラームを設定できるのは、メールや SMS テキストメッセージで通知するようにアラームを設定した場合だけです。
- OK アラーム通知をテストできるのは、メールや SMS テキストメッセージで通知するようにアラームを設定した場合だけです。
- アラームの状態が `INSUFFICIENT_DATA` に変わったときに通知するようにアラームを設定できるのは、メールや SMS テキストメッセージで通知するようにアラームを設定した場合と、欠落しているデータポイントに対して [欠落しているデータを評価しない] オプションを選択した場合だけです。

- 通知をテストできるのは、アラームが OK 状態の場合だけです。

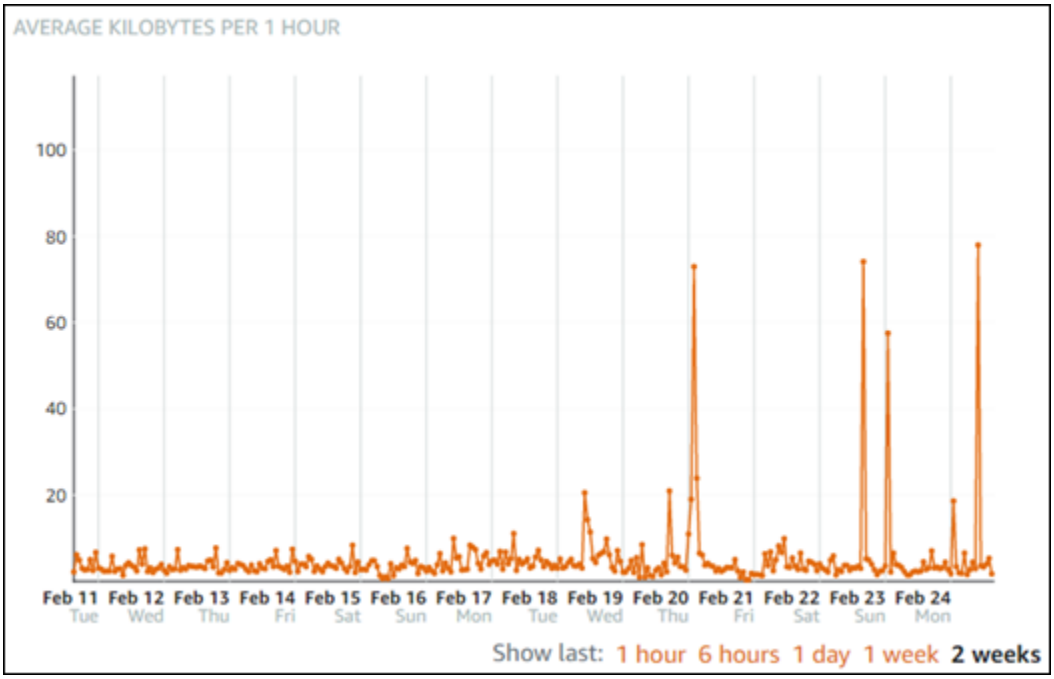
ロードバランサーアラームの設定に関するベストプラクティス

アラームには、以下の制限が適用されます。

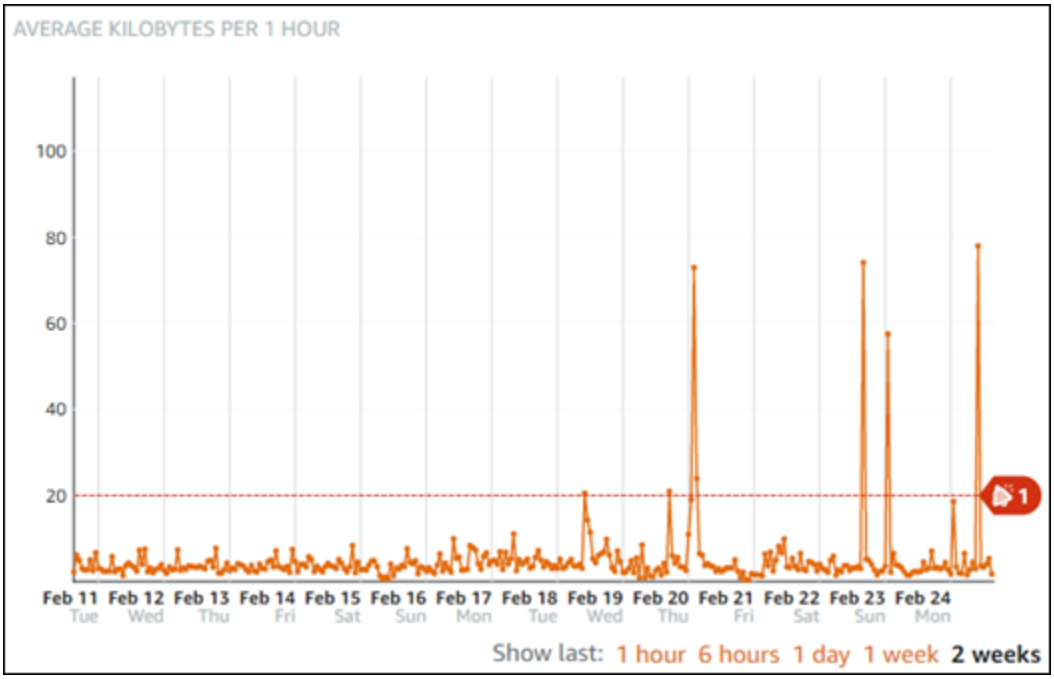
- メトリクスごとに 2 つのアラームを設定できます。
- アラームは 5 分間隔で評価され、アラームの各データポイントは、集計されたメトリクスデータの 5 分間隔を表します。
- アラームの状態が OK に変わったときに通知するようにアラームを設定できるのは、メールや SMS テキストメッセージで通知するようにアラームを設定した場合だけです。
- OK アラーム通知をテストできるのは、メールや SMS テキストメッセージで通知するようにアラームを設定した場合だけです。
- アラームの状態が `INSUFFICIENT_DATA` に変わったときに通知するようにアラームを設定できるのは、メールや SMS テキストメッセージで通知するようにアラームを設定した場合と、欠落しているデータポイントに対して [欠落しているデータを評価しない] オプションを選択した場合だけです。
- 通知をテストできるのは、アラームが OK 状態の場合だけです。

デフォルトのアラーム設定

メトリクスアラームを設定する前に、メトリクスの履歴データを表示する必要があります。過去 2 週間のメトリクスの低レベル、中間レベル、高レベルを識別します。次のインスタンスの発信ネットワークトラフィック (`NetworkOut`) メトリクスグラフの例では、低レベルは 1 時間あたり 0 ~ 10 KB、中間レベルは 1 時間あたり 10 ~ 20 KB、高レベルは 1 時間あたり 20 ~ 80 KB です。



アラームしきい値を低レベル範囲 (1 時間あたり 5 KB など) 以上に設定すると、アラーム通知が頻繁に送信され、不要なアラーム通知が発生する可能性があります。アラームしきい値を高レベルの範囲 (たとえば、1 時間あたり 20 KB) 以上に設定した場合、アラーム通知の頻度は低くなりますが、調査がさらに意義のあるものになる場合があります。アラームを設定して有効にすると、次の例に示すように、しきい値を表すアラームラインがグラフに表示されます。1 というラベルの付いたアラームラインはアラーム 1 のしきい値を表し、2 というラベルのアラームラインはアラーム 2 のしきい値を表します。



Lightsail コンソールを使用してロードバランサーのメトリクスアラームを作成する

Lightsail コンソールを使用してロードバランサーのメトリクスアラームを作成するには、次の手順を実行します。

1. Lightsail コンソールにサインインします。
2. Lightsail のホームページで、[ネットワーキング] タブを選択します。
3. アラームを作成するロードバランサーの名前を選択します。
4. ロードバランサーの管理ページで [Metrics (メトリクス)] タブを選択します。
5. [Metrics Graphs (メトリクスグラフ)] 見出しの下のドロップダウンメニューで、アラームを作成するメトリクスを選択します。詳細については、「リソースのメトリクス」を参照してください。
6. アラームセクションのページでアラームの追加を選択します。
7. ドロップダウンメニューから比較演算子の値を選択します。例の値は、greater than or equal to、greater than、less than、less than or equal to です。
8. アラームのしきい値を入力します。
9. アラームへのデータポイントを入力します。
10. 評価期間を選択します。期間は、5 分から 24 時間まで 5 分単位で指定できます。
11. 次のいずれかの通知方法を選択します。

    - メール — アラームの状態が ALARM に変わると、メールで通知されます。
    - SMS テキストメッセージ — アラームの状態が ALARM に変わると、SMS テキストメッセージによって通知されます。SMS メッセージングは、Lightsail リソースを作成できるすべての AWS リージョンでサポートされているわけではなく、また、一部の国/地域には SMS テキストメッセージを送信できません。詳細については、「SMS テキストメッセージングのサポート」を参照してください。

    > **Note**
    >
    > メールまたは SMS による通知を選択したが、リソースの AWSリージョンで通知の連絡先をまだ設定していない場合は、メールアドレスまたは携帯電話番号を追加する必要があります。詳細については、「通知」を参照してください。

12. (オプション) [Send me a notification when the alarm state change to OK (アラームの状態が OK に変わったときに通知を送信する)] を選択して、アラームの状態が OK に変わったときに通知を

受け取ります。このオプションは、メールまたは SMS テキストメッセージで通知されるように選択した場合にのみ使用できます。

13. (オプション) [Advanced settings (詳細設定)] を選択し、次のいずれかのオプションを選択します。

    - アラームによる欠落データの処理方法を選択します。次のオプションを使用できます。
      - しきい値の範囲内にないと仮定する(しきい値を超過) — 欠落しているデータポイントは「不良」として扱われ、しきい値を超えています。
      - しきい値内にあると仮定する (しきい値を超過していない) — 欠落しているデータポイントは、「良い」と見なされ、しきい値の範囲内で処理されます。
      - 最後の正常なデータポイントの値を使用する (無視して現在のアラーム状態を維持する) — 現在のアラーム状態が維持されます。
      - 評価しない (欠落しているデータを欠落しているデータとして扱う) — 状態を変更するかどうかを評価する際に、欠落データポイントを考慮に入れません。
    - [Send a notification if there is insufficient data (データが不足している場合に通知を送信)] を選択して、アラームの状態が INSUFFICIENT_DATA に変わったときに通知を受け取ります。このオプションは、メールまたは SMS テキストメッセージで通知されるように選択した場合にのみ使用できます。

14. [Create (作成)] を選択してアラームを追加します。

    後でアラームを編集するには、編集するアラームの横にある省略記号アイコン (:) を選択し、[アラームの編集] を選択します。

Lightsail コンソールを使用してロードバランサーのメトリクスアラームをテストする

Lightsail コンソールを使用してアラームをテストするには、次の手順を実行します。アラームをテストして、アラームがトリガーされたときに メールや SMS テキストメッセージを受信するなど、設定された通知オプションが機能していることを確認します。

1. Lightsail コンソールにサインインします。
2. Lightsail のホームページで、[ネットワーキング] タブを選択します。
3. アラームをテストするロードバランサーの名前を選択します。
4. ロードバランサーの管理ページで [Metrics (メトリクス)] タブを選択します。
5. [Metrics Graphs (メトリクスグラフ)] 見出しの下のドロップダウンメニューで、アラームをテストするメトリクスを選択します。

6. ページの [アラーム] セクションまでスクロールし、テストするアラームの横にある省略記号アイコン (⋮) を選択します。
7. 以下のオプションのいずれかを選択します。

   - [アラーム通知のテスト] - このオプションを選択すると、アラームの状態が ALARM に変わったときの通知をテストできます。
   - [OK 通知のテスト] — このオプションを選択すると、アラームの状態が OK に変わったときの通知をテストできます。

   > **Note**
   >
   > これらのオプションのいずれかが使用できない場合は、アラームの通知オプションが設定されていないか、アラームが現在 ALARM 状態になっている可能性があります。詳細については、「ロードバランサーのアラーム制限」を参照してください。

   選択したテストオプションに応じて、アラームが一時的に ALARM または OK 状態に変化し、アラームの通知方法として設定した内容に応じて、メールまたは SMS テキストメッセージが送信されます。通知バナーは、ALARM 通知のテストを選択した場合にのみ Lightsail コンソールに表示されます。OK 通知のテストを選択した場合、通知バナーは表示されません。アラームは、通常、数秒後に実際の状態に戻ります。

ロードバランサーアラームの作成後の次のステップ

ロードバランサーのアラームに対して実行できる追加のタスクがいくつかあります。

- 通知の受信を停止するには、Lightsail から メールと携帯電話を削除します。詳細については、「通知連絡先の削除」を参照してください。また、アラームを無効化または削除して、特定のアラームの通知の受信を停止することもできます。詳細については、「メトリクスアラームを削除または無効化する」を参照してください。

Lightsail で通知連絡先を追加する

インスタンス、データベース、ロードバランサーないしコンテンツ配信ネットワーク (CDN) ディストリビューションのメトリクスが指定した値を超えた場合に通知を受けられるよう、Amazon Lightsail を設定することが可能す。通知は、Lightsail コンソールに表示されるバナー、指定したアド

レスに送信される メール、指定した携帯電話番号に送信される SMS テキストメッセージの形式にすることができます。メールおよび SMS テキストメッセージで通知を受けるには、リソースを監視する各 AWS リージョン で、メールアドレスと携帯電話番号を通知連絡先として追加する必要があります。通知の詳細については、「通知」を参照してください。

> **Important**
>
> SMS テキストメッセージ機能は一時的に無効になっており、現在、Lightsail リソースを作成できる AWS リージョン ではサポートされていません。詳細については、「SMS テキストメッセージングのサポート」を参照してください。

目次

リージョンの通知の連絡先の制限

各 AWS リージョン に追加できるメールアドレスと携帯電話番号は 1 つだけです。すでにメールアドレスや携帯電話番号追加されているリージョンにメールアドレスまたは携帯電話番号を追加すると、既存の通知連絡先を新しい連絡先に置き換えるかどうか尋ねられます。

AWS リージョン に複数のメール受信者が必要な場合は、複数の受信者に転送する配布リストを構成し、配布リストのメールアドレスを通知連絡先として追加できます。

SMS テキストメッセージングのサポート

> **Important**
>
> SMS テキストメッセージ機能は一時的に無効になっており、現在、Lightsail リソースを作成できる AWS リージョン ではサポートされていません。または、メールのメッセージを設定したり、Lightsail コンソールに表示される通知バナーを利用したりすることも可能です。

SMS テキストメッセージサポートに関する次の情報は、この機能を無効にする前に SMS テキストメッセージを設定したお客様向けに公開されています。

SMS テキストメッセージングは、Lightsail リソースを作成できるすべての AWS リージョン でサポートされているわけではありません。また、SMSテキストメッセージは、世界の一部の国や地域に送信することはできません。SMS メッセージングがサポートされていない AWS リージョン では、メール通知連絡先のみを設定できます。

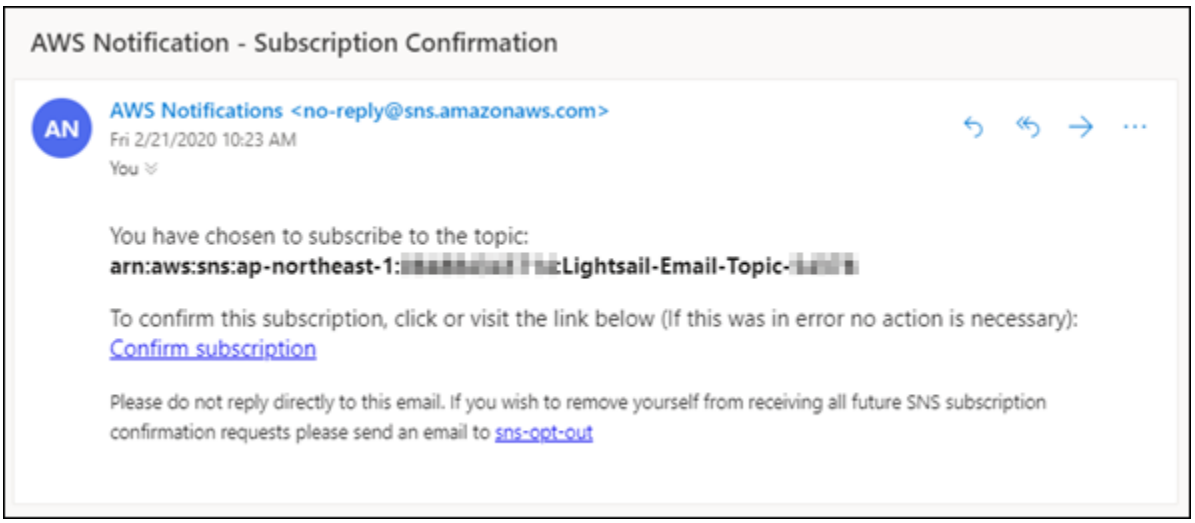
SMS メッセージングは、以下の AWS リージョン でサポートされています。これらは Amazon Simple Notification Service (Amazon SNS)が SMS テキストメッセージングをサポートしているリージョンで、このリージョンは、Lightsail が通知を送信するために使用されます。

- 米国東部 (バージニア北部) (us-east-1)
- 米国西部 (オレゴン) (us-west-2)
- アジアパシフィック (シンガポール) (ap-southeast-1)
- アジアパシフィック (シドニー) (ap-southeast-2)
- アジアパシフィック (東京) (ap-northeast-1)
- 欧州 (アイルランド) (eu-west-1)

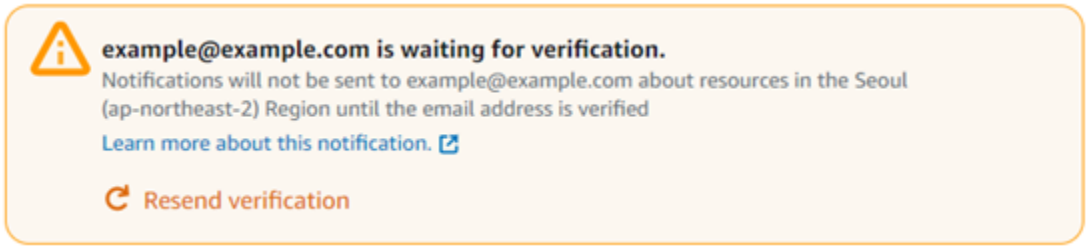
SMS テキストメッセージを送信できる国と地域、および SMS テキストメッセージングがサポートされている AWS リージョン と国の最新のリストは、Amazon SNS デベロッパーガイドの「サポートされているリージョンと国」を参照ください。

メールによる連絡先の確認

Lightsail でメールアドレスを通知連絡先として追加すると、そのアドレスに確認要求が送信されます。確認要求の メールには、受信者が Lightsail 通知を受信することを確認するためにクリックする必要があるリンクが含まれています。通知は、確認が完了するまでメールアドレスに送信されません。検証は、AWS 通知 <no-reply@sns.amazonaws.com> から行われ、件名は AWS 通知 - サブスクリプションの確認です。SMS メッセージングは検証を必要としません。



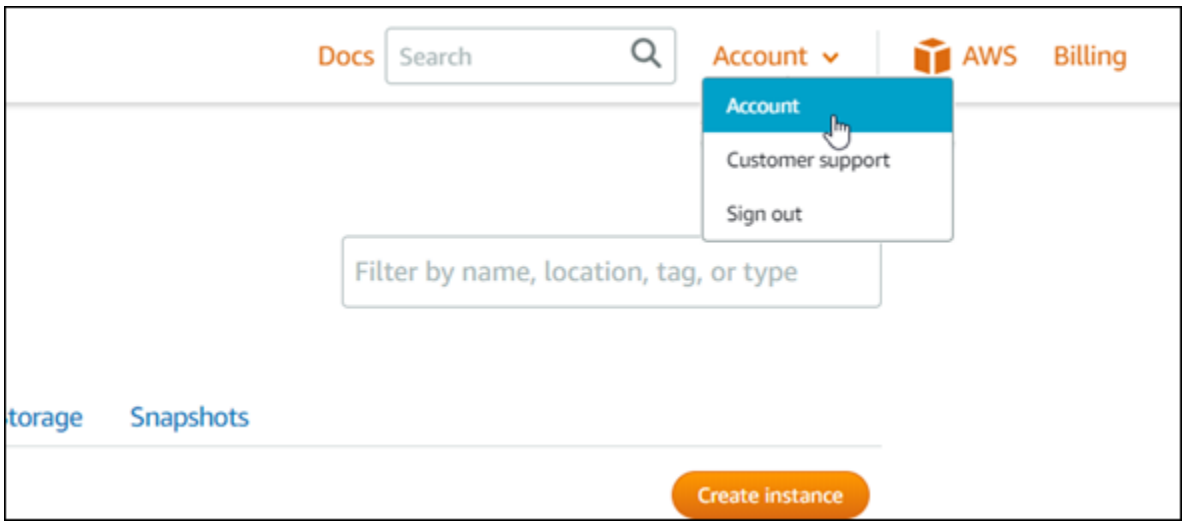
確認要求が受信トレイフォルダにない場合は、メールボックスのスパムフォルダと迷惑メールフォルダを確認してください。検証リクエストが失われた、または削除された場合は、Lightsail コンソールやアカウントページに表示される通知バナーで検証の再送を選択してください。



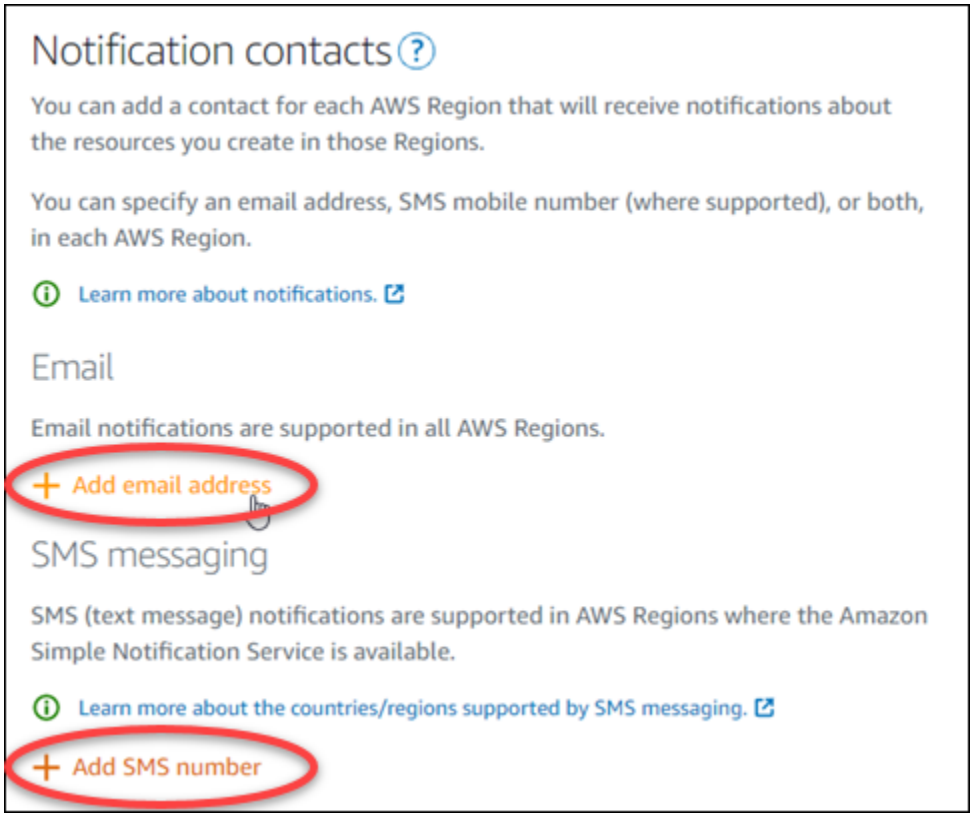
Lightsail コンソールを使用した通知連絡先の追加

Lightsail コンソールを使用して通知連絡先を追加するには、次の手順を完了します。

1. Lightsail コンソールにサインインします。
2. Lightsail ホームページの上部にあるナビゲーションメニューで [Account (アカウント)] を選択します。
3. ドロップダウンメニューで [Account (アカウント)] を選択します。

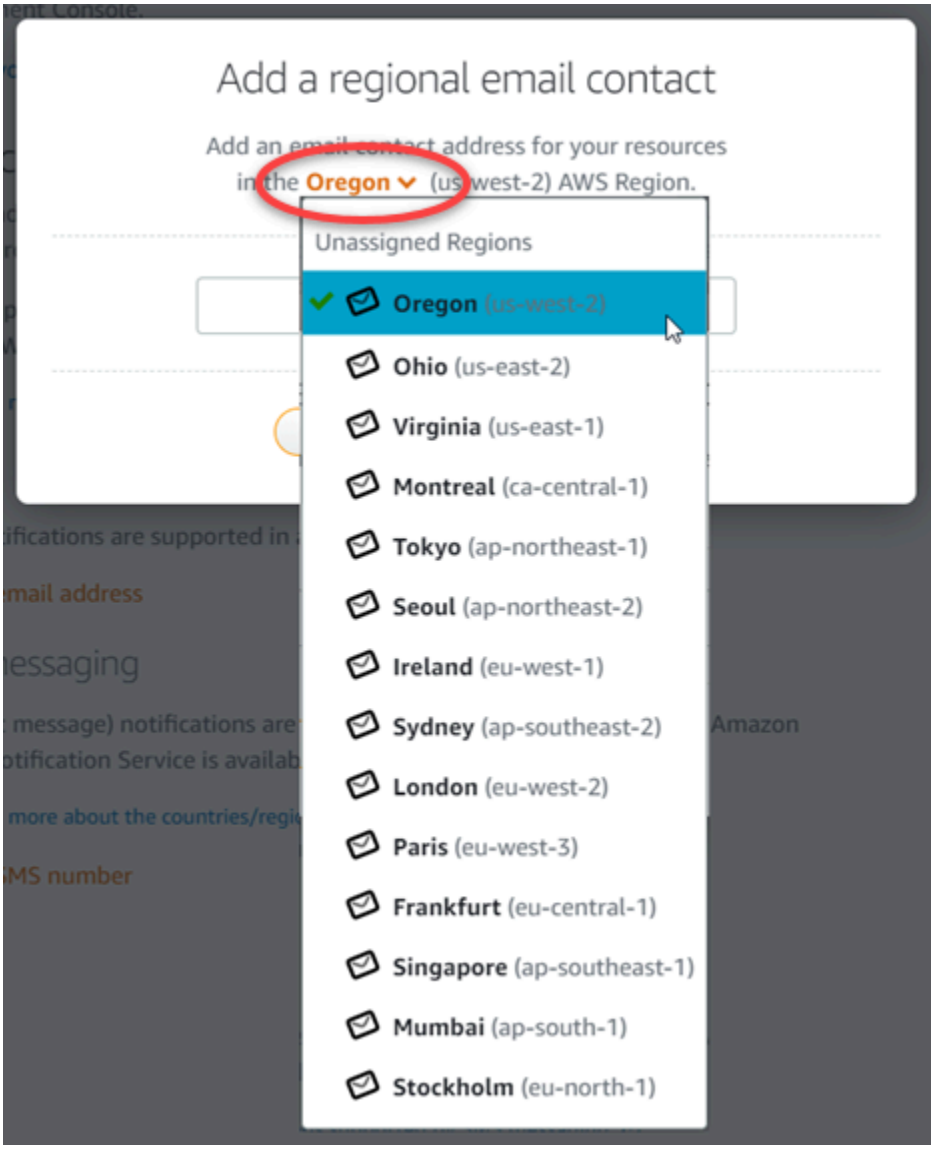


4. 通知連絡先のプロフィールと連絡先タブで、メールアドレスの追加を選択、または SMS 番号を追加するを選択します。



5. 次のいずれかのステップを完了します。

   - メールアドレスを追加する場合は、通知連絡先を追加する AWS リージョン を選択します。テキストボックスにメールアドレスを入力します。

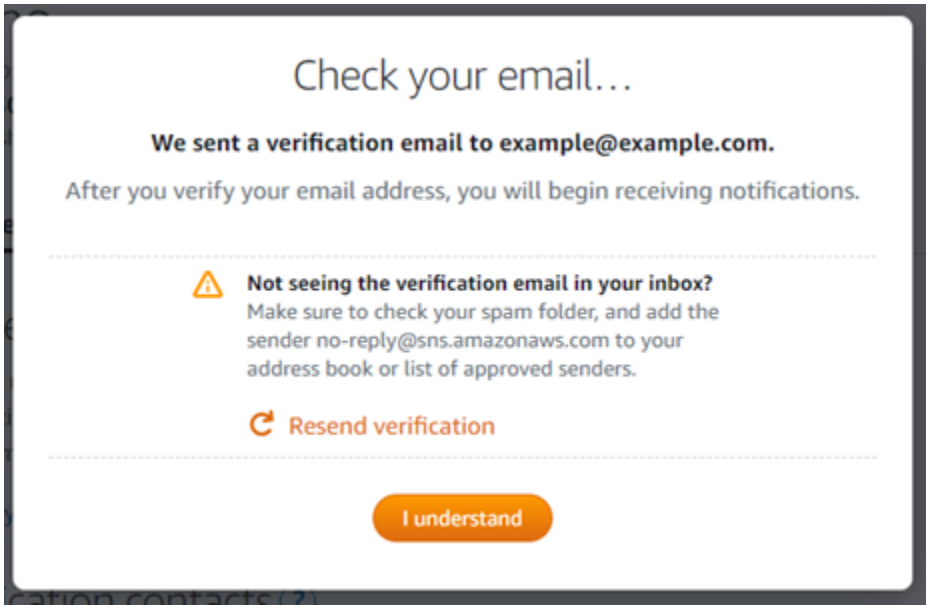


- SMS 番号を追加する場合は、通知連絡先を追加する AWS リージョン を選択します。携帯電話番号の国を選択し、テキストボックスに入力します。国コードは既に入力されています。

> **Important**
>
> SMS テキストメッセージ機能は一時的に無効になっており、現在、Lightsail リソースを作成できる AWS リージョン ではサポートされていません。詳細については、「SMS テキストメッセージングのサポート」を参照してください。

6. [Add Contact (連絡先を追加)] を選択します。

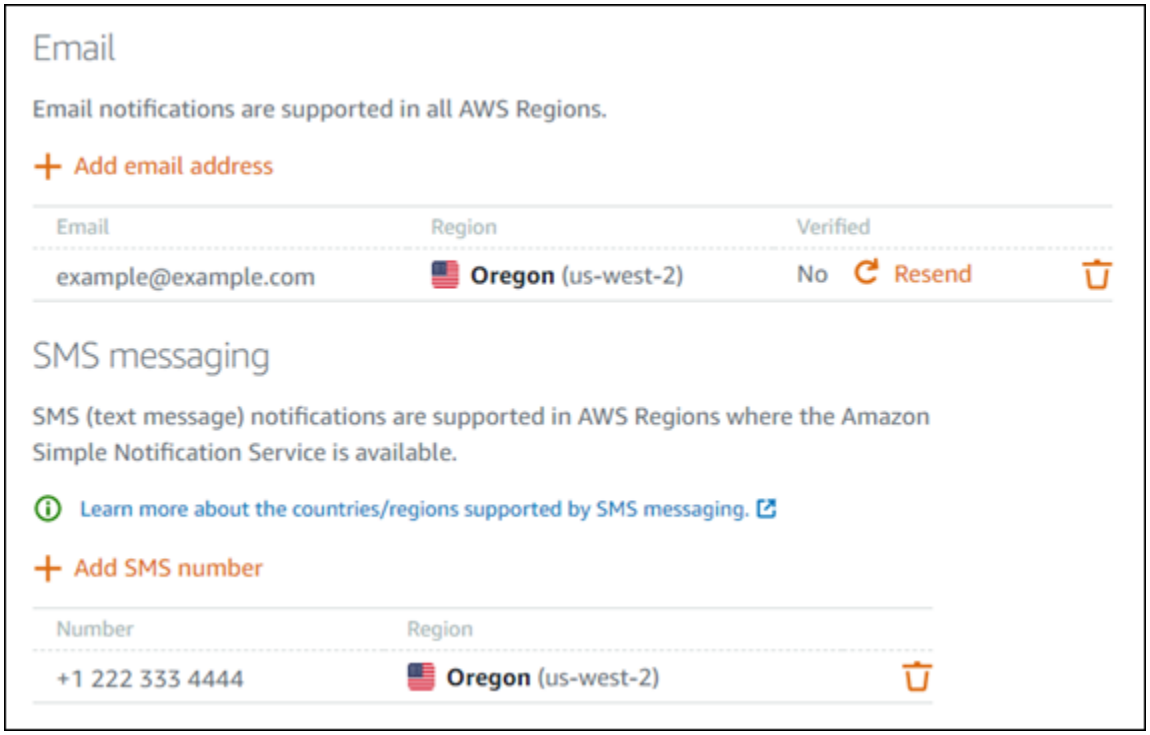
   メールアドレスを通知連絡先として追加すると、そのアドレスに確認要求が送信されます。確認要求の メールには、受信者が Lightsail 通知を受信することを確認するためにクリックする必要があるリンクが含まれています。SMS メッセージングは検証を必要としません。



7. [I understand (理解する)] を選択します。

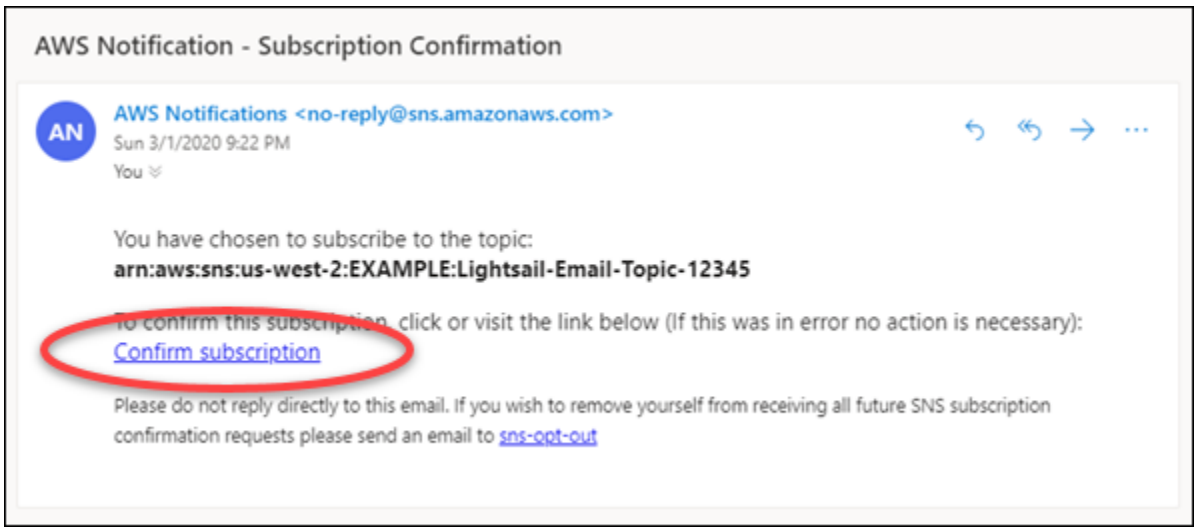
メールアドレスまたは携帯電話番号が [通知連絡先] セクションに追加されます。次の手順で確認プロセスを完了するまで、メールアドレスは確認されません。確認が完了するまで、通知はメールアドレスに送信されません。認証リクエストが紛失したか、削除された場合は、リージョンのメールアドレスの横にある [Resend (再送信)] を選択して、別の認証リクエストを送信します。

> **Note**
>
> SMS メッセージングは検証を必要としません。したがって、SMS 通知の連絡先を追加した後、この手順の手順 8 ~ 10 を完了する必要はありません。



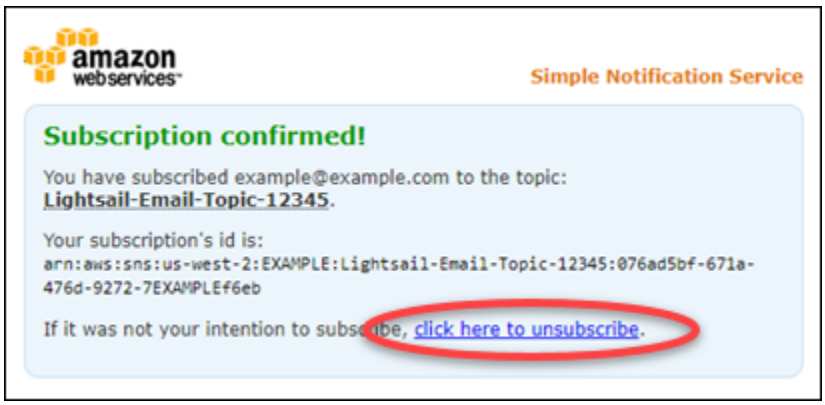
8. Lightsail で通知連絡先として追加したメールアドレスの受信トレイを開きます。
9. no-reply@sns.amazonaws.com からの AWS 通知 - サブスクリプションの確認 メールを開きます。

> **Note**
>
> 確認要求が受信トレイフォルダにない場合は、メールボックスのスパムフォルダと迷惑メールフォルダを確認してください。



10. メールで [Confirm subscription (サブスクリプションを確認)] を選択して、Lightsail 通知を受信することを確認します。

    ブラウザウィンドウが開き、サブスクリプションを確認する次のページが表示されます。登録を解除するには、ページの [click here to unsubscribe (ここをクリックしてページから登録を解除します)] を選択します。または、ページを閉じた場合は、通知連絡先を削除する手順を実行します。



AWS CLI を使用した通知連絡先の追加

Lightsail に AWS Command Line Interface (AWS CLI) を使用するための通知連絡先を追加するには、次の手順を完了します。

1. ターミナルまたはコマンドプロントウィンドウを開きます。

まだインストールしていない場合は、AWS CLI をインストールし、Lightsail と連携するように設定します。

2. 次のコマンドを入力して、通知連絡先を追加します。

```
aws lightsail create-contact-method --region Region --notificationProtocol Protocol
 --contact-endpoint Destination
```

コマンドを、以下のように置き換えます。

- *#####*は、通知連絡先を追加する AWS リージョン にします。
- 連絡先の通知プロトコルを使用する*#####*。メール、または SMS にする必要があります。
- メールアドレスまたは携帯電話番号の*##* 。

> **Note**
>
> 携帯電話番号を指定する場合は、E.164 形式を使用します。E.164 は、国際的な音声通信に使用される電話番号の構造の規格です。この形式に従う電話番号には最大 15 桁を設定でき、プラス記号 (+) および国コードのプレフィックスがついています。たとえば、E.164 形式の米国の電話番号は +1XXX5550100 として表示されます。詳細については、Wikipedia の E.164 を参照してください。

例:

```
aws lightsail create-contact-method --region us-west-2 --notificationProtocol Email
 --contact-endpoint example@example.com
```

```
aws lightsail create-contact-method --region us-east-1 --notificationProtocol SMS
 --contact-endpoint +14445556666
```

Enter キーを押すと、オペレーションの応答にリクエストの詳細が表示されます。

通知の連絡先として指定したメールアドレスに確認リクエストが送信されます。これにより、受信者が Lightsail 通知のサブスクリプションを希望していることが確認されます。メールアドレスは、以下の手順で確認処理が完了するまで確認されません。メールアドレスが確認されるま

で、通知はメールアドレスに送信されません。元の通知が間違っている場合は、リージョンのメールアドレスの横にある [Resend (再送信)] を選択して、別の確認リクエストを送信します。

> **Note**
>
> SMS メッセージングは検証を必要としません。したがって、SMS 通知の連絡先を追加するときに、この手順の手順 8 ~ 10 を完了する必要はありません。

3. 通知連絡先として追加したメールアドレスの受信トレイを開きます。
4. no-reply@sns.amazonaws.com からの AWS 通知 - サブスクリプションの確認 メールを開きます。
5. メールで [Confirm subscription (サブスクリプションの確認)] を選択して、Lightsail からの メール通知を受信することを確認します。

   ブラウザウィンドウが開き、サブスクリプションを確認する次のページが表示されます。登録を解除するには、ページの [click here to unsubscribe (ここをクリックしてページから登録を解除します)] を選択します。または、ページを閉じた場合は、通知連絡先を削除する手順を実行します。

通知連絡先を追加した後の次の手順

通知連絡先に対して実行できる追加のタスクがいくつかあります。

- 通知連絡先を追加した AWS リージョン にアラームを追加します。アラームの開始時に、メールおよび SMS テキストメッセージで通知されるように選択できます。詳細については、「アラーム」を参照してください。
- 通知が予定されているときに通知を受け取らない場合は、通知の連絡先が正しく設定されていることを確認するためにいくつかの点を確認してください。詳細については、「通知のトラブルシューティング」を参照してください。
- 通知の受信を停止するには、Lightsail から メールと携帯電話を削除します。詳細については、「メトリクスアラームを削除または無効化する」を参照してください。また、アラームを無効化または削除して、特定のアラームの通知の受信を停止することもできます。詳細については、「メトリクスアラームを削除または無効化する」を参照してください。

Lightsail の通知連絡先を削除する

メールおよび携帯電話番号の通知連絡先を Amazon Lightsail から削除して、Lightsail リソースのメールおよび SMS テキストメッセージ通知の受信を停止します。通知の詳細については、「通知」を参照してください。

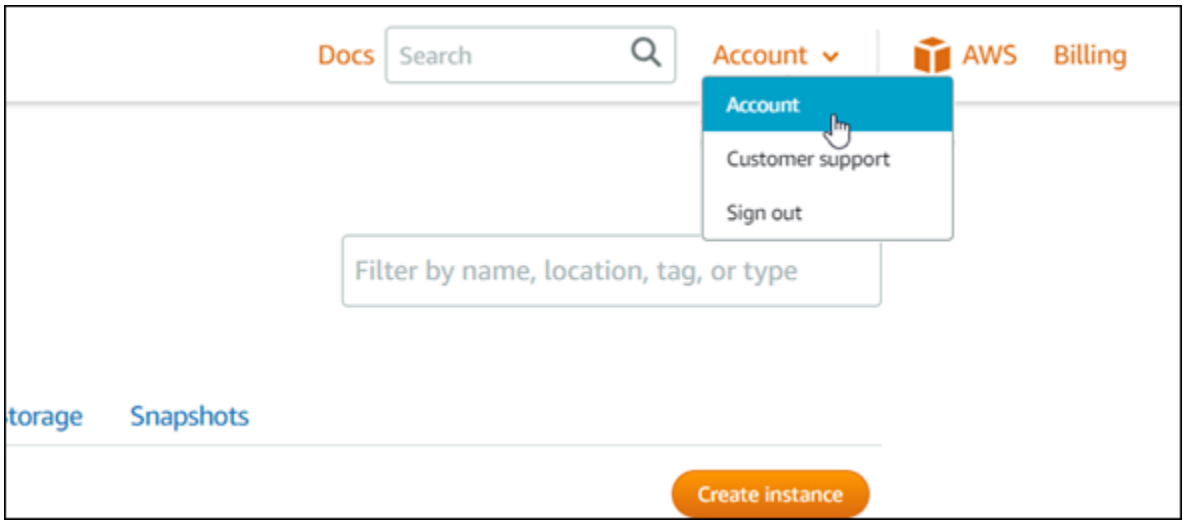
また、アラームを無効にするか、削除して、特定のアラームの通知の受信を停止することもできます。詳細については、「メトリクスアラームを削除または無効化する」を参照してください。

目次

Lightsail コンソールを使用した通知連絡先の削除

Lightsail コンソールを使用して通知連絡先を削除するには、次の手順を実行します。

1. Lightsail コンソールにサインインします。
2. Lightsail ホームページの上部にあるナビゲーションメニューで [Account (アカウント)] を選択します。
3. ドロップダウンメニューで [Account (アカウント)] を選択します。



4. 削除するメールアドレスまたは携帯電話番号の横にある削除アイコンを [Profile & contacts] (プロファイルと連絡先) タブの [Notification contacts] (通知連絡先) セクションで選択します。

5. [Yes (はい)] を選択して、通知連絡先を削除することを確認します。

AWS CLI を使用した通知連絡先の削除

AWS Command Line Interface (AWS CLI) を使用して Lightsail の通知連絡先を削除するには、次の手順を実行します。

1. ターミナルまたはコマンドプロントウィンドウを開きます。

   まだインストールしていない場合は、AWS CLI をインストールし、Lightsail と連携するように設定します。

2. 通知連絡先を削除するには、次のコマンドを入力します。

   ```
   aws lightsail delete-contact-method --region Region --notificationProtocol Protocol
   ```

   コマンドを、以下のように置き換えます。

   - *Region* を通知連絡先を削除する AWS リージョン に置き換えます。
   - メールや SMS など、削除する連絡先の通知プロトコルを使用する*#####*。

   例:

   ```
   aws lightsail delete-contact-method --region us-west-2 --notificationProtocol SMS
   ```

   Enter キーを押すと、オペレーションの応答にリクエストの詳細が表示されます。

通知連絡先を削除した後の次の手順

通知連絡先の削除後に実行できる追加のタスクがいくつかあります。

- 通知連絡先を削除すると、メールおよび SMS テキストメッセージングの通知は停止しますが、通知バナーが Lightsail コンソールに表示されるのを停止することはできません。通知バナーを停止し、メールおよび SMS テキストメッセージング通知も停止するには、バナーの原因となっているアラームを無効にするか削除します。詳細については、「メトリクスアラームを削除または無効化する」を参照してください。

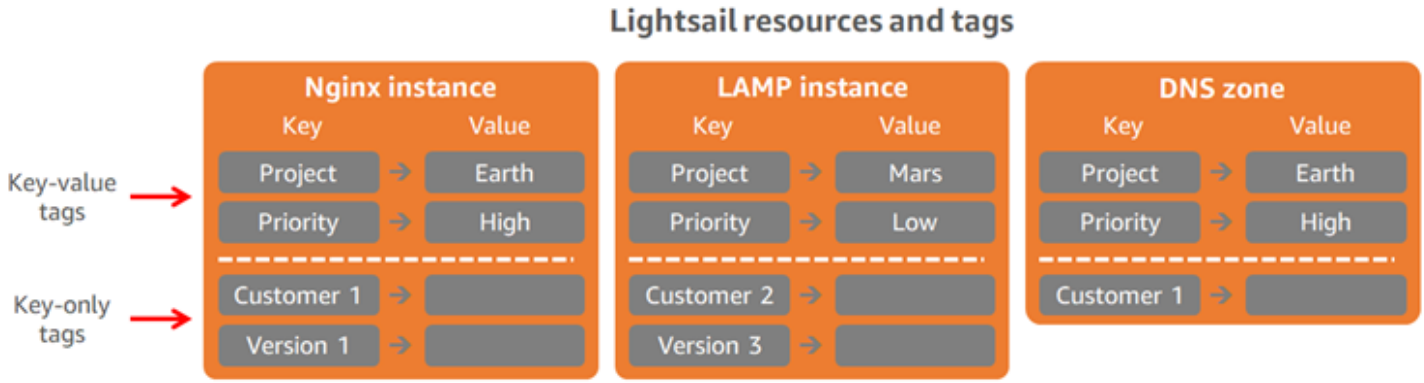
- メールおよび SMS テキストメッセージング通知の受信を再開するには、通知連絡先 Lightsail としてメールアドレスと携帯電話番号を追加します。詳細については、「通知連絡先を追加する」を参照してください。

Amazon Lightsail のタグ

Amazon Lightsail は、ラベルをタグとしてリソースに割り当てることができます。各タグは、キーおよび値 (省略可能) で構成されるラベルです。タグを使うと、リソースの管理、検出、およびフィルタ処理が容易になります。

Amazon Lightsail は、ラベルをタグとしてリソースに割り当てることができます。各タグは、キーと値 (省略可能)で構成されるラベルです。タグを使うと、リソースの管理、検出、およびフィルタ処理が容易になります。タグには、固有なタイプはありませんが、Lightsail リソースを用途、所有者、環境などの基準で分類できます。これは、同じ種類のリソースが多い場合に役立ちます。リソースに割り当てたタグに基づいて、特定のリソースをすばやく識別できます。たとえば、各リソースのプロジェクトや優先度の追跡に役立つ一連のタグを定義できます。

値のないキーは、Lightsail でキーのみのタグと呼ばれ ます。値のあるキーは、キーと値のタグと呼ばれ ます。次の図は、タグの機能を示しています。この例では、各リソースに複数の「キーと値のタグ」があり、1 つ以上の「キーのみのタグ」があります。キーと値のタグはプロジェクトと優先度を識別し、キーのみのタグは顧客とアプリケーションバージョンを識別します。



タグを使用して請求を整理し、アクセスをコントロールする

タグを使用して、請求の整理、Lightsail のリソースとリクエストへのアクセスコントロール、およびタグキーへのアクセスコントロールを行うこともできます。詳細については、以下のいずれかのガイドを参照してください。

- タグを使用してリソースのコストを整理する
- タグを使用してリソースアクセスを制御する

タグ付けをサポートする Lightsail のリソース

ほとんどの Lightsail リソースには、作成時または作成後にタグを付けることができます。リソースの作成時にタグを適用できない場合、Lightsail はリソースの作成プロセスをロールバックします。これにより、リソースはタグ付きで作成されるか、まったく作成されないことになり、タグ付けを要するリソースにタグが付いていない状態はなくなります。

以下の Lightsail リソースには、Lightsail コンソールでタグを付けることができます。

- インスタンス
- コンテナサービス
- コンテンツ配信ネットワーク (CDN) の配信
- バケット
- データベース
- Disks
- DNS ゾーン
- ロードバランサー

> **Important**
>
> Lightsail コンソールを使用して作成されたスナップショットは、ソースリソースからタグを自動的に継承します。スナップショットから作成した Lightsail リソースには、スナップショットの作成時にソースリソースと同じタグが適用されます。

以下のリソースは、Lightsail API、AWS Command Line Interface (AWS CLI)、または SDK を使用してタグ付けできます。

- データベーススナップショット
- データベース
- ディスクスナップショット
- Disks
- ドメイン (DNS ゾーン)
- インスタンススナップショット

- インスタンス
- キーペア
- ロードバランサーの TLS 証明書 (Lightsail で作成された TLS 証明書)
- ロードバランサー

> **Important**
>
> Lightsail API、AWS CLI、または SDK を使用して作成されたスナップショットは、ソースリソースからタグを自動的に継承しません。代わりに、`tags` パラメータを使用してソースリソースのタグを手動で指定する必要があります。

タグの制限

タグには以下のベーシックな制限があります。

- リソースあたりのタグの最大数 – 50。
- リソースごとに各タグキーを一意にする必要があります。各タグキーが保持できる値は 1 つのみです。
- キーの最大長 – 128 Unicode 文字 (UTF-8)
- 値の最大長 - 256 Unicode 文字 (UTF-8)。
- 複数のサービス間およびリソース間でタグ付けスキーマを使用する場合、他のサービスでも許可される文字に制限が適用されることがあるのでご注意ください。通常使用できる文字は、文字、数字、スペース、および特殊文字 +、-、= . _ : / @ です。
- タグのキーと値は大文字と小文字が区別されます。
- キーや値には `aws:` プレフィックスは使用しないでください。このプレフィックスは AWS 専用として予約されています。

Lightsail リソースタグを追加する

Amazon Lightsail のタグを使用し、リソースを目的、所有者、環境などの基準別に分類します。タグは、リソースの作成時または作成後に追加できます。作成後のリソースにタグを追加するには、以下の手順を実行します。

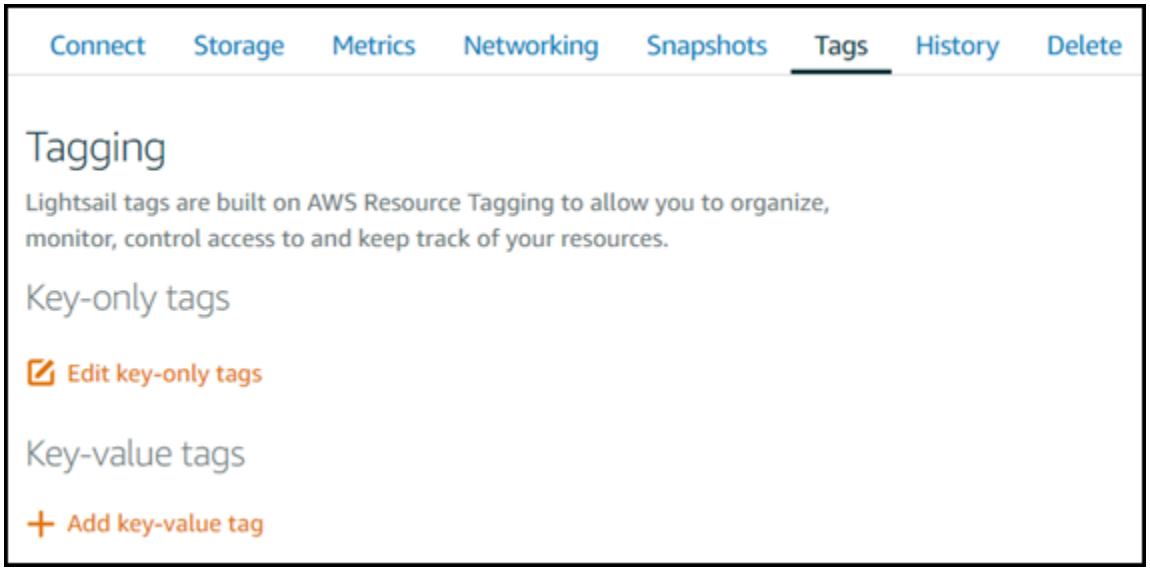
> **Note**
>
> タグ、タグを追加できるリソース、および制限の詳細については、「タグ」を参照してください。

リソースにタグを追加するには

1. Lightsail コンソールにサインインします。
2. Lightsail のホームページで、タグを追加するリソースタイプのタブを選択します。たとえば、DNS ゾーンにタグを追加するには、[ネットワーキング] タブを選択します。インスタンスにタグを追加するには、[インスタンス] タブを選択します。

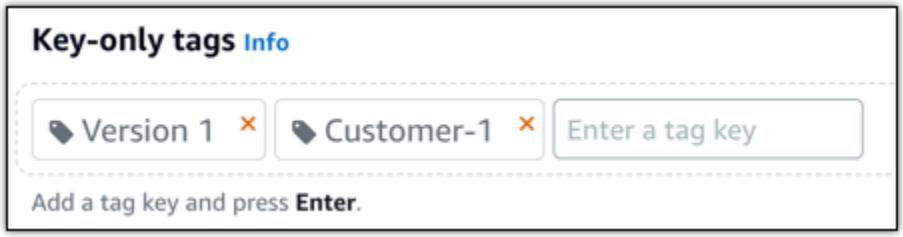
   > **Note**
   >
   > Lightsail コンソールを使用して、インスタンス、コンテナサービス、 CDN ディストリビューションバケット、データベース、ディスク、DNS ゾーン、およびロードバランサーをタグ付けすることができます。ただし、Lightsail API オペレーション、AWS Command Line Interface (AWS CLI)、または SDK を使用すると、より多くの Lightsail リソースにタグを付けることができます。タグ付けをサポートする Lightsail リソースの詳細なリストについては、「タグ」を参照してください。

3. タグを追加するリソースを選択します。
4. 選択したリソースの管理ページで、[タグ] タブを選択します。



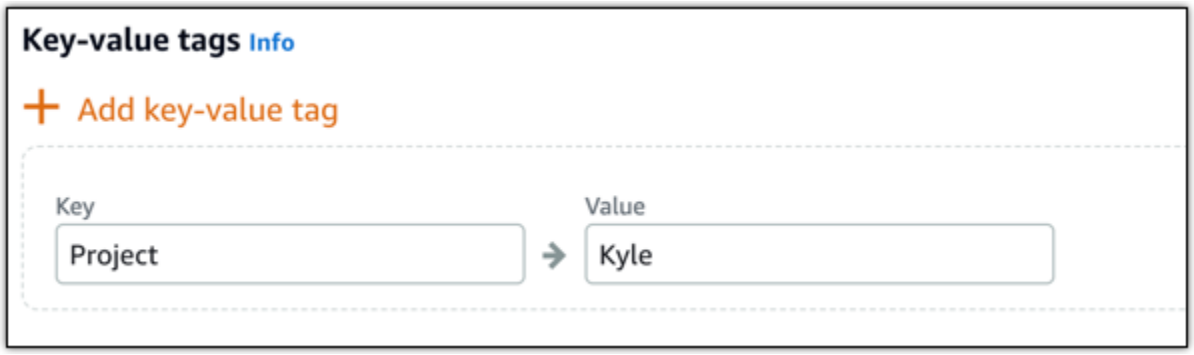
5. 追加するタグのタイプに応じて、以下のいずれかのオプションを選択します。

- [key-only タグの追加] または [key-only タグの編集] (タグが追加済みの場合)。タグキーのテキストボックスに新しいタグを入力し、Enter キーを押します。タグの入力を完了したら、[保存] を選択してタグを追加し、追加しない場合は、[キャンセル] を選択します。



- [key-value タグの作成] から [キー] テキストボックスにキーを入力し、[値] テキストボックスに値を入力します。タグの入力を完了したら、[保存] を選択し、追加しない場合は、[キャンセル] を選択します。

  キーバリューのタグは、保存する際に一つずつ追加することができます。さらに key-value タグを追加するには、以上のステップを繰り返します。



次のステップ

リソースにタグを追加した後で実行できるタスクの詳細については、以下のガイドを参照してください。

- タグを使用して、リソースを整理する
- タグを使用してリソースのコストを整理する
- タグを使用してリソースへのアクセスを制御する
- タグの削除

Lightsail でタグを削除する

Amazon Lightsail リソースからタグを削除できます。1 つのリソースからタグを削除しても、他のすべてのリソースから同じタグが削除されるわけではありません。すべてのリソースからタグを完全に削除するには、そのタグを各リソースから削除する必要があります。このガイドでは、リソースからタグを削除する手順を示します。

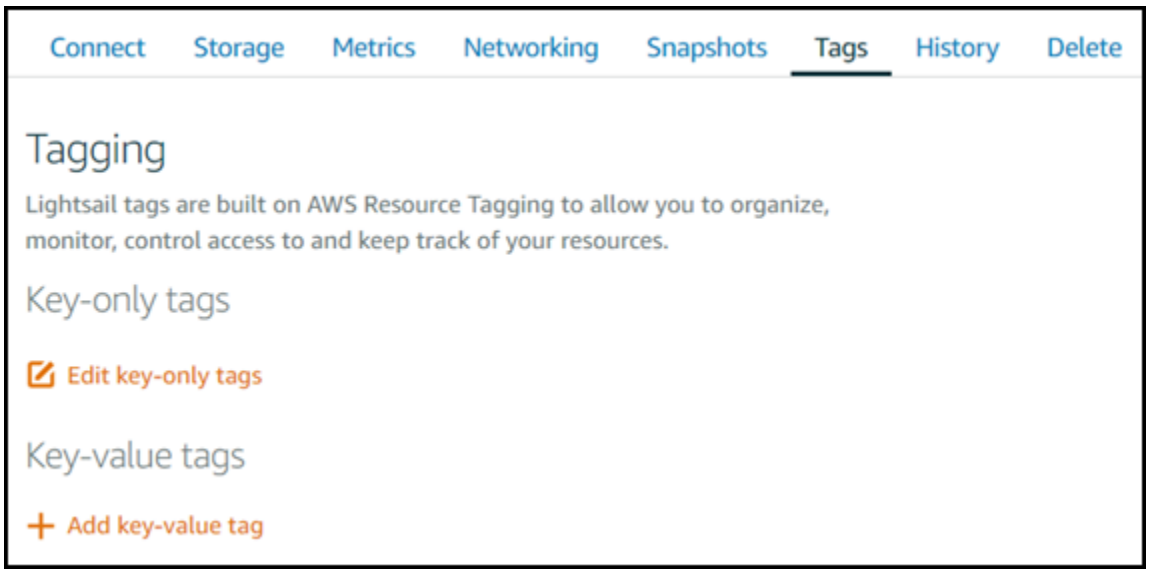
> **Note**
>
> タグ、タグを追加できるリソース、およびタグの制限の詳細については、「タグ」を参照してください。

リソースからタグを削除するには

1. Lightsail コンソールにサインインします。
2. Lightsail のホームページで、タグを削除するリソースタイプのタブを選択します。たとえば、DNS ゾーンからタグを削除するには、[ネットワーキング] タブを選択します。インスタンスからタグを削除するには、[インスタンス] タブを選択します。

   > **Note**
   >
   > Lightsail コンソールでは、インスタンス、コンテナサービス、CD N ディストリビューション、バケット、データベース、ディスク、DNS ゾーン、およびロードバランサーにタグを追加できます。Lightsail API オペレーション、または AWS Command Line Interface (AWS CLI) もしくは SDKs を使用すると、より多くの Lightsail リソースのタグ付けが可能です。タグ付けをサポートする Lightsail リソースの詳細なリストについては、「タグ」を参照してください。

3. タグを削除するリソースを選択します。
4. 選択したリソースの管理ページで、[タグ] タブを選択します。



5. リソースから削除するタグのタイプに応じて、以下のいずれかを選択します。

   a. [Edit key-only tags (キーのみのタグを編集)] を選択し、リソースから削除するタグの削除アイコン (X) を選択します。タグをリソースから削除することを確定する場合は、[保存] を選択します。タグを削除しない場合は、[キャンセル] を選択します。

   b. キーと値のタグを削除するには、キーと値のタグの削除アイコン (X) を選択します。プロンプトで、キーと値のタグを削除する場合は [はい、削除します] を選択します。削除しない場合は [いいえ、キャンセルします] を選択します。

リソースレベルのアクセス許可および Lightsail タグに基づく承認のサポート

Lightsail は、リソースレベルのアクセス許可および一部の API アクションのタグに基づく承認をサポートします。詳細については、「サービス承認リファレンス」の「Amazon Lightsail のアクション、リソース、および条件キー」を参照してください。

タグを使用して Lightsail リソースのアクセスを制御する

Amazon Lightsail のタグを使用して、リソース、リクエスト、およびタグキーへのアクセスをコントロールできます。このガイドでは、Lightsail リソースの作成や削除に必要な key-value タグを指定する AWS Identity and Access Management (IAM) ポリシーを作成し、また、これらのリクエストを行うユーザーやグループにポリシーをアタッチする方法について説明します。

> **Note**
>
> Lightsail のタグの詳細については、「タグ」を参照してください。

ステップ 1: IAM ポリシーを作成する

まず、IAM コンソールで以下の IAM ポリシーを作成します。IAM ポリシーの作成の詳細については、IAM ドキュメントの「IAM ポリシーの作成」を参照してください。

次に続くポリシーは、作成リクエストに `allow` のキータグと、 `true` の値が定義されていない限り、ユーザーによる新しい Lightsail リソースの作成も禁止します。このポリシーは、 `allow/true` のキーバリューのタグが定義されていない限り、ユーザーによるリソースの削除も禁止します。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "lightsail:Create*",
                "lightsail:TagResource",
                "lightsail:UntagResource"
            ],
```

```
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "aws:RequestTag/allow": "true"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "lightsail:Delete*",
                "lightsail:TagResource",
                "lightsail:UntagResource"
            ],
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "aws:ResourceTag/allow": "true"
                }
            }
        }
    ]
}
```

次に続くポリシーでは、キーバリューのタグが allow/false ではないリソースのタグの変更をユーザーに禁止します。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Deny",
            "Action": [
                "lightsail:TagResource"
            ],
            "Resource": "*",
            "Condition": {
                "StringNotEquals": {
                    "aws:ResourceTag/allow": "false"
                }
            }
```

```
        }
    ]
}
```

ステップ 2: ユーザーまたはグループにポリシーをアタッチする

IAM ポリシーを作成したら、キーと値のペアを使用して、Lightsail リソースを作成する必要があるユーザーやグループにアタッチします。ユーザーまたはグループに IAM ポリシーをアタッチする方法の詳細については、IAM ドキュメントの「IAM ポリシーの追加と削除」を参照してください。

タグを使用して、Lightsail リソースのコストを整理する

Amazon Lightsail でタグを使用し、自分のコスト構造に合わせて AWS の請求を整理できます。そのためには、Lightsail リソースにキーと値のタグを追加します。次に、これらのタグを AWS Billing and Cost Management コンソールで有効にします。最後に、サインアップし、AWS アカウントの請求書とタグキー値が追加されたコスト配分レポートを取得します。このセットアップ手順について以下に説明します。

> **Note**
>
> Lightsail のタグ、タグ付けできるリソース、およびタグの制限の詳細については、「タグ」を参照してください。

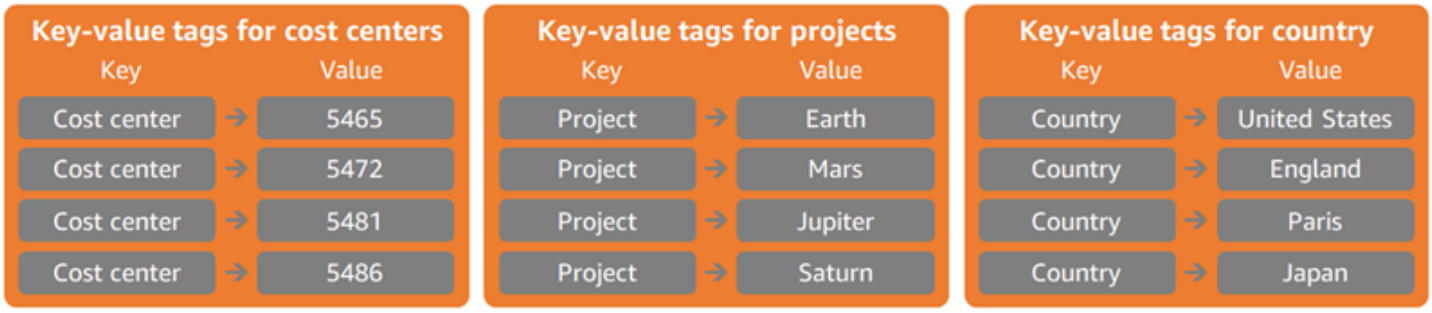
> **Important**
>
> 現時点では、コスト配分タグを追加しても、Lightsail データベーススナップショットをコスト配分レポートで追跡することはできません。

ステップ 1: キーと値のタグを リソースに追加する

請求コンソールで整理する対象の Lightsail リソースにキーと値のタグを追加します。キーバリュー型のタグの詳細については、「リソースにタグを追加する」を参照してください。

コストの分類方法を表すタグキーのセットを規定しておくことをお勧めします。コスト配分レポートの追加の列にタグキー、各行に該当値が表示されます。したがって、一貫したタグキーのセットを使用すると、より効率的にコストを追跡できます。たとえば、複数の Lightsail リソースに特定のコス

トセンターをタグとして追加できます。これを行うには、「Cost center」キーと数値のペアを使用します。次に、このコストセンターに関する請求を複数のリソースをまたいで表示するように請求情報を整理します。次の例は、コスト配分を整理するために使用できるキーと値のタグを示しています。



ステップ 2: ユーザー定義のコスト配分タグを有効にする

必要なタグを Lightsail リソースに追加したら、これらのタグを[請求とコスト管理]コンソールでコスト配分用に有効化します。たとえば、「Cost center」キータグを作成したら、このキータグを [請求とコスト管理] コンソールで有効にして、このタグのコスト配分レポートを生成します。詳細については、AWS Billing and Cost Management ドキュメントの「ユーザー定義のコスト配分タグの有効化」を参照してください。

ステップ 3: コスト配分レポートを設定して表示する

月次コスト配分レポートには、アカウントの AWS 使用状況が製品カテゴリ別およびリンクされたユーザー別に表示されます。このレポートには、詳細な請求レポートと同じ明細項目が表示され、さらに追加してタグキー用の列が表示されます。月別コスト配分レポートを設定するには、AWS Billing and Cost Management ドキュメントの「月別コスト配分レポートの設定」を参照してください。

レポートの保存先の Amazon Simple Storage Service (Amazon S3) バケットは、コスト配分レポートの設定時に定義済みです。この定義済みの Amazon S3 バケットを開き、利用可能になったコスト配分レポートを開きます。コスト配分レポートの内容の詳細については、AWS Billing and Cost Management ドキュメントの「コスト配分レポートの表示」を参照してください。

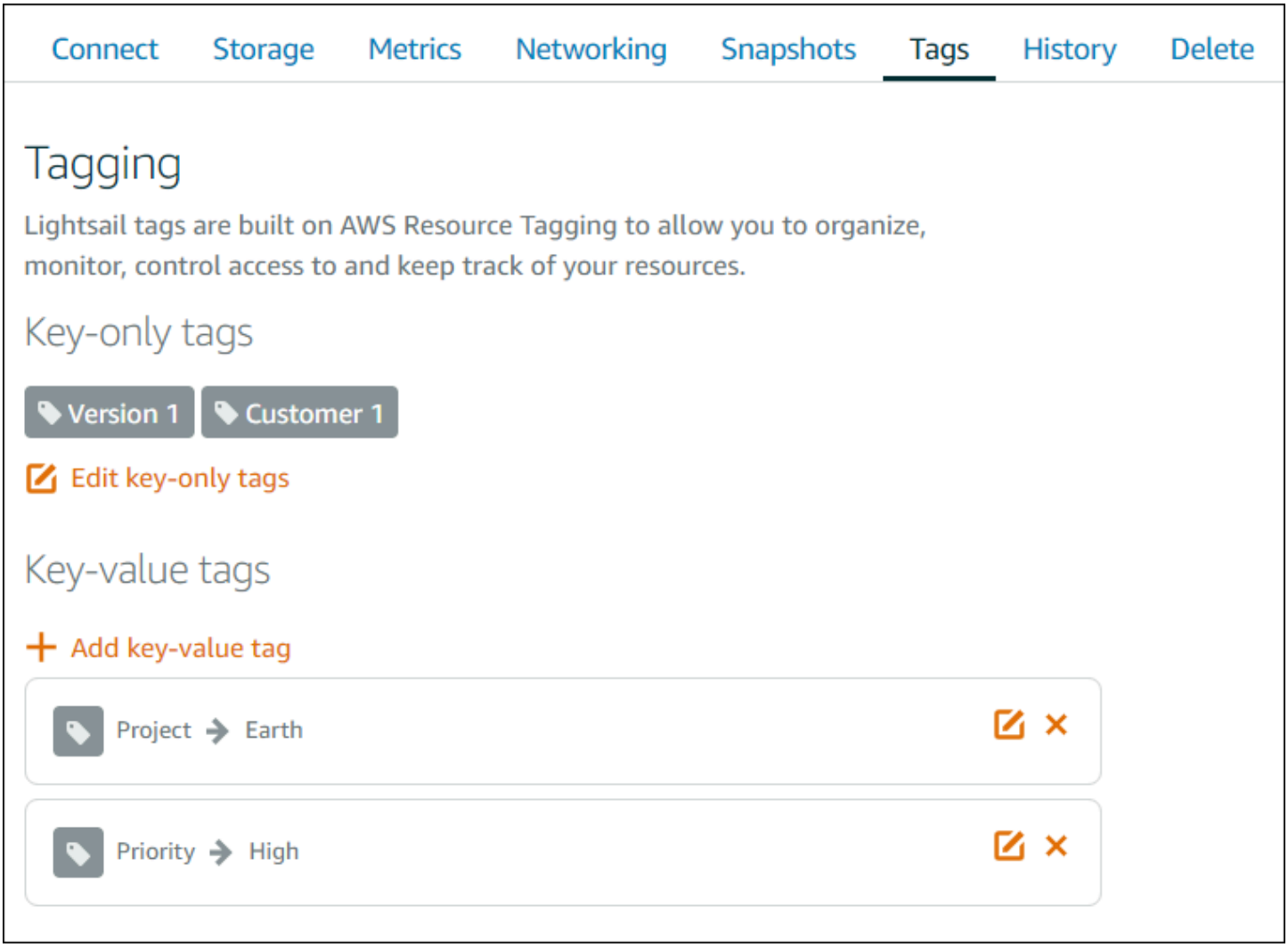
タグを使用して、Lightsail リソースを整理する

Amazon Lightsail リソースにタグを追加し、追加したタグでリソースをフィルタ処理できます。この操作を行うには、Lightsail コンソールでタグを選択または検索します。このガイドでは、タグを使用して Lightsail リソースを表示およびフィルタ処理する方法を示します。

> **Note**
>
> タグ、タグ付けできるリソース、およびタグの制限の詳細については、「タグ」を参照してください。

リソースのタグを表示する

インスタンス、コンテナサービス、CDN ディストリビューション、バケット、データベース、ディスク、DNS ゾーン、およびロードバランサーには、Lightsail コンソールを使ってタグ付けできるので、[Tags (タグ)] タブを含みます。このタブには、リソースの管理ページからアクセスできます。次の例は、インスタンスリソースの [タグ] タブです。[タグ] タブでは、タグを追加、編集、または削除できます。詳細については、「リソースにタグを追加する」と「タグを削除する」を参照してください。



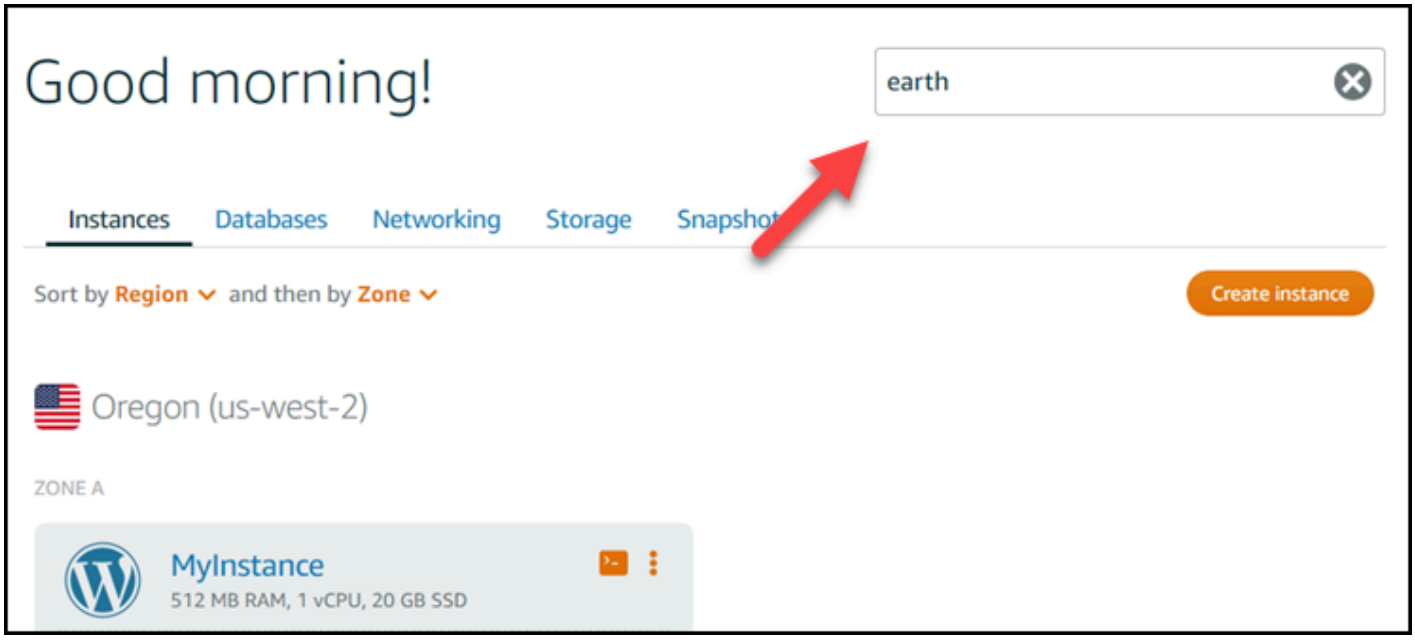
> **Note**
>
> Lightsail コンソールを使用して、インスタンス、コンテナサービス、 CDN ディストリビューションバケット、データベース、ディスク、DNS ゾーン、およびロードバランサーをタグ付けすることができます。ただし、Lightsail API オペレーション、AWS Command Line Interface (AWS CLI)、または SDK を使用すると、より多くの Lightsail リソースにタグを付けることができます。タグ付けをサポートする Lightsail リソースの詳細なリストについては、「タグ」を参照してください。

タグを使用してリソースをフィルタ処理する

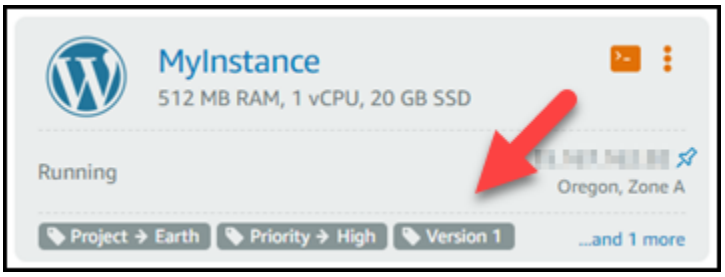
Lightsail コンソールでは、タグを使用してリソースをフィルタ処理するための以下のオプションを利用できます。どのオプションを使用した場合でも、Lightsail のホームページが更新され、検索または選択したタグのみが表示されます。

> **Note**
>
> フィルタ処理の各オプションは永続的です。タグでフィルタ処理した後で、Lightsail のホームページでセクション間を移動すると、フィルタが継続して適用されます。

- Lightsail のホームページで、[検索] テキストボックスにキーのみのタグまたはフィルタ処理用の値を入力し、Enter キーを押します。



- Lightsail のホームページでリソースの下に表示されるタグを選択します。



- リソースの見出しに表示されるタグを選択します。

Amazon Lightsail リソースのトラブルシューティング

以下のトピックは、Amazon Lightsail リソースで発生する可能性のある問題のトラブルシューティングに役立ちます。

トピック

WordPress Lightsail でのセットアップのトラブルシューティング

Amazon Lightsail WordPress のセットアップワークフローでは、次の 2 種類のエラーメッセージが表示されることがあります。

一般的なエラー

これらのタイプのエラーは、ワークフローの最終ステップで [証明書の作成] を選択した直後に発生します。これらのエラーは Lightsail コンソールの上部にあるバナーに表示されます。通常、WordPress 古いインスタンスでセットアップワークフローを実行したり、誤った情報を送信したりすることが原因です。たとえば、インスタンスのパブリック IP アドレスを指していない DNS レコードを選択した場合などです。

セットアップ失敗

この種のエラーは、ワークフローの最終ステップを完了してから数分以内に発生します。これらのエラーメッセージは、インスタンスの [Connect] タブの [WordPress ウェブサイトの設定] セク

ションに表示されます。これらのエラーは、インスタンスで Let's Encrypt HTTPS 証明書を設定できない場合に発生します。

以下のトピックの情報を参考にして、 WordPress セットアップガイド付きワークフローで発生する可能性のあるエラーの診断と修正に役立ててください。

トピック

- Lightsail WordPress のセットアップの一般的なエラーのトラブルシューティング
- Lightsail WordPress でのセットアップエラーのトラブルシューティング

Amazon Lightsail WordPress のセットアップガイド付きワークフローの詳細については、「WordPressインスタンスの設定」を参照してください。

Lightsail WordPress のセットアップの一般的なエラーのトラブルシューティング

ワークフロー中に送信された情報に問題がある場合、Lightsail コンソールの上部にエラーメッセージが表示されます。

メッセージの 1 行目には、セットアップでエラーが発生したことが通知されます。

*InstanceNameInstanceRegion*リージョンのインスタンスでセットアップを完了できませんでした。

2 行目には、セットアップで発生したエラーが含まれています。

エラーが発生したため、インスタンスに接続できなかったか、接続を維持できませんでした。

トラブルシューティングを開始するには、メッセージに表示されたエラーを以下のエラーのいずれかと照合します。

エラー

- DNS レコードが見つかりません。ドメインの DNS レコードがインスタンスのパブリック IP アドレスを指していることを確認し、DNS の変更が反映されるまで待ってください。
- DNS レコードが一致しません。ドメインの DNS レコードがインスタンスのパブリック IP アドレスを指していることを確認し、DNS の変更が反映されるまで待ってください。

- インスタンスに接続できない。SSH 接続の準備が完了するまで数分お待ちください。次に、セットアップを再度開始します。
- サポートされていないバージョンです WordPress 。 WordPress セットアップはバージョン 6 以降のみをサポートします。
- セットアップは 2023 年 1 月 1 WordPress 日以降に作成されたインスタンスのみをサポートします。
- インスタンスのファイアウォールポート 22、80、443 は、セットアップワークフロー中に任意の IP アドレスからの TCP 接続を許可する必要があります。これらの設定はインスタンスの [ネットワーク] タブで変更できます。

DNS レコードが見つかりません。ドメインの DNS レコードがインスタンスのパブリック IP アドレスを指していることを確認し、DNS の変更が反映されるまで待ってください。

理由

このエラーは、DNS レコードの設定が誤っているか、DNS レコードがインターネットの DNS 全体に伝達されるのに十分な時間がなかったことが原因です。

[修正]

A または AAAA DNS レコードが DNS ゾーンに存在し、インスタンスのパブリック IP アドレスを指していることを確認します。詳細については、「Lightsail の DNS」を参照してください。

Apex ドメイン (`example.com`) `www` とそのサブドメイン () からのトラフィックを指す DNS レコードを追加または更新する場合、それらはインターネットの DNS 全体に伝播する必要があります。`www.example.com`DNS の変更が反映されたかどうかは、nslookup や DNS 検索フォームなどのツールを使用して確認できます。MxToolbox

> **Note**
>
> DNS レコードの変更がインターネットの DNS 全体に反映されるまでしばらくお待ちください。この処理には数時間かかる場合があります。

DNS レコードが一致しません。ドメインの DNS レコードがインスタンスのパブリック IP アドレスを指していることを確認し、DNS の変更が反映されるまで待ってください。

理由

A または AAAA DNS レコードがインスタンスのパブリック IP アドレスを指していない。

[修正]

A または AAAA DNS レコードが DNS ゾーンに存在し、インスタンスのパブリック IP アドレスを指していることを確認します。詳細については、「Lightsail の DNS」を参照してください。

> **Note**
>
> DNS レコードの変更がインターネットの DNS に反映されるまでしばらくお待ちください。この処理には数時間かかる場合があります。

インスタンスに接続できない。SSH 接続の準備が完了するまで数分お待ちください。次に、セットアップを再度開始します。

理由

インスタンスは作成または再起動されたばかりで、SSH 接続の準備ができていません。

[修正]

SSH 接続の準備が完了するまで数分お待ちください。次に、ガイド付きワークフローを再試行してください。詳細については、「Lightsail での SSH のトラブルシューティング」を参照してください。

サポートされていないバージョンです WordPress 。 WordPress セットアップはバージョン 6 以降のみをサポートします。

理由

WordPress インスタンスにインストールされているバージョンはバージョン 6 より古いです。WordPress WordPress 古いバージョンには、互換性のないソフトウェアや依存関係が含まれているため、HTTPS 証明書が生成されません。

[修正]

Lightsail WordPress コンソールから新しいインスタンスを作成します。次に、 WordPress ウェブサイトを古いインスタンスから新しいインスタンスに移行します。詳細については、「WordPress 既存のブログの移行」を参照してください。

既存のインスタンスを置き換える新しいインスタンスを作成する場合は、新しいインスタンスへのアプリケーションの依存関係を必ず更新してください。

セットアップは 2023 年 1 月 1 WordPress 日以降に作成されたインスタンスのみをサポートします。

理由

セットアップで使用されているインスタンスには、古いソフトウェアが含まれている可能性があります。古いソフトウェアでは HTTPS 証明書は生成されません。

修正してください。

Lightsail WordPress コンソールから新しいインスタンスを作成します。次に、 WordPress ウェブサイトを古いインスタンスから新しいインスタンスに移行します。詳細については、「WordPress 既存のブログの移行」を参照してください。

既存のインスタンスを置き換える新しいインスタンスを作成する場合は、新しいインスタンスへのアプリケーションの依存関係を必ず更新してください。

インスタンスのファイアウォールポート 22、80、443 は、セットアップワークフロー中に任意の IP アドレスからの TCP 接続を許可する必要があります。これらの設定はインスタンスの [ネットワーク] タブで変更できます。

理由

インスタンスファイアウォールのポート 22、80、443 は、セットアップの実行中は任意の IP アドレスからの TCP 接続を許可する必要があります。このエラーは、これらのポートの 1 つ以上が閉じられたときに生成されます。詳細については、「インスタンスのファイアウォール」を参照してください。

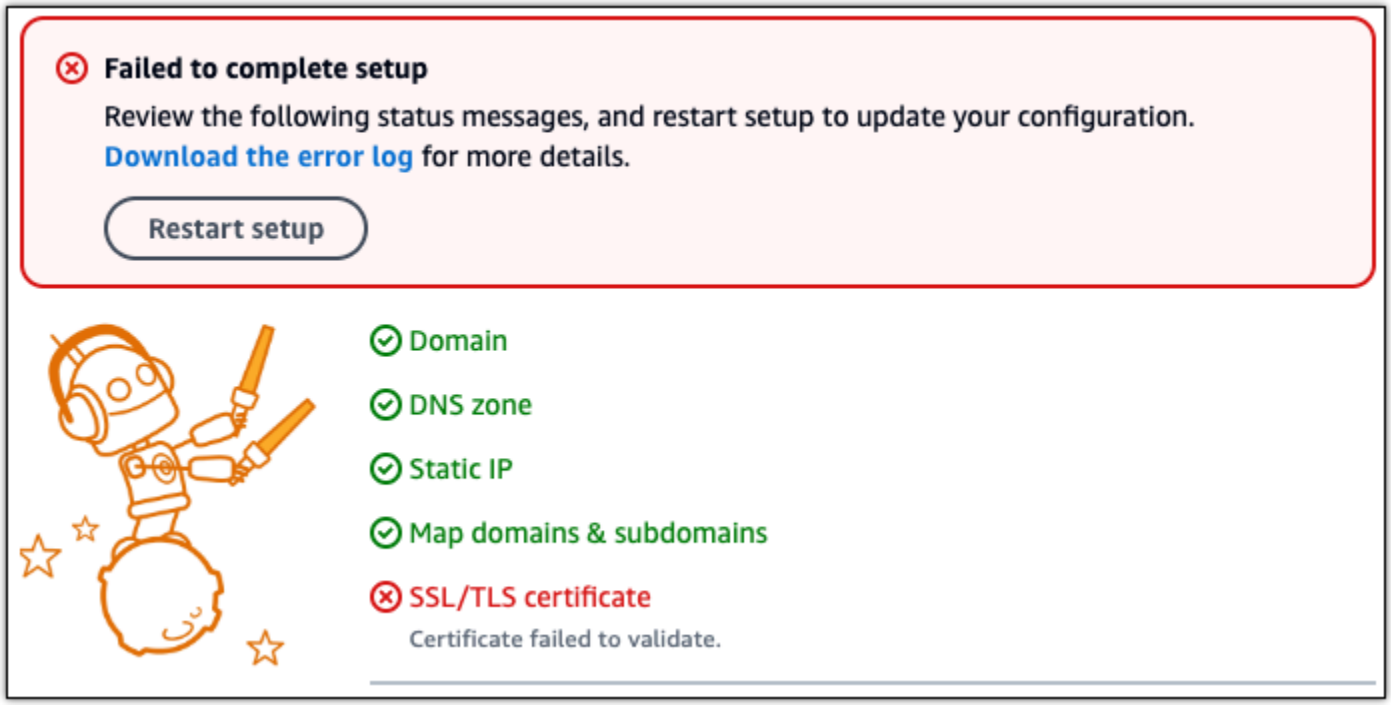
[修正]

インスタンスの IPv4 と IPv6 のファイアウォールルールを追加または編集して、ポート 22、80、443 を介した TCP 接続を許可します。詳細については、「インスタンスファイアウォールルールの追加と編集」を参照してください。

Lightsail WordPress でのセットアップエラーのトラブルシューティング

セットアップ失敗メッセージは、インスタンスの [Connect] タブの [WordPressウェブサイトのセットアップ] セクションに表示されます。セットアップの失敗は、ワークフローの最終ステップを完了してから数分以内に発生する可能性があります。Let's Encrypt HTTPS 証明書をインスタンスに設定できない場合に発生します。

セットアップを完了できませんでした — 以下のステータスメッセージを確認し、セットアップを再開して設定を更新してください。詳細については、エラーログをダウンロードしてください。



失敗メッセージから [エラーログをダウンロード] リンクを選択し、セットアップで生成されたエラーログをダウンロードして表示します。トラブルシューティングを開始するには、ログのエラーメッセージを以下のエラーのいずれかと照合します。

エラー

- CertBot.Errors。 AuthorizationError: 一部のチャレンジは失敗しました

- Certbot は一部のドメインを認証できませんでした。
- 過去 168 時間に、この一連のドメインに対して既に発行された証明書が多すぎます (5)。
- 認証に失敗した回数が多すぎる

CertBot.Errors。 AuthorizationError: 一部のチャレンジは失敗しました

理由

このエラーは、DNS レコードの設定が誤っているか、DNS レコードがインターネット全体に拡散するのに十分な時間がなかったことが原因です。

[修正]

A または AAAA DNS レコードが DNS ゾーンに存在し、インスタンスのパブリック IP アドレスを指していることを確認します。詳細については、「Lightsail の DNS」を参照してください。

Apex ドメイン (example.com) www とそのサブドメイン () からのトラフィックを指す DNS レコードを追加または更新する場合、それらはインターネット全体に伝播する必要があります。www.example.comDNS の変更が反映されたかどうかは、nslookup やからの DNS 検索などのツールを使用して確認できます。MxToolbox

> **Note**
>
> DNS レコードの変更がインターネットの DNS 全体に反映されるまでしばらくお待ちください。この処理には数時間かかる場合があります。

Certbot は一部のドメインを認証できませんでした。

理由

このエラーは、インスタンスで HTTPS 証明書が設定されている間に別のプロセスがポート 80 を使用している場合に発生することがあります。

[修正]

WordPress インスタンスを再起動します。次に、ガイド付きワークフローをもう一度実行します。再起動しても問題が解決しない場合は、以下の手順を使用して、ポート 80 で実行されているインスタンス上の実行中のプロセスをすべて終了します。

手順

1. Lightsail ブラウザベースの SSH クライアントを使用するか、を使用してインスタンス Connect。AWS CloudShell
2. インスタンスで実行されている Bitnami プロセスを停止します。

```
$ sudo /opt/bitnami/ctlscript.sh stop
```

   Bitnami プロセスが停止していることを確認します。

```
sudo /opt/bitnami/ctlscript.sh status
```

3. ポート 80 を使用しているプロセスが他にないか確認します。

```
fuser -n tcp 80
```

4. 他のアプリケーションが必要としないプロセスをすべて終了します。

```
fuser -k -n tcp 80
```

5. WordPress セットアップを再開します。

過去 168 時間に、この一連のドメインに対して既に発行された証明書が多すぎます (5)。

理由

過去 1 週間に、1 つ以上のドメインまたはサブドメインを使用して 5 つの証明書が作成されました。詳細については、Let's Encrypt ウェブサイトの「レート制限」を参照してください。

修正

1 週間 (168 時間) 待ってから、このドメインのガイド付きワークフローを再開してください。

認証に失敗した回数が多すぎる

理由

リクエスト内の 1 つ以上のドメインまたはサブドメインが、1 時間あたり 5 回の検証という制限を超えました。詳細については、Let's Encrypt ウェブサイトの「レート制限」を参照してください。

修正

1 時間待ってから、 WordPress セットアップを再実行してください。セットアップを再開する前に、他の検証エラーが修正されていることを確認してください。

Lightsail の 403 (アクセス拒否) エラーをトラブルシューティングする

Lightsail コンソールにアクセスしようとしたときに403 エラーが表示された場合、あわてる必要はありません。問題のトラブルシューティングを行うには、以下のステップを試してください。

- AWS アカウントまたは AWS Identity and Access Management (IAM) ユーザーを最近作成した場合は、数分待ってから、ブラウザを更新してください。
- 最後にサインインしてから時間が経っている場合は、ブラウザを更新します。再度サインインするように求められた場合は、必ず Lightsail にアクセスできる IAM ユーザーを使用してください。
- IAM ユーザーに Lightsail にアクセスする権利がない場合は、AWS アカウントのルートユーザーまたは管理者アクセス権を持つ IAM ユーザーに問い合わせて、Lightsail へのアクセス権をリクエストします。詳細については、「IAM ユーザーの Amazon Lightsail へのアクセスを管理します。」を参照してください。
- 上記のステップを試した後で 403 エラーが続く場合は、AWS カスタマーサポートにお問い合わせください。2011 年より前に作成された AWS アカウントでは、まれにサポートがユーザーのアカウントを手動で Lightsail にサブスクライブする必要があります。

Lightsail ディスクの問題のトラブルシューティング

Lightsail のブロックストレージディスクでエラーが発生する可能性があります。このトピックでは、一般的な問題とそれらのエラーの回避策について説明します。

一般的なディスクエラー

以下の問題の中から発生している問題に最も近いものを選択し、リンク先に移動して問題を解決します。リストにない問題が発生した場合、このページの一番下にある [ご質問は? [意見などありますか ?] このページ下部のリンクにアクセスしてフィードバックを送信するか、AWS サポートにお問い合わせください。

インスタンスにアタッチされたままのためディスクを削除できない。

まず、ディスクをインスタンスからデタッチし、その後ディスクを削除してください。詳細については、「ブロックストレージディスクをデタッチおよび削除する」を参照してください。

実際のエラーメッセージ: You can't perform this operation because the disk is still attached to a Lightsail instance: ***YOUR_INSTANCE***

ディスクのステータスがエラーです。

[エラー] ステータスは、Lightsail ディスクに関連する基盤ハードウェアに障害が発生したことを示します。ディスクを最新のスナップショットから復元できます。そうしない場合、ディスクに関連するデータを回復できません。詳細については、「スナップショットからブロックストレージディスクを作成する」を参照してください。

[エラー] のステータスのディスクについては請求されません。

Lightsail インスタンスはまだ実行中のためディスクをデタッチできない。

まず、インスタンスを停止し、その後ディスクをデタッチしてください。詳細については、「インスタンスの停止」を参照してください。

実際のエラーメッセージ:You can't detach this disk right now. このディスクの状態: ***DISK_STATE***

16 TB (16,384 GB) より大きいカスタムディスクサイズを指定できない。

小さいディスクを作成してみます。追加ディスクは、最大 16 TB です。ディスクが 16 TB 未満の場合でも作成できない場合、リスト内の次のエラーが発生する可能性があります (大きいディスクが多すぎる)。これは、AWS アカウント全体の追加ディスクストレージが 20 TB を超えることができないためです。詳細については、「ブロックストレージディスク」を参照してください。

実際のエラーメッセージ:The size of a block storage disk must be between 8 and 16384 GB.

Lightsail でさらにディスクを作成できない。

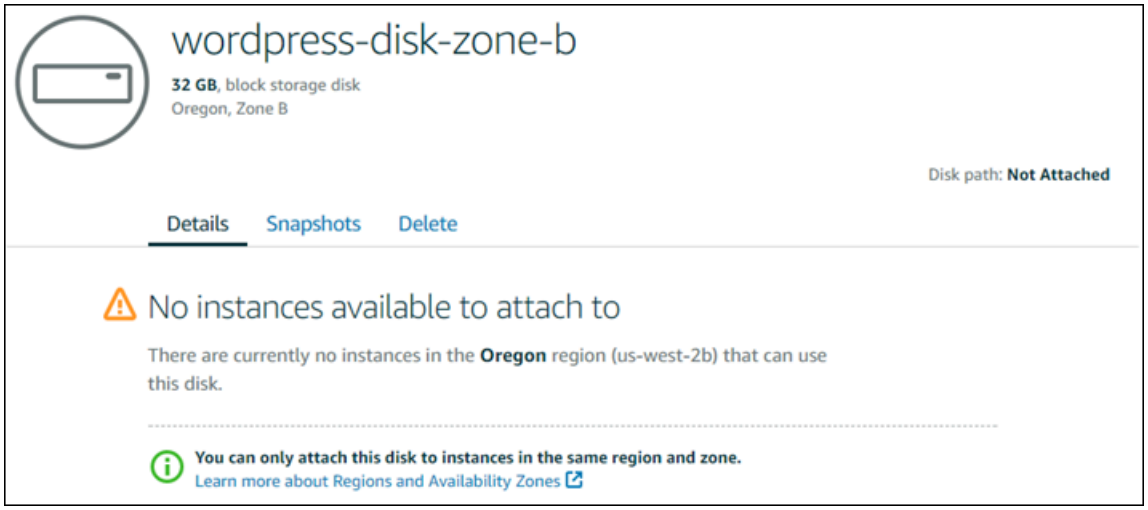
作成できるディスク数のクォータに達した可能性があります。または、AWS アカウントで大きいディスクを多く作成しすぎた可能性があります (ディスクストレージの合計サイズが 20 TB を

超えることはできません)。詳細については、「ブロックストレージディスク」を参照してください。

実際のエラーメッセージ: You've reached the maximum size limit of all disks in this account. or You've reached the limit of disks in this account.

ディスクを Lightsail インスタンスにアタッチできない

次のエラーが発生した場合、ディスクをアタッチする予定のインスタンスと同じ AWS リージョンおよびアベイラビリティーゾーンにディスクを再作成する必要があります。



実際のエラーメッセージ:There are currently no instances in the ***AWS Region*** that can use this disk.

Lightsail ブラウザベースの SSH または RDP クライアントの接続に関する問題のトラブルシューティング

Amazon Lightsail コンソールで使用可能なブラウザベースの SSH または RDP クライアントを使用してインスタンスに接続しようとすると、エラーメッセージが表示されることがあります。このエラーが表示される理由として考えられるものは、次のセクションで説明します。

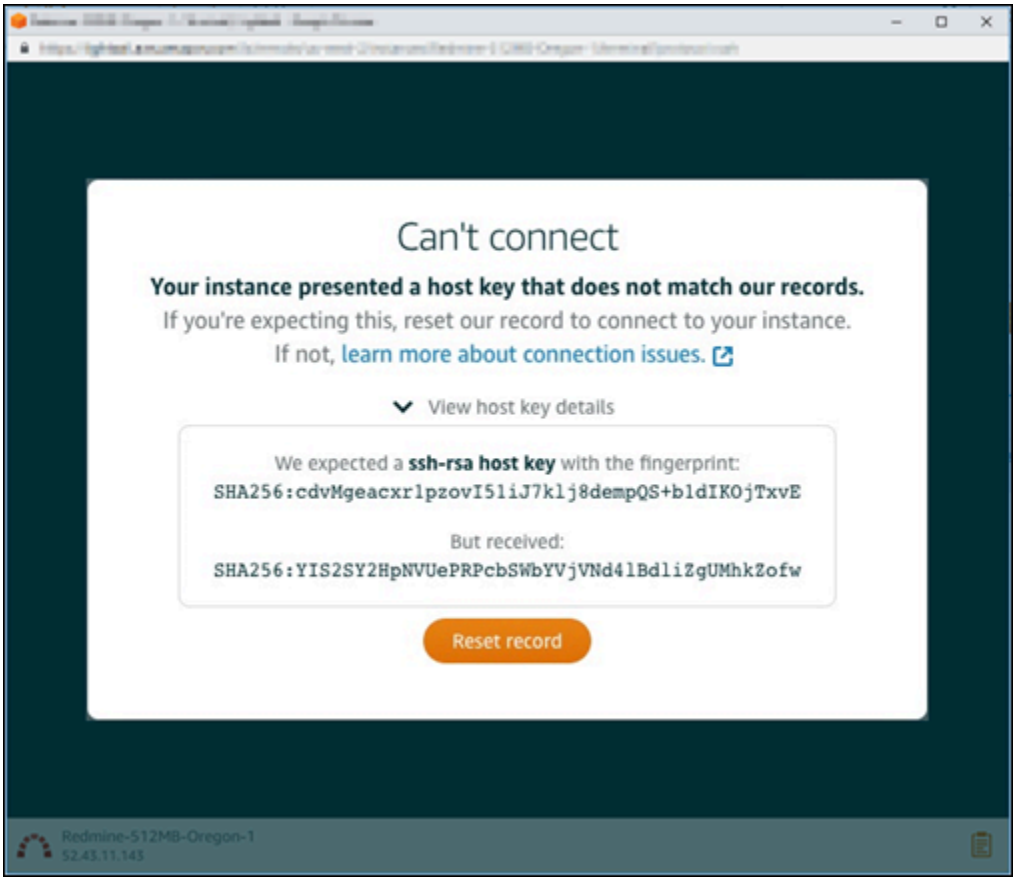
> **Important**
>
> Lightsail ブラウザベースの SSH/RDP クライアントは IPv4 トラフィックのみを受け入れます。サードパーティーのクライアントを使用して、IPv6 経由でインスタンスに SSH または RDP 接続します。詳細については、「インスタンスに接続します」を参照してください。

エラーメッセージ: 接続できません

接続を試みると、SSH および RDP ブラウザベースのクライアントは、ホストキーまたは証明書の検証を使用してインスタンスを認証します。インスタンスが Lightsail が記録しているホストキーまたは証明書と一致しないホストキーまたは証明書を提示すると、2 つのエラーメッセージのうち 1 つが表示されます。このセクションでは、両方のエラーメッセージが表示・説明されています。

接続できません。レコードをリセットしてください

次のエラーメッセージは、ホストキーまたは証明書の不一致があり、その不一致が最新のオペレーティングシステムのアップグレード、またはユーザーまたは別のユーザーによるホストキーまたは証明書の意図的な更新によって発生した可能性があると Lightsail が判断した場合に表示されます。この場合、Lightsail は、ホストキーまたは証明書の不一致は、ブラウザとインスタンス間のネットワークの不正なアクターによって発生していないと判断しました。



不一致が予想される場合、[Reset record (レコードをリセット)] を選択します。このアクションは、Lightsail がインスタンスのレコードに登録しているホストキーまたは証明書を削除し、ブラウザベースの SSH または RDP セッションがインスタンスに接続できるようにします。

次の AWS Command Line Interface (AWS CLI) コマンドを使用して、Lightsail が記録しているホストキーまたは証明書を削除することもできます。には*InstanceName*、既知のホストキーまたは証明書を削除するインスタンスの名前を入力します。[*Region*] では、インスタンスの AWS リージョンを入力します。

```
aws lightsail delete-known-host-keys --region Region --instance-name InstanceName
```

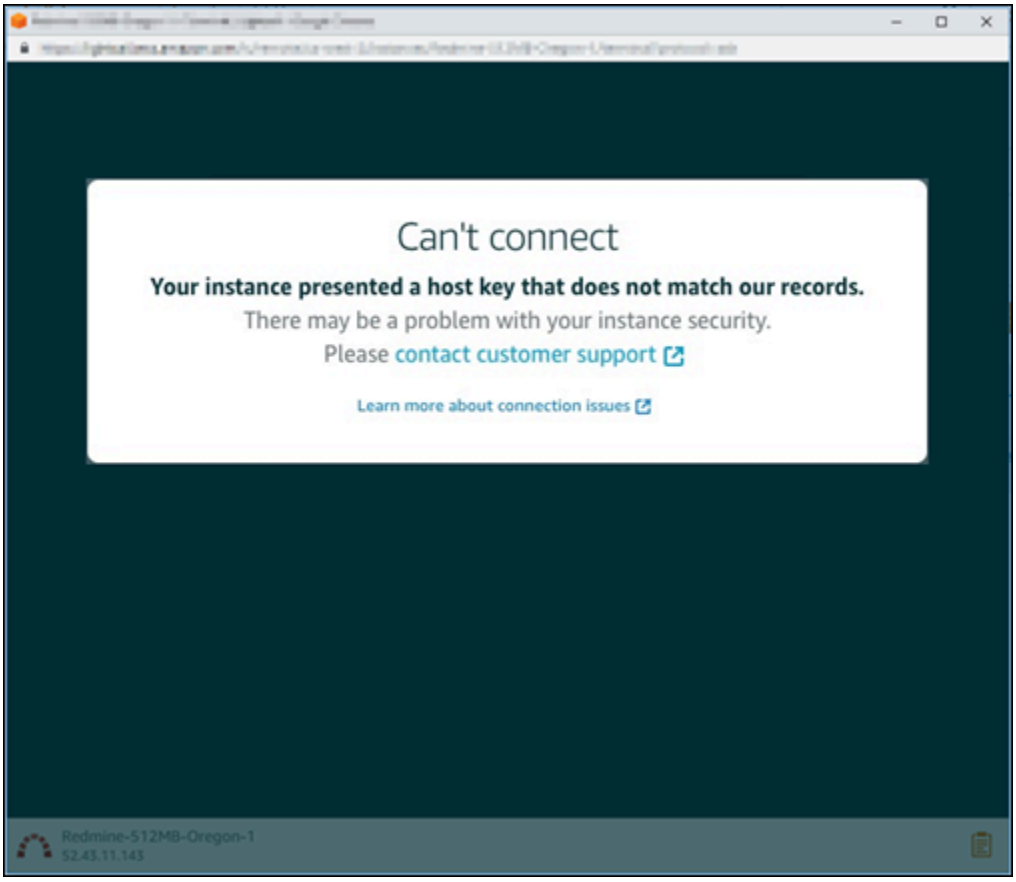
例：

```
aws lightsail delete-known-host-keys --region us-west-2 --instance-
name WordPress-512MB-Oregon-1
```

> **Note**
>
> の詳細についてはAWS CLI、「Lightsail と連携AWS CLIするように を設定する」を参照してください。

接続できません。カスタマーサポートにご連絡ください

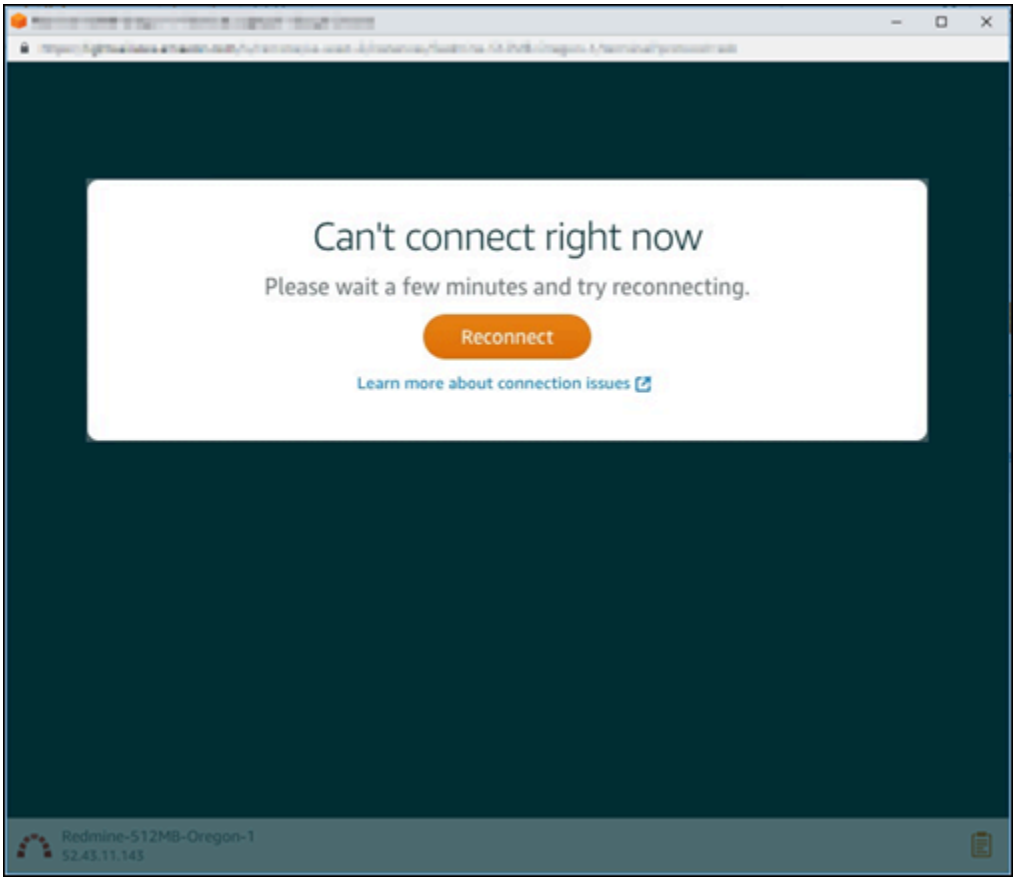
次のエラーメッセージは、ホストキーまたは証明書の不一致があり、Lightsail が man-in-the-middle 攻撃など、さらなる調査を必要とする疑わしいアクティビティがあると判断した場合に表示されます。



このエラーメッセージは、ブラウザベースの SSH または RDP クライアントを使用してインスタンスに接続できないことを意味します。サポートにご連絡ください。

エラーメッセージ: 現在接続できません。

インスタンスを作成、再起動、または再起動のいずれかを行った後にまだ起動していないインスタンスに接続しようとすると、次のエラーメッセージが表示されます。数分間待機した後、[Reconnect (再接続)] を選択して再度試してください。



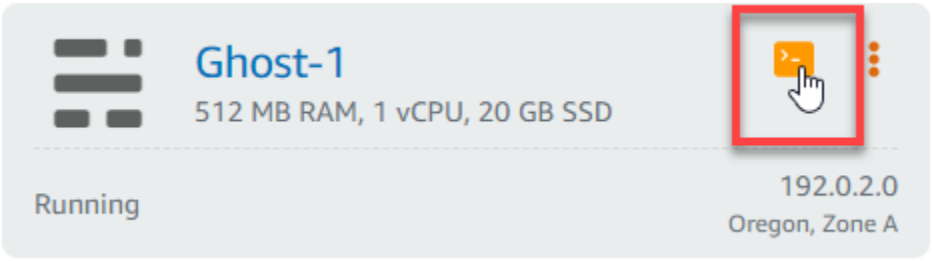
上記の手順を試しても接続できない場合は、AWS サポートにお問い合わせください。

Lightsail における Ghost インスタンスの 503 サービス使用不可エラーのトラブルシューティング

Amazon Lightsail で新しい Ghost インスタンスを作成し、ウェブサイトにアクセスしようとすると、サービスが利用できないことを示すエラー (503) が表示される場合があります。場合によっては、インスタンスの作成時にインスタンスの Ghost サービスが自動的に開始されないことがあります。このエラーは、インスタンスに 3.50 USD/月のバンドルを選択したときに発生する可能性があります。以下の手順を使用して Ghost サービスを開始し、サービス使用不可エラーを解決します。

Ghost サービスの開始

1. Lightsail コンソールにサインインします。
2. Lightsail ホームページで、[Instances (インスタンス)] タブを選択します。
3. Ghost インスタンスのブラウザベースの SSH クライアントアイコンを選択します。



4. SSH クライアントが接続されたら、以下のコマンドを入力してインスタンスですべてのサービスを再起動します。

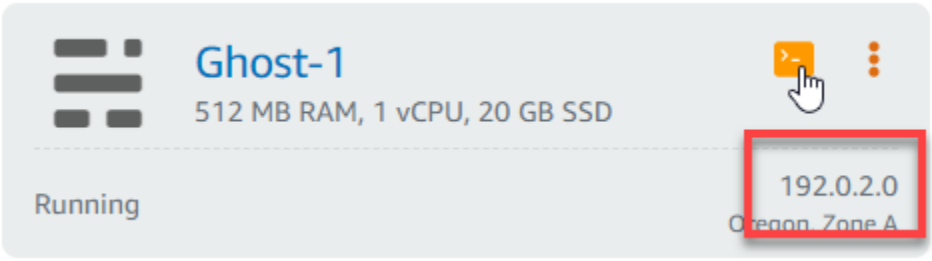
```
sudo /opt/bitnami/ctlscript.sh restart
```

以下の例のような結果が表示されるはずです。

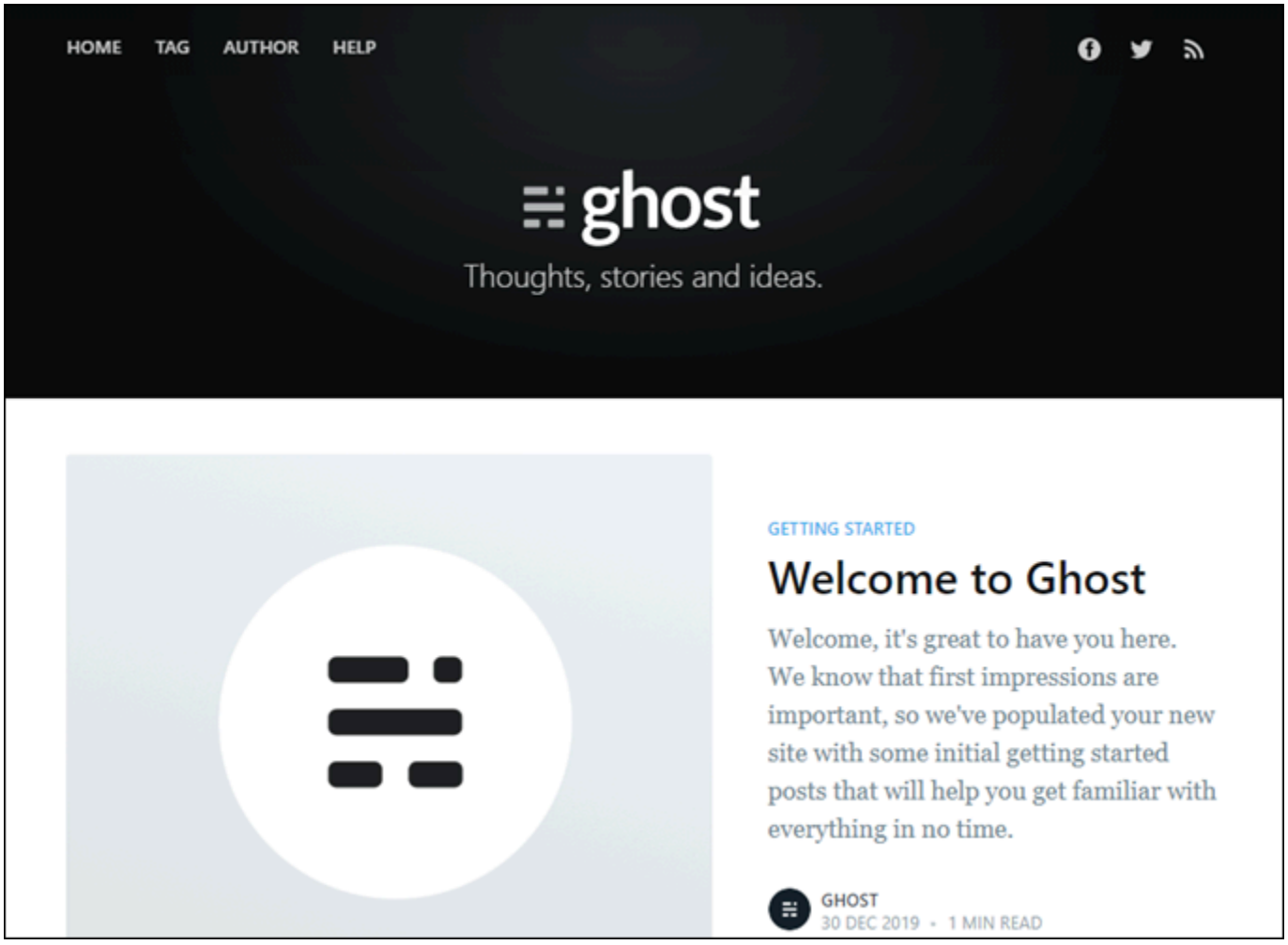
```
bitnami@ip-172-26-11-214:~$ sudo /opt/bitnami/ctlscript.sh restart
Syntax OK
/opt/bitnami/apache2/scripts/ctl.sh : httpd stopped
/opt/bitnami/apps/ghost/scripts/ctl.sh : ghost not running
/opt/bitnami/mysql/scripts/ctl.sh : mysql stopped
/opt/bitnami/mysql/scripts/ctl.sh : mysql  started at port 3306
» Ensuring user is not logged in as ghost user [skipped]
» Checking if logged in user is directory owner [skipped]
✔ Checking current folder permissions
✔ Validating config
✔ Checking memory availability
✔ Checking binary dependencies
✔ Starting Ghost: 127-0-0-1

------------------------------------------------------------------------------

Your admin interface is located at:

    http://18.237.117.48:80/ghost/

/opt/bitnami/apps/ghost/scripts/ctl.sh : ghost started
Syntax OK
/opt/bitnami/apache2/scripts/ctl.sh : httpd started at port 80
```

5. インスタンスのパブリック IP アドレスを参照して、Ghost ウェブサイトが稼働中であることを確認します。

インスタンスのパブリック IP アドレスは、Lightsail コンソールの [インスタンス] タブのインスタンス名の横に記載されています。



新しい Ghost インスタンスのパブリック IP を参照すると、デフォルトの Ghost ウェブサイトテンプレートが表示されます。



Lightsail の Identity and Access Management (IAM) のトラブルシューティング

次の情報は、Lightsail と IAM の使用に伴って発生する可能性がある一般的な問題の診断や修復に役立ちます。

Lightsail でアクションを実行する権限がない

AWS Management Console から、アクションを実行することが認可されていないと通知された場合、管理者に問い合わせ、サポートを依頼する必要があります。担当の管理者はお客様のユーザー名とパスワードを発行した人です。

次のエラー例は、`mateojackson` IAM ユーザーが Lightsail コンソールにアクセスしようとしたが、`lightsail:*` (フルアクセス) アクセス許可がない場合に発生します。

この場合、Mateo は、`lightsail:*` (フルアクセス) アクセス許可を使用して Lightsail コンソールにアクセスできるように、ポリシーの更新を管理者に依頼します。

iam:PassRole を実行する権限がありません

iam:PassRole アクションを実行する権限がないというエラーが表示された場合は、ポリシーを更新して Amazon Lightsail にロールを渡すことができるようにする必要があります。

一部の AWS のサービス では、新しいサービスロールまたはサービスにリンクされたロールを作成せずに、既存のロールをサービスに渡すことが許可されています。そのためには、サービスにロールを渡す許可が必要です。

以下の例のエラーは、 marymajor という IAM ユーザーがコンソールを使用して Amazon Lightsail でアクションを実行しようする場合に発生します。ただし、このアクションをサービスが実行するには、サービスロールから付与されたアクセス許可が必要です。Mary には、ロールをサービスに渡す許可がありません。

```
User: arn:aws:iam::123456789012:user/marymajor is not authorized to perform:
 iam:PassRole
```

この場合、メアリーのポリシーを更新してメアリーに iam:PassRole アクションの実行を許可する必要があります。

サポートが必要な場合は、AWS 管理者に問い合わせてください。サインイン資格情報を提供した担当者が管理者です。

アクセスキーを表示したい

IAM ユーザーアクセスキーを作成した後は、いつでもアクセスキー ID を表示できます。ただし、シークレットアクセスキーを再表示することはできません。シークレットアクセスキーを紛失した場合は、新しいキーペアを作成する必要があります。

アクセスキーは、アクセスキー ID (例: AKIAIOSFODNN7EXAMPLE) とシークレットアクセスキー (例: wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY) の 2 つの部分から構成されています。ユーザー名とパスワードと同様に、リクエストを認証するために、アクセスキー ID とシークレットアクセスキーの両方を使用する必要があります。ユーザー名とパスワードと同様に、アクセスキーを安全に管理してください。

> **Important**
>
> 正規のユーザー ID を確認するためであっても、アクセスキーをサードパーティーに提供しないでください。これを行うと、AWS アカウントへの永続的なアクセス権が第三者に付与される可能性があります。

アクセスキーペアを作成する場合、アクセスキー ID とシークレットアクセスキーを安全な場所に保存するように求めるプロンプトが表示されます。このシークレットアクセスキーは、作成時にのみ使用できます。シークレットアクセスキーを紛失した場合、IAM ユーザーに新しいアクセスキーを追加する必要があります。アクセスキーは最大 2 つまで持つことができます。既に 2 つある場合は、新しいキーペアを作成する前に、いずれかを削除する必要があります。手順を表示するには、「IAM ユーザーガイド」の「アクセスキーの管理」を参照してください。

管理者として Lightsail へのアクセスを他のユーザーに許可したい

Amazon Lightsail へのアクセスを他のユーザーに許可するには、アクセスを必要とする人またはアプリケーションの IAM エンティティ (ユーザーまたはロール) を作成する必要があります。ユーザーまたはアプリケーションは、このエンティティの認証情報を使用して AWS にアクセスします。次に、Amazon Lightsail の適切なアクセス許可を付与するポリシーを、そのエンティティにアタッチする必要があります。

すぐにスタートするには、「IAM ユーザーガイド」の「IAM が委任した初期のユーザーおよびグループの作成」を参照してください。

AWS アカウント以外の人が私の Lightsail リソースにアクセスできるようにしたい

他のアカウントのユーザーや組織外のユーザーが、リソースにアクセスするために使用できるロールを作成できます。ロールの引き受けを委託するユーザーを指定することができます。リソースベースのポリシーまたはアクセス制御リスト (ACL) をサポートするサービスの場合、それらのポリシーを使用して、リソースへのアクセスを付与できます。

詳細については、以下を参照してください。

- Amazon Lightsail がこれらの機能をサポートしているかどうかを確認するには、Amazon Lightsail と IAM の連携について をご参照ください。

- 所有している AWS アカウント 全体のリソースへのアクセス権を提供する方法については、「IAM ユーザーガイド」の「所有している別の AWS アカウント アカウントへのアクセス権を IAM ユーザーに提供」を参照してください。
- サードパーティーの AWS アカウント にリソースへのアクセス権を提供する方法については、「IAM ユーザーガイド」の「第三者が所有する AWS アカウント へのアクセス権を付与する」を参照してください。
- ID フェデレーションを介してアクセスを提供する方法については、「IAM ユーザーガイド」の「外部で認証されたユーザー (ID フェデレーション) へのアクセスの許可」を参照してください。
- クロスアカウントアクセスでのロールとリソースベースのポリシーの使用の違いの詳細については、「IAM ユーザーガイド」の「IAM ロールとリソースベースのポリシーとの相違点」を参照してください。

Lightsail で IPv6 到達可能性を検証する

ping ツールを使用して、ローカルコンピュータから Amazon Lightsail インスタンスへの IPv6 接続を検証できます。Ping は、2 つ以上のネットワークデバイス間の接続の問題をトラブルシューティングするために使用されるネットワーク診断ユーティリティです。ping が成功すると、IPv6 経由でインスタンスに接続できるようになります。ネットワーク設定またはデバイスが IPv6 を許可するように設定されていない場合、ping コマンドは失敗します。詳細については、「IPv6 に関する考慮事項」を参照してください。

内容

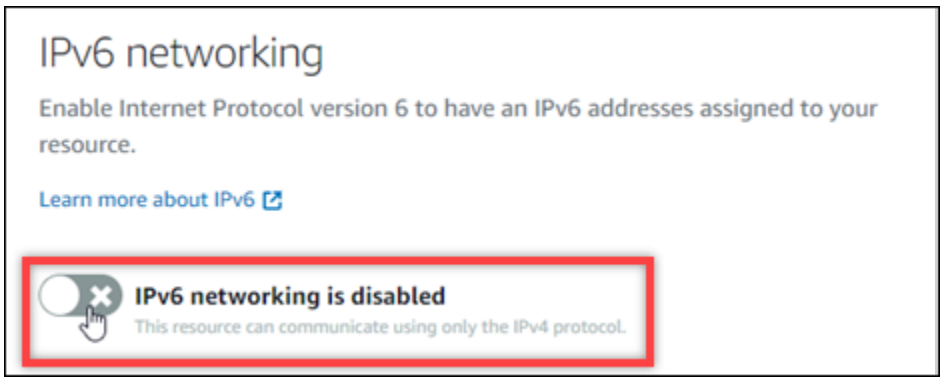
デュアルスタックインスタンスで IPv6 を有効にする

テストを開始する前に、デュアルスタックインスタンスの IPv6 を有効にします。IPv6 のみのインスタンスではIPv6-onlyは常にオンになっています。

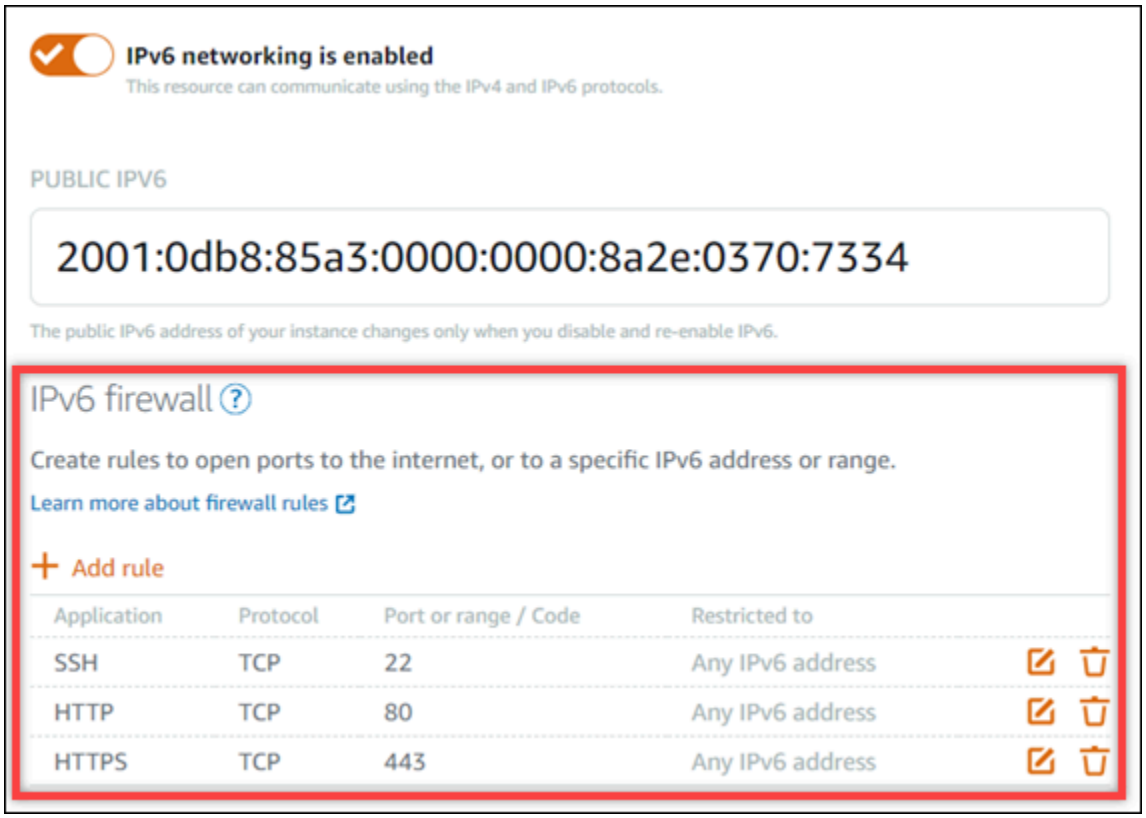
デュアルスタックインスタンスが有効でない場合は、次の手順を実行して IPv6 を有効にします。

1. Lightsail コンソールにサインインします。
2. IPv6 を有効にするインスタンスの名前を選択します。インスタンスが実行中であることを確認します。

3. インスタンス管理ページからネットワークタブを選択します。
4. ページの IPv6 ネットワークセクションで IPv6 を有効にします。



IPv6 を有効にすると、パブリック IPv6 アドレスがインスタンスに割り当てられ、IPv6 ファイアウォールが使用可能になります。



5. ページの上部にあるインスタンスのパブリック IPv4 アドレスとパブリック IPv6 アドレスを書き留めておきます。これらは以下のセクションで使用します。

インスタンスのファイアウォールを設定する

Lightsail コンソールのファイアウォールは仮想ファイアウォールとして機能します。つまり、パブリック IP アドレスを介してインスタンスに接続できるトラフィックを制御します。Lightsail で作成する各デュアルスタックインスタンスには、IPv4 アドレス用のファイアウォールと IPv6 アドレス用のファイアウォールがあります。各ファイアウォールには、インスタンスに着信するトラフィックをフィルタリングする一連のルールが含まれています。どちらのファイアウォールも互いに独立しています。IPv4 と IPv6 のファイアウォールルールを個別に設定する必要があります。IPv6-onlyインスタンスプランを持つインスタンスには、設定できる IPv4 ファイアウォールがありません。

Internet Control Message Protocol (ICMP) トラフィック用にインスタンスのファイアウォールを設定するには、以下の手順を実行します。ping ユーティリティは ICMP プロトコルを使用してインスタンスと通信します。詳細については、「Amazon Lightsail のインスタンスファイアウォール」を参照してください。

> **Important**
>
> Windows および Linux には、ping コマンドをブロックできるオペレーティングシステム (OS) レベルのファイアウォールが含まれています。続行する前に、インスタンスの OS ファイアウォールが IPv4 および IPv6 経由の ICMP トラフィックを受け入れることができることを確認します。詳細については、次のドキュメントを参照してください。
>
> - Lightsail Windows インスタンスに接続する
> - Lightsail Linux または Unix インスタンスに接続する

1. Lightsail コンソールにサインインします。
2. ファイアウォールを設定するインスタンスの名前を選択します。
3. インスタンス管理ページからネットワーク タブを選択し、使用するファイアウォールのタイプに適したセクションの残りのステップを完了します。IPv4 の場合は、「IPv4 ファイアウォール」セクションの手順を完了します。IPv6 の場合は、「IPv6 ファイアウォール」セクションの手順を完了します。

   a. アプリケーションドロップダウンメニューから Ping (ICMP) を選択します。

   b. IP アドレスに制限 ボックスを選択して、ローカル送信元 IP アドレスまたは範囲からの接続を許可し、送信元 IP アドレスを入力します。 (オプション) 任意の IP アドレスからの接続を許可するには、ボックスを選択しないままにしておくことができます。このオプションはテスト環境でのみ使用することをお勧めします。

c. 新しいルールをインスタンスに適用するには、作成 を選択します。

インスタンスへの到達可能性をテストする

ローカルコンピュータまたはネットワークから Lightsail インスタンスへの IPv4 または IPv6 到達可能性をテストするには、以下の手順を実行します。でメモしたインスタンスのパブリック IPv4 アドレスと IPv6 アドレスが必要ですStep 5。

Linux、Unix、または macOS デバイスから

1. ローカルデバイスでターミナルウィンドウを開きます。
2. Lightsail インスタンスに ping を実行するには、次のいずれかのコマンドを入力します。コマンドにある *IP ####*の例を、インスタンスのパブリック IPv4 または IPv6 アドレスに置き換えます。

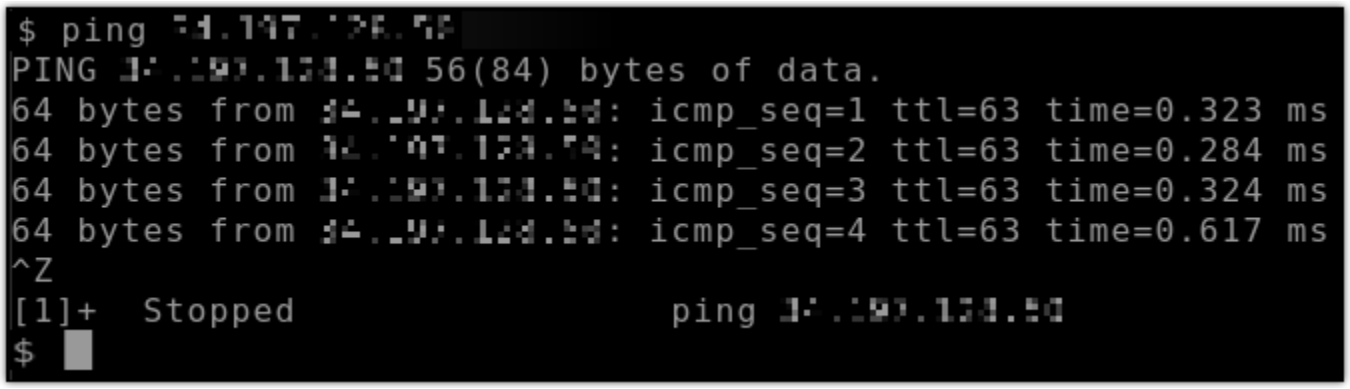
   IPv4 でテストするには

   ```
   ping 192.0.2.0
   ```

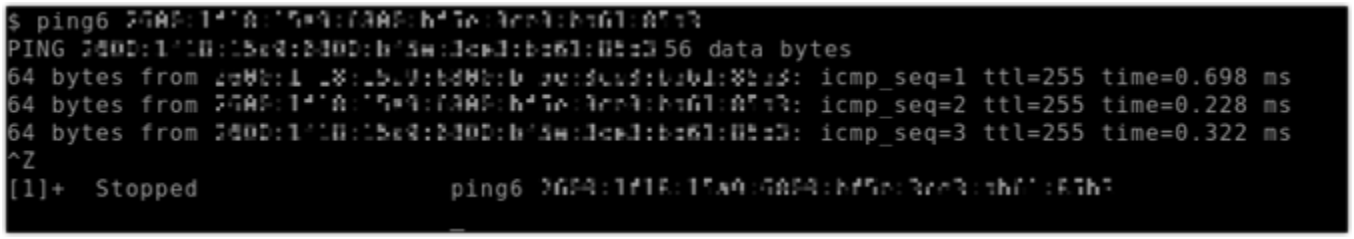
   IPv6 でテストするには

   ```
   ping6 2001:db8::
   ```
3. コマンドがいくつかの返信を返したら、デバイスのキーボードctrl+zに を入力してコマンドを停止します。

ping コマンドは、インスタンスの IPv4 アドレスが成功した場合、成功した返信を返します。結果は次の例のようになります。



ping6 コマンドは、インスタンスの IPv6 アドレスが成功した場合、成功した返信を返します。結果は次の例のようになります。



インスタンスに到達できない場合、どちらのコマンドもリクエストタイムアウトを返します。

Windows デバイスから

1. コマンドプロントを開きます。
2. Lightsail インスタンスに ping を実行するには、次のいずれかのコマンドを入力します。コマンド内の *IP ####*の例を、インスタンスのパブリック IPv4 または IPv6 アドレスに置き換えます。

   IPv4 でテストするには

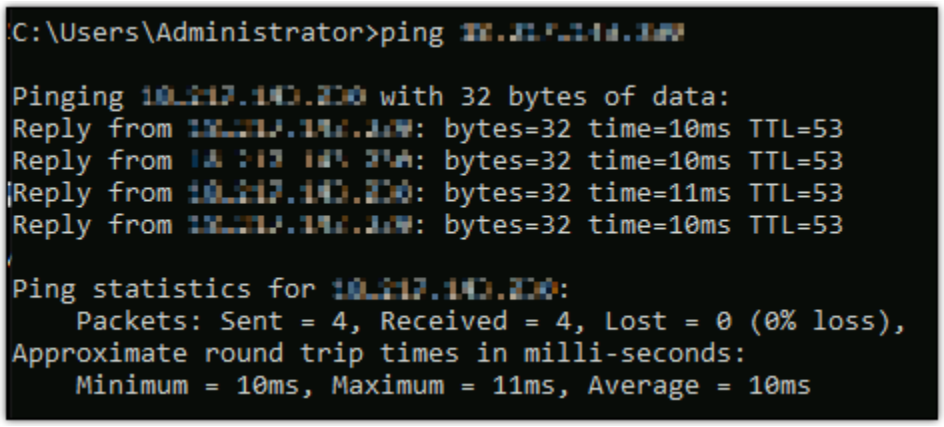
   ```
   ping 192.0.2.0
   ```

   IPv6 でテストするには

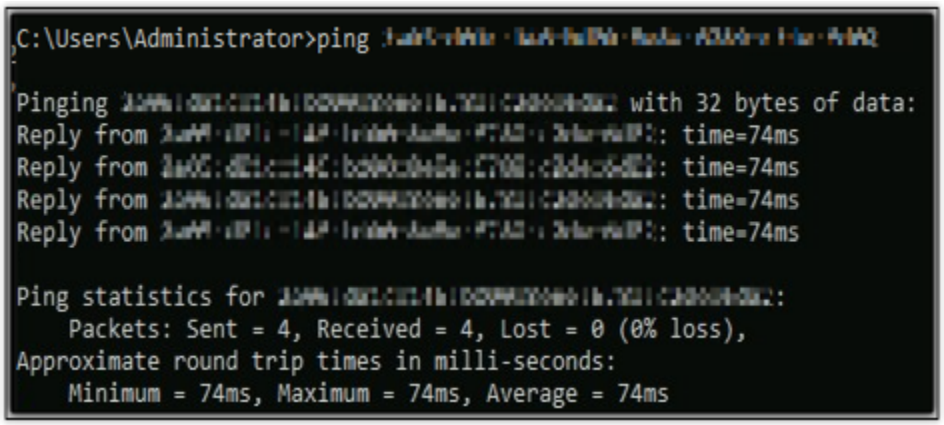
   ```
   ping 2001:db8::
   ```

3. コマンドがいくつかの返信を返したら、デバイスのキーボードctrl+zに を入力してコマンドを停止します。

ping コマンドは、インスタンスの IPv4 アドレスが成功した場合、成功した返信を返します。結果は次の例のようになります。



ping コマンドは、インスタンスの IPv6 アドレスが成功した場合、成功した返信を返します。結果は次の例のようになります。



どちらのコマンドも、インスタンスに到達できない場合にリクエストタイムアウトを返します。

Lightsail のインスタンス容量不足エラー

インスタンスを起動するか、停止したインスタンスを再度スタートしようとするとき、不足エラーが発生する場合があります。つまり、現時点では、AWS にはリクエストを満たすためのインスタンス容量がありません。次の内容はインスタンス容量不足エラーの例です。

InsufficientInstanceCapacity: インスタンスリクエストを満たすために十分な容量がありません。リクエスト内のインスタンス数を減らすか、追加の容量が利用可能になるまでお待ちください。より小さな Lightsail プラン (これは後でサイズ変更ができます) を選択することでインスタンスを起動することもできます。

このガイドでは、インスタンス容量不足エラーが発生した場合に実行できるアクションについて説明します。

目次

新しいインスタンスを起動するときの容量不足

新しいインスタンスを起動するときにインスタンス容量不足エラーが発生した場合、次のオプションを使用してください。各オプションを順番に入力することも、合ったオプションを選択することもできます。

1. 数分間待ってからリクエストを再度送信してください。インスタンス容量は頻繁に変化します。数分待ってもインスタンスを作成できない場合、オプション 2 に進みます。
2. インスタンスを作成するときは、別のアベイラビリティーゾーン (AZ) を選択します。各 AWS リージョン には 3 つ以上の AZ が含まれ、各 AZ が維持しているインスタンス容量は異なります。別の AZ を選択することにより、現在のインスタンス容量を活用できます。別の AWS リージョン または AZ にインスタンスを作成できない場合、オプション 3 に進んでください。
3. リクエスト内のインスタンスの数を減らします。複数のインスタンスを同時に作成する場合、インスタンスの数を減らしてリクエストを再送信してください。インスタンスの数を減らしても問題が解決しない場合、オプション 4 に進みます。
4. インスタンスを作成するときは別のインスタンスプランを選択してください。別の AZ または リージョンにインスタンスを作成できない場合、別のインスタンスプランを選択してください。インスタンスのサイズ変更は後で行えます。インスタンスのサイズ変更の詳細については、「スナップショットからインスタンスを作成する」を参照してください。

停止したインスタンスをスタートするときの容量不足

以前に停止した既存のインスタンスをスタートしたとき、インスタンス容量不足エラーが発生した場合、次のオプションを使用してください。

1. 数分間待ってからリクエストを再度送信してください。インスタンス容量は頻繁に変化します。数分待ってもインスタンスを作成できない場合、オプション 2 に進みます。
2. スナップショットから新しいインスタンスを作成します。停止したインスタンスのスナップショットを作成します。次に、スナップショットを使用し、元のインスタンスとは異なる新しいインスタンスを AZ に作成します。例えば、インスタンスが現在 us-east-2a (ゾーン A) にある場

合、新しいインスタンスを作成するときに us-east-2c (ゾーン C) を選択します。詳細については、「スナップショットからインスタンスを作成する」を参照してください。

3. スナップショットから新しいインスタンスを作成するとき、別のインスタンスプランを選択することもできます。このアクションはオプションです。

> **Important**
>
> 新しいインスタンスが実行状態になった後、新しいインスタンスにアクセスできてすべてが正常に動作していることを確認します。例えば、インスタンスがアプリケーションを実行していた場合、アプリケーションが期待どおりに動作していることを確認してください。このような場合、以前のインスタンスを削除できます。

関連情報

よくある質問

Lightsail での回復力

Lightsail ロードバランサーのトラブルシューティング

Lightsail ロードバランサーでエラーが発生する可能性があります。このトピックでは、一般的な問題とそれらのエラーの回避策について説明します。

ロードバランサーの一般的なエラー

以下の問題の中から発生している問題に最も近いものを選択し、リンク先に移動して問題を解決します。リストにない問題が発生した場合、このページの一番下にある [ご質問は? コメント?] リンクを使用してフィードバックを送信するか、AWS カスタマーサポートにお問い合わせください。

証明書を作成できません。

AWS アカウントに作成できる証明書の数にはクォータがあります。詳細については、AWS Certificate Manager ユーザーガイドの「クォータ」を参照してください。同じクォータがロードバランサーの Lightsail 証明書に適用されます。

実際のエラーメッセージ:Sorry, you've requested too many certificates for your account.

ロードバランサーに追加のインスタンスをアタッチできません。

AWS アカウントあたり合計 20 Lightsail インスタンスというクォータを超えない限り、任意の数の Lightsail インスタンスをロードバランサーにアタッチできます。

実際のエラーメッセージ:Sorry, you've reached the maximum number of instances you can attach to this load balancer.

ロードバランサーに特定のインスタンスをアタッチできません。

まず、Lightsail インスタンスが実行されていることをチェックします。停止している場合、インスタンス管理ページから開始することができます。ロードバランサーに正常にアタッチされるには、Lightsail インスタンスが実行されている必要があります。

同じインスタンスを多数のロードバランサーにアタッチした可能性があります。

実際のエラーメッセージ:Sorry, you've reached the maximum number of times an instance can be registered with a load balancer.

ロードバランサーをアタッチしようとしているインスタンスを Lightsail が見つけることができません

存在しなくなったか、ターゲットグループと同じ VPC 内に存在しないインスタンスをアタッチしようとしている可能性があります。

実際のエラーメッセージ:Sorry, the instance you specified doesn't exist, isn't in the same VPC as the target group, or has an unsupported instance type.

Lightsail での通知のトラブルシューティング

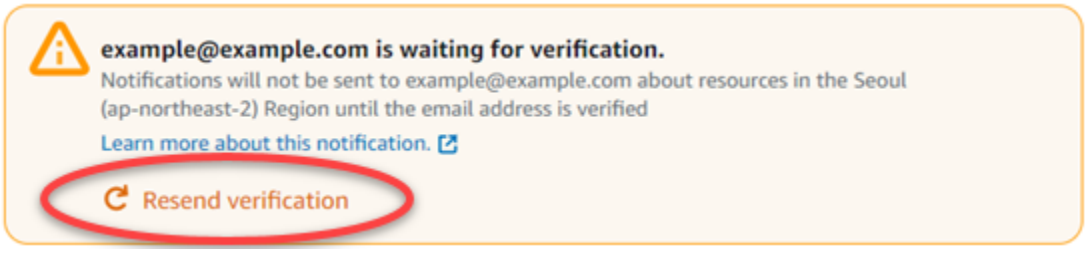
通知が予定されているときに通知を受け取らない場合は、通知の連絡先が正しく設定されていることを確認するためにいくつかの点を確認してください。通知の詳細については、「??? の通知」を参照してください 。

次の一覧では、発生する可能性がある一般的な通知連絡先の問題、原因とその解決方法について説明します。リストにない問題が発生した場合、このページの一番下にある 「ご質問は？ このページ下部の [フィードバック] リンクにアクセスしてフィードバックを送信するか、「AWS Support センター」にお問い合わせください。

メールアドレスを通知連絡先として追加しましたが、メール通知が届きません

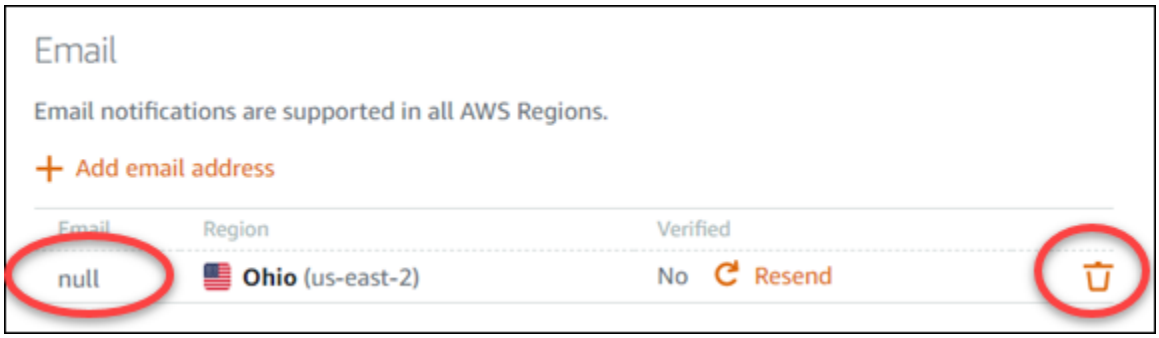
Lightsail でメールアドレスを通知連絡先として追加すると、そのアドレスに確認要求が送信されます。確認要求の メールには、受信者が Lightsail 通知を受信することを確認するためにクリックする必要があるリンクが含まれています。通知は、確認が完了するまでメールアドレスに送信されません。検証は、AWS 通知 <no-reply@sns.amazonaws.com> から行われ、件名は AWS 通知 - サブスクリプションの確認です。SMS メッセージングは検証を必要としません。

確認要求が受信トレイフォルダにない場合は、メールボックスのスパムフォルダと迷惑メールフォルダを確認してください。検証リクエストが失われた、または削除された場合は、Lightsail コンソールやアカウントページに表示される通知バナーで検証の再送を選択してください。



メール通知の連絡先として null が表示されています。

メールアドレスは、追加後 24 時間以内に確認する必要があります。24 時間以内に メールの確認に失敗した場合、その メールは自動的に `invalid` のステータスになり、Lightsail から削除されます。そのため、1 つ以上の メール通知連絡先に null の値が表示されることがあります。



この問題を解決するには、null メール通知の連絡先を削除し、正しいメールアドレスを再度追加します。メールアドレスは、Lightsail に追加した直後に確認してください。詳細については、「通知」を参照してください。

SMS テキストメッセージの通知が届かない、または最近受信が停止した

SMS テキストメッセージ通知の受信をオプトアウトしている可能性があります。ARRET (フランス語)、CANCEL、END、OPT-OUT、OPTOUT、QUIT、REMOVE、STOP、TD、UNSUBSCRIBE を使用して SMS テキストメッセージ通知に応答することでオプトアウトできます。携帯電話番号を

オプトアウトする場合、その携帯電話番号を Lightsail で通知連絡先として再度追加できるようになるまで 30 日待つ必要があります 。

Lightsail の SSL/TLS 証明書のトラブルシューティング

Lightsail ロードバランサーでエラーが発生する可能性があります。このトピックでは、一般的な問題とそれらのエラーの回避策について説明します。

以下の問題の中から発生している問題に最も近いものを選択し、リンク先に移動して問題を解決します。リストにない問題が発生した場合、このページの一番下にある [ご質問は? コメント?] リンクを使用してフィードバックを送信するか、AWS カスタマーサポートにお問い合わせください。

証明書を作成できません。

AWS アカウントに作成できる証明書の数にはクォータがあります。詳細については、AWS Certificate Manager ユーザーガイドの「クォータ」を参照してください。同じクォータがロードバランサーの Lightsail 証明書に適用されます。

実際のエラーメッセージ:Sorry, you've requested too many certificates for your account.

証明書リクエストに失敗しました。

証明書リクエストに失敗した場合、ロードバランサー管理ページの [インバウンドトラフィック] タブで [再試行] を実行できます。

原因がまだわからない場合は、AWS カスタマーサポートまでご連絡ください。

ドメインが無効と表示されました。

ドメインを管理していることを確認できない場合、DNS 管理にアクセスできることを確認します。アクセスできる場合は、こちらの手順に従います。この手順でも検証できない場合は、AWS カスタマーサポートにご連絡ください。

Amazon Lightsail のチュートリアル

以下のチュートリアルでは、一般的な Amazon Lightsail のユースケースについて説明します。たとえば、これらのチュートリアルでは、Lightsail のトラブルシューティング方法や他の AWS サービスと Lightsail との併用方法について説明しています。さらに、Bitnami WordPress や LAMP、Windows Server などのさまざまな Lightsail ブループリントを操作する方法も紹介します。

トピック

- Amazon Lightsail のクイックスタートガイド
- Amazon Lightsail の Bitnami チュートリアル
- WordPress Amazon Lightsail の チュートリアル
- Amazon Lightsail の WordPress マルチサイトチュートリアル
- Amazon Lightsail の Let's Encrypt チュートリアル
- Amazon Lightsail に関するネットワーキングのチュートリアル
- Amazon Lightsail で操作する

Amazon Lightsail のクイックスタートガイド

次のクイックスタートガイドを使用して、Lightsail ブループリントを始めましょう。Lightsail では、ブループリントはオペレーティングシステムとアプリケーションにあらかじめパッケージ化された仮想イメージです。アプリケーションには、WordPress、WordPress Multisite、cPanel & WHM、PrestaShop、Drupal、Ghost、Joomla!、Magento、Redmine、LAMP、Nginx (LEMP)、Node.js などがあります。

トピック

- クイックスタートガイド: cPanel & WHM
- クイックスタートガイド: Drupal
- クイックスタートガイド: Ghost
- クイックスタートガイド： GitLab CE
- クイックスタートガイド: Joomla!
- クイックスタートガイド: LAMP
- クイックスタートガイド: Magento

クイックスタートガイド: cPanel & WHM

ここでは、cPanel & WHM インスタンスが Amazon Lightsail で起動して実行された後に開始するためのいくつかのステップを示します。

> **Important**
>
> cPanel & WHM インスタンスには、15日間のトライアルライセンスが含まれています。15日後も継続して cPanel & WHM を使用するには cPanel からライセンスを購入する必要があります。ライセンスの購入をご検討の際は、ライセンスを購入する前にこのガイドのステップ 1~7 を完了してください。

目次

ステップ 1: ルートユーザーパスワードの変更

cPanel インスタンスのルートユーザーのパスワードを変更するには、次の手順を実行します。ルートユーザーとパスワードは、ウェブホストマネージャー (WHM) コンソールに後ほどサインインする際に使用します。

1. インスタンス管理ページの [接続] タブで、[SSH を使用して接続] を選択します。
2. 接続後、次のコマンドを入力してルートユーザーのパスワードを変更します。

```
sudo passwd
```

3. 強力なパスワードを入力し、もう一度入力してパスワードを確認します。

> **Note**
>
> パスワードは 7 文字以上で、一般的な単語は含まない必要があります。このガイドラインに沿わない場合、BAD PASSWORD の警告が表示されます。

このパスワードは、このガイドの後半で WHM コンソールにサインインする際に使用するので、覚えておいてください。

ステップ 2: cPanel & WHM インスタンスに静的 IP アドレスをアタッチする

インスタンスにアタッチしたデフォルトの動的なパブリック IP アドレスは、インスタンスを停止して開始するたびに変わります。パブリック IP アドレスが変わらないようにするには、静的 IP アドレスを作成してインスタンスにアタッチします。その後にドメイン名をインスタンスで使用すると、インスタンスを停止して開始するたびにドメインの DNS レコードを更新する必要がなくなります。または、インスタンスに障害が発生した場合にバックアップからインスタンスを復元し、新しいインスタンスに 静的 IP を再指定できます。1 つの静的 IP を 1 つのインスタンスにアタッチできます。

> **Important**
>
> cPanel からライセンスを購入する際は、cPanel & WHM インスタンスのパブリック IP アドレスを指定する必要があります。購入したライセンスは、その IP アドレスに関連付けられます。このため、cPanel からのライセンス購入をご検討される場合は、cPanel & WHM インスタンスに静的 IP をアタッチする必要があります。cPanel からライセンスを購入する際に静的 IP を指定し、Lightsail インスタンスで cPanel & WHM ライセンスを使用する予定がある

限り、静的 IP を保持します。後に別の IP アドレスにライセンスを移転する必要がある場合は、cPanel にリクエストを送信することができます。詳細については、WHM ドキュメントの「ライセンスの移転」を参照してください。

インスタンス管理ページで、[ネットワーキング] タブの [静的 IP の作成] を選択し、ページの手順に従います。

詳細については、「静的 IP を作成してインスタンスにアタッチする」を参照してください。

ステップ 3: ウェブホストマネージャーに初めてサインインする

WHM コンソールに初めてサインインするには、次の手順を使用します。

1. ウェブブラウザを開き、次のウェブアドレスに移動します。*<StaticIP>* をインスタンスの静的 IP アドレスに置き換えます。アドレスの末尾に :2087 を必ず追加してください。これがインスタンスへの接続を確立するためのポートになります。

   ```
   https://<StaticIP>:2087
   ```

   例:

   ```
   https://192.0.2.0:2087
   ```

   > **Important**
   >
   > インスタンスの IP アドレスとポートに移動する際は、ブラウザのアドレスバーに https:// を含める必要があります。そうしないと、サイトにアクセスできないというエラーが表示されます。

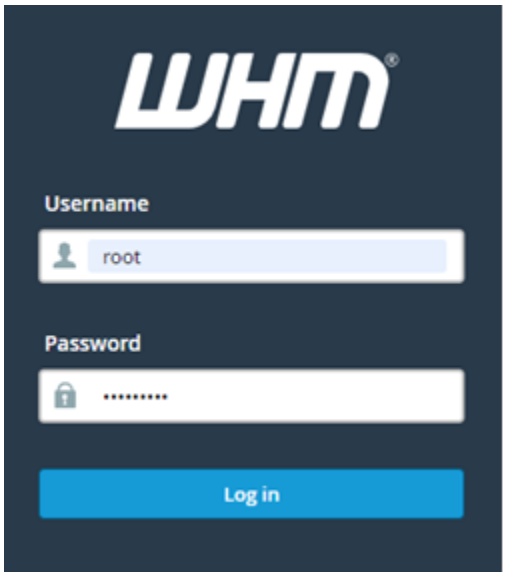
   ポート 2087 経由でインスタンスの静的 IP アドレスにブラウジングする際に、接続を確立できなかった場合は、ルーター、VPN、またはインターネットサービスプロバイダーがポート 2087 経由の HTTP/HTTPS 接続を許可しているかを確認してください。許可していない場合は、別のネットワークを使用して接続を試みてください。

   接続がプライベートではない、セキュリティで保護されていない、またはセキュリティ上のリスクがある、などの警告がブラウザに表示されることがあります。これは、 SSL/TLS 証明書

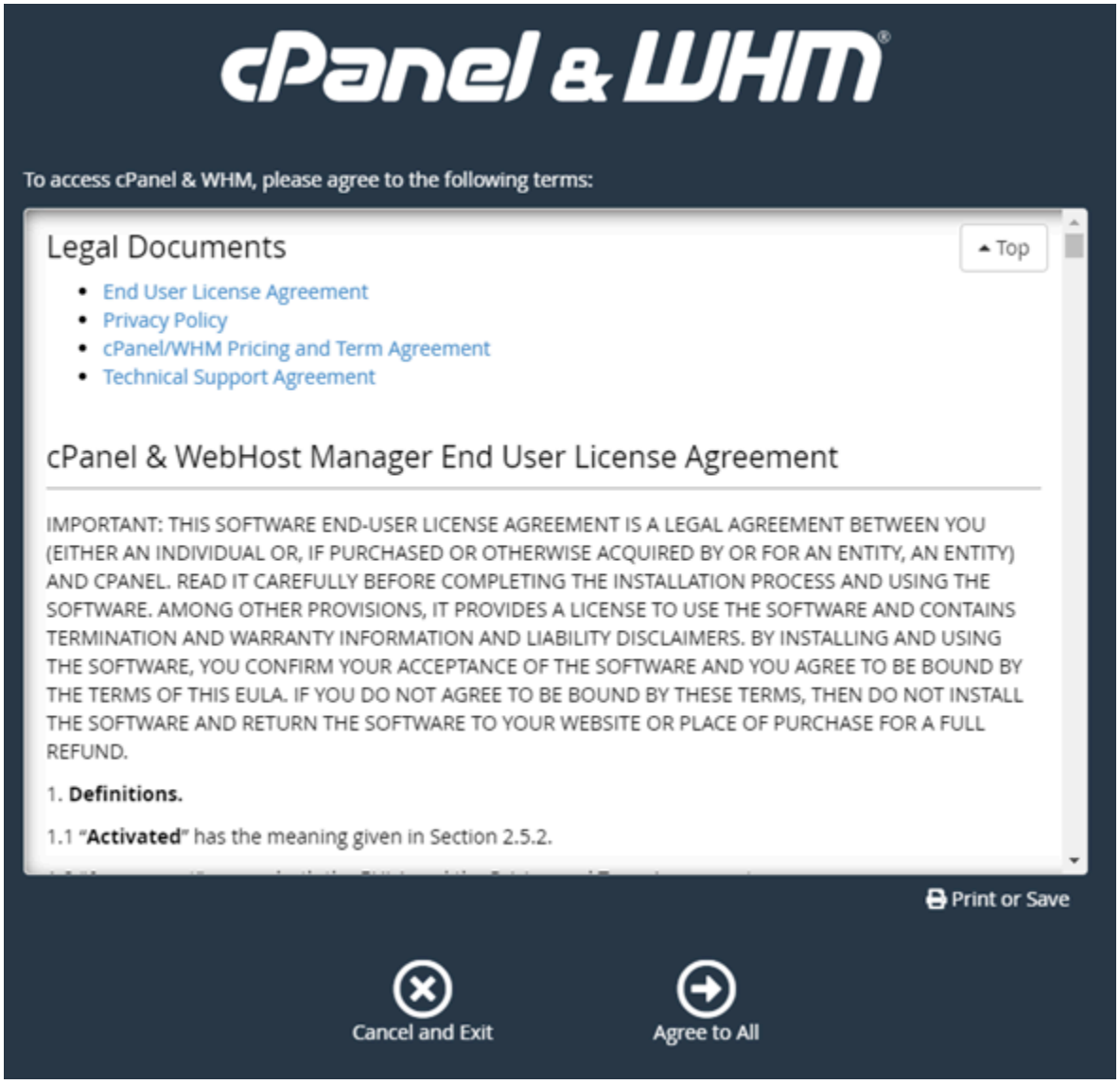
がまだ cPanel インスタンスに適用されていない場合に発生します。ブラウザウィンドウで、[Advanced] (詳細設定)、[Details] (詳細)、または [More information] (詳細情報) を選択して、使用可能なオプションを表示します。次に、プライベートまたは安全でない場合でも、ウェブサイトにアクセスすることを選択します。

2. [ユーザーネーム] テキストボックスに `root` を入力します。
3. ルートユーザーパスワードを [パスワード] テキストボックスに入力します。

   これは、このガイドの「ステップ 1: ルートユーザーパスワードを変更する」セクションで先ほど指定したパスワードになります。
4. [ログイン] を選択します。

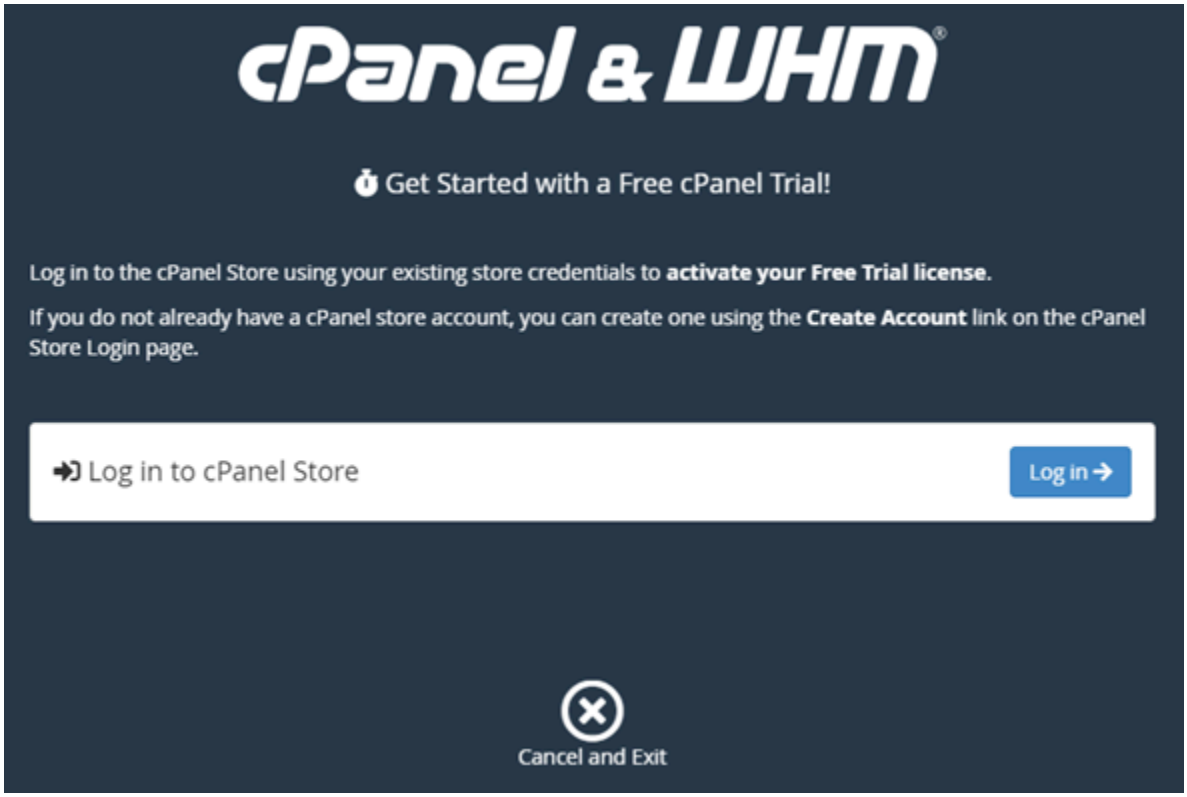


5. 続行する場合は、cPanel & WHM 規約を読んで、「すべてに同意する」を選択します。



6. cPanel の無料トライアルを開始するページで [ログイン] を選択して cPanel ストアにログインします。

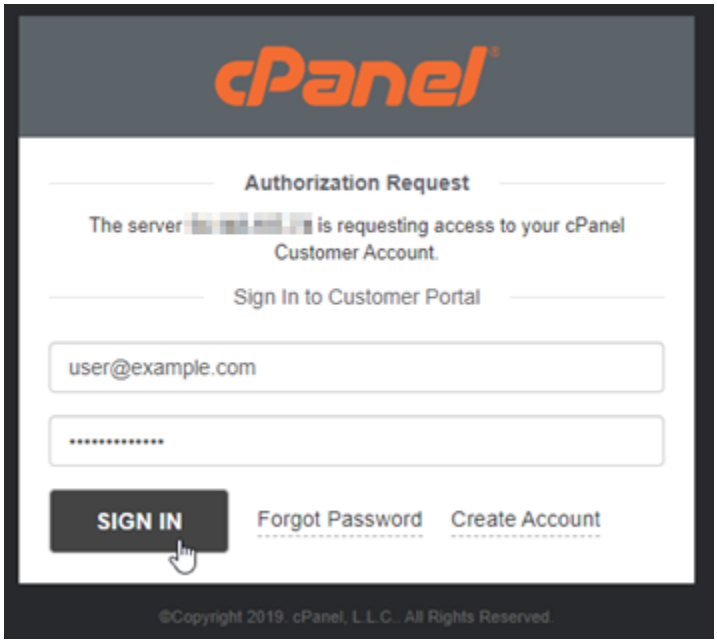
   ユーザーのアカウントにトライアルライセンスを関連付けるには、cPanel ストアにサインインする必要があります。cPanelストアのアカウントをお持ちでない場合も ログイン を選択します。アカウント作成のオプションが表示されます。



7. [認可リクエスト] ページが表示されたら、cPanel ストアのアカウント用のメールアドレスまたはユーザーネーム、およびパスワードを入力します。

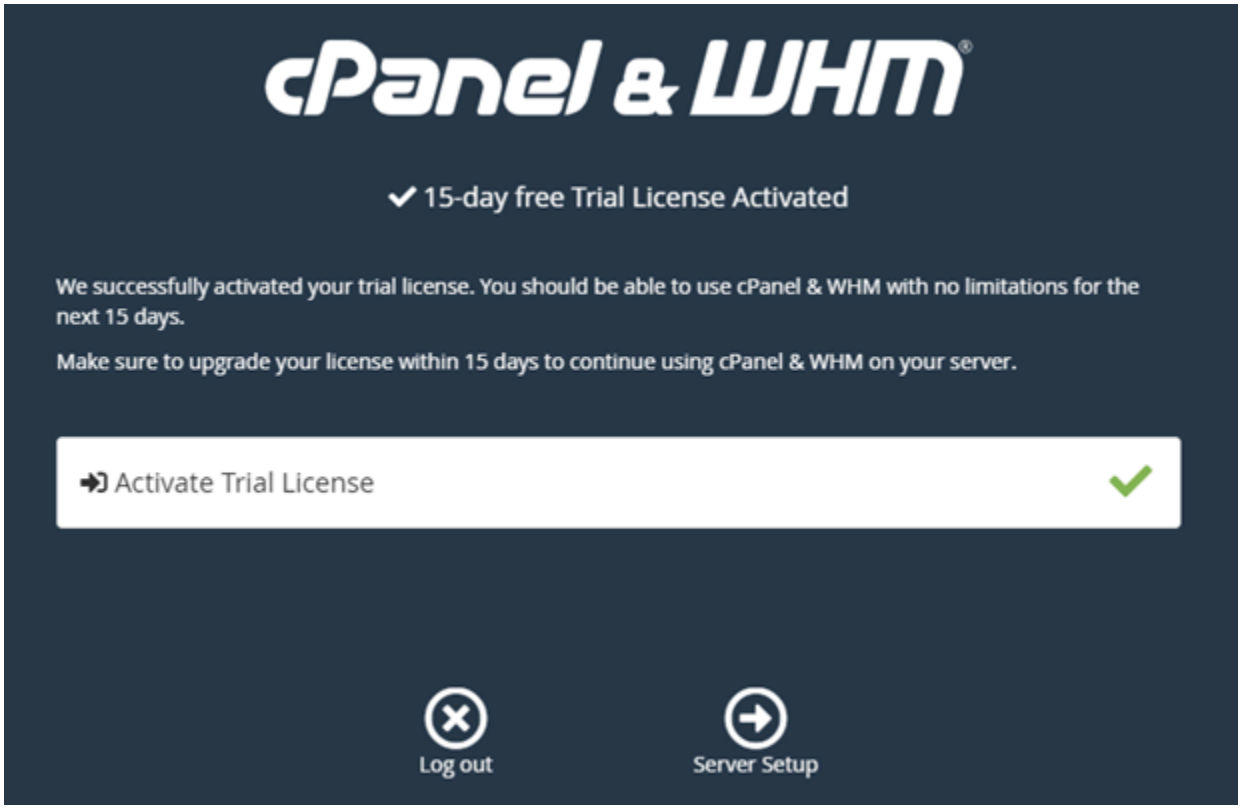
   cPanel ストアのアカウントをお持ちでない場合は、[アカウントの作成] を選択し、プロンプトに従って新しい cPanel ストアのアカウントを作成します。メールアドレスを入力するように求められます。cPanelストアアカウントのパスワードを設定するためのメールが送信されます。新しいブラウザを使用して cPanel ストアアカウントのパスワード設定を行うことをお勧めします。パスワードが設定されたらタブを閉じて、インスタンスに戻ってアカウント認証を行い、この手順の次のステップに進みます。

8. [サインイン] を選択します。

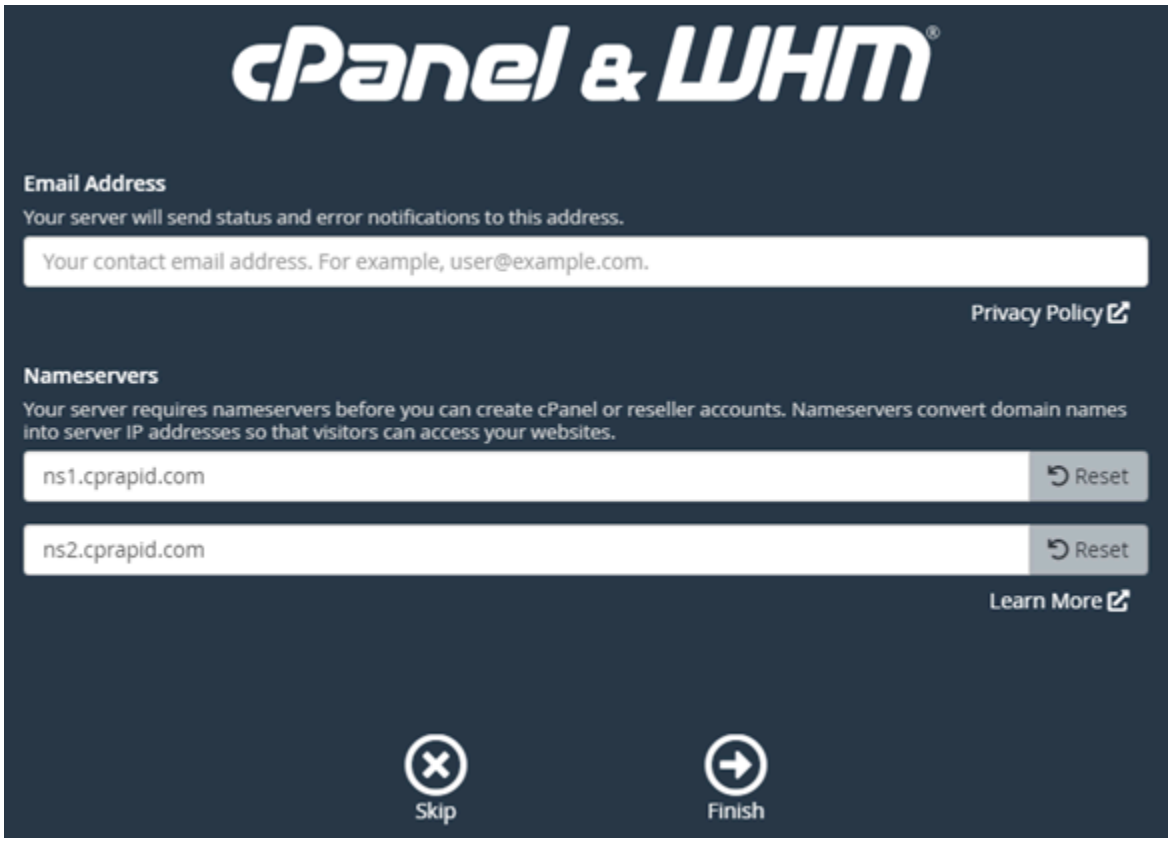


サインイン後、cPanel & WHM インスタンスは 15 日間のトライアルライセンスを取得します。これはユーザーの cPanel ストアのアカウントに関連付けられています。cPanel ストアの [ライセンスの管理] ページに移動して、トライアルライセンスを含む発行されたライセンスを確認します。

9. [サーバーのセットアップ] を選択して続行します。



10. メールアドレスとネームサーバーページの [スキップ] を選択します。これは後から設定することが可能です。

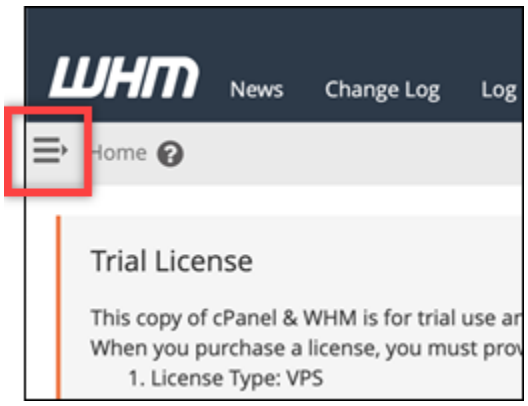


cPanel の設定と機能を管理することができる WHM コンソールが表示されます。

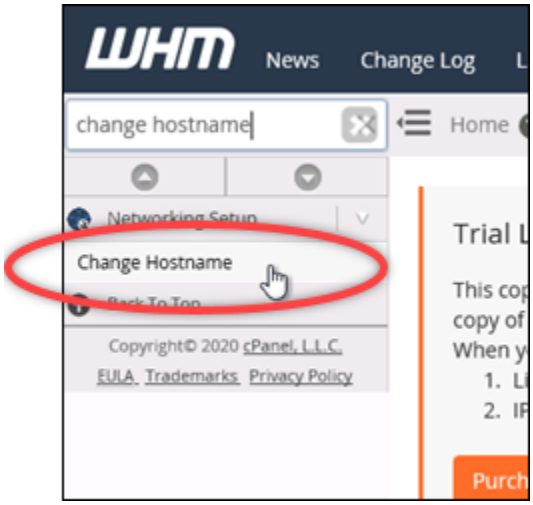
ステップ4：cPanel＆WHM インスタンスのホスト名と IPアドレスを変更する

インスタンスのホスト名を変更して、パブリック IP アドレスを使用しなくても WHM コンソールにアクセスできるようにするには、次の手順を実行します。このガイドの「ステップ 2: cPanel & WHM インスタンスに静的 IP アドレスをアタッチする」のセクションでインスタンスにアタッチした新しい静的 IP アドレスに、インスタンスの IP アドレスを変更する必要があります。

1. WHM コンソールの左上のセクションにある、ナビゲーションメニューアイコンを選択します。

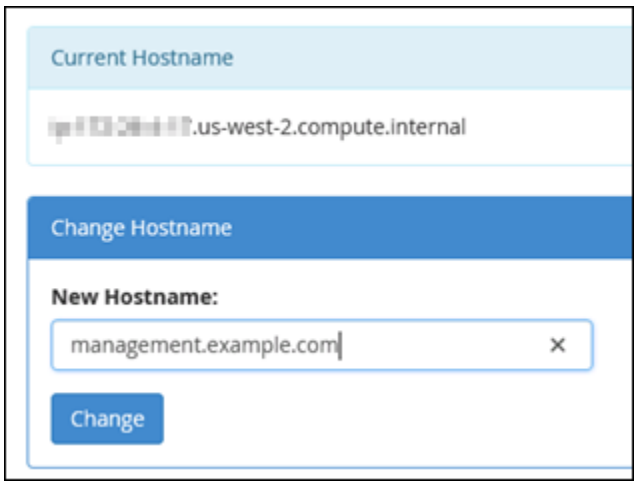


2. WHM コンソールの検索テキストボックスに Change hostname を入力して、検索結果の [ホスト名を変更] のオプションを選択します。

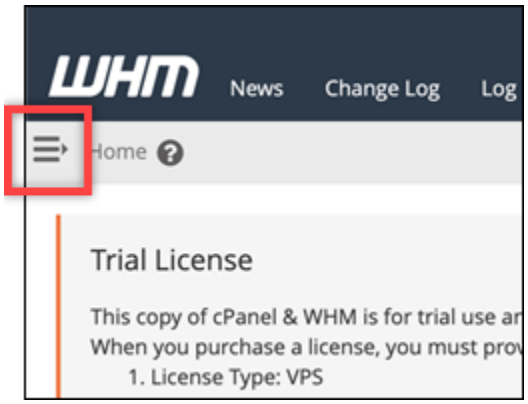


3. WHM コンソールへのアクセスに使用したいホスト名を、[新しいホスト名] テキストボックスに入力します。たとえば、management.example.com ないし administration.example.com を入力します。

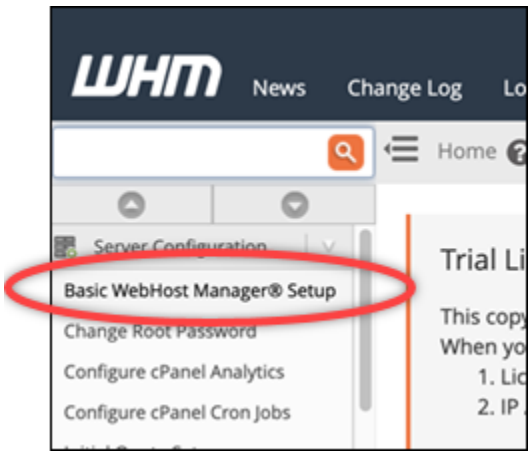
> **Note**
>
> サブドメインは、ホスト名としてのみ指定できます。whm や cpanel はサブドメインとして指定できません。



4. [Change] を選択します。
5. WHM コンソールの左上のセクションにあるナビゲーションメニューアイコンを選択します。

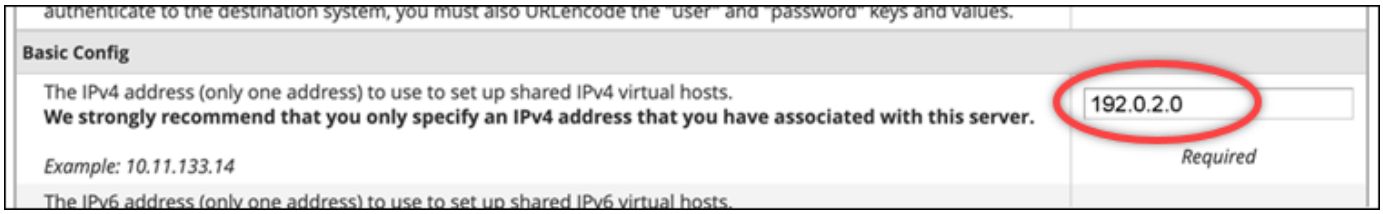


6. Basic WebHost Manager Setup を選択します。



7. [すべて] のタブで下にスクロールして、ページの [ベーシック Config] セクションを見つけます。

8. IPv4 アドレステキストボックスに、インスタンスの新しい静的 IP アドレスを入力します。IPv6 の情報については、「cPanel インスタンスで IPv6 の設定」を参照してください。



9. ページの最下部までスクロールして [保存] を選択します。

> **Note**
>
> [無効なライセンスファイル] のエラーメッセージを受けた場合は、しばらく待ってから再度 IP アドレスの変更を試みてください。

インスタンスのホスト名と IP アドレスは変更されましたが、まだ cPanel & WHM インスタンスにドメイン名をマッピングする必要があります。これは、登録済みドメイン名のドメインネームシステム (DNS) に、アドレス (A) レコードを追加することで可能です。A レコードは、インスタンスの静的 IP アドレスにインスタンスのホスト名を解決します。このガイドの次のセクションで、これを行う方法を解説します。

ステップ 5: cPanel & WHM インスタンスにドメイン名をマッピングする

> **Note**
>
> cPanel & WHM インスタンスにドメインをマッピングすることが可能で、これは WHM コンソールにアクセスする際に使用します。WHM 内で複数のドメインをマッピングすることも可能で、WHM 内のウェブサイトを管理する際に使用します。このセクションでは、cPanel & WHM インスタンスにドメインをマッピングする方法について説明します。新しいアカウントを作成する際、 WHM コンソール内に複数のドメインをマッピングしますが、この詳細については、WHMドキュメントの「新しいアカウントを作成」を参照してください。

`management.example.com` や `administration.example.com` などのドメイン名をインスタンスにマッピングするには、ドメインの DNS に A レコードを追加します。A レコードは、cPanel & WHM インスタンスのホスト名をインスタンスの静的 IP アドレスに マッピングします。A レコードで指定するサブドメインは、このガイドの「ステップ 4: cPanel & WHM インスタンスのホスト名と IP アドレスを変更する」のセクションで指定したホスト名と一致する必要があります。A レコード

が追加されたら、インスタンスの静的 IP アドレスを使用する代わりに、次のアドレスを使用してインスタンスの WHM コンソールにアクセスできます。*#InstanceHostName#* をインスタンスのホスト名に置き換えます。

```
https://<InstanceHostName>/whm
```

例:

```
https//management.example.com/whm
```

DNS レコードは、通常、ドメインの登録先であるレジストラが管理またはホストします。ただし、Lightsail コンソールを使用して管理できるように、ドメインの DNS レコードの管理を Lightsail に転送することをお勧めします。これを行うには、Lightsail コンソールにサインインします。Lightsail コンソールのホームページで、ドメインと DNS タブを選択し、DNS ゾーンの作成 を選択します。ページの手順に従って、ドメイン名を Lightsail に追加します。詳細については、「Lightsail でドメインの DNS レコードを管理する DNS ゾーンの作成」を参照してください。

ステップ 6: インスタンスのファイアウォールを編集する

次のファイアウォールポートはデフォルトで cPanel & WHM インスタンスで開いています。

- SSH - TCP - 22
- DNS (UDP) - UDP - 53
- DNS (TCP) - TCP - 53
- HTTP - TCP - 80
- HTTPS - TCP - 443
- カスタム - TCP - 2078
- カスタム - TCP - 2083
- カスタム - TCP - 2087
- カスタム - TCP - 2089

インスタンスで使用する予定のサービスやアプリケーションによっては、追加でポートを開く必要がある場合もあります。例えば、電子メールサービスの場合はポート 25、143、465、587、993、995、2096 を開き、カレンダーサービスの場合はポート 2080、2091 を開きます。インスタンス管理ページの [ネットワーク] タブで、ページのファイアウォールのセ

クションまでスクロールして [追加ルール] を選択します。アプリケーション、プロトコル、そしてポートまたは開けるポート範囲を選択します。完了したら、[作成] を選択します。

開くべきポートの詳細については、cPanel ドキュメント の「cPanel サービスのファイアウォールを設定する方法」を参照してください。Lightsail でインスタンスのファイアウォールを編集する方法の詳細については、「Amazon Lightsail でのインスタンスのファイアウォールルールの追加と編集」を参照してください。

ステップ 7: Lightsail インスタンスから SMTP 制限を削除する

AWS は、すべての Lightsail インスタンスのポート 25 のアウトバウンドトラフィックをブロックします。ポート 25 でアウトバウンドトラフィックを送信するには、この制限の解除をリクエストします。詳細については、「Lightsail インスタンスからポート 25 の制限を解除するにはどうすればよいですか？」を参照してください。

> **Important**
>
> ポート 25、465、または 587 を使用するように SMTP を設定する場合は、Lightsail コンソールのインスタンスのファイアウォールでそれらのポートを開く必要があります。詳細については、「Amazon Lightsail でのインスタンスファイアウォールルールの追加と編集」を参照してください。

ステップ 8: cPanel および WHM のドキュメントを読み込んでサポートを受ける

cPanel & WHM のドキュメントを読んで、cPanel と WHM を使ってウェブサイトを管理する方法を確認ください。詳細については、cPanel & WHM ドキュメント を参照してください。

cPanel & WHM についての質問がある場合やサポートが必要な際は、次のリソースを使用して cPanel にお問い合わせ頂けます。

- インストールの cPanel トラブルシューティング
- cPanel ディスコード チャンネル

ステップ 9: cPanel および WHM のライセンスの購入

cPanel & WHM インスタンスには、15日間のトライアルライセンスが含まれています。15 日後も継続して cPanel & WHM を使用するには cPanel からライセンスを購入する必要があります。詳細については、cPanel ドキュメントの「cPanel のライセンスを購入する方法」を参照してください。

> **Important**
>
> cPanel からライセンスを購入する際は、cPanel & WHM インスタンスのパブリック IP アドレスを指定する必要があります。購入したライセンスは、その IP アドレスに関連付けられます。そのため、このガイドの「ステップ 2: cPanel & WHM インスタンスに静的 IP アドレスをアタッチする」のセクションで解説されているように、cPanel & WHM インスタンスに静的 IP をアタッチする必要があります。cPanel からライセンスを購入する際に静的 IP を指定し、Lightsail インスタンスで cPanel & WHM ライセンスを使用する予定がある限り、静的 IP を保持します。後に別の IP アドレスにライセンスを移転する必要がある場合は、cPanel にリクエストを送信することができます。詳細については、WHM ドキュメント の「ライセンスの移転」を参照してください。

ステップ 10: cPanel および WHM のインスタンスのスナップショットを作成する

スナップショットは、インスタンスのシステムディスクおよびオリジナル設定のコピーです。スナップショットには、インスタンスの復元に必要なすべてのデータ (スナップショットが作成された時点のデータ) が含まれます。スナップショットを、新しいインスタンスのベースラインまたはデータのバックアップとして使用できます。手動スナップショットはいつでも作成できます。また自動スナップショットを有効にすると、 Lightsail に毎日スナップショットを自動的に作成させることが可能です。

> **Note**
>
> - の現世代の設計図 cPanel & WHM の AlmaLinuxインスタンススナップショットは、Amazon EC2 にエクスポートできます。
> - 前世代のブループリントである cPanel & WHM for AlmaLinux のインスタンススナップショットは、現時点では Amazon EC2 にエクスポートできません。
> - スナップショットから新しいインスタンスを作成する場合、ステップ 3 で説明したように、インスタンスが完全に起動するまでしばらく待ってから WHM にサインインしてください。

インスタンス管理ページの [スナップショット] タブで、スナップショット名を入力して [スナップショットの作成] を選択します。または、ページの [自動スナップショット] セクションまでスクロールして、トグルで選択して自動スナップショットを有効にします。

詳細については、「Linux または Unix インスタンスのスナップショットを作成する」および「Amazon Lightsail でインスタンスまたはディスクの自動スナップショットを有効または無効にする」を参照してください。

クイックスタートガイド: Drupal

Amazon Lightsail で起動した Drupal インスタンスの使用を開始するステップについて説明します。

目次

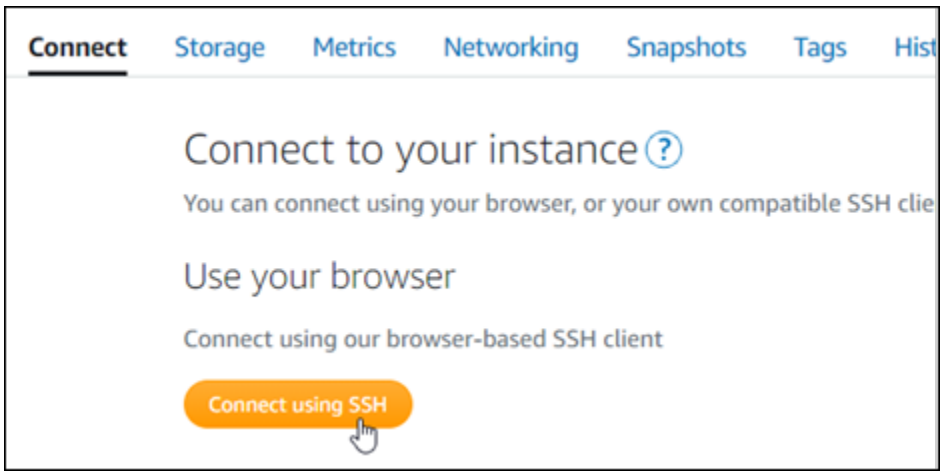
ステップ 1: Bitnami のドキュメントを確認する

Drupal アプリケーションの設定方法については、Bitnami ドキュメントを参照してください。詳細については、「AWS クラウド 用に Bitnami がパッケージ化した Drupal」を参照してください。

ステップ 2: Drupal の管理ダッシュボードにアクセスするため、デフォルトのアプリケーションパスワードを取得する

次の手順を完了して、Drupal ウェブサイトの管理ダッシュボードにアクセスする際に必要となるデフォルトのアプリケーションパスワードを取得します。詳細については、「Amazon LightsailのBitnami インスタンス向けにアプリケーションのユーザー名とパスワードを取得する」を参照してください。

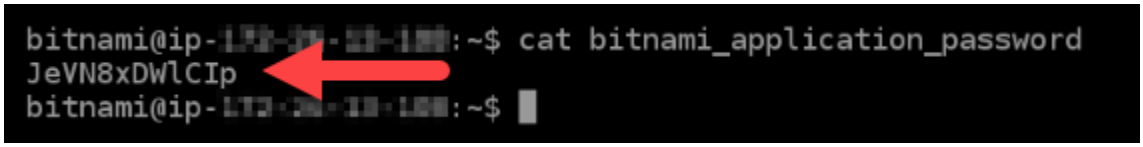
1. インスタンス管理ページの [接続] タブで、[SSH を使用して接続] を選択します。



2. 接続後に、次のコマンドを入力してアプリケーションのパスワードを取得します。

```
cat $HOME/bitnami_application_password
```

アプリケーションのデフォルトパスワードを含んだ、次の例のようなレスポンスが表示されます。



ステップ 3: インスタンスに静的 IP アドレスをアタッチする

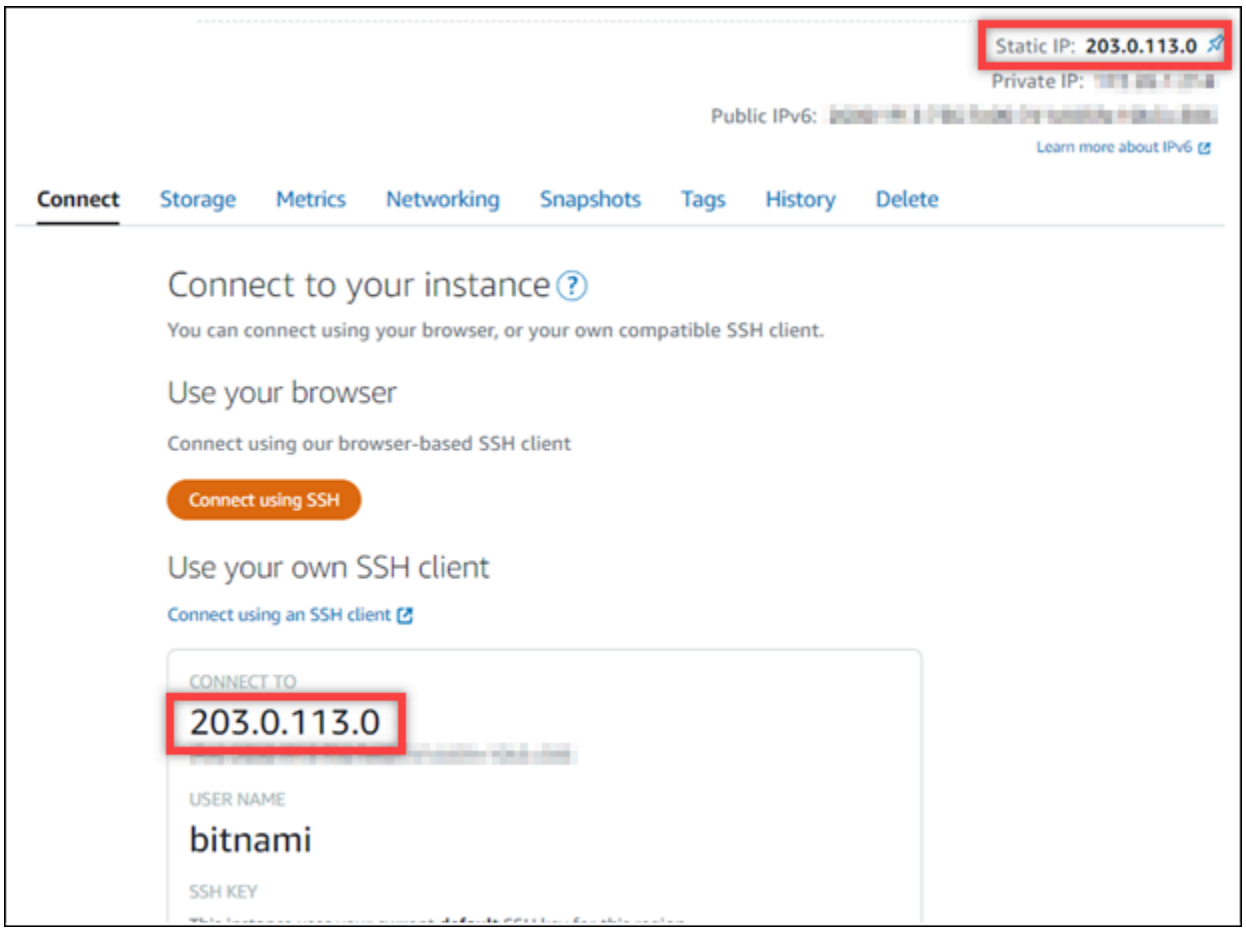
インスタンスを最初に作成した際に割り当てられたパブリック IP アドレスは、インスタンスを停止してスタートするたびに変更されます。パブリック IP アドレスが変更されないように、静的 IP アドレスを作成してインスタンスにアタッチする必要があります。それ以降、example.com などの登録したドメイン名をインスタンスで使用する際、毎回インスタンスを停止してスタートするたびにドメインの DNS レコードを更新する必要がなくなります。1 つの静的 IP を 1 つのインスタンスにアタッチできます。

インスタンス管理ページの [ネットワーク] タブで、[静的 IP の作成] または [静的 IP のアタッチ] (インスタンスにアタッチできる静的 IP を既に作成している場合) を選択して、ページの手順に従います。詳細については、「静的 IP を作成してインスタンスにアタッチする」を参照してください。

ステップ 4: Drupal ウェブサイトの管理ダッシュボードにサインインする

デフォルトのユーザーパスワードを取得したら、Drupal ウェブサイトのホームページに移動し、管理ダッシュボードにサインインします。サインイン後に、ウェブサイトをカスタマイズしたり管理上の変更を行うことができます。Drupal で実行できる事項の詳細については、本ガイドの後半にある「ステップ 7: Drupal のドキュメントを読み、引き続きウェブサイトの設定を続行する」のセクションを参照してください。

1. インスタンス管理ページの [Connect] (接続) タブにあるパブリック IP アドレスを書き留めます。パブリック IP アドレスは、インスタンス管理ページのヘッダーセクションにも表示されます。



2. インスタンスのパブリック IP アドレスを参照します (例: `http://203.0.113.0` に移動します)。

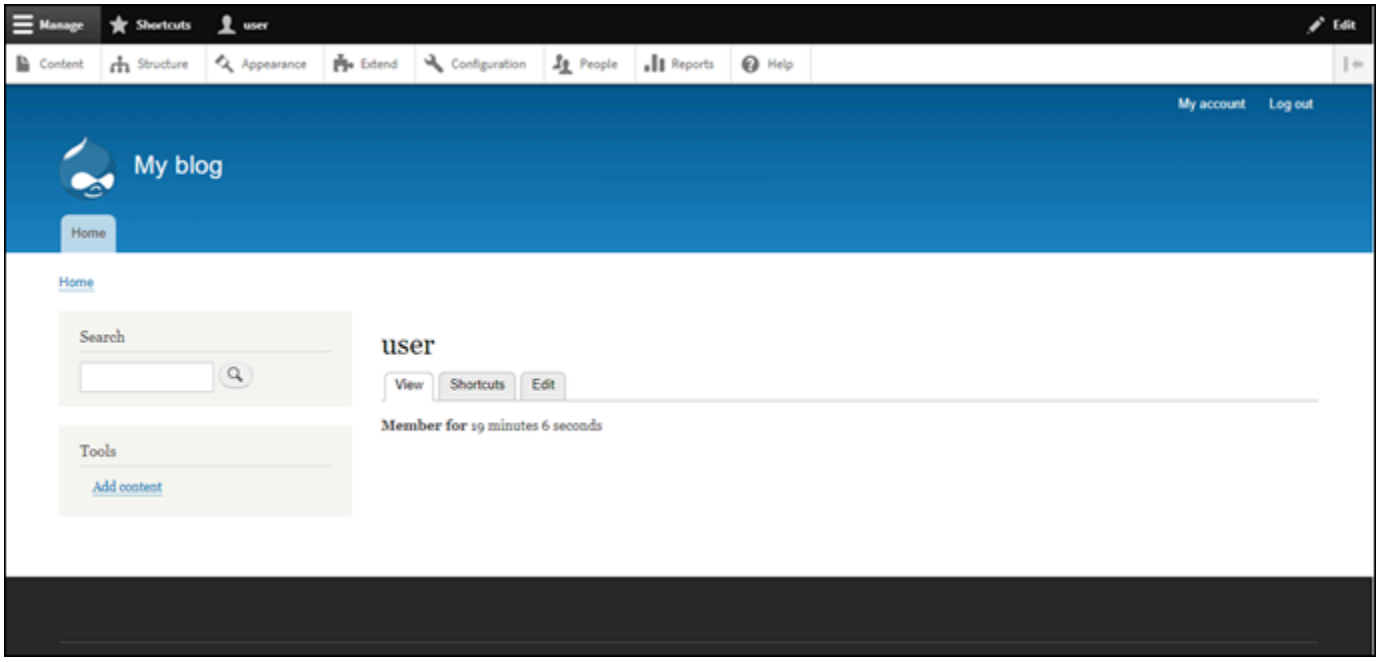
   Drupal ウェブサイトのホームページが表示されます。

3. Drupal ウェブサイトのホームページで、右下にある [Manage] (管理) を選択します。

   [Manage] (管理) バナーが表示されない場合は、`http://`*`<PublicIP>`*`/user/login` を参照することでサインインページにアクセスすることができます。*`<PublicIP>`* を、インスタンスのパブリック IP アドレスに置き換えます。

4. デフォルトのユーザー名 (`user`) と、先ほど取得したデフォルトのパスワードを使用してサインインします。

   Drupal の管理ダッシュボードが表示されます。



ステップ 5: 登録済みドメイン名へのトラフィックを Drupal ウェブサイトに送信する

`example.com` などの登録済みドメイン名のトラフィックを Drupal ウェブサイトに送信するには、ドメインのドメインネームシステム (DNS) にレコードを追加します。DNS レコードは、通常、ドメインの登録先であるレジストラが管理またはホストします。ただし、ドメインの DNS レコードの管理を Lightsail に引き渡して、Lightsail コンソールで管理できるようにすることをお勧めします。

Lightsail コンソールのホームページの [Domains & DNS] (ドメインと DNS) タブで、[Create DNS zone] (DNS ゾーンの作成) を選択し、ページに記載される手順に従います。詳細については、「Lightsail で DNS ゾーンを作成し、ドメインの DNS レコードを管理する」を参照してください。

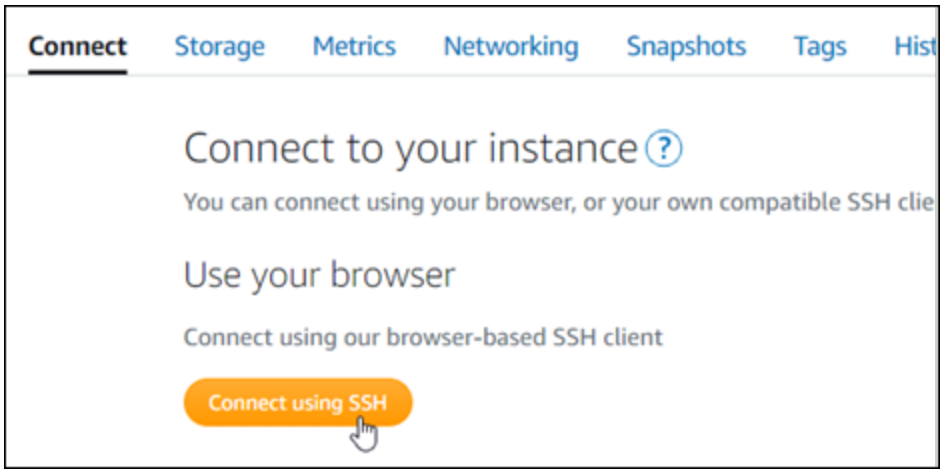
インスタンスに設定したドメイン名を参照すると、Drupal ウェブサイトのホームページへと移動します。次に、SSL/TLS 証明書を生成して設定し、Drupal ウェブサイトの HTTPS 接続を有効にします。詳細については、本ガイドの次の「ステップ 6: Drupal ウェブサイトの HTTPS を設定する」のセクションを参照してください。

ステップ 6: Drupal ウェブサイトの HTTPS を設定する

Drupal ウェブサイトで HTTPS を設定するには、以下の手順を実行します。次の手順では、Bitnami HTTPS 設定ツール (`bncert-tool`) の使い方を説明しています。これは、Let's Encrypt SSL/TLS 証明書を要求するコマンドラインツールです。詳細については、Bitnami ドキュメントの「Bitnami 設定ツールの詳細を確認する」を参照してください。

> **Important**
>
> この手順を開始する前に、Drupal インスタンスにトラフィックがルーティングされるようにドメインが設定済みであることを確認してください。設定されていない場合、SSL/TLS 証明書の検証プロセスが失敗します。

1. インスタンス管理ページの [接続] タブで、[SSH を使用して接続] を選択します。



2. 接続されたら、以下のコマンドを入力し、インスタンスに bncert ツールがインストールされていることを確認します。

```
sudo /opt/bitnami/bncert-tool
```

   以下のレスポンスのいずれかが表示されます。

   - レスポンスにコマンドが見つからないと表示された場合、bncert ツールがインスタンスにインストールされていないことを示しています。この手順の次のステップに進み、bncert ツールをインスタンスにインストールします。
   - レスポンスに「Welcome to the Bitnami HTTPS configuration tool (Bitnami HTTPS 設定ツールへようこそ)」と表示された場合は、インスタンスに bncert ツールがインストールされています。この手順のステップ 8 に進んでください。
   - bncert ツールがインスタンスにインストールされてからしばらく経っている場合、ツールのアップデートバージョンが利用可能であることを示すメッセージが表示されることがあります。ダウンロードすることを選択し、`sudo /opt/bitnami/bncert-tool` コマンドを入力して bncert ツールを再度実行してください。この手順のステップ 8 に進んでください。

3. 以下のコマンドを入力して、bncert の実行ファイルをインスタンスにダウンロードします。

```
wget -O bncert-linux-x64.run https://downloads.bitnami.com/files/bncert/latest/
bncert-linux-x64.run
```

4. 以下のコマンドを入力して、インスタンスに bncert ツールの実行ファイル用のディレクトリを作成します。

```
sudo mkdir /opt/bitnami/bncert
```

5. 以下のコマンドを入力して、プログラムとして実行できるファイルを bncert に実行させます。

```
sudo chmod +x /opt/bitnami/bncert/bncert-linux-x64.run
```

6. 次のコマンドを入力して、sudo /opt/bitnami/bncert-tool コマンドを入力すると bncert ツールを実行するシンボリックリンクを作成します。

```
sudo ln -s /opt/bitnami/bncert/bncert-linux-x64.run /opt/bitnami/bncert-tool
```

これでインスタンスに bncert ツールをインストールする手順は完了です。

7. 次のコマンドを入力して、bncert ツールを実行しましょう。

```
sudo /opt/bitnami/bncert-tool
```

8. 次の例に示すように、プライマリドメイン名と代替ドメイン名の間はスペースで区切って入力します。

ドメインがインスタンスのパブリック IP アドレスにトラフィックをルーティングするように設定されていない場合、bncert ツールは、続行する前にその設定を行うように要求します。ドメインは、bncert ツールを使用して HTTPS を有効にしているインスタンスでのパブリック IP アドレスにトラフィックをルーティングする必要があります。これはドメインを所有していることを確認し、証明書の検証として機能します。

```
------------------------------------------------------------------------------
Welcome to the Bitnami HTTPS Configuration tool.

------------------------------------------------------------------------------
Domains

Please provide a valid space-separated list of domains for which you wish to
configure your web server.

Domain list []: example.com www.example.com
```

9. bncert ツールは、ウェブサイトのリダイレクトの設定方法を尋ねます。使用できるオプションは次のとおりです。

   - HTTP から HTTPS へのリダイレクトを有効にする - HTTP バージョンのウェブサイトを閲覧するユーザー (例: http:/example.com) を自動的に HTTPS バージョン (例: https://example.com) にリダイレクトするかどうかを決定します。すべての訪問者が暗号化された接続を使用するように強制されるため、このオプションを有効にすることをお勧めします。Y を入力して Enter を押すと、有効になります。
   - www なしから www ありへのリダイレクトの有効化 - ドメインの頂点 (例: https://example.com) まで閲覧するユーザー を自動的にドメインの www サブドメイン (例: https://www.example.com) にリダイレクトするかを指定します。このオプションを有効にすることをお勧めします。ただし、ドメインの頂点を Google のウェブマスターツールなどの検索エンジンツールで希望のウェブサイトアドレスとして指定した場合、または頂点が IP を直接指しており、www のサブドメインが CNAME レコードを介してリファレンスしている場合は、無効にして代替オプションを有効にすることをお勧めします (www ありから www なしへのリダイレクトを有効化) 。Y を入力し、Enter を押して有効にします。
   - www ありから www なしへのリダイレクトを有効にする - ドメインの www サブドメイン (例: https://www.example.com) まで閲覧するユーザーを、自動的にドメインの頂点 (例: https://example.com) にリダイレクトするかを指定します。www なしから www ありへのリダイレクトを有効にした場合は、これを無効にすることをお勧めします。N を入力し、Enter を押して無効にします。

   選択した結果は次の例のようになります。

```
Enable/disable redirections

Please select the redirections you wish to enable or disable on your Bitnami
installation.



Enable HTTP to HTTPS redirection [Y/n]: Y



Enable non-www to www redirection [Y/n]: Y



Enable www to non-www redirection [y/N]: N
```

10. これから実行される変更が一覧表示されます。Y と入力し、Enter を押して確認し、続行します。

```
Changes to perform

The following changes will be performed to your Bitnami installation:

1. Stop web server
2. Configure web server to use a free Let's Encrypt certificate for the domains:
example.com www.example.com
3. Configure a cron job to automatically renew the certificate each month
4. Configure web server name to: example.com
5. Enable HTTP to HTTPS redirection (example: redirect http://example.com to
https://example.com)
6. Enable non-www to www redirection (example: redirect example.com to
www.example.com)
7. Start web server once all changes have been performed



Do you agree to these changes? [Y/n]: Y
```

11. Let's Encrypt 証明書に関連付けるメールアドレスを入力し、Enter を押します。

```
Create a free HTTPS certificate with Let's Encrypt

Please provide a valid e-mail address for which to associate your Let's Encrypt
certificate.

Domain list: example.com www.example.com

Server name: example.com

E-mail address []:
```

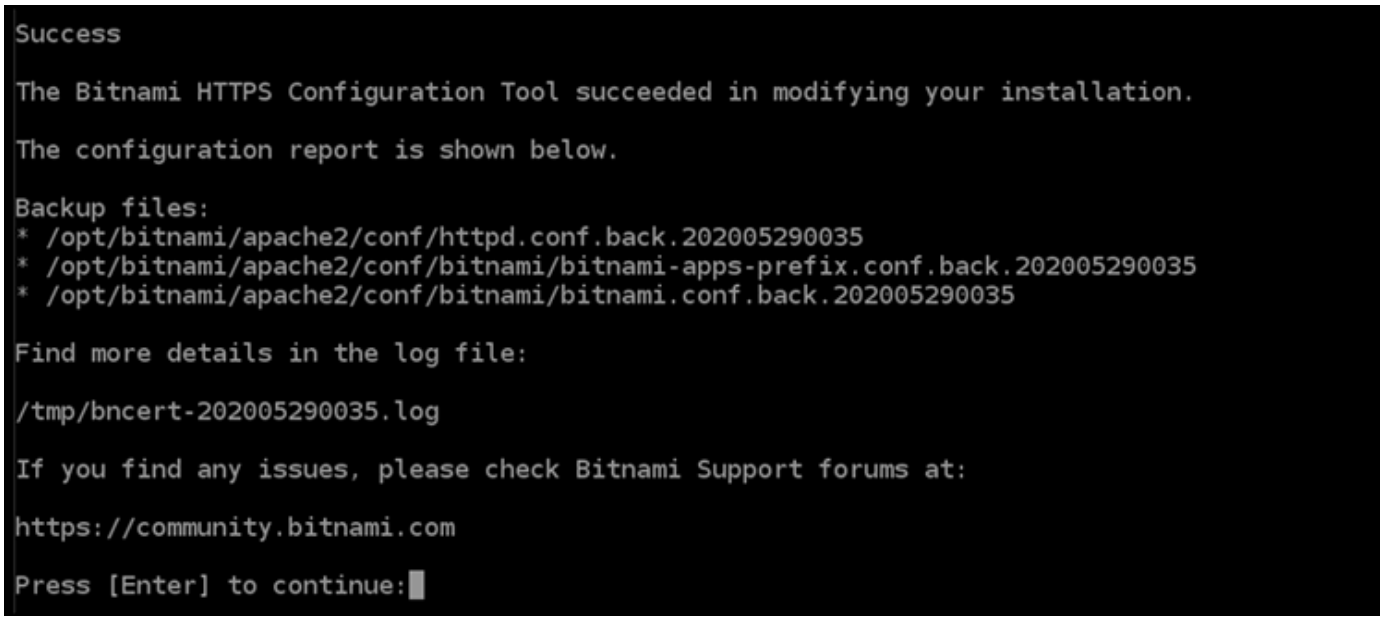
12. Let's Encrypt サブスクライバー合意書を確認します。Y と入力し、Enter を押して契約に同意し、続行します。

```
The Let's Encrypt Subscriber Agreement can be found at:

https://letsencrypt.org/documents/LE-SA-v1.2-November-15-2017.pdf

Do you agree to the Let's Encrypt Subscriber Agreement? [Y/n]:
```

これらのアクションは、証明書のリクエストや指定したリダイレクトの設定など、インスタンスで HTTPS を有効にするために実行されます。

```
Performing changes to your installation

The Bitnami HTTPS Configuration Tool will perform any necessary actions to your
Bitnami installation. This may take some time, please be patient.

|
```

次の例のようなメッセージが表示された場合は、証明書は正常に発行され、検証され、インスタンスでリダイレクトが正常に設定されています。



bncert ツールは、有効期限が切れる前、 80 日ごとに証明書の自動更新を実行します。インスタンスで追加のドメインやサブドメインを使用し、それらのドメインで HTTPS を有効にする場合は、上記のステップを繰り返します。

これで、Drupal インスタンスでの HTTPS の有効化が完了しました。次回に、設定したドメインを使用して Drupal ウェブサイトを参照するときには、HTTPS 接続にリダイレクトされるはずです。

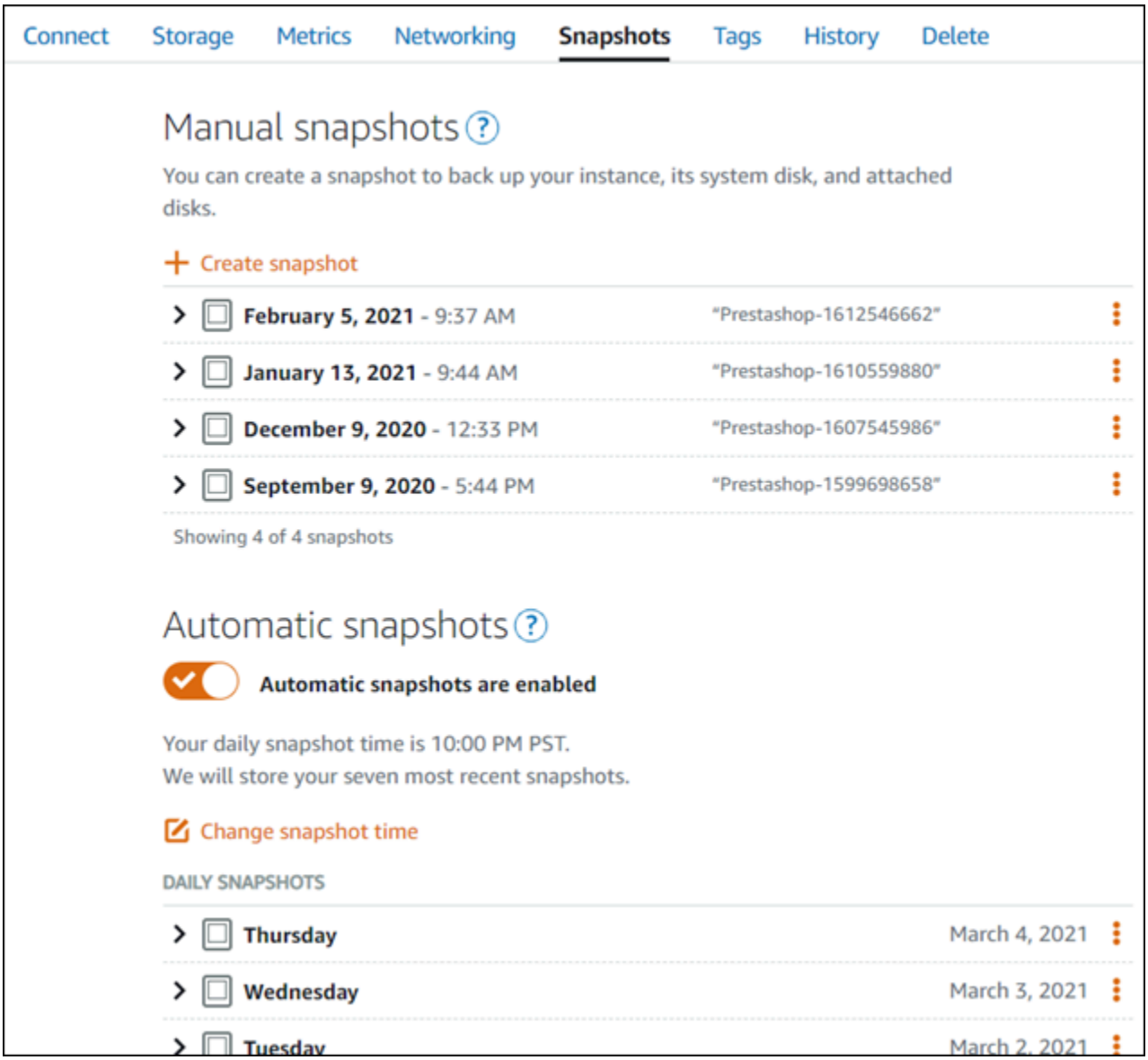
ステップ 7: Drupal のドキュメントを読み、引き続きウェブサイトの設定を続行する

Drupal のドキュメントを読み、ウェブサイトを管理およびカスタマイズする方法を確認します。詳細については、「Drupal Documentation」(Drupal ドキュメント) を参照してください。

ステップ 8: インスタンスのスナップショットを作成する

Drupal ウェブサイトを希望どおりに設定したら、インスタンスの定期的なスナップショットを作成してバックアップします。スナップショットは手動で作成するか、自動スナップショットを有効にして Lightsail に毎日のスナップショットを作成させることができます。インスタンスに問題が発生した場合は、スナップショットを使用して新しい代替インスタンスを作成できます。詳細については、「スナップショット」を参照してください。

インスタンス管理ページの [スナップショット] タブで [スナップショットを作成する] を選択するか、[自動スナップショットを有効にする] を選択します。



詳細については、「Amazon Lightsail で Linux または Unix インスタンスのスナップショットを作成」および「Amazon Lightsail のインスタンスまたはディスクの自動スナップショットの有効化と無効化」を参照してください。

クイックスタートガイド: Ghost

Amazon Lightsail で起動した Ghost インスタンスの使用を開始するステップについて説明します。

目次

- ステップ 1: Bitnami のドキュメントを確認する
- ステップ 2: Ghost の管理ダッシュボードにアクセスするため、デフォルトのアプリケーションパスワードを取得する
- ステップ 3: インスタンスに静的 IP アドレスをアタッチする
- ステップ 4: Ghost ウェブサイトの管理ダッシュボードにサインインする
- ステップ 5: 登録済みドメイン名へのトラフィックを Ghost ウェブサイトに送信する
- ステップ 6: Ghost ウェブサイトの HTTPS を設定する
- ステップ 7: Ghost のドキュメントを読み、引き続きウェブサイトの設定を続行する
- ステップ 8: インスタンスのスナップショットを作成する

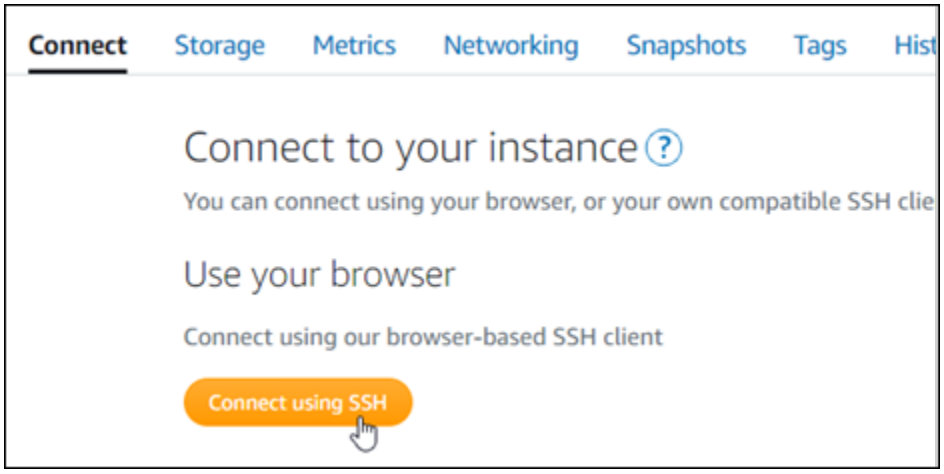
ステップ 1: Bitnami のドキュメントを確認する

Ghost アプリケーションの設定方法については、Bitnami ドキュメントを参照してください。詳細については、「AWS クラウド 用に Bitnami がパッケージ化した Ghost」を参照してください。

ステップ 2: Ghost の管理ダッシュボードにアクセスするため、デフォルトのアプリケーションパスワードを取得する

次の手順を完了して、Ghost ウェブサイトの管理ダッシュボードにアクセスする際に必要となるデフォルトのアプリケーションパスワードを取得します。詳細については、「Amazon LightsailのBitnami インスタンス向けにアプリケーションのユーザー名とパスワードを取得する」を参照してください。

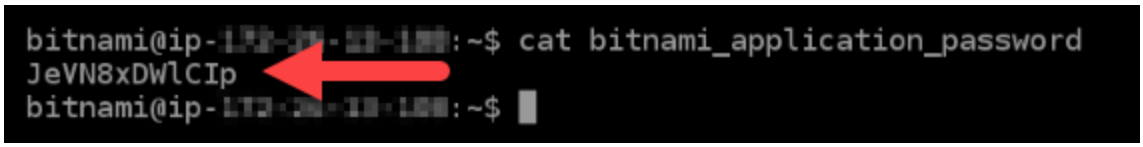
1. インスタンス管理ページの [接続] タブで、[SSH を使用して接続] を選択します。



2. 接続後に、次のコマンドを入力してアプリケーションのパスワードを取得します。

```
cat $HOME/bitnami_application_password
```

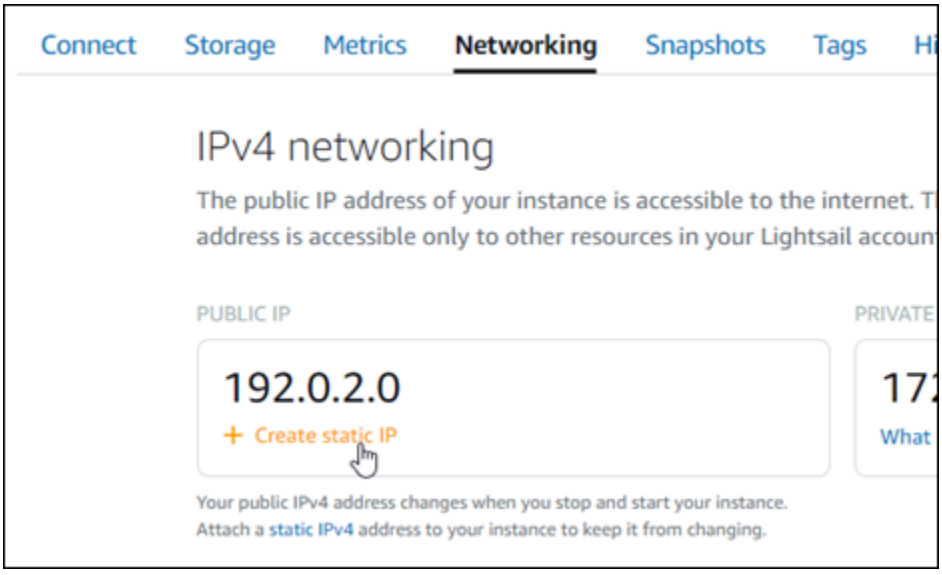
アプリケーションのデフォルトパスワードを含んだ、次の例のようなレスポンスが表示されます。



ステップ 3: インスタンスに静的 IP アドレスをアタッチする

インスタンスを最初に作成した際に割り当てられたパブリック IP アドレスは、インスタンスを停止してスタートするたびに変更されます。パブリック IP アドレスが変更されないように、静的 IP アドレスを作成してインスタンスにアタッチする必要があります。それ以降、example.com などの登録したドメイン名をインスタンスで使用する際、毎回インスタンスを停止してスタートするたびにドメインの DNS レコードを更新する必要がなくなります。1 つの静的 IP を 1 つのインスタンスにアタッチできます。

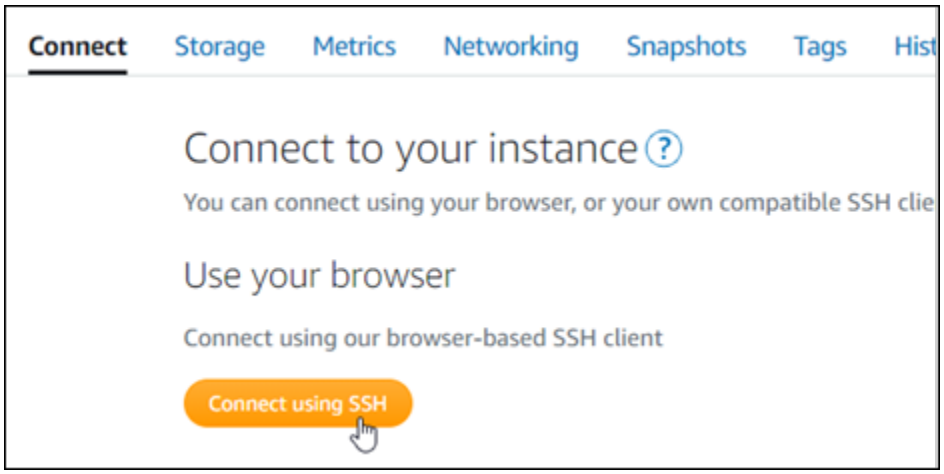
インスタンス管理ページの [ネットワーク] タブで、[静的 IP の作成] または [静的 IP のアタッチ] (インスタンスにアタッチできる静的 IP を既に作成している場合) を選択して、ページの手順に従います。詳細については、「静的 IP を作成してインスタンスにアタッチする」を参照してください。



新しい静的 IP アドレスがインスタンスに添付されたら、次の手順を実行して、アプリケーションに新しい静的 IP アドレスを認識させる必要があります。

1. インスタンスの静的 IP アドレスは書き留めておきます。このIP アドレスはインスタンス管理ページの ヘッダーセクションに表示されます。

2. インスタンス管理ページの [Connect] (接続) タブで、[SSH を使用して接続] を選択します。



3. 接続後に、次のコマンドを入力します。*<StaticIP>* をインスタンスの新しい静的 IP アドレスに置き換えます。

```
sudo /opt/bitnami/configure_app_domain --domain <StaticIP>
```

例:

```
sudo /opt/bitnami/configure_app_domain --domain 203.0.113.0
```

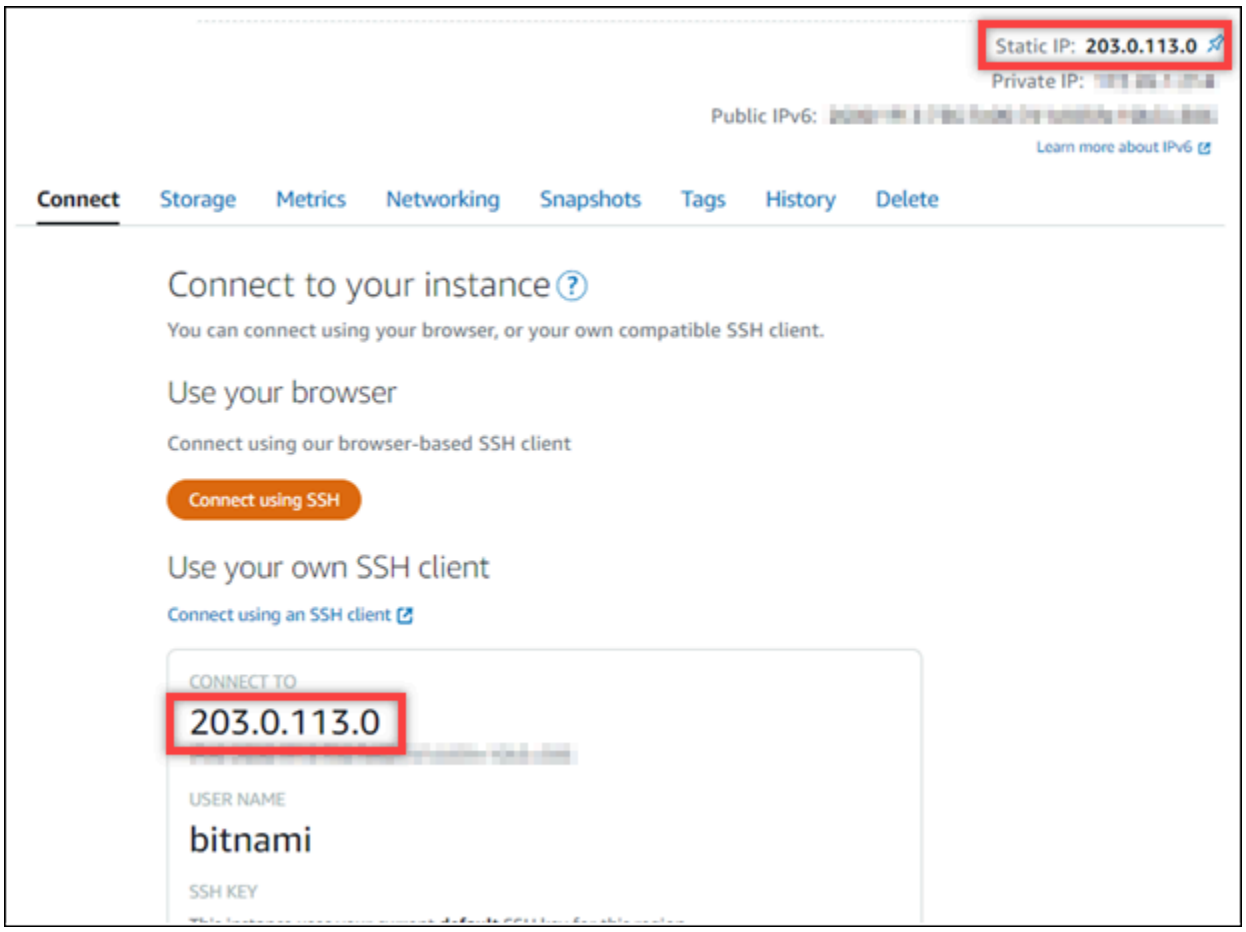
次の例のようなレスポンスが表示されます。これで、インスタンス上のアプリケーションが新しい静的 IP アドレスを認識するようになります。

```
bitnami@ip-[redacted]:~$ sudo /opt/bitnami/configure_app_domain --domain 203.0.113.0
Configuring domain to 203.0.113.0
2021-03-12T15:49:22.000Z - info: Saving configuration info to disk
prestashop 15:49:22.41 INFO  ==> Trying to connect to the database server
prestashop 15:49:22.44 INFO  ==> Updating hostname in database
prestashop 15:49:22.46 INFO  ==> Purging cache
Disabling automatic domain update for IP address changes
```

ステップ 4: Ghost ウェブサイトの管理ダッシュボードにサインインする

デフォルトのユーザーパスワードを取得したら、以下の手順に従って Ghost ウェブサイトのホームページに移動し、管理ダッシュボードにサインインします。サインイン後に、ウェブサイトをカスタマイズしたり管理上の変更を行うことができます。Ghost で実行できる事項の詳細については、本ガイドの後半にある「ステップ 6: Ghost のドキュメントを読み、引き続きウェブサイトの設定を続行する」のセクションを参照してください。

1. インスタンス管理ページの [Connect] (接続) タブにあるパブリック IP アドレスを書き留めます。パブリック IP アドレスは、インスタンス管理ページのヘッダーセクションにも表示されます。



2. インスタンスのパブリック IP アドレスを参照します (例: `http://203.0.113.0` に移動します)。

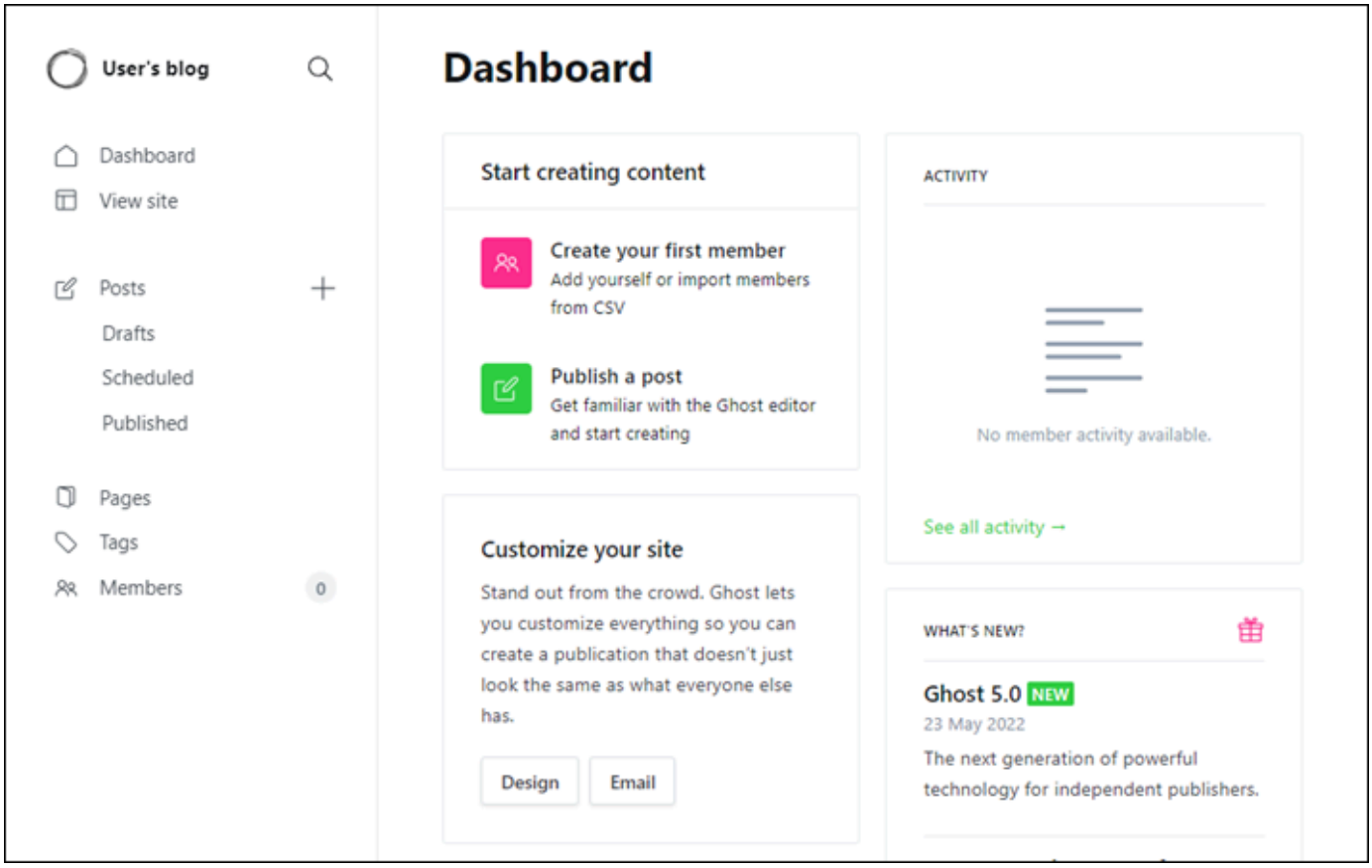
   Ghost ウェブサイトのホームページが表示されます。

3. Ghost ウェブサイトのホームページで、右下にある [Manage] (管理) を選択します。

   [Manage] (管理) バナーが表示されない場合は、`http://`*`<PublicIP>`*`/ghost` を参照することでサインインページにアクセスすることができます。*`<PublicIP>`* を、インスタンスのパブリック IP アドレスに置き換えます。

4. デフォルトのユーザー名 (`user@example.com`) と、先ほど取得したデフォルトのパスワードを使用してサインインします。

   Ghost の管理ダッシュボードが表示されます。



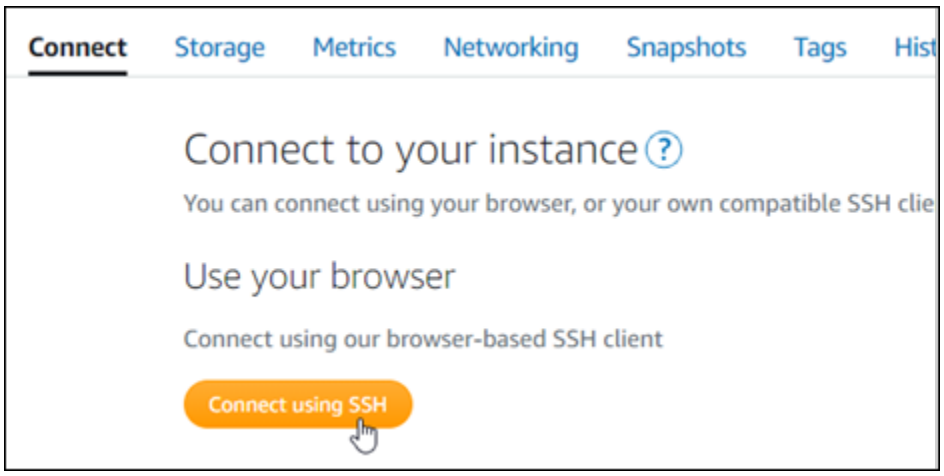
ステップ 5: 登録済みドメイン名へのトラフィックを Ghost ウェブサイトに送信する

`example.com` などの登録済みドメイン名のトラフィックを Ghost ウェブサイトに送信するには、ドメインの DNS にレコードを追加します。DNS レコードは、通常、ドメインの登録先であるレジストラが管理またはホストします。ただし、ドメインの DNS レコードの管理を Lightsail に引き渡して、Lightsail コンソールで管理できるようにすることをお勧めします。

Lightsail コンソールのホームページの [Domains & DNS] (ドメインと DNS) タブで、[Create DNS zone] (DNS ゾーンの作成) を選択し、ページに記載される手順に従います。詳細については、「Lightsail で DNS ゾーンを作成し、ドメインの DNS レコードを管理する」を参照してください。

ドメイン名へのトラフィックがインスタンスにルーティングされたら、次の手順を実行して、Ghost アプリケーションにドメイン名を認識させます。

1. インスタンス管理ページの [Connect] (接続) タブで、[SSH を使用して接続] を選択します。



2. 接続後に、次のコマンドを入力します。*<DomainName>* は、Ghost インスタンスにトラフィックをルーティングするドメイン名に置き換えてください。

```
sudo /opt/bitnami/configure_app_domain --domain <DomainName>
```

例:

```
sudo /opt/bitnami/configure_app_domain --domain example.com
```

次の例のようなレスポンスが表示されます。これで、Ghost アプリケーションがドメインを認識するようになりました。

```
bitnami@ip-[redacted]:~$ sudo /opt/bitnami/configure_app_domain --domain example.com
Configuring domain to example.com
2022-06-09T22:25:58.177Z - info: Saving configuration info to disk
ghost 22:25:58.57 INFO  ==> Configuring Ghost URL to http://example.com
Disabling automatic domain update for IP address changes
```

インスタンスに設定したドメイン名を参照すると、Ghost ウェブサイトのホームページへと移動します。次に、SSL/TLS 証明書を生成して設定し、Ghost ウェブサイトの HTTPS 接続を有効にします。詳細については、本ガイドの次の「ステップ 6: Ghost ウェブサイトの HTTPS を設定する」のセクションを参照してください。

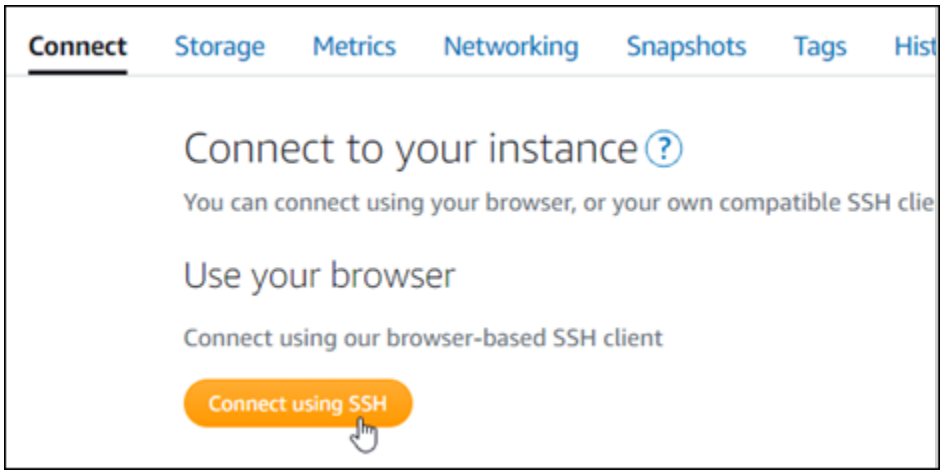
ステップ 6: Ghost ウェブサイトの HTTPS を設定する

Ghost ウェブサイトで HTTPS を設定するには、以下の手順を実行します。次の手順では、Bitnami HTTPS 設定ツール (`bncert-tool`) の使い方を説明しています。これは、Let's Encrypt SSL/TLS 証

明書を要求するコマンドラインツールです。詳細については、Bitnami ドキュメントの「Bitnami 設定ツールの詳細を確認する」を参照してください。

> **Important**
>
> この手順を開始する前に、Ghost インスタンスにトラフィックがルーティングされるようにドメインが設定済みであることを確認してください。設定されていない場合、SSL/TLS 証明書の検証プロセスが失敗します。

1. インスタンス管理ページの [接続] タブで、[SSH を使用して接続] を選択します。



2. 接続されたら、以下のコマンドを入力し、インスタンスに bncert ツールがインストールされていることを確認します。

```
sudo /opt/bitnami/bncert-tool
```

以下のレスポンスのいずれかが表示されます。

- レスポンスにコマンドが見つからないと表示された場合、bncert ツールがインスタンスにインストールされていないことを示しています。この手順の次のステップに進み、bncert ツールをインスタンスにインストールします。
- レスポンスに「Welcome to the Bitnami HTTPS configuration tool (Bitnami HTTPS 設定ツールへようこそ)」と表示された場合は、インスタンスに bncert ツールがインストールされています。この手順のステップ 8 に進んでください。
- bncert ツールがインスタンスにインストールされてからしばらく経っている場合、ツールのアップデートバージョンが利用可能であることを示すメッセージが表示されることがありま

す。ダウンロードすることを選択し、sudo /opt/bitnami/bncert-tool コマンドを入力して bncert ツールを再度実行してください。この手順のステップ 8 に進んでください。

3. 以下のコマンドを入力して、bncert の実行ファイルをインスタンスにダウンロードします。

```
wget -O bncert-linux-x64.run https://downloads.bitnami.com/files/bncert/latest/
bncert-linux-x64.run
```

4. 以下のコマンドを入力して、インスタンスに bncert ツールの実行ファイル用のディレクトリを作成します。

```
sudo mkdir /opt/bitnami/bncert
```

5. 以下のコマンドを入力して、プログラムとして実行できるファイルを bncert に実行させます。

```
sudo chmod +x /opt/bitnami/bncert/bncert-linux-x64.run
```

6. 次のコマンドを入力して、sudo /opt/bitnami/bncert-tool コマンドを入力すると bncert ツールを実行するシンボリックリンクを作成します。

```
sudo ln -s /opt/bitnami/bncert/bncert-linux-x64.run /opt/bitnami/bncert-tool
```

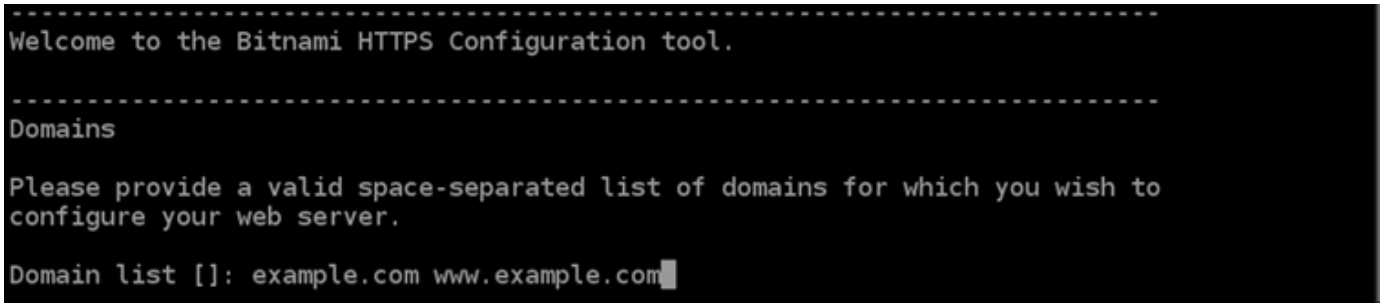
これでインスタンスに bncert ツールをインストールする手順は完了です。

7. 次のコマンドを入力して、bncert ツールを実行しましょう。

```
sudo /opt/bitnami/bncert-tool
```

8. 次の例に示すように、プライマリドメイン名と代替ドメイン名の間はスペースで区切って入力します。

ドメインがインスタンスのパブリック IP アドレスにトラフィックをルーティングするように設定されていない場合、bncert ツールは、続行する前にその設定を行うように要求します。ドメインは、bncert ツールを使用して HTTPS を有効にしているインスタンスでのパブリック IP アドレスにトラフィックをルーティングする必要があります。これはドメインを所有していることを確認し、証明書の検証として機能します。



9. bncert ツールは、ウェブサイトのリダイレクトの設定方法を尋ねます。使用できるオプションは次のとおりです。

   - HTTP から HTTPS へのリダイレクトを有効にする - HTTP バージョンのウェブサイトを閲覧するユーザー (例: `http:/example.com`) を自動的に HTTPS バージョン (例: `https://example.com`) にリダイレクトするかどうかを決定します。すべての訪問者が暗号化された接続を使用するように強制されるため、このオプションを有効にすることをお勧めします。Y を入力して Enter を押すと、有効になります。
   - www なしから www ありへのリダイレクトの有効化 - ドメインの頂点 (例: `https://example.com`) まで閲覧するユーザー を自動的にドメインの www サブドメイン (例: `https://www.example.com`) にリダイレクトするかを指定します。このオプションを有効にすることをお勧めします。ただし、ドメインの頂点を Google のウェブマスターツールなどの検索エンジンツールで希望のウェブサイトアドレスとして指定した場合、または頂点が IP を直接指しており、www のサブドメインが CNAME レコードを介してリファレンスしている場合は、無効にして代替オプションを有効にすることをお勧めします (www ありから www なしへのリダイレクトを有効化) 。Y を入力し、Enter を押して有効にします。
   - www ありから www なしへのリダイレクトを有効にする - ドメインの www サブドメイン (例: `https://www.example.com`) まで閲覧するユーザーを、自動的にドメインの頂点 (例: `https://example.com`) にリダイレクトするかを指定します。www なしから www ありへのリダイレクトを有効にした場合は、これを無効にすることをお勧めします。N を入力し、Enter を押して無効にします。

   選択した結果は次の例のようになります。

```
Enable/disable redirections

Please select the redirections you wish to enable or disable on your Bitnami
installation.



Enable HTTP to HTTPS redirection [Y/n]: Y



Enable non-www to www redirection [Y/n]: Y



Enable www to non-www redirection [y/N]: N
```

10. これから実行される変更が一覧表示されます。Y と入力し、Enter を押して確認し、続行します。

```
Changes to perform

The following changes will be performed to your Bitnami installation:

1. Stop web server
2. Configure web server to use a free Let's Encrypt certificate for the domains:
example.com www.example.com
3. Configure a cron job to automatically renew the certificate each month
4. Configure web server name to: example.com
5. Enable HTTP to HTTPS redirection (example: redirect http://example.com to
https://example.com)
6. Enable non-www to www redirection (example: redirect example.com to
 www.example.com)
7. Start web server once all changes have been performed



Do you agree to these changes? [Y/n]: Y
```

11. Let's Encrypt 証明書に関連付けるメールアドレスを入力し、Enter を押します。

```
Create a free HTTPS certificate with Let's Encrypt

Please provide a valid e-mail address for which to associate your Let's Encrypt
certificate.

Domain list: example.com www.example.com

Server name: example.com

E-mail address []:
```

12. Let's Encrypt サブスクライバー合意書を確認します。Y と入力し、Enter を押して契約に同意し、続行します。

```
The Let's Encrypt Subscriber Agreement can be found at:

https://letsencrypt.org/documents/LE-SA-v1.2-November-15-2017.pdf

Do you agree to the Let's Encrypt Subscriber Agreement? [Y/n]:
```

これらのアクションは、証明書のリクエストや指定したリダイレクトの設定など、インスタンスで HTTPS を有効にするために実行されます。

```
Performing changes to your installation

The Bitnami HTTPS Configuration Tool will perform any necessary actions to your
Bitnami installation. This may take some time, please be patient.
```

次の例のようなメッセージが表示された場合は、証明書は正常に発行され、検証され、インスタンスでリダイレクトが正常に設定されています。

```
Success

The Bitnami HTTPS Configuration Tool succeeded in modifying your installation.

The configuration report is shown below.

Backup files:
* /opt/bitnami/apache2/conf/httpd.conf.back.202005290035
* /opt/bitnami/apache2/conf/bitnami/bitnami-apps-prefix.conf.back.202005290035
* /opt/bitnami/apache2/conf/bitnami/bitnami.conf.back.202005290035

Find more details in the log file:

/tmp/bncert-202005290035.log

If you find any issues, please check Bitnami Support forums at:

https://community.bitnami.com

Press [Enter] to continue:
```

`bncert` ツールは、有効期限が切れる前、 80 日ごとに証明書の自動更新を実行します。インスタンスで追加のドメインやサブドメインを使用し、それらのドメインで HTTPS を有効にする場合は、上記のステップを繰り返します。

これで、Ghost インスタンスでの HTTPS の有効化が完了しました。次回に、設定したドメインを使用して Ghost ウェブサイトを参照するときには、HTTPS 接続にリダイレクトされるはずです。

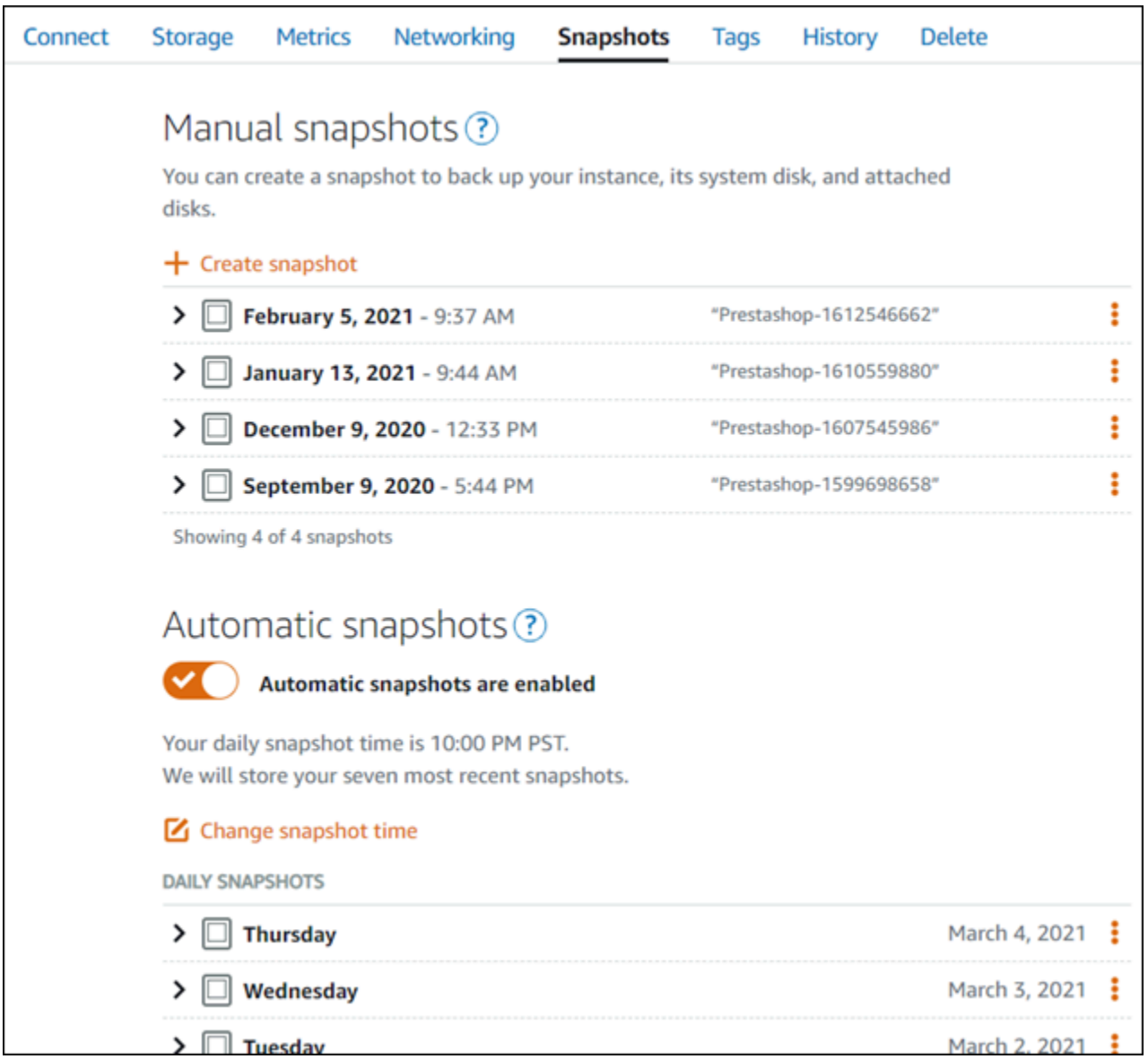
ステップ 7: Ghost のドキュメントを読み、引き続きウェブサイトの設定を続行する

Ghost のドキュメントを読み、ウェブサイトを管理およびカスタマイズする方法を確認します。詳細については、 Ghost のドキュメントを参照してください。

ステップ 8: インスタンスのスナップショットを作成する

Ghost ウェブサイトを希望どおりに設定したら、インスタンスの定期的なスナップショットを作成してバックアップします。スナップショットは手動で作成するか、自動スナップショットを有効にして Lightsail に毎日のスナップショットを作成させることができます。インスタンスに問題が発生した場合は、スナップショットを使用して新しい代替インスタンスを作成できます。詳細については、「スナップショット」を参照してください。

インスタンス管理ページの [スナップショット] タブで [スナップショットを作成する] を選択するか、[自動スナップショットを有効にする] を選択します。



詳細については、「Amazon Lightsail で Linux または Unix インスタンスのスナップショットを作成」および「Amazon Lightsail のインスタンスまたはディスクの自動スナップショットの有効化と無効化」を参照してください。

クイックスタートガイド： GitLab CE

GitLab CE インスタンスが Amazon Lightsail で起動して実行されたら、開始するためのいくつかのステップを以下に示します。

目次

- ステップ 2: GitLab CE 管理エリアにアクセスするためのデフォルトのアプリケーションパスワードを取得する
- ステップ 3: インスタンスに静的 IP アドレスをアタッチする
- ステップ 4: GitLab CE ウェブサイトの管理エリアにサインインする
- ステップ 5: 登録済みドメイン名へのトラフィックを GitLab CE ウェブサイトに送信する
- ステップ 6: GitLab CE ウェブサイトの HTTPS を設定する
- ステップ 7: GitLab CE ドキュメントを読み、引き続きウェブサイトを設定する
- ステップ 8: インスタンスのスナップショットを作成する

ステップ 1: Bitnami のドキュメントを確認する

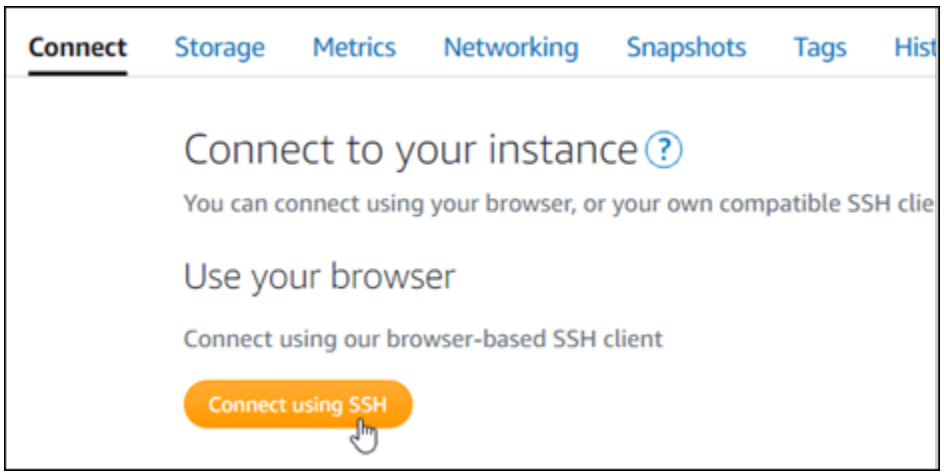
GitLab CE アプリケーションの設定方法については、Bitnami のドキュメントを参照してください。詳細については、「 用に GitLab Bitnami によってパッケージ化された CEAWS クラウド」を参照してください。

ステップ 2: GitLab CE 管理領域にアクセスするためのデフォルトのアプリケーションパスワードを取得する

GitLab CE ウェブサイトの管理エリアにアクセスするために必要なデフォルトのアプリケーションパスワードを取得するには、以下の手順を実行します。詳細については、「Amazon Lightsail での Bitnami インスタンスのアプリケーションユーザー名とパスワードの取得」を参照してください。

> **Important**
>
> Lightsail ブラウザベースの SSH/RDP クライアントは、IPv4 トラフィックのみを受け入れます。サードパーティーのクライアントを使用して、IPv6 経由でインスタンスに SSH または RDP 接続します。詳細については、「インスタンスに接続します」を参照してください。

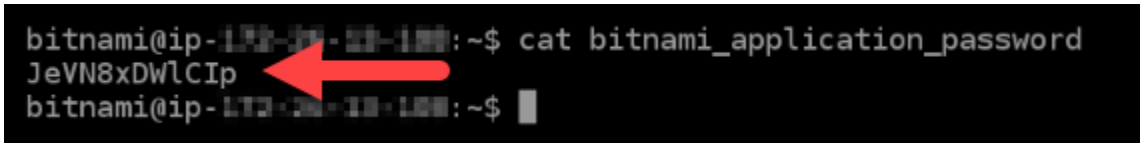
1. インスタンス管理ページの [接続] タブで、[SSH を使用して接続] を選択します。



2. 接続後に、次のコマンドを入力してアプリケーションのパスワードを取得します。

```
cat $HOME/bitnami_application_password
```

アプリケーションのデフォルトパスワードを含んだ、次の例のようなレスポンスが表示されます。



ステップ 3: インスタンスに静的 IP アドレスをアタッチする

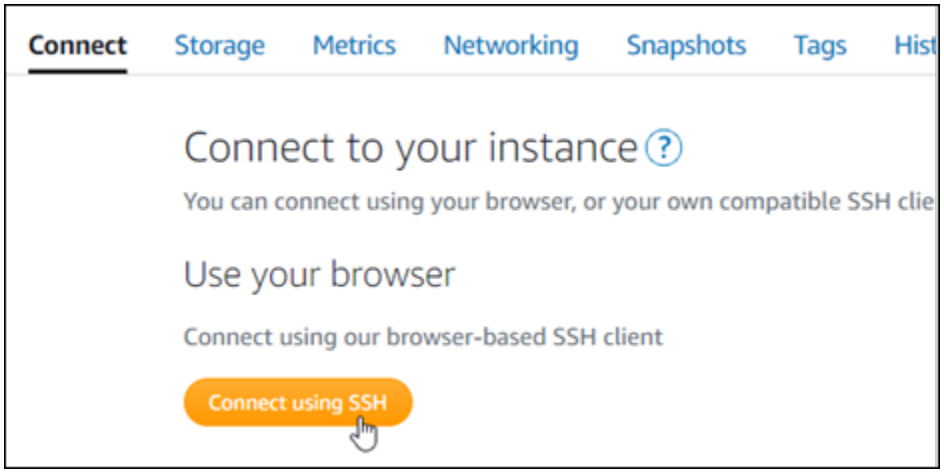
インスタンスを最初に作成した際に割り当てられたパブリック IP アドレスは、インスタンスを停止してスタートするたびに変更されます。パブリック IP アドレスが変更されないように、静的 IP アドレスを作成してインスタンスにアタッチする必要があります。それ以降、example.com などの登録したドメイン名をインスタンスで使用する際、毎回インスタンスを停止してスタートするたびにドメインの DNS レコードを更新する必要がなくなります。1 つの静的 IP を 1 つのインスタンスにアタッチできます。

インスタンス管理ページの [ネットワーク] タブで、[静的 IP の作成] または [静的 IP のアタッチ] (インスタンスにアタッチできる静的 IP を既に作成している場合) を選択して、ページの手順に従います。詳細については、「静的 IP を作成してインスタンスにアタッチする」を参照してください。

新しい静的 IP アドレスがインスタンスに添付されたら、次の手順を実行して、アプリケーションに新しい静的 IP アドレスを認識させる必要があります。

1. インスタンスの静的 IP アドレスは書き留めておきます。このIP アドレスはインスタンス管理ページの ヘッダーセクションに表示されます。

2. インスタンス管理ページの [Connect] (接続) タブで、[SSH を使用して接続] を選択します。



3. 接続後に、次のコマンドを入力します。*<StaticIP>* をインスタンスの新しい静的 IP アドレスに置き換えます。

```
sudo /opt/bitnami/configure_app_domain --domain <StaticIP>
```

例:

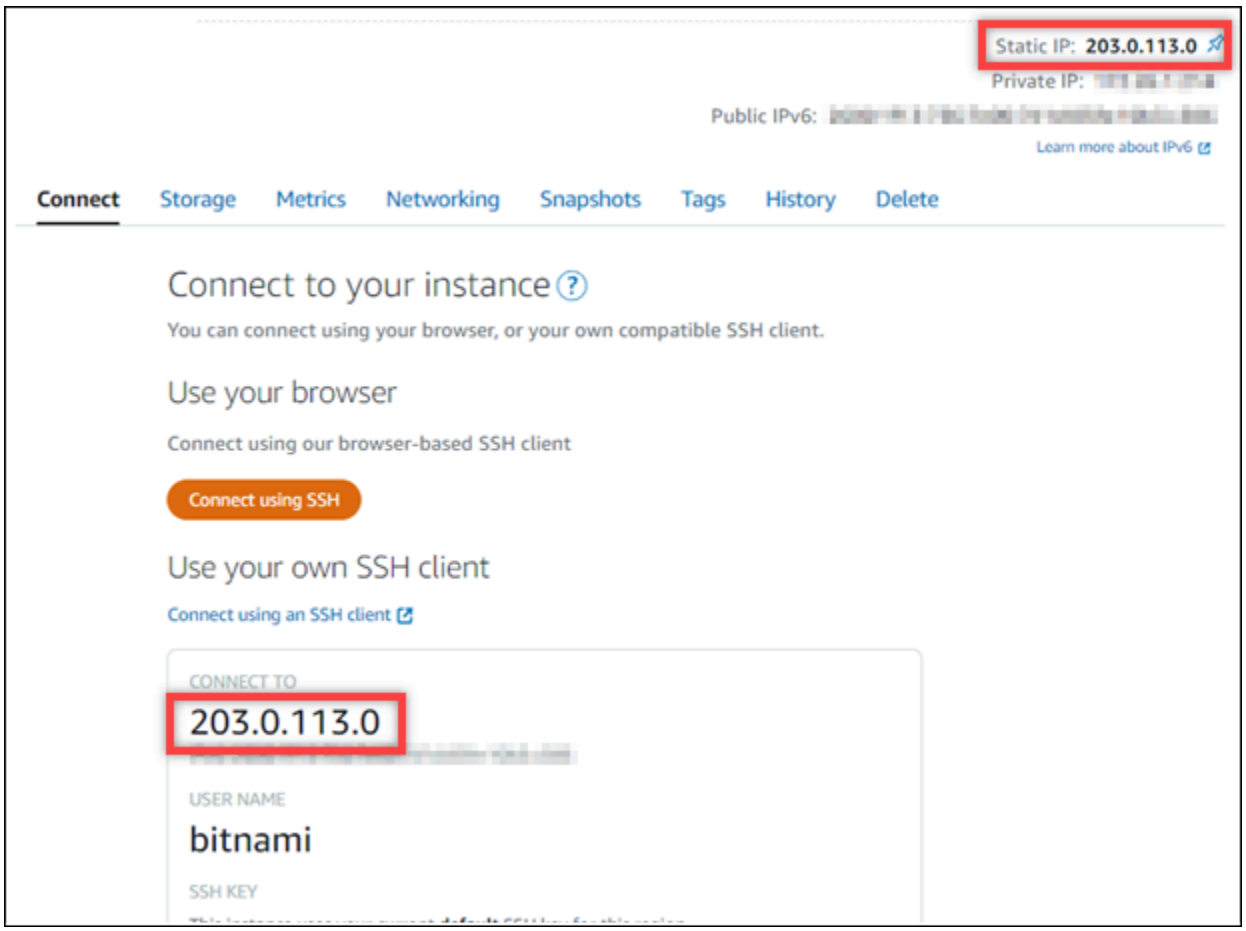
```
sudo /opt/bitnami/configure_app_domain --domain 203.0.113.0
```

次の例のようなレスポンスが表示されます。これで、インスタンス上のアプリケーションが新しい静的 IP アドレスを認識するようになります。

ステップ 4: GitLab CE ウェブサイトの管理エリアにサインインする

デフォルトのユーザーパスワードを取得したら、 GitLab CE ウェブサイトのホームページに移動し、管理エリアにサインインします。サインイン後に、ウェブサイトをカスタマイズしたり管理上の変更を行うことができます。 GitLab CE でできることの詳細については、このガイドの後半にある「ステップ 7: GitLab CE ドキュメントを読み、引き続きウェブサイトの設定を続行する」セクションを参照してください。

1. インスタンス管理ページの [Connect] (接続) タブにあるパブリック IP アドレスを書き留めます。パブリック IP アドレスは、インスタンス管理ページのヘッダーセクションにも表示されます。

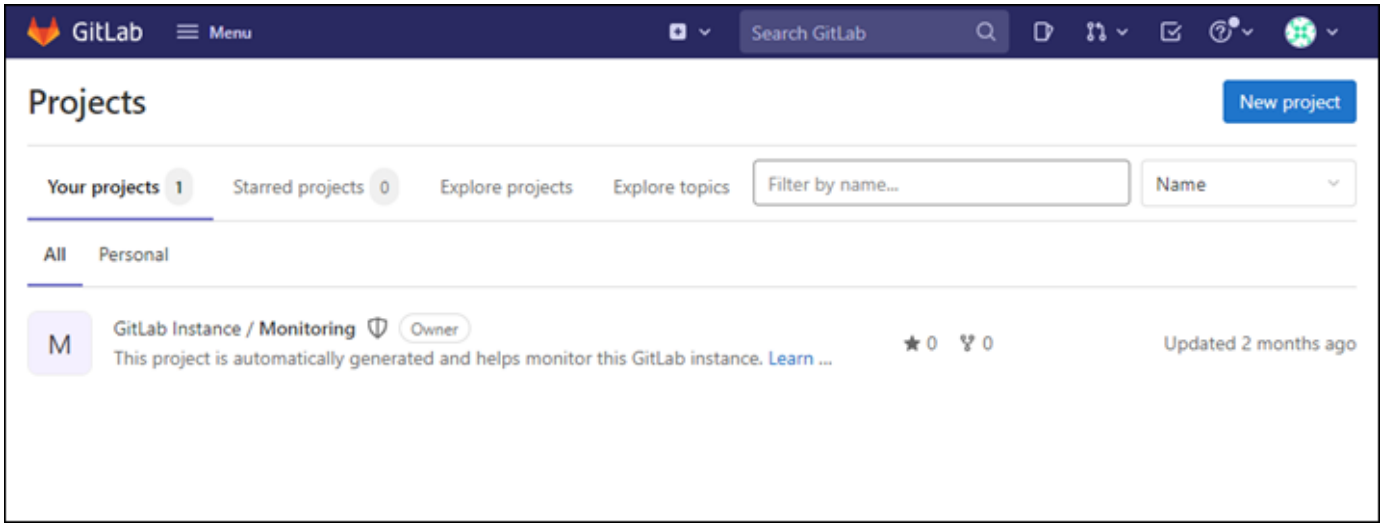


2. インスタンスのパブリック IP アドレスを参照します (例: `http://203.0.113.0` に移動します)。

   GitLab CE ウェブサイトのホームページが表示されます。接続がプライベートではない、セキュリティで保護されていない、またはセキュリティ上のリスクがある、などの警告がブラウザに表示されることがあります。これは、 GitLab CE インスタンスに SSL/TLS 証明書がまだ適用されていないために発生します。ブラウザウィンドウで、[Advanced] (詳細設定)、[Details] (詳細)、または [More information] (詳細情報) を選択して、使用可能なオプションを表示します。次に、プライベートまたは安全でない場合でも、ウェブサイトにアクセスすることを選択します。

3. デフォルトのユーザー名 (`root`) と、先ほど取得したデフォルトのパスワードを使用してサインインします。

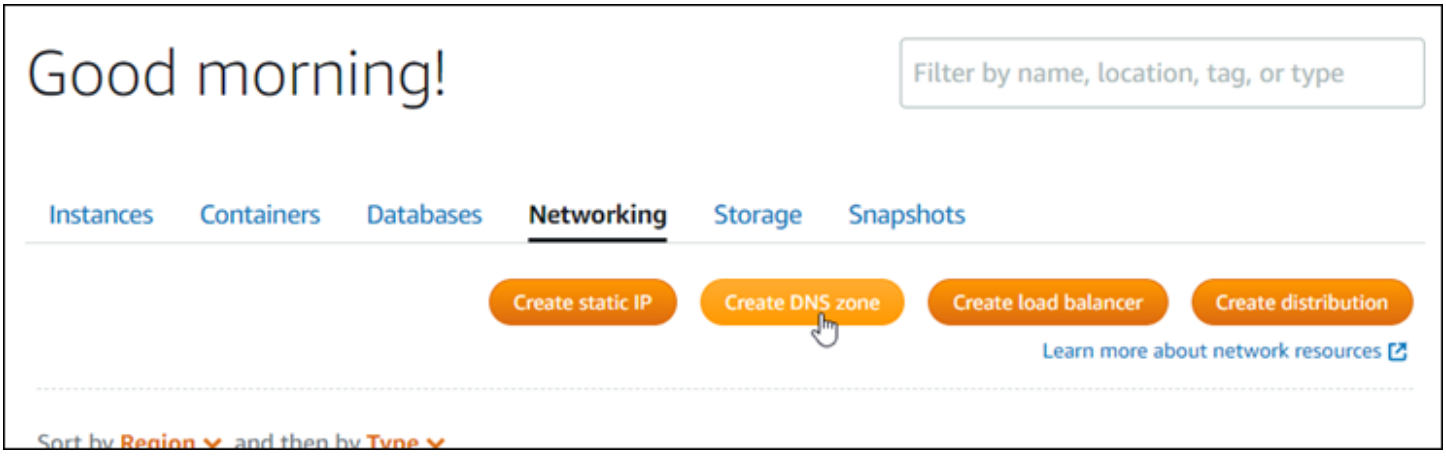
   Gitlab CE の管理ダッシュボードが表示されます。



ステップ 5: 登録済みドメイン名へのトラフィックを GitLab CE ウェブサイトに送信する

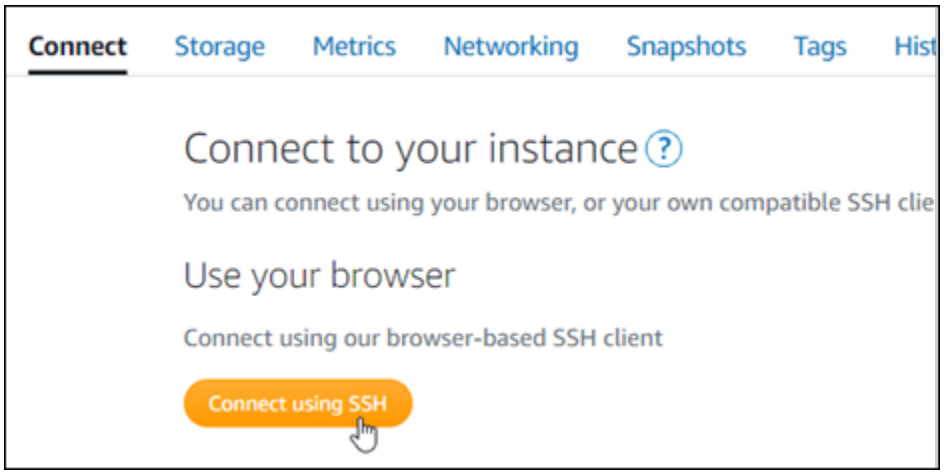
などの登録済みドメイン名のトラフィックを GitLab CE ウェブサイト`example.com`に送信するには、ドメインのドメインネームシステム (DNS) にレコードを追加します。DNS レコードは、通常、ドメインの登録先であるレジストラが管理またはホストします。ただし、Lightsail コンソールを使用して管理できるように、ドメインの DNS レコードの管理を Lightsail に転送することをお勧めします。

Lightsail コンソールのホームページのネットワークタブで、DNS ゾーンの作成 を選択し、ページの手順に従います。詳細については、「DNS ゾーンを作成してドメインの DNS レコードを管理する」を参照してください。



ドメイン名へのトラフィックがインスタンスにルーティングされたら、次の手順を実行して、GitLab CE にドメイン名を認識させる必要があります。

1. インスタンス管理ページの [Connect] (接続) タブで、[SSH を使用して接続] を選択します。



2. 接続後に、次のコマンドを入力します。*#DomainName#* を、インスタンスにトラフィックをルーティングするドメイン名に置き換えます。

```
sudo /opt/bitnami/configure_app_domain --domain <DomainName>
```

例:

```
sudo /opt/bitnami/configure_app_domain --domain example.com
```

次の例のようなレスポンスが表示されます。これで、 GitLab CE インスタンスがドメイン名を認識するようになります。

```
bitnami@ip-[redacted]:~$ sudo /opt/bitnami/configure_app_domain --domain example.com
Configuring domain to example.com
2022-06-09T18:44:00.235Z - info: Saving configuration info to disk
gitlab 18:44:00.36 INFO  ==> Updating external URL in GitLab configuration
gitlab 18:44:00.37 INFO  ==> Reconfiguring GitLab
gitlab 18:44:38.79 INFO  ==> Starting GitLab services
Disabling automatic domain update for IP address changes
```

このコマンドが失敗した場合、古いバージョンの GitLab CE インスタンスを使用している可能性があります。代わりに次のコマンドを実行してみてください。*#DomainName#* を、インスタンスにトラフィックをルーティングしているドメイン名に置き換えます。

```
cd /opt/bitnami/apps/gitlab
sudo ./bnconfig --machine_hostname <DomainName>
```

コマンドの実行が終了したら次のコマンドを入力し、サーバーが再起動するたびに bnconfig ツールが自動的に実行されないようにします。

```
sudo mv bnconfig bnconfig.disabled
```

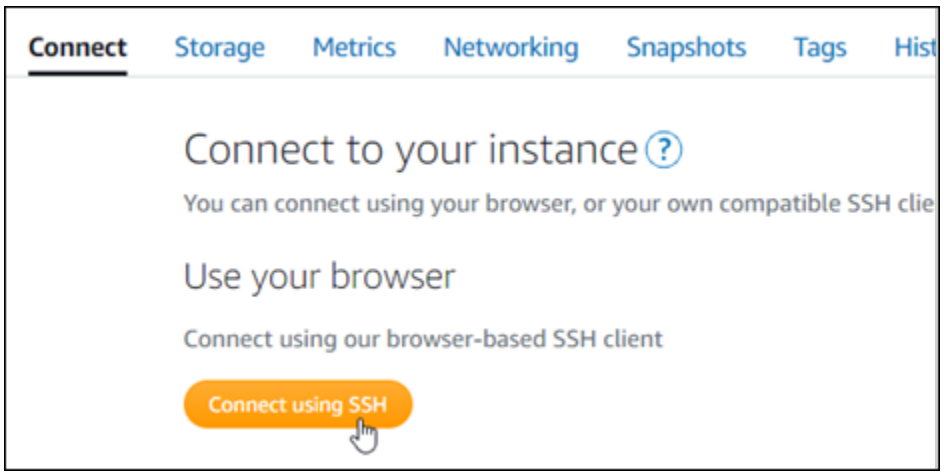
次に、SSL/TLS 証明書を生成して設定し、 GitLab CE ウェブサイトの HTTPS 接続を有効にする必要があります。詳細については、このガイドの次の「ステップ 6: GitLab CE ウェブサイトの HTTPS を設定する」のセクションに進んでください。

ステップ 6: CE ウェブサイトの GitLab HTTPS を設定する

GitLab CE ウェブサイトで HTTPS を設定するには、以下の手順を実行します。次の手順では、Lego クライアント の使い方を説明しています。これは、Let's Encrypt SSL/TLS 証明書を要求するコマンドラインツールです。

> **Important**
>
> この手順を開始する前に、トラフィックが GitLab CE インスタンスにルーティングされるようにドメインが設定されていることを確認してください。設定されていない場合、SSL/TLS 証明書の検証プロセスが失敗します。登録したドメイン名のトラフィックをルーティングするために、ドメインの DNS にレコードを追加します。DNS レコードは、通常、ドメインの登録先であるレジストラが管理またはホストします。ただし、Lightsail コンソールを使用して管理できるように、ドメインの DNS レコードの管理を Lightsail に転送することをお勧めします。
>
> Lightsail コンソールのホームページの「ドメインと DNS」タブで「DNS ゾーンの作成」を選択し、ページの手順に従います。詳細については、「Lightsail でドメインの DNS レコードを管理する DNS ゾーンの作成」を参照してください。

1. インスタンス管理ページの [接続] タブで、[SSH を使用して接続] を選択します。



2. 接続したら、次のコマンドを入力して、ディレクトリを一時ディレクトリ (/tmp) に変更します。

```
cd /tmp
```

3. 次のコマンドを入力して、Lego クライアントの最新バージョンをダウンロードします。このコマンドは、テープアーカイブ (tar) ファイルをダウンロードします。

```
curl -Ls https://api.github.com/repos/xenolf/lego/releases/latest | grep
 browser_download_url | grep linux_amd64 | cut -d '"' -f 4 | wget -i -
```

4. 次のコマンドを入力して tar ファイルからファイルを抽出します。*X.Y.Z* をダウンロードした Lego クライアントのバージョンに置き換えます。

```
tar xf lego_vX.Y.Z_linux_amd64.tar.gz
```

例:

```
tar xf lego_v4.7.0_linux_amd64.tar.gz
```

5. 以下のコマンドを入力して、Lego クライアントファイルを移動させる `/opt/bitnami/letsencrypt` ディレクトリを作成します。

```
sudo mkdir -p /opt/bitnami/letsencrypt
```

6. 以下のコマンドを入力して、Lego クライアントファイルを作成したディレクトリに移動します。

```
sudo mv lego /opt/bitnami/letsencrypt/lego
```

7. 次のコマンドを 1 つずつ入力して、インスタンスで実行されているアプリケーションサービスを停止します。

```
sudo service bitnami stop
sudo service gitlab-runsvdir stop
```

8. 次のコマンドを入力して、Lego クライアントを使って Let's Encrypt SSL/TLS 証明書を要求します。

```
sudo /opt/bitnami/letsencrypt/lego --tls --email="EmailAddress" --
domains="RootDomain" --domains="WwwSubDomain" --path="/opt/bitnami/letsencrypt" run
```

コマンド内の次のサンプルテキストを独自のテキストに置き換えます。

- *EmailAddress* — 登録通知用のメールアドレスです。
- *RootDomain* — GitLab CE ウェブサイトにトラフィックをルーティングするプライマリルートドメイン (例: `example.com`) 。
- *WwwSubDomain* — GitLab CE ウェブサイトにトラフィックをルーティングするプライマリルートドメインのwwwサブドメイン (例: `www.example.com`) 。

  コマンドに `--domains` パラメータを追加して指定することで、証明書に複数のドメインを指定することができます。複数のドメインを指定すると、Lego はサブジェクト別名 (SAN) 証明書を作成します。これにより、指定したすべてのドメインに対する有効な証明書が 1 つのみになります。リストの最初のドメインは証明書のCommonName「」として追加され、残りは「DNSNames」として証明書内の SAN 拡張に追加されます。

例:

```
sudo /opt/bitnami/letsencrypt/lego --tls --email="user@example.com" --
domains="example.com" --domains="www.example.com" --path="/opt/bitnami/letsencrypt"
 run
```

9. プロンプトが表示されたら、Y と Enter を押して利用規約に同意します。

次の例に示すようなレスポンスが表示されます。

```
2022/06/09 19:23:27 [INFO] [ example.com ] Server responded with a certificate.
```

成功した場合、一連の証明書が /opt/bitnami/letsencrypt/certificates ディレクトリに保存されます。このセットには、サーバー証明書ファイル (例: example.com.crt) とサーバ証明書キーファイル (例: example.com.key) が含まれています。

10. 次のコマンドを 1 つずつ入力して、インスタンス上の既存の証明書の名前を変更します。後で、これらの既存の証明書は新しい Let's Encrypt 証明書に置き換えます。

```
sudo mv /etc/gitlab/ssl/server.crt /etc/gitlab/ssl/server.crt.old
sudo mv /etc/gitlab/ssl/server.key /etc/gitlab/ssl/server.key.old
sudo mv /etc/gitlab/ssl/server.csr /etc/gitlab/ssl/server.csr.old
```

11. 次のコマンドを 1 つずつ入力して、 GitLab CE インスタンスのデフォルトの証明書ディレクトリである /etc/gitlab/ssl ディレクトリに新しい Let's Encript 証明書のシンボリックリンクを作成します。

```
sudo ln -sf /opt/bitnami/letsencrypt/certificates/Domain.key /etc/gitlab/ssl/
server.key
sudo ln -sf /opt/bitnami/letsencrypt/certificates/Domain.crt /etc/gitlab/ssl/
server.crt
```

コマンド内の *Domain* を Let's Encrypt 証明書を要求するときに指定したプライマリルートドメインに置き換えます。

例:

```
sudo ln -sf /opt/bitnami/letsencrypt/certificates/example.com.key /etc/gitlab/ssl/
server.key
sudo ln -sf /opt/bitnami/letsencrypt/certificates/example.com.crt /etc/gitlab/ssl/
server.crt
```

12. 次のコマンドを 1 つずつ入力して、移動先のディレクトリにある新しい Let's Encrypt 証明書のアクセス許可を変更します。

```
sudo chown root:root /etc/gitlab/ssl/server*
sudo chmod 600 /etc/gitlab/ssl/server*
```

13. 次のコマンドを入力して、CE インスタンス上のアプリケーションサービスを再起動します GitLab。

```
sudo service bitnami start
```

次回、設定したドメインを使用して GitLab CE ウェブサイトを参照すると、HTTPS 接続にリダイレクトされます。 GitLab CE インスタンスが新しい証明書を認識するまでに最大 1 時間かかる場合があることに注意してください。 GitLab CE ウェブサイトが接続を拒否した場合は、インスタンスを停止して起動し、もう一度試してください。

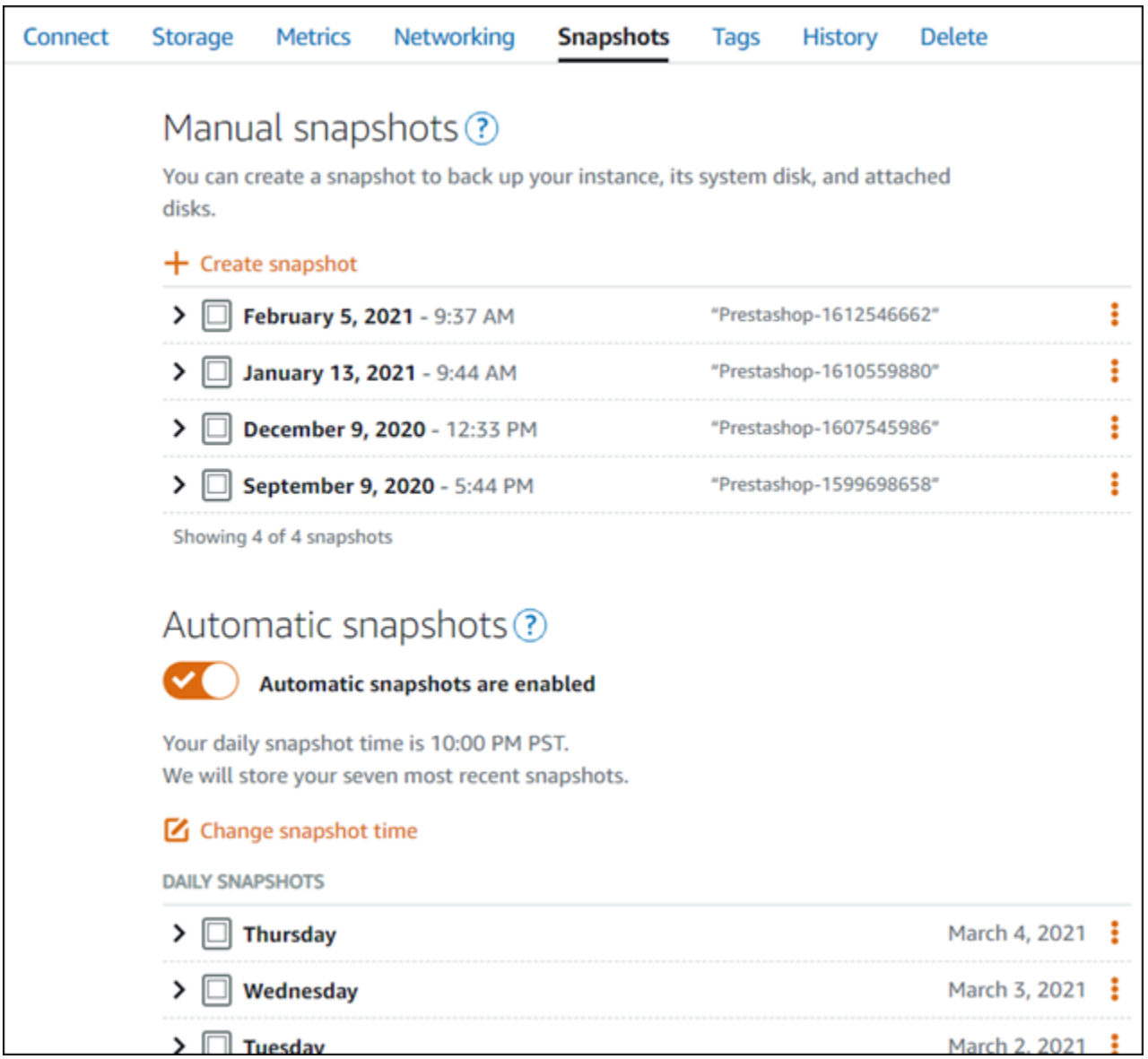
ステップ 7: GitLab CE ドキュメントを読み、引き続きウェブサイトを設定する

ウェブサイトを管理およびカスタマイズする方法については、 GitLab CE ドキュメントを参照してください。詳細については、「 GitLab ドキュメント」を参照してください。

ステップ 8: インスタンスのスナップショットを作成する

GitLab CE ウェブサイトを希望どおりに設定したら、インスタンスの定期的なスナップショットを作成してバックアップします。スナップショットを手動で作成することも、自動スナップショットを有効にして Lightsail に毎日のスナップショットを作成させることもできます。インスタンスに問題が発生した場合は、スナップショットを使用して新しい代替インスタンスを作成できます。詳細については、「スナップショット」を参照してください。

インスタンス管理ページの [スナップショット] タブで [スナップショットを作成する] を選択するか、[自動スナップショットを有効にする] を選択します。



詳細については、「Amazon Lightsail での Linux または Unix インスタンスのスナップショットの作成」または「Amazon Lightsail でのインスタンスまたはディスクの自動スナップショットの有効化または無効化」を参照してください。

クイックスタートガイド: Joomla!

Amazon Lightsail で起動した Joomla! インスタンスの使用を開始するステップについて説明します。

目次

- ステップ 2: Joomla! コントロールパネルにアクセスするためのデフォルトのアプリケーションパスワードを取得する
- ステップ 3: インスタンスに静的 IP アドレスをアタッチする
- ステップ 4: Joomla! ウェブサイトのコントロールパネルにサインインする
- ステップ 5: 登録済みドメイン名へのトラフィックを Joomla! ウェブサイトに送信する
- ステップ 6: Joomla! ウェブサイトの HTTPS を設定する
- ステップ 7: Joomla! のドキュメントを読み、引き続きウェブサイトの設定を続行する
- ステップ 8: インスタンスのスナップショットを作成する

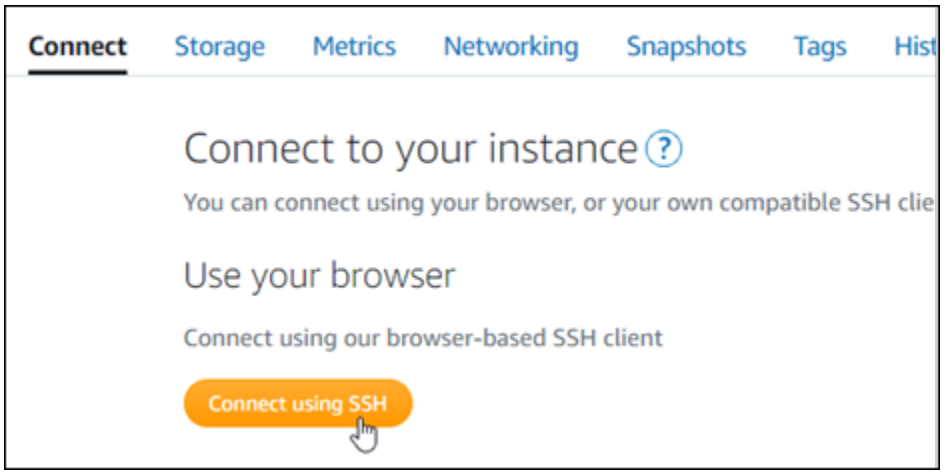
ステップ 1: Bitnami のドキュメントを確認する

Joomla! アプリケーションの設定方法については、Bitnami ドキュメントを参照してください。詳細については、Joomla! を参照してください。Bitnami により AWS クラウド 用にパッケージ化されています。

ステップ 2: Joomla! コントロールパネルにアクセスするためのデフォルトのアプリケーションパスワードを取得する

次の手順を完了して、Joomla! ウェブサイトのコントロールパネルにアクセスする際に必要となるデフォルトのアプリケーションパスワードを取得します。詳細については、「Amazon LightsailのBitnami インスタンス向けにアプリケーションのユーザー名とパスワードを取得する」を参照してください。

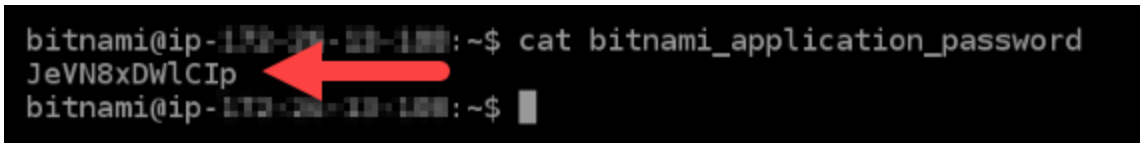
1. インスタンス管理ページの [接続] タブで、[SSH を使用して接続] を選択します。



2. 接続後に、次のコマンドを入力してアプリケーションのパスワードを取得します。

```
cat $HOME/bitnami_application_password
```

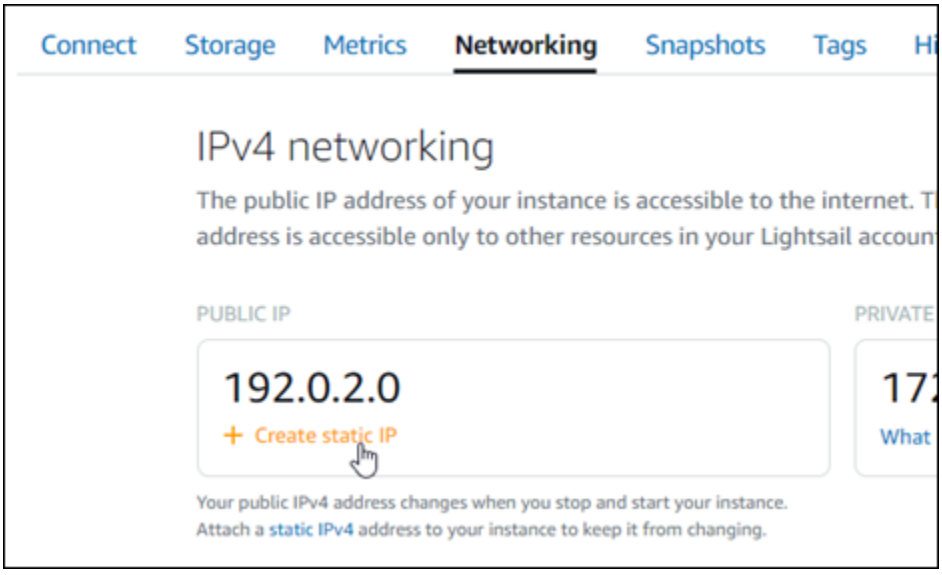
アプリケーションのデフォルトパスワードを含んだ、次の例のようなレスポンスが表示されます。



ステップ 3: インスタンスに静的 IP アドレスをアタッチする

インスタンスを最初に作成した際に割り当てられたパブリック IP アドレスは、インスタンスを停止してスタートするたびに変更されます。パブリック IP アドレスが変更されないように、静的 IP アドレスを作成してインスタンスにアタッチする必要があります。それ以降、example.com などの登録したドメイン名をインスタンスで使用する際、毎回インスタンスを停止してスタートするたびにドメインの DNS レコードを更新する必要がなくなります。1 つの静的 IP を 1 つのインスタンスにアタッチできます。

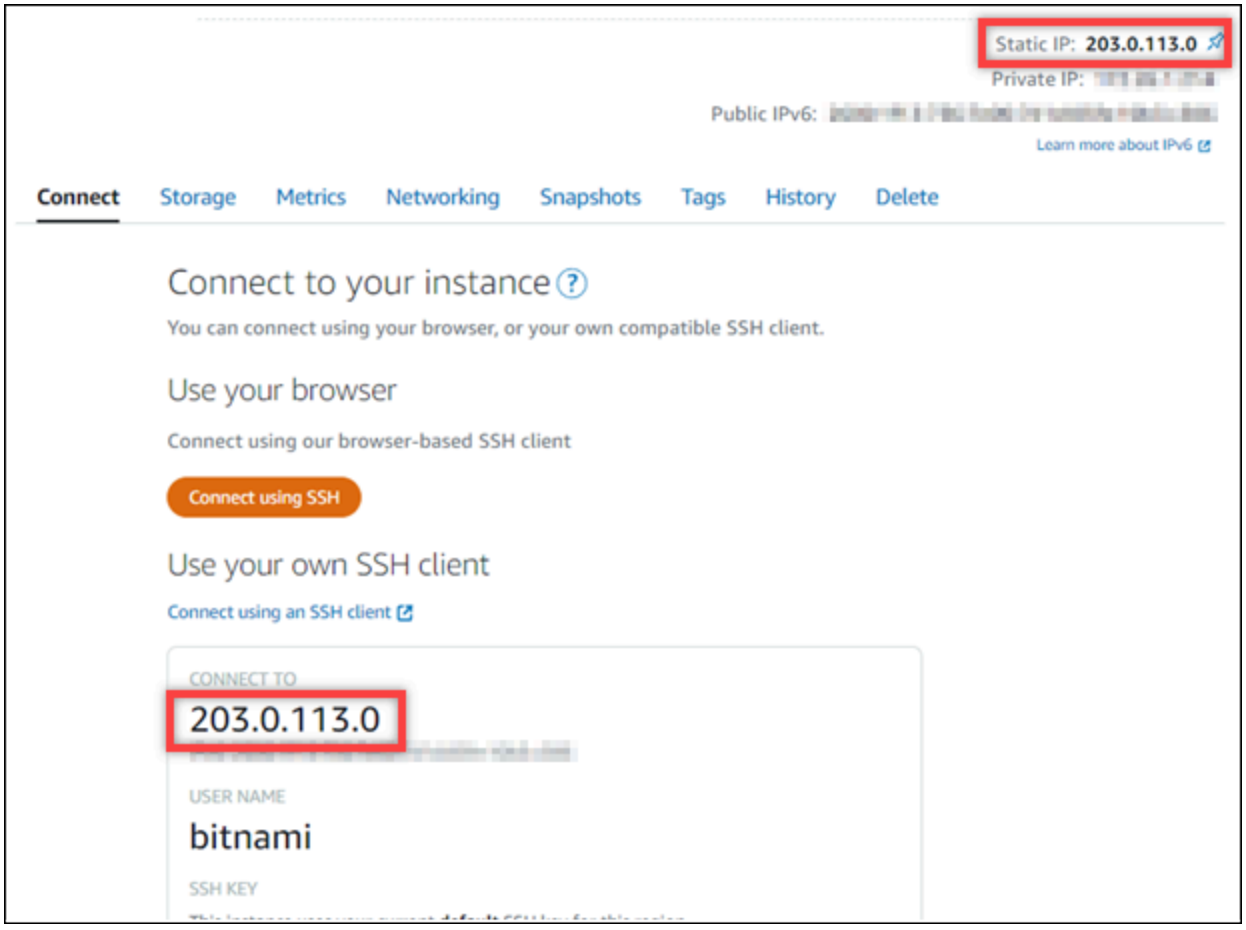
インスタンス管理ページの [ネットワーク] タブで、[静的 IP の作成] または [静的 IP のアタッチ] (インスタンスにアタッチできる静的 IP を既に作成している場合) を選択して、ページの手順に従います。詳細については、「静的 IP を作成してインスタンスにアタッチする」を参照してください。



ステップ 4: Joomla! ウェブサイトのコントロールパネルにサインインする

デフォルトのユーザーパスワードを取得したら、以下の手順に従って Joomla! ウェブサイトのホームページに移動し、コントロールパネルにサインインします。サインイン後に、ウェブサイトをカスタマイズしたり管理上の変更を行うことができます。Joomla! で実行できる事項の詳細については、「ステップ 7: Joomla! のドキュメントを読み、引き続きウェブサイトの設定を続行する」のセクションを参照してください。

1. インスタンス管理ページの [Connect] (接続) タブにあるパブリック IP アドレスを書き留めます。パブリック IP アドレスは、インスタンス管理ページのヘッダーセクションにも表示されます。



2. インスタンスのパブリック IP アドレスを参照します (例: `http://203.0.113.0` に移動します)。

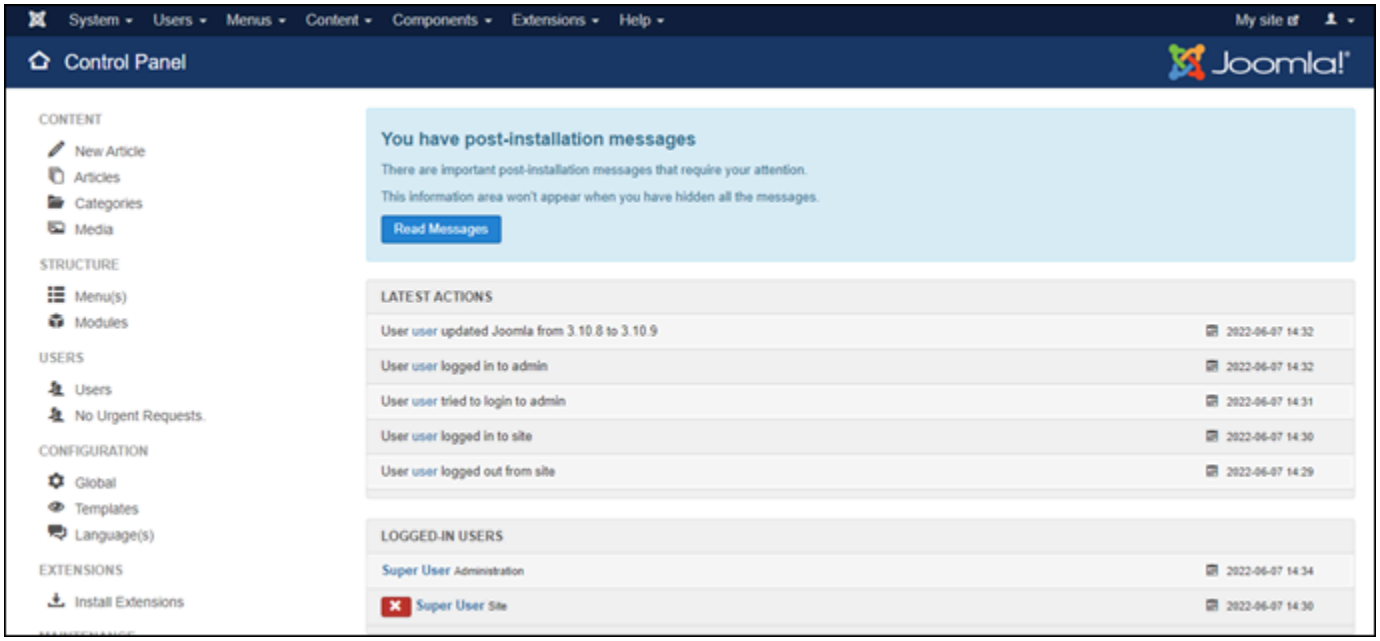
   Joomla! ウェブサイトのホームページが表示されます。

3. Joomla! ウェブサイトのホームページで、右下にある [Manage] (管理) を選択します。

[Manage] (管理) バナーが表示されない場合は、`http://<PublicIP>/administrator/` を参照することでサインインページにアクセスすることができます。*<PublicIP>* を、インスタンスのパブリック IP アドレスに置き換えます。

4. デフォルトのユーザー名 (`user`) と、先ほど取得したデフォルトのパスワードを使用してサインインします。

   Joomla! の管理コントロールパネルが表示されます。



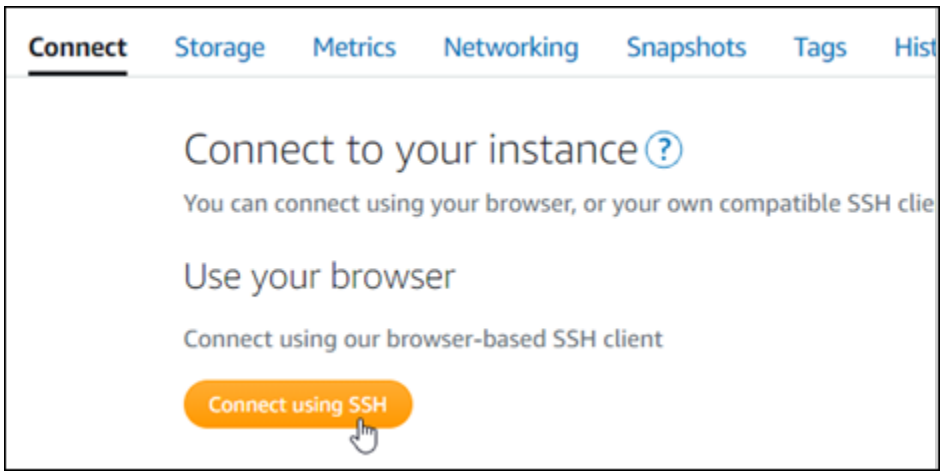
ステップ 5: 登録済みドメイン名へのトラフィックを Joomla! ウェブサイトに送信する

`example.com` などの登録済みドメイン名のトラフィックを Joomla! ウェブサイトに送信するには、ドメインのドメインネームシステム (DNS) にレコードを追加します。DNS レコードは、通常、ドメインの登録先であるレジストラが管理またはホストします。ただし、ドメインの DNS レコードの管理を Lightsail に引き渡して、Lightsail コンソールで管理できるようにすることをお勧めします。

Lightsail コンソールのホームページの [Domains & DNS] (ドメインと DNS) タブで、[Create DNS zone] (DNS ゾーンの作成) を選択し、ページに記載される手順に従います。詳細については、「Lightsail で DNS ゾーンを作成し、ドメインの DNS レコードを管理する」を参照してください。

ドメイン名へのトラフィックがインスタンスにルーティングされたら、次の手順を実行して、Joomla! ソフトウェアにドメイン名を認識させます。

1. インスタンス管理ページの [Connect] (接続) タブで、[SSH を使用して接続] を選択します。



2. Bitnami とは、多くのブループリントのファイル構造を変更するプロセスです。この手順にあるファイルパスは、Bitnami ブループリントがネイティブ Linux システムパッケージを使用しているか (アプローチ A)、または自己完結型インストール (アプローチ B) であるかによって変わる場合があります。Bitnami のインストールタイプと従うべき方法を特定するには、接続後に次のコマンドを実行します。

```
test ! -f "/opt/bitnami/common/bin/openssl" && echo "Approach A: Using system
 packages." || echo "Approach B: Self-contained installation."
```

3. 前のコマンドで得られる結果に、アプローチ A を使用すべきと示されている場合は次の手順を実行します。そうでない場合、前のコマンドで得られる結果にアプローチ B を使用すべきと示されている場合は、ステップ 4 に進みます。

   1. 以下のコマンドを入力して、Vim を使用して Apache 仮想ホスト設定ファイルを開き、ドメイン名の仮想ホストを作成します。

      ```
      sudo vim /opt/bitnami/apache2/conf/vhosts/joomla-vhost.conf
      ```

   2. I キーを押して Vim の挿入モードに移ります。
   3. 次の例に示されているように、ドメイン名をファイルに追加します。この例では、`example.com` および `www.example.com` ドメインを使用しています。

```
<VirtualHost 127.0.0.1:80 _default_:80>
  ServerName www.example.com
  ServerAlias example.com
  DocumentRoot /opt/bitnami/joomla
  <Directory "/opt/bitnami/joomla">
    Options -Indexes +FollowSymLinks -MultiViews
    AllowOverride None
    Require all granted
  </Directory>
  Include "/opt/bitnami/apache/conf/vhosts/htaccess/joomla-htaccess.conf"
</VirtualHost>
```

4. ESC キーを押して「:wq!」と入力し、編集内容を保存 (書き込んで) Vim を終了します。
5. 次のコマンドを入力して Apache サーバーを再起動します。

```
sudo /opt/bitnami/ctlscript.sh restart apache
```

4. 前のコマンドで得られる結果にアプローチ B を使用すべきと示された場合は、次の手順を実行します。

1. 以下のコマンドを入力して、Vim を使用して Apache 仮想ホスト設定ファイルを開き、ドメイン名の仮想ホストを作成します。

```
sudo vim /opt/bitnami/apps/joomla/conf/httpd-vhosts.conf
```

2. I キーを押して Vim の挿入モードに移ります。
3. 次の例に示されているように、ドメイン名をファイルに追加します。この例では、example.com および www.example.com ドメインを使用しています。

```
<VirtualHost *:80>
  ServerName example.com
  ServerAlias www.example.com
  ...
```

4. ESC キーを押して「:wq!」と入力し、編集内容を保存 (書き込んで) Vim を終了します。
5. 以下のコマンドを入力して、bitnami-apps-vhosts.conf ファイルに Joomla! の httpd-vhosts.conf ファイルが含まれていることを確認します。

```
sudo vim /opt/bitnami/apache2/conf/bitnami/bitnami-apps-vhosts.conf
```

ファイル内で、次の行を見つけます。ない場合には追加してください。

```
Include "/opt/bitnami/apps/joomla/conf/httpd-vhosts.conf"
```

6. 次のコマンドを入力して Apache サーバーを再起動します。

```
sudo /opt/bitnami/ctlscript.sh restart apache
```

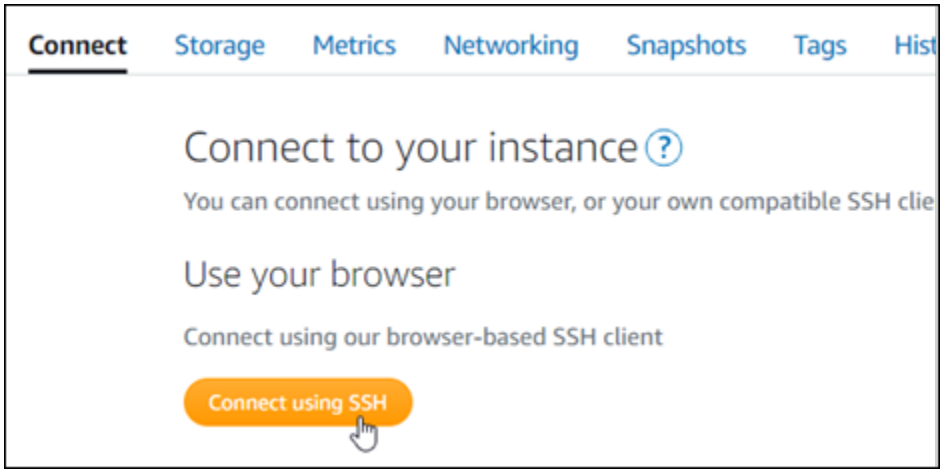
インスタンスに設定したドメイン名を参照すると、Joomla! ウェブサイトのホームページへと移動します。次に、SSL/TLS 証明書を生成して設定し、Joomla! ウェブサイトの HTTPS 接続を有効にします。詳細については、本ガイドの次の「ステップ 6: Joomla! ウェブサイトの HTTPS を設定する」のセクションを参照してください。

ステップ 6: Joomla! ウェブサイトの HTTPS を設定する

Joomla! ウェブサイトで HTTPS を設定するには、以下の手順を実行します。次の手順では、Bitnami HTTPS 設定ツール (`bncert-tool`) の使い方を説明しています。これは、Let's Encrypt SSL/TLS 証明書を要求するコマンドラインツールです。詳細については、Bitnami ドキュメントの「Bitnami 設定ツールの詳細を確認する」を参照してください。

> **Important**
>
> この手順を開始する前に、Joomla! インスタンスにトラフィックがルーティングされるようにドメインが設定済みであることを確認してください。設定されていない場合、SSL/TLS 証明書の検証プロセスが失敗します。

1. インスタンス管理ページの [接続] タブで、[SSH を使用して接続] を選択します。



2. 接続されたら、以下のコマンドを入力し、インスタンスに bncert ツールがインストールされていることを確認します。

```
sudo /opt/bitnami/bncert-tool
```

以下のレスポンスのいずれかが表示されます。

- レスポンスにコマンドが見つからないと表示された場合、bncert ツールがインスタンスにインストールされていないことを示しています。この手順の次のステップに進み、bncert ツールをインスタンスにインストールします。
- レスポンスに「Welcome to the Bitnami HTTPS configuration tool (Bitnami HTTPS 設定ツールへようこそ)」と表示された場合は、インスタンスに bncert ツールがインストールされています。この手順のステップ 8 に進んでください。
- bncert ツールがインスタンスにインストールされてからしばらく経っている場合、ツールのアップデートバージョンが利用可能であることを示すメッセージが表示されることがあります。ダウンロードすることを選択し、`sudo /opt/bitnami/bncert-tool` コマンドを入力して bncert ツールを再度実行してください。この手順のステップ 8 に進んでください。

3. 以下のコマンドを入力して、bncert の実行ファイルをインスタンスにダウンロードします。

```
wget -O bncert-linux-x64.run https://downloads.bitnami.com/files/bncert/latest/
bncert-linux-x64.run
```

4. 以下のコマンドを入力して、インスタンスに bncert ツールの実行ファイル用のディレクトリを作成します。

```
sudo mkdir /opt/bitnami/bncert
```

5. 以下のコマンドを入力して、プログラムとして実行できるファイルを bncert に実行させます。

```
sudo chmod +x /opt/bitnami/bncert/bncert-linux-x64.run
```

6. 次のコマンドを入力して、sudo /opt/bitnami/bncert-tool コマンドを入力すると bncert ツールを実行するシンボリックリンクを作成します。

```
sudo ln -s /opt/bitnami/bncert/bncert-linux-x64.run /opt/bitnami/bncert-tool
```

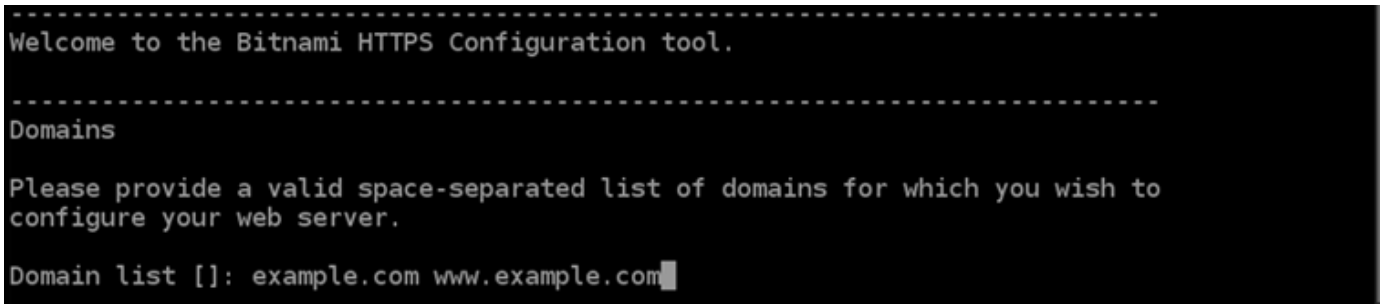
これでインスタンスに bncert ツールをインストールする手順は完了です。

7. 次のコマンドを入力して、bncert ツールを実行しましょう。

```
sudo /opt/bitnami/bncert-tool
```

8. 次の例に示すように、プライマリドメイン名と代替ドメイン名の間はスペースで区切って入力します。

   ドメインがインスタンスのパブリック IP アドレスにトラフィックをルーティングするように設定されていない場合、bncert ツールは、続行する前にその設定を行うように要求します。ドメインは、bncert ツールを使用して HTTPS を有効にしているインスタンスでのパブリック IP アドレスにトラフィックをルーティングする必要があります。これはドメインを所有していることを確認し、証明書の検証として機能します。



9. bncert ツールは、ウェブサイトのリダイレクトの設定方法を尋ねます。使用できるオプションは次のとおりです。

   - HTTP から HTTPS へのリダイレクトを有効にする - HTTP バージョンのウェブサイトを閲覧するユーザー (例: http:/example.com) を自動的に HTTPS バージョン (例: https://example.com) にリダイレクトするかどうかを決定します。すべての訪問者が暗号化された接続を使用するように強制されるため、このオプションを有効にすることをお勧めします。Y を入力して Enter を押すと、有効になります。
   - www なしから www ありへのリダイレクトの有効化 - ドメインの頂点 (例: https://example.com) まで閲覧するユーザー を自動的にドメインの www サブドメイン (例: https://www.example.com) にリダイレクトするかを指定します。このオプションを有効にすることをお勧めします。ただし、ドメインの頂点を Google のウェブマスターツールなどの検索エンジンツールで希望のウェブサイトアドレスとして指定した場合、または頂点が IP を直接指しており、www のサブドメインが CNAME レコードを介してリファレンスしている場合は、無効にして代替オプションを有効にすることをお勧めします (www ありから www なしへのリダイレクトを有効化) 。Y を入力し、Enter を押して有効にします。
   - www ありから www なしへのリダイレクトを有効にする - ドメインの www サブドメイン (例: https://www.example.com) まで閲覧するユーザーを、自動的にドメインの頂点 (例: https://example.com) にリダイレクトするかを指定します。www なしから www あ

りへのリダイレクトを有効にした場合は、これを無効にすることをお勧めします。N を入力し、Enter を押して無効にします。

選択した結果は次の例のようになります。

```
Enable/disable redirections

Please select the redirections you wish to enable or disable on your Bitnami
installation.



Enable HTTP to HTTPS redirection [Y/n]: Y



Enable non-www to www redirection [Y/n]: Y



Enable www to non-www redirection [y/N]: N
```

10. これから実行される変更が一覧表示されます。Y と入力し、Enter を押して確認し、続行します。

```
Changes to perform

The following changes will be performed to your Bitnami installation:

1. Stop web server
2. Configure web server to use a free Let's Encrypt certificate for the domains:
example.com www.example.com
3. Configure a cron job to automatically renew the certificate each month
4. Configure web server name to: example.com
5. Enable HTTP to HTTPS redirection (example: redirect http://example.com to
https://example.com)
6. Enable non-www to www redirection (example: redirect example.com to
 www.example.com)
7. Start web server once all changes have been performed



Do you agree to these changes? [Y/n]: Y
```

11. Let's Encrypt 証明書に関連付けるメールアドレスを入力し、Enter を押します。

```
Create a free HTTPS certificate with Let's Encrypt

Please provide a valid e-mail address for which to associate your Let's Encrypt
certificate.

Domain list: example.com www.example.com

Server name: example.com

E-mail address []:
```

12. Let's Encrypt サブスクライバー合意書を確認します。Y と入力し、Enter を押して契約に同意し、続行します。

```
The Let's Encrypt Subscriber Agreement can be found at:

https://letsencrypt.org/documents/LE-SA-v1.2-November-15-2017.pdf

Do you agree to the Let's Encrypt Subscriber Agreement? [Y/n]:
```

これらのアクションは、証明書のリクエストや指定したリダイレクトの設定など、インスタンスで HTTPS を有効にするために実行されます。

```
Performing changes to your installation

The Bitnami HTTPS Configuration Tool will perform any necessary actions to your
Bitnami installation. This may take some time, please be patient.

|
```

次の例のようなメッセージが表示された場合は、証明書は正常に発行され、検証され、インスタンスでリダイレクトが正常に設定されています。

```
Success

The Bitnami HTTPS Configuration Tool succeeded in modifying your installation.

The configuration report is shown below.

Backup files:
* /opt/bitnami/apache2/conf/httpd.conf.back.202005290035
* /opt/bitnami/apache2/conf/bitnami/bitnami-apps-prefix.conf.back.202005290035
* /opt/bitnami/apache2/conf/bitnami/bitnami.conf.back.202005290035

Find more details in the log file:

/tmp/bncert-202005290035.log

If you find any issues, please check Bitnami Support forums at:

https://community.bitnami.com

Press [Enter] to continue:
```

`bncert` ツールは、有効期限が切れる前、 80 日ごとに証明書の自動更新を実行します。インスタンスで追加のドメインやサブドメインを使用し、それらのドメインで HTTPS を有効にする場合は、上記のステップを繰り返します。

これで、Joomla! インスタンスでの HTTPS の有効化が完了しました。次回に、設定したドメインを使用して Joomla! ウェブサイトを参照するときには、HTTPS 接続にリダイレクトされるはずです。

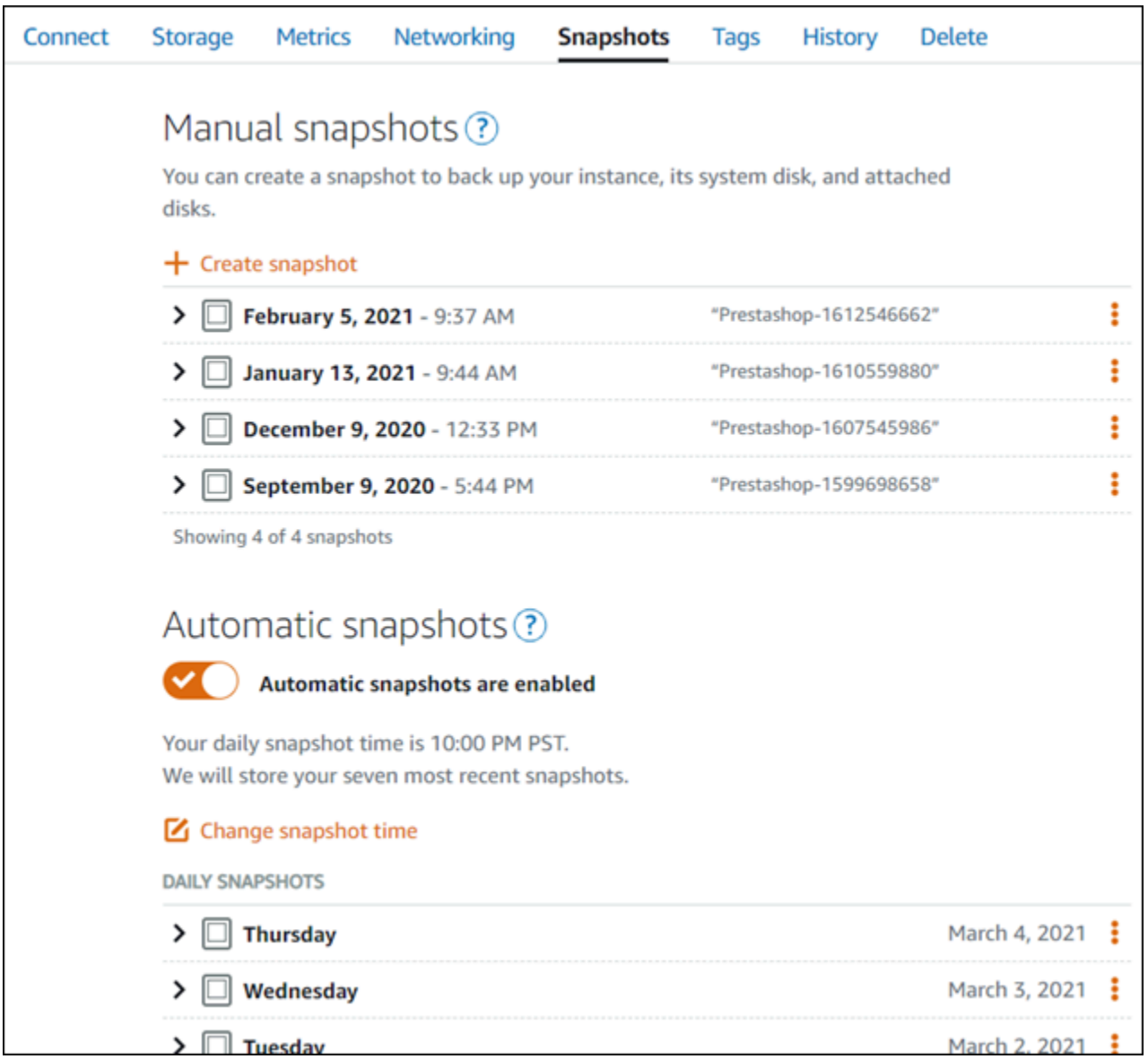
ステップ 7: Joomla! のドキュメントを読み、引き続きウェブサイトの設定を続行する

Joomla! のドキュメントを読み、ウェブサイトを管理およびカスタマイズする方法を確認します。詳細については、Joomla! を参照してください。ドキュメント

ステップ 8: インスタンスのスナップショットを作成する

Joomla! ウェブサイトを希望どおりに設定したら、インスタンスの定期的なスナップショットを作成してバックアップします。スナップショットは手動で作成するか、自動スナップショットを有効にして Lightsail に毎日のスナップショットを作成させることができます。インスタンスに問題が発生した場合は、スナップショットを使用して新しい代替インスタンスを作成できます。詳細については、「スナップショット」を参照してください。

インスタンス管理ページの [スナップショット] タブで [スナップショットを作成する] を選択するか、[自動スナップショットを有効にする] を選択します。



詳細については、「Amazon Lightsail で Linux または Unix インスタンスのスナップショットを作成」および「Amazon Lightsail のインスタンスまたはディスクの自動スナップショットの有効化と無効化」を参照してください。

クイックスタートガイド: LAMP

Amazon Lightsail で起動した LAMP インスタンスの使用を開始するステップについて説明します。

ステップ 1: LAMP インスタンスのデフォルトのアプリケーションパスワードを取得する

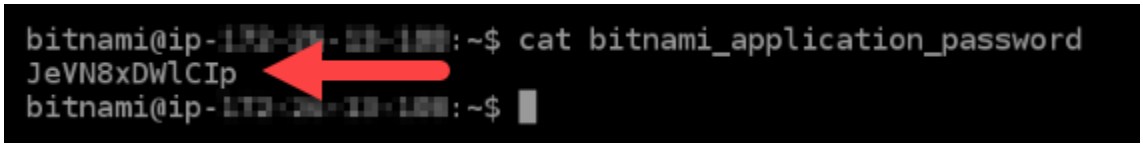
インスタンスにプリインストールされているアプリケーションやサービスにアクセスするには、アプリケーションのデフォルトパスワードが必要です。

1. インスタンス管理ページの [接続] タブで、[SSH を使用して接続] を選択します。
2. 接続後に、次のコマンドを入力してアプリケーションのパスワードを取得します。

```
cat bitnami_application_password
```

> **Note**
>
> ユーザーのホームディレクトリ以外のディレクトリで作業している場合は、「`cat $HOME/bitnami_application_password`」と入力します。

次のようなレスポンスにアプリケーションのデフォルトパスワードが表示されます。



詳細については、「Amazon Lightsailの Bitnami インスタンス向けにアプリケーションのユーザー名とパスワードを取得する」を参照してください。

ステップ 2: LAMP インスタンスに静的 IP アドレスをアタッチする

インスタンスにアタッチしたデフォルトの動的なパブリック IP アドレスは、インスタンスを停止して開始するたびに変わります。パブリック IP アドレスが変わらないようにするには、静的 IP アドレスを作成してインスタンスにアタッチします。その後にドメイン名をインスタンスで使用すると、インスタンスを停止して開始するたびにドメインの DNS レコードを更新する必要がなくなります。1 つの静的 IP を 1 つのインスタンスにアタッチできます。

インスタンス管理ページで、[ネットワーキング] タブの [静的 IP の作成] を選択し、ページの手順に従います。

詳細については、「静的 IP を作成してインスタンスにアタッチする」を参照してください。

ステップ 3: LAMP インスタンスのウェルカムページにアクセスする

インスタンスのパブリック IP アドレスに移動してインスタンスにインストールされているアプリケーションにアクセスし、phpMyAdmin または Bitnami ドキュメントを参照します。

1. インスタンス管理ページの [接続] タブで、パブリック IP を書き留めます。
2. パブリック IP アドレスを参照します (例: `http://192.0.2.3` に移動します)。

詳細については、「Amazon Lightsailの Bitnami インスタンス向けにアプリケーションのユーザー名とパスワードを取得する」を参照してください。

ステップ 4: ドメイン名を LAMP インスタンスにマッピングする

ドメイン名 (`example.com` など) をインスタンスにマッピングするには、ドメインのドメインネームシステム (DNS) にレコードを追加します。DNS レコードは、通常、ドメインの登録先であるレジストラが管理またはホストします。ただし、ドメインの DNS レコードの管理を Lightsail に引き渡して、Lightsail コンソールで管理できるようにすることをお勧めします。

Lightsail コンソールのホームページの [Domains & DNS] (ドメインと DNS) タブで、[Create DNS zone] (DNS ゾーンの作成) を選択し、ページに記載される手順に従います。

詳細については、「Lightsail で DNS ゾーンを作成し、ドメインの DNS レコードを管理する」を参照してください。

ステップ 5: Bitnami のドキュメントを確認する

Bitnami のドキュメントで、アプリケーションのデプロイ、SSL 証明書による HTTPS サポートの有効化、SFTP を使用したファイルのサーバーへのアップロードなどの方法を確認します。

詳細については、「AWS クラウド 用の Bitnami LAMP」を参照してください。

ステップ 6: LAMP インスタンスのスナップショットを作成する

スナップショットは、インスタンスのシステムディスクおよびオリジナル設定のコピーです。スナップショットには、メモリ、CPU、ディスクサイズ、データ転送レートなどの情報が含まれています。スナップショットを、新しいインスタンスのベースラインまたはデータのバックアップとして使用できます。

インスタンス管理ページの [スナップショット] タブで、スナップショット名を入力して [スナップショットの作成] を選択します。

詳細については、「Linux または Unix インスタンスのスナップショットを作成する」を参照してください。

クイックスタートガイド: Magento

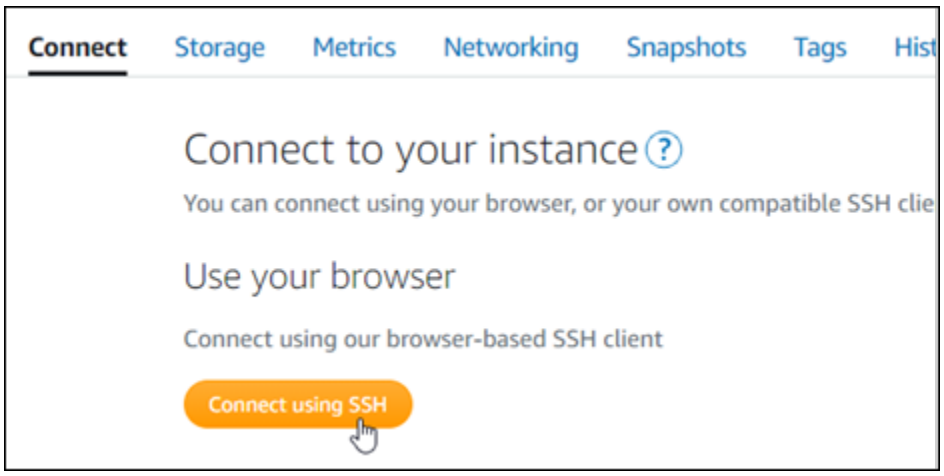
Amazon Lightsail で起動した Magento インスタンスの使用を開始するステップについて説明します。

目次

ステップ 1: Magento ウェブサイトのデフォルトのアプリケーションパスワードを取得する

次のステップを完了して、Magento ウェブサイトのデフォルトのアプリケーションパスワードを取得します。詳細については、「Amazon Lightsailの Bitnami インスタンス向けにアプリケーションのユーザー名とパスワードを取得する」を参照してください。

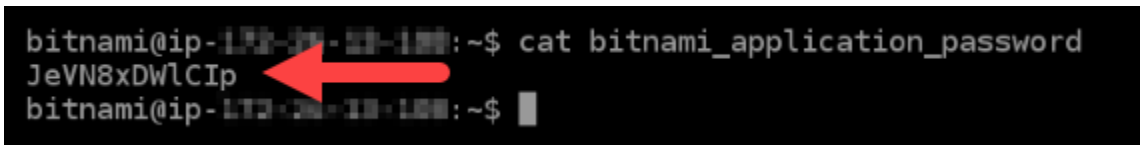
1. インスタンス管理ページの [Connect] (接続) タブで、[SSH を使用して接続] を選択します。



2. 接続後に、次のコマンドを入力してデフォルトのアプリケーションのパスワードを取得します。

```
cat $HOME/bitnami_application_password
```

アプリケーションのデフォルトパスワードを含んだ、次の例のようなレスポンスが表示されます。このパスワードを安全な場所に保存します。このチュートリアルの次のセクションで、Magento ウェブサイトの管理ダッシュボードにサインインする際に使用します。



ステップ 2: Magento インスタンスに静的 IP アドレスをアタッチする

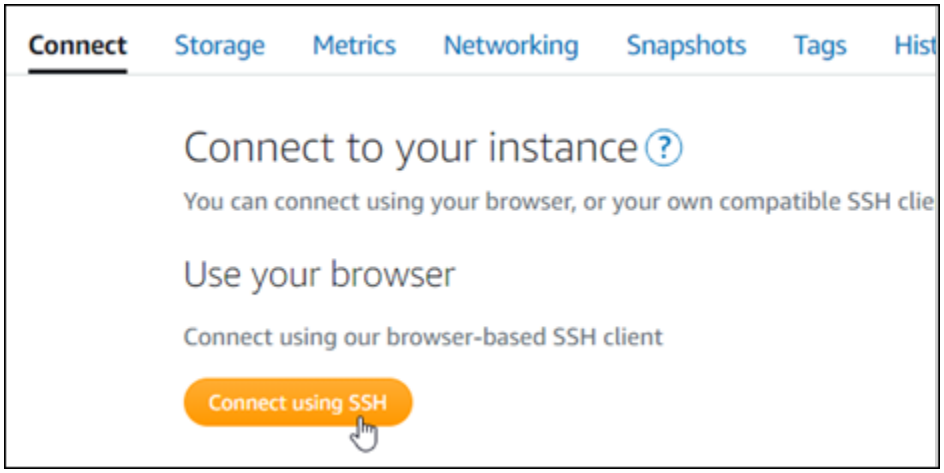
インスタンスを最初に作成した際に割り当てられたパブリック IP アドレスは、インスタンスを停止してスタートするたびに変更されます。パブリック IP アドレスが変更されないように、静的 IP アドレスを作成してインスタンスにアタッチする必要があります。それ以降、`example.com` などの登録したドメイン名をインスタンスで使用する際、毎回インスタンスを停止してスタートするたびにドメインの DNS レコードを更新する必要がなくなります。1 つの静的 IP を 1 つのインスタンスにアタッチできます。

インスタンス管理ページの [ネットワーク] タブで、[静的 IP の作成] または [静的 IP のアタッチ] (インスタンスにアタッチできる静的 IP を既に作成している場合) を選択して、ページの手順に従います。詳細については、「静的 IP を作成してインスタンスにアタッチする」を参照してください。

新しい静的 IP アドレスがインスタンスに添付されたら、次の手順を実行して、Magento ソフトウェアに新しい静的 IP アドレスを認識させる必要があります。

1. インスタンスの静的 IP アドレスは書き留めておきます。このIP アドレスはインスタンス管理ページの ヘッダーセクションに表示されます。

2. インスタンス管理ページの [Connect] (接続) タブで、[SSH を使用して接続] を選択します。



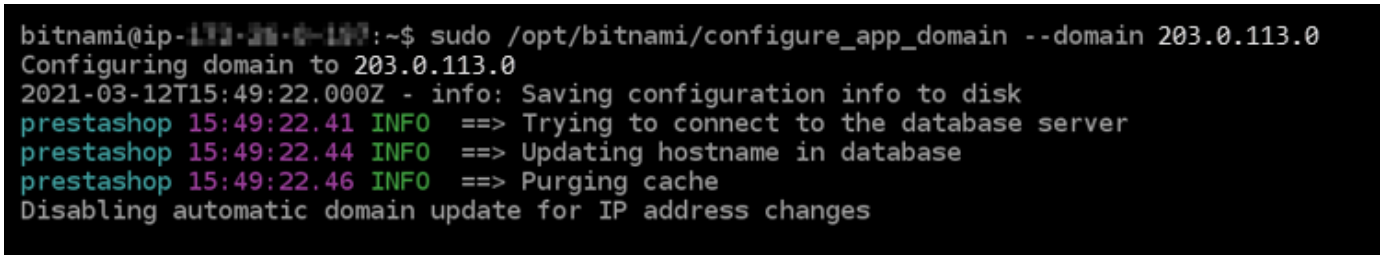
3. 接続後に、次のコマンドを入力します。*<StaticIP>* はインスタンスの新しい静的 IP アドレス必ず置き換えてください。

```
sudo /opt/bitnami/configure_app_domain --domain <StaticIP>
```

例:

```
sudo /opt/bitnami/configure_app_domain --domain 203.0.113.0
```

次の例のようなレスポンスが表示されます。これで、Magento ソフトウェアは新しい静的 IP アドレスを認識するようになります。



> **Note**
>
> Magento は現在、IPv6 アドレスをサポートしていません。インスタンスの IPv6 を有効にすることはできますが、Magento ソフトウェアは IPv6 ネットワーク経由のリクエストに応答しません。

ステップ 3: Magento ウェブサイトの管理ダッシュボードにサインインする

Magento ウェブサイトにアクセスして管理ダッシュボードにサインインするには、以下のステップを実行します。サインインするには、このガイドの前のセクションで取得したデフォルトのユーザー名 (`user`) とデフォルトのアプリケーションパスワードを使用します。

1. Lightsail コンソールで、インスタンス管理ページのヘッダー部分に記載されているパブリック IP アドレスないし静的 IP アドレスを書き留めます。

2. 以下のアドレスまで移動して、Magento ウェブサイトの管理ダッシュボードのサインインページにアクセスします。*PublicIPAddress* は、インスタンスのパブリック IP アドレスないし静的 IP アドレスに必ず置き換えます。

```
http://<InstanceIpAddress>/admin
```

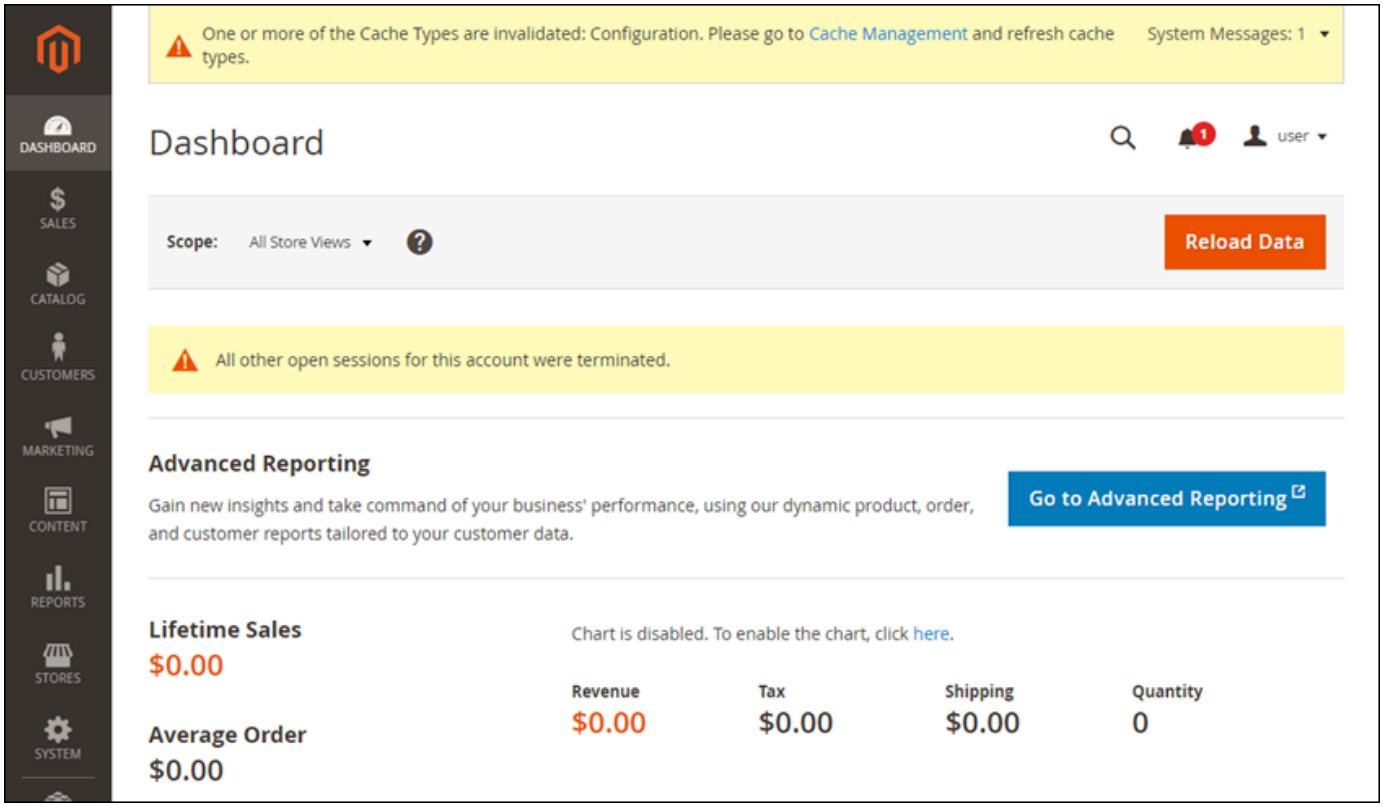
例:

```
http://203.0.113.0/admin
```

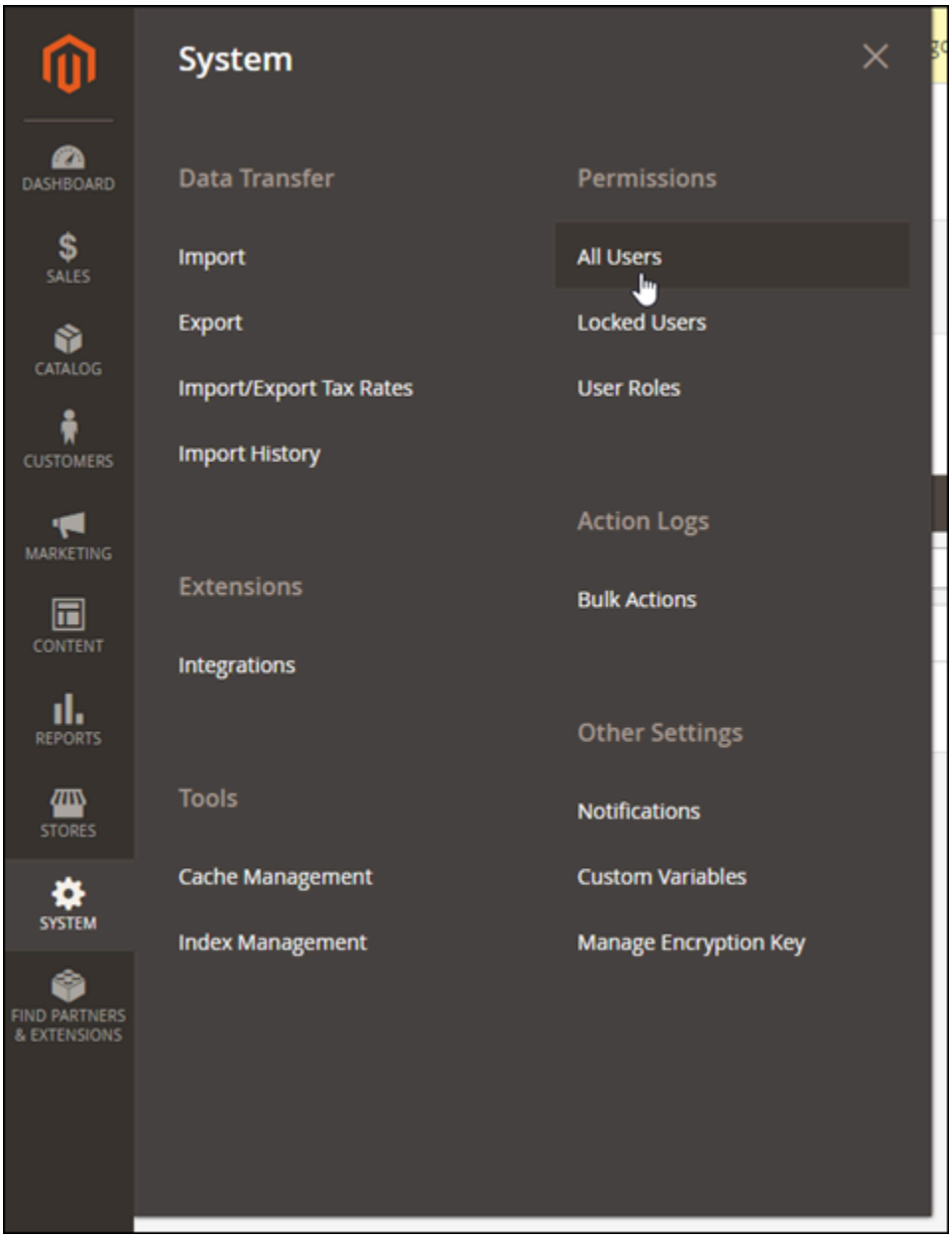
> **Note**
>
> Magento 管理用ダッシュボードのサインインページにアクセスできない場合、インスタンスを再起動する必要があるかもしれません。

3. デフォルトのユーザー名 (user) と、このガイドの前半のセクションで取得したデフォルトのアプリケーションパスワードを入力して [Sign in] (サインイン) を選択します。

Magento の管理ダッシュボードが表示されます。



Magento ウェブサイトの管理ダッシュボードへサインインする際に使用するデフォルトのユーザー名またはパスワードを変更するには、ナビゲーションペインの [System] (システム) を選択し、[All Users] (すべてのユーザー) を選択します。詳細については、Magento ドキュメントの「ユーザーの追加」を参照してください。



管理ダッシュボードの詳細については、「Magento 2.4 ユーザーガイド」を参照してください。

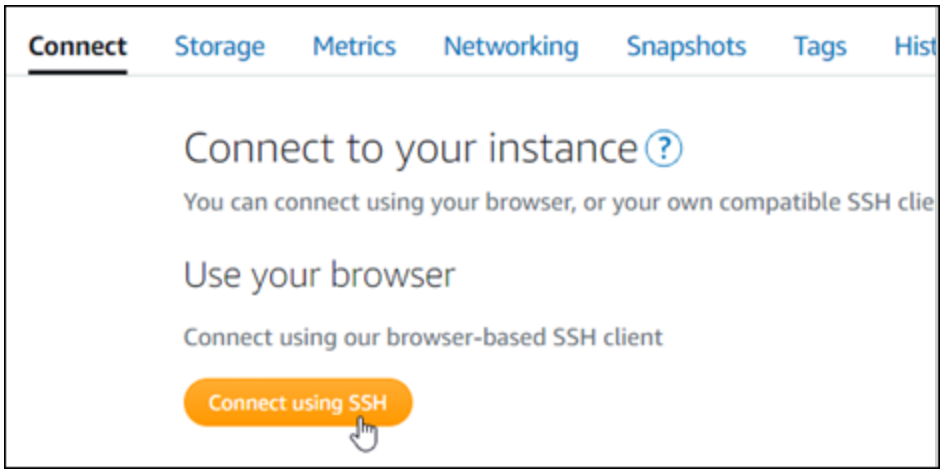
ステップ 4: 登録済みドメイン名へのトラフィックを Magento ウェブサイトに送信する

example.com などの登録済みドメイン名のトラフィックを Magento ウェブサイトに送信するには、ドメインのドメインネームシステム (DNS) にレコードを追加します。DNS レコードは、通常、ドメインの登録先であるレジストラが管理またはホストします。ただし、ドメインの DNS レコードの管理を Lightsail に引き渡して、Lightsail コンソールで管理できるようにすることをお勧めします。

Lightsail コンソールのホームページの [Domains & DNS] (ドメインと DNS) タブで、[Create DNS zone] (DNS ゾーンの作成) を選択し、ページに記載される手順に従います。詳細については、「Lightsail で DNS ゾーンを作成し、ドメインの DNS レコードを管理する」を参照してください。

ドメイン名へのトラフィックがインスタンスにルーティングされたら、次の手順を実行して、Magento ソフトウェアにドメイン名を認識させます。

1. インスタンス管理ページの [Connect] (接続) タブで、[SSH を使用して接続] を選択します。



2. 接続後に、次のコマンドを入力します。*<DomainName>* は、インスタンスにトラフィックを送信しているドメイン名に必ず置き換えてください。

```
sudo /opt/bitnami/configure_app_domain --domain <DomainName>
```

例:

```
sudo /opt/bitnami/configure_app_domain --domain www.example.com
```

次の例のようなレスポンスが表示されます。これで、Magento ソフトウェアはドメイン名を認識できるようになります。

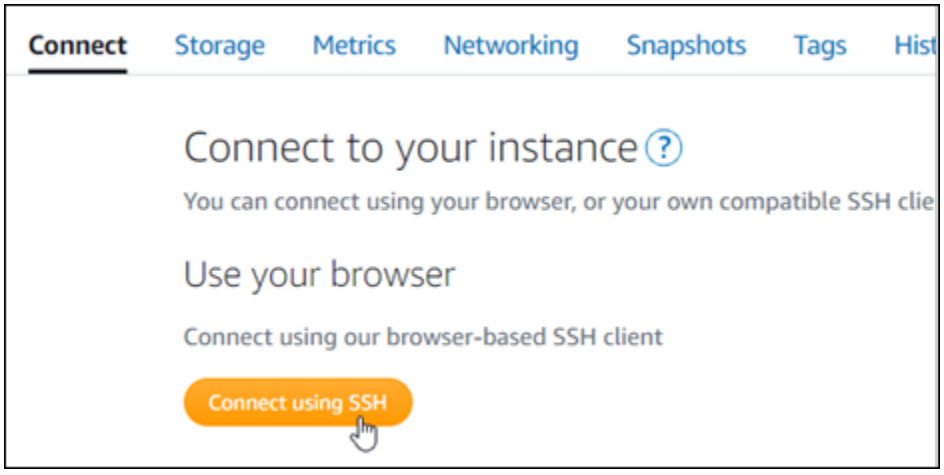
```
bitnami@ip-[redacted]:~$ sudo /opt/bitnami/configure_app_domain --domain www.example.com
Configuring domain to www.example.com
2021-03-12T15:49:22.000Z - info: Saving configuration info to disk
prestashop 15:49:22.41 INFO  ==> Trying to connect to the database server
prestashop 15:49:22.44 INFO  ==> Updating hostname in database
prestashop 15:49:22.46 INFO  ==> Purging cache
Disabling automatic domain update for IP address changes
```

ステップ 5: Magento ウェブサイトの HTTPS を設定する

Magento ウェブサイトで HTTPS を設定するには、以下の手順を実行します。次の手順では、Bitnami HTTPS 設定ツール (bncert) を使用する方法を示しています。このツールは、SSL/TLS 証明書のリクエスト、リダイレクトの設定 (例: HTTP から HTTPS)、および証明書の更新を行うためのコマンドラインツールです。

> **Important**
>
> bncert ツールは現在、 Magento インスタンスのパブリック IP アドレスにトラフィックをルーティングしているドメインに対してのみ証明書を発行します。これらの手順を開始する前に、Magento ウェブサイトで使用するすべてのドメインの DNS に、 DNS レコードが追加されていることを確認してください。

1. インスタンス管理ページの [Connect] (接続) タブで、[SSH を使用して接続] を選択します。



2. 接続後に、次のコマンドを入力して bncert-tool をスタートします。

```
sudo /opt/bitnami/bncert-tool
```

次の例のようなレスポンスが表示されます:

```
bitnami@ip-[redacted]:~$ sudo /opt/bitnami/bncert-tool
Warning: Custom redirections are not supported in the Bitnami Magento Stack.
This tool will not be able to enable/disable redirections.
Press [Enter] to continue:
```

3. 次の例に記載されているように、プライマリドメイン名と代替ドメイン名をスペースで区切って入力します。

```
------------------------------------------------------------------------------
Welcome to the Bitnami HTTPS Configuration tool.

------------------------------------------------------------------------------
Domains

Please provide a valid space-separated list of domains for which you wish to
configure your web server.

Domain list []: example.com www.example.com
```

4. これから実行される変更が一覧表示されます。Y と入力し、Enter を押して確認し、続行します。

```
------------------------------------------------------------------------------
Changes to perform

The following changes will be performed to your Bitnami installation:

1. Stop web server
2. Configure web server to use a free Let's Encrypt certificate for the domains:
 example.com www.example.com
3. Configure a cron job to automatically renew the certificate each month
4. Configure web server name to: example.com
5. Start web server once all changes have been performed



Do you agree to these changes? [Y/n]: Y
```

5. Let's Encrypt 証明書に関連付けるメールアドレスを入力し、Enter を押します。

```
Create a free HTTPS certificate with Let's Encrypt

Please provide a valid e-mail address for which to associate your Let's Encrypt
certificate.

Domain list: example.com www.example.com

Server name: example.com

E-mail address []:
```

6. Let's Encrypt サブスクライバー合意書を確認します。Y と入力し、Enter を押して契約に同意し、続行します。

```
The Let's Encrypt Subscriber Agreement can be found at:

https://letsencrypt.org/documents/LE-SA-v1.2-November-15-2017.pdf

Do you agree to the Let's Encrypt Subscriber Agreement? [Y/n]:
```

これらのアクションは、証明書のリクエストや指定したリダイレクトの設定など、インスタンスで HTTPS を有効にするために実行されます。

```
Performing changes to your installation

The Bitnami HTTPS Configuration Tool will perform any necessary actions to your
Bitnami installation. This may take some time, please be patient.
```

次の例のようなメッセージが表示された場合は、証明書は正常に発行され、検証され、インスタンスでリダイレクトが正常に設定されています。

```
Success

The Bitnami HTTPS Configuration Tool succeeded in modifying your installation.

The configuration report is shown below.

Backup files:
* /opt/bitnami/apache/conf/httpd.conf.back.202104052147
* /opt/bitnami/apache/conf/bitnami/bitnami.conf.back.202104052147
* /opt/bitnami/apache/conf/bitnami/bitnami-ssl.conf.back.202104052147
* /opt/bitnami/apache/conf/vhosts/magento-https-vhost.conf.back.202104052147
* /opt/bitnami/apache/conf/vhosts/magento-vhost.conf.back.202104052147

Find more details in the log file:

/tmp/bncert-202104052147.log

If you find any issues, please check Bitnami Support forums at:

https://community.bitnami.com

Press [Enter] to continue:

bitnami@ip-172-26-3-145:~$
```

bncert ツールは、有効期限が切れる前、 80 日ごとに証明書の自動更新を実行します。次の一連のステップに進み、Magento ウェブサイトでの HTTPS の有効化を完了します。

7. 以下のアドレスまで移動して、Magento ウェブサイトの管理ダッシュボードのサインインページにアクセスします。*<DomainName>* は、インスタンスにトラフィックを送信している登録済みドメイン名に必ず置き換えてください。

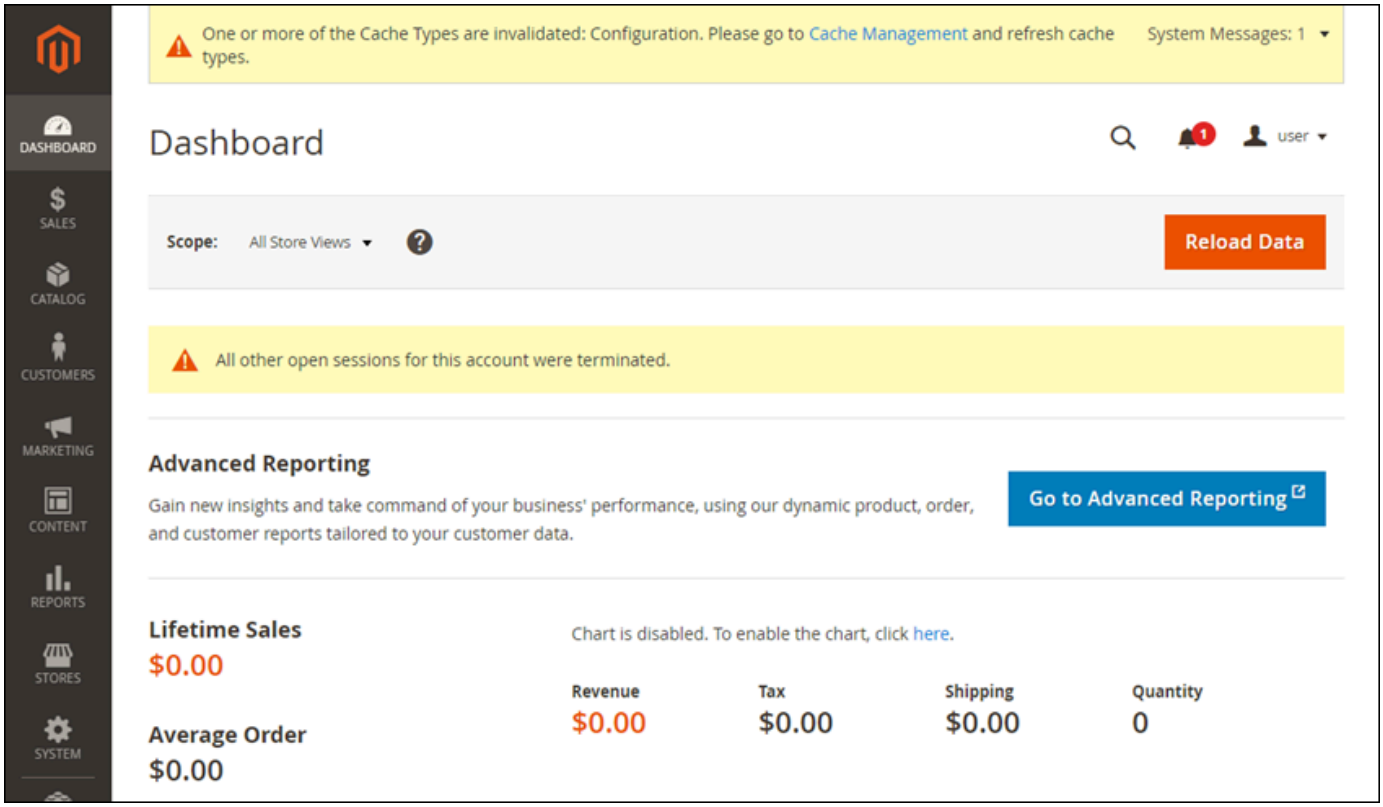
```
http://<DomainName>/admin
```

例:

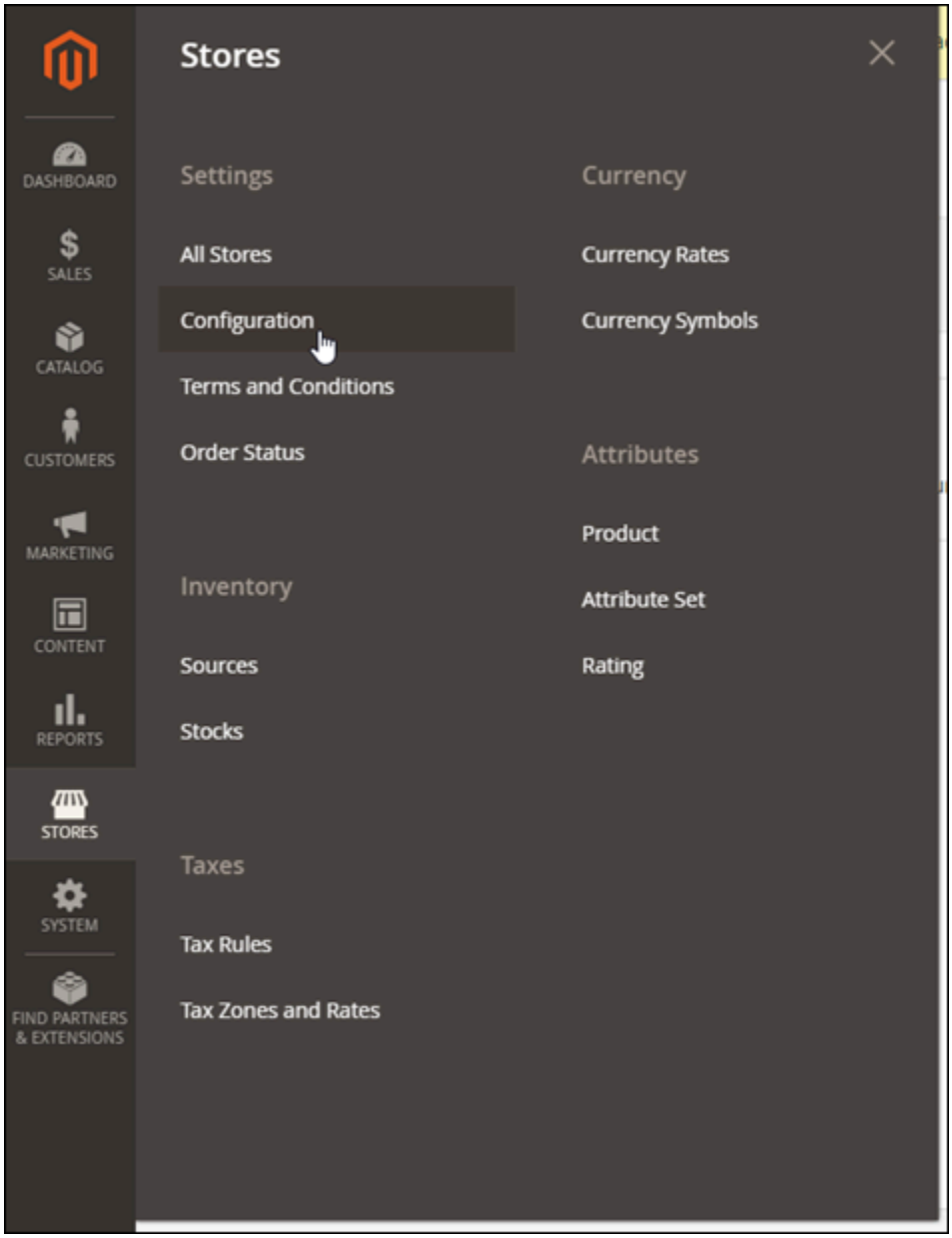
```
http://www.example.com/admin
```

8. デフォルトのユーザー名 (user) と、このガイドの前半のセクションで取得したデフォルトのアプリケーションパスワードを入力して [Sign in] (サインイン) を選択します。

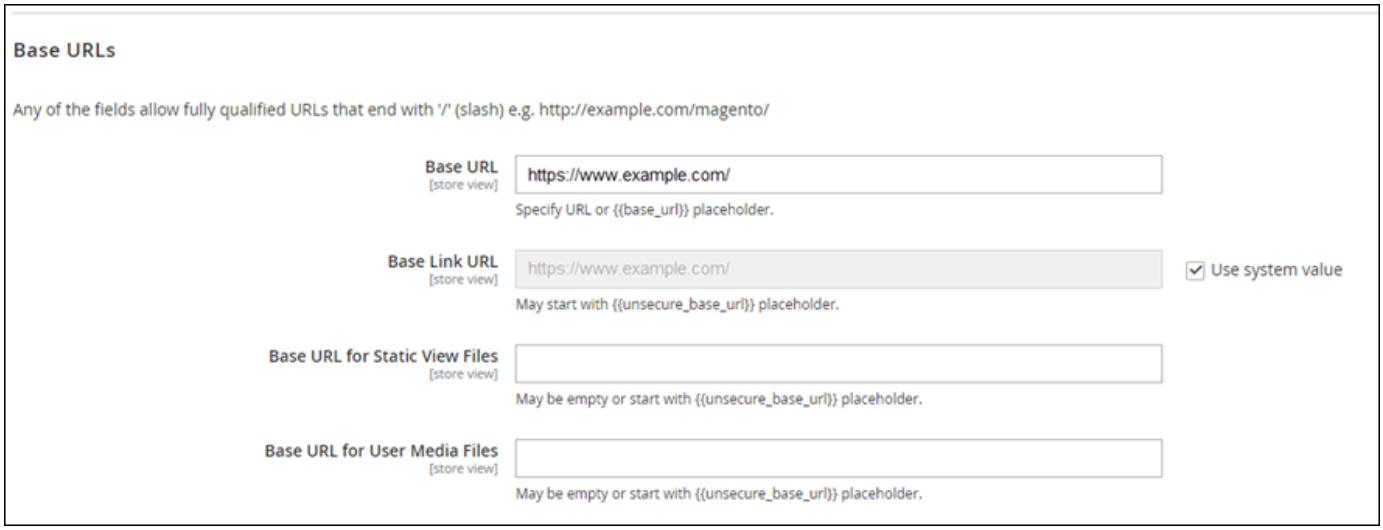
Magento の管理ダッシュボードが表示されます。



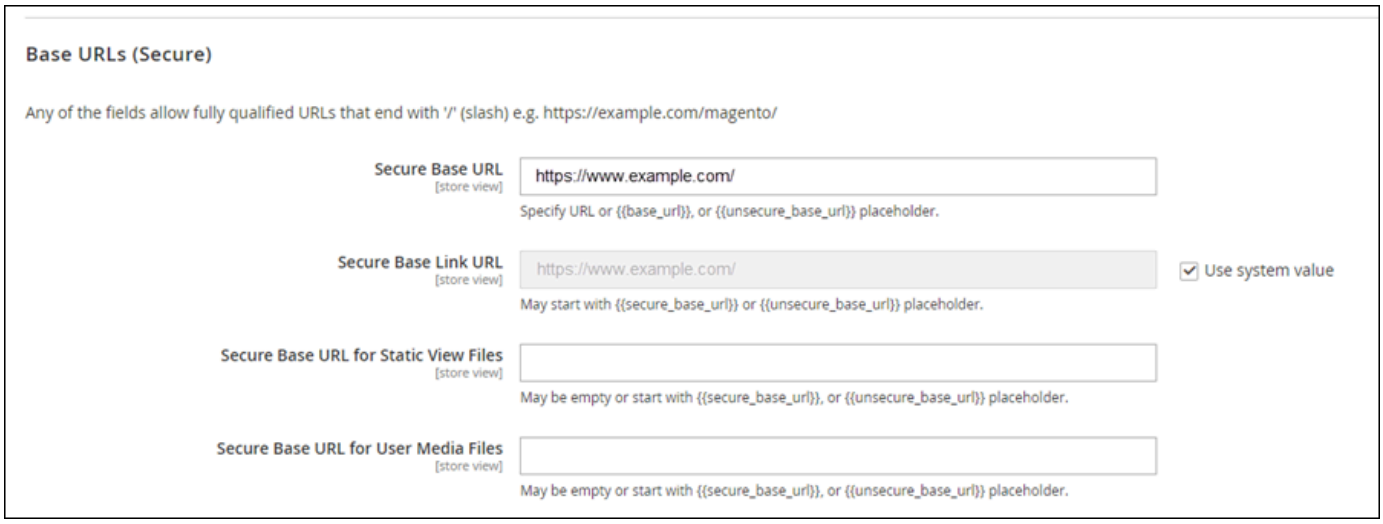
9. ナビゲーションペインで [Stores] (ストア)、[Configuration] (設定) の順に選択します。



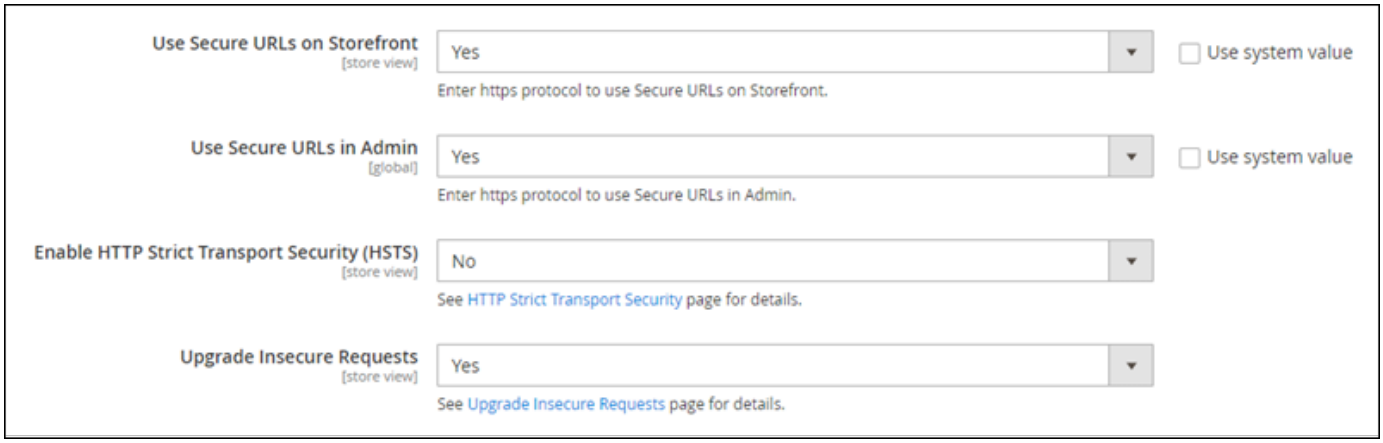
10. [Web] (ウェブ) を選択し、[Base URLs] (ベース URL) ノードを展開します。
11. [Base URLs] (ベース URL) テキストボックスに、ウェブサイトの完全な URL を入力します (例: `https://www.example.com/`)。



12. ベース URL (セキュア) ノードを展開します。
13. [Secure Base URL] (セキュアベース URL) テキストボックスに、ウェブサイトの完全な URL を入力します (例: `https://www.example.com/`)。



14. [Use Secure URLs on Storefront] (ストアフロントでセキュリティで保護された URL を使用する)、[Use Secure URLs in Admin] (管理者でセキュア URL を使用する)、および [Upgrade Insecure Requests] (安全でないリクエストをアップグレードする) で [Yes] (はい) を選択します。



15. ページの上部にある [Save Config] (設定の保存) を選択します。

    これで、Magento ウェブサイトに HTTPS が設定されました。ユーザーが HTTP バージョン (例: `http://www.example.com`) の Magento ウェブサイトを参照すると、ユーザーは自動的に HTTPS バージョン (例: `https://www.example.com`) にリダイレクトされます。

ステップ 6: メール通知用の SMTP を設定する

Magento ウェブサイトの SMTP 設定を構成して、メール通知を有効にします。詳細については、Bitnami ドキュメントの「Magento Magepal SMTP拡張機能をインストールする」を参照してください。

> **Important**
>
> ポート 25、465、または 587 を使用するように SMTP を設定する場合、Lightsail コンソールでインスタンスファイアウォールのポートを開ける必要があります。詳細については、「Amazon Lightsail でインスタンスファイアウォールルールの追加および編集」を参照してください。
>
> Gmail アカウントを設定して Magento ウェブサイトでメールを送信できるようにする場合は、Gmail のログインに使用する通常のパスワードではなく、アプリケーションのパスワードを使用する必要があります。詳細については、「アプリケーションのパスワードでサインイン」を参照してください。

ステップ 7: Bitnami と Magento のドキュメントを読む

Bitnami のドキュメントを読んで、Magento インスタンスとウェブサイト上でプラグインのインストールやテーマのカスタマイズなどの管理タスクを実行する方法を確認します。詳細について

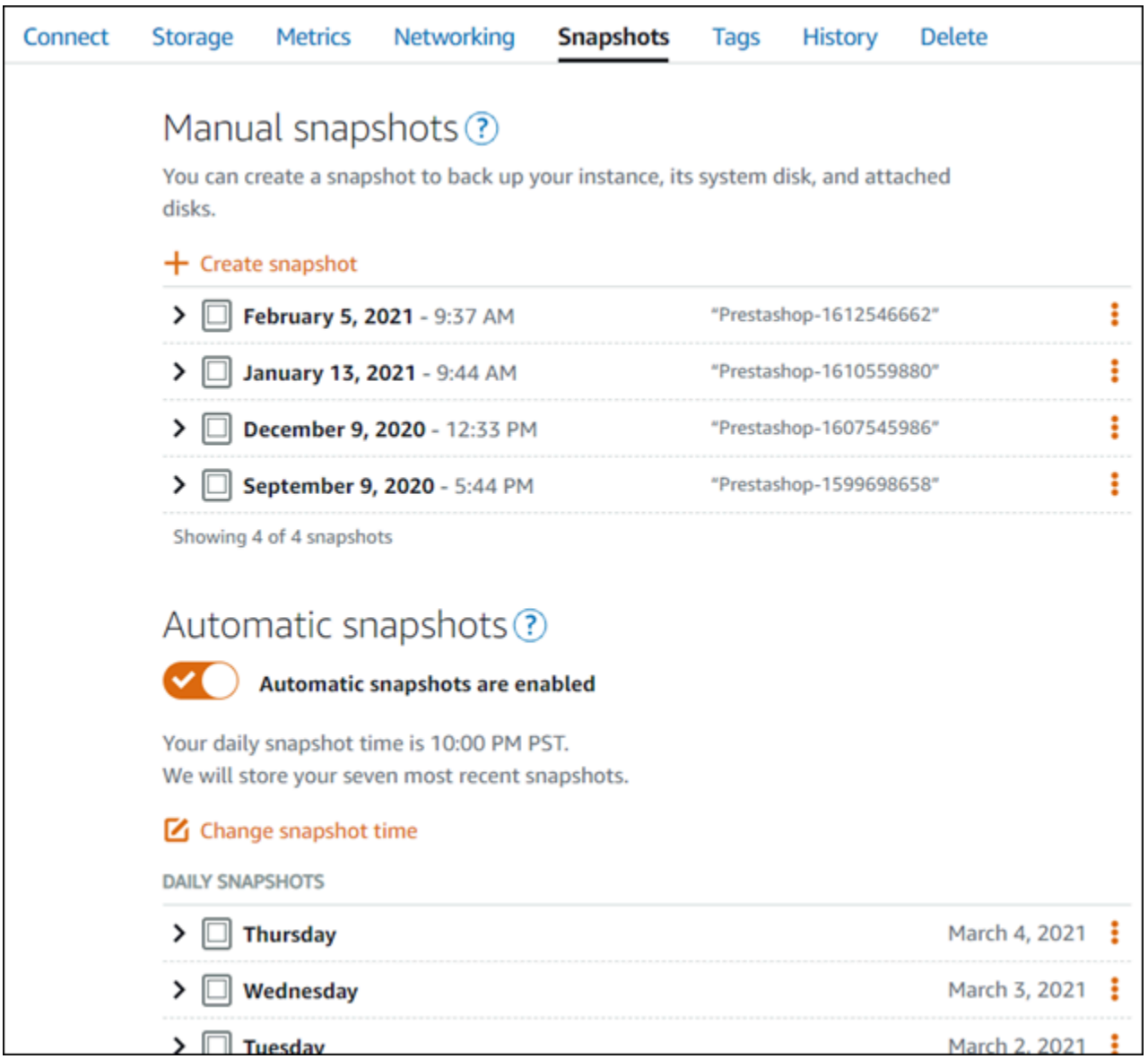
は、Bitnami ドキュメントの「AWS クラウド用の Bitnami PrestaShop スタック」を参照してください。

Magento のドキュメントを読んで、Magento のウェブサイトの管理方法も確認してください。詳細については、「Magento 2.4ユーザーガイド」を参照してください。

ステップ 8: Magento インスタンスのスナップショットを作成する

Magento ウェブサイトを希望どおりに設定したら、インスタンスの定期的なスナップショットを作成してバックアップします。スナップショットは手動で作成するか、自動スナップショットを有効にして Lightsail に毎日のスナップショットを作成させることができます。インスタンスに問題が発生した場合は、スナップショットを使用して新しい代替インスタンスを作成できます。詳細については、「スナップショット」を参照してください。

インスタンス管理ページの [スナップショット] タブで [スナップショットを作成する] を選択するか、[自動スナップショットを有効にする] を選択します。



詳細については、「Amazon Lightsail で Linux または Unix インスタンスのスナップショットを作成」および「Amazon Lightsail のインスタンスまたはディスクの自動スナップショットの有効化と無効化」を参照してください。

クイックスタートガイド: Nginx

Amazon Lightsail で起動した Nginx インスタンスの使用を開始するステップをいくつか示します。

ステップ 1: Nginx インスタンスのデフォルトのアプリケーションパスワードを取得する

インスタンスにプリインストールされているアプリケーションやサービスにアクセスするには、アプリケーションのデフォルトパスワードが必要です。

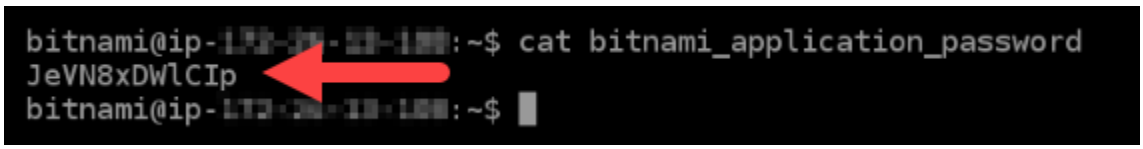
> **Important**
>
> Lightsail ブラウザベースの SSH/RDP クライアントは IPv4 トラフィックのみを受け入れます。サードパーティーのクライアントを使用して、IPv6 経由でインスタンスに SSH または RDP 接続します。詳細については、「インスタンスに接続します」を参照してください。

1. インスタンス管理ページの [接続] タブで、[SSH を使用して接続] を選択します。
2. 接続後に、次のコマンドを入力してデフォルトのアプリケーションのパスワードを取得します。

```
cat bitnami_application_password
```

> **Note**
>
> ユーザーのホームディレクトリ以外のディレクトリで作業している場合は、「`cat $HOME/bitnami_application_password`」と入力します。

次のようなレスポンスにアプリケーションのデフォルトパスワードが表示されます。



詳細については、「Amazon Lightsail での Bitnami インスタンスのアプリケーションユーザー名とパスワードの取得」を参照してください。

ステップ 2: Nginx インスタンスに静的 IP アドレスをアタッチする

インスタンスにアタッチしたデフォルトの動的なパブリック IP アドレスは、インスタンスを停止して開始するたびに変わります。パブリック IP アドレスが変わらないようにするには、静的 IP アドレスを作成してインスタンスにアタッチします。その後にドメイン名をインスタンスで使用すると、イ

ンスタンスを停止して開始するたびにドメインの DNS レコードを更新する必要がなくなります。1 つの静的 IP を 1 つのインスタンスにアタッチできます。

インスタンス管理ページの [Domains & DNS] (ドメインと DNS) タブで、[Create static IP] (静的 IP の作成) を選択し、ページに記載される手順に従います。

詳細については、「静的 IP を作成して Lightsail のインスタンスにアタッチする」を参照してください。

ステップ 3: Nginx インスタンスのウェルカムページにアクセスする

インスタンスのパブリック IP アドレスに移動して、インスタンスにインストールされているアプリケーションにアクセスするか、 にアクセスするか phpMyAdmin、Bitnami ドキュメントにアクセスします。

1. インスタンス管理ページの [接続] タブで、パブリック IP を書き留めます。
2. パブリック IP アドレスを参照します (例: `http://192.0.2.3` に移動します)。

詳細については、「Amazon Lightsail での Bitnami インスタンスのアプリケーションユーザー名とパスワードの取得」を参照してください。

ステップ 4: ドメイン名を Nginx インスタンスにマッピングする

ドメイン名 (`example.com` など) をインスタンスにマッピングするには、ドメインのドメインネームシステム (DNS) にレコードを追加します。DNS レコードは、通常、ドメインの登録先であるレジストラが管理またはホストします。ただし、Lightsail コンソールを使用して管理できるように、ドメインの DNS レコードの管理を Lightsail に転送することをお勧めします。

Lightsail コンソールのホームページの「ネットワーク」タブで「DNS ゾーンの作成」を選択し、ページの手順に従います。

詳細については、「DNS ゾーンを作成してドメインの DNS レコードを管理する」を参照してください。

ステップ 5: Bitnami のドキュメントを確認する

Bitnami のドキュメントで、Nginx アプリケーションのデプロイ、SSL 証明書による HTTPS サポートの有効化、SFTP を使用したファイルのサーバーへのアップロードなどの方法を確認します。

詳細については、「AWS クラウド 用の Bitnami Nginx」を参照してください。

ステップ 6: Nginx インスタンスのスナップショットを作成する

スナップショットは、インスタンスのシステムディスクおよびオリジナル設定のコピーです。スナップショットには、メモリ、CPU、ディスクサイズ、データ転送レートなどの情報が含まれています。スナップショットを、新しいインスタンスのベースラインまたはデータのバックアップとして使用できます。

インスタンス管理ページの [スナップショット] タブで、スナップショット名を入力して [スナップショットの作成] を選択します。

詳細については、「Linux または Unix インスタンスのスナップショットを作成する」を参照してください。

クイックスタートガイド: Node.js

Amazon Lightsail で起動した Node.js インスタンスの使用を開始するステップについて説明します。

ステップ 1: Node.js インスタンスのデフォルトのアプリケーションパスワードを取得する

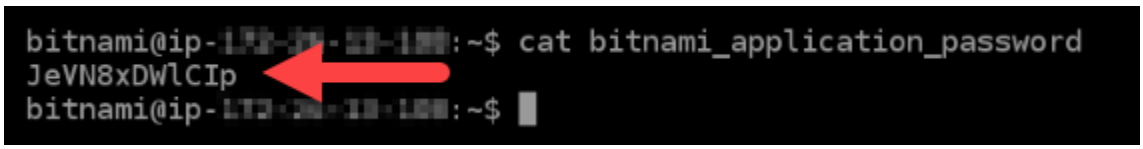
インスタンスにプリインストールされているアプリケーションやサービスにアクセスするには、アプリケーションのデフォルトパスワードが必要です。

1. インスタンス管理ページの [接続] タブで、[SSH を使用して接続] を選択します。
2. 接続後に、次のコマンドを入力してデフォルトのアプリケーションのパスワードを取得します。

```
cat bitnami_application_password
```

> **Note**
>
> ユーザーのホームディレクトリ以外のディレクトリで作業している場合は、「`cat $HOME/bitnami_application_password`」と入力します。

次のようなレスポンスにアプリケーションのデフォルトパスワードが表示されます。



詳細については、「Amazon Lightsailの Bitnami インスタンス向けにアプリケーションのユーザー名とパスワードを取得する」を参照してください。

ステップ 2: Node.js インスタンスに静的 IP アドレスをアタッチする

インスタンスにアタッチしたデフォルトの動的なパブリック IP アドレスは、インスタンスを停止して開始するたびに変わります。パブリック IP アドレスが変わらないようにするには、静的 IP アドレスを作成してインスタンスにアタッチします。その後にドメイン名をインスタンスで使用すると、インスタンスを停止して開始するたびにドメインの DNS レコードを更新する必要がなくなります。1 つの静的 IP を 1 つのインスタンスにアタッチできます。

インスタンス管理ページの [Domains & DNS] (ドメインと DNS) タブで、[Create static IP] (静的 IP の作成) を選択し、ページに記載される手順に従います。

詳細については、「静的 IP を作成して Lightsail インスタンスにアタッチする」を参照してください。

ステップ 3: Node.js インスタンスのウェルカムページにアクセスする

インスタンスのパブリック IP アドレスに移動してインスタンスにインストールされているアプリケーションにアクセスし、phpMyAdmin または Bitnami ドキュメントを参照します。

1. インスタンス管理ページの [接続] タブで、パブリック IP を書き留めます。
2. パブリック IP アドレスを参照します (例: `http://192.0.2.3` に移動します)。

詳細については、「Amazon Lightsailの Bitnami インスタンス向けにアプリケーションのユーザー名とパスワードを取得する」を参照してください。

ステップ 4: ドメイン名を Node.js インスタンスにマッピングする

ドメイン名 (`example.com` など) をインスタンスにマッピングするには、ドメインのドメインネームシステム (DNS) にレコードを追加します。DNS レコードは、通常、ドメインの登録先であるレジストラが管理またはホストします。ただし、ドメインの DNS レコードの管理を Lightsail に引き渡して、Lightsail コンソールで管理できるようにすることをお勧めします。

Lightsail コンソールのホームページで、[ネットワーキング] タブの [DNS ゾーンの作成] を選択し、ページに表示される手順に従います。

詳細については、「DNS ゾーンを作成してドメインの DNS レコードを管理する」を参照してください。

ステップ 5: Bitnami のドキュメントを確認する

Bitnami のドキュメントで、Node.js アプリケーションのデプロイ、SSL 証明書による HTTPS サポートの有効化、SFTP を使用したファイルのサーバーへのアップロードなどの方法を確認します。

詳細については、「AWS クラウド 用の Bitnami Node.js」を参照してください。

ステップ 6: Node.js インスタンスのスナップショットを作成する

スナップショットは、インスタンスのシステムディスクおよびオリジナル設定のコピーです。スナップショットには、メモリ、CPU、ディスクサイズ、データ転送レートなどの情報が含まれています。スナップショットを、新しいインスタンスのベースラインまたはデータのバックアップとして使用できます。

インスタンス管理ページの [スナップショット] タブで、スナップショット名を入力して [スナップショットの作成] を選択します。

詳細については、「Linux または Unix インスタンスのスナップショットを作成する」を参照してください。

クイックスタートガイド: Plesk

Plesk インスタンスが Amazon Lightsail で起動して実行されたら、開始するためのいくつかのステップを以下に示します。

> **Important**
>
> Plesk インスタンスの起動後に問題が発生した場合は、Plesk のサポートページにアクセスして、インスタンスにインストールする必要がある更新があるかどうかを確認します。詳細については、Plesk ヘルプセンターおよびPPA ドキュメントとヘルプポータルのPlesk アップデートを参照してください。

ステップ 1: Plesk インスタンスのワンタイムログイン URL を取得する

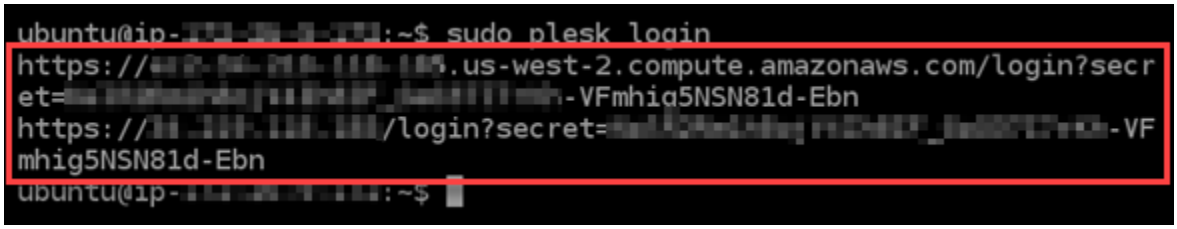
管理者として Plesk パネルにアクセスするには、ワンタイムログイン URL が必要です。

> **Important**
>
> Lightsail ブラウザベースの SSH/RDP クライアントは、IPv4 トラフィックのみを受け入れます。サードパーティーのクライアントを使用して、IPv6 経由でインスタンスに SSH または RDP 接続します。詳細については、「インスタンスに接続します」を参照してください。

1. インスタンス管理ページの [接続] タブで、[SSH を使用して接続] を選択します。
2. 接続後に、次のコマンドを入力してワンタイムログイン URL を取得します。

```
sudo plesk login | grep -v internal:8
```

ワンタイムログイン URL を含む以下の例のようなレスポンスが表示されます。



> **Important**
>
> 最近 Plesk インスタンスに静的 IP をアタッチした場合、古いパブリック IP アドレスを使用するワンタイムログイン URL を取得することがあります。インスタンスを再起動し、上のコマンドを再度実行して、新しい静的なパブリック IP アドレスを使用するワンタイムログイン URL を取得してください。

3. URL をクリップボードにコピーするか、書き留めます。後で Plesk パネルの初回サインイン時に必要になります。

詳細については、「Lightsail で Plesk をセットアップして設定する」を参照してください。

ステップ 2: 初めて Plesk パネルにサインインする

ワンタイムログイン URL をウェブブラウザに貼り付けます。ページの手順に従って Plesk のサインイン認証情報を作成します。初めてサインインするときに、Plesk にドメインを追加するオプションが表示されます。

> **Note**
>
> 接続がプライベートではないか、セキュリティで保護されていないか、またはセキュリティ上のリスクがあることを示すブラウザの警告が表示されることがあります。これは、Plesk インスタンスに SSL/TLS 証明書がまだ適用されていない場合に発生します。ブラウザウィンドウで、[Advanced] (詳細設定)、[Details] (詳細)、または [More information] (詳細情報) を選択して、使用可能なオプションを表示します。次に、プライベートまたは安全でない場合でも、ウェブサイトにアクセスすることを選択します。

詳細については、「Lightsail で Plesk をセットアップして設定する」を参照してください。

ステップ 3: Plesk インスタンスに静的 IP アドレスをアタッチする

インスタンスにアタッチしたデフォルトの動的なパブリック IP アドレスは、インスタンスを停止して開始するたびに変わります。パブリック IP アドレスが変わらないようにするには、静的 IP アドレスを作成してインスタンスにアタッチします。その後にドメイン名をインスタンスで使用すると、インスタンスを停止して開始するたびにドメインの DNS レコードを更新する必要がなくなります。1 つの静的 IP を 1 つのインスタンスにアタッチできます。

インスタンス管理ページで、[ネットワーキング] タブの [静的 IP の作成] を選択し、ページの手順に従います。

詳細については、「静的 IP を作成してインスタンスにアタッチする」を参照してください。

ステップ 4: ドメイン名を Plesk インスタンスにマッピングする

> **Note**
>
> Plesk インスタンスにドメインをマッピングして、Plesk パネルにアクセスできるようにします。Plesk パネル内で複数のドメインをマッピングすることもできます。これにより、Plesk パネル内でウェブサイトを管理できるようになります。このセクションでは、Plesk インスタンスにドメインをマッピングする方法について説明します。Plesk パネル内での、複数のドメインのマッピングについての詳細は、Plesk ドキュメントとヘルプポータルの「Plesk でドメインを追加」を参照してください。

ドメイン名 (`example.com` など) をインスタンスにマッピングするには、ドメインのドメインネームシステム (DNS) にレコードを追加します。DNS レコードは、通常、ドメインの登録先であるレジ

ストラが管理またはホストします。ただし、Lightsail コンソールを使用して管理できるように、ドメインの DNS レコードの管理を Lightsail に転送することをお勧めします。

Lightsail コンソールのホームページの「ドメインと DNS」タブで「DNS ゾーンの作成」を選択し、ページの手順に従います。

詳細については、「Lightsail でドメインの DNS レコードを管理する DNS ゾーンの作成」を参照してください。

ステップ 5: PLesk のドキュメントを確認する

Plesk のドキュメントで、Plesk を使用したウェブサイトの管理、Plesk パネルのカスタマイズなどの方法を確認します。

詳細については、Plesk ドキュメントとヘルプポータルの「Plesk ウェブサイト管理の開始」を参照してください。

ステップ 6: Plesk インスタンスのスナップショットを作成する

スナップショットは、インスタンスのシステムディスクおよびオリジナル設定のコピーです。スナップショットには、メモリ、CPU、ディスクサイズ、データ転送レートなどの情報が含まれています。スナップショットを、新しいインスタンスのベースラインまたはデータのバックアップとして使用できます。

インスタンス管理ページの [スナップショット] タブで、スナップショット名を入力して [スナップショットの作成] を選択します。

詳細については、「Linux または Unix インスタンスのスナップショットを作成する」を参照してください。

クイックスタートガイド: PrestaShop

Amazon Lightsail PrestaShop でインスタンスを起動して実行した後に、開始するために完了する必要があるいくつかのステップを次に示します。

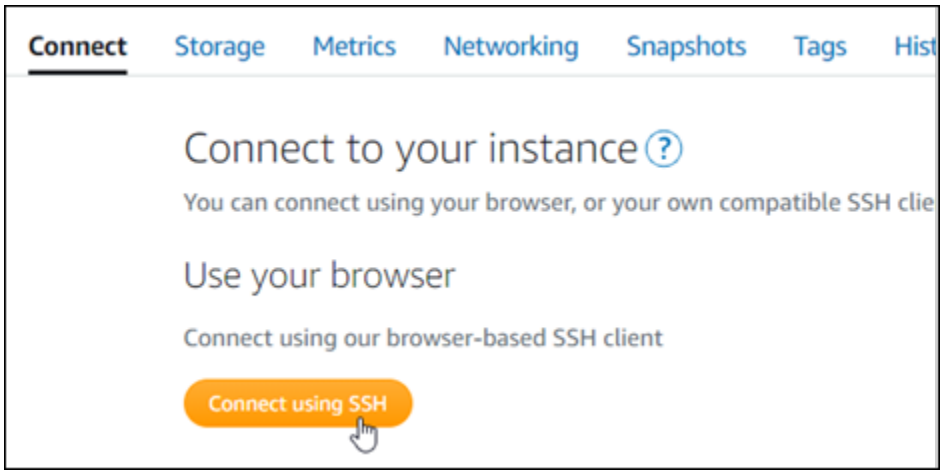
目次

- ステップ 5: PrestaShop ウェブサイトに HTTPS を設定する
- ステップ 6: メール通知用の SMTP を設定する
- ステップ 7: Bitnami とドキュメントを読む PrestaShop
- ステップ 8: インスタンスのスナップショットを作成する PrestaShop

ステップ 1: PrestaShop ウェブサイトのデフォルトアプリケーションパスワードを取得する

PrestaShopWeb サイトのデフォルトアプリケーションパスワードを取得するには、次の手順を実行します。

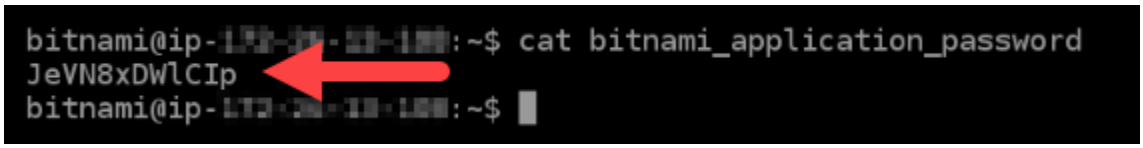
1. インスタンス管理ページの [Connect] (接続) タブで、[SSH を使用して接続] を選択します。



2. 接続後に、次のコマンドを入力してデフォルトのアプリケーションのパスワードを取得します。

```
cat $HOME/bitnami_application_password
```

アプリケーションのデフォルトパスワードを含んだ、次の例のようなレスポンスが表示されます。このパスワードを安全な場所に保存します。このチュートリアルの次のセクションでは、これを使用してウェブサイトの管理ダッシュボードにログインします PrestaShop。



詳細については、「Amazon Lightsail での Bitnami インスタンスのアプリケーションユーザー名とパスワードの取得」を参照してください。

ステップ 2: 静的 IP アドレスをインスタンスにアタッチする PrestaShop

インスタンスを最初に作成した際に割り当てられたパブリック IP アドレスは、インスタンスを停止してスタートするたびに変更されます。パブリック IP アドレスが変更されないように、静的 IP アドレスを作成してインスタンスにアタッチする必要があります。それ以降、`example.com` などの登録したドメイン名をインスタンスで使用する際、毎回インスタンスを停止してスタートするたびにドメインの DNS レコードを更新する必要がなくなります。1 つの静的 IP を 1 つのインスタンスにアタッチできます。

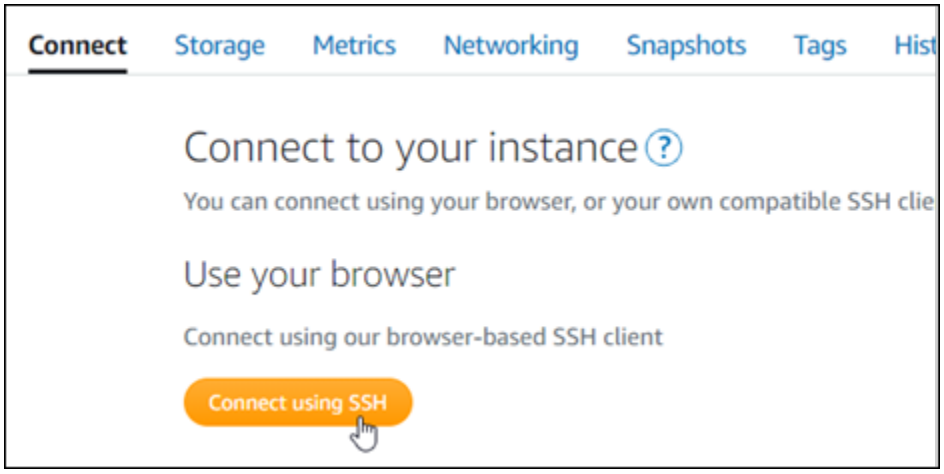
インスタンス管理ページの [ネットワーク] タブで、[静的 IP の作成] または [静的 IP のアタッチ] (インスタンスにアタッチできる静的 IP を既に作成している場合) を選択して、ページの手順に従います。

詳細については、「静的 IP を作成してインスタンスにアタッチする」を参照してください。

新しい静的 IP アドレスがインスタンスにアタッチされたら、次の手順を実行して、 PrestaShop ソフトウェアに新しい静的 IP アドレスを認識させる必要があります。

1. インスタンスの静的 IP アドレスは書き留めておきます。このIP アドレスはインスタンス管理ページの ヘッダーセクションに表示されます。

2. インスタンス管理ページの [Connect] (接続) タブで、[SSH を使用して接続] を選択します。



3. 接続後に、次のコマンドを入力します。*<StaticIP>* はインスタンスの新しい静的 IP アドレス必ず置き換えてください。

```
sudo /opt/bitnami/configure_app_domain --domain <StaticIP>
```

例:

```
sudo /opt/bitnami/configure_app_domain --domain 203.0.113.0
```

次の例のようなレスポンスが表示されます。これで、 PrestaShop ソフトウェアは新しい静的 IP アドレスを認識しているはずです。

```
bitnami@ip-[redacted]:~$ sudo /opt/bitnami/configure_app_domain --domain 203.0.113.0
Configuring domain to 203.0.113.0
2021-03-12T15:49:22.000Z - info: Saving configuration info to disk
prestashop 15:49:22.41 INFO  ==> Trying to connect to the database server
prestashop 15:49:22.44 INFO  ==> Updating hostname in database
prestashop 15:49:22.46 INFO  ==> Purging cache
Disabling automatic domain update for IP address changes
```

> **Note**
>
> PrestaShop 現在 IPv6 アドレスはサポートされていません。インスタンスでは IPv6 を有効にできますが、 PrestaShop ソフトウェアは IPv6 ネットワーク経由のリクエストには応答しません。

ステップ 3: PrestaShop ウェブサイトの管理ダッシュボードにサインインします。

次の手順を実行して PrestaShop Web サイトにアクセスし、その管理ダッシュボードにログインします。サインインするには、このガイドの前のセクションで取得したデフォルトのユーザー名 (`user@example.com`) とデフォルトのアプリケーションパスワードを使用します。

1. Lightsail コンソールで、インスタンス管理ページのヘッダー領域にリストされているパブリック IP アドレスまたは静的 IP アドレスをメモします。

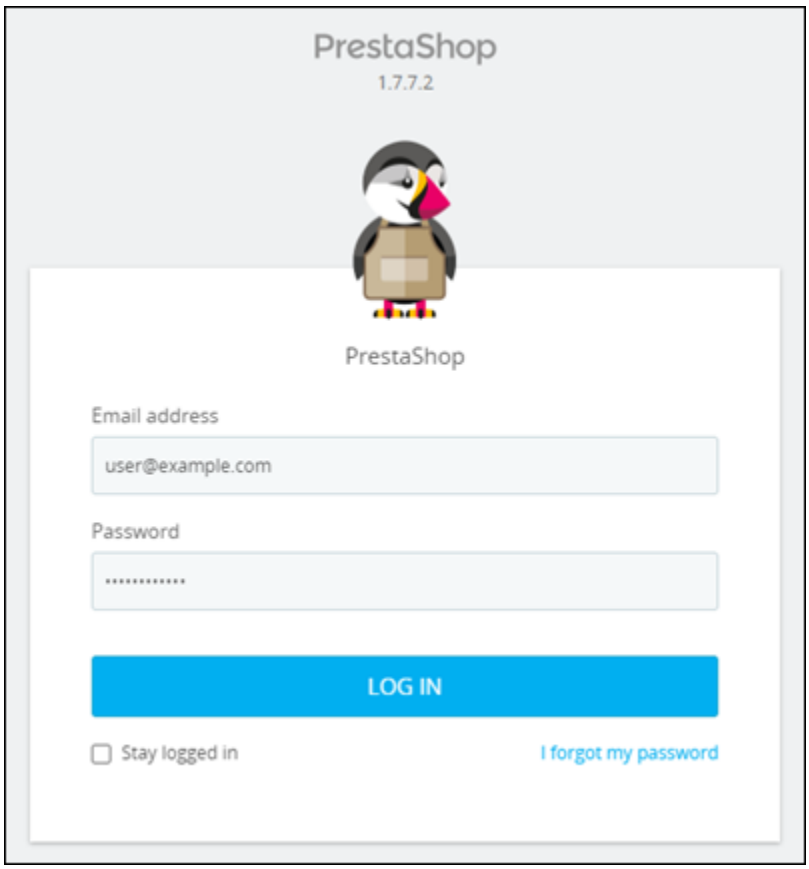
2. 次のアドレスを参照して、ウェブサイトの管理ダッシュボードのサインインページにアクセスします PrestaShop 。*< InstanceIpAddress >* は必ず、インスタンスのパブリック IP アドレスまたは静的 IP アドレスに置き換えてください。

```
http://<InstanceIpAddress>/administration
```

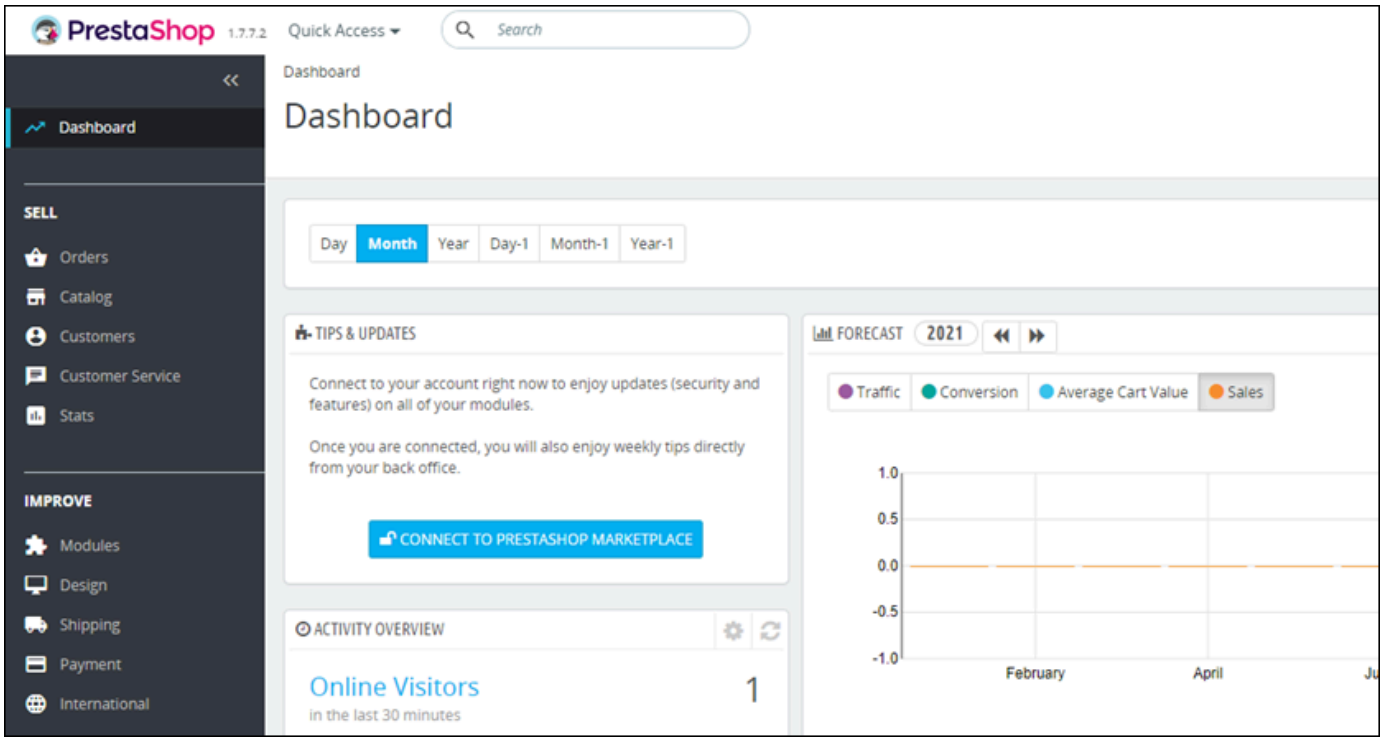
例:

```
http://203.0.113.0/administration
```

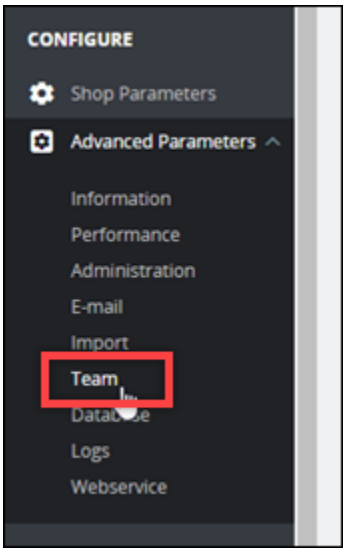
3. デフォルトのユーザー名 (`user@example.com`) と、このガイドの前のセクションで取得したデフォルトのアプリケーションパスワードを入力して [ログイン] を選択します。



PrestaShop 管理ダッシュボードが表示されます。



PrestaShop Web サイトの管理ダッシュボードへのサインインに使用するデフォルトのユーザー名またはパスワードを変更するには、ナビゲーションペインで [詳細パラメータ] を選択し、[チーム] を選択します。詳しくは、 PrestaShopPrestaShop ドキュメントの「ユーザーガイド」を参照してください。



管理ダッシュボードについて詳しくは、 PrestaShopPrestaShop ドキュメントの「ユーザーガイド」を参照してください。

ステップ 4: PrestaShop 登録したドメイン名のトラフィックをウェブサイトにルーティングする

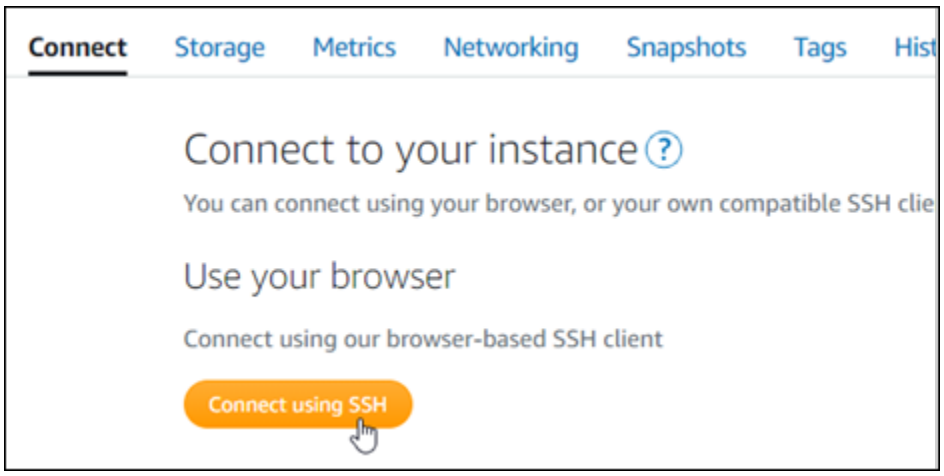
登録したドメイン名 (など`example.com`) PrestaShop のトラフィックをウェブサイトにルーティングするには、ドメインのドメインネームシステム (DNS) にレコードを追加します。DNS レコードは、通常、ドメインの登録先であるレジストラが管理またはホストします。ただし、ドメインの DNS レコードの管理を Lightsail に移管して、Lightsail コンソールを使用して管理できるようにすることをお勧めします。

Lightsail コンソールのホームページの [ドメインと DNS] タブで、[DNS ゾーンの作成] を選択し、ページの指示に従います。

詳細については、「Lightsail でドメインの DNS レコードを管理するための DNS ゾーンの作成」を参照してください。

ドメイン名がトラフィックをインスタンスにルーティングしたら、 PrestaShop 次の手順を実行してソフトウェアにドメイン名を認識させる必要があります。

1. インスタンス管理ページの [Connect] (接続) タブで、[SSH を使用して接続] を選択します。



2. 接続後に、次のコマンドを入力します。*< DomainName >* は必ず、トラフィックをインスタンスにルーティングするドメイン名に置き換えてください。

```
sudo /opt/bitnami/configure_app_domain --domain <DomainName>
```

例:

```
sudo /opt/bitnami/configure_app_domain --domain www.example.com
```

次の例のようなレスポンスが表示されます。これで、 PrestaShop ソフトウェアはドメイン名を認識できるはずです。

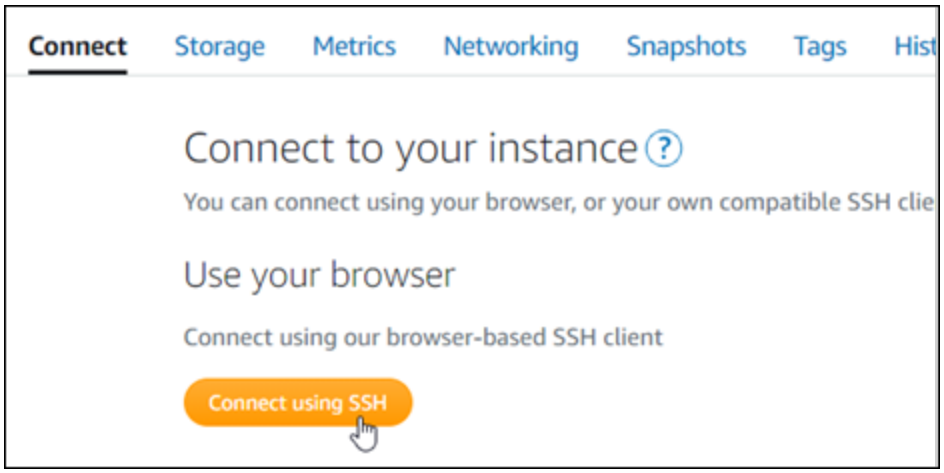
```
bitnami@ip-[redacted]:~$ sudo /opt/bitnami/configure_app_domain --domain www.example.com
Configuring domain to www.example.com
2021-03-12T15:49:22.000Z - info: Saving configuration info to disk
prestashop 15:49:22.41 INFO  ==> Trying to connect to the database server
prestashop 15:49:22.44 INFO  ==> Updating hostname in database
prestashop 15:49:22.46 INFO  ==> Purging cache
Disabling automatic domain update for IP address changes
```

ステップ 5: PrestaShop ウェブサイトに HTTPS を設定する

次の手順を実行して、 PrestaShop ウェブサイトに HTTPS を設定します。次の手順では、Bitnami HTTPS 設定ツール (bncert) を使用する方法を示しています。このツールは、SSL/TLS 証明書のリクエスト、リダイレクトの設定 (例: HTTP から HTTPS)、および証明書の更新を行うためのコマンドラインツールです。

> **Important**
>
> bncert ツールは、現在トラフィックをインスタンスのパブリック IP アドレスにルーティングしているドメインに対してのみ証明書を発行します PrestaShop 。これらの手順を開始する前に、ウェブサイトで使用したいすべてのドメインの DNS に DNS レコードを追加してください PrestaShop 。

1. インスタンス管理ページの [Connect] (接続) タブで、[SSH を使用して接続] を選択します。



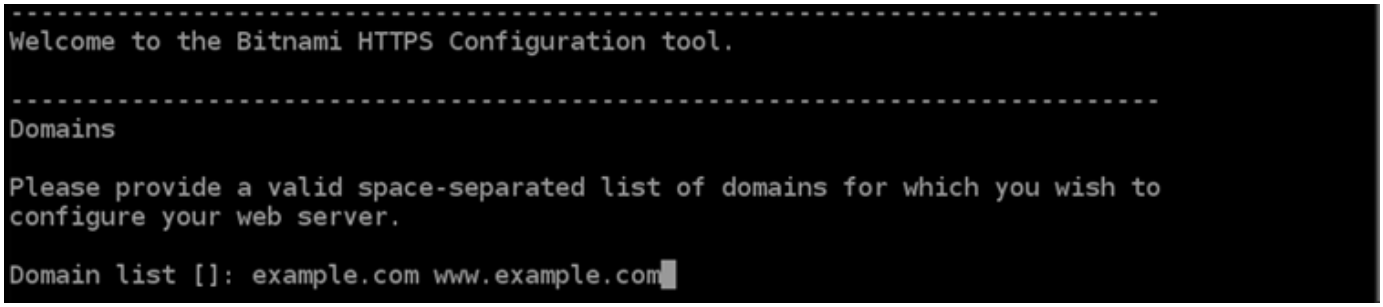
2. 接続後に、次のコマンドを入力して bncert-tool をスタートします。

```
sudo /opt/bitnami/bncert-tool
```

次の例のようなレスポンスが表示されます:

```
bitnami@ip-[redacted]:~$ sudo /opt/bitnami/bncert-tool
--------------------------------------------------------------------------
Welcome to the Bitnami HTTPS Configuration tool.

--------------------------------------------------------------------------
Domains

Please provide a valid space-separated list of domains for which you wish to
configure your web server.

Domain list []:
```

3. 次の例に記載されているように、プライマリドメイン名と代替ドメイン名をスペースで区切って入力します。



4. bncert ツールは、ウェブサイトのリダイレクトをどのように設定したいかをユーザーに確認します。使用できるオプションは、次のとおりです。

   - HTTP から HTTPS へのリダイレクトを有効にする - HTTP バージョンのウェブサイトを閲覧するユーザー (例: `http:/example.com`) を自動的に HTTPS バージョン (例: `https://example.com`) にリダイレクトするかどうかを決定します。すべての訪問者が暗号化された接続を使用するように強制されるため、このオプションを有効にすることをお勧めします。Y を入力して Enter を押すと、有効になります。
   - www なしから www ありへのリダイレクトの有効化 - ドメインの頂点 (例: `https://example.com`) まで閲覧するユーザー を自動的にドメインの www サブドメイン (例: `https://www.example.com`) にリダイレクトするかを指定します。このオプションを有効にすることをお勧めします。ただし、ドメインの頂点を Google のウェブマスターツールなどの検索エンジンツールで希望のウェブサイトアドレスとして指定した場合、または頂点が IP を直接指しており、www のサブドメインが CNAME レコードを介してリファレンスしている場合は、無効にして代替オプションを有効にすることをお勧めします (www ありから www なしへのリダイレクトを有効化) 。Y を入力し、Enter を押して有効にします。
   - www ありから www なしへのリダイレクトを有効にする - ドメインの www サブドメイン (例: `https://www.example.com`) まで閲覧するユーザーを、自動的にドメインの頂点 (例: `https://example.com`) にリダイレクトするかを指定します。www なしから www ありへのリダイレクトを有効にした場合は、これを無効にすることをお勧めします。N を入力し、Enter を押して無効にします。

   選択した結果は次の例のようになります。

```
Enable/disable redirections

Please select the redirections you wish to enable or disable on your Bitnami
installation.



Enable HTTP to HTTPS redirection [Y/n]: Y



Enable non-www to www redirection [Y/n]: Y



Enable www to non-www redirection [y/N]: N
```

5. これから実行される変更が一覧表示されます。Y と入力し、Enter を押して確認し、続行します。

```
Changes to perform

The following changes will be performed to your Bitnami installation:

1. Stop web server
2. Configure web server to use a free Let's Encrypt certificate for the domains:
example.com www.example.com
3. Configure a cron job to automatically renew the certificate each month
4. Configure web server name to: example.com
5. Enable HTTP to HTTPS redirection (example: redirect http://example.com to
https://example.com)
6. Enable non-www to www redirection (example: redirect example.com to
 www.example.com)
7. Start web server once all changes have been performed



Do you agree to these changes? [Y/n]: Y
```

6. Let's Encrypt 証明書に関連付けるメールアドレスを入力し、Enter を押します。

```
Create a free HTTPS certificate with Let's Encrypt

Please provide a valid e-mail address for which to associate your Let's Encrypt
certificate.

Domain list: example.com www.example.com

Server name: example.com

E-mail address []:
```

7. Let's Encrypt サブスクライバー合意書を確認します。Y と入力し、Enter を押して契約に同意し、続行します。

```
The Let's Encrypt Subscriber Agreement can be found at:

https://letsencrypt.org/documents/LE-SA-v1.2-November-15-2017.pdf

Do you agree to the Let's Encrypt Subscriber Agreement? [Y/n]:
```

これらのアクションは、証明書のリクエストや指定したリダイレクトの設定など、インスタンスで HTTPS を有効にするために実行されます。

```
Performing changes to your installation

The Bitnami HTTPS Configuration Tool will perform any necessary actions to your
Bitnami installation. This may take some time, please be patient.
```

次の例のようなメッセージが表示された場合は、証明書は正常に発行され、検証され、インスタンスでリダイレクトが正常に設定されています。

```
Success

The Bitnami HTTPS Configuration Tool succeeded in modifying your installation.

The configuration report is shown below.

Backup files:
* /opt/bitnami/apache2/conf/httpd.conf.back.202005290035
* /opt/bitnami/apache2/conf/bitnami/bitnami-apps-prefix.conf.back.202005290035
* /opt/bitnami/apache2/conf/bitnami/bitnami.conf.back.202005290035

Find more details in the log file:

/tmp/bncert-202005290035.log

If you find any issues, please check Bitnami Support forums at:

https://community.bitnami.com

Press [Enter] to continue:
```

bncert ツールは、有効期限が切れる前、 80 日ごとに証明書の自動更新を実行します。次の手順に進み、ウェブサイトでの HTTPS の有効化を完了します。 PrestaShop

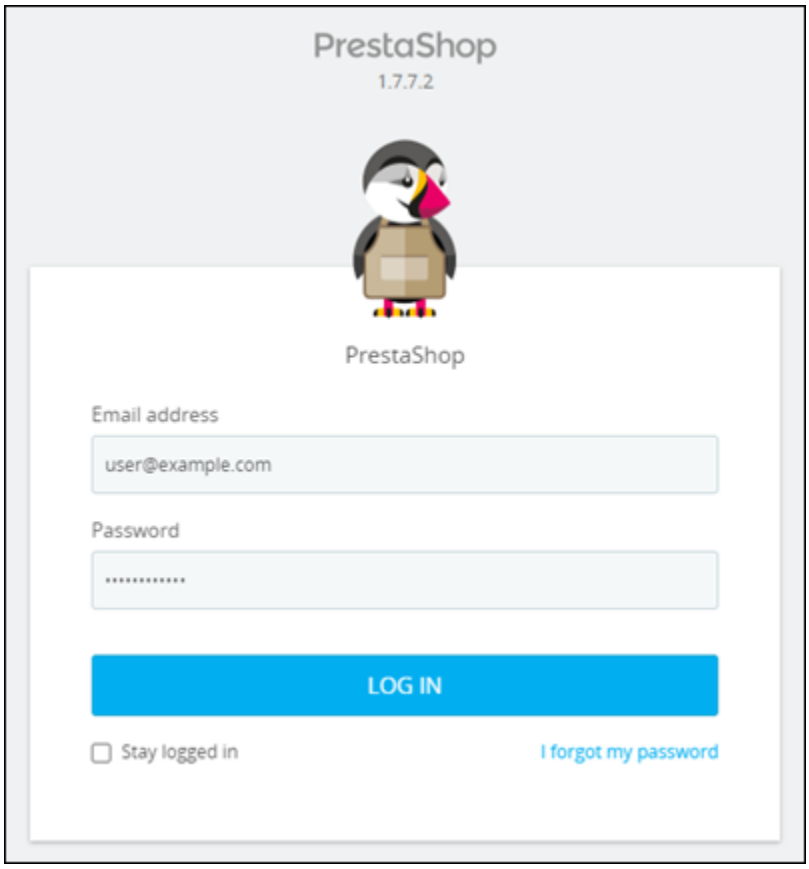
8. 次のアドレスを参照して、 PrestaShop Web サイトの管理ダッシュボードのサインインページにアクセスします。*< DomainName >* は必ず、トラフィックをインスタンスにルーティングする登録済みドメイン名に置き換えてください。

```
http://<DomainName>/administration
```

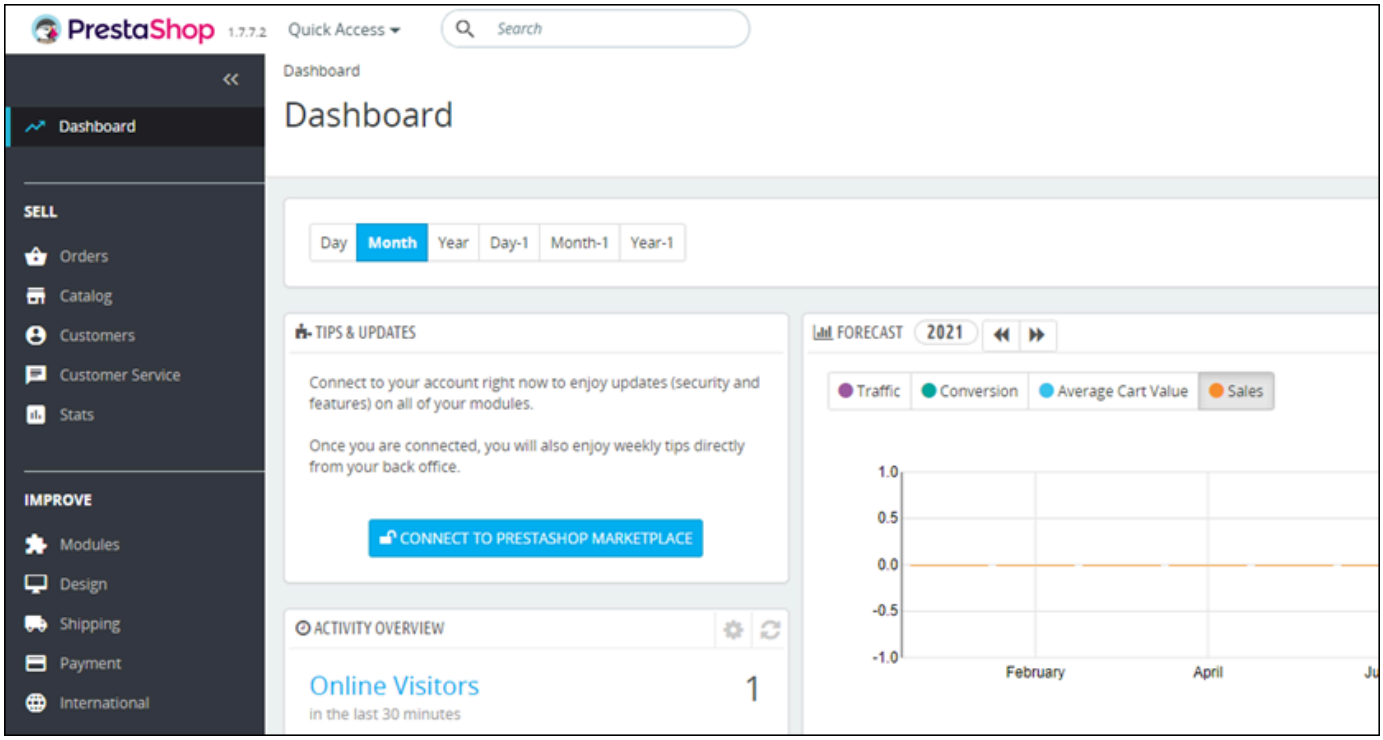
例:

```
http://www.example.com/administration
```

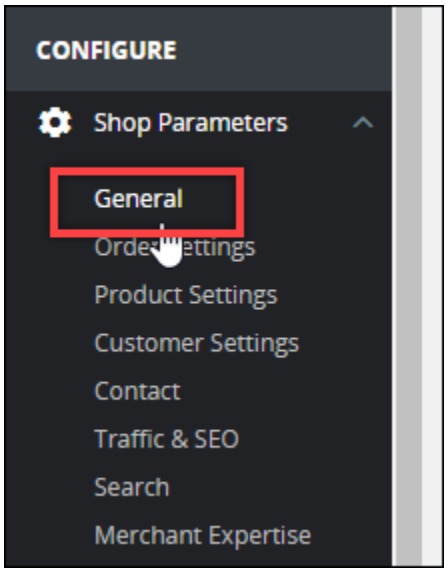
9. デフォルトのユーザー名 (user@example.com) と、このガイドの前のセクションで取得したデフォルトのアプリケーションパスワードを入力して [ログイン] を選択します。



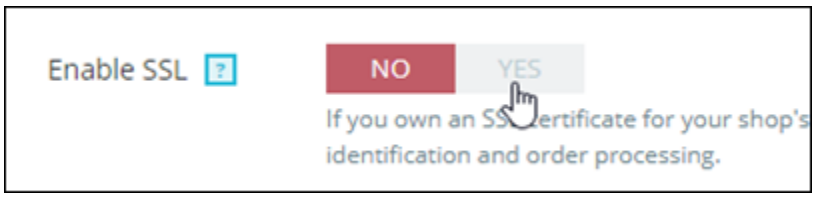
PrestaShop 管理ダッシュボードが表示されます。



10. ナビゲーションペインの [ショップパラメータ] を選択して、次に [General] (一般) を選択します。



11. [SSL を有効にする] の横にある [はい] を選択します。



12. ページの下部までスクロールし、[保存] を選択します。

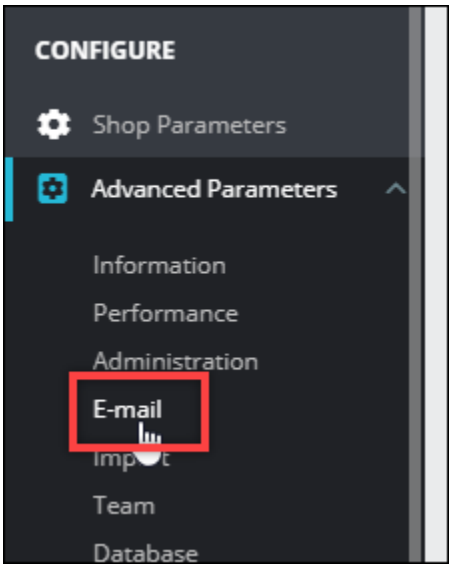
13. [General] ページが再読み込みしたら、[すべてのページでSSLを有効にする] の横にある [はい] を選択します。

14. ページの下部までスクロールし、[保存] を選択します。

    これで、 PrestaShop ウェブサイトに HTTPS が設定されました。 PrestaShop お客様がウェブサイトの HTTP バージョン (など`http://www.example.com`) を閲覧すると、自動的に HTTPS バージョン (など`https://www.example.com`) にリダイレクトされます。

ステップ 6: メール通知用の SMTP を設定する

PrestaShop ウェブサイトの SMTP 設定を構成して、メール通知を有効にします。そのためには、PrestaShop Web サイトの管理ダッシュボードにログインします。ナビゲーションペインの [アドバンストパラメータ] を選択して、[E メール] を選択します。E メールの連絡先もこれに応じて調整する必要があります。ナビゲーションペインの [Shop Parameters] (ショップパラメータ) をクリックしてから、[Contact] (連絡先) を選択します。



詳細については、ドキュメントの「ユーザーガイド PrestaShop」と Bitnami PrestaShop ドキュメントの「送信メール用の SMTP の設定」を参照してください。

> **Important**
>
> ポート 25、465、または 587 を使用するように SMTP を設定する場合、Lightsail コンソールのインスタンスのファイアウォールでそれらのポートを開く必要があります。詳細については、「Amazon Lightsail でのインスタンスファイアウォールルールの追加と編集」を参照してください。
> PrestaShop ウェブサイトで E メールを送信するように Gmail アカウントを設定する場合は、Gmail へのログインに使用する標準パスワードの代わりにアプリパスワードを使用する必要があります。詳細については、「アプリケーションのパスワードでサインイン」を参照してください。

ステップ 7: Bitnami とドキュメントを読む PrestaShop

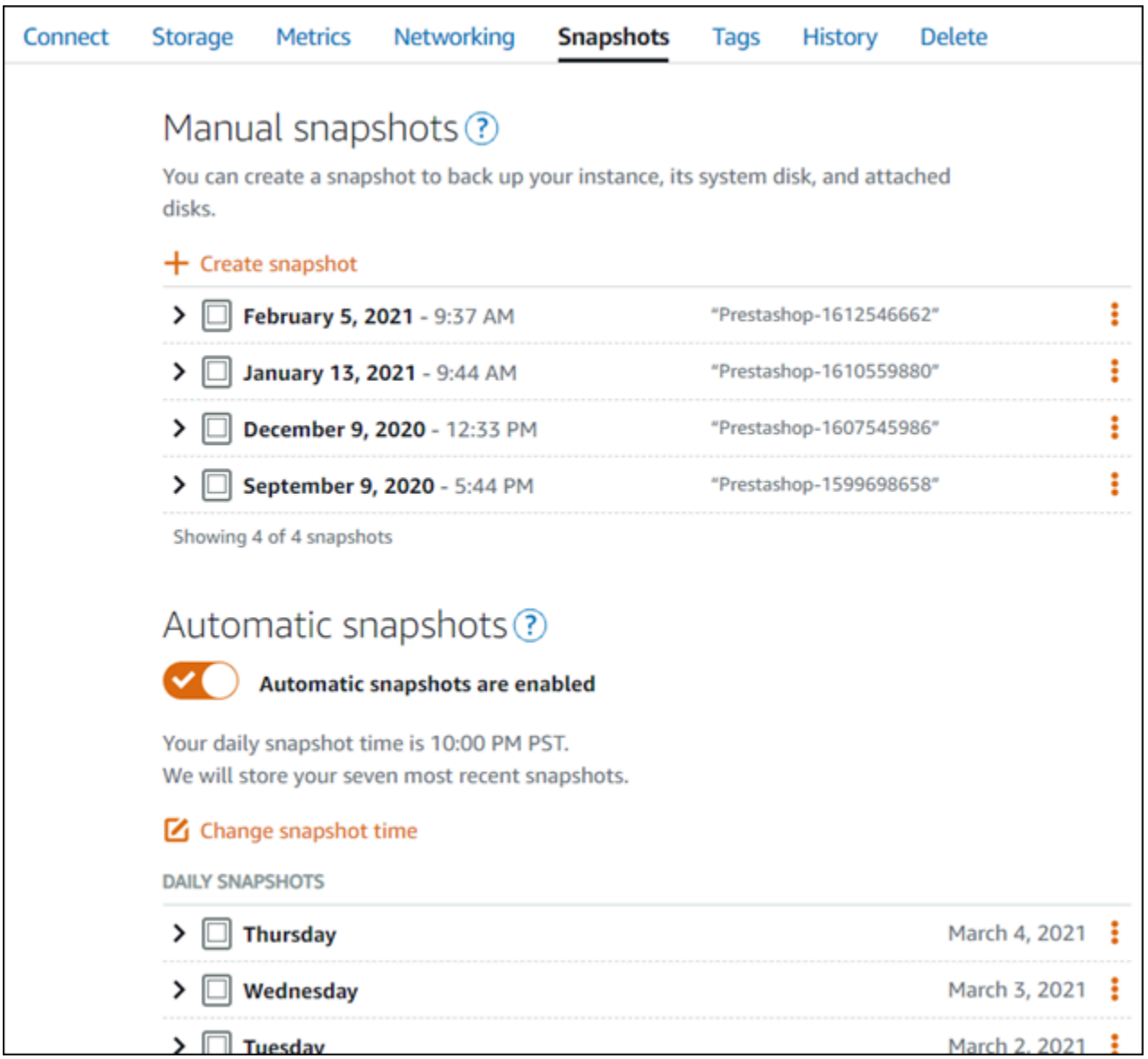
Bitnami のドキュメントを読んで、プラグインのインストールやテーマのカスタマイズなど、PrestaShop インスタンスと Web サイトの管理タスクを実行する方法を確認してください。詳細については、Bitnami ドキュメントの「AWS クラウド用 Bitnami PrestaShop スタック」を参照してください。

また、 PrestaShop ドキュメントを読んでウェブサイトの管理方法を学ぶ必要があります。PrestaShop 詳細については、 PrestaShopPrestaShop ドキュメントの『ユーザーガイド』を参照してください。

ステップ 8: PrestaShop インスタンスのスナップショットを作成する

PrestaShop ウェブサイトを希望どおりに設定したら、インスタンスのスナップショットを定期的に作成してバックアップします。スナップショットは手動で作成することも、自動スナップショットを有効にして Lightsail に毎日スナップショットを作成させることもできます。インスタンスに問題が発生した場合は、スナップショットを使用して新しい代替インスタンスを作成できます。詳細については、「スナップショット」を参照してください。

インスタンス管理ページの [スナップショット] タブで [スナップショットを作成する] を選択するか、[自動スナップショットを有効にする] を選択します。



詳細については、「Amazon Lightsail での Linux または Unix インスタンスのスナップショットの作成」または「Amazon Lightsail でのインスタンスまたはディスクの自動スナップショットの有効化または無効化」を参照してください。

クイックスタートガイド: Redmine

Amazon Lightsail で起動した Redmine インスタンスの使用を開始するステップについて説明します。

目次

- ステップ 2: Redmine の管理ダッシュボードにアクセスするため、デフォルトのアプリケーションパスワードを取得する
- ステップ 3: インスタンスに静的 IP アドレスをアタッチする
- ステップ 4: Redmine ウェブサイトの管理ダッシュボードにサインインする
- ステップ 5: 登録済みドメイン名へのトラフィックを Redmine ウェブサイトに送信する
- ステップ 6: Redmine ウェブサイトの HTTPS を設定する
- ステップ 7: Redmine のドキュメントを読み、引き続きウェブサイトの設定を続行する
- ステップ 8: インスタンスのスナップショットを作成する

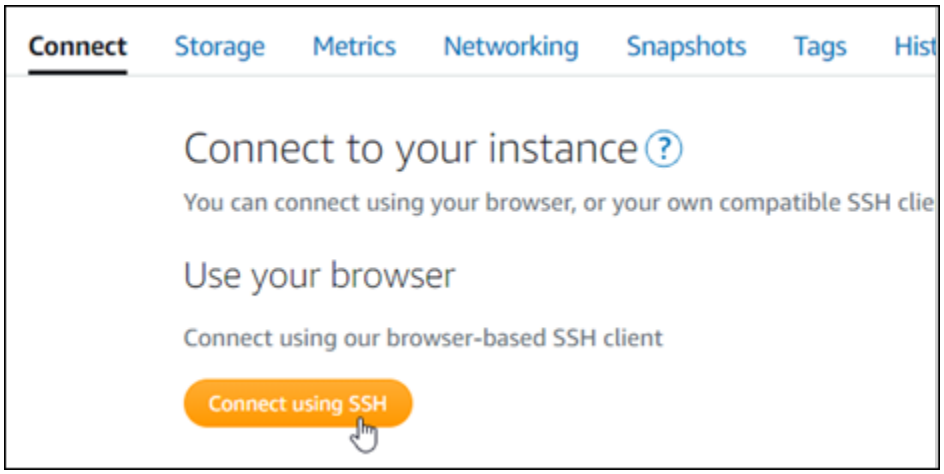
ステップ 1: Bitnami のドキュメントを確認する

Bitnami のドキュメントを読み、Redmine アプリケーションの設定方法については確認します。詳細については、「AWS クラウド 用に Bitnami がパッケージ化した Redmine」を参照してください。

ステップ 2: Redmine の管理ダッシュボードにアクセスするため、デフォルトのアプリケーションパスワードを取得する

次の手順を完了して、Redmine ウェブサイトの管理ダッシュボードにアクセスする際に必要となるデフォルトのアプリケーションパスワードを取得します。詳細については、「Amazon LightsailのBitnami インスタンス向けにアプリケーションのユーザー名とパスワードを取得する」を参照してください。

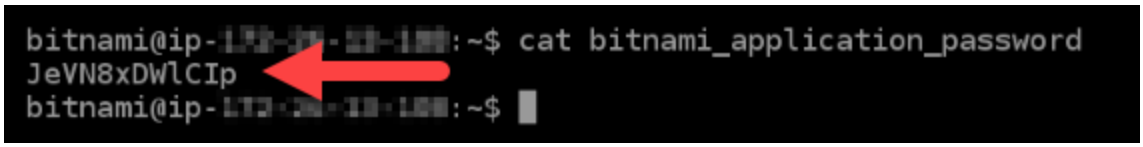
1. インスタンス管理ページの [接続] タブで、[SSH を使用して接続] を選択します。



2. 接続後に、次のコマンドを入力してアプリケーションのパスワードを取得します。

```
cat $HOME/bitnami_application_password
```

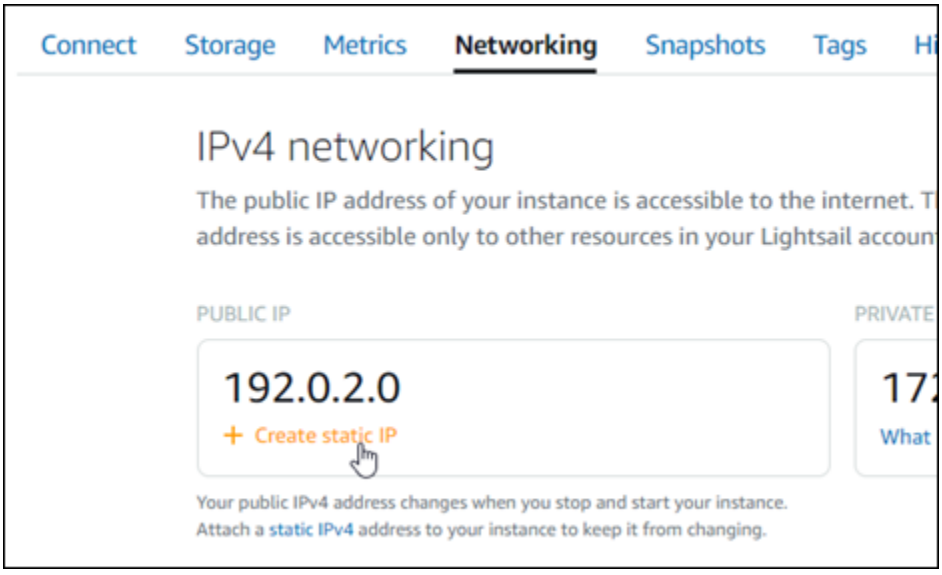
アプリケーションのデフォルトパスワードを含んだ、次の例のようなレスポンスが表示されます。



ステップ 3: インスタンスに静的 IP アドレスをアタッチする

インスタンスを最初に作成した際に割り当てられたパブリック IP アドレスは、インスタンスを停止してスタートするたびに変更されます。パブリック IP アドレスが変更されないように、静的 IP アドレスを作成してインスタンスにアタッチする必要があります。それ以降、example.com などの登録したドメイン名をインスタンスで使用する際、毎回インスタンスを停止してスタートするたびにドメインの DNS レコードを更新する必要がなくなります。1 つの静的 IP を 1 つのインスタンスにアタッチできます。

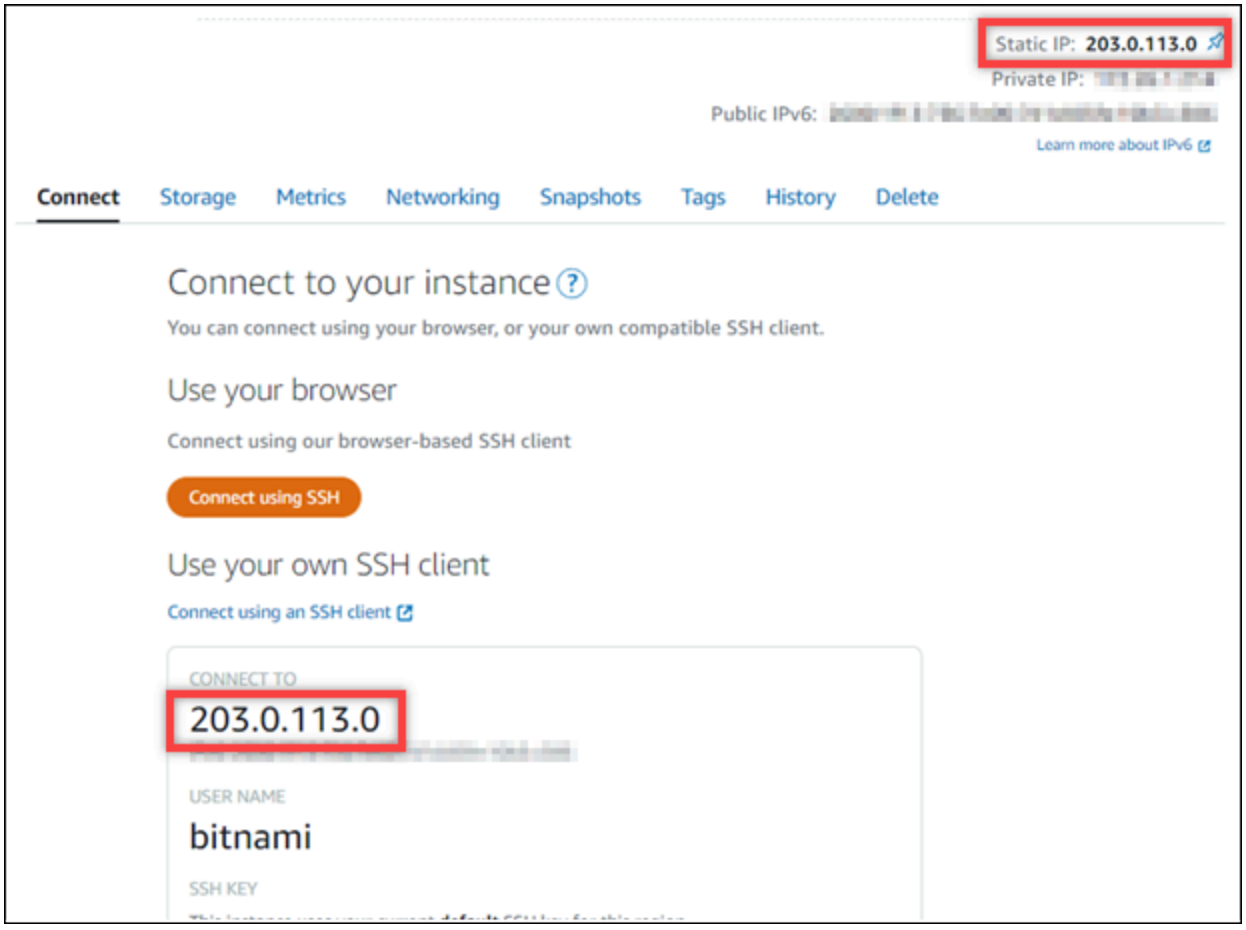
インスタンス管理ページの [ネットワーク] タブで、[静的 IP の作成] または [静的 IP のアタッチ] (インスタンスにアタッチできる静的 IP を既に作成している場合) を選択して、ページの手順に従います。詳細については、「静的 IP を作成してインスタンスにアタッチする」を参照してください。



ステップ 4: Redmine ウェブサイトの管理ダッシュボードにサインインする

デフォルトのユーザーパスワードを取得したら、以下の手順に従ってRedmine ウェブサイトのホームページに移動し、管理ダッシュボードにサインインします。サインイン後に、ウェブサイトをカスタマイズしたり管理上の変更を行うことができます。Joomla! で実行できる事項の詳細については、「ステップ 7: Redmine のドキュメントを読み、引き続きウェブサイトの設定を続行する」のセクションを参照してください。

1. インスタンス管理ページの [Connect] (接続) タブにあるパブリック IP アドレスを書き留めます。パブリック IP アドレスは、インスタンス管理ページのヘッダーセクションにも表示されます。



2. インスタンスのパブリック IP アドレスを参照します (例: `http://203.0.113.0` に移動します)。

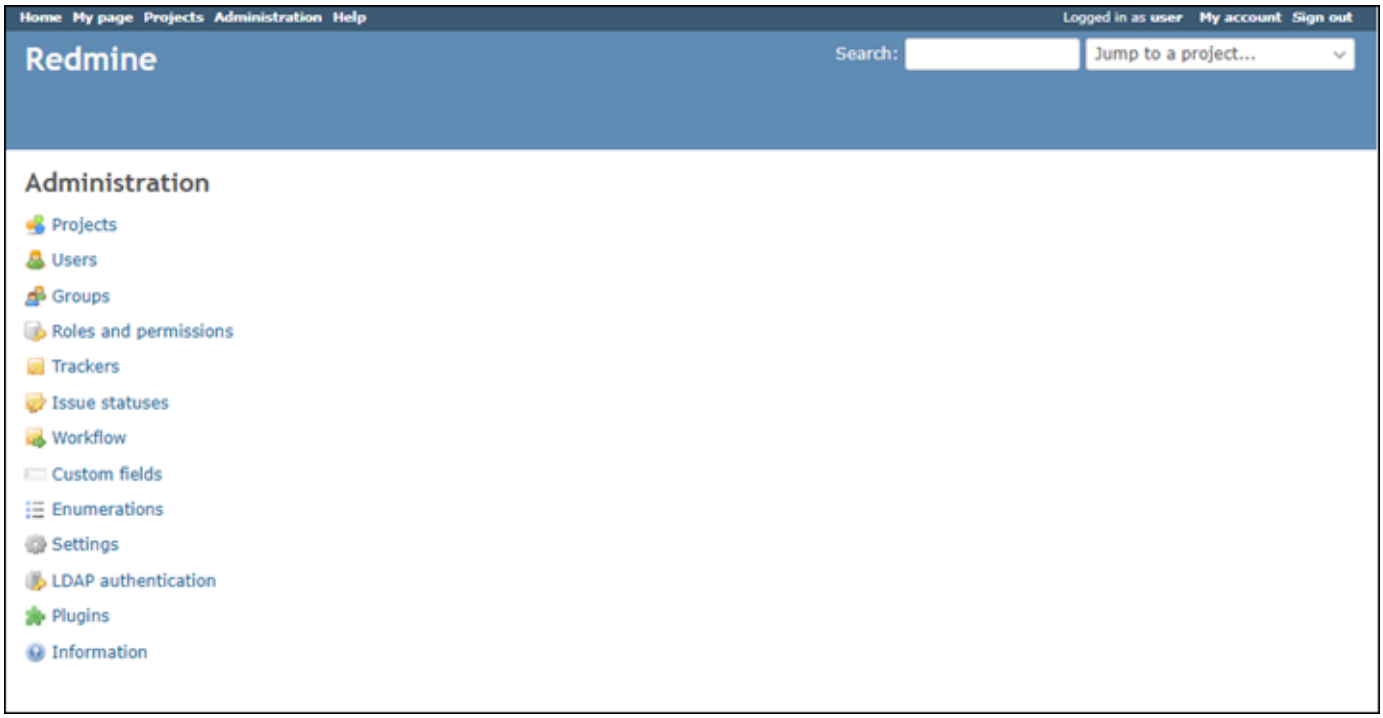
   Redmine ウェブサイトのホームページが表示されます。

3. Redmine ウェブサイトのホームページで、右下にある [Manage](管理) を選択します。

[Manage] (管理) バナーが表示されない場合は、`http://<PublicIP>/admin` を参照することでサインインページにアクセスすることができます。*<PublicIP>* を、インスタンスのパブリック IP アドレスに置き換えます。

4. デフォルトのユーザー名 (`user`) と、先ほど取得したデフォルトのパスワードを使用してサインインします。

   Redmine の管理ダッシュボードが表示されます。



ステップ 5: 登録済みドメイン名へのトラフィックを Redmine ウェブサイトに送信する

`example.com` などの登録済みドメイン名のトラフィックを Redmine ウェブサイトに送信するには、ドメインの DNS にレコードを追加します。DNS レコードは、通常、ドメインの登録先であるレジストラが管理またはホストします。ただし、ドメインの DNS レコードの管理を Lightsail に引き渡して、Lightsail コンソールで管理できるようにすることをお勧めします。

Lightsail コンソールのホームページの [Domains & DNS] (ドメインと DNS) タブで、[Create DNS zone] (DNS ゾーンの作成) を選択し、ページに記載される手順に従います。詳細については、「Lightsail で DNS ゾーンを作成し、ドメインの DNS レコードを管理する」を参照してください。

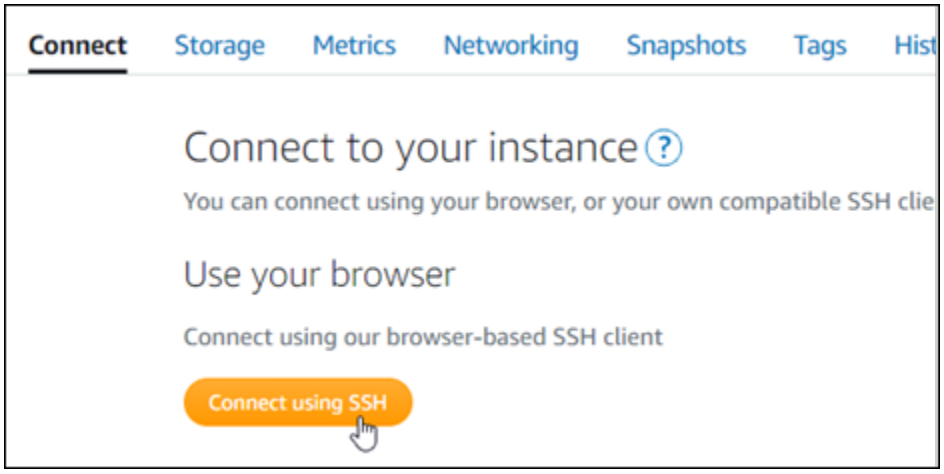
インスタンスに設定したドメイン名を参照すると、Redmine ウェブサイトのホームページへと移動します。次に、SSL/TLS 証明書を生成して設定し、Redmine ウェブサイトの HTTPS 接続を有効にします。詳細については、本ガイドの次の「ステップ 6: Redmine ウェブサイトの HTTPS を設定する」のセクションを参照してください。

ステップ 6: Redmine ウェブサイトの HTTPS を設定する

Redmine ウェブサイトで HTTPS を設定するには、以下の手順を実行します。次の手順では、Bitnami HTTPS 設定ツール (`bncert-tool`) の使い方を説明しています。これは、Let's Encrypt SSL/TLS 証明書を要求するコマンドラインツールです。詳細については、Bitnami ドキュメントの「Bitnami 設定ツールの詳細を確認する」を参照してください。

> **Important**
>
> この手順を開始する前に、Redmine インスタンスにトラフィックがルーティングされるようにドメインが設定済みであることを確認してください。設定されていない場合、SSL/TLS 証明書の検証プロセスが失敗します。

1. インスタンス管理ページの [接続] タブで、[SSH を使用して接続] を選択します。



2. 接続されたら、以下のコマンドを入力し、インスタンスに bncert ツールがインストールされていることを確認します。

```
sudo /opt/bitnami/bncert-tool
```

以下のレスポンスのいずれかが表示されます。

- レスポンスにコマンドが見つからないと表示された場合、bncert ツールがインスタンスにインストールされていないことを示しています。この手順の次のステップに進み、bncert ツールをインスタンスにインストールします。
- レスポンスに「Welcome to the Bitnami HTTPS configuration tool (Bitnami HTTPS 設定ツールへようこそ)」と表示された場合は、インスタンスに bncert ツールがインストールされています。この手順のステップ 8 に進んでください。
- bncert ツールがインスタンスにインストールされてからしばらく経っている場合、ツールのアップデートバージョンが利用可能であることを示すメッセージが表示されることがあります。ダウンロードすることを選択し、`sudo /opt/bitnami/bncert-tool` コマンドを入力して bncert ツールを再度実行してください。この手順のステップ 8 に進んでください。

3. 以下のコマンドを入力して、bncert の実行ファイルをインスタンスにダウンロードします。

```
wget -O bncert-linux-x64.run https://downloads.bitnami.com/files/bncert/latest/
bncert-linux-x64.run
```

4. 以下のコマンドを入力して、インスタンスに bncert ツールの実行ファイル用のディレクトリを作成します。

```
sudo mkdir /opt/bitnami/bncert
```

5. 以下のコマンドを入力して、プログラムとして実行できるファイルを bncert に実行させます。

```
sudo chmod +x /opt/bitnami/bncert/bncert-linux-x64.run
```

6. 次のコマンドを入力して、sudo /opt/bitnami/bncert-tool コマンドを入力すると bncert ツールを実行するシンボリックリンクを作成します。

```
sudo ln -s /opt/bitnami/bncert/bncert-linux-x64.run /opt/bitnami/bncert-tool
```

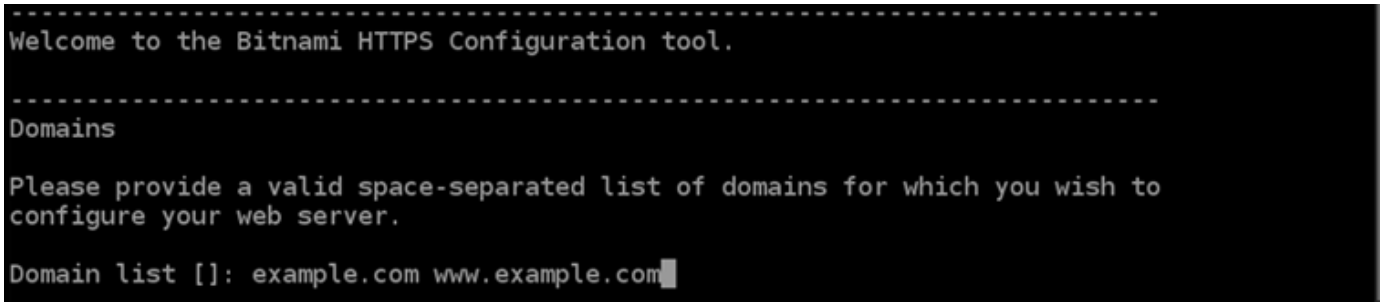
これでインスタンスに bncert ツールをインストールする手順は完了です。

7. 次のコマンドを入力して、bncert ツールを実行しましょう。

```
sudo /opt/bitnami/bncert-tool
```

8. 次の例に示すように、プライマリドメイン名と代替ドメイン名の間はスペースで区切って入力します。

ドメインがインスタンスのパブリック IP アドレスにトラフィックをルーティングするように設定されていない場合、bncert ツールは、続行する前にその設定を行うように要求します。ドメインは、bncert ツールを使用して HTTPS を有効にしているインスタンスでのパブリック IP アドレスにトラフィックをルーティングする必要があります。これはドメインを所有していることを確認し、証明書の検証として機能します。



9. bncert ツールは、ウェブサイトのリダイレクトの設定方法を尋ねます。使用できるオプションは次のとおりです。

   - HTTP から HTTPS へのリダイレクトを有効にする - HTTP バージョンのウェブサイトを閲覧するユーザー (例: http:/example.com) を自動的に HTTPS バージョン (例: https://example.com) にリダイレクトするかどうかを決定します。すべての訪問者が暗号化された接続を使用するように強制されるため、このオプションを有効にすることをお勧めします。Y を入力して Enter を押すと、有効になります。
   - www なしから www ありへのリダイレクトの有効化 - ドメインの頂点 (例: https://example.com) まで閲覧するユーザー を自動的にドメインの www サブドメイン (例: https://www.example.com) にリダイレクトするかを指定します。このオプションを有効にすることをお勧めします。ただし、ドメインの頂点を Google のウェブマスターツールなどの検索エンジンツールで希望のウェブサイトアドレスとして指定した場合、または頂点が IP を直接指しており、www のサブドメインが CNAME レコードを介してリファレンスしている場合は、無効にして代替オプションを有効にすることをお勧めします (www ありから www なしへのリダイレクトを有効化) 。Y を入力し、Enter を押して有効にします。
   - www ありから www なしへのリダイレクトを有効にする - ドメインの www サブドメイン (例: https://www.example.com) まで閲覧するユーザーを、自動的にドメインの頂点 (例: https://example.com) にリダイレクトするかを指定します。www なしから www ありへのリダイレクトを有効にした場合は、これを無効にすることをお勧めします。N を入力し、Enter を押して無効にします。

   選択した結果は次の例のようになります。

```
Enable/disable redirections

Please select the redirections you wish to enable or disable on your Bitnami
installation.



Enable HTTP to HTTPS redirection [Y/n]: Y



Enable non-www to www redirection [Y/n]: Y



Enable www to non-www redirection [y/N]: N
```

10. これから実行される変更が一覧表示されます。Y と入力し、Enter を押して確認し、続行します。

```
Changes to perform

The following changes will be performed to your Bitnami installation:

1. Stop web server
2. Configure web server to use a free Let's Encrypt certificate for the domains:
example.com www.example.com
3. Configure a cron job to automatically renew the certificate each month
4. Configure web server name to: example.com
5. Enable HTTP to HTTPS redirection (example: redirect http://example.com to
https://example.com)
6. Enable non-www to www redirection (example: redirect example.com to
 www.example.com)
7. Start web server once all changes have been performed



Do you agree to these changes? [Y/n]: Y
```

11. Let's Encrypt 証明書に関連付けるメールアドレスを入力し、Enter を押します。

```
Create a free HTTPS certificate with Let's Encrypt

Please provide a valid e-mail address for which to associate your Let's Encrypt
certificate.

Domain list: example.com www.example.com

Server name: example.com

E-mail address []:
```

12. Let's Encrypt サブスクライバー合意書を確認します。Y と入力し、Enter を押して契約に同意し、続行します。

```
The Let's Encrypt Subscriber Agreement can be found at:

https://letsencrypt.org/documents/LE-SA-v1.2-November-15-2017.pdf

Do you agree to the Let's Encrypt Subscriber Agreement? [Y/n]:
```

これらのアクションは、証明書のリクエストや指定したリダイレクトの設定など、インスタンスで HTTPS を有効にするために実行されます。

```
Performing changes to your installation

The Bitnami HTTPS Configuration Tool will perform any necessary actions to your
Bitnami installation. This may take some time, please be patient.
```

次の例のようなメッセージが表示された場合は、証明書は正常に発行され、検証され、インスタンスでリダイレクトが正常に設定されています。

```
Success

The Bitnami HTTPS Configuration Tool succeeded in modifying your installation.

The configuration report is shown below.

Backup files:
* /opt/bitnami/apache2/conf/httpd.conf.back.202005290035
* /opt/bitnami/apache2/conf/bitnami/bitnami-apps-prefix.conf.back.202005290035
* /opt/bitnami/apache2/conf/bitnami/bitnami.conf.back.202005290035

Find more details in the log file:

/tmp/bncert-202005290035.log

If you find any issues, please check Bitnami Support forums at:

https://community.bitnami.com

Press [Enter] to continue:
```

`bncert` ツールは、有効期限が切れる前、 80 日ごとに証明書の自動更新を実行します。インスタンスで追加のドメインやサブドメインを使用し、それらのドメインで HTTPS を有効にする場合は、上記のステップを繰り返します。

これで、Redmine インスタンスでの HTTPS の有効化が完了しました。次回、設定したドメインを使用して Redmine ウェブサイトを閲覧する際には、HTTPS 接続にリダイレクトされるはずです。

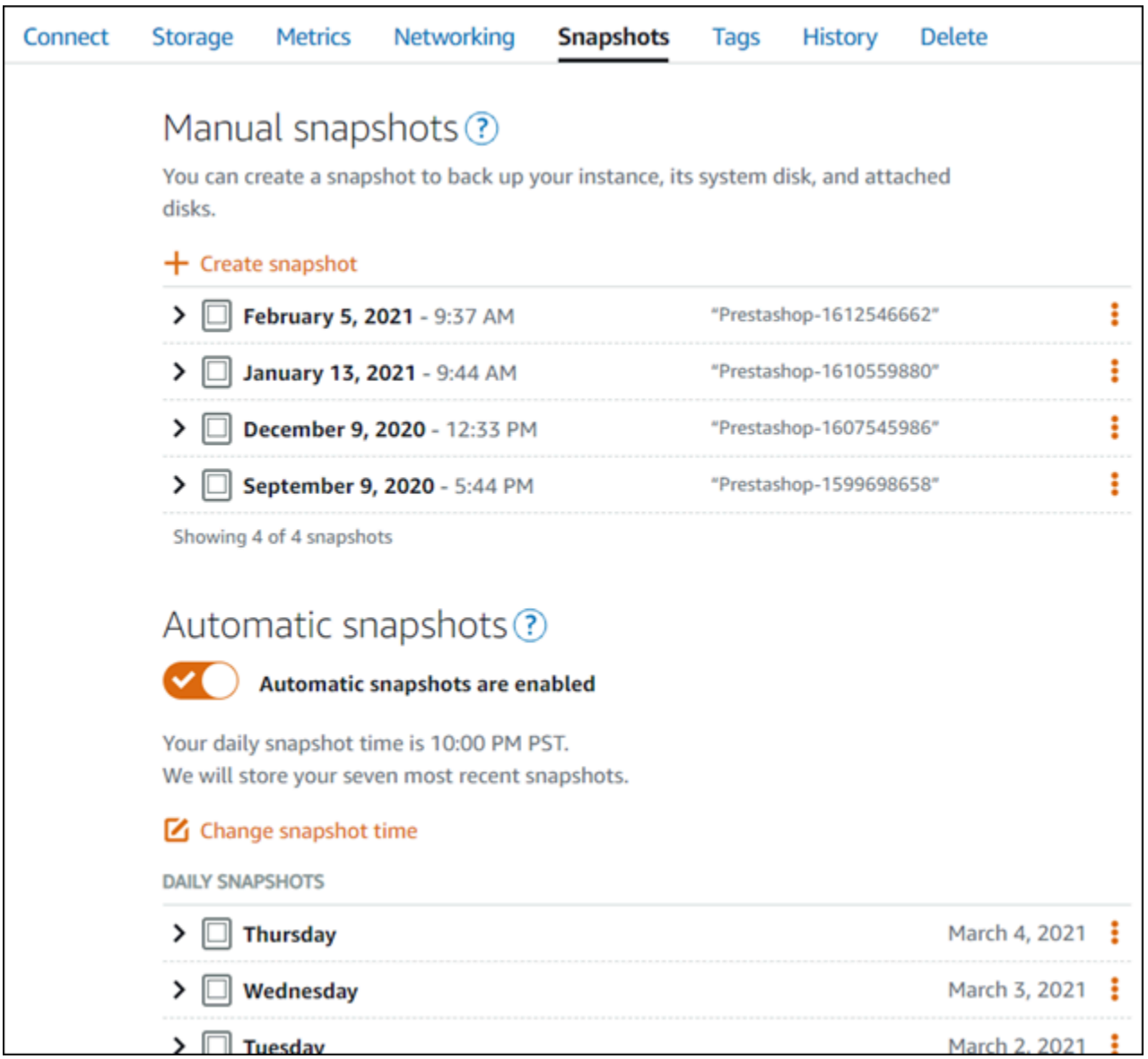
ステップ 7: Redmine のドキュメントを読み、引き続きウェブサイトの設定を続行する

Redmine のドキュメントを読み、ウェブサイトを管理およびカスタマイズする方法を確認します。詳細については、「Redmine ガイド」を参照してください。

ステップ 8: インスタンスのスナップショットを作成する

Redmine ウェブサイトを希望どおりに設定したら、インスタンスの定期的なスナップショットを作成してバックアップします。スナップショットは手動で作成するか、自動スナップショットを有効にして Lightsail に毎日のスナップショットを作成させることができます。インスタンスに問題が発生した場合は、スナップショットを使用して新しい代替インスタンスを作成できます。詳細については、「スナップショット」を参照してください。

インスタンス管理ページの [スナップショット] タブで [スナップショットを作成する] を選択するか、[自動スナップショットを有効にする] を選択します。



詳細については、「Amazon Lightsail で Linux または Unix インスタンスのスナップショットを作成」および「Amazon Lightsail のインスタンスまたはディスクの自動スナップショットの有効化と無効化」を参照してください。

クイックスタートガイド: WordPress

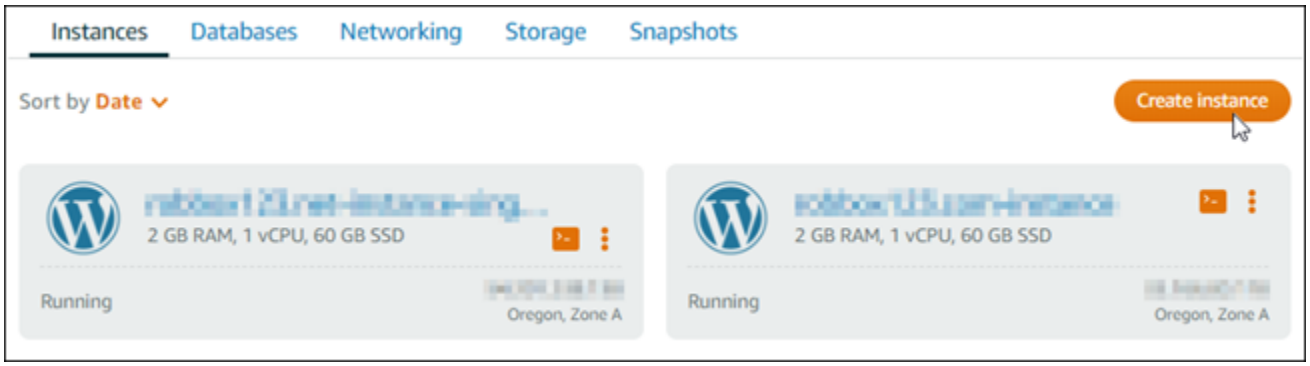
このクイックスタートガイドでは、Amazon Lightsail WordPress でインスタンスを起動して設定する方法を学習します。

ステップ 1: インスタンスを作成する WordPress

以下のステップを実行して、 WordPress インスタンスを起動して稼働させます。

用の Lightsail インスタンスを作成するには WordPress

1. Lightsail コンソールにサインインします。
2. Lightsail ホームページのインスタンスセクションで、[インスタンスを作成] を選択します。



3. インスタンスのアベイラビリティーゾーンを選択します。 AWS リージョン

4. インスタンスのイメージを次のように選択します。

   a. [プラットフォームの選択] で [Linux/Unix] を選択します。

   b. [ブループリントの選択] では、を選択します。WordPress

5. インスタンスプランを選択します。

プランには、予測可能な低コストでのマシン構成 (RAM、SSD、vCPU) と、データ転送許容量が含まれます。

6. インスタンスの名前を入力します。リソース名:

   - Lightsail AWS リージョン アカウント内のそれぞれで一意である必要があります。
   - 2〜255 文字を使用する必要があります。
   - 先頭と末尾は英数字または数字を使用する必要があります。
   - 英数字、数字、ピリオド、ダッシュ、アンダースコアを使用することができます。

7. [インスタンスの作成] を選択します。

8. テストブログ投稿を表示するには、インスタンス管理ページに移動し、ページの右上隅に表示されているパブリック IPv4 アドレスをコピーします。インターネットに接続されたウェブブラウザのアドレスフィールドにアドレスを貼り付けます。ブラウザにテストブログ投稿が表示されます。

ステップ 2: WordPress インスタンスを設定する

WordPress step-by-step 以下を設定するガイド付きのワークフローを使用してインスタンスを設定できます。

- 登録済みドメイン名 — WordPress サイトには覚えやすいドメイン名が必要です。 WordPress ユーザーはこのドメイン名を指定してサイトにアクセスします。詳細については、「ドメインと DNS」を参照してください。
- DNS 管理 — ドメインの DNS レコードの管理方法を決定する必要があります。DNS レコードは、ドメインまたはサブドメインが関連付けられている IP アドレスまたはホスト名を DNS サーバーに伝えます。DNS ゾーンにはドメインの DNS レコードが含まれます。詳細については、「the section called “Lightsail の DNS”」を参照してください。
- 静的 IP アドレス — インスタンスを停止して起動すると、 WordPress インスタンスのデフォルトのパブリック IP アドレスが変更されます。静的 IP アドレスをインスタンスにアタッチすると、インスタンスを停止して起動しても同じままです。詳細については、「the section called “IP アドレス”」を参照してください。
- SSL/TLS 証明書 — 検証済みの証明書を作成してインスタンスにインストールしたら、WordPress ウェブサイトの HTTPS を有効にして、登録したドメインを経由してインスタンスにルーティングされるトラフィックが HTTPS を使用して暗号化されるようにすることができます。詳細については、「the section called “HTTPS を有効にする”」を参照してください。

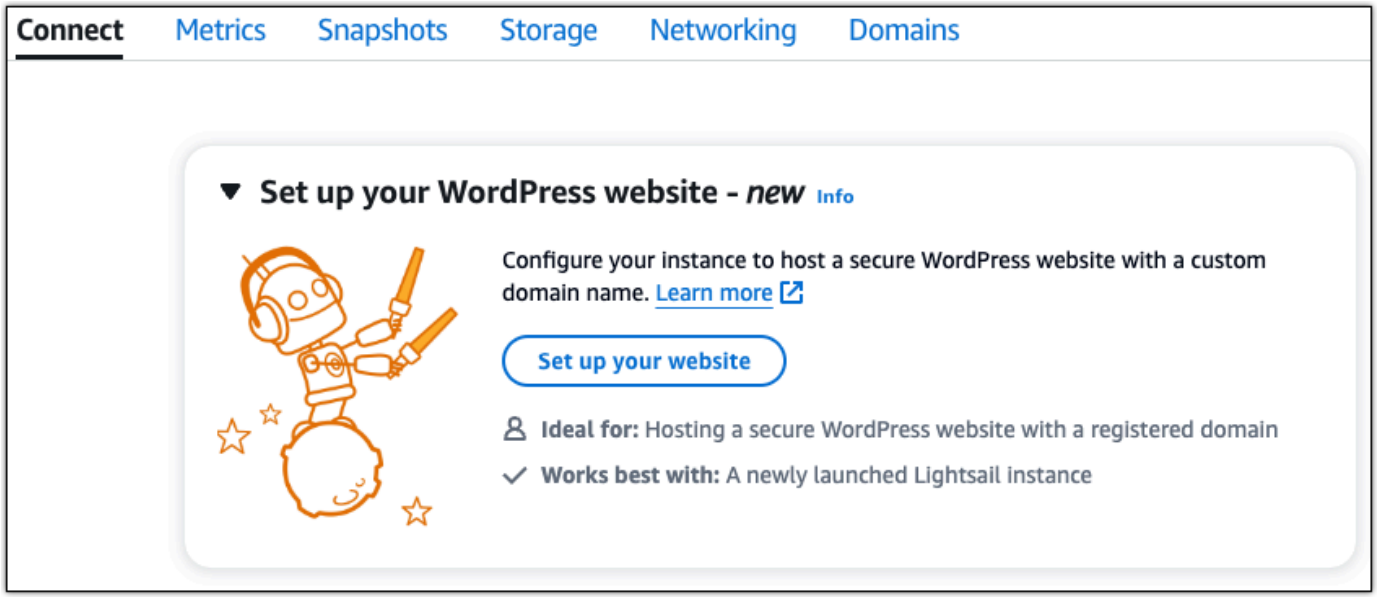
Tip

始める前に、以下のヒントを確認してください。トラブルシューティング情報については、「WordPress トラブルシューティング設定」を参照してください。

- セットアップは、2023 年 1 月 1 WordPress 日以降に作成されたバージョン 6 以降の Lightsail インスタンスをサポートします。
- インスタンスは Running 状態である必要があります。インスタンスが起動されたばかりの場合は、SSH 接続の準備が整うまで数分お待ちください。
- インスタンスのファイアウォールのポート 22、80、443 は、セットアップの実行中は任意の IP アドレスからの TCP 接続を許可する必要があります。詳細については、「インスタンスのファイアウォール」を参照してください。
- Apex ドメイン (`example.com`) `www` とそのサブドメイン () からのトラフィックを指す DNS レコードを追加または更新する場合、それらはインターネット全体に伝播する必要があります。`www.example.com`DNS の変更が反映されたかどうかは、nslookup やからの DNS 検索などのツールを使用して確認できます。MxToolbox
- 2023 年 1 月 1 日より前に作成された Wordpress インスタンスには、廃止された Certbot 個人Package アーカイブ (PPA) リポジトリが含まれている可能性があり、これによりウェブサイトの設定が失敗する可能性があります。セットアップ中にこのリポジトリが存在する場合、既存のパスから削除され、インスタンスの次の場所にバックアップされます。`~/opt/bitnami/lightsail/repo.backup`廃止された PPA の詳細については、Canonical ウェブサイトの Certbot PPA を参照してください。
- Let's Encrypt の証明書は 60 日から 90 日ごとに自動的に更新されます。
- セットアップ中は、インスタンスを停止したり変更したりしないでください。インスタンスの設定には最大 15 分かかることがあります。インスタンス接続タブで各ステップの進行状況を確認できます。

ウェブサイトセットアップウィザードを使用してインスタンスを設定するには

1. インスタンス管理ページの「Connect」タブで、「ウェブサイトを設定」を選択します。



2. [ドメイン名を指定] では、既存の Lightsail 管理ドメインを使用するか、Lightsail に新しいドメインを登録するか、別のドメインレジストラを使用して登録したドメインを使用します。[このドメインを使用する] を選択して次のステップに進みます。
3. [DNS の設定] で、次のいずれかを実行します。
   - Lightsail DNS ゾーンを使用するには、Lightsail マネージドドメインを選択してください。[この DNS ゾーンを使用する] を選択して次のステップに進みます。
   - ドメインの DNS レコードを管理するホスティングサービスを使用するには、「サードパーティードメイン」を選択します。後で使用することを決めた場合に備えて、Lightsail アカウントに一致する DNS ゾーンが作成されることに注意してください。[サードパーティ DNS を使用] を選択して次のステップに進みます。
4. [固定 IP アドレスの作成] に、固定 IP アドレスの名前を入力し、[静的 IP アドレスの作成] を選択します。
5. [ドメイン割り当ての管理] で [割り当てを追加] を選択し、ドメインの種類を選択して、[追加] を選択します。[続行] を選択して次のステップに進みます。
6. [SSL/TLS 証明書の作成] では、ドメインとサブドメインを選択し、メールアドレスを入力し、[Lightsail に権限を付与してインスタンスに Let's Encrypt 証明書を設定する] を選択し、[証明書の作成] を選択します。Lightsail リソースの設定を開始します。

   セットアップ中は、インスタンスを停止したり変更したりしないでください。インスタンスの設定には最大 15 分かかることがあります。インスタンス接続タブで各ステップの進行状況を確認できます。

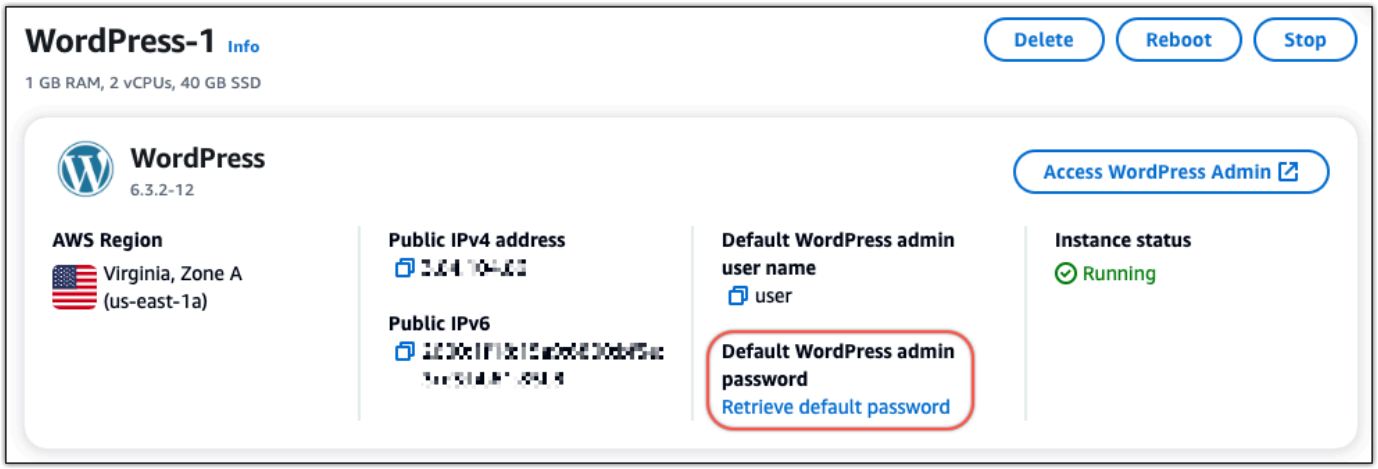
7. ウェブサイトの設定が完了したら、ドメイン割り当て手順で指定した URL WordPress がサイトを開くことを確認します。

ステップ 3: Web サイトのデフォルトアプリケーションパスワードを取得する WordPress

WordPress Web サイトの管理ダッシュボードにサインインするには、デフォルトのアプリケーションパスワードが必要です。

WordPress 管理者のデフォルトパスワードを取得するには

1. インスタンスのインスタンス管理ページを開きます。 WordPress
2. WordPressパネルで [デフォルトパスワードを取得] を選択します。これにより、ページの下部にある Access のデフォルトパスワードが展開されます。



3. [起動] を選択します。 CloudShellページの下部にパネルが開きます。
4. [Copy] を選択し、 CloudShell 内容をウィンドウに貼り付けます。 CloudShell プロンプトにカーソルを置いて Ctrl+V を押すか、右クリックしてメニューを開き、[貼り付け] を選択します。
5. ウィンドウに表示されたパスワードを書き留めておきます。 CloudShell WordPress Web サイトの管理ダッシュボードにサインインする際に必要です。

```
[cloudshell-user@ip-[redacted] ~]$ AWS_REGION=us-east-1 ~/lightsail_connect WordPress-1 cat bitnami_applic
ation_password
JKzh8wB5FAR!
```

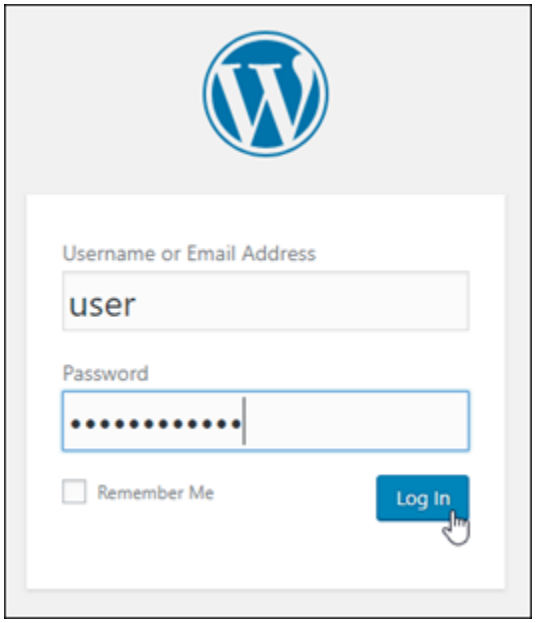
ステップ 4: ウェブサイトへのサインイン WordPress

デフォルトのユーザーパスワードがわかったので、 WordPress Web サイトのホームページに移動し、管理ダッシュボードにログインします。サインイン後に、デフォルトのパスワードを変更できます。

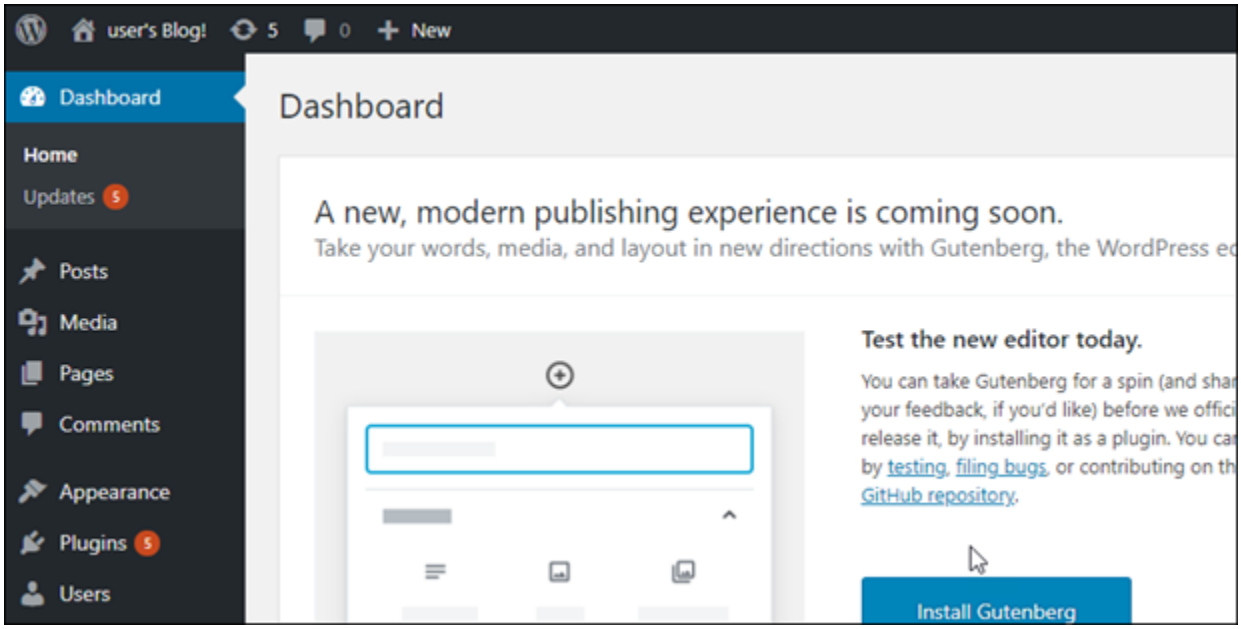
管理ダッシュボードにサインインするには

1. インスタンスのインスタンス管理ページを開きます。 WordPress
2. WordPressパネルで [WordPress 管理者にアクセス] を選択します。
3. 「 WordPress 管理者ダッシュボードへのアクセス」パネルの「パブリック IP アドレスを使用」で、次の形式のリンクを選択します。

   http://*##### IPV4-####*。 /wp-管理者
4. [ユーザー名] または [メールアドレス] に、と入力します。**user**
5. [パスワード] には、前のステップで取得したパスワードを入力します。
6. [ログイン] を選択します。



これで WordPress Web サイトの管理ダッシュボードにサインインし、管理アクションを実行できます。 WordPress Web サイトの管理について詳しくは、ドキュメントの WordPressCodex を参照してください。 WordPress



ステップ 5: Bitnami のドキュメントを確認する

Bitnami のドキュメントを読んで、プラグインのインストール、テーマのカスタマイズ、バージョンのアップグレードなど、 WordPress Web サイトの管理タスクを実行する方法を確認してください。 WordPress

詳細については、 WordPress Bitnami フォームを参照してください。 AWS クラウド

クイックスタートガイド: WordPress Multisite

Amazon Lightsail で起動した WordPress Multisite インスタンスの使用を開始するステップについて説明します。

目次

- ステップ 7: WordPress Multisite のドキュメントを読み、引き続きウェブサイトの設定を続行する
- ステップ 8: インスタンスのスナップショットを作成する

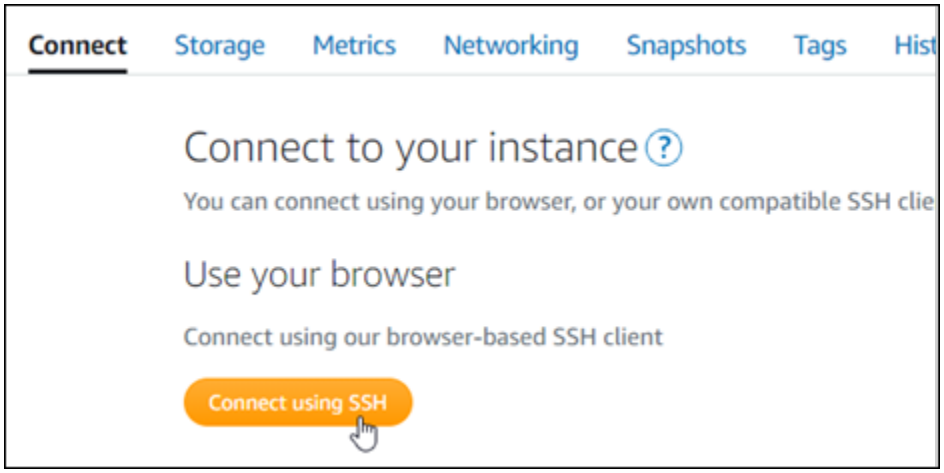
ステップ 1: Bitnami のドキュメントを確認する

Bitnami のドキュメントを読んで WordPress Multisite インスタンスを設定する方法を確認します。詳細については、「AWS クラウド 用に Bitnami がパッケージ化した WordPress Multisite」を参照してください。

ステップ 2: WordPress の管理ダッシュボードにアクセスするため、デフォルトのアプリケーションパスワードを取得する

次の手順を完了して、WordPress Multisite ウェブサイトの管理ダッシュボードにアクセスする際に必要となるデフォルトのアプリケーションパスワードを取得します。詳細については、「Amazon Lightsailの Bitnami インスタンス向けにアプリケーションのユーザー名とパスワードを取得する」を参照してください。

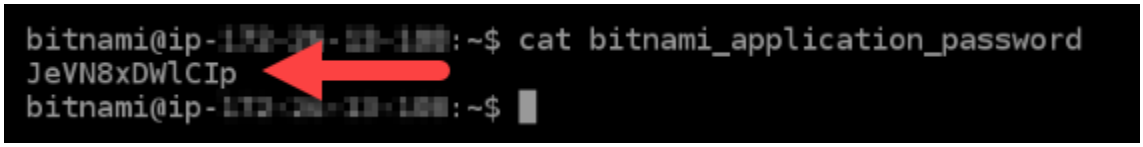
1. インスタンス管理ページの [接続] タブで、[SSH を使用して接続] を選択します。



2. 接続後に、次のコマンドを入力してデフォルトのアプリケーションのパスワードを取得します。

```
cat $HOME/bitnami_application_password
```

アプリケーションのデフォルトパスワードを含んだ、次の例のようなレスポンスが表示されます。このパスワードを使用して、WordPress Multisite ウェブサイトの管理ダッシュボードにサインインします。



ステップ 3: インスタンスに静的 IP アドレスをアタッチする

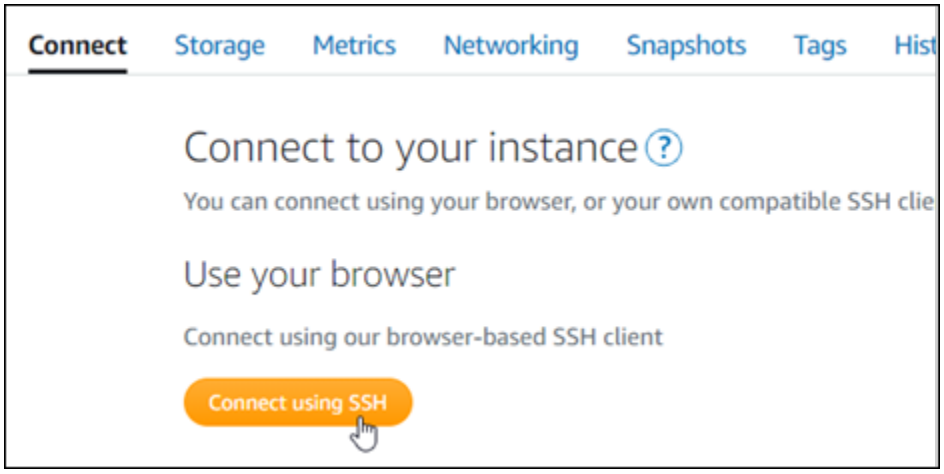
インスタンスを最初に作成した際に割り当てられたパブリック IP アドレスは、インスタンスを停止してスタートするたびに変更されます。パブリック IP アドレスが変更されないように、静的 IP アドレスを作成してインスタンスにアタッチする必要があります。後ほど、example.com などの登録したドメイン名をインスタンスで使用する際、毎回インスタンスを停止して開始するたびにドメインのドメインネームシステム (DNS) を更新する必要がなくなります。1 つの静的 IP を 1 つのインスタンスにアタッチできます。

インスタンス管理ページの [ネットワーク] タブで、[静的 IP の作成] または [静的 IP のアタッチ] (インスタンスにアタッチできる静的 IP を既に作成している場合) を選択して、ページの手順に従います。詳細については、「静的 IP を作成してインスタンスにアタッチする」を参照してください。

新しい静的 IP アドレスがインスタンスに添付されたら、次の手順を実行して、WordPress に新しい静的 IP アドレスを認識させる必要があります。

1. インスタンスの新しい静的 IP アドレスは書き留めておきます。このIP アドレスはインスタンス管理ページの ヘッダーセクションに表示されます。

2. インスタンス管理ページの [Connect] (接続) タブで、[SSH を使用して接続] を選択します。



3. 接続後に、次のコマンドを入力します。*<StaticIP>* をインスタンスの新しい静的 IP アドレスに置き換えます。

```
sudo /opt/bitnami/configure_app_domain --domain <StaticIP>
```

例:

```
sudo /opt/bitnami/configure_app_domain --domain 203.0.113.0
```

次の例のようなレスポンスが表示されます。これで、インスタンスの WordPress ウェブサイトが新しい静的 IP アドレスを認識するようになります。

```
bitnami@ip-[redacted]:~$ sudo /opt/bitnami/configure_app_domain --domain 203.0.113.0
Configuring domain to 203.0.113.0
2021-03-12T15:49:22.000Z - info: Saving configuration info to disk
prestashop 15:49:22.41 INFO  ==> Trying to connect to the database server
prestashop 15:49:22.44 INFO  ==> Updating hostname in database
prestashop 15:49:22.46 INFO  ==> Purging cache
Disabling automatic domain update for IP address changes
```

このコマンドが失敗した場合、古いバージョンの WordPress Multisite インスタンスを使用している可能性があります。代わりに次のコマンドを実行してみてください。*<StaticIP>* をインスタンスの新しい静的 IP アドレスに置き換えます。

```
cd /opt/bitnami/apps/wordpress
sudo ./bnconfig --machine_hostname <StaticIP>
```

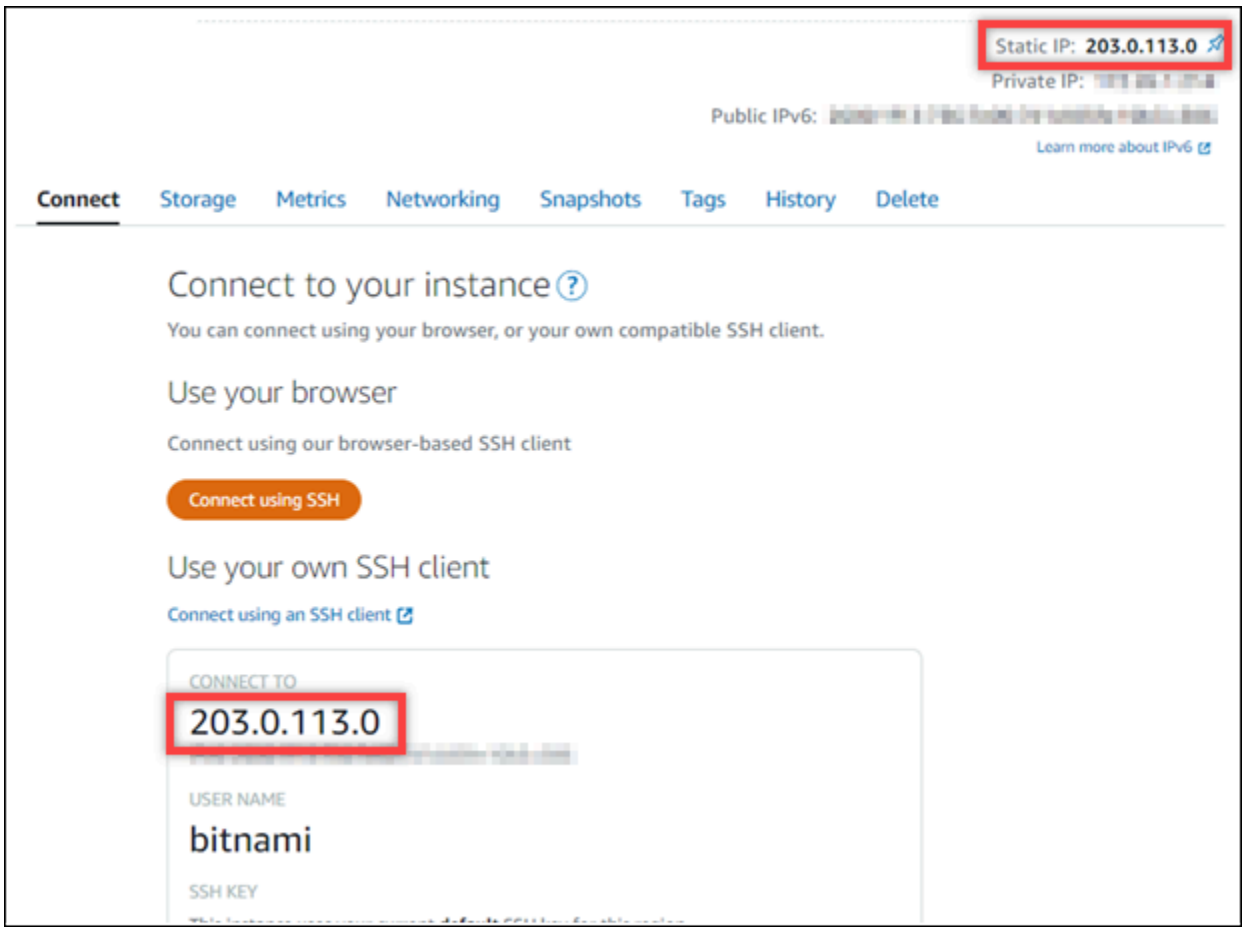
コマンドの実行が終了したら次のコマンドを入力し、サーバーが再起動するたびに bnconfig ツールが自動的に実行されないようにします。

```
sudo mv bnconfig bnconfig.disabled
```

ステップ 4: WordPress Multisite ウェブサイトの管理ダッシュボードにサインインする

デフォルトのユーザーパスワードを取得したら、以下の手順に従ってWordPress Multisite ウェブサイトのホームページに移動し、管理ダッシュボードにサインインします。サインイン後に、ウェブサイトをカスタマイズしたり管理上の変更を行うことができます。WordPress で実行できる事項の詳細については、本ガイドの後半にある「ステップ 7: WordPress Multisite のドキュメントを読み、引き続きウェブサイトの設定を続行する」のセクションを参照してください。

1. インスタンス管理ページの [Connect] (接続) タブにあるパブリック IP アドレスを書き留めます。パブリック IP アドレスは、インスタンス管理ページのヘッダーセクションにも表示されます。



2. インスタンスのパブリック IP アドレスを参照します (例: `http://203.0.113.0` に移動します)。

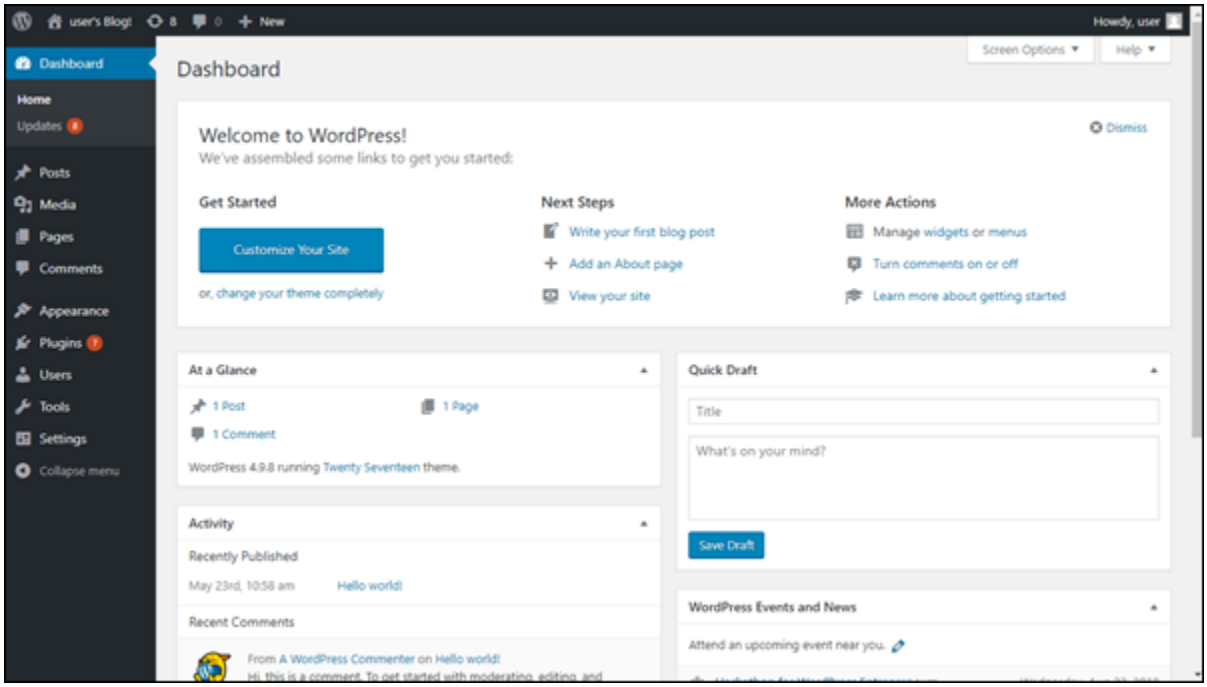
   WordPress ウェブサイトのホームページが表示されます。

3. WordPress ウェブサイトのホームページで、右下にある [Manage] (管理) を選択します。

   [Manage] (管理) バナーが表示されない場合は、`http://`*`<PublicIP>`*`/wp-login.php` を参照することでサインインページにアクセスすることができます。*`<PublicIP>`* を、インスタンスのパブリック IP アドレスに置き換えます。

4. デフォルトのユーザー名 (`user`) と、先ほど取得したデフォルトのパスワードを使用してサインインします。

   WordPress の管理ダッシュボードが表示されます。



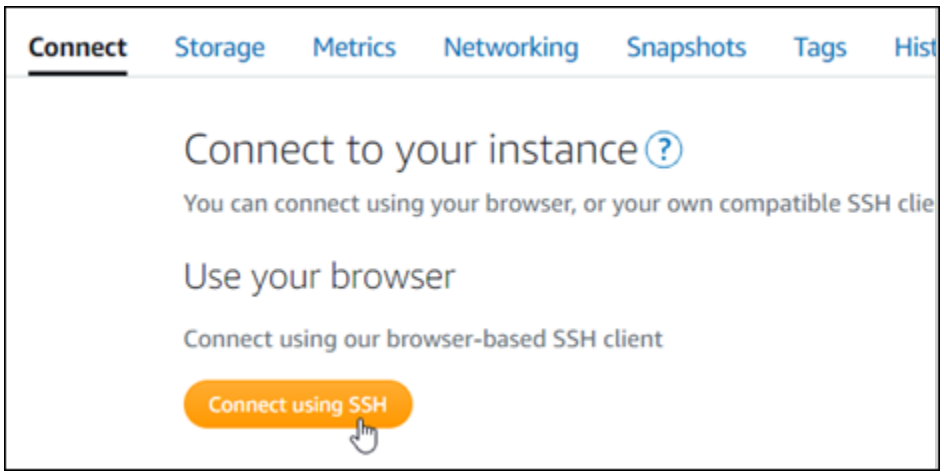
ステップ 5: 登録済みドメイン名へのトラフィックを WordPress Multisite ウェブサイトに送信する

`example.com` などの登録済みドメイン名のトラフィックを WordPress Multisite ウェブサイトに送信するには、ドメインの DNS にレコードを追加します。DNS レコードは、通常、ドメインの登録先であるレジストラが管理またはホストします。ただし、ドメインの DNS レコードの管理を Lightsail に引き渡して、Lightsail コンソールで管理できるようにすることをお勧めします。

Lightsail コンソールのホームページの [Domains & DNS] (ドメインと DNS) タブで、[Create DNS zone] (DNS ゾーンの作成) を選択し、ページに記載される手順に従います。詳細については、「Lightsail で DNS ゾーンを作成し、ドメインの DNS レコードを管理する」を参照してください。

ドメイン名へのトラフィックがインスタンスにルーティングされたら、次の手順を実行して、WordPress にドメイン名を認識させます。

1. インスタンス管理ページの [Connect] (接続) タブで、[SSH を使用して接続] を選択します。



2. 接続後に、次のコマンドを入力します。*<DomainName>* は、インスタンスにトラフィックをルーティングするドメイン名に置き換えてください。

```
sudo /opt/bitnami/configure_app_domain --domain <DomainName>
```

例:

```
sudo /opt/bitnami/configure_app_domain --domain www.example.com
```

次の例のようなレスポンスが表示されます。これで、 WordPress Multisite ソフトウェアがドメイン名を認識できるようになります。

```
bitnami@ip-[redacted]:~$ sudo /opt/bitnami/configure_app_domain --domain www.example.com
Configuring domain to www.example.com
2021-03-12T15:49:22.000Z - info: Saving configuration info to disk
prestashop 15:49:22.41 INFO  ==> Trying to connect to the database server
prestashop 15:49:22.44 INFO  ==> Updating hostname in database
prestashop 15:49:22.46 INFO  ==> Purging cache
Disabling automatic domain update for IP address changes
```

このコマンドが失敗した場合、古いバージョンの WordPress Multisite インスタンスを使用している可能性があります。代わりに次のコマンドを実行してみてください。*<DomainName>* は、インスタンスにトラフィックをルーティングするドメイン名に置き換えてください。

```
cd /opt/bitnami/apps/wordpress
sudo ./bnconfig --machine_hostname <DomainName>
```

コマンドの実行が終了したら次のコマンドを入力し、サーバーが再起動するたびに bnconfig ツールが自動的に実行されないようにします。

```
sudo mv bnconfig bnconfig.disabled
```

インスタンスに設定したドメイン名を参照すると、WordPress Multisite ウェブサイトのホームページへと移動します。次に、WordPress Multisite ウェブサイトにブログをドメインとして追加するか、サブドメインとして追加するかを決定する必要があります。詳細については、本ガイドの次のステップにある「ステップ 6: WordPress Multisite ウェブサイトにブログをドメインまたはサブドメインとして追加する」セクションを参照してください。

ステップ 6: WordPress Multisite ウェブサイトにブログをドメインまたはサブドメインとして追加する

WordPress Multisite は、WordPress の 1 つのインスタンスで複数のブログウェブサイトをホストするように設計されています。新しいブログウェブサイトを WordPress Multisite に追加すると、独自のドメインまたは WordPress Multisite のプライマリドメインのサブドメインを使用するように設定できます。WordPress Multisite で使用するように設定できるのは、これらのオプションのうちの 1 つのみです。例えば、ブログサイトをドメインとして追加する場合、ブログサイトをサブドメインとして追加することはできず、またその逆も同様です。これらのオプションのいずれかを設定するには、次の説明のいずれかに従います。

- ブログサイトを `example1.com` や `example2.com` などのドメインとして追加する場合は、「Lightsail の WordPress Multisite インスタンスにブログをドメインとして追加する」を参照してください。
- ブログサイトを `one.example.com` や `two.example.com` などの WordPress Multisite のプライマリドメインのドメインとして追加する場合は、「Lightsail の WordPress Multisite インスタンスにブログをサブドメインとして追加する」を参照してください。

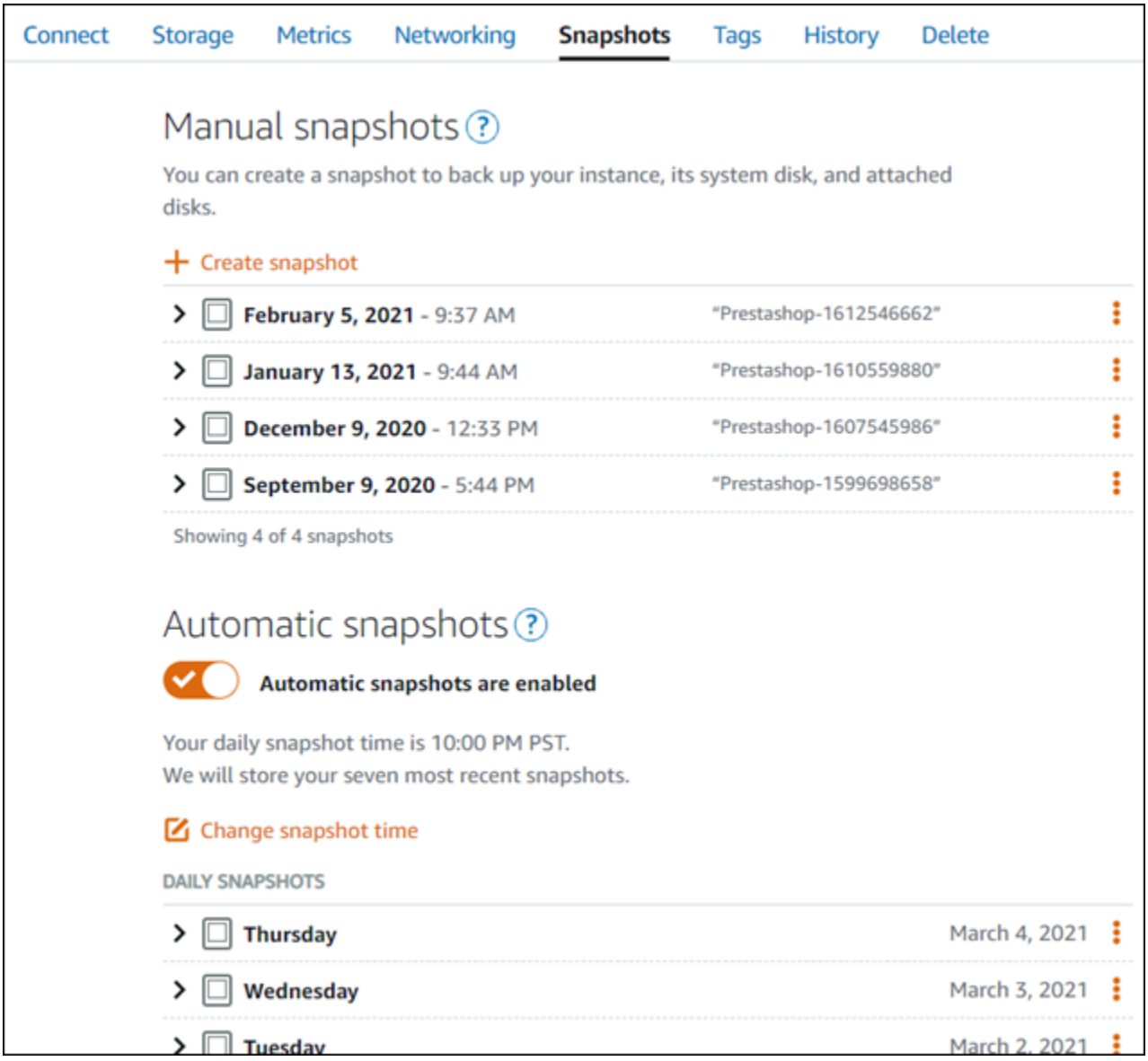
ステップ 7: WordPress Multisite のドキュメントを読み、引き続きウェブサイトの設定を続行する

WordPress Multisite のドキュメントを読み、ウェブサイトを管理およびカスタマイズする方法を確認します。詳細については、「WordPress Multisite ネットワーク管理ドキュメント」を参照してください。

ステップ 8: インスタンスのスナップショットを作成する

WordPress Multisite ウェブサイトを希望どおりに設定したら、インスタンスの定期的なスナップショットを作成してバックアップします。スナップショットは手動で作成するか、自動スナップショットを有効にして Lightsail に毎日のスナップショットを作成させることができます。インスタンスに問題が発生した場合は、スナップショットを使用して新しい代替インスタンスを作成できます。詳細については、「スナップショット」を参照してください。

インスタンス管理ページの [スナップショット] タブで [スナップショットを作成する] を選択するか、[自動スナップショットを有効にする] を選択します。



詳細については、「Amazon Lightsail で Linux または Unix インスタンスのスナップショットを作成」および「Amazon Lightsail のインスタンスまたはディスクの自動スナップショットの有効化と無効化」を参照してください。

Amazon Lightsail の Bitnami チュートリアル

Bitnami は、さまざまなプラットフォーム向けにパッケージ化され、すぐに実行できる開発スタックとアプリケーションを提供することで、ソフトウェアアプリケーションのデプロイを簡略化します。以下のチュートリアルを使用して、Lightsail で Bitnami を操作する方法について説明します

トピック

- Bitnami インスタンス用のアプリケーションのユーザー名とパスワードの取得
- Lightsail の Bitnami のブループリント インスタンスから Bitnami バナーを削除する

Bitnami インスタンス用のアプリケーションのユーザー名とパスワードの取得

Bitnami は、Amazon Lightsail インスタンス (仮想プライベートサーバー) として作成できるアプリケーションインスタンスイメージ (設計図) の多くを提供します。これらの設計図は、Lightsail コンソールのインスタンス作成ページに「Packaged by Bitnami」として示されます。

Bitnami ブループリントを使用してインスタンスを作成したら、このインスタンスにサインインして管理します。これを行うには、インスタンスで実行されるアプリケーションやデータベースのデフォルトのユーザー名とパスワードを取得する必要があります。この記事では、以下の設計図から作成された Lightsail インスタンスにサインインして管理するために必要な情報を取得する方法を示します。

- WordPress ブログおよびコンテンツ管理アプリケーション
- 同じインスタンスで複数のウェブサイトをサポートする WordPress Multisite ブログおよびコンテンツ管理アプリケーション
- Django 開発スタック
- Ghost ブログおよびコンテンツ管理アプリケーション
- LAMP 開発スタック (PHP 7)
- Node.js 開発スタック
- Joomla コンテンツ管理アプリケーション

- Magento e コマースアプリケーション
- MEAN 開発スタック
- Drupal コンテンツ管理アプリケーション
- GitLab CE リポジトリアプリケーション
- Redmine プロジェクト管理アプリケーション
- Nginx (LEMP) 開発スタック

Bitnami アプリケーションおよびデータベースのデフォルトユーザー名を取得する

以下は、Bitnami 設計図を使用して作成された Lightsail インスタンスのアプリケーションおよびデータベースのデフォルトユーザー名です。

> **Note**
>
> すべての Bitnami 設計図にアプリケーションやデータベースが含まれているわけではありません。設計図にアプリケーションやデータベースが含まれていない場合、ユーザー名は該当なし (N/A) として表示されます。

- WordPress (WordPress Multisite を含む)
  - アプリケーションユーザー名: `user`
  - データベースユーザー名: `root`
- PrestaShop
  - アプリケーションユーザー名: `user@example.com`
  - データベースユーザー名: `root`
- Django
  - アプリケーションユーザー名: N/A
  - データベースユーザー名: `root`
- Ghost
  - アプリケーションユーザー名: `user@example.com`
  - データベースユーザー名: `root`
- LAMP スタック (PHP 5 および PHP 7)
  - アプリケーションユーザー名: N/A

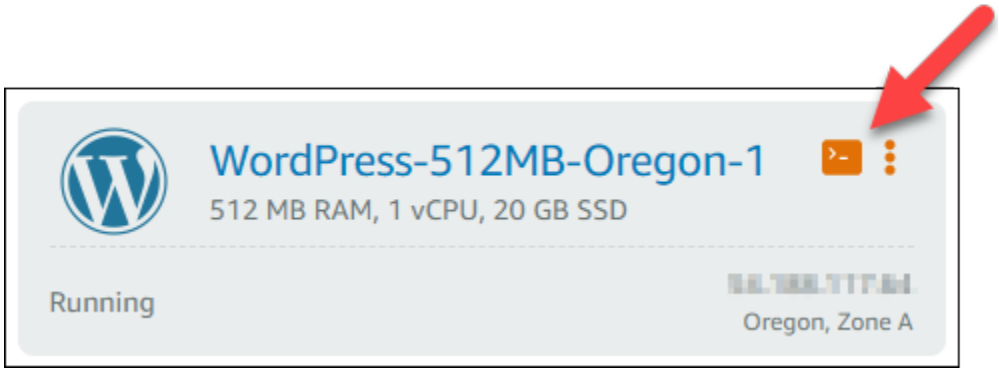
  - データベースユーザー名: `root`
- Node.js
  - アプリケーションユーザー名: N/A
  - データベースユーザー名: N/A
- Joomla
  - アプリケーションユーザー名: `user`
  - データベースユーザー名: `root`
- Magento
  - アプリケーションユーザー名: `user`
  - データベースユーザー名: `root`
- MEAN
  - アプリケーションユーザー名: N/A
  - データベースユーザー名: `root`
- Drupal
  - アプリケーションユーザー名: `user`
  - データベースユーザー名: `root`
- GitLab CE
  - アプリケーションユーザー名: `user`
  - データベースユーザー名: `postgres`
- Redmine
  - アプリケーションユーザー名: `user`
  - データベースユーザー名: `root`
- Nginx
  - アプリケーションユーザー名: N/A
  - データベースユーザー名: `root`

Bitnami アプリケーションおよびデータベースのデフォルトパスワードを取得する

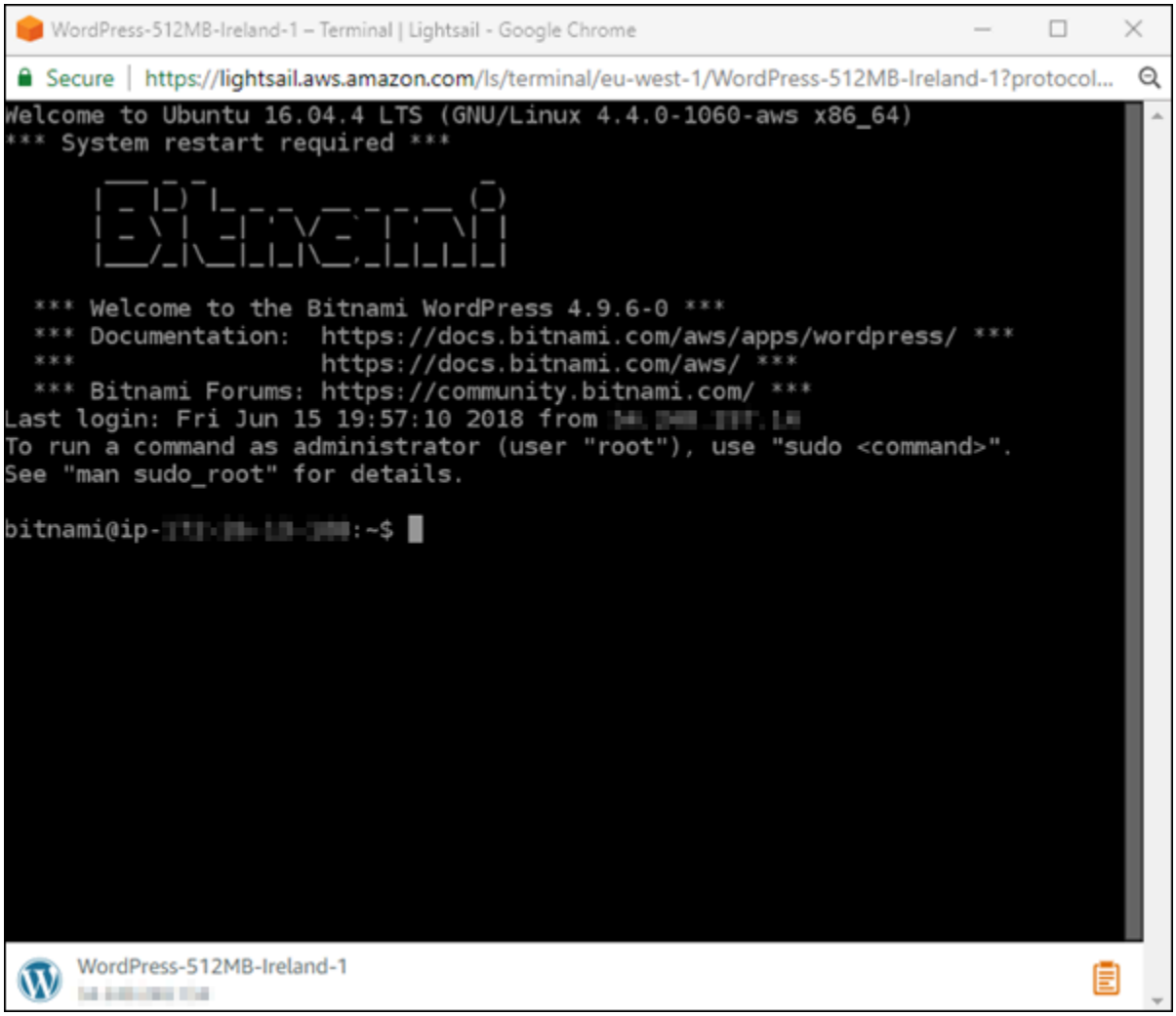
アプリケーションおよびデータベースのデフォルトパスワードはインスタンスに保存されています。これを取得するには、Lightsail コンソールのブラウザベースの SSH ターミナルを使用し、専用のコマンドを実行して接続します。

Bitnami アプリケーションおよびデータベースのデフォルトパスワードを取得するには

1. Lightsail コンソールにサインインします。
2. Bitnami ブループリントを使用してインスタンスを作成します (まだ作成していない場合)。詳細については、「Amazon Lightsail VPS を作成する」を参照してください。
3. Lightsail のホームページで、接続先のインスタンスのクイック接続アイコンを選択します。



以下の例に示すように、ブラウザベースの SSH クラインとウィンドウが開きます。

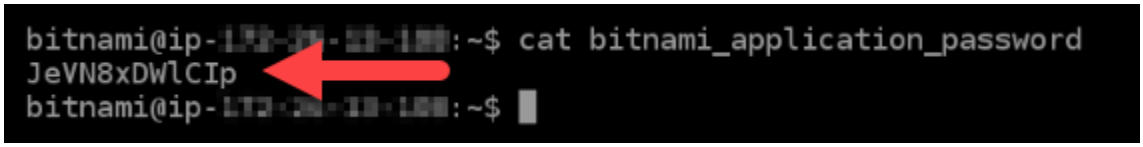


4. 次のコマンドを入力してデフォルトのアプリケーションのパスワードを取得します。

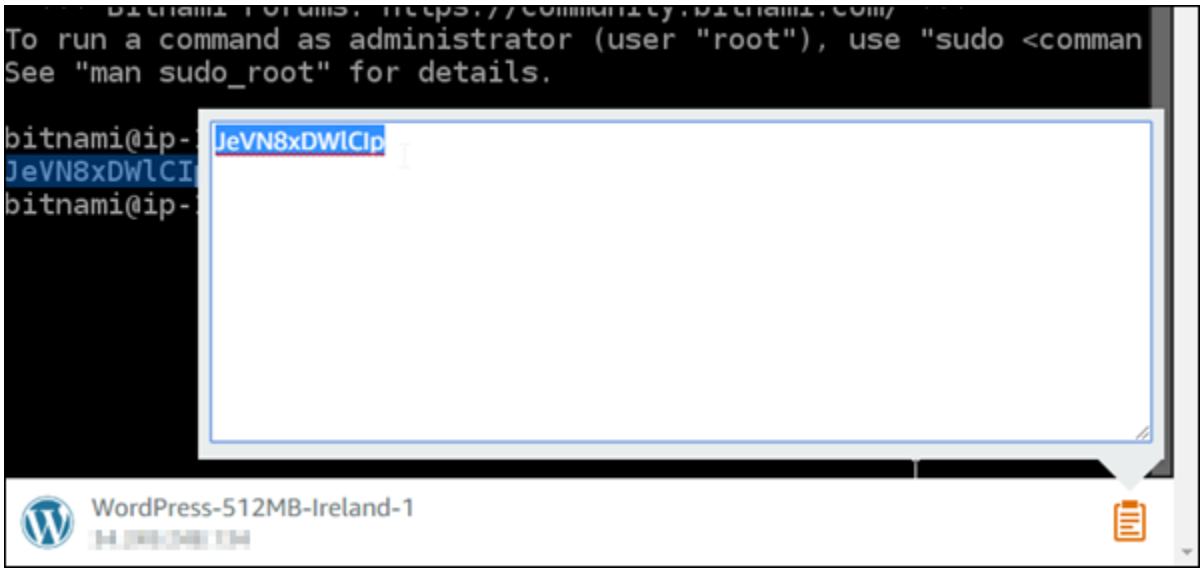
```
cat bitnami_application_password
```

> **Note**
>
> ユーザーのホームディレクトリ以外のディレクトリで作業している場合は、「`cat $HOME/bitnami_application_password`」と入力します。

次のようなレスポンスにアプリケーションのパスワードが表示されます。



5. ターミナル画面でパスワードを強調表示し、ブラウザベースの SSH クライアントウィンドウの右下でクリップボードアイコンを選択します。
6. クリップボードテキストボックスで、コピーするテキストを強調表示し、Ctrl+C または Cmd+C を押してテキストをローカルクリップボードにコピーします。



> **Important**
>
> この時点で、任意の場所にパスワードを保存します。インスタンスの Bitnami アプリケーションにサインインした後に変更することもできます。

インスタンスで Bitnami アプリケーションにサインインする

WordPress、Joomla、Magento、Drupal、GitLab CE、および Redmine 設計図から作成したインスタンスの場合は、インスタンスのパブリック IP アドレスを参照してアプリケーションにサインインします。

Bitnami アプリケーションにサインインするには

1. ブラウザウィンドウで、インスタンスのパブリック IP アドレスに移動します。

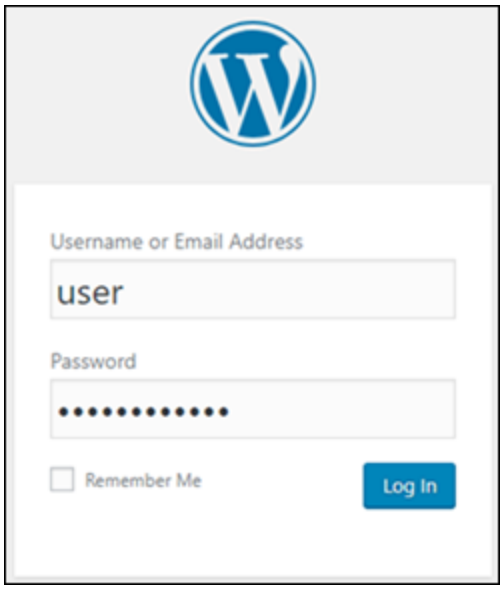
   Bitnami アプリケーションのホームページが開きます。ホームページは、インスタンスで選択した Bitnami 設計図に応じて表示されます。たとえば、これは WordPress アプリケーションのホームページです。

2. アプリケーションのホームページの右下にある Bitnami ロゴを選択し、アプリケーション情報ページに移動します。

> **Note**
>
> GitLab CE アプリケーションには Bitnami ロゴが表示されません。代わりに、GitLab CE ホームページに表示されるユーザー名とパスワードのテキストフィールドを使用してサインインします。

アプリケーション情報ページには、インスタンスのアプリケーションのユーザー名とログインページへのリンクが表示されます。

3. ページのログインリンクを選択し、インスタンスのアプリケーションのログインページに移動します。
4. 取得したユーザー名とパスワードを入力し、[Log In (ログイン)] を選択します。



次のステップ

以下のリンクを使用して、Bitnami 設計図の詳細を確認し、チュートリアルを表示します。たとえば、WordPress インスタンスのプラグインをインストールしたり、SSL 証明書を使用して HTTPS サポートを有効化したりできます。

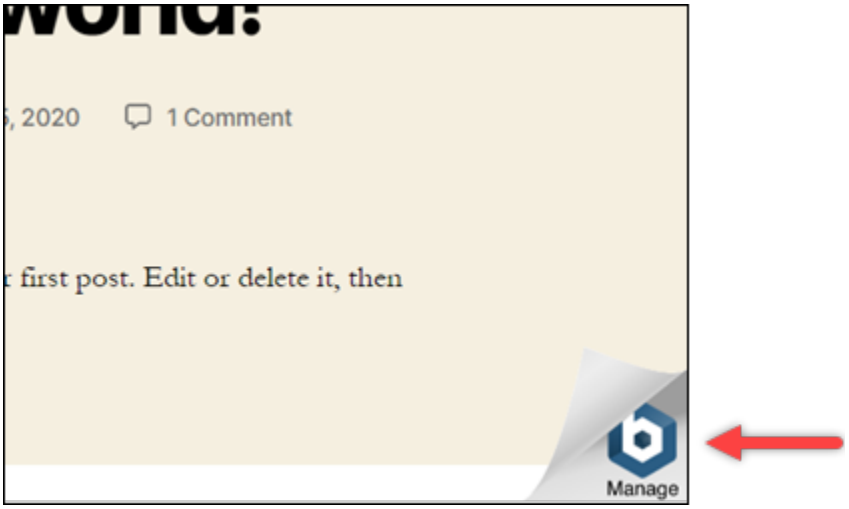
- Bitnami WordPress for Amazon Web Services

- Bitnami LAMP stack for Amazon Web Services
- Bitnami Node.js for Amazon Web Services
- Bitnami Joomla for Amazon Web Services
- Bitnami Magento for Amazon Web Services
- Bitnami MEAN stack for Amazon Web Services
- Amazon Web Services Bitnami Drupal
- Bitnami GitLab for Amazon Web Services
- Bitnami Redmine for Amazon Web Services
- Bitnami Nginx (LEMP stack) for Amazon Web Services

詳細については、「Get Started with Bitnami Applications using Amazon Lightsail」または「Using Amazon Lightsail FAQ」を参照してください。

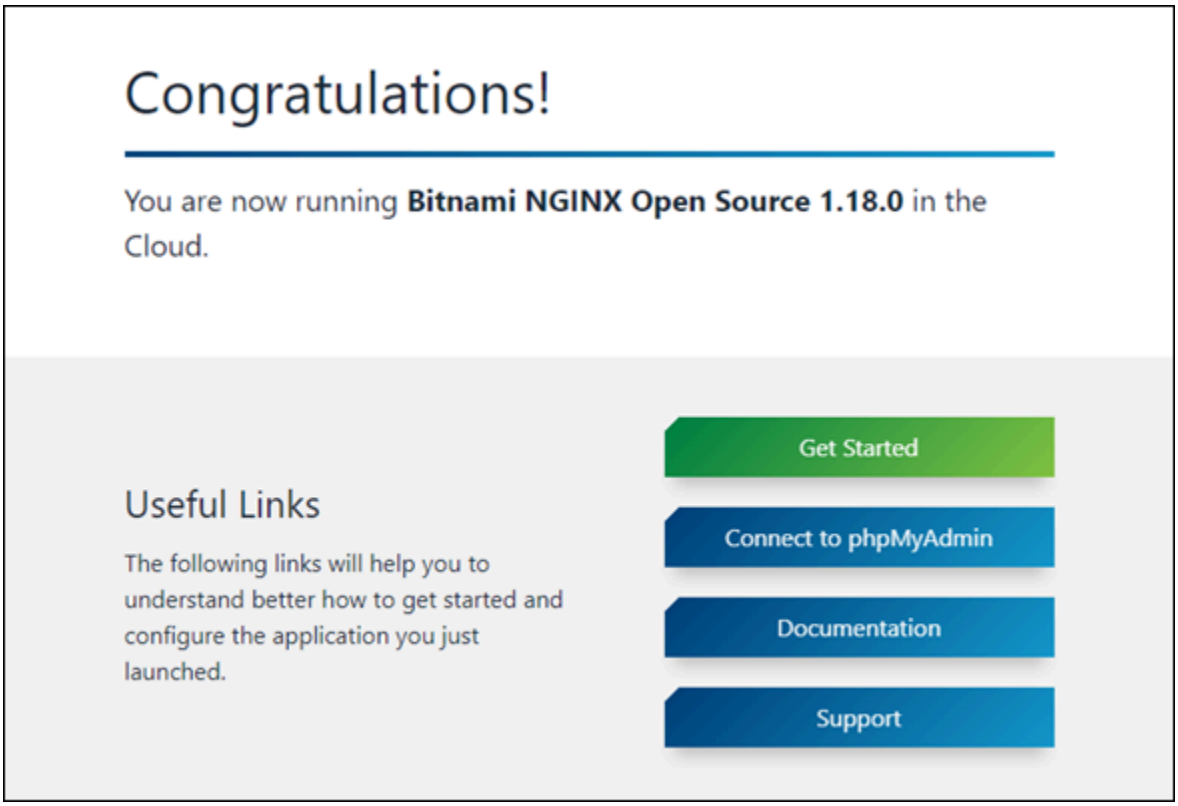
Lightsail の Bitnami のブループリント インスタンスから Bitnami バナーを削除する

Amazon Lightsail インスタンス用に選択することができる Bitnami ブループリントの一部は、アプリケーションのホームページに Bitnami バナーを表示します。「Certified by Bitnami」WordPress インスタンスの以下の例では、Bitnami バナーはホームページの右下隅に表示されています。このガイドでは、インスタンスのアプリケーションのホームページから Bitnami アイコンを完全に削除する方法を解説しています。



すべての Bitnami ブループリントアプリケーションが、アプリケーションのホームページに Bitnami バナーを表示するわけではありません。Lightsail インスタンスのホームページに移動して、Bitnami

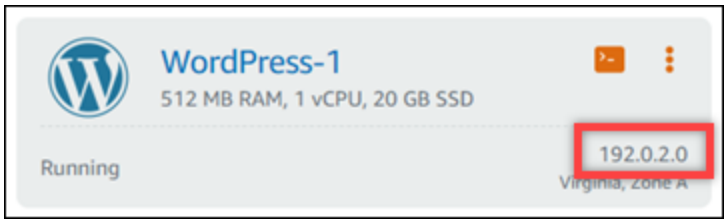
バナーが表示されるかどうかを確認してください。「Packaged by Bitnami」Nginx インスタンスの次の例では、Bitnami アイコンは表示されていません。代わりに、プレースホルダー情報ページが表示されます。このページは、最終的にインスタンスにデプロイすることを選択したアプリケーションに置き換えられます。インスタンスに Bitnami バナーが表示されない場合は、このガイドの手順に従う必要はありません。



インスタンスから Bitnami バナーを削除する

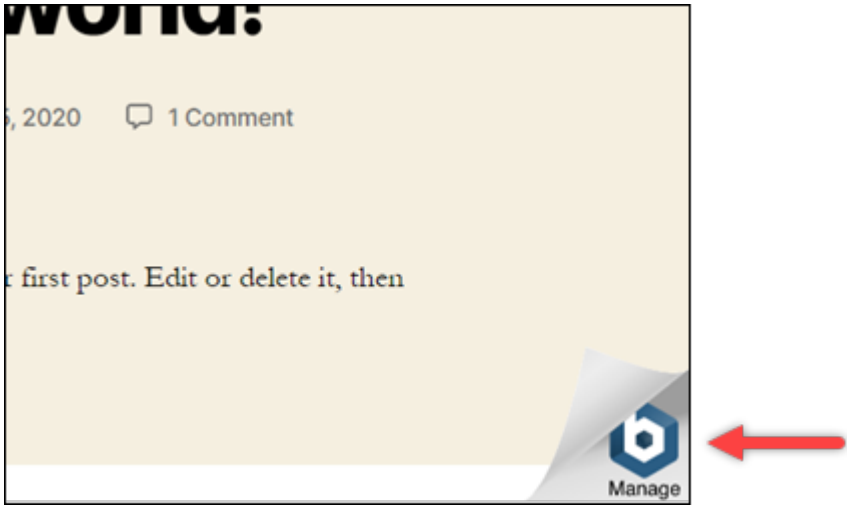
次の手順を実行して、インスタンスのアプリケーションのホームページに Bitnami アイコンが表示されていることを確認し、削除します。

1. Lightsail コンソールにサインインします。
2. Lightsail ホームページの [インスタンス] タブで、確認したいインスタンスのパブリック IP アドレスをコピーします。



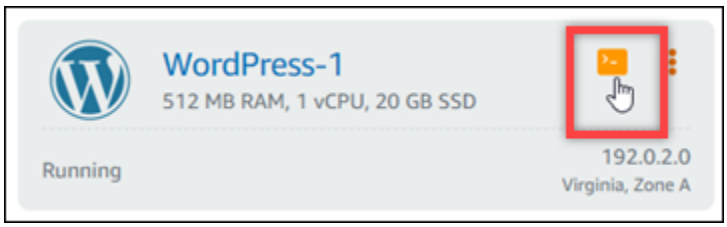
3. 新しいブラウザタブを開き、インスタンスのパブリック IP アドレスをアドレスバーに入力し、Enter を押します。
4. 以下のいずれかのオプションを確認します:

   1. Bitnami アイコンがページに表示されていない場合は、以下の手順に従う必要はありません。アプリケーションのホームページから Bitnami アイコンを削除する必要はありません。
   2. 次の例に示すように、Bitnami アイコンがページの右下隅に表示されている場合は、以下の一連のステップに従ってアイコンを削除します。



   以下の一連のステップでは、Lightsail ブラウザベースの SSH クライアントを使用してインスタンスに接続します。接続後、Bitnami 設定ツール (bnconfig) ツールを実行して、アプリケーションのホームページから Bitnami アイコンを削除します。bnconfig ツールは、 Bitnami ブループリントインスタンス上のアプリケーションを設定できるコマンドラインツールです。詳細については、Bitnami ドキュメントの「Bitnami 設定ツールの詳細を確認」を参照してください。

5. Lightsail ホームページ上にある [ブラウザ] タブに戻ります。
6. 接続先のインスタンス名の横に表示されているブラウザベースの SSH クライアントアイコンを選択します。



7. SSH クライアントがインスタンスに接続されたら、以下のいずれかのコマンドを入力します。

1. インスタンスが Apache を使用している場合は、以下のコマンドのいずれかを入力します。一方のコマンドが失敗した場合は、他方のコマンドを試してください。このコマンドの前半部分が Bitnami バナーを無効にし、後半部分が Apache サービスを再起動させます。

```
sudo /opt/bitnami/apps/wordpress/bnconfig --disable_banner 1 && sudo /opt/
bitnami/ctlscript.sh restart apache
```

```
sudo /opt/bitnami/wordpress/bnconfig --disable_banner 1 && sudo /opt/bitnami/
ctlscript.sh restart apache
```

プロセスが成功したことを確認するには、インスタンスのパブリック IP アドレスを参照し、Bitnami アイコンが表示されていないことを確認します。

WordPress Amazon Lightsail の チュートリアル

WordPress は、ユーザーがウェブサイトやブログを簡単に作成および管理できるようにするオープンソースのコンテンツ管理システムです。Lightsail WordPress で を使用する方法については、以下のチュートリアルを参照してください。

タスク

- チュートリアル:Lightsail WordPress でのインスタンスの起動と設定
- チュートリアル: Lightsail の WordPress ウェブサイトを Amazon S3 バケットに接続する
- チュートリアル: Lightsail 内の WordPress インスタンスを Amazon Aurora データベースに接続する
- チュートリアル: WordPress ウェブサイトを Lightsail の MySQL マネージドデータベースに接続する
- チュートリアル: Lightsail バケットに WordPress インスタンスを接続する
- Lightsail でコンテンツ配信ネットワークディストリビューションを使用するようにインスタンスを設定する WordPress
- Lightsail の WordPress インスタンスでメールを有効にする
- Lightsail WordPress のインスタンスで HTTPS を有効にする
- 既存の WordPressブログを Amazon Lightsail に移行する

チュートリアル:Lightsail WordPress でのインスタンスの起動と設定

インスタンス (仮想プライベートサーバー) だけが必要な場合、Amazon Lightsail はAmazon Web Services (AWS) を使い始める最も簡単な方法です。Lightsail には、インスタンス、マネージドデータベース、SSD ベースのストレージ、バックアップ (スナップショット)、データ転送、ドメイン DNS 管理、静的 IP、ロードバランサーなど、プロジェクトを迅速に開始するために必要なすべてのものが、予測可能な低価格で含まれています。

このチュートリアルでは、Lightsail WordPress でインスタンスを起動して設定する方法を学習します。カスタムドメイン名の設定、HTTPS によるインターネットトラフィックの保護、SSH によるインスタンスへの接続、 WordPress ウェブサイトへのサインインなどの手順が含まれています。このチュートリアルを完了すると、Lightsail でインスタンスを起動して実行するための基本が身に付きます。

> **Note**
>
> AWS 無料利用枠の一部として、一部のインスタンスバンドルで Amazon Lightsail を無料で開始できます。詳細については、Amazon Lightsail 料金ページの「AWS 無料利用枠」を参照してください。

コンテンツ

ステップ 1: サインアップする AWS

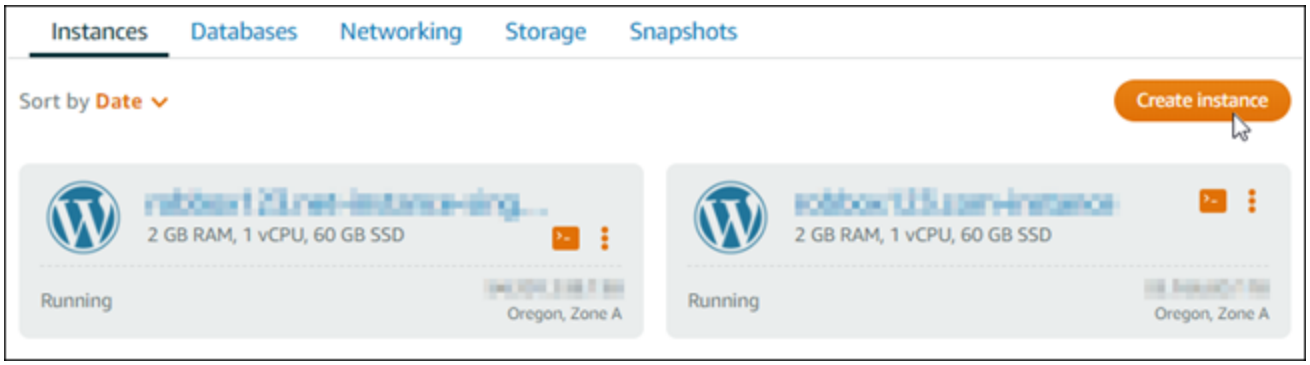
Amazon Lightsail にはが必要です。 AWS アカウントサインアップするか AWS、AWS既にアカウントをお持ちの場合はサインインしてください。

ステップ 2: WordPress インスタンスを作成する

以下のステップを実行して、 WordPress インスタンスを起動して稼働させます。詳細については、「the section called “インスタンスを作成する”」を参照してください。

用の Lightsail インスタンスを作成するには WordPress

1. Lightsail コンソールにサインインします。
2. Lightsail ホームページのインスタンスセクションで、[インスタンスを作成] を選択します。



3. インスタンスのアベイラビリティーゾーンを選択します。 AWS リージョン

4. インスタンスのイメージを次のように選択します。

a. [プラットフォームの選択] で [Linux/Unix] を選択します。

b. [ブループリントの選択] では、を選択します。WordPress

5. インスタンスプランを選択します。

   プランには、予測可能な低コストでのマシン構成 (RAM、SSD、vCPU) と、データ転送許容量が含まれます。

6. インスタンスの名前を入力します。リソース名:

   - Lightsail AWS リージョン アカウント内のそれぞれで一意である必要があります。
   - 2〜255 文字を使用する必要があります。
   - 先頭と末尾は英数字または数字を使用する必要があります。
   - 英数字、数字、ピリオド、ダッシュ、アンダースコアを使用することができます。

7. [インスタンスの作成] を選択します。

8. テストブログ投稿を表示するには、インスタンス管理ページに移動し、ページの右上隅に表示されているパブリック IPv4 アドレスをコピーします。インターネットに接続されたウェブブラウザのアドレスフィールドにアドレスを貼り付けます。ブラウザにテストブログ投稿が表示されます。

ステップ 3: WordPress インスタンスを設定する

WordPress step-by-step ガイド付きのワークフローを使用してインスタンスを設定することも、個々のタスクを完了することもできます。いずれのオプションを使用しても、以下を設定します。

- 登録済みドメイン名 — WordPress サイトには覚えやすいドメイン名が必要です。 WordPress ユーザーはこのドメイン名を指定してサイトにアクセスします。詳細については、「ドメインと DNS」を参照してください。
- DNS 管理 — ドメインの DNS レコードの管理方法を決定する必要があります。DNS レコードは、ドメインまたはサブドメインが関連付けられている IP アドレスまたはホスト名を DNS サーバーに伝えます。DNS ゾーンにはドメインの DNS レコードが含まれます。詳細については、「the section called “Lightsail の DNS”」を参照してください。
- 静的 IP アドレス — インスタンスを停止して起動すると、 WordPress インスタンスのデフォルトのパブリック IP アドレスが変更されます。静的 IP アドレスをインスタンスにアタッチすると、インスタンスを停止して起動しても同じままです。詳細については、「the section called “IP アドレス”」を参照してください。

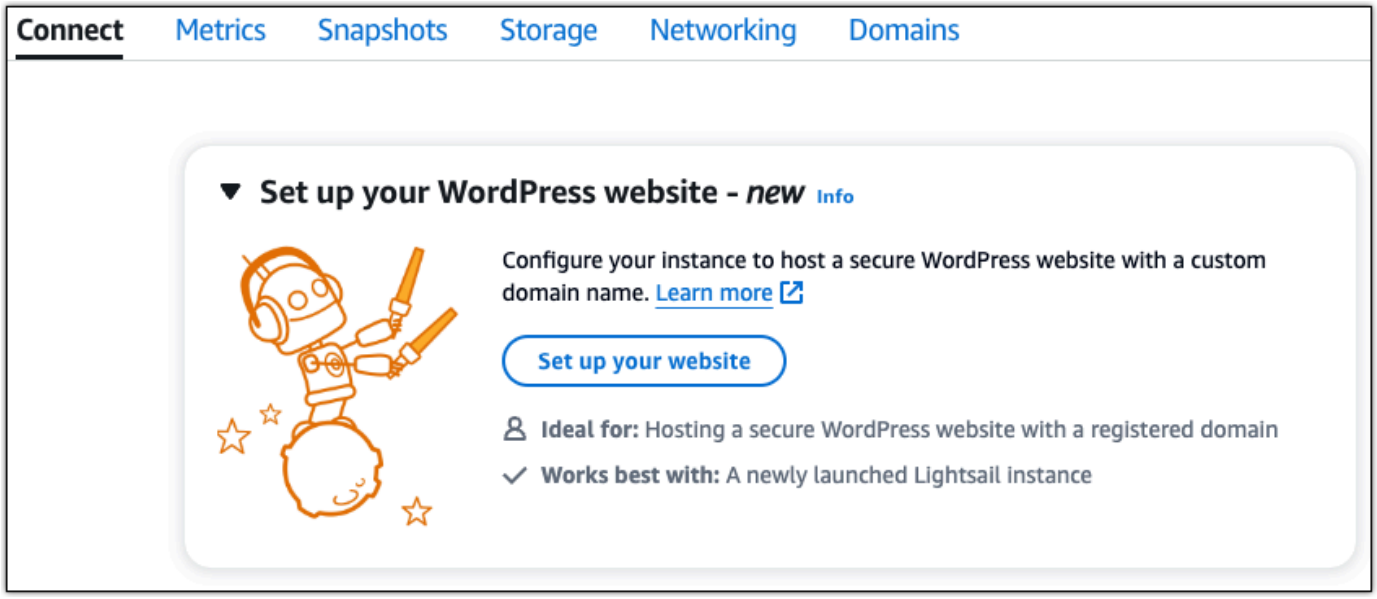
- SSL/TLS 証明書 — 検証済みの証明書を作成してインスタンスにインストールしたら、WordPress ウェブサイトの HTTPS を有効にして、登録したドメインを経由してインスタンスにルーティングされるトラフィックが HTTPS を使用して暗号化されるようにすることができます。詳細については、「the section called “HTTPS を有効にする”」を参照してください。

オプション:ガイド付きワークフロー

> **Tip**
>
> 始める前に、以下のヒントを確認してください。トラブルシューティング情報については、「WordPress トラブルシューティング設定」を参照してください。
>
> - セットアップは、2023 年 1 月 1 WordPress 日以降に作成されたバージョン 6 以降の Lightsail インスタンスをサポートします。
> - インスタンスは Running 状態である必要があります。インスタンスが起動されたばかりの場合は、SSH 接続の準備が整うまで数分お待ちください。
> - インスタンスのファイアウォールのポート 22、80、443 は、セットアップの実行中は任意の IP アドレスからの TCP 接続を許可する必要があります。詳細については、「インスタンスのファイアウォール」を参照してください。
> - Apex ドメイン (`example.com`) `www` とそのサブドメイン () からのトラフィックを指す DNS レコードを追加または更新する場合、それらはインターネット全体に伝播する必要があります。`www.example.com`DNS の変更が反映されたかどうかは、nslookup やからの DNS 検索などのツールを使用して確認できます。MxToolbox
> - 2023 年 1 月 1 日より前に作成された Wordpress インスタンスには、廃止予定の Certbot 個人Package アーカイブ (PPA) リポジトリが含まれている可能性があり、これによりウェブサイトの設定が失敗する可能性があります。セットアップ中にこのリポジトリが存在する場合、既存のパスから削除され、インスタンスの次の場所にバックアップされます。`~/opt/bitnami/lightsail/repo.backup`廃止された PPA の詳細については、Canonical ウェブサイトの Certbot PPA を参照してください。
> - Let's Encrypt の証明書は 60 日から 90 日ごとに自動的に更新されます。
> - セットアップ中は、インスタンスを停止したり変更したりしないでください。インスタンスの設定には最大 15 分かかることがあります。各ステップの進行状況は、インスタンス接続タブで確認できます。

ウェブサイトセットアップウィザードを使用してインスタンスを設定するには

1. インスタンス管理ページの「Connect」タブで、「ウェブサイトを設定」を選択します。



2. [ドメイン名を指定] では、既存の Lightsail 管理ドメインを使用するか、Lightsail に新しいドメインを登録するか、別のドメインレジストラを使用して登録したドメインを使用します。[このドメインを使用する] を選択して次のステップに進みます。
3. [DNS の設定] で、次のいずれかを実行します。
   - Lightsail DNS ゾーンを使用するには、Lightsail マネージドドメインを選択してください。[この DNS ゾーンを使用する] を選択して次のステップに進みます。
   - ドメインの DNS レコードを管理するホスティングサービスを使用するには、「サードパーティードメイン」を選択します。後で使用することを決めた場合に備えて、Lightsail アカウントに一致する DNS ゾーンが作成されることに注意してください。[サードパーティ DNS を使用] を選択して次のステップに進みます。
4. [固定 IP アドレスの作成] に、固定 IP アドレスの名前を入力し、[静的 IP アドレスの作成] を選択します。
5. [ドメイン割り当ての管理] で [割り当てを追加] を選択し、ドメインの種類を選択して、[追加] を選択します。[続行] を選択して、次のステップに進みます。
6. [SSL/TLS 証明書の作成] では、ドメインとサブドメインを選択し、メールアドレスを入力し、[Lightsail に権限を付与してインスタンスに Let's Encrypt 証明書を設定する] を選択し、[証明書の作成] を選択します。Lightsail リソースの設定を開始します。

セットアップ中は、インスタンスを停止したり変更したりしないでください。インスタンスの設定には最大 15 分かかることがあります。各ステップの進行状況は、インスタンス接続タブで確認できます。

7. ウェブサイトの設定が完了したら、ドメイン割り当て手順で指定した URL WordPress がサイトを開くことを確認します。

オプション:個別のタスク

個々のタスクを完了してインスタンスを設定するには

1. 静的 IP アドレスの作成

   インスタンス管理ページの [ネットワーク] タブで [静的 IP の作成] を選択します。静的 IP の場所とインスタンスが自動的に選択されます。固定 IP アドレスの名前を指定し、[作成してアタッチ] を選択します。

2. DNS ゾーンの作成

   ナビゲーションペインで [ドメインと DNS] を選択します。[DNS ゾーンの作成] を選択し、ドメインを入力して [DNS ゾーンの作成] を選択します。ウェブトラフィックが現在ドメインにルーティングされている場合は、ドメインの現在の DNS ホスティングプロバイダーのネームサーバーを変更する前に、既存の DNS レコードがすべて Lightsail DNS ゾーンに存在することを確認してください。これにより、Lightsail DNS ゾーンへの転送後もトラフィックは途切れることなく流れ続けます。

3. ドメイン割り当ての管理

   DNS ゾーンのページの [割り当て] タブで、[割り当てを追加] を選択します。ドメインまたはサブドメインを選択し、インスタンスを選択して固定 IP アドレスをアタッチし、[Assign] を選択します。

   > **Tip**
   > ドメインがインスタンスへのトラフィックのルーティングを開始する前に、これらの変更がインターネットに反映されるまでしばらくお待ちください WordPress 。

4. SSL/TLS 証明書を作成してインストールします。

   step-by-step 手順については、を参照してください。the section called “HTTPS を有効にする”

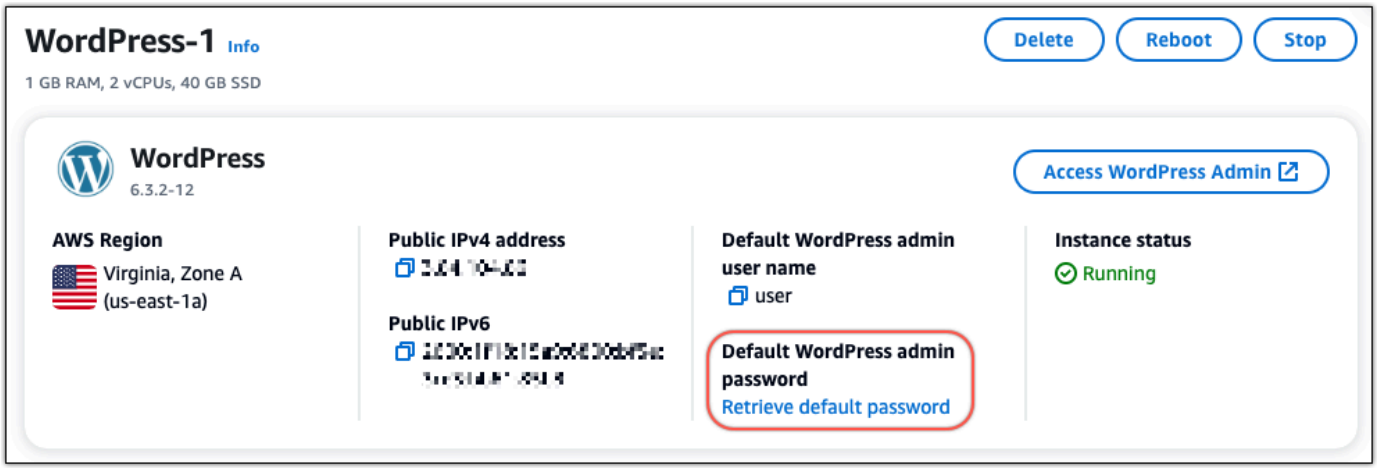
5. ドメイン割り当て手順で指定した URL WordPress がサイトを開くことを確認します。

ステップ 4: WordPress Web サイトの管理者パスワードを取得する

WordPress ウェブサイトの管理ダッシュボードにサインインするためのデフォルトパスワードは、インスタンスに保存されます。パスワードを取得するには、次の手順を実行します。

WordPress 管理者のデフォルトパスワードを取得するには

1. インスタンスのインスタンス管理ページを開きます。 WordPress
2. WordPressパネルで [デフォルトパスワードを取得] を選択します。これにより、ページの下部にある Access のデフォルトパスワードが展開されます。



3. [起動] を選択します。 CloudShellページの下部にパネルが開きます。
4. [Copy] を選択し、 CloudShell 内容をウィンドウに貼り付けます。 CloudShell プロンプトにカーソルを置いて Ctrl+V を押すか、右クリックしてメニューを開き、[貼り付け] を選択します。
5. ウィンドウに表示されたパスワードを書き留めておきます。 CloudShell WordPress Web サイトの管理ダッシュボードにログインするには、これが必要です。

```
[cloudshell-user@ip-[redacted] ~]$ AWS_REGION=us-east-1 ~/lightsail_connect WordPress-1 cat bitnami_application_password
JKzh8wB5FAR!
```

ステップ 5: ウェブサイトの管理ダッシュボードにログインします。 WordPress

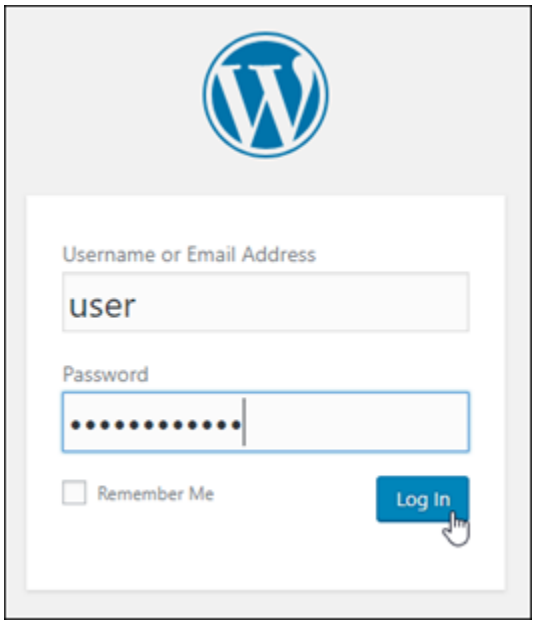
WordPress Web サイトの管理ダッシュボードのパスワードがわかったので、ログインできます。管理ダッシュボードでは、ユーザーパスワードの変更、プラグインのインストール、ウェブサイトのテーマの変更などを行うことができます。

WordPressWeb サイトの管理ダッシュボードにログインするには、次の手順を実行します。

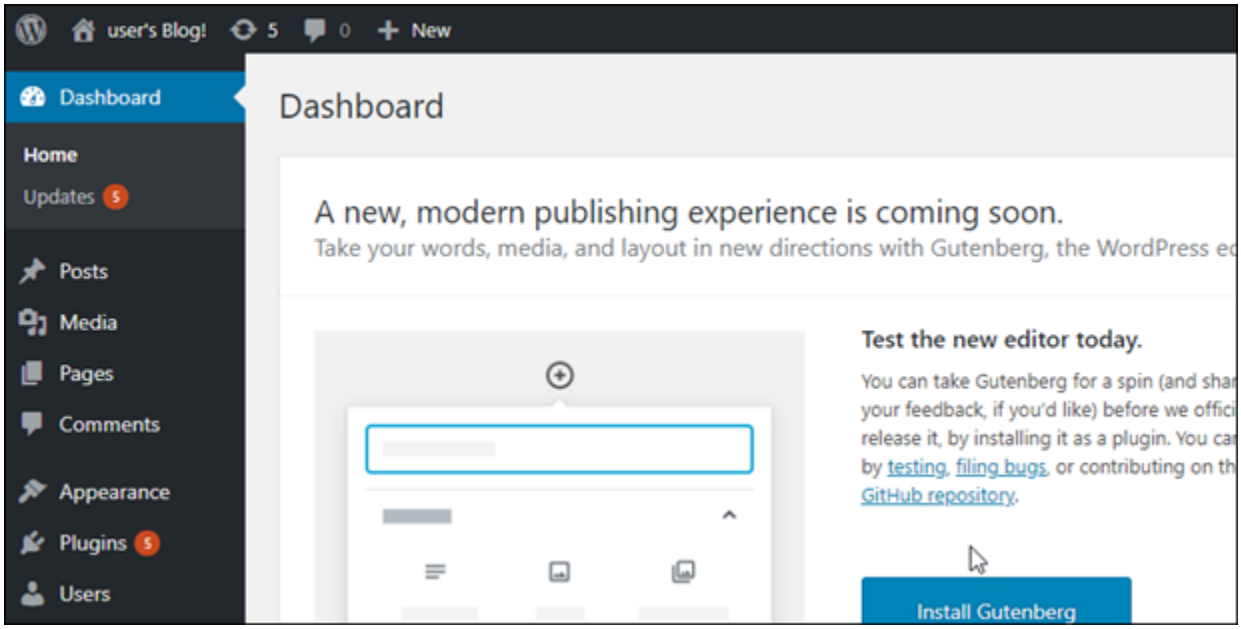
管理ダッシュボードにサインインするには

1. インスタンスのインスタンス管理ページを開きます。 WordPress
2. WordPressパネルで [WordPress 管理者にアクセス] を選択します。
3. 「 WordPress 管理者ダッシュボードへのアクセス」パネルの「パブリック IP アドレスを使用」で、次の形式のリンクを選択します。

   http://*##### IPV4-####*。 /wp-管理者
4. [ユーザー名] または [メールアドレス] に、と入力します。**user**
5. [パスワード] には、前のステップで取得したパスワードを入力します。
6. [ログイン] を選択します。



これで WordPress Web サイトの管理ダッシュボードにサインインし、管理アクションを実行できます。 WordPress Web サイトの管理について詳しくは、ドキュメントの WordPressCodex を参照してください。 WordPress



追加情報

Amazon Lightsail WordPress でインスタンスを起動した後に実行できる追加のステップは次のとおりです。

- the section called “CDN を設定する”
- Linux または Unix インスタンスのスナップショットを作成する
- インスタンスまたはディスクの自動スナップショットの有効化または無効化
- 追加のブロックストレージディスクを作成して Linux ベースの インスタンスにアタッチする

チュートリアル: Lightsail の WordPress ウェブサイトを Amazon S3 バケットに接続する

このチュートリアルでは、Amazon Lightsail インスタンスで実行されている WordPress ウェブサイトを Amazon Simple Storage Service (Amazon S3) に接続して、ウェブサイトの画像とアタッチメントを保存するのに必要なステップについて説明します。そのためには、Amazon Web Services (AWS) アカウントの認証情報のセットを使用して、WordPress プラグインを設定します。これで、プラグインによって Amazon S3 バケットが作成され、インスタンスのディスクの代わりに、バケットをウェブサイトの画像とアタッチメントに使用するようにウェブサイトが設定されます。

目次

ステップ 1: 前提条件を満たす

開始する前に、Lightsail に WordPress インスタンスを作成し、実行状態になっていることを確認します。詳細については、「チュートリアル: WordPress インスタンスを起動して設定する」を参照してください。

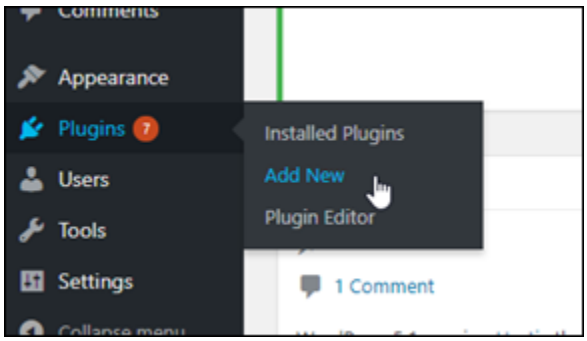
ステップ 2: WordPress ウェブサイトに WP Offload Media プラグインをインストールする

プラグインを使用して、Amazon S3 バケットを使用するようにウェブサイトを設定する必要があります。設定するために利用できるプラグインは多数あります。そのようなプラグインのひとつに WP Offload Media Lite があります。

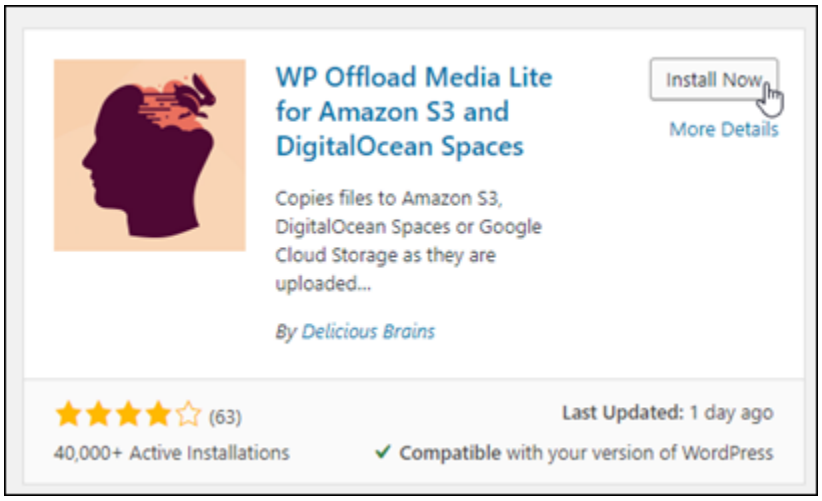
次のステップに従って、WordPress ウェブサイトに WP Offload Media プラグインをインストールします。

1. 管理者として WordPress のダッシュボードにサインインします。

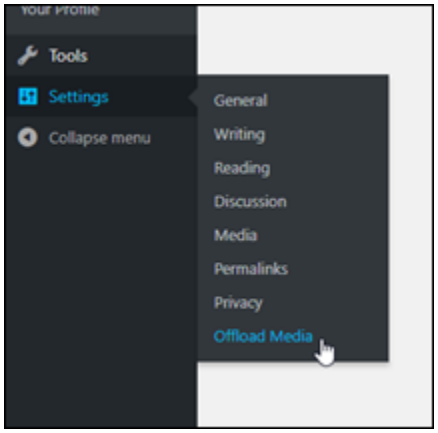
   詳細については、「Amazon Lightsailの Bitnami インスタンス向けにアプリケーションのユーザー名とパスワードを取得する」を参照してください。
2. 左側のナビゲーションメニューの [プラグイン] にカーソルを合わせ、[Add New (新規追加)] を選択します。



3. [WP Offload Media Lite] を検索します。
4. 検索結果の中から WP Offload Media プラグインの横の [Install Now] (今すぐインストール)を選択します。



5. プラグインのインストールが完了したら、[アクティベート] を選択します。
6. 左ナビゲーションメニューで、[設定]、[Offload Media] の順に選択します。



7. [オフロードメディア] ページで、ストレージプロバイダーとして [Amazon S3] を選択し、[wp-config.php でアクセスキーを定義する] を選択します。

   このオプションでは、インスタンスの `wp-config.php` に AWS アカウント認証情報を追加する必要があります。これらのステップについては、このチュートリアルの後半で説明します。

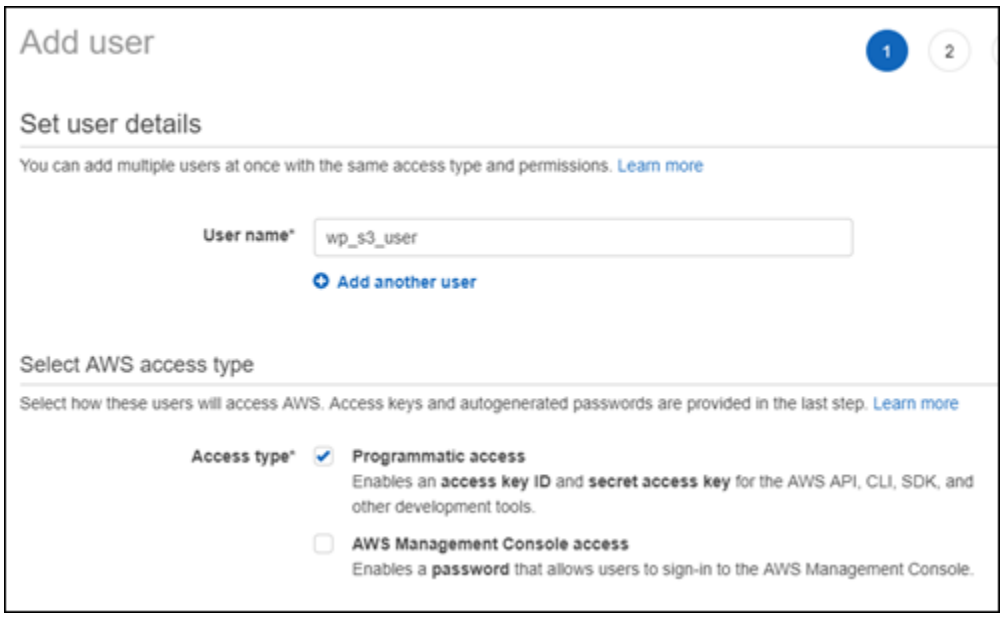
[Offload Media] ページは開いたままにします。このチュートリアルの後半で使用します。このチュートリアルの「ステップ 3: IAM ユーザーとポリシーを作成する」セクションに進みます。

ステップ 3: IAM ユーザーとポリシーを作成する

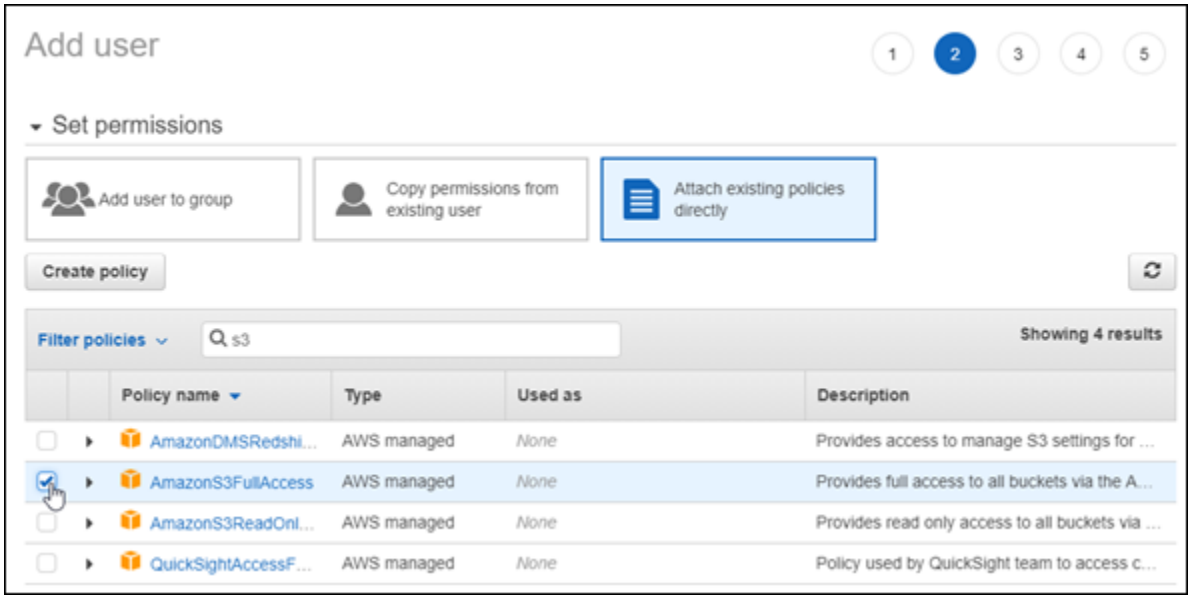
WP Offload Media プラグインは、AWS のアカウントにアクセスして、Amazon S3 バケットを作成し、ウェブサイトの画像とアタッチメントをアップロードします。

次のステップに従って、WP Offload Media プラグインの新しい AWS Identity and Access Management (IAM) ユーザーとポリシーを作成します。

1. 新しいブラウザタブを開き、IAM コンソールにサインインします。
2. 左のナビゲーションメニューの [ユーザー] を選択します。
3. [Add user] (ユーザーを追加) を選択します。
4. [ユーザー名] テキストボックスに、新しいユーザーの名前を入力します。`wp_s3_user` や `wp_offload_media_plugin_user` など、分かりやすい名前を入力して、将来メンテナンスを実行するときに簡単に識別できるようにします。
5. [アクセスの種類] セクションで、[プログラムによるアクセス] を選択します。



6. [Next: Permissions] (次へ: 許可) を選択します。
7. [既存のポリシーを直接アタッチします] を選択し、[S3] を検索して、検索結果で [AmazonS3FullAccess] を選択します。



8. [次へ: タグ]、[次へ: 確認] の順に選択します。
9. ページに表示されるユーザーの詳細を確認し、[ユーザーの作成] を選択します。
10. ユーザーの [アクセスキーID] と [シークレットアクセスキー] をメモするか、[.csv のダウンロード] を選択してこれらの値のコピーをローカルドライブに保存します。これらの値は、WordPress インスタンスで `wp-config.php` ファイルを編集する際、次のいくつかの手順で必要になります。

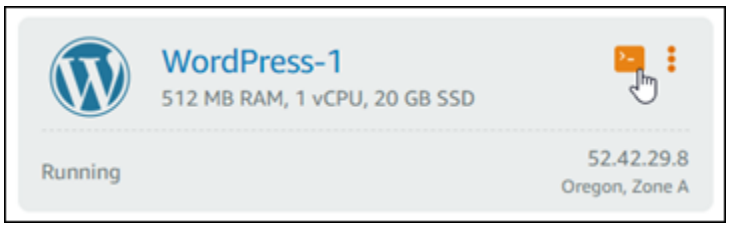
ステップ 4: WordPress 設定ファイルを編集する

次の手順に従って、Lightsail コンソールでブラウザーベースの SSH クライアントを使用して WordPress インスタンスに接続し、wp-config.php ファイルを編集します。

wp-config.php ファイルには、データベース接続情報など、ウェブサイトの基本設定の詳細が含まれています。

> **Note**
>
> 独自の SSH クライアントを使用してインスタンスに接続することもできます。詳細については、「Amazon Lightsail で PuTTY をダウンロードし、SSH を使用して接続するようにセットアップする」を参照してください。

1. Lightsail コンソールにサインインします。
2. WordPress インスタンスのブラウザベースの SSH クライアントアイコンを選択します。



3. 表示される SSH クライアントウィンドウで、次のコマンドを入力して、問題が発生した場合に備えて wp-config.php ファイルのバックアップを作成します。

```
sudo cp /opt/bitnami/wordpress/wp-config.php /opt/bitnami/wordpress/wp-config.php.backup
```

4. 次のコマンドを入力して、テキストエディタ nano を使用し、wp-config.php ファイルを開きます。

```
nano /opt/bitnami/wordpress/wp-config.php
```

5. テキスト /* That's all, stop editing! Happy blogging. */ の上に次のテキストを入力します。

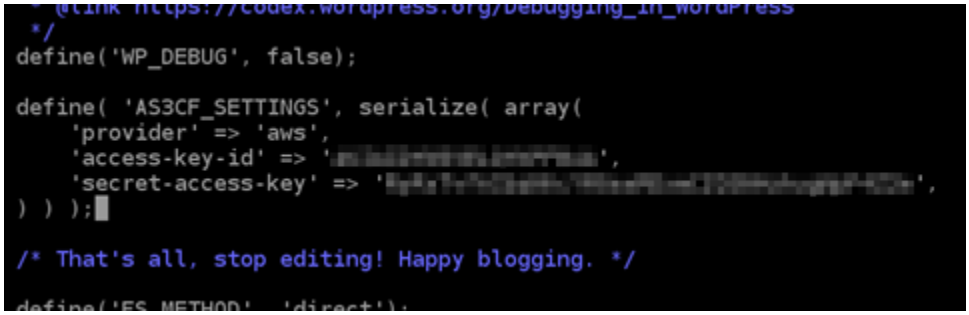
   AccessKeyID をアクセスキー ID に、*SecretAccessKey* をこれらのステップの前半で作成した IAM ユーザーのシークレットアクセスキーに置き換えます。

```
define( 'AS3CF_SETTINGS', serialize( array(
    'provider' => 'aws',
    'access-key-id' => 'AccessKeyID',
    'secret-access-key' => 'SecretAccessKey',
) ) );
```

例:

```
define( 'AS3CF_SETTINGS', serialize( array(
    'provider' => 'aws',
    'access-key-id' => 'AKIAIOSFODNN7EXAMPLE',
    'secret-access-key' => 'wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY',
) ) );
```

結果は次の例のようになります。



6. **Ctrl+X** を押して Nano を終了してから **Y**、**Enter** の順に押して編集内容を `wp-config.php` ファイルに保存します。
7. 次のコマンドを入力して、インスタンス上のサービスを再起動します。

```
sudo /opt/bitnami/ctlscript.sh restart
```

サービスが再起動されると次のような結果が表示されます。

SSH ウィンドウを閉じ、このチュートリアルの前半で開いたままにした [Offload Media] ページに戻ります。これで、WP Offload Media プラグインを使用して Amazon S3 バケットを作成する準備ができました。

ステップ 5: WP Offload Media プラグインを使用して Amazon S3 バケットを作成する

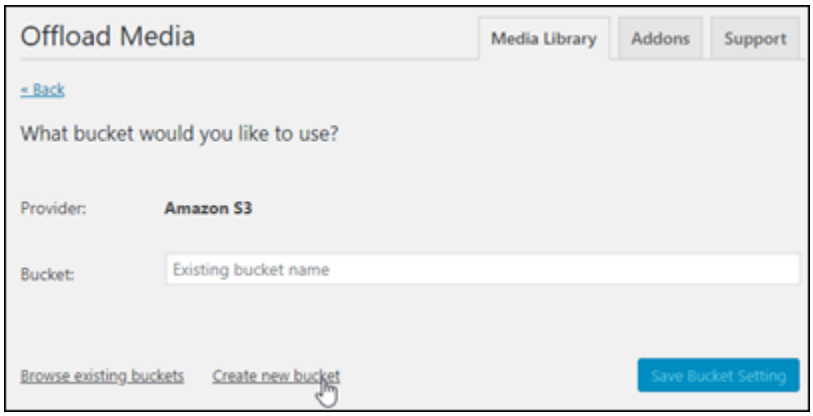
AWS 認証情報で `wp-config.php` ファイルが設定されたので、[Offload Media] ページに戻ってプロセスを完了します。

次のステップを実行して、WP Offload Media プラグインを使用して Amazon S3 バケットを作成します。

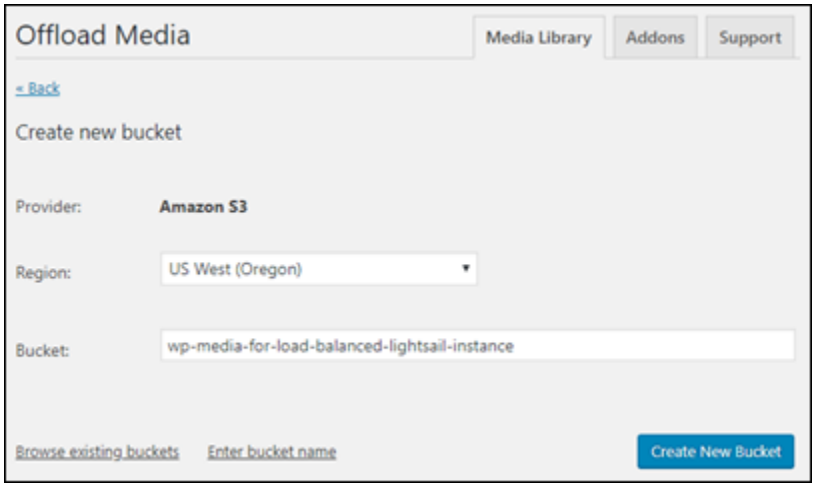
1. [Offload Media] ページを更新するか、[Nesxt] を選択します。

   Amazon S3 プロバイダーが設定されていることがわかります。

2. [新しいバケットの作成] を選択します。

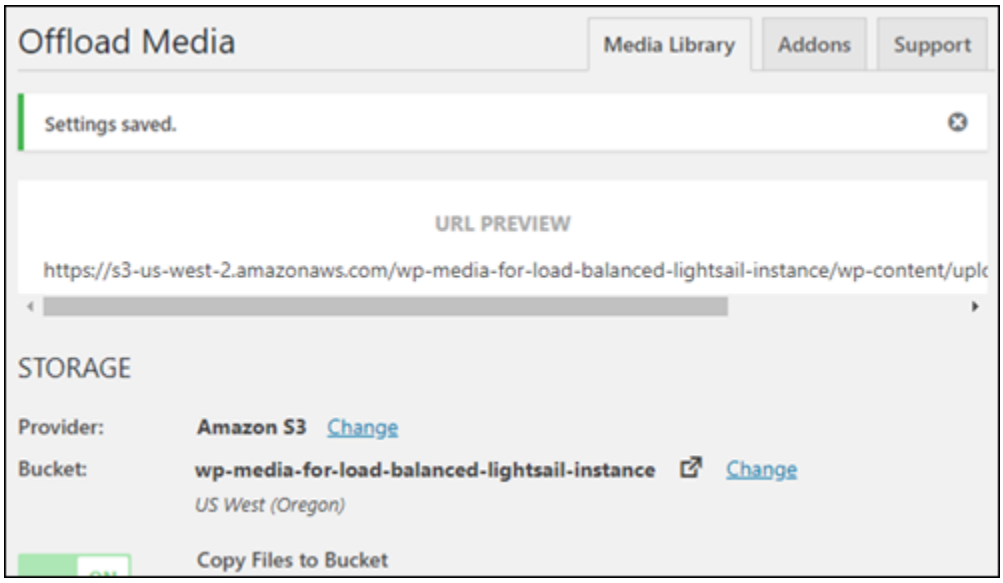


3. [リージョン] ドロップダウンメニューで、目的の AWS リージョンを選択します。WordPress インスタンスがあるリージョンと同じリージョンを選択することをお勧めします。
4. [バケット] テキストボックスに、新しい S3 バケットの名前を入力します。



5. [新しいバケットの作成] を選択します。

   ページが更新され、新しいバケットが作成されたことを確認します。表示される設定を確認し、WordPress ウェブサイトの動作に合わせて調整します。



   今後、ブログ投稿に追加された画像やアタッチメントは、作成した Amazon S3 バケットに自動的にアップロードされます。

ステップ 6：次のステップ

WordPress ウェブサイトを Amazon S3 バケットに接続したら、WordPress インスタンスのスナップショットを作成して、行った変更をバックアップする必要があります。詳細については、「Linux または Unix インスタンスのスナップショットを作成する」を参照してください。

チュートリアル: Lightsail 内の WordPress インスタンスを Amazon Aurora データベースに接続する

投稿、ページ、およびユーザーのウェブサイトデータは、Amazon Lightsail の WordPress インスタンス上で実行するデータベースに保存されています。WordPress インスタンスに障害が発生した場合、データが回復不可能になる場合があります。このシナリオを回避するには、Amazon Relational Database Service (Amazon RDS) の Amazon Aurora データベースにウェブサイトのデータを転送する必要があります。

Amazon Aurora はクラウド用に構築された MySQL と PostgreSQL 互換のリレーショナルデータベースです。これは従来のエンタープライズデータベースのパフォーマンスと可用性に、オープンソースデータベースのシンプルさと費用対効果を組み合わせています。Aurora は Amazon RDS の一部として提供されています。Amazon RDS は、クラウドでリレーショナルデータベースを簡単に設定、運用、およびスケールすることができるマネージドデータベースサービスです。詳細については、「Amazon Relational Database Service ユーザーガイド」と「Aurora の Amazon Aurora ユーザーガイド」を参照してください。

このチュートリアルでは、Lightsail 内の WordPress インスタンスからウェブサイトデータベースを Amazon RDS 内の Aurora マネージドデータベースに接続する方法について説明します。

目次

ステップ 1: 前提条件を満たす

開始する前に次の前提条件を完了します。

1. Lightsail で WordPress インスタンスを作成し、アプリケーションを設定します。続行する前に、インスタンスは実行中状態になっていることを確認してください。詳細については、「チュートリアル: Amazon Lightsail で WordPress インスタンスを起動して設定する」を参照してください。

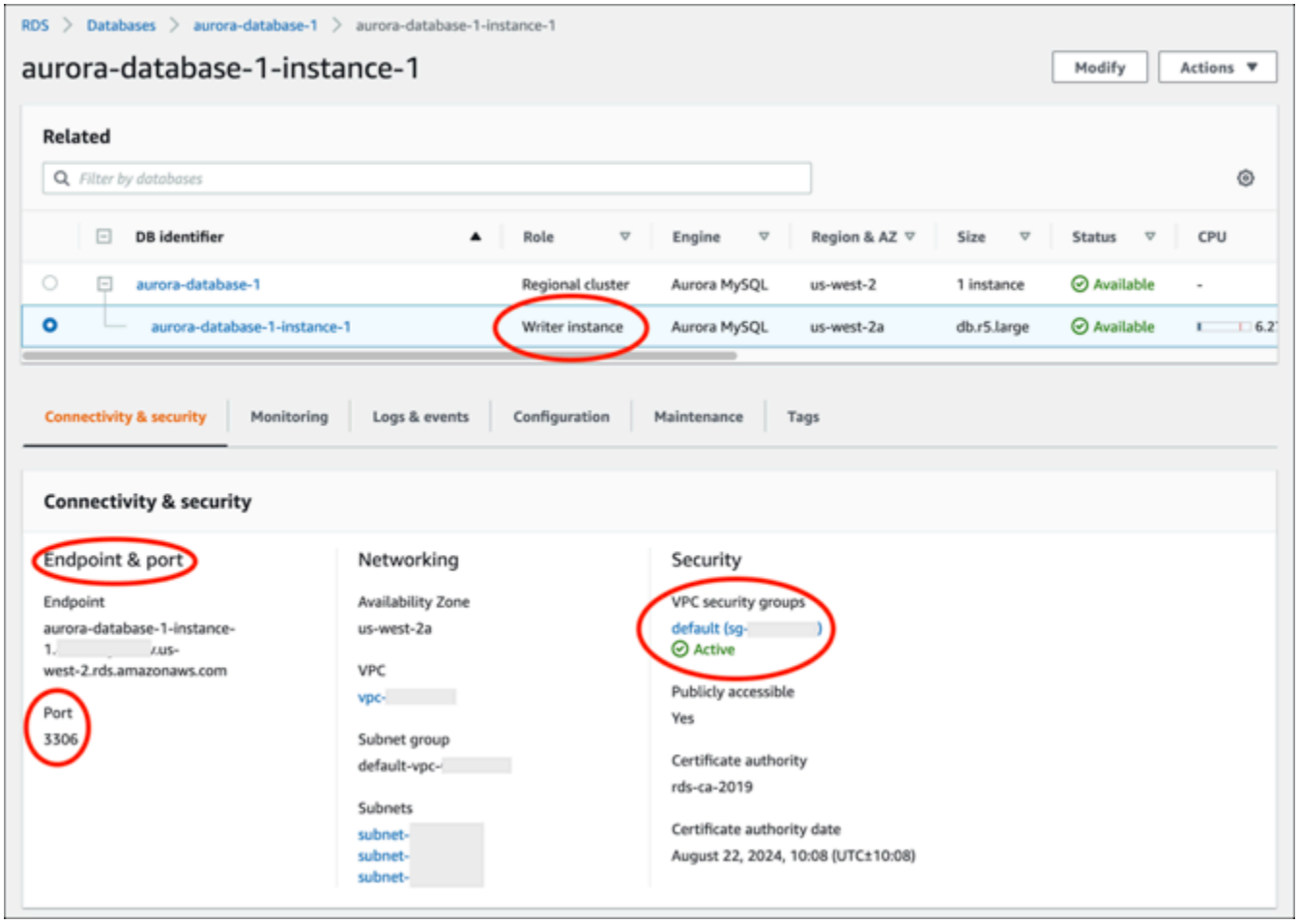
2. Lightsail アカウントで VPC ピアリングを有効にします。詳細については、「Lightsail 外の AWS リソースを使用するためにピア接続をセットアップする」を参照してください。
3. Amazon RDS に Aurora マネージドデータベースを作成します。データベースは、WordPress リソースと同じ AWS リージョン にある必要があります。続行する前に、データベースが実行中状態になっていることを確認してください。詳細については、「Amazon Aurora ユーザーガイド」の「Amazon Aurora で使用開始」を参照してください。

ステップ 2: Aurora データベースのセキュリティグループを設定する

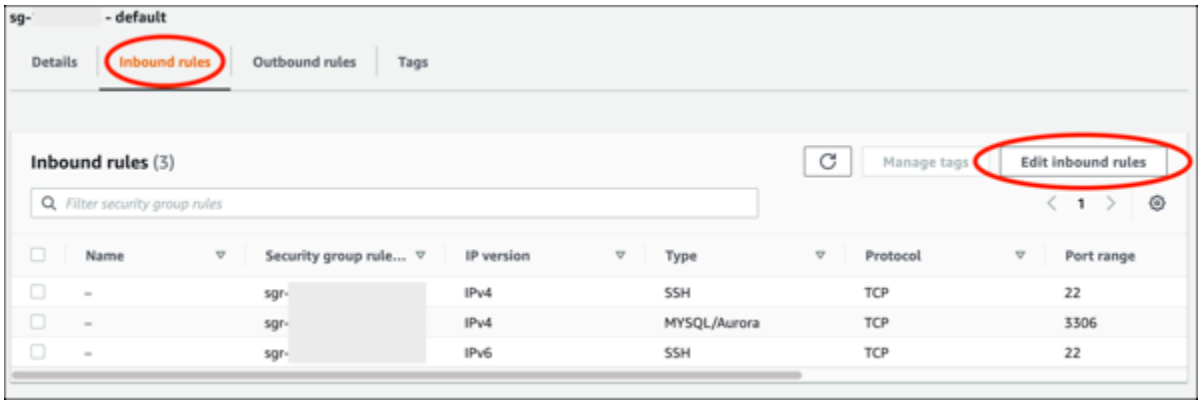
AWS セキュリティグループは AWS リソースの仮想ファイアウォールとして機能します。Amazon RDS 内の Aurora データベースに接続できる送受信トラフィックを制御します。詳細については、「Amazon Virtual Private Cloud ユーザーガイド」の「セキュリティグループを使用してリソースへのトラフィックを制御する」を参照してください。

WordPress インスタンスが Aurora データベースへの接続を確立できるよう、以下の手順を完了してセキュリティグループを設定します。

1. Amazon RDS コンソールにサインインします。
2. ナビゲーションペインで、[Databases] (データベース) を選択します。
3. WordPress インスタンスが接続する Aurora データベースの[ライターインスタンス]を選択します。
4. [Connectivity & security (接続とセキュリティ)] タブを選択します。
5. [Endpoint & port] (エンドポイントとポート) セクションに表示されるライターインスタンスのエンドポイント名とポートを記録します。これらの情報は、データベースに接続する Lightsail インスタンスを設定するときに必要になります。
6. [Security] (セキュリティ) セクションでアクティブな VPC セキュリティグループのリンクを選択します。データベースのセキュリティグループにリダイレクトされます。



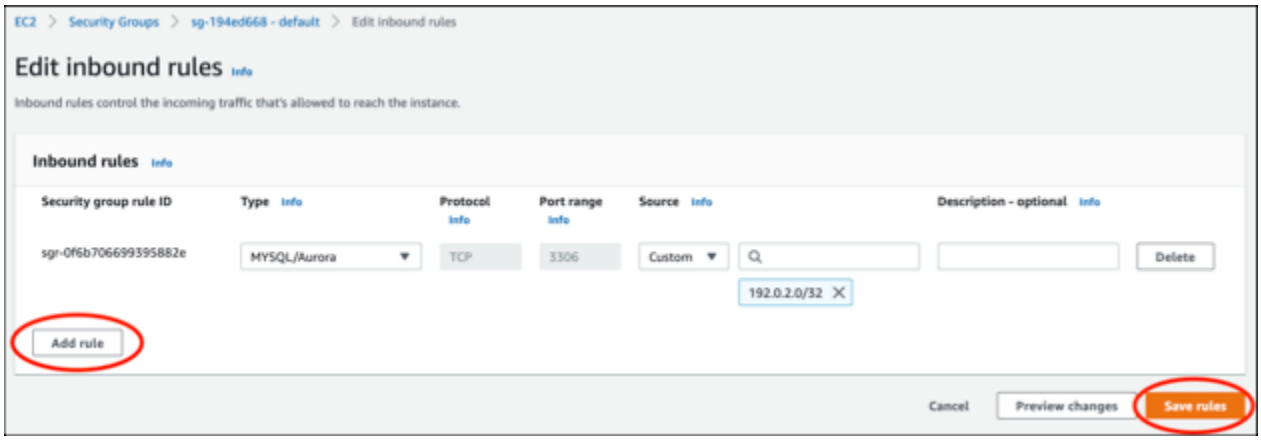
7. Aurora データベースのセキュリティグループが選択されていることを確認します。
8. [Inbound rules] (インバウンドルール) タブを開きます。
9. [Edit inbound rules] (インバウンドルールの編集) を選択します。



10. [Edit inbound rules] (インバウンドルールの編集) ページで [Add rule] (ルールの追加) を選択します。
11. 次のいずれかのステップを完了します。

- デフォルトの MySQL ポート 3306 を使用する場合は、[Type] (タイプ) ドロップダウンメニューから [MySQL/Aurora] を選択します。
- データベースのカスタムポートを使用する場合は、[Type] (タイプ) ドロップダウンメニューから [Custom TCP] (カスタム TCP) を選択し、[Port Range] (ポート範囲) テキストボックスにポート番号を入力します。

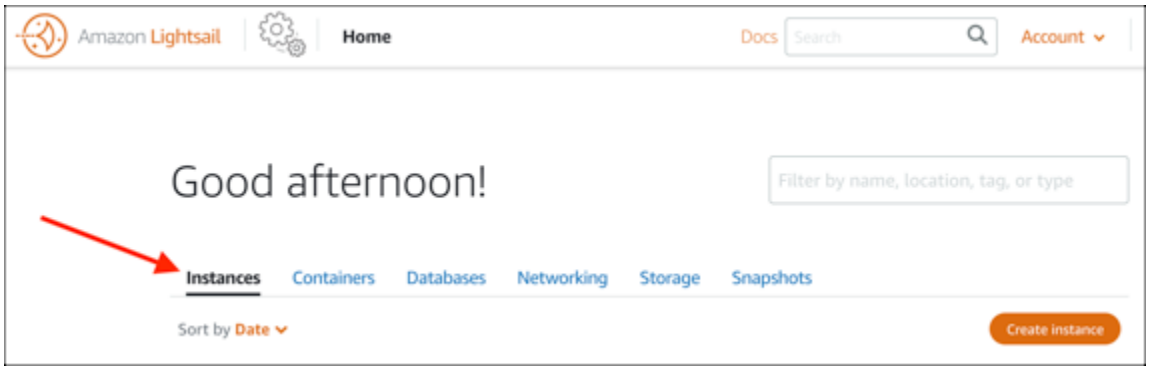
12. [Source] (ソース) テキストボックスに WordPress インスタンスのプライベート IP アドレスを追加します。IP アドレスは、 CIDR 表記で入力する必要があります (/32 を追加する必要があります)。例えば、`192.0.2.0` を許可するには「`192.0.2.0/32`」と入力します。
13. [Save Rules] (ルールの保存) を選択します。



ステップ 3: Lightsail インスタンスから Aurora データベースに接続する

以下の手順を完了して、Lightsail インスタンスから Aurora データベースに接続できることを確認します。

1. Lightsail コンソールにサインインします。
2. Lightsail ホームページで、[Instances (インスタンス)] タブを選択します。

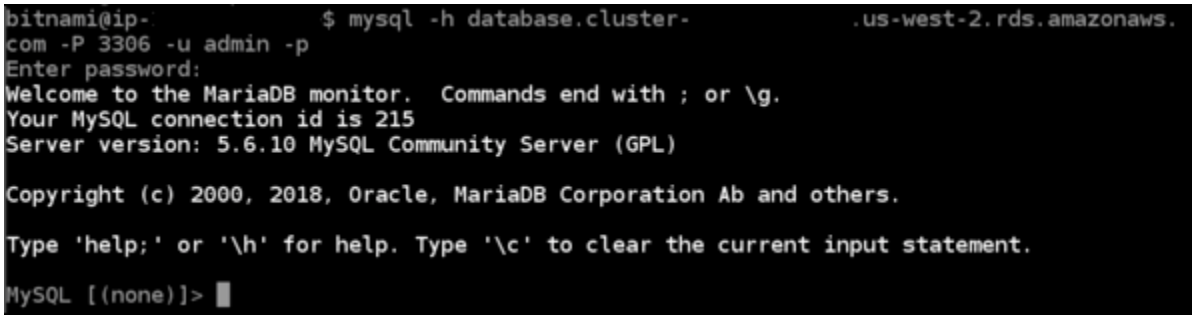


3. SSH を使用して接続する WordPress インスタンスのブラウザベースの SSH クライアントアイコンを選択します。

4. インスタンスに接続したら、次のコマンドを入力して、Aurora データベースに接続します。このコマンドで、*DatabaseEndpoint* を実際の Aurora データベースのエンドポイントアドレスで置き換え、*Port* をデータベースのポートで置き換えます。*MyUsername* は、データベースを作成したときに入力したユーザーの名前で置き換えます。

```
mysql -h DatabaseEndpoint -P Port -u MyUserName -p
```

インスタンスが Aurora データベースにアクセスおよび接続できれば、次の例のような応答が表示されます。



この応答が表示されない場合、またはエラーメッセージが表示された場合は、Aurora データベースのセキュリティグループを設定して、Lightsail インスタンスのプライベート IP アドレスからその接続を許可する必要がある場合があります。詳細については、このガイドの「Aurora データベースのセキュリティグループを設定する」を参照してください。

ステップ 4: データベースを WordPress インスタンスから Aurora データベースに転送する

インスタンスからデータベースに接続できることを確認した後は、WordPress ウェブサイトデータを Aurora データベースに転送する必要があります。

1. Lightsail コンソールにサインインします。
2. [Instances] (インスタンス) タブで、WordPress インスタンス用ブラウザベースの SSH クライアントを選択します。

3. ブラウザベースの SSH クライアントが WordPress インスタンスに接続されたら、以下のコマンドを入力します。このコマンドは、インスタンス上の bitnami_wordpress データベースのデータを転送し、Aurora データベースに移動します。このコマンドでは、*DatabaseUserName* は、Aurora データベースを作成したときに入力したプライマリユーザーの名前で置き換えます。*DatabaseEndpoint* は、Aurora データベースのエンドポイントアドレスに置き換えます。

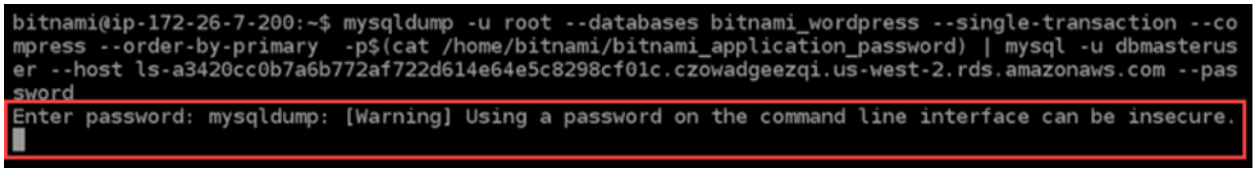
```
sudo mysqldump -u root --databases bitnami_wordpress --single-transaction --
compress --order-by-primary -p$(cat /home/bitnami/bitnami_application_password) |
 sudo mysql -u DatabaseUserName --host  DatabaseEndpoint --password
```

例

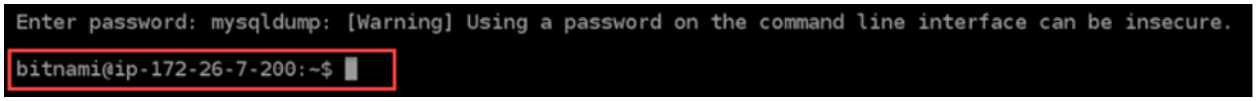
```
sudo mysqldump -u root --databases bitnami_wordpress --single-transaction --
compress --order-by-primary -p$(cat /home/bitnami/bitnami_application_password)
 | sudo mysql -u DBuser --host  abc123exampleE67890.czowadgeezqi.us-
west-2.rds.amazonaws.com --password
```

4. Enter password プロンプトで、Aurora データベースのパスワードを入力し、Enter キーを押します。

   入力中にパスワードを表示することはできません。



   データが正常に転送されると、次の例のような応答が表示されます。



エラーが表示される場合は、正しいデータベース、ユーザー名、パスワード、およびエンドポイントが使用されていることを確認して、もう一度試してください。

ステップ 5: Aurora データベースに WordPress を接続するように設定する

アプリケーションデータを Aurora データベースに転送した後、WordPress を設定して接続する必要があります。ウェブサイトが Aurora データベースに接続されるように、以下の手順を実行して WordPress 設定ファイル (wp-config.php) を編集します。

1. WordPress インスタンスに接続されているブラウザベースの SSH クライアントで、以下のコマンドを入力し、wp-config.php ファイルのバックアップを作成します。

```
cp /opt/bitnami/wordpress/wp-config.php /opt/bitnami/wordpress/wp-config.php-backup
```

2. 次のコマンドを入力して wp-config.php ファイルを書き込み可能にします。

```
sudo chmod 664 /opt/bitnami/wordpress/wp-config.php
```

3. config ファイル内のデータベースユーザー名を Aurora データベースを作成したときに入力したプライマリユーザーの名前に編集します。

```
sudo wp config set DB_USER DatabaseUserName
```

4. config ファイル内のデータベースホストを Aurora データベースのエンドポイントアドレスとポート番号で編集します。例えば、abc123exampleE67890.czowadgeezqi.us-west-2.rds.amazonaws.com:3306 です。

```
sudo wp config set DB_HOST DatabaseEndpoint:Port
```

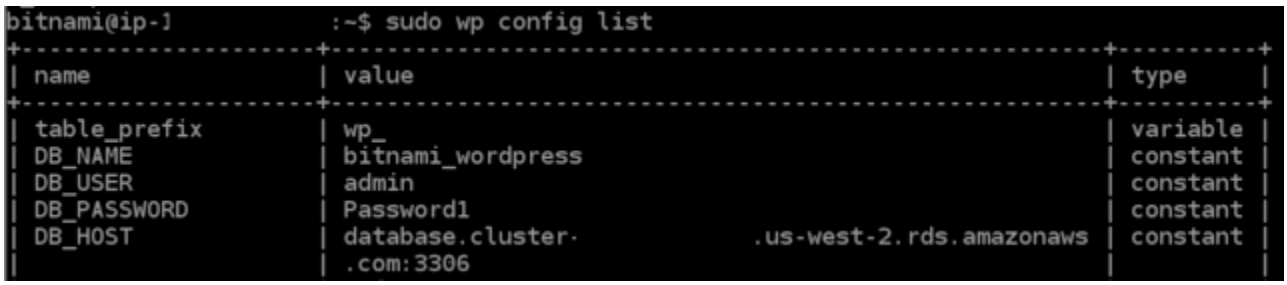
5. config ファイル内のデータベースパスワードを Aurora データベースのパスワードで編集します。

```
sudo wp config set DB_PASSWORD DatabasePassword
```

6. wp config list コマンドを入力して、wp-config.php ファイルに入力した情報が正しいことを確認します。

```
sudo wp config list
```

次の例のような結果で設定の詳細が表示されます。



7. 以下のコマンドを入力して、インスタンス上のウェブサービスを再起動します。

```
sudo /opt/bitnami/ctlscript.sh restart
```

サービスが再起動されると、次の例のような結果が表示されます。

お疲れ様でした。これで、WordPress サイトが Aurora データベースを使用できるように設定されました。

> **Note**
>
> 元の wp-config.php ファイルを復元する必要がある場合は、以下のコマンドを入力し、前にこのチュートリアルで作成したバックアップを使用して復元します。
>
> ```
> cp /opt/bitnami/wordpress/wp-config.php-backup /opt/bitnami/wordpress/wp-config.php
> ```

チュートリアル: WordPress ウェブサイトを Lightsail の MySQL マネージドデータベースに接続する

投稿、ページ、およびユーザーの重要な WordPress ウェブサイトデータは、Amazon Lightsail のインスタンスで実行されている MySQL データベースに保存されています。WordPress インスタンスに障害が発生した場合、データが回復不可能になる場合があります。このシナリオを回避するには、MySQL マネージドデータベースにウェブサイトのデータを転送する必要があります。

このチュートリアルでは、WordPress ウェブサイトデータを Lightsail の MySQL マネージドデータベースに転送する方法について説明します。ウェブサイトがマネージドデータベースに接続され、インスタンスで実行されているデータベースへの接続を停止するために、インスタンス上の WordPress 設定 (`wp-config.php`) ファイルを編集する方法についても説明します。

目次

ステップ 1: 前提条件を満たす

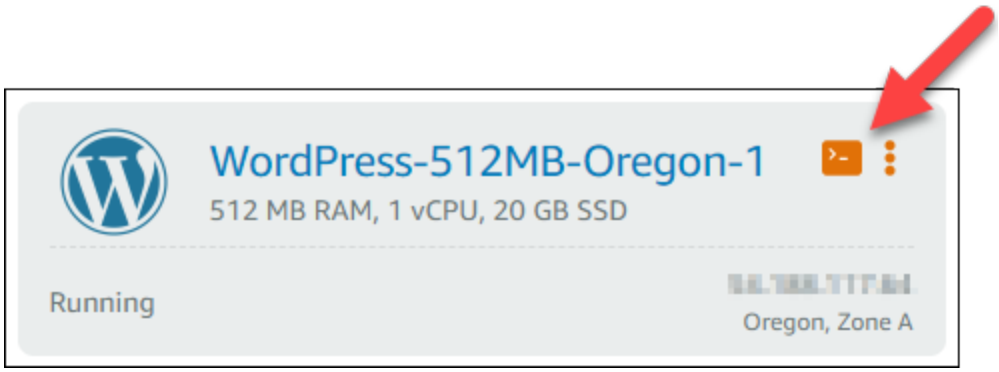
開始する前に、前提条件として以下の作業を実行します。

- Lightsail に WordPress インスタンスを作成し、実行状態になっていることを確認します。詳細については、「チュートリアル: Amazon Lightsail で WordPress インスタンスを起動して設定する」を参照してください。
- WordPress インスタンスと同じ AWS リージョン の Lightsail にMySQL マネージド型のデータベースを作成し、実行状態にあることを確認します。WordPress は、Lightsail で使用できるすべての MySQL データベースオプションで使用できます。詳細については、「Amazon Lightsail でデータベースを作成する」を参照してください。
- MySQL マネージドデータベースのパブリックおよびデータインポートモードを有効化します。このチュートリアルの手順を完了した後でこれらのモードを無効にできます。詳細については、「データベースのパブリックモードを設定する」および「データベースのデータインポートモードを設定する」を参照してください。

ステップ 2: MySQL マネージドデータベースに WordPress データベースを転送する

次の手順を実行して、WordPress ウェブサイトデータを、Lightsail の MySQL マネージドデータベースに転送します。

1. Lightsail コンソールにサインインします。
2. [Instances] (インスタンス) タブで、WordPress インスタンス用ブラウザベースの SSH クライアントアイコンを選択します。



3. ブラウザベースの SSH クライアントを WordPress インスタンスに接続後、以下のコマンドを入力して、bitnami_wordpress データベースのデータを MySQL マネージドデータベースに転送します。*DbUserName* をマネージドデータベースのユーザー名に置き換え、*DbEndpoint* をマネージドデータベースのエンドポイントアドレスに置き換えてください。

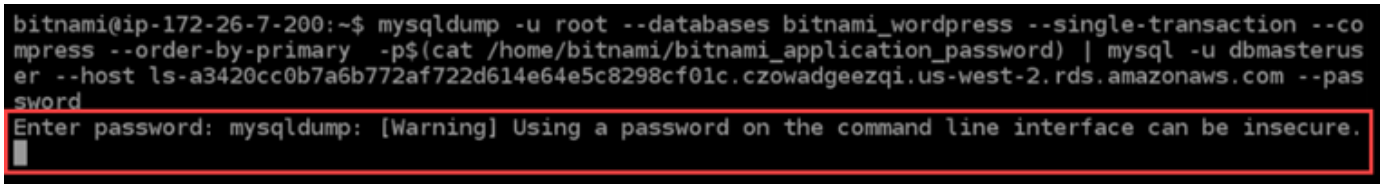
```
sudo mysqldump -u root --databases bitnami_wordpress --single-transaction --
compress --order-by-primary  -p$(cat /home/bitnami/bitnami_application_password) |
 sudo mysql -u DbUserName --host DbEndpoint --password
```

例

```
sudo mysqldump -u root --databases bitnami_wordpress --single-transaction --
compress --order-by-primary -p$(cat /home/bitnami/bitnami_application_password)
 | sudo mysql -u dbmasteruser --host ls-abc123exampleE67890.czowadgeezqi.us-
west-2.rds.amazonaws.com --password
```

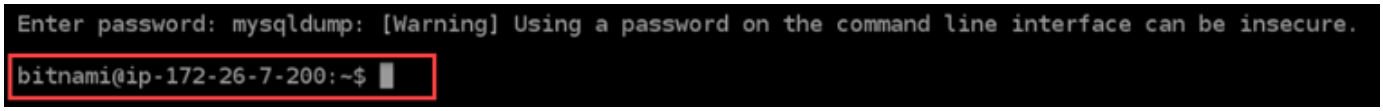
4. プロンプトで、MySQL マネージド型データベースのパスワードを入力し、Enter を押します。

   入力中にパスワードを表示することはできません。



5. データが正常に転送されると以下のような表示が出ます。

   エラーが表示される場合は、正しいデータベース、ユーザー名、パスワード、またはエンドポイントが使用されていることを確認して、もう一度試してください。



ステップ 3: MySQL マネージド型データベースに WordPress を接続するように設定する

ウェブサイトが MySQL マネージドデータベースに接続されるように、以下の手順を実行して WordPress 設定ファイル (`wp-config.php`) を編集します。

1. 問題が発生した場合は、WordPress インスタンスに接続されているブラウザベースの SSH クライアントで、以下のコマンドを入力し、`wp-config.php` ファイルのバックアップを作成します。

   ```
   cp /opt/bitnami/wordpress/wp-config.php /opt/bitnami/wordpress/wp-config.php-backup
   ```

2. 以下のコマンドを入力して、Nano テキストエディタ を使用し、`wp-config.php` ファイルを開きます。

   ```
   nano /opt/bitnami/wordpress/wp-config.php
   ```

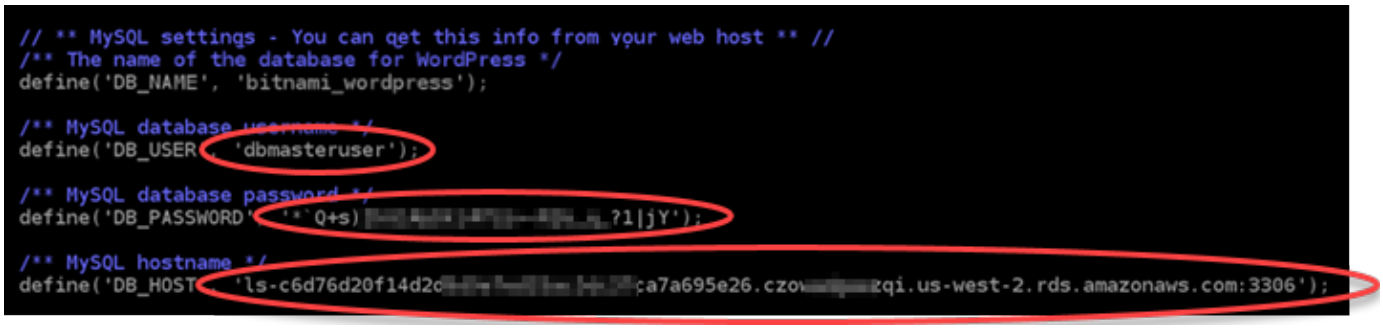
3. 以下の例のように`DB_USER`、`DB_PASSWORD`、および`DB_HOST` 値が見つかるまで下にスクロールします。

```
// ** MySQL settings - You can get this info from your web host ** //
/** The name of the database for WordPress */
define('DB_NAME', 'bitnami_wordpress');

/** MySQL database username */
define('DB_USER', 'bn_wordpress');

/** MySQL database password */
define('DB_PASSWORD', 'd6ab501583');

/** MySQL hostname */
define('DB_HOST', 'localhost:3306');
```

4. 次の値を変更します。

   - DB_USER — これを編集して MySQL マネージドデータベースのユーザー名に一致させます。Lightsail マネージドデータベースのデフォルトプライマリーユーザー名はdbmasteruser です。
   - DB_PASSWORD — これを編集して MySQL マネージドデータベースのパスワードに一致させます。詳細については、「データベースのパスワードを管理する」を参照してください。
   - DB_HOST — これを編集して MySQL マネージドデータベースのエンドポイントに一致させます。ホストアドレスの末尾に必ず :3306 ポート番号を入力します。例えば、「ls-abc123exampleE67890.czowadgeezqi.us-west-2.rds.amazonaws.com:3306」と入力します。

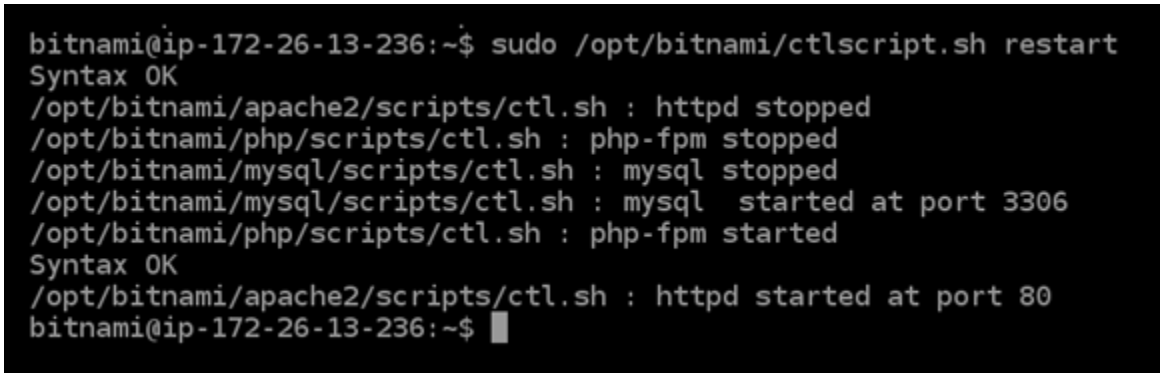
   結果は次の例のようになります。



5. Ctrl+X を押して Nano を終了し、Y および Enter を押して編集内容を保存します。
6. 以下のコマンドを入力して、インスタンス上のウェブサービスを再起動します。

```
sudo /opt/bitnami/ctlscript.sh restart
```

   サービスが再起動されると以下のような結果が表示されます。



お疲れ様でした。これで、WordPress サイトは MySQL マネージド型データベースを使用できるように設定されました。

> **Note**
>
> 何らかの理由で、元の wp-config.php ファイルを復元する必要がある場合は、以下のコマンドを入力し、前にこのチュートリアルで作成したバックアップを使用して復元します。
>
> ```
> cp /opt/bitnami/wordpress/wp-config.php-backup /opt/bitnami/wordpress/wp-config.php
> ```

ステップ 4: 次のステップを完了する

WordPress ウェブサイトを MySQL マネージドデータベースに接続したら、以下の追加ステップを実行します。

- WordPress インスタンスのスナップショットを作成します。詳細については、「Linux または Unix インスタンスのスナップショットを作成する」を参照してください。
- MySQL マネージドデータベースのスナップショットを作成します。詳細については、「データベースのスナップショットを作成する」を参照してください。
- MySQL マネージドデータベースのパブリックおよびデータインポートモードを無効化します。詳細については、「データベースのパブリックモードを設定する」および「データベースのデータインポートモードを設定する」を参照してください。

チュートリアル: Lightsail バケットに WordPress インスタンスを接続する

このチュートリアルでは、Amazon Lightsail インスタンスで実行されている WordPress ウェブサイトを Lightsail バケットに接続するために必要な手順について説明します。バケットを使用して、画像や添付ファイルなどの静的コンテンツをホストすることが可能です。これを行うには、WP Offload Media Lite プラグインを WordPress ウェブサイトにインストールし、Lightsail バケットに接続するように設定する必要があります。プラグインを設定すると、 WordPress ウェブサイトにアップロードしたすべてのメディアが、インスタンスのディスクではなく、バケットに自動的に追加されます。

目次

ステップ 1: 前提条件を満たす

以下の前提条件を完了します (まだの場合)。

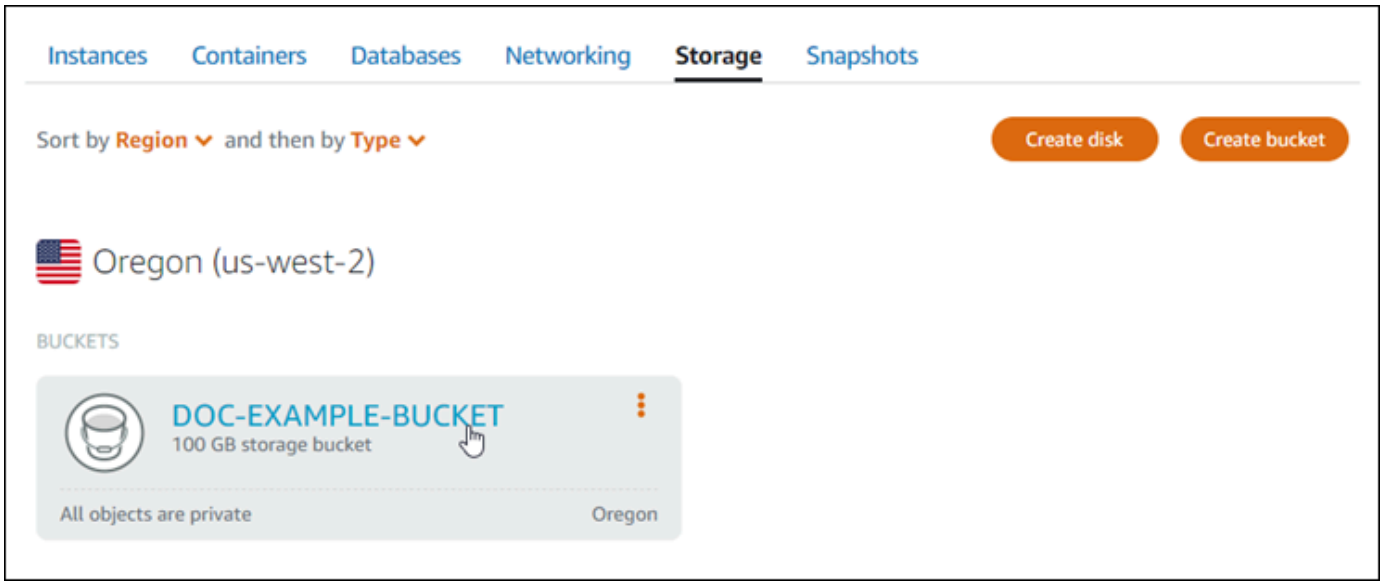
- Lightsail で WordPress インスタンスを作成します。詳細については、「チュートリアル: Amazon Lightsail で WordPress インスタンスを起動して設定する」を参照してください。
- Lightsail オブジェクトストレージサービスでバケットを作成します。詳細については、「バケットの作成」を参照してください。

ステップ 2: バケットのアクセス許可を変更する

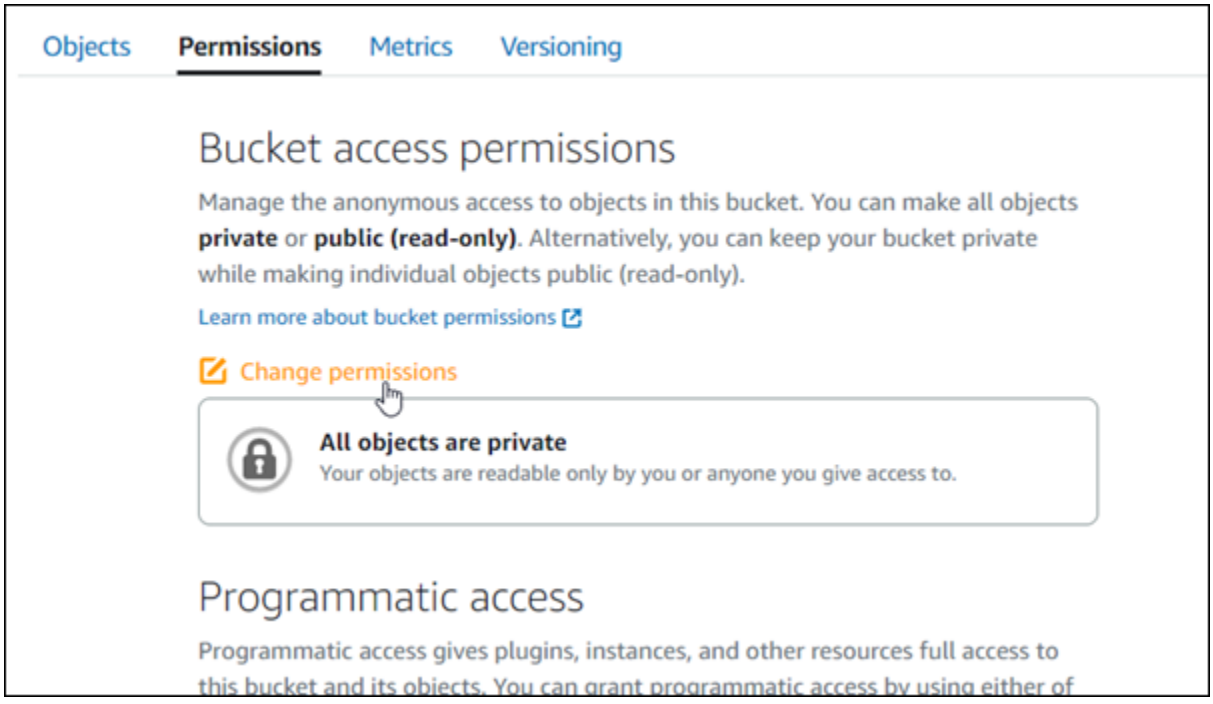
次の手順を実行して、バケットのアクセス許可を変更し、 WordPress インスタンスと Offload Media Lite プラグインへのアクセスを許可します。バケットのアクセス許可は個々のオブジェクトを公開 (読み取り専用)に設定する必要があります。また、 WordPress インスタンスをバケットのアクセスロールにアタッチする必要があります。バケット許可の詳細については、「バケットのアクセス許可」を参照してください。

1. Lightsail コンソールにサインインします。
2. Lightsail ホームページで、ストレージタブを選択します。

3. WordPress ウェブサイトで使用するバケットの名前を選択します。

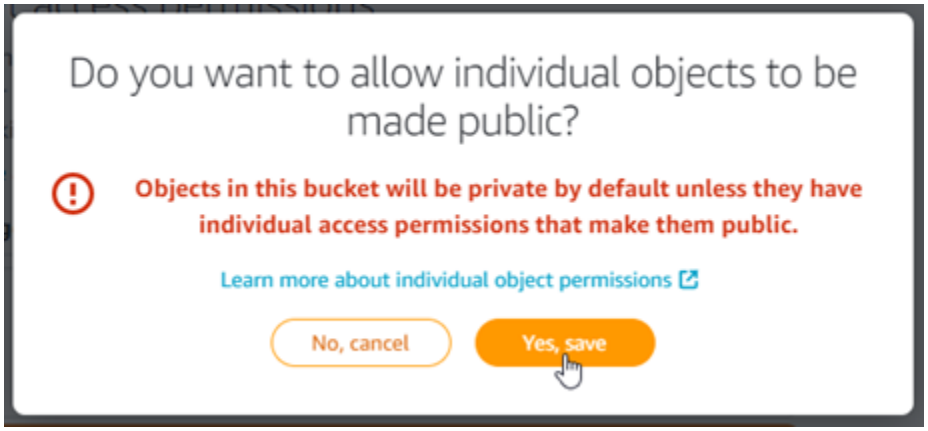


4. バケット管理ページで [Permissions] (許可) タブを選択します。
5. ページの「バケットのアクセス許可」セクションで [Change permissions](許可の変更) を選択します。



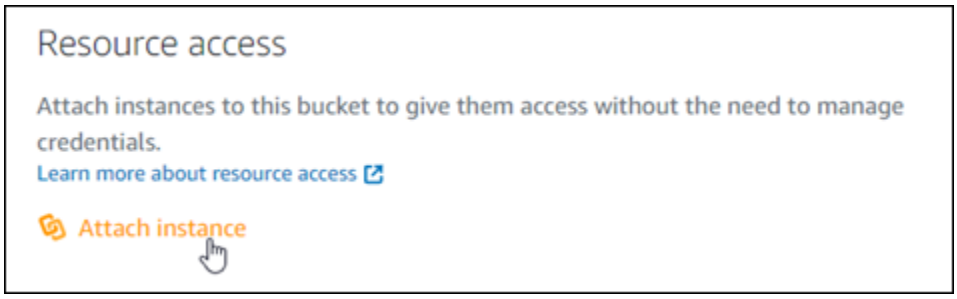
6. 個々のオブジェクトを選択して公開し、読み取り専用にすることができます。

7. [保存] を選択します。
8. 表示される確認プロンプトで、[はい、選択]を選択します。

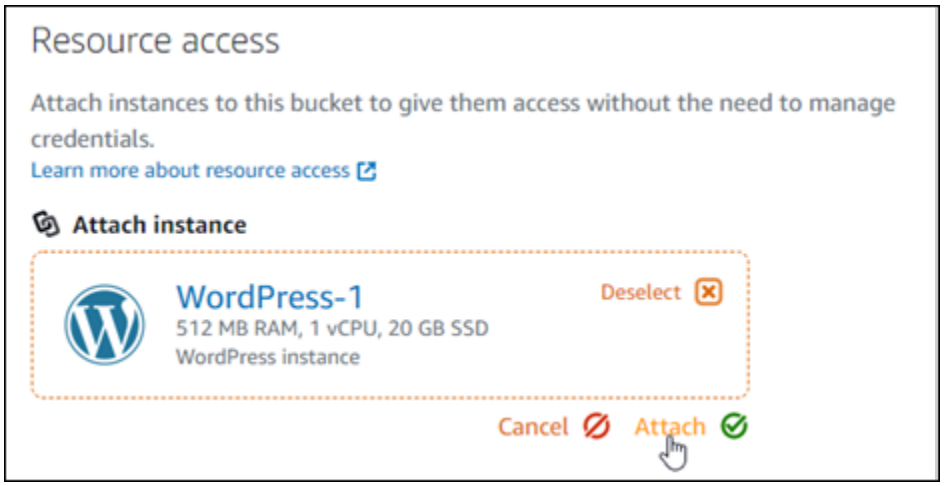


しばらくすると、バケットは個々のオブジェクトにアクセスを許可するように設定されます。これにより、Offload Media Lite プラグインを使用して WordPress ウェブサイトからバケットにアップロードされたオブジェクトを顧客が読み取れるようになります。

9. ページの [リソースアクセス] セクションまでスクロールし、[Attach instance] (インスタンスの添付) を選択します。



10. 表示されるドロップダウンリストで WordPress インスタンスの名前を選択し、アタッチを選択します。



しばらくすると、 WordPress インスタンスがバケットにアタッチされます。これにより、バケットとそのオブジェクトを管理するためのアクセス権が WordPress インスタンスに付与されます。

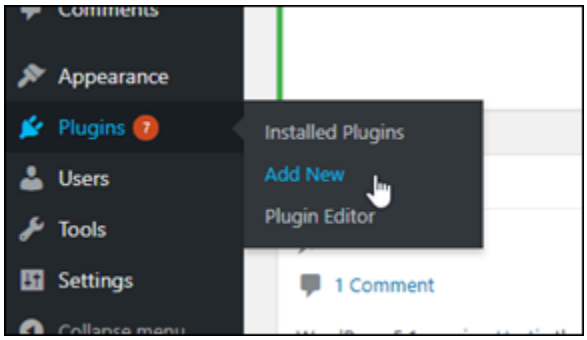
ステップ 3: WordPress ウェブサイトに WP Offload Media Lite プラグインをインストールする

WordPress ウェブサイトに WP Offload Media Lite プラグインをインストールするには、以下の手順を実行します。このプラグインは、 WordPress メディアアップローダーを介して追加されたイメージ、動画、ドキュメント、およびその他のメディアを Lightsail バケットに自動的にコピーします。詳細については、 WordPress ウェブサイトの「WP Offload Media Lite」を参照してください。

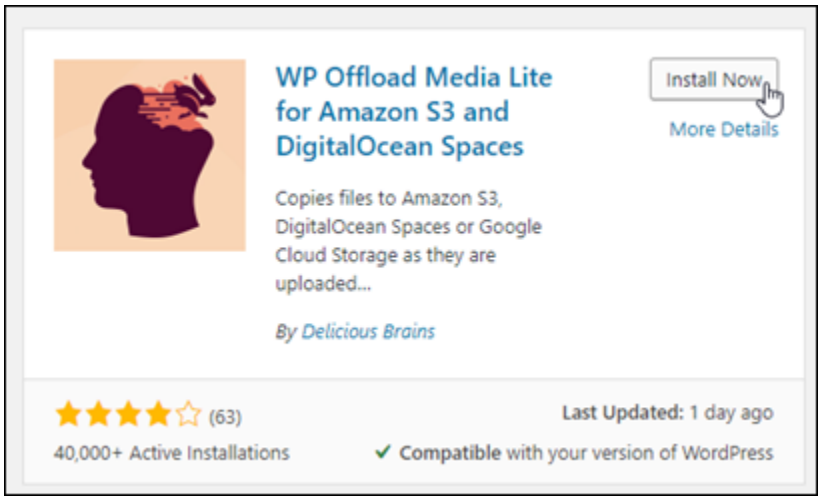
1. 管理者として WordPress ウェブサイトのダッシュボードにサインインします。

詳細については、「Amazon Lightsail での Bitnami インスタンスのアプリケーションユーザー名とパスワードの取得」を参照してください。

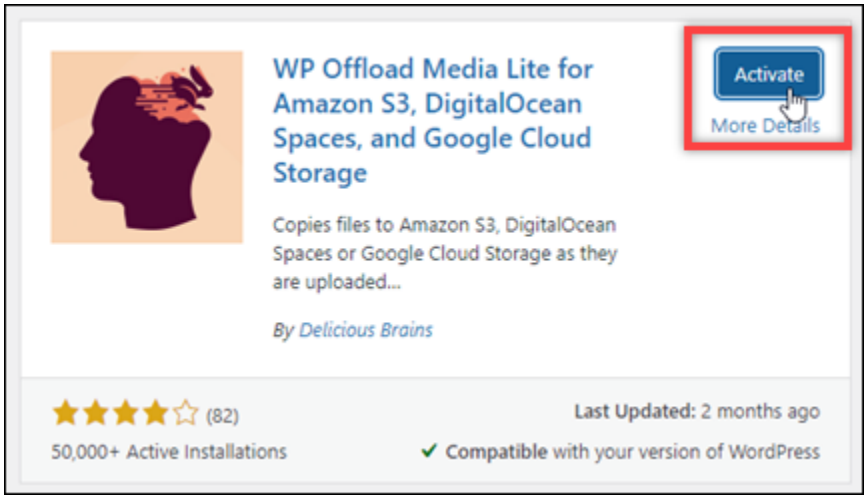
2. 左側のナビゲーションメニューの [プラグイン] を一時停止し、[Add New] (新規追加) を選択します。



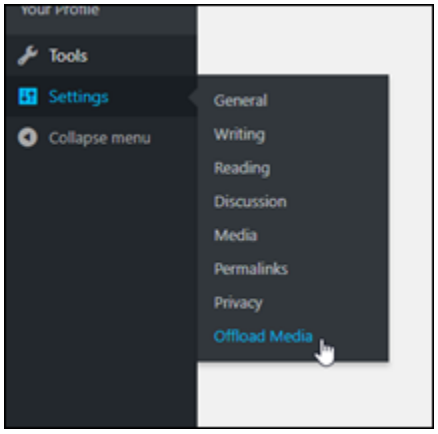
3. [WP Offload Media Lite] を検索します。
4. 検索結果の中から WP Offload Media プラグインの横の [Install Now] (今すぐインストール)を選択します。



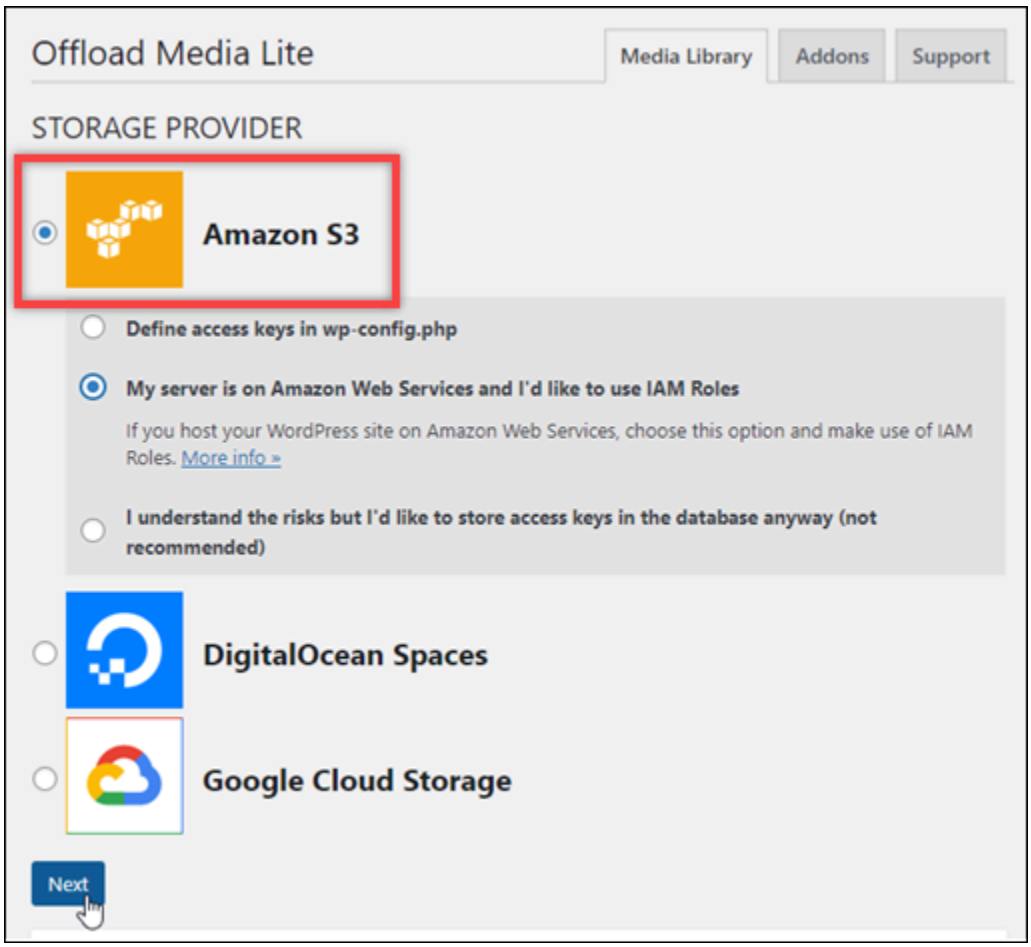
5. プラグインのインストールが完了したら、[アクティベート] を選択します。



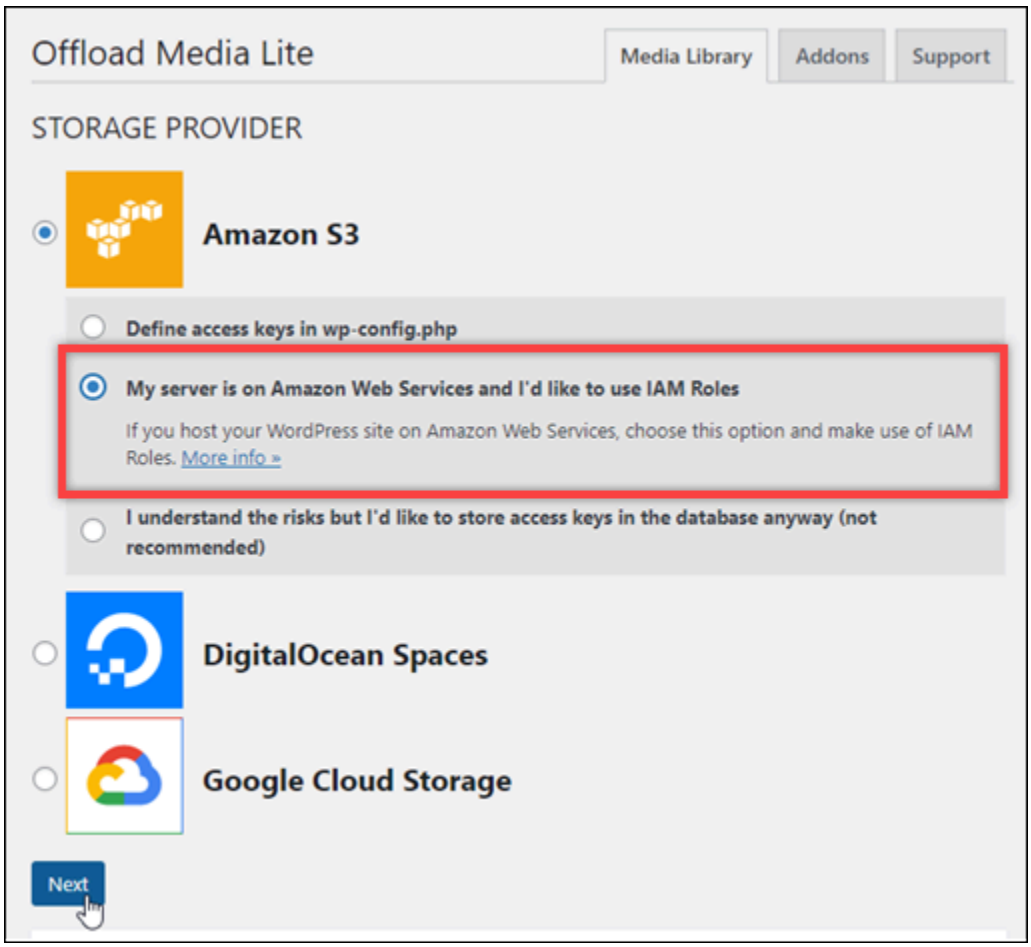
6. 左のナビゲーションメニューで、[Settings] (設定) を選択し、[Offload Media] を選択します。



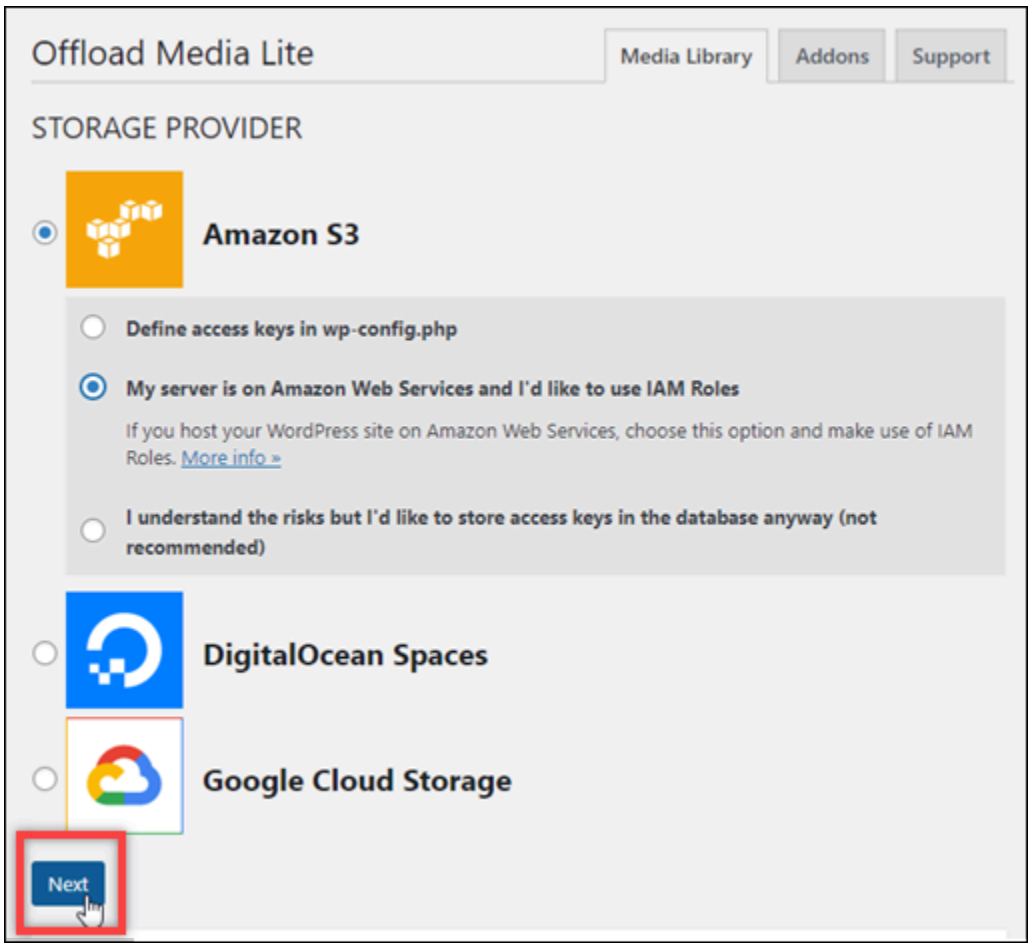
7. Offload Media ページで [Amazon S3] をストレージプロバイダとして選択します。



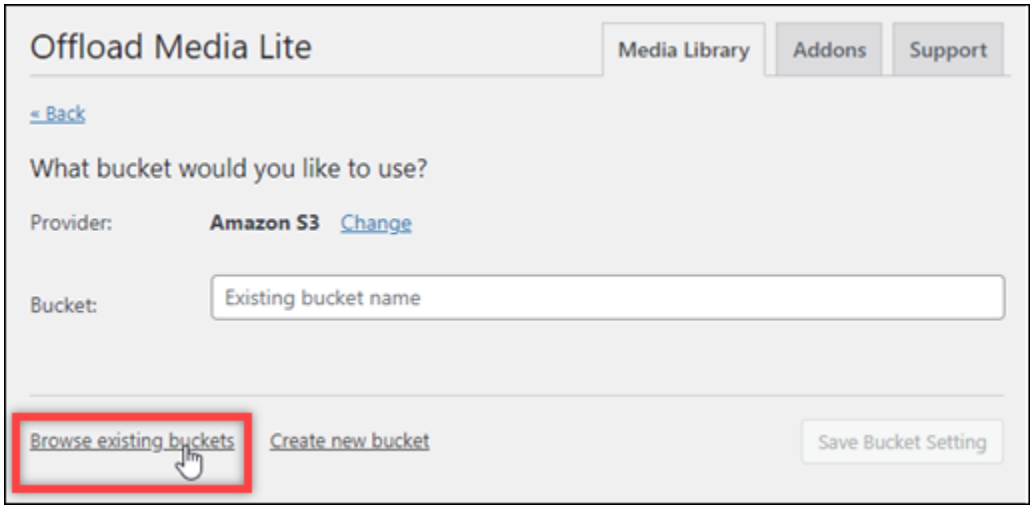
8. [私のサーバーは Amazon Web Services 上にあり、IAM ロールを使いたい] を選択します。


Offload Media Lite
Media Library
Addons
Support
STORAGE PROVIDER
Amazon S3
Define access keys in wp-config.php
My server is on Amazon Web Services and I'd like to use IAM Roles
If you host your WordPress site on Amazon Web Services, choose this option and make use of IAM Roles. More info »
I understand the risks but I'd like to store access keys in the database anyway (not recommended)
DigitalOcean Spaces
Google Cloud Storage
Next

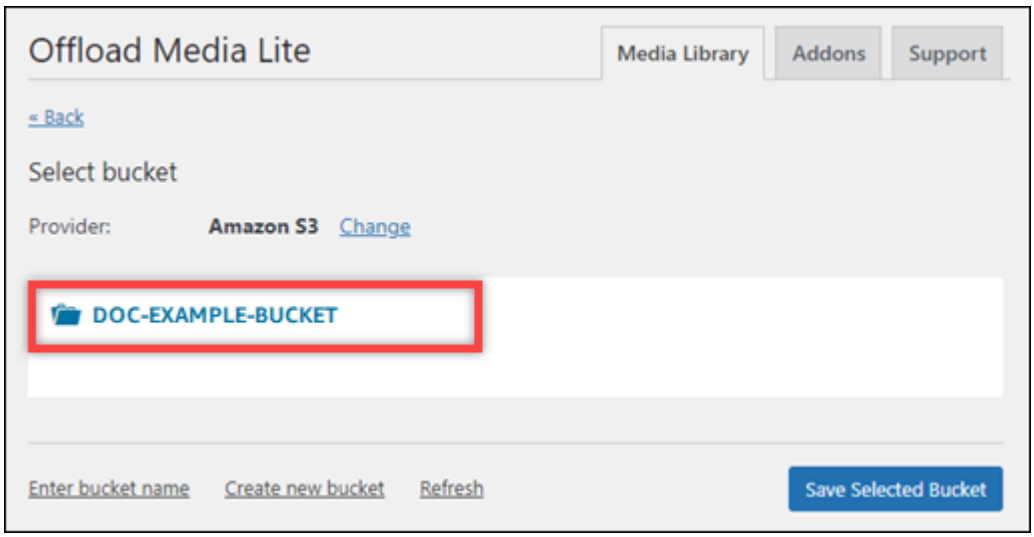
9. [次へ] を選択します。



10. 「どのバケットを使用しますか?」のページで [Browse existing buckets] (既存のバケットを参照する) を選択します。

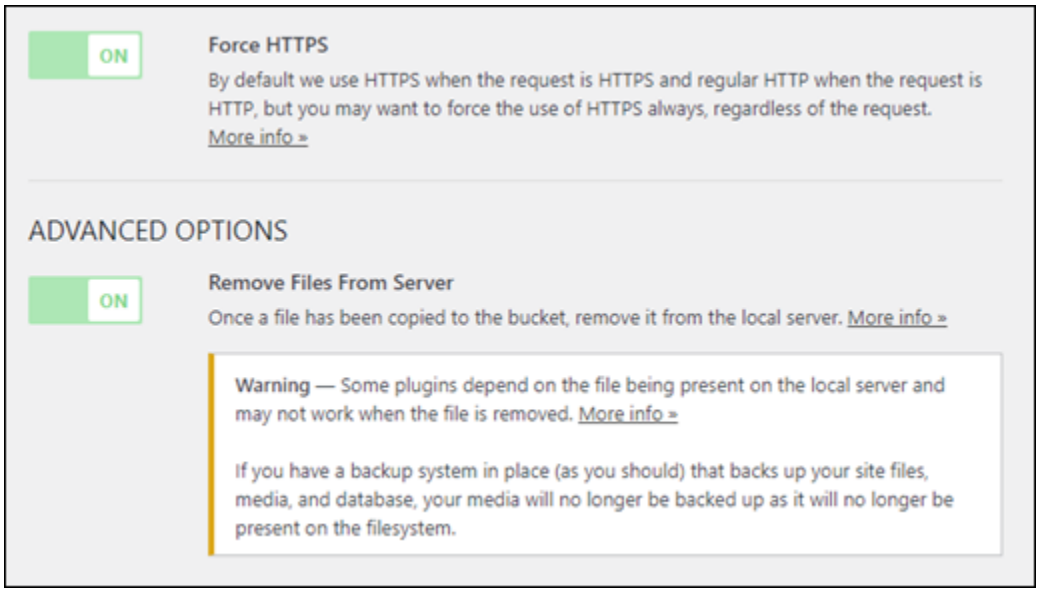


11. インスタンスで使用する WordPressバケットの名前を選択します。



12. 表示されるOffload Media Lite Settings画面で、HTTPS の強制実行およびサーバーからファイルを削除をオンにします。

    - Lightsail バケットはデフォルトで HTTPS を使用してメディアファイルを提供するため、強制 HTTPS 設定をオンにする必要があります。この機能をオンにしないと、 WordPress ウェブサイトから Lightsail バケットにアップロードされたメディアファイルはウェブサイトの訪問者に正しく提供されません。
    - サーバーからファイルを削除する設定では、Lightsail バケットにアップロードされたメディアがインスタンスのディスクにも保存されないようにします。この機能をオンにしない場合、Lightsail バケットにアップロードされたメディアファイルも WordPress インスタンスのローカルストレージに保存されます。



13. [変更の保存] をクリックします。

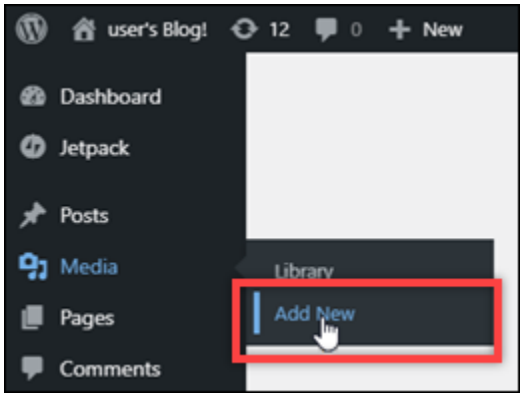
> **Note**
>
> 後でOffload Media Lite Settings画面に戻るには、左のナビゲーションメニューで [設定] を一時停止し、[Offload Media Lite] を選択します。

これで、Media Lite プラグインを使用するように WordPress ウェブサイトが設定されました。次回 を介してメディアファイルをアップロードすると WordPress、そのファイルは自動的に Lightsail バケットにアップロードされ、バケットによって提供されます。設定をテストするには、このチュートリアルの次のセクションに進みます。

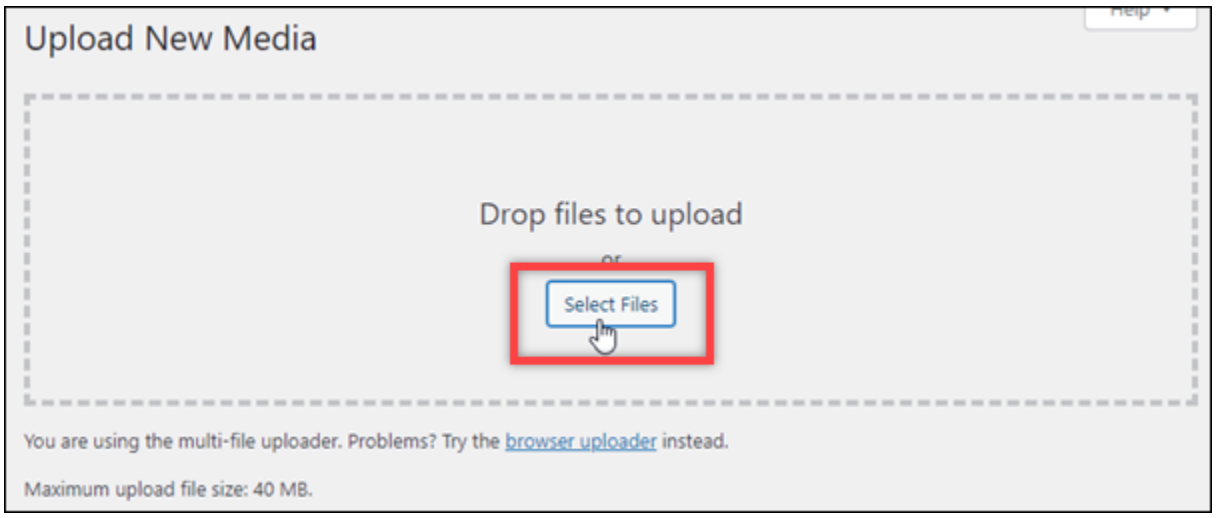
ステップ 4: WordPress ウェブサイトと Lightsail バケット間の接続をテストする

次の手順を実行して、メディアファイルを WordPress インスタンスにアップロードし、Lightsail バケットにアップロードされ、提供されていることを確認します。

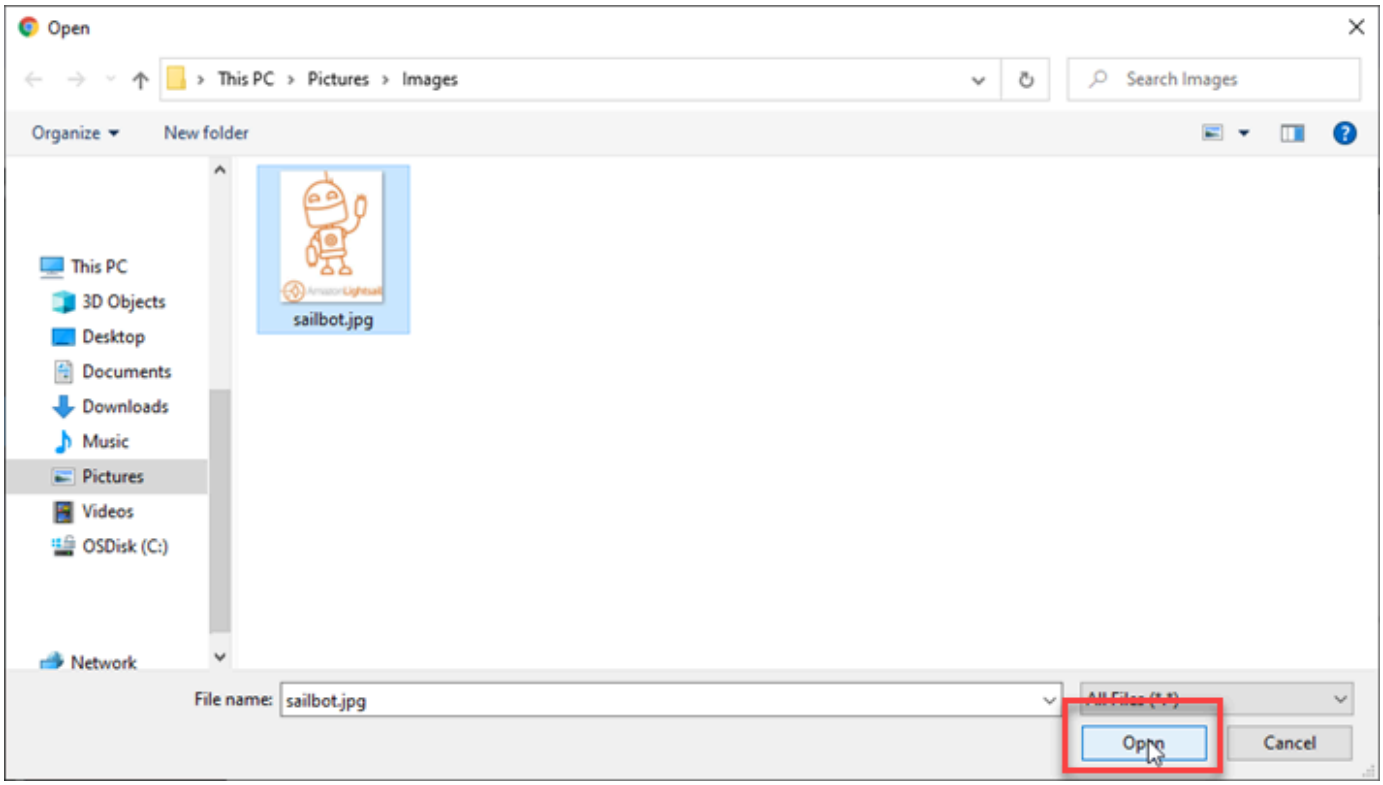
1. ダッシュボードの WordPress左側のナビゲーションメニューにあるメディアで一時停止し、新規追加を選択します。



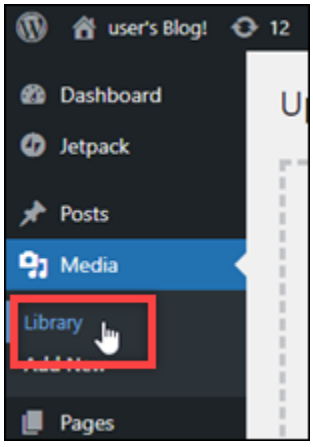
2. 表示される、[新しいメディアをアップロード] 画面で [ファイルを選択] を選択します。



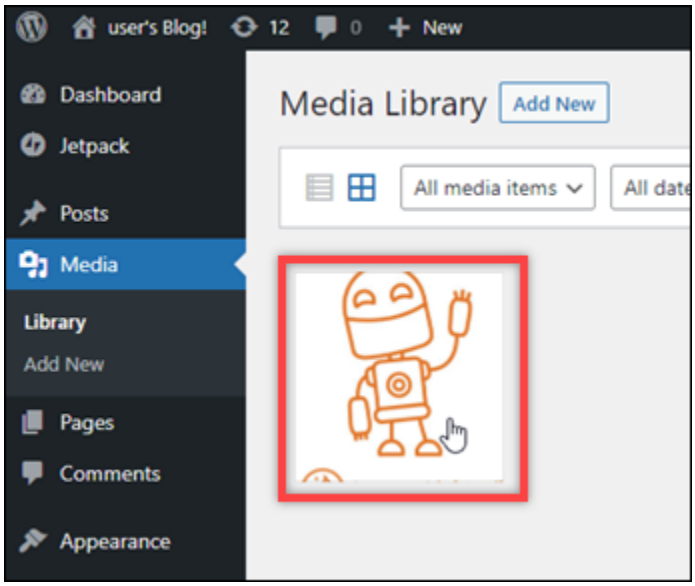
3. ローカルコンピュータからアップロードするメディアファイルを選択し、[開く] を選択します。



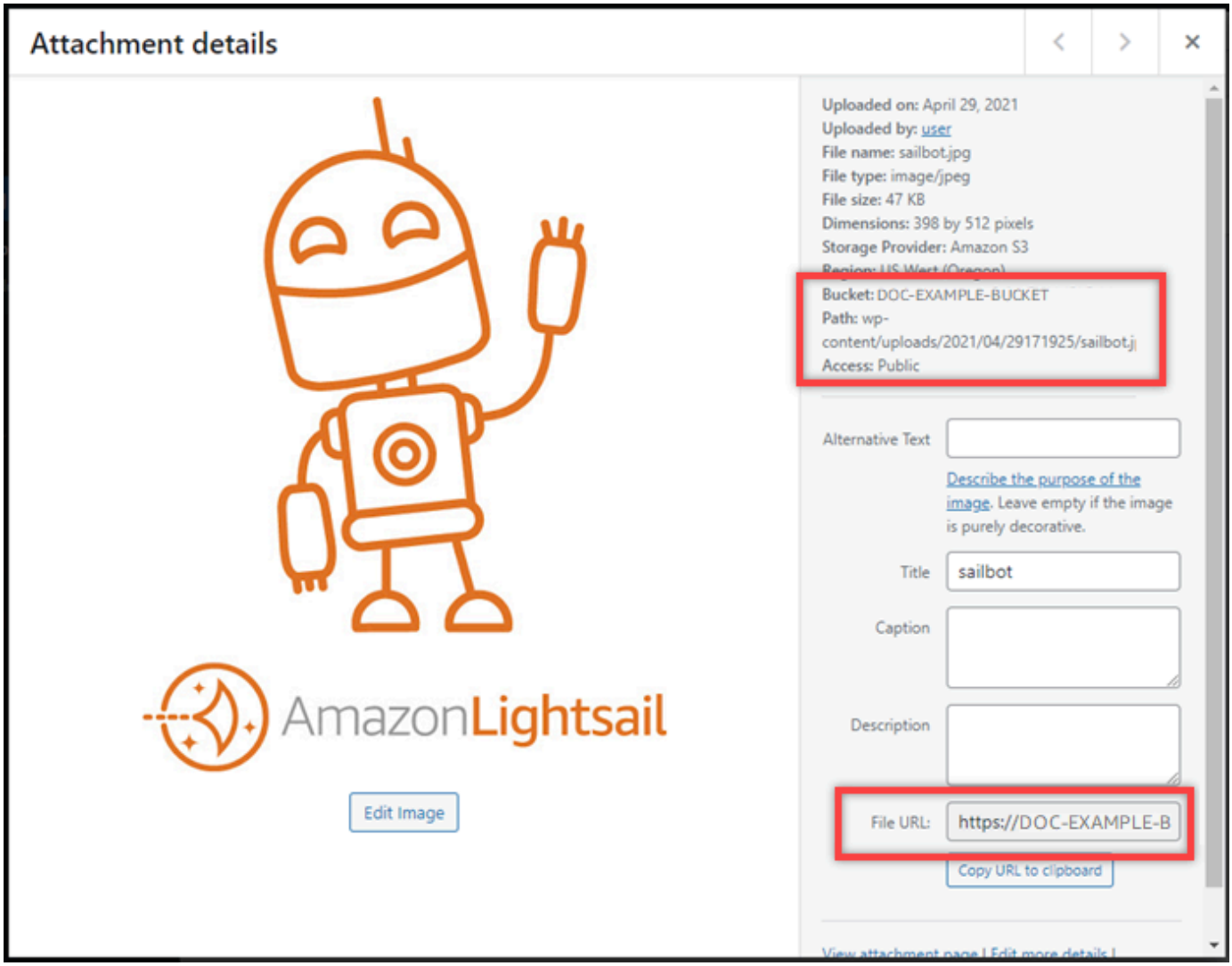
4. ファイルのアップロードが完了したら、左のナビゲーションメニューにある [メディア] の [ライブラリ] を選択します。



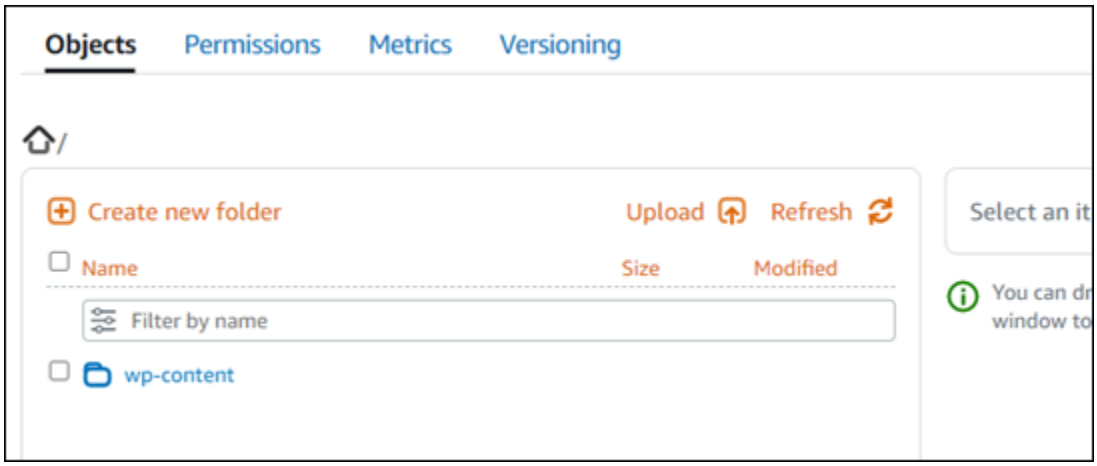
5. 最近アップロードしたファイルを選択します。



6. ファイルの詳細パネルで、バケットおよびファイルの URL フィールドにバケット名が表示されます。



7. Lightsail バケット管理ページのオブジェクトタブに移動すると、wp-content フォルダが表示されます。このフォルダは、Offload Media Lite プラグインによって作成され、アップロードしたメディアファイルを保存するために使用されます。



バケットとオブジェクトを管理する

Lightsail オブジェクトストレージバケットを管理する一般的な手順は次のとおりです。

1. Amazon Lightsail オブジェクトストレージサービスのオブジェクトとバケットについて説明します。詳細については、Amazon Lightsail のオブジェクトストレージ を参照してください。
2. Amazon Lightsail でバケットに付けることができる名前について説明します。詳細については、「Amazon Lightsail のバケット命名規則」を参照してください。
3. バケットを作成して Lightsail オブジェクトストレージサービスの使用を開始します。詳細については、「Amazon Lightsail でのバケットの作成」を参照してください。
4. バケットのセキュリティのベストプラクティスと、バケットに設定できるアクセス許可について説明します。バケット内のすべてのオブジェクトをパブリックまたはプライベートにすることも、オブジェクトを個別に選択してパブリックにすることもできます。また、アクセスキーを作成し、インスタンスをバケットに追加し、他の AWS アカウントにアクセス権を付与することで、バケットへのアクセスを許可することもできます。詳細については、「Amazon Lightsail オブジェクトストレージのセキュリティのベストプラクティス」および「Amazon Lightsail でのバケットのアクセス許可について」を参照してください。

   バケットのアクセス許可について理解したら、以下のガイドを参照してバケットへのアクセスを許可してください。

   - Amazon Lightsail のバケットへのパブリックアクセスをブロックする
   - Amazon Lightsail でのバケットアクセス許可の設定
   - Amazon Lightsail のバケット内の個々のオブジェクトに対するアクセス許可の設定
   - Amazon Lightsail でのバケットのアクセスキーの作成
   - Amazon Lightsail でのバケットのリソースアクセスの設定
   - Amazon Lightsail でのバケットのクロスアカウントアクセスの設定
5. バケットのアクセスログの記録を有効にする方法と、アクセスログを使用してバケットのセキュリティを監査する方法について説明します。詳細については、以下のガイドを参照してください。

   - Amazon Lightsail オブジェクトストレージサービスでのバケットへのアクセスの口グ記録
   - Amazon Lightsail オブジェクトストレージサービスのバケットのアクセスログ形式
   - Amazon Lightsail オブジェクトストレージサービスでのバケットのアクセスログ記録の有効化
   - Amazon Lightsail でバケットのアクセスログを使用してリクエストを識別する

6. Lightsail でバケットを管理する権限をユーザーに付与する IAM ポリシーを作成します。詳細については、「Amazon Lightsail でバケットを管理する IAM ポリシー」を参照してください。
7. バケット内のオブジェクトにラベルを付けて識別する方法について説明します。詳細については、「Amazon Lightsail でのオブジェクトキー名の理解」を参照してください。
8. ファイルをアップロードしてバケット内のオブジェクトを管理する方法について説明します。詳細については、以下のガイドを参照してください。
   - Amazon Lightsail のバケットにファイルをアップロードする
   - マルチパートアップロードを使用した Amazon Lightsail のバケットへのファイルのアップロード
   - Amazon Lightsail でバケット内のオブジェクトを表示する
   - Amazon Lightsail のバケット内のオブジェクトのコピーまたは移動
   - Amazon Lightsail のバケットからオブジェクトをダウンロードする
   - Amazon Lightsail のバケット内のオブジェクトのフィルタリング
   - Amazon Lightsail のバケット内のオブジェクトのタグ付け
   - Amazon Lightsail でバケット内のオブジェクトを削除する
9. オブジェクトのバージョニングを有効にすると、バケットに保存されたあらゆるオブジェクトのあらゆるバージョンを保存、取得、復元します。詳細については、「Amazon Lightsail のバケットでのオブジェクトのバージョニングの有効化と一時停止」を参照してください。
10. オブジェクトのバージョニングを有効にすると、バケット内のオブジェクトの以前のバージョンを復元できます。詳細については、「Amazon Lightsail のバケット内のオブジェクトの以前のバージョンの復元」を参照してください。
11. バケットの使用率を監視します。詳細については、「Amazon Lightsail でのバケットのメトリクスの表示」を参照してください。
12. バケットの使用率がしきい値を超えたときにバケットメトリクスが通知されるよう、アラームを設定します。詳細については、「Amazon Lightsail でのバケットメトリクスアラームの作成」を参照してください。
13. ストレージとネットワーク転送量が不足している場合は、バケットのストレージプランを変更します。詳細については、「Amazon Lightsail でのバケットのプランの変更」を参照してください。
14. バケットを他のリソースに接続する方法について説明します。詳細については、以下のチュートリアルを参照してください。
   - チュートリアル: WordPress インスタンスを Amazon Lightsail バケットに接続する

- チュートリアル: Lightsail コンテンツ配信ネットワークディストリビューションで Amazon Lightsail バケットを使用する

15.使用しなくなったバケットを削除します。詳細については、「Amazon Lightsail でのバケットの削除」を参照してください。

Lightsail でコンテンツ配信ネットワークディストリビューションを使用するようにインスタンスを設定する WordPress

このガイドでは、Amazon Lightsail ディストリビューションで動作するように WordPress インスタンスを設定する方法について説明します。

すべての Lightsail ディストリビューションでは、デフォルトドメイン (例:) で HTTPS がデフォルトで有効になっています`123456abcdef.cloudfront.net`。ディストリビューションの設定によって、ディストリビューションとインスタンス間の接続が暗号化されるかどうかが決まります。

- WordPress ウェブサイトが HTTP のみを使用している – ウェブサイトがディストリビューションのオリジンとして HTTP のみを使用し、HTTPS を使用するように設定されていない場合は、SSL/TLS を終了し、暗号化されていない接続を使用してすべてのコンテンツリクエストをインスタンスに転送するようにディストリビューションを設定できます。
- WordPress ウェブサイトが HTTPS を使用している – ウェブサイトがディストリビューションのオリジンとして HTTPS を使用している場合は、暗号化された接続を使用してすべてのコンテンツリクエストをインスタンスに転送するようにディストリビューションを設定できます。この設定は end-to-end 暗号化と呼ばれます。

ディストリビューションを作成する

WordPress インスタンスの Lightsail ディストリビューションを設定するには、次のステップを実行します。詳細については、「the section called “ディストリビューションを作成する”」を参照してください。

前提条件

「」の説明に従って WordPress インスタンスを作成して設定しますthe section called “WordPress”。

WordPress インスタンスのディストリビューションを作成するには

1. Lightsail ホームページで、ネットワーク を選択します。

2. [ディストリビューションの作成] を選択します。
3. オリジンの選択 で、 WordPress インスタンスを実行しているリージョンを選択し、 WordPress インスタンスを選択します。インスタンスにアタッチした静的 IP アドレスが自動的に使用されます。
4. キャッシュ動作 で、 の最適 WordPressを選択します。
5. (オプション) end-to-end 暗号化を設定するには、オリジンプロトコルポリシーを HTTPS のみに変更します。詳細については、「the section called “オリジンプロトコルポリシー”」を参照してください。
6. 残りのオプションを設定し、ディストリビューションの作成を選択します。
7. カスタムドメイン タブで、証明書の作成 を選択します。証明書の一意の名前を入力し、ドメインとサブドメインの名前を入力して、証明書の作成を選択します。
8. [証明書のアタッチ] を選択します。
9. 「DNS レコードを更新する」で、「 を理解します」を選択します。

DNS レコードを更新する

Lightsail DNS ゾーンの DNS レコードを更新するには、次のステップを実行します。

ディストリビューションの DNS レコードを更新するには

1. Lightsail ホームページで、ドメインと DNS を選択します。
2. DNS ゾーンを選択し、DNS レコードタブを選択します。
3. 証明書で指定したドメインの A レコードと AAAA レコードを削除します。
4. レコードを追加を選択し、ドメインをディストリビューションのドメインに解決する CNAME レコードを作成します (例: d2vbec9EXAMPLE.cloudfront.net).。
5. [保存] を選択します。

ディストリビューションによる静的コンテンツのキャッシュを許可する

ディストリビューションで動作するように WordPress インスタンス内の `wp-config.php` ファイルを編集するには、以下の手順を実行します。

> **Note**
>
> この手順を開始する前に、 WordPress インスタンスのスナップショットを作成することをお勧めします。スナップショットは、何か問題が発生した場合、これを元に別のインスタンスを作成できるバックアップとして使用できます。詳細については、「Linux または Unix インスタンスのスナップショットを作成する」を参照してください。

1. Lightsail コンソールにサインインします。
2. Lightsail ホームページで、 WordPress インスタンスの横に表示されるブラウザベースの SSH クライアントアイコンを選択します。
3. インスタンスに接続したら、次のコマンドを入力して、wp-config.php ファイルのバックアップを作成します。何らかの問題が発生した場合は、バックアップを使用してファイルを復元することができます。

```
sudo cp /opt/bitnami/wordpress/wp-config.php /opt/bitnami/wordpress/wp-
config.php.backup
```

4. 次のコマンドを入力して、 Vim を使用し、wp-config.php ファイルを開きます。

```
sudo vim /opt/bitnami/wordpress/wp-config.php
```

5. I キーを押して Vim の挿入モードに移ります。
6. ファイルで、次のコード行を削除します。

```
define('WP_SITEURL', 'http://' . $_SERVER['HTTP_HOST'] . '/');
define('WP_HOME', 'http://' . $_SERVER['HTTP_HOST'] . '/');
```

7. WordPress 使用している のバージョンに応じて、次のいずれかのコード行を ファイルに追加します。

   - バージョン 3.3 以前を使用している場合、以前にコードを削除した箇所に次のコード行を追加します。

```
define('WP_SITEURL', 'https://' . $_SERVER['HTTP_HOST'] . '/');
define('WP_HOME', 'https://' . $_SERVER['HTTP_HOST'] . '/');
if (isset($_SERVER['HTTP_CLOUDFRONT_FORWARDED_PROTO'])
&& $_SERVER['HTTP_CLOUDFRONT_FORWARDED_PROTO'] === 'https') {
$_SERVER['HTTPS'] = 'on';
```

```
}
```

- バージョン 3.3.1-5 以降を使用している場合、ファイルの削除した箇所に、次のコード行を追加します。

```
define('WP_SITEURL', 'http://DOMAIN/');
define('WP_HOME', 'http://DOMAIN/');
if (isset($_SERVER['HTTP_CLOUDFRONT_FORWARDED_PROTO'])
&& $_SERVER['HTTP_CLOUDFRONT_FORWARDED_PROTO'] === 'https') {
$_SERVER['HTTPS'] = 'on';
}
```

8. ESC キーを押して Vim の挿入モードを終了し、`:wq!` を入力して Enter キーで編集内容を保存しで (書き込んで) Vim を終了します。
9. 次のコマンドを入力して、インスタンス上の Apache サービスを再起動します。

```
sudo /opt/bitnami/ctlscript.sh restart apache
```

10. Apache サービスが再起動するまでしばらく待ってから、ディストリビューションがコンテンツをキャッシュしているかどうかをテストします。詳細については、「Amazon Lightsail ディストリビューションをテストする」を参照してください。
11. 何らかの問題が発生した場合は、ブラウザベースの SSH クライアントを使用してインスタンスに再接続します。次のコマンドを実行して、このガイドで先に作成したバックアップを使用して `wp-config.php` ファイルで復元します。

```
sudo cp /opt/bitnami/wordpress/wp-config.php.backup /opt/bitnami/wordpress/wp-
config.php
```

ファイルを復元したら、次のコマンドを入力して Apache サービスを再起動します。

```
sudo /opt/bitnami/ctlscript.sh restart apache
```

ディストリビューションに関する追加情報

Lightsail でのディストリビューションの管理に役立つ記事をいくつか紹介します。

- コンテンツ配信ネットワークディストリビューション
- ディストリビューションの作成

- ディストリビューションのリクエストとレスポンスを理解する
- ディストリビューションをテストする
- ディストリビューションのオリジンを変更する
- ディストリビューションのキャッシュ動作を変更する
- ディストリビューションのキャッシュをリセットする
- ディストリビューションのプランを変更する
- ディストリビューションのカスタムドメインを有効にする
- ドメインをディストリビューションにポイントする
- ディストリビューションのカスタムドメインを変更する
- ディストリビューションのカスタムドメインを無効にする
- ディストリビューションのメトリクスを表示する
- ディストリビューションを削除する

Lightsail の WordPress インスタンスでメールを有効にする

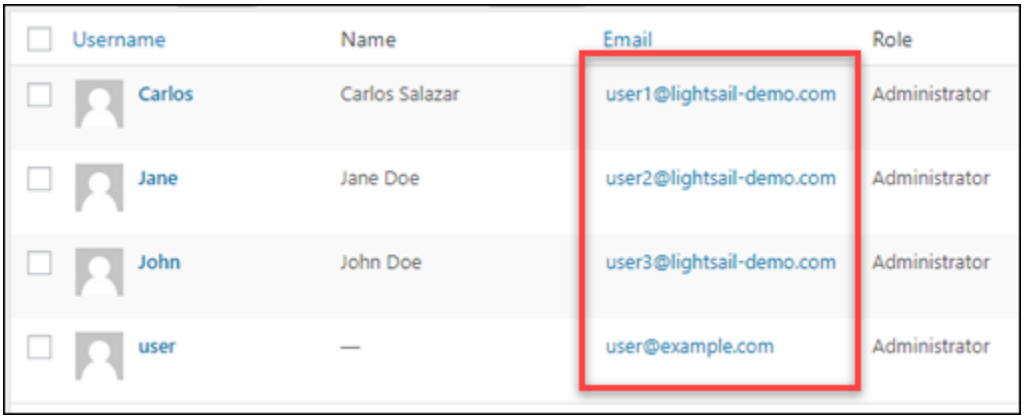
Amazon Lightsail の WordPress インスタンスで E メール を有効にできます。Amazon Simple Email Service (Amazon SES) で SMTP サービスを設定します。次に、インスタンスで WP Mail SMTP プラグインを有効化して設定します。E メールが有効になると、WordPress 管理者はユーザープロファイルのパスワードリセットをリクエストできます。ブログの投稿、ウェブサイトの更新、その他のプラグインメッセージに関する E メール通知が送信されます。このガイドでは、Amazon SES を使用して、Amazon Lightsail の WordPress インスタンスでメールを有効にする方法について説明します。

目次

詳細については、Amazon SES ドキュメントの「メールを送信するための Amazon SES SMTP インターフェイスの使用」を参照してください。

ステップ 1: 制限のレビュー

Amazon SES サンドボックスの新しい Amazon Web Services (AWS) アカウントは、確認済みのアドレスおよびドメインのみにメールを送信することができます。このような場合は、ウェブサイトのドメインを確認し、WordPress 管理者のメールアドレスを確認することをお勧めします。自分のメールアドレスを取得するには、WordPress ウェブサイトのダッシュボードにサインインし、左側のナビゲーションメニューで [ユーザー] を選択します。以下の例のように、[メール] 列に管理者のメールアドレスが表示されます。



> **Note**
>
> デフォルトの `user` プロファイルは `user@example.com`メールアドレスを使用して設定されます。これを有効なメールアドレスに変更する必要があります。詳細については、WordPress ドキュメントの「ユーザープロファイル画面」を参照してください。

任意のアドレスおよびドメインにメールを送信するには、アカウントを Amazon SES サンドボックスの外に移動するようにリクエストする必要があります。詳細については、Amazon SES ドキュメントの「Amazon SES サンドボックスの外への移動」を参照してください。

ステップ 2: 前提条件を完了させる

WordPress インスタンスで E メールを有効にするには、次のタスクを完了する必要があります。

- Lightsail で WordPress インスタンスを作成します。詳細については、「チュートリアル: Amazon Lightsail で WordPress インスタンスを起動して設定する」を参照してください。
- Lightsail DNS ゾーンを使用して登録されたドメインを WordPress インスタンスに選択します。詳細については、「DNS ゾーンを作成してドメインの DNS レコードを管理する」を参照してください。

- Amazon SES にサインアップし、サービスの詳細について参照します。Amazon SES へのサインアップの詳細については、Amazon SES ドキュメントの「Amazon SES クイックスタート」を参照してください。Amazon SES の詳細については、Amazon SES ドキュメントの以下のガイドを参照してください。
  - Amazon SES デベロッパーガイド
  - Amazon SES のよくある質問
  - Amazon SES 料金表
  - Amazon SES Service Quotas

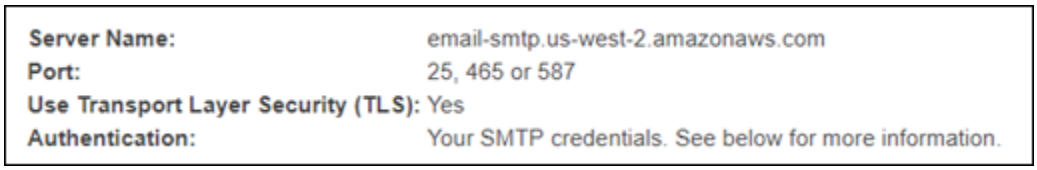
ステップ 3: Amazon SES で SMTP 認証情報を作成する

このガイドの後半で設定する WP Mail SMTP プラグインを設定するには、Amazon SES アカウントで SMTP 認証情報を作成する必要があります。詳細については、Amazon SES ドキュメントの「Amazon SES の SMTP 認証情報の取得」を参照してください。

Amazon SES で SMTP 認証情報を作成するには

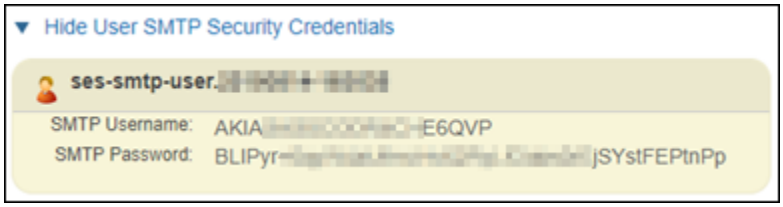
1. Amazon SES コンソールにサインインします。
2. 左側のナビゲーションメニューから、[SMTP 設定] を選択します。

   [SMTP 設定] ページには、SMTP サーバー名、ポート、および TLS 設定が表示されます。WordPress インスタンスの WP Mail SMTP プラグインを設定する際は、後で必要になるため、これらの値を記録します。

3. [SMTP 認証情報の作成] を選択します。
4. [IAM ユーザー名] テキストボックスで [作成] を選択します。ユーザー名はデフォルトのままにします。

5. [Show User SMTP Security Credentials (ユーザー SMTP セキュリティ認証情報の表示)] を選択して SMTP ユーザー名とパスワードを表示するか、[認証情報のダウンロード] を選択して同じ情報を含む CSV ファイルをダウンロードします。WordPress インスタンスで WP Mail SMTP プラグインを設定するときに、後でこれらの認証情報が必要になります。



> **Note**
>
> Amazon SES コンソールに作成された認証情報は、アカウントの AWS Identity and Access Management (IAM) に自動的に追加されます。

ステップ 4: Amazon SES のドメインを検証する

Amazon SES では、ドメインを検証して、それを所有していることを確認し、他のユーザーに使用されないようにする必要があります。ドメインを検証すると、そのドメインのすべてのメールアドレスを検証することになるため、そのドメインのメールアドレスを個別に検証する必要はありません。たとえば、ドメイン `example.com` を検証する場合、`user1@example.com`、`user2@example.com`、または `example.com` の他の任意のユーザーから E メールを送信できます。詳細については、Amazon SES ドキュメントの「Amazon SES のドメインの検証」を参照してください。

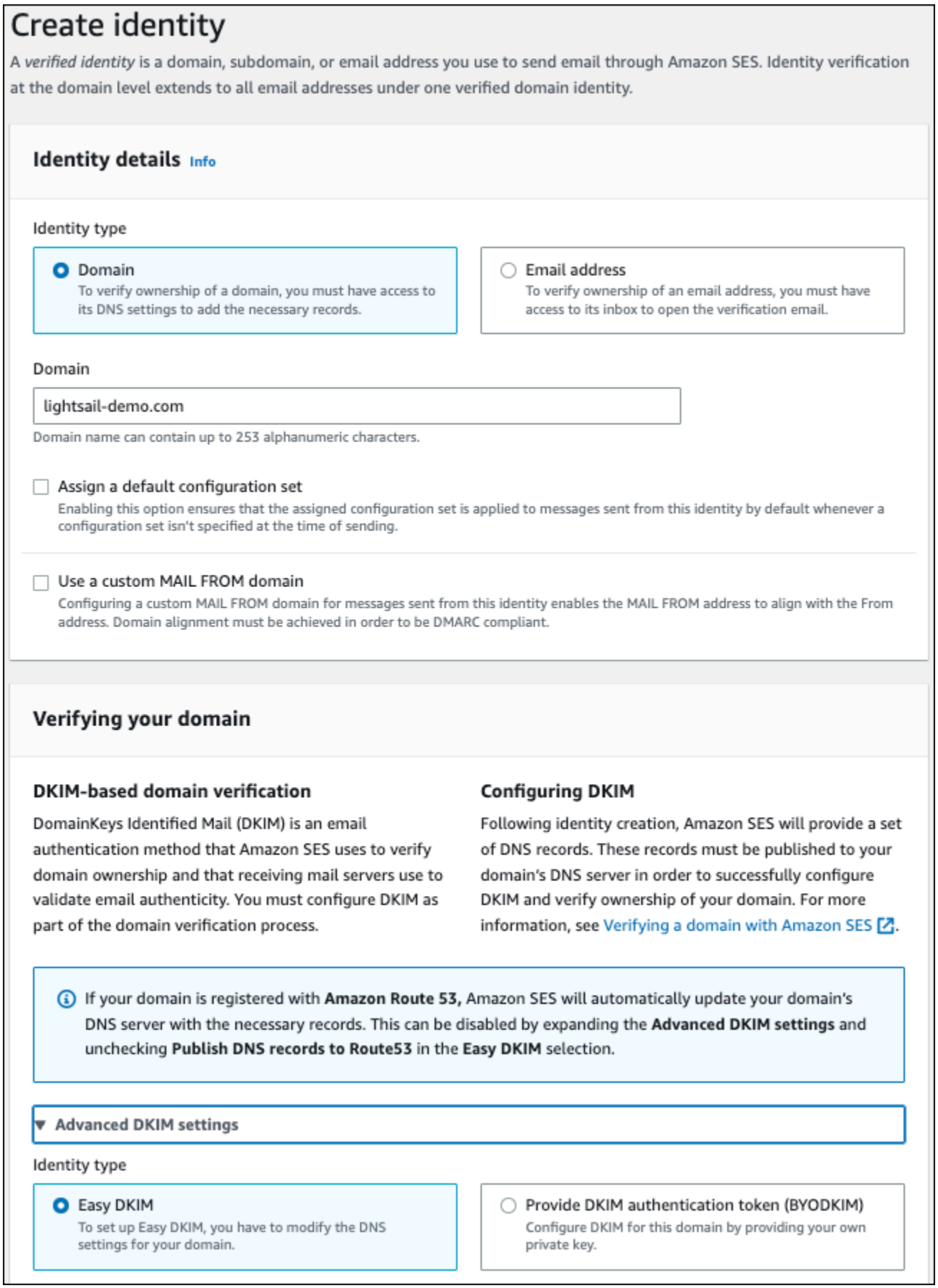
Amazon SES のドメインを検証するには

1. Amazon SES コンソールで、左ナビゲーションメニューから [検証済み ID] を選択します。
2. [ID の作成] を選択します。

3. 検証するドメインを入力し、[アイデンティティの作成] を選択します。

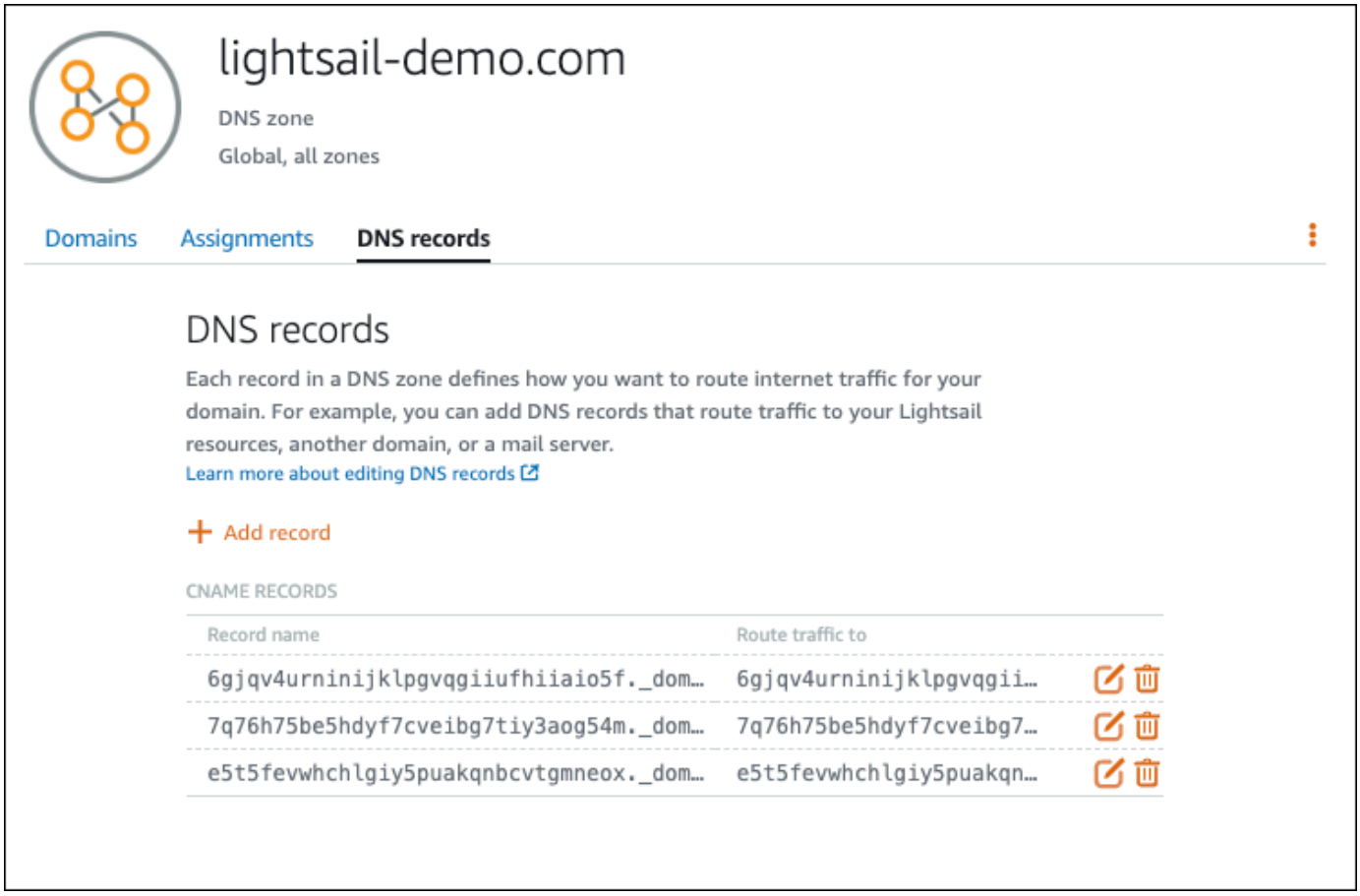
   検証するドメインは、Lightsail の WordPress インスタンスで使用しているドメインと一致している必要があります。

> **Important**
>
> レガシー TXT レコード
> Amazon SES のドメイン検証は、現在は DomainKeys Identified Mail (DKIM) に基づいています。これは受信メールサーバーがメールの信頼性を検証するために使用するメール認証規格です。ドメインの DNS 設定で DKIM を設定すると、ユーザーがアイデンティティの所有者であることが SES に確認されるため、TXT レコードは不要になります。TXT レコードを使用して検証されたドメインアイデンティティは再検証する必要はありませんが、DKIM 準拠のメールプロバイダーでメールを配信しやすくするため、DKIM 署名を有効にすることをお勧めします。


Create identity
A verified identity is a domain, subdomain, or email address you use to send email through Amazon SES. Identity verification at the domain level extends to all email addresses under one verified domain identity.
Identity details Info
Identity type
Domain
To verify ownership of a domain, you must have access to its DNS settings to add the necessary records.
Email address
To verify ownership of an email address, you must have access to its inbox to open the verification email.
Domain
lightsail-demo.com
Domain name can contain up to 253 alphanumeric characters.
Assign a default configuration set
Enabling this option ensures that the assigned configuration set is applied to messages sent from this identity by default whenever a configuration set isn't specified at the time of sending.
Use a custom MAIL FROM domain
Configuring a custom MAIL FROM domain for messages sent from this identity enables the MAIL FROM address to align with the From address. Domain alignment must be achieved in order to be DMARC compliant.
Verifying your domain
DKIM-based domain verification
DomainKeys Identified Mail (DKIM) is an email authentication method that Amazon SES uses to verify domain ownership and that receiving mail servers use to validate email authenticity. You must configure DKIM as part of the domain verification process.
Configuring DKIM
Following identity creation, Amazon SES will provide a set of DNS records. These records must be published to your domain's DNS server in order to successfully configure DKIM and verify ownership of your domain. For more information, see Verifying a domain with Amazon SES.
If your domain is registered with Amazon Route 53, Amazon SES will automatically update your domain's DNS server with the necessary records. This can be disabled by expanding the Advanced DKIM settings and unchecking Publish DNS records to Route53 in the Easy DKIM selection.
Advanced DKIM settings
Identity type
Easy DKIM
To set up Easy DKIM, you have to modify the DNS settings for your domain.
Provide DKIM authentication token (BYODKIM)
Configure DKIM for this domain by providing your own private key.

4. Easy DKIM でドメインアイデンティティを作成した後、ドメインの DNS プロバイダーに公開するため、次の生成した CNAME レコードをコピーして DKIM 認証での検証プロセスを完了する必要があります。これらのレコードの検出には最大 72 時間かかる場合があります。詳細については、「DKIM でドメインアイデンティティの検証」および「Easy DKIM」を参照してください。
5. 新しいブラウザタブを開き、Lightsail コンソールに移動します。
6. Lightsail ホームページで [ドメインと DNS] を選択し、ドメインの DNS ゾーンを選択します。
7. Amazon SES コンソールから DNS レコードを追加します。Lightsail のDNS ゾーンの編集の詳細については、「Edit a DNS zone in Amazon Lightsail の DNS ゾーンを編集する」を参照してください。

   結果は次の例のようになります。

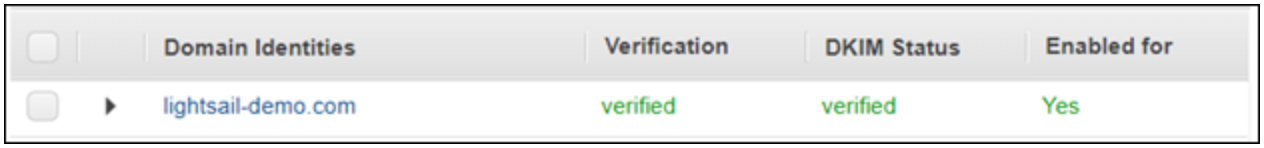


> **Note**
>
> [サブドメイン] テキストボックスに @ のシンボルを入力して、MX レコードにドメインの頂点を使用します。また、Amazon SES で指定された MX レコード値は 10

`inbound-smtp.us-west-2.amazonaws.com` になります。`10` を [Priority] (優先)、および `inbound-smtp.us-west-2.amazonaws.com` を [Maps to] (マップ先) としてドメインに入力します。

8. Amazon SES コンソールで、[新しいドメインを検証する] ページを閉じます。

   数分後、以下の例のように、Amazon SES コンソールに表示されるドメインには検証済みのラベルが付き、送信できるようになります。



   Amazon SES の SMTP サービスは、ドメインからメールを送信する準備ができました。

ステップ 5: Amazon SES のメールアドレスを検証する

Amazon SES の新規ユーザーの場合は、メールを送信する宛先のメールアドレスを検証する必要があります。これを行うには、Amazon SES コンソールにメールアドレスを追加します。詳細については、Amazon SES ドキュメントの「Amazon SES のメールアドレスの検証」を参照してください。

WordPress ウェブサイトの管理者のメールアドレスを追加することをお勧めします。こうすることで WordPress 管理者はユーザープロファイルのパスワードリセットをリクエストできます。またブログの投稿、ウェブサイトの更新、その他のプラグインメッセージに関する E メール通知を受信できます。

> **Note**
>
> 検証なしで任意のアドレスにメールを送信する場合は、Amazon SES アカウントをサンドボックスの外に移動するようリクエストする必要があります。詳細については、Amazon SES ドキュメントの「Amazon SES サンドボックスの外への移動」を参照してください。

E メールアドレスの ID を作成するには

1. Amazon SES コンソールで、左ナビゲーションメニューから [検証済み ID] を選択します。
2. [ID の作成] を選択します。
3. [メールアドレス] を選択します。次に、検証するメールアドレスを入力します。

4. [ID の作成] を選択します。

検証するメールアドレスすべてに対し、ステップ 1～4 を繰り返します。確認メールが入力したメールアドレスに送信されます。アドレスが「検証待ち」ステータスで検証済みの E メール ID のリストに追加されます。ユーザーが E メールメッセージを開いて検証プロセスを完了すると、ステータスは「検証済み」と表示されます。

E メールアドレス ID を検証するには

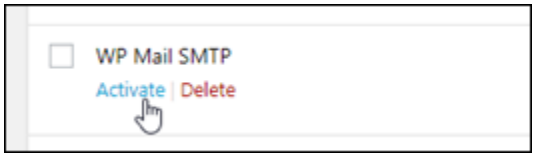
1. ID の作成に使用したメールアドレスの受信トレイを確認し、no-reply-aws@amazon.com からのメールを探します。
2. E メールを開き、E メールのリンクをクリックして、E メールアドレスの検証プロセスを完了します。完了したら、ID の状態が検証済みに更新されます。

ステップ 6: WordPress インスタンスに WP Mail SMTP プラグインを設定する

最後のステップでは、WordPress インスタンスに WP Mail SMTP プラグインを設定します。Amazon SES コンソールで、このガイドで先に作成した SMTP 認証情報を使用します。

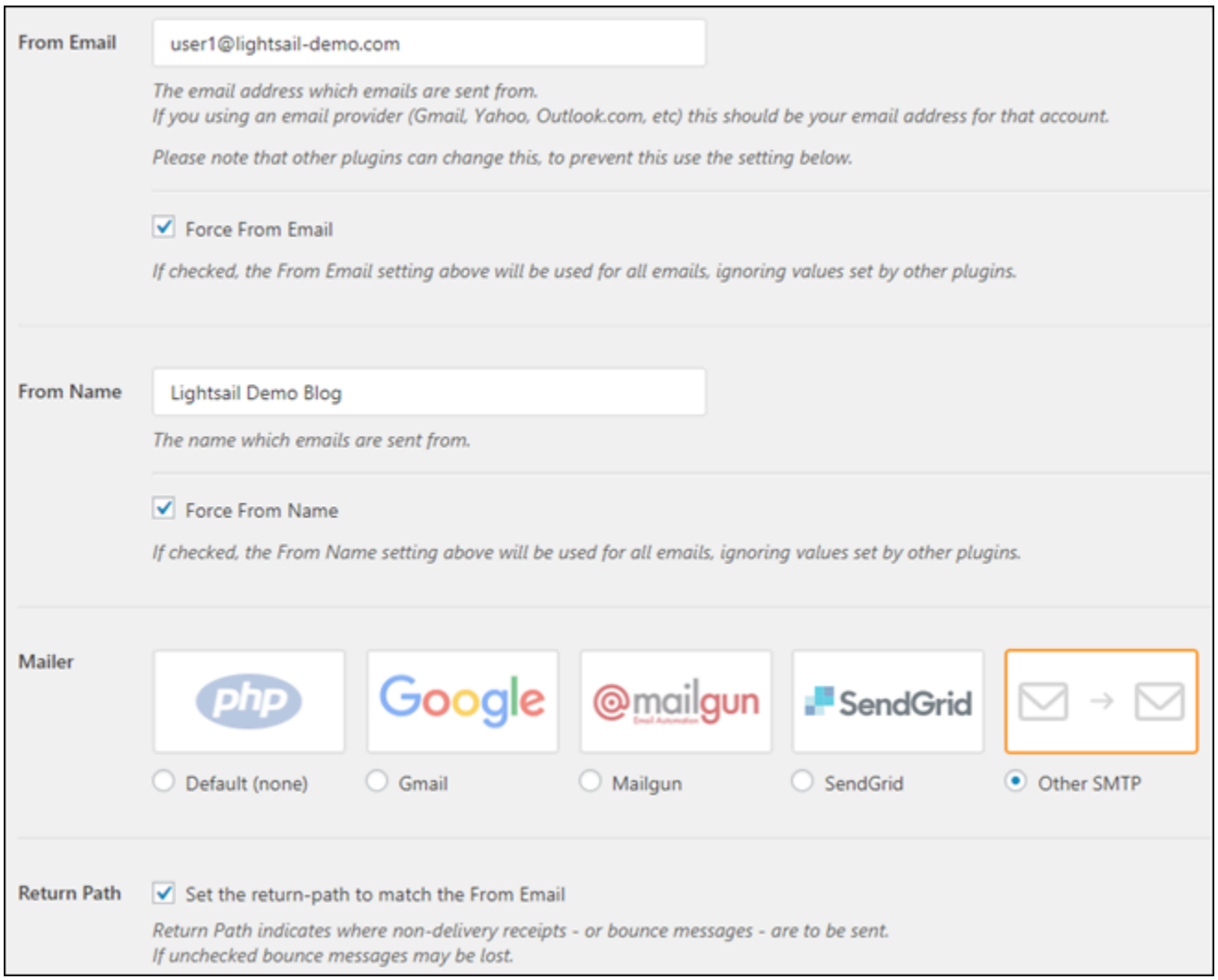
WordPress インスタンスで WP Mail SMTP プラグインを設定するには

1. 管理者として WordPress ウェブサイトのダッシュボードにサインインします。
2. 左側のナビゲーションメニューから、[プラグイン] を選択し、続いて [Installed Plugins (インストール済みのプラグイン)] を選択します。
3. WP Mail SMTP プラグインまで下にスクロールし、[有効化] を選択します。プラグインの新しいバージョンがある場合は、次のステップに進む前に更新してください。

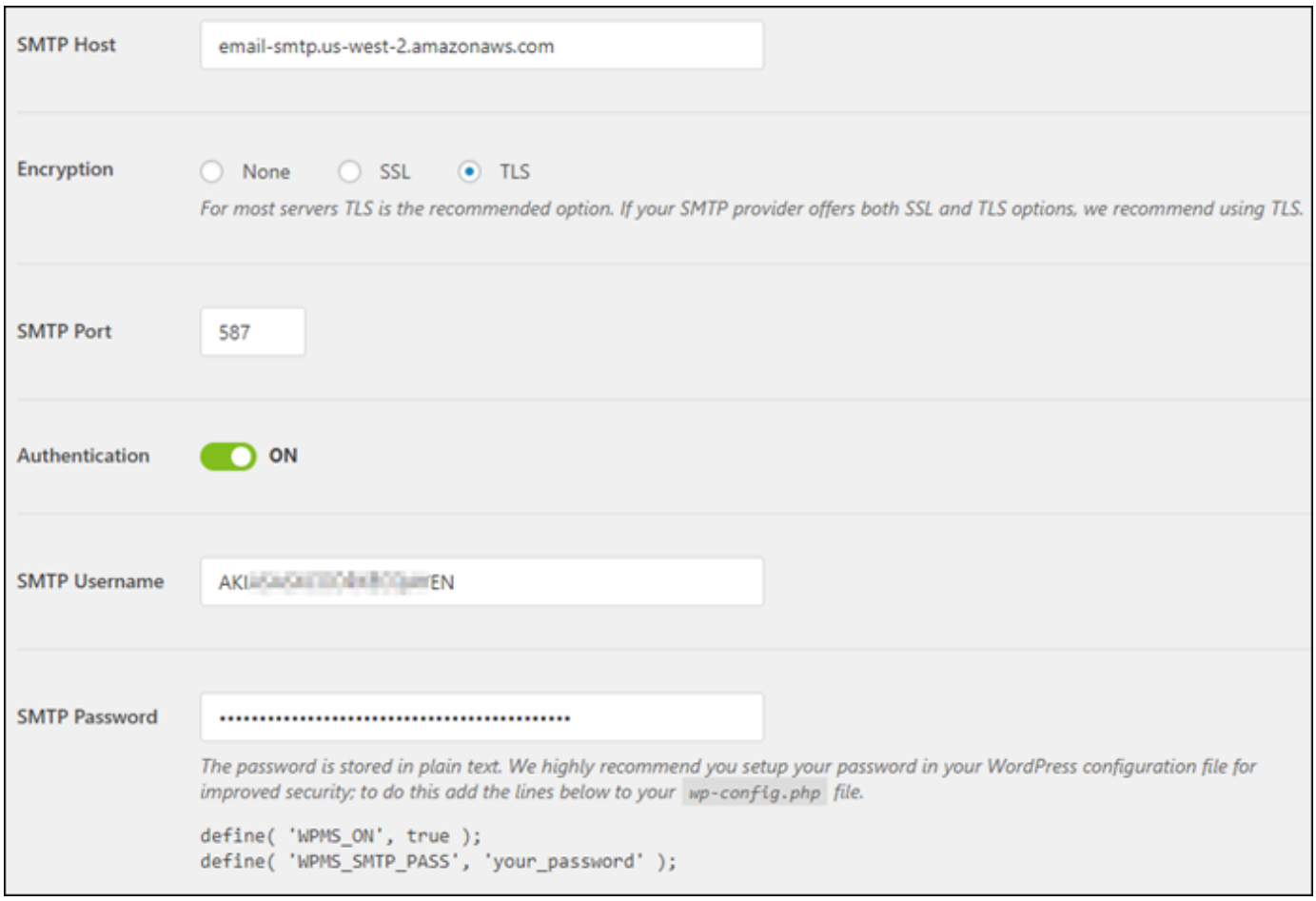


4. WP Mail SMTP プラグインが有効になったら、[設定] を選択します。下にスクロールしてプラグインを見つける必要がある場合があります。

5. [送信元メールアドレス] テキストボックスに、メールの送信元のメールアドレスを入力します。入力するメールアドレスは、このガイドの先のステップを使用して、Amazon SES で確認されている必要があります。
6. [メールから実行] を選択して、[送信元メールアドレス] テキストボックスで入力するメールアドレスを使用して実行し、他のプラグインで設定された「送信元メールアドレス」値を無視します。
7. [From Name (送信元名)] テキストボックスに E メールを送信する送信元の名前を入力するか、そのままにして WordPress ブログの名前を使用します。
8. [Force From Name (名前から実行)] を選択して、[From Name (送信元名)] テキストボックスに入力した名前を使用して実行します。このオプションを選択すると他のプラグインによって設定された「送信元名」値は無視され、WordPress は [From Name (送信元名)] テキストボックスに入力する名前を使用します。
9. ページのメーラーセクションで、[Other SMTP mailer (その他の SMTP メーラー)] を選択します。
10. [Return-Path を送信元メールアドレスに一致するよう設定する] を選択して、配信不能レシートが [送信元メールアドレス] テキストボックスで入力するメールアドレスに送信されるように設定します。



11. [SMTP ホスト] テキストボックスに、このガイドの前半で Amazon SES コンソールの [SMTP の設定] ページから取得した、SMTP サーバー名を入力します。
12. このページの [暗号化] セクションで [TLS] を選択して、Amazon SES の SMTP サービスが TLS 暗号化を使用していることを確認します。
13. [SMTP ポート] テキストボックスは、デフォルトの値 [587] のままにしておきます。
14. [認証] トグルを [オン] に切り替え、このガイドの前半で Amazon SES コンソールから取得した、SMTP ユーザー名とパスワードを入力します。

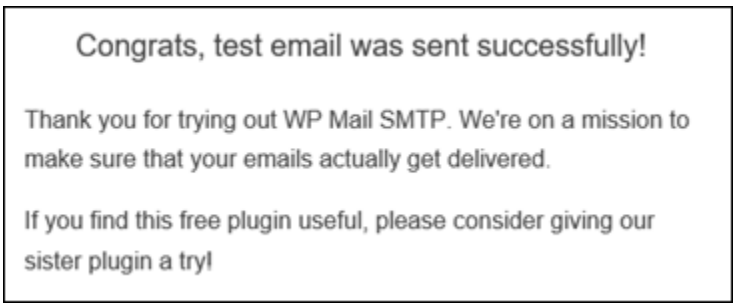


15. [Save settings (設定を保存)] を選択します。設定が正常に保存されたことを確認するプロンプトが表示されます。
16. [Email Test (E メールテスト)] タブを選択します。

    次のステップで、テスト用の E メールを送信して E メールサービスが動作していることを確認します。
17. [送信先] テキストボックスにメールアドレスを入力し、[メールの送信] を選択します。入力するメールアドレスは、このガイドの先のステップを使用して、Amazon SES で確認されている必要があります。

    2 つの可能な結果が表示されるはずです。

    - 成功の確認が表示された場合は、WordPress ウェブサイトは E メールが有効になっています。以下のテスト E メールが指定されたメールボックスに到達することを確認します。



WordPress ウェブサイトのサインインページで [Lost your password? (パスワードを忘れましたか?)] が選択できるようになりました。Amazon SES で WordPress ユーザープロファイルのメールアドレスが確認されると、新しいパスワードがメールで送信されます。

- 失敗通知が表示された場合は、WP Mail SMTP プラグインに入力した SMTP 設定が、Amazon SES アカウントの SMTP サービスのものと一致していることを確認します。また、Amazon SES で検証したメールアドレスを使用していることを確認します。

Lightsail WordPress のインスタンスで HTTPS を有効にする

WordPress ウェブサイトでハイパーテキスト転送プロトコルセキュア (HTTPS) を有効にすると、訪問者はウェブサイトが安全であること、暗号化されたデータを送受信していることを保証できます。セキュリティで保護されていないウェブサイトのアドレスは `http://example.com` などの、`http` で始まり、セキュリティで保護されたウェブサイトのアドレスは `https://example.com` などの `https` で始まります。ウェブサイトが主に情報提供を目的としたものでも、HTTPS を有効にすることをお勧めします。これは、HTTPS が有効になっていない場合、ほとんどのウェブブラウザがウェブサイトの訪問者にウェブサイトが安全でないことを通知し、その結果ウェブサイトの検索エンジンの結果でランクが下がるためです。

> **Tip**
>
> Lightsail には、インスタンスへの SSL/TLS Let's Encrypt 証明書のインストールと設定を自動化するガイド付きワークフローが用意されています。 WordPress このチュートリアルの手動の手順に従うのではなく、このワークフローを使用することを強くお勧めします。詳細については、「WordPress インスタンスの起動と設定」を参照してください。

このガイドでは、Bitnami HTTPS 設定ツール (`bncert`) を使用して Amazon Lightsail の Certified by Bitnami WordPress インスタンスで HTTPS を有効にする方法を説明します。これは、リクエスト時に指定するドメインおよびサブドメインに対してのみ証明書を要求することを許可します。また

Certbot を使用して、ドメインに証明書を、そしてサブドメインにワイルドカード証明書をリクエストできます。ワイルドカード証明書はドメインのすべてのサブドメインに使用できます。これは、トラフィックをインスタンスに誘導するために使用するサブドメインがどれかわからない場合に役立ちます。ただし、bncert ツールと違い、Certbot は証明書を自動的に更新しません。Certbot を使用する場合は、90 日ごとに証明書を手動で更新する必要があります。Certbot を使用して HTTPS を有効にする方法の詳細については、「チュートリアル:インスタンスで Let's Encrypt SSL 証明書を使用する」を参照してください。 WordPress

目次

ステップ 1: プロセスについて学ぶ

> **Note**
>
> このセクションでは、プロセスの高度な概要を説明します。このプロセスを実行する具体的なステップについては、このガイドの以降のステップで説明します。

WordPress ウェブサイトで HTTPS を有効にするには、SSH を使用して Lightsail インスタンスに接続し、bncertツールを使用して Let's Encrypt 認証局に SSL/TLS 証明書をリクエストします。証明書をリクエストする際は、ウェブサイトのプライマリドメイン (example.com) や代替ドメイン (www.example.com や blog.example.com など) を指定します。Let's Encrypt は、ドメインの DNS で TXT レコードを作成するように求めるか、またはそれらのドメインがリクエスト元のインスタンスのパブリック IP アドレスにトラフィックをすでに送信していることを確認することによって、ドメインを所有していることを確認します。

証明書が検証されたら、訪問者を HTTP から HTTPS に自動的にリダイレクト (http://example.comリダイレクト先https://example.com) WordPress するようにウェブサイトを設定して、訪問者に暗号化された接続を強制的に使用させることができます。また、www サブドメ

インをドメインの頂点 (https://www.example.com を https://example.com にリダイレクト) またはその逆 (https://example.com を https://www.example.com にリダイレクト) に自動的にリダイレクトするようにウェブサイトを設定することもできます。これらのリダイレクトは、bncert ツールを使って設定することもできます。

Let's Encrypt では、ウェブサイトで HTTPS を維持するために 90 日ごとに証明書を更新する必要があります。bncert ツールは証明書を自動的に更新するので、ウェブサイトに専念する時間を増やすことができます。

bncert ツールの制限事項

bncert ツールには次の制約事項があります。

- Certifated by WordPress Bitnami インスタンスの作成時にすべてにプリインストールされているわけではありません。 WordPress しばらく前に Lightsail で作成されたインスタンスでは、ツールを手動でインストールする必要があります。bncertこのガイドのステップ 4 は、ツールがインスタンスにインストールされていることを確認する方法と、されていない場合にインストールする方法を示します。
- 証明書をリクエストできるのは、リクエスト時に指定したドメインおよびサブドメインに対してのみです。ドメインの証明書とサブドメインのワイルドカード証明書のリクエストを可能にする Certbot ツールとは異なります。ワイルドカード証明書はどのサブドメインにも使用できます。これは、トラフィックをインスタンスに誘導するために使用するサブドメインがわからない場合に役立ちます。ただし、bncert ツールと違い、Certbot は証明書を自動的に更新しません。Certbot を使用する場合は、90 日ごとに証明書を手動で更新する必要があります。Certbot を使用して HTTPS を有効にする方法の詳細については、「チュートリアル:Amazon Lightsail WordPress のインスタンスで SSL 証明書を暗号化しよう」を参照してください。

ステップ 2: 前提条件を完了させる

以下の前提条件を満たします (まだ満たしていない場合)。

- Lightsail WordPress でインスタンスを作成し、そのインスタンスでウェブサイトを設定します。詳細については、「Amazon Lightsail で Linux/UNIX ベースのインスタンスを使い始める」を参照してください。
- 静的 IP をインスタンスに添付します。インスタンスを停止してまた開始すると、インスタンスのパブリック IP アドレスは変わります。インスタンスを停止してまた開始しても、静的 IP は変更されません。詳細については、「静的 IP を作成してAmazon Lightsail のインスタンスにアタッチする」を参照してください。

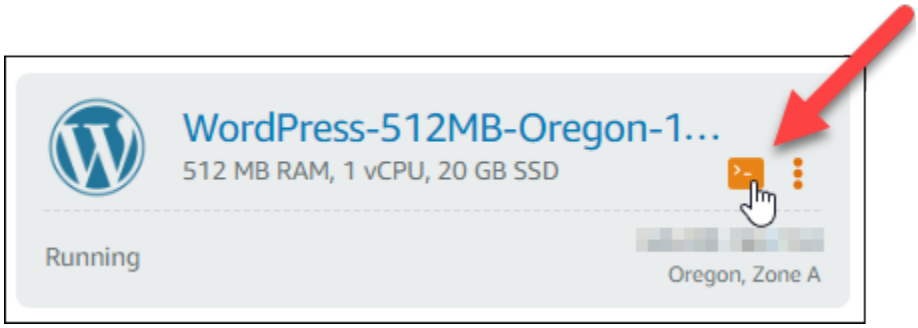
- WordPress 設定が完了したらインスタンスのスナップショットを作成するか、自動スナップショットを有効にします。スナップショットは、インスタンスに何か問題が発生した場合、これを元に別のインスタンスを作成できるバックアップとして使用できます。詳細については、「Linux または Unix インスタンスのスナップショットの作成」または「Amazon Lightsail のインスタンスまたはディスクの自動スナップショットの有効化または無効化」を参照してください。
- ドメイン Apex () `www` とそのサブドメイン (`example.com`) のトラフィックを Lightsail WordPress のインスタンスのパブリック IP アドレスに転送する DNS レコードをドメインの DNS に追加します。`www.example.com`これらのアクションは、ドメインの現在の DNS ホスティングプロバイダーで実行することができます。また、ドメインの DNS の管理を Lightsail に移管した場合は、Lightsail の DNS ゾーンを使用してこれらのアクションを実行できます。詳細については、「DNS」を参照してください。

> **Important**
>
> ウェブサイトで使用したいすべてのドメインの DNS に DNS レコードを追加します。WordPressこれらのドメインはすべて、 WordPress ウェブサイトのパブリック IP アドレスにトラフィックをルーティングする必要があります。`bncert`このツールは、WordPress現在トラフィックをインスタンスのパブリック IP アドレスに転送しているドメインに対してのみ証明書を発行します。

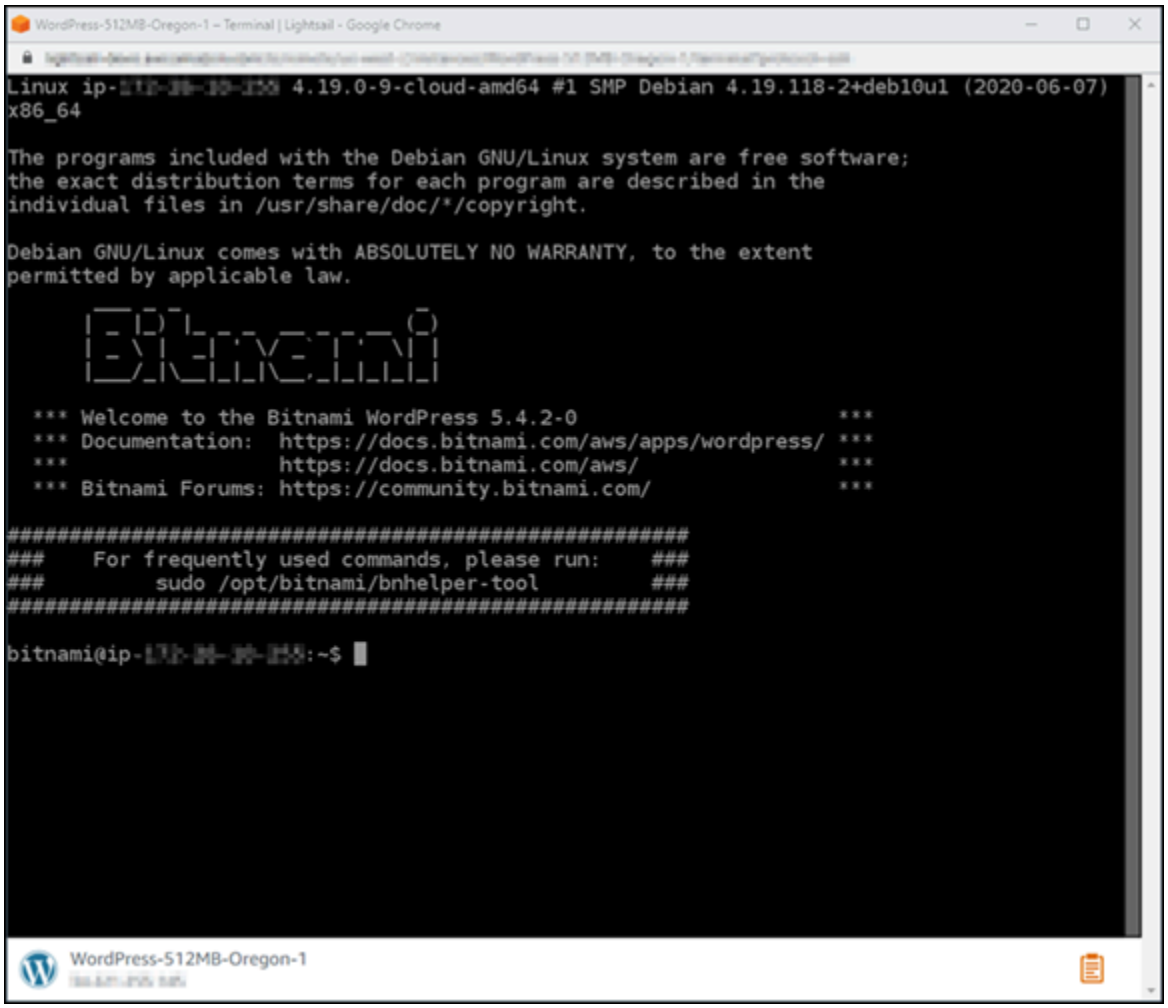
ステップ 3: インスタンスに接続する

Lightsail コンソールのブラウザベースの SSH クライアントを使用してインスタンスに接続するには、以下の手順を実行します。

1. Lightsail コンソールにサインインします。
2. Lightsail ホームページで、インスタンスの SSH クイック接続アイコンを選択します。WordPress



ブラウザベースの SSH クライアントターミナルウィンドウが開きます。SSH 経由でインスタンスに正常に接続されていると、次の例に示すように Bitnami 口ゴが表示されます。



ステップ 4: インスタンスに bncert ツールがインストールされていることを確認

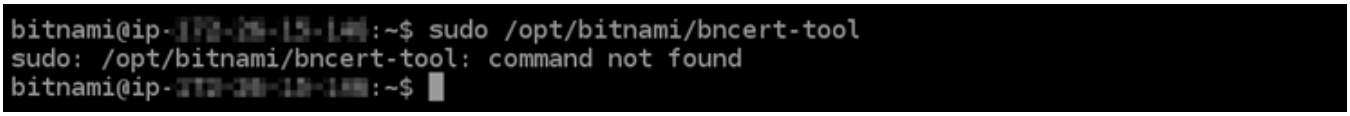
次のステップを完了して Bitnami HTTPS 設定ツール (bncert) がインスタンスにインストールされていることを確認します。すべての Certified by WordPress Bitnami インスタンスの作成時にプリインストールされているわけではありません。 WordPress しばらく前に Lightsail で作成されたインスタンスでは、ツールを手動でインストールする必要があります。bncertこのステップでは、ツールがインストールされていない場合にツールをインストールする方法を説明します。

1. bncert ツールを実行するには、次のコマンドを入力します。

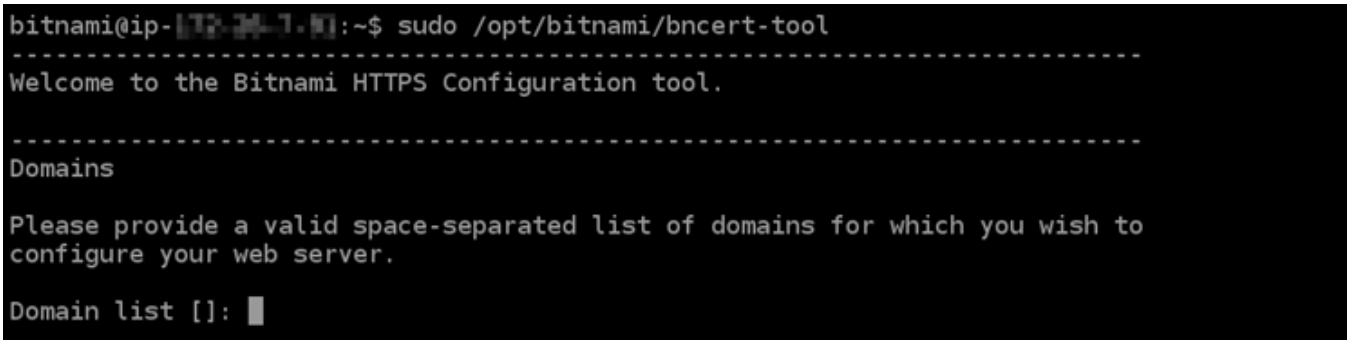
```
sudo /opt/bitnami/bncert-tool
```

- 次の例に示すように、command not found が応答で表示された場合、これは bncert ツールがインストールされていないことを示します。このステップの次のステップに進み、bncert ツールをインスタンスにインストールします。

> **Important**
>
> bncertこのツールは Bitnami WordPress によって認定されたインスタンスでのみ使用できます。または、Certbot ツールを使用してインスタンスで HTTPS を有効にすることもできます。 WordPress 詳細については、「チュートリアル:インスタンスで Let's Encrypt SSL 証明書を使用する」を参照してください。 WordPress



- 次の例に示すように、Welcome to the Bitnami HTTPS configuration tool がレスポンスで表示された場合は、bncert ツールがインストールされていることを示します。このガイドの「ステップ 5: WordPress インスタンスで HTTPS を有効にする」セクションに進んでください。



2. 以下のコマンドを入力して、bncert 実行ファイルをインスタンスにダウンロードします。

```
wget -O bncert-linux-x64.run https://downloads.bitnami.com/files/bncert/latest/bncert-linux-x64.run
```

3. 以下のコマンドを入力して、bncert 実行ファイルへのディレクトリを作成します。

```
sudo mkdir /opt/bitnami/bncert
```

4. 以下のコマンドを入力して、ダウンロードした bncert 実行ファイルを、作成した新しいディレクトリに移動させます。

```
sudo mv bncert-linux-x64.run /opt/bitnami/bncert/
```

5. 以下のコマンドを入力して、プログラムとして実行できるファイルを bncert に実行させます。

```
sudo chmod +x /opt/bitnami/bncert/bncert-linux-x64.run
```

6. 次のコマンドを入力することによって、sudo /opt/bitnami/bncert-tool コマンドを入力すると bncert ツールを実行するシンボリックリンクを作成します。

```
sudo ln -s /opt/bitnami/bncert/bncert-linux-x64.run /opt/bitnami/bncert-tool
```

これで bncert ツールのインストールは完了しました。このガイドの「ステップ 5: WordPress インスタンスで HTTPS を有効にする」セクションに進んでください。

ステップ 5: WordPress インスタンスで HTTPS を有効にする

bncertツールがインスタンスにインストールされていることを確認したら、 WordPress 次の手順を実行してインスタンスで HTTPS を有効にします。

1. bncert ツールを実行するには、次のコマンドを入力します。

```
sudo /opt/bitnami/bncert-tool
```

次の例に示すようなメッセージが表示されます。

```
bitnami@ip-[redacted]:~$ sudo /opt/bitnami/bncert-tool
------------------------------------------------------------------------
Welcome to the Bitnami HTTPS Configuration tool.

------------------------------------------------------------------------
Domains

Please provide a valid space-separated list of domains for which you wish to
configure your web server.

Domain list []:
```

bncert ツールがしばらく前にインスタンスにインストールされていると、ツールの更新バージョンが利用可能であることを示すメッセージが表示される場合があります。次の例に示すよう

に、ダウンロードすることを選択し、sudo /opt/bitnami/bncert-tool コマンドを入力して bncert ツールを再度実行します。

```
bitnami@ip-[redacted]:~$ sudo /opt/bitnami/bncert-tool
An updated version is available. Would you like to download it? You would need to run it
manually later. [Y/n]: Y
```

2. 次の例に示すように、プライマリドメイン名と代替ドメイン名の間はスペースで区切って入力します。

   ドメインがインスタンスのパブリック IP アドレスにトラフィックをルーティングするように設定されていない場合、bncert ツールは、続行する前にその設定を行うように要求します。ドメインは、bncert ツールを使用して HTTPS を有効にしているインスタンスでのパブリック IP アドレスにトラフィックをルーティングする必要があります。これはドメインを所有していることを確認し、証明書の検証として機能します。

```
------------------------------------------------------------------------------
Welcome to the Bitnami HTTPS Configuration tool.

------------------------------------------------------------------------------
Domains

Please provide a valid space-separated list of domains for which you wish to
configure your web server.

Domain list []: example.com www.example.com
```

3. bncert ツールは、ウェブサイトのリダイレクトの設定方法を尋ねます。使用できるオプションは次のとおりです。

   - HTTP から HTTPS へのリダイレクトを有効にする - HTTP バージョンのウェブサイトを閲覧するユーザー (例: http:/example.com) を自動的に HTTPS バージョン (例: https://example.com) にリダイレクトするかどうかを決定します。すべての訪問者が暗号化された接続を使用するように強制されるため、このオプションを有効にすることをお勧めします。Y を入力して Enter を押すと、有効になります。
   - www なしから www ありへのリダイレクトの有効化 - ドメインの頂点 (例: https://example.com) まで閲覧するユーザー を自動的にドメインの www サブドメイン (例: https://www.example.com) にリダイレクトするかを指定します。このオプションを有効にすることをお勧めします。ただし、ドメインの頂点を Google のウェブマスターツールなどの検索エンジンツールで希望のウェブサイトアドレスとして指定した場合、または頂点が IP を直接指しており、www のサブドメインが CNAME レコードを介してリファレンスしている場合は、無効にして代替オプションを有効にすることをお勧めします (www ありから www なしへのリダイレクトを有効化) 。Y を入力し、Enter を押して有効にします。

- www ありから www なしへのリダイレクトを有効にする - ドメインの www サブドメイン (例: `https://www.example.com`) まで閲覧するユーザーを、自動的にドメインの頂点 (例: `https://example.com`) にリダイレクトするかを指定します。www なしから www ありへのリダイレクトを有効にした場合は、これを無効にすることをお勧めします。N を入力し、Enter を押して無効にします。

選択した結果は次の例のようになります。

```
Enable/disable redirections

Please select the redirections you wish to enable or disable on your Bitnami
installation.



Enable HTTP to HTTPS redirection [Y/n]: Y



Enable non-www to www redirection [Y/n]: Y



Enable www to non-www redirection [y/N]: N
```

4. これから実行される変更が一覧表示されます。Y と入力し、Enter を押して確認し、続行します。

```
Changes to perform

The following changes will be performed to your Bitnami installation:

1. Stop web server
2. Configure web server to use a free Let's Encrypt certificate for the domains:
example.com www.example.com
3. Configure a cron job to automatically renew the certificate each month
4. Configure web server name to: example.com
5. Enable HTTP to HTTPS redirection (example: redirect http://example.com to
https://example.com)
6. Enable non-www to www redirection (example: redirect example.com to
 www.example.com)
7. Start web server once all changes have been performed



Do you agree to these changes? [Y/n]: Y
```

5. Let's Encrypt 証明書に関連付けるメールアドレスを入力し、Enter を押します。

```
Create a free HTTPS certificate with Let's Encrypt

Please provide a valid e-mail address for which to associate your Let's Encrypt
certificate.

Domain list: example.com www.example.com

Server name: example.com

E-mail address []:
```

6. Let's Encrypt サブスクライバー合意書を確認します。Y と入力し、Enter を押して契約に同意し、続行します。

```
The Let's Encrypt Subscriber Agreement can be found at:

https://letsencrypt.org/documents/LE-SA-v1.2-November-15-2017.pdf

Do you agree to the Let's Encrypt Subscriber Agreement? [Y/n]:
```

これらのアクションは、証明書のリクエストや指定したリダイレクトの設定など、インスタンスで HTTPS を有効にするために実行されます。

```
Performing changes to your installation

The Bitnami HTTPS Configuration Tool will perform any necessary actions to your
Bitnami installation. This may take some time, please be patient.

|
```

次の例のようなメッセージが表示された場合は、証明書は正常に発行され、検証され、インスタンスでリダイレクトが正常に設定されています。

```
Success

The Bitnami HTTPS Configuration Tool succeeded in modifying your installation.

The configuration report is shown below.

Backup files:
* /opt/bitnami/apache2/conf/httpd.conf.back.202005290035
* /opt/bitnami/apache2/conf/bitnami/bitnami-apps-prefix.conf.back.202005290035
* /opt/bitnami/apache2/conf/bitnami/bitnami.conf.back.202005290035

Find more details in the log file:

/tmp/bncert-202005290035.log

If you find any issues, please check Bitnami Support forums at:

https://community.bitnami.com

Press [Enter] to continue:
```

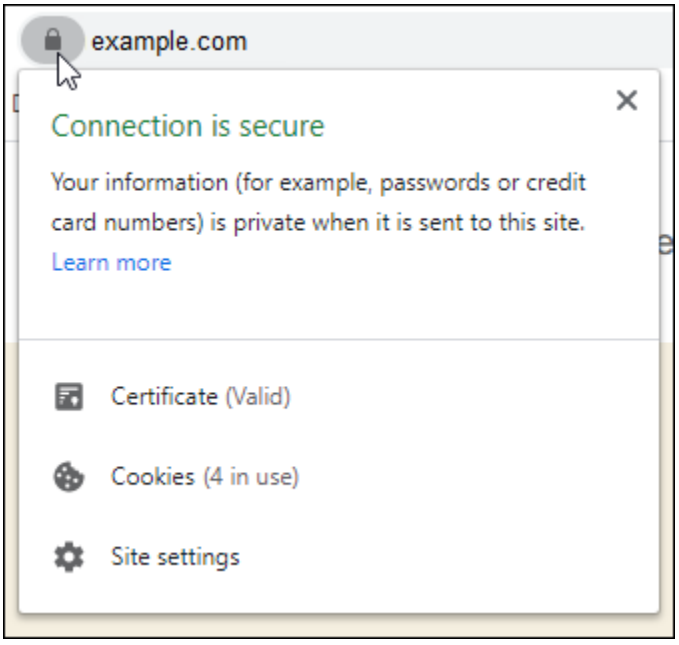
bncert ツールは、有効期限が切れる前、 80 日ごとに証明書の自動更新を実行します。インスタンスで追加のドメインやサブドメインを使用し、それらのドメインで HTTPS を有効にする場合は、上記のステップを繰り返します。

これで、 WordPress インスタンスでの HTTPS の有効化が完了しました。本ガイドのステップ 6: ウェブサイトで HTTPS を使用しているかどうかをテストするセクションに進んでください。

ステップ 6: ウェブサイトで HTTPS を使用しているかどうかをテストする

WordPress インスタンスで HTTPS を有効にしたら、bncertツールの使用時に指定したすべてのドメインを参照して、ウェブサイトが HTTPS を使用していることを確認する必要があります。次の例に示すように、各ドメインにアクセスすると、セキュリティで保護された接続を使用していることがわかります。

> **Note**
>
> 変更を確認するには、ブラウザのキャッシュを更新し、消去する必要がある場合もあります。



bncert ツールの実行時に選択したオプションに応じて、www なしアドレスがドメインの www ありサブドメインへリダイレクトするか、その逆が実行されます。

既存の WordPressブログを Amazon Lightsail に移行する

WordPress ホスティングプロバイダーを変更する場合 Amazon Lightsail は、 で WordPress サイトを実行する最も簡単な方法ですAWS。

当社の料金プラン (1 か月あたり 3.50 USD から) の 1 つを選択し、プラグイン、テーマなど、WordPress インストールを完全に制御できます。

Lightsail WordPress インスタンスの作成には数分しかかかりません。このチュートリアルに従って既存の WordPress ブログをバックアップし、Lightsail で実行されている新しいインスタンスにインポートします。

プロセスの簡単な概要を次に示します。

引き続き「」を読み、使用を開始します。

前提条件

始める前に、以下の準備が必要です。

1. AWS アカウントが必要です。AWS にサインアップするか、アカウントを既にお持ちの場合は AWS にサインインしてください。
2. Lightsail を使用するようにアカウントが設定されていることを確認します。アカウントを作成してから時間が経っている場合、またはクレジットカード情報をまだ入力していない場合は、まず AWS Management Console にログインしてアカウントを更新する必要があります。

ステップ 1: 既存の WordPress ブログをバックアップする

WordPress を使用して、既存のブログをバックアップできます。 WordPress 管理者コンソールにログインしてブログを管理するだけで済みます。

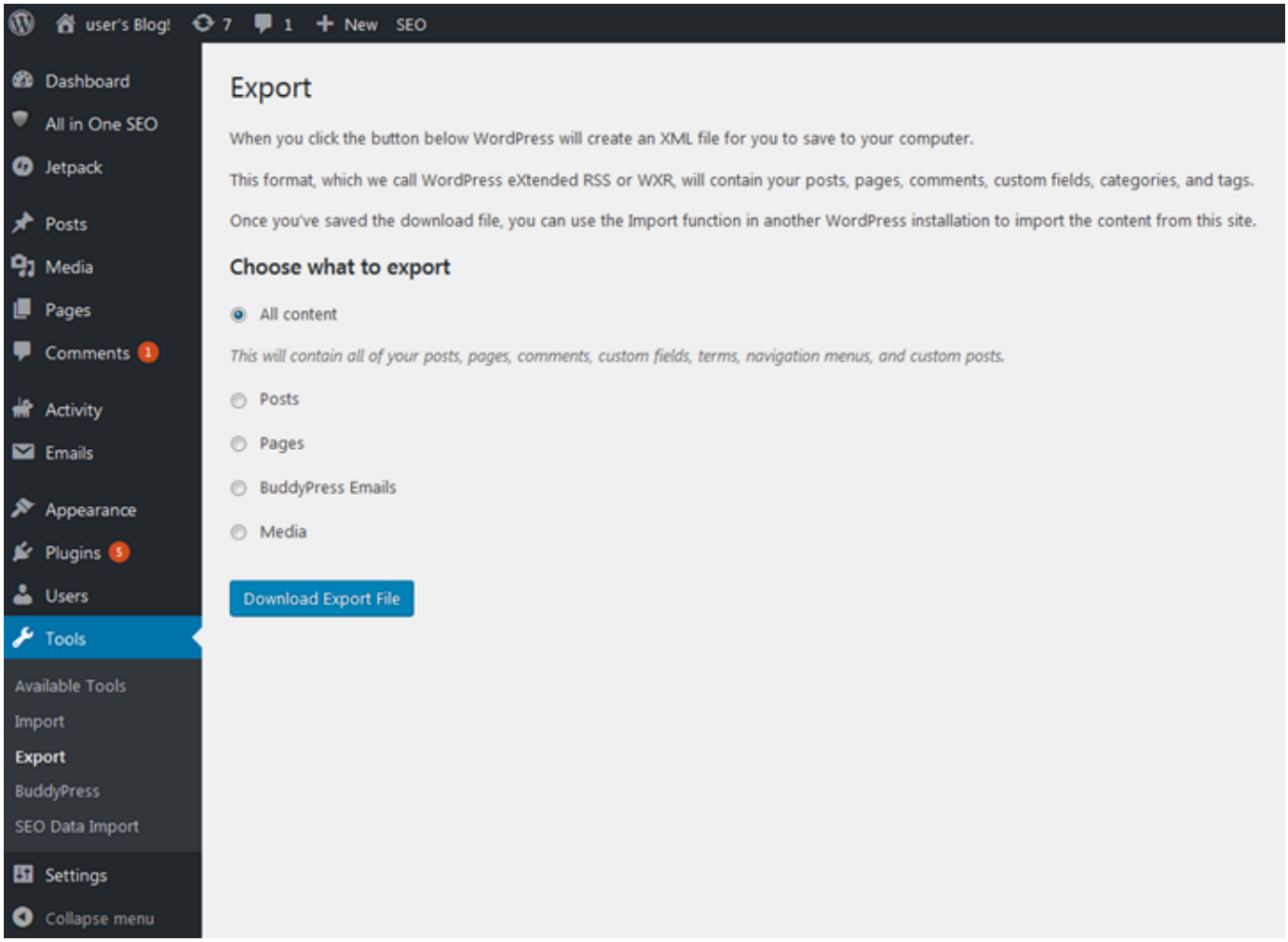
1. ブログに移動して [管理] を選択します。

   [Manage] (管理) バナーが表示されない場合は、`http://<PublicIP>/wp-login.php` を参照することでサインインページにアクセスすることができます。*<PublicIP>* を、インスタンスのパブリック IP アドレスに置き換えます。

2. WordPress 管理者コンソールにログインするには、ユーザー名とパスワードを入力します。

3. WordPress ダッシュボード で、ツール を選択し、エクスポート を選択します。

4. [エクスポート] ページで、[すべてのコンテンツ] を選択して、すべてを XML ファイルとしてエクスポートします。



5. [エクスポートファイルをダウンロード] を選択して、以前のブログを XML ファイルとしてダウンロードします。

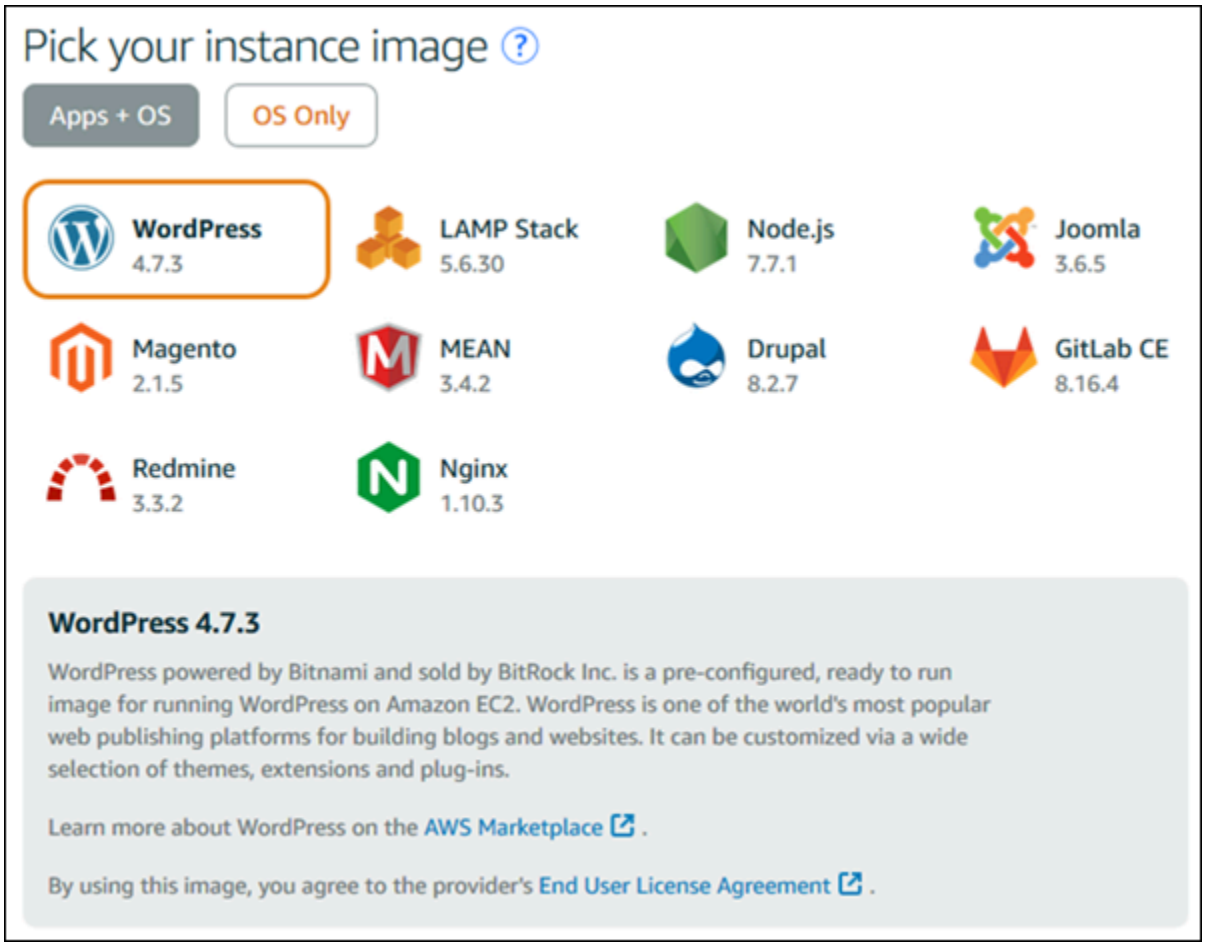
   その XML ファイルを見つけやすい場所に保存します。このファイルはステップ 4 で必要になります。

ステップ 2: Lightsail で新しい WordPress インスタンスを作成する

Lightsail で新しい WordPress インスタンスをわずか数分で作成できます。その方法は次のとおりです。

1. Lightsail のホームページに移動してログインします。
2. [インスタンスの作成] を選択します。
3. ブログを作成する AWS リージョン リージョンを選択します。

   AWS リージョン を選択したら、デフォルトのアベイラビリティーゾーンを選択または変更できます。
4. を選択しますWordPress。



5. インスタンスプラン (またはバンドル) を選択します。

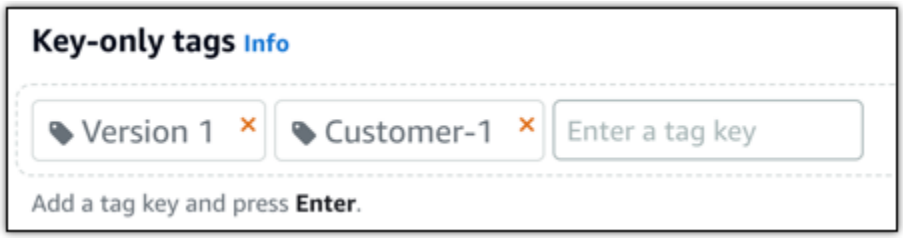
   Lightsail プランは、必要に応じて後でアップグレードできます。詳細については、「Lightsail でスナップショットからインスタンスを作成する」を参照してください。
6. インスタンスの名前を入力します。

リソース名:

- AWS リージョン Lightsail アカウントの各 内で一意である必要があります。
- 2～255 文字を使用する必要があります。
- 先頭と末尾は英数字にする必要があります。
- 英数字、ピリオド、ダッシュ、アンダースコアを含めることができます。

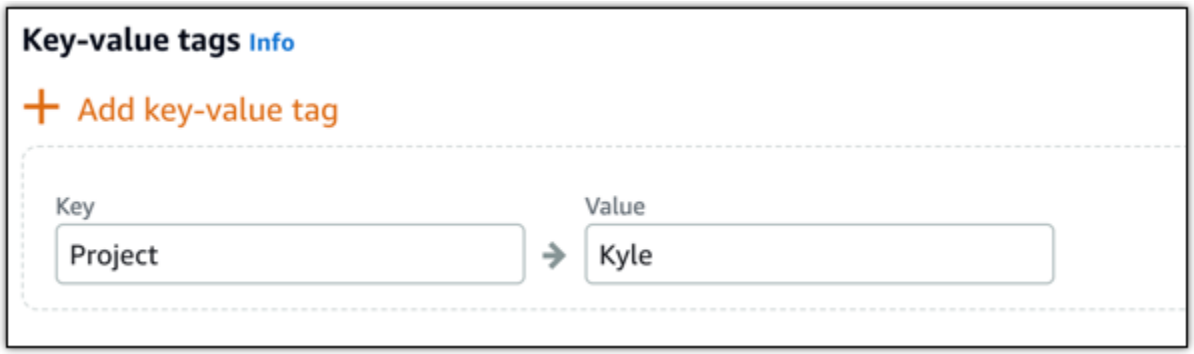
7. 以下のいずれかのオプションを選択して、インスタンスにタグを追加します。

- [Add key-only tags] (キーのみのタグを追加) または [Edit key-only tags] (キーのみのタグを編集) (タグが追加済みの場合)を追加。タグキーのテキストボックスに新しいタグを入力し、Enter キーを押します。タグの入力を完了したら、[保存] を選択してタグを追加し、追加しない場合は、[キャンセル] を選択します。



- [key-value タグの作成] から [キー] テキストボックスにキーを入力し、[値] テキストボックスに値を入力します。タグの入力を完了したら、[保存] を選択し、追加しない場合は、[キャンセル] を選択します。

  キーバリューのタグは、保存する際に一つずつ追加することができます。さらに key-value タグを追加するには、以上のステップを繰り返します。



> **Note**
>
> 「キーのみ」のタグと「キーバリュー」のタグの詳細については、「タグ」を参照してください。

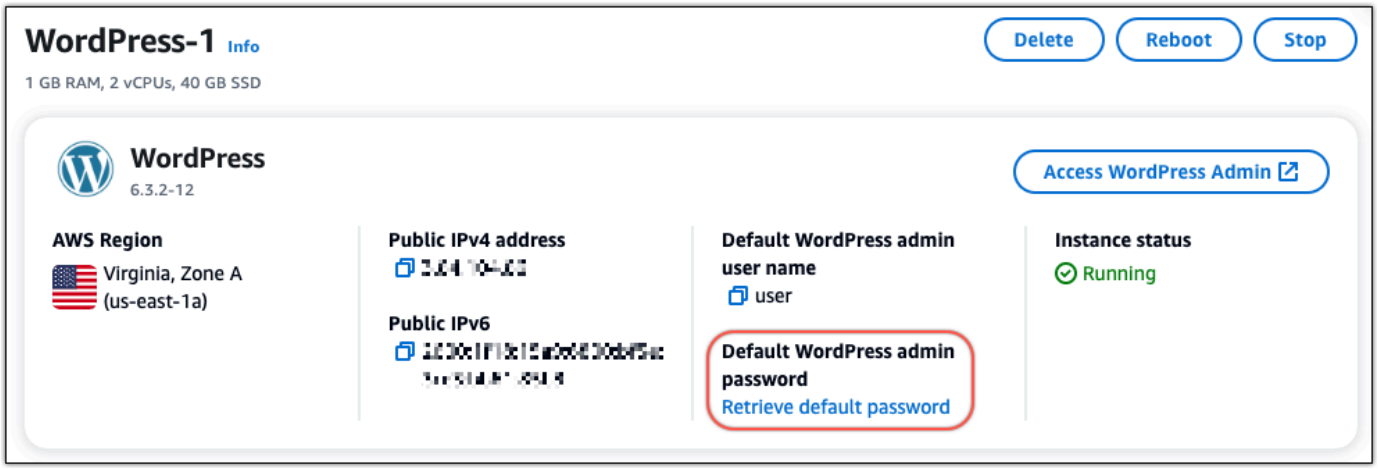
8. [インスタンスの作成] を選択します。

ステップ 3: 新しい Lightsail WordPress ブログにログインする

Lightsail に新しいブログを作成したので、 WordPress ダッシュボードにアクセスして古いブログデータをインポートする必要があります。 WordPress ウェブサイトの管理ダッシュボードにサインインするためのデフォルトのパスワードは、インスタンスに保存されます。パスワードを取得するには、次のステップを実行します。

WordPress 管理者のデフォルトパスワードを取得するには

1. インスタンスの WordPress インスタンス管理ページを開きます。
2. WordPress パネルで、デフォルトのパスワードの取得を選択します。これにより、ページの下部にある アクセスのデフォルトパスワードが展開されます。



3. 起動 CloudShellを選択します。これにより、ページの下部にパネルが開きます。
4. コピーを選択し、コンテンツをウィンドウに貼り付けます CloudShell 。 CloudShell プロンプトにカーソルを置き、Ctrl+V を押すか、右クリックしてメニューを開き、貼り付け を選択します。
5. CloudShell ウィンドウに表示されるパスワードを書き留めます。これは、 WordPress ウェブサイトの管理ダッシュボードにサインインするために必要です。

```
[cloudshell-user@ip-[redacted] ~]$ AWS_REGION=us-east-1 ~/lightsail_connect WordPress-1 cat bitnami_applic
ation_password
JKzh8wB5FAR!
```

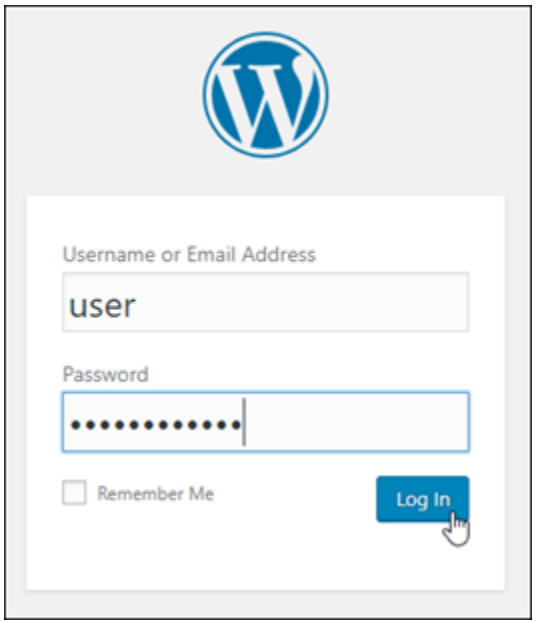
WordPress ウェブサイトの管理ダッシュボードのパスワードを取得したら、サインインできます。管理ダッシュボードでは、ユーザーパスワードの変更、プラグインのインストール、ウェブサイトのテーマの変更などを行うことができます。

ウェブサイトの管理ダッシュボードにサインインするには、次のステップを実行します WordPress。

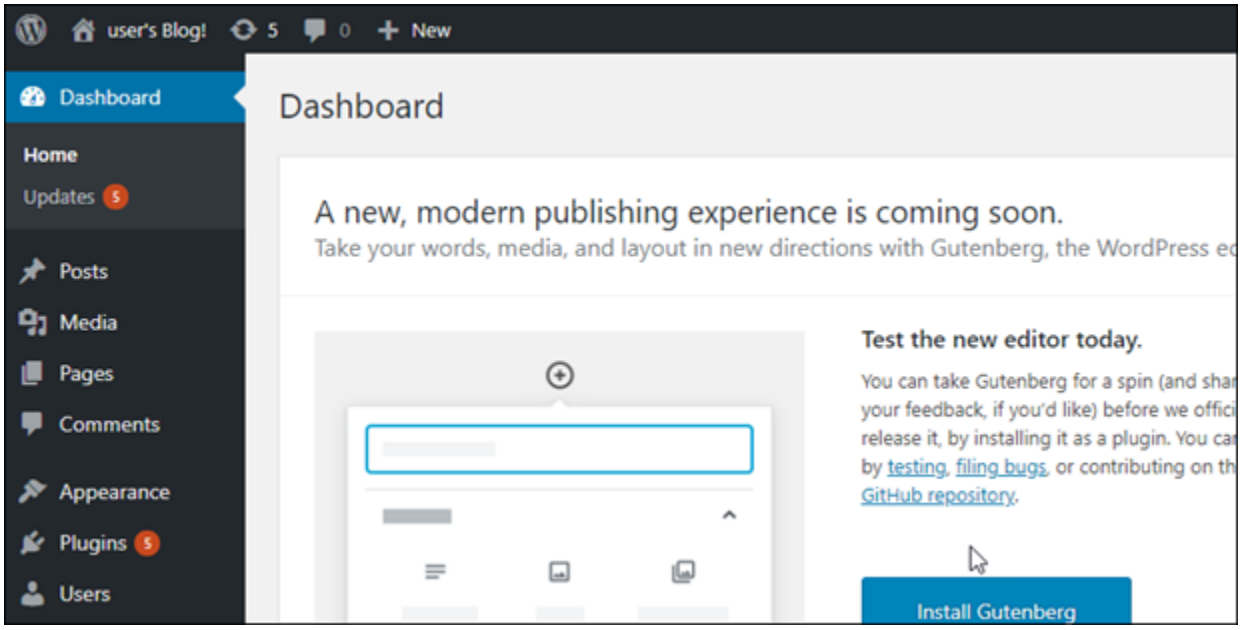
管理ダッシュボードにサインインするには

1. インスタンスの WordPress インスタンス管理ページを開きます。
2. WordPress パネルで、アクセス WordPress 管理者 を選択します。
3. WordPress 管理者ダッシュボードへのアクセスパネルの「パブリック IP アドレスを使用する」で、次の形式のリンクを選択します。

   http://*public-ipv4-address* ./wp-admin
4. ユーザー名 または E メールアドレス には、 と入力します**user**。
5. パスワード には、前のステップで取得したパスワードを入力します。
6. [ログイン] を選択します。



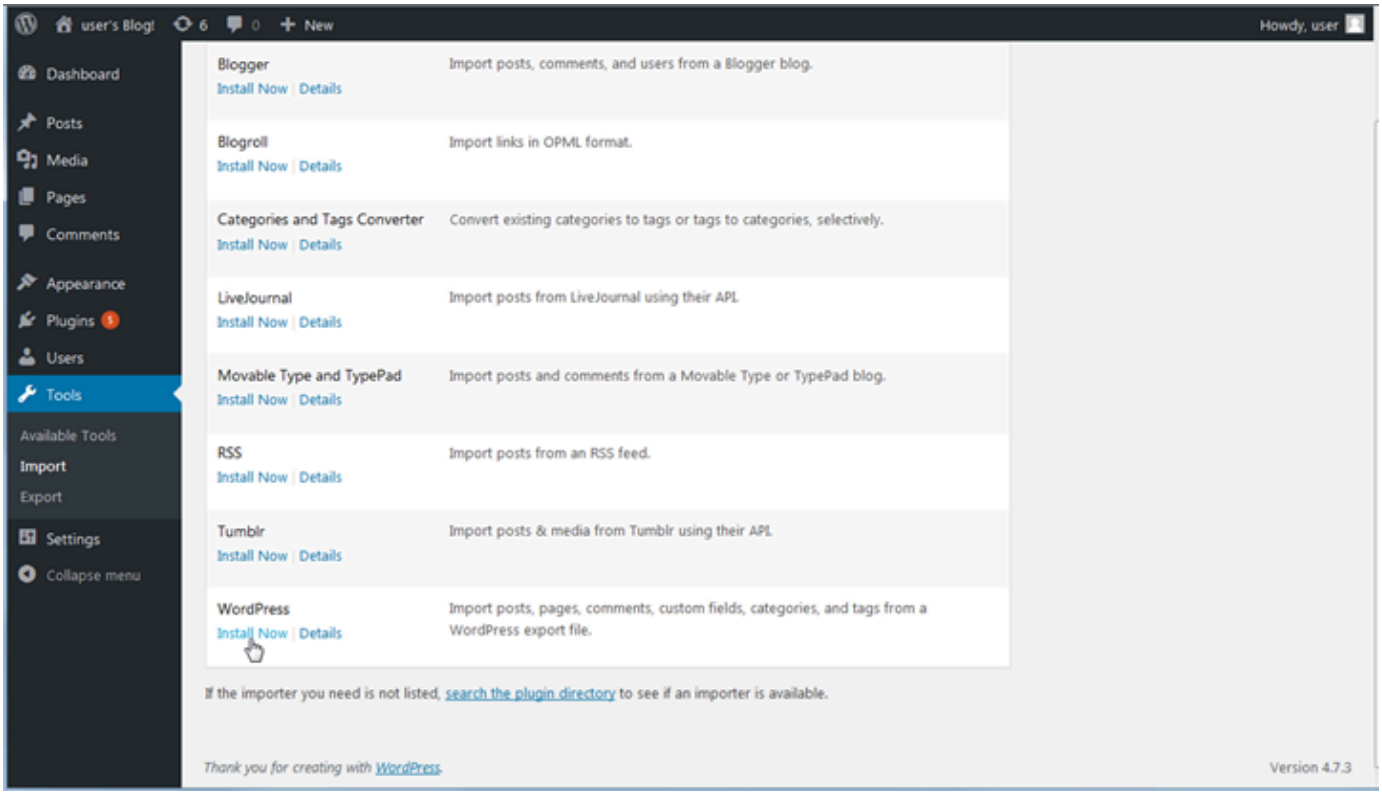
これで、管理アクションを実行できる WordPress ウェブサイトの管理ダッシュボードにサインインしました。 WordPress ウェブサイトの管理の詳細については、 WordPress ドキュメントのWordPress「Codex」を参照してください。



ステップ 4: XML ファイルを新しい Lightsail ブログにインポートする

新しい Lightsail インスタンスの WordPress ダッシュボードに正常にログインしたら、次のステップに従って XML ファイルを新しい Lightsail ブログにインポートします。

1. 新しい Lightsail インスタンスの WordPress ダッシュボードから、ツール を選択します。
2. インポート を選択し、今すぐインストール を選択して WordPress インポートツールをインストールします。



3. ツールのインストールが完了したら、[インポーターの実行] を選択してインポートツールを実行します。
4. インポート WordPressページで、参照を選択します。
5. ステップ 1: 既存の WordPressブログ をバックアップ で保存した XML ファイルを見つけ、 を開く を選択します。
6. [ファイルをアップロードしてインポート] を選択します。

   残りはデフォルトのままにして、[送信] を選択します。

次のステップ

ブログ (ホームアイコンの横) を選択し、 WordPress ダッシュボードからサイトにアクセスを選択することで、すべてが機能していることを確認できます。ブラウザに IP アドレスを入力してブログを表示することもできます。

次のステップを以下に示します。

- ドメインネームサーバーが新しいバージョンのブログを指すように、DNS を移行します。
- 新しいブログの外観をカスタマイズしたり、プラグインをインストールしたりします WordPress 。

- SSL 証明書による HTTPS サポートの有効化

Amazon Lightsail の WordPress マルチサイトチュートリアル

WordPress Multisite を使用すると、管理者は同じ WordPress インスタンスから複数のウェブサイトをホストして管理できます。以下のチュートリアルを使用して、Lightsail の WordPress マルチサイトをで作業する方法を説明します。

トピック

- Lightsail の WordPress Multisite インスタンスにブログをドメインとして追加する
- Lightsail の WordPress Multisite インスタンスにブログをサブドメインとして追加する
- Lightsail で WordPress Multisite インスタンスのプライマリドメインを定義する

Lightsail の WordPress Multisite インスタンスにブログをドメインとして追加する

Amazon Lightsail の WordPress Multisite インスタンスは、インスタンス内で作成するブログサイトごとに、複数のドメインまたはサブドメインを使用するように設計されています。このガイドでは、WordPress Multisite インスタンスでメインブログのプライマリドメインとは異なるドメインを使用してブログサイトを追加する方法を示します。たとえば、メインブログのプライマリドメインが `example.com` である場合、同じインスタンスで `another-example.com` ドメインや `third-example.com` ドメインを使用する新しいブログサイトを作成できます。

> **Note**
>
> また、サブドメインを使用するサイトを WordPress Multisite インスタンスに追加することもできます。詳細については、「WordPress Multisite インスタンスにブログをサブドメインとして追加する」を参照してください。

前提条件

次の前提条件を以下に示す順に実行してください。

1. Lightsail で WordPress Multisite インスタンスを作成します。詳細については、「インスタンスを作成する」を参照してください。

2. 静的 IP を作成して Lightsail の WordPress Multisite インスタンスにアタッチします。詳細については、「静的 IP を作成してインスタンスにアタッチする」を参照してください。
3. DNS ゾーンを作成してドメインを Lightsail に追加し、このドメインが WordPress Multisite インスタンスにアタッチした静的 IP をポイントするように設定します。詳細については、「DNS ゾーンを作成してドメインの DNS レコードを管理する」を参照してください。
4. WordPress Multisite インスタンスのプライマリドメインを定義する。詳細については、「WordPress Multisite インスタンスのプライマリドメインを定義する」を参照してください。

WordPress Multisite インスタンスにブログをドメインとして追加する

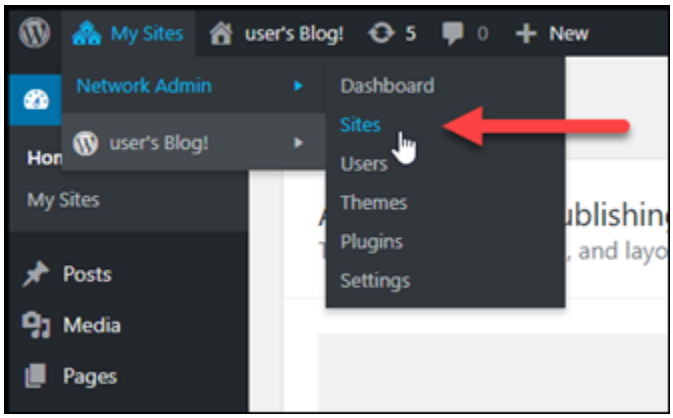
以下の手順を実行し、メインブログのプライマリドメインとは異なるドメインを使用するブログサイトを WordPress Multisite インスタンスで作成します。

> **Important**
>
> 次の手順を実行する前に、このガイドの前提条件のセクションに記載されているステップ 4 を完了する必要があります。

1. WordPress Multisite インスタンスの管理ダッシュボードにサインインします。

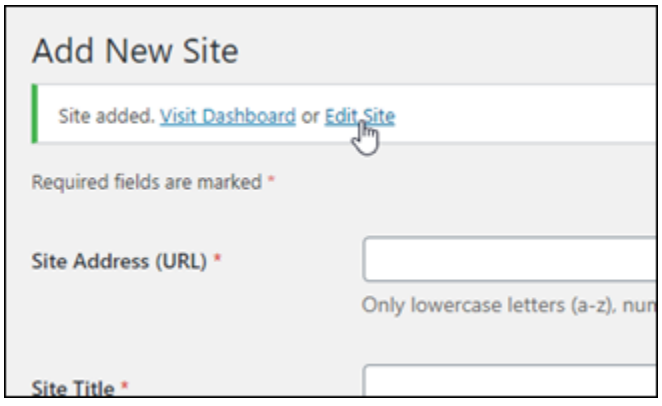
   > **Note**
   >
   > 詳細については、「Bitnami インスタンス向けにアプリケーションのユーザー名とパスワードを取得する」を参照してください。

2. 上部のナビゲーションペインで [My Sites] (自分のサイト)、[Network Admin] (ネットワーク管理者)、[Sites] (サイト) の順に選択します。

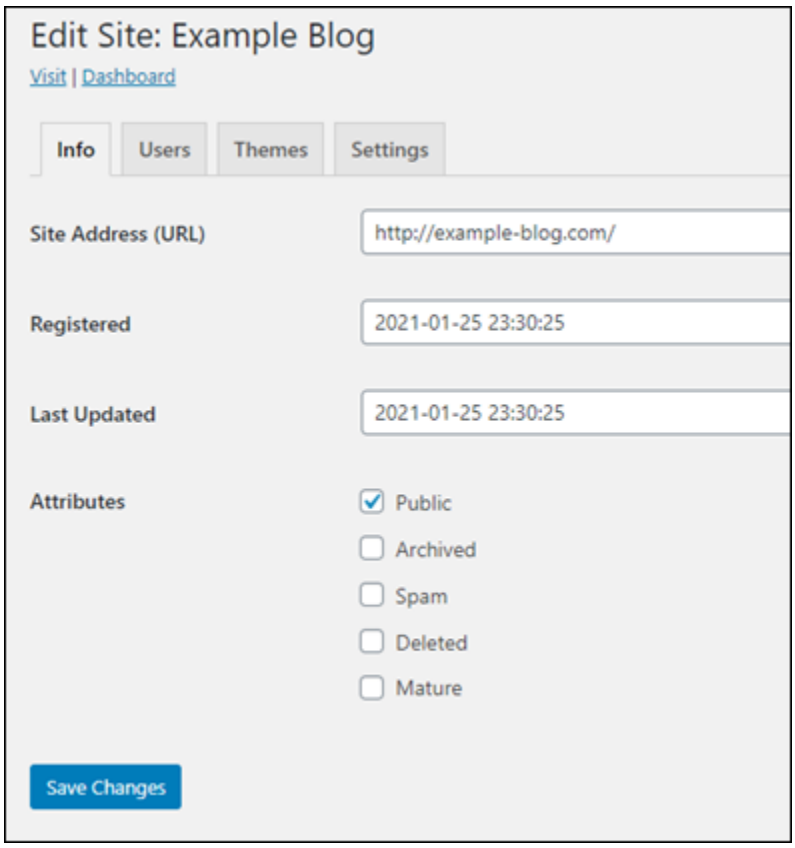


3. [Add New] (新規追加) を選択して新しいブログサイトを追加します。
4. サイトのアドレスをサイトアドレス (URL)テキストボックスに入力します。こちらが新しいブログサイトに使われるドメインになります。例えば、新しいブログサイトで `example-blog.com` をドメインとして使用する場合は、サイトアドレス (URL) テキストボックスに`example-blog`と入力します。ページに表示されるプライマリドメインのサフィックスは無視します。

5. サイトのタイトルを入力し、サイトの言語を選択して、管理者の E メールアドレスを入力します。
6. [Add Site] (サイトの追加) を選択します。
7. ページに表示させる確認バナーでサイトの編集を選択します。最近作成したサイトの詳細編集にリダイレクトされます。

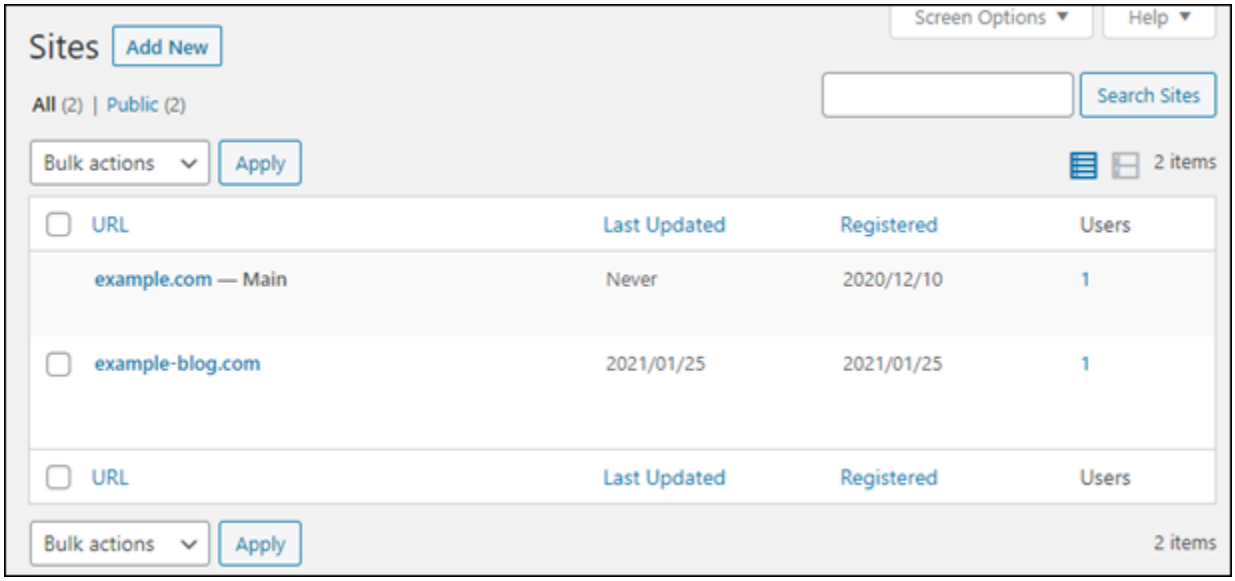


8. サイトの編集ページ上で、サイトアドレス (URL)テキストボックスにリストされているサブドメインを使用したい apex ドメインに変更します。この例では、`http://example-blog.com`を指定しました。



9. [Save Changes] (変更を保存する) を選択します。

この時点で、新しいブログサイトは WordPress Multisite インスタンスに作成されましたが、ドメインは新しいブログサイトにルーティングされるようにまだ設定されていません。次のステップに進み、アドレスレコード (A レコード) をドメインの DNS ゾーンに追加します。



アドレスレコード (A レコード) をドメインの DNS ゾーンに追加する

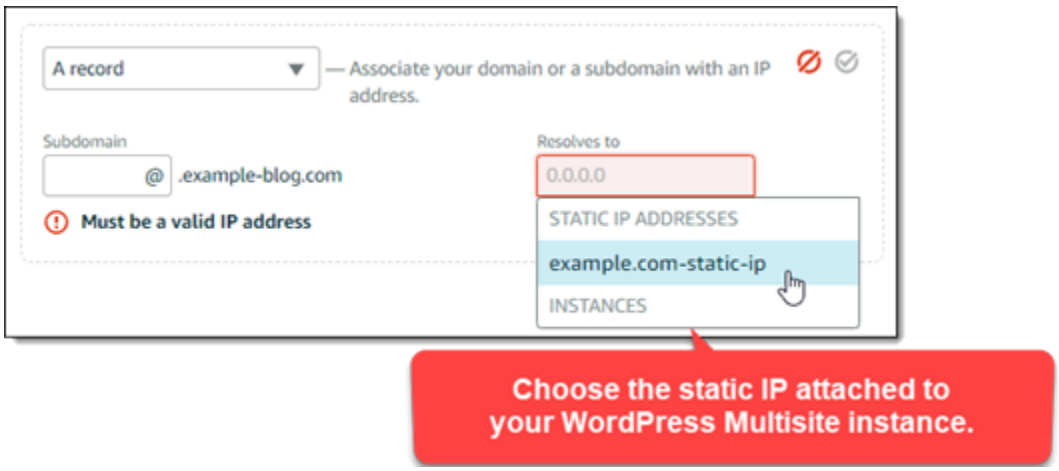
以下の手順を実行し、新しいブログサイトのドメインが WordPress Multisite インスタンスをポイントするように設定します。以下の手順は、WordPress Multisite インスタンスで作成するブログサイトごとに実行する必要があります。

デモの目的で、Lightsail の DNS ゾーンを使用します。ただし、ドメインレジストラがホストする他の一般的な DNS ゾーンでも手順は同様です。

> **Important**
>
> Lightsail コンソールでは、最大 6 つの DNS ゾーンを作成できます。さらに DNS ゾーンを増やす場合は、Amazon Route 53 を使用してドメインの DNS レコードを管理することをお勧めします。詳細については、「Amazon Route 53 を既存ドメインの DNS サービスにする」を参照してください。

1. Lightsail コンソールにサインインします。
2. Lightsail のホームページで [Domains & DNS] (ドメインと DNS) タブを選択します。
3. ページの [DNS ゾーン] セクションで、新しいブログサイトのドメインの DNS ゾーンを選択します。
4. DNS ゾーンエディタで [DNS records] (DNS レコード) タブを選択します。次に、[Add record] (レコードの追加) を選択します。

5. レコードタイプのドロップダウンメニューで [A レコード] を選択します。
6. [Record name] (レコード名) テキストボックスに、「at」記号 (@) を入力し、ドメインのルートのレコードを作成します。
7. [Resolves to] (解決先) テキストボックスで、WordPress Multisite インスタンスにアタッチされている静的 IP アドレスを選択します。



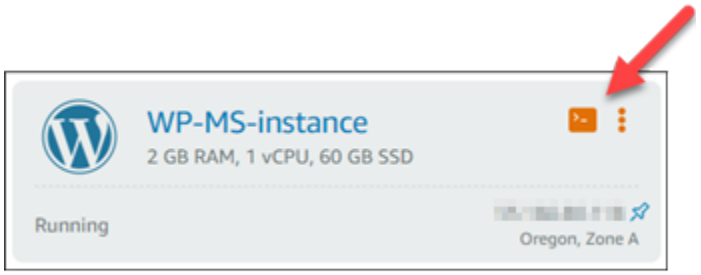
8. 保存アイコンを選択します。

   変更がインターネットの DNS を通じて伝播されると、ドメインは WordPress Multisite インスタンスの新しいブログサイトにルートされます。

Cookie Support を有効にして、ブログサイトへのサインインを許可する

ブログサイトをドメインとして WordPress Multisite インスタンスに追加する場合は、cookie support を有効化するために、インスタンスにある WordPress の設定 (`wp-config`) ファイルをアップデートする必要があります。Cookie support を有効にしない場合、ユーザーがブログサイトの WordPress 管理ダッシュボードにサインインしようとすると、「エラー: Cookie がブロックされているかサポートされていません」とエラーが表示されることがあります。

1. Lightsail コンソールにサインインします。
2. Lightsail のホームページで、WordPress Multisite インスタンスの SSH クイック接続アイコンを選択します。



3. Lightsail ブラウザベースの SSH セッションに接続後、以下のコマンドを入力して Vim を利用し、インスタンスの`wp-config.php`ファイルを開けて編集します。

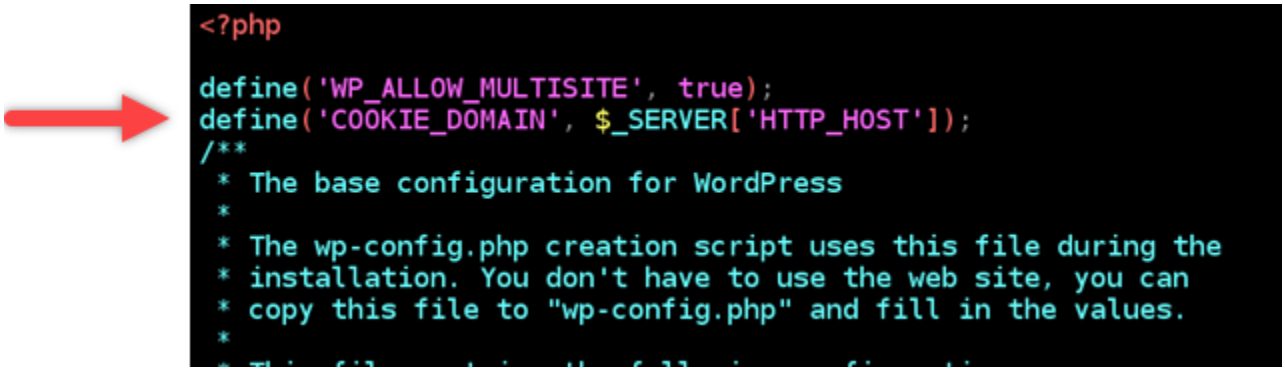
```
sudo vim /opt/bitnami/wordpress/wp-config.php
```

> **Note**
>
> このコマンドが失敗した場合は、古いバージョンの WordPress Multisite インスタンスを使用している可能性があります。代わりに次のコマンドを実行してみてください。
>
> ```
> sudo vim /opt/bitnami/wordpress/wp-config.php
> ```

4. I を押して Vim モード挿入を入力します。
5. 以下のテキストの行を `define('WP_ALLOW_MULTISITE', true);` テキストの行の下に追加します。

```
define('COOKIE_DOMAIN', $_SERVER['HTTP_HOST']);
```

完了すると、ファイルは次のようになります。



6. ESC キーを押して Vim モード挿入を終了後、`:wq!` を入力して Enter を押して編集 (書き込み) を保存して Vim を終了します。
7. 次のコマンドを入力して、WordPress インスタンス上の基本サービスを再起動します。

```
sudo /opt/bitnami/ctlscript.sh restart
```

これで、WordPress Multisite インスタンスで cookie が有効化されて、ブログサイトにサインインしようとしているユーザーに「エラー:Cookie がブロックされているかサポートされていません」というエラーが発生しなくなります。

次のステップ

WordPress Multisite インスタンスにブログをドメインとして追加した後、WordPress Multisite の管理に慣れることをお勧めします。詳細については 、WordPress ドキュメントにある Multisite ネットワーク管理 を参照してください。

Lightsail の WordPress Multisite インスタンスにブログをサブドメインとして追加する

Amazon Lightsail の WordPress Multisite インスタンスは、インスタンス内で作成するブログサイトごとに、複数のドメインまたはサブドメインを使用するように設計されています。このガイドでは、WordPress Multisite インスタンスのサブドメインとしてブログサイトを追加する方法について説明します。たとえば、メインブログのプライマリドメインが `example.com` である場合、同じインスタンスで `earth.example.com` サブドメインや `moon.example.com` サブドメインを使用する新しいブログサイトを作成できます。

> **Note**
>
> また、ドメインを使用するサイトを WordPress Multisite インスタンスに追加することもできます。詳細については、「WordPress Multisite インスタンスにブログをドメインとして追加する」を参照してください。

前提条件

次の前提条件を以下に示す順に実行してください。

1. WordPress Multisite インスタンスを作成する。詳細については、「インスタンスを作成する」を参照してください。
2. 静的 IP を作成して WordPress Multisite インスタンスにアタッチします。詳細については、「静的 IP を作成してインスタンスにアタッチする」を参照してください。
3. DNS ゾーンを作成してドメインを Lightsail に追加し、このドメインが WordPress Multisite インスタンスにアタッチした静的 IP をポイントするように設定します。詳細については、「DNS ゾーンを作成してドメインの DNS レコードを管理する」を参照してください。
4. WordPress Multisite インスタンスのプライマリドメインを定義する。詳細については、「WordPress Multisite インスタンスのプライマリドメインを定義する」を参照してください。

WordPress Multisite インスタンスにブログをサブドメインとして追加する

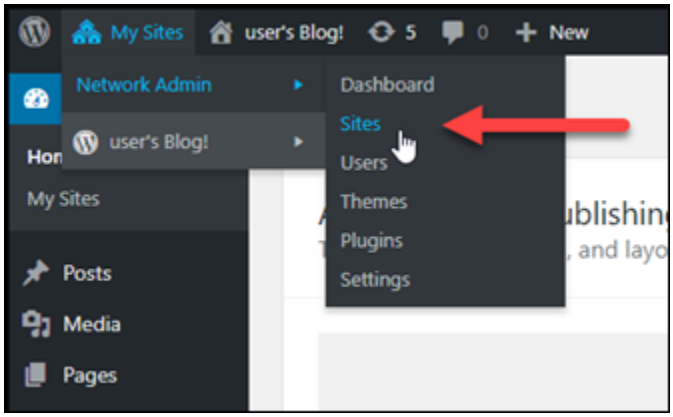
以下の手順を実行し、メインブログのプライマリドメインのサブドメインを使用する新しいブログを WordPress Multisite インスタンスで作成します。

> **Important**
>
> 次の手順を実行する前に、このガイドの前提条件のセクションに記載されているステップ 4 を完了する必要があります。

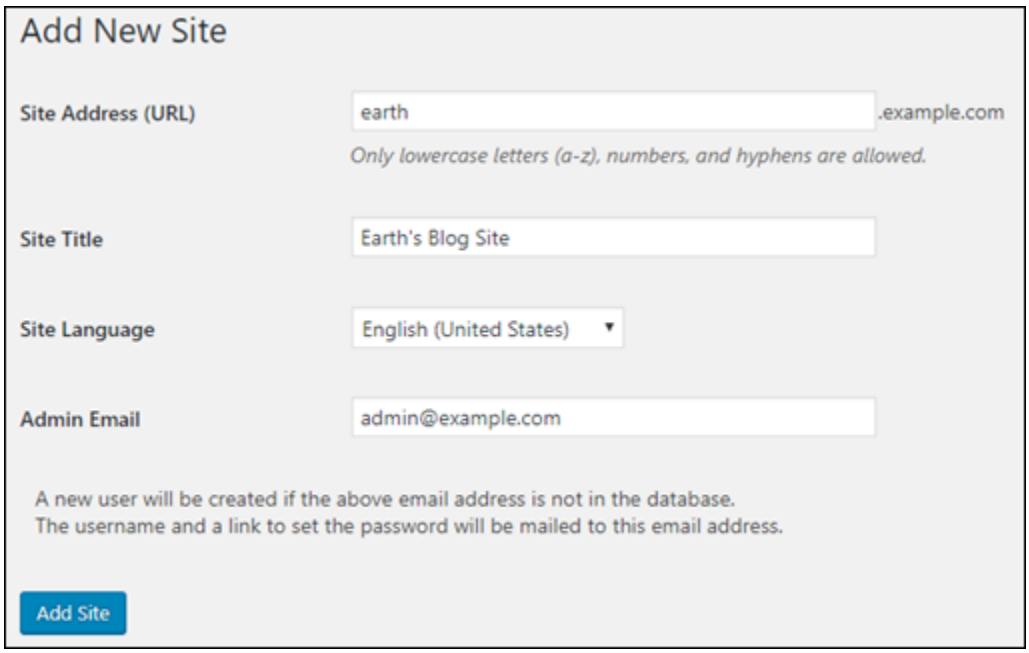
1. WordPress Multisite インスタンスの管理ダッシュボードにサインインします。

   > **Note**
   >
   > 詳細については、「Bitnami インスタンス向けにアプリケーションのユーザー名とパスワードを取得する」を参照してください。

2. 上部のナビゲーションペインで [My Sites] (自分のサイト)、[Network Admin] (ネットワーク管理者)、[Sites] (サイト) の順に選択します。

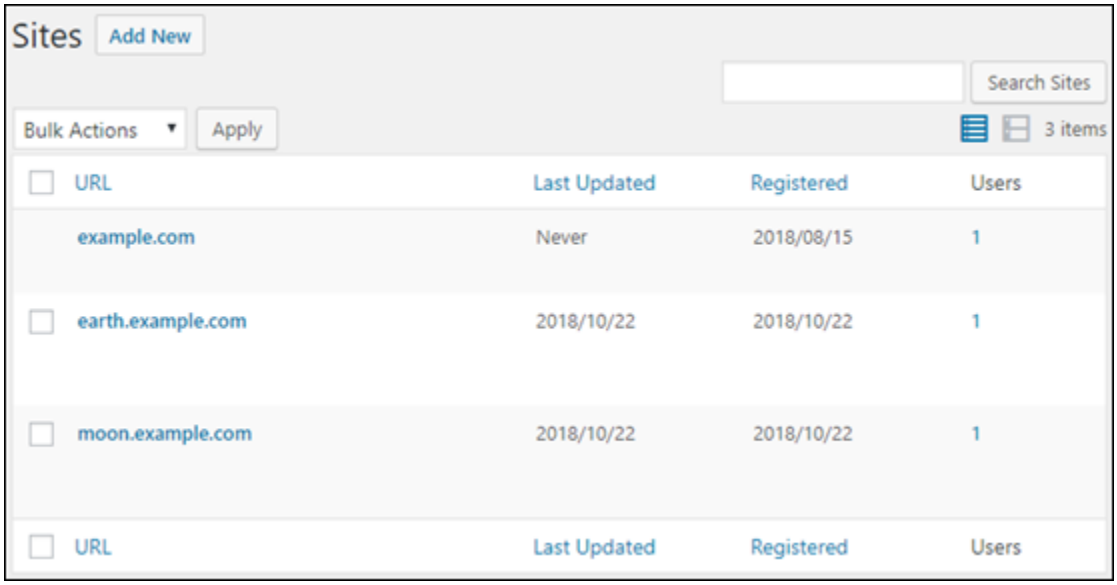


3. [Add New] (新規追加) を選択して新しいブログサイトを追加します。
4. 新しいブログサイトのサブドメインとして使用するサイトアドレスを入力します。



5. サイトのタイトルを入力し、サイトの言語を選択して、管理者の E メールアドレスを入力します。
6. [Add Site] (サイトの追加) を選択します。

この時点で、新しいブログサイトは WordPress Multisite インスタンスに作成されましたが、サブドメインは新しいブログサイトにルーティングされるようにまだ設定されていません。次のステップに進み、アドレスレコード (A レコード) をドメインの DNS ゾーンに追加します。

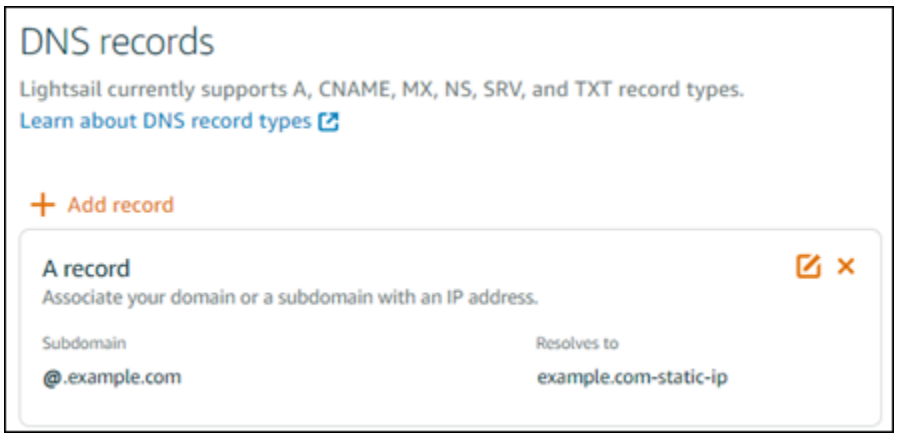


アドレスレコード (A レコード) をドメインの DNS ゾーンに追加する

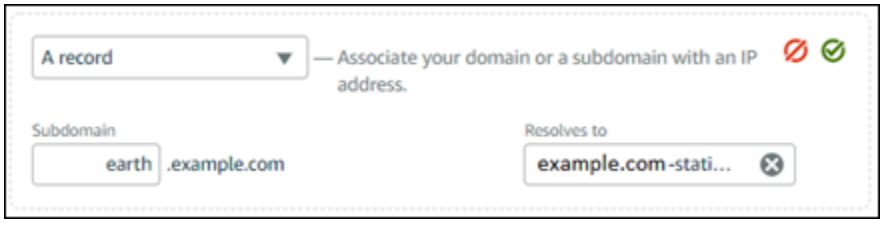
以下の手順を実行し、新しいブログサイトのサブドメインが WordPress Multisite インスタンスをポイントするように設定します。以下の手順は、WordPress Multisite インスタンスで作成するブログサイトごとに実行する必要があります。

デモの目的で、Lightsail の DNS ゾーンを使用します。ただし、ドメインレジストラがホストする他の一般的な DNS ゾーンでも手順は同様です。

1. Lightsail コンソールにサインインします。
2. Lightsail のホームページで [Domains & DNS] (ドメインと DNS) タブを選択します。
3. ページの [DNS ゾーン] セクションで、WordPress Multisite インスタンスのプライマリドメインとして定義したドメインの DNS ゾーンを選択します。
4. DNS ゾーンエディタで [DNS records] (DNS レコード) タブを選択します。次に、[Add record] (レコードの追加) を選択します。



5. レコードタイプのドロップダウンメニューで [A レコード] を選択します。
6. [Record name] (レコード名) テキストボックスに、WordPress Multisite インスタンスで新しいブログサイトを作成したときにサイトアドレスとして指定したサブドメインを入力します。
7. [Resolves to] (解決先) テキストボックスで、WordPress Multisite インスタンスにアタッチされている静的 IP アドレスを選択します。



8. 保存アイコンを選択します。

必要な操作は以上のみです。変更がインターネットの DNS を通じて伝播されると、ドメインは WordPress Multisite インスタンスの新しいブログサイトにリダイレクトされます。

次のステップ

WordPress Multisite インスタンスにブログをサブドメインとして追加した後は、WordPress Multisite の管理に慣れることをお勧めします。詳細については、WordPress ドキュメントにあるマルチサイトネットワーク管理を参照してください。

Lightsail で WordPress Multisite インスタンスのプライマリドメインを定義する

Amazon Lightsail の WordPress Multisite インスタンスは、インスタンス内で作成するブログサイトごとに、複数のドメインまたはサブドメインを使用するように設計されています。このため、WordPress Multisite インスタンスのメインブログで使用するプライマリドメインを定義する必要があります。

前提条件

次の前提条件を以下に示す順に実行してください。

1. Lightsail で WordPress Multisite インスタンスを作成します。詳細については、「インスタンスを作成する」を参照してください。
2. 静的 IP を作成して Lightsail の WordPress Multisite インスタンスにアタッチします。詳細については、「静的 IP を作成してインスタンスにアタッチする」を参照してください。

> **Important**
>
> WordPress マルチサイトインスタンスを再起動するには、静的 IP をアタッチした後に再起動する必要があります。これにより、インスタンスは、それに関連付けられた新しい静的 IP を認識できるようになります。

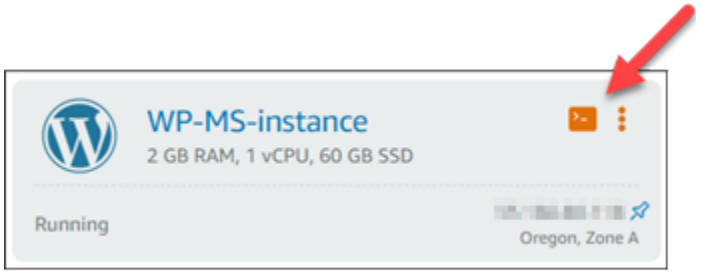
3. DNS ゾーンを作成してドメインを Lightsail に追加し、このドメインが WordPress Multisite インスタンスにアタッチした静的 IP をポイントするように設定します。詳細については、「DNS ゾーンを作成してドメインの DNS レコードを管理する」を参照してください。

4. DNS の変更がインターネットの DNS を通じて伝播されるまで待ちます。その後、このガイドの「WordPress マルチサイトインスタンスのプライマリドメインを定義する」セクションに進みます。

WordPress Multisite インスタンスのプライマリドメインを定義する

以下のステップを実行し、ドメイン (example.com など) が WordPress Multisite インスタンスのメインブログにリダイレクトされるようにします。

1. Lightsail コンソールにサインインします。
2. Lightsail のホームページで、WordPress Multisite インスタンスの SSH クイック接続アイコンを選択します。



3. 次のコマンドを入力して、WordPress Multisite インスタンスのプライマリドメイン名を定義します。*<domain>* は WordPress Multisite の正しいドメイン名に置き換えてください。

```
sudo /opt/bitnami/configure_app_domain --domain <domain>
```

例:

```
sudo /opt/bitnami/configure_app_domain --domain example.com
```

> **Note**
>
> このコマンドが失敗した場合は、古いバージョンの WordPress Multisite インスタンスを使用している可能性があります。*<domain>* を WordPress Multisite の正しいドメイン名で置き換えて以下のコマンドを実行してください。
>
> ```
> cd /opt/bitnami/apps/wordpress
> sudo ./bnconfig --machine_hostname <domain>
> ```

コマンドの実行が終了したら次のコマンドを入力し、サーバーが再起動するたびに bnconfig ツールが自動的に実行されないようにします。

```
sudo mv bnconfig bnconfig.disabled
```

この時点で、定義したドメインを参照すると、WordPress Multisite インスタンスのメインブログにリダイレクトされます。

次のステップ

WordPress Multisite インスタンスのプライマリドメインを定義したら、以下のステップを実行します。

- WordPress Multisite インスタンスにブログをサブドメインとして追加する
- WordPress Multisite インスタンスにブログをドメインとして追加する

Amazon Lightsail の Let's Encrypt チュートリアル

Let's Encrypt は無料の SSL/TLS 証明書を発行し、ウェブサイト、アプリケーション、オンラインサービスの安全で暗号化された通信を可能にします。以下のチュートリアルで、Lightsail で Let's Encrypt を使用する方法を紹介します。

トピック

- チュートリアル: Lightsail の LAMP インスタンスで Let's Encrypt の SSL 証明書を使用する
- チュートリアル: Lightsail の Nginx インスタンスで Let's Encrypt の SSL 証明書を使用する
- チュートリアル: WordPress Lightsail インスタンスで SSL 証明書を暗号化しよう

チュートリアル: Lightsail の LAMP インスタンスで Let's Encrypt の SSL 証明書を使用する

Amazon Lightsail では、Lightsail ロードバランサーを使用すると、SSL/TLS でウェブサイトとアプリケーションのセキュリティを簡単に強化できます。ただし、Lightsail ロードバランサーの使用は一般的に最適な選択肢ではない場合があります。ロードバランサーが提供するスケーラビリティや耐障

害性がサイトでは必要ない場合や、コストを最適化するためにロードバランサーを使用しない場合があります。

後者の場合は、Let's Encrypt で無料の SSL 証明書を入手できます。無料の証明書を使用することに問題はありません。これらの証明書は Lightsail インスタンスに統合できます。このチュートリアルでは、Certbot を使用して Let's Encrypt ワイルドカード証明書をリクエストし、これを LAMP インスタンスに統合する方法を示します。

> **Important**
>
> - Bitnami インスタンスで使用されている Linux ディストリビューションは、2020 年 7 月に Ubuntu から Debian に変更されました。この変更により、このチュートリアルのいくつかのステップは、インスタンスの Linux ディストリビューションによって異なります。変更後に作成された Bitnami ブループリントインスタンスはすべて Debian Linux ディストリビューションを使用します。変更前に作成されたインスタンスは、Ubuntu Linux ディストリビューションを引き続き使用します。インスタンスのディストリビューションをチェックするには、`uname -a` コマンドを実行します。応答には、インスタンスの Linux ディストリビューションとして Ubuntu または Debian のいずれかが表示されます。
> - Bitnami は、多くのスタックのファイル構造を変更するプロセスです。このチュートリアルのファイルパスは、Bitnami スタックがネイティブ Linux システムパッケージを使用しているか (アプローチ A)、または自己完結型インストール (アプローチ B) であるかによって、変更される場合があります。Bitnami のインストールタイプと取るべき方法を特定するには、次のコマンドを実行します。
>
>   ```
>   test ! -f "/opt/bitnami/common/bin/openssl" && echo "Approach
>   A: Using system packages." || echo "Approach B: Self-contained
>   installation."
>   ```

目次

- ステップ 6: Let's Encrypt の SSL 証明書リクエストを完了する
- ステップ 7: Apache サーバーディレクトリで Let's Encrypt の証明書ファイルへのリンクを作成する
- ステップ 8: ウェブアプリケーションの HTTP から HTTPS へのリダイレクトを設定する
- ステップ 9: Let's Encrypt 証明書を 90 日ごとに更新する

ステップ 1: 前提条件を満たす

以下の前提条件を満たします (まだ満たしていない場合)。

- Lightsail で LAMP インスタンスを作成します。詳細については、「インスタンスを作成する」を参照してください。
- ドメイン名を登録し、その DNS レコードを編集するための管理アクセスを取得します。詳細については、「Amazon Lightsail の DNS」を参照してください。

> **Note**
>
> ドメインの DNS レコードは、Lightsail の DNS ゾーンを使用して管理することをお勧めします。詳細については、「 DNS ゾーンを作成し、ドメインの DNS レコードを管理する」を参照してください。

- Lightsail コンソールでブラウザベースの SSH ターミナルを使用して、このチュートリアルのステップを実行します。ただし、独自の SSH クライアント (PuTTY など) を使用することもできます。PuTTY の設定の詳細については、「PuTTY をダウンロード、SSH を使用して接続するようにセットアップする」を参照してください。

前提条件が完了したら、このチュートリアルの「次のセクション」に進みます。

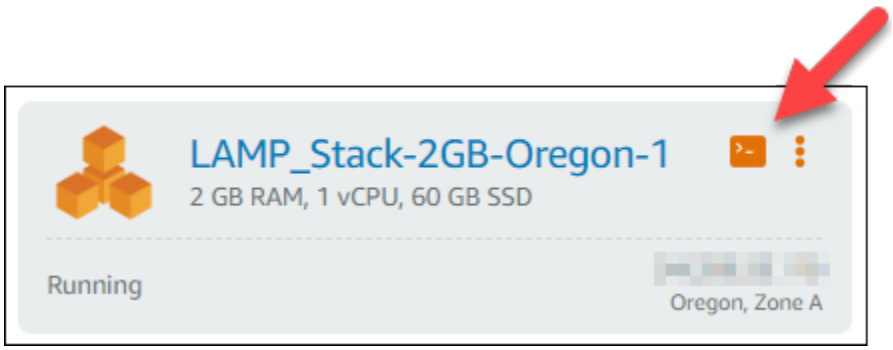
ステップ 2: インスタンスに Certbot をインストールする

Certbot は、Let's Encrypt の証明書をリクエストしてウェブサーバーにデプロイするために使用するクライアントです。Let's Encrypt は ACME プロトコルを使用して証明書を発行します。Certbot は、Let's Encrypt とやり取りする ACME 対応のクライアントです。

Lightsail インスタンスに Certbot をインストールするには

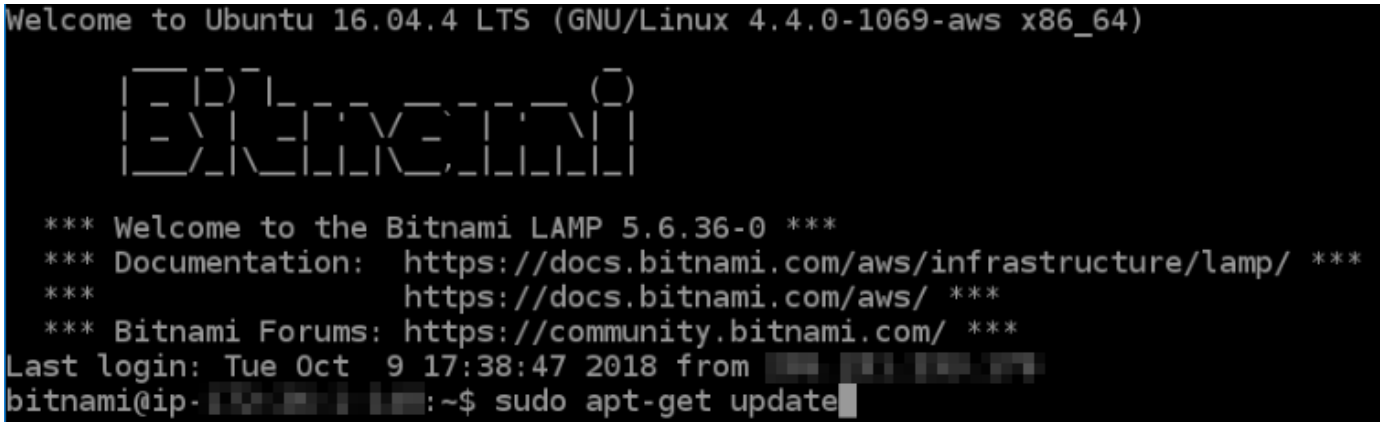
1. Lightsail コンソールにサインインします。

2. Lightsail のホームページで、接続先のインスタンスの SSH クイック接続アイコンを選択します。



3. Lightsail のブラウザベースの SSH セッションに接続したら、次のコマンドを入力してインスタンスのパッケージを更新します。

```
sudo apt-get update
```



4. 次のコマンドを入力してソフトウェアプロパティパッケージをインストールします。Certbot の開発者は、Personal Package Archive (PPA) を使用して Cerbot を配信します。ソフトウェアプロパティパッケージを使用すると、PPA をより効率的に操作できます。

```
sudo apt-get install software-properties-common
```

> **Note**
>
> `sudo apt-get install` コマンドを実行したときに `Could not get lock` エラーが発生した場合は、約 15 分待ってから再試行してください。このエラーは、自動アップグレードをインストールするために Apt パッケージ管理ツールを使用している cron ジョブが原因で発生している可能性があります。

5. 次のコマンドを入力して Certbot をローカル apt リポジトリに追加します。

> **Note**
>
> ステップ 5 は、Ubuntu Linux ディストリビューションを使用するインスタンスにのみ適用されます。インスタンスが Debian Linux ディストリビューションを使用している場合は、このステップをスキップしてください。

```
sudo apt-add-repository ppa:certbot/certbot -y
```

6. 次のコマンドを入力して apt を更新し、新しいリポジトリを含めます。

```
sudo apt-get update -y
```

7. 次のコマンドを入力して Cerbot をインストールします。

```
sudo apt-get install certbot -y
```

これで Lightsail インスタンスに Cerbot がインストールされました。

8. ブラウザベースの SSH ターミナルウィンドウは開いたままにします。このチュートリアルで後ほど戻ります。このチュートリアルの「次のセクション」に進みます。

ステップ 3: Let's Encrypt の SSL ワイルドカード証明書をリクエストする

Let's Encrypt の証明書をリクエストするプロセスを開始します。Certbot を使用してワイルドカード証明書をリクエストします。この 1 つの証明書をドメインとそのサブドメインの両方に使用できます。たとえば、1 つのワイルドカード証明書を `example.com` 最上位ドメイン、`blog.example.com` サブドメイン、および `stuff.example.com` サブドメインに使用できます。

Let's Encrypt の SSL ワイルドカード証明書をリクエストするには

1. このチュートリアルのステップ 2 で使用した同じブラウザベースの SSH ターミナルウィンドウで、以下のコマンドを入力してドメインの環境変数を設定します。より効率的にコマンドをコピーして貼り付け、証明書を取得できます。

```
DOMAIN=Domain
```

```
WILDCARD=*.$DOMAIN
```

コマンドで、*Domain* を登録済みのドメイン名に置き換えます。

例:

```
DOMAIN=example.com
```

```
WILDCARD=*.$DOMAIN
```

2. 次のコマンドを入力し、変数が正しい値を返すことを確認します。

```
echo $DOMAIN && echo $WILDCARD
```

次のような結果が表示されます。

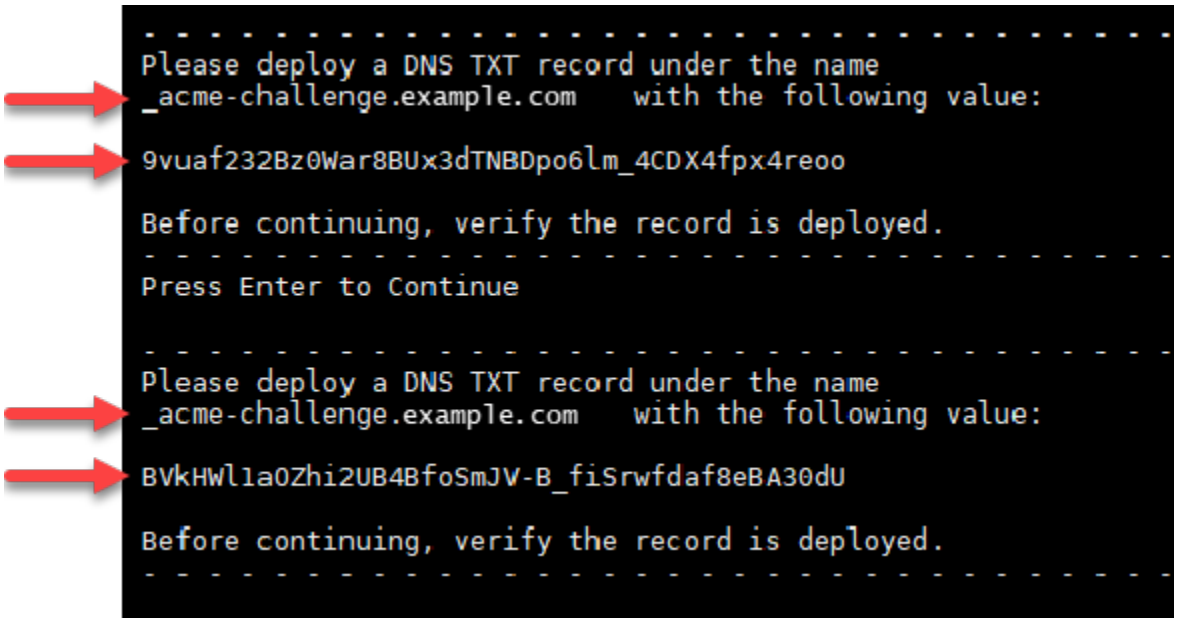
3. 次のコマンドを入力して Certbot をインタラクティブモードで起動します。このコマンドでは、DNS チャレンジで手動認証を使用してドメインの所有権を検証することを Certbot に指示します。また、最上位ドメインとそのサブドメイン用にワイルドカード証明書をリクエストします。

```
sudo certbot -d $DOMAIN -d $WILDCARD --manual --preferred-challenges dns certonly
```

4. プロンプトに応じて E メールアドレスを入力します。これで更新とセキュリティに関する通知を受信します。
5. Let's Encrypt のサービス利用規約を読みます。読み終わり、同意する場合は A キーを押します。同意しない場合は、Let's Encrypt の証明書を取得できません。
6. E メールアドレスの共有と IP アドレスのログ記録に関するプロンプトに適宜応答します。

7. Let's Encrypt から、指定されたドメインの所有者であることの検証を求められます。これを行うには、ドメインの DNS レコードに TXT レコードを追加します。以下の例に示すように 2 組の TXT レコード値が提供されます。

> **Note**
>
> Let's Encrypt では検証に必要な TXT レコードを 1 つまたは複数提供する場合があります。この例では、検証に使用する 2 つの TXT レコードが提供されました。



8. Lightsail のブラウザベースの SSH セッションは開いたままにします。このチュートリアルで後ほど戻ります。このチュートリアルの「次のセクション」に進みます。

ステップ 4: ドメインの DNS ゾーンに TXT レコードを追加する

ドメインの DNS ゾーンに TXT レコードを追加すると、自分がドメインを所有していることが検証されます。ここでは、デモの目的で Lightsail の DNS ゾーンを使用します。ただし、ドメインレジストラがホストする他の一般的な DNS ゾーンでも手順はほぼ同じです。

> **Note**
>
> ドメインの Lightsail DNS ゾーンの作成方法の詳細については、「Lightsail で DNS ゾーンを作成し、ドメインの DNS レコードを管理する」を参照してください。

Lightsail でドメインの DNS ゾーンに TXT レコードを追加するには

1. Lightsail のホームページで [Domains & DNS] (ドメインと DNS) タブを選択します。
2. ページの [DNS ゾーン] セクションで、Certbot 証明書リクエストで指定したドメインの DNS ゾーンを選択します。
3. DNS ゾーンエディタで [DNS records] (DNS レコード) を選択します。
4. [レコードの追加] を選択します。
5. [Record type] (レコードタイプ) のドロップダウンメニューで [TXT record] (TXT レコード) を選択します。
6. Let's Encrypt 証明書のリクエストで指定された値を [Record name] (レコード名) と [Responds with] (応答) フィールドに入力します。

   > **Note**
   >
   > Lightsail コンソールには、ドメインの頂点部分があらかじめ入力されています。たとえば、*_acme-challenge.example.com* サブドメインを追加する場合は、*_acme-challenge* をテキストボックスに入力するだけで、レコードを保存するときに Lightsail が `.example.com` の部分を追加します。

7. [Save (保存)] を選択します。
8. ステップ 4 ~ 7 を繰り返して、Let's Encrypt の証明書リクエストで指定された 2 番目の TXT レコードのセットを追加します。
9. Lightsail コンソールのブラウザウィンドウは、このチュートリアルで後ほど戻るので開いたままにします。このチュートリアルの「次のセクション」に進みます。

ステップ 5: TXT レコードが反映されたことを確認する

MxToolbox ユーティリティを使用して TXT レコードがインターネットの DNS に反映されたことを確認します。DNS レコードの反映には、DNS ホスティングプロバイダーと DNS レコードの有効期限 (TTL) の設定によって時間がかかる場合があります。このステップを完了し、TXT レコードが反映されたことを確認した上で、Certbot 証明書のリクエストに進むことが重要です。そうしないと、証明書のリクエストは失敗します。

TXT レコードがインターネットの DNS に反映されたことを確認するには

1. 新しいブラウザウィンドウを開き、https://mxtoolbox.com/TXTLookup.aspx に移動します。

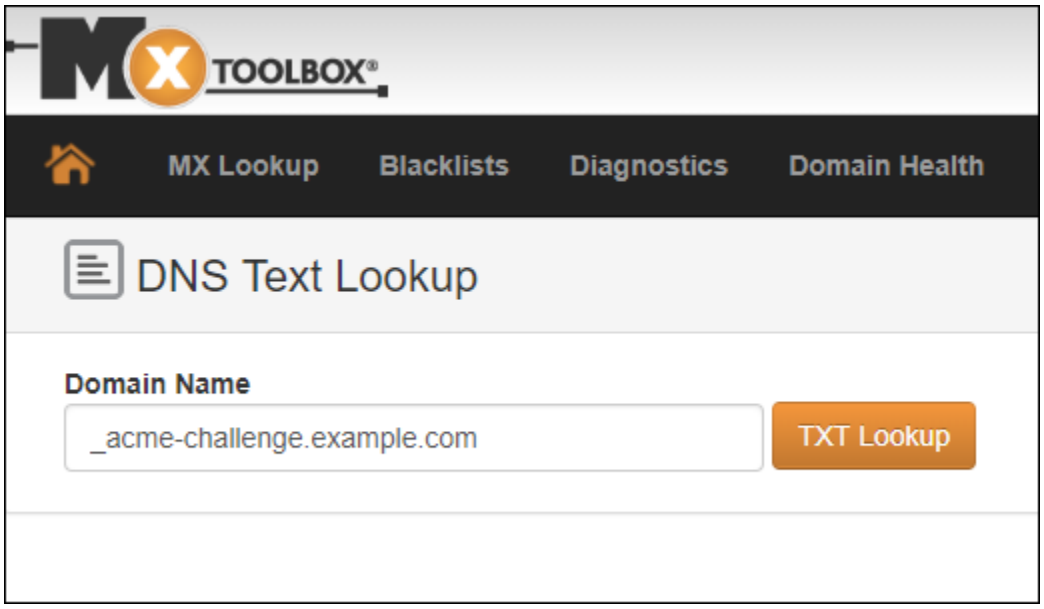
2. 次の内容をテキストボックスに入力します。

```
_acme-challenge.Domain
```

Domain は、登録済みのドメイン名に置き換えます。

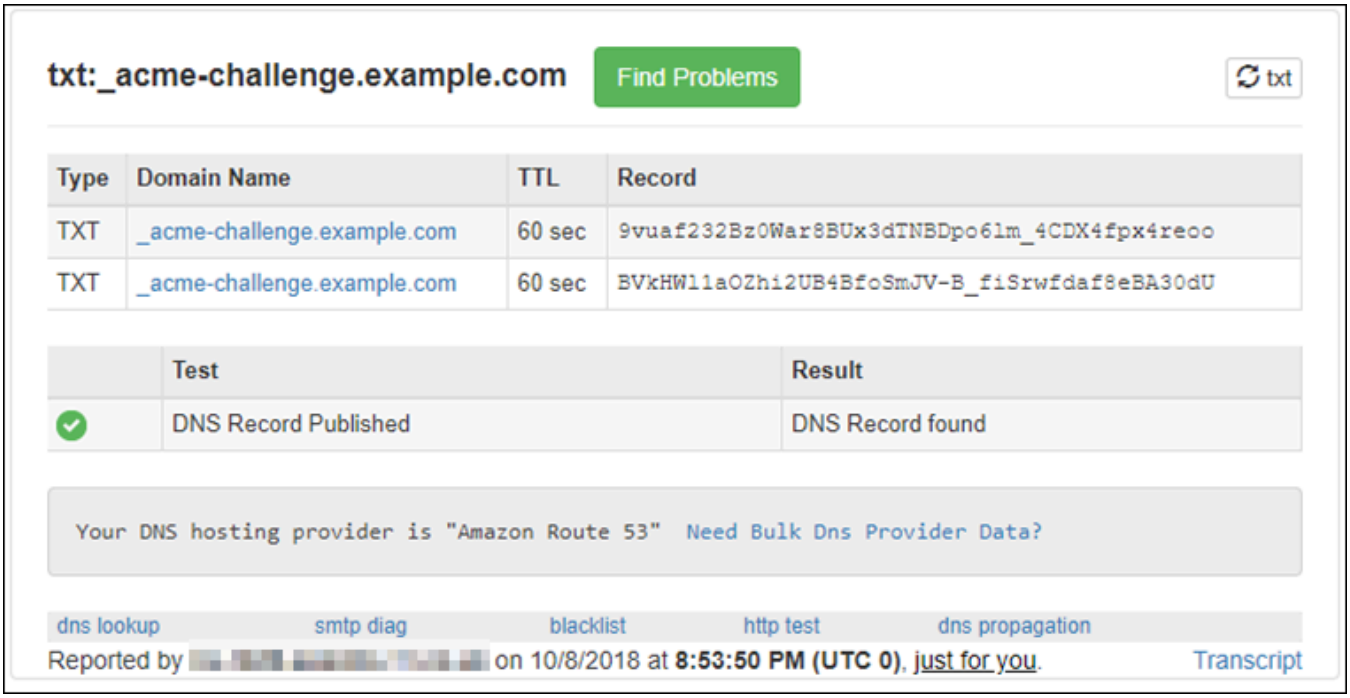
例:

```
_acme-challenge.example.com
```



3. [TXT Lookup (TXT ルックアップ)] を選択して確認を行います。
4. 以下のいずれかのレスポンスが返されます。

   - TXT レコードがインターネットの DNS に反映された場合は、次のスクリーンショットに示すようなレスポンスが表示されます。ブラウザウィンドウを閉じて、このチュートリアルの「次のセクション」に進みます。



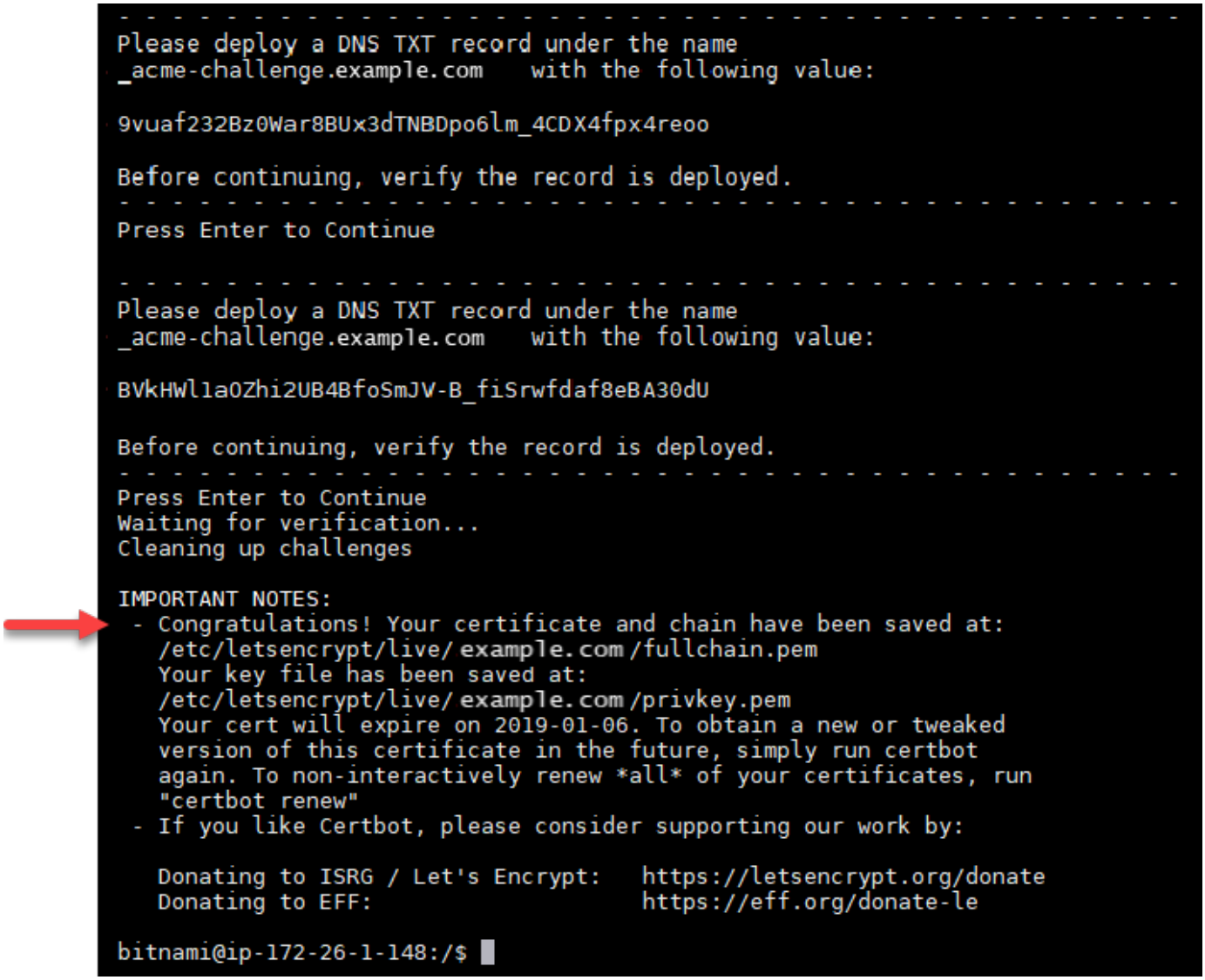
- TXT レコードがインターネットの DNS に反映されていない場合は、[DNS Record not found (DNS レコードが見つかりません)] というレスポンスが返されます。適切な DNS レコードをドメインの DNS ゾーンに追加したことを確認してください。適切なレコードを追加した場合は、ドメインの DNS レコードが反映されるまでしばらく待ってから、TXT のルックアップを再実行します。

ステップ 6: Let's Encrypt の SSL 証明書リクエストを完了する

LAMP インスタンスの Lightsail ブラウザベースの SSH セッションに戻り、Let's Encrypt 証明書のリクエストを完了します。Certbot は、SSL 証明書、チェーン、およびキーファイルを LAMP インスタンスの特定のディレクトリに保存します。

Let's Encrypt の SSL 証明書リクエストを完了するには

1. LAMP インスタンスの Lightsail ブラウザベースの SSH セッションで、Enter キーを押し、Let's Encrypt SSL 証明書のリクエストを続行します。成功すると、次のスクリーンショットに示すようなレスポンスが表示されます。



証明書、チェーン、およびキーファイルが /etc/letsencrypt/live/*Domain*/ ディレクトリに保存されたことを確認するメッセージが表示されます。*[Domain]* (ドメイン) は、登録済みのドメイン名 (/etc/letsencrypt/live/*example.com*/ など) になります。

2. メッセージに記載されている有効期限を書き留めておきます。この期限日までに証明書を更新する必要があります。

```
IMPORTANT NOTES:
 - Congratulations! Your certificate and chain have been saved at:
   /etc/letsencrypt/live/example.com/fullchain.pem
   Your key file has been saved at:
   /etc/letsencrypt/live/example.com/privkey.pem
   Your cert will expire on 2019-01-06. To obtain a new or tweaked
   version of this certificate in the future, simply run certbot
   again. To non-interactively renew *all* of your certificates, run
   "certbot renew"
 - If you like Certbot, please consider supporting our work by:

   Donating to ISRG / Let's Encrypt:   https://letsencrypt.org/donate
   Donating to EFF:                    https://eff.org/donate-le
```

3. これで Let's Encrypt SSL 証明書が手に入ったので、このチュートリアルの「次のセクション」に進みます。

ステップ 7: Apache サーバーディレクトリで Let's Encrypt の証明書ファイルへのリンクを作成する

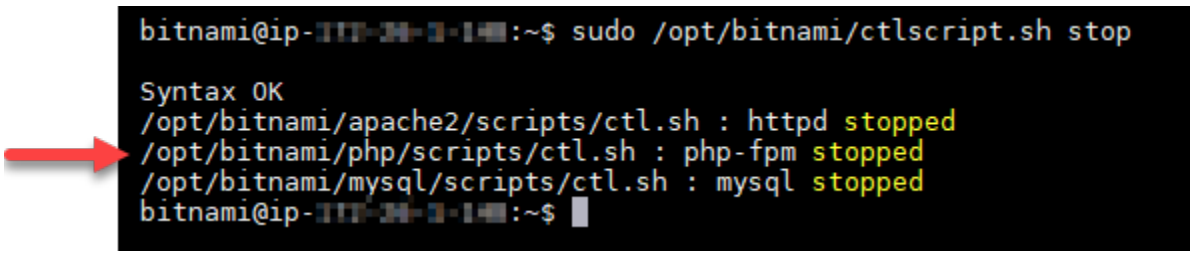
LAMP インスタンスの Apache サーバーディレクトリにある Let's Encrypt の SSL 証明書ファイルへのリンクを作成します。また、必要になる場合に備えて既存の証明書をバックアップします。

Apache サーバーディレクトリで Let's Encrypt の証明書ファイルへのリンクを作成するには

1. LAMP インスタンスの Lightsail ブラウザベースの SSH セッションで、次のコマンドを入力して基盤となる LAMP スタックサービスを停止します。

   ```
   sudo /opt/bitnami/ctlscript.sh stop
   ```

   次のようなレスポンスが表示されます。

2. 次のコマンドを入力してドメインの環境変数を設定します。

   ```
   DOMAIN=Domain
   ```

   コマンドで、*Domain* を登録済みのドメイン名に置き換えます。

例:

```
DOMAIN=example.com
```

3. 次のコマンドを入力し、変数が正しい値を返すことを確認します。

```
echo $DOMAIN
```

次のような結果が表示されます。

4. バックアップとして既存の証明書ファイルがある場合、以下のコマンドを個別に入力して名前を書き換えます。さまざまなディストリビューションとファイル構造の詳細については、このチュートリアルの冒頭の重要ブロックを参照してください。

   - Debian Linux ディストリビューションの場合

     アプローチ A (システムパッケージを使用した Bitnami インストール):

     ```
     sudo mv /opt/bitnami/apache2/conf/bitnami/certs/server.crt /opt/bitnami/apache2/
     conf/bitnami/certs/server.crt.old
     ```

     ```
     sudo mv /opt/bitnami/apache2/conf/bitnami/certs/server.key /opt/bitnami/apache2/
     conf/bitnami/certs/server.key.old
     ```

     アプローチ B (自己完結型 Bitnami インストール):

     ```
     sudo mv /opt/bitnami/apache2/conf/server.crt /opt/bitnami/apache2/conf/
     server.crt.old
     ```

     ```
     sudo mv /opt/bitnami/apache2/conf/server.key /opt/bitnami/apache2/conf/
     server.key.old
     ```

   - Ubuntu Linux ディストリビューションを使用する古いインスタンスの場合：

```
sudo mv /opt/bitnami/apache/conf/bitnami/certs/server.crt /opt/bitnami/apache/
conf/bitnami/certs/server.crt.old
```

```
sudo mv /opt/bitnami/apache/conf/bitnami/certs/server.key /opt/bitnami/apache/
conf/bitnami/certs/server.key.old
```

5. 以下のコマンドを個別に入力し、Apache2 ディレクトリで Let's Encrypt の証明書ファイルへのリンクを作成します。さまざまなディストリビューションとファイル構造の詳細については、このチュートリアルの冒頭の重要ブロックを参照してください。

   - Debian Linux ディストリビューションの場合

     アプローチ A (システムパッケージを使用した Bitnami インストール):

     ```
     sudo ln -sf /etc/letsencrypt/live/$DOMAIN/privkey.pem /opt/bitnami/apache2/conf/
     bitnami/certs/server.key
     ```

     ```
     sudo ln -sf /etc/letsencrypt/live/$DOMAIN/fullchain.pem /opt/bitnami/apache2/
     conf/bitnami/certs/server.crt
     ```

     アプローチ B (自己完結型 Bitnami インストール):

     ```
     sudo ln -sf /etc/letsencrypt/live/$DOMAIN/privkey.pem /opt/bitnami/apache2/conf/
     server.key
     ```

     ```
     sudo ln -sf /etc/letsencrypt/live/$DOMAIN/fullchain.pem /opt/bitnami/apache2/
     conf/server.crt
     ```

   - Ubuntu Linux ディストリビューションを使用する古いインスタンスの場合：

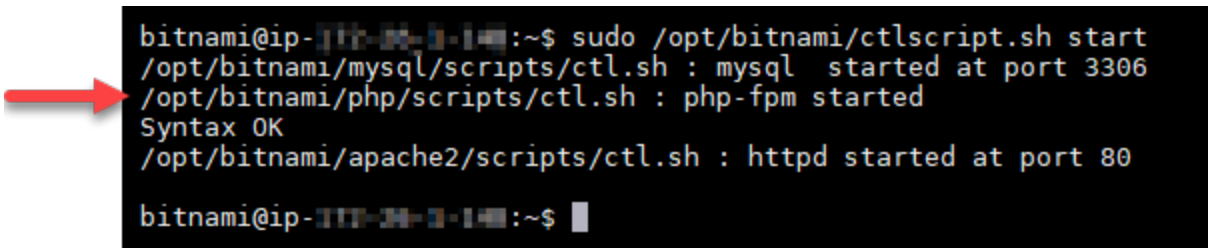
     ```
     sudo ln -s /etc/letsencrypt/live/$DOMAIN/privkey.pem /opt/bitnami/apache/conf/
     bitnami/certs/server.key
     ```

     ```
     sudo ln -s /etc/letsencrypt/live/$DOMAIN/fullchain.pem /opt/bitnami/apache/conf/
     bitnami/certs/server.crt
     ```

6. 次のコマンドを入力して、以前に停止した基盤となる LAMP スタックサービスを開始します。

```
sudo /opt/bitnami/ctlscript.sh start
```

次のような結果が表示されます。



これで SSL 暗号化を使用するように LAMP インスタンスが設定されました。ただし、トラフィックは HTTP から HTTPS に自動的にはリダイレクトされません。

7. このチュートリアルの「次のセクション」に進みます。

ステップ 8: ウェブアプリケーションの HTTP から HTTPS へのリダイレクトを設定する

LAMP インスタンスの HTTP から HTTPS へのリダイレクトを設定できます。HTTP から HTTPS へのリダイレクトを自動的に行うことで、SSL を使用するユーザーにのみ (HTTP を使用して接続した場合でも) サイトへのアクセスを許可できます。

ウェブアプリケーションの HTTP から HTTPS へのリダイレクトを設定するには

1. LAMP インスタンスの Lightsail ブラウザベースの SSH セッションで、次のコマンドを入力し、Vim テキストエディタを使用して Apache ウェブサーバー設定ファイルを編集します。

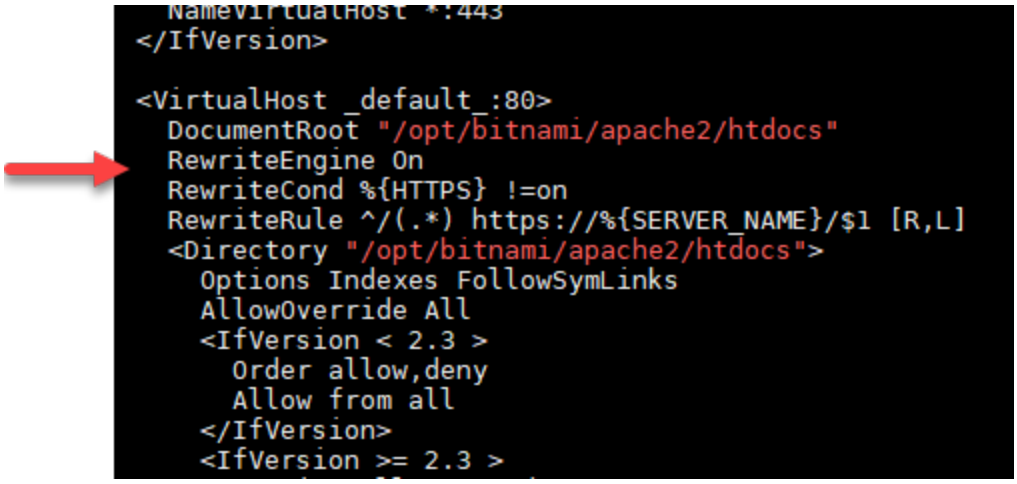
```
sudo vim /opt/bitnami/apache2/conf/bitnami/bitnami.conf
```

> **Note**
>
> このチュートリアルではデモの目的で Vim を使用していますが、任意のテキストエディタを使用できます。

2. `i` キーを押して Vim エディタを挿入モードにします。
3. このファイルで、`DocumentRoot "/opt/bitnami/apache2/htdocs"` と `<Directory "/opt/bitnami/apache2/htdocs">` の間に次のテキストを入力します。

```
RewriteEngine On
RewriteCond %{HTTPS} !=on
RewriteRule ^/(.*) https://%{SERVER_NAME}/$1 [R,L]
```

結果は次のようになります。



4. ESC キーを押して「:wq」と入力し、編集内容を書き込んで (保存して) Vim を終了します。
5. 次のコマンドを入力して基盤となる LAMP スタックサービスを再開し、編集内容を反映します。

```
sudo /opt/bitnami/ctlscript.sh restart
```

これで、HTTP から HTTPS へ自動的に接続をリダイレクトするように LAMP インスタンスが設定されました。訪問者が `http://www.example.com` にアクセスすると、暗号化された `https://www.example.com` アドレスに自動的にリダイレクトされます。

ステップ 9: Let's Encrypt 証明書を 90 日ごとに更新する

Let's Encrypt 証明書の有効期間は 90 日間です。証明書は有効期限が切れる 30 日前から更新できます。Let's Encrypt 証明書を更新するには、取得するために使用した元のコマンドを実行します。このチュートリアルの「Let's Encrypt の SSL ワイルドカード証明書をリクエストする」セクションのステップを繰り返します。

チュートリアル: Lightsail の Nginx インスタンスで Let's Encrypt の SSL 証明書を使用する

Amazon Lightsail では、Lightsail ロードバランサーを使用すると、SSL/TLS でウェブサイトとアプリケーションのセキュリティを簡単に強化できます。ただし、Lightsail ロードバランサーの使用は一般的に最適な選択肢ではない場合があります。ロードバランサーが提供するスケーラビリティや耐障害性がサイトでは必要ない場合や、コストを最適化するためにロードバランサーを使用しない場合があります。

後者の場合は、Let's Encrypt で無料の SSL 証明書を入手できます。無料の証明書を使用することに問題はありません。これらの証明書は Lightsail インスタンスに統合できます。このチュートリアルでは、Certbot を使用して Let's Encrypt ワイルドカード証明書をリクエストし、これを Nginx インスタンスに統合する方法を示します。

Important

- Bitnami インスタンスで使用されている Linux ディストリビューションは、2020 年 7 月に Ubuntu から Debian に変更されました。この変更により、このチュートリアルのいくつかのステップは、インスタンスの Linux ディストリビューションによって異なります。変更後に作成された Bitnami ブループリントインスタンスはすべて Debian Linux ディストリビューションを使用します。変更前に作成されたインスタンスは、Ubuntu Linux ディストリビューションを引き続き使用します。インスタンスのディストリビューションをチェックするには、`uname -a` コマンドを実行します。応答には、インスタンスの Linux ディストリビューションとして Ubuntu または Debian のいずれかが表示されます。
- Bitnami は、多くのスタックのファイル構造を変更するプロセスです。このチュートリアルのファイルパスは、Bitnami スタックがネイティブ Linux システムパッケージを使用しているか (アプローチ A)、または自己完結型インストール (アプローチ B) であるかによって、変更される場合があります。Bitnami のインストールタイプと取るべき方法を特定するには、次のコマンドを実行します。

  ```
  test ! -f "/opt/bitnami/common/bin/openssl" && echo "Approach
  A: Using system packages." || echo "Approach B: Self-contained
  installation."
  ```

目次

ステップ 1: 前提条件を満たす

以下の前提条件を満たします (まだ満たしていない場合)。

- Lightsail に Nginx インスタンスを作成します。詳細については、「インスタンスを作成する」を参照してください。
- ドメイン名を登録し、その DNS レコードを編集するための管理アクセスを取得します。詳細については、「DNS」を参照してください。

> **Note**
>
> ドメインの DNS レコードは、Lightsail の DNS ゾーンを使用して管理することをお勧めします。詳細については、「DNS ゾーンを作成してドメインの DNS レコードを管理する」を参照してください。

- Lightsail コンソールでブラウザベースの SSH ターミナルを使用して、このチュートリアルのステップを実行します。ただし、独自の SSH クライアント (PuTTY など) を使用することもできます。PuTTY の設定の詳細については、「Amazon Lightsail で PuTTY をダウンロードし、SSH を使用して接続するようにセットアップする」を参照してください。

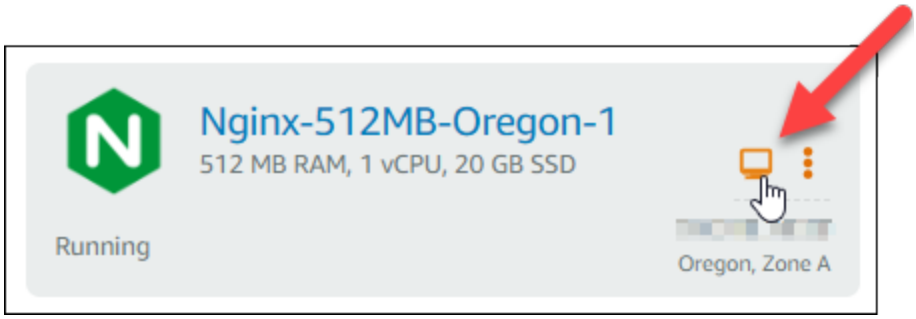
前提条件が完了したら、このチュートリアルの「次のセクション」に進みます。

ステップ 2: Lightsail インスタンスに Certbot をインストールする

Certbot は、Let's Encrypt の証明書をリクエストしてウェブサーバーにデプロイするために使用するクライアントです。Let's Encrypt は ACME プロトコルを使用して証明書を発行します。Certbot は、Let's Encrypt とやり取りする ACME 対応のクライアントです。

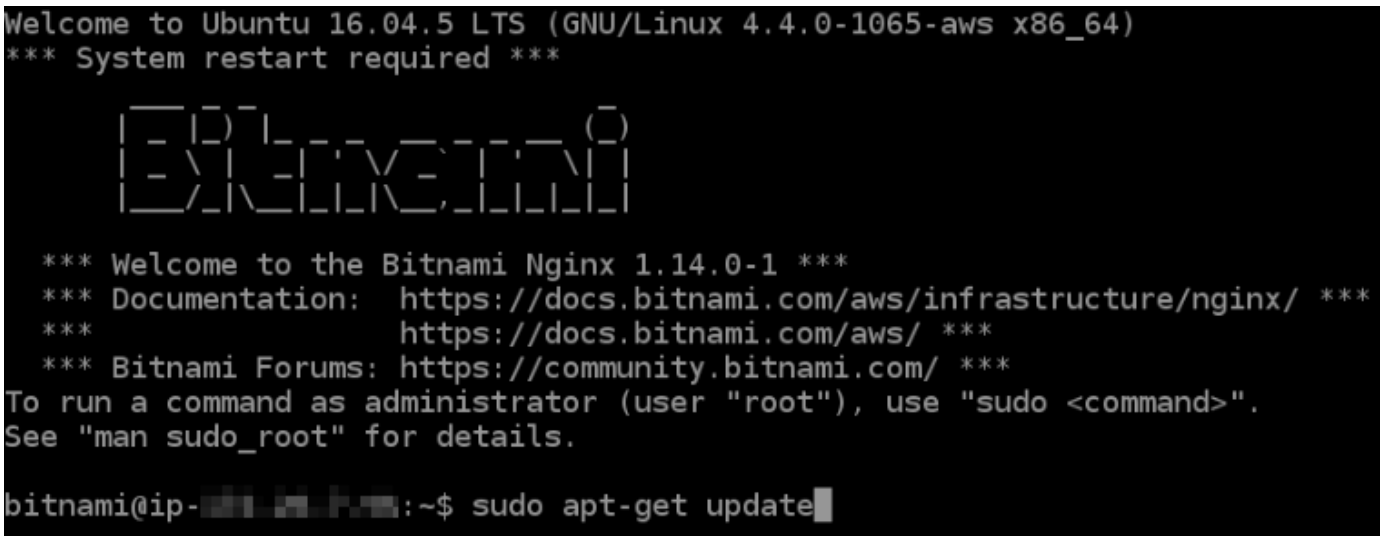
Lightsail インスタンスに Certbot をインストールするには

1. Lightsail コンソールにサインインします。
2. Lightsail のホームページで、接続先のインスタンスの SSH クイック接続アイコンを選択します。



3. Lightsail のブラウザベースの SSH セッションに接続したら、次のコマンドを入力してインスタンスのパッケージを更新します。

```
sudo apt-get update
```



4. 次のコマンドを入力してソフトウェアプロパティパッケージをインストールします。Certbot の開発者は、Personal Package Archive (PPA) を使用して Cerbot を配信します。ソフトウェアプロパティパッケージを使用すると、PPA をより効率的に操作できます。

```
sudo apt-get install software-properties-common
```

> **Note**
>
> `sudo apt-get install` コマンドを実行したときに `Could not get lock` エラーが発生した場合は、約 15 分待ってから再試行してください。このエラーは、自動アップグレードをインストールするために Apt パッケージ管理ツールを使用している cron ジョブが原因で発生している可能性があります。

5. 次のコマンドを入力して Certbot をローカル apt リポジトリに追加します。

> **Note**
>
> ステップ 5 は、Ubuntu Linux ディストリビューションを使用するインスタンスにのみ適用されます。インスタンスが Debian Linux ディストリビューションを使用している場合は、このステップをスキップしてください。

```
sudo apt-add-repository ppa:certbot/certbot -y
```

6. 次のコマンドを入力して apt を更新し、新しいリポジトリを含めます。

```
sudo apt-get update -y
```

7. 次のコマンドを入力して Certbot をインストールします。

```
sudo apt-get install certbot -y
```

これで Lightsail インスタンスに Cerbot がインストールされました。

8. ブラウザベースの SSH ターミナルウィンドウは開いたままにします。このチュートリアルで後ほど戻ります。このチュートリアルの「次のセクション」に進みます。

ステップ 3: Let's Encrypt の SSL ワイルドカード証明書をリクエストする

Let's Encrypt の証明書をリクエストするプロセスを開始します。Certbot を使用してワイルドカード証明書をリクエストします。この 1 つの証明書をドメインとそのサブドメインの両方に使用できます。たとえば、1 つのワイルドカード証明書を `example.com` 最上位ドメイン、`blog.example.com` サブドメイン、および `stuff.example.com` サブドメインに使用できます。

Let's Encrypt の SSL ワイルドカード証明書をリクエストするには

1. このチュートリアルのステップ 2 で使用した同じブラウザベースの SSH ターミナルウィンドウで、以下のコマンドを入力してドメインの環境変数を設定します。より効率的にコマンドをコピーして貼り付け、証明書を取得できます。*domain* を登録済みのドメイン名に置き換えます。

   ```
   DOMAIN=domain
   ```

   ```
   WILDCARD=*.$DOMAIN
   ```

   例:

   ```
   DOMAIN=example.com
   ```

   ```
   WILDCARD=*.$DOMAIN
   ```

2. 次のコマンドを入力し、変数が正しい値を返すことを確認します。

   ```
   echo $DOMAIN && echo $WILDCARD
   ```

   次のような結果が表示されます。

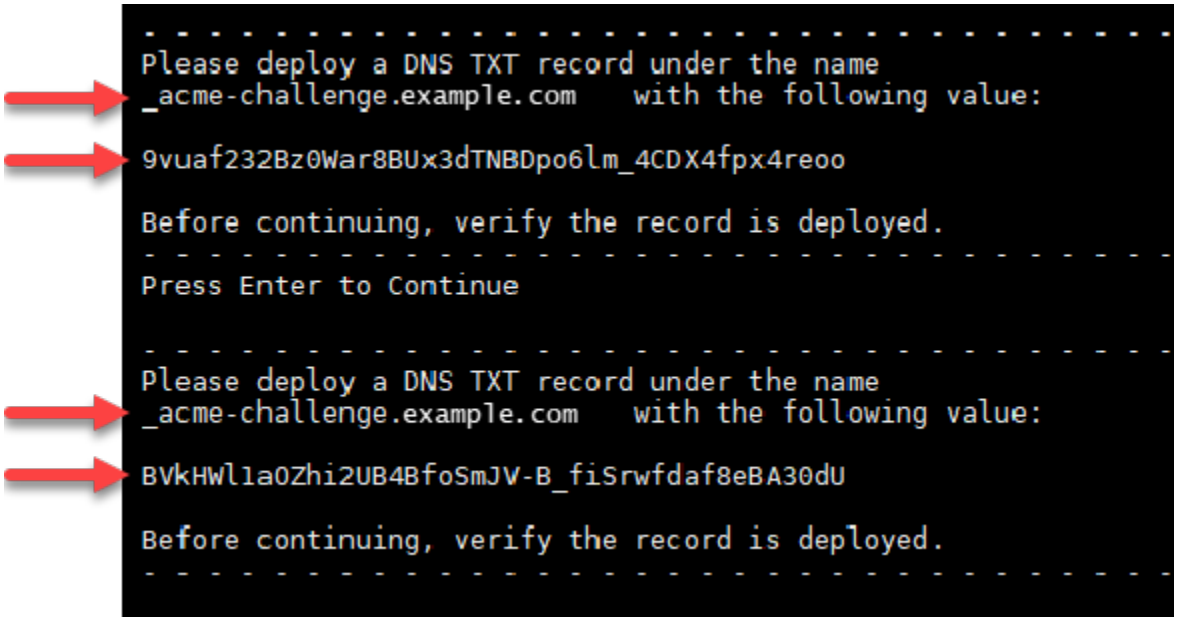
3. 次のコマンドを入力して Certbot をインタラクティブモードで起動します。このコマンドでは、DNS チャレンジで手動認証を使用してドメインの所有権を検証することを Certbot に指示

します。また、最上位ドメインとそのサブドメイン用にワイルドカード証明書をリクエストします。

```
sudo certbot -d $DOMAIN -d $WILDCARD --manual --preferred-challenges dns certonly
```

4. プロンプトに応じて E メールアドレスを入力します。これで更新とセキュリティに関する通知を受信します。
5. Let's Encrypt のサービス利用規約を読みます。読み終わり、同意する場合は A キーを押します。同意しない場合は、Let's Encrypt の証明書を取得できません。
6. E メールアドレスの共有と IP アドレスのログ記録に関するプロンプトに適宜応答します。
7. Let's Encrypt から、指定されたドメインの所有者であることの検証を求められます。これを行うには、ドメインの DNS レコードに TXT レコードを追加します。以下の例に示すように 2 組の TXT レコード値が提供されます。

> **Note**
>
> Let's Encrypt では検証に必要な TXT レコードを 1 つまたは複数提供する場合があります。この例では、検証に使用する 2 つの TXT レコードが提供されました。



8. Lightsail のブラウザベースの SSH セッションは開いたままにします。このチュートリアルで後ほど戻ります。このチュートリアルの「次のセクション」に進みます。

ステップ 4: ドメインの DNS ゾーンに TXT レコードを追加する

ドメインの DNS ゾーンに TXT レコードを追加すると、自分がドメインを所有していることが検証されます。ここでは、デモの目的で Lightsail の DNS ゾーンを使用します。ただし、ドメインレジストラがホストする他の一般的な DNS ゾーンでも手順はほぼ同じです。

> **Note**
>
> ドメインの Lightsail DNS ゾーンの作成方法の詳細については、「Lightsail で DNS ゾーンを作成し、ドメインの DNS レコードを管理する」を参照してください。

Lightsail でドメインの DNS ゾーンに TXT レコードを追加するには

1. Lightsail のホームページで [Domains & DNS] (ドメインと DNS) タブを選択します。
2. ページの [DNS ゾーン] セクションで、Certbot 証明書リクエストで指定したドメインの DNS ゾーンを選択します。
3. DNS ゾーンエディタで [DNS records] (DNS レコード) を選択します。
4. [レコードの追加] を選択します。
5. [Record type] (レコードタイプ) のドロップダウンメニューで [TXT record] (TXT レコード) を選択します。
6. Let's Encrypt 証明書のリクエストで指定された値を [Record name] (レコード名) と [Responds with] (応答) フィールドに入力します。

   > **Note**
   >
   > Lightsail コンソールには、ドメインの頂点部分があらかじめ入力されています。たとえば、*_acme-challenge.example.com* サブドメインを追加する場合は、*_acme-challenge* をテキストボックスに入力するだけで、レコードを保存するときに Lightsail が `.example.com` の部分を追加します。

7. [Save (保存)] を選択します。
8. ステップ 4 ~ 7 を繰り返して、Let's Encrypt の証明書リクエストで指定された 2 番目の TXT レコードのセットを追加します。
9. Lightsail コンソールのブラウザウィンドウは、このチュートリアルで後ほど戻るので開いたままにします。このチュートリアルの「次のセクション」に進みます。

ステップ 5: TXT レコードが反映されたことを確認する

MxToolbox ユーティリティを使用して TXT レコードがインターネットの DNS に反映されたことを確認します。DNS レコードの反映には、DNS ホスティングプロバイダーと DNS レコードの有効期限 (TTL) の設定によって時間がかかる場合があります。このステップを完了し、TXT レコードが反映されたことを確認した上で、Certbot 証明書のリクエストに進むことが重要です。そうしないと、証明書のリクエストは失敗します。

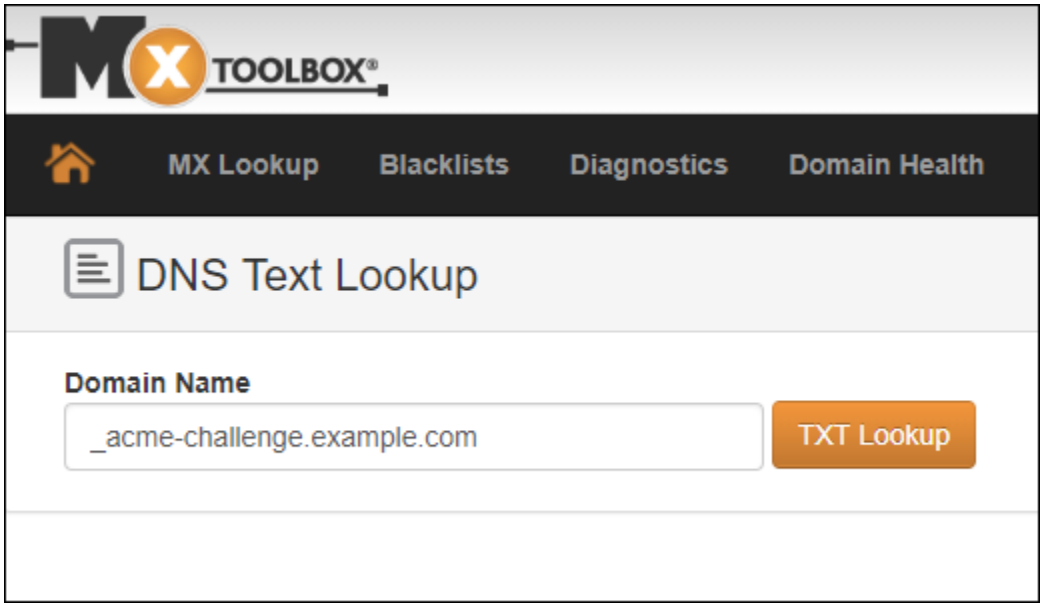
TXT レコードがインターネットの DNS に反映されたことを確認するには

1. 新しいブラウザウィンドウを開き、https://mxtoolbox.com/TXTLookup.aspx に移動します。
2. 次の内容をテキストボックスに入力します。*domain* は実際のドメインに置き換えてください。

   ```
   _acme-challenge.domain
   ```

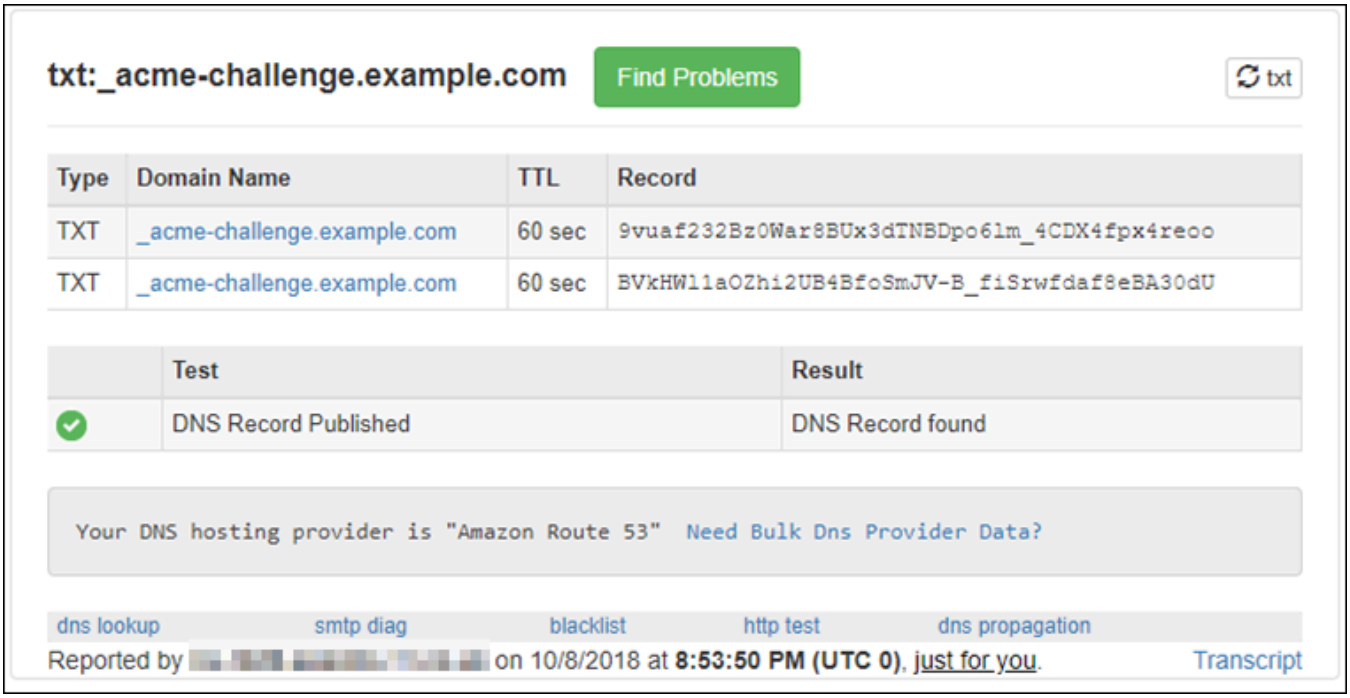
   例:

   ```
   _acme-challenge.example.com
   ```



3. [TXT Lookup (TXT ルックアップ)] を選択して確認を行います。
4. 以下のいずれかのレスポンスが返されます。

- TXT レコードがインターネットの DNS に反映された場合は、次のスクリーンショットに示すようなレスポンスが表示されます。ブラウザウィンドウを閉じて、このチュートリアルの「次のセクション」に進みます。



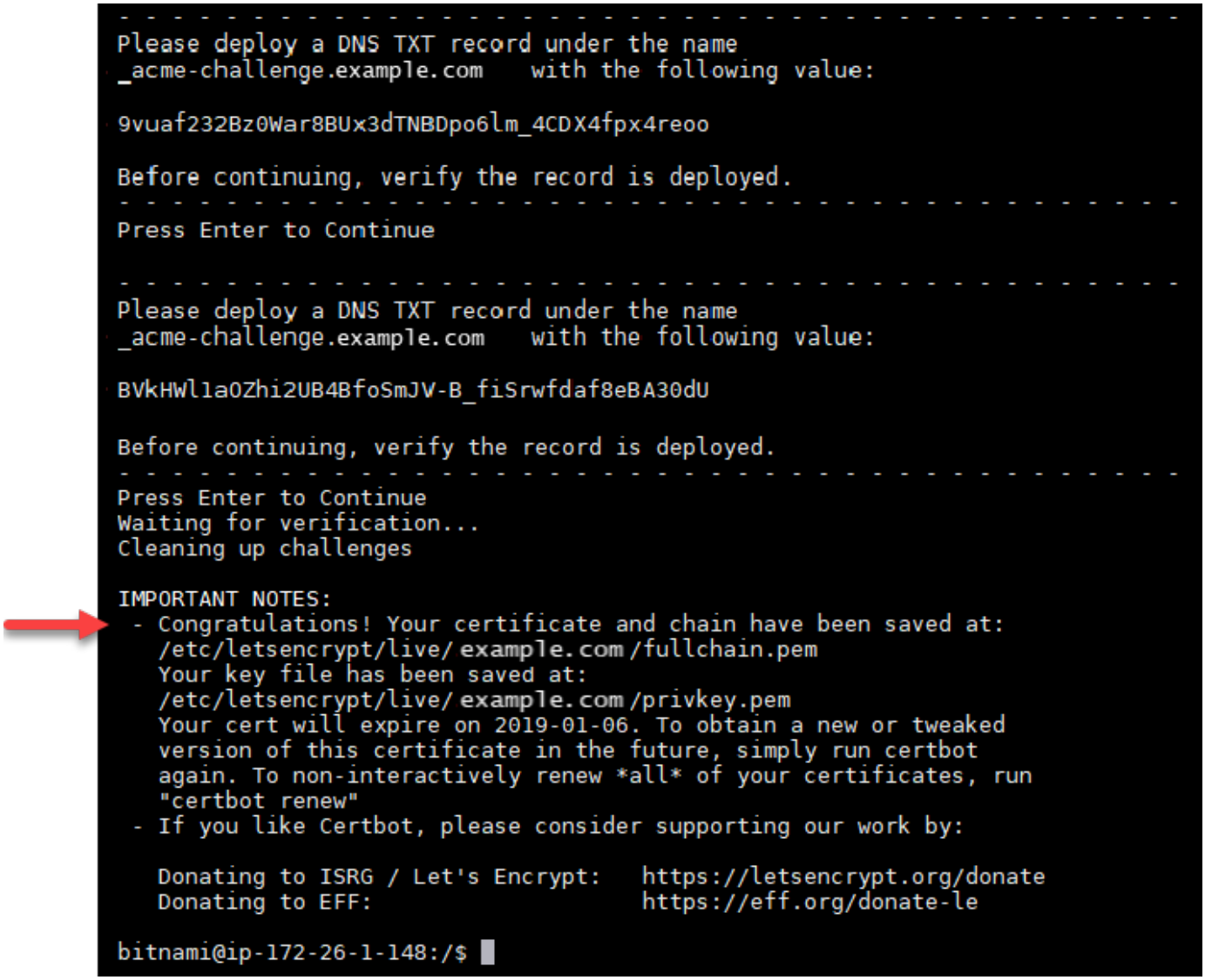
- TXT レコードがインターネットの DNS に反映されていない場合は、[DNS Record not found (DNS レコードが見つかりません)] というレスポンスが返されます。適切な DNS レコードをドメインの DNS ゾーンに追加したことを確認してください。適切なレコードを追加した場合は、ドメインの DNS レコードが反映されるまでしばらく待ってから、TXT のルックアップを再実行します。

ステップ 6: Let's Encrypt の SSL 証明書リクエストを完了する

Nginx インスタンスの Lightsail ブラウザベースの SSH セッションに戻り、Let's Encrypt 証明書のリクエストを完了します。Certbot は、SSL 証明書、チェーン、およびキーファイルを Nginx インスタンスの特定のディレクトリに保存します。

Let's Encrypt の SSL 証明書リクエストを完了するには

1. Nginx インスタンスの Lightsail ブラウザベースの SSH セッションで、Enter キーを押し、Let's Encrypt SSL 証明書のリクエストを続行します。成功すると、次のスクリーンショットに示すようなレスポンスが表示されます。



証明書、チェーン、およびキーファイルが /etc/letsencrypt/live/*domain*/ ディレクトリに保存されたことを確認するメッセージが表示されます。*domain* は、実際のドメイン (/etc/letsencrypt/live/*example.com*/ など) に置き換えてください。

2. メッセージに記載されている有効期限を書き留めておきます。この期限日までに証明書を更新する必要があります。

```
IMPORTANT NOTES:
 - Congratulations! Your certificate and chain have been saved at:
   /etc/letsencrypt/live/example.com/fullchain.pem
   Your key file has been saved at:
   /etc/letsencrypt/live/example.com/privkey.pem
   Your cert will expire on 2019-01-06. To obtain a new or tweaked
   version of this certificate in the future, simply run certbot
   again. To non-interactively renew *all* of your certificates, run
   "certbot renew"
 - If you like Certbot, please consider supporting our work by:

   Donating to ISRG / Let's Encrypt:   https://letsencrypt.org/donate
   Donating to EFF:                    https://eff.org/donate-le
```

3. これで Let's Encrypt SSL 証明書が手に入ったので、このチュートリアルの「次のセクション」に進みます。

ステップ 7: Nginx サーバーディレクトリに Let's Encrypt の証明書ファイルへのリンクを作成する

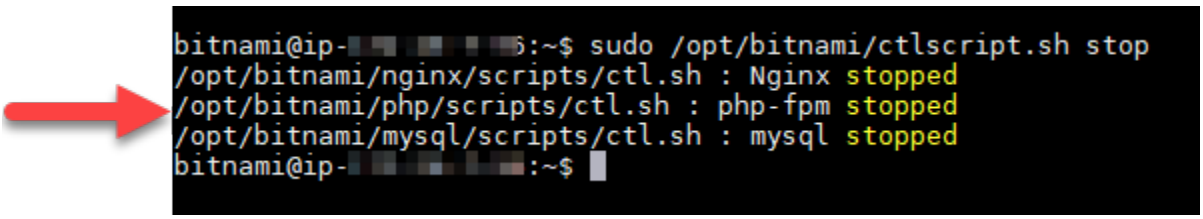
Let's Encrypt の SSL 証明書ファイルへのリンクを、Nginx インスタンスの Nginx サーバーディレクトリに作成します。また、必要になる場合に備えて既存の証明書をバックアップします。

Nginx サーバーディレクトリで Let's Encrypt の証明書ファイルへのリンクを作成するには

1. Nginx インスタンスの Lightsail ブラウザベースの SSH セッションで、次のコマンドを入力して基本サービスを停止します。

   ```
   sudo /opt/bitnami/ctlscript.sh stop
   ```

   次のようなレスポンスが表示されます。

2. 次のコマンドを入力してドメインの環境変数を設定します。コマンドのコピー＆ペーストで、より効率的に証明書ファイルにリンクを張れます。*domain* は登録済みのドメイン名に置き換えてください。

   ```
   DOMAIN=domain
   ```

例:

```
DOMAIN=example.com
```

3. 次のコマンドを入力し、変数が正しい値を返すことを確認します。

```
echo $DOMAIN
```

次のような結果が表示されます。

4. バックアップとして既存の証明書ファイルがある場合、以下のコマンドを個別に入力して名前を書き換えます。さまざまなディストリビューションとファイル構造の詳細については、このチュートリアルの冒頭の重要ブロックを参照してください。

   - Debian Linux ディストリビューションの場合

     アプローチ A (システムパッケージを使用した Bitnami インストール):

     ```
     sudo mv /opt/bitnami/nginx/conf/bitnami/certs/server.crt /opt/bitnami/nginx/conf/
     bitnami/certs/server.crt.old
     ```

     ```
     sudo mv /opt/bitnami/nginx/conf/bitnami/certs/server.key /opt/bitnami/nginx/conf/
     bitnami/certs/server.key.old
     ```

     アプローチ B (自己完結型 Bitnami インストール):

     ```
     sudo mv /opt/bitnami/nginx/conf/server.crt /opt/bitnami/nginx/conf/server.crt.old
     ```

     ```
     sudo mv /opt/bitnami/nginx/conf/server.key /opt/bitnami/nginx/conf/server.key.old
     ```

   - Ubuntu Linux ディストリビューションを使用する古いインスタンスの場合 :

     ```
     sudo mv /opt/bitnami/nginx/conf/bitnami/certs/server.crt /opt/bitnami/nginx/conf/
     bitnami/certs/server.crt.old
     ```

```
sudo mv /opt/bitnami/nginx/conf/bitnami/certs/server.key /opt/bitnami/nginx/conf/
bitnami/certs/server.key.old
```

5. 以下のコマンドを個別に入力し、Nginx サーバーディレクトリにある Let's Encrypt の証明書ファイルへのリンクを作成します。さまざまなディストリビューションとファイル構造の詳細については、このチュートリアルの冒頭の重要ブロックを参照してください。

   - Debian Linux ディストリビューションの場合

     アプローチ A (システムパッケージを使用した Bitnami インストール):

     ```
     sudo ln -sf /etc/letsencrypt/live/$DOMAIN/privkey.pem /opt/bitnami/nginx/conf/
     bitnami/certs/server.key
     ```

     ```
     sudo ln -sf /etc/letsencrypt/live/$DOMAIN/fullchain.pem /opt/bitnami/nginx/conf/
     bitnami/certs/server.crt
     ```

     アプローチ B (自己完結型 Bitnami インストール):

     ```
     sudo ln -sf /etc/letsencrypt/live/$DOMAIN/privkey.pem /opt/bitnami/nginx/conf/
     server.key
     ```

     ```
     sudo ln -sf /etc/letsencrypt/live/$DOMAIN/fullchain.pem /opt/bitnami/nginx/conf/
     server.crt
     ```

   - Ubuntu Linux ディストリビューションを使用する古いインスタンスの場合：

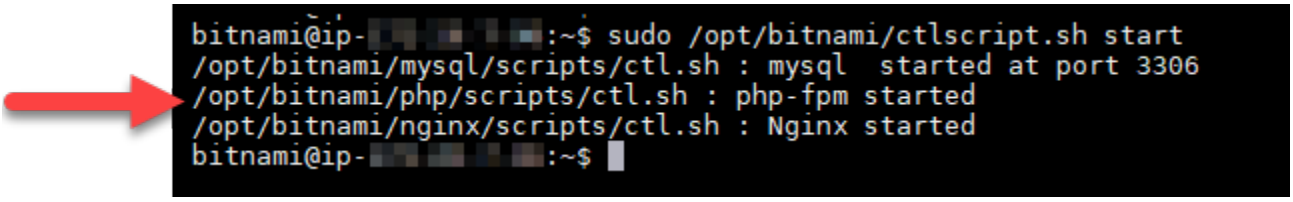
     ```
     sudo ln -s /etc/letsencrypt/live/$DOMAIN/privkey.pem /opt/bitnami/nginx/conf/
     bitnami/certs/server.key
     ```

     ```
     sudo ln -s /etc/letsencrypt/live/$DOMAIN/fullchain.pem /opt/bitnami/nginx/conf/
     bitnami/certs/server.crt
     ```

6. 次のコマンドを入力して、先ほど停止した基本サービスを開始します。

   ```
   sudo /opt/bitnami/ctlscript.sh start
   ```

   次のような結果が表示されます。



これで SSL 暗号化を使用するように Nginx インスタンスが設定されました。ただし、トラフィックは HTTP から HTTPS に自動的にはリダイレクトされません。

7. このチュートリアルの「次のセクション」に進みます。

ステップ 8: ウェブアプリケーションの HTTP から HTTPS へのリダイレクトを設定する

Nginx インスタンスの HTTP から HTTPS へのリダイレクトを設定することができます。HTTP から HTTPS へのリダイレクトを自動的に行うことで、SSL を使用するユーザーにのみ (HTTP を使用して接続した場合でも) サイトへのアクセスを許可できます。さまざまなディストリビューションとファイル構造の詳細については、このチュートリアルの冒頭の重要ブロックを参照してください。

このチュートリアルではデモの目的で Vim を使用していますが、任意のテキストエディタを使用できます。

Debian Linux ディストリビューション – ウェブアプリケーションの HTTP から HTTPS へのリダイレクトを設定する

1. Nginx インスタンスの Lightsail ブラウザベースの SSH セッションで、次のコマンドを入力し、サーバーブロック設定ファイルを変更します。アプリケーションの名前を <ApplicationName> に置き換えます。

```
sudo vim /opt/bitnami/nginx/conf/server_blocks/<ApplicationName>-server-block.conf
```

2. i キーを押して Vim エディタを挿入モードにします。
3. 次の例の情報を使用してファイルを編集します。

```
server {
    listen 80 default_server;
    root /opt/bitnami/APPNAME;
    return 301 https://$host$request_uri;
}
```

4. ESC キーを押して「:wq」と入力し、編集内容を書き込んで (保存して) Vim を終了します。

5. 次のコマンドを入力して、Nginx 設定ファイルのサーバーセクションを変更します。

```
sudo vim /opt/bitnami/nginx/conf/nginx.conf
```

6. i キーを押して Vim エディタを挿入モードにします。
7. 次の例の情報を使用してファイルを編集します。

```
server {
      listen 80;
      server_name localhost;
      return 301 https://$host$request_uri;
  }
```

8. ESC キーを押して「:wq」と入力し、編集内容を書き込んで (保存して) Vim を終了します。
9. 次のコマンドを入力して基本サービスを再開し、編集内容を反映させます。

```
sudo /opt/bitnami/ctlscript.sh restart
```

アプローチ B (自己完結型 Bitnami インストール):

1. Nginx インスタンスの Lightsail ブラウザベースの SSH セッションで、次のコマンドを入力し、Nginx 設定ファイルを変更します :

```
sudo vim /opt/bitnami/nginx/conf/nginx.conf
```

2. i キーを押して Vim エディタを挿入モードにします。
3. 次の例の情報を使用してファイルを編集します。

```
server {
      listen 80;
      server_name localhost;
      return 301 https://$host$request_uri;
  }
```

4. ESC キーを押して「:wq」と入力し、編集内容を書き込んで (保存して) Vim を終了します。
5. 次のコマンドを入力して基本サービスを再開し、編集内容を反映させます。

```
sudo /opt/bitnami/ctlscript.sh restart
```

Ubuntu Linux ディストリビューションを使用する古いインスタンスの場合 – ウェブアプリケーションの HTTP から HTTPS へのリダイレクトを設定する

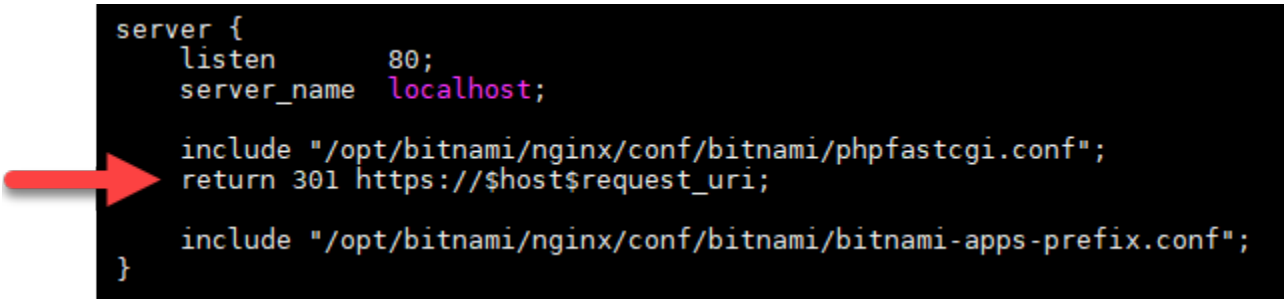
1. Nginx インスタンスの Lightsail ブラウザベースの SSH セッションで、次のコマンドを入力し、Vim テキストエディタを使用して Nginx ウェブサーバー設定ファイルを編集します。

   ```
   sudo vim /opt/bitnami/nginx/conf/bitnami/bitnami.conf
   ```

2. i キーを押して Vim エディタを挿入モードにします。
3. このファイルで、`server_name localhost;` と `include "/opt/bitnami/nginx/conf/bitnami/bitnami-apps-prefix.conf";` の間に次のテキストを入力します。

   ```
   return 301 https://$host$request_uri;
   ```

   結果は次のようになります。

4. ESC キーを押して「:wq」と入力し、編集内容を書き込んで (保存して) Vim を終了します。
5. 次のコマンドを入力して基本サービスを再開し、編集内容を反映させます。

   ```
   sudo /opt/bitnami/ctlscript.sh restart
   ```

   これで、HTTP から HTTPS へ自動的に接続をリダイレクトするように Nginx インスタンスが設定されました。訪問者が `http://www.example.com` にアクセスすると、暗号化された `https://www.example.com` アドレスに自動的にリダイレクトされます。

ステップ 9: Let's Encrypt 証明書を 90 日ごとに更新する

Let's Encrypt 証明書の有効期間は 90 日間です。証明書は有効期限が切れる 30 日前から更新できます。Let's Encrypt 証明書を更新するには、取得するために使用した元のコマンドを実行します。このチュートリアルの「Let's Encrypt の SSL ワイルドカード証明書をリクエストする」セクションのステップを繰り返します。

チュートリアル: WordPress Lightsail インスタンスで SSL 証明書を暗号化しよう

> **Tip**
>
> Lightsail には、インスタンスへの Let's Encrypt 証明書のインストールと設定を自動化するガイド付きワークフローが用意されています。 WordPress このチュートリアルの手動の手順に従うのではなく、このワークフローを使用することを強くお勧めします。詳細については、「WordPress インスタンスの起動と設定」を参照してください。

Amazon Lightsail では、Lightsail ロードバランサーを使用して SSL/TLS を使用してウェブサイトやアプリケーションを簡単に保護できます。ただし、Lightsail ロードバランサーの使用は一般的に適切な選択ではない場合があります。お使いのサイトではロードバランサーが提供するスケーラビリティや耐障害性が必要ない、またはコストのために最適化しているという可能性があります。後者の場合は、Let's Encrypt で無料の SSL 証明書を入手できます。無料の証明書を使用することに問題はありません。これらの証明書は Lightsail インスタンスと統合できます。

このガイドでは、Certbot を使用して Let's Encrypt ワイルドカード証明書をリクエストする方法と、Really Simple SSL WordPress プラグインを使用してインスタンスと統合する方法を学習します。

- Bitnami インスタンスで使用されている Linux ディストリビューションは、2020 年 7 月に Ubuntu から Debian に変更されました。この変更により、このチュートリアルのいくつかのステップは、インスタンスの Linux ディストリビューションによって異なります。変更後に作成された Bitnami ブループリントインスタンスはすべて Debian Linux ディストリビューションを使用します。変更前に作成されたインスタンスは、Ubuntu Linux ディストリビューションを引き続き使用します。インスタンスのディストリビューションをチェックするには、`uname -a` コマンドを実行します。応答には、インスタンスの Linux ディストリビューションとして Ubuntu または Debian のいずれかが表示されます。
- Bitnami は多くのスタックのファイル構造を変更しました。このチュートリアルのファイルパスは、Bitnami スタックがネイティブ Linux システムパッケージを使用しているか (アプローチ A)、または自己完結型インストール (アプローチ B) であるかによって、変更される場合があります。Bitnami のインストールタイプと取るべき方法を特定するには、次のコマンドを実行します。

  ```
  test ! -f "/opt/bitnami/common/bin/openssl" && echo "Approach A: Using
  system packages." || echo "Approach B: Self-contained installation."
  ```

目次

チュートリアルを開始する前に

このチュートリアルを開始する前に、以下の点を考慮する必要があります。

Bitnami HTTPS 設定 (`bncert`) ツールを代わりに使用する

このチュートリアルに記載されている手順は、手動プロセスを使用して SSL/TLS 証明書を実装する方法を説明しています。ただし、Bitnami には、通常 Lightsail のインスタンスにプリインストールされている Bitnami HTTPS 設定 (`bncert`) ツールを使用する、より自動化されたプロセスがあります。 WordPressこのチュートリアルの手動手順を実行する代わりに、このツールを使用することが強く推奨されます。このチュートリアルは、`bncert` ツールが利用可能になる前に作成されたものです。`bncert`ツールの使用方法の詳細については、「Amazon Lightsail WordPress のインスタンスでの HTTPS の有効化」を参照してください。

インスタンスの Linux ディストリビューションを特定してください。 WordPress

Bitnami インスタンスで使用されている Linux ディストリビューションは、2020 年 7 月に Ubuntu から Debian に変更されました。変更後に作成された Bitnami ブループリントインスタンスはすべて Debian Linux ディストリビューションを使用します。変更前に作成されたインスタンスは、Ubuntu Linux ディストリビューションを引き続き使用します。この変更により、このチュートリアルのいくつかのステップは、インスタンスの Linux ディストリビューションによって異なります。このチュー

トリアルでどの手順を使用するのかを把握するために、インスタンスの Linux ディストリビューションを特定する必要があります。インスタンスのディストリビューションを特定するには、`uname -a` コマンドを実行します。応答には、インスタンスの Linux ディストリビューションとして Ubuntu または Debian のいずれかが表示されます。

インスタンスに適用されるチュートリアルアプローチを特定する

Bitnami は、多くのスタックのファイル構造を変更するプロセスです。このチュートリアルのファイルパスは、Bitnami スタックがネイティブ Linux システムパッケージを使用しているか (アプローチ A)、または自己完結型インストール (アプローチ B) であるかによって、変更される場合があります。Bitnami のインストールタイプと取るべき方法を特定するには、次のコマンドを実行します。

```
test ! -f "/opt/bitnami/common/bin/openssl" && echo "Approach A: Using system packages." || echo "Approach B: Self-contained installation."
```

ステップ 1: 前提条件を満たす

以下の前提条件を満たします (まだ満たしていない場合)。

- WordPress Lightsail でインスタンスを作成します。詳細については、「インスタンスを作成する」を参照してください。
- ドメイン名を登録し、その DNS レコードを編集するための管理アクセスを取得します。詳細については、「DNS」を参照してください。

  Lightsail DNS ゾーンを使用してドメインの DNS レコードを管理することをお勧めします。詳細については、「DNS ゾーンを作成してドメインの DNS レコードを管理する」を参照してください。
- Lightsail コンソールのブラウザベースの SSH ターミナルを使用して、このチュートリアルの手順を実行します。ただし、独自の SSH クライアント (PuTTY など) を使用することもできます。PuTTY の設定の詳細については、「Amazon Lightsail で SSH を使用して接続するための PuTTY をダウンロードしてセットアップする」を参照してください。

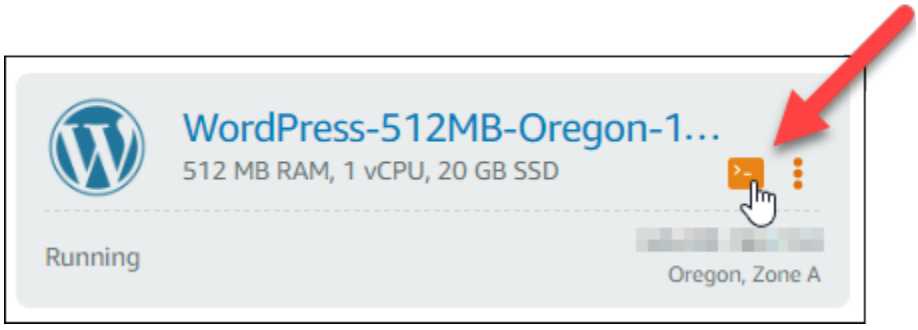
前提条件が完了したら、このチュートリアルの「次のセクション」に進みます。

ステップ 2: Lightsail インスタンスに Certbot をインストールする

Certbot は、Let's Encrypt の証明書をリクエストしてウェブサーバーにデプロイするために使用するクライアントです。Let's Encrypt は ACME プロトコルを使用して証明書を発行します。Certbot は、Let's Encrypt とやり取りする ACME 対応のクライアントです。

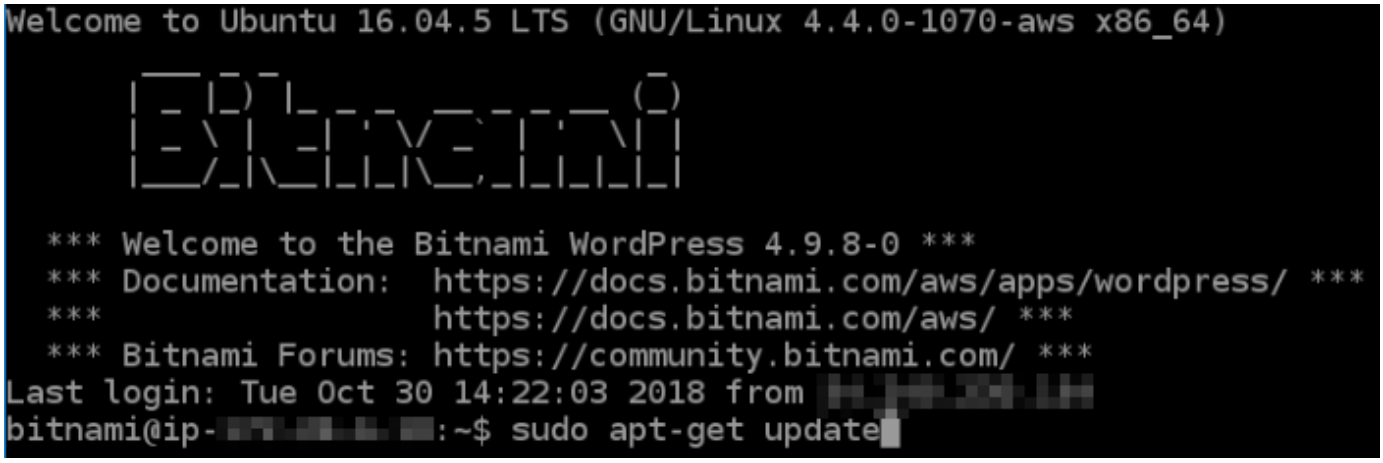
Lightsail インスタンスに Certbot をインストールするには

1. Lightsail コンソールにサインインします。
2. Lightsail ホームページで、接続するインスタンスの SSH クイック接続アイコンを選択します。



3. Lightsail ブラウザベースの SSH セッションが接続されたら、次のコマンドを入力してインスタンスのパッケージを更新します。

```
sudo apt-get update
```



4. 次のコマンドを入力してソフトウェアプロパティパッケージをインストールします。Certbot の開発者は、Personal Package Archive (PPA) を使用して Cerbot を配信します。ソフトウェアプロパティパッケージを使用すると、PPA をより効率的に操作できます。

```
sudo apt-get install software-properties-common
```

> **Note**
>
> `sudo apt-get install` コマンドを実行したときに `Could not get lock` エラーが発生した場合は、約 15 分待ってから再試行してください。このエラーは、自動アッ

プグレードをインストールするために Apt パッケージ管理ツールを使用している cron ジョブが原因で発生している可能性があります。

5. 次のコマンドを入力して GPG パッケージをインストールし、Certbot をローカルの apt リポジトリに追加します。

> **Note**
>
> ステップ 5 は、Ubuntu Linux ディストリビューションを使用するインスタンスにのみ適用されます。インスタンスが Debian Linux ディストリビューションを使用している場合は、このステップをスキップしてください。

```
sudo apt-get install gpg -y
```

```
sudo apt-add-repository ppa:certbot/certbot -y
```

6. 次のコマンドを入力して apt を更新し、新しいリポジトリを含めます。

```
sudo apt-get update -y
```

7. 次のコマンドを入力して Certbot をインストールします。

```
sudo apt-get install certbot -y
```

これで Certbot が Lightsail インスタンスにインストールされました。

8. ブラウザベースの SSH ターミナルウィンドウは開いたままにします。このチュートリアルで後ほど戻ります。このチュートリアルの「次のセクション」に進みます。

ステップ 3: Let's Encrypt の SSL ワイルドカード証明書をリクエストする

Let's Encrypt の証明書をリクエストするプロセスを開始します。Certbot を使用してワイルドカード証明書をリクエストします。この 1 つの証明書をドメインとそのサブドメインの両方に使用できます。たとえば、1 つのワイルドカード証明書を `example.com` 最上位ドメイン、`blog.example.com` サブドメイン、および `stuff.example.com` サブドメインに使用できます。

Let's Encrypt の SSL ワイルドカード証明書をリクエストするには

1. このチュートリアルのステップ 2 で使用した同じブラウザベースの SSH ターミナルウィンドウで、以下のコマンドを入力してドメインの環境変数を設定します。より効率的にコマンドをコピーして貼り付け、証明書を取得できます。*domain* は登録済みのドメイン名に置き換えてください。

   ```
   DOMAIN=domain
   ```

   ```
   WILDCARD=*.$DOMAIN
   ```

   例：

   ```
   DOMAIN=example.com
   ```

   ```
   WILDCARD=*.$DOMAIN
   ```

2. 次のコマンドを入力し、変数が正しい値を返すことを確認します。

   ```
   echo $DOMAIN && echo $WILDCARD
   ```

   次のような結果が表示されます。

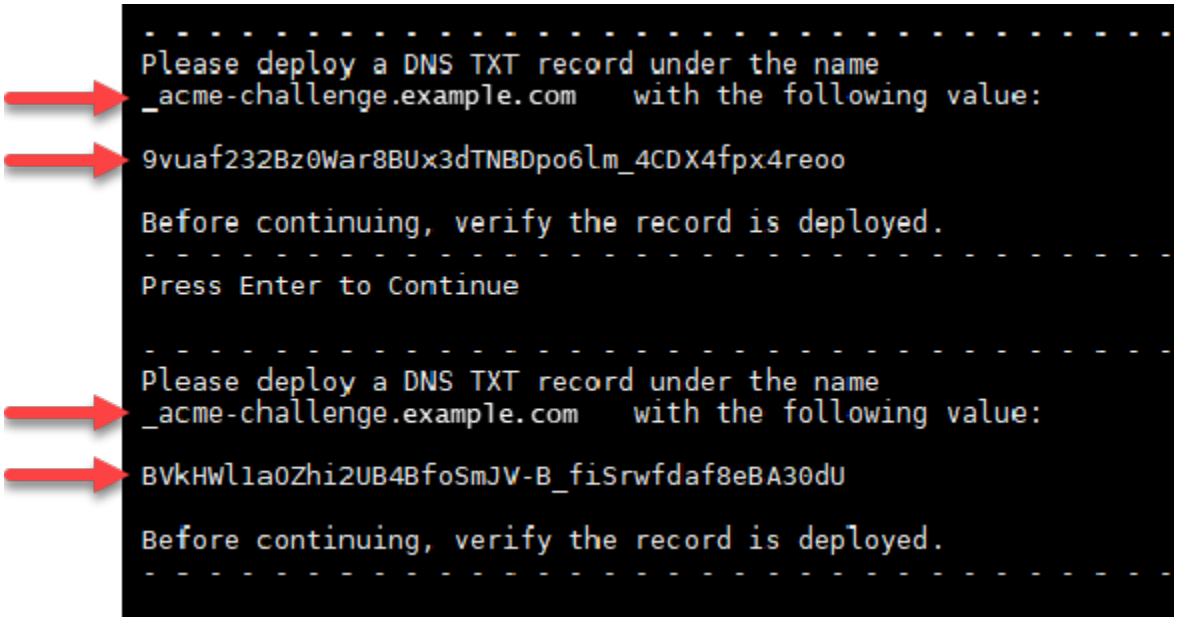
3. 次のコマンドを入力して Certbot をインタラクティブモードで起動します。このコマンドでは、DNS チャレンジで手動認証を使用してドメインの所有権を検証することを Certbot に指示します。また、最上位ドメインとそのサブドメイン用にワイルドカード証明書をリクエストします。

   ```
   sudo certbot -d $DOMAIN -d $WILDCARD --manual --preferred-challenges dns certonly
   ```

4. プロンプトに応じて E メールアドレスを入力します。これで更新とセキュリティに関する通知を受信します。

5. Let's Encrypt のサービス利用規約を読みます。読み終わり、同意する場合は A キーを押します。同意しない場合は、Let's Encrypt の証明書を取得できません。
6. E メールアドレスの共有と IP アドレスのログ記録に関するプロンプトに適宜応答します。
7. Let's Encrypt から、指定されたドメインの所有者であることの検証を求められます。これを行うには、ドメインの DNS レコードに TXT レコードを追加します。以下の例に示すように 2 組の TXT レコード値が提供されます。

> **Note**
>
> Let's Encrypt では検証に必要な TXT レコードを 1 つまたは複数提供する場合があります。この例では、検証に使用する 2 つの TXT レコードが提供されました。



8. Lightsail ブラウザーベースの SSH セッションは開いたままにしておきます。このセッションについては、このチュートリアルの後半で説明します。このチュートリアルの「次のセクション」に進みます。

ステップ 4: ドメインの DNS ゾーンに TXT レコードを追加する

ドメインの DNS ゾーンに TXT レコードを追加すると、自分がドメインを所有していることが検証されます。デモンストレーションの目的で、Lightsail DNS ゾーンを使用します。ただし、ドメインレジストラがホストする他の一般的な DNS ゾーンでも手順はほぼ同じです。

> **Note**
>
> ドメインの Lightsail DNS ゾーンを作成する方法の詳細については、「Lightsail でドメインの DNS レコードを管理するための DNS ゾーンの作成」を参照してください。

Lightsail のドメインの DNS ゾーンに TXT レコードを追加するには

1. Lightsail のホームページで、[Domains & DNS] (ドメイン & DNS) タブを選択します。
2. ページの [DNS ゾーン] セクションで、Certbot 証明書リクエストで指定したドメインの DNS ゾーンを選択します。
3. DNS ゾーンエディタで [DNS records] (DNS レコード) を選択します。
4. [レコードの追加] を選択します。
5. [Record type] (レコードタイプ) のドロップダウンメニューで [TXT record] (TXT レコード) を選択します。
6. Let's Encrypt 証明書のリクエストで指定された値を [Record name] (レコード名) と [Responds with] (応答) フィールドに入力します。

   > **Note**
   >
   > Lightsail コンソールは、ドメインの頂点部分を事前に入力します。たとえば、*_acme-challenge.example.com* サブドメインを追加する場合は、*_acme-challenge* テキストボックスに入力するだけで、レコードを保存するときに Lightsail が `.example.com` の部分を追加します。

7. [保存] を選択します。
8. ステップ 4 ~ 7 を繰り返して、Let's Encrypt の証明書リクエストで指定された 2 番目の TXT レコードのセットを追加します。
9. Lightsail コンソールのブラウザウィンドウは開いたままにしておきます。このチュートリアルの後半で再び開きます。このチュートリアルの「次のセクション」に進みます。

ステップ 5: TXT レコードが反映されたことを確認する

MxToolbox ユーティリティを使用して、TXT レコードがインターネットの DNS に伝播したことを確認します。DNS レコードの反映には、DNS ホスティングプロバイダーと DNS レコードの有効期限 (TTL) の設定によって時間がかかる場合があります。このステップを完了し、TXT レコードが反映さ

れたことを確認した上で、Certbot 証明書のリクエストに進むことが重要です。そうしないと、証明書のリクエストは失敗します。

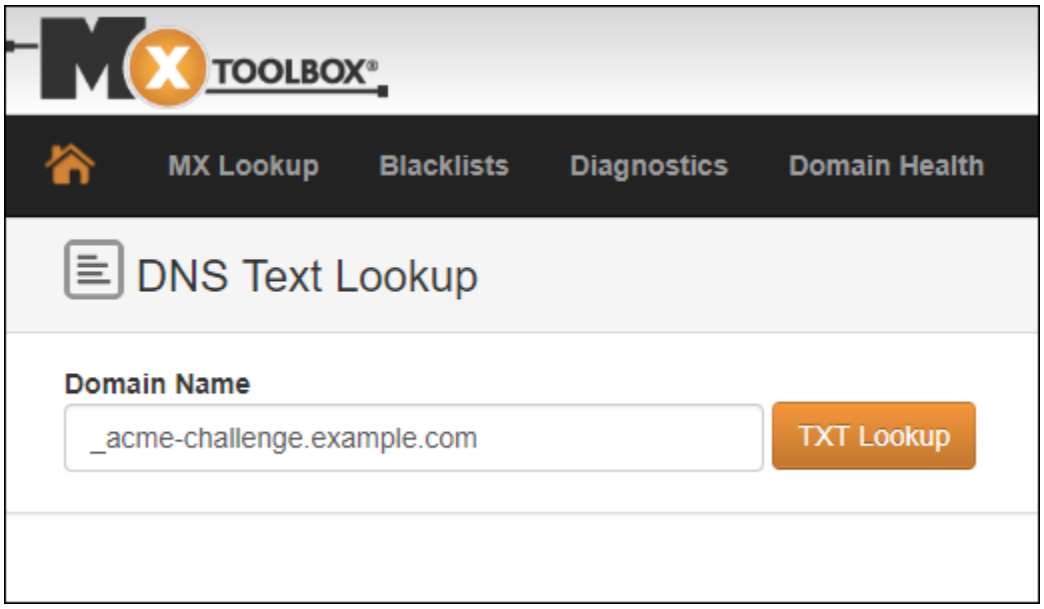
TXT レコードがインターネットの DNS に反映されたことを確認するには

1. 新しいブラウザウィンドウを開き、https://mxtoolbox.com/TXTLookup.aspx に移動します。
2. 次の内容をテキストボックスに入力します。*domain* は実際のドメインに置き換えてください。

   ```
   _acme-challenge.domain
   ```

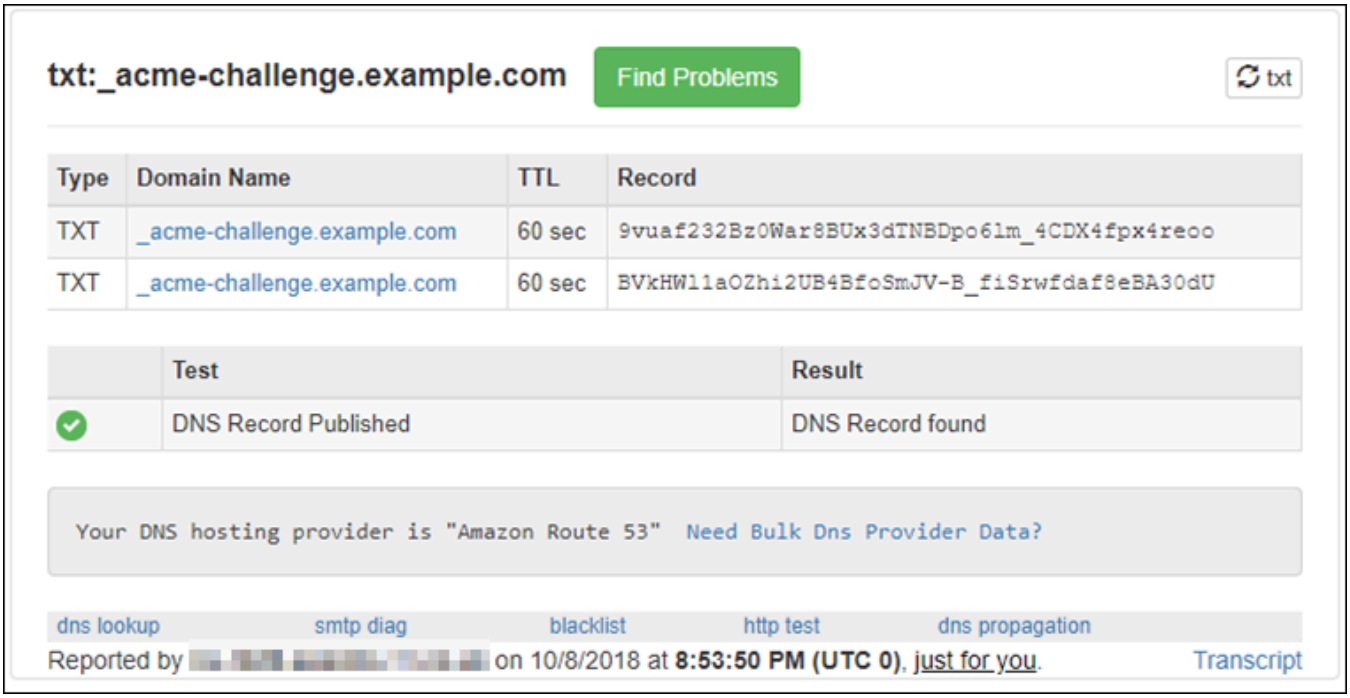
   例：

   ```
   _acme-challenge.example.com
   ```



3. [TXT Lookup (TXT ルックアップ)] を選択して確認を行います。
4. 以下のいずれかのレスポンスが返されます。

   - TXT レコードがインターネットの DNS に反映された場合は、次のスクリーンショットに示すようなレスポンスが表示されます。ブラウザウィンドウを閉じて、このチュートリアルの「次のセクション」に進みます。



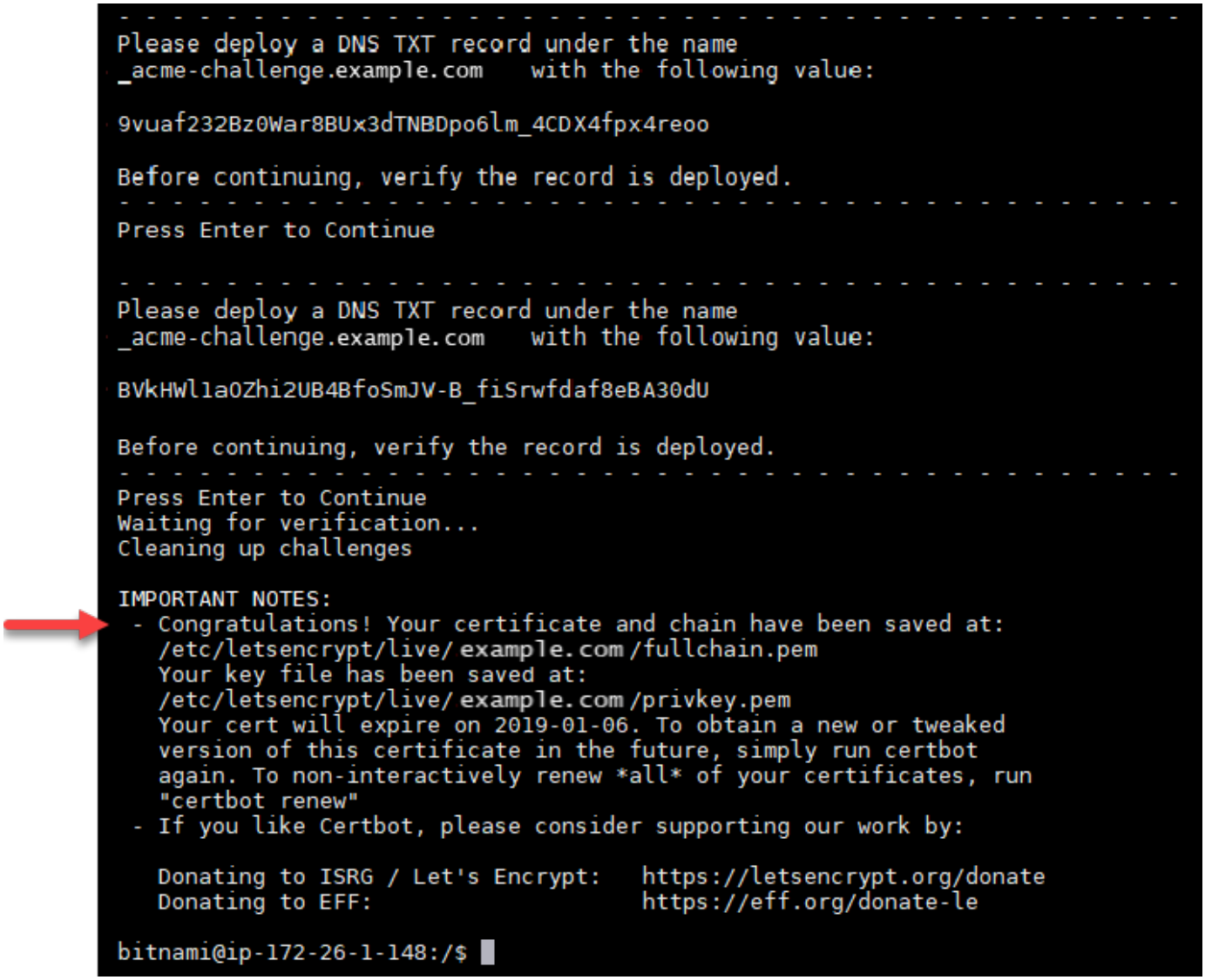
- TXT レコードがインターネットの DNS に反映されていない場合は、[DNS Record not found (DNS レコードが見つかりません)] というレスポンスが返されます。適切な DNS レコードをドメインの DNS ゾーンに追加したことを確認してください。適切なレコードを追加した場合は、ドメインの DNS レコードが反映されるまでしばらく待ってから、TXT のルックアップを再実行します。

ステップ 6: Let's Encrypt の SSL 証明書リクエストを完了する

WordPress インスタンスの Lightsail ブラウザーベースの SSH セッションに戻り、Let's Encrypt 証明書リクエストを完了します。Certbot は SSL 証明書、チェーン、キーファイルをインスタンスの特定のディレクトリに保存します。 WordPress

Let's Encrypt の SSL 証明書リクエストを完了するには

1. WordPress インスタンスの Lightsail ブラウザーベースの SSH セッションで Enter キーを押して Let's Encrypt SSL 証明書リクエストを続行します。成功すると、次のスクリーンショットに示すようなレスポンスが表示されます。



証明書、チェーン、およびキーファイルが /etc/letsencrypt/live/*domain*/ ディレクトリに保存されたことを確認するメッセージが表示されます。*domain* は、実際のドメイン (/etc/letsencrypt/live/*example.com*/ など) に置き換えてください。

2. メッセージに記載されている有効期限を書き留めておきます。この期限日までに証明書を更新する必要があります。

```
IMPORTANT NOTES:
 - Congratulations! Your certificate and chain have been saved at:
   /etc/letsencrypt/live/example.com/fullchain.pem
   Your key file has been saved at:
   /etc/letsencrypt/live/example.com/privkey.pem
   Your cert will expire on 2019-01-06. To obtain a new or tweaked
   version of this certificate in the future, simply run certbot
   again. To non-interactively renew *all* of your certificates, run
   "certbot renew"
 - If you like Certbot, please consider supporting our work by:

   Donating to ISRG / Let's Encrypt:   https://letsencrypt.org/donate
   Donating to EFF:                    https://eff.org/donate-le
```

3. これで Let's Encrypt SSL 証明書が手に入ったので、このチュートリアルの「次のセクション」に進みます。

ステップ 7: Apache サーバーディレクトリで Let's Encrypt の証明書ファイルへのリンクを作成する

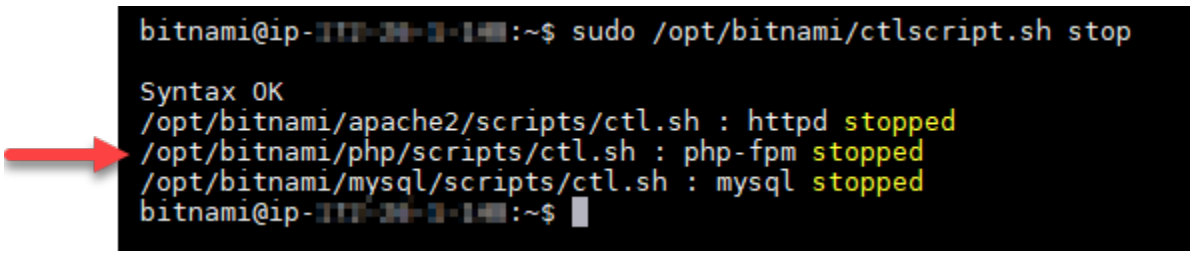
インスタンスの Apache サーバーディレクトリにある Let's Encrypt SSL 証明書ファイルへのリンクを作成します。 WordPress また、必要になる場合に備えて既存の証明書をバックアップします。

Apache サーバーディレクトリで Let's Encrypt の証明書ファイルへのリンクを作成するには

1. WordPress インスタンスの Lightsail ブラウザベースの SSH セッションで、以下のコマンドを入力して基盤となるサービスを停止します。

   ```
   sudo /opt/bitnami/ctlscript.sh stop
   ```

   次のようなレスポンスが表示されます。

2. 次のコマンドを入力してドメインの環境変数を設定します。コマンドのコピー＆ペーストで、より効率的に証明書ファイルにリンクを張れます。*domain* を登録済みのドメイン名に置き換えます。

```
DOMAIN=domain
```

例：

```
DOMAIN=example.com
```

3. 次のコマンドを入力し、変数が正しい値を返すことを確認します。

```
echo $DOMAIN
```

次のような結果が表示されます。

4. バックアップとして既存の証明書ファイルがある場合、以下のコマンドを個別に入力して名前を書き換えます。さまざまなディストリビューションとファイル構造の詳細については、このチュートリアルの冒頭の重要ブロックを参照してください。

- Debian Linux ディストリビューションの場合

  アプローチ A (システムパッケージを使用した Bitnami インストール):

```
sudo mv /opt/bitnami/apache2/conf/bitnami/certs/server.crt /opt/bitnami/apache2/
conf/bitnami/certs/server.crt.old
```

```
sudo mv /opt/bitnami/apache2/conf/bitnami/certs/server.key /opt/bitnami/apache2/
conf/bitnami/certs/server.key.old
```

  アプローチ B (自己完結型 Bitnami インストール):

```
sudo mv /opt/bitnami/apache2/conf/server.crt /opt/bitnami/apache2/conf/
server.crt.old
```

```
sudo mv /opt/bitnami/apache2/conf/server.key /opt/bitnami/apache2/conf/
server.key.old
```

- Ubuntu Linux ディストリビューションを使用する古いインスタンスの場合：

```
sudo mv /opt/bitnami/apache/conf/bitnami/certs/server.crt /opt/bitnami/apache/
conf/bitnami/certs/server.crt.old
```

```
sudo mv /opt/bitnami/apache/conf/bitnami/certs/server.key /opt/bitnami/apache/
conf/bitnami/certs/server.key.old
```

```
sudo mv /opt/bitnami/apache/conf/bitnami/certs/server.csr /opt/bitnami/apache/
conf/bitnami/certs/server.csr.old
```

5. 以下のコマンドを個別に入力し、Apache ディレクトリで Let's Encrypt の証明書ファイルへのリンクを作成します。さまざまなディストリビューションとファイル構造の詳細については、このチュートリアルの冒頭の重要ブロックを参照してください。

   - Debian Linux ディストリビューションの場合

     アプローチ A (システムパッケージを使用した Bitnami インストール):

     ```
     sudo ln -sf /etc/letsencrypt/live/$DOMAIN/privkey.pem /opt/bitnami/apache2/conf/
     bitnami/certs/server.key
     ```

     ```
     sudo ln -sf /etc/letsencrypt/live/$DOMAIN/fullchain.pem /opt/bitnami/apache2/
     conf/bitnami/certs/server.crt
     ```

     アプローチ B (自己完結型 Bitnami インストール):

     ```
     sudo ln -sf /etc/letsencrypt/live/$DOMAIN/privkey.pem /opt/bitnami/apache2/conf/
     server.key
     ```

     ```
     sudo ln -sf /etc/letsencrypt/live/$DOMAIN/fullchain.pem /opt/bitnami/apache2/
     conf/server.crt
     ```

   - Ubuntu Linux ディストリビューションを使用した古いインスタンスの場合:

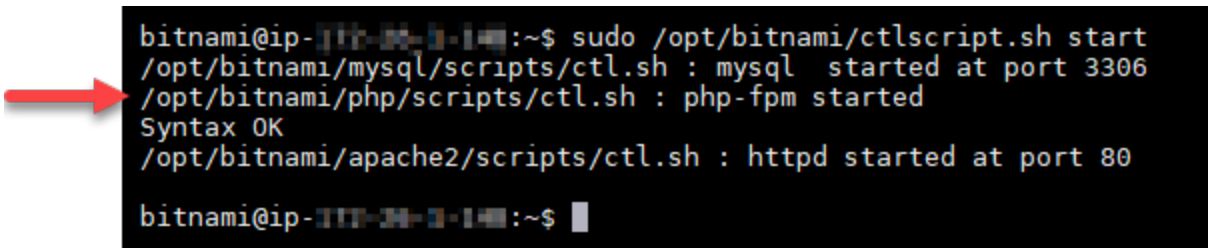
     ```
     sudo ln -s /etc/letsencrypt/live/$DOMAIN/privkey.pem /opt/bitnami/apache/conf/
     bitnami/certs/server.key
     ```

```
sudo ln -s /etc/letsencrypt/live/$DOMAIN/fullchain.pem /opt/bitnami/apache/conf/
bitnami/certs/server.crt
```

6. 次のコマンドを入力して、先ほど停止した基本サービスを開始します。

```
sudo /opt/bitnami/ctlscript.sh start
```

次のような結果が表示されます。



これで、 WordPress インスタンスの SSL 証明書ファイルが正しいディレクトリに配置されました。

7. このチュートリアルの「次のセクション」に進みます。

ステップ 8: Really Simple SSL プラグインを使用して SSL WordPress 証明書をサイトと統合します。

Really Simple SSL WordPress プラグインをサイトにインストールし、それを使用して SSL 証明書を統合します。Really Simple SSL では、サイトを訪問するユーザーが常に HTTPS 接続を利用できるように、HTTP から HTTPS へのリダイレクトも設定します。

Really Simple SSL プラグインを使用して SSL WordPress 証明書をサイトと統合するには

1. WordPress インスタンスの Lightsail ブラウザーベースの SSH セッションで、次のコマンドを入力して、wp-config.phphtaccess.confおよびファイルを書き込み可能に設定します。Really Simple SSL プラグインは、wp-config.php ファイルに書き込むことで証明書を設定します。

   - Debian Linux ディストリビューションを使用する新しいインスタンスの場合：

```
sudo chmod 666 /opt/bitnami/wordpress/wp-config.php && sudo chmod 666 /opt/
bitnami/apache/conf/vhosts/htaccess/wordpress-htaccess.conf
```

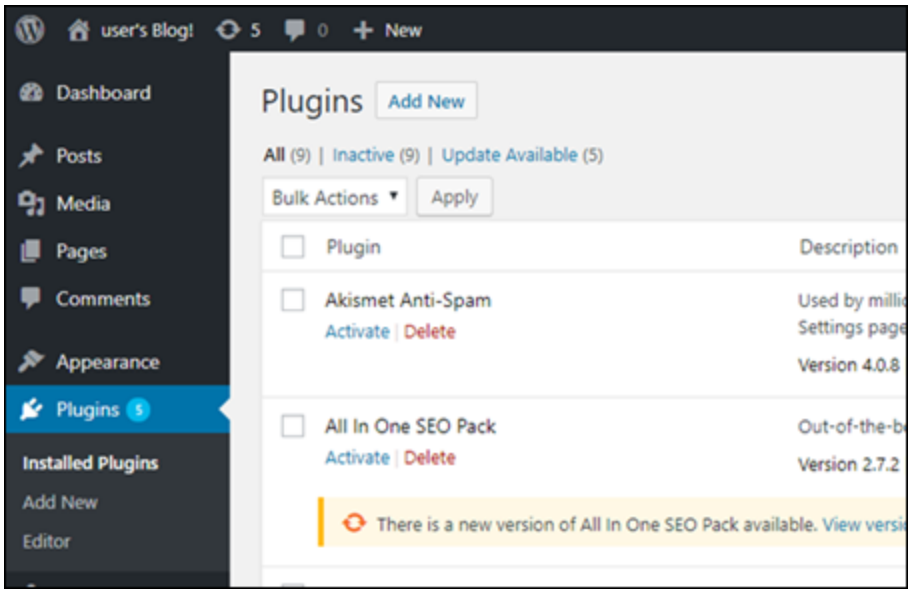
- Ubuntu Linux ディストリビューションを使用する古いインスタンスの場合：

```
sudo chmod 666 /opt/bitnami/apps/wordpress/htdocs/wp-config.php && sudo chmod
 666 /opt/bitnami/apps/wordpress/conf/htaccess.conf
```

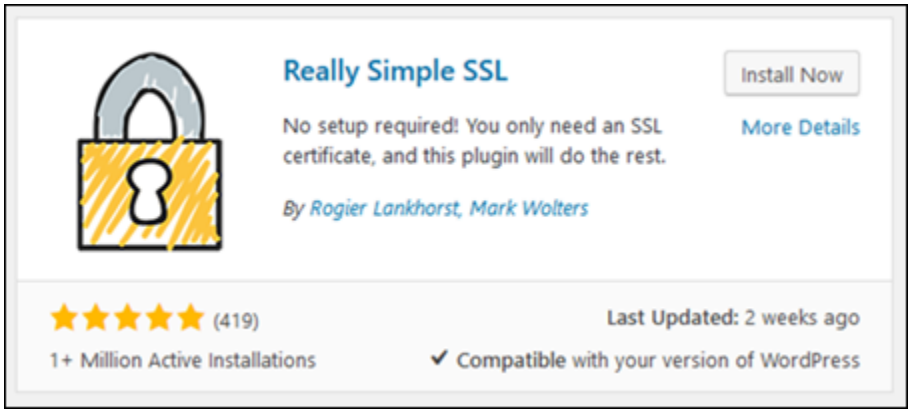
2. 新しいブラウザウィンドウを開き、インスタンスの管理ダッシュボードにサインインします。WordPress

> **Note**
>
> 詳細については、「Amazon Lightsail での Bitnami インスタンスのアプリケーションユーザー名とパスワードの取得」を参照してください。

3. 左のナビゲーションペインから、[Plugins] (プラグイン) を選択します。
4. プラグインページの上部で、[Add New] (新規追加) を選択します。



5. [Really Simple SSL] を探します。
6. 検索結果の Really Simple SSL プラグインの横にある [Install Now (今すぐインストール)] を選択します。



7. インストールが完了したら、[Activate] (有効化) を選択します。
8. 表示されるプロンプトで [Go ahead, activate SSL!] (SSL の有効化を開始!) を選択します。WordPressインスタンスの管理ダッシュボードのサインインページにリダイレクトされる場合があります。

   これで、 WordPress インスタンスは SSL 暗号化を使用するように設定されました。さらに、接続を HTTP から HTTPS WordPress に自動的にリダイレクトするようにインスタンスが設定されました。訪問者が `http://example.com` にアクセスすると、暗号化された HTTPS 接続 (`https://example.com`) に自動的にリダイレクトされます。

ステップ 9: Let's Encrypt 証明書を 90 日ごとに更新する

Let's Encrypt 証明書の有効期間は 90 日間です。証明書は有効期限が切れる 30 日前から更新できます。Let's Encrypt 証明書を更新するには、取得するために使用した元のコマンドを実行します。このチュートリアルの「Let's Encrypt の SSL ワイルドカード証明書をリクエストする」セクションのステップを繰り返します。

Amazon Lightsail に関するネットワーキングのチュートリアル

以下のネットワーキングチュートリアルでは、Amazon VPC ピアリングの設定やリバース DNS の設定などの Lightsail 関連トピックについて説明します。

トピック

- Lightsail の cPanel インスタンスで IPv6 を設定する
- Lightsail の Debian 8 インスタンスで IPv6 を設定する
- Lightsail の GitLab インスタンス用に IPv6 を設定する
- Lightsail の Nginx インスタンスで IPv6 を設定する

- Lightsail の Plesk インスタンスで IPv6 を設定する
- Lightsail で Ubuntu 16 インスタンス用に IPv6 を設定する

Lightsail の cPanel インスタンスで IPv6 を設定する

Amazon Lightsail のすべてのインスタンスには、デフォルトでパブリック IPv4 アドレスとプライベート IPv4 アドレスが割り当てられています。オプションで、インスタンスの IPv6 を有効にして、パブリック IPv6 アドレスを割り当てることができます。詳細については、「Amazon Lightsail IP アドレス」および「IPv6 を有効または無効にする」を参照してください。

cPanel と WHM ブループリントを使用するインスタンスで IPv6 を有効化した後、インスタンスに IPv6 アドレスを認識させるために追加のステップを実行する必要があります。このガイドでは、cPanel と WHM インスタンスで実行する必要がある追加の手順を説明します。

前提条件

以下の前提条件を完了します (まだの場合)。

- Lightsail で cPanel と WHM インスタンスを作成します。詳細については、「インスタンスを作成する」を参照してください。
- cPanel と WHM インスタンスを設定します。詳細については、Amazon Lightsail の「クイックスタートガイド: cPanel & WHM Amazon Lightsail」を参照してください。

> **Important**
>
> このガイドの手順を続行する前に、すべてのソフトウェアの更新と必要なシステムの再起動が実行されていることを確認してください。

- cPanel と WHMインスタンスの IPv6 を有効化します。詳細については、「IPv6 を有効化または無効化する」を参照してください。

> **Note**
>
> 2021 年 1 月 12 日以降に作成された新しい cPanel と WHM インスタンスでは、Lightsail コンソールで作成されるときに IPv6 がデフォルトで有効になっています。インスタンスの作成時に IPv6 がデフォルトで有効になっていても、インスタンスで IPv6 を設定するには、このガイドの以下の手順を実行する必要があります。

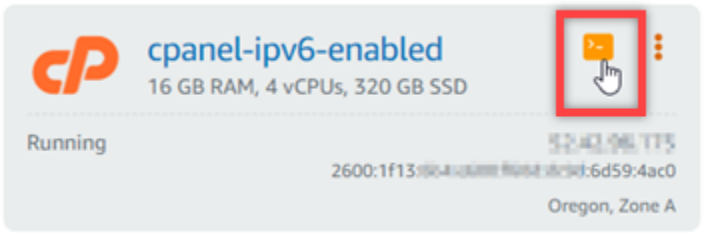
cPanel と WHM インスタンスで Pv6 を設定する。

Lightsail の cPanel と WHM インスタンス上で IPv6 を設定するには、以下の手順を実行します。

1. Lightsail コンソールにサインインします。
2.

> **Important**
>
> Lightsail ブラウザベースの SSH/RDP クライアントは IPv4 トラフィックのみを受け入れます。サードパーティーのクライアントを使用して、IPv6 経由でインスタンスに SSH または RDP 接続します。詳細については、「インスタンスに接続します」を参照してください。

   Lightsail ホームページのインスタンスセクションで、設定する cPanel & WHM インスタンスを探し、ブラウザベースの SSH クライアントアイコンを選択して SSH を使用して接続します。



3. インスタンスに接続後、次のコマンドを入力して`ifcfg-eth0` ネットワークインターフェース設定ファイルを Nano を使用して開きます。

```
sudo nano /etc/sysconfig/network-scripts/ifcfg-eth0
```

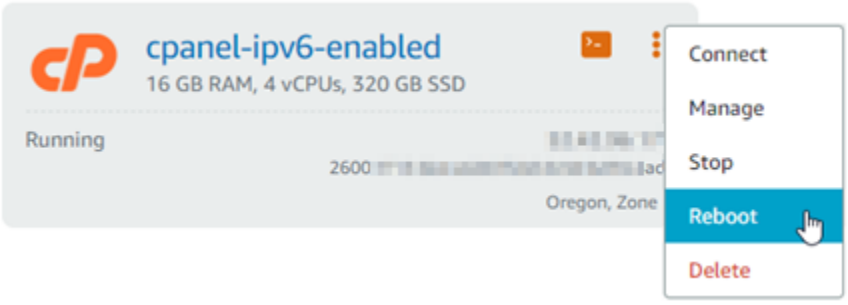
4. ファイルにテキストが追加されていない場合、次のテキストを追加します。

```
IPV6INIT=yes
IPV6_AUTOCONF=yes
DHCPV6C=yes
```

   結果は次の例のようになります。

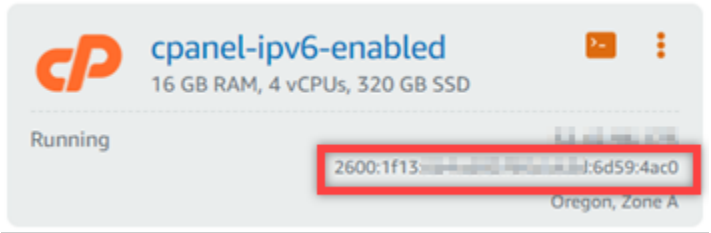
```
# Automatically generated by the vm import process
TYPE=Ethernet
PROXY_METHOD=none
BROWSER_ONLY=no
BOOTPROTO=dhcp
DEFROUTE=yes
IPV4_FAILURE_FATAL=no
NAME=eth0
DEVICE=eth0
ONBOOT=yes
IPV6INIT=yes
IPV6_FAILURE_FATAL=no
DHCPV6C=yes
IPV6_AUTOCONF=yes
```

5. CTRL+C を押してファイルを終了します。
6. Y を修正したバッファを保存するプロンプトが表示されたら押します。Enter を押してして既存のファイルに保存します。これにより、`ifcfg-eth0`ネットワークインターフェース設定ファイルに編集が保存されます。
7. ブラウザベースの SSH ウィンドウを閉じ、Lightsail コンソールに戻ります。
8. Lightsail ホームページの [インスタンス] タブで、cPanel と WHM インスタンスのアクションメニュー (⋮) を選択し、[再起動] を選択します。



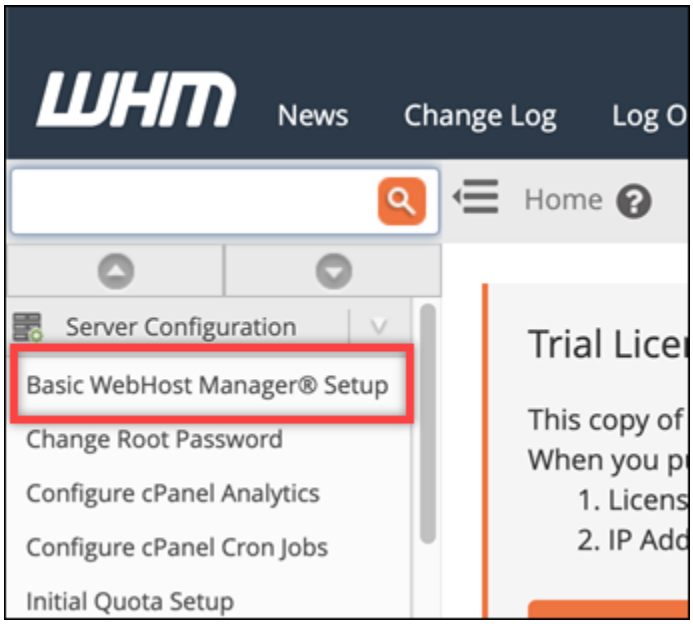
インスタンスが再起動するまで数分待ってから、次のステップに進みます。

9. Lightsail ホームページの [インスタンス] タブで、cPanel と WHM インスタンスに割り当てられた IPv6 アドレスを確認します。

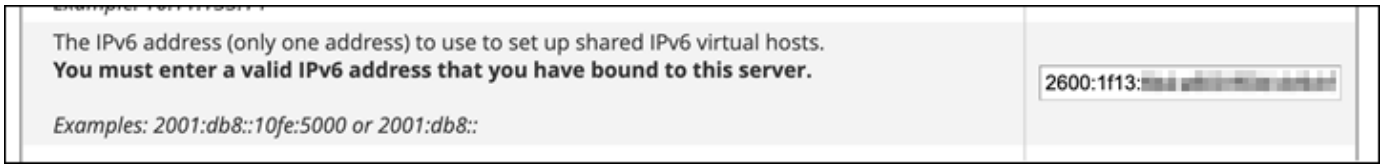


10. 新しいブラウザタブを開き、cPanel と WHM インスタンスのウェブホストマネージャ (WHM) にサインインします。

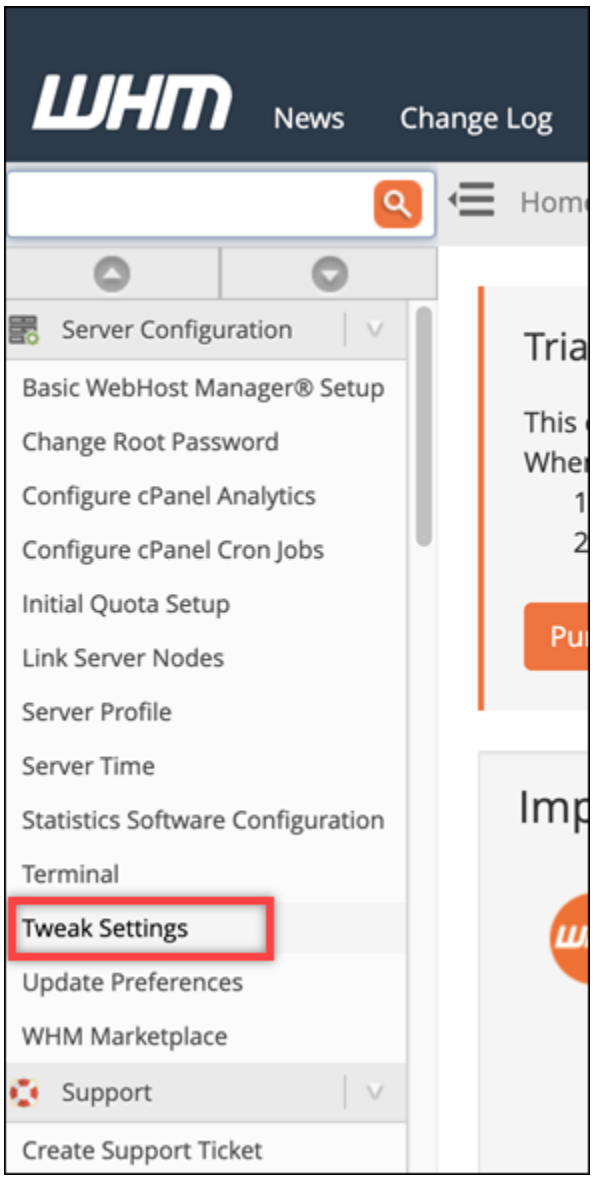
11. WHM コンソールの左側のナビゲーションペインで、「Basic WebHost Manager Setup」を選択します。



12. [All] (すべて)タブで、使用する IPv6 アドレスのテキストを探して、インスタンスに割り当てられている IPv6 アドレスを入力します。手順のステップ 9 からインスタンスに割り当てられた IPv6 アドレスを書き留めておく必要があります。

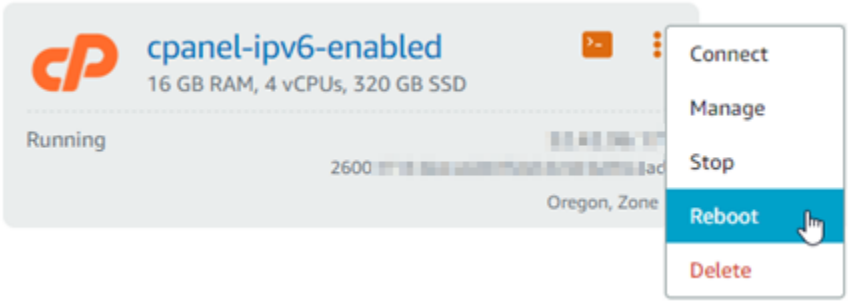


13. ページの最下部までスクロールし、[変更の保存] を選択します。
14. WHM コンソールの左のナビゲーションペインで [設定を微調整する] を選択します。



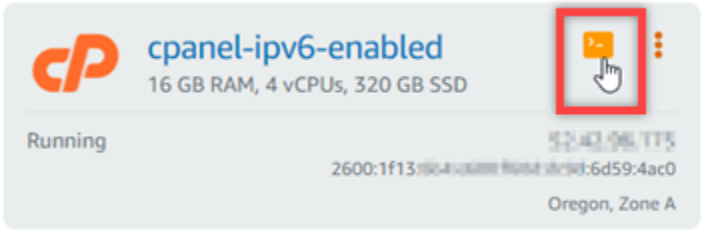
15. [All] (すべて)タブで、下にスクロールしてIPv6 アドレスをリッスンする設定を探して、Onに設定します。

16. ページの下部にスクロールし、保存を選択します。
17. Lightsail コンソールに戻ります。
18. Lightsail ホームページの [インスタンス] タブで、cPanel と WHM インスタンスのアクションメニュー (⋮) を選択し、[再起動] を選択します。



インスタンスが再起動するまで数分待ってから、次のステップに進みます。

19. SSH を使用して接続する cPanel と WHM インスタンスのブラウザベースの SSH クライアントアイコンを選択します。



20. インスタンスに接続後、以下のコマンドを入力して、インスタンスに設定された IP アドレスを表示し、割り当てられた IPv6 アドレスを認識していることを確認します。

```
ip addr
```

次のようなレスポンスが表示されます。インスタンスが IPv6 アドレスを認識している場合はこの例のように、レスポンスに [scope global] のラベルでリスト表示されます。

```
[centos@ ~]$ ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 9001 qdisc mq state UP group default qlen 1000
    link/ether 02:9b:51:92:50:45 brd ff:ff:ff:ff:ff:ff
    inet /20 brd  scope global dynamic eth0
    inet6 2600:1f13:  :6d59:4ac0/128 scope global dynamic
    inet6 fe80:  :5045/64 scope link
       valid_lft forever preferred_lft forever
```

21. 以下のコマンドを入力して、インスタンスが IPv6 アドレスに ping されているかを確認します。

```
ping6 ipv6.google.com -c 6
```

結果は、以下の例のように表示され、インスタンスが IPv6 アドレスに ping されていることが確認できます。

```
[centos@52-42-98-179 ~]$ ping6 ipv6.google.com
PING ipv6.google.com(sea15s12-in-x0e.1e100.net (2607:f8b0:400a:809::200e)) 56 data bytes
64 bytes from sea15s12-in-x0e.1e100.net (2607:f8b0:400a:809::200e): icmp_seq=1 ttl=103 tim
e=7.66 ms
64 bytes from sea15s12-in-x0e.1e100.net (2607:f8b0:400a:809::200e): icmp_seq=2 ttl=103 tim
e=7.70 ms
64 bytes from sea15s12-in-x0e.1e100.net (2607:f8b0:400a:809::200e): icmp_seq=3 ttl=103 tim
e=7.68 ms
64 bytes from sea15s12-in-x0e.1e100.net (2607:f8b0:400a:809::200e): icmp_seq=4 ttl=103 tim
e=7.69 ms
64 bytes from sea15s12-in-x0e.1e100.net (2607:f8b0:400a:809::200e): icmp_seq=5 ttl=103 tim
e=7.70 ms
64 bytes from sea15s12-in-x0e.1e100.net (2607:f8b0:400a:809::200e): icmp_seq=6 ttl=103 tim
e=7.68 ms
^C
--- ipv6.google.com ping statistics ---
6 packets transmitted, 6 received, 0% packet loss, time 5008ms
rtt min/avg/max/mdev = 7.667/7.690/7.702/0.052 ms
```

Lightsail の Debian 8 インスタンスで IPv6 を設定する

Amazon Lightsail のすべてのインスタンスには、デフォルトでパブリック IPv4 アドレスとプライベート IPv4 アドレスが割り当てられています。オプションで、インスタンスの IPv6 を有効にして、パブリック IPv6 アドレスを割り当てることができます。詳細については、「Amazon Lightsail IP アドレス」および「IPv6 を有効または無効にする」を参照してください。

Debian 8 ブループリントを使用するインスタンスで IPv6 を有効化した後、インスタンスに IPv6 アドレスを認識させるために追加のステップを実行する必要があります。このガイドでは、Debian 8 インスタンスで実行しなければいけないステップを説明します。

前提条件

以下の前提条件を完了します (まだの場合)。

- Lightsail で Debian 8 インスタンスを作成します。詳細については、「インスタンスを作成する」を参照してください。
- Debian 8 インスタンスで IPv6 を有効にします。詳細については、「IPv6 を有効化または無効化する」を参照してください。

> **Note**
>
> 2021 年 1 月 12 日以降に作成された新しい Debian インスタンスは Lightsail コンソールで作成されている場合、IPv6 がデフォルトで有効になっています。インスタンスの作成時に IPv6 がデフォルトで有効になっていても、インスタンスで IPv6 を設定するには、このガイドの以下のステップを実行する必要があります。

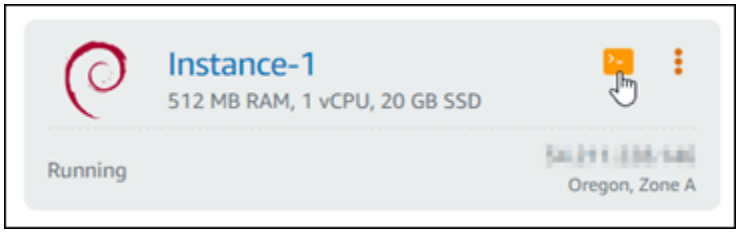
Debian 8 インスタンスで IPv6 を設定する

Lightsail の Debian 8 インスタンスで IPv6 を設定するには、以下のステップを実行します。

1. Lightsail コンソールにサインインします。
2.

> **Important**
>
> Lightsail ブラウザベースの SSH/RDP クライアントは IPv4 トラフィックのみを受け入れます。サードパーティーのクライアントを使用して、IPv6 経由でインスタンスに SSH または RDP 接続します。詳細については、「インスタンスに接続します」を参照してください。

   Lightsail ホームページのインスタンスセクションで、設定する Debian 8 インスタンスを探し、ブラウザベースの SSH クライアントアイコンを選択して SSH を使用して接続します。

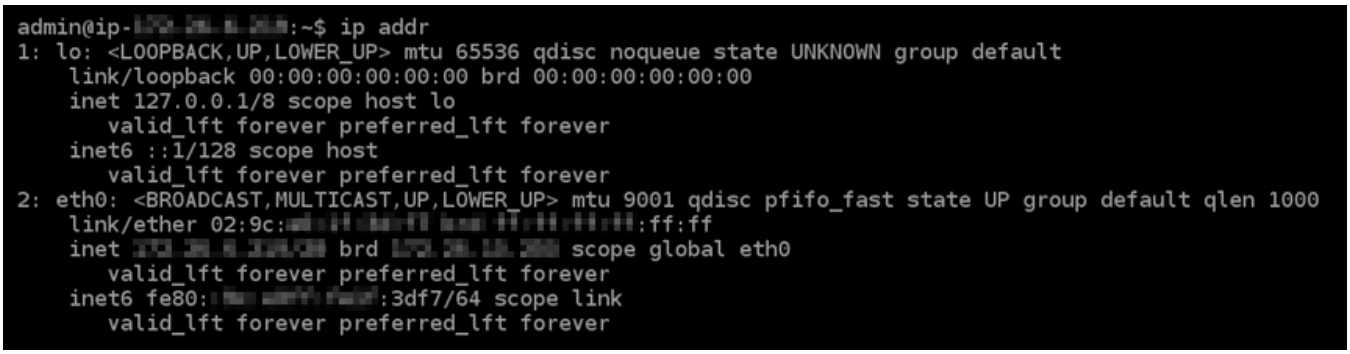


3. インスタンスに接続したら、以下のコマンドを入力し、インスタンスに設定されている IP アドレスを表示します。

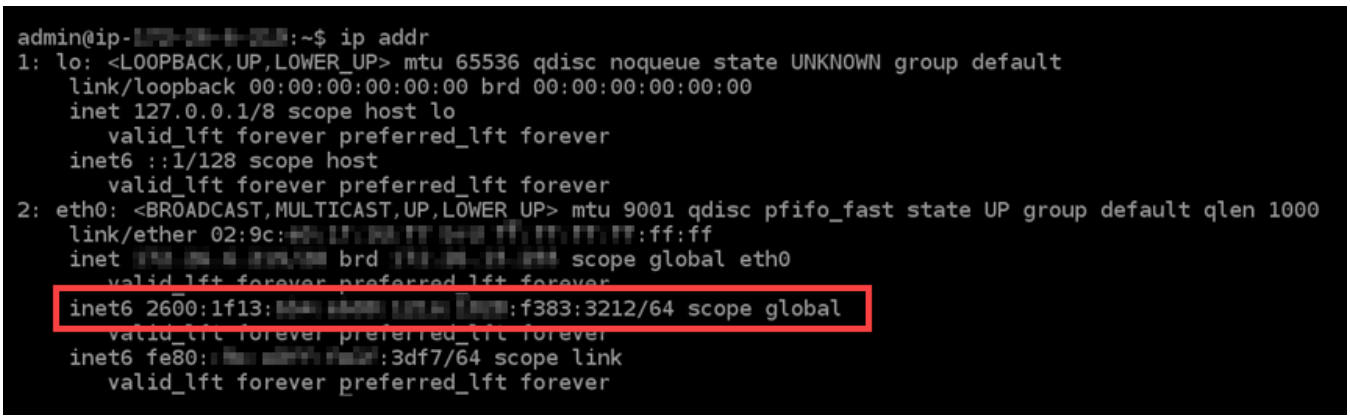
```
ip addr
```

   以下のようなレスポンスが表示されます。

- インスタンスが IPv6 アドレスを認識しない場合、レスポンスに表示されません。この手順のステップ 4～9 を続行する必要があります。



- インスタンスが IPv6 アドレスを認識している場合は、レスポンスに `scope global` の例のように表示されます。ここで停止します。インスタンスが IPv6 アドレスを認識するように設定済みなので、この手順のステップ 4 ～ 9 を実行する必要はありません。



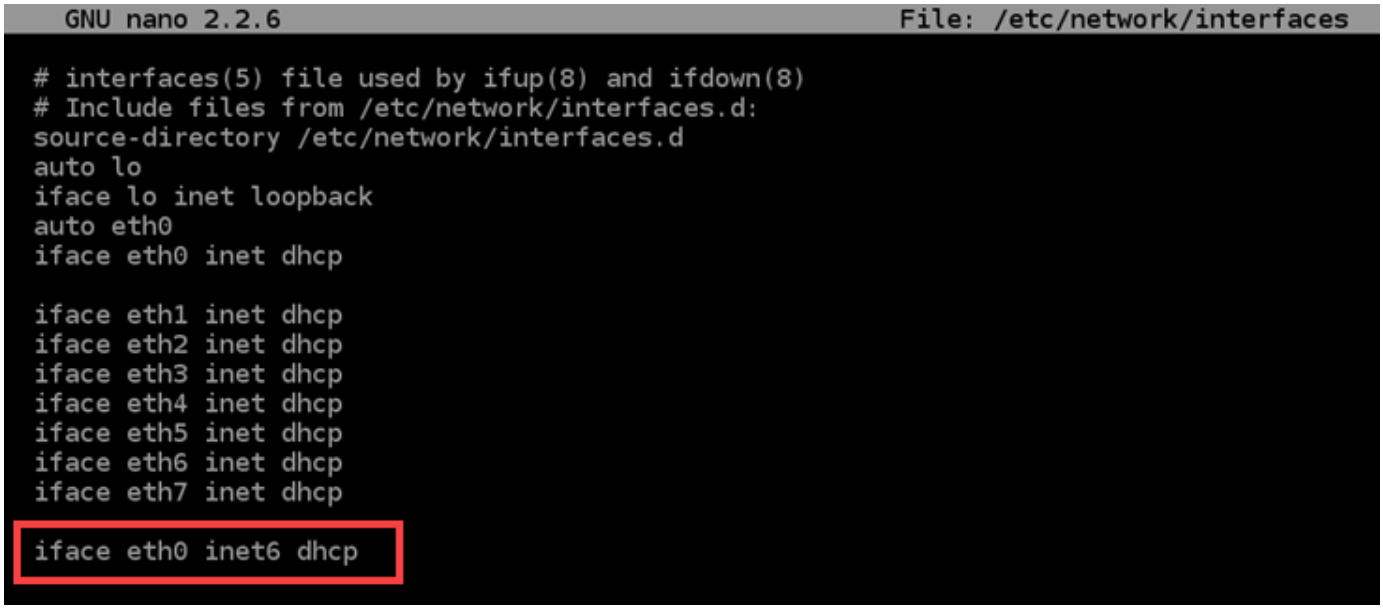
4. 以下のコマンドを入力して、nano を使用して`interfaces`設定ファイルを開きます。

```
sudo nano /etc/network/interfaces
```

5. ファイルの末尾に次の行を追加します。

```
iface eth0 inet6 dhcp
```

完了したファイルは以下のようになります。



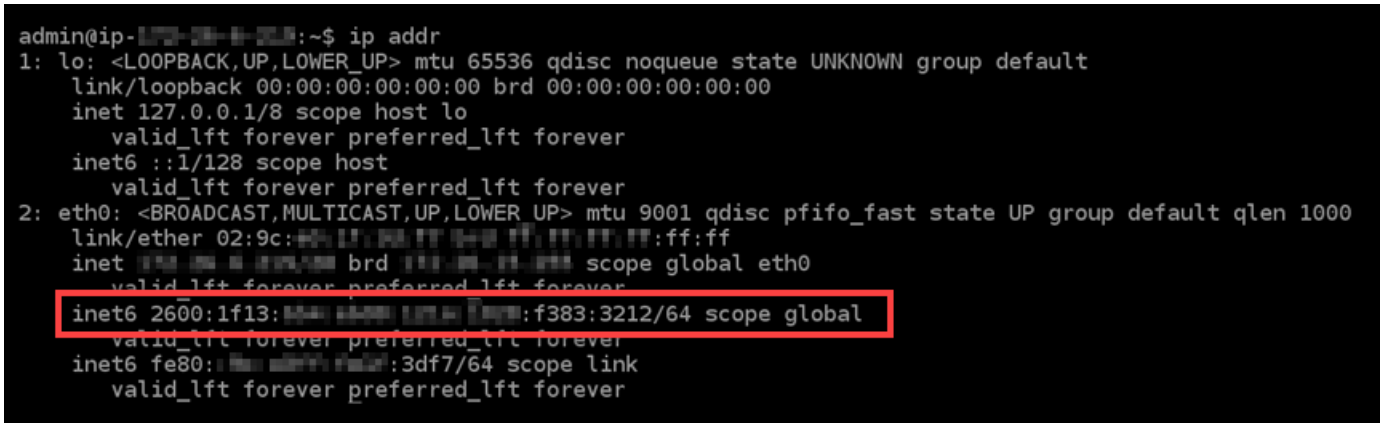
6. Ctrl+Escキーを押して nano を終了します。
7. 変更バッファーを保存するか、しないかと表示がでたらYを押してから入力を押し、既存のインターフェース設定ファイルに保存します。
8. 以下のコマンドを入力して、インスタンス上のネットワークサービスを再起動します。

   ```
   sudo systemctl restart networking
   ```

   インスタンスのネットワークサービスを再起動した後、インスタンスが IPv6 アドレスを認識するまで数分待たなければならない場合があります。
9. 以下のコマンドを入力して、インスタンスに設定された IP アドレスを表示し、割り当てられた IPv6 アドレスを認識していることを確認します。

   ```
   ip addr
   ```

   以下のようなレスポンスが表示されます。インスタンスが IPv6 アドレスを認識している場合は、この例にあるようにscope globalのラベル付けと一緒にレスポンスに表示されます。



Lightsail の GitLab インスタンス用に IPv6 を設定する

Amazon Lightsail のすべてのインスタンスには、デフォルトでパブリック IPv4 アドレスとプライベート IPv4 アドレスが割り当てられます。オプションで、インスタンスの IPv6 を有効にして、パブリック IPv6 アドレスを割り当てることができます。詳細については、「Amazon Lightsail IP アドレス」および「IPv6 を有効または無効にする」を参照してください。

GitLab ブループリントを使用するインスタンスで IPv6 を有効にした後、追加のステップを実行して、インスタンスに IPv6 アドレスを認識させる必要があります。このガイドでは、 GitLab インスタンスに対して実行する必要がある追加の手順について説明します。

前提条件

以下の前提条件を完了します (まだの場合)。

- Lightsail で GitLab インスタンスを作成します。詳細については、「インスタンスを作成する」を参照してください。
- GitLab インスタンスの IPv6 を有効にします。詳細については、「IPv6 を有効化または無効化する」を参照してください。

> **Note**
>
> 2021 年 1 月 12 日以降に作成された新しい GitLab インスタンスは、Lightsail コンソールで作成するときに IPv6 がデフォルトで有効になっています。インスタンスの作成時に IPv6 がデフォルトで有効になっていても、インスタンスで IPv6 を設定するには、このガイドの以下の手順を実行する必要があります。

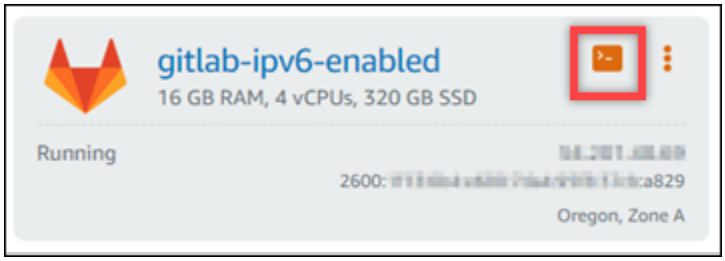
GitLab インスタンスで IPv6 を設定する

Lightsail の GitLab インスタンスで IPv6 を設定するには、以下の手順を実行します。

1. Lightsail コンソールにサインインします。
2.

> **Important**
>
> Lightsail ブラウザベースの SSH/RDP クライアントは、IPv4 トラフィックのみを受け入れます。サードパーティーのクライアントを使用して、IPv6 経由でインスタンスに SSH または RDP 接続します。詳細については、「インスタンスに接続します」を参照してください。

   Lightsail ホームページのインスタンスセクションで、設定する GitLab インスタンスを探し、ブラウザベースの SSH クライアントアイコンを選択して SSH を使用して接続します。



3. インスタンスに接続したら、以下のコマンドを入力し、インスタンスに設定されている IP アドレスを表示します。

```
ip addr
```

   以下のようなレスポンスが表示されます。

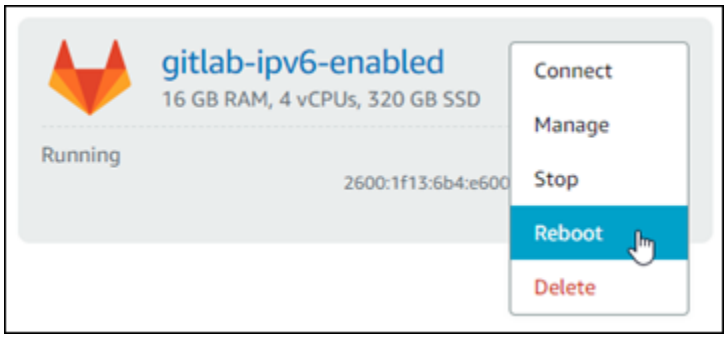
   - インスタンスが IPv6 アドレスを認識しない場合、レスポンスに表示されません。この手順のステップ 4～9 を続行する必要があります。

```
admin@ip-[redacted]:~$ ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 9001 qdisc pfifo_fast state UP group default qlen 1000
    link/ether 02:9c:[redacted]:ff:ff
    inet [redacted] brd [redacted] scope global eth0
       valid_lft forever preferred_lft forever
    inet6 fe80:[redacted]:3df7/64 scope link
       valid_lft forever preferred_lft forever
```

- インスタンスが IPv6 アドレスを認識している場合は、レスポンスに scope global の例のように表示されます。ここで停止します。インスタンスが IPv6 アドレスを認識するように設定済みなので、この手順のステップ 4～9 を実行する必要はありません。

```
admin@ip-[redacted]:~$ ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 9001 qdisc pfifo_fast state UP group default qlen 1000
    link/ether 02:9c:[redacted]:ff:ff
    inet [redacted] brd [redacted] scope global eth0
       valid_lft forever preferred_lft forever
    inet6 2600:1f13:[redacted]:f383:3212/64 scope global
       valid_lft forever preferred_lft forever
    inet6 fe80:[redacted]:3df7/64 scope link
       valid_lft forever preferred_lft forever
```

4. Lightsail コンソールに戻ります。
5. Lightsail ホームページのインスタンスタブで、 GitLab インスタンスのアクションメニュー (⋮) を選択し、 の再起動を選択します。

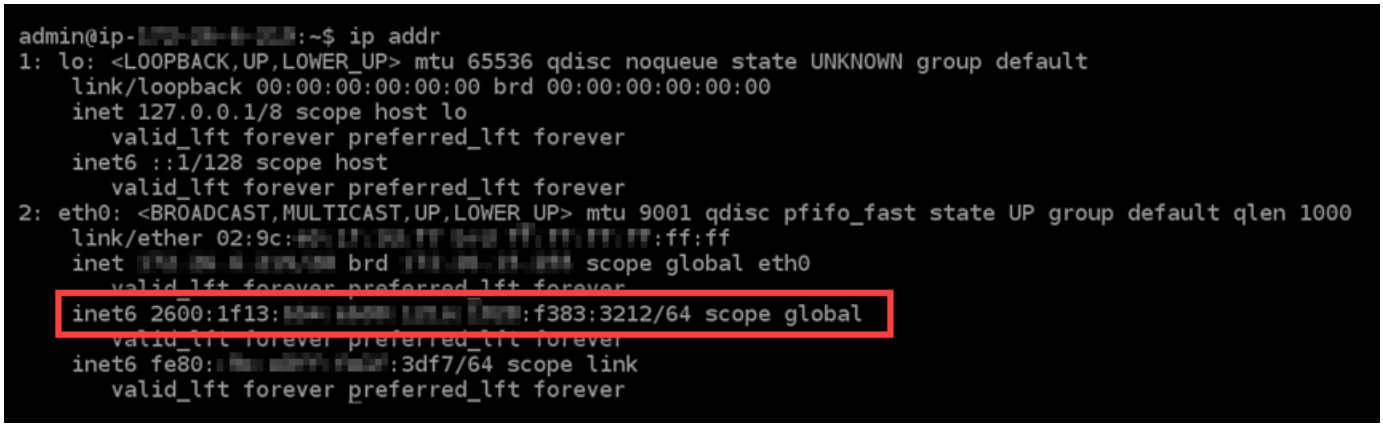


インスタンスが再起動するまで数分待ってから、次のステップに進みます。

6. GitLab インスタンスの SSH セッションに戻ります。
7. 以下のコマンドを入力して、インスタンスに設定された IP アドレスを表示し、割り当てられた IPv6 アドレスを認識していることを確認します。

```
ip addr
```

以下のようなレスポンスが表示されます。インスタンスが IPv6 アドレスを認識している場合は、レスポンスにラベルscope global の例ように表示されます。



Lightsail の Nginx インスタンスで IPv6 を設定する

Amazon Lightsail のすべてのインスタンスには、デフォルトでパブリック IPv4 アドレスとプライベート IPv4 アドレスが割り当てられています。オプションで、インスタンスの IPv6 を有効にして、パブリック IPv6 アドレスを割り当てることができます。詳細については、「Amazon Lightsail IP アドレス」および「IPv6 を有効または無効にする」を参照してください。

Nginx ブループリントを使用するインスタンスで IPv6 を有効化した後、インスタンスに IPv6 アドレスを認識させる追加のステップを実行する必要があります。このガイドでは、Nginx インスタンスで実行しなければいけない追加のステップを説明します。

前提条件

以下の前提条件を完了します (まだの場合)。

- Lightsail で Nginx インスタンスを作成します。詳細については、「インスタンスを作成する」を参照してください。
- Nginx インスタンスでI Pv6 を有効化します。詳細については、「IPv6 を有効化または無効化する」を参照してください。

> **Note**
>
> 2021 年 1 月 12 日以降に作成された新しい Nginx インスタンスは、Lightsail コンソールで作成されている場合、 IPv6 がデフォルトで有効になっています。インスタンスの作成時に IPv6 がデフォルトで有効になっていても、インスタンスで IPv6 を設定するには、このガイドの以下のステップを実行する必要があります。

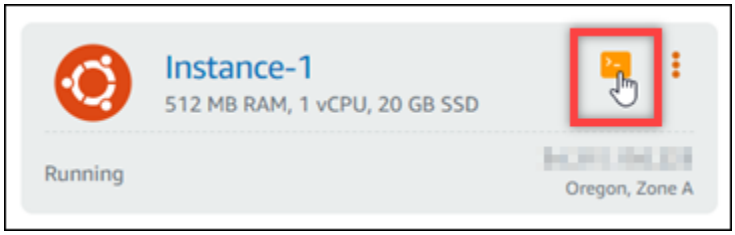
Nginx インスタンスで IPv6 を構成する

Lightsail の Nginx インスタンスで IPv6 を設定するには、以下のステップを実行します。

1. Lightsail コンソールにサインインします。
2.

> **Important**
>
> Lightsail ブラウザベースの SSH/RDP クライアントは IPv4 トラフィックのみを受け入れます。サードパーティーのクライアントを使用して、IPv6 経由でインスタンスに SSH または RDP 接続します。詳細については、「インスタンスに接続します」を参照してください。

   Lightsail ホームページのインスタンスセクションで、設定する Ubuntu 16 インスタンスを探し、ブラウザベースの SSH クライアントアイコンを選択して SSH を使用して接続します。



3. インスタンスに接続後、以下のコマンドを入力し、インスタンスがポート 80 で IPv6 リクエストをリッスンしているかを判断します。*<IPv6Address>*をインスタンスに割り当てられた IPv6 アドレスに必ず置き換えてください。

```
curl -g -6 'http://[<IPv6Address>]'
```

   例：

```
curl -g -6 'http://[2001:0db8:85a3:0000:0000:8a2e:0370:7334]'
```

   以下のようなレスポンスが表示されます。

   - インスタンスがポート 80 で IPv6 リクエストをリッスンしていない場合は、接続失敗というエラーメッセージが表示されます。この手順のステップ 4 ~ 9 を続行する必要があります。

```
bitnami@ip-[redacted]:~$ curl -g -6 'http://[2600:1f13:[redacted]:985b:25d9]:80'
curl: (7) Failed to connect to 2600:1f13:[redacted]:985b:25d9 port 80: Connection refused
```

- インスタンスがポート 80 で IPv6 リクエストをリッスンしている場合、以下のようにインスタンスのホームページの HTML コードを含んだレスポンスが表示されます。ここで停止します。インスタンスが IPv6 アドレスを認識するように設定済みなので、この手順のステップ 4 ~ 9 を実行する必要はありません。

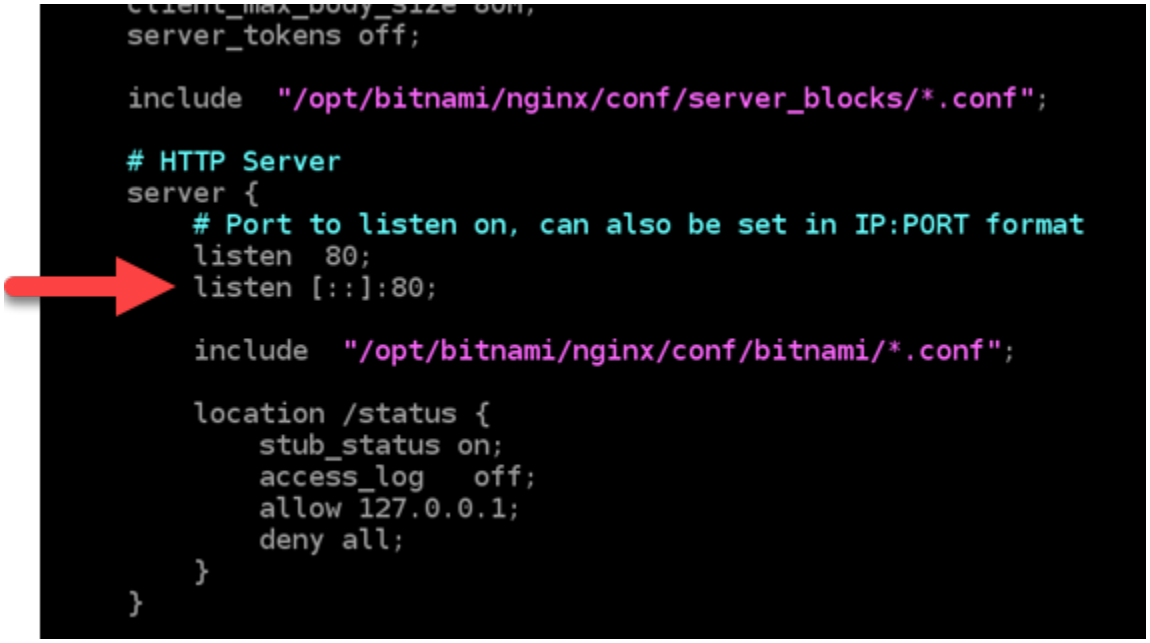
4. 次のコマンドを入力し、Vim を使用して nginx.conf 設定ファイルを開きます。

```
sudo vim /opt/bitnami/nginx/conf/nginx.conf
```

5. I キーを押して Vim 挿入モードにします。
6. 以下のテキストを`listen 80;` ファイルに既に存在しているテキストに追加します。Vim で下にスクロールして、 テキストを追加するセクションを見つける必要があるかもしれません。

```
listen [::]:80;
```

完了したらファイルは以下のようになります。



7. ESC キーを押して Vim 挿入モードを終了して、:wq! を入力してEnterを押して編集内容を保存し、Vim を終了します。
8. 以下のコマンドを入力して、インスタンスのサービスを再起動します。

```
sudo /opt/bitnami/ctlscript.sh restart
```

9. 以下のコマンドを入力して、インスタンスがポート 80 で IPv6 リクエストをリッスンしているかを判断します。*<IPv6Address>*をインスタンスに割り当てられた IPv6 アドレスに必ず置き換えてください。

```
curl -g -6 'http://[<IPv6Address>]'
```

例：

```
curl -g -6 'http://[2001:0db8:85a3:0000:0000:8a2e:0370:7334]'
```

以下のようなレスポンスが表示されます。インスタンスがポート 80 で IPv6 リクエストをリッスンしている場合、インスタンスのホームページの HTML コードを含むレスポンスが表示されます。

Lightsail の Plesk インスタンスで IPv6 を設定する

Amazon Lightsail のすべてのインスタンスには、デフォルトでパブリック IPv4 アドレスとプライベート IPv4 アドレスが割り当てられています。オプションで、インスタンスの IPv6 を有効にして、パブリック IPv6 アドレスを割り当てることができます。詳細については、「Amazon Lightsail IP アドレス」および「IPv6 を有効または無効にする」を参照してください。

Plesk ブループリントを使用するインスタンスで IPv6 を有効化した後で、インスタンスに IPv6 アドレスを認識させる追加のステップを実行する必要があります。このガイドでは、Plesk インスタンスで実行しなければいけない追加のステップを説明します。

前提条件

以下の前提条件を完了します (まだの場合)。

- Lightsail で Plesk インスタンスを作成します。詳細については、「インスタンスを作成する」を参照してください。
- Plesk インスタンスで IPv6 を有効化します。詳細については、「IPv6 を有効化または無効化する」を参照してください。

> **Note**
>
> 2021 年 1 月 12 日以降に作成された新しい Plesk インスタンスでは、Lightsail コンソールで作成されている場合 IPv6 がデフォルトで有効になっています。インスタンスの作成時に IPv6 がデフォルトで有効になっていても、インスタンスで IPv6 を設定するには、このガイドの以下のステップを実行する必要があります。

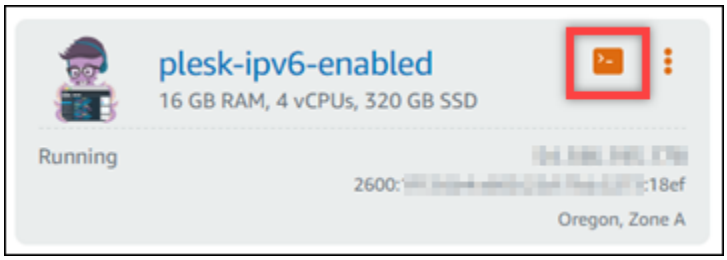
Plesk インスタンスで IPv6 を設定する

Lightsail の Plesk インスタンスで IPv6 を設定するには、以下のステップを実行します。

1. Lightsail コンソールにサインインします。
2.

> **Important**
>
> Lightsail ブラウザベースの SSH/RDP クライアントは IPv4 トラフィックのみを受け入れます。サードパーティーのクライアントを使用して、IPv6 経由でインスタンスに SSH または RDP 接続します。詳細については、「インスタンスに接続します」を参照してください。

Lightsail ホームページのインスタンスセクションで、設定する Plesk インスタンスを探し、ブラウザベースの SSH クライアントアイコンを選択して SSH を使用して接続します。



3. インスタンスに接続したら、以下のコマンドを入力し、インスタンスに設定されている IP アドレスを表示します。

```
ip addr
```

以下のようなレスポンスが表示されます。

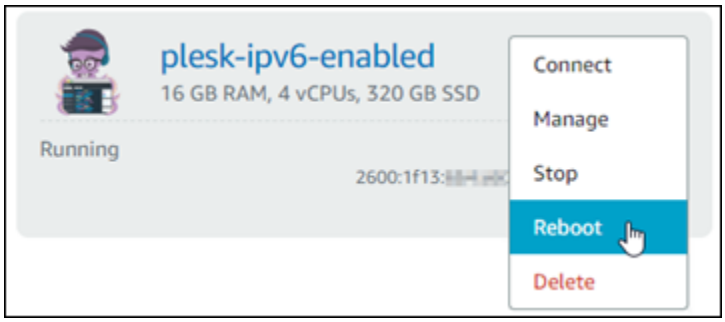
- インスタンスが IPv6 アドレスを認識しない場合、レスポンスに表示されません。この手順のステップ 4～7 を続行する必要があります。

```
admin@ip-[redacted]:~$ ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 9001 qdisc pfifo_fast state UP group default qlen 1000
    link/ether 02:9c:[redacted]:ff:ff
    inet [redacted] brd [redacted] scope global eth0
       valid_lft forever preferred_lft forever
    inet6 fe80:[redacted]:3df7/64 scope link
       valid_lft forever preferred_lft forever
```

- インスタンスが IPv6 アドレスを認識している場合は、レスポンスにscope global の例のように表示されます。ここで停止します。インスタンスが IPv6 アドレスを認識するように設定済みなので、この手順のステップ 4～7 を実行する必要はありません。

```
admin@ip-[redacted]:~$ ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 9001 qdisc pfifo_fast state UP group default qlen 1000
    link/ether 02:9c:[redacted]:ff:ff
    inet [redacted] brd [redacted] scope global eth0
       valid_lft forever preferred_lft forever
    inet6 2600:1f13:[redacted]:f383:3212/64 scope global
       valid_lft forever preferred_lft forever
    inet6 fe80:[redacted]:3df7/64 scope link
       valid_lft forever preferred_lft forever
```

4. Lightsail コンソールに戻ります。
5. Lightsail ホームページの [Instances] (インスタンス)タブで、Plesk インスタンスのアクションメニュー (⋮)を選択して、[Reboot] (再起動) を選択します。

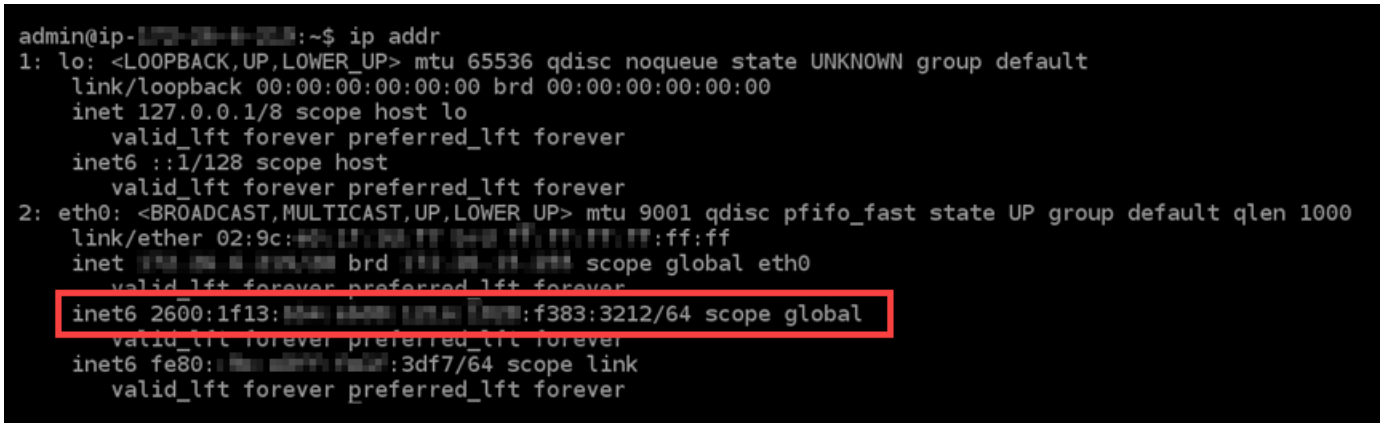


インスタンスが再起動するまで数分待ってから、次のステップに進みます。

6. Plesk インスタンスの SSH セッションに戻ります。
7. 以下のコマンドを入力して、インスタンスに設定された IP アドレスを表示し、割り当てられた IPv6 アドレスを認識していることを確認します。

```
ip addr
```

以下のようなレスポンスが表示されます。インスタンスが IPv6 アドレスを認識している場合は、レスポンスにラベルscope global の例ように表示されます。



Lightsail で Ubuntu 16 インスタンス用に IPv6 を設定する

Amazon Lightsail のすべてのインスタンスには、デフォルトでパブリック IPv4 アドレスとプライベート IPv4 アドレスが割り当てられています。オプションで、インスタンスの IPv6 を有効にして、パブリック IPv6 アドレスを割り当てることができます。詳細については、「Amazon Lightsail での IP アドレス」および「IPv6 の有効化または無効化」を参照してください。 IPv6 Amazon Lightsail

Ubuntu 16 ブループリントを使用するインスタンスで IPv6 を有効化した後、インスタンスに IPv6 アドレスを認識させる追加のステップを実行する必要があります。このガイドでは、Ubuntu 16 インスタンスで実行しなければいけない追加のステップを説明します。

前提条件

以下の前提条件を完了します (まだの場合)。

- Lightsail で Ubuntu 16 インスタンスを作成します。詳細については、「インスタンスを作成する」を参照してください。
- Ubuntu 16 インスタンスの IPv6 を有効化します。詳細については、「IPv6 を有効化または無効化する」を参照してください。

> **Note**
>
> 2021 年 1 月 12 日以降に作成された新しい Ubuntu インスタンスでは、Lightsail コンソールで作成されている場合 IPv6 がデフォルトで有効になっています。インスタンスの作成時に IPv6 がデフォルトで有効になっていても、インスタンスで IPv6 を設定するには、このガイドの以下のステップを実行する必要があります。

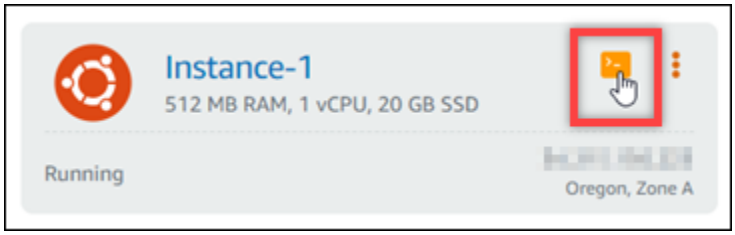
Ubuntu 16 インスタンスで IPv6 を設定する

Lightsail の Ubuntu 16 インスタンスで IPv6 を設定するには、以下のステップを実行します。

1. Lightsail コンソールにサインインします。
2.

> **Important**
>
> Lightsail ブラウザベースの SSH/RDP クライアントは IPv4 トラフィックのみを受け入れます。サードパーティーのクライアントを使用して、IPv6 経由でインスタンスに SSH または RDP 接続します。詳細については、「インスタンスに接続します」を参照してください。

   Lightsail ホームページのインスタンスセクションで、設定する Ubuntu 16 インスタンスを探し、ブラウザベースの SSH クライアントアイコンを選択して SSH を使用して接続します。



3. インスタンスに接続したら、以下のコマンドを入力し、インスタンスに設定されている IP アドレスを表示します。

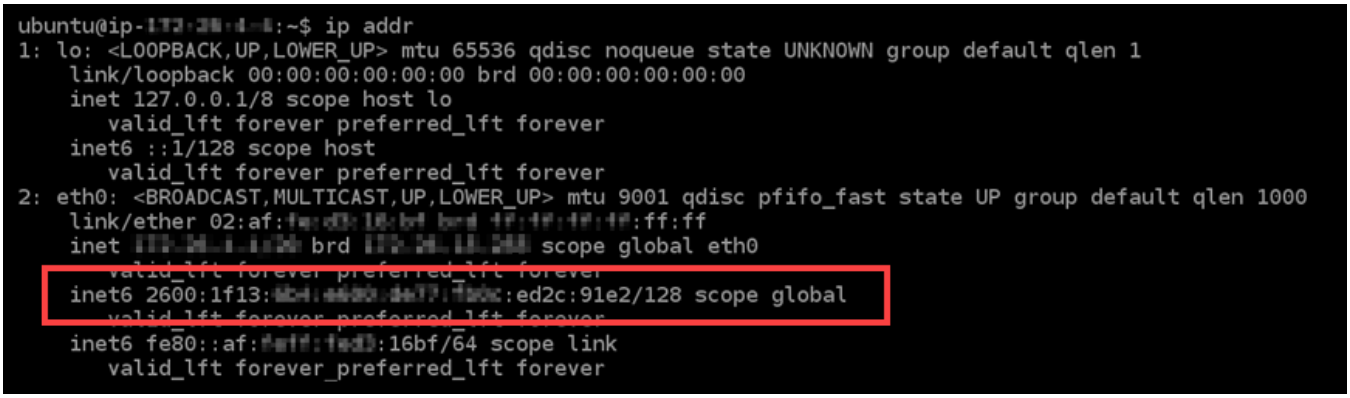
```
ip addr
```

   以下のようなレスポンスが表示されます。

   - インスタンスが IPv6 アドレスを認識しない場合、レスポンスに表示されません。この手順のステップ 4～9 を続行する必要があります。

```
ubuntu@ip-172-26-4-4:~$ ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 9001 qdisc pfifo_fast state UP group default qlen 1000
    link/ether 02:af:[redacted]:ff:ff
    inet [redacted] brd [redacted] scope global eth0
       valid_lft forever preferred_lft forever
    inet6 fe80::af:[redacted]:16bf/64 scope link
       valid_lft forever preferred_lft forever
```

- インスタンスが IPv6 アドレスを認識している場合は、レスポンスに `scope global` の例のように表示されます。ここで停止します。インスタンスが IPv6 アドレスを認識するように設定済みなので、この手順のステップ 4～9 を実行する必要はありません。



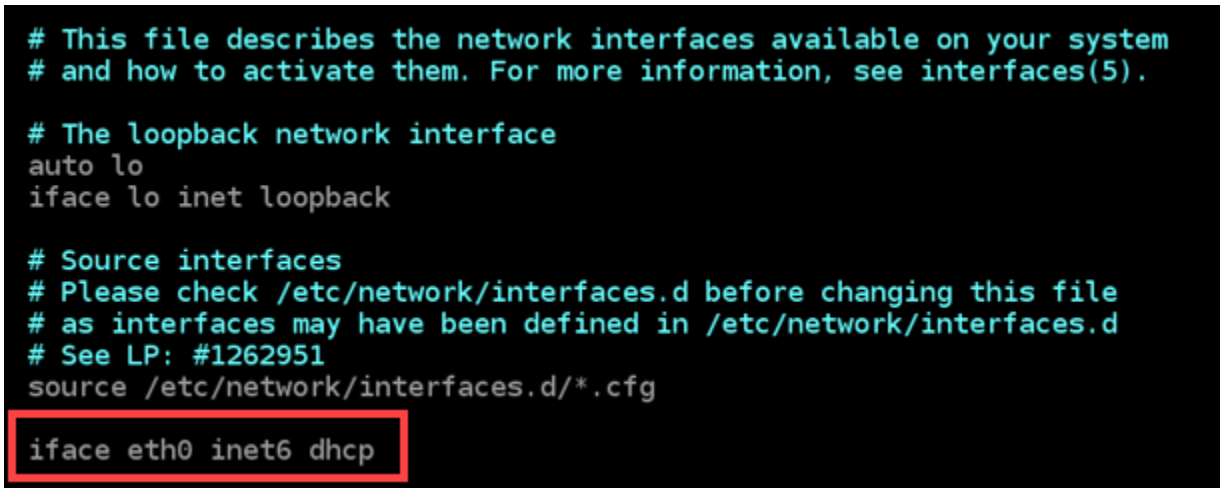
4. 以下のコマンドを入力して、Vim を使用して設定ファイルを開きます。

```
sudo vim /etc/network/interfaces
```

5. I キーを押して Vim 挿入モードにします。
6. ファイルの末尾に次の行を追加します。

```
iface eth0 inet6 dhcp
```

完了したファイルは以下のようになります。



7. ESC キーを押して Vim 挿入モードを終了して、`:wq!` を入力して Enter を押して編集内容を保存し、Vim を終了します。
8. 以下のコマンドを入力して、インスタンス上のネットワークサービスを再起動します。

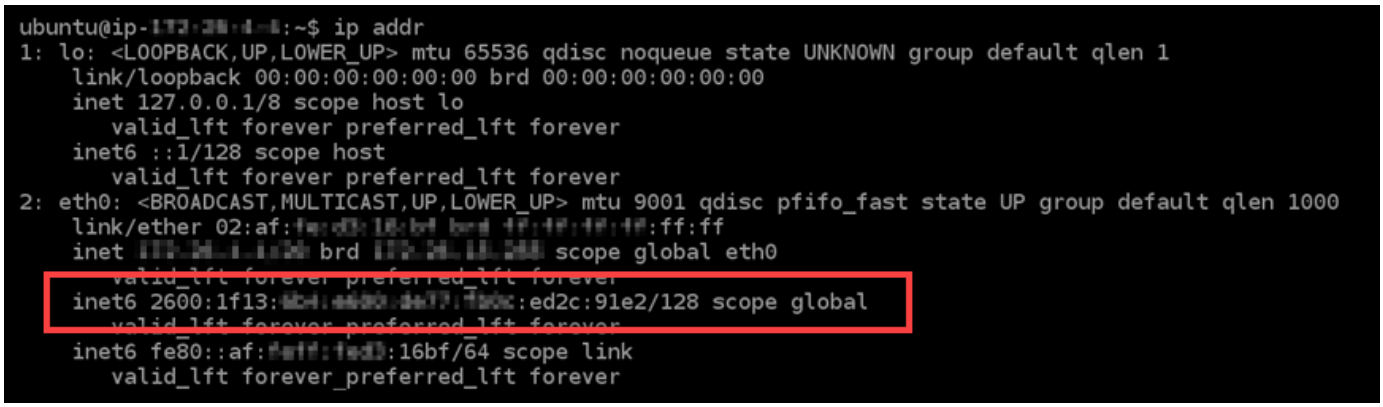
```
sudo service networking restart
```

インスタンスのネットワークサービスを再起動した後、インスタンスが IPv6 アドレスを認識するまで数分待たなければならない場合があります。

9. 以下のコマンドを入力して、インスタンスに設定された IP アドレスを表示し、割り当てられた IPv6 アドレスを認識していることを確認します。

```
ip addr
```

以下のようなレスポンスが表示されます。インスタンスが IPv6 アドレスを認識している場合は、レスポンスにラベルscope global の例ように表示されます。



Amazon Lightsail で操作する

Lightsail で完遂できる様々なタスクについて知るには、以下のチュートリアルをご利用ください。たとえば、トラブルシューティング用の HAR ファイルを作成したり、LAMP インスタンスを起動して設定したり、MySQL データベースを移行したりできます。

トピック

- Lightsail でAWS Command Line Interface を操作する
- Lightsail API または AWS Command Line Interface を使用するためのアクセスキーを作成する
- AWS CloudShell Lightsail の
- AWS CloudTrail による Lightsail API コールのログ記録
- チュートリアル: Lightsail LAMP インスタンスを Aurora データベースに接続する
- チュートリアル: HAR ファイルの作成方法

- Lightsail インスタンスを強制停止する
- チュートリアル: Linux ベースの Lightsail インスタンスに Prometheus をインストールする
- チュートリアル: Lightsail LAMP インスタンスを起動して設定する
- チュートリアル: Windows Server 2016 インスタンスを起動して設定する
- Amazon Lightsail の詳細情報
- チュートリアル: MySQL 5.6 データベースから最新のデータベースバージョンにデータを移行する
- Lightsail で Plesk をセットアップして設定する
- チュートリアル: コンテンツ配信ネットワークディストリビューションで Lightsail バケットを使用する
- Lightsail を他の AWS サービスで使用する
- AWS CloudFormation で Lightsail リソースを作成する

Lightsail でAWS Command Line Interface を操作する

AWS Command Line Interface (AWS CLI) は、上級ユーザーと開発者向けのツールであり、ターミナル (Linux と Unix) またはコマンドプロンプト (Windows) でコマンドを入力することで、Amazon Lightsail サービスを制御できます。また、Lightsail コンソール、グラフィカルユーザーインターフェイス、および Lightsail アプリケーションプログラムインターフェイス (API) を使用して、Lightsail を制御することもできます。

Lightsail では、AWS CLI をローカルデスクトップまたは Lightsail インスタンスにインストールできます。

AWS CLI の詳細については、「AWS Command Line Interface ユーザーガイド」を参照してください。「AWS CLI コマンドリファレンス」で Amazon Lightsail コマンドを確認できます。

- AWS CLI をローカルデスクトップにインストールするには、AWS Command Line Interface ドキュメント内の「AWS CLI のインストール」を参照してください。
- AWS CLI を Ubuntu ベースの Lightsail インスタンスにインストールするには、インスタンスに接続し、「`sudo apt-get -y install awscli`」と入力します。

> **Note**
>
> AWS CLI は Amazon Linux の Lightsail インスタンスにインストール済みです。再インストールする必要がある場合は、インスタンスに接続し、「`sudo yum install aws-cli`」と入力します。

AWS CLI をインストールした後、アクセスキーを取得してから、それらを使用するように AWS CLI を設定する必要があります。詳細については、「Lightsail API または AWS Command Line Interface を使用するためのアクセスキーを作成する」を参照してください。

Lightsail API または AWS Command Line Interface を使用するためのアクセスキーを作成する

Lightsail API または AWS Command Line Interface (AWS CLI) を使用するには、新規アクセスキーを作成する必要があります。アクセスキーは、アクセスキー ID とシークレットアクセスキーで構成されます。以下の手順に従って、キーを作成し、Lightsail API を呼び出すように AWS CLI を設定します。

ステップ 1: 新規アクセスキーを作成する

新規アクセスキーは AWS Identity and Access Management (IAM) コンソールで作成できます。

1. IAM コンソールにサインインします。
2. アクセスキーを作成するユーザーの名前を選択します。選択するユーザーには、Lightsail アクションを行うフルアクセス権または特定のアクセス権が必要です。
3. [Security credentials] タブを選択します。
4. [アクセスキー] セクションで、[アクセスキーの作成] を選択します。

   > **Note**
   >
   > 一度に 1 人のユーザーが持つことができるのは、最大 2 つのアクセスキー (有効または無効) です。すでに 2 つある場合は、新しいものを作成する前に、いずれかを削除する必要があります。アクセスキーを削除する前に、アクセスキーがアクティブに使用されていないことを確認してください。

5. アクセスキー ID とリストされたシークレットアクセスキーを記録します。[シークレットアクセスキー] 列の [Show] を選択して、シークレットアクセスキーを表示します。

これらをこの画面からコピーするか、[キーファイルのダウンロード] を選択して、アクセスキー IDとシークレットアクセスキーが含まれている .csv ファイルをダウンロードします。

> **Important**
>
> アクセスキーを安全な場所に保管します。後で見つけやすいように、そのファイルに `MyLightsailKeys.csv` のような名前を付けます。IAM コンソールから CSV ファイルをダウンロードした場合は、ステップ 2 の完了後に削除する必要があります。後で新しいアクセスキーを作成できます。

ステップ 2: AWS CLI の設定

AWS CLI をインストールしていない場合はインストールします。「AWS Command Line Interface のインストール」を参照してください。AWS CLI をインストールした後に、CLI を使用できるように設定する必要があります。

1. ターミナルウィンドウまたはコマンドプロントを開きます。
2. タイプ `aws configure`。
3. 前のステップで作成した .csv ファイルから AWS アクセスキー ID を貼り付けます。
4. 入力を求められたら、AWS シークレットアクセスキーを貼り付けます。
5. リソースが存在している AWS リージョン を入力します。たとえば、リソースが主に Ohio に置かれている場合は、`[us-east-2デフォルトリージョン名]` の入力を求められたときに [] を選択します。

   AWS CLI の `--region` オプションの使用方法の詳細については、『リファレンス』の「AWS CLI汎用オプション」を参照してください。
6. [Default output format (デフォルトの出力形式)] (`json` など) を選択します。

次のステップ

- SDK のインストール
- AWS Command Line Interface と連携するための Amazon Lightsail の設定
- API のドキュメントを読む

AWS CloudShell Lightsail の

AWS CloudShell はブラウザベースの事前認証済みシェルで、Amazon Lightsail コンソールから直接起動できます。を使用して CloudShell 、コマンドラインインターフェイスから Lightsail リソースを管理します。 AWS Command Line Interface （ AWS CLI) コマンドは、Bash、 PowerShellZ シェルなどの任意のシェルを使用して実行できます。この手順は、コマンドラインツールのダウンロードもインストールも不要です。を起動すると CloudShell、Amazon Linux 2 に基づくコンピューティング環境が作成されます。この環境では、 AWS CLIなど、プリインストールされている広範な開発ツールにアクセスできます。プリインストールされたツールの完全なリストについては、「 CloudShell ユーザーガイド」の「プリインストールされたソフトウェア」を参照してください。

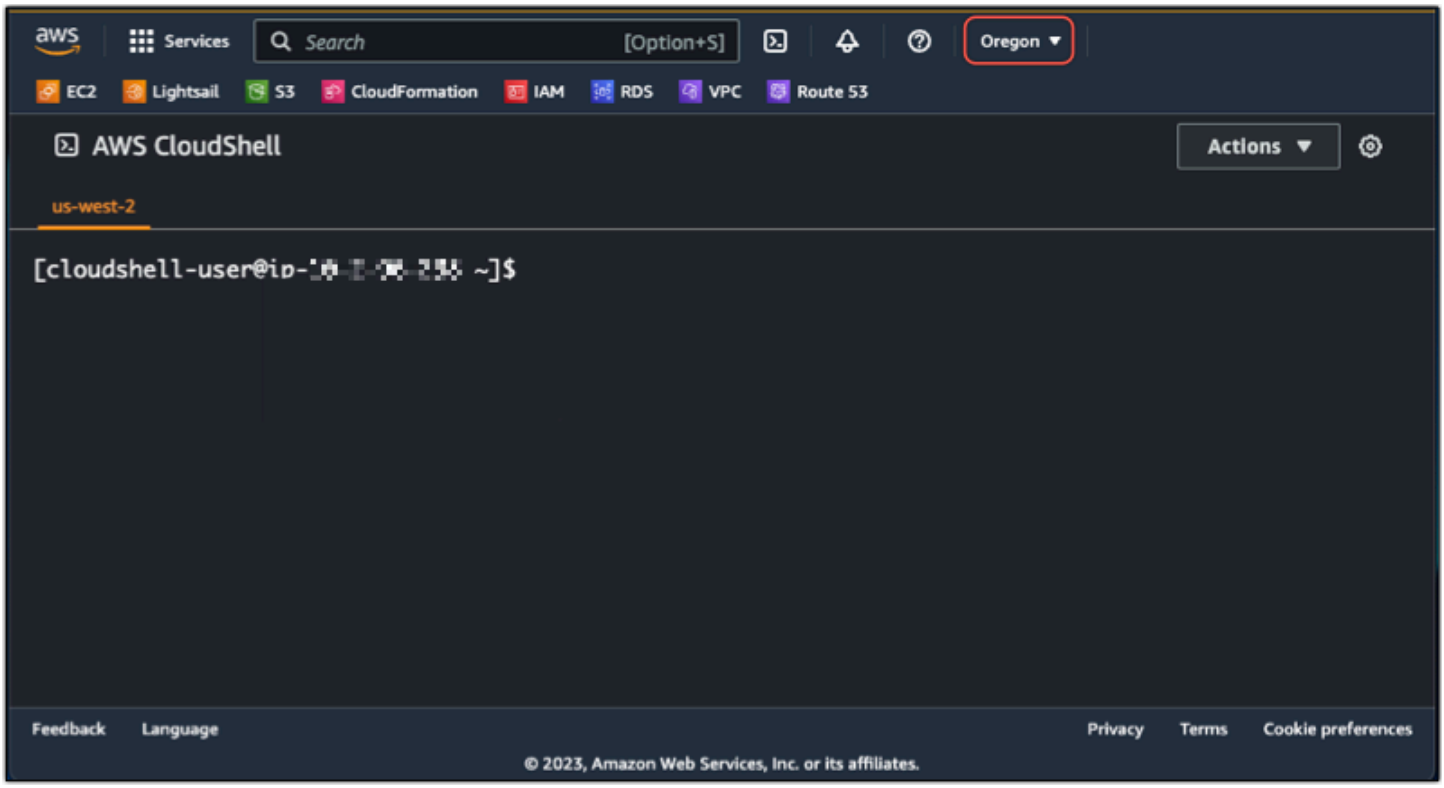
永続的ストレージ

では AWS CloudShell、追加料金 AWS リージョン なしで、各 で最大 1 GB の永続的ストレージを使用できます。永続的ストレージはホームディレクトリ ($HOME) にあり、ユーザーのプライベートな記憶域です。各シェルセッションが終了した後に削除されるエフェメラル環境リソースとは異なり、ホームディレクトリ内のデータはセッション間で保持されます。永続的ストレージでのデータの保持の詳細については、 CloudShell ユーザーガイドの永続的ストレージを参照してください。

AWS リージョン

Lightsail では、物理的な場所へのレイテンシーが最も少ない CloudShell セッション AWS リージョン が で開きます。つまり、 はセッション間で変更 AWS リージョン される可能性があります。1 GB の永続ストレージを使用できるように、 CloudShell セッションがどの AWS リージョン TAK にあるかを書き留めておきます。セッションの AWS リージョンを変更するには、[新しいブラウザタブで開く] アイコンを選択します。これにより、新しいブラウザウィンドウで CloudShell セッションにアクセスするオプションが提供されます。

新しいブラウザタブのナビゲーションバーで、現在表示されている AWS リージョン の名前を選択します。次に、切り替え AWS リージョン る を選択します。



の詳細については CloudShell、「CloudShell ユーザーガイド」を参照してください。

を起動して使用する AWS CloudShell

Lightsail 内で AWS CloudShell セッションを起動して使用する方法について説明します。を実行するアクセス許可がない場合は CloudShell、使用している AWS Identity and Access Management (IAM)

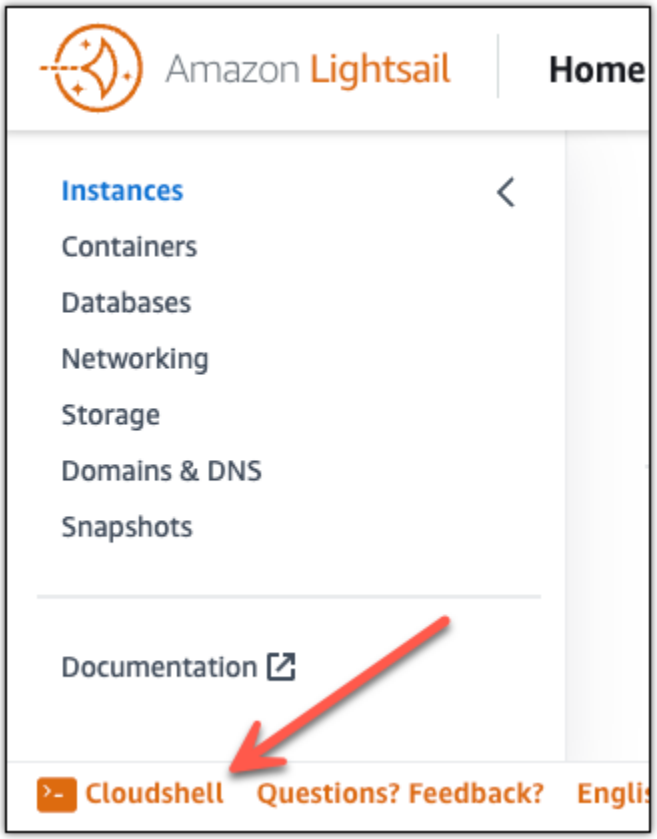
ID にarn:aws:iam::aws:policy/AWSCloudShellFullAccessポリシーを追加する必要があります。arn:aws:iam::aws:policy/AdministratorAccess ポリシーが既にアタッチされている場合は、 にアクセスできます CloudShell。詳細については、「???」を参照してください。

起動 AWS CloudShell

Amazon Lightsail コンソール CloudShell から を起動できます。セッションが開始されたら、Bash、PowerShell、または Z shell などのお好みのシェルに切り替えることができます。

Lightsail で新しい AWS CloudShell セッションを起動するには、次のステップを実行します。

1. https://lightsail.aws.amazon.com/ で Lightsail コンソールにサインインします。
2. コンソールの左下にあるコンソールツールバーCloudShellで を選択します。コマンドプロンプトが表示されたら、シェルは対話的な操作の準備ができています。



3. (オプション) 使用するプリインストールされたシェルを選択するには、コマンドラインプロンプトで次のプログラム名のいずれかを入力します。

Bash: **bash**

Bash に切り替えると、コマンドプロンプトの記号が$にアップロードします。Bash は のデフォルトシェルです AWS CloudShell。

PowerShell: **pwsh**

に切り替えると PowerShell、コマンドプロンプトの記号が に更新されますPS>。

Z シェル: **zsh**

Z シェルに切り替えると、コマンドプロンプトの記号が%に講アップロードします。

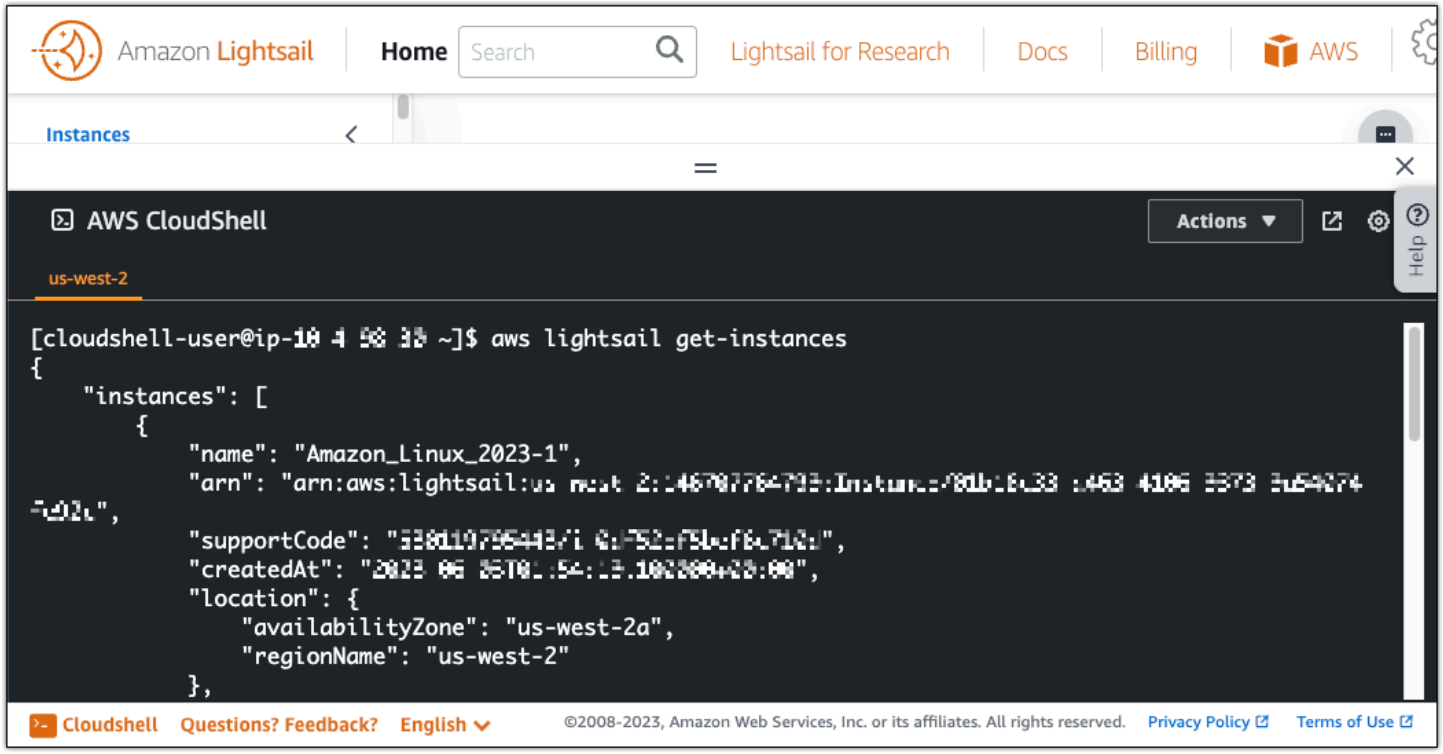
Example の Lightsail API コマンドの例 AWS CloudShell

セッションにプリインストールされている CloudShell複数のコマンドラインツールを使用できます。この例では、Lightsail `GetInstances` API オペレーションを使用して、Lightsail アカウントにあるインスタンスを表示します。`GetInstances` API オペレーションの詳細については、「Amazon Lightsail API リファレンスGetInstances」の「」を参照してください。

1. https://lightsail.aws.amazon.com/ で Lightsail コンソールにサインインします。
2. コンソールの左下にあるコンソールツールバーCloudShellで を選択します。
3. AWS CloudShell プロンプトが表示されたら、次のコマンドを入力します。

```
aws lightsail get-instances
```

Lightsail アカウントにあるインスタンスの完全なリストが表示されます。



追加情報

の詳細については、次のドキュメントを参照してください AWS CloudShell。

- Amazon Lightsail API リファレンス
- に関するよくある質問 AWS CloudShell
- でサポートされているブラウザ AWS CloudShell
- でのトラブルシューティング AWS CloudShell
- AWS のサービス での の使用 AWS CloudShell

AWS CloudTrail による Lightsail API コールのログ記録

Amazon Lightsail は AWS CloudTrail と統合されています。このサービスは、ユーザーやロール、または AWS の Lightsail のサービスによって実行されたアクションをレコードするサービスです。CloudTrail は、Lightsail のすべての API コールをイベントとしてキャプチャします。キャプチャされた呼び出しには、Lightsail コンソールの呼び出しと、Lightsail API オペレーションへのコード呼び出しが含まれます。トレイルを作成すると、Lightsail のイベントを含む CloudTrail イベントの Amazon S3 ケットへの継続的な配信が可能になります。追跡を設定しない場合でも、CloudTrail コンソールの [Event history] (イベント履歴) で最新のイベントを表示できます。CloudTrail によって

収集された情報を使用して、Lightsail に対して行われた要求、要求が行われた IP アドレス、要求を行った人、要求が行われた日時、および追加の詳細を判別できます。

CloudTrail の詳細については、「AWS CloudTrail ユーザーガイド」を参照してください。

LightsailCloudTrail での 情報

AWS アカウントを作成すると、そのアカウントに対して CloudTrail が有効になります。Lightsail でアクティビティが発生すると、そのアクティビティは [Event history (イベント履歴)] で AWS のその他のサービスのイベントと共に CloudTrail イベントに記録されます。AWS アカウントで最近のイベントを表示、検索、ダウンロードできます。詳細については、「Viewing Events with CloudTrail Event History」(CloudTrail イベント履歴でのイベントの表示) を参照してください。

AWS のイベントなど、Lightsail アカウントのイベントの継続的なレコードについては、追跡を作成します。追跡により、CloudTrail はログファイルを Simple Storage Service (Amazon S3) バケットに配信できます。デフォルトでは、コンソールで作成した追跡がすべての AWS リージョンに適用されます。追跡は、AWS パーティションのすべてのリージョンからのイベントを記録し、指定した Simple Storage Service (Amazon S3) バケットにログファイルを配信します。さらに、CloudTrail ログで収集したイベントデータをより詳細に分析し、それに基づく対応するためにその他の AWS のサービスを設定できます。詳細については、次を参照してください。

- 追跡を作成するための概要
- CloudTrail のサポート対象サービスと統合
- Amazon SNS の CloudTrail の通知の設定
- 「複数のリージョンから CloudTrail ログファイルを受け取る」および「複数のアカウントから CloudTrail ログファイルを受け取る」

すべての Lightsail アクションは CloudTrail によってログに記録され、「Amazon Lightsail API リファレンス」に記録されます。たとえば、[GetInstance]、[AttachStaticIp]、[RebootInstance] の各セクションの呼び出しにより、CloudTrail ログファイルにエントリが生成されます。

各イベントまたはログエントリには、リクエストの生成者に関する情報が含まれます。アイデンティティ情報は、以下を判別するのに役立ちます。

- リクエストが、ルート認証情報と AWS Identity and Access Management (IAM) ユーザー認証情報のどちらを使用して送信されたか。
- リクエストがロールまたはフェデレーティッドユーザーのテンポラリなセキュリティ認証情報を使用して行われたかどうか。

- リクエストが、別の AWS のサービスによって送信されたかどうか。

詳細については、「CloudTrail userIdentity エレメント」を参照してください。

Lightsail ログファイルエントリの概要

「トレイル」は、指定した Simple Storage Service (Amazon S3) バケットにイベントをログファイルとして配信するように設定できます。CloudTrail のログファイルには、単一か複数のログエントリがあります。イベントはあらゆるソースからの単一のリクエストを表し、リクエストされたアクション、アクションの日時、リクエストのパラメータなどの情報が含まれます。CloudTrail ログファイルは、パブリック API コールの順序付けられたスタックトレースではないため、特定の順序では表示されません。

チュートリアル: Lightsail LAMP インスタンスを Aurora データベースに接続する

投稿、ページ、およびユーザーのアプリケーションデータは、Amazon Lightsail の LAMP インスタンスで実行している MariaDB データベースに保存されています。WordPress インスタンスに障害が発生した場合、データが回復不可能になる場合があります。このシナリオを回避するには、 MySQL マネージドデータベースにアプリケーションのデータを転送する必要があります。

Amazon Aurora はクラウド用に構築された MySQL と PostgreSQL 互換のリレーショナルデータベースです。これは従来のエンタープライズデータベースのパフォーマンスと可用性に、オープンソースデータベースのシンプルさと費用対効果を組み合わせています。Aurora は、Amazon Relational Database Service (Amazon RDS) の一部として提供されています。Amazon RDS は、クラウドでリレーショナルデータベースを簡単に設定、運用、およびスケールすることができるマネージドデータベースサービスです。詳細については、「Amazon Relational Database Service ユーザーガイド」と「Aurora の Amazon Aurora ユーザーガイド」を参照してください。

このチュートリアルでは、Lightsail 内の LAMP インスタンスからアプリケーションデータベースを Amazon RDS 内の Aurora マネージドデータベースに接続する方法について説明します。

目次

- ステップ 4: MariaDB データベースを LAMP インスタンスから Aurora データベースに転送する
- ステップ 5: Aurora マネージドデータベースに接続するようアプリケーションを設定する

ステップ 1: 前提条件を満たす

開始する前に次の前提条件を完了します。

1. Lightsail で LAMP インスタンスを作成し、アプリケーションを設定します。続行する前に、インスタンスは実行中状態になっていることを確認してください。詳細については、「チュートリアル: Lightsail で LAMP インスタンスを起動して設定する」を参照してください。
2. Lightsail アカウントで VPC ピアリングを有効にします。詳細については、「Lightsail の外で AWS リソースを使用するために Amazon VPC ピア接続をセットアップする」を参照してください。
3. Amazon RDS に Aurora マネージドデータベースを作成します。データベースは、LAMP リソースと同じ AWS リージョン にある必要があります。続行する前に、データベースが実行中状態になっていることを確認してください。詳細については、「Aurora の Amazon Aurora ユーザーガイド」の「Amazon Aurora の使用開始」を参照してください。

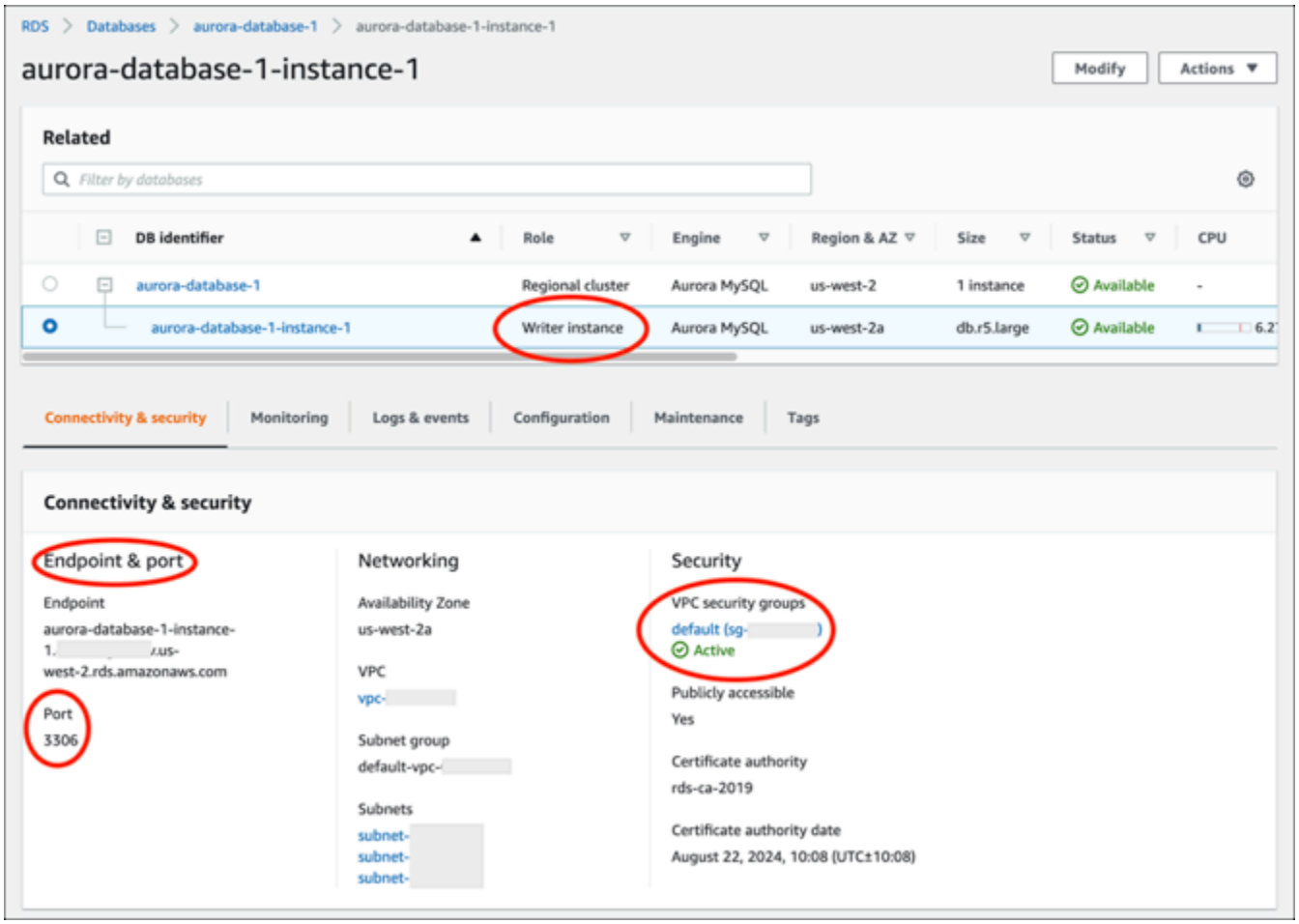
ステップ 2: Aurora データベースのセキュリティグループを設定する

AWS セキュリティグループは AWS リソースの仮想ファイアウォールとして機能します。Amazon RDS 内の Aurora データベースに接続できる送受信トラフィックを制御します。詳細については、「Amazon Virtual Private Cloud ユーザーガイドのセキュリティグループを使用してリソースへのトラフィックを制御する」を参照してください。

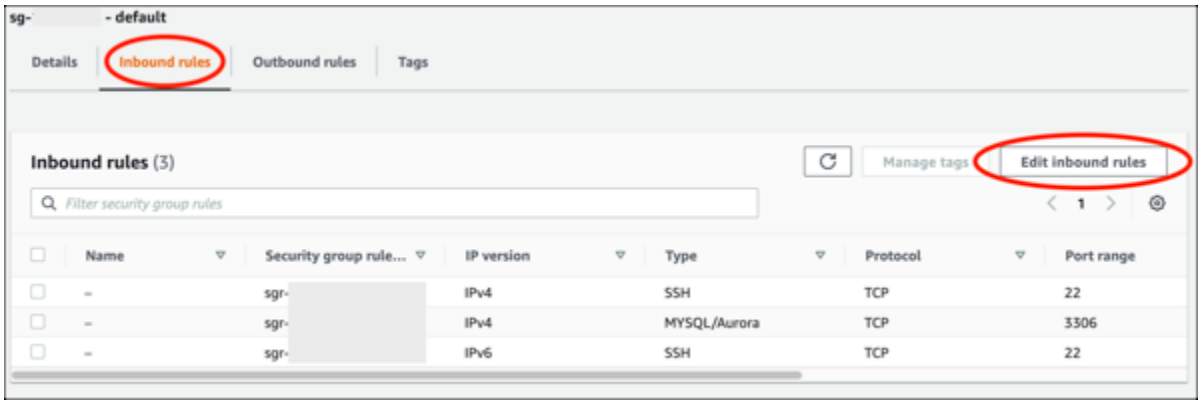
LAMP インスタンスが Aurora データベースへの接続を確立できるよう、以下の手順を完了してセキュリティグループを設定します。

1. Amazon RDS コンソールにサインインします。
2. ナビゲーションペインで、[Databases] (データベース) を選択します。
3. LAMP インスタンスが接続する Aurora データベースの[ライターインスタンス]を選択します。
4. [Connectivity & security (接続とセキュリティ)] タブを選択します。
5. [Endpoint & port] (エンドポイントとポート) セクションに表示されるライターインスタンスのエンドポイント名とポートを記録します。これらの情報は、データベースに接続する Lightsail インスタンスを設定するときに必要になります。

6. [Security] (セキュリティ) セクションでアクティブな VPC セキュリティグループのリンクを選択します。データベースのセキュリティグループにリダイレクトされます。



7. Aurora データベースのセキュリティグループが選択されていることを確認します。
8. [Inbound rules] (インバウンドルール) タブを開きます。
9. [Edit inbound rules] (インバウンドルールの編集) を選択します。



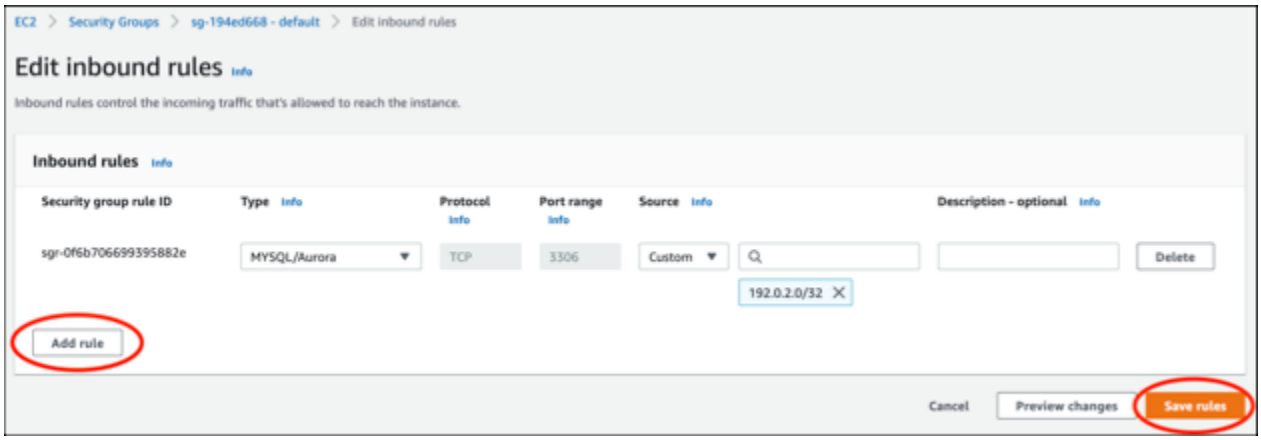
10. [Edit inbound rules] (インバウンドルールの編集) ページで [Add rule] (ルールの追加) を選択します。

11. 次のいずれかのステップを完了します。

    - デフォルトの MySQL ポート 3306 を使用する場合は、[Type] (タイプ) ドロップダウンメニューから [MySQL/Aurora] を選択します。
    - データベースのカスタムポートを使用する場合は、[Type] (タイプ) ドロップダウンメニューから [Custom TCP] (カスタム TCP) を選択し、[Port Range] (ポート範囲) テキストボックスにポート番号を入力します。

12. [Source] (ソース) テキストボックスに LAMP インスタンスのプライベート IP アドレスを追加します。IP アドレスは、 CIDR 表記で入力する必要があります (/32 を追加する必要があります)。例えば、`192.0.2.0` を許可するには「`192.0.2.0/32`」と入力します。

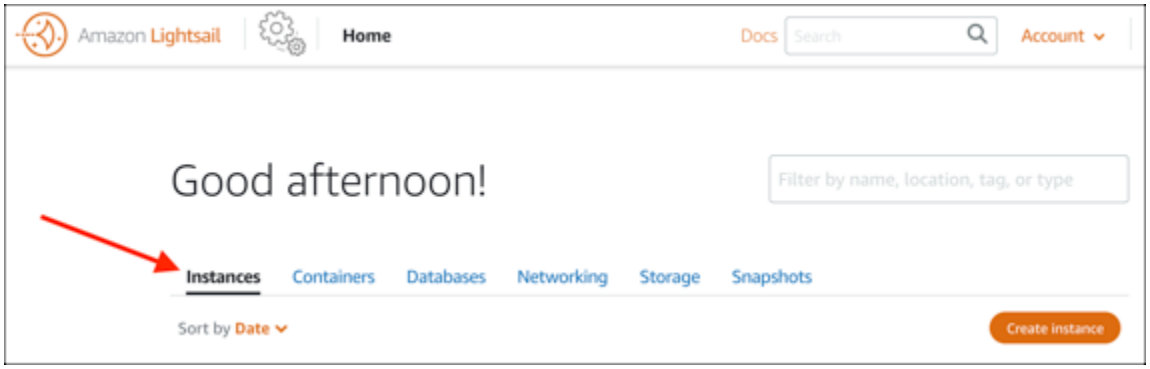
13. [Save Rules] (ルールの保存) を選択します。



ステップ 3: Lightsail インスタンスから Aurora データベースに接続する

以下の手順を完了して、Lightsail インスタンスから Aurora データベースに接続できることを確認します。

1. Lightsail コンソールにサインインします。
2. Lightsail ホームページで、[Instances (インスタンス)] タブを選択します。



3. SSH を使用して接続する LAMP インスタンスのブラウザベースの SSH クライアントアイコンを選択します。

4. インスタンスに接続したら、次のコマンドを入力して、Aurora データベースに接続します。このコマンドでは、*DatabaseEndpoint* を実際の Aurora データベースのエンドポイントアドレスで置き換え、*Port* をデータベースのポートで置き換えます。*MyUsername* は、データベースを作成したときに入力したユーザーの名前で置き換えます。

```
mysql -h DatabaseEndpoint -P Port -u MyUserName -p
```

インスタンスが Aurora データベースにアクセスおよび接続できれば、次の例のような応答が表示されます。

```
bitnami@ip-            $ mysql -h database.cluster-            .us-west-2.rds.amazonaws.
com -P 3306 -u admin -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MySQL connection id is 215
Server version: 5.6.10 MySQL Community Server (GPL)

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MySQL [(none)]>
```

この応答が表示されない場合、またはエラーメッセージが表示された場合は、データベースのセキュリティグループを設定して、Lightsail インスタンスのプライベート IP アドレスからの接続を許可する必要があることがあります。詳細については、このガイドの「Aurora データベースのセキュリティグループを設定する」を参照してください。

ステップ 4: MariaDB データベースを LAMP インスタンスから Aurora データベースに転送する

インスタンスからデータベースに接続できることを確認した後は、データを LAMP インスタンスデータベースから Aurora データベースに移行する必要があります。詳細については、「Aurora の

Amazon Aurora ユーザーガイド」の「Amazon Aurora MySQL DB クラスターのモニタリングデータ」を参照してください。

ステップ 5: Aurora マネージドデータベースに接続するようアプリケーションを設定する

アプリケーションデータを Aurora データベースに転送した後、LAMP インスタンス上で実行するアプリケーションを設定して Aurora データベースに接続します。SSH を使用して LAMP インスタンスに接続し、アプリケーションのデータベース設定ファイルにアクセスします。設定ファイルで、Aurora データベースのエンドポイントアドレス、データベースユーザー名、およびパスワードを定義します。設定ファイルの例を以下に示します。

```
bitnami@ip-            :~/htdocs$ cat connectvalues.php
<?php
$host           = 'database.cluster-            .us-west-2.rds.amazonaws.com';
$username       = 'admin';
$password       = 'Password1';
```

チュートリアル: HAR ファイルの作成方法

Amazon Lightsail コンソールまたは Lightsail 仮想プライベートサーバー (VPS) で問題が発生した場合、ウェブブラウザから HAR ファイルを送信するように AWS Support から求められる場合があります。HAR ファイルには、一般的で診断が難しい問題のトラブルシューティングに役立つ重要な情報が含まれています。HAR ファイルにより、AWS Support がこれらの問題を調査または再現することもできます。

> **Important**
>
> HAR ファイルには、ユーザー名、パスワード、キーなどの機密情報が取り込まれることがあります。共有する前に、必ず HAR ファイルから機密情報を削除してください。

このガイドでは、ウェブブラウザから HAR ファイルを作成する方法を説明します。HTTP アーカイブ (HAR) ファイルとは、ブラウザによって記録された最新のネットワークアクティビティを含む JSON ファイルです。HAR ファイルを作成するには、次の手順に従います。

目次

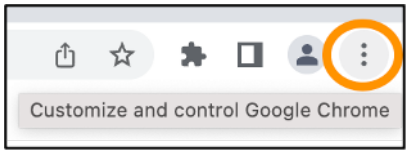
- ステップ 3: HAR ファイルをレビュー用に送信する

ステップ 1: ブラウザで HAR ファイルを作成する

> **Note**
>
> これらの手順の最新のテストは、Google Chrome バージョン 101.0.4951.64、Microsoft Edge (Chromium) バージョン 101.0.1210.47、および Mozilla Firefox バージョン 91.9 で行いました。これらのブラウザはサードパーティ製品であるため、これらの手順は最新バージョンまたは使用しているバージョンでの実際と一致しない場合があります。古い Microsoft Edge (EdgeHTML) や Apple Safari for macOS などの別のブラウザでは、HAR ファイルを生成するプロセスは似ているかもしれませんが、手順は異なります。

Google Chrome

1. ブラウザの右上にある [Customize and control Google Chrome] (Google Chrome の設定) を選択します。

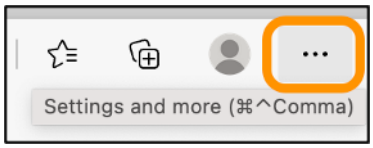


2. [More tools] (その他のツール) で、[Developer tools] (デベロッパーツール) を選択します。
3. ブラウザで DevTools を開いた状態で、[Network] (ネットワーク) パネルを選択します。
4. [Preserve log] (ログを保持) チェックボックスを選択します。
5. 現在のネットワークリクエストをすべてクリアするには、[Clear] (クリア) を選択します。
6. 直面している問題を再現します
7. DevTools で、ネットワークリクエストでコンテキスト (右クリック) メニューを開きます。
8. [Save all as HAR with content] (コンテンツと一緒に HAR としてすべて保存) を選択し、そのファイルを保存します。

詳しくは、Google Developers ウェブサイトの「Open Chrome DevTools」(Chrome DevTools を開く) と「Save all network requests to a HAR file」(すべてのネットワークリクエストを HAR ファイルに保存する) を参照してください。

Microsoft Edge (Chromium)

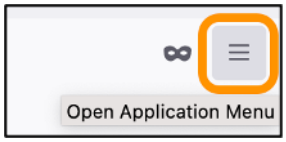
1. ブラウザの右上にある [設定など] を選択します。



2. [More tools] (その他のツール) で、[Developer tools] (デベロッパーツール) を選択します。
3. ブラウザで DevTools を開いた状態で、[Network] (ネットワーク) パネルを選択します。
4. [Preserve log] (ログを保持) チェックボックスを選択します。
5. 現在のネットワークリクエストをすべてクリアするには、[Clear] (クリア) を選択します。
6. 直面している問題を再現します
7. DevTools で、ネットワークリクエストでコンテキスト (右クリック) メニューを開きます。
8. [Save all as HAR with content] (コンテンツと一緒に HAR としてすべて保存) を選択し、そのファイルを保存します。

Mozilla Firefox

1. ブラウザの右上にある [Open Application Menu] (アプリケーションメニューを開きます) を選択します。



2. [More tools] (その他のツール) を選択し、[Web Developer tools] (ウェブ開発ツール) を選択します。
3. [Web Developer] (ウェブ開発) メニューで、[Network] (ネットワーク) を選択します。(Firefox の一部のバージョンでは、[Web Developer] (ウェブ開発) メニューは [Tools] (ツール) メニューの中にあります)。
4. 歯車アイコンを選択し、[Persist Logs] (永続ログ) を選択します。
5. ゴミ箱アイコン ([Clear] (消去)) を選択すると、現在のネットワークリクエストがすべてクリアされます。
6. 直面している問題を再現します。
7. [Network Monitor] (ネットワークモニター) で、リクエストリスト内のネットワークリクエストでコンテキストメニュー (右クリック) を開きます。
8. [Save All As HAR] (HAR 形式ですべて保存) を選択し、ファイルを保存します。

ステップ 2: HAR ファイルを編集して機密情報を削除する

1. テキストエディタアプリケーションで HAR ファイルを開きます。
2. テキストエディタの検索および置換ツールを使用して、HAR ファイルに取り込まれたすべての機密情報を特定して置換します。これには、ファイルの作成時にブラウザに入力したユーザー名、パスワード、およびキーが含まれます。
3. 編集した HAR ファイルを、機密情報を削除した状態で保存します。

ステップ 3: HAR ファイルをレビュー用に送信する

1. AWS Support Center Console の [サポートケースをオープンする] で、サポートケースを選択します。
2. サポートケースで、希望の連絡オプションを選択し、編集した HAR ファイルをアタッチして送信します。

Lightsail インスタンスを強制停止する

まれに、インスタンスが `Stopping` 状態でスタックすることがあります。これが発生した場合は、Lightsail インスタンスをホストする基盤となるハードウェアに問題がある可能性があります。このガイドでは、`stopping` 状態でスタックしたインスタンスを強制停止する方法を説明します。インスタンスの状態の詳細については、「Amazon Lightsail インスタンスの開始、停止、または再起動」を参照してください。

インスタンスを強制停止する方法

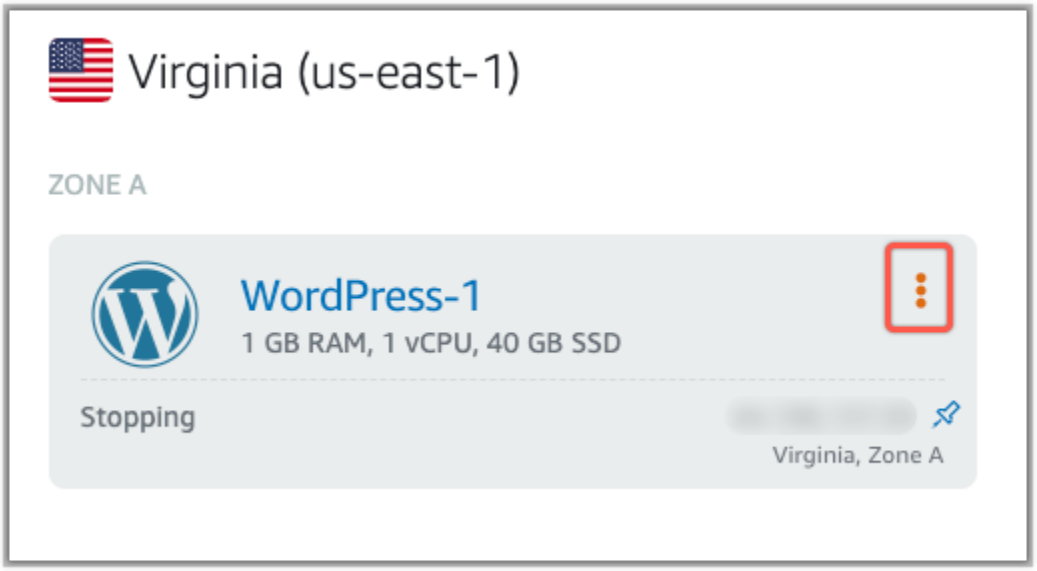
Lightsail コンソールを使用してインスタンスを強制停止できますが、これはインスタンスが `stopping` 状態にある間のみ可能です。または、インスタンスが `shutting-down` と `terminated` の状態にあるとき以外であれば、AWS Command Line Interface (AWS CLI) を使用してインスタンスを強制停止することもできます。強制停止が完了するまでに数分かかる場合があります。10 分経ってもインスタンスが停止しない場合は、再度強制停止します。

インスタンスが強制的に停止される際に、ファイルシステムのキャッシュやファイルシステムのメタデータをフラッシュする機会はありません。インスタンスを強制停止した後、ファイルシステムのチェックと修復手順を実行する必要があります。

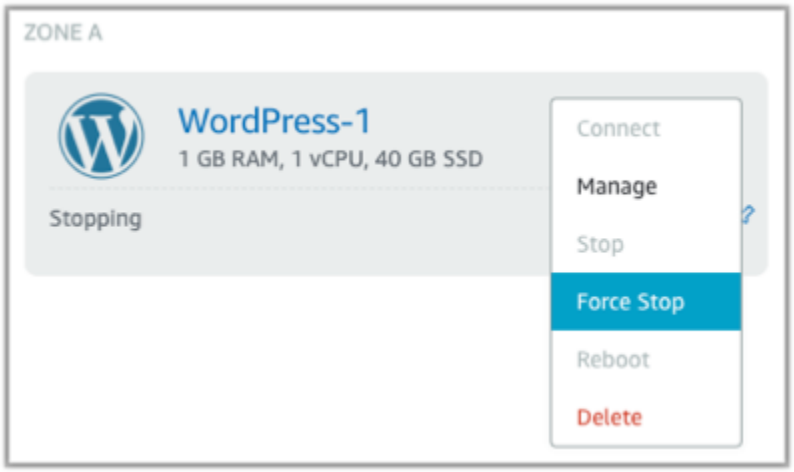
次の手順では、Lightsail インスタンスを強制停止するさまざまな方法について説明します。

Lightsail コンソールでインスタンスを強制停止する

1. Lightsail コンソールにサインインします。
2. [Instances] タブを選択します。
3. Stopping 状態でスタックしているインスタンスを特定します。その後、インスタンス名の横に表示されるアクションメニューアイコン (⋮) を選択します。



4. 表示されるドロップダウンリストで [強制停止] を選択します。



あるいは、インスタンス名を選択してインスタンス管理ページにアクセスすることもできます。その後、[強制停止] ボタンを選択します。

AWS CLI を使用してインスタンスを強制停止する

1. 開始する前に、AWS CLI をインストールする必要があります。詳細については、「AWS Command Line Interface のインストール」を参照してください。インストール後は必ず AWS CLI を設定してください。
2. stop-instance コマンドと `--force` パラメータを次のように使用します。

```
aws lightsail stop-instance --instance-name Wordpress-1 --force
```

チュートリアル: Linux ベースの Lightsail インスタンスに Prometheus をインストールする

Prometheus は、さまざまなシステムリソースとアプリケーションを管理するためのオープンソースの時系列監視ツールです。多次元データモデル、収集されたデータのクエリ機能、Grafana による詳細なレポート作成とデータの視覚化を提供します。

デフォルトでは、Prometheus はインストール先のサーバーでメトリクスを収集できるようになっています。ノードエクスポータを使用すると、ウェブサーバー、コンテナ、データベース、カスタムアプリケーション、その他のサードパーティシステムなどの他のリソースからメトリクスを収集できます。このチュートリアルでは、Lightsail インスタンスでノードエクスポーターを使用して Prometheus をインストールして設定する方法を説明します。使用可能なエクスポーターの詳細なリストについては、Prometheus ドキュメントの「Exporters and integrations」(エクスポーターとインテグレーション) を参照してください。

目次

ステップ 1: 前提条件を満たす

Prometheus を Amazon Lightsail インスタンスにインストールするには、次の作業を行う必要があります。

- Lightsail のインスタンスを作成します。インスタンスには Ubuntu 20.04 LTS ブループリントを使用することをお勧めします。詳細については、「Amazon Lightsail でインスタンスを作成する」を参照してください。
- 静的 IP アドレスを作成して新規インスタンスにアタッチします。詳細については、「Amazon Lightsail で静的 IP アドレスを作成する」を参照してください。
- 新しいインスタンスのファイアウォールのポート 9090 と 9100 を開きます。Prometheus では、ポート 9090 と 9100 が開いている必要があります。詳細については、「Amazon Lightsail でインスタンスファイアウォールルールの追加および編集」を参照してください。

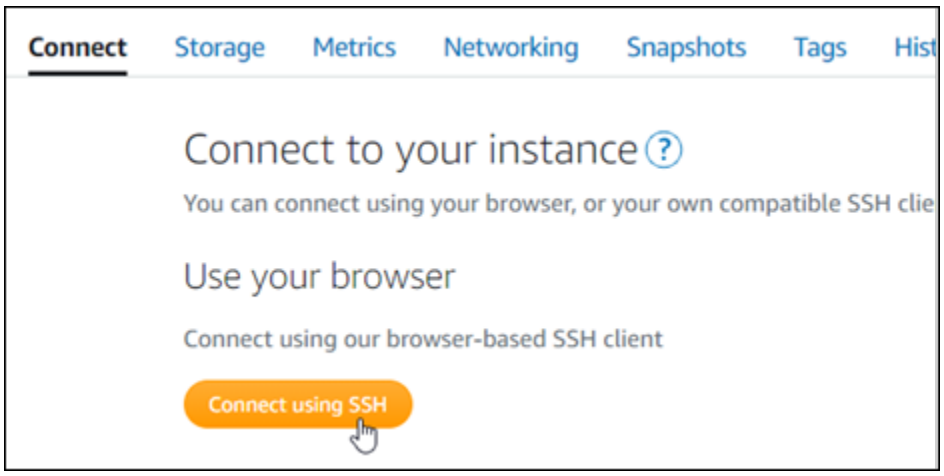
ステップ 2: Lightsail インスタンスにユーザーとローカルシステムディレクトリを追加する

以下の手順を実行して Lightsail インスタンスに接続し SSH を使用してユーザーとシステムディレクトリを追加します。この手順では、次の Linux ユーザーアカウントを作成します。

- `prometheus` – このアカウントは、サーバー環境のインストールと構成に使用されます。
- `exporter` – このアカウントは、`node_exporter` 拡張の構成に使用されます。

これらのユーザーアカウントは管理のみを目的として作成されるため、この設定の範囲を超える追加のユーザーサービスや権限は必要ありません。この手順では、Prometheus がリソースを監視するために使用するファイル、サービス設定、およびデータを保存および管理するためのディレクトリも作成します。

1. Lightsail コンソールにサインインします。
2. インスタンス管理ページの [接続] タブで、[SSH を使用して接続] を選択します。



3. 接続後に、次のコマンドを 1 つずつ入力して、2 つの Linux ユーザーアカウント (`prometheus` および `exporter`) を作成します。

```
sudo useradd --no-create-home --shell /bin/false prometheus
```

```
sudo useradd --no-create-home --shell /bin/false exporter
```

4. 次のコマンドを 1 つずつ入力して、ローカルシステムディレクトリを作成します。

```
sudo mkdir /etc/prometheus /var/lib/prometheus
```

```
sudo chown prometheus:prometheus /etc/prometheus
```

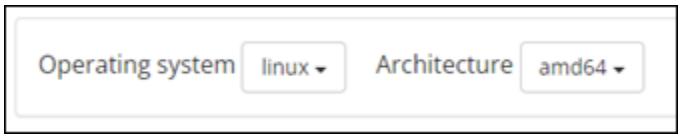
```
sudo chown prometheus:prometheus /var/lib/prometheus
```

ステップ 3: Prometheus バイナリパッケージをダウンロードする

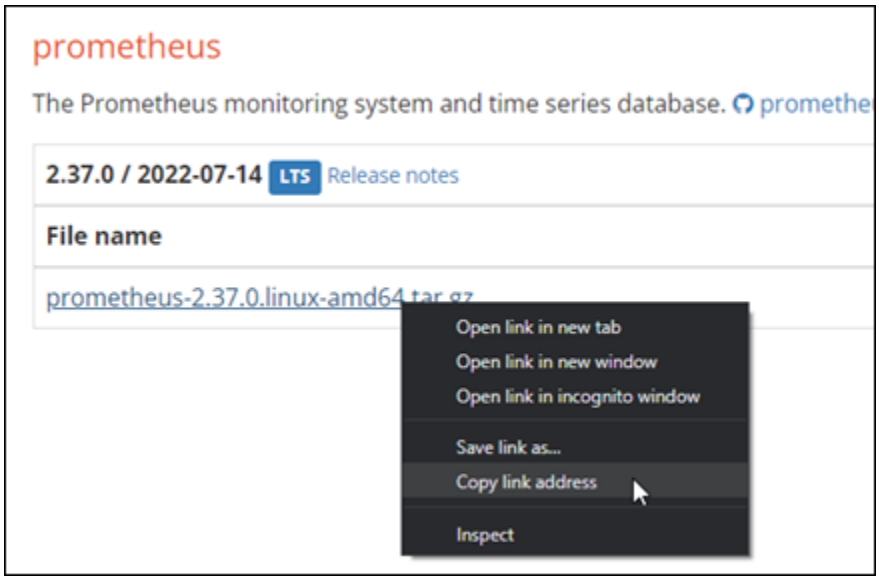
以下の手順を実行して、Prometheus バイナリパッケージを Lightsail インスタンスにダウンロードします。

1. ローカルコンピュータでウェブブラウザを開き、Prometheus のダウンロードページに移動します。

2. ページの上部で、[Operating system] (オペレーティングシステム) ドロップダウンから [Linux] を選択します。[Architecture] (アーキテクチャ) で [amd64] を選択します。



3. 表示される [Prometheus] ダウンロードリンクを選択または右クリックし、リンクアドレスをコンピュータ上のテキストファイルにコピーします。表示される [node_exporter] ダウンロードリンクにも同じ操作を行います。この手順の後半で、コピーした両方のアドレスを使用します。



4. SSH を使用して Lightsail インスタンスに接続します。
5. 次のコマンドを入力して、ディレクトリをホームディレクトリに変更します。

```
cd ~
```

6. 以下のコマンドを入力して、Prometheus バイナリパッケージをインスタンスにダウンロードします。

```
curl -LO prometheus-download-address
```

prometheus-download-address を、この手順で先ほどでコピーしたアドレスに置き換えます。アドレスの追加時は、コマンドは次の例のようになります。

```
curl -LO https://github.com/prometheus/prometheus/releases/download/v2.37.0/
prometheus-2.37.0.linux-amd64.tar.gz
```

7. 以下のコマンドを入力して、node_exporter バイナリパッケージをインスタンスにダウンロードします。

```
curl -LO node_exporter-download-address
```

node_exporter-download-address を、この手順の前のステップでコピーしたアドレスに置き換えます。アドレスの追加時は、コマンドは次の例のようになります。

```
curl -LO https://github.com/prometheus/node_exporter/releases/download/v1.3.1/
node_exporter-1.3.1.linux-amd64.tar.gz
```

8. 次のコマンドを 1 つずつ実行して、ダウンロードされた Prometheus と Node Exporter ファイルの内容を抽出します。

```
tar -xvf prometheus-2.37.0.linux-amd64.tar.gz
```

```
tar -xvf node_exporter-1.3.1.linux-amd64.tar.gz
```

ダウンロードしたファイルの内容が抽出された後、いくつかのサブディレクトリが作成されます。

9. 次のコマンドを 1 つずつ入力して、prometheus および promtool の抽出されたファイルを /usr/local/bin プログラムのディレクトリにコピーします。

```
sudo cp -p ./prometheus-2.37.0.linux-amd64/prometheus /usr/local/bin
```

```
sudo cp -p ./prometheus-2.37.0.linux-amd64/promtool /usr/local/bin
```

10. 次のコマンドを入力して、prometheus および promtool のファイルをこのチュートリアルの前半で作成した prometheus ユーザーに変更します。

```
sudo chown prometheus:prometheus /usr/local/bin/prom*
```

11. 次のコマンドを 1 つずつ入力して、consoles と console_libraries のサブディレクトリを /etc/prometheus にコピーします。-r オプションは階層内のすべてのディレクトリを再帰的にコピーします。

```
sudo cp -r ./prometheus-2.37.0.linux-amd64/consoles /etc/prometheus
```

```
sudo cp -r ./prometheus-2.37.0.linux-amd64/console_libraries /etc/prometheus
```

12. 次のコマンドを 1 つずつ入力して、コピーしたファイルの所有権をこのチュートリアルの前半で作成した prometheus ユーザーに変更します。-R オプションは階層内のすべてのファイルおよびディレクトリの所有権を再帰的に変更します。

```
sudo chown -R prometheus:prometheus /etc/prometheus/consoles
```

```
sudo chown -R prometheus:prometheus /etc/prometheus/console_libraries
```

13. 次のコマンドを 1 つずつ入力して、設定ファイル prometheus.yml を /etc/prometheus ディレクトリにコピーし、コピーしたファイルの所有権をこのチュートリアルの前半で作成した prometheus ユーザーに変更します。

```
sudo cp -p ./prometheus-2.37.0.linux-amd64/prometheus.yml /etc/prometheus
```

```
sudo chown prometheus:prometheus /etc/prometheus/prometheus.yml
```

14. 以下のコマンドを入力して、./node_exporter* サブディレクトリから /usr/local/bin プログラムのディレクトリに node_exporter ファイルをコピーします。

```
sudo cp -p ./node_exporter-1.3.1.linux-amd64/node_exporter /usr/local/bin
```

15. 次のコマンドを入力して、このチュートリアルの前半で作成した exporter ユーザーにファイルの所有権を変更します。

```
sudo chown exporter:exporter /usr/local/bin/node_exporter
```

ステップ 4: Prometheus を設定する

Prometheus を設定するには、次の手順を実行します。この手順では、prometheus.yml ファイルを開いて編集します。このファイルには、Prometheus ツールのさまざまな設定が含まれています。Prometheus は、ファイルに設定した設定に基づいて監視環境を確立します。

1. SSH を使用して Lightsail インスタンスに接続します。
2. prometheus.yml ファイルを開いて編集する前に、次のコマンドを入力してこのファイルのバックアップコピーを作成します。

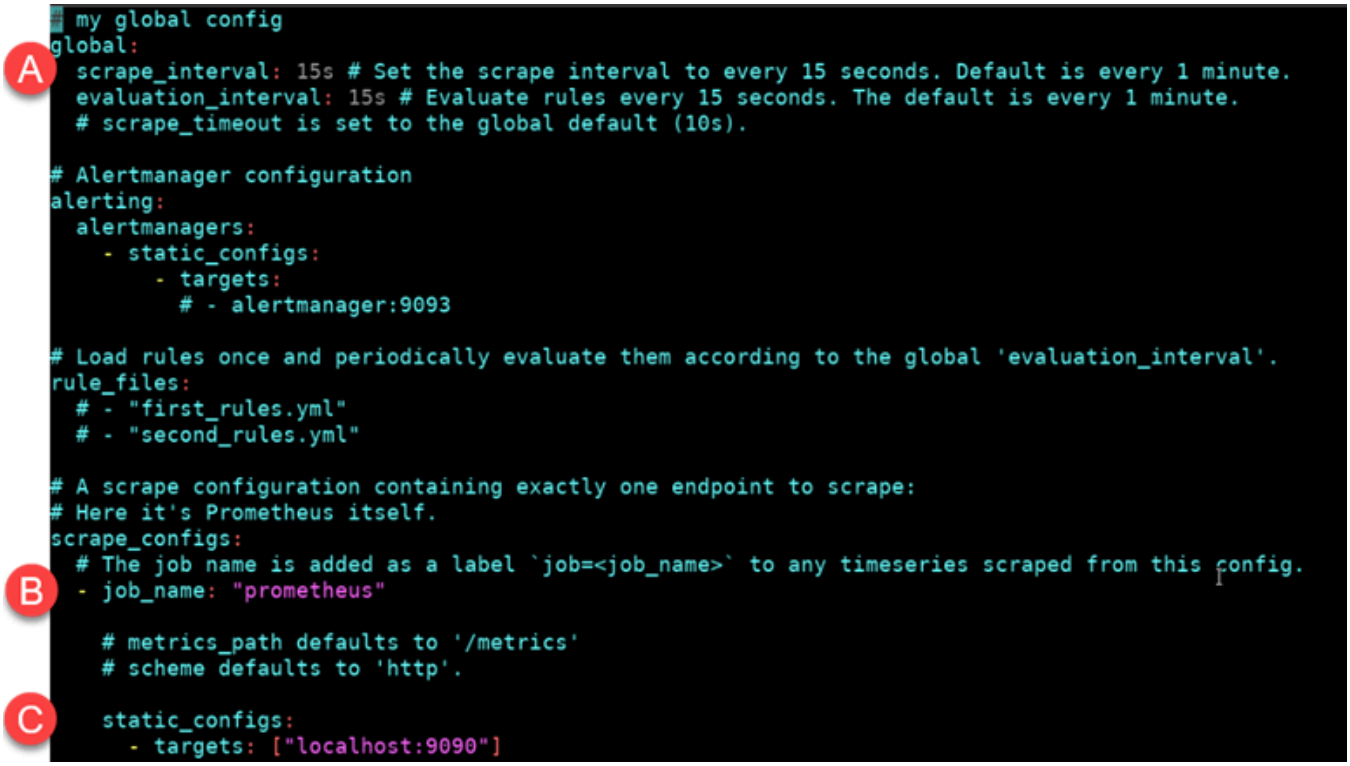
```
sudo cp /etc/prometheus/prometheus.yml /etc/prometheus/prometheus.yml.backup
```

3. 次のコマンドを入力して、 Vim を使用し、prometheus.yml ファイルを開きます。

```
sudo vim /etc/prometheus/prometheus.yml
```

以下に、prometheus.yml ファイルに設定する必要があるいくつかの重要なパラメータを示します。

- scrape_interval — このパラメータは、global ヘッダーの下に置かれ、Prometheus が特定のターゲットのメトリックスデータをどの頻度で収集するか、またはスクレイプするかの時間間隔 (秒) を定義します。global タグで示されているように、この設定は Prometheus が監視するすべてのリソースに共通です。この設定は、個々のエクスポーターがグローバル値をオーバーライドする別の値を提供しない限り、エクスポーターにも適用されます。このパラメータは、現在の 15 秒に設定したままにしておくことができます。
- job_name — scrape_configs ヘッダー下に配置されるこのパラメータは、データクエリまたはビジュアルディスプレイの結果セット内のエクスポーターを識別するラベルです。ジョブ名の値は、環境内で監視されているリソースを最もよく反映するように指定できます。たとえば、ウェブサイトを管理するジョブに business-web-app というラベルを付けたり、データベースに mysql-db-1 というラベルを付けることができます。この初期設定では、Prometheus サーバーのみを監視しているので、最新の prometheus 値を保つことができます。
- targets — static_configs ヘッダー下に配置される targets 設定では、特定のエクスポーターが実行されている場所を識別するために ip_addr:port キーバリューのペアを使用します。この手順のステップ 4 ~ 7 で、デフォルト設定を変更できます。



> **Note**
>
> この初期セットアップでは、`alerting` および `rule_files` のパラメータを設定する必要はありません。

4. Vim で開いている `prometheus.yml` ファイルでは、[I] キーを押して Vim 挿入モードに移ります。
5. `static_configs` ヘッダーの下に置かれている `targets` パラメータをスクロールして見つけます。
6. デフォルト設定を *<ip_addr>*:9090 に変更します。インスタンスの静的 IP アドレスを *<ip_addr>* に置き換えます。変更されたパラメータは、次の例のようになります。

```
static_configs:
  - targets: ["192.0.2.0:9090"]
```

7. [Esc] キーを押して挿入モードを終了し、[:wq!] と入力して変更内容を保存して Vim を終了します。
8. (オプション) 何か問題が発生した場合は、次のコマンドを入力してこの手順で前に作成したバックアップと `prometheus.yml` を置き換えます。

```
sudo cp /etc/prometheus/prometheus.yml.backup /etc/prometheus/prometheus.yml
```

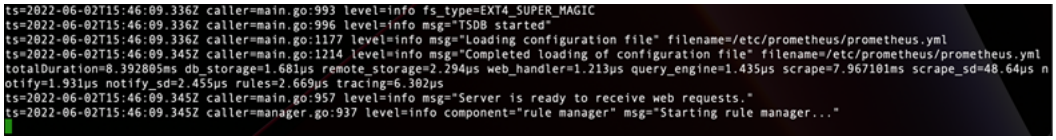
ステップ 5: Prometheus をスタートする

インスタンスで Prometheus サービスを開始するには、次のステップを実行します。

1. SSH を使用して Lightsail インスタンスに接続します。
2. 次のコマンドを入力して Prometheus サービスを開始します。

   ```
   sudo -u prometheus /usr/local/bin/prometheus --config.file /etc/prometheus/
   prometheus.yml --storage.tsdb.path /var/lib/prometheus --web.console.templates=/
   etc/prometheus/consoles --web.console.libraries=/etc/prometheus/console_libraries
   ```

   コマンドラインは、起動プロセスやその他のサービスの詳細を出力します。また、サービスがポート 9090 でリッスンしていることも示しているはずです。

   サービスが起動しない場合、このポートでのトラフィックを許可するインスタンスファイアウォールルールの作成については、このチュートリアルの「ステップ 1: 前提条件を満たす」セクションを参照してください。その他のエラーについては、prometheus.yml ファイルを見直して構文エラーがないことを確認します。
3. 実行中のサービスが検証されたら、[Ctrl+C] を押してストップします。
4. 以下のコマンドを入力し、Vim を使用して systemd 設定ファイルを開きます。このファイルは Prometheus を起動するために使用されます。

   ```
   sudo vim /etc/systemd/system/prometheus.service
   ```
5. ファイルに以下の行を挿入します。

   ```
   [Unit]
   Description=PromServer
   Wants=network-online.target
   After=network-online.target

   [Service]
   ```

```
User=prometheus
Group=prometheus
Type=simple
ExecStart=/usr/local/bin/prometheus \
--config.file /etc/prometheus/prometheus.yml \
--storage.tsdb.path /var/lib/prometheus/ \
--web.console.templates=/etc/prometheus/consoles \
--web.console.libraries=/etc/prometheus/console_libraries

[Install]
WantedBy=multi-user.target
```

前述の手順は Linux `systemd` サービスマネージャーがサーバー上で Prometheus を起動するために使用されます。Prometheus は、呼び出されると `prometheus` ユーザーとして実行され `prometheus.yml` ファイルを参照して、設定を読み込み `/var/lib/prometheus` ディレクトリの時系列データを保存するします。コマンドラインから `man systemd` を実行し、サービスの詳細情報を確認できます。

6. [Esc] キーを押して挿入モードを終了し、[:wq!] と入力して変更内容を保存して Vim を終了します。
7. 次のコマンドを入力して、`systemd` サービスマネージャに情報を読み込みます。

```
sudo systemctl daemon-reload
```

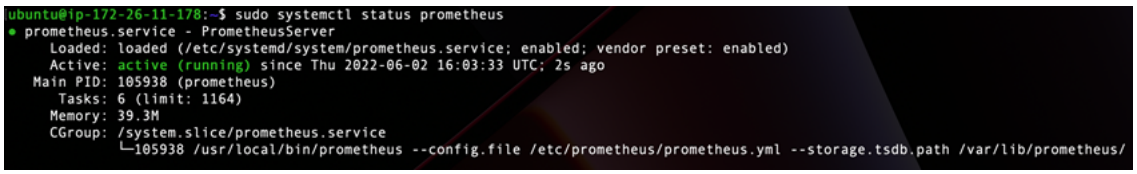
8. 次のコマンドを入力して Prometheus を再起動します。

```
sudo systemctl start prometheus
```

9. Prometheus サービスの状態を確認するには、次のコマンドを入力します。

```
sudo systemctl status prometheus
```

サービスが正常に起動された場合は、次の例のような出力が表示されます。



10. [Q] を押して、ステータスコマンドを終了します。

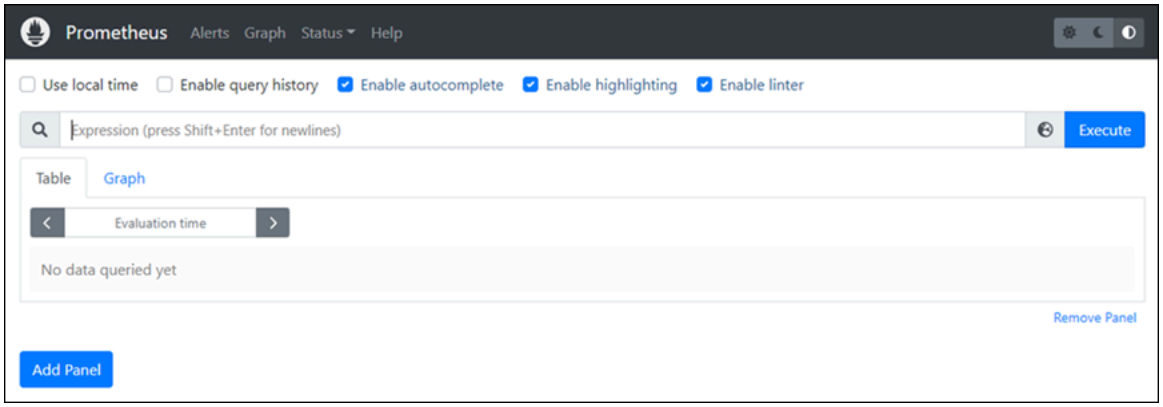
11. 次のコマンドを入力して、インスタンスが起動されたときに Prometheus が起動できるようにします。

```
sudo systemctl enable prometheus
```

12. ローカルコンピュータで Web ブラウザを開き、次の Web アドレスにアクセスして Prometheus 管理インターフェイスを表示します。

```
http:<ip_addr>:9090
```

<ip_addr> を Lightsail インスタンスの静的 IP アドレスに置き換えます。次の例に示すようなダッシュボードが表示されます。



ステップ 6: Node Exporter をスタートする

Node Exporter サービスを開始するには、以下の手順を実行します。

1. SSH を使用して Lightsail インスタンスに接続します。
2. 次のコマンドを入力し、Vim を使用して node_exporter の systemd サービスファイルを作成します。

```
sudo vim /etc/systemd/system/node_exporter.service
```

3. [I] キーを押して、Vim を挿入モードにします。
4. ファイルの末尾に次の行を追加します。これにより、CPU 負荷、ファイルシステムの使用状況、およびメモリリソースの監視コレクターを使用して node_exporter を設定します。

```
[Unit]
Description=NodeExporter
```

```
Wants=network-online.target
After=network-online.target

[Service]
User=exporter
Group=exporter
Type=simple
ExecStart=/usr/local/bin/node_exporter --collector.disable-defaults \
--collector.meminfo \
--collector.loadavg \
--collector.filesystem

[Install]
WantedBy=multi-user.target
```

> **Note**
>
> これらの手順では、Node Exporter のデフォルトのマシンメトリックを無効にします。Ubuntu で利用できるメトリクスの詳しいリストについては、Ubuntu ドキュメンテーションの Prometheus node_exporter マニュアルのページを参照してください。

5. [Esc] キーを押して挿入モードを終了し、[:wq!] と入力して変更内容を保存して Vim を終了します。
6. 次のコマンドを入力して、systemd プロセスをリロードします。

```
sudo systemctl daemon-reload
```

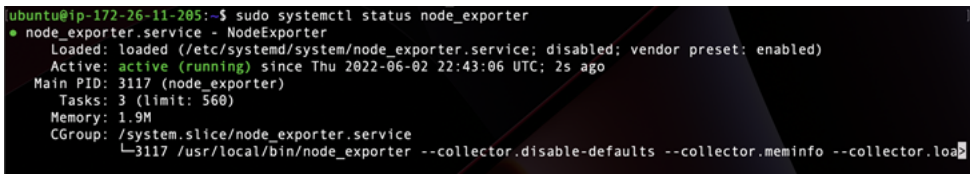
7. 次のコマンドを入力して node_exporter サービスを開始します。

```
sudo systemctl start node_exporter
```

8. node_exporter サービスの状態を確認するには、次のコマンドを入力します。

```
sudo systemctl status node_exporter
```

サービスが正常に起動された場合は、次の例のような出力が表示されます。



9. [Q] を押して、ステータスコマンドを終了します。
10. 次のコマンドを入力して、インスタンスが起動されたときに Node Exporter が起動できるようにします。

```
sudo systemctl enable node_exporter
```

ステップ 7: Node Exporter データコレクタで Prometheus を設定する

以下の手順を実行して、Node Exporter データコレクタで Prometheus を設定します。そのためには、prometheus.yml ファイルの node_exporter に新しい job_name パラメータを追加します。

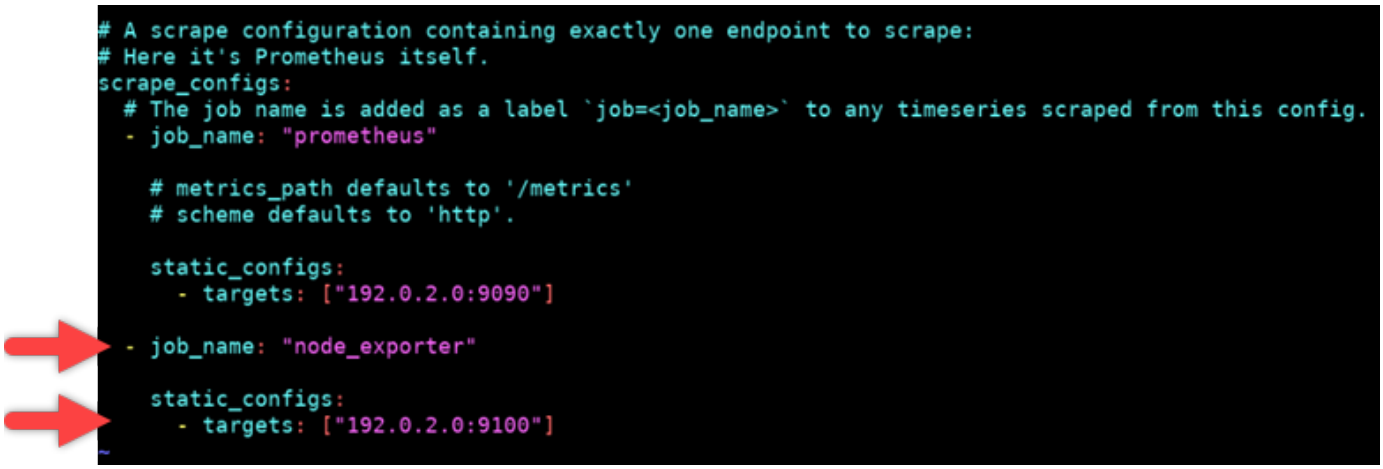
1. SSH を使用して Lightsail インスタンスに接続します。
2. 次のコマンドを入力して、 Vim を使用し、prometheus.yml ファイルを開きます。

```
sudo vim /etc/prometheus/prometheus.yml
```

3. [I] キーを押して、Vim を挿入モードにします。
4. 既存の - targets: ["*<ip_addr>*:9090"] パラメータの下で、次のテキスト行をファイルに追加します。

```
- job_name: "node_exporter"

static_configs:
- targets: ["<ip_addr>:9100"]
```

prometheus.yml ファイルの変更されたパラメータは、次の例のようになります。



次の点に注意してください。

- Node Exporter は、prometheus サーバーのポート 9100 をリッスンしてデータをスクレイピングします。このチュートリアルの「Step 1: Complete the prerequisites」(ステップ 1: 前提条件を満たす) セクションで説明されているように、インスタンスのファイアウォールルールを作成する手順に従っていることを確認します。
- prometheus job_name の設定と同様に、Lightsail インスタンスに添付される性的 IP アドレスと *<ip_addr>* を置き換えます。

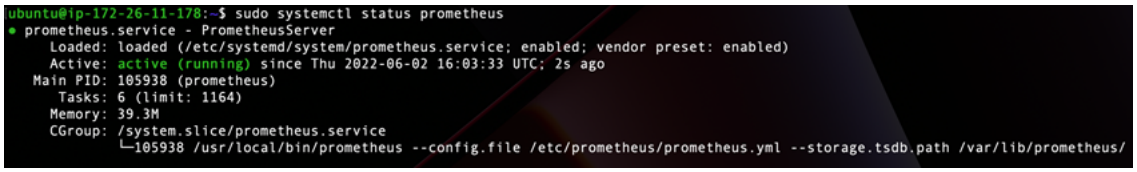
5. [Esc] キーを押して挿入モードを終了し、[:wq!] と入力して変更内容を保存して Vim を終了します。
6. 以下のコマンドを入力して Prometheus サービスを再起動し、設定ファイルへの変更を確定します。

```
sudo systemctl restart prometheus
```

7. Prometheus サービスの状態を確認するには、次のコマンドを入力します。

```
sudo systemctl status prometheus
```

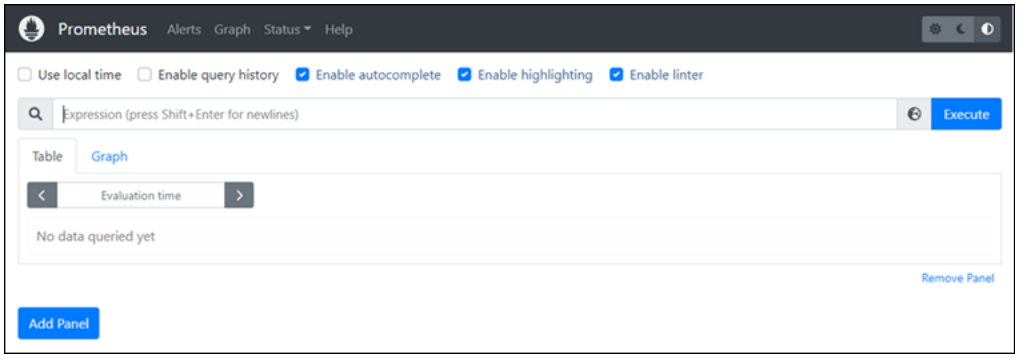
サービスが正常に再起動された場合は、次のような出力が表示されます。



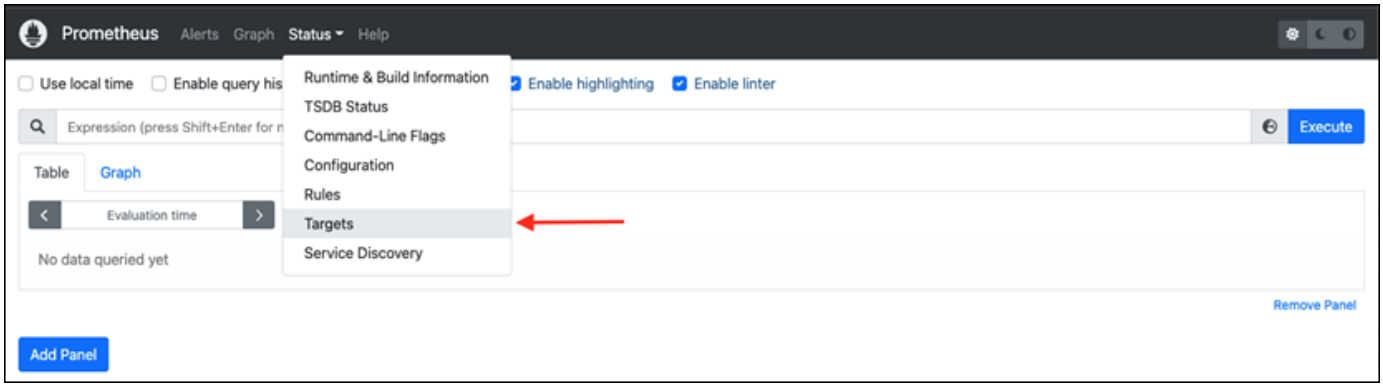
8. [Q] を押して、ステータスコマンドを終了します。
9. ローカルコンピュータで Web ブラウザを開き、次の Web アドレスにアクセスして Prometheus 管理インターフェイスを表示します。

```
http:<ip_addr>:9090
```

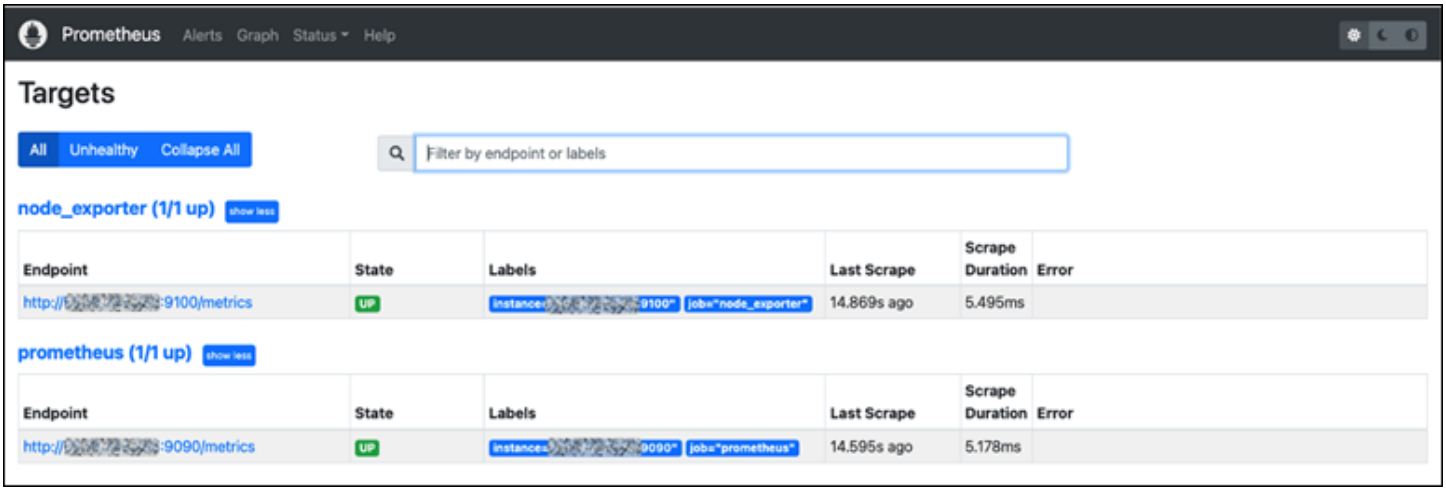
<ip_addr> を Lightsail インスタンスの静的 IP アドレスに置き換えます。次の例に示すようなダッシュボードが表示されます。



10. メインメニューで、[Status] (ステータス) ドロップダウンを選択し、[Targets] (ターゲット) を選択します。



次の画面には、2 つのターゲットが表示されます。最初のターゲットは [node_exporter] メトリクスコレクターのジョブで、2 つ目のターゲットは [Prometheus] ジョブです。



これで、メトリックの収集とサーバーの監視のための環境が適切に設定されました。

チュートリアル: Lightsail LAMP インスタンスを起動して設定する

Amazon Lightsail は、仮想プライベートサーバーだけが必要な場合に、Amazon Web Services (AWS) の使用を開始する最も簡単な方法です。Lightsail には、仮想マシン、SSD ベースのストレージ、データ転送、DNS 管理、静的 IP など、プロジェクトをすばやく起動するために必要なすべてが含まれており、予測可能な低価格で提供されます。

このチュートリアルでは、Lightsail で LAMP インスタンスを起動して設定する方法を示します。SSH 経由でのインスタンスへの接続、インスタンスのアプリケーションパスワードの取得、静的 IP の作成とインスタンスへのアタッチ、DNS ゾーンの作成とドメインのマッピングに関するステップが含まれています。このチュートリアルを終了すると、Lightsail でインスタンスを起動して実行するための基礎が得られます。

目次

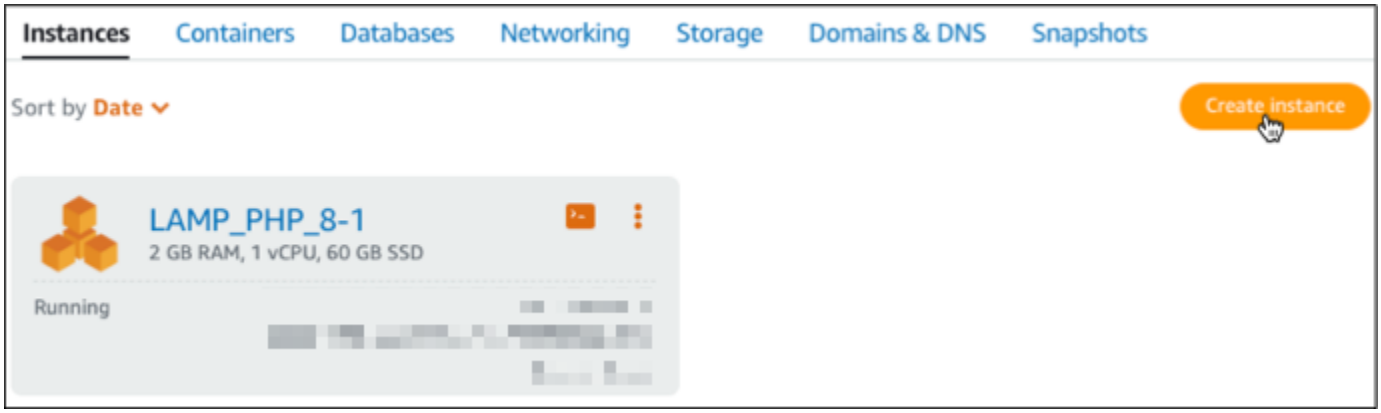
ステップ 1: AWS にサインアップ

このチュートリアルでは、 AWS アカウントが必要です。にサインアップ AWSするか、 アカウントを既にお持ちの場合は にサインイン AWSします。

ステップ 2: LAMP インスタンスを作成する

Lightsail で LAMP インスタンスを起動して実行します。Lightsail でのインスタンスの作成の詳細については、Lightsail ドキュメントの「Amazon Lightsail インスタンスの作成」を参照してください。

1. Lightsail コンソールにサインインします。
2. Lightsail ホームページのインスタンスタブで、インスタンスの作成を選択します。

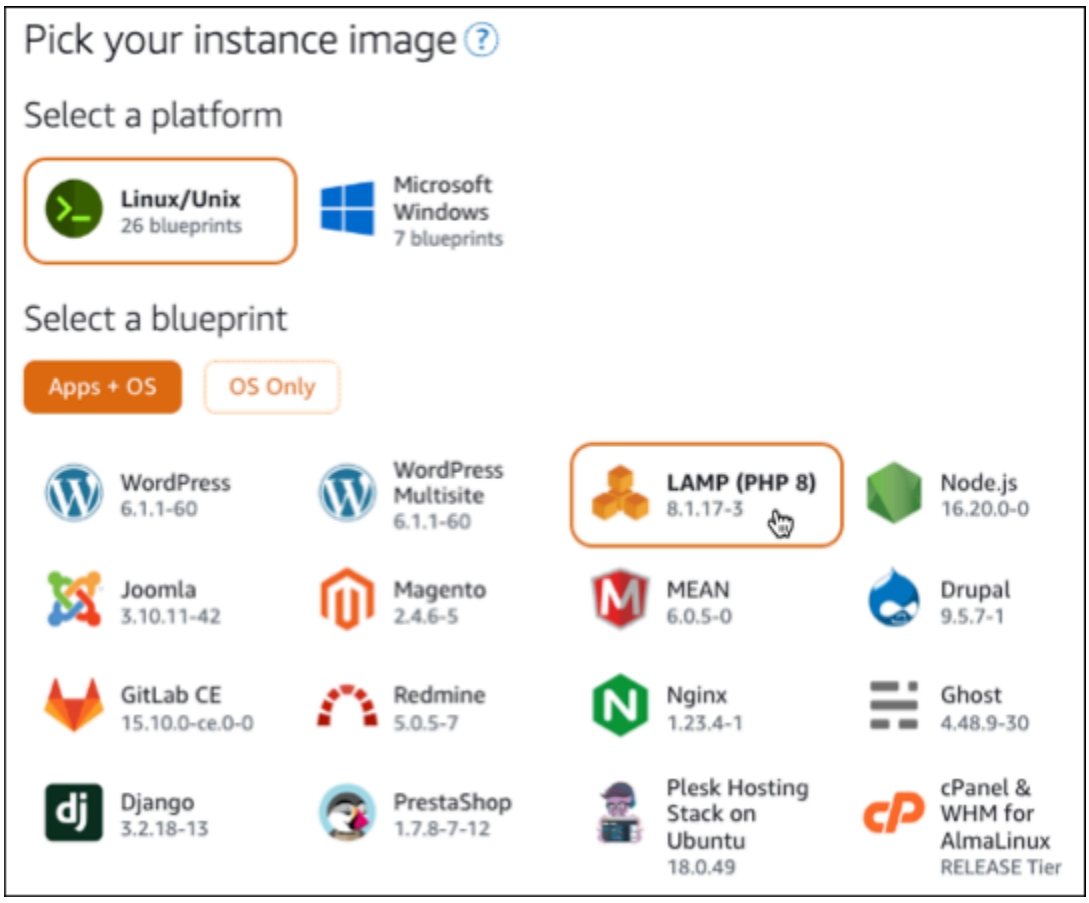


3. インスタンスの AWS リージョン とアベイラビリティーゾーンを選択します。

4. インスタンスイメージを選択します。

   a. プラットフォームとして [Linux/Unix] を選択します。

   b. ブループリントとして [LAMP (PHP 8)] を選択します。



5. インスタンスプランを選択します。

   プランには、低額で予測可能なコスト、マシン設定 (RAM、SSD、vCPU)、およびデータ転送枠が含まれます。3.50 USD Lightsail プランは、1 か月間 (最大 750 時間) 無料で試すことができます。 は、アカウントに 1 か月の無料 AWS クレジットを付与します。

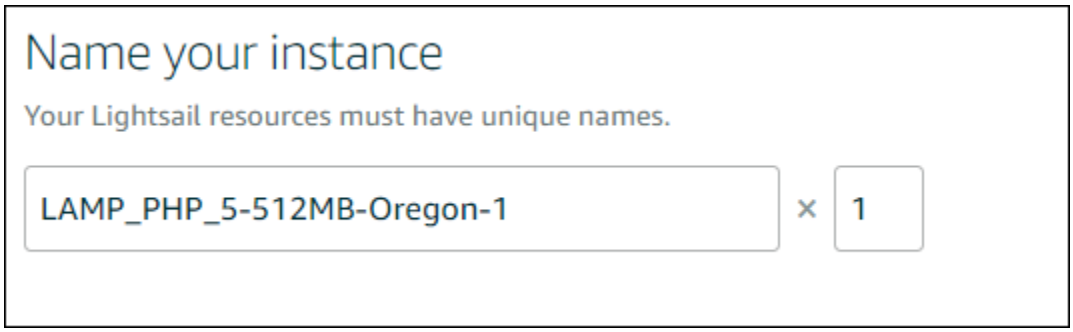
   > **Note**
   >
   > AWS 無料利用枠の一部として、選択したインスタンスバンドルで Amazon Lightsail を無料で使い始めることができます。詳細については、「Amazon Lightsail 料金表」ページのAWS 「 無料利用枠」を参照してください。

6. インスタンスの名前を入力します。

   リソース名:

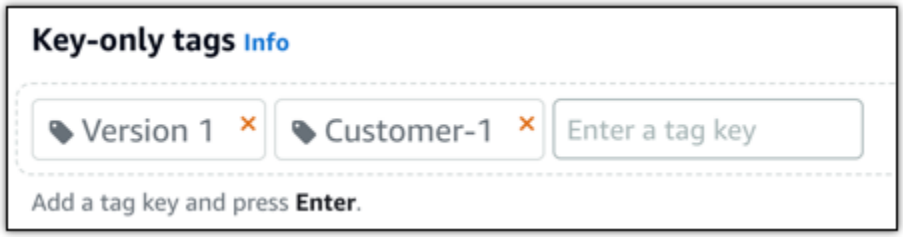
   - AWS リージョン Lightsail アカウントの各 内で一意である必要があります。

- 2〜255 文字を使用する必要があります。
- 先頭と末尾は英数字または数字を使用する必要があります。
- 英数字、数字、ピリオド、ダッシュ、アンダースコアを使用することができます。



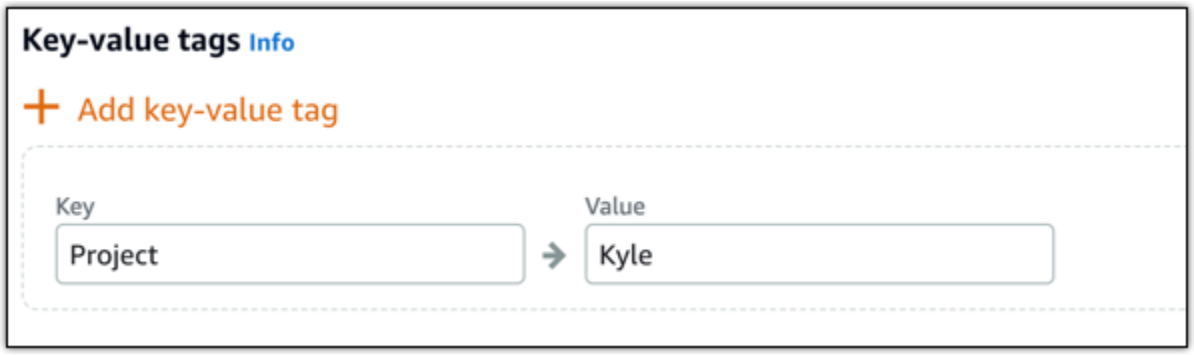
7. 以下のいずれかのオプションを選択して、インスタンスにタグを追加します。

   - [Add key-only tags] (キーのみのタグを追加) または [Edit key-only tags] (キーのみのタグを編集) (タグが追加済みの場合)を追加。タグキーのテキストボックスに新しいタグを入力し、Enter キーを押します。タグの入力を完了したら、[保存] を選択してタグを追加し、追加しない場合は、[キャンセル] を選択します。

   - [key-value タグの作成] から [キー] テキストボックスにキーを入力し、[値] テキストボックスに値を入力します。タグの入力を完了したら、[保存] を選択し、追加しない場合は、[キャンセル] を選択します。

     キーバリューのタグは、保存する際に一つずつ追加することができます。さらに key-value タグを追加するには、以上のステップを繰り返します。



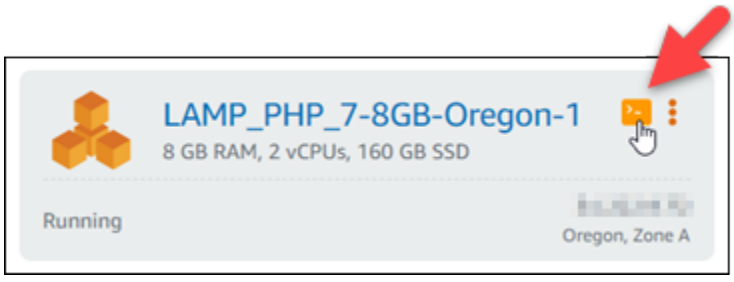
> **Note**
>
> 「キーのみ」のタグと「キーバリュー」のタグの詳細については、「タグ」を参照してください。

8. [インスタンスの作成] を選択します。

ステップ 3: SSH 経由でインスタンスに接続し、LAMP インスタンスのアプリケーションパスワードを取得します。

LAMP のデータベースにサインインするためのデフォルトのパスワードがインスタンスに保存されます。Lightsail コンソールでブラウザベースの SSH ターミナルを使用してインスタンスに接続し、特別なコマンドを実行して、インスタンスを取得します。詳細については、「Amazon Lightsail での Bitnami インスタンスのアプリケーションユーザー名とパスワードの取得」を参照してください。

1. Lightsail ホームページのインスタンスタブで、LAMP インスタンスの SSH クイック接続アイコンを選択します。



2. ブラウザベースの SSH クライアントのウィンドウが表示されたら、次のコマンドを入力してデフォルトのアプリケーションのパスワードを取得します。

```
cat bitnami_application_password
```

> **Note**
>
> ユーザーのホームディレクトリ以外のディレクトリで作業している場合は、「`cat $HOME/bitnami_application_password`」と入力します。

3. 画面に表示されたパスワードを書き留めておきます。このパスワードを後で使用して Bitnami アプリケーションをインスタンスにインストールします。または、ユーザー名 `root` を使用して MySQL データベースにアクセスします。

ステップ 4: LAMP インスタンス上にアプリケーションをインストールする

LAMP インスタンス上に PHP アプリケーションをデプロイするか、Bitnami アプリケーションをインストールします。PHP アプリケーションをデプロイするメインディレクトリは `/opt/bitnami/apache2/htdocs` です。このディレクトリに PHP アプリケーションのファイルをコピーし、インスタンスのパブリック IP アドレスを参照してアプリケーションにアクセスします。

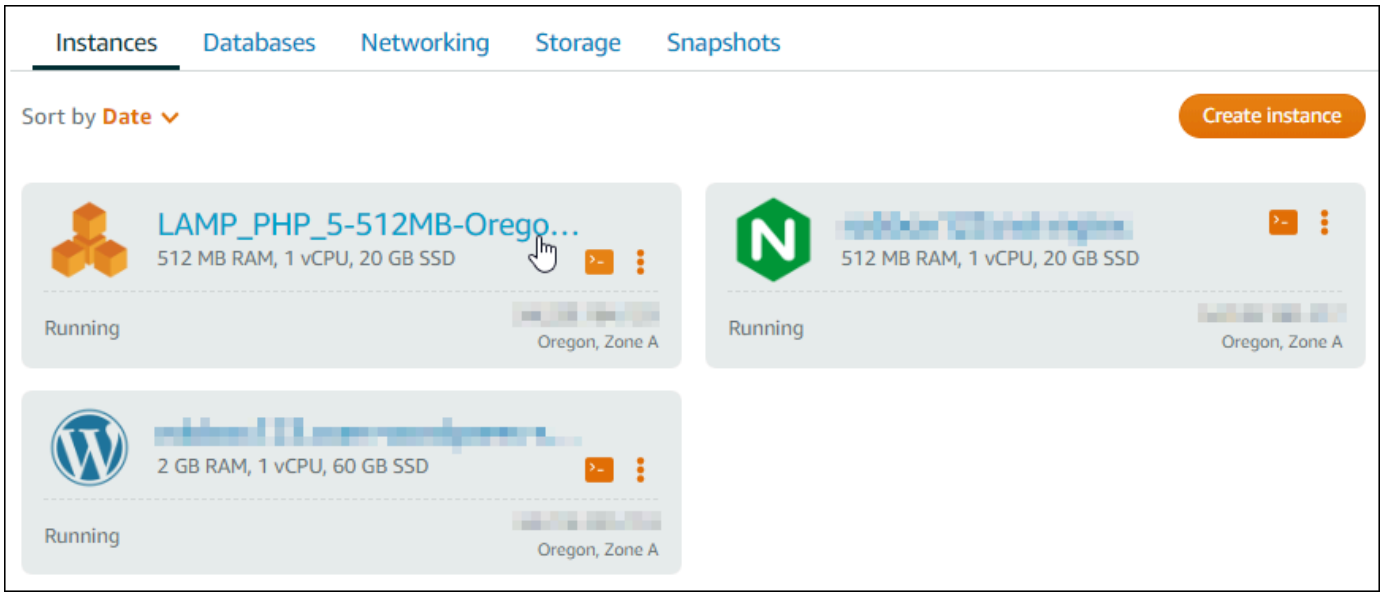
Bitnami アプリケーションは、モジュールのインストーラを使用してインストールすることもできます。Bitnami ウェブサイトから WordPress、Drupal、Magento、Moodle などのアプリケーションをダウンロードし、サーバーの機能を拡張します。Bitnami アプリケーションのインストールの詳細については、Bitnami ドキュメントの「開始方法」を参照してください。

ステップ 5: 静的 IP アドレスを作成して LAMP インスタンスにアタッチする

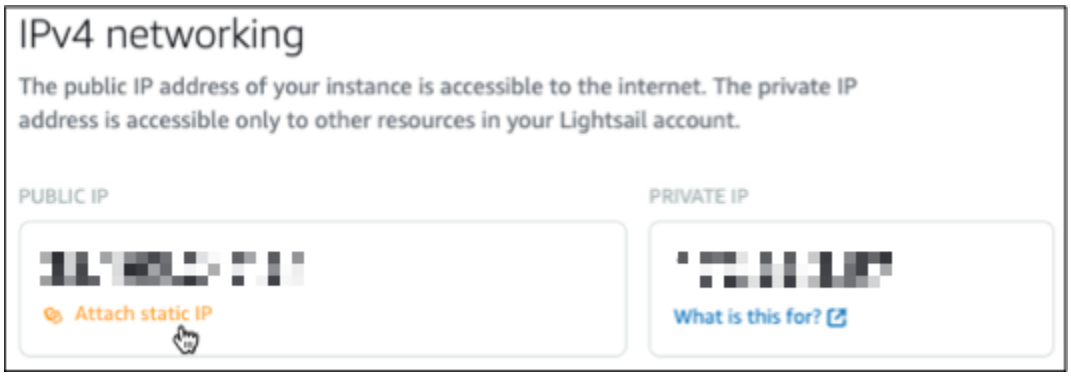
LAMP インスタンスのデフォルトのパブリック IP は、インスタンスを停止して開始すると変わります。インスタンスにアタッチした静的 IP アドレスは、インスタンスを停止して開始しても変わりません。

静的 IP アドレスを作成して LAMP インスタンスにアタッチします。詳細については、Lightsail ドキュメントの「静的 IP を作成してインスタンスにアタッチする」を参照してください。

1. Lightsail ホームページのインスタンスタブで、実行中の LAMP インスタンスを選択します。



2. [ネットワーキング] タブを開き、次に [静的 IP をアタッチする] を選択します。



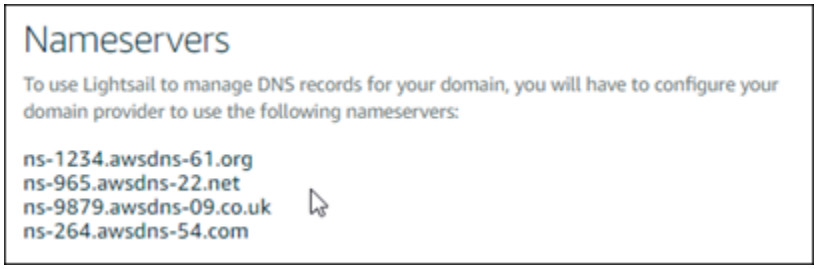
3. 静的 IP に名前を付け、[作成してアタッチ] を選択します。

ステップ 6: DNS ゾーンを作成し、ドメインを LAMP インスタンスにマッピングする

ドメインの DNS レコードの管理を Lightsail に転送します。これにより、ドメインを LAMP インスタンスにマッピングし、Lightsail コンソールを使用してウェブサイトのすべてのリソースを簡単に管理できます。詳細については、「DNS ゾーンを作成し、ドメインの DNS レコードを管理する」を参照してください。

1. Lightsail ホームページのドメインと DNS タブで、DNS ゾーンの作成を選択します。
2. ドメインを入力し、[DNS ゾーンの作成] を選択します。
3. ページに表示されたネームサーバーのアドレスを書き留めておきます。

   これらのネームサーバーアドレスをドメイン名のレジストラに追加して、ドメインの DNS レコードの管理を Lightsail に移管します。



4. ドメインの DNS レコードの管理が Lightsail に転送されたら、次のように A レコードを追加して、ドメインの頂点を LAMP インスタンスにポイントします。

   a. DNS ゾーンの [Assignments] (割り当て) タブで [Add assignment] (割り当てを追加) を選択します。

b. [Select a domain] (ドメインの選択) フィールドで、ドメインまたはサブドメインを選択します。

c. [Select a resource] (リソースの選択) ドロップダウンで、このチュートリアルで以前に作成した LAMP インスタンスを選択します。

d. [Assign] (割り当て) を選択します。

変更内容がインターネットの DNS を通じて伝播されるまで待ってから、LAMP インスタンスへのトラフィックのルーティングを開始します。

次のステップ

Amazon Lightsail で LAMP インスタンスを起動した後に実行できる追加の手順をいくつか示します。

- Linux または Unix インスタンスのスナップショットを作成する
- 追加のブロックストレージディスクを作成して Linux ベースの インスタンスにアタッチする

チュートリアル: Windows Server 2016 インスタンスを起動して設定する

Amazon Lightsail は、仮想プライベートサーバーだけが必要な場合に、Amazon Web Services (AWS) の使用を開始する最も簡単な方法です。Lightsail には、仮想マシン、SSD ベースのストレージ、データ転送、DNS 管理、静的 IP など、プロジェクトをすばやく起動するために必要なすべてが含まれており、予測可能な低価格で提供されます。

このチュートリアルでは、Lightsail で Windows Server 2016 インスタンスを起動して設定する方法を示します。RDP 経由でのインスタンスへの接続、静的 IP の作成とインスタンスへのアタッチ、DNS ゾーンの作成とドメインのマッピングに関するステップが含まれています。このチュートリアルを終了すると、Lightsail でインスタンスを起動して実行するための基礎が得られます。

目次

- ステップ 5: DNS ゾーンを作成し、ドメインを Windows Server 2016 インスタンスにマッピングする
- 次のステップ

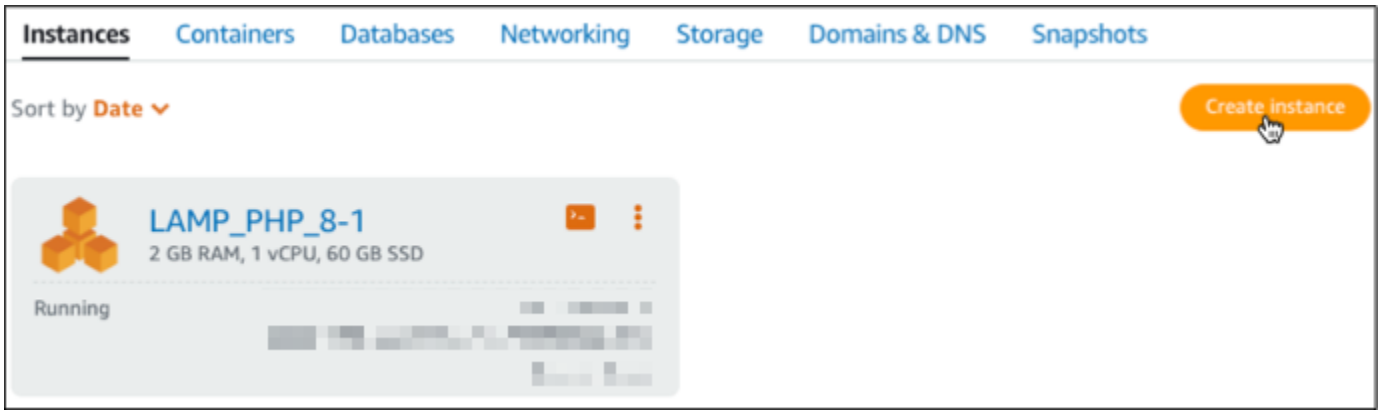
ステップ 1: AWS にサインアップ

このチュートリアルには AWS アカウントが必要です。AWS にサインアップするか、アカウントを既にお持ちの場合は AWS にサインインしてください。

ステップ 2: Lightsail で Windows Server 2016 インスタンスを作成する

Lightsail で Windows Server 2016 インスタンスを起動して実行します。詳細については、「Windows Server ベースのインスタンスの使用を開始する」を参照してください。

1. Lightsail コンソールにサインインします。
2. Lightsail ホームページのインスタンスタブで、インスタンスの作成を選択します。

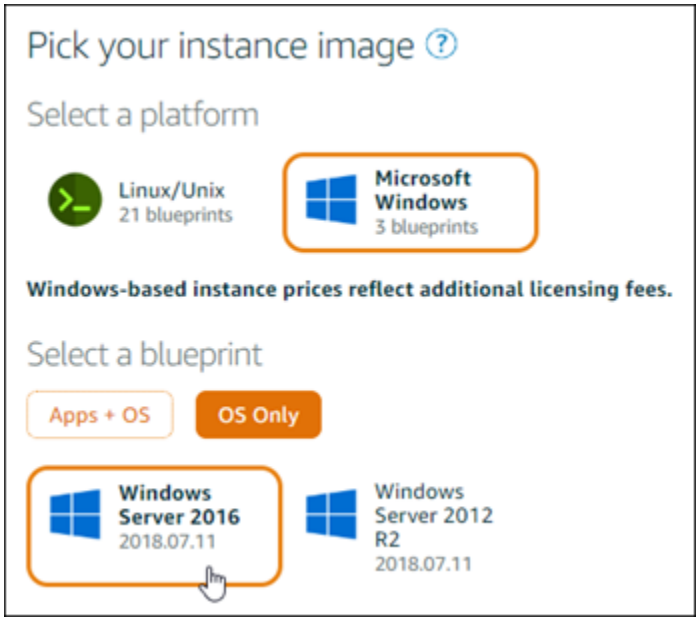


3. インスタンスの AWS リージョン およびアベイラビリティーゾーンを選択します。

4. インスタンスイメージを選択します。

   a. プラットフォームとして [Microsoft Windows] を選択します。

   b. [OS のみ] を選択し、設計図として [Windows Server 2016] を選択します。



5. インスタンスプランを選択します。

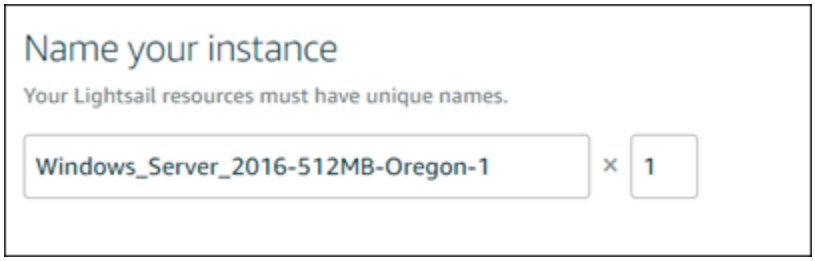
   プランには、低額で予測可能なコスト、マシン設定 (RAM、SSD、vCPU)、およびデータ転送枠が含まれます。8 USD Lightsail プランは 1 か月間 (最大 750 時間) 無料で試すことができます。は 1 か月分をアカウントにAWSクレジットします。

   > **Note**
   >
   > AWS 無料利用枠の一部として、一部のインスタンスバンドルで Amazon Lightsail を無料で使い始めることができます。詳細については、「Amazon Lightsail 料金表」ページのAWS「 無料利用枠」を参照してください。

6. インスタンスの名前を入力します。

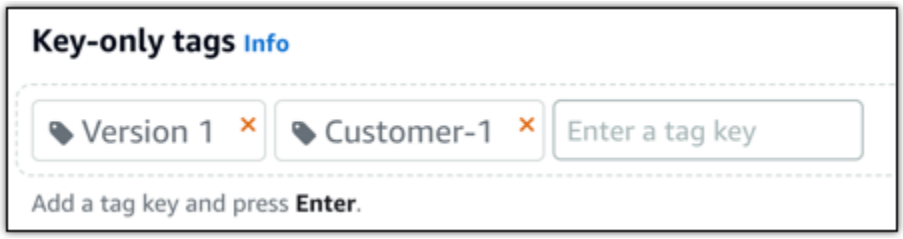
   リソース名:

   - AWS リージョン Lightsail アカウントの各 内で一意である必要があります。
   - 2〜255 文字を使用する必要があります。
   - 先頭と末尾は英数字または数字を使用する必要があります。
   - 英数字、数字、ピリオド、ダッシュ、アンダースコアを使用することができます。



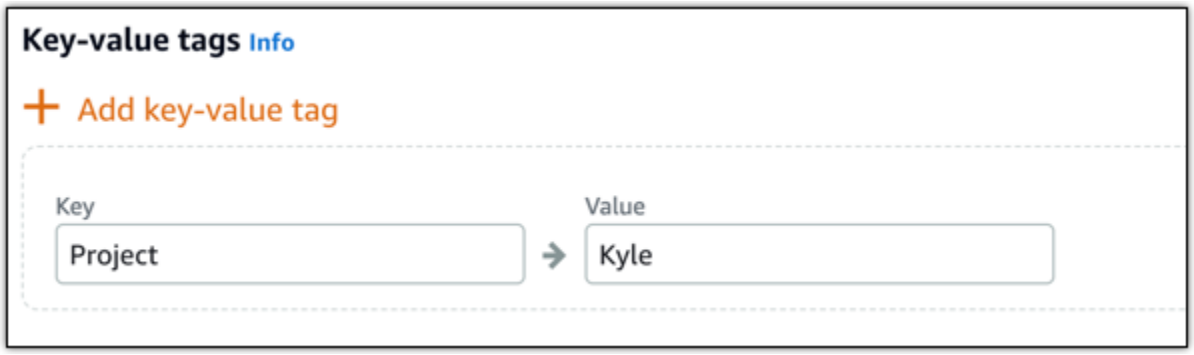
7. 以下のいずれかのオプションを選択して、インスタンスにタグを追加します。

   - [Add key-only tags] (キーのみのタグを追加) または [Edit key-only tags] (キーのみのタグを編集) (タグが追加済みの場合)を追加。タグキーのテキストボックスに新しいタグを入力し、Enter キーを押します。タグの入力を完了したら、[保存] を選択してタグを追加し、追加しない場合は、[キャンセル] を選択します。



- [key-value タグの作成] から [キー] テキストボックスにキーを入力し、[値] テキストボックスに値を入力します。タグの入力を完了したら、[保存] を選択し、追加しない場合は、[キャンセル] を選択します。

  キーバリューのタグは、保存する際に一つずつ追加することができます。さらに key-value タグを追加するには、以上のステップを繰り返します。



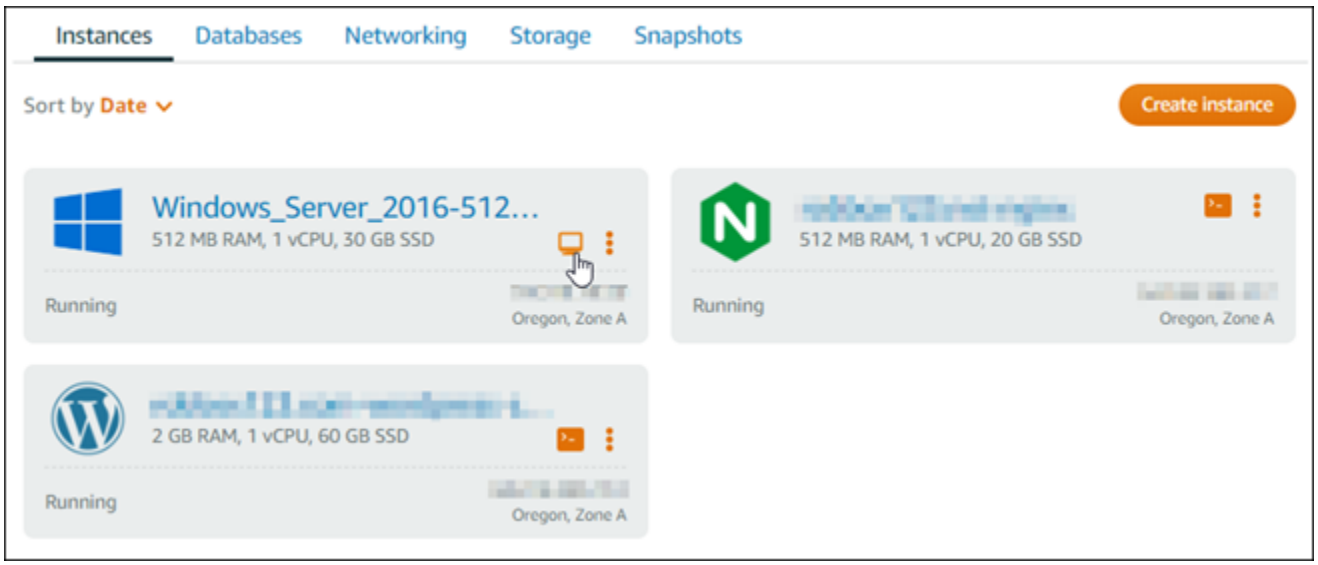
> **Note**
>
> 「キーのみ」のタグと「キーバリュー」のタグの詳細については、「タグ」を参照してください。

8. [インスタンスの作成] を選択します。

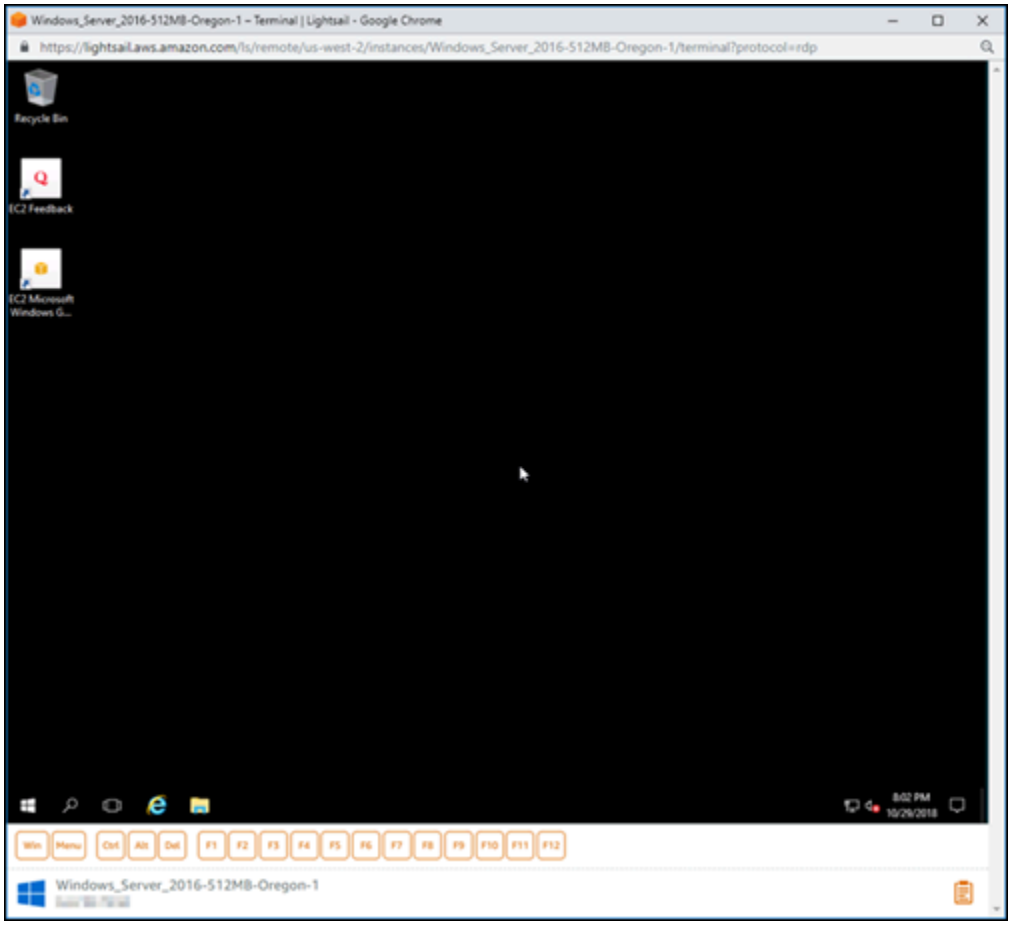
ステップ 3: RDP 経由で Windows Server 2016 インスタンスに接続する

Lightsail コンソールでブラウザベースの RDP クライアントを使用して Windows Server 2016 インスタンスに接続します。詳細については、「Windows インスタンスに接続する」を参照してください。

1. Lightsail ホームページのインスタンスタブで、Windows Server 2016 インスタンスの RDP クイック接続アイコンを選択します。



2. ブラウザベースの RDP クライアントウィンドウが表示されたら、Windows Server 2016 インスタンスの設定を開始できます。

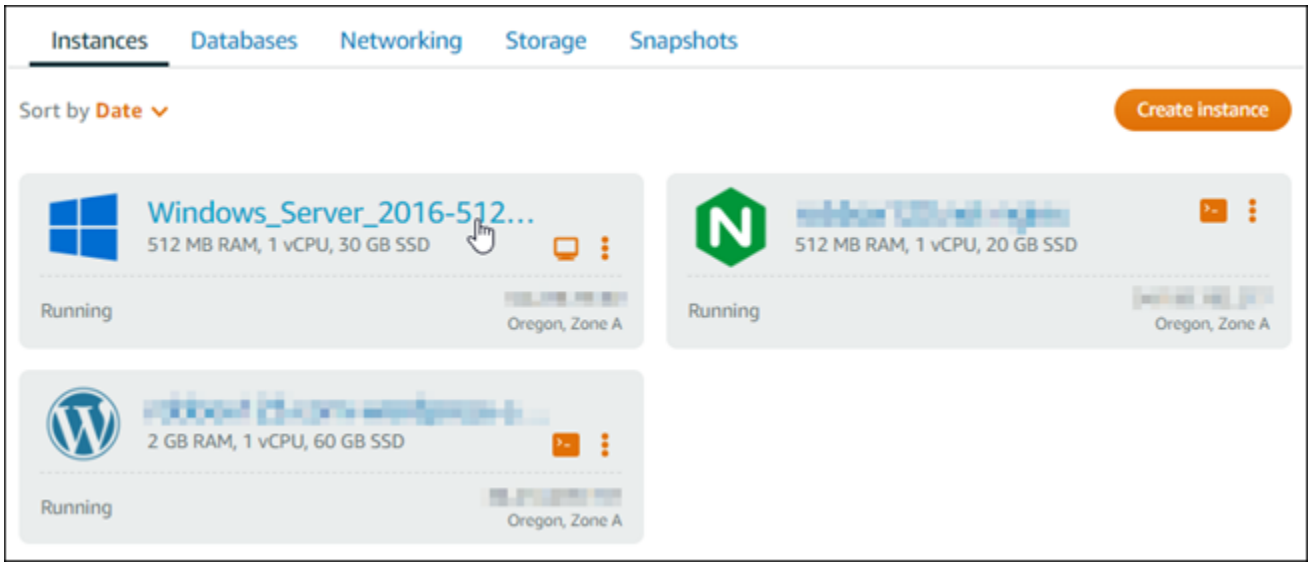


ステップ 4: 静的 IP アドレスを作成して Windows Server 2016 インスタンスにアタッチする

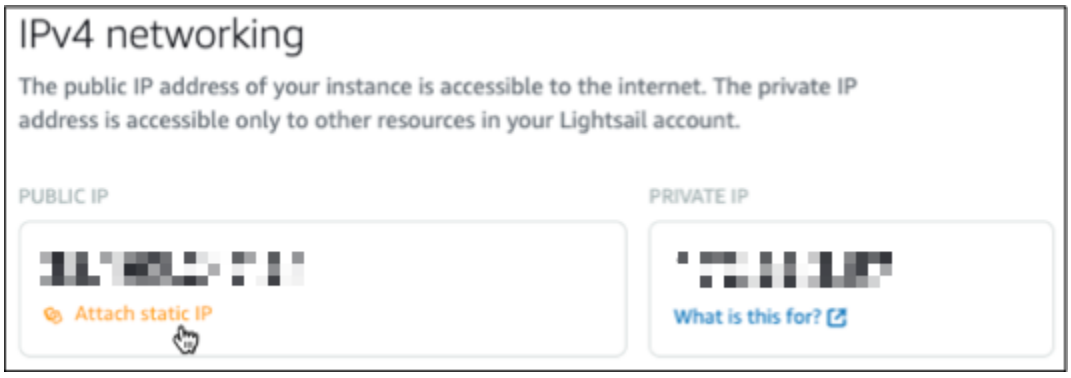
Windows Server 2016 インスタンスのデフォルトのパブリック IP は、インスタンスを停止して開始すると変わります。インスタンスにアタッチした静的 IP アドレスは、インスタンスを停止して開始しても変わりません。

静的 IP アドレスを作成して Windows Server 2016 インスタンスにアタッチします。詳細については、Lightsail ドキュメントの「静的 IP を作成してインスタンスにアタッチする」を参照してください。

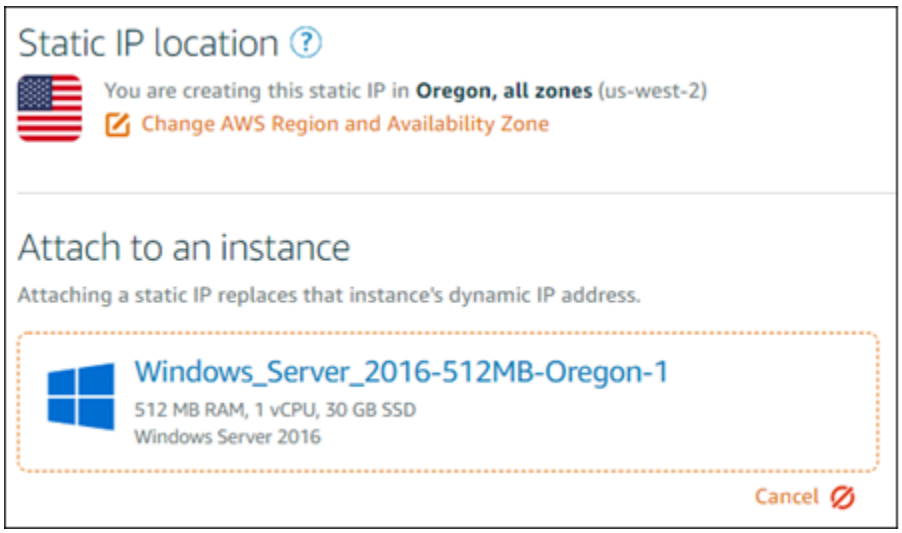
1. Lightsail ホームページのインスタンスタブで、実行中の Windows Server 2016 インスタンスを選択します。



2. [ネットワーキング] タブ、[静的 IP の作成] の順に選択します。



3. このチュートリアルで前に選択したインスタンスに基づいて、静的 IP の場所とアタッチ済みインスタンスが事前に選択されます。



4. 静的 IP の名前を入力します。

   リソース名:

   - AWS リージョン Lightsail アカウントの各 内で一意である必要があります。
   - 2〜255 文字を使用する必要があります。
   - 先頭と末尾は英数字または数字を使用する必要があります。
   - 英数字、数字、ピリオド、ダッシュ、アンダースコアを使用することができます。

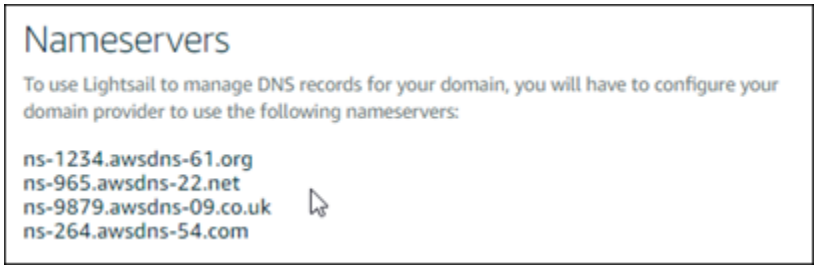
5. [Create(作成)] を選択します。

ステップ 5: DNS ゾーンを作成し、ドメインを Windows Server 2016 インスタンスにマッピングする

ドメインの DNS レコードの管理を Lightsail に転送します。これにより、ドメインを Windows Server 2016 インスタンスにマッピングし、Lightsail コンソールを使用してウェブサイトのすべてのリソースを簡単に管理できます。詳細については、Lightsail ドキュメントの「DNS ゾーンを作成してドメインの DNS レコードを管理する」を参照してください。

1. Lightsail ホームページのドメインと DNS タブで、DNS ゾーンの作成を選択します。
2. ドメインを入力し、[DNS ゾーンの作成] を選択します。
3. ページに表示されたネームサーバーのアドレスを書き留めておきます。

   これらのネームサーバーアドレスをドメイン名のレジストラに追加して、ドメインの DNS レコードの管理を Lightsail に移管します。

4. ドメインの DNS レコードの管理が Lightsail に転送されたら、次のように A レコードを追加して、ドメインの頂点を LAMP インスタンスにポイントします。

   a. DNS ゾーンの [Assignments] (割り当て) タブで [Add assignment] (割り当てを追加) を選択します。

   b. [Select a domain] (ドメインの選択) フィールドで、ドメインまたはサブドメインを選択します。

   c. [Select a resource] (リソースの選択) ドロップダウンで、このチュートリアルで以前に作成した LAMP インスタンスを選択します。

   d. [Assign] (割り当て) を選択します。

   変更内容がインターネットの DNS を通じて伝播されるまで待ってから、LAMP インスタンスへのトラフィックのルーティングを開始します。

次のステップ

Amazon Lightsail で Windows Server 2016 インスタンスを起動した後に実行できる追加手順をいくつか示します。

- Windows Server インスタンスのスナップショットの作成
- Windows Server ベースの Lightsail インスタンスを保護するためのベストプラクティス
- ブロックストレージディスクを作成して Windows Server インスタンスにアタッチする
- Windows Server インスタンスのストレージ領域を拡張する

Amazon Lightsail の詳細情報

次のリストには、Lightsail ユーザーガイドで公開されていない Amazon Lightsail の追加情報へのリンクが含まれています。

目次

ブログ

- Datadog を使用した Amazon Lightsail インスタンスの正常性のモニタリング

  2022 年 3 月 30 日 Datadog を使用した Lightsail ワークロードのモニタリングが、アプリケーションのパフォーマンスを確保し、コストを管理するのにどのように役立つのかを詳しく見ていきます。

- Amazon Lightsail を使用して AWS での研究のために Galaxy を設定する方法

  2022 年 1 月 13 日 – 科学ワークフロー、データ統合、およびデジタル保存プラットフォームである Galaxy を Lightsail でデプロイします。

- What happens when you type a URL into your browser (ブラウザに URL を入力したときの挙動)

  2021 年 8 月 26 日 – ブラウザに URL を入力して Enter キーを押すとどうなりますか?

- Amazon Lightsail インスタンスでのメモリの使用状況のモニタリング

2021 年 6 月 14 日 – モニタリング、アラーム、および通知のためにメモリの使用状況を Amazon CloudWatch に送信する Lightsail インスタンスを設定します。

- Amazon Lightsail を使用したコンテナ化済み ASP.NET ウェブアプリケーションのフリクションレスホスティング

  2021 年 6 月 10 日 – PostgreSQL データベースに接続するコンテナ化された ASP.NET ウェブアプリケーションを取得し、Lightsail にデプロイする方法。

- Amazon Lightsail コンテナを使用した WordPress ウェブサイトの立ち上げ

  2021 年 4 月 5 日 – Lightsail コンテナと Lightsail データベースを使用して WordPress ウェブサイトを立ち上げます。

- Lightsail コンテナ: クラウドでコンテナを実行する簡単な方法

  2020 年 11 月 13 日 – Lightsail でコンテナベースのワークロードをデプロイします。

- Amazon Lightsail から Amazon EC2 へのウェブサービスの移行

  2020 年 10 月 16 日 – Amazon EC2 で本番稼働環境を設定し、ウェブサービスを Lightsail からその環境に移行します。

- Amazon Lightsail インスタンス上で実行する Graylog サーバーの構築

  2020 年 7 月 28 日 – Lightsail で Graylog サーバーを構築する方法。

- Lightsail コンテンツ配信ネットワークを使用したウェブサイトのパフォーマンスの向上

  2020 年 7 月 23 日 – WordPress に加えて標準のウェブサーバーの両方で動作するように Lightsail ディストリビューションを設定します。

- Amazon Lightsail インスタンスでシステムパフォーマンスをプロアクティブにモニタリングする

  2020 年 6 月 4 日 – ユーザーに影響を与える前にシステムパフォーマンスの問題を防ぐことができるように、バースト可能な容量アラートを設定します。

- 新しい Lightsail ファイアウォール機能を使用したサイトセキュリティの強化

  2020 年 5 月 7 日 – SSH を使用したリモートアクセスを単一の送信元 IP アドレスに制限します。

- CodeDeploy と CodePipeline を使用してアプリケーションを Amazon Lightsail にデプロイする

  2020 年 4 月 23 日 – GitHub に変更をプッシュするたびに、CodeDeploy および CodePipeline と連携してアプリケーションを自動的にデプロイ (または更新) するように Lightsail を設定します。

- Amazon Lightsail でのロードバランサーの使用

2020 年 4 月 21 日 – Amazon Lightsail ロードバランサーを使用して単純な Node.js ウェブアプリケーションのロードバランスを実行する方法。

- Ghost を使用して Amazon Lightsail で写真日記を作成する

2020 年 3 月 23 日 – Lightsail で Ghost を使用して写真日記を開始します。

- Amazon Lightsail データベースのヒントとコツ

2020 年 3 月 23 日 — Amazon Relational Database Service (Amazon RDS) に搭載されている高度な機能を使用します。

- モニタリングと通知の設定と使用

2020 年 2 月 27 日 – 通知先の作成、新しいアラームの作成、およびリソースモニタリングを使用した通知のテスト。

- Amazon Lightsail での高可用性 WordPress サイトのデプロイ、パート 1: WordPress を使用した高可用性 Lightsail データベースの実装

2019 年 10 月 22 日 – Lightsail で可用性の高い WordPress サイトを構築します、(パート 1)。

- Amazon Lightsail で高可用性の WordPress サイトのデプロイ、パート 2: Amazon S3 と WordPress を併用したメディアファイルの安全な配信

2019 年 10 月 31 日 – Lightsail で可用性の高い WordPress サイトを構築します、(パート 2)。

- Amazon Lightsail での高可用性の WordPress サイトのデプロイ、パート 3: Amazon CloudFront を使用したセキュリティとパフォーマンスの向上

2019 年 11 月 7 日 – Lightsail で可用性の高い WordPress サイトを構築します (パート 3)。

- Amazon Lightsail での高可用性の WordPress サイトのデプロイ、パート 4: Lightsail ロードバランサーを使用したパフォーマンスとスケーラビリティの向上

2019 年 11 月 14 日 – Lightsail で可用性の高い WordPress サイトを構築します (パート 4)。

- Amazon Lightsail によるポケットの Platform as a Service の構築

2019 年 10 月 8 日 – Lightsail でポケットプラットフォームをアセンブルします。

- Amazon Lightsail を使用して Nginx ベースの HTTP/HTTPS ロードバランサーをデプロイする

2019 年 7 月 8 日 – Lightsail インスタンス内に NGINX ベースのロードバランサーを設定します。

- AWS クラウド は初めてですか? Amazon Lightsail がお手伝いできます

2019 年 3 月 27 日 – Amazon Lightsail の開始方法。

- 新規 – Amazon Lightsail 用マネージドデータベース

  2018 年 10 月 16 日 – 数回クリックするだけでマネージドデータベースを作成できます。

- Amazon Lightsail の更新: インスタンスサイズの増加と値下げ

  2018 年 8 月 23 日 – Lightsail インスタンスの概要。

- Amazon Lightsail: AWS のパワー、VPS のシンプルさ

  2016 年 11 月 30 日 – Lightsail のリリースの発表。

チュートリアル

上位 5 位の実践チュートリアル:

1. 負荷分散された WordPress ウェブサイトを作成する

   2021 年 9 月 8 日 – Lightsail で高可用性の WordPress ウェブサイトをリリースします。

2. Amazon Lightsail を使用して WordPress ウェブサイトを移行して管理する

   2021 年 2 月 22 日 – Seahorse ソフトウェアを使用して Lightsail に WordPress ウェブサイトのクローンを起動します。

3. Linux 仮想マシンを起動する

   2020 年 9 月 11 日 – Lightsail を使用して Linux インスタンスを起動、設定、および接続します。

4. Windows 仮想マシンを起動する

   2020 年 9 月 11 日 – Lightsail を使用して Windows インスタンスを起動、設定、および接続します。

5. Amazon Lightsail で cPanel と WHM インスタンスを起動する

   2020 年 7 月 27 日 – このチュートリアルでは、Lightsail で cPanel と WHM インスタンスを起動して実行した後に実行できるいくつかのステップについて説明します。

- Amazon Lightsail で Magento をセットアップして設定する方法

  2021 年 8 月 11 日 – e コマースサイトを立ち上げて稼働させます。

- How to connect your WordPress site to an object storage bucket (WordPress サイトをオブジェクトストレージバケットに接続する方法)

  2021 年 7 月 14 日 – Lightsail で WordPress サイトを設定し、ウェブサイトを Lightsail バケットに接続します。
- Create object storage buckets (オブジェクトストレージバケットを作成する)

  2021 年 7 月 14 日 – Amazon Lightsail でオブジェクトストレージバケットを作成します。
- WordPress ウェブサイトを Amazon Lightsail バケットに接続し、配信する

  2021 年 7 月 14 日 – Lightsail コンテンツ配信ネットワーク (CDN) ディストリビューションのオリジンとして Lightsail バケットを設定します。
- How to setup and configure Plesk (Plesk のセットアップおよび設定方法)

  2021 年 4 月 22 日 – Lightsail で Plesk ホスティングスタックを起動して実行します。
- Prestashop e コマースサイトの設定方法

  2021 年 4 月 1 日 – Bitnami ブループリントによって認定された PrestaShop を使用して Lightsail インスタンスを起動および設定します。
- Amazon Lightsail で Amazon EFS を使用する方法

  2021 年 3 月 15 日 – VPC ピアリングを使用して、Lightsail インスタンスから Amazon EFS ファイルシステムを作成して接続します。
- How to setup a Nginx reverse proxy (Nginx リバースプロキシを設定する方法)

  2021 年 2 月 10 日 – Lightsail コンテナを使用して Nginx リバースプロキシを設定します。
- How to Serve a Flask pp (Flask pp を提供する方法)

  2021 年 2 月 3 日 – Flask アプリケーションを Lightsail コンテナで提供する方法について説明します。
- Amazon Lightsail でのコンテナイメージの作成、プッシュ、デプロイ

  2020 年 11 月 11 日 – Dockerfile を使用して、ローカルマシンにコンテナイメージを作成します。
- Build a Drupal website (Drupal のウェブサイトを構築する)

  2020 年 9 月 11 日 – Lightsail で本番稼働環境に対応した Drupal ウェブサイトをデプロイしてホストします。
- Build a LAMP stack web App (LAMP スタックウェブアプリケーションを構築する)

2020 年 9 月 9 日 – Lightsail で高可用性 PHP ウェブアプリケーションを起動して実行します。

- ディストリビューションと動作するように WordPress インスタンスを設定する

  2020 年 7 月 16 日 – Lightsail ディストリビューションで動作するように WordPress インスタンスを設定します。

- Launch a WordPress website (WordPress ウェブサイトを起動する)

  2020 年 3 月 23 日 – Lightsail 仮想マシンに WordPress をインストールしてウェブサイトを立ち上げて実行します。

- Host a .NET application (.NET アプリケーションをホストする)

  2020 年 3 月 20 日 – Lightsail を使用して .NET アプリケーションを構築およびデプロイします。

- Amazon Route 53 のドメインを Lightsail リソースにマッピングする

  example.com などのドメインのトラフィックを Lightsail リソースにルーティングします。

動画

- Amazon Lightsail チュートリアル: Django アプリケーションをデプロイする

  2021 年 7 月 14 日 – このチュートリアルでは、Django アプリケーションを作成します。

- Amazon Lightsail チュートリアル: Flask アプリケーションをデプロイする

  2021 年 7 月 14 日 – このチュートリアルでは、Flask アプリケーションを作成します。

- Amazon Lightsail チュートリアル: NGINX リバースプロキシをデプロイする

  2021 年 7 月 14 日 – Flask アプリケーションを作成し、Docker コンテナを構築し、Lightsail にコンテナサービスを作成してから、アプリケーションをデプロイします。

- Amazon Lightsail チュートリアル: e コマースサイトをデプロイする

  2021 年 7 月 14 日 – Bitnami ブループリントによって認定された PrestaShop を使用して Lightsail インスタンスを起動して、設定します。

- Amazon Lightsail でコンテナ化されたアプリケーションをデプロイする

  2020 年 12 月 29 日 – Lightsail でコンテナ化されたアプリケーションをデプロイする方法について説明します。

- Amazon Lightsail チュートリアル: Drupal ウェブサイトを構築する

2020 年 8 月 31 日 – Drupal インスタンスを起動して設定します。

- Amazon Lightsail チュートリアル: LAMP スタックアプリケーションをデプロイする

  2020 年 8 月 31 日 – LAMP (Linux Apache MySQL PHP) スタックアプリケーションを単一の Lightsail インスタンスにデプロイします。

- Amazon Lightsail チュートリアル: Linux インスタンスを起動する

  2020 年 8 月 31 日 – Linux インスタンスを起動する方法について説明します。

- Amazon Lightsail チュートリアル: Windows インスタンスを起動する

  2020 年 8 月 31 日 – Windows インスタンスを起動する方法について説明します。

- Amazon Lightsail チュートリアル: 独自の Minecraft サーバーを実行する

  2020 年 8 月 31 日 – 専用の Minecraft サーバーを設定する方法について説明します。

- Amazon Lightsail チュートリアルのご紹介

  2020 年 8 月 31 日 – Lightsail を使用してクラウドジャーニーを始めましょう。

- Amazon Lightsail: AWS を始める最も簡単な方法

  2020 年 3 月 20 日 – Lightsail はAWS を始める最も簡単な方法です。仮想サーバー、ストレージ、データベース、ネットワークに加えて、コスト効率の良い月額プランをご利用いただけます。

- Amazon Lightsail での Plesk インスタンスの設定

  2019 年 3 月 27 日 – Lightsail で Plesk インスタンスを設定する方法について説明します。

- Amazon Lightsail で WordPress Multisite を設定する

  2019 年 1 月 15 日 – Lightsail で WordPress Multisite インスタンスを設定する方法について説明します。

- 管理 Lightsail

  2018 年 10 月 9 日 – Lightsail の主な機能を簡単に確認します。

- Amazon Lightsail で MEAN スタックアプリケーションをデプロイする

  2018 年 6 月 5 日 – Lightsail の MEAN ブループリントを使用して、カスタムアプリケーションをクラウドにデプロイします。

- Amazon Lightsail で WordPress インスタンスをデプロイする

2018 年 6 月 5 日 – Lightsail で WordPress インスタンスをデプロイします。

チュートリアル: MySQL 5.6 データベースから最新のデータベースバージョンにデータを移行する

このチュートリアルでは、MySQL 5.6 データベースから Amazon Lightsail の新しい MySQL 5.7 データベースにデータを移行する方法について解説しています。移行を実行するには、MySQL 5.6 データベースに接続し、既存のデータをエクスポートします。次に、MySQL 5.7 データベースに接続し、データをインポートします。新しいデータベースに必要なデータを取得したら、アプリケーションを再設定して新しいデータベースに接続できるようにします。

目次

ステップ 1: 変更を確認する

MySQL 5.6 データベースから MySQL 5.7 データベースへの移行は、メジャーバージョンへのアップグレードと見なされます。メジャーバージョンのアップグレードには、既存のアプリケーションとの下位互換性のないデータベースの変更が含まれる場合があります。本稼働インスタンスへの適用前に、いずれのアップグレードも徹底的にテストすることをお勧めします。詳細については、MySQL ドキュメントのMySQL 5.7 での変更を参照してください。

まず、既存の MySQL 5.6 データベースから新しい MySQL 5.7 データベースにデータを移行することをお勧めします。次に、本番前のインスタンスで新しい MySQL 5.7 データベースを使用してアプリケーションをテストします。アプリケーションが期待どおりに動作する場合は、本番環境のインスタンスのアプリケーションに変更を適用します。さらなる措置を取る場合は、既存の MySQL 5.7 データベースから新しい MySQL 8.0 データベースにデータを移行し、本番前のアプリケーションで再度テストし、本番環境のアプリケーションに変更を適用します。

ステップ 2: 前提条件を完了させる

このチュートリアルの次のセクションに進むには、次の必要条件を満たす必要があります。

- ローカルコンピュータに MySQL Workbench をインストールします。このコンピュータを使用して、データベースに接続してデータをエクスポートおよびインポートします。詳細については、MySQL ウェブサイトのMySQL Workbench のダウンロードを参照してください。
- Lightsail で MySQL 5.7 データベースを作成します。詳細については、「Amazon Lightsail でデータベースを作成する」を参照してください。
- データベースのパブリックモードを有効にします。これにより、MySQL Workbench を使用してデータベースに接続することができるようになります。データのエクスポートとインポートが完了したら、データベースのパブリックモードを無効にすることができます。詳細については、「データベースのパブリックモードの設定」を参照してください。
- MySQL Workbench を設定してデータベースに接続する。詳細については、「MySQL データベースに接続する」を参照してください。

ステップ 3: MySQL 5.6 データベースに接続してデータをエクスポートする

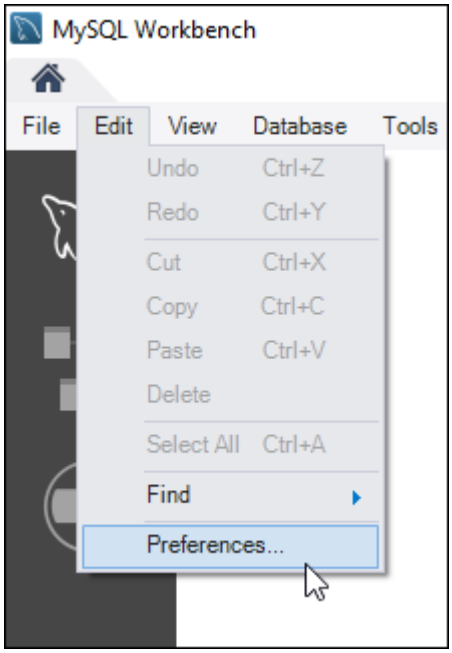
チュートリアルのこのセクションでは、MySQL 5.6 データベースに接続し、MySQL Workbench を使用してそのデータベースからデータをエクスポートします。MySQL Workbench を使用してデータをエクスポートする方法の詳細については、MySQL Workbench マニュアルの「SQL Data のエクスポートとインポートウィザード」を参照してください。

1. MySQL Workbench を使用して MySQL 5.6 データベースに接続する。

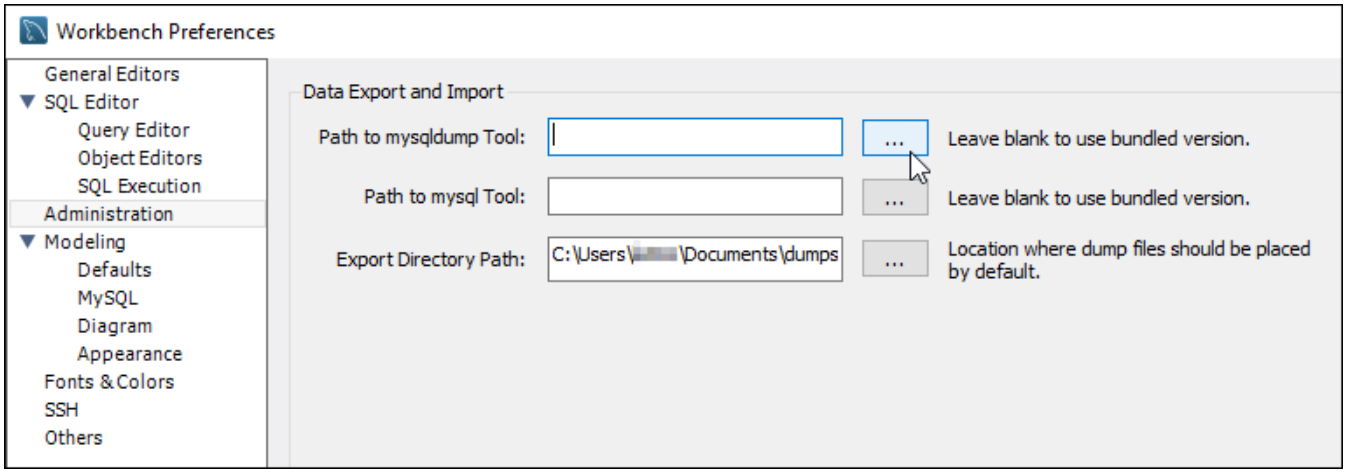
   MySQL ワークベンチでは、mysqldump を使用してデータをエクスポートします。MySQL Workbench で使用される mysqldump のバージョンは、データをエクスポートする MySQL データベースのバージョンと同じ (またはそれ以降) である必要があります。たとえば、MySQL 5.6.51 データベースからデータをエクスポートする場合は、バージョン 5.6.51 かそれ以降の mysqldump を使用する必要があります。正しいバージョンの mysqldump を使用しているか確認するために、ローカルコンピュータに適切なバージョンの MySQL サーバーをダウンロードしてインストールする必要がある場合があります。MySQL サーバーの特定のバージョンをダウンロードするには、MySQL ウェブサイトのMySQL コミュニティダウンロードを参照してください。Windows MSI 用の MySQL インストーラでは、ダウンロードする MySQL サーバーのバージョンを選択できます。

MySQL ワークベンチで使用する mysqldump の正しいバージョンを選択するには、以下の手順を実行します。

1. MySQL ワークベンチで [Edit] (編集)、[Preferences] (設定) の順に選択します。

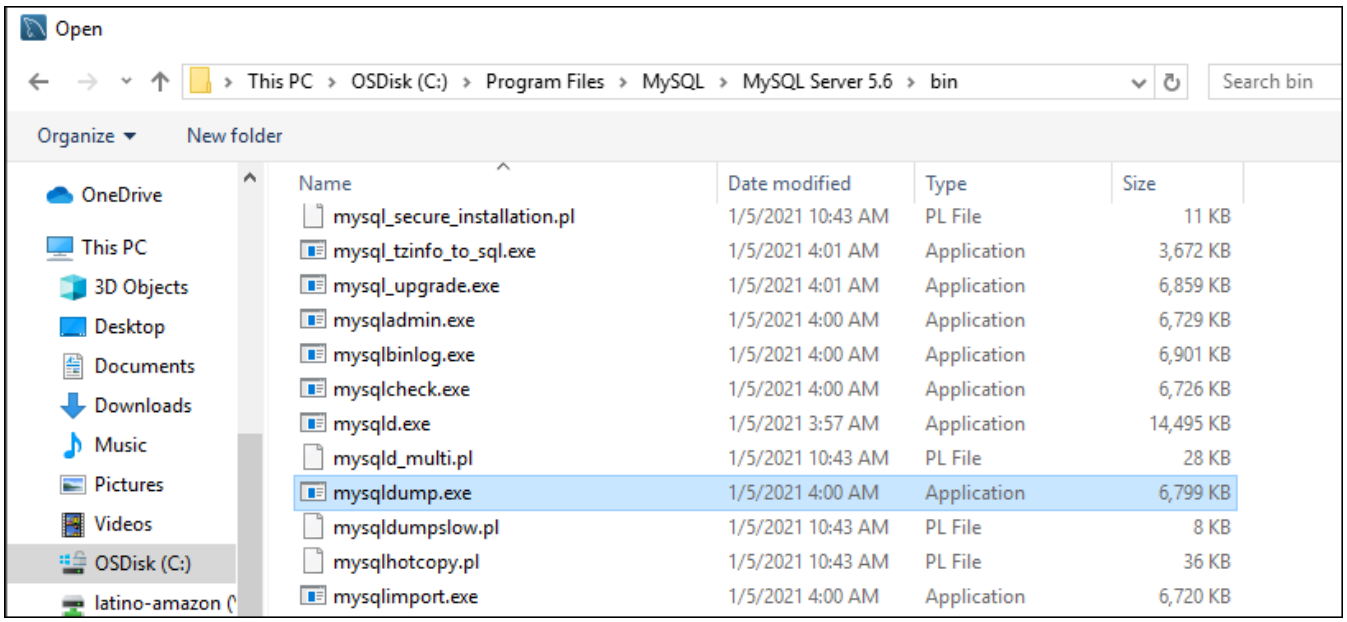


2. ナビゲーションペインで [Administration] (管理) を選択します。
3. Workbench Preferences ウィンドウが表示されたら、Path to mysqldump Tool のテキストボックスの横にある省略記号ボタンを選択します。

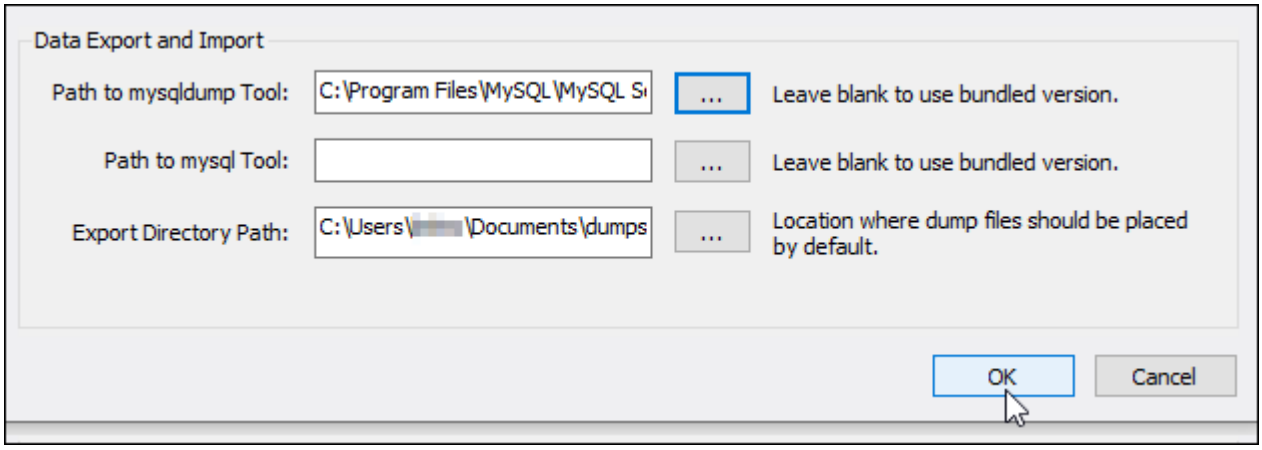


4. 適切なmysqldump 実行可能ファイルの場所まで移動したら、ダブルクリックします。

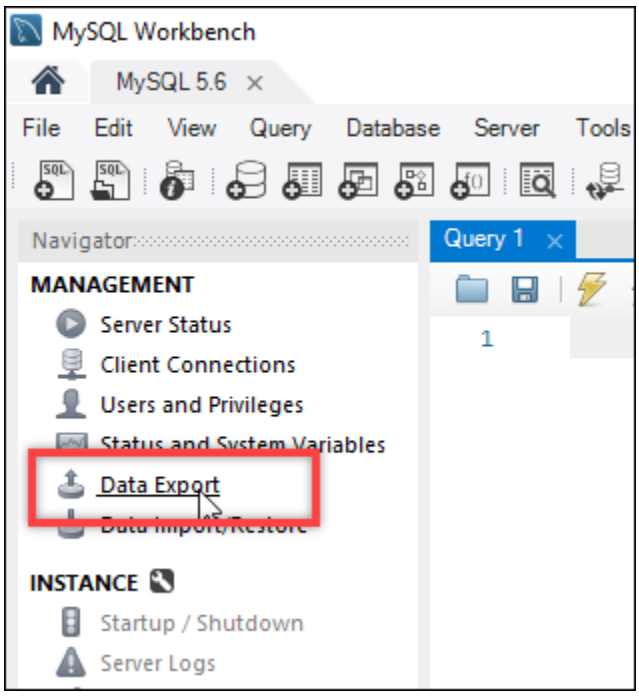
   Windows の場合、mysqldump.exe ファイルは通常 C:\Program Files\MySQL\MySQL Server 5.6\bin ディレクトリにあります。Linux の場合、ターミナルに which mysqldump を入力して mysqldump ファイルの位置を確認します。



5. Workbench Preferences ウィンドウで [OK] を選択します。

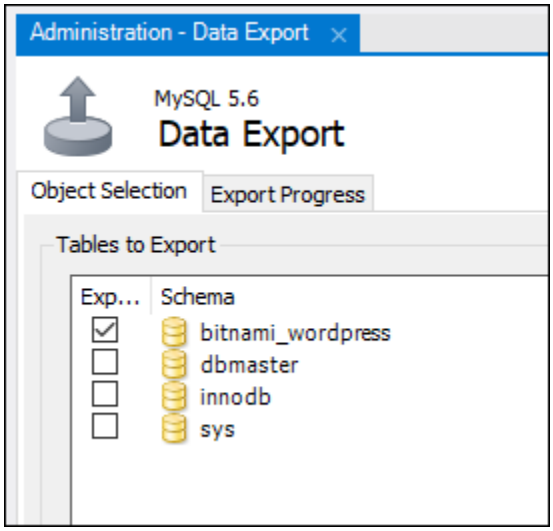


2. [Navigator] (ナビゲーター) ペインで [Data Export] (データエクスポート) を選択します。

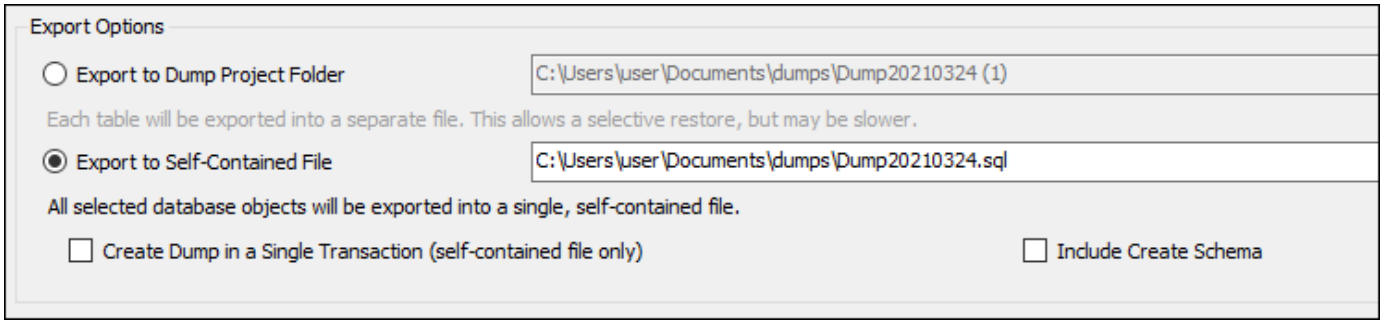


3. [データのエクスポート] タブが表示されたら、エクスポートするテーブルの横にチェックマークを追加します。

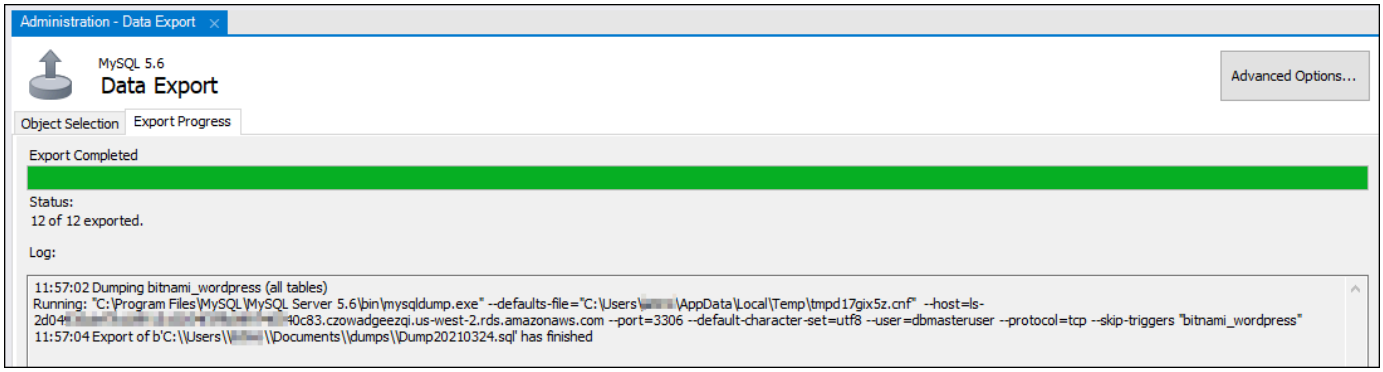
> **Note**
>
> この例で選択した `bitnami_wordpress` テーブルには、「Certified by Bitnami」WordPress インスタンスにある WordPress ウェブサイトのデータが含まれています。



4. [Export Options] (エクスポートオプション) セクションで、[Export to Self-Contained File] (自己完結型ファイルにエクスポート) を選択してエクスポートファイルが保存されるディレクトリを書き留めておきます。



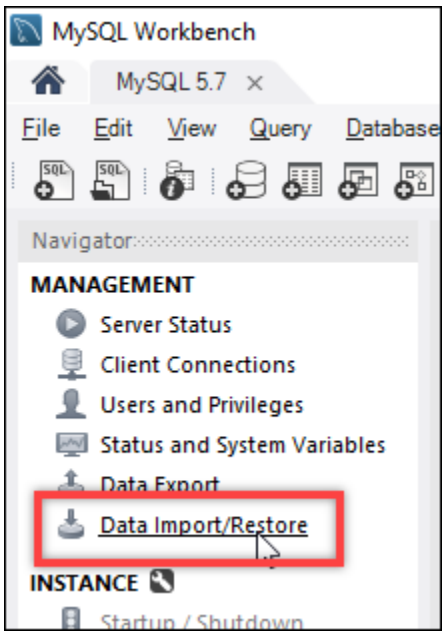
5. [Start Export] (エクスポートの開始) を選択します。
6. このチュートリアルの次のセクションに進む前に、エクスポートが完了するのを待ちます。



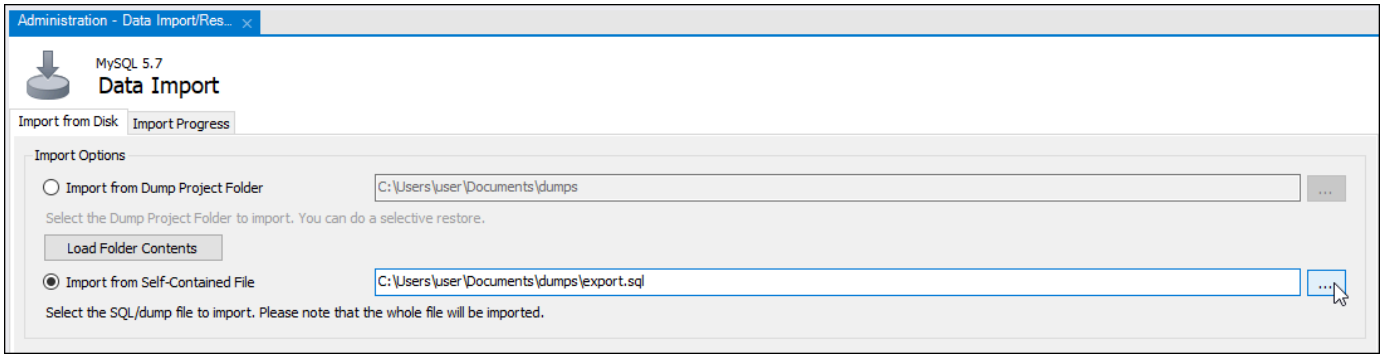
ステップ 4: MySQL 5.7 データベースに接続してデータをインポートする

チュートリアルのこのセクションでは、MySQL 5.7 データベースに接続し、MySQL Workbench を使用してそのデータベースにデータをエクスポートします。

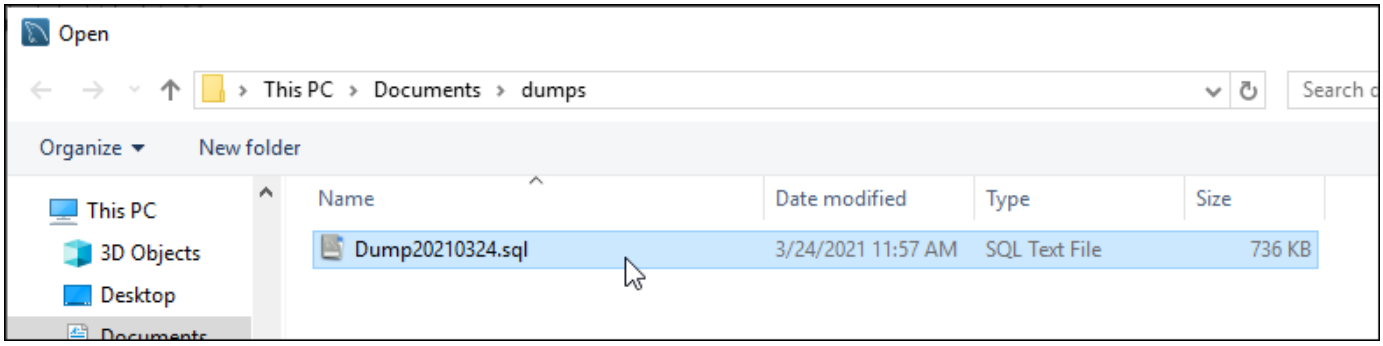
1. ローカルコンピュータの MySQL Workbench を使用して MySQL 5.7 データベースに接続する。
2. [Navigator] (ナビゲーター) ペインの [Data Import/Restore] (データのインポート/復元) を選択します。



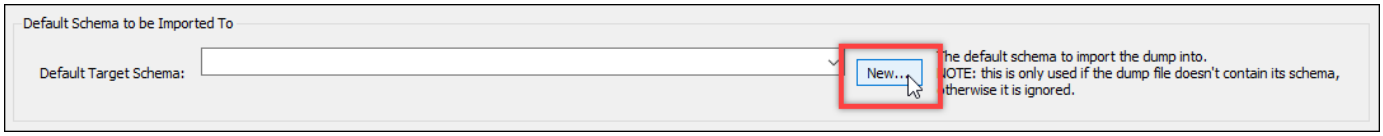
3. [Data Import] (データのインポート) タブが表示されたら、[Export to Self-Contained File] (自己完結型ファイルにエクスポート) を選択して、テキストボックスの横にある省略記号ボタンを選択します。



4. エクスポートファイルが保存された場所まで移動したら、ダブルクリックします。

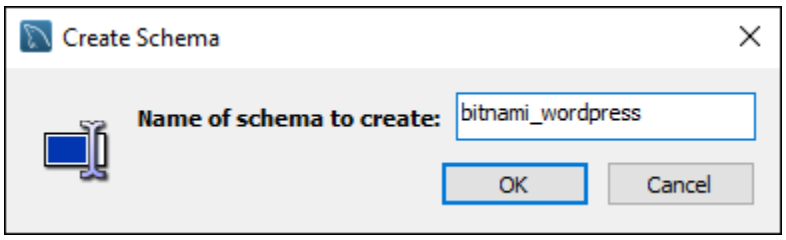


5. [Default Schema to be imported To] (インポート先のデフォルトスキーマ) のセクションで、[New] (新規) を選択します。

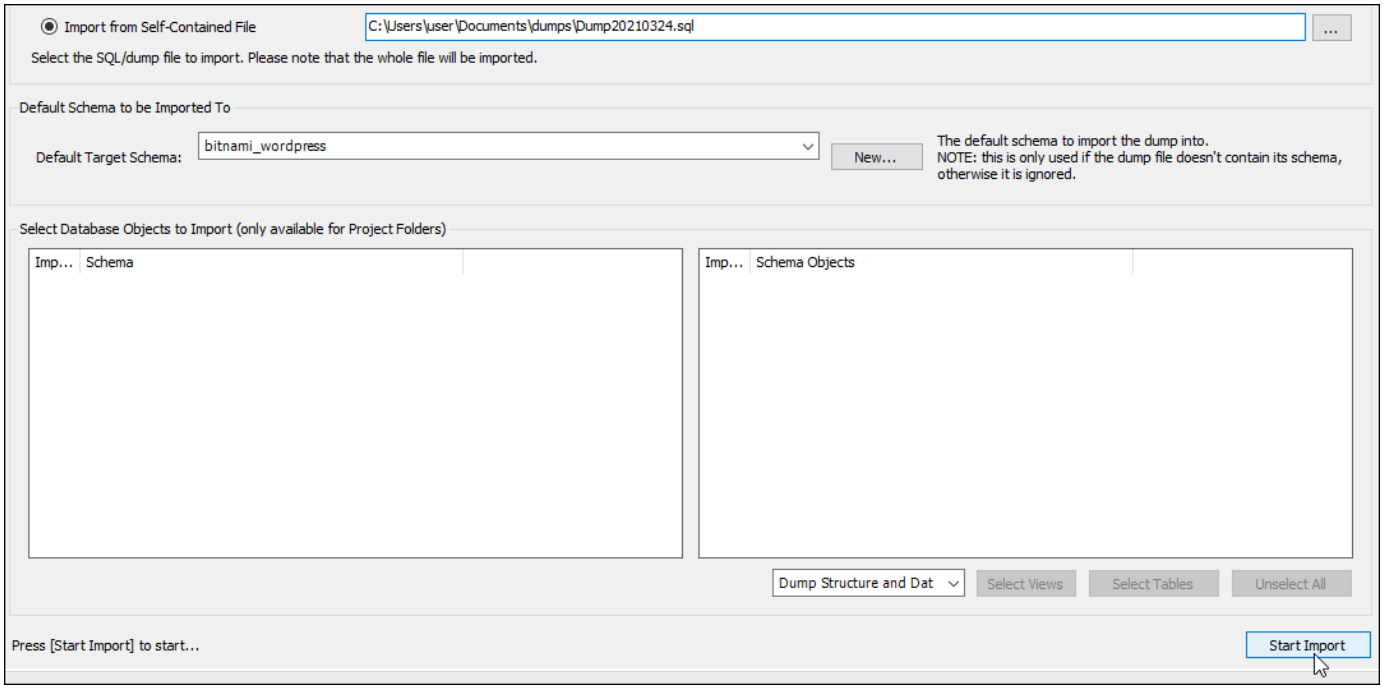


6. [Create Schema] (スキーマの作成) ウィンドウが表示されたら、スキーマの名前を入力します。

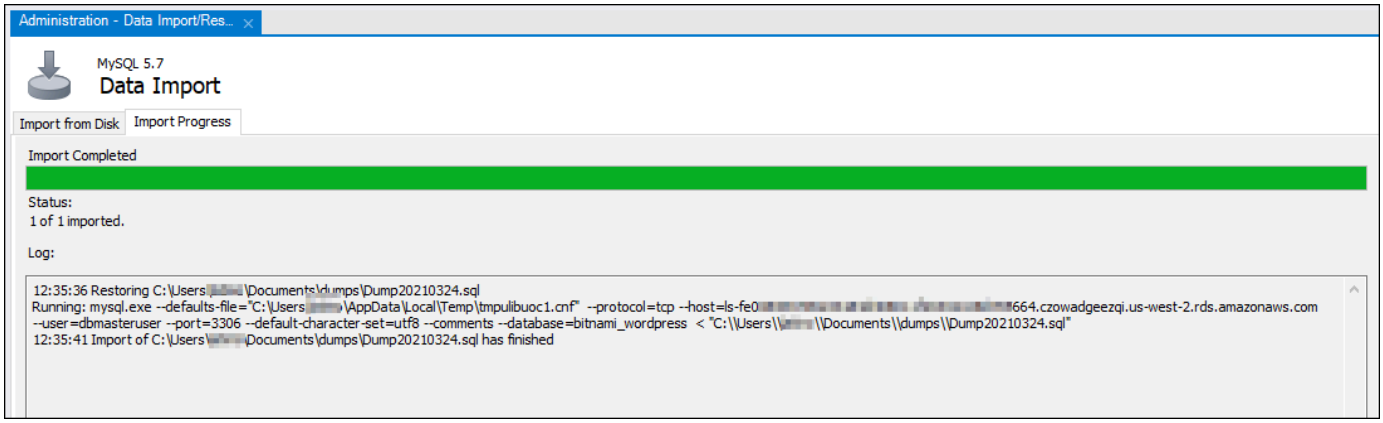
> **Note**
>
> この例では、エクスポートされたデータベーステーブルの名前が `bitnami_wordpress` であるため、それを入力しています。



7. [Start import] (インポートの開始) を選択します。



8. このチュートリアルの次のセクションに進む前に、インポートが完了するのを待ちます。



ステップ 5: アプリケーションをテストして移行を完了する

この時点で、データは新しい MySQL 5.7 データベースに格納されます。本番前の環境でアプリケーションを設定し、アプリケーションと新しい MySQL 5.7 データベース間で接続テストを行います。アプリケーションが期待どおりに動作する場合は、本番環境でアプリケーションに変更を反映します。

移行が完了したら、データベースのパブリックモードを無効にする必要があります。不要になった MySQL 5.6 データベースは削除できます。ただし、削除する前に MySQL 5.6 データベースのスナップショットを作成することをお勧めします。またこの作業をする際、新しい MySQL 5.7 データベースのスナップショットも作成しておくことをお勧めします。詳細については、「データベースのスナップショットを作成する」を参照してください。

Lightsail で Plesk をセットアップして設定する

Amazon Lightsail では、次の機能を持つ Plesk ホスティングスタックを作成できます。

- グラフィカルユーザーインターフェイスでの自動化機能を備えた WordPress Toolkit
- SSL 証明書での Let's Encrypt のサポートと、単一のインスタンスでの暗号化された (HTTPS) トラフィックの設定
- インスタンス間でファイルを転送する FTP アクセス
- Docker プロキシルール
- ウェブベースのサーバー管理およびセキュリティツール (Plesk Firewall、Logs、ModSecurity)

このガイドでは、Lightsail で Plesk インスタンスを作成する方法と、ユーザー名とパスワードを作成して Plesk パネルに初めてサインインする方法について説明します。

> **Important**
>
> Plesk インスタンスの起動後に問題が発生した場合は、Plesk のサポートページにアクセスして、インスタンスにインストールする必要がある更新があるかどうかを確認します。詳細については、Plesk ヘルプセンターおよびPPA ドキュメントとヘルプポータルのPlesk アップデートを参照してください。

Plesk インスタンスの作成

Lightsail で Plesk インスタンスを作成するには、以下の手順を実行します。

1. https://lightsail.aws.amazon.com/ で Lightsail コンソールにサインインします。
2. Lightsail ホーム画面の [インスタンス] タブで、[インスタンスの作成] を選択します。
3. インスタンスを作成する場所を選択します。

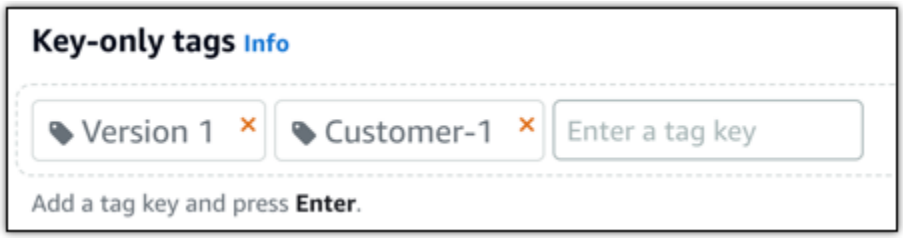
   インスタンスの場所を変更する場合は、[AWS リージョン とアベイラビリティーゾーンの変更] を選択します。
4. [アプリ + OS] で、[Plesk Hosting Stack on Ubuntu (Ubuntu の Plesk ホスティングスタック)] を選択します。
5. インスタンスプランを選択します。

   > **Note**
   >
   > Plesk は、月額 3.50 USD の Lightsail プランではサポートされていません。
6. インスタンスの名前を入力します。

   リソース名:

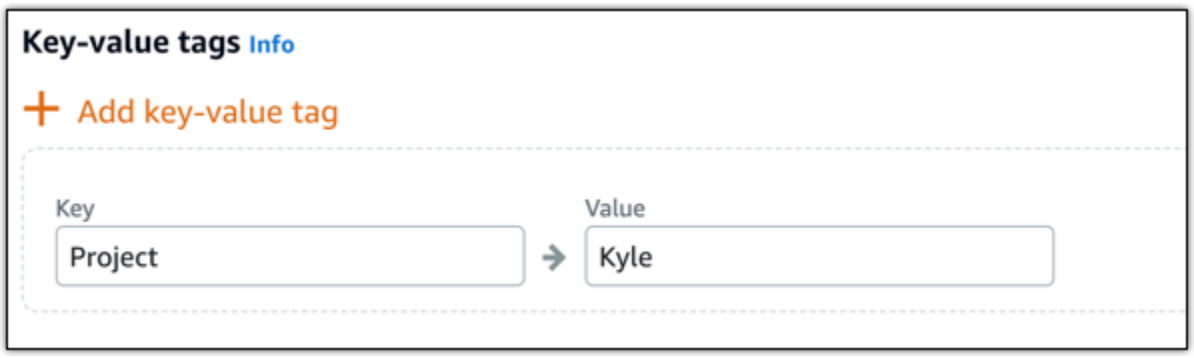
   - Lightsail アカウントの各 AWS リージョン 内で一意である必要があります。
   - 2〜255 文字を使用する必要があります。
   - 先頭と末尾は英数字または数字を使用する必要があります。
   - 英数字、数字、ピリオド、ダッシュ、アンダースコアを使用することができます。
7. 以下のいずれかのオプションを選択して、インスタンスにタグを追加します。

- [Add key-only tags] (キーのみのタグを追加) または [Edit key-only tags] (キーのみのタグを編集) (タグが追加済みの場合)を追加。タグキーのテキストボックスに新しいタグを入力し、Enter キーを押します。タグの入力を完了したら、[保存] を選択してタグを追加し、追加しない場合は、[キャンセル] を選択します。



- [key-value タグの作成] から [キー] テキストボックスにキーを入力し、[値] テキストボックスに値を入力します。タグの入力を完了したら、[保存] を選択し、追加しない場合は、[キャンセル] を選択します。

  キーバリューのタグは、保存する際に一つずつ追加することができます。さらに key-value タグを追加するには、以上のステップを繰り返します。



> **Note**
>
> 「キーのみ」のタグと「キーバリュー」のタグの詳細については、「タグ」を参照してください。

8. [インスタンスの作成] を選択します。

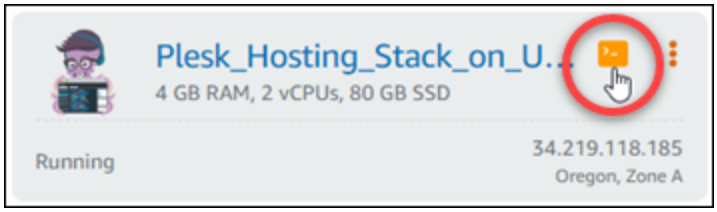
   インスタンスをプロビジョニングし、作成後に使用可能になるまでに数分かかります。

> **Note**
>
> Amazon Lightsail でウェブホスティングに Plesk を使用する場合、静的 IP アドレスをインスタンスにアタッチする必要があります。静的 IP をアタッチした場合、初めてログインする前に Lightsail でインスタンスを再起動する必要があります。

Plesk インスタンスのユーザー名とパスワードの設定

Plesk インスタンスのユーザー名とパスワードを設定し、初めて Plesk パネルにサインインするには、以下の手順を実行します。

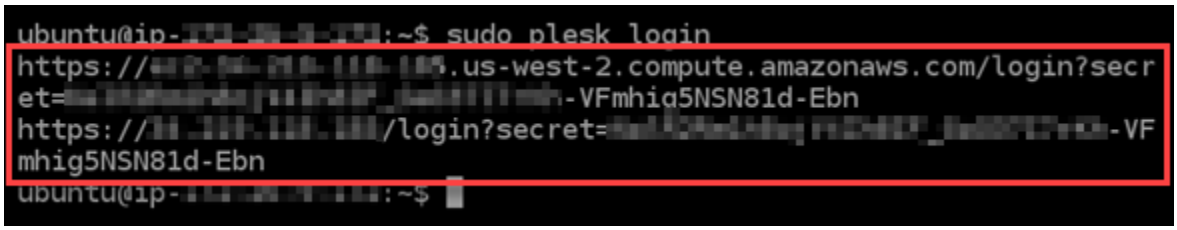
1. Lightsail ホームページの [Instances (インスタンス)] タブで、設定する Plesk インスタンスの SSH クイック接続アイコンを選択します。



2. 次のコマンドを入力します。

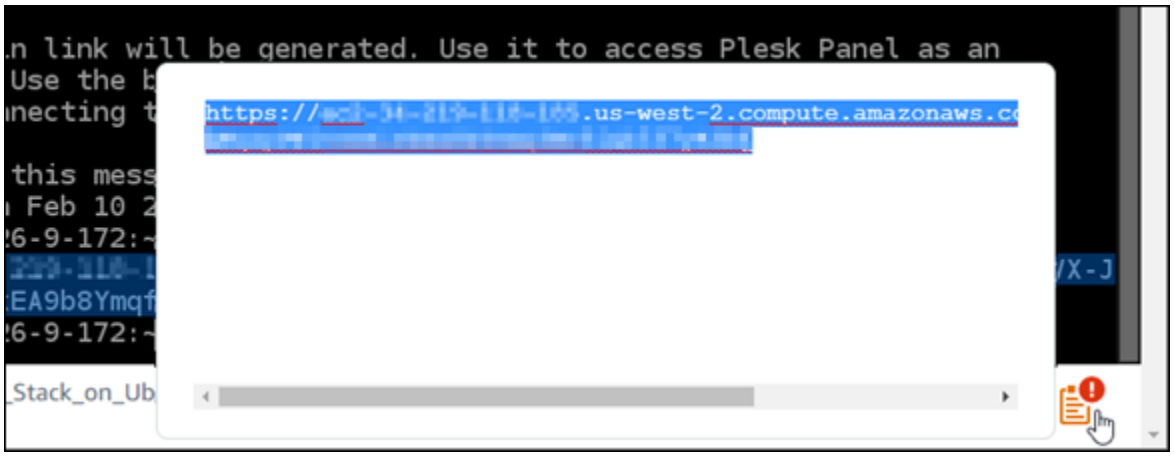
```
sudo plesk login | grep -v internal:8
```

次の例のような結果が表示されます。



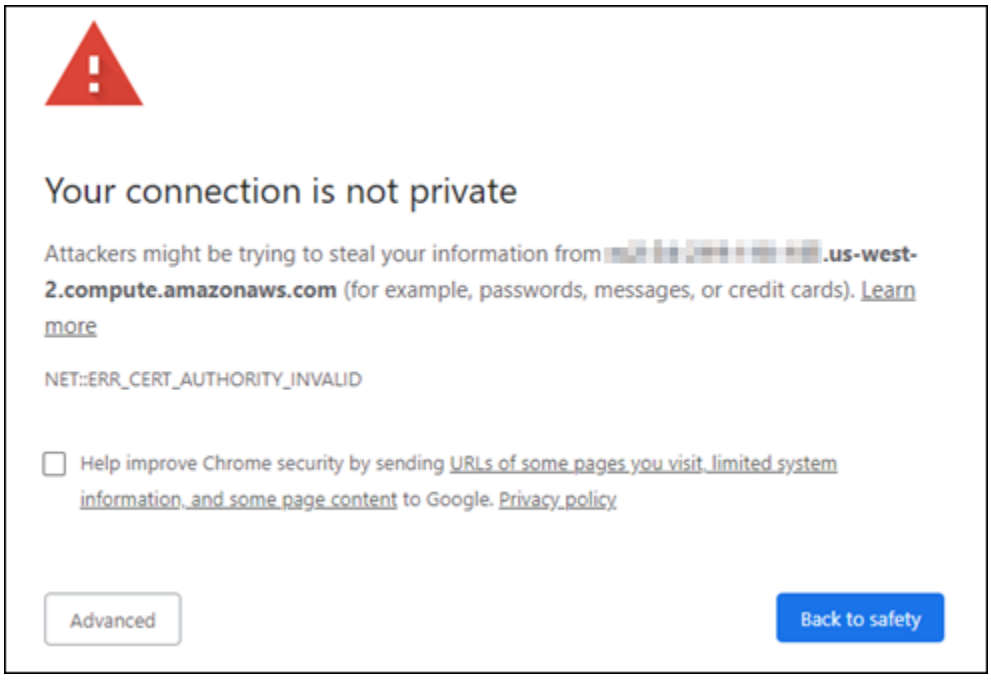
> **Important**
>
> 最近 Plesk インスタンスに静的 IP をアタッチした場合、古いパブリック IP アドレスを使用するワンタイムログイン URL を取得することがあります。インスタンスを再起動し、上のコマンドを再度実行して、新しい静的 IP アドレスを使用するワンタイムログイン URL を取得してください。

3. ブラウザベースの SSH ウィンドウに表示される URL をハイライトし、クリップボードのアイコンを選択して、URL をローカルのクリップボードにコピーします。



4. 新しいブラウザウィンドウを開き、コピーした URL を参照します。

   接続がプライベートではないか、セキュリティで保護されていないか、またはセキュリティ上のリスクがあることを示すブラウザの警告が表示されることがあります。これは、Plesk インスタンスに SSL/TLS 証明書がまだ適用されていない場合に発生します。プロンプトは、使用するブラウザによって次の例に示されているものとは異なる場合があります。



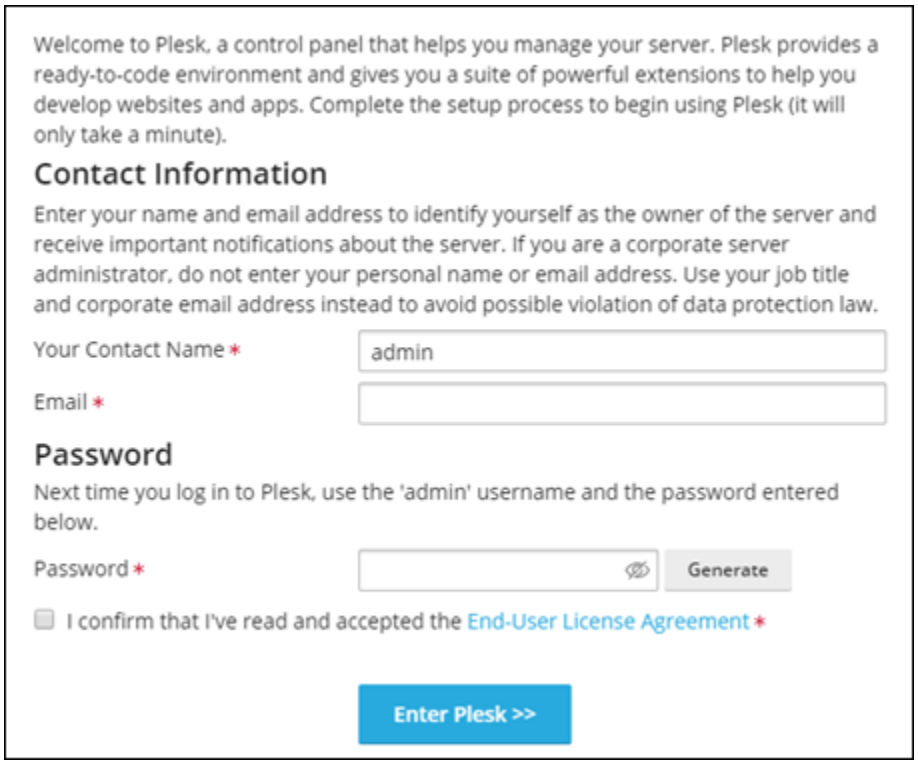
5. 使用するブラウザに応じて、次のいずれかの手順を実行します。

   - Chrome — [Advanced (詳細設定)] を選択し、[Proceed (続行)] を選択して Plesk の設定ページに進みます。

- Edge — [Details (詳細)] を選択し、[Go on to the webpage (Not recommended) (ウェブページへ移動 (非推奨))] を選択して Plesk の設定ページに進みます。
- Firefox — [Advanced (詳細情報)] を選択し、 [Accept the Risk and Continue (危険性を承知で続行)] を選択して Plesk の設定ページに進みます。
- Internet Explorer — [More information (詳細情報)] を選択し、[Go on to the webpage (Not recommended) (ウェブページへ移動 (非推奨))] を選択して Plesk の設定ページに進みます。

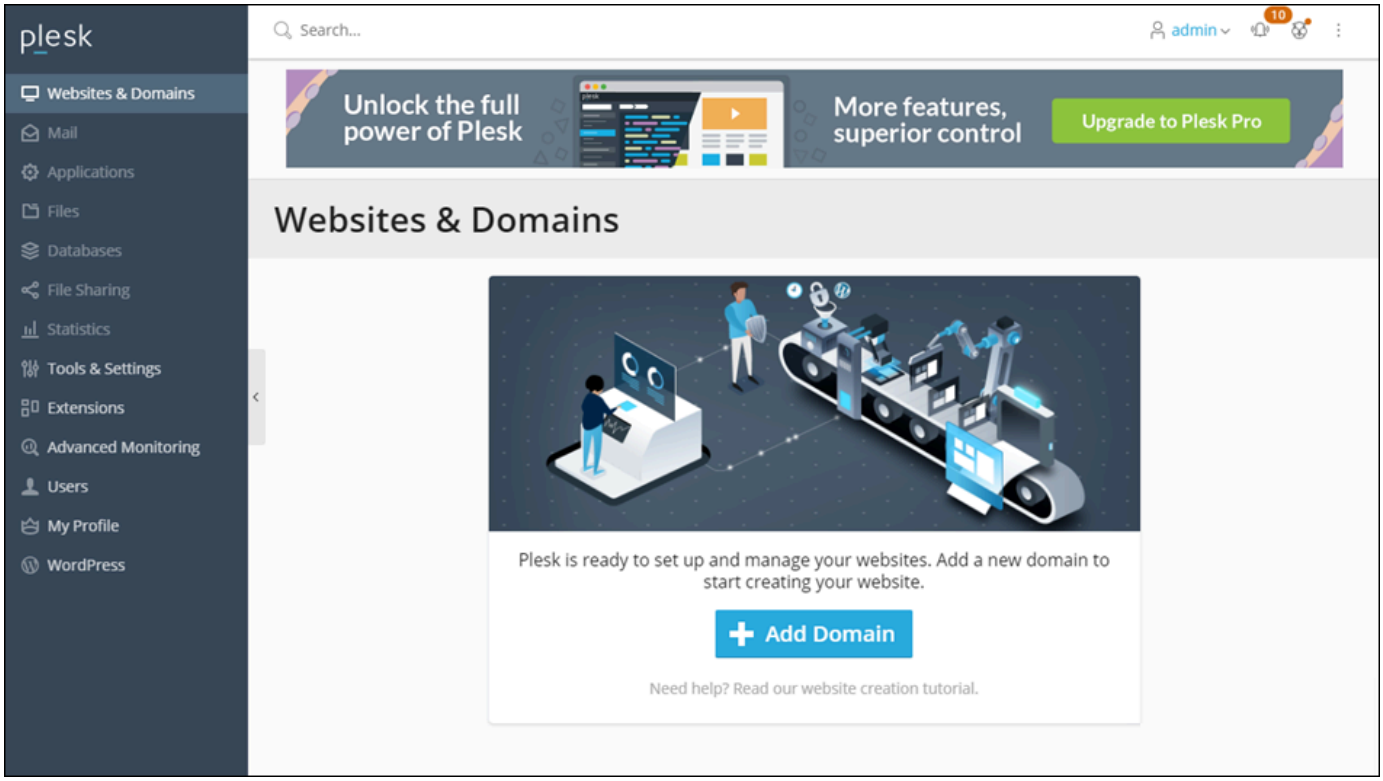
6. 担当者名、メールアドレス、パスワードを入力します。

   このページでは、別の連絡先を使用する場合、デフォルトの `admin` の担当者名を変更できます。ただし、これは表示名にすぎません。Plesk にサインインするためのユーザー名はその後も `admin` になります。



7. エンドユーザー使用許諾契約に同意したことを確認し、 [Enter Plesk (Plesk にアクセスする)] を選択します。

   成功すると、Plesk パネルにサインインされ、そこでドメインを追加して、ウェブサイトの管理を開始できるようになります。

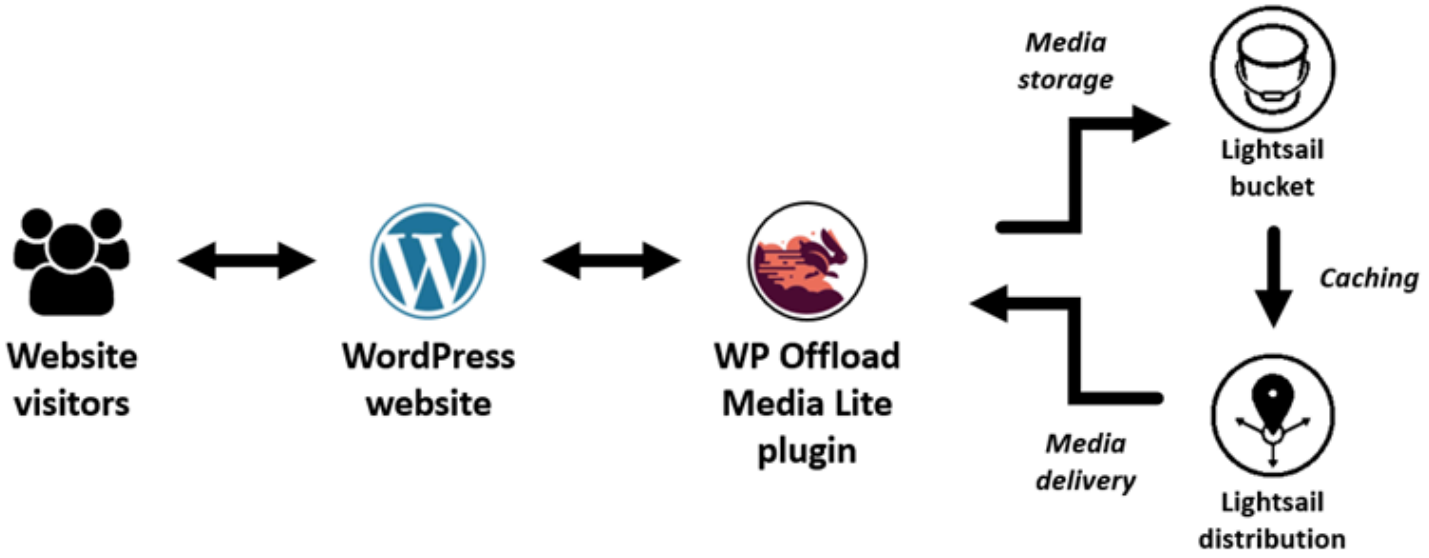


後で再度サインインする必要がある場合は、`https://PublicIPAddress:8443` に移動します 。*PublicIPAddress* は、インスタンスのパブリック IP アドレスまたは静的 IP アドレスに置き換えます。例えば、`https://192.0.2.0/:8443` です。次に、前に作成したユーザー名とパスワードを入力して Plesk パネルにサインインします。

Plesk の使用の詳細については、Plesk ドキュメントとヘルプポータルの「Plesk のウェブサイト管理の開始」を参照してください。

チュートリアル: コンテンツ配信ネットワークディストリビューションで Lightsail バケットを使用する

このチュートリアルでは、Amazon Lightsail バケットを Lightsail コンテンツ配信ネットワーク (CDN) ディストリビューションのオリジンとして設定するために必要な手順について説明します。また、バケットにメディア (イメージや映画ファイルなど) をアップロードして保存し、ディストリビューションからメディアを配信するように WordPress ウェブサイトを設定する方法についても説明します。その方法の一例として、「WP Offload Media Lite プラグイン」があります。次の図にその概念を示します。



ウェブサイトメディアを Lightsail バケットに保存すると、それらのファイルを保存して提供する必要がなくなり、インスタンスの負荷が軽減されます。Lightsail ディストリビューションからメディアをキャッシュして提供すると、ウェブサイトの訪問者へのこれらのファイルの配信が高速化され、ウェブサイト全体のパフォーマンスが向上します。ディストリビューションの詳細については、「コンテンツ配信ネットワークディストリビューション」を参照してください。バケットについての詳細は、「オブジェクトストレージ」を参照してください。

目次

ステップ 1: 前提条件を満たす

以下の前提条件を完了します (まだの場合)。

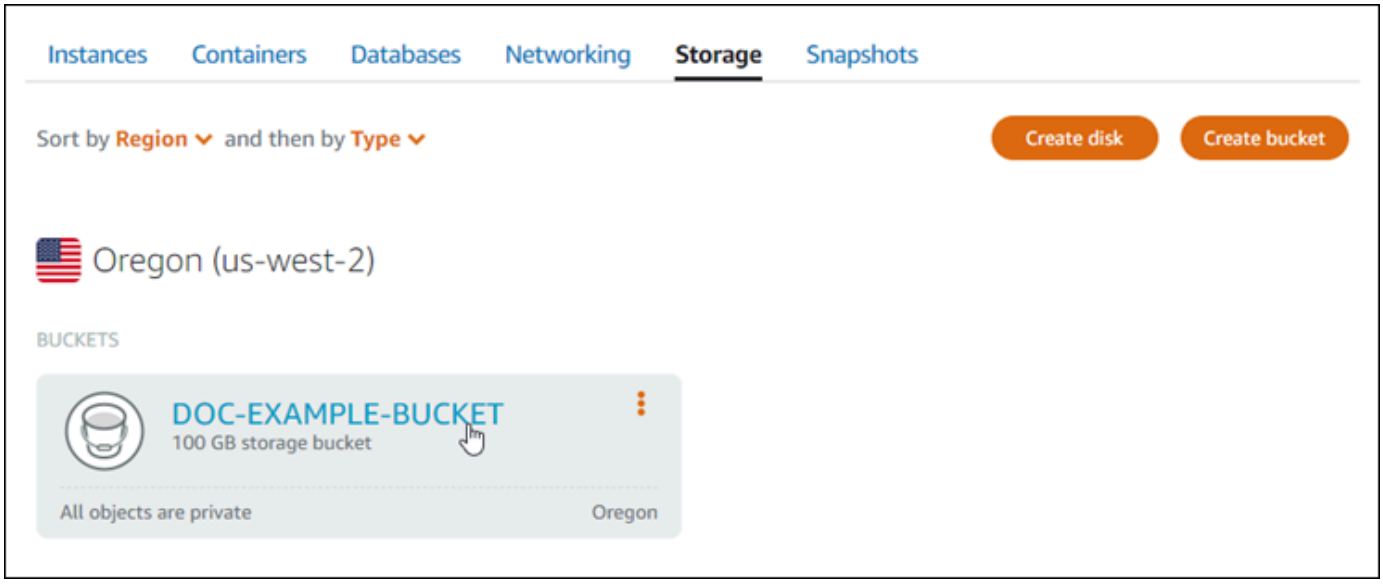
- Lightsail で WordPress インスタンスを作成して設定し、管理ダッシュボードにサインインするためのパスワードを取得します。詳細については、「チュートリアル: Amazon Lightsail で WordPress インスタンスを起動して設定する」を参照してください。

- Lightsail オブジェクトストレージサービスでバケットを作成します。詳細については、「Lightsail でのバケットの作成」を参照してください。

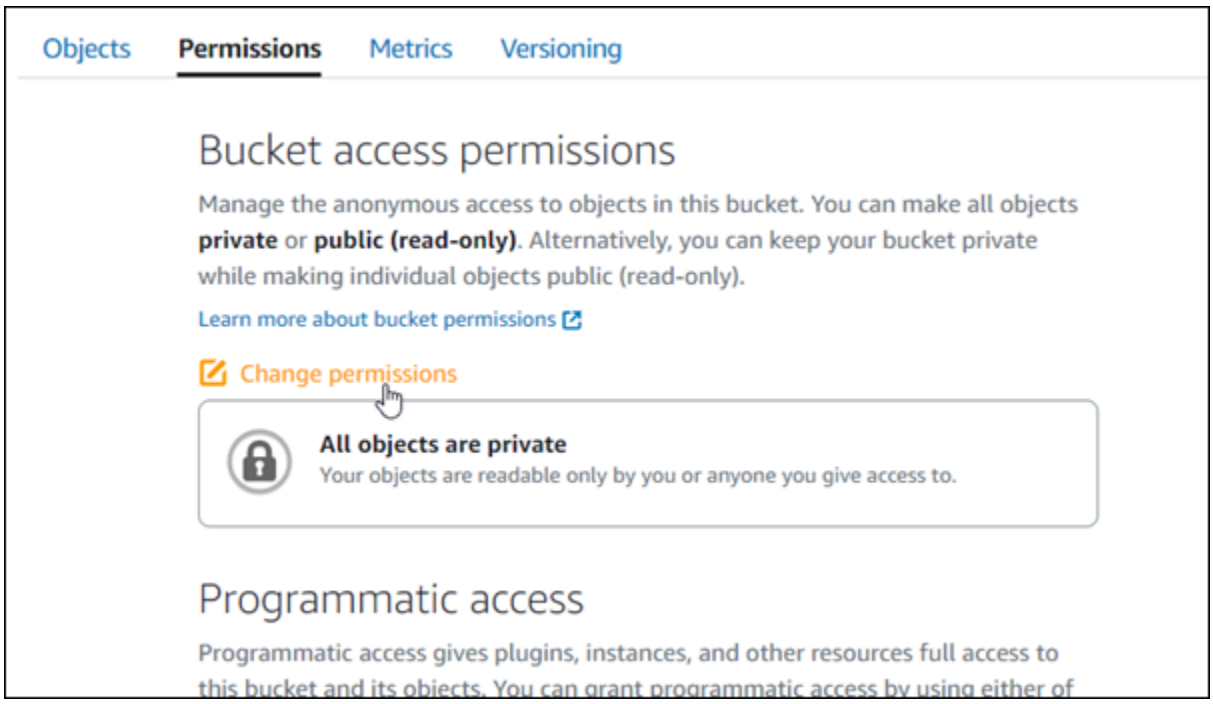
ステップ 2: バケットのアクセス許可を変更する

次の手順を実行して、 WordPress インスタンスと WP Offload Media Lite プラグインにバケットへのアクセスを許可します。バケットのアクセス許可は個々のオブジェクトを公開 (読み取り専用) に設定する必要があります。また、 WordPress インスタンスをバケットにアタッチする必要があります。バケット許可の詳細については、「バケットのアクセス許可」を参照してください。

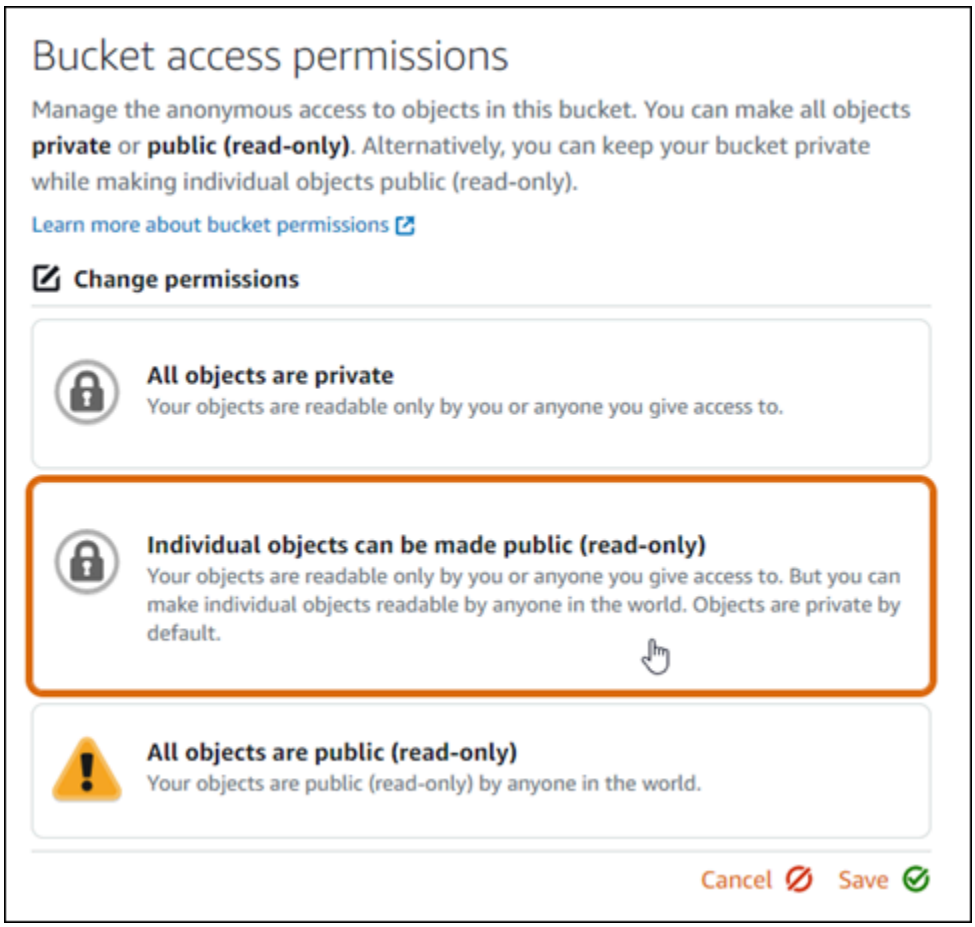
1. Lightsail コンソールにサインインします。
2. Lightsail ホームページで、ストレージタブを選択します。
3. WordPress ウェブサイトで使用するバケットの名前を選択します。



4. バケット管理ページで [Permissions] (許可) タブを選択します。
5. ページの「バケットのアクセス許可」セクションで [Change permissions](許可の変更) を選択します。

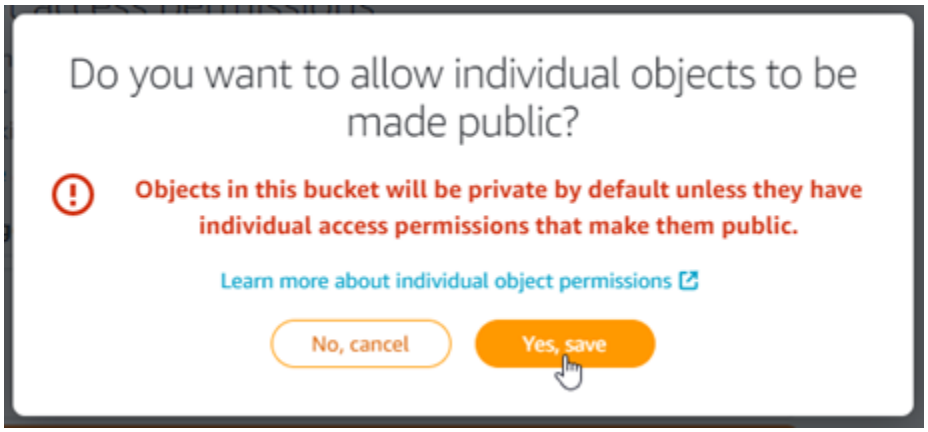


6. 個々のオブジェクトを選択して公開し、読み取り専用にすることができます。



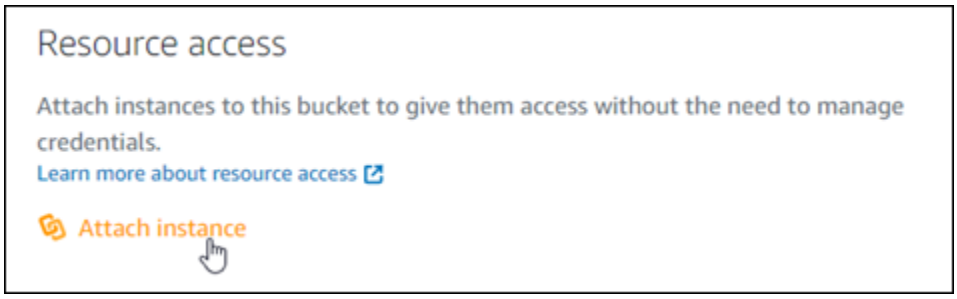
7. [保存] を選択します。

8. 表示される確認プロンプトで、[はい、選択]を選択します。

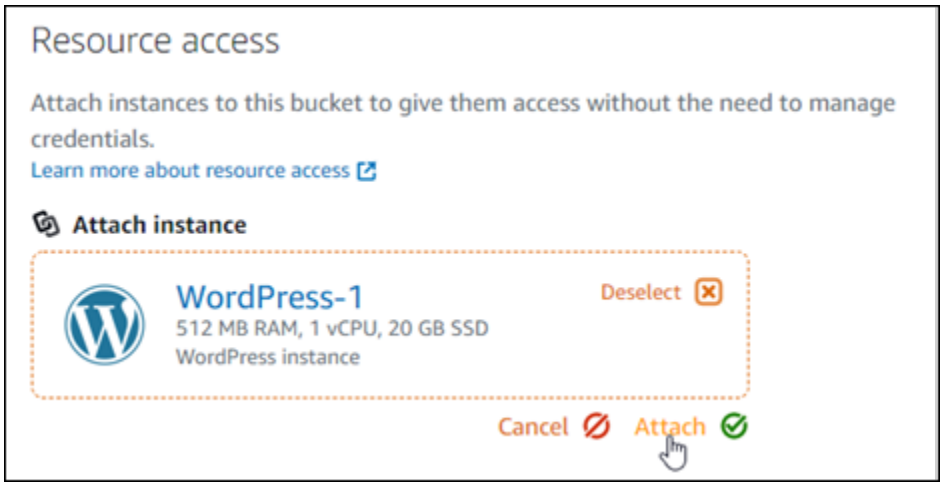


しばらくすると、バケットは個々のオブジェクトにアクセスを許可するように設定されます。これにより、Offload Media Lite プラグインを使用して WordPress ウェブサイトからバケットにアップロードされたオブジェクトを顧客が読み取れるようになります。

9. ページの [リソースアクセス] セクションまでスクロールし、[Attach instance] (インスタンスの添付) を選択します。



10. 表示されるドロップダウンで WordPress インスタンスの名前を選択し、アタッチを選択します。



しばらくすると、 WordPress インスタンスがバケットにアタッチされます。これにより、バケットとそのオブジェクトを管理するためのアクセス権が WordPress インスタンスに付与されます。

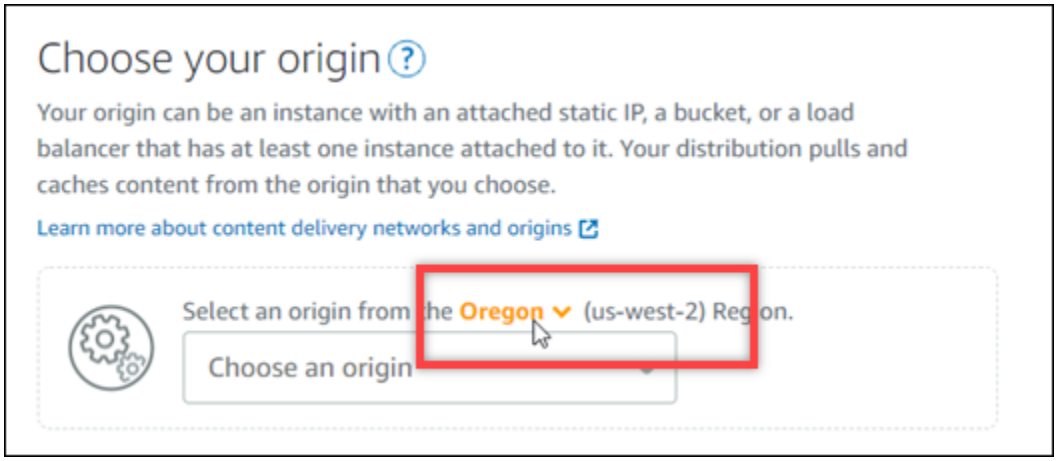
ステップ 3: オリジンとしてのバケットを持つディストリビューションを作成する

Lightsail ディストリビューションを作成し、オリジンとして Lightsail バケットを選択するには、以下の手順を実行します。

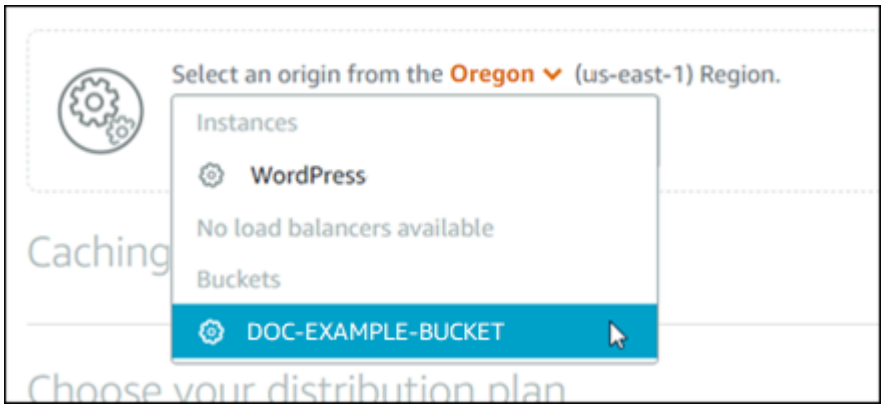
1. Lightsail コンソールの上部のナビゲーションメニューで「ホーム」を選択します。
2. lightsail のホームページで [Networking] (ネットワーク) タブを選択します。
3. [ディストリビューションの作成] を選択します。

4. このページの [オリジンの選択] セクションで、バケットを作成した AWS リージョン を選択します。

   ディストリビューションはグローバルリソースです。どの でもバケットを参照し AWS リージョン、そのコンテンツをグローバルに配信できます。

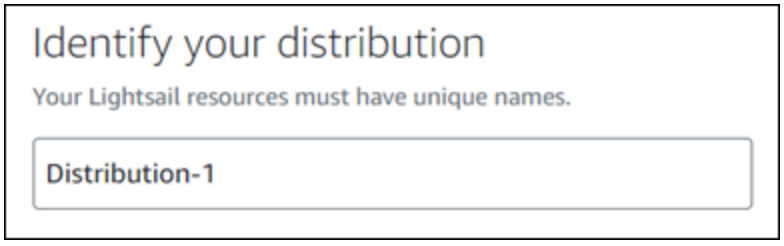


5. バケットをオリジンとして選択します。



> **Note**
>
> バケットのアクセス許可は個々のオブジェクトを公開（読み取り専用）に設定する必要があります。公開として設定されている個々のオブジェクトだけがキャッシュされ、ディストリビューションで配信されます。ディストリビューションのオリジンとしてバケットを選択すると、オリジンプロトコルポリシー、キャッシュ動作、デフォルトの動作、ディレクトリとファイルの上書きを指定するオプションが使用できなくなり、編集もできなくなります。オリジンプロトコルポリシーのデフォルトはバケットに対してのみ [HTTPS Only] に設定され、キャッシュ動作のデフォルトは [すべてをキャッシュする] です。ディストリビューションのアドバンストキャッシュ設定は、ディストリビューションの作成後に変更できます。

6. ディストリビューションプランを選択します。
7. ディストリビューションの名前を入力します。



ディストリビューション名：

- AWS リージョン Lightsail アカウントの各 内で一意である必要があります。
- 2～255 文字を使用する必要があります。
- 先頭と末尾は英数字または数字を使用する必要があります。
- 英数字、数字、ピリオド、ダッシュ、アンダースコアを使用することができます。

8. [ディストリビューションの作成] を選択します。

しばらくすると、ディストリビューションが作成されます。新しいディストリビューションが [Enabled] (有効) になると、バケット内のオブジェクトを提供してキャッシュする準備が整った状態です。

ステップ 4: ディストリビューションのカスタムサブドメインを有効にする

ディストリビューションを作成すると、`123abc.cloudfront.net` と同様のデフォルトドメインで構成されます。WP Offload Media Lite プラグインを設定するときに、そのデフォルトドメインをメディアファイルのソースとして指定することができます。ただし、ディストリビューションのカスタムドメインを有効にすることを強くお勧めします。ディストリビューションで有効にするカスタムドメインは、ウェブサイトで WordPress使用しているドメインのサブドメインである必要があります。例えば、ウェブサイト`mycustomdomain.com`で WordPress を使用している場合、ディストリビューション`media.mycustomdomain.com`でカスタムドメインを使用することを選択できます。ウェブサイトとディストリビューションの間で WordPress同じドメインとサブドメインの組み合わせを使用すると、ウェブサイトの検索エンジン最適化スコアが向上します。

ディストリビューション用のカスタムドメインを設定するには、以下のステップを実行します。

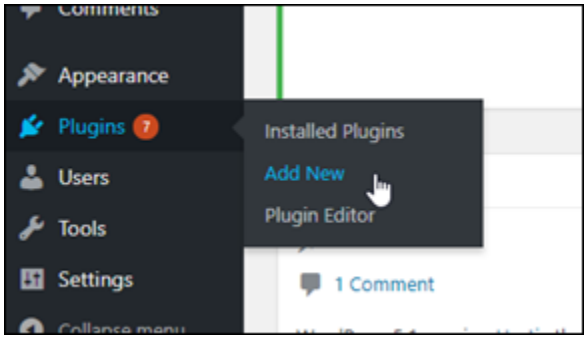
1. ディストリビューションで使用するドメインの Lightsail SSL/TLS 証明書を作成します。Lightsail ディストリビューションには HTTPS が必要なため、ディストリビューションで使用する前に、ドメインの SSL/TLS 証明書をリクエストする必要があります。詳細については、「ディストリビューションの SSL/TLS 証明書を作成する」を参照してください。
2. ディストリビューションでカスタムドメインを有効にして、ディストリビューションでドメインを使用できるようにします。カスタムドメインを有効にするには、ドメイン用に作成した Lightsail SSL/TLS 証明書を指定する必要があります。これにより、ドメインがディストリビューションに追加され、HTTPS が有効になります。詳細については、「ディストリビューション用のカスタムドメインを有効にする」を参照してください。
3. ドメインの DNS ゾーンにエイリアス レコードを追加 エイリアスレコードを追加すると、ドメインにアクセスするユーザーはディストリビューションを通じてルーティングされます。詳細については、「ドメインをディストリビューションにポイントする」を参照してください。

ステップ 5: WordPress ウェブサイトに WP Offload Media Lite プラグインをインストールする

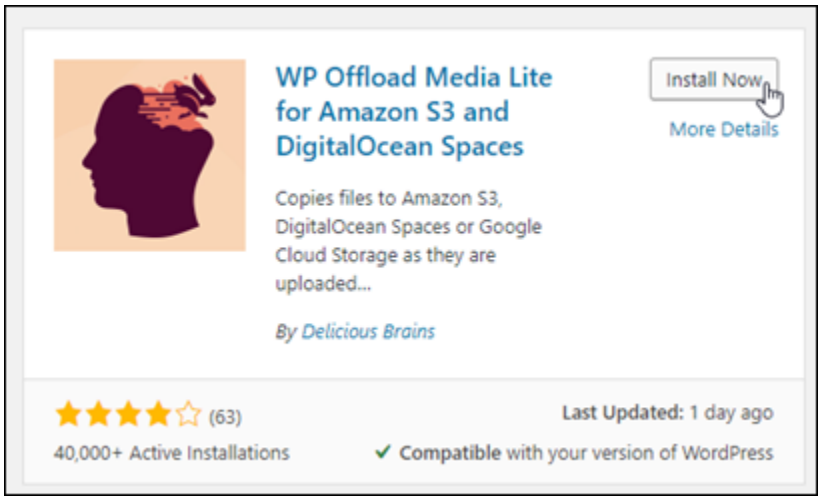
WordPress ウェブサイトに WP Offload Media Lite プラグインをインストールするには、以下の手順を実行します。このプラグインは、 WordPressのメディアアップローダーを介して追加されたイメージ、動画、ドキュメント、およびその他のメディアを Lightsail バケットに自動的にコピーします。Lightsail ディストリビューションを介してバケットからメディアを提供するように設定することもできます。詳細については、 WordPress ウェブサイトの「WP Offload Media Lite」を参照してください。

1. 管理者として WordPress ウェブサイトのダッシュボードにサインインします。

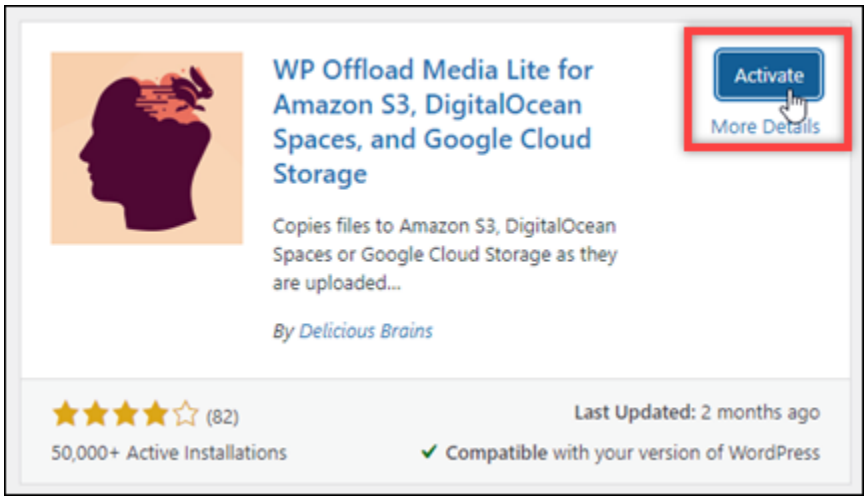
   詳細については、「Amazon Lightsail での Bitnami インスタンスのアプリケーションユーザー名とパスワードの取得」を参照してください。
2. 左側のナビゲーションメニューの [プラグイン] を一時停止し、[Add New] (新規追加) を選択します。



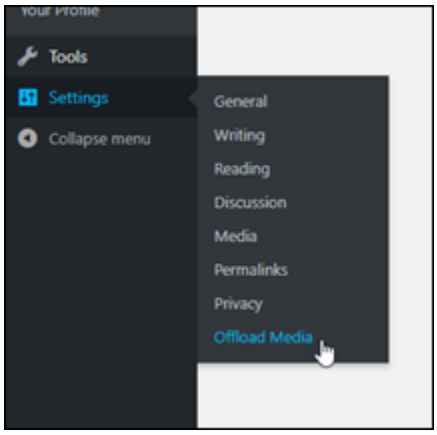
3. [WP Offload Media Lite] を検索します。
4. 検索結果で、[WP Offload MediaLite] プラグインの横にある [Install Now] (今すぐインストール) を選択します。



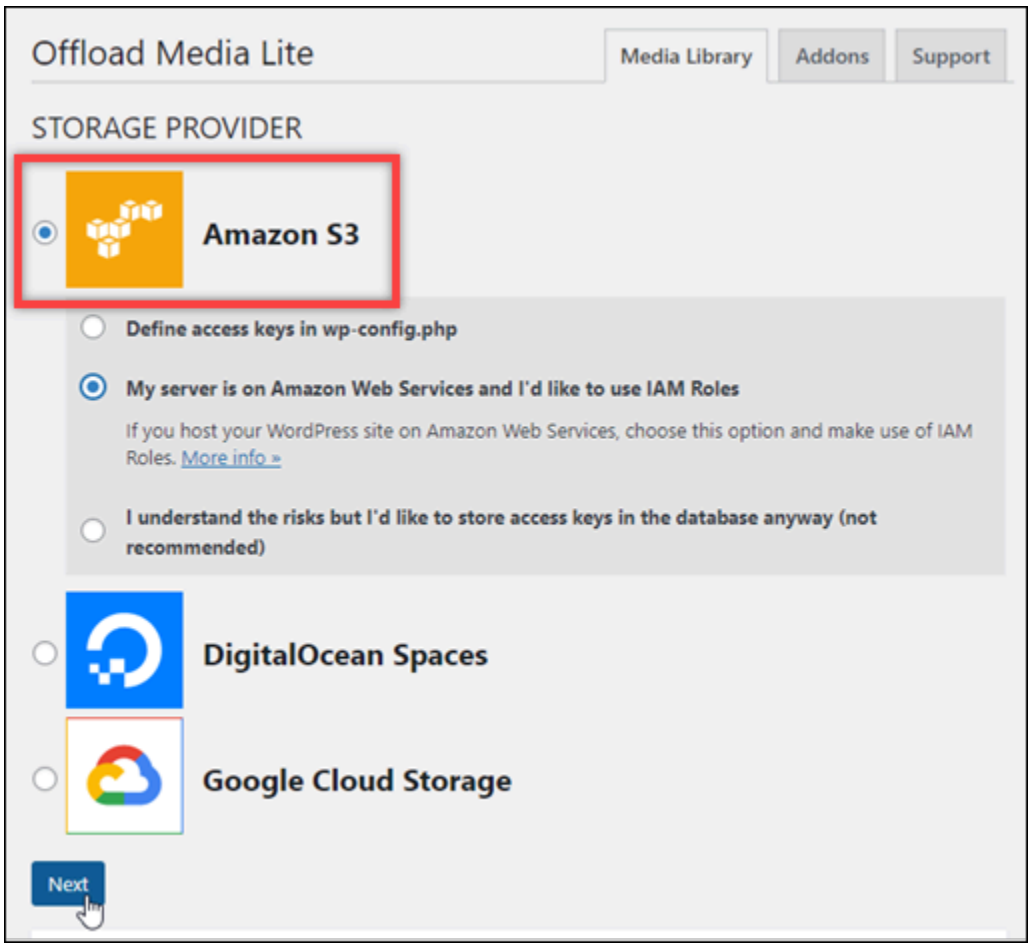
5. プラグインのインストールが完了したら、[アクティベート] を選択します。



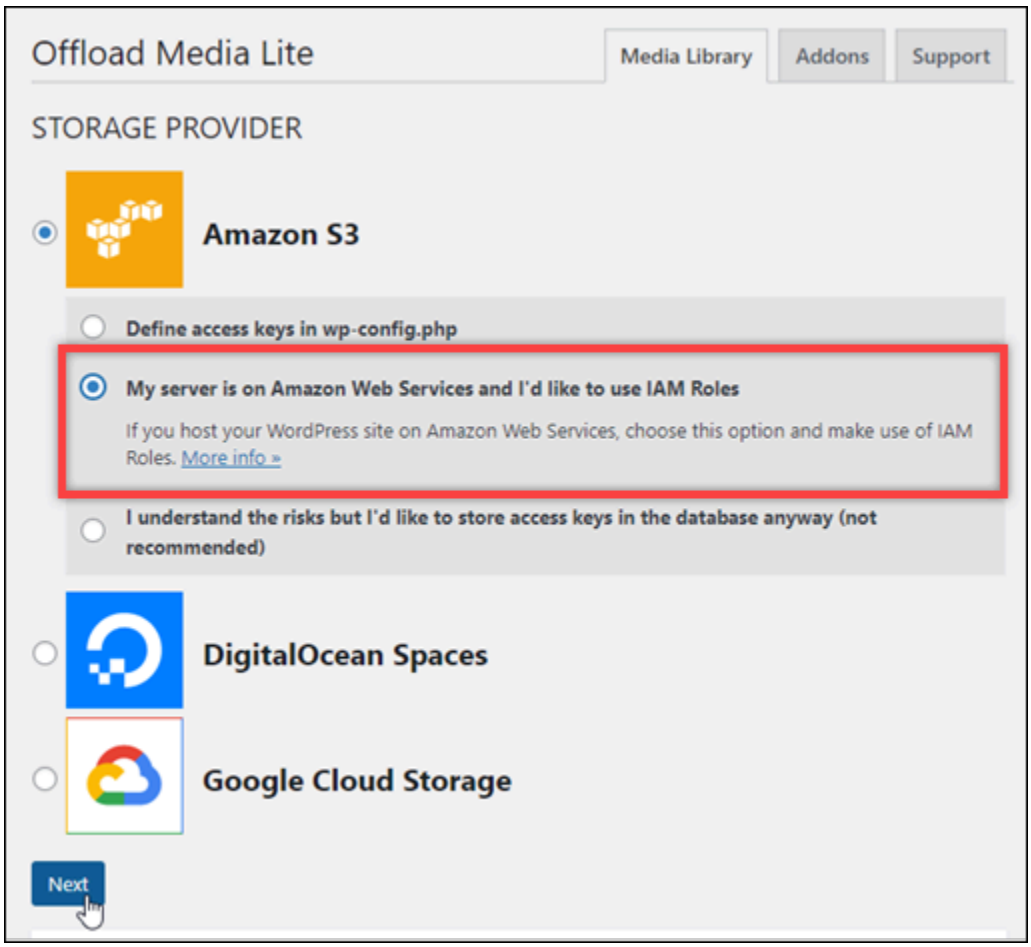
6. 左ナビゲーションメニューで、[設定]、[Offload Media] の順に選択します。



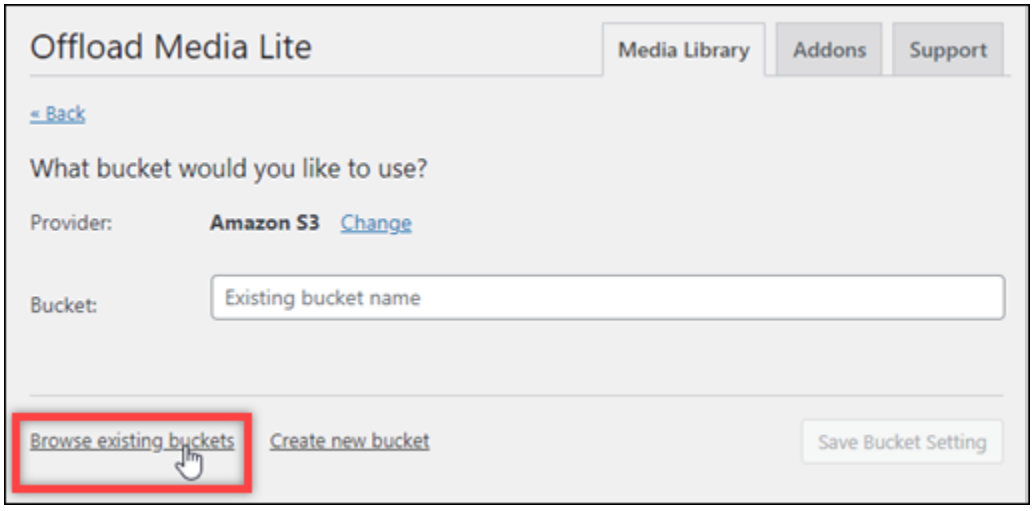
7. [Offload Media Lite] ページで、ストレージプロバイダーとして [Amazon S3] を選択します。



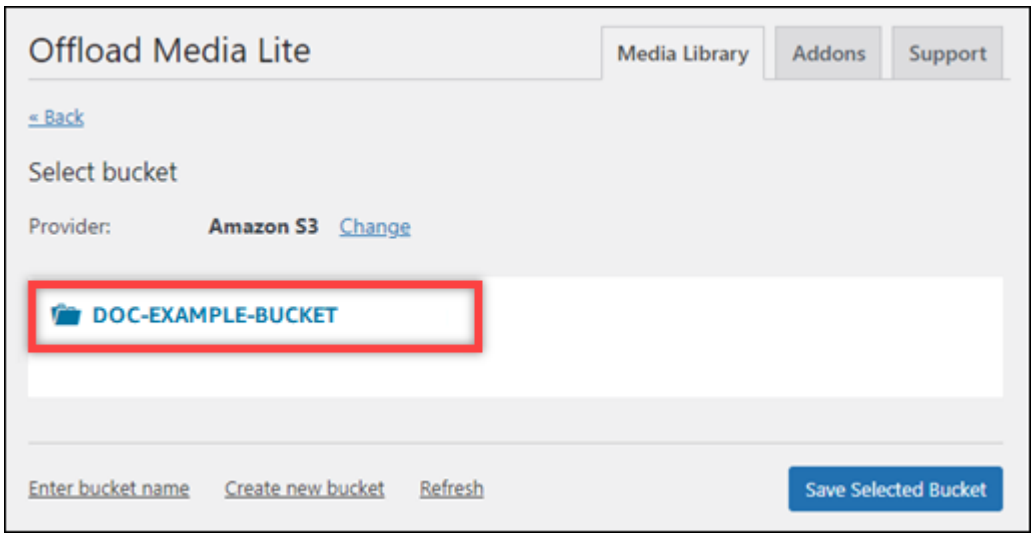
8. [私のサーバーは Amazon Web Services 上にあり、IAM ロールを使いたい] を選択します。



9. [次へ] を選択します。
10. [どのバケットを使用しますか?] と表示される画面で、[Browse existing buckets] (既存のバケットを参照する) を選択します。



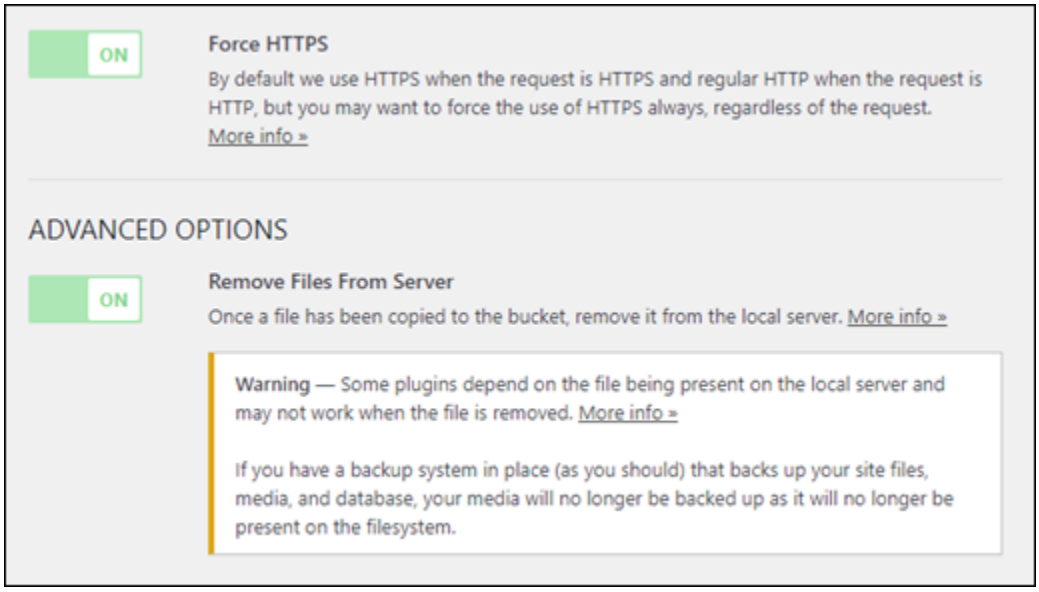
11. インスタンスで WordPress使用するために作成したバケットの名前を選択します。



12. 表示される [Offload Media Lite] 設定画面で、[Force HTTPS] (HTTPS の強制実行) と [Remove Files From Server](サーバーからファイルの削除) をオンにします。

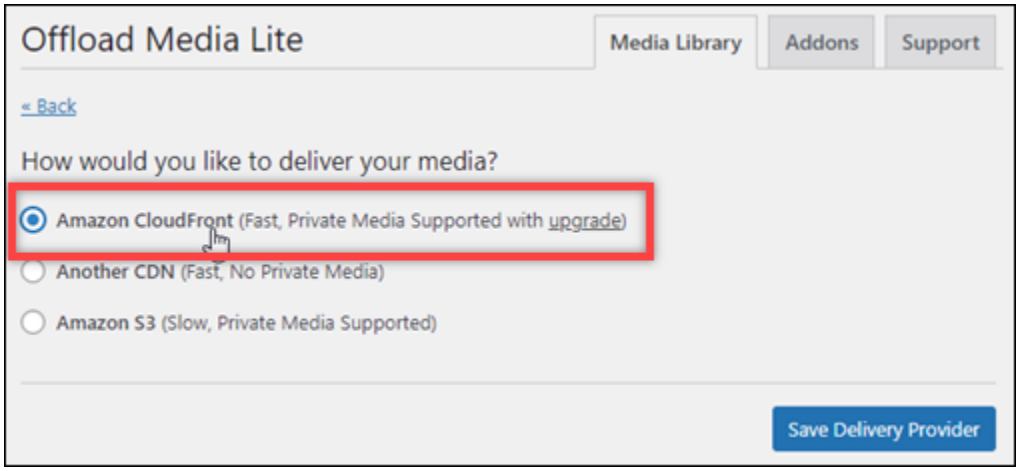
    - Lightsail バケットはデフォルトで HTTPS を使用してメディアファイルを提供するため、強制 HTTPS 設定をオンにする必要があります。この機能をオンにしないと、 WordPress ウェブサイトから Lightsail バケットにアップロードされたメディアファイルはウェブサイトの訪問者に正しく提供されません。

      サーバーからファイルを削除する設定では、Lightsail バケットにアップロードされたメディアがインスタンスのディスクにも保存されないようにします。この機能をオンにしない場合、Lightsail バケットにアップロードされたメディアファイルも WordPress インスタンスのローカルストレージに保存されます。



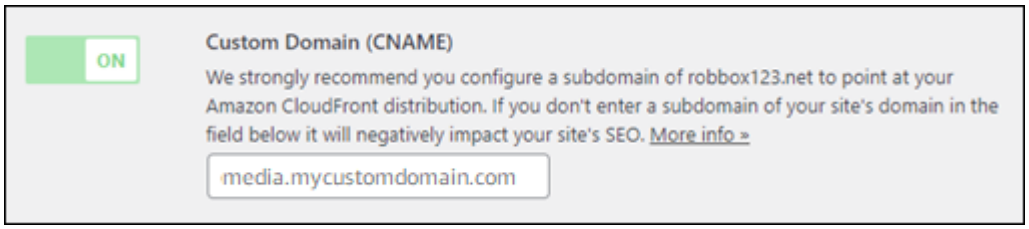
13. ページの [Delivery] セクションで、Amazon S3 ラベルの横にある [変更] を選択します。

14. 「How to deliver your media?」ページが表示されたら、Amazon CloudFrontを選択します。



15. 配信プロバイダを保存を選択。
16. 表示される [Offload Media Lite 設定] 画面で、[カスタムドメイン (CNAME)]をオンにします。次に、Lightsail ディストリビューションのドメインをテキストボックスに入力します。これは、ディストリビューションのデフォルトドメイン (例：`123abc.cloudfront.net`) や、ディス

トリビューションのカスタムドメイン（例：media.mycustomdomain.com ）有効にしている場合は、そのドメインになります。



17. [変更の保存] をクリックします。

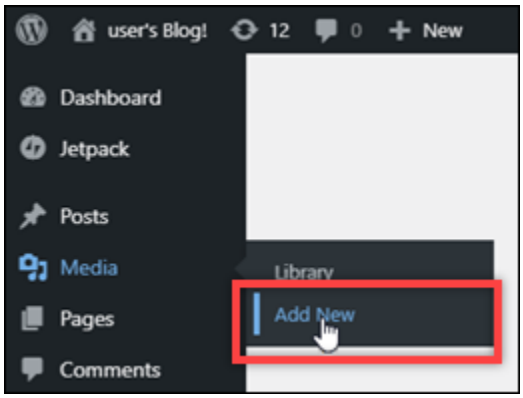
> **Note**
>
> 後で [Offload Media Lite 設定] ページに戻るには、左のナビゲーション メニューで [設定] を一時停止し、[Offload Media] を選択します。

これで、Media Lite プラグインを使用するように WordPress ウェブサイトが設定されました。次回 を介してメディアファイルをアップロードすると WordPress、そのファイルは自動的に Lightsail バケットにアップロードされ、ディストリビューションによって提供されます。設定をテストするには、このチュートリアルの次のセクションに進みます。

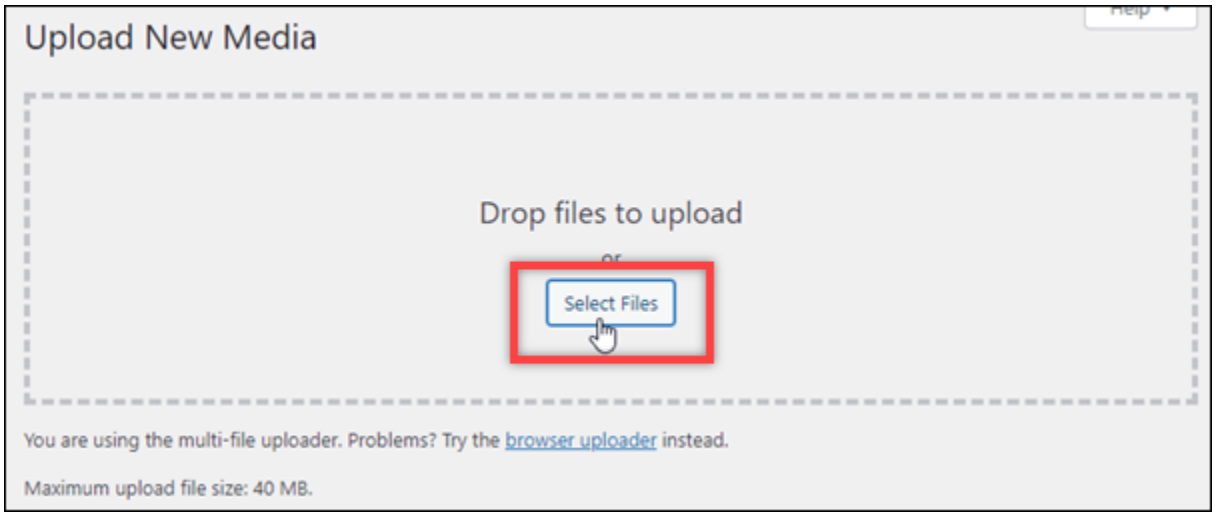
ステップ 6: WordPress ウェブサイトと Lightsail バケットおよびディストリビューション間の接続をテストする

次の手順を実行して、メディアファイルを WordPress インスタンスにアップロードし、Lightsail バケットにアップロードされ、ディストリビューションから提供されることを確認します。

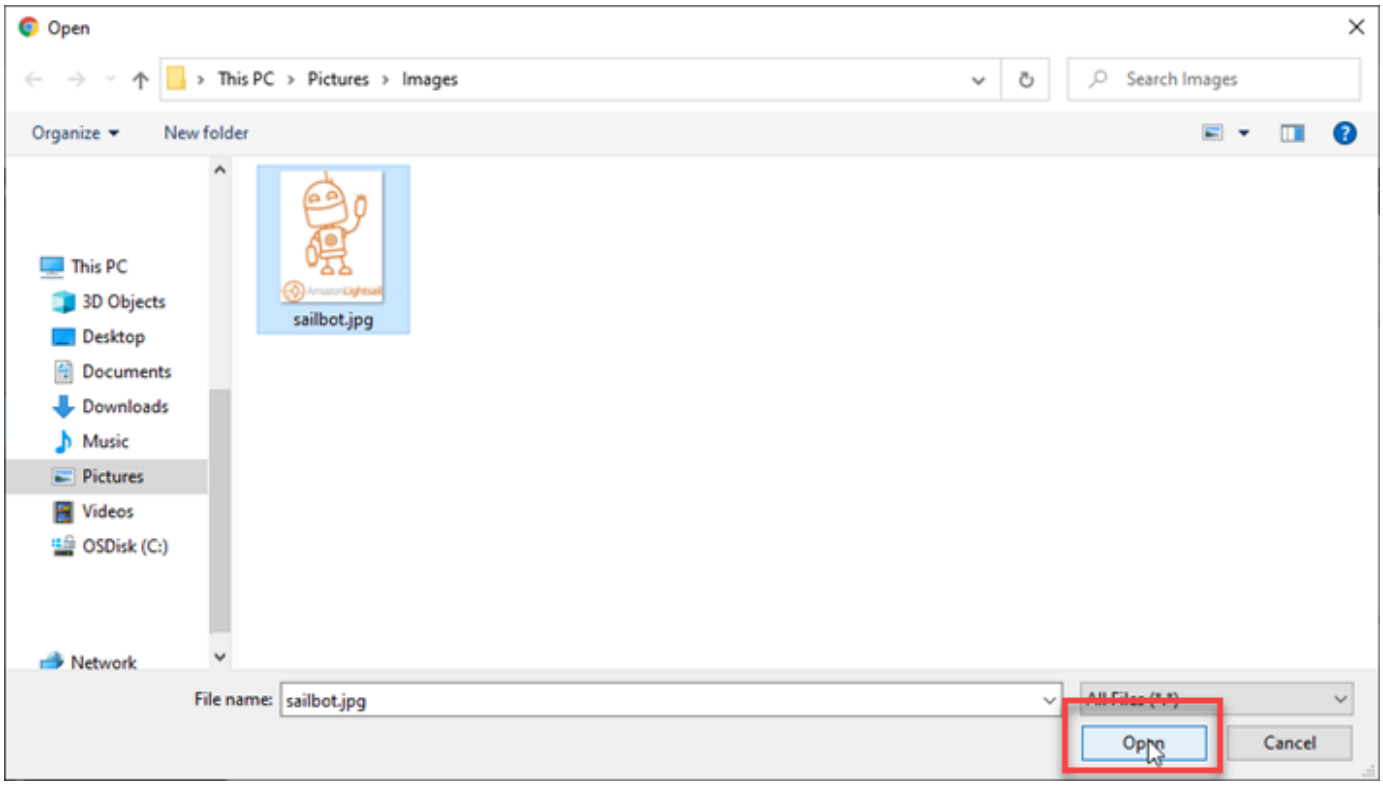
1. ダッシュボードの WordPress左側のナビゲーションメニューでメディアで一時停止し、新規追加を選択します。



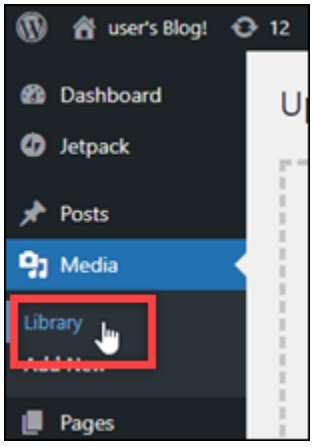
2. 表示される [新しいメディアのアップロード] ページで [ファイルを選択] を選択します。



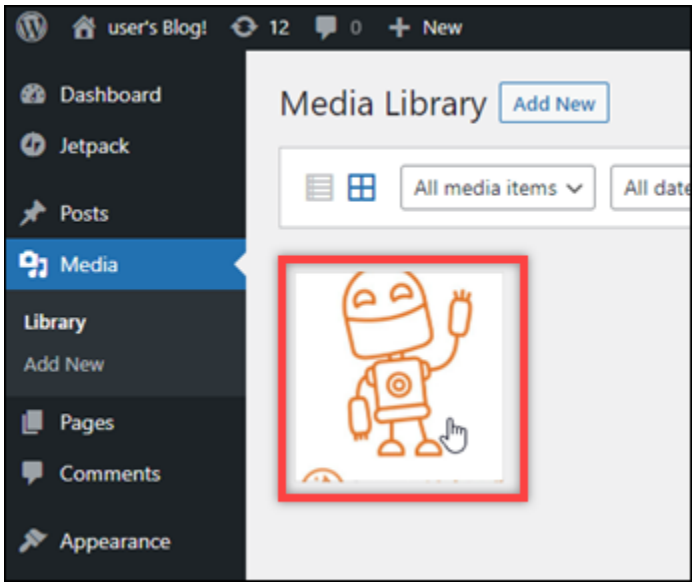
3. ローカルコンピュータからアップロードするメディアファイルを選択し、[開く] を選択します。



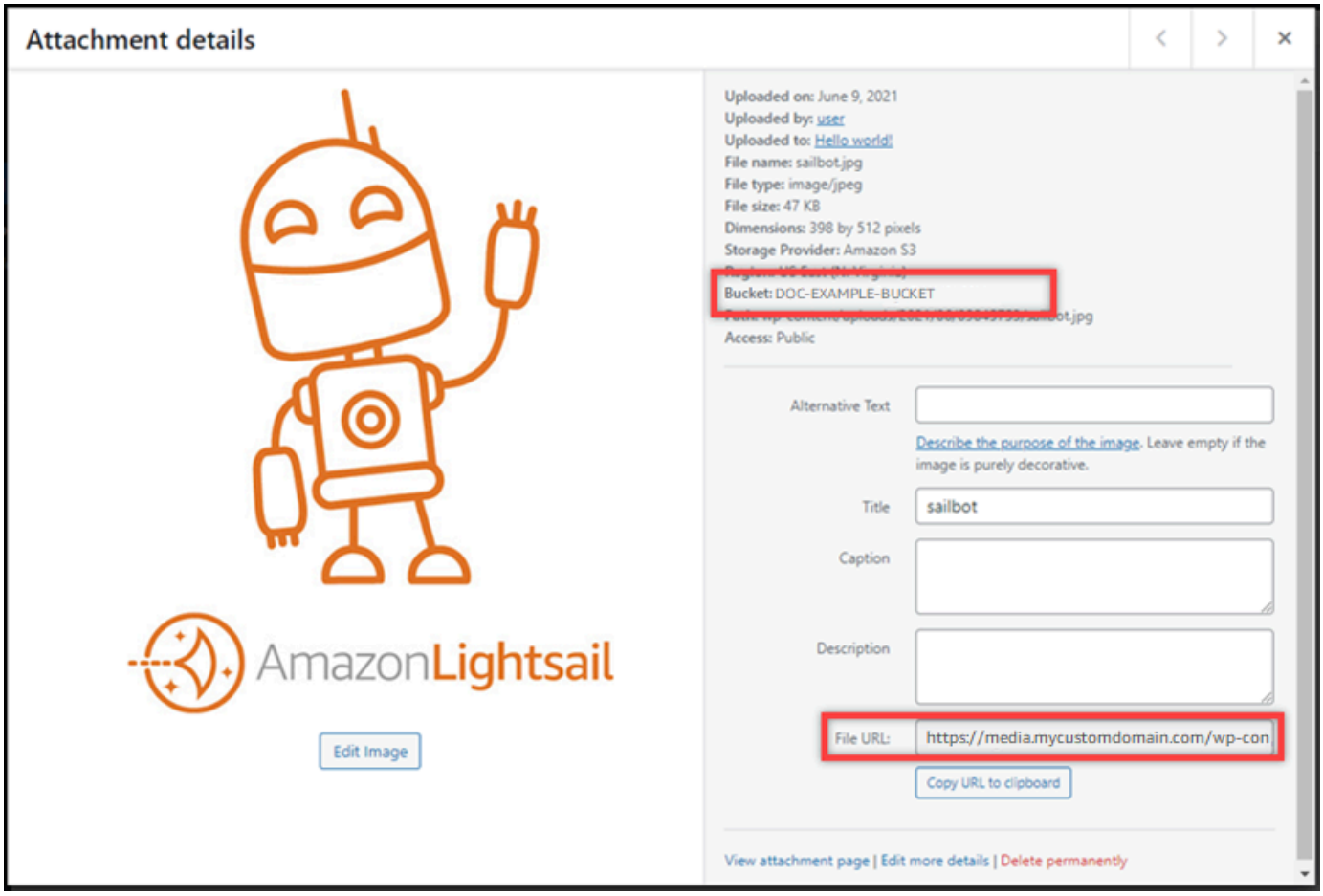
4. ファイルのアップロードが完了したら、左のナビゲーションメニューにある [メディア] の [ライブラリ] を選択します。



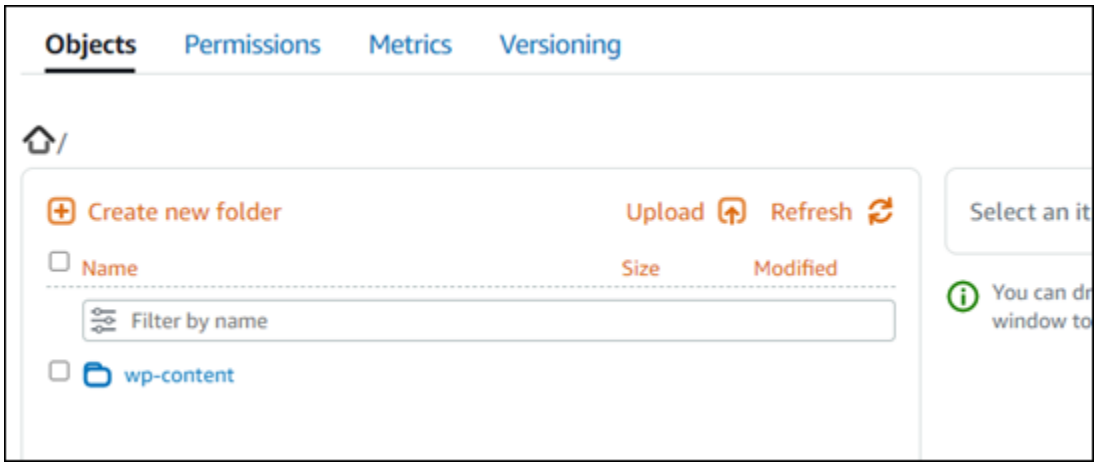
5. 最近アップロードしたファイルを選択します。



6. ファイルの詳細パネルで、[バケット] フィールドにバケットの名前が表示されます。[ファイルの URL」フィールドには、ディストリビューションの URL が表示されます。



7. Lightsail バケット管理ページのオブジェクトタブに移動すると、wp-content フォルダが表示されます。このフォルダは、Offload Media Lite プラグインによって作成され、アップロードしたメディアファイルを保存するために使用されます。



バケットとオブジェクトを管理する

Lightsail オブジェクトストレージバケットを管理する一般的な手順は次のとおりです。

1. Amazon Lightsail オブジェクトストレージサービスのオブジェクトとバケットについて説明します。詳細については、Amazon Lightsail のオブジェクトストレージ を参照してください。
2. Amazon Lightsail でバケットに付けることができる名前について説明します。詳細については、「Amazon Lightsail のバケット命名規則」を参照してください。
3. バケットを作成して Lightsail オブジェクトストレージサービスの使用を開始します。詳細については、「Amazon Lightsail でのバケットの作成」を参照してください。
4. バケットのセキュリティのベストプラクティスと、バケットに設定できるアクセス許可について説明します。バケット内のすべてのオブジェクトをパブリックまたはプライベートにすることも、オブジェクトを個別に選択してパブリックにすることもできます。また、アクセスキーを作成し、インスタンスをバケットに追加し、他の AWS アカウントにアクセス権を付与することで、バケットへのアクセスを許可することもできます。詳細については、「Amazon Lightsail オブジェクトストレージのセキュリティのベストプラクティス」および「Amazon Lightsail でのバケットのアクセス許可について」を参照してください。

   バケットのアクセス許可について理解したら、以下のガイドを参照してバケットへのアクセスを許可してください。

   - Amazon Lightsail のバケットへのパブリックアクセスをブロックする
   - Amazon Lightsail でのバケットアクセス許可の設定
   - Amazon Lightsail のバケット内の個々のオブジェクトに対するアクセス許可の設定
   - Amazon Lightsail でのバケットのアクセスキーの作成
   - Amazon Lightsail でのバケットのリソースアクセスの設定
   - Amazon Lightsail でのバケットのクロスアカウントアクセスの設定
5. バケットのアクセスログの記録を有効にする方法と、アクセスログを使用してバケットのセキュリティを監査する方法について説明します。詳細については、以下のガイドを参照してください。

   - Amazon Lightsail オブジェクトストレージサービスでのバケットのアクセスログ記録
   - Amazon Lightsail オブジェクトストレージサービスのバケットのアクセスログ形式
   - Amazon Lightsail オブジェクトストレージサービスでのバケットのアクセスログ記録の有効化
   - Amazon Lightsail でバケットのアクセスログを使用してリクエストを識別する

6. Lightsail でバケットを管理する権限をユーザーに付与する IAM ポリシーを作成します。詳細については、「Amazon Lightsail でバケットを管理する IAM ポリシー」を参照してください。
7. バケット内のオブジェクトにラベルを付けて識別する方法について説明します。詳細については、「Amazon Lightsail でのオブジェクトキー名の理解」を参照してください。
8. ファイルをアップロードしてバケット内のオブジェクトを管理する方法について説明します。詳細については、以下のガイドを参照してください。
   - Amazon Lightsail のバケットにファイルをアップロードする
   - マルチパートアップロードを使用した Amazon Lightsail のバケットへのファイルのアップロード
   - Amazon Lightsail でバケット内のオブジェクトを表示する
   - Amazon Lightsail のバケット内のオブジェクトのコピーまたは移動
   - Amazon Lightsail のバケットからオブジェクトをダウンロードする
   - Amazon Lightsail のバケット内のオブジェクトのフィルタリング
   - Amazon Lightsail のバケット内のオブジェクトのタグ付け
   - Amazon Lightsail でバケット内のオブジェクトを削除する
9. オブジェクトのバージョニングを有効にすると、バケットに保存されたあらゆるオブジェクトのあらゆるバージョンを保存、取得、復元します。詳細については、「Amazon Lightsail のバケットでのオブジェクトのバージョニングの有効化と一時停止」を参照してください。
10. オブジェクトのバージョニングを有効にすると、バケット内のオブジェクトの以前のバージョンを復元できます。詳細については、「Amazon Lightsail のバケット内のオブジェクトの以前のバージョンの復元」を参照してください。
11. バケットの使用率を監視します。詳細については、「Amazon Lightsail でのバケットのメトリクスの表示」を参照してください。
12. バケットの使用率がしきい値を超えたときにバケットメトリクスが通知されるよう、アラームを設定します。詳細については、「Amazon Lightsail でのバケットメトリクスアラームの作成」を参照してください。
13. ストレージとネットワーク転送量が不足している場合は、バケットのストレージプランを変更します。詳細については、「Amazon Lightsail でのバケットのプランの変更」を参照してください。
14. バケットを他のリソースに接続する方法について説明します。詳細については、以下のチュートリアルを参照してください。
   - チュートリアル: WordPress インスタンスを Amazon Lightsail バケットに接続する

- チュートリアル: Lightsail コンテンツ配信ネットワークディストリビューションで Amazon Lightsail バケットを使用する

15.使用しなくなったバケットを削除します。詳細については、「Amazon Lightsail でのバケットの削除」を参照してください。

Lightsail を他の AWS サービスで使用する

Amazon Lightsail では、Amazon EC2 や AWS などのサービスセットを重点 AWS Identity and Access Management 的に使用して、簡単に使用を開始できるようにします。ただし、サービスがこれらに限定されるわけではありません。

Amazon VPC ピアリングを介して Lightsail リソースを他の AWS サービスと統合できます。VPC ピア接続をセットアップする方法をご覧ください。

その他の AWS サービスの詳細については、以下のリンクを参照してください。

仮想マシン (仮想プライベートサーバー)

Amazon EC2

Amazon Elastic Compute Cloud (Amazon EC2) は、クラウド内で安心でサイズを変更できるコンピューティング性能を提供するウェブサービスです。ウェブスケールのクラウドコンピューティングを開発者が簡単に利用できるように設計されています。

Amazon EC2 では、最小限の手間で容量を取得して設定できます。使用するコンピューティングリソースのあらゆる面をお客様自身でコントロールできることと、Amazon の実績あるコンピューティング環境で実行できることが特徴です。Amazon EC2 であれば、新規サーバーインスタンスの取得と起動に要する時間が数分にまで短縮されるため、キャパシティーの拡張や縮小も、コンピューティング要件の変化に合わせてすばやく実行できます。Amazon EC2 はコンピューティングの経済性も変革します。料金のお支払いは、実際に使用したキャパシティーの分だけです。Amazon EC2 には、障害に強いアプリケーションを構築して、一般的な障害シナリオに影響されないようにするための開発者向けツールが用意されています。

Amazon EC2 の詳細をご覧ください。

Amazon VPC

Amazon Virtual Private Cloud (Amazon VPC) では、 AWS クラウドの論理的に隔離されたセクションをプロビジョニングすることで、ユーザーが定義した仮想ネットワーク内で AWS リソー

スを起動できます。独自の IP アドレス範囲の選択、サブネットの作成、ルートテーブル、ネットワークゲートウェイの設定など、仮想ネットワーク環境全体をお客様がコントロールできます。

Amazon VPC のネットワーク設定は容易にカスタマイズすることができます。たとえば、インターネットにアクセスが可能なウェブサーバーのパブリックサブネットを作成し、データベースやアプリケーションサーバーなどのバックエンドシステムをインターネットにアクセスできないプライベートサブネットに配置できます。セキュリティグループやネットワークアクセスコントロールリストなどの複数のセキュリティレイヤーを活用し、各サブネットの Amazon EC2 インスタンスへのアクセスをコントロールすることができます。

加えて、既存のデータセンターと自分の VPC 間にハードウェア Virtual Private Network (VPN) 接続を作成することができるので、AWS クラウドを既存のデータセンターを拡張するかのように活用することができます。

Amazon VPC の詳細をご覧ください。

サーバーレスコンピューティング

AWS Lambda

AWS Lambda では、サーバーのプロビジョニングや管理を行わずにコードを実行できます。使用したコンピューティング時間に対してのみお支払いいただきます。コードが実行中でなければ料金はかかりません。Lambda を使用すれば、実質どのようなタイプのアプリケーションやバックエンドサービスでも、管理をまったく必要とせずに実行できます。コードをアップロードするだけで、コードの実行とスケールに必要な処理はすべて Lambda により自動的に実行され、高い可用性が維持されます。コードは、他の AWS サービスから自動的にトリガーするよう設定することも、ウェブやモバイルアプリケーションから直接呼び出すよう設定することもできます。

の詳細をご覧ください AWS Lambda。

Amazon API Gateway

Amazon API Gateway は、デベロッパーがあらゆる規模で API の公開、保守、モニタリング、セキュリティ保護を簡単に行えるフルマネージドサービスです。 AWS Management Console で数回クリックするだけで、アプリケーションの「玄関」として機能する API を作成できます。これによってバックエンドサービスからデータ、ビジネスロジック、機能にアクセスできます。これには Lambda または任意のウェブアプリケーションで実行中のコードを、Amazon EC2 で実行しているワークロードが含まれます。Amazon API Gateway では、最大で数十万個の同時 API コールの受け入れ処理に伴うすべてのタスクを取り扱います。これにはトラフィック管理、認証とア

クセスコントロール、モニタリング、API バージョン管理が含まれます。Amazon API Gateway には最低利用料金がなく、初期費用はかかりません。受信した API 呼び出しと、送出されたデータ量に対してのみ料金が発生します。

Amazon API Gateway の詳細をご覧ください。

データベース

Amazon DynamoDB

Amazon DynamoDB は、あらゆるスケールで 1 桁台のミリ秒単位の安定したレイテンシーを必要とする全アプリケーションに対応した、高速かつフレキシブルな NoSQL データベースサービスです。完全マネージド型のクラウドデータベースで、ドキュメントとキー値のストアモデルの両方をサポートしています。データモデルの柔軟性が高く、パフォーマンスが信頼できるため、モバイル、ウェブ、ゲーム、広告、IoT、他の多くのアプリケーションに最適です。

DynamoDB の詳細をご覧ください。

Amazon RDS

Amazon Relational Database Service (Amazon RDS) を使用して、クラウドでリレーショナルデータベースをセットアップ、運用、スケーリングできます。これにより、時間のかかるデータベース管理作業をお客様の代わりに実行して、お客様を管理業務から解放し、アプリケーションとビジネスに集中させることができます。このサービスはコスト効率もよく、データベース容量の変更にも柔軟に対応します。Amazon RDS では、Amazon Aurora、PostgreSQL、MySQL、MariaDB、Oracle、および Microsoft SQL Server の 6 つの使い慣れたデータベースエンジンから選択できます。

Amazon Aurora の詳細をご覧ください。

Amazon Aurora

Amazon Aurora は、MySQL と互換性のあるリレーショナルデータベースエンジンで、高性能の商用データベースの可用性とスピード、およびオープンソースデータベースのシンプルさとコスト効果性を併せ持っています。Aurora は、MySQL よりも最大 5 倍のパフォーマンスを発揮するだけでなく、商用データベースのセキュリティ、可用性、および信頼性を 10 分の 1 のコストで実現します。

Amazon Aurora の詳細を確認してください。

ロードバランサー

Elastic Load Balancing

Elastic Load Balancing は、受信アプリケーショントラフィックを複数の Amazon EC2 インスタンスに自動的に分散します。これにより、アプリケーションの耐障害性の向上を可能にし、アプリケーショントラフィックのルーティングに必要なロードバランシング能力をシームレスに提供します。

Elastic Load Balancing では、2 種類のロードバランサーがサポートされています。いずれも高可用性、自動スケーリング、および強固なセキュリティを備えています。これには、アプリケーションまたはネットワークレベルの情報に基づいてトラフィックをルーティングする Classic Load Balancer と、リクエストのコンテンツを含むアプリケーションレベルの詳細情報に基づいてトラフィックをルーティングする Application Load Balancer が含まれます。Classic Load Balancer は、複数の Amazon EC2 インスタンス間でのシンプルなトラフィックのロードバランシングに最適です。Application Load Balancer は、高度なルーティング機能、マイクロサービス、およびコンテナベースのアーキテクチャが必要なアプリケーションに最適です。Application Load Balancer は、トラフィックを複数のサービスにルーティングしたり、負荷を同じ Amazon EC2 インスタンスの複数のポートに分散したりする機能を提供します。

Elastic Load Balancing の詳細をご覧ください。

Application Load Balancer

Application Load Balancer は、アプリケーションレイヤーで動作する サービスのロードバランシングオプションで、1 つ以上の Amazon EC2 インスタンスで実行されている複数のサービスやコンテナのコンテンツに基づいてルーティングルールを定義できます。

Application Load Balancer の詳細をご覧ください。

ビッグデータ

Amazon Kinesis のサービス

Amazon Kinesis サービスにより、AWS クラウドのリアルタイムストリーミングデータとの連携が容易になります。Amazon Kinesis サービスには、大量のストリーミングデータを AWS に簡単にロードするための Amazon Data Firehose、標準 SQL を使用してストリーミングデータを分析する Amazon Managed Service for Apache Flink、ストリーミングデータを処理または分析する独自のカスタムアプリケーションを構築する Amazon Kinesis Data Streams が含まれます。

Amazon Kinesis サービスの詳細をご覧ください。

Amazon EMR

Amazon EMR は、動的にスケーラブルな Amazon EC2 インスタンス間で大量のデータを簡単、迅速、費用効率よく処理できるマネージド型 Hadoop フレームワークを提供します。また、Apache Spark、HBase、Presto、Flink などの他の一般的なフレームワークを Amazon EMR で実行したり、Amazon S や DynamoDB などの他の AWS データストア内のデータを操作したりもできます。

Amazon EMR では、ログの解析、ウェブインデックス作成、データ変換 (ETL)、機械学習、財務分析、科学シミュレーション、バイオインフォマティクスを含む、さまざまなビッグデータのユースケースが安全かつ確実に処理されます。

Amazon EMR の詳細をご覧ください。

Amazon Redshift

Amazon Redshift は、高速で完全マネージド型のペタバイト規模を誇るデータウェアハウスです。シンプルで費用対効果の高さが特長であり、お客様はすべてのデータを既存のビジネスインテリジェンスツールで分析できます。

Amazon Redshift の詳細をご覧ください。

[Storage (ストレージ)]

Amazon Simple Storage Service (Amazon S3)

Amazon S3 では、開発者や IT チームのための安全で耐久性に優れ、高度にスケーラブルなクラウドストレージが用意されています。Amazon S3 は easy-to-use オブジェクトストレージで、ウェブ上の任意の場所から任意の量のデータを保存および取得するためのシンプルなウェブサービスインターフェイスを備えています。Amazon S3 のお支払いは、実際に使用したストレージ分のみです。最低料金もセットアップ費用も不要です。

Amazon S3 は、頻繁にアクセスするデータの汎用ストレージのための Amazon S3 Standard 、長期保存を要し、かつアクセス頻度の低いデータのための Amazon S3 Standard -Infrequent Access (Standard - IA)、長期アーカイブのための S3 Glacier など、さまざまなユースケースに応じて設計された各種のストレージクラスを提供します。Amazon S3 はまた、データのライフサイクルを通したデータ管理のために設定可能なライフサイクルポリシーを提供します。ポリシーを設定すると、データは自動的に最も適切なストレージクラスに移行します。アプリケーションの変更は一切必要ありません。

Amazon S3 は単独で使用することも、Amazon EC2 や IAM など他の AWS サービス、および、初回のみまたは継続的にデータの取り込みを行うクラウドデータ移行サービスやゲートウェイと一緒に使用することもできます。Amazon S3 では、バックアップと復元、nearline アーカイブ、ビッグデータ分析、ディザスタリカバリ、クラウドアプリケーション、コンテンツ配信など、さまざまなユースケースにおいてコスト効率に優れたオブジェクトストレージを利用できます。

Amazon S3 の詳細をご覧ください。

Amazon Elastic Block Store (Amazon EBS)

Amazon EBS は、AWS クラウドの Amazon EC2 インスタンス用の永続的なブロックストレージボリュームを提供します。コンポーネントに障害が発生した場合でも高い可用性と耐久性を提供できるように、各 Amazon EBS ボリュームはアベイラビリティーゾーン内で自動的にレプリケートされます。Amazon EBS のボリュームは、ワークロードの実行に必要な一貫した低レイテンシーのパフォーマンスを実現します。Amazon EBS を使用すると、使用量の拡張または縮小を数分で行うことができます – プロビジョニングしているサイズに合わせて、低料金でご利用いただけます。

Amazon EBS の詳細をご覧ください。

モニタリングとアラーム

Amazon CloudWatch

Amazon CloudWatch は、AWS クラウドリソースと AWS で実行するアプリケーションのモニタリングサービスです。 CloudWatch を使用して、メトリクスの収集と追跡、ログファイルの収集とモニタリング、アラームの設定、AWS リソースの変更への自動対応を行うことができます。CloudWatch は、Amazon EC2 インスタンス、Amazon DynamoDB テーブル、Amazon RDS DB インスタンスなどの AWS リソース、およびアプリケーションとサービスによって生成されたカスタムメトリクス、およびアプリケーションが生成するログファイルをモニタリングできます。を使用して CloudWatch 、リソースの使用率、アプリケーションのパフォーマンス、および運用状態をシステム全体で把握できます。これらの洞察を使用して対応し、アプリケーションのスムーズな動作を維持できます。

Amazon の詳細をご覧ください CloudWatch。

アプリケーションのデプロイ

AWS Elastic Beanstalk

AWS Elastic Beanstalk は、Java、.NET、PHP、Node.js、Python、Ruby、Go、Docker で開発されたウェブアプリケーションとサービスを Apache、Nginx、Passenger、IIS などの使い慣れたサーバーにデプロイおよびスケーリングするための easy-to-use のサービスです。

お客様はコードをアップロードするだけで、Elastic Beanstalk が、キャパシティーのプロビジョニング、ロードバランシング、自動スケーリング からアプリケーションの状態モニタリングまで、デプロイを自動的に処理します。同時に、お客様のアプリケーションが稼動している AWS リソースの完全なコントロールを維持でき、いつでも基本的なリソースにアクセスすることができます。

Elastic Beanstalk の詳細をご覧ください。

アプリケーションコンテナ

Amazon Elastic Container Service (Amazon ECS)

Amazon ECS は非常にスケーラブルかつ高性能なコンテナ管理サービスで、Docker コンテナに対応しており、Amazon EC2 インスタンスのマネージド型クラスターでアプリケーションを簡単に実行できます。Amazon ECS により、クラスター管理インフラストラクチャのインストール、運用、スケールを行う必要がなくなります。簡単な API コールを使用して、Docker 対応のアプリケーションを起動/停止したり、クラスターの詳細な状態を問い合わせたり、多くの使い慣れた機能 (セキュリティグループ、Elastic Load Balancing、Amazon EBS ボリューム、 IAM ロールなど) にアクセスしたりできます。Amazon ECS を使用することで、リソースニーズと可用性要件に基づいて、クラスター全体のコンテナの配置をスケジューリングできます。また、ビジネスやアプリケーションに固有のニーズに合わせた独自のスケジューラやサードパーティー製スケジューラを統合することもできます。

Amazon ECS の詳細をご覧ください。

セキュリティとユーザーサインイン

AWS Identity and Access Management (IAM)

IAM を利用すると、AWS のサービスおよびリソースに対するお客様のユーザーのアクセスを安全にコントロールすることができます。IAM を使用すると、AWS のユーザーとグループを作成および管理し、アクセス権を使用して AWS リソースへのアクセスを許可および拒否できます。

IAM の詳細をご覧ください。

Amazon Cognito ユーザープール

Amazon Cognito でモバイルアプリやウェブアプリに、ユーザーのサインアップやサインインを簡単に追加できます。Amazon Cognito を使用すると、Facebook、Twitter、Amazon などのソーシャル ID プロバイダ経由で、SAML ID ソリューションを使用して、または独自の ID システムを使用してユーザーを認証することもできます。さらに、Amazon Cognito では、ユーザーのデバイスにローカルでデータを保存し、デバイスがオフラインであってもアプリケーションが機能するようにもできます。その後、ユーザーのデバイス間でデータを同期して、使用するデバイスを問わずアプリのエクスペリエンスに整合性を持たせることができます。

Amazon Cognito を使用すると、ユーザーの管理、認証、デバイス間の同期を行うソリューションの構築、安全性の確保、スケーリングに煩わされることなく、優れたアプリのエクスペリエンスを作成することに集中できます。

Amazon Cognito の詳細をご覧ください。

ソース管理とアプリケーションライフサイクル管理

AWS CodeCommit

AWS CodeCommit は、安全で高度にスケーラブルなプライベート Git リポジトリを簡単にホストできるフルマネージド型のソース管理サービスです。 AWS CodeCommit は、独自のソース管理システムを運用する必要性や、インフラストラクチャのスケーリングについて懸念を排除します。を使用して AWS CodeCommit 、ソースコードからバイナリまですべてを安全に保存でき、既存の Git ツールとシームレスに連携します。

AWS CodeCommit の詳細は、こちらを参照してください。

キューとメッセージング

Amazon SQS

Amazon Simple Queue Service (Amazon SQS) は、高速で、信頼性が高く、スケーラブルな、完全マネージド型のメッセージキューイングサービスです。Amazon SQS を利用すると、簡単かつコスト効率良く、クラウドアプリケーションのコンポーネントを切り離すことができます。Amazon SQS を使用すると、メッセージを失うことなく、どのような量のデータでも転送できます。他のサービスが常に利用可能である必要はありません。Amazon SQS には、高スループットと at-least-once 処理のスタンダードキュー、およびFIFO (先入れ先出し) 配信と正確に 1 回限りの処理を提供するFIFOキューが含まれています。

Amazon SQS を使用すると、可用性の高いメッセージングクラスターの運用とスケーリングという管理上の負担を軽減しつつ、使用した分だけ低額の料金を支払うことができます。

Amazon SQS の詳細をご覧ください。

Amazon SNS

Amazon Simple Notification Service (Amazon SNS) は、高速かつ柔軟な完全マネージド型のプッシュ通知サービスです。このサービスを使用すると、個々のメッセージを送信したり、多数の受信者にメッセージをファンアウトしたりできます。Amazon SNS により、簡単かつコスト効率の高い方法で、モバイルデバイスユーザーやメール受信者にプッシュ通知を送信したり、他の分散型サービスにメッセージを送信したりもできます。

Amazon SNS を使用すれば、Apple Push Notification Service (APNS)、Google クラウドメッセージング (GCM)、Fire OS、Windows の各デバイス、さらには中国では Baidu Cloud Push を使用して、Android デバイスにも通知を送信できます。Amazon SNS を使用して、世界中のモバイルデバイスユーザーに SMS メッセージを送信できます。

Amazon SNS では、これらのエンドポイント以外にも、Amazon SQS、 AWS Lambda 関数、または任意の HTTP エンドポイントにメッセージを配信できます。

Amazon SNS の詳細をご覧ください。

Amazon SES

Amazon Simple Email Service (Amazon SES) は、Amazon.com が自社の顧客ベースを対象に開発した信頼性の高いスケーラブルなインフラストラクチャを基盤とする費用対効果の高い E メールサービスです。Amazon SES を使用すると、最低料金が適用されることなく、E メールを送受信できます。使用したときに使用した分のみのお支払いとなります。

Amazon SES の詳細をご覧ください。

ワークフロー

Amazon Simple Workflow Service (Amazon SWF)

Amazon SWF は、開発者が並列またはシーケンシャルステップを含むバックグラウンドジョブを構築、実行、拡張するのを支援します。Amazon SWF は、クラウド内の完全マネージド型ステータストラッカーやタスクコーディネーターと考えることができます。

アプリケーションのステップが完了するまでに 500 ミリ秒以上かかる場合は、処理の状態を追跡する必要があります。タスクが失敗した場合は、復旧または再試行する必要があります。Amazon SWF はそのお手伝いをします。

Amazon SWF の詳細をご覧ください。

ストリーミングアプリケーション

Amazon AppStream

Amazon AppStream では、Windows アプリケーションを任意のデバイスに配信できます。

Amazon AppStream では、既存の Windows アプリケーションをクラウドからストリーミングできるため、コードを変更することなく、より多くのデバイスでより多くのユーザーにアクセスできます。Amazon では AppStream、アプリケーションが AWS インフラストラクチャにデプロイおよびレンダリングされ、出力はパーソナルコンピュータ、タブレット、携帯電話などの市場投入デバイスにストリーミングされます。アプリケーションはクラウドで実行されるため、お客様が使用するデバイスと関係なく、処理とストレージの膨大なニーズに応じてスケールできます。Amazon は、クラウドからアプリケーションをストリーミングするための SDK AppStream を提供します。独自のカスタムクライアント、サブスクリプション、アイデンティティ、ストレージソリューションを Amazon と統合 AppStream して、ビジネスニーズに合ったカスタムストリーミングソリューションを構築できます。

Amazon の詳細をご覧ください AppStream。

AWS CloudFormation で Lightsail リソースを作成する

Amazon Lightsail は、リソースとインフラストラクチャの作成と管理の所要時間を短縮できるように AWS リソースをモデル化して設定するためのサービスである AWS CloudFormation と統合されてい

ます。必要なすべての AWS リソース (インスタンスやディスクなど) を説明するテンプレートを作成すると、AWS CloudFormation がユーザーに代わってこれらのリソースのセットアップや構成を行います。

AWS CloudFormation を使用すると、テンプレートを再利用して Lightsail リソースを同じように繰り返してセットアップできます。リソースを一度記述するだけで、同じリソースを複数の AWS アカウント とリージョンで何度でもプロビジョニングできます。

Lightsail および AWS CloudFormation のテンプレート

Lightsail および関連サービスのリソースをプロビジョニングして設定するには、AWS CloudFormation テンプレートについて理解しておく必要があります。テンプレートは、JSON または YAML でフォーマットされたテキストファイルです。これらのテンプレートには、AWS CloudFormation スタックにプロビジョニングしたいリソースを記述します。JSON や YAML に不慣れな方は、AWS CloudFormation Designer を使えば、AWS CloudFormation テンプレートを使いこなすことができます。詳細については、「AWS CloudFormation ユーザーガイド」の「AWS CloudFormation Designer とは」を参照してください。

Lightsail は AWS CloudFormation でのインスタンスとディスクの作成をサポートします。詳細については、「AWS CloudFormation ユーザーガイド」の「Lightsail リソースタイプのリファレンス」を参照してください。

AWS CloudFormation の詳細はこちら

AWS CloudFormation の詳細については、以下のリソースを参照してください。

- AWS CloudFormation
- AWS CloudFormation ユーザーガイド
- AWS CloudFormation API リファレンス
- AWS CloudFormation コマンドラインインターフェイスユーザーガイド

Lightsail 用の AWS CloudFormation スタック

Amazon Lightsail で AWS CloudFormation を使用し、エクスポートしたスナップショットから Amazon Elastic Compute Cloud (Amazon EC2) インスタンスを作成します。Lightsail コンソールまたは Lightsail API を使用して Amazon EC2 インスタンスの作成をリクエストすると、CloudFormation スタックが作成されます。スタックは、Amazon Web Services (AWS) アカウン

トで一連のアクションを実行し、インスタンスに関連するすべてのリソースを作成します。たとえば、Amazon マシンイメージ (AMI) から Amazon EC2 インスタンスを作成し、EBS スナップショットから Elastic Block Store (EBS) システムボリュームを作成して、インスタンスのセキュリティグループを作成します。AWS CloudFormation スタックの詳細については、AWS CloudFormation ドキュメントの「スタックの操作」を参照してください。

AWS CloudFormation スタックにアクセスするには、Lightsail コンソールまたは AWS CloudFormation コンソールを使用できます。このガイドでは両方のアクセス方法を示します。

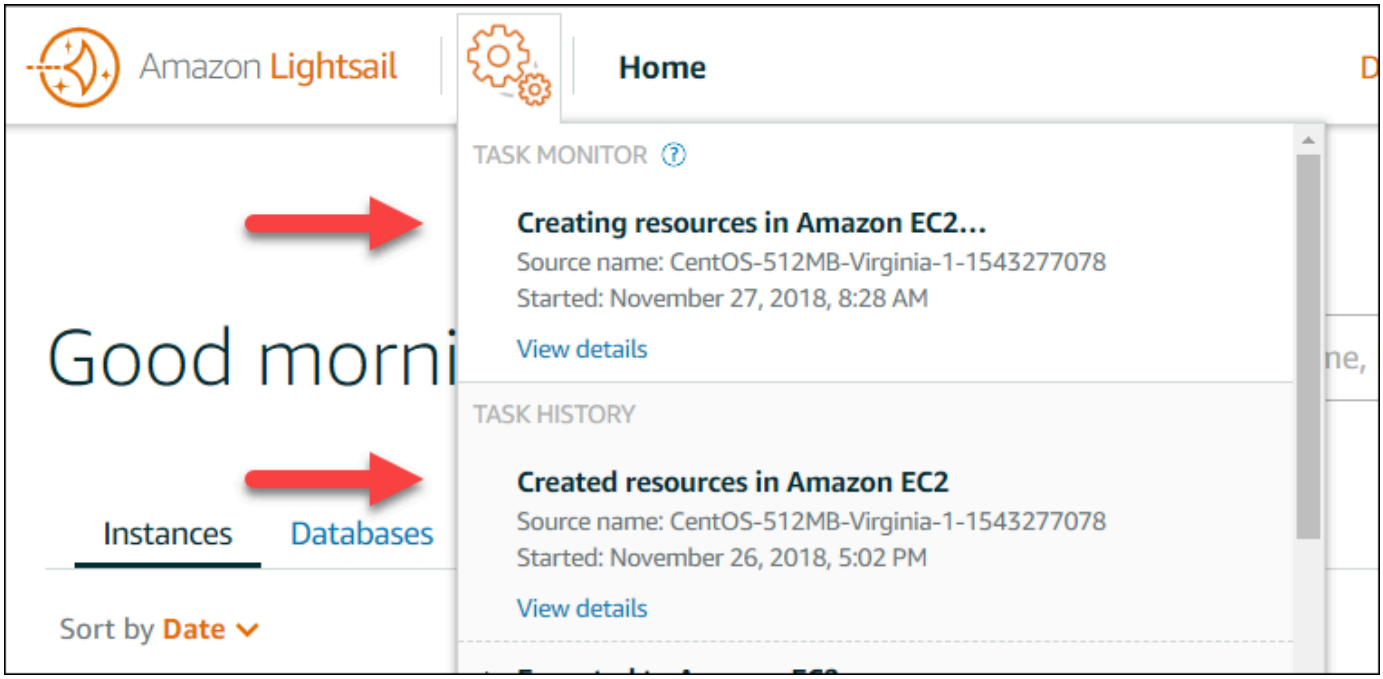
> **Note**
>
> Amazon EC2 リソースの作成に使用した AWS CloudFormation スタックは、Amazon EC2 リソースに永続的にリンクされます。スタックを削除すると、すべての関連リソースが自動的に削除されます。このため、Lightsail で作成した AWS CloudFormation スタックは一切削除しないでください。代わりに、EC2 コンソールを使用して Amazon EC2 リソースを削除します。

Lightsail コンソールから AWS CloudFormation スタックにアクセスする

Lightsail コンソールまたは Lightsail API を使用して Amazon EC2 でインスタンスを作成することを選択すると、AWS CloudFormation スタックが作成され、そのステータスがタスクモニターで追跡されます。タスクモニターの詳細については、「タスクモニター」を参照してください。

AWS CloudFormation スタックを Lightsail コンソールで表示するには

1. Lightsail コンソールにサインインします。
2. 上部のナビゲーションペインでタスクモニターを選択します。
3. Amazon EC2 インスタンスに以前作成された CloudFormation スタックにアクセスするには、Amazon EC2 でのリソースの作成 またはAmazon EC2で作成されたリソース でラベル付けされたタスクの詳細を表示 を選択してください。

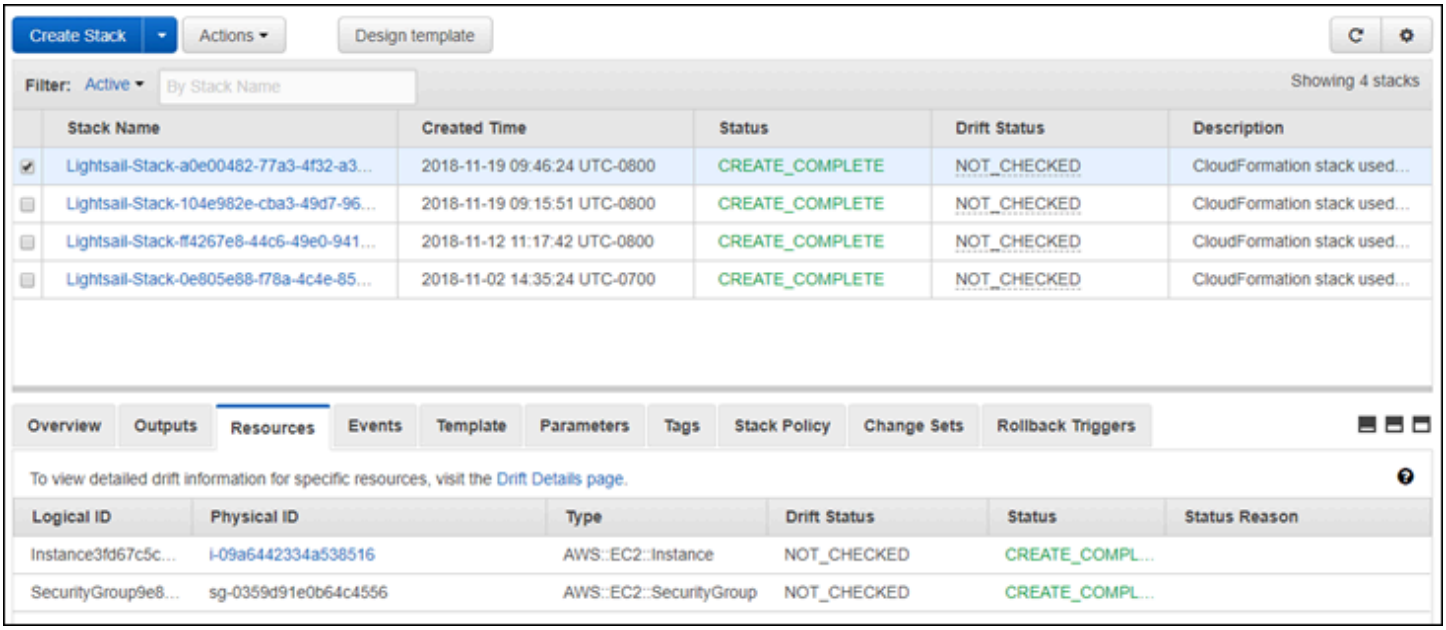


4. 確認ページにタスクの CloudFormation スタックが表示されます。スタック名を選択し、AWS CloudFormation コンソールでスタックの詳細を開きます。

AWS CloudFormation コンソールからスタックにアクセスする

AWS CloudFormation コンソールからスタックの詳細にアクセスすることもできます。Lightsail で作成されたスタックは、次のスクリーンショットに示すように、名前が「Lightsail -stack」で始まり、説明が「CloudFormation stack used to create Amazon EC2 resources」と表示されます。

スタックのステータスが [CREATE_IN_PROGRESS] である場合、エクスポートした Lightsail スナップショットから Amazon EC2 リソースが作成中です。スタックのステータスが [CREATE_COMPLETED] である場合、Amazon EC2 リソースの作成プロセスは完了しています。スタックで作成されたリソースを表示するには、スタック名の横にあるチェックボックスをオンにして、[リソース] タブを選択します。


Create Stack
Actions
Design template
Filter: Active
By Stack Name
Showing 4 stacks
Stack Name
Created Time
Status
Drift Status
Description
Lightsail-Stack-a0e00482-77a3-4f32-a3...
2018-11-19 09:46:24 UTC-0800
CREATE_COMPLETE
NOT_CHECKED
CloudFormation stack used...
Lightsail-Stack-104e982e-cba3-49d7-96...
2018-11-19 09:15:51 UTC-0800
CREATE_COMPLETE
NOT_CHECKED
CloudFormation stack used...
Lightsail-Stack-ff4267e8-44c6-49e0-941...
2018-11-12 11:17:42 UTC-0800
CREATE_COMPLETE
NOT_CHECKED
CloudFormation stack used...
Lightsail-Stack-0e805e88-f78a-4c4e-85...
2018-11-02 14:35:24 UTC-0700
CREATE_COMPLETE
NOT_CHECKED
CloudFormation stack used...
Overview
Outputs
Resources
Events
Template
Parameters
Tags
Stack Policy
Change Sets
Rollback Triggers
To view detailed drift information for specific resources, visit the Drift Details page.
Logical ID
Physical ID
Type
Drift Status
Status
Status Reason
Instance3fd67c5c...
i-09a6442334a538516
AWS::EC2::Instance
NOT_CHECKED
CREATE_COMPL...
SecurityGroup9e8...
sg-0359d91e0b64c4556
AWS::EC2::SecurityGroup
NOT_CHECKED
CREATE_COMPL...

Amazon Lightsail 請求

Amazon Lightsail の請求は、Amazon Web Services (AWS) の請求によって処理されます。Lightsail 請求書を表示するには、AWS Billing and Cost Management ダッシュボードを選択するか、Lightsail コンソールのナビゲーションバーにある [Billing] (請求書) を選択します。料金の詳細については、「Lightsail の料金」を参照してください。

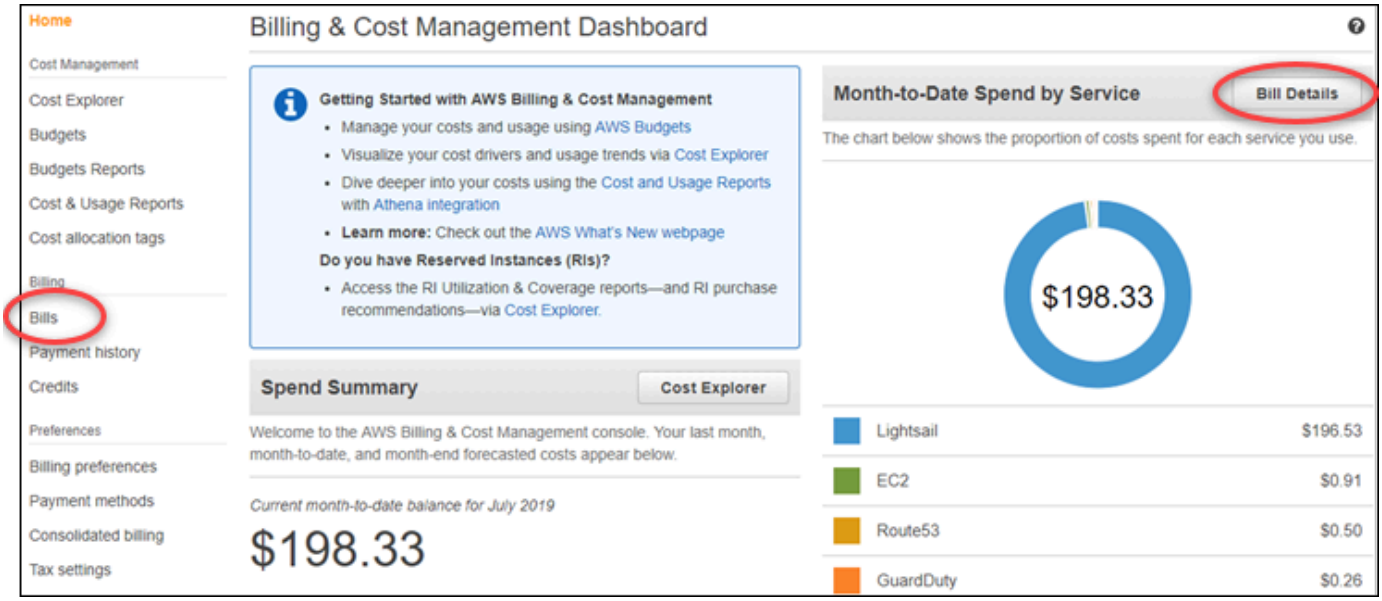
Lightsail の請求明細を表示する

月額の Lightsail の請求の詳細な内訳を表示するには:

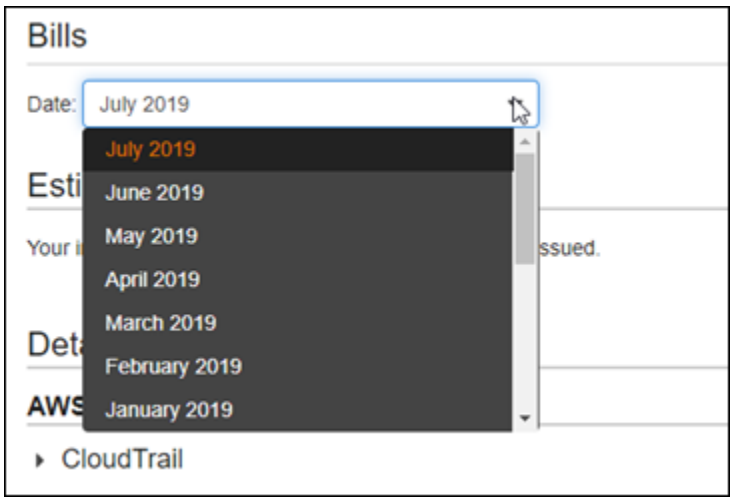
1. AWS Billing and Cost Management ダッシュボードにサインインします。

   請求ダッシュボードのホームページには、請求書の過去 1 か月の内訳が表示されます。

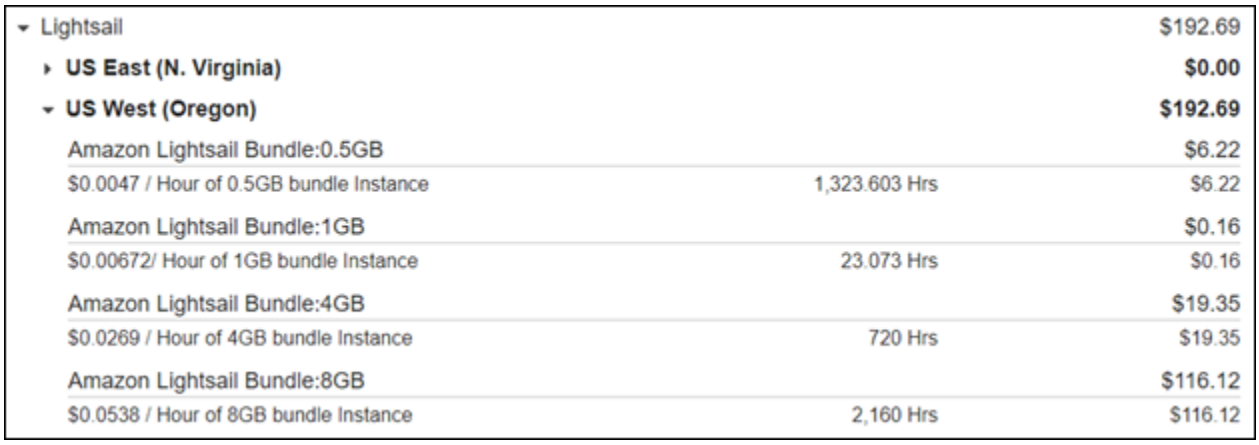
2. 月額の料金の詳細バージョンを表示するには、ダッシュボードのホームページで [料金明細] を選択するか、左側のナビゲーションペインで [請求] を選択します。



3. [Date (日付)] ドロップダウンメニューを選択して、現在の月以外の月を選択します。



4. [請求] ページを下にスクロールし、Lightsail の明細項目を展開して、各リージョンの詳細な使用状況を表示します。



請求の使用タイプ

次のリストは、Lightsail の請求および使用状況レポートに表示される使用タイプを示します。これらの使用タイプは、Lightsail リソースに対する月額料金を識別するのに役立ちます。

> **Note**
>
> リージョンコードを指定する以下の使用タイプについては、このガイドの「請求のリージョンコード」を参照して、対応する AWS リージョン を識別します。

- Amazon Lightsail Bundle:SizeGB: 使用される Linux または Unix インスタンスプラン (時間単位)。サイズは、使用されるインスタンスプランのメモリ仕様を定義します。たとえば、4 GB のメ

モリが指定されている場合、20 USD/月の Linux または Unix インスタンスプランの請求時間が表示されます。

- Amazon Lightsail Bundle:SizeGB (Windows): 使用された Windows インスタンスプラン (時間単位)。サイズは、使用されるインスタンスプランのメモリ仕様を定義します。たとえば、4 GB のメモリが指定されている場合、40 USD/月の Windows インスタンスプランの請求時間が表示されます。
- Amazon Lightsail RelationalDatabase:SizeGB: 使用される標準データベースプラン (時間単位)。サイズは、使用されるデータベースプランのメモリ仕様を定義します。たとえば、4 GB のメモリが指定されている場合、60 USD/月の標準データベースの請求時間が表示されます。
- Amazon Lightsail RelationalDatabase:SizeGB (high availability): 使用される高可用性データベースプラン (時間単位)。サイズは、使用されるデータベースプランのメモリ仕様を定義します。たとえば、4 GB のメモリが指定されている場合、120 USD/月の高可用性データベースプランの請求時間が表示されます。
- Amazon Lightsail Region-DiskUsage: 使用されるブロックストレージディスクの量 (GB/月)。
- Amazon Lightsail DNS-Queries: 当月の DNS クエリの数 (カウント)。
- Amazon Lightsail Load Balancer: 使用されるロードバランサーの量 (時間単位)。
- Amazon Lightsail Region-SnapshotUsage: 保存されたスナップショットデータの量 (GB/月)。
- Amazon Lightsail Region-UnusedStaticIP: 接続されていない静的 IP の量 (時間単位)。
- Amazon Lightsail Region-TotalDataXfer-In-Bytes: 転送 (in) されたデータの合計量 (GB 単位)。
- Amazon Lightsail Region-TotalDataXfer-Out-Bytes: 転送 (out) されたデータの合計量 (GB 単位)。
- Amazon Lightsail Region-DataXfer-Out-Overage-Bytes: 使用されたインスタンスまたはデータベースプランの許容範囲を超えてインターネットまたはパブリック IP に転送されたデータの量 (GB 単位)。
- Amazon Lightsail Region-DataXfer-Out-Free-Bytes (廃止): 使用されたインスタンスまたはデータベースプランの許容範囲内にある転送されたデータの量 (GB 単位)。
- Amazon Lightsail Region-DataXfer-Out-Other-Bytes (廃止): 使用されているインスタンスまたはデータベースプランの許容範囲を超えるプライベート IP に転送されたデータの量 (GB 単位)。プライベート IP を介した AWS リソースへの転送の場合、この超過分は無料です。

請求のリージョンコード

Lightsail 請求および使用状況レポートはコードおよび略名を使用します。たとえば、使用タイプの場合、リージョンは次の略語のいずれかに置き換えられます。

- APN1: アジアパシフィック (東京) (ap-northeast-1)
- APN2: アジアパシフィック (ソウル) (ap-northeast-2)
- APS1: アジアパシフィック (シンガポール)(ap-southeast-1)
- APS2: アジアパシフィック (シドニー)(ap-southeast-2)
- APS3: アジアパシフィック (ムンバイ)(ap-south-1)
- CAN1: カナダ (中部)(ca-central-1)
- EU: 欧州 (アイルランド)(eu-west-1)
- EUC1: 欧州 (フランクフルト)(eu-central-1)
- EUW2: 欧州 (ロンドン)(eu-west-2)
- EUW3: 欧州 (パリ) (eu-west-3)
- EUN1: 欧州 (ストックホルム) (eu-north-1)
- USE1: 米国東部 (バージニア北部) (us-east-1)
- USE2: 米国東部 (オハイオ)(us-east-2)
- USW2: 米国西部 (オレゴン)(us-west-2)

Lightsail に関するよくある質問

このトピックでは、よくある質問 (FAQ) に回答しています。ここに回答のない質問については、ページの下部にあるご質問は? コメント?リンクから [フィードバックを送る] ボタンをご利用ください Lightsail ディスカッションフォーラムに質問を投稿することもできます。

目次

全般

Amazon Lightsail とは何ですか?

Amazon Lightsail は、 AWS ウェブサイトやウェブアプリケーションをクラウドで構築してホストするためのソリューションを必要とする開発者、中小企業、学生、その他のユーザーに

とって、最も簡単に使い始める方法です。Lightsail は、開発者にコンピューティング、ストレージ、ネットワークの機能を提供します。Lightsail には、仮想マシン、コンテナ、データベース、CDN、ロードバランサー、DNS 管理など、プロジェクトを迅速に開始するために必要なすべてのものが、低価格で予測可能な月額料金で含まれています。

Lightsail では何ができますか?

アプリケーションを簡単にデプロイして管理するためのすべてを含む事前設定された仮想プライベートサーバー (インスタンス) を作成したり、基盤となるインフラストラクチャーとオペレーティングシステムのセキュリティと状態を Lightsail が管理するデータベースを作成したりできます。Lightsail は、必要なインスタンスが数十個以下のプロジェクトや、シンプルな管理インターフェースを好む開発者に最適です。Lightsail の一般的なユースケースには、ウェブサイト、ウェブアプリケーション、ビジネスソフトウェア、ブログ、e コマースサイトなどの運営が含まれます。プロジェクトが拡大するにつれて、ロードバランサーと接続されたブロックストレージをインスタンスで使用して冗長性と稼働時間を増やしたり、他の多数のサービスにアクセスして新しい機能を追加したりできます。 AWS

Lightsail は API を提供していますか?

はい。Lightsail コンソールで行うことはすべて、公開されている API によって支えられています。Lightsail CLI と API をインストールして使用する方法を学びましょう。

Lightsail にサインアップするにはどうすればいいですか ?

Lightsail の使用を開始するには、[はじめに] を選択してログインします。Amazon Web Services アカウントを使用して Lightsail にアクセスします。まだアカウントをお持ちでない場合は、アカウントを作成するように求められます。

Lightsail AWS リージョンはどのシステムで利用できますか?

Lightsail は現在、 AWS リージョン以下のすべてのアベイラビリティーゾーンでご利用いただけます。

- 米国東部 (オハイオ) (us-east-2)
- 米国東部 (バージニア北部) (us-east-1)
- 米国西部 (オレゴン) (us-west-2)
- アジアパシフィック (ムンバイ) (ap-south-1)
- アジアパシフィック (ソウル) (ap-northeast-2)

- アジアパシフィック (シンガポール) (ap-southeast-1)
- アジアパシフィック (シドニー) (ap-southeast-2)
- アジアパシフィック (東京) (ap-northeast-1)
- カナダ (中部) (ca-central-1)
- 欧州 (フランクフルト) (eu-central-1)
- 欧州 (アイルランド) (eu-west-1)
- 欧州 (ロンドン) (eu-west-2)
- 欧州 (パリ) (eu-west-3)

欧州 (ストックホルム) eu-north-1

詳細については、「Lightsail のアベイラビリティーゾーン」を参照してくださいAWS リージョン。

アベイラビリティーゾーンとは何ですか?

アベイラビリティーゾーンは、物理的独自性を持った独立したインフラストラクチャで実行されるデータセンターの集合体で、高度な信頼性を実現できるよう設計されています。発電機や冷却装置などの一般的な障害発生点は、アベイラビリティーゾーン間では共有されていません。加えて、アベイラビリティーゾーンは物理的に離れているため、火災や竜巻、洪水などの極めてまれな災害は、単独のアベイラビリティーゾーンにしか影響しません。

Lightsail のサービスクォータにはどのようなものがありますか?

どのクォータを増やすことができるかなど、最新の Lightsail サービスクォータについては、の Lightsail サービスクォータをご覧ください。AWS 全般のリファレンスクォータを増やす必要がある場合は、AWS Support でケースを開いてください。

より詳細なヘルプを得るにはどうすればよいですか?

ご安心ください。Lightsail の状況依存ヘルプパネルには、コンソールでのアクションに関する役立つヒントがすぐに表示されます。Lightsail コンソールから、入門ガイド、概要、ハウツートピックのライブラリにアクセスすることもできます。また、Lightsail API を使用する場合や、Lightsail には AWS CLI、サポートされているすべてのプログラミング言語に関する完全な API リファレンスが用意されています。Lightsail サポートリソースを使用することもできます。

アカウントや請求に関して問題がある場合は、 AWS Support までオンラインでお問い合わせください。Lightsail アカウントでは、24 時間 365 日無料でアクセスできます。

Lightsail の使用方法に関する一般的な質問がある場合は、Lightsail のドキュメントとサポートフォーラムを検索してください。

さらに、 AWS Support では、個々のニーズを満たすさまざまな有料プランを提供しています。

インスタンス

Lightsail インスタンスとは何ですか?

Lightsail インスタンスはクラウドに存在する仮想プライベートサーバー (VPS) です。 AWS Lightsail インスタンスを使用して、データを保存し、コードを実行し、ウェブベースのアプリケーションやウェブサイトを構築します。インスタンスは、パブリック (インターネット) ネットワークとプライベート (VPC) ネットワークの両方を介して相互に接続したり、 AWS 他のリソースに接続したりできます。Lightsail コンソールから直接、インスタンスを簡単に作成、管理、接続できます。

Lightsail プランとは何ですか?

バンドルとも呼ばれる Lightsail プランには、固定量のメモリ (RAM) とコンピューティング (vCPUs) を備えた仮想サーバー、SSD ベースのストレージ (ディスク)、および無料のデータ転送許容量が含まれます。Lightsail プランでは、静的 IPv4 アドレスと DNS 管理も提供されます。Lightsail プランは時間単位のオンデマンドで課金されるため、プランの使用時にのみ料金が発生します。

インスタンスでは何のソフトウェアが実行できますか?

Lightsail には、新しい Lightsail インスタンスを作成すると自動的にインストールされるさまざまなオペレーティングシステムとアプリケーションテンプレートが用意されています。アプリケーションテンプレートには WordPress、 WordPress マルチサイト、cPanel & WHM、Django、Drupal、Ghost PrestaShop、Joomla! などがあります。 、Magento、Redmine、LAMP、Nginx (LEMP)、MEAN、Node.js。

ブラウザ内の SSH または独自の SSH クライアントを使用して、インスタンスに追加のソフトウェアをインストールすることが可能です。

Amazon Lightsail ではどのオペレーティングシステムを使用できますか?

Lightsail は現在、 AlmaLinux OS 9、Amazon Linux 2、Amazon Linux 2023、CentOS、Debian、FreeBSD、openSUSE、Ubuntu の 7 つの Linux または UNIX ライクな

ディストリビューションと、2016 年、2019 年、2022 年の 3 つの Windows Server バージョンをサポートしています。

Lightsail インスタンスを使用するには自分のライセンスを持参する必要がありますか?

Lightsail で利用できるすべてのインスタンスブループリントには、cPanel と WHM ブループリントを除くライセンスが含まれています。そのブループリントには 15 日間の試用ライセンスが含まれています。詳細については、「クイックスタートガイド:Amazon Lightsail の cPanel と WHM」を参照してください。他のすべてのインスタンスブループリントでは、Bring-Your-Own-License (BYOL) は必要ありません。

Lightsail インスタンスを作成するにはどうすればよいですか?

Lightsail にログインすると、Lightsail コンソール、コマンドラインインターフェイス (CLI)、または API を使用してインスタンスを作成および管理できます。

コンソールへの初回ログインの際に、インスタンスの作成を選択します。インスタンスの作成ページでは、ソフトウェア、ロケーション、およびインスタンスの名前が選択できます。作成 を選択すると、数分以内に新しいインスタンスが自動的にスピンアップします。

Lightsail インスタンスはどのように機能しますか?

Lightsail インスタンスは、ウェブサーバー、開発者環境、小規模データベースのユースケース向けに特別に設計されています。 AWS このようなワークロードは常時または一貫して CPU をフルに使用するわけではありませんが、パフォーマンスのバーストが必要な場合があります。Lightsail は、ベースラインレベルの CPU パフォーマンスを提供し、さらにベースラインを超えるバースト機能を提供するバースト可能なパフォーマンスインスタンスを使用します。この設計により、必要なときに必要なパフォーマンスを得ることが可能です。その一方で、他環境のオーバーサブスクリプションによって引き起こされれがちなパフォーマンスの変動やそのほかの副作用からユーザーを保護します。

動画エンコーディングや HPC アプリケーションなどのアプリケーションのため、一貫して高い CPU パフォーマンスを有する高度な環境設定とインスタンスが必要な場合は、Amazon EC2 を使用することをお勧めします。

インスタンスがバーストしているかどうか、どうやって確認できますか?

インスタンスの CPU 使用率メトリクスグラフに、持続可能領域とバースト領域が表示されます。Lightsail インスタンスは、システムの動作に影響を与えずに、持続可能ゾーンで無期限に運用できます。負荷が高い場合、インスタンスはバースト領域で作動し始める可能性があります。バースト領域で作動している間、インスタンスは大量の CPU サイクルを消費しています。した

がって、この領域では限られた期間しか作動できません。詳細については、「Amazon Lightsail でのインスタンスメトリクスの表示」を参照してください。

メトリクスアラームを追加して、インスタンスの CPU 使用率が持続可能領域からバースト領域に移動した際に通知を受けます。詳細については、「Amazon Lightsail でのインスタンスメトリックスアラームの作成」を参照してください。

Lightsail インスタンスに接続する方法を教えてください。

Lightsail では、ブラウザからインスタンスのターミナルにワンクリックで安全に接続できます。Linux/UNIX ベースのインスタンスには SSH アクセスを、Windows ベースのインスタンスには RDP アクセスをサポートしています。ワンクリック接続を利用するには、インスタンス管理画面を起動し、SSH を使用して接続 または RDP を使用して接続 を選択します。新しいブラウザウィンドウが開き、インスタンスに自動接続されます。

独自のクライアントを使用して Linux/UNIX ベースのインスタンスに接続したい場合は、Lightsail が SSH キーの保存と管理作業を代行し、SSH クライアントで使用できる安全なキーを提供します。

インスタンスをバックアップするには、どうすればよいですか?

データをバックアップする場合は、Lightsail コンソールまたは API を使用してインスタンスの手動スナップショットを作成するか、自動スナップショットを有効にして Lightsail に毎日のスナップショットを作成させることができます。障害やコードのデプロイに不具合が発生した場合は、インスタンスのスナップショットを後から使用して、新しいインスタンスを作成することができます。詳細については、「スナップショット」を参照してください。

プランをアップグレードできますか?

はい。インスタンスのスナップショットを使用して、より大きいサイズの新しいインスタンスを作成できます。詳細については、「スナップショット」を参照してください。

Lightsail インスタンスをアカウント内の他のリソースに接続する方法を教えてください。 AWS

VPC ピアリングを使用して、Lightsail AWS インスタンスをアカウントの Amazon VPC リソースにプライベートに接続できます。Lightsail アカウントページで [VPC ピアリングを有効化] を選択するだけで、Lightsail が自動的に作業を行います。VPC ピアリングを有効にすると、デフォルト Amazon VPC AWS 内の他のリソースのプライベート IP を使用してアドレス指定できます。こちらから手順を確認いただけます。

> **Note**
>
> Lightsail との VPC ピアリングを機能させるには、 AWS アカウントにデフォルトの Amazon VPC を設定する必要があることに注意してください。 AWS 2013 年 12 月以前に作成されたアカウントにはデフォルト VPC がないため、デフォルト VPC を設定する必要があります。デフォルト VPC のセットアップに関する詳細は、こちら。

インスタンスの停止と削除の違いは何ですか?

インスタンスを停止すると、現在の状態で電源がオフになり、いつでも再開することができます。インスタンスを停止すると、そのパブリック IPv4 は解放されるため、停止して再開した後も同じIP を保持する必要のあるインスタンスには、静的 IP を使用することをお勧めします。インスタンスにアタッチされているパブリック IPv6 アドレスは、インスタンスを停止して再開しても変更されないので注意してください。

インスタンスの削除は、破壊的なアクションです。インスタンスのスナップショットを作成していない限り、すべてのインスタンスデータが失われ、復元できなくなります。自動スナップショットも、手動スナップショットとしてコピーして保持しない限り、インスタンスと共に削除されます。インスタンスのパブリック IP とプライベート IP アドレスも解放されます。インスタンスで静的 IPv4 アドレス を使用していた場合、その静的 IP アドレス はデタッチされますが、アカウントには残ります。

オブジェクトストレージとバケット

lightsail オブジェクトストレージでどのようなことができますか?

ユーザーは画像、動画、HTML ファイルなどの静的コンテンツを Lightsail オブジェクトストレージサービスのバケットに保存することができます。バケットに保存されているオブジェクトは、ウェブサイトやアプリケーションで使用できます。Lightsail オブジェクトストレージは数回のクリックだけでLightsail CDN ディストリビューションに関連付けることができます。これにより、世界中のオーディエンスにコンテンツをすばやく簡単に配信できます。また、低コストで安全なバックアップ ソリューションとしても使用できます。詳細については、「オブジェクトストレージ」を参照してください。

lightsail オブジェクトストレージの料金を教えてください。

Lightsail オブジェクトストレージには、Lightsail AWS リージョンが利用可能なすべてのパッケージに 3 種類の固定価格バンドルがあります。最初のバンドルは 1 USD/月で最初の12カ月間は無

料です。このバンドルには 5 GB のストレージ容量と 25 GB 分のデータ転送が含まれます。2 つ目のバンドルは毎月 3 USD で、100 GB のストレージ容量と 250 GB 分のデータ転送が含まれています。最後に、3 番目のバンドルは毎月 5 USD で、250 GB のストレージ容量と 500 GB 分のデータ転送が含まれています。Lightsail オブジェクトストレージには、バケットへの無制限のデータ転送が含まれます。バンドルされたデータ転送許容値は、バケットからのデータ転送にのみ使用されます。

Lightsail オブジェクトストレージには超過料金がかかりますか?

個々のバケットに選択されたストレージプランの月間ストレージ容量またはデータ転送許容量を超えると、追加の容量に対する料金が請求されます。詳細については、Lightsail の料金 ページを参照してください。

データ転送許容値はオブジェクトストーレッジでどのように機能しますか?

Lightsail オブジェクトストレージへのデータの入出転送によって、データ転送許容値を消費します。ただし、以下の項目は除きます。

- インターネットから Lightsail オブジェクトストレージに転送されたデータ
- Lightsail オブジェクトストレージリソース間でのデータ移動
- Lightsail オブジェクトストレージから同一の別の Lightsail リソース AWS リージョン (AWS 異なるアカウントにあるが同じリソースを含む) に転送されたデータ AWS リージョン
- Lightsail オブジェクトストレージから Lightsail CDN ディストリビューションに転送されたデータ

Lightsailバケットに関連するプランの変更はできますか?

はい。個々の Lightsail バケットのストレージプランは、 AWS 毎月の請求サイクル内に 1 回変更できます。

Lightsail オブジェクトストレージから Amazon S3 にオブジェクトをコピーできますか?

はい。Lightsail オブジェクトストレージから Amazon S3 へのコピーはサポートされています。詳細については、「AWS Premium Support ナレッジセンター」の「 Amazon S3 バケットから別のバケットにすべてのオブジェクトをコピーするにはどうすればよいですか?」を参照してください。

Lightsail オブジェクトストレージの使用を開始するには、どうすればよいですか?

Lightsail オブジェクトストレージを使用するには、データを保存するために使用するバケットをまず作成する必要があります。詳細については、「バケットの作成」を参照してください。バ

ケットが起動し作動し始めた後、バケットへのオブジェクトの追加が開始できます。Lightsail コンソールを使用してファイルをアップロードするか、ログやそのほかのアプリケーションデータなどのコンテンツをバケットに入れるようアプリケーションを設定します。または、 AWS Command Line Interface ()AWS CLIを使用して Lightsail オブジェクトストレージを使い始めることもできます。

バケットにオブジェクトをアップロードするにはどうすればよいですか?

画像やそのほかの静的ファイルなどのオブジェクトをバケットにアップロードする場合、トップナビゲーションタブ [Objects] (オブジェクト) から [Upload] (アップロード) を選択し、コンピュータから正しいファイルまたはディレクトリを選択します。または、デスクトップから Lightsail オブジェクトストレージコンソール内のマークされた領域にファイルやディレクトリをドラッグアンドドロップします。

バケットへのパブリックアクセスをブロックできますか?

Lightsail バケットとオブジェクトは、デフォルトでプライベートに設定されています。つまり、適切な権限を持つユーザーのみがバケットとオブジェクトにアクセスできます。ユーザーは、このデフォルト設定を変更し、個々のオブジェクトをプライベートバケットで公開して読み取り専用にするか、バケット全体を公開して読み取り専用にするかを選択できます。ユーザーがバケットまたはオブジェクトを公開すると、世界中のすべての人がそのコンテンツを読むことができます。詳細については、「バケットのアクセス許可」を参照してください。

バケットにプログラムにようるアクセスを追加するにはどうすればよいですか?

バケットへのプログラムによるアクセスは、アクセスキーまたはロールのいずれかを使用します。まず、プログラムで接続したいLightsail コンソールのバケットを選択します。次に、[権限] タブで Lightsail インスタンスにアクセスキーを作成するか、ロールを割り当ててから、バケットを使用するようにウェブサイトまたはアプリケーションコードを設定します。ウェブサイトまたはアプリケーションでオブジェクトストレージをどのように使うかの用途に応じて、この動作は異なる場合があります。詳細については、「バケットのアクセス許可」を参照してください。

バケットを他のアカウントと共有する方法を教えてください。 AWS

Lightsail では、 AWS バケット管理ページのクロスアカウントアクセスセクションで指定したアカウント ID でバケットへのアクセスを共有できるため、クロスアカウント共有が容易になります。 AWS アカウント ID を指定すると、そのアカウントにはバケットへの読み取り専用アクセス権が付与されます。詳細については、「バケットのアクセス許可」を参照してください。

バージョニングとは何ですか?

バージョニングは、バケット内の全てのオブジェクトストレージのすべてのバージョンを保存、取得、復元することができます。これは、偶発的な上書きや削除からの更なる保護となります。

詳細については、「バケットのオブジェクトのバージョニングを有効化または一時停止する」を参照してください。

Lightsail バケットを Lightsail CDN ディストリビューションに関連付けるにはどうすればよいですか?

Lightsail オブジェクトストレージは、数回のクリックで Lightsail CDN ディストリビューションに関連付けることができます。これにより、世界中のオーディエンスにコンテンツをすばやく簡単に配信できます。これを行うには、Lightsail CDN ディストリビューションを作成し、Lightsail CDN ディストリビューションのオリジンとして Lightsail バケットを選択します。詳細については、Amazon Lightsail バケットを Lightsail コンテンツ配信ネットワークディストリビューションで使用するを参照してください。

Lightsail オブジェクトストレージサービスにはどのような制限がありますか?

Lightsail オブジェクトストレージサービスでは、アカウントごとに最大 20 のバケットが作成できます。バケットに保存できるオブジェクト数に制限はありません。すべてのオブジェクトを 1 つのバケットに保存したり、複数のバケットに分けて整理することも可能です。

Lightsail オブジェクトストレージはモニタリングとアラートをサポートしていますか?

Lightsail オブジェクトストレージでは、バケット内の合計使用容量とバケット内のオブジェクト数に関するメトリクスを簡単に表示できます。これらのメトリクスに基づいたアラートもサポートされています。詳細については、「Amazon Lightsail でバケットのメトリックスを表示する」と「バケットメトリクスアラームを作成する」を参照してください。

コンテナサービス

Lightsail コンテナサービスでは何ができますか?

Lightsail コンテナサービスは、コンテナ化されたアプリケーションをクラウドで簡単に実行する方法を提供します。コンテナサービスでは、シンプルなウェブアプリから多階層のマイクロサービスまで、さまざまなアプリケーションを実行できます。ユーザーが行うのは、コンテナサービスに必要なコンテナイメージ、パワー (CPU、RAM)、スケール (ノード数) を指定することだけです。Lightsail は、基盤となるインフラストラクチャーを管理しなくてもコンテナサービスの実行を引き受けます。Lightsail は、コンテナサービスで実行されているアプリケーションにアクセスするための、負荷分散された TLS エンドポイントを提供します。

Lightsail コンテナサービスは Docker コンテナを実行できますか?

はい。Lightsail は Linux ベースの Docker コンテナをサポートしています。Windows コンテナは現在サポートされていません。

Lightsail コンテナサービスでパブリックコンテナイメージを使用するにはどうすればよいですか?

Amazon ECR Public Registry などのオンラインパブリックレジストリーのコンテナイメージを使用することも、独自のカスタムイメージを作成して Lightsail にプッシュすることもできます。いくつかの簡単な手順は、です。 AWS CLI詳しくは、「コンテナイメージをプッシュして管理する」を参照してください。

プライベートコンテナレジストリからコンテナイメージをプルできますか?

現在、Lightsail コンテナサービスではパブリックコンテナレジストリのみがサポートされています。または、カスタムコンテナイメージをローカルマシンから Lightsail にプッシュして非公開にすることもできます。

需要に応じてサービスのパワーとスケールを変更することはできますか?

はい。コンテナサービスのパワーとスケールは、サービスの作成後であっても常時変更できます。

Lightsail コンテナサービスによって作成された HTTPS エンドポイントの名前をカスタマイズできますか?

Lightsail は、すべてのコンテナサービスに HTTPS エンドポイントをこの形式で提供します。*<service-name>*.*<random-guid>*.*<aws-region-name>*.cs.amazonlightsail.comカスタマイズできるのは、サービス名だけです。代案として、カスタムドメイン名を使用することができます。詳細については、「カスタムドメインの有効化と管理」を参照してください。

Lightsail コンテナサービスの HTTPS エンドポイントにカスタムドメインを使用できますか?

はい。カスタムドメイン名の SSL/TLS 証明書を作成して Lightsail のコンテナサービスに添付できます。証明書はドメイン検証済みである必要があります。ドメインの DNS が Lightsail DNS ゾーンを使用している場合は、ドメイン (example.com) またはサブドメイン () のトラフィックをコンテナサービスにルーティングできます。www.example.com別の方法として、ALIAS レコードの追加をサポートしている DNS ホスティングプロバイダーを使用して、ドメイン (example.com) の Apex を Lightsail コンテナサービスのデフォルトドメイン (パブリック DNS) にマッピングすることもできます。詳細については、「カスタムドメインの有効化と管理」を参照してください。

Lightsail コンテナサービスの費用はいくらですか?

Lightsail コンテナサービスはオンデマンドの時間単位で請求されるため、お支払いいただくのは使用した分のみです。お客様が使用する Lightsail コンテナサービスごとに、毎月の最大サービス

料金を上限として、時間単位の固定料金を請求します。月間最大サービス料金は、サービスのパワーの基本料金にサービスのスケールを掛けることによって算出できます。例えば、マイクロパワーとスケールが 2 のサービスの場合、最大 10 USD*2=20 USD/月のコストがかかります。最も安価な Lightsail コンテナサービスは、1 時間あたり 0.0094 米ドル (1 か月あたり 7 米ドル) から始まります。各サービスについて、月間 500 GB の無料クォータを超える使用については、追加データ転送料金が請求される場合があります。

コンテナサービスを数日しか実行なくても、1 か月分が請求されますか?

Lightsail コンテナサービスは、実行中または無効状態の場合にのみ課金されます。月末までに Lightsail コンテナサービスを削除した場合、Lightsail コンテナサービスを使用した合計時間数に基づいて日割り計算された料金が請求されます。たとえば、パワーがマイクロ、スケールが 1 の Lightsail コンテナサービスを 1 か月に 100 時間使用した場合、1.34 ドル (0.0134*100 ドル) が課金されます。

コンテナサービスとのデータ転送は課金されますか?

すべてのコンテナサービスには、データ転送クォータ (月々 500 GB) が付属しています。これはサービスへの流入と流出 両方のデータ転送でカウントされます。クォータを超えると、Lightsail コンテナサービスからインターネット、 AWS リージョン 別のサービス、またはパブリック IP AWS アドレスを使用する場合と同じリージョンのリソースへのデータ転送 OUT に対して課金されます。無料容量分を超過した際のこれらのデータ転送料金は、以下の通りです。

- 米国東部 (オハイオ) (us-east-2): 0.09 USD/GB
- 米国東部 (バージニア北部) (us-east-1): 0.09 USD/GB
- 米国西部 (オレゴン) (us-west-2): 0.09 USD/GB
- アジアパシフィック (ムンバイ) (ap-south-1): 0.13 USD/GB
- アジアパシフィック (ソウル) (ap-northeast-2): 0.13 USD/GB
- アジアパシフィック (シンガポール) (ap-southeast-1): 0.12 USD/GB
- アジアパシフィック (シドニー) (ap-southeast-2): 0.17 USD/GB
- アジアパシフィック (東京) (ap-northeast-1): 0.14 USD/GB
- カナダ (中部) (ca-central-1): 0.09 USD/GB
- 欧州 (フランクフルト) (eu-central-1): 0.09 USD/GB

- 欧州 (アイルランド) (eu-west-1): 0.09 USD/GB
- 欧州 (ロンドン) (eu-west-2): 0.09 USD/GB
- 欧州 (パリ) (eu-west-3): 0.09 USD/GB
- 欧州 (ストックホルム) (eu-north-1): 0.09 USD/GB

コンテナサービスの停止と削除の違いは何ですか?

コンテナサービスを無効にすると、コンテナノードは無効状態となり、サービスのパブリックエンドポイントは HTTP ステータスコード「503」を出力します。サービスを有効にすると、最後にアクティブだったデプロイが復元されます。パワーとスケールの設定も保持されます。パブリックエンドポイントの名前は、再有効にした後も変更されません。デプロイ履歴とコンテナイメージは保持されます。

コンテナサービスの削除は、破壊的なアクションです。サービスのすべてのコンテナノードは永久的に削除されます。サービスと関連付いている HTTPS パブリックエンドポイントのアドレス、コンテナイメージ、デプロイ履歴、ログも永久的に削除されます。エンドポイントアドレスを復元することはできません。

コンテナサービスが無効状態でも、課金されますか?

はい。コンテナサービスが無効な状態であっても、コンテナサービスのパワーとスケールの設定に従って課金されます。

Lightsail コンテンツ配信ネットワーク (CDN) ディストリビューションのオリジンとしてコンテナサービスを使用できますか?

コンテナサービスは現在、Lightsail CDN ディストリビューションのオリジンとしてはサポートされていません。

Lightsail ロードバランサーのターゲットとしてコンテナサービスを使用できますか?

いいえ。 現在、コンテナサービスは Lightsail ロードバランサーのターゲットとして使用できません。ただし、コンテナサービスのパブリックエンドポイントにはビルトインロードバランシングが備わっています。

HTTP リクエストを HTTPS にリダイレクトするよう、コンテナサービスのパブリックエンドポイントを設定できますか?

Lightsail コンテナサービスのパブリックエンドポイントは、コンテンツが安全に配信されるように、すべての HTTP リクエストを HTTPS に自動的にリダイレクトします。

コンテナサービスはモニタリングととアラートをサポートしていますか?

コンテナサービスは、サービスのノード全体の CPU 使用率とメモリ使用率に関するメトリクスを表示します。これらのメトリクスに基づくアラートは、現在サポートされていません。

Lightsail コンテナサービスは IPv6 をサポートしていますか?

Lightsail コンテナサービス HTTPS エンドポイントは IPv4 と IPv6 の両方をサポートしています。コンテナサービスでは Pv6 を無効にすることはできません。

データベース

Lightsail マネージドデータベースとは何ですか?

Lightsail マネージドデータベースは、ウェブサーバーやメールサーバーなどの他のワークロードではなく、データベースの実行専用のインスタンスです。マネージドデータベースには、複数のユーザーが作成したデータベースを含めることができ、スタンドアロンデータベースで使用する同じツールやアプリケーションを使用してアクセスできます。Lightsail はデータベースの基盤となるインフラストラクチャーとオペレーティングシステムのセキュリティと状態を維持するため、インフラストラクチャー管理に関する深い専門知識がなくてもデータベースを実行できます。

通常の Lightsail インスタンスと同様に、Lightsail マネージドデータベースのプランには固定量のメモリ、計算能力、SSD ベースのストレージが付属しており、時間の経過とともにスケールアップできます。Lightsail は、作成時に選択したデータベースを自動的にインストールして設定します。

Lightsail マネージドデータベースでは何ができますか?

Lightsail マネージドデータベースは、データをクラウドに保存するための簡単でメンテナンスの少ない方法を提供します。マネージドデータベースは、新しいデータベースとして実行することも、既存のオンプレミスデータベースまたはホストデータベースから Lightsail に移行することによって実行することもできます。

また、データベースをハードウェア専有インスタンス内に分離することで、より大量のトラフィックとより集中的な負荷を受け入れるようにアプリケーションをスケールすることもできます。Lightsail マネージドデータベースは、1 つのインスタンスを超えてスケーリングする際にデータの同期を維持する必要があるステートフルアプリケーション (一般的な CMS など WordPress) に特に役立ちます。マネージドデータベースを Lightsail ロードバランサーと 2 つ以

上の Lightsail インスタンスと組み合わせて、強力でスケーラブルなアプリケーションを作成できます。Lightsail の高可用性マネージドデータベースプランを使用すると、データベースに冗長性を追加して、アプリケーションの稼働時間を長くすることができます。

Lightsail は私にとって何を管理してくれますか？

Lightsail は、マネージドデータベースとその基盤となるインフラストラクチャのさまざまなメンテナンスアクティビティとセキュリティを管理します。Lightsail はデータベースを自動的にバックアップし、データベース復元ツールを使用して過去 7 日間の時点の復元を可能にします。これにより、データ損失やコンポーネント障害からの保護に役立ちます。また、Lightsail は安全性を高めるために保存中と移動中のデータを自動的に暗号化し、データベースに簡単かつ安全に接続できるようにデータベースパスワードを保存します。メンテナンス側では、Lightsail は設定したメンテナンス期間中にデータベースのメンテナンスを実行します。このメンテナンスには、最新のマイナーデータベースバージョンへの自動アップグレードと、基盤となるインフラストラクチャおよびオペレーティングシステムの全面的な管理が含まれます。

Lightsail はどのような種類のデータベースとどのバージョンをサポートしていますか?

Lightsail マネージドデータベースは MySQL と PostgreSQL の最新のメジャーバージョンをサポートしています。現在、これらのバージョンは MySQL 5.7、MySQL 8.0、PostgreSQL 9、PostgreSQL 10、PostgreSQL 11、および PostgreSQL 12 です。Lightsail は、各メジャーバージョンオプションに対して最新のマイナーバージョンのみを提供します。

Lightsail はどのようなマネージドデータベースプランを提供していますか?

Lightsail は、標準プランと高可用性プランで 4 種類のマネージドデータベースを提供しています。各プランには固定のストレージ容量と月間データ転送許容枠が付いています。しばらくしてから必要に応じてより大きなプランにスケールアップしたり、スタンダートプランと高可用性プランを切り替えたりすることもできます。高可用性プランにはスタンダートプランと同じリソースが含まれるほかに、プライマリデータベースとは別のアベイラビリティーゾーンで実行されるスタンバイデータベースが含まれているので、冗長性に富んでいます。

高可用性プランとは何ですか?

Lightsail マネージドデータベースは、標準プランと高可用性プランで利用できます。スタンダードプランと高可用性プランには、メモリやストレージ、データ転送許容枠など、同じプランリソースが含まれています。高可用性プランは、プライマリデータベースとは別のアベイラビリティーゾーンにスタンバイデータベースを自動的に作成し、スタンバイデータベースにデータを同期的に複製し、インフラストラクチャー障害時やメンテナンス時にスタンバイデータベースへのフェイルオーバーを提供することで、データベースの冗長性と耐久性を高めます。これにより、データベースが Lightsail によって自動的にアップグレード/保守されている場合でも稼働時間

を確保できます。高可用性プランは、高いアップタイムが要求されるプロダクション用のアプリケーションやソフトウェアを実行する場合に使用します。

Lightsail マネージドデータベースをスケールアップまたはスケールダウンする方法を教えてください。

Lightsail マネージドデータベースは、スナップショットを取得してスナップショットから新しい大規模なデータベースプランを作成するか、緊急復元機能を使用して新しい大規模なデータベースを作成することでスケールアップできます。また、これらのいずれかの方法でスタンダードプランと高可用性プランを切り替えることも可能です。データベースをスケールダウンすることはできません。詳細については、「Amazon Lightsail のスナップショットからデータベースを作成する」を参照してください。

Lightsail マネージドデータベースをバックアップする方法を教えてください。

Lightsail はデータを自動的にバックアップし、このデータを特定の時点から新しいデータベースに復元できるようにします。自動バックアップはデータベースの無料サービスですが、過去 7 日分のデータしか保存されません。データベースを削除すると、自動バックアップレコードはすべて削除され、 point-in-time 復元はできなくなります。データベース削除後にデータのバックアップを保持したり、過去7日以前のバックアップを保持したい場合は、手動スナップショットを使用します。

Lightsail 管理データベースのスナップショットは、データベース管理ページから手動で作成できます。手動スナップショットにはデータベース内のすべてのデータが含まれるので、永続的に保存したいデータのバックアップとして使用できます。手動スナップショットを使用して、より大きな新規データベースを作成したり、スタンダートプランと高可用性プランを切り替えたりすることもできます。手動スナップショットは削除するまで保存され、0.05 USD/GB (毎月) が請求されます。

Lightsail マネージドデータベースを削除するとデータはどうなりますか?

Lightsail 管理データベースを削除すると、データベース自体とすべての自動バックアップの両方が削除されます。データベースを削除する前に手動スナップショットを作成した場合を除き、このデータを復元する方法はありません。Lightsail では、データベースの削除中に、必要に応じて手動でスナップショットを取得できるワンクリックオプションが用意されています。これにより、データが誤って失われるのを防ぐことができます。削除前の手動スナップショット作成は任意となりますが、強くお勧めします。手動スナップショットは、保存したデータが不要になった時点で削除できます。

AWS リージョン別のアベイラビリティーゾーンまたは異なるアベイラビリティーゾーンで実行されている Lightsail マネージドデータベースにインスタンスを接続できますか?

Lightsail マネージドデータベースは、 AWS リージョン異なるインスタンスで実行されているインスタンスでは使用できません。ただし、ユーザーのインスタンスとは異なるアベイラビリティーゾーンのデータベースは使用いただけます。

Lightsail 管理データベースにデータをロードする方法を教えてください。

Lightsail 管理データベースにデータをロードするには、まずデータインポートモードを有効にする必要があります。データのインポートモードを有効にすると、お好みのデータベースクライアントを使用してデータを手動でアップロードできます。データのロードが完了したら、必ずデータのインポートモードをオフにし、データベースの自動バックアップと ログ記録が再開されるようにしてください。詳細については、「MySQL データベースにデータをインポートする」および「PostgreSQL データベースにデータをインポートする」を参照してください。

Lightsail マネージドデータベースのデータにアクセスする方法を教えてください。

一般的な SQL クライアントアプリケーションを使用してデータベースに接続し、データをクエリできます。GUI ベースの管理とクエリには MySQL Workbench をお勧めします。エンドポイント URL や DNS 名などの接続データは、データベース内のデータベース管理画面で確認できます。詳細については、「MySQL データベースConnect する」または「Amazon Lightsail で PostgreSQL データベースに接続する」を参照してください。

Lightsail マネージドデータベースは Lightsail インスタンスとどのように連携しますか?

Lightsail マネージドデータベースを作成したら、Lightsail インスタンスをウェブサーバーまたはアプリのその他の専用ワークロードとして使用して、すぐにアプリケーションで使用を開始できます。Lightsail インスタンスをデータベースに接続するには、データベースエンドポイントを使用し、安全に保存されたパスワードを参照して、データベースをアプリケーションのコード内のデータストアとして設定します。接続データはデータベース管理画面で確認できます。データベース設定ファイルのファイル名とロケーションはアプリケーションによって異なります。なお、同じテーブルまたは別のテーブルを使用して、複数のインスタンスを 1 つのデータベースに接続することが可能です。

Lightsail AWS マネージドデータベースを自分のアカウントで実行されている EC2 インスタンスに接続する方法を教えてください。

Lightsail マネージドデータベースは、パブリックインターネット経由で接続することで EC2 インスタンスに接続できます。 AWS すべてのサービスに接続するとデータベースのデータ転送容量が消費され、 AWS データ転送許容量を超えるデータをパブリックインターネット経由でサービ

スに送信すると超過料金が発生することに注意してください。Lightsail マネージドデータベースと EC2 インスタンス間では VPC ピアリングを使用できません。

Lightsail マネージドデータベースのパブリックモードとプライベートモードの違いは何ですか?

デフォルトでは、Lightsail マネージドデータベースはプライベートモードで作成され、Lightsail インスタンスのみがアクセスできるようにすることで安全性が確保されます。パブリックインターネットを介してソフトウェアやサービスに接続する必要がある場合は、データベースをパブリックモードに設定します。データの安全性を維持するため、パブリックモードを長期的に有効にしておくことはお勧めしません。パブリックモードとプライベートモードは、データベース管理画面からいつでも切り替えることができます。

Lightsail 管理データベースが使用するポートを管理できますか?

いいえ。Lightsail はセキュリティ上の目的でポートを自動的に管理し、すべての Lightsail 管理データベースの MySQL 用ポート 3306 をパブリックモードで開きます。データベースがプライベートモードの場合、データベースは内部ネットワーク経由で Lightsail アカウントで実行されているリソースにのみ公開されます。

Lightsail マネージドデータベースサービスは IPv6 をサポートしていますか?

Lightsail マネージドデータベースは IPv6 をサポートしていません。

ブロックストレージ

Lightsail ブロックストレージでは何ができますか?

Lightsail ブロックストレージは、個々のハードドライブと同様に、Lightsail インスタンスに接続できる追加のストレージボリューム (Lightsail では「アタッチされたディスク」と呼ばれます) を提供します。アタッチ済みディスクは、特定のデータをコアサービスから分離する必要のあるアプリケーションやソフトウェアに役立ちます。インスタンスやその他のシステムディスクに障害や不具合が発生した場合に、アプリケーションデータを保護することが可能です。保存されたデータに頻繁にアクセスするアプリケーションやソフトウェアは、一貫したパフォーマンスと低レイテンシーを必要としますが、アタッチ済みディスクはそれを実現します。

Lightsail ブロックストレージディスクはソリッドステートドライブ (SSD) を使用します。このタイプのブロックストレージは、低価格と優れたパフォーマンスを兼ね備えており、Lightsail で実行されるワークロードの大部分をサポートすることを目的としています。持続的な IOPS パフォーマンス、ディスクあたりの高いスループットを必要とするアプリケーション、または MongoDB、Cassandra などの大規模なデータベースを実行しているアプリケーションを使用す

るお客様には、Lightsail の代わりに Amazon EC2 と GP2 またはプロビジョンド IOPS SSD ストレージを使用することをお勧めします。

アタッチされたディスクは、私の Lightsail プランに含まれるストレージとどう違うのですか?

Lightsail プランに含まれるシステムディスクは、インスタンスのルートデバイスです。インスタンスを終了すると、システムディスクも削除されます。インスタンスに障害が発生した場合、システムディスクにも影響が及ぶ可能性があります。またシステムディスクをデタッチしたり、インスタンスと切り離してバックアップすることができません。アタッチ済みディスクに保存されたデータは、インスタンスから独立て存続します。アタッチ済みディスクはデタッチしたり、インスタンス間で移動させることができます。またディスクの手動スナップショットを作成することで、インスタンスから独立してバックアップできます。データを保護するために、Lightsail インスタンスのシステムディスクは一時データにのみ使用することをお勧めします。より高いレベルの耐久性が必要なデータには、アタッチ済みディスクを使用す、ディスクまたはインスタンスのスナップショットでディスクを定期的にバックアップすることをお勧めします。

アタッチ済みディスクの容量は、どれくらいまで増やせますか?

接続する各ディスクは最大 16 TB で、Lightsail アカウントにアタッチされるブロックストレージの合計容量は 20 TB を超えてはなりません。

Lightsail インスタンスごとに何台のディスクを接続できますか?

Lightsail インスタンスには最大 15 個のディスクをアタッチできます。

1 台のディスクを複数のインスタンスにアタッチすることはできますか?

できません。ディスクは一度に 1 つのインスタンスにだけアタッチできます。

ディスクはインスタンスにアタッチする必要がありますか?

いいえ、ディスクをインスタンスにアタッチしない選択も可能です。ディスクは、アタッチされていない状態でアカウントに残ります。ディスクがインスタンスにアタッチされていなくても、料金の違いはありません。

アタッチ済みディスクの容量を拡張することはできますか?

はい、ディスクの容量を拡張するには、ディスクのスナップショットを取得し、そのスナップショットからより大きいディスクを新規作成します。

Lightsail ブロックストレージは暗号化に対応していますか?

はい。データを安全に保つため、Lightsail に接続されたディスクとディスクスナップショットはすべて、Lightsail がユーザーに代わって管理するキーを使用して、デフォルトで保存時に暗号化

されます。Lightsail は、Lightsail インスタンスと接続されたディスクの間を移動するデータの暗号化も行います。

Lightsail ブロックストレージにはどのような可用性が期待できますか?

Lightsail ブロックストレージは、可用性と信頼性が高いように設計されています。コンポーネントの障害から保護するために、各アタッチ済みディスクはアベイラビリティーゾーン内で自動的にレプリケートされます。Lightsail ブロックストレージディスクは 99.99% の可用性を実現するように設計されています。Lightsail はディスクスナップショットもサポートしているため、データを定期的にバックアップできます。

アタッチ済みディスクをバックアップするには、どうすればよいですか?

ディスクの手動スナップショットを作成することで、ディスクをバックアップできます。またインスタンスの手動スナップショットを作成すれば、インスタンス全体とアタッチされたすべてのディスクをバックアップできます。なお、ディスクがアタッチされているインスタンスの自動スナップショットを有効にするとでも、バックアップは可能です。インスタンスにアタッチされたディスクはインスタンスの手動および自動スナップショットに含まれます。

ロードバランサー

Lightsail ロードバランサーでは何ができますか?

Lightsail ロードバランサーを使用すると、可用性の高いウェブサイトやアプリケーションを構築できます。Lightsail ロードバランサーは、異なるアベイラビリティーゾーンのインスタンスにトラフィックを分散し、トラフィックを正常なターゲットインスタンスのみに向けることで、インスタンスの問題やデータセンターの停止によりアプリケーションが停止するリスクを軽減します。Lightsail ロードバランサーと複数のターゲットインスタンスを使用すると、ウェブサイトまたはアプリケーションはウェブトラフィックの増加に対応し、負荷のピーク時に訪問者のパフォーマンスを良好に保つこともできます。

さらに、Lightsail ロードバランサーを使用すると、安全なアプリケーションを構築し、HTTPS トラフィックを受け入れることができます。Lightsail は SSL/TLS 証明書のリクエスト、プロビジョニング、保守の複雑さを軽減します。ビルトインの証明書管理は、ユーザーの代わりに証明書をリクエストおよび更新し、証明書をロードバランサーに自動的に追加します。

ロードバランサーは、異なるアベイラビリティーゾーンまたは異なるアベイラビリティーゾーンのインスタンスで使用できますか? AWS リージョン

ロードバランサーは、異なる s で実行されているインスタンスでは使用できません。 AWS リージョンただし、異なるアベイラビリティーゾーンのターゲットインスタンスでは、ロードバラン

サーを使用できます。そのため、ターゲットインスタンスを複数のアベイラビリティーゾーンに分散して、アプリケーションの可用性を最大化することをお勧めしています。

Lightsail ロードバランサーはトラフィックの急増にどのように対処しますか?

Lightsail ロードバランサーは、アプリケーションへのトラフィックの急増に対応するように自動的にスケーリングされ、手動で調整する必要はありません。アプリケーションのトラフィックが一時的に急増した場合、Lightsail ロードバランサーは自動的にスケーリングし、引き続きトラフィックを Lightsail インスタンスに効率的に転送します。Lightsail ロードバランサーはトラフィックの急増を簡単に管理できるように設計されていますが、トラフィック量が常に非常に多いアプリケーションでは、パフォーマンスの低下やスロットリングが発生する可能性があります。5 GB/時間を超える継続的なデータ処理や、大量接続 (新規接続 40 万/時間以上、アクティブな同時接続 1.5 万以上) がお客様のアプリケーションに発生すると予想される場合、Application Load Balancerを備えた Amazon EC2 の使用をお勧めします。

Lightsail ロードバランサーはどのようにトラフィックをターゲットインスタンスにルーティングしますか?

Lightsail ロードバランサーは、ラウンドロビンアルゴリズムに基づいてトラフィックを正常なターゲットインスタンスに転送します。

Lightsail はターゲットインスタンスが正常かどうかをどのように判断しますか?

ロードバランサーを作成してインスタンスをアタッチすると、Lightsail はヘルスチェックリクエストをウェブアプリケーションのルートに送信します。Lightsail が ping するパス (共通ファイルまたはウェブページ URL) を指定することで、場所をカスタマイズできます。このパスを使用してターゲットインスタンスにアクセスできる場合、Lightsail はトラフィックをそこにルーティングします。ターゲットインスタンスの 1 つが応答しない場合、ヘルスチェックは失敗し、Lightsail はそのインスタンスにトラフィックをルーティングしません。ヘルスチェックの詳細

ロードバランサーにアタッチできるインスタンスの数を教えてください。

Lightsail アカウントのインスタンスクォータまで、必要な数のターゲットインスタンスをロードバランサーに追加できます。

1 つのインスタンスを複数のロードバランサーに割り当てることはできますか?

はい。Lightsail では、必要に応じて複数のロードバランサーのターゲットインスタンスとしてインスタンスを追加できます。

ロードバランサーを削除すると、ターゲットインスタンスはどうなりますか?

ロードバランサーを削除しても、アタッチされたターゲットインスタンスは引き続き正常に動作し、Lightsail コンソールには通常の Lightsail インスタンスとして表示されます。ロードバランサーを削除した際は、 DNS レコードを更新して過去のターゲットインスタンスのいずれかにトラフィックを送信する必要がある場合が多いので、ご注意ください。

セッション永続性とは何ですか?

セッション永続性を使用すると、ロードバランサーは特定のターゲットインスタンスに訪問者のセッションをバインドすることができます。これにより、セッション中にそのユーザーから来たリクエストをすべて同じターゲットインスタンスに送信することができます。Lightsail は、データの一貫性を保つために訪問者が同じターゲットインスタンスにアクセスする必要があるアプリケーションのセッション永続性をサポートします。例えば、ユーザー認証を必要とする多くのアプリケーションは、セッション永続性を使用する利点があります。ロードバランサーの作成後、ロードバランサー管理画面より指定したロードバランサーに対するセッション永続性が有効にできます。詳細については、「ロードバランサーのセッション永続性を有効にする」を参照してください。

Lightsail ロードバランサーはどのような接続をサポートしていますか?

Lightsail ロードバランサーは HTTP 接続と HTTPS 接続をサポートしています。

Lightsail ロードバランサーは IPv6 をサポートしていますか?

2021 年 1 月 12 日以降に作成された Lightsail ロードバランサーは、デフォルトではデュアルスタックモードで動作します (つまり、IPv4 と IPv6 の両方のプロトコルでクライアントトラフィックを受け入れます)。この日付より前に作成されたロードバランサーでは、IPv6 はロードバランサー管理ページの [Networking] (ネットワーク) タブのトグルより有効にすることができます。このトグルを使用して、ロードバランサーの IPv6 を無効にすることも可能です。

IPv6 が有効になっているロードバランサーを使うには、そのロードバランサーの背後にあるインスタンスの IPv6 も有効になっている必要がありますか?

いいえ。 ロードバランサーは IPv4 と IPv6 両方のトラフィックを受け入れ、バックエンドのインスタンスと通信する際は、シームレスに IPv4 へ変換します。したがって、ロードバランサーの背後にあるインスタンスはデュアルスタックまたは IPv4 のみの、いずれも可能です。

コンテンツ配信ネットワークディストリビューション

Lightsail CDN ディストリビューションでは何ができますか?

Lightsail コンテンツ配信ネットワーク (CDN) ディストリビューションでは、Amazon が提供する Amazon のグローバル配信ネットワークにコンテンツを保存して配信することで、Lightsail リソースにホストされているコンテンツの配信を簡単に高速化できます。 CloudFrontまたディストリビューションでは、簡単な SSL 証明書の作成とホスティングが可能なので、ユーザーのウェブサイトにおける HTTPS トラフィックのサポートをするのにも役立ちます。最後に、ディストリビューションは Lightsail リソースの負荷を軽減し、ウェブサイトが大量のトラフィックの急増に対処するのに役立ちます。Lightsail のすべての機能と同様に、セットアップは数回クリックするだけで完了し、お支払いいただくのは簡単な月額料金です。

ディストリビューションのオリジンとして、どのような種類のリソースを使用できますか?

Lightsail ディストリビューションでは、Lightsail インスタンスとロードバランサーをオリジンとして使用できます。Lightsail コンテナは現在、オリジンとしてサポートされていません。S3 バケットなどの Lightsail 以外のリソースはサポートされていません。

Lightsail ディストリビューションのオリジンとして使用するには、Lightsail インスタンスに静的 IPv4 アドレスをアタッチする必要がありますか?

はい。静的 IPv4 アドレスは、オリジンとして指定されたインスタンスにアタッチする必要があります。Lightsail ディストリビューションは現在 IPv6 をサポートしていません。

自分のウェブサイトで Lightsail ディストリビューションをセットアップする方法を教えてください WordPress 。

ディストリビューションを作成し、 WordPress オリジンとしてインスタンスを選択し、プランを選択すれば準備は完了です。Lightsail ディストリビューションは、ほとんどの設定のパフォーマンスを最適化するようにディストリビューション設定を自動的に設定します。 WordPress

複数のオリジンをアタッチできますか?

Lightsail ディストリビューションに複数のオリジンをアタッチすることはできませんが、複数のインスタンスを Lightsail ロードバランサーにアタッチし、ディストリビューションのオリジンとして指定できます。

Lightsail ディストリビューションは証明書の作成をサポートしていますか?

はい。Lightsail ディストリビューションでは、ディストリビューションの管理ページから直接証明書を簡単に作成、検証、添付できます。

証明書は必要ですか?

カスタムドメイン名をディストリビューションで使用する場合に、証明書が必要となります。すべての Lightsail ディストリビューションは、HTTPS 対応の固有の Amazon CloudFront ドメイン名で作成されています。しかしカスタムドメインをディストリビューションで使用する場合は、カスタムドメインの証明書をディストリビューションにアタッチする必要があります。

作成できる証明書の数に制限はありますか?

はい。詳細は Lightsail サービスクォータを参照してください。

HTTP リクエストを HTTPS にリダイレクトするようにディストリビューションを設定するには、どうすればよいですか?

Lightsail ディストリビューションは、コンテンツが安全に配信されるように、すべての HTTP リクエストを HTTPS に自動的にリダイレクトします。

Lightsail ディストリビューションを指すように Apex ドメインを設定する方法を教えてください。

apex ドメインを CDN ディストリビューションに向けるには、ディストリビューションのデフォルトドメインに apex ドメインをマッピングするドメインのドメインネームシステム (DNS) に、エイリアスレコードを作成する必要があります。DNS ホスティングプロバイダーが ALIAS レコードをサポートしていない場合は、Lightsail DNS ゾーンを使用して、ディストリビューションのドメインを指すように apex ドメインを簡単に設定できます。

Lightsail のインスタンスデータ転送クォータとディストリビューションデータ転送クォータにはどのような違いがありますか?

インスタンスのデータ転送は流入と流出がデータ転送クォータの使用にカウントされますが、オリジンとビューアへのデータ転送は流出のみがディストリビューションクォータの使用にカウントされます。また、ディストリビューションのクォータを超えて流出するデータ転送はすべて超過料金が課金されますが、インスタンスの一部の流出データ転送は無料です。最後に、Lightsail のディストリビューションでは地域によって異なる超過料金モデルが使用されていますが、料金の大部分は超過料金などの請求額と同じです。

ディストリビューションと関連付いているプランを変更することはできますか?

はい。1 か月に 1 回ディストリビューションのプランを変更できます。2回目のプラン変更をご希望の場合は、翌月になるまでお待ちいただきます。

自分のディストリビューションが機能しているかどうか、どうすれば分かりますか?

Lightsail ディストリビューションには、ディストリビューションが受信したリクエストの総数、ディストリビューションがクライアントとオリジンに送信したデータ量、エラーが発生したリク

エストの割合など、ディストリビューションのパフォーマンスを追跡するさまざまな指標が用意されています。さらに、ディストリビューションメトリクスにリンクしたアラートも作成できます。

Lightsail ディストリビューションのキャッシュされたコンテンツを削除できますか?

キャッシュされたコンテンツはすべて削除できますが、削除できない特定のファイルやフォルダがあります。

Lightsail ディストリビューションと Amazon ディストリビューションのどちらを使うべきですか? CloudFront

Lightsail ディストリビューションは、インスタンスやロードバランサーなどの Lightsail リソースでウェブサイトやウェブアプリケーションをホストするユーザー向けに特別に設計されています。 AWS で別のサービスを使用してウェブサイトやアプリをホストしている場合、複雑な設定が必要な場合、または 1 秒あたりのリクエスト数が多い場合や大量のビデオストリーミングを伴うワークロードがある場合は、Amazon の使用をお勧めします CloudFront。

Lightsail コンテンツ配信ネットワーク (CDN) ディストリビューションを Amazon に移すことはできますか? CloudFront

はい。Amazon で同様に設定されたディストリビューションを作成することで Lightsail ディストリビューションを移動できます。 CloudFrontLightsail ディストリビューションで設定できるすべての設定は、ディストリビューションでも設定できます。 CloudFront ディストリビューションをに移動するには、以下の手順を実行してください。 CloudFront

- ディストリビューションのオリジンとして設定されている Lightsail インスタンスのスナップショットを作成します。スナップショットを Amazon EC2 にエクスポートし、EC2 のスナップショットから新しいインスタンスを作成します。詳細については、「スナップショットを Amazon EC2 にエクスポートする」を参照してください。

  > **Note**
  >
  > ウェブサイトまたはウェブアプリケーションをロードバランスする必要がある場合は、Elastic Load Balancing に Application Load Balancer を作成します。詳細については、Elastic Load Balancing ユーザーガイドを参照してください。

- Lightsail ディストリビューションのカスタムドメインを無効にして、アタッチした可能性のある証明書をデタッチします。詳細については、「Amazon Lightsail ディストリビューションのカスタムドメインの無効化」を参照してください。

- AWS Command Line Interface (AWS CLI) を使用して get-distributions コマンドを実行し、Lightsail ディストリビューションの設定のリストを取得します。詳細については、「AWS CLI リファレンス」の「get-distributions」を参照してください。
- CloudFrontコンソールにサインインし、Lightsail ディストリビューションと同じ設定でディストリビューションを作成します。詳細については、Amazon CloudFront 開発者ガイドの「ディストリビューションの作成」を参照してください。
- AWS Certificate Manager c (ACM) で証明書を作成し、 CloudFrontディストリビューションに添付します。詳細については、「ACM ユーザーガイド」の「パブリック証明書をリクエストする」を参照してください。
- 作成した ACM CloudFront 証明書を使用するようにディストリビューションを更新します。詳細については、『CloudFront ユーザーガイド』の「 CloudFront ディストリビューションの更新」を参照してください。

Lightsail CDN はどのように使用されることを意図していますか?

Lightsail CDN ディストリビューションは、サービスの使用コストをシンプルかつ予測可能なものにするために、固定価格のデータ転送バンドルを使用して作成されています。ディストリビューションバンドルは、1 か月分相当の使用量を カバーするように設計されています。ディストリビューションバンドルを超過料金の発生を防ぐような方法 (バンドルの頻繁なアップグレードまたはダウングレード、または異常に多量なディストリビューションを一つのオリジンで使用するなどを含むが、これらに限らない方法) で使用することは、本来の使用目的の範囲を超えるため、許可されていません。さらに、1 秒あたりのリクエスト数が多いワークロードや、大量のビデオストリーミングを伴うワークロードは許可されません。これらの動作に従事すると、データサービスまたはアカウントがスロットリングされたり停止される可能性があります。

Lightsail の CDN ディストリビューションは IPv6 をサポートしていますか?

すべての Lightsail CDN ディストリビューションでは、デフォルトで IPv6 が有効になっています。ディストリビューションのホスト名は IPv4 アドレスと IPv6 アドレスの両方に解決されます。IPv6 は、CDN の管理ページにあるネットワークタブのトグルから無効にすることができます。

オリジンの IPv6 を有効にしないと、Lightsail CDN ディストリビューションは動作しませんか?

いいえ。 CDN ディストリビューションは IPv6 と IPv4 の両方のトラフィックを受け入れ、バックエンドのオリジンと通信するときにはシームレスに IPv4 へ変換します。したがって、オリジンの背後にあるディストリビューションはデュアルスタックまたは IPv4 のみの、いずれも可能です。

証明書

Lightsail がプロビジョニングした証明書はどのように使用できますか?

SSL/TLS 証明書は、ウェブサイトまたはアプリケーションのアイデンティティを確立し、ブラウザとウェブサイトの間の接続を保護するために使用されます。Lightsail はロードバランサーで使用する署名付き証明書を提供し、ロードバランサーは検証済みのトラフィックを安全なネットワーク経由でターゲットインスタンスにルーティングする前に SSL/TLS ターミネーションを行います。 AWS Lightsail 証明書は Lightsail ロードバランサーでのみ使用でき、個々の Lightsail インスタンスでは使用できません。

証明書を認証するには、どうすればよいですか?

Lightsail 証明書はドメイン検証済みです。つまり、認証局が証明書をプロビジョニングする前に、ウェブサイトのドメインを所有しているか、アクセスできることを検証して ID を証明する必要があります。新しい証明書をリクエストすると、Lightsail は証明書を自動的に検証しようとします。証明書を自動的に検証できない場合、Lightsail は検証対象の 1 つまたは複数のドメインの DNS ゾーンに CNAME レコードを追加するように求めます。現在 DNS ゾーン (Lightsail DNS 管理または外部 DNS ホスティングプロバイダー) を管理しているすべての場所で、72 時間以内に CNAME レコードを追加できます。

ドメインを認証できない場合はどうなりますか?

安全上の理由で、ユーザーはドメイン所有者であることを認証できる必要があります。つまり、あなたや組織内の誰かが何らかの理由で証明書を検証するための DNS レコードを追加できない場合、Lightsail では HTTPS 対応のロードバランサーを使用できなくなります。

証明書に追加できるドメインおよびサブドメインの数を教えてください。

証明書ごとにドメインまたはサブドメインを最大 10 個追加できます。Lightsail は現在、ワイルドカードドメインをサポートしていません。

証明書に関連付けられたドメインを変更するには、どうすればよいですか?

証明書に関連付けられたドメインを変更 (追加/削除)する場合は、証明書を再提出してドメインの所有権を再認証する必要があります。証明書管理画面のステップに沿って証明書を再発行し、促しに応じてドメインを追加または削除します。

証明書を更新するには、どうすればよいですか?

Lightsail では、SSL/TLS 証明書の更新を管理して更新することができます。つまり、Lightsail は証明書の有効期限が切れる前に自動的に証明書を更新しようとしますが、ユーザーによるアク

ションは必要ありません。Lightsail 証明書は、自動的に更新する前にロードバランサーにアクティブに関連付ける必要があります。

ロードバランサーを削除すると、証明書はどうなりますか?

ロードバランサーが削除された場合、証明書も削除されます。その後、同じドメインに証明書を使用する必要がある場合、新しい証明書をリクエストして検証する必要があります。

Lightsail から提供された証明書をダウンロードできますか?

いいえ、Lightsail 証明書はお客様の Lightsail アカウントに紐付けられており、削除して Lightsail 以外で使用することはできません。

手動および自動スナップショット

スナップショットとは何ですか?

スナップショットはインスタンス、データベース、 point-in-time またはブロックストレージディスクのバックアップです。リソースのスナップショットはいつでも作成できます。また、インスタンスとディスクの自動スナップショットを有効にして Lightsail にスナップショットを作成させることができます。スナップショットをベースラインとして使用して、新しいリソースを作成したりデータをバックアップすることが可能です。スナップショットには、リソースの復元に必要なすべてのデータ (スナップショットが作成された時点のデータ) が含まれます。スナップショットからリソースを作成して復元すると、その新しいリソースはスナップショットの作成に使用された元のリソースの正確なレプリカとして始まります。

Lightsail インスタンス、ディスク、データベースのスナップショットを手動で作成することも、自動スナップショットを使用して、インスタンスとディスクのスナップショットを毎日自動的に作成するように Lightsail に指示することもできます。詳細については、「スナップショット」を参照してください。

自動スナップショットとは何ですか?

自動スナップショットは、Amazon Lightsail の Linux/Unix インスタンスのスナップショットを毎日スケジュールする方法です。時間帯を選択すると、Lightsail は毎日選択した時間に自動的にスナップショットを撮り、常に最新の 7 つの自動スナップショットを保持します。スナップショットの有効化は無料です。スナップショットで使われた実際のストレージの料金のみをお支払いいただきます。

手動スナップショットと自動スナップショットの違いは何ですか?

自動スナップショットはタグ付けしたり、Amazon EC2 に直接エクスポートしたりすることができません。ただし、自動スナップショットはコピーして手動のスナップショットに変換することができます。自動スナップショットを手動スナップショットにコピーするには、自動スナップショットのコンテキストメニューから [Keep] を選択して、手動スナップショットとしてコピーします。

どのようなリソースがスナップショットをサポートしていますか?

インスタンス、データベース、ディスクの手動スナップショットが作成できます。

自動スナップショットは、Linux または Unix インスタンスでは Lightsail コンソール、Lightsail API、または AWS CLI、または Lightsail API のみを使用するディスクで有効にできます。 AWS CLI自動スナップショットは現在、Windows インスタンスまたはマネージドデータベースではサポートされていません。

スナップショットはどれくらいの期間保存できますか?

手動スナップショットはユーザーが削除することを選択するまで保存されます。詳細については、「Amazon Lightsail でのスナップショットの削除」を参照してください。

自動スナップショットは、新しい自動スナップショットに置き換えられるまで保存されます。Lightsail は最新の 7 つの自動スナップショットを保存してから、最も古いものを削除して最新のものに置き換えます。ただし、手動スナップショットとしてコピーすることで、特定の自動スナップショットを保持できます。詳細については、「Amazon Lightsail でのインスタンスまたはディスクの自動スナップショットの保存」を参照してください。アカウントに保存されている自動スナップショットに対してスナップショットストレージ料金が請求されます。

自動スナップショットを有効にするには、どうすればよいですか?

自動スナップショットは、Lightsail コンソール、Lightsail API を使用するか、Linux または Unix AWS CLI インスタンスを作成するとき、またはインスタンスの実行後に有効にできます。

ディスクの自動スナップショットは、作成時または作成後に有効にすることもできますが、Lightsail API または AWS CLI を使用してのみ実行できます。

詳細については、「Amazon Lightsail のインスタンスまたはディスクの自動スナップショットの有効化または無効化」を参照してください。

自動スナップショットはいつ作成されますか?

自動スナップショットを有効にすると、リソースが配置されている AWS リージョン に基づいてデフォルト時間が設定されます。自動スナップショットの時間は 1 時間単位で希望の時刻に変更

できます。詳細については、「Amazon Lightsail のインスタンスまたはディスクの自動スナップショット時間の変更」を参照してください。

保存できるスナップショットの数を教えてください。

手動スナップショットは必要な数だけ保存できます。ただし、自動スナップショット は最新の 7 つのみが保存され、最も古いスナップショットは最新のスナップショットに置き換えられます。

スナップショットはどのように課金されますか?

お支払いいただくのは、Lightsail アカウントに保存されているスナップショットの分のみです。Lightsail スナップショット (手動および自動) の保存には 1 か月あたり 0.05 USD/GB かかります。

自動スナップショットを無効にすると、スナップショットは失われますか?

いいえ。 自動スナップショットを無効にすると、Lightsail は日次スナップショットの作成を停止し、既存の自動スナップショットは保持されます。自動スナップショットを再度有効にすると、Lightsail は毎日のスナップショットの作成を再開し、最も古いスナップショットを削除して最新のものに置き換えます。

自動スナップショットが置き換えられないようにする場合は、どうすればよいですか?

特定の自動スナップショットを、手動スナップショットとしてコピーすることで保持できます。詳細については、「Amazon Lightsail でのインスタンスまたはディスクの自動スナップショットの保存」を参照してください。

自動スナップショットは削除できますか?

自動スナップショットのコンテキストメニューから [Delete] (削除) を選択することで、いつでも自動スナップショットを削除できます。詳細については、「自動インスタンスのスナップショットを削除する」を参照してください。

スナップショットはどのように使用できますか?

スナップショットは、元のリソースに不具合が発生した場合などに、ベースラインとして使用したり、新しいリソースの作成に使用できます。スナップショットがほかにできること。詳細については、「スナップショット」を参照してください。

スナップショットを Amazon EC2 にエクスポートして、そのサービス内に新しいリソースを作成することもできます。詳細については、「スナップショットを Amazon EC2 にエクスポートする」を参照してください。

ネットワーク

Lightsail で IP を使用するにはどうすればよいですか?

各 Lightsail インスタンスは、プライベート IPv4 アドレス、パブリック IPv4 アドレス、またはパブリック IPv6 アドレスを自動的に取得します (2021 年 1 月 12 日より前に作成されたインスタンスでは IPv6 を手動で有効にする必要があります)。プライベート IP を使用して、Lightsail AWS インスタンスとリソース間のデータをプライベートに無料で送信できます。パブリック IP を使用すると、登録済みのドメイン名や SSH または ローカルコンピュータからの RDP 接続などを介して、インターネットからインスタンスに接続することができます。また、パブリック IPv4 アドレスの代わりに、インスタンスが停止および開始されても IPv4 アドレスが変化しない静的 IPv4 アドレスをインスタンスに接続することもできます。インスタンスに割り当てられた IPv6 アドレスは、インスタンスが削除されるか、インスタンスの IPv6 を無効化して IPv6 アドレスを手動で解放するまで変わりません。

Lightsail は IPv6 のみのインスタンスをサポートしていますか?

はい。Lightsail インスタンスはデュアルスタック (IPv4 と IPv6) と IPv6 のみの構成をサポートしています。

静的 IP とは何ですか?

静的 IP は、Lightsail アカウント専用の固定のパブリック IP です。静的 IPv4 アドレスをインスタンスに割り当て、パブリック IPv4 と置き換えることができます。インスタンスを別のインスタンスに置き換える場合は、静的 IP を新しいインスタンスに再割り当てすることができます。この方法では、インスタンスを置き換えるたびに外部システム (DNS レコードなど) を再設定して、新しい IP アドレスを指す必要はありません。Lightsail は現在 IPv4 の静的 IP のみをサポートしています。静的 IPv6 アドレスは使用できません。ただし、インスタンスに割り当てられた IPv6 アドレスは、インスタンスが削除されるか、インスタンス上で IPv6 を無効にして IPv6 アドレスを手動で解放するまで、変更されません。

インスタンスには静的 IP をいくつアタッチできますか?

1 つの静的 IP を 1 つのインスタンスにアタッチできます。

DNS レコードとは何ですか?

DNS は世界中に分散して存在するサービスで、`www.example.com` などの人が読むことができる名前を、コンピュータが互いに接続する際に使用される `192.0.2.1` などの英数字の IP アドレスに変換します。Lightsail を使用すると、登録したドメイン名を Lightsail インスタンスのパブリック IP `photos.example.com` などに簡単にマッピングできます。このよう

に、example.comユーザーがブラウザに人間が読めるような名前を入力すると、Lightsail はそのアドレスをユーザーを誘導したいインスタンスの IP に自動的に変換します。これらの変換は、それぞれ DNS クエリと呼ばれます。

Lightsail でドメインを使用するには、まずドメインを登録する必要があることを知っておくことが重要です。Lightsail またはお好みの DNS レジストラを使用してドメインを登録できます。

インスタンスのファイアウォール設定を管理することはできますか?

はい。Lightsail ファイアウォールを使用してインスタンスのデータトラフィックを制御できます。Lightsail コンソールから、さまざまなタイプのトラフィックに対してインスタンスのどのポートにパブリックにアクセスできるかに関するルールを設定できます。

ドメイン

Lightsail ドメインでは何ができますか?

Lightsail ドメインを使用すると、ウェブサイトまたはアプリケーションのドメインを登録および管理できます。他のプロバイダーに登録されているドメインがある場合は、それらのドメインの管理を Lightsail に移管できます。これらのドメインを Lightsail リソースにポイントすることもできます。

どの最上位ドメイン (TLD) を使用できますか?

Lightsail は Amazon Route 53 と同じ汎用 TLD を使用しています。地理的ドメインを登録する場合は、Route 53 コンソールを使用することをお勧めします。地域ドメインは Route 53 を使用して登録されると、Lightsail コンソールで利用できるようになります。Lightsail がサポートする TLD の詳細については、Amazon Route 53 開発者ガイドの「Amazon Route 53 に登録できるドメイン」を参照してください。

Lightsail を既存のドメインの DNS サービスにすることはできますか?

別の DNS サービスプロバイダーを使用して登録したドメインの DNS 管理を Lightsail に移管できます。詳細については、「DNS ゾーンを作成してドメインの DNS レコードを管理する」を参照してください。

Lightsail でドメイン登録を開始するにはどうすればよいですか?

Lightsail にログインすると、Lightsail コンソールを使用してドメインを作成および管理できます。詳細については、「ドメインの登録」を参照してください。

Lightsail と Route 53 ではどのような場合にドメインを登録すべきですか?

ドメインの登録、DNS ゾーンの作成、ドメインのトラフィックの Lightsail リソースへのルーティングなどのタスクは Lightsail で行われます。ドメイン登録の延長、トラフィックポリシーを含むドメインの移管、プライベートホストゾーンの作成などの高度なタスクには Route 53 を使用することをお勧めします。

ドメインを Lightsail に移管することはできますか?

ドメインは Route 53 に移管できます。ドメイン移管が完了すると、ドメインは Lightsail コンソールで利用できるようになります。詳細については、「Amazon Route 53 での Lightsail ドメインの管理」を参照してください。

ドメインではどの Lightsail リソースを使用できますか?

Lightsail にドメインを登録すると、ドメインを Lightsail インスタンス、コンテナ、ロードバランサー、静的 IP、またはコンテンツ配信ネットワーク (CDN) にポイントできます。

請求とアカウント管理

Lightsail プランにはどれくらいの費用がかかりますか?

Lightsail プランはオンデマンドの時間単位で請求されるため、お支払いいただくのは使用した分のみです。使用する Lightsail プランごとに、月額プランの最大費用を上限とする固定時間料金を請求します。最も安価な Lightsail プランは、1 時間あたり 0.0047 米ドル (月額 3.50 米ドル) から始まります。Windows Server ライセンスを含む Lightsail プランは、1 時間あたり 0.01075 米ドル (1 か月あたり 8 米ドル) から始まります。

プランに対して課金されるのは、どのようなときですか?

Lightsail インスタンスとマネージドデータベースは、削除されるまで料金が発生します。月末までに Lightsail インスタンスまたはマネージドデータベースを削除した場合、その月に Lightsail インスタンスまたはマネージドデータベースを使用した合計時間数に基づいて日割り計算された費用のみが請求されます。たとえば、最も安価な Lightsail インスタンスプランを 1 か月で 100 時間使用した場合、46 セント (100*0.0046) が課金されます。

Lightsail インスタンスを無料で試すことはできますか?

はい。 AWS 既存のお客様でも新規のお客様でも、3.50 USD の Lightsail プランを 750 時間無料でご利用いただけます。また、Windows Server ライセンスを含む Lightsail プランを 8 米ドルの Windows プランで無料で試すこともできます。

750 時間内で使用するインスタンスの数に制限はありません。たとえば、1 つの Lightsail インスタンスを 1 か月間実行したり、10 個の Lightsail インスタンスを 75 時間実行したりできます。無料試用版は、Lightsail の使用にサインアップしてから最初の暦月以内の使用にのみ適用されます。アカウントが (AWS Organizations 配下の) 組織にリンクされている場合、AWS 無料利用枠を利用できるのは組織内の 1 つのアカウントのみです。

> **Note**
>
> AWS 無料利用枠の一部として、一部のインスタンスバンドルで Amazon Lightsail を無料で開始できます。詳細については、Amazon Lightsail 料金ページの「AWS 無料利用枠」を参照してください。

Lightsail の無料トライアルはいつ開始されますか?

Lightsail 無料トライアルの特典は、最初の無料トライアル対象リソースがリリースされた時点で開始されます。

インスタンスとデータベースの 90 日間の延長無料トライアルは、一部のプラン (バンドル) にのみ適用されます。このオファーは、2021 年 7 月 8 日以降に Lightsail AWS の使用を開始した新規または既存のアカウントに適用されます。詳細については、Lightsail の料金 ページを参照してください。

Lightsail マネージドデータベースにはどれくらいの費用がかかりますか?

Lightsail マネージドデータベースには 4 種類のプランサイズがあり、40 GB の SSD ストレージと 100 GB のデータ転送許容量を備えた 1 GB の RAM データベースインスタンスの場合、1 か月あたり 15 USD からご利用いただけます。高可用性プランはスタンダードプランの 2 倍の費用がかかります。これは、冗長性のために別のアベイラビリティーゾーンで追加のデータベースインスタンスとストレージが実行されるためです。

Lightsail マネージドデータベースを無料で試すことはできますか?

はい。Lightsail を初めてご利用のお客様は、15 米ドルの Lightsail プランの 1 か月分を無料でご利用いただけます。

Lightsail ブロックストレージの費用はいくらですか?

Lightsail ブロックストレージの料金は 1 GB あたり 1 か月あたり 0.10 米ドルです。

Lightsail ロードバランサーにはどれくらいの費用がかかりますか?

Lightsail ロードバランサーの料金は 1 か月あたり 18 米ドルです。

証明書管理の課金対象を教えてください。

Lightsail ロードバランサーを使用すると、Lightsail 証明書と証明書管理は無料になります。

Lightsail のスタティック IPv4 アドレスにはどれくらいの費用がかかりますか?

静的 IP アドレスを Lightsail インスタンスにアタッチする場合、それに関連するコストは発生しません。静的 IP は IPv6 専用インスタンスにはアタッチできません。IPv4 アドレスは希少なリソースであり、Lightsail はそれらを効率的に使用できるよう努めています。そのため、インスタンスに 1 時間以上アタッチされていない静的 IP には、1 時間あたり 0.005 USD の少額の料金を請求します。

データ転送の課金対象を教えてください。

インスタンス、データベース、およびコンテンツ配信ネットワーク (CDN) の配信プランには、データ転送枠が含まれます。

Lightsail インスタンスの場合、インスタンスからのデータ転送とインスタンスからのデータ転送の両方がデータ転送許容量にカウントされます。データ転送許容量を超えた場合、Lightsail インスタンスからインターネット、またはインスタンスのパブリック IP AWS アドレスを使用するリソースへのデータ転送 OUT に対してのみ課金されます。インスタンスのプライベート IP アドレスを使用する場合の Lightsail インスタンスへのデータ転送 IN と Lightsail インスタンスからのデータ転送 OUT はいずれも、データ転送許容量を超えて無料です。

Lightsail が管理するデータベースでは、データ転送 OUT のみが許容量にカウントされます。データ転送許容量を超えた場合、Lightsail が管理するデータベースからインターネットへのデータ転送 OUT に対してのみ課金されます。

Lightsail CDN ディストリビューションでは、ディストリビューションからのすべてのデータ転送が許容量にカウントされます。ディストリビューションのデータ転送許容量を超えると、ディストリビューションからのすべてのデータ転送に料金が発生します。

データ転送枠はロードバランサーではどのように機能しますか?

ロードバランサーはデータ転送枠を消費しません。インターネットに出入りするトラフィックが、ロードバランサーの背後にない Lightsail インスタンスのデータ転送許容量にカウントされるのと同様に、ロードバランサーとターゲットインスタンスまたはディストリビューション間のトラフィックは計測され、インスタンスまたはディストリビューションのデータ転送許容量にカウントされます。ロードバランサーとインターネットの間を行き来するトラフィックは、インスタンスのデータ転送枠の利用としてカウントされません。

プランのデータ転送許容量を超えた場合は、どうすればよいですか?

データ転送プランは、大部分のお客様の使用量が枠内で収まり、追加料金の請求が発生しないように設計されています。インスタンスがデータ転送枠を超えた場合は、使用したデータ転送量に対して GB 単位で超過料金が発生します (インターネットへのデータ転送に対してのみ)。

インスタンスがプランのデータ転送枠を超過した場合でも、多くのタイプのデータ転送が無料です。Lightsail インスタンスとデータベースへのデータ転送は常に無料です。プライベート IP アドレスが使用されている場合、Lightsail インスタンスから別の Lightsail インスタンスへのデータ転送、Lightsail インスタンスと Lightsail マネージドデータベース間、 AWS または同じリージョンのリソースへのデータ転送も無料です。

どのような種類のデータ転送が課金されますか?

インスタンスプランの毎月の無料データ転送許容量を超えると、パブリック IP アドレスを使用する場合に、Lightsail インスタンスからインターネット、別のインスタンス、または同じリージョンのリソースへのデータ転送、 AWS リージョン AWS または同じリージョンのリソースへのデータ転送に対して課金されます。無料容量分を超過した際のこれらのデータ転送料金は、以下の通りです。

- 米国東部 (オハイオ) (us-east-2): 0.09 USD/GB
- 米国東部 (バージニア北部) (us-east-1): 0.09 USD/GB
- 米国西部 (オレゴン) (us-west-2): 0.09 USD/GB
- アジアパシフィック (ムンバイ) (ap-south-1): 0.13 USD/GB
- アジアパシフィック (ソウル) (ap-northeast-2): 0.13 USD/GB
- アジアパシフィック (シンガポール) (ap-southeast-1): 0.12 USD/GB
- アジアパシフィック (シドニー) (ap-southeast-2): 0.17 USD/GB
- アジアパシフィック (東京) (ap-northeast-1): 0.14 USD/GB
- カナダ (中部) (ca-central-1): 0.09 USD/GB
- 欧州 (フランクフルト) (eu-central-1): 0.09 USD/GB
- 欧州 (アイルランド) (eu-west-1): 0.09 USD/GB
- 欧州 (ロンドン) (eu-west-2): 0.09 USD/GB
- 欧州 (パリ) (eu-west-3): 0.09 USD/GB

- 欧州 (ストックホルム) (eu-north-1): 0.09 USD/GB

複数のアベイラビリティーゾーンで作成されたインスタンスは、ゾーン間でプライベートに無料通信でき、同時に障害が発生しにくくなります。アベイラビリティーゾーンでは、データ転送コストが増加したり、アプリケーションの安全性を損なうことなく、可用性の高いアプリケーションまたはウェブサイトを構築できます。

Lightsail CDN 配信プランのデータ転送許容量を超えると、すべてのデータ転送 OUT に対して料金が請求されます。ディストリビューションの許容量を超えるデータ転送の料金は Lightsail インスタンスとは異なり、以下のとおりです。

- アジアパシフィック: 0.13 USD/GB
- カナダ：0.09 USD/GB
- 欧州：0.09 USD/GB
- インド：0.13 USD/GB
- 日本：0.14 USD/GB
- 中東：0.11 USD/GB
- 南アフリカ：0.11 USD/GB
- 南アメリカ：0.11 USD/GB
- 米国：0.09 USD/GB

インスタンスのデータ転送プランの許容量は、 AWS リージョンによってどのように異なりますか?

AWS リージョンアジアパシフィック (ムンバイ) リージョンとアジアパシフィック (シドニー) リージョンを除き、すべてのインスタンスには amazonlightsail.com と amazonlightsail.com/pricingesに記載されているものと同じデータ転送プランの許容量が適用されます。これら 2 つのインスタンスにおけるデータ転送プランの許容量は以下のとおりです AWS リージョン。

- 3.50 USD/月プラン: 0.5 TB
- 5 USD/月プラン: 1 TB
- 10 USD/月プラン: 1.5 TB
- 20 USD/月プラン: 2 TB
- 40 USD/月プラン: 2.5 TB

- 80 USD/月プラン: 3 TB
- 160 USD/月プラン: 3.5 TB

Lightsail マネージドデータベースのデータ転送許容量は、すべての地域で同じです。

インスタンスにおけるデータ転送枠はどのように機能しますか?

すべての Lightsail インスタンスプランにはデータ転送許容量が含まれています。例えば 1 か月あたり 3.50 USD のプランを利用すると、追加料金なしで 毎月最大 1 TB のデータを、インスタンスとインターネット間で送受信できます。データ転送枠は毎月リセットされますが、その月の間であれば必要なときにいつでも消費できます。

インスタンスがその月のデータ転送許容量に達した場合は、インスタンスが置かれている AWS リージョン に応じて、GB あたり 0.09 USD からのインターネットへのデータ転送 OUT の料金が請求されます。同じ月にインスタンスを削除して別のインスタンスを作成した場合、無料のデータ転送上限は 2 つのインスタンス間で共有されます。 AWS リージョン

Lightsail ドメインにはどれくらいの費用がかかりますか?

リンク先の .pdf ファイルに記載されている料金は、2021 年 12 月 22 日以降の新規ドメイン名登録、既存のドメイン名登録の更新に適用されます。すべての料金には DNS ゾーンとプライバシー保護が含まれています。ドメイン登録のコストの詳細については、「Amazon Route 53 のドメイン登録の料金」および「ドメイン登録」を参照してください。

Lightsail DNS 管理にはどれくらいの費用がかかりますか?

Lightsail 内では DNS 管理は無料です。最大で 6 つの DNS ゾーンと、各 DNS ゾーンに必要な数だけのレコードを作成できます。1 か月あたり 300 万 DNS クエリの月間許容枠が、お客様のゾーンでご利用いただけます。1 か月のクエリ数が最初の 300 万を超えると、100万 DNS クエリあたり 0.40 USD の料金が適用されます。

Lightsail スナップショットにはいくらかかりますか?

Lightsail スナップショット (手動および自動) の保存には 1 か月あたり 0.05 USD/GB かかります。つまり、28 GB のスペースを使用しているインスタンスのスナップショットを作成して、それを1 か月間保持した場合、その月は 1.40 USD のお支払いとなります。

同じインスタンスの複数のスナップショットを連続して作成すると、Lightsail はスナップショットのコストを自動的に最適化します。新しいスナップショットを作成するたびに、変更されたデータ部分に対してのみ課金されます。上記の例でいうと、インスタンスのデータが 2 GB 分のみ変更された場合、2 番目のインスタンススナップショットにかかる料金は 1 か月あたり 0.10 USD のみとなります。

AWS アカウントを管理するには、どうすればよいですか?

Lightsail AWS AWS は信頼性が高く実績のあるクラウドインフラストラクチャ上で稼働するサービスです。 AWS 同じアカウントと認証情報を使用して Lightsail と AWS マネジメントコンソールにログインします。

AWS アカウントのパスワード、ユーザー名、連絡先情報、請求情報の変更など、 AWS アカウントを管理するには、AWS Billing and Cost Management コンソールを使用します。

Lightsail の法的利用規約にはどのようなものがありますか?

Lightsail はAmazon ウェブサービスです。Lightsail を使用するには、AWS まずカスタマー契約とサービス条件に同意する必要があります。Lightsail インスタンスを作成する場合、ソフトウェアの使用には販売者のエンドユーザー使用許諾契約が適用されることにも同意したものとみなされます。この契約は、インスタンスの作成ページで確認できます。

Lightsail の請求書の支払い方法を教えてください。

AWS Billing and Cost Management コンソールから請求書の支払いと管理を行うことができます。 AWS ほとんどの主要クレジットカードに対応しています。お支払い方法の管理についての詳細は、こちらを参照してください。

Amazon Elastic Compute Cloud (Amazon EC2) へのエクスポート

Amazon EC2 へのエクスポートとは

Amazon EC2 へのエクスポートは、Amazon EC2 で Lightsail インスタンスのコピーを作成できるようにする機能です。Amazon EC2 にエクスポートすると、Amazon EC2 が提供する様々なインスタンスタイプ、環境設定、および料金モデルから選択できます。ネットワーキングやストレージ、コンピューティング環境をより細かく調整することが可能になります。

Amazon EC2 にエクスポートする理由は何ですか?

Lightsail では、さまざまなクラウドベースのアプリケーションを、バンドルされた予測可能な低価格で簡単に実行およびスケーリングできます。Lightsail は、ネットワークやアクセス管理などのクラウド環境設定も自動的にセットアップします。

Amazon EC2 にエクスポートすることで、様々なインスタンスタイプでのアプリケーションの実行が可能になります。より強力な CPU パワーやメモリ、またネットワーキング機能を備えた仮想マシンから、FPGA や GPU を備えた特殊化および高速化されたインスタンスまで、多岐にわたります。さらに、Amazon EC2 では自動で実行される管理やセットアップが少ないため、VPC などのクラウド環境の設定をより細かく調整できます。

Amazon EC2 へのエクスポートの仕組みを教えてください。

開始するには、Lightsail インスタンスまたはブロックストレージディスクの手動スナップショットをエクスポートする必要があります。Amazon EC2 の使用に慣れているユーザーであれば、Amazon EC2 作成ウィザードまたは API を使用して、新しい Amazon EC2 インスタンスまたは Amazon EBS ボリュームを作成できます。これらは、既存の EC2 AMI や EBS ボリュームから作成されます。また、Lightsail にはガイド付きの Lightsail コンソールエクスペリエンスも用意されているため、新しい EC2 インスタンスを簡単に作成できます。

> **Note**
>
> 現時点では、cPanel & WHM、Django、および Ghost インスタンスのスナップショットを Amazon EC2 にエクスポートすることはできません。

どのように請求されますか?

Amazon EC2 へのエクスポート機能の使用は無料です。手動スナップショットを Amazon EC2 にエクスポートすると、Lightsail の手動スナップショットに加えて Amazon EC2 イメージの料金が別途請求されます。また新しい Amazon EC2 インスタンスを起動した場合にも、Amazon EC2 によって課金されます。これには、インスタンスの Amazon EBS ストレージボリュームやデータ転送が含まれます。新しいインスタンスとリソースの料金の詳細については、「Amazon EC2 料金」ページを参照してください。Lightsail アカウントで引き続き実行されている Lightsail リソースは、削除されるまで通常の料金で請求され続けます。

マネージドデータベースやディスクのスナップショットはエクスポートできますか?

エクスポート機能を使用すると、手動の Lightsail ディスクスナップショットをエクスポートできますが、現在、管理対象データベースの手動スナップショットはサポートされていません。ディスクスナップショットは、Amazon EC2 コンソールまたは API から、Amazon EBS ボリュームとして復元できます。

どのような Lightsail リソースをエクスポートできますか?

Lightsail の Amazon EC2 へのエクスポート機能は、Linux および Windows インスタンスのスナップショットの Amazon EC2 へのエクスポートをサポートするように設計されています。またこれは、Amazon EBS へのブロックストレージディスクのスナップショットのエクスポートもサポートしています。データベース、コンテナサービス、コンテンツ配信ネットワーク (CDN) ディストリビューション、ロードバランサー、静的 IP、および DNS レコードのエクスポートは現在サポートされていません。さらに、現時点では Django、 Ghost、および cPanel と WHM の インスタンススナップショットは Amazon EC2 にエクスポートすることはできません。

Lightsail のタグ

タグとは何ですか?

タグは Lightsail リソースに割り当てるラベルです。タグはそれぞれ、1 つのキーと 1 つの値で構成されており、どちらもお客様側が定義します。タグ値はオプションなので、Lightsail コンソールでリソースをフィルタリングするための「キー専用」タグを作成することを選択できます。

Lightsail でタグを使用するにはどうすればよいですか?

タグには複数のユースケースがあります。タグを使用すると、Lightsail コンソールと API でリソースをグループ化してフィルタリングしたり、請求書のコストを追跡して整理したり、アクセス管理ルールを通じてリソースを表示または変更できるユーザーを制限したりできます。リソースにタグを付けることで、以下のことができます。

• 整理-Lightsail コンソールと API フィルターを使用して、割り当てたタグに基づいてリソースを表示および管理します。これは、同じタイプのリソースが多数ある場合に役立ちます。割り当てたタグに基づいて特定のリソースをすばやく識別できます。

• コスト配分 – リソースにタグを付けて請求コンソールで「コスト配分タグ」を作成することで、さまざまなプロジェクトやユーザー間のコストの追跡や配分を行います。例えば、請求書を分割してプロジェクト別またはクライアント別にコストを確認することができます。

• アクセス管理- AWS アカウントへのアクセス権を持つユーザーがポリシーを使用して AWS Identity and Access Management Lightsail リソースを編集、作成、削除する方法を制御します。これにより、他のユーザーに Lightsail リソースへのフルアクセスを許可しなくても、他のユーザーとのコラボレーションがより簡単になります。

Lightsail でのタグの使用について詳しくは、「タグ」を参照してください。

タグ付けできるのはどのようなリソースですか?

Lightsail は現在、以下のリソースのタグ付けをサポートしています。

• インスタンス (Linux および Windows)

・ コンテナサービス

• ブロックストレージディスク

• ロードバランサー

• データベース

- DNS ゾーン
- インスタンス、ディスク、データベースの手動スナップショット

手動スナップショットはタグをサポートしていますが、Lightsail API を使用するか、 AWS CLI スナップショットにタグを付ける必要があります。Lightsail コンソールを使用してタグ付けされたインスタンス、ディスク、またはデータベースの手動スナップショットを作成する場合、手動スナップショットにはソースリソースと同じタグが自動的に割り当てられます。Lightsail コンソールを使用してタグ付けされた手動スナップショットから新しいリソースを作成する場合、これらのタグを編集できます。

自動スナップショットにタグを付けることはできません。

Lightsail スナップショットにタグを付けるにはどうすればよいですか?

Lightsail コンソールは、手動スナップショットにソースリソースと同じタグを自動的にタグ付けします。Lightsail API を使用する場合、 AWS CLI またはスナップショットを作成する場合は、スナップショットのタグを自分で選択できます。

> **Important**
>
> データベースの手動スナップショットのタグ (コスト配分タグ) は現在、請求レポートに含まれません。

キー値タグとキーのみタグの違いは何ですか?

Lightsail タグはキーと値のペアで、インスタンスなどのリソースをさまざまなカテゴリ (プロジェクト:ブログ、プロジェクト:ゲーム、プロジェクト:テストなど) にまたがって整理できます。これにより、リソースの整理、請求レポート、アクセス管理などのすべてのユースケースを全面的に管理できます。Lightsail コンソールでは、リソースにキーのみのタグを付け、コンソールですばやくフィルタリングすることもできます。

連絡先および通知

通知とは何ですか?

インスタンス、データベース、またはロードバランサーのいずれかのメトリクスが指定したしきい値を超えたときに通知するように Lightsail を設定できます。通知は、Lightsail コンソールに表示されるバナー、指定したメールアドレスに送信される メール、または指定した携帯電話番号に

送信される SMS テキストメッセージの形式になります。E メールと SMS テキストメッセージで通知を受けるには、 AWS リージョン リソースを監視したいそれぞれの通知連絡先として E メールアドレスと携帯電話番号を追加する必要があります。通知の詳細については、「通知」を参照してください。

連絡先はいくつ追加できますか?

AWS リージョン リソースを監視したい場所には、メールアドレスと携帯電話番号をそれぞれ1つずつ追加できます。SMS テキストメッセージは、Lightsail AWS リージョンリソースを作成できるすべてのものでサポートされているわけではなく、テキストメッセージを世界の一部の国や地域に送信することはできません。通知の詳細については、「通知」を参照してください。

メトリクスおよびアラーム

メトリクスとは何ですか?

Lightsail は、インスタンス、データベース、ロードバランサーのメトリクスデータをレポートします。一部のメトリクスには、インスタンスの CPU 使用率 (%)、インバウンドおよびアウトバウンドネットワークトラフィックの量、システムおよびインスタンスのエラー数、データベースディスクキューの深さ、データベース空きストレージ容量、ロードバランサーのエラー数、ロードバランサーの応答時間などがあります。メトリクスを使用すると、リソースの信頼性、可用性、パフォーマンスを監視および維持することができます。リソースから定期的にメトリクスデータをモニタリングして収集し、マルチポイント障害が発生した場合に、より簡単にデバッグできるようにします。詳細については、「リソースのメトリクス」を参照してください。

アラームとは何ですか?

Lightsail では、インスタンス、データベース、ロードバランサーのメトリクスを監視するアラームを作成できます。アラームは、指定したしきい値を基準にしたメトリクスの値に基づいて通知するように設定できます。詳細については、「 アラーム」を参照してください。

通知は、Lightsail コンソールに表示されるバナー、メールアドレスに送信される メール、携帯電話番号に送信される SMS テキストメッセージです。通知の詳細については、「通知」を参照してください。

アラームはいくつ追加できますか?

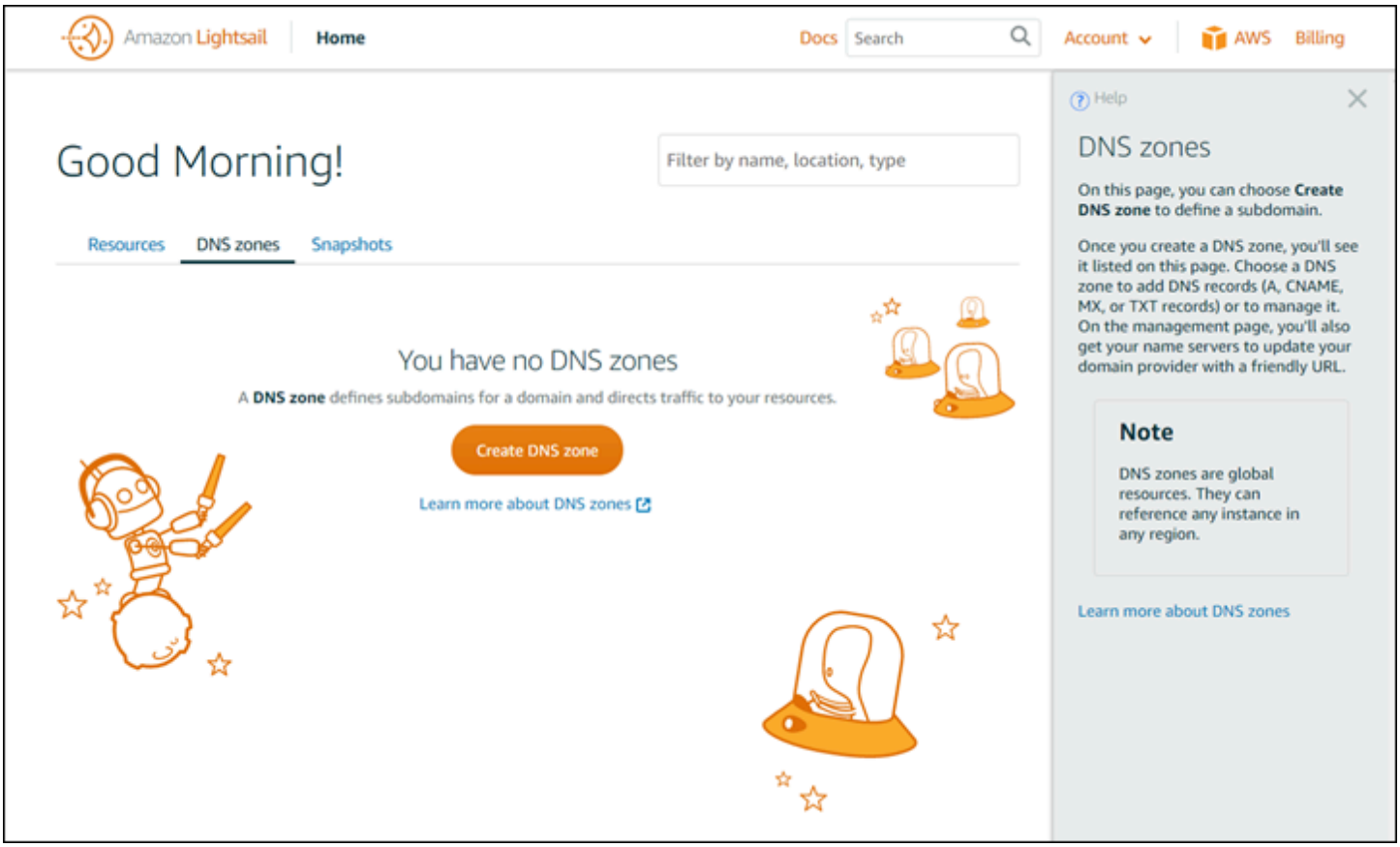
インスタンス、データベース、ロードバランサーに使用できるメトリクスごとに 2 つのアラームを設定できます。詳細については、「 アラーム」を参照してください。

Amazon Lightsail のヘルプを参照する

Amazon Lightsail では、いくつかの方法でヘルプを使用できます。

コンテキスト依存のヘルプパネル

Lightsail には、コンソールの各ページにコンテキスト依存の [ヘルプ] パネルがあり、現在のページに固有の追加のヒントと情報が表示されます。現在のページについて何か質問がある場合はいつでもヘルプパネルを開き、終わったらヘルプパネルを閉じます。ヘルプパネルを開くには、ページにある [ヘルプ] を選択するか、ユーザーインターフェイスの随所にある小さい疑問符を選択します。



本ユーザーガイドについて

Amazon Lightsail ユーザーガイドには、Lightsail での作業に役立つ操作方法のトピックやコンセプトの説明が記載されています。たとえば、インスタンスの作成、インスタンスへの接続、ドメインの管理を行うことができます。

検索の使用

Lightsail の任意のページで、各ページの上部にある検索ボックスを使用して、ドキュメントのトピックを検索できます。ドキュメントの検索ページでもう一度検索すると、検索を絞り込むことができます。

探していたものが見つからなかった場合。申し訳ありません。当社にフィードバックを送信していただければ対処いたします。Lightsail の各ページで [ご質問は? コメント?] を選択して、ご提案のフィードバックを送信できます。折り返しご連絡いたします。

Lightsail の CLI および API の使用

AWS Command Line Interface (AWS CLI) または Lightsail REST API を使用して、Lightsail リソースを作成、読み取り、更新、および削除できます。REST API に加えて、Java、Ruby、JavaScript (Node.js)、Go、PHP、Python、.NET (C#)、C++ などの複数の言語の SDK も提供しています。Lightsail API に関する詳細については、「Lightsail API レファレンス」を参照してください。

> **Note**
>
> LightsailAPI を使用するには、アクセスキーを生成する必要があります。Lightsail API を使用するためのアクセスキーのセットアップ方法を確認してください。

AWS CLI は、Lightsail リソースを使用する場合に役立ちます。AWS AWS CLI で、「`aws lightsail help`」と入力すると、使用可能なコマンドの説明が表示されます。特定の CLI コマンドのヘルプを表示する場合は、コマンド名の後に `help` と入力すると、そのパラメーターと例外の詳細が表示されます。詳細については、「Lightsail CLI リファレンス」を参照してください。

AWS フォーラムおよびその他のコミュニティリソース

AWS のディスカッションフォーラムである AWS フォーラムに質問を投稿することもできます。

翻訳は機械翻訳により提供されています。提供された翻訳内容と英語版の間で齟齬、不一致または矛盾がある場合、英語版が優先します。