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

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すべてのユーザー向けに、Amazon Web Services (AWS) の使用を開始する最も簡単な方法です。これには、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DNSの管理、リソーススナップショット (バックアップ) など、プロジェクトをすばやく起動するために必要なすべてが含まれており、月額料金は低く、予測可能です。

Lightsail には Amazon Lightsail for Research も用意されています。Lightsail for Research を使用すると、学者や研究者は で強力な仮想コンピュータを作成できます 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、Amazon Lightsail for Research ユーザーガイド」を参照してください。

トピック

Lightsail の機能

Lightsail には、次の高レベルの機能があります。

インスタンス

Lightsail は、 のパワーと信頼性によって簡単にセットアップおよびバックアップできる仮想プライベートサーバー (インスタンス) を提供します 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 からインスタンスを管理できますAPI。

インスタンスを作成するときは、 click-to-launch 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 WordPress、Windows、Plesk、、LAMPNginx などの

開発スタックを使用します。すべての Lightsail インスタンスには、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するために使用できるファイアウォールが組み込まれています。詳細はこちら

コンテナ

クラウドで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

ウェブトラフィックをインスタンス全体にルーティングして、ウェブサイトやアプリケーションがトラフィックのバリエーションに対応し、停止から保護し、シームレスな訪問者エクスペリエンスを提供できるようにします。詳細はこちら

マネージドデータベース

Lightsail には、メモリ、処理、ストレージ、転送許容量を含む、完全に設定された MySQL または PostgreSQL データベースプランが用意されています。Lightsail マネージドデータベースを使用すると、仮想サーバーとは別にデータベースを簡単にスケールしたり、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ます。Linux または Windows 仮想サーバー用の高可用性 SSD-backed ストレージを使用すると、ストレージを迅速かつ簡単にスケーリングできます。詳細はこちら

Lightsail Object ストレージバケットを使用すると、いつでもインターネット上のどこからでもオブジェクトを保存および取得できます。クラウドで静的コンテンツを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) ディストリビューションを有効にします CloudFront。世界中のユーザーにコンテンツを簡単に配信するには、プロキシサーバーをセットアップして、ユーザーが地理的に近い場所にウェブサイトにアクセスできるようにし、レイテンシーを短縮します。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セットを使用して、簡単に使用を開始できるようにします。ただし、これらのオプションに制限されているわけではありません。Amazon VPCピアリング AWS を介して、Lightsail プロジェクトを の 90 以上の他のサービスの一部と統合できます。詳細はこちら

Lightsail の詳細については、Amazon Lightsail」を参照してください。

Lightsail とは

Lightsail はすべてのユーザーを対象としています。Lightsail インスタンスのイメージを選択してプロジェクトをジャンプスタートできるため、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費やす必要はありません。

個人プロジェクトに取り組んでいる個人開発者またはホビイストの場合、Lightsail は基本的なクラウドリソースのデプロイと管理に役立ちます。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、すぐに使用を開始できます。

Lightsail には、ベースオペレーティングシステム、、 LEMP (Nginx)LAMP、SQLServer Express などの開発スタック、、Drupal WordPress、Magento 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と、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。Lightsail インスタンスが Lightsail の外部で他の AWS リソースを使用VPCできるように、 をピアリング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高可用性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するようにロードバランサーを設定することもできます。

Lightsail にアクセスする

Lightsail リソースは、次のインターフェイスを使用して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および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ている場合は、 にサインイン AWS Management Console し、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 Command Line Interface

コマンドラインシェルのコマンドを使用して AWS 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 にあります。

AWS Tools for PowerShell

によって公開される機能上に構築された PowerShell モジュールのセット AWS SDK for .NET。Tools for PowerShell を使用すると、 PowerShell コマンドラインから AWS リソースに対するオペレーション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、AWS Tools for PowerShell 「 コマンドレットリファレンス」にあります。

クエリ API

Lightsail はクエリ を提供しますAPI。これらのリクエストは、動HTTP詞 または HTTP POSTおよび という名前のクエリパラメータを使用する GETまたは HTTPSリクエストですAction。Lightsail のAPIアクションの詳細については、「Amazon Lightsail リファレンス」の「アクション」を参照してください。 Amazon Lightsail API

AWS SDKs

HTTP または 経由でリクエストを送信するAPIsのではなく、言語固有の を使用してアプリケーションを構築する場合はHTTPS、 はソフトウェアデベロッパー向けにライブラリ、サンプルコード、チュートリアル、その他のリソース AWS を提供します。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、「 で構築するツール AWS」を参照してください。

Lightsail の使用を開始する

Lightsail を使用するように をセットアップしたら、インスタンスを起動、接続、クリーンアップLightsail での仮想プライベートサーバーの開始方法する手順を実行できます。

関連サービス

Lightsail を使用して、インスタンスやディスクなどの Lightsail リソースを直接プロビジョニングできます。さらに、次のような他の AWS 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、つまり文字通り Amazon のデータセンター内のサーバーを提供します。Lightsail と Amazon を比較するにはEC2、Amazon LightsailEC2」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使用できる Amazon EC2インスタンスの数が正しいことを確認するのに役立ちます。
- 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で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

AWS ユースケースの見積りを作成するには、 を使用します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 アカウントの請求の詳細については、AWS 「請求とコスト管理ユーザーガイド」を参照してください。

AWS 請求、アカウント、イベントに関するご質問は、 AWS サポート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。

Lightsail のセットアップ AWS アカウント と管理ユーザー

新規の AWS お客様は、Amazon Lightsail の使用を開始する前に、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してください。これらのセットアップ手順では、 AWS Identity and Access Management （ IAM) サービスを使用します。の詳細についてはIAM、「 IAMユーザーガイド」を参照してください。

にサインアップする AWS アカウント

がない場合は AWS アカウント、次の手順を実行して作成します。

にサインアップするには AWS アカウント

1. https://portal.aws.amazon.com/billing/サインアップ 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、電話キーパッドで検証コードを入力するように求められます。

   にサインアップすると AWS アカウント、 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 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、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。

AWS サインアッププロセスが完了すると、 から確認メールが送信されます。https://aws.amazon.com/ の アカウント] をクリックして、いつでもアカウントの現在のアクティビティを表示し、アカウントを管理することができます。

管理アクセスを持つユーザーを作成する

にサインアップしたら AWS アカウント、 を保護し AWS アカウントのルートユーザー、 を有効にして AWS IAM Identity Center、日常的なタスクにルートユーザーを使用しないように管理ユーザーを作成します。

のセキュリティ保護 AWS アカウントのルートユーザー

1. ルートユーザーを選択し、 AWS アカウント E メールアドレスを入力して、アカウント所有者AWS Management Consoleとして にサインインします。次のページでパスワードを入力します。

   ルートユーザーを使用してサインインする方法については、AWS サインイン ユーザーガイドのルートユーザーとしてサインインするを参照してください。

2. ルートユーザーの多要素認証 (MFA) を有効にします。

   手順については、「 ユーザーガイド」の AWS アカウント 「ルートユーザーの仮想MFAデバイスを有効にする (コンソール) IAM」を参照してください。

管理アクセスを持つユーザーを作成する

1. IAM Identity Center を有効にします。

   手順については、「AWS IAM Identity Center ユーザーガイド」の「AWS IAM Identity Centerの有効化」を参照してください。

2. IAM Identity Center で、ユーザーに管理アクセス権を付与します。

   を ID ソース IAM アイデンティティセンターディレクトリ として使用する方法のチュートリアルについては、「 ユーザーガイド」の「デフォルト でユーザーアクセスを設定する IAM アイデンティティセンターディレクトリAWS IAM Identity Center 」を参照してください。

管理アクセス権を持つユーザーとしてサインインする

- IAM Identity Center ユーザーでサインインするには、IAMIdentity Center ユーザーの作成時に E メールアドレスにURL送信されたサインインを使用します。

  IAM Identity Center ユーザーを使用してサインインする方法については、「 ユーザーガイド」の AWS 「 アクセスポータルにサインインする」を参照してください。 AWS サインイン

追加のユーザーにアクセス権を割り当てる

1. IAM Identity Center で、最小特権のアクセス許可を適用するベストプラクティスに従うアクセス許可セットを作成します。

手順については、「AWS IAM Identity Center ユーザーガイド」の「権限設定を作成する」を参照してください。

2. グループにユーザーを割り当て、そのグループにシングルサインオンアクセス権を割り当てます。

   手順については、「AWS IAM Identity Center ユーザーガイド」の「グループの参加」を参照してください。

Lightsail での仮想プライベートサーバーの開始方法

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

初めて AWS のお客様は、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、「Lightsail のセットアップ AWS アカウント と管理ユーザー」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、「Lightsail インスタンスのブループリントサービスを確認する」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. 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 とアベイラビリティーゾーン) を選択します。レイテンシーを短縮するために、物理的な場所に最も AWS リージョン 近い 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。

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Lightsail インスタンスのブループリントサービスを確認する。

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のみの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 のみIPv6のネットワークをサポートしていません。専用ネットワークをサポートするブループリントを確認するには、IPv6「」を参照してくださいLightsail インスタンスのブループリントサービスを確認する。

   5 USD の USD Lightsail プランを 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:

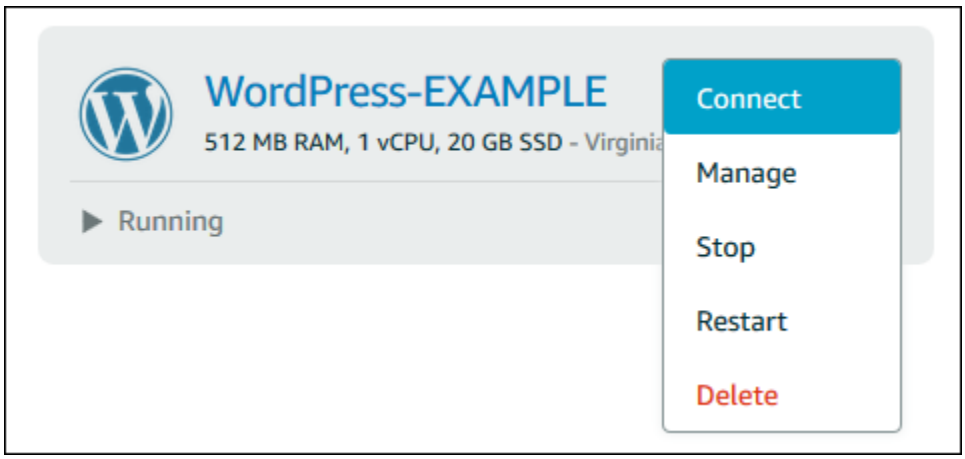
   - Lightsail アカウントの各 AWS リージョン 内で一意である必要があります。
   - 2〜255 文字を使用する必要があります。
   - 先頭と末尾は英数字または数字を使用する必要があります。
   - 英数字、数字、ピリオド、ダッシュ、アンダースコアを使用することができます。

7. [インスタンスの作成] を選択します。

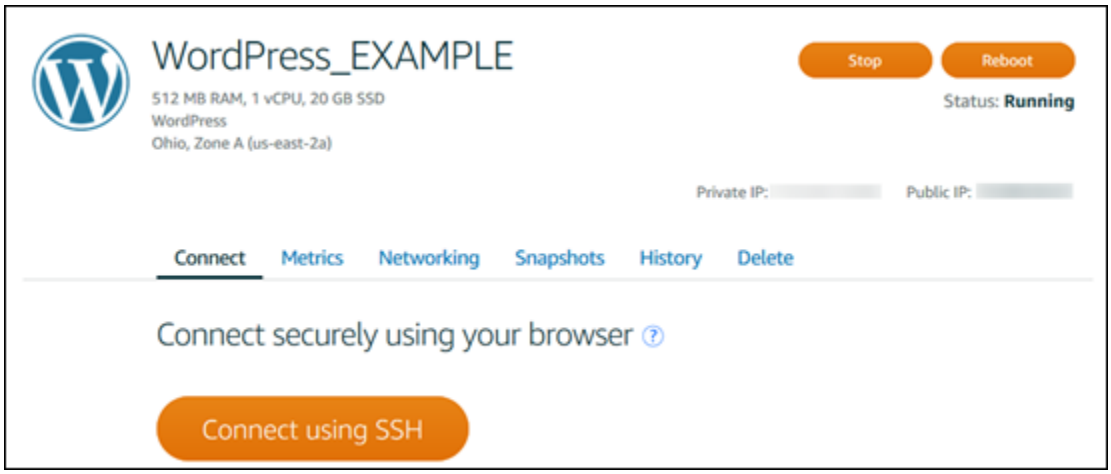
Lightsail インスタンスは数分以内に準備でき、接続できます。

ステップ 3: インスタンスに接続する

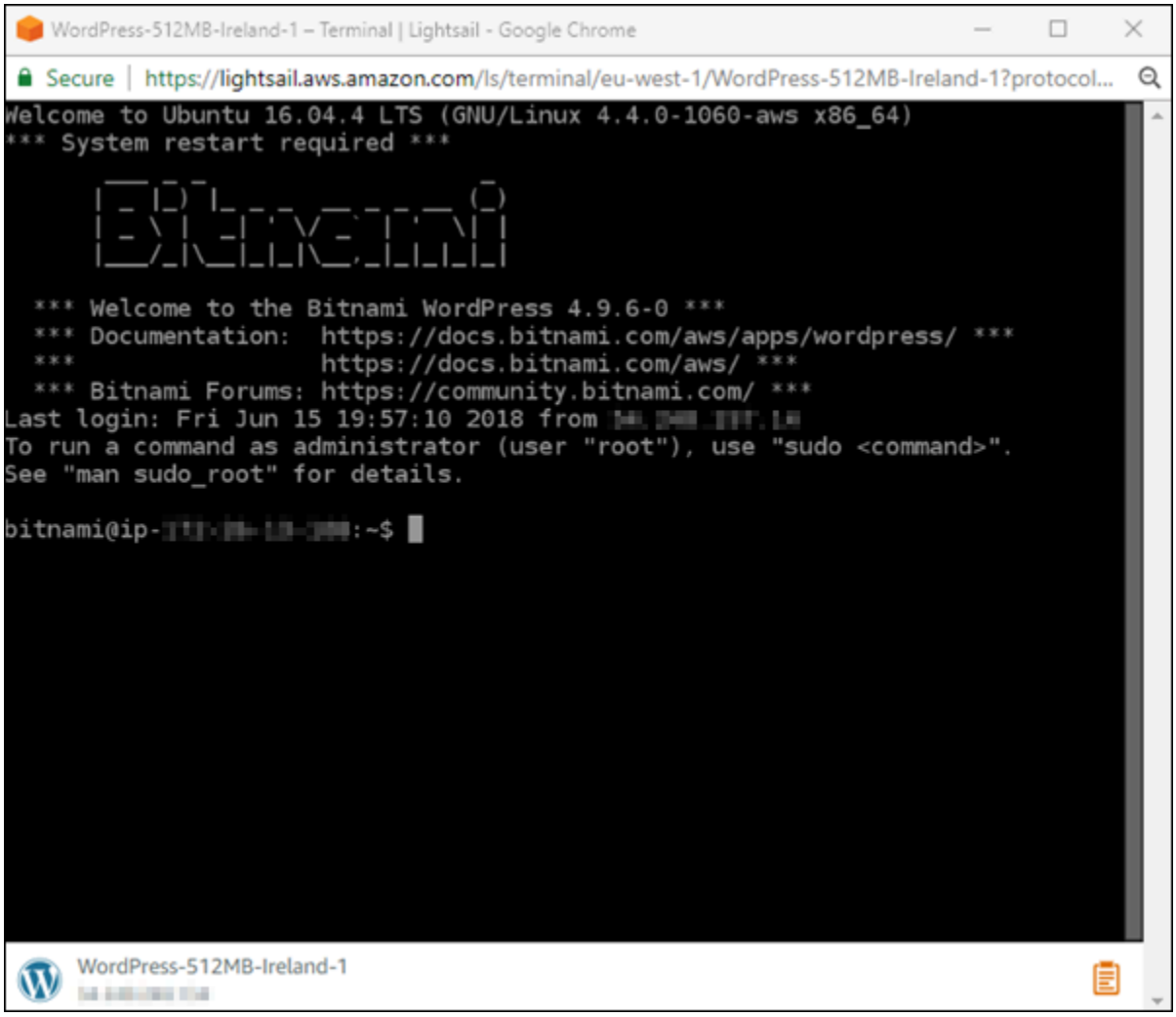
1. Lightsail のホームページで、インスタンス名の右側にあるメニューを選択し、接続を選択します。



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、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 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、「スナップショットを使用して Linux/Unix Lightsail インスタンスをバックアップする」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

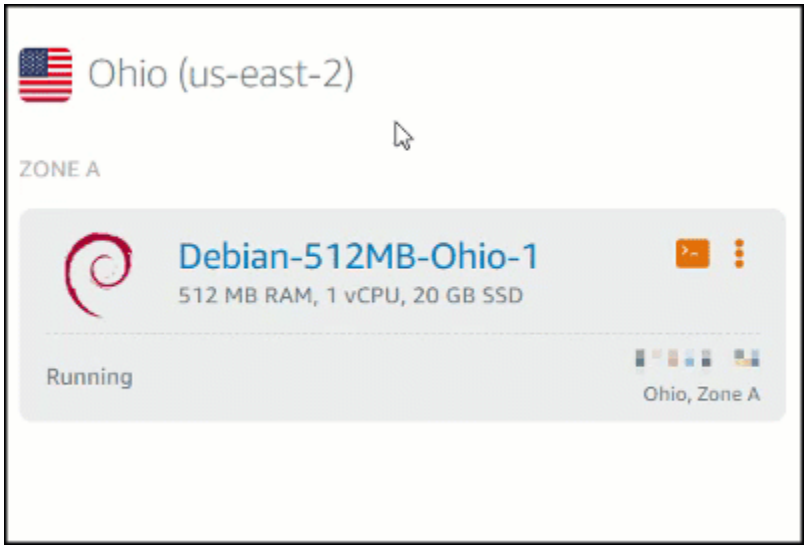
このチュートリアル用に作成したインスタンスを使用して操作した後に、削除することができます。これにより、不要になったインスタンスに対する料金は発生しなくなります。

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 の料金」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、「スナップショットを使用して Linux/Unix Lightsail インスタンスをバックアップする」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. 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。

- Lightsail でアプリケーションを使用して Linux/Unix インスタンスを作成する
- Lightsail で Windows Server インスタンスを作成する

Lightsail の仮想プライベートサーバーインスタンス

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、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 インスタンスのブループリントサービスを確認する
- Lightsail のファイアウォールでインスタンストラフィックを制御する
- 最適なパフォーマンスを得るための Lightsail インスタンスバーストの検出
- Lightsail インスタンスに接続して管理する
- Lightsail インスタンスを削除する
- SSH キーペアを管理し、Lightsail インスタンスに接続する
- Lightsail でインスタンスメタデータサービス (IMDS) とユーザーデータにアクセスする

Lightsail インスタンスを作成する

このセクションでは、Amazon Lightsail でのインスタンスの作成に関連する以下のトピックについて説明します。

トピック

- Lightsail でアプリケーションを使用して Linux/Unix インスタンスを作成する
- Lightsail で Windows Server インスタンスを作成する

Lightsail でアプリケーションを使用して Linux/Unix インスタンスを作成する

などのアプリケーション WordPress または などの開発スタックを実行する Linux/Unix ベースの Amazon Lightsail インスタンス (仮想プライベートサーバー) を作成しますLAMP。インスタンスの実行が開始されたら、Lightsail を離れSSHることなく、 を介して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」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 と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」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のみの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 のみIPv6のネットワークをサポートしていません。専用ネットワークをサポートするブループリントを確認するには、IPv6「」を参照してくださいLightsail インスタンスのブループリントサービスを確認する。

   5 USD の USD Lightsail プランを 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 の料金」ページのAWS 「無料利用枠」を参照してください。

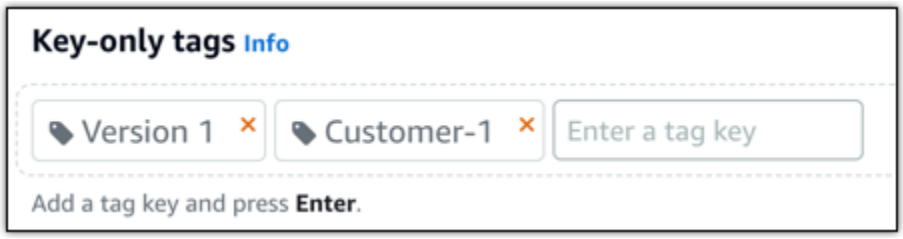
7. インスタンスの名前を入力します。

   リソース名:

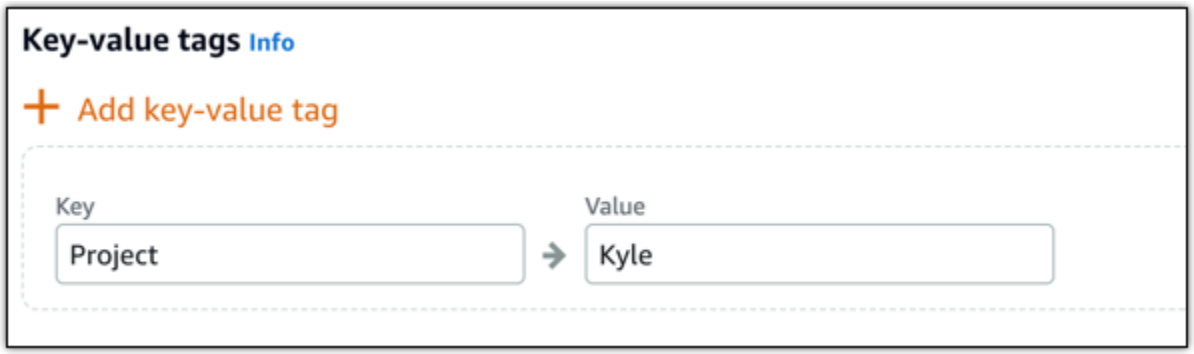
   - Lightsail アカウントの各 AWS リージョン 内で一意である必要があります。
   - 2〜255 文字を使用する必要があります。
   - 先頭と末尾は英数字または数字を使用する必要があります。
   - 英数字、数字、ピリオド、ダッシュ、アンダースコアを使用することができます。

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。

   - [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。

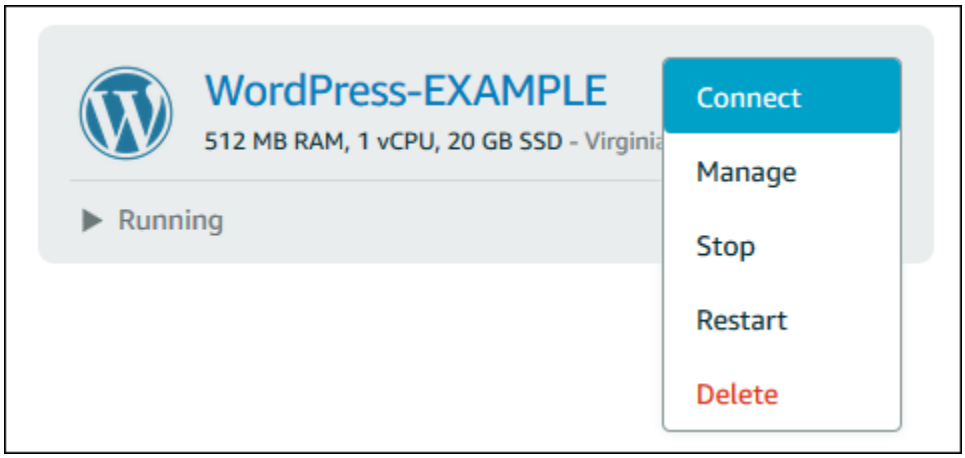
9. [インスタンスの作成] を選択します。

   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SSHのセットアップ」を参照してください。

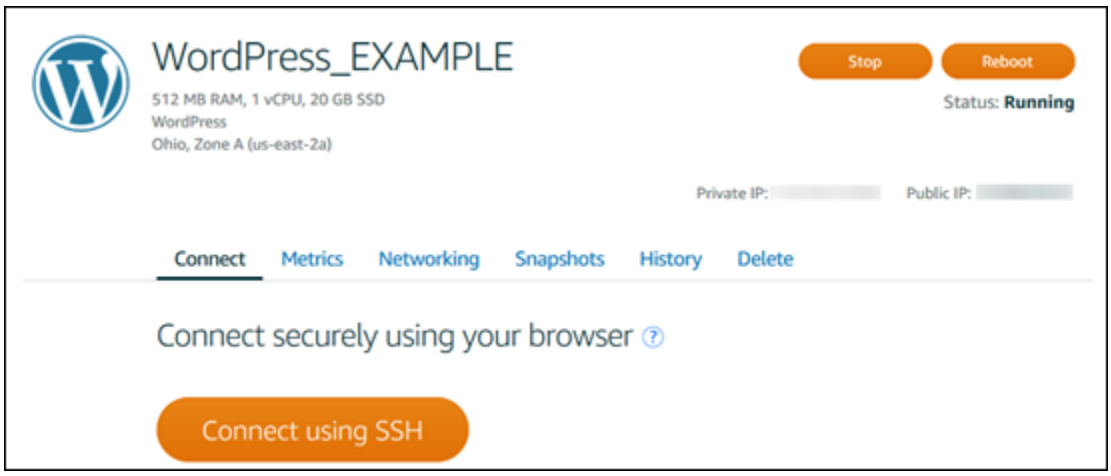
Lightsail インスタンスは数分以内に準備が完了しSSH、Lightsail を離れることなく 経由で接続できます。

インスタンスへの接続

1. 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。

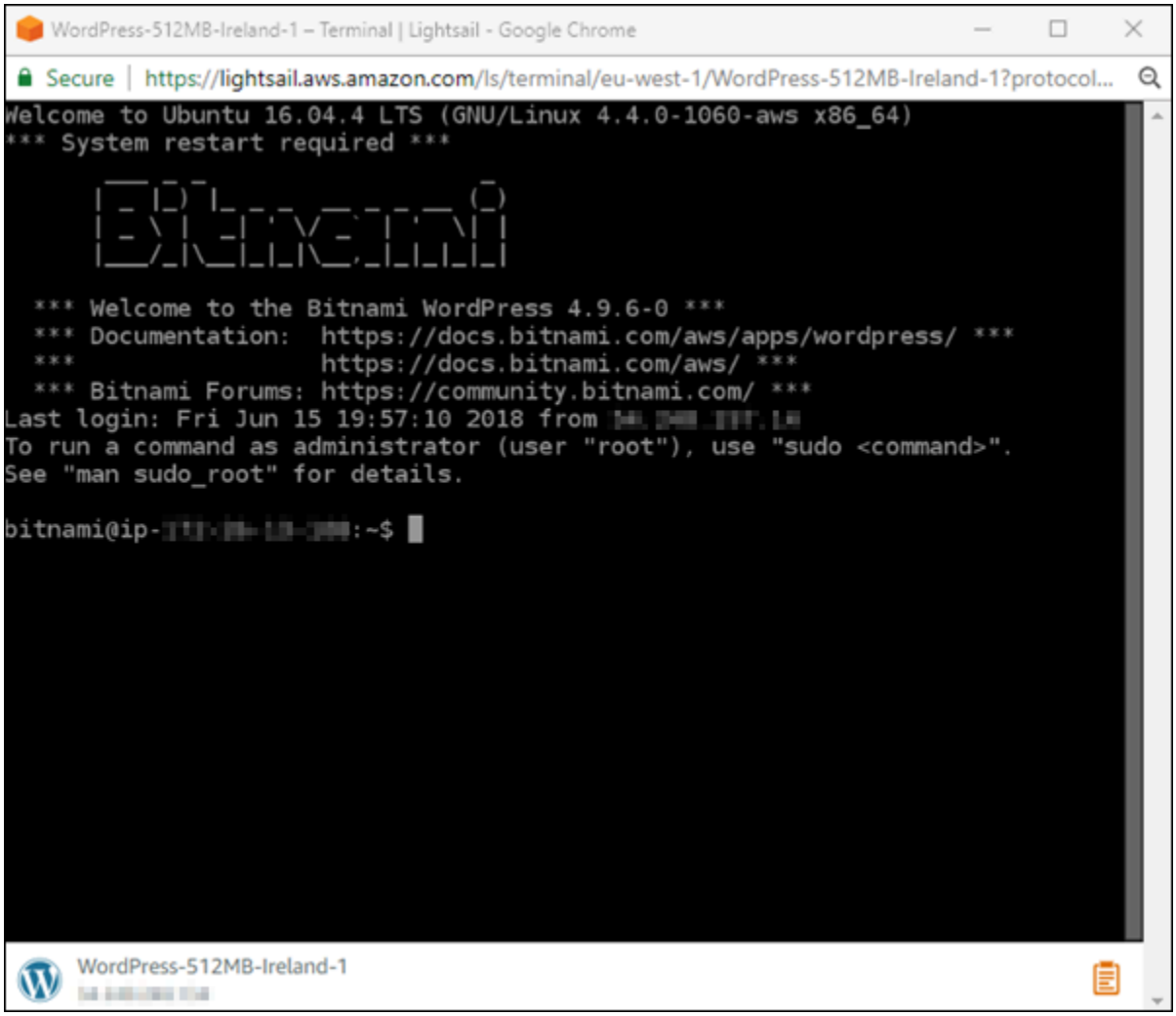


   または、インスタンス管理ページを開き、[接続] タブを選択することもできます。



> **Note**
>
> Pu 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TTY、「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、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。
- 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ように、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 で Windows Server インスタンスを作成する

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します。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。さらに、SQLServer 2022、2019、および 2016 Express で事前設定された設計図があります。

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。

での Windows Server AWS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 Windows ベースのソリューションを実行できます。一般的な Windows のユースケースには、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のホスティング、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、および Windows ソフトウェアを必要とするその他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 2016 または Windows Server 2019 を実行する必要がある場合を除き、Windows Server 2022 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、 の AWS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 のユースケースには、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 ソフトウェアを必要とするその他のアプリケーションなどがあります。

Windows Server 2019 イメージの詳細

Windows Server 2016

何らかの理由で Windows Server 2016 または Windows Server 2019 を実行する必要がある場合を除き、Windows Server 2022 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、 のAWS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 のユースケースには、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 ソフトウェアを必要とするその他のアプリケーションなどがあります。

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 イメージの詳細はこちら

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 イメージの詳細はこちら

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 イメージの詳細はこちら

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 AWS リージョン を作成する を選択します。

   例えば、Ohio (us-east-2) と指定します。
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 ブループリントを選択するには、Apps + 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のみの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 のみIPv6のネットワークをサポートしていません。専用ネットワークをサポートするブループリントを確認するには、IPv6「」を参照してくださいLightsail インスタンスのブループリントサービスを確認する。

   プランには、低コスト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。

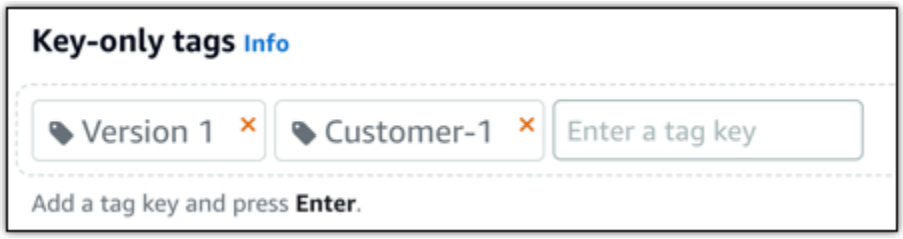
   > **Note**
   >
   > 設計図によっては、一部のインスタンスプランを使用できません。例えば、SQLServer Express ブループリントで 2 つの最小プランを使用することはできません。少なくとも、2 GB RAMと 50 GB のプランを使用するかSSD、大きいプランのいずれかを選択する必要があります。
7. インスタンスの名前を入力します。

   リソース名:

   - 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。

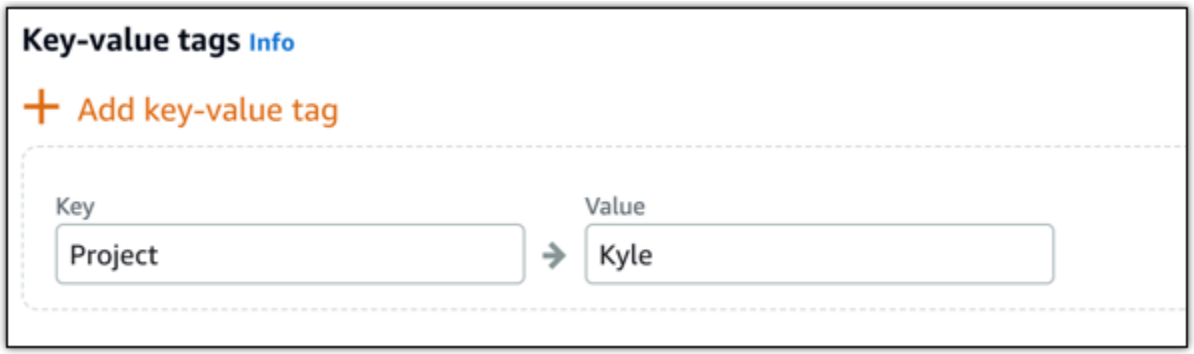
8. 以下のいずれかのオプションを選択して、インスタンスにタグを追加します。

   - [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。

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 で動作する AWS Command Line Interface ように を設定する Amazon Lightsail」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、Lightsail リソースを作成する AWS リージョン `aws configure`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 USD Windows Server 2022 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_2022 --bundle-id medium_win_3_0
```

コマンドで、*InstanceName* 新しいインスタンスの名前を入力します。

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について詳しくは、こちらをご覧ください。

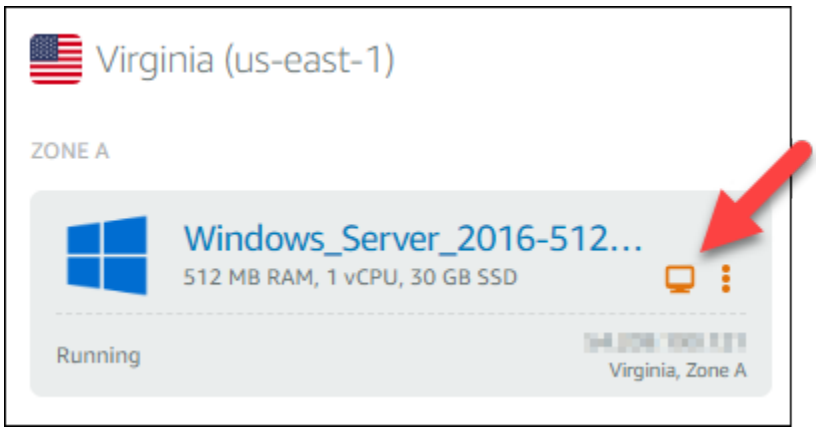
インスタンスへの接続

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. ホームページで、インスタンスの横にある Connect using RDP アイコンを選択します。

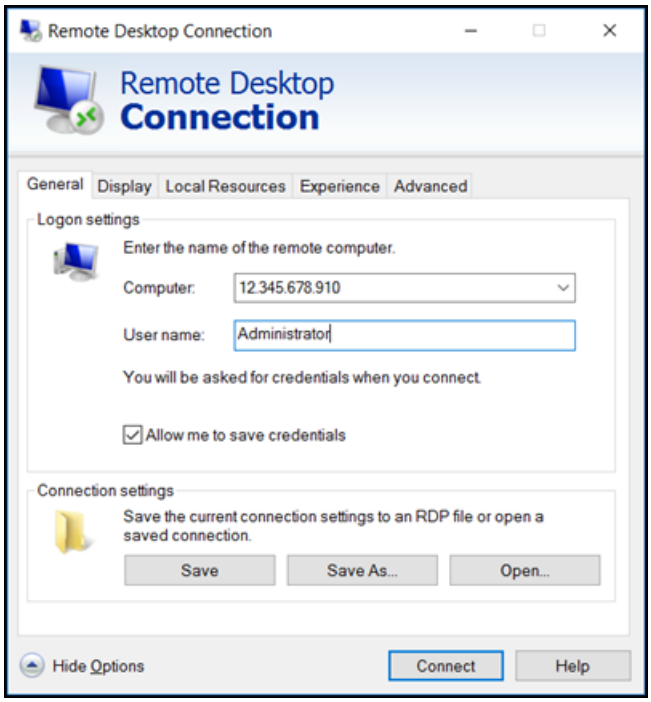


2. または、ショートカットメニューやインスタンス管理ページからインスタンスに接続することもできます。

独自のRDPクライアントを使用して接続するには

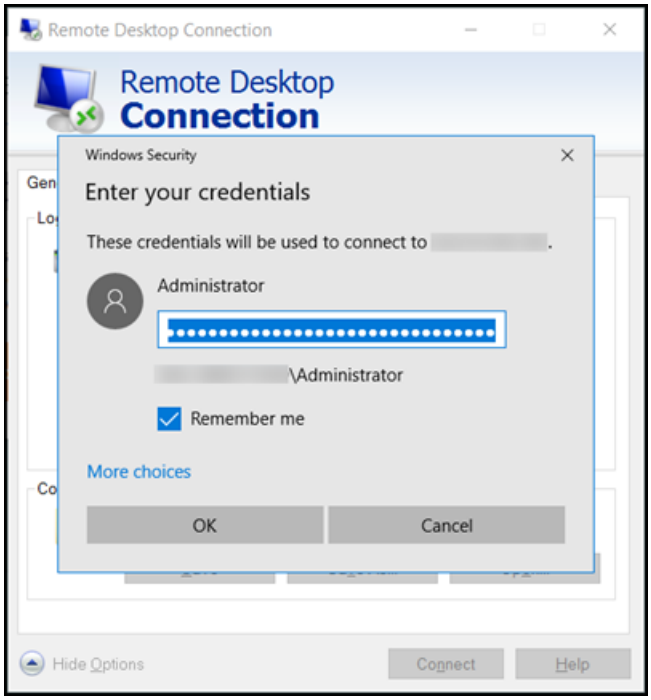
1. IP アドレスを取得するには、Lightsail ホームページに移動します。
2. IP アドレスをクリップボードにコピーします。
3. Windows でリモートデスクトップ接続などのRDPクライアントを開きます。
4. IP アドレスを [コンピューター] フィールドに貼り付けます。

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する (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] を選択します。
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。

step-by-step 手順に従って、Amazon Linux、Ubuntu、Debian、または Windows Server 2022、2019、2016 などの Windows Server オペレーティングシステムなどの Linux および Unix ディストリビューションを実行するインスタンスを作成します。

Linux および Unix インスタンスでは、、 WordPress、、 LAMPなどのさまざまなアプリケーションブループリントから選択したりLEMP、オペレーティングシステムのみを選択したりできます。Windows Server インスタンスの場合、Windows Server ブループリントまたは SQL Server Express ブループリントから選択できます。

このガイドでは、 AWS リージョン とアベイラビリティーゾーンの選択、必要なコンピューティングリソースとストレージリソースを含むインスタンスプラン (バンドル) の選択、 IPv4 や などのネットワークオプションの設定IPv6、インスタンスの命名、タグの追加について説明します。インスタンスを作成したら、Lightsail 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するか、提供された接続の詳細で独自の SSHまたはRDPクライアントを使用できます。このガイドに従うことで、特定の要件に合わせて、Linux、Unix、または Windows Server インスタンスを Lightsail ですばやく起動してアクセスできます。

Lightsail インスタンスのブループリントサービスを確認する

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。

オペレーティングシステム

Lightsail には、Linux/Unix ベースまたは Windows ベースのオペレーティングシステムがいくつか用意されています。

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 Windows ベースのソリューションを実行できます。一般的な Windows のユースケースには、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のホスティング、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、および Windows ソフトウェアを必要とするその他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のみのインスタンスプランと互換性があります。

Windows Server 2022 の詳細をご覧ください。

Windows Server 2019

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 Windows ベースのソリューションを実行できます。一般的な Windows のユースケースには、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のホスティング、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、および Windows ソフトウェアを必要とするその他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のみのインスタンスプランと互換性があります。

Windows Server 2019 の詳細をご覧ください。

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、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 のユースケースには、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のホスティング、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、および Windows ソフトウェアを必要とするその他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のみのインスタンスプランと互換性があります。

Windows Server 2016 の詳細をご覧ください。

Amazon Linux 2023

Amazon Linux 2023 (AL2023) は次世代の Amazon Linux であり、 の汎用ワークロードに最適です AWS。AL2023 は、一般公開されてから 5 年間サポートされます。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、更新をいつどのように吸収するかを制御できます。AL2023 には、頻繁に更新を行う機能もあり、コンプライアンスのニーズを満たすのに役立つ機能も付属しています。

AL2023 から起動された Lightsail インスタンスでは、インスタンスメタデータサービスバージョン 2 (IMDSv2) がデフォルトで適用されます。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のみのインスタンスプランと互換性があります。

Amazon Linux 2023 の詳細をご覧ください。

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 2FAQs」を参照してください。

このブループリントは、Lightsail IPv6のみの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および管理する永続的なエンタープライズ Linux ディストリビューションであり、長期的な安定性に焦点を当てており、堅牢な本番稼働用プラットフォームを提供します。 AlmaLinux は 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のみのインスタンスプランと互換性があります。

AlmaLinux OS 9 の詳細をご覧ください。

CentOS ストリーム 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) 開発の直前を追跡する継続的な配信ディストリビューションで、Fedora Linux と の間の中間ストリームとして配置されています RHEL。これは、機能的に RHELと互換性があるように設計されており、安定し、予測可能で、管理しやすく、再現可能な Linux 環境を提供します。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のみのインスタンスプランと互換性があります。

詳細については、CentOS Stream ウェブサイトを参照してください。

Debian 11、12

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のみのインスタンスプランと互換性があります。

詳細については、Debian ウェブサイトを参照してください。

無料BSD 13

FreeBSD は、サーバー、デスクトップ、および組み込みシステムに電力を供給するために使用されるオペレーティングシステムです。カリフォルニア大学バークレー校でUNIX開発された BSD

のバージョンである から派生した FreeBSD は、大規模なコミュニティによって 30 年以上にわたって継続的に開発されてきました。pf BSDファイアウォール、Capsicum および クラウドABI機能フレームワーク、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の のネットワーク、セキュリティ、ストレージ、モニタリング機能を無料で利用できるため、最もビジーなウェブサイトや最も広範な組み込みネットワークおよびストレージシステム向けに、Free BSD が最適なプラットフォームとなっています。サポート終了情報については、 無料BSD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のみのインスタンスプランと互換性があります。

詳細については、 無料BSDウェブサイトを参照してください。

開くSUSE 15

オープンSUSEディストリビューションは、安定した使いやすい完全な汎用 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。サポート終了情報については、オープンSUSE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のみのインスタンスプランと互換性があります。

詳細については、オープンSUSEウェブサイトを参照してください。

Ubuntu 20、22

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 LibreOffice、Thunderbird、および Transmission を含む幅広い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 () を使用して、Evolution、GIMP、Pidgin、Sinaptic など、多く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apt-get。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のみのインスタンスプランと互換性があります。

詳細については、Ubuntu ウェブサイトを参照してください。

データベースアプリケーション

Lightsail では、次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 Server 2022 のベース OS で実行されます。

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

SQL Server 2022 Express の詳細をご覧ください。

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 Server 2022 のベース OS で実行されます。

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

SQL Server 2019 Express の詳細をご覧ください。

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 Server 2016 のベース OS で実行されます。

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

SQL Server 2016 Express の詳細をご覧ください。

CMS アプリケーション

Lightsail では、次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

Bitnami WordPress は、Lightsail WordPress で実行するための事前設定された ready-to-use イメージです。 WordPress は、ブログやウェブサイトを構築するための一般的なウェブ公開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 は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ルックアンドフィールを変更できます。既存の無料または商用 WordPress のテーマを使用することもできます。 WordPress は、World Wide Web Consortium (W3C) の標準に完全に準拠しています。

Lightsail WordPress で を起動して設定する

の詳細については、Bitnami ウェブサイトWordPressを参照してください。

WordPress Bitnami によって認定されたマルチサイト

WordPress マルチサイトを使用すると、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および管理できます。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 Multisite は、大学、企業、政府機関などの組織にとって最適であり、多くの人が中央管理者に全体的な管理を与えながら、独自のウェブサイトをホストできるようにする必要があります。

Lightsail でマルチサイトを設定する WordPress

WordPress マルチサイトの詳細については、Bitnami ウェブサイトを参照してください。

cPanel & WebHost Manager (WHM)

cPanel & WHMは Linux OS 用に構築されたツールのスイートで、シンプルなグラフィカルユーザーインターフェイスを使用してウェブホスティングタスクを自動化できます。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。

Lightsail で cPanel & WHM を使用してウェブサイト、E メール、サービスをホストする

cPanel および WHM の詳細については、 cPanelウェブサイトを参照してください。

PrestaShop Bitnami によってパッケージ化

PrestaShop は、世界でも最も多用性の高い 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ばやく起動して稼働させるように設計されており、テーマが事前設定されているため、ライブコンフィギュレータと一緒にほぼすぐに販売を開始してサイトの外観を簡単にカスタマイズできます。 PrestaShop マルチストアサポート、カスタマイズ可能な URLs、複数の支払いゲートウェイオプション (PayPal および Stripe を含む) 、Amazon、eBay、Facebook などのマーケットプレイス統合を備えています。

Lightsail で PrestaShop ウェブサイトを設定する

の詳細については、 PrestaShopウェブサイトPrestaShopを参照してください。

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。

Lightsail に Ghost ウェブサイトをデプロイする

Bitnami Ghost の詳細については、Bitnami ウェブサイトを参照してください。

Joomla! (Bitnami によってパッケージ化)

Bitnami Joomla! は、Lightsail で Joomla! を実行するための事前設定された ready-to-use イメージです。Joomla! は、さまざまなウェブサイトやポータルを構築するためにCMS使用できる 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 、LDAPOpenID など、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。インプレッション番号、特別な URLsなどの設定を含むメトリクスを追跡できます。

Lightsail で Joomla! の使用を開始する

Joomla! の詳細については、Bitnami ウェブサイトを参照してください。

Drupal (Bitnami によってパッケージ化)

Bitnami Drupal は、Lightsail で Drupal を実行するための事前設定された ready-to-use 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、ハイパフォーマンス向けに構築されており、多くのサーバーにスケーラブルで、REST、、JSONSOAP、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。

Lightsail で Drupal ウェブサイトを設定およびカスタマイズする

Drupal の詳細については、Bitnami ウェブサイトを参照してください。

アプリケーションスタックとサーバー

Lightsail には、さまざまな開発プロジェクト用の 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 Bitnami によってパッケージ化された スタック (PHP 8)

Bitnami LAMPスタックは、PHPアプリケーションの開発とデプロイを簡素化します。これには、Apache、My SQL、PHP、および ready-to-run 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、バンドルされた各コンポーネントの最新の安定版リリースに常にアクセスできます。

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

Lightsail で LAMP スタックを設定する

Bitnami LAMPスタックの詳細については、Bitnami ウェブサイトを参照してください。

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、Python、Django、My SQL、および Apache ready-to-run のバージョンが含まれています。

Bitnami Django スタックの詳細については、Bitnami ウェブサイトを参照してください。

Node.js (Bitnami によってパッケージ化)

Bitnami Node.js は、Lightsail で Node.js を実行するための事前設定された ready-to-use イメージです。Node.js は、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するため

の Chrome の JavaScript ランタイム上に構築されたプラットフォームです。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。

Lightsail で Node.js の使用を開始する

Node.js スタックの詳細については、Bitnami ウェブサイトを参照してください。

MEAN Bitnami によってパッケージ化された スタック

Bitnami MEANスタックは、ワンクリックでデプロイできる MongoDB と Node.js の完全な開発環境を提供します。これには、MongoDB 、Express、Angular、Node.js、Git、、PHPおよび の最新安定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のみのインスタンスプランと互換性があります。

MEAN スタックの詳細については、Bitnami ウェブサイトを参照してください。

GitLab Bitnami によってパッケージ化された CE

Bitnami GitLab Community Edition (CE) は、Lightsail GitLab で実行するための事前設定された ready-to-use イメージです。 GitLab は、Ruby on Rails に基づく高速で安全なセルフホスト型の Git 管理ソフトウェアです。 GitLab CI (別売り) は、Git および と緊密に統合されたオープンソースの Continuous Integration (CI) サーバーです GitLab。

では GitLab、コードを独自のサーバーで安全に保ち、リポジトリ、ユーザー、アクセス許可を管理します。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。

Lightsail で GitLab CE インスタンスをセットアップおよび設定する

GitLab スタックの詳細については、Bitnami ウェブサイトを参照してください。

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

Bitnami NGINX スタックはPHP、ワンクリックで起動できる完全な 、My SQL、および NGINX開発環境を提供します。また phpMyAdmin、、SQLite、、高速 ImageMagick、MemcacheCGI、GD、、CURL、PEARPECL、およびその他のコンポーネントもバンドルされます。

NGINX は非同期サーバーであり、主な利点はスケーラビリティです。NGINX スタックは LEMP (Linux、、My NGINX、) とも呼ばれSQLますPHP。

Lightsail で Nginx ウェブサーバーをデプロイして管理する

Nginx スタックの詳細については、Bitnami ウェブサイトを参照してください。

Ubuntu の Plesk ホスティングスタック

Plesk を搭載したホスティングスタックAWSを使用して、Lightsail 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、すべての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に加えて、グラフィカルユーザーインターフェイスでの WordPress オートメーションが含まれます。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 スタックの詳細については、Plesk ウェブサイトを参照してください。

e コマース アプリケーション

Lightsail には現在、e コマースアプリケーションイメージ Magento が 1 つあります。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)

Bitnami Magento は、Lightsail で Magento を実行するための事前設定された ready-to-use 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。

Lightsail で Magento をセットアップおよび設定する

Magento スタックの詳細については、Bitnami ウェブサイトを参照してください。

プロジェクト管理アプリケーション

Lightsail には現在、プロジェクト管理アプリケーションイメージ Redmine 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。

Bitnami によってパッケージ化された Redmine

Bitnami Redmine は、Lightsail で Redmine を実行するための事前設定された ready-to-use 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。これには、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統合などのサポートが含まれます。

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

Lightsail で Redmine インスタンスを設定して保護する

Redmine スタックの詳細については、Bitnami ウェブサイトを参照してください。

Lightsail のファイアウォールでインスタンストラフィックを制御する

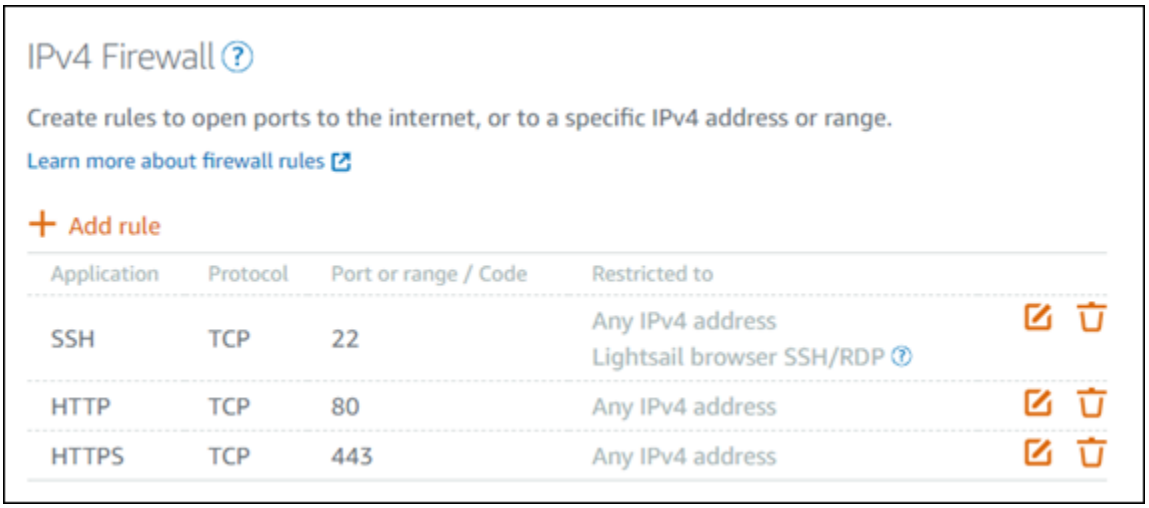
Amazon Lightsail コンソールのファイアウォールは、パブリック IP アドレスを介してインスタンスに接続できるトラフィックを制御する仮想ファイアウォールとして機能します。Lightsail で作成する各インスタンスには、2 つのファイアウォールがあります。1 つはIPv4アドレス用、もう 1 つはIPv6アドレス用です。各ファイアウォールには、インスタンスに着信するトラフィックをフィルタリングする一連のルールが含まれています。両方のファイアウォールは互いに独立しています。 IPv4とに対してファイアウォールルールを個別に設定する必要がありますIPv6。インスタンスのファイアウォールは、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。

Lightsail ファイアウォール

各 Lightsail インスタンスには 2 つのファイアウォールがあります。1 つはIPv4アドレス用、もう 1 つはIPv6アドレス用です。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、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。両方のファイアウォールは互いに独立していることに注意してください。ファイアウォールルールは IPv4と に対して個別に設定する必要がありますIPv6。

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、次の例 (の場合) に示すように、使用するプロトコル、開くポート、インスタンスへの接続が許可されている IPv4および IPv6 アドレスを指定しますIPv4。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。インスタンスのプライベート IP アドレスを介して流れるトラフィックには影響しません。このトラフィックは、同じ のアカウント内の Lightsail リソース AWS リージョン、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があります 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。このルールがインスタンスのファイアウォールで既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 、 AWS Command Line Interface （ AWS CLI ）API、または AWS を使用しますSDK。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、ファイアウォールには、IPv4トラフィック用に 60 個のインバウンドルールとIPv6、トラフィック用に 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、接続を確立する必要があるコンピュータの IP アドレスからのみポート 22 TCP経由で を有効にするファイアウォールルールを設定することで、SSHクライアントがインスタンスに接続し、サーバーで管理タスクを実行できるようにすることもできます。この場合、SSHインスタンスへの接続を確立する IP アドレスを許可したくないとします。許可すると、インスタンスのセキュリティリスクにつながる可能性があるためです。

> **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。例えば、IP アドレス 192.0.2.1 からTCPポート 22 (SSH) へのアクセスを許可するルールを追加するとします。次に、すべてのユーザーから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。その結果、すべてのユーザーがTCPポート 22 にアクセスできます。

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、データ交換が完了するまで、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および維持するために使用します。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、送信されたデータが欠落していないこと、および送信されたすべてのデータが目的の受信者に

送信され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。

- User Datagram Protocol (UDP) 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低レイテンシーおよび損失許容接続を確立するために使用されます。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。
- Internet Control Message Protocol (ICMP) は、主に、データが意図した宛先にタイムリーに到達しているかどうかを判断するなど、ネットワーク通信の問題を診断するために使用されます。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、ルールは を使用するように自動的に設定されますICMPv6。詳細については、Wikipedia の「Internet Control Message Protocol for IPv6 」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。

- File Transfer Protocol (FTP) を介したデータ転送では、ポート 20 が使用されます。
- に対するコマンドコントロールはポート 21 FTPを使用します。
- Secure Shell (SSH) 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。
- Simple Mail Transfer Protocol (SMTP) の E メールルーティングでは、ポート 25 を使用します。

> **Important**
>
> インスタンスSMTPで を有効にするには、インスタンスDNSのリバース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 で制限される場合があります。詳細については、「Amazon Lightsail インスタンスの DNS E メールサーバーのリバースの設定 Amazon Lightsail」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ハイパーテキスト転送プロトコル (HTTP) 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 Post Office Protocol (POP3) は、ポート 110 を使用します。
- Network News Transfer Protocol (NNTP) はポート 119 を使用します。
- Network Time Protocol (NTP) はポート 123 を使用します。
- デジタルメールの管理に使用されるインターネットメッセージアクセスプロトコル (IMAP) は、ポート 143 を使用します。
- Simple Network Management Protocol (SNMP) はポート 161 を使用します。
- HTTP ウェブサイトへの暗号化された接続を確立するためにウェブブラウザで TLS/SSL HTTPを介して (HTTPS) を保護するには、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、、、 などのサービスに使用する一般的なプロトコルSSHRDPHTTPとポートを検索する必要はありません。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 、 AWS Command Line Interface （ AWS CLI ）API、または 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SDKs。

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 TCP — TCPプロトコルを使用するにはこのオプションを選択しますが、開くポートがわかりません。これにより、すべてのポート (0 ~ 65535) TCPで が有効になります。
- すべて UDP — UDPプロトコルを使用するにはこのオプションを選択しますが、開くポートがわかりません。これにより、すべてのポート (0 ~ 65535) UDPで が有効になります。
- すべて 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タイプとコードを定義するには、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「コントロールメッセージ」を参照してください。
- DNS – インスタンスDNSで 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 UDP経由で TCPと が有効になります。
- HTTP — ウェブブラウザがインスタンスでホストされているウェブサイトに接続できるようにする場合は、このオプションを選択します。これにより、ポート 80 TCP以上が有効になります。

- HTTPS – ウェブブラウザを有効にして、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を確立する場合は、このオプションを選択します。これにより、ポート 443 TCP経由で が有効になります。
- My SQL/Aurora – クライアントがインスタンスでホストされている MySQL または Aurora データベースに接続できるようにするには、このオプションを選択します。これにより、ポート 3306 TCP経由で が有効になります。
- Oracle RDS- クライアントがインスタンスでホストされている Oracle またはRDSデータベースに接続できるようにするには、このオプションを選択します。これにより、ポート 1521 TCP経由でが有効になります。
- Ping (ICMP) – インスタンスが Ping ユーティリティを使用してリクエストに応答できるようにするには、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) を有効にします。IPv6 ファイアウォールでは、ICMPタイプ 129 (エコーリプライ) とコード 0 が有効になります。
- RDP – RDPクライアントがインスタンスに接続できるようにするには、このオプションを選択します。これにより、ポート 3389 TCP経由で が有効になります。
- SSH – SSHクライアントがインスタンスに接続できるようにするには、このオプションを選択します。これにより、ポート 22 TCP経由で 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および 経由のウェブブラウザなどのトラフィックに最適ですHTTPS。ただし、すべての IP アドレスがそ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できるようにしたくないためRDP、 SSHや などのトラフィックにはセキュリティ上のリスクがあります。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、または IP アドレスの範囲に制限することを選択できます。

- IPv4 ファイアウォールの場合 - 1 つのIPv4アドレス (203.0.113.1 など) またはIPv4アドレスの範囲を指定できます。Lightsail コンソールでは、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) を使用して範囲を指定できます。CIDR ブロック表記の詳細については、Wikipedia の「Classless Inter-Domain Routing」を参照してください。
- IPv6 ファイアウォールの場合 - 1 つのIPv6アドレス (2001:0db8:85a3:0000:0000:8a2e:0370:7334 など) またはIPv6アドレスの範囲を指定できます。Lightsail コンソールでは、IPv6範囲はCIDRブロック表記 (2001:db8::/32 など) を使用して指定できます。IPv6 CIDR ブロック表記の詳細については、Wikipedia の「 IPv6CIDR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 IPv4および IPv6ファイアウォールには、インスタンスへの基本的なアクセスを許可する次の一連のデフォルトルールが事前設定されます。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、Free BSD、open SUSE、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 (eCommerce 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、SQLServer Express 2019、および 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

Lightsail インスタンスにファイアウォールルールを追加する

Amazon Lightsail インスタンスの IPv4および IPv6ファイアウォールにルールを追加して、そのインスタンスへの接続が許可されてい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ときは、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、およびインスタンスへの接続が許可されている送信元IPv4または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。

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、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。

   ネットワークタブには、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、およびインスタンス用に設定された IPv4または IPv6ファイアウォ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IPv6に対して を有効にした場合にのみ表示されます。詳細については、「 を有効または無効にするIPv6」を参照してください。

5. ルールの送信元 IP が IPv4または IPv6 アドレスかどうかに応じて、次のいずれかの手順を実行します。

   - IPv4 ファイアウォールルールを追加するには、ページのIPv4ファイアウォールセクションまで下にスクロールし、ルールの追加 を選択します。
   - IPv6 ファイアウォールルールを追加するには、ページのIPv6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、カスタム 、すべての TCP、すべての UDP、カスタム ICMP、SSH、および ですRDP。

   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と ですUDP。

     [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) カスタム ICMP オプションを選択した場合は、タイプ フィールドに ICMPタイプを指定し、ICMPコード フィールドに コードを指定できます。ICMP タイプとコードの詳細については、Wikipedia 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、ルールは を使用するように自動的に設定されますICMPv6。詳細については、Wikipedia の「Internet Control Message Protocol for IPv6 」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 (など`203.0.113.1`) またはIPv4アドレスの範囲を入力できます。範囲は、ダッシュ (など`192.0.2.0-192.0.2.255`) またはCIDRブロック表記 (など) を使用して指定できます`192.0.2.0/24`。CIDR ブロック表記の詳細については、Wikipedia の「Classless Inter-Domain Routing」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のプロトコルタイプを選択し、IP アドレスに制限 を選択した場合、Lightsail コンソールで利用可能なブラウザベースとクライアントを使用して、Lightsail ブラウザSSH/RDP を許可 を選択してインスタンスへの接続を許可

できます。 SSH RDP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。

ファイアウォールルールを削除する

ファイアウォールルールの追加と編集に加えて、Amazon Lightsail インスタンスの既存のルールを削除することもできます。特定のインバウンドトラフィックをインスタンスに許可する必要がなくなった場合は、ファイアウォールルールの削除が必要になる場合があります。IPv4 およびIPv6ファイアウォールルールを削除するプロセスは簡単で、Lightsail コンソールから直接実行できます。Lightsail コンソールでインスタンスファイアウォールルールを削除するには、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 ホームページで、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または IPv6 アドレスかどうかに応じて、次のいずれかの手順を実行します。

   - IPv4 ファイアウォールルールを削除するには、ページのIPv4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 (ごみ箱アイコン) を選択して削除します。
   - IPv6 ファイアウォールルールを削除するには、ページのIPv6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 (ごみ箱アイコン) 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。インスタンスのプライベート IP アドレスを介して流れるトラフィックには影響しません。このトラフィックは、同じ のアカウント内の Lightsail リソース AWS リージョン、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があります AWS リージョン。例えば、インスタンスファイアウォールからSSHルール (TCPポート 22) を削除すると、同じ Lightsail アカウントおよび同じ 内の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SSHして AWS リージョンを使用してルールに接続し続けることができます。

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。

Lightsail インスタンスのファイアウォールルールリファレンス

インスタンスのロールを反映するルールを Amazon Lightsail インスタンスのファイアウォールに追加できます。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、デフォルトで次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、次のいずれかのファイアウォールルールがデフォルトで設定されています。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。

最適なパフォーマンスを得るための Lightsail インスタンスバーストの検出

Amazon Lightsail インスタンスはベースラインCPUのパフォーマンスを提供しますが、必要に応じて一時的にベースラインを超えるCPUパフォーマンスを提供する機能もあります。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 — インスタンスで使用されている割り当てられたコンピューティングユニットの割合。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 バーストキャパシティの割合 — インスタンスで使用可能なCPUパフォーマンスの割合。
- CPU バーストキャパシティ分 – インスタンスが 100% のCPU使用率でバーストするのに使用可能な時間。

以下のトピックでは、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について説明します。

トピック

- Lightsail インスタンスのベースラインCPUパフォーマンスとバーストキャパシティの蓄積を理解する
- Lightsail インスタンスのCPUバーストキャパシティの蓄積を表示する
- Lightsail インスタンスがバーストするタイミングを特定する
- Lightsail インスタンスの CPU バーストキャパシティをモニタリングする

- Lightsail インスタンスのCPU使用率とバーストキャパシティを表示する
- Lightsail インスタンスの CPU 使用率が高い場合のトラブルシューティング

Lightsail インスタンスのベースラインCPUパフォーマンスとバーストキャパシティの蓄積を理解する

Lightsail インスタンスは 1 時間あたりのバーストCPUキャパシティーの設定レートを継続的に獲得します (ミリ秒レベルの解像度) 。これは、インスタンスのCPU使用率が 0% を超えるときにも消費されます。バーストキャパシティーが蓄積されるか消費されるかのアカウンティングプロセスはミリ秒レベルの解像度でも発生するため、CPUバーストキャパシティーの過剰消費を心配する必要はありません。 の短いバーストでは、バーストキャパシティーのごく一部CPUが使用されます。

インスタンスがベースラインパフォーマンスに必要なリソースよりも少ないCPUリソースを使用する場合 (アイドル状態の場合など) 、未使用のCPUバーストキャパシティはCPUバーストキャパシティの割合と分単位で蓄積されます。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は、蓄積されたCPUバースト容量が消費されます。インスタンスが蓄積したCPUバーストキャパシティーが多いほど、パフォーマンスの向上が必要な場合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

次の表は、Lightsail のデュアルスタック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。IPv6のみのプランの料金は異なりますが、パフォーマンスベースラインは同じです。

インスタンスプラン	vCPUs	「メモリ」	[Storage (ストレージ)]	パフォーマンスのベースライン
Linux または Unix 5 USD および Windows 9.50 USD	2	512 MB	20 GB	5%
Linux または Unix 7 USD および Windows 14 USD	2	1 GB	40 GB	10%
Linux または Unix 12 USD および Windows 22 USD	2	2 GB	60 GB	20%

インスタンスプラン	vCPUs	「メモリ」	[Storage (ストレージ)]	パフォーマンスのベースライン
Linux または Unix 24 USD および Windows 44 USD	2	4 GB	80 GB	20%
Linux または Unix 44 USD および Windows 74 USD	2	8 GB	160 GB	30%
Linux または Unix 84 USD および Windows 124 USD	4	16 GB	320 GB	40%
Linux または Unix 164 USD および Windows 244 USD	8	32 GB	640 GB	40%
* Linux または Unix 384 USD および Windows 574 USD	16	64 GB	1,280 GB	40%

* Linux または Unix の 384 USD および Windows の 574 USD のインスタンスプランでは、CPU バーストキャパシティは発生しません。必要に応じて自動的にバーストします。

これらのパフォーマンスベースラインは v ごとですCPU。Lightsail コンソールのCPU使用率メトリクスグラフは、複数の v 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CPU。例えば、Linux または Unix ベースの 44 USDUSD/月インスタンスには 2 つ vCPUs あり、平均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。

- 一方の vCPU は 50% で動作し、もう一方の v は 0% で動作し、グラフには 25% の平均CPU使用率が表示されます。これにより、インスタンスのCPU使用率がベースラインの 30% を下回っ、サステナブルゾーンに配置されます。
- 一方の vCPU は 30% で動作し、もう一方の v は 20% で動作し、平均CPU使用率 25% がグラフに表示されます。これにより、インスタンスのCPU使用率がベースラインの 30% を下回っ、サステナブルゾーンに配置されます。
- 1 つの vCPU は 35% で動作し、もう 1 つは 25% で動作し、平均CPU使用率 30% 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% ベースラインになります。

- 一方の vCPU は 100% で動作し、もう一方の v は 90% で動作し、平均CPU使用率の 95% がグラフに表示されます。これにより、インスタンスのCPU使用率がベースラインの 30% を超え、バーストゾーンに配置されます。

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。

旧世代CPUのパフォーマンス

次の表は、2023 年 6 月 29 日より前に作成された Lightsail インスタンスのパフォーマンスベースラインの概要を示しています。これらのパフォーマンスベースラインは v ごとですCPU。

インスタンスプラン	vCPUs	「メモリ」	[Storage (ストレージ)]	パフォーマンスのベースライン
Linux または Unix 5 USD および Windows 9.50 USD	1	512 MB	20 GB	5%
Linux または Unix 7 USD および Windows 14 USD	1	1 GB	40 GB	10%
Linux または Unix 12 USD および Windows 22 USD	1	2 GB	60 GB	20%
Linux または Unix 24 USD および Windows 44 USD	2	4 GB	80 GB	20%
Linux または Unix 44 USD および Windows 74 USD	2	8 GB	160 GB	30%
Linux または Unix 84 USD および Windows 124 USD	4	16 GB	320 GB	22.5%
Linux または Unix 164 USD および Windows 244 USD	8	32 GB	640 GB	17%

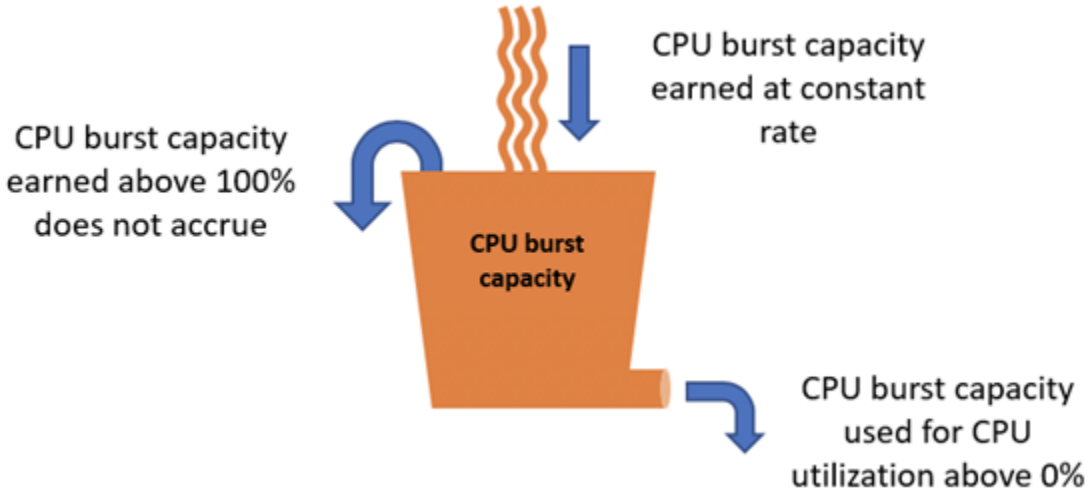
Lightsail インスタンスのCPUバーストキャパシティの蓄積を表示する

Amazon Lightsail インスタンスプランは、Linux または Unix の 384 USD および Windows の 574 USD プランを除き、1 時間あたりのCPUバーストキャパシティの 4.17% が発生します。蓄積できる最大CPUバーストキャパシティは、24 時間で獲得できるCPUバーストキャパシティの割合に相当します。CPU バーストキャパシティの割合が 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発生を停止します。

Important

蓄積されたCPUバーストキャパシティ

- Linux または Unix の 384 USD および Windows の 574 USD のインスタンスプラン – これらのプランにはCPUバーストキャパシティは発生しません。必要に応じて自動的にバーストします。
- 2023 年 6 月 29 日より前に作成されたインスタンス – インスタンスが停止しても、CPU バーストキャパシティは保持されません。インスタンスを停止すると、蓄積されたバースト容量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 – バーストキャパシティは、インスタンスの停止と起動の 7 日間保持されます。 CPU
- 実行中のインスタンスで蓄積されたCPUバーストキャパシティは期限切れになりません。

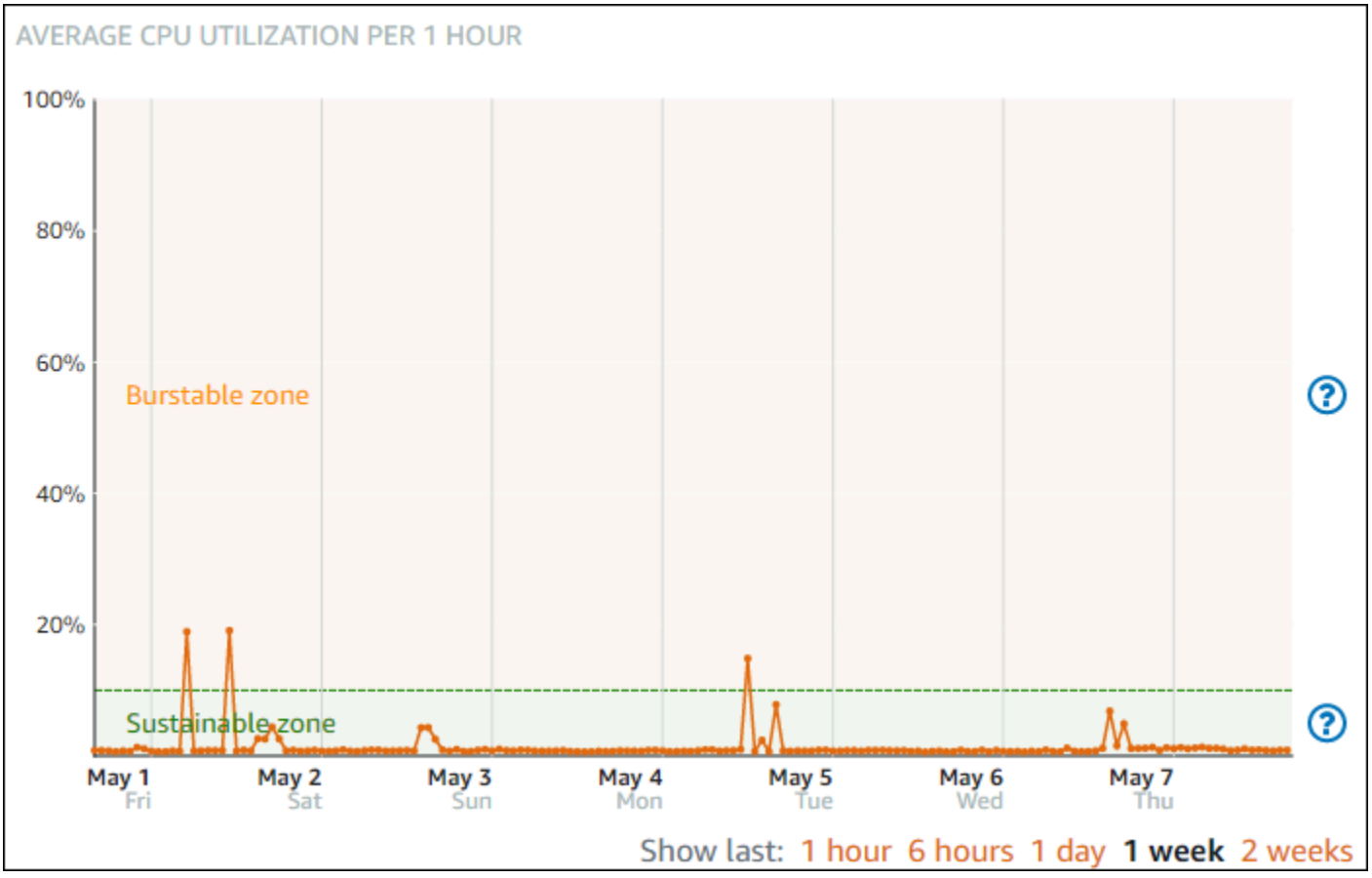


Lightsail インスタンスは起動時に追加のCPUバーストキャパシティを受け取ります。これは起動CPUバーストキャパシティと呼ばれます。起動CPUバーストキャパシティーを使用すると、インスタンスは起動直後にバーストし、その後に追加のバーストキャパシティーが蓄積されます。起動CPUバーストキャパシティは、バーストキャパシティの制限にはカウントされません。インスタンスが起動CPUバースト容量を消費しておらず、24 時間アイドル状態でバースト容量が増えている場合、CPUそのバースト容量 (パーセンテージ) メトリクスグラフは 100% 以上と表示されます。

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。

Lightsail インスタンスがバーストするタイミングを特定する

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 7 USD/月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。

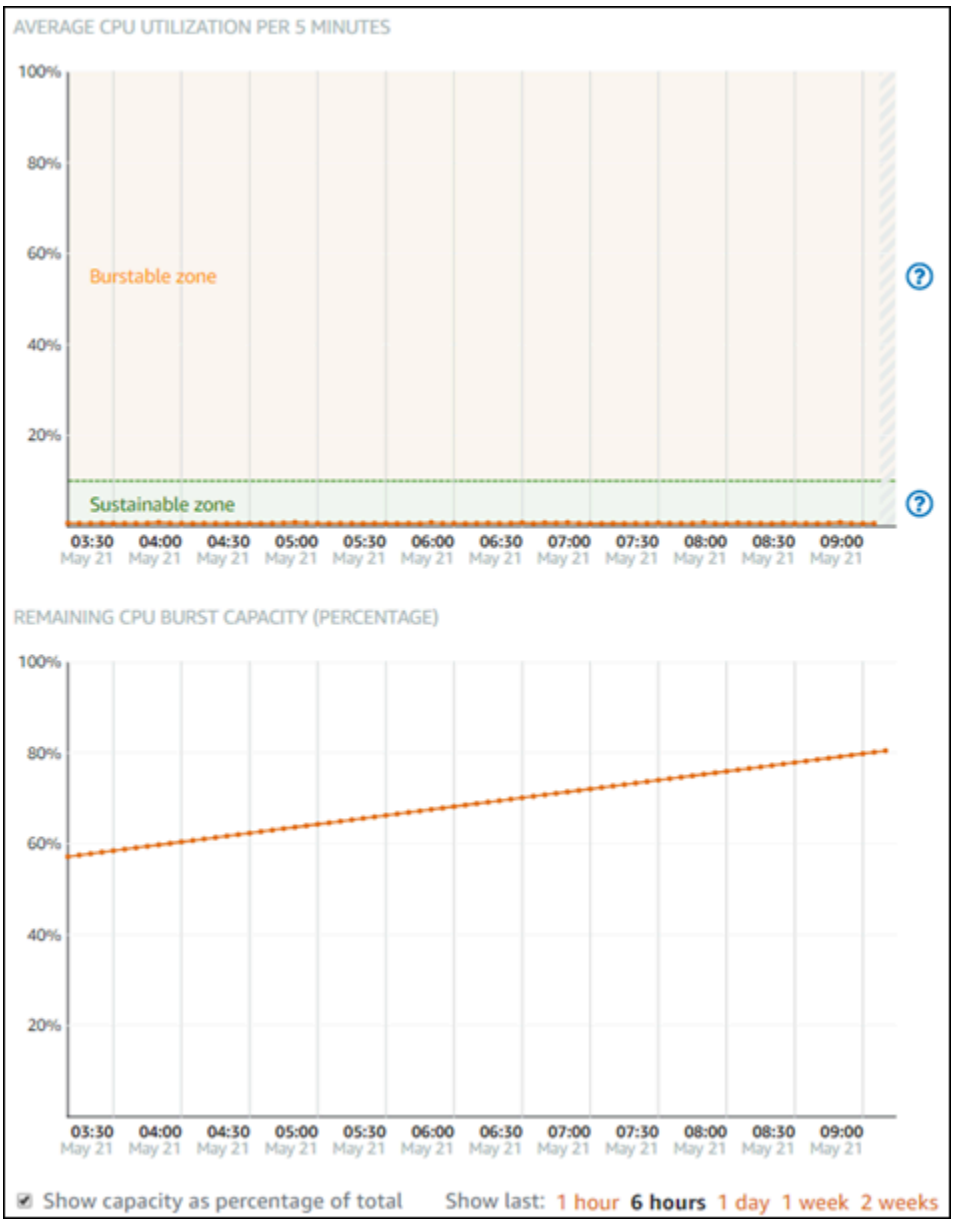


Lightsail インスタンスは、システムの動作に影響を与えずに、持続可能ゾーンで無期限に運用できます。コードのコンパイル、新しいソフトウェアのインストール、バッチジョブの実行、ピークの負荷リクエストの処理など、負荷が高い場合、インスタンスがバースト可能なゾーンで動作し始めることがあります。バーストな可能ゾーンで動作している間、インスタンスは大量の CPU サイクルを消費します。したがって、この領域では限られた期間しか作動できません。

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。

Lightsail インスタンスの CPU バーストキャパシティをモニタリングする

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 ベースの 7 USD/月インスタンスの CPU 使用率ベースラインは 10% で、1 時間あたり 6 分間の CPU バーストキャパシティが発生します。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。

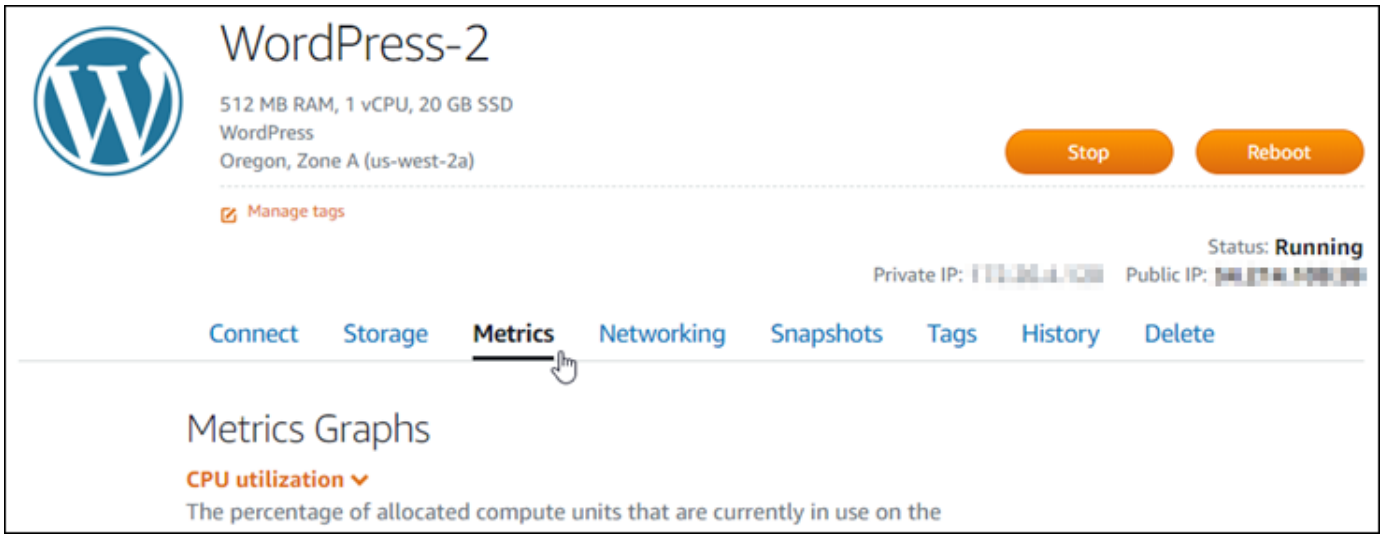
- 60 分間、バースト可能なゾーンでの CPU 使用率が 50% のとき、その期間中、CPU バースト容量 (分数) が 50% のレートで消費されます。インスタンスは 30 分の CPU バーストキャパシティを消費し、6 分を累積するため、正味 24 分が消費されます。
- 60 分間、インスタンスのベースラインでの CPU 使用率が 10% のとき、その期間中、CPU バースト容量 (分数) が 10% のレートで消費されます。インスタンスは 6 分の CPU バースト容量を消費し、6 分を蓄積します。インスタンスがベースラインで動作しているとき、CPU バースト容量の分数は増減しません。
- 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。

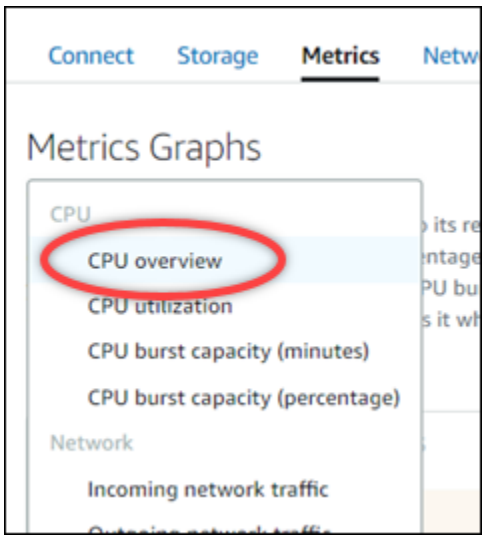
Lightsail インスタンスのCPU使用率とバーストキャパシティを表示する

概要CPUページにアクセスし、インスタンスのCPU使用率と残りのCPUバーストキャパシティを表示するには、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 ホームページで、CPU使用率とバーストキャパシティを表示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で [Metrics (メトリクス)] タブを選択します。



4. メトリクスグラフの見出しのドロップダウンメニューでCPU概要を選択します。



このページには、5 分あたりの平均CPU使用率と残りのCPUバーストキャパシティグラフが表示されます。

> **Note**
>
> インスタンスの作成後、残りのCPUバーストキャパシティグラフに短時間起動モードゾーンが表示されることがあります。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。

5. 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 使用率とバーストキャパシティが指定したしきい値を超えたときに通知されるアラームを追加します。CPU 概要ページにアラームを追加することはできません。個々のCPU使用率、CPUバーストキャパシティの割合、CPUバーストキャパシティ分メトリクスグラフページに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 インスタンスの 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し、インスタンスのサイズを増やす必要がなくなります。

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。例えば、前のインスタンスで使用した Linux または Unix ベースの 1 か月あたり 12 USD プランではなく、新しいインスタンスで Linux または Unix ベースの 1 か月あたり 24 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。

Lightsail インスタンスに接続して管理する

このガイドでは、Amazon Lightsail インスタンスの管理と接続に関連する以下のトピックについて説明します。

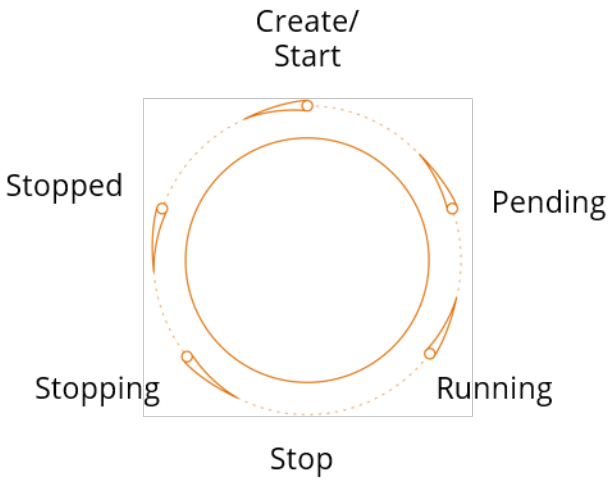
トピック

- Lightsail インスタンスを起動、停止、または再起動する

- Lightsail インスタンスの強制停止
- Amazon EC2 インスタンスの拡張ネットワーキングを有効にする
- Lightsail で Windows Server インスタンスのファイルシステムを拡張する
- Lightsail で起動スクリプトを使用して Linux/Unix インスタンスを設定する
- PowerShell および バッチスクリプトを使用して Windows Lightsail インスタンスを設定する
- Lightsail で Windows Server インスタンスを保護する

Lightsail インスタンスを起動、停止、または再起動する

Amazon Lightsail がインスタンスを作成すると、マシンは の実行を開始する前に保留状態になります。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。

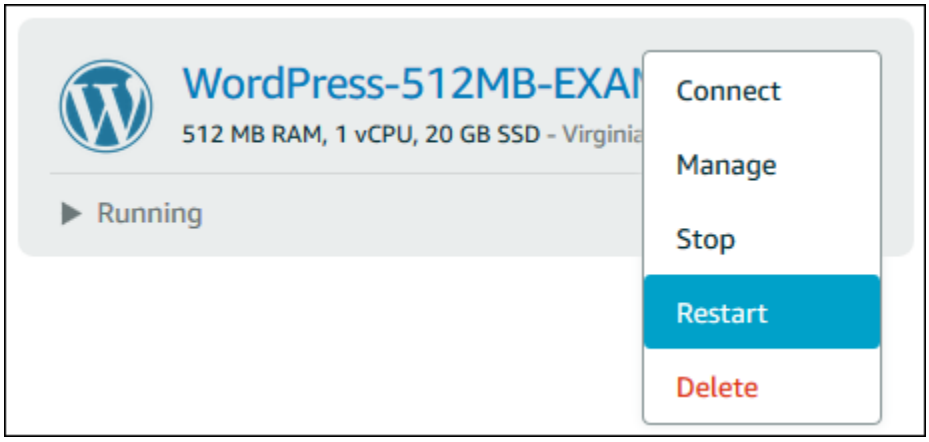


インスタンスの状態は、インスタンスを管理する場合やホームページでインスタンスを表示する場合に確認できます。

> **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パブリックIPv4アドレスは、インスタンスを停止して起動すると変更されます。オプションで、静的IPv4アドレスを作成してインスタンスにアタッチできます。静的IPv4アドレスはインスタンスのデフォルトのパブリックIPv4アドレスを置き換え、インスタンスを停止して起動しても同じままになります。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。

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。

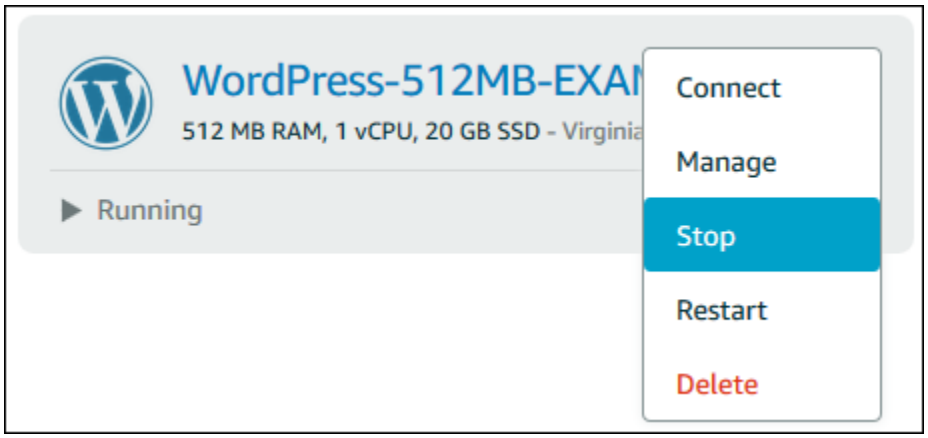


インスタンス管理ページでインスタンスを表示している場合は、[再起動] を選択し、プロンプトが表示されたら [確認] を選択します。

> **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。

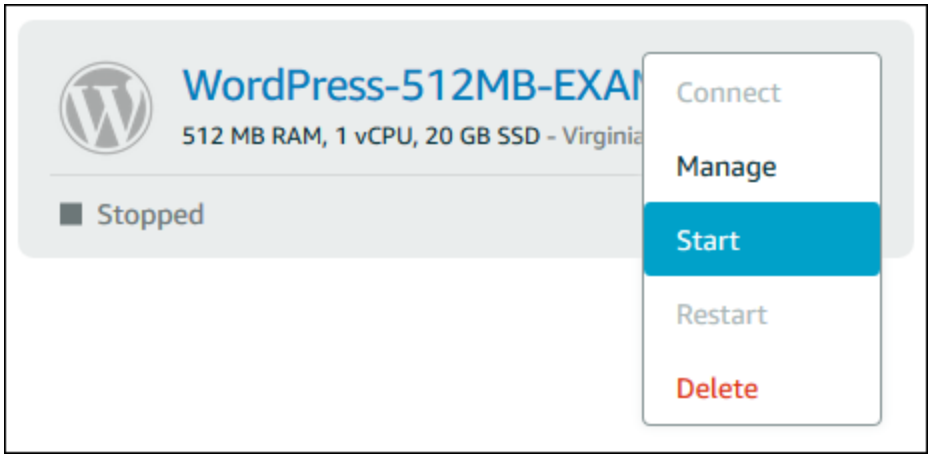


インスタンス管理ページでインスタンスを表示している場合は、[停止] を選択し、プロンプトが表示されたら [確認] を選択します。

> **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。

Lightsail インスタンスの強制停止

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。この場合、Amazon 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、「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

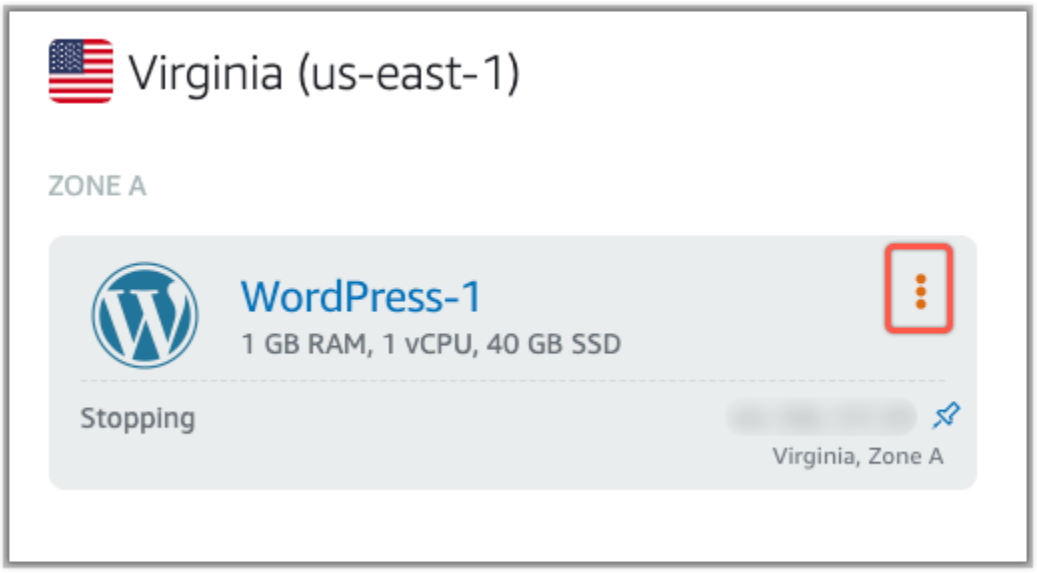
Lightsail コンソールを使用してインスタンスを強制停止できますが、インスタンスが `stopping`状態の間のみ可能です。または、インスタンスが `shutting-down` と `terminated` の状態にあるとき以外であれば、 AWS Command Line Interface (AWS CLI) を使用してインスタンスを強制停止することもできます。強制停止が完了するまでに数分かかる場合があります。10 分経ってもインスタンスが停止しない場合は、再度強制停止します。

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。

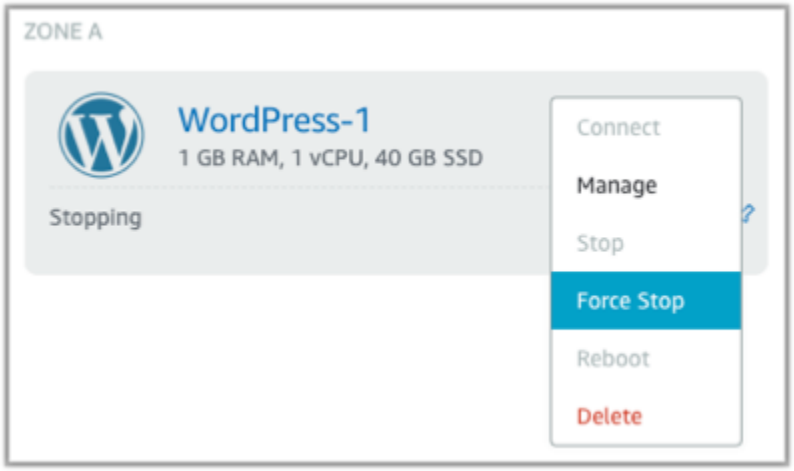
次の手順では、Lightsail インスタンスを強制停止するさまざまな方法について説明します。

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。

を使用してインスタンスを強制停止する AWS CLI

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
   ```

Amazon EC2 インスタンスの拡張ネットワーキングを有効にする

Lightsail インスタンスの中には、拡張ネットワーキングが有効になっていないため、現行世代の EC2 M5, C5、または R5) と互換性がないものがあります。ソース 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、M4, C4、または R4) を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 インスタンスの作成」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。 Amazon EC2

> **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 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、前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クドライバーを更新してから、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アップグレードする必要があります。

前提条件

エクスポートされた 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旧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 でのエクスポートされたスナップショットからの Amazon EC2 インスタンスの作成」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

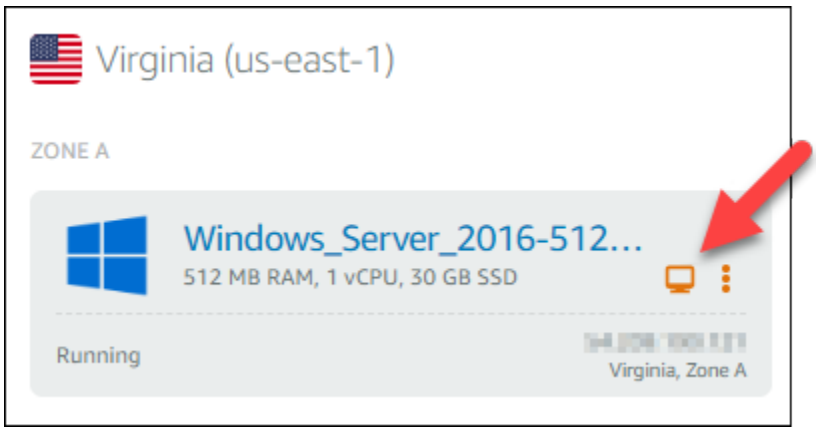
Lightsail で Windows Server インスタンスのファイルシステムを拡張する

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。

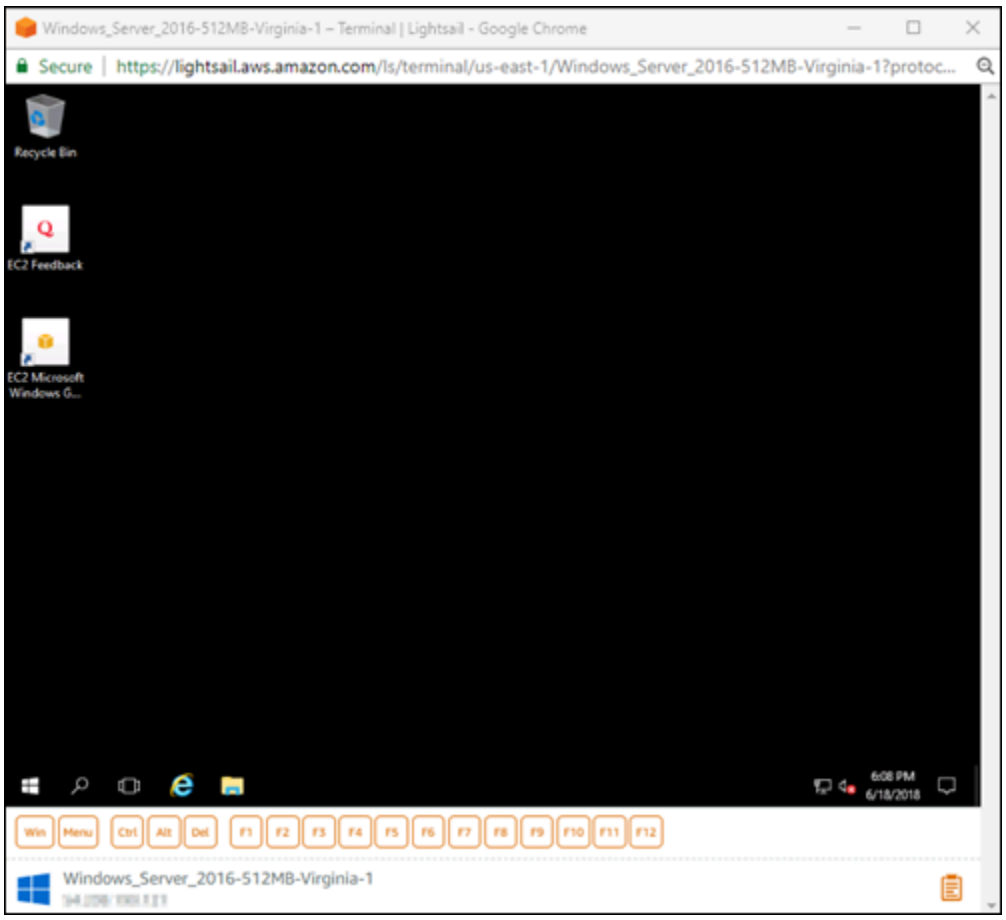
> **Note**
>
> この状況が発生するのは、システム準備 (Sysprep) ユーティリティの実行前に作成したスナップショットを使用して Windows Server インスタンスを作成した場合に限ります。詳細については、「Windows Server インスタンスのスナップショットを作成する」を参照してください。

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 ホームページで、接続するインスタンスの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。



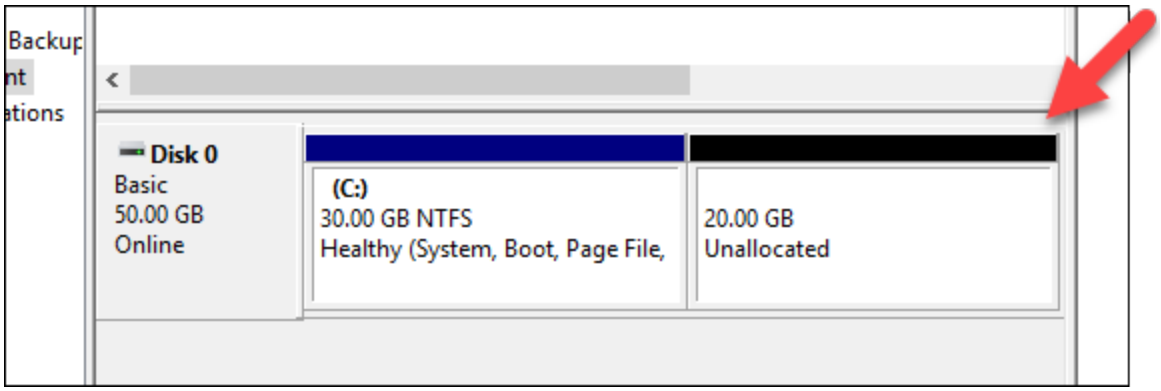
3. タスクバーで Windows アイコンを選択し、以下のいずれかのオプションを選択します。

   - Windows Server 2022、Windows Server 2019、および Windows Server 2016 インスタンスで、「 の開始」を選択し、「Windows 管理ツール」を選択します。

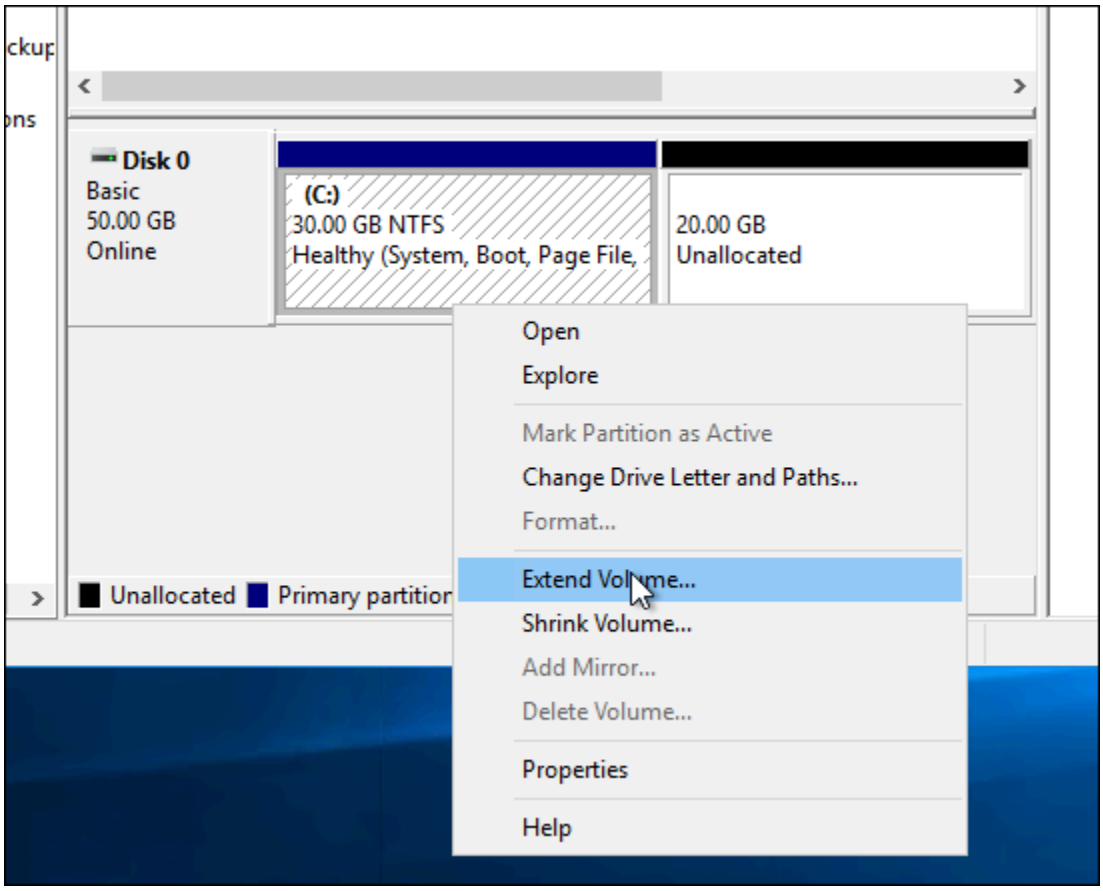
4. [コンピューターの管理] を選択します。
5. [コンピューターの管理] コンソールの左側のペインで、[ディスクの管理] を選択します。

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。

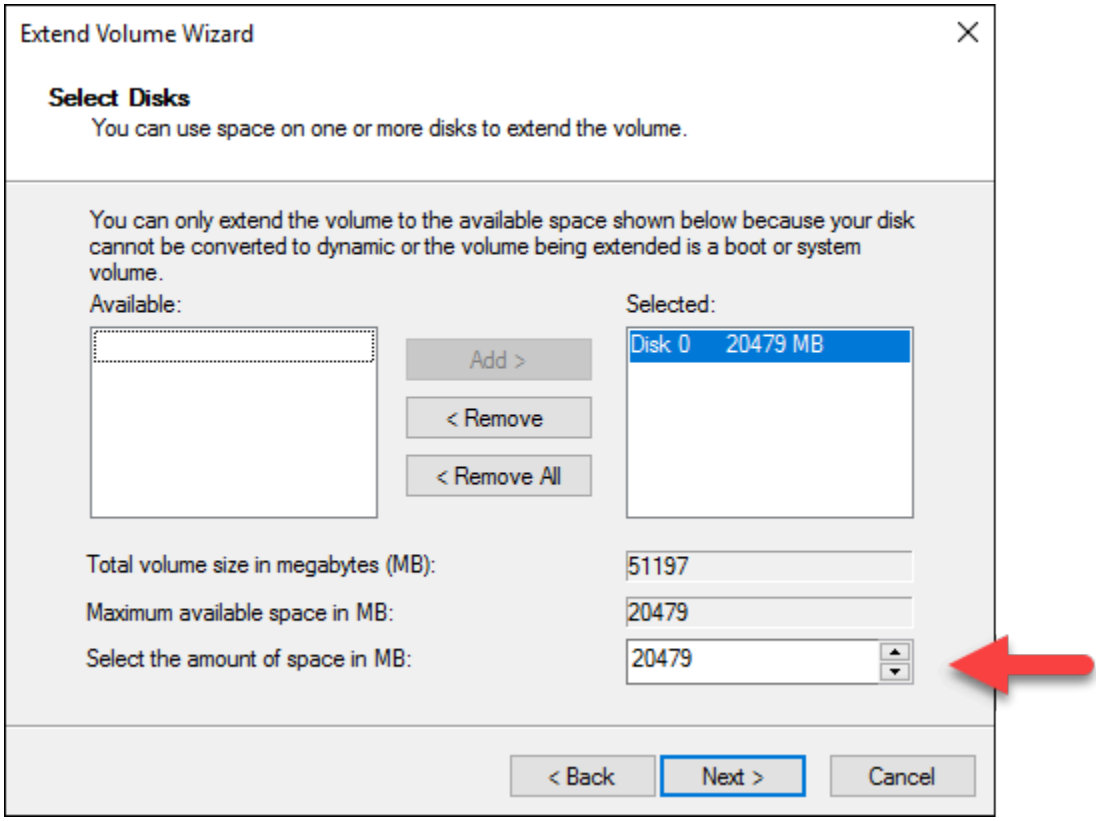


7. 未割り当て領域と同じディスクのアクティブボリュームを右クリックし、[ボリュームの拡張] を選択します。



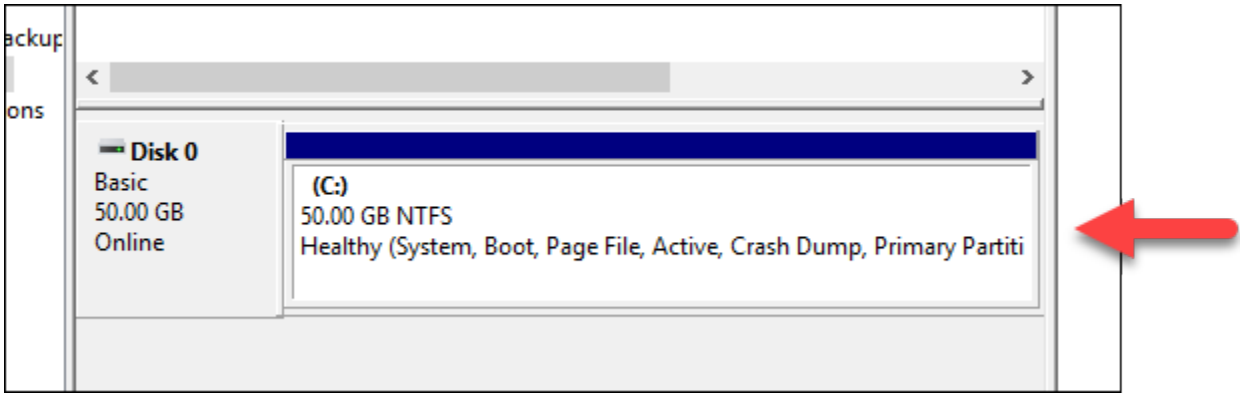
8. ボリュームの拡張ウィザードが開いたら、[次へ] を選択します。

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 で起動スクリプトを使用して Linux/Unix インスタンスを設定する

Linux または Unix ベースのインスタンスを作成するときは、起動スクリプトを追加してソフトウェアを追加または更新したり、別の方法でインスタンスを設定したりできます。追加のデータを使用して Windows ベースのインスタンスを設定するには、「Windows を使用して新しい Lightsail インスタンスを設定する PowerShell」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 は を使用しますがyum、Debian と Ubuntu はどちらも を使用しますapt-get。 WordPress 他のアプリケーションイメージは Debian をオペレーティングシステムとして実行apt-getするため、 を使用します。無料BSDおよびオープンSUSEの場合、 freebsd-updateや zypper (オープン) などのカスタムツールを使用するには、追加のユーザー設定が必要ですSUSE。

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

> **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サーバーを設定する WordPress

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

1. 「インスタンスの作成」ページで、「」を選択しますWordPress。
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 update
# download wordpress plugin
wget "https://downloads.wordpress.org/plugin/buddypress.14.0.0.zip"
apt-get install unzip
# unzip into wordpress plugin directory
unzip buddypress.14.0.0.zip -d /bitnami/wordpress/wp-content/plugins
```

4. [インスタンスの作成] を選択します。

PowerShell および バッチスクリプトを使用して Windows 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は、Windows PowerShell スクリプトまたはその他のバッチスクリプトを使用して設定できます。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 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 と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、Windows Server 2022、Windows Server 2019、Windows Server 2016 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>
```

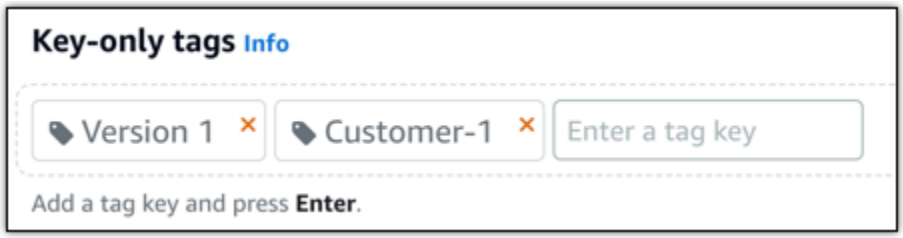
> **Note**
>
> PowerShell スクリプトは常に<powershell></powershell>タグでラップする必要があります。タグを使用するか、<script></script>タグなしで、非PowerShell コマンドまたはバッチスクリプトを入力できます。

7. インスタンスの名前を入力します。

   リソース名:

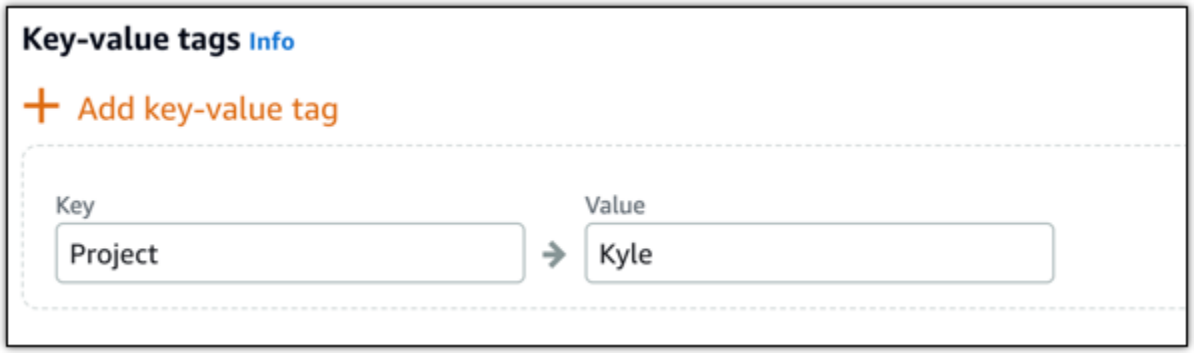
   - Lightsail アカウントの各 AWS リージョン 内で一意である必要があります。
   - 2〜255 文字を使用する必要があります。
   - 先頭と末尾は英数字または数字を使用する必要があります。
   - 英数字、数字、ピリオド、ダッシュ、アンダースコアを使用することができます。
8. 以下のいずれかのオプションを選択して、インスタンスにタグを追加します。

   - [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。

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接続を受け入れる準備が整うまでに、最大 15 分かかる場合があります。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したら、コマンドプロンプトを開きます (または Windows Explorer 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
```

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 で Windows Server インスタンスを保護する

この記事では、Windows Server を実行している Lightsail インスタンスを使用する際のセキュリティリスクを回避するのに役立つヒントとコツを提供します。

Lightsail 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、Lightsail は推測が難し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 () を使用してインスタンスにすばやく接続できますRDP。Lightsail インスタンスでは、常に管理者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便利です。Lightsail は、生成したパスワードを保存しません。

> **Note**
>
> Lightsail のブラウザベースのRDPクライアントでは、Lightsail が生成し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したい場合は、ブラウザベースのRDPクライアントで Lightsail 生成のデフォルトパスワードを使用する方が簡単です。

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 はパスワードを保存しないため、パスワードは必ず記録しておきます。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。

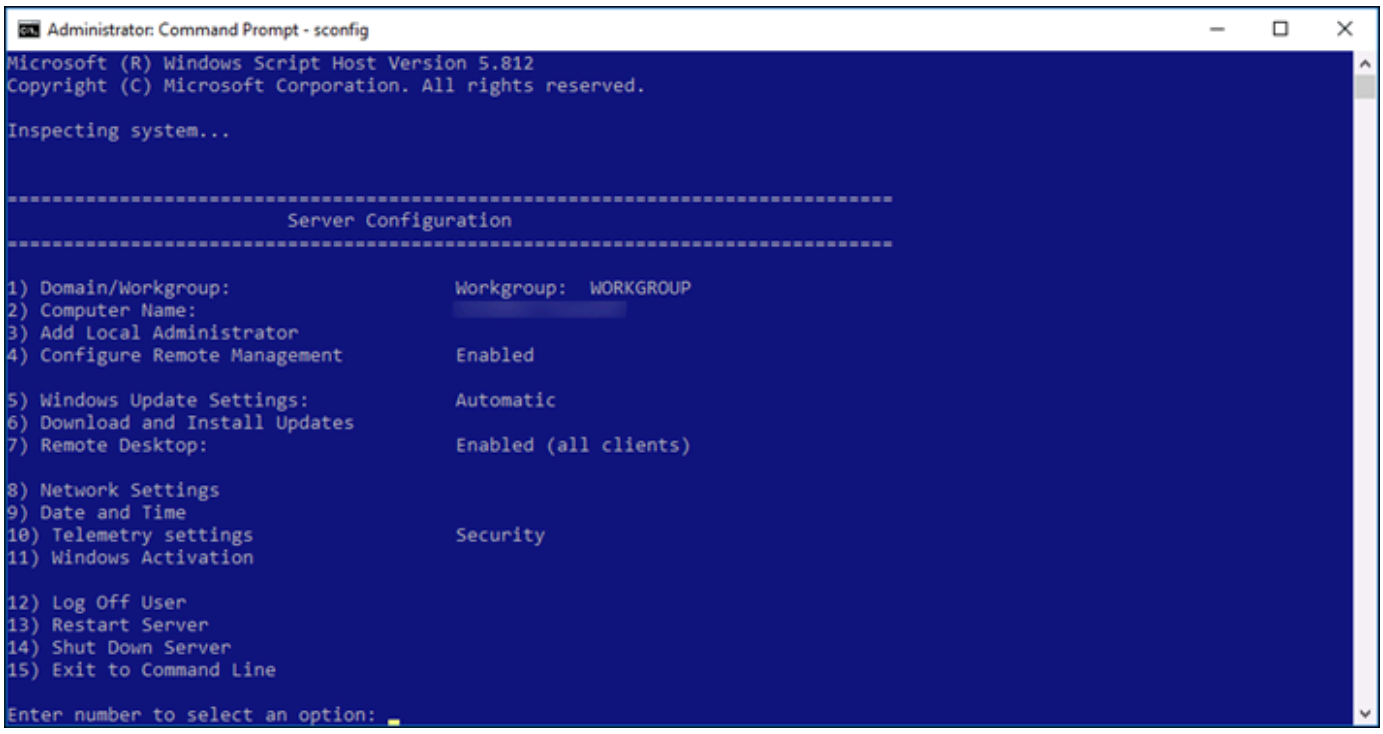
パスワードを一意のデフォルトパスワードから変更する場合、必ず強力なパスワードを使用してください。名前や辞書に載っている単語をベースとしたパスワードや、一連の文字の繰り返しは避けてください。

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しておく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 を実行している Lightsail インスタンスでこれを直接実行する方法について説明します。

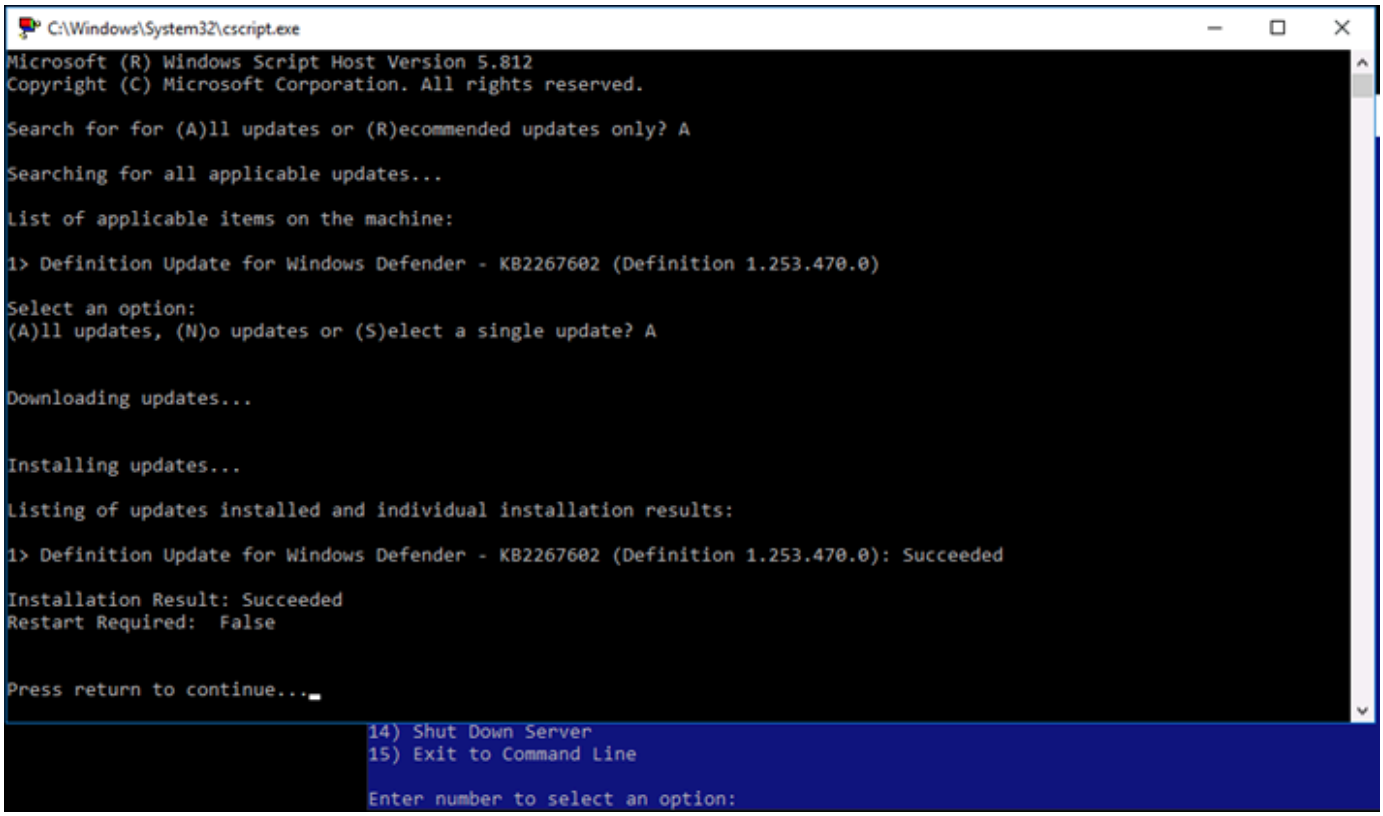
1. Windows Server ベースのインスタンスで、コマンドプロンプトを開きます。
2. 「`sconfig`」と入力し、`Enter` を押します。

   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)。



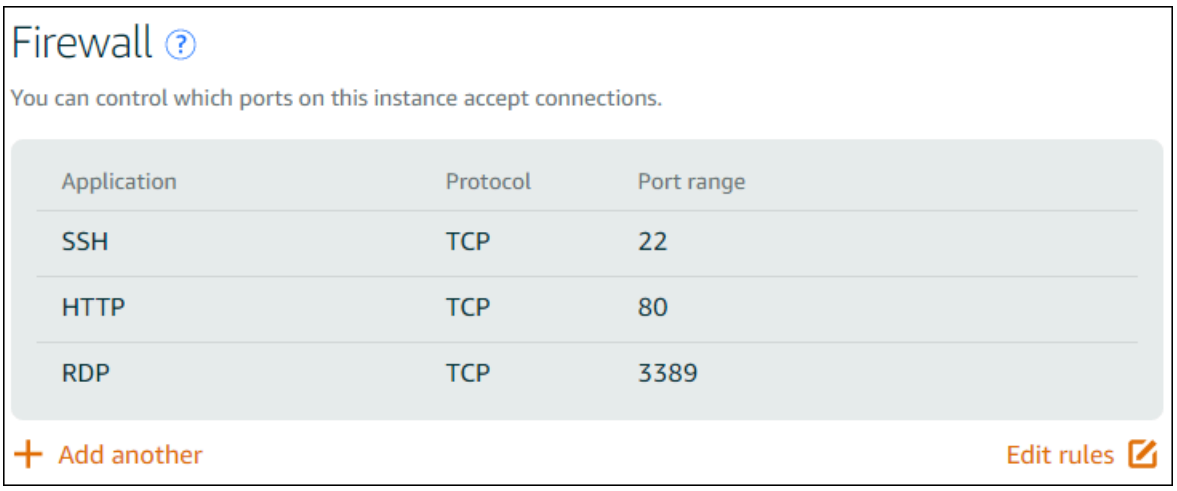
Windows Server でアカウントロックアウトポリシーを有効にする

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。

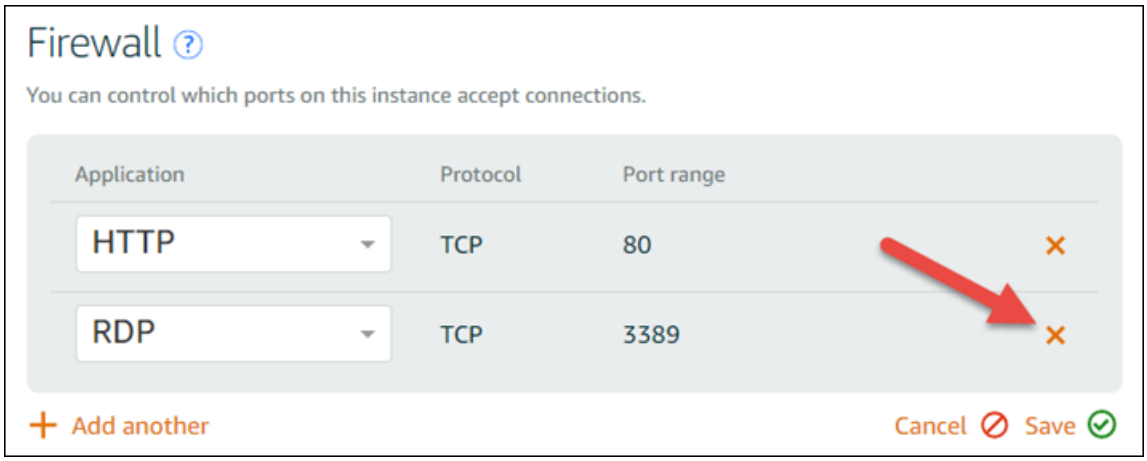
ポートとファイアウォールの設定

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、 の管理を選択します。
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」を選択して、RDP/TCP/3389 ルールを削除します。



5. [Save] を選択します。

step-by-step 手順に従って、インスタンスの状態の制御、停止したインスタンスの強制停止、拡張ネットワーキング用のインスタンスの更新、Windows Server インスタンスのファイルシステムの拡張、スクリプトを使用した起動時のインスタンスの設定、Windows Server インスタンスの保護を行う方法を学びます。

このガイドでは、Linux インスタンス、Unix インスタンス、Windows Server インスタンスの両方について説明し、ソフトウェアのインストール、設定の更新、パスワードの管理、セキュリティパッチの有効化、ファイアウォール設定の設定などのタスクに関するヒントとベストプラクティスを提供します。このガイドに従うことで、Lightsail インスタンスを効果的に管理および保護し、特定のユースケースに最適なパフォーマンス、セキュリティ、カスタマイズを確保できます。

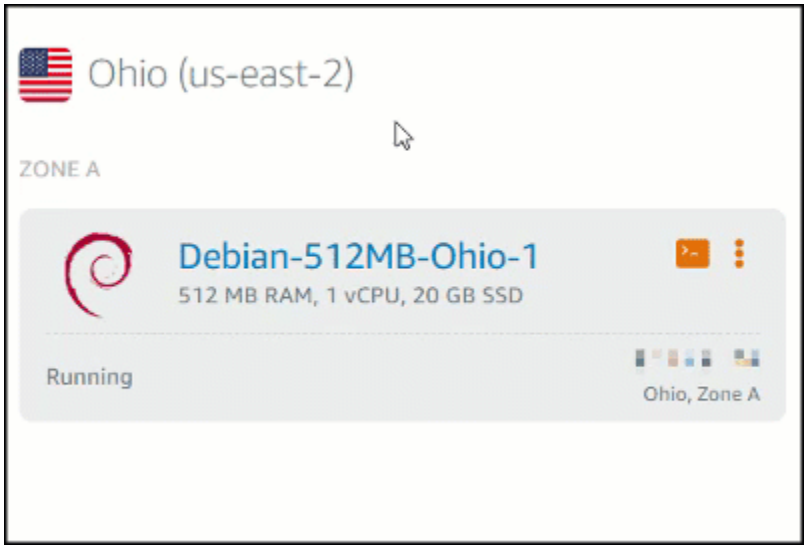
Lightsail インスタンスを削除する

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 () を使用して削除できますAWS CLI。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、静的インスタンスIPsやスナップショットなど、削除されたインスタンスにアタッチされたリソースは、削除するまで引き続き料金が発生します。

> **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からインスタンスを削除する

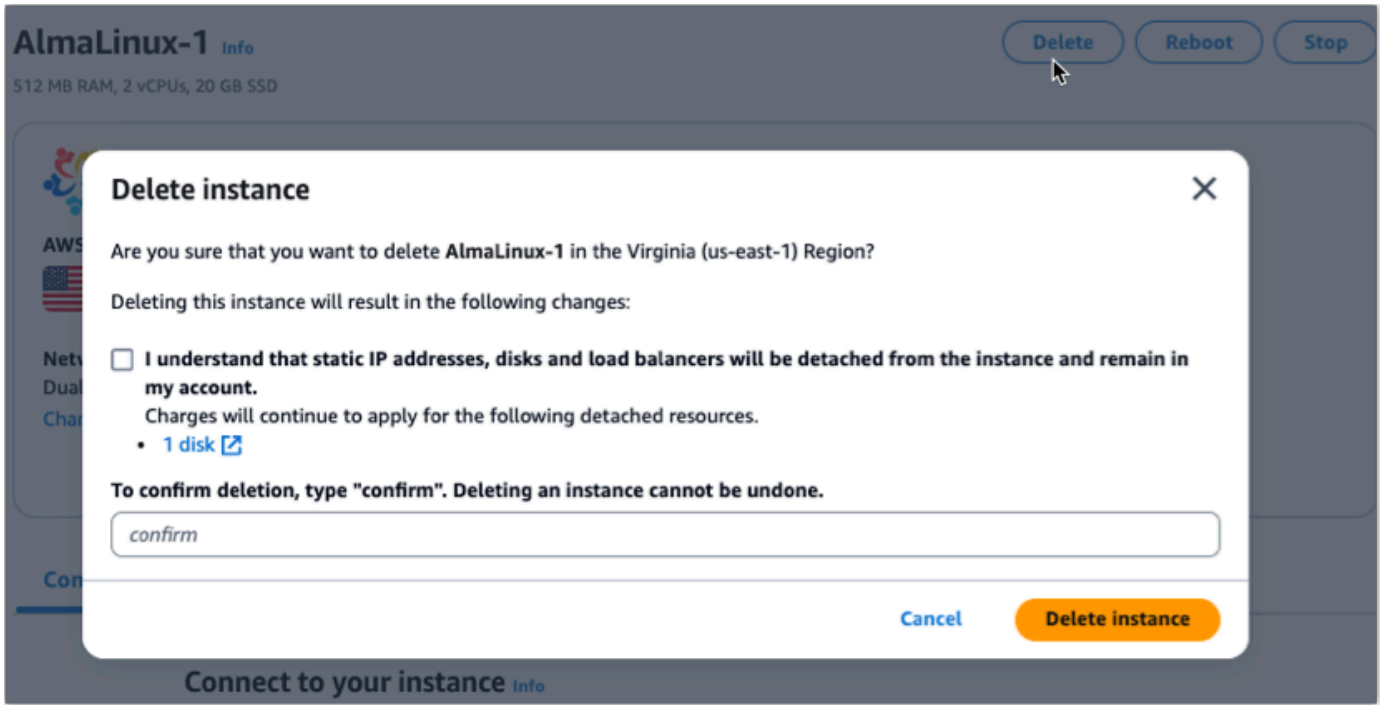
1. Lightsail コンソール にサインインします。
2. 削除するインスタンスのアクションメニューアイコン (⋮) を選択し、[削除] を選択します。



3. [はい、削除します] 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からインスタンスを削除する

1. ホームページの Lightsail コンソールで、削除するインスタンスを選択します。
2. 削除 ボタンを選択し、インスタンス の削除 を選択します。



3. チェックボックスを選択し、入力フィールドに確認と入力して、インスタンスを削除することを確認します。
4. インスタンスの削除 を選択して、削除を確定します。

を使用してインスタンスを削除する AWS CLI

1. まだ完了していない場合は、次の前提条件を満たします。
   a. をインストールします AWS CLI。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、「Configuring the AWS CLI」を参照してください。

c. (オプション) を使用します AWS CloudShell。詳細については、「???」を参照してください。

2. ターミナル、コマンドプロンプト、または CloudShell 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。

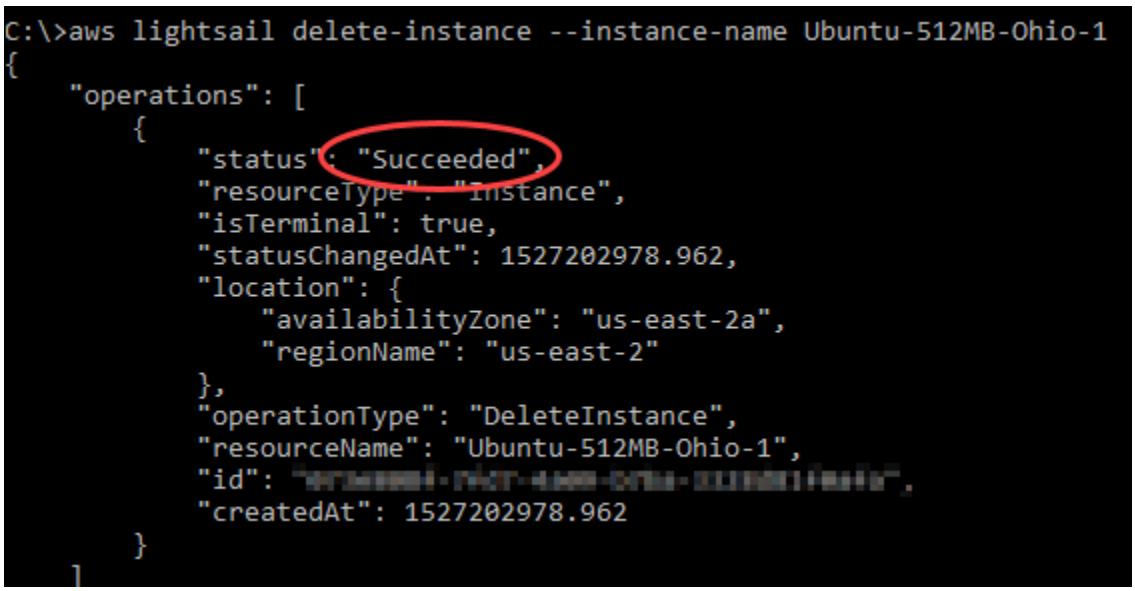
> **Note**
>
> 削除するインスタンスが表示されない場合は、インスタンス AWS リージョン が配置されている に対して AWS CLI が設定されていることを確認します。詳細については、「Configuring the AWS CLI」を参照してください。

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* インスタンスの名前。

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

インスタンスを削除すると、インスタンスに関連付けられた静的 IP、スナップショット、ブロックストレージディスク、ロードバランサーは Lightsail に残り、追加料金が発生します。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

SSH キーペアを管理し、Lightsail インスタンスに接続する

キーペアは、Amazon Lightsail インスタンスに接続するときに ID 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。

Example public key file text:

```
ssh-rsa
AAAAB3NEXAMPLEEAAAADAQABAAABgQDsF85afw9ctjz6maFf1c+1ZmTaFW0N5a+9nVVwKnsLeO
KG0Zm7XuTc6IGM5/ouFg45b2wO7L+5bNBl+jqYMYknAMFkidE0EXAMPLECs6AOnDOsq919TT85
8iGdo/7IxffHzlIOYZFQmTk1OvzUQ6EXAMPLEcUHzhUnjOiZSHLyYkja14vOO0TUdynFU1OWJw
zgGbprltuSUldHljvXTUFPEvLivBOTJn9JOyTLs0ixqb9tKtk/VWgxFhg4ggZQqTHyDfa3neKH
8TfUTod/t2HYpoh6dXvVVa+wec2s2p6Y2eKEXAMPLEkOK664F9pncYbGhrDGJUfuMBgXp/M0jm
8iTMC/ha/LMEXAMPLEiqVLj2RJaxE0StcoybaNwwhH9wFAJH5Uhr+iJAvlhPcrkCEw43jPzMQ
BILmDLiq0wH4/95joC9+W/u+g6p9quCiTSYmuqgXO/j/ZeNZ4OAe9DfjEXAMPLEQ/knkTdrXmo
Li2mkGm4qqgAVcN28/aoLcrKo
```

Example private key file text:

```
-----BEGIN OPENSSH PRIVATE KEY-----
b3BlbnNzaC1rZXktdjEAAAAACmFlczI1NiljdHIAAAAGYmNyeXB0AAAAGAAAABCrku9Liu
ySnuNMLn7e+gX4AEXAMPLEEAAAA6KAAAAB3NzaC1yc2EAAAADAQABAAABgQDsF85afw9cAS
tjz6maFf1c+1ZmTaFW0N5a+9EXAMPLEKnsLeOK90ZmTXuTc6IGM5/ouFg45b2wO7L+5bNB
qMYknAMFkidE0MBhpBUUCs4AOnDOsq919TT8Sl1Gdo/7IxffHzlIOYZFQmTk1OvzUQ6S9
Zmg8koUHznUnjOiZSHLyYkEXAMPLEvOUO7UdynFU1OWJwzgGbprltuUldHljvXTUFPEvLi
OTZn9JOyTLs0izxpb9tkk/VWgxFhg4ggZQqTHyDfa3nekN8TfUTod/t2HYpoh6dXvVVa+w
ec2s2p6Y2eKPborIkmkOK664F9pncYbGhrDGJUfuMBgXp/M0jm8iTMC/ha/LM3ODMHTLiq
VLj2RJaxE0StcoybaNwwhH0wFAJH5Uhr+iJAvlhPcrkCEw43jPzMQBILmDLiq0wH4/95j
oC9+W/u+g6p9quCiTSYmuqgXO/j/ZeNZ4OAe9DfjeRiGZBKQ/kmkTdrXmoLi2mkGm4qqgA
VcN28/aoLcrKoAAAWQ4oeC1hamC7lwPoFU3lEGVTydc6B6KZ+XdzchZC/tkIvmZNs1epjf
3UiLfuFTBhKJWxDunq1gLkP/ODpzpTAe8AjJbz6p/kf29TTBfvMO8v9HbNcN82YYmbjKk3q
i8kYQWg/3KNtQrhjY0SEXAMPLEzsef2stjA/ydXt53E8+ObvUdiaiJT9Q3nlkKlzERpDaV
sMh5bA9jA5y4f6OJkFsOS6xChtqyojZ6QQeViVetB3BflqvfxP3aVOrEllykfcF3kWQHgf
rFXgSAUhhnsg4wZbREAdedPK35vBsCIkfZKcMxblA5MZeYkOGLP2dYv9XcD2lTzvRqooGy
d5uYwlF9q5dkkhYv5YqW5k/XzwS89LImplqlE/Sobh6nblXiUnzGgtt9tCrAnTcKjLYzzk
zSwqxi4i0F7BFYn3fX2VMaTedKEXAMPLElT95uBuEnQ0xLsK/nn6AIQAMH0jZ5ZTnxZA3D
kImyn3WXifdVQpquzhyZN/Nc59Y8hu5BEoeghnlWi2tzbXY2NFIej/WIiV72tXfkOqE4
rVSkVvb4ZRGQ5aFNckvPeeqmEXAMPLEH4WhGDXKOmKaG79gImy7hg7nAiUMVWSUnJMFhLa
9bQJmQq7/zw/NnFE6g5mOgnrB6LzN/pcU4JZQH67RLPjVkviP/VTSkk09mYA34M0WAtr2N
Bx7BLinlqX7DAz6zwOAvIoHDjw5/AL4mQiN2kDewrV26Xtlim9QNL/st/hV5dyxfLqMgYs
8PB4yO0MjSM/yYEKqeKX5Mn7t+dKhHpoK5uFFfNue9YOn0Ekcl66O5yvCNMwESuehlFfnL
qQmFGpRNjfIfPIUYomaT9rPxB6q24nNneaB+PpWTaVvz9ogvVyW4/LAhlviYPgkrKxlhC4
NCHrI5lqGflupQZN7GGzOiM4/nYCdAuEXAMPLEACYkP6rMnPCMJMov7ObrAmWNLZ3hDS
QmiUWIDujql79ZUZxEo3I7WDbi+2LydILKTt5exuFZM85sAcpVTHTWv7OdOBBl/VNF/krx
iXOIxRFizKaI4OdqYvU3EWdAi4qqmcoaaqdztrotyAdER+Mr4jiff8GczDiSUhgqo5naTqA
rdPHG+9v5fQ659NkE9S5UmJNJ0boIQOB+i/eYA87GqnsdanM9WJ45dMOBdHtBo3H6C2vFRK
vrL89eCJeS+6MoMy2bJw9i3HH2QXb5bJiUALXE69V2NLtacgKQMe1ZA+TIWOlkM4DMQjM3
svli8KIFfc1zKZULMXci9lyAryAAZycCUGLhmW72tTV8e13ExttzJdvlM/T0z0zTzOXQry5
BIKbmjKoTGAGNm23eKEAxYEiTPtcEm07eLmfnr6IqGpD9FtspfwrIhBkRXhXTQ8fQH/Co
C+dgDEHUOnLCfezWIdoUZNNmbeIbALQghuOPLmUD9rOwybwjHT47N+NoQ7dbJOJoTW+Ic
Oh4vbPXU1SimQgfaCEJoqeTb/t3W7zn4cg3qSNCmyduRlkfDf9JkXzVwkr3etp4e0U8z2B
FOIf6WzU893L3bYnEEP0F3cxCqQ9ZyT3jn3oToneW5OBiTffFBIws4KPwQ4v0ayFq7keeb
J71N6HXiXrJzi56xz66vcUQzHcfi+N5Z8zetAh8csjknXLNAbjajM0BH6Zl5/LYZwOpaOz
2AZo0P1PpRsdpRJRtHQyOPtzFEkogs45fiGhjOLSQkQoYuILeb4QPa9k6avjM+LIcoORm4
xwy3ydPcvHs5aXuUS56s9TwvE+auhCsXiquOgjv/wVgA4OPWCTbE16JW7TtdSmDcVMEWRy
it3YrmLFYDyhQyWJXyH04XsBVj7o8PzEooJvUFlVnrgK2AWlipluWZfdV5TuvotN5LeAdm
v6Umk3avHUEh65Ie2qx75kB+hE=
-----END OPENSSH PRIVATE KEY-----
```

Linux および Unix インスタンスでは、プライベートキーを使用してインスタンスへの安全なSSH接続を確立できます。Windows インスタンスでは、プライベートキーは、インスタンスへの安全なRDP接続を確立するために使用するデフォルトの管理者パスワードを復号します。

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

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 にデフォルトキーペアを自動的に作成 AWS リージョン 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。各 に最大 1 つのデフォルトキーペアを設定できます AWS リージョン。
- 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、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。
- キーのアップロード (Linux および Unix インスタンス) – 独自の既存のキーペアを使用するには、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、一意の名前が付けられ、Lightsail によってインスタンスに保存されます。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

次のいずれかのオプションを使用して、Lightsail インスタンスに接続できます。

Lightsail ブラウザベース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 インスタンスに即座に接続し、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。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ときに、コンピュータに クライアントをインストールしたり、キーペアを設定したり、管理者パスワードを指定したり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SSHとRDPクライアント

サードパーティー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。SSH クライアントを使用する場合は、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場合は、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 – PuTTY を使用して、 を使用して Linux または Unix インスタンスに接続しますSSH。詳細については、「インスタンス に接続するための PuTTY のセットアップ」を参照してください。
- リモートデスクトップ接続 — リモートデスクトップ接続クライアントを使用して、 を使用して Windows インスタンスに接続しますRDP。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。

- ターミナルのネイティブSSHクライアント – ターミナルのネイティブ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ます。詳細については、「ターミナル で を使用して Linux または Unix インスタンスに接続するSSH」を参照してください。
- Microsoft リモートデスクトップ – macOS 用の Microsoft リモートデスクトップクライアントを使用して、 を使用して Windows インスタンスに接続しますRDP。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。別の例として、誰かが組織を離れ、プライベートキー (.PEM) ファイルのコピーを持っている場合が考えられます。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 のSSHキーを設定する
- Lightsail SSH キーを使用して安全なインスタンス接続を制御する
- Lightsail Linux インスタンスで SSH キーを管理する
- Lightsail で Linux または Unix インスタンスに接続する
- を使用して Lightsail Windows インスタンスに接続する RDP
- で Lightsail リソースを管理する AWS CloudShell

Lightsail のSSHキーを設定する

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、リモートサーバーと承認されたユーザーを照合するパブリックキーとプライベートキーを作成することで機能します。このキーペアを使用すると、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。

の詳細についてはSSH、「 についてSSH」を参照してください。

Lightsail インスタンスを作成する場合、デフォルトのオプションは Lightsail がSSHキーを管理できるようにすることです。Lightsail 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クライアントを提供します。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は、 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SSH。Lightsail での Windows ベースのインスタンスの詳細については、「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。

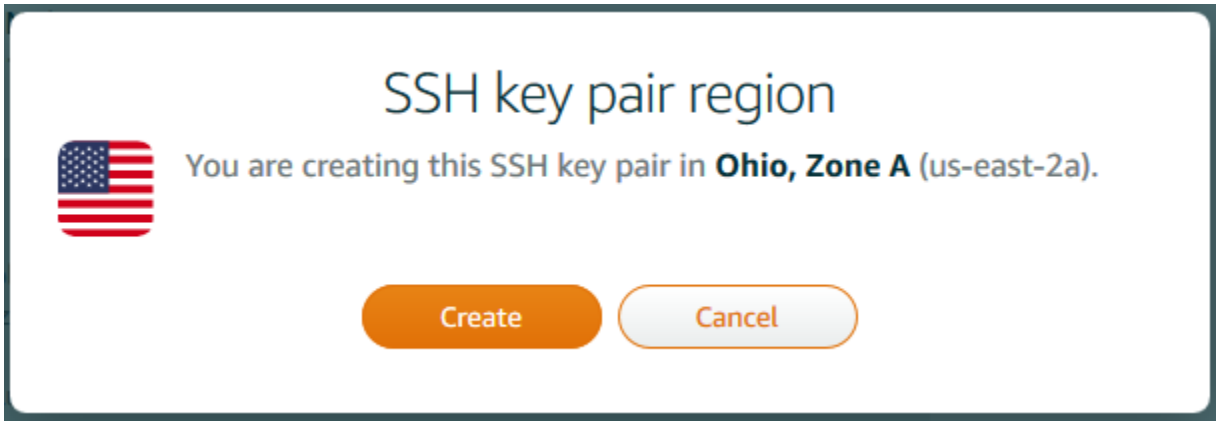
> **Important**
>
> SSH キー管理はリージョン別です。新しい でインスタンスを作成すると AWS リージョン、そのリージョンのデフォルトのキーペアを使用するオプションが表示されます。そのリージョンでカスタムキーを使用することもできます。独自のキーをアップロードする場合は、Lightsail インスタンスがあるリージョンごとにアップロードする必要があることに注意してください。

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。インスタンスの作成後にキーをダウンロードする場合は、アカウントページのSSHキーでダウンロードできます。

新規キーの作成

デフォルトキーを使用しない場合は、Lightsail インスタンスの作成時に新しい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. Lightsail は、新しいキーを作成するリージョンを表示します。



   [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。

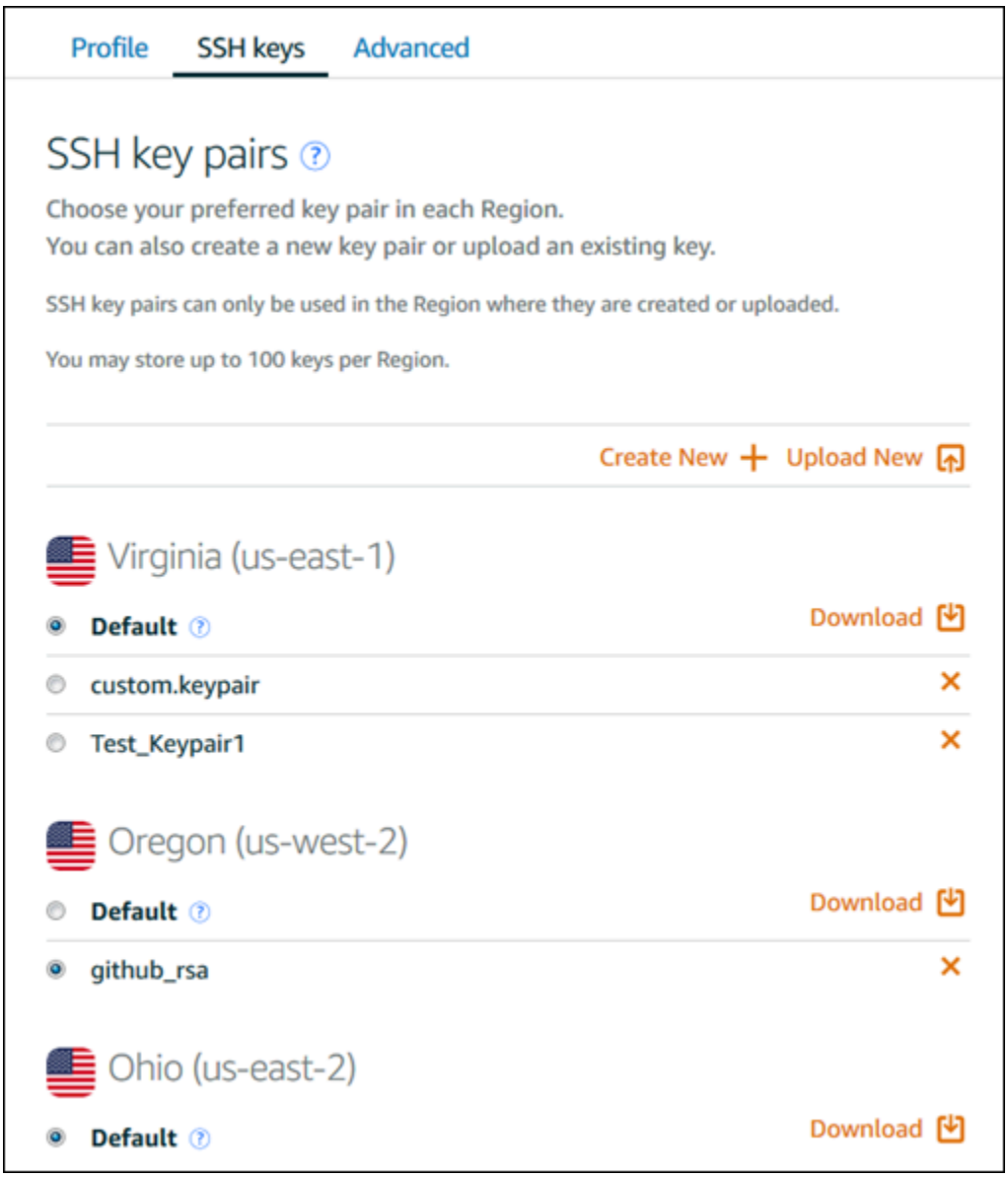
1. まだ作成していない場合は [インスタンスの作成] を選択します。
2. 「インスタンスの作成」ページで、SSHキーペアの変更を選択します。
3. [今すぐアップロード] を選択します。
4. Lightsail は、新しいキーをアップロードするリージョンを表示します。

   [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` と指定します。
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。



このページでは、新しい Lightsail インスタンスを作成するときにデフォルトで使用すべき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 などのSSHクライアントを使用して接続する場合、キーのプライベート半分が必要になります。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 に接続するための PuTTY の設定について説明します。

Lightsail SSH キーを使用して安全なインスタンス接続を制御する

キーペアを使用して、Amazon Lightsail インスタンスへの安全な接続を確立できます。Amazon Lightsail インスタンスを初めて作成するときは、Lightsail が作成するキーペア (Lightsail のデフォル

トキーペア) または作成したカスタムキーペアを使用できます。詳細については、「キーペア」と Amazon Lightsail 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。

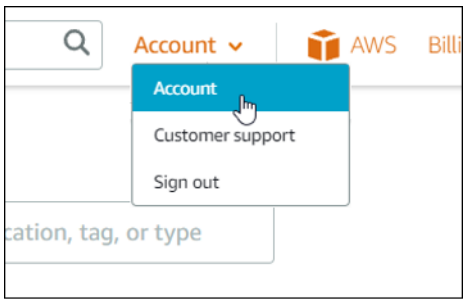
このガイドでは、Lightsail インスタンスで使用できるキーを管理する方法について説明します。キーの表示、既存キーの削除、および新しいキーの作成やアップロードを実行できます。

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とカスタムキーを表示するには、次の手順を実行します。

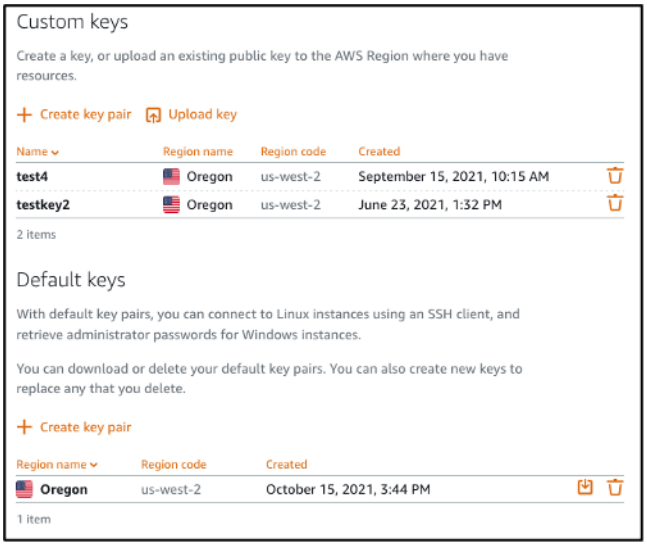
1. Lightsail コンソール にサインインします。
2. Lightsail ホームページのトップナビゲーションメニューで [Account (アカウント)] を選択します。
3. ドロップダウンメニューで [Account (アカウント)] を選択します。



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 – 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作成するキーです。各 に多数のカスタムキーを持つことができます AWS リージョン。
- デフォルトキー – 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 のインスタンスへの接続」を参照してください。

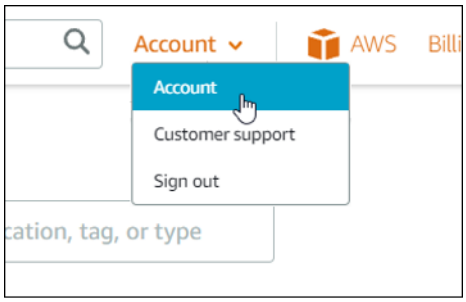
SSH キーページで、キーペアの作成、キーのアップロード、キーの削除、Lightsail のデフォルトキーペアのプライベートキーのダウンロードを行うことができます。

> **Note**
>
> Lightsail はそのキーを保存しないため、カスタムキーペアの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、このガイドの後半の「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「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」を参照してください。

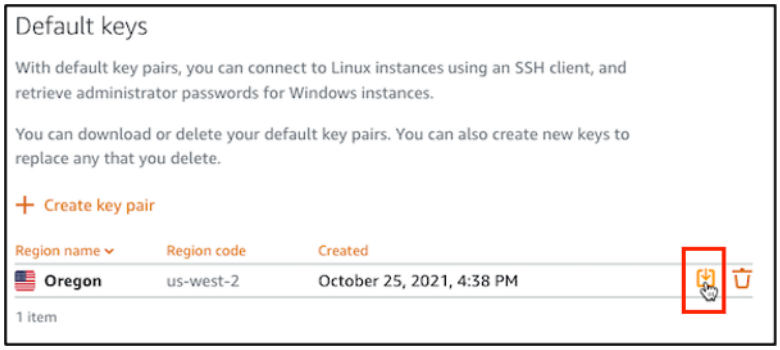
Lightsail コンソールからデフォルトキーのプライベートキーをダウンロードする

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、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、上部のナビゲーションペインの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。
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。



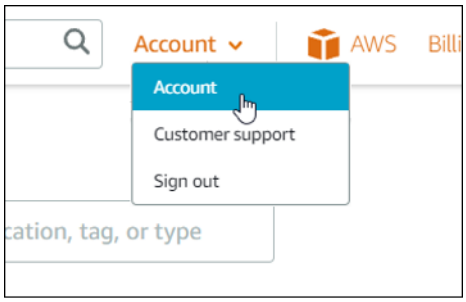
> **Important**
>
> プライベートキーは安全な場所に保存してください。このキーはインスタンスへの接続に使用できるため、公開しないでください。

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、次の手順を実行します。これにより、Lightsail で作成した新しいインスタンスでカスタムキーが設定されなくなります。

1. Lightsail コンソール にサインインします。
2. Lightsail ホームページで、上部のナビゲーションペインの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。



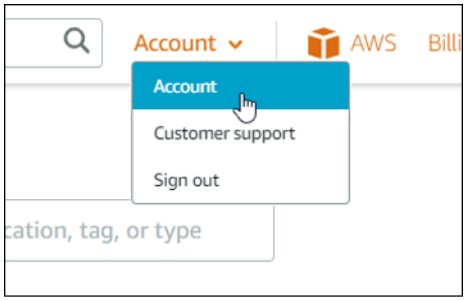
4. [SSH キー] タブを選択します。
5. そのページの [Custom keys] (カスタムキー) セクションで、削除するキーの削除アイコンを選択します。

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したパブリックキーを削除するには、Amazon Lightsail」を参照してください。

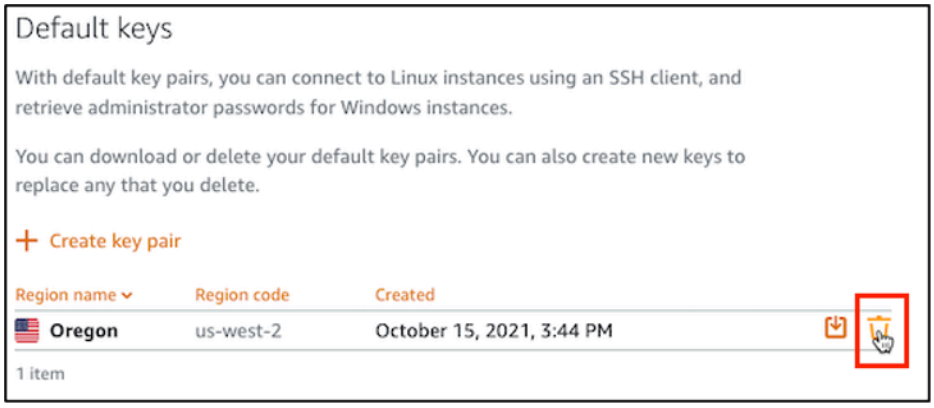
Lightsail コンソールでデフォルトキーを削除し、新しいキーを作成します。

Lightsail コンソールでデフォルトキーを削除するには、次の手順を実行します。これにより、Lightsail で作成した新しいインスタンスでデフォルトキーが設定されなくなります。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 で作成した新しいインスタンスで、新しいデフォルトキーを設定できます。

1. Lightsail コンソール にサインインします。
2. Lightsail ホームページで、上部のナビゲーションペインの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。

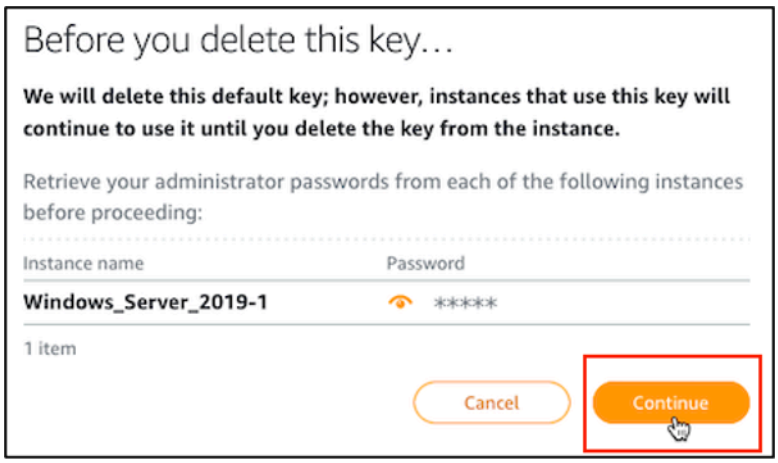


4. [SSH キー] タブを選択します。
5. そのページの [Default keys] (デフォルトキー) セクションで、削除するデフォルトキーの削除アイコンを選択します。

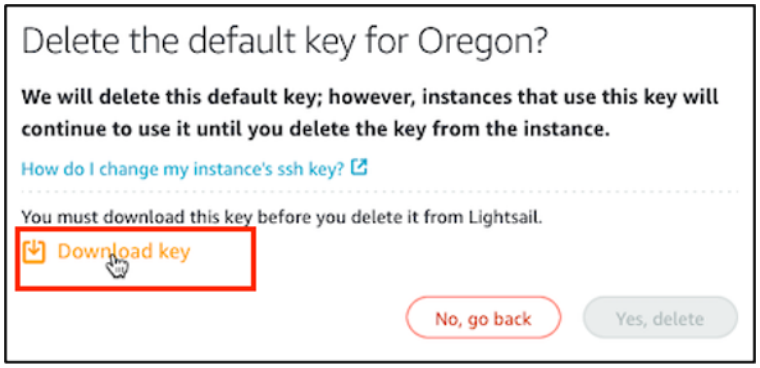


> **Important**
>
> デフォルトキーを削除しても、以前に作成された現在実行中のインスタンスからカスタムキーペアの公開キーが削除されることはありません。詳細については、Amazon Lightsail のインスタンスに保存されているキーの管理」を参照してください。

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。



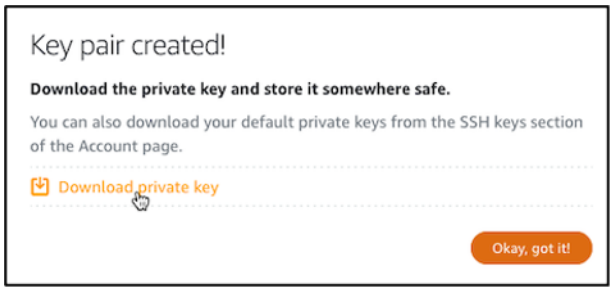
9. デフォルトキーが削除されました。[OK] を選択します。

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。

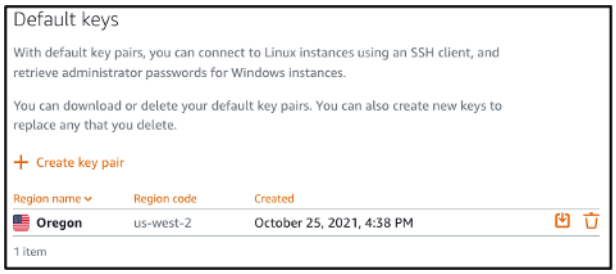
10. このページの [Default keys] (デフォルトキー) セクションで、[Create key pair] (キーペアを作成) を選択します。
11. 表示されるリージョンの選択プロンプト AWS リージョン で、新しいデフォルトキーを作成する を選択します。同じ AWS リージョン内の新しいインスタンスには、新しいデフォルトキーを設定することができます。

> **Note**
>
> これらのステップを使用すると、Lightsail リソースを AWS リージョン作成した でのみデフォルトのキーペアを作成できます。新しいリージョンでデフォルトの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。

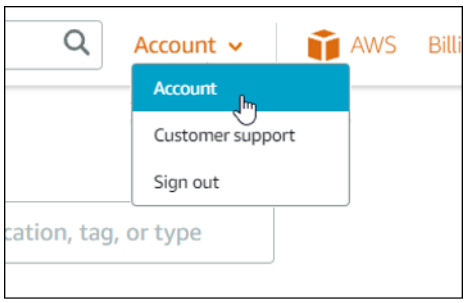


Lightsail で作成した新しいインスタンスで、新しいデフォルトキーを設定できます。以前に作成され、現在実行中のインスタンスで新しいデフォルトキーを設定するには、Amazon Lightsail のインスタンスに保存されているキーを管理する」を参照してください。

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、次の手順を実行します。Lightsail で作成した新しいインスタンスで、新しいカスタムキーを設定できます。

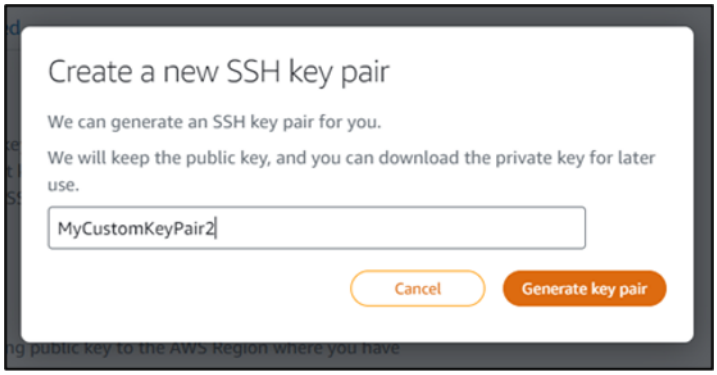
1. Lightsail コンソール にサインインします。
2. Lightsail ホームページで、上部のナビゲーションペインのアカウントを選択します。
3. ドロップダウンメニューで [Account (アカウント)] を選択します。



4. [SSH キー] タブを選択します。
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。

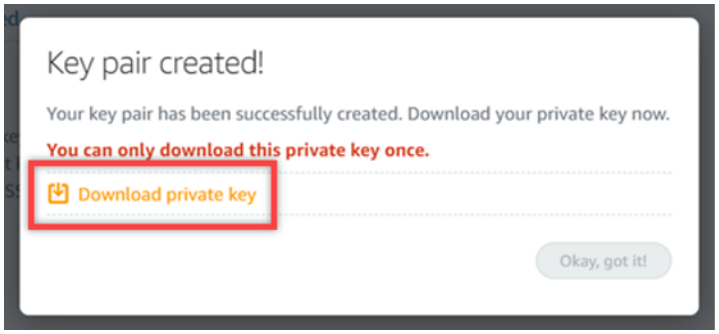
6. 表示される [Select a region] (リージョンの選択) プロンプトで、新しいカスタムキーを作成する AWS リージョン を選択します。同じ AWS リージョン内の新しいインスタンスには、新しいカスタムキーを設定することができます。

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。

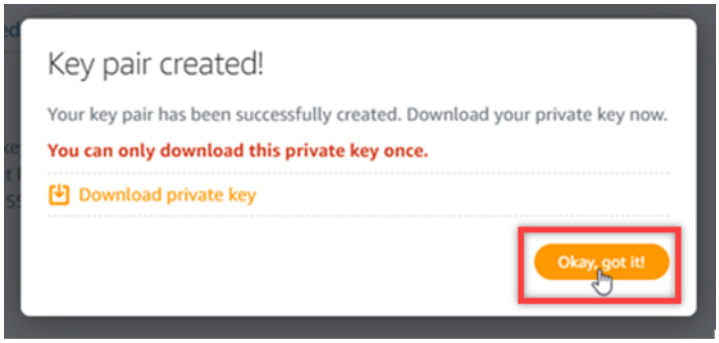


8. 表示される [Key pair created!] (キーペアが作成されました!) プロンプトで [Download private key] (プライベートキーのダウンロード) を選択して、プライベートキーをローカルコンピュータに保存します。

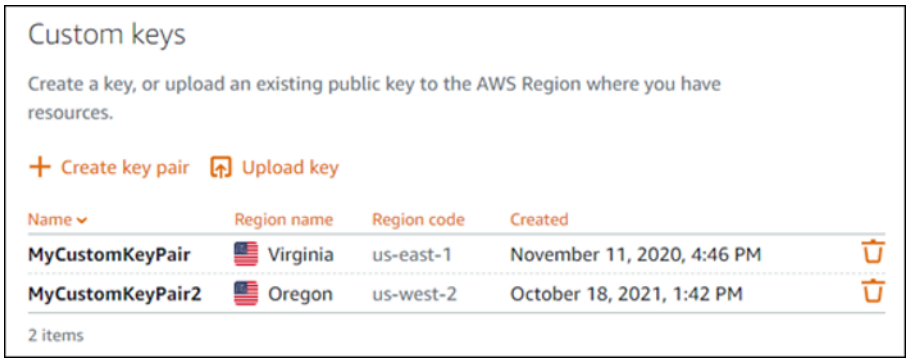
> **Important**
>
> プライベートキーは安全な場所に保存してください。このキーはインスタンスへの接続に使用できるため、公開しないでください。
> カスタムキーのプライベートキーをダウンロードできるのは、この時だけです。Lightsail は、カスタムキーペアのプライベートキーを保存しません。このプロンプトを閉じてしまうと、再度ダウンロードすることはできません。



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 で作成した新しいインスタンスで、新しいカスタムキーを設定できます。以前に作成され、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 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、Lightsail コンソールにアップロードできます。Lightsail で作成した新しいインスタンスで、新しいカスタムキーを設定できます。

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。

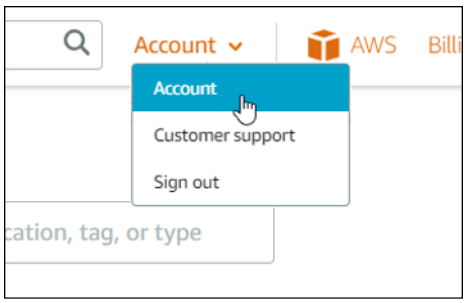
```
Enter same passphrase again:
```

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、キーペアのパブリックキーを Lightsail コンソールにアップロードします。

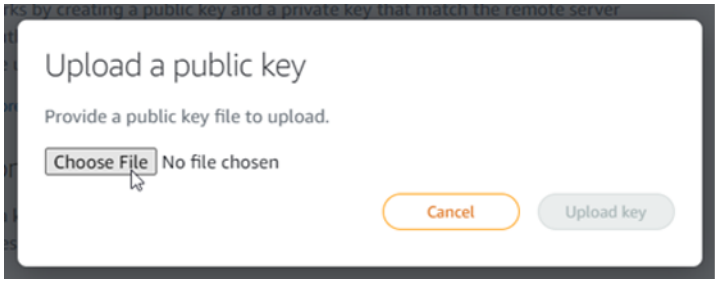
7. Lightsail コンソール にサインインします。
8. Lightsail ホームページで、上部のナビゲーションペインのアカウントを選択します。
9. ドロップダウンメニューで [Account (アカウント)] を選択します。



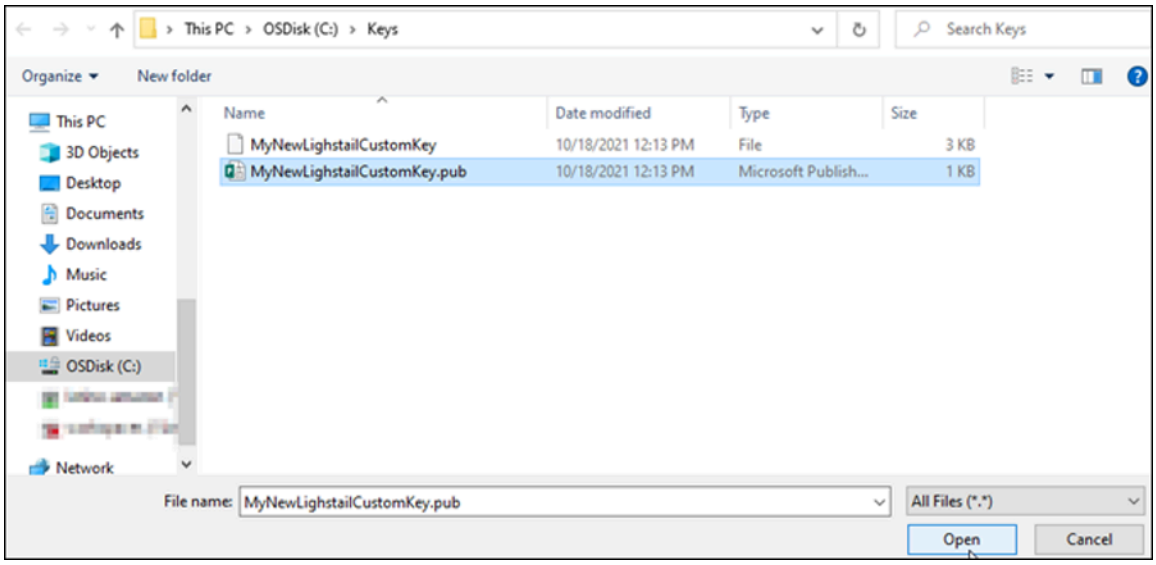
10. [SSH キー] タブを選択します。
11. そのページの [Custom keys] (カスタムキー) セクションで、[Upload key] (キーのアップロード) を選択します。

12. 表示されるリージョンの選択プロンプト AWS リージョン で、新しいカスタムキーをアップロードする 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。

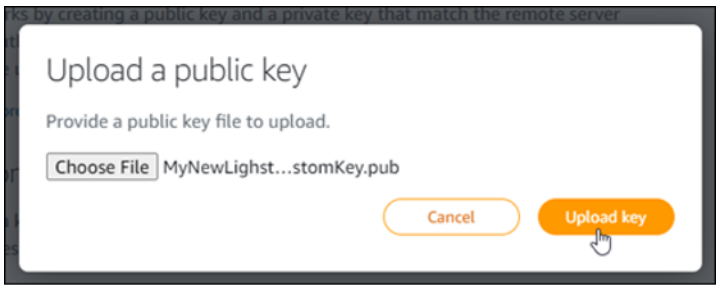
13. [アップロード] を選択します。
14. 表示される [Upload a public key] (公開キーのアップロード) プロンプトで、[Choose File] (ファイルを選択) をクリックします。



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、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 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 Linux インスタンスで SSH キーを管理する

キーペアを使用して、Amazon Lightsail インスタンスへの安全な接続を確立できます。Lightsail は、Linux または Unix インスタンスを初めて作成するときに、キーペアのパブリックキーを設定します。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 コンソールを使用するか、ssh-keygen などのサードパーティー製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またはデフォルトキーペアを作成できます。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>*

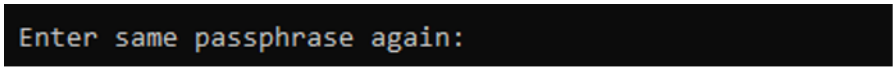
   <UserName> を現在サインインしているユーザーの名前に置き換えて、*<KeyPairName>* を新しいキーペアの名前に置き換えます。

   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。

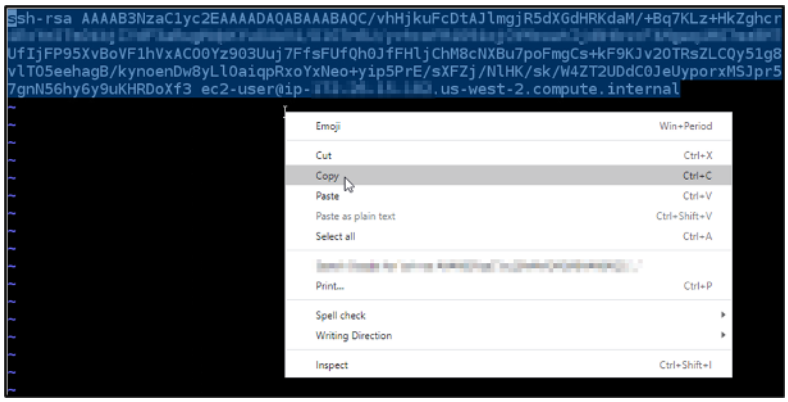
5. 確認のためパスフレーズをもう一度入力して、Enter を押します。パスフレーズの入力中にパスフレーズは表示されません。



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。

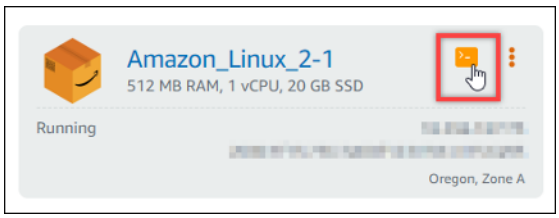


このガイドの次のセクションに進み、新しいパブリックキーを Lightsail インスタンスに追加します。

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。この場合、 AWS リージョン インスタンスが作成された の Lightsail デフォルトキーは、インスタンスで設定された唯一のパブリックキーです。

5. [I] キーを押して、Vim 工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 工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 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。

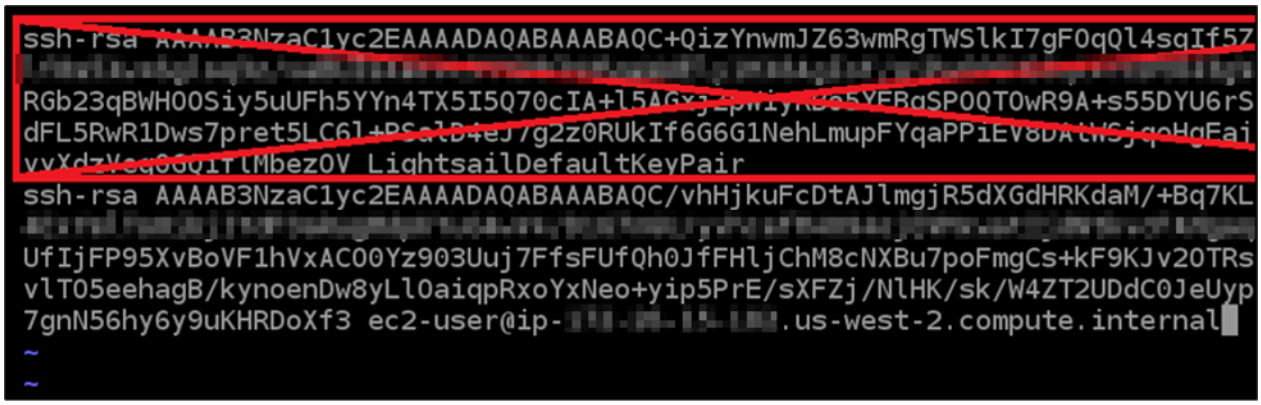
ステップ 5: インスタンスから既存の公開キーを削除する

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。

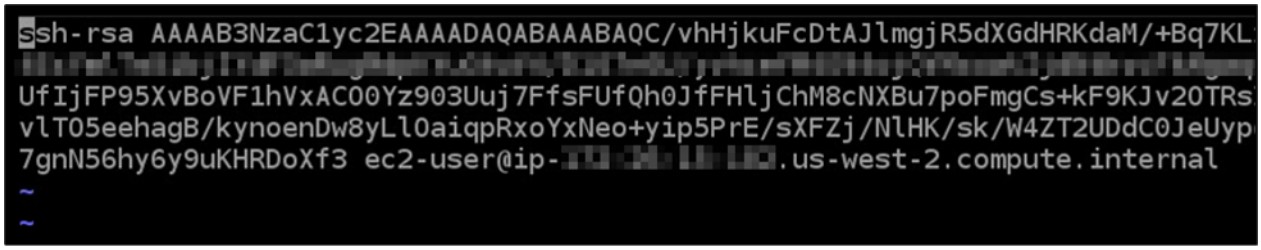
1. SSH を使用してインスタンスに接続します。
2. 任意のテキストエディアを使用して、authorized_keys ファイルを編集するための以下のコマンドを入力します。以下の手順では、デモ用に Vim を使用します。

```
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 で 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、「Pu のダウンロードとセットアップTTY」を参照してください。

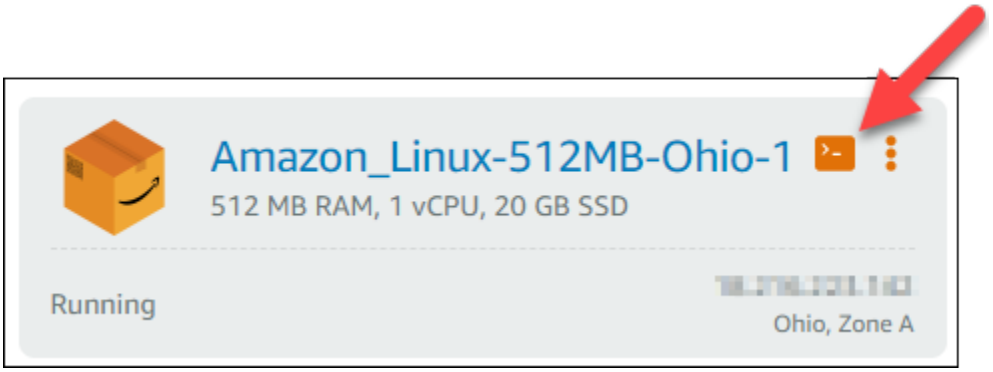
を使用してインスタンスに接続SSHし、ソフトウェアパッケージのインストールやウェブアプリケーションの設定などの管理タスクをサーバーで実行します。ブラウザベースのSSHクライアントはソフトウェアのインストールを必要とせず、インスタンスの作成直後に利用できます。

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

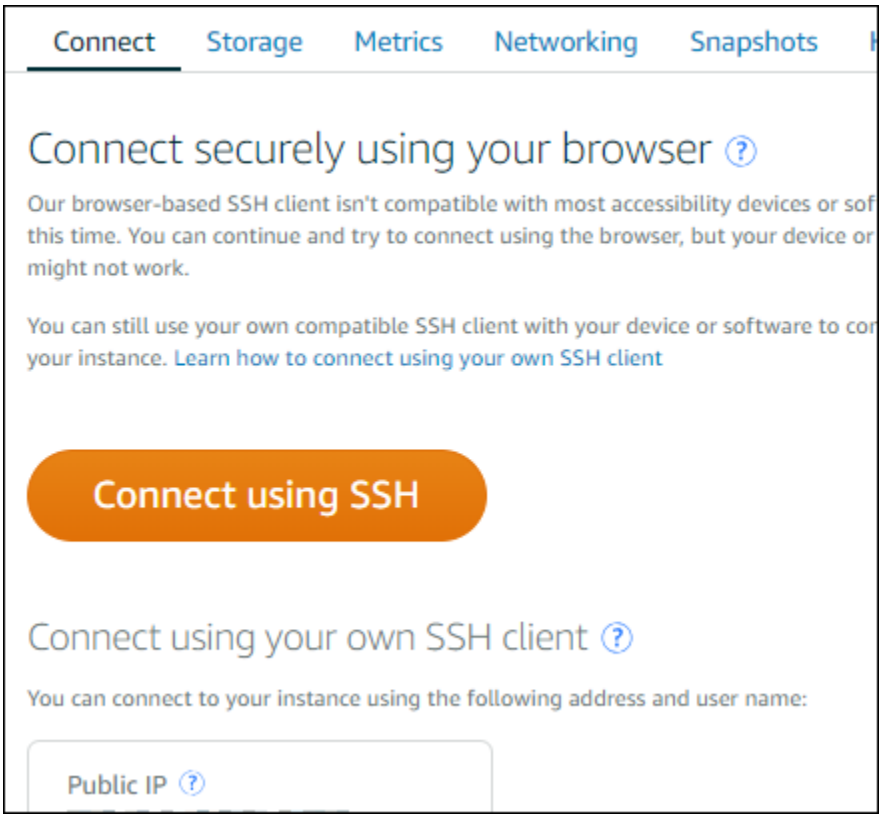
1. Lightsail コンソール にサインインします。
2. 次のいずれかを使用して、接続先のインスタンスのブラウザベースのSSHクライアントにアクセスします。

   - 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。

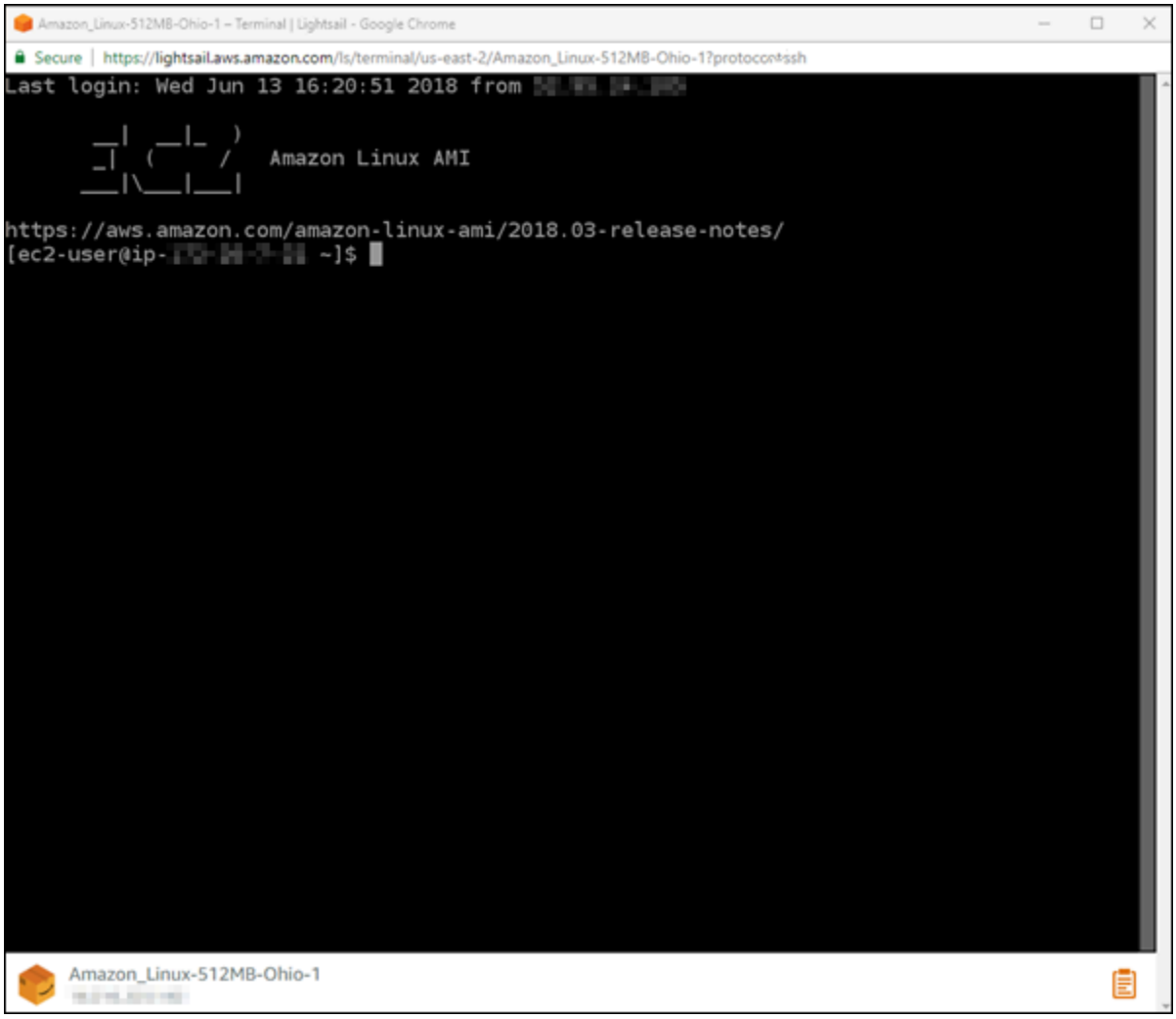


   - アクションメニューアイコン (⋮) を選択し、次に [接続] を選択します。

- インスタンスの名前を選択し、Connect タブで を使用して Connect SSHを選択します。



ブラウザベースのSSHクライアントが開き、次の例に示すように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**
>
> Connect 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、「Pu のダウンロードとセットアップTTY」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 インスタンスを操作する

Linux または Unix コマンドをターミナル画面に直接入力するか、テキストをターミナル画面に貼り付けるか、ブラウザベースのSSHクライアントのターミナル画面からテキストをコピーします。以下のセクションでは、 のクリップボードとの間でテキストをコピーして貼り付ける方法を示しますSSH。

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隅で、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て、ブラウザベースのSSHクライアント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隅で、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。

SSH コマンドを使用して Lightsail Linux または Unix インスタンスに接続する

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を介して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。その他の方法の詳細については、「 SSH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、「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。

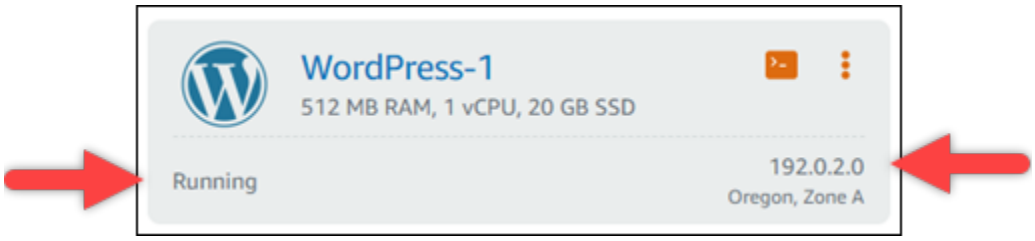
内容

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

次の手順では、Lightsail コンソールにサインインして、インスタンスが実行中の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の状態である必要があり、このガイドの後半でインスタンスに接続するために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。インスタンスを認証し、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です。

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。

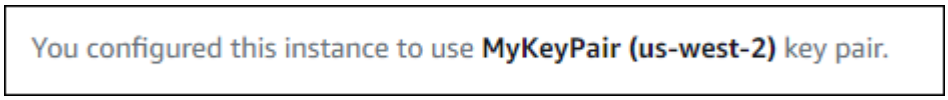
2. [接続] タブで、下にスクロールして、インスタンスで使用されているキーペアを確認します。考えられる可能性は 2 つあります。

   1. 次の例は、インスタンスを作成したAWSリージョンのデフォルトキーペアを使用す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のキーペアに対してのみプライベートキーを保存します。

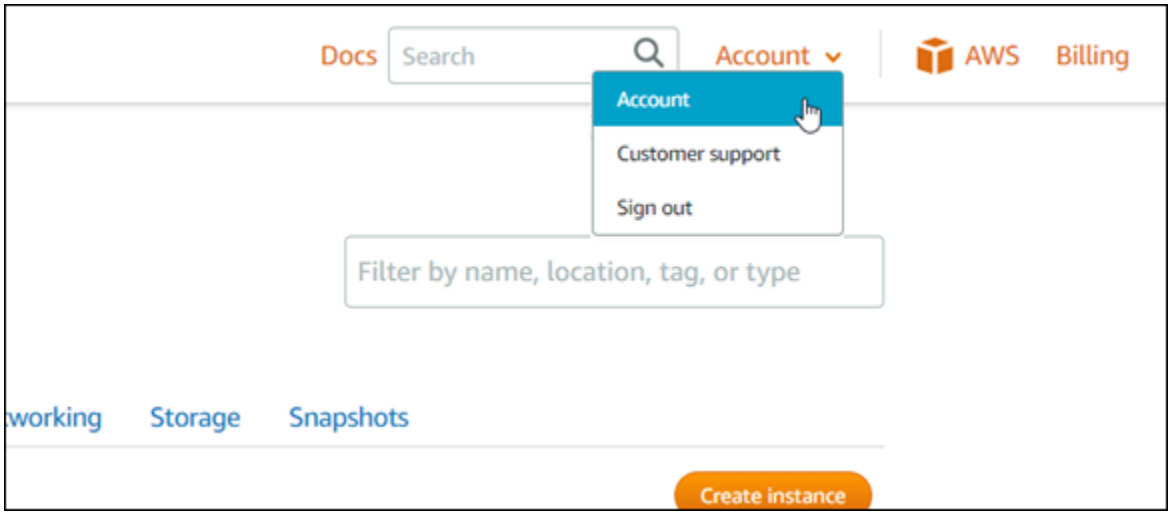
      > You configured this instance to use **default (us-west-2)** key pair.
      > You can download your default private key from the Account page.

   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紛失した場合、独自のクライアントを使用してインスタンスSSHへの接続を確立することはできません。ただし、Lightsail コンソールで利用可能なブラウザベースのSSHクライアントを引き続き使用できます。カスタムキーペアの

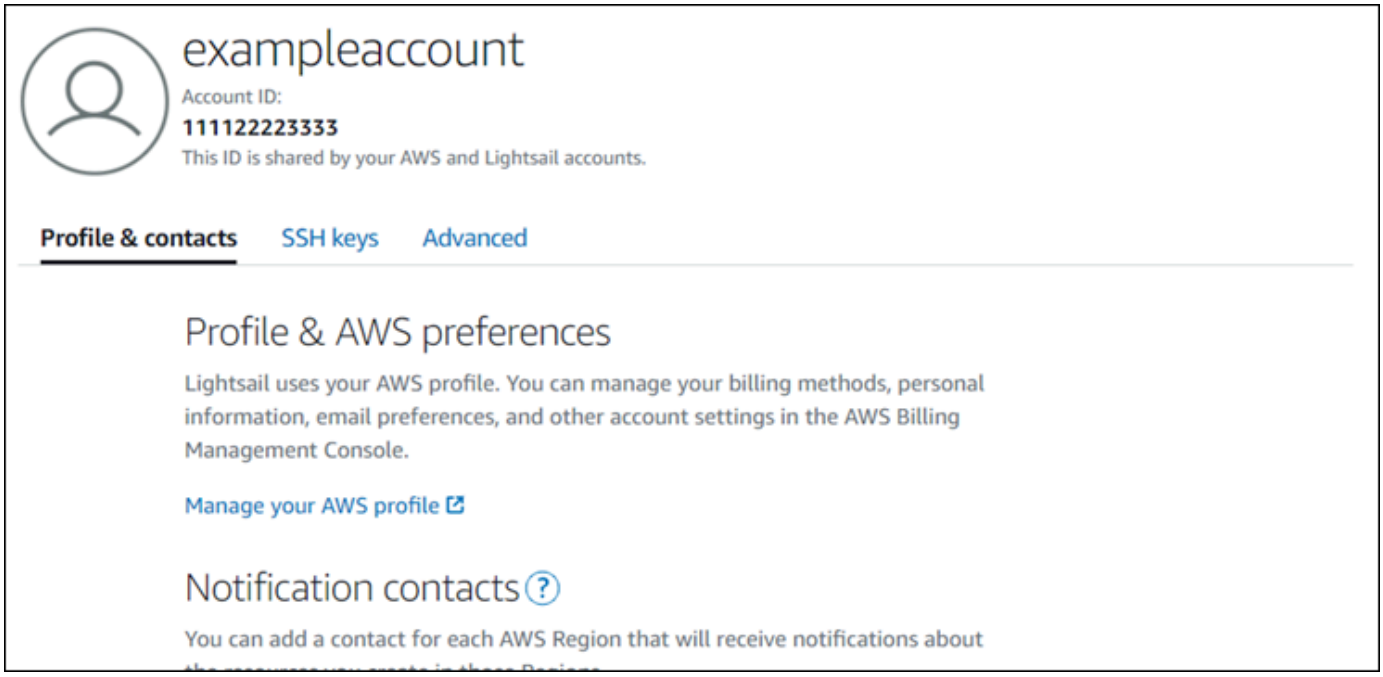
プライベートキーを見つけたら、このガイドの「」セクションを使用して、次のステップ 3: プライベートキーのアクセス許可を変更し、インスタンスに接続しますSSH。



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。

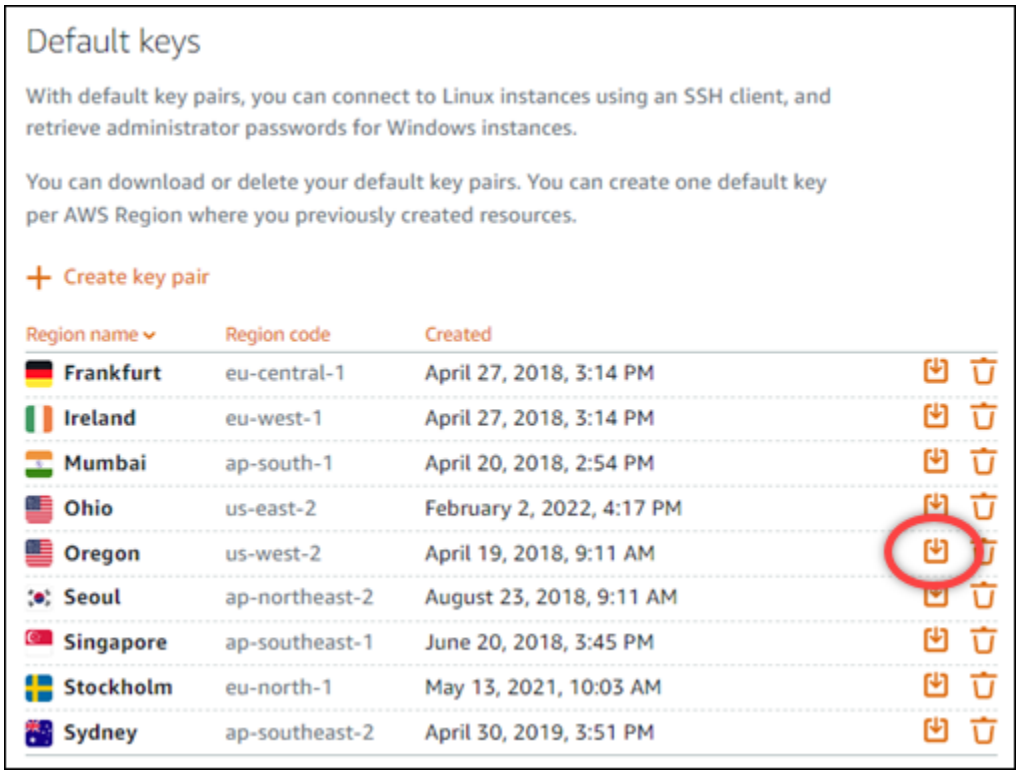


アカウント管理ページが表示され、アカウント設定を管理するためのさまざまなタブオプションが表示されます。



4. SSH キータブを選択します。

5. 下にスクロールし、接続するインスタンスの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。ダウンロードしたキーを、ユーザーのホームディレクトリの「Keys」フォルダなど、すべてのSSHキーを保存するディレクトリに移動することもできます。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 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を提供していません。プライベートキーファイルの形式を変換する必要がある場合は、Open SSLなどの無料のオープンソースツールがすぐに利用できます。

このガイドの「ステップ 3: プライベートキーのアクセス許可を変更し、」セクションを使用してインスタンスに接続SSHし、ダウンロードしたプライベートキーを使用してインスタンスSSHへの接続を確立します。

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

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 で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. 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。

   ```
   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* インスタンスで使用されているキーペアのプライベートキーを保存したディレクトリパス。
   - *username* インスタンスのユーザー名を入力します。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、無料 BSD、オープンSUSEインスタンス: `ec2-user`
     - 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 および WHM インスタンス: `centos`

- 置換 *public-ip-address* このガイドの前半で Lightsail コンソールから書き留めたインスタンスのパブリック IP アドレス。

絶対パスの例:

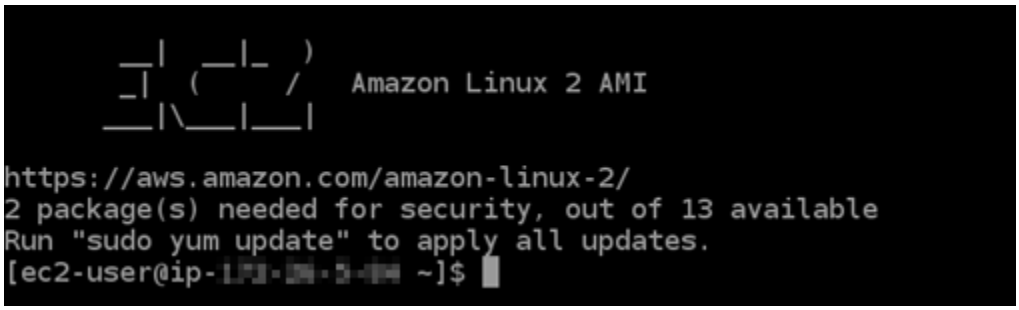
```
ssh -i /Users/user/Keys/LightsailDefaultKey-us-west-2.pem ec2-user@192.0.1.0
```

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。接続後、Lightsail で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。



Pu を使用して Linux/Unix Lightsail インスタンスに接続するTTY

Lightsail のブラウザベースのSSHターミナルに加えて、Pu 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TTY。Pu をセットアップする方法についてはTTY、Lightsail SSHの「 を使用して接続するための PuTTY のダウンロードとセットアップ」を参照してください。

> **Note**
>
> を使用して Windows ベースのインスタンスに接続するにはRDP、「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. スタート PuTTY (例えば、スタートメニューからすべてのプログラム 、Pu 、Pu) TTYを選択します。 TTY
2. [ロード] を選択し、保存済みセッションを見つけます。

   保存された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、無料 BSD、オープン SUSEインスタンス: `ec2-user`
   - 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 および 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。

の詳細についてはSSH、「」を参照SSHし、Amazon Lightsail インスタンス に接続します。

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

PuTTY などのSSHクライアントを使用して、Amazon Lightsail インスタンスに接続できます。PuTTY にはプライベートSSHキーのコピーが必要です。キーが既にある場合や、Lightsail が作成するキーペアを使用する場合があります。どちらの方法についても説明しています。の詳細につ

いてはSSH、「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ための PuTTY を設定する手順について説明します。

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。その他の方法の詳細については、「SSH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」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。
- 後でパブリック IP アドレスが変更されても PuTTY を再設定する必要がないように、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ことをお勧めします。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 1: Pu をダウンロードしてインストールするTTY

PuTTY は for Windows SSH の無料実装です。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 を参照してください。既に Pu をお持ちの場合はTTY、ステップ 2 に進んでください。

1. 次のリンクから 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: Pu のダウンロードTTY。

   どのダウンロードを選択するかを決める際にサポートが必要な場合は、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. Pu を設定する前に、ステップ 2 に進み、プライベートキーを取得しますTTY。

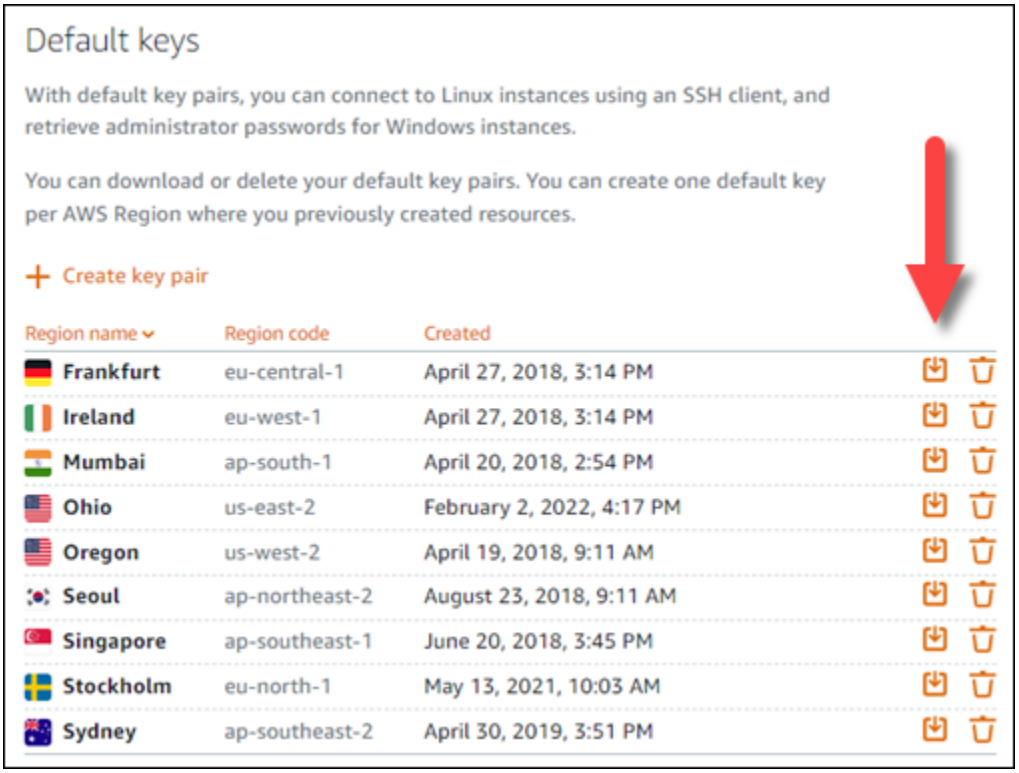
ステップ 2: プライベートキーの準備を整える

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る場合があります。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 を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、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、簡単に見つけられる場所に.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uTTYgen を使用して P を設定する

.pem キーファイルのコピーを取得したら、PuTTY Key Generator (P) を使用して PuTTY を設定できますuTTYgen。

1. スタート P uTTYgen （ 例えば、スタートメニューからすべてのプログラム 、Pu 、PuTTYgen を選択します ）。 TTY
2. ロード] を選択します。

   デフォルトでは、P は .ppk拡張子を持つファイルのみuTTYgen を表示します。.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。

   P はキーが正常にインポートされたuTTYgen ことを確認し、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手段としてパスフレーズを作成する場合は、Pu 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TTY。
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. P を閉じますuTTYgen。

ステップ 4: プライベートキーとインスタンス情報を使用して 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。

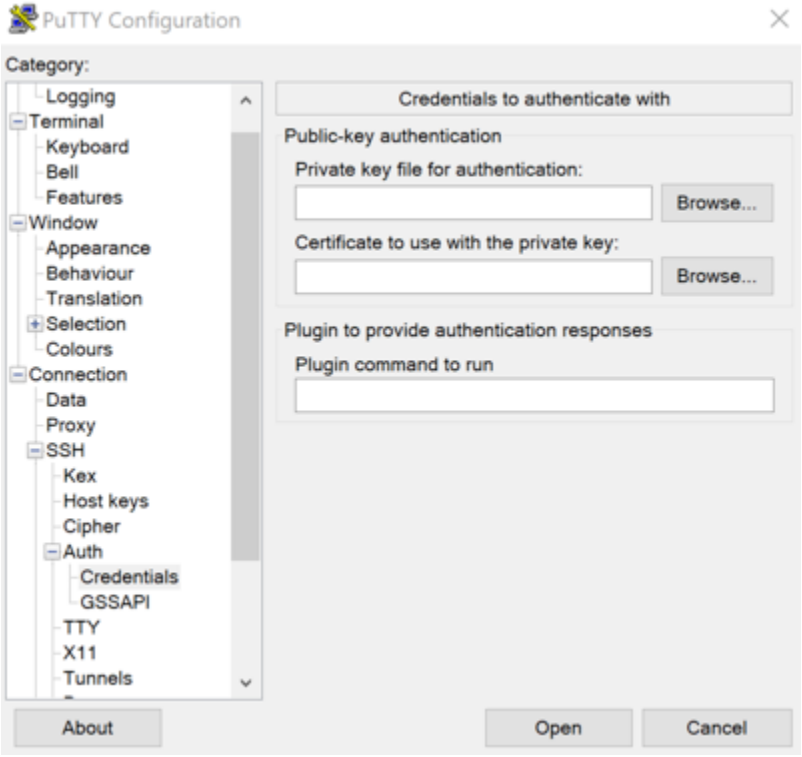
1. Pu を開きますTTY。
2. Lightsail から、インスタンス管理ページからパブリック IP アドレスを取得します (静的 IP アドレス を使用していることが望まれます ）。

Lightsail 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SSHでポート 22 が 用に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 で、 SSHを展開して認証し、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、無料 BSD、オープンSUSEインスタンス: `ec2-user`

- 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 および 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、「Pu を使用して Linux/Unix ベースのインスタンスに接続する TTY」を参照してください。

を使用して Lightsail Linux インスタンスにファイルを安全に転送する SFTP

SFTP (SSHファイル転送プロトコル) 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し、それを使用してFTPクライアントを設定する必要があります。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クライアントを設定する FileZilla FTP方法を示します。これらの手順は、他のFTPクライアントにも適用される場合があります。

内容

- 前提条件
- インスタンスの SSHキーを取得する
- インスタンスを設定して FileZilla 接続する

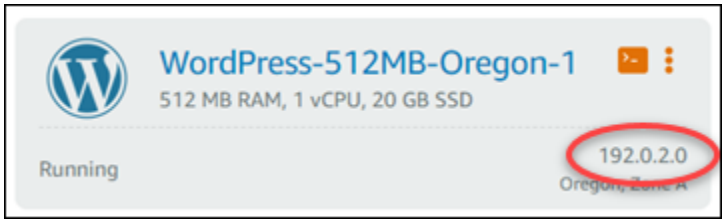
前提条件

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 FileZilla クライアントをダウンロードする
- 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。

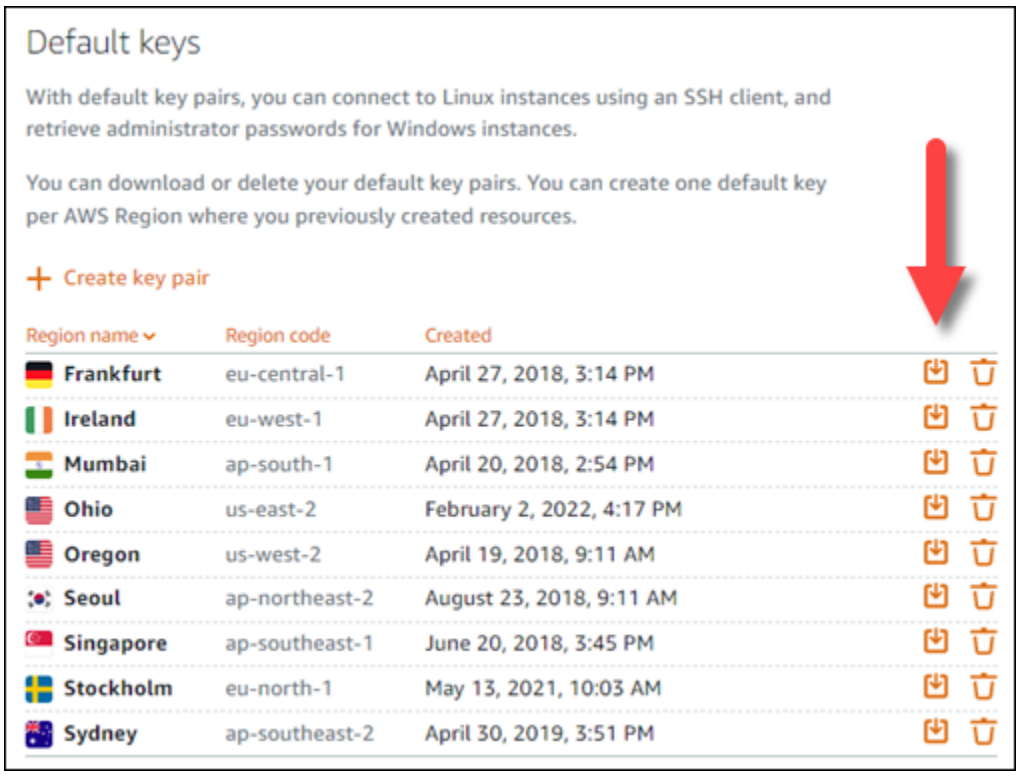


インスタンスの SSHキーを取得する

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見つけて、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すると、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SFTPがないと、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。

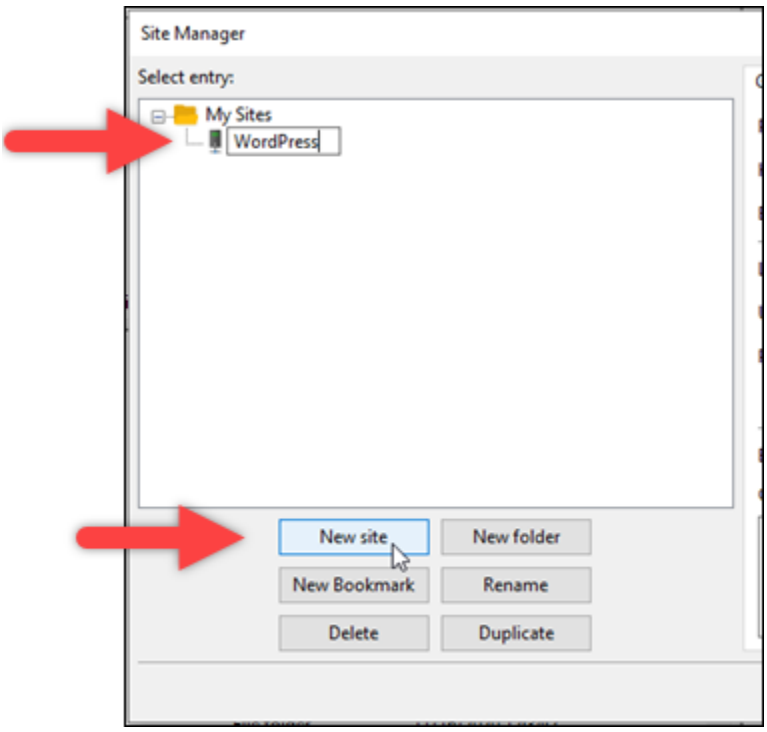


6. ローカルドライブのセキュリティが確保された場所にプライベートキーを保存します。

インスタンスを設定して FileZilla 接続する

インスタンスに接続するように を設定するには FileZilla 、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。

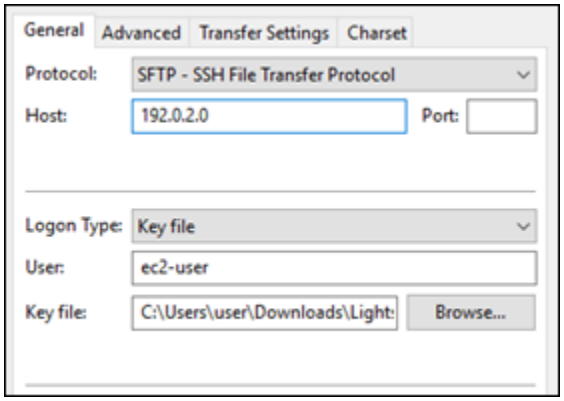


4. プロトコルドロップダウンで、SFTPSSH「ファイル転送プロトコル」を選択します。
5. [ホスト] テキストボックスに、インスタンスのパブリック IP アドレスを入力するか、貼り付けます。
6. [ログオンタイプ] ドロップダウンで、[キーファイル] を選択します。
7. [ユーザー] テキストボックスに、インスタンスのオペレーティングシステムに応じて、次のいずれかのデフォルトのユーザー名を入力します。

   - AlmaLinux、Amazon Linux 2、Amazon Linux 2023、CentOS Stream 9、無料 BSD、オープン SUSEインスタンス: `ec2-user`
   - 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 および WHM インスタンス: `centos`

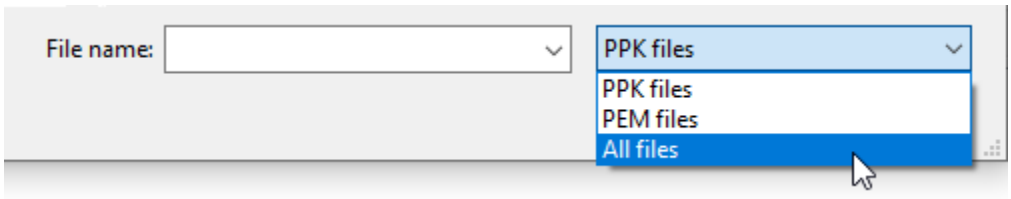
> **Important**
>
> ここにリストされているデフォルトのユーザー名とは異なるユーザー名を使用している場合は、ユーザーにインスタンスへの書き込み許可を付与します。

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。

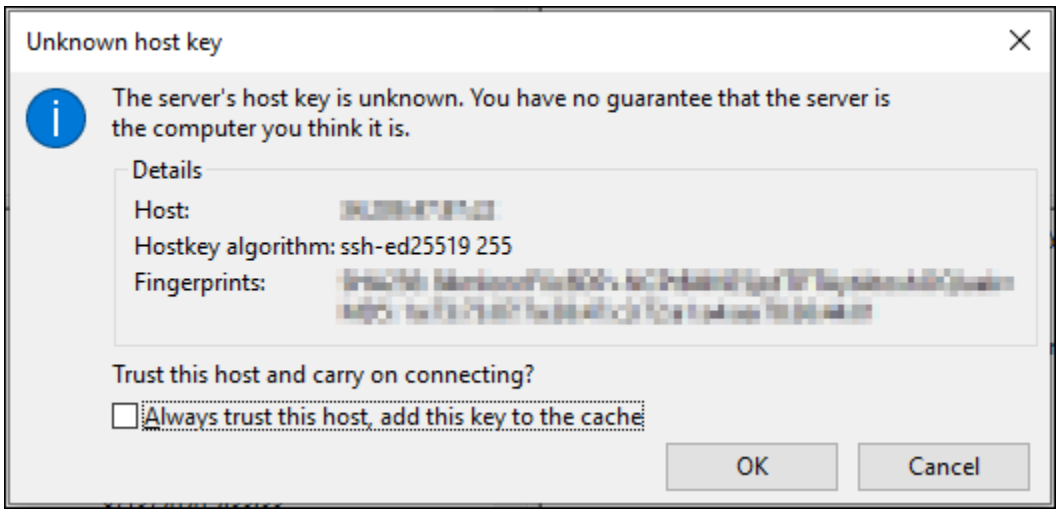


9. この手順の前半で Lightsail コンソールからダウンロードしたプライベートキーファイルを見つけ、 を開くを選択します。

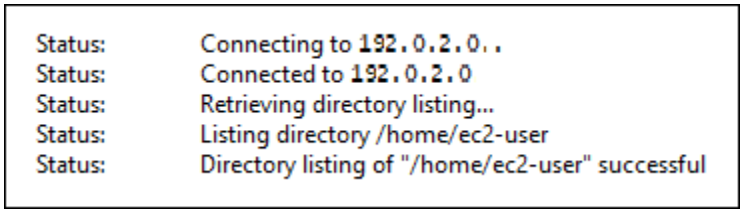
> **Note**
>
> Windows を使用している場合は、pem ファイルを検索する際に既定のファイルの種類を [すべてのファイル] に変更します。



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。

を使用して Lightsail Windows インスタンスに接続する RDP

Amazon Lightsail の Windows Server インスタンスに接続できます。 RDP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 インスタンスを作成したらすぐに接続でき、使用可能になります。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。

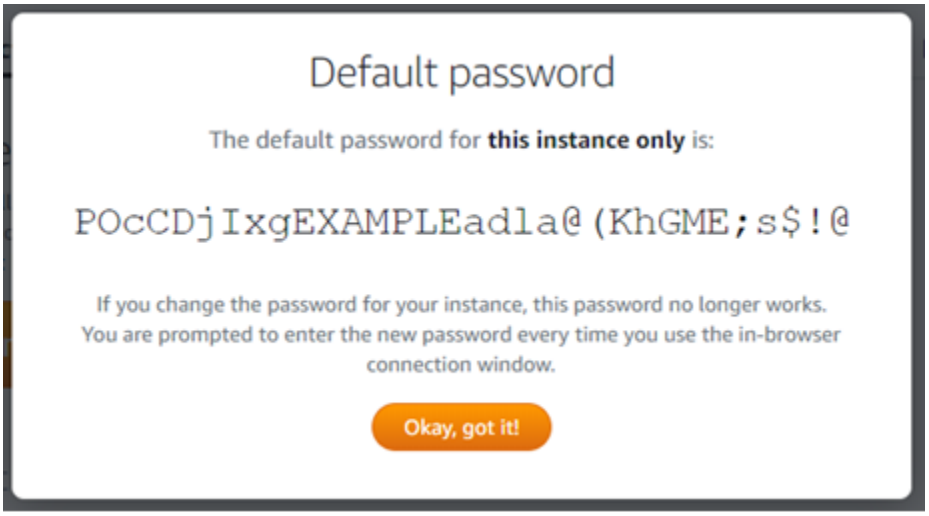
Windows にバンドルされているリモートデスクトップ接続など、独自のRDPクライアントを使用してインスタンスに接続することもできます。独自のRDPクライアントの設定の詳細については、「リモートデスクトップ接続クライアントを使用して Windows インスタンスに接続する」を参照してください。Lightsail で Linux または Unix インスタンスに接続するには、「Linux または Unix インスタンスに接続する」を参照してください。

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求められます。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、 AWS Secrets Manager など、紛失した場合に後で取得できる安全な場所に保存します。詳細については、AWS Secrets Manager 「 ユーザーガイド」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インスタンスにアクセスするたびに、管理者パスワードが要求されないように、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戻すことができます。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確認できます。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。

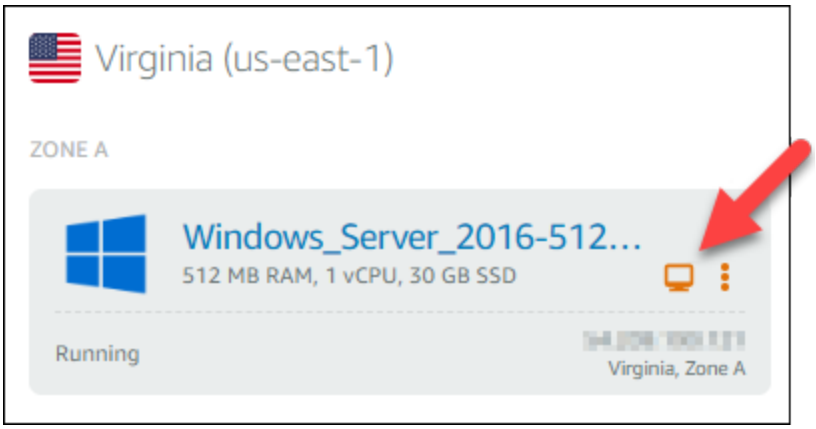


ブラウザベースのRDPクライアントを使用して Windows Server インスタンスに接続する

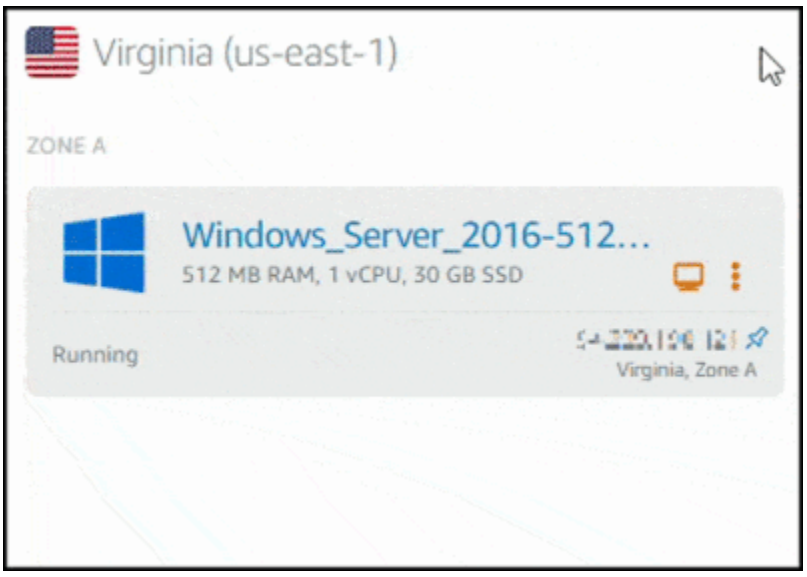
Lightsail コンソールのブラウザベースのRDPクライアントを使用して Windows Server インスタンスに接続するには、次の手順に従います。

1. Lightsail コンソール にサインインします。
2. 次のいずれかのステップを使用して、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。

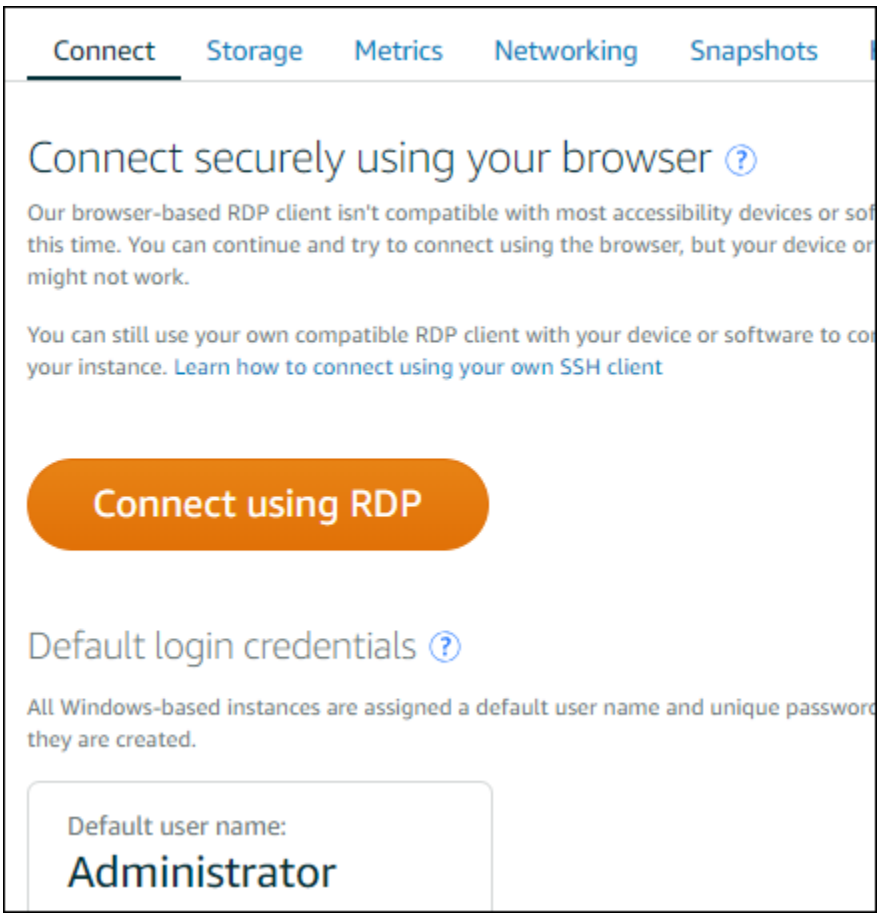
   - 次の例に示すように、ブラウザベースのRDPクライアントアイコンを選択します。



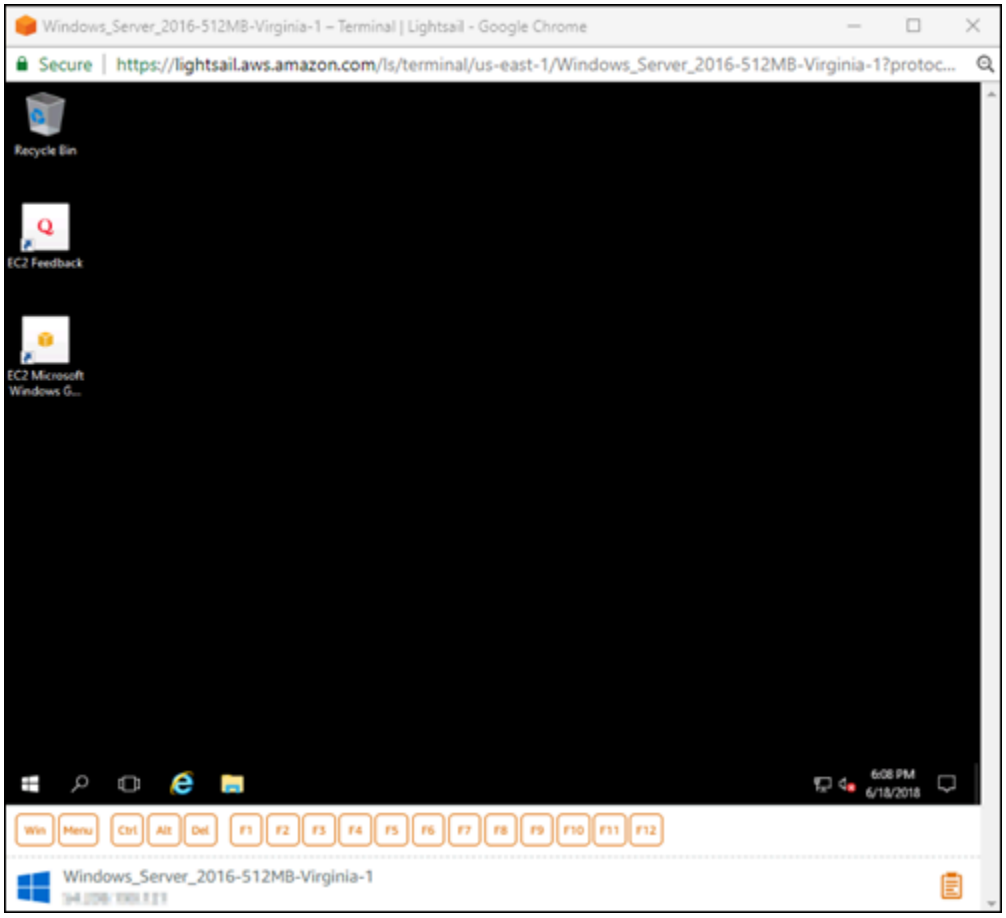
   - 以下の例に示すように、アクションメニューアイコン (⋮) を選択してから、[接続] を選択します。



   - インスタンスの名前を選択し、Connect タブで を使用して Connect RDPを選択します。



ブラウザベースのRDPクライアントが開き、次の例に示すように 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**
>
> Connect タブには、Windows インスタンスのデフォルトのユーザー名とパスワードなど、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て Amazon Lightsail で 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RDPクライアントを使用して Windows インスタンスを操作する

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、インスタンスの操作に役立つ関数キーと Windows に固有のその他のキーが含まれています。以下のセクションでは、 のクリップボードとの間でテキストをコピーして貼り付ける方法を示しますRDP。

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隅で、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て、ブラウザベースのRDPクライアント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隅で、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を変更する

Windows Server ベースの Lightsail インスタンスを作成すると、インスタンス AWS リージョン を作成する のデフォルトパスワードが使用されます。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することが容易になります。

> **Important**
>
> Lightsail にインスタンスのパスワードを生成させ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すると 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Delで Ctrl Alt + + と入力し、パスワードの変更 を選択します。

を使用して Lightsail キーペアの暗号文を取得する AWS CLI

Windows Server ベースの Lightsail インスタンスでパスワードを変更する場合は、 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。

暗号文を取得する

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 で動作する AWS Command Line Interface ように を設定する Amazon Lightsail」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
   ```

   各パラメータの意味は次のとおりです。*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。

> **Note**
>
> Lightsail には、.pem ファイルを操作するためのユーティリティはありません。プライベートキーファイルの形式を変換する必要がある場合は、Open for Linux や base64 for Windows SSLなどの無料のオープンソースツールがすぐに利用できます。

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

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、「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。

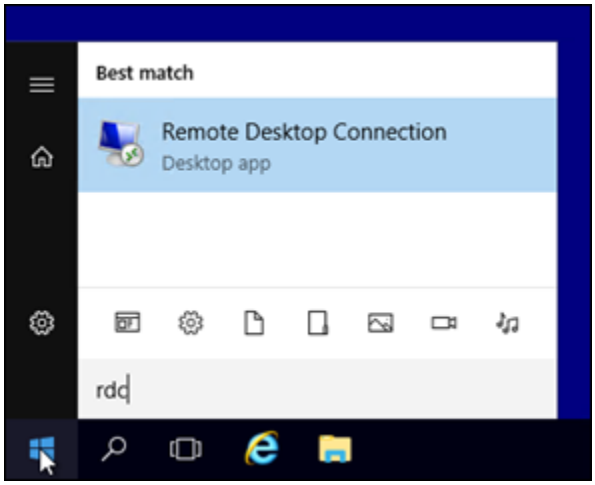
4. 表示されるデフォルトのパスワードをハイライト表示し、[Ctrl+C] または [Cmd+C] を押してコピーします。これで、パスワードがクリップボードにコピーされます。

   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。

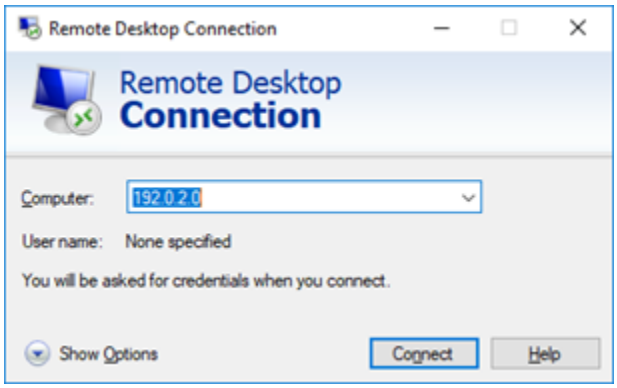
RDC を設定し、Windows インスタンスに接続する

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。

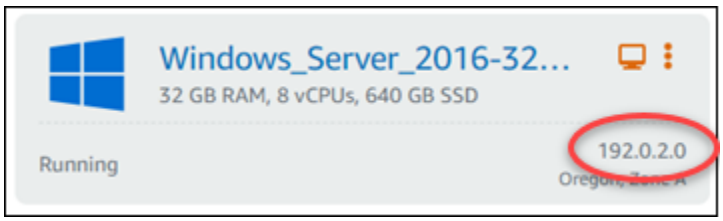
1. Windows メニューを開き、`Remote Desktop Connection` または `RDC` を検索します。
2. 検索結果の [リモートデスクトップ接続] を選択します。



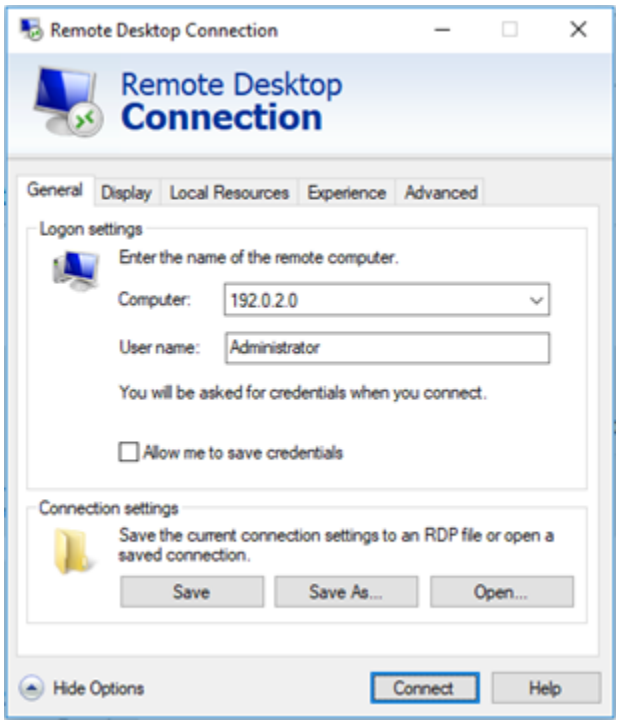
3. [コンピュータ] テキストボックスに、Windows インスタンスのパブリック IP アドレスを入力します。



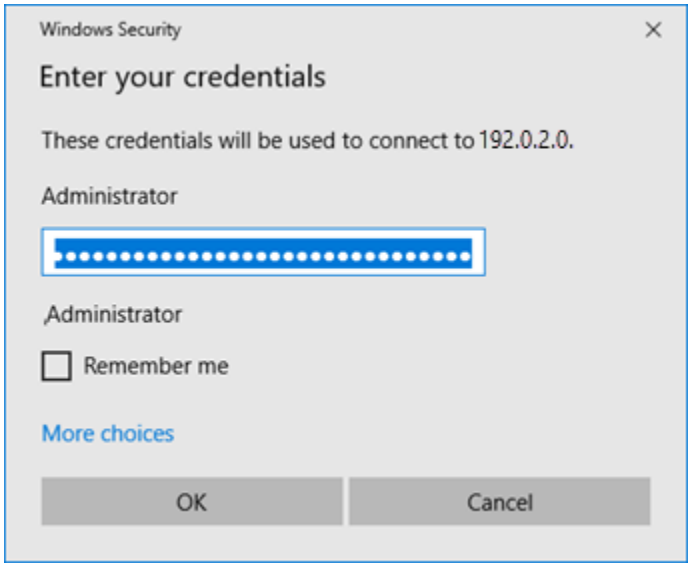
   次の例に示すように、パブリック IP は Lightsail コンソールのインスタンスの横に表示されます。



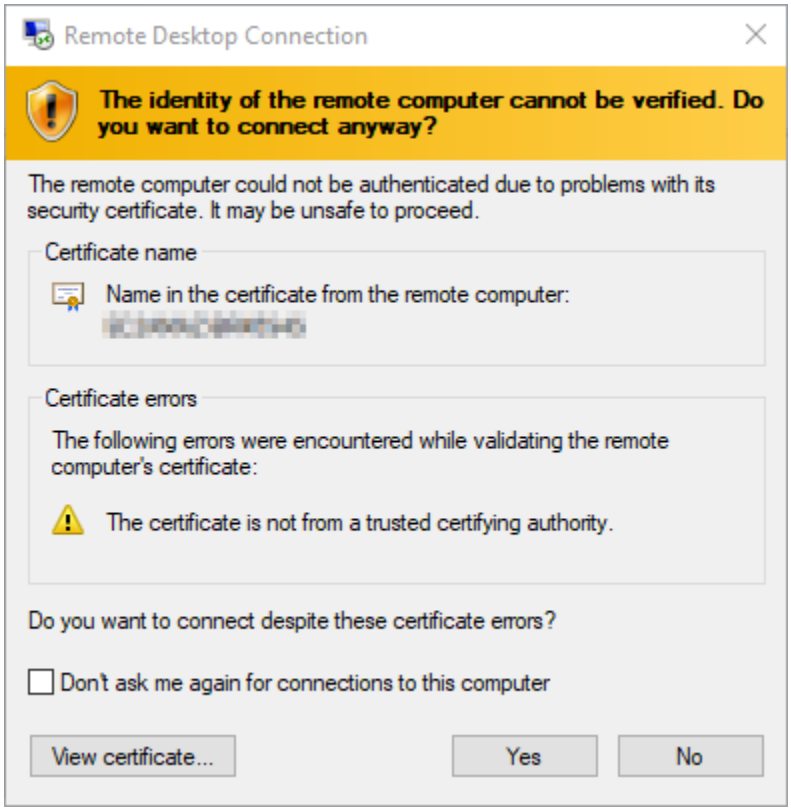
4. [オプションの表示] を選択して追加の接続オプションを表示します。
5. 「ユーザー名」テキストボックスに「」と入力します。これは`Administrator`、Lightsail のすべての Windows インスタンスのデフォルトのユーザー名です。



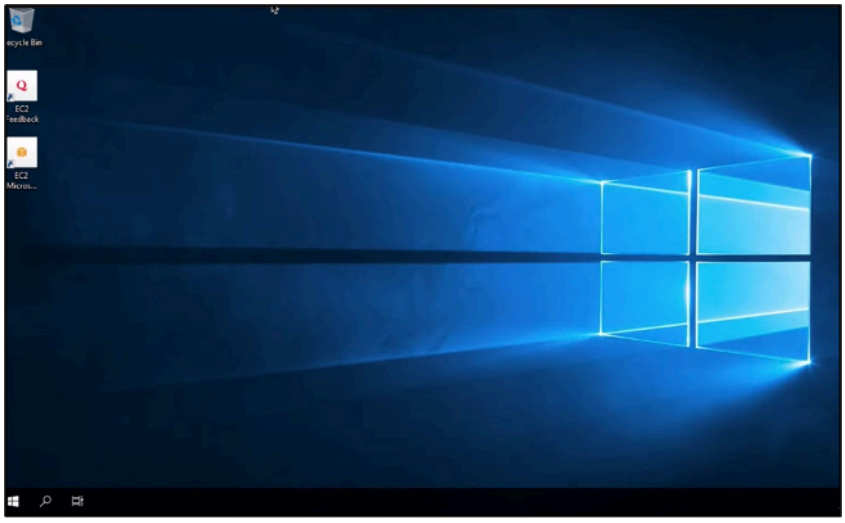
6. [接続]を選択します。
7. 表示されるプロンプトで、この手順の前半で Lightsail コンソールからコピーしたデフォルトの管理者パスワードを入力または貼り付け、OK を選択します。



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。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。

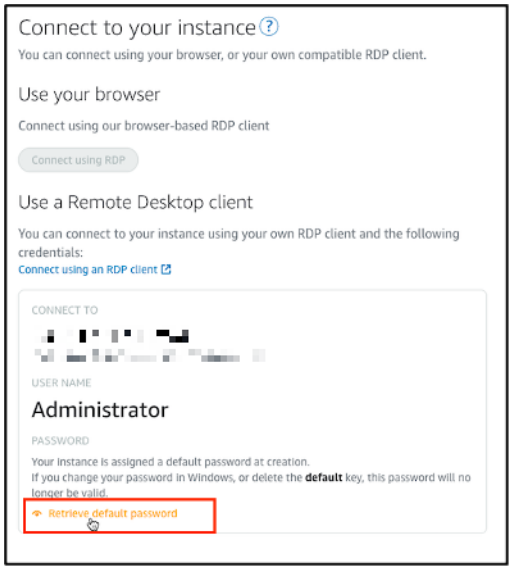
Windows インスタンスについて必要な接続情報を取得する

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。

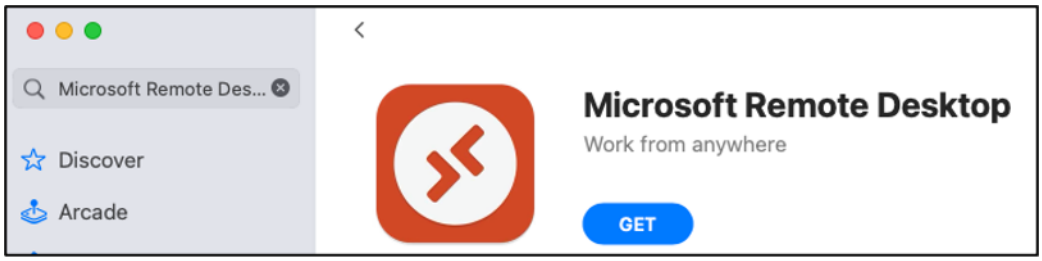
7. 管理者パスワードをコピーします。これは、本ガイドの後半で Microsoft リモートデスクトップクライアントを使用してインスタンスにサインインするために使用します。

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

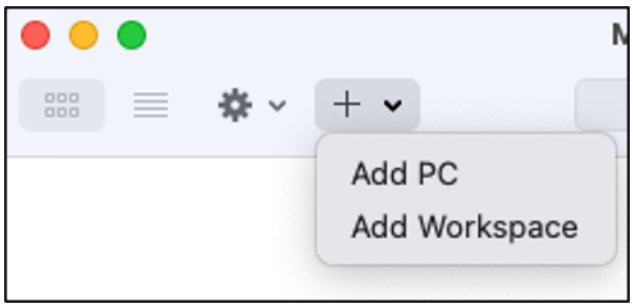
Mac に Microsoft リモートデスクトップクライアントをインストールして、インスタンスに接続するようにクライアントを設定するには、以下の手順を実行します。

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。

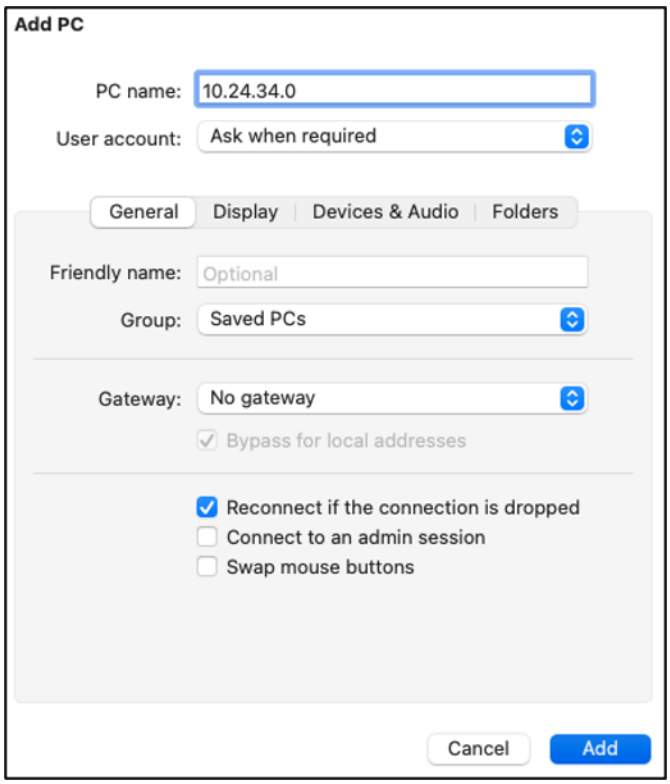
2. 検索結果で [Microsoft Remote Desktop] (Microsoft リモートデスクトップ) を見つけ、[GET] (入手) をクリックしてアプリケーションをインストールします。



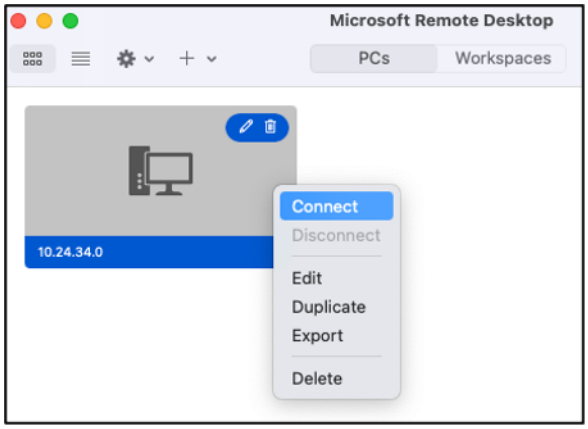
3. インストールが完了したら、[Microsoft Remote Desktop] (Microsoft リモートデスクトップ) を開きます。
4. 上部で [+] アイコンを選択し、[PC の追加] を選択します。



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。

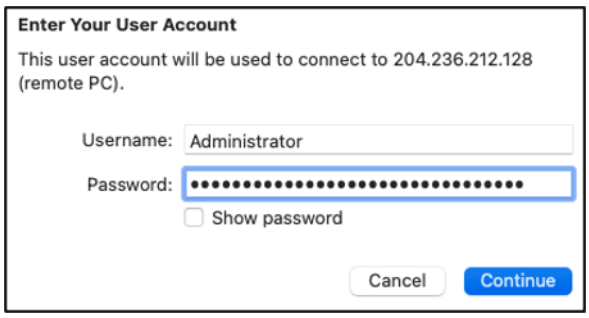


7. インスタンスのアイコンを右クリックし、[Connect] (接続) を選択します。

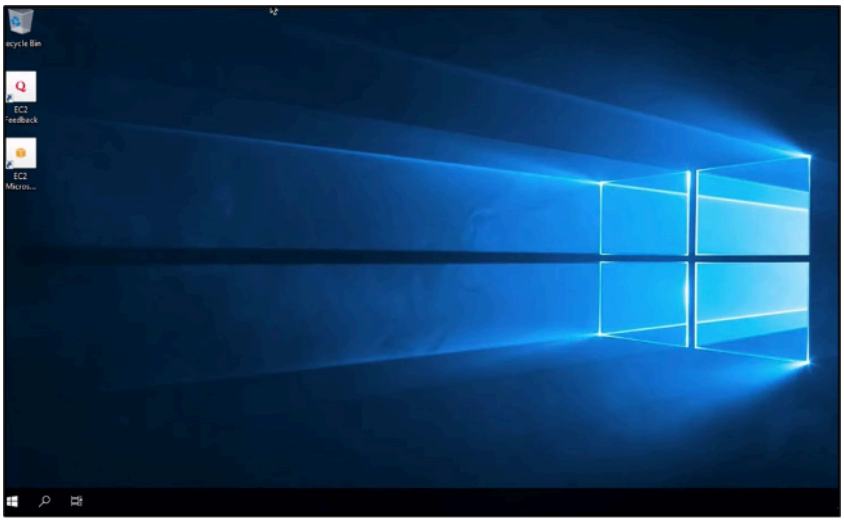


8. [Username]] (ユーザーネーム) テキストボックスに [Administrator (管理者) と入力し、[Password] (パスワード) テキストボックスに本ガイドで先ほど取得したデフォルトの管理者パスワードを入力します。

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。



で Lightsail リソースを管理する AWS CloudShell

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、それぞれに最大 1 GB の永続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

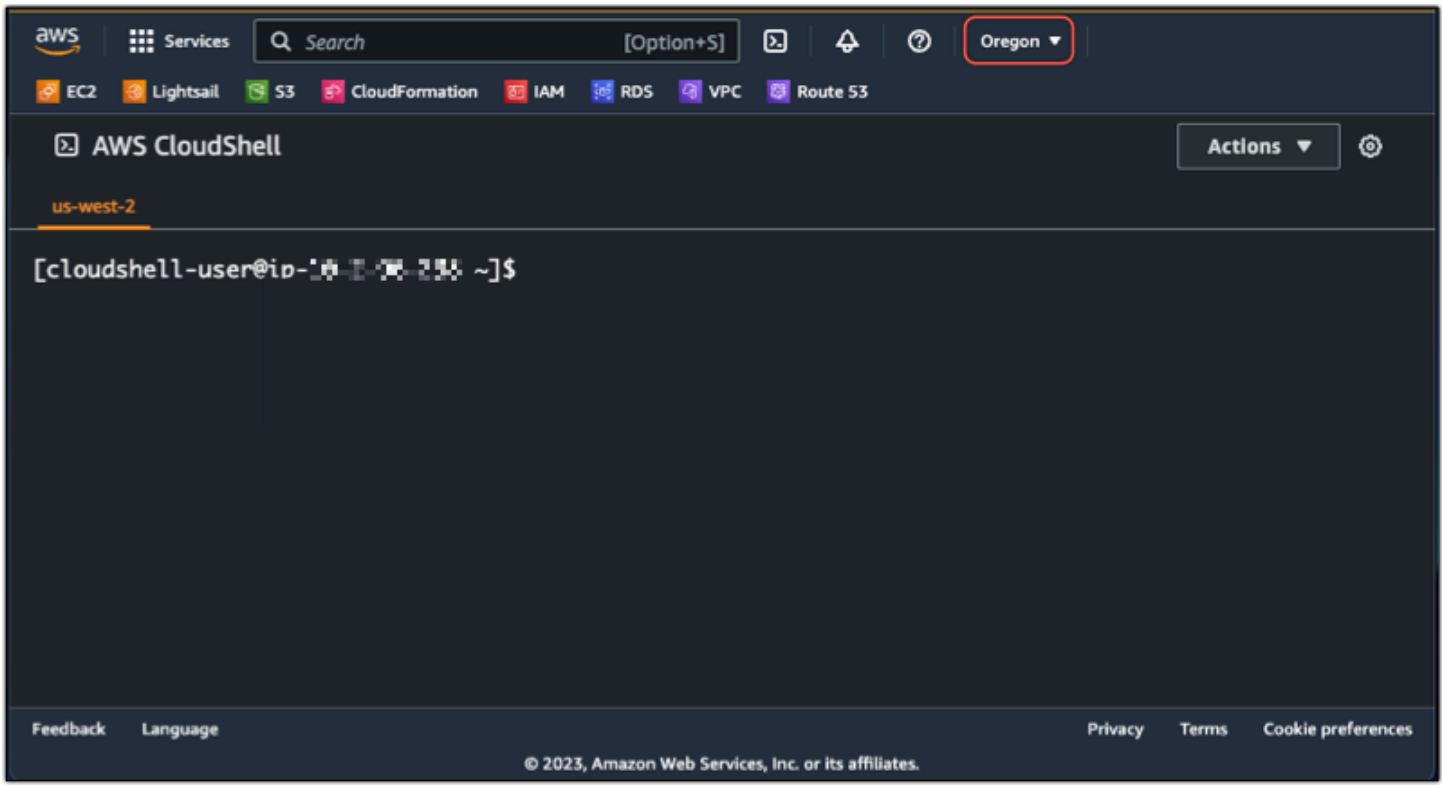
トな記憶域です。各シェルセッションが終了した後に削除されるエフェメラル環境リソースとは異なり、ホームディレクトリ内のデータはセッション間で保持されます。

AWS CloudShell で の使用を停止した場合 AWS リージョン、データは最後のセッションの終了から 120 日間、そのリージョンの永続ストレージに保持されます。アクションを実行しない限り、データは 120 日後にそのリージョンの永続的ストレージから自動的に削除されます。その AWS CloudShell で AWS リージョンを再度起動すれば削除を防止することができます。永続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「永続ストレージ」を参照してください。

AWS リージョン

Lightsail では、 CloudShell セッションが で開き AWS リージョン 、物理的な場所へのレイテンシーが最も少なくなります。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--> にあるかを書き留めます。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

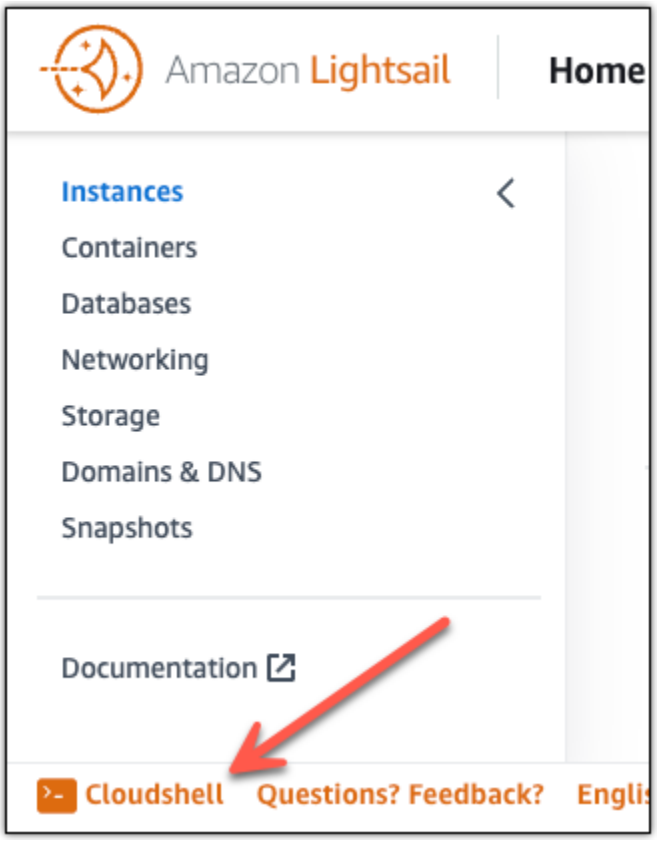
Lightsail 内で AWS CloudShell セッションを起動して使用する方法について説明します。を実行するアクセス許可がない場合は CloudShell、使用している AWS Identity and Access Management (IAM) ID に`arn:aws:iam::aws:policy/AWSCloudShellFullAccess`ポリシーを追加する必要があります。`arn:aws:iam::aws:policy/AdministratorAccess` ポリシーが既にアタッチされている場合は、 にアクセスできます CloudShell。詳細については、「???」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CloudShellの左下にあるコンソールツールバーで 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

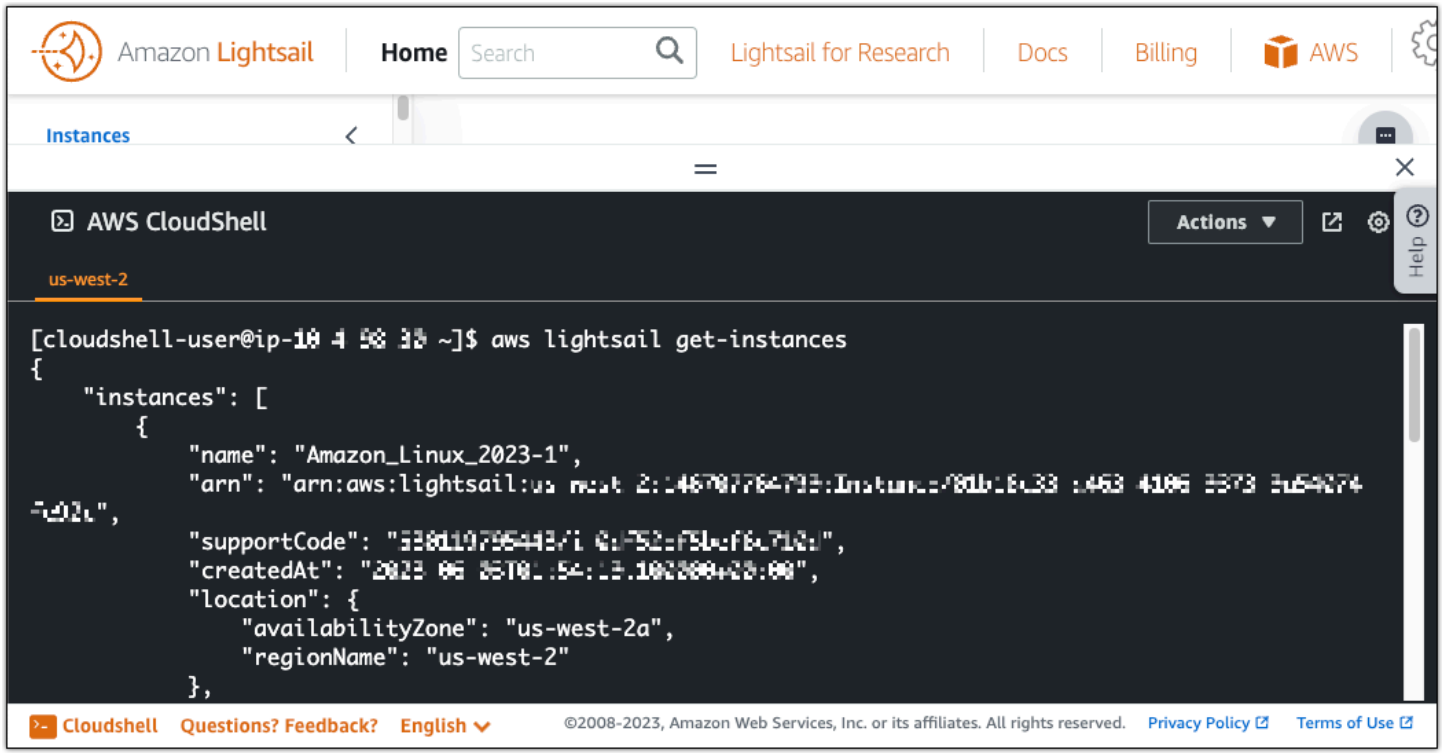
セッションに CloudShellプリインストールされている複数のコマンドラインツールを使用できます。この例では、Lightsail `GetInstances`APIオペレーションを使用して、Lightsail アカウ

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、GetInstancesAmazon Lightsail APIリファレンス」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CloudShellの左下にあるコンソールツールバーで を選択します。
3. AWS CloudShell プロンプトの後に次のコマンドを入力します。

```
aws lightsail get-instances
```

これで、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
- でのよくある質問 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 サービス での の使用 AWS CloudShell

Lightsail でインスタンスメタデータサービス (IMDS) とユーザーデータにアクセスする

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

Lightsail で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するには、次のいずれかの方法を使用します。

- 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 をサポートし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 でのインスタンスメタデータサービス (IMDS) へのアクセスと設定

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 をサポートし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、IMDSv2 がデフォルトで設定されます。

次の例では、Linux と PowerShell シェルスクリプト、IMDSv2 を使用して、最上位のインスタンスメタデータ項目を取得します。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。トークンの使用要件の変更については、「 コマンドリファレンスupdate-instance-metadata-options」の「」を参照してください。 AWS CLI

- 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、 コマンドリファレンスupdate-instance-metadata-optionsの「」を参照してください。 AWS CLI

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： AWS SDKs の最新バージョンと AWS CLI サポート IMDSv2。IMDSv2 を使用するには、インスタンスに最新バージョンの AWS SDKsと があることを確認します AWS CLI。の更新の詳細については AWS CLI、「 ユーザーガイド」の「 のインストール、更新、アンインストール AWS CLI」を参照してください。 AWS Command Line Interface 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」を参照してください。
- Lightsail API オペレーションと AWS CLI コマンドの更新: 既存のインスタンスでは、 update-instance-metadata-options 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。をインスタンスの名前*`InstanceName`*に置き換え、 *`RegionName`* AWS リージョン をインスタンスが存在することを確認してください。

```
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

インスタンスでロール認証情報を使用する AWS SDKs AWS CLI、ソフトウェアを IMDSv2-compatibleバージョンに更新します。の更新の詳細については AWS CLI、「 AWS Command Line Interface ユーザーガイド」の「 の最新バージョンへのアップグレード AWS CLI」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。 IMDSv2

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。この段階では、 update-instance-metadata-options コマンドを使用して IMDSv2 の使用を要求できます。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 を介してのみ使用できます AWS CLI。現在、Lightsail コンソールでは使用できません。詳細については、「」を参照してくださいupdate-instance-metadata-options。

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 ブロックストレージディスクでストレージとパフォーマンスを拡張する

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、数分以内に使用量をスケールアップまたはスケールダウンでき、プロビジョニングした分だけ低価格でお支払いいただけます。

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 の「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」と呼ばれます) は、サーバーにアタッチできる個々のハードディスクのように動作します。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 は、ブロックストレージ用のソリッドステートドライブ (SSD) を提供します。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。Lightsail の追加ブロックストレージディスクは、一貫したパフォーマンスと、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低レイテンシーを提供します。

> **Note**
>
> ディスクあたりの持続的なIOPSパフォーマンスや大量のスループットを必要とするアプリケーションをご利用のお客様、または MongoDB や Cassandra などの大規模なデータベースを実行しているお客様は、Lightsail の代わりに GP2または プロビジョンドIOPSSSDストレージEC2で Amazon を使用することをお勧めします。

Amazon EBSボリュームの詳細については、「Amazon EC2ユーザーガイド」を参照してください。

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 インスタンスにアタッチする

Amazon 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、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。

このトピックでは、新しいディスクを作成し、Lightsail を使用してアタッチする方法について説明します。また、 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。これによりSSH、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 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 と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。

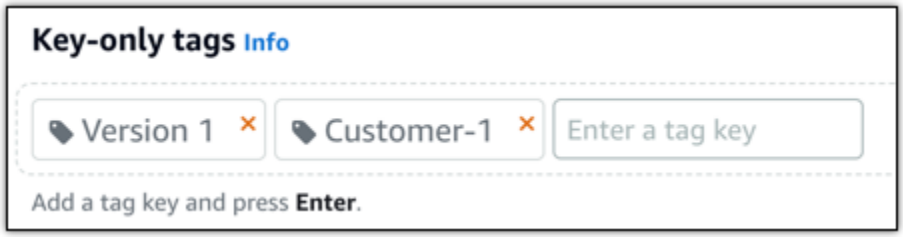
   リソース名:

   - 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。

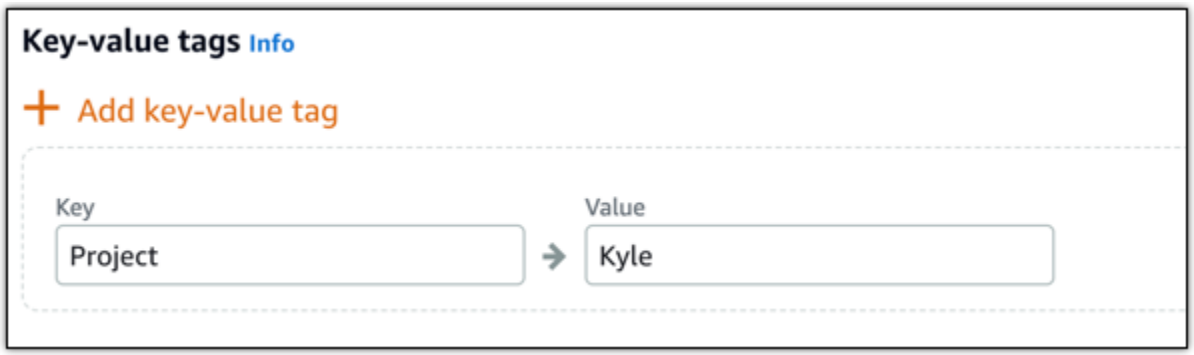
6. 以下のいずれかのオプションを選択して、ディスクにタグを追加します。

   - [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。

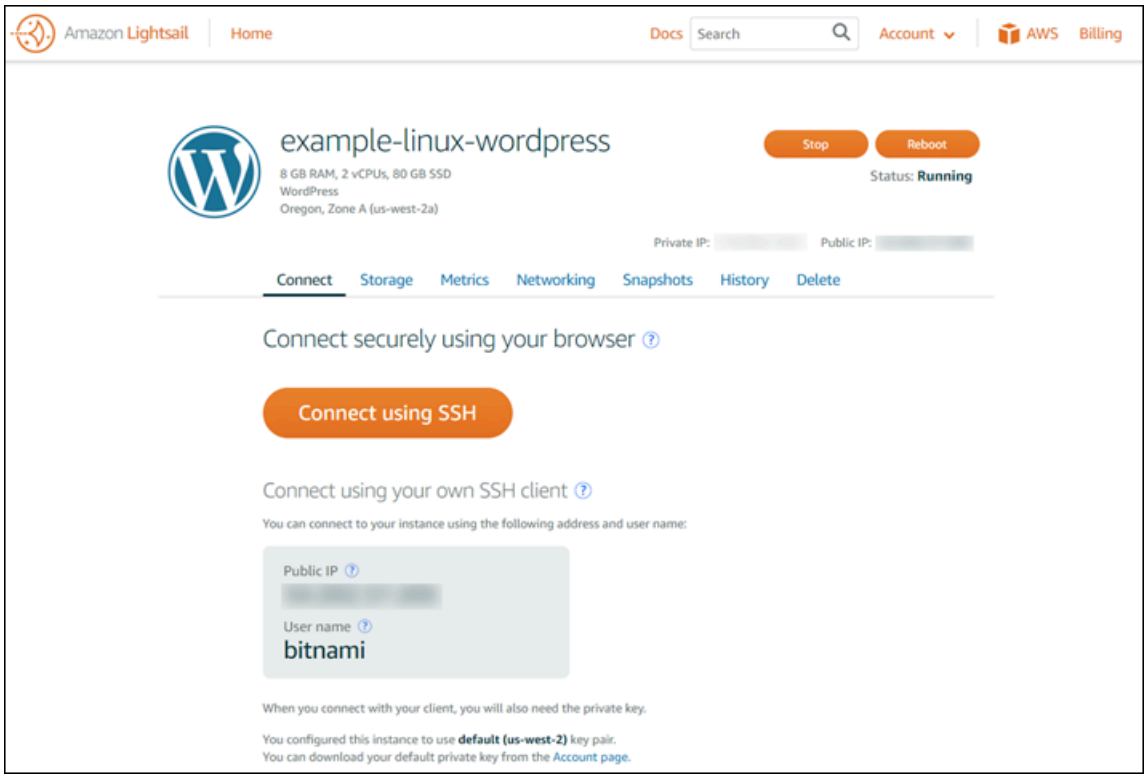
   数秒後、ディスクが作成され、新しいディスク管理ページが表示されます。

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 の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. を使用して接続SSHを選択し、インスタンスに接続します。
3. ターミナルウィンドウに次のコマンドを入力します。

```
lsblk
```

   の出力では、ディスクパスから/dev/プレフィックスが`lsblk`省略されます。

> **Note**
>
> 2023 年 6 月 29 日に、Lightsail インスタンスの基盤となるハードウェアを更新しました。次の例では、旧世代のインスタンスのデバイス名は として表示されます/dev/

xvda。この日付以降に作成されたインスタンスのデバイス名は、 として表示されます/dev/nvme0n1。

Current generation instances

次の出力例では、ルートボリューム (nvme0n1) には 2 つのパーティション (nvme0n1p1 および nvme0n1p128) がありますが、追加のボリューム (nvme1n1) にはパーティションがありません。

```
[ec2-user ~]$ sudo lsblk
NAME          MAJ:MIN RM SIZE RO TYPE MOUNTPOINT
nvme1n1       259:0    0  30G  0 disk /data
nvme0n1       259:1    0  16G  0 disk
##nvme0n1p1   259:2    0   8G  0 part /
##nvme0n1p128 259:3    0   1M  0 part
```

Previous generation instances

次の出力例では、ルートボリューム (xvda) にはパーティション (xvda1) がありますが、追加のボリューム (xvdf) にはパーティションがありません。

```
[ec2-user ~]$ sudo lsblk
NAME    MAJ:MIN RM SIZE RO TYPE MOUNTPOINT
xvda    202:0    0  16G  0 disk
##xvda1 202:1    0   8G  0 part /
xvdf    202:80   0  24G  0 disk
```

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。

   以下を使用して、ディスクにファイルシステムがあるかどうかを判断します。ディスクにファイルシステムがない場合は、ステップ 2.5 に進みます。ディスクにファイルシステムがある場合は、ステップ 2.6 に進みます。

Current generation instances

```
sudo file -s /dev/nvme1n1
```

新しいディスクでは、次のような出力が表示されます。

```
/dev/nvme1n1: 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/nvme1n1: SGI XFS filesystem data (blksz 4096, inosz 512, v2 dirs)
```

Previous generation instances

```
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. ディスクに新しいファイルシステムを作成するには、次のコマンドを使用します。デバイス名 (など/dev/nvme1n1) を に置き換えます。*device_name*。 アプリケーションの要件やオペレーティングシステムの制限に応じて、 ext3 や などの別のファイルシステムタイプを選択できますext4。

> **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、ステップ 2.6 に進

み、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。

Current generation instances

```
sudo mkfs -t xfs device_name
```

次のような出力が表示されます。

```
meta-data=/dev/nvme1n1           isize=512    agcount=16, agsize=1048576 blks
         =                       sectsz=512   attr=2, projid32bit=1
         =                       crc=1        finobt=1, sparse=1, rmapbt=0
         =                       reflink=1    bigtime=1 inobtcount=1
data     =                       bsize=4096   blocks=16777216, imaxpct=25
         =                       sunit=1      swidth=1 blks
naming   =version 2              bsize=4096   ascii-ci=0, ftype=1
log      =internal log           bsize=4096   blocks=16384, version=2
         =                       sectsz=512   sunit=1 blks, lazy-count=1
realtime =none                   extsz=4096   blocks=0, rtextents=0
```

Previous generation instances

```
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。の場所を に置き換える *mount_point*など、未使用のスペースの場合は /data。

```
sudo mkdir mount_point
```

7. 次のコマンドを入力して、ディスクにファイルシステムがあることを確認できます。

   Current generation instances

   ```
   sudo file -s /dev/nvme1n1
   ```

   /dev/nvme1n1: 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/nvme1n1: SGI XFS filesystem data (blksz 4096, inosz 512, v2 dirs)
   ```

   Previous generation instances

   ```
   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。ファイルアクセス許可の詳細については、「Amazon 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EC2」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

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たい場合があります。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。

   変更を保存できるように、 ファイルを開くsudo前に を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。

   Current generation instances

   ```
   /dev/nvme1n1 /data xfs defaults,nofail 0 2
   ```

   Previous generation instances

   ```
   /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

追加のストレージ領域が必要な場合は、ブロックストレージディスクを作成して Amazon Lightsail の 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を説明します。また、ディスクをオンラインにして初期化RDPできるように、 を使用して Windows Server インスタンスに接続する方法についても説明します。

> **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 と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。

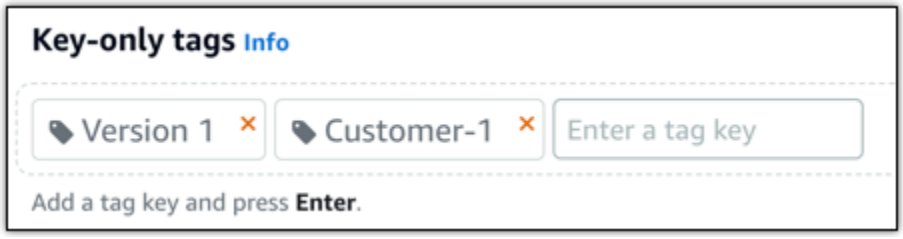
   リソース名:

   - 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。

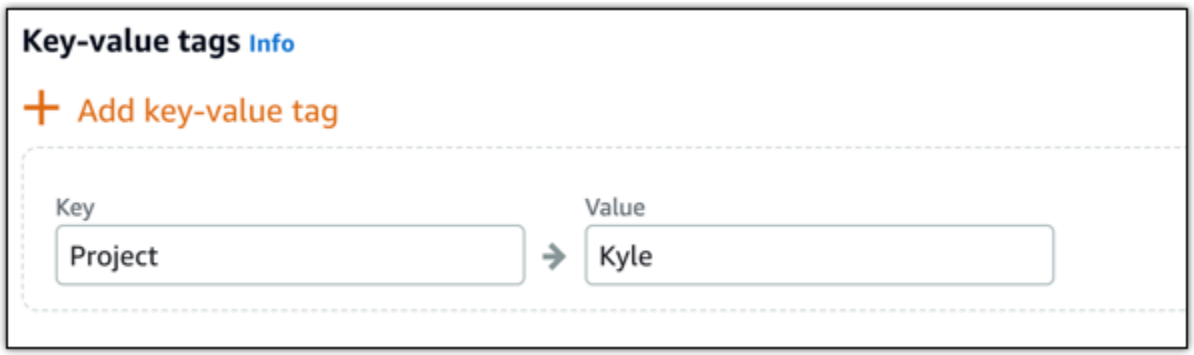
6. 以下のいずれかのオプションを選択して、ディスクにタグを追加します。

   - [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。

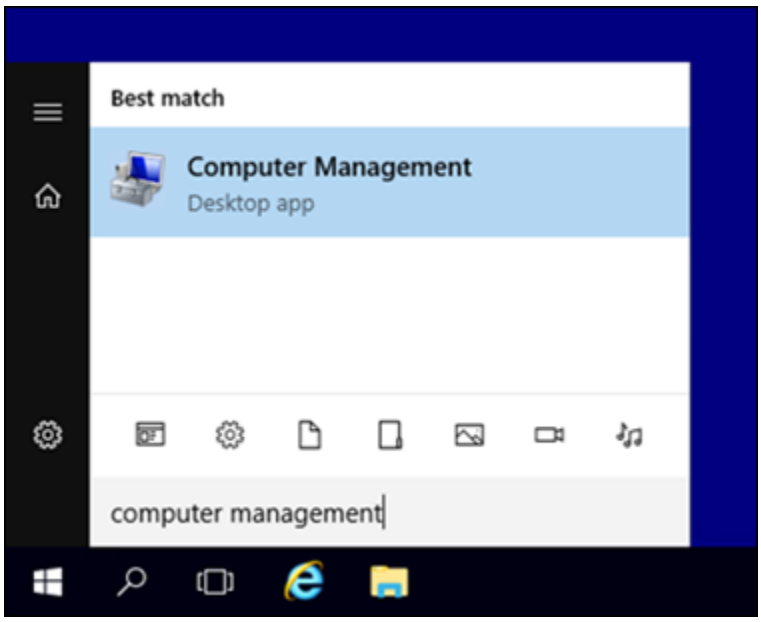
ブロックストレージディスクをオンラインにするには、このガイドの「ステップ 2: インスタンスに接続し、ブロックストレージディスクをオンラインにする」セクションに進みます。

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

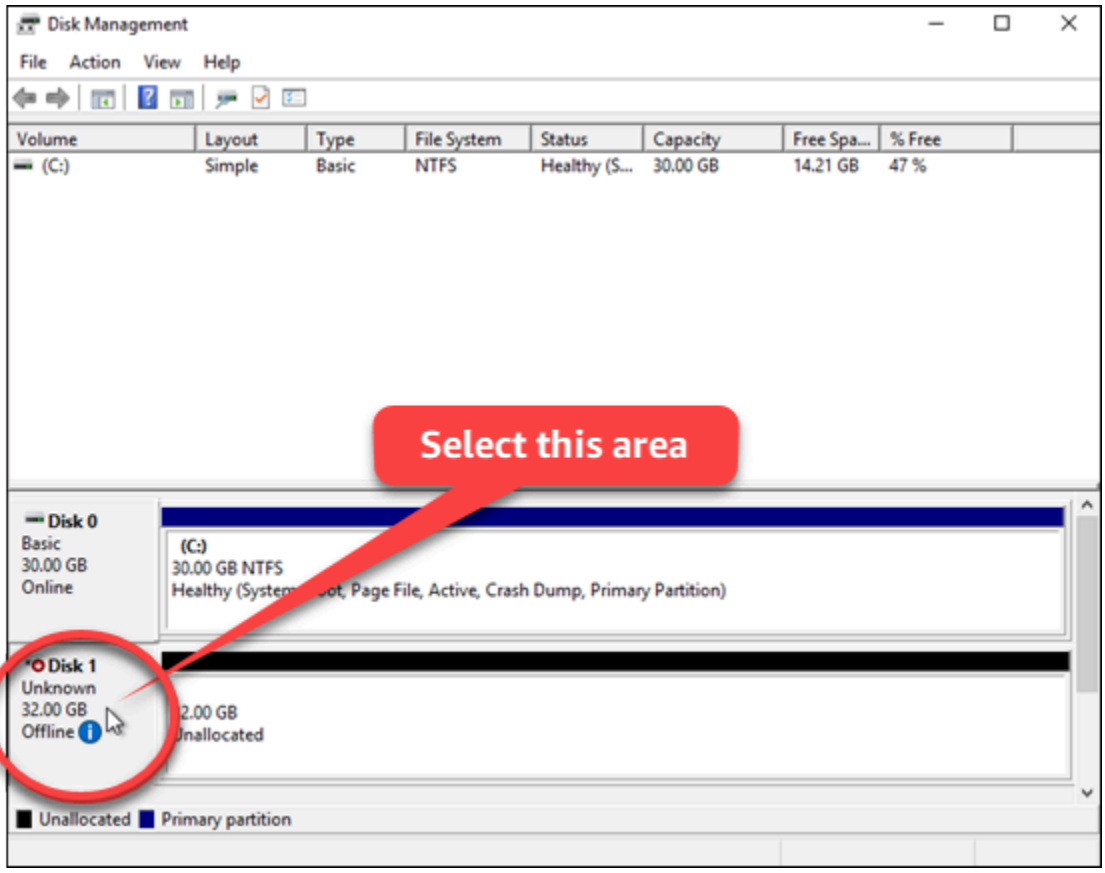
Windows Server インスタンスに接続し、ディスクの管理ユーティリティを使用して、前にアタッチしたブロックストレージディスクをオンラインにします。

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

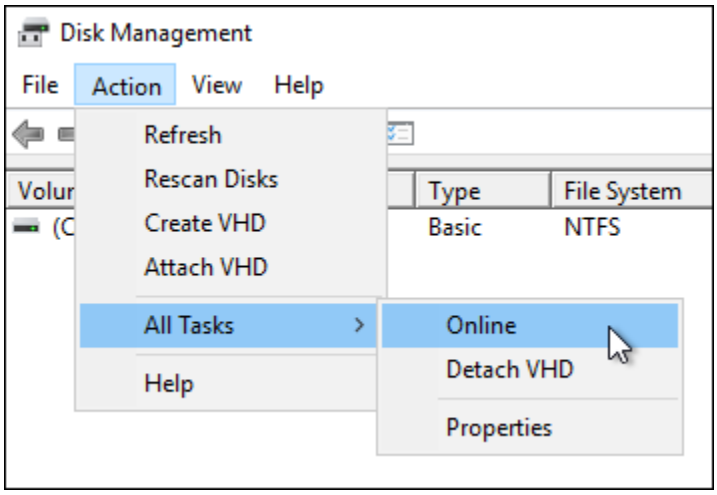
1. Lightsail コンソールのホームページに移動します。
2. このガイドで前に追加のストレージディスクをアタッチしたインスタンスの名前を選択します。
3. Connect タブで、 を使用して Connect RDPを選択します。
4. Windows のスタートメニューで、コンピューターの管理を検索し、検索結果から [コンピューターの管理] を選択します。



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。

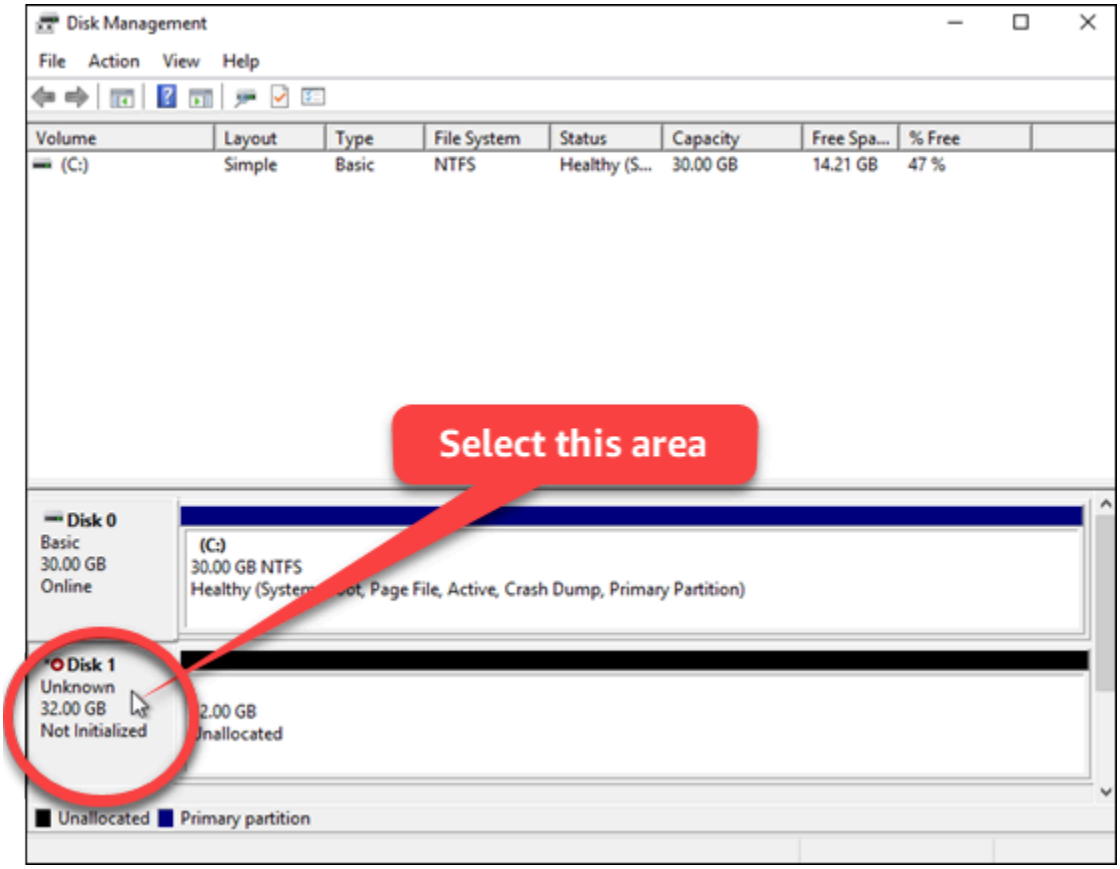
ステップ 3: ブロックストレージディスクを初期化する

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。

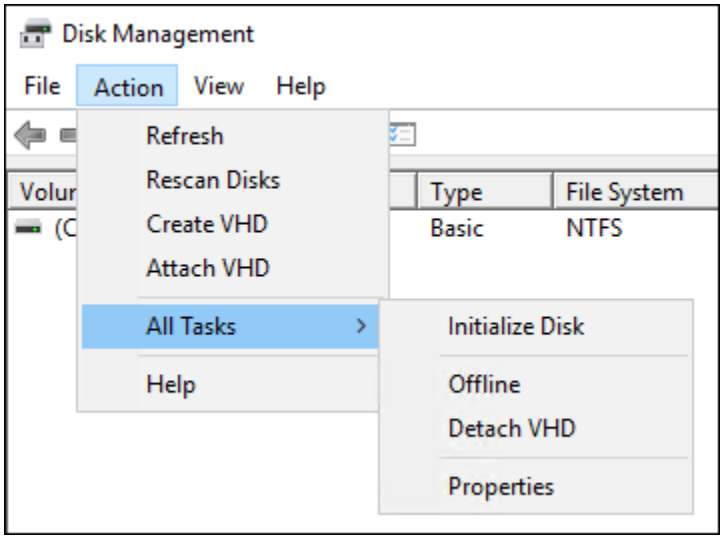
> **Important**
>
> すでにデータが含まれているディスク (スナップショットから作成したディスクなど) をマウントする場合は、ディスクを再フォーマットしないように注意してください。再フォーマットすると、既存のデータが削除されます。

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「パーティションスタイルについて - GPTおよび MBR記事」を参照してください。

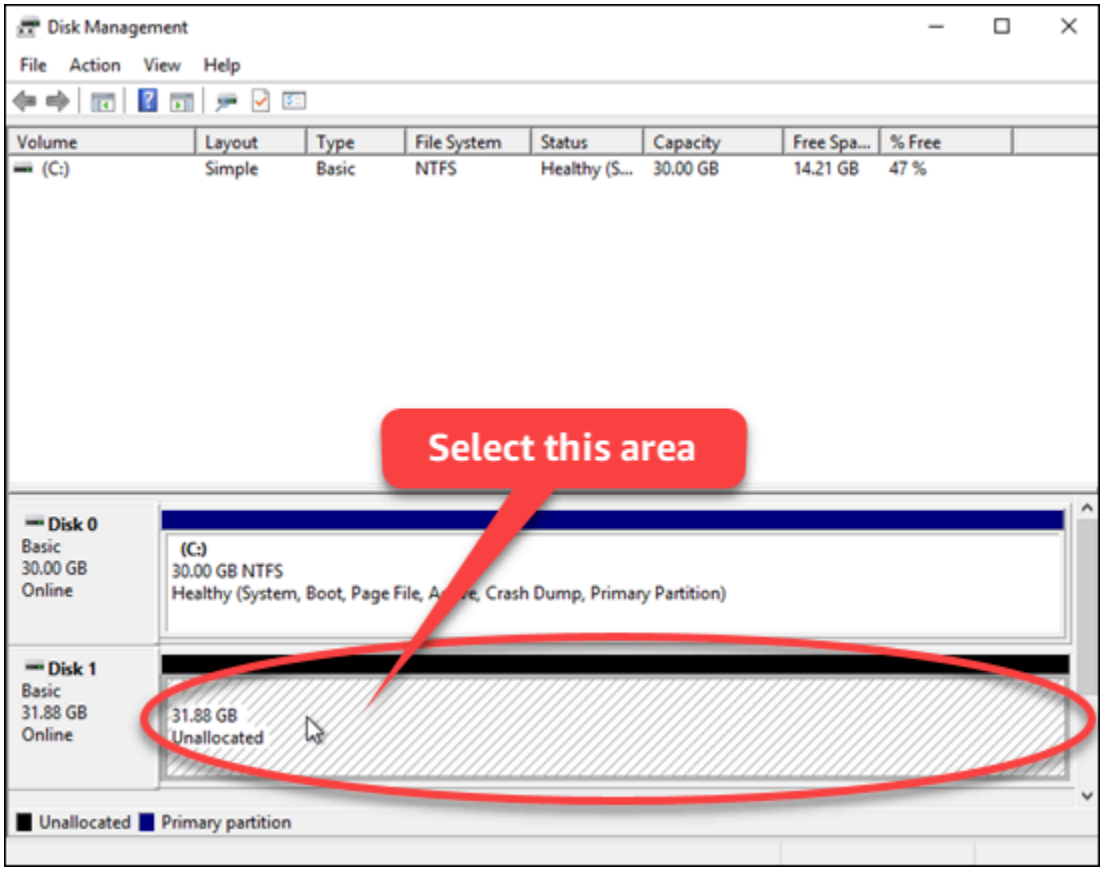
ブロックストレージディスクのステータスが [オンライン] に更新されるのがわかります。ファイルシステムでブロックストレージディスクをフォーマットにするには、このガイドの「ステップ 4: ディスクをファイルシステムでフォーマットする」セクションに進みます。

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

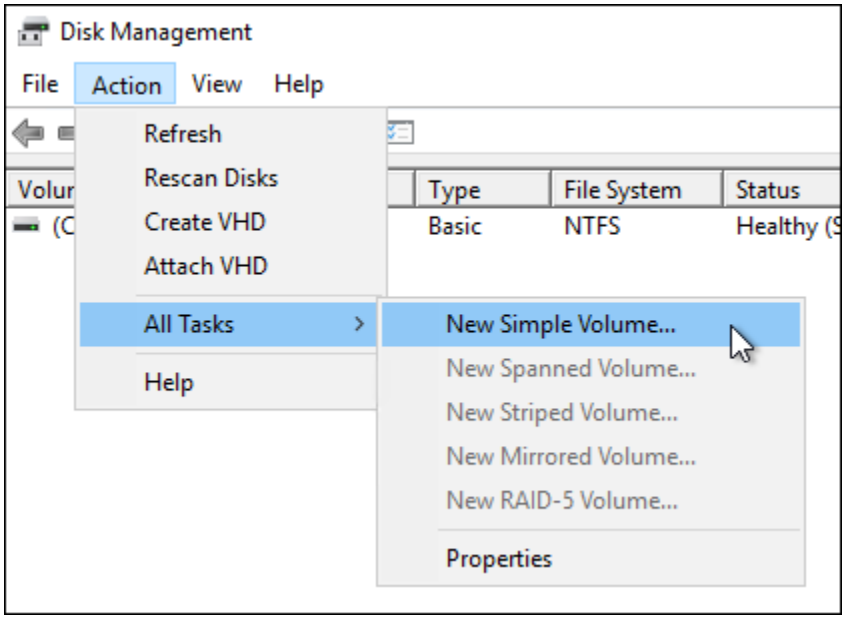
Windows Server の新しいシンプルボリュームウィザードを使用して、ドライブ文字を割り当て、ディスクをファイルシステムでフォーマットします。

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。



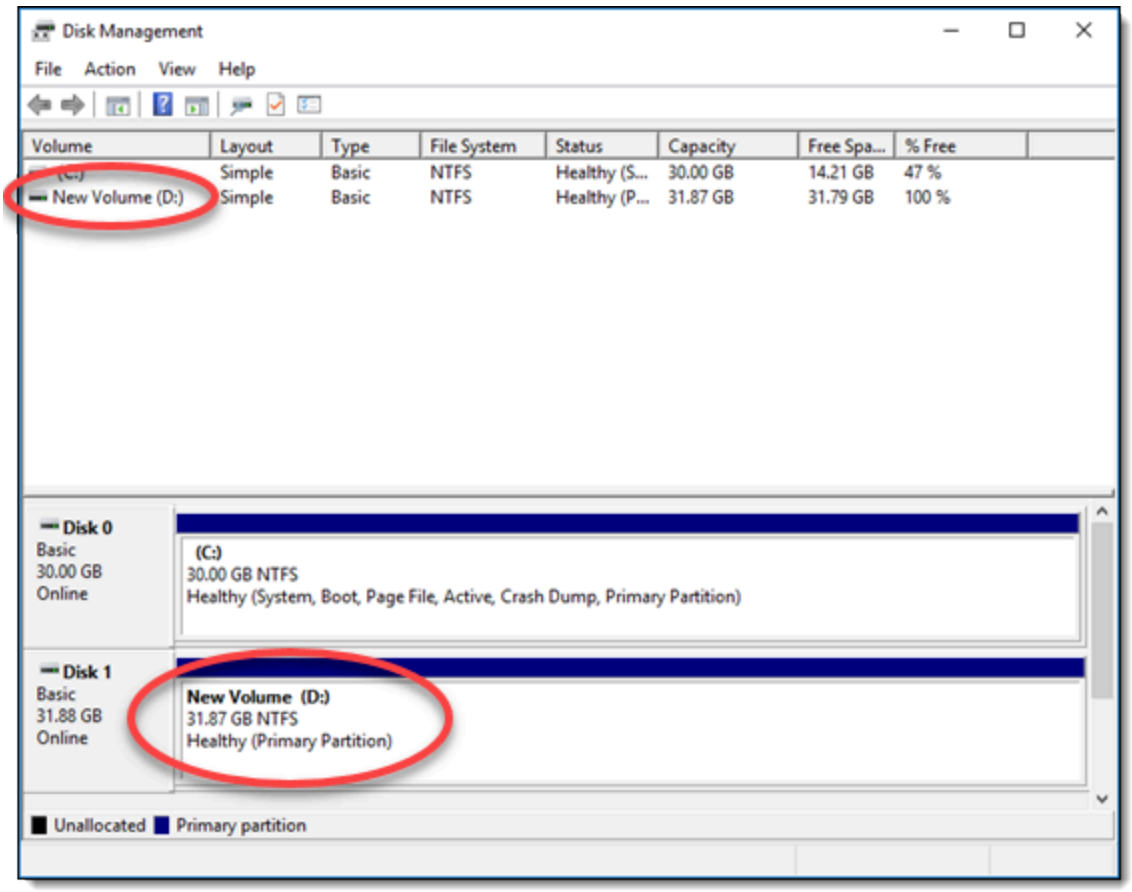
2. パーティションを選択した状態で、[アクション] メニューの [すべてのタスク] をポイントし、[新しいシンプルボリューム] を選択します。



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、、、または ReFS ファイルシステムタイプを選択しFAT32、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 のファイルシステム記事NTFSの概要、障害耐性ファイルシステム (ReFS) の概要、および説明を参照してください。 FAT32

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のデタッチと削除

ブロックストレージディスクが不要になった場合は、停止した Amazon Lightsail インスタンスからデタッチしてから削除できます。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。

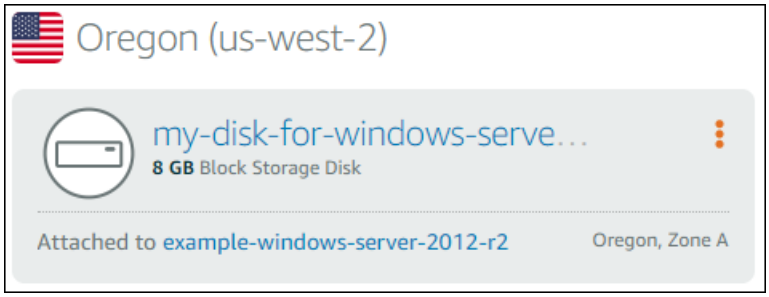
前提条件

- 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すると、ディスクを安全にデタッチおよび削除できます。

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、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、スナップショットのストレージ料金が請求されます。データ破損やディスク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、Lightsail がユーザーに代わってインスタンスの日次バックアップを作成するか、手動で操作することなくストレージディスクをブロックします。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、またはより大きな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、より小さな Lightsail プランを使用して新しいリソースを作成するためには使用できません。

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、ある Amazon Web Services (AWS) リージョンから同じ Lightsail アカウント内の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、「ある 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。

スナップショットを Amazon にエクスポートする EC2

Lightsail は、 の使用を開始する最も簡単な方法です AWS。ただし、Amazon EC2や他の AWS サービスには存在しない Lightsail には制限があります。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 にエクスポートEC2して、利用可能な幅広いインスタンスタイプを活用し、 の サービスの全範囲を使用します AWS。詳細については、「Amazon へのスナップショットのエクスポートEC2」を参照してください。

> **Note**
>
> cPanel & WHM (CentOS 7) インスタンスのスナップショットを Amazon にエクスポートすることはできませんEC2。

スナップショットを削除する

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、不要になった Lightsail 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。

Lightsail インスタンスとディスクの自動スナップショットを設定する

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にすると、Amazon Lightsail は、デフォルトの自動スナップショット時間中、または を指定している間に、リソースのスナップショットを毎日作成します。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のスナップショットストレージ料金が請求されます。

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 SDKs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では、デフォルトで point-in-time バックアップ機能も有効になっています。この機能を使用して、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、 point-in-time 「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 アカウントに保持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。

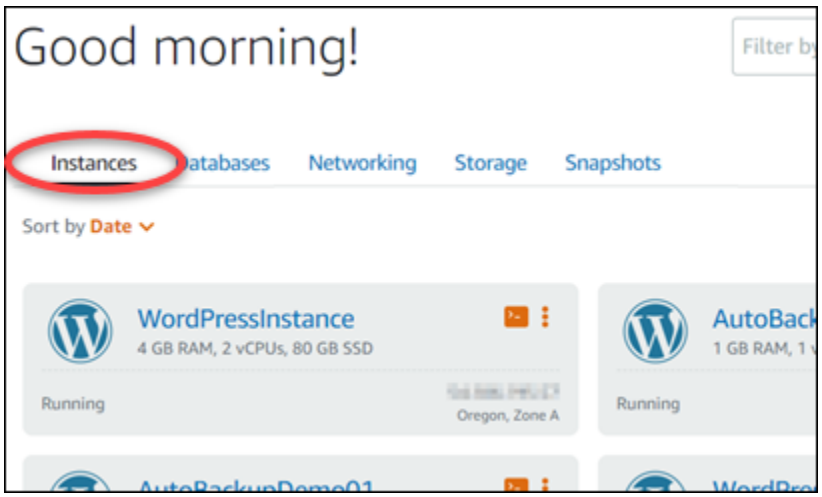
リソースの自動スナップショット機能を無効化した時、リソースの既存の自動スナップショットは、以下のいずれかの操作を行うまで、ソースリソースとともに保持されます。

- 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。

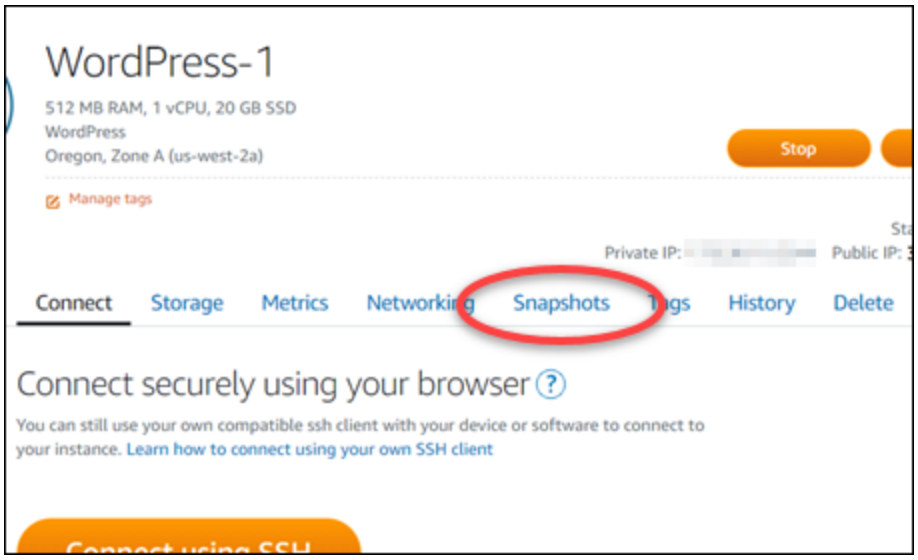
Lightsail コンソールを使用して自動インスタンススナップショットを有効または無効にする

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、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 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の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。

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 AWS CLI

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、 をインストール AWS CLIし、Lightsail と連携するように設定してください。

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 はリソースにデフォルトのスナップショット時間を割り当てます。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。

- リソース AWS リージョン が配置されている がある*#####*。
- *ResourceName* リソースの名前。

- *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。

- インスタンスを作成する AWS リージョン がある*#####*。
- *AvailabilityZone* インスタンスを作成するアベイラビリティーゾーンを持つ 。
- *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* インスタンスに使用する名前の 。
- *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。

- AWS リージョン ディスクを作成する がある*#####*。
- *AvailabilityZone* ディスクを作成するアベイラビリティーゾーンを持つ 。
- *Size* は、ディスクの希望サイズ (GB 単位) に置き換えます。
- *DiskName* ディスクに使用する名前の 。
- *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。

- リソース AWS リージョン が配置されている がある*#####*。
- *ResourceName* リソースの名前。

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の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**
>
> これらのコマンドの EnableAddOn および 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DisableAddOnのEnableAddOn「」および「」を参照してください。

Lightsail インスタンスとディスクの自動スナップショットスケジュールを調整する

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にすると、Lightsail はデフォルトの自動スナップショット時間 または指定した時間にリソースのスナップショットを毎日作成します。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 SDKs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、指定した時点から最大 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が満たされると、Lightsail コンソールにメッセージが表示され、新しいスナップショットが有効になるまでに最大 24 時間かかる可能性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、Lightsail は次のいずれかのデフォルトの自動スナップショット時間を割り当てます。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 AWS リージョン が配置されている によって異なります。

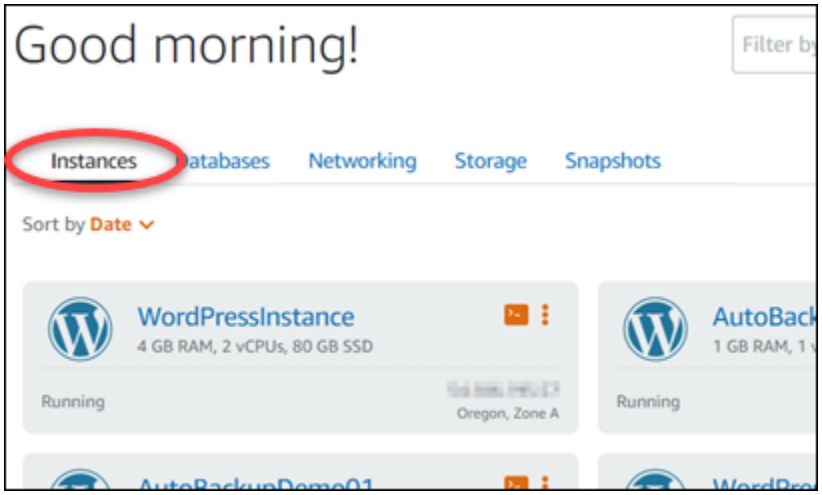
- 米国東部 (オハイオ) (us-east-2): 03:00 UTC
- 米国東部 (バージニア北部) (us-east-1): 06:00 UTC
- 米国西部 (オレゴン) (us-west-2): 06:00 UTC
- アジアパシフィック (ムンバイ) (ap-south-1): 17:00 UTC
- アジアパシフィック (ソウル) (ap-northeast-2): 13:00 UTC
- アジアパシフィック (シンガポール) (ap-southeast-1): 14:00 UTC
- アジアパシフィック (シドニー) (ap-southeast-2)): 12:00 UTC
- 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

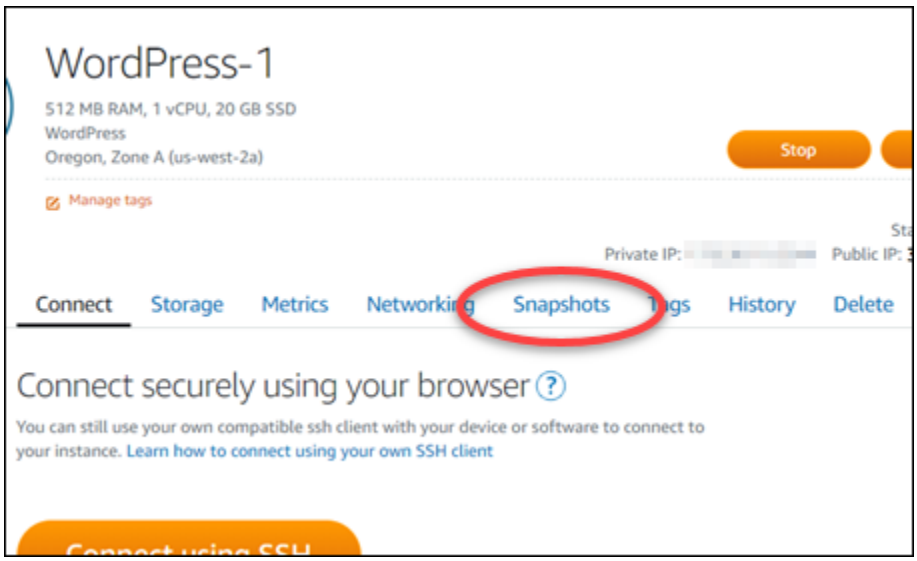
Lightsail コンソールを使用して自動スナップショット時間を変更する

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、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 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 で自動スナップショットを作成す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。

を使用して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時間を変更する AWS CLI

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、 をインストール AWS CLIし、Lightsail と連携するように設定してください。
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。

   - リソース AWS リージョン が配置されている がある*#####*。
   - *ResourceName* リソースの名前。
   - *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**
>
> このコマンドの 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EnableAddOnの「」を参照してください。

未使用の Lightsail インスタンスとディスクスナップショットを削除する

Amazon Lightsail の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、この機能が有効かどうか、または有効になった後に無効になっているかどうかなど、いつでも削除できます。Lightsail アカウントに保存されている自動スナップショットの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

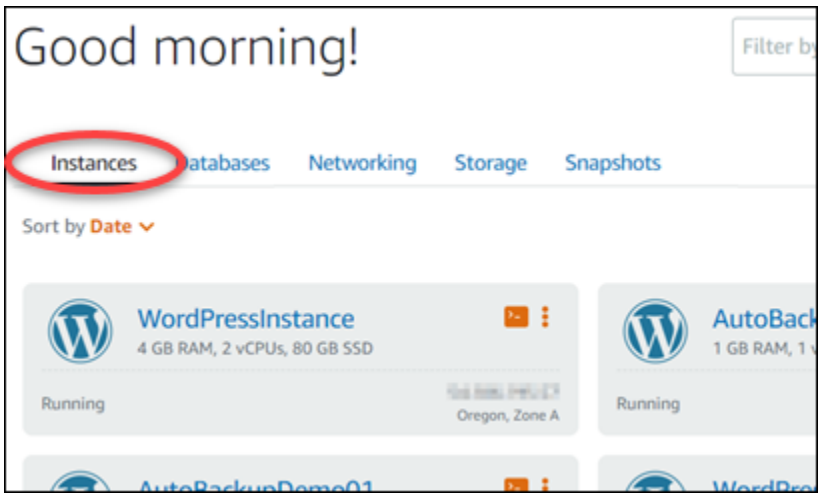
自動スナップショットの削除に関する制限

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 SDKs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。

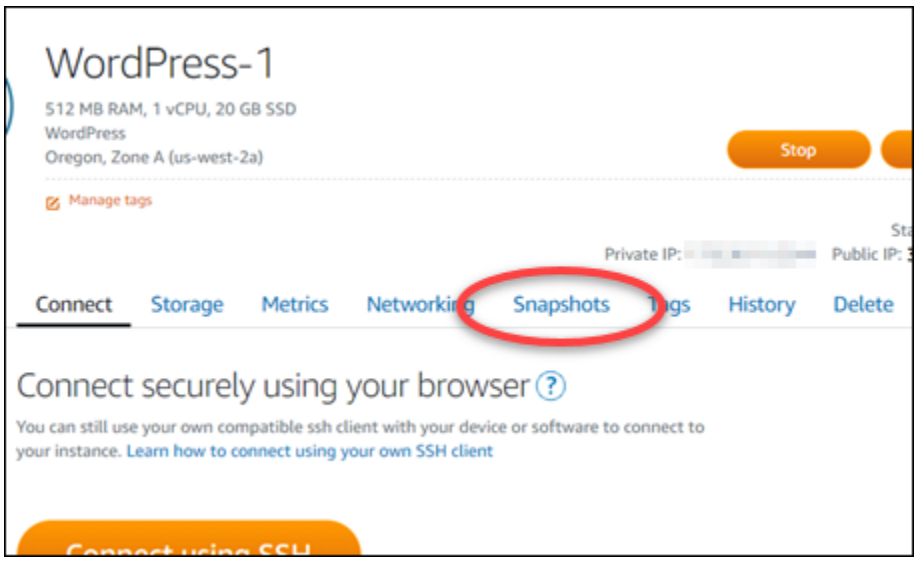
Lightsail コンソールを使用してインスタンスの自動スナップショットを削除する

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、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 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。

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

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、 をインストール AWS CLIし、Lightsail で動作するように設定してください。
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。

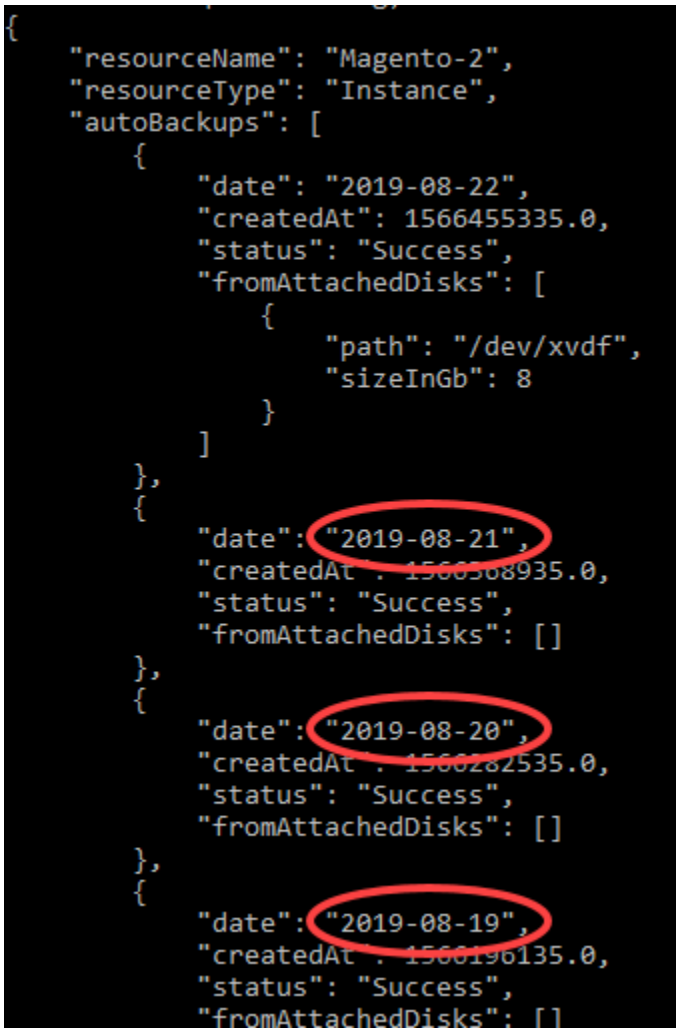
   - リソース AWS リージョン が配置されている がある#####。

- *ResourceName* リソースの名前。

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。

- リソース AWS リージョン が配置されている がある*#####*。
- *ResourceName* リソースの名前。
- *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 GetAutoSnapshots 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DeleteAutoSnapshotのGetAutoSnapshots「」および「」を参照してください。

Lightsail で自動スナップショットが置き換えられないようにする

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機能を有効にすると、リソースの最新 7 日の自動スナップショットのみが保存されます。 Amazon Lightsail

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のスナップショットストレージ料金が請求されます。

> **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

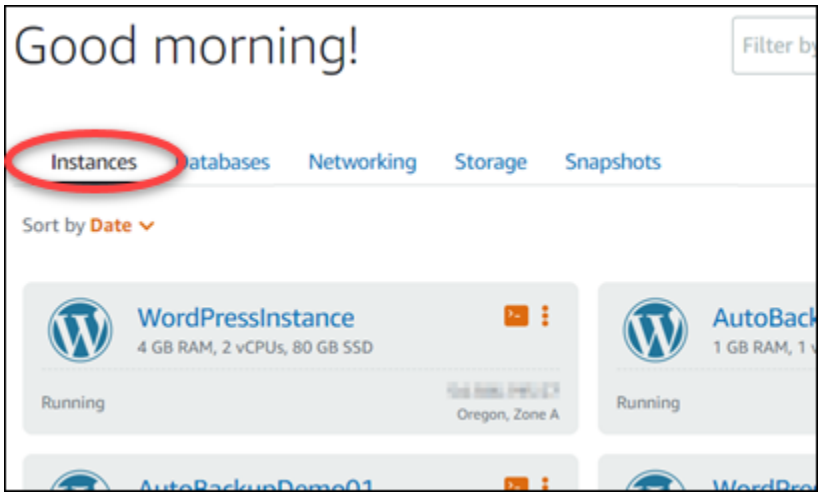
ブロックストレージディスクの自動スナップショットは、Lightsail コンソールを使用して手動スナップショットにコピーすることはできません。ブロックストレージディスクの自動スナップショットをコピーするには、Lightsail API、 AWS Command Line Interface （ AWS CLI ）、または SDKsを使用する必要があります。詳細については、「AWS CLIを使用したインスタンスおよびブロックストレージディスクの自動スナップショットの保持」を参照してください。

Lightsail コンソールを使用してインスタンスの自動スナップショットを保持する

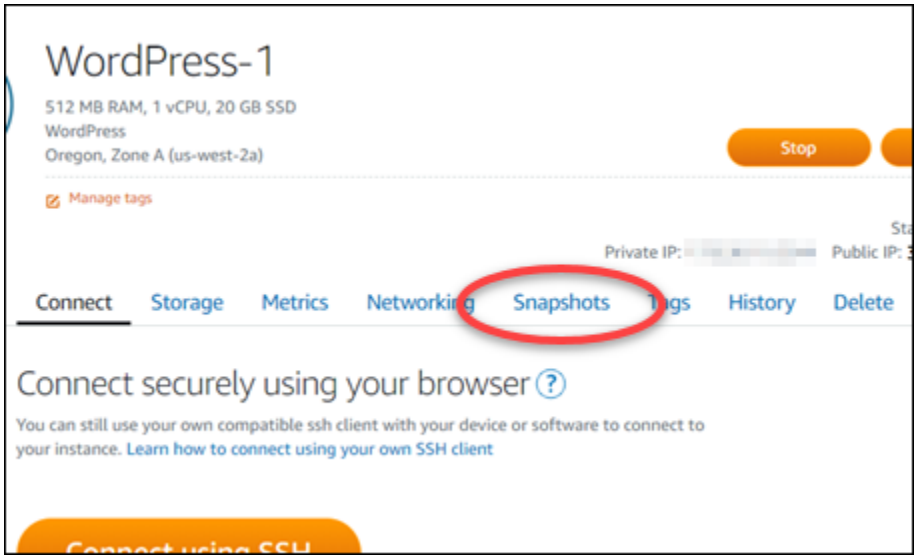
Lightsail コンソールを使用してインスタンスの自動スナップショットを保持するには、次のステップを実行します。

1. Lightsail コンソール にサインインします。

2. Lightsail 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。それ以外の場合は、自動スナップショットと Lightsail アカウントに保存されている重複した手動スナップショットのスナップショットストレージ料金が請求されます。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。

を使用して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を保持する AWS CLI

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、 をインストール AWS CLIし、Lightsail と連携するように設定してください。

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
   ```

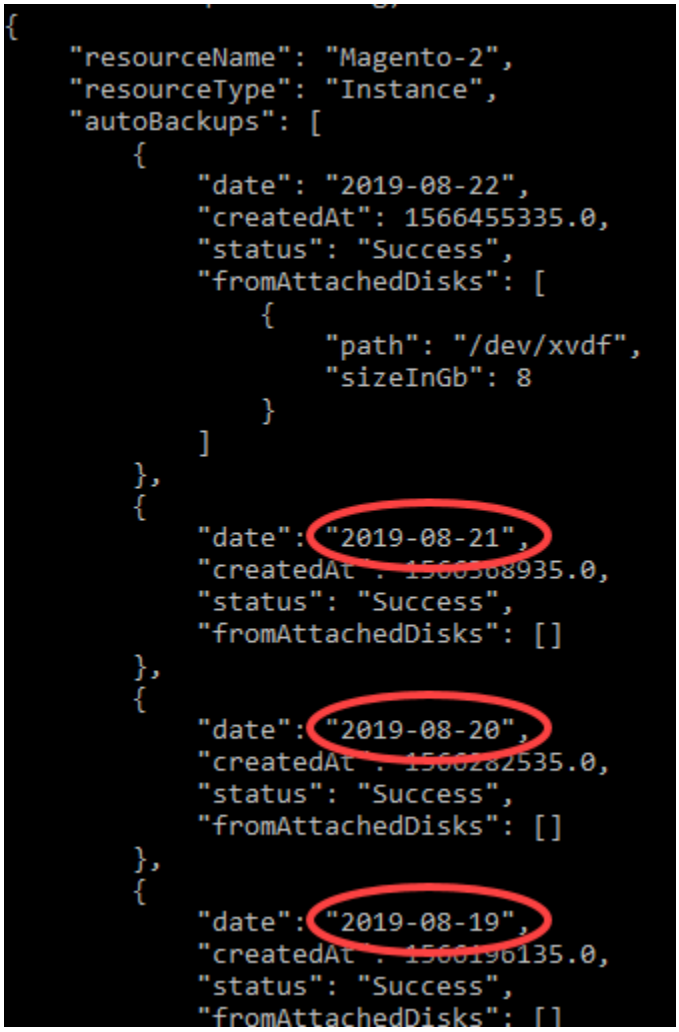
   コマンドを、以下のように置き換えます。

   - リソース AWS リージョン が配置されている がある*#####*。
   - *ResourceName* リソースの名前。

   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* スナップショットをコピーする AWS リージョン を使用します。
- *ResourceName* リソースの名前。
- *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* 自動スナップショットが現在 AWS リージョン 存在する を持つ 。
- *SnapshotName* 作成する新しいスナップショットの名前。

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。それ以外の場合は、自動スナップショットと Lightsail アカウントに保存されている重複した手動スナップショットのスナップショットストレージ料金が請求されます。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**
>
> これらのコマンドの GetAutoSnapshots および CopySnapshot API オペレーションの詳細については、Lightsail API ドキュメントCopySnapshotのGetAutoSnapshots「」および「」を参照してください。

スナップショットを使用して Linux/Unix Lightsail インスタンスをバックアップする

Linux/Unix ベースの Amazon 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、元のマシン設定 (メモリ、CPU、ディスクサイズ、データ転送レート) と一致します。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ている場合、Lightsail はスナップショットの一部としてそれらの追加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。

Lightsail のスナップショットを作成するには、そのインスタンスが既に存在し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 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**
>
> 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、Lightsail では、同じサイズ以上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。

スナップショットからより大きなインスタンスサイズを作成するには、Lightsail コンソール、create-instances-from-snapshot CLI コマンド、または CreateInstancesFromSnapshotAPIオペレーションを使用できます。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 バンドルの詳細については、「Lightsail の料金」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、データ転送速度などの情報が含まれます。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、まず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。その後、そのスナップショットからインスタンスを作成すると、その Windows インスタンスを初めて実行したかのように out-of-box エクスペリエンスが得られます。

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。

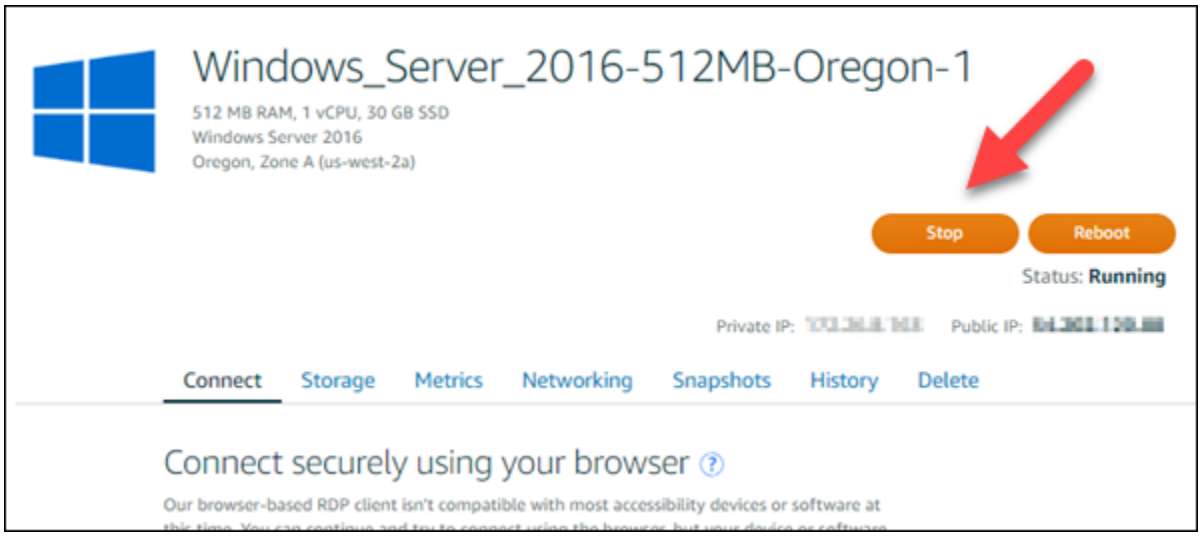
Sysprep を実行する前にスナップショットを作成すると、このバックアップスナップショットを使用して作成するインスタンスでは、元のインスタンスと同じ管理者パスワードが使用されます。Lightsail コンソールでブラウザベースのRDPクライアントを使用してこれらのインスタンスに接続することはできません。ただし、独自のRDPクライアントと元のインスタンスと同じ管理者パスワードを使用して接続できます。詳細については、「Windows コンピュータの Amazon Lightsail でリモートデスクトップ接続クライアントを使用して Windows インスタンスに接続する」を参照してください。

> **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 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

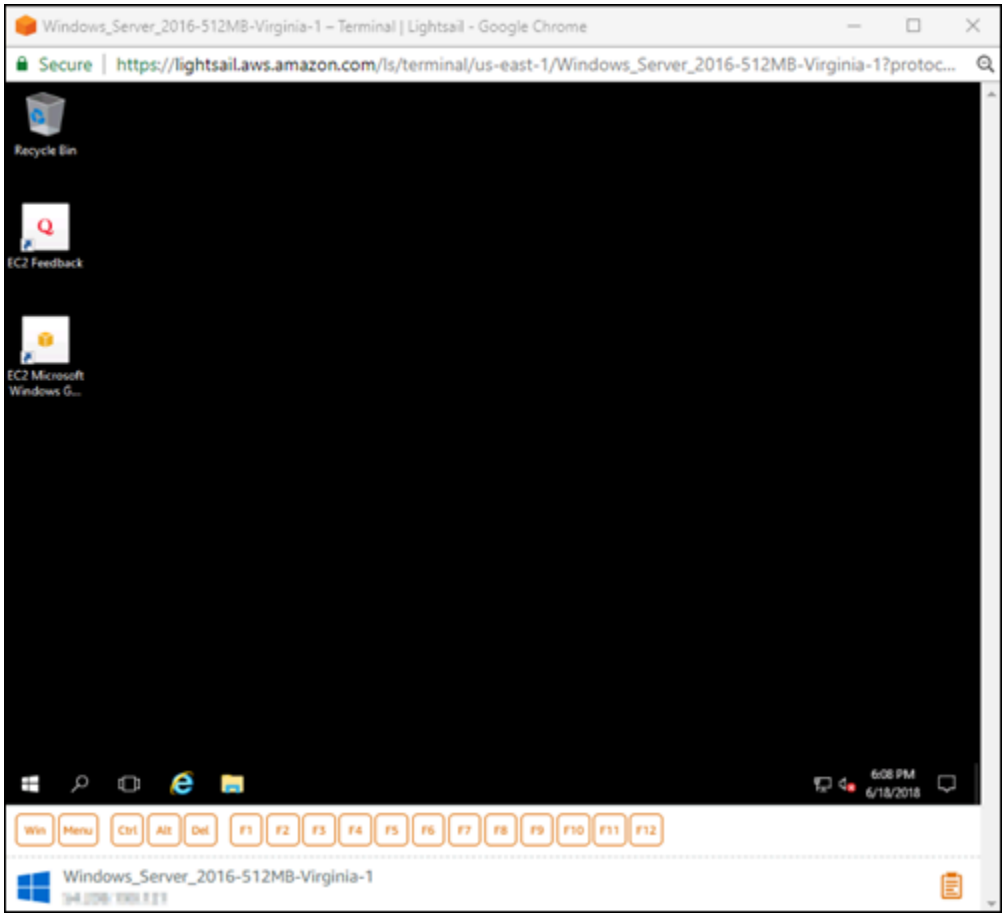
バックアップスナップショットを作成したので、次は Windows Server インスタンスで Sysprep を実行します。これに伴ってインスタンスがシャットダウンされ、スナップショットを作成できるようになります。Sysprep の詳細については、Microsoft のドキュメントで「Sysprep Overview」を参照してください。

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 インスタンスEC2LaunchSettingsで呼び出され、Windows Server 2012 インスタンスでは Ec2ConfigService 2Settings で呼び出されます。

インスタンスに接続して Sysprep を実行するには

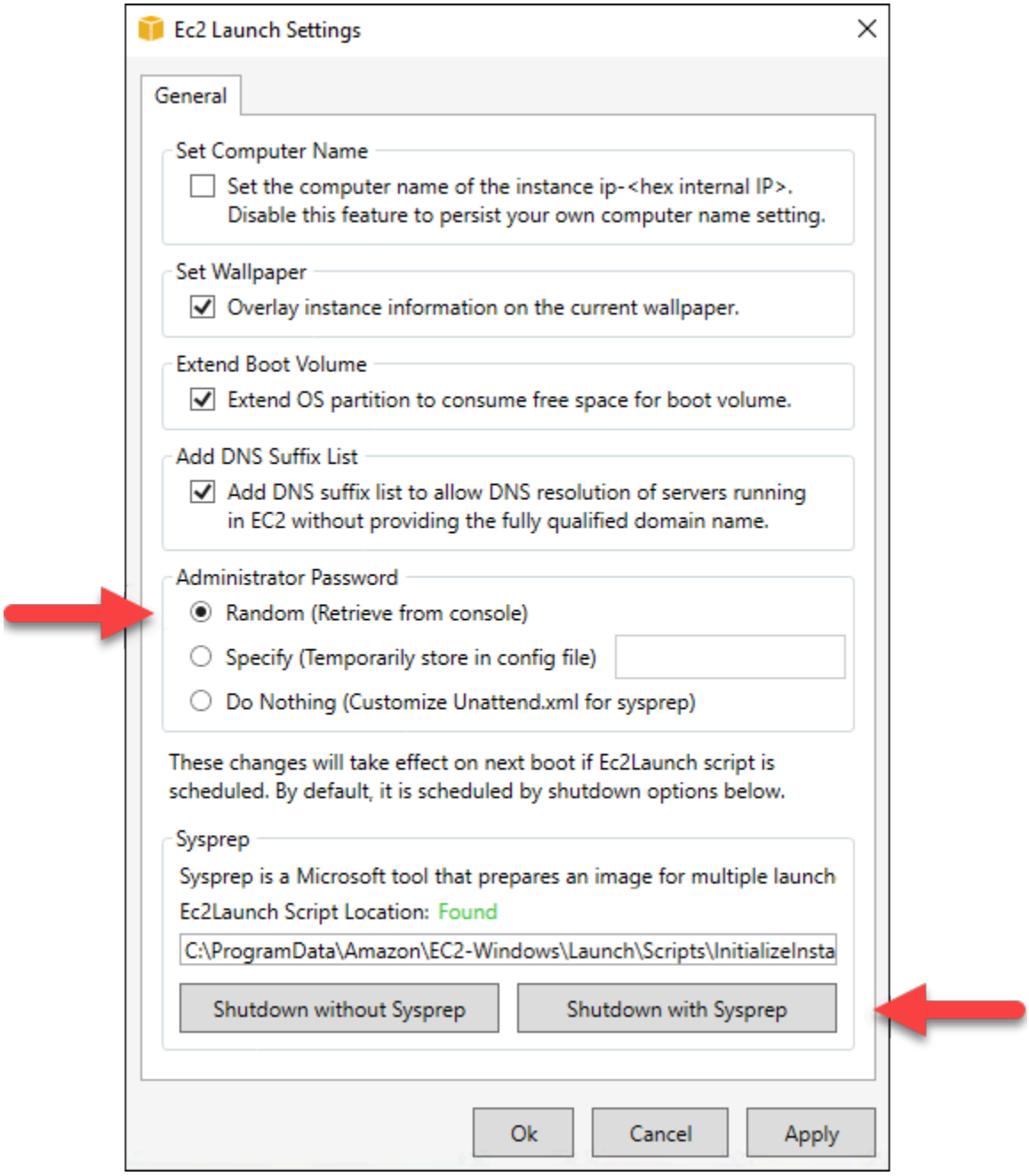
1. インスタンス管理ページで、接続タブを選択し、 を使用して接続を選択しますRDP。

   次の例に示すように、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 2022、Windows Server 2019、および Windows Server 2016 インスタンスで、「 の開始」を選択し、Ec2LaunchSettings」を選択します。
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。



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。

   インスタンスが Sysprep の実行を開始し、RDP接続がシャットダウンし、数分後に Lightsail インスタンスの実行が停止します。

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。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、これらのインスタンスに接続できます。

Lightsail コンソールでスナップショットを作成するには

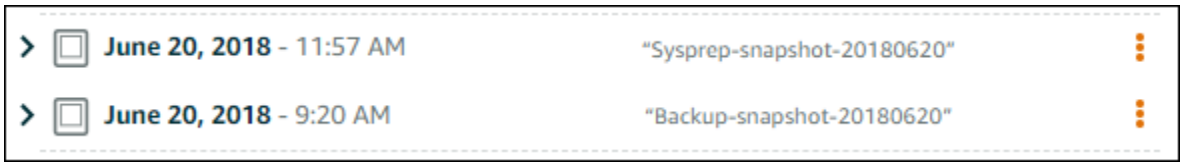
1. Lightsail コンソールに戻ります。
2. Windows Server インスタンスのインスタンス管理ページで、[Snapshots (スナップショット)] タブを選択します
3. このページの [手動スナップショット] セクションで、[スナップショットの作成] を選択し、スナップショットの名前を入力します。

   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。

バックアップまたはベースライン用の Lightsail ブロックストレージディスクスナップショットを作成する

Amazon Lightsail では、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ディスクスナップショットを作成できます。

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 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。

Lightsail の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

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。

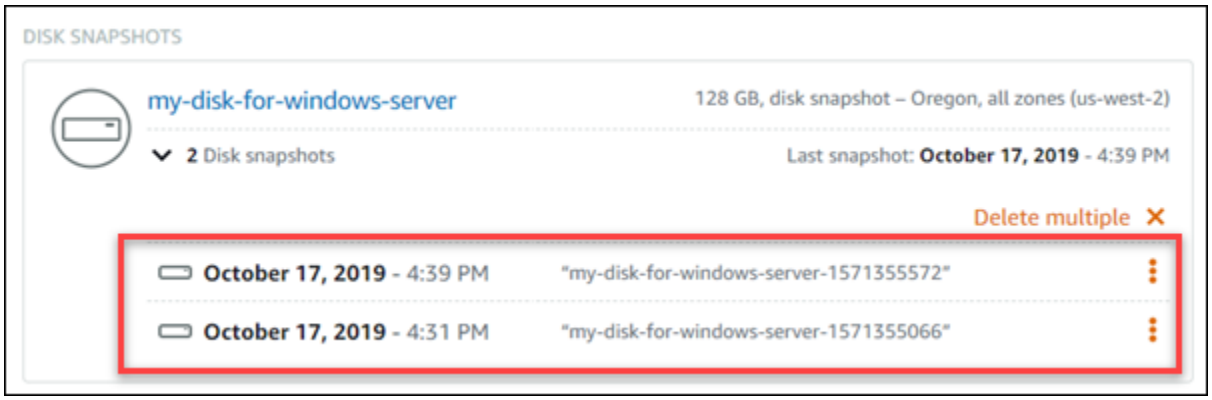
ブロックストレージディスクのスナップショットを、新しいディスクまたはデータバックアップのためのベースラインとして使用できます。ディスクのスナップショットを定期的に作成する場合、スナップショットは差分になります。最後にスナップショットを作成した時点から、ディスク上で変更があったブロックだけが、新しいスナップショットに保存されます。スナップショットの保存は増分ベースで行われるものの、最新のスナップショットさえあればディスク全体を復元できるようにスナップショット削除プロセスは設計されています。ブロックストレージディスクのスナップショットを作成するには、「ブロックストレージディスクのスナップショットを作成する」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

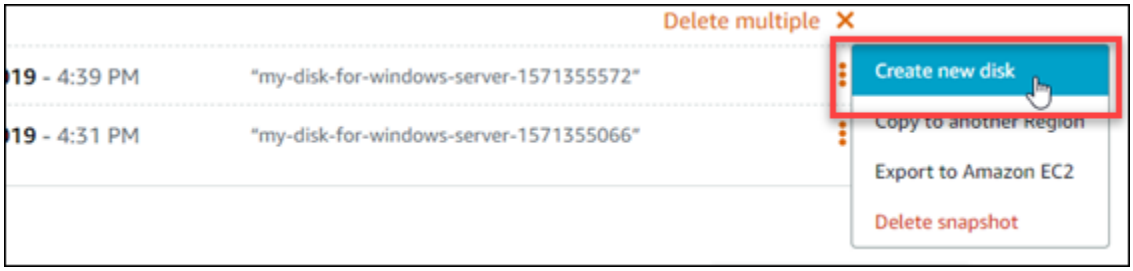
Lightsail の 2 つの場所のいずれかで、ディスクスナップショットから新しいインスタンスを作成できます。Lightsail ホームページのスナップショットタブ、またはディスク管理ページのスナップショットタブです。

Lightsail ホームページから

1. Lightsail ホームページの左側のナビゲーションバーで、スナップショット 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。

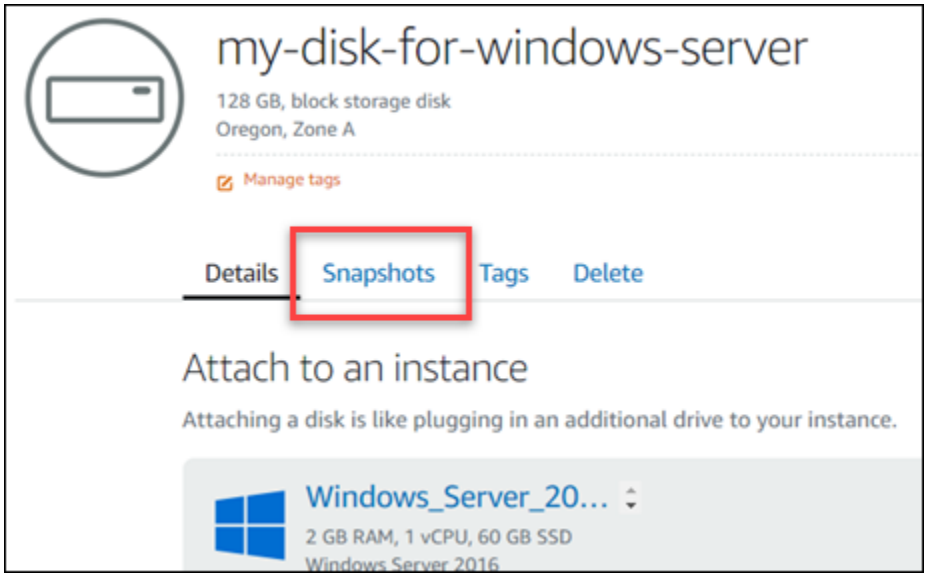


3. 新しいディスクを作成するスナップショットの横にあるアクションメニューアイコン (⋮) を選択し、[新しいディスクの作成] を選択します。

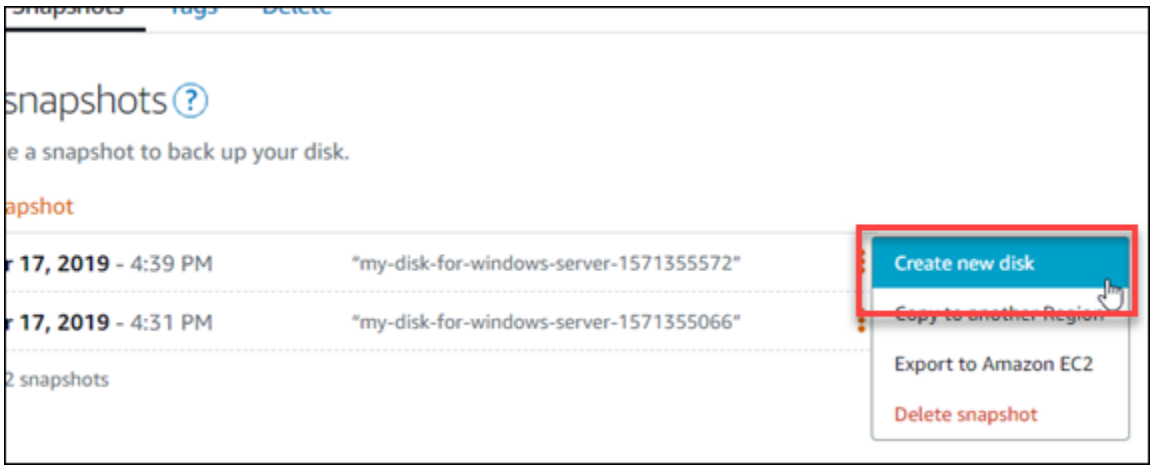


Lightsail のディスク管理ページから

1. Lightsail ホームページの左側のナビゲーションバーで、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

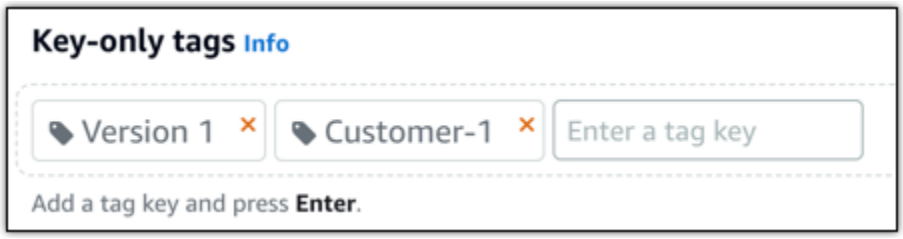
1. 新しいディスクのアベイラビリティーゾーンを選択するか、デフォルト () を受け入れますus-east-2a。

   新しいディスクは、ソースディスク AWS リージョン と同じ 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:

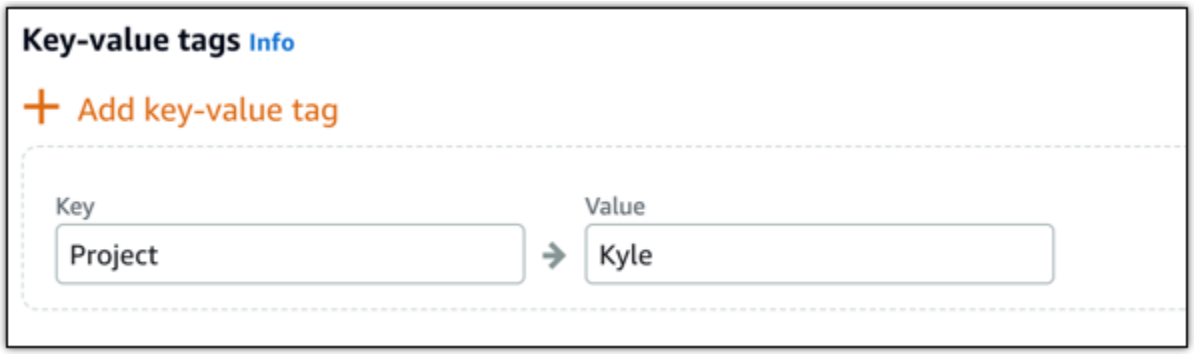
   - Lightsail アカウントの各 AWS リージョン 内で一意である必要があります。
   - 2〜255 文字を使用する必要があります。
   - 先頭と末尾は英数字または数字を使用する必要があります。
   - 英数字、数字、ピリオド、ダッシュ、アンダースコアを使用することができます。
4. 以下のいずれかのオプションを選択して、ディスクにタグを追加します。

   - [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。

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。

インスタンスのルートボリュームスナップショットは、 AWS Command Line Interface （ AWS CLI) または を使用して作成します AWS CloudShell。スナップショットを作成したら、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

AWS Command Line Interface （ AWS CLI ）、または AWS CloudShell を使用して、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します。 CloudShell はブラウザベースの事前認証済みシェルで、Lightsail コンソールから直接起動できます。詳細については、「Lightsail オペレーション AWS CLI の をセットアップする 」および「で Lightsail リソースを管理する AWS CloudShell」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 CloudShell 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インスタンスルートボリュームスナップショットを作成します。

```
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。

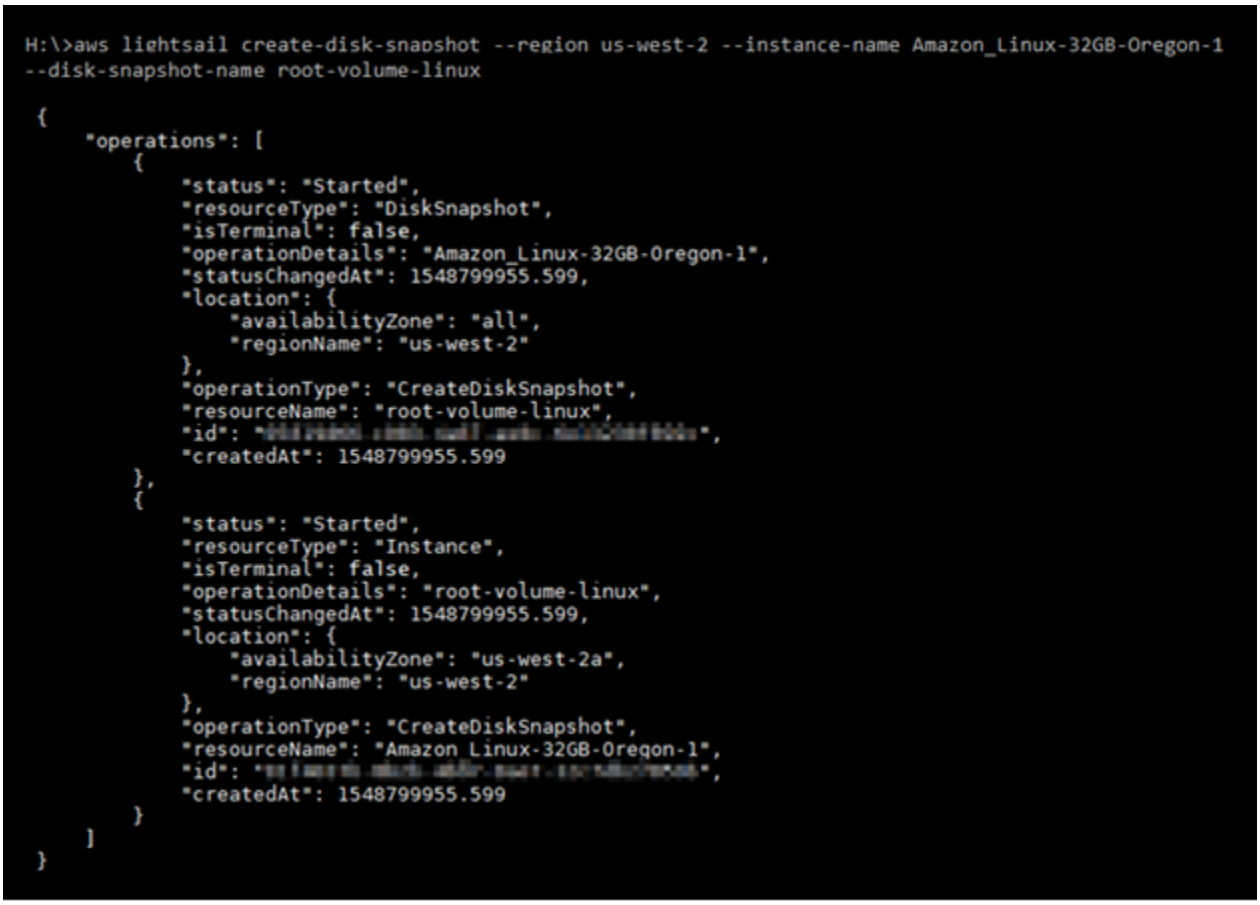
- *AWSRegion* インスタンス AWS リージョン の を使用します。
- *InstanceName* を、ルートボリュームをバックアップするインスタンスの名前に置き換えます。

- *DiskSnapshotName* 作成する新しいディスクスナップショットの名前。

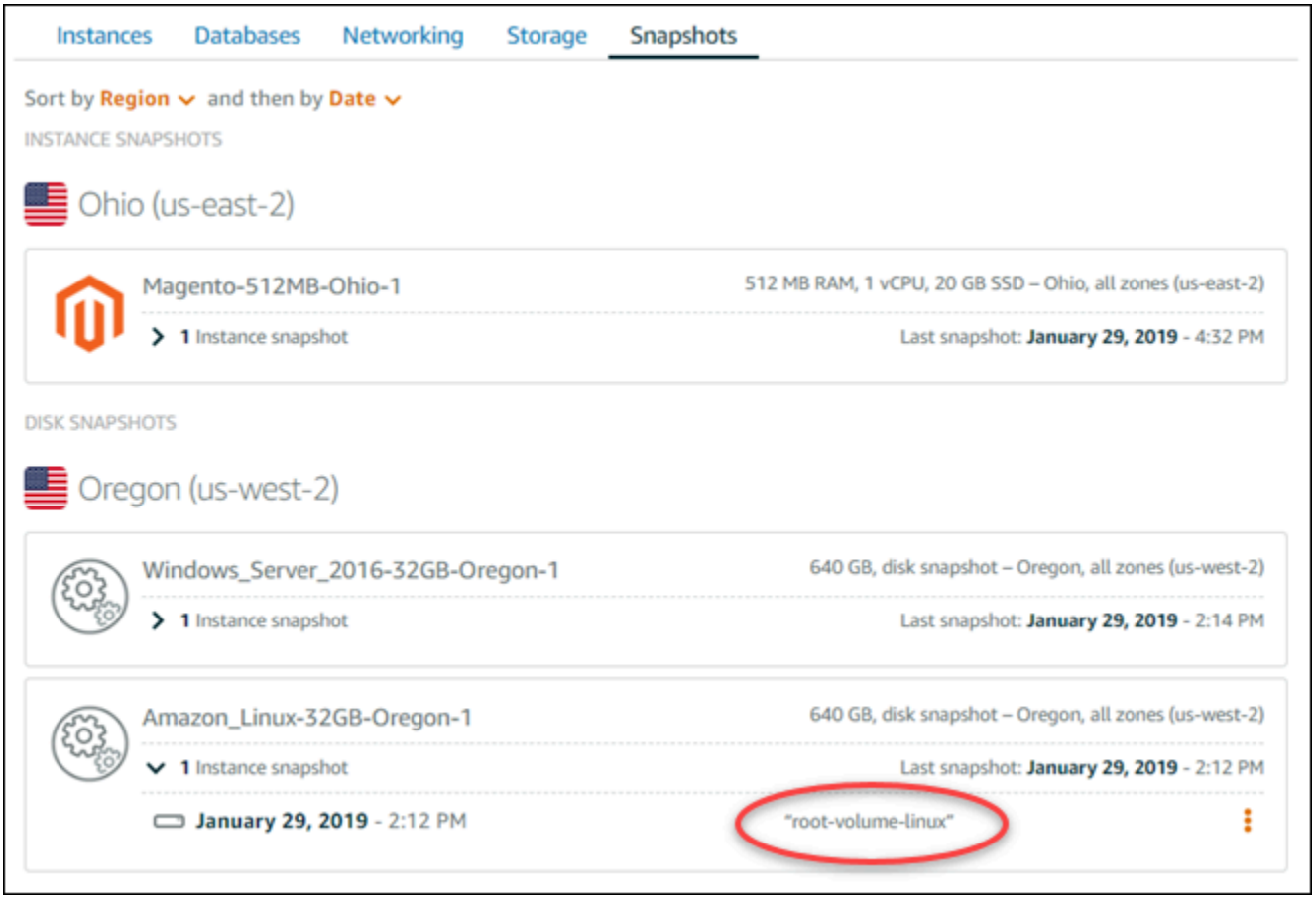
例:

```
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、次の例に示すように、スナップショットタブを選択し、ディスクスナップショットセクションにスクロールすることで、Lightsail ホームページで表示できます。



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**
>
> 新しいブロックストレージディスクは、ソーススナップショット AWS リージョン と同じ 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、「ある 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。

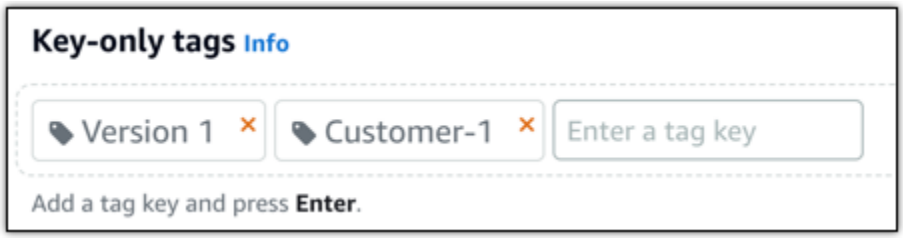
1. Lightsail コンソール にサインインします。

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。

   - [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。

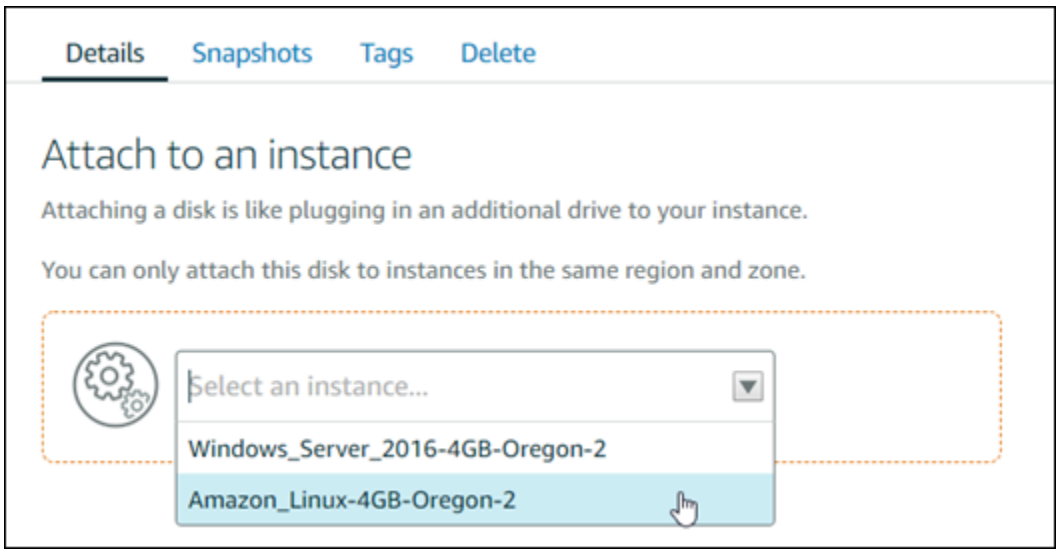
     

   - [key-value タグの作成] から [キー] テキストボックスにキーを入力し、[値] テキストボックスに値を入力します。タグの入力を完了したら、[保存] を選択し、追加しない場合は、[キャンセル] を選択します。

     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. [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。

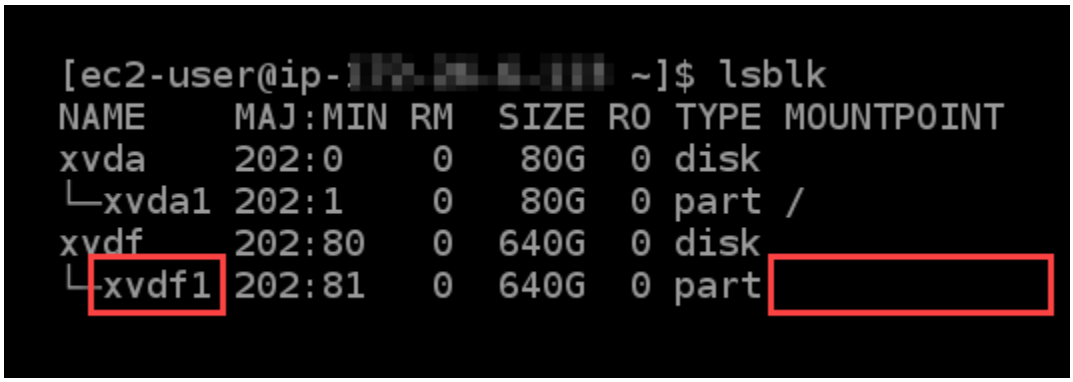
Linux または Unix インスタンスにブロックストレージディスクをマウントしてアクセスする

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、次のコマンドを入力して、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* ブロックストレージディスクがマウントされ、アクセス可能なディレクトリの名前。

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
```

コマンドは、以下のように置き換えます。

- *DeviceName* ブロックストレージディスクデバイスの名前。
- *MountPoint* を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。

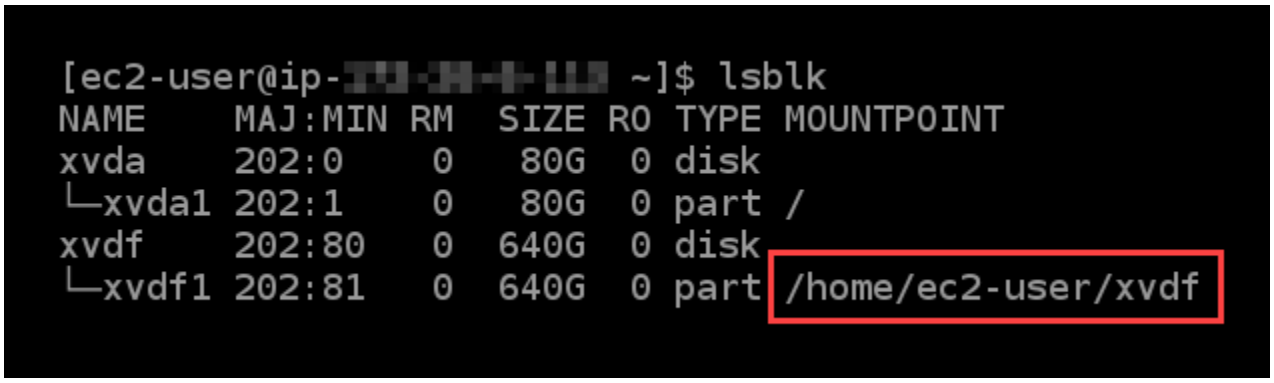
例:

```
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。

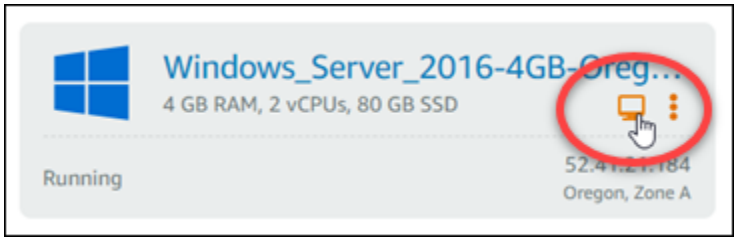
```
lsblk
```

次の例のような結果が表示されます。この例では、*xvdf1* デバイスがマウントされ、 でアクセス可能になりました */home/ec2-user/xvdf* ディレクトリ。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。

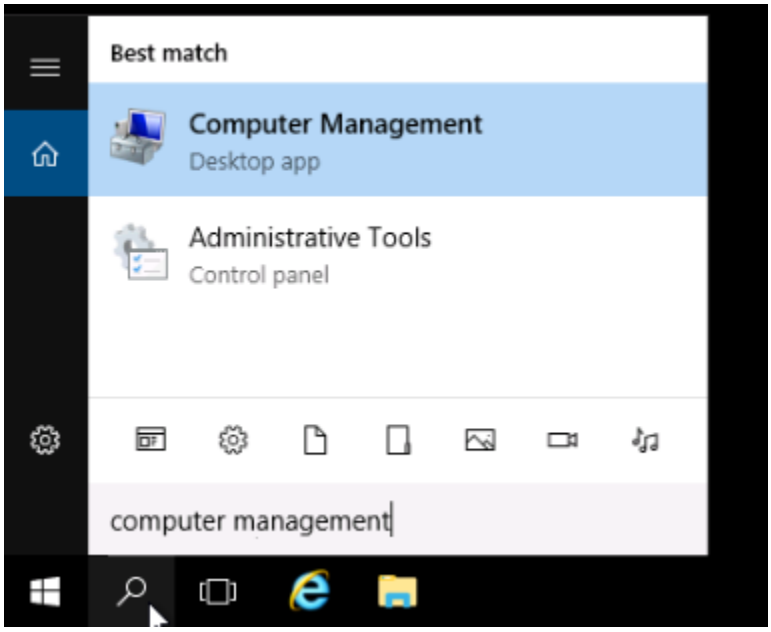


Windows インスタンスでブロックストレージディスクをオンラインにしてアクセスします。

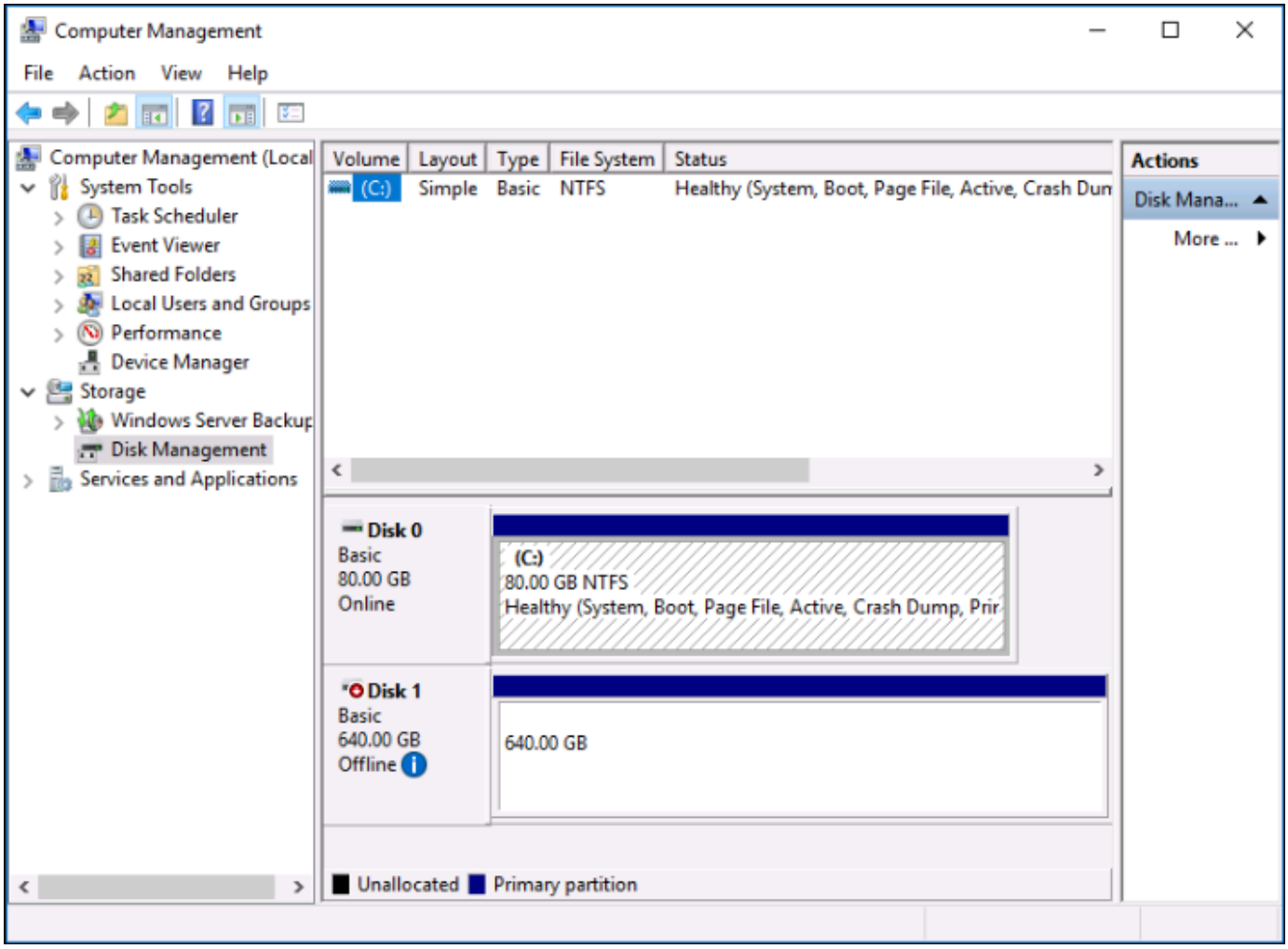
1. Lightsail ホームページ で、ブロックストレージディスクをアタッチした Windows インスタンスのブラウザベースのRDPクライアントアイコンを選択します。



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管理を検索し、結果からコンピュータ管理を選択します。

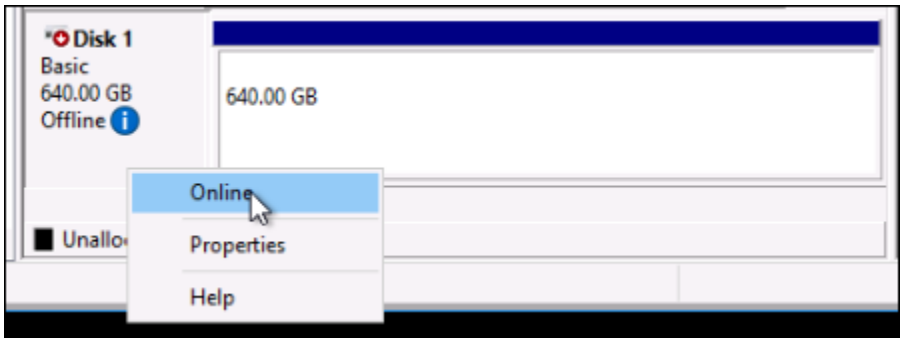


3. [コンピュータの管理] コンソールの左側のナビゲーションメニューで、以下の例のように [ディスク管理] を選択します。

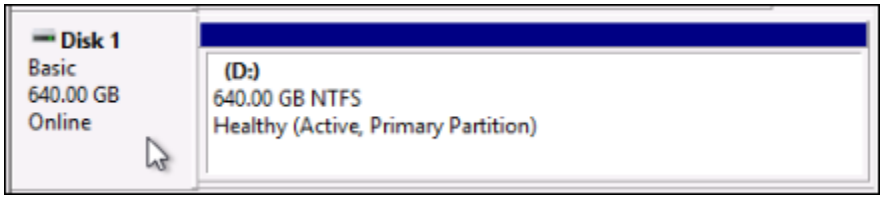


4. 最近インスタンスにアタッチしたディスクを見つけます。[オフライン] とラベル付けされているはずです。

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。

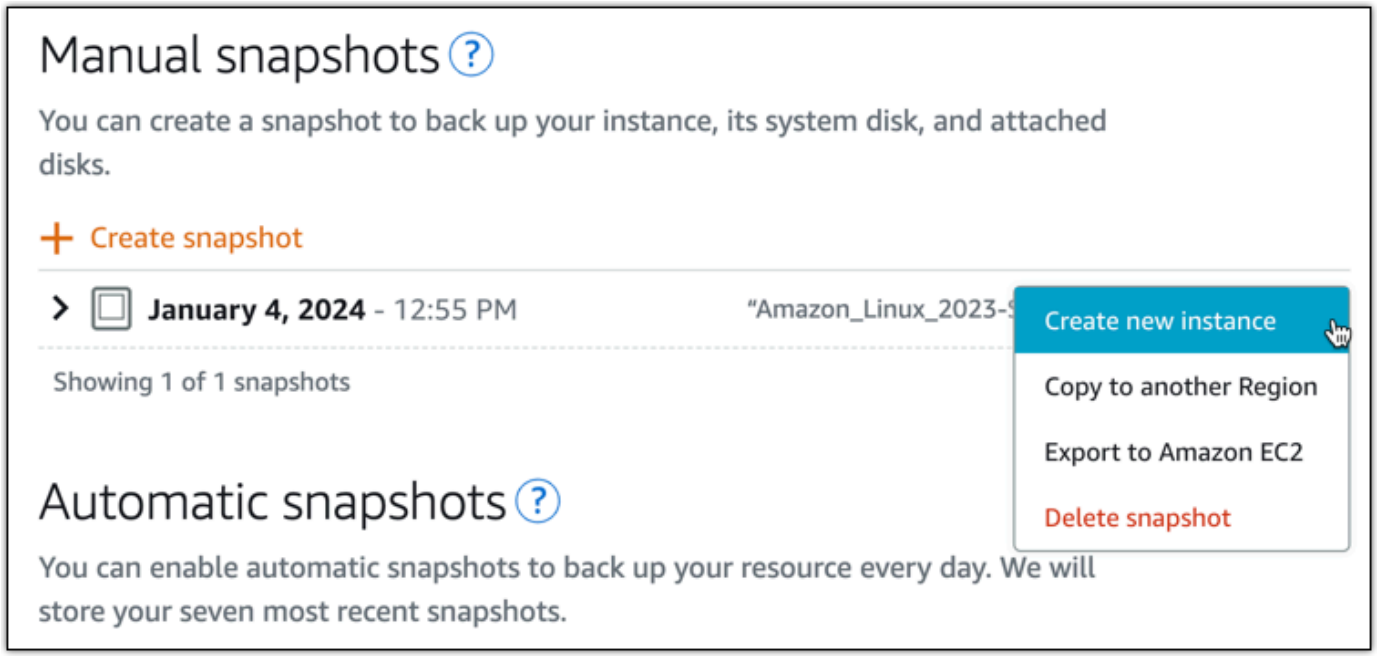


スナップショットから Lightsail インスタンスを作成する

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など、新しいインスタンスの属性はデュアルスタックまたは IPv6のみ変更できます。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。

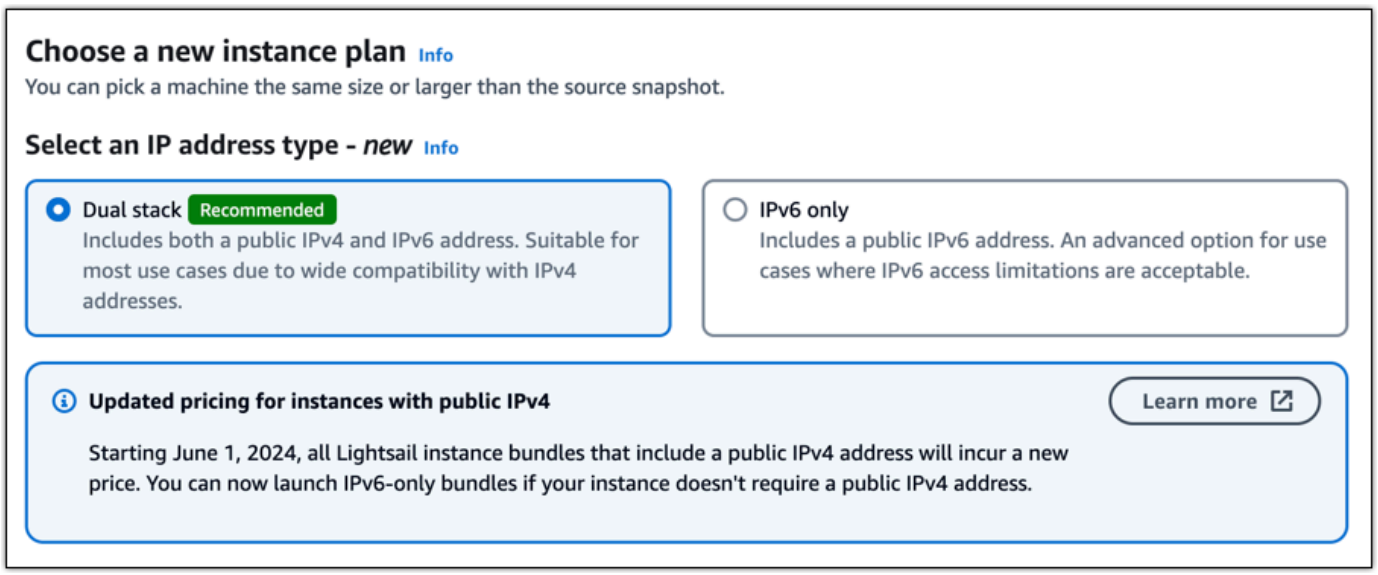
そのスナップショットから別のインスタンスを作成する前に、インスタンスのスナップショットが必要です。詳細については、スナップショットを使用して Linux/Unix Lightsail インスタンスをバックアップする または Lightsail Windows Server インスタンスのスナップショットを作成する を参照してください。

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のみの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。のみのインスタンスプランの詳細については、IPv6「」を参照してくださいLightsail インスタンスの IPv6-onlyネットワークを設定する。

> **Note**
>
> 元のインスタンスよりも小さい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。

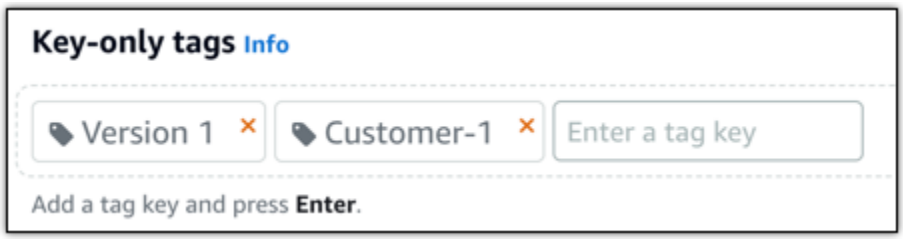


6. インスタンスの名前を入力します。

   リソース名:

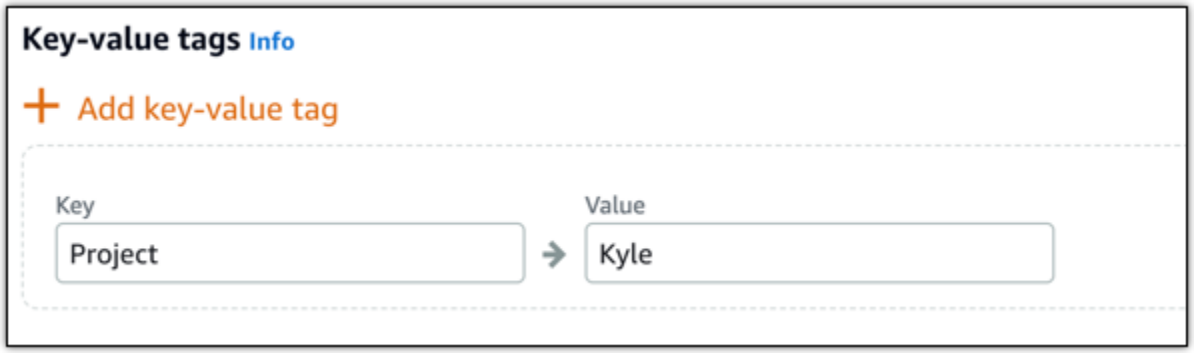
   - Lightsail アカウント AWS リージョン ごとに一意である必要があります。
   - 2～255 文字を使用する必要があります。
   - 英数字で開始および終了する必要があります。
   - 英数字、ピリオド、ダッシュ、アンダースコアを含めることができます。
7. 以下のいずれかのオプションを選択して、インスタンスにタグを追加します。

- [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。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。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

タンスにコピーされます。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。

スナップショットから Lightsail インスタンス、ストレージ、またはデータベースをアップグレードする

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をアップサイズするには、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いより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パブリックIPv4アドレスは、インスタンスを停止して起動すると変更されます。オプションで、静的IPv4アドレスを作成してインスタンスにアタッチできます。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指すように、ドメインのDNSレコードで静的 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です。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. 新しい大きなリソースの作成に使用するスナップショットを持つ 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、起動スクリプト を追加するか、そのスクリプトへの接続に使用するSSHキーを変更できます。

   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. [Create] (作成) を選択します。

   Lightsail では、新しいリソースの管理ページが表示され、管理を開始できます。

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 AWS CLI

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 コンソール内からすべてを実行するか、 AWS Command Line Interface （ AWS CLI) を使用して実行できます。

現在の Lightsail インスタンスのスナップショットを作成し、そのスナップショットに基づいて必要なコンピューティング能力を備えた新しいより大きなインスタンスを作成する方法を示します。

> **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、まだインストールしていない場合は、 をインストールする必要があります AWS CLI。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. で AWS CLI、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。例えば、中規模 Linux ベースのバンドルは 1 か月USDあたり 24 USD で、4.0 GB の RAM、80 GB のSSDストレージなどがあります。

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": [
        {
            "price": 5.0,
            "cpuCount": 2,
            "diskSizeInGb": 20,
            "bundleId": "nano_3_0",
            "instanceType": "nano",
            "isActive": true,
            "name": "Nano",
            "power": 298,
            "ramSizeInGb": 0.5,
            "transferPerMonthInGb": 1024,
            "supportedPlatforms": [
                "LINUX_UNIX"
            ],
            },
        {
            "price": 7.0,
            "cpuCount": 2,
            "diskSizeInGb": 40,
            "bundleId": "micro_3_0",
            "instanceType": "micro",
            "isActive": true,
            "name": "Micro",
            "power": 500,
            "ramSizeInGb": 1.0,
            "transferPerMonthInGb": 2048,
            "supportedPlatforms": [
                "LINUX_UNIX"
```

```
            ],
            },
        {
            "price": 12.0,
            "cpuCount": 2,
            "diskSizeInGb": 60,
            "bundleId": "small_3_0",
            "instanceType": "small",
            "isActive": true,
            "name": "Small",
            "power": 1000,
            "ramSizeInGb": 2.0,
            "transferPerMonthInGb": 3072,
            "supportedPlatforms": [
                "LINUX_UNIX"
            ],
            },
        {
            "price": 24.0,
            "cpuCount": 2,
            "diskSizeInGb": 80,
            "bundleId": "medium_3_0",
            "instanceType": "medium",
            "isActive": true,
            "name": "Medium",
            "power": 2000,
            "ramSizeInGb": 4.0,
            "transferPerMonthInGb": 4096,
            "supportedPlatforms": [
                "LINUX_UNIX"
            ],
            },
        {
            "price": 44.0,
            "cpuCount": 2,
            "diskSizeInGb": 160,
            "bundleId": "large_3_0",
            "instanceType": "large",
            "isActive": true,
            "name": "Large",
            "power": 3000,
            "ramSizeInGb": 8.0,
            "transferPerMonthInGb": 5120,
            "supportedPlatforms": [
```

```
                "LINUX_UNIX"
            ],
            },
    ]
}
```

2. 必要なバンドルbundleIdの値を見つけます。詳細については、「Lightsail の料金」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しい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**
>
> 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 CLI。`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. 次に、Lightsail コンソールで新しいインスタンスを開き、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。

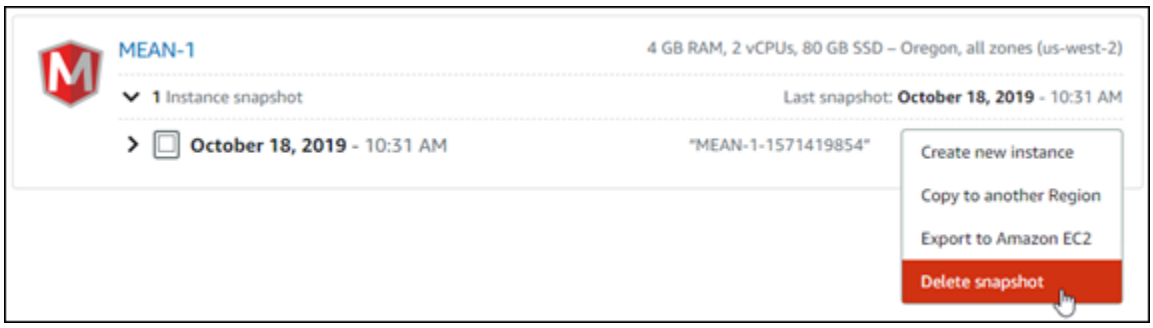
- 以前のインスタンスを使用しない場合は削除できます。これを行うには、Lightsail コンソールまたは delete-instance CLI コマンド を使用します。
- 以前のスナップショットが必要ない場合は削除できます。これを行うには、Lightsail コンソールまたはdelete-instance-snapshot CLIコマンド を使用します。
- 以前のインスタンスに静的 IP アドレスがアタッチされていた場合は、そのまま新しいインスタンスにアタッチできます。これを行うには、コンソールを使用します。「静的 IP を作成してインスタンスにアタッチする」を参照してください。

未使用の Lightsail スナップショットを削除して月額料金が発生しないようにする

月額料金が発生しないように、不要になったインスタンス、データベース、ディスクスナップショットは Amazon Lightsail で削除します。

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。

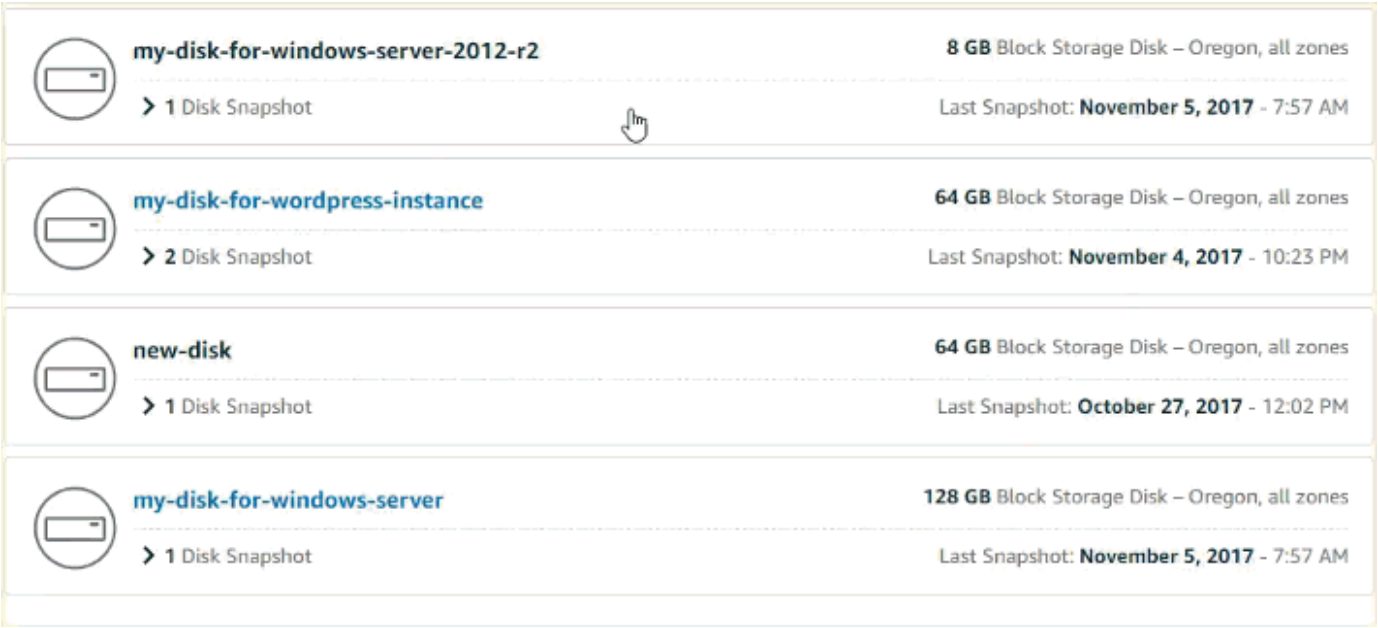


4. [はい] を選択して、スナップショットを削除することを確定します。

> **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 ホームページから、スナップショット 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。

間で Lightsail スナップショットをコピーする AWS リージョン

Amazon Lightsail では、インスタンススナップショットをコピーし、ストレージディスクスナップショット AWS リージョン をある から別の へ、または同じリージョン内でブロックできます。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 スナップショットのコピー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

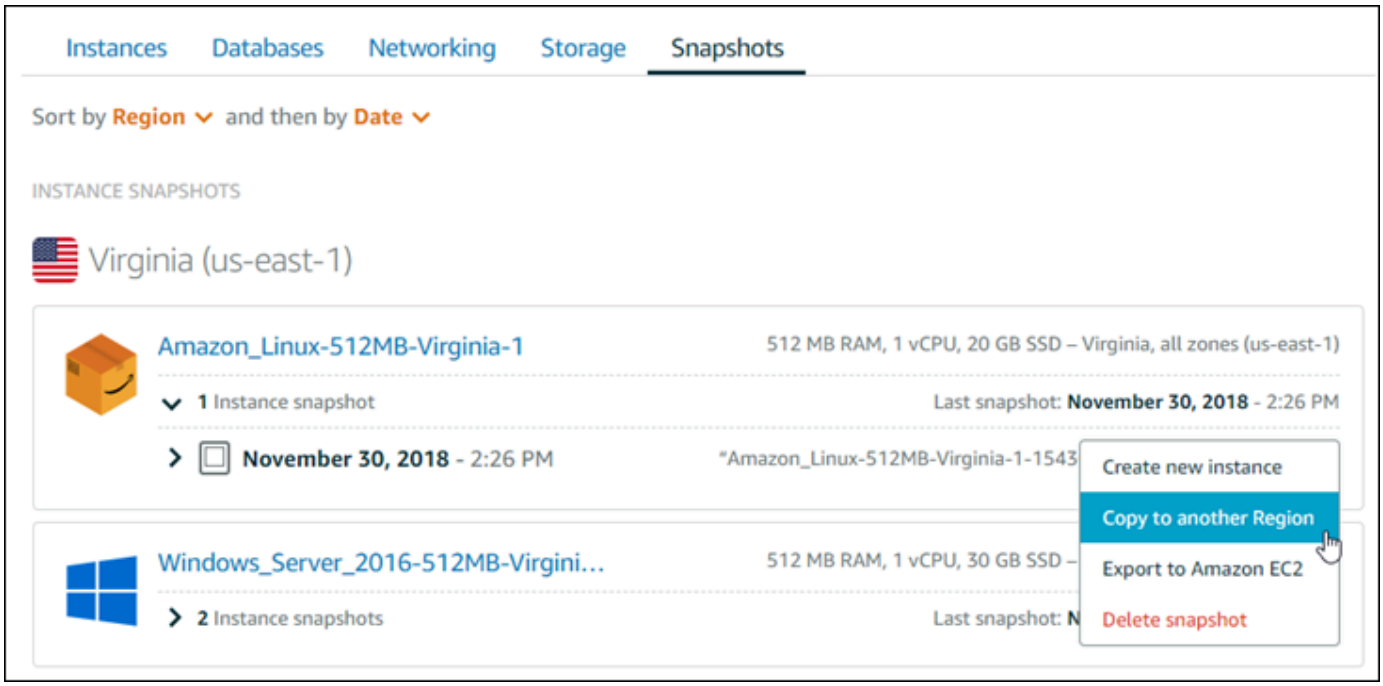
のスナップショットをコピーする

Lightsail インスタンススナップショットをコピーし、ストレージディスクスナップショット AWS リージョン を 1 つの から別の へ、または同じリージョン内でブロックできます。

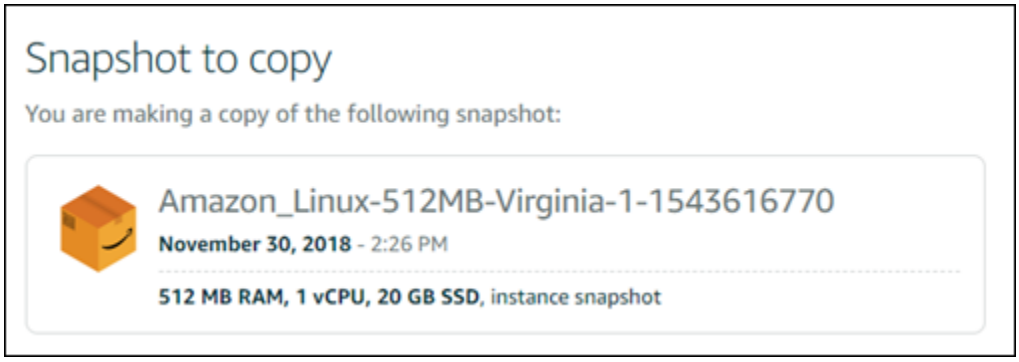
Lightsail スナップショットをコピーするには

1. Lightsail コンソール にサインインします。
2. Lightsail のホームページから、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。



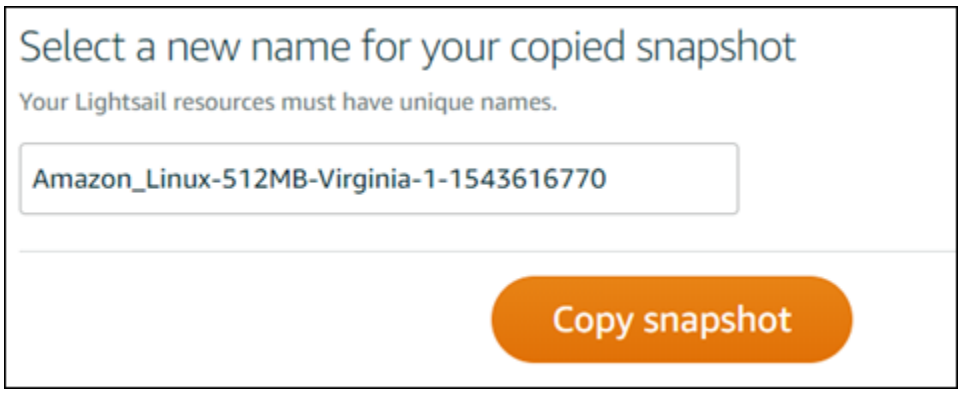
6. このページの [リージョンの選択] セクションで、スナップショットのコピー先のリージョンを選択します。
7. スナップショットコピーの名前を入力します。

   リソース名:

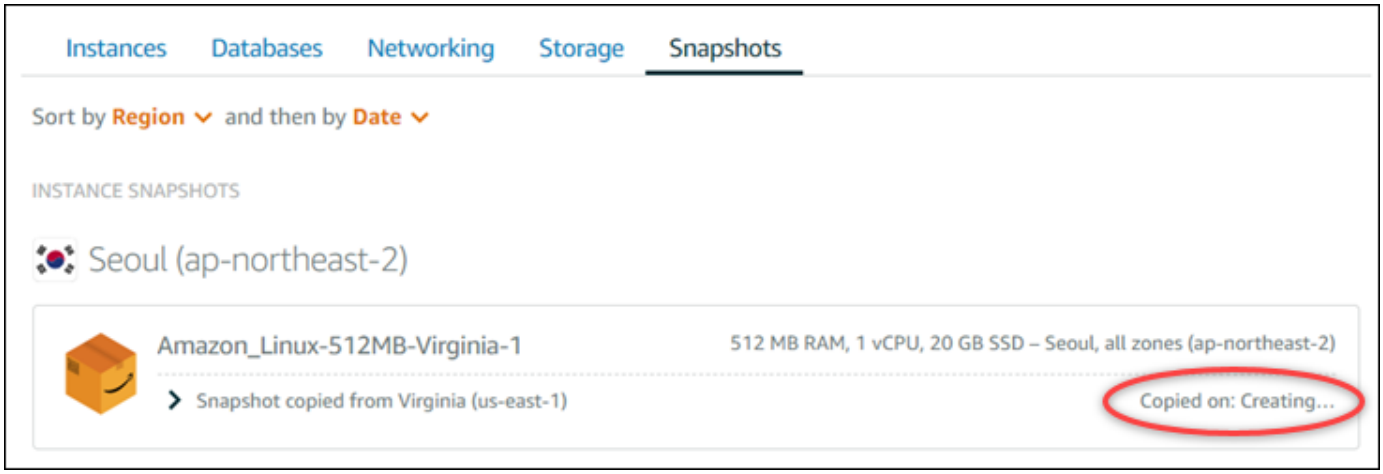
   - Lightsail アカウントの各 AWS リージョン 内で一意である必要があります。
   - 2〜255 文字を使用する必要があります。
   - 先頭と末尾は英数字または数字を使用する必要があります。

- 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。Lightsail ホームページの「スナップショット」タブを参照し、次のスクリーンショットに示すように、ステータスが「Creating」のスナップショットを検索することで、スナップショットコピーのステータスを確認できます。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 でスナップショットを別のリージョンにコピーした後に実行できる追加の手順をいくつか紹介します。

- 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。

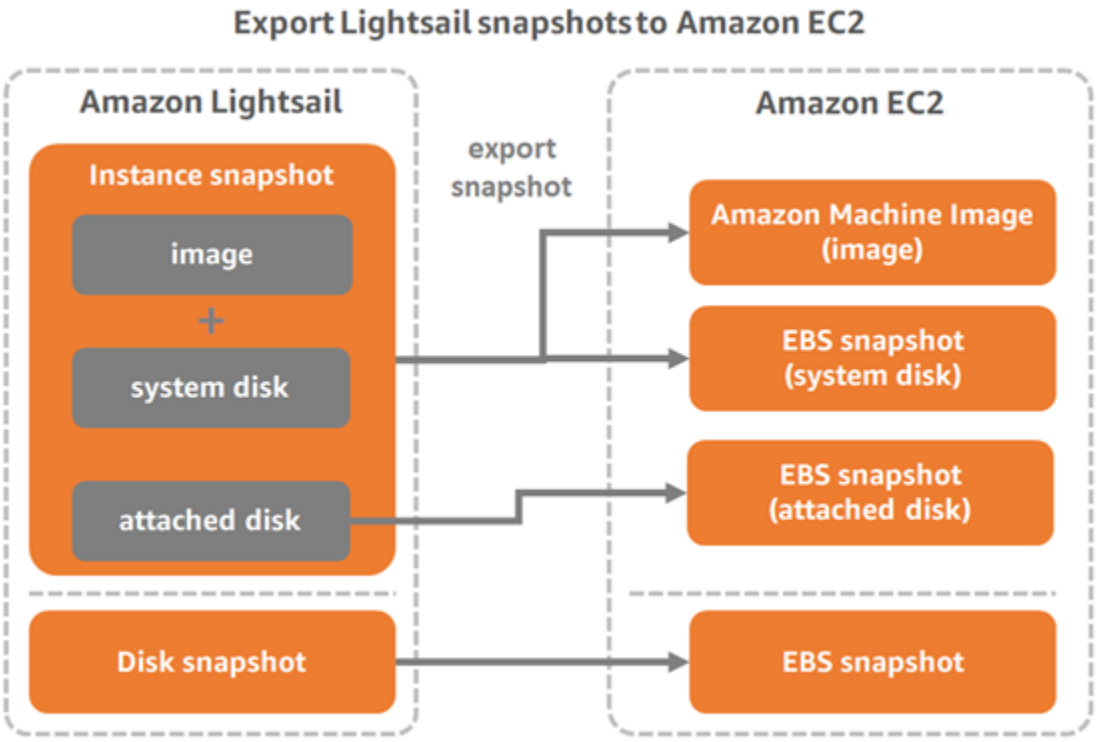
Lightsail スナップショットを Amazon にエクスポートする方法について説明します。 EC2

Lightsail スナップショットを Amazon にエクスポートする方法EC2、エクスポートされたスナップショットからEC2リソースを作成する方法、互換性のあるEC2インスタンスタイプを選択する方法、EC2インスタンスに接続する方法、および Lightsail スナップショットからEC2作成されたインスタンスを保護する方法について説明します。Amazon Lightsail インスタンスとブロックストレージディスクスナップショットは、次のいずれかの方法を使用して Amazon Elastic Compute Cloud (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、「Amazon へのスナップショットのエクスポートEC2」を参照してください。
- Lightsail API、 AWS Command Line Interface （ AWS CLI ）、または SDKs。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、 cPanel & WHM (CentOS 7) インスタンスのスナップショットを Amazon にエクスポートすることはできませんEC2。スナップショットは Lightsail AWS リージョン から Amazon に同じ にエクスポートされますEC2。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 の別のリージョンにスナップショットをコピーしてから、エクスポートを実行します。詳細については、「ある 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。

Lightsail インスタンス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) と Amazon Elastic Block Store (Amazon EBS) スナップショットが Amazon に作成されますEC2。これは、Lightsail インスタンスがイメージとシステムディスクで構成されているが、どちらも Lightsail コンソールで単一のインスタンスエンティティとしてグループ化されているためです。スナップショットの作成時にソース 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、アタッチされたディスクごとに追加のEBSスナップショットが Amazon に作成されますEC2。Lightsail ブロックストレージディスクスナップショットをエクスポートすると、Amazon で単一のEBSスナップショットが作成されますEC2。Amazon でエクスポートされたすべてのリソースEC2には、Lightsail に対応するものとは異なる独自の一意の識別子があります。



> **Note**
>
> Lightsail 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 にエクスポートしますEC2。の詳細についてはSLRs、「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 コンソールのエクスポートセクションを使用して、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、「Lightsail でスナップショットのエクスポートステータスを追跡する」を参照してください。

エクスポートされた Lightsail スナップショットから Amazon EC2リソースを作成する

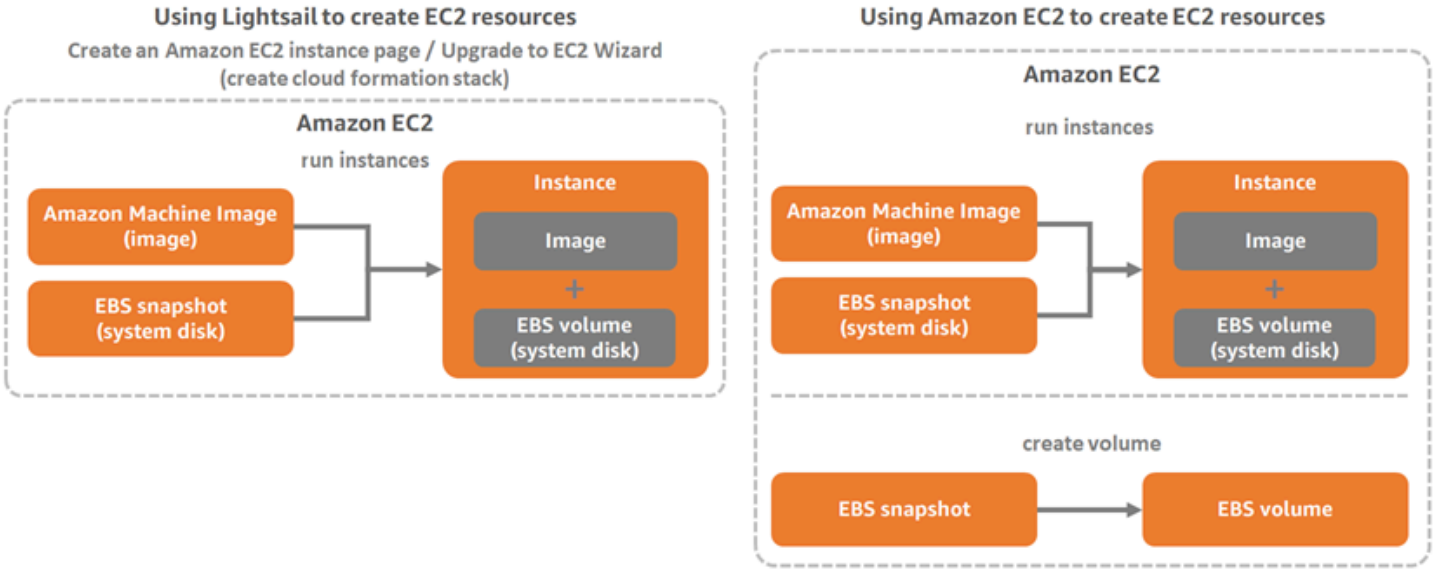
Lightsail スナップショットがエクスポートされ、Amazon で EC2 (AMI、EBSスナップショット、またはその両方として) 使用可能になったら、次のいずれかの方法を使用してスナップショットから Amazon EC2リソースを作成できます。

- Lightsail コンソールの「Amazon EC2インスタンスの作成」ページ。Amazon Wizard へのアップグレードとも呼ばれますEC2。詳細については、「エクスポートされ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 SDKs。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 を使用して、エクスポートされたEC2インスタンススナップショットから Amazon インスタンスを作成できますが、エクスポートされ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。そのためには、Amazon EC2コンソール、API、または を使用する必要があります AWS CLI。詳細については、「エクスポートされ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 SDKs。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。

エクスポートされたEC2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 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EBSスナップショットのエクスポートによって生成された AMIおよび スナップショットは、自動的にリンクされてEC2インスタンスを形成します。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 (EBS スナップショット) を使用して、Amazon でEBSボリュームを作成できますEC2。



> **Note**
>
> Lightsail は CloudFormation スタックを使用して、 でインスタンスおよび関連リソースを作成しますEC2。詳細については、AWS CloudFormation 「Lightsail のスタック」を参照してください。

エクスポートされ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には、しばらく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。Lightsail コンソールのエクスポートセクションを使用して、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、「」を参照してくださいLightsail でスナップショットのエクスポートステータスを追跡する。

Amazon EC2インスタンスタイプの選択

Amazon EC2 は、Lightsail で利用可能なものよりも幅広いインスタンスオプションを提供しています。Amazon ではEC2、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 は、Amazon EC2インスタンスの作成ページでこれらのオプションを提供します。ただし、エクスポートされたスナップショットから新しいインスタンスを作成EC2するために Amazon を使用する場合、より多くのインスタンスタイプオプションを使用できます。EC2 インスタンスタイプの詳細については、Amazon 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。 EC2

エクスポートされ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 のインスタンス料金の違いを理解することが重要ですEC2。インスタンスの料金の詳細については、Lightsail の料金ページと Amazon のEC2料金ページを参照してください。

Lightsail と Amazon EC2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が有効になっていないため、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 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インスタンスを作成するときに、旧世代のEC2インスタンスタイプ (T2、M4, C4、または R4) を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インスタンスの作成」ページを使用してインスタンスを作成するときに表示されます。 EC2

ソース Lightsail EC2インスタンスに互換性がない場合に最新世代のインスタンスタイプを使用するには、以前の世代のEC2インスタンスタイプ (T2、M4, C4を使用して新しいインスタンスを作成し、ネットワークドライバーを更新してから、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アップグレードする必要があります。 R4 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 EC2インスタンスへの接続方法と同様に、Amazon 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SSHの場合は を使用し、Windows Server インスタンスRDPの場合は を使用します。ただし、使用しているブラウザのバージョンEC2によっては、Lightsail コンソールで使用したブラウザベースの SSH/RDP クライアントが Amazon で使用できない場合があるため、EC2インスタンスに接続するように独自の SSH/RDP クライアントを設定する必要がある場合があります。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 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 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

エクスポートされた Lightsail スナップショットからEC2インスタンスを作成した後、新しいインスタンスのセキュリティを向上させるためにいくつかのアクションを実行する必要がある場合があります。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 での Linux および Unix インスタンスの保護 EC2

(コンソール、、、または EC2EC2EC2API AWS CLI の EC2) を使用してエクスポートされたスナップショットEC2から Amazon で Linux または Unix インスタンスを作成する場合EC2、新しいEC2インスタンスには Lightsail SDKs サービスの残余SSHキーが含まれている可能性があります。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 から作成された Amazon EC2 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 での Windows Server インスタンスの保護 EC2

エクスポートされたスナップショットEC2から Amazon で Windows Server インスタンスを作成すると、Lightsail と EC2 にアクセスできる AWS アカウント内のユーザーは、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でもあります。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まだ変更していない場合は、変更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 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 Amazon にエクスポートする EC2

Amazon Lightsail インスタンスとブロックストレージディスクスナップショットを Amazon Elastic Compute Cloud (Amazon) にエクスポートできますEC2。Lightsail インスタンス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) と Amazon Elastic Block Store (Amazon EBS) スナップショットが Amazon に作成されますEC2。これは、Lightsail インスタンスがイメージとシステムディスクで構成されているが、どちらも Lightsail コンソールで単一のインスタンスエンティティとしてグループ化されているためです。スナップショットの作成時にソース 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、アタッチされたディスクごとに追加のEBSスナップショットが Amazon に作成されますEC2。

Lightsail ブロックストレージディスクスナップショットをエクスポートすると、Amazon で 1 つのEBSスナップショットが作成されますEC2。Amazon でエクスポートされたすべてのリソースEC2には、Lightsail に対応するものとは異なる独自の一意の識別子があります。

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されたスナップショットが Amazon で使用可能になった後の次のステップ EC2 (AMI、EBSスナップショット、またはその両方) について説明します。

> **Important**
>
> このガイドのステップを完了する前に、Lightsail エクスポートプロセスに慣れておくことをお勧めします。詳細については、「Amazon へのスナップショットのエクスポートEC2」を参照してください。

目次

サービスにリンクされたロールと Lightsail スナップショットをエクスポートするために必要なIAMアクセス許可

Lightsail 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 にエクスポートしますEC2。の詳細についてはSLRs、「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IAMによっては、 で次の追加のアクセス許可を設定する必要がある場合があります。

- 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、 AWS アカウント管理者はユーザーに次のポリシーを追加する必要があります。ユーザーのアクセス許可を変更する方法の詳細については、 IAMドキュメントのIAM「ユーザーのアクセス許可の変更」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 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EC2。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

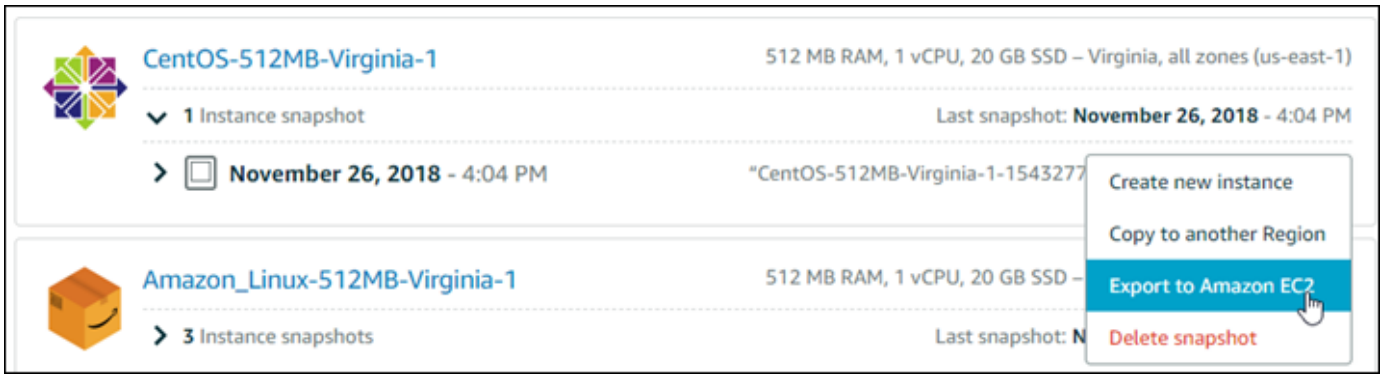
Lightsail スナップショットを Amazon にエクスポートする EC2

スナップショットを Amazon 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EC2です。Lightsail 、 AWS Command Line Interface （ AWS CLI ）API、または を使用してスナップショットをエクスポートすることもできますSDKs。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 Lightsail AWS リージョン から Amazon に同じ にエクスポートされますEC2。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 の別のリージョンにスナップショットをコピーしてから、エクスポートを実行します。詳細については、「ある 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。

Lightsail スナップショットを Amazon にエクスポートするには EC2

1. Lightsail 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でスナップショットを選択します。
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 にエクスポートを選択しますEC2。



> **Note**
>
> cPanel & WHM (CentOS 7) インスタンスのスナップショットを Amazon にエクスポートすることはできませんEC2。

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 へのエクスポートに同意する場合はEC2、はい、プロセスを続行します。

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。Lightsail コンソールのエクスポートセクションを使用して、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、「Lightsail でスナップショットのエクスポートステータスを追跡する」を参照してください。

タスクのステータスを追跡する

Lightsail コンソールのエクスポートセクションでエクスポートのステータスを追跡します。Lightsail コンソールのすべてのページの左側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、「Lightsail でスナップショットのエクスポートステータスを追跡する」を参照してください。

エクスポート には、次の情報が表示されます。

- スナップショット名 — ソース Lightsail スナップショットの名前。
- ステータス — エクスポートのステータス。これは、`In progress`、`Successful`、または `Failed` です。
- 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。
- ソースの詳細 — メモリ、処理、ストレージなど、ソース Lightsail インスタンスの仕様。
- ソースインスタンス名 — スナップショットのソースインスタンスの名前。
- 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。
- 作成されたスナップショット — ソース Lightsail スナップショットが作成された日時。

完了したエクスポートのタスク履歴セクションには、次の情報が表示されます。

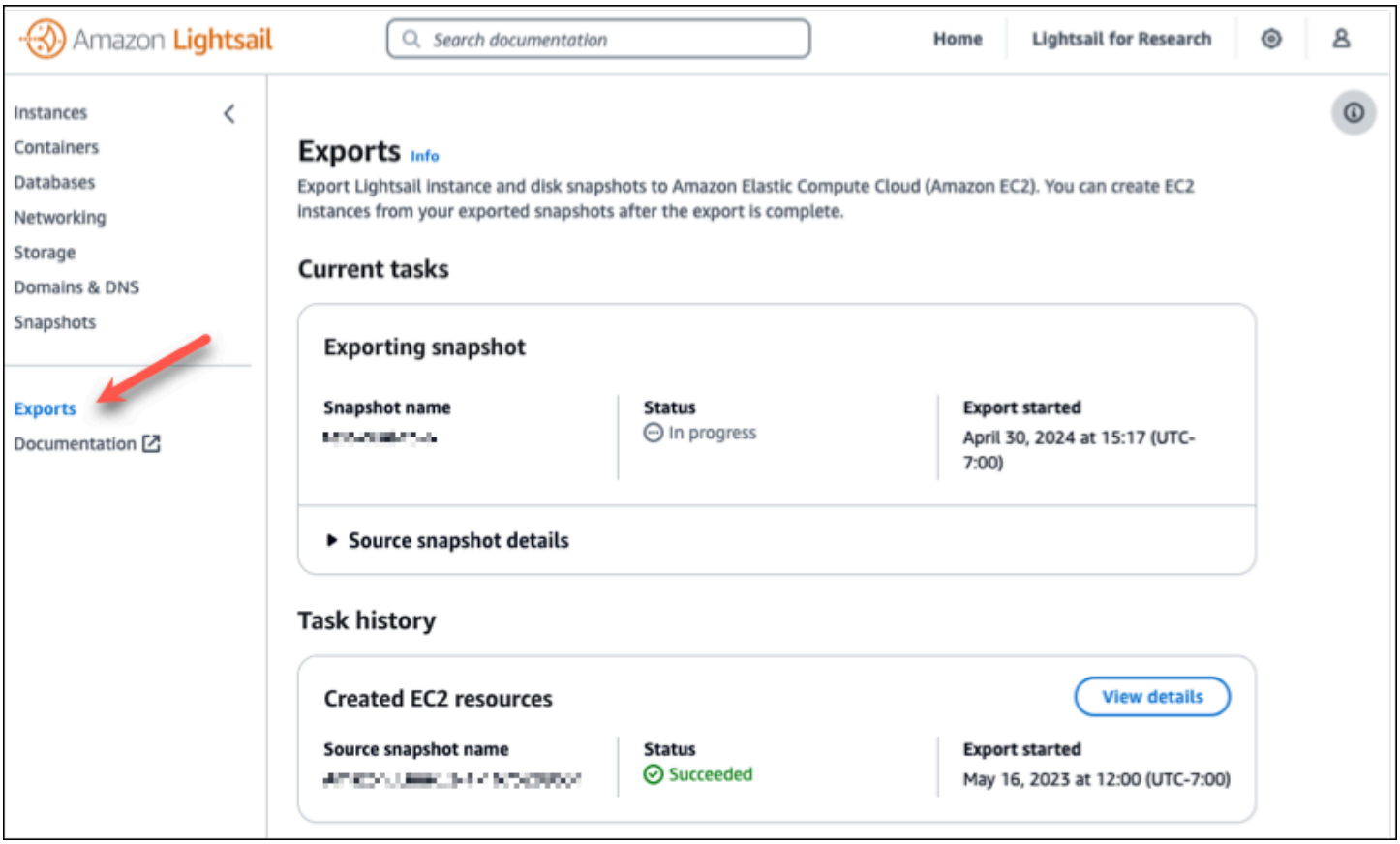
- でインスタンスを作成する EC2 — Lightsail コンソールEC2を使用して Amazon で新しいインスタンスを作成するには、このオプションを選択します。詳細については、「エクスポートされたスナップショットから Amazon EC2インスタンスを作成する」を参照してください。
- 開く EC2 — Amazon EC2コンソールを使用してエクスポートしたスナップショットから新しいEC2リソースを作成するには、このオプションを選択します。Lightsail ブロックストレージディスクスナップショットをエクスポートした場合は、Amazon EC2 を使用してスナップショット (EBS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**
>
> 不要になったソース Lightsail スナップショットを削除します。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。

Lightsail でスナップショットのエクスポートステータスを追跡する

Amazon Lightsail コンソールのエクスポートセクションでは、Lightsail スナップショットを Amazon EC2 にエクスポートしたり、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新しい EC2 インスタンスを作成したりするステータスを追跡できます。エクスポートタスクは、ソースインスタンスまたはブロックストレージディスクのサイズと設定によっては、時間がかかる場合があります。エ

クスポートには、Lightsail コンソールのすべてのページの左側のナビゲーションペインからアクセスできます。



Lightsail スナップショットを Amazon EC2 にエクスポートする方法、またはエクスポートされたスナップショットから EC2 インスタンスを作成する方法の詳細については、次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

エクスポートされた 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 インスタンススナップショットがエクスポートされAmazon EC2 で (AMI および EBS スナップショットとして) 使用可能になったら、Amazon Lightsail コンソールの Amazon EC2 インスタンスの作成ページを使用して、スナップショットから Amazon EC2 インスタンスを作成できます。これは、Amazon EC2 へのアップグレードウィザードとも呼ばれます。 Amazon EC2 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は、新しい EC2 インスタンスとその関連リソースを作成するプロセスを簡素化します。

> **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。

Lightsail API、 AWS CLI、または SDK を使用して新しい EC2 インスタンスを作成することもできます。 SDKs 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 に慣れている場合は、EC2 コンソール、Amazon EC2 API AWS CLI、または SDKs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**
>
> このガイドのステップを完了する前に、Lightsail エクスポートプロセスに慣れておく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

AWS CloudFormation Lightsail の スタック

Lightsail は、 AWS CloudFormation スタックを使用して EC2 インスタンスとその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、「AWS CloudFormation Stacks for Lightsail」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。ルートユーザーには、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が既にあります。
- IAM ユーザーが EC2 インスタンスを作成する場合、 AWS アカウント管理者はユーザーに次のアクセス許可を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、次のアクセス許可が必要です。

    > **Note**
    >
    > これら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。この場合、 AWS CloudFormation コンソールに移動し、EC2 リソースを手動で削除できます。詳細については、AWS CloudFormation「Stacks for Lightsail」を参照してください。

    - ec2:DescribeAvailabilityZones
    - ec2:DescribeSubnets
    - ec2:DescribeRouteTables
    - ec2:DescribeInternetGateways
    - ec2:DescribeVpcs
    - クラウドフォーメーション：CreateStack
    - クラウドフォーメーション：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

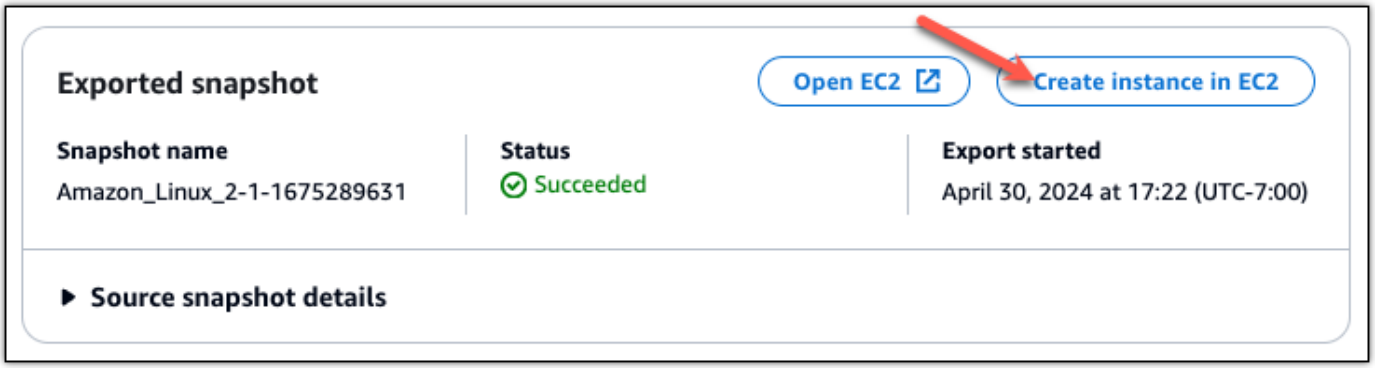
Lightsail インスタンススナップショットを Amazon EC2 にエクスポートします。詳細については、「スナップショットを Amazon EC2 にエクスポートする」を参照してください。

Lightsail コンソールで 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には、インスタンス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。



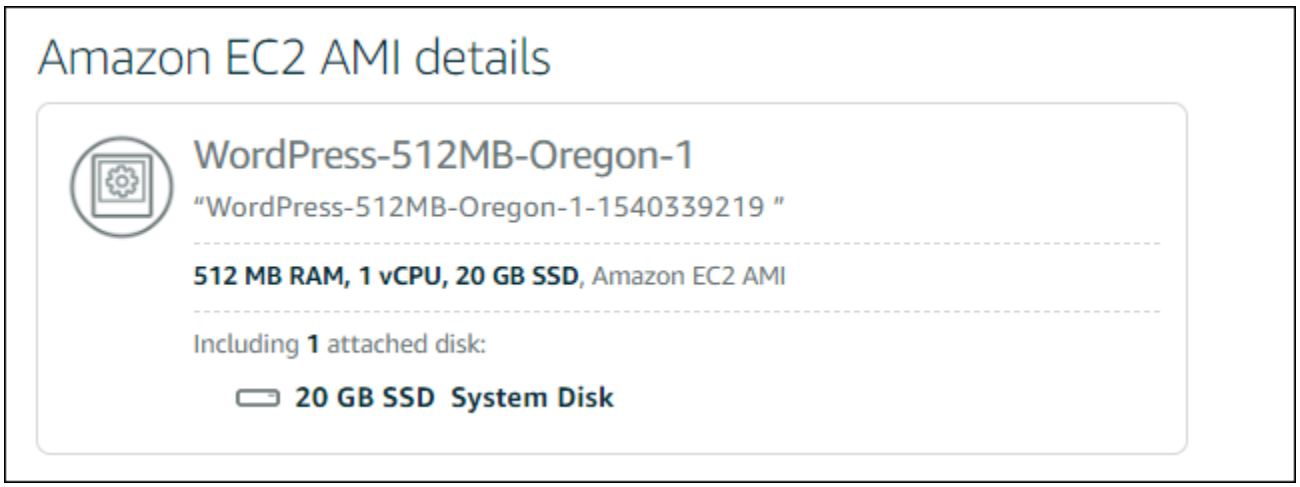
[Amazon EC2 インスタンスの作成] ページが表示されます。このガイドの次の「Amazon EC2 インスタンスの作成」セクションに進み、このページを使用して EC2 インスタンスを設定して作成する方法を確認します。

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、次のステップを複数回繰り返しますが、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。

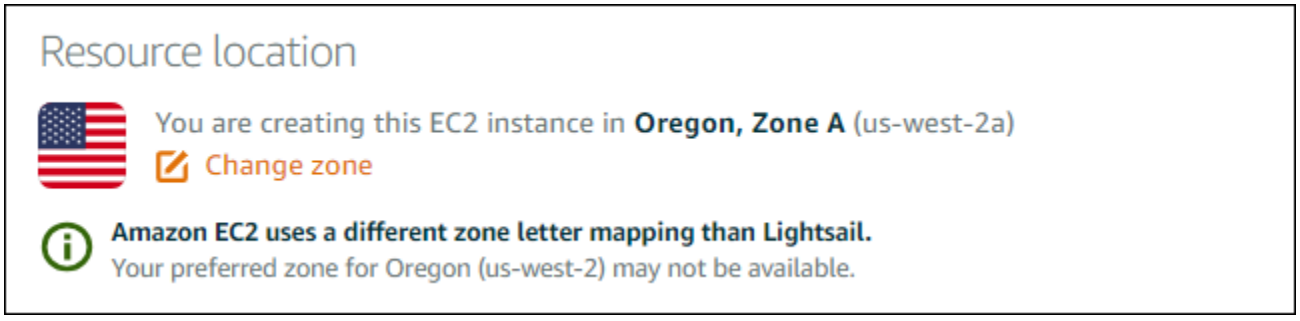
Amazon EC2 インスタンスを作成するには

1. ページの Amazon EC2 AMI の詳細セクションで、表示される Amazon マシンイメージ (AMI) の詳細がソース 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。

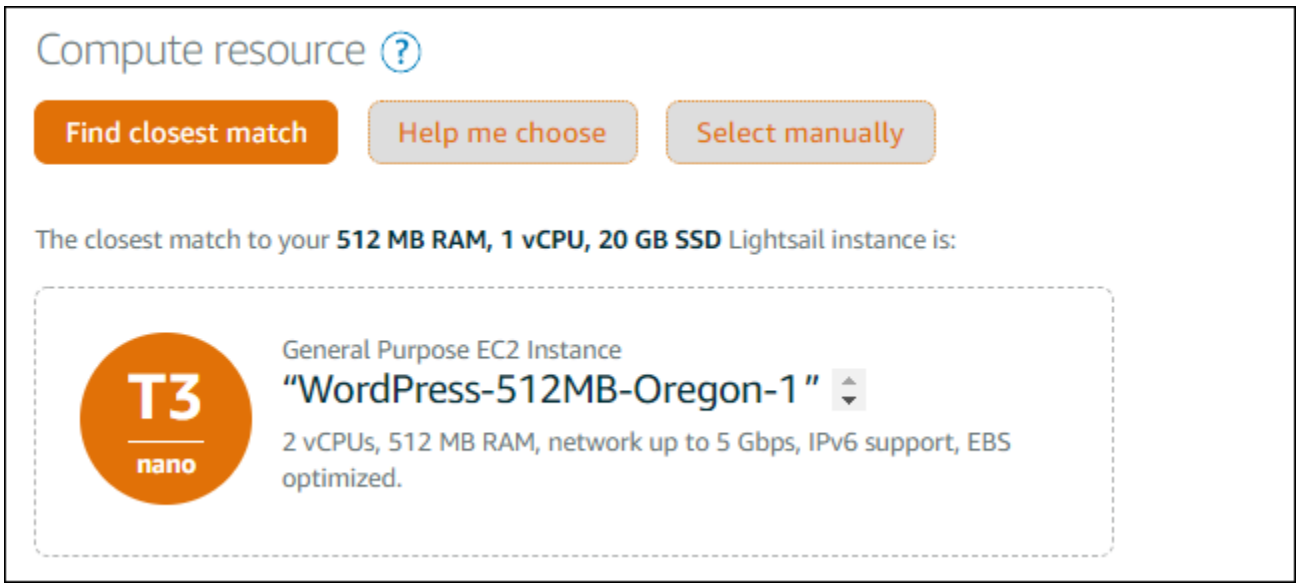


2. ページの [Resource location (リソースの場所)] セクションで、必要に応じてインスタンスのアベイラビリティーゾーンを変更します。Amazon EC2 リソースは、ソース Lightsail スナップショット AWS リージョン と同じ に作成されます。

> **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。



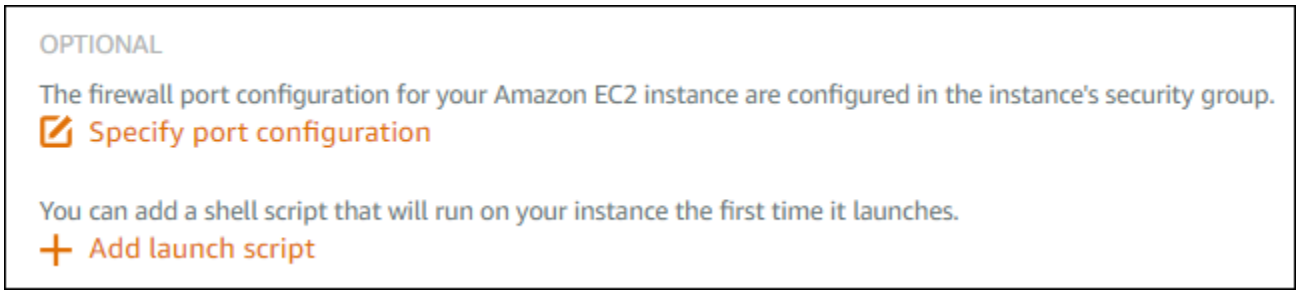
a. 最も近い一致を検索して、ソース Lightsail インスタンスの仕様に密接に一致する Amazon EC2 インスタンスタイプを自動的に選択します。

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。

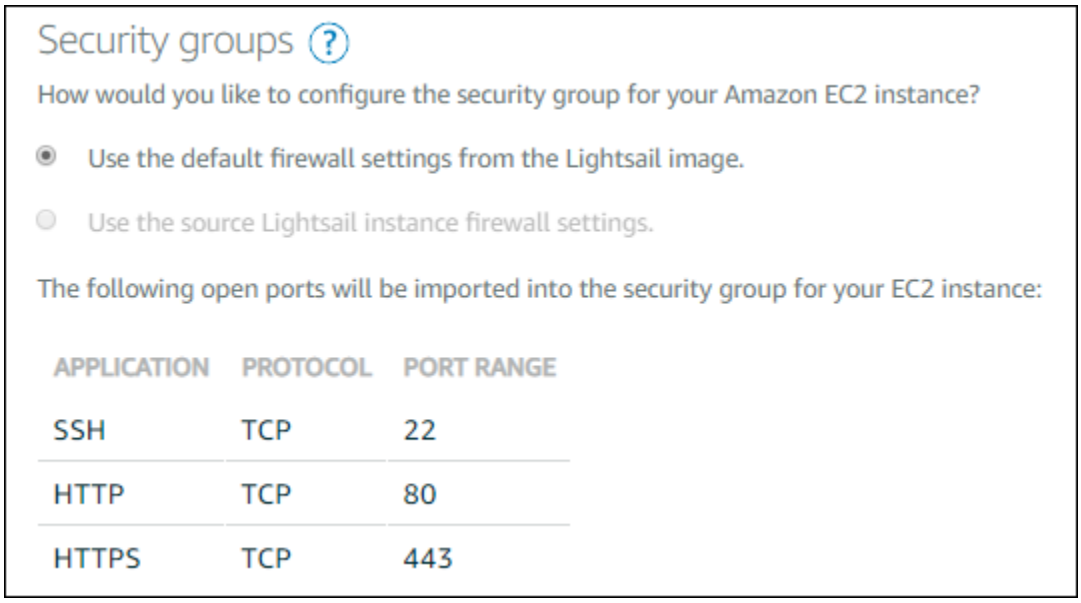
c. [手動で選択] では、[Amazon EC2 インスタンスの作成] ページから利用可能なインスタンスタイプが一覧表示されます。

> **Note**
>
> 一部の Lightsail インスタンスは、拡張ネットワーキングが有効になっていないため、現行世代の EC2 M5, C5、または R5) と互換性がありません。ソース 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、M4, C4、または R4) を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 インスタンスの作成」ページに表示されます。
>
> ソース 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、前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、ネットワークドライバーを更新してから、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アップグレードする必要があります。 M4, C4 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て、新しい EC2 インスタンスのソース Lightsail ブループリント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 Lightsail インスタンスのファイアウォール設定を使用して、新しい EC2 インスタンスのソース Lightsail インスタンスからのポートを設定します。このオプションは、ソース Lightsail インスタンスがまだ実行中の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「接続セキュリティ」セクションで、ソース Lightsail インスタンスへの接続方法を確認します。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には、一意のデフォルトの Lightsail キーをダウンロード AWS リージョン して使用します。

> **Note**
>
> デフォルトの Lightsail キーペアは、Lightsail の Windows Server インスタンスで常に使用されます。

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**
>
> Lightsail は、個人のプライベートキーを保存しません。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、作成される EBS ボリュームが、ソース Lightsail インスタンスのシステムディスクおよびアタッチされた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 の外部でのリソース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。

   Lightsail は、インスタンスが作成されていること、および AWS CloudFormation スタックに関する情報が表示されることを確認します。Lightsail は CloudFormation スタックを使用して EC2 インスタンスとその関連リソースを作成します。詳細については、AWS CloudFormation「Lightsail のスタック」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 コンソールのエクスポートセクションを使用して、EC2 インスタンスのステータスを追跡します。詳細については、「Lightsail でスナップショットのエクスポートステータスを追跡する」を参照してください。

作成される EC2 インスタンスには、次の情報が表示されます。

- ソース名 — ソース 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。
- 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。

エクスポートされた Lightsail ディスクスナップショットから Amazon Elastic Block Store ボリュームを作成する

Lightsail ブロックストレージディスクスナップショットがエクスポートされAmazon EC2 で (EBS スナップショットとして) 使用可能になったら、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 でエクスポートされ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 API、 AWS CLI、または SDKsボリュームを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**
>
> このガイドのステップを完了する前に、Lightsail のエクスポートプロセスに慣れておく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

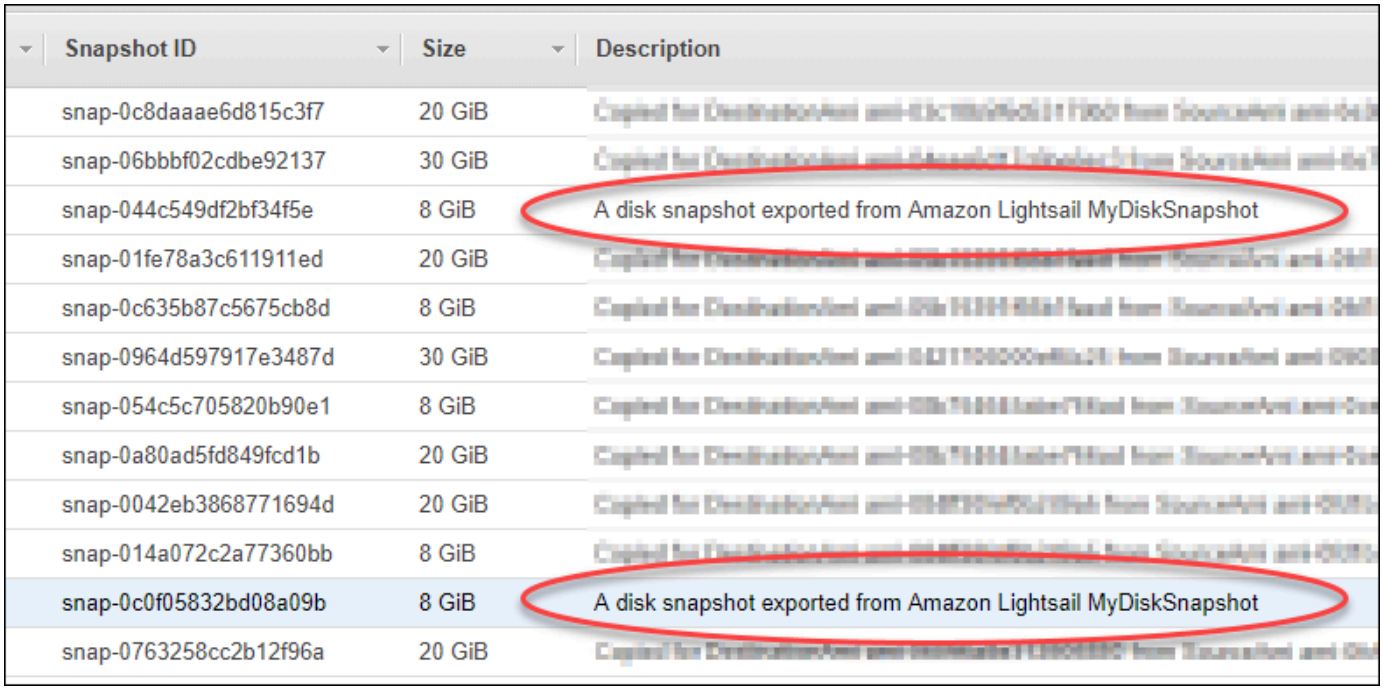
Amazon EC2 コンソールを使用して、エクスポートされた Lightsail ブロックストレージディスクスナップショットから新しい EBS ボリュームを作成します。

> **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。

   エクスポートされ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 Amazon Lightsail の説明からエクスポートされたディスクスナップショットによって識別できます。



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。

Lightsail スナップショットから作成された Linux Amazon EC2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SSHと同様に、

を介してインスタンスに接続できます。インスタンスに対して認証するには、ソースインスタンスののデフォルトの Lightsail キーペア AWS リージョン、または独自のキーペアを使用します。このガイドでは、Pu EC2を使用して で Linux または Unix インスタンスに接続する方法を示しますTTY。

> **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

インスタンスのキーを取得する

新しい Amazon EC2インスタンスへの接続に必要な正しいキーを取得します。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にデフォルトの Lightsail キーペアを使用する — Lightsail アカウントページのキータブからデフォルトのプライベートSSH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EC2作成された Amazon 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。

- 独自のキーペアの使用 — プライベートキーを見つけて、それを使用して Amazon EC2インスタンスに接続します。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紛失した場合は、Amazon EC2インスタンスに接続できません。

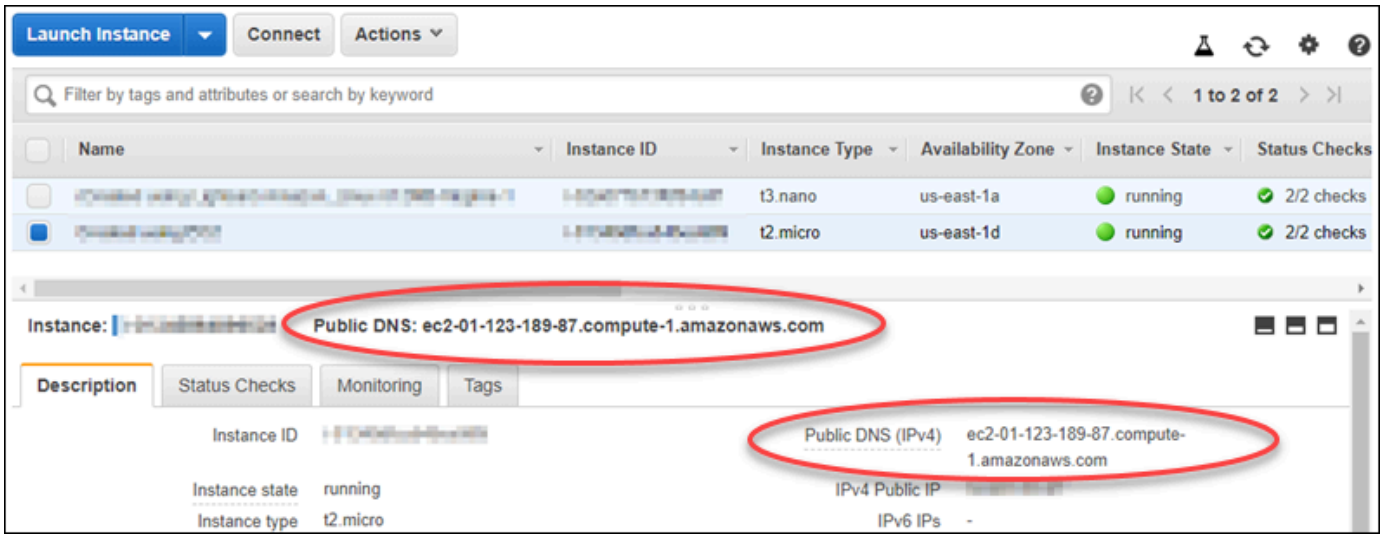
インスタンスのパブリックDNSアドレスを取得する

Amazon EC2インスタンスのパブリックDNSアドレスを取得して、Pu などのSSHクライアントを設定してインスタンスTTYに接続できるようにします。

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。

   これは、インスタンスに接続するように SSHクライアントを設定するときに使用するアドレスです。このガイドの「Pu のダウンロードとインストールTTY」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する方法について説明します。



Pu をダウンロードしてインストールするTTY

PuTTY は Windows 用の無料SSHクライアントです。Pu の詳細についてはTTY、「Pu TTY: 無料SSHおよび Telnet クライアント」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。既に Pu をお持ちの場合はTTY、このガイドの「P でキーuTTYgenを設定する」セクションに進んでください。

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については、PuTTY ドキュメント を参照してください。

このガイドの「P でキーuTTYgenを設定する」セクションに進み、P でキーを設定しますuTTYgen。

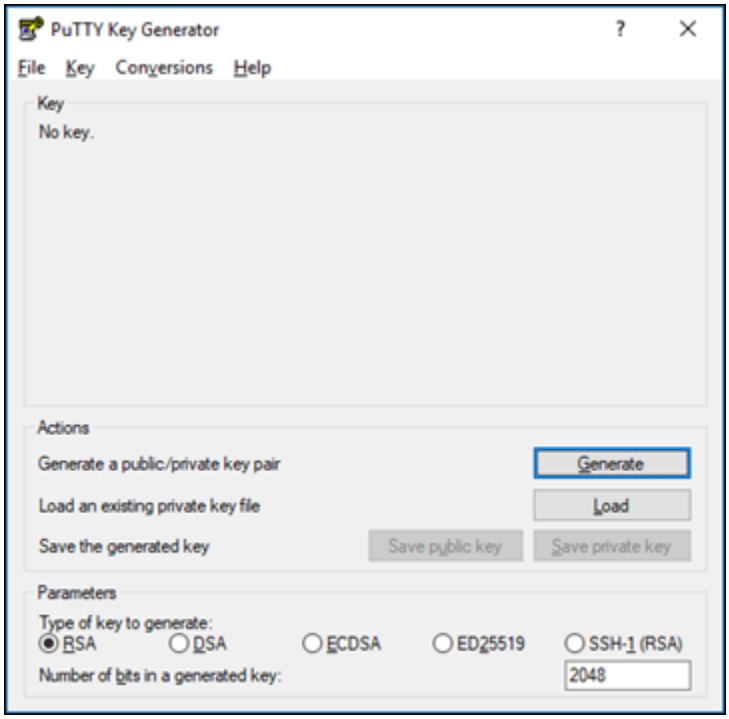
P でキーを設定するuTTYgen

P は、Pu で使用するパブリックキーとプライベートキーのペアuTTYgen を生成しますTTY。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。

P でキーを設定するにはuTTYgen

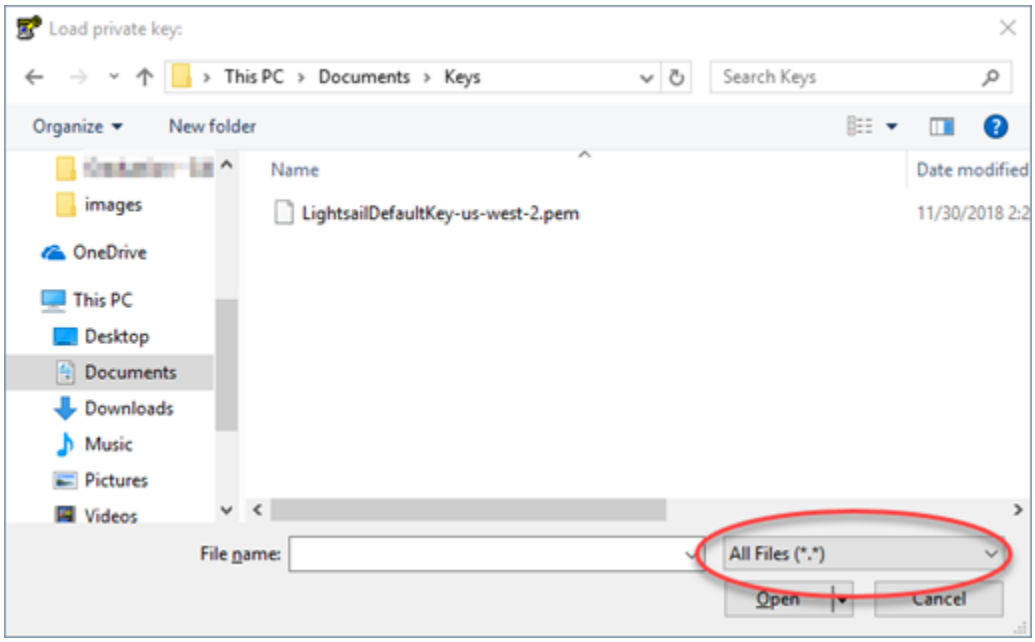
1. P を起動しますuTTYgen。

   例えば、Windows のスタートメニューを選択し、すべてのプログラム を選択し、Pu TTYを選択し、P uTTYgenを選択します。

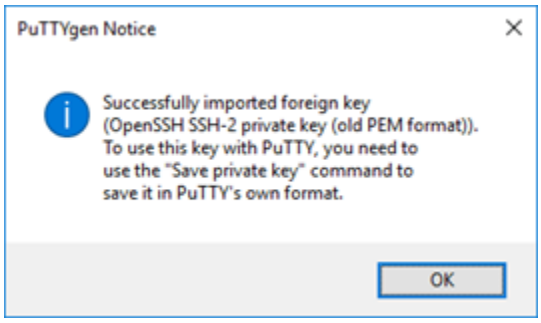


2. ロード] を選択します。

デフォルトでは、P は拡張子 PPKのファイルのみuTTYgen を表示します。.PEM ファイルを検索するには、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。

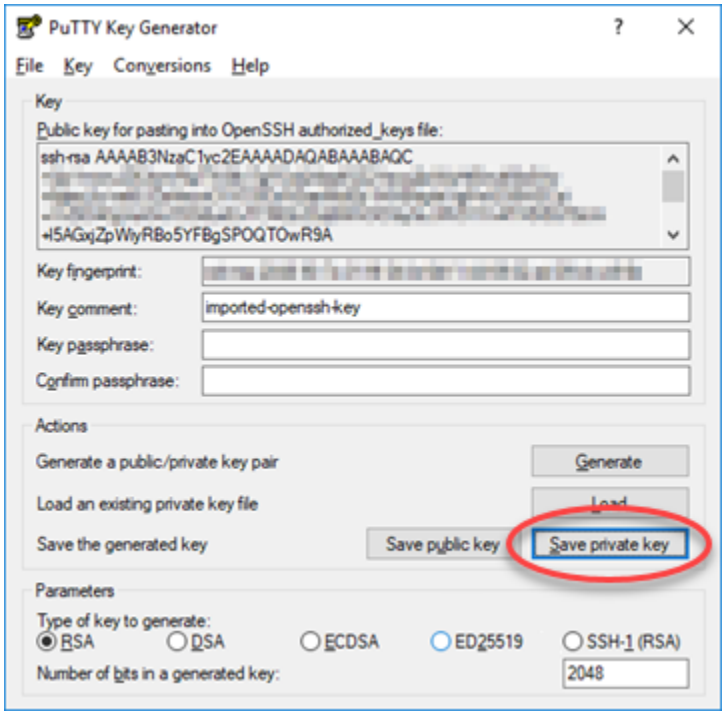


3. このガイドの前半でダウンロードしたデフォルトの Lightsail キーファイル (.PEM) を選択し、「 を開く」を選択します。
4. P がキーが正常にインポートされたuTTYgen ことを確認したら、OK を選択します。



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場合は、Pu 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TTY。



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。

   P は新しいキーファイルを PPK. ファイルタイプとしてuTTYgen 保存します。

7. P を閉じますuTTYgen。

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

インスタンスに接続するように PuTTY を設定する

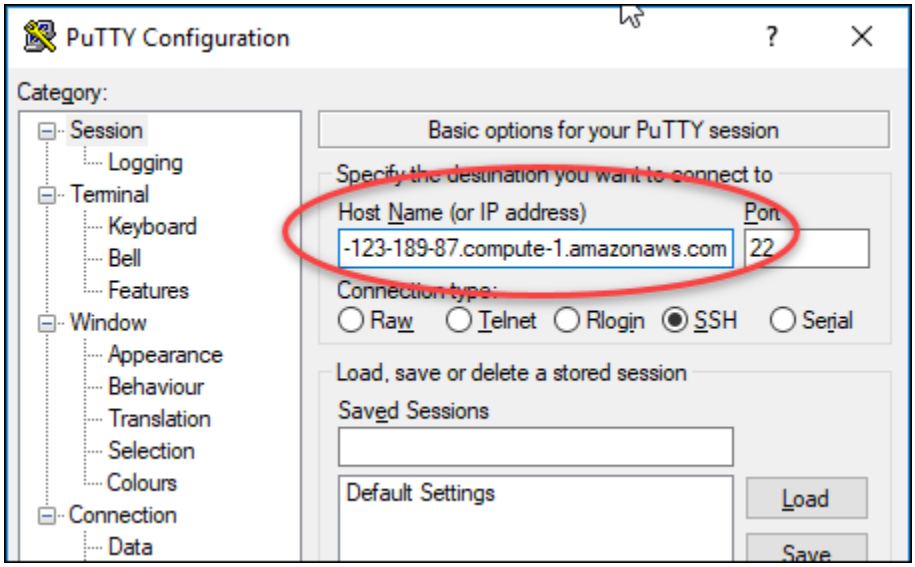
を使用して Linux または Unix インスタンスに接続するための要件がすべて揃ったのでTTY、Pu を設定しますSSH。

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

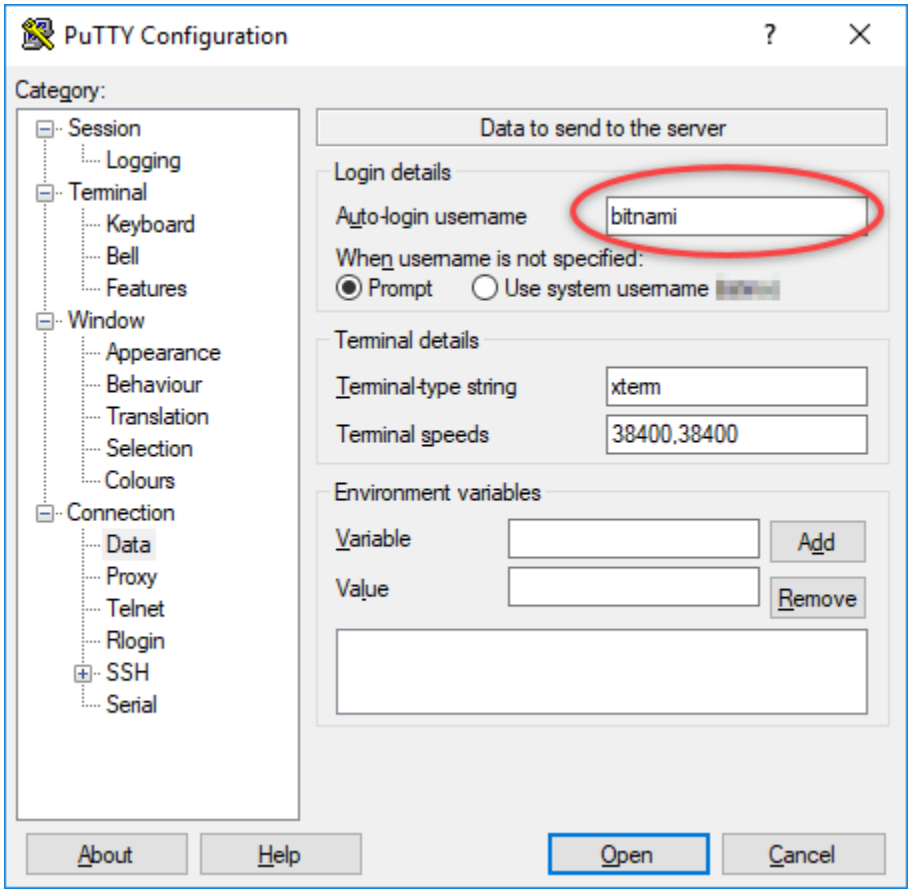
1. Pu を開きますTTY。

   例えば、Windows のスタートメニューを選択し、すべてのプログラム を選択し、Pu TTYを選択し、Pu TTYを選択します。

2. ホスト名テキストボックスに、このガイドの前半で 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。



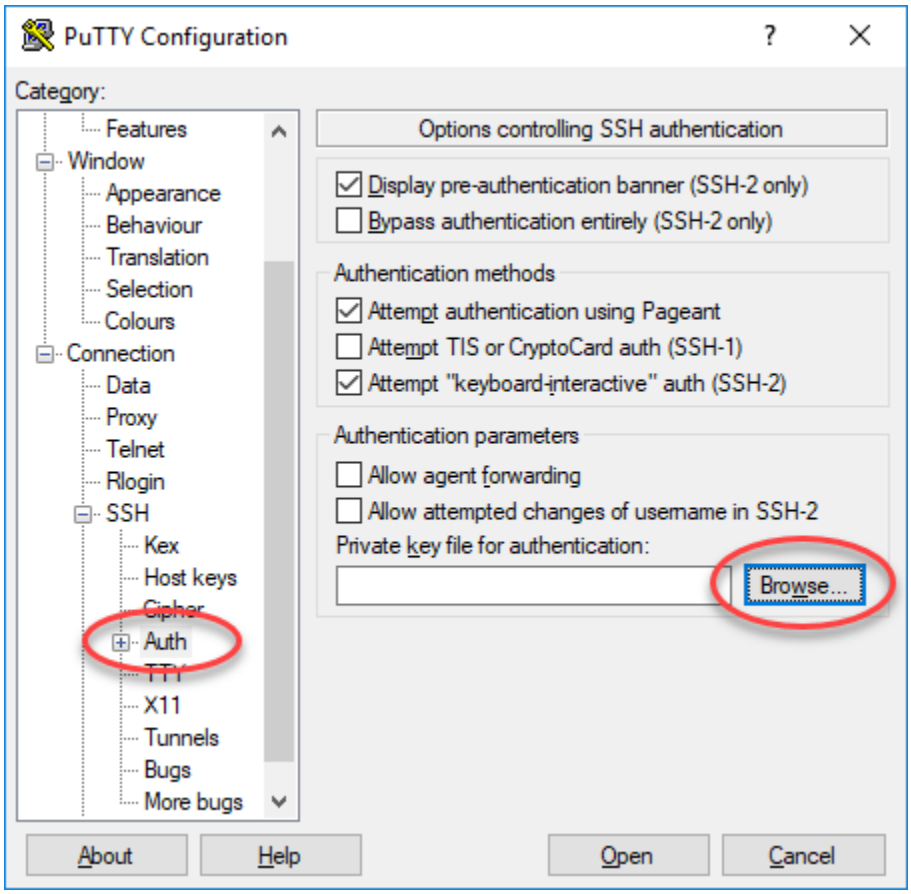
3. 左のナビゲーションペインの [Connection (接続)] セクションで、[Data (データ)] を選択します。
4. [Auto-login username (自動ログインのユーザー名)] テキストボックスに、インスタンスにログインするときに使用するユーザー名を入力します。



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ユーザー名を入力します。

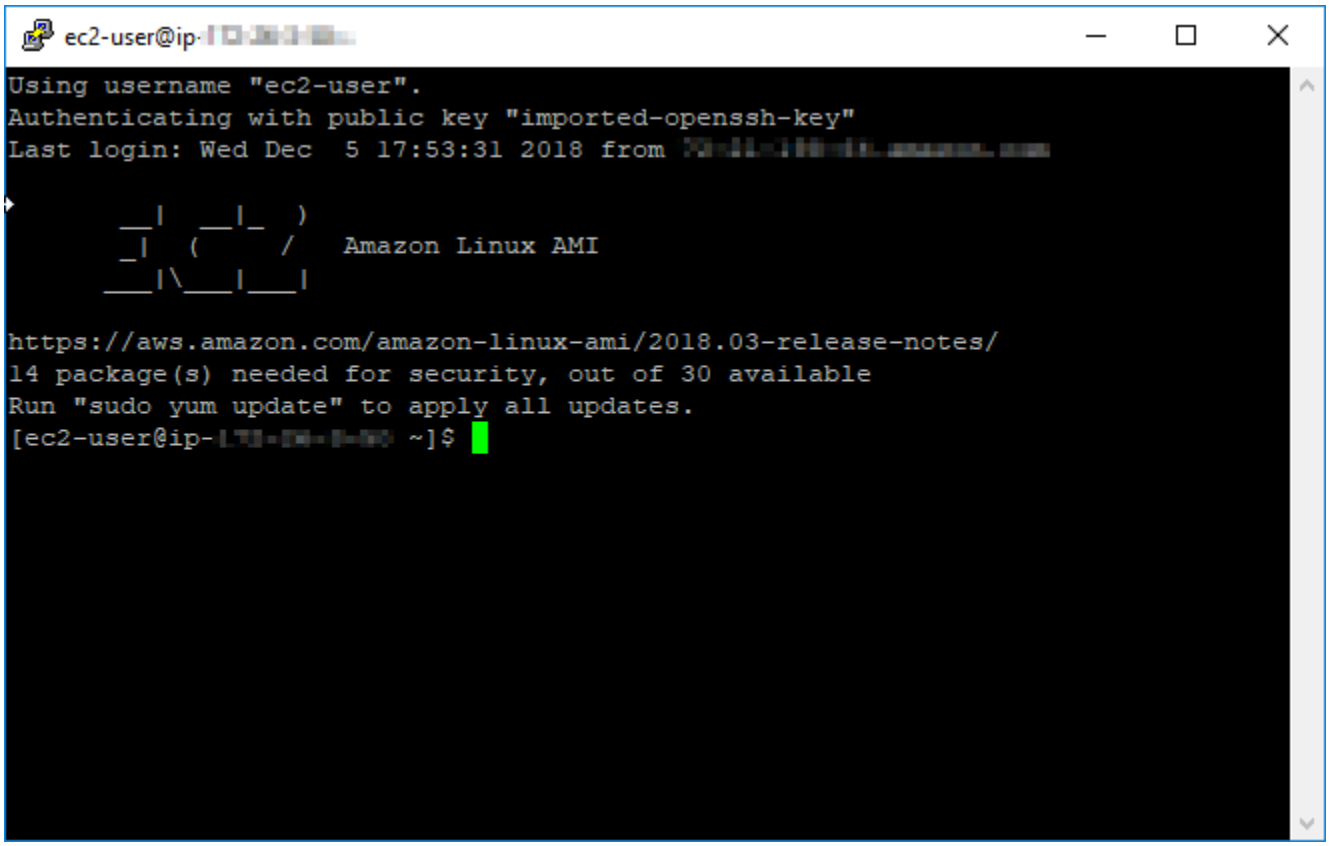
- AlmaLinux、Amazon Linux 2、Amazon Linux 2023、CentOS Stream 9、無料 BSD、オープン SUSEインスタンス: `ec2-user`
- Debian インスタンス: `admin`
- Ubuntu インスタンス: `ubuntu`
- Bitnami インスタンス: `bitnami`
- Plesk インスタンス: `ubuntu`
- cPanel および WHM インスタンス: `centos`

5. 左側のナビゲーションペインの接続セクションで、 を展開しSSH、認証 を選択します。
6. 参照 を選択して、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、 を開く 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 を使用してエクスポートされ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、Amazon の新しい Linux または Unix インスタンスEC2には Lightsail サービスの残差キーEC2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EC2作成された Amazon 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起動された安全な Amazon EC2 インスタンス

Amazon Lightsail 、および Amazon Elastic Compute Cloud (Amazon EC2) は、パブリックキー暗号化を使用してログイン情報を暗号化および復号します。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 Lightsail インスタンスを EC2 にエクスポートすると、新しい EC2 インスタンスには Lightsail サービスの残余キーが含まれます。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された 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次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への接続に使用した場合は、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するか、Lightsail コンソールでインスタンスのキーを作成した場合、Lightsail のデフォルトキーは Amazon EC2 インスタンスに存在しません。
- Lightsail システムキーを削除します。これはキーとも呼ばれます`lightsail_instance_ca.pub`。Linux および Unix インスタンスのこのキーにより、Lightsail ブラウザベースの SSH クライアントが接続できるようになります。Lightsail コンソールの Amazon EC2 インスタンスの作成ページまたは Lightsail API を使用して Amazon EC2 インスタンスが作成されると、`lightsail_instance_ca.pub`キーは自動的に削除されます。

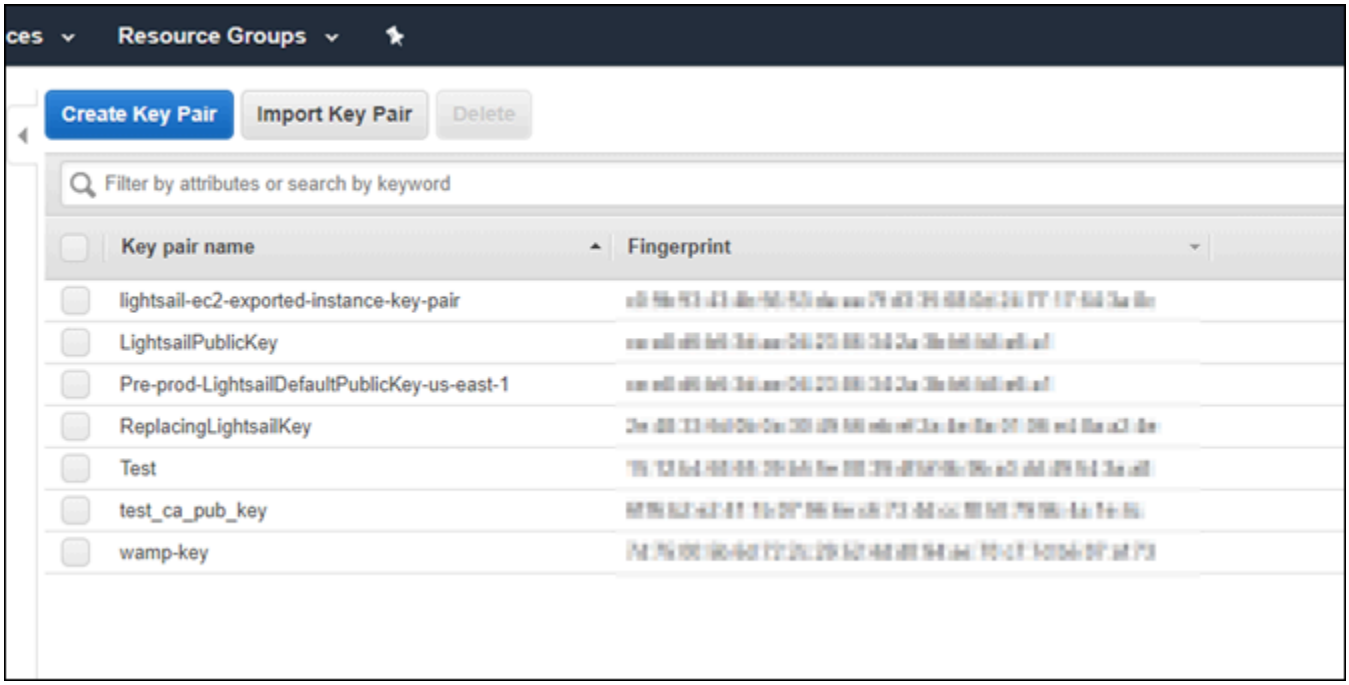
目次

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 のデフォルトキーペアの置き換えに使用できる新しいキーペアを作成します。

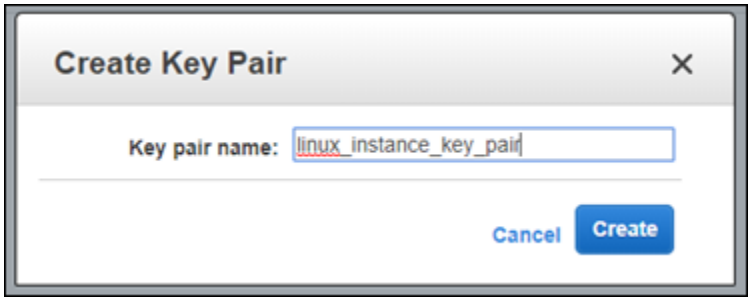
Amazon EC2 を使用してプライベート キーを作成するには

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

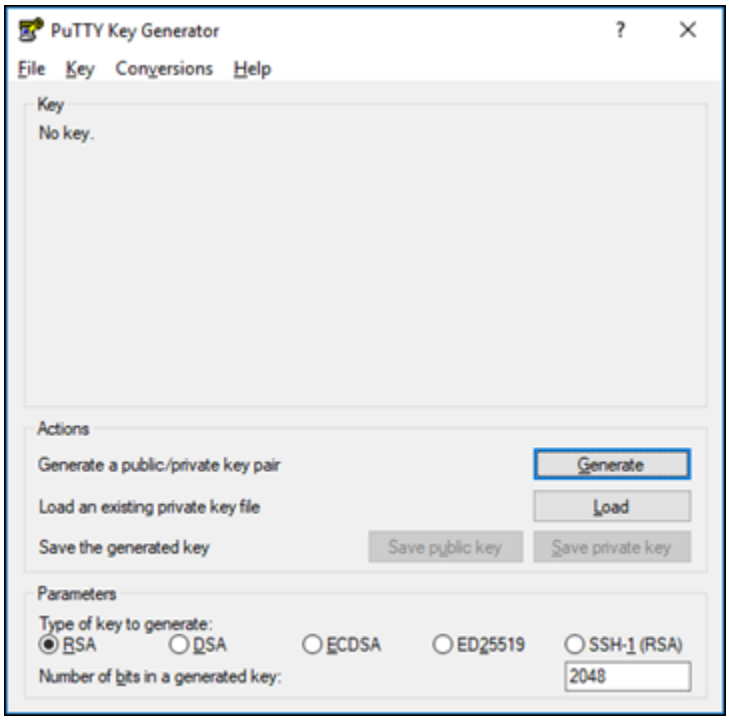
PuTTYgen は PuTTY に含まれているツールです。PuTTYgen では、このガイドで後ほどインスタンスに追加するパブリックキーテキストを生成します。

> **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 から作成された Amazon EC2 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

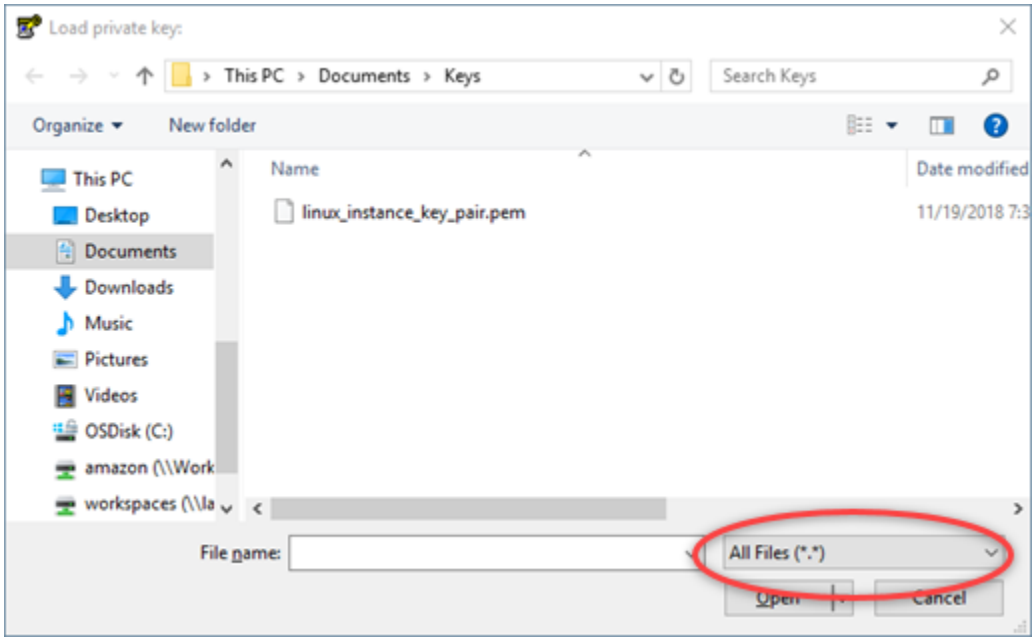
1. PuTTYgen を起動します。

   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。



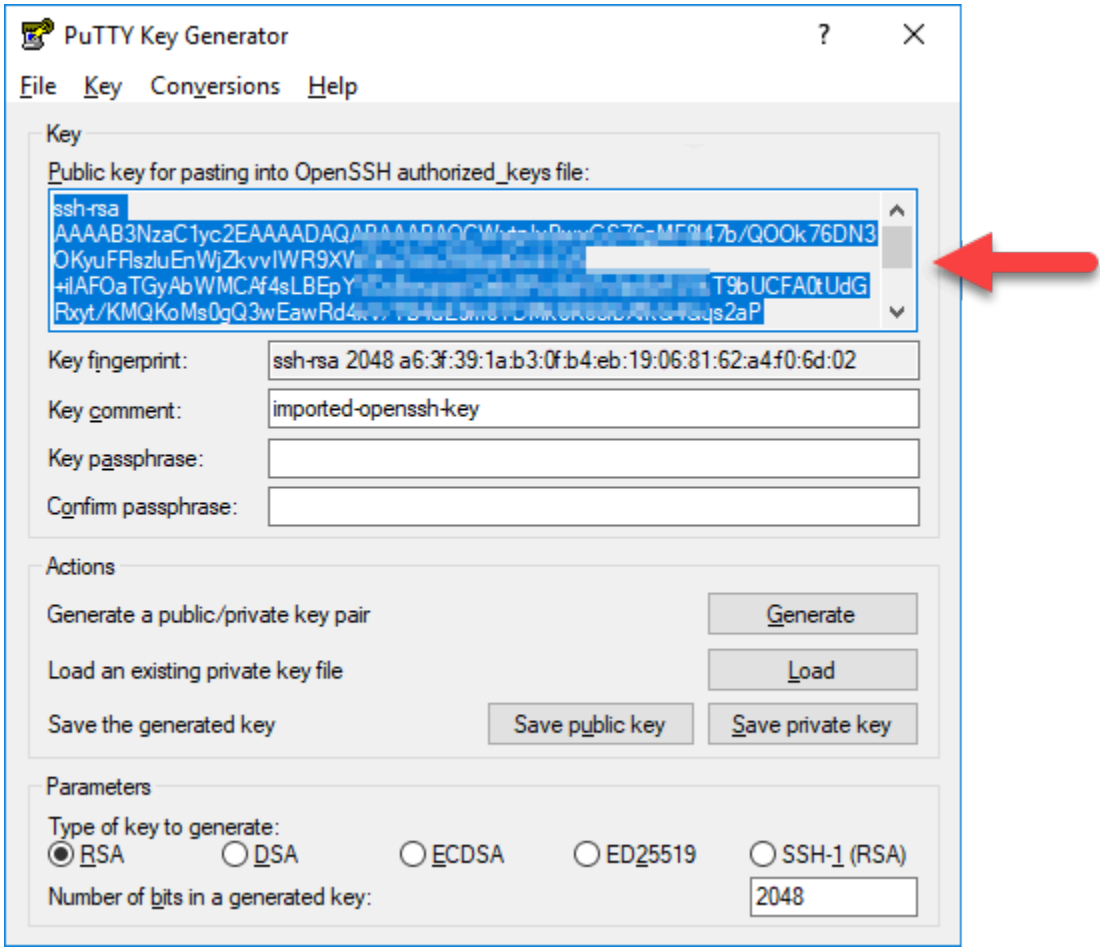
2. ロード] を選択します。

   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 などのテキストエディタを開き TextEdit、Windows を使用している場合は Ctrl +V、macOS を使用している場合は Cmd+V を押して、パブリックキーテキストをそのエディタに貼り付けます。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 の 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 から作成された Amazon EC2 の Linux または Unix インスタンスに接続する Amazon Lightsail」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

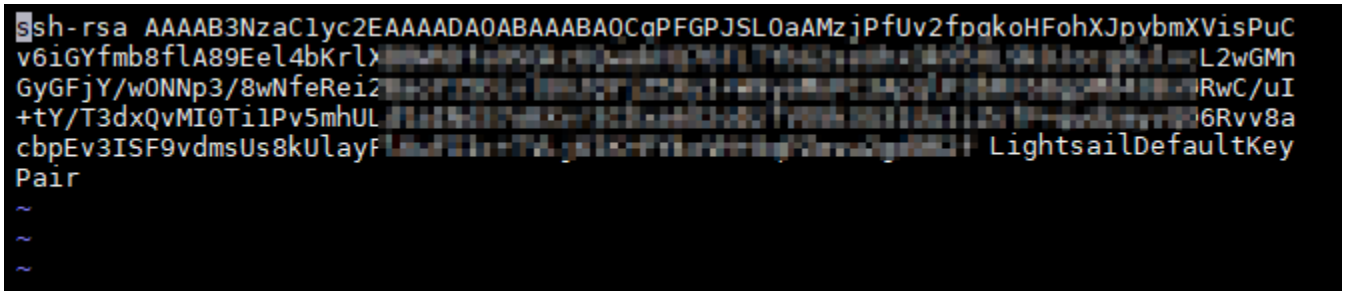
公開キーの内容は、Linux および Unix インスタンスの ~/.ssh/authorized_keys ファイルに保存されています。ファイルを編集して、Amazon EC2 の Linux または Unix インスタンスから Lightsail のデフォルトキーを削除して置き換えます。

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。

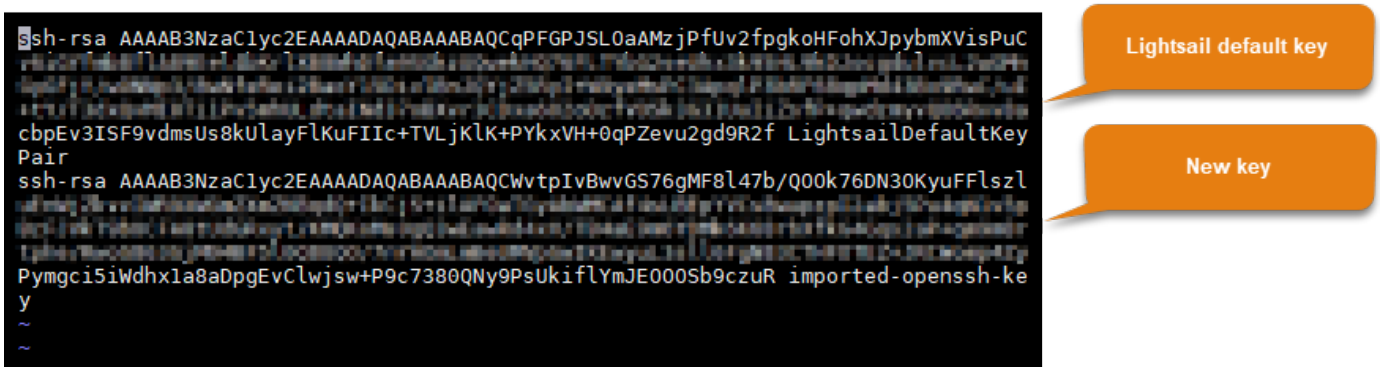
```
sudo vim ~/.ssh/authorized_keys
```

> **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余分な行を入力します。
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。

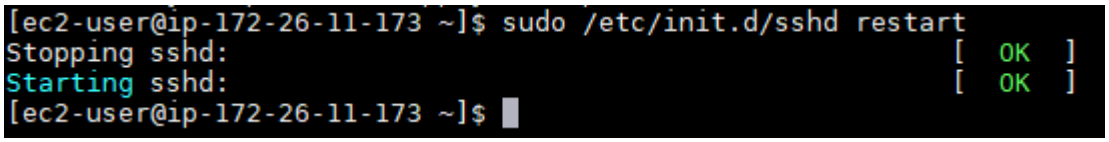
   結果は次のようになります。



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。



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 の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、このガイドの「Lightsail デフォルトキーの削除」セクションに進み、Lightsail 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
```

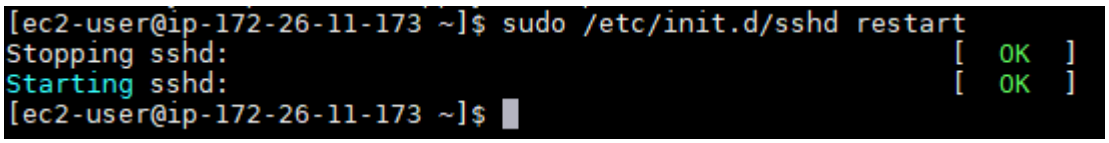
2. I キーを押して Vim エディタを挿入モードにします。

3. `LightsailDefaultKeyPair` で終わる行を削除します。これは 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。



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、インスタンスは Lightsail のデフォルトキーを使用する接続を拒否します。このガイドの「Lightsail システムキーの削除」セクションに進み、Lightsail システムキーを削除します。

Lightsail システムキーを削除する

Linux および Unix インスタンスでは、キーとも呼ばれる Lightsail システム`lightsail_instance_ca.pub`キーにより、Lightsail ブラウザベースの SSH クライアントが接続できるようになります。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 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。

```
[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_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 スナップショットから作成された Windows Server Amazon EC2 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。これは、ソース Amazon Lightsail インスタンスへの接続方法に似ています。ソースインスタンスの のデフォルトの Lightsail キーペアを使用して EC2 インスタンスに接続します AWS リージョン。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 から作成された 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デフォルトの Lightsail 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 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、デフォルトの Lightsail キーペアと Amazon EC2 の Windows Server インスタンスとの関連付けが削除されます。詳細については、「Lightsail スナップショット から作成された Amazon EC2 Windows Server インスタンスを保護する」を参照してください。

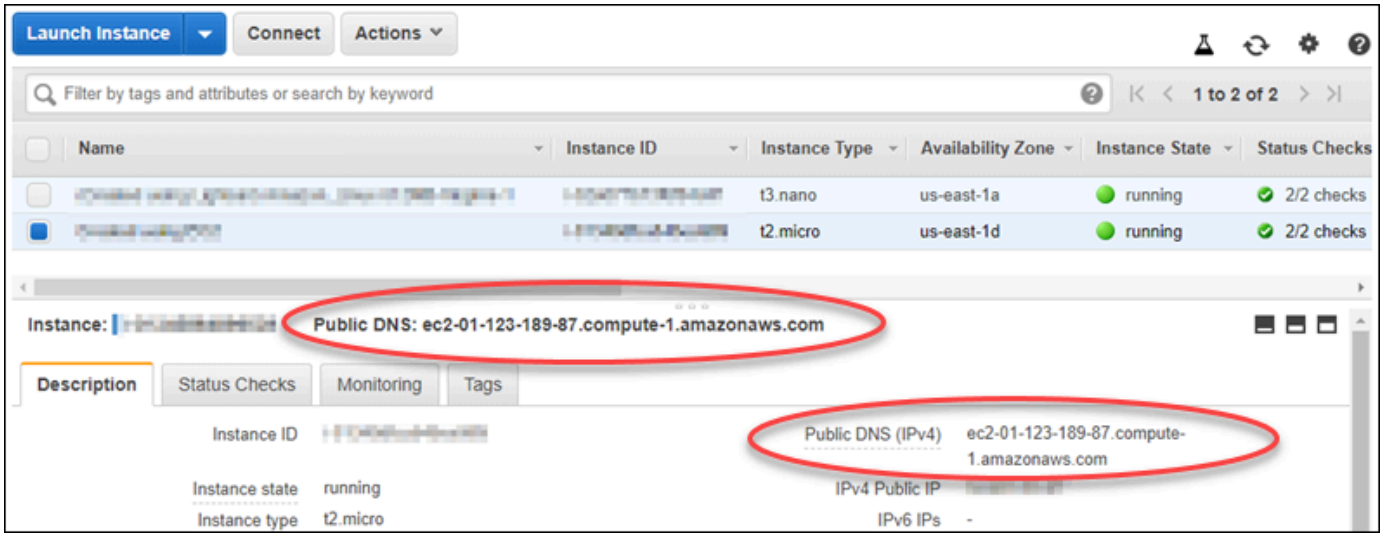
インスタンスのパブリック DNS アドレスを取得する

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。

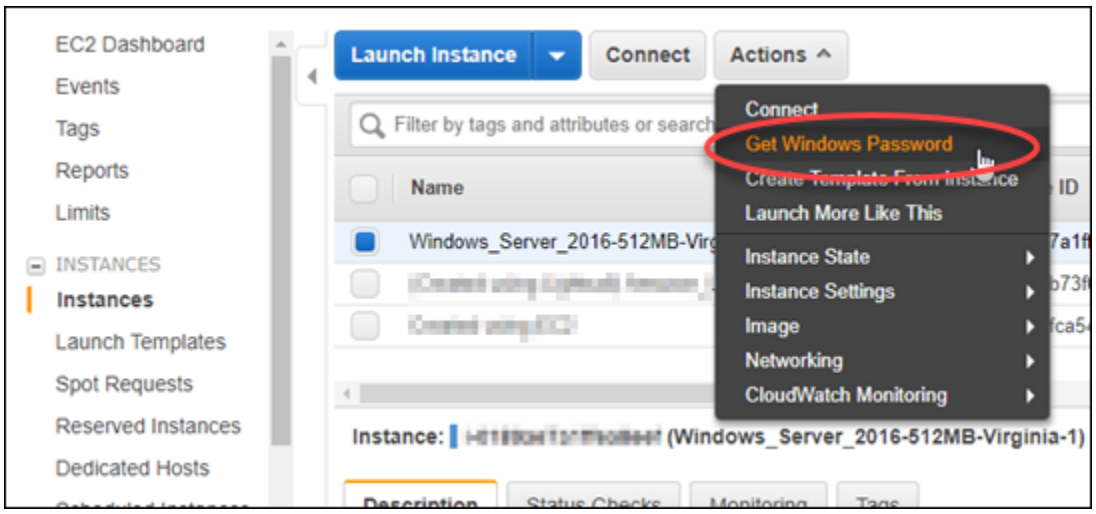


Windows Server インスタンスのパスワードを取得する

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。

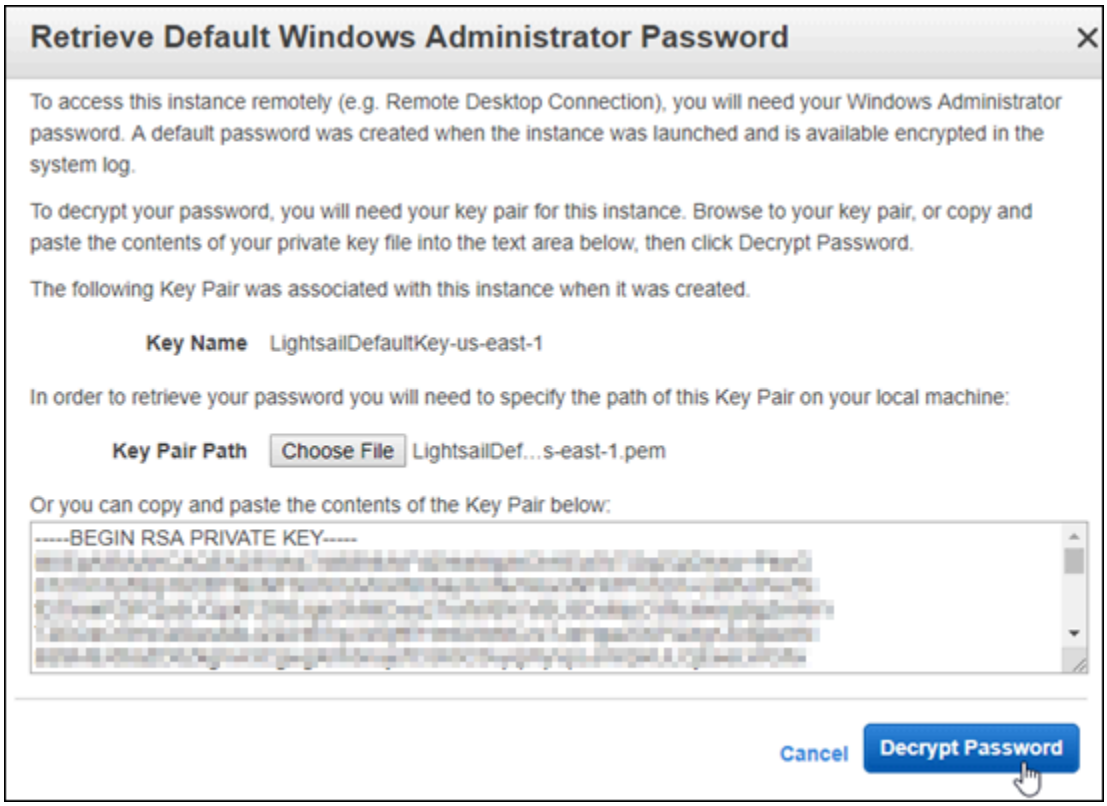
Windows Server インスタンスのパスワードを取得するには

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。

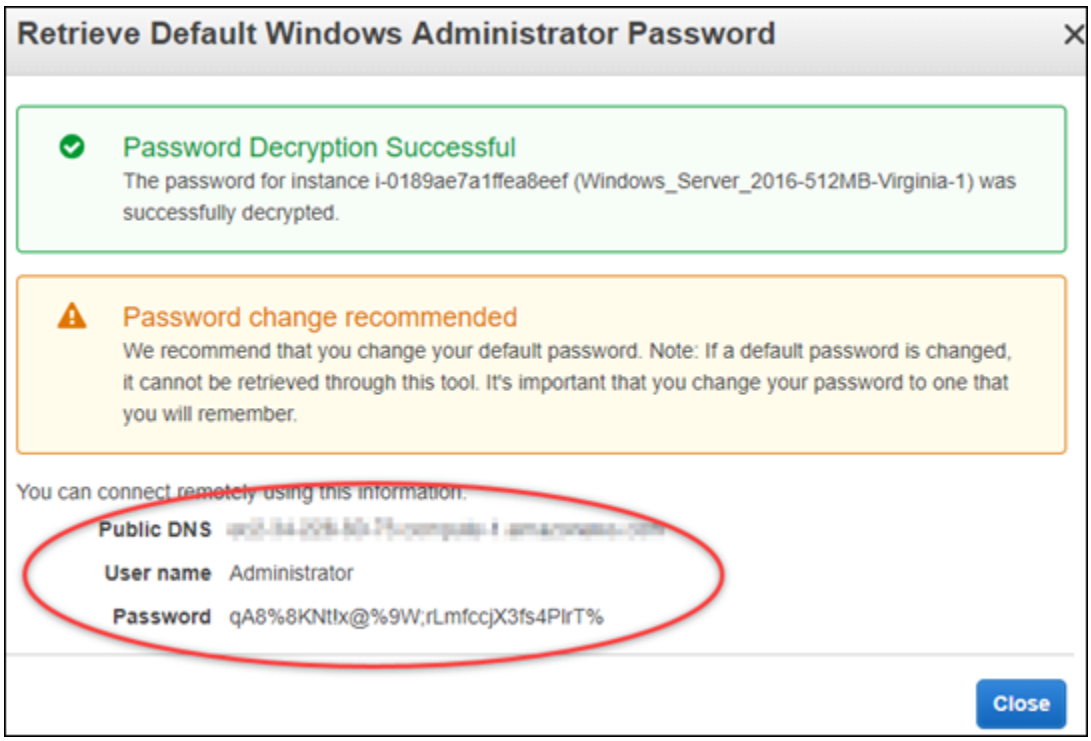


5. プロンプトで、このガイドの前半で Lightsail からダウンロードしたデフォルトのプライベートキーファイルを参照して開きます。

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

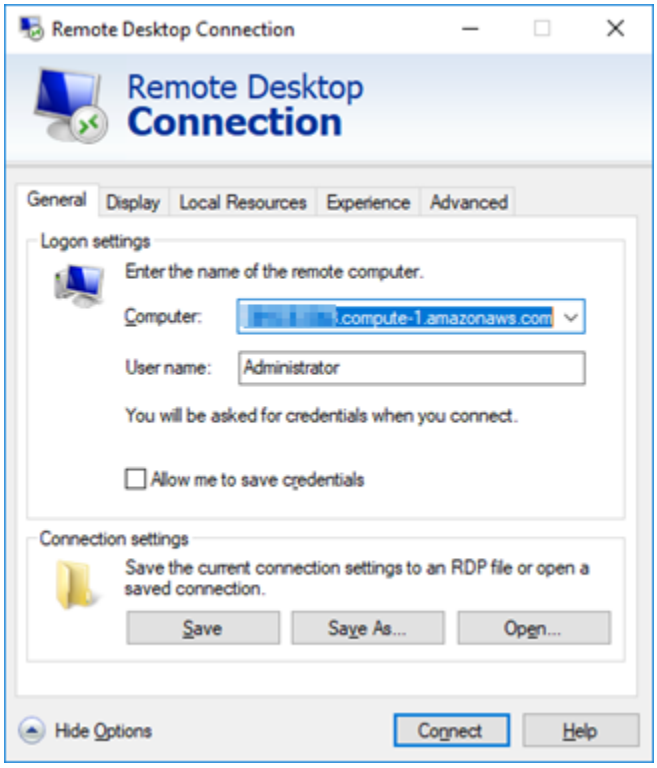
リモートデスクトップ接続は、ほとんどの Windows オペレーティングシステムにプリインストールされている RDP クライアントです。これを使用して、Amazon EC2 の Windows Server インスタンスにグラフィカルに接続します。

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

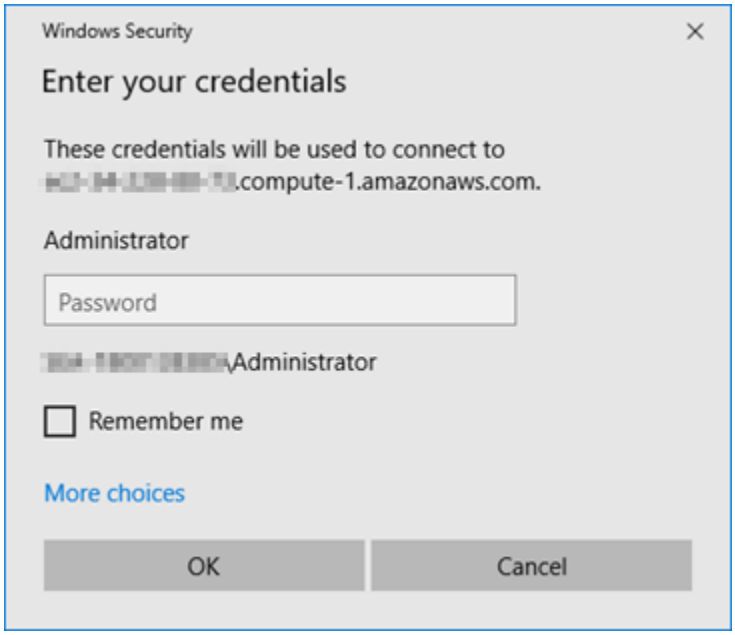
1. リモートデスクトップ接続を開きます。

   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。

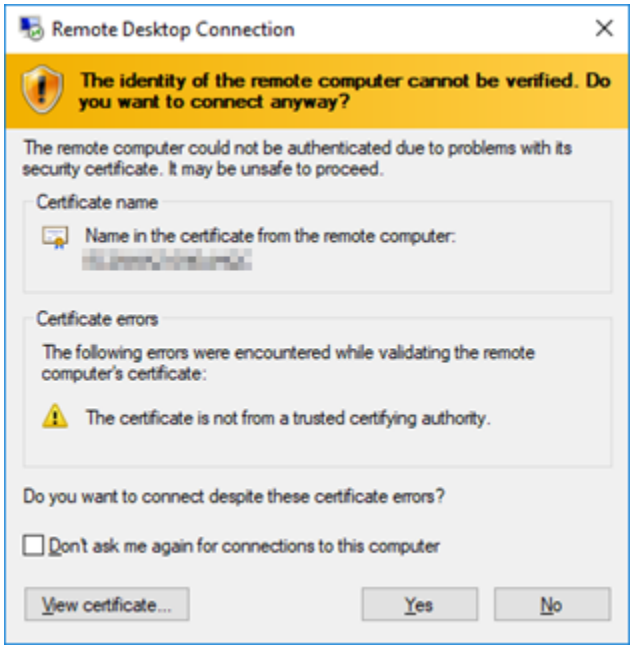
2. [コンピューター] テキストボックスに、このガイドで前に取得した Amazon EC2 の Windows Server インスタンスのパブリック DNS アドレスを入力します。
3. [オプションの表示] を選択して追加のオプションを表示します。
4. `Administrator` をユーザー名テキストボックスに入力します。



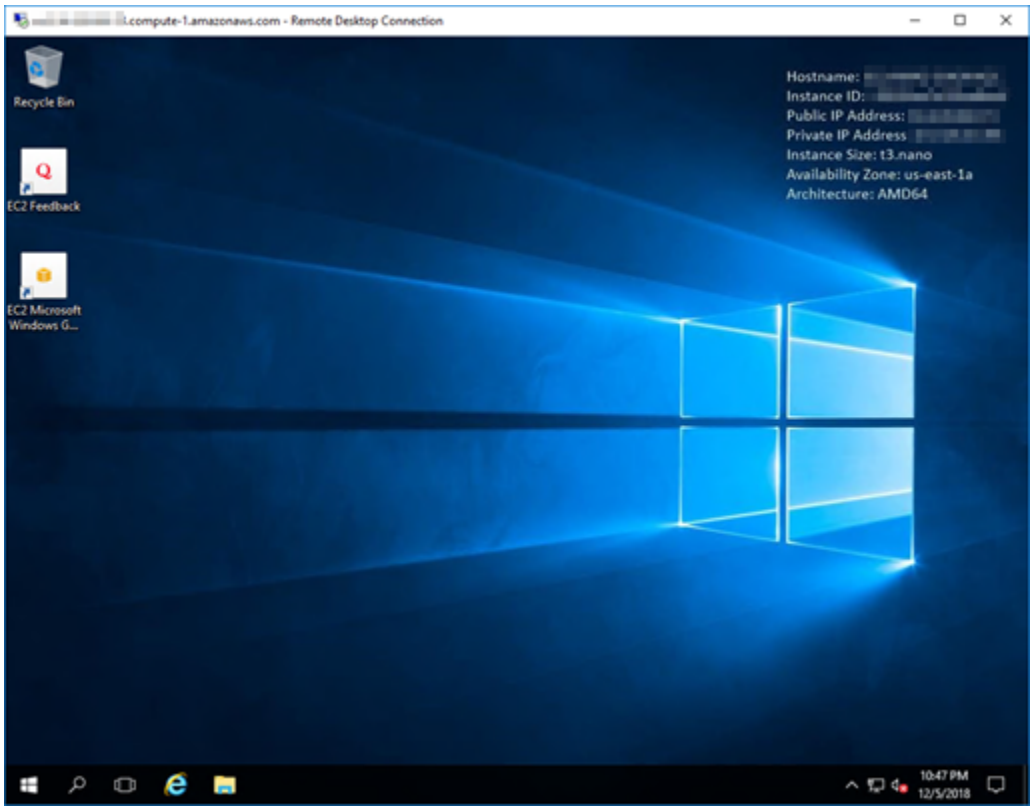
5. [接続] を選択して Windows Server インスタンスに接続します。
6. Windows セキュリティのプロンプトで、[パスワード] テキストボックスに Windows Server インスタンスのパスワードを入力し、[OK] を選択します。



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、デフォルトの Lightsail キーペアと Amazon EC2 の Windows Server インスタンスとの関連付けが削除されます。詳細については、「Lightsail スナップショット から作成された 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 スナップショットから起動される Windows Server Amazon EC2 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された 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には、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 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が削除されます。

> **Note**
>
> Lightsail スナップショットから Amazon EC2 で Linux または Unix インスタンスを作成した場合は、いくつかのステップを実行してそれらのインスタンスを保護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 から作成された Amazon EC2 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 から作成された 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の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て、Lightsail 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削除します。

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。

   ```
   C:\Users\Administrator>net user Administrator "%4=Bwk^GEAg8$u@5"
   The command completed successfully.


   C:\Users\Administrator>
   ```

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 インスタンスの AWS CloudFormation スタックを表示する

Amazon Lightsail は を使用して AWS CloudFormation 、エクスポートされ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「スタックの使用」を参照してください。

AWS CloudFormation スタックには、Lightsail コンソールまたは AWS CloudFormation コンソールからアクセスできます。このガイドでは両方のアクセス方法を示します。

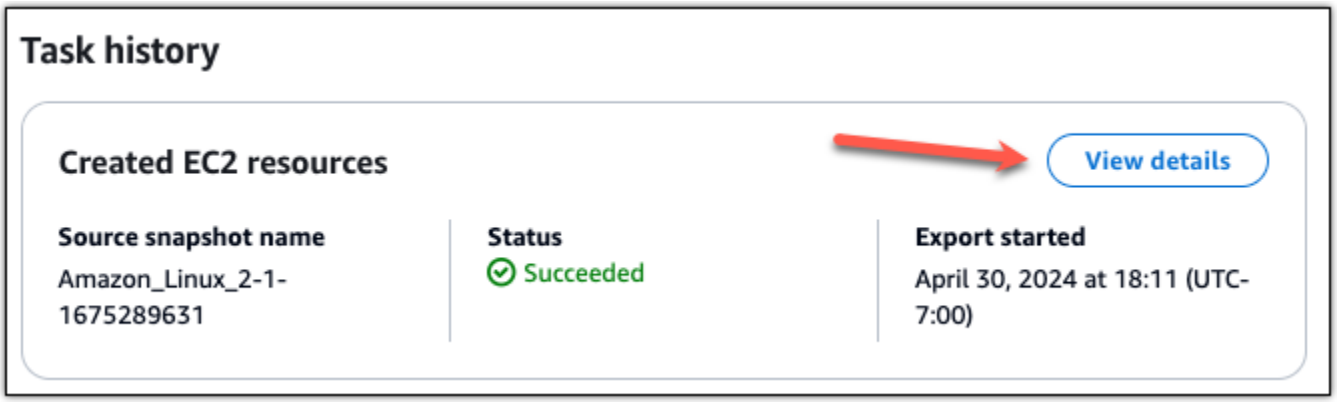
> **Note**
>
> Amazon EC2 リソースの作成に使用される AWS CloudFormation スタックは、Amazon EC2 リソースに永続的にリンクされます。スタックを削除すると、すべての関連リソースが自動的に削除されます。このため、Lightsail によって作成された AWS CloudFormation スタックを削除せず、代わりに Amazon EC2 EC2 リソースを削除する必要があります。

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と、AWS CloudFormation スタックが作成され、そのステータスが Lightsail コンソールのエクスポートセクションで追跡されます。エクスポートの詳細については、「」を参照してくださいLightsail でスナップショットのエクスポートステータスを追跡する。

Lightsail コンソールで AWS CloudFormation スタックを表示するには

1. Lightsail 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でエクスポートを選択します。
3. 以前に作成した Amazon EC2 インスタンスの CloudFormation スタックにアクセスするには、「Created EC2 resources」というラベルが付いたタスクの詳細を表示する」を選択します。

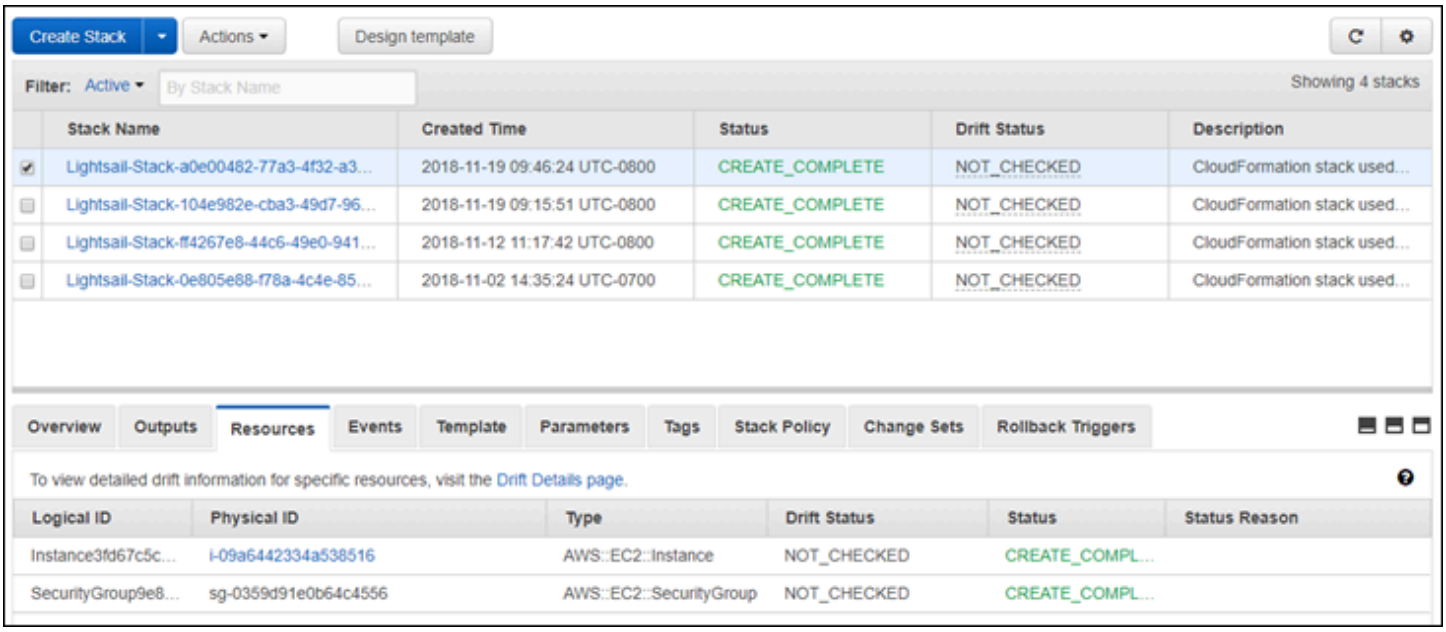


4. 表示される確認ページには、タスクの CloudFormation スタックが一覧表示されます。スタック名を選択して、 AWS CloudFormation コンソールでスタックの詳細を開きます。

AWS CloudFormation コンソールでのスタックへのアクセス

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 によって作成されたスタックは「Lightsail-stack」で始まり、次のスクリーンショットに示すように Amazon EC2 リソースの作成に使用されるCloudFormation スタック」の説明があります。

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。



Lightsail でウェブサイトのドメインを登録して管理する

ウェブサイトには example.com などの名前が必要です。Amazon Lightsail では、ドメイン名と呼ばれるウェブサイトの名前を登録できます。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。 Amazon Lightsail Lightsail は、可用性が高くスケーラブルな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 を使用して、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るか、ドメインの DNS レコードを管理できます。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、引き続きお読みください。

目次

ドメイン登録の仕組み

次の概要は、Amazon Lightsail にドメイン名を登録する方法を示しています。

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 がサポートする最上位ドメイン (TLDsAmazon Lightsail」を参照してください。
2. ドメイン名を Lightsail に登録します。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確認するには、Amazon Lightsail」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 は一連のネームサーバーをドメインに割り当てることで、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、ドメインの DNS サービスとして以下を自動的に実行します。

- ドメインと同じ名前の Lightsail DNS ゾーンを作成します。
- Lightsail DNS ゾーンに 4 つのネームサーバーのセットを割り当てます。
- 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 の他の機能を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 に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 (TLDs) を使用します。 Route 53 Lightsail でドメインを登録するために使用できる汎用 TLDsAmazon Route 53」の「Amazon Route 53 に登録できるドメイン」を参照してください。 Amazon Route 53

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 を使用して登録された後、Lightsail コンソールで使用できます。詳細については、「Amazon Route 53 デベロッパーガイド」の「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 はドメイン登録に Route 53 を使用します。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

以下の記事は、Lightsail でドメインを管理するのに役立ちます。

- 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

Lightsail DNSの理解

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) アドレスを参照することで、Lightsail インスタンスのウェブアプリケーションにアクセスできます。これは、 IPv4またはIPv6 アドレスです。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。したがって、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には、 などの easy-to-remember ドメイン名を参照`example.com`するようにユーザーに指示する必要があります。これは、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するドメインネームシステム (DNS) を通じて実現されます。

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は、ドメイン名の登録時に作成されたDNSゾーンの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ている場合は、レジストラでDNSレコードを管理するか、ドメインの の管理を Lightsail DNS に移管できます。

ドメイン名を Lightsail インスタンスにマッピングしやすくするために、DNSゾーンを作成してドメインのDNSレコードの管理を Lightsail に転送することをお勧めします。詳細については、「ドメインのDNSレコードを管理するDNSゾーンを作成する」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、Route 53 を使用してすべてのドメインDNSの を管理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 インスタンスにポイントできます。Route 53 DNSで を管理する方法の詳細については、Amazon Route 53を使用してドメインをインスタンス にポイントする」を参照してください。

DNS の用語

ドメインDNSの を管理できるように、使い慣れた用語が必要です。

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、 などの easy-to-remember ドメイン名をウェブサーバーの IP アドレス`example.com`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。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。

詳細については、Wikipedia のDNS「レコードタイプのリスト」を参照してください。

DNS ゾーン

DNS ゾーンは、 などの特定のドメイン、`example.com`および などのそのサブドメインに対してインターネット上でトラフィックを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`blog.example.com`。

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、ネームサーバーは、 easy-to-remember ドメイン名を IP 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 AWS インスタンスです。Lightsail には、トラフィックをドメインにルーティングするための AWS ネームサーバーオプションがいくつか用意されています (例: `ns-NN.awsdns-NN.com`) 。ドメインレジストラを使用してドメインを変更するときに、これらの AWS 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レコードの有効期間を決定します。例えば、時間が短いほど、変更が有効になるまでの待機時間が短くなります。TTL Lightsail DNS ゾーンではを設定できません。代わりに、すべての Lightsail DNSレコードのデフォルトは 60 秒TTLです。

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のリクエストに一致します。ワイルドカードDNSレコードは、アスタリスク記号 (*) を `*.example.com`や などのドメイン名の左端部分として使用して指定します`*example.com`。

> **Note**
>
> Lightsail DNS ゾーンは、ネームサーバー (NS*awsdns.com) レコードで定義されたネームサーバードメイン () のワイルドカードレコードをサポートします。

DNS Lightsail DNS ゾーンでサポートされているレコードタイプ

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。

例えば、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転送します。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「レコードタイプのリスト」を参照してください。

AAAA レコード

AAAA 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をウェブサーバーのIPv6アドレスblog.example.comにマッピングします。

例えば、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IPv6プロトコル経由でインスタンスに転送します。AAAA レコードを作成し、サブドメインテキストボックスに@記号を入力し、ウェブサーバーの IP アドレスを解決してアドレスを指定する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「 のドメインネームシステムIPv6」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する場合、または同じリソースIPv6で 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。

正規名 (CNAME) レコード

CNAME レコードは、 などのエイリアスまたはサブドメインを別のドメインまたはサブドメインblog.example.comにマッピングします。

例えば、Lightsail DNS ゾーンでは、 のウェブトラフィックを www.example.comに転送しますexample.com。のエイリアスCNAMEレコードを「解決先」アドレス wwwで作成しますexample.com。

詳細については、Wikipedia のCNAME「レコード」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。

例えば、Lightsail DNS ゾーンでは、10 inbound-smtp.us-west-2.amazonaws.comAmazon WorkMail サーバーmail.example.comにメールを送ります。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、 などのサブドメインをservice.example.com、優先度、重み、ポート番号の値を持つサービスアドレスにマッピングします。テレフォニーまたはインスタントメッセージングは、レコードに通常関連づけられるサービスの一部ですSRV。

例えば、Lightsail DNS ゾーンでは、 のトラフィックを service.example.comに転送します1 10 5269 xmpp-server.example.com。優先度が 1、重みが 、ポート番号が 10、住所5269が の「マップ先」のSRVレコードを作成しますxmpp-server.example.com。

詳細については、Wikipedia のSRV「レコード」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 レコードを作成して、サービスプロバイダーへのドメインの所有権を確認します。

例えば、Lightsail DNS ゾーンでは、`_amazonchime.example.com`ホスト名がクエリされた`23223a30-7f1d-4sx7-84fb-31bdes7csdbb`ときに で応答します。サブドメイン値が で`_amazonchime`、「 と応答する」値が のTXTレコードを作成します`23223a30-7f1d-4sx7-84fb-31bdes7csdbb`。

詳細については、Wikipedia のTXT「レコード」を参照してください。

Lightsail インスタンスのドメインレコードを管理するDNSゾーンを作成する

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、Lightsail で効率的に管理できます。Lightsail DNS ゾーンを作成することで、Lightsail を使用してドメインのDNSレコードを管理できます。最大 6 つの Lightsail 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、6 つ以上のドメイン名を管理するため、Amazon Route 53 を使用してすべてのドメインDNSの を管理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 DNSで を管理する方法の詳細については、Amazon Route 53を使用してドメインをインスタンス にポイントする」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 に転送する方法について説明します。ドメインのDNSレコードの管理を Lightsail に転送した後も、ドメインのレジストラでドメインの更新と請求は引き続き管理されます。

> **Important**
>
> ドメインDNSの に加えた変更は、インターネットの を伝播するのに数時間かかる場合がありますDNS。このため、Lightsail への管理の移管が伝達される間は、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に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。

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 と AAAA) 、正規名 ()、メールエクスチェンジャー (MXCNAME) 、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 を使用することをお勧めします。詳細については、「Amazon Route 53 デベロッパーガイド」の「サポートされているDNSレコードタイプ」およびAmazon Route 53を既存のドメインのサービスにする」を参照してください。 Amazon Route 53 DNS
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は、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. 左側のナビゲーションペインで、ドメインと DNSを選択します。次に、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。このガイドの後半にある「ステップ 3: DNS ゾーンにレコードを追加する」セクションの一部として、アドレス (A) レコードwwwを使用して を追加できます。

   > **Note**
   >
   > Lightsail DNS ゾーンはバージニア (`us-east-1`) で作成されます 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。

   ゾーンDNSの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できます。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、次のステップを実行します。DNS レコードは、ドメインのインターネットトラフィックのルーティング方法を指定します。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 ゾーン割り当てページで、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タブに一覧表示されます。

> **Note**
>
> DNS ゾーン割り当てページで、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 レコードページで、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のアドレス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指定して、トラフィックを Lightsail とディストリビューションやコンテナサービスなどの AWS リソースにルーティングします。ゾーン内の 1 つのレコードから別のレコードDNSにトラフィックをルーティングすることもできます。

      > **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、 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のアドレスまたはインスタンスのIPv6アドレスにマッピングします`2001:0db8:85a3:0000:0000:8a2e:0370:7334`。

> **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IPv6で 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. 「解決先への解決」テキストボックスに、レコードのターゲットIPv6アドレスを入力するか、実行中のインスタンスを選択するか、ロードバランサーを設定します。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指定して、トラフィックを Lightsail およびディストリビューションやコンテナサービスなどの AWS リソースにルーティングします。ゾーン内の 1 つのレコードから別のレコードDNSにトラフィックをルーティングすることもできます。

正規名 (CNAME) レコード

CNAME レコードは、 などのエイリアスまたはサブドメインを`www.example.com`、などの別のドメイン`example.com`、または などの別のサブドメインにマッピングします`blog.example.com`。

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、 などのサブドメインをservice.example.com、優先度、重み、ポート番号の値を持つサービスアドレスにマッピングします。テレフォニーまたはインスタントメッセージングは、レコードに通常関連づけられるサービスの一部ですSRV。

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. 重みテキストボックスに、同じ優先度の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 レコードを作成して、サービスプロバイダーへの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。上記の手順を繰り返して、ドメインのDNSゾーンに複数のレコードを追加します。

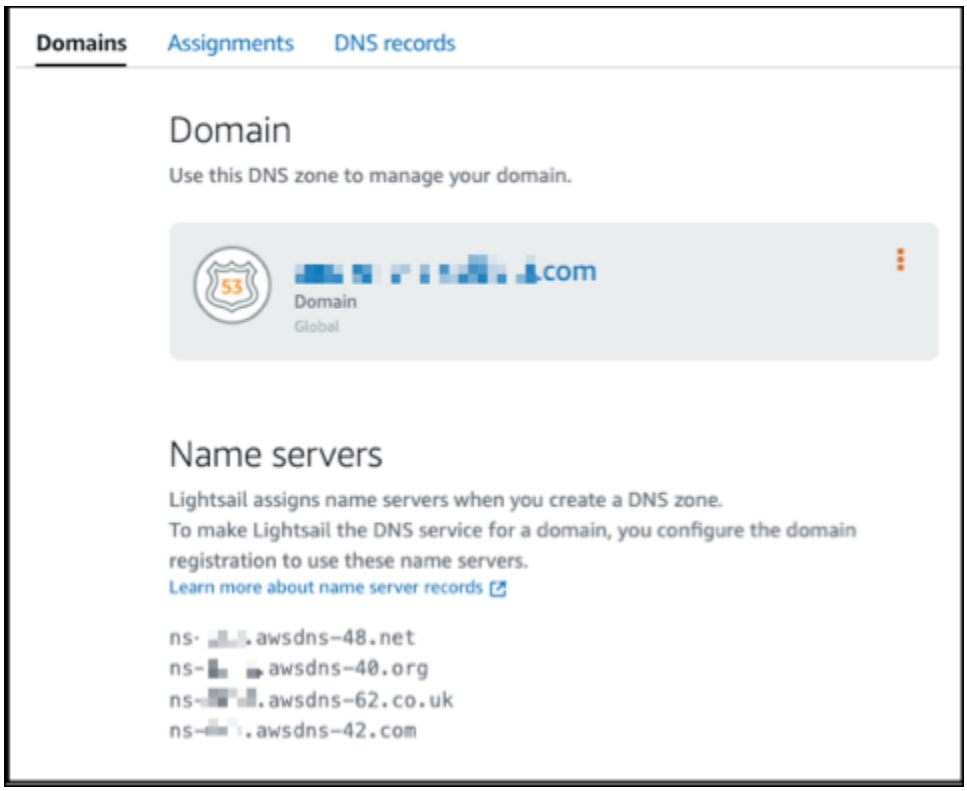
> **Note**
>
> DNS レコードの有効期限 (TTL) は Lightsail DNS ゾーンで設定できません。代わりに、すべての Lightsail DNSレコードのデフォルトは 60 秒TTLです。詳細については、Wikipedia の「Time to live」を参照してください。

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の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、トラフィックは Lightsail DNS ゾーンへの転送後も中断されずに継続的に流れます。

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 ゾーンのドメインタブにあります。 DNS



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。

   このページの検索の詳細については、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、リスト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 を介して伝播されるまでDNSに数時間かかることがあります。その後、ドメインのインターネットトラフィックは Lightsail DNS ゾーン経由のルーティングを開始する必要があります。

次のステップ

- DNS ゾーンを編集する
- 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 ゾーンを編集する

ドメインのDNSゾーンのDNSレコードを編集します。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移管する場合、またはドメインを登録したレジストラに移管する場合は、Amazon Lightsail でドメインのDNSゾーンを削除することもできます。詳細については、「???」を参照してください

> **Note**
>
> Lightsail コンソールのDNSエディタを使用してレコードを編集する前に、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、「ドメインのDNSレコードを管理するDNSゾーンを作成する」を参照してください。

DNS レコードの編集

Lightsail コンソールを使用して、ドメインのDNSゾーンのDNSレコードを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. Lightsail コンソールのホームページの左側のナビゲーションペインで、ドメインと DNSを選択します。
3. 編集するDNSゾーンの名前を選択します。
4. DNS ゾーンDNSレコードページで、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。
5. 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 を介して伝達DNSされるまでに数時間かかることがあります。

Lightsail でDNSゾーンを削除する

場合によっては、Amazon Lightsail で設定したDNSゾーンを完全に削除して、ドメインのDNSレコードを管理できます。DNS 管理を別のプロバイダーに移管するか、ドメインレジストラに移管し直す必要があるかもしれません。DNS ゾーンの削除は簡単なプロセスですが、ドメインのトラ

フィックが正しくルーティングし続けるように事前に計画することが重要です。Lightsail でDNS ゾーンを削除する手順について説明します。

> **Important**
>
> ドメイン経由でトラフィックをルーティングし続ける場合は、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てください。それ以外の場合、Lightsail DNS ゾーンを削除すると、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の左側のナビゲーションペインで、ドメインと DNS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. DNS ゾーンの削除 を選択して、削除を確定します。

   DNS ゾーンは Lightsail から削除されます。

Lightsail でインターネットトラフィックがウェブサイトにルーティングされる方法について説明します。

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

この概要では、Lightsail を使用して、インターネットトラフィックをウェブサイトまたはウェブアプリケーションにルーティングするドメインを登録および設定する方法について説明します。

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 はドメインと同じ名前の DNS ゾーンを自動的に作成します。
3. Lightsail コンソールを使用すると、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、ドメイン名をインスタンスの静的 IPv4 アドレスを指す 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。
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。Lightsail インスタンス、Lightsail コンテナサービス、およびその他の Lightsail リソースにトラフィックを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、www.example.com のコンテンツをリクエストすると、次のようになります。

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や、ウェブサイトエンドポイントとして設定されたコンテナサービスなど、コンテンツがある場所です。
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

ドメイントラフィックを Lightsail インスタンスにルーティングする

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、仮想プライベートサーバー (VPS) とも呼ばれる Lightsail インスタンスで実行されているウェブサイトにポイントできます。Lightsail アカウントでは、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。

Lightsail でサポートされていない DNS ゾーンを 6 つ以上作成する場合や、DNS レコードタイプを使用する場合は、Amazon Route 53 ホストゾーンを使用することをお勧めします。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 インスタンスを指すようにする方法を示します。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。
- ドメインを既に登録しているが、Lightsail を使用してレコードを管理していない場合は、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 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、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 IPv6 を有効にしたままにすることができます。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 のインスタンスにドメインを割り当てるには、次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 コンソールのインスタンスドメインタブでドメインを Lightsail インスタンスに割り当てるには、次の手順を実行します。

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 コンソールの静的 IP ドメインタブでドメインを Lightsail インスタンスに割り当てるには、次の手順を実行します。

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 ゾーンの割り当てタブでドメインを Lightsail インスタンスに割り当てるには、次の手順を実行します。

[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. [Domains & DNS] (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。

ドメインを Lightsail ロードバランサーにポイントする

暗号化 (HTTPS) トラフィック があるドメインを制御していることを確認したら、ドメインを Lightsail ロードバランサーにポイントするアドレス (A) レコードをドメインの 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、A レコードを Lightsail DNS ゾーンと Amazon Route 53 ホストゾーンに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 ホームページで、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 ホームページで、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)

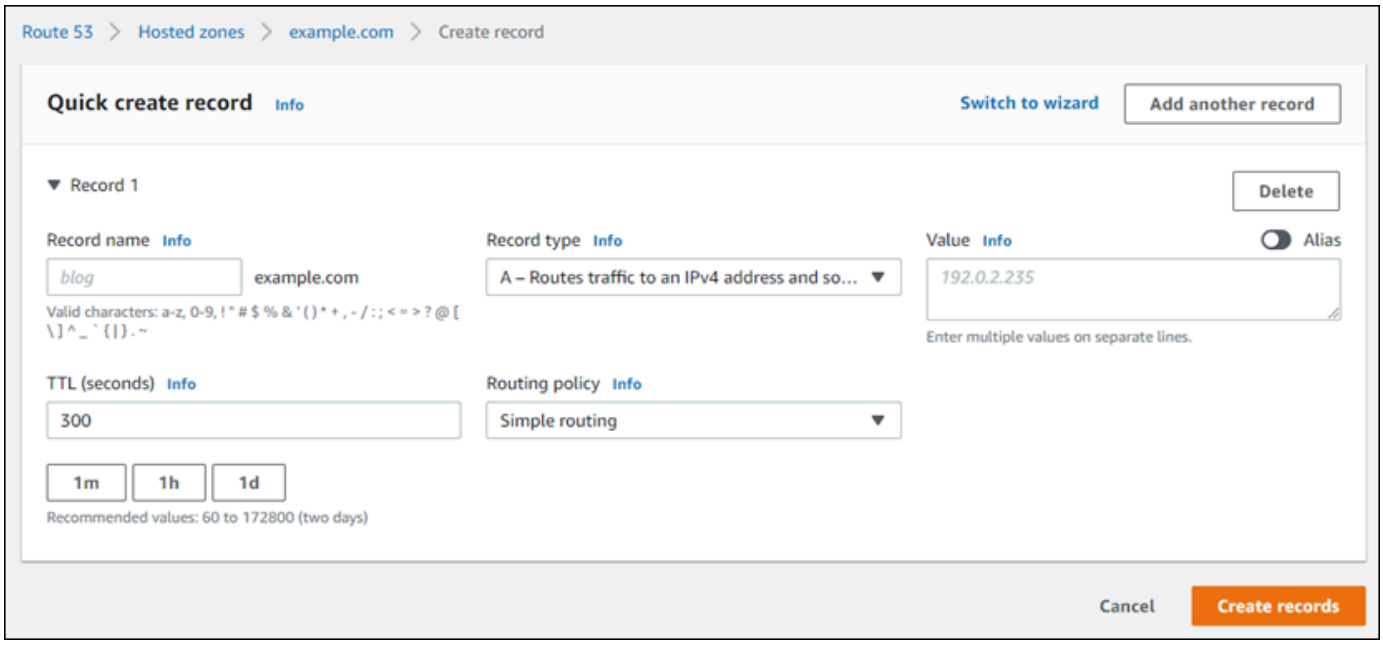
7. 「解決先」テキストボックスで、Lightsail ロードバランサーの名前を選択します。
8. 保存アイコンを選択します。

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。

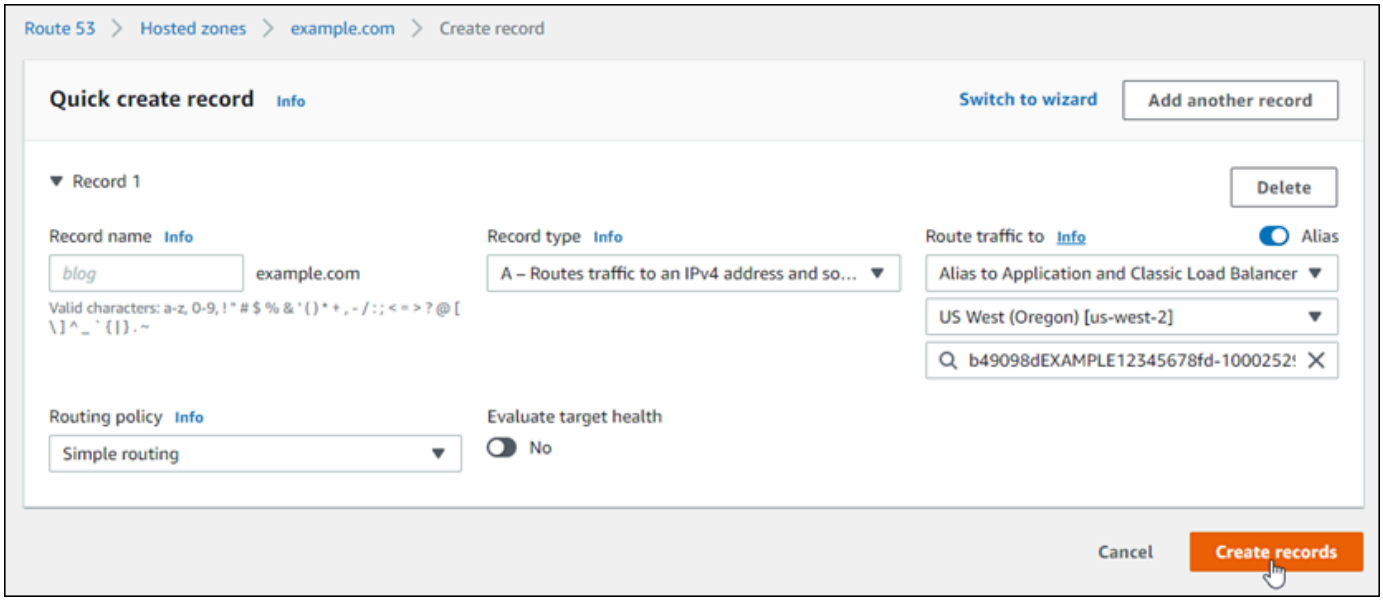
   a. エンドポイントの選択は、アプリケーションにエイリアスおよびClassic Load Balancerを選択します。

   b. リージョンの選択 で、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 で、Lightsail ロードバランサーのエンドポイント URL (DNS 名) を入力または貼り付けます。
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 はロードバランサーのヘルスチェックをすでに実行しています。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。

Lightsail ドメインの DNS 管理を移管する

Amazon Lightsail DNS ゾーンを使用して、Lightsail を使用して登録したドメインの DNS レコードを管理できます。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 ホスティングプロバイダーに転送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

Lightsail の登録済みドメインに別の DNS ホスティングプロバイダーのネームサーバーを追加するには、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. [Domains & DNS] (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、 などの登録済みドメイン名を `example.com`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。最大 6 つの Lightsail DNS ゾーンを作成できますが、すべての DNS レコードタイプが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は、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 インスタンスを指す方法について説明します。

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、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

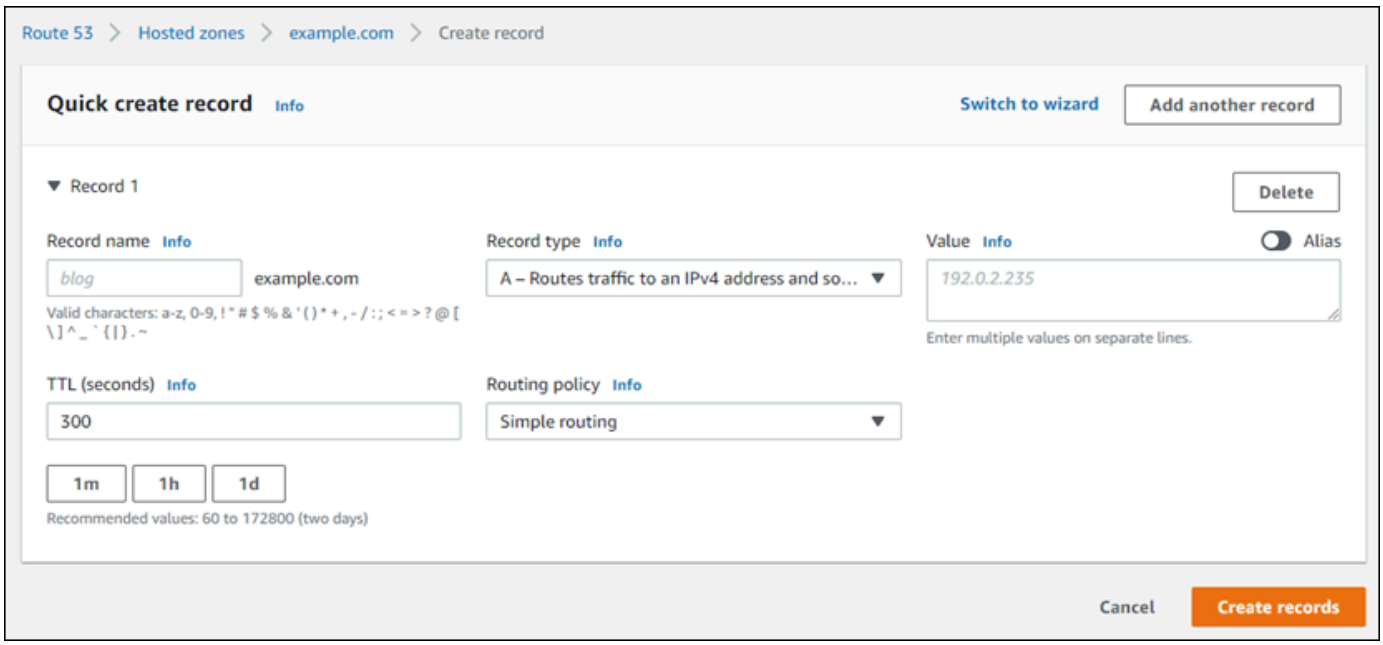
Route 53 でドメインを Lightsail インスタンスを指すように、2 つの最も一般的な DNS レコードであるアドレスと正規名を設定するには、次のステップを実行します。

> **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。

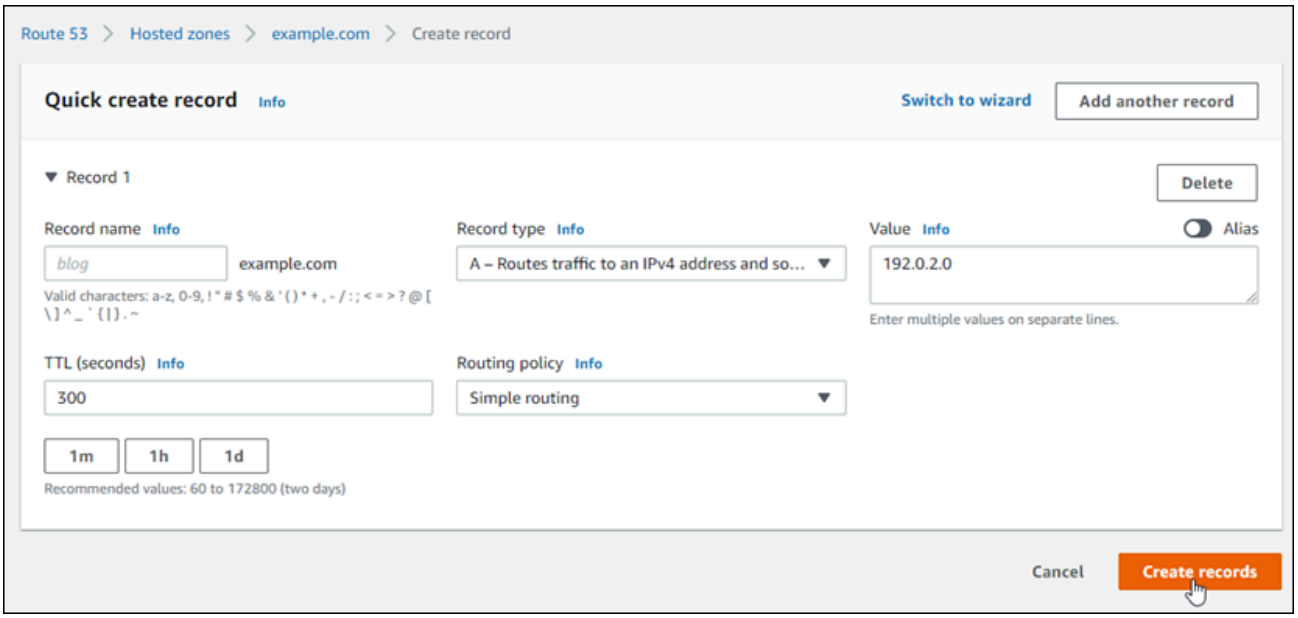
5. [レコードのタイプ] で、以下のいずれかのオプションを選択します。

   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。

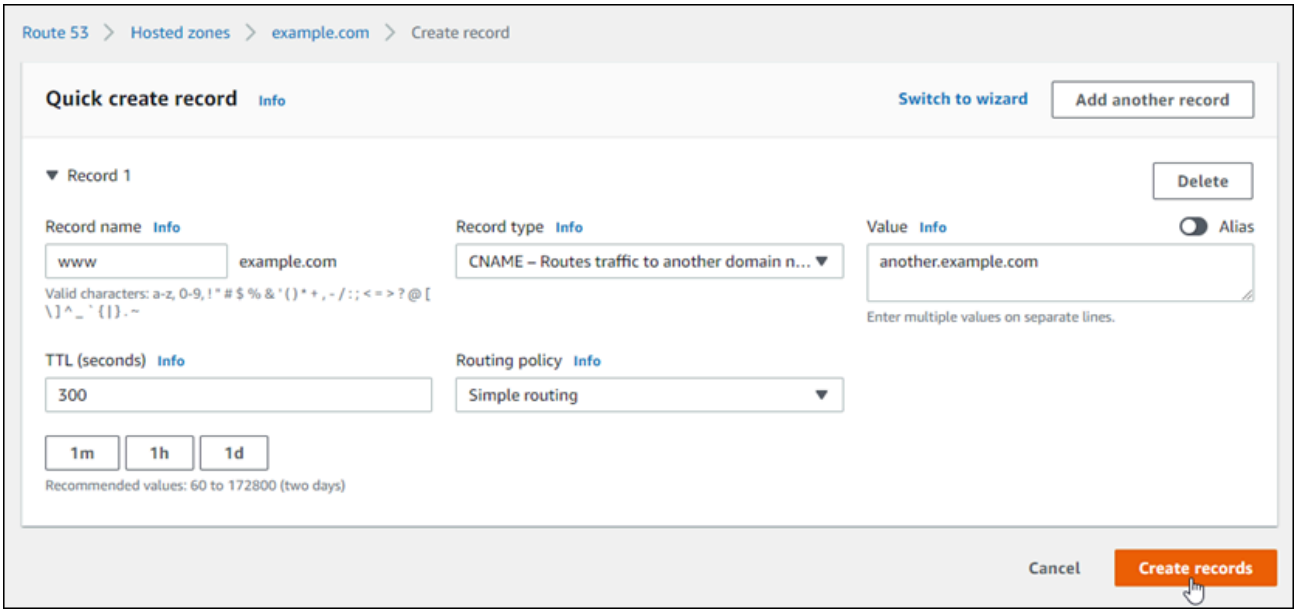
3. Lightsail インスタンスの静的 IP アドレス (パブリック IP アドレス) を値テキストボックスに入力します。
4. TTL を 300 に保ち、ルーティングポリシーを [シンプルルーティング] のままにしておきます。



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 に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。Lightsail ドメインは、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 を通じて登録されます。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 DNS 管理を Lightsail に転送できます。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。

Lightsail に新しいドメインを登録するには、次のいずれかの手順を選択します。

- 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。

- ドメインを別のレジストラに移動するには、Amazon Route 53 で Lightsail ドメインを管理する」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 は、すべての汎用最上位ドメイン (TLDs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

Lightsail にドメインを登録すると、ドメインの DNS ゾーンが自動的に作成されます。Lightsail では、DNS ゾーンの料金は発生しません。

Lightsail を使用して新しいドメインを登録する

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)、正規名 (CNAME)、メールエクスチェンジャー (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、ドメインの DNS ホスティングプロバイダーとして Route 53 を使用することをお勧めします。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. [Domains & DNS] (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、Lightsail を DNS サービスとして使用するか、別の DNS サービスを使用するかを選択します。
   - Lightsail

     ドメインの登録時に Lightsail が作成した DNS ゾーンで、ドメインとサブ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を Lightsail に伝えるレコードを作成します。

     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 に転送された場合、Lightsail はウェブサーバーの IP アドレスまたはロードバランサーの名前でクエリに応答しますか？ 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

     DNS クエリを Lightsail 以外の DNS サービスに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の登録の詳細を表示する

Amazon Registrar がレジストラである Amazon Lightsail と Amazon Route 53 を使用して登録された .com、.net、.org 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 に登録されていて、プライバシー保護がアクティブである場合、WHOISクエリへのレスポンスにおける登録者の E メールの値は と似ていますowner1234@example.com.whoisprivacyservice.org。

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、違法行為または利用ポリシー の違反を報告するには、trustandsafety@support.aws.com 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、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 で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ています。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 は、アルファベット文字を大文字 (A～Z) で指定した場合でも、小文字 (a～z) として保存します。

Lightsail は、汎用 TLD と地理的 TLDs の両方の DNS ゾーンをサポートしています。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 NXDOMAIN (存在しないドメイン) を使用して example.com の DNS クエリに応答します。

Lightsail を設定して、同じレベルのすべてのサブドメインとドメイン名の DNS クエリに同じレスポンスを返すことができます。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

高度な Route 53 機能を使用して Lightsail ドメインを管理する

Amazon Lightsail 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と、Lightsail と Route 53 の両方でドメインを管理できます。

ドメインの登録、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 コンソールで行われます。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に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は、登録期間 (期間) やドメインの連絡先情報などのドメイン情報を指定します。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

言葉

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

Lightsail にドメインを登録すると、自動的に更新されるようにドメインが設定されます。自動更新期間は通常 1 年間です。Lightsail が有効期限が切れる前にドメインを自動的に更新するかどうかを選択します。登録料は AWS アカウントに請求されます。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。

都道府県

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 でのドメイン登録の更新または無効化

Amazon Lightsail にドメインを登録すると、デフォルトで自動的に更新されるようにドメインが設定されます。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**
>
> を閉じる場合は、必ず自動更新を無効にしてください AWS アカウント。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

Lightsail に新しいドメイン名を登録すると、自動的に更新されるようにドメインが設定されます。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. [Domains & DNS] (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、[Check availability] (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 が有効期限の直前にドメインの登録を自動的に更新するかどうかを変更する場合、または自動更新の現在の設定を表示する場合は、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. [Domains & DNS] (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 でドメイン連絡先のプライバシー保護を管理する

Amazon Lightsail にドメインを登録すると、すべてのドメイン連絡先に対してプライバシー保護がデフォルトで有効になります。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

Lightsail にドメインを登録します。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 でドメイン連絡先情報を更新する

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。

現在 Lightsail に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. [Domains & DNS] (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. [保存] を選択します。

Lightsail でのリレーショナルデータベースの作成と管理

Amazon Lightsail で MySQL または PostgreSQL マネージドデータベースをいくつかのステップで作成できます。 Amazon Lightsail Lightsail は、一般的なメンテナンスおよび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。 AWS Command Line Interface （ AWS CLI ）、API、または SDK を使用して Lightsail データベースを管理することもできます。

プロジェクトに適した Lightsail データベースを選択する

Amazon Lightsail は、MySQL データベースと 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 には、SQL Server を搭載した Windows Server 2022 インスタンスも用意されています。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 で利用できます。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で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、16 は Lightsail で利用できます。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で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、あるアベイラビリティーゾーンのプライマリデータベースと別のアベイラビリティーゾーンのセカンダリスタンバイデータベースとのフェイルオーバー

サポートを提供します。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 で使用できる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。

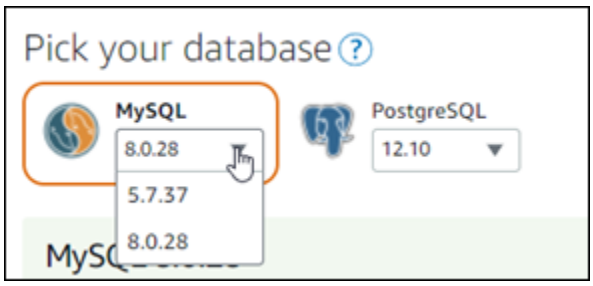
高可用性を備えた Lightsail データベースを作成する

Amazon Lightsail でマネージドデータベースを数分で作成します。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、「データベースの選択」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 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 とアベイラビリティーゾーンを選択します。
   1. 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、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。使用可能なデータベースエンジンオプションの 1 つで、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いずれかを選択します。



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、Lightsail は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 字目は英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 字目は英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を指定する - 独自のプライマリデータベース名を指定するか、Lightsail で名前を指定します。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。

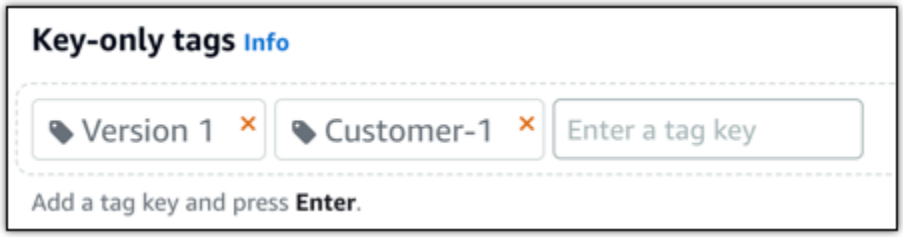
8. データベースの名前を入力します。

   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。

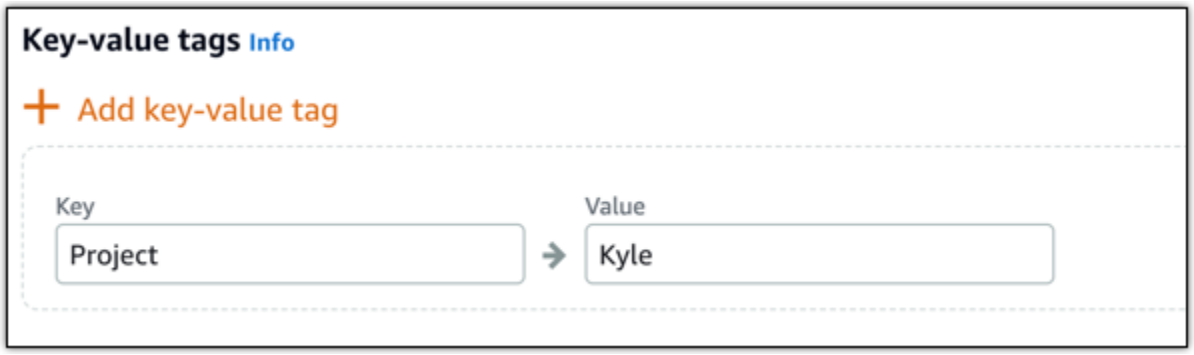
9. 以下のいずれかのオプションを選択して、データベースにタグを追加します。

   - [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。

    Lightsail データベースの準備は数分で完了します。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を起動して実行した後で管理するためのガイドをいくつか紹介します。

- 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 で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

クライアントアプリケーションから 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したら、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エンドポイント、ポート、ユーザー名、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。

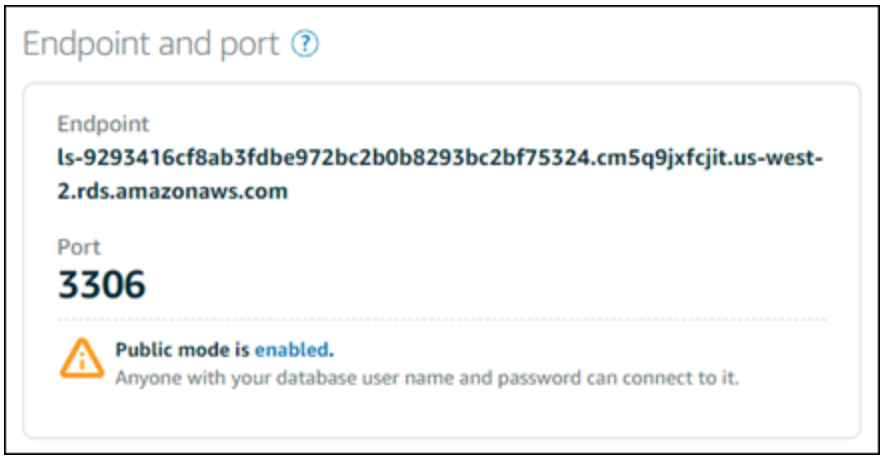
ステップ 1: MySQL データベース接続の詳細を取得する

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 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

データベースを外部に接続するには、パブリックモードを有効にするか、データベース AWS リージョン とは異なる の Lightsail インスタンスから接続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。

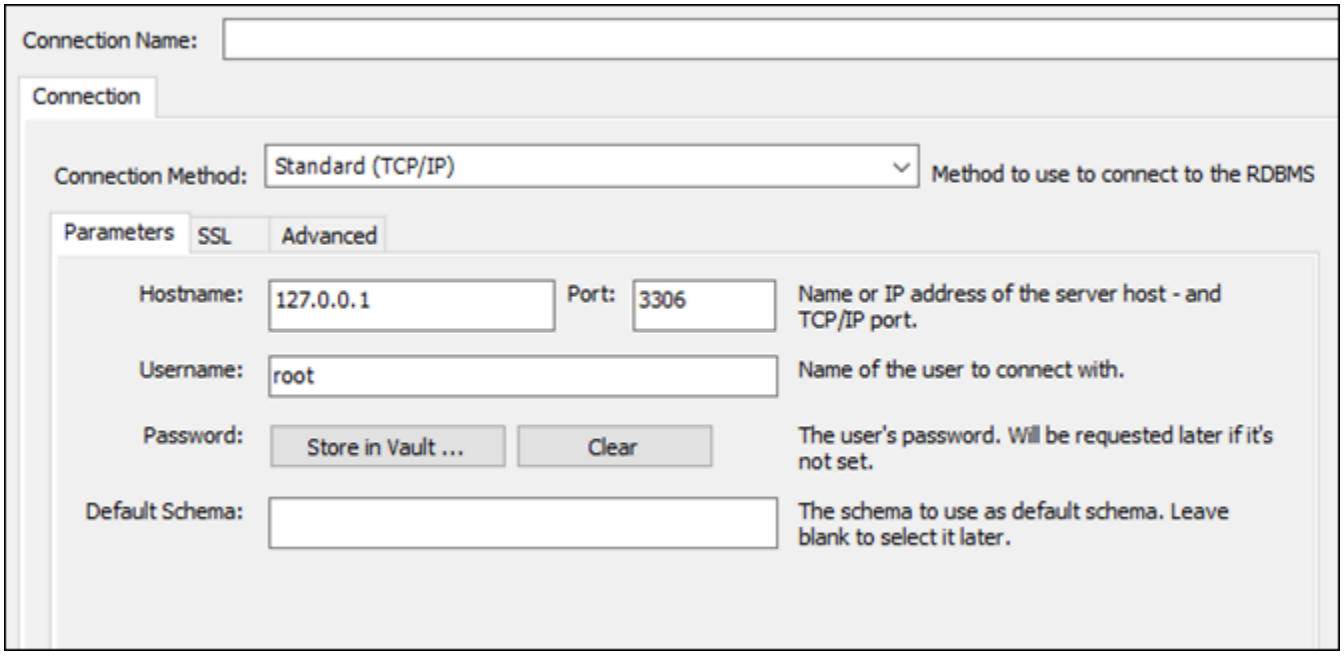
ステップ 3: MySQL データベースに接続するようにデータベースクライアントを設定する

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

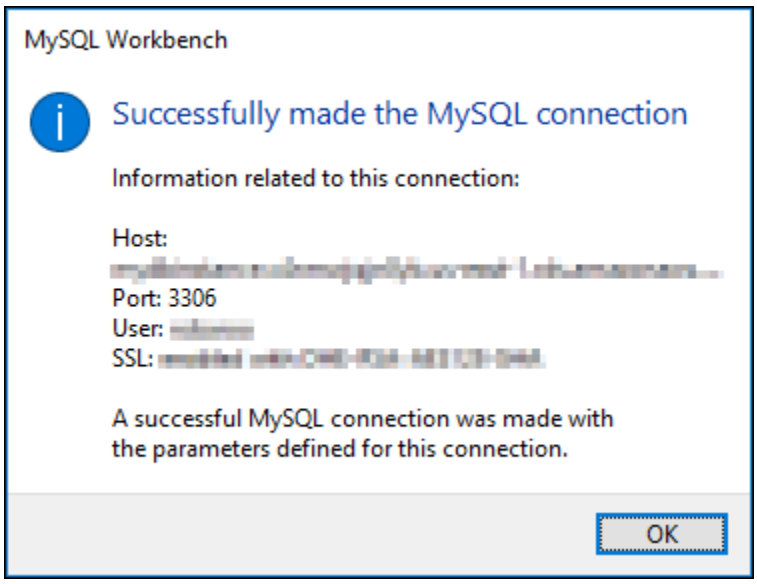
1. MySQL Workbench を開きます。
2. [Database (データベース)] メニュー、[Manage connections (接続の管理)] の順に選択します。
3. 表示されるフォームに以下の情報を入力します。



- 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。Lightsail コンソールからデータベースエンドポイントをコピーし、まだクリップボードにある場合は、Windows を使用している場合は Ctrl+V、macOS を使用している場合は Cmd+V 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。Lightsail コンソールからパスワードをコピーし、まだクリップボードにある場合は、Windows を使用している場合は Ctrl+V、macOS を使用している場合は Cmd+V 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。

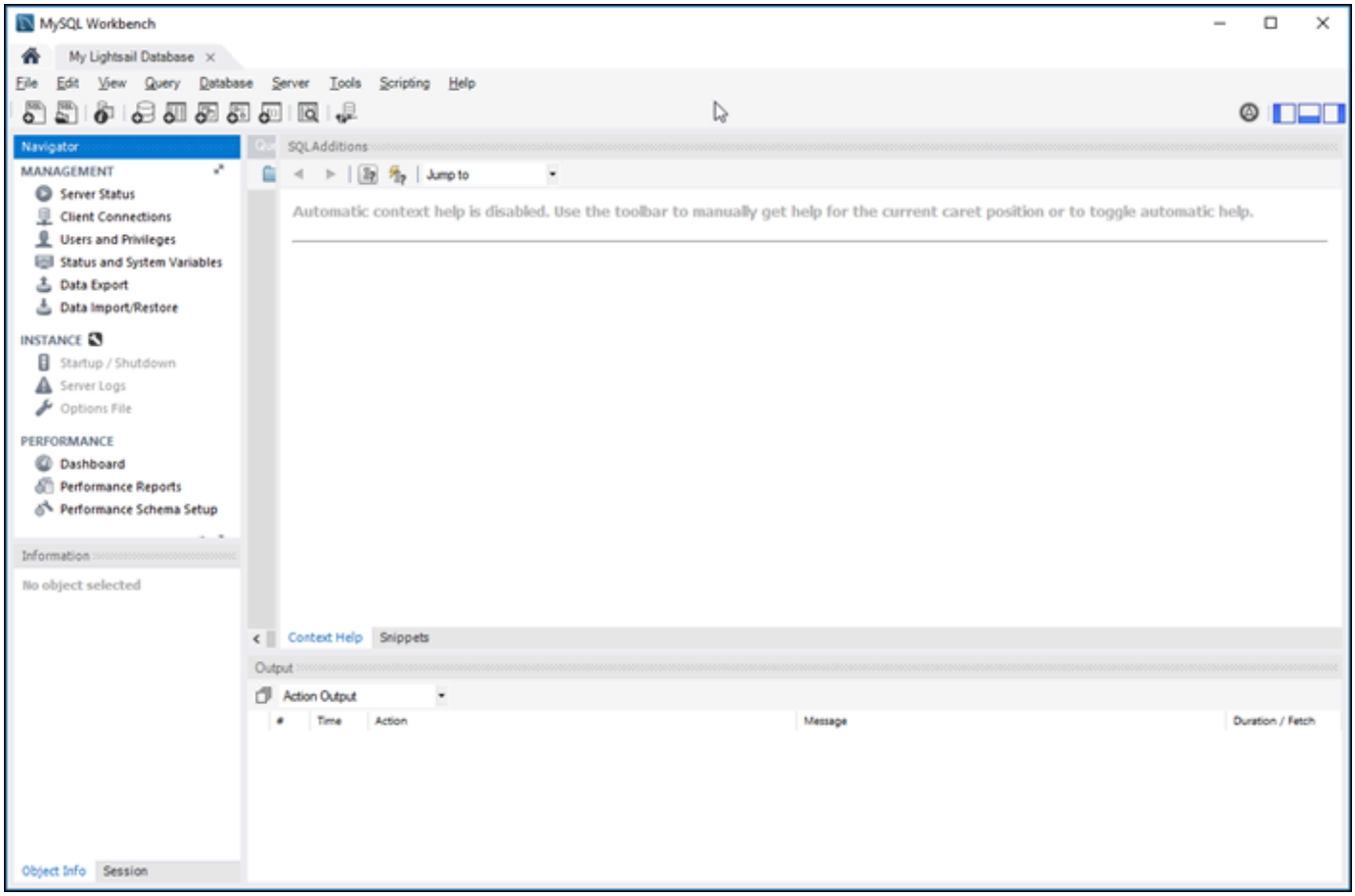


5. [New (新規)] を選択して新しい接続の詳細を保存し、[Close (閉じる)] を選択して接続管理ウィンドウを閉じます。

   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

以下は、Lightsail のデータベースにデータをインポートするのに役立つガイドです。

- MySQL データベースにデータをインポートする

SSL/TLS を使用して Lightsail MySQL データベースに安全に接続する

Amazon Lightsail は SSL 証明書を作成し、プロビジョニング時に MySQL マネージドデータベースにインストールします。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、ほとんどの場合機能しますが、アプリケーションが証明書チェーンを受け入れない場合、失敗する可能性がありま

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* データベースのエンドポイントを使用する。
     - 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を含む */path/to/certificate/rds-combined-ca-bundle.pem*。
     - *UserName* データベースのユーザー名を入力します。

     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。

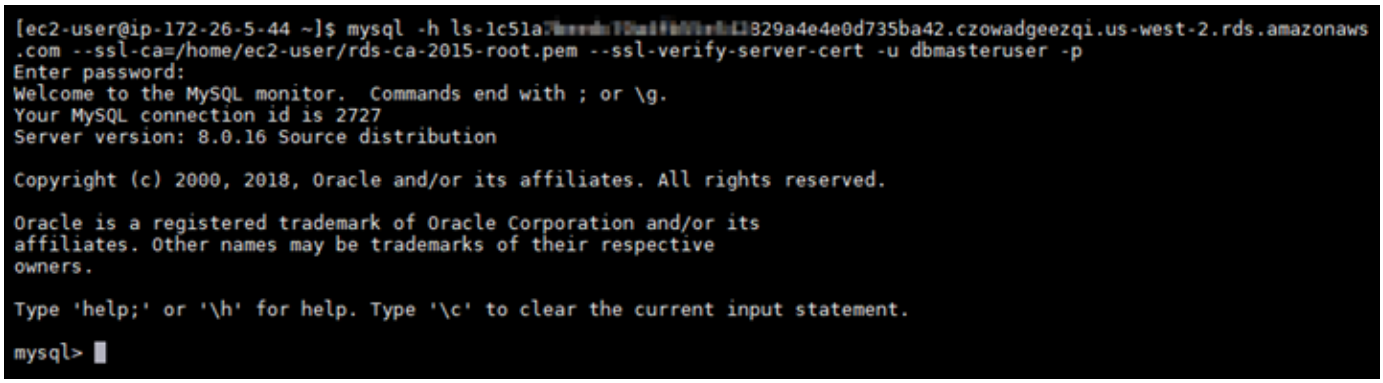
     - *DatabaseEndpoint* データベースのエンドポイントを使用する。
     - データベースの証明書をダウンロードして保存したローカルパスを含む */path/to/certificate/rds-combined-ca-bundle.pem*。
     - *UserName* データベースのユーザー名を入力します。

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
```

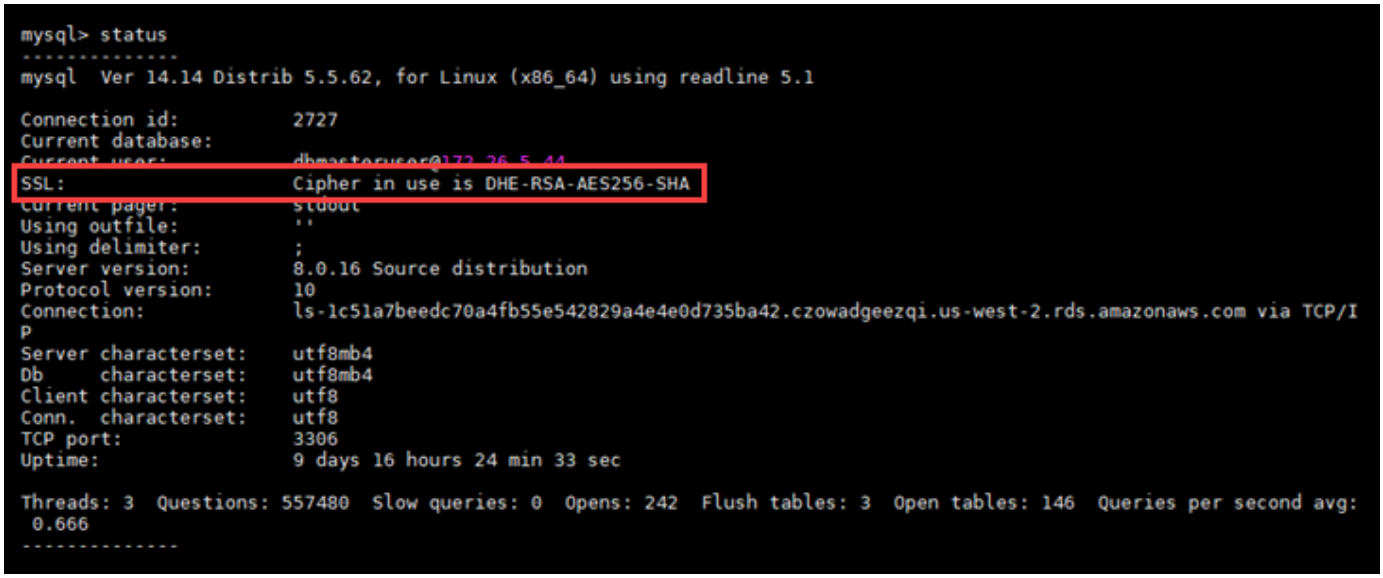
3. プロンプトが表示されたら、前のコマンドで指定したデータベースユーザーのパスワードを入力し、Enter キーを押します。

   以下の例のような結果が表示されるはずです。



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。



Lightsail PostgreSQL データベースインスタンスに接続する

Amazon Lightsail で PostgreSQL マネージドデータベースを作成したら、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エンドポイント、ポート、ユーザー名、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。

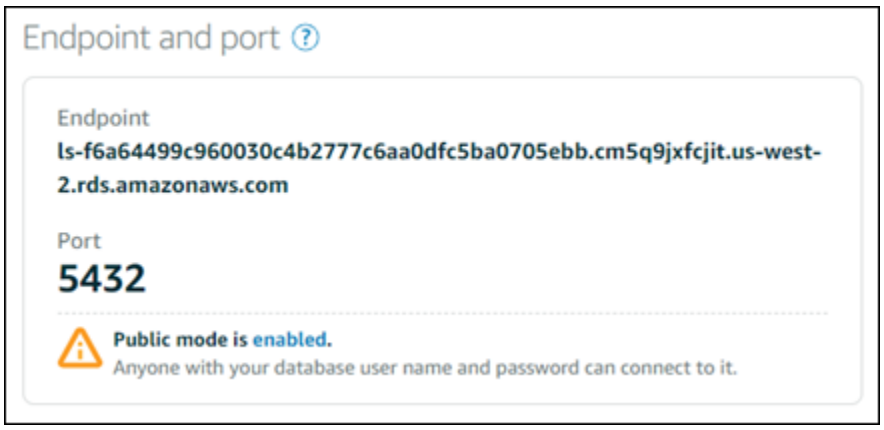
ステップ 1: PostgreSQL データベース接続の詳細を取得する

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 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

データベースを外部に接続するには、パブリックモードを有効にするか、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。

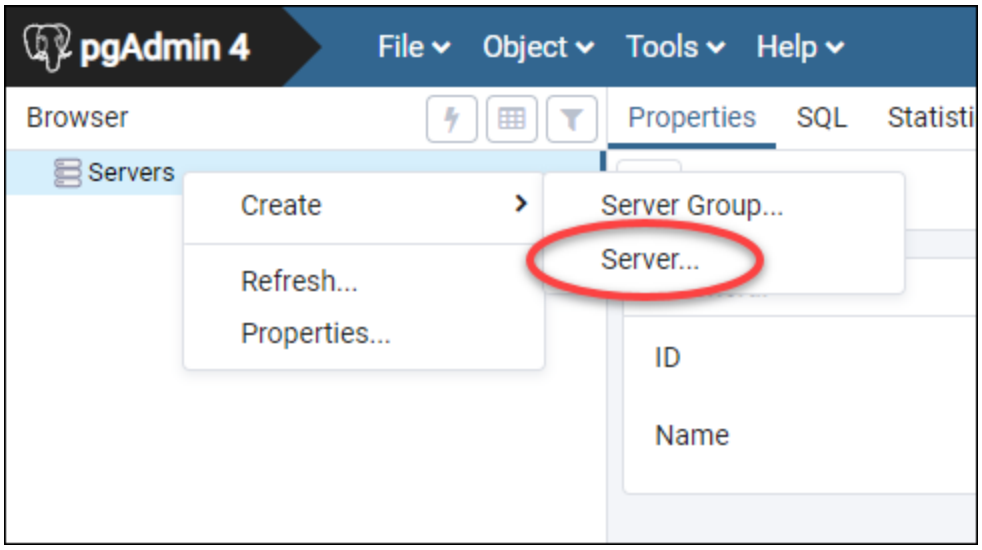
> **Note**
>
> データベースと同じリージョンにある Lightsail インスタンスのいずれかからデータベースに接続する場合は、ステップ 3 に進みます。

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

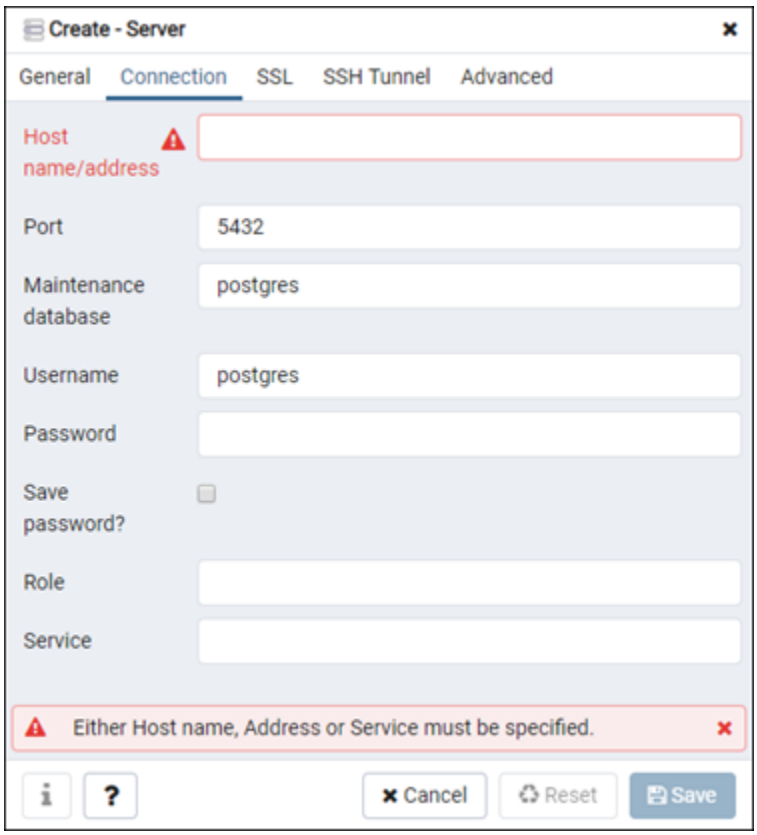
PostgreSQL データベースに接続するには、前に取得したエンドポイントとポートを使用するようにデータベースクライアントを設定します。以下のステップは、pgAdmin を設定する方法を示していますが、これらのステップは他のクライアントと同様の場合があります。

> **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。Lightsail コンソールからデータベースエンドポイントをコピーし、まだクリップボードにある場合は、Windows を使用している場合は Ctrl+V、macOS を使用している場合は Cmd+V 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 で 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名です。

  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。Lightsail コンソールからパスワードをコピーし、まだクリップボードにある場合は、Windows を使用している場合

は Ctrl+V、macOS を使用している場合は Cmd+V 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。

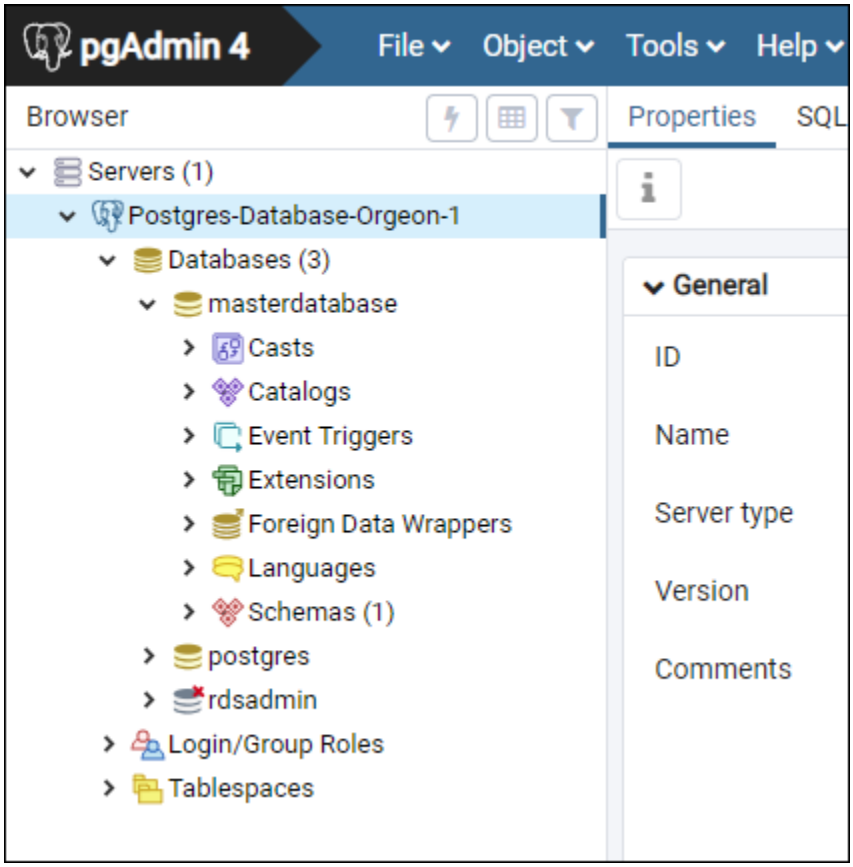
- ロールとサービス — これらのフィールドは空白のままにしておきます。

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

以下は、Lightsail のデータベースにデータをインポートするのに役立つガイドです。

- データを PostgreSQL データベースにインポートする

を使用して Lightsail PostgreSQL データベースに安全に接続する SSL

Amazon Lightsail はSSL証明書を作成し、プロビジョニング時に PostgreSQL (Postgres) マネージドデータベースにインストールします。証明書は認証局 (CA) によって署名され、なりすまし攻撃から保護するための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証明書は信頼できるルートエンティティであり、ほとんどの場合機能しますが、アプリケーションが証明書チェーンを受け入れない場合に失敗する可能性があります。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、サポートされている 、およびアプリケーションの中間証明書をダウンロードする方法の詳細については、 AWS リージョン「マネージドデータベースのSSL証明書をダウンロードする」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、PostgresSQL ウェブサイトの「Postgre ダウンロード」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する」を参照してください。

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、次の手順を実行しますSSL。

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、PostgreSQL データベースに接続します。

```
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* データベースのエンドポイントを使用する。
- *DatabaseName* 接続先のデータベースの名前。

- *UserName* データベースのユーザー名を入力します。
- */path/to/certificate/rds-combined-ca-bundle.pem* 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。

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。値が「接続SSL」と表示されると、接続は暗号化されます。

Lightsail データベースを削除して最終スナップショットを作成する

不要になった場合は、Amazon Lightsail でマネージドデータベースを削除します。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。

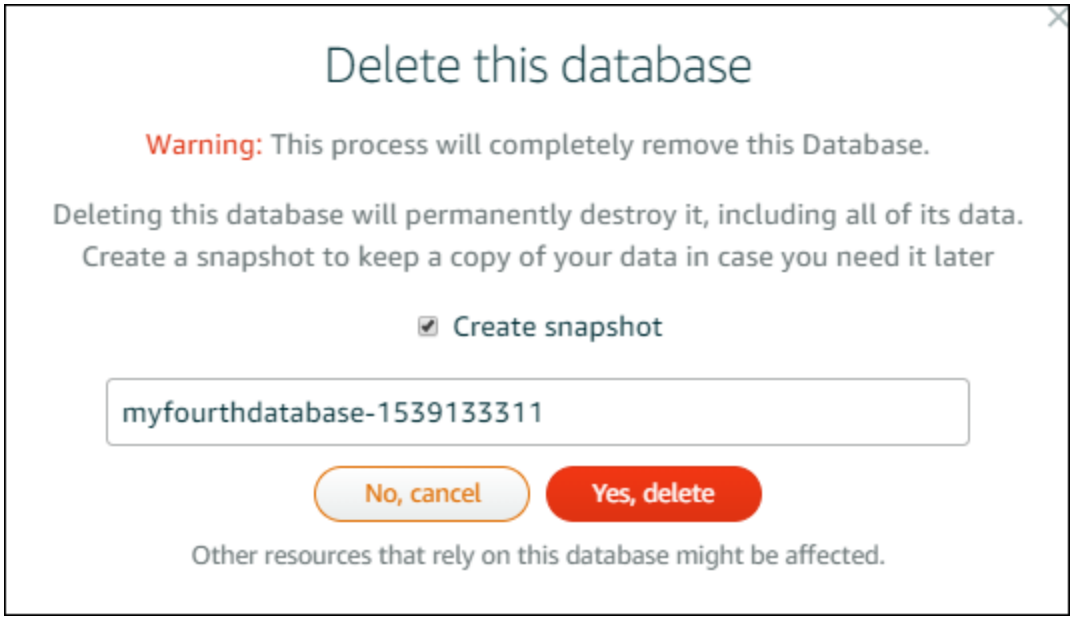
データベースを削除するには

1. Lightsail コンソール にサインインします。

2. Lightsail 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スナップショットを表示できます。

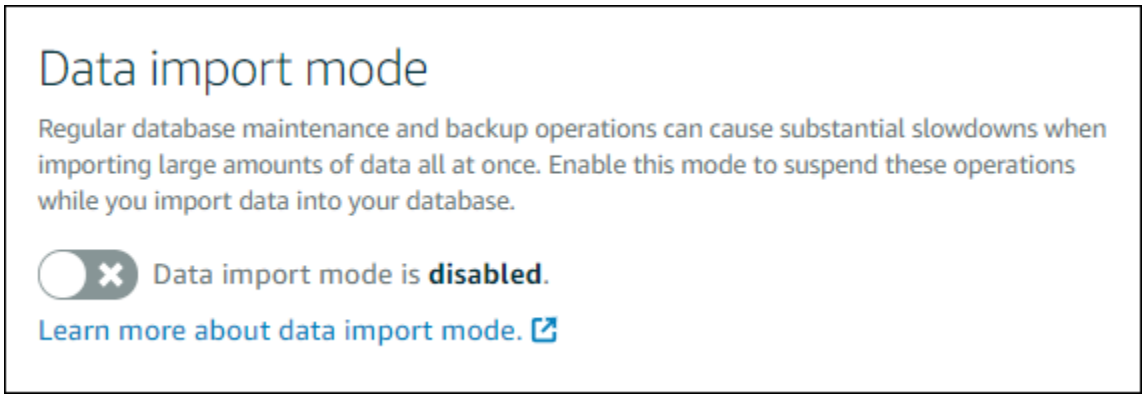
大きなデータセットを Lightsail データベースに遅延なくインポートする

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。Amazon Lightsail マネージドデータベースのデータインポートモードを有効にして、大量のデータをインポートしている間にこれらのオペレーションを一時停止します。

> **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 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。

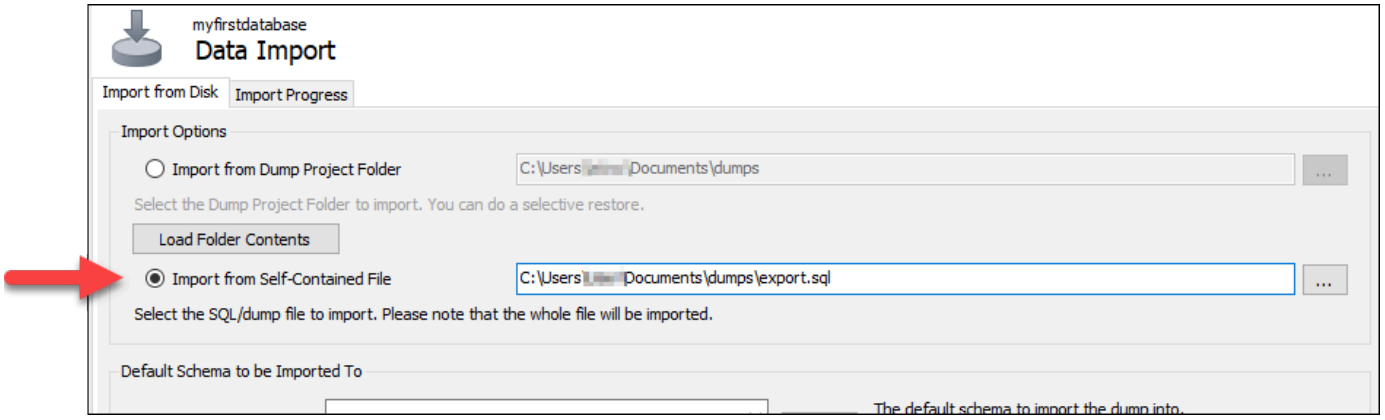
SQL データを Lightsail MySQL データベースにインポートする

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 (.MySQLをインポートできます。

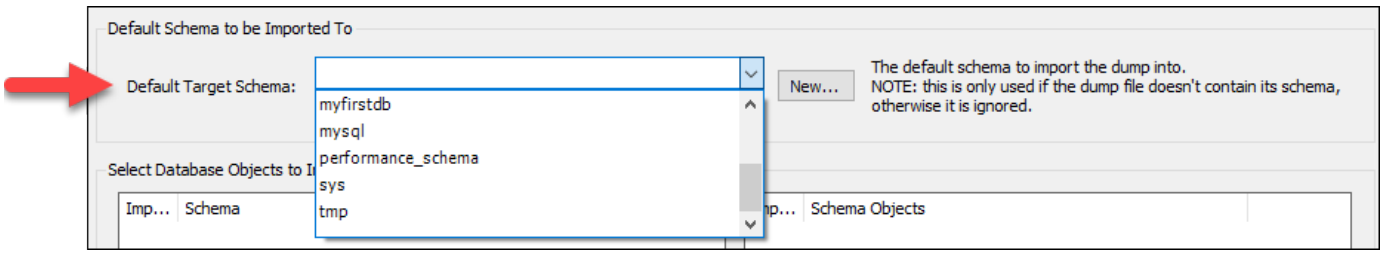
> **Note**
>
> MySQL Workbench をデータベースに接続する方法については、「MySQL データベースに接続する」を参照してください。

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。

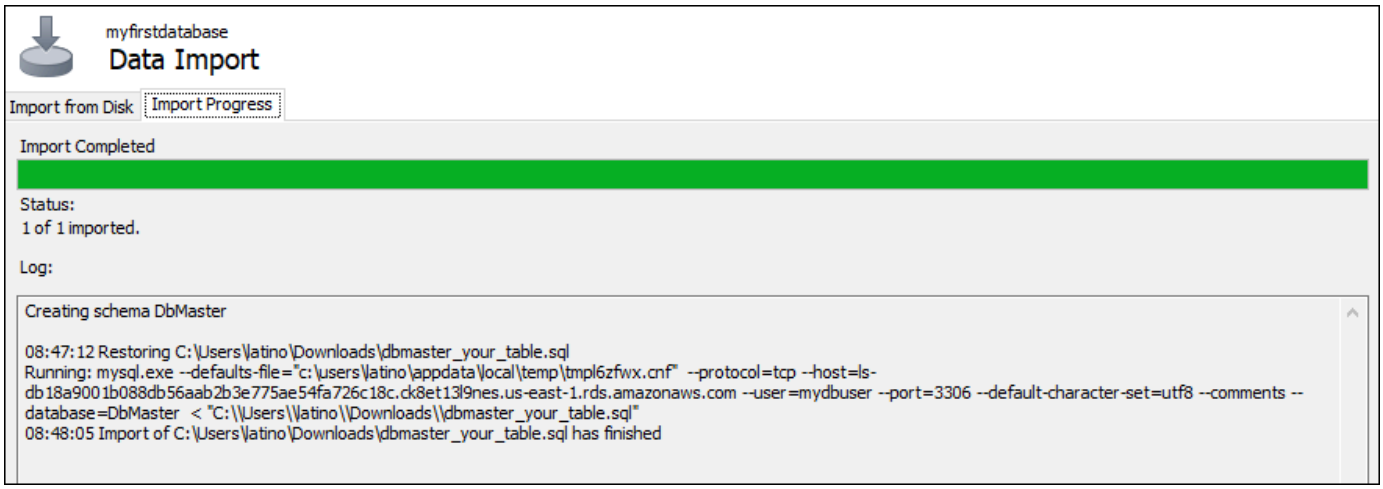


5. 省略記号ボタンを選択し、インポートする .SQL ファイルのローカルドライブを参照します。
6. インポートする .SQL ファイルを選択し、[Open (開く)] を選択します。
7. [Default Target Schema (デフォルトのターゲットスキーマ)] ドロップダウンメニューを選択し、ファイルをインポートする先の既存のデータベースを選択します。[New (新規)] を選択して新しいデータベースを作成することもできます。



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。



PostgreSQL データベースバックアップを Lightsail マネージドデータベースにインポートする

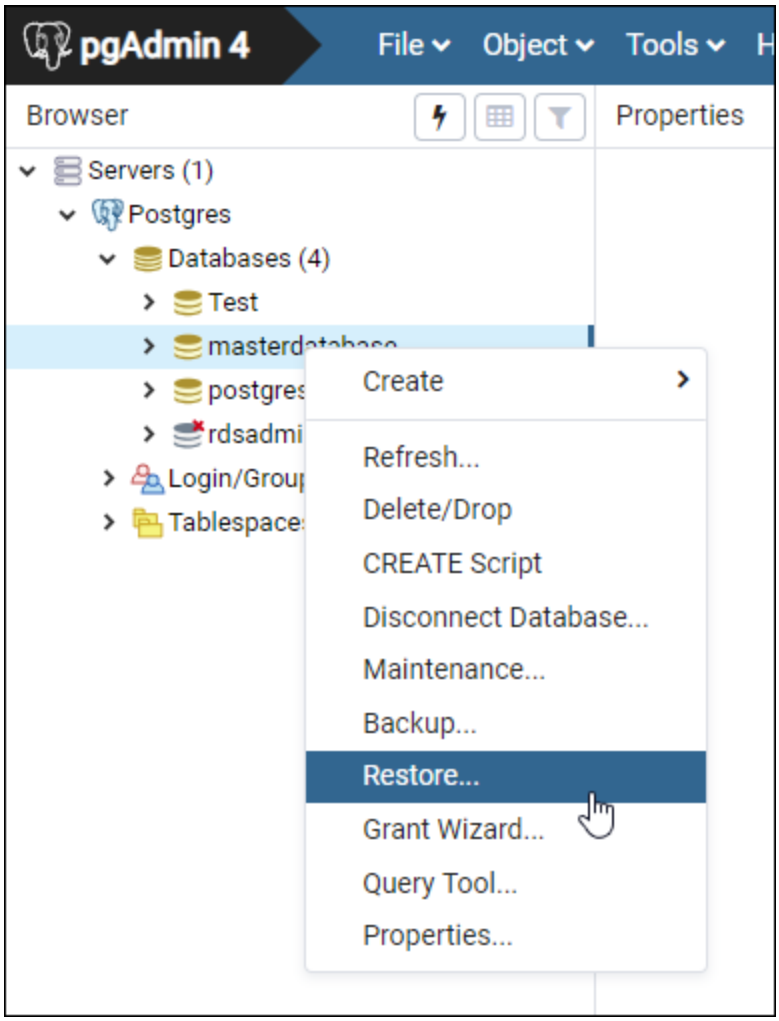
pgAdmin を使用して、Amazon Lightsail の PostgreSQL マネージドデータベースにデータベースバックアップファイルをインポートできます。

> **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。

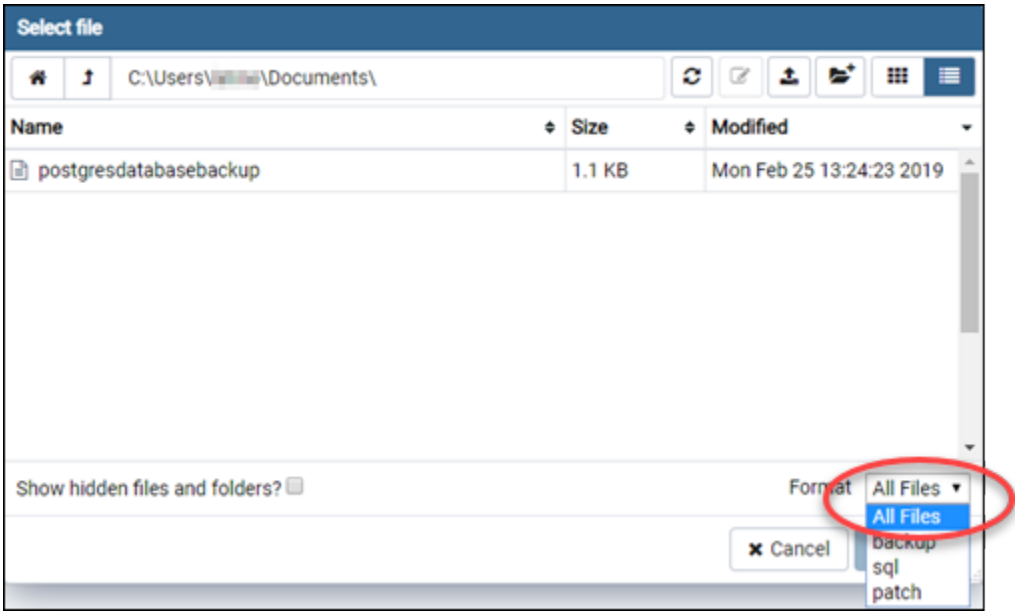


5. [Restore (復元)] フォームで、次のフィールドに入力します。

   - 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。

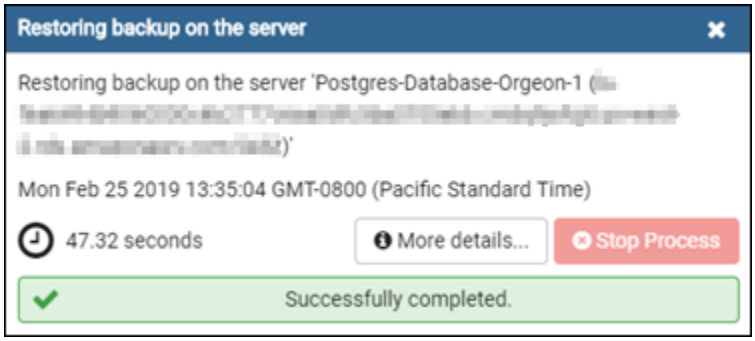
> **Note**
>
> [Format (形式)] ドロップダウンメニューを選択し、[All files (すべてのファイル)] を選択してローカルドライブにあるすべてのファイル形式を表示します。バックアップファイルは、デフォルトで選択されているファイル形式 (sql) とは異なる形式で保存されている場合があります。



- 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 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 の一般クエリログとスロークエリログを使用して MySQL クエリのパフォーマンスをモニタリングする

Lightsail の一般クエリログとスロークエリログを使用して MySQL クエリのパフォーマンスをモニタリングする

Amazon Lightsail の MySQL データベースでは、一般クエリ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、 を使用してデータベースパラメータ AWS CLI を更新し、一般クエリログとスロークエリログを有効にする方法について説明します。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 で動作する AWS Command Line Interface ように を設定する Amazon Lightsail」を参照してください。

Lightsail コンソールで一般的なクエリログとスロークエリログを有効にする

Lightsail コンソールで一般クエリ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すFILE。

Lightsail コンソールで一般的なクエリ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* データベースの名前を入力します。
   - データベース AWS リージョン の を持つ*#####*。
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* データベースの名前を入力します。
   - データベース AWS リージョン の を持つ*#####*。
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* データベースの名前を入力します。
- データベース AWS リージョン の を持つ*#####*。

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* データベースの名前を入力します。
- データベース AWS リージョン の を持つ*#####*。

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、データベースの一般的なクエリ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」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、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TABLE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、マイクロ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ログに記録されます。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ロ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ログに記録されます。

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ログファイルの削除が定期的に実行されます。これは、ロ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、ログ記録は次のように処理されます。

- FILE ログ記録が有効になっている場合、ログファイルの検査が 1 時間ごとに実行され、作成後 24 時間を超えた古いログファイルは削除されます。場合によっては、削除後の残りのロ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、ログファイルのサイズがしきい値以下になるまで、最も大きいログファイルから順に削除されます。
- TABLE ログ記録を有効化すると、24 時間ごとにロ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、すべてのログの合計サイズが 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、ロ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、すべてのログの合計サイズが 5 GB を超えると、ロ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、ロ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 point-in-time バックアップを無効にする

Lightsail マネージドデータベースの point-in-time バックアップを無効にするには、次の手順に従います。

> **Important**
>
> point-in-time バックアップを使用すると、データベースに障害が発生した場合に簡単にデータを復元できます。Lightsail マネージドデータベースでポイントインタイムバックアップを有効にしておくことをお勧めします。

前提条件

AWS Command Line Interface （ AWS CLI) または を使用して AWS CloudShell 、Lightsail データベースの point-in-time バックアップを有効または無効にします。 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。詳細については、「Lightsail オペレーション AWS CLI の をセットアップする 」および「で Lightsail リソースを管理する AWS CloudShell」を参照してください。

データベース point-in-timeバックアップを無効にする

Lightsail でマネージドデータベースの point-in-time バックアップを無効にするには、 の `update-relational-database` Lightsail コマンドを使用してデータベースを更新する必要があります AWS CLI。詳細については、update-relational-databaseAWS CLI コマンドリファレンスの「」を参照してください。

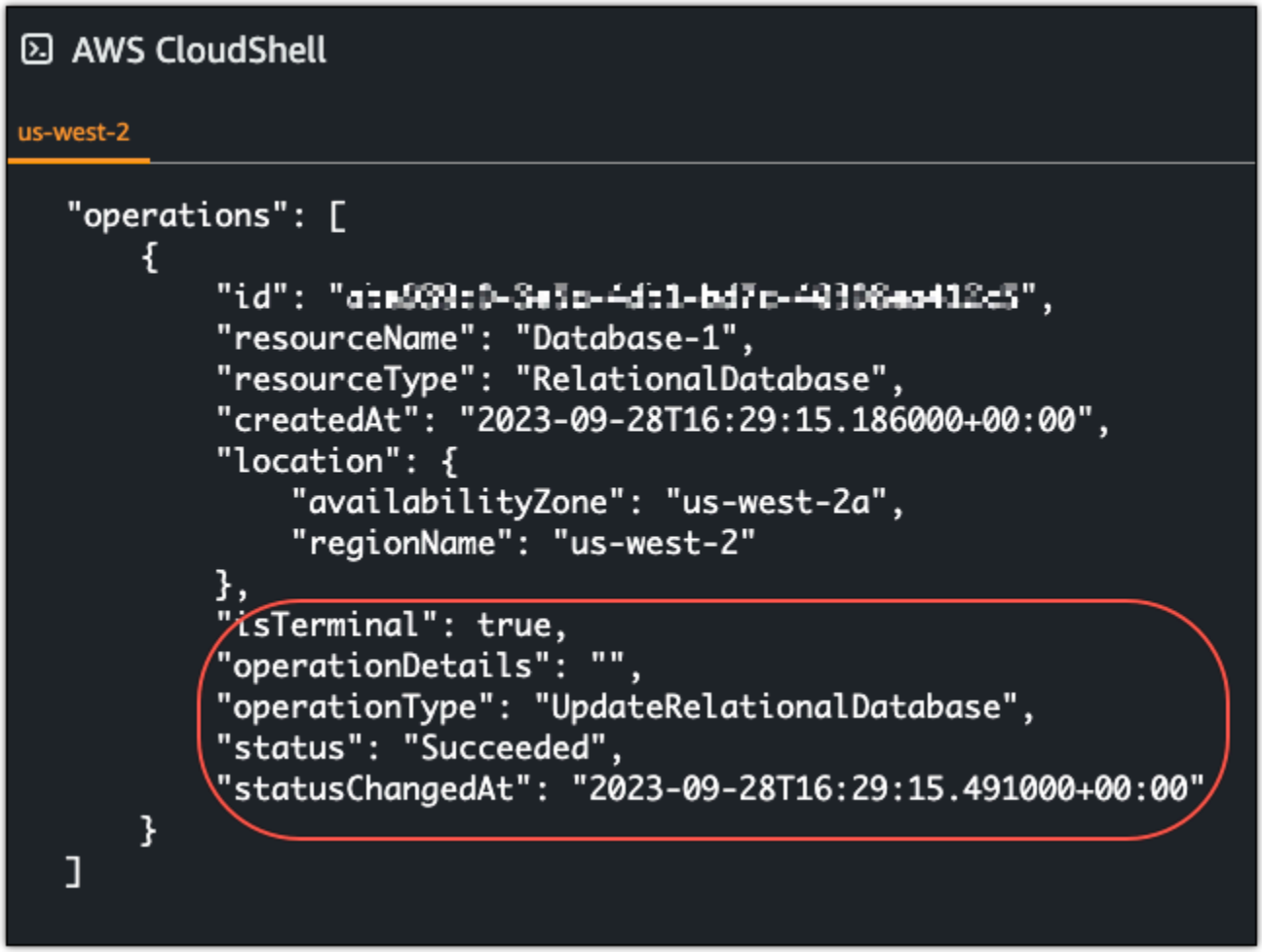
- ターミナル、コマンドプロンプト、または CloudShellウィンドウで次のコマンドを入力します。

  ```
  aws lightsail update-relational-database --region Region --relational-database-name DatabaseName --disable-backup-retention --apply-immediately
  ```

  コマンド`--disable-backup-retention`の値は、指定されたデータベースの point-in-time バックアップをオフにします。コマンドを、以下のように置き換えます。

  - *DatabaseName* データベースの名前を入力します。
  - データベース AWS リージョン の を持つ*#####*。

ステータスが のオペレーションレスポンスが表示されますSucceeded。データベースのステータスは、更新中、しばらくの間、 の変更 に変わります。データベースのステータスが使用可能 に戻ると、次の例に示すように point-in-time 復元オプションは無効になります。



> **Note**
>
> point-in-time バックアップを有効にするには、前にリストしたのと同じコマンドを実行しますが、代わりに `--enable-backup-retention`パラメータを使用します。

Lightsail データベースをスナップショットでバックアップする

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。

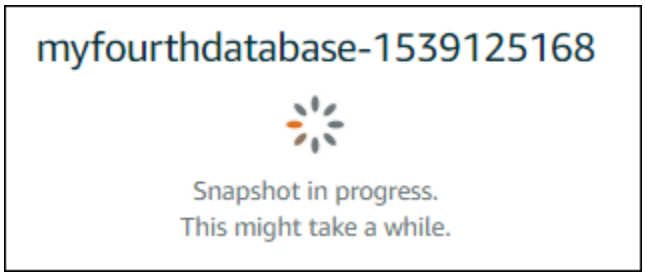
データベースのスナップショットを作成するには

1. Lightsail コンソール にサインインします。
2. Lightsail 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 ホームページのスナップショットタブで表示することもできます。

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

- Lightsail の point-in-time バックアップからデータベースを復元する
- Lightsail のスナップショットからマネージドデータベースを作成する

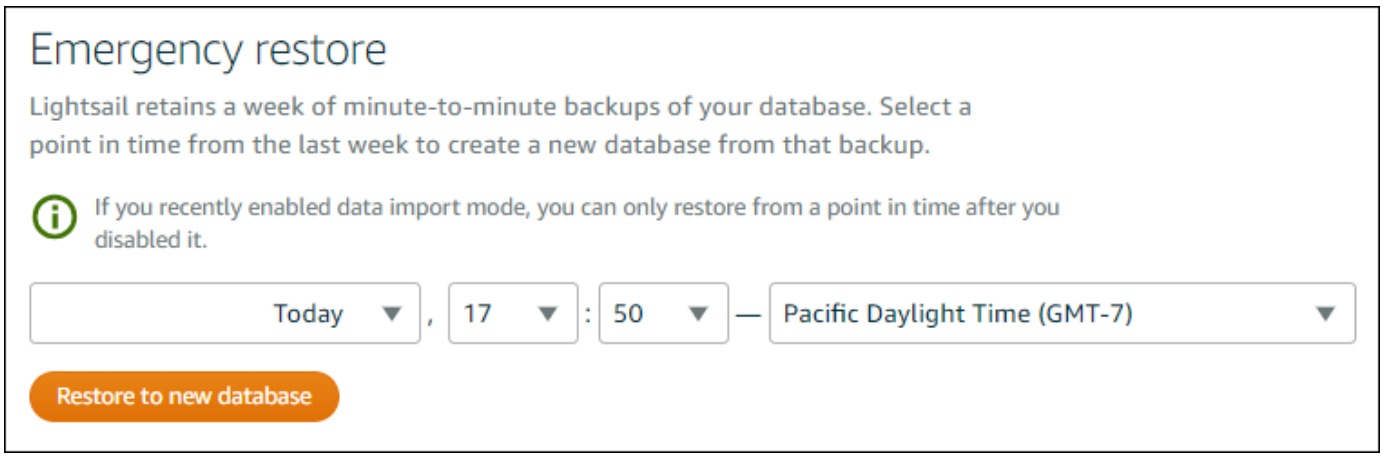
Lightsail の point-in-time バックアップからデータベースを復元する

Amazon Lightsail の point-in-time 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。データベースの P oint-in-time バックアップは 5 分単位で、過去 7 日間使用できます。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」を参照してください。

point-in-time 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 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。

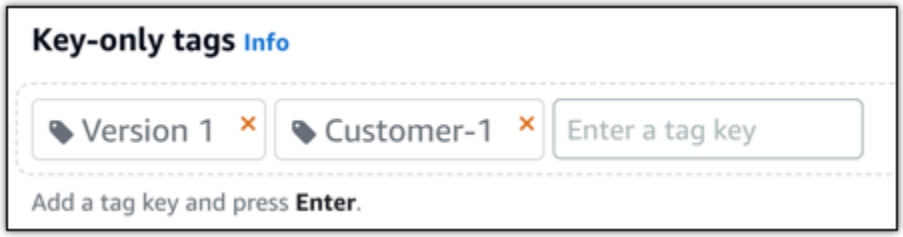
   > **Note**
   >
   > 元のデータベースプランより小さいデータベースプランを選択することはできません。

9. データベースの名前を入力します。

   リソース名:

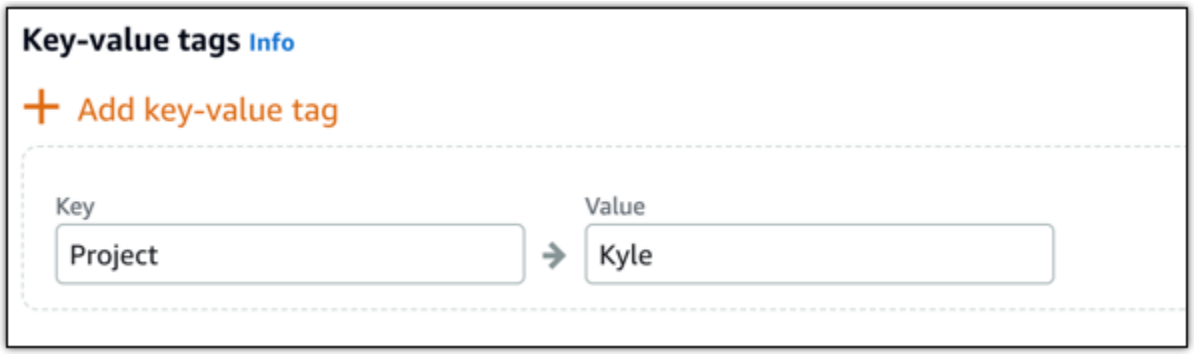
   - Lightsail アカウントの各 AWS リージョン 内で一意である必要があります。
   - 2〜255 文字を使用する必要があります。
   - 先頭と末尾は英数字または数字を使用する必要があります。
   - 英数字、数字、ピリオド、ダッシュ、アンダースコアを使用することができます。
10. 以下のいずれかのオプションを選択して、データベースにタグを追加します。

- [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 データベースは新しいデータベースプランまたはバンドルで準備が整います。

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。
- point-in-time バックアップから作成されたデータベースは、Lightsail によって作成された強力なパスワードを使用するように設定されています。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 のスナップショットからマネージドデータベースを作成する

元のデータベースで問題が発生した場合は、Amazon Lightsail の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。元のデータベースのバックアップから point-in-time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 の point-in-time バックアップからデータベースを作成する Amazon Lightsail」を参照してください。

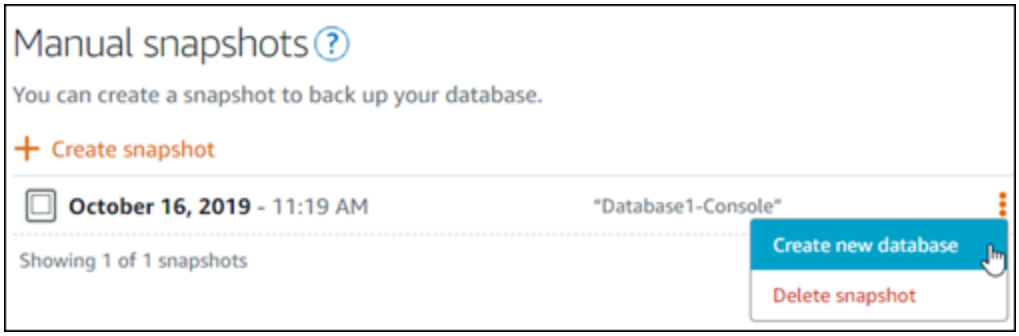
データベースを複製する際は、元のデータベースとは異なるプランやよりサイズの大きなプランを選択できます。ただし、元のデータベースよりサイズの小さいプランを選択することはできません。

> **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 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。

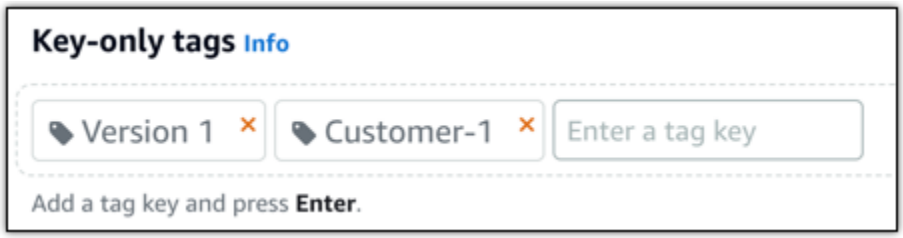
   リソース名:

   - 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。

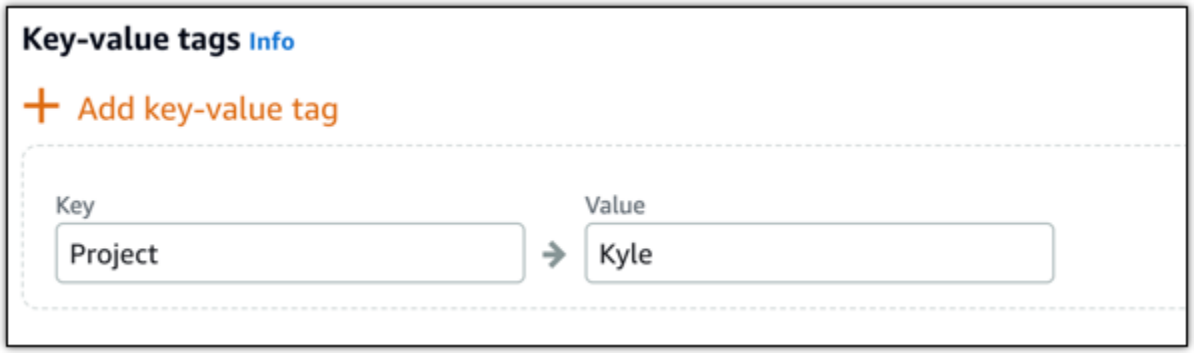
10. 以下のいずれかのオプションを選択して、データベースにタグを追加します。

    - [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 データベースは新しいデータベースプランまたはバンドルで準備が整います。

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されたデータベースは、Lightsail によって作成された強力なパスワードを使用するように設定されます。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 データベースへの安全なアプリケーション接続のための SSL/TLS 証明書のダウンロード

アプリケーションから 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) のラベルが付けられていますが、Lightsail のマネージドデータベースでも機能します。

すべての の証明書バンドル AWS リージョン

すべての 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 AWS リージョン、またはアプリケーションが Microsoft Windows にあり、PKCS7 ファイルが必要な場合は、Amazon Relational Database Service ユーザーガイドの「すべての の証明書バンドル AWS リージョン」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 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 AWS リージョン、「Amazon Relational Database Service ユーザーガイド」の AWS リージョン「特定の の証明書バンドル」を参照してください。 Amazon Relational Database Service

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/ 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TLS。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-databaseAPI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/ 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TLS、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 を使用して変更することができます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、、および rds-ca-2019証明書に更新されるまでrds-ca-rsa4096-g1、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

- この手順のステップを完了する前に、新しい SSL/TLS 証明書を使用するようにデータベースクライアントアプリケーションを更新します。

  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って異なります。アプリケーションデベロッパーと協力して、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 証明書のアプリケーションの更新の詳細については、Amazon 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 DBSQL インスタンスに接続する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アプリケーションの更新」を参照してください。
- 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、AWS CloudShell Lightsail の CloudShell「」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 証明書を識別します。

   ```
   aws lightsail get-relational-database --relational-database-name DatabaseName --
   region DatabaseRegion | grep "caCertificateIdentifier"
   ```

   コマンドで、*DatabaseName* 変更するデータベースの名前、および *DatabaseRegion* データベースインスタンス AWS リージョン がある を持つ 。

   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 で新しい CA 証明書 (、rds-ca-rsa2048-g1、および) のいずれかを使用するようにマネージドデータベースを変更するにはrds-ca-rsa4096-g1、次のステップを実行しますrds-ca-ecc384-g1。

> **Important**
>
> データベースの CA 証明書を更新する前に、CA 証明書を使用するクライアントアプリケーションを更新します。

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新しい証明書を使用します。

```
aws lightsail update-relational-database --relational-database-name DatabaseName --
region DatabaseRegion --ca-certificate-identifier rds-ca-rsa2048-g1
```

コマンドで、*DatabaseName* 変更するデータベースの名前、および *DatabaseRegion* データベースインスタンス AWS リージョン がある を持つ 。

例

```
aws lightsail update-relational-database --relational-database-name Database-1 --
region us-east-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。コマンドの最後に --apply-immediatelyパラメータを追加すると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 で古い CA 証明書 () を使用するようにマネージドデータベースを変更するには、次のステップを実行します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ます。

> **Important**
>
> データベースの CA 証明書を更新する前に、CA 証明書を使用するクライアントアプリケーションを更新します。

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. 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 --
   region DatabaseRegion --ca-certificate-identifier rds-ca-2019
   ```

   コマンドで、*DatabaseName* 変更するデータベースの名前、および *DatabaseRegion* データベースインスタンス AWS リージョン がある を持つ 。

   例

   ```
   aws lightsail update-relational-database --relational-database-name Database-1 --
   region us-east-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。コマンドの最後に --apply-immediatelyパラメータを追加するとすぐに更新されます。

Lightsail データベースのメンテナンスとバックアップをスケジュールする

Amazon Lightsail でデータベースの新しいバージョンがサポートされている場合、既存のマネージドデータベースをデータベース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アップグレードのみを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。推奨されるメンテナンスウィンドウは、 ごとに 8 時間の時間ブロックからランダムに選択された 30 分のウィンドウです AWS リージョン。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。優先バックアップウィンドウは、 ごとに 8 時間の時間ブロックからランダムに選択された 30 分のウィンドウです AWS リージョン。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

AWS Command Line Interface （ AWS CLI) を使用して、データベースの優先メンテナンスウィンドウとバックアップウィンドウを変更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。

- のインストール AWS CLI — 詳細については、AWS CLI「I のインストール」を参照してください。
- の設定 AWS CLI — 詳細については、「 の設定 AWS CLI」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

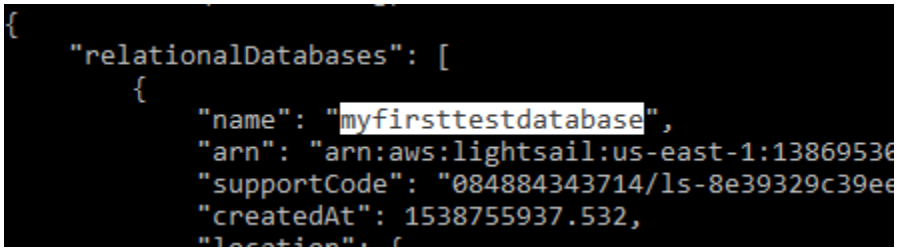
1. ターミナルまたはコマンドプロントウィンドウを開きます。
2. 次のコマンドを入力して、メンテナンスウィンドウを変更するデータベースの名前を取得します。

```
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、データベース AWS リージョン が配置されている に対して 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。

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* をデータベースの名前に置き換えます。
     - *MaintenanceWindow* 新しいメンテナンスウィンドウの時間枠を使用します。

       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* をデータベースの名前に置き換えます。
     - *BackupWindow* 新しいバックアップウィンドウの時間枠を使用します。

       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。

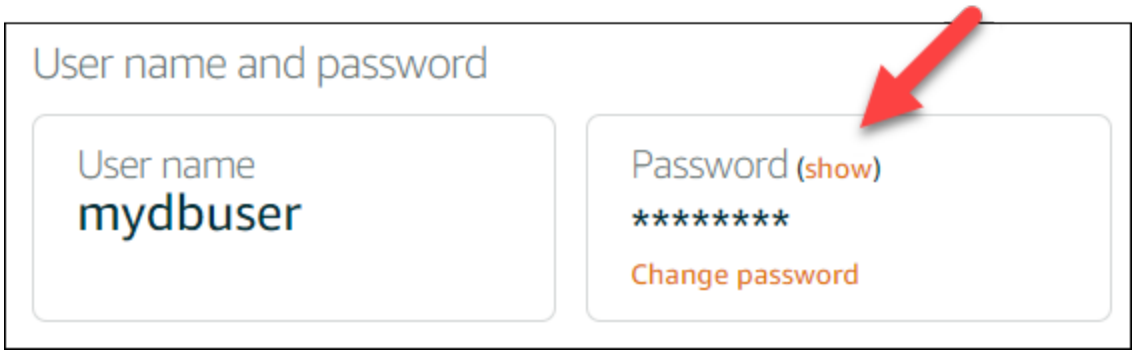
- データベースのデータのインポートモードを設定する
- データベースのパブリックモードを設定する
- データベースのパスワードを管理する
- MySQL データベースに接続する
- PostgreSQL データベースに接続する
- MySQL データベースにデータをインポートする
- データを PostgreSQL データベースにインポートする
- データベースのスナップショットを作成する

Lightsail データベースのパスワードを変更する

Amazon Lightsail で新しいデータベースを作成するときに、Lightsail に強力なパスワードを作成させるか、独自のパスワードを指定できます。Lightsail コンソールでは、現在のデータベースパスワードをいつでも表示または変更できます。

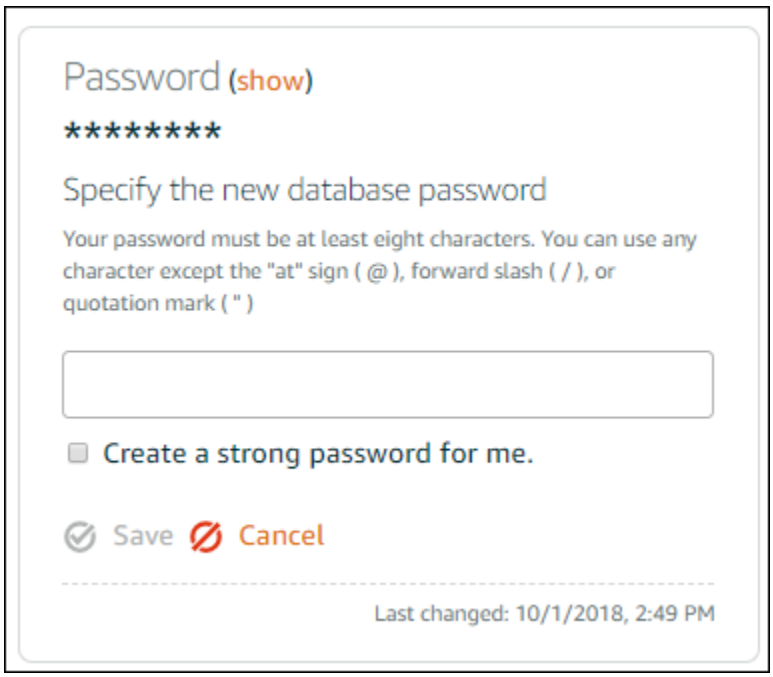
データベースのパスワードを管理するには

1. Lightsail コンソール にサインインします。
2. Lightsail 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 に強力なパスワードを作成するか、テキストボックスに独自のパスワードを入力することもできます。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 がパスワードを作成した場合、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 でデータベースを管理するのに役立つガイドをいくつか紹介します。

- 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

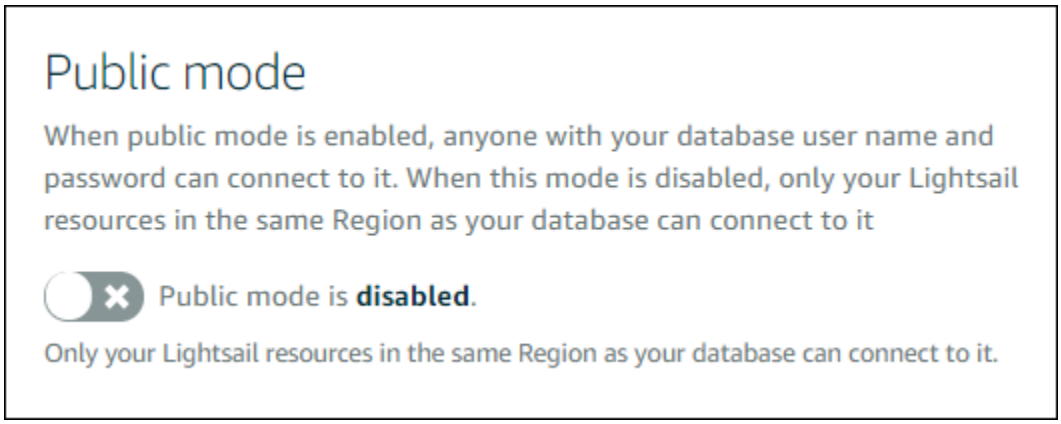
Lightsail データベースのパブリックアクセスを設定する

Amazon Lightsail のマネージドデータベースには、同じ Lightsail アカウントにある Lightsail リソース (インスタンス、ロードバランサーなど) のみがアクセスできます。一般的なシナリオの 1 つは、一般公開されているウェブアプリケーションとパブリックにアクセスできない Lightsail データベースの両方を使用して Lightsail インスタンスを作成し、その 2 つを接続することです。

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 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

パラメータの更新による Lightsail データベースのパフォーマンスの最適化

データベースシステム変数とも呼ばれるデータベースパラメータは、Amazon Lightsail のマネージドデータベースの基本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、マネージドデータベースのパラメータのリストを取得する方法と、 AWS Command Line Interface () を使用してパラメータを更新する方法について説明しますAWS CLI。

> **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 で動作する AWS CLI ように 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

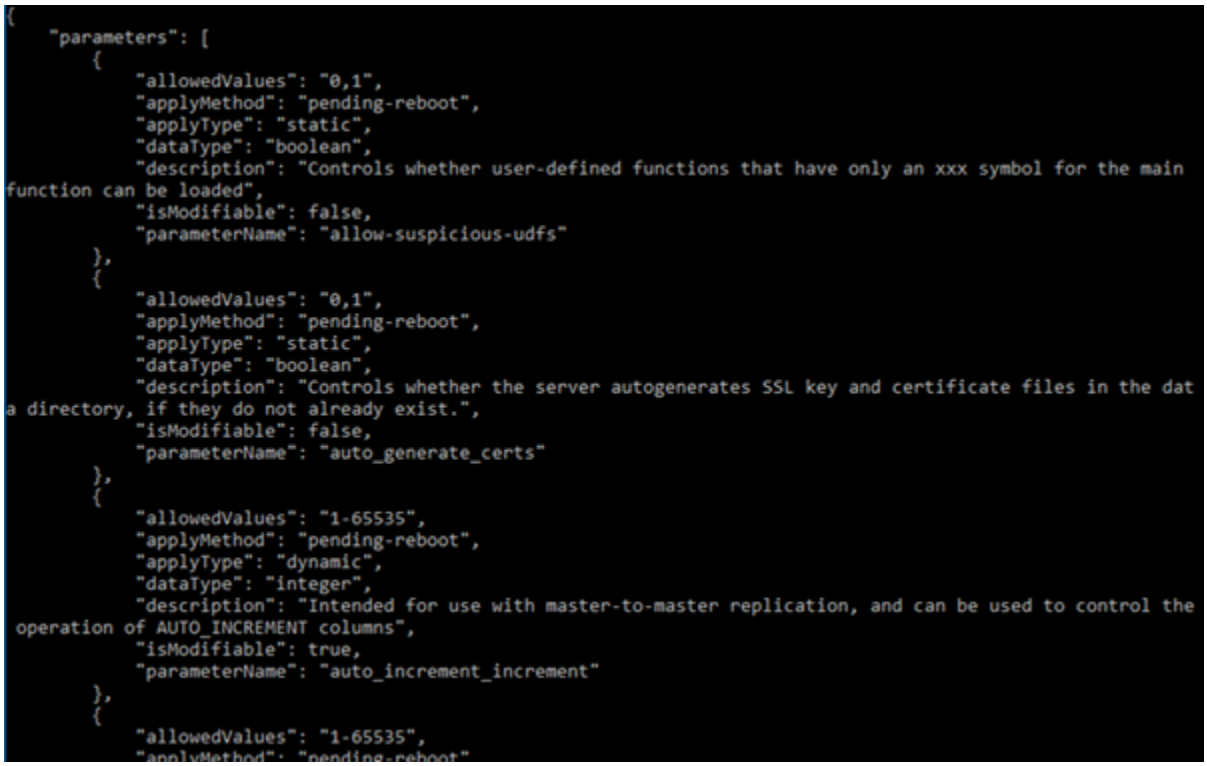
1. ターミナルまたはコマンドプロントウィンドウを開きます。

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、 をデータベースの名前*DatabaseName*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* データベースの名前を入力します。
- 次のページのトークン *NextPageTokenID* を含む ID。

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取得したら、次のステップを実行して Lightsail で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* データベースの名前を入力します。
- *ParameterName* は、変更するパラメータの名前で指定します。
- *NewParameterValue* パラメータの新しい値。
- *ApplyMethod* パラメータの apply メソッドを使用します。

  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。のセットアップと使用方法の詳細については、AWS CloudShell Lightsail の CloudShell「」を参照してください。

変更を理解する

メジャーバージョンのアップグレードでは、以前のバージョンとの非互換性がいくつか発生す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 ウェブサイトで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。

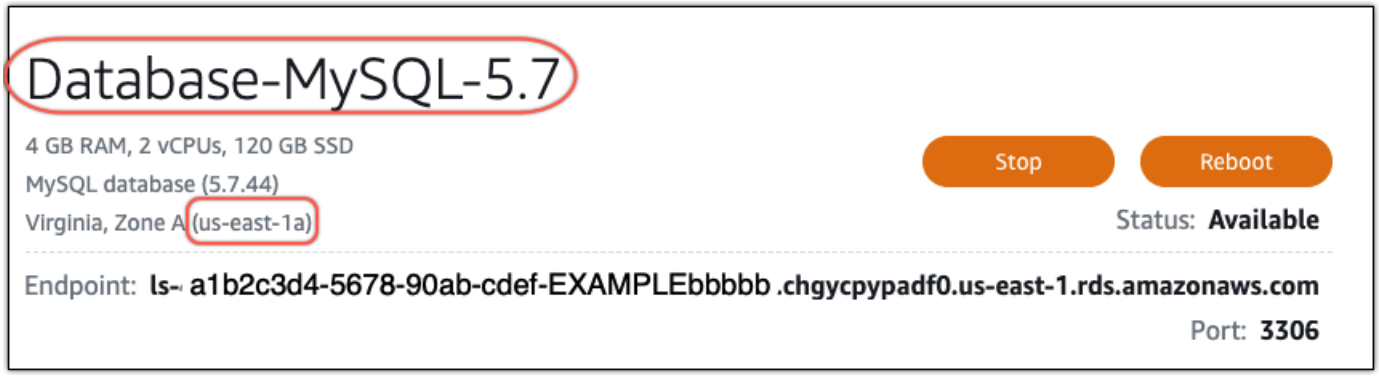
2. 変更を行う前に、データベースインスタンスのスナップショットを作成することをお勧めします。詳細については、「Lightsail データベースのスナップショットを作成する」を参照してください。
3. (オプション) 作成したスナップショットから新しいデータベースインスタンスを作成します。データベースの更新にはダウンタイムが必要なため、現在アクティブなデータベースをアップグレードする前に、新しいデータベースでアップグレードをテストできます。データベースのコピー作成の詳細については、「Lightsail データベースのスナップショットを作成する」を参照してください。

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、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. 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 のメモ。



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 のメモ。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期間中に行われます。コマンドで、 をデータベースの名前*DatabaseName*に置き換え、*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*に置き換え、 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。次の例のようなレスポンスが表示され、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の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。

Lightsail で MySQL 5.6 データベースから新しい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する。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

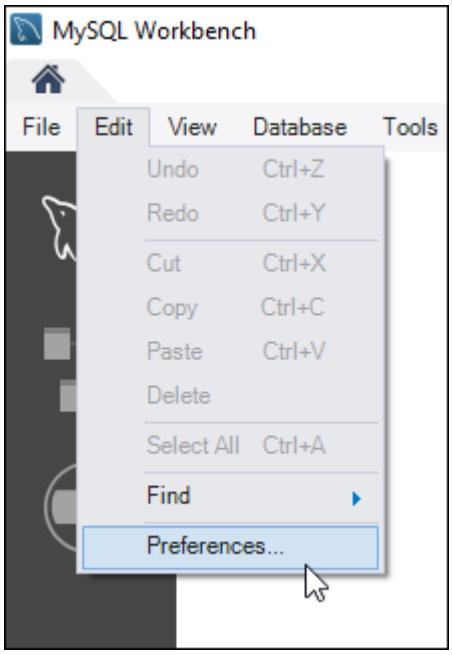
チュートリアルのこのセクションでは、MySQL 5.6 データベースに接続し、MySQL Workbench を使用してそのデータベースからデータをエクスポートします。MySQL Workbench を使用してデータをエクスポートする方法の詳細については、MySQL Workbench マニュアルの「SQL Data のエクスポートとインポートウィザード」を参照してください。

1. MySQL Workbench を使用して MySQL 5.6 データベースに接続する。

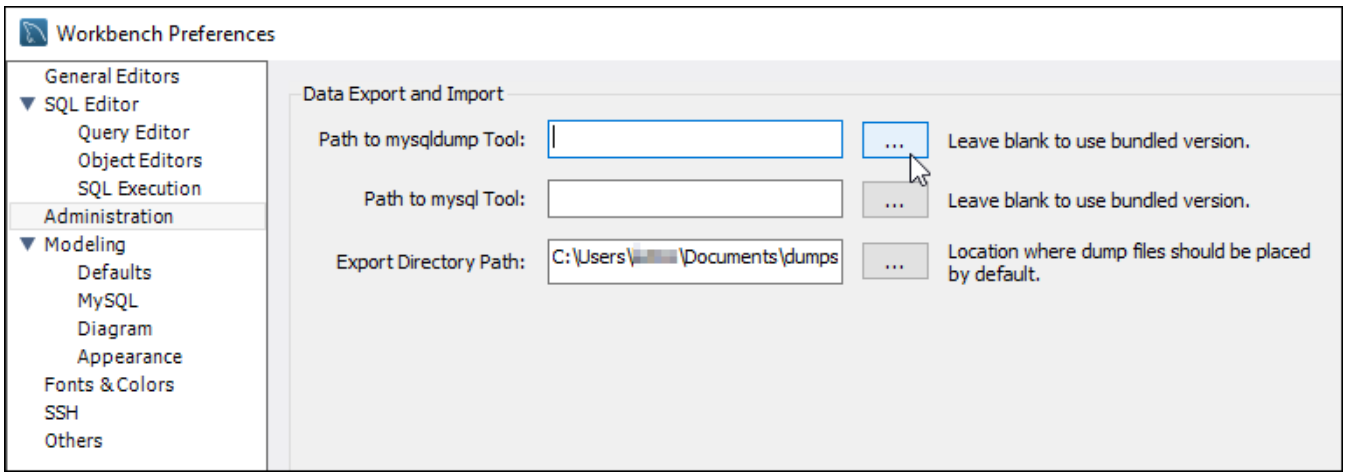
   MySQL ワークベンチでは、mysqldump を使用してデータをエクスポートします。MySQL Workbench で使用される mysqldump のバージョンは、データをエクスポートする MySQL データベースのバージョンと同じ (またはそれ以降) である必要があります。たとえば、MySQL 5.6.51 データベースからデータをエクスポートする場合は、バージョン 5.6.51 かそれ以降の mysqldump を使用する必要があります。正しいバージョンの mysqldump を使用しているか確認するために、ローカルコンピュータに適切なバージョンの MySQL サーバーをダウンロード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。

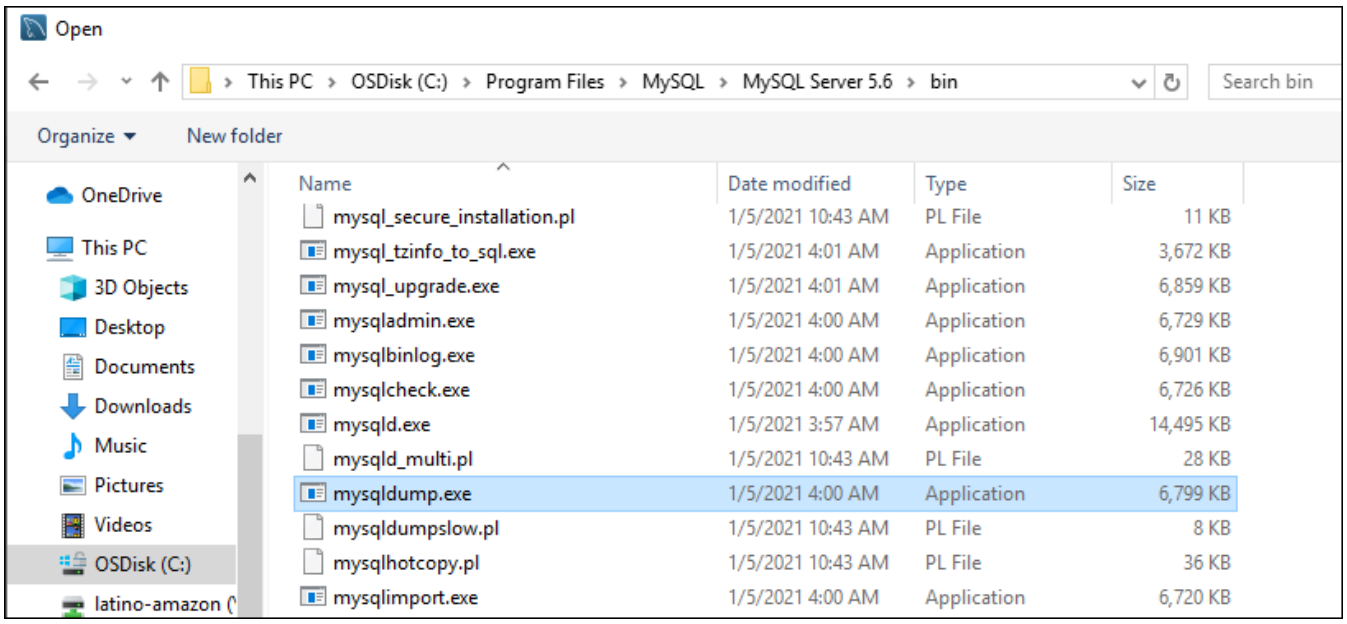
      

   2. ナビゲーションペインで [Administration] (管理) を選択します。
   3. Workbench Preferences ウィンドウが表示されたら、Path to mysqldump Tool のテキストボックスの横にある省略記号ボタンを選択します。

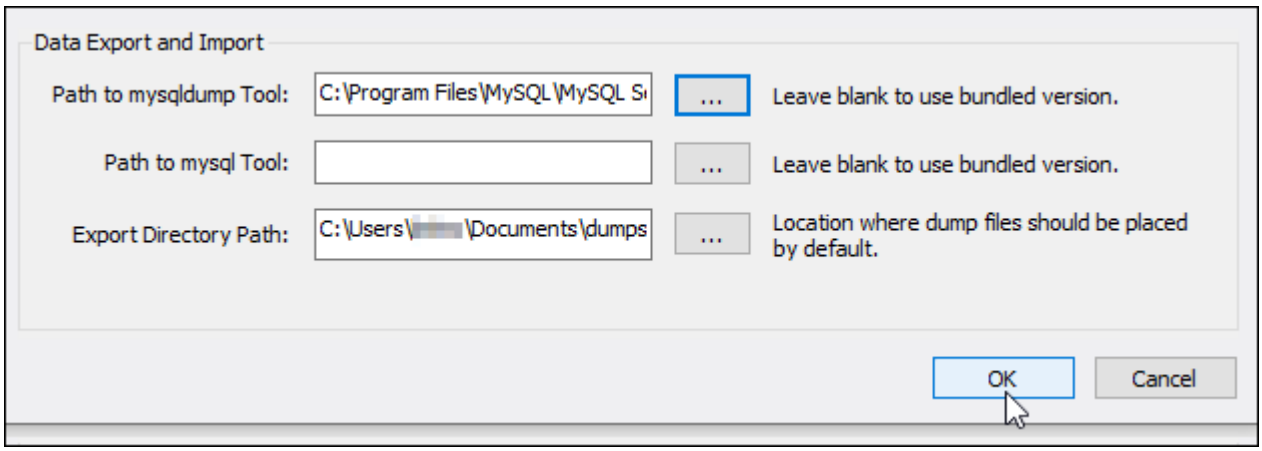


4. 適切なmysqldump 実行可能ファイルの場所まで移動したら、ダブルクリックします。

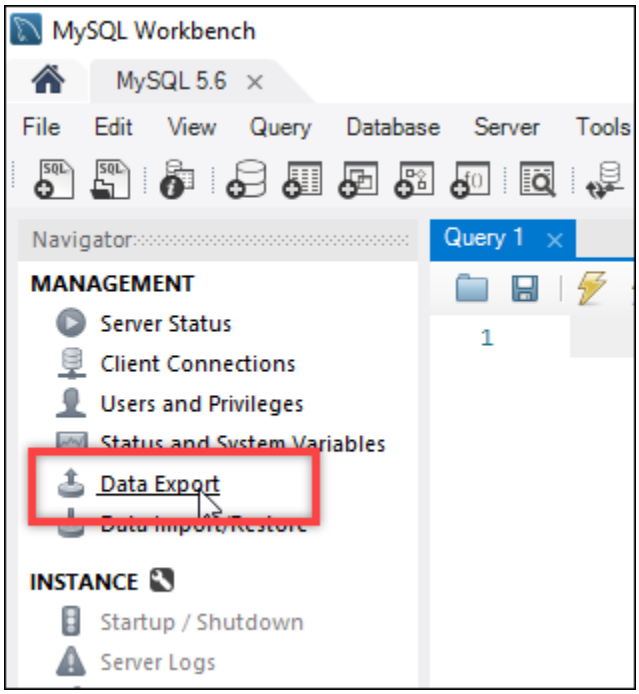
   Windows の場合、mysqldump.exe ファイルは通常 C:\Program Files\MySQL \MySQL Server 5.6\bin ディレクトリにあります。Linux の場合、ターミナルに which mysqldump を入力して mysqldump ファイルの位置を確認します。



5. Workbench Preferences ウィンドウで [OK] を選択します。

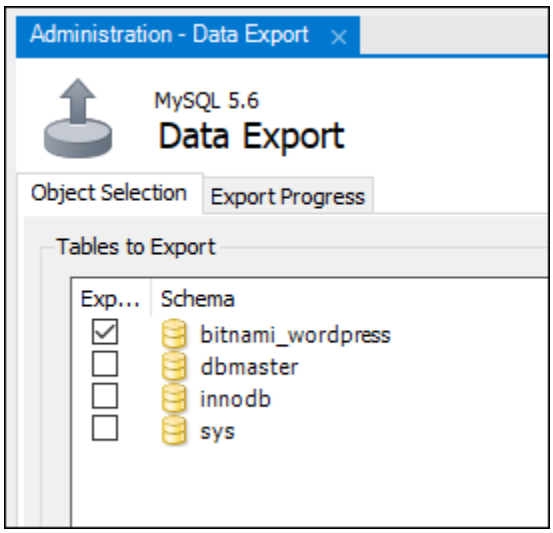


2. [Navigator] (ナビゲーター) ペインで [Data Export] (データエクスポート) を選択します。

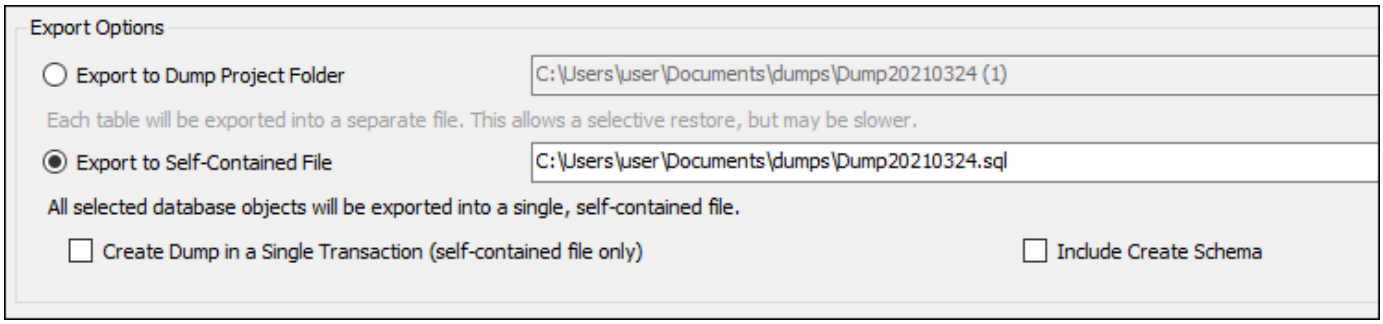


3. [データのエクスポート] タブが表示されたら、エクスポートするテーブルの横にチェックマークを追加します。

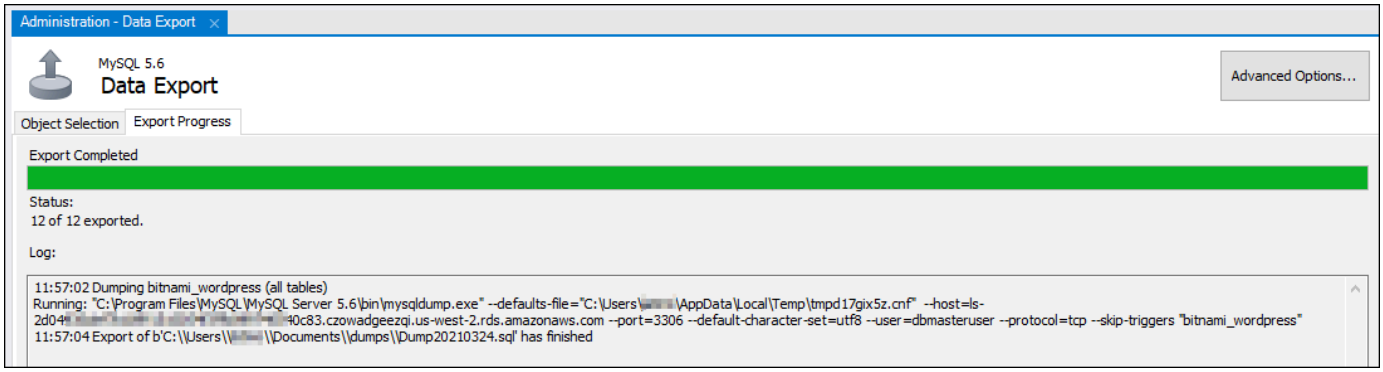
> **Note**
>
> この例では、「Certified by Bitnami WordPress 」インスタンス WordPress のウェブサイトのデータを含む`bitnami_wordpress`テーブルを選択しました。



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。



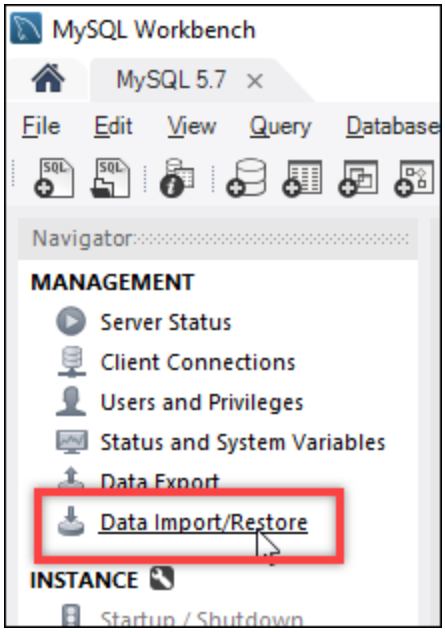
5. [Start Export] (エクスポートの開始) を選択します。
6. このチュートリアルの次のセクションに進む前に、エクスポートが完了するのを待ちます。



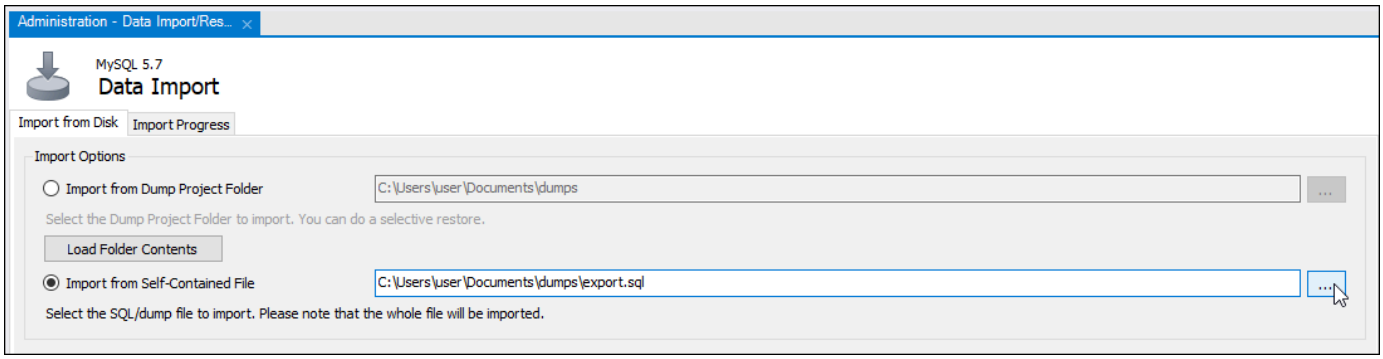
ステップ 4: MySQL 5.7 データベースに接続してデータをインポートする

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。

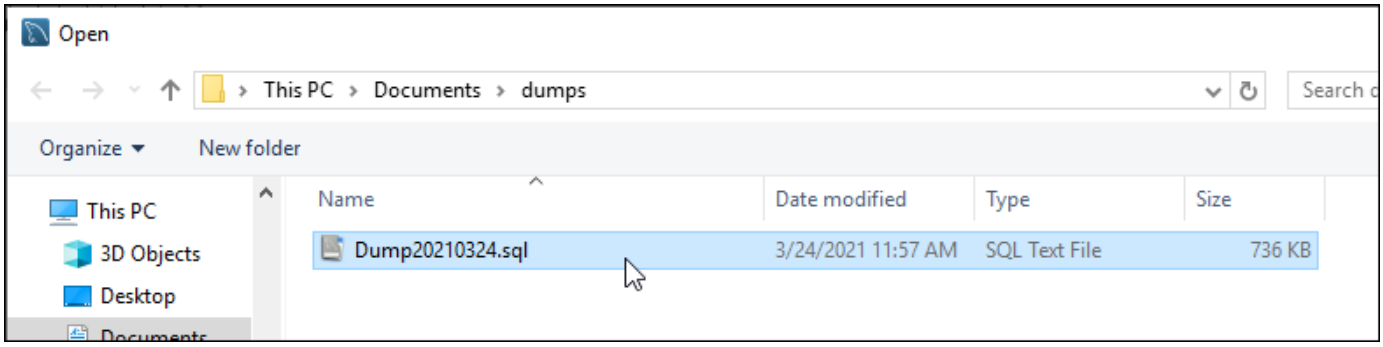
1. ローカルコンピュータの MySQL Workbench を使用して MySQL 5.7 データベースに接続する。
2. [Navigator] (ナビゲーター) ペインの [Data Import/Restore] (データのインポート/復元) を選択します。



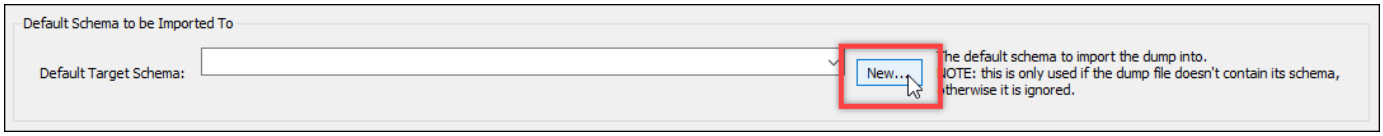
3. [Data Import] (データのインポート) タブが表示されたら、[Export to Self-Contained File] (自己完結型ファイルにエクスポート) を選択して、テキストボックスの横にある省略記号ボタンを選択します。



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。

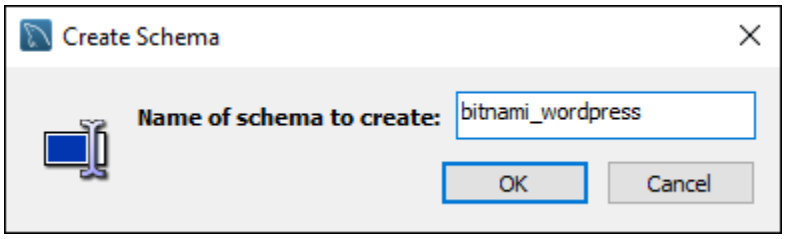


5. [Default Schema to be imported To] (インポート先のデフォルトスキーマ) のセクションで、[New] (新規) を選択します。

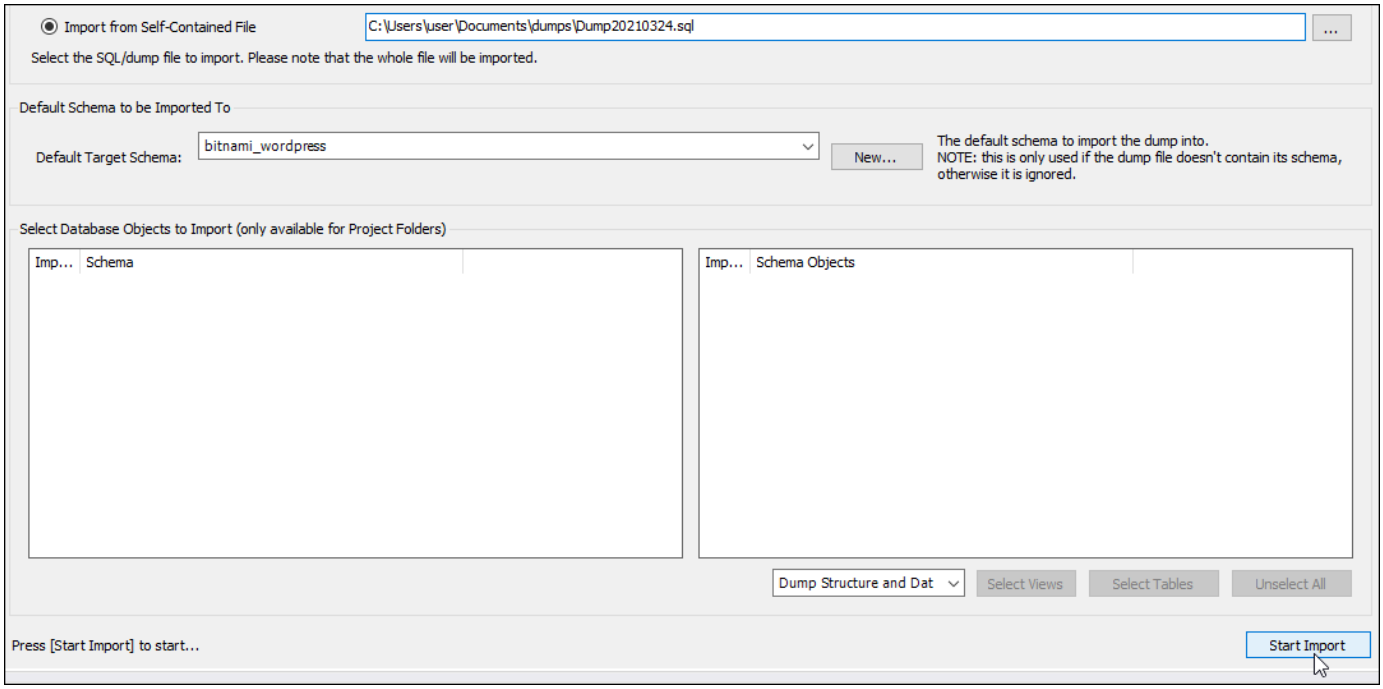


6. [Create Schema] (スキーマの作成) ウィンドウが表示されたら、スキーマの名前を入力します。

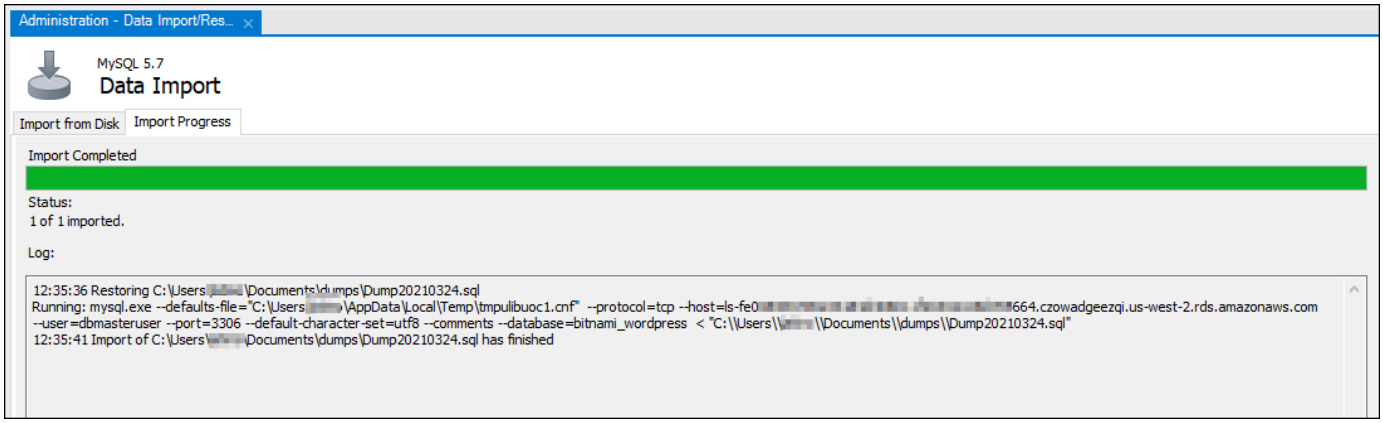
> **Note**
>
> この例では、エクスポートされたデータベーステーブルの名前が `bitnami_wordpress` であるため、それを入力しています。



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 ロードバランサーを使用してウェブトラフィックを分散する

Lightsail ロードバランサーは、受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内の複数の Lightsail インスタンスに分散します。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全体的なフロー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インスタンスを追加および削除できます。

Lightsail ロードバランシングでは、DNSホスト名を作成し、このホスト名に送信されたリクエストをターゲット Lightsail インスタンスのプールにルーティングします。Lightsail アカウントのインスタンスの合計数のクォータの範囲内であれば、ロードバランサーにターゲットインスタンスをいくつでも追加できます。

ロードバランサーの機能

Lightsail ロードバランサーには次の機能があります。

- 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暗号化を有効にします。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できます。詳細については、SSL「/TLS certificates」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リダイレクトへ — から HTTPへのHTTPSリダイレクトを有効にすると、HTTPS暗号化された接続にHTTPリクエストが自動的にリダイレクトされます。詳細については、「ロードバランサーのHTTPSリダイレクトHTTPを設定する」を参照してください。
  - TLS セキュリティポリシー — ロードバランサーに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 ロードバランサーのセキュリティTLS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された Lightsail インスタンスでショッピングカートを使用して Magento e コマースアプリケーションを実行している場合があります。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た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、まず、すべてのターゲットインスタンスが読み書きするための一元化された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、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できます。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

Lightsail コンソール、 AWS Command Line Interface （ AWS CLI ）、または Lightsail を使用してロードバランサーを作成できますAPI。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済みインスタンスをアタッチしたら、次のトピックHTTPSを使用して を有効にできます。詳細については、「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。

Lightsail ロードバランサーを使用してウェブトラフィックを分散する

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。ロードバランサーを作成したら、バランスを取る 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に Lightsail インスタンスを準備していることを確認してください。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. [ロードバランサーを作成] を選択します。
4. AWS リージョン ロードバランサーが作成される 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、ロードバランサーはHTTPリクエストを受け入れるためにポート 80 を開いた状態で作成されます。ロードバランサーを作成したら、SSL/TLS 証明書を作成して を設定できますHTTPS。詳細については、「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。

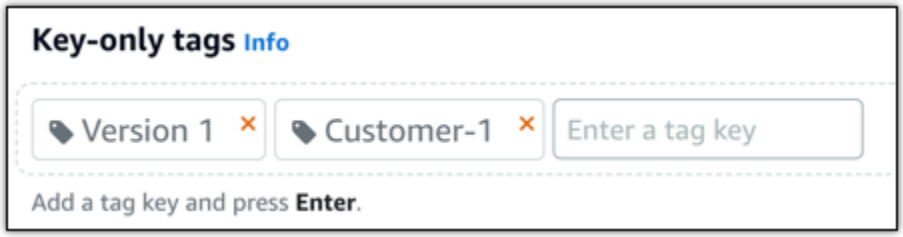
5. ロードバランサーの名前を入力します。

   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。

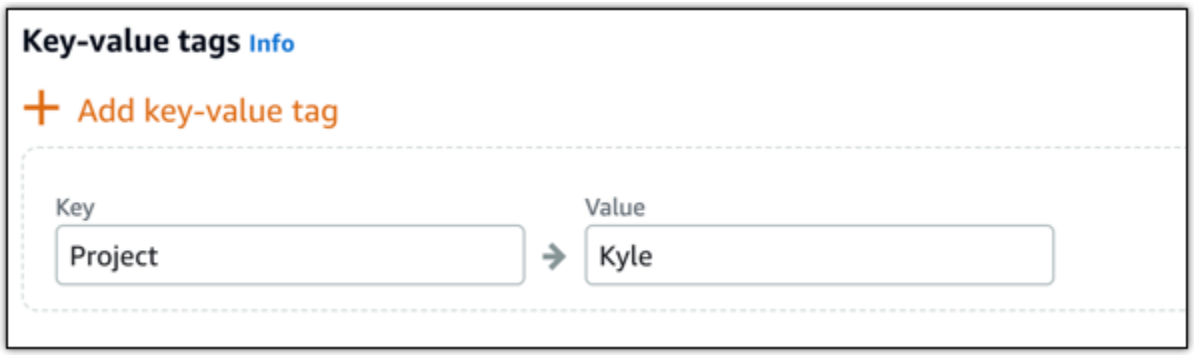
6. 以下のいずれかのオプションを選択して、ロードバランサーにタグを追加します。

   - [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. [ロードバランサーを作成] 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

ロードバランサーが作成されると、Lightsail はロードバランサーの管理ページに移動します。そのページを再度見つける必要がある場合は、Lightsail ホームページのネットワークタブを選択し、Lightsail ロードバランサーの名前を選択して管理します。

> **Note**
>
> Lightsail インスタンスをロードバランサーに正常にアタッチする前に、インスタンスが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. 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。

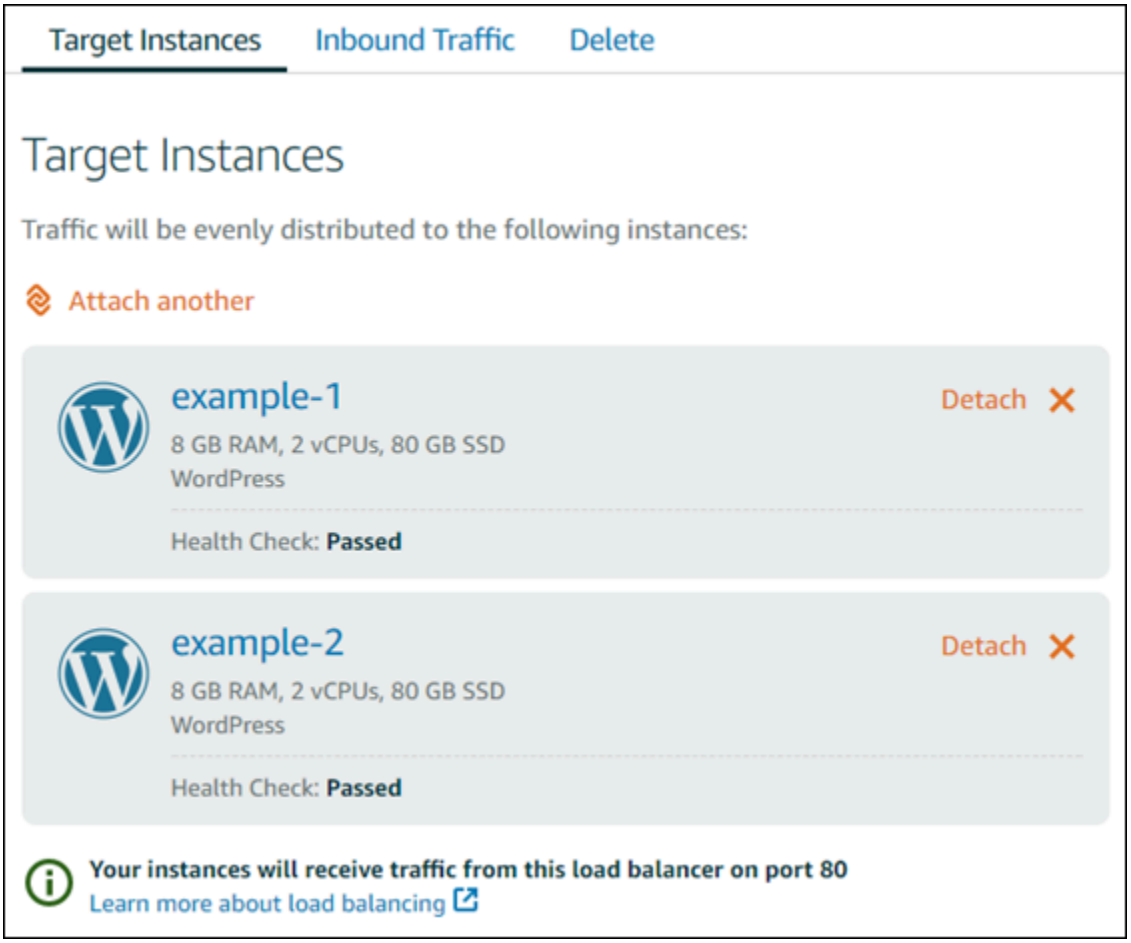
Lightsail ロードバランサーのヘルスチェックとHTTPS設定をカスタマイズする

Lightsail ロードバランサーを作成するときは、 AWS リージョン と名前を選択します。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。例えば、ホームページのロードが遅い場合や、イメージが多数ある場合は、ロードが速い別のページをチェックするように Lightsail を設定できます。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)

ウェブサイトユーザー向けにより安全なエクスペリエンスを作成するHTTPSように 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。

の詳細 HTTPS

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に Lightsail インスタンスを設定する

Amazon Lightsail ロードバランサーにインスタンスをアタッチする前に、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 (または水平スケーリング) を行うかどうか、およびアプリケーションに最適な設定方法についてレコメンデーションを提供します。

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をアプリケーションの残りの部分から分離して、データベースインスタンスを 1 つだけに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のいずれかです。

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すべての WordPress インスタンスが同じ場所から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。これを行うには、WordPress 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: インスタンスを WordPress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に設定に関する推奨事項

- Lightsail では、Bitnami によってパッケージ化された Node.js スタックには、Node.js、Apache、Redis (インメモリデータベース) 、および 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 データベースなどの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。Terraform Magento Adobe Commerce on AWS。
- 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に設定に関する推奨事項

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)
- アプリケーションの一元化されたデータベース (MySQL または Postgre SQL) 。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**
>
> 上記の共有ネットワークストレージサーバー (NFS) は、現在 GitLab 設計図では使用できません。

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 (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) を共有する方法を検討します。

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に設定に関する推奨事項

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に設定に関する推奨事項

- 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。

1 つ以上の Lightsail インスタンスで Nginx を実行し、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、「 を使用したウェブアプリケーションのスケーリングNGINX」、パート 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に設定に関する推奨事項

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。

- セッション永続化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 を実行しているいくつかの Lightsail インスタンスは、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ています。これは、Rsync などのツール、すべての Lightsail インスタンス間でコンテンツの同期を担当する NFSサーバー、または を使用したファイルの共有を使用して実行できます AWS。
- 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。

- 各 Lightsail インスタンスで設定された同じキャッシュシステム。など、便利な拡張機能がいくつかありますJotCache。

Lightsail ロードバランサーの TLS セキュリティポリシーを設定する

Amazon Lightsail ロードバランサーで HTTPS を有効にした後、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 ロードバランシングは、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間の SSL 接続をネゴシエートします。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。

TLS-2016-08 セキュリティポリシーは、Lightsail ロードバランサーで HTTPS を有効にすると、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、次のセキュリティポリシーとプロトコルで設定できます。

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、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 [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する 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、 コマンドリファレンスupdate-load-balancer-attributeの「」を参照してください。AWS CLI

> **Note**
>
> この手順を続行する前に、 をインストール AWS CLI し、Lightsail 用に設定する必要があります。詳細については、「Lightsail で動作する AWS CLI 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するロードバランサーの名前。
- *AttributeValue* TLS-2016-08または TLS-FS-1-2-Res-2019-08 セキュリティポリシーを使用する。

> **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 ロードバランサーの HTTP を HTTPS にリダイレクトする

Amazon Lightsail ロードバランサーで HTTPS を設定した後、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参照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、HTTP から HTTPS へのリダイレクトを設定できます。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

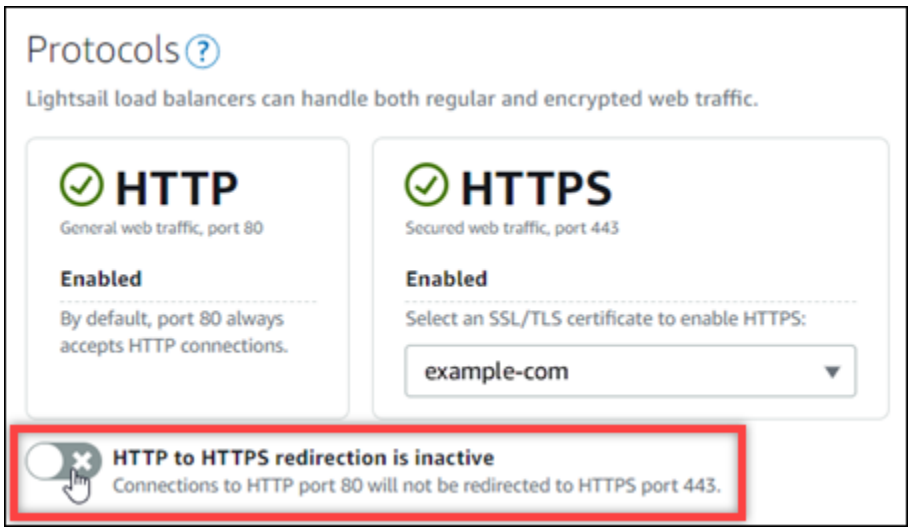
以下の前提条件を完了します (まだの場合)。

- 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、「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ロードバランサーで 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 [Networking] (ネットワーク) 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。

を使用してロードバランサーの HTTP から HTTPS へのリダイレクトを設定する AWS CLI

AWS Command Line Interface () 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AWS CLI。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、 コマンドリファレンスupdate-load-balancer-attributeの「」を参照してください。 AWS CLI

> **Note**
>
> この手順を続行する前に、 をインストール AWS CLI し、Lightsail 用に設定する必要があります。詳細については、「Lightsail で動作する AWS CLI 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* 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。
- *AttributeValue* を使用してリダイレクトtrueをアクティブ化するか、 false を使用してリダイレクトを非アクティブ化します。

> **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永続化を有効にする

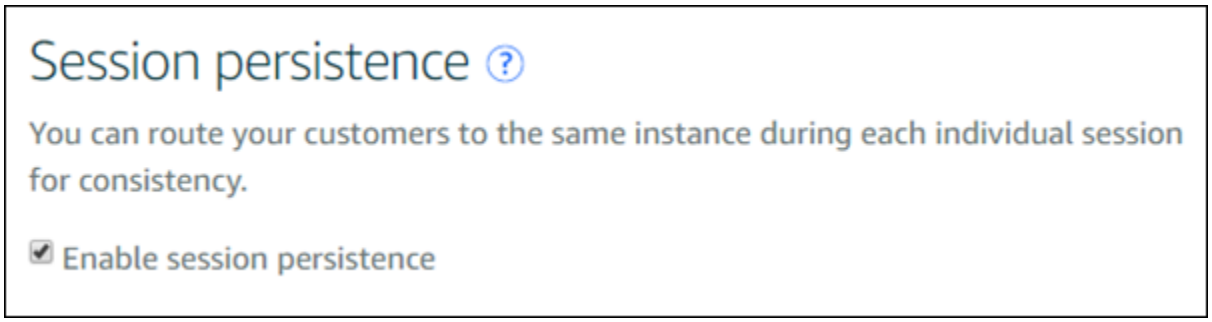
ユーザーのセッション永続性を有効にすることができます。これは、ユーザーのブラウザでセッション情報をローカルに保存する場合に役立ちます。例えば、Amazon Lightsail でショッピングカートを

使用して 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 を使用して Cookie の期間を調整することもできますAPI。

セッション永続性を有効にする

1. Lightsail ホームページで、ネットワーク 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。

まだ設定していない場合は、 をセットアップ AWS CLI して設定します。

Amazon Lightsail と連携 AWS Command Line Interface するように を設定する

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 コマンドを入力して、Cookie 期間を 3 日間 (259,200 秒) に増やします。

```
aws lightsail update-load-balancer-attribute --load-balancer-name LoadBalancerName
 --attribute-name SessionStickiness_LB_CookieDurationSeconds --attribute-value
 259200
```

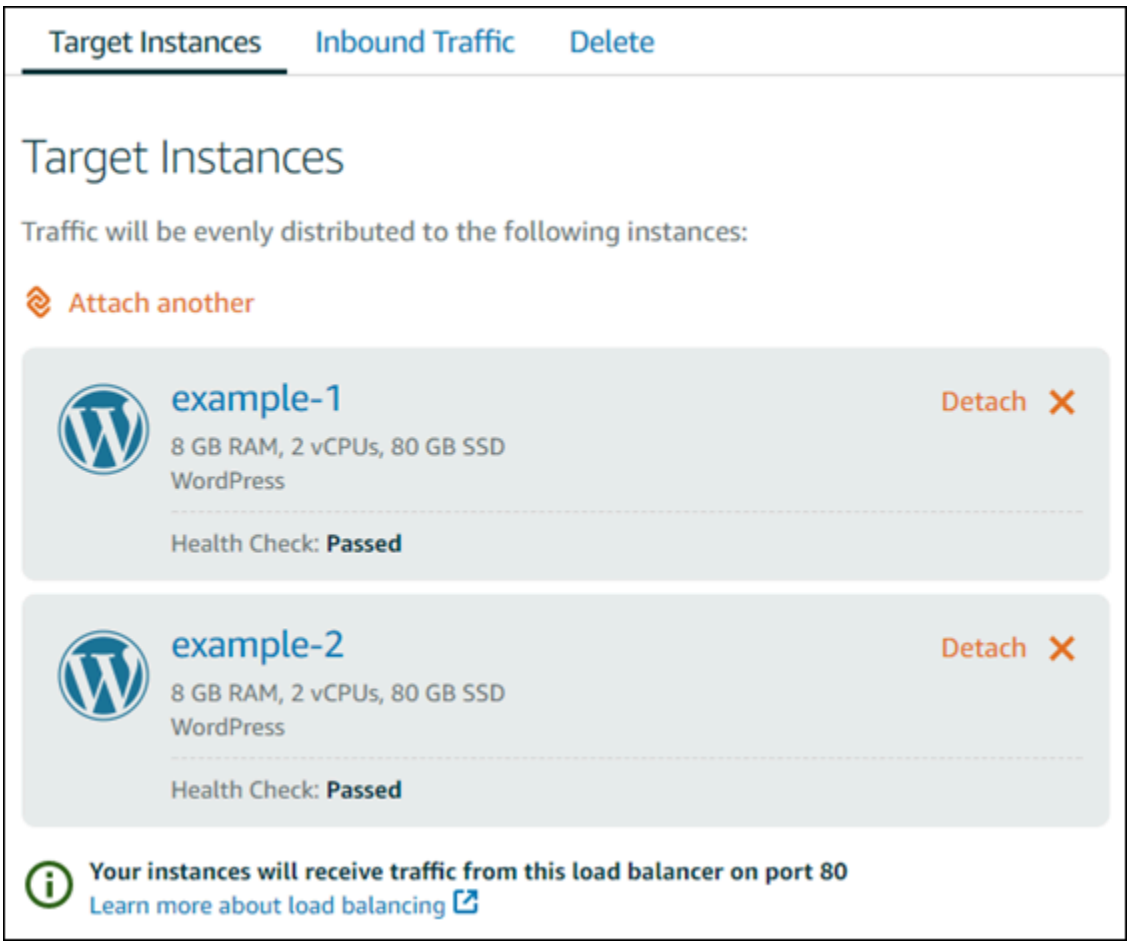
コマンドで、*LoadBalancerName* をロードバランサーの名前に置き換えます。

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
        }
    ]
}
```


Lightsail ロードバランサーのヘルスチェック設定を構成する

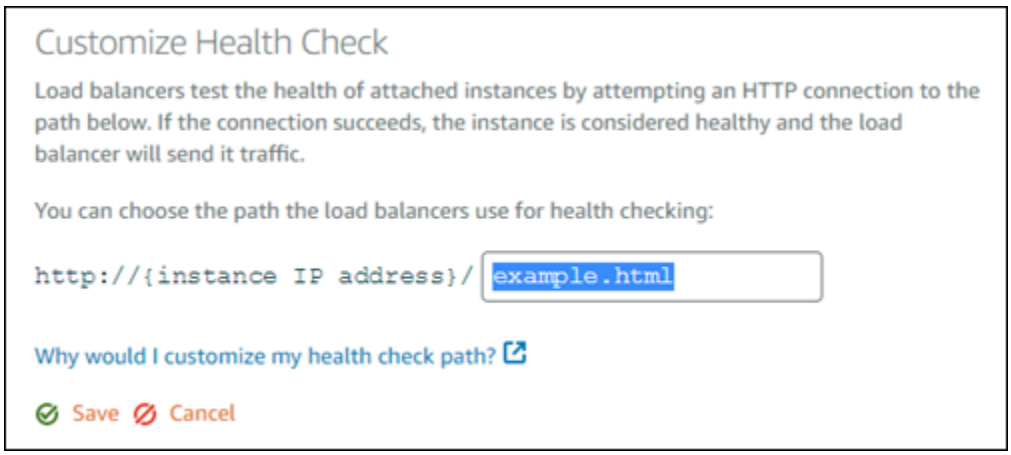
ヘルスチェックは、Lightsail インスタンスをロードバランサーにアタッチするとすぐに開始され、その後 30 秒ごとに行われます。ヘルスチェックのステータスを表示するには、ロードバランサーの管理ページを参照してください。



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。例えば、ホームページのロードが遅い場合や、イメージが多数ある場合は、ロードが速い別のページをチェックするように Lightsail を設定できます。

1. Lightsail ホームページで、ネットワーク 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 を使用してAPI、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** - クライアントによって開始され、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。

  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発信される 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発信される 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によって生成されたHTTPレスポンスコードの数。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によって生成されたHTTPレスポンスコードの数。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によって生成されたHTTPレスポンスコードの数。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によって生成されたHTTPレスポンスコードの数。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** - で処理されたリクエストの数IPv4。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 ロードバランサーのヘルスチェックを設定する

デフォルトでは、Lightsail はウェブアプリケーションのルート ("/") でインスタンスの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。

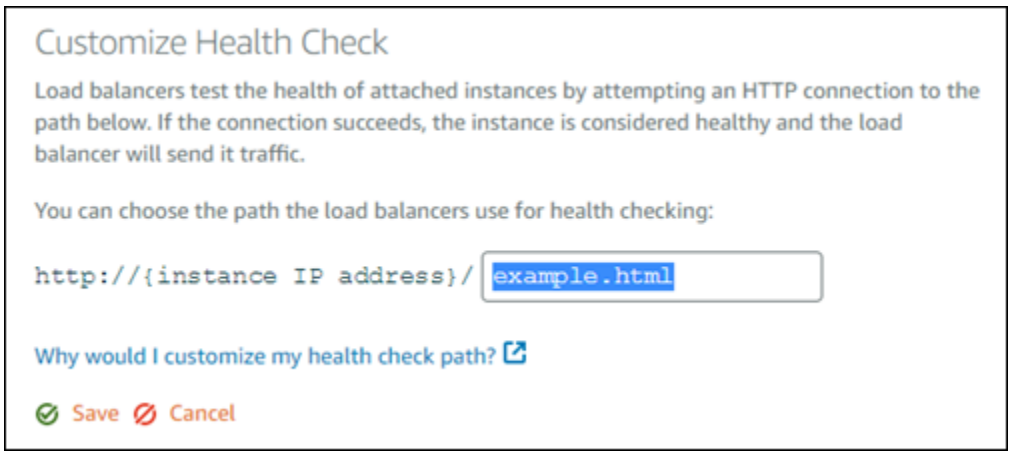
- 成功
- [失敗]

ヘルスチェックが失敗した場合は、 または Lightsail を使用して AWS Command Line Interface 問題を特定できますAPI。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。例えば、ホームページのロードが遅い場合や、イメージが多数ある場合は、ロードが速い別のページをチェックするように Lightsail を設定できます。

1. Lightsail ホームページで、ネットワーク 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

Amazon Lightsail ロードバランサーにインスタンスをアタッチする必要がなくなった場合は、デタッチできます。Lightsail インスタンスをロードバランサーからデタッチすると、指定されたインスタンスが不要になるまで待ってからデタッチします。

1. Lightsail ホームページで、ネットワーク 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 Lightsail ロードバランサーは削除できます。ロードバランサーを削除すると、ロードバランサーにアタッチされた Lightsail インスタンスも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ている 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の削除は最終であり、元に戻すことはできません。

1. Lightsail ホームページで、ネットワーク 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. [削除] を選択します。
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。

Lightsail コンテンツ配信ディストリビューションを使用してウェブコンテンツをグローバルに配信する

Lightsail ディストリビューションは、エッジロケーション とも呼ばれるグローバルに分散されたサーバーのネットワークを使用して、ユーザーへのコンテンツ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には、まず Lightsail インスタンスまたはコンテナサービス、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でウェブサイトまたはウェブアプリケーションを作成してホストするか、静的コンテンツを Lightsail バケットに保存します。次に、Lightsail ディストリビューションを作成して設定し、インスタンス、コンテナサービス、ロードバランサー、またはバケットからコンテンツをプル、キャッシュ、配信します。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。

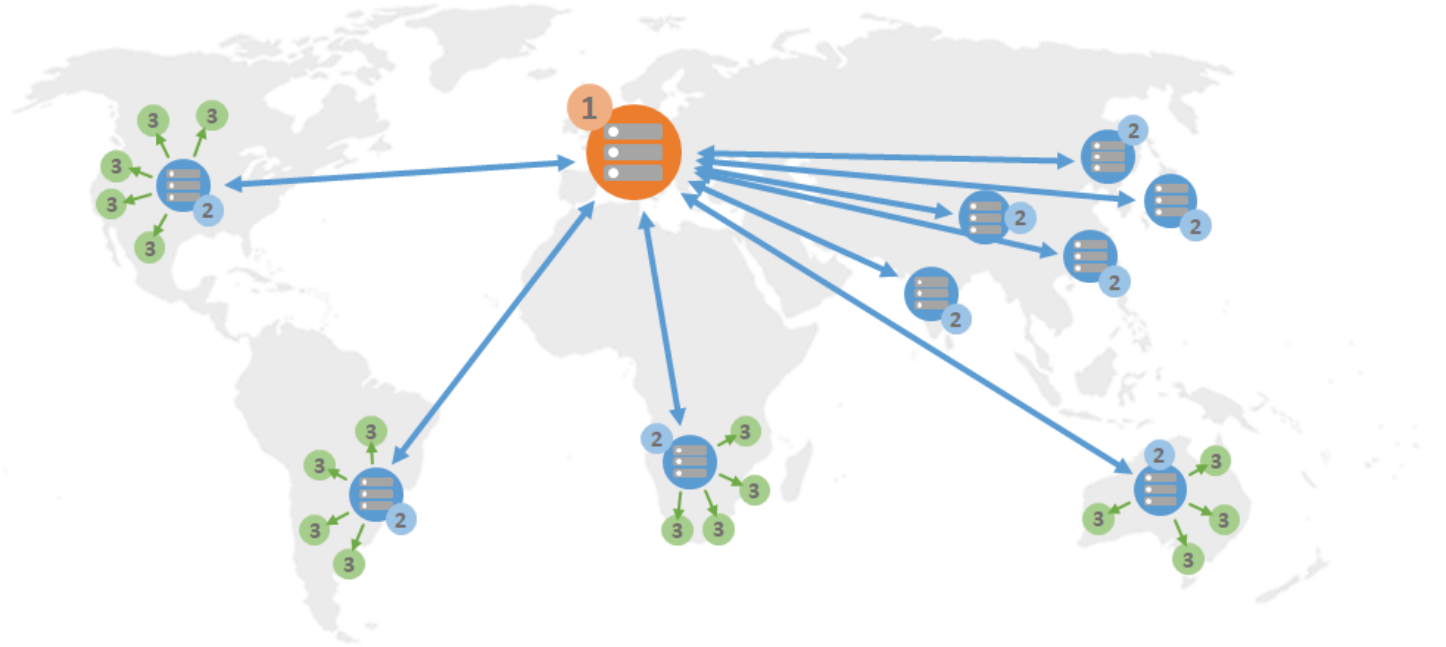
ユーザーがディストリビューションを通じて提供されているウェブサイトにアクセスしてコンテンツをリクエストすると、リクエストはレイテンシーの点から最も近い場所にルーティングされます。次に、ディストリビューションは以下のいずれかのアクションを実行します。

- 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表します。
- 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。

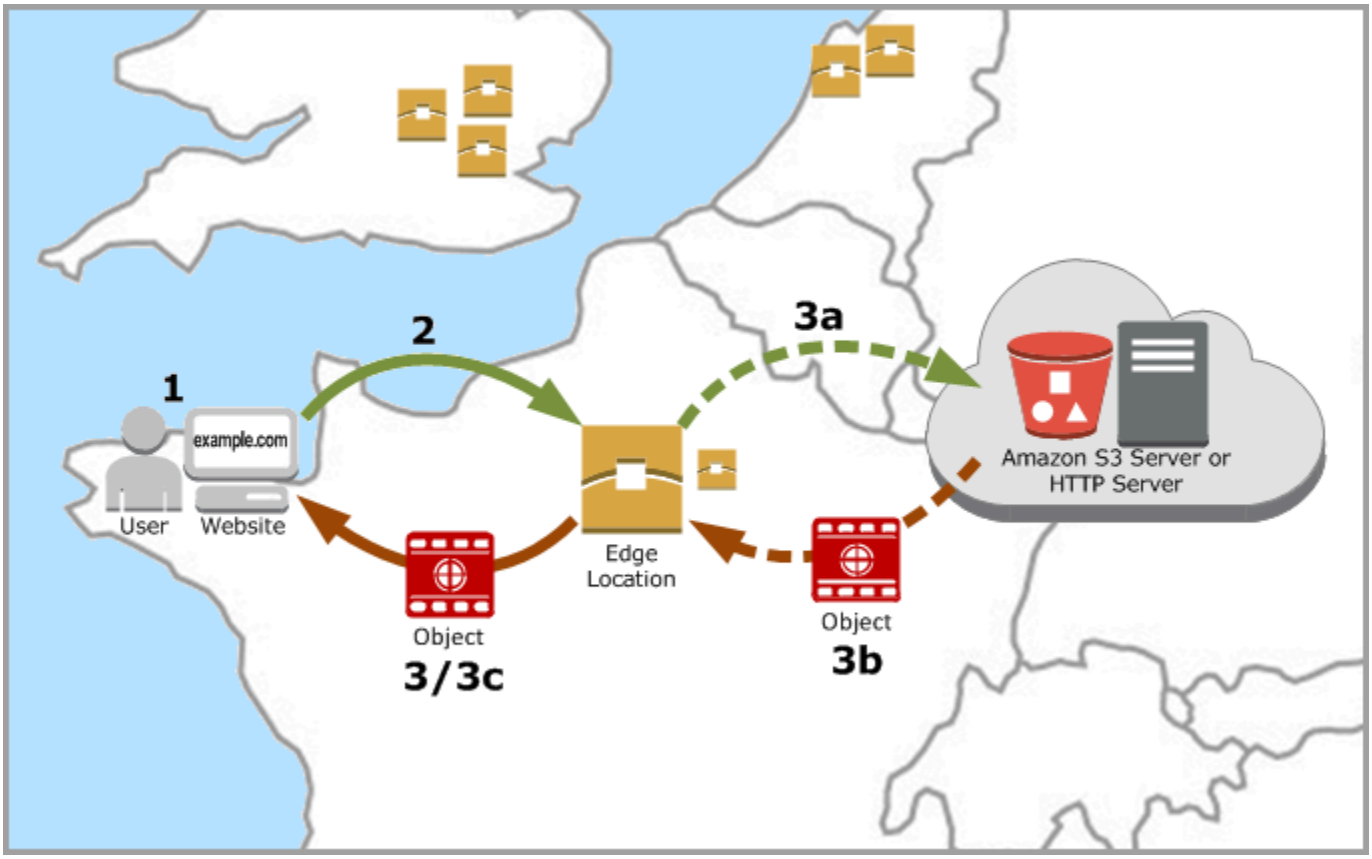


> **Note**
>
> この図表は図示のみを目的とし、実際のエッジロケーションは表示していません。エッジロケーションの詳細については、このガイドの後半のエッジロケーションと IP アドレスの範囲を参照してください。

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 ディストリビューションは、世界中の視聴者へのコンテンツ (ウェブサイトページ、イメージ JavaScript、スタイルシートなど) 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を Lightsail SSL/TLS 証明書と一緒に使用して、ディストリビューションの Hypertext Transfer Protocol Secure (HTTPS) を有効にできます。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

以下は、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またはウェブアプリケーションを提供するための一般的な手順です。

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。

   - 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。

- ウェブサイトまたは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) インスタンスでホストされているウェブサイトに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。
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 インスタンスの場合は、インスタンス WordPress の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 WordPressで動作するように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 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。
6. 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。
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。

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 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。 CloudFront エッジサーバーの場所のリストについては、「Amazon CloudFront 製品の詳細」ページを参照してください。 CloudFront IP 範囲のリストについては、 CloudFront 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。
   - ウェブサイトまたは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 ）

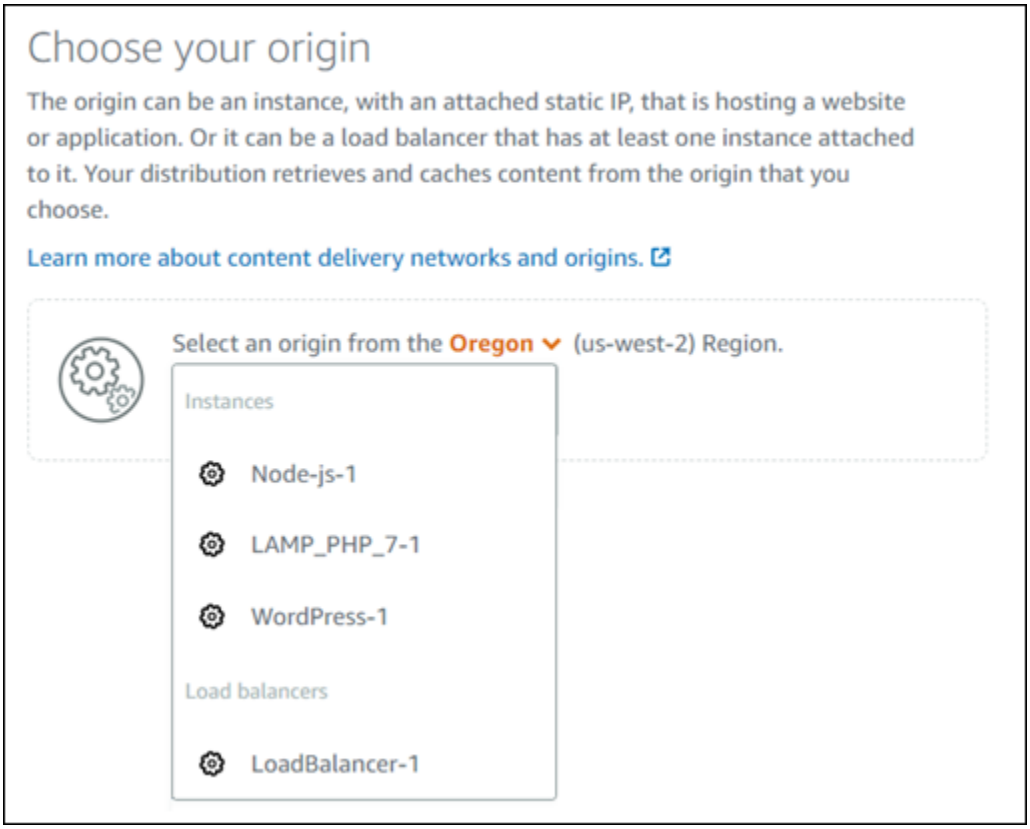
をディストリビューションのオリジンとして設定することができます。詳細については、「ロードバランサーを作成してインスタンスをアタッチする」を参照してください。

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 のみになります。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、Best for preset 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ます。

> **Note**
>
> キャッシュプリセットオプションは、ディストリビューションのオリジンとして Lightsail バケットを選択した場合は使用できません。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 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 Bitnami 認定ブループリントと 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、「プリセットの WordPress 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 および 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キャッシュに最適な WordPressプリセットを適用するかどうかを尋ねます。現在を適用した場合、ディストリビューションは WordPress ウェブサイトと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の何もキャッシュしないプ

リセットに最適な 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 WordPressで動作するように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すると、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**
>
> ディレクトリとファイルのオーバーライドオプションは、ディストリビューションのオリジンとして Lightsail バケットを選択した場合は使用できません。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**
>
> ディストリビューションのオリジンとして Lightsail バケットを選択した場合、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 AWS 、請求サイクルごとに 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。

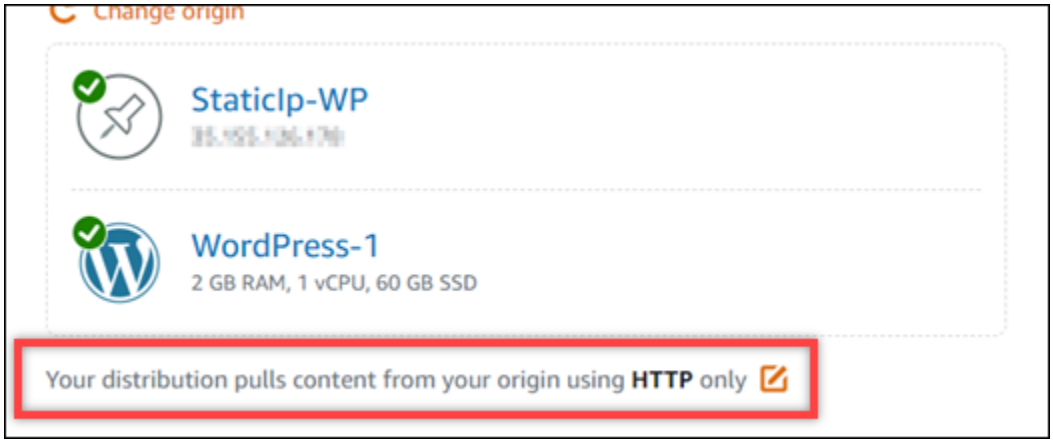
   ディストリビューションはグローバルリソースです。任意の でオリジンを参照し AWS リージョン、そのコンテンツをグローバルに配信できます。
5. オリジンの選択。オリジンは、Lightsail インスタンス、コンテナサービス、バケット、またはロードバランサー (1 つ以上のインスタンスがアタッチされている) です。詳細については、オリジンリソースを参照してください。

   > **Important**
   >
   > Lightsail コンテナサービス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、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**
>
> キャッシュプリセットオプションは、ディストリビューションのオリジンとして Lightsail バケットを選択した場合は使用できません。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**
>
> 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、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すると、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**
>
> ディレクトリとファイルのオーバーライドオプションは、ディストリビューションのオリジンとして Lightsail バケットを選択した場合は使用できません。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**
>
> ディストリビューションのオリジンとして Lightsail バケットを選択した場合、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:

    - 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。

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 WordPressで動作するように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

Amazon Lightsail ディストリビューションは、今後使用していない場合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 [Networking] (ネットワーク) 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を設定する

キャッシュ動作を使用すると、Amazon Lightsail ディストリビューションによってオリジンからキャッシュされるものとキャッシュされないものを設定できます。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、Best for preset 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。
- 最適 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の何も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 Bitnami 認定および Automattic 設計図 (マルチサイト設計図を除く) 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、「プリセットのキャッシュに最適 WordPress」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。

キャッシュプリセットに最適 WordPress

ディストリビューションのオリジンとして、WordPress Certified by Bitnami および 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キャッシュ用ベスト WordPressプリセットを適用するかどうかを尋ねます。現在を適用した場合、ディストリビューションは WordPress ウェブサイトと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の何もキャッシュしないプリセットに最適な 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 WordPressで動作するように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。

- 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/
  ```

- 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 [Networking] (ネットワーク) 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 から有効 InProgress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する

キャッシュの有効期間 (有効期限) 設定は、コンテンツが Amazon Lightsail ディストリビューションのキャッシュに保持される時間を制御します。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 [Networking] (ネットワーク) 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コンテンツ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の作成後にそのオリジンを変更する方法について説明します。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、Lightsail バケット、または Lightsail 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。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 [Networking] (ネットワーク) 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。

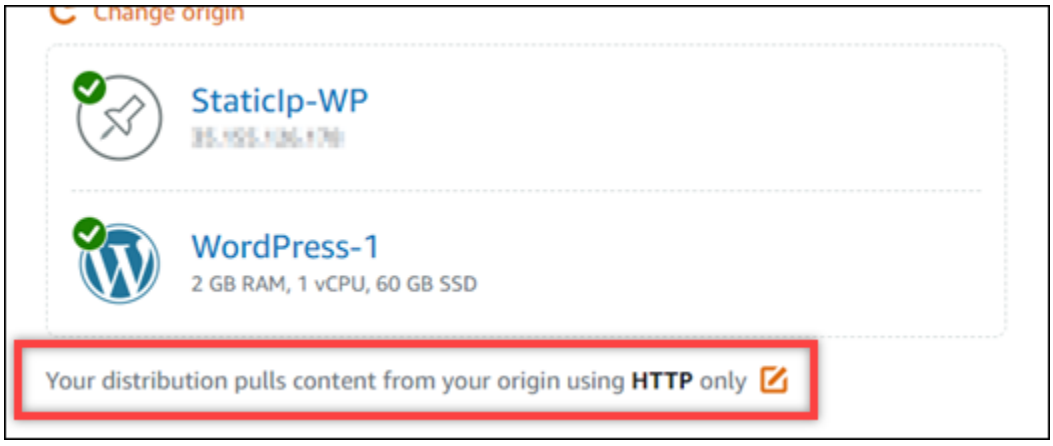
5. オリジンを変更を選択します。
6. オリジンのリソースが作成された AWS リージョンを選択します。

   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、オリジンプロトコルポリシーはデフォルトで HTTPS のみになります。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。



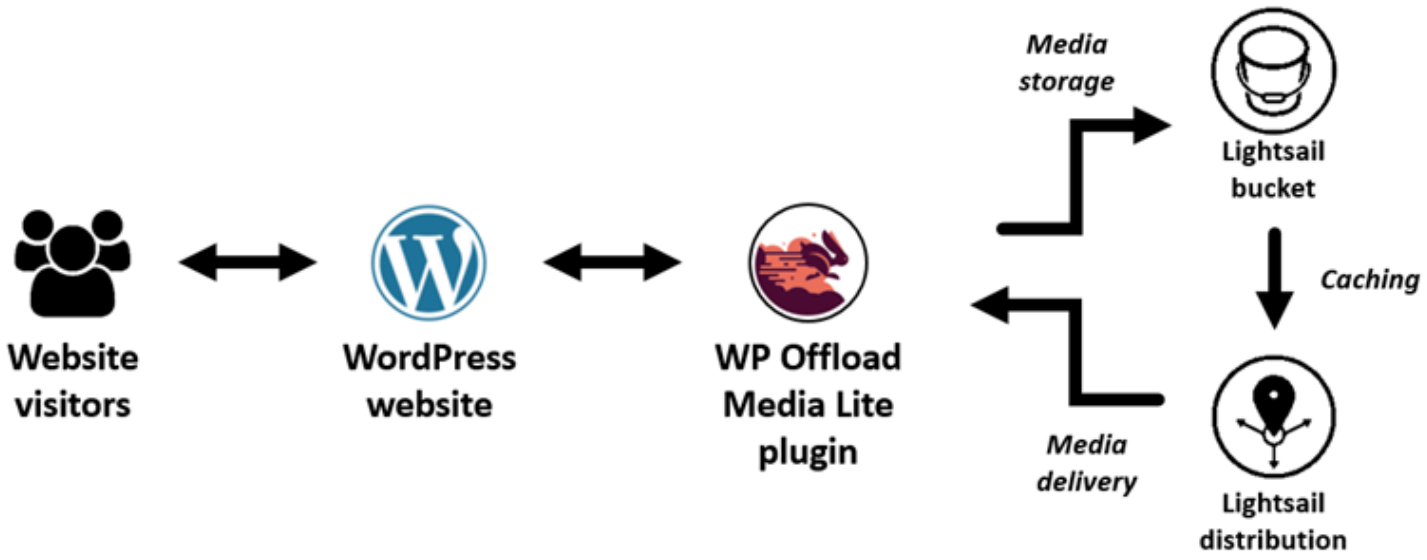
10. HTTP Only または HTTPS Only を選択して、保存を選択してオリジンプロトコルポリシーを保存します。

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 から有効 InProgress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 バケットと CDN ディストリビューションを使用してメディアファイルを効率的に提供する

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。



Lightsail バケットにウェブサイトメディアを保存すると、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配信すると、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)。

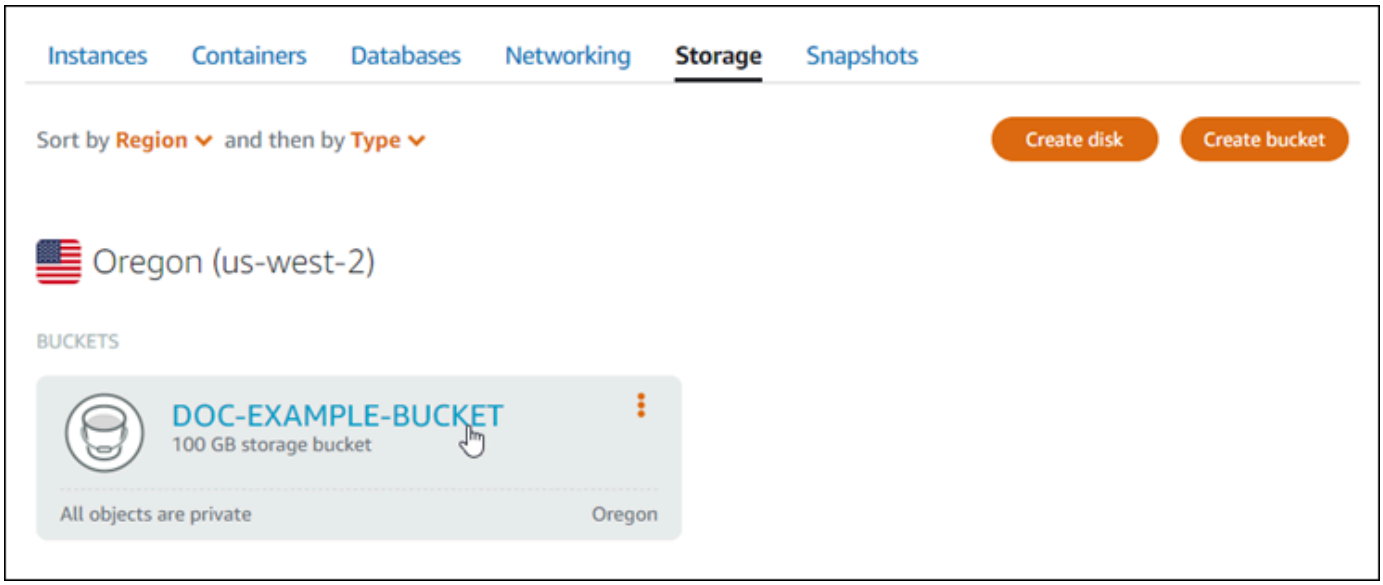
- Lightsail で WordPress インスタンスを作成して設定し、管理ダッシュボードにサインインするためのパスワードを取得します。詳細については、「チュートリアル: Amazon Lightsail で WordPress インスタンスを起動して設定する Amazon Lightsail」を参照してください。
- Lightsail オブジェクトストレージサービスでバケットを作成します。詳細については、「Lightsail でのバケットの作成」を参照してください。

ステップ 2: バケットのアクセス許可を変更する

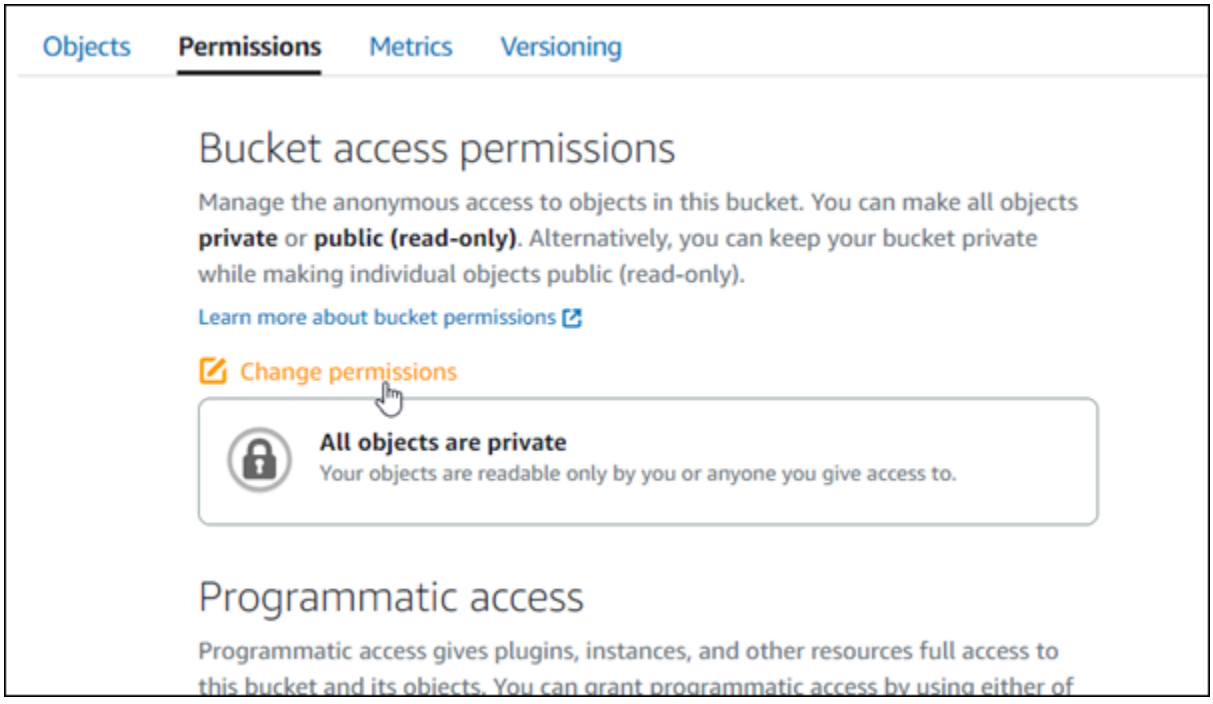
次の手順を実行して、 WordPress インスタンスと WP Offload Media Lite プラグインにバケットへのアクセスを許可します。バケットのアクセス許可は個々のオブジェクトを公開 (読み取り専用) に設定する必要があります。また、 WordPress インスタンスをバケットにアタッチする必要があります。バケット許可の詳細については、「バケットのアクセス許可」を参照してください。

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。

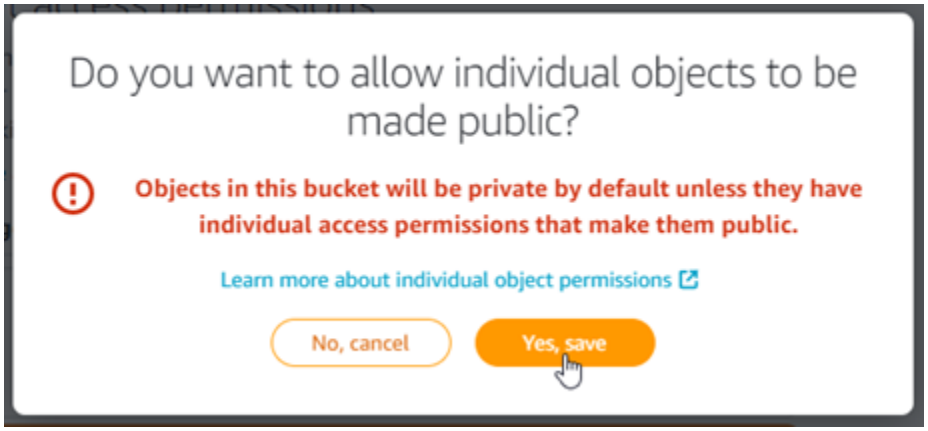


4. バケット管理ページで [Permissions] (許可) タブを選択します。
5. ページの「バケットのアクセス許可」セクションで [Change permissions](許可の変更) を選択します。



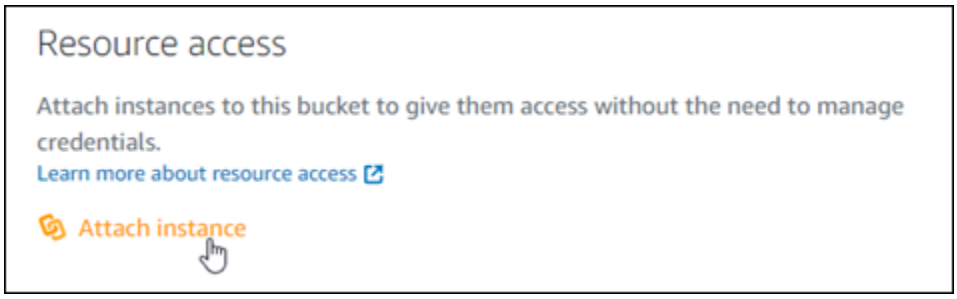
6. 個々のオブジェクトを選択して公開し、読み取り専用にすることができます。

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。

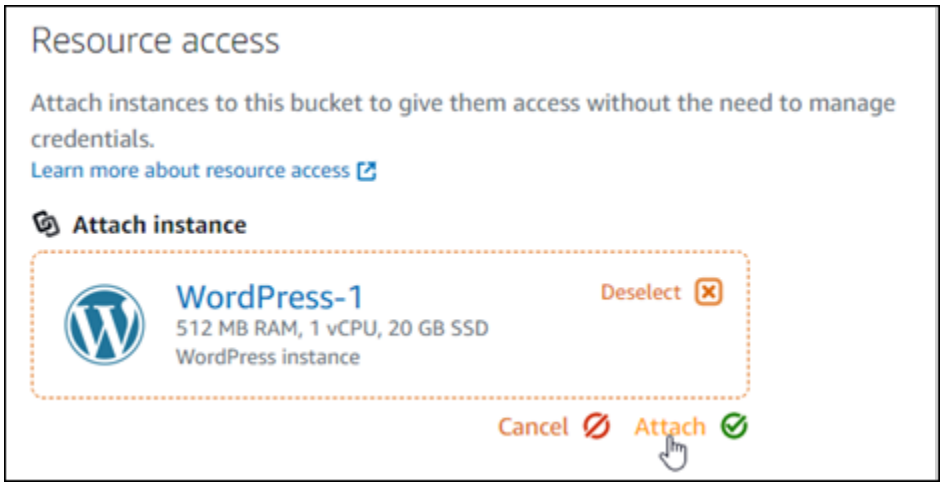


しばらくすると、バケットは個々のオブジェクトにアクセスを許可するように設定されます。これにより、 Offload Media Lite プラグインを使用して WordPress ウェブサイトからバケットにアップロードされたオブジェクトを顧客が読み取ることができます。

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。

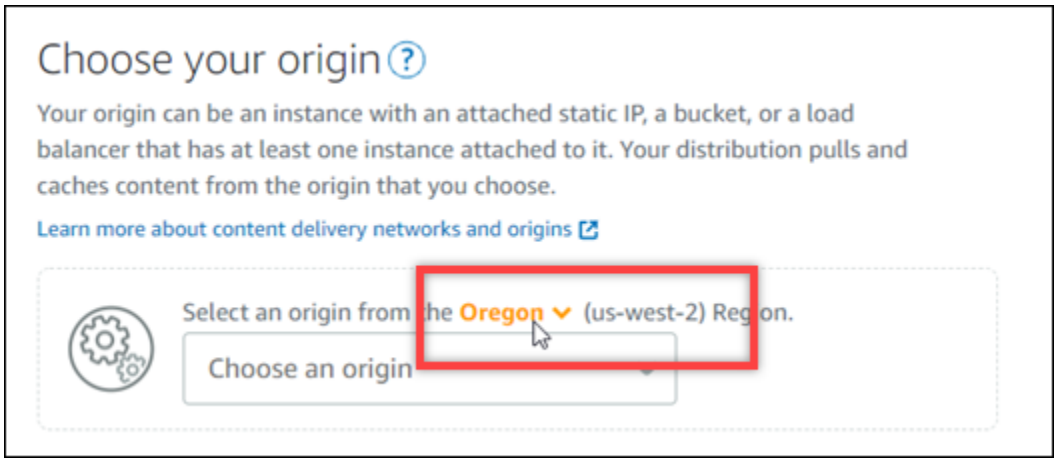
ステップ 3: オリジンとしてのバケットを持つディストリビューションを作成する

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、次の手順を実行します。

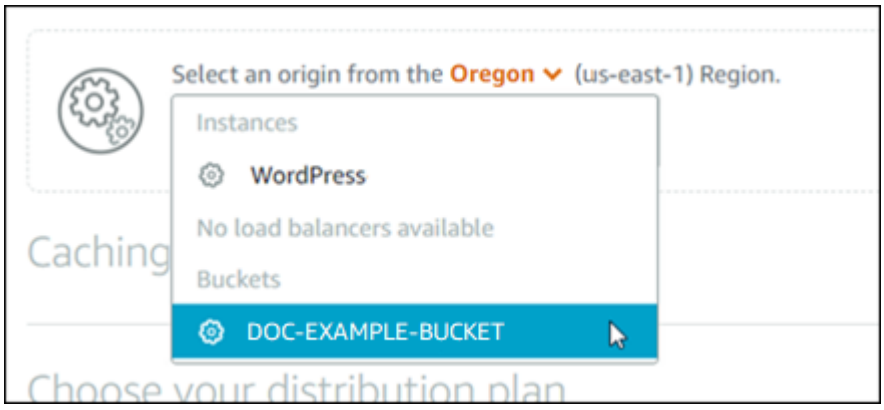
1. Lightsail コンソールの上部のナビゲーションメニューでホームを選択します。
2. lightsail のホームページで [Networking] (ネットワーク) タブを選択します。
3. [ディストリビューションの作成] を選択します。

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。任意の でバケットを参照し 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。

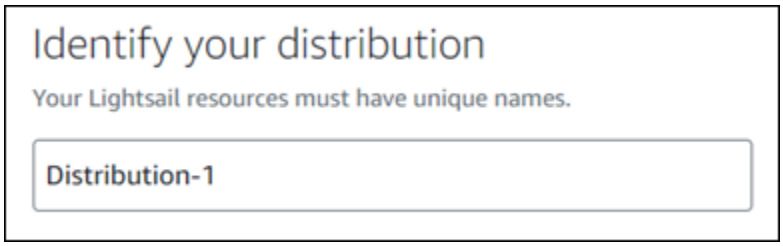


5. バケットをオリジンとして選択します。



> **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：

- 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. [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。

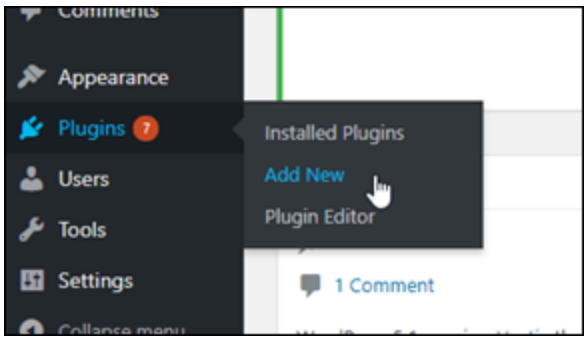
ステップ 5: WordPress ウェブサイトに WP Offload Media Lite プラグインをインストールする

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、次の手順を実行します。このプラグインは、 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 WordPress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。

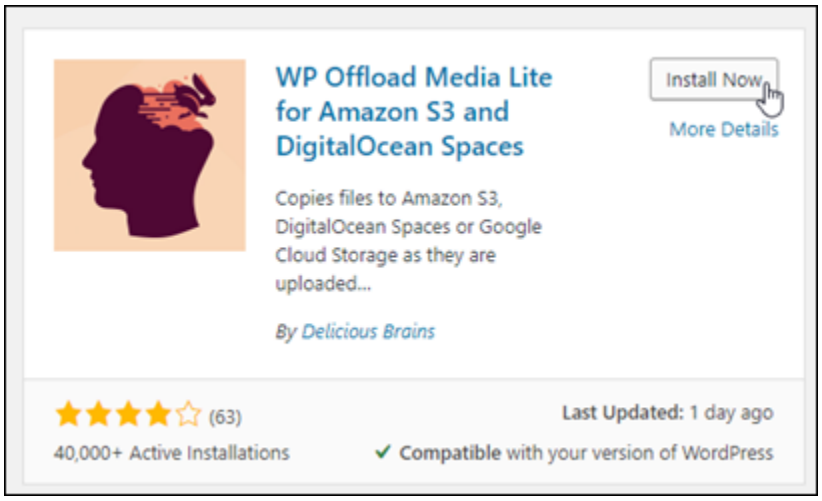
   詳細については、Amazon Lightsail」を参照してください。

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。

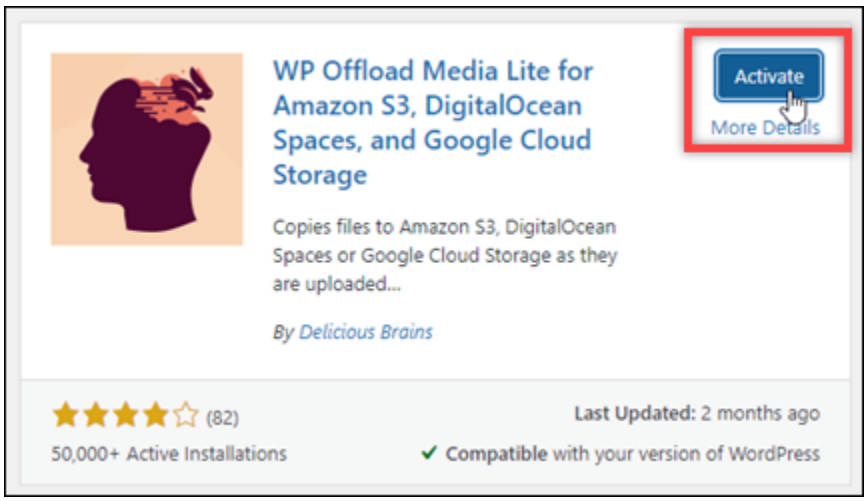


3. [WP Offload Media Lite] を検索します。

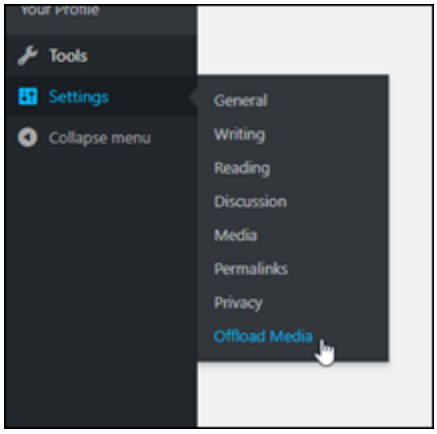
4. 検索結果で、[WP Offload MediaLite] プラグインの横にある [Install Now] (今すぐインストール) を選択します。



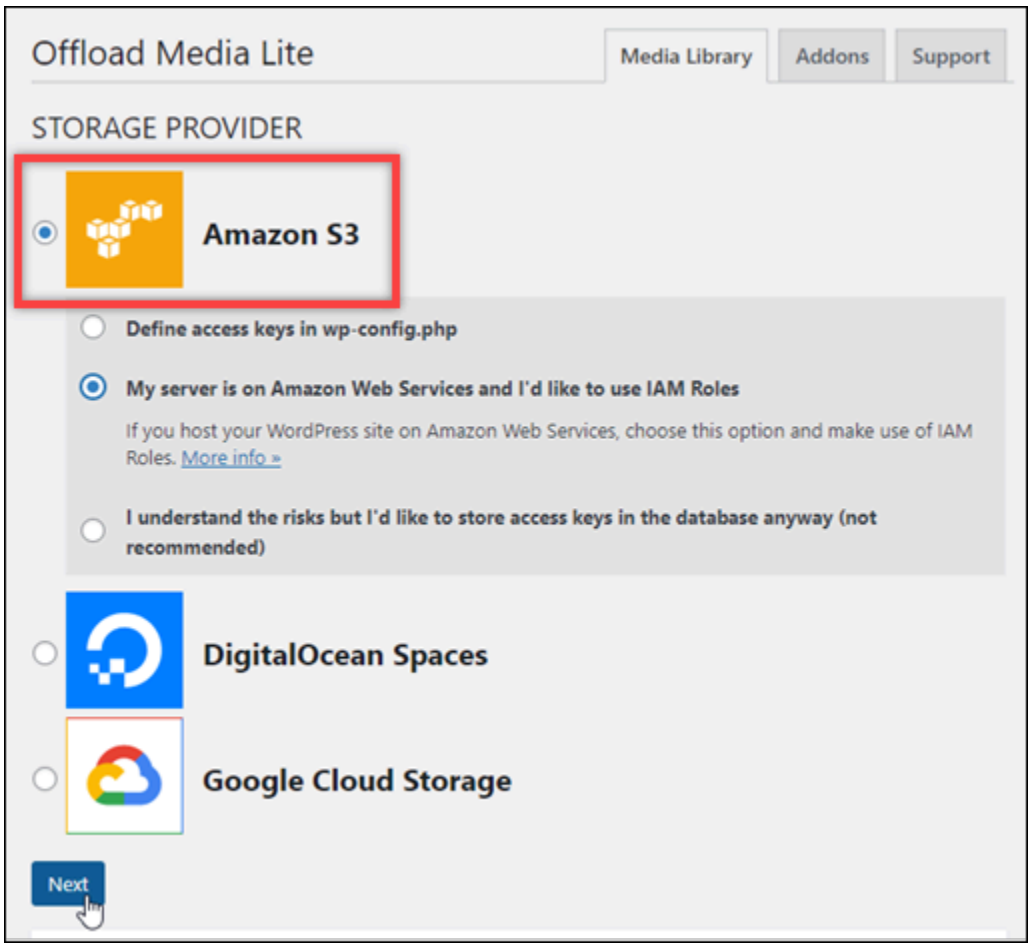
5. プラグインのインストールが完了したら、[アクティベート] を選択します。



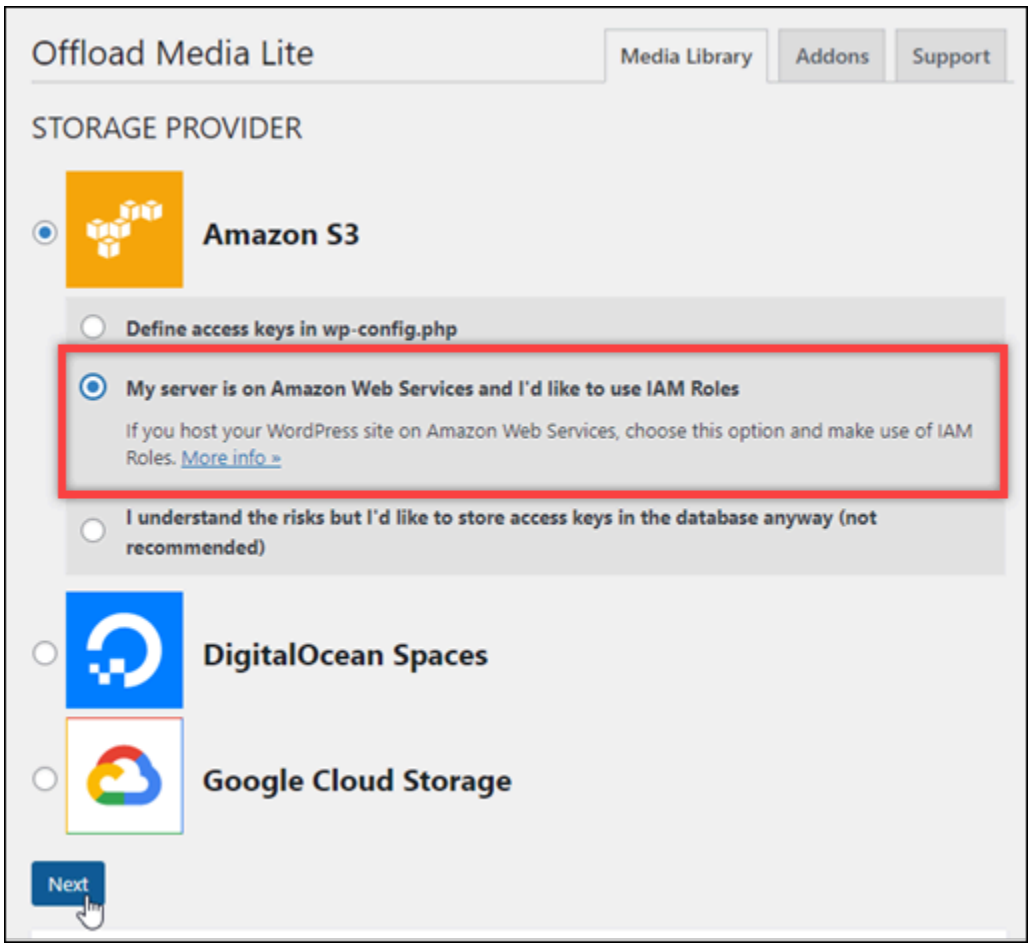
6. 左ナビゲーションメニューで、[設定]、[Offload Media] の順に選択します。



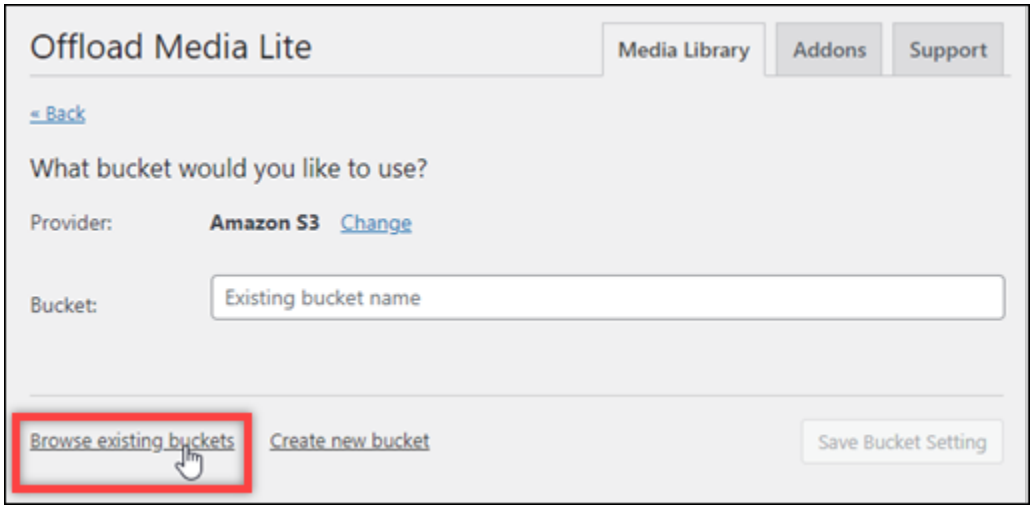
7. [Offload Media Lite] ページで、ストレージプロバイダーとして [Amazon S3] を選択します。



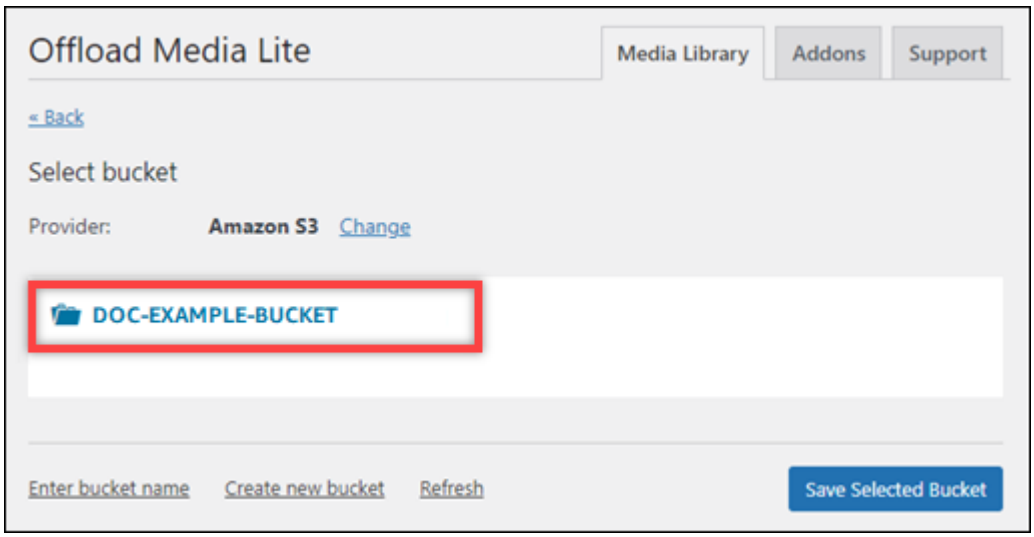
8. [私のサーバーは Amazon Web Services 上にあり、IAM ロールを使いたい] を選択します。



9. [次へ] をクリックします。
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。



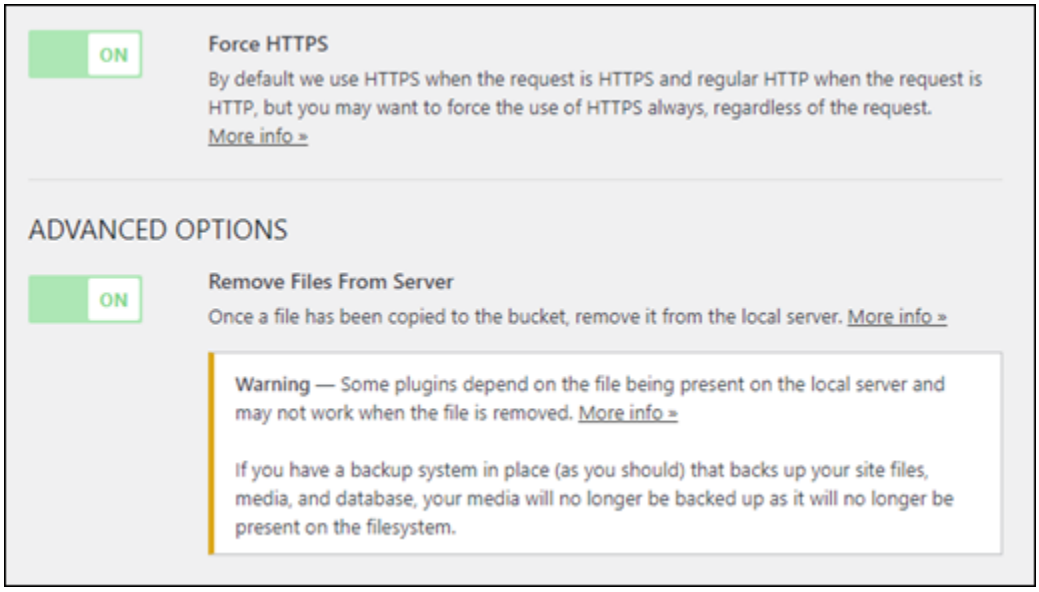
11. インスタンスで使用する WordPressように作成したバケットの名前を選択します。



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。

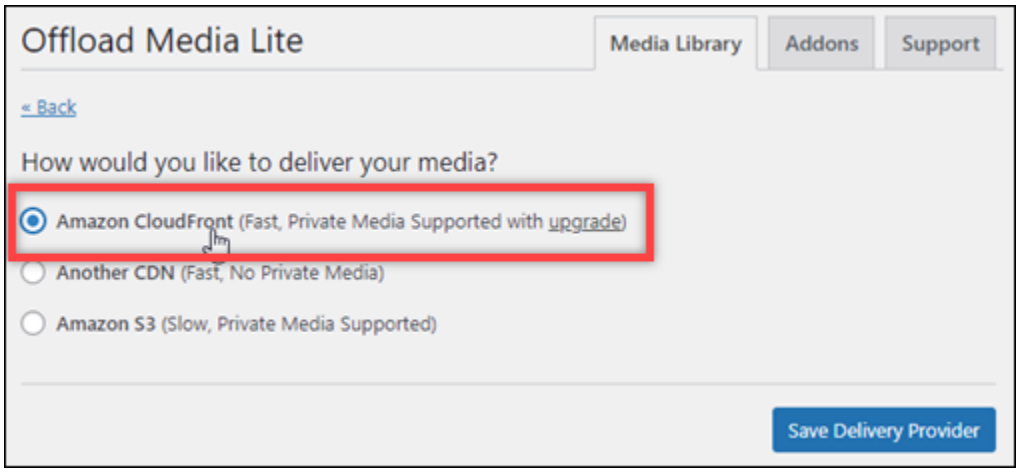
    - Lightsail バケットはデフォルトで HTTPS を使用してメディアファイルを配信するため、強制 HTTPS 設定を有効にする必要があります。この機能をオンにしないと、 WordPress ウェブサイトから Lightsail バケットにアップロードされたメディアファイルは、ウェブサイトの訪問者に正しく提供されません。

      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もインスタンスのディスクに保存されません。この機能を有効にしない場合、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。

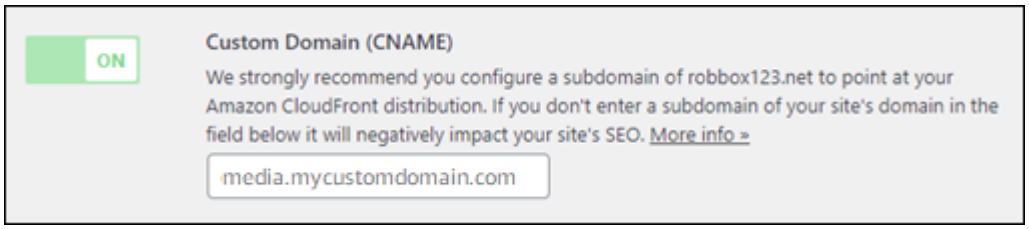
14. 表示されるメディアの配信方法ページで、Amazon CloudFrontを選択します。



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。

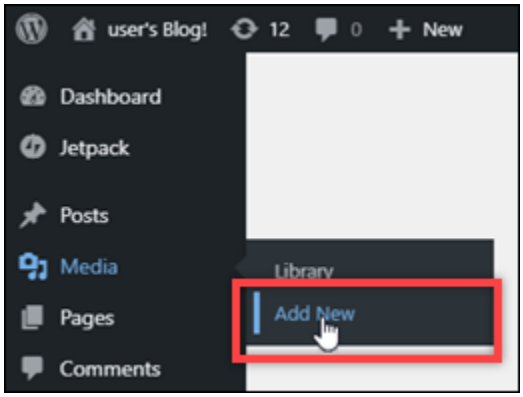
> **Note**
>
> 後で [Offload Media Lite 設定] ページに戻るには、左のナビゲーション メニューで [設定] を一時停止し、[Offload Media] を選択します。

これで WordPress 、ウェブサイトが Media Lite プラグインを使用するように設定されました。次に 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 Lightsail バケットに自動的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。

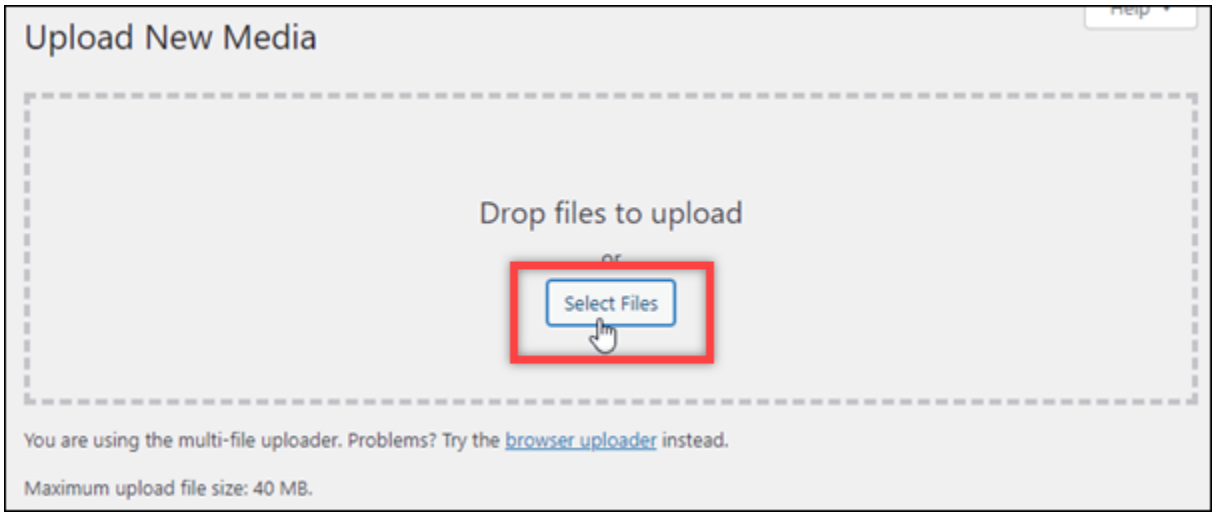
ステップ 6: WordPress ウェブサイトと Lightsail バケットおよびディストリビューション間の接続をテストする

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。

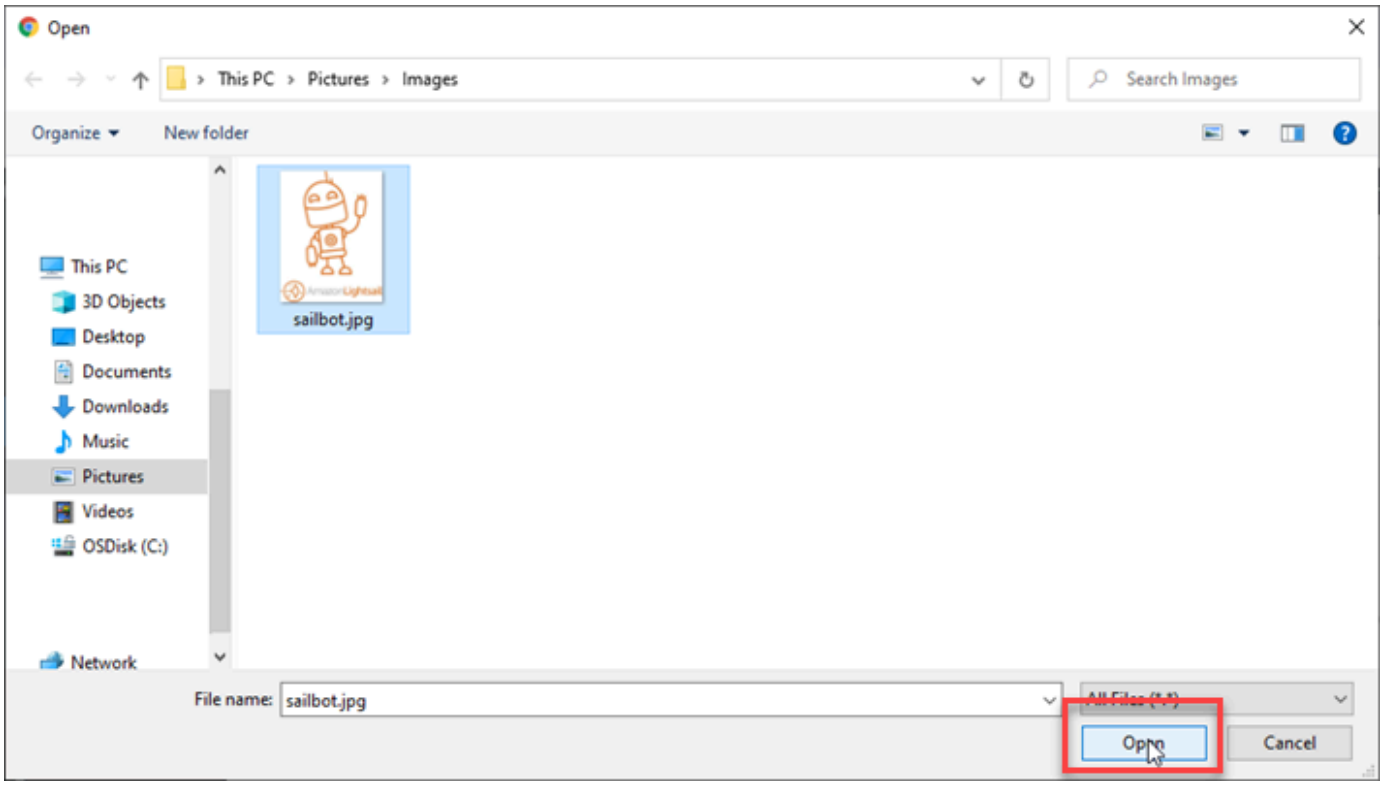
1. ダッシュボードの WordPress左側のナビゲーションメニューでメディアで一時停止し、新しいを追加 を選択します。



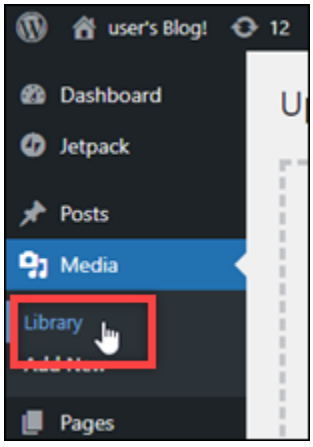
2. 表示される [新しいメディアのアップロード] ページで [ファイルを選択] を選択します。



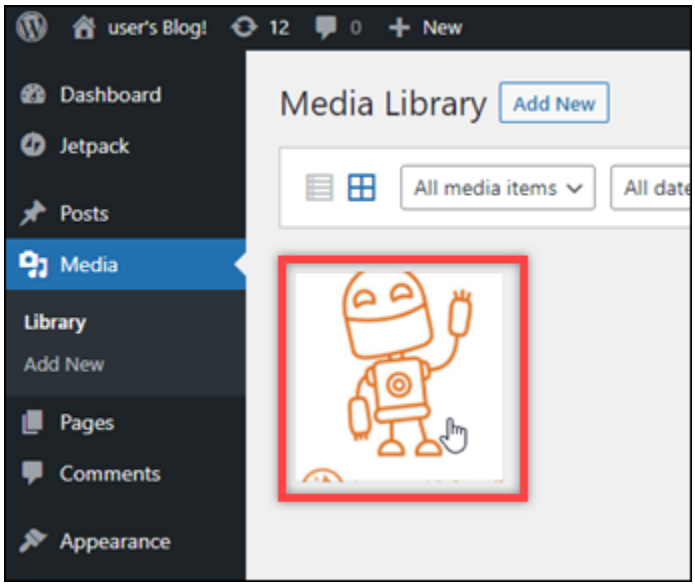
3. ローカルコンピュータからアップロードするメディアファイルを選択し、[開く] を選択します。



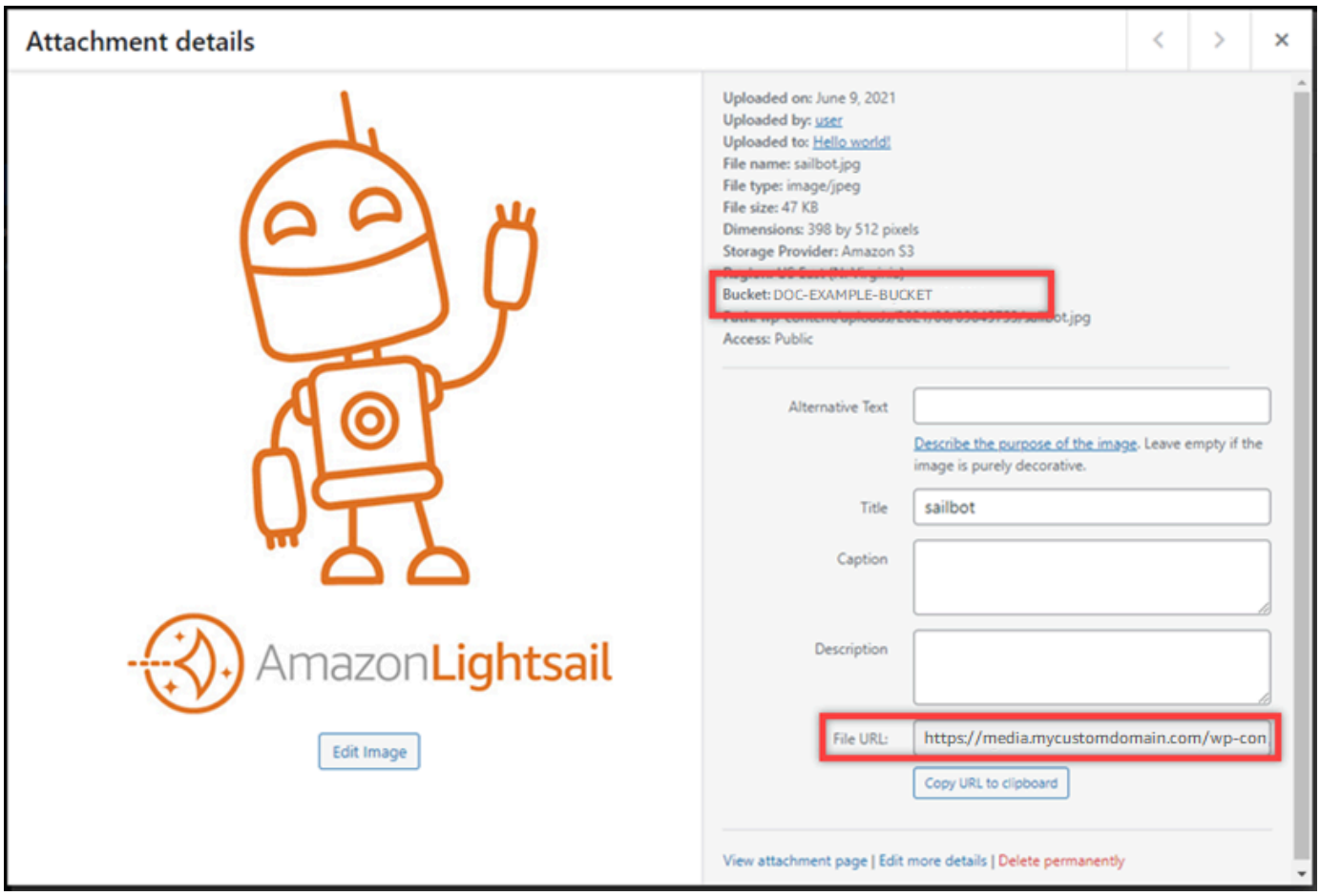
4. ファイルのアップロードが完了したら、左のナビゲーションメニューにある [メディア] の [ライブラリ] を選択します。



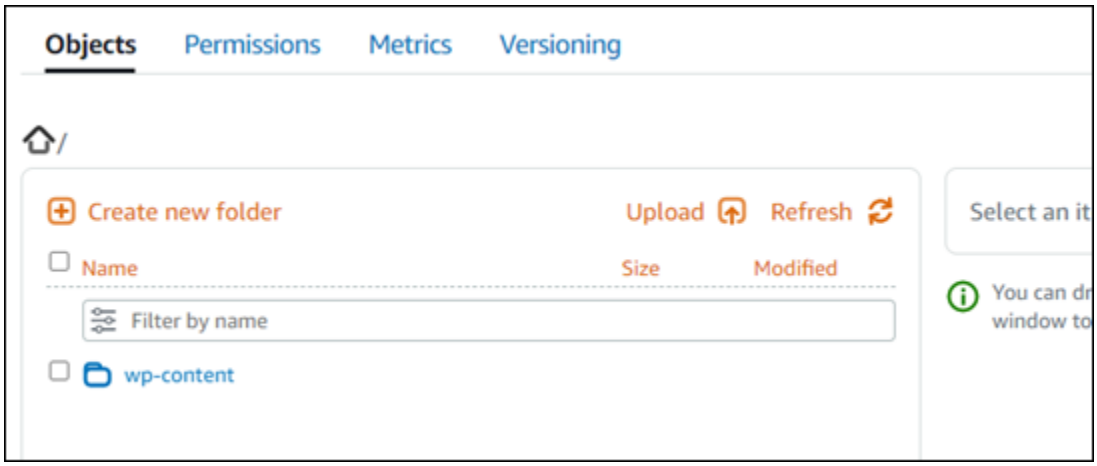
5. 最近アップロードしたファイルを選択します。



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」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」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 でバケット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の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バケットのアクセスログ記録を有効にする
   - Amazon Lightsail の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」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について」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 で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」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」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」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」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: Amazon Lightsail バケットへの WordPress インスタンスの接続
   - 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 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」を参照してください。

Lightsail ディストリビューションのデータ転送クォータを調整する

Amazon Lightsail ディストリビューションを作成するときは、ディストリビューションの毎月のデータ転送クォータとコストを指定する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 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 AWS 、請求サイクルごとに 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 [Networking] (ネットワーク) 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でコンテンツを提供する

Amazon Lightsail ディストリビューションのカスタムドメインを有効にして、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ときは、ディストリビューションで使用するドメイン用に作成した 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) アカウントで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。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 [Networking] (ネットワーク) 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

- カスタムドメインを 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 SSL/TLS 証明書ドメインを更新する
- 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

カスタムドメインを 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後、登録されたドメイン名を Amazon Lightsail ディストリビューションにポイントする必要があります。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 ゾーンを使用してドメインをディストリビューションにポイントする手順について説明します。Domain.com や など、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ポイントする手順は似ている GoDaddy場合があります。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

開始する前に、Lightsail 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。

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 [Networking] (ネットワーク) 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 & 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 を使用すると、他の DNS ホスティングプロバイダーで通常必要となる追加の手順なしで、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 SSL/TLS 証明書ドメインを更新する

Amazon Lightsail ディストリビューションで使用されるカスタムドメインを別のドメインまたはドメインのセットに変更できます。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 [Networking] (ネットワーク) 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、[Attach] (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て、ディストリビューションで登録済みドメイン名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 [Networking] (ネットワーク) 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の管理ページの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

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 Amazon Lightsail コンテナサービスを選択できます。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 コンテナサービスで Lightsail ディストリビューションを使用している場合、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。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。
- コンテナサービスで 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使用している場合。

AWS Command Line Interface () を使用してのみ、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AWS CLI。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 () を使用して Lightsail のコンテナサービスにディストリビューションのデフォルトドメインを追加するには、以下の手順を実行しますAWS CLI。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、 コマンドリファレンスupdate-container-serviceの「」を参照してください。 AWS CLI

> **Note**
>
> この手順を続行する前に、 をインストール AWS CLI し、Lightsail 用に設定する必要があります。詳細については、「Lightsail で動作する AWS CLI 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*「 - 現在コンテナサービスにアタッチされているカスタムドメインの 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を管理する

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、オリジンからのレスポンスを処理する際の 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
- キャッシュ期間
- クライアント IP アドレス
- クライアント側の SSL 認証
- 圧縮
- 条件付きリクエスト
- cookie
- クロスオリジンリソース共有 (CORS)
- 暗号化
- 本文を含む GET リクエスト
- HTTP メソッド
- HTTP リクエストヘッダーとディストリビューション動作
- HTTP バージョン
- リクエストの最大長と URL の最大長
- OCSP Stapling
- 持続的接続
- プロトコル
- クエリ文字列
- オリジン接続のタイムアウトと試行
- オリジン応答タイムアウト
- 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
- ユーザーエージェントヘッダー

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、クライアント側の 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フィールド値 "identity"および を持つリクエストを転送します"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 - なし (接続を除く WebSocket)

  設定されていない場合の動作 - WebSocket 接続を確立しない限り、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 - 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 のオリジン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 のみです。

[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。

オリジンがレスポンスVary:*で を返し、対応するキャッシュ動作の最小 TTL の値が他の値である場合、 はディストリビューションが を削除または置き換える HTTP レスポンスヘッダーで説明Varyされているように ヘッダー CloudFront を処理します。 ???

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。これにより、オブジェクトのキャッシュ性能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、ヘッダー名を ヘッダーVaryのディストリビューション (などVary:Accept-Charset,Accept-

Language) に返すようにオリジンを設定すると、ディストリビューションはそれらの値を持つ Varyヘッダーをビューワーに返します。

- 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 ディストリビューションは、 `Transfer-Encoding`ヘッダー`chunked`の値のみをサポートします。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のコンテンツキャッシュを検証する

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ことをテストする方法について説明します。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。

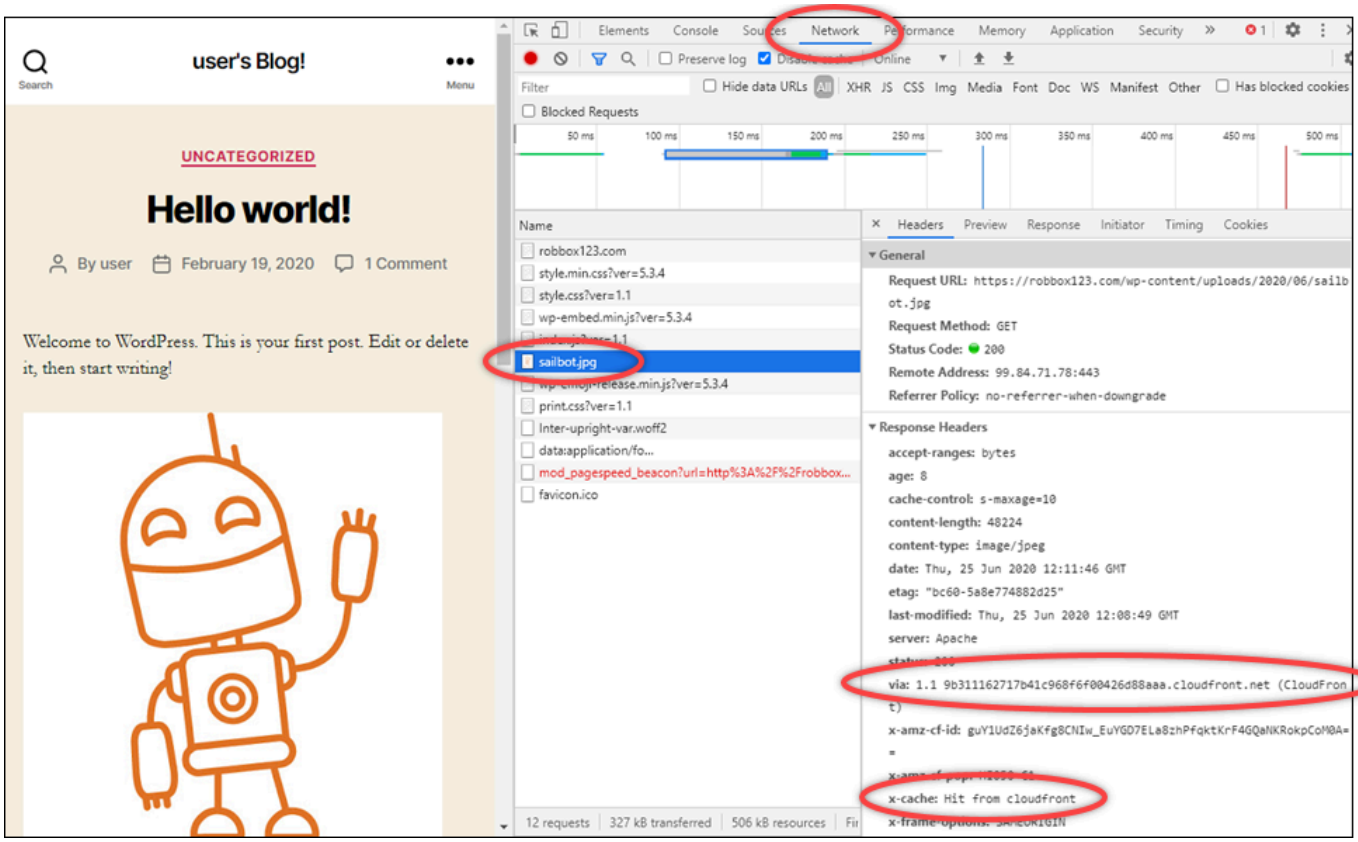
ディストリビューションをテスト

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 upper-right-hand 隅にある Chrome メニューを開き、その他のツール > 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、 ヘッダー`via`と `x-cache`ヘッダーの両方に が言及されていることがわかります CloudFront。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
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 のネットワークリソース

Lightsail ネットワークリソースは、ユーザーおよび外部サービスが 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 IPs

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。

Lightsail リソースの IP アドレスの表示と管理

IP アドレスを使用して、Lightsail インスタンスやその他の Lightsail リソースと通信できます。例えば、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 (を使用PING) を確認し、インスタンスSSHへの接続を確立し、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 アドレス指定プロトコルの両方をサポートします。これらのリソースはデフォルトでIPv4アドレス指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IPv6に対して 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を説明します。

目次

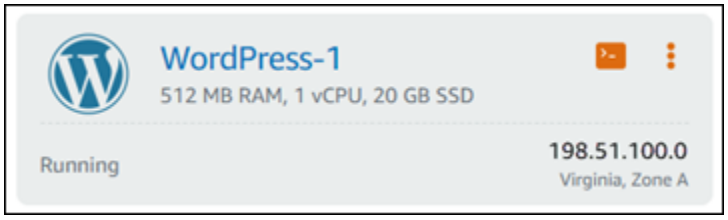
インスタンスのプライベートアドレスとパブリックIPv4アドレス

Lightsail インスタンスを作成すると、パブリック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、VPCピアリングを有効にすると、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、Lightsail アカウント外からアクセスできます。詳細については、「Amazon VPCピアリングをセットアップして Lightsail の外部のAWSリソースと連携させる」を参照してください。

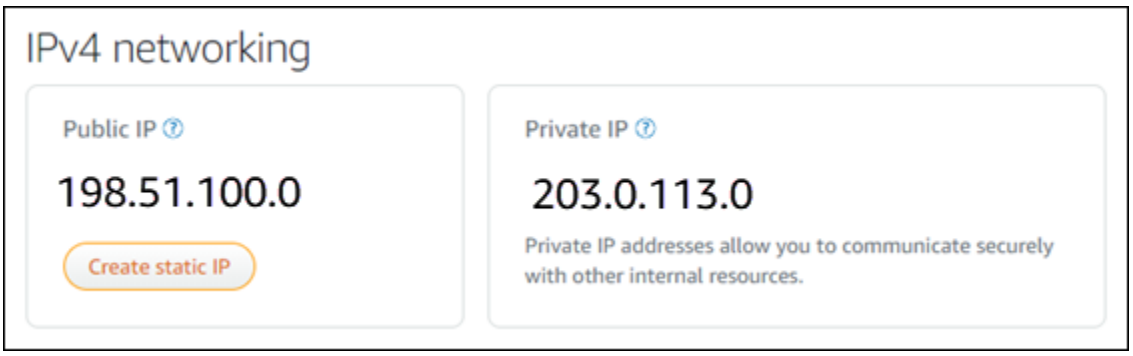
インスタンスの IP アドレスは、Lightsail コンソールの次の領域に表示されます。

- 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、次の点に注意してください。

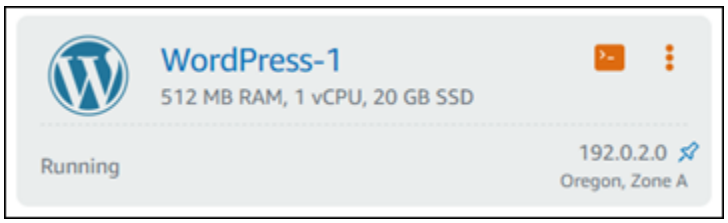
- インスタンスのパブリック IP アドレスは変わることがあります。インスタンスに静的 IP を添付することで、変更されない IP アドレスを割り当てます。詳細については、このガイドのインスタンスの静的 IP アドレスセクションを参照してください。
- Lightsail はデフォルトでIPv4アドレスを使用します。ただし、2021 年 1 月 12 日より前に作成された一部の Lightsail リソースIPv6では、オプションで を有効にできます。2021 年 1 月 12 日以降に作成されたリソースは、デフォルトでIPv6有効になっています。詳細については、このガイドIPv6の「インスタンス、コンテナサービス、CDNディストリビューション、ロードバランサー」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パブリックIPv4アドレスは、インスタンスを停止して起動すると変更されます。オプションで、静的IPv4アドレスを作成してインスタンスにアタッチできます。静的IPv4アドレスはインスタンスのデフォルトのパブリックIPv4アドレスを置き換え、インスタンスを停止して起動しても同じままになります。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 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。

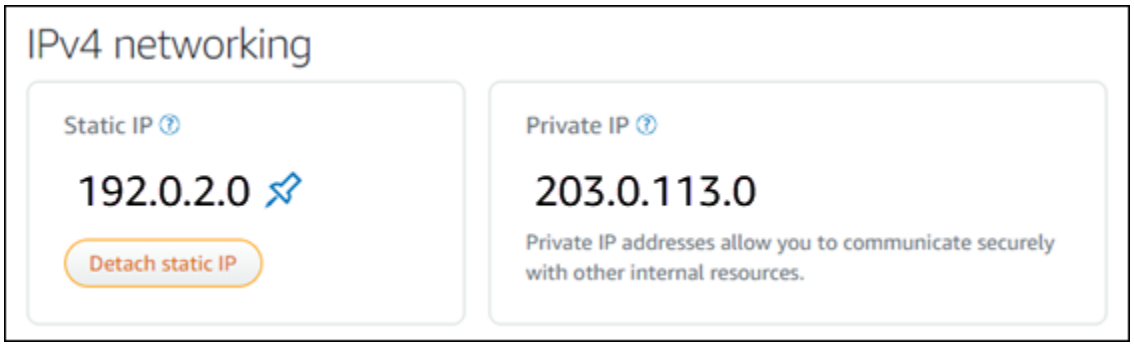
静的 IP を作成してインスタンスにアタッチすると、Lightsail コンソールの次の領域に表示されます。

- 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。

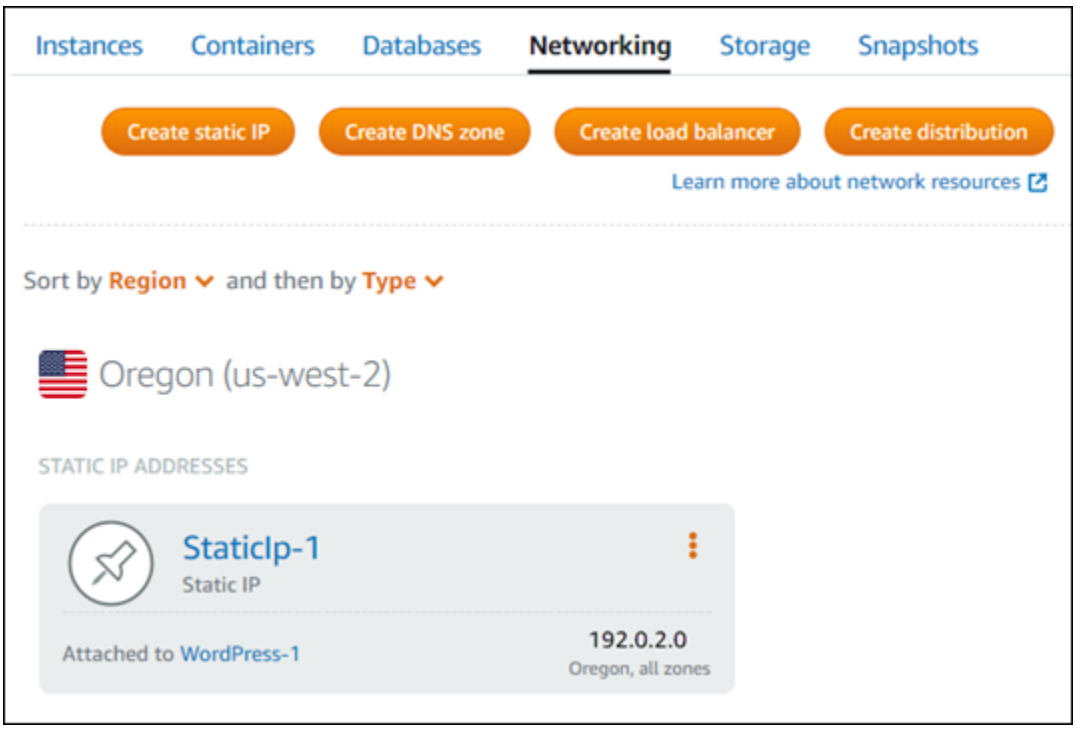


- 次の例は、インスタンス管理ページのヘッダー領域にあるインスタンスの静的 IP アドレスを示しています。画鋲アイコンは、パブリック IP アドレスが静的であることを示します。

- 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、IPs作成したすべての静的 を表示できます。



IPv6 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、ロードバランサー用の

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、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。オプションで、2021 年 1 月 12 日より前に作成されたリソースIPv6に対して を有効にできます。特定のリソースIPv6に対して 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。自分でIPv6アドレスを選択または指定することはできません。詳細については、「 を有効または無効にするIPv6」を参照してください。

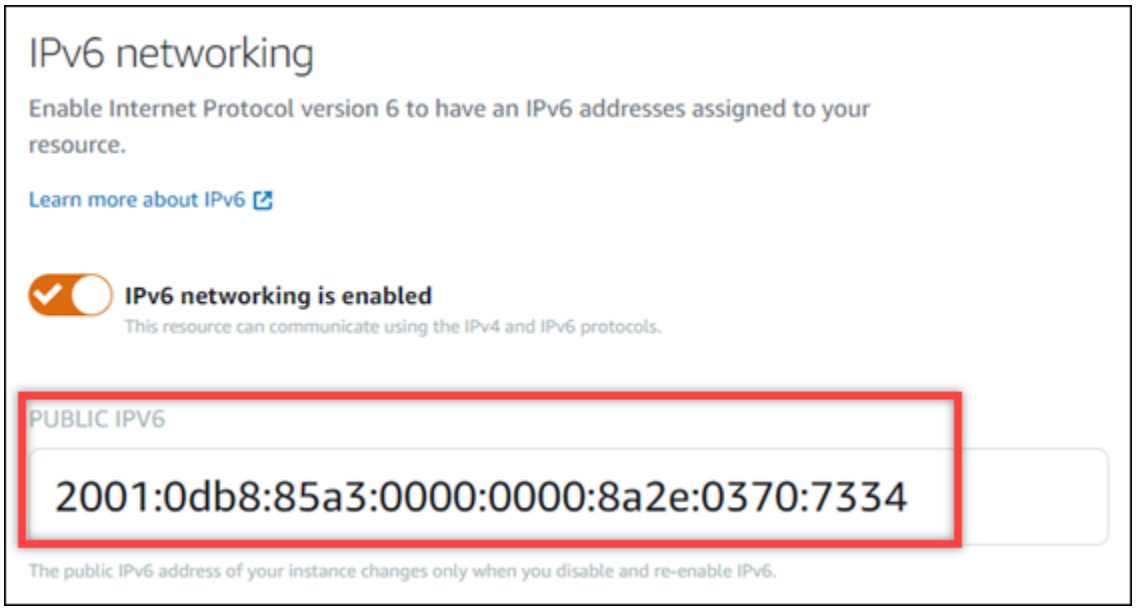
IPv6のみのインスタンスを作成することもできます。のみIPv6のインスタンスは、 経由でIPv6のみパブリック通信でき、パブリックIPv4アドレスはありません。詳細については、「Lightsail インスタンスの IPv6-onlyネットワークを設定する」を参照してください

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エリアにあるリソースの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 を有効にして使用する場合IPv6は、次の点に注意してください。

- リソースIPv6に対して を有効にすると、 IPv4および IPv6 (デュアルスタックモード) を介して、または を介してIPv4のみ、リソースと通信できます。

- リソースIPv6に対して 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。自分でIPv6アドレスを選択または指定することはできません。リソースIPv6に対して を有効にすると、IPv6プロトコルを介した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起動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IPv6に対して を無効にした場合にのみリリースされます。これらのアクションのいずれかを実行した後で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ているすべてのIPv6アドレスはパブリックであり、インターネット経由でアクセスできます。インスタンスに割り当てられるプライベートIPv6アドレスはありません。
- IPv4 インスタンスの と IPv6 アドレスは互いに独立しています。 IPv4と に対してインスタンス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。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、 が有効になっているIPv6ときにに自動的に設定IPv6されるわけではありません。次のブループリントを使用するインスタンスでは、 を有効にした後、追加の設定ステップが必要ですIPv6。
  - cPanel – 詳細については、cPanel「インスタンス の設定IPv6」を参照してください。
  - Debian 8 – 詳細については、「Debian 8 インスタンスの設定IPv6」を参照してください。
  - GitLab – 詳細については、「インスタンス の設定IPv6 GitLab」を参照してください。
  - Nginx – 詳細については、「Configure IPv6 for Nginx instances」を参照してください。
  - Plesk – 詳細については、「Plesk インスタンスの設定IPv6」を参照してください。
  - Ubuntu 16 – 詳細については、「Ubuntu 16 インスタンスの設定IPv6」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IPv6に対して を有効にできますが、 PrestaShop ソフトウェアはIPv6ネットワーク経由でリクエストに応答しません。

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 アドレスを設定していない場合、インスタンスを停止または再起動するたびに、Lightsail は新しいパブリック IP アドレスを割り当てます。

> **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。

内容

Lightsail インスタンスに静的 IP を作成してアタッチする

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。後で、登録済みドメイン名をインスタンスにポイントすると、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はありません。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 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 AWS リージョン を作成する 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。

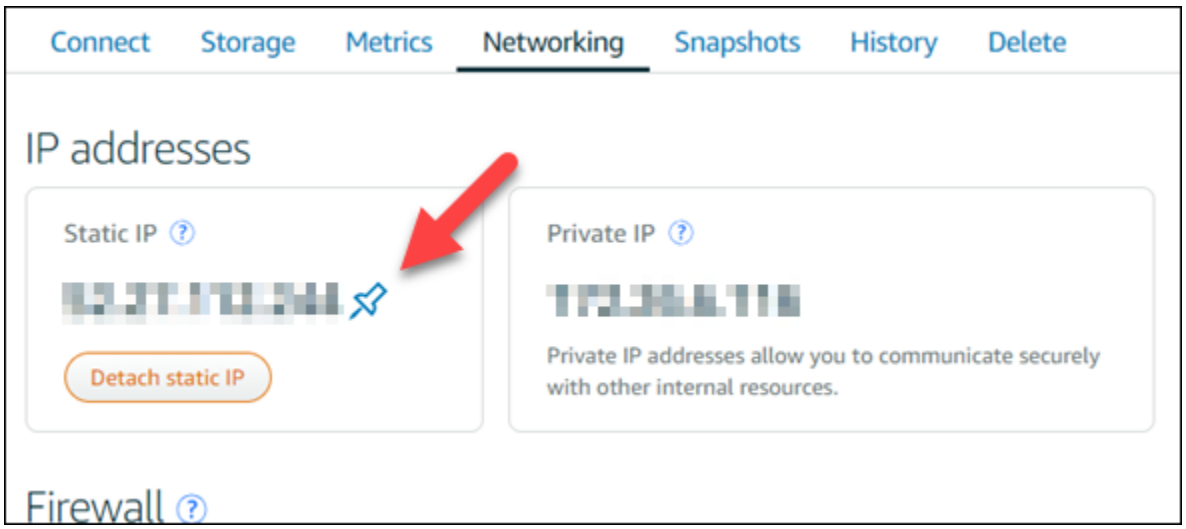
5. 静的 IP をアタッチする Lightsail リソースを選択します。
6. 静的 IP の名前を入力します。

   リソース名:

   - 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. [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

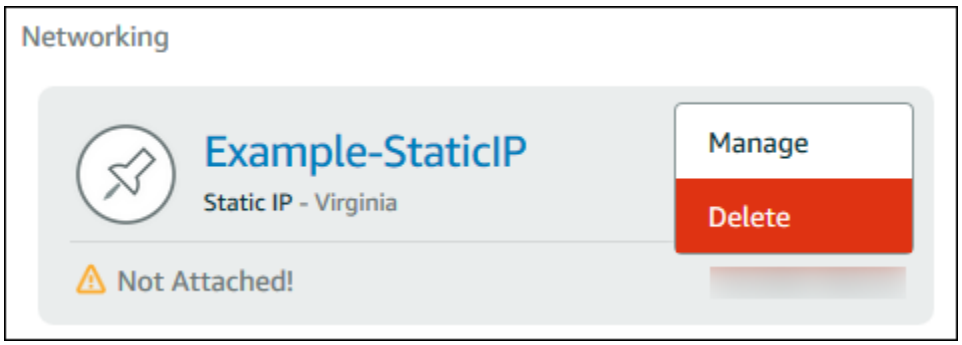
Amazon Lightsail アカウント AWS リージョン では、 IPsごとに最大 5 つの静的 を作成できます。静的 IP アドレスが割り当てられているインスタンスを削除しても、静的 IP アドレスはアカウントに残ります。静的 IP アドレスが不要になった場合は、Lightsail コンソールまたは AWS Command Line Interface () を使用して削除できますAWS CLI。このガイドでは、Lightsail アカウントから静的 IP アドレスを削除する方法について説明します。静的 の詳細についてはIPs、「IP アドレス」を参照してください。

> **Important**
>
> 静的 IP を削除すると、Lightsail アカウントから静的 IP が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、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、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の省略記号 (⋮) アイコンを選択し、「削除」を選択します。



を使用して静的 IP を削除する AWS CLI

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、まだインストールしていない場合は、 をインストールする必要があります AWS CLI。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。これは、 `get-static-ips` AWS CLI コマンド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、*StaticIpName* 静的 IP の名前を入力します。

   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
        }
    ]
}
```

Lightsail リソースのデュアルスタックネットワーク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 デュアルスタックインスタンス、コンテナサービス、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、デュアルスタックインスタンスの IPv6 ネットワークを有効または無効にする方法を示します。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。

デュアルスタックに関する考慮事項

IPv6 は 2021 年 1 月 12 日に Lightsail で利用可能になったため、以下のガイドラインに従って、一部のリソースで 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とロードバランサーは、有効にするまで IPv6 が無効になります。ただし、1 月 12 日以降に作成されたインスタンスと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。コンテナサービスまたはロードバランサーで 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の受け入れを開始します。
- 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。
- IPv6 が有効になっている場合、Lightsail で使用できるすべてのインスタンスブループリントが 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、「インスタンスの IPv6 を設定する GitLab」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。

トピック

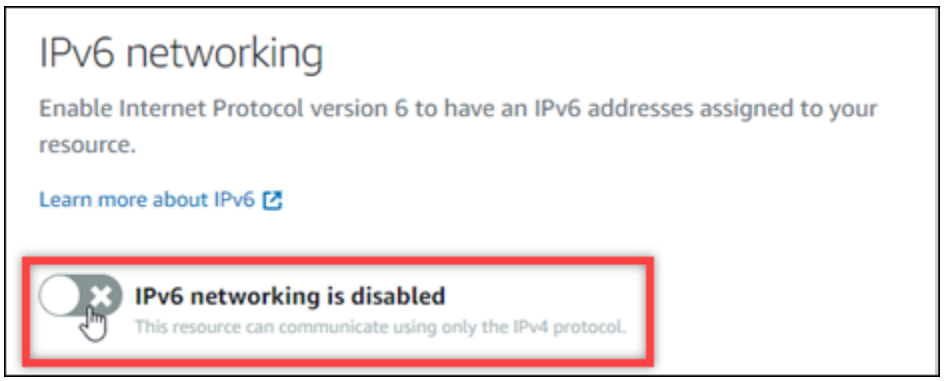
- Lightsail リソースのIPv6ネットワークを有効にする
- Lightsail リソースのIPv6ネットワークを無効にする

Lightsail リソースのIPv6ネットワークを有効にする

インスタンス、CDNディストリビューション、ロードバランサーIPv6に対して を有効にするには、次の手順を実行します。

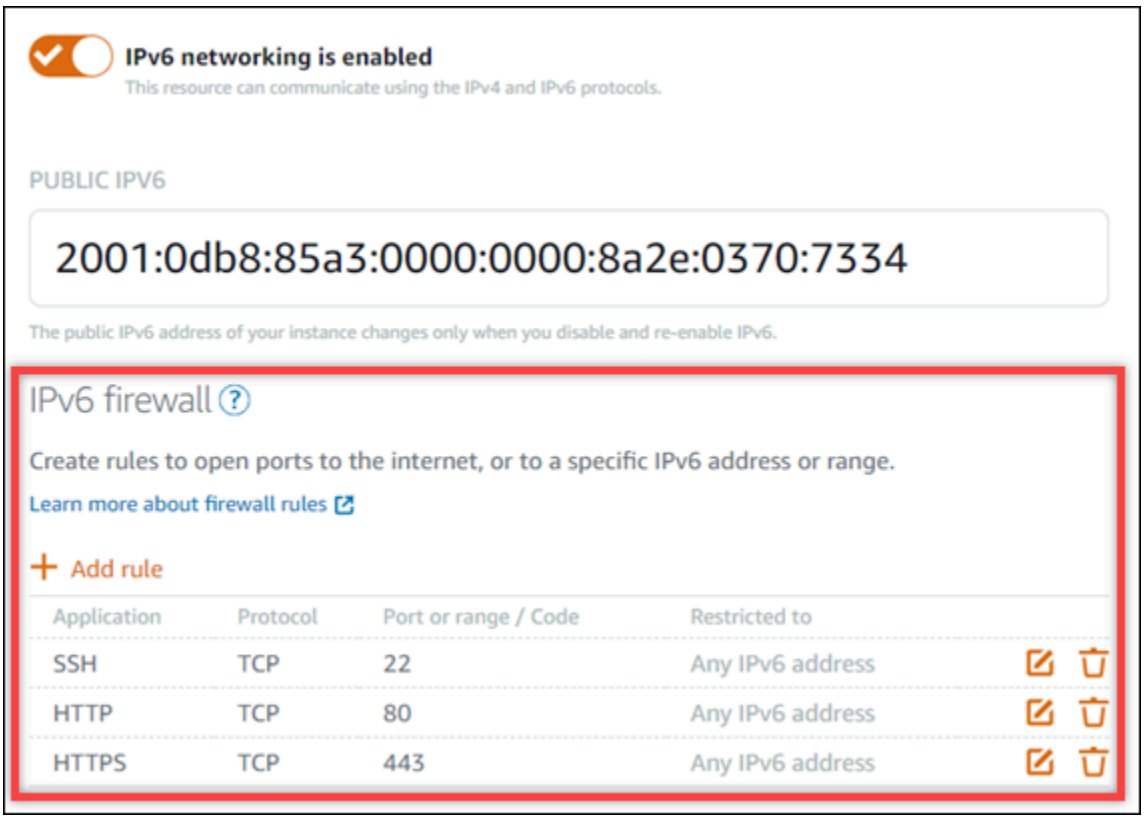
1. Lightsail コンソール にサインインします。
2. を有効にするリソースに応じて、次のいずれかの手順を実行しますIPv6。

   - インスタンスIPv6に対して 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選択し、 を有効にするインスタンスの名前を選択しますIPv6。
   - CDN ディストリビューションまたはロードバランサーIPv6に対して を有効にするには、Lightsail ホームページのネットワークタブを選択し、 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IPv6。
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ネットワークセクションで、トグルを選択してリソースIPv6を有効にします。



リソースに対して を有効にした後は、次の項目IPv6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IPv6に対して を有効にすると、そのリソースはIPv6トラフィックの受け入れを開始します。インスタンスIPv6に対して 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ファイアウォールが使用可能IPv6になります。

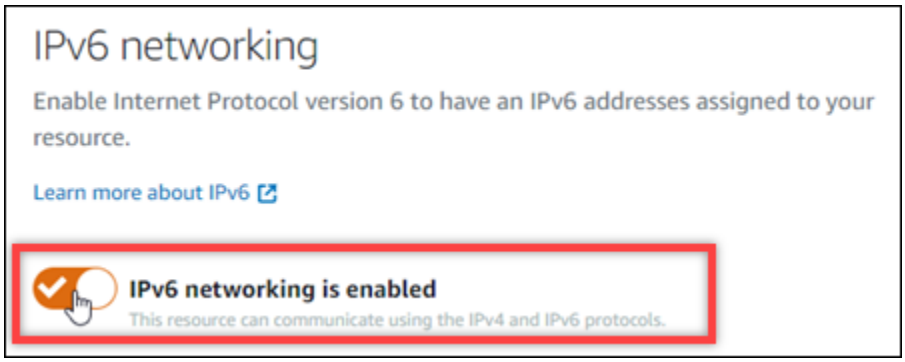


- 次のブループリントを使用するインスタンスは、 を有効にした後、インスタンスが新しいIPv6アドレスを認識IPv6できるように追加のステップが必要です。
  - cPanel – 詳細については、cPanel「インスタンス の設定IPv6」を参照してください。
  - Debian 8 – 詳細については、「Debian 8 インスタンスの設定IPv6」を参照してください。
  - GitLab – 詳細については、「インスタンス の設定IPv6 GitLab」を参照してください。
  - Nginx – 詳細については、「Configure IPv6 for Nginx instances」を参照してください。
  - Plesk – 詳細については、「Plesk インスタンスの設定IPv6」を参照してください。
  - Ubuntu 16 – 詳細については、「Ubuntu 16 インスタンスの設定IPv6」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にトラフィックを送信する登録済みドメイン名がある場合は、ドメインDNSの にIPv6アドレスレコード (AAAA) を作成して、IPv6トラフィックをリソースにルーティングします。

Lightsail リソースのIPv6ネットワークを無効にする

インスタンス、CDNディストリビューション、ロードバランサーIPv6の 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. 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IPv6。
   - インスタンスIPv6の 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選択し、 を無効にするインスタンスの名前を選択しますIPv6。
   - CDN ディストリビューションまたはロードバランサーIPv6の を無効にするには、Lightsail ホームページのネットワークタブを選択し、 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IPv6。
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ネットワークセクションで、トグルを選択してリソースIPv6に対して無効にします。



Lightsail インスタンスの IPv6-onlyネットワークを設定する

Lightsail インスタンスは、デュアルスタックネットワーク (IPv4 および IPv6) と IPv6-only種類のネットワークをサポートします。デュアルスタックネットワークでは、インスタンスにパブリック IPv4 とパブリック IPv6 アドレスが割り当てられます。必要に応じて IPv6 を有効または無効にできます。

IPv6-onlyネットワークでは、インスタンスにはパブリック IPv6 アドレスが割り当てられ、パブリック IPv4 トラフィックはサポートされません。すべての Lightsail ブループリントが IPv6 と互換性があるわけではありません。IPv6-only、「」を参照してくださいIPv6 互換の設計図。

> **Warning**
>
> Amazon Lightsail パブリックエンドポイントは、現時点では IPv6 をサポートしていません。詳細については、「Amazon VPC ユーザーガイド」のIPv6 をサポートするサービス」を参照してください。

パブリック IPv6-onlyネットワークを使用します。 IPv4 ただし、まず、ローカルネットワーク、コンピュータ、デバイス、エンドユーザーが IPv6 を使用して通信できることを確認します。詳細については、「」のIPv6 到達可能性」を参照してくださいLightsail インスタンスの IPv6 到達可能性を検証する。既存のインスタンスのネットワークタイプを変更するには、「」を参照してくださいLightsail でインスタンスネットワークタイプを IPv6またはデュアルスタックに切り替える。

トピック

- Lightsail でインスタンスネットワークタイプを IPv6またはデュアルスタックに切り替える
- IPv6 互換の設計図

Lightsail でインスタンスネットワークタイプを IPv6またはデュアルスタックに切り替える

インスタンスのネットワークタイプによって、インターネットを介した通信に使用するプロトコルが決まります。インスタンスを作成するときは、デュアルスタックネットワークまたは IPv6専用ネットワークのいずれかを選択します。既存のインスタンスのネットワークタイプをデュアルスタックから IPv6のみに変更したり、その逆に変更したりすることもできます。ネットワークタイプを変更するには、ガイド付き、 step-by-step ワークフローを使用するか、個々のステップを実行します。

ガイド付きワークフローでは、新しいネットワークタイプが設定されている間もインスタンスは引き続き実行されます。このオプションは、変更の実行中にインスタンスがインターネット経由でアクセス可能にするために使用します。ただし、まず、ローカルネットワーク、コンピュータ、デバイス、エンドユーザーが を使用して通信できることを確認しますIPv6。詳細については、「Lightsail インスタンスの IPv6 到達可能性を検証する」を参照してください。

個々のステップでは、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。新しいインスタンスを作成するときに、別のネットワークタイプを選択できます。このオプションを使用して、他のインスタンスの設定を変更する前にIPv6互換性を検証します。開始する前に、 を確認することをお勧めしますIPv6のみに関する考慮事項。

IPv6のみに関する考慮事項

次の考慮事項を確認してください。

- インスタンスプランは、ネットワークタイプが変更されるたびに変更されます。詳細については、AWS コンピューティングブログの「Amazon Lightsail でのIPv6インスタンスバンドルの発表と料金の更新 Amazon Lightsail」を参照してください。

- Amazon Lightsail パブリックエンドポイントIPv6は、現時点では をサポートしていません。詳細については、「Amazon ユーザーガイド」の「 をサポートするサービスIPv6」を参照してください。 VPC
- インスタンスは 経由でパブリックに通信しますIPv6。受信または送信パブリックIPv4トラフィックはサポートされません。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アドレスを受け取ります。詳細については、「Lightsail リソースの IP アドレスの表示と管理」を参照してください。
- IPv6のみのインスタンスは、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。
- Lightsail ロードバランサーには IPv6のみのインスタンスを追加できます。
- ネットワークタイプを変更すると、インスタンスのデータ転送プランの許容量が引き継がれます。リセットされません。
- ローカルデバイス、ネットワーク、インターネットサービスプロバイダー (ISP) が IPv6と互換性があることを確認します。詳細については、「Lightsail インスタンスの IPv6 到達可能性を検証する」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

ウィザードを使用してインスタンスネットワークタイプを設定するには

1. インスタンス管理ページの情報パネルで、ネットワークタイプの変更を選択します。
2. ネットワークタイプの選択 で、デュアルスタックまたは IPv6のみを選択します。選択したオプションの下に強調表示されている情報を確認し、次へ を選択します。
3. リソースの確認 で、インスタンスに現在関連付けられているリソースに加えられる変更を確認します。リソースは、静的 IP アドレスでも Lightsail ロードバランサーでもかまいません。インスタンスにアタッチされたリソースがない場合、変更は行われません。リソースの変更は、次のステップでワークフローを完了するまで行われません。[次へ] を選択して続行します。
4. 変更の確認 で、新しいインスタンスのネットワークタイプ、料金、リソースの変更を確認し、変更の確認 を選択します。Lightsail リソースの設定を開始します。
5. (オプション) ワークフローが完了したら、インスタンス設定を更新します。例えば、インスタンスに静的 IP をアタッチするか、 の DNS A レコードIPv4と のAAAAレコードを更新しますIPv6。次のステップについては、このガイドのthe section called “次のステップ”「」セクションを参照してください。

オプション: 個々のステップ

個々のステップを実行してインスタンスネットワークタイプを設定するには

1. インスタンス管理ページのスナップショットタブで、スナップショットの作成 を選択します。詳細については、次のトピックのいずれかを参照してください。
   - スナップショットを使用して Linux/Unix Lightsail インスタンスをバックアップする
   - Lightsail Windows Server インスタンスのスナップショットを作成する
2. スナップショットに名前を付け、 の作成を選択します。
3. スナップショットアクションメニュー (⋮) から、新しいインスタンスの作成 を選択します。詳細については、「スナップショットから Lightsail インスタンスを作成する」を参照してください。
4. 「ネットワークタイプの選択」セクションで、「デュアルスタック」またはIPv6「 のみ」を選択します。
5. 残りのオプションを確認し、インスタンスの作成 を選択します。新しいインスタンスが作成されます。
6. (オプション) ワークフローが完了したら、インスタンス設定を更新します。例えば、インスタンスに静的 IP をアタッチするか、 の DNS A レコードIPv4と のAAAAレコードを更新しますIPv6。次のステップについては、このガイドのthe section called “次のステップ”「」セクションを参照してください。

次のステップ

インスタンスのネットワークタイプを変更した後に実行できる追加のタスクがいくつかあります。

- (IPv6のみ) アプリケーションとユーザーが 経由で通信できることを確認しますIPv6。詳細については、「Lightsail インスタンスの IPv6 到達可能性を検証する」を参照してください。
- (デュアルスタック) インスタンスに静的 IP アドレスをアタッチします。詳細については、「静的 IP をインスタンスにアタッチする」を参照してください。
- (デュアルスタック) Lightsail ディストリビューションのオリジンとしてインスタンスを設定します。詳細については、CDN「Lightsail のディストリビューション」を参照してください。
- (両方) インスタンスのファイアウォール設定を追加または更新します。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。
- (両方) の DNS A レコードIPv4と のAAAAレコードを追加または更新しますIPv6。詳細については、「ドメインをインスタンス にポイントする」を参照してください。

- （両方）Lightsail ロードバランサーにインスタンスを追加します。詳細については、Lightsail の「ロードバランサー」を参照してください。

IPv6 互換の設計図

次の Lightsail ブループリントは、IPv6-onlyインスタンスプランと互換性があります。

- Windows Server 2022
- Windows Server 2019
- Windows Server 2016
- Amazon Linux 2023
- Amazon Linux 2
- AlmaLinux OS 9
- CentOS Stream 9
- Debian 11, and 12
- FreeBSD 13
- Ubuntu 20, and 22
- SQL Server 2022 Express
- SQL Server 2019 Express
- SQL Server 2016 Express
- LAMP stack (PHP 8) packaged by Bitnami
- MEAN stack packaged by Bitnami
- Redmine packaged by Bitnami

Lightsail ブループリントの詳細については、「」を参照してくださいthe section called “設計図”。

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 AWS リージョン 近い でリソースを作成します。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、次の で使用できます AWS リージョン。

- 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、 でインスタンスを作成するとすぐに AWS リージョン、そのリージョンにデフォルト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

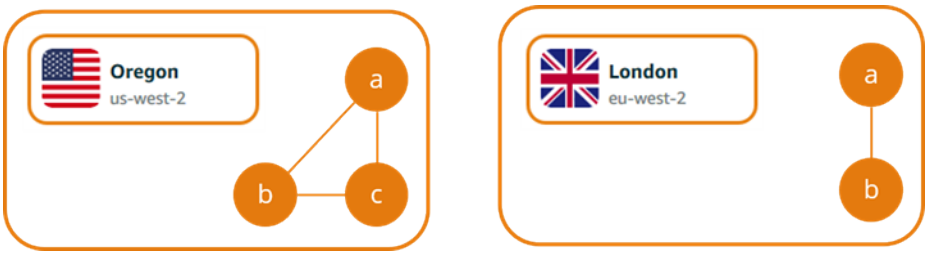
各 AWS リージョン は、他の から完全に分離されるように設計されています AWS リージョン。これにより、最大限の耐障害性と安定性が達成されます。

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場合は、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。コマンドで `--region`オプションを使用して、DNSゾーンとネットワークリソースに関する情報を返す AWS CLI `us-east-1`には を指定します。オプションの使用の詳細については、 AWS CLI `--region`「 AWS CLI リファレンス」の「一般的なオプション」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、リージョン名 () に続く文字で示される複数の分離されたアベイラビリティーゾーンがありますus-east-2a。Lightsail インスタンスは、一度に 1 つのアベイラビリティーゾーンにのみ作成できます。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 「 およびアベイラビリティーゾーン」を参照してください。

VPC ピアリングを使用して Lightsail リソースを AWS サービスに接続する

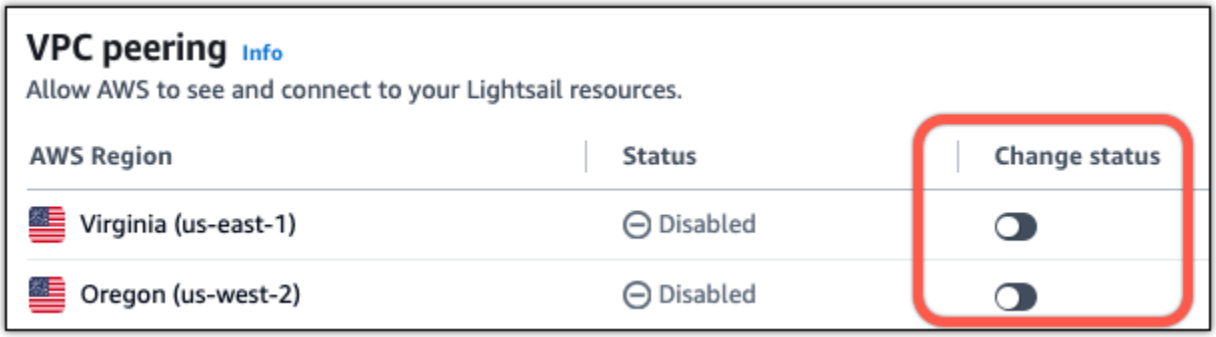
Lightsail を使用すると、仮想プライベートクラウド (VPC) ピアリングを介して Amazon RDS データベースなどの AWS リソースに接続できます。VPC は、 AWS アカウント専用の仮想ネットワークです。Lightsail 内に作成するものはすべて 内にありVPC、Lightsail を VPC Amazon に接続できますVPC。

Amazon S3、Amazon 、Amazon DynamoDB などの一部の AWS リソースでは CloudFront、VPCピアリングを有効にする必要はありません。

> **Note**
>
> Lightsail でVPCピアリングを有効にするには、デフォルトの Amazon が必要ですVPC。デフォルトの Amazon がない場合はVPC、作成できます。詳細については、「Amazon ユーザーガイド」のVPC「デフォルトの作成」を参照してください。 VPC
> AWS リージョンは互いに分離されているため、 VPCはそれを作成したリージョンでも分離されます。Lightsail リソースがある各リージョンでVPCピアリングを有効にする必要があります。

デフォルトの Amazon を作成したらVPC、以下の手順に従って Lightsail を Amazon VPCとピアリングしますVPC。

1. Lightsail コンソール で、上部のナビゲーションメニューでユーザー名を選択します。
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. VPC ピアリングを有効にする AWS リージョン の横にあるステータスを切り替えます。



ピアリング接続が失敗した場合は、VPCピアリングを再度有効にしてみてください。機能しない場合は、 にお問い合わせくださいAWS Support。

ピアリングリクエストが成功すると、 AWS アカウントにピアリング接続が作成されます。Amazon VPC Dashboard に移動し、ナビゲーションペインでピアリング接続を選択して、作成されたピアリング接続を表示します。

Amazon の詳細についてはVPC、「Amazon VPCユーザーガイド」のVPC「」および「サブネット」を参照してください。

SSLLightsail の /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 () 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 CDN) ドメインを検証します。これらの Lightsail リソースのいずれかに検証済み証明書がアタッチされると、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、Hypertext Transfer Protocol Secure () を使用して暗号化されますHTTPS。

Amazon Lightsail で Transport Layer Security (TLS) 証明書を作成して、Lightsail ロードバランサーコンテンツ配信ネットワーク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できます。TLS は、Secure Socket

Layer () のより安全な更新バージョンですSSL。Lightsail のドキュメントとコンソール全体では、/ と呼ばれていますSSLTLS。

> **Important**
>
> 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、 AWS Certificate Manager （ ACM) サービスによって発行されます。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、 CAsがACM管理する複数の中間認証機関 (ICAs) または下位機関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 が動的中間認証機関を導入する」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 は、 を使用してデータを移動するためTLS、セキュリティを強化します。HTTPS 暗号化は、トラフィックを復号できる唯一の 2 つのエンティティであるため、ウェブサーバーとクライアントのブラウザの間では機密です。HTTPS また、クライアントがサーバーと交換するデータは、別の当事者によって変更できないため、 接続はより安全です。

上記のセキュリティ上の利点以外にも、 HTTPS以外にも を使用する理由がありますHTTP。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、 を使用するサイトはHTTPS、 のみを使用するサイトと比較して、検索結果の上部に近いランクになります HTTP (他のすべてのモノは等しい) 。

ランキングシグナルHTTPSとしての の詳細はこちら

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 に正規名 (CNAME) DNS レコードを追加して証明書を検証する
4. 検証済みの証明書を Lightsail リソースにアタッチします。

5. ドメインDNSの 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アタッチされると、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 を使用して暗号化されますHTTPS。

ディストリビューションまたはコンテナサービスで SSL/TLS 証明書を使用する

HTTPS は、Lightsail ディストリビューションとコンテナサービスで必要です。これらのリソースのいずれかを作成すると、リソースのデフォルトドメイン (ディストリビューションまたはコンテナサービスhttps://123456abcdef.cloudfront.net/など) https://container-service-1.123456abcdef.us-west-2.cs.amazonlightsail.com/に対して HTTPSがデフォルトで有効になります。ディストリビューションまたはコンテナサービスで登録済みドメイン名 (例: example.com) を使用する場合は、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。

これらのリンクから、ディストリビューションHTTPSでカスタムドメインと を有効にすることができます。

- 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 証明書を検証する
- 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。

これらのリンクから、コンテナサービスHTTPSでカスタムドメインと を有効にすることができます。

- コンテナサービスSSL/TLS証明書を作成する
- コンテナサービス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 SSL/TLS 証明書を使用する

Lightsail ロードバランサーを作成すると、ポート 80 はデフォルトで通常のHTTPトラフィックを処理するために開かれます。ポート 443 経由の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て、ロードバランサーにアタッチする必要があります。

ロードバランサーごとに最大 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。ロードバランサーから有効な使用中の証明書を削除すると、別の有効な証明書をアタッチするまで、ロードバランサーは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HTTPSで を有効にするには、次のリンクから開始できます。

- 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
- 検証済み証明書をアタッチして有効にする HTTPS

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。

安全な Lightsail コンテナサービスドメイン用の SSL/TLS 証明書を作成する

Lightsail コンテナサービス用に 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 Service Quotas 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

スタートする前に、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 のホームページで [Containers] (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。

   他の Lightsail リソース用に作成された証明書や、使用中で使用されていない証明書など、すべての証明書はページの「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 [Attempting to validate your certificate] (証明書の検証を試行しています) 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 [Valid] (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、作成後、Lightsail コンテナサービスで使用する前に検証する必要があります。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、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しようとします。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手順について説明します。Domain.com や など、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はGoDaddy似ている場合があります。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 のホームページで [Containers] (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。

   他の Lightsail リソース用に作成された証明書や検証保留中の証明書など、すべての証明書はページの「アタッチ済み証明書」セクションに表示されます。
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 & 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 コンソールは、ドメインの頂点部分を事前に入力します。たとえば、`www.example.com` サブドメインを追加する場合は、`www` 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 のホームページで、[Containers] (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。

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、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。

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 コンソールで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

スタートする前に、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 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。

SSL/TLS 証明書による Lightsail CDN ディストリビューションの保護

Lightsail ディストリビューションの 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 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) アカウントで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。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

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」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 証明書を検証する
- Lightsail ディストリビューションを最小 TLS プロトコルバージョンで保護する

- Lightsail ディストリビューションから未使用の 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

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」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 証明書を検証する

Amazon Lightsail SSL/TLS 証明書は、作成後、Lightsail ディストリビューションで使用する前に検証する必要があります。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、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しようとします。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手順について説明します。Domain.com や など、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は似ている GoDaddy場合があります。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

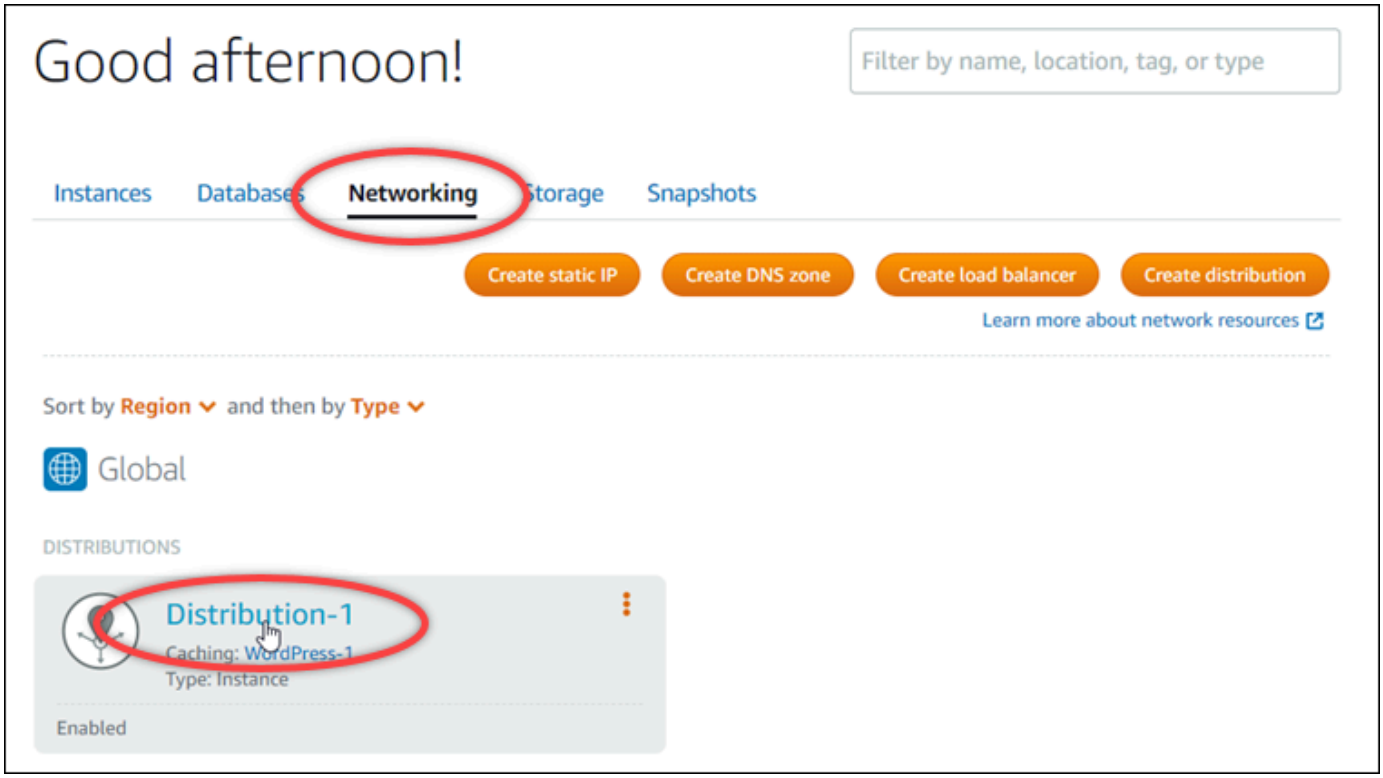
前提条件

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。

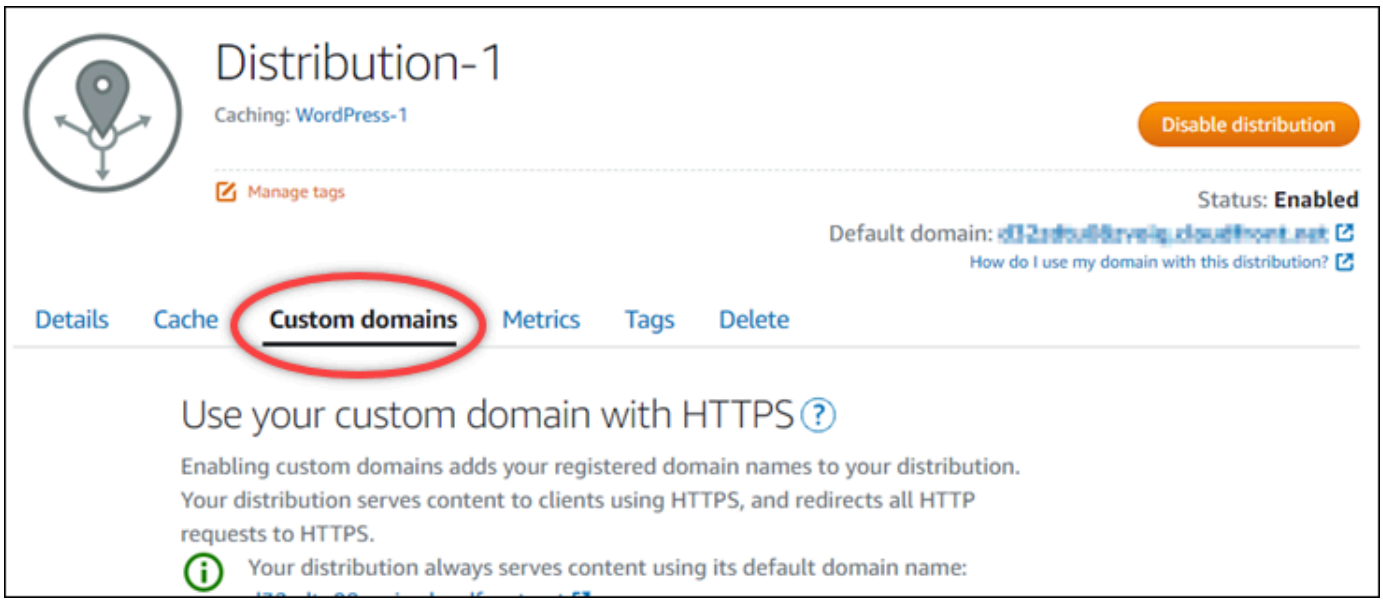
CNAME レコードの値を取得して証明書を検証する

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 [Networking] (ネットワーク) 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は、ページの「アタッチされた証明書」セクションに表示されます。

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 & 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 コンソールは、ドメインの頂点部分を事前に入力します。たとえば、`www.example.com` サブドメインを追加する場合は、`www` 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。

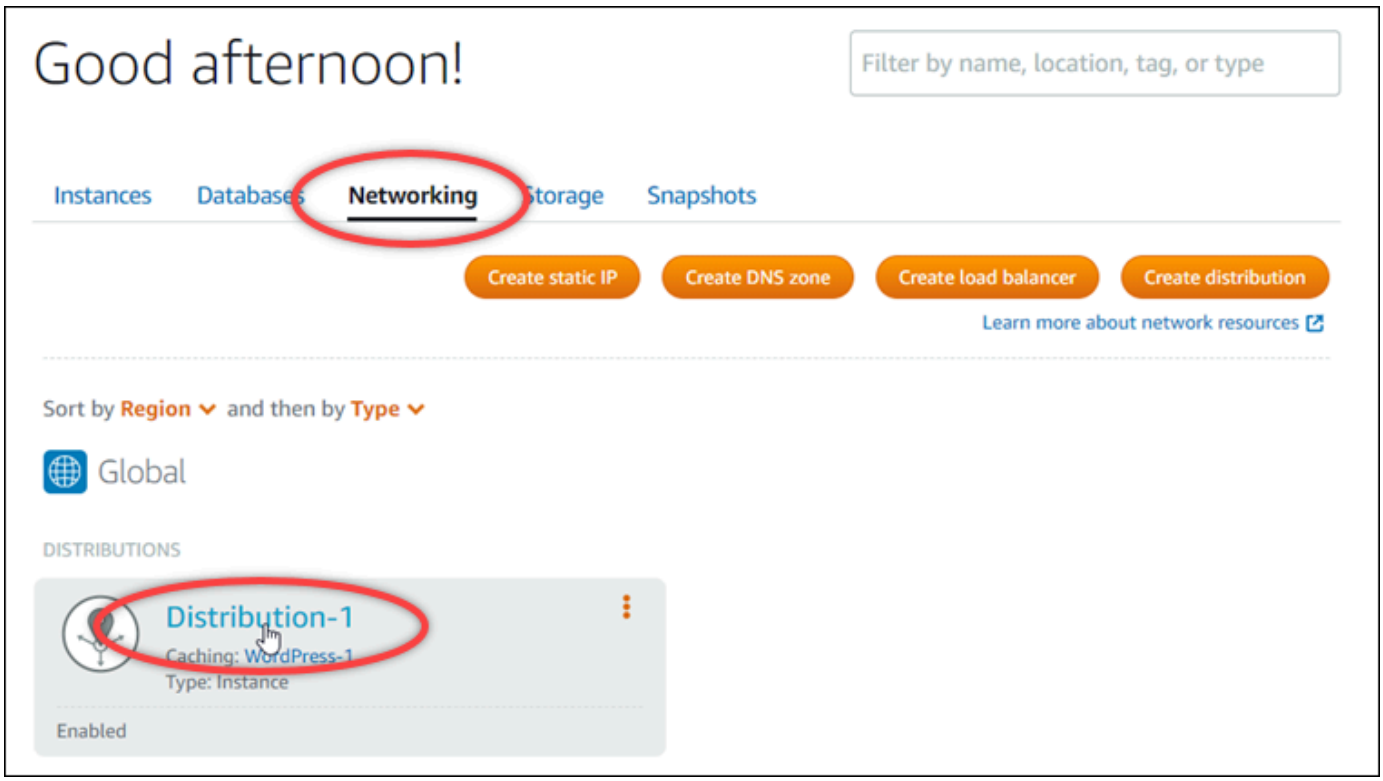
9. 保存アイコンを選択して、レコードを DNS ゾーンに保存します。

   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。

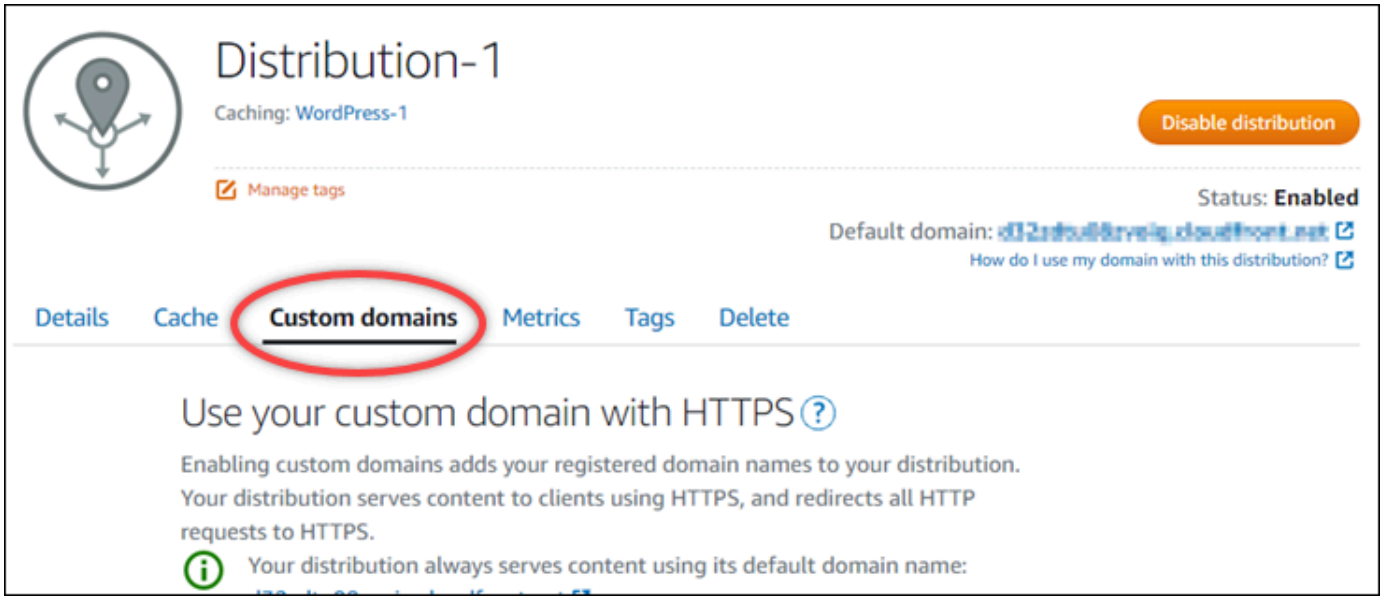
ディストリビューション証明書のステータスを表示する

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 [Networking] (ネットワーク) 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。

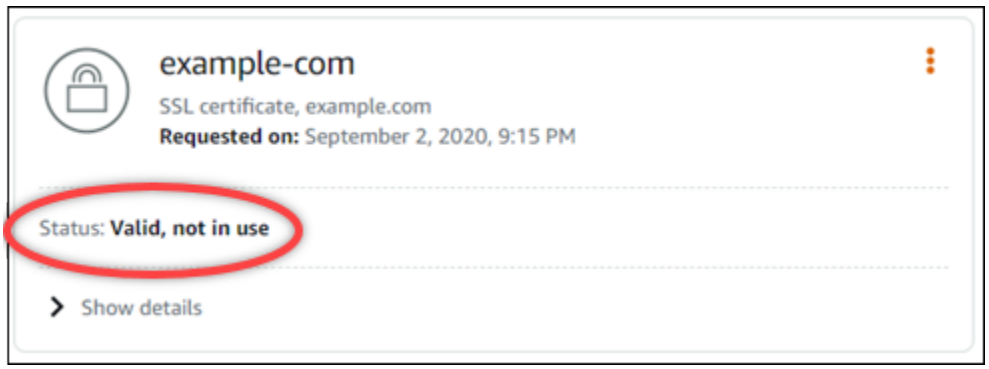


3. ディストリビューションの管理ページのカスタムドメインタブを選択します。



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。

   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、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。

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を最小 TLS プロトコルバージョンで保護する

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 証明書用に設定できるビューワーの最小 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。ビューワーは、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、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、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、次の 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を作成する
- 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ポイントする

ディストリビューションの最小 TLS プロトコルバージョンを特定する

Lightsail ディストリビューションの最小 TLS プロトコルバージョンを特定するには、次のステップを実行します。

> **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、 PowerShellZ シェルなどの任意のシェルを使用して 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。

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最小 TLS プロトコル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、 を、変更するディストリビューションの名前*DistributionName*に置き換えます。

例

```
aws lightsail get-distributions --distribution-name Distribution-1 --region us-east-1 | grep "viewerMinimumTlsProtocolVersion"
```

コマンドは、ディストリビューションの最小 TLS プロトコルバージョンの 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する AWS CLI

AWS Command Line Interface () を使用して TLS プロトコルのバージョンを設定するには、以下の手順を実行しますAWS CLI。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、AWS CLI 「 コマンドリファレンス」の「update-distribution」属性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. ディストリビューションの最小 TLS プロトコルバージョンを変更するには、次のコマンドを入力します。

```
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。
- *ProtocolVersion* 有効な TLS プロトコルバージョンを持つ 。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から未使用の 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、「」の「Lightsail サービスクォータ」を参照してくださいAWS 全般のリファレンス。

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 [Networking] (ネットワーク) 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。

Lightsail ロードバランサーの SSL/TLS 証明書HTTPSで を有効にする

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 を有効にできますHTTPS。SSL/TLS 証明書を使用すると、ロードバランサーは暗号化されたウェブトラフィックを処理できるため、ユーザーにより安全なエクスペリエンスを提供できます。詳細については、SSL「/TLS Certificates」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 ホームページで、ネットワーク 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、SSL「/TLS 証明書に代替ドメインとサブドメインを追加する」を参照してください。
8. [証明書の作成] を選択します。

Lightsail は検証プロセスを開始します。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。ドメインと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

- Lightsail SSL/TLS 証明書に代替ドメインとサブドメインを追加する
- Lightsail でCNAMEレコードを使用してドメインを検証SSL/TLS証明書する
- Lightsail ロードバランサーに検証済み SSL/TLS 証明書をアタッチする
- Lightsail ロードバランサーから SSL/TLS 証明書を削除する

Lightsail SSL/TLS 証明書に代替ドメインとサブドメインを追加する

Lightsail ロードバランサーの SSL/TLS 証明書を作成するときに、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 [Networking] (ネットワーク) 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

  検証したら、検証済みの証明書を選択して 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

Lightsail でCNAMEレコードを使用してドメインを検証SSL/TLS証明書する

Lightsail で SSL/TLS 証明書を作成したら、証明書に追加したすべてのドメインとサブドメインを制御し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

ステップ 1: ドメインの Lightsail DNS ゾーンを作成する

まだ作成していない場合は、ドメインの Lightsail DNS ゾーンを作成します。詳細については、「ドメインのDNSレコードを管理するDNSゾーンを作成する」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

作成した証明書は、一連の正規名 (CNAME) レコードを提供します。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または管理していることを確認します。

> **Important**
>
> Lightsail は、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をユーザーが制御していることを自動的に検証しようとします。証明書の作成 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。

次のステップでは、Lightsail コンソールでCNAMEレコードを取得し、ドメインのDNSゾーンに追加する方法を示します。

1. Lightsail コンソール にサインインします。
2. Lightsail ホームページのトップ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. 検証する証明書を検索し、各ドメインに追加する必要があるCNAMEレコードの名前と値を書き留めます。

   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、または Mac TextEdit を使用している場合は、メモ帳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は、これらのCNAME値が必要になります。

7. Lightsail コンソールの上部のナビゲーションバーでホームを選択します。
8. Lightsail ホームページでドメインと 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. レコードタブでレコードの追加を選択します。 DNS
11. レコードタイプCNAMEに 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` と指定します。
13. DNS レコードページに切り替え、名前をレコード名フィールドに貼り付けます。

> **Important**
>
> ドメイン名 (など`.example.com`) を含むCNAMEレコードを追加すると、ドメイン名 (など) が重複します`.example.com.example.com`。重複を回避

するには、CNAME必要な の部分のみが追加されるようにエントリを編集します。_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. と指定します。
15. DNS レコードページに切り替わり、 値をルートトラフィックの 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。

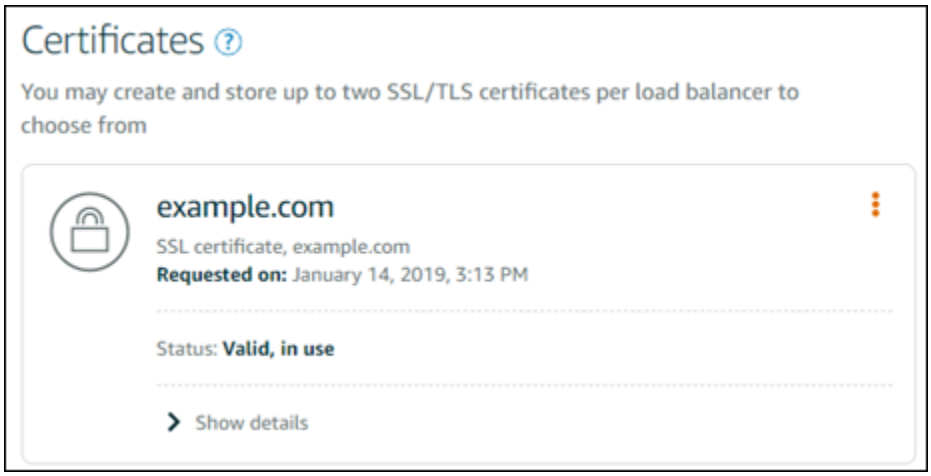
> **Note**
>
> 代替ドメインまたはサブドメインの詳細については、「Amazon Lightsail の SSL/TLS 証明書に代替ドメインとサブドメインを追加する Amazon Lightsail」を参照してください。

18. ステップ 11 ~ 17 を繰り返して、代替サブドメインのCNAMEレコードを追加します (複数可) 。

    DNS ゾーン管理ページで、ロードバランサー または他の Lightsail リソースを指すエイリアス (A) レコードを追加することもできます。

    完了すると、DNSゾーンは次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。

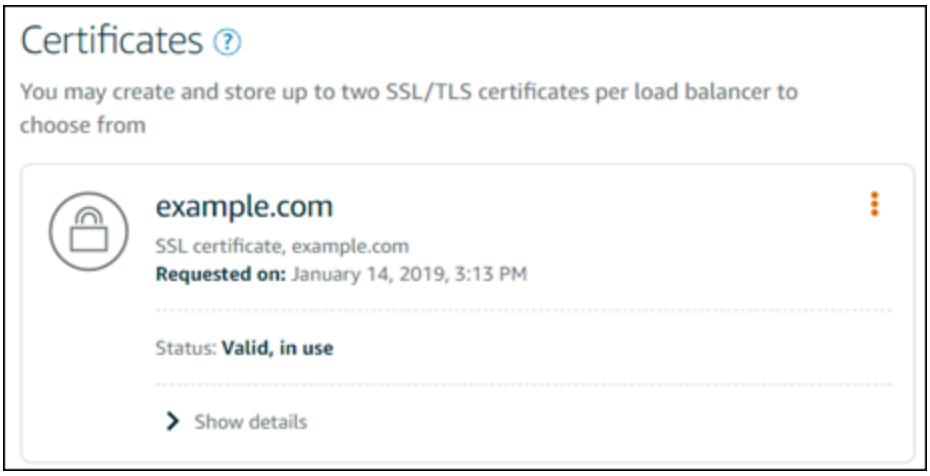


次のステップ

ドメインが検証されたら、検証済み 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。

Lightsail ロードバランサーに検証済み SSL/TLS 証明書をアタッチする

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、証明書を Lightsail ロードバランサーにアタッチします。

1. Lightsail ホームページから、ネットワーク 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. アタッチ を選択し、証明書をアタッチします。

Lightsail ロードバランサーから 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使用中の場合、ロードバランサーは暗号化された (HTTPS) トラフィックを処理できなくなります。Lightsail ロードバランサーは、暗号化されていない (HTTP) トラフィックを引き続きサポートします。

SSL/TLS 証明書の削除は最終であり、元に戻すことはできません。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 ホームページで、ネットワーク 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HTTPSで を無効にする必要があります。

Lightsail インスタンスの E メールスパムを防ぐようにリバース 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、前提条件を満たしてから、AWS サポートにアウトバウンドメッセージングクォータを削除するリクエストを送信します。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 に転送することをお勧めします。これにより、ドメインを含むすべてのリソースを Lightsail コンソールの 1 か所で管理できます。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 はデフォルトでポート 25 を介してアウトバウンドメッセージを制限します。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. AWS アカウントのルートユーザーとして Lightsail コンソールにサインインします。

> **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 IPs や Lightsail インスタンスなどの Lightsail リソースに 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、AWS サポートにリクエストを送信して、逆引き DNS 設定を削除する必要があります。逆引き DNS 設定が削除されたら、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。

Lightsail オブジェクトストレージバケットを使用したデータの保存と管理

Amazon Lightsail オブジェクトストレージサービスを使用して、いつでもインターネット上のどこからでもオブジェクトを保存および取得できます。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を使用すると、Amazon が独自のウェブサイトのグローバルネットワークを実行するために使用する、スケーラビリティ、信頼性、高速性、安価なデータストレージインフラストラクチャにアクセスできます。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

Lightsail オブジェクトストレージには、次の概念と用語が適用されます。

バケット

バケットは、Lightsail オブジェクトストレージサービスに保存されているオブジェクトのコンテナです。すべてのオブジェクトは、独自の を持つバケットに含まれていますURL。例えば、 という名前のオブジェクトmedia/sailbot.jpg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ている場合、 に似URLた を使用してアドレス指定できますhttps://DOC-EXAMPLE-BUCKET.us-east-1.amazonaws.com/media/sailbot.jpg。

Lightsail AWS リージョン が利用可能な でバケットを作成できます。 AWS リージョン Lightsail が利用可能な 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

のバンドルと呼ばれるストレージプランは AWS API、バケットの月額コスト、ストレージ領域、データ転送クォータを指定します。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、毎月の AWS 請求サイクル内で 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に対して不透明です。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、デフォルトのメタデータ (最終更新日など) と標準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。したがって、Lightsail オブジェクトストレージは、「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的な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すべてのオブジェクトは、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て一意にアドレス指定できます。例えば、では`https://DOC-EXAMPLE-BUCKET.us-east-1.amazonaws.com/media/sailbot.jpg`、URL `DOC-EXAMPLE-BUCKET`はバケットの名前、 `media/sailbot.jpg` はオブジェクトキー名です。

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。つまり、バケット所有者、それを作成した 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。Lightsail インスタンスをバケットにアタッチするか、バケットのアクセスキーを作成することで、プログラムによるフルアクセスを許可することもできます。最後に、他の AWS アカウントに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できます。

AWS リージョン

Lightsail オブジェクトストレージバケットは、Lightsail が利用可能なすべての AWS リージョン に作成できます。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。に保存されているオブジェクトは、明示的に別のリージョンに転送しない限り、リージョンを離れ AWS リージョン 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に焦点を当てた最小限の機能セットで意図的に構築されています。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。バケットは Lightsail オブジェクトストレージサービスの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 を使用してバケットにファイルをアップロードします 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。

- 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、ウェブサイトのイメージと添付ファイルを保存します。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バケットを指定することもできます。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: 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バケットを使用する」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。

オブジェクトストレージ用の 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できたら、Amazon Lightsail オブジェクトストレージサービスに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する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、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 AWS リージョン と同じ に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、毎月の AWS 請求サイクル内で 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 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

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」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」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 でバケット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の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バケットのアクセスログ記録を有効にする
   - 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」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 のオブジェクトキー名について」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 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」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」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」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」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: Amazon Lightsail バケットへの WordPress インスタンスの接続
    - 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 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」を参照してください。

Lightsail オブジェクトストレージ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バケットを使用しなくなった場合は、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。例えば、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です。

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 サービスによって、これらの条件のうち、どの条件が存在するかを尋ねられます。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。を使用する場合は AWS CLI、`delete-bucket`リクエストを行うときに `--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、「Lightsail コンソールを使用したバケットの削除」および「このガイド」のセクションを使用したバケットの削除 AWS CLI」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、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、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。

を使用してバケットを削除する AWS CLI

AWS Command Line Interface () を使用してバケットを削除するには、次の手順を実行しますAWS CLI。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、 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 で動作する AWS CLI 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. 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* は、削除するバケットの名前で指定します。

   例:

```
aws lightsail delete-bucket --bucket-name amzn-s3-demo-bucket
```

以下の例のような結果が表示されるはずです。

```
C:\>aws lightsail delete-bucket --bucket-name DOC-EXAMPLE-BUCKET
{
    "operations": [
        {
            "id": "6example-4d30-4442-ae9a-examplef4f52",
            "resourceName": "DOC-EXAMPLE-BUCKET",
            "resourceType": "Bucket",
            "createdAt": "2021-06-30T13:42:43.873000-07:00",
            "location": {
                "availabilityZone": "all",
                "regionName": "us-west-2"
            },
            "isTerminal": true,
            "operationDetails": "62example362/DOC-EXAMPLE-BUCKET/small_1_0",
            "operationType": "DeleteBucket",
            "status": "Succeeded",
            "statusChangedAt": "2021-06-30T13:42:43.873000-07:00",
            "errorCode": "",
            "errorDetails": ""
        }
    ]
}
```

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」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、アクセスキーの作成、バケットへのインスタンスのアタッチ、他のAWSアカウントへのアクセスの付与によって、

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」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 でバケット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の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バケットのアクセスログ記録を有効にする
   - 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、IAMAmazon Lightsail」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について」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」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」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」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」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: Amazon Lightsail バケットへの WordPress インスタンスの接続
    - 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 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」を参照してください。

Lightsail オブジェクトストレージバケット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 APIsを使用してバケットへの完全な読み取り/書き込みアクセスを許可できます。 AWS SDKs 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 ホームページで、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、サービスおよびキーが最後に使用された AWS リージョン 場所を表示します。

Lightsail バケットとオブジェクトへのパブリックアクセスを制限する

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ブロックパブリックアクセスは、既存の個々のバケットとオブジェクトのアクセス許可に関係なく、 内のすべての S3 バケットを AWS アカウント プライベートにすることができます。

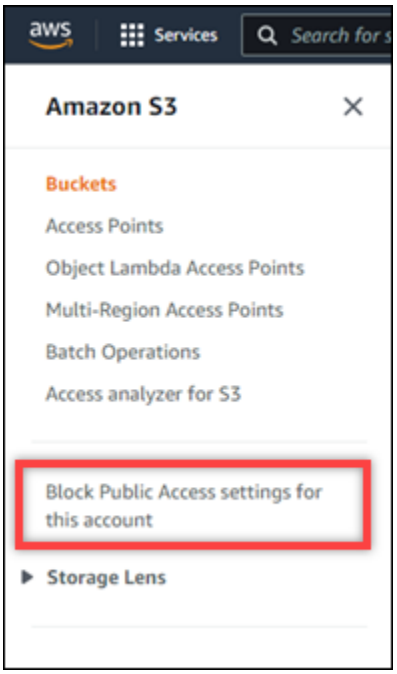
パブリックアクセスを許可または拒否する場合、Lightsail オブジェクトストレージバケットは以下を考慮します。

- Lightsail バケットのアクセス許可。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ブロックパブリックアクセス設定。Lightsail バケットのアクセス許可を上書きします。

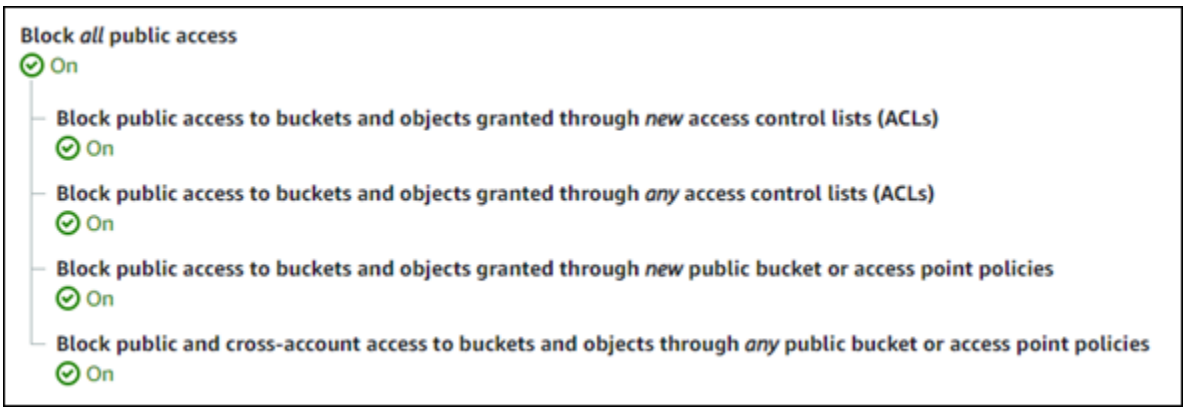
アカウントレベルの Amazon S3 ですべてのパブリックアクセスをブロックをオンにすると、パブリック Lightsail バケットとオブジェクトはプライベートになり、パブリックアクセスできなくなります。 Amazon S3

アカウントのブロックパブリックアクセス設定の構成

Amazon S3 コンソール、 AWS Command Line Interface （ AWS CLI ）、 AWS SDKs、および REST API を使用して、ブロックパブリックアクセス設定を設定できます。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。以下のセクションでは、S3 バケットと Lightsail バケットに対する各設定の効果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 SDKs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**
>
> WordPress オフロード Media Light プラグインなどの Lightsail バケットにメディアを配置する プラグインは、この設定がオンになっていると動作を停止することがあります。これは、ほとんどの WordPress プラグインが オブジェクトに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 プラグインも機能しなくなる可能性があるためです。 ACLs

- 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すべてのオブジェクトに設定することはパブリックで読み取り専用であるか、個々のオブジェクトのアクセス許可をパブリック (読み取り専用) に設定することは、パブリック ACL をどちらかに配置するのと同じです。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とオブジェクトへ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、すべてのオブジェクトがパブリックで読み取り専用のバケットアクセス許可が Lightsail バケットに設定されないようにします。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。
- 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介してバケットとオブジェクトへのパブリックアクセスとクロスアカウントアクセスをブロックする — 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、すべての Lightsail バケットがパブリックで読み取り専用のバケットアクセス許可で設定されていても、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**
>
> この設定では、Lightsail バケットに設定されたクロスアカウントアクセスもブロックされます。このアクセスは、Lightsail のすべてのオブジェクトがパブリックで読み取り専用のバケットアクセス許可で設定されています。クロスアカウントアクセスを許可し続けるには、Amazon S3 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設定を介

してバケットとオブジェクトへのパブリックアクセスとクロスアカウントアクセスのブロックを有効にする前に、必ず Lightsail バケットに「すべてのオブジェクトは Lightsail のプライベートバケットアクセス許可です」で設定してください。

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 SDKs、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、サービスにリンクされたロールを使用して、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 Amazon S3 から取得し、Lightsail オブジェクトストレージリソースに適用します。Amazon S3 でパブリックアクセスのブロックを設定したら、Lightsail で有効になるまで少なくとも 1 時間待ちます。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

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」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」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 でバケット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の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バケットのアクセスログ記録を有効にする
   - 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」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について」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 で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 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」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」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」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」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: Amazon Lightsail バケットへの WordPress インスタンスの接続
    - 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 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」を参照してください。

アクセスログを使用してオブジェクトストレージバケットのリクエストを追跡する

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、プレフィックス値 を指定した場合`logs/`、Lightsail が作成する各ログオブジェクトは、キー内の`logs/`プレフィックス で始まります。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 は、ターゲットバケットにアップロードするログオブジェクトに次のオブジェクトキー形式を使用します。

```
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 は定期的にアクセスログレコードを収集し、そのレコードをログファイルに統合してから、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

Lightsail バケットログを使用してオブジェクトストレージアクセスを分析する

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
 amzn-s3-demo-bucket [06/Feb/2019:00:00:38 +0000] 192.0.2.3
 79a59df900b949e55d96a1e698fbacedfd6e09d98eacf8f8d5218e7cd47ef2be 3E57427F3EXAMPLE
 REST.GET.VERSIONING - "GET /amzn-s3-demo-bucket?versioning HTTP/1.1" 200 - 113 - 7 -
 "-" "S3Console/0.4" - s9lzHYrFp76ZVxRcpX9+5cjAnEH2ROuNkd2BHfIa6UkFVdtjf5mKR3/eTPFvsiP/
XV/VLi31234= SigV2 ECDHE-RSA-AES128-GCM-SHA256 AuthHeader amzn-s3-demo-bucket.s3.us-
west-1.amazonaws.com TLSV1.1
```

```
79a59df900b949e55d96a1e698fbacedfd6e09d98eacf8f8d5218e7cd47ef2be
 amzn-s3-demo-bucket [06/Feb/2019:00:00:38 +0000] 192.0.2.3
 79a59df900b949e55d96a1e698fbacedfd6e09d98eacf8f8d5218e7cd47ef2be 891CE47D2EXAMPLE
 REST.GET.LOGGING_STATUS - "GET /amzn-s3-demo-bucket?logging HTTP/1.1" 200 -
 242 - 11 - "-" "S3Console/0.4" - 9vKBE6vMhrNiWHZmb2L0mXOcqPGzQOI5XLnCtZNPxev+Hf
+7tpT6sxDwDty4LHBUOZJG96N1234= SigV2 ECDHE-RSA-AES128-GCM-SHA256 AuthHeader amzn-s3-
demo-bucket.s3.us-west-1.amazonaws.com TLSV1.1
```

```
79a59df900b949e55d96a1e698fbacedfd6e09d98eacf8f8d5218e7cd47ef2be
 amzn-s3-demo-bucket [06/Feb/2019:00:00:38 +0000] 192.0.2.3
 79a59df900b949e55d96a1e698fbacedfd6e09d98eacf8f8d5218e7cd47ef2be A1206F460EXAMPLE
```

```
 REST.GET.BUCKETPOLICY - "GET /amzn-s3-demo-bucket?policy HTTP/1.1" 404
 NoSuchBucketPolicy 297 - 38 - "-" "S3Console/0.4" - BNaBsXZQQDbssi6xMBdBU2sLt
+Yf5kZDmeBUP35sFoKa3sLLeMC78iwEIWxs99CRUrbS4n11234= SigV2 ECDHE-RSA-AES128-GCM-SHA256
 AuthHeader amzn-s3-demo-bucket.s3.us-west-1.amazonaws.com TLSV1.1
```

```
79a59df900b949e55d96a1e698fbacedfd6e09d98eacf8f8d5218e7cd47ef2be
 amzn-s3-demo-bucket [06/Feb/2019:00:01:00 +0000] 192.0.2.3
 79a59df900b949e55d96a1e698fbacedfd6e09d98eacf8f8d5218e7cd47ef2be 7B4A0FABBEXAMPLE
 REST.GET.VERSIONING - "GET /amzn-s3-demo-bucket?versioning HTTP/1.1" 200 -
 113 - 33 - "-" "S3Console/0.4" - Ke1bUcazaN1jWuUlPJaxF64cQVpUEhoZKEG/hmy/gijN/
I1DeWqDfFvnpybfEseEME/u7ME1234= SigV2 ECDHE-RSA-AES128-GCM-SHA256 AuthHeader amzn-s3-
demo-bucket.s3.us-west-1.amazonaws.com TLSV1.1
```

```
79a59df900b949e55d96a1e698fbacedfd6e09d98eacf8f8d5218e7cd47ef2be
 amzn-s3-demo-bucket [06/Feb/2019:00:01:57 +0000] 192.0.2.3
 79a59df900b949e55d96a1e698fbacedfd6e09d98eacf8f8d5218e7cd47ef2be
 DD6CC733AEXAMPLE REST.PUT.OBJECT s3-dg.pdf "PUT /amzn-s3-demo-bucket/
s3-dg.pdf HTTP/1.1" 200 - - 4406583 41754 28 "-" "S3Console/0.4" -
 10S62Zv81kBW7BB6SX4XJ48o6kpcl6LPwEoizZQQxJd5qDSCTLX0TgS37kYUBKQW3+bPdrg1234= SigV4
 ECDHE-RSA-AES128-SHA AuthHeader amzn-s3-demo-bucket.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 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) 。アクセスポイントの形式ARNが正しくないか、使用されていない場合、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。の詳細についてはARNs、「 AWS全般のリファレンス」の「Amazon リソースネーム (ARN) 」の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AWS、アカウント ID の別の形式です。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「 アカウント識別子」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、「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AWS。

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

```
amzn-s3-demo-bucket
```

Time (時間)

リクエストが受信された時刻。これらの日付と時刻は協定世界時 () ですUTC。を使用した形式 *strftime()* の用語は次のとおりです。*[%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 によって生成される文字列。

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

オペレーションがキーパラメータを取らない場合、エンURLコードされたリクエストの「key」部分、または「-」部分。

エントリ例

```
/photos/2019/08/puppy.jpg
```

リクエスト -URI

リクエスト -URI HTTPリクエストメッセージの一部。

エントリ例

```
"GET /amzn-s3-demo-bucket/photos/2019/08/puppy.jpg?x-foo=bar HTTP/1.1"
```

HTTP ステータス

レスポンスの数値HTTPステータスコード。

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

HTTP プロトコルオーバーヘッドを除く、送信されたレスポンスバイト数。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 がリクエストの処理に費やしたミリ秒数。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

存在する場合、Referer HTTP ヘッダーの値。HTTP user-agents (ブラウザなど) は通常、リクエストを行うときにURL、このヘッダーをリンクまたは埋め込みページの 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 拡張リクエスト 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 -の HTTP。

エントリ例

```
ECDHE-RSA-AES128-GCM-SHA256
```

認証タイプ

認証ヘッダーAuthHeader、QueryStringクエリ文字列 (署名付き URL) 、または認証されていないリクエスト-に使用されるリクエスト認証のタイプ。

エントリ例

```
AuthHeader
```

ホストヘッダー

Lightsail への接続に使用されるエンドポイント。

エントリ例

```
s3.us-west-2.amazonaws.com
```

TLS 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または が使用され-ていない場合TLSは のいずれかです。

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AWS、アカウント ID の別の形式です。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「 アカウント識別子」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、「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AWS。

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

```
amzn-s3-demo-bucket
```

Time (時間)

リクエストが受信された時刻。これらの日付と時刻は協定世界時 () ですUTC。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 によって生成される文字列。

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
```

リクエスト -URI

リクエスト -URI HTTPリクエストメッセージの一部。

エントリ例

```
"GET /amzn-s3-demo-bucket/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 部分の数値HTTPステータスコード。

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

HTTP プロトコルオーバーヘッドを除く、送信されたレスポンスバイト数。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 がリクエストの処理に費やしたミリ秒数。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

存在する場合、Referer HTTP ヘッダーの値。HTTP user-agents (ブラウザなど) は通常、リクエストを行うときにURL、このヘッダーをリンクまたは埋め込みページの 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 拡張リクエスト 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 -の HTTP。

エントリ例

```
ECDHE-RSA-AES128-GCM-SHA256
```

認証タイプ

認証ヘッダーAuthHeader、QueryStringクエリ文字列 (署名付き URL) 、または認証されていないリクエスト-に使用されるリクエスト認証のタイプ。

エントリ例

```
AuthHeader
```

ホストヘッダー

Lightsail への接続に使用されるエンドポイント。

エントリ例

```
s3.us-west-2.amazonaws.com
```

TLS 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または が使用され-ていない場合TLSは のいずれかです。

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、URLリクエストの にカスタムクエリ文字列パラメータを追加します。Lightsail は、「x-」で始まるクエリ文字列パラメータを無視しますが、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それらのパラメータを含めます。

例えば、GET の "s3.amazonaws.com/amzn-s3-demo-bucket/photos/2019/08/puppy.jpg?x-user=johndoe" リクエストは、"s3.amazonaws.com/amzn-s3-demo-

bucket/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使用できます。

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。

Lightsail でバケットアクセスログ記録を有効にする

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 はバケットのアクセスログを収集しません。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したターゲットバケットに配信します。レプリケート元バケットとレプリケート先バケットの両方が同じ にあり AWS リージョン 、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 、 AWS Command Line Interface （ AWS CLI ）API、または を使用してバケットのアクセスログ記録を有効または無効にする方法を示しますAWSSDKs。

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。

を使用してアクセスログ記録を有効にする AWS CLI

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 AWS リージョン がある各 に専用のログ記録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、 をインストール AWS CLI し、Lightsail 用に設定する必要があります。詳細については、「Lightsail で動作する AWS CLI ように 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。

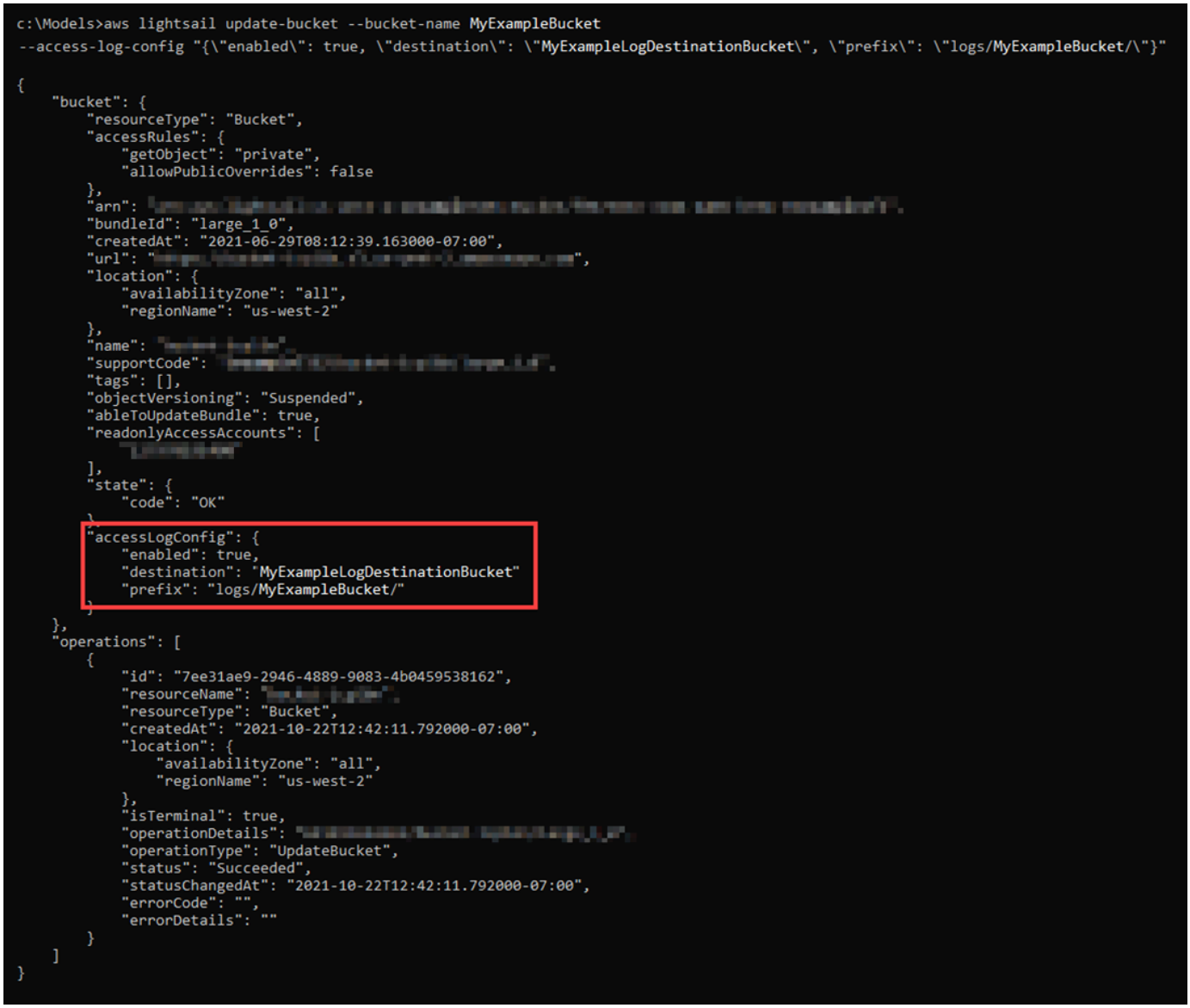
- *SourceBucketName* - アクセスログが作成されるソースバケットの名前。
- *TargetBucketName* – アクセスログを保存するターゲットバケットの名前。
- *ObjectKeyNamePrefix/* - アクセスログのオプションのオブジェクトキー名のプレフィックス。このプレフィックスは、スラッシュ (/) で終わる必要があります。

例

```
aws lightsail update-bucket --bucket-name amzn-s3-demo-bucket1 --access-log-config
 "{\"enabled\": true, \"destination\": \"amzn-s3-demo-bucket2\", \"prefix\":
 \"logs/amzn-s3-demo-bucket1/\"}"
```

この例では、*amzn-s3-demo-bucket1* は、アクセスログが作成されるソースバケットです。*amzn-s3-demo-bucket2* は、アクセスログが保存される送信先バケットです。*logs/amzn-s3-demo-bucket1/* 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。



を使用したアクセスログ記録の無効化 AWS CLI

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、 をインストール AWS CLI し、Lightsail 用に設定する必要があります。詳細については、「Lightsail で動作する AWS CLI ように 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。

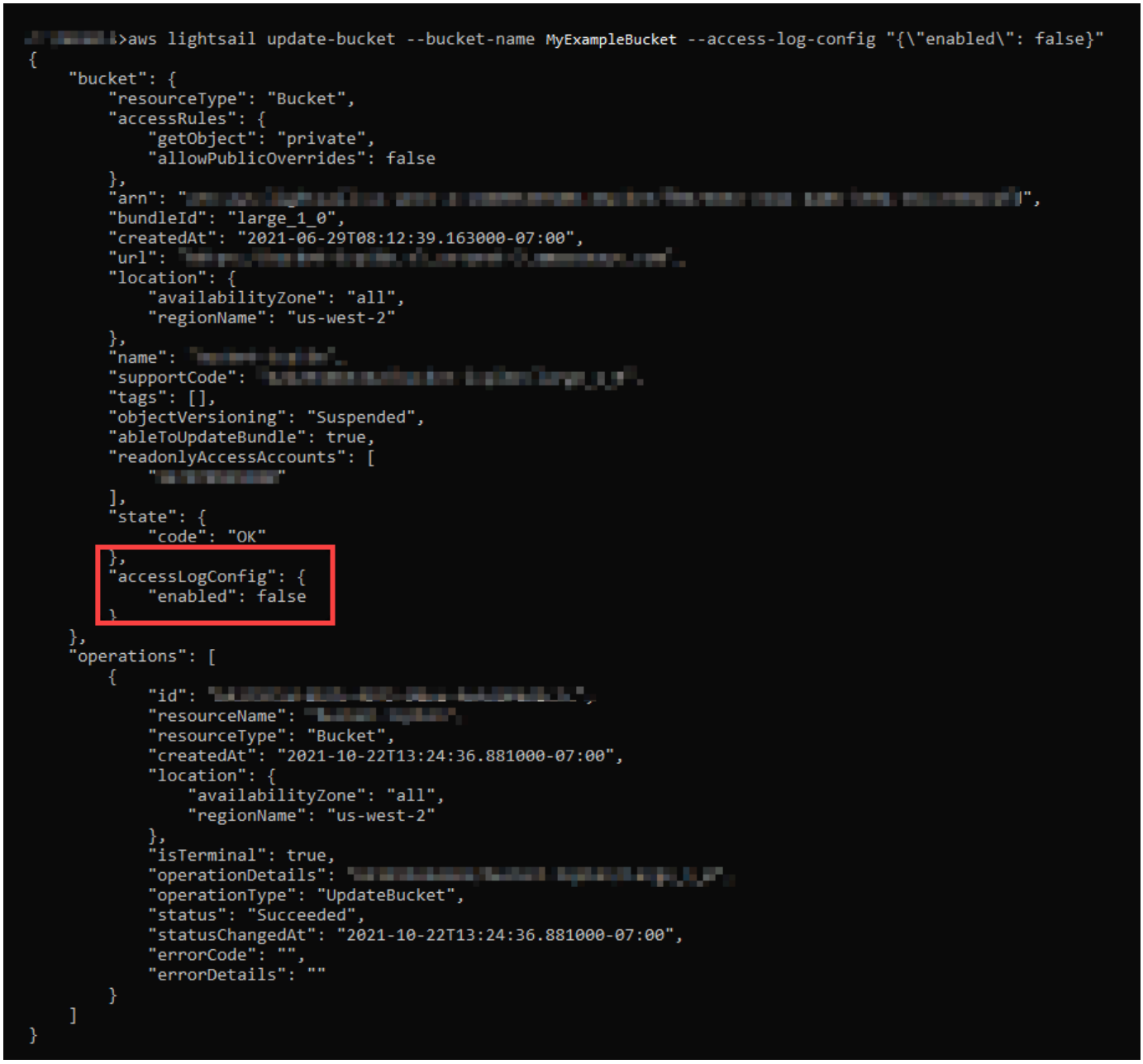
```
aws lightsail update-bucket --bucket-name SourceBucketName --access-log-config
 "{\"enabled\": false}"
```

コマンドで、*SourceBucketName* アクセスログ記録を無効にするソースバケットの名前。

例

```
aws lightsail update-bucket --bucket-name amzn-s3-demo-bucket --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。



Lightsail で Amazon Athena を使用してバケットアクセスログを分析する

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 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、次のようにURI、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 S3 としてフォーマットする必要があります。 `s3://amzn-s3-demo-bucket1-logs/prefix/`

1. https://console.aws.amazon.com/athena/ から 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**
>
> S3 バケット AWS リージョン と同じ にデータベースを作成するのがベストプラクティスです。

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.amzn-s3-demo-bucket_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://amzn-s3-demo-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 — オブジェクトを削除したユーザーと、いつ (タイムスタンプ、IP アドレス、IAMユーザー) を表示する

```
SELECT RequestDateTime, RemoteIP, Requester, Key
FROM s3_access_logs_db.amzn-s3-demo-bucket_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.amzn-s3-demo-bucket_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.amzn-s3-demo-bucket_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.amzn-s3-demo-bucket_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、、、 などのオペレーションのオブジェクトアクセスリクエストを識別しGETPUTDELETE、それらのリクエストに関する詳細情報を確認できます。

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 のすべての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.amzn-s3-demo-bucket_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。

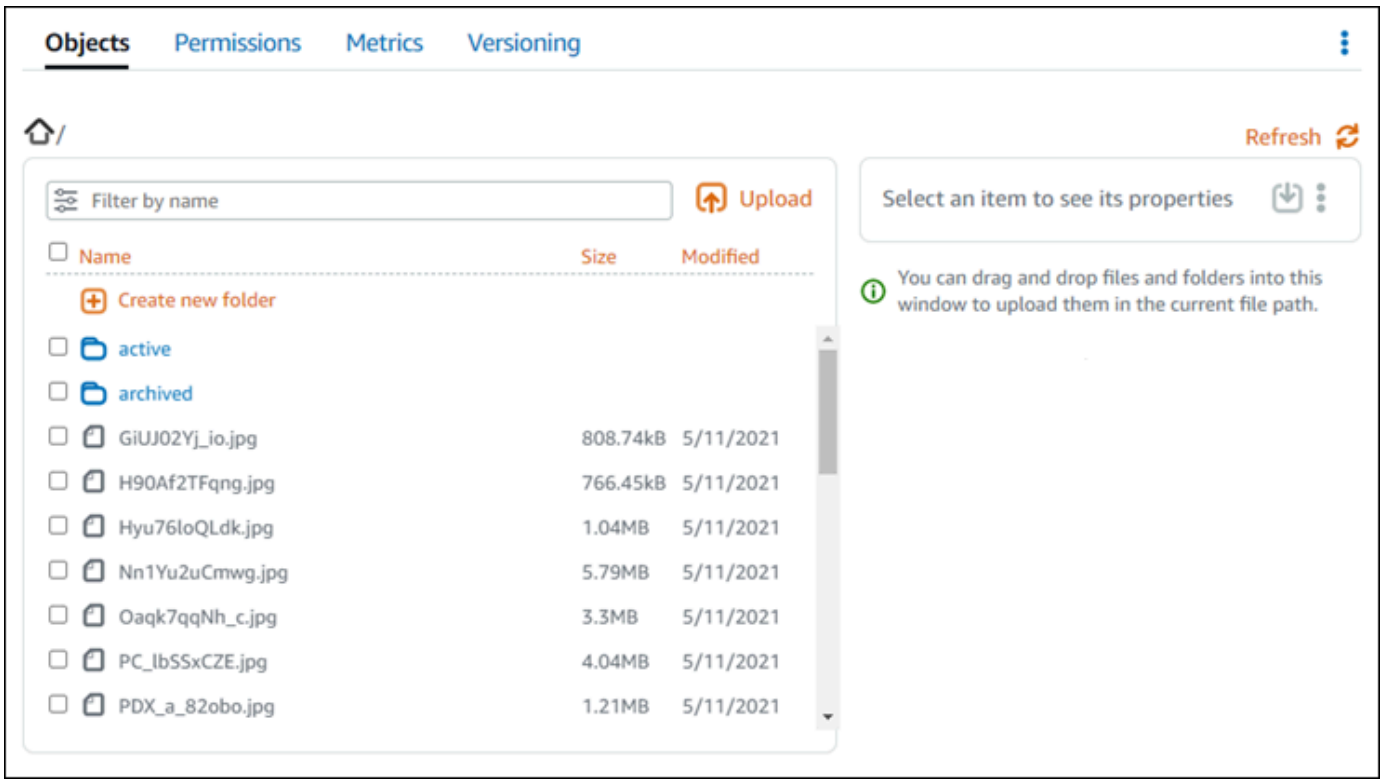
Lightsail バケット内のファイルとフォルダを管理する

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) と を使用してAWSSDKs、バケット内のオブジェクトキーを一覧表示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。

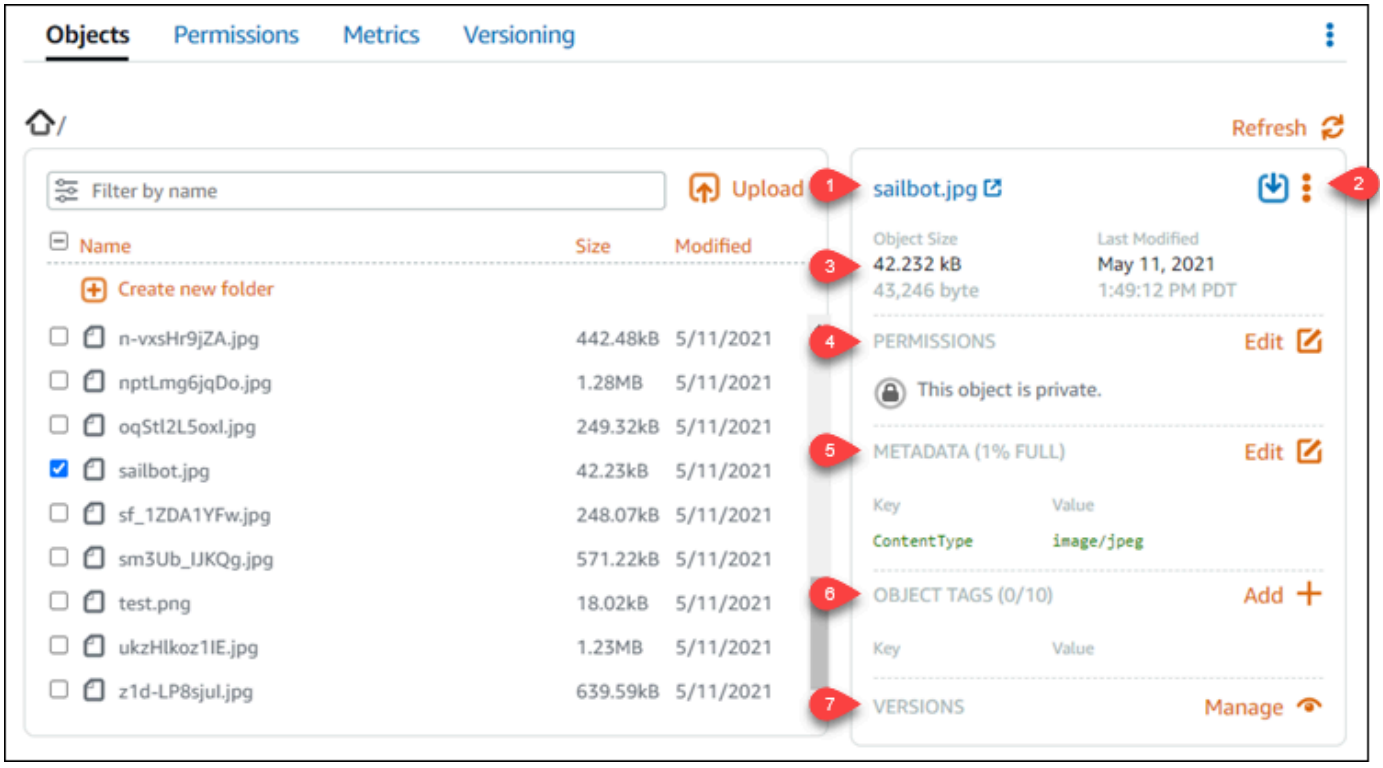
Lightsail コンソールを使用してオブジェクトをフィルタリングする

Lightsail コンソールを使用してバケットに保存されているオブジェクトを表示するには、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、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」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、現時点では Lightsail オブジェクトストレージサービスでサポートされてい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。

を使用してオブジェクトを表示する AWS CLI

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-v「2」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、「Amazon Lightsail と連携 AWS Command Line Interface するように を設定する Amazon Lightsail」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* すべてのオブジェクトを一覧表示するバケットの名前。
   - 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一覧表示するバケットの名前。

- *ObjectKeyNamePrefix* - 指定されたプレフィックスで始まるキーにレスポンスを制限するオブジェクトキー名のプレフィックス。

> **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 amzn-s3-demo-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 amzn-s3-demo-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

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」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、アクセスキーの作成、バケットへのインスタンスのアタッチ、他のAWSアカウントへのアクセスの付与によって、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」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 でバケット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の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バケットのアクセスログ記録を有効にする
   - 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、IAMAmazon Lightsail」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について」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」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」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」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」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: Amazon Lightsail バケットへの WordPress インスタンスの接続
- 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 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」を参照してください。

トピック

- Lightsail バケット間でオブジェクトをコピーして移動する
- オブジェクトを削除して Lightsail バケットストレージをクリアする
- Lightsail バケットからオブジェクトをダウンロードする
- Lightsail バケット内のオブジェクトを名前プレフィックスでフィルタリングする
- Lightsail でオブジェクトのバージョニングを有効化および停止する
- Lightsail バケットで以前のオブジェクトバージョンを復元する
- Lightsail バケット内のオブジェクトにタグを付ける

Lightsail バケット間でオブジェクトをコピーして移動する

Amazon Lightsail オブジェクトストレージサービスのバケットにすで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 コンソールと AWS Command Line Interface () を使

用してオブジェクトをコピーする方法について説明しますAWS CLI。バケット内のオブジェクトをコピーして、オブジェクトの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 の場所間でオブジェクトを移動したりします (たとえば、Lightsail が利用可能なオブジェクト AWS リージョン を 1 つの場所から別の場所に移動するなど)。オブジェクトは、、 AWS APIs AWS SDKs、および AWS Command Line Interface () を使用してのみ、ロケーション間でコピーできます AWS CLI。

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 コンソールを使用して、最大 2 GB のサイズのオブジェクトのコピーを作成できます。AWS Command Line Interface （ AWS CLI ）、 AWS APIs、および を使用して、1 つのコピーオブジェクトアクションで最大 5 GB のサイズのオブジェクトのコピーを作成できます AWS SDKs。サイズが 5 GB を超えるオブジェクトをコピーするには、 AWS CLI、、 AWS APIsおよび 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 AWS SDKs。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てバケットに保存されているオブジェクトをコピーするには、次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 ホームページで、ストレージタブを選択します。
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口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口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。

を使用してオブジェクトをコピーする AWS CLI

AWS Command Line Interface () を使用してバケット内のオブジェクトをコピーするには、次の手順を実行しますAWS CLI。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、 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 で動作する AWS CLI 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. 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、およびコピーするオブジェクトの完全なオブジェクトキー。たとえば、amzn-s3-demo-bucketバケットからオブジェクトimages/sailbot.jpgをコピーするには、amzn-s3-demo-bucket/images/sailbot.jpgを指定します。
- *DestinationObjectKey* - 新しいオブジェクトコピーの完全なオブジェクトキー。
- *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 amzn-s3-demo-bucket1/images/sailbot.jpg
 --key media/sailbot.jpg --bucket amzn-s3-demo-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 amzn-s3-demo-bucket1/images/sailbot.jpg --
key images/sailbot.jpg --bucket amzn-s3-demo-bucket2 --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

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」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、アクセスキーの作成、バケットへのインスタンスのアタッチ、他のAWSアカウントへのアクセスの付与によって、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」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 でバケット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の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バケットのアクセスログ記録を有効にする
   - 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、IAMAmazon Lightsail」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について」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」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」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」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」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: Amazon Lightsail バケットへの WordPress インスタンスの接続
- 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 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」を参照してください。

オブジェクトを削除して Lightsail バケットストレージをクリアする

Amazon Lightsail オブジェクトストレージサービスのバケットからオブジェクトを削除できます。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

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、または を使用して AWS APIs、オブジェクトの任意のバージョンを削除できます AWS SDKS。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とオブジェクトバージョンを削除する

Lightsail コンソールの Objects タブの Objects 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、オブジェクトの以前のバージョン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、このガイドの後半の「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 を使用してオブジェクトとオブジェクトバージョンを削除する AWS SDKs

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する

Lightsail コンソールを使用して、保存された以前のバージョンを含むオブジェクトを削除するには、以下の手順を実行します。Lightsail コンソールを使用して一度に削除できるオブジェクトは 1 つだけです。を使用して AWS CLI 、複数のオブジェクトを一度に削除します。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 ホームページで、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 ホームページで、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。

を使用して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 CLI

AWS Command Line Interface () を使用してバケット内の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、次の手順を実行しますAWS CLI。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、 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、「Amazon Lightsail で動作する AWS Command Line Interface ように を設定する Amazon Lightsail」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* - オブジェクトを削除するバケットの名前。
- *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 amzn-s3-demo-bucket --key images/sailbot.jpg
```

オブジェクトバージョンの削除：

```
aws s3api delete-object --bucket amzn-s3-demo-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、 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、「Amazon Lightsail で動作する AWS Command Line Interface ように を設定する Amazon Lightsail」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* は、削除するオブジェクトのオブジェクトキーを使用します。

  ```
  {
    "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。置換 *ObjectKey* また、*VersionID* オブジェクトキーと削除するIDsオブジェクトバージョンの 。

  > **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。

  ```
  {
    "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 amzn-s3-demo-bucket --delete file://home/user/
Documents/delete-objects.json
```

- Windows コンピュータの場合:

```
aws s3api delete-objects --bucket amzn-s3-demo-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

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」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、アクセスキーの作成、バケットへのインスタンスのアタッチ、他のAWSアカウントへのアクセスの付与によって、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」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 でバケット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の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バケットのアクセスログ記録を有効にする
   - Amazon Lightsail の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、IAMAmazon Lightsail」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について」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 で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」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」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」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」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: Amazon Lightsail バケットへの WordPress インスタンスの接続
   - 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 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」を参照してください。

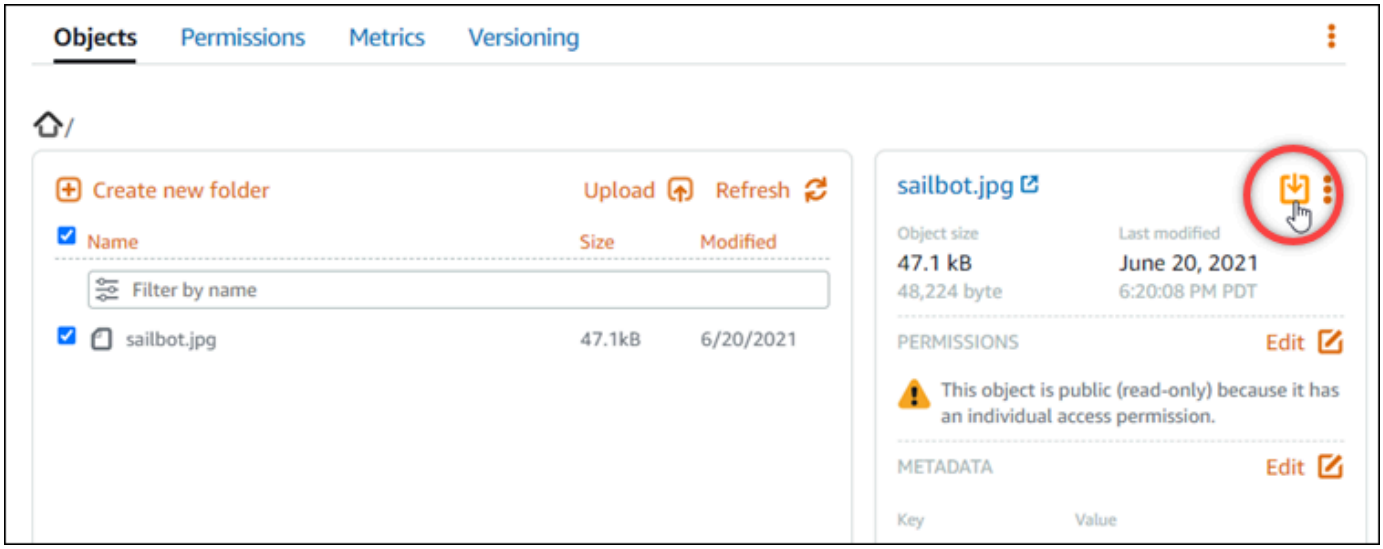
Lightsail バケットからオブジェクトをダウンロードする

Amazon Lightsail オブジェクトストレージサービスでは、アクセスできるバケット、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できます。Lightsail コンソールを使用して、一度に 1 つのオブジェクトをダウンロードできます。1 つ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 SDKs、または REST を使用しますAPI。このガイドでは、Lightsail コンソールと を使用してオブジェクトをダウンロードする方法について説明します AWS CLI。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 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。

を使用してオブジェクトをダウンロードする AWS CLI

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、 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、「Amazon Lightsail で動作する AWS Command Line Interface ように を設定する Amazon Lightsail」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* - オブジェクトをダウンロードするバケットの名前。
- *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 amzn-s3-demo-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

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」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、アクセスキーの作成、バケットへのインスタンスのアタッチ、他のAWSアカウントへのアクセスの付与によって、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」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 でバケット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の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バケットのアクセスログ記録を有効にする
   - 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、IAMAmazon Lightsail」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について」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」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」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」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」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: Amazon Lightsail バケットへの WordPress インスタンスの接続
- 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 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」を参照してください。

Lightsail バケット内のオブジェクトを名前プレフィックスでフィルタリングする

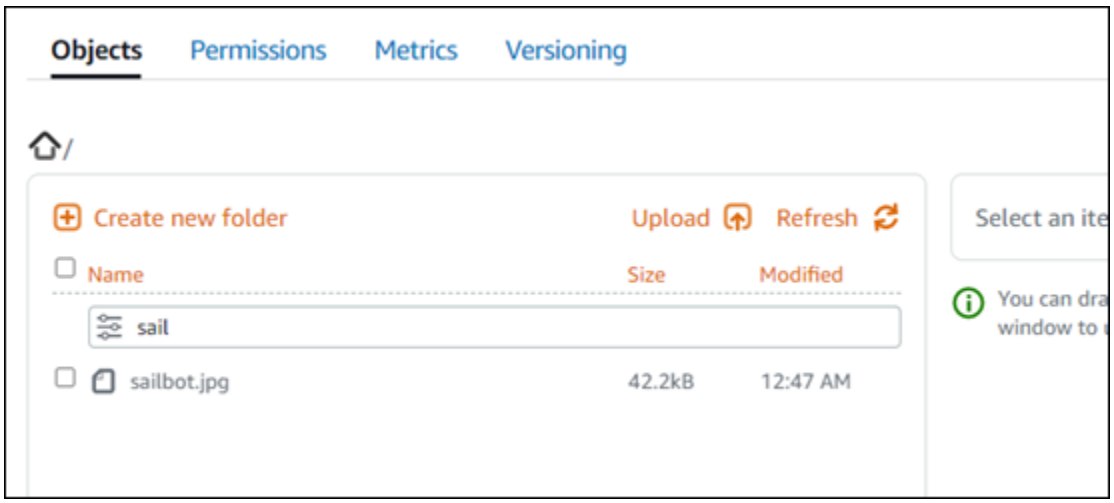
フィルタリングを使用して、Amazon Lightsail オブジェクトストレージサービスのバケット内のオブジェクトを検索できます。このガイドでは、Lightsail コンソールと AWS Command Line Interface () を使用してオブジェクトをフィルタリングする方法を示しますAWS CLI。バケットについての詳細は、「オブジェクトストレージ」を参照してください。

Lightsail コンソールを使用してオブジェクトをフィルタリングする

Lightsail コンソールを使用してバケット内のオブジェクトをフィルタリングするには、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、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。

を使用してオブジェクトをフィルタリングする AWS CLI

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、「 コマンドリファレンス」のlist-objects-v「2」を参照してください。 AWS CLI

> **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、「Amazon Lightsail で動作する AWS Command Line Interface ように を設定する Amazon Lightsail」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一覧表示するバケットの名前。

- *ObjectKeyNamePrefix* - 指定されたプレフィックスで始まるキーにレスポンスを制限するオブジェクトキー名のプレフィックス。

> **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 amzn-s3-demo-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

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」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、アクセスキーの作成、バケットへのインスタンスのアタッチ、他のAWSアカウントへのアクセスの付与によって、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」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 でバケット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の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バケットのアクセスログ記録を有効にする
   - Amazon Lightsail の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、IAMAmazon Lightsail」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について」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 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」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」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」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」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」を参照してください。

Lightsail でオブジェクトのバージョニングを有効化および停止する

Amazon Lightsail オブジェクトストレージサービスのバージョニングは、オブジェクトの複数のバリアントを同じバケットに保持する手段です。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を有効にすると、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、それらのすべてのオブジェクトが保存されます。Lightsail オブジェクトストレージサービスのバケットでは、バージョニングが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

バケットのバージョニングは、Lightsail コンソールの 3 つの状態のいずれかになります。

- 無効 (`NeverEnabled` APIおよび の SDKs)
- 有効 (`Enabled` APIおよび の SDKs)
- 停止 (`Suspended`APIおよび の SDKs)

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 IDs

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、保存されているオブジェクトの一意のバージョン ID を自動的に生成します。例えば、1 つのバケットに、

(バージョン 111111) や photo.gif (バージョン 121212) などIDs、同じキーでバージョン が異なる 2 photo.gif つのオブジェクトを含めることができます。

バージョンは編集IDsできません。これらは Unicode、UTF-8 でエンコードされ、URLすぐに使用できる不透明な文字列で、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化または停止するには、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. 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。

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停止する AWS CLI

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、 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 で動作する AWS CLI 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、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するバケットの名前。
- *VersioningState* - 次のいずれかです。
  - 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 amzn-s3-demo-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

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」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、アクセスキーの作成、バケットへのインスタンスのアタッチ、他のAWSアカウントへのアクセスの付与によって、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」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 でバケット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の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バケットのアクセスログ記録を有効にする
   - Amazon Lightsail の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、IAMAmazon Lightsail」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について」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 で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」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」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」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」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: Amazon Lightsail バケットへの WordPress インスタンスの接続
    - 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 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」を参照してください。

Lightsail バケットで以前のオブジェクト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ン対応の場合は、オブジェクトの以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 コンソールを使用して、オブジェクトの以前のバージョンを復元できます。 AWS Command Line Interface （ AWS CLI) を使用して、 AWS SDKsオブジェクトの以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 コンソールを使用してオブジェクトの以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以前のバージョンを復元するには、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、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) を選択します。
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。

を使用してオブジェクトの以前のバージョンを復元する AWS CLI

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、 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、「Amazon Lightsail で動作する AWS Command Line Interface ように を設定する Amazon Lightsail」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* - オブジェクトの以前のバージョンを復元するバケットの名前。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* - 現在のバージョンに復元する以前のオブジェクトバージョンの ID。list-object-versions コマンドを使用して、バケット内のIDsオブジェクトのバージョンのリストを取得します。

例:

```
aws s3api copy-object --copy-source "amzn-s3-demo-bucket/sailbot.jpg?
versionId=GQWEexample87Mdl8Q_DKdVTiVMi_VyU" -key sailbot.jpg --bucket amzn-s3-demo-
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

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」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、アクセスキーの作成、バケットへのインスタンスのアタッチ、他のAWSアカウントへのアクセスの付与によって、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」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 でバケット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の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バケットのアクセスログ記録を有効にする
   - Amazon Lightsail の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、IAMAmazon Lightsail」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について」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 で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 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」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」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」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」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: Amazon Lightsail バケットへの WordPress インスタンスの接続
    - 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 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」を参照してください。

Lightsail バケット内のオブジェクトにタグを付ける

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する

Lightsail コンソールを使用してバケット内のオブジェクトにタグを追加または削除するには、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、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 () 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、以下の手順を実行しますAWS CLI。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、 コマンドリファレンスdelete-object-taggingのput-object-tagging「」および「」を参照してください。 AWS CLI

> **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 で動作する AWS CLI 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* - 最初のタグのキー値。
- *ValueTag1* - 最初のタグの値。
- *KeyTag2* - 2 番目のタグのキー値。
- *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。

例:

```
aws s3api delete-object --bucket amzn-s3-demo-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

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」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、アクセスキーの作成、バケットへのインスタンスのアタッチ、他のAWSアカウントへのアクセスの付与によって、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」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 でバケット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の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バケットのアクセスログ記録を有効にする
   - Amazon Lightsail の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、IAMAmazon Lightsail」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について」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 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」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」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」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: Amazon Lightsail バケットへの WordPress インスタンスの接続
- 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 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」を参照してください。

インスタンスの Lightsail バケットへのアクセスを制御する

Amazon Lightsail インスタンスを Lightsail バケットにアタッチして、バケットとそ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などです。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」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 ホームページで、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、バケット AWS リージョン と同じ にあるインスタンスのみをアタッチできます。
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などです。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」を参照してください。

Lightsail バケットストレージプランの使用変動を調整する

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で月額コスト、ストレージスペースクォータ、データ転送クォータを指定します。バケットのストレージプランは、毎月の AWS 請求サイクル内で 1 回だけ更新できます。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、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、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。

を使用してバケットのストレージプランを変更する AWS CLI

AWS Command Line Interface () を使用してバケットのプランを変更するには、次の手順を実行しますAWS CLI。これは、`update-bucket-bundle` コマンドを使用して実行できます。バケットストレージプランは、 ではバケットバンドルと呼ばれることに注意してくださいAPI。詳細については、コマンドリファレンスupdate-bucket-bundleの「」を参照してください。 AWS CLI

> **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 で動作する AWS CLI 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. 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* - ストレージプランを更新するバケットの名前。
- *BundleID* - バケットに適用する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 のリストを表示しますIDs。詳細については、 コマンドリファレンスget-bucket-bundlesの「」を参照してください。 AWS CLI

例:

```
aws lightsail update-bucket-bundle --bucket-name amzn-s3-demo-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": ""
        }
    ]
}
```

セキュリティを強化するために Lightsail バケットのアクセス許可を管理する

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。

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 ホームページで、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、バケットで個々のオブジェクトを公開することを許可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: 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バケットを使用する」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。

AWS アカウント間で Lightsail バケットへの読み取り専用アクセスを許可する

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、オブジェクトを別の AWS アカウントと共有する場合に最適です。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`)。バケットには、最大 10 個の AWS アカウントへのアクセスを許可できます。

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 ホームページで、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の追加 を選択して、別の AWS アカウントのバケットへのアクセスを許可します。
6. アクセスを許可する 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、ページのクロスアカウントアクセスセクションに一覧表示されます。 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。

Amazon Lightsail の個々のバケットオブジェクトへのパブリックアクセスを許可する

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、バケットで個々のオブジェクトを公開できるように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: 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バケットを使用する」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 ホームページで、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。

マルチパートアップロードで Lightsail 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部分がアップロードされると、Amazon S3 はこれらの部分をアセンブルし、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

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 ステップのプロセスです。

1. CreateMultipartUpload 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ます。
2. UploadPart アクションを使用してファイルパーツをアップロードします。
3. CompleteMultipartUpload アクションを使用してマルチパートアップロードを完了します。

> **Note**
>
> マルチパートアップロードは、 AbortMultipartUploadアクションを使用して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 Simple Storage Service 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。

完全なマルチパートアップロードリクエストには、アップロード ID と、パート番号と対応するETag値の両方のリストが含まれている必要があります。Amazon Simple Storage Service レスポンスには、結合されたオブジェクトデータを一意に識別ETagする が含まれます。ETag これは必ずしもオブジェクトデータのMD5ハッシュではありません。

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。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、Amazon Simple Storage Service が返す対応するETag値のリストを保持します。

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、 コマンドリファレンスcreate-multipart-uploadの「」を参照してください。AWS CLI

> **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 で動作する AWS CLI 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、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 amzn-s3-demo-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。

を使用してパートをアップロードする AWS CLI

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、 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 で動作する AWS CLI 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、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。

例:

```
aws s3api upload-part --bucket amzn-s3-demo-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\""
}
```

を使用してマルチパートアップロードの一部を一覧表示する AWS CLI

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、 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 で動作する AWS CLI 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. 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一覧表示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。

   例:

```
aws s3api list-parts --bucket amzn-s3-demo-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。

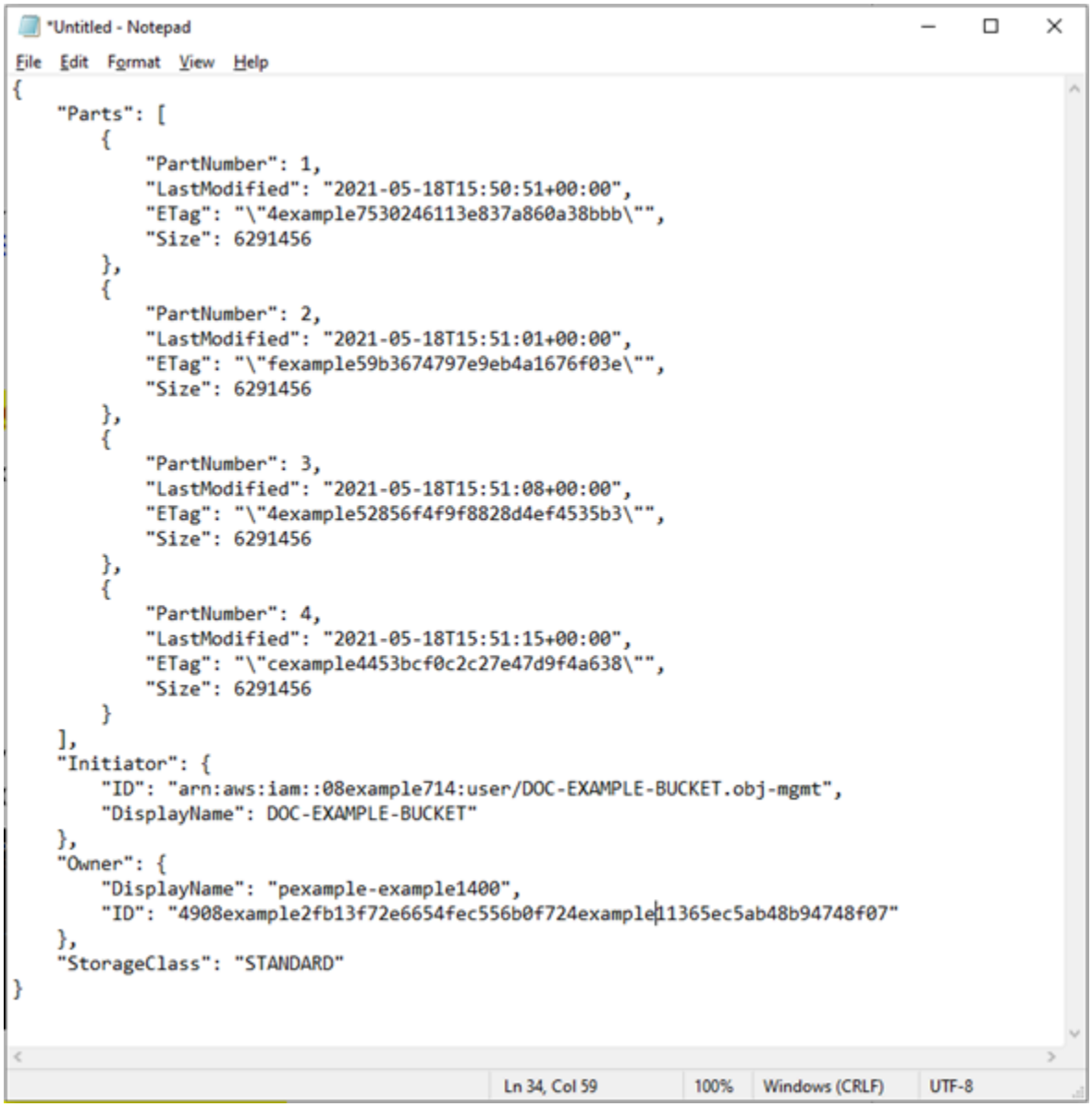
```
C:\>aws s3api list-parts --bucket DOC-EXAMPLE-BUCKET --key sailbot.mp4 --upload-id "R4QU.mO.exampleiHWiLOeNw7JtXX7OotR
hTLsXXCzF21CZdYlfj5lfjtiMnpzVw2WPj.exampleBTmL_N_.42.DlHYOTsITFsX.tO3XOUTTAHiCxY5VR8jWRGdkVkUG"
{
    "Parts": [
        {
            "PartNumber": 1,
            "LastModified": "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

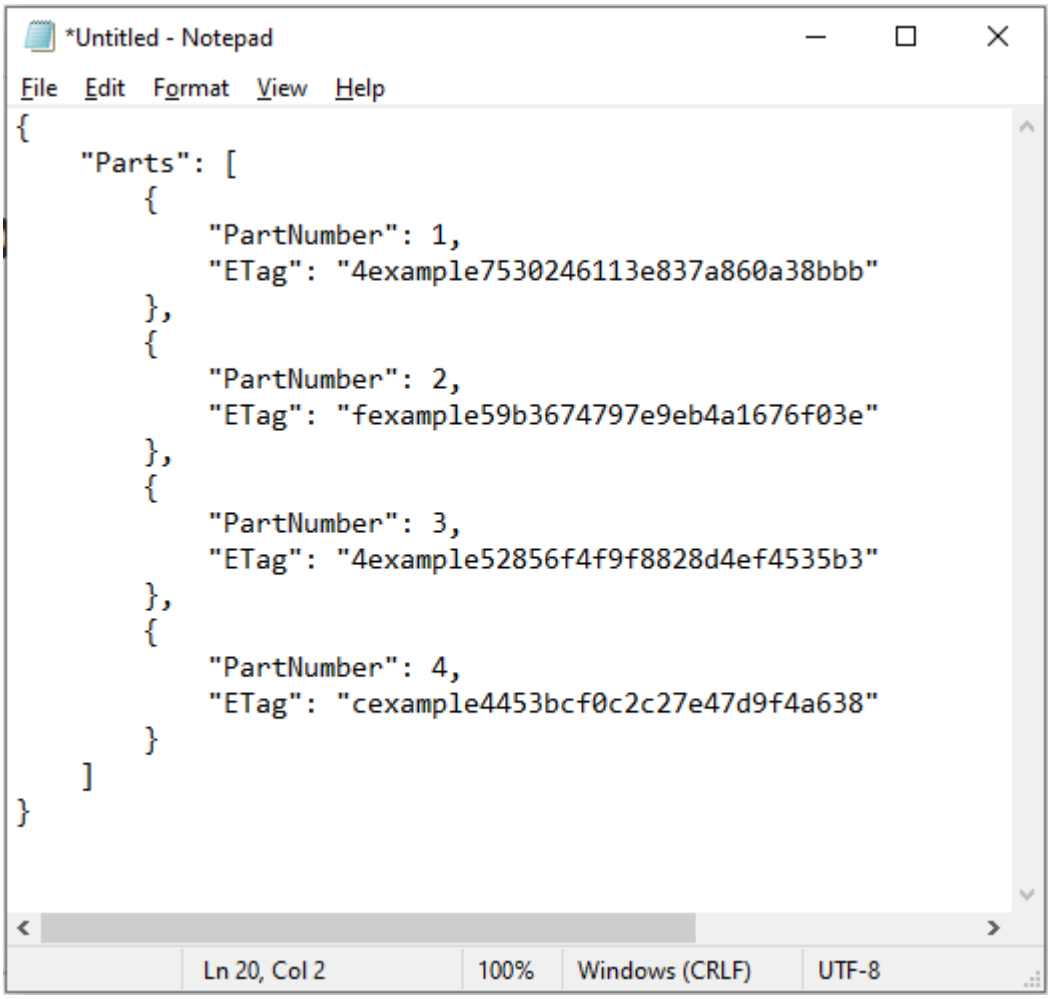
以下の手順で、アップロードしたすべてのパーツとそのETag 値を定義したマルチパートアップロード .json ファイルを作成します。これは、このガイドの後半で、マルチパートアップロードを完了するために必要です。

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 としてコンピュータに保存しmpstructure.json、このガイドの AWSセクションを使用してマルチパートアップロードCLIを完了するに進みます。

を使用してマルチパートアップロードを完了する AWS CLI

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、 コマンドリファレンスcomplete-multipart-uploadの「」を参照してください。 AWS CLI

> **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 で動作する AWS CLI 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、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 amzn-s3-demo-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\""
}
```

を使用してバケットのマルチパートアップロードを一覧表示する AWS CLI

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、 コマンドリファレンスlist-multipart-uploadsの「」を参照してください。 AWS CLI

> **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 で動作する AWS CLI 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、パートをバケットにアップロードします。

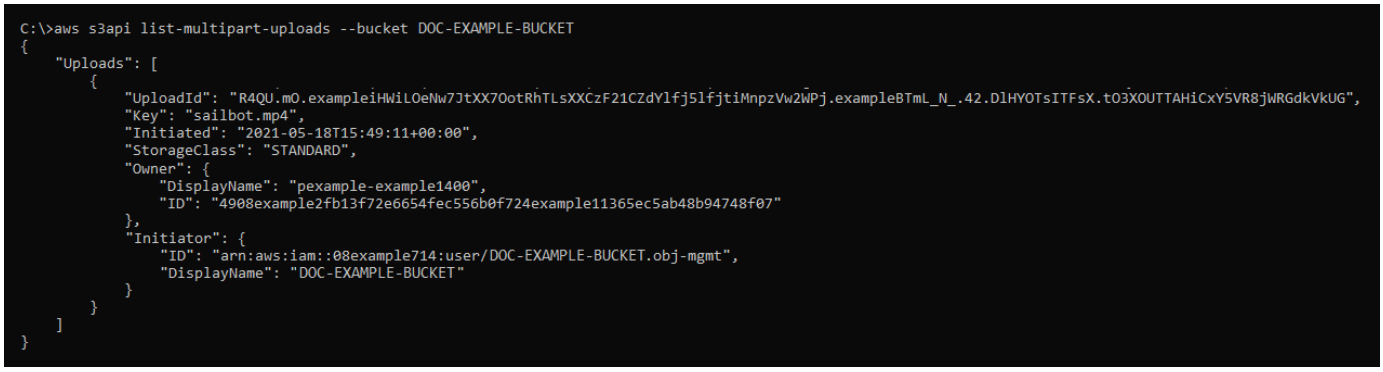
   ```
   aws s3api list-multipart-uploads --bucket BucketName
   ```

   コマンドで、*BucketName*#すべてのマルチパートアップロードを一覧表示するバケットの名前。

   例:

   ```
   aws s3api list-multipart-uploads --bucket amzn-s3-demo-bucket
   ```

   次の例のようなレスポンスが表示されます。



を使用してマルチパートアップロードを停止する AWS CLI

AWS Command Line Interface () を使用してマルチパートアップロードを停止するには、次の手順を実行しますAWS CLI。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、 コマンドリファレンスabort-multipart-uploadの「」を参照してください。 AWS CLI

> **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 で動作する AWS CLI 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、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 amzn-s3-demo-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。

Lightsail オブジェクトストレージのバケット命名要件に従う

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するときは、名前を付ける必要があります。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URL顧客が使用する の一部です。例えば、 DOC-EXAMPLE-BUCKETでバケットに名前を付ける

とus-east-1 AWS リージョン、バケットURLの は になりますDOC-EXAMPLE-BUCKET.s3.us-east-1.amazonaws.com。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 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 コンソールで表示されるバケットのフォルダ構造を構成するキー名とキー名の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データモデルは、ファイルシステムに表示される階層構造の代わりにフラット構造を使用します。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 コンソールでは、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 コンソールでは、キー名のプレフィックス (`Development/`、`Finance/`、および `Private/`) と区切り記号 (`/`) を使用してフォルダ構造を表示します。`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 コンソールで `Development/`フォルダを参照すると、 `Projects.xls` オブジェクトが表示されます。`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区切り記号の値をキー名としてゼロバイトのオブジェクトを作成することで、フォルダを作成できます。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 、AWS Command Line Interface （ AWS CLI) API、または を使用して表示および操作できます AWS SDKs。

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任意の 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、、ウェブセーフ文字DNS、パーサー、およびその他の XML へのコンプライアンスを最大化するのに役立ちますAPIs。

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 を使用する必要があります AWS SDKs。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

キー名の次の文字では、追加のコード処理が必要になる場合があり、エンURLコードまたは として参照する必要がありますHEX。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 ("$")
- ASCII 文字範囲 00 ~ 1F 16 進数 (0 ~ 31 進数) および 7F (127 進数)
- 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 ("{")
- 印刷できないASCII文字 (128 ~ 255 進数文字)
- 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 関連するオブジェクトキーの制約

XML end-of-line の処理に関する標準で指定されているように、すべてのXMLテキストが正規化され、単一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ラインフィード (ASCIIコード 10) が 1 行フィード文字に置き換えられます。XML リクエスト内の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。 XML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`

次の例は、キャリッジリターンの代替として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>
```

Secure Lightsail オブジェクトストレージバケット

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べき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 バケットのセキュリティインシデントを防ぐのに役立ちます。

最小特権アクセスの実装

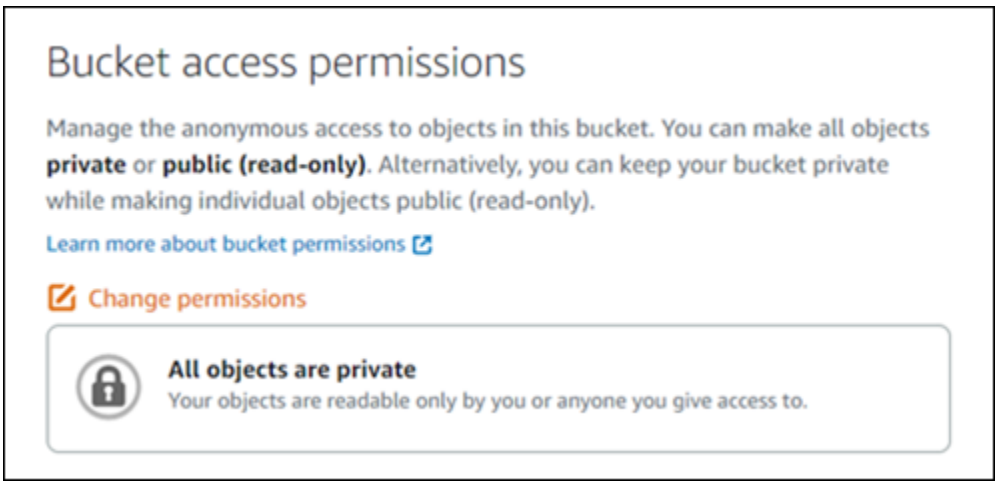
アクセス許可を付与するときは、Lightsail リソースに対するアクセス許可を取得するユーザーを決定します。つまり、該当リソースに対して許可する特定のアクションを有効にするということです。

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。最小限の特権アクセスの実装は、セキュリティリスクはもちろん、エラーや悪意ある行動によってもたらされる可能性のある影響を減らす上での基本となります。

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。 Amazon Lightsail

Lightsail バケットがパブリックアクセス可能でないことを確認する

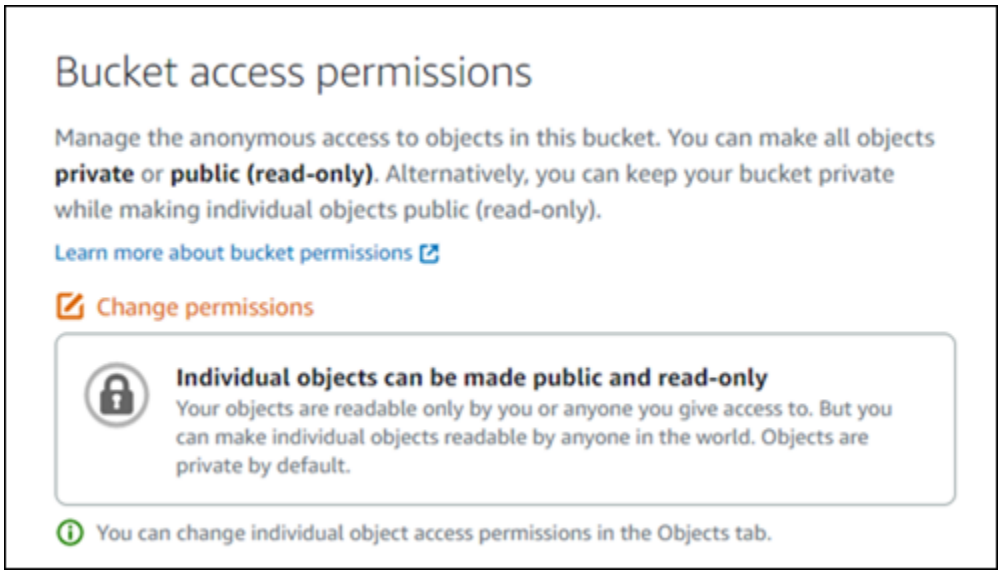
デフォルトでは、 バケットとオブジェクトはプライベートです。バケットのアクセス許可セットをすべてのオブジェクトはプライベートに設定して、バケットをプライベートに保ちます。大部分のユースケースでは、バケットや個々のオブジェクトをパブリックにする必要はありません。詳細については「バケット内の個々のオブジェクトに対するアクセス許可の設定」を参照してください。



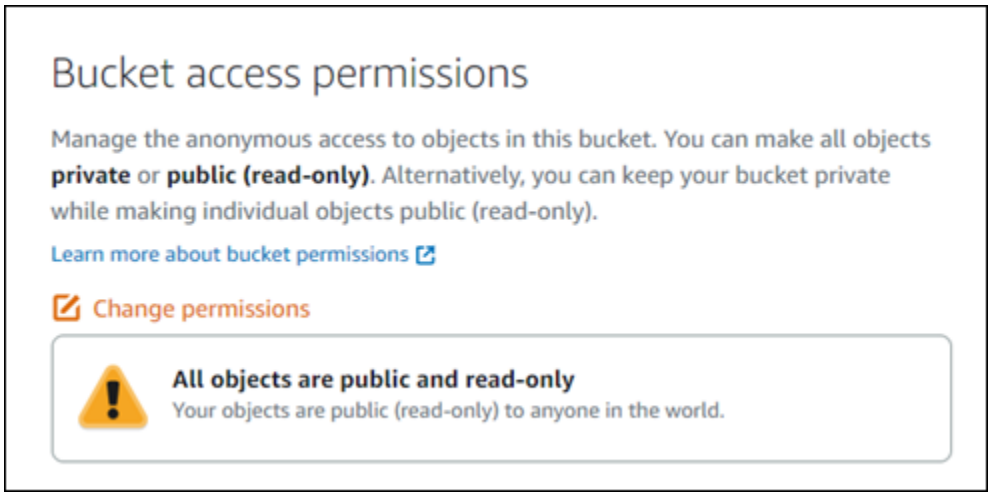
ただし、バケットを使用してウェブサイトやアプリケーションのメディアをホストしている場合は、特定のシナリオでは、バケットまたは個々のオブジェクトをパブリックにする必要があります。次のいずれかのオプションを設定して、バケットまたは個々のオブジェクトをパブリックにすることができます。

- 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

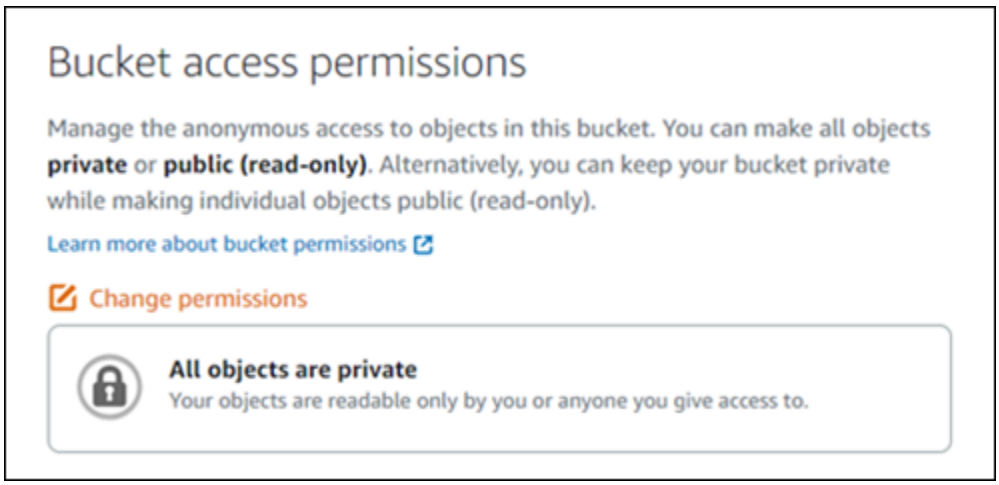
パブリックにする場合は、個々のオブジェクトのパブリックアクセシビリティを定期的に検証する必要があります。



- 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。

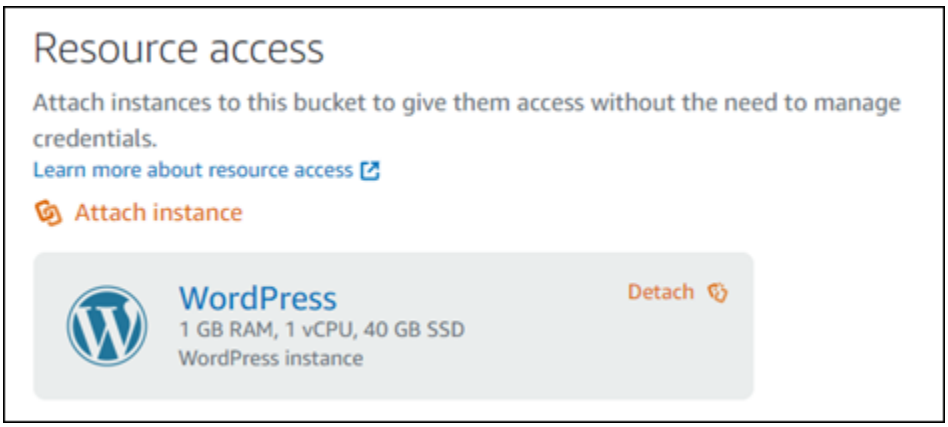


Amazon S3 でパブリックアクセスのブロックを有効にする

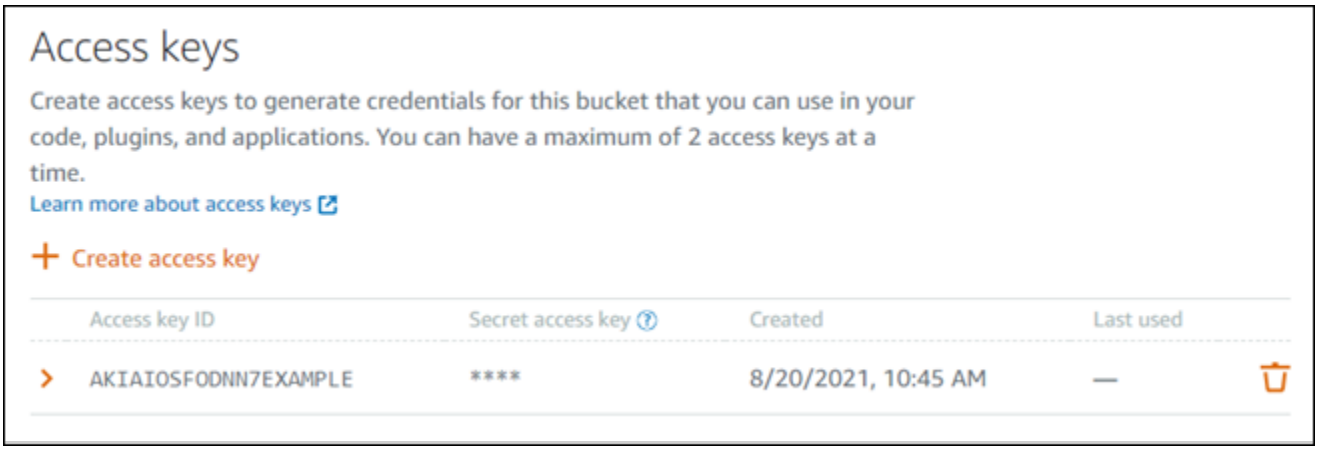
Lightsail オブジェクトストレージリソースは、パブリックアクセスを許可または拒否するときに、Lightsail バケットのアクセス許可と Amazon S3 アカウントレベルのブロックパブリックアクセス設定の両方を考慮します。Amazon S3 アカウントレベルのブロックパブリックアクセスを使用すると、アカウント管理者とバケット所有者はAmazon S3 および Lightsail バケットへのパブリックアクセスを一元的に制限できます。ブロックパブリックアクセスは、リソースの作成方法や、設定された個々のバケットとオブジェクトのアクセス許可に関係なく、すべての Amazon S3 バケットと Lightsail 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アタッチすることは、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。



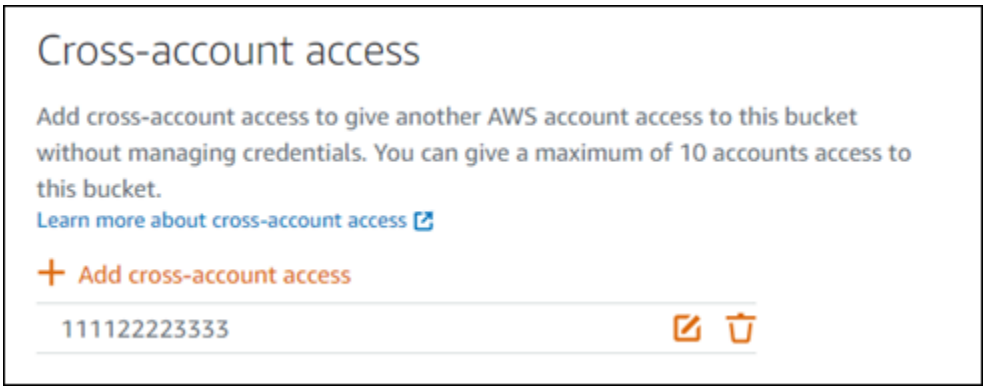
アクセスキーを作成して使用し、アプリケーションまたはプラグインにバケット内のオブジェクトへの完全なプログラムによるアクセスを付与できます。バケットでアクセスキーを使用する場合は、キーを定期的にローテーションし、既存のキーのインベントリを作成する必要があります。アクセスキーが最後に使用された日付と、そのキーが使用された AWS リージョン が、キーの使用方法に関する想定と一致していることを確認します。アクセスキーが最後に使用された日付が Lightsail コンソールに表示されます。バケットの管理ページのアクセス許可タブのアクセスキーセクションに表示されます。 使用されていないアクセスキーを削除します。

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、他の AWS アカウントにバケット内のオブジェクトへのアクセスを許可する

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公開することなく、 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 マネージドキーによるサーバー側の暗号化と、HTTPS (TLS) の適用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、転送中のデータの暗号化は、潜在的な攻撃者が 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 person-in-the-middle または操作するのを防ぐのに役立ちます。

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ている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 バケットのセキュリティ上の潜在的な弱点や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 はバケットのアクセスログを収集しません。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、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 バケットや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な部分です。Lightsail では、バケットサイズ (`BucketSizeBytes`) および `Number of objects` (NumberOfObjects) バケットメトリクスの通知アラームをモニタリングおよび作成できます。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。

を使用する AWS CloudTrail

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記録を提供します。によって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、リクエスト者、リクエスト日時などの詳細

CloudTrail を確認できます。例えば、データアクセスに影響するアクションのエントリ、特に CreateBucketAccessKey、GetBucketAccessKeys、DeleteBucketAccessKey、SetResourceAcc よび を識別 CloudTrailできますUpdateBucket。 AWS アカウントを設定すると、デフォルトで CloudTrail が有効になります。 CloudTrail コンソールで最近のイベントを表示できます。Lightsail バケットのアクティビティとイベントの継続的な記録を作成するには、 CloudTrail コンソールで証跡を作成します。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のモニタリング

AWS アカウントに登録されているプライマリ E メールアドレスを積極的にモニタリングします。AWS は、この E メールアドレスを使用して、ユーザーに影響を与える可能性のある新たなセキュリティ問題について連絡します。

AWS 広範な影響を与える運用上の問題は、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。

Lightsail バケットとオブジェクトへのアクセスを制御する

デフォルトでは、バケットとオブジェクトのすべての Amazon Lightsail オブジェクトストレージリソースはプライベートです。つまり、バケット所有者、それを作成した 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。（AWS CLI）、 AWS Command Line Interface AWS APIs AWS SDKs。
  - 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 のパブリックアクセスブロック – Amazon Simple Storage Service (Amazon S3) のアカウントレベルの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 の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アクセス許可に関係なく、すべての Amazon S3 バケットと Lightsail 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して、バケット内のすべてのオブジェクトへの認証された読み取り専用アクセスを他の AWS アカウントとそのユーザーに付与します。クロスアカウントアクセスは、オブジェクトを別の AWS アカウントと共有する場合に最適です。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 アカウントへのアクセスを許可できます。

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。API 、、AWS SDKs AWS APIs を使用してバケットとそのオブジェクトにアクセスできるように、アプリケーションでアクセスキーを設定できます。 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て、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 つは、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ことです。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: 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バケットを使用する」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 の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 の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アクセス許可に関係なく、すべての Amazon S3 バケットと Lightsail 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 SDKs、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のバケットを含む、アカウント内のすべてのバケットのブロックパブリックアクセス設定を設定できます。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。

Lightsail オブジェクトストレージ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。より大きなファイルをアップロードするには、Lightsail 、 AWS Command Line Interface （ AWS CLI ）API、または を使用します AWS SDKs。

Lightsail には、アップロードするファイルのサイズに応じて以下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 コンソールを使用すると、最大 2 GB のサイズのオブジェクトを 1 つアップロードできます。詳細については、このガイドの後半の「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 SDKsRESTAPIまたは を使用した 1 回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 AWS CLIする — 1 回の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 SDKs、RESTAPI、または を使用してオブジェクトをパート単位でアップロード AWS CLIする — マルチパートアップロード を使用するとAPI、5 MB から 5 TB までのサイズの単一の大きな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、大きなオブジェクトのアップロードエクスペリエンスを向上させ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、2 つのファイル sample1.jpgと 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 で動作する AWS CLI 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* ファイルをアップロードするバケットの名前。
- *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。
- *LocalDirectoryFire* 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。

例:

- Linux または Unix コンピュータの場合：

```
aws s3api put-object --bucket amzn-s3-demo-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 amzn-s3-demo-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のみのリクエストAWSCLI用に を設定する

Amazon S3 は、 経由のバケットアクセスをサポートしていますIPv6。デュアルスタックのエンドポイントを使用して、Amazon S3 API呼び出しIPv6でリクエストを行います。このセクションでは、 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IPv6。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 Amazon S3」を参照してください。の設定手順については AWS CLI、「Amazon Lightsail で動作する AWS Command Line Interface ように を設定する Amazon Lightsail」を参照してください。

> **Important**
>
> を使用するには、クライアントとバケットにアクセスするネットワークを有効にする必要がありますIPv6。詳細については、IPv6「到達可能性」を参照してください。

IPv6専用インスタンスから 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、指定された のデュアルスタック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。

を設定する AWS CLI

Config ファイルのプロファイルtrueで設定値を AWS 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送信します。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、 --endpoint-urlパラメータを任意の 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 または に設定します。以下の例では、*bucketname* また、*aws-region* をバケットの名前と 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」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、アクセスキーの作

成、バケットへのインスタンスのアタッチ、他のAWSアカウントへのアクセスの付与によって、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」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 でバケット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の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バケットのアクセスログ記録を有効にする
   - 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、IAMAmazon Lightsail」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について」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 で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」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」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」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」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: Amazon Lightsail バケットへの WordPress インスタンスの接続
    - 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 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」を参照してください。

Amazon Lightsail でのコンテナのデプロイと管理

Amazon Lightsail コンテナサービスは、コンテナをデプロイ、実行、管理できる非常にスケーラブルなコンピューティングおよびネットワークリソースです。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 コンテナサービスは、ローカルマシンで作成したイメージ、または Amazon ECR Public Gallery などのオンラインリポジトリからのイメージを使用して、 AWS インフラストラクチャでコンテナを実行できるようにするコンピューティング環境と考えることができます。

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。

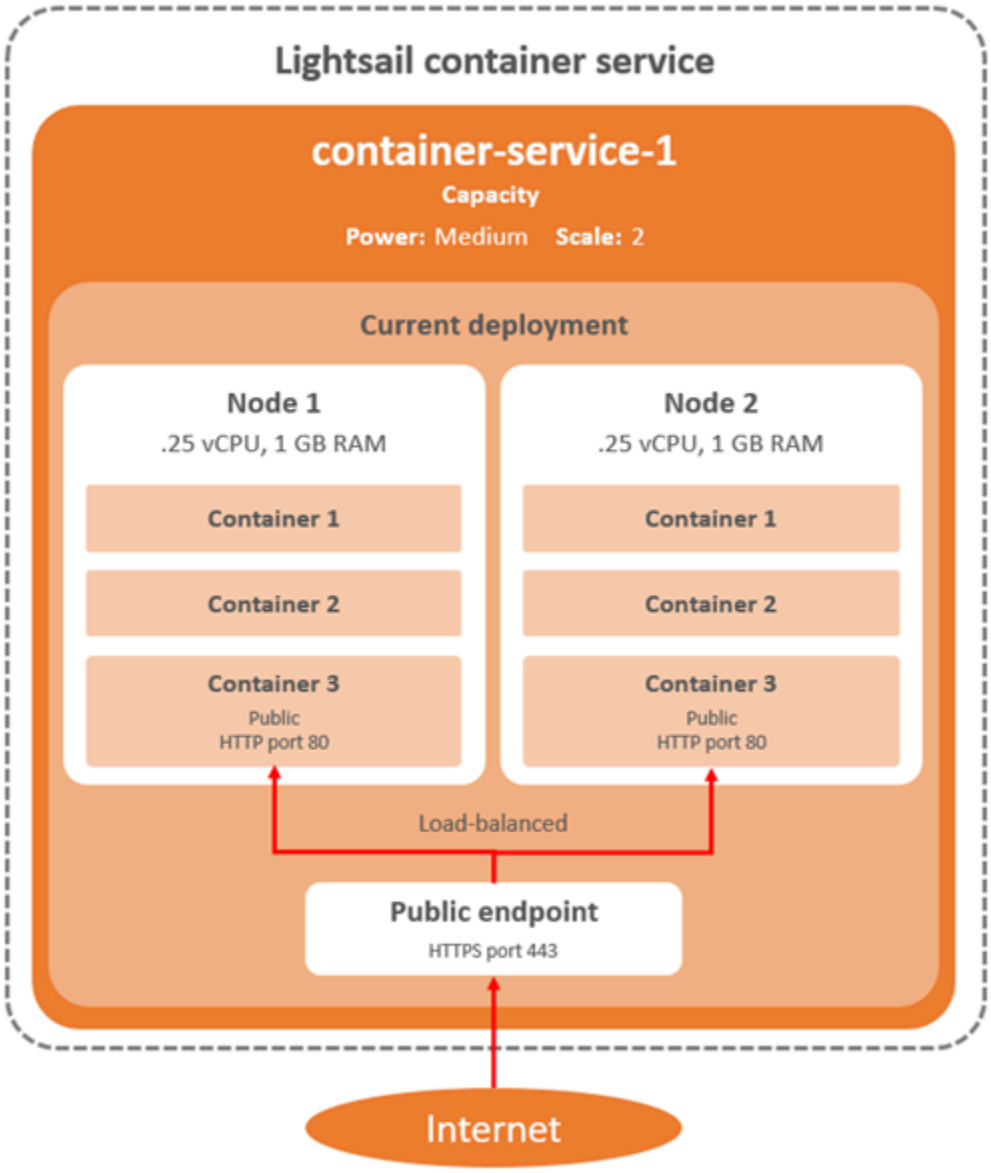
コンテナとコンテナイメージの詳細については、Docker ドキュメントの「コンテナとは」を参照してください。

Lightsail コンテナサービスの要素

以下は、開始する前に理解しておくべき Lightsail コンテナサービスの主要な要素です。

Lightsail コンテナサービス

コンテナサービスは、Lightsail が利用可能な任意の AWS リージョン で作成できる Lightsail コンピューティングリソースです。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 コンテナサービスの作成」および「Lightsail コンテナサービスの削除」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、現在のデプロイとともに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。クォータを超えるインターネットへのデータ転送では、超過料金が発生します。超過料金は によって異なり AWS リージョン 、GB あたり 0.09 USD から開始されます。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 の料金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、Windows で Docker、 AWS Command Line Interface （ AWS CLI ）、Lightsail Control (lightsailctl) プラグインを実行して、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。

コンテナサービスの ARN

Amazon リソースネーム (ARNs AWS 、リソースを一意に識別します。IAM ポリシーや API コールなど AWS、すべての でリソースを明確に指定する必要がある場合は、ARN が必要です。

コンテナサービスの ARN を取得するには、Lightsail API `GetContainerServices` アクションを使用し、 `serviceName`パラメータを使用してコンテナサービスの名前を指定します。次の例に示すように、コンテナサービスの ARN は、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、GetContainerServicesAmazon Lightsail API リファレンス」の「」を参照してください。

結果は次のように表示されます:

```
{
    "containerServices": [
        {
            "containerServiceName": "container-service-1",
            "arn": "arn:aws:lightsail: :111122223333:ContainerService/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
```

```
            "createdAt": "2024-01-01T00:00:00+00:00",
            "location": {
                "availabilityZone": "all",
                "regionName": "us-west-2"
        },
        .....
}
```

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。デフォルトドメインは としてフォーマットされます。ここでhttps://*<ServiceName>*.*<RandomGUID>*.*<AWSRegion>*.cs.amazonlightsail.com、*#ServiceName#* はコンテナサービスの名前、*<RandomGUID >* は AWS リージョン Lightsail アカウントの でランダムに生成されたコンテナサービスのグローバル AWS リージョン に一意の識別子、*#AWSRegion#* はコンテナサービスが作成された 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していません。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

すべてのコンテナサービスには、 という形式のプライベートドメインもあります。*<ServiceName>*.service.local*#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。これを行うには、リクエストのドメインを としてフォーマットします。ここで*<ServiceName>*.service.local:*<PortNumber>*、*#ServiceName#* はコンテナサービスの名前、*#PortNumber#* は接続先の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 証明書の作成」、「Lightsail コンテナサービスの SSL/TLS 証明書の検証」、および「Lightsail コンテナサービス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

コンテナイメージをローカルマシンからサービスにプッシュし、デプロイで使用する予定がある場合は、Lightsail コンテナサービスを管理するための一般的な手順は次のとおりです。

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 コンテナサービスの作成」を参照してください。
2. 独自のコンテナイメージを作成したいローカルマシンにソフトウェアをインストールして、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。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 コンテナサービスで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の作成と管理」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の表示と管理」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の有効化と管理」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 コンテナサービスの削除」を参照してください。

デプロイでパブリックレジストリのコンテナイメージを使用する予定がある場合は、Lightsail コンテナサービスを管理するための一般的な手順は次のとおりです。

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 コンテナサービスの作成」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の作成と管理」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の表示と管理」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の有効化と管理」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 コンテナサービスの削除」を参照してください。

Lightsail で高可用性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 Amazon Lightsail コンテナサービスを作成する方法と、設定できるコンテナサービス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要素を理解して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、現在のデプロイとともに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 の料金 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 コンテナサービスの削除」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

Lightsail コンテナサービスを作成するには、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、[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」ページで「 の変更 AWS リージョン AWS リージョン 」を選択し、コンテナサービスの 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。

      - デプロイ例を選択する – このオプションを選択すると、一連の事前設定されたデプロイパラメータを使用して Lightsail チームによってキュレートされたコンテナイメージを使用してデプロイが作成されます。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の作成と管理ガイドの「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。

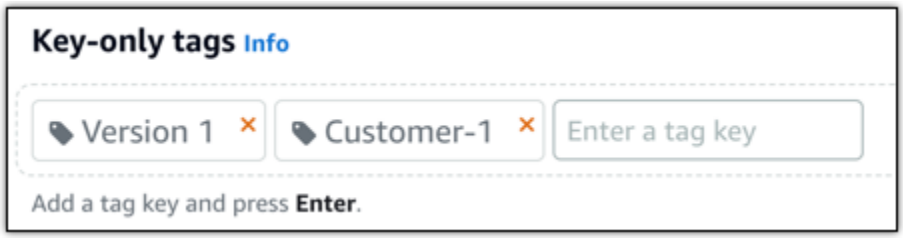
   コンテナサービス名は、次のものである必要があります。

   - 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。

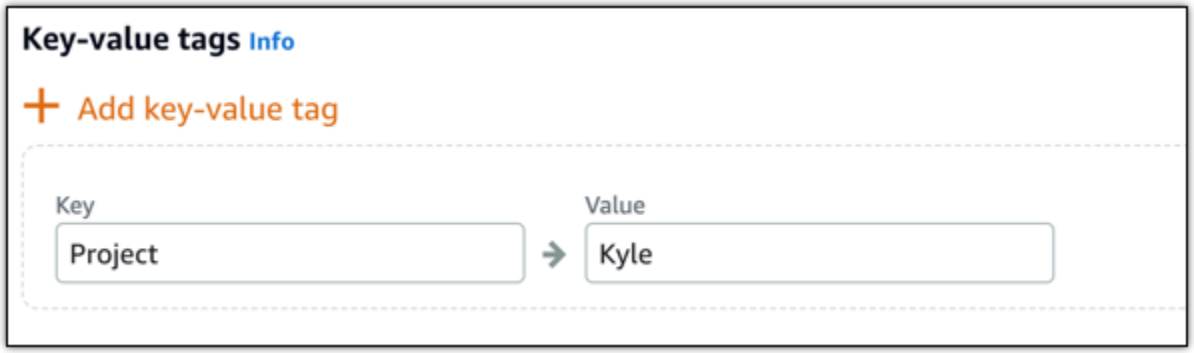
8. 以下のいずれかのオプションを選択して、コンテナサービスにタグを追加します。

   - [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 コンテナサービスの Docker イメージを構築してテストする

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
```

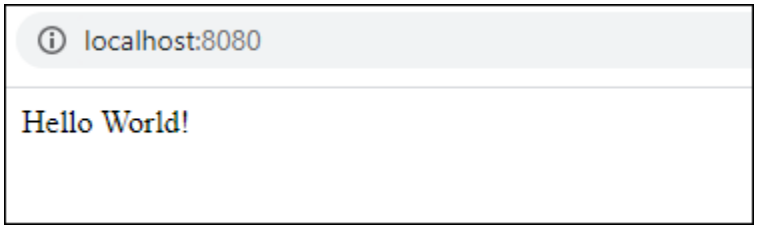
2. 次のコマンドを入力して、実行中のコンテナを表示します。

```
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 コンテナサービスで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> を停止するコンテナの名前に置き換え、削除します。

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、デプロイする準備ができたら、 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の管理」を参照してください。

トピック

- Lightsail コンテナサービスのコンテナイメージをプッシュ、表示、削除する
- Docker AWS CLI、およびコンテナ用の Lightsail Control プラグインをインストールする
- Lightsail コンテナサービスに Amazon ECR プライベートリポジトリへのアクセスを許可する

Lightsail コンテナサービスのコンテナイメージをプッシュ、表示、削除する

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保存されているコンテナイメージを削除する

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。
- 独自のコンテナイメージを作成したいローカルマシンにソフトウェアをインストールして、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。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 コンテナサービス用のコンテナイメージを作成を参照してください。

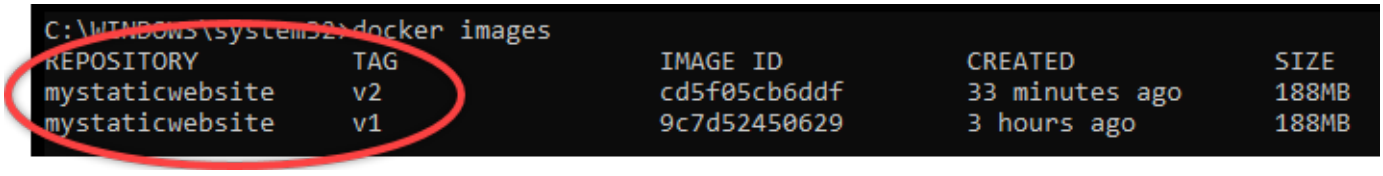
コンテナイメージをローカルマシンからコンテナサービスにプッシュする。

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#* とコンテナサービスの名前。
- *#ContainerImageLabel#* と、コンテナサービスに保存されているときにコンテナイメージに付けるラベル。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するコンテナイメージの名前で指定します。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。
- *#ImageTag#* コンテナサービスにプッシュするコンテナイメージのタグ。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。

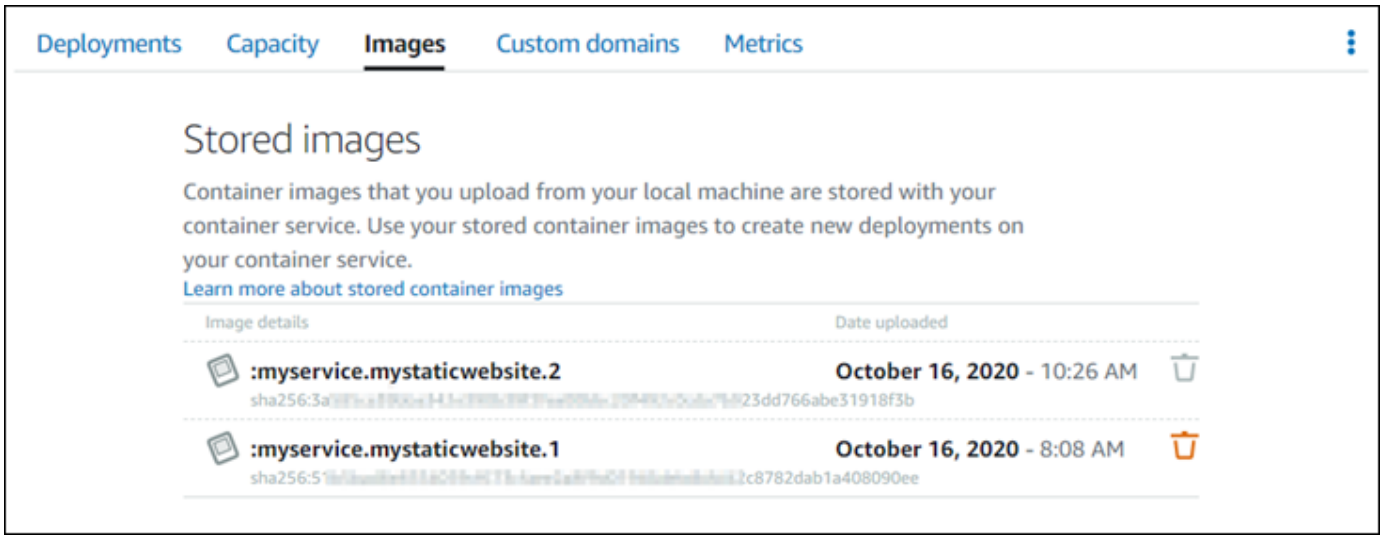
コンテナサービスに保存されているコンテナイメージを表示する

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 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 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。

Docker AWS CLI、およびコンテナ用の Lightsail Control 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し、Amazon ECR Public Gallery などのオンラインパブリックレジストリのコンテナイメージを使用して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 – Lightsail コンテナサービスで使用できる独自のコンテナイメージを実行、テスト、作成します。
- 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、Lightsail コンテナサービスにプッシュします。バージョン 2.1.1 以降は Lightsail Control プラグインで動作します。
- Lightsail Control (lightsailctl) プラグイン — がローカルマシン上のコンテナイメージにアクセス AWS CLI できるようにします。

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

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できる独自のコンテナイメージを作成する場合は、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 コンテナサービスのコンテナイメージを作成する」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。古いバージョンの Docker は、このガイドで後述する AWS CLI および Lightsail Control (lightsailctl) プラグインで動作するとは限りません。

のインストール AWS CLI

AWS CLI は、コマンドラインシェルのコマンドを使用して Lightsail などの AWS サービスとやり取りできるオープンソースツールです。をインストールして使用し、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 AWS CLI プッシュする必要があります。

AWS CLI は、次のバージョンで使用できます。

- 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。これは の最新のメジャーバージョン AWS CLI であり、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、すべての最新機能をサポートしています。バージョン 2.1.1 以降は Lightsail Control プラグインで動作します。
- バージョン 1.x – 下位互換性のために AWS CLI 利用可能な の以前のバージョン。このバージョンは、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機能をサポートしていません。したがって、代わりに AWS CLI バージョン 2 をインストールする必要があります。

AWS CLI バージョン 2 は、Linux、macOSオペレーティングシステムで使用できます。 AWS CLI これらのオペレーティングシステムに 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」を参照してください。

Lightsail コントロールプラグインをインストールする

Lightsail Control (lightsailctl) プラグインは、 がローカルマシンで作成したコンテナイメージ AWS CLI にアクセスできるようにする軽量アプリケーションです。これにより、コンテナイメージを Lightsail コンテナサービスにプッシュして、サービスにデプロイできます。

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。

- AWS CLI lightsailctl プラグインを使用するには、 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、機能強化のために時折更新されます。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、lightsailctl 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 日にリリース) — 初回リリースでは、 AWS CLI バージョン 2 で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追加されました。

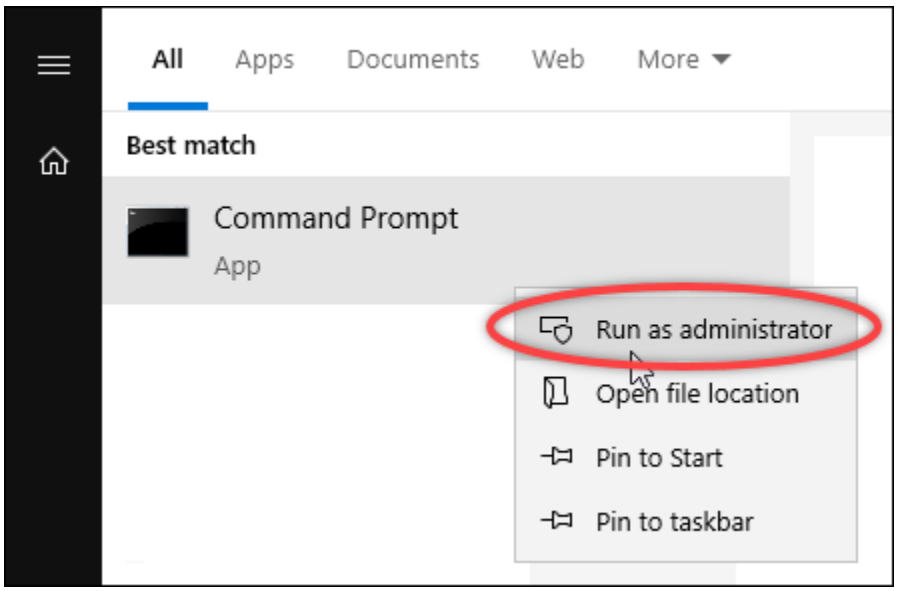
Windows に lightsailctl プラグインをインストールする

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
```

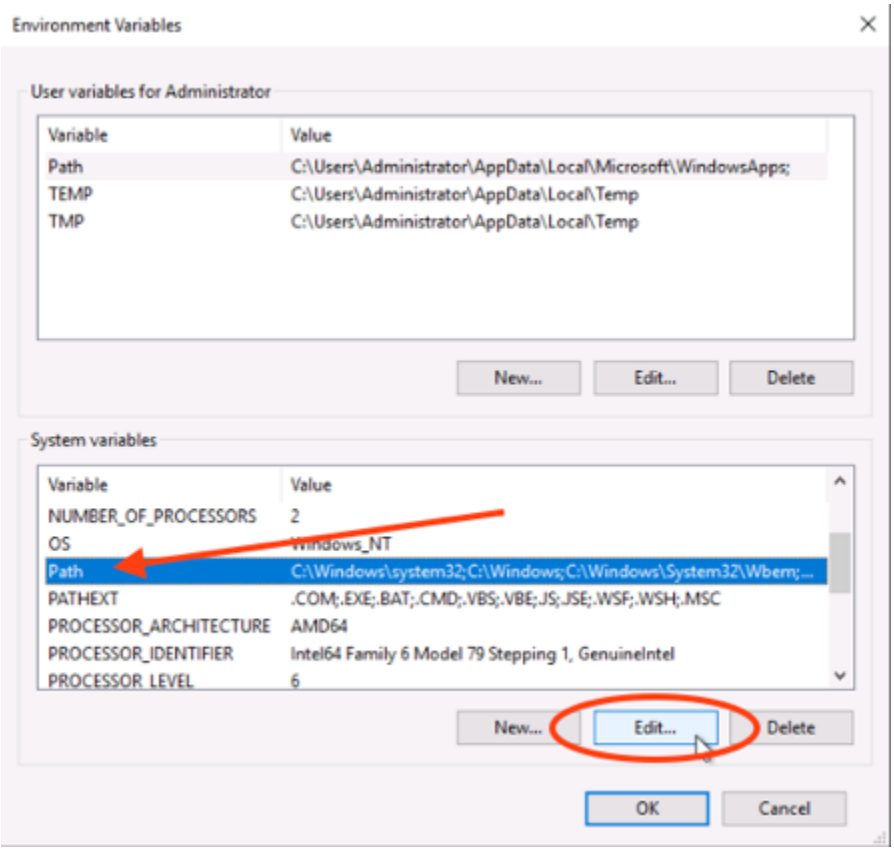
次の例のような結果が表示されます。

```
C:\WINDOWS\system32>setx PATH "%PATH%;C:\Temp\lightsailctl\" /M

SUCCESS: Specified value was saved.
```

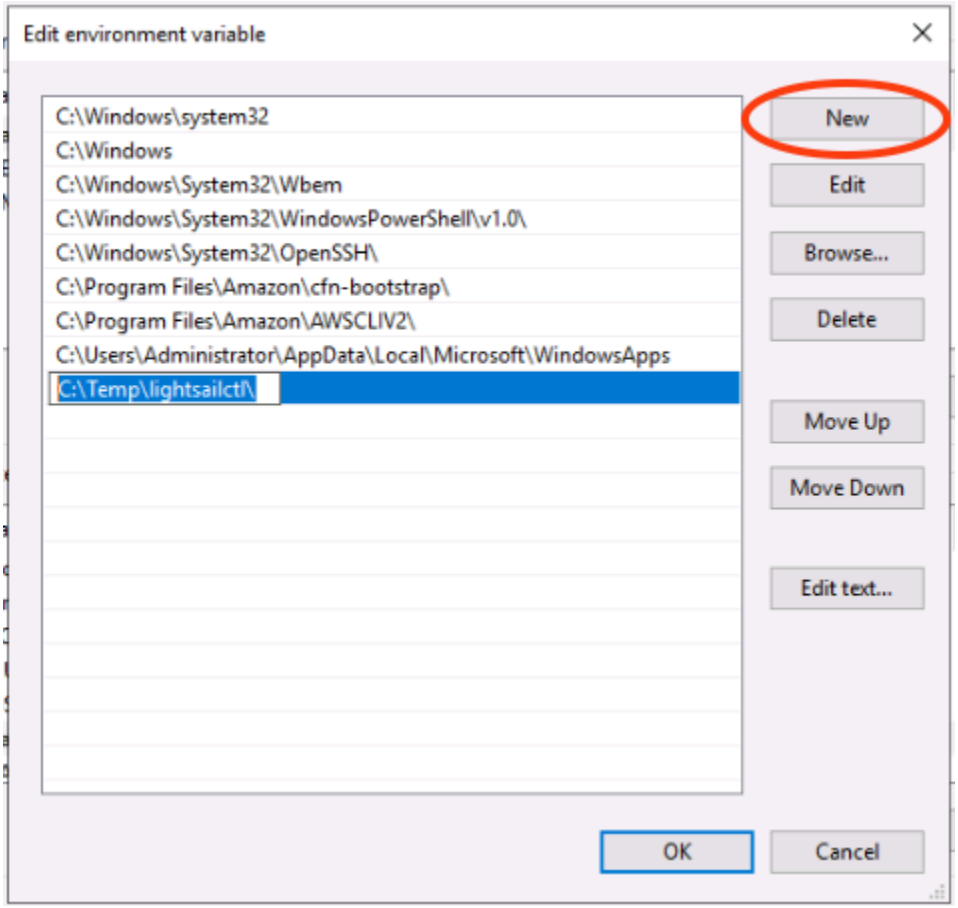
`setx` コマンドは 1,024 文字を超えると切り捨てられます。PATH に複数の変数がすでに設定されている場合は、以下の手順を使用して PATH 環境変数を手動で設定します。

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、lightsailctl 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、lightsailctl 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、 を使用してコンテナイメージ AWS CLI 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 に lightsailctl 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、lightsailctl 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、 を使用してコンテナイメージ AWS CLI 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 コンテナサービスに Amazon ECR プライベートリポジトリへのアクセスを許可する

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許可を持つ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 Amazon ECR プライベートリポジトリ へのアクセスを許可できます AWS リージョン。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 () 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へのアクセスを管理できますAWS CLI。ただし、プロセスが簡素化されるため、Lightsail コンソールを使用することをお勧めします。

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

Amazon ECR プライベートリポジトリへの Lightsail コンテナサービスへのアクセスを管理するユーザーには、IAM で次のいずれかのアクセス許可ポリシーが必要です。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/*"
        }
    ]
}
```

ポリシーで、 を AWS アカウント ID 番号*AwsAccountId*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 コンテナサービスは、アクセスするプライベートリポジトリ AWS リージョン と同じ にある必要があります。
- *AwsAccountId* — AWS アカウント ID 番号。
- *RepositoryName* — アクセスを管理するプライベートリポジトリの名前。

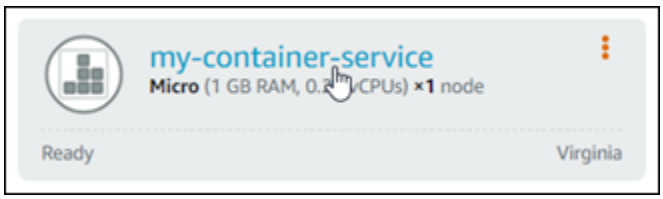
以下は、アクセス許可ポリシーに例の値を入力した一例です。

```
{
    "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"
        }
    ]
}
```

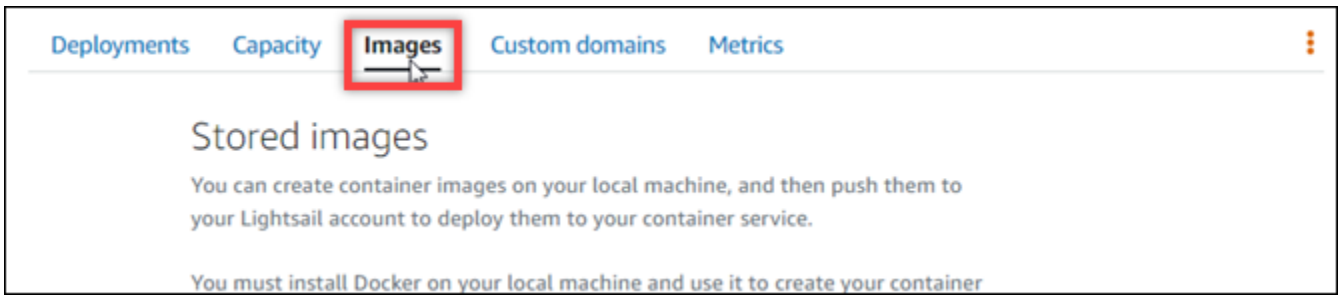
Lightsail コンソールを使用してプライベートリポジトリへのアクセスを管理する

Lightsail コンソールを使用して、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、[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。

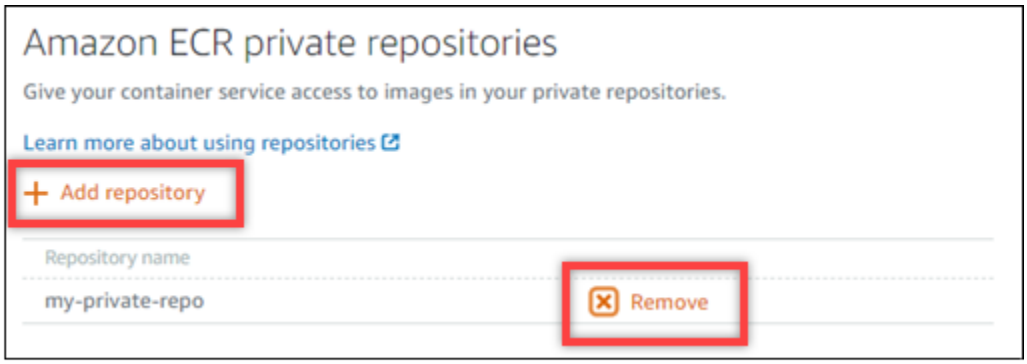


4. [Images] (イメージ) タブを選択します。

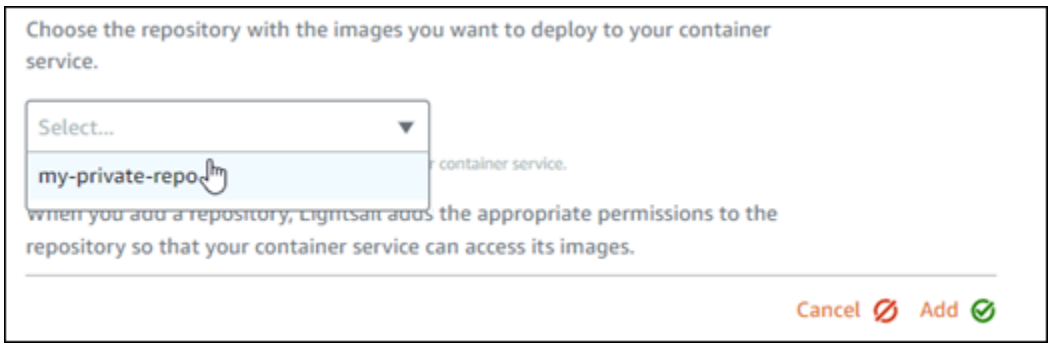


5. [リポジトリの追加] を選択すると、コンテナサービスの Amazon ECR プライベートリポジトリへのアクセス権が付与されます。

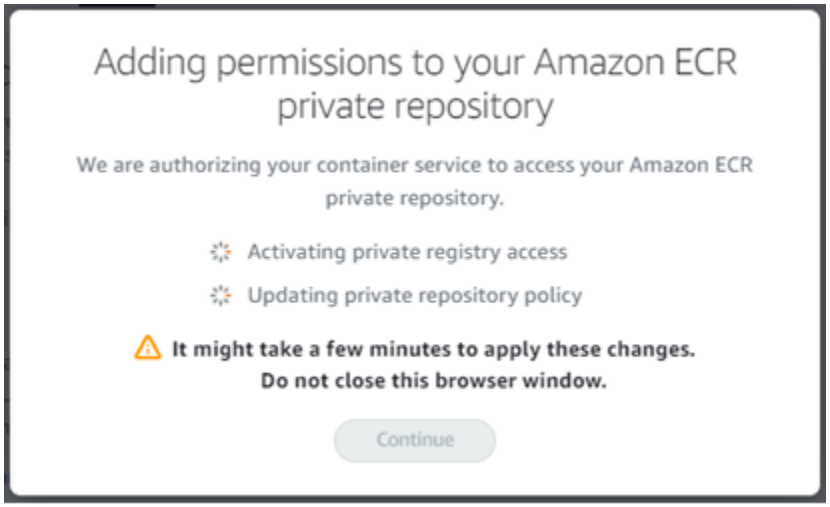
> **Note**
>
> [削除] を選択すると、以前に追加した Amazon ECR プライベートリポジトリからコンテナサービスのアクセスが削除されます。



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。

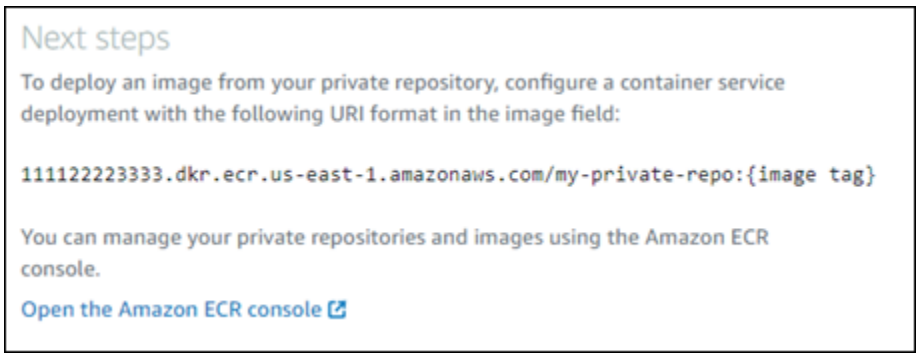


Lightsail は、プリンシパル Amazon リソースネーム (ARN) を含むコンテナサービスの Amazon ECR イメージプーラー IAM ロールをアクティブ化するのに少し時間がかかります。Lightsail は、選択した Amazon ECR プライベートリポジトリのアクセス許可ポリシーに IAM ロールプリンシパル ARN を自動的に追加します。これにより、コンテナサービスはプライベートリポジトリとそのイメージにアクセスできるようになります。プロセスが完了し、[Continue] (続行) を選択できることを示すモーダルが表示されるまで、ブラウザウィンドウは閉じないでください。



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記載されています。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 Lightsail コンテナサービスから 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、次のステップが必要です。

> **Important**
>
> Lightsail コンソールを使用して、Amazon ECR プライベートリポジトリへの Lightsail コンテナサービスへのアクセスを管理することをお勧めします。このプロセスが簡素化されるためです。詳細については、このガイドの前半の「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にする AWS CLI `update-container-service` — Lightsail の 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 `set-repository-policy` コマンドを使用して AWS CLI 、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、既存のポリシーを持つプライベートリポジトリに追加します。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つプライベートリポジトリにポリシーを追加する — Amazon ECR の `set-repository-policy` コマンドを使用して AWS CLI 、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します。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

Lightsail コンテナサービスの Amazon ECR イメージプーラー IAM ロールをアクティブ化または非アクティブ化するには、次の手順を実行します。Lightsail の `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 AWS CLI アクティブ化または非アクティブ化できます。詳細については、「 コマンドリファレンスupdate-container-service」の「」を参照してください。 AWS CLI

> **Note**
>
> この手順を続行する前に、 をインストール AWS CLI し、Lightsail 用に設定する必要があります。詳細については、「Lightsail で動作するように を設定する AWS CLI」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 ロールをアクティブ化または非アクティブ化するコンテナサービスの名前。
- *RoleActivationState* — Amazon ECR イメージプーラー IAM ロールのアクティベーション状態。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* — AWS リージョン コンテナサービスのコード (例: `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 ロールプリンシパル ARN を取得するコンテナサービスの名前。
- *AwsRegionCode* — AWS リージョン コンテナサービスのコード (例: 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 には コマンド AWS CLI get-repository-policyを使用できます。詳細については、 コマンドリファレンスupdate-container-serviceの「」を参照してください。 AWS CLI

> **Note**
>
> この手順を続行する前に、 をインストール AWS CLI 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 コンテナサービスのアクセスを設定するプライベートリポジトリの名前。
- *AwsRegionCode* — AWS リージョン プライベートリポジトリのコード (例: us-east-1) 。

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 が含まれている必要があります。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、このガイドの前半の「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、 コマンドリファレンスset-repository-policyの「」を参照してください。 AWS CLI

> **Note**
>
> この手順を続行する前に、 をインストール AWS CLI 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"
```

```
      ]
    }
  ]
}
```

テキストで、 を、この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 ロールプリンシパル ARN *IamRolePrincipalArn*に置き換えます。

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。
- *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。
- *AwsRegionCode* — AWS リージョン プライベートリポジトリのコード (例: us-east-1) 。

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 番号。
- *AwsRegionCode* — AWS リージョン プライベートリポジトリのコード (例: `us-east-1`) 。
- *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。
- *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からのコンテナイメージのタグ。

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 を含む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、このガイドの前半

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、 コマンドリファレンスset-repository-policyの「」を参照してください。 AWS CLI

> **Note**
>
> この手順を続行する前に、 をインストール AWS CLI 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 コンテナサービスのアクセスを設定するプライベートリポジトリの名前。
   - *AwsRegionCode* — AWS リージョン プライベートリポジトリのコード (例: `us-east-1`) 。

   例：

   ```
   aws ecr get-repository-policy --repository-name my-private-repo --region us-east-1
   ```

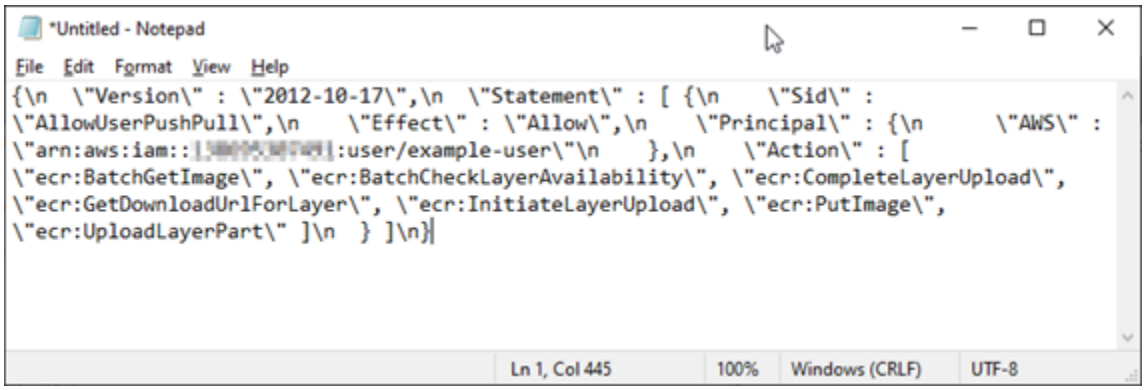
3. レスポンスに、既存のポリシーをコピーし、次のステップに進みます。

   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。

```
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}"
}
```

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. テキストで、 を、この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 ロールプリンシパル ARN *IamRolePrincipalArn*に置き換えます。

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。
- *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。
- *AwsRegionCode* — AWS リージョン プライベートリポジトリのコード (例: us-east-1) 。

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 番号。
- *AwsRegionCode* — AWS リージョン プライベートリポジトリのコード (例: us-east-1) 。
- *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。
- *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からのコンテナイメージのタグ。

例：

```
111122223333.dkr.ecr.us-east-1.amazonaws.com/my-private-repo:myappimage
```

Lightsail でのコンテナサービスのデプロイの作成と管理

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。

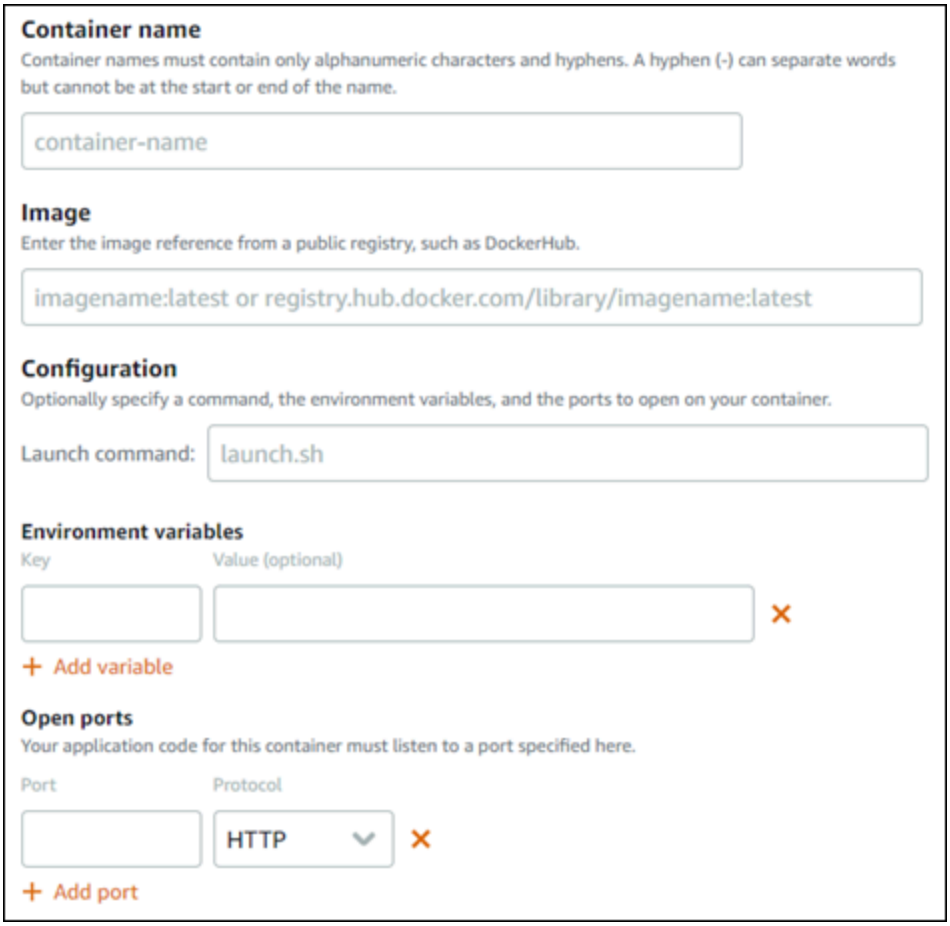
- Lightsail アカウントでコンテナサービスを作成します。詳細については、Amazon Lightsail コンテナサービスの作成を参照してください。
- コンテナサービスでコンテナを起動する際に使用するコンテナイメージを特定します。
  - Amazon ECR Public Gallery などのパブリックレジストリでコンテナイメージを検索します。詳細については、「Amazon ECR Public ユーザーガイド」の「Amazon ECR Public Gallery」を参照してください。
  - ローカルマシンでコンテナイメージを作成し、Lightsail コンテナサービスにプッシュします。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 コンテナサービス用にコンテナイメージを作成するおよび Amazon Lightsail コンテナサービスで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。

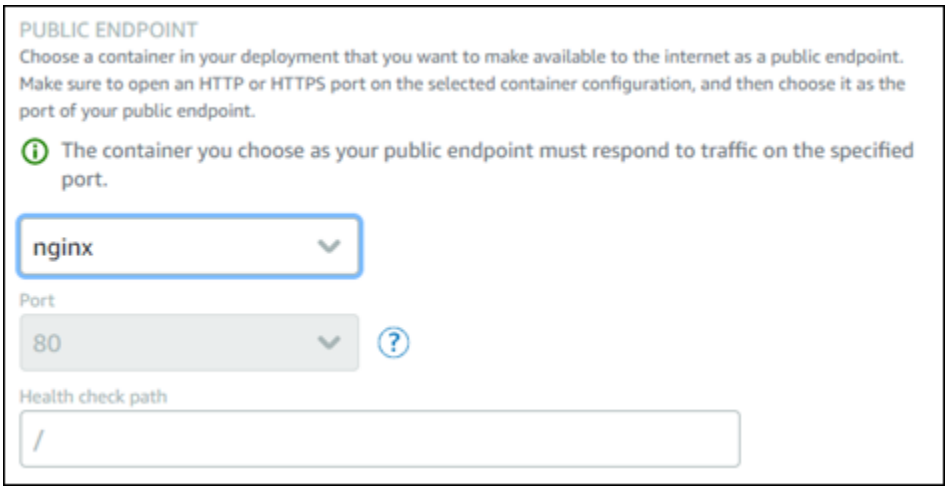
- 起動コマンド – シェルスクリプト、または コンテナの作成時にコンテナを設定する bash スクリプトを実行するための起動コマンドを指定します。起動コマンドでは、ソフトウェアの追加、ソフトウェアの更新、あるいはコンテナの設定などを他の方法で行うことができます。
- 環境可変 – 環境可変を指定します。環境可変は、コンテナによって実行されるアプリケーションまたはスクリプトの動的設定を提供するキーバリューパラメーターです。
- オープンポート – コンテナで開くポートとプロトコルを指定します。HTTP、HTTPS、TCP、および UDP 経由でポートが開くように指定できます。コンテナサービスの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。デフォルトドメインは の形式です。ここでhttps://*<ServiceName>*.*<RandomGUID>*.*<AWSRegion>*.cs.amazonlightsail.com、*#ServiceName#* はコンテナサービスの名前、*<RandomGUID >* は Lightsail アカウントの AWS リージョンでランダムに生成されたコンテナサービスのグローバルに一意の識別子、*#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していません。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

すべてのコンテナサービスには、 という形式のプライベートドメインもあります。*<ServiceName>*.service.local*#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。これを行うには、リクエストのドメインを としてフォーマットします。ここで*<ServiceName>*.service.local:*<PortNumber>*、*#ServiceName#* はコンテナサービスの名前、*#PortNumber#* は接続先の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。

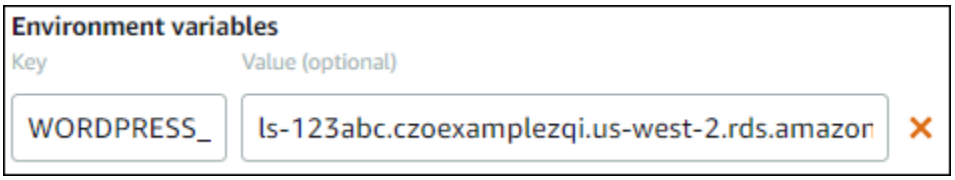
コンテナ間の通信

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 です`ls-123abc.czoexamplezqi.us-west-2.rds.amazonaws.com`。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ログの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、同じコンテナサービスに新しいデプロイを作成できます。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ログの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を表示する

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 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、[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

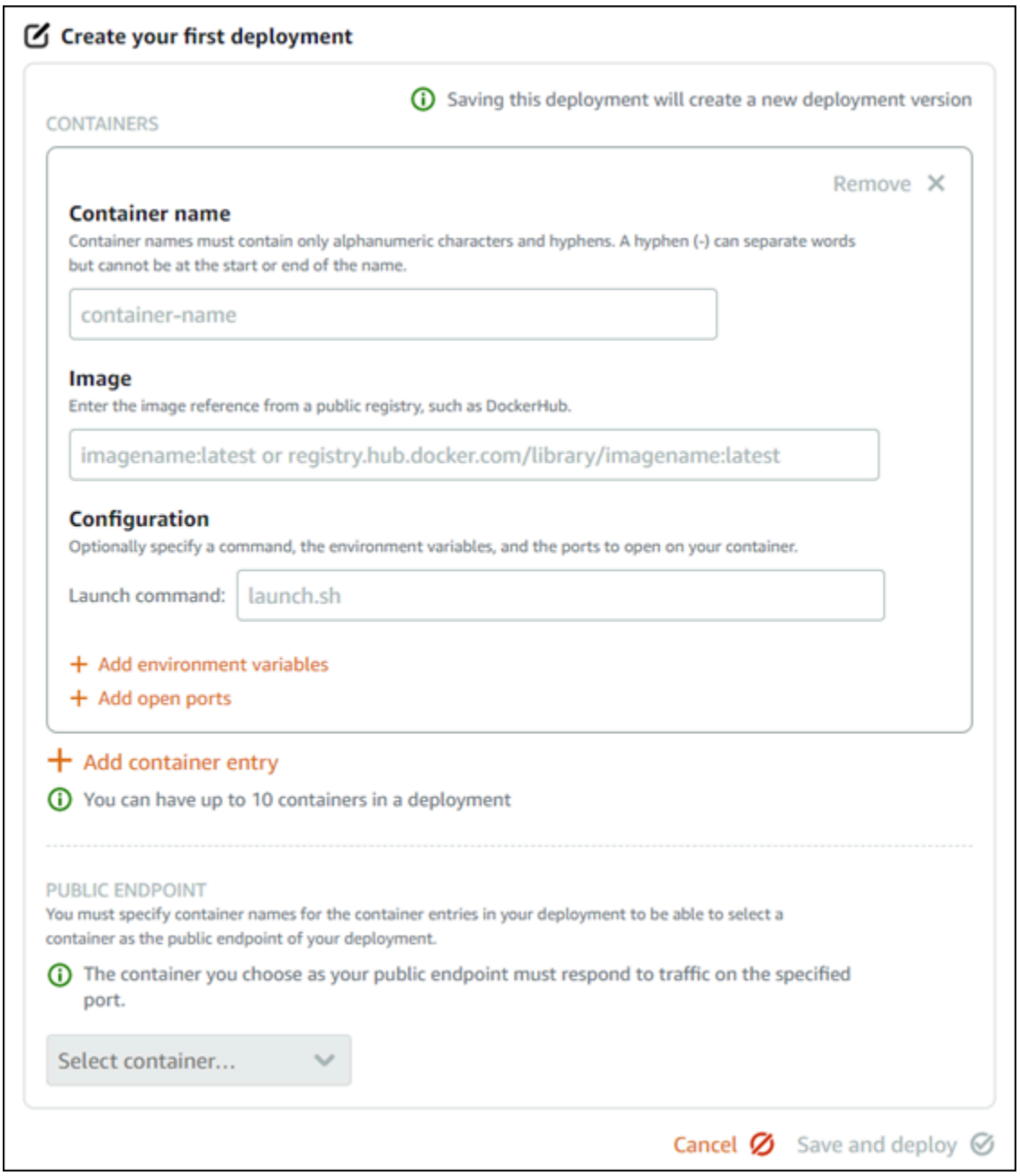
プロイが新しいデプロイバージョンとして保存されます。詳細については、Amazon Lightsail コンテナサービスのデプロイバージョンの表示と管理を参照してください。

1. Lightsail コンソール にサインインします。
2. Lightsail 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スケールする

Amazon Lightsail コンテナサービスの容量は、その規模と能力で構成されます。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、現在のデプロイとともに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

Lightsail コンテナサービスの容量を変更するには、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、[Containers] (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した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

スタートする前に、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 ホームページで [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 コンテナサービスログを分析する

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

始める前に、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。

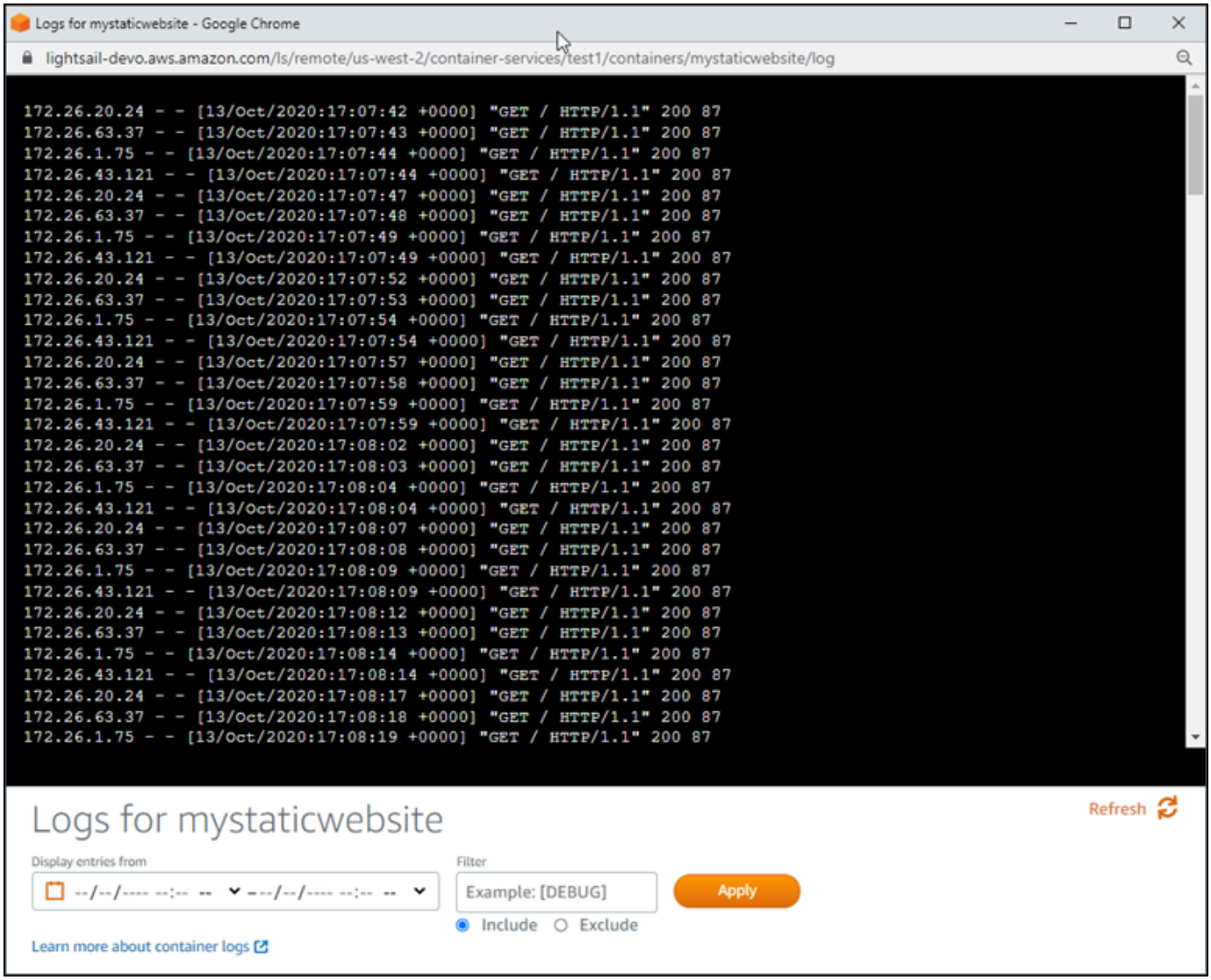
1. Lightsail コンソール にサインインします。
2. Lightsail ホームページで [コンテナ] タブを選択します。
3. コンテナログを表示するコンテナサービスの名前を選択します。
4. コンテナサービス管理ページで、[デプロイ] タブを選択します。

   [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 でカスタムドメインによる安全なウェブアクセスを有効にする

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**
>
> Lightsail コンテナサービス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、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。

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、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、コンテナサービスにドメインの頂点 (例: `example.com`) とサブドメイン (例: `www.example.com`) のトラフィックをルートできます。

前提条件

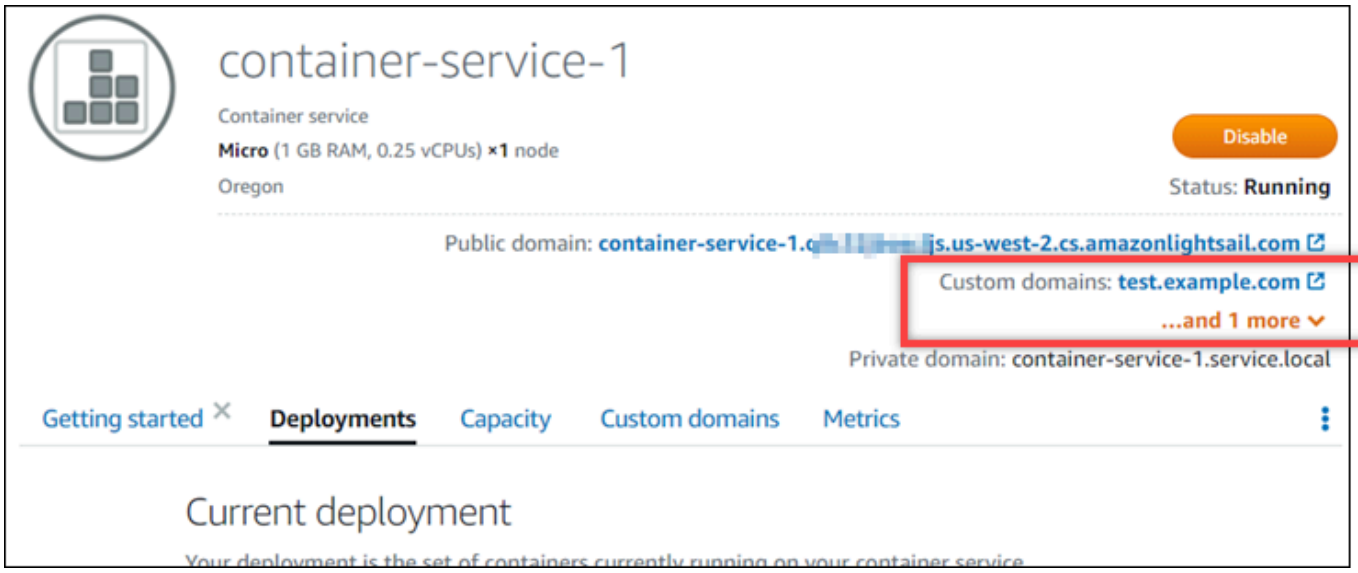
始める前に、Lightsail コンテナサービスを作成する必要があります。詳細については、「Amazon Lightsail コンテナサービスの作成」を参照してください。。

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 ホームページで [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添付します。

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 [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 ホームページで [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 を使用してドメイントラフィックを 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

サービスのカスタムドメインを有効にした場合は、Amazon Lightsail コンテナサービスにメンバードメイン名を紐づける必要があります。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ています。Lightsail DNS ゾーンの詳細については、Amazon Lightsail 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 にルーティングする」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 のホームページで、[Containers] (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 & 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 を使用してドメイントラフィックを Lightsail コンテナサービスにルーティングする

などの登録済みドメインのトラフィックをexample.com、Amazon 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。そのためには、Lightsail コンテナサービスのデフォルトドメインを指すエイリアスレコードをドメインのホストゾーンに追加します。

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示します。これは、 AWS Command Line Interface () を使用してのみ実行できますAWS CLI。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 DNS 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」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。
- Lightsail コンテナサービスにアプリケーションをデプロイします。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 AWS CLI で を設定します。詳細については、「Lightsail で動作する AWS CLI ように 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 IDs 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) にある場合 AWS リージョン、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Z0959753D43BBB908BAVときに、ホストゾーン ID を指定する必要があります。 Route 53

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 IDs を次に示します。

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

を使用して Route 53 のドメインのホストゾーンに DNS レコードを追加する場合は AWS CLI、レコードの設定パラメータのセット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 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。
- Lightsail コンテナサービスで使用する登録済みドメイン名を持つ#### (例: `example.com`または `www.example.com`)。Lightsail コンテナサービスでドメインのルートを使用するには、ドメインのサブドメインスペースに @記号を指定する必要があります (例: `@.example.com`)。
- 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 を含む *LightsailContainerServiceHostedZoneID*。詳細については、このガイドの前半のIDs を取得する」を参照してください。
- *LightsailContainerServiceAddress* Lightsail コンテナサービスのパブリックドメイン名。これを取得するには、Lightsail コンソールにサインインし、コンテナサービスを参照し、コンテナ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: `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、 コマンドリファレンスchange-resource-record-setsの「」を参照してください。 AWS CLI

> **Note**
>
> この手順を続行する前に、 をインストール AWS CLI 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 で動作する AWS CLI 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、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。

- Route 53 で登録されたドメインのホストゾーンの ID を持つ *HostedZoneID*。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 IDs のリストを取得します。 Route 53
- *PathToJsonFile* 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。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:

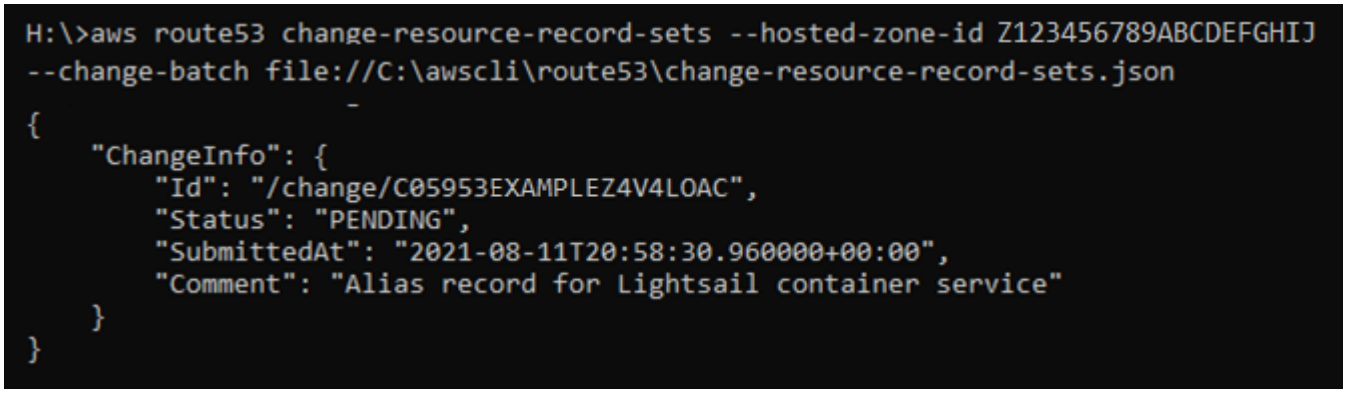
Linux または Unix コンピュータの場合は、次の操作を行います。

```
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。



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。その後、Route 53 に登録された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のルーティングを開始する必要があります。

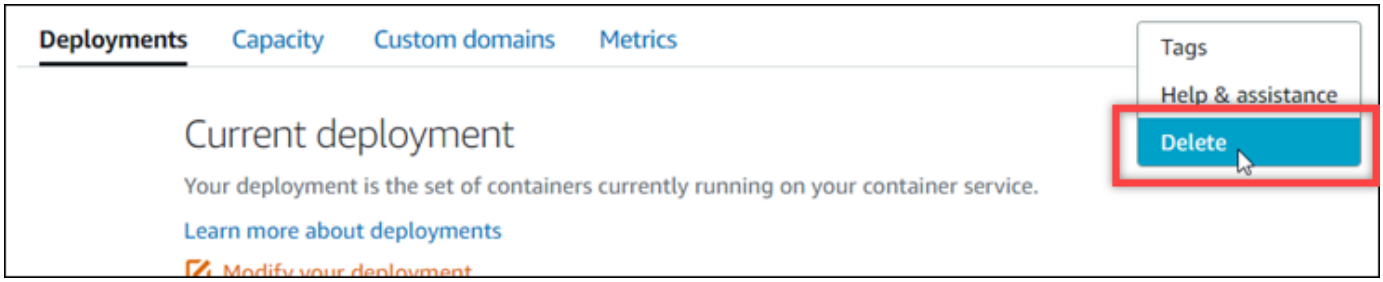
Lightsail コンテナサービスを削除する

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 ホームページで [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。

Amazon Lightsail のセキュリティ

のクラウドセキュリティが最優先事項 AWS です。お客様は AWS 、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メリットを得られます。

セキュリティは、 AWS とユーザーの間で共有される責任です。責任共有モデルでは、これを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しています。

- クラウドのセキュリティ — AWS クラウドで AWS 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 AWS を担います。 AWS また、 では、安全に使用できるサービスも提供しています。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。
- クラウドのセキュリティ — お客様の責任は、使用する AWS サービスによって決まります。また、お客様は、データの機密性、会社の要件、適用される法律や規制など、その他の要因についても責任を負います。

このドキュメントは、Amazon Lightsail を使用する際の責任共有モデルの適用方法を理解するのに役立ちます。以下のトピックでは、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ように Amazon Lightsail を設定する方法について説明します。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 のサービスの使用方法についても説明します。

Amazon Lightsail 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 がインフラストラクチャ AWS 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環境を AWS 設計するには、「 Security Pillar AWS Well-Architected Framework」の「Infrastructure Protection」を参照してください。

が AWS 公開した 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 は必須で TLS 1.3 がお勧めです。
- DHE (楕円ディフィー・ヘルマン鍵共有) や ECDHE (楕円曲線ディフィー・ヘルマン鍵共有) などの完全前方秘匿性 (PFS) による暗号スイート。これらのモードは、Java 7 以降など、ほとんどの最新システムでサポートされています。

また、リクエストには、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 ServiceAWS 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 の耐障害性

AWS グローバルインフラストラクチャは、 AWS リージョンと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 リージョンは、低レイテンシー、高スループット、および高度に冗長なネットワークで接続された、物理的に分離および分離された複数のアベイラビリティーゾーンを提供します。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 および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。

Amazon Lightsail には、 AWS 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をサポートするのに役立ついくつかの機能が用意されています。

- 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 Identity and Access Management

対象者

AWS Identity and Access Management （ IAM) の使用方法は、Amazon Lightsail で行う作業によって異なります。

サービスユーザー – ジョブを実行するために Amazon Lightsail サービスを使用する場合、管理者から必要な認証情報とアクセス許可が与えられます。さらに多くの Amazon Lightsail 機能を使用して

作業を行う場合は、追加のアクセス許可が必要になることがあります。アクセスの管理方法を理解しておくと、管理者に適切な許可をリクエストするうえで役立ちます。Amazon Lightsail の機能にアクセスできない場合は、「ID とアクセス管理のトラブルシューティング (IAM) 」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 へのフルアクセスがあります。従業員がどの Amazon Lightsail 機能やリソースにアクセスするかを決めるのは管理者の仕事です。次に、サービスユーザーのアクセス許可を変更するリクエストをIAM管理者に送信する必要があります。このページの情報を確認して、 の基本概念を理解してください IAM。会社で Amazon Lightsail IAMと を併用する方法の詳細については、Amazon LightsailIAM」を参照してください。

IAM 管理者 – IAM管理者は、Amazon Lightsail へのアクセスを管理するポリシーの作成方法の詳細について確認する場合があります。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を表示するにはIAM、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

認証とは、ID 認証情報 AWS を使用して にサインインする方法です。を使用したサインインの詳細については AWS Management Console、 ユーザーガイドのIAM「コンソールとサインインページIAM」を参照してください。

AWS アカウント ルートユーザー、 IAM ユーザー、または IAMロールを引き受けることによって認証 (にサインイン AWS) される必要があります。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セットアップしていました。別の会社の認証情報 AWS を使用して にアクセスすると、間接的にロールを引き受けることになります。

に直接サインインするにはAWS Management Console、ルートユーザーの E メールまたはIAMユーザー名でパスワードを使用します。ルートユーザーまたはIAMユーザーアクセスキーを使用して AWS プログラムで にアクセスできます。 AWS は、 認証情報を使用してリクエストに暗号で署名するための SDKおよび 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。これを行うには、署名バージョン 4 を使用します。これは、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 では、アカウントのセキュリティを高め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、「 ユーザーガイド」の「 での多要素認証 (MFA) AWS の使用IAM」を参照してください。

AWS アカウント ルートユーザー

を作成するときは AWS アカウント、アカウント内のすべての およびリソースへの AWS サービス 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 ID は AWS アカウント 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でアクセスできます。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を必要とするタスク」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、単一のユーザーまたはアプリケーションに対して特定のアクセス許可 AWS アカウント を持つ 内のアイデンティティです。可能な場合は、パスワードやアクセスキーなどの長期的な認証情報を持つIAMユーザーを作成するのではなく、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ユーザーとの長期的な認証情報を必要とする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で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ユーザーのコレクション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、 という名前のグループを作成しIAMAdmins、そのグループにIAMリソースを管理するアクセス許可を付与できます。

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、ロールでは一時認証情報が提供されます。詳細については、「 ユーザーガイド」のIAM「 (ロールではなく) ユーザーを作成する場合IAM」を参照してください。

IAM ロール

IAM ロールは、特定のアクセス許可 AWS アカウント を持つ 内のアイデンティティです。これは IAM ユーザーと似ていますが、特定のユーザーに関連付けられていません。IAM ロール を切り替え

る AWS Management Console ことで、 で 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、 または AWS API オペレーションを AWS CLI 呼び出すか、カスタム を使用しますURL。ロールの使用方法の詳細については、「 ユーザーガイド」のIAM「ロールの使用IAM」を参照してください。

IAM 一時的な認証情報を持つ ロールは、以下の状況で役立ちます。

- フェデレーションユーザーアクセス – フェデレーティッド ID に許可を割り当てるには、ロールを作成してそのロールの許可を定義します。フェデレーティッド ID が認証されると、その ID はロールに関連付けられ、ロールで定義されている許可が付与されます。フェデレーションのロールの詳細については、「 ユーザーガイド」の「サードパーティー ID プロバイダーのロールの作成IAM」を参照してください。IAM Identity Center を使用する場合は、アクセス許可セットを設定します。ID が認証後にアクセスできる内容を制御するために、IAMIdentity Center はアクセス許可セットを のロールに関連付けますIAM。アクセス許可セットの詳細については、「AWS IAM Identity Center ユーザーガイド」の「アクセス許可セット」を参照してください。
- 一時的なIAMユーザーアクセス許可 – IAM ユーザーまたはロールは、 IAMロールを引き受けて、特定のタスクに対して異なるアクセス許可を一時的に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、別のアカウントのユーザー (信頼されたプリンシパル) が自分のアカウントのリソースにアクセスすることを許可できます。クロスアカウントアクセスを許可する主な方法は、ロールを使用することです。ただし、一部の では AWS サービス、 (ロールをプロキシとして使用する代わりに) ポリシーをリソースに直接アタッチできます。クロスアカウントアクセスのロールとリソースベースのポリシーの違いについては、「 ユーザーガイド」の「 でのクロスアカウントリソースアクセスIAMIAM」を参照してください。
- クロスサービスアクセス — 一部の は、他の の機能 AWS サービス を使用します AWS サービス。例えば、サービスで呼び出しを行うと、そのサービスが Amazon でアプリケーションを実行EC2したり、Amazon S3 にオブジェクトを保存したりするのが一般的です。サービスでは、呼び出し元プリンシパルの許可、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) – IAM ユーザーまたはロールを使用して でアクションを実行すると AWS、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、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。FAS は、 を呼び出すプリンシパルのアクセス許可を AWS サービス、ダウンストリームサービス AWS サービス へのリクエストのリクエストと組み合わせて使用します。FAS リクエストは、サービスが他の AWS サービス またはリソースとのやり取りを完了する必要があるリクエストを受け取った場合にのみ行われます。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 – サービスロールは、ユーザーに代わってアクションを実行するためにサービスが引き受ける IAMロールです。IAM 管理者は、 内からサービスロールを作成、変更、削除できますIAM。詳細については、「 ユーザーガイド」の「 にアクセス許可を委任するロールの作成 AWS サービスIAM」を参照してください。
- サービスにリンクされたロール – サービスにリンクされたロールは、 にリンクされたサービスロールの一種です AWS サービス。サービスは、ユーザーに代わってアクションを実行するロールを引き受けることができます。サービスにリンクされたロールは に表示され AWS アカウント 、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アクセス許可を表示できますが、編集することはできません。

- Amazon で実行されているアプリケーション EC2 – IAMロールを使用して、EC2インスタンスで実行され、 AWS CLI または AWS API リクエストを行うアプリケーションの一時的な認証情報を管理できます。これは、EC2インスタンス内にアクセスキーを保存するよりも望ましいです。 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、インスタンスにアタッチ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 ロールが含まれており、EC2インスタンスで実行されているプログラムが一時的な認証情報を取得できるようにします。詳細については、「 ユーザーガイド」の「 IAMロールを使用して Amazon EC2インスタンスで実行されているアプリケーションにアクセス許可を付与するIAM」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、「 ユーザーガイド」の「 (ユーザーではなく) IAMロールを作成する場合IAM」を参照してください。

IAM 一時的な認証情報を持つ ロールは、以下の状況で役立ちます。

- 一時的なIAMユーザーアクセス許可 – IAMユーザーは、 IAMロールを引き受けて、特定のタスクに対して異なるアクセス許可を一時的に引き受けることができます。
- フェデレーションユーザーアクセス – フェデレーティッド ID に許可を割り当てるには、ロールを作成してそのロールの許可を定義します。フェデレーティッド ID が認証されると、その ID はロールに関連付けられ、ロールで定義されている許可が付与されます。フェデレーションのロールの詳細については、「 ユーザーガイド」の「サードパーティー ID プロバイダーのロールの作成IAM」を参照してください。IAM Identity Center を使用する場合は、アクセス許可セットを設定します。ID が認証後にアクセスできる内容を制御するために、IAMIdentity Center はアクセス許可セットを のロールに関連付けますIAM。アクセス許可セットの詳細については、「AWS IAM Identity Center ユーザーガイド」の「アクセス許可セット」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、別のアカウントのユーザー (信頼されたプリンシパル) がアカウントのリソースにアクセスすることを許可できます。クロスアカウントアクセスを許可する主な方法は、ロールを使用することです。ただし、一部の では AWS サービス、(ロールをプロキシとして使用する代わりに) ポリシーをリソースに直接アタッチできます。クロスアカウントアクセスの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の違い」を参照してください。
- クロスサービスアクセス — 一部の は、他の の機能 AWS サービス を使用します AWS サービス。例えば、サービスで呼び出しを行うと、そのサービスが Amazon でアプリケーションを実行EC2したり、Amazon S3 にオブジェクトを保存したりするのが一般的です。サービスでは、呼び出し元プリンシパルの許可、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) – IAM ユーザーまたはロールを使用して でアクションを実行すると AWS、プリンシパルと見なされます。ポリシーによって、プリンシパルに許可が付与されます。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、別サービスの別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、「サービス認証リファレンス」のAmazon Lightsail」を参照してください。
  - サービスロール – サービスロールは、ユーザーに代わってアクションを実行するためにサービスが引き受ける IAMロールです。IAM 管理者は、 内からサービスロールを作成、変更、削除できますIAM。詳細については、「 ユーザーガイド」の「 にアクセス許可を委任するロールの作成 AWS サービスIAM」を参照してください。
  - サービスにリンクされたロール – サービスにリンクされたロールは、 にリンクされたサービスロールの一種です AWS サービス。サービスは、ユーザーに代わってアクションを実行するロールを引き受けることができます。サービスにリンクされたロールは に表示され AWS アカウント 、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アクセス許可を表示できますが、編集することはできません。
- Amazon で実行されているアプリケーション EC2 – IAMロールを使用して、EC2インスタンスで実行され、 AWS CLI または AWS API リクエストを行うアプリケーションの一時的な認証情報を管理できます。これは、EC2インスタンス内にアクセスキーを保存するよりも望ましいです。 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、インスタンスにアタッチ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 ロールが含まれており、EC2インスタンスで実行されているプログラムが一時的な認証情報を取得できるようにします。詳細については、「 ユーザーガイド」の「 IAMロールを使用して Amazon EC2インスタンスで実行されているアプリケーションにアクセス許可を付与するIAM」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、「 ユーザーガイド」の「 (ユーザーではなく) IAMロールを作成する場合IAM」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

でアクセスを制御する AWS には、ポリシーを作成し、 AWS ID またはリソースにアタッチします。ポリシーは AWS 、アイデンティティまたはリソースに関連付けられているときにアクセス許可を定義する のオブジェクトです。 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きに、これらのポリシー AWS 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。ほとんどのポリシーはJSONドキュメント AWS として 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、「 ユーザーガイド」のJSON「ポリシーの概要IAM」を参照してください。

管理者はポリシーを使用して AWS JSON、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。

デフォルトでは、ユーザーやロールに権限はありません。必要なリソースに対してアクションを実行するアクセス許可をユーザーに付与するために、IAM管理者はIAMポリシーを作成できます。その後、管理者はIAMポリシーをロールに追加し、ユーザーはロールを引き受けることができます。

IAM ポリシーは、オペレーションの実行に使用するメソッドに関係なく、アクションのアクセス許可を定義します。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。そのポリシーを持つユーザーは、 AWS Management Console、、 AWS CLIまたは AWS からロール情報を取得できますAPI。

管理者はポリシーを使用して AWS JSON、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、アクセス許可なしで始まります。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に使用する方法に関係なく、アクションのアクセス許可を定義します。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。そのポリシー

を持つユーザーは、 AWS Management Console、、 AWS CLIまたは からロール情報を取得できます AWS API。

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のポリシーは、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アクセス許可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、および条件をコントロールします。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、「 ユーザーガイド」のIAM「ポリシーの作成IAM」を参照してください。

アイデンティティベースのポリシーは、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ています。管理ポリシーは、 内の複数のユーザー、グループ、ロールにアタッチできるスタンドアロンポリシーです AWS アカウント。管理ポリシーには、 AWS 管理ポリシーとカスタマー管理ポリシーが含まれます。管理ポリシーとインラインポリシーのどちらかを選択する方法については、「 IAM ユーザーガイド」の「管理ポリシーとインラインポリシーの選択」を参照してください。

アイデンティティベースのポリシーは、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アクセス許可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、および条件をコントロールします。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、「 ユーザーガイド」のIAM「ポリシーの作成IAM」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、リソースにアタッチする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としては、IAMロールの信頼ポリシー や Amazon S3 バケットポリシー などがあります。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コントロールできます。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、フェデレーティッドユーザー、または を含めることができます AWS サービス。

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IAMでは、 の AWS 管理ポリシーを使用できません。

リソースベースのポリシーは、リソースにアタッチする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としては、IAMロールの信頼ポリシー や Amazon S3 バケットポリシー など

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コントロールできます。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、フェデレーティッドユーザー、または を含めることができます AWS サービス。

アクセスコントロールリスト (ACLs)

アクセスコントロールリスト (ACLs) は、リソースへのアクセス許可を持つプリンシパル (アカウントメンバー、ユーザー、またはロール) を制御します。ACLs 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を使用しません。

Amazon S3、 AWS WAF、および Amazon VPCは、 をサポートするサービスの例ですACLs。の詳細についてはACLs、Amazon Simple Storage Service デベロッパーガイドの「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 (ACLs) は、リソースへのアクセス許可を持つプリンシパル (アカウントメンバー、ユーザー、またはロール) を制御します。ACLs 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を使用しません。

その他のポリシータイプ

AWS は、一般的ではない追加のポリシータイプをサポートします。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。

- アクセス許可の境界 – アクセス許可の境界は、アイデンティティベースのポリシーがIAMエンティティ (IAMユーザーまたはロール) に付与できるアクセス許可の上限を設定する高度な機能です。エンティティにアクセス許可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アクセス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、アクセス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。アクセス許可の境界の詳細については、「 IAMユーザーガイド」の「 IAMエンティティのアクセス許可の境界」を参照してください。
- サービスコントロールポリシー (SCPs) – SCPsは、 の組織または組織単位 (OU) に対する最大アクセス許可を指定するJSONポリシーです AWS Organizations。 AWS Organizations は、 AWS アカウント ビジネスが所有する複数の をグループ化して一元管理するためのサービスです。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 (SCPs) をアカウントの一部またはすべてに適用できます。は、各 を含むメンバーアカウントのエンティティのアクセス許可SCP

を制限します AWS アカウントのルートユーザー。Organizations と の詳細についてはSCPs、「AWS Organizations ユーザーガイド」の「サービスコントロールポリシー」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、「 ユーザーガイド」の「セッションポリシーIAM」を参照してください。

- アクセス許可の境界 – アクセス許可の境界は、アイデンティティベースのポリシーがIAMエンティティ (IAMユーザーまたはロール) に付与できるアクセス許可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。アクセス許可の境界の詳細については、「 IAMユーザーガイド」の「 IAMエンティティのアクセス許可の境界」を参照してください。
- サービスコントロールポリシー (SCPs) – SCPsは、 の組織または組織単位 (OU) に対する最大アクセス許可を指定するJSONポリシーです AWS Organizations。 AWS Organizations は、 AWS アカウント ビジネスが所有する複数の をグループ化して一元管理するためのサービスです。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 (SCPs) をアカウントの一部またはすべてに適用できます。は、各 AWS アカウント ルートユーザーを含むメンバーアカウントのエンティティのアクセス許可SCPを制限します。Organizations と の詳細についてはSCPs、「AWS Organizations ユーザーガイド」のSCPs「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、「 ユーザーガイド」の「セッションポリシーIAM」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与する場合にリクエストを許可するかどうか

AWS を決定する方法については、「 ユーザーガイド」の「ポリシー評価ロジックIAM」を参照してください。

トピック

- AWS Amazon Lightsail の マネージドポリシー
- Amazon Lightsail と の連携方法 IAM
- IAM ユーザーに Lightsail アクセスを付与する

AWS Amazon Lightsail の マネージドポリシー

ユーザー、グループ、ロールにアクセス許可を追加するには、自分でポリシーを記述するよりも、AWS 管理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。すぐに開始するには、 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 管理ポリシーの詳細については、「IAM ユーザーガイド」の「 AWS 管理ポリシー」を参照してください。

AWS 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 管理ポリシーのアクセス許可は変更できません。サービスは、新機能をサポートするために、 AWS マネージドポリシーに追加のアクセス許可を追加することがあります。この種の更新は、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、ロール) に影響を与えます。サービスは、新機能の起動時または新しいオペレーションが利用可能になったときに、 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アクセス許可を削除しないため、ポリシーの更新によって既存のアクセス許可が破損することはありません。

さらに、 は、複数の サービスにまたがる職務機能の マネージドポリシー AWS をサポートします。例えば、 ReadOnlyAccess AWS 管理ポリシーは、すべての AWS サービスとリソースへの読み取り専用アクセスを提供します。サービスが新機能を起動すると、 は新しいオペレーションとリソースの読み取り専用アクセス許可 AWS を追加します。ジョブ機能のポリシーの一覧および詳細については、「IAM ユーザーガイド」の「AWS のジョブ機能のマネージドポリシー」を参照してください。

AWS マネージドポリシー: LightsailExportAccess

IAM エンティティ LightsailExportAccess に をアタッチすることはできません。このポリシーは、Lightsail がユーザーに代わってアクションを実行できるように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 がインスタンスと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、Amazon Simple Storage Service (Amazon S3) から現在のアカウントレベルのパブリックアクセスブロック設定を取得できるようにするアクセス許可を付与します。

アクセス許可の詳細

このポリシーには、以下のアクセス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PublicAccessBlockの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": "*"
   }
  ]
}
```

AWS 管理ポリシーに対する Lightsail の更新

- `LightsailExportAccess` 管理ポリシーの編集

  `LightsailExportAccess` マネージドポリシーに `s3:GetAccountPublicAccessBlock` アクションが追加されました。これにより、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できます。

  2022 年 1 月 14 日

- Lightsail が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 の連携方法 IAM

IAM を使用して Lightsail へのアクセスを管理する前に、Lightsail で使用できるIAM機能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 と連携する方法の概要を把握するにはIAM、「 IAMユーザーガイド」のAWS 「 と連携IAMする 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 アイデンティティベースのポリシーでは、許可または拒否されたアクションとリソース、およびアクションが許可または拒否される条件を指定できます。Lightsail は、特定のアクション、リソース、および条件キーをサポートします。JSON ポリシーで使用するすべての要素については、「 ユーザーガイド」のIAMJSON「 ポリシー要素リファレンスIAM」を参照してください。

アクション

管理者はポリシーを使用して AWS JSON、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要素は、ポリシーで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を記述します。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オペレーションがないアクセス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権限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、アクションの前にプレフィックス を使用しますlightsail:。例えば、Lightsail CreateInstancesAPIオペレーションで Lightsail インスタンスを実行するアクセス許可を付与するには、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。

単一ステートメントに複数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 (*) を使用して複数アクションを指定できます。例えば、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確認するには、「 IAMユーザーガイド」のAmazon Lightsail で定義されるアクション」を参照してください。

リソース

管理者はポリシーを使用して AWS JSON、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。

Policy ResourceJSON要素は、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、リソースレベルの権限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、一部のAPIアクションのリソースレベルのアクセス許可をサポートしていません。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 インスタンスリソースには、次の がありますARN。

```
arn:${Partition}:lightsail:${Region}:${Account}:Instance/${InstanceId}
```

の形式の詳細についてはARNs、「Amazon リソースネーム (ARNs) 」および AWS 「サービス名前空間」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 を使用しますARN。

```
"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アクションの多くには、複数のリソースが含まれます。例えば、 は Lightsail ブロックストレージディスクをインスタンスにアAttachDiskタッチするため、IAMユーザーはディスクとイ

ンスタンスを使用するためのアクセス許可を持っている必要があります。1 つのステートメントで複数のリソースを指定するには、 をカンマARNsで区切ります。

```
"Resource": [
      "resource1",
      "resource2"
```

Lightsail リソースタイプとその のリストを確認するにはARNs、「 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。各リソースARNの を指定できるアクションについては、Amazon Lightsail」を参照してください。

条件キー

管理者はポリシーを使用して AWS JSON、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。1 つの条件キーに複数の値を指定すると、 は論理ORオペレーションを使用して条件 AWS を評価します。ステートメントの権限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ユーザー名でタグ付けされている場合にのみ、リソースにアクセスするアクセス許可をIAMユーザーに付与できます。詳細については、「 ユーザーガイド」のIAM「ポリシー要素: 変数とタグIAM」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ます。すべての AWS グローバル条件キーを確認するには、「 ユーザーガイド」のAWS 「 グローバル条件コンテキストキーIAM」を参照してください。

Lightsail にはサービス固有の条件キーはありませんが、一部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AWS 」の「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 の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」を参照してください。条件キーを使用できるアクションとリソースについては、Amazon Lightsail で定義されるアクション」を参照してください。

例

Lightsail アイデンティティベースのポリシーの例を表示するには、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ていません。

アクセスコントロールリスト (ACLs)

Lightsail はアクセスコントロールリスト () をサポートしていませんACLs。

Lightsail タグに基づく認可

Lightsail リソースにタグを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、ポリシーの条件要素でタグ情報を提供します。

> **Important**
>
> Lightsail は、一部のAPIアクションのタグに基づいた認可をサポートしていません。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を表示するには、「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、特定のアクセス許可を持つ AWS アカウント内のエンティティです。

Lightsail での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したり、 IAMロールを引き受けたり、クロスアカウントロールを引き受けたりすることができます。一時的なセキュリティ

認証情報を取得するには、 AssumeRoleや などの AWS STS APIオペレーションを呼び出しますGetFederationToken。

Lightsail は、一時的な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を使用すると、 AWS サービスが他のサービスのリソースにアクセスして、ユーザーに代わってアクションを実行できます。サービスにリンクされたロールはIAM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、サービスにリンクされたロールのアクセス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ています。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

- Lightsail で IAM ID ポリシーを使用して最小特権のアクセス許可を付与する
- IAM ポリシーを使用して特定の Lightsail リソースへのアクセスを許可する
- Amazon Lightsail のサービスにリンクされたロールを使用する
- IAM ポリシーを使用して Lightsail バケットを管理する

Lightsail で IAM ID ポリシーを使用して最小特権のアクセス許可を付与する

デフォルトでは、IAMユーザーとロールには 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。また、、 AWS Management Console AWS CLI、または を使用してタスクを実行することはできません AWS API。IAM 管理者は、必要な特定のリソースに対して特定のAPIオペレーションを実行するアクセス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。その後、管理者は、これらのアクセス許可を必要とするIAMユーザーまたはグループにこれらのポリシーをアタッチする必要があります。

これらのポリシードキュメント例を使用してIAMアイデンティティベースのJSONポリシーを作成する方法については、 ユーザーガイドのJSON「 タブでのポリシーの作成IAM」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 管理ポリシーを開始し、最小特権のアクセス許可に移行する – ユーザーとワークロードにアクセス許可を付与するには、多くの一般的なユースケースにアクセス許可を付与する AWS 管理ポリシーを使用します。これらは で使用できます AWS アカウント。ユースケースに固有の AWS カスタマー管理ポリシーを定義して、アクセス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、「 ユーザーガイド」の「 AWS 管理ポリシー」または「 ジョブ機能の 管理ポリシーIAM」を参照してください。 AWS
- 最小特権のアクセス許可を適用する – IAMポリシーでアクセス許可を設定する場合は、タスクの実行に必要なアクセス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 を使用してアクセス許可を適用する方法の詳細については、「 ユーザーガイド」の「 のポリシーとアクセス許可IAMIAM」を参照してください。
- IAM ポリシーの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、アクションと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 を使用して送信する必要があることを指定できますSSL。条件を使用して、 などの特定の を介してサービスアクションが使用される場合に AWS サービス、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 AWS CloudFormation。詳細については、「 ユーザーガイド」のIAMJSON「ポリシー要素: 条件IAM」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、安全で機能的なアクセス許可を確保する – IAM Access Analyzer は、ポリシーがポリシー言語 (JSON) とIAMベストプラクティスに準拠するように、新規および既存のIAMポリシーを検証します。IAM Access Analyzer には、安全で機能的なポリシーの作成に役立つ 100 を超えるポリシーチェックと実用的な推奨事項が用意されています。詳細については、「 ユーザーガイド」のIAM「Access Analyzer ポリシーの検証IAM」を参照してください。
- 多要素認証を要求する (MFA) – でIAMユーザーまたはルートユーザーを必要とするシナリオがある場合は AWS アカウント、セキュリティを強化MFAするために をオンにします。API オペレーションが呼び出されるMFAタイミングを要求するには、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「 で保護されたAPIアクセスの設定」を参照してください。

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「 のセキュリティのベストプラクティスIAMIAM」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へのフルアクセス許可が必要です。これらのアクセス許可により、 AWS アカウントの Lightsail リソースの詳細を一覧表示および表示できます。最小限必要なアクセス許可 (フルアクセスではない) 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、そのポリシーを持つエンティティ (IAMユーザーまたはロール) に対して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するには、エンティティに次のポリシーをアタッチします。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 のみを呼び出すユーザーには、最小限のコンソールアクセス許可を付与する必要はありません AWS API。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を許可します。

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、IAMユーザーがユーザー ID にアタッチされているインラインポリシーと管理ポリシーを表示できるようにするポリシーを作成する方法を示します。このポリシーには、コンソールで、または AWS CLI または を使用してプログラムでこのアクションを実行するアクセス許可が含まれています AWS API。

```
{
    "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制御できます。この例では、 のキータグallowと の値が作成リクエストでtrue定義されていない限り、ユーザーが新しい Lightsail リソースを作成することを制限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、「 ユーザーガイド」のIAMJSON「ポリシー要素: 条件IAM」を参照してください。

IAM ポリシーを使用して特定の Lightsail リソースへのアクセスを許可する

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 は、リソースレベルのアクセス許可をサポートしています。つまり、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または編集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使用できるタイミングを制御できます。例えば、特定の Amazon リソースネーム () を持つ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できますARN。

> **Important**
>
> Lightsail は、一部のAPIアクションのリソースレベルのアクセス許可をサポートしていません。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 アクションによって作成または変更されるリソース、およびIAMポリシーステートメントで使用できる ARNsおよび Lightsail 条件キーの詳細については、 IAM ユーザーガイドの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。

特定のインスタンスの管理を許可する

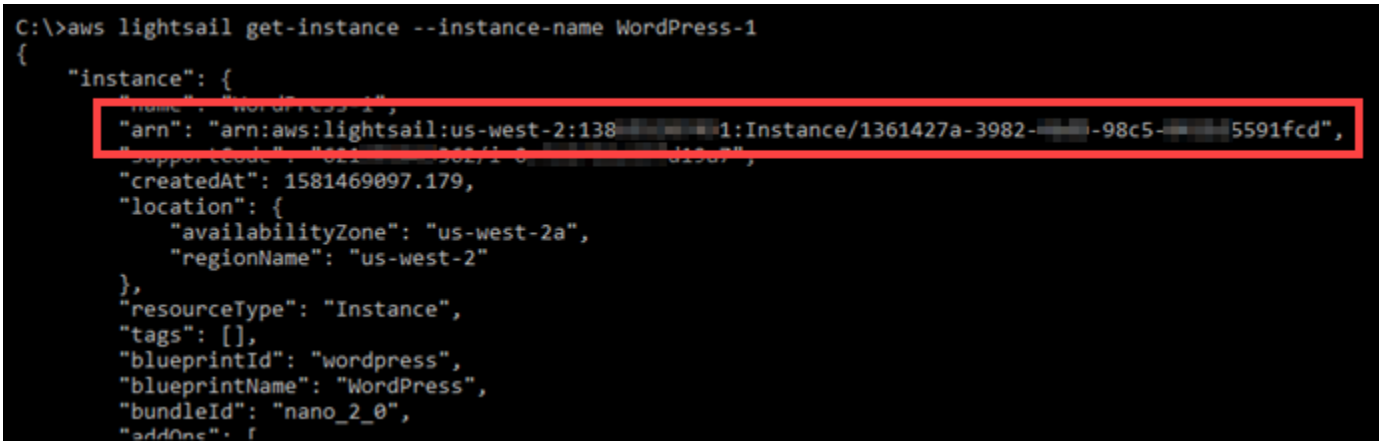
次のポリシーでは、インスタンスの再起動/開始/停止、インスタンスポートの管理、特定のインスタンスのインスタンススナップショットの作成へのアクセス権を付与します。また、Lightsail アカウントの他のインスタンス関連情報やリソースへの読み取り専用アクセスも提供します。ポリシーで、*InstanceARN* インスタンスの Amazon リソースネーム (ARN) を指定します。

```
{
    "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ARNの を取得するには、Lightsail `GetInstance` 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ARNされます。詳細については、GetInstanceAmazon Lightsail APIリファレンス」の「」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

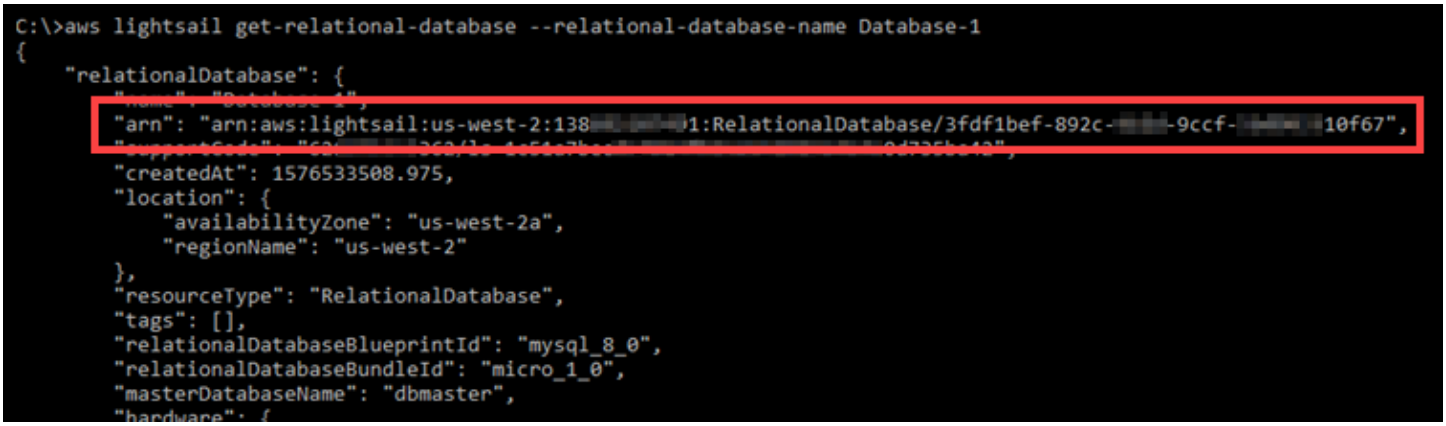
次のポリシーは、特定のデータベースの再起動/開始/停止および更新へのアクセス権を付与します。また、Lightsail アカウントの他のデータベース関連情報やリソースへの読み取り専用アクセスも提供します。ポリシーで、*DatabaseARN* データベースの Amazon リソースネーム (ARN) を指定します。

```
{
    "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ARNの を取得するには、Lightsail GetRelationalDatabase 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。次の例に示すように、データベースはそのアクションの結果に一覧表示ARNされます。詳細については、GetRelationalDatabaseAmazon Lightsail APIリファレンス」の「」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使用する

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 によって事前定義されており、Lightsail がユーザーに代わって他の AWS サービスを呼び出すために必要なすべてのアクセス許可が含まれています。

サービスにリンクされたロールを使用すると、必要なアクセス許可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。

サービスリンクロールは、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 Amazon Lightsail リソースへのアクセス許可が誤って削除されないため、Amazon Lightsail リソースが保護されます。

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 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探してください。サービスのサービスリンクロールに関するドキュメンテーションを表示するには、[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、Lightsail インスタンスとブロックストレージディスクスナップショットを Amazon Elastic Compute Cloud (Amazon EC2) にエクスポートし、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ます。

AWSServiceRoleForLightsail サービスにリンクされたロールは、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により、Amazon Lightsail は指定されたリソースに対して次のアクションを実行できます。

- アクション: すべての AWS リソース`ec2:CopySnapshot`で。
- アクション: すべての AWS リソース`ec2:DescribeSnapshots`で。
- アクション: すべての AWS リソース`ec2:CopyImage`で。
- アクション: すべての AWS リソース`ec2:DescribeImages`で。
- アクション: `cloudformation:DescribeStacks`すべての AWS AWS CloudFormation スタックで。
- アクション: すべての AWS リソース`s3:GetAccountPublicAccessBlock`で。

サービスリンクロールのアクセス許可

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 マネージドポリシーを使用して、サービスへのフルアクセスを提供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

サービスリンク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 CLI、または AWS API で Lightsail AWS AWS Management Consoleバケットを作成または更新すると、Amazon Lightsail によって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、Lightsail バケットを作成または更新すると、Amazon Lightsail によってサービスにリンクされたロールが再度作成されます。

> **Important**
>
> Amazon Lightsail がサービスにリンクされたロールを作成できるようにするには、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、 AWSServiceRoleForLightsail 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。サービスリンクロールを作成した後は、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「サービスリンクロールの編集」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

サービスリンク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、モニタリングや保守が積極的に行わ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、サービスにリンクされたロールを削除する AWSServiceRoleForLightsail前に、保留中のコピー状態の Amazon Lightsail 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。

サービスにリンクされたロールを IAM で手動削除するには

IAM コンソール、、または AWS API を使用して AWS CLI、 AWSServiceRoleForLightsail サービスにリンクされたロールを削除します。詳細については、「IAM ユーザーガイド」の「サービスリンクロールの削除」を参照してください。

Amazon Lightsail サービスにリンクされたロールでサポートされているリージョン

Amazon Lightsail は、サービスが利用可能な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ています。Lightsail が利用可能な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。

IAM ポリシーを使用して Lightsail バケットを管理する

次のポリシーは、Amazon Lightsail オブジェクトストレージサービス内の特定のバケットを管理するためのアクセス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 SDKsを介してバケットへのアクセスを許可します。ポリシーで、*#BucketName#* を管理対象のバケットの名前に置き換えます。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 コンソールでバケット管理ページのオブジェクトタブを表示すると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

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」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」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 でバケット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の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バケットのアクセスログ記録を有効にする
- Amazon Lightsail の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」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について」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 で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」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」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」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」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: Amazon Lightsail バケットへの WordPress インスタンスの接続
- 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 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」を参照してください。

IAM ユーザーに Lightsail アクセスを付与する

AWS アカウントのルートユーザー 、または管理者権限を持つ AWS Identity and Access Management (IAM) ユーザーとして、 AWS アカウントに 1 人以上の IAM ユーザーを作成できます。これらのユーザーは、 が提供する のサービスへのさまざまなレベルのアクセスで設定できます AWS。

Amazon Lightsail の場合、Lightsail サービスにのみアクセスできる IAM ユーザーを作成することをお勧めします。これは、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を必要とするが、 によって提供される他のサービスへのアクセスを必要としないユーザーがチームに参加するときに行います AWS。これを設定するには、まず Lightsail へのアクセスを許可する IAM ポリシーを作成し、次に IAM グループを作成し、そのポリシーをグループにアタッチする必要があります。次に、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、ユーザーは Lightsail にアクセスできます。

誰かがチームを離れても、Lightsail アクセスグループからユーザーを削除して、例えば、そのユーザーがチームを離れても会社で作業している場合は、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。

> **Warning**
>
> このシナリオでは、プログラムによるアクセスと長期的な認証情報を持つ IAM ユーザーが必要です。これはセキュリティ上のリスクをもたらします。このリスクを軽減するために、これらのユーザーにはタスクの実行に必要な権限のみを付与し、不要になったユーザーを削除することをお勧めします。アクセスキーは、必要に応じて更新できます。詳細については、「IAM ユーザーガイド」の「アクセスキーの更新」を参照してください。

目次

Lightsail アクセス用の IAM ポリシーを作成する

Lightsail アクセス用の IAM ポリシーを作成するには、次の手順に従います。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. [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": "*"
        }
    ]
}
```

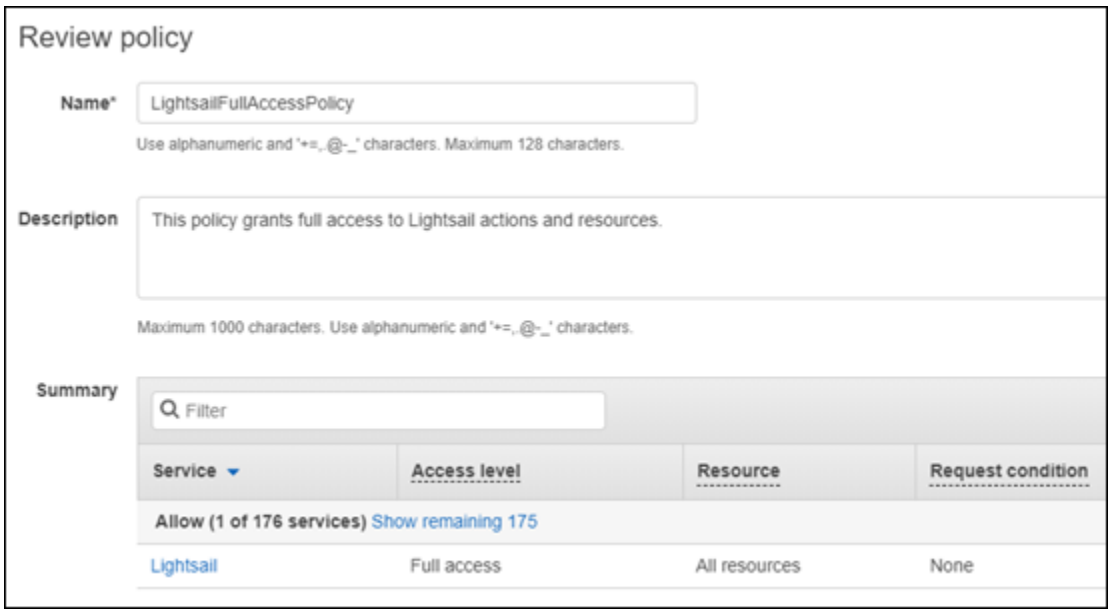
結果は次の例のようになります。

これにより、すべての Lightsail アクションとリソースへのアクセスが許可されます。VPC ピアリングの有効化 AWS、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、 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ていない追加のアクセス許可が必要です。詳細については、以下のガイドを参照してください。

- Amazon Lightsail の外部にある AWS リソースと連携するように Amazon VPC ピアリングを設定する Amazon Lightsail
- Amazon Lightsail スナップショットを Amazon EC2 にエクスポートする
- Lightsail でエクスポートされたスナップショットからの Amazon EC2 インスタンスの作成

付与できるアクション固有およびリソース固有のアクセス許可の例については、Amazon Lightsail リソースレベルのアクセス許可ポリシーの例」を参照してください。

6. [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 アクセス用の 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 アクセス用の IAM グループを作成し、このガイドの前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、次の手順に従います。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. 「ポリシーのアタッチ」ページで、このガイドの前半で作成した Lightsail ポリシーを検索します。例えば、「」です`LightsailFullAccessPolicy`。
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 アクションとリソースにアクセスできます。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、そのユーザーを Lightsail アクセスグループに追加する

以下の手順に従って IAM ユーザーを作成し、そのユーザーを Lightsail アクセスグループに追加します。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. [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 「アクセスタイプを選択」セクションで、次のオプションから選択します。

   a. プログラムによるアクセスを選択して、 AWS API、CLI、SDK、および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ます。これらは Lightsail のアクションとリソースに使用できます。詳細については、「Lightsail で動作する AWS CLI ように 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アクセスを選択して、ユーザーが AWS マネジメントコンソール、つまり Lightsail コンソールにサインインできるようにするパスワードを有効にします。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。

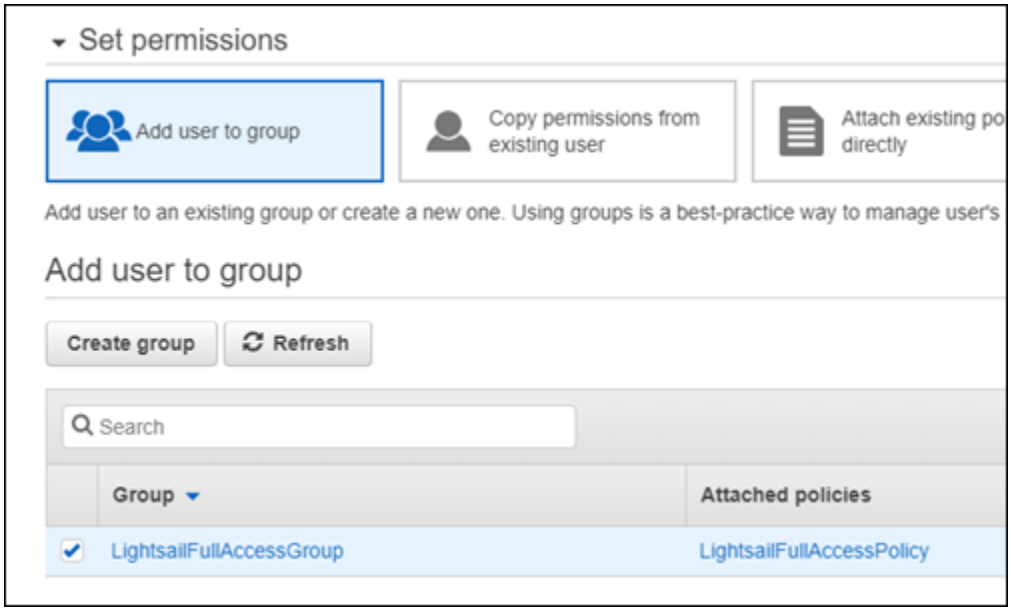
      i. [自動生成パスワード] を選択すると IAM がパスワードを生成し、または、[カスタムパスワード] を選択すると独自のパスワードを入力できます。

      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**
   >
   > プログラムによるアクセスオプションのみを選択した場合、ユーザーは AWS コンソールと Lightsail コンソールにサインインできません。

5. [Next: Permissions] (次のステップ: 許可) を選択します。

6. ページの「アクセス許可の設定」セクションで「ユーザーをグループ に追加」を選択し、このガイドの前半で作成した「Lightsail アクセスグループ」を選択します。例えば、「」ですLightsailFullAccessGroup。



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. [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、ユーザーは Lightsail にアクセスできます。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。

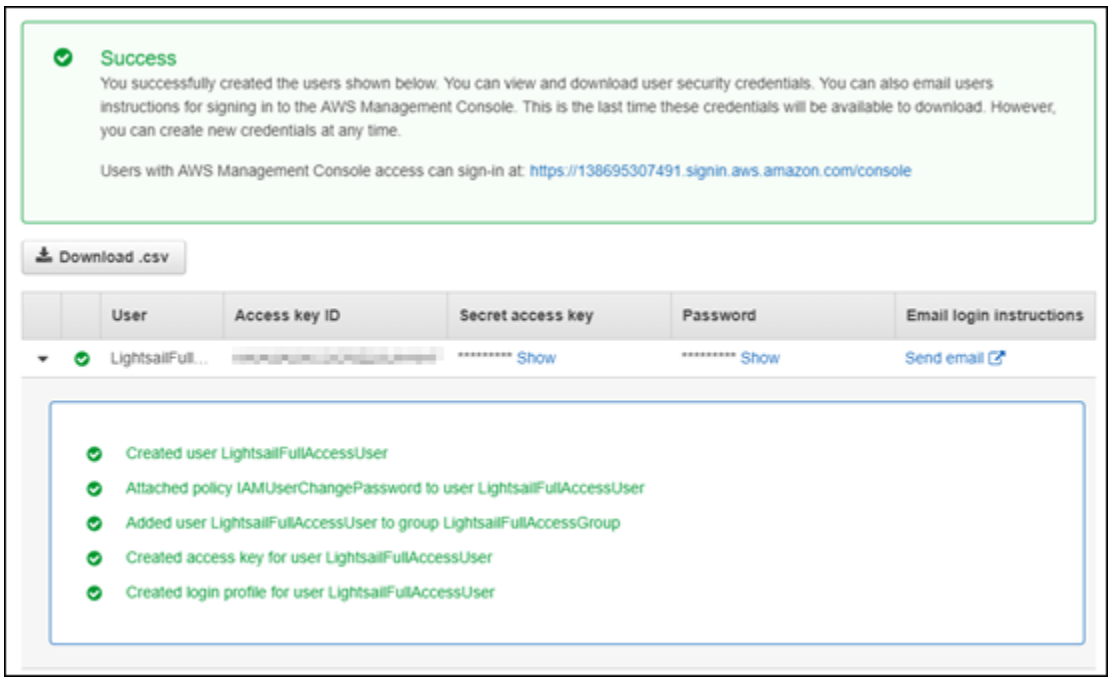
    a. .csv のダウンロード を選択して、アカウントのユーザー名、パスワード、アクセスキー ID、シークレットアクセスキー、 AWS コンソールログインリンクを含むファイルをダウンロードします。

    b. シークレットアクセスキーの下に表示 を選択して、Lightsail にプログラムでアクセスするために使用できるアクセスキーを表示します (AWS API、CLI、SDK、およびその他の開発ツールを使用) 。

> **Important**
>
> これは、シークレットアクセスキーを表示またはダウンロードする唯一の機会であり、ユーザーが AWS API を使用する前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 メールを送信を選択して、Lightsail にアクセスできるようになったことを知らせる E メールをユーザーに送信します。



更新管理により Lightsail インスタンスとコンテナを安全に保つ

Amazon Web Services (AWS) 、Amazon Lightsail 、およびサードパーティーアプリケーションベンダーは、Lightsail 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) を定期的に更新してパッチを適用します。 AWS Lightsail は、インスタンスの作成後にオペレーティングシステムやアプリケーションを更新したりパッチを適用したりしません。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とソフトウェアを更新またはパッチ適用しません。し

たがって、Amazon Lightsail インスタンスとコンテナサービスでオペレーティングシステムとアプリケーションを定期的に更新、パッチ適用、保護することをお勧めします。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

次の 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されています。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
- 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 リソースのコンプライアンスを検証する

AWS では、コンプライアンスに役立つ以下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、セキュリティとコンプライアンスに重点を置いたベースライン環境を にデプロイする手順について説明します AWS。
- AWS コンプライアンスリソース – このワークブックとガイドのコレクションは、お客様の業界や地域に適用される場合があります。
- 「 デベロッパーガイド」の「 ルールによるリソースの評価」 – この AWS Config サービスは、リソース設定が社内プラクティス、業界ガイドライン、および規制にどの程度準拠しているかを評価します。 AWS Config
- AWS Security Hub – この AWS サービスは、 内のセキュリティ状態を包括的に把握 AWS し、セキュリティ業界標準とベストプラクティスへの準拠を確認するのに役立ちます。

Lightsail リソースメトリクスのモニタリング

Amazon Lightsail のインスタンス、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、バケットのパフォーマンスをモニタリングするには、メトリクスデータをチェックして収集します。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をお勧めします (5 USD/月プランではなく 7 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、リソースが指定された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

以下の用語と概念は、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。例えば、Lightsail が提供するインスタンスメトリクスには、CPU 使用率 (CPUUtilization) 、受信ネットワークトラフィック ()、送信ネットワークトラフィック (NetworkIn) などがありますNetworkOut。Lightsail で利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 で利用可能なメトリクス」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。これらの期間よりも長いメトリクスの可用性が必要な場合は、Lightsail API、 AWS Command Line Interface （ AWS CLI ）、および SDKs を使用して、オフラインまたは異なるストレージのデータポイントを取得できます。

詳細については、GetRelationalDatabaseMetricDataLightsail API リファレンスGetInstanceMetricDataのGetBucketMetricDataGetLoadBalancerMetricData「」、「」、GetDistributionMetricData「」、「」、「」、「」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。

使用可能なメトリクス統計のリストについては、Lightsail API リファレンスの「 の統計 GetInstanceMetricData」、 GetLoadBalancerMetricData「」の統計「」、「」の統計「」、「「」の統計 GetDistributionMetricData「」、「「」の統計 GetRelationalDatabaseMetricData」を参照してください。 GetBucketMetricData

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。単位の完全なリストについては、Lightsail API リファレンスの「 の単位 GetInstanceMetricData」、「 GetLoadBalancerMetricDataの単位」、「 GetDistributionMetricDataの単位」、「 の単位 GetRelationalDatabaseMetricData」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 はメトリクスデータを 1 分または 5 分単位でレポートするため、60 秒と 300 秒の倍数で期間を指定する必要があります。開始時刻と終了時刻に指定する値は、Lightsail が返す期間の数を決定します。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。

Lightsail アラームでは、期間も重要です。Lightsail は 5 分ごとにアラームのデータポイントを評価し、アラームの各データポイントは 5 分間の集計データを表します。特定のメトリクスをモニタリングするアラームを作成すると、そのメトリクスを指定したしきい値と比較するように Lightsail に要求します。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。

アラーム

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 E メール

アドレスに送信される E 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にすることができます。詳細については、「アラーム」を参照してください。

Lightsail で利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」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。インスタンスにフルプロセッサコアが割り当てられていない場合、オペレーティングシステムのツールの割合は Lightsail よりも低くなります。

  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、持続可能で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。インスタンスバーストキャパシティの詳細については、Amazon Lightsail」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」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」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」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」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」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。

ヘルスメトリクスを使用して Lightsail リソースをモニタリングする

さまざまな期間にわたって、次の Amazon Lightsail 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」を参照してください。

- CPU 使用率 (**CPUUtilization**) — インスタンスで現在使用されている割り当て済みコンピューティングユニットの割合。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。インスタンスにフルプロセッサコアが割り当てられていない場合、オペレーティングシステムのツールは Lightsail よりも低い割合で表示されることがあります。

  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、持続可能で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「使用率の持続可能ゾーンとバースト可能ゾーン」を参照してください。

- バーストキャパシティ分 (**BurstCapacityTime**) と割合 (**BurstCapacityPercentage**) — バーストキャパシティ分は、インスタンスが 100% のCPU使用率でバーストするのに使用できる時間を表します。バーストキャパシティの割合は、インスタンスで使用可能な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。バーストキャパシティの分は、インスタンスが 100% のCPU使用率で動作している場合にのみ、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。
- ディスクキューの深さ (**DiskQueueDepth**) — ディスクへのアクセスを待っている未処理 IOs (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」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 リクエストと リクエストの両方について、ディストリビューションが受信したビューワーHTTPHTTPSリクエストの合計数。

- アップロードされたバイト数 — POSTおよび 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、、HEADおよび OPTIONSリクエストについてビューワーがダウンロードしたバイト数。
- 合計エラー率 — レスポンスHTTPの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 エラー率 — レスポンスHTTPの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 エラー率 — レスポンスHTTPの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 HTTP2XX (**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 レスポンスコードの数。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 レスポンスコードの数。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 HTTP4XX (**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 レスポンスコードの数。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 レスポンスコードの数。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)。
- リクエスト数 (**RequestCount**) — で処理されたリクエストの数IPv4。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**) — ロードバランサーによって生成されたTLSエラーが原因でロードバランサーとのセッションを確立しなかった、クライアントによって開始された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すべての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パーセンテージ。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 リソースのメトリクス通知を設定する

Lightsail を設定して、インスタンス、データベース、ロードバランサー、または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いずれかのメトリクスが指定されたしきい値を超えたときに通知を受け取ることができます。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。

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。

アラームがトリガーされると、Lightsail コンソールに通知バナーが表示されます。E メールと SMS テキストメッセージで通知を受けるには、リソースをモニタリング AWS リージョン する各 で、E 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、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。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、E メールと携帯電話を Lightsail から削除します。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。

メトリクスによる Lightsail インスタンスのパフォーマンスのモニタリング

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メトリクスタブに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 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

- Lightsail で利用可能なインスタンスメトリクス
- 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 コンソールでインスタンスメトリクスを表示する
- インスタンスメトリクスの表示後の次のステップ

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。インスタンスにフルプロセッサコアが割り当てられていない場合、オペレーティングシステムのツールの割合は Lightsail よりも低くなります。

  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、持続可能で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

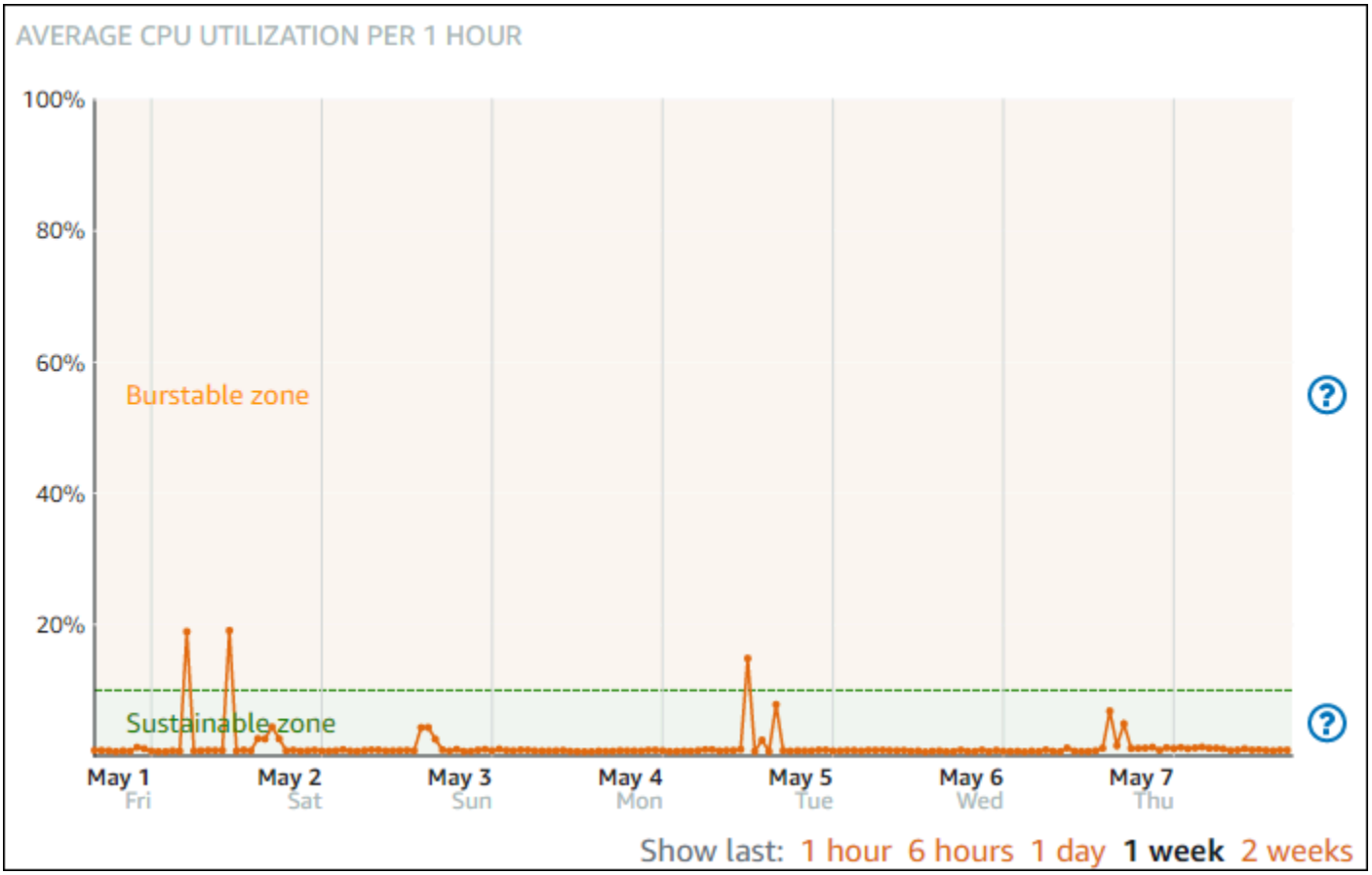
メトリクスは 0 (合格) または 1 (失敗) となります。このメトリクスは、1 分間の頻度で利用できます。

- 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、ベースラインの CPU パフォーマンスを提供するバーストインスタンスを使用しますが、必要に応じてベースラインを超える追加の 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。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 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。

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もできます (5 USD/月プランではなく 12 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、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、持続可能で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。
- アラームがトリガーされると、Lightsail コンソールに通知バナーが表示されます。E メールと SMS テキストメッセージで通知を受け取るには、リソースをモニタリングする各 AWS リージョ

ン で、E 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、E メールと携帯電話を Lightsail から削除します。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、インスタンス、データベース、ロードバランサー、および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単一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、または 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 5 分ごとにアラームのデータポイントを評価し、各データポイントは 5 分間の集計データを表します。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。アラームの状態が 状態から OK状態に変わると、通知バナーは常に Lightsail コンソールに表示されますALARM。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。

アラームの詳細設定内で、Lightsail が欠落しているメトリクスデータをどのように処理するかを選択できます。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。

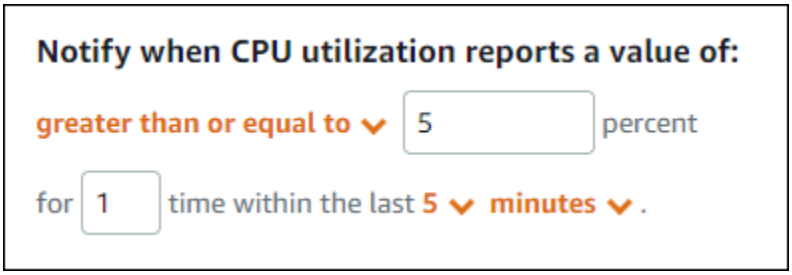
  たとえば、greater than 比較演算子を選択した場合、メトリクスが指定したしきい値を超えると、アラームは ALARM 状態になります。less than 比較演算子を選択した場合、メトリクスが指定したしきい値を下回ると、アラームは ALARM 状態になります。
- OK: — メトリクスの値が、定義されたしきい値の範囲内にあります。

  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。

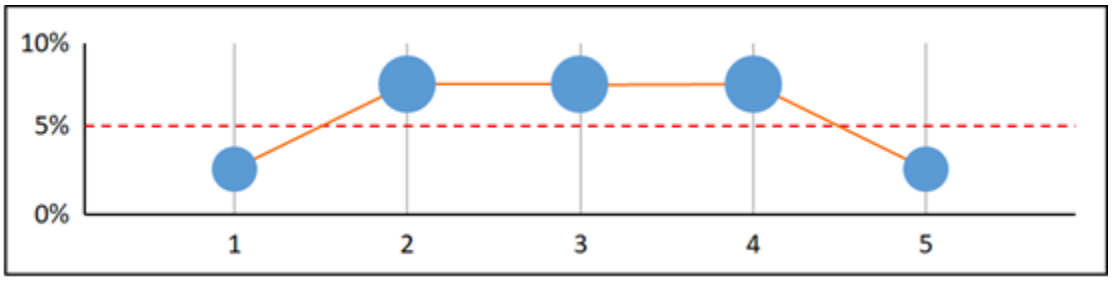
アラームの例

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ているデータポイントの処理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 を指定して、欠落しているデータポイントを次のいずれかとして処理できます。

- 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関係なく、アラームが状態を変更するかどうかを評価すると、Lightsail は評価期間 で指定された数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、次のようになります。

- 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。
- 評価範囲内の一部のデータポイントが欠落しているが、収集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、Lightsail は正常に収集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状態を評価します。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。
- 評価範囲内の一部のデータポイントが欠落していて、収集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 の数より少ない場合、Lightsail は欠落データの処理方法に指定した結果で欠落データポイントを入力し、アラームを評価します。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、欠落しているデータポイントをできるだけ少ない回数だけ使用します。

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。

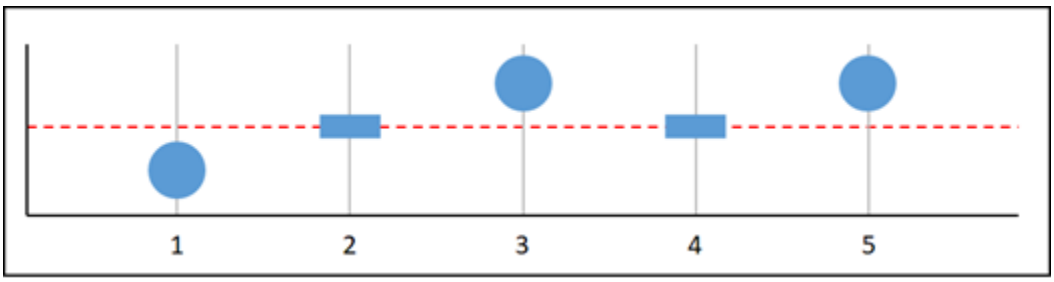
> **Note**
>
> この動作の特定のケースは、Lightsail アラームが、メトリクスのフローが停止した後、一定期間、最後のデータポイントのセットを繰り返し再評価する可能性があることです。この再評価により、メトリクスのストリームが停止する直前にアラームの状態が変更されていた場合に、アクションが再実行される可能性があります。この動作を軽減するには、より短い期間を使用します。

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の取得を試みます。

> **Note**
>
> アラームの作成直後にデータポイントが欠落し、アラームを作成する前にメトリクスが Lightsail に報告されていた場合、Lightsail はアラームの評価時にアラームが作成される前にから最新のデータポイントを取得します。

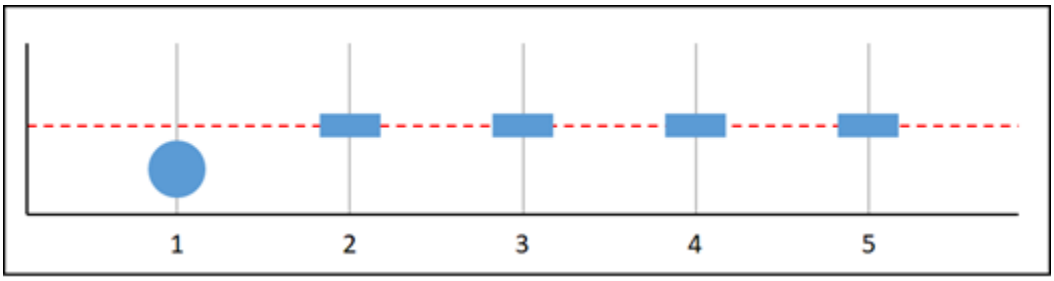
グラフ A



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

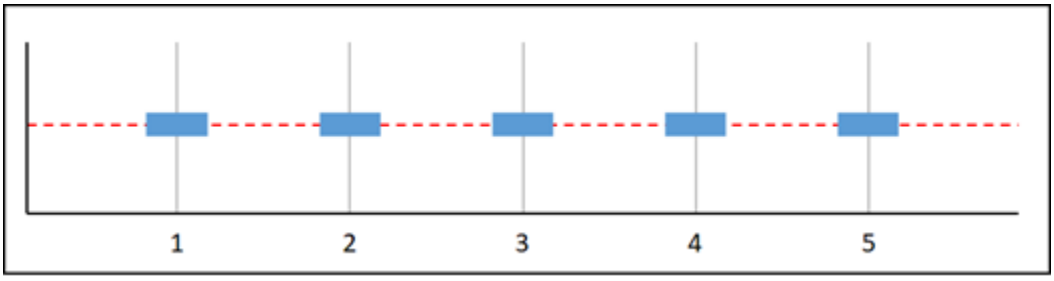


前述のグラフ化メトリクスでは、データポイント 1 がしきい値内にあり、データポイント 2~5 が欠落しています。評価範囲内にデータポイントが 1 つしかないので、このメトリクスには 2 つの欠落データポイントがあります。欠落しているデータポイントを次のように扱うようにアラームを設定した場合:

- 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

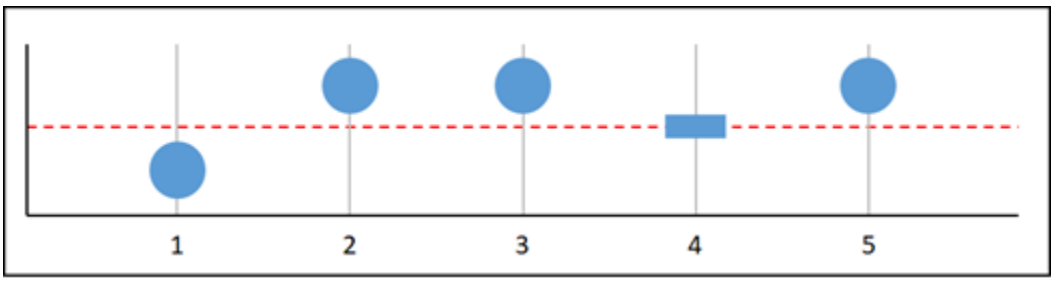


前述のグラフ化メトリクスでは、すべてのデータポイントが欠落しています。評価範囲内のすべてのデータポイントが欠落している場合、このメトリクスには 3 つの欠落データポイントがあります。欠落しているデータポイントを次のように扱うようにアラームを設定した場合:

- 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

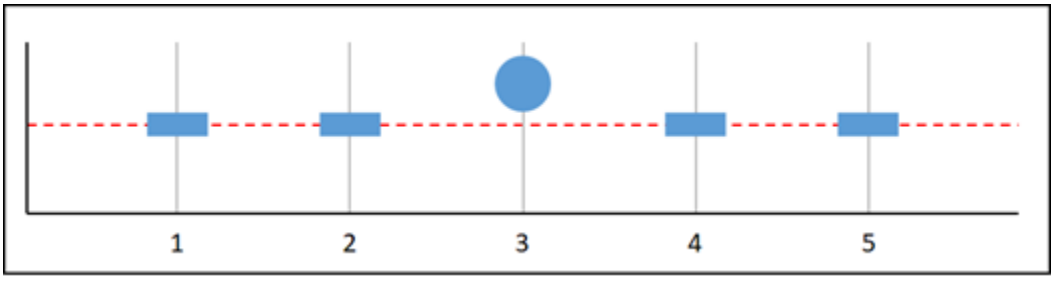


前述のグラフ化メトリクスでは、データポイント 1 がしきい値内、データポイント 2 がしきい値を超過し、データポイント 3 がしきい値を超過し、データポイント 4 が欠落し、データポイント 5 がしきい値を超過しています。評価範囲内に有効なデータポイントが 4 つあるので、このメトリクスの欠落しているデータポイントはゼロになります。欠落しているデータポイントを次のように扱うようにアラームを設定した場合:

- 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



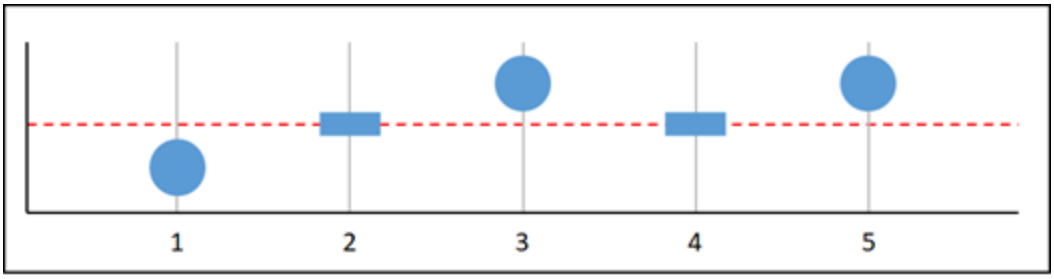
前述のグラフ化メトリクスでは、データポイント 1 と 2 が欠落し、データポイント 3 がしきい値を超過し、データポイント 4 と 5 が欠落しています。評価範囲内にデータポイントが 1 つしかないの

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。

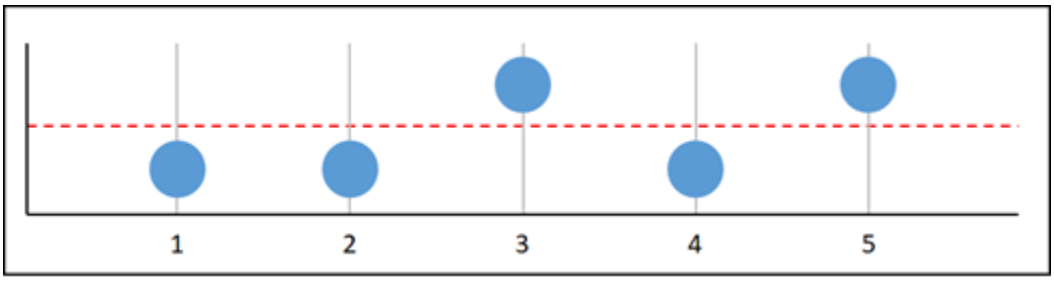
グラフ F



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。

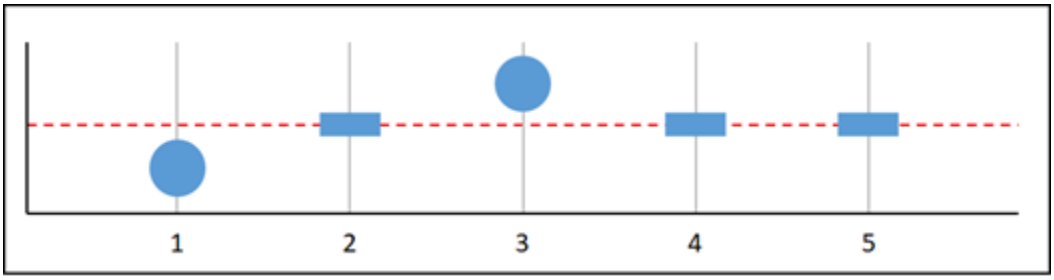
グラフ G



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。

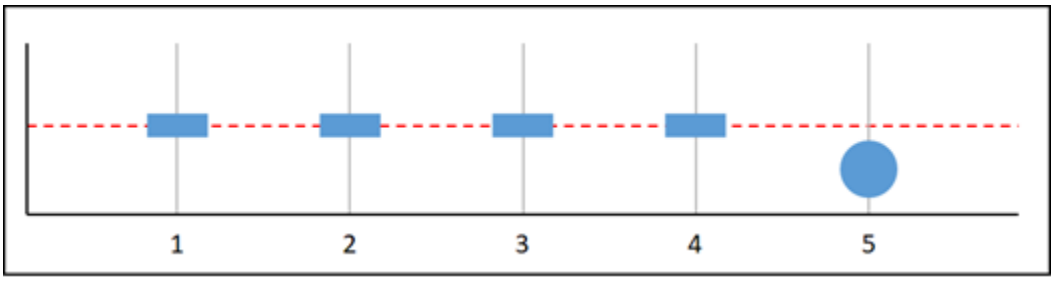
グラフ H



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。

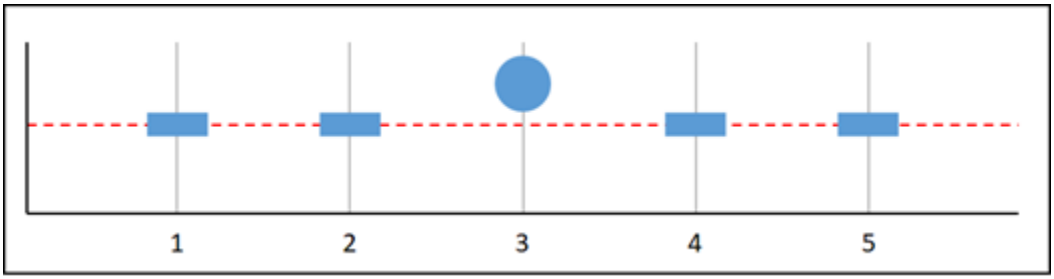
グラフ I



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

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

1 つ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

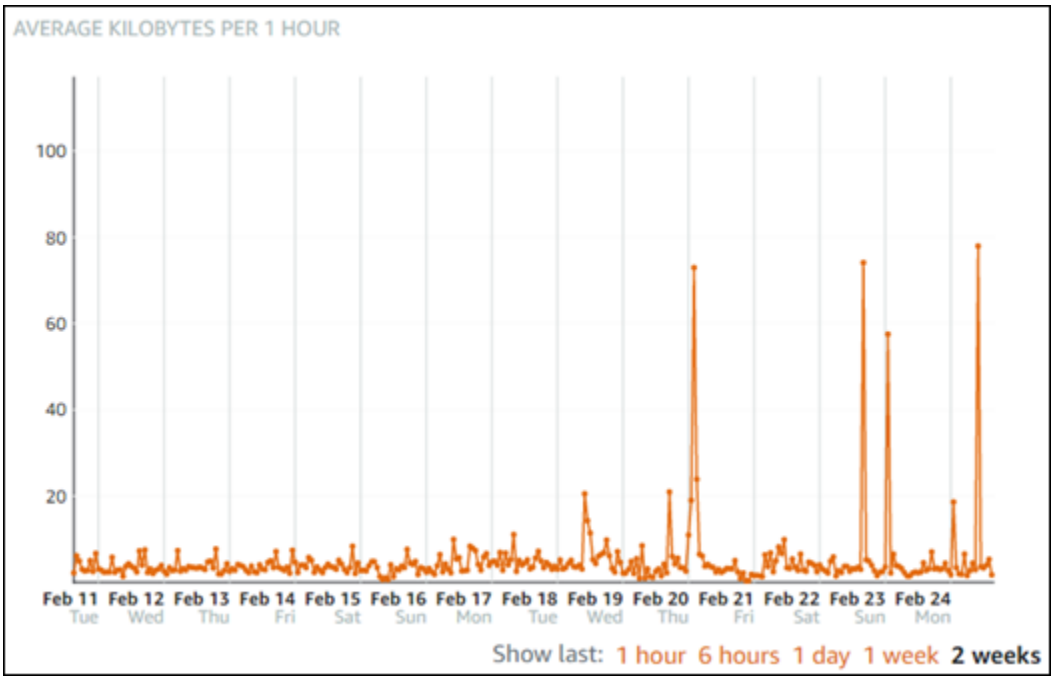
アラームには、以下の制限が適用されます。

- 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。

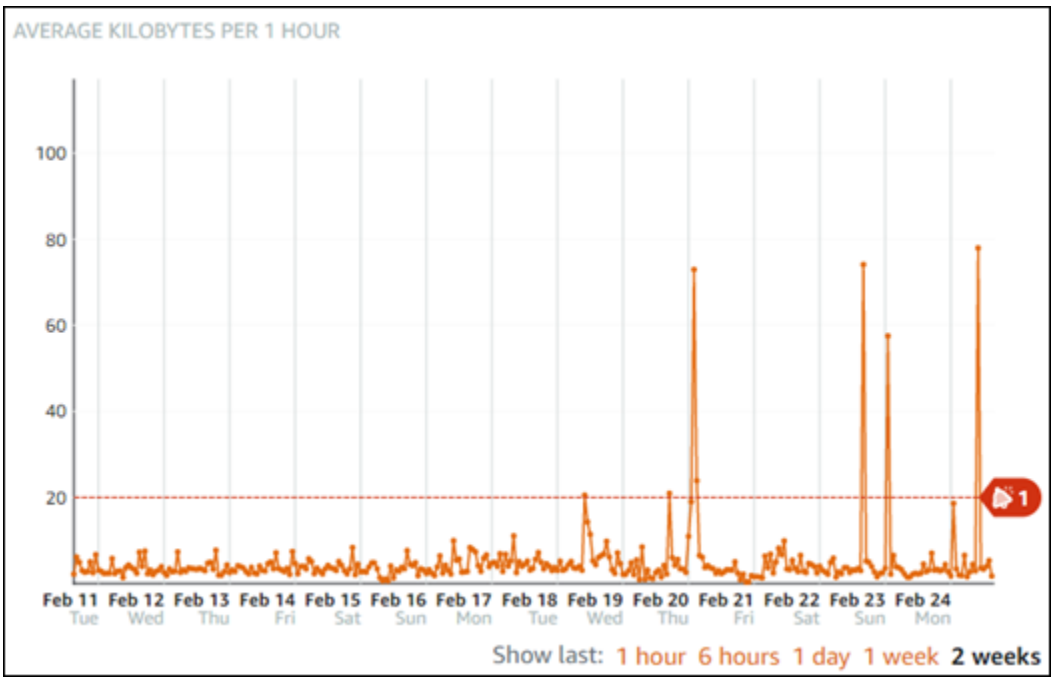
- OK アラーム通知をテストできるのは、メールや SMS テキストメッセージで通知するようにアラームを設定した場合だけです。
- アラームの状態が INSUFFICIENT_DATA に変わったときに通知するようにアラームを設定できるのは、メールや SMS テキストメッセージで通知するようにアラームを設定した場合と、欠落しているデータポイントに対して [欠落しているデータを評価しない] オプションを選択した場合だけです。
- 通知をテストできるのは、アラームが OK 状態の場合だけです。

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事前に入力されます。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、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 ホームページで、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、SMS テキストメッセージをすべての国/リージョンに送信することはできません。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。

    - アラームが欠落データをどのように処理するかを選択します。以下のオプションが利用できます。
        - 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、次のステップを実行します。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、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。通知をテストすることを選択した場合のみ、ALARMLightsail コンソールに通知バナーが表示されます。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、E メールと携帯電話を Lightsail から削除します。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を削除または無効化する

Amazon Lightsail アラームを削除して、アラームによってモニタリングされ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)]、[Databases (データベース)]、または [Networking (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. [削除] を選択します。
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 コンソールを使用したメトリクスアラームの無効化と有効化

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)]、[Databases (データベース)]、または [Networking (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 バケットのパフォーマンスと使用状況のモニタリング

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ら、バケット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、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 ホームページで、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

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」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」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 でバケット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の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バケットのアクセスログ記録を有効にする
   - Amazon Lightsail の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」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について」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 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」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」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」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」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: Amazon Lightsail バケットへの WordPress インスタンスの接続
- 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 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」を参照してください。

トピック

- メトリクスアラームを使用して Lightsail バケットストレージをモニタリングする

メトリクスアラームを使用して Lightsail バケットストレージをモニタリングする

1 つ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 コンソールに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事前に入力されます。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 ホームページで、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、次のステップを実行します。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、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。通知をテストすることを選択した場合のみ、ALARMLightsail コンソールに通知バナーが表示されます。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、E メールと携帯電話を Lightsail から削除します。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 コンテナサービスのリソース使用率のモニタリング

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 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 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 データベースのパフォーマンスメトリクスのモニタリング

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、リソースが指定されたしきい値を超過し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、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。

- アラームがトリガーされると、Lightsail コンソールに通知バナーが表示されます。E メールと SMS テキストメッセージで通知を受けるには、リソースをモニタリング AWS リージョン する各で、E 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、E メールと携帯電話を Lightsail から削除します。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

- メトリクスアラームで Lightsail データベースの状態をモニタリングする

メトリクスアラームで Lightsail データベースの状態をモニタリングする

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

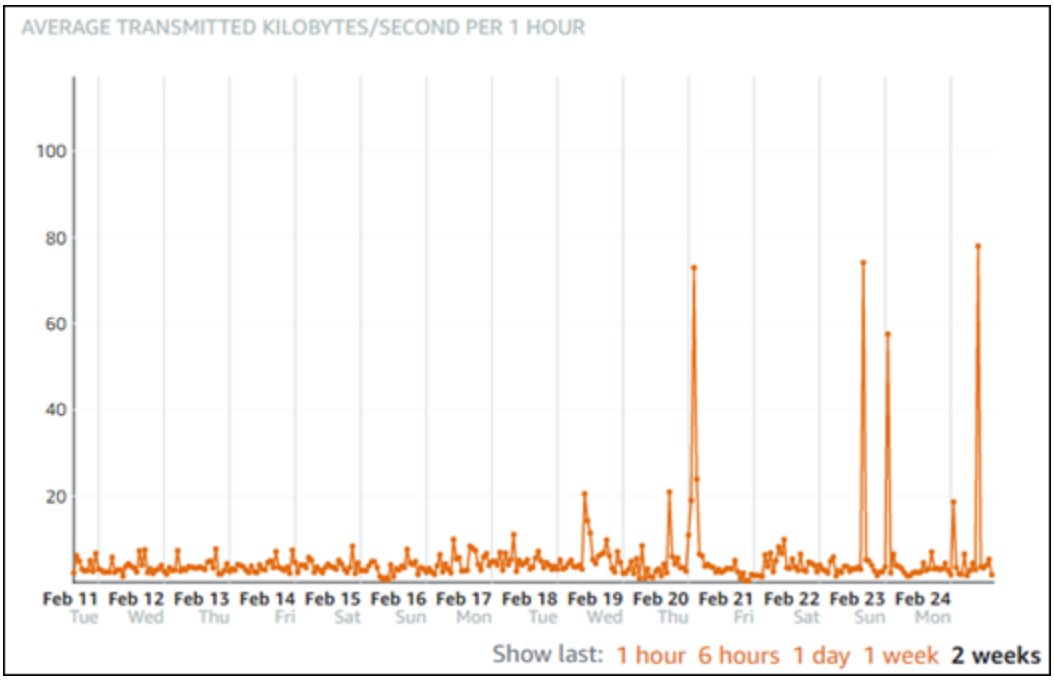
アラームには、以下の制限が適用されます。

- 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。

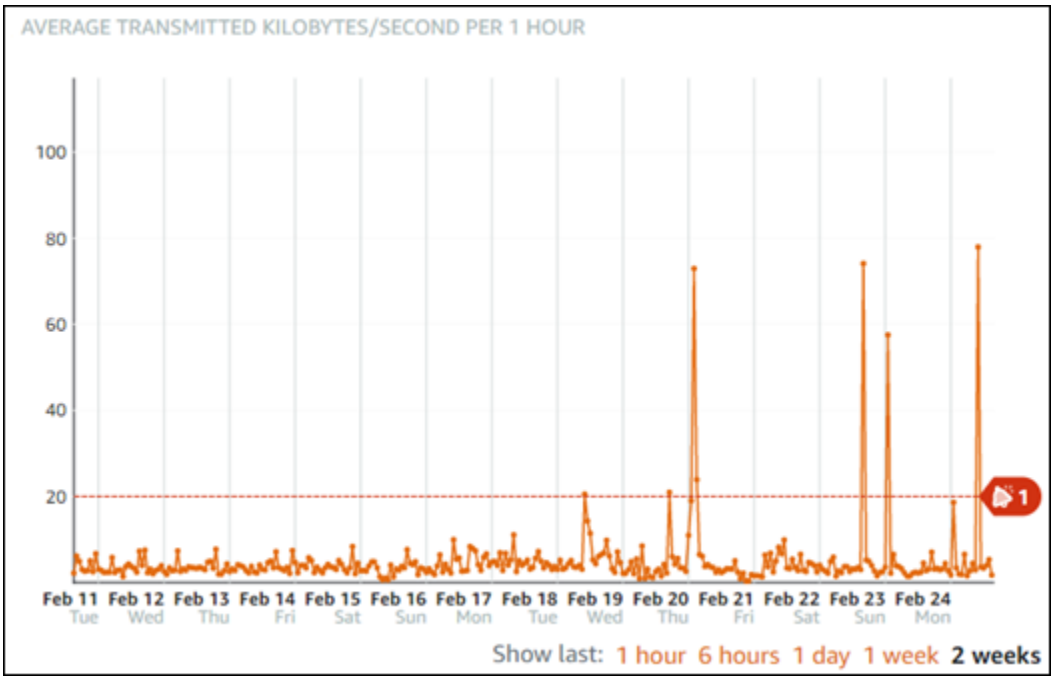
データベースアラームの設定に関するベストプラクティス

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事前に入力されます。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、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、SMS テキストメッセージをすべての国/リージョンに送信することはできません。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、次のステップを実行します。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、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。通知をテストすることを選択した場合のみ、ALARMLightsail コンソールに通知バナーが表示されます。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、E メールと携帯電話を Lightsail から削除します。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パフォーマンスメトリクスのモニタリング

Amazon Lightsail でディストリビューションを作成したら、ディストリビューション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、「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。

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

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 [Networking] (ネットワーク) 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。
- アラームがトリガーされると、Lightsail コンソールに通知バナーが表示されます。E メールと SMS テキストメッセージで通知を受けるには、リソースをモニタリング AWS リージョン する各で、E 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、E メールと携帯電話を Lightsail から削除します。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

- メトリクスアラームで Lightsail ディストリビューションの状態をモニタリングする

メトリクスアラームで Lightsail ディストリビューションの状態をモニタリングする

単一のディストリビューション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

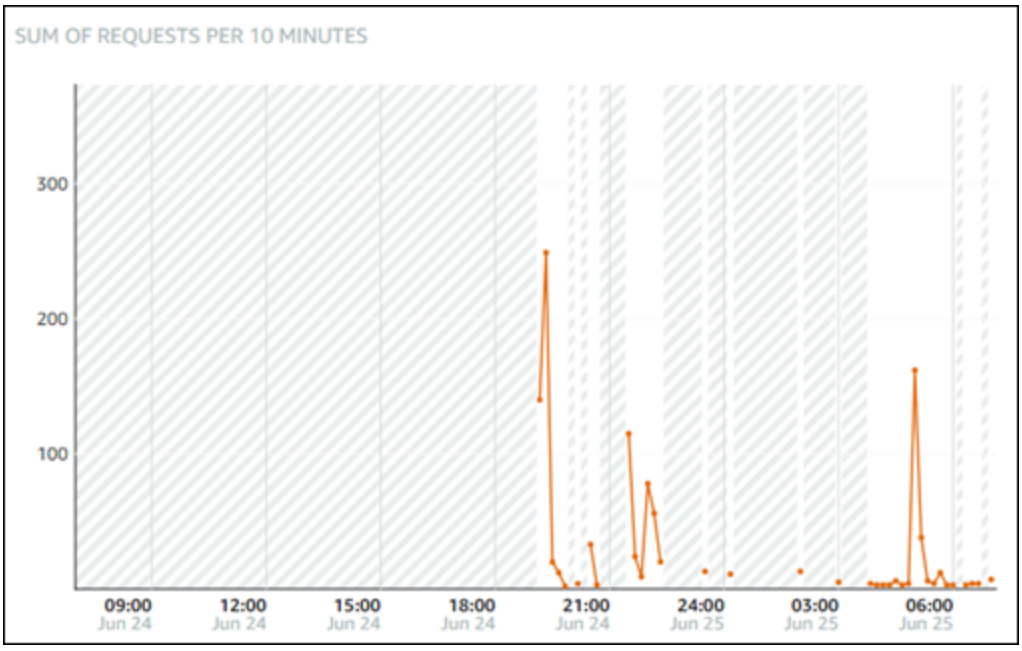
ディストリビューションアラームの制限

アラームには、以下の制限が適用されます。

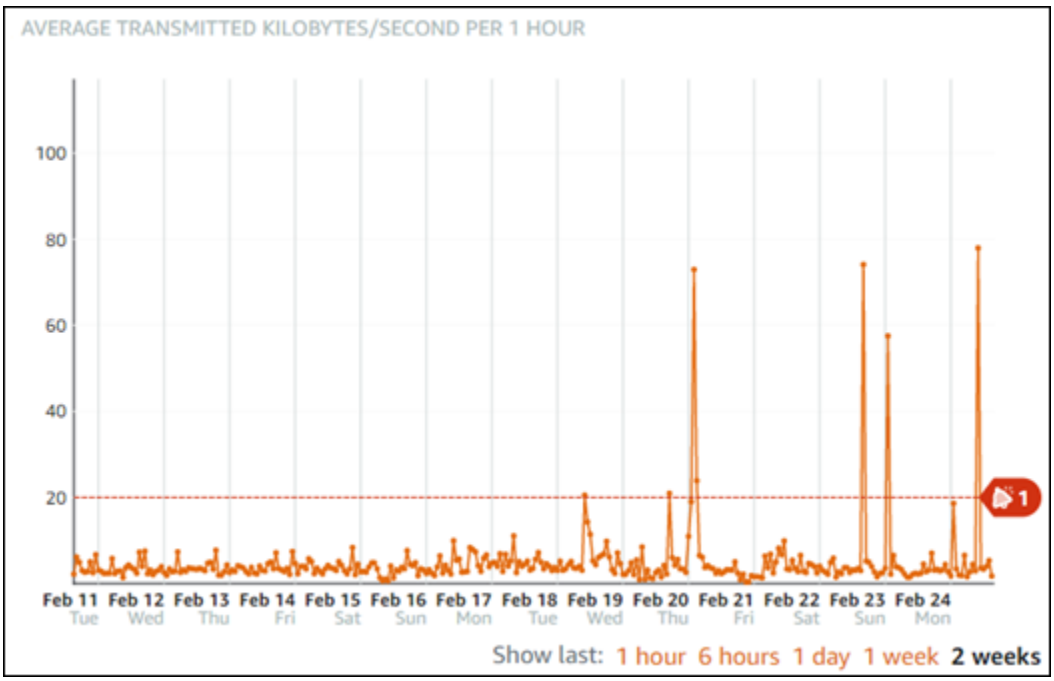
- メトリクスごとに 2 つのアラームを設定できます。
- アラームは 5 分間隔で評価され、アラームの各データポイントは、集計されたメトリクスデータの 5 分間隔を表します。
- アラームの状態が OK に変わったときに通知するようにアラームを設定できるのは、メールや SMS テキストメッセージで通知するようにアラームを設定した場合だけです。
- OK アラーム通知をテストできるのは、メールや SMS テキストメッセージで通知するようにアラームを設定した場合だけです。
- アラームの状態が `INSUFFICIENT_DATA` に変わったときに通知するようにアラームを設定できるのは、メールや 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事前に入力されます。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、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 [Networking] (ネットワーク) 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、SMS テキストメッセージをすべての国/リージョンに送信することはできません。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、次のステップを実行します。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 [Networking] (ネットワーク) 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。通知をテストすることを選択した場合のみ、ALARMLightsail コンソールに通知バナーが表示されます。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、E メールと携帯電話を Lightsail から削除します。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のモニタリング

Amazon Lightsail でロードバランサーを作成し、インスタンスをアタッチすると、ロードバランサー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、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、リソースが指定されたしきい値を超過し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 [Networking] (ネットワーク) 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。
- アラームがトリガーされると、Lightsail コンソールに通知バナーが表示されます。E メールと SMS テキストメッセージで通知を受け取るには、リソースをモニタリングする各 AWS リージョン で、E 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、E メールと携帯電話を Lightsail から削除します。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

- アラームによる Lightsail ロードバランサーメトリクスのモニタリング

アラームによる Lightsail ロードバランサーメトリクスのモニタリング

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。

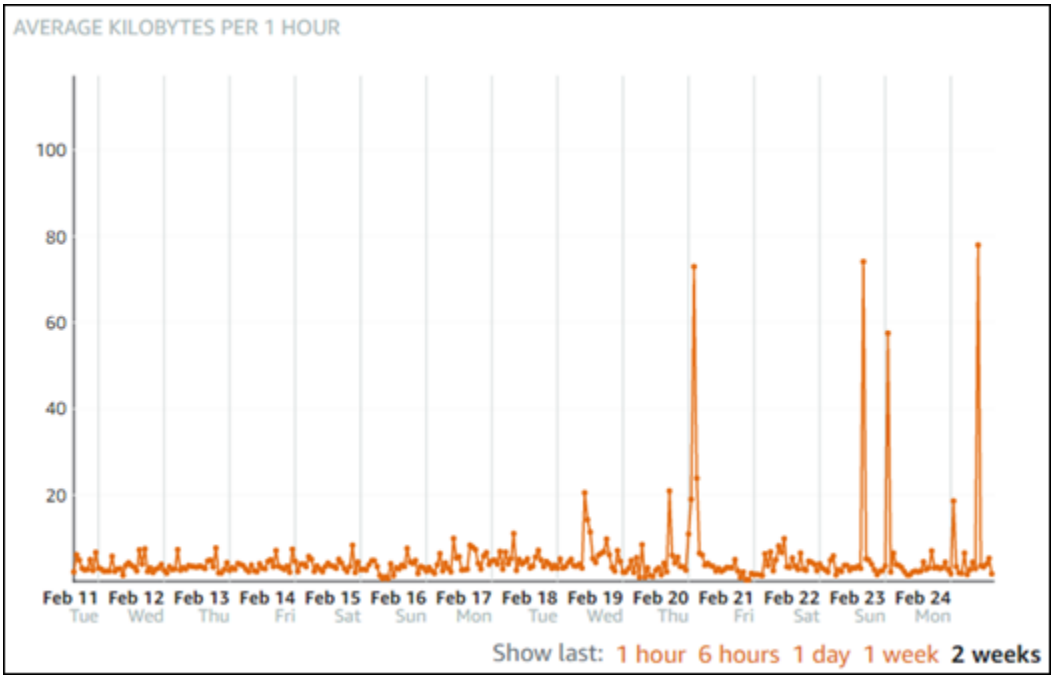
ロードバランサーアラームの設定に関するベストプラクティス

アラームには、以下の制限が適用されます。

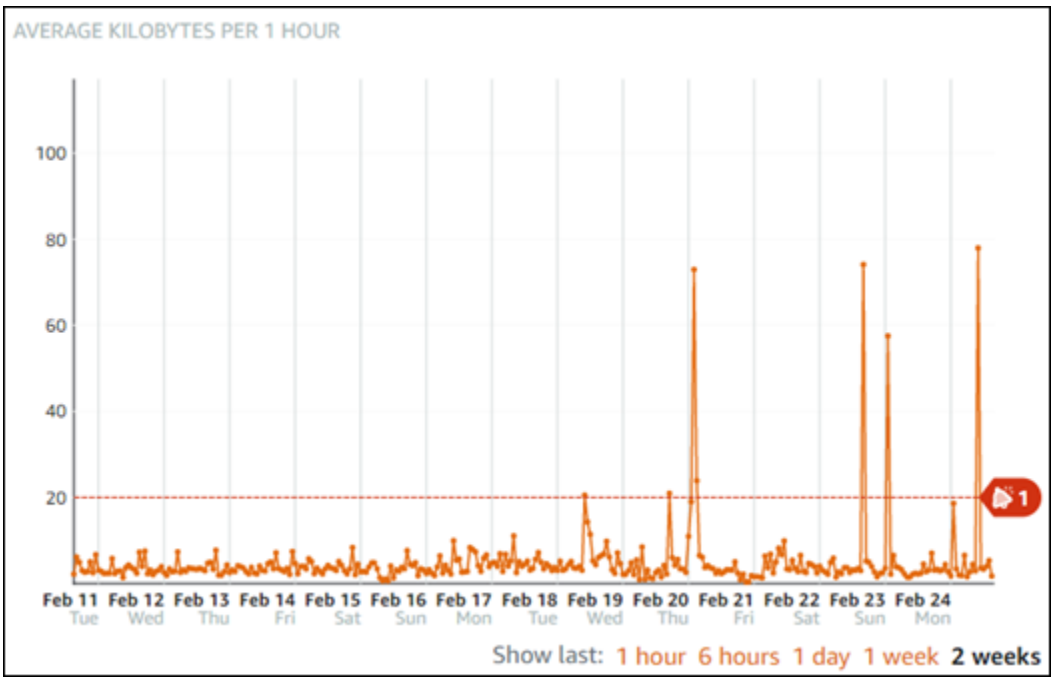
- メトリクスごとに 2 つのアラームを設定できます。
- アラームは 5 分間隔で評価され、アラームの各データポイントは、集計されたメトリクスデータの 5 分間隔を表します。
- アラームの状態が OK に変わったときに通知するようにアラームを設定できるのは、メールや SMS テキストメッセージで通知するようにアラームを設定した場合だけです。
- OK アラーム通知をテストできるのは、メールや SMS テキストメッセージで通知するようにアラームを設定した場合だけです。
- アラームの状態が INSUFFICIENT_DATA に変わったときに通知するようにアラームを設定できるのは、メールや SMS 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 [Networking] (ネットワーク) 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、SMS テキストメッセージをすべての国/リージョンに送信することはできません。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、次のステップを実行します。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 [Networking] (ネットワーク) 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。通知をテストすることを選択した場合のみ、ALARMLightsail コンソールに通知バ

ナーが表示されます。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、E メールと携帯電話を Lightsail から削除します。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 モニタリングの通知連絡先を設定する

インスタンス、データベース、ロードバランサー、または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されたしきい値を超えたときに通知するように Amazon 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、リソースをモニタリング AWS リージョン する各 で、E 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

- 通知連絡先を追加した後の次の手順

リージョンの通知の連絡先の制限

各 に追加できる E メールアドレスと携帯電話番号は 1 つだけです AWS リージョン。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。

で複数の E メール受信者が必要な場合は AWS リージョン、複数の受信者に転送するディストリビューションリストを設定し、ディストリビューションリストの E 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、E メールメッセージングを設定したり、Lightsail コンソールに表示される通知バナーに依存したりすることもできます。
> 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は、 AWS リージョン Lightsail リソースを作成できるすべての 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 AWS リージョンがサポートされていない では、E 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、次の でサポート AWS リージョンされています。これらは、SMS テキストメッセージが Amazon Simple Notification Service (Amazon SNS) でサポートされているリージョンです。Amazon SNS は、Lightsail が通知を送信するために使用します。

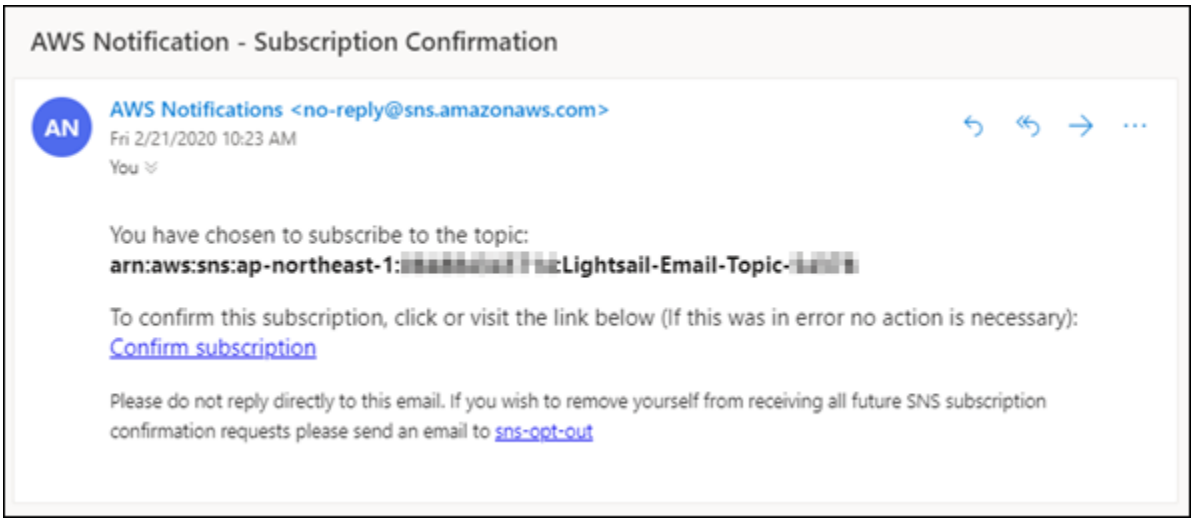
- 米国東部 (バージニア北部) (us-east-1)
- 米国西部 (オレゴン) (us-west-2)
- アジアパシフィック (シンガポール) (ap-southeast-1)
- アジアパシフィック (シドニー) (ap-southeast-2)

- 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)

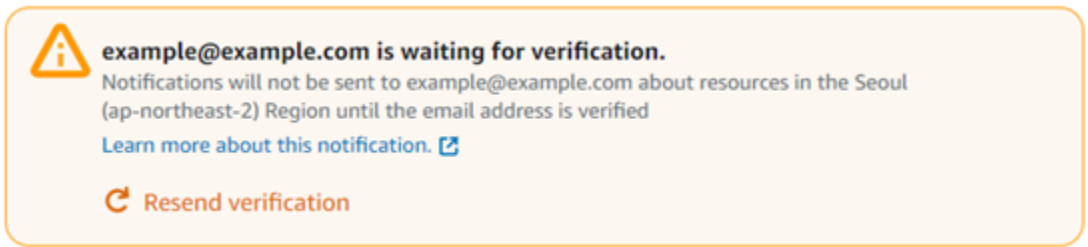
SMS テキストメッセージを送信できる国と地域、および SMS テキストメッセージがサポートされている最新の のリストについては、 AWS リージョンAmazon SNS デベロッパーガイド」の「サポートされているリージョンと国」を参照してください。

メールによる連絡先の確認

Lightsail の通知連絡先として E メールアドレスを追加すると、検証リクエストがそのアドレスに送信されます。検証リクエスト E 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。



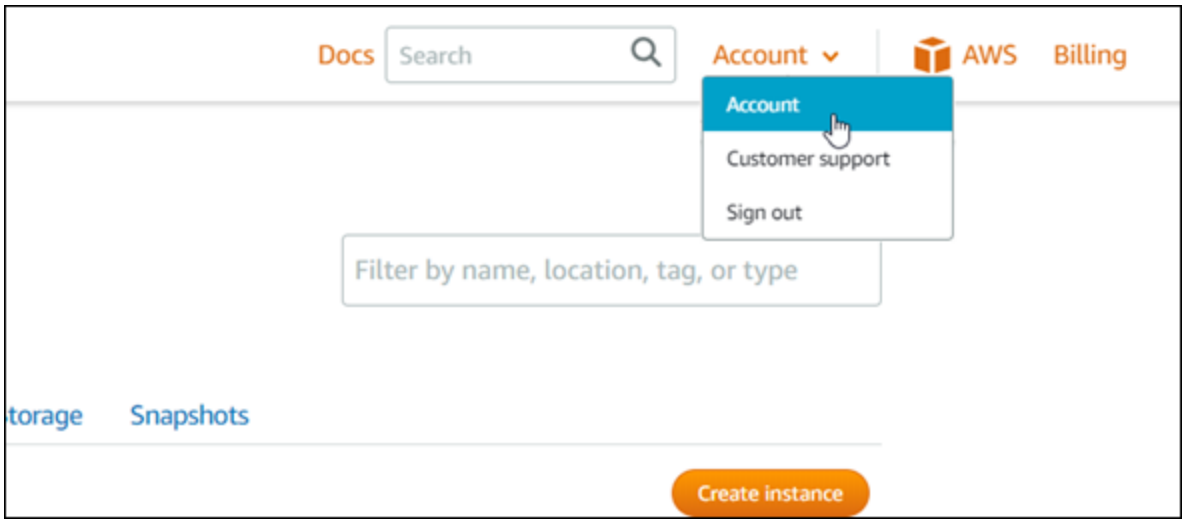
確認要求が受信トレイフォルダにない場合は、メールボックスのスパムフォルダと迷惑メールフォルダを確認してください。検証リクエストが失われた場合、または削除された場合は、Lightsail コンソールとアカウントページに表示される通知バナーで検証を再送信を選択します。



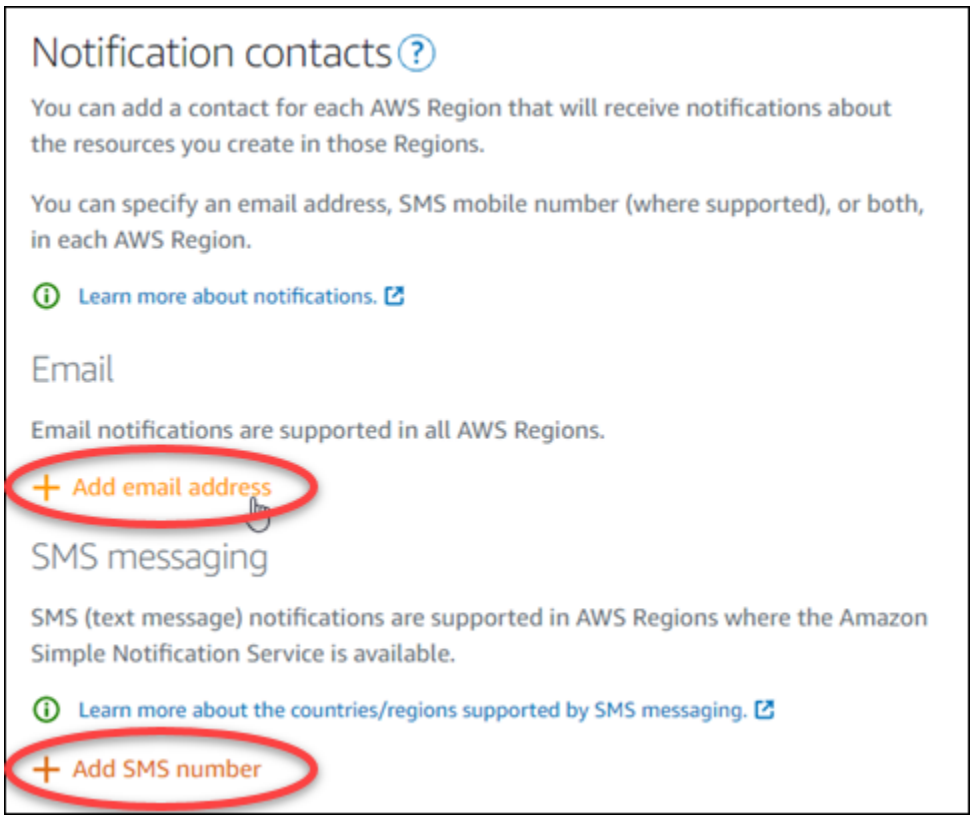
Lightsail コンソールを使用した通知連絡先の追加

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 ホームページのトップ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。

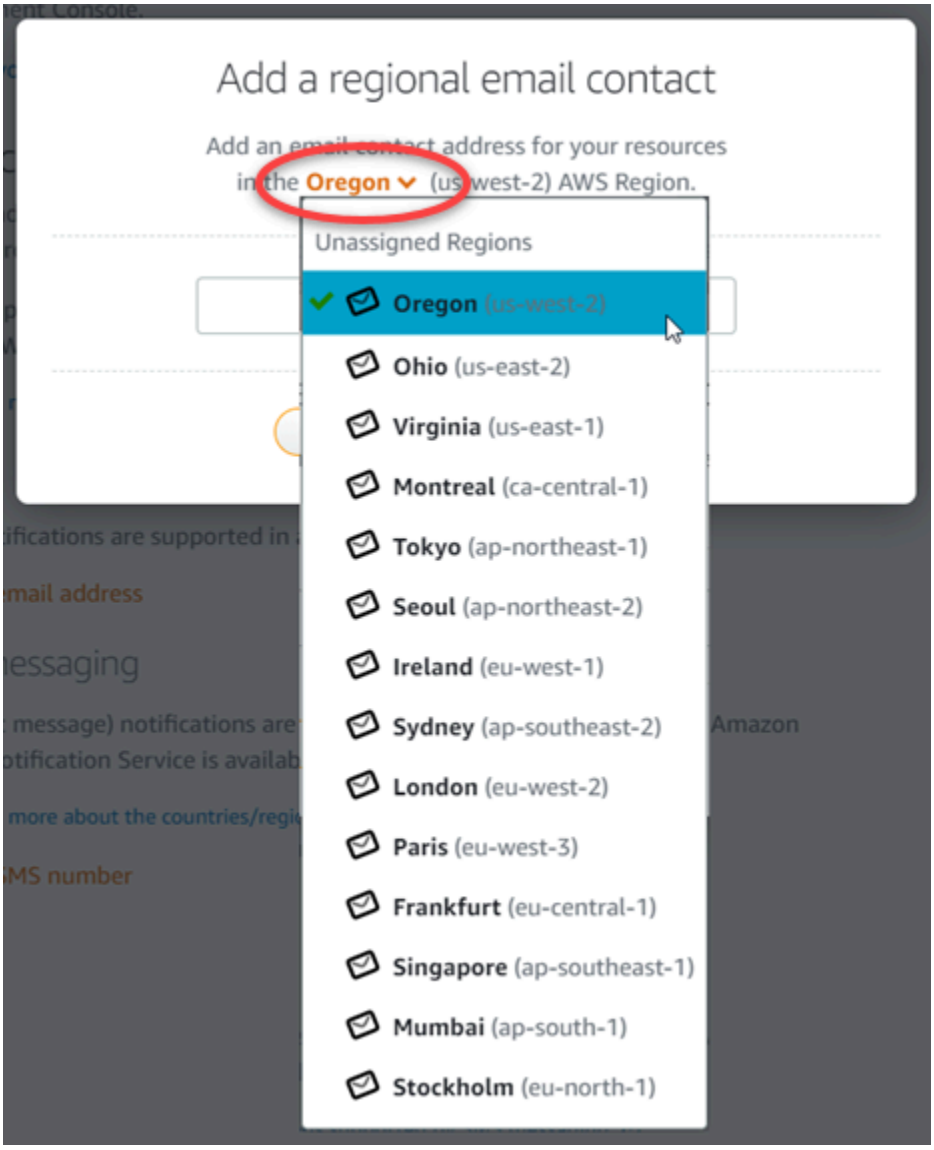


4. 通知連絡先のプロフィールと連絡先タブで、メールアドレスの追加を選択、または SMS 番号を追加するを選択します。



5. 次のいずれかのステップを完了します。

   - E メールアドレスを追加する場合は、通知連絡先 AWS リージョン を追加する 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。

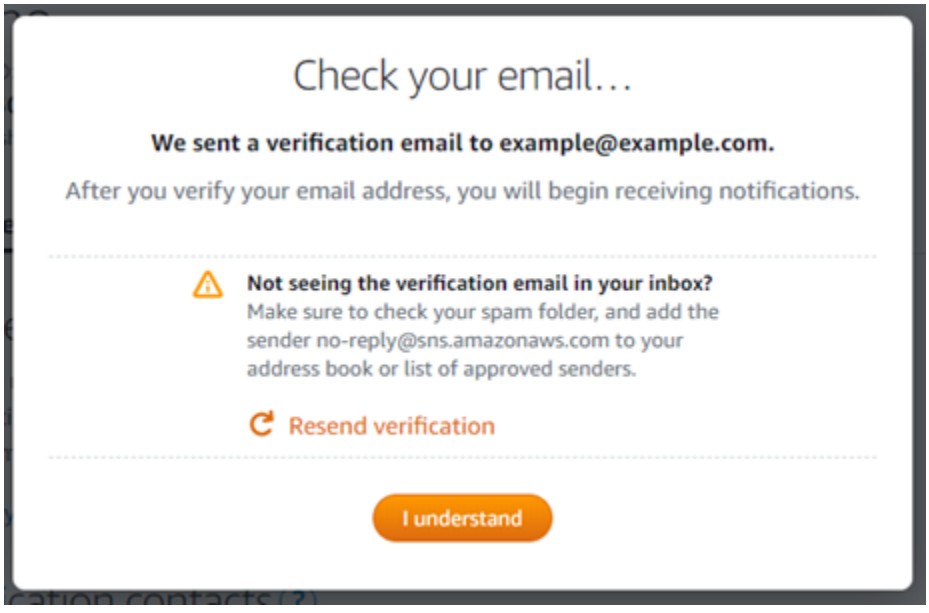


- SMS 番号を追加する場合は、通知連絡先 AWS リージョン を追加する を選択します。携帯電話番号の国を選択し、テキストボックスに入力します。国コードは既に入力されています。

> **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。

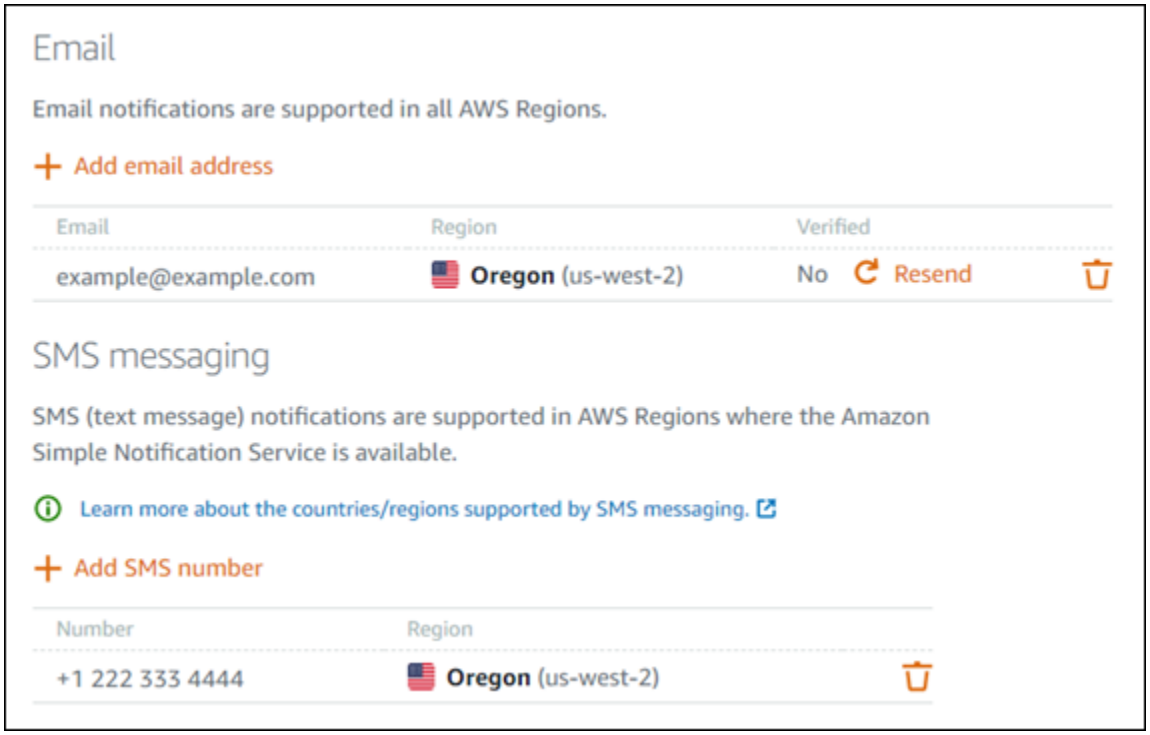
   メールアドレスを通知連絡先として追加すると、そのアドレスに確認要求が送信されます。検証リクエスト E メールには、受信者が Lightsail 通知を受信することを確認するためにクリックする必要があるリンクが含まれています。SMS メッセージングは検証を必要としません。



7. [I understand (理解する)] を選択します。

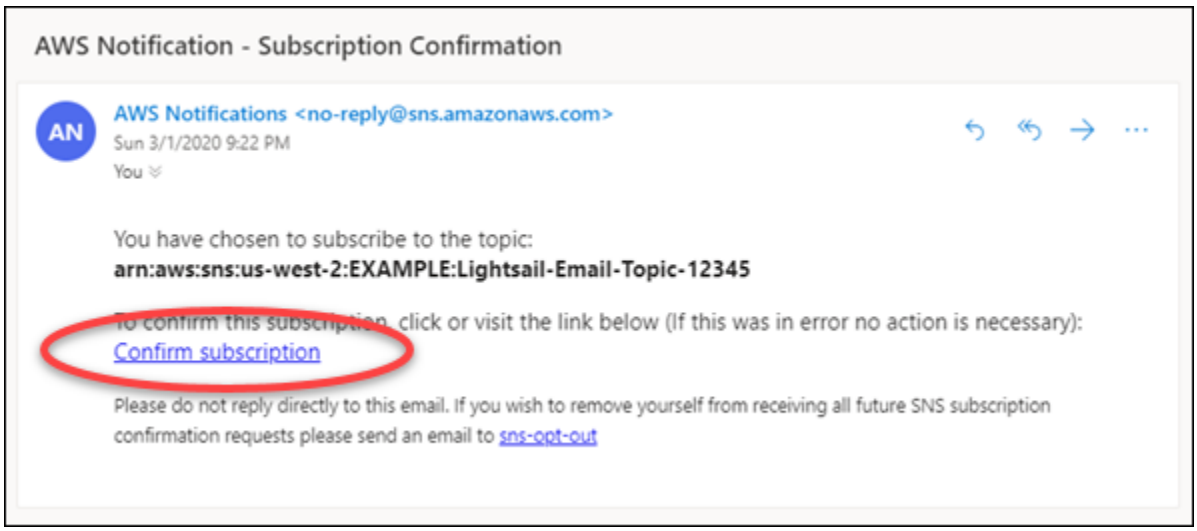
メールアドレスまたは携帯電話番号が [通知連絡先] セクションに追加されます。次の手順で確認プロセスを完了するまで、メールアドレスは確認されません。確認が完了するまで、通知はメールアドレスに送信されません。認証リクエストが紛失したか、削除された場合は、リージョンのメールアドレスの横にある [Resend (再送信)] を選択して、別の認証リクエストを送信します。

> **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。



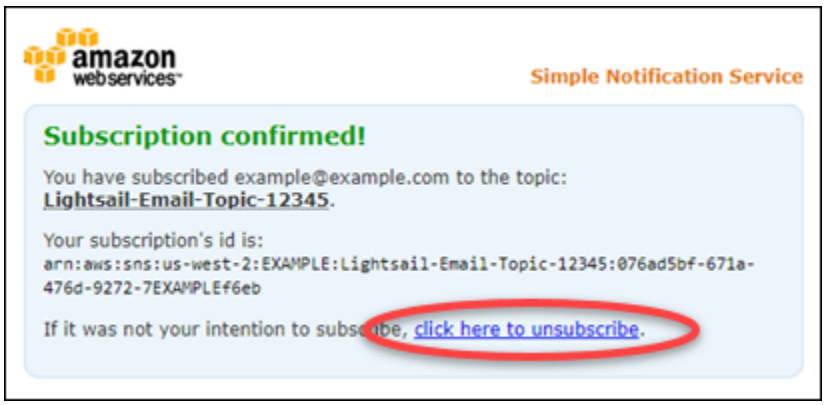
8. Lightsail の通知連絡先として追加した E メールアドレスの受信トレイを開きます。
9. no-reply@sns.amazonaws.com からの AWS 通知 - サブスクリプションの確認 メールを開きます。

> **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. E メールの「サブスクリプションの確認」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

AWS Command Line Interface () を使用して Lightsail の通知連絡先を追加するには、次のステップを実行しますAWS CLI。

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、 をインストール AWS CLIし、Lightsail で動作するように設定してください。

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。

- 通知連絡先を追加する AWS リージョン がある*#####*。
- 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 通知をサブスクライブすること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. E メールの「サブスクリプションの確認」を選択して、Lightsail から E 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 AWS リージョン した 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、E メールと携帯電話を Lightsail から削除します。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 で通知連絡先を削除する

Amazon Lightsail から E メールと携帯電話番号の通知連絡先を削除して、Lightsail リソースの E メールと SMS テキストメッセージの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。

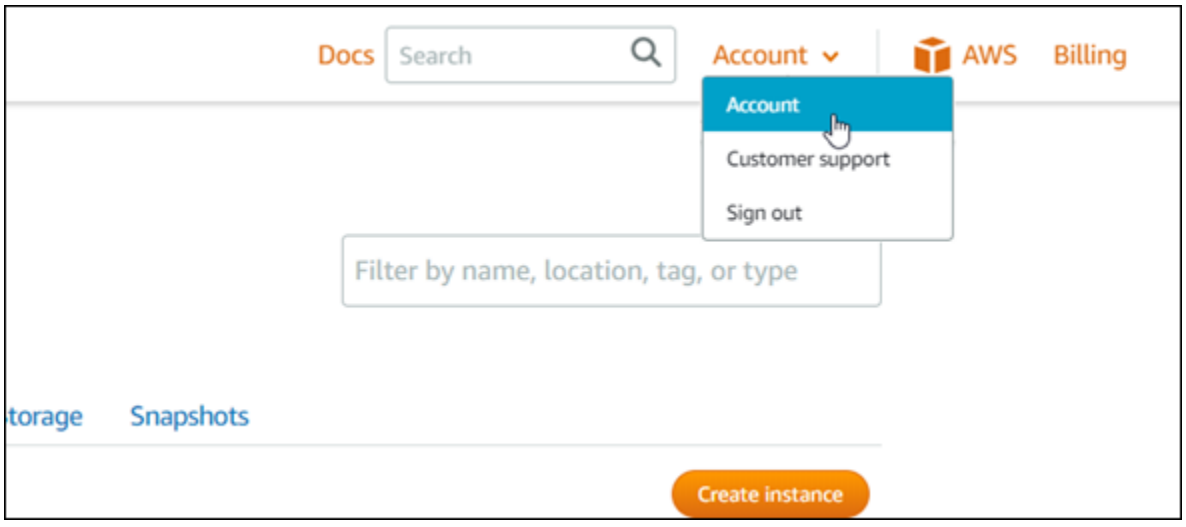
また、アラームを無効にするか、削除して、特定のアラームの通知の受信を停止することもできます。詳細については、「メトリクスアラームを削除または無効化する」を参照してください。

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 ホームページのトップ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 () を使用して Lightsail の通知連絡先を削除するには、次のステップを実行しますAWS CLI。

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、 をインストール AWS CLIし、Lightsail と連携するように設定してください。

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。

   - 通知連絡先を削除する AWS リージョン がある*#####*。
   - 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。

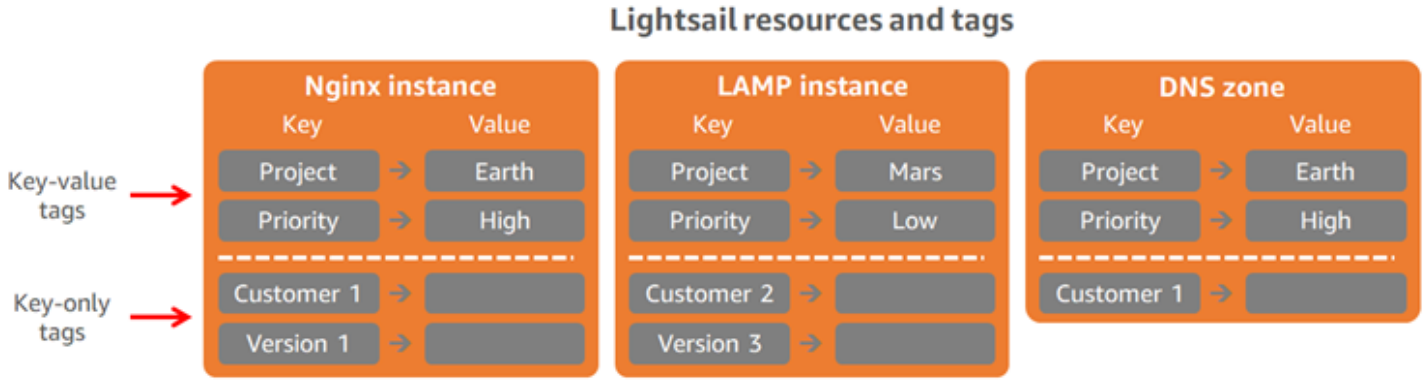
- 通知連絡先を削除しても、E メールと SMS テキストメッセージの通知は停止しますが、Lightsail コンソールに通知バナーが表示されなくなります。通知バナーを停止し、メールおよび SMS テキストメッセージング通知も停止するには、バナーの原因となっているアラームを無効にするか削除します。詳細については、「メトリクスアラームを削除または無効化する」を参照してください。
- Lightsail で E メールアドレスと携帯電話番号を通知連絡先として追加し、E メールと SMS テキストメッセージの通知の受信を再開します。詳細については、「通知連絡先を追加する」を参照してください。

タグを使用して Lightsail リソースを整理およびフィルタリングする

Amazon Lightsail では、リソースにタグとしてラベルを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 では、リソースにタグとしてラベルを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。タグには固有のタイプはありませんが、Lightsail リソースを目的、所有者、環境、またはその他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 では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 でのリソースとリクエストへのアクセスの制御、タグキーへのアクセスの制御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 リソース

ほとんどの Lightsail リソースは、作成時または作成後にタグ付けできます。リソースの作成中にタグを適用できない場合、Lightsail はリソース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。

Lightsail コンソールでは、次の Lightsail リソースに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。そのスナップショットから作成された 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に存在していたのと同じタグが付けられます。

Lightsail API 、 ()AWS Command Line InterfaceAWS CLI、または SDKs を使用して、次のリソースにタグを付けることができます。

- 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 を使用して作成された TLS 証明書)
- ロードバランサー

> **Important**
>
> Lightsail API、、または SDKs を使用して作成されたスナップショットは AWS CLI、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 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、他のサービスにも許容される文字数に制限がある可能性があることに注意してください。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 リソースをタグで分類する

Amazon Lightsail のタグを使用して、リソースを目的、所有者、環境、またはその他の基準で分類します。タグは、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。

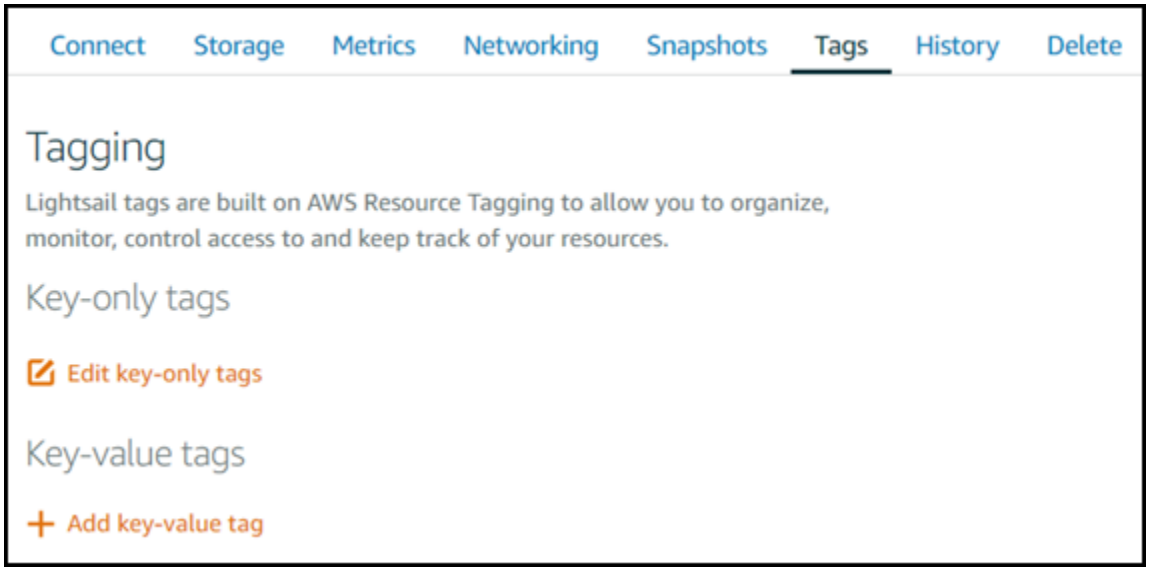
> **Note**
>
> タグ、タグを追加できるリソース、および制限の詳細については、「タグ」を参照してください。

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 ホームページで、タグ付け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。

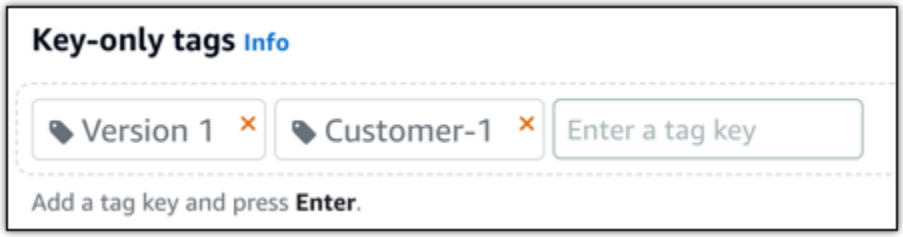
   > **Note**
   >
   > インスタンス、コンテナサービス、CDN ディストリビューション、バケット、データベース、ディスク、DNS ゾーン、ロードバランサーは、Lightsail コンソールを使用してタグ付けできます。ただし、Lightsail API オペレーション 、 () または SDK を使用して、より多くの Lightsail リソースにタグ付けできます。 AWS Command Line InterfaceAWS CLI SDKs タグ付けをサポートする Lightsail リソースの完全なリストについては、「タグ」を参照してください。

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。



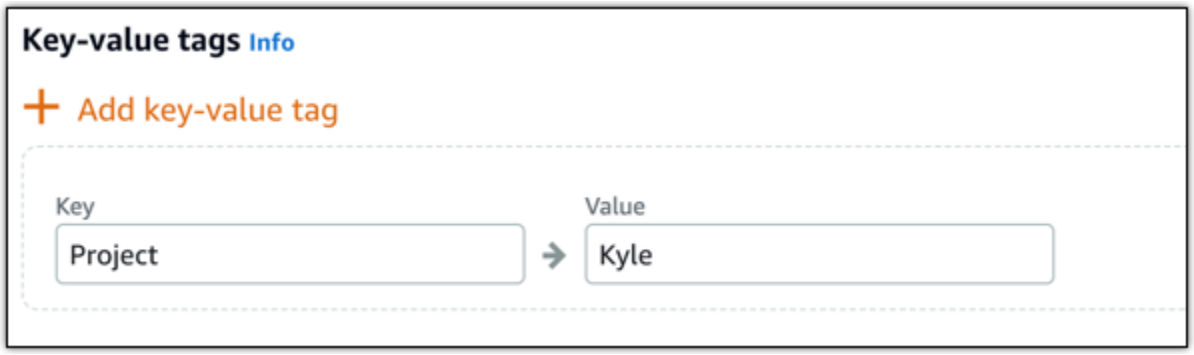
5. 追加するタグのタイプに応じて、以下のいずれかのオプションを選択します。

- [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 リソースからタグを削除する

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。

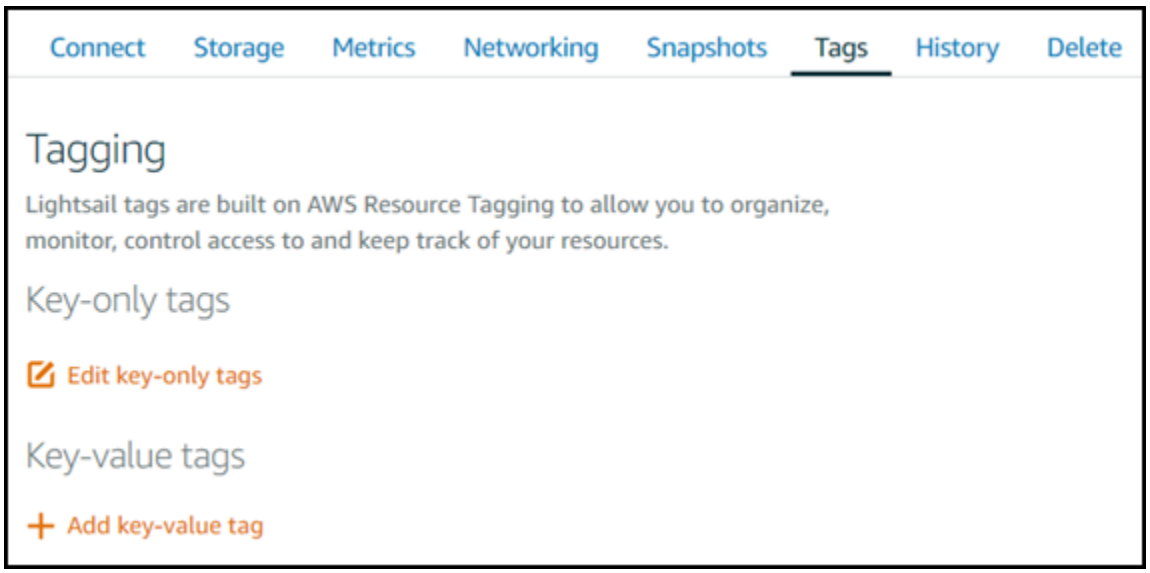
> **Note**
>
> タグ、タグを追加できるリソース、およびタグの制限の詳細については、「タグ」を参照してください。

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 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**
   >
   > 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、ロードバランサーは、Lightsail コンソールを使用してタグ付けできます。ただし、Lightsail API オペレーション 、 コマンドラインインターフェイス () または SDK を使用して、より多くの Lightsail SDKsタグ付けできます。 AWSAWS CLIタグ付けをサポートする Lightsail リソースの完全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とタグベースの承認を使用して Lightsail リソースへのアクセスを制御する

Lightsail は、一部のAPIアクションのタグに基づくリソースレベルのアクセス許可と認可をサポートしています。詳細については、「サービス認証リファレンス」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 リソースアクセスを制御する

Amazon Lightsail のタグを使用して、リソースへのアクセスの制御、リクエストへのアクセスの制御、タグキーへのアクセスの制御を行うことができます。このガイドでは、Lightsail リソースの作成または削除に必要なキーバリュータグを指定する AWS Identity and Access Management (IAM) ポリシーを作成し、それらのリクエストを行う必要があるユーザーまたは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、タグ付けできるリソース、および制限の詳細については、「タグ」を参照してください。

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。

次のポリシーtrueでは、 のキータグallowと の値が作成リクエストで定義されていない限り、ユーザーが新しい Lightsail リソースを作成することを制限します。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、キーバリューのペアを使用して 、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 リソースコストを整理する

Amazon Lightsail のタグを使用して、独自のコスト構造を反映するように AWS 請求を整理できます。これを行うには、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 コンソールでアクティブ化します。最後に、コスト配分レポートに含まれるタグキー値を使用して AWS アカウント請求書を取得するようにサインアップします。このセットアップ手順について以下に説明します。

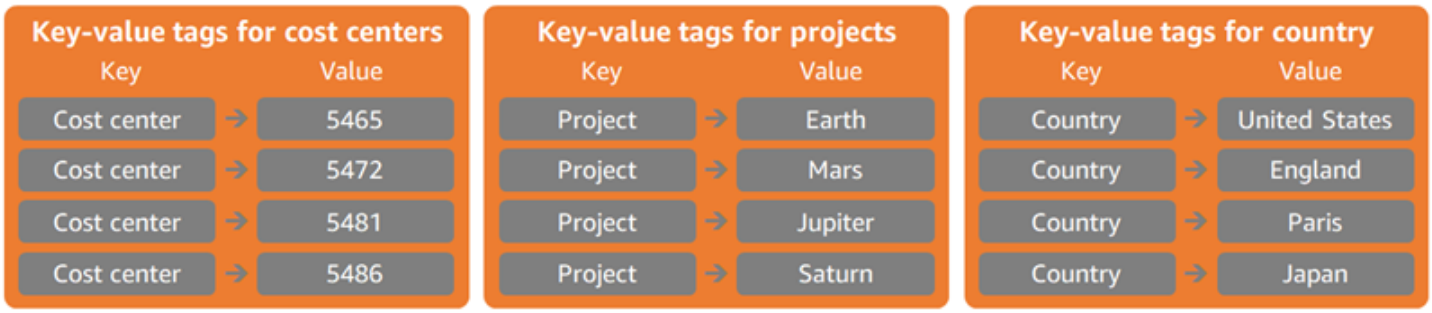
> **Note**
>
> Lightsail のタグ、タグ付けできるリソース、およびタグの制限の詳細については、「タグ」を参照してください。

> **Important**
>
> Lightsail データベーススナップショットは、コスト配分タグが追加された後でも、現時点ではコスト配分レポートで追跡できません。

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 Lightsail リソースにキーバリュータグを追加します。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。例えば、複数の Lightsail リソースに特定のコストセンターをタグ付けできます。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

Lightsail リソースに必要なタグを追加したら、請求情報とコスト管理コンソールでコスト配分のためにアクティブ化します。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アクティブ化」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

月別コスト配分レポートには、製品カテゴリ別およびリンクされたアカウントユーザー別のアカウント AWS の使用量が一覧表示されます。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。

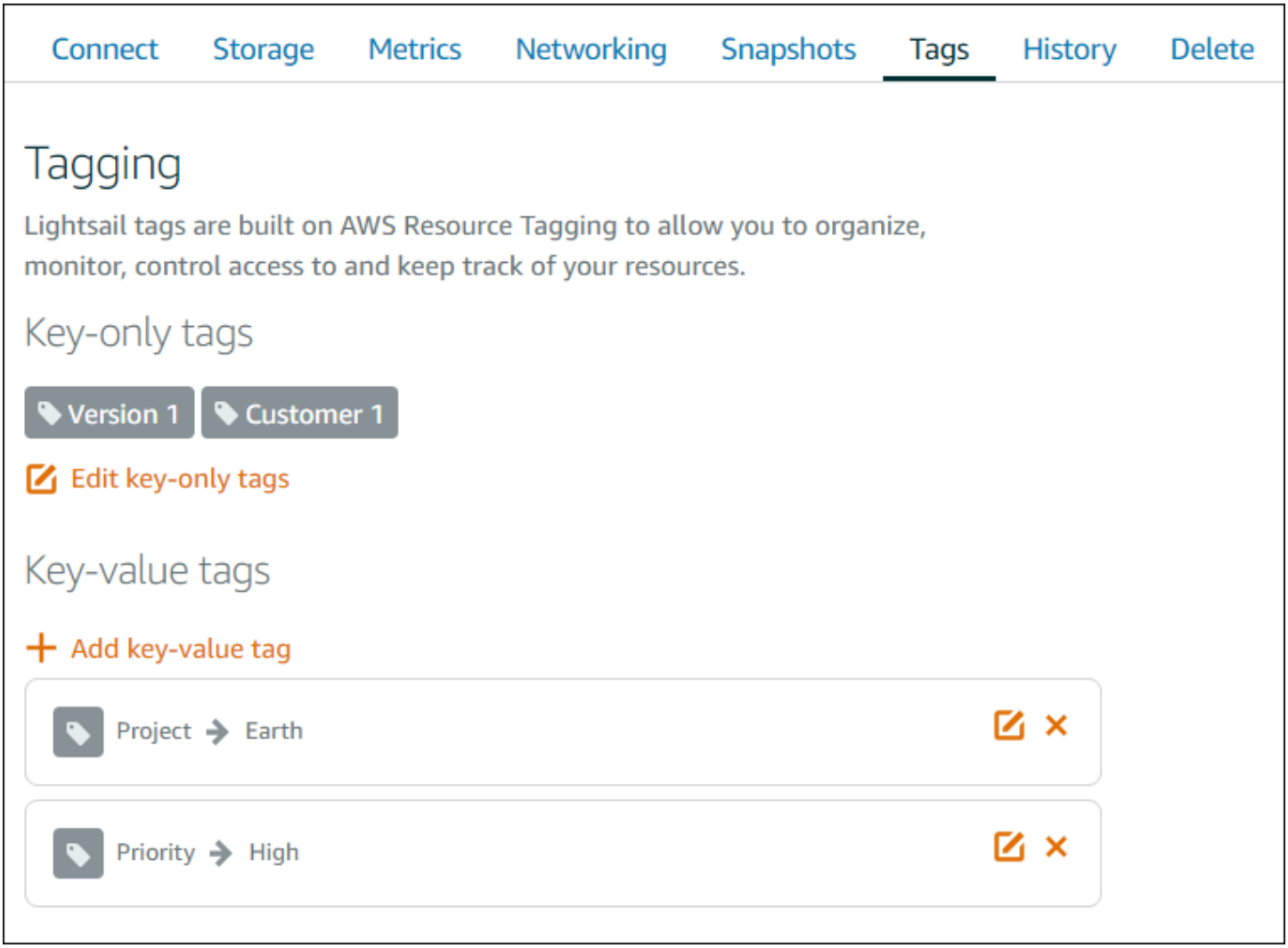
Lightsail リソースにタグ付けして整理およびフィルタリングする

Amazon Lightsail リソースにタグを付けたら、追加したタグでリソースをフィルタリングできます。これは、Lightsail コンソールでタグを選択または検索して行います。このガイドでは、Lightsail リソースをタグで表示およびフィルタリングする方法を説明します。

> **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、ロードバランサーは、Lightsail コンソールを使用してタグ付けできるため、タグタブを含めることができます。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。



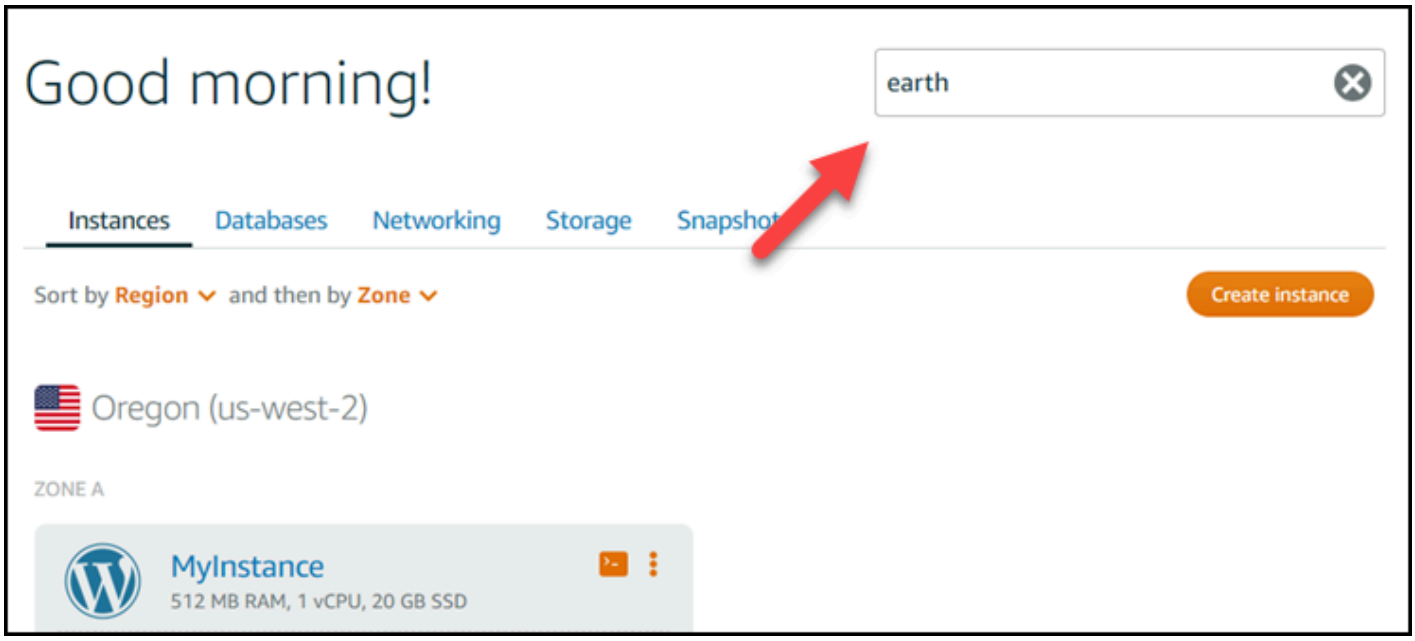
> **Note**
>
> インスタンス、コンテナサービス、CDN ディストリビューション、バケット、データベース、ディスク、DNS ゾーン、ロードバランサーは、Lightsail コンソールを使用してタグ付けできます。ただし、Lightsail API オペレーション 、 () または SDK を使用して、より多くの Lightsail リソースにタグ付けできます。 AWS Command Line InterfaceAWS CLI SDKs タグ付けをサポートする Lightsail リソースの完全なリストについては、「タグ」を参照してください。

タグを使用してリソースをフィルタ処理する

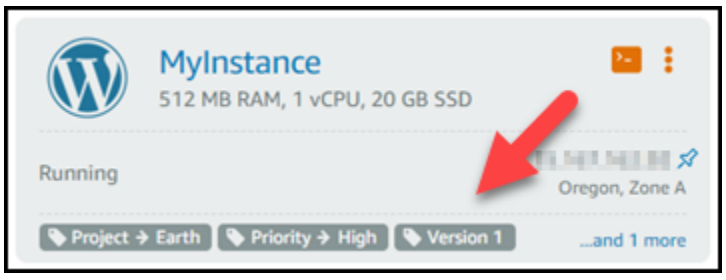
Lightsail コンソールでは、タグを使用してリソースをフィルタリングするために以下のオプションを使用できます。これらのオプションはすべて、Lightsail ホームページを更新して、検索または選択したタグのみを表示します。

> **Note**
>
> フィルタ処理の各オプションは永続的です。タグでフィルタリングし、Lightsail ホームページのセクション間を移動しても、フィルターは引き続き適用されます。

- Lightsail 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リングする値を入力し、Enter キーを押します。



- Lightsail ホームページの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。


MyInstance
512 MB RAM, 1 vCPU, 20 GB SSD
WordPress
Oregon, Zone A (us-west-2a)
Stop
Reboot
Project → Earth
Priority → High
Version 1
Customer 1
Manage 4 tags
Status: Running
Private IP:
Static IP:
Connect
Storage
Metrics
Networking
Snapshots
Tags
History
Delete

Lightsail リソースに関する一般的な問題のトラブルシューティング

このセクションでは、以下の Amazon Lightsail リソースのトラブルシューティングトピックについて説明します。 step-by-step 指示とガイダンスに従って、Lightsail インスタンス、データベース、ネットワーク、ロードバランサー、その他のリソースの使用中に発生する可能性がある一般的な問題を診断して解決します。

トラブルシューティングのトピックでは、設定の失敗、IAMアクセス許可の問題、ディスクエラー、接続の問題、サービス利用不能、IPv6接続、インスタンス容量の制限、ロードバランサーのエラー、通知配信の失敗、SSL/TLS 証明書の問題など WordPress、さまざまなシナリオについて説明します。このガイドに従うことで、Lightsail リソースに関連するさまざまな問題を効果的にトラブルシューティングして解決し、アプリケーションとワークロードの円滑な運用と最適なパフォーマンスを確保できます。

トピック

Lightsail インスタンス WordPress のセットアップに関する問題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中に 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、ワークフローの最後のステップで証明書の作成を選択した後すぐに発生します。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、古い WordPress 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発生します。例えば、インスタンスのパブリック IP アドレスを指さないDNSレコードを選択します。

セットアップの失敗

これらのタイプのエラーは、ワークフローの最後のステップを完了してから数分以内に発生します。これらの失敗メッセージは、インスタンスの接続タブのウェブサイトのセットアップ WordPress セクションに表示されます。これらのエラーは、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。

以下のトピックの情報は、 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ちます。

トピック

- Lightsail WordPress のセットアップエラーを解決する
- Lightsail WordPress のセットアップ失敗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、「インスタンスを設定する WordPress」を参照してください。

Lightsail WordPress のセットアップエラーを解決する

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。

メッセージの最初の行は、セットアップでエラーが発生したことを知らせるものです。

インスタンスのセットアップを完了できませんでした *InstanceName* の *InstanceRegion* リージョン 。

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、インスタンスに接続したり、インスタンスに接続したままにしたりできませんでした

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次のいずれかのエラーと一致させます。

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変更が反映されるまで待ちます。
- 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変更が反映されるまで待ちます。
- インスタンスに接続できません。SSH 接続の準備が完了するまで数分かかります。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 WordPress バージョン。セットアップは WordPress 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 日以降に作成された WordPress インスタンスのみをサポートします。
- インスタンスファイアウォールポート 22、80、および 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から変更できます。

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変更が反映されるまで待ちます。

理由

このエラーは、DNSレコードの設定ミス、またはインターネットの 全体に伝播するのに十分な時間がないDNSレコードが原因で発生しますDNS。

修正

A または AAAADNS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、DNSLightsail の「」を参照してください。

apex ドメイン (`example.com`) とそのwwwサブドメイン (`www.example.com`) からのトラフィックをポイントするDNSレコードを追加または更新する場合、レコードはインターネットの 全体に

伝播する必要がありますDNS。nslookup や DNS Lookup from などのツールを使用して、DNS変更が有効になったことを確認できますMxToolbox。

> **Note**
>
> DNS レコードの変更がインターネットの を介して伝播DNSされるまでに数時間かかることがあります。

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変更が反映されるまで待ちます。

理由

A レコードまたは AAAADNSレコードは、インスタンスのパブリック IP アドレスを指していません。

修正

A または AAAADNS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、DNSLightsail の「」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 を介して伝播DNSされるまでに数時間かかることがあります。

インスタンスに接続できません。SSH 接続の準備が完了するまで数分かかります。次に、セットアップを再度開始します。

理由

インスタンスが作成または再起動されたばかりであり、SSH接続の準備が完了していません。

修正

SSH 接続の準備が完了するまで数分かかります。次に、ガイド付きワークフローを再試行します。詳細については、Lightsail 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 WordPress バージョン。セットアップは WordPress バージョン 6 以降のみをサポートします。

理由

インスタンスにインストール WordPress されている のバージョンが WordPress バージョン 6 より古い。古い WordPress バージョンには、HTTPS証明書の生成を妨げる互換性のないソフトウェアと依存関係が含まれています。

修正

Lightsail コンソールから新しい WordPress インスタンスを作成します。次に、古いインスタンスから新しいインスタンスに WordPress ウェブサイトを移行します。詳細については、「既存の WordPress 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、アプリケーションの依存関係を新しいインスタンスに更新してください。

セットアップは、2023 年 1 月 1 日以降に作成された WordPress 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。古いソフトウェアを使用すると、HTTPS証明書が生成されなくなります。

修正

Lightsail コンソールから新しい WordPress インスタンスを作成します。次に、古いインスタンスから新しいインスタンスに WordPress ウェブサイトを移行します。詳細については、「既存の WordPress 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、アプリケーションの依存関係を新しいインスタンスに更新してください。

インスタンスファイアウォールポート 22、80、および 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から変更できます。

理由

インスタンスファイアウォールポート 22、80、および 443 は、セットアップの実行中に任意の 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。

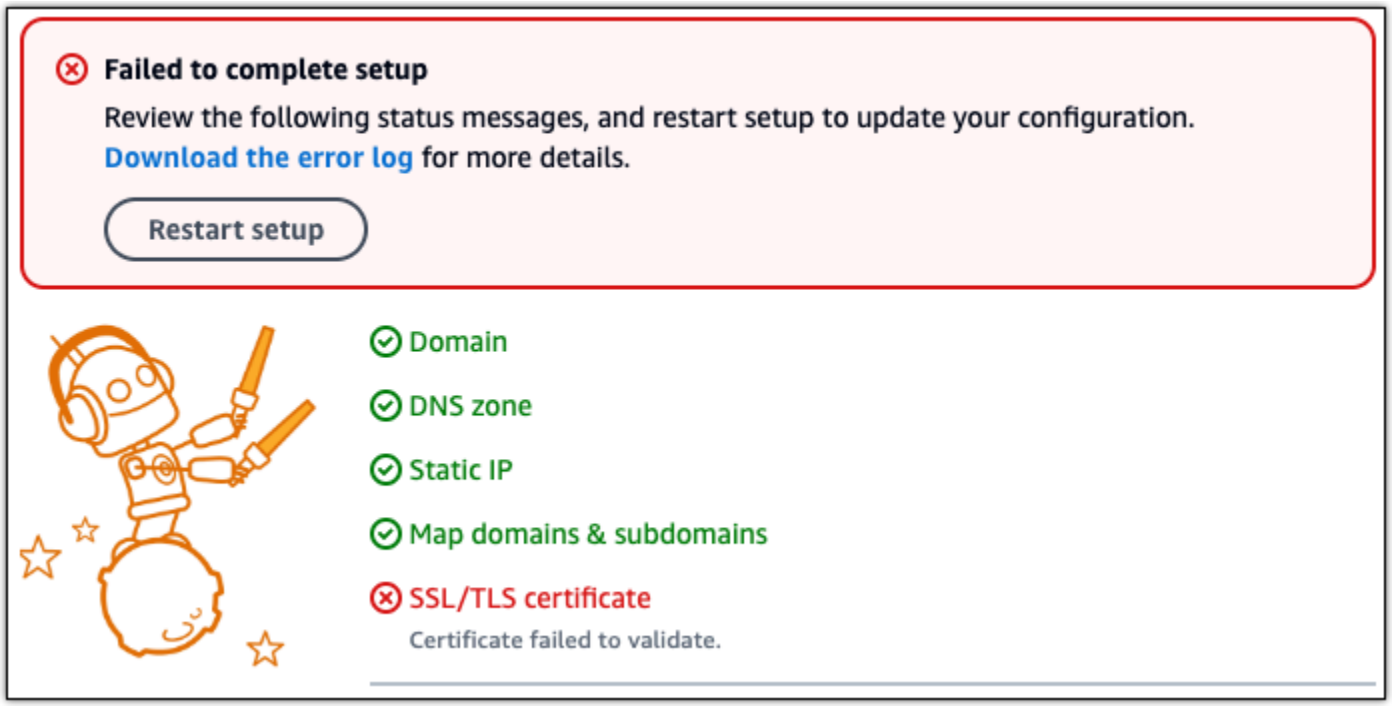
修正

インスタンスの IPv4およびIPv6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、および 443 経由TCPの接続を許可します。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 のセットアップ失敗のトラブルシューティング

以下の情報は、インスタンス 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る可能性のある障害メッセージのトラブルシューティングに役立ちます。セットアップの失敗は、ワークフローの最後の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で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 – 次のステータスメッセージを確認し、セットアップを再起動して設定を更新します。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、エラーログのダウンロードリンクを選択して、セットアップによって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、ログからのエラーメッセージを次のいずれかのエラーと一致させます。

エラー

- Certbot.errorsAuthorizationError: 一部のチャレンジが失敗しました
- Certbot が一部のドメインの認証に失敗しました
- リポジトリの http://cdn-aws.deb.debian.org/debian インスペクターバックポートに Release ファイルがありません
- リポジトリ http://ppa.launchpad.net/certbot/certbot/ubuntu lunar Release に Release ファイルがありません
- 過去 168 時間にこの正確なドメインセットに対して発行された証明書が多すぎます (5)
- 失敗した認証が多すぎる

Certbot.errorsAuthorizationError: 一部のチャレンジが失敗しました

理由

このエラーは、DNS レコードの設定ミス、またはインターネット全体に伝達するのに十分な時間がない DNS レコードが原因で発生します。

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) とそのwwwサブドメイン () からのトラフィックをポイントする DNS レコードを追加または更新するときwww.example.comは、インターネット全体に伝達する必要があります。nslookup や からの DNS Lookup などのツールを使用して、DNS の変更が有効になったことを確認できますMxToolbox。

> **Note**
>
> DNS レコードの変更がインターネットの DNS を介して伝播されるまでに時間を確保します。これには数時間かかる場合があります。

Certbot が一部のドメインの認証に失敗しました

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表示されることがあります。

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再度実行します。再起動しても問題が解決しない場合は、ポート 80 で実行されているインスタンスで実行中のプロセスをすべて終了するには、次の手順を使用します。

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に接続します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もあるかどうかを確認します。

```
$ fuser -n tcp 80
```

4. 別のアプリケーションで必要のないプロセスをすべて終了します。

```
$ fuser -k -n tcp 80
```

5. WordPress セットアップを再起動します。

リポジトリの http://cdn-aws.deb.debian.org/debian インスペクターバックポートに Release ファイルがありません

理由

インスタンスには、更新できない非推奨の Debian リポジトリがあります。

修正

次の手順を使用して、Debian リポジトリファイルにリストされているリポジトリ URL を編集します。

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に接続しますAWS CloudShell。
2. `/etc/apt/sources.list.d/` ディレクトリに移動します。

```
$ cd /etc/apt/sources.list.d/
```

3. 選択したテキストエディタを使用してbuster-backports.listファイルを開きます。ファイルがこのディレクトリにない場合は、 でチェックインすることもできます/etc/apt/sources.list。プリインストールされた Vim テキストエディタは、 コマンド例で使用されます。詳細については、Vimドキュメント を参照してください。

```
$ vim buster-backports.list
```

4. 次のテキストを含む行を見つけます: `http://deb.debian.org/debian buster-backports main`。

   `deb.debian.org` を `archive.debian.org` に置き換えます。例えば、`http://deb.debian.org/debian buster-backports main contrib non-free`は になりますhttp://**archive**.debian.org/debian buster-backports main contrib non-free。

5. ファイルを保存して閉じます。
6. WordPress セットアップを再起動します。

リポジトリ http://ppa.launchpad.net/certbot/certbot/ubuntu lunar Release に Release ファイルがありません

理由

インスタンスには、更新できない非推奨の Certbot Personal Package Archive (PPA) リポジトリがあります。

修正

非推奨の PPA リポジトリをインスタンスから手動で削除するには、次の手順に従います。

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に接続しますAWS CloudShell。
2. `/etc/apt/sources.list.d/` ディレクトリに移動します。

```
$ cd /etc/apt/sources.list.d/
```

3. 選択したテキストエディタを使用して `certbot-ubuntu-certbot-`**`version`**`.list` ファイルを開きます。プリインストールされた Vim テキストエディタは、 コマンド例で使用されます。詳細については、Vimドキュメント を参照してください。

   コマンドで、 をリポジトリ**version**と互換性のない Ubuntu のバージョンに置き換えます。これはエラーメッセージに表示されるのと同じバージョンになります。例えば、**lunar**、**mantic** などです。

```
$ vim certbot-ubuntu-certbot-version.list
```

4. 次のテキストを含む行をすべて削除します: `http://ppa.launchpad.net/certbot/certbot/ubuntu`。
5. ファイルを保存して閉じます。
6. WordPress セットアップを再起動します。

過去 168 時間にこの正確なドメインセットに対して発行された証明書が多すぎます (5)

理由

1 つ以上のドメインまたはサブドメインが、過去 1 週間以内に 5 つの証明書を作成するために既に使用されています。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起動します。

失敗した認証が多すぎる

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 つの検証の制限を超えています。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 、セットアップを再度実行します。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ます。

Lightsail コンソールで 403 (未承認) エラーを解決する

Lightsail コンソール にアクセスしようとしたときに 403 エラーが発生した場合は、パニックに陥らないでください。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) ユーザーが最近作成された場合は、数分待ってからブラウザを更新します。
- 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、Lightsail にアクセスできるIAMユーザーを使用してください。
- IAM ユーザーが Lightsail にアクセスできない場合は、AWS アカウントのルートユーザーまたは管理者権限を持つIAMユーザーに連絡して Lightsail へのアクセス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する」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 年以前に作成された AWS アカウントでは、まれに、アカウントを Lightsail に手動でサブスクライブする必要があります。

Lightsail ディスクのアタッチメントと使用状況の問題を解決する

Lightsail でブロックストレージディスクに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? コメント： このページの下部にあるリンクから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： ディスクがまだ Lightsail インスタンスにアタッチされているため、このオペレーションを実行できません。***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となるハードウェアが失敗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 インスタンスがまだ実行中であるため、ディスクをデタッチできません。

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は次のとおりです。*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アカウント全体で 20 TB を超えるディスクストレージを追加できないためです。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.

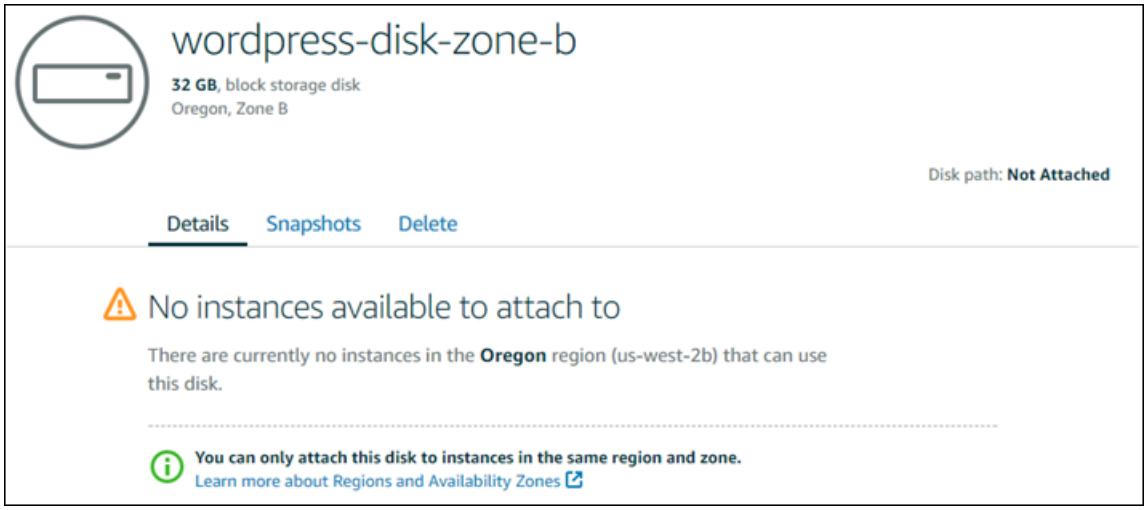
Lightsail でこれ以上ディスクを作成することはできません。

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アカウントで作成した大きなディスクが多すぎる可能性があります (ディスクストレージの合計サイズは 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.

Lightsail インスタンスにディスクをアタッチできない

次のエラーが発生した場合は、ディスクをアタッチするインスタンスと同じAWSリージョンと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： には現在インスタンスがありません ***AWS Region*** このディスクを使用できる 。

Lightsail ブラウザベースSSHおよびRDPクライアントでの接続エラーの解決

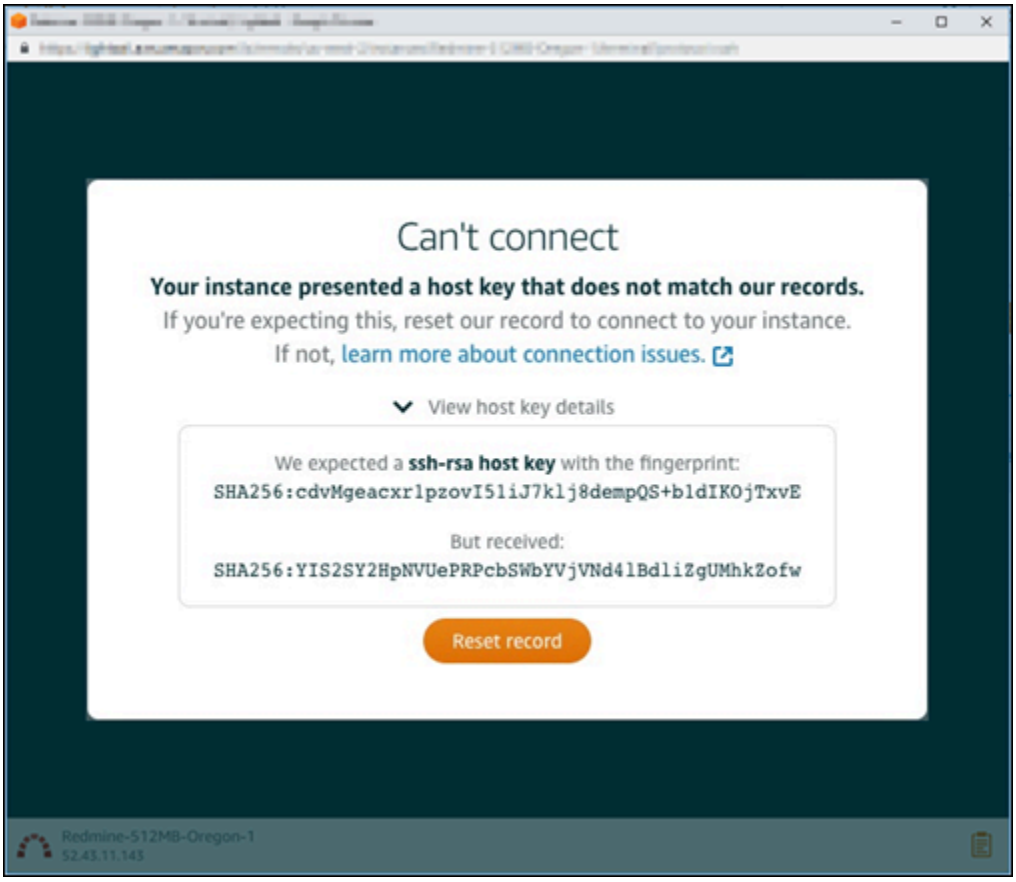
ブラウザベースSSHまたは Amazon Lightsail コンソールで利用可能なRDPクライアントを使用してインスタンスに接続しようとすると、エラーメッセージが表示されることがあります。このエラーが表示される理由として考えられるものは、次のセクションで説明します。

エラーメッセージ: 接続できません

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、インスタンスに接続しようとするときに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いずれかが表示されます。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、Lightsail が、その不一致が最近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引き起こされた可能性があると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が、ブラウザとインスタンス間のネットワーク上の不正なアクターによって引き起こされ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保持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。[*InstanceName*で、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*、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
```

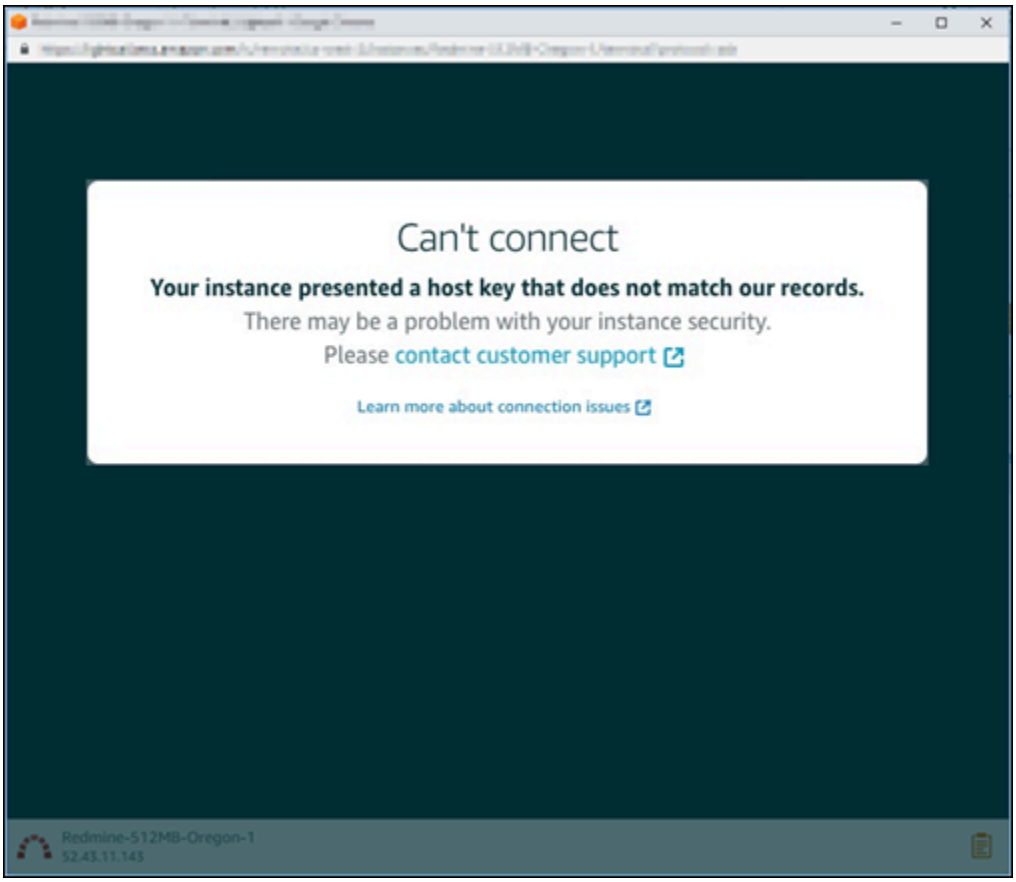
例:

```
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

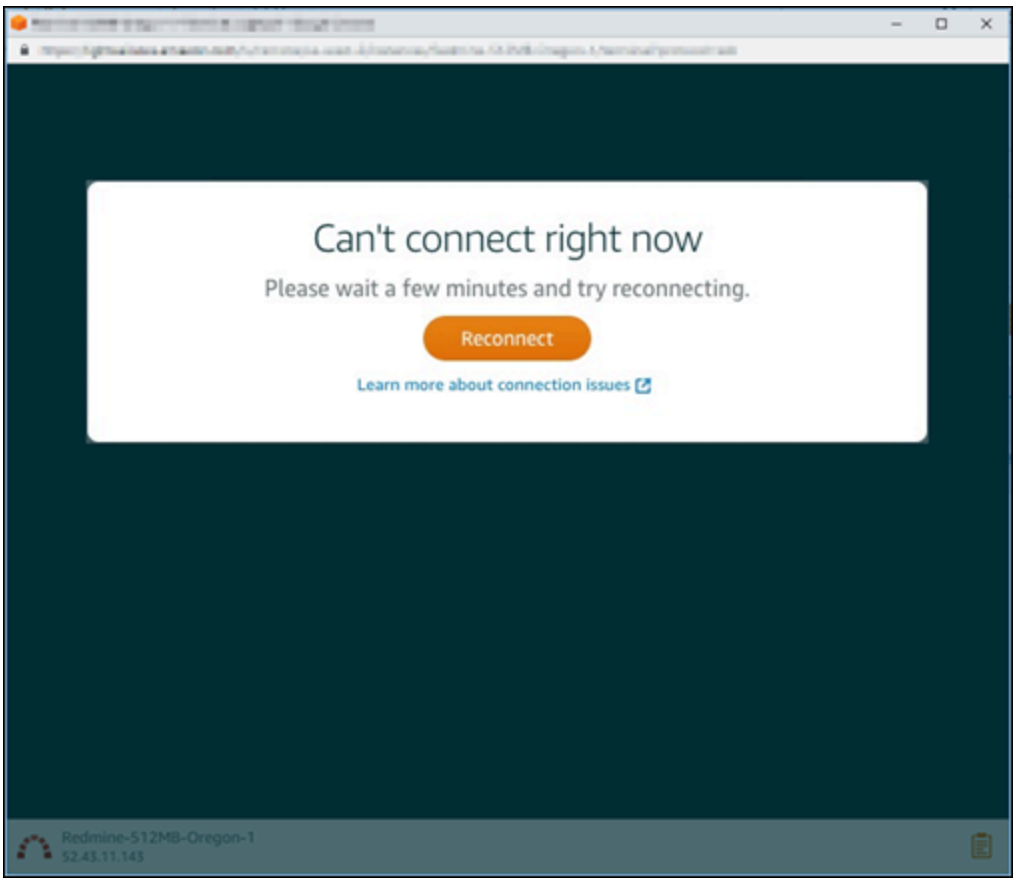
次のエラーメッセージは、ホストキーまたは証明書の不一致があり、Lightsail が man-in-the-middle 攻撃など、さらなる調査を必要とする疑わしいアクティビティがあると判断した場合に表示されます。



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。



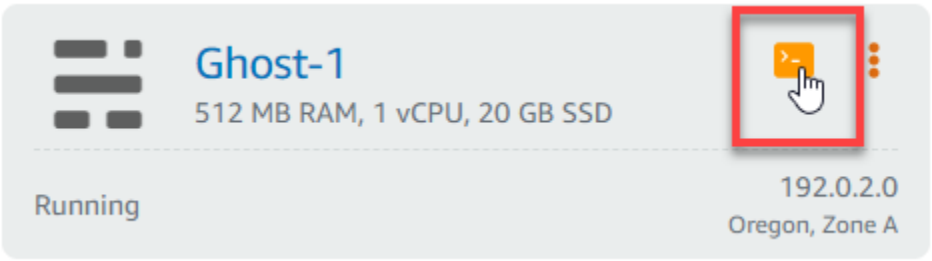
それでも接続できない場合は、 AWS サポートにお問い合わせください。

Lightsail での Ghost インスタンス 503 サービス利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) が表示されることがあります。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。これは、インスタンスの 5 USDUSD/月バンドルを選択した場合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、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、次のコマンドを入力して、インスタンス上のすべてのサービスを再起動します。

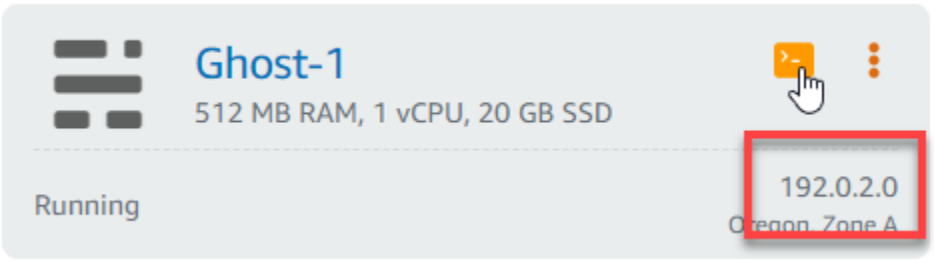
```
sudo /opt/bitnami/ctlscript.sh restart
```

以下の例のような結果が表示されるはずです。

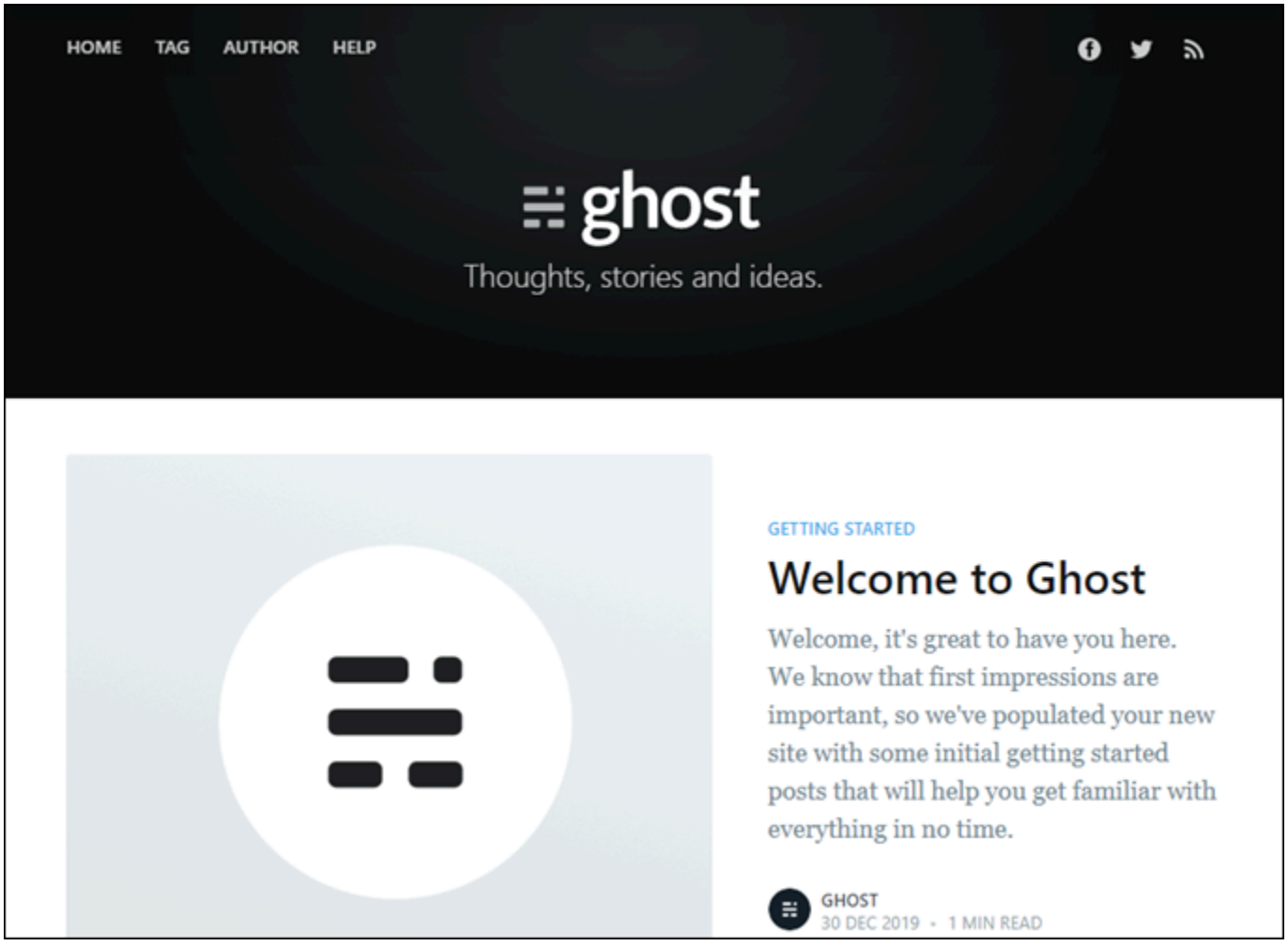
```
bitnami@ip-172-26-11-214:~$ sudo /opt/bitnami/ctlscript.sh restart
Syntax OK
/opt/bitnami/apache2/scripts/ctl.sh : httpd stopped
/opt/bitnami/apps/ghost/scripts/ctl.sh : ghost not running
/opt/bitnami/mysql/scripts/ctl.sh : mysql stopped
/opt/bitnami/mysql/scripts/ctl.sh : mysql  started at port 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表示されます。



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 での Identity and Access Management (IAM) のトラブルシューティング

以下の情報は、Lightsail と の使用時に発生する可能性がある一般的な問題の診断と修正に役立ちますIAM。

Lightsail でアクションを実行する権限がない

がアクションを実行する権限がないと AWS Management Console 通知した場合は、管理者に連絡して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。

次の例のエラーは、`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を更新するよう管理者に依頼します。

iam を実行する権限がありません。PassRole

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 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する代わりに、そのサービスに既存のロールを渡すことができます。そのためには、サービスにロールを渡す権限が必要です。

次の例のエラーは、 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marymajor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権限が必要です。メアリーには、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、Mary 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 管理者にお問い合わせください。サインイン認証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ら、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、新しいアクセス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 つで構成されています。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、アクセスキーは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、アクセスキーを第三者に提供しないでください。これにより、自分の への永続的なアクセスを誰かに許可することができます AWS アカウント。

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は、新しいアクセスキーをIAMユーザーに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、新規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。手順を確認するには、「 ユーザーガイド」の「アクセスキーの管理IAM」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ユーザーまたはアプリケーションにアクセス許可を付与する必要があります。を使用して AWS IAM Identity Center ユーザーとアプリケーションを管理する場合は、アクセスレベルを定義するアクセス許可セットをユーザーまたはグループに割り当てます。アクセス許可セットは、ユーザーまたはアプリケーションに関連付けられたIAMロールにIAMポリシーを自動的に作成して割り当てます。詳細については、「 ユーザーガイド」の「アクセス許可セットAWS IAM Identity Center 」を参照してください。

IAM Identity Center を使用していない場合は、アクセスが必要なユーザー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。次に、Amazon Lightsail の適切なアクセス許可を付与するポリシーをエンティティにアタッチする必要があります。アクセス許可が付与されたら、ユーザーまたはアプリケーション開発者に認証情報を提供します。これらの認証情報を使用して にアクセスします AWS。IAM ユーザー、グループ、ポリシー、アクセス許可の作成の詳細については、 IAMユーザーガイドのIAM「 での ID とポリシー、アクセス許可IAM」を参照してください。

AWS アカウント外のユーザーに Lightsail リソースへのアクセスを許可したい

他のアカウントのユーザーや組織外の人が、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できます。リソースベースのポリシーまたは

アクセスコントロールリスト (ACLs) をサポートするサービスでは、これらのポリシーを使用して、ユーザーにリソースへのアクセスを許可できます。

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、「」を参照してくださいAmazon Lightsail と の連携方法 IAM。
- 所有している のリソースへのアクセスを提供する方法については、 AWS アカウント 「 ユーザーガイド」の「所有 AWS アカウント している別の のIAMユーザーへのアクセスを提供するIAM」を参照してください。
- リソースへのアクセスをサードパーティー に提供する方法については AWS アカウント、 IAM ユーザーガイドの「サードパーティー AWS アカウント が所有する へのアクセスを提供する」を参照してください。
- ID フェデレーションを通じ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「外部認証されたユーザーへのアクセスの提供 (ID フェデレーション)」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については、「 ユーザーガイド」の「 でのクロスアカウントリソースアクセスIAMIAM」を参照してください。

Lightsail インスタンスの 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 接続を確認できます。Ping は、2 つ以上のネットワークデバイス間の接続問題のトラブルシューティングに使用されるネットワーク診断ユーティリティです。ping が成功した場合、IPv6 経由でインスタンスに接続できるはずです。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のみに関する考慮事項」を参照してください。

内容

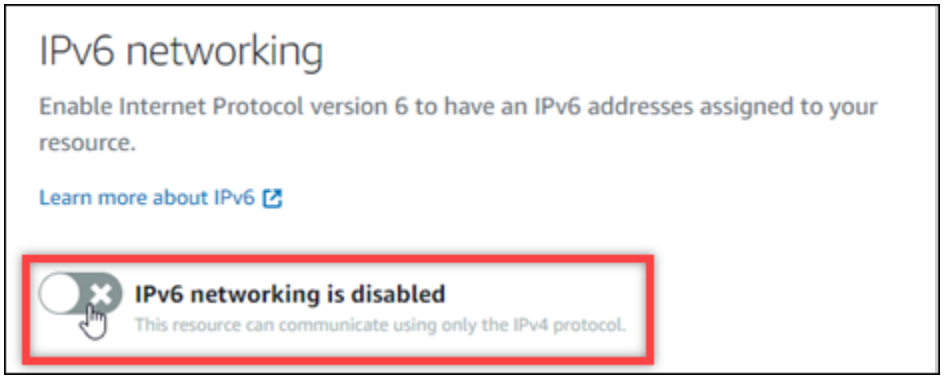
- デュアルスタックインスタンスで IPv6 を有効にする
- インスタンスのファイアウォールを設定する
- インスタンスへの到達可能性をテストする

デュアルスタックインスタンスで IPv6 を有効にする

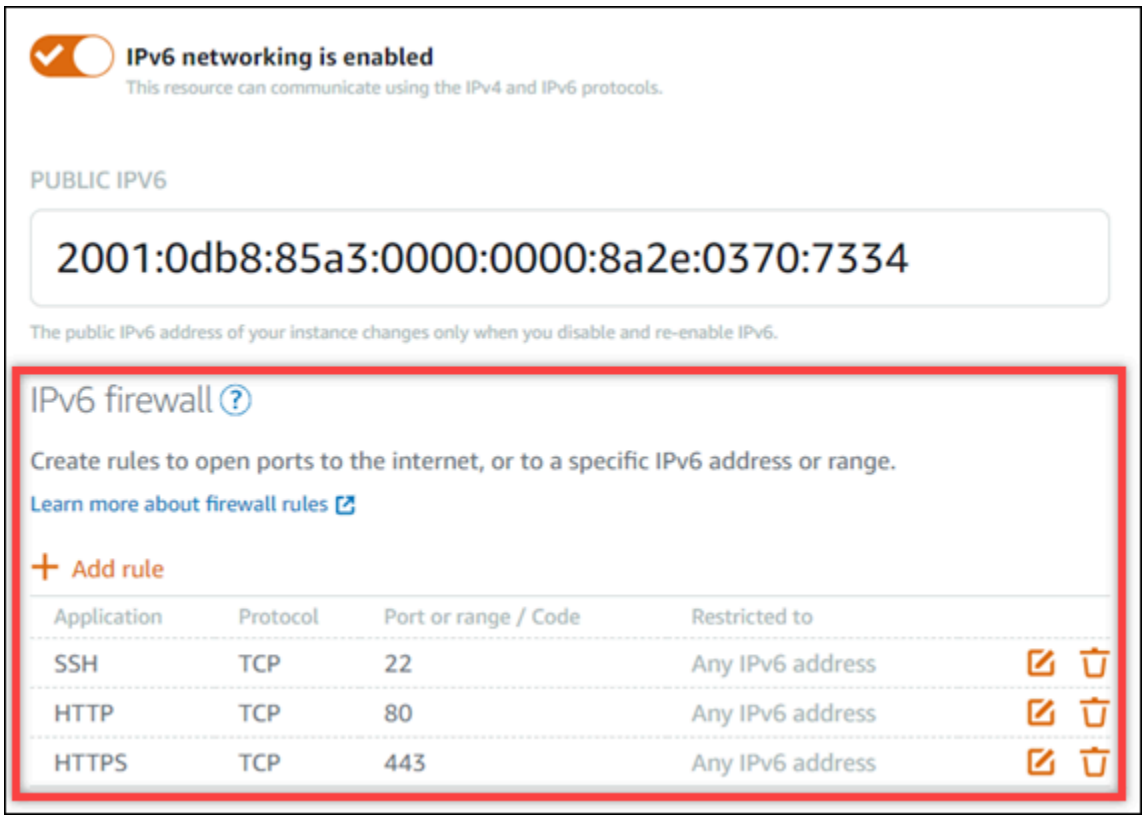
テストを開始する前に、デュアルスタックインスタンスの IPv6 を有効にします。IPv6 のみのインスタンスでは、IPv6-onlyは常にオンになっています。

デュアルスタックインスタンスが有効になっていない場合は、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されてい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 アドレス用に個別のファイアウォールがあり、IPv6 アドレス用に別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の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、「Lightsail のファイアウォールでインスタントラフィックを制御する」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。
>
> - を使用して Lightsail Windows インスタンスに接続する RDP
> - Lightsail で 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ついて、該当する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 Firewall」セクションの手順を完了します。IPv6 の場合は、IPv6 Firewall」セクションの手順を完了します。
   a. アプリケーションの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、ローカルの送信元 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、ボックスを選択したままにしておきます。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. Create を選択して、新しいルールをインスタンスに適用します。

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。コマンドのサンプル *IP ####*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 経由でテストするには

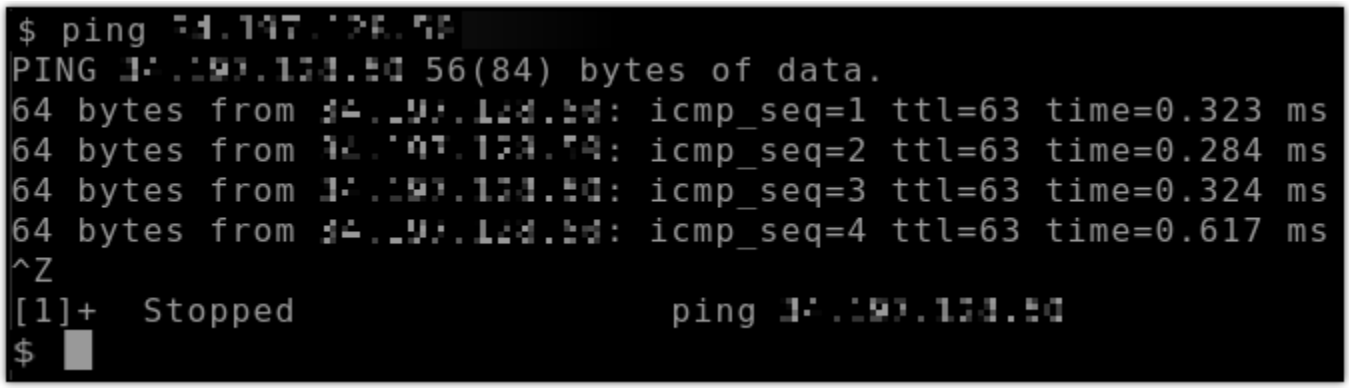
```
ping 192.0.2.0
```

IPv6 経由でテストするには

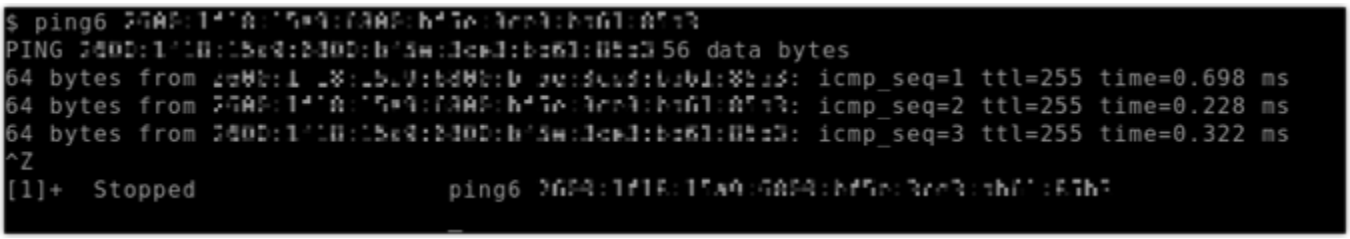
```
ping6 2001:db8::
```

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 と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 アドレスから成功した場合、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、成功した場合、インスタンスの IPv6 アドレスからの正常な返信を返します。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。コマンドのサンプル *IP ####*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 経由でテストするには

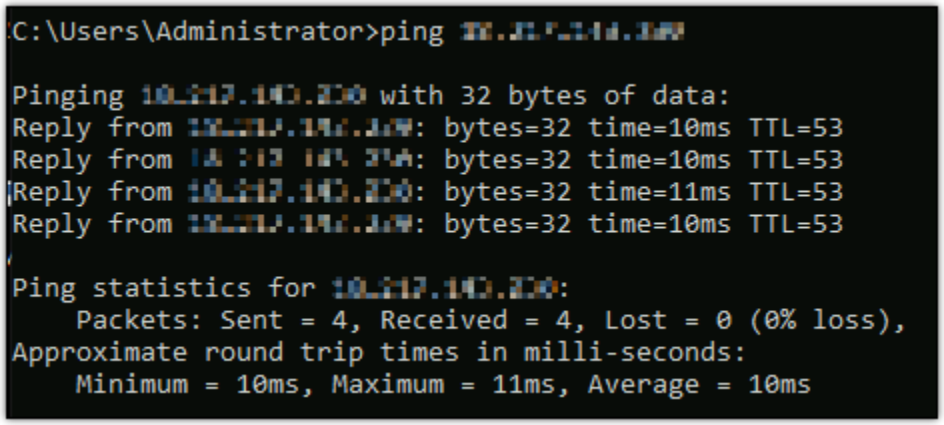
   ```
   ping 192.0.2.0
   ```

   IPv6 経由でテストするには

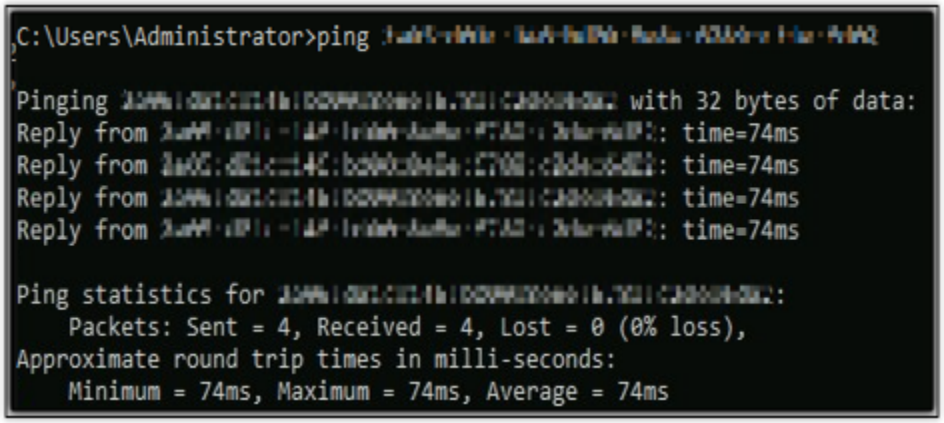
   ```
   ping 2001:db8::
   ```

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 と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 アドレスから成功した場合、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 アドレスから成功した場合、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。

Lightsail でのインスタンス容量不足エラーの解決

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、 AWS には、現在、リクエストを満たすために使用可能な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の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 (後の段階でサイズを変更可能) を選択して、インスタンスの起動を試みることもできます。

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は、オプション 3 に進みます。
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 の耐障害性

Lightsail ロードバランサーの問題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： このページの下部にあるリンクからフィードバックを送信したり、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 アカウントで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。ロードバランサーの Lightsail 証明書にも同じクォータが適用されます。

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。

アカウントあたり AWS 合計 20 個の Lightsail インスタンスのクォータの範囲内であれば、ロードバランサーにいくつでも Lightsail インスタンスをアタッチできます。

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確認します。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。Lightsail インスタンスをロードバランサーに正常にアタッチするには、実行中である必要があります。

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.

Lightsail がロードバランサーにアタッチしようとしているインスタンスを見つけられません

インスタンスが存在しなくなったか、ターゲットグループVPCと同じでは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： 残念ながら、指定したインスタンスが存在しないか、ターゲットグループVPCと同じ にないか、サポートされていないインスタンスタイプがあります。

Lightsail での通知配信のトラブルシューティング

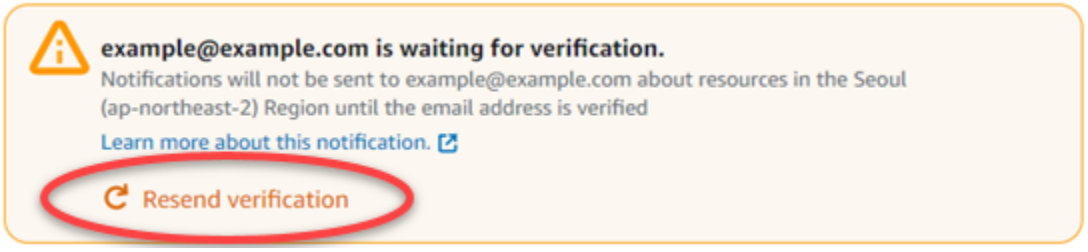
通知が予定されているときに通知を受け取らない場合は、通知の連絡先が正しく設定されていることを確認するためにいくつかの点を確認してください。通知の詳細については、「??? の通知」を参照してください 。

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

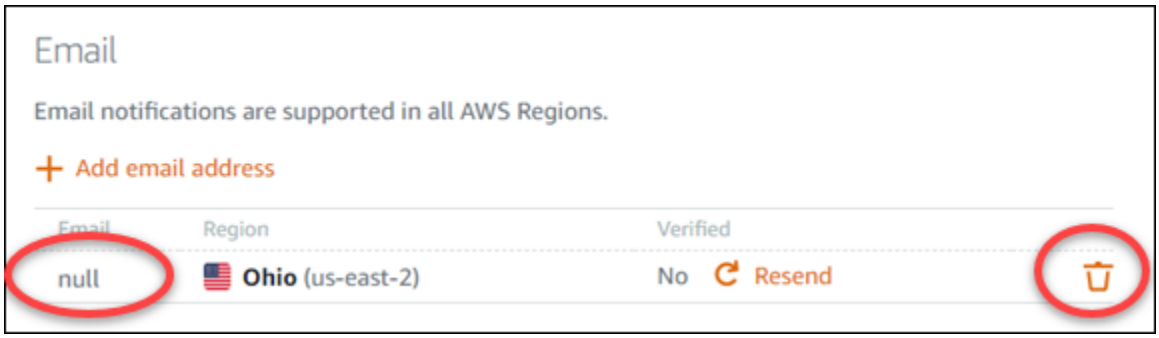
Lightsail で E メールアドレスを通知連絡先として追加すると、検証リクエストがそのアドレスに送信されます。検証リクエスト E メールには、受信者が Lightsail 通知を受信することを確認するためにクリックする必要があるリンクが含まれています。通知は、確認が完了するまでメールアドレスに送信されません。検証は、AWS 通知 <no-reply@sns.amazonaws.com> から行われ、件名は AWS 通知 - サブスクリプションの確認です。SMS メッセージングは検証を必要としません。

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場合、または削除された場合は、Lightsail コンソールに表示される通知バナーとアカウントページで検証を再送信を選択します。



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 E メールを検証できなかった場合、その E メールのステータスは自動的に `invalid`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。Lightsail に追加した直後に E メールアドレスを確認してください。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は、Lightsail の通知連絡先としてその携帯電話番号を再度追加できるようになるまで 30 日間待つ必要があります。

Lightsail での 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： このペー

ジの下部にあるリンクからフィードバックを送信したり、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 アカウントで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。ロードバランサーの Lightsail 証明書にも同じクォータが適用されます。

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。

それでも問題が解決しない場合は、AWSカスタマーサポートにお問い合わせください。

ドメインが無効と表示されました。

ドメインをコントロールしていることの確認に問題がある場合は、DNS管理にアクセスできることを確認してください。これらの指示に従っても検証できない場合は、AWSカスタマーサポートにお問い合わせください。

チュートリアルで Lightsail の機能を調べる

このセクションでは、Amazon Lightsail に関連する以下のトピックについて説明します。

トピック

Lightsail を使用する際のさまざまな側面に関する step-by-step ガイド、ベストプラクティス、追加情報にアクセスするには、各カテゴリに記載されているリンクに従ってください。

各トピックでは、アプリケーションのデプロイ、ネットワークの設定、モニタリングとログ記録、他の AWS サービスとの統合などの情報について説明します。このセクションでは、Lightsail を効果的に活用し、他の AWS サービスとの統合を活用し、豊富なチュートリアルやリソースにアクセスしてクラウドコンピューティングエクスペリエンスを向上させる方法について説明します。

Lightsail ブループリントを使用してアプリケーションをすばやくデプロイする

Lightsail ブループリントの使用を開始するには、次のクイックスタートガイドを使用します。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、、Nginx (LEMP) LAMP、Node.js などがあります。

トピック

- Lightsail で AlmaLinux インスタンスを起動してセットアップする
- Lightsail で cPanel & WHM を使用してウェブサイト、E メール、サービスをホストする
- Lightsail で Drupal ウェブサイトを設定およびカスタマイズする
- Lightsail に Ghost ウェブサイトをデプロイする
- Lightsail で GitLab CE インスタンスをセットアップおよび設定する
- Lightsail で Joomla! の使用を開始する
- Lightsail で LAMP スタックを設定する
- Lightsail で Magento をセットアップおよび設定する
- Lightsail で Nginx ウェブサーバーをデプロイして管理する
- Lightsail で Node.js の使用を開始する
- Lightsail に Plesk ホスティングスタックをデプロイする
- Lightsail で PrestaShop ウェブサイトを設定する
- Lightsail で Redmine インスタンスを設定して保護する
- Lightsail WordPress で を起動して設定する
- Lightsail でマルチサイトを設定する WordPress

Lightsail で AlmaLinux インスタンスを起動してセットアップする

このクイックスタートガイドでは、Amazon Lightsail プラットフォームで AlmaLinux インスタンスを作成および設定する step-by-step 手順について説明します。このトピックでは、インスタンスの場所と計画の選択、ネットワークとセキュリティの設定、CentOS から への移行など、重要なステップについて説明します AlmaLinux。これらのステップに従うことで、Lightsail で AlmaLinux インスタンスをすばやく起動して実行できます。

トピック

前提条件

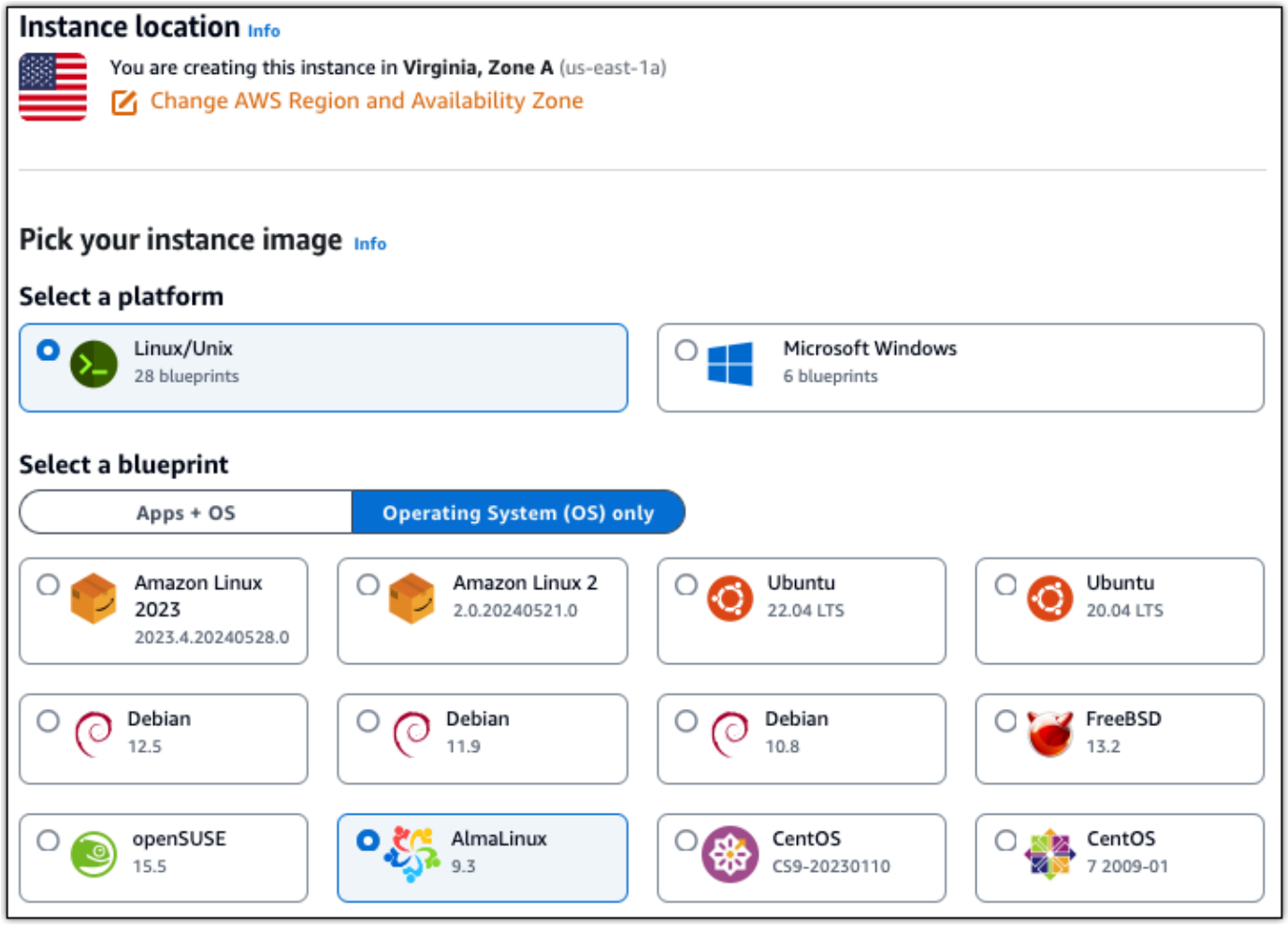
- 新規の AWS お客様は、Amazon Lightsail の使用を開始する前に、セットアップの前提条件を完了してください。詳細については、「Lightsail のセットアップ AWS アカウント と管理ユーザー」を参照してください。
- AlmaLinux Wiki サイトの AlmaLinux ドキュメントをお読みください。

Lightsail で AlmaLinux インスタンスを作成する

Lightsail コンソール を使用して AlmaLinux 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. 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 とアベイラビリティーゾーン) を選択します。レイテンシーを短縮するために、物理的な場所に最も AWS リージョン 近い を選択します。

   アベイラビリティーゾーンを変更 を選択して、別の場所にインスタンスを作成します。
4. Linux プラットフォームを選択します。
5. オペレーティングシステム (OS) のみを選択し、AlmaLinux設計図を選択します。



6. オプションで、次のことができます。

   a. 起動スクリプトの追加 を選択して、インスタンスが初めて起動したときに実行されるシェルスクリプトを追加します。詳細については、「Lightsail で起動スクリプトを使用して Linux/Unix インスタンスを設定する」を参照してください。

   b. SSH キーペアの変更 を選択して、インスタンスの SSH キーペアを変更します。詳細については、「Lightsail のSSHキーを設定する」を参照してください。

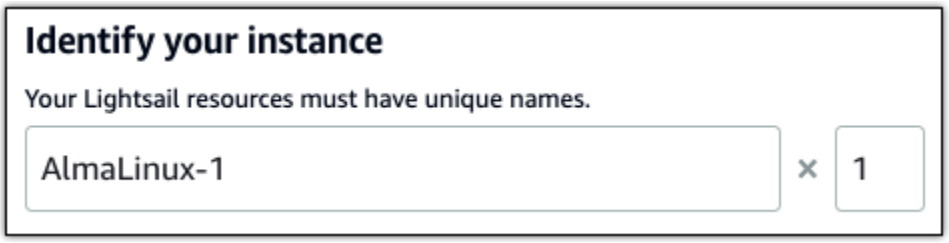
   c. 自動スナップショットを有効にする を選択して、インスタンスとアタッチされたディスクの自動スナップショットを有効にします。詳細については、「Lightsail インスタンスとディスクの自動スナップショットを設定する」を参照してください。

7. 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できます。 AlmaLinux ブループリントは、デュアルスタックと IPv6-onlyバンドルの両方をサポートします。IPv6-only「」を参照してくださいLightsail インスタンスの IPv6-onlyネットワークを設定する。

8. インスタンスの名前を入力します。

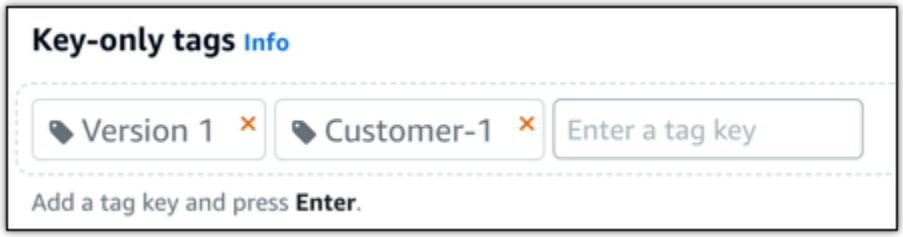
   リソース名:

   - 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。

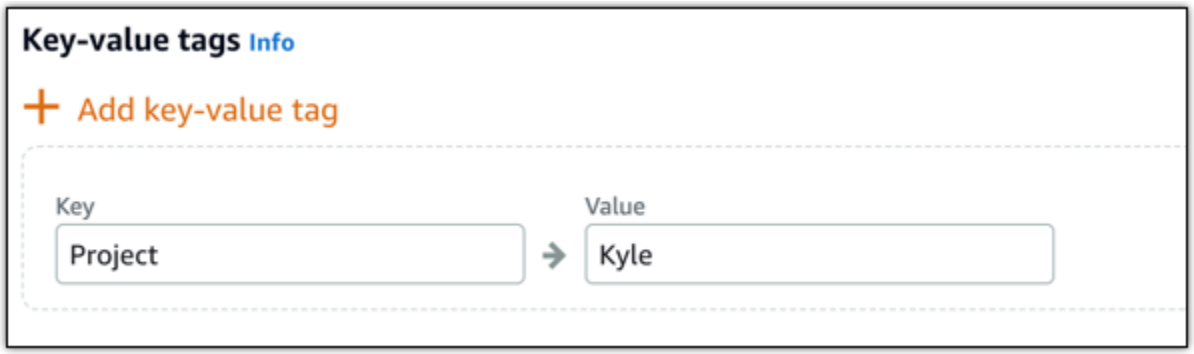


9. 以下のいずれかのオプションを選択して、インスタンスにタグを追加します。

   - [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。

   キーのみのタグとキー値タグの詳細については、「」を参照してください[タグを使用して Lightsail リソースを整理およびフィルタリングする](#)。

10. [インスタンスの作成] を選択します。

Lightsail インスタンスは数分以内に準備でき、接続できます。

(オプション) 追加セットアップ

以下は、 AlmaLinux インスタンスが Lightsail で起動して実行された後に開始するために実行する必要があるいくつかのステップです。

- 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および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ネットワークタブで、静的 IP 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 インスタンスに静的 IP を作成してアタッチする」を参照してください。

- Lightsail にドメインを登録し、Lightsail でドメイン名を管理します。Lightsail は、可用性が高くスケーラブルな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 を使用して、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るか、ドメインの DNS レコードを管理できます。詳細については、「Lightsail でウェブサイトのドメインを登録して管理する」を参照してください。

- ドメイン名をインスタンスにマッピングする – などのドメイン名をインスタンス`example.com`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 に転送して、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 セクションで、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 インスタンスのドメインレコードを管理するDNSゾーンを作成する」を参照してください。

- 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、システムディスクのコピーとインスタンスの元の設定です。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、「スナップショットを使用して Linux/Unix Lightsail インスタンスをバックアップする」を参照してください。

CentOS から に移行する方法については AlmaLinux、次のトピック に進みますLightsail AlmaLinux で CentOS から にデータを移行する。

Lightsail AlmaLinux で CentOS から にデータを移行する

CentOS から への移行 AlmaLinux は、Lightsail の 1 つのインスタンスから別のインスタンスにデータを移動する簡単なプロセスです。このトピックでは、データの移行に使用できる 2 つのオプションの概要を説明します。

詳細については、AlmaLinux Wiki サイトの AlmaLinux ドキュメントを参照してください。

目次

前提条件

- まだ作成していない場合は、Lightsail AlmaLinux インスタンスを作成します。詳細については、「Lightsail で AlmaLinux インスタンスを起動してセットアップする」を参照してください。
- AlmaLinux インスタンスに移動する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、「バックアップまたはベースライン用の Lightsail ブロックストレージディスクスナップショットを作成する」を参照してください。

(オプション) セキュアコピー (scp) を使用してインスタンス間でファイルを転送します。

Linux の Secure Copy コマンドを使用して、CentOS インスタンスから新しい AlmaLinuxインスタンスにファイルを安全に転送できます。詳細については、「scp を使用して Lightsail 上の Linux インスタンス間でファイルを転送する」を参照してください。

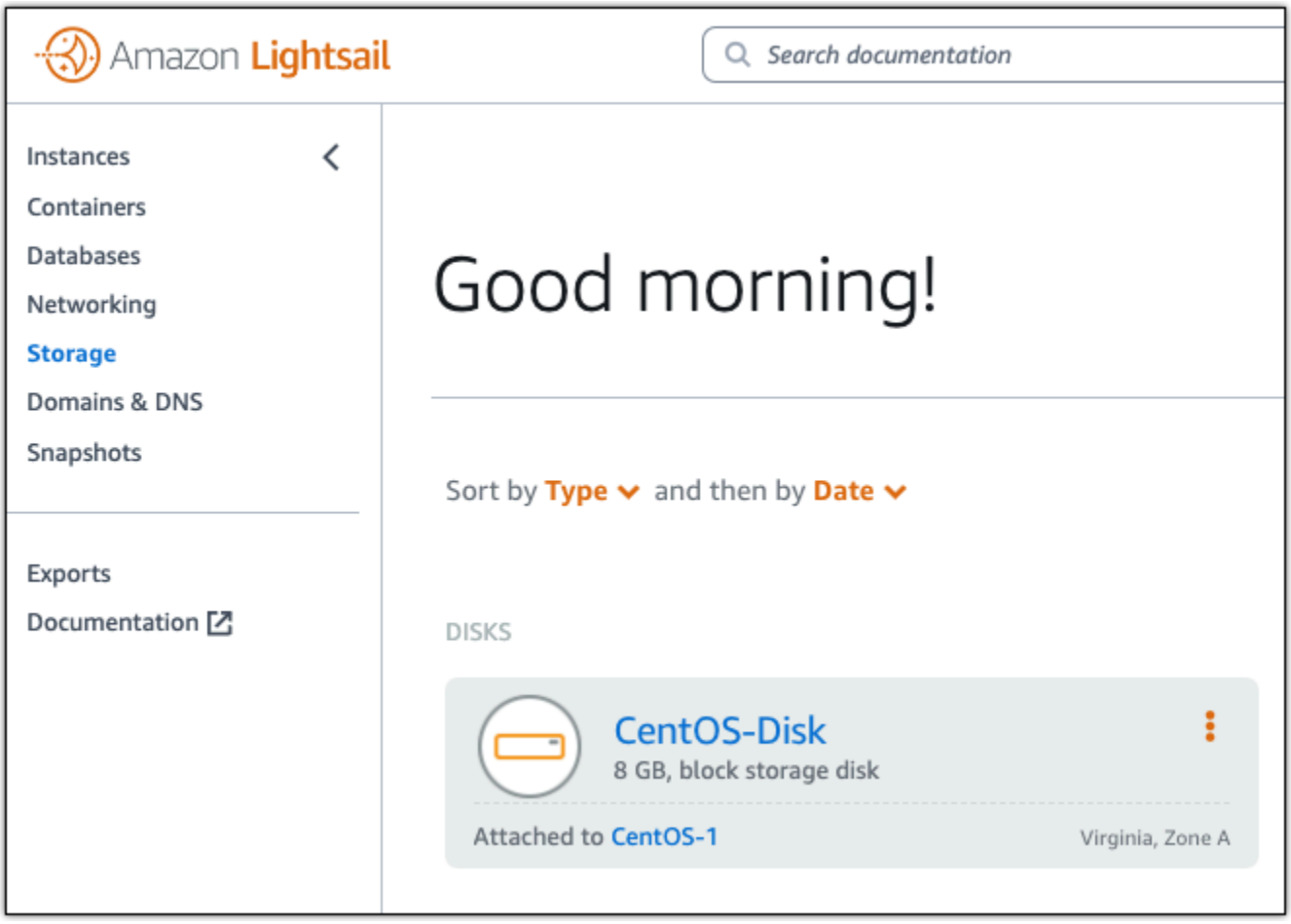
(オプション) ブロックストレージディスクを CentOS インスタンスからインスタンスに移動する AlmaLinux

セカンダリブロックストレージディスクを CentOS インスタンスバンドルから AlmaLinux バンドルに移動するには、次の手順に従います。インスタンスのブートボリュームディスク、オペレーティングシステムを含むディスクをデタッチすることはできません。ディスクを AlmaLinux インスタンス

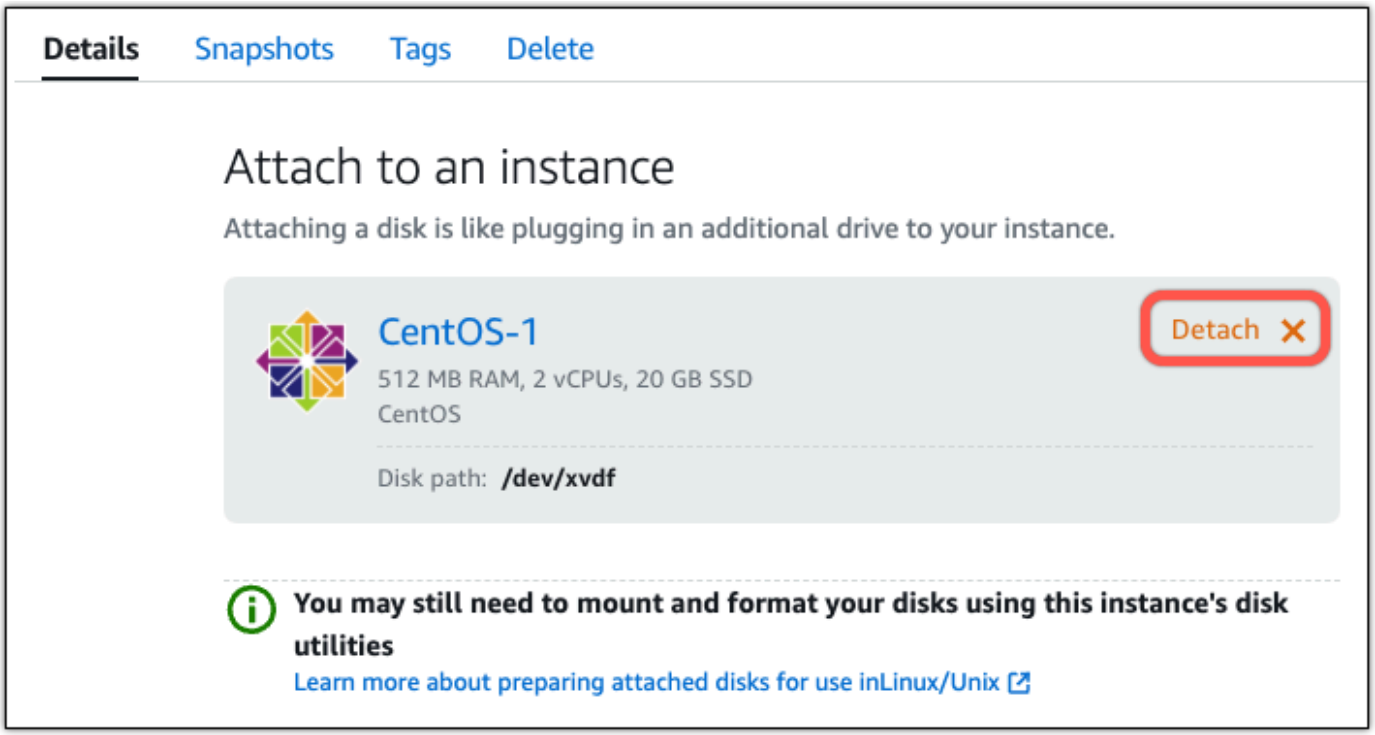
にアタッチしたら、そのインスタンスに接続してディスクをマウントする必要があります。詳細については、「Lightsail ブロックストレージディスクでストレージとパフォーマンスを拡張する」を参照してください。

CentOS インスタンスが実行されている場合は、ディスクをデタッチする前にインスタンスを停止する必要があります。詳細については、「実行中のインスタンスを停止する」を参照してください。

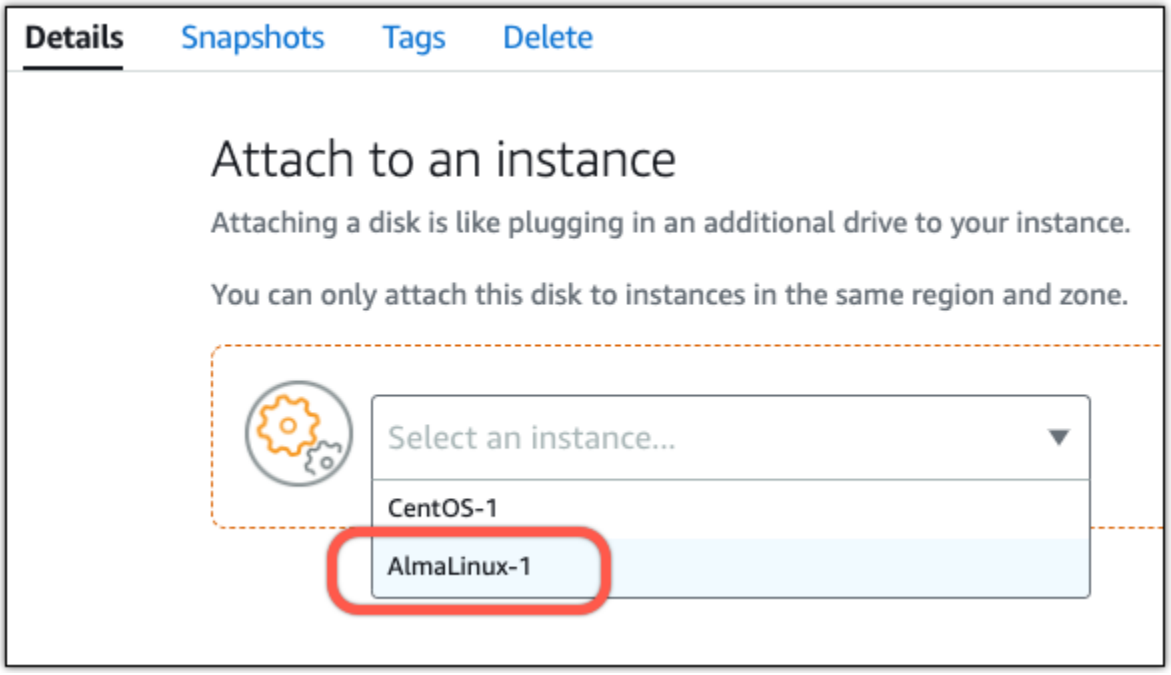
1. Lightsail コンソールのストレージセクションで、CentOS インスタンスからデタッチするディスクを選択します。



2. 詳細 タブで、 をデタッチ を選択します。



3. ディスクの詳細ページから、インスタンスにアタッチドロップダウンメニューを選択します。次に、 AlmaLinux インスタンスの名前を選択します。



4. 添付を選択します。
5. (オプション) データにアクセスする前に、 AlmaLinux インスタンスに接続してディスクをマウントする必要がある場合があります。詳細については、「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」を参照してください。

> **Warning**
>
> 上記のリンクは、アタッチされたディスクをマウントしてフォーマットする方法を示しています。 AlmaLinux インスタンスにアタッチしたディスクをフォーマットしないでください。フォーマットすると、ディスクに保存されているすべての情報が永続的に消去されます。

Lightsail で cPanel & WHM を使用してウェブサイト、E メール、サービスをホストする

cPanel & WHM インスタンスが Amazon Lightsail で起動および実行された後に開始するために実行する必要があるいくつかのステップを次に示します。

> **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ときに静的 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。

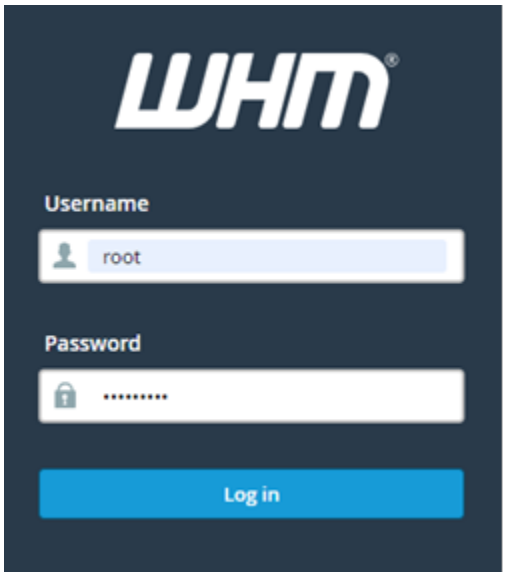
   ポート 2087 経由でインスタンスの静的 IP アドレスにブラウジングする際に、接続を確立できなかった場合は、ルーター、VPN、またはインターネットサービスプロバイダーがポート 2087 経由の HTTP/HTTPS 接続を許可しているかを確認してください。許可していない場合は、別のネットワークを使用して接続を試みてください。

   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

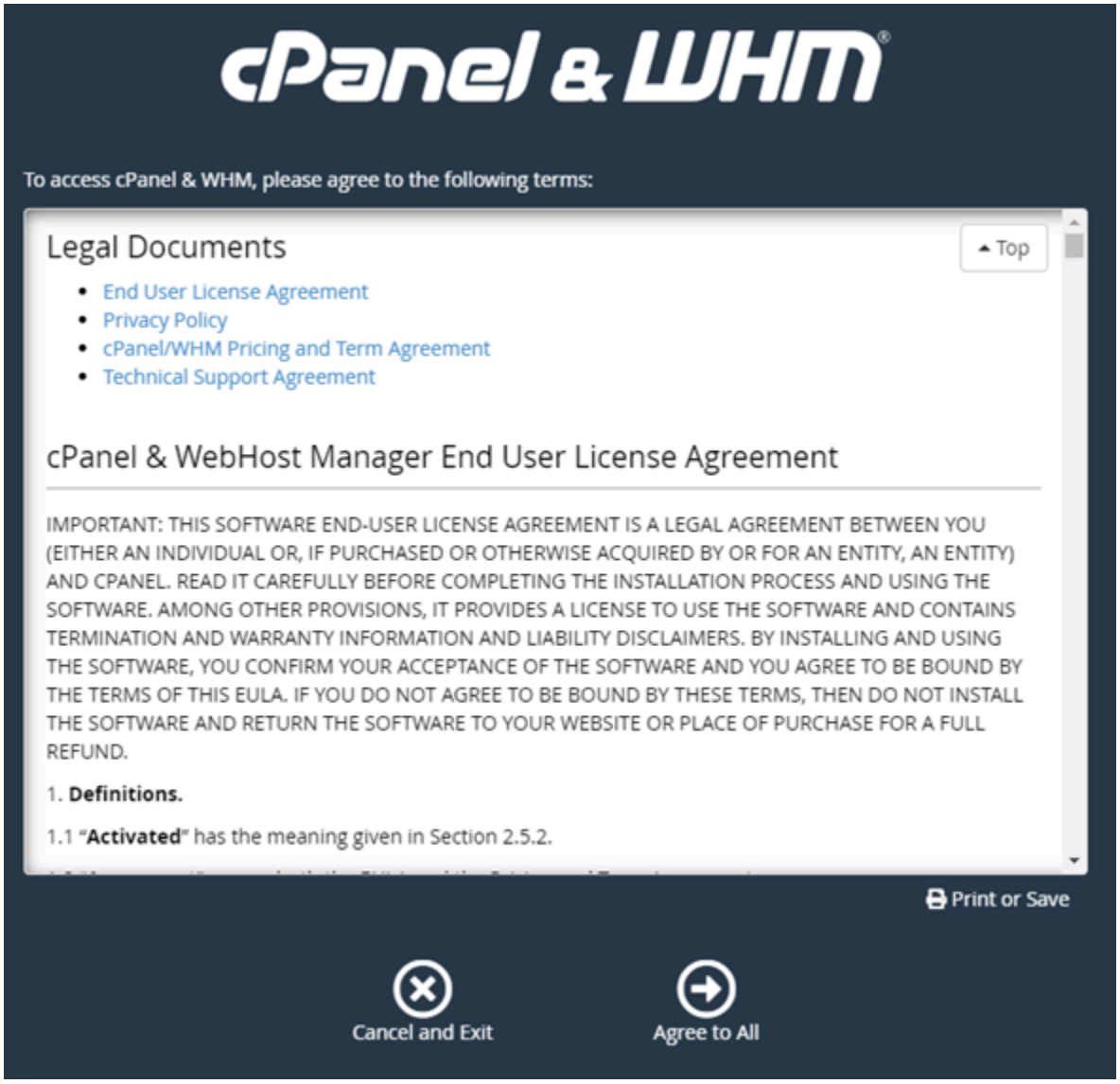
がまだ cPanel インスタンスに適用されていない場合に発生します。ブラウザウィンドウで、[Advanced] (詳細設定)、[Details] (詳細)、または [More information] (詳細情報) を選択して、使用可能なオプションを表示します。次に、プライベートまたは安全でない場合でも、ウェブサイトにアクセスすることを選択します。

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。

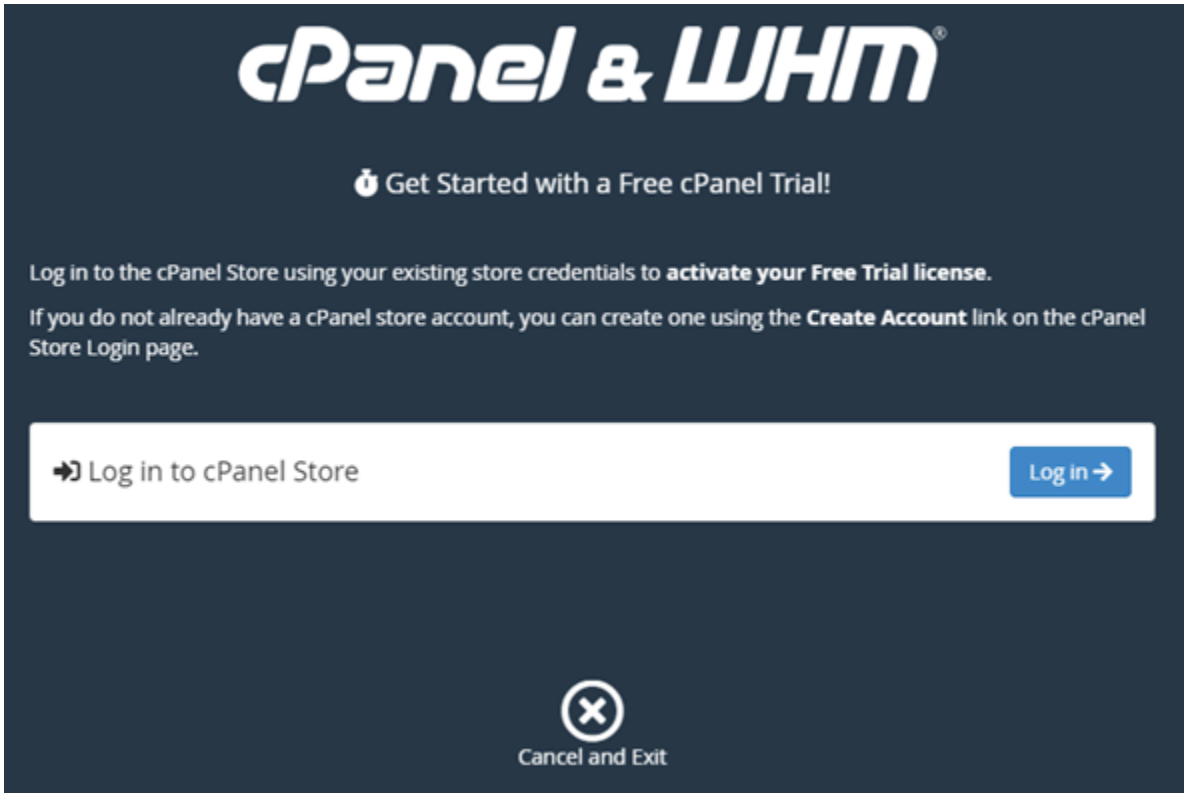


5. 続行する場合は、cPanel & WHM 規約を読んで、「すべてに同意する」を選択します。



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。

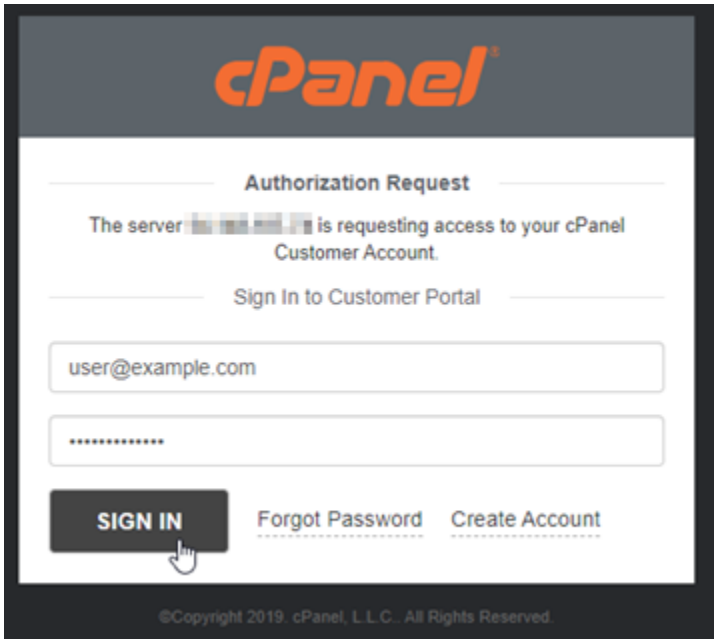
   ユーザーのアカウントにトライアルライセンスを関連付けるには、cPanel ストアにサインインする必要があります。cPanelストアのアカウントをお持ちでない場合も ログイン を選択します。アカウント作成のオプションが表示されます。



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。

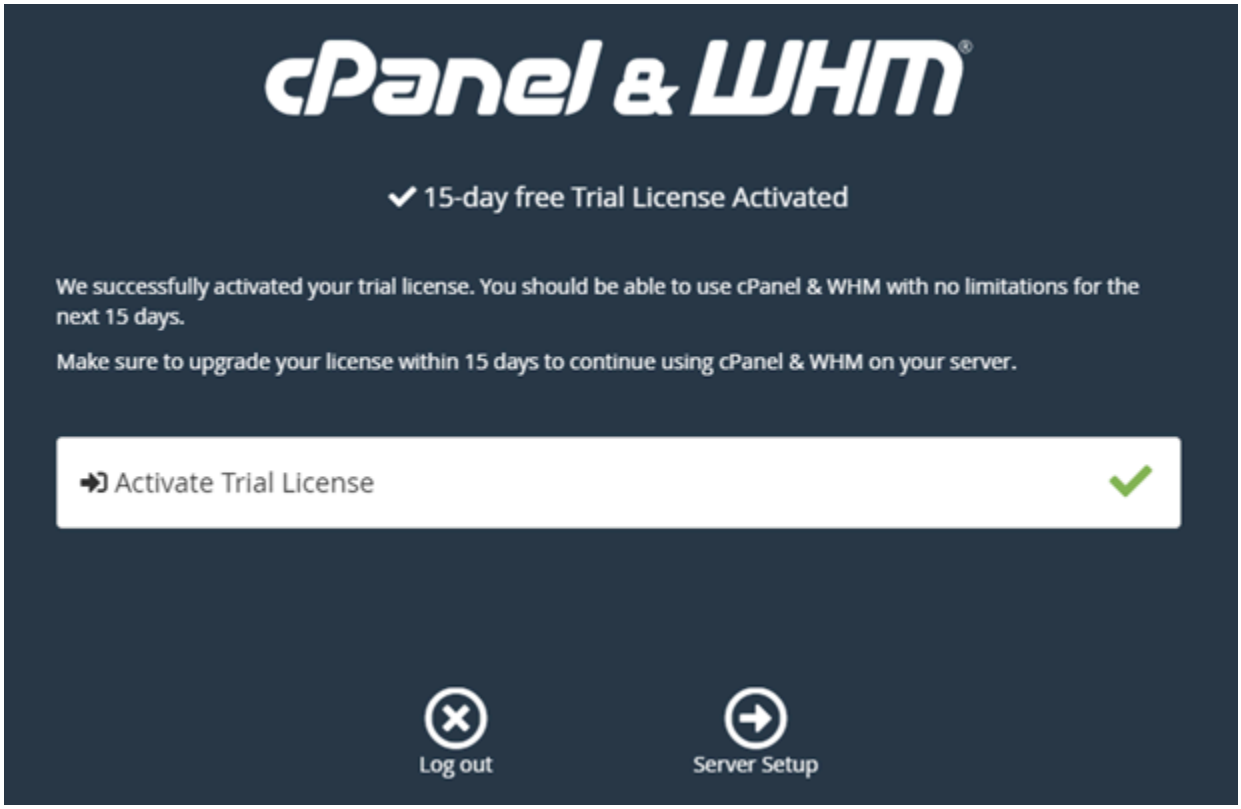
   cPanel ストアのアカウントをお持ちでない場合は、[アカウントの作成] を選択し、プロンプトに従って新しい cPanel ストアのアカウントを作成します。メールアドレスを入力するように求められます。cPanelストアアカウントのパスワードを設定するためのメールが送信されます。新しいブラウザを使用して cPanel ストアアカウントのパスワード設定を行うことをお勧めします。パスワードが設定されたらタブを閉じて、インスタンスに戻ってアカウント認証を行い、この手順の次のステップに進みます。

8. [Sign in] (サインイン) を選択します。

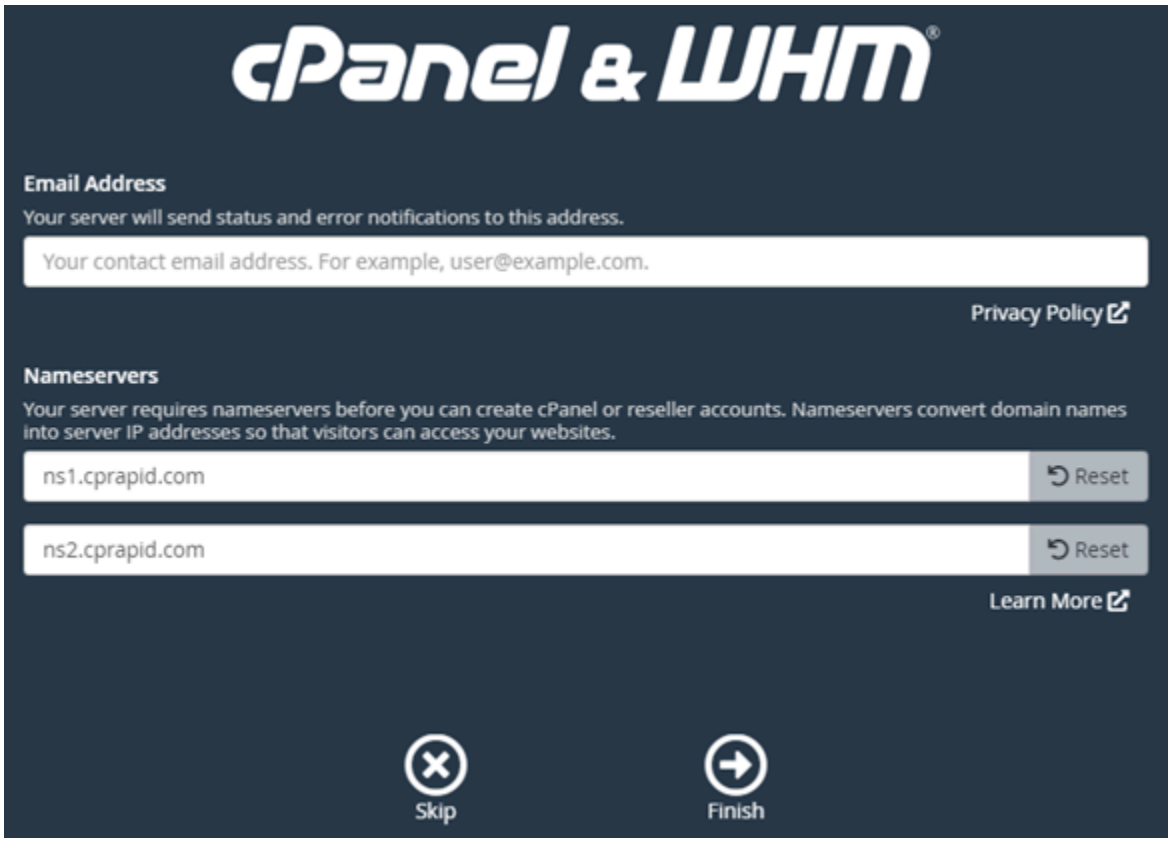


サインイン後、cPanel & WHM インスタンスは 15 日間のトライアルライセンスを取得します。これはユーザーの cPanel ストアのアカウントに関連付けられています。cPanel ストアの [ライセンスの管理] ページに移動して、トライアルライセンスを含む発行されたライセンスを確認します。

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。

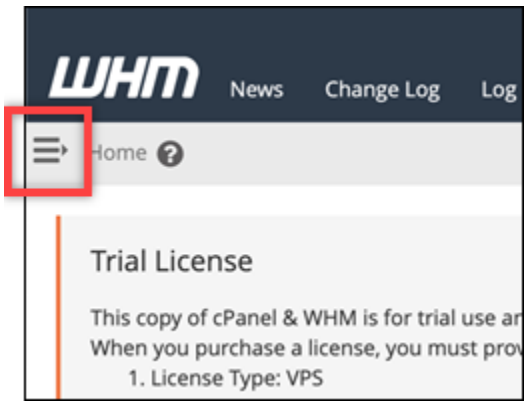


cPanel の設定と機能を管理することができる WHM コンソールが表示されます。

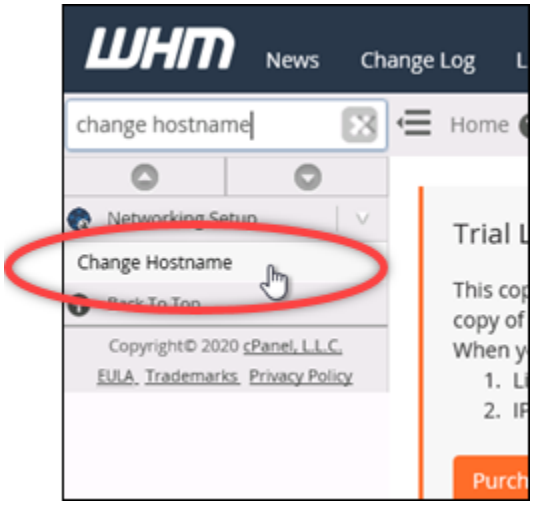
ステップ4：cPanel＆WHM インスタンスのホスト名と IPアドレスを変更する

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。

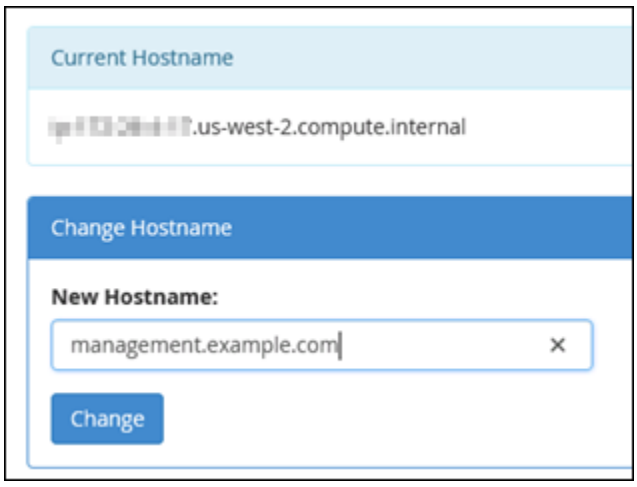


2. WHM コンソールの検索テキストボックスに Change hostname を入力して、検索結果の [ホスト名を変更] のオプションを選択します。

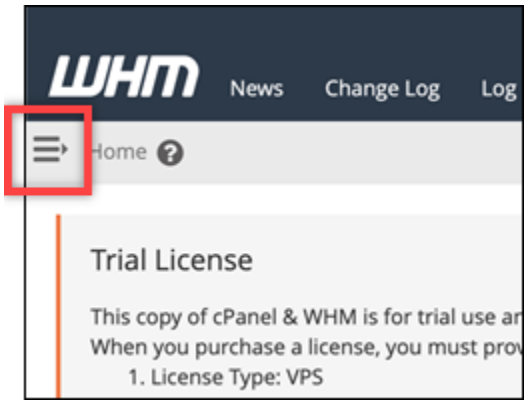


3. WHM コンソールへのアクセスに使用したいホスト名を、[新しいホスト名] テキストボックスに入力します。たとえば、management.example.com ないし administration.example.com を入力します。

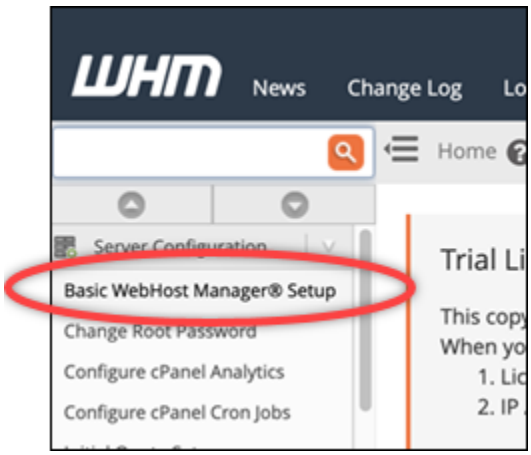
> **Note**
>
> サブドメインは、ホスト名としてのみ指定できます。whm や cpanel はサブドメインとして指定できません。



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。

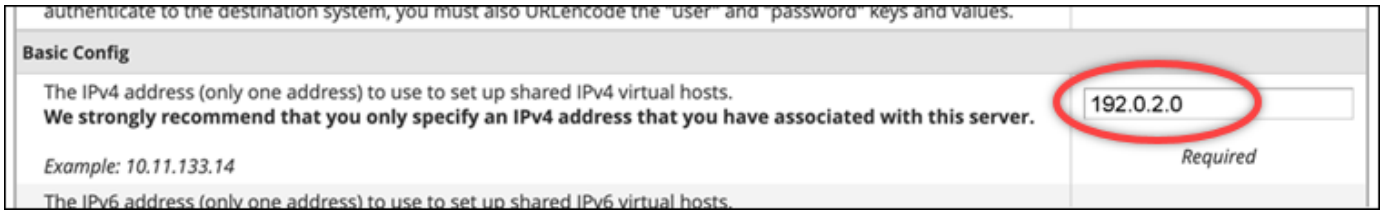


6. Basic WebHost Manager Setup を選択します。



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、ドメインの DNS レコードの管理を Lightsail に転送して、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。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」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 インスタンスでポート 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 の制限を削除する方法を教えてください。」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 コンソールでインスタンスのファイアウォールでこ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ときに静的 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**
>
> - の現行世代のブループリント cPanel & WHM AlmaLinux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」を参照してください。

Lightsail で Drupal ウェブサイトを設定およびカスタマイズする

Drupal インスタンスが Amazon Lightsail で起動および実行された後に開始するには、いくつかのステップを実行する必要があります。

目次

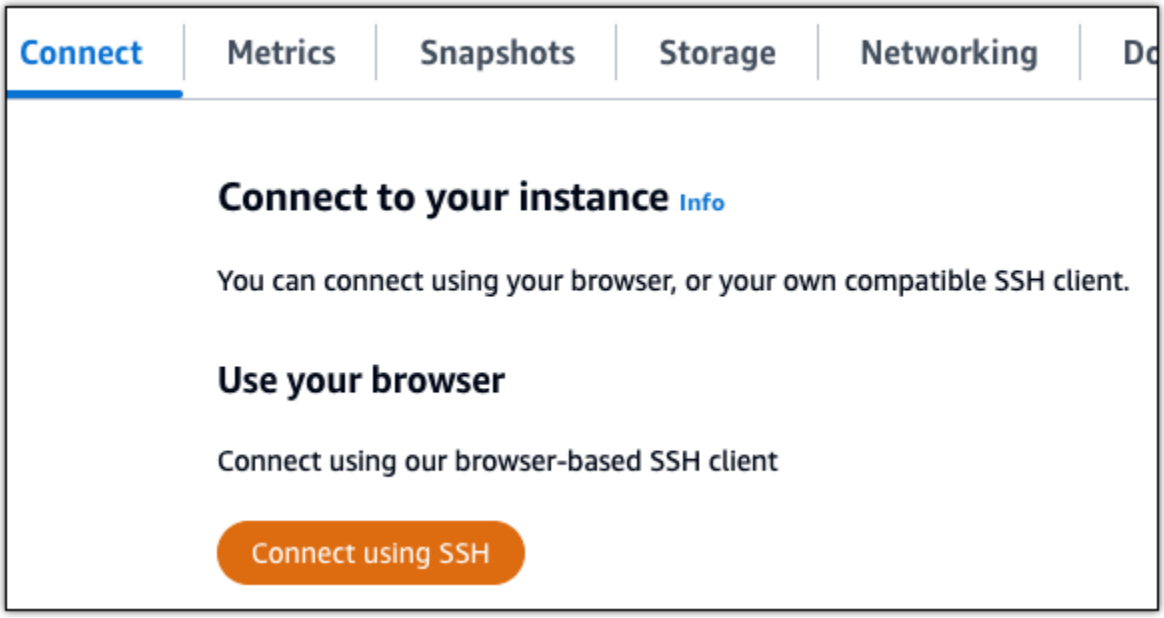
ステップ 1: Bitnami のドキュメントを確認する

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」を参照してください。

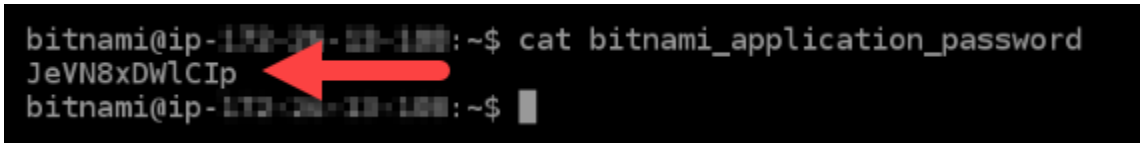
1. インスタンス管理ページの [接続] タブで、[SSH を使用して接続] を選択します。



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
```

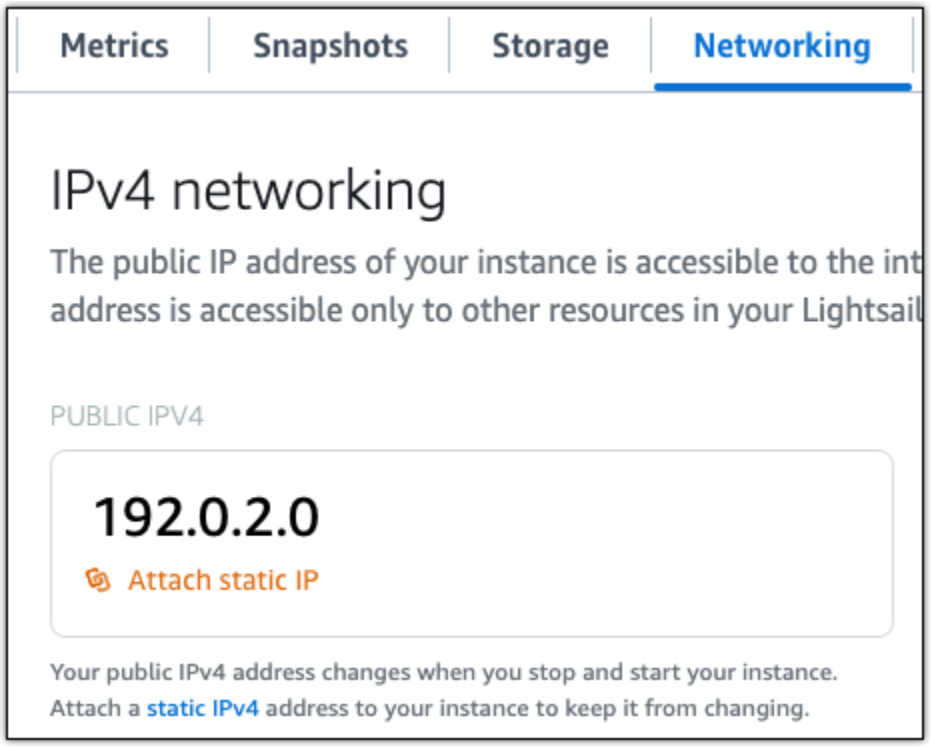
アプリケーションのデフォルトパスワードを含んだ、次の例のようなレスポンスが表示されます。



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)。

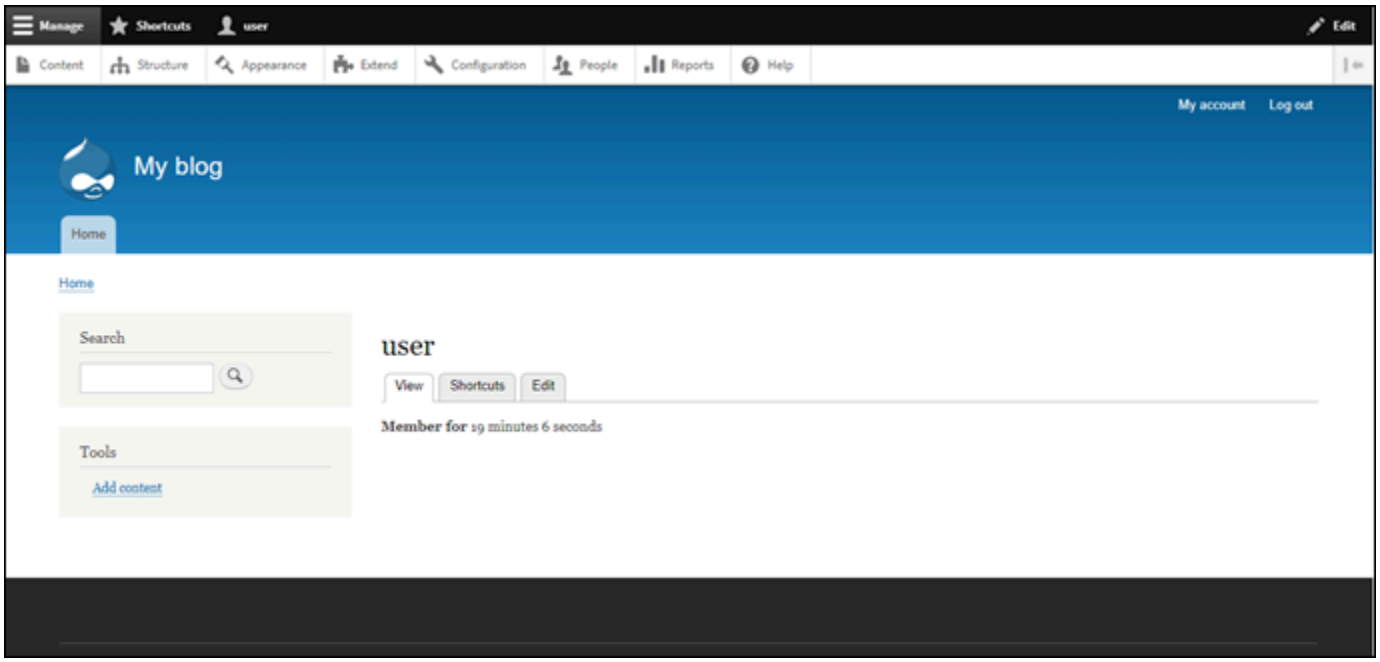
   Drupal ウェブサイトのホームページが表示されます。

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 に転送して、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 DNS ゾーンの作成」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

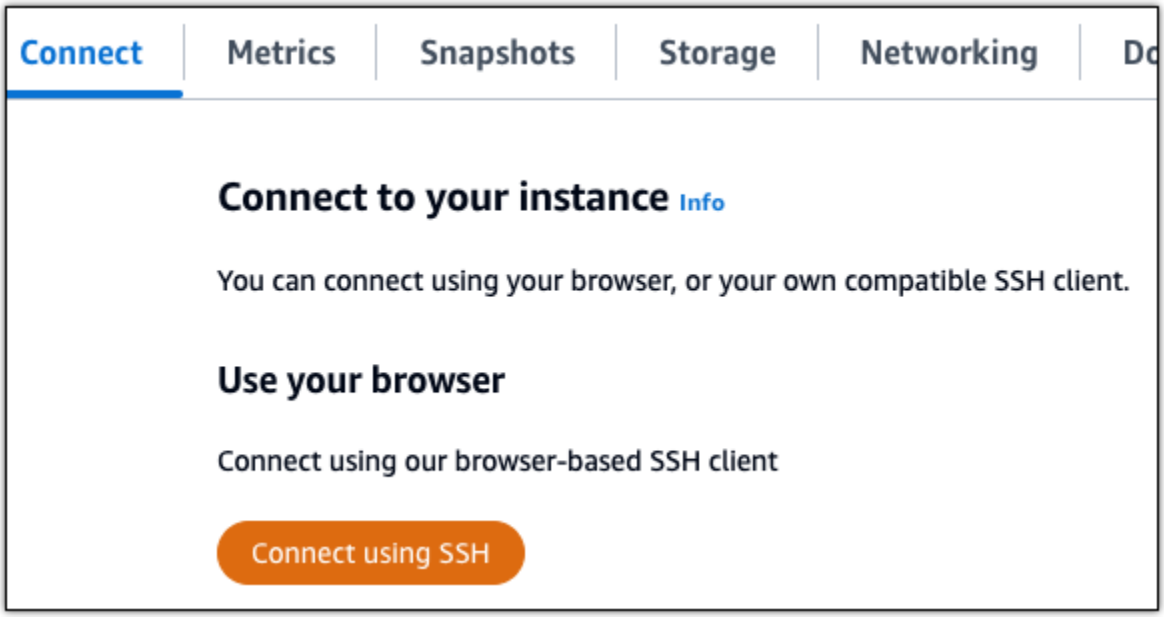
す。詳細については、本ガイドの次の「ステップ 6: Drupal ウェブサイトの HTTPS を設定する」のセクションを参照してください。

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
```

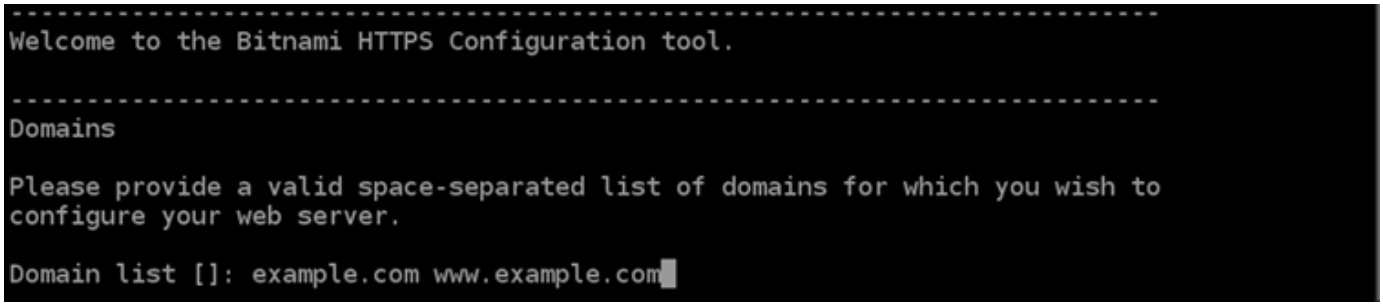
これでインスタンスに bncert ツールをインストールする手順は完了です。

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。



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、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。

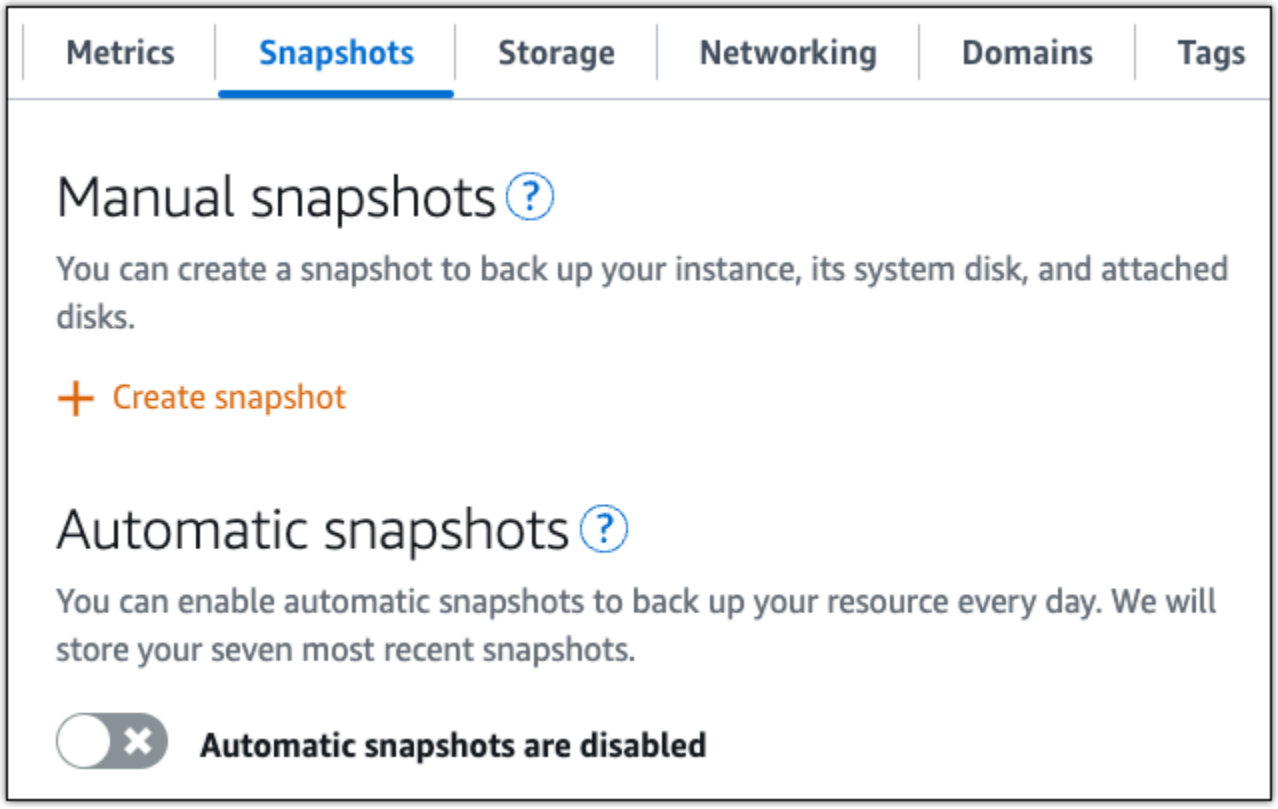
ステップ 7: Drupal のドキュメントを読み、引き続きウェブサイトの設定を続行する

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。

Lightsail に Ghost ウェブサイトをデプロイする

Ghost 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に実行すべきいくつかのステップを次に示します。

目次

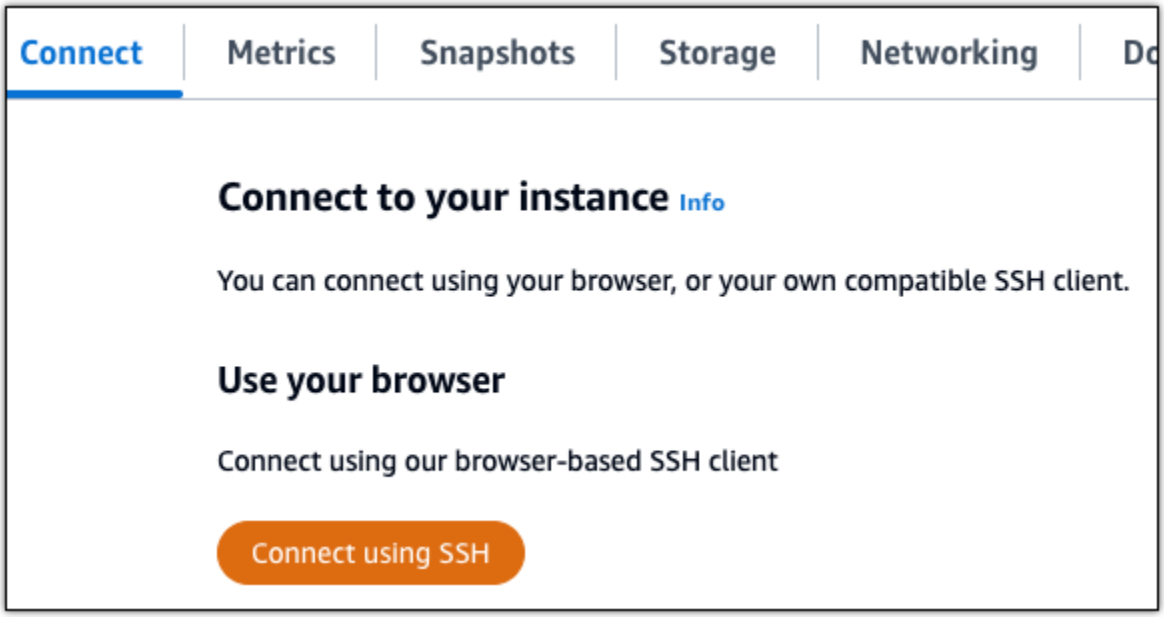
ステップ 1: Bitnami のドキュメントを確認する

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」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
```

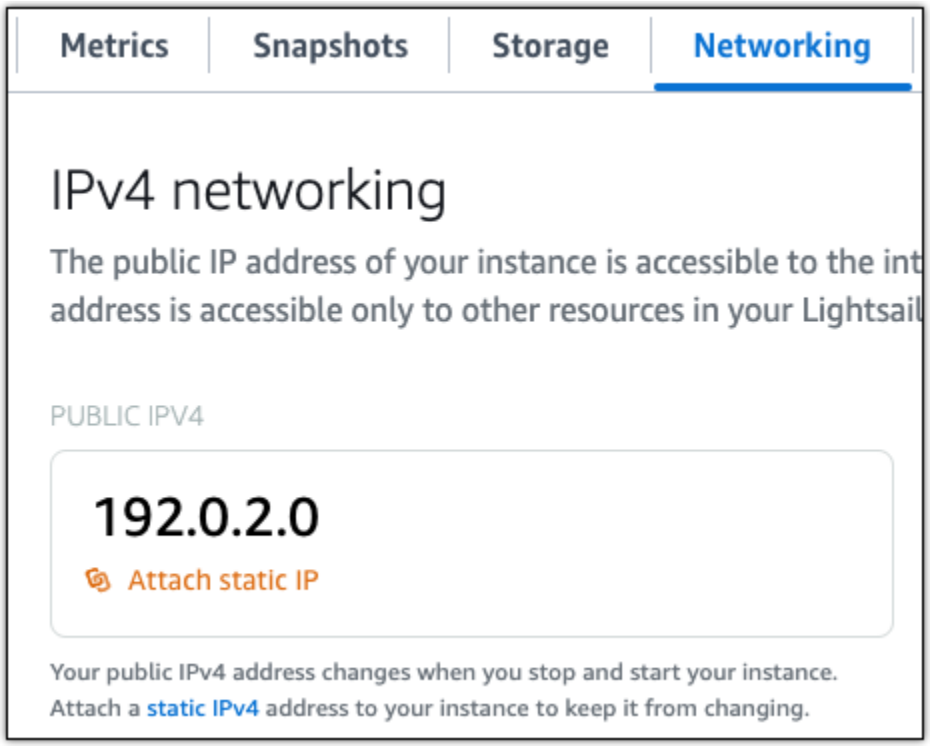
デフォルトのアプリケーションパスワードを含む次のようなレスポンスが表示されます。

```
bitnami@ip-192-0-2-0:~$ cat $HOME/bitnami_application_password
wB2Ex@mplEK6
```

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。

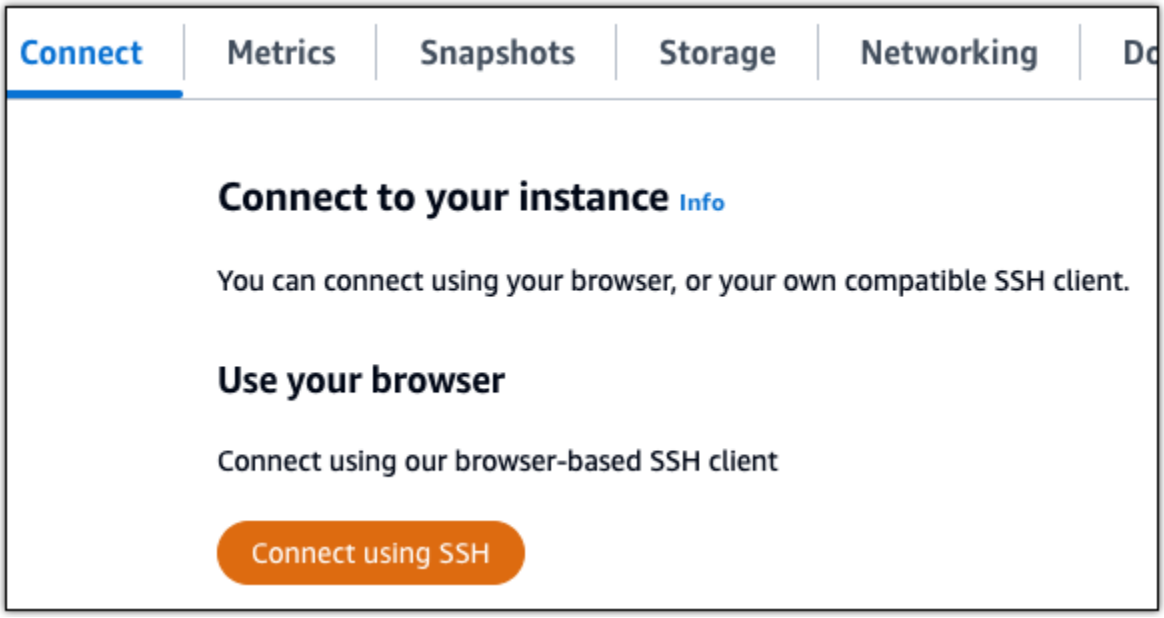
インスタンス管理ページの [ネットワーク] タブで、[静的 IP の作成] または [静的 IP のアタッチ] (インスタンスにアタッチできる静的 IP を既に作成している場合) を選択して、ページの手順に従います。詳細については、「静的 IP を作成してインスタンスにアタッチする」を参照してください。



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-203.0.113.0:~$ sudo /opt/bitnami/configure_app_domain --domain
 203.0.113.0
Configuring domain to 203.0.113.0
2024-06-06T21:43:42.393Z - info: Saving configuration info to disk
ghost 21:43:42.78 INFO  ==> Configuring Ghost URL to http://203.0.113.0
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。以前にインスタンスに静的 IP をアタッチしたことがある場合、これは静的 IP アドレスになります。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)。

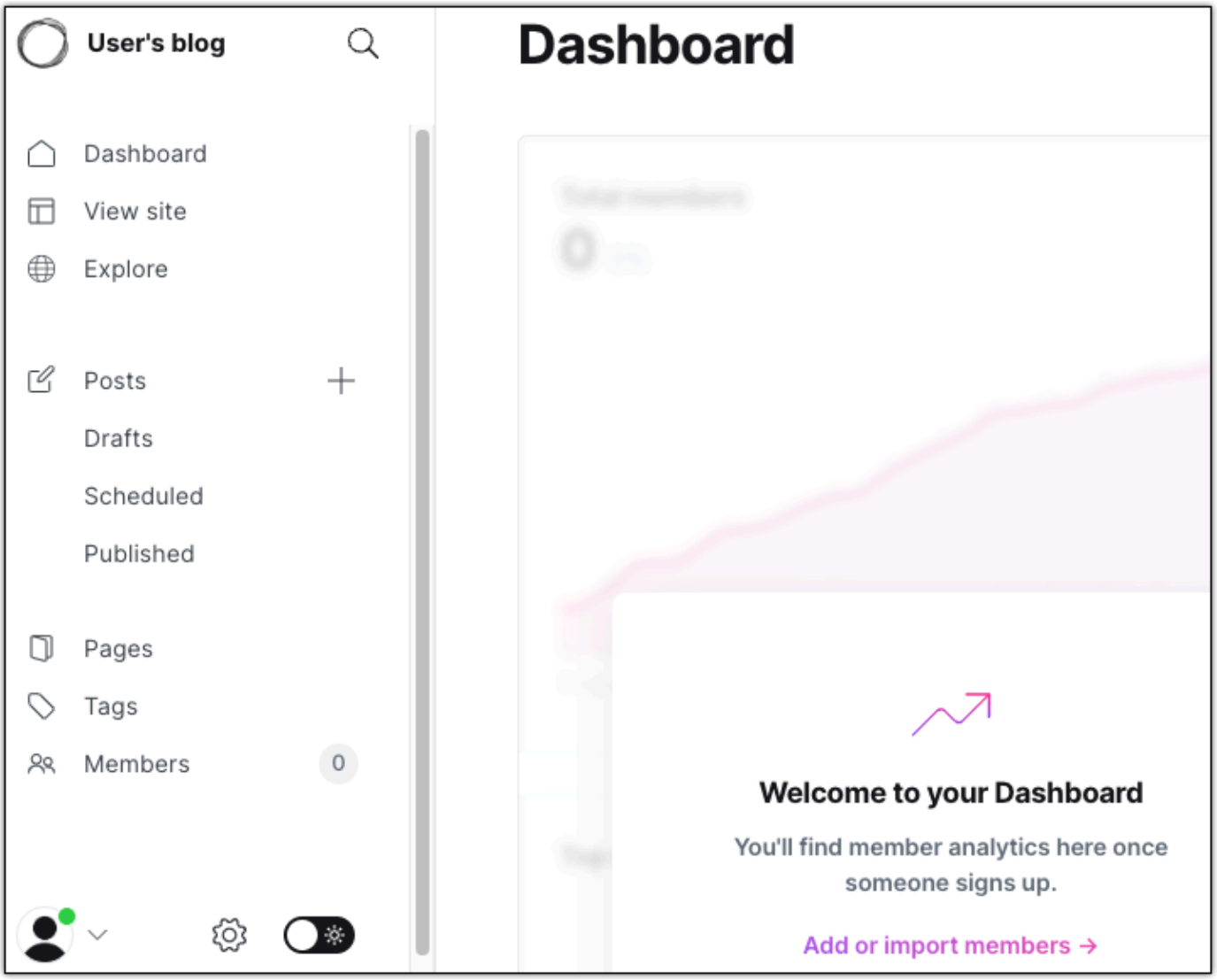
   Ghost ウェブサイトのホームページが表示されます。

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 に転送して、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」セクションで「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 DNS ゾーンの作成」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#* を、トラフィックを Ghost インスタンスに転送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、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-203.0.113.0:~$ sudo /opt/bitnami/configure_app_domain --domain
 example.com
Configuring domain to example.com
2024-06-06T21:50:00.393Z - info: Saving configuration info to disk
ghost 21:50:25.78 INFO  ==> Configuring Ghost URL to http://example.com
Disabling automatic domain update for IP address changes
```

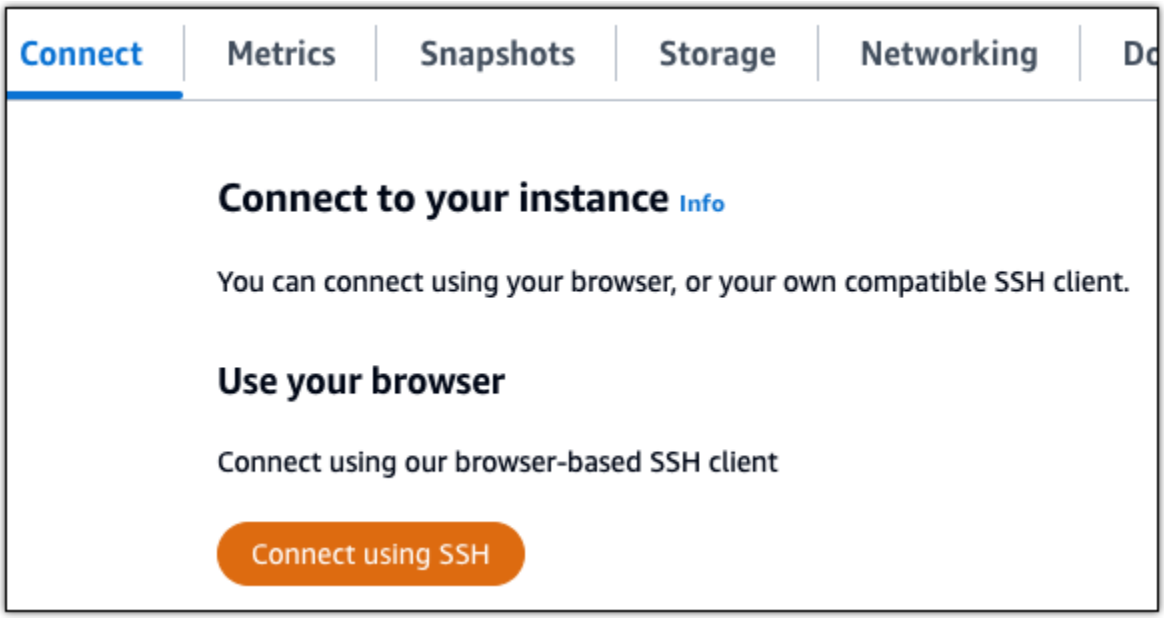
インスタンスに設定したドメイン名を参照すると、Ghost ウェブサイトのホームページへと移動します。次に、SSL/TLS 証明書を生成して設定し、Ghost ウェブサイトの HTTPS 接続を有効にします。詳細については、本ガイドの次の「ステップ 6: Ghost ウェブサイトの HTTPS を設定する」のセクションを参照してください。

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。

```
--------------------------------------------------------------------------------
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
```

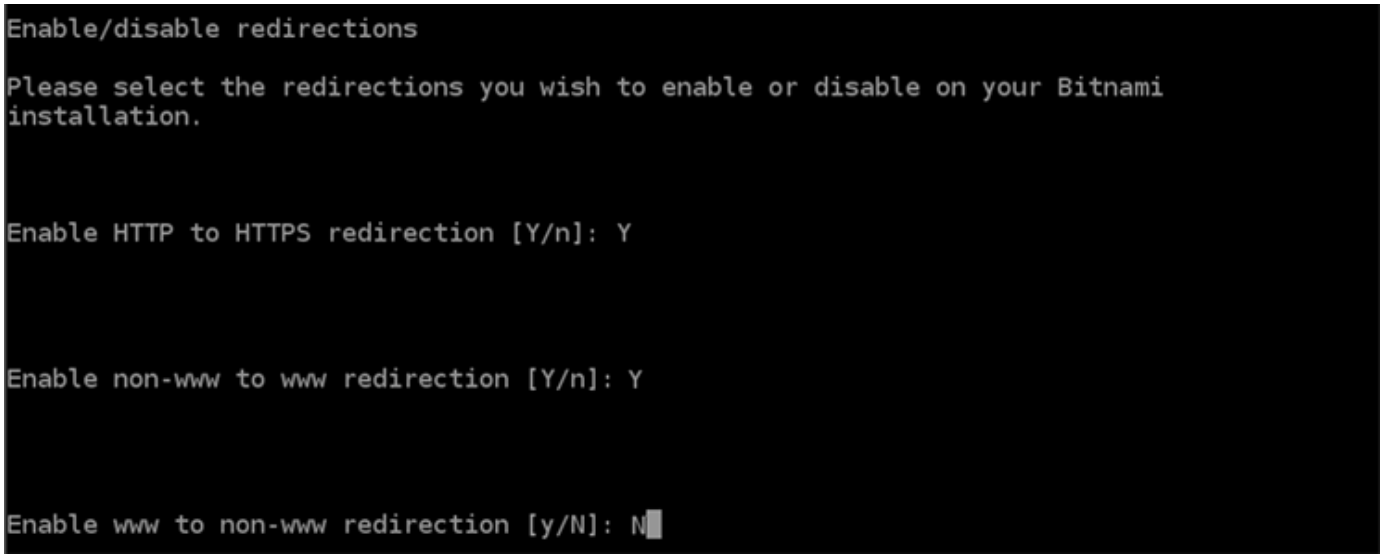
9. bncert ツールは、ウェブサイトのリダイレクトの設定方法を尋ねます。使用できるオプションは次のとおりです。

- 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。



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。

> **Tip**
>
> 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
> ```

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。

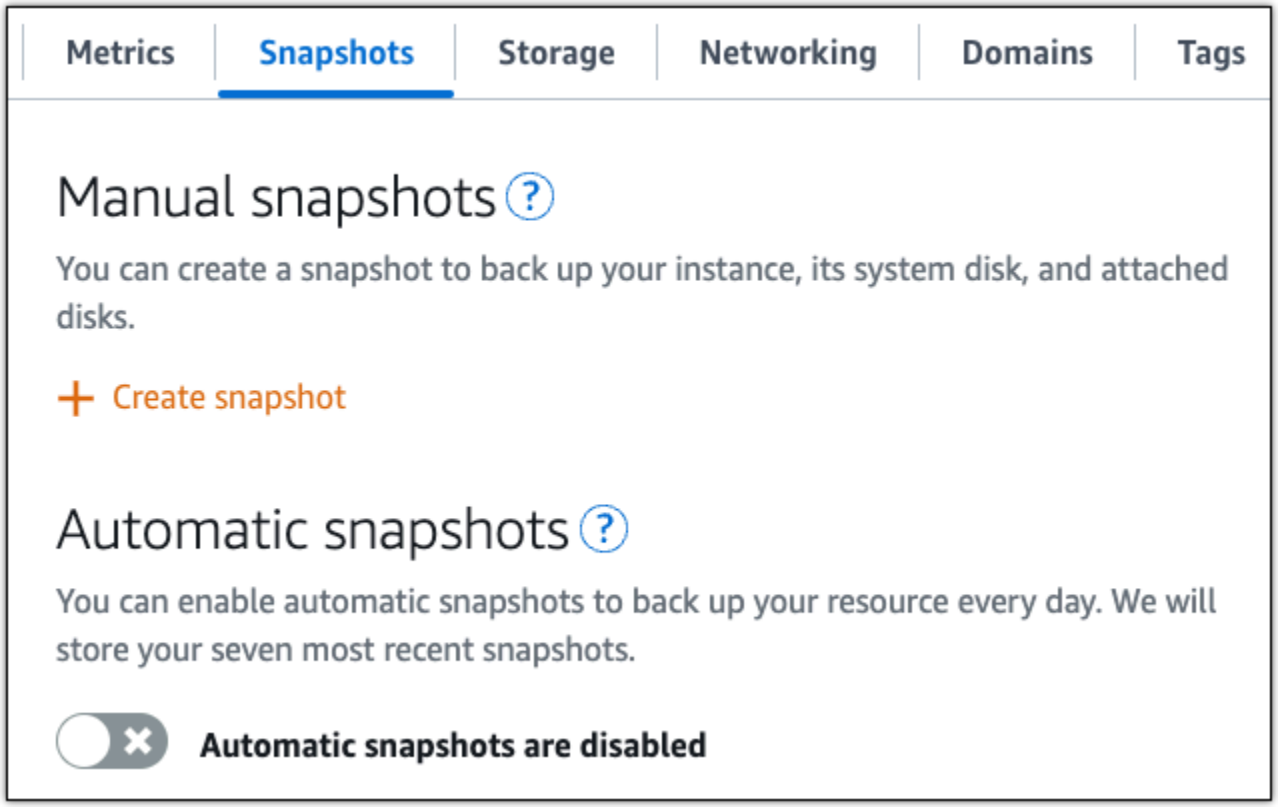
ステップ 7: Ghost のドキュメントを読み、引き続きウェブサイトの設定を続行する

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。

Lightsail で GitLab CE インスタンスをセットアップおよび設定する

GitLab CE インスタンスが Amazon Lightsail で起動および実行された後に開始するために実行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

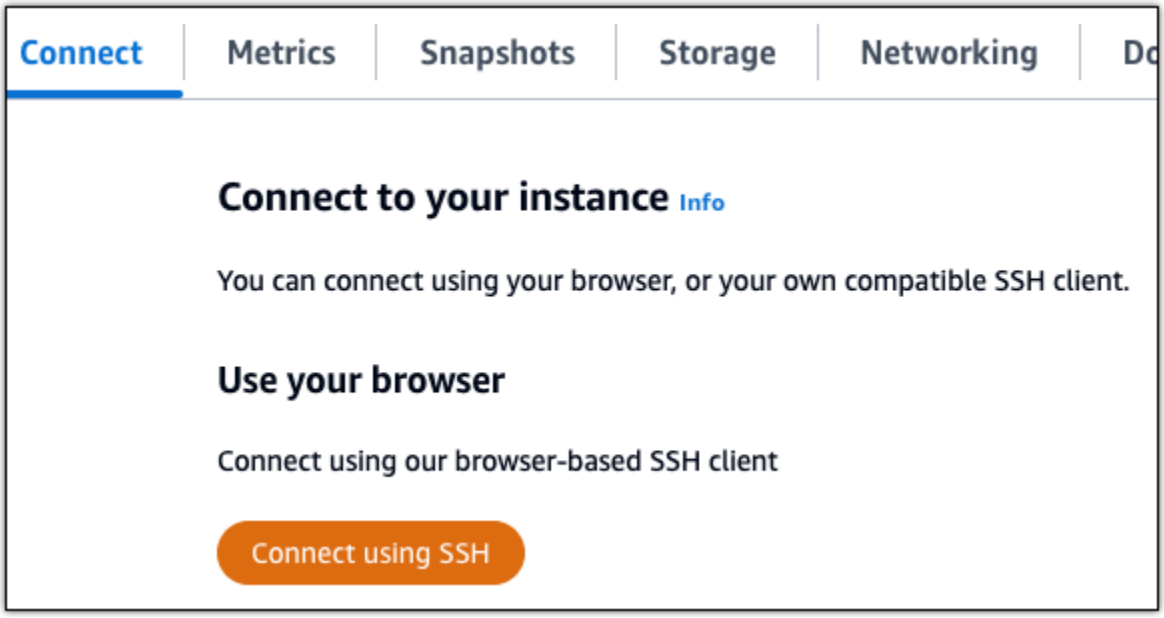
ステップ 1: Bitnami のドキュメントを確認する

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、 の GitLab Bitnami によってパッケージ化された CE AWS クラウドを参照してください。

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

GitLab CE ウェブサイトの管理領域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」を参照してください。

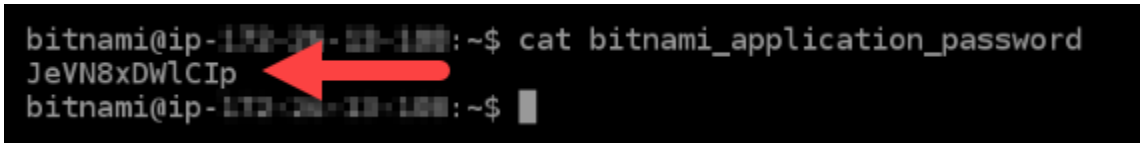
1. インスタンス管理ページの接続タブで、 を使用して接続を選択しますSSH。



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
```

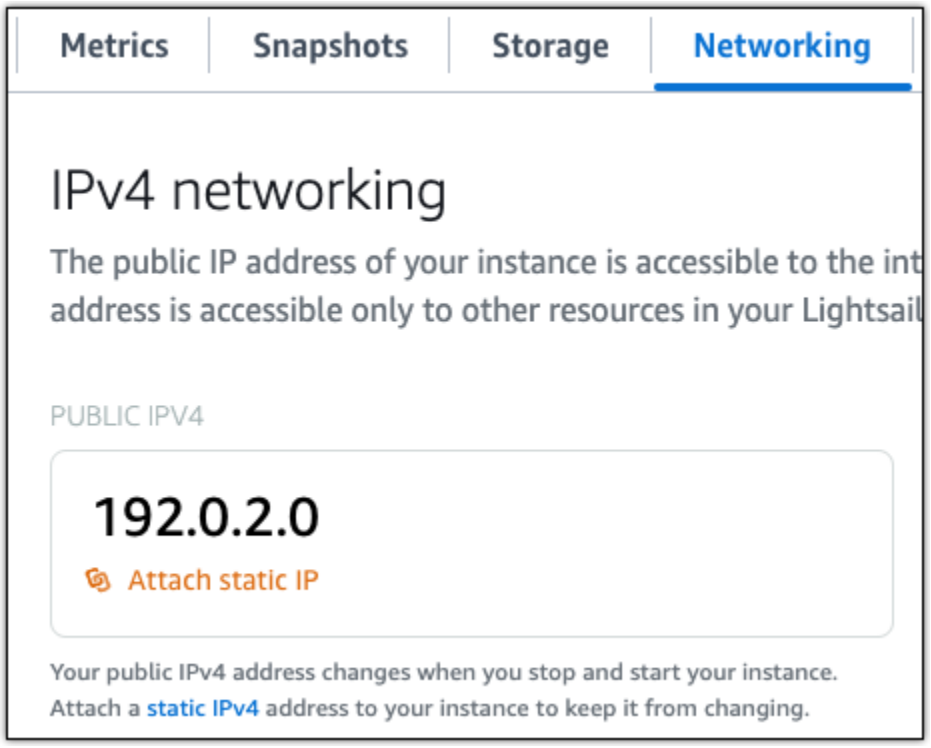
アプリケーションのデフォルトパスワードを含んだ、次の例のようなレスポンスが表示されます。



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。後で、インスタンスexample.comでなどの登録済みドメイン名を使用する場合、インスタンスを停止および起動するたびにドメインのDNSレコードを更新する必要はありません。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。

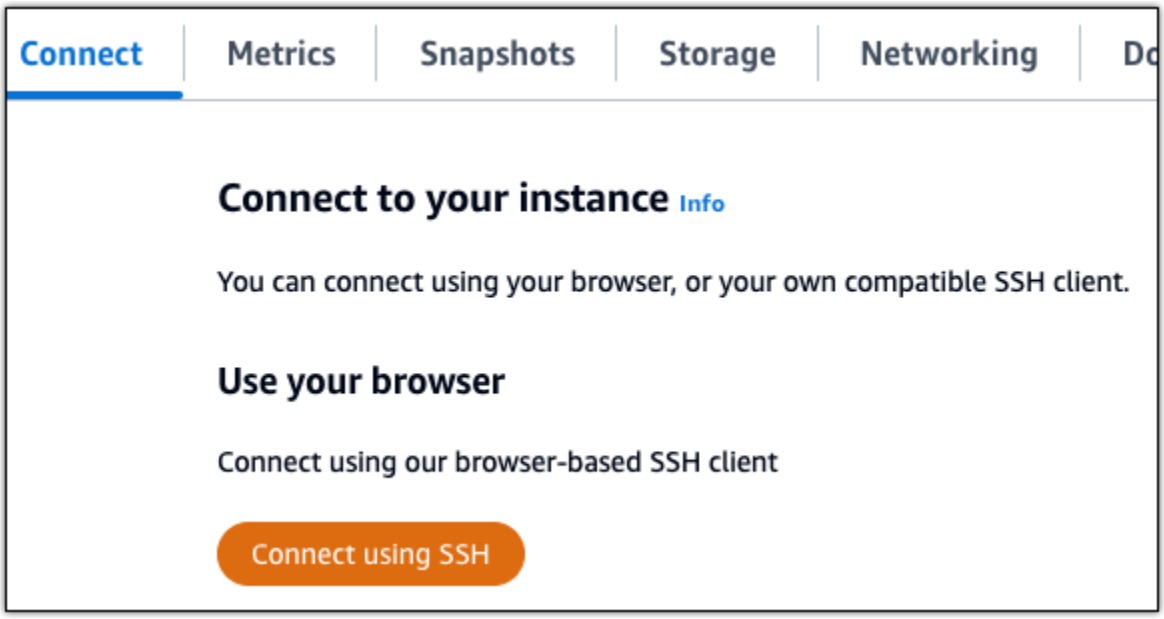
インスタンス管理ページの [ネットワーク] タブで、[静的 IP の作成] または [静的 IP のアタッチ] (インスタンスにアタッチできる静的 IP を既に作成している場合) を選択して、ページの手順に従います。詳細については、「静的 IP を作成してインスタンスにアタッチする」を参照してください。



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接続タブで、 を使用して接続を選択しますSSH。



3. 接続後に、次のコマンドを入力します。置換 *<StaticIP>* インスタンスの新しい静的 IP アドレス。

```
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
2022-06-09T16:47:06.737Z - info: Saving configuration info to disk
gitlab 16:47:06.86 INFO  ==> Updating external URL in GitLab configuration
gitlab 16:47:06.88 INFO  ==> Reconfiguring GitLab
gitlab 16:47:45.29 INFO  ==> Starting GitLab services
Disabling automatic domain update for IP address changes
```

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、管理者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、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。

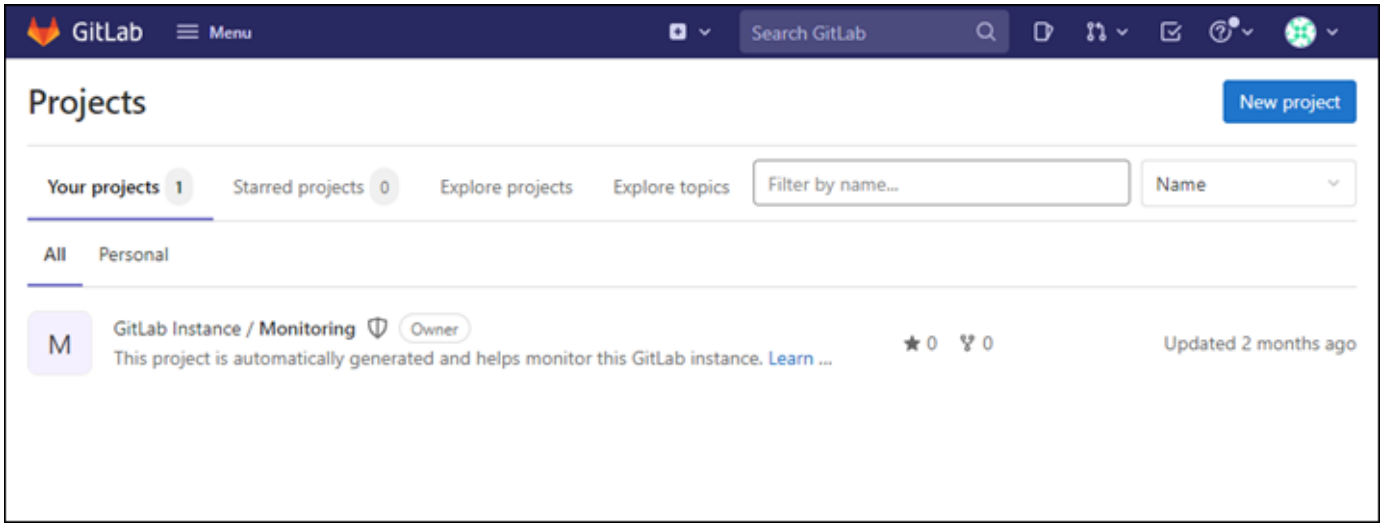
1. インスタンス管理ページの [Connect] (接続) タブにあるパブリック IP アドレスを書き留めます。パブリック IP アドレスは、インスタンス管理ページのヘッダーセクションにも表示されます。

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。

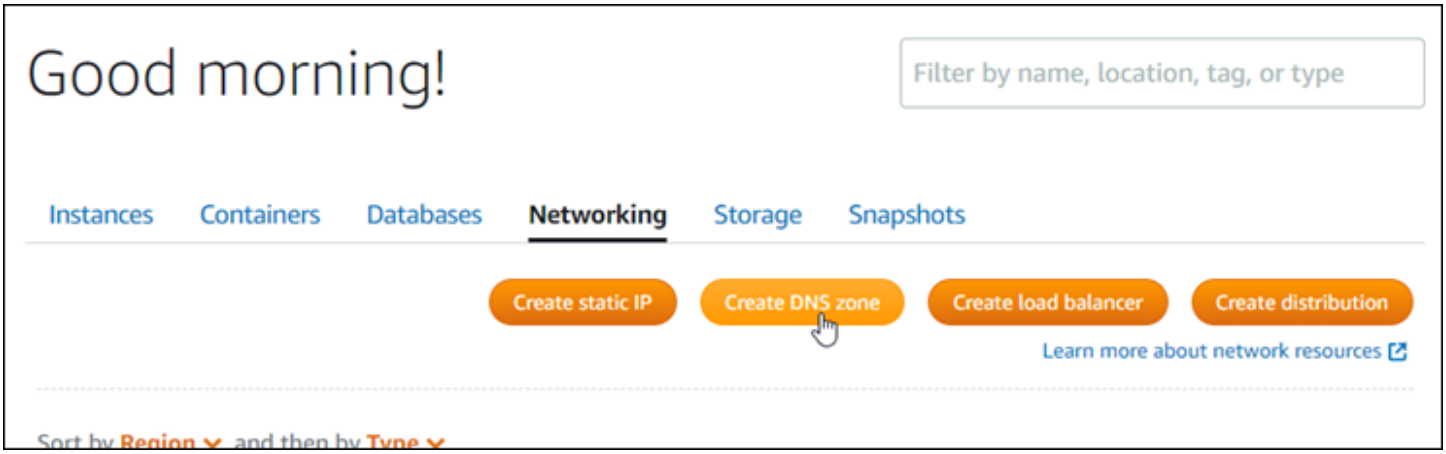
   Gitlab CE の管理ダッシュボードが表示されます。



ステップ 5: 登録済みドメイン名のトラフィックを GitLab CE ウェブサイトにルーティングする

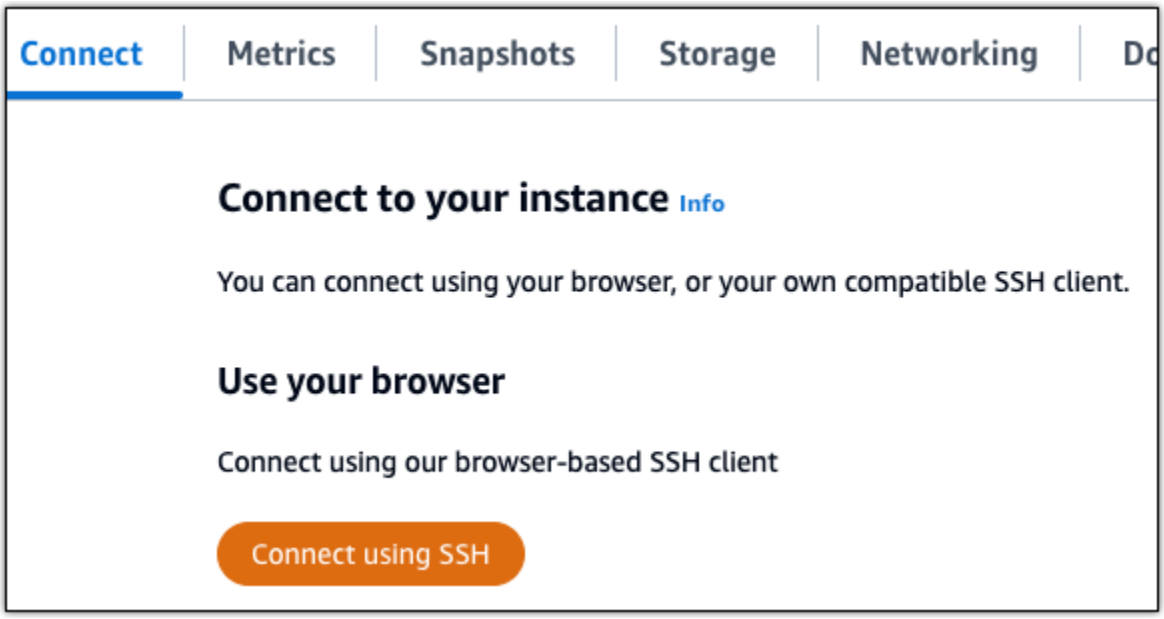
などの登録済みドメイン名のトラフィックを GitLab CE ウェブサイト`example.com`にルーティングするには、ドメインのドメインネームシステム (DNS) にレコードを追加します。DNS レコードは通常、ドメインを登録したレジストラで管理およびホストされます。ただし、ドメインのDNSレコードの管理を Lightsail に転送して、Lightsail コンソールを使用して管理できるようにすることをお勧めします。

Lightsail コンソールのホームページのネットワークタブで、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、「ドメインのDNSレコードを管理するDNSゾーンを作成する」を参照してください。



ドメイン名がインスタンスにトラフィックをルーティングしたら、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接続タブで、 を使用して接続を選択しますSSH。



2. 接続後に、次のコマンドを入力します。置換 *<DomainName>* 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。

```
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 インスタンスはドメイン名を認識しているはずです。

```
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。置換 *<DomainName>* 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。

```
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
```

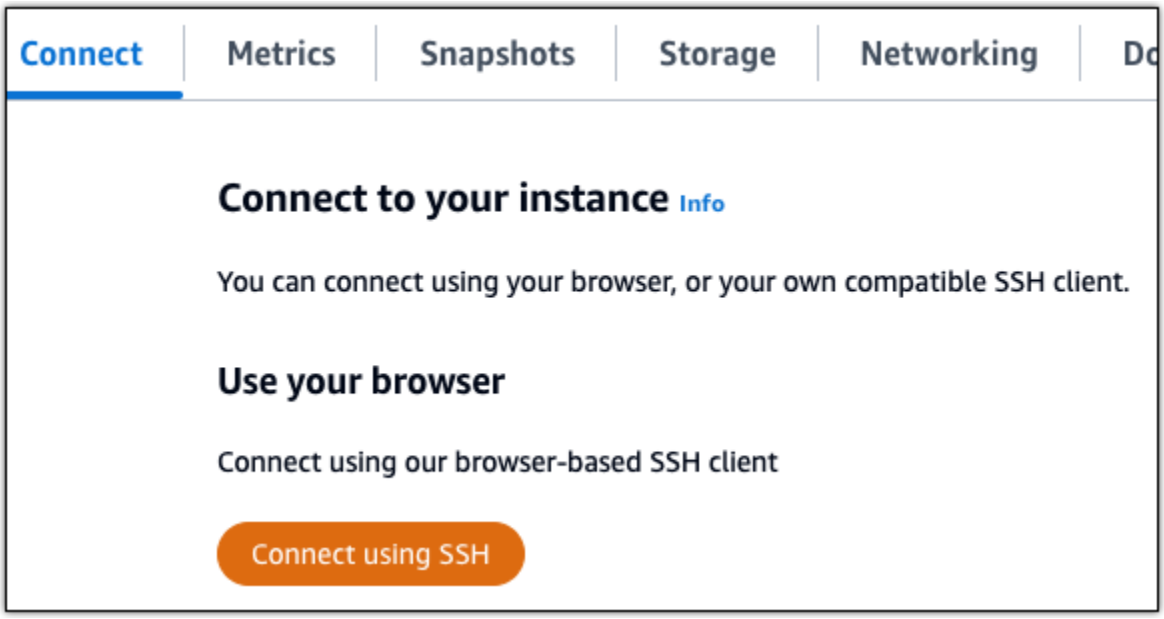
次に、 GitLab CE ウェブサイトHTTPSの接続を有効にする SSL/TLS 証明書を生成して設定する必要があります。詳細については、このガイドの次の「ステップ 6: GitLab CE ウェブサイト用に を設定するHTTPS」セクションに進んでください。

ステップ 6: CE ウェブサイトHTTPS用に を設定する GitLab

GitLab CE ウェブサイトHTTPSで を設定するには、以下の手順を実行します。これらのステップでは、Let's Encrypt / 証明書をリクエストするためのコマンドラインツールである Lego クライアントの使用方法を示します。 SSLTLS

> **Important**
>
> この手順を開始する前に、トラフィックを GitLab CE インスタンスにルーティングす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。そうしないと、SSL/TLS 証明書の検証プロセスが失敗します。登録済みドメイン名のトラフィックをルーティングするには、ドメインDNSの にレコードを追加します。DNS レコードは通常、ドメインを登録したレジストラで管理およびホストされます。ただし、ドメインのDNSレコードの管理を Lightsail に転送して、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、ページの手順に従ってください。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、 を使用して接続を選択しますSSH。



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。置換 *X.Y.Z* ダウンロードした Lego クライアントのバージョン。

```
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 クライアントを使用して Let's Encrypt SSL/TLS 証明書をリクエストします。

```
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 はサブジェクト代替名 (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して有効な証明書は 1 つだけになります。リストの最初のドメインは証明書のCommonName「」として追加され、残りは証明書内のSAN拡張機能にDNSNames「」として追加されます。

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
```

コマンドで、*Domain* Let's Encrypt 証明書をリクエストするときに指定したプライマリルートドメインを持つ 。

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るはずです。 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。

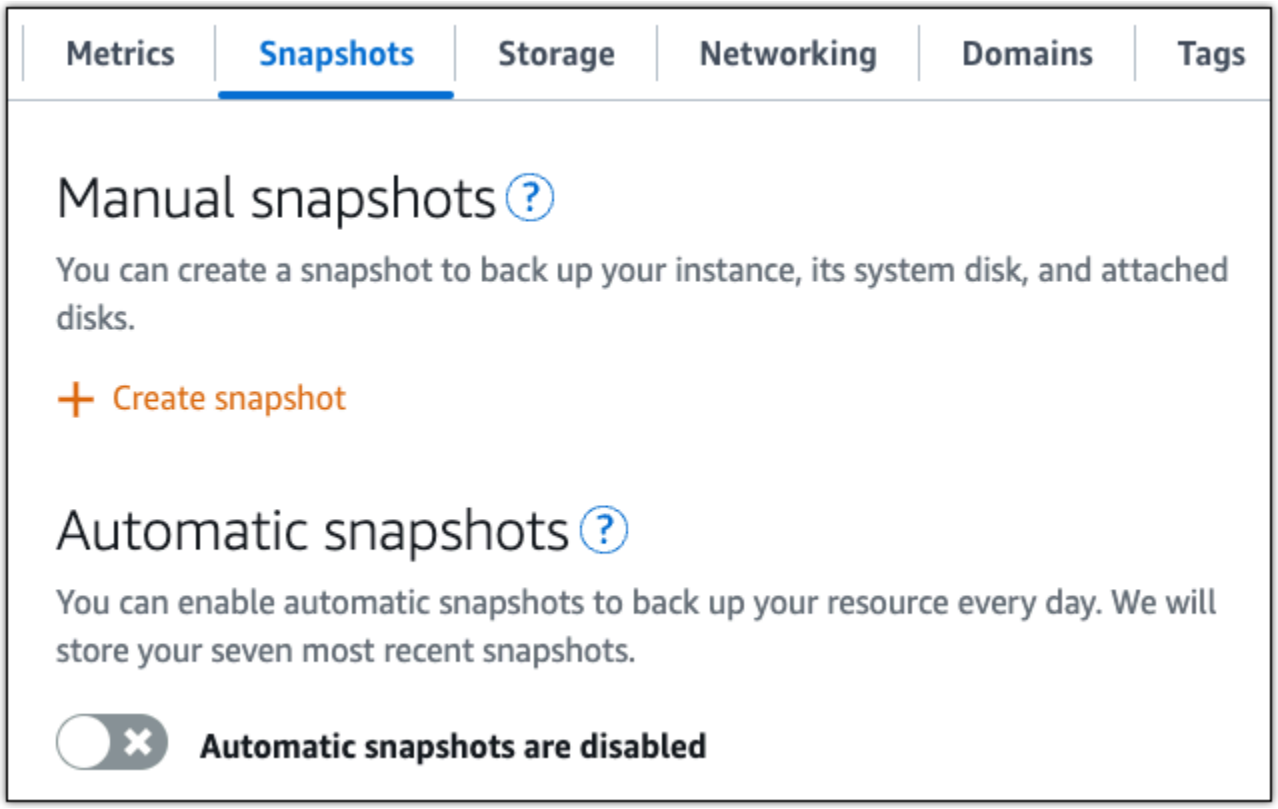
ステップ 7: GitLab CE ドキュメントを読み、ウェブサイトの設定を続ける

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。

Lightsail で Joomla! の使用を開始する

Joomla! インスタンスが Amazon Lightsail で起動および実行された後に開始するために実行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 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

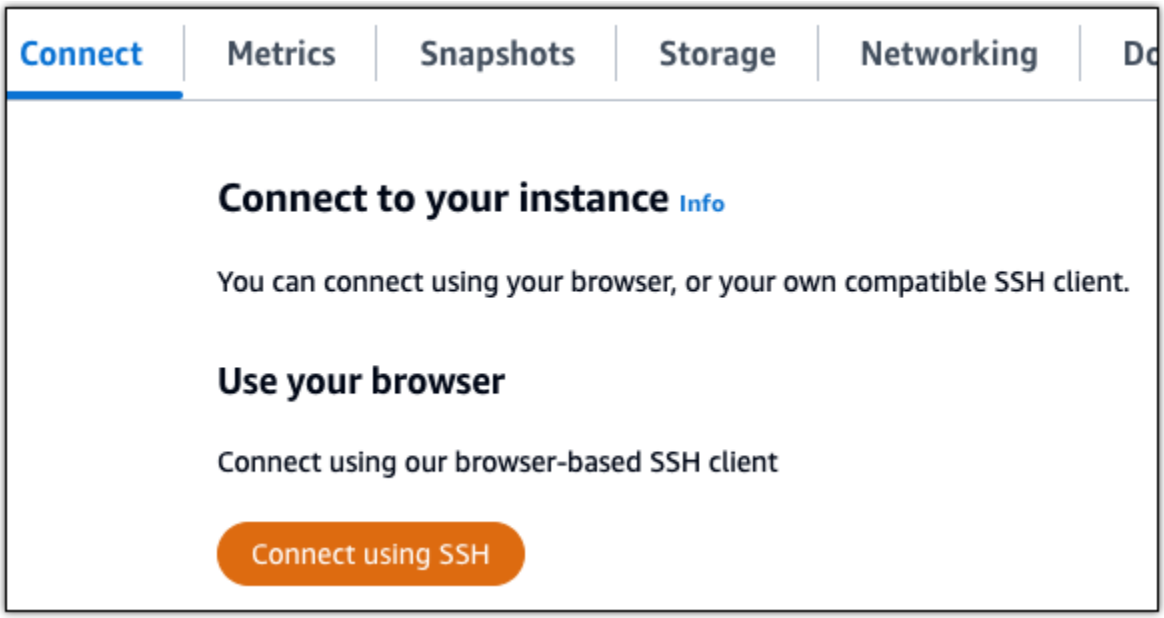
ステップ 1: Bitnami のドキュメントを確認する

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 によって にパッケージ化されました AWS クラウド。

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」を参照してください。

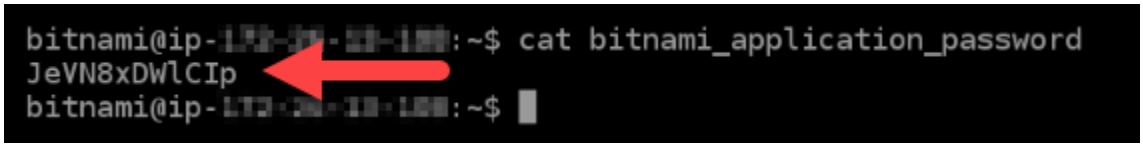
1. インスタンス管理ページの [接続] タブで、[SSH を使用して接続] を選択します。



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
```

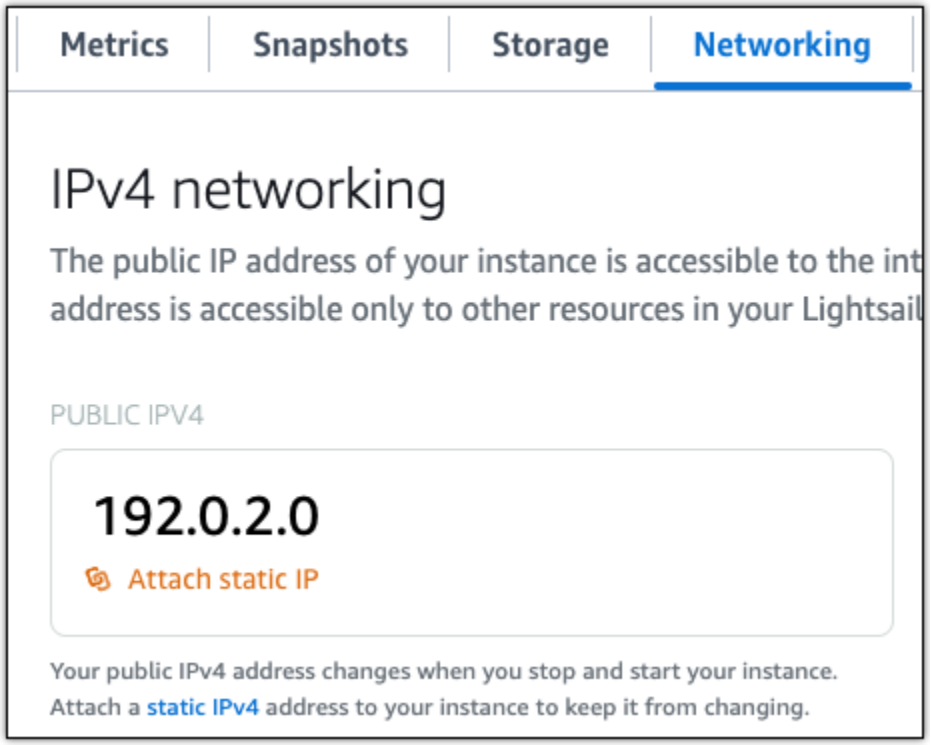
アプリケーションのデフォルトパスワードを含んだ、次の例のようなレスポンスが表示されます。



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)。

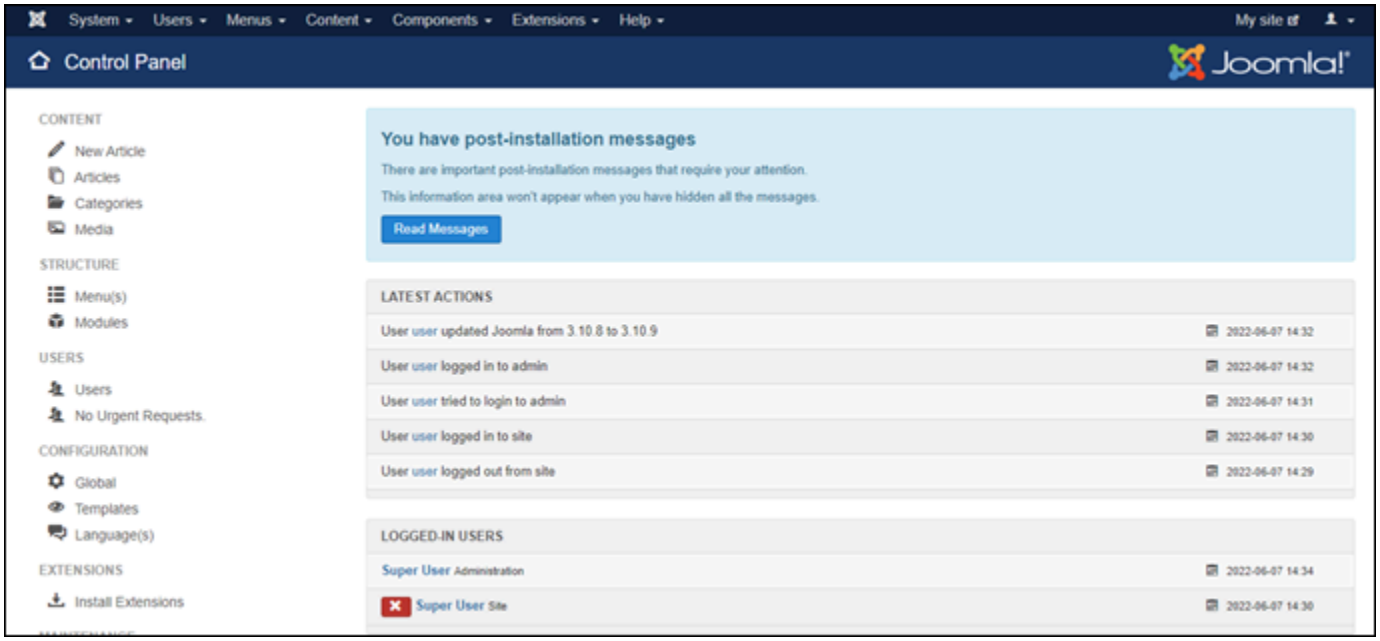
   Joomla! ウェブサイトのホームページが表示されます。

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。



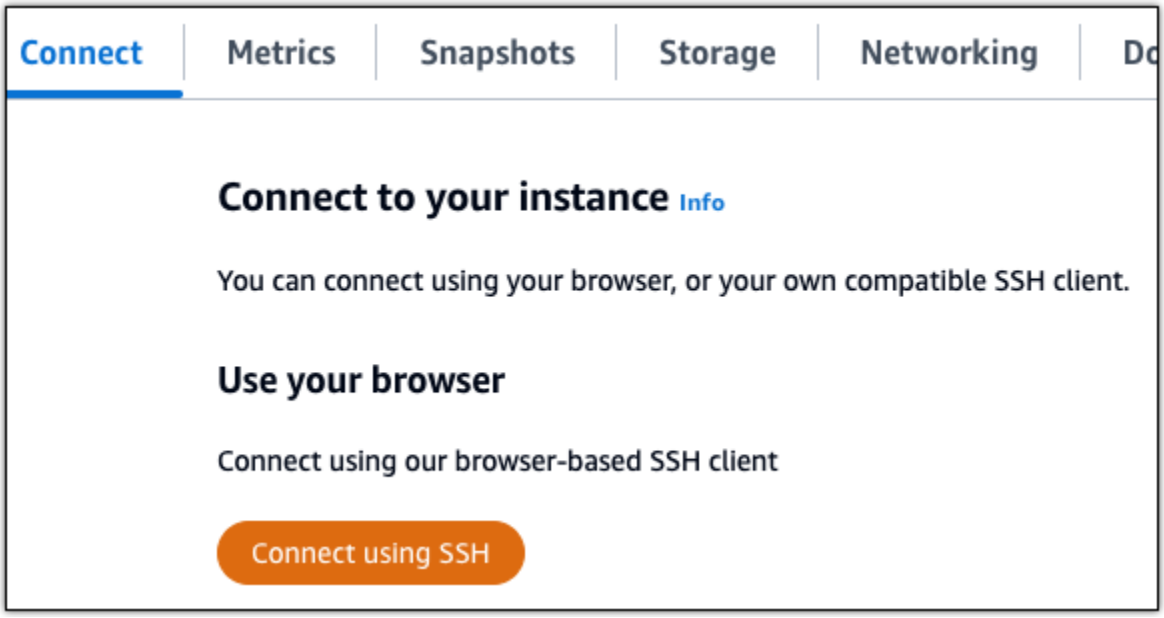
ステップ 5: 登録済みドメイン名へのトラフィックを Joomla! ウェブサイトに送信する

`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 に転送して、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。

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
```

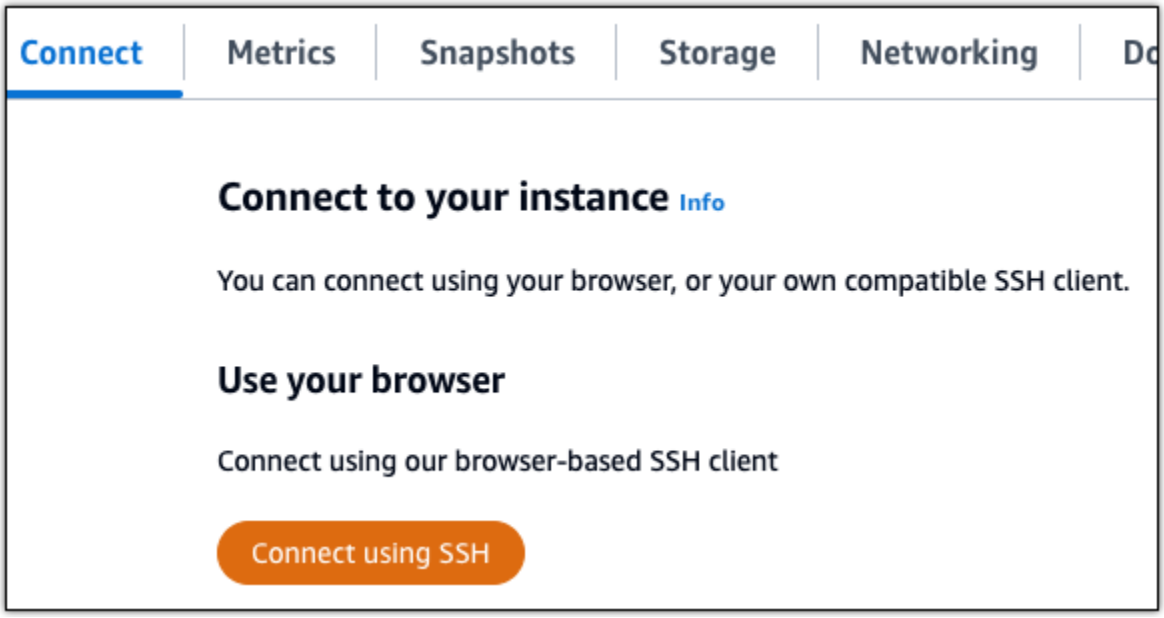
インスタンスに設定したドメイン名を参照すると、Joomla! ウェブサイトのホームページへと移動します。次に、SSL/TLS 証明書を生成して設定し、Joomla! ウェブサイトの HTTPS 接続を有効にします。詳細については、本ガイドの次の「ステップ 6: Joomla! ウェブサイトの HTTPS を設定する」のセクションを参照してください。

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。

```
--------------------------------------------------------------------------------
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
```

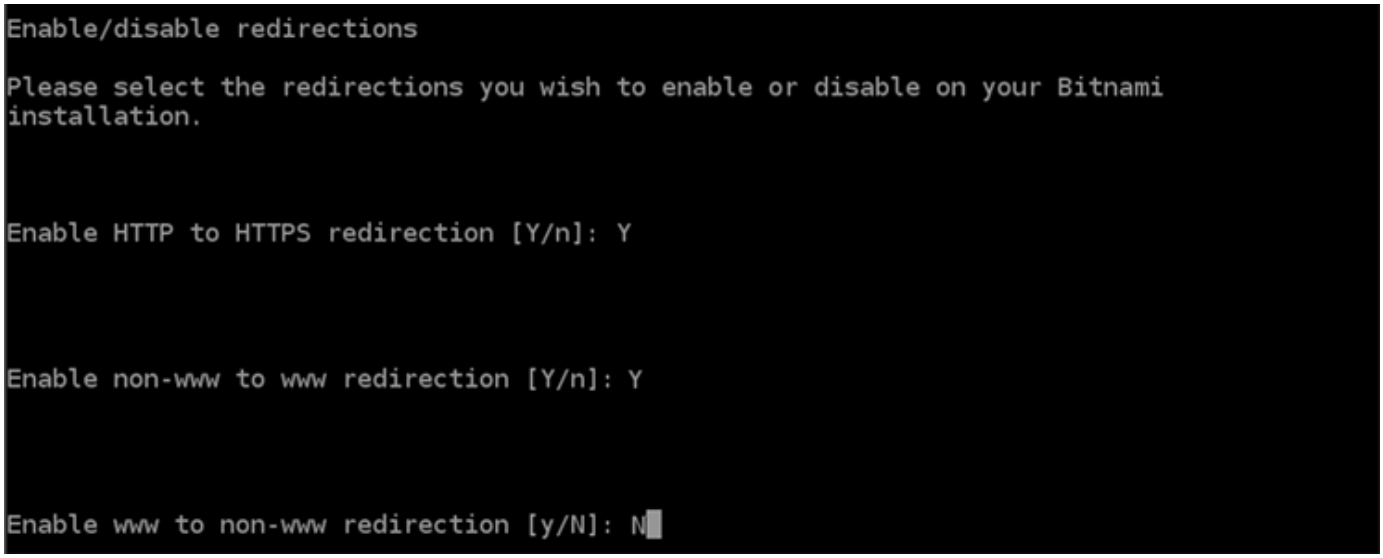
9. bncert ツールは、ウェブサイトのリダイレクトの設定方法を尋ねます。使用できるオプションは次のとおりです。

- 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。



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 []:
```

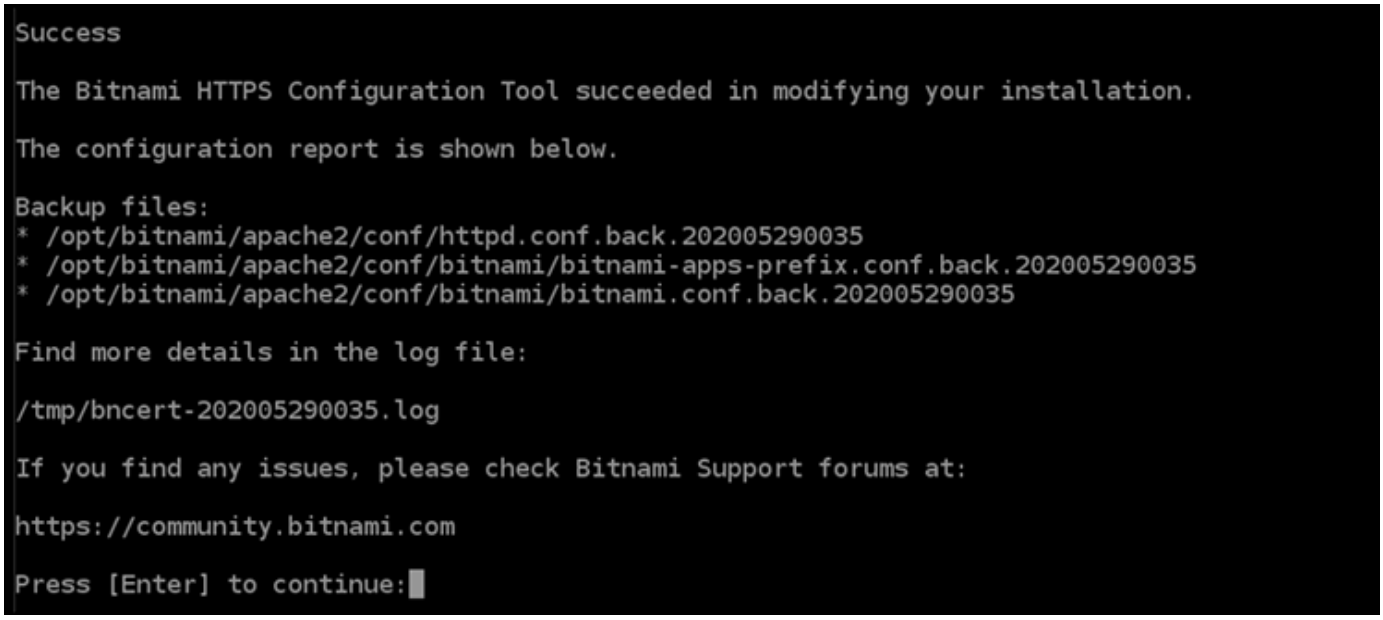
12. Let's Encrypt サブスクライバー合意書を確認します。Y と入力し、Enter を押して契約に同意し、続行します。

```
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、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。

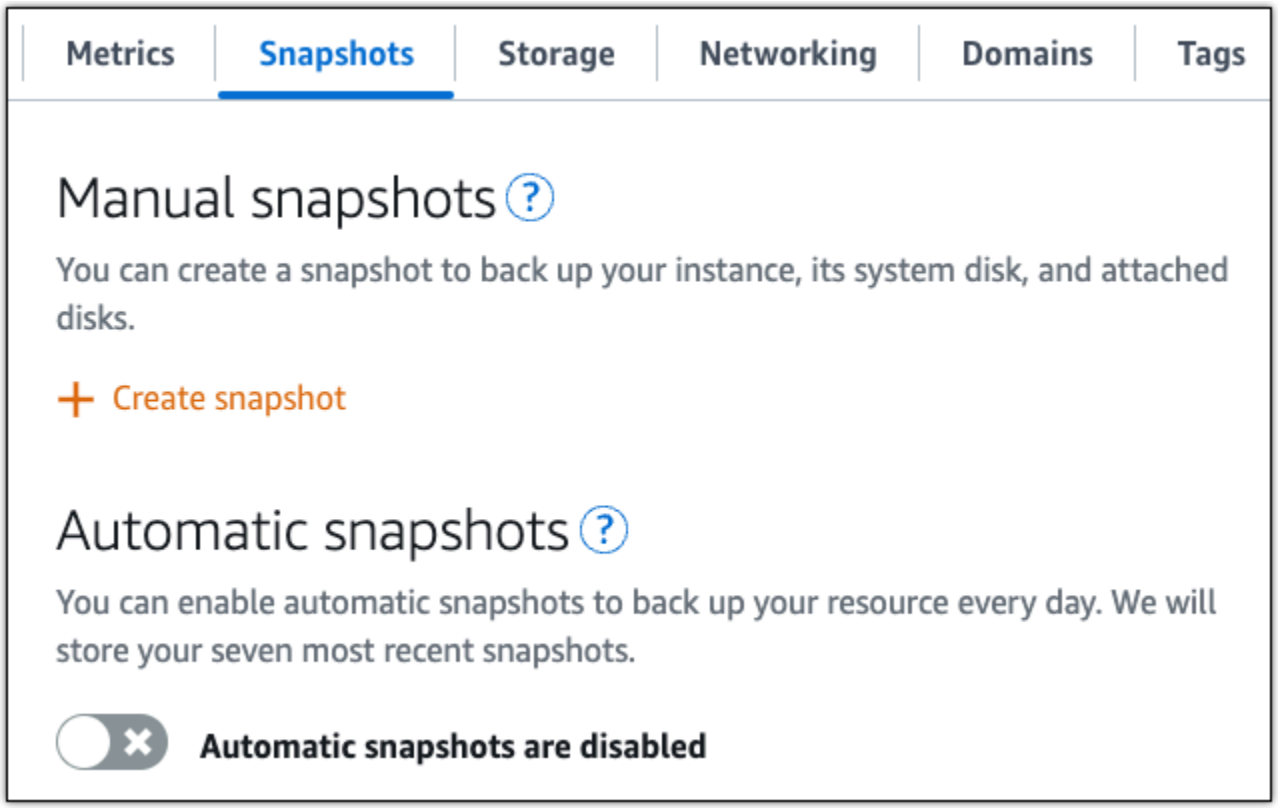
ステップ 7: Joomla! のドキュメントを読み、引き続きウェブサイトの設定を続行する

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。

Lightsail で LAMP スタックを設定する

LAMP インスタンスが Amazon Lightsail で起動および実行された後に開始するために実行する必要があるいくつかのステップを次に示します。

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

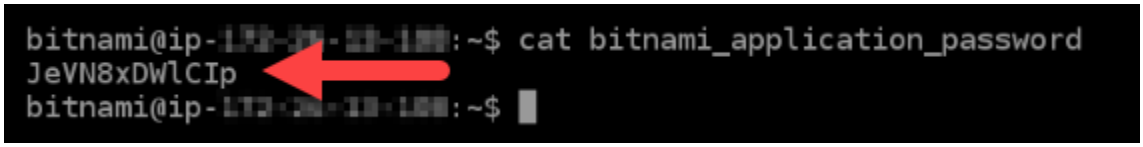
インスタンスにプリインストールされているアプリケーションやサービスにアクセスするには、アプリケーションのデフォルトパスワードが必要です。

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」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」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 に転送して、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: 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。

Lightsail で Magento をセットアップおよび設定する

Magento 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に、いくつかのステップを完了する必要があります。

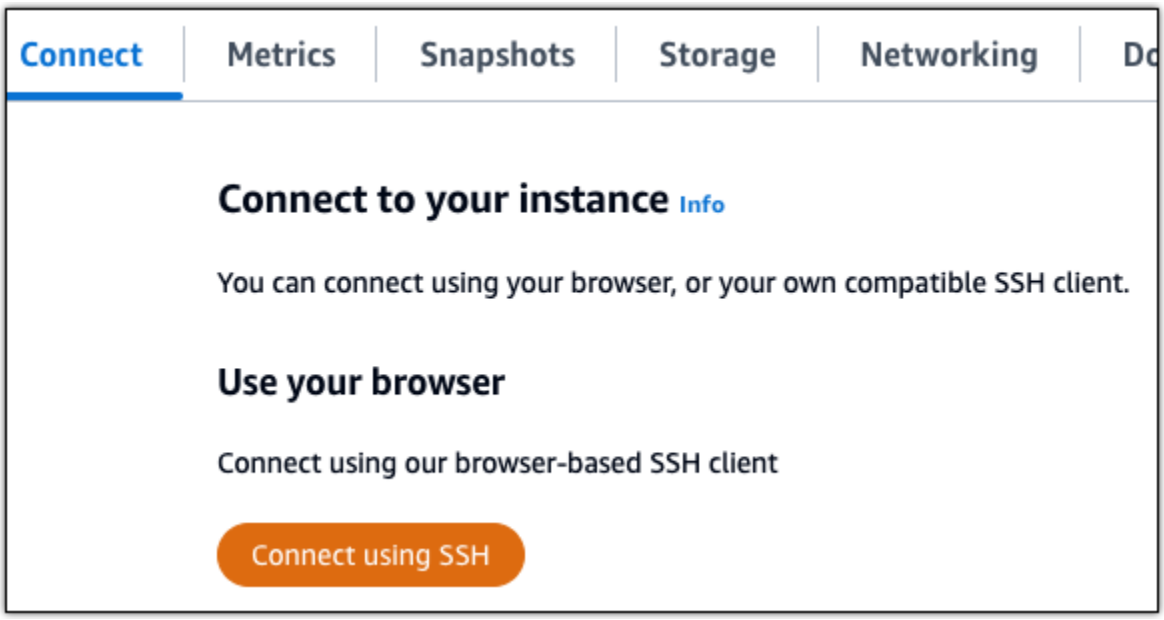
目次

- 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
- 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
- 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
- ステップ 5: Magento ウェブサイトの 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 Magento のドキュメントを読む
- 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

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」を参照してください。

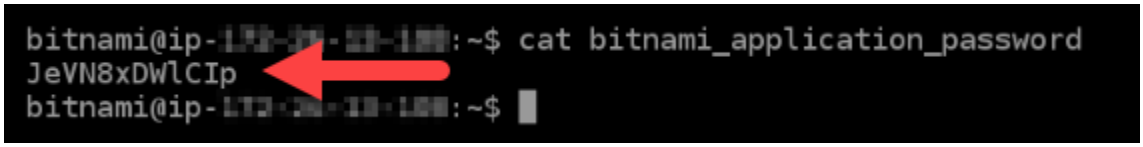
1. インスタンス管理ページの [Connect] (接続) タブで、[SSH を使用して接続] を選択します。



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
```

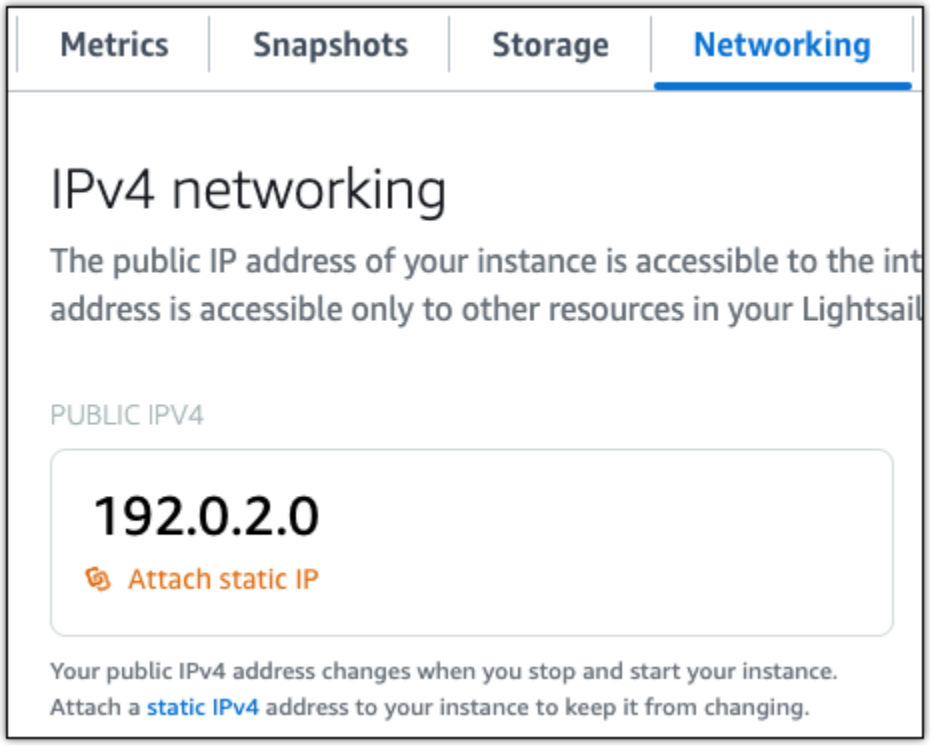
アプリケーションのデフォルトパスワードを含んだ、次の例のようなレスポンスが表示されます。このパスワードを安全な場所に保存します。このチュートリアルの次のセクションで、Magento ウェブサイトの管理ダッシュボードにサインインする際に使用します。



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。

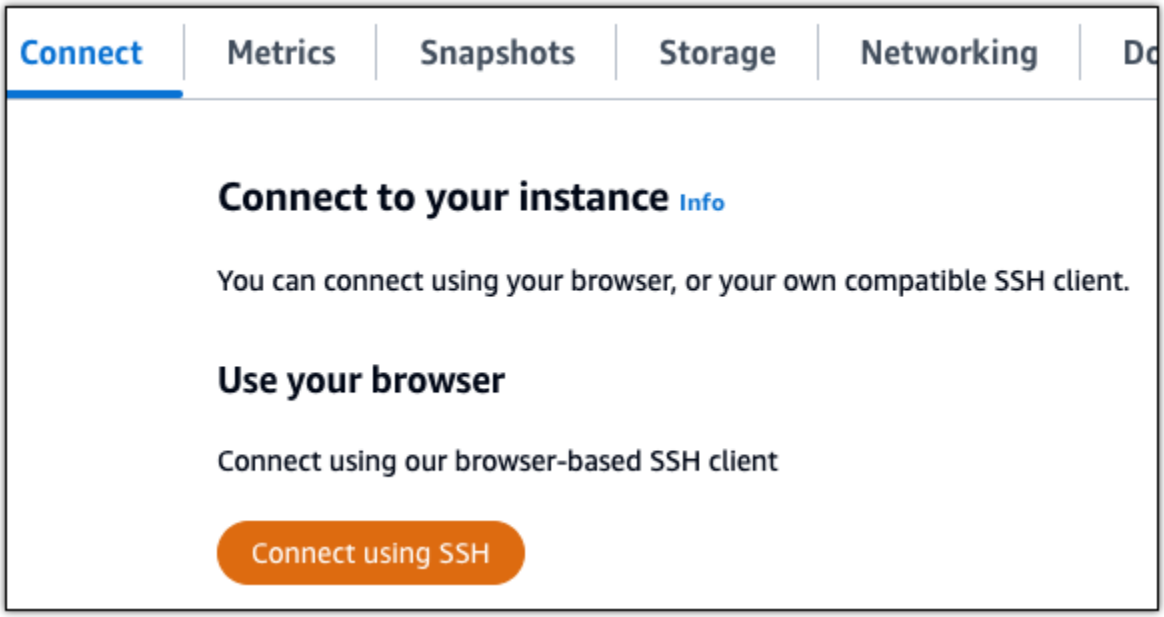
インスタンス管理ページの [ネットワーク] タブで、[静的 IP の作成] または [静的 IP のアタッチ] (インスタンスにアタッチできる静的 IP を既に作成している場合) を選択して、ページの手順に従います。詳細については、「静的 IP を作成してインスタンスにアタッチする」を参照してください。



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。

```
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**
>
> 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。*#InstanceIpAddress#* を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>/admin
```

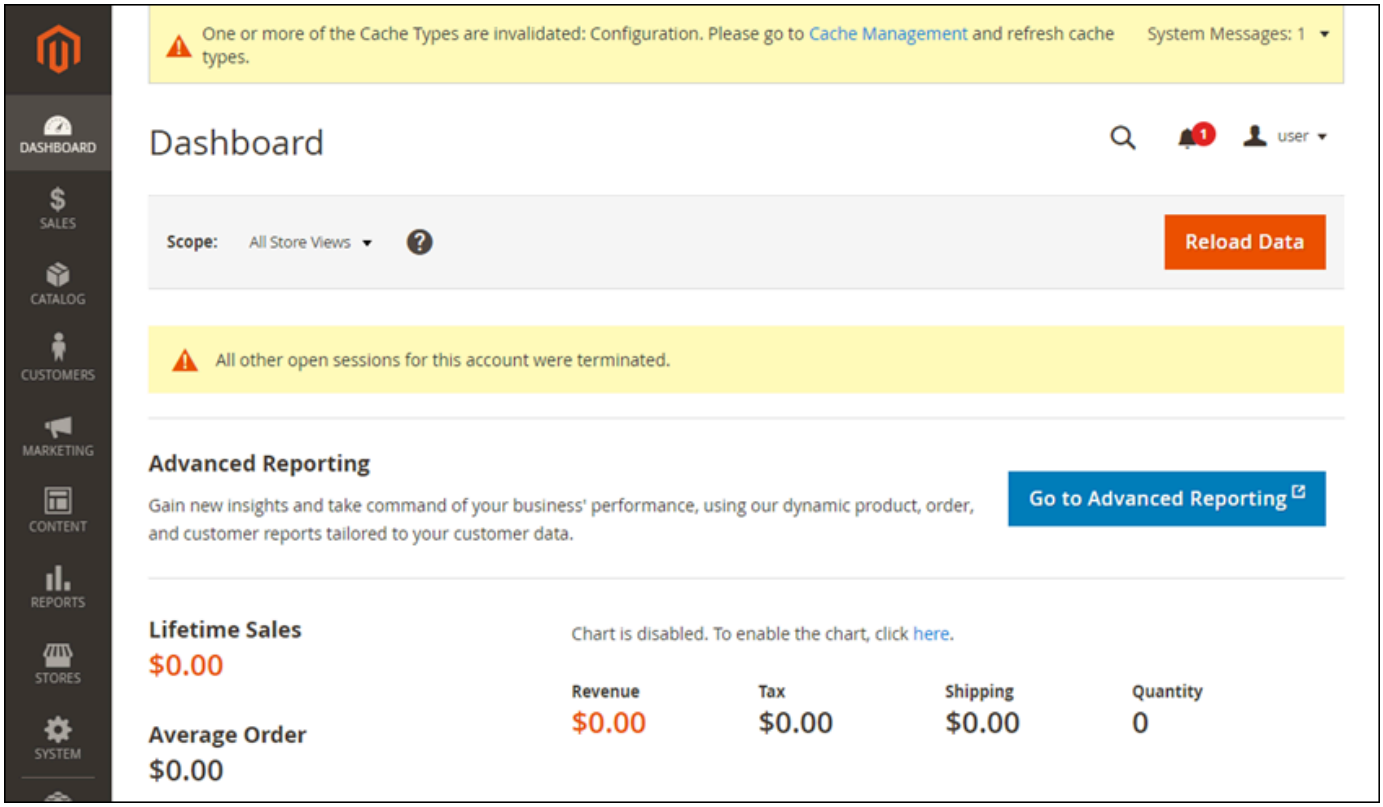
例:

```
http://203.0.113.0/admin
```

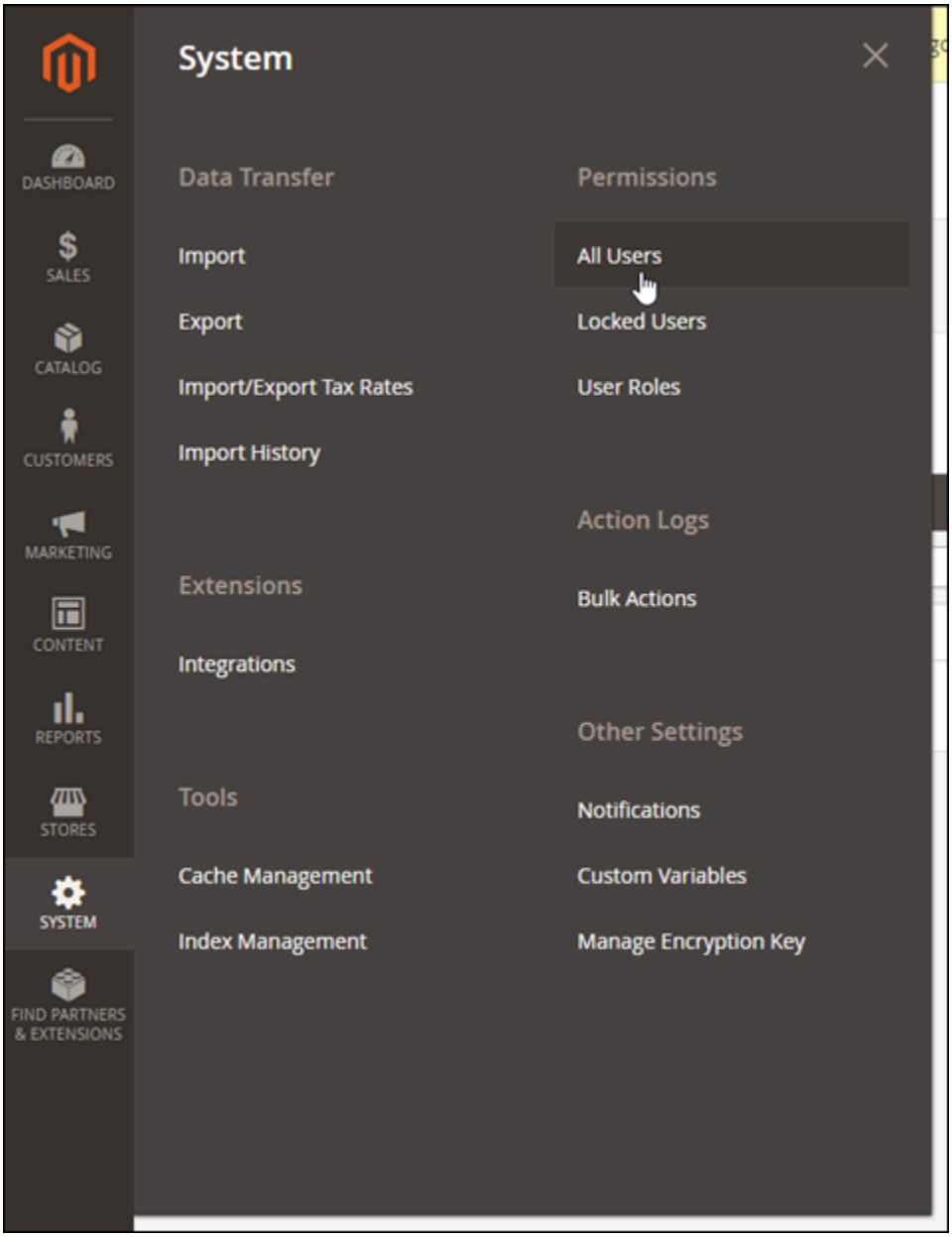
> **Note**
>
> Magento 管理用ダッシュボードのサインインページにアクセスできない場合、インスタンスを再起動する必要があるかもしれません。

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。

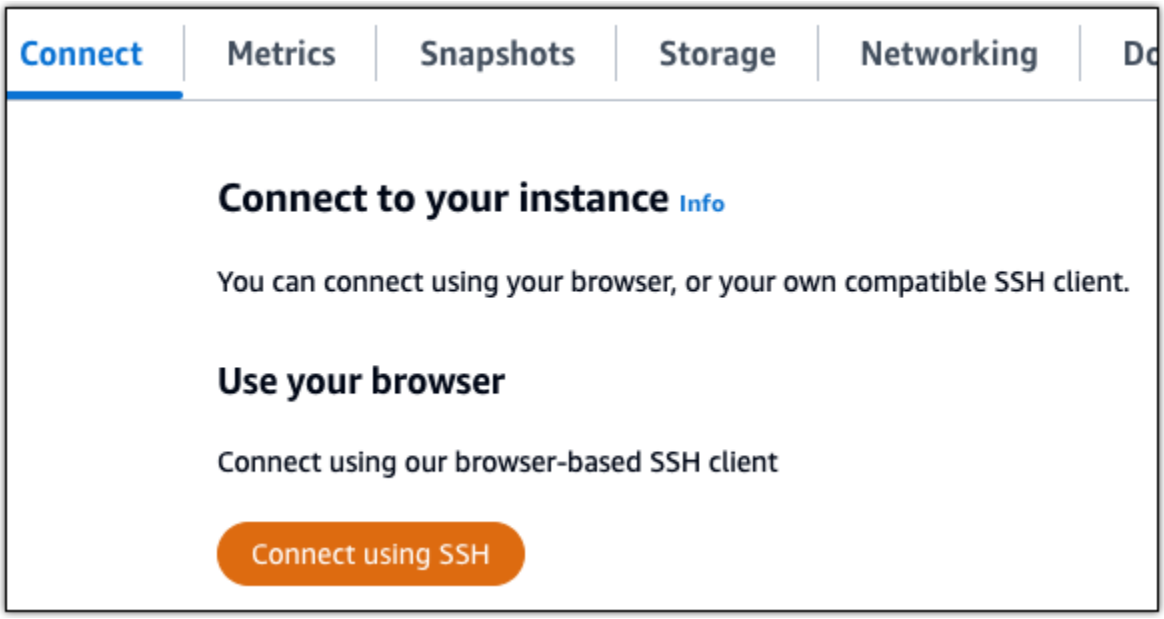
ステップ 4: 登録済みドメイン名へのトラフィックを Magento ウェブサイトに送信する

`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 に転送して、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 DNS ゾーンの作成」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#* 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。

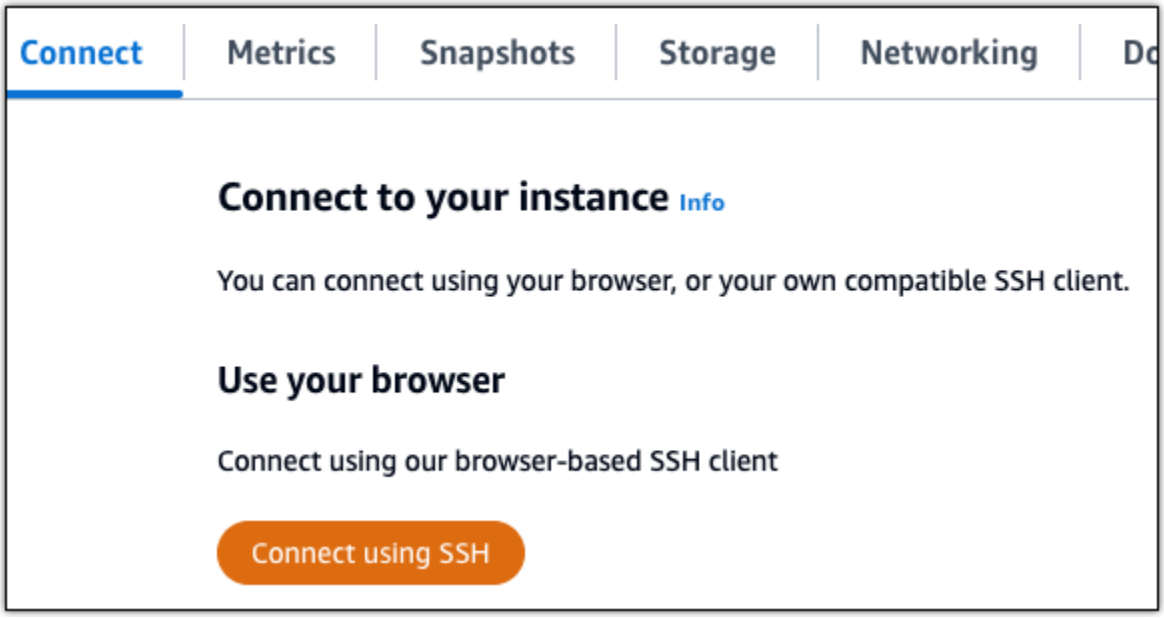
```
bitnami@ip-[redacted]:~$ sudo /opt/bitnami/configure_app_domain --domain www.example.com
Configuring domain to www.example.com
2021-03-12T15:49:22.000Z - info: Saving configuration info to disk
prestashop 15:49:22.41 INFO  ==> Trying to connect to the database server
prestashop 15:49:22.44 INFO  ==> 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 []:
```

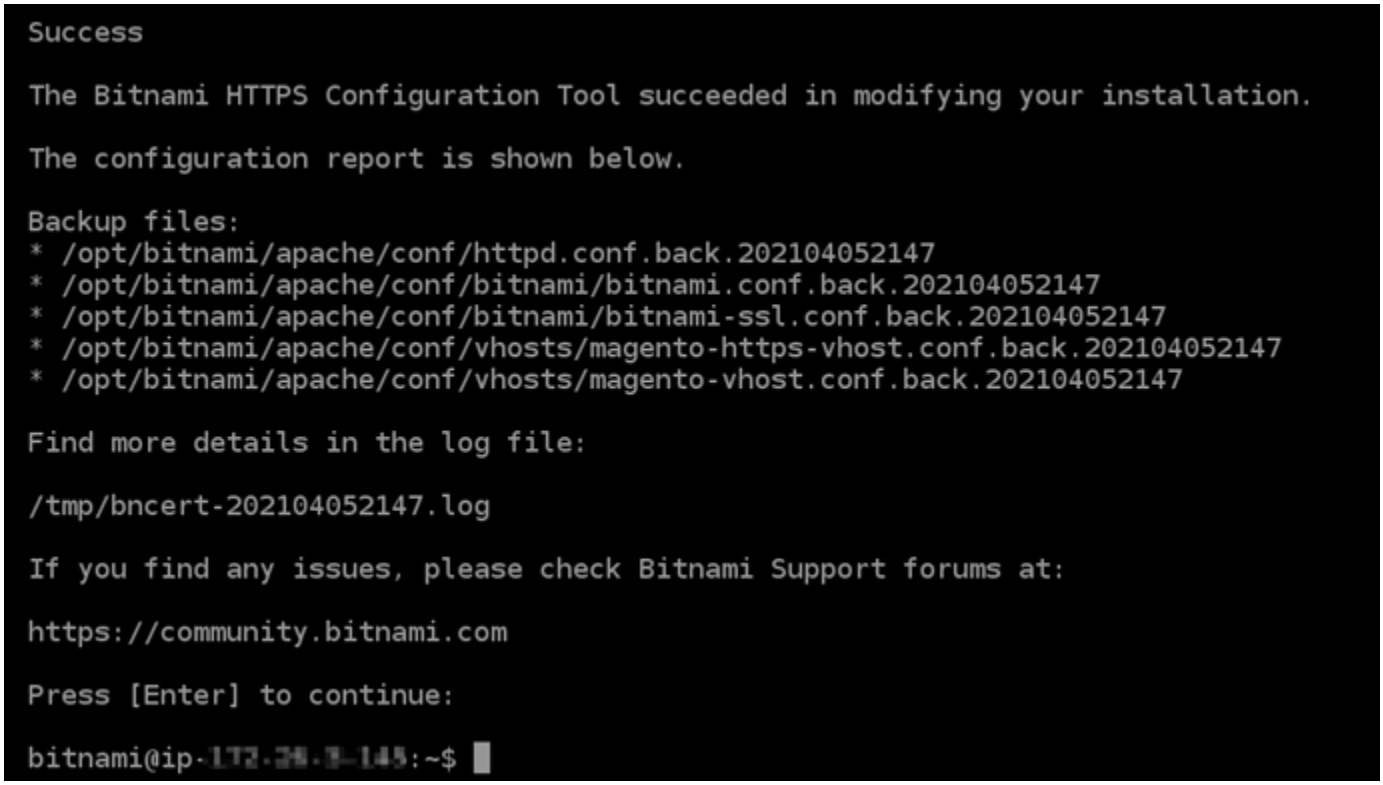
6. Let's Encrypt サブスクライバー合意書を確認します。Y と入力し、Enter を押して契約に同意し、続行します。

```
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。



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#* 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登録済みドメイン名に置き換えてください。

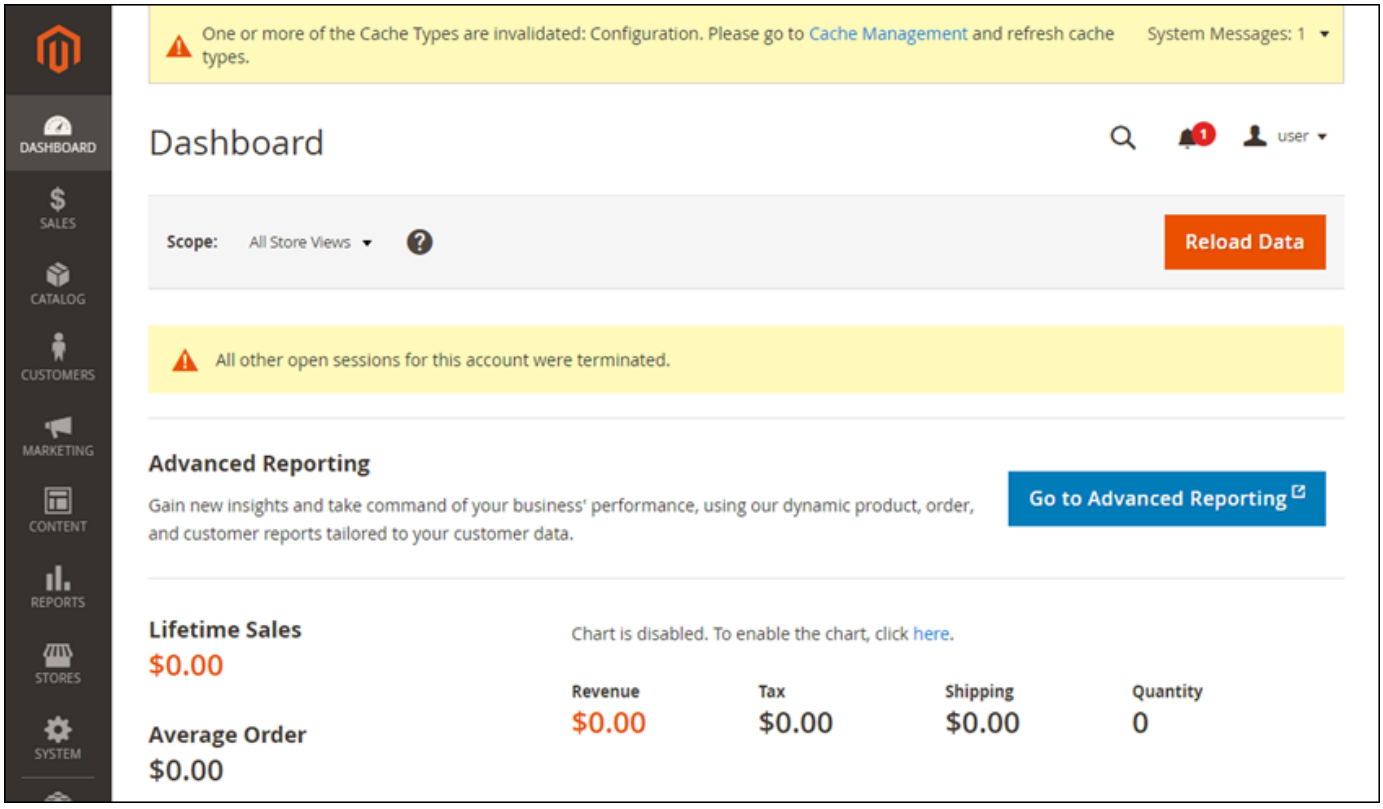
```
http://<DomainName>/admin
```

例:

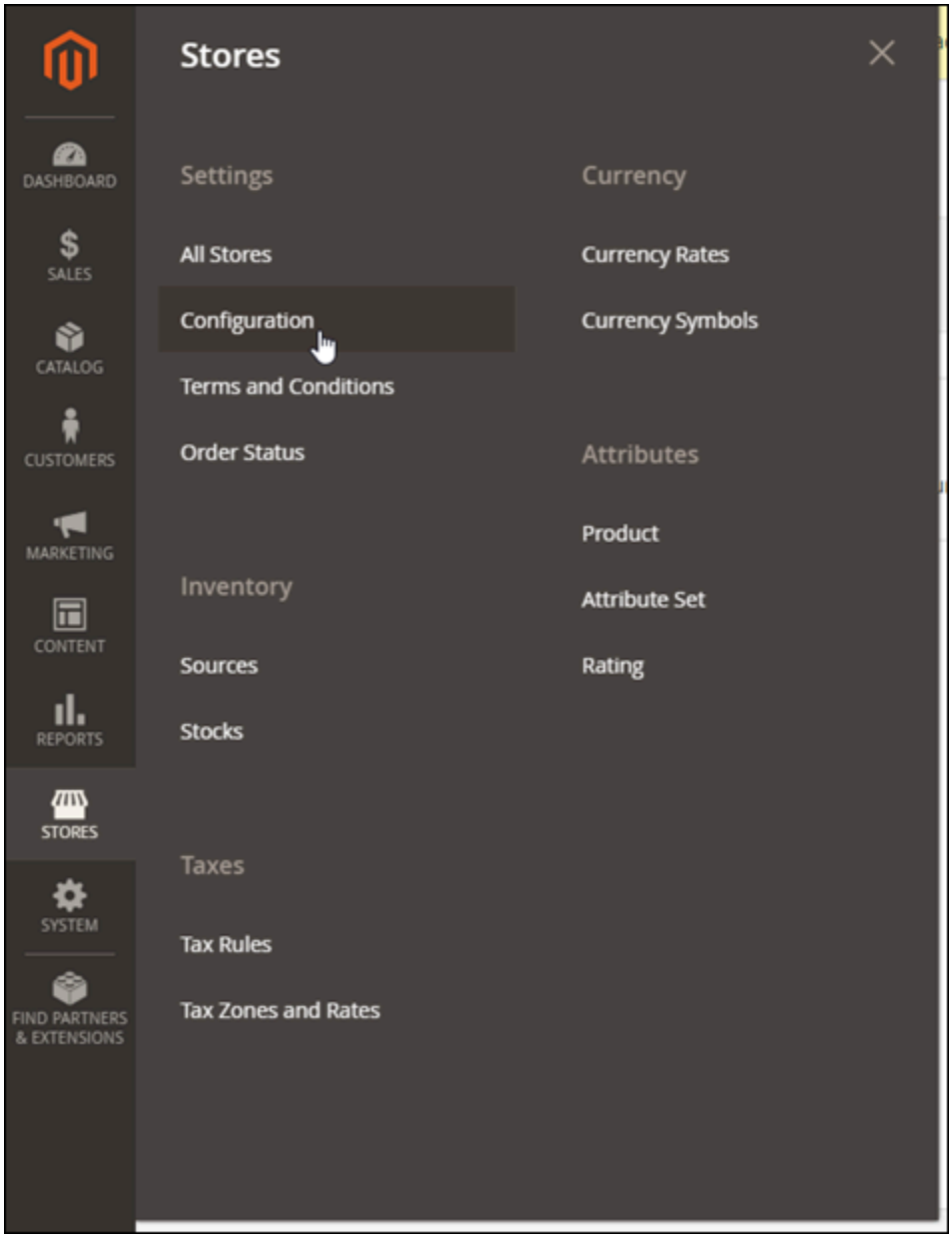
```
http://www.example.com/admin
```

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。

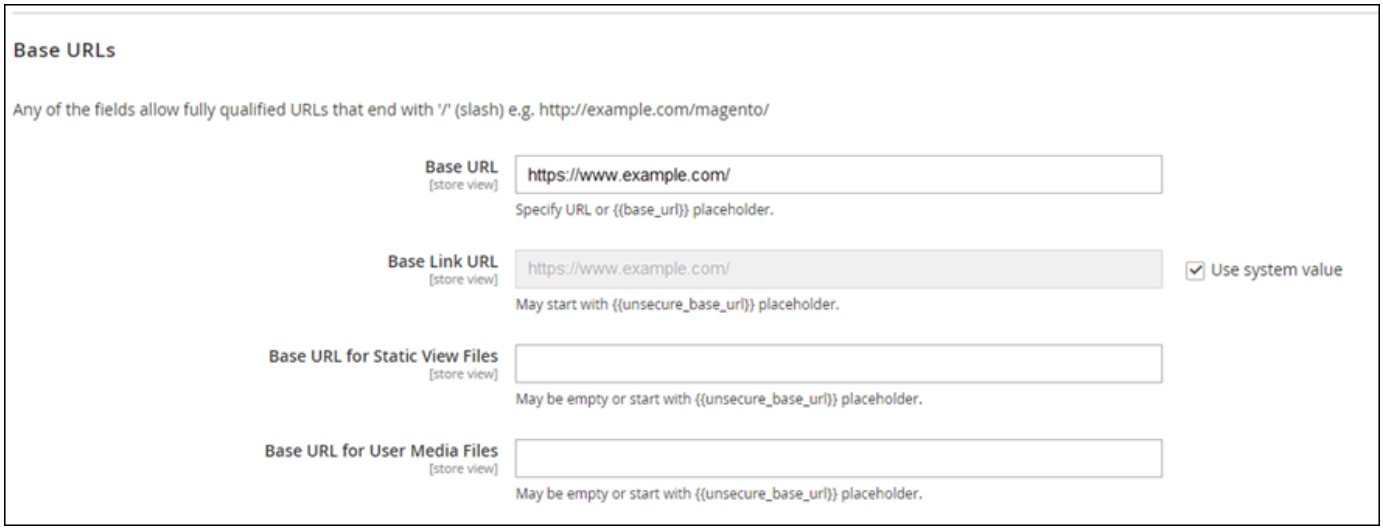
Magento の管理ダッシュボードが表示されます。



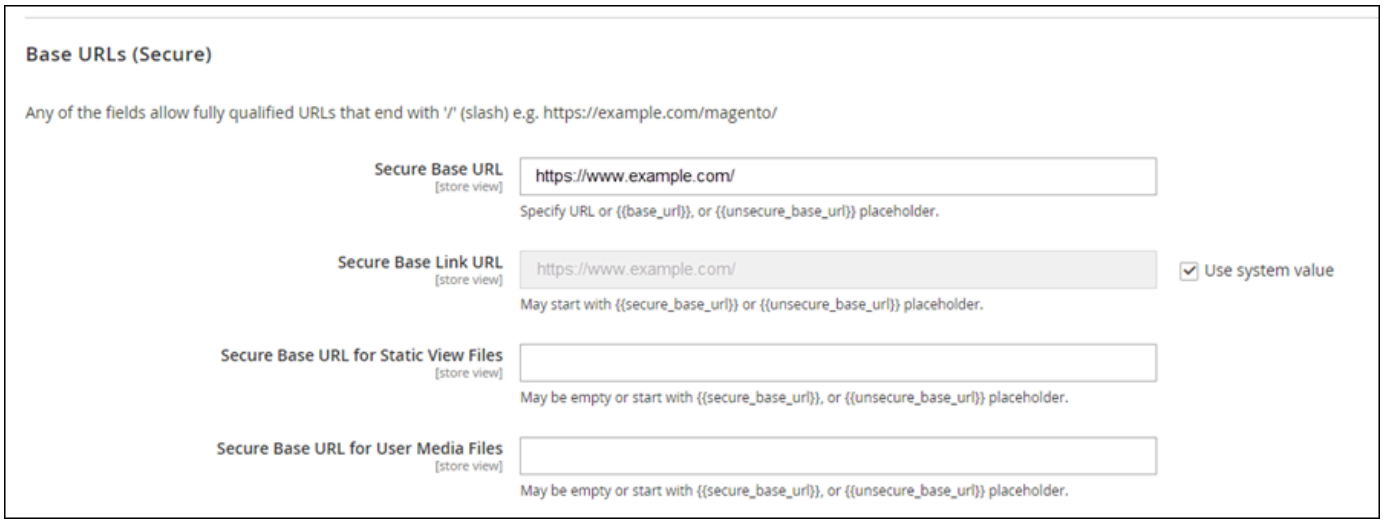
9. ナビゲーションペインで [Stores] (ストア)、[Configuration] (設定) の順に選択します。



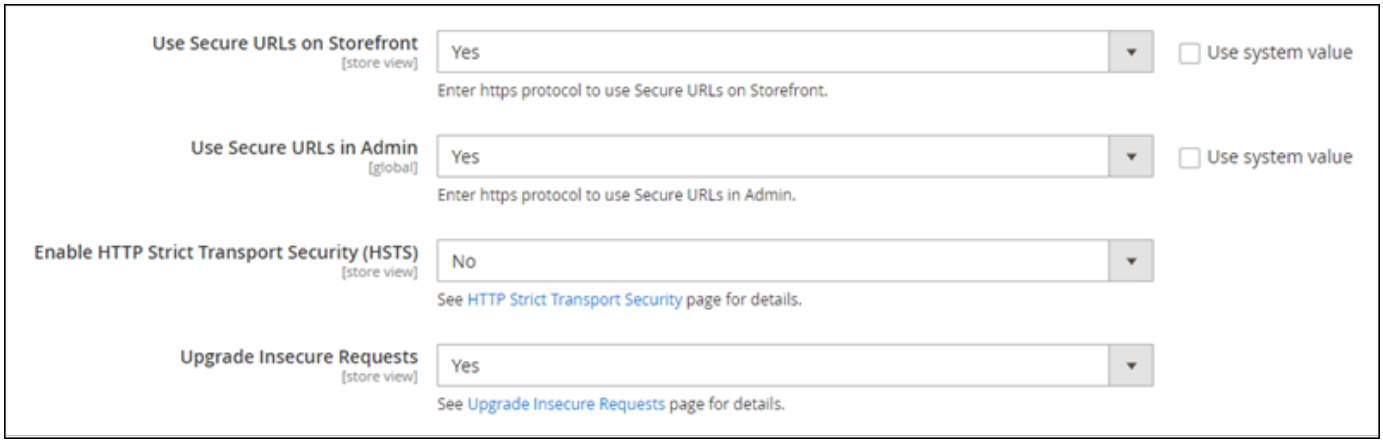
10. [Web] (ウェブ) を選択し、[Base URLs] (ベース URL) ノードを展開します。
11. [Base URLs] (ベース URL) テキストボックスに、ウェブサイトの完全な URL を入力します (例: `https://www.example.com/`)。



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 を設定する場合は、Lightsail コンソールでインスタンスのファイアウォールでこ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。
>
> 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

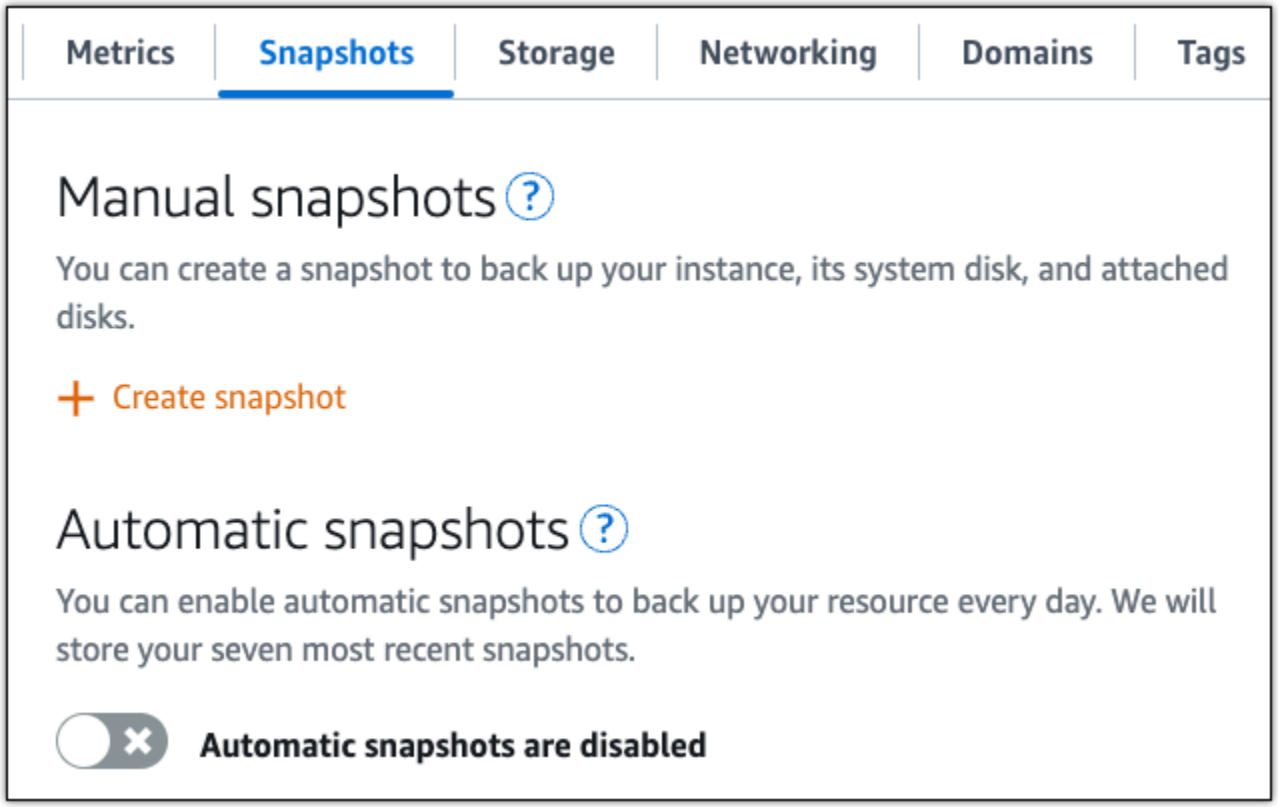
は、Bitnami ドキュメントの「AWS クラウド用の Bitnami PrestaShop スタック」を参照してください。

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。

Lightsail で Nginx ウェブサーバーをデプロイして管理する

Nginx インスタンスが Amazon Lightsail で起動および実行された後に開始するために実行する必要があるいくつかのステップを次に示します。

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

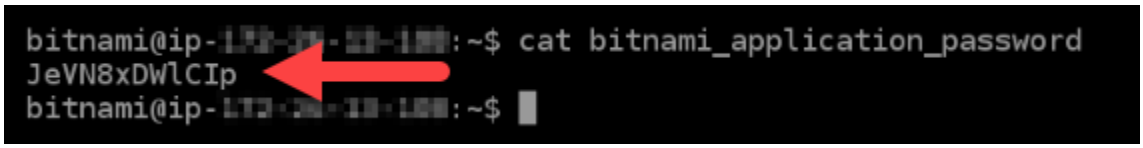
インスタンスにプリインストールされているアプリケーションやサービスにアクセスするには、アプリケーションのデフォルトパスワードが必要です。

1. インスタンス管理ページの接続タブで、 を使用して接続を選択しますSSH。
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」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。後で、インスタンスでドメイン名を使用する場合、インスタンスを停止および起動するたびにドメインのDNSレコードを更新する必要はありません。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。

インスタンス管理ページのドメインとDNSタブで、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 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」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

などのドメイン名をインスタンス`example.com`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、ドメインを登録したレジストラで管理およびホストされます。ただし、ドメインのDNSレコードの管理を Lightsail に転送して、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ネットワークタブで、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、「ドメインのDNSレコードを管理するDNSゾーンを作成する」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 のドキュメントを読んで、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、 を使用したサーバーへのファイルのアップロードSFTPなどを行う方法を確認してください。

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、データ転送速度などの情報が含まれます。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。

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。

Lightsail で Node.js の使用を開始する

Node.js インスタンスが Amazon Lightsail で起動および実行された後に開始するために実行する必要があるいくつかのステップを次に示します。

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

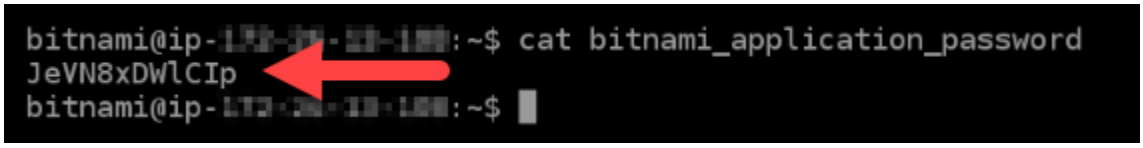
インスタンスにプリインストールされているアプリケーションやサービスにアクセスするには、アプリケーションのデフォルトパスワードが必要です。

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」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 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: 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」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 に転送して、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ネットワークタブで、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、「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。

Lightsail に Plesk ホスティングスタックをデプロイする

Amazon 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 ユーザーインターフェイスに初めてサインインする方法について説明します。また、Plesk インスタンスが起動して実行された後に、そのインスタンスに接続して設定する方法についても説明します。

> **Important**
>
> Plesk インスタンスには 30 日間のトライアルライセンスが含まれています。Plesk アプリケーションを引き続き使用するには、30 日後に Plesk からライセンスを購入する必要があります。

Lightsail の Plesk ホスティングスタックには、次の機能が含まれています。

- WordPress グラフィカルユーザーインターフェイスで自動化機能を備えたツールキット
- Let's Encrypt でのSSL証明書のサポートと、1 つ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

- FTP インスタンスとの間でファイルを転送するための アクセス
- Docker プロキシルール
- Plesk Firewall、Logs、 などの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 ModSecurity

ステップ 1: Plesk インスタンスを作成する

Lightsail で Plesk インスタンスを作成するには、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. インスタンスのホームページで、インスタンスの作成 を選択します。
3. インスタンスを作成する場所を選択します。

   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、インスタンスの場所を変更します。
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。Lightsail プランUSDは、Plesk ホスティングスタックをサポートしていません。
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. (オプション) インスタンスにタグを追加します。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。

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 2: Plesk ユーザーインターフェイスに初めてサインインする

ワンタイムログイン を取得するには、次の手順に従いますURL。管理者として Plesk ユーザーインターフェイスURLにアクセスするには、1 回限りのログインが必要です。

1. インスタンス管理ページの接続タブで、 を使用して接続を選択しますSSH。
2. 接続したら、次のコマンドを入力して 1 回限りのログインを取得しますURL。

   ```
   sudo plesk login | grep -v internal:8
   ```

   ワンタイムログイン を含む次の例のようなレスポンスが表示されますURL。

   ```
   https://heuristic-bassi.192-0-2-0.plesk.page/login?secret=ce-
   e3b0c44298fc1c149afbf4c8996fb92427
   ```

   > **Tip**
   >
   > 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URLを使用する 1 回限りのログインを取得することがあります。インスタンスを再起動し、上記のコマンドを再度実行して、新しい静的パブリック IP アドレスURLを使用する 1 回限りのログインを取得します。

3. 1 回限りのログインをコピーしてウェブブラウザURLに貼り付けます。

   > **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。

4. 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。

後で再度サインインするには、 に移動しますhttps://*PublicIPAddress*:8443。置換 *PublicIPAddress* インスタンスのパブリック IP アドレスまたは静的 IP アドレスを指定します。例えば、https://*192.0.2.0*/:8443 と指定します。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 ユーザーインターフェイスにサインインします。

ステップ 3: Plesk ドキュメントを読む

Plesk ドキュメントを読んで、ウェブサイトの管理、Plesk ユーザーインターフェイスのカスタマイズなどを行う方法を確認してください。

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 4: 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。後で、インスタンスでドメイン名を使用する場合、インスタンスを停止および起動するたびにドメインのDNSレコードを更新する必要はありません。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。

インスタンス管理ページのネットワークタブで、静的 IP をアタッチを選択し、ページの手順に従ってください。

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 5: ドメイン名を Plesk インスタンスにマッピングする

ドメインを Plesk インスタンスにマッピングします。これを使用して Plesk ユーザーインターフェイスにアクセスできます。Plesk ユーザーインターフェイス内で複数のドメインをマッピングすることもできます。これを使用してウェブサイトを管理できます。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 ユーザーインターフェイス内の複数のドメインのマッピングの詳細については、Plesk ドキュメントの「Plesk でのドメインの追加」および「ヘルプポータル」を参照してください。

などのドメイン名をインスタンスexample.com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、ドメインを登録したレジストラで管理およびホストされます。ただし、ドメインのDNSレコードの管理を Lightsail に転送して、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で、DNSゾーンの作成 を選択し、ページの手順に従ってください。

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: Plesk ライセンスを購入する

Plesk インスタンスには 30 日間のトライアルライセンスが含まれています。30 日後に Plesk からライセンスを購入して、引き続き使用する必要があります。詳細については、Plesk ウェブサイトの「 の料金」を参照してください。

Plesk からライセンスを購入した後、ライセンスをインストールする必要があります。Plesk ライセンスをインストールするには、Plesk サポートウェブサイトの「Plesk ライセンスをインストールする方法」を参照してください。

ステップ 7: 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、データ転送速度などの情報が含まれます。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。

インスタンスの管理ページのスナップショットタブで、スナップショットの作成 を選択します。次に、ページの手順に従ってください。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。

Lightsail で PrestaShop ウェブサイトを設定する

PrestaShop 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には、いくつかのステップを完了する必要があります。

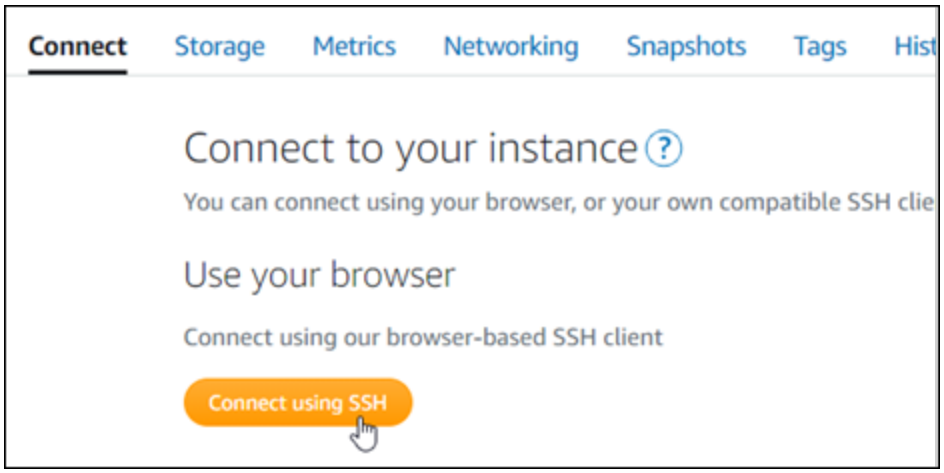
目次

- ステップ 5: PrestaShop ウェブサイトの 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 の PrestaShop ドキュメントを読む
- 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

ステップ 1: PrestaShop 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

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には、次のステップを実行します PrestaShop。

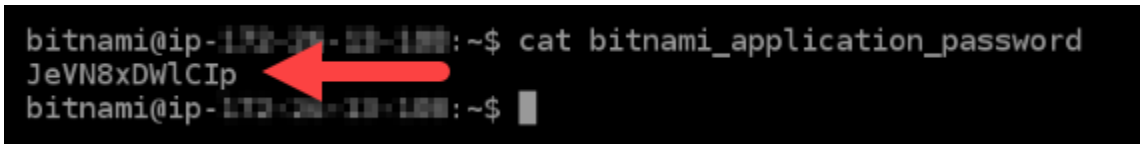
1. インスタンス管理ページの [Connect] (接続) タブで、[SSH を使用して接続] を選択します。



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、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使用します PrestaShop。



詳細については、Amazon Lightsail」を参照してください。

ステップ 2: PrestaShop 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。

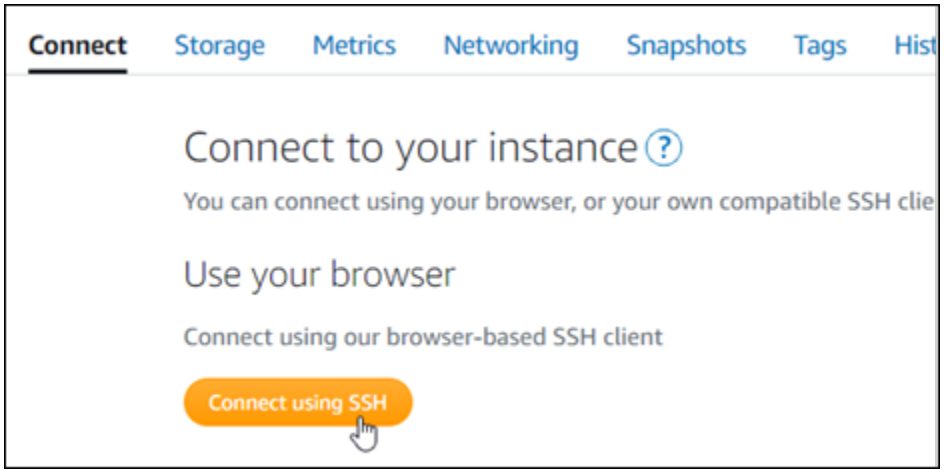
インスタンス管理ページの [ネットワーク] タブで、[静的 IP の作成] または [静的 IP のアタッチ] (インスタンスにアタッチできる静的 IP を既に作成している場合) を選択して、ページの手順に従います。

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 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 を有効にできますが、 PrestaShop ソフトウェアは IPv6 ネットワーク経由でリクエストに応答しません。

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する

次の手順を実行してウェブサイトにアクセスし PrestaShop 、管理ダッシュボードにサインインします。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 アドレスを書き留めます。

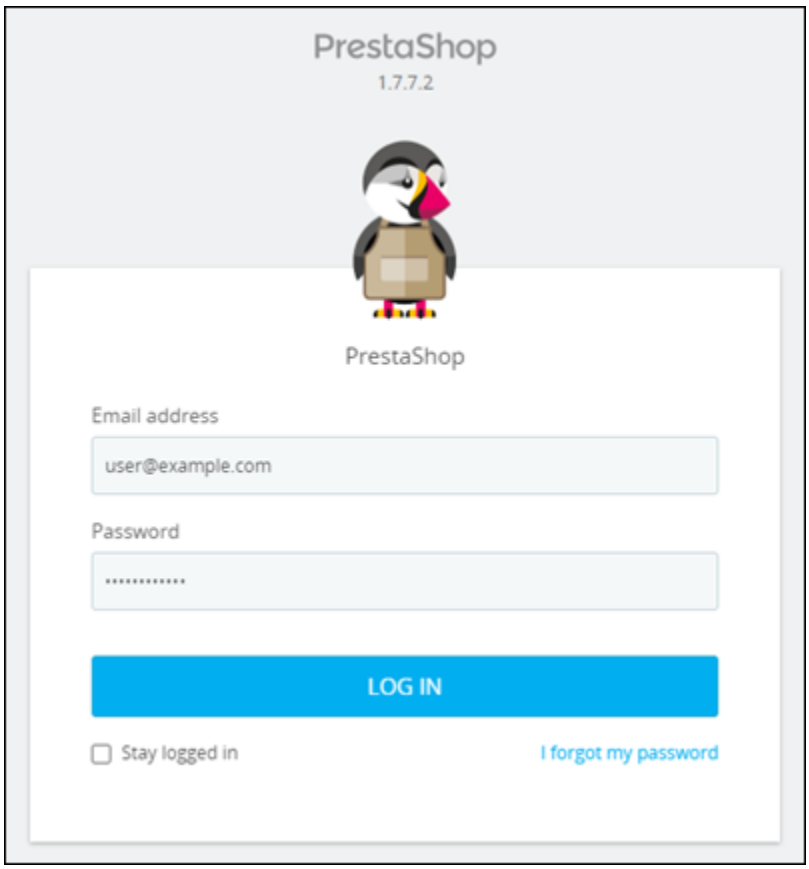
2. 次のアドレスを参照して、 PrestaShop ウェブサイトの管理ダッシュボードのサインインページにアクセスします。*#InstanceIpAddress#* をインスタンスのパブリック IP アドレスまたは静的 IP アドレスに置き換えてください。

   ```
   http://<InstanceIpAddress>/administration
   ```

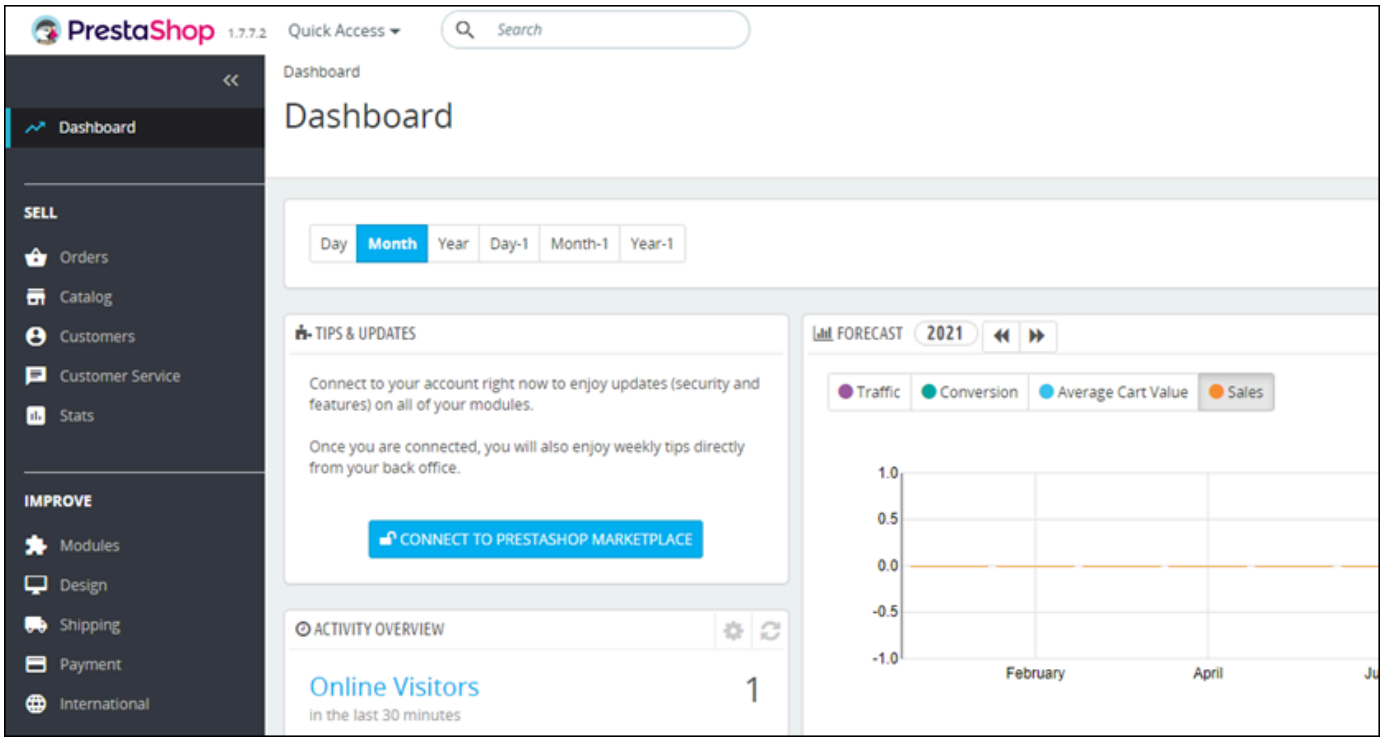
   例:

   ```
   http://203.0.113.0/administration
   ```

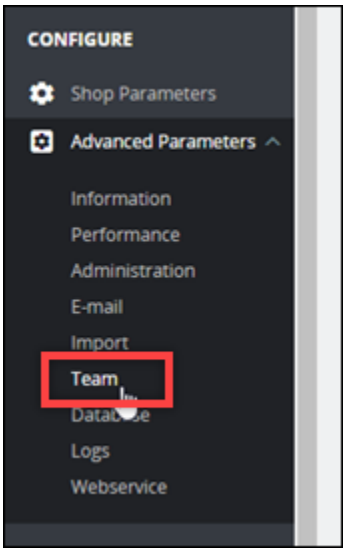
3. デフォルトのユーザー名 (`user@example.com`) と、このガイドの前のセクションで取得したデフォルトのアプリケーションパスワードを入力して [ログイン] を選択します。



PrestaShop 管理ダッシュボードが表示されます。



PrestaShop ウェブ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。詳細については、 PrestaShop ドキュメントの「 ユーザーガイド PrestaShop」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、「 PrestaShop ドキュメント」の「 ユーザーガイド PrestaShop」を参照してください。

ステップ 4: 登録済みドメイン名のトラフィックを PrestaShop ウェブサイトにルーティングする

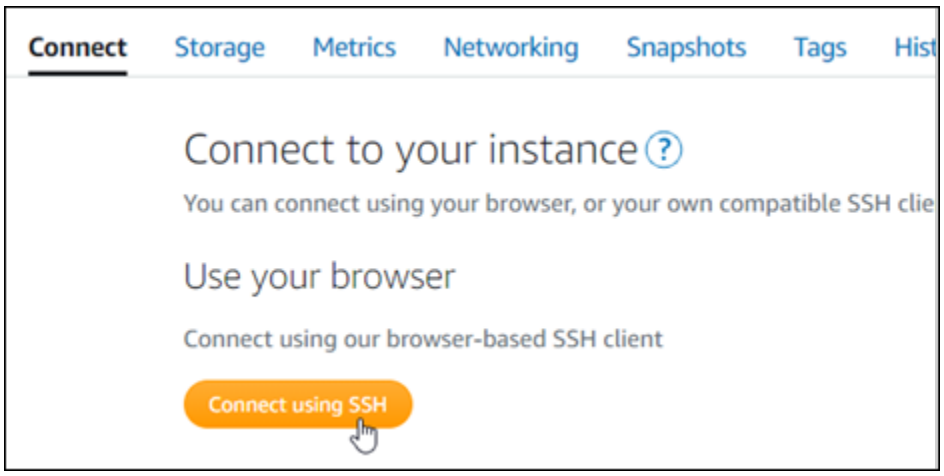
などの登録済みドメイン名のトラフィックを PrestaShop ウェブサイト`example.com`にルーティングするには、ドメインのドメインネームシステム (DNS) にレコードを追加します。DNS レコードは、通常、ドメインの登録先であるレジストラが管理またはホストします。ただし、ドメインの DNS レコードの管理を Lightsail に転送して、Lightsail コンソールを使用して管理できるようにすることをお勧めします。

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 DNS ゾーンの作成」を参照してください。

ドメイン名がインスタンスにトラフィックをルーティングしたら、次の手順を実行して、PrestaShop 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#* 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しているはずです。

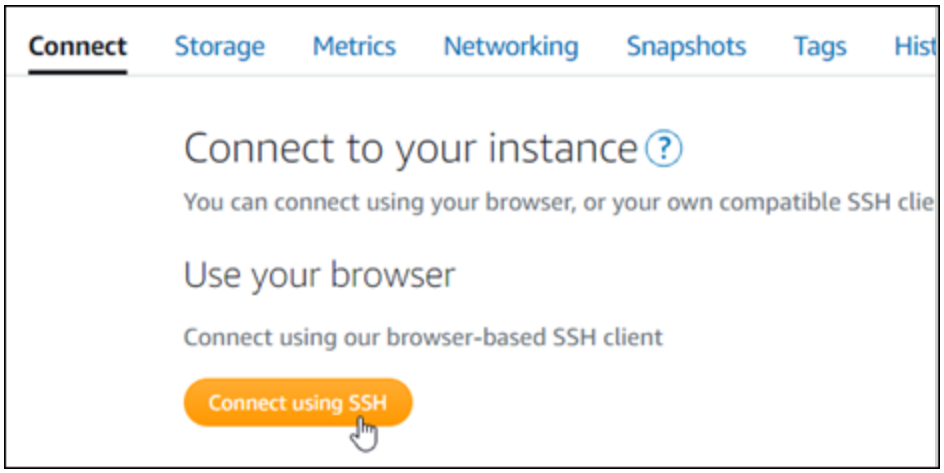
```
bitnami@ip-[redacted]:~$ sudo /opt/bitnami/configure_app_domain --domain www.example.com
Configuring domain to www.example.com
2021-03-12T15:49:22.000Z - info: Saving configuration info to disk
prestashop 15:49:22.41 INFO  ==> Trying to connect to the database server
prestashop 15:49:22.44 INFO  ==> 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 ウェブサイトの HTTPS を設定する

PrestaShop ウェブサイトで HTTPS を設定するには、次のステップを実行します。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 PrestaShop 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。これらのステップを開始する前に、 PrestaShop ウェブサイトで使用するすべてのドメインの DNS に DNS レコードを追加してください。

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。



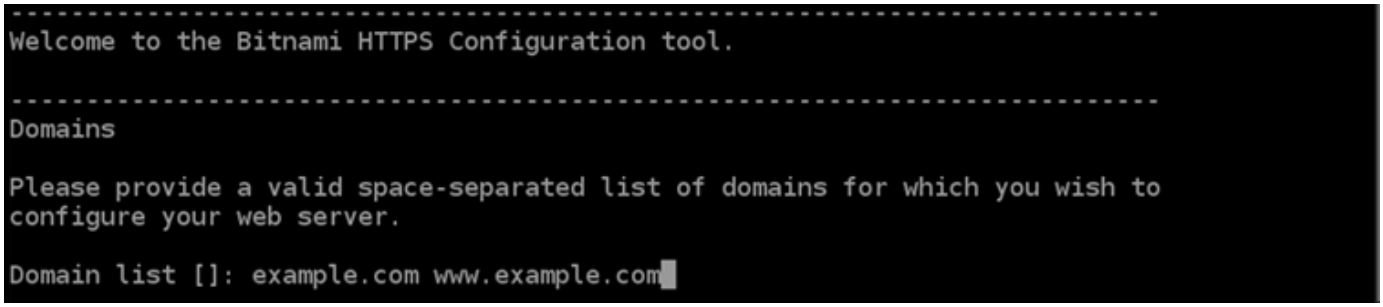
2. 接続後に、次のコマンドを入力して bncert-tool をスタートします。

```
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
--------------------------------------------------------------------------
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。次の一連のステップに進み、 PrestaShop ウェブサイトで HTTPS の有効化を完了します。

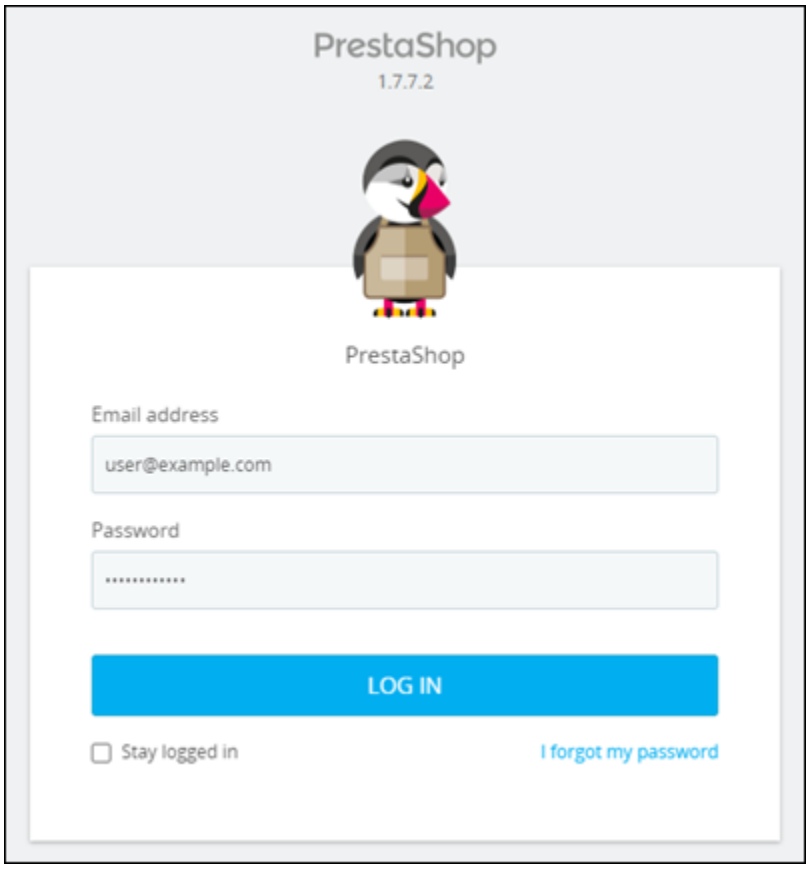
8. 次のアドレスを参照して、 PrestaShop ウェブサイトの管理ダッシュボードのサインインページにアクセスします。*#DomainName#* は、インスタンスにトラフィックをルーティングしている登録済みドメイン名に置き換えてください。

```
http://<DomainName>/administration
```

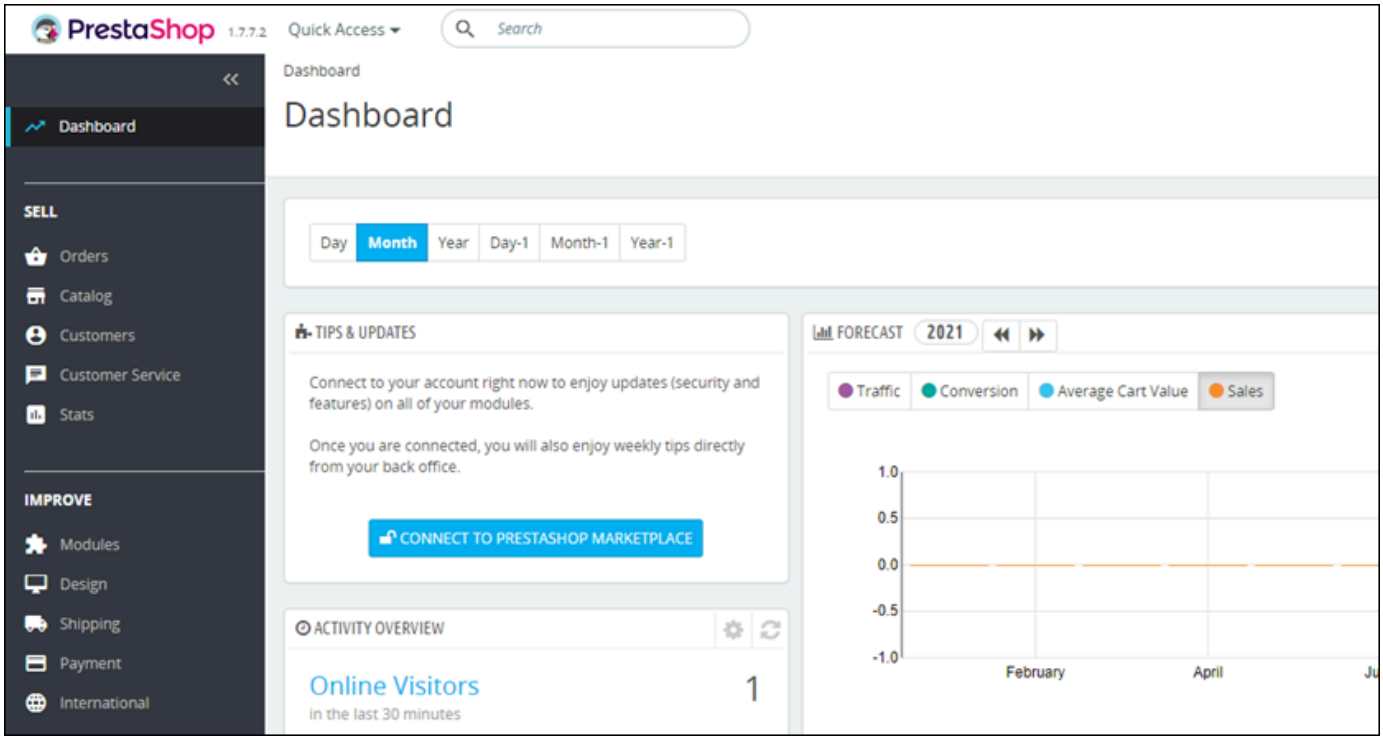
例:

```
http://www.example.com/administration
```

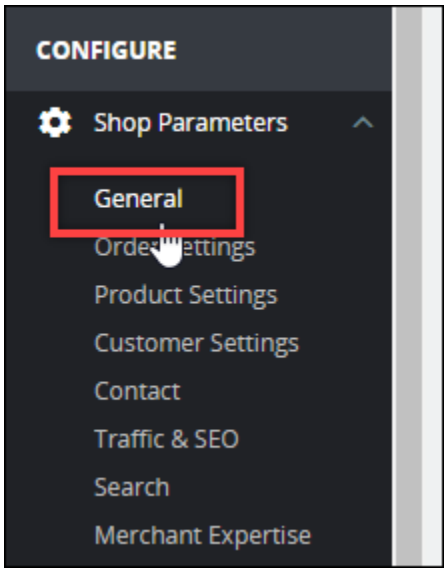
9. デフォルトのユーザー名 (user@example.com) と、このガイドの前のセクションで取得したデフォルトのアプリケーションパスワードを入力して [ログイン] を選択します。



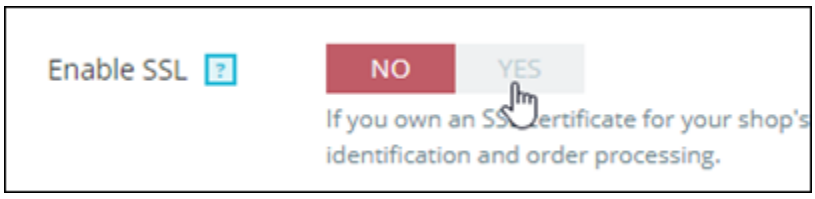
PrestaShop 管理ダッシュボードが表示されます。



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。

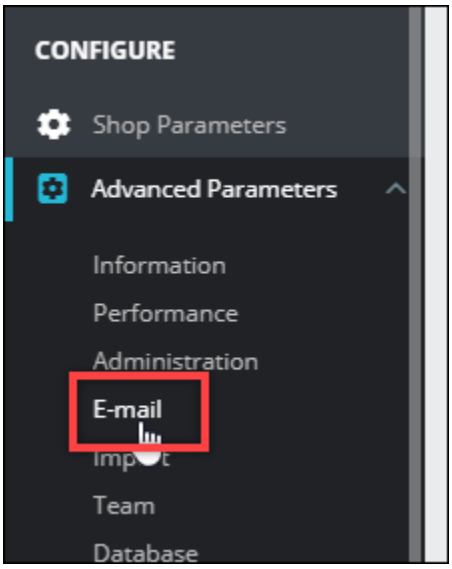
13. [General] ページが再読み込みしたら、[すべてのページでSSLを有効にする] の横にある [はい] を選択します。

14. ページの下部までスクロールし、[保存] を選択します。

    HTTPS が PrestaShop ウェブサイト用に設定されました。顧客が PrestaShop ウェブサイトの HTTP バージョン (例: `http://www.example.com`) を参照すると、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

PrestaShop ウェブサイトの SMTP 設定を設定して、E メール通知を有効にします。そのためには、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。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、 PrestaShop ドキュメントの「 ユーザーガイド PrestaShop」および Bitnami ドキュメントの「アウトバウンド E メールの 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 を設定する場合は、Lightsail コンソールで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 へのサインインに使用する標準パスワードではなく、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 の PrestaShop ドキュメントを読む

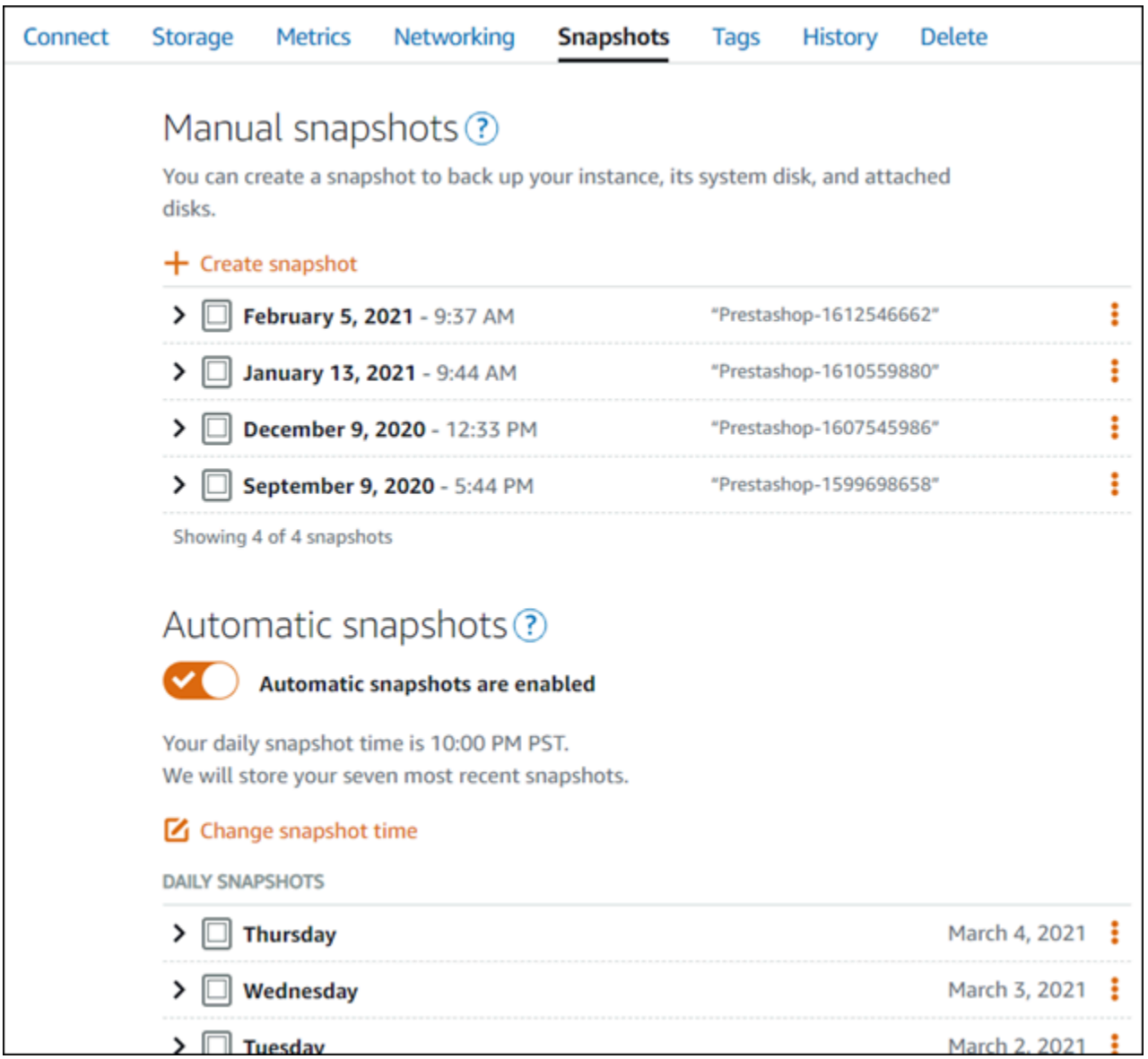
Bitnami のドキュメントを読んで、プラグインのインストールやテーマのカスタマイズなど、PrestaShop インスタンスとウェブサイトで管理タスクを実行する方法を確認してください。詳細については、Bitnami PrestaShop ドキュメントの「AWS クラウド用の Bitnami スタック」を参照してください。

また、 PrestaShop ウェブサイトの管理方法については、 PrestaShop ドキュメントもお読みください。詳細については、 PrestaShop ドキュメントの「 ユーザーガイド PrestaShop」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。

Lightsail で Redmine インスタンスを設定して保護する

Redmine インスタンスが Amazon Lightsail で起動および実行された後に開始するために実行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 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

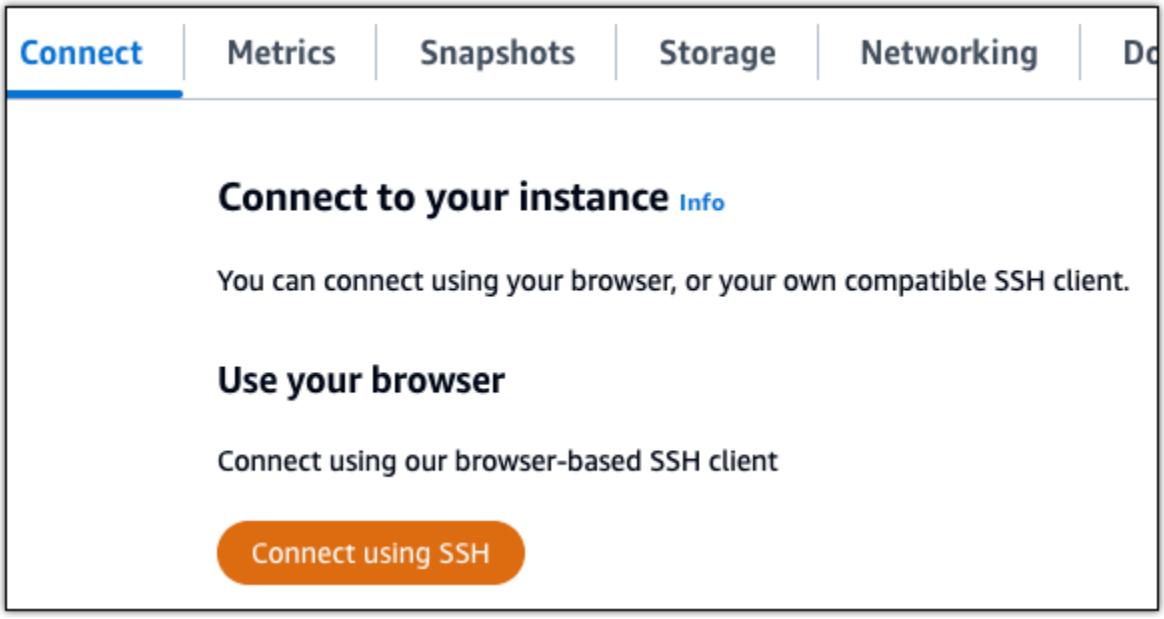
ステップ 1: Bitnami のドキュメントを確認する

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」を参照してください。

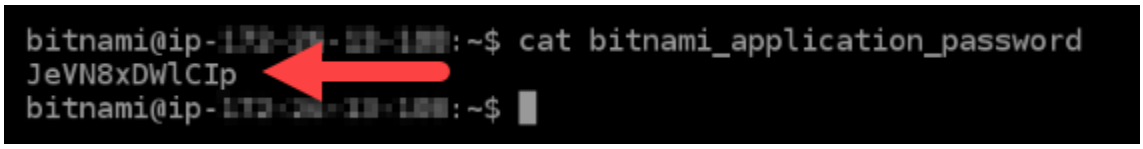
1. インスタンス管理ページの [接続] タブで、[SSH を使用して接続] を選択します。



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
```

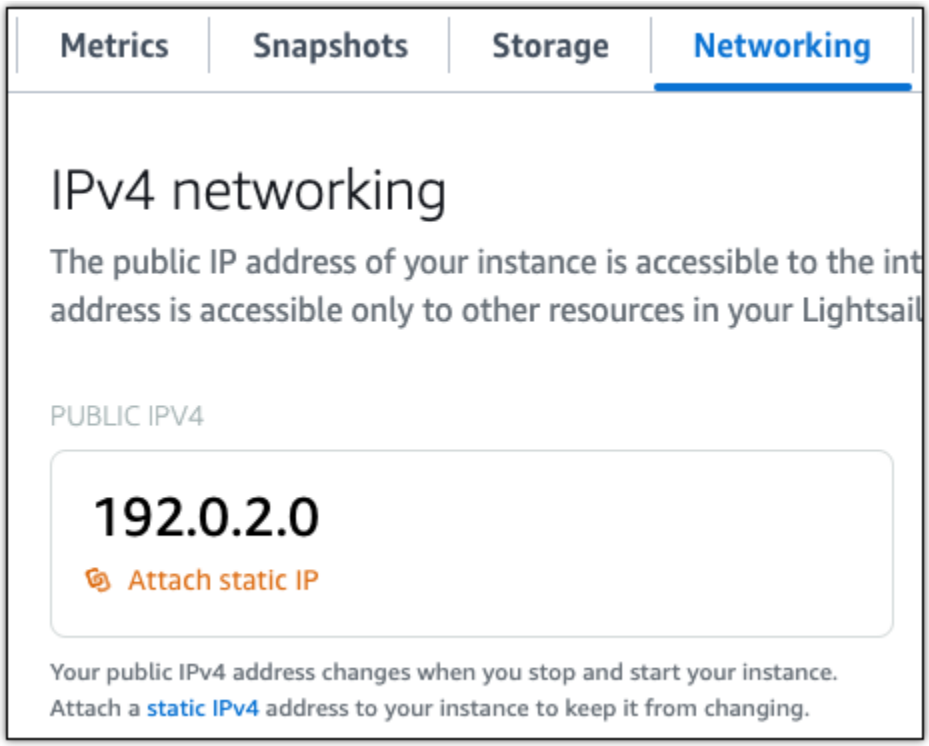
アプリケーションのデフォルトパスワードを含んだ、次の例のようなレスポンスが表示されます。



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)。

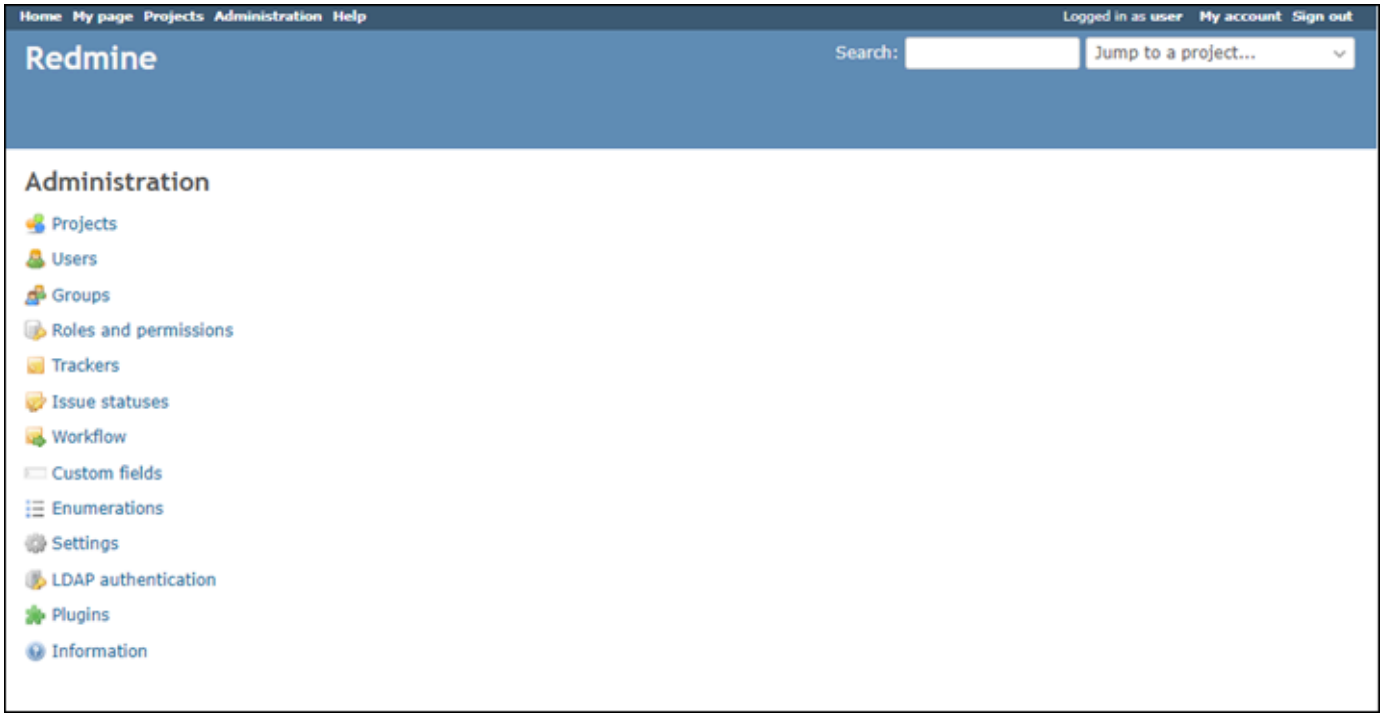
   Redmine ウェブサイトのホームページが表示されます。

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 に転送して、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 DNS ゾーンの作成」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。

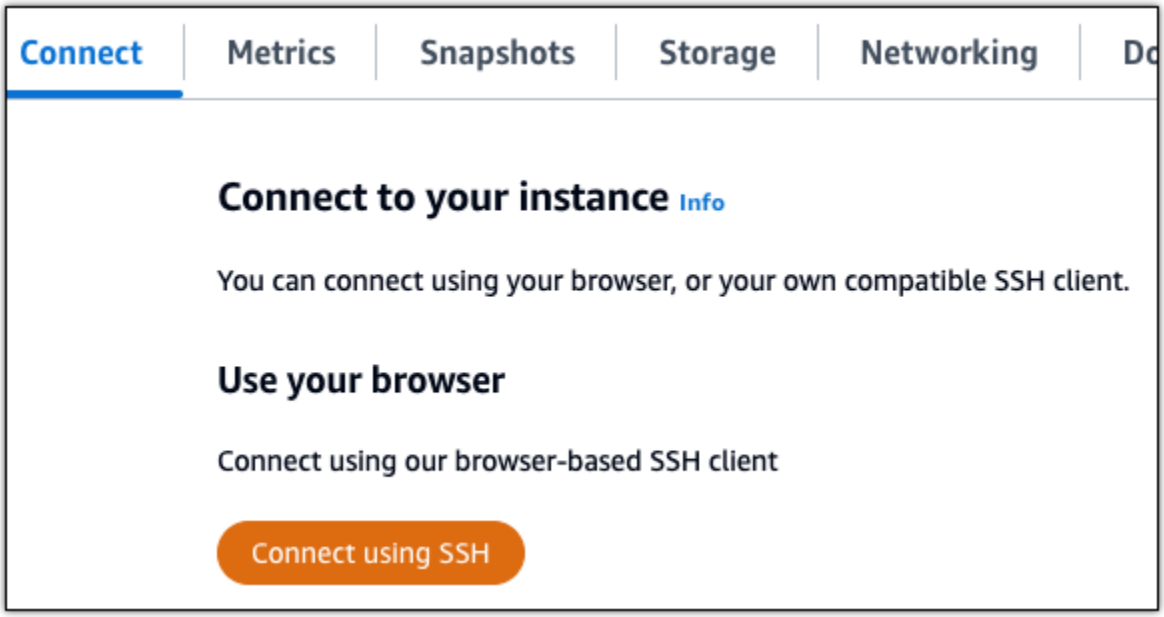
ステップ 6: Redmine ウェブサイトの HTTPS を設定する

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
```

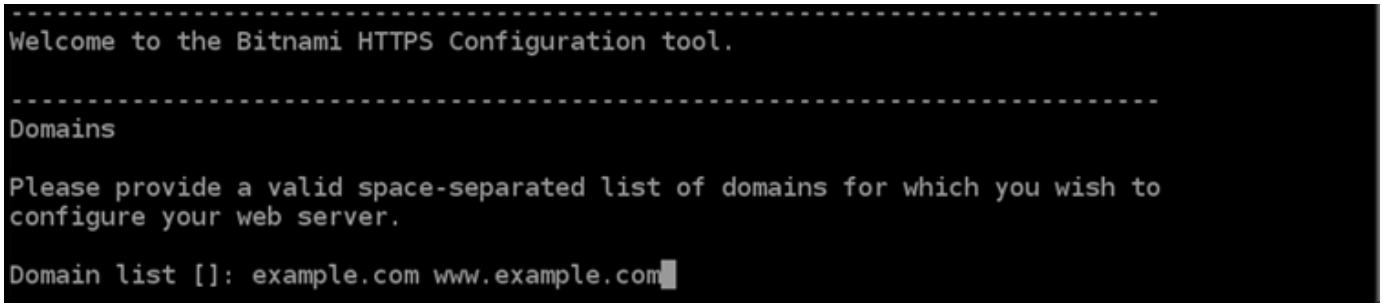
これでインスタンスに bncert ツールをインストールする手順は完了です。

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。

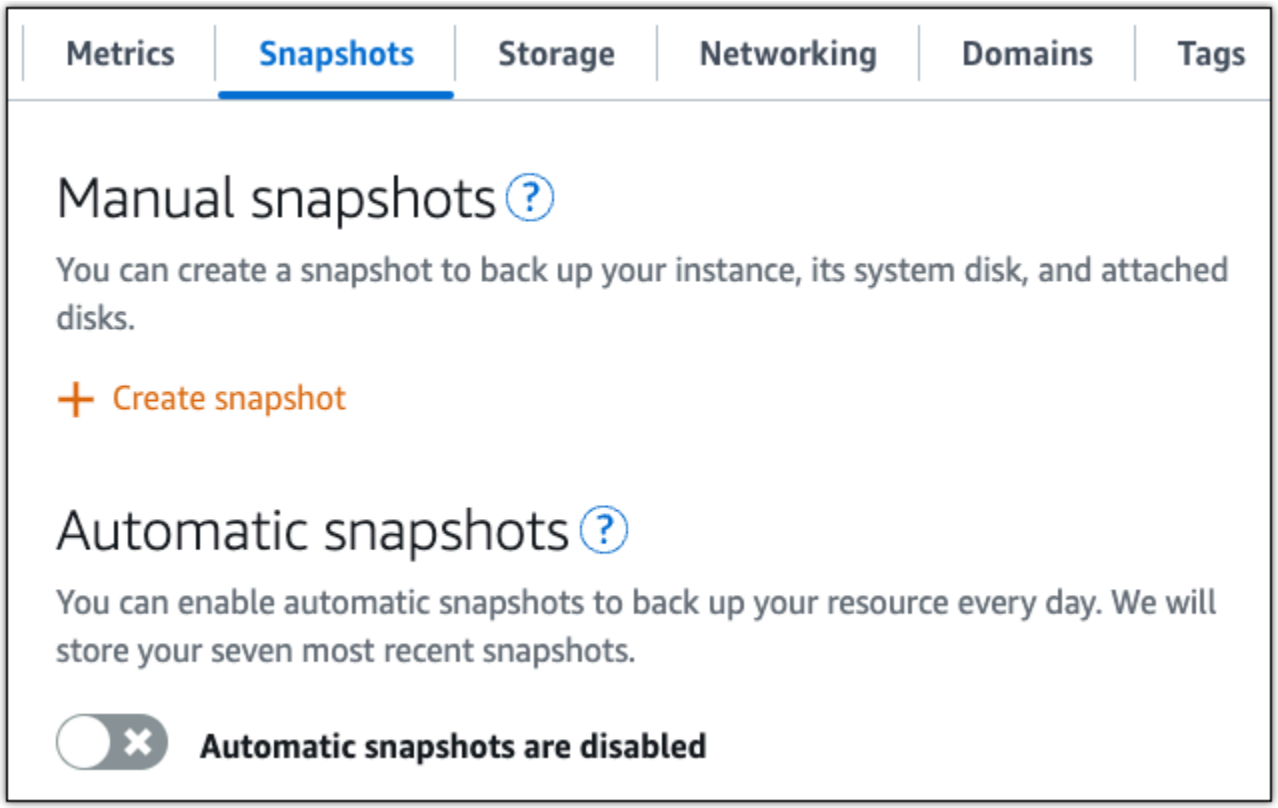
ステップ 7: Redmine のドキュメントを読み、引き続きウェブサイトの設定を続行する

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。

Lightsail WordPress で を起動して設定する

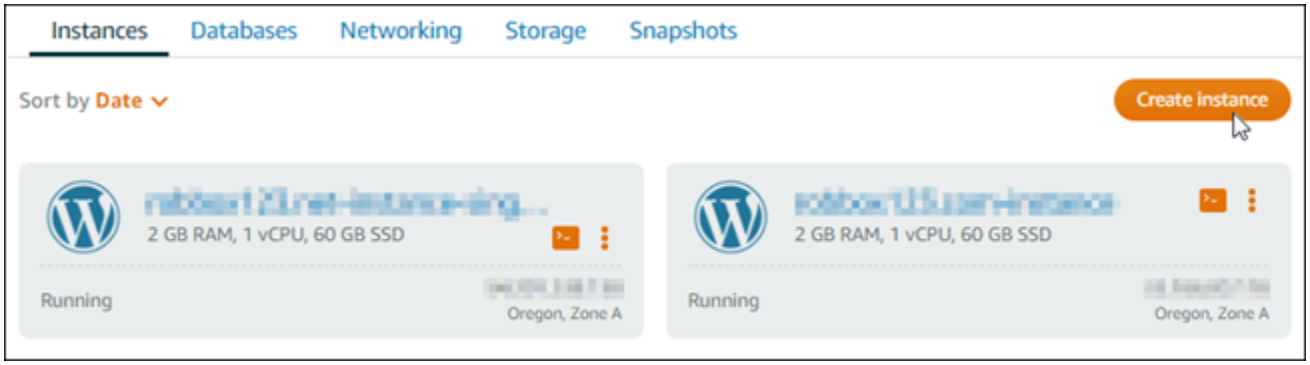
このクイックスタートガイドでは、Amazon Lightsail で WordPress インスタンスを起動して設定する方法について説明します。

ステップ 1: インスタンスを作成する WordPress

WordPress インスタンスを起動して実行するには、次のステップを実行します。

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. 次のようにインスタンスのイメージを選択します。

   a. プラットフォームの選択 で、Linux/Unix を選択します。

   b. 設計図の選択 で、 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のマシン設定 (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 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. [インスタンスの作成] を選択します。

8. テストブログ記事を表示するには、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の投稿が表示されます。

ステップ 2: インスタンスを設定する WordPress

WordPress インスタンスは、以下を設定するガイド付き step-by-step ワークフローを使用して設定できます。

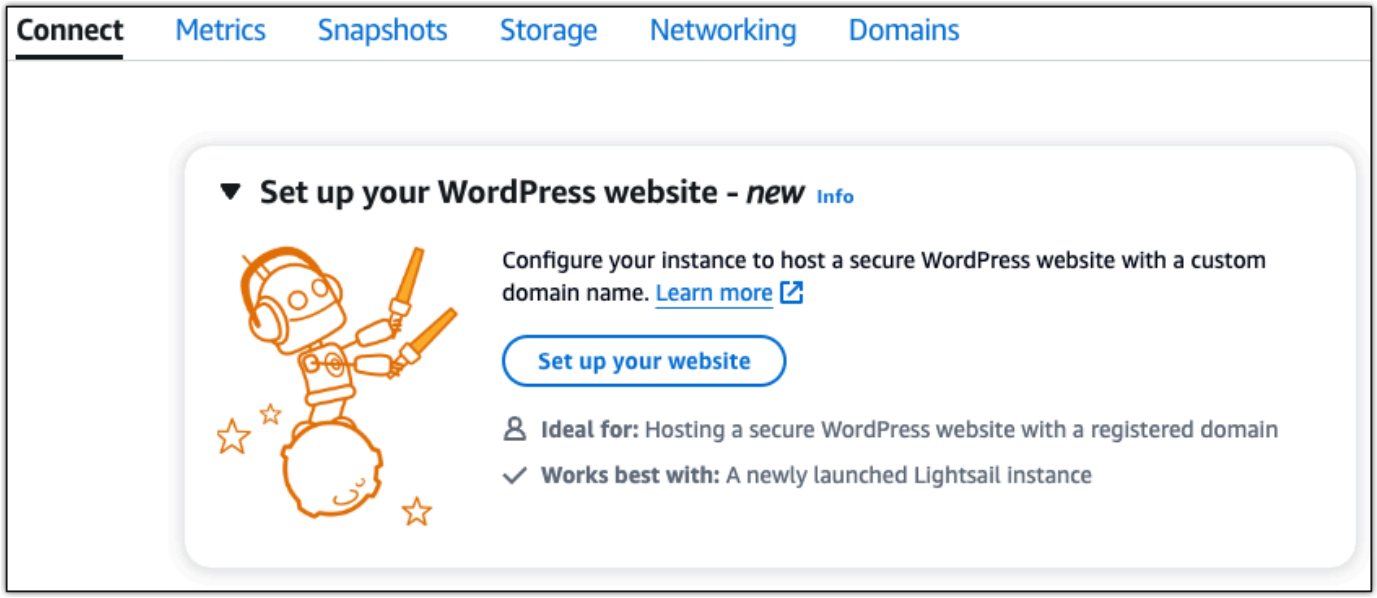
- 登録済みドメイン名 – WordPress サイトには覚えやすいドメイン名が必要です。ユーザーは、WordPress サイトにアクセスするためにこのドメイン名を指定します。詳細については、「ドメインと DNS」を参照してください。
- DNS 管理 – ドメインの DNS レコードを管理する方法を決定する必要があります。DNS レコードは、ドメインまたはサブドメインが関連付けられている IP アドレスまたはホスト名を DNS サーバーに伝えます。DNS ゾーンには、ドメインの DNS レコードが含まれます。詳細については、「the section called “DNS Lightsail の”」を参照してください。
- 静的 IP アドレス – WordPress 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ます。静的 IP アドレスをインスタンスにアタッチしても、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、「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 ウェブサイトで HTTPS を有効にして、登録済みドメインを介してインスタンスにルーティングされるトラフィックを HTTPS を使用して暗号化できます。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**
>
> 開始する前に、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の詳細については、「セットアップのトラブルシューティング WordPress 」を参照してください。
>
> - セットアップは、 WordPress 2023 年 1 月 1 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。
> - セットアップ中に実行される Certbot 依存関係ファイル、HTTPS 書き換えスクリプト、および証明書更新スクリプトは、インスタンスの `/opt/bitnami/lightsail/scripts/` ディレクトリに保存されます。
> - インスタンスは実行状態である必要があります。インスタンスが起動したばかりの場合、SSH 接続の準備が完了するまで数分かかります。

- インスタンスファイアウォールのポート 22、80、および 443 は、セットアップの実行中に任意の IP アドレスからの TCP 接続を許可する必要があります。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) とそのwwwサブドメイン () からのトラフィックをポイントする DNS レコードを追加または更新するときwww.example.comは、インターネット全体に伝達する必要があります。nslookup や からの DNS Lookup などのツールを使用して、DNS の変更が有効になったことを確認できます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、ウェブサイトの設定が失敗する原因となる、非推奨の Certbot Personal Package Archive (PPA) リポジトリが含まれている場合があります。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、インスタンス上の次の場所にバックアップされます: ~/opt/bitnami/lightsail/repo.backup。非推奨の PPA の詳細については、正規ウェブサイトの「Certbot PPA」を参照してください。
- Let's Encrypt 証明書は 60 ~ 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 分かかる場合があります。インスタンス接続タブで各ステップの進行状況を表示できます。

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、ウェブサイトのセットアップ を選択します。



2. ドメイン名 を指定するには、既存の Lightsail マネージド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ます。次のステップに進むには、この DNS ゾーンを使用する を選択します。
   - サードパーティードメインを選択して、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します。後で使用する場合に備えて、Lightsail アカウントに一致する DNS ゾーンを作成することに注意してください。サードパーティーの DNS を使用する を選択して、次のステップに進みます。
4. 「静的 IP アドレスの作成」で、静的 IP アドレスの名前を入力し、「静的 IP の作成」を選択します。
5. ドメイン割り当ての管理 で、割り当ての追加 を選択し、ドメインタイプを選択し、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 証明書を作成する」で、ドメインとサブドメインを選択し、E メールアドレスを入力し、Lightsail にインスタンス で Let's Encrypt 証明書を設定することを承認し、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。

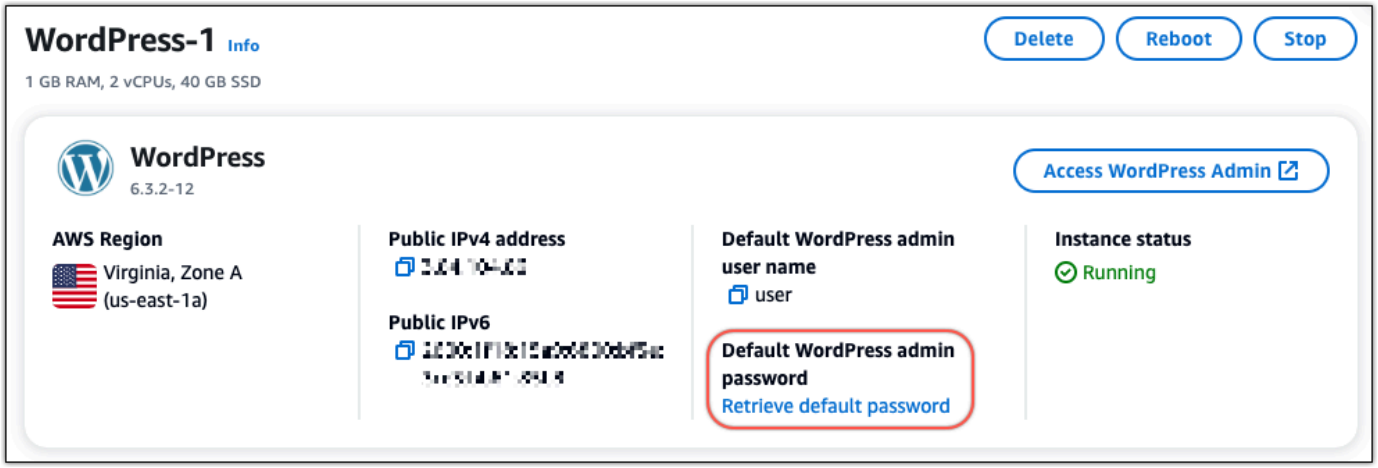
   セットアップ中は、インスタンスを停止したり変更したりしないでください。インスタンスの設定には最大 15 分かかる場合があります。インスタンス接続タブで各ステップの進行状況を表示できます。
7. ウェブサイトの設定が完了したら、ドメイン割り当てステップで指定した URLsが WordPress サイトを開くことを確認します。

ステップ 3: WordPress 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

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 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 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。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_application_password
JKzh8wB5FAR!
```

ステップ 4: ウェブサイトにサインインする WordPress

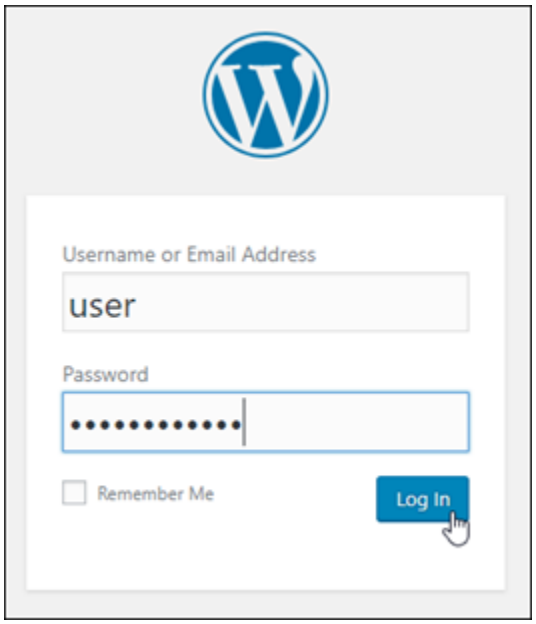
デフォルトのユーザーパスワードを取得したら、 WordPress ウェブサイトのホームページに移動し、管理ダッシュボードにサインインします。サインイン後に、デフォルトのパスワードを変更できます。

管理ダッシュボードにサインインするには

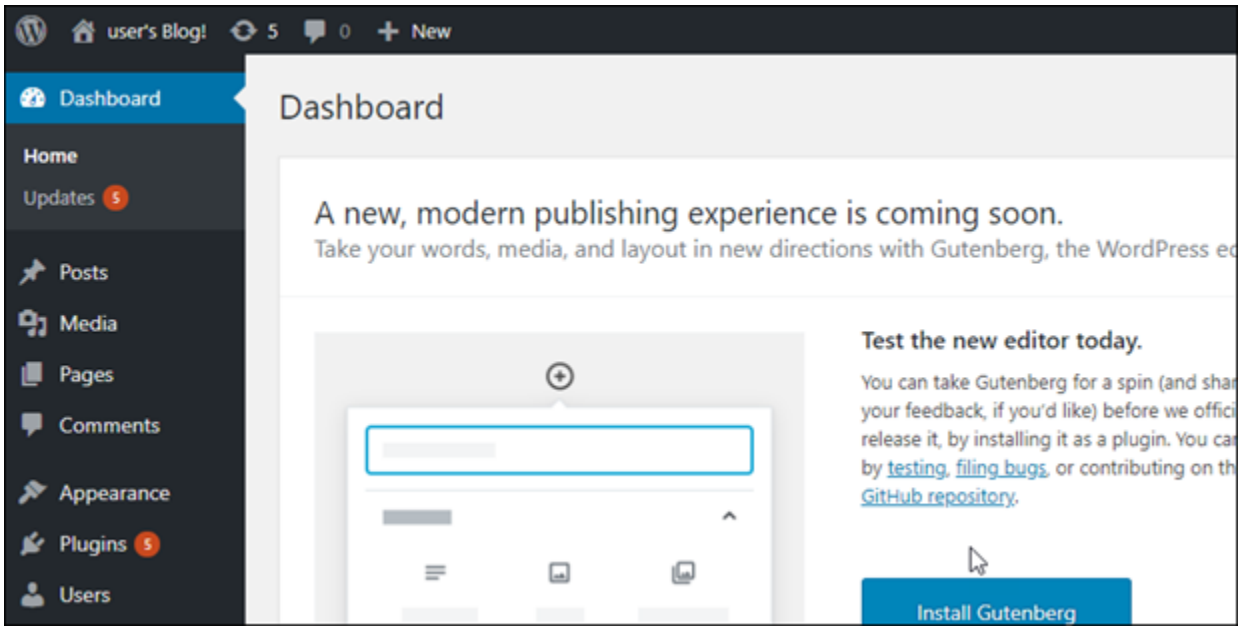
1. インスタンスの WordPress インスタンス管理ページを開きます。
2. WordPress パネルで、アクセス WordPress 管理者 を選択します。
3. WordPress 管理者ダッシュボードへのアクセス パネルのパブリック IP アドレスを使用 で、次の形式のリンクを選択します。

   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 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、 のバージョンのアップグレードなど、 WordPress ウェブサイトで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 の Bitnami AWS クラウドを参照してください。

Lightsail でマルチサイトを設定する WordPress

Amazon Lightsail で WordPress マルチサイトインスタンスを起動して実行した後、開始するために実行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

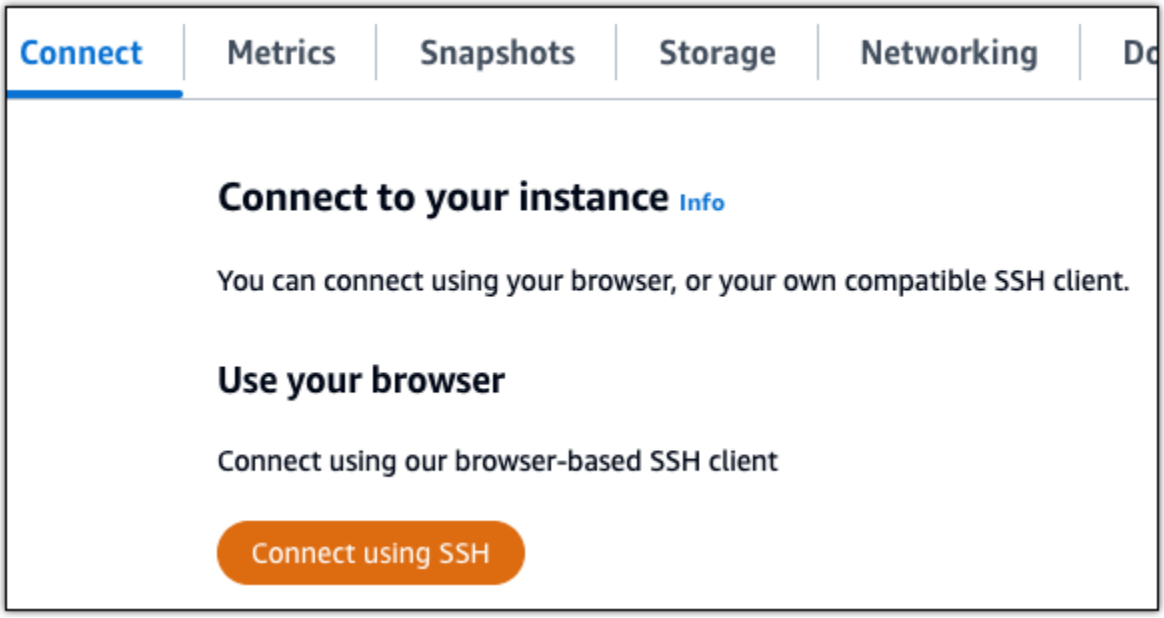
ステップ 1: Bitnami のドキュメントを確認する

マルチサイトインスタンスの設定方法については、Bitnami WordPress のドキュメントを参照してください。詳細については、「 用 WordPress Bitnami によってパッケージ化されたマルチサイト AWS クラウド」を参照してください。

ステップ 2: WordPress 管理ダッシュボード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

WordPress マルチサイトウェブサイトの管理ダッシュボード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」を参照してください。

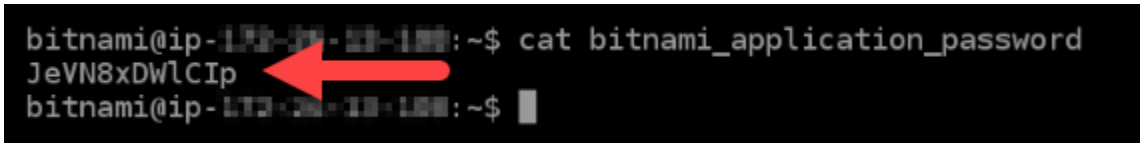
1. インスタンス管理ページの [接続] タブで、[SSH を使用して接続] を選択します。



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 マルチサイト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。

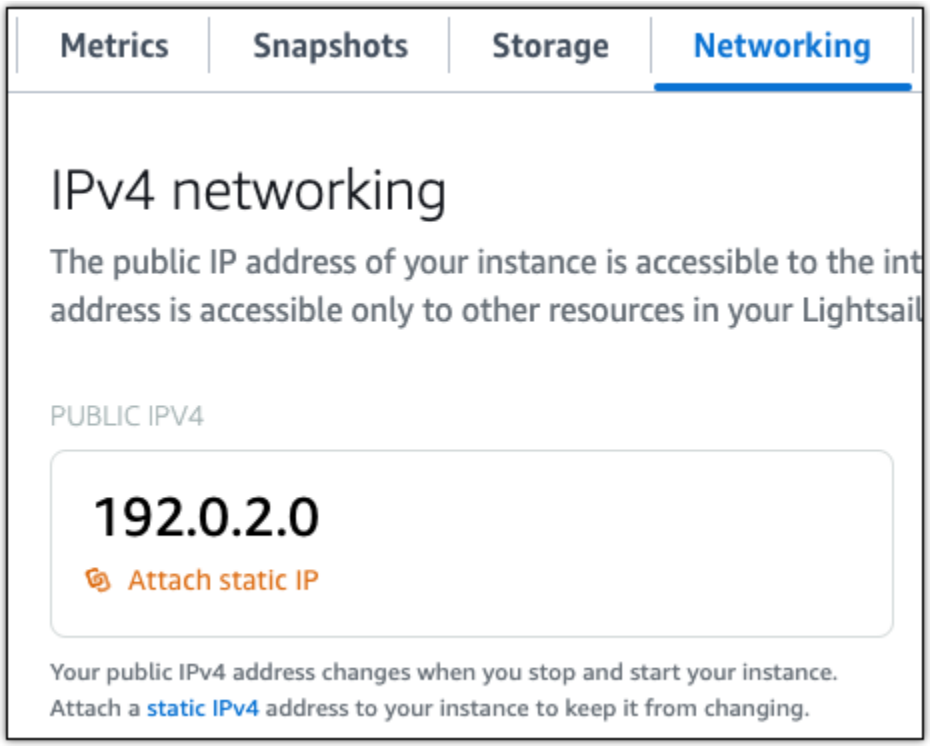


ステップ 3: インスタンスに静的 IP アドレスをアタッチする

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。

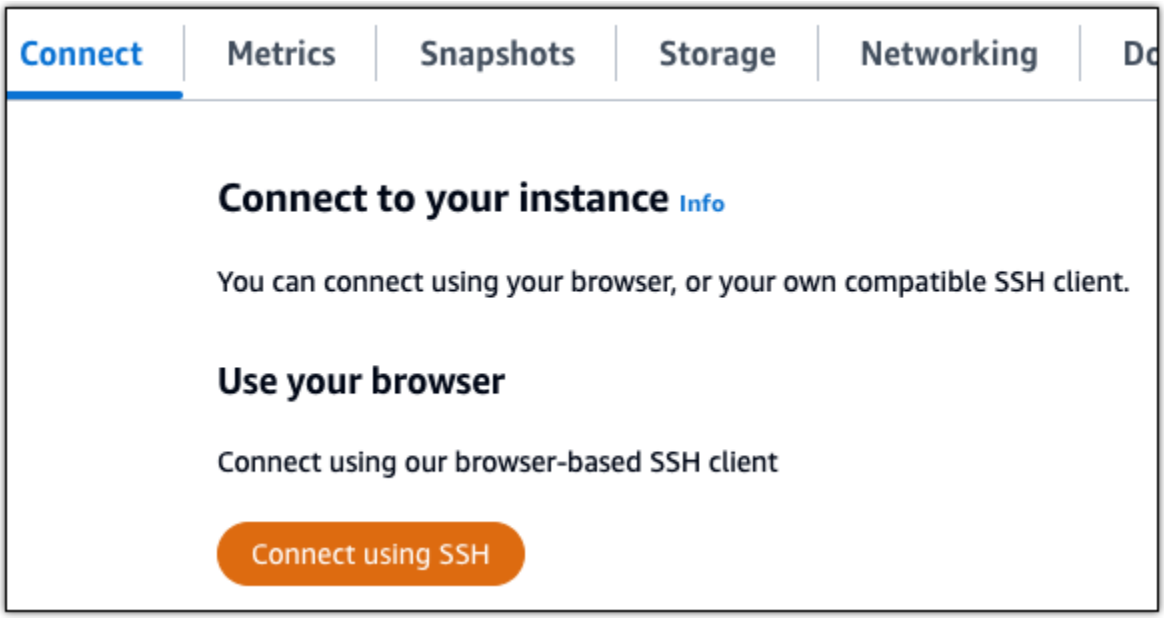
インスタンス管理ページの [ネットワーク] タブで、[静的 IP の作成] または [静的 IP のアタッチ] (インスタンスにアタッチできる静的 IP を既に作成している場合) を選択して、ページの手順に従います。詳細については、「静的 IP を作成してインスタンスにアタッチする」を参照してください。



新しい静的 IP アドレスがインスタンスにアタッチされたら、次の手順を実行して、新しい静的 IP アドレス WordPress を認識する必要があります。

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 WordPress、インスタンスのウェブサイトに新しい静的 IP アドレスが認識されるはずです。

```
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
```

このコマンドが失敗した場合、古いバージョンの WordPressマルチサイト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 マルチサイトウェブサイトの管理ダッシュボードにサインインする

デフォルトのアプリケーションパスワードを取得したら、次の手順を実行して WordPress マルチサイト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。でできることの詳細については WordPress、このガイドの後半にある「ステップ 7: WordPress マルチサイトドキュメントを読み、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)。

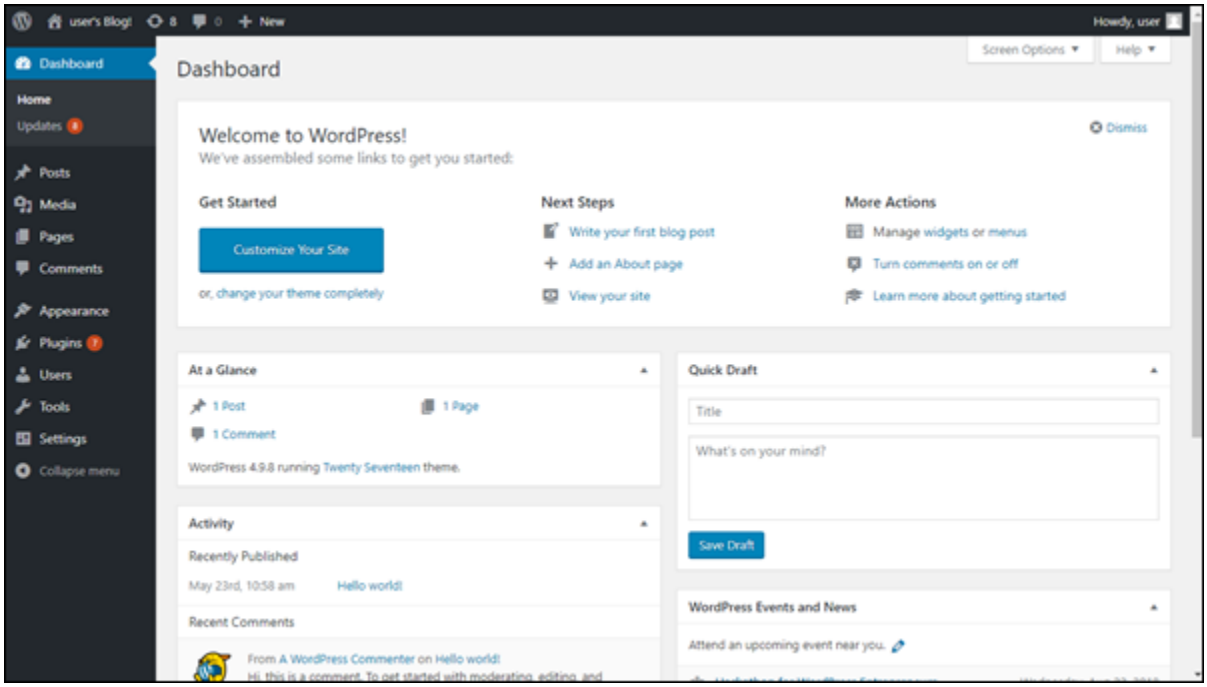
   WordPress ウェブサイトのホームページが表示されます。

3. WordPress ウェブサイトのホームページの右下隅にある管理を選択します。

   [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 管理ダッシュボードが表示されます。



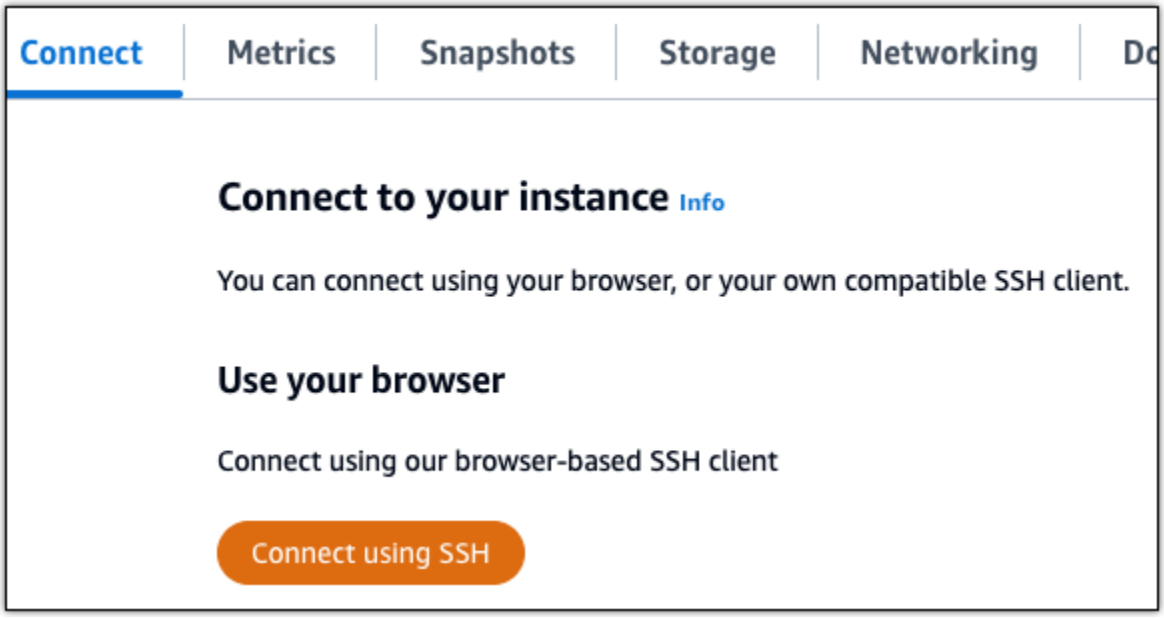
ステップ 5: 登録済みドメイン名のトラフィックを WordPress マルチサイトウェブサイトにルーティングする

などの登録済みドメイン名のトラフィックを WordPress マルチサイトウェブサイトexample.comにルーティングするには、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 に転送して、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 DNS ゾーンの作成」を参照してください。

ドメイン名がインスタンスにトラフィックをルーティングしたら、次の手順を実行してドメイン名 WordPress を認識する必要があります。

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 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マルチサイトソフトウェアはドメイン名を認識しているはずです。

```
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
```

このコマンドが失敗した場合、古いバージョンの WordPressマルチサイト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するドメイン名に置き換えます。

```
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 マルチサイトウェブサイトのメインブログにリダイレクトされます。次に、 WordPress マルチサイト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、このガイドの「マルチ WordPress サイト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」セクションに進んでください。

ステップ 6: ブログをドメインまたはサブドメインとして WordPress マルチサイトウェブサイトに追加する

WordPress Multisite は、 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 WordPress。新しいブログウェブサイトを WordPress マルチサイトに追加すると、独自のドメインまたは WordPress マルチサイト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。これらのオプションのいずれかのみを使用するように WordPress マルチサイトを設定できます。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 などのドメインとして追加するには、「Lightsail の WordPress マルチサイトインスタンスにブログをドメインとして追加するexample2.com」を参照してください。
- one.example.com や など、 WordPress マルチサイトの主要ドメインのサブドメインとしてブログサイトを追加するにはtwo.example.com、「Lightsail の WordPress マルチサイト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。

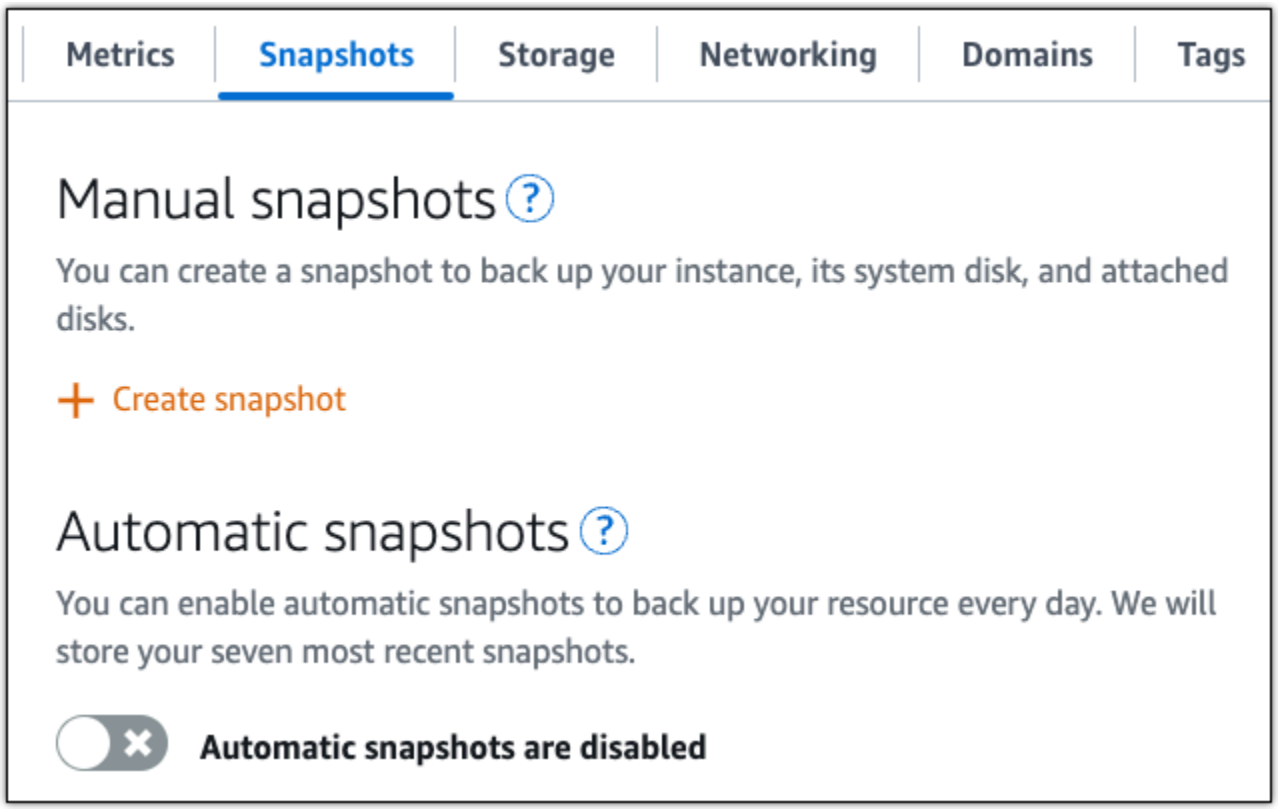
ステップ 7: WordPress マルチサイトドキュメントを読み、ウェブサイトの設定を続行する

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、 WordPress「マルチサイトドキュメント」を参照してください。詳細については、WordPress「マルチサイト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 マルチサイトウェブサイトを目的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。

Lightsail で Bitnami アプリケーションとスタックを操作する

このセクションでは、Amazon Lightsail インスタンスの Bitnami アプリケーションに関連する以下のトピックについて説明します。

トピック

Lightsail Bitnami インスタンスのデフォルトのアプリケーションユーザー名とパスワードを取得する

Bitnami には、仮想プライベートサーバーである Amazon Lightsail インスタンス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またはブループリントが多数用意されています。これらのブループリントは、Lightsail コンソールのインスタンス作成ページで「Bitnami によってパッケージ化」として記述されます。

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、次のブループリントから作成された 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について説明します。

- WordPress ブログとコンテンツ管理アプリケーション
- WordPress 同じインスタンス上の複数のウェブサイトをサポートするマルチサイト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

Bitnami ブループリントを使用して作成された Lightsail インスタンスのデフォルトのアプリケーションとデータベースのユーザー名は次のとおりです。

> **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 マルチサイトを含む
  - 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`

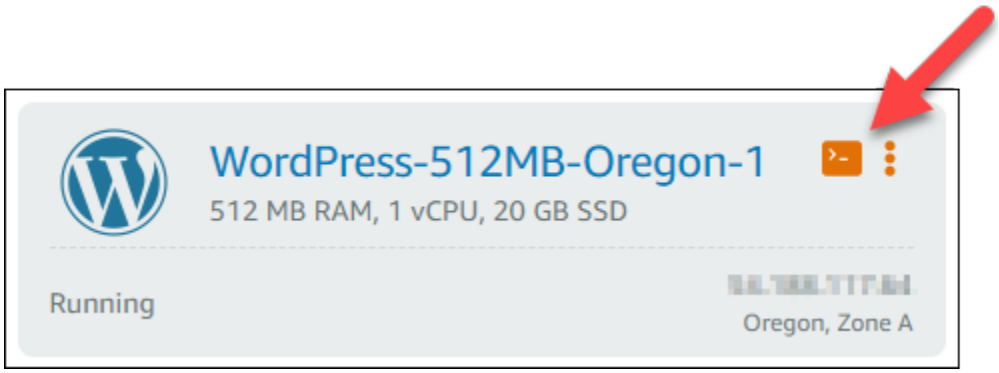
- Magento
  - アプリケーションユーザー名: `user`
  - データベースユーザー名: `root`
- MEAN
  - アプリケーションユーザー名: N/A
  - データベースユーザー名: `root`
- Drupal
  - アプリケーションユーザー名: `user`
  - データベースユーザー名: `root`
- GitLab CE
  - アプリケーションユーザー名: `user`
  - データベースユーザー名: `postgres`
- Redmine
  - アプリケーションユーザー名: `user`
  - データベースユーザー名: `root`
- 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

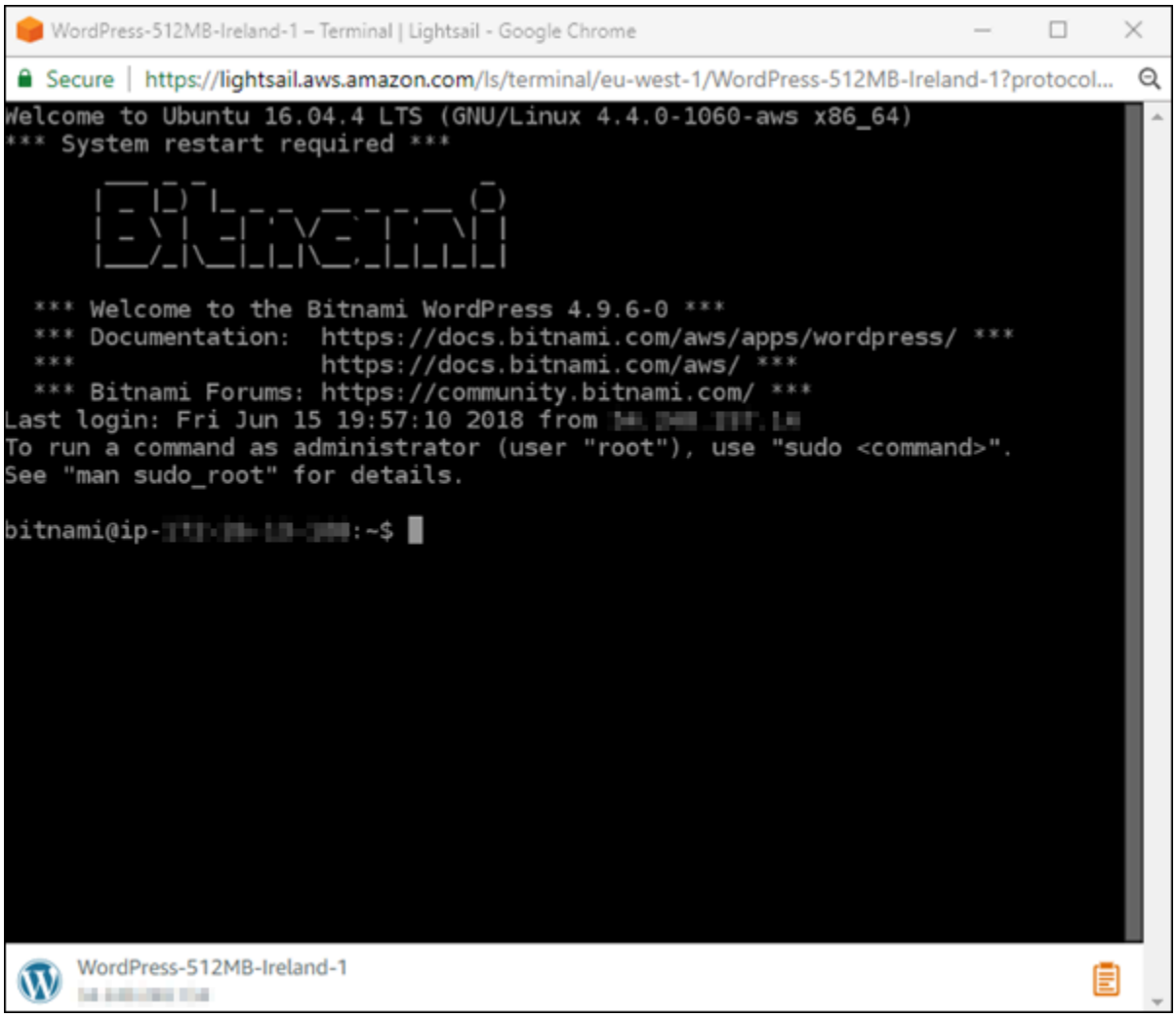
アプリケーションおよびデータベースのデフォルトパスワードはインスタンスに保存されています。Lightsail コンソールのブラウザベースのSSHターミナルを使用して接続し、特別なコマンドを実行して取得します。

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 の作成VPS」を参照してください。
3. Lightsail ホームページで、接続するインスタンスのクイック接続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SSHクライアントウィンドウが開きます。

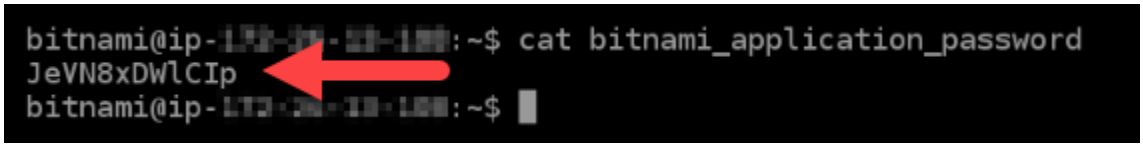


4. 次のコマンドを入力してデフォルトのアプリケーションのパスワードを取得します。

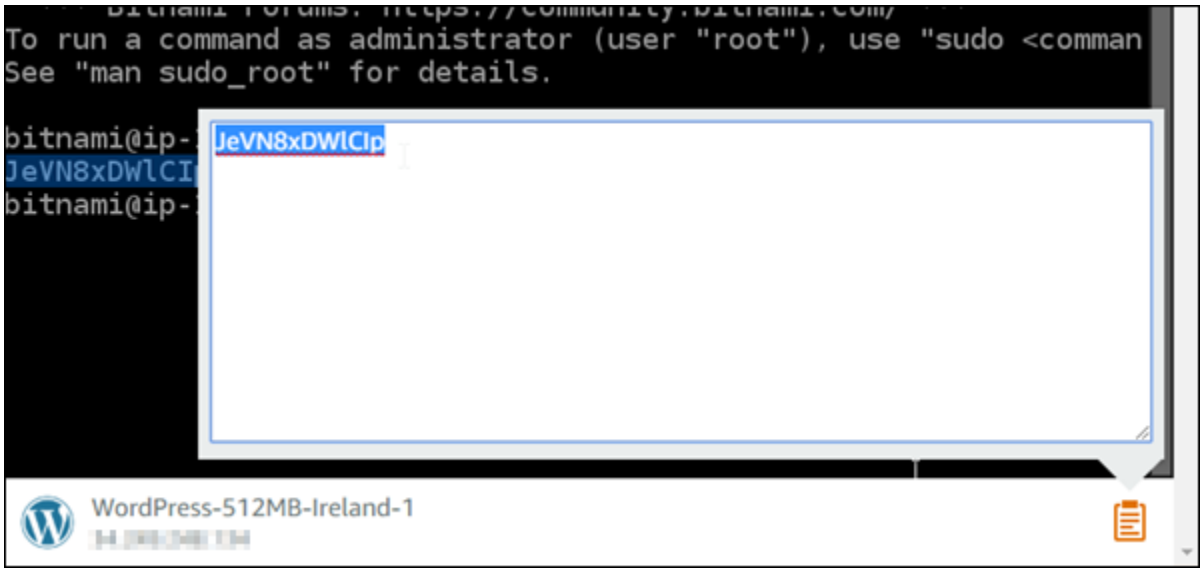
```
cat bitnami_application_password
```

> **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隅にある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 の設計図から作成され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

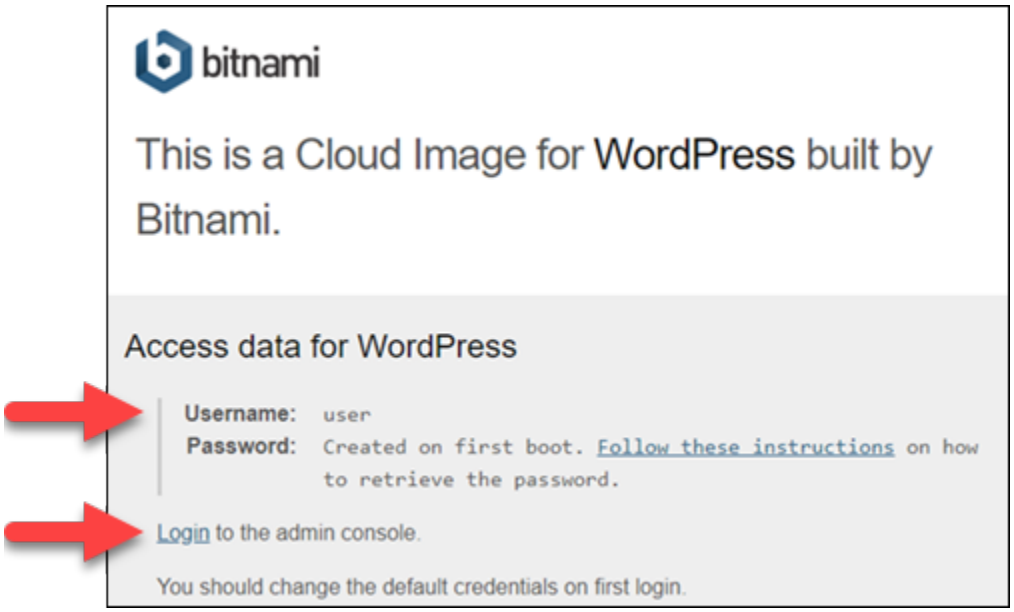
1. ブラウザウィンドウで、インスタンスのパブリック IP アドレスに移動します。

   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。例えば、アプリケーションのホームページは次のとおりです WordPress。

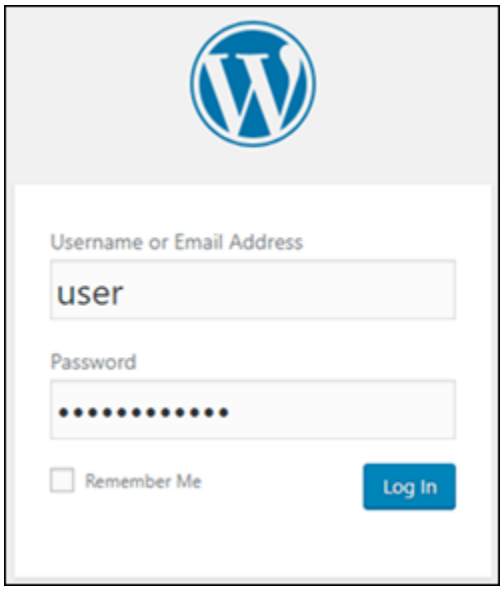
2. アプリケーションのホームページの右下にある Bitnami ロゴを選択し、アプリケーション情報ページに移動します。

> **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

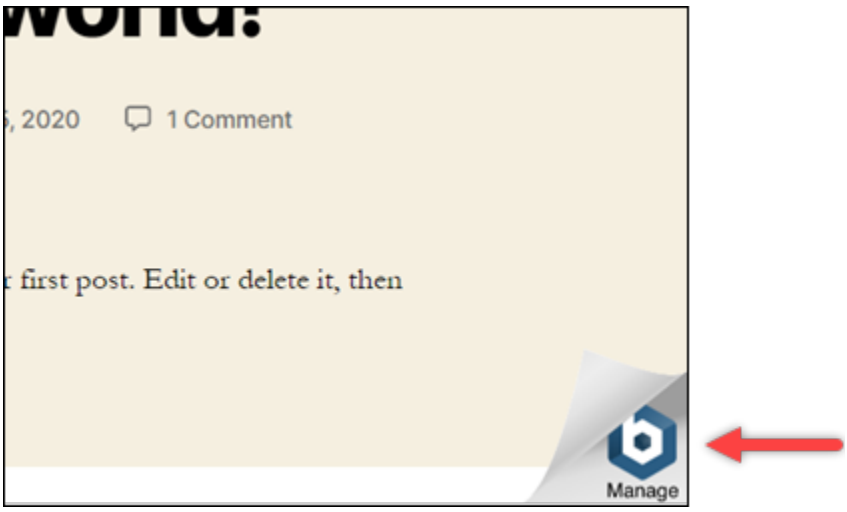
以下のリンクを使用して、Bitnami 設計図の詳細を確認し、チュートリアルを表示します。例えば、プラグインをインストールしたり、 WordPress インスタンスのSSL証明書によるHTTPSサポートを有効にしたりできます。

- Amazon Web Services WordPress 用 Bitnami
- Amazon Web Services の Bitnami LAMPスタック
- Bitnami Node.js for Amazon Web Services
- Bitnami Joomla for Amazon Web Services
- Bitnami Magento for Amazon Web Services
- Amazon Web Services の Bitnami MEANスタック
- Amazon Web Services Bitnami Drupal
- Amazon Web Services GitLab 用 Bitnami
- Bitnami Redmine for Amazon Web Services
- Amazon Web Services の Bitnami Nginx (LEMP スタック)

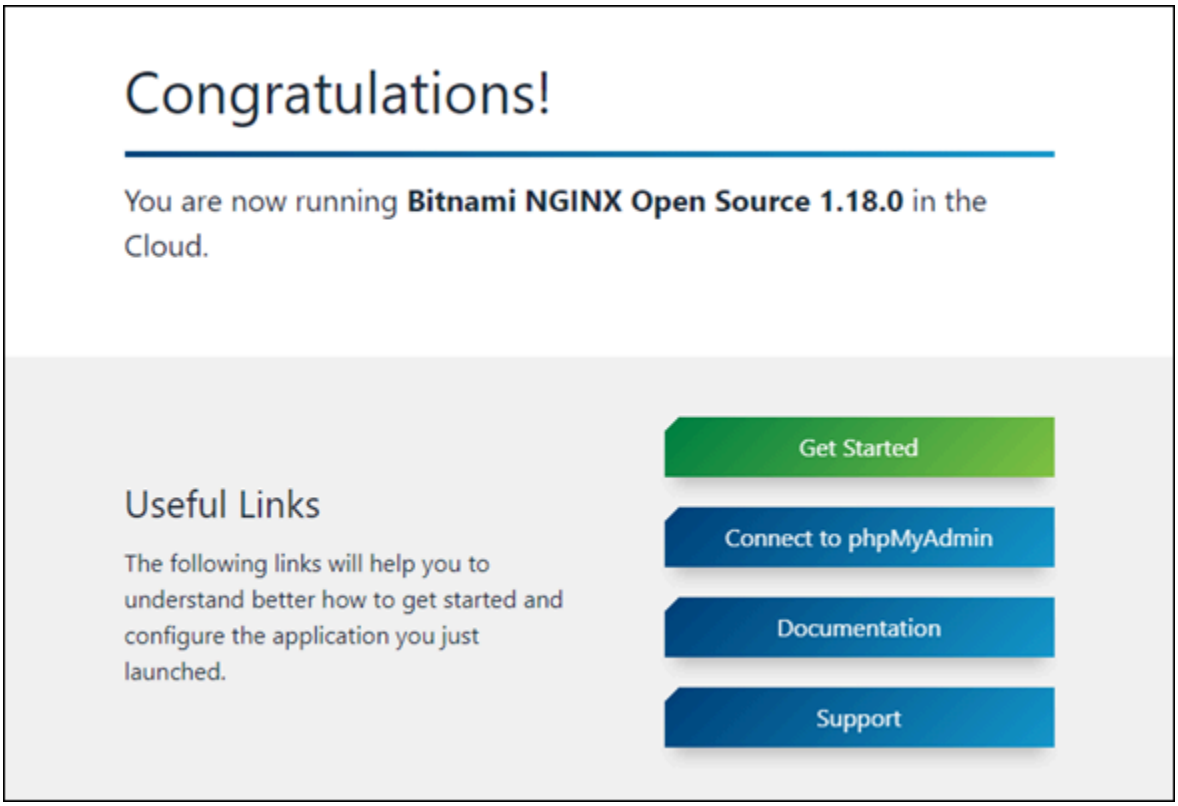
詳細については、Amazon Lightsail を使用する」または「Amazon Lightsail を使用する Amazon LightsailFAQ」を参照してください。

Lightsail 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できる Bitnami ブループリントの一部では、アプリケーションのホームページに Bitnami バナーが表示されます。次の「Certified by Bitnami WordPress 」インスタンスの例では、Bitnami バナーがホームページの右下隅に表示されます。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。



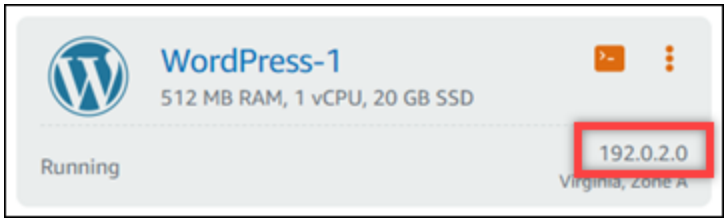
すべての Bitnami ブループリントアプリケーションが、アプリケーションのホームページに Bitnami バナーを表示するわけではありません。Lightsail インスタンスのホームページにアクセスして、Bitnami バナーが表示されるかどうかを確認します。「Packaged by Bitnami」Nginx インスタンスの次の例では、Bitnami アイコンは表示されていません。代わりに、プレースホルダー情報ページが表示されます。このページは、最終的にインスタンスにデプロイすることを選択したアプリケーションに置き換えられます。インスタンスに Bitnami バナーが表示されない場合は、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

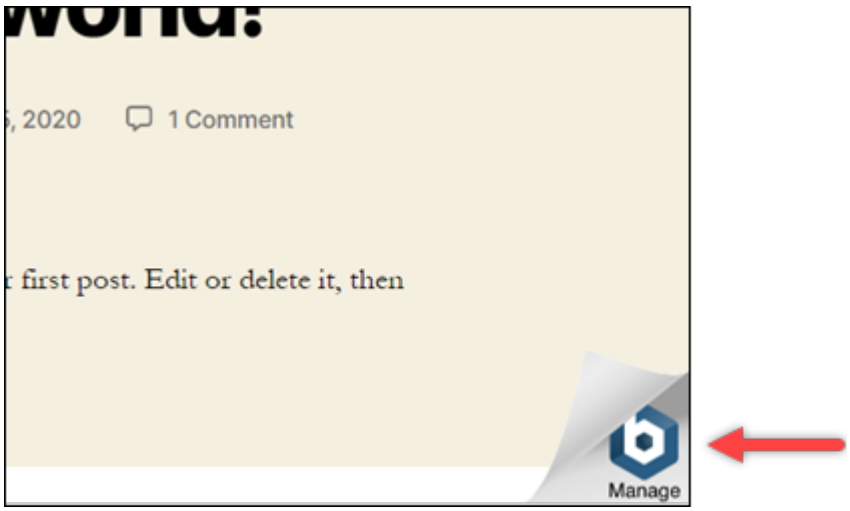
次の手順を実行して、インスタンスのアプリケーションのホームページに Bitnami アイコンが表示されていることを確認し、削除します。

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、確認する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:

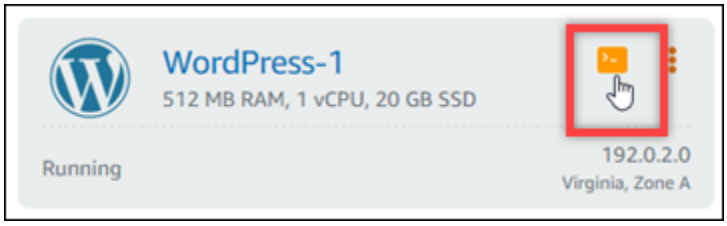
    1. Bitnami アイコンがページに表示されていない場合は、以下の手順に従う必要はありません。アプリケーションのホームページから Bitnami アイコンを削除する必要はありません。
    2. 次の例に示すように、Bitnami アイコンがページの右下隅に表示されている場合は、以下の一連のステップに従ってアイコンを削除します。



    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。

step-by-step 手順に従って、Bitnami アプリケーションとデータベースのデフォルトの認証情報を取得し、アプリケーションの管理パネルにサインインし、オプションで Bitnami ブランドバナーをアプリケーションのホームページから削除する方法について説明します。

このガイドでは、、Joomla、Drupal、Ghost WordPress、、、、MEANNode.js など、Lightsail で使用できるさまざまな Bitnami LAMP LEMPブループリントについて説明します。これは、アプリケーションとデータベースの両方のデフォルトのユーザー名と、デフォルトのパスワードを安全に取得するためのコマンドを提供します。このガイドに従うことで、Lightsail インスタンスで実行されている Bitnami アプリケーションに簡単にアクセスおよび管理し、要件に応じてカスタマイズし、不要なブランド要素を削除できます。

Lightsail WordPress インスタンスの設定と管理

このガイドでは、Lightsail の WordPress インスタンスに関連する以下のトピックについて説明します。

トピック

Lightsail で WordPress インスタンスを起動して設定する

Amazon Lightsail は、Amazon Web Services () の使用を開始する最も簡単な方法です AWS。Lightsail には、インスタンス (仮想プライベートサーバー) 、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 IPs、ロードバランサーなど、プロジェクトをすばやく起動するために必要なものがすべて含まれており、予測可能な低価格で提供されます。

このチュートリアルでは、Lightsail で WordPress インスタンスを起動して設定する方法について説明します。これには、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 を使用したインスタンスへの接続、 WordPress ウェブサイトへのサインインの手順が含まれます。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礎が整います。

> **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 の料金」ページのAWS 「無料利用枠」を参照してください。

内容

ステップ 1: にサインアップする AWS

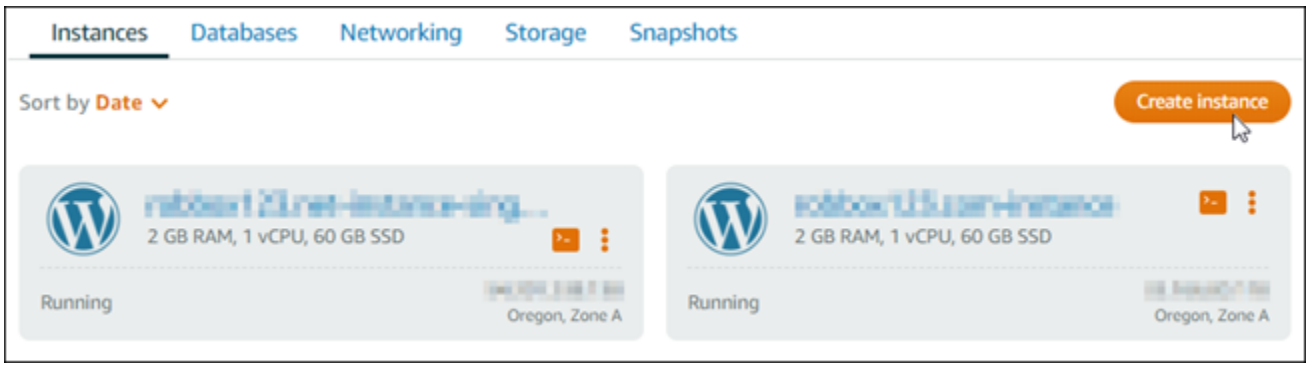
Amazon Lightsail には が必要です AWS アカウント。にサインアップ AWSするか、アカウントを既にお持ちの場合は にサインイン AWSします。

ステップ 2: WordPress インスタンスを作成する

WordPress インスタンスを起動して実行するには、次のステップを実行します。詳細については、「the section called “インスタンスを作成する”」を参照してください。

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. 次のようにインスタンスのイメージを選択します。

   a. プラットフォームの選択 で、Linux/Unix を選択します。

   b. 設計図の選択 で、 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のマシン設定 (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 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. [インスタンスの作成] を選択します。

8. テストブログ記事を表示するには、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の投稿が表示されます。

ステップ 3: インスタンスを設定する WordPress

ガイド付きの step-by-step ワークフローを使用して WordPress 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。いずれかのオプションを使用して、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。ユーザーは、WordPress サイトにアクセスするためにこのドメイン名を指定します。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 レコードを管理する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 “DNS Lightsail の”」を参照してください。
- 静的 IP アドレス – WordPress 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ます。静的 IP アドレスをインスタンスにアタッチしても、インスタンスを停止して起動しても同じままになります。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 ウェブサイトで HTTPS を有効にして、登録済みドメインを介してインスタンスにルーティングされるトラフィックを HTTPS を使用して暗号化できます。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。

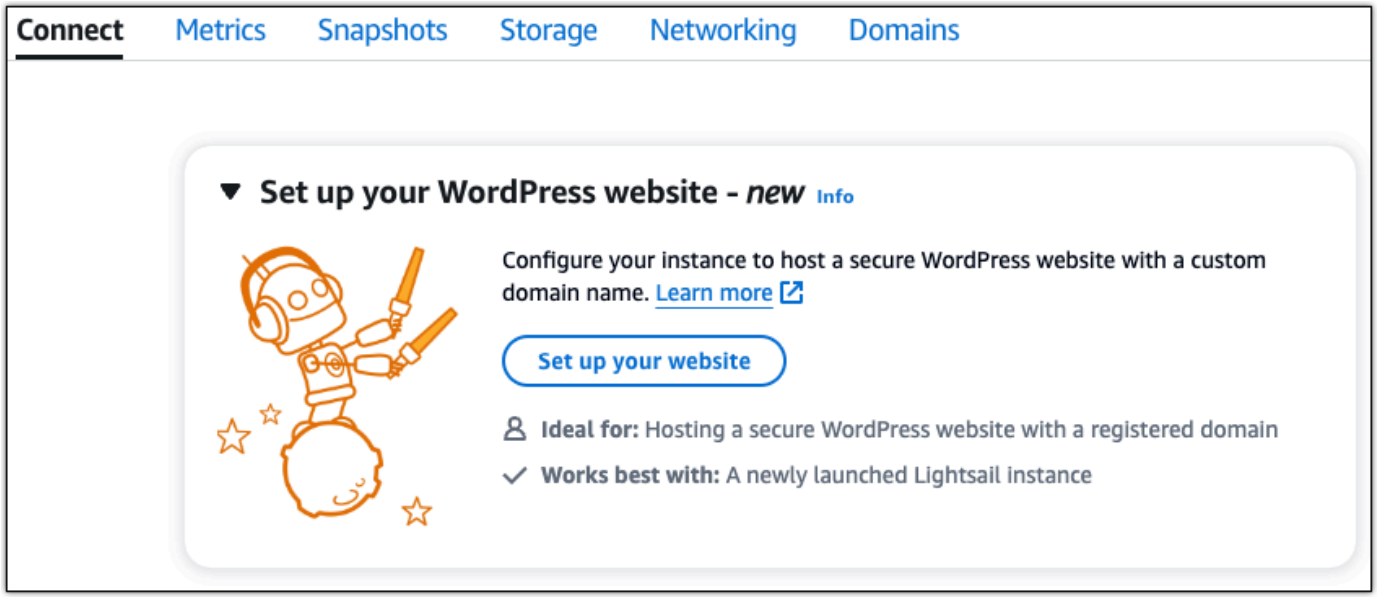
オプション: ガイド付きワークフロー

> **Tip**
>
> 開始する前に、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の詳細については、「セットアップのトラブルシューティング WordPress 」を参照してください。
>
> - セットアップは、 WordPress 2023 年 1 月 1 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。
> - セットアップ中に実行される Certbot 依存関係ファイル、HTTPS 書き換えスクリプト、および証明書更新スクリプトは、インスタンスの `/opt/bitnami/lightsail/scripts/` ディレクトリに保存されます。

- インスタンスは実行状態である必要があります。インスタンスが起動したばかりの場合、SSH 接続の準備が整うまで数分かかります。
- インスタンスファイアウォールのポート 22、80、および 443 では、セットアップの実行中に任意の IP アドレスからの TCP 接続を許可する必要があります。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`) とその`www`サブドメイン () からのトラフィックをポイントする DNS レコードを追加または更新するとき`www.example.com`は、インターネット全体に伝達する必要があります。nslookup や からの DNS Lookup などのツールを使用して、DNS の変更が有効になったことを確認できます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、ウェブサイトの設定が失敗する原因となる、非推奨の Certbot Personal Package Archive (PPA) リポジトリが含まれている場合があります。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、インスタンス上の次の場所にバックアップされます: `~/opt/bitnami/lightsail/repo.backup`。非推奨の PPA の詳細については、正規ウェブサイトの「Certbot PPA」を参照してください。
- Let's Encrypt 証明書は 60 ~ 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 分かかる場合があります。インスタンス接続タブで各ステップの進行状況を表示できます。

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、ウェブサイトのセットアップ を選択します。



2. ドメイン名 を指定するには、既存の Lightsail マネージド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ます。次のステップに進むには、この DNS ゾーンを使用する を選択します。
   - サードパーティードメインを選択して、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します。後で使用する場合に備えて、Lightsail アカウントに一致する DNS ゾーンを作成することに注意してください。サードパーティーの DNS を使用する を選択して、次のステップに進みます。
4. 「静的 IP アドレスの作成」で、静的 IP アドレスの名前を入力し、「静的 IP の作成」を選択します。
5. 「ドメイン割り当ての管理」で、「割り当ての追加」を選択し、ドメインタイプを選択し、「 の追加」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 証明書を作成する」で、ドメインとサブドメインを選択し、E メールアドレスを入力し、Lightsail にインスタンス で Let's Encrypt 証明書を設定することを承認し、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 分かかる場合があります。インスタンス接続タブで各ステップの進行状況を表示できます。

7. ウェブサイトの設定が完了したら、ドメイン割り当てステップで指定した URLsが WordPress サイトを開くことを確認します。

オプション: 個々のタスク

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選択されます。静的 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、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、トラフィックが中断されずに継続的に流れるようになります。

3. ドメイン割り当ての管理

   DNS ゾーンのページで、割り当て タブで、割り当ての追加 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、静的 IP アドレスをアタッチしてから、 の割り当てを選択します。

   > **Tip**
   >
   > ドメインが WordPress 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に時間を確保してください。

4. SSL/TLS 証明書を作成してインストールする

   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”。

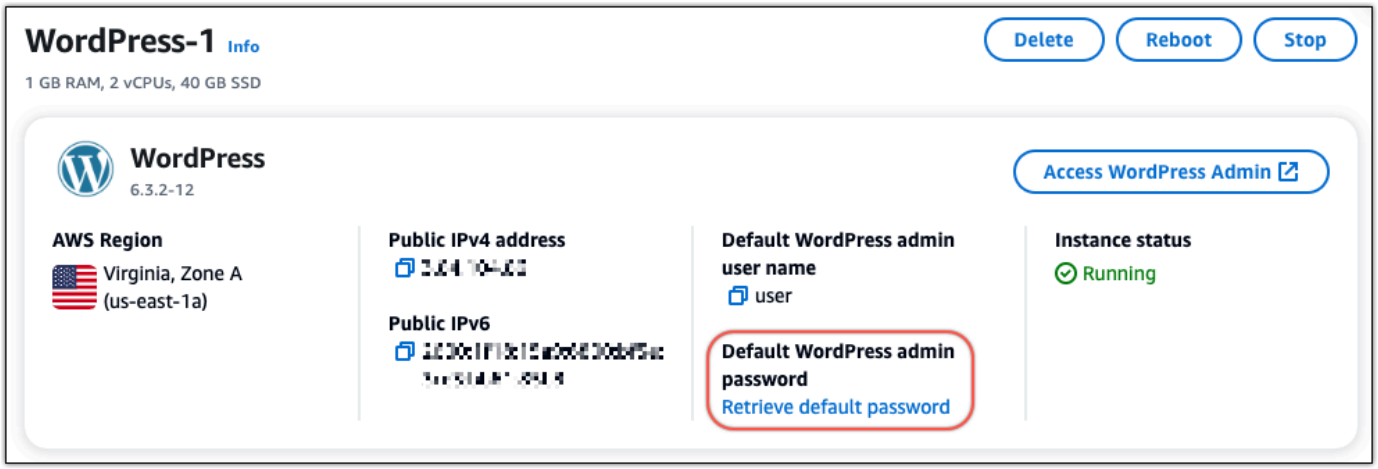
5. ドメイン割り当てステップで指定した URLs で WordPress サイトが開いていることを確認します。

ステップ 4: WordPress ウェブ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 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。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。

ステップ 5: 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

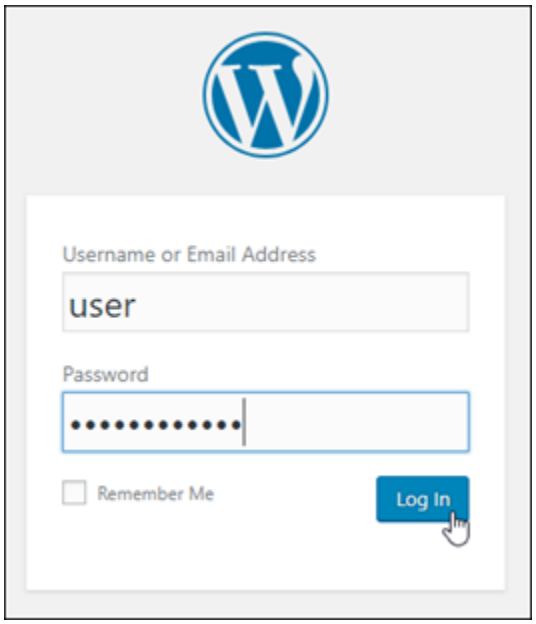
WordPress ウェブサイトの管理ダッシュボードのパスワードを取得したら、サインインできます。管理ダッシュボードでは、ユーザーパスワードの変更、プラグインのインストール、ウェブサイトのテーマの変更などを行うことができます。

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。

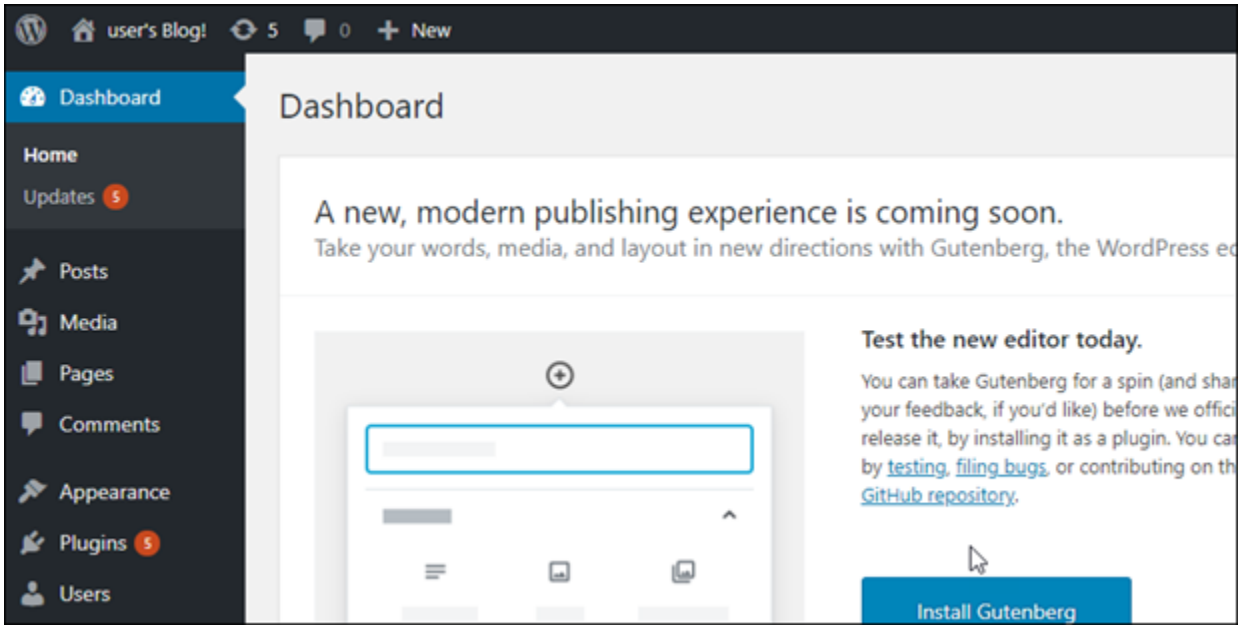
管理ダッシュボードにサインインするには

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 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。



追加情報

Amazon Lightsail で WordPress インスタンスを起動した後に実行できる追加の手順は次のとおりです。

- 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

WP Offload Media を使用して Lightsail WordPress のウェブサイトを Amazon S3 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) バケットに接続して、ウェブサイトのイメージと添付ファイルを保存するために必要な手順について説明します。これを行うには、一連の Amazon Web Services (AWS) アカウントの認証情報を使用して 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 で WordPress インスタンスを作成し、実行中の状態であることを確認します。詳細については、「チュートリアル: WordPress インスタンスの起動と設定」を参照してください。

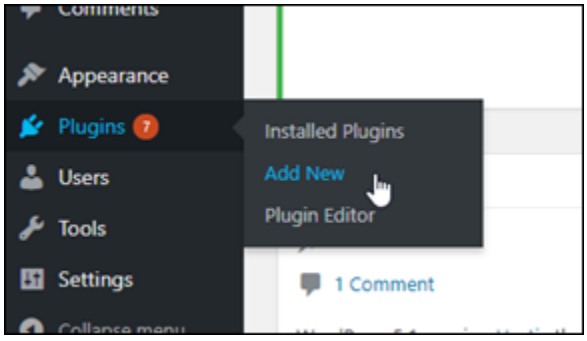
ステップ 2: WordPress ウェブサイトに WP Offload Media プラグインをインストールする

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。

ウェブサイトに WP Offload Media プラグインをインストールするには、次のステップを実行します WordPress。

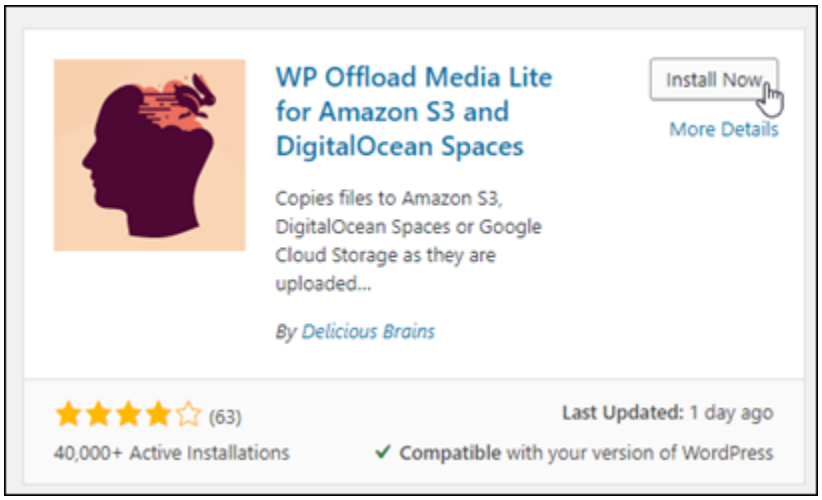
1. 管理者として WordPress ダッシュボードにサインインします。

   詳細については、Amazon Lightsail」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。

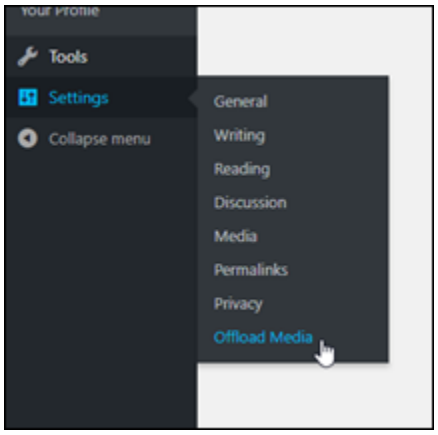


3. [WP Offload Media Lite] を検索します。

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`wp-config.php`の 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

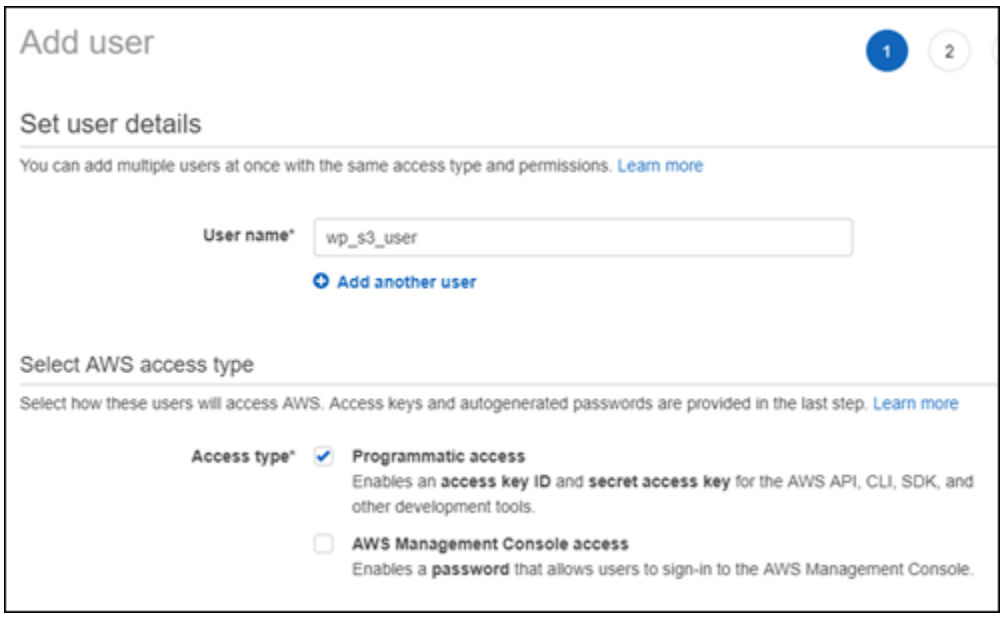
> **Warning**
>
> このシナリオでは、プログラムによるアクセスと長期的な認証情報を持つIAMユーザーが必要です。これはセキュリティ上のリスクをもたらします。このリスクを軽減するために、これらのユーザーにはタスクの実行に必要な権限のみを付与し、不要になったユーザーを削除することをお勧めします。アクセスキーは、必要に応じて更新できます。詳細については、「 IAMユーザーガイド」の「アクセスキーの更新」を参照してください。

WP Offload Media プラグインでは、Amazon S3 バケットを作成し、ウェブサイトのイメージと添付ファイルをアップロードするために、 AWS アカウントにアクセスする必要があります。

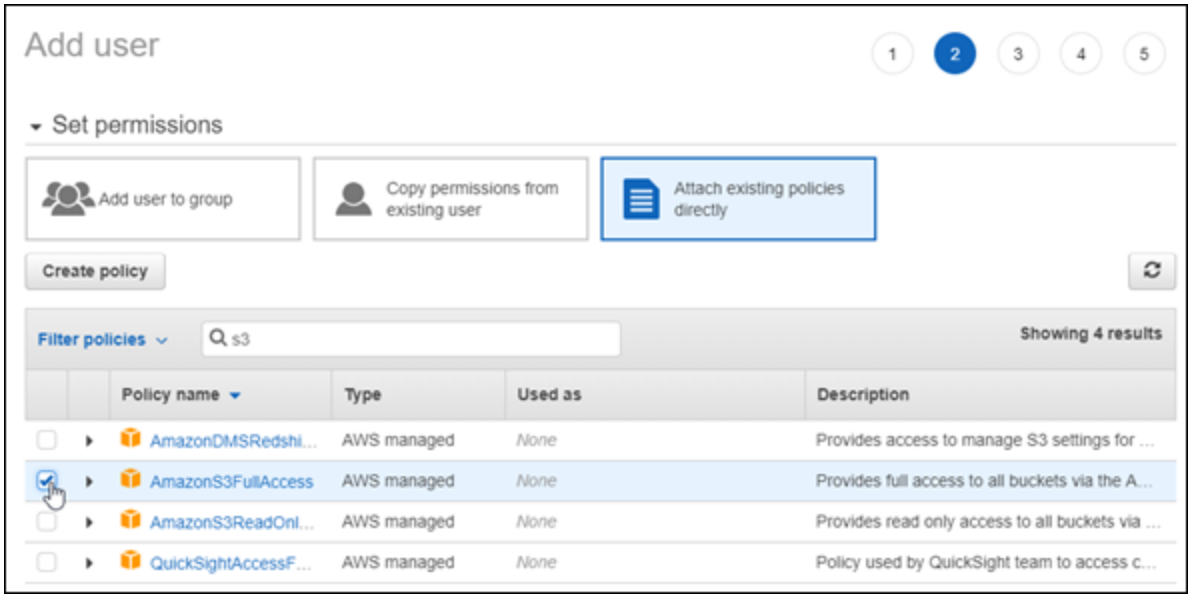
WP Offload Media プラグインの新しい AWS Identity and Access Management （ IAM) ユーザーとポリシーを作成するには、次のステップを実行します。

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: アクセス許可) を選択します。
7. 既存のポリシーを直接アタッチ を選択し、S3 を検索してから、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。これらは、 WordPress インスタンスでwp-config.phpファイルを編集する次のステップで必要になります。

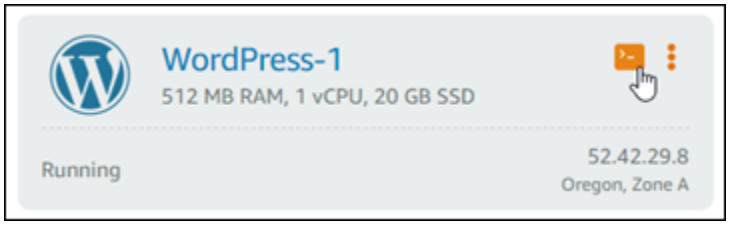
ステップ 4: WordPress設定ファイルを編集する

Lightsail コンソールでブラウザベースの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 ファイルを編集するには、次の手順を実行します。

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 SSHで 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 Amazon Lightsail」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。

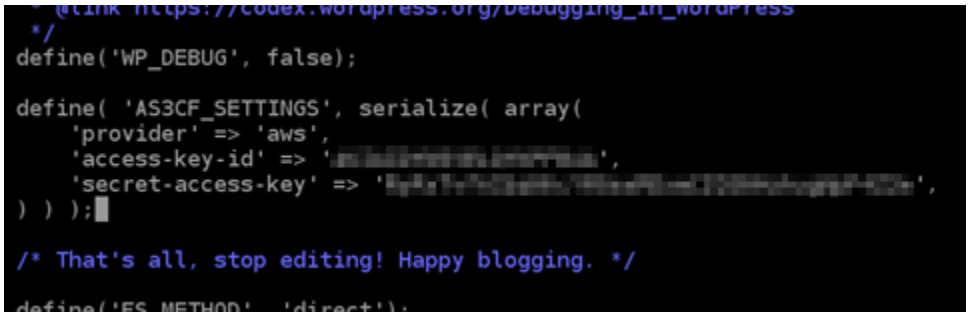
   必ず を置き換えてください。*AccessKeyID* アクセスキー ID と *SecretAccessKey* を、これらのステップで先ほど作成したIAMユーザーのシークレットアクセスキーで使用します。

   ```
   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。

```
bitnami@ip-172-26-13-236:~$ sudo /opt/bitnami/ctlscript.sh restart
Syntax OK
/opt/bitnami/apache2/scripts/ctl.sh : httpd stopped
/opt/bitnami/php/scripts/ctl.sh : php-fpm stopped
/opt/bitnami/mysql/scripts/ctl.sh : mysql stopped
/opt/bitnami/mysql/scripts/ctl.sh : mysql  started at port 3306
/opt/bitnami/php/scripts/ctl.sh : php-fpm started
Syntax OK
/opt/bitnami/apache2/scripts/ctl.sh : httpd started at port 80
bitnami@ip-172-26-13-236:~$
```

SSH ウィンドウを閉じて、このチュートリアルの前半で開いたオフロードメディアページに戻ります。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

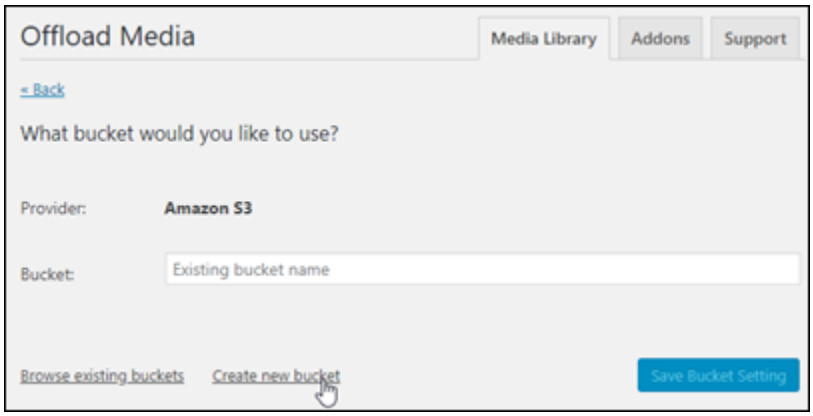
`wp-config.php` ファイルがAWS認証情報で設定されたので、オフロードメディアページに戻ってプロセスを完了できます。

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。

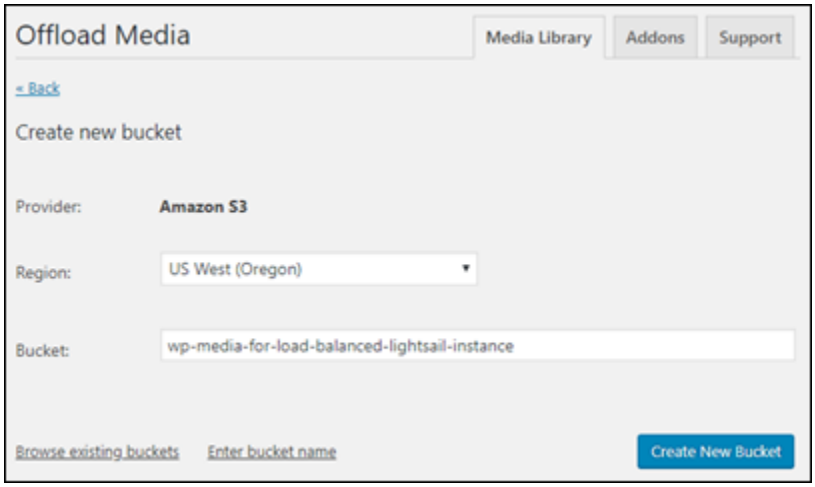
1. [Offload Media] ページを更新するか、[Nesxt] を選択します。

   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。

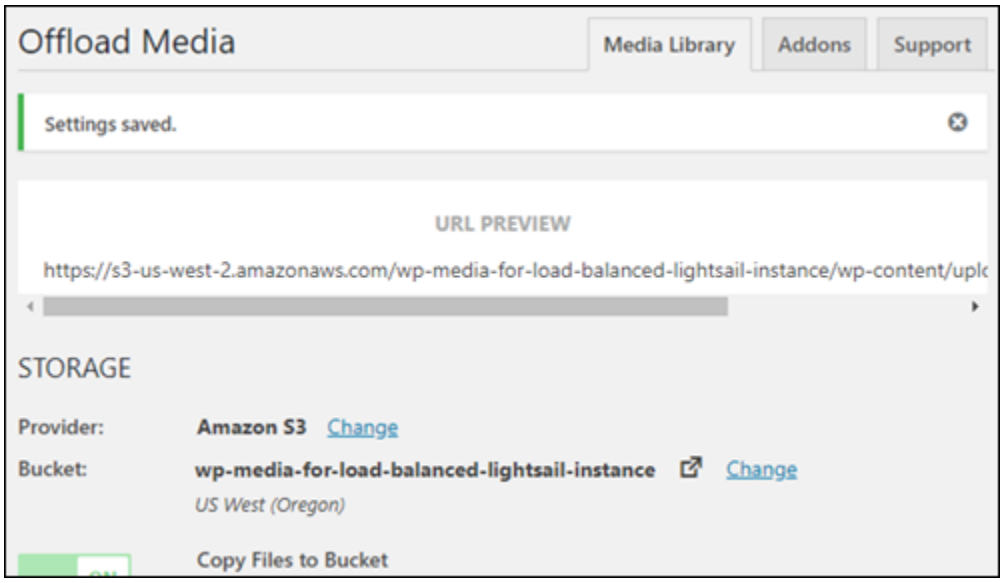


3. リージョンドロップダウンメニューで、目的のAWSリージョンを選択します。 WordPress インスタンスがあるリージョンと同じリージョンを選択することをお勧めします。
4. [バケット] テキストボックスに、新しい S3 バケットの名前を入力します。



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応じて調整します。



   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。

Lightsail 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、ユーザーのウェブサイトデータは、Amazon Lightsail の WordPress インスタンスで実行されているデータベースに保存されます。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 の WordPress インスタンスから Amazon RDS の Aurora マネージドデータベースにウェブサイトデータベースを接続する方法を示します。

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、そのインスタンスで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 Amazon Lightsail」を参照してください。

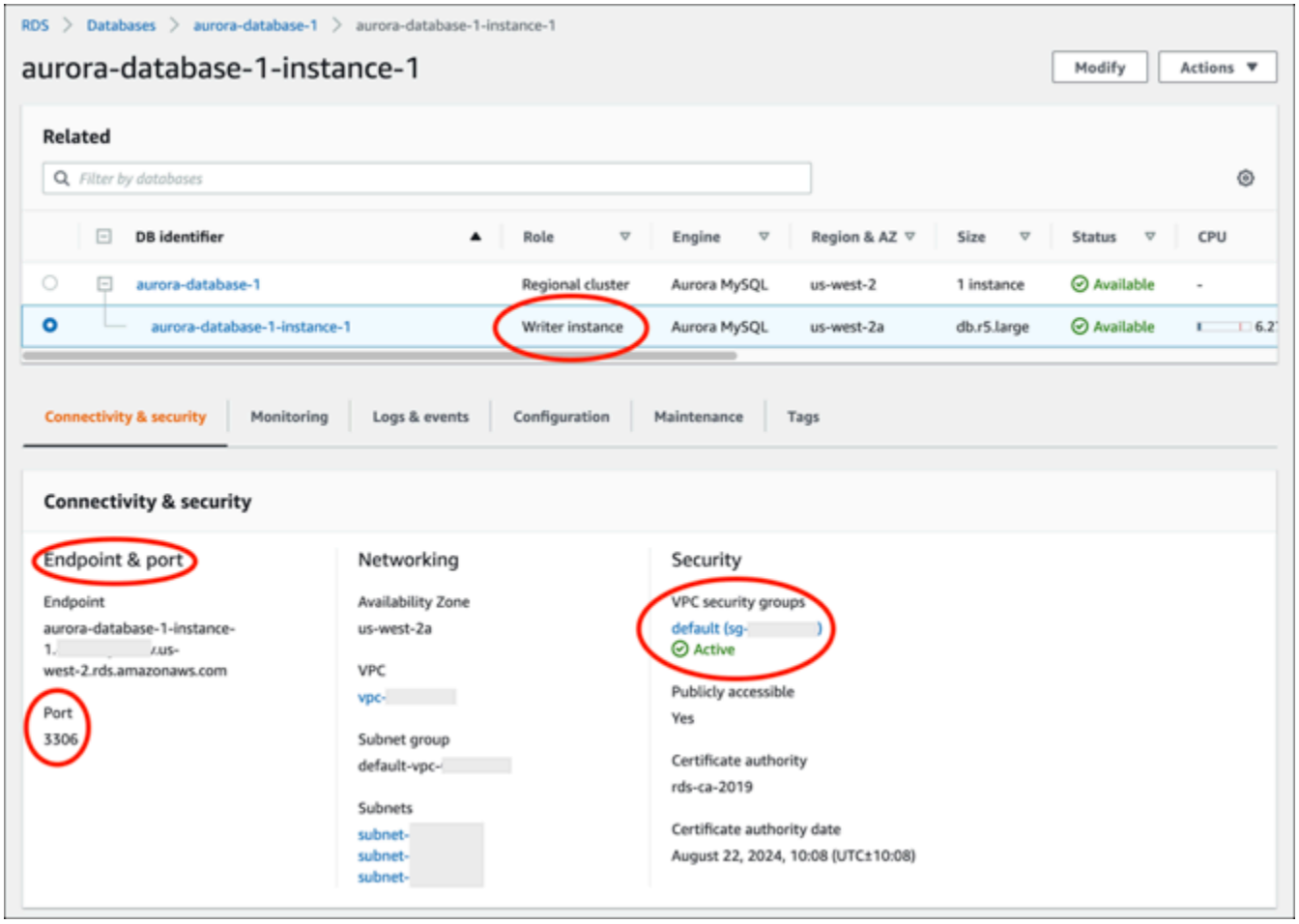
2. Lightsail アカウントで VPC ピアリングを有効にします。詳細については、「Lightsail の外部の AWS リソースを操作するようにピアリングを設定する」を参照してください。
3. Amazon RDS に Aurora マネージドデータベースを作成します。データベースは、 WordPress インスタンス AWS リージョン と同じ に配置する必要があります。続行する前に、データベースが実行中状態になっていることを確認してください。詳細については、「Amazon Aurora ユーザーガイド」の「Amazon Aurora で使用開始」を参照してください。

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

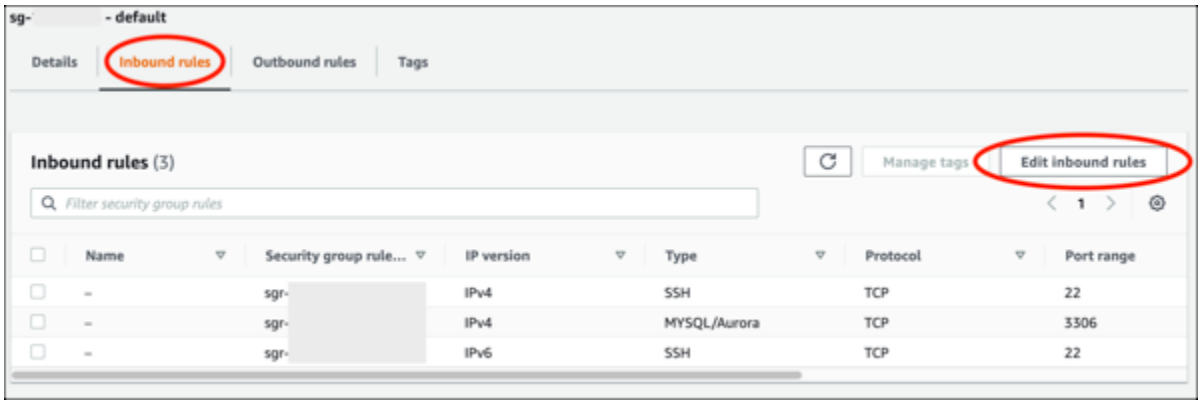
AWS セキュリティグループは、 AWS リソースの仮想ファイアウォールとして機能します。Amazon RDS 内の Aurora データベースに接続できる送受信トラフィックを制御します。詳細については、「Amazon Virtual Private Cloud ユーザーガイド」の「セキュリティグループを使用してリソースへのトラフィックを制御する」を参照してください。

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に WordPressセキュリティグループを設定するには、以下の手順を実行します。

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. インスタンスが接続する Aurora データベースのライター WordPress 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。これらは、後でデータベースに接続するように 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。



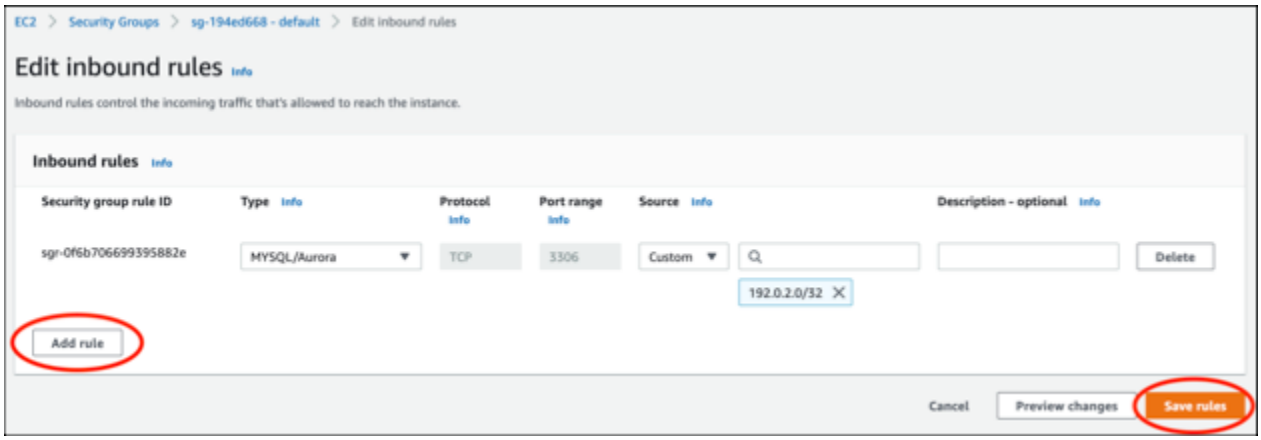
7. Aurora データベースのセキュリティグループが選択されていることを確認します。
8. [Inbound rules] (インバウンドルール) タブを開きます。
9. [Edit inbound rules] (インバウンドルールの編集) を選択します。



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。

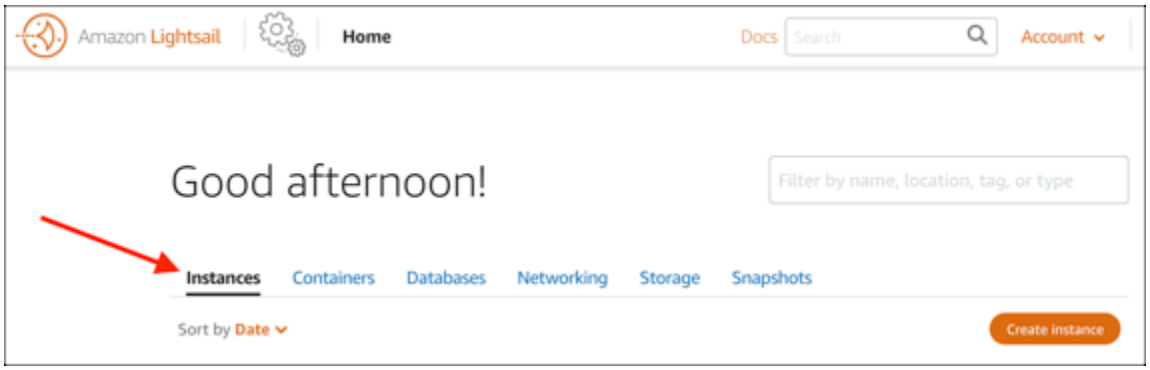
12. ソーステキストボックスに、 WordPress インスタンスのプライベート IP アドレスを追加します。IP アドレスは、 CIDR 表記で入力する必要があります (/32 を追加する必要があります)。例えば、192.0.2.0 を許可するには「192.0.2.0/32」と入力します。
13. [Save Rules] (ルールの保存) を選択します。



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

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。

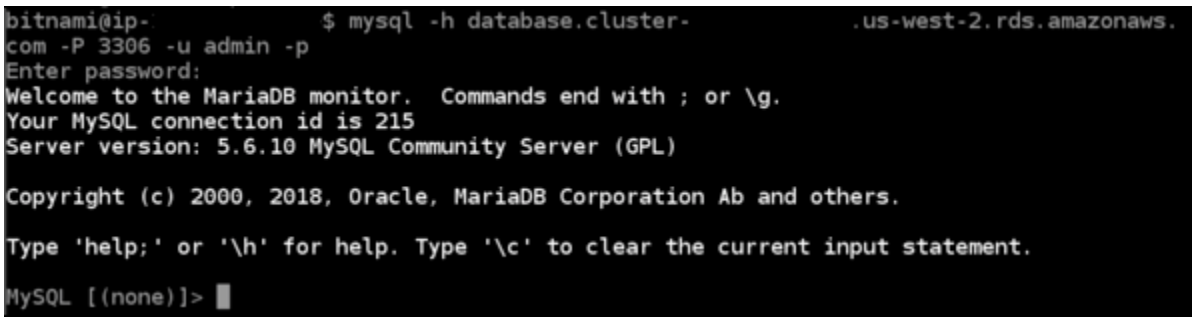


3. SSH を使用して接続する WordPress インスタンスのブラウザベースの SSH クライアントアイコンを選択します。

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。コマンドで、 を Aurora データベースのエンドポイントアドレス*DatabaseEndpoint*に置き換え、*Port* をデータベースのポートに置き換えます。を、データベースの作成時に入力したユーザーの名前*MyUserName*に置き換えます。

```
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。



このレスポンス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る場合は、Lightsail インスタンスのプライベート IP アドレスが接続できるように Aurora データベースのセキュリティグループを設定する必要があります。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: WordPress インスタンスから Aurora データベースにデータベースを転送する

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ので、 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. インスタンス タブで、 WordPress インスタンスのブラウザベースの SSH クライアントを選択します。

3. ブラウザベースの SSH クライアントを WordPress インスタンスに接続したら、次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。コマンドで、 を Aurora データベースの作成時に入力したプライマリユーザーの名前*DatabaseUserName*に置き換えます。を Aurora データベースのエンドポイントアドレス*DatabaseEndpoint*に置き換えます。

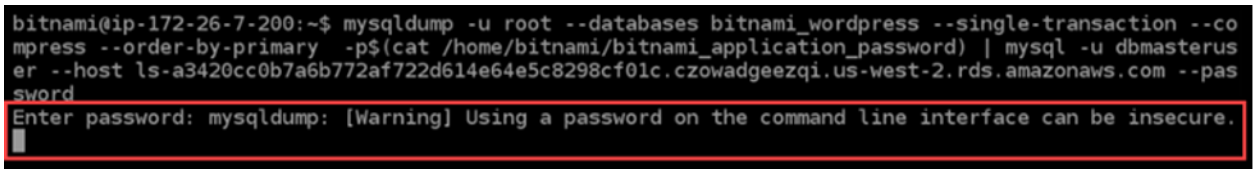
```
sudo mysqldump -u root --databases bitnami_wordpress --single-transaction --
compress --order-by-primary -p$(cat /home/bitnami/bitnami_application_password) |
 sudo mysql -u DatabaseUserName --host  DatabaseEndpoint --password
```

例

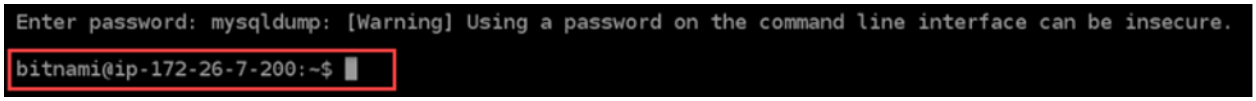
```
sudo mysqldump -u root --databases bitnami_wordpress --single-transaction --
compress --order-by-primary -p$(cat /home/bitnami/bitnami_application_password)
 | sudo mysql -u DBuser --host  abc123exampleE67890.czowadgeezqi.us-
west-2.rds.amazonaws.com --password
```

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 WordPress に接続するように 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ら、接続 WordPress するように を設定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設定ファイル (wp-config.php) を編集する WordPressには、以下の手順を実行します。

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、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。

   ```
   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
   ```

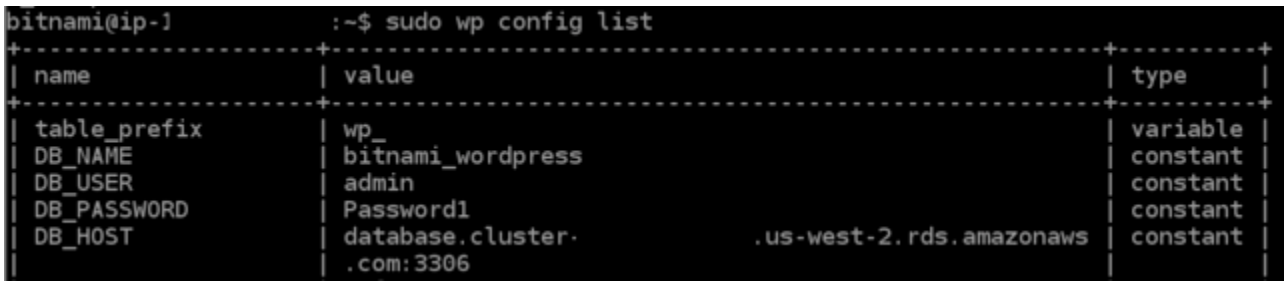
5. config ファイル内のデータベースパスワードを Aurora データベースのパスワードで編集します。

   ```
   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す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
> ```

Lightsail で MySQL マネージドデータベースに WordPress データを転送する

投稿、ページ、ユーザーの 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を示します。また、ウェブサイトがマネージドデータベースに接続し、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ように、インスタンス WordPress の設定 (`wp-config.php`) ファイルを編集する方法も示します。

目次

ステップ 1: 前提条件を満たす

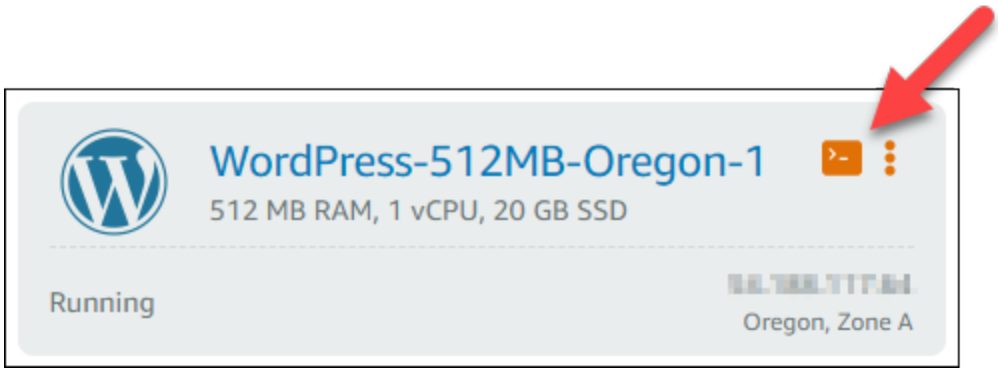
開始する前に、前提条件として以下の作業を実行します。

- Lightsail で WordPress インスタンスを作成し、実行中の状態であ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 Amazon Lightsail」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 マネージドデータベースを作成し、 WordPress Lightsail で使用可能なすべての MySQL データベースオプションを使用して、そのデータベースが実行中の状態になっていることを確認します。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。
- 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: WordPress データベースを MySQL マネージドデータベースに転送する

Lightsail で WordPress ウェブサイトデータを MySQL マネージドデータベースに転送するには、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. インスタンス タブで、 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. 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したら、次のコマンドを入力して、インスタンス上のbitnami_wordpressデータベース内のデータを MySQL マネージドデータベースに転送します。をマネージドデータベースのユーザー名*DbUserName*に置き換え、 をマネージドデータベースのエンドポイントアドレス*DbEndpoint*に置き換えてください。

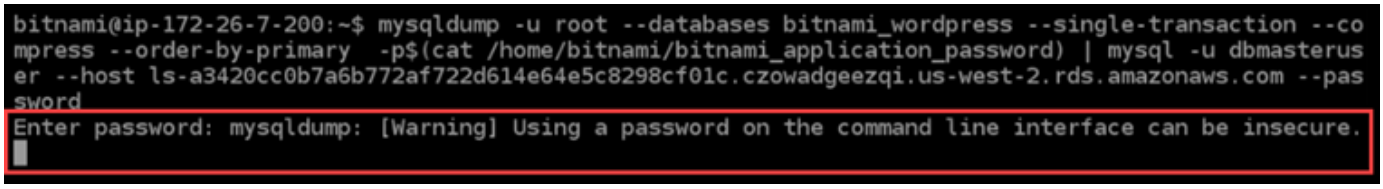
```
sudo mysqldump -u root --databases bitnami_wordpress --single-transaction --
compress --order-by-primary  -p$(cat /home/bitnami/bitnami_application_password) |
 sudo mysql -u DbUserName --host DbEndpoint --password
```

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
```

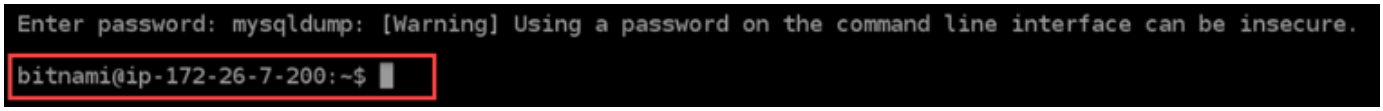
4. プロンプトで、MySQL マネージド型データベースのパスワードを入力し、Enter を押します。

   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 マネージドデータベース WordPress に接続するように を設定する

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 WordPress 設定ファイル (wp-config.php) を編集するには、以下の手順を実行します。

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、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。

   ```
   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
   ```

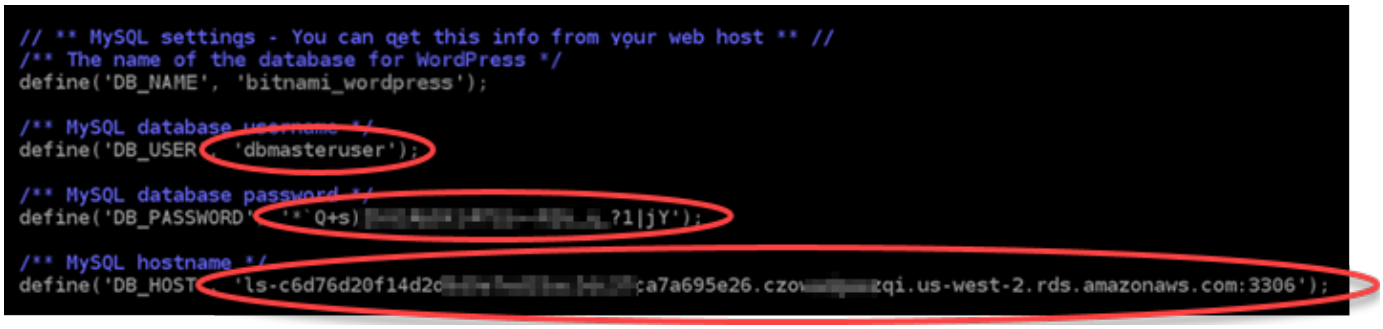
3. 以下の例のようにDB_USER、DB_PASSWORD、およびDB_HOST 値が見つかるまで下にスクロールします。

```
// ** 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のプライマリユーザー名は ですdbmasteruser。
   - 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 。

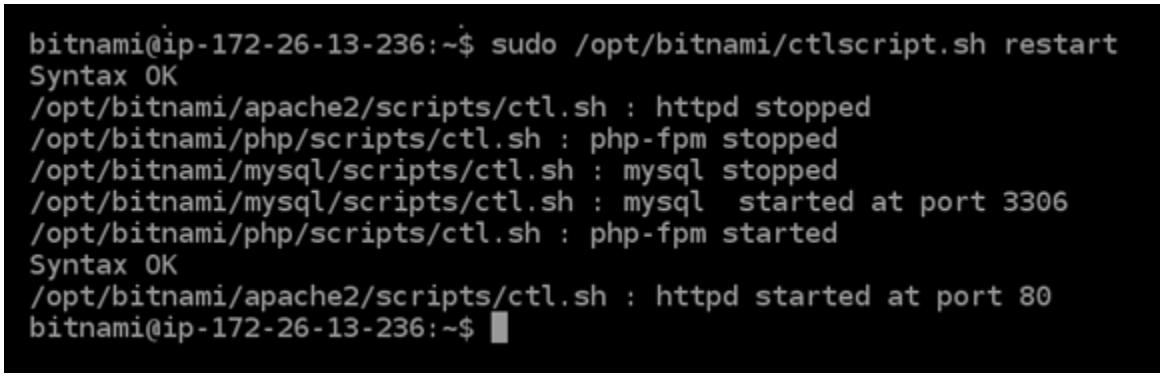
   結果は次の例のようになります。



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、MySQL マネージドデータベースを使用するように WordPress サイトが設定されました。

> **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

ウェブサイトを WordPress MySQL マネージドデータベースに接続したら、これらの追加手順を完了する必要があります。

- 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。

静的コンテンツ用に WordPress インスタンスを Lightsail 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 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、WordPress 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。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)。

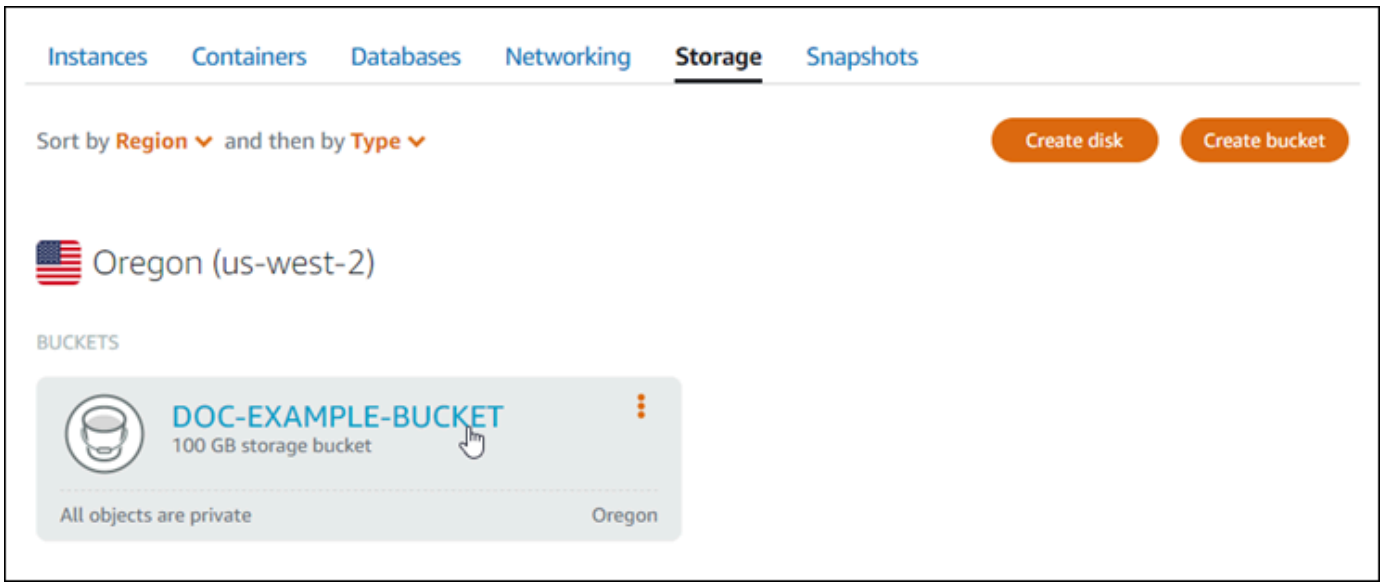
- Lightsail で WordPress インスタンスを作成します。詳細については、「チュートリアル: Amazon Lightsail で WordPress インスタンスを起動して設定する Amazon Lightsail」を参照してください。
- Lightsail オブジェクトストレージサービスでバケットを作成します。詳細については、「バケットの作成」を参照してください。

ステップ 2: バケットのアクセス許可を変更する

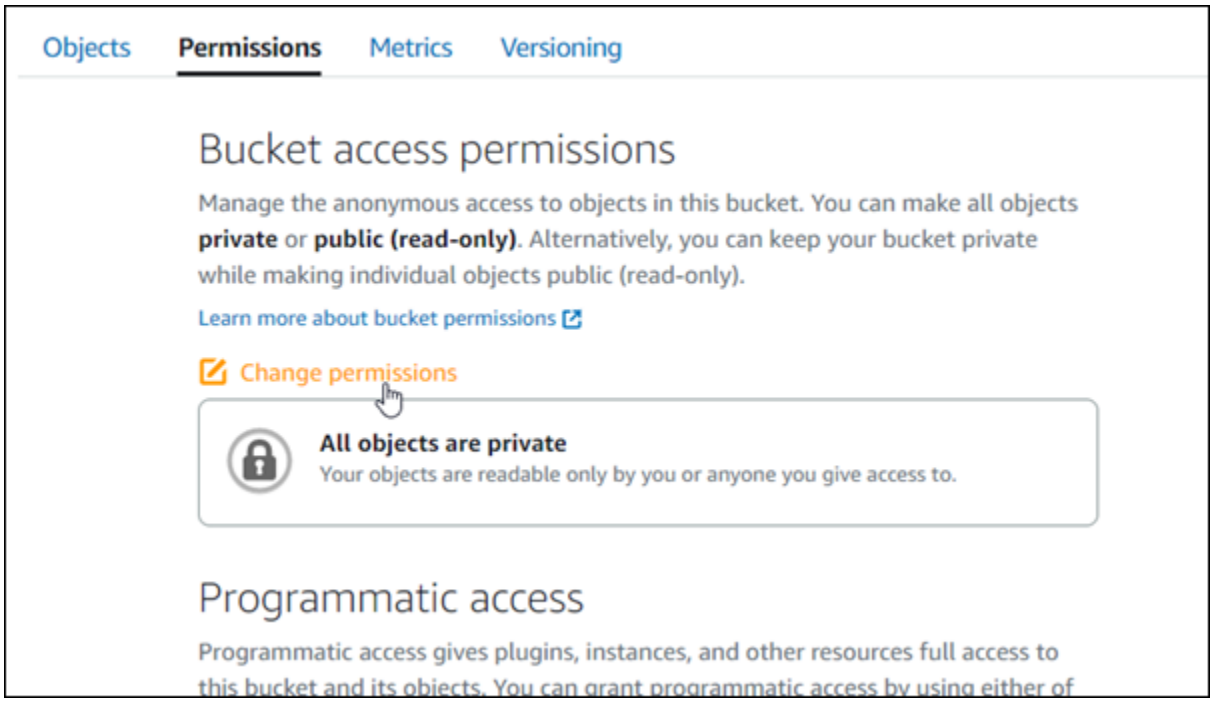
バケットのアクセス許可を変更して、 WordPress インスタンスと Offload Media Lite プラグインへのアクセスを許可するには、次の手順を実行します。バケットのアクセス許可は個々のオブジェクトを公開 (読み取り専用)に設定する必要があります。また、バケットのアクセスロールに WordPress インスタンスをアタッチする必要があります。バケット許可の詳細については、「バケットのアクセス許可」を参照してください。

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。

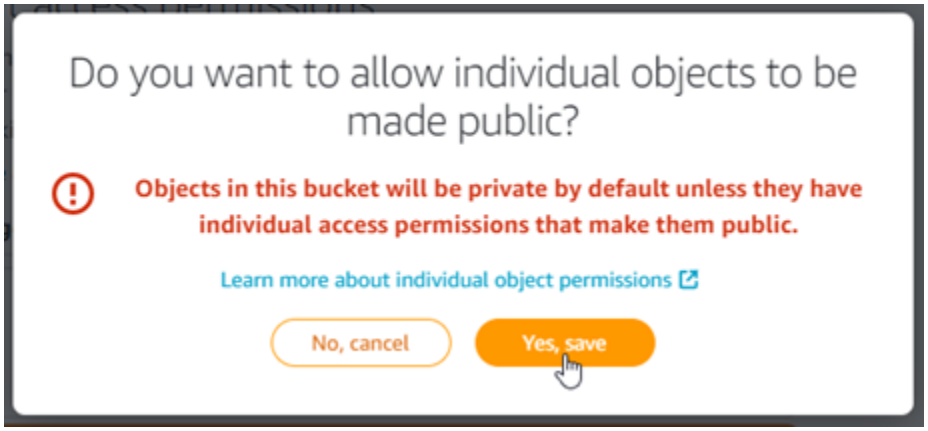


4. バケット管理ページで [Permissions] (許可) タブを選択します。
5. ページの「バケットのアクセス許可」セクションで [Change permissions](許可の変更) を選択します。



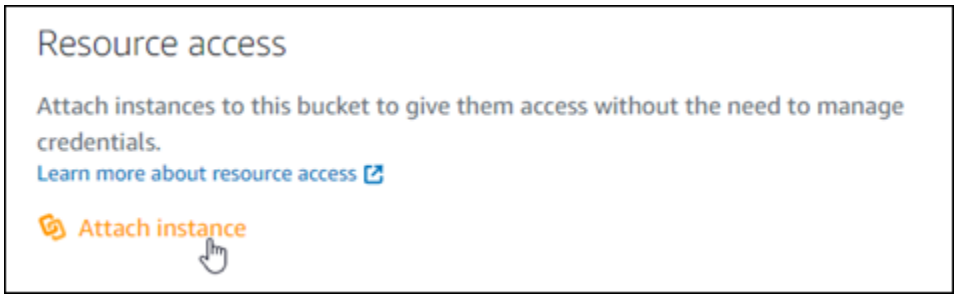
6. 個々のオブジェクトを選択して公開し、読み取り専用にすることができます。

7. [Save] 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。

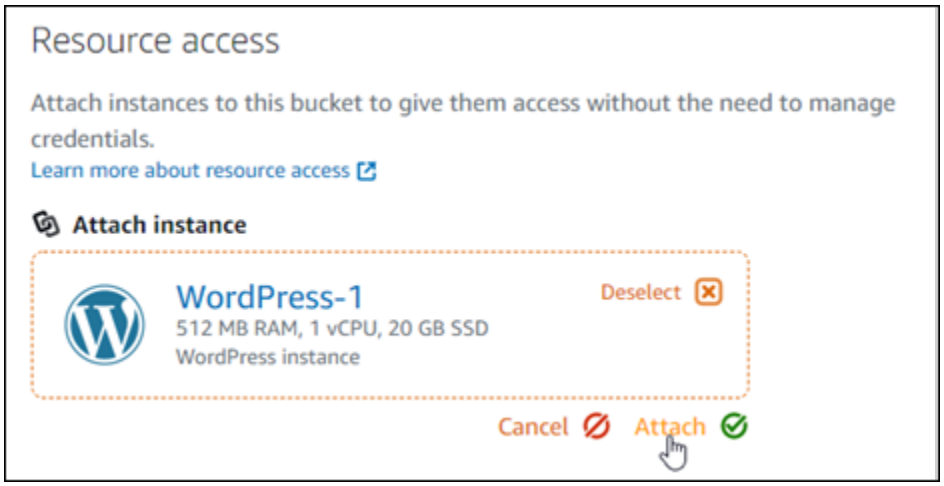


しばらくすると、バケットは個々のオブジェクトにアクセスを許可するように設定されます。これにより、 Offload Media Lite プラグインを使用して WordPress ウェブサイトからバケットにアップロードされたオブジェクトを顧客が読み取ることができます。

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、 をアタッチ 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。

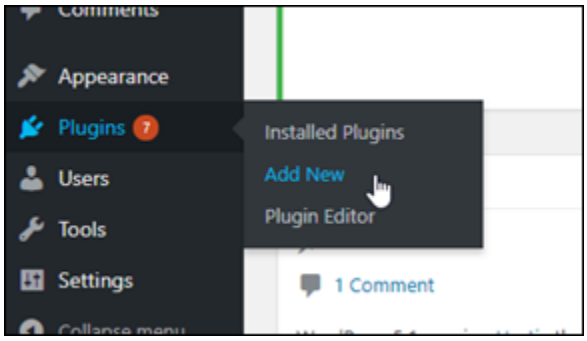
ステップ 3: WordPress ウェブサイトに WP Offload Media Lite プラグインをインストールする

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、次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。

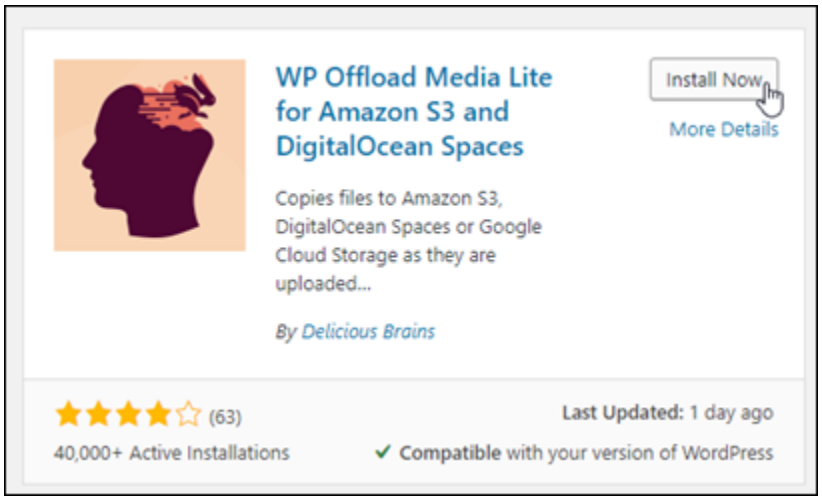
1. 管理者として WordPress ウェブサイトのダッシュボードにサインインします。

   詳細については、Amazon Lightsail」を参照してください。

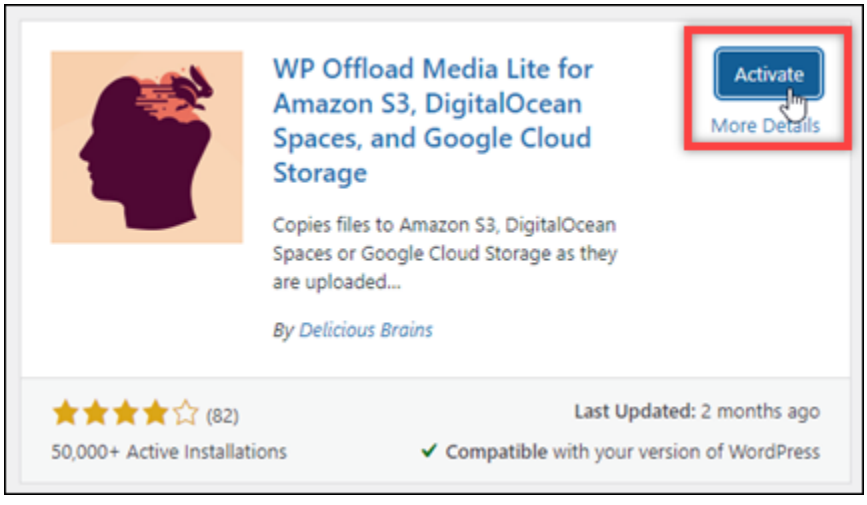
2. 左側のナビゲーションメニューの [プラグイン] を一時停止し、[Add New] (新規追加) を選択します。



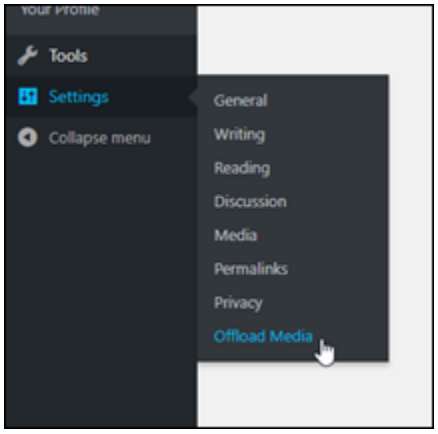
3. [WP Offload Media Lite] を検索します。
4. 検索結果の中から WP Offload Media プラグインの横の [Install Now] (今すぐインストール)を選択します。



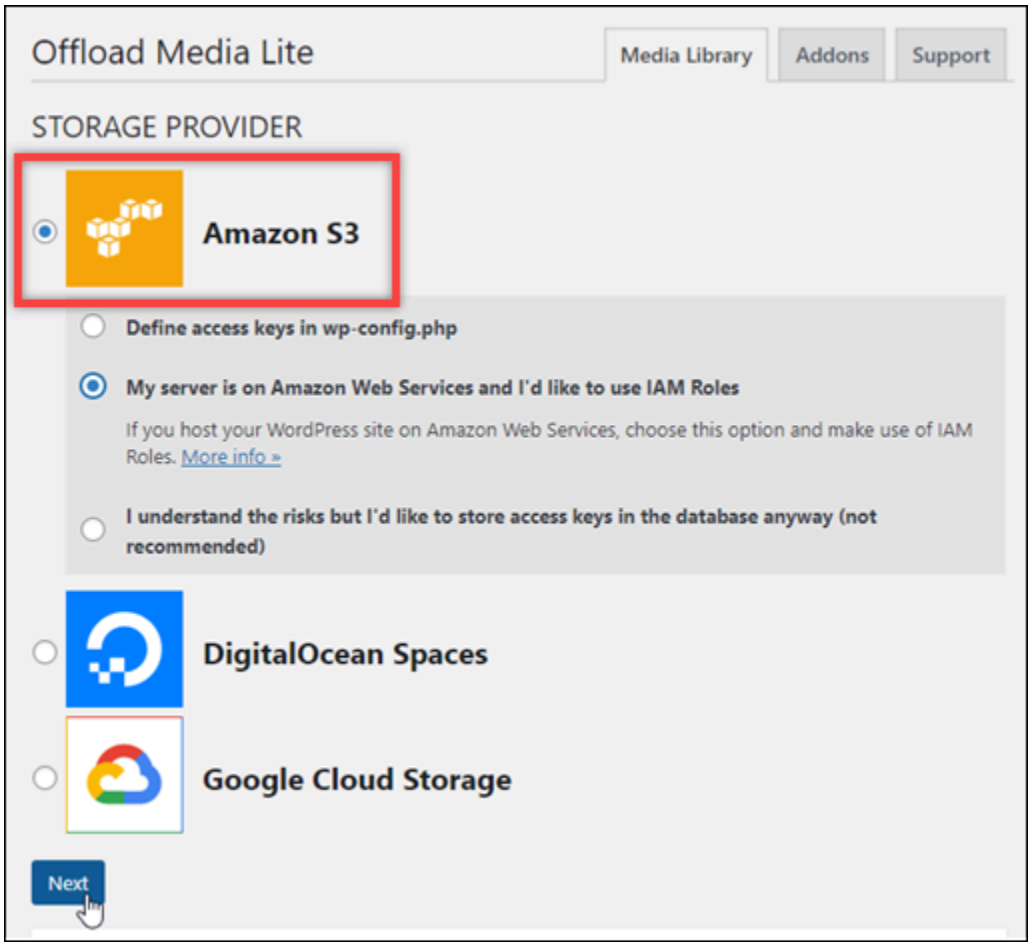
5. プラグインのインストールが完了したら、[アクティベート] を選択します。



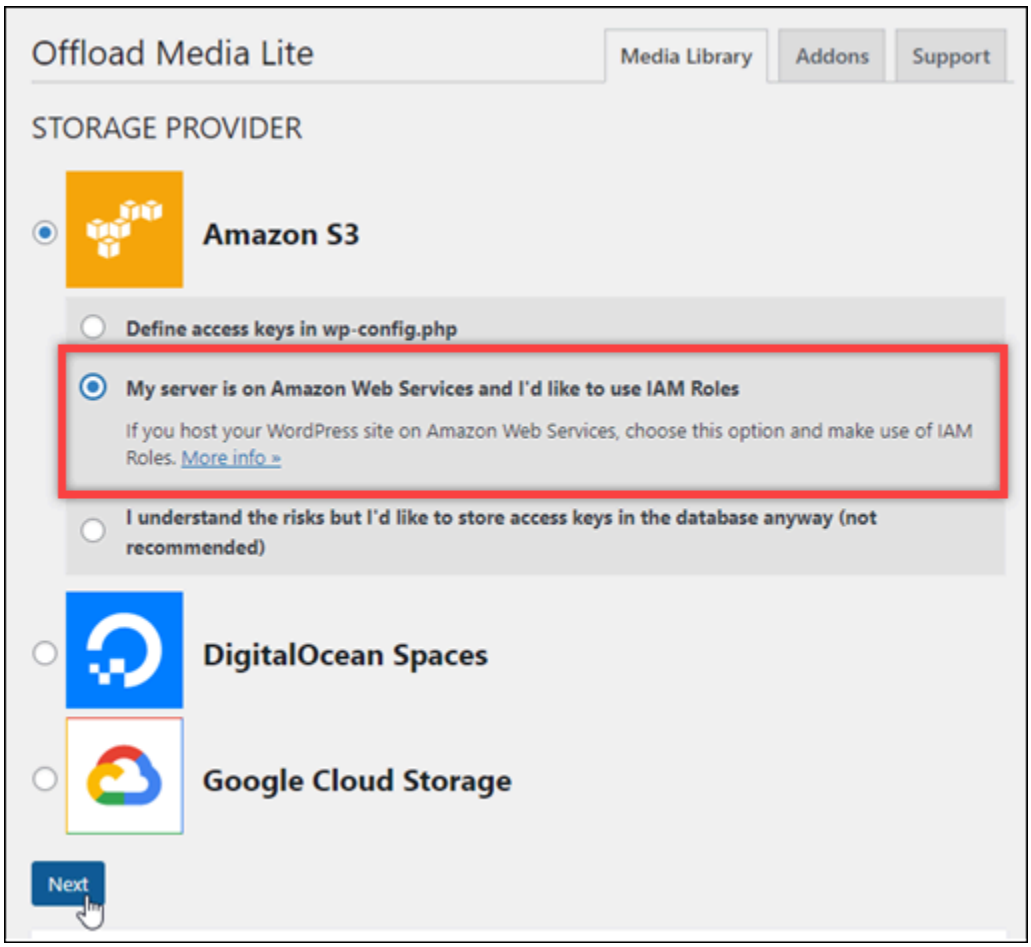
6. 左のナビゲーションメニューで、[Settings] (設定) を選択し、[Offload Media] を選択します。



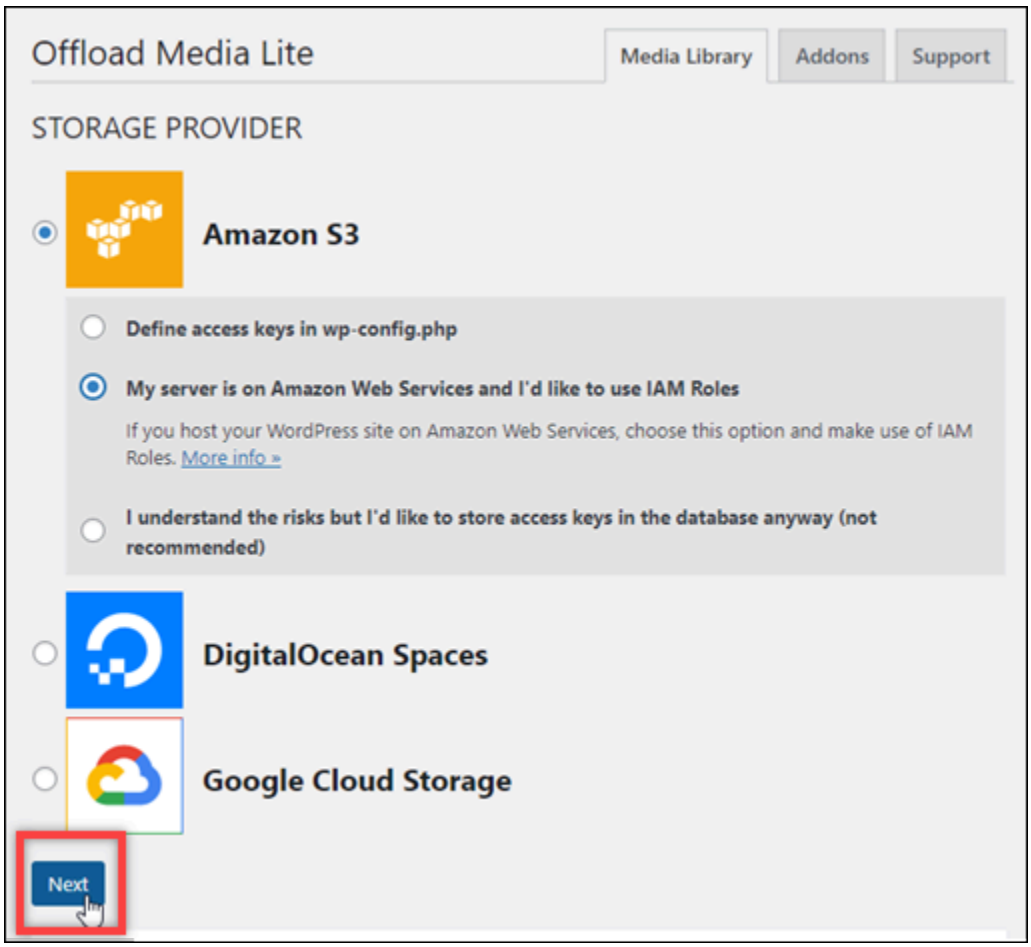
7. Offload Media ページで [Amazon S3] をストレージプロバイダとして選択します。



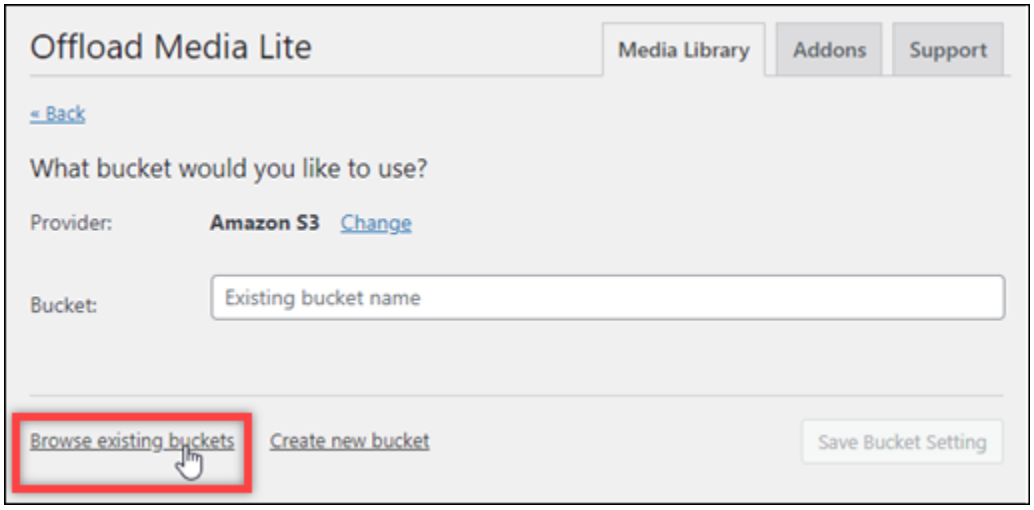
8. 「サーバーが Amazon Web Services にあり、IAMロール を使用する」を選択します。



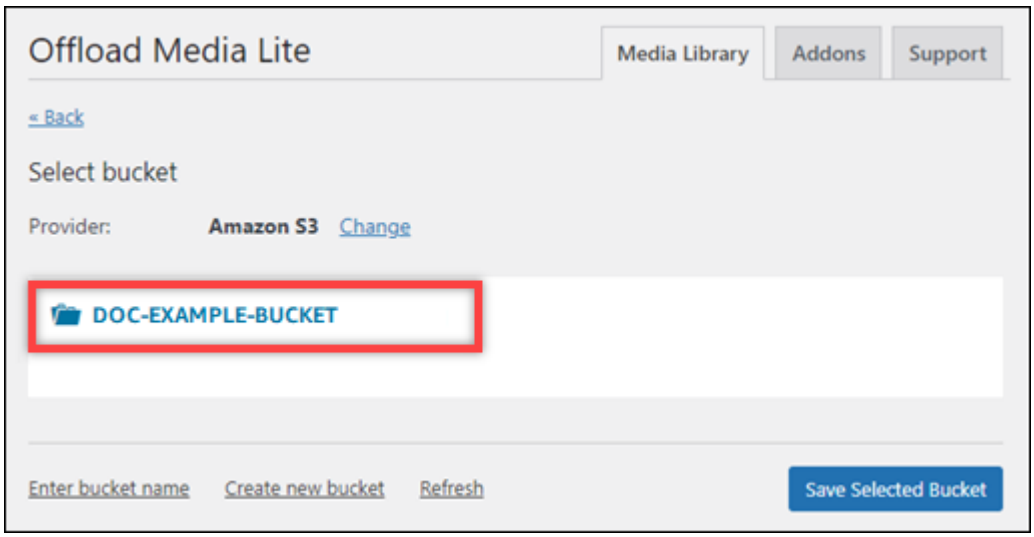
9. [Next (次へ)] を選択します。



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。

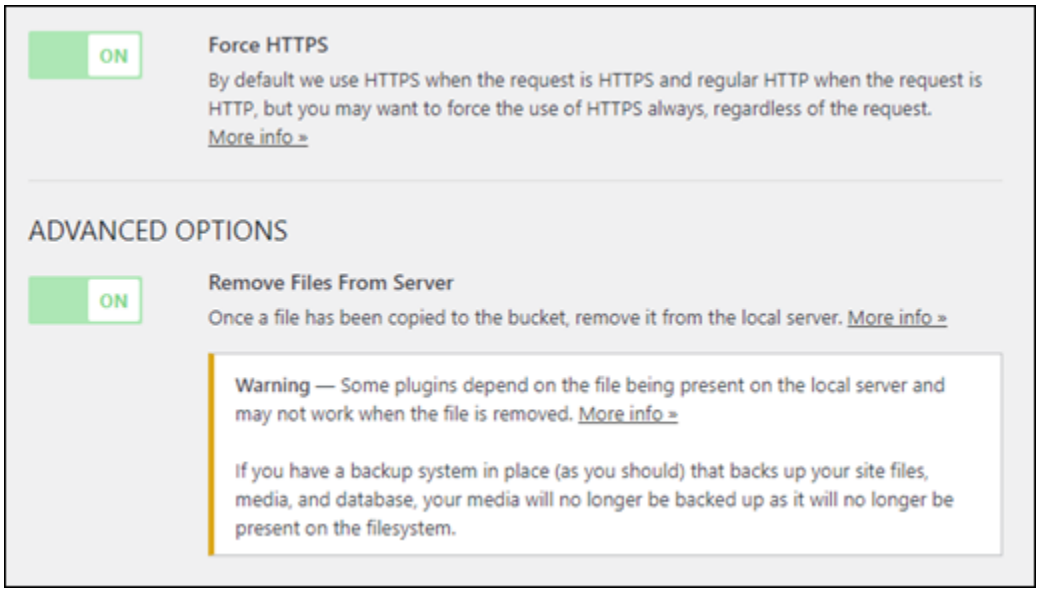


11. インスタンスで使用する WordPressバケットの名前を選択します。



12. 表示される「メディアライト設定のオフロード」ページで、サーバー からファイルを強制 HTTPSおよび削除してください。

    - Lightsail バケットはHTTPSデフォルトで を使用してメディアファイルを配信するため、強制HTTPS設定を有効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。この機能を有効にしない場合、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. [Save Changes] を選択します。

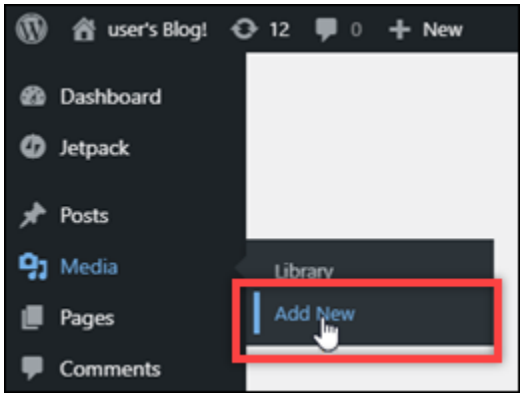
> **Note**
>
> 後でOffload Media Lite Settings画面に戻るには、左のナビゲーションメニューで [設定] を一時停止し、[Offload Media Lite] を選択します。

これで WordPress 、ウェブサイトが Media Lite プラグインを使用するように設定されました。次に 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 Lightsail バケットに自動的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。

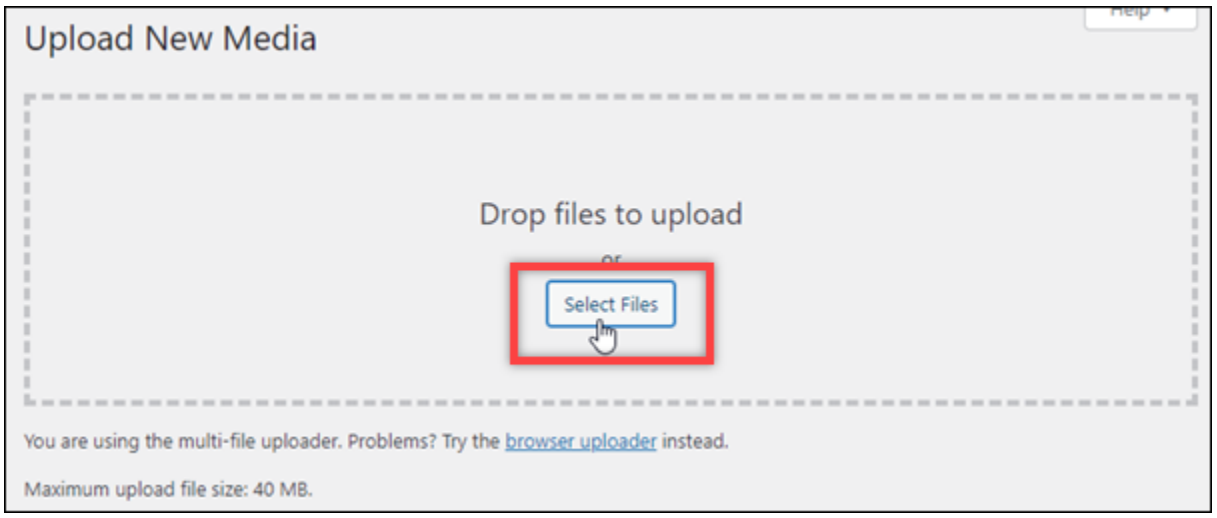
ステップ 4: WordPress ウェブサイトと Lightsail バケット間の接続をテストする

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、Lightsail バケットから提供されることを確認します。

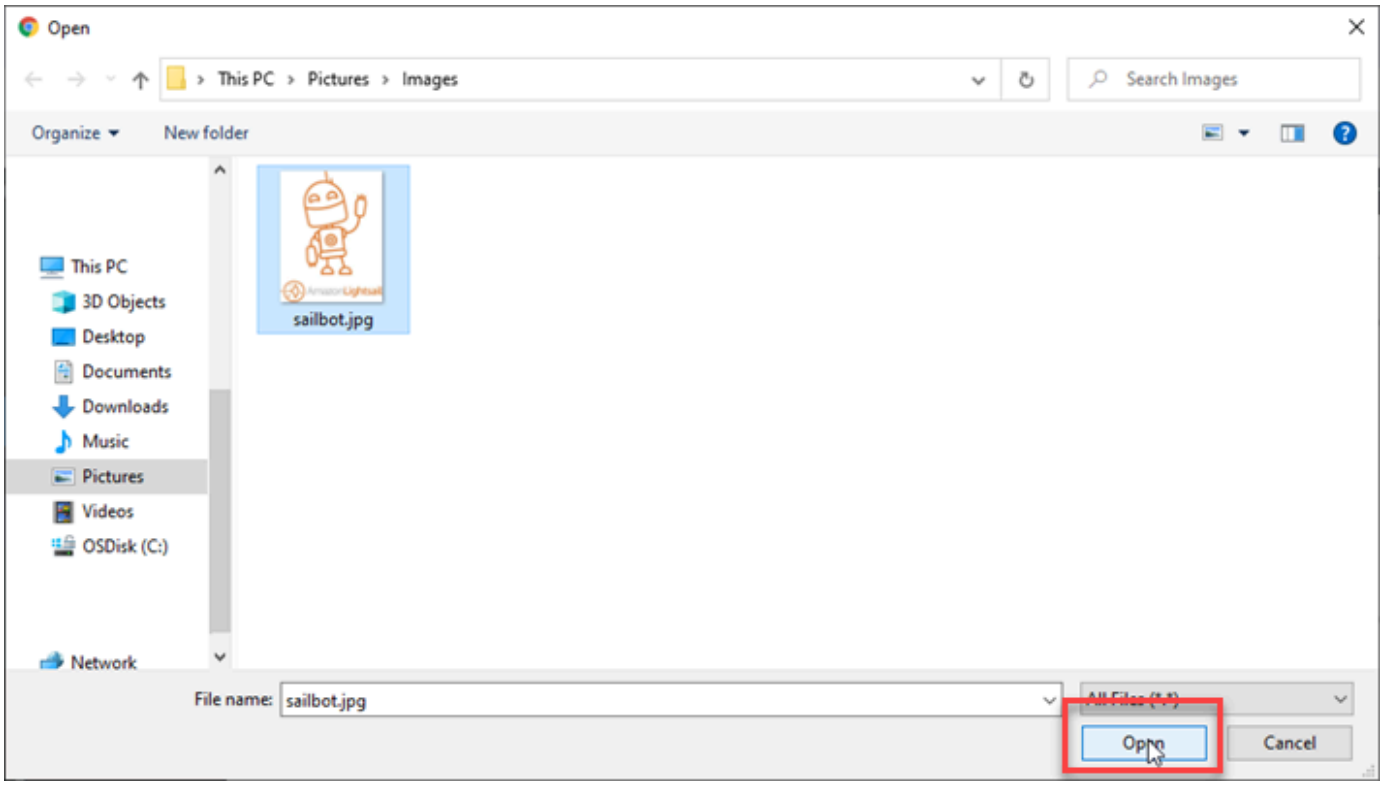
1. ダッシュボードの左側のナビゲーションメニューでメディアで WordPress一時停止し、新しいを追加を選択します。



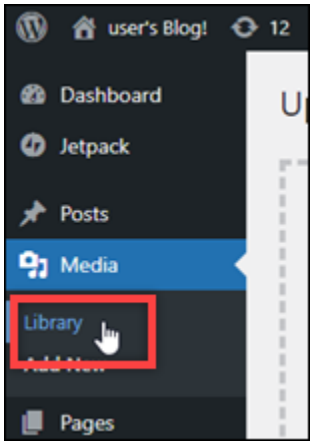
2. 表示される、[新しいメディアをアップロード] 画面で [ファイルを選択] を選択します。



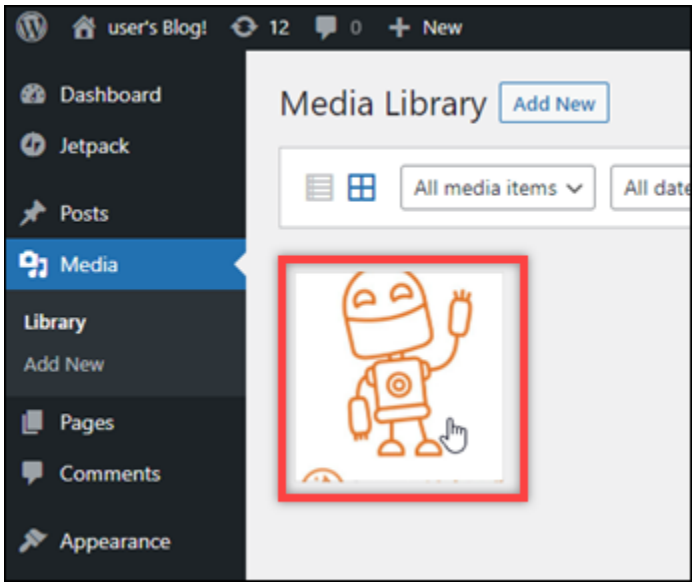
3. ローカルコンピュータからアップロードするメディアファイルを選択し、[開く] を選択します。



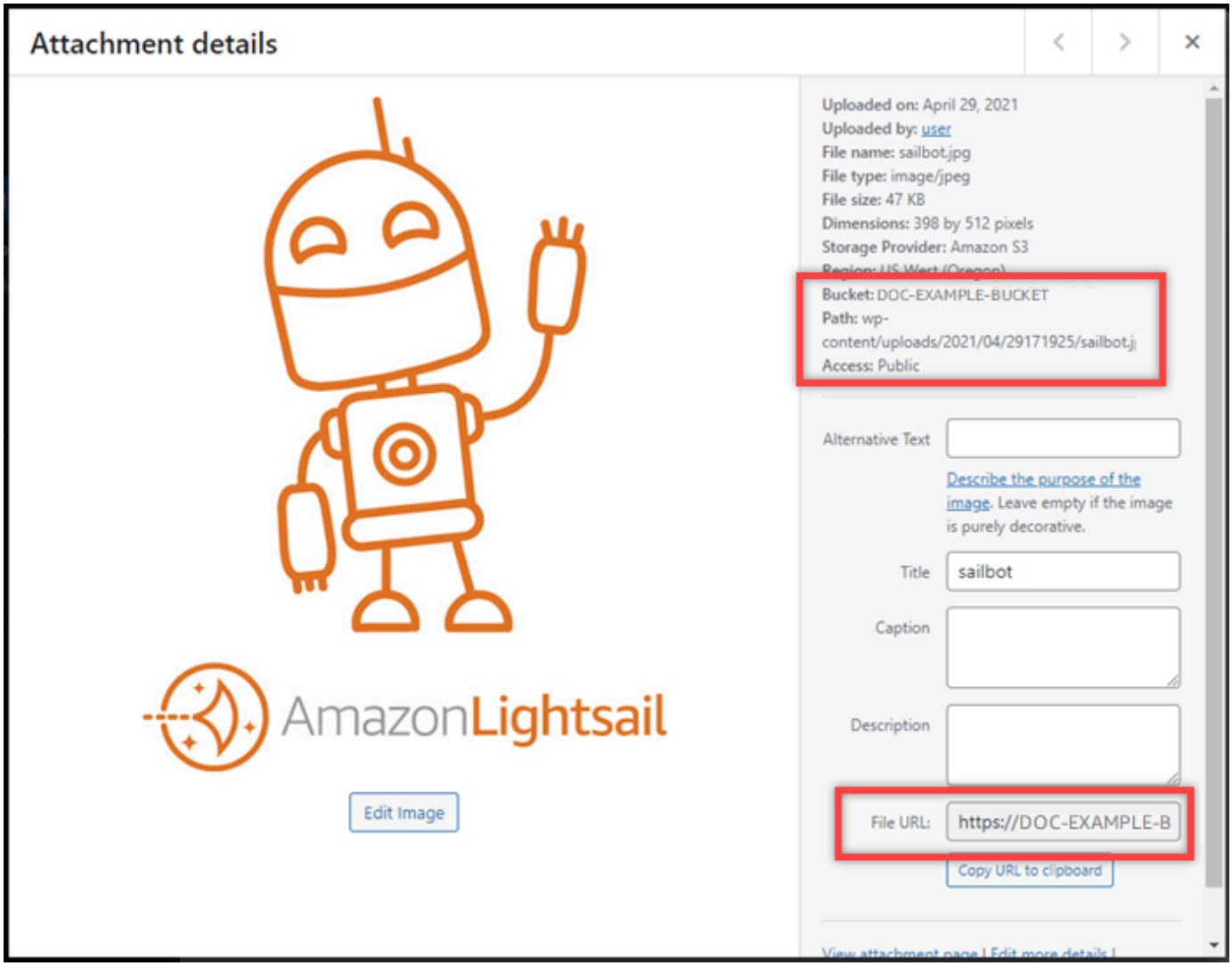
4. ファイルのアップロードが完了したら、左のナビゲーションメニューにある [メディア] の [ライブラリ] を選択します。



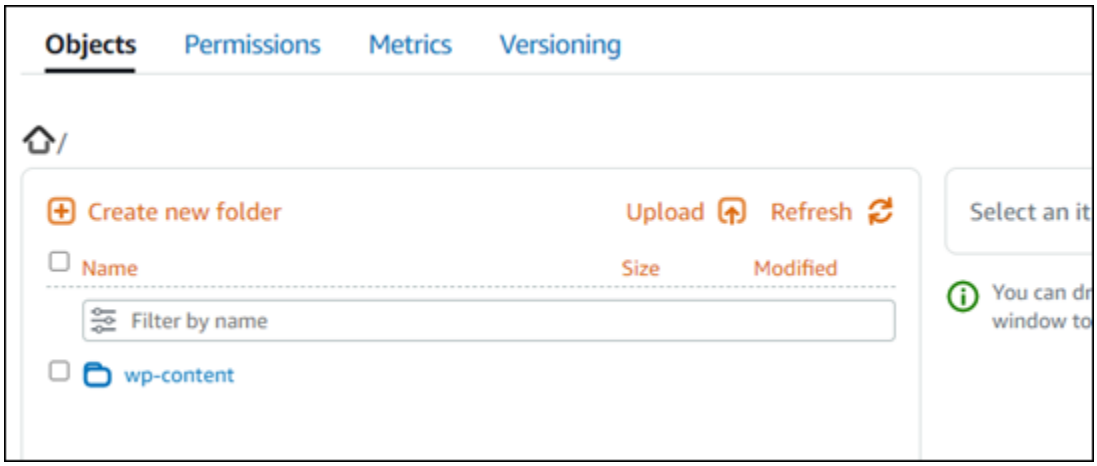
5. 最近アップロードしたファイルを選択します。



6. ファイルの詳細パネルには、バケットフィールドとファイルURLフィールドにバケットの名前が表示されます。



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」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、アクセスキーの作成、バケットへのインスタンスのアタッチ、他のAWSアカウントへのアクセスの付与によって、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」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 でバケット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の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バケットのアクセスログ記録を有効にする
   - 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、IAMAmazon Lightsail」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 のオブジェクトキー名について」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」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」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」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」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: Amazon Lightsail バケットへの WordPress インスタンスの接続
   - 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 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」を参照してください。

Lightsail コンテンツ配信ネットワーク WordPress で を設定する

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、デフォルトのドメイン (など) でデフォルトで HTTPS が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 のみを使用する – 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、暗号化されていない接続を使用して SSL/TLS を終了し、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 を使用する – 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

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 WordPress。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. オリジンを選択 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、 に最適 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、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 レコードの更新 で、理解している 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のテスト」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 でインスタンスの WordPress E メールを有効にする

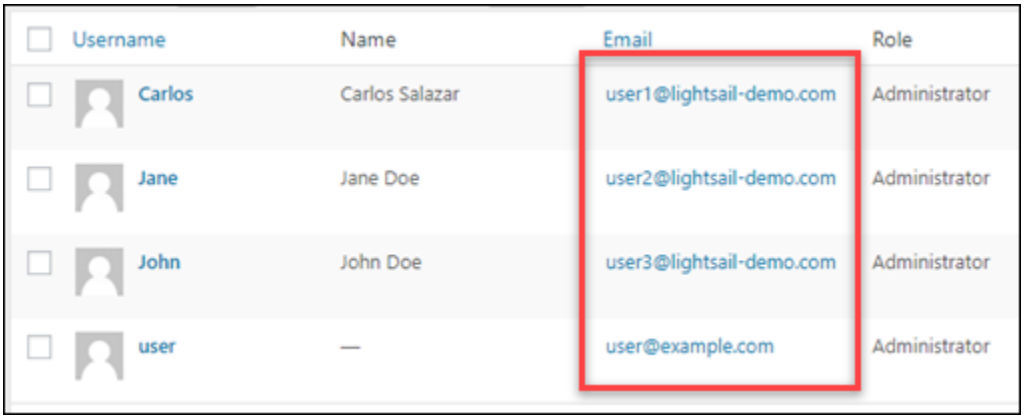
Amazon Lightsail の WordPress インスタンスで E メールを有効にできます。Amazon Simple Email Service (Amazon SES) で SMTP サービスを設定します。次に、インスタンスで WP Mail SMTP プラグインを有効化して設定します。E メールを有効にすると、 WordPress 管理者はユーザープロファイルのパスワードリセットをリクエストでき、ブログ投稿、ウェブサイトの更新、その他のプラグインメッセージに関する E メール通知が送信されます。このガイドでは、Amazon SES を使用して Amazon Lightsail の WordPress インスタンスで E メールを有効にする方法について説明します。Amazon SES

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。アカウントでその場合は、ウェブサイトのドメインを確認し、管理者の WordPress E メールアドレスを確認することをお勧めします。E 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

インスタンスで WordPress E メールを有効にする前に、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 Amazon Lightsail」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 インスタンスにポイントします。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

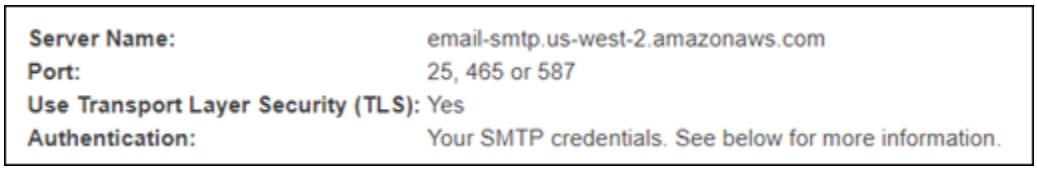
ステップ 3: Amazon SES で SMTP 認証情報を作成する

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。

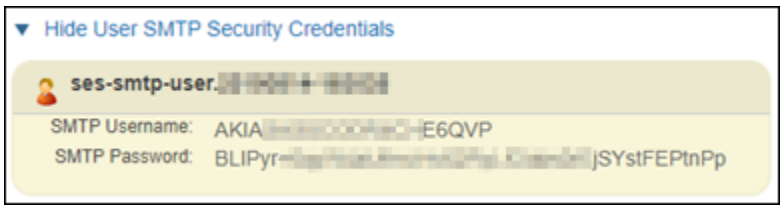
   [SMTP 設定] ページには、SMTP サーバー名、ポート、および TLS 設定が表示されます。WordPress インスタンスで WP Mail SMTP プラグインを設定するときに、このガイドの後半で必要になるため、これらの値を書き留めておきます。

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。これらの認証情報は、後で WordPress インスタンスで WP Mail SMTP プラグインを設定するときに必要になります。



> **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。

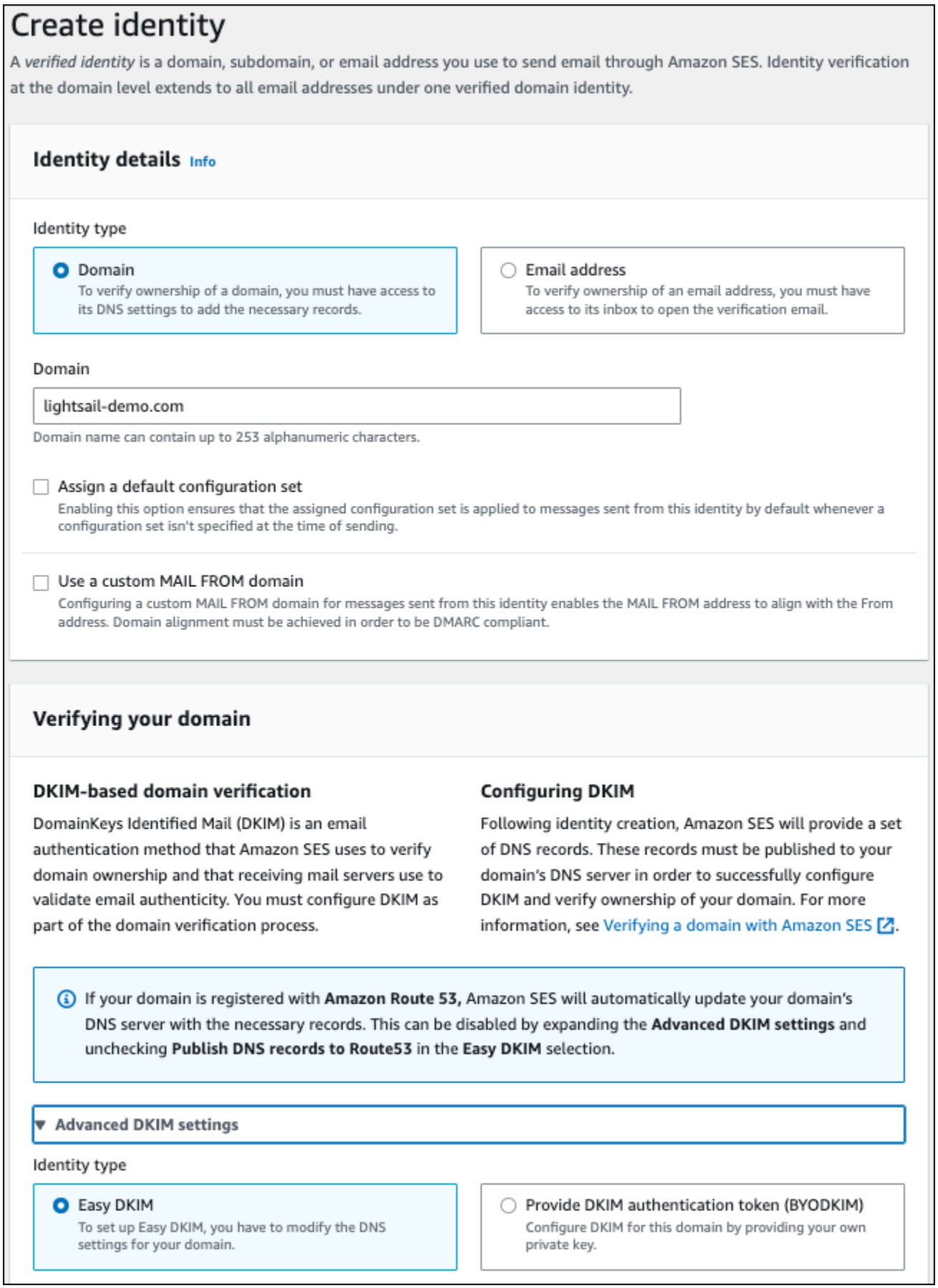
Amazon SES のドメインを検証するには

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のと同じドメインである必要があります。

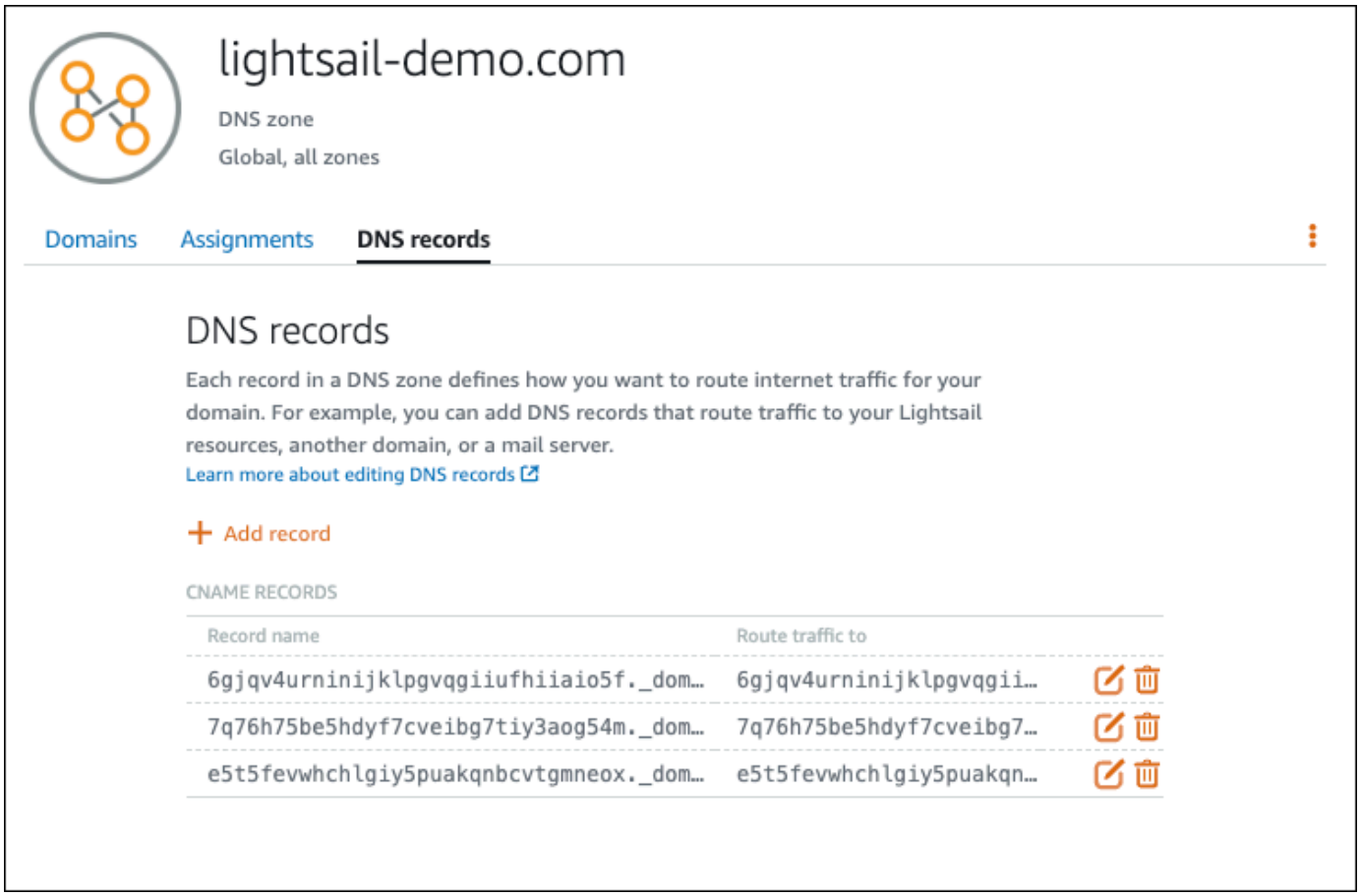
> **Important**
>
> レガシー TXT レコード
> Amazon SES でのドメイン検証は、E メールサーバーを受信して E メールの信頼性を検証するために使用する E メール認証標準である Identified DomainKeys Mail (DKIM) に基づくようになりました。ドメインの DNS 設定で DKIM を設定すると、ユーザーがアイデンティティの所有者であることが SES に確認されるため、TXT レコードは不要に

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 で DNS ゾーンを編集する方法の詳細については、Amazon Lightsail」を参照してください。

   結果は次の例のようになります。

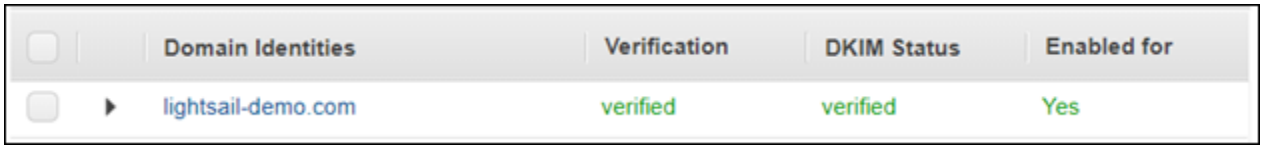


> **Note**
>
> [サブドメイン] テキストボックスに @ のシンボルを入力して、MX レコードにドメインの頂点を使用します。また、Amazon SES で指定された MX レコード値は `10 inbound-smtp.us-west-2.amazonaws.com` になります。10 を [Priority] (優先)、お

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 ウェブサイト管理者の E 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

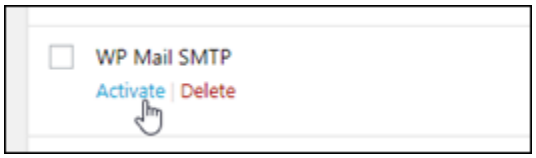
1. ID の作成に使用した E メールアドレスの受信トレイをチェックし、no-reply-aws@amazon.com からの E メールを探します。
2. E メールを開き、E メールのリンクをクリックして、E メールアドレスの検証プロセスを完了します。完了したら、ID の状態が検証済みに更新されます。

ステップ 6: WordPress インスタンスで WP メール SMTP プラグインを設定する

最後のステップは、 WordPress インスタンスで WP メール SMTP プラグインを設定することです。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。

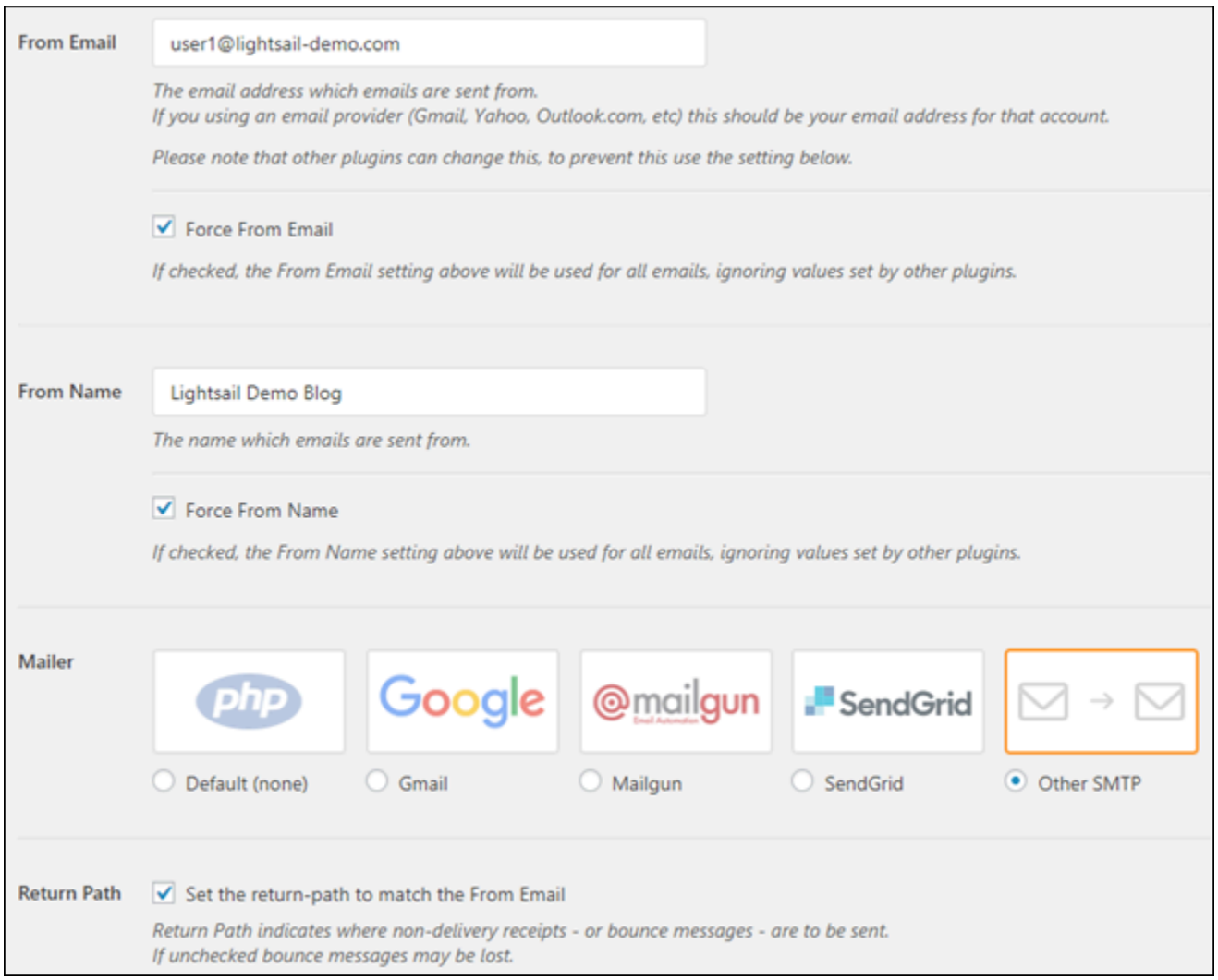
WordPress インスタンスで WP メール SMTP プラグインを設定するには

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。

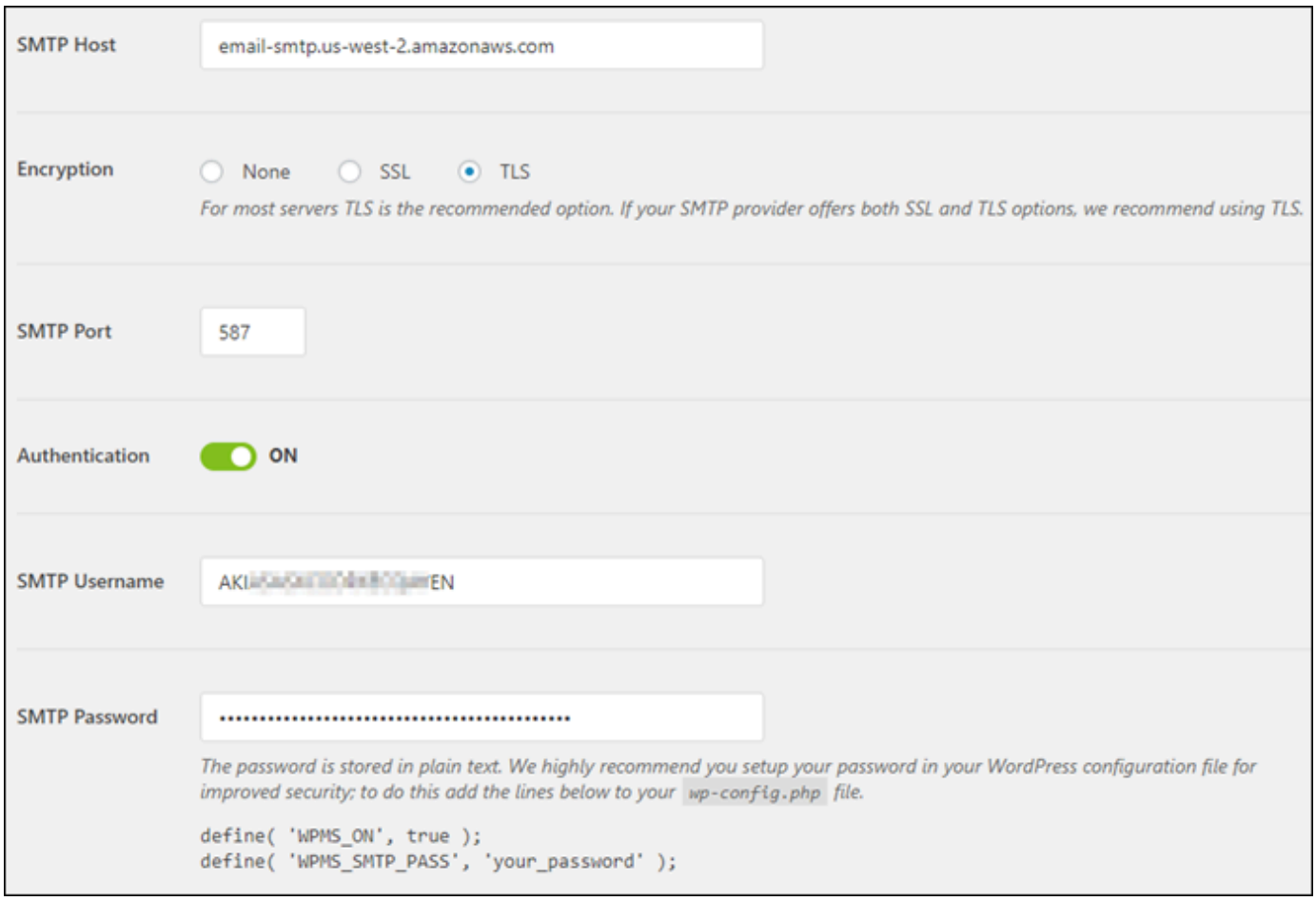


4. WP Mail SMTP プラグインが有効になったら、[設定] を選択します。下にスクロールしてプラグインを見つける必要がある場合があります。

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 テキストボックスに、E メールの送信元となる名前を入力するか、 WordPress ブログの名前をそのまま使用します。
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「from name」値が無視され、From Name テキストボックスに入力した名前が強制 WordPress 的に使用されます。
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。

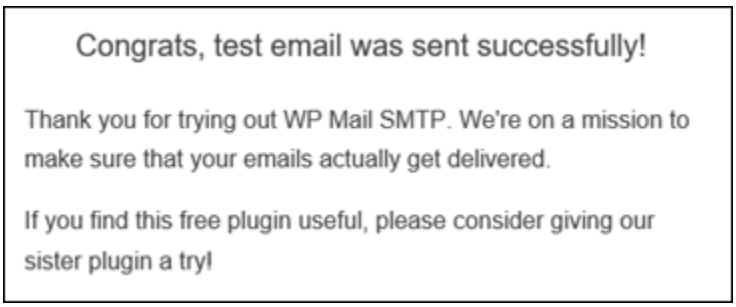


15. [Save settings (設定を保存)] を選択します。設定が正常に保存されたことを確認するプロンプトが表示されます。
16. [Email Test (E メールテスト)] タブを選択します。

    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 ウェブサイトで E メールが有効になります。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 ウェブサイトのダッシュボードのサインインページで、パスワードの紛失を選択できるようになりました。 WordPress ユーザープロファイルの E メールアドレスが Amazon SES で確認されると、新しいパスワードが E 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 で HTTPS を使用して WordPress サイトを保護する

WordPress ウェブサイトの Hypertext Transfer Protocol Secure (HTTPS) を有効にすると、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訪問者に保証します。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、 WordPress インスタンスでの 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。このチュートリアルの手動ステップに従う代わりに、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 を有効にする方法について説明します。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: WordPress 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。

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、訪問者が暗号化された接続の使用を強制されるように、訪問者を HTTP から HTTPS に自動的にリダイレクトするように WordPress ウェブサイトを設定できます (http://example.com にリダイレクトhttps://example.com) 。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。

- 作成時に Bitnami によって認定 WordPressされたすべてのインスタンスにプリインストールされているわけではありません。しばらくの間 Lightsail で作成された WordPress インスタンスでは、bncertツールを手動でインストールする必要があります。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 の WordPress インスタンスで Let's Encrypt SSL 証明書を使用する Amazon Lightsail」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 で WordPress インスタンスを作成し、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」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 インスタンスの設定が完了したら、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を作成する」またはAmazon Lightsail」を参照してください。

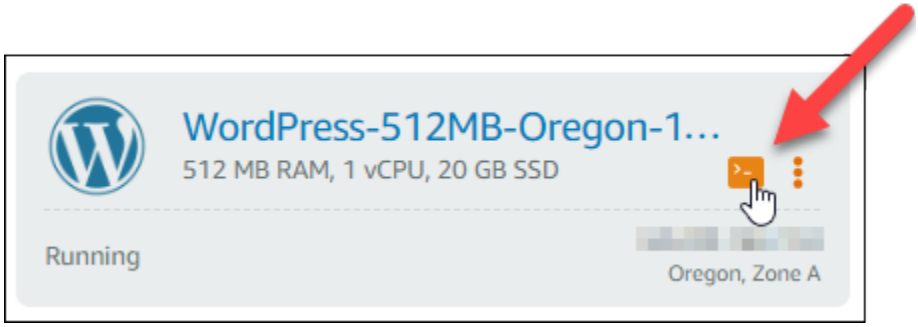
- ドメインの頂点 (example.com) とそのwwwサブドメイン () のトラフィックを Lightsail の WordPress インスタンスのパブリック IP アドレスに送信する DNS レコードwww.example.comをドメインの DNS に追加します。これらのアクションは、ドメインの現在の DNS ホスティングプロバイダーで実行することができます。または、ドメインの DNS の管理を Lightsail に移管した場合は、Lightsail の DNS ゾーンを使用してこれらのアクションを実行できます。詳細については、「DNS」を参照してください。

> **Important**
>
> ウェブサイトで使用するすべてのドメインの DNS に DNS WordPressレコードを追加します。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。このbncertツールは、現在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 WordPress。

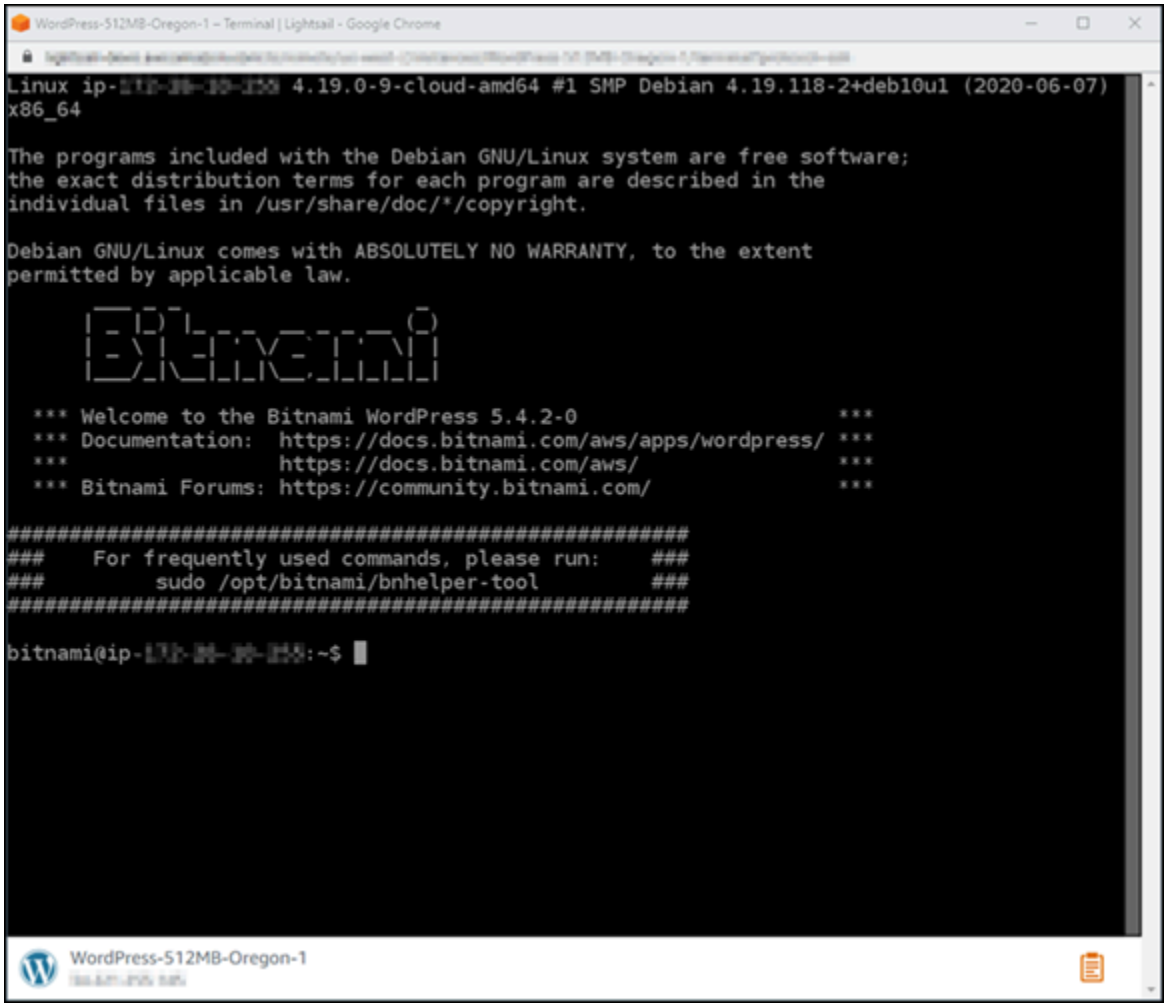
ステップ 3: インスタンスに接続する

Lightsail コンソールでブラウザベースの 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、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 ホームページで、インスタンスの WordPress SSH クイック接続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

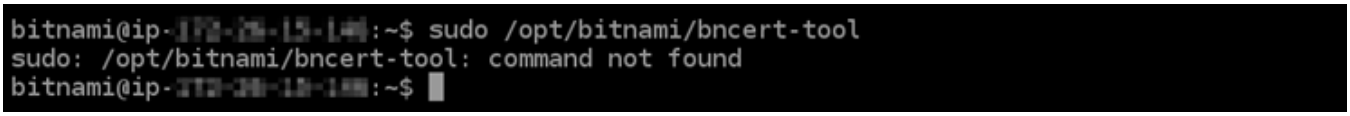
次のステップを完了して Bitnami HTTPS 設定ツール (bncert) がインスタンスにインストールされていることを確認します。Bitnami インスタンスの作成時に、すべての認定 WordPress インスタンスにプリインストールされているわけではありません。 WordPress Lightsail でしばらく前に作成された インスタンスでは、bncertツールを手動でインストールする必要があります。このステップでは、ツールがインストールされていない場合にツールをインストールする方法を説明します。

1. bncert ツールを実行するには、次のコマンドを入力します。

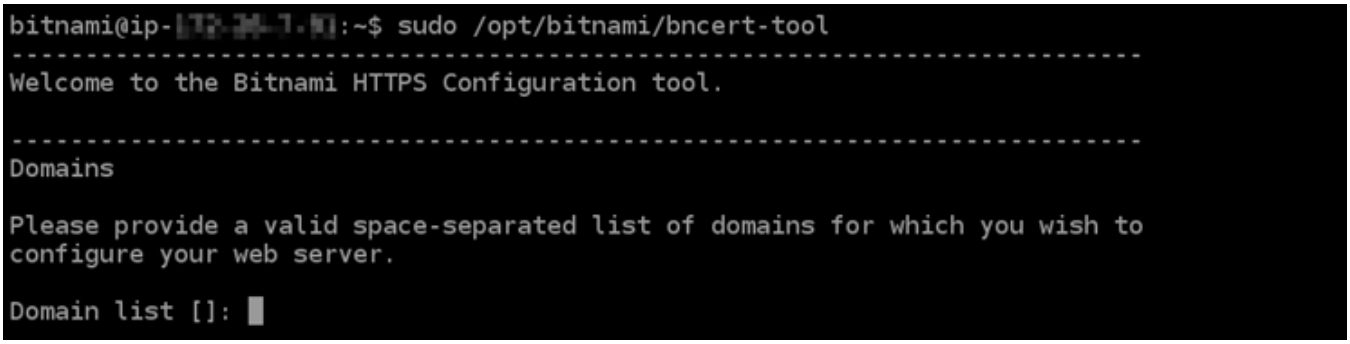
```
sudo /opt/bitnami/bncert-tool
```

- 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**
  >
  > このbncertツールは、Bitnami によって認定された WordPress 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 WordPress インスタンスで HTTPS を有効にすることもできます。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 WordPress」を参照してください。



- 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」セクションに進みます。



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」セクションに進みます。

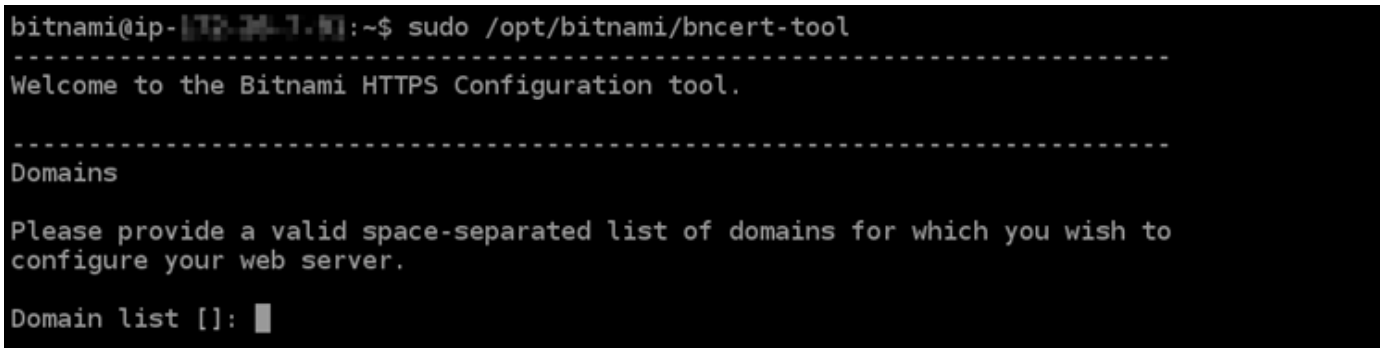
ステップ 5: インスタンスで WordPress HTTPS を有効にする

ツールが WordPress インスタンスにbncertインストールされていることを確認したら、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。



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:
```

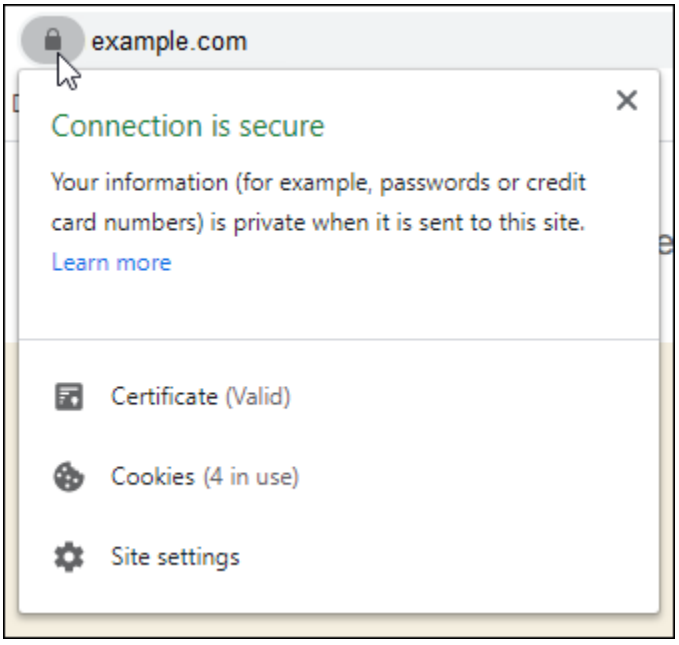
bncert ツールは、有効期限が切れる前、 80 日ごとに証明書の自動更新を実行します。インスタンスで追加のドメインやサブドメインを使用し、それらのドメインで HTTPS を有効にする場合は、上記のステップを繰り返します。

これで、 WordPress インスタンスで 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。

WordPress ブログを 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。

料金プラン (1 か月USDあたり 5 USD から) のいずれかを選択し、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。

「」を読んで開始してください。

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。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。
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しばらく経っている場合、またはクレジットカードをまだ提供していない場合は、まずにログイン AWS Management Console 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。

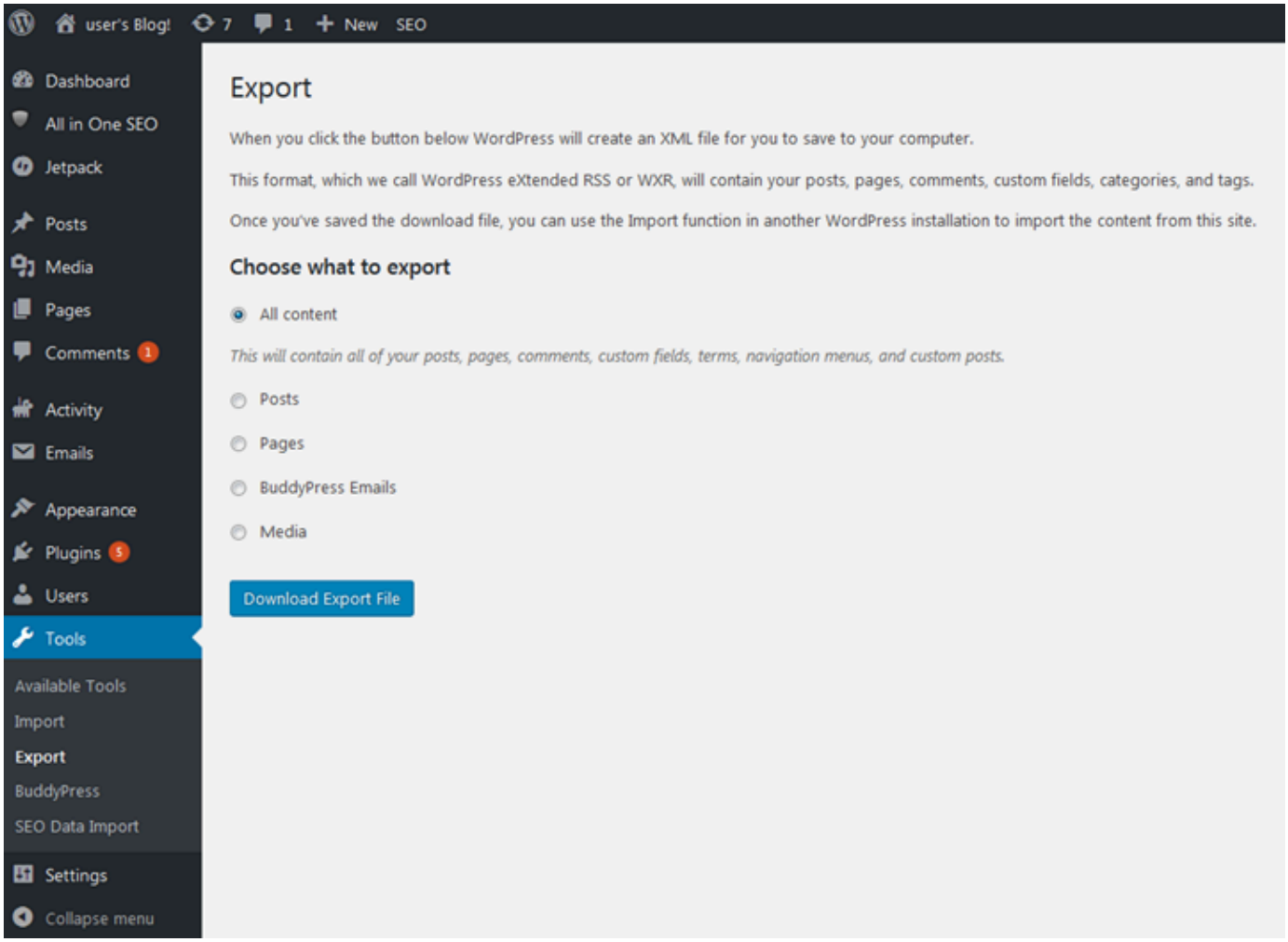
1. ブログに移動して [管理] を選択します。

   [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ファイルとしてエクスポートするには、すべてXMLコンテンツを選択します。



5. エクスポートファイルのダウンロード を選択して、古いブログを XML ファイルとしてダウンロードします。

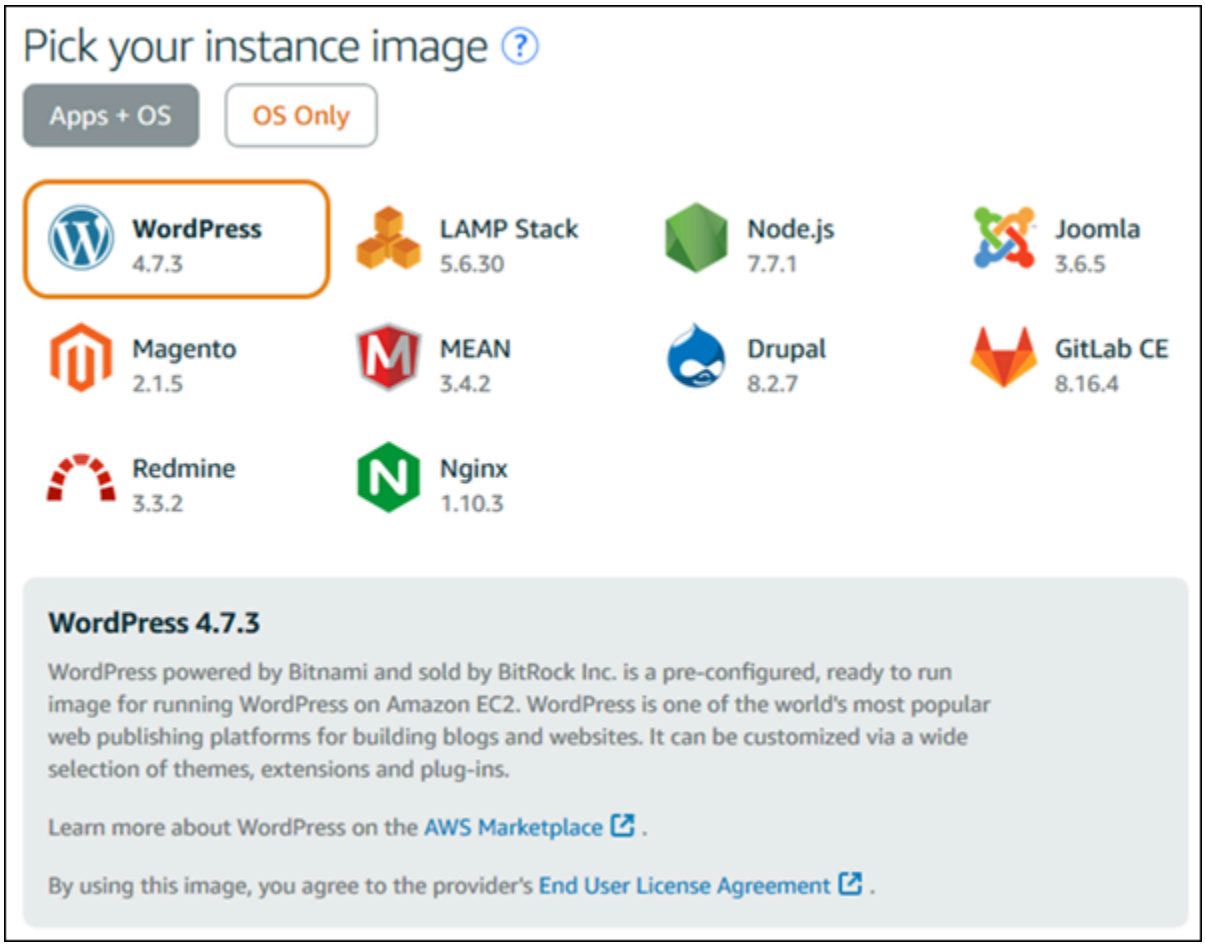
   見つけやすい場所にXMLファイルを保存します。このファイルはステップ 4 で必要になります。

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、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 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。

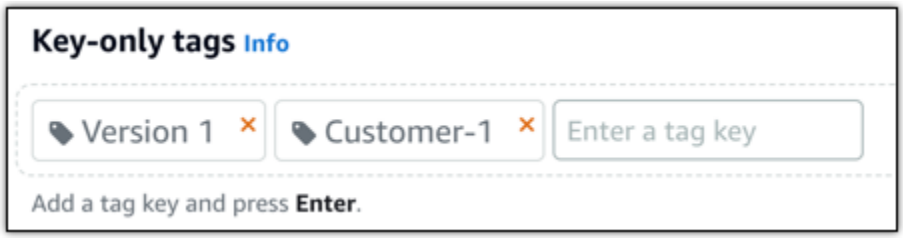
   Lightsail プランは、必要に応じて後でアップグレードできます。詳細については、「Lightsail でスナップショットからインスタンスを作成する」を参照してください。
6. インスタンスの名前を入力します。

リソース名:

- 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。
- 英数字で開始および終了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。

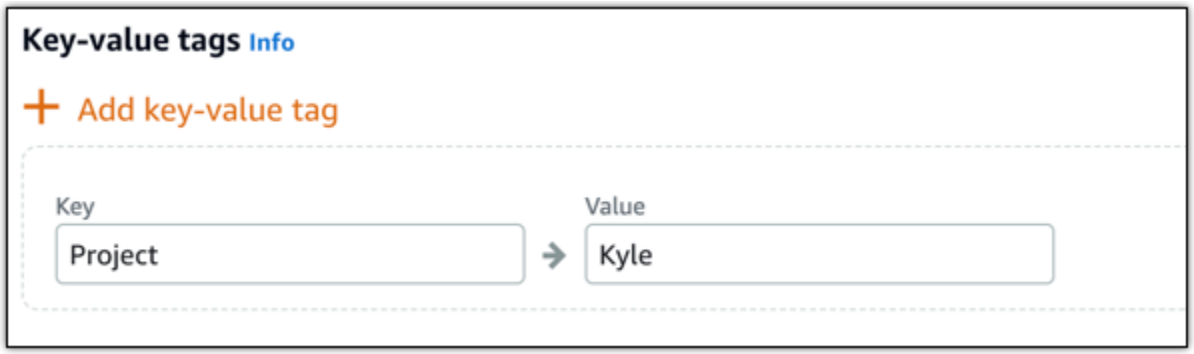
7. 以下のいずれかのオプションを選択して、インスタンスにタグを追加します。

- [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。

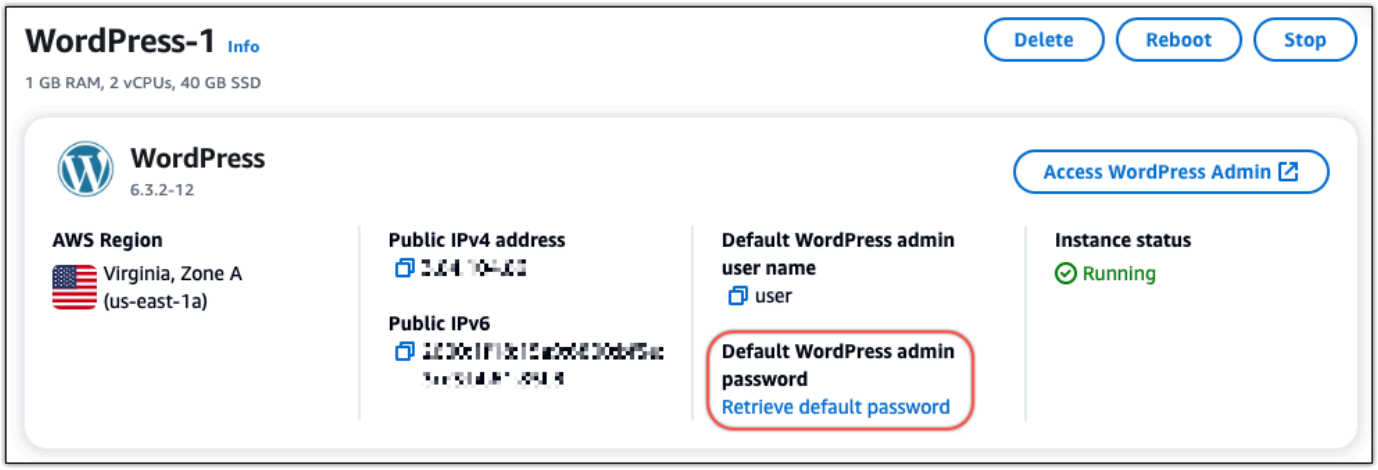
8. [インスタンスの作成] を選択します。

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。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 キーを押すか、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!
```

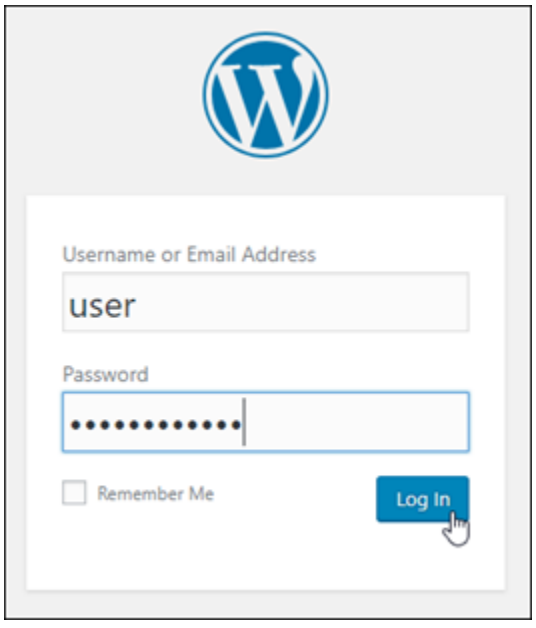
WordPress ウェブサイトの管理ダッシュボードのパスワードを取得したら、サインインできます。管理ダッシュボードでは、ユーザーパスワードの変更、プラグインのインストール、ウェブサイトのテーマの変更などを行うことができます。

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。

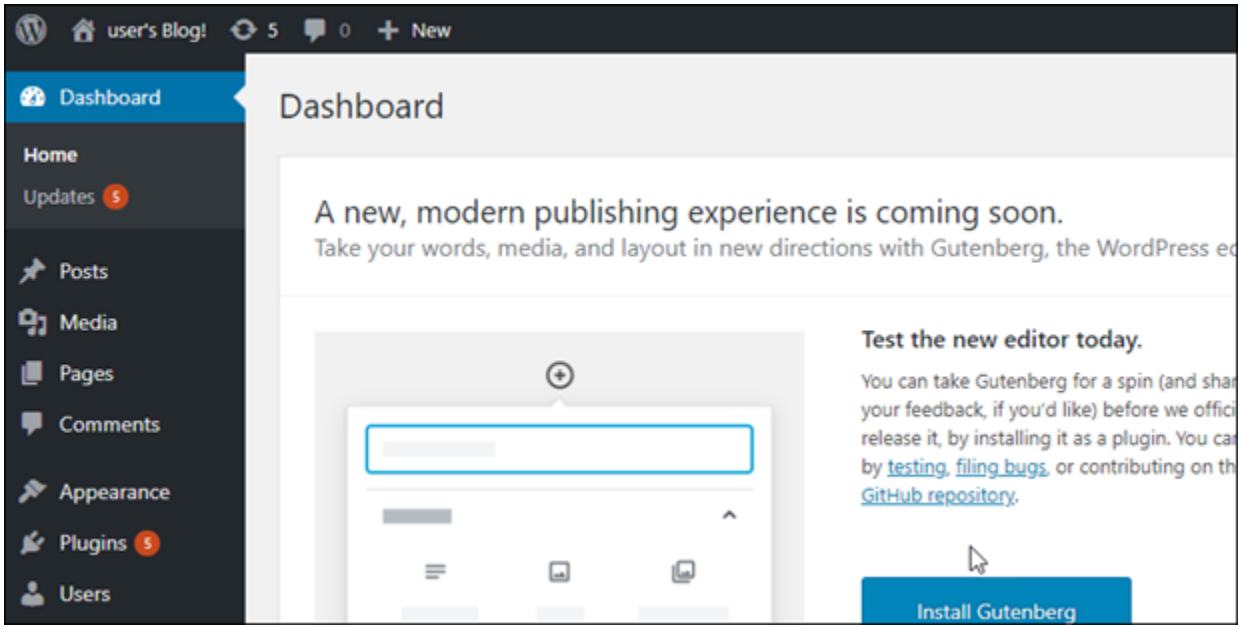
管理ダッシュボードにサインインするには

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 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。



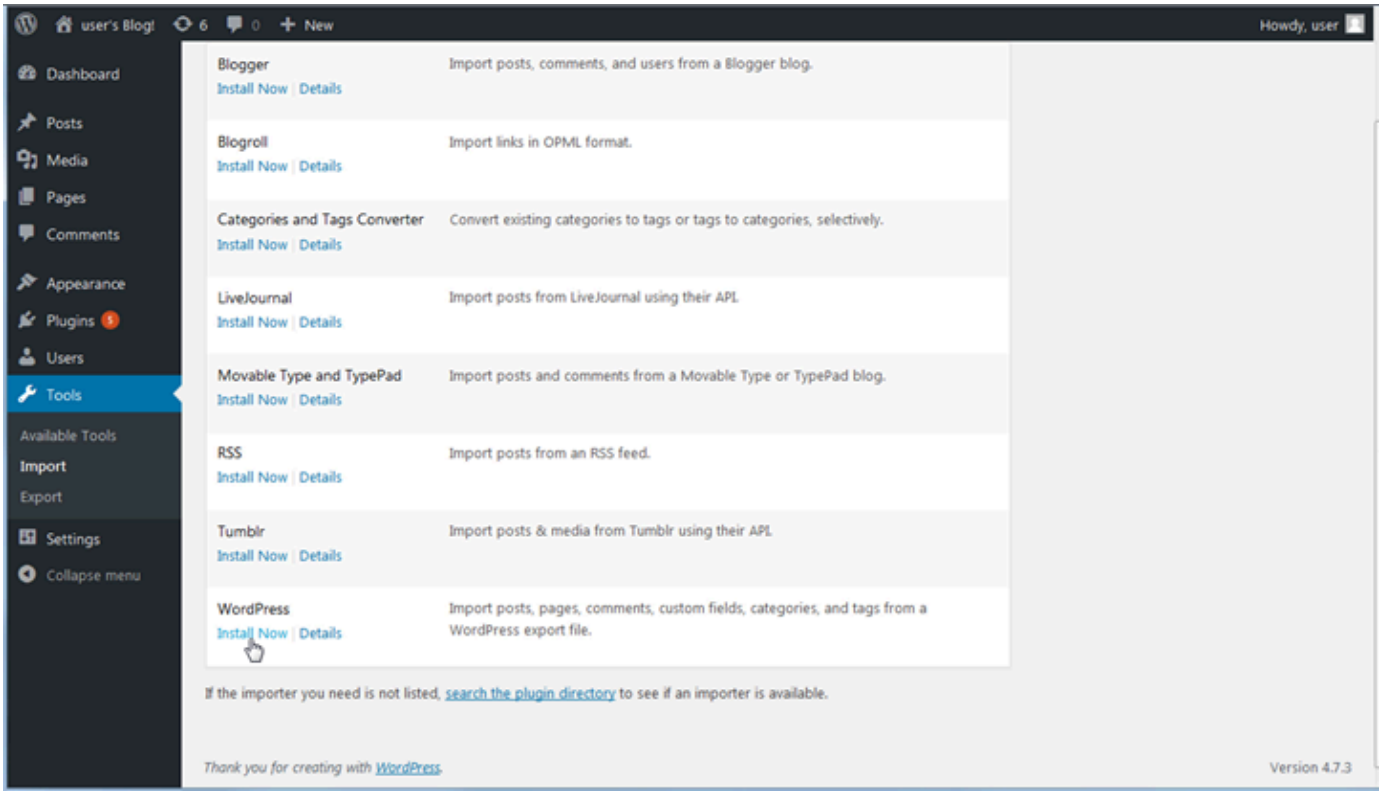
これで、管理アクションを実行できる WordPress ウェブサイトの管理ダッシュボードにサインインしました。 WordPress ウェブサイトの管理の詳細については、 WordPress ドキュメントのWordPress「Codex」を参照してください。



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 インスタンスで WordPress Dashboard 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 をバックアップして保存した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。

- を移行して、ドメインネームサーバーがブログの新しいバージョンを参照DNSするようにします。

- 新しいブログの外観をカスタマイズしたり、 WordPress プラグインをインストールしたりします。
- SSL証明書でHTTPSサポートを有効にする

step-by-step 手順に従って、 WordPress インスタンスを起動して設定し、 で保護しHTTPS、外部データベースまたはストレージサービスに接続し、既存のブログを Lightsail に移行します。このチュートリアルでは、 WordPress 管理者認証情報の取得、プラグインのインストール、 DNSとドメインの設定、Amazon S3、Amazon Aurora、Amazon などの他の との統合 AWS サービス などの重要なタスクについて説明しますSES。このガイドに従うことで、Lightsail プラットフォームで、安全でスケーラブル、高性能な WordPress ウェブサイトを簡単にセットアップおよび管理できます。

マルチサイト on Lightsail で複数の WordPress サイトを管理する

このセクションでは、Amazon Lightsail のマルチサイトインスタンスでの WordPressブログの管理に関連する以下のトピックについて説明します。

トピック

- ブログをドメインとして Lightsail の WordPress マルチサイトに追加する
- Lightsail の WordPress マルチサイトにブログをサブドメインとして追加する
- Lightsail で 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

ブログをドメインとして Lightsail の WordPress マルチサイトに追加する

Amazon Lightsail の WordPress マルチサイトインスタンスは、そのインスタンス内に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 マルチサイト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 `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**
>
> サブドメインを使用するサイトを WordPress マルチサイト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「マルチサイト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 マルチサイト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、Lightsail の WordPress マルチサイト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、 WordPress マルチサイトインスタンスにアタッチした静的 IP にポイントします。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 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します。詳細については、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」を参照してください。

ブログをドメインとして WordPress マルチサイトインスタンスに追加する

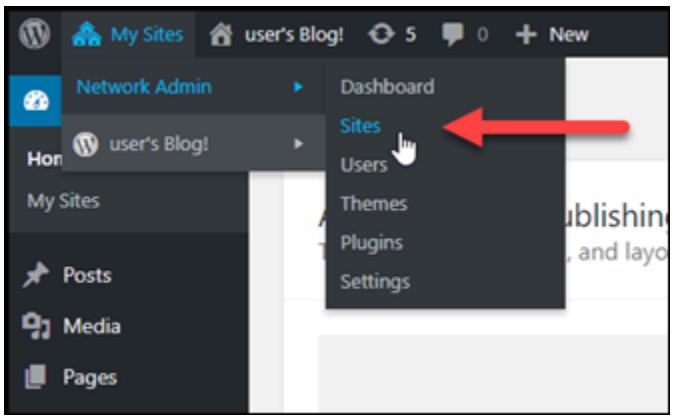
以下のステップを実行して、メインブログのプライマリドメインとは異なるドメインを使用するブログサイトを WordPress マルチサイトインスタンスに作成します。

> **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 マルチサイト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。

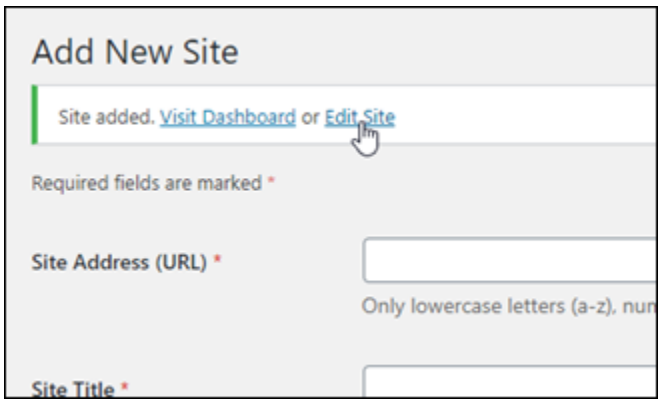
   > **Note**
   >
   > 詳細については、「Bitnami インスタンス向けにアプリケーションのユーザー名とパスワードを取得する」を参照してください。

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。

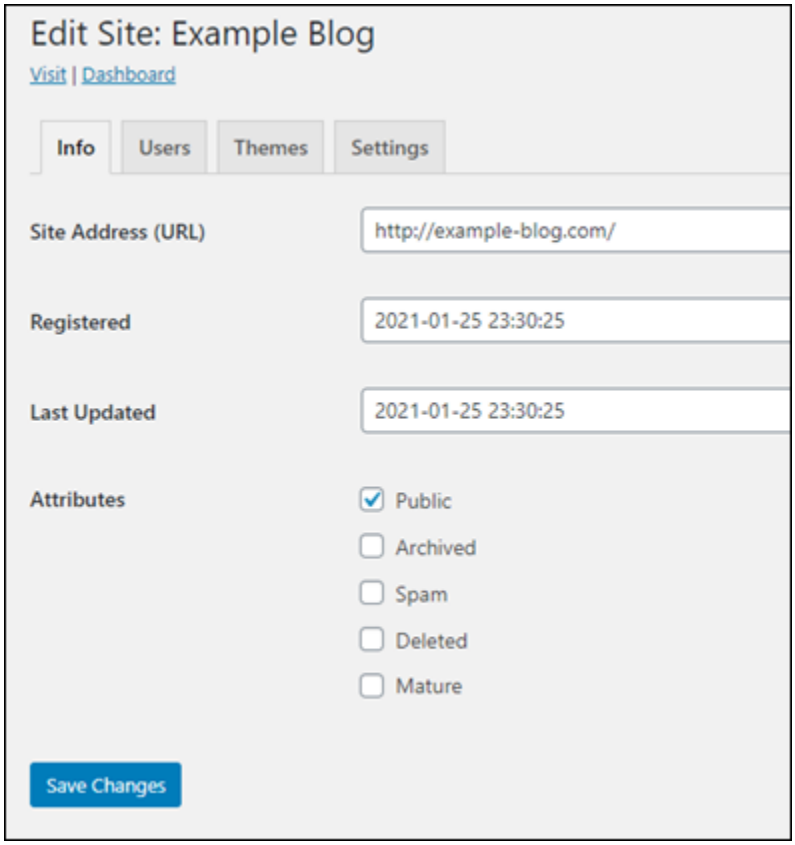


3. [Add New] (新規追加) を選択して新しいブログサイトを追加します。
4. サイトのアドレスをサイトアドレス (URL)テキストボックスに入力します。こちらが新しいブログサイトに使われるドメインになります。例えば、新しいブログサイトで `example-blog.com` をドメインとして使用する場合は、サイトアドレス (URL) テキストボックスに`example-blog`と入力します。ページに表示されるプライマリドメインのサフィックスは無視します。

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。

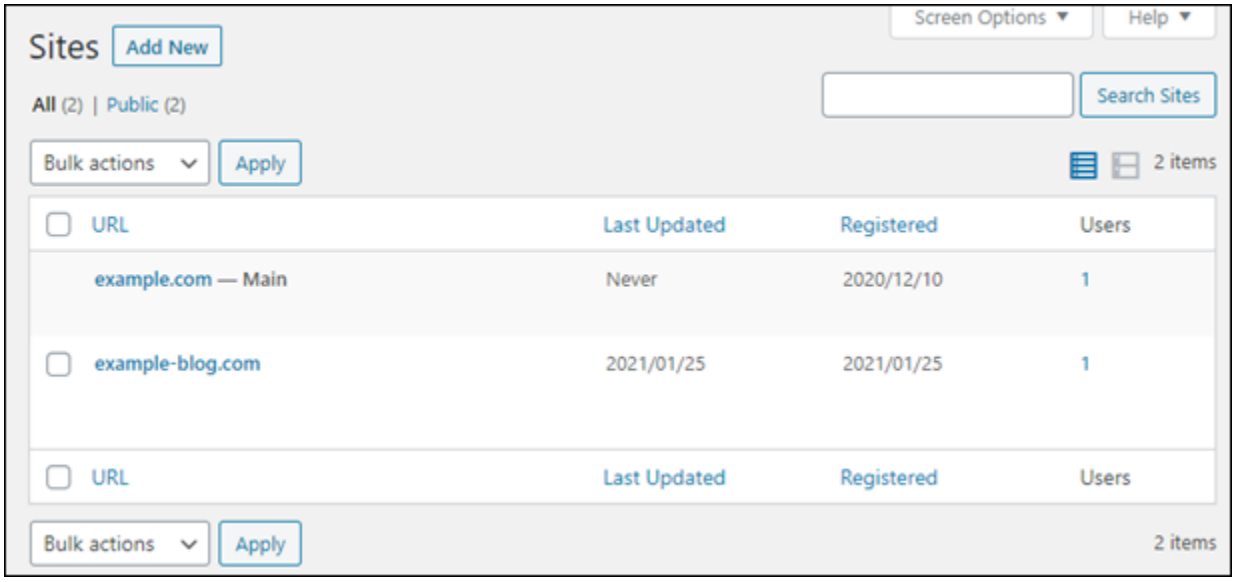


8. サイトの編集ページ上で、サイトアドレス (URL)テキストボックスにリストされているサブドメインを使用したい apex ドメインに変更します。この例では、`http://example-blog.com`を指定しました。



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 マルチサイトインスタンスに作成されていますが、ドメインはまだ新しいブログサイトにルーティングするように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

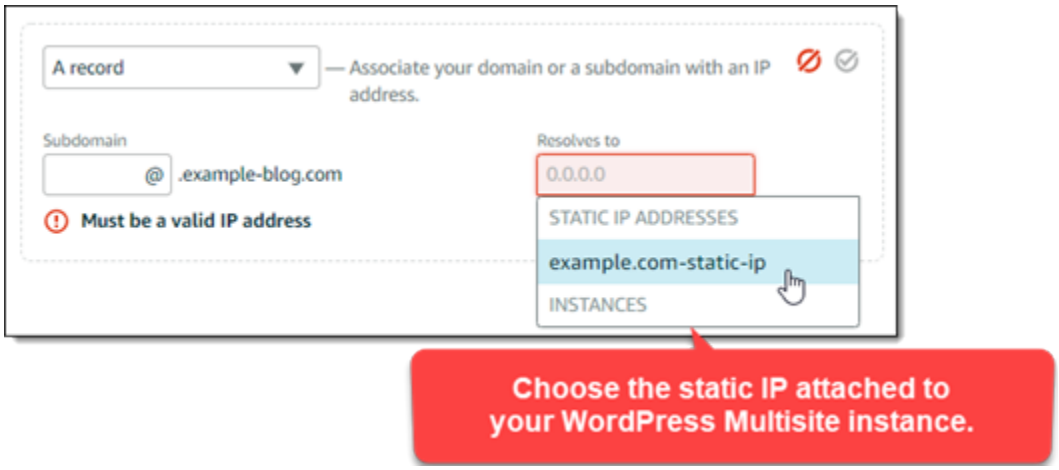
以下のステップを実行して、新しいブログサイトのドメインをマルチサイトインスタンスにポイントします WordPress。これらの手順は、 WordPress マルチサイトインスタンスで作成するすべてのブログサイトに対して実行する必要があります。

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 & 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. 「解決先への解決」テキストボックスで、 WordPress マルチサイト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。

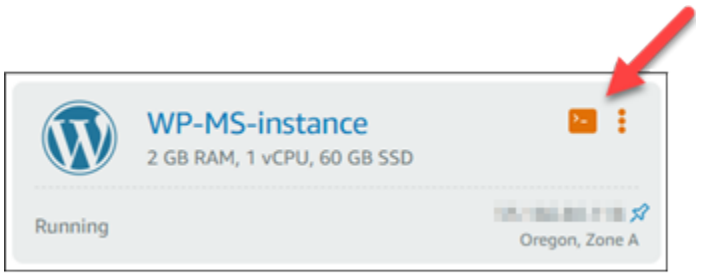
   変更がインターネットの DNS を介して伝播されると、ドメインは WordPress マルチサイトインスタンスの新しいブログサイトにトラフィックをルーティングします。

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 マルチサイトインスタンスに追加する場合は、インスタンス WordPress の設定 (`wp-config`) ファイルも更新して Cookie サポートを有効にする必要があります。Cookie サポートを有効にしない場合、ブログサイトの管理ダッシュボードに WordPressサインインしようとすると、ユーザーに「エラー: Cookie がブロックされているか、サポートされていない」という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 ホームページで、マルチサイトインスタンスの WordPress 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、Vim を使用してインスタンスの wp-config.php ファイルを開いて編集します。

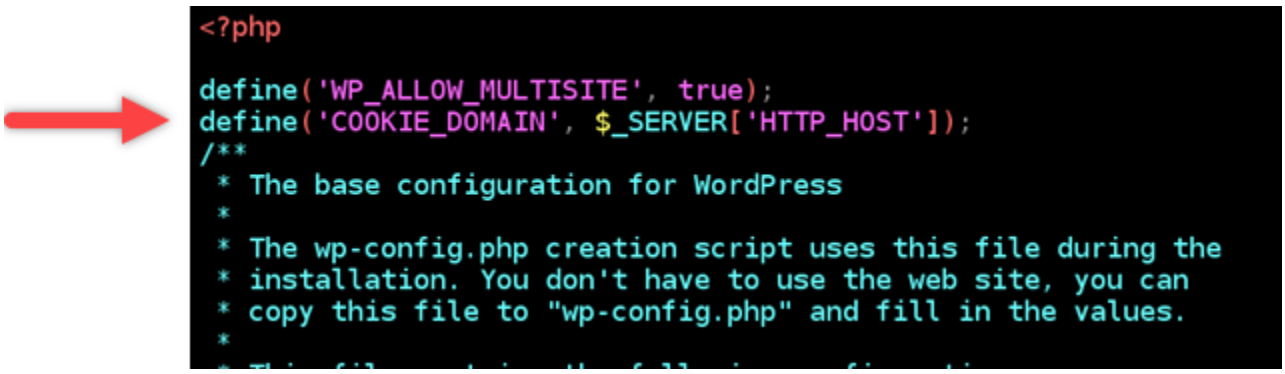
```
sudo vim /opt/bitnami/wordpress/wp-config.php
```

> **Note**
>
> このコマンドが WordPress失敗した場合、古いバージョンのマルチサイト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、インスタンスの基盤となるサービスを再起動します WordPress。

```
sudo /opt/bitnami/ctlscript.sh restart
```

これで、 WordPress マルチサイトインスタンスで Cookie を有効にする必要があり、ブログサイトにサインインしようとしているユーザーは「Error: Cookies are blocked or not supported」エラーに遭遇しません。

次のステップ

ブログを WordPress マルチサイトインスタンスにドメインとして追加したら、 WordPress マルチサイト管理に慣れることをお勧めします。詳細については、 WordPress ドキュメントの「マルチサイトネットワーク管理」を参照してください。

Lightsail の WordPress マルチサイトにブログをサブドメインとして追加する

Amazon Lightsail の WordPress マルチサイトインスタンスは、そのインスタンス内に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、ブログサイトを WordPress マルチサイトインスタンスのサブドメインとして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**
>
> ドメインを使用するサイトを WordPress マルチサイト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「マルチサイトインスタンス 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 マルチサイト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、 WordPress マルチサイト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、 WordPress マルチサイトインスタンスにアタッチした静的 IP にポイントします。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 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します。詳細については、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」を参照してください。

ブログをサブドメイン WordPressとしてマルチサイトインスタンスに追加する

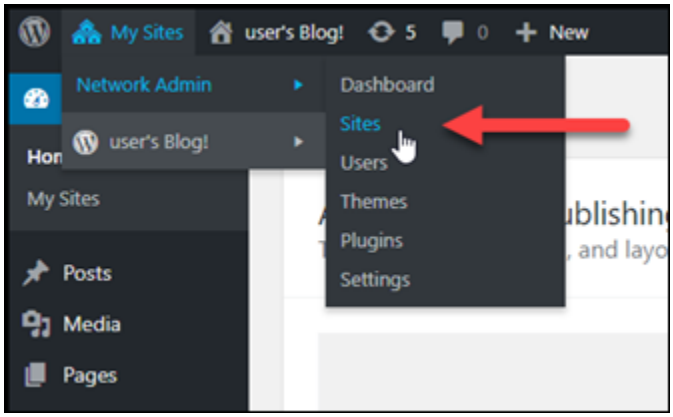
メインブログのプライマリドメインのサブドメインを使用する WordPress マルチサイトインスタンスに新しいブログを作成するには、以下の手順を実行します。

> **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。

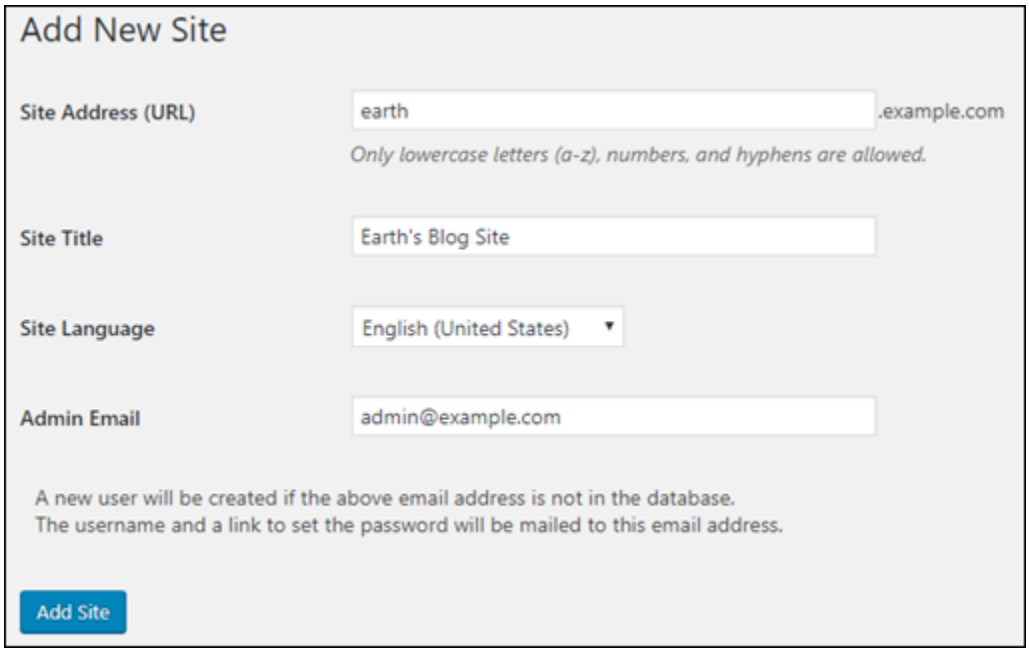
1. WordPress マルチサイトインスタンスの管理ダッシュボードにサインインします。

   > **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。

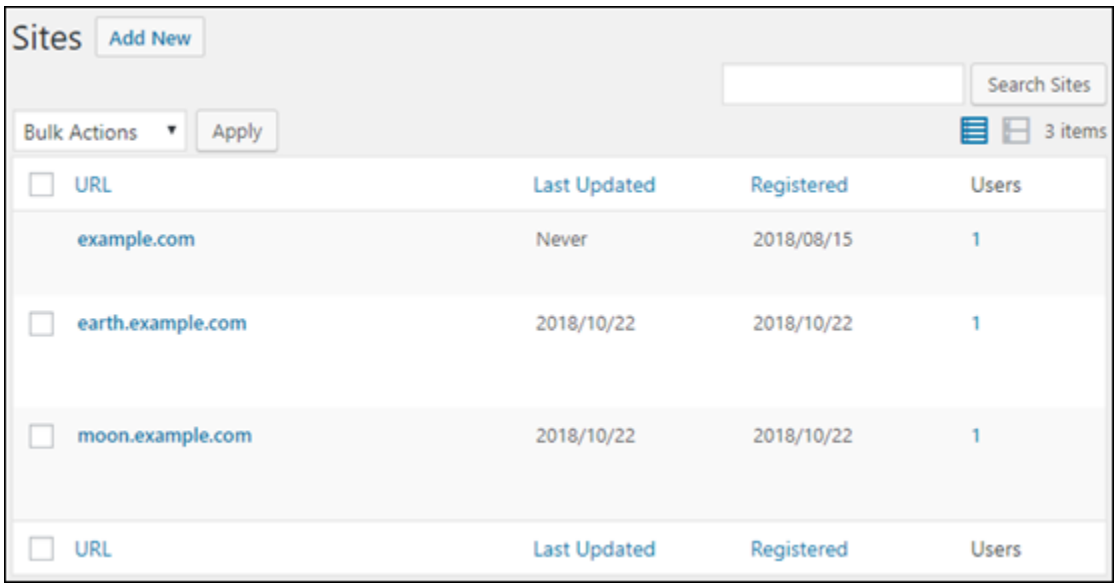


3. [Add New] (新規追加) を選択して新しいブログサイトを追加します。
4. 新しいブログサイトのサブドメインとして使用するサイトアドレスを入力します。



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 マルチサイトインスタンスに作成されていますが、サブドメインはまだ新しいブログサイトにルーティングするように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。

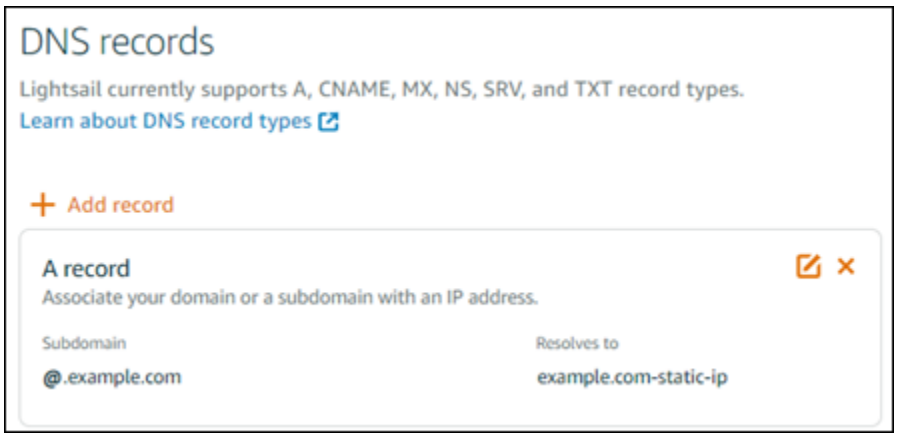


アドレスレコード (A レコード) をドメインの DNS ゾーンに追加する

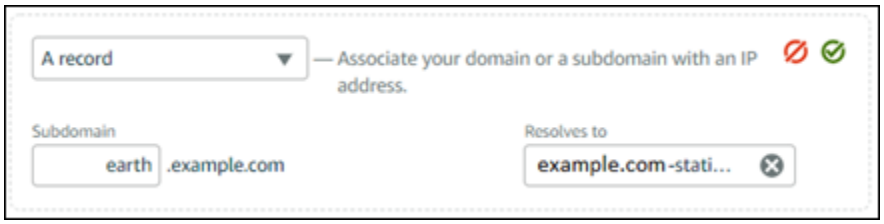
以下のステップを実行して、新しいブログサイトのサブドメインをマルチサイトインスタンスにポイントします WordPress。これらの手順は、 WordPress マルチサイトインスタンスで作成するすべてのブログサイトに対して実行する必要があります。

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 & 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 マルチサイト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. 「レコード名」テキストボックスに、 WordPress マルチサイトインスタンスで新しいブログサイトを作成するときに、サイトアドレスとして指定されたサブドメインを入力します。
7. 「解決先への解決」テキストボックスで、 WordPress マルチサイト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 を介して伝播されると、ドメインは WordPress マルチサイト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

ブログを WordPress マルチサイトインスタンスにサブドメインとして追加したら、 WordPress マルチサイト管理に慣れることをお勧めします。詳細については、 WordPress ドキュメントの「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 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

Amazon Lightsail の WordPress マルチサイトインスタンスは、そのインスタンス内に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 マルチサイトインスタンスのメインブログに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 マルチサイト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、Lightsail の WordPress マルチサイト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。

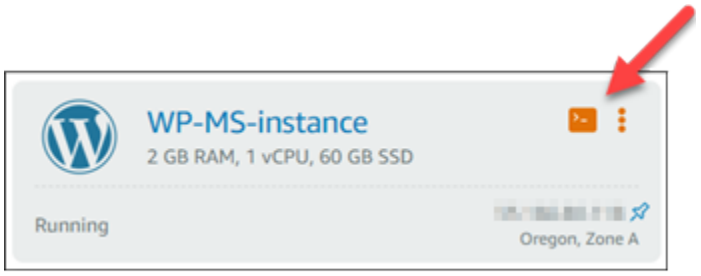
> **Important**
>
> 静的 IP をアタッチした後、 WordPress マルチサイトインスタンスを再起動する必要があります。これにより、インスタンスは、それに関連付けられた新しい静的 IP を認識できるようになります。

3. DNS ゾーンを作成してドメインを Lightsail に追加し、 WordPress マルチサイトインスタンスにアタッチした静的 IP にポイントします。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。次に、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 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

などのドメインが WordPress マルチサイト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するには`example.com`、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、マルチサイトインスタンスの WordPress 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、マルチサイトインスタンスの WordPressプライマリドメイン名を定義します。を WordPress マルチサイトに適したドメイン名*<domain>*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 WordPress失敗した場合、古いバージョンのマルチサイト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、 を WordPress マルチサイトに適したドメイン名*<domain>*に置き換えてください。
>
> ```
> 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 マルチサイト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

マルチサイトインスタンスのプライマリドメイン WordPressを定義したら、次のステップを実行します。

- ブログをサブドメインとして WordPress マルチサイトインスタンスに追加する
- ブログをドメインとして WordPress マルチサイトインスタンスに追加する

step-by-step 手順に従って、個別のドメインまたはサブドメインを使用して新しいブログサイトを追加する方法と、 WordPress マルチサイトインスタンスでメインブログのプライマリドメインを定義する方法について説明します。

このガイドでは、 WordPress マルチサイトインスタンスの作成、静的 IP のアタッチ、DNSゾーンの作成、プライマリドメインの設定などの前提条件について説明します。次に、ブログをドメインまたはサブドメインとして追加し、DNSレコードを更新し、Cookie サポートを有効にし、その他の必要な設定を実行するための詳細な手順を提供します。このガイドに従うことで、マルチサイトインスタンス内で複数のブログを効果的に管理および整理し、ブログサイトごとに個別のドメインまたはサブドメインを使用する柔軟性を活用できます WordPress。

Let's Encrypt で Lightsail リソースの暗号化された通信を有効にする

このガイドでは、Amazon Lightsail の Let's Encrypt に関連する以下のトピックについて説明します。開始する前に、次の前提条件を満たしていることを確認してください。

前提条件

- 、NginxLAMP、または を実行する Lightsail インスタンスを作成する WordPress
- ドメイン名を登録し、そのDNSレコードを編集するためのアクセス権を持つ
- Lightsail ブラウザベースのSSHターミナルまたは独自のSSHクライアントを使用します。

トピック

- Let's Encrypt SSL 証明書を使用して Lightsail LAMP インスタンスを保護する
- Let's Encrypt SSL/TLS を使用して Lightsail Nginx ウェブサイトを保護する
- 無料の Let's Encrypt SSL 証明書で Lightsail WordPress インスタンスを保護する

Let's Encrypt SSL 証明書を使用して Lightsail LAMP インスタンスを保護する

Amazon Lightsail を使用すると、Lightsail ロードバランサーを使用して SSL/TLS で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、一般的には適切な選択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 インスタンスと統合できます。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 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 DNS」を参照してください。

> **Note**
>
> 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、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、「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。

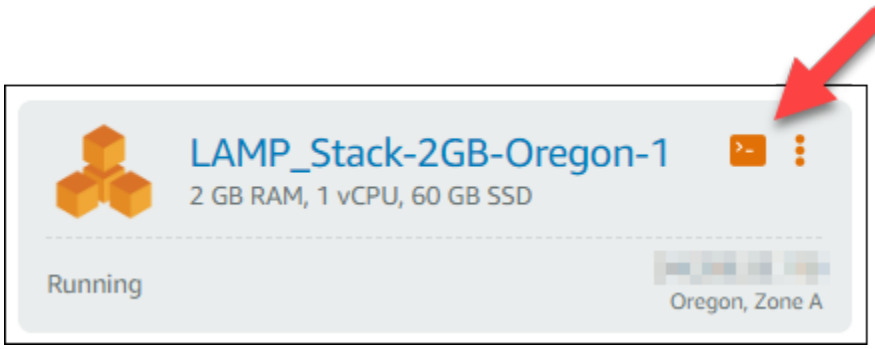
前提条件が完了したら、このチュートリアルの「次のセクション」に進みます。

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。

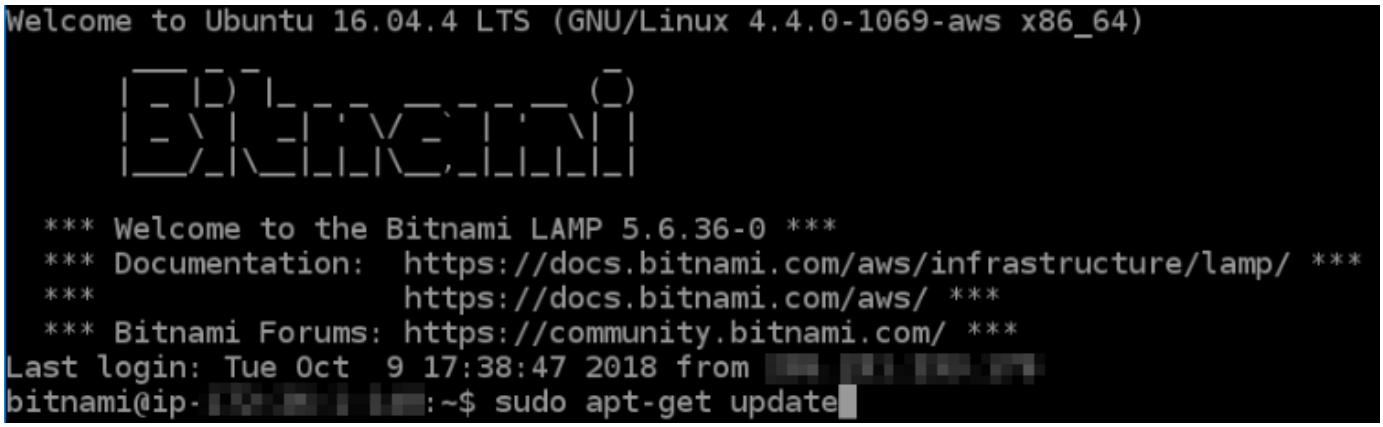
Lightsail インスタンスに Certbot をインストールするには

1. Lightsail コンソール にサインインします。
2. Lightsail 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 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、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=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。

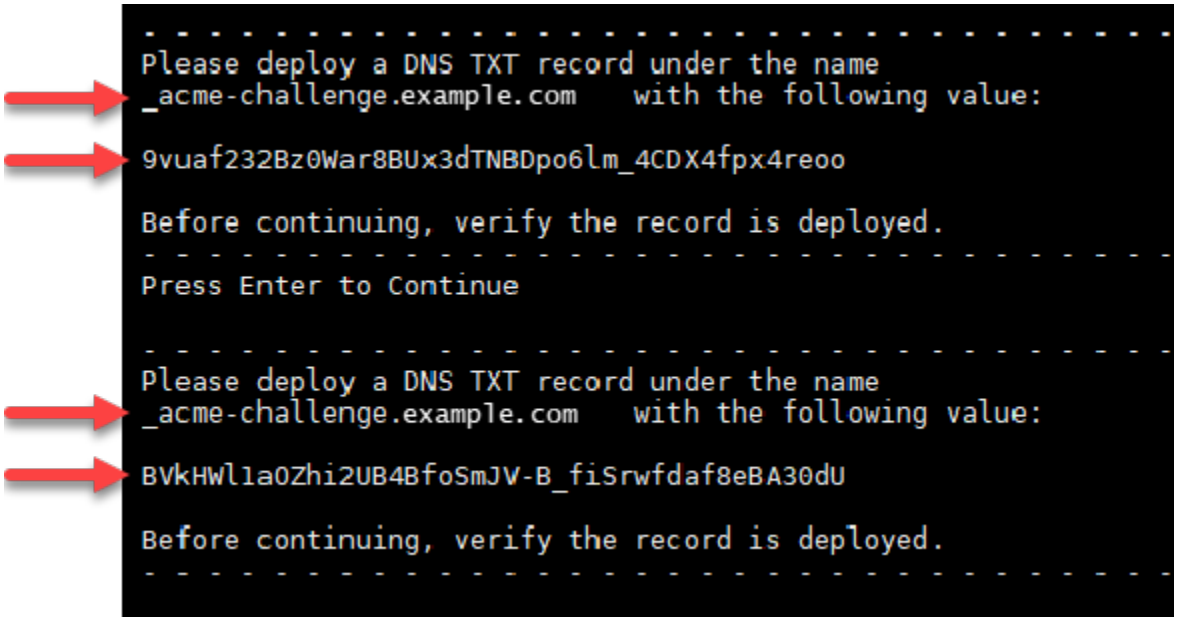
3. 次のコマンドを入力して Certbot をインタラクティブモードで起動します。このコマンドでは、DNS チャレンジで手動認証を使用してドメインの所有権を検証することを Certbot に指示します。また、最上位ドメインとそのサブドメイン用にワイルドカード証明書をリクエストします。

```
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 ブラウザベースの SSH セッションを開いたままにしておきます。このチュートリアルの後半に戻ります。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。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 DNS ゾーンの作成」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 & 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を開いたままにしておきます。このチュートリアルの後半に戻ります。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 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

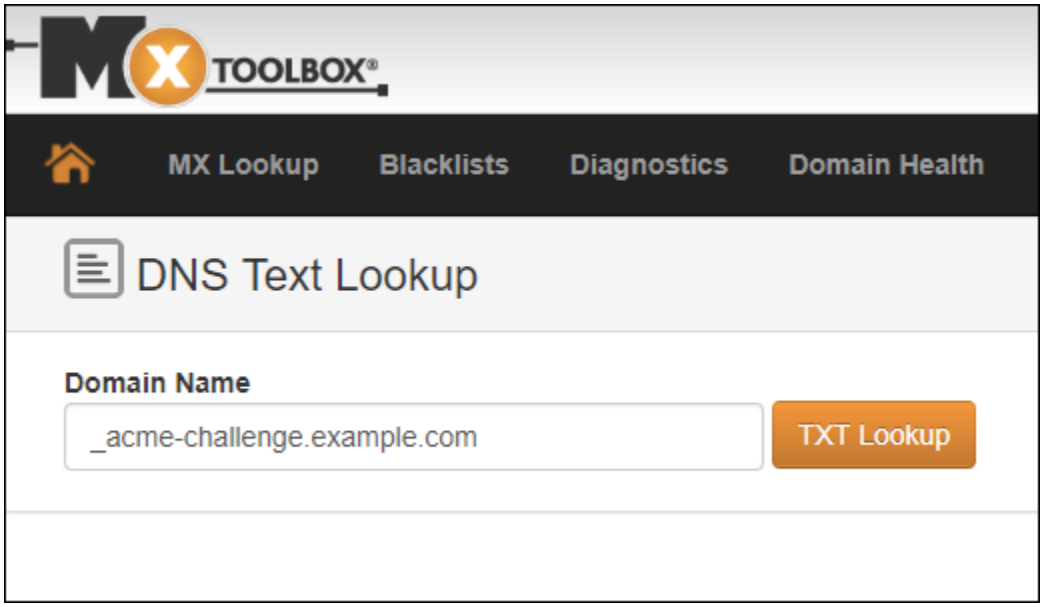
1. 新しいブラウザウィンドウを開き、https://mxtoolbox.com/TXTLookup.aspx に移動します。
2. 次の内容をテキストボックスに入力します。

   ```
   _acme-challenge.Domain
   ```

   Domain は、登録済みのドメイン名に置き換えます。

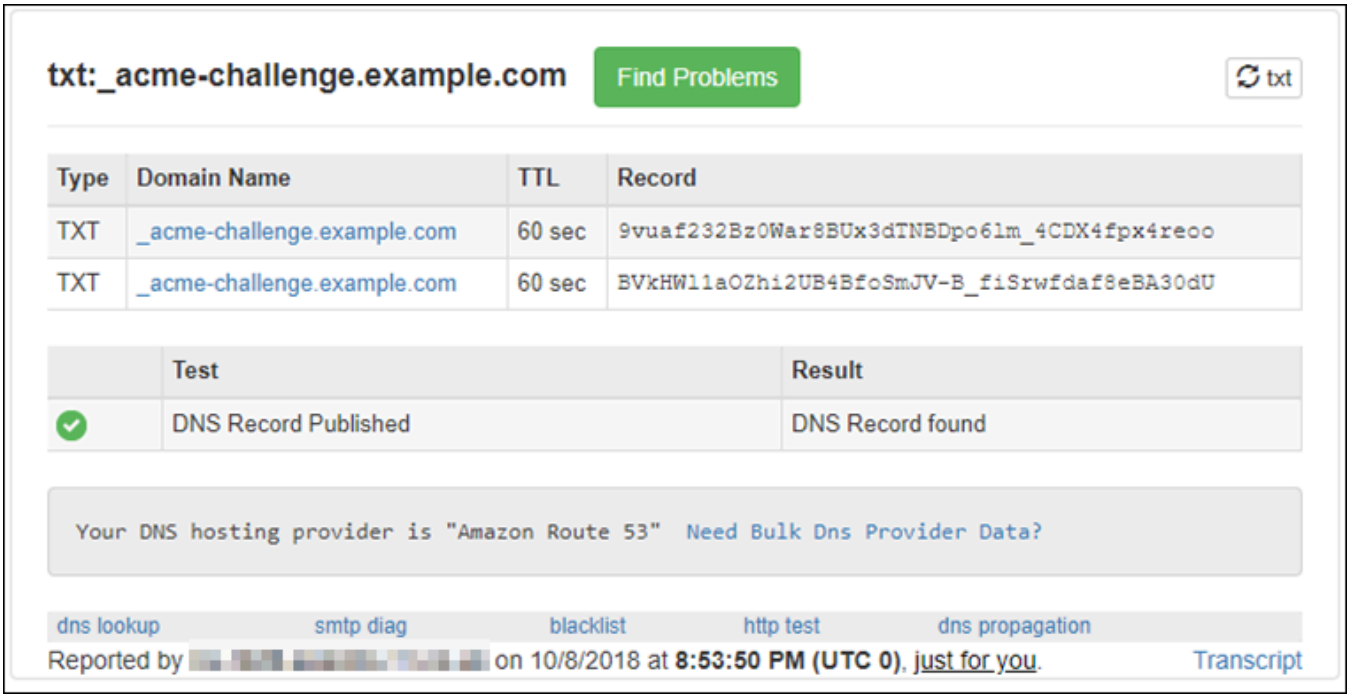
   例：

   ```
   _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。



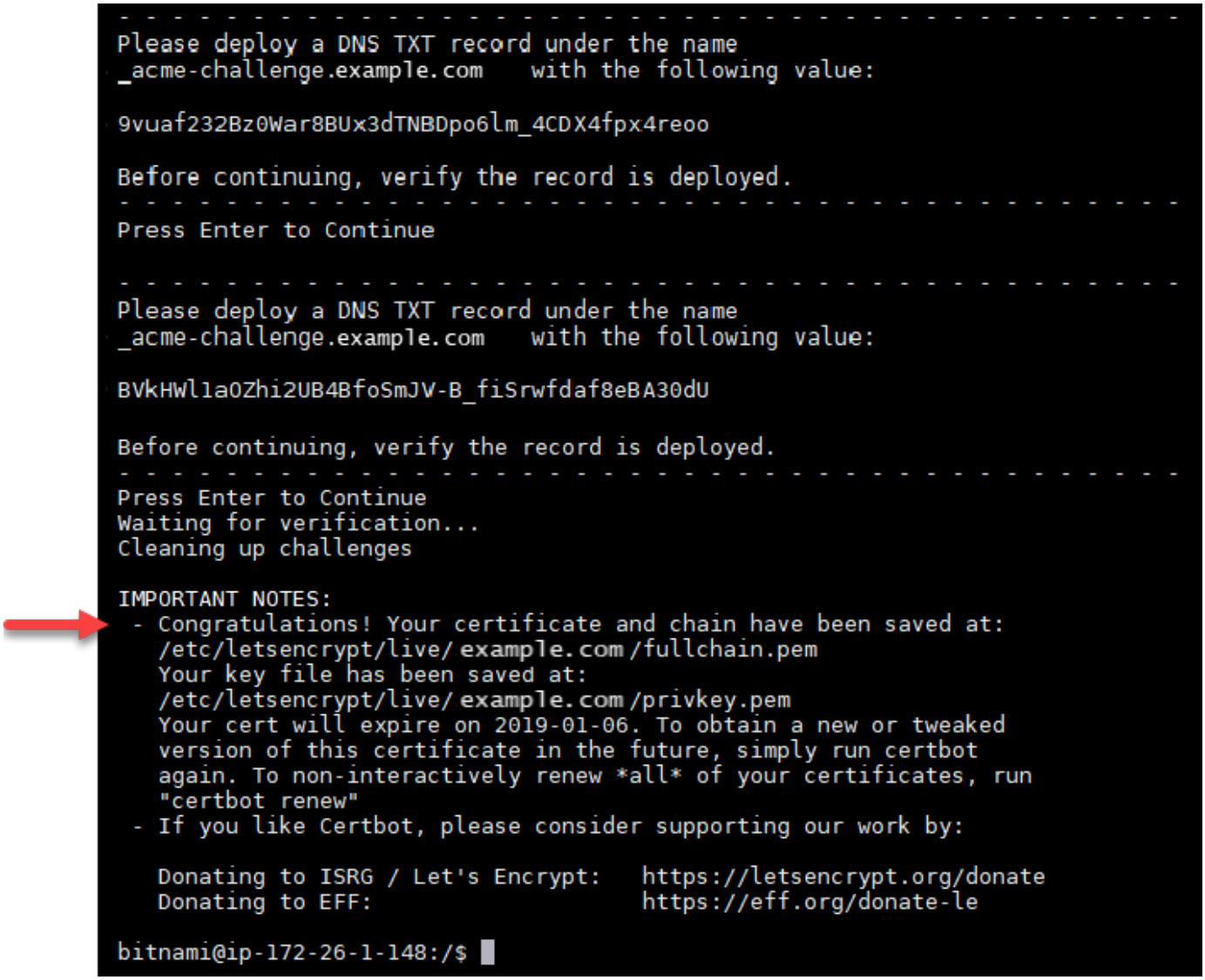
- TXT レコードがインターネットの DNS に反映されていない場合は、[DNS Record not found (DNS レコードが見つかりません)] というレスポンスが返されます。適切な DNS レコードをドメインの DNS ゾーンに追加したことを確認してください。適切なレコードを追加した場合は、ドメインの DNS レコードが反映されるまでしばらく待ってから、TXT のルックアップを再実行します。

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]* (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

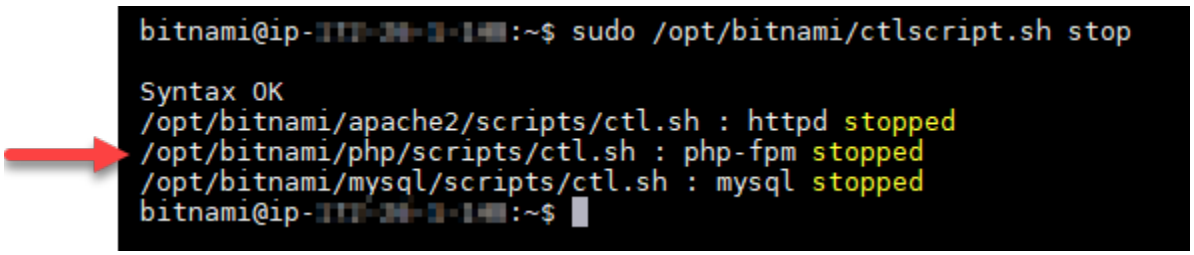
LAMP インスタンスの Apache サーバーディレクトリにある Let's Encrypt の SSL 証明書ファイルへのリンクを作成します。また、必要になる場合に備えて既存の証明書をバックアップします。

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：

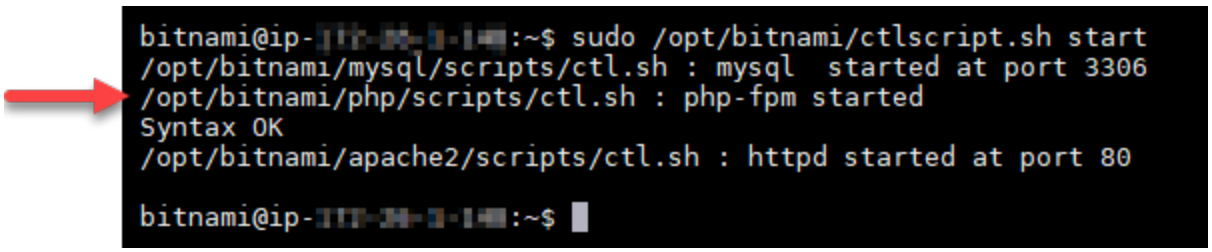
     ```
     sudo ln -s /etc/letsencrypt/live/$DOMAIN/privkey.pem /opt/bitnami/apache/conf/
     bitnami/certs/server.key
     ```

     ```
     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て、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。

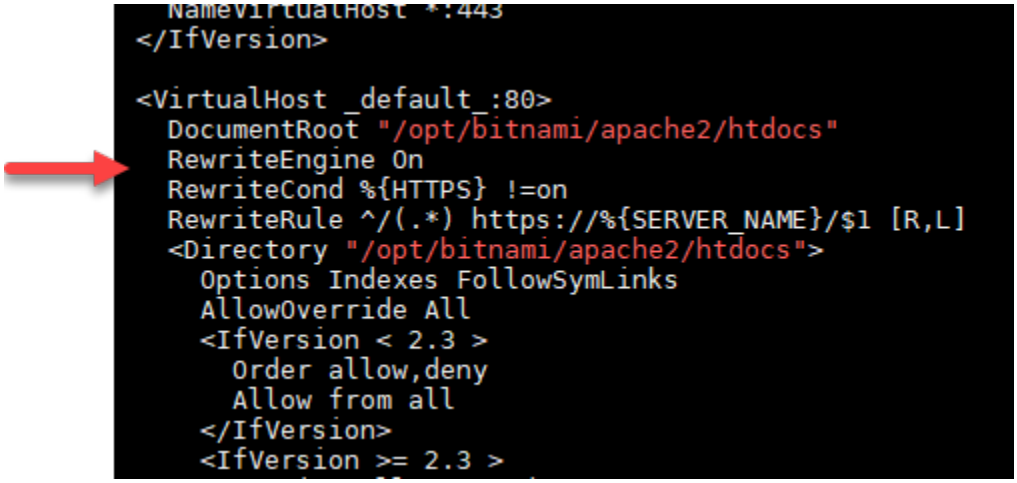
```
sudo vim /opt/bitnami/apache2/conf/bitnami/bitnami.conf
```

> **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。

Let's Encrypt SSL/TLS を使用して Lightsail Nginx ウェブサイトを保護する

Amazon Lightsail を使用すると、Lightsail ロードバランサーを使用してウェブサイトやアプリケーションを SSL/TLS で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、通常、適

切な選択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 インスタンスと統合できます。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 6: Let's Encrypt の SSL 証明書リクエストを完了する
- 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
- 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 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**
>
> 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ように PuTTY をダウンロードしてセットアップする Amazon Lightsail」を参照してください。

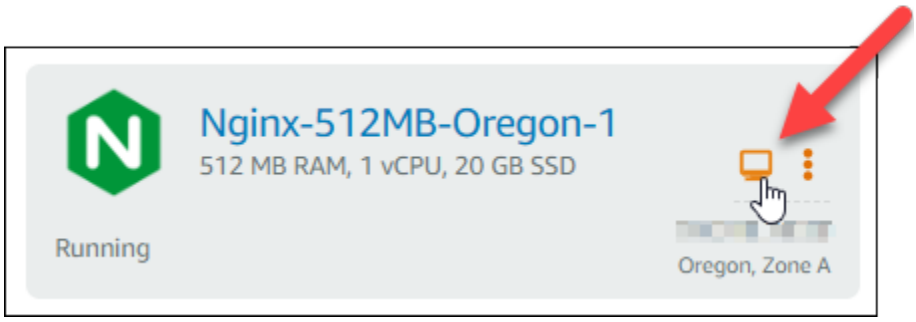
前提条件が完了したら、このチュートリアルの「次のセクション」に進みます。

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。

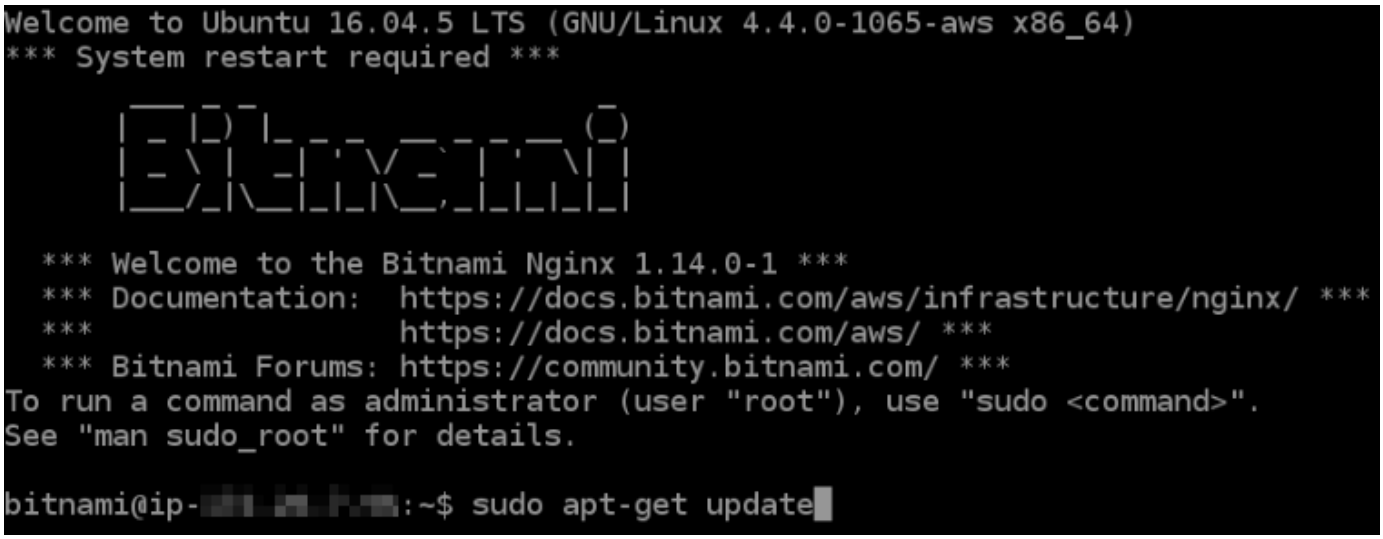
Lightsail インスタンスに Certbot をインストールするには

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 Certbot 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
```

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`* 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
```

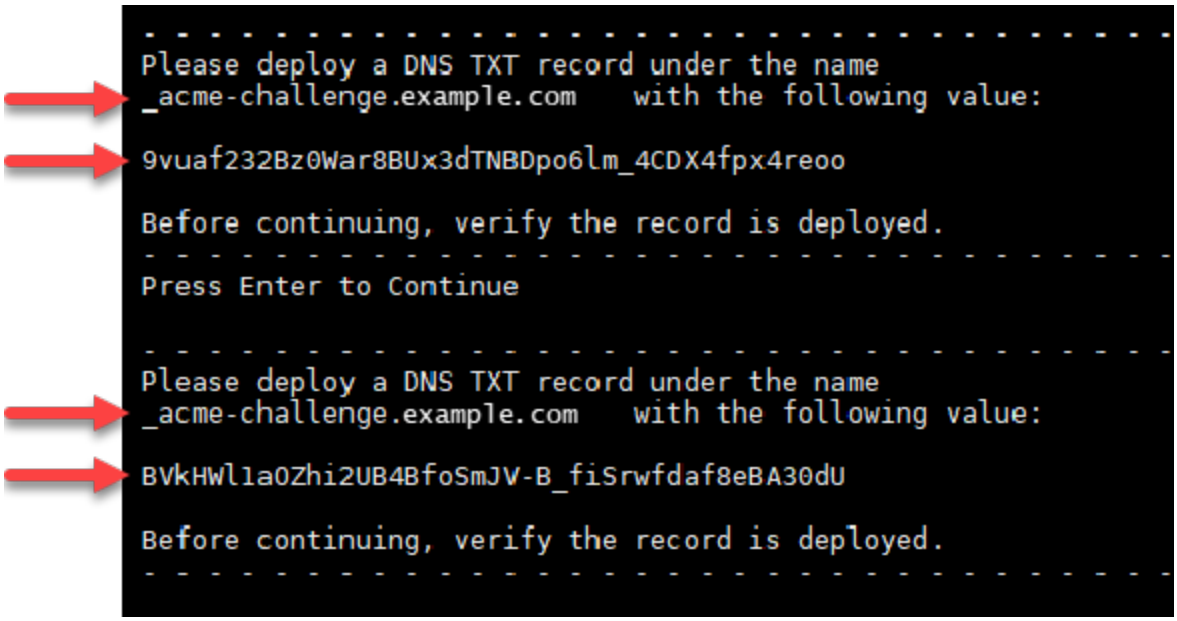
次のような結果が表示されます。

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 ブラウザベースの SSH セッションを開いたままにしておきます。このチュートリアルの後半に戻ります。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。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 DNS ゾーンの作成」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 & 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を開いたままにしておきます。このチュートリアルの後半に戻ります。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 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

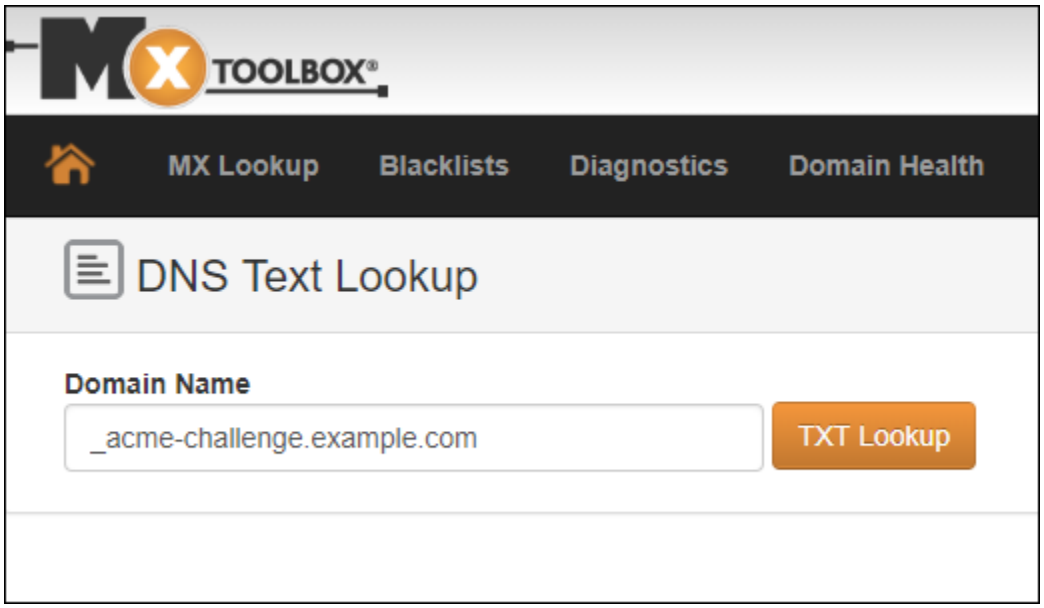
1. 新しいブラウザウィンドウを開き、https://mxtoolbox.com/TXTLookup.aspx に移動します。

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
```

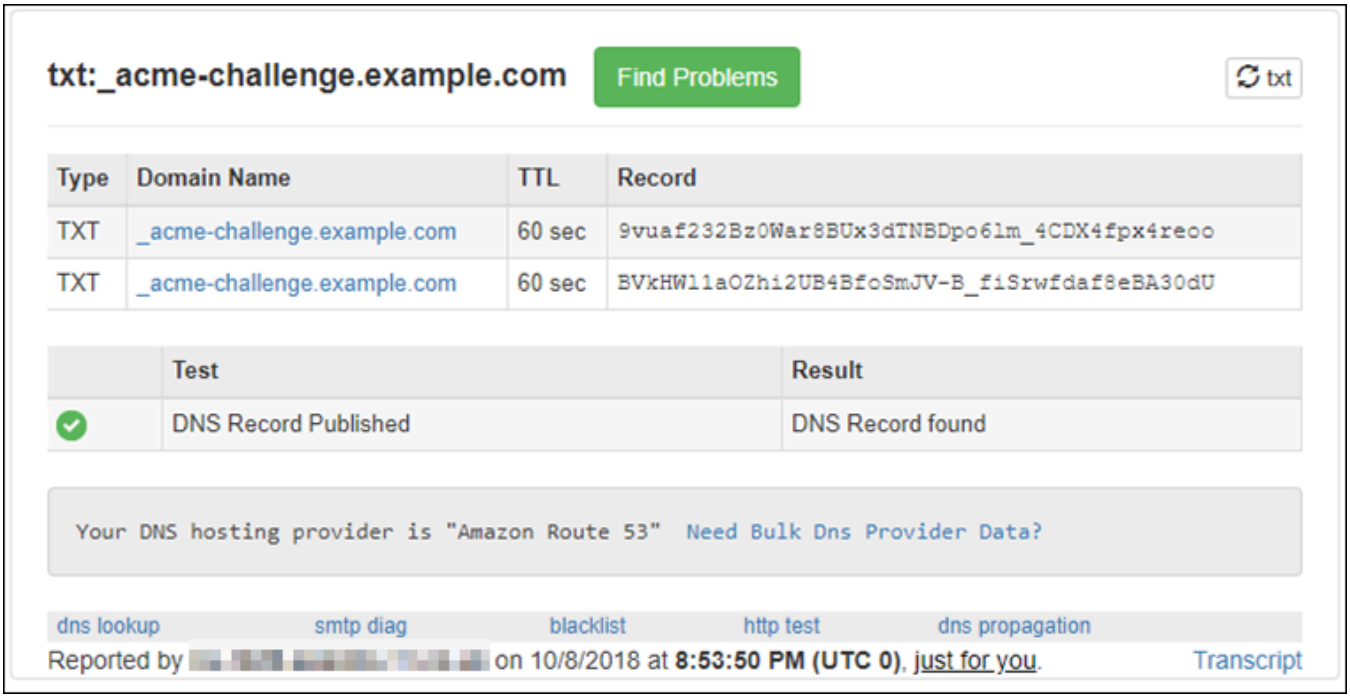
例：

```
_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。



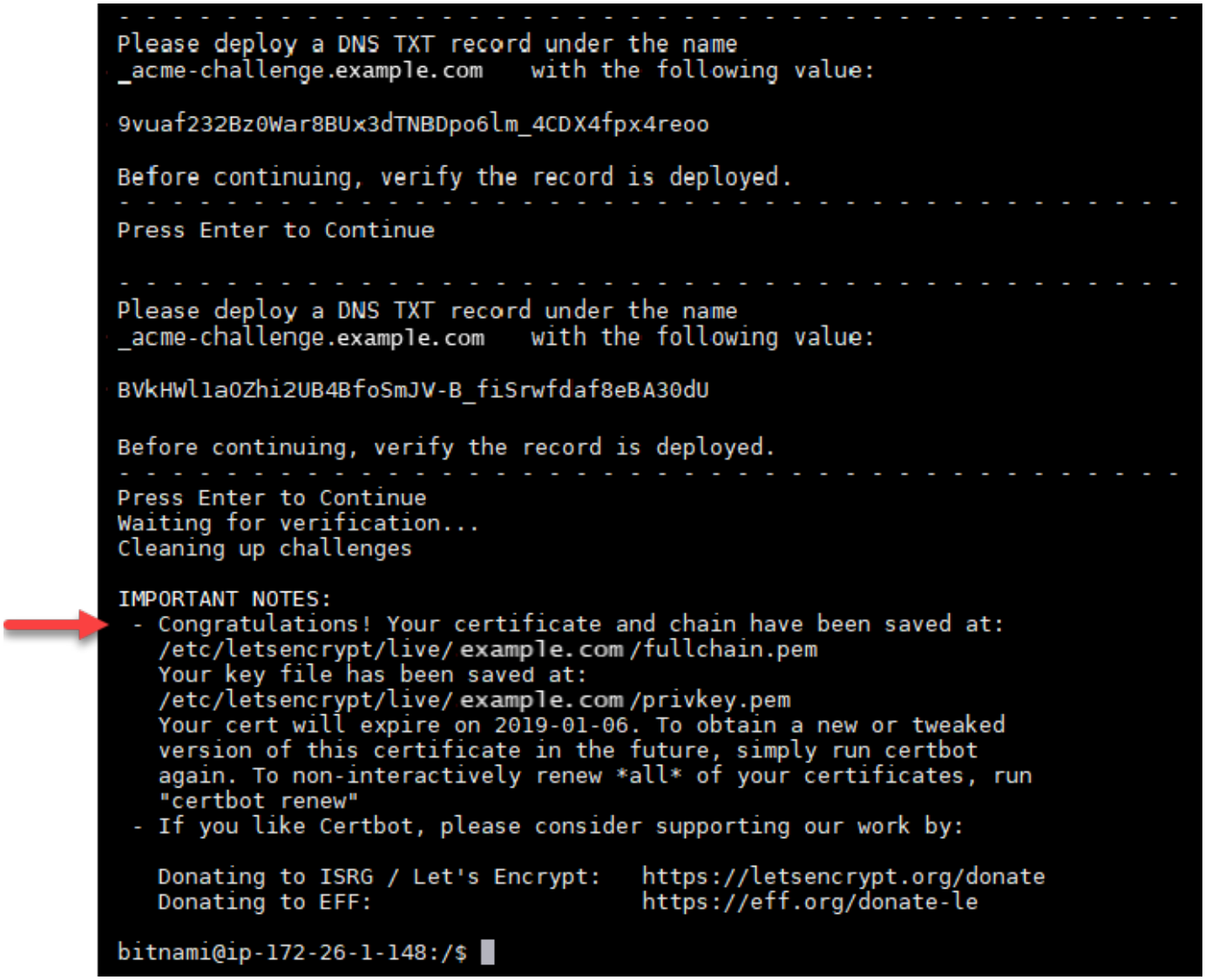
- TXT レコードがインターネットの DNS に反映されていない場合は、[DNS Record not found (DNS レコードが見つかりません)] というレスポンスが返されます。適切な DNS レコードをドメインの DNS ゾーンに追加したことを確認してください。適切なレコードを追加した場合は、ドメインの DNS レコードが反映されるまでしばらく待ってから、TXT のルックアップを再実行します。

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: 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

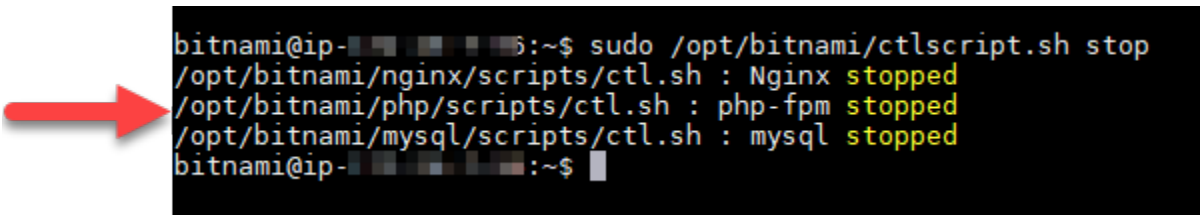
Let's Encrypt の SSL 証明書ファイルへのリンクを、Nginx インスタンスの Nginx サーバーディレクトリに作成します。また、必要になる場合に備えて既存の証明書をバックアップします。

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* 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：

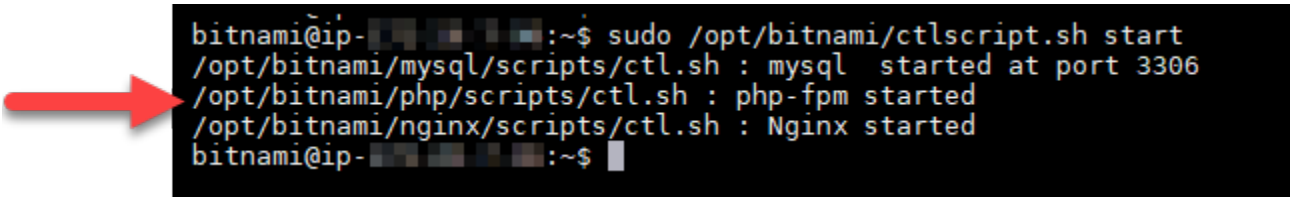
     ```
     sudo ln -s /etc/letsencrypt/live/$DOMAIN/privkey.pem /opt/bitnami/nginx/conf/
     bitnami/certs/server.key
     ```

     ```
     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て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
```

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

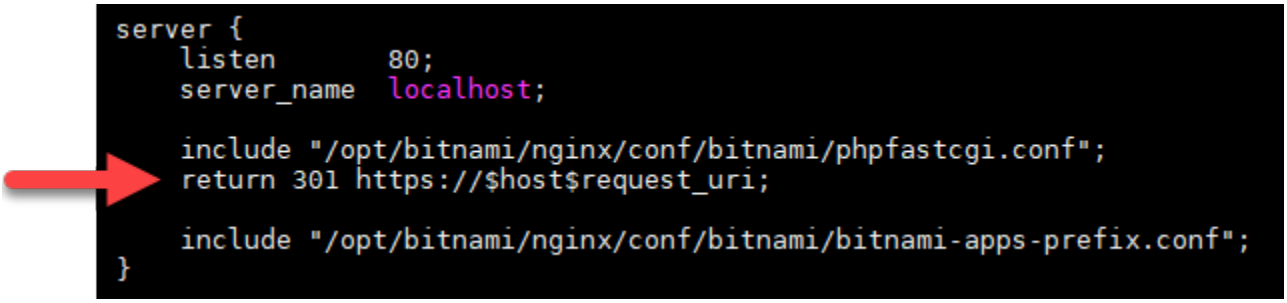
1. Nginx インスタンスの Lightsail ブラウザベースの SSH セッションで、次のコマンドを入力して、Vim テキストエディタを使用して Nginx ウェブサーバー設定ファイルを編集します。

   ```
   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。

無料の Let's Encrypt SSL 証明書で Lightsail WordPress インスタンスを保護する

> **Tip**
>
> Amazon Lightsail には、 WordPress 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。このチュートリアルの手動ステップに従う代わりに、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。

Lightsail を使用すると、Lightsail ロードバランサーを使用して SSL/TLS で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、通常、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し、Really Simple SSL プラグインを使用して WordPress インスタンスと統合する方法について説明します。

- 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 は、通常 Lightsail のインスタンスに WordPress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を提供します。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` ツールの使用の詳細については、「Amazon Lightsail でのインスタンスでの HTTPS の WordPress有効化 Amazon Lightsail」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する 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)。

- Lightsail で WordPress 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ように PuTTY をダウンロードしてセットアップする Amazon Lightsail」を参照してください。

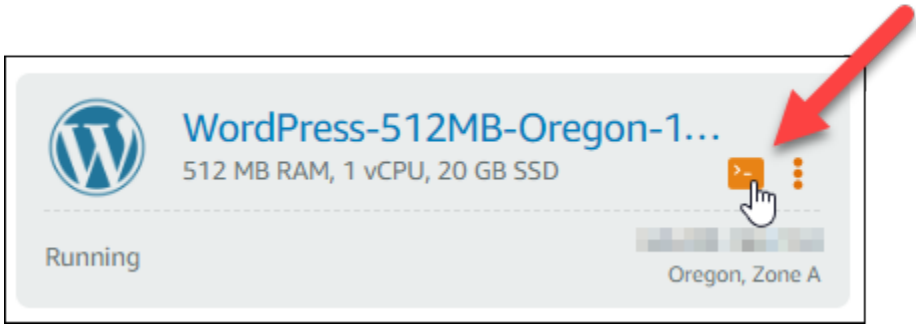
前提条件が完了したら、このチュートリアルの「次のセクション」に進みます。

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。

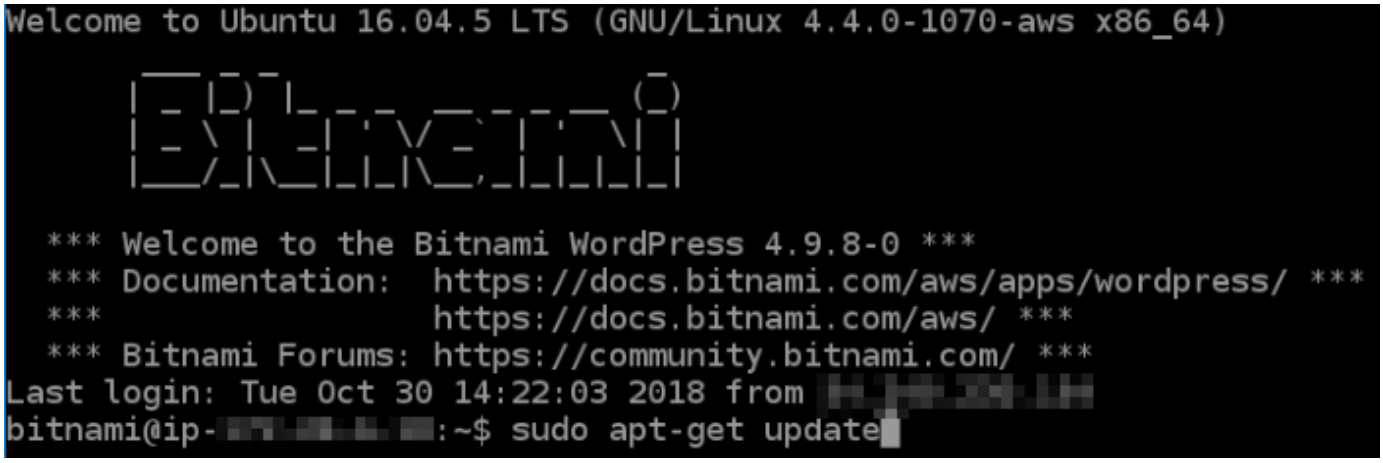
Lightsail インスタンスに Certbot をインストールするには

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 Cerbot をインストールします。

```
sudo apt-get install certbot -y
```

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。

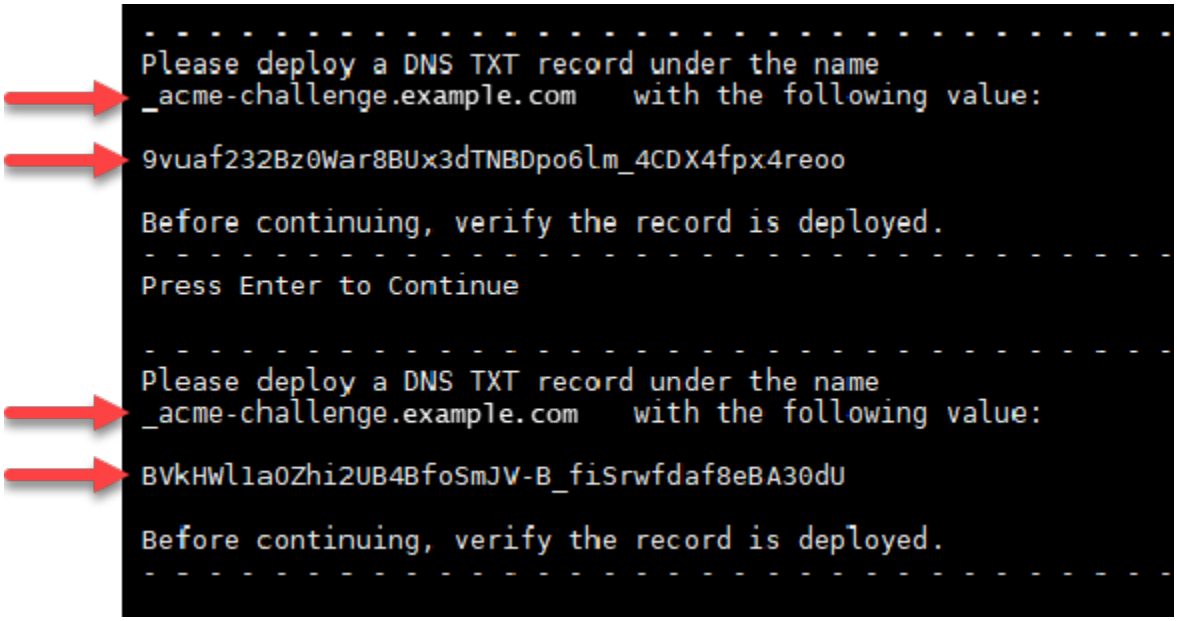
3. 次のコマンドを入力して Certbot をインタラクティブモードで起動します。このコマンドでは、DNS チャレンジで手動認証を使用してドメインの所有権を検証することを Certbot に指示します。また、最上位ドメインとそのサブドメイン用にワイルドカード証明書をリクエストします。

   ```
   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 ブラウザベースの SSH セッションを開いたままにしておきます。このチュートリアルの後半に戻ります。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。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 DNS ゾーンの作成」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 & 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を開いたままにしておきます。このチュートリアルの後半に戻ります。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 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。

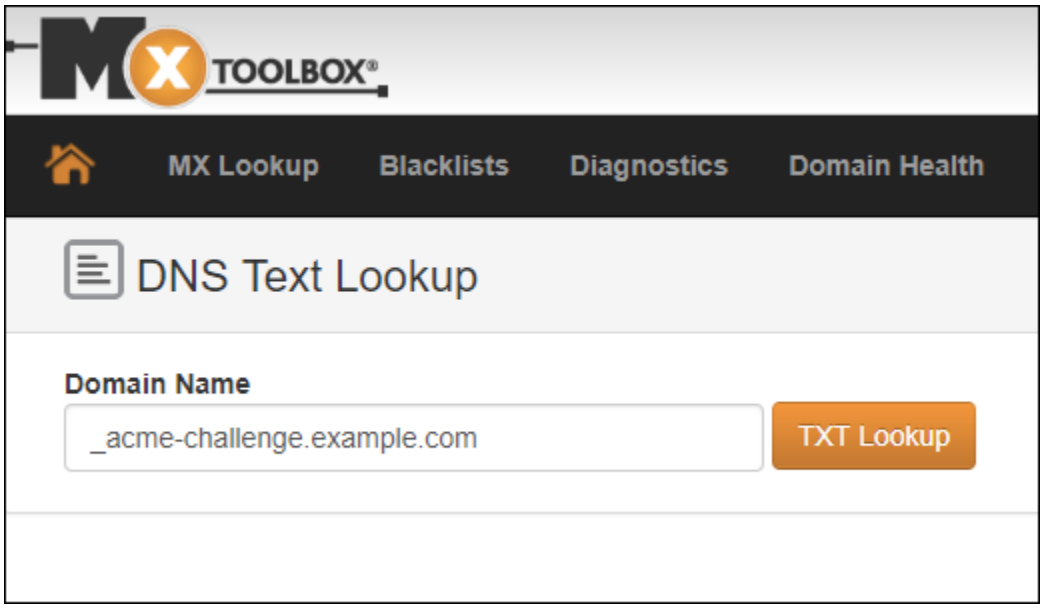
TXT レコードがインターネットの DNS に反映されたことを確認するには

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
   ```

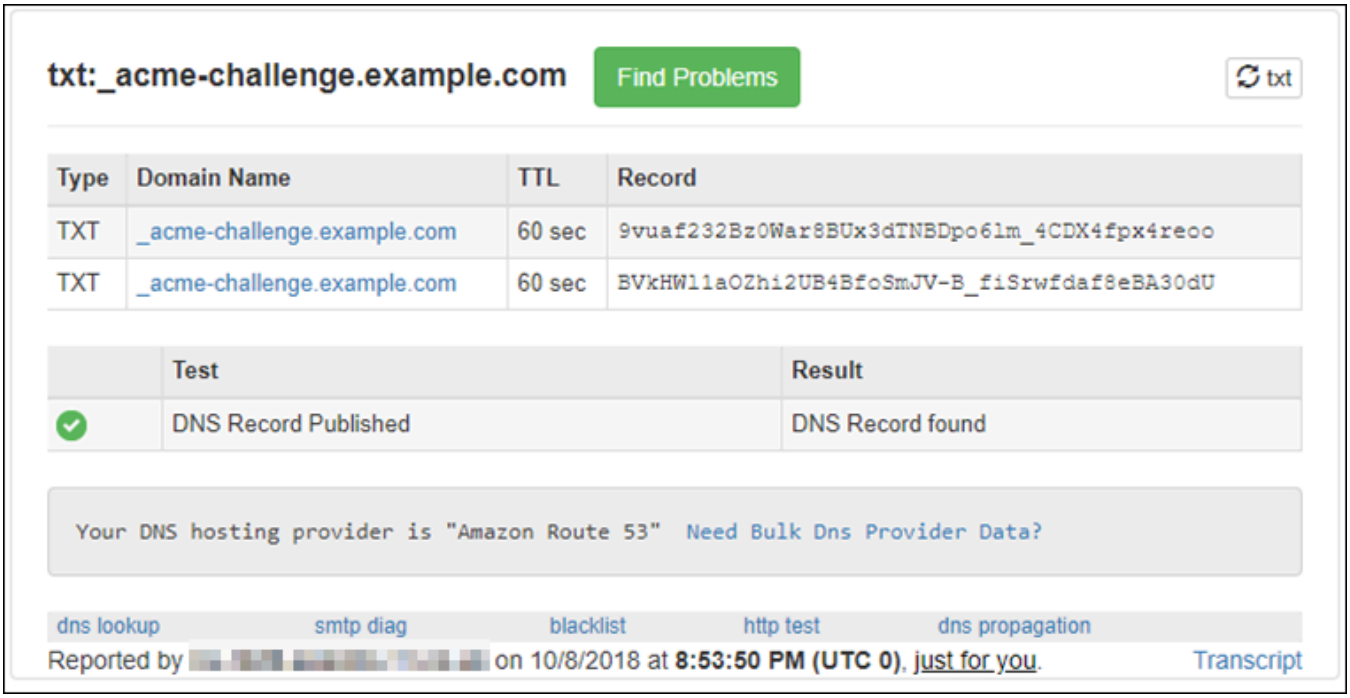
   例：

   ```
   _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。



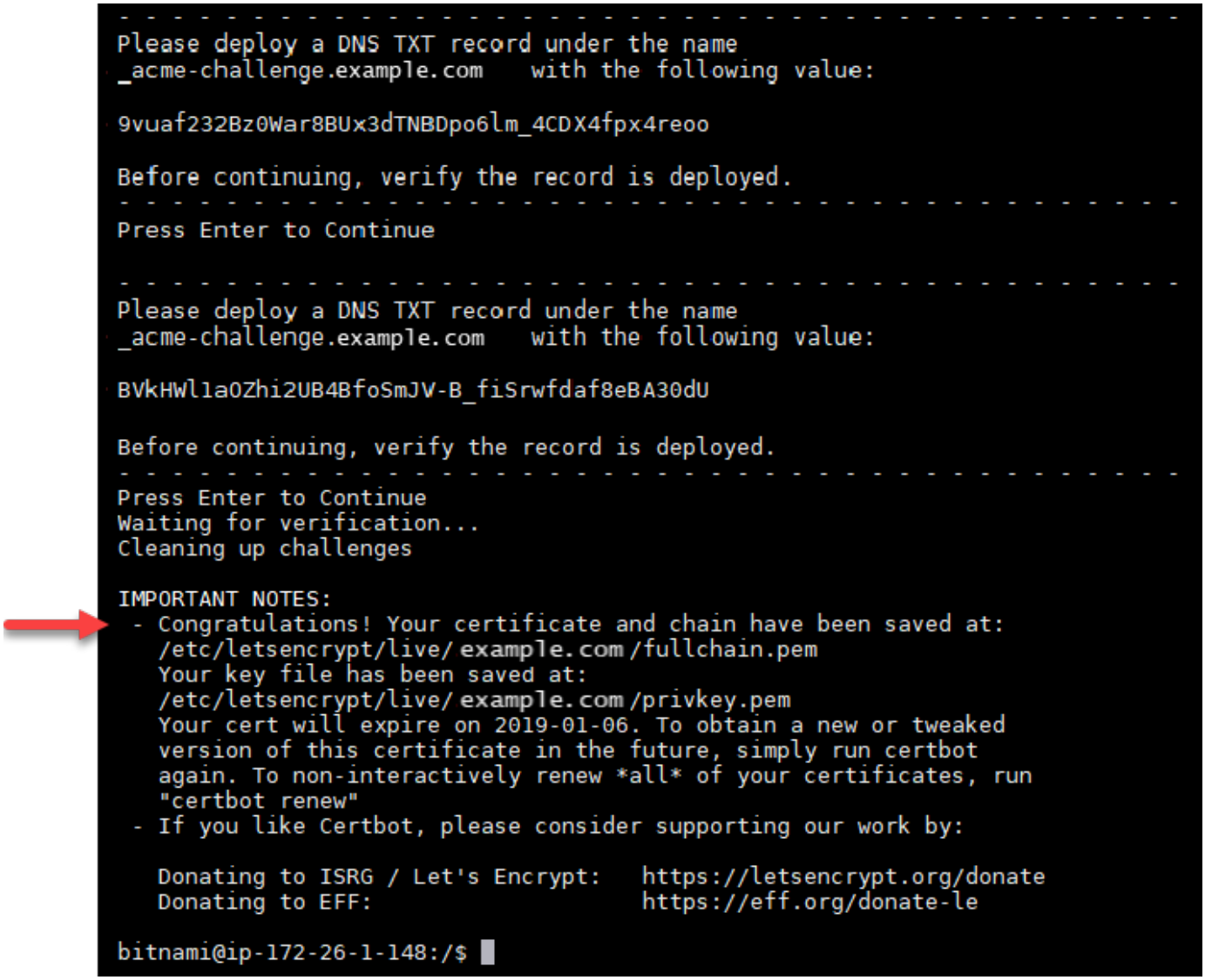
- TXT レコードがインターネットの DNS に反映されていない場合は、[DNS Record not found (DNS レコードが見つかりません)] というレスポンスが返されます。適切な DNS レコードをドメインの DNS ゾーンに追加したことを確認してください。適切なレコードを追加した場合は、ドメインの DNS レコードが反映されるまでしばらく待ってから、TXT のルックアップを再実行します。

ステップ 6: Let's Encrypt の SSL 証明書リクエストを完了する

WordPress 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 WordPress インスタンス上の特定のディレクトリに保存します。

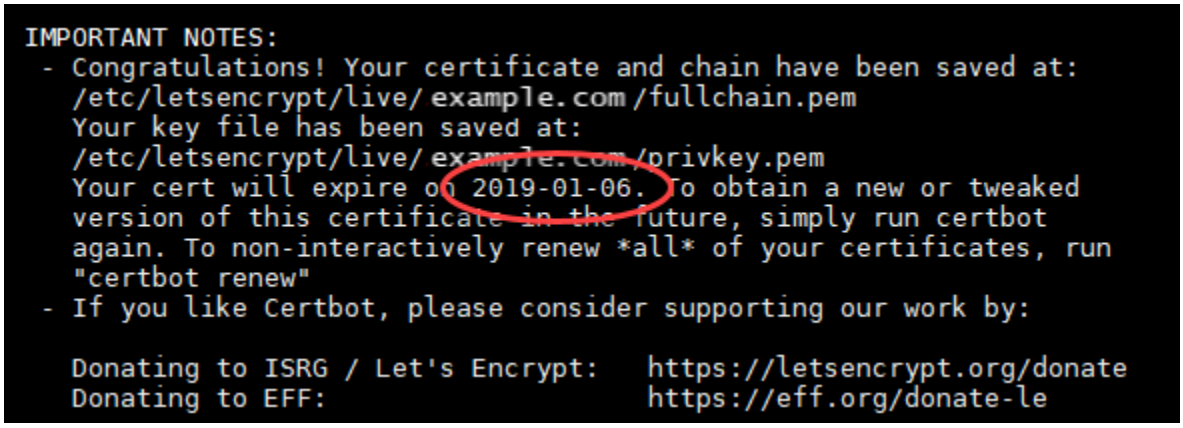
Let's Encrypt の SSL 証明書リクエストを完了するには

1. WordPress 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。



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

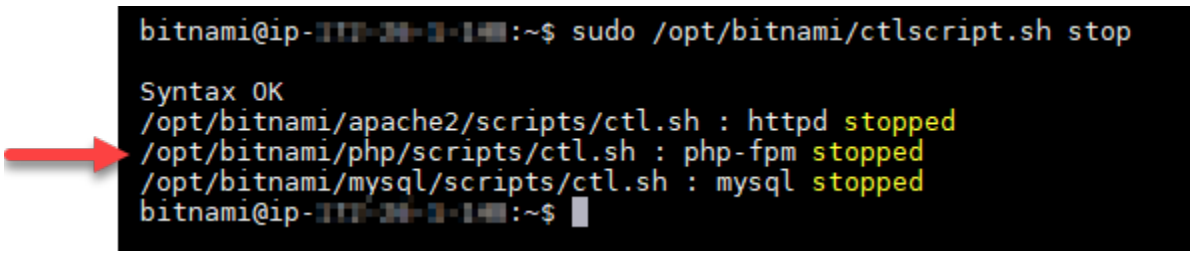
WordPress インスタンスの Apache サーバーディレクトリに Let's Encrypt SSL 証明書ファイルへのリンクを作成します。また、必要になる場合に備えて既存の証明書をバックアップします。

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. WordPress 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:

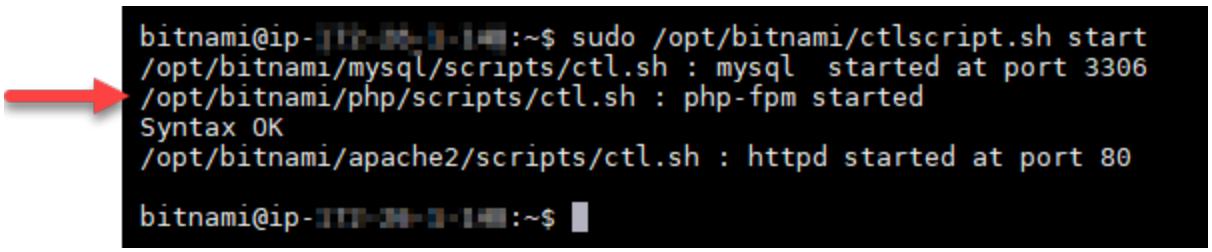
     ```
     sudo ln -s /etc/letsencrypt/live/$DOMAIN/privkey.pem /opt/bitnami/apache/conf/
     bitnami/certs/server.key
     ```

```
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 証明書ファイルは正しいディレクトリに保存されます。

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 証明書を WordPress サイトに統合する

Really Simple SSL プラグインを WordPress 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 証明書を WordPress サイトに統合するには

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、次のコマンドを入力して、 `wp-config.php`および `htaccess.conf` 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
```

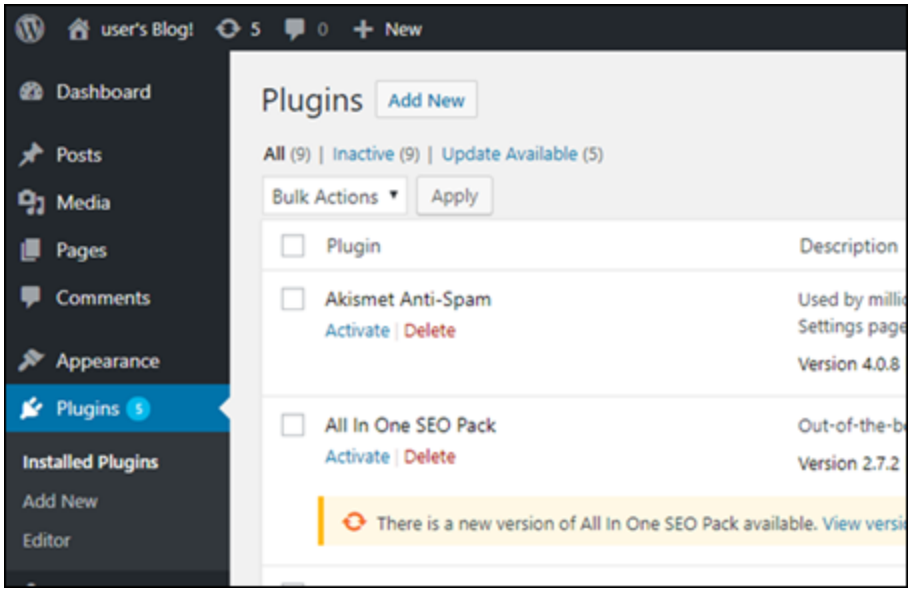
- Ubuntu Linux ディストリビューションを使用する古いインスタンスの場合：

```
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、 WordPress 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、Amazon Lightsail」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。インスタンスの管理ダッシュボードの WordPress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 、インスタンスが SSL 暗号化を使用するように設定されました。さらに、WordPress インスタンスが HTTP から HTTPS に自動的に接続をリダイレクトするように設定されました。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。

特定のインスタンスタイプの step-by-step 手順に従ってください。各トピックでは、インスタンスの Linux ディストリビューション (Ubuntu または Debian) および Bitnami インストールタイプ (システムパッケージまたは自己完結型) に合わせた詳細なコマンドと設定手順について説明します。このトピックに従うことで、Let's Encrypt の無料 SSL/TLS 証明書で Lightsail のウェブサイトとアプリケーションを保護し、暗号化された通信を確保し、訪問者のセキュリティを強化できます。

Lightsail インスタンスのIPv6ネットワークを設定する

このセクションでは、Lightsail インスタンスブループリントIPv6での の設定に関連する以下のトピックについて説明します。

トピック

Lightsail で cPanel インスタンスIPv6の接続を設定する

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、インスタンスにパブリックIPv6アドレスを割り当てるために を有効にIPv6できます。詳細については、Amazon Lightsail IP アドレス」および「 を有効または無効にするIPv6」を参照してください。

cPanel & WHMブループリントを使用するインスタンスIPv6に対して を有効にした後、追加の一連のステップを実行して、インスタンスにその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、 cPanel インスタンスと WHMインスタンスに対して実行する必要がある追加のステップを示します。

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 および 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 および WHMインスタンスを設定します。詳細については、Amazon Lightsail WHMの「クイックスタートガイド： cPanel &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。

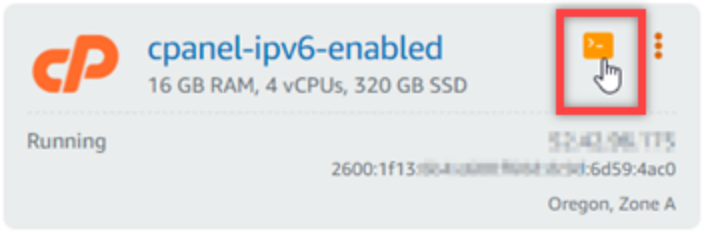
- インスタンス cPanel と WHMインスタンスIPv6に対して を有効にします。詳細については、「 を有効または無効にするIPv6」を参照してください。

> **Note**
>
> 2021 年 1 月 12 日以降に作成された新しい cPanel および WHMインスタンスは、Lightsail コンソールで作成されたときにデフォルトでIPv6有効になっています。インスタンスの作成時に がデフォルトで有効IPv6になっている場合でも、インスタンスIPv6で を設定するには、このガイドの次のステップを完了する必要があります。

cPanel および WHMインスタンスIPv6で を設定する

Lightsail の cPanel および WHMインスタンスIPv6で 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 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インスタンスを見つけ、ブラウザベースのSSHクライアントアイコンを選択して を使用して接続しますSSH。



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
```

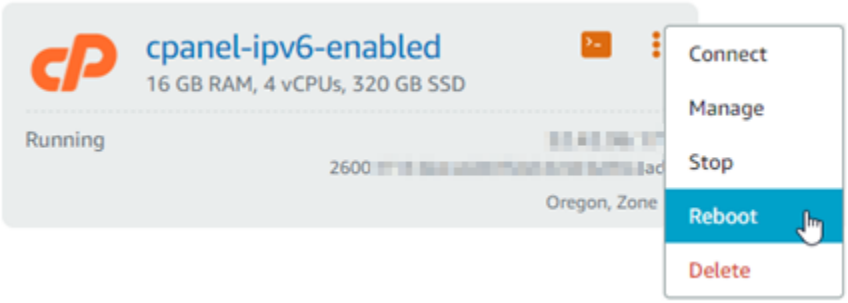
4. ファイルにテキストが追加されていない場合、次のテキストを追加します。

```
IPV6INIT=yes
IPV6_AUTOCONF=yes
DHCPV6C=yes
```

結果は次の例のようになります。

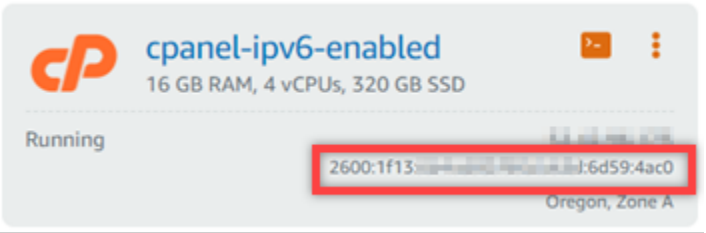
```
# Automatically generated by the vm import process
TYPE=Ethernet
PROXY_METHOD=none
BROWSER_ONLY=no
BOOTPROTO=dhcp
DEFROUTE=yes
IPV4_FAILURE_FATAL=no
NAME=eth0
DEVICE=eth0
ONBOOT=yes
IPV6INIT=yes
IPV6_FAILURE_FATAL=no
DHCPV6C=yes
IPV6_AUTOCONF=yes
```

5. キーボードの 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ウィンドウを閉じて、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 および 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。



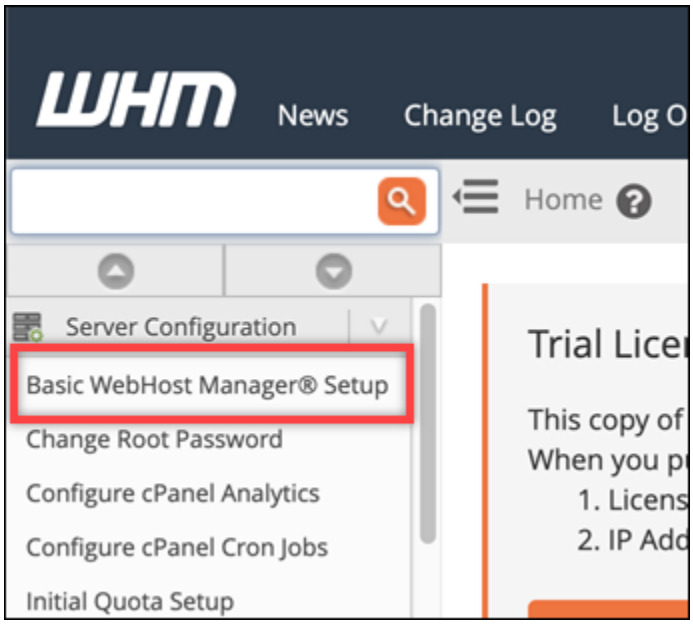
インスタンスが再起動するまで数分待ってから、次のステップに進みます。

9. Lightsail ホームページのインスタンスタブで、 cPanel および WHMインスタンスに割り当てられたIPv6アドレスを書き留めます。

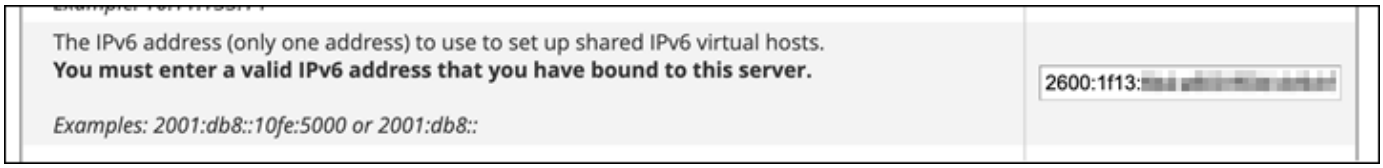


10. 新しいブラウザタブを開き、 cPanel および WHMインスタンスの Web Host Manager (WHM) にサインインします。

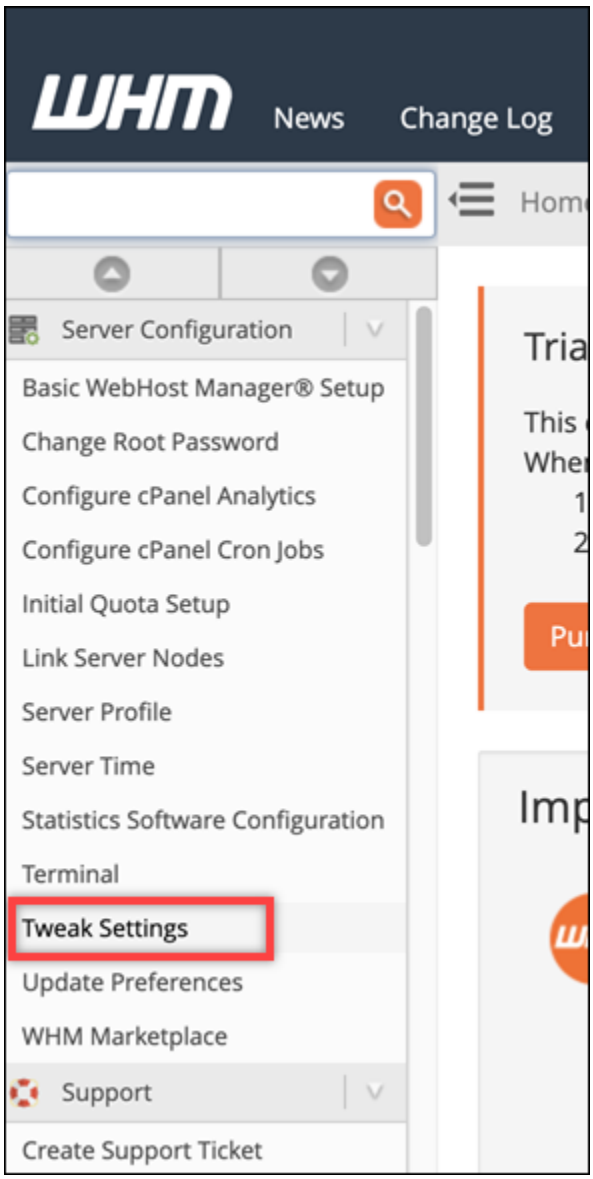
11. WHM コンソールの左側のナビゲーションペインで、ベーシック WebHost マネージャーセットアップ を選択します。



12. すべて タブで、 IPv6 を使用するアドレスのテキストを検索し、インスタンスに割り当てられたIPv6アドレスを入力します。この手順のステップ 9 で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。

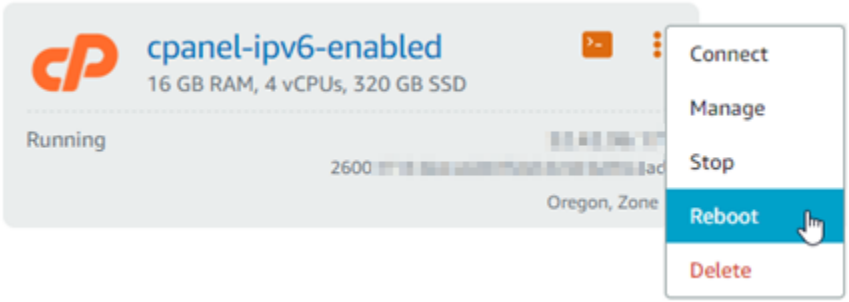


13. ページの最下部までスクロールし、[変更の保存] を選択します。
14. WHM コンソールの左側のナビゲーションペインで、設定の微調整 を選択します。



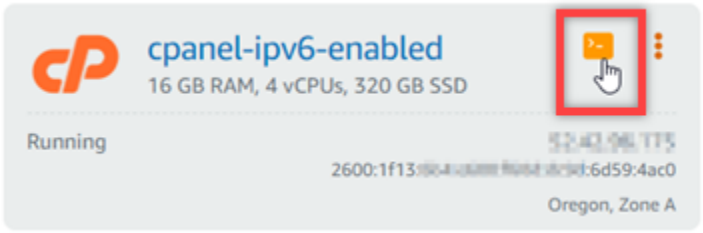
15. All タブで、下にスクロールして Listen on IPv6 Addresses 設定を見つけ、On に設定します。

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 および 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. を使用して接続する cPanel & 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SSH。



20. インスタンスに接続したら、次のコマンドを入力して、インスタンスに設定されている IP アドレスを表示し、割り当てられたIPv6アドレスが認識され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、この例に示すように、スコープグローバルのラベルが付いたレスポンスにそのアドレスが表示されます。

```
[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. 次のコマンドを入力して、インスタンスがIPv6アドレスに ping を実行できることを確認します。

```
ping6 ipv6.google.com -c 6
```

結果は次の例のようになります。これにより、インスタンスがIPv6アドレスに ping を実行できることを確認できます。

```
[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 で Debian 8 インスタンスIPv6の接続を設定する

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、インスタンスにパブリックIPv6アドレスを割り当てるために を有効にIPv6できます。詳細については、Amazon Lightsail IP アドレス」および「 を有効または無効にするIPv6」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IPv6に対して を有効にした後、追加の一連のステップを実行して、インスタンスにその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、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)。

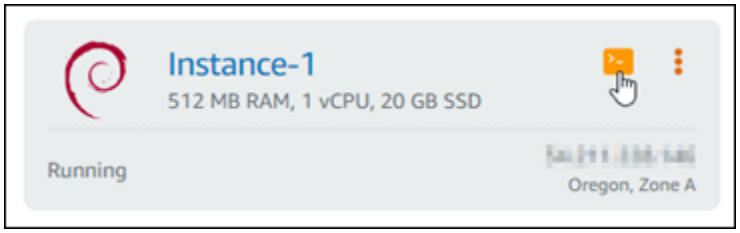
- Lightsail で Debian 8 インスタンスを作成します。詳細については、「インスタンスを作成する」を参照してください。
- Debian 8 インスタンスIPv6で を有効にします。詳細については、「 を有効または無効にするIPv6」を参照してください。

> **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たときにデフォルトでIPv6有効になっています。インスタンスの作成時に がデフォルトで有効になっている場合でも、インスタンスIPv6で を設定するにはIPv6、このガイドの次のステップを完了する必要があります。

Debian 8 インスタンスIPv6で を設定する

Lightsail の Debian 8 インスタンスIPv6で 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 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 インスタンスを見つけ、ブラウザベースのSSHクライアントアイコンを選択して を使用して接続しますSSH。



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。

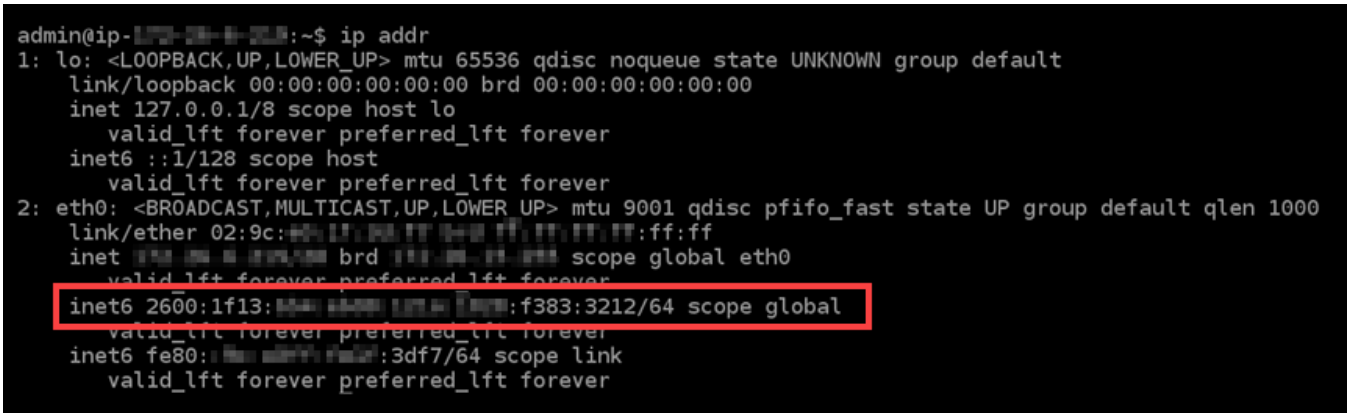
```
ip addr
```

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、レスポンスにはリスト表示されません。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、この例scope globalに示すように、レスポンスに と表示されます。ここで停止する必要があります。インスタンスはIPv6アドレスを認識するように既に設定されているため、この手順のステップ 4 ~ 9 を完了する必要はありません。



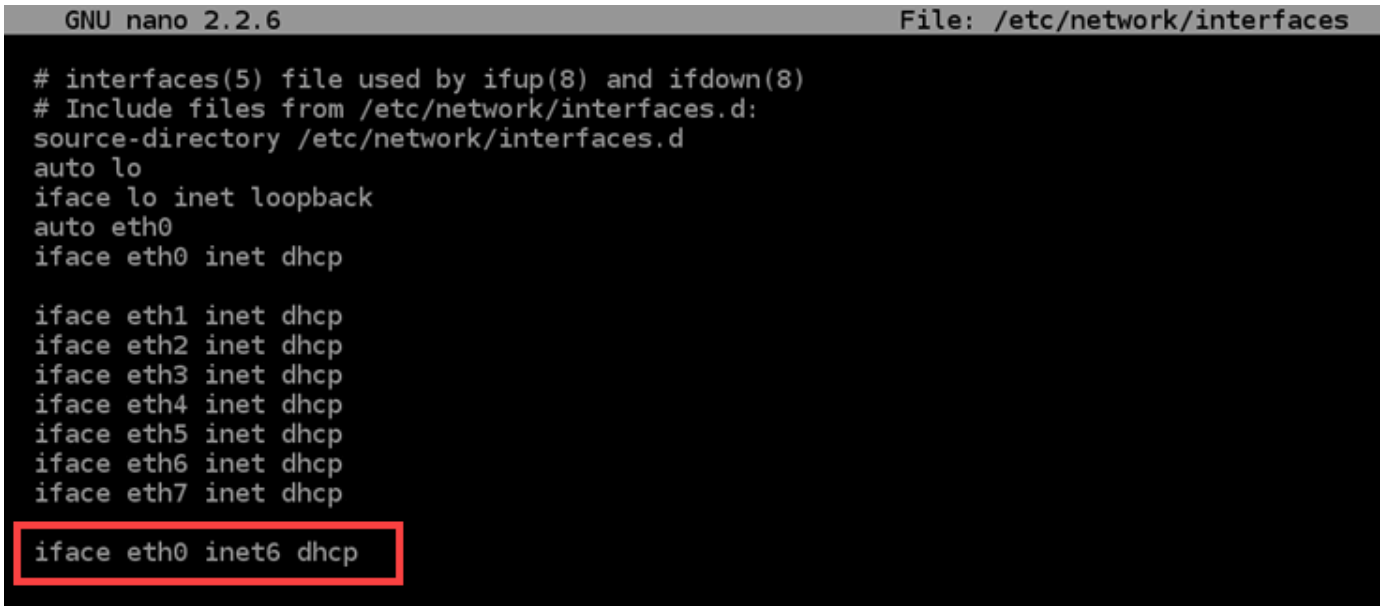
4. 以下のコマンドを入力して、nano を使用してinterfaces設定ファイルを開きます。

```
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。

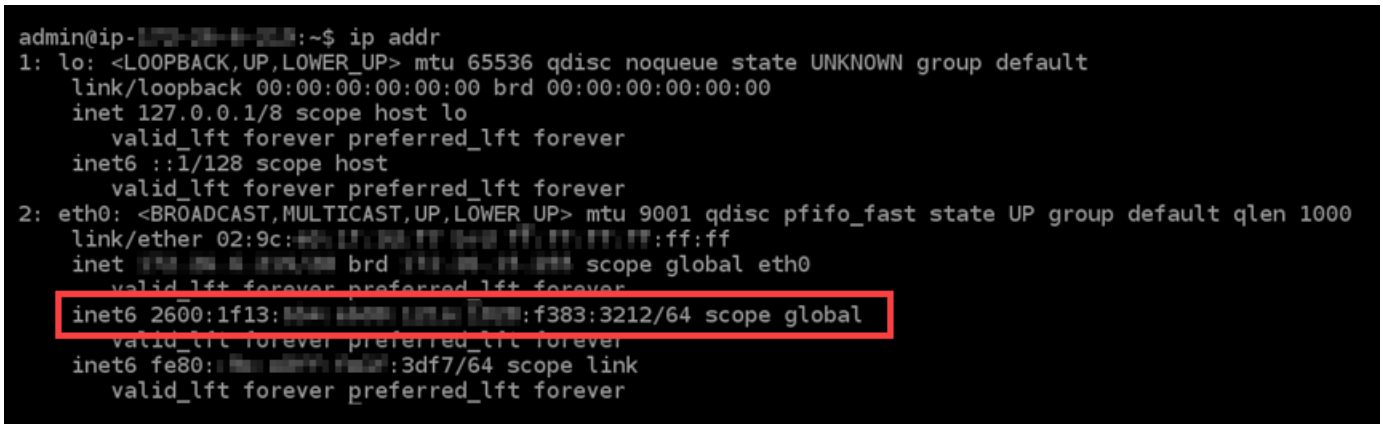
```
sudo systemctl restart networking
```

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アドレスを認識できるように、さらに数分かかる場合があります。

9. 次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アドレスが認識され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、この例scope globalに示すように、 のラベルが付いたレスポンスにそのアドレスが表示されます。



Lightsail で GitLab インスタンスIPv6の接続を設定する

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、インスタンスにパブリックIPv6アドレスを割り当てるために を有効にIPv6できます。詳細については、Amazon Lightsail IP アドレス」および「 を有効または無効にするIPv6」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IPv6で を有効にした後、追加の一連のステップを実行して、インスタンスにIPv6その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ステップ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)。

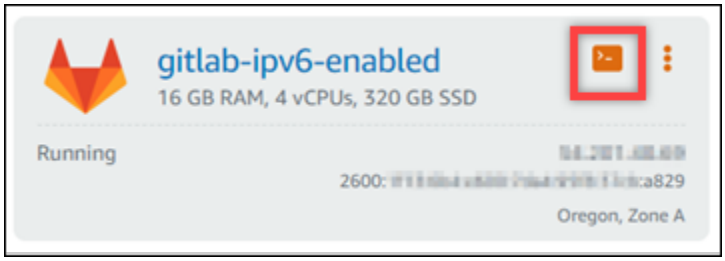
- Lightsail で GitLab インスタンスを作成します。詳細については、「インスタンスを作成する」を参照してください。
- GitLab インスタンスIPv6で を有効にします。詳細については、「 を有効または無効にするIPv6」を参照してください。

> **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 コンソールで作成されたときにデフォルトでIPv6有効になっています。インスタンスの作成時に がデフォルトで有効になっている場合でも、インスタンスIPv6で を設定するにはIPv6、このガイドの次のステップを完了する必要があります。

GitLab インスタンスIPv6で を設定する

Lightsail の GitLab インスタンスIPv6で 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 ホームページの「インスタンス」セクションで、設定する GitLab インスタンスを見つけ、ブラウザベースのSSHクライアントアイコンを選択して を使用して接続しますSSH。



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。

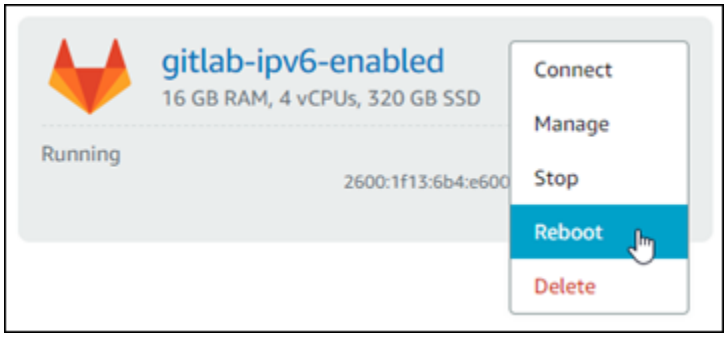
- インスタンスがIPv6アドレスを認識しない場合、レスポンスにはリスト表示されません。この手順のステップ 4 ~ 9 を続行する必要があります。

```
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、この例scope globalに示すように、レスポンスに と表示されます。ここで停止する必要があります。インスタンスはIPv6アドレスを認識するように既に設定されているため、この手順のステップ 4 ~ 9 を完了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、再起動 を選択します。

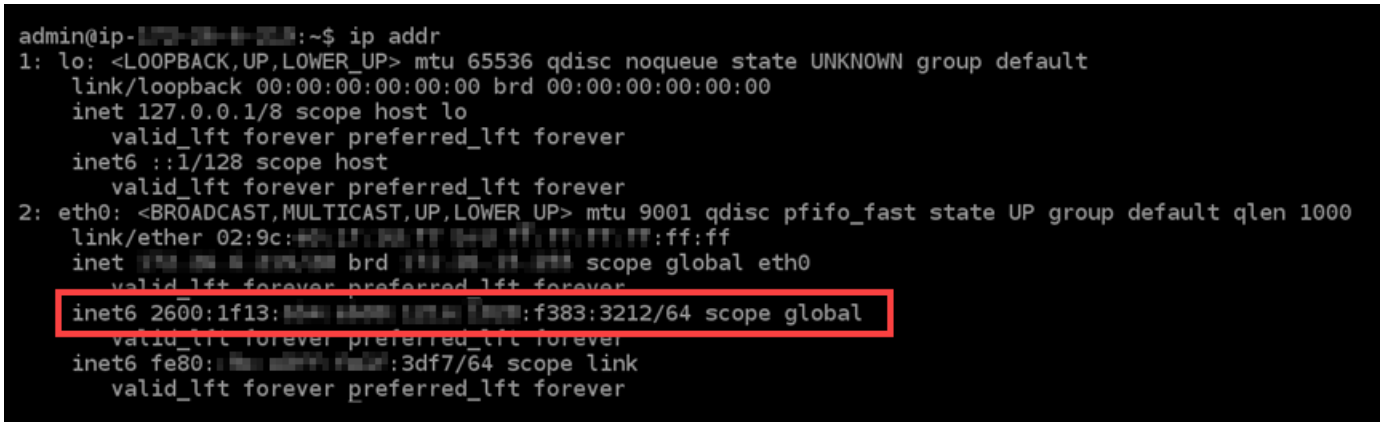


インスタンスが再起動するまで数分待ってから、次のステップに進みます。

6. GitLab インスタンスのSSHセッションに戻ります。
7. 次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アドレスが認識され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、この例scope globalに示すように、 のラベルが付いたレスポンスにそのアドレスが表示されます。



Lightsail で Nginx インスタンスIPv6の接続を設定する

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、インスタンスにパブリックIPv6アドレスを割り当てるために を有効にIPv6できます。詳細については、Amazon Lightsail IP アドレス」および「 を有効または無効にするIPv6」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IPv6に対して を有効にした後、追加の一連のステップを実行して、インスタンスにその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、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。

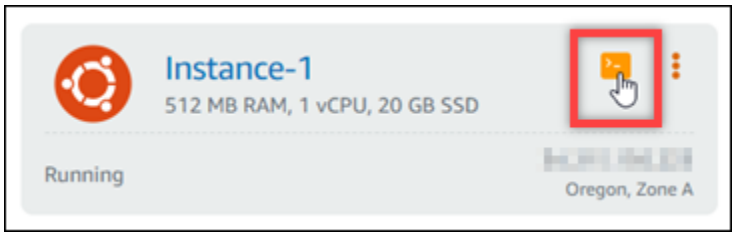
- Nginx インスタンスIPv6で を有効にします。詳細については、「 を有効または無効にするIPv6」を参照してください。

> **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たときにデフォルトでIPv6有効になっています。インスタンスの作成時に がデフォルトで有効IPv6になっている場合でも、インスタンスIPv6で を設定するには、このガイドの次のステップを完了する必要があります。

Nginx インスタンスIPv6で を設定する

Lightsail の Nginx インスタンスIPv6で 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 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 インスタンスを見つけ、ブラウザベースのSSHクライアントアイコンを選択して を使用して接続しますSSH。



3. インスタンスに接続したら、次のコマンドを入力して、インスタンスがポート 80 経由でIPv6リクエストをリッスンしているかどうかを確認します。必ず を置き換えてください。*<IPv6Address>* インスタンスに割り当てられたIPv6アドレス。

```
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 経由で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 経由でIPv6リクエストをリッスンしている場合、次の例に示すように、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。ここで停止する必要があります。インスタンスが の にすでに設定されているため、この手順のステップ 4 ~ 9 を実行する必要はありませんIPv6。

```
bitnami@ip-[redacted]:~$ curl -g -6 'http://[2600:[redacted]:985b:25d9]:80'
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8">
    <title>Bitnami NGINX Open Source</title>
    <meta name="description" content="Bitnami: Open Source. Simplified.">
    <meta name="author" content="Bitnami">
    <link rel="stylesheet" media="screen" href="//unpkg.com/@bitnami/hex/dist/hex.min.css">
  </head>
  <body>
    <main class="margin-t-huge">
      <section aria-labelledby="installation-title" aria-describedby="installation-desc" class="container container-tiny margin-b-gi
        <h1 id="installation-title">Congratulations!</h1>
        <div aria-hidden="true" style="height: 4px; width: 100%;" class="gradient-135-brand"></div>
        <p id="installation-desc" class="type-big">You are now running <strong>Bitnami NGINX Open Source 1.18.0</strong> in the Clou
      </section>
      <section aria-labelledby="links-title" aria-describedby="links-desc" class="bg-light padding-v-bigger margin-v-enormous">
        <div class="container container-tiny">
          <div class="row row-collapse-b-tablet align-center ">
            <div class="col-6">
              <h3 id="links-title" class="margin-t-reset">Useful Links</h3>
              <p id="links-desc" class="margin-b-reset">The following links will help you to understand better how to get started an
unched.</p>
```

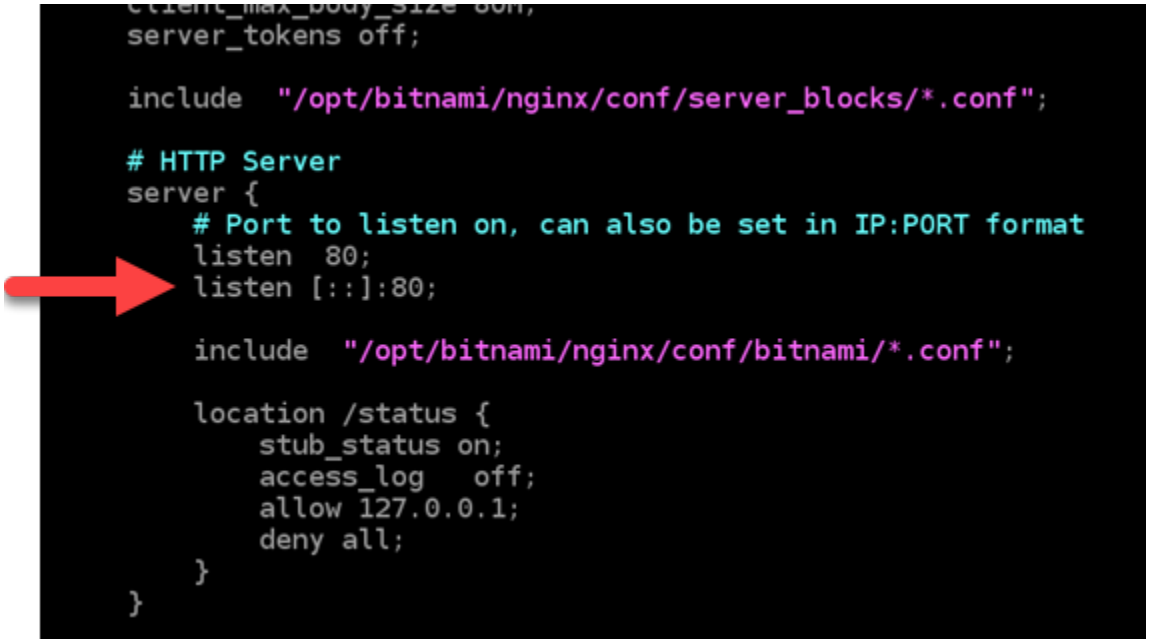
4. 次のコマンドを入力し、Vim を使用して nginx.conf 設定ファイルを開きます。

```
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. 次のコマンドを入力して、インスタンスがポート 80 経由でIPv6リクエストをリッスンしているかどうかを確認します。必ず を置き換えてください。*<IPv6Address>* インスタンスに割り当てられたIPv6アドレス。

```
curl -g -6 'http://[<IPv6Address>]'
```

例:

```
curl -g -6 'http://[2001:0db8:85a3:0000:0000:8a2e:0370:7334]'
```

次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 経由で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 で Plesk インスタンスIPv6の接続を設定する

Plesk ブループリントを使用するインスタンスにIPv6アドレスを認識させるには、追加の一連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)。

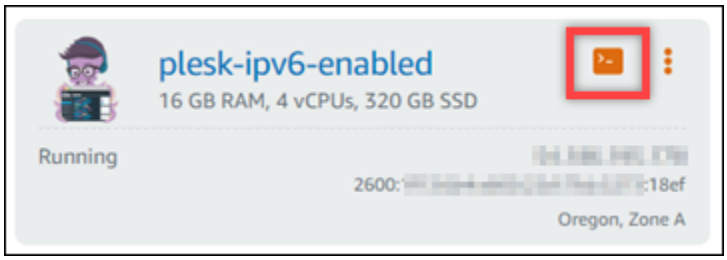
- Lightsail で Plesk インスタンスを作成します。詳細については、「インスタンスを作成する」を参照してください。
- Plesk インスタンスIPv6で を有効にします。詳細については、「 を有効または無効にするIPv6」を参照してください。

> **Note**
>
> 2021 年 1 月 12 日以降に作成された Lightsail Plesk インスタンスは、デフォルトでIPv6有効になっています。インスタンスの作成時に がデフォルトで有効になっている場合でも、インスタンスIPv6で を設定するにはIPv6、このガイドの次のステップを完了する必要があります。

Plesk インスタンスIPv6で を設定する

Lightsail の Plesk インスタンスIPv6で 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 ホームページの「インスタンス」セクションで、設定する Plesk インスタンスを見つけ、ブラウザベースのSSHクライアントアイコンを選択して を使用して接続しますSSH。



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。

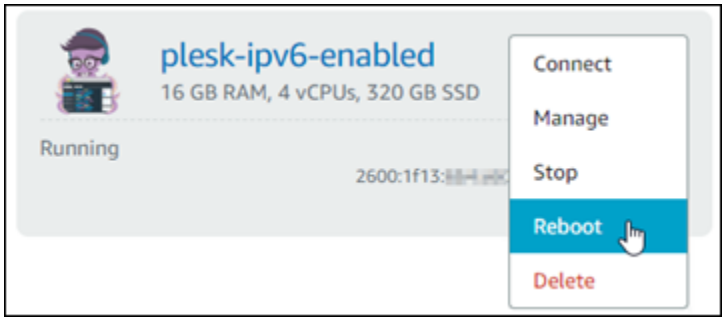
- インスタンスがIPv6アドレスを認識しない場合、レスポンスにはリスト表示されません。この手順のステップ 4 ~ 7 を続行する必要があります。

```
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、この例scope globalに示すように、レスポンスに と表示されます。ここで停止する必要があります。インスタンスはIPv6アドレスを認識するように既に設定されているため、この手順のステップ 4 ~ 7 を完了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。

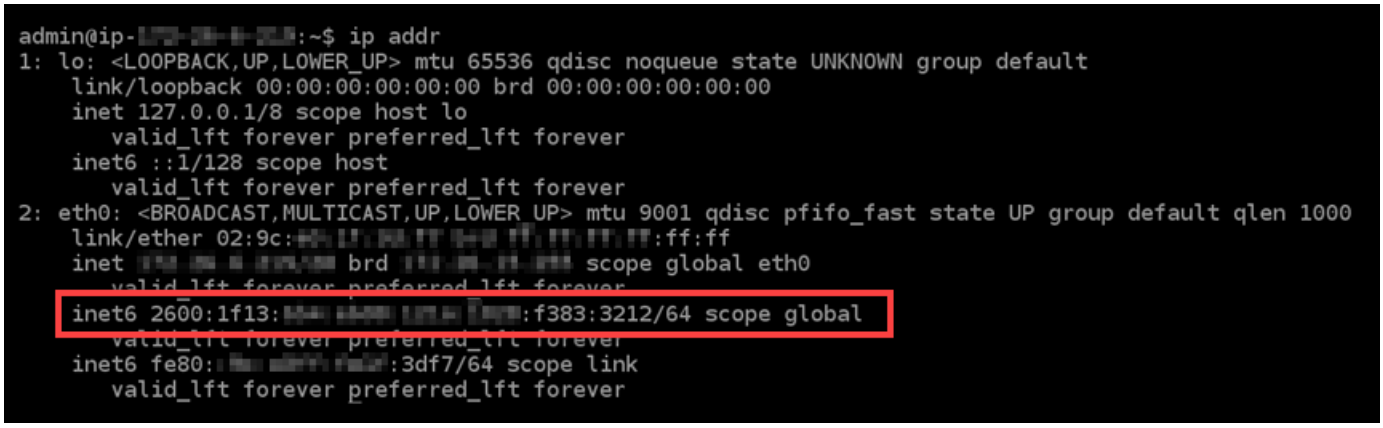


インスタンスが再起動するまで数分待ってから、次のステップに進みます。

6. Plesk インスタンスのSSHセッションに戻ります。
7. 次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アドレスが認識され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、この例`scope global`に示すように、 のラベルが付いたレスポンスにそのアドレスが表示されます。



Lightsail で Ubuntu 16 インスタンスIPv6の接続を設定する

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、インスタンスにパブリックIPv6アドレスを割り当てるために を有効にIPv6できます。詳細については、「IP アドレス」およびAmazon Lightsail IPv6での有効化または無効化」を参照してください。

Ubuntu 16 ブループリントを使用するインスタンスIPv6に対して を有効にした後、追加の一連のステップを実行して、インスタンスにその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、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)。

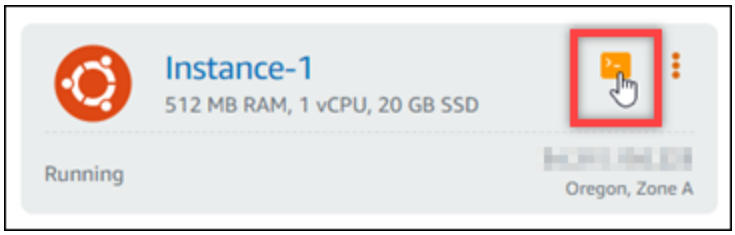
- Lightsail で Ubuntu 16 インスタンスを作成します。詳細については、「インスタンスを作成する」を参照してください。
- Ubuntu 16 インスタンスIPv6で を有効にします。詳細については、「 を有効または無効にするIPv6」を参照してください。

> **Note**
>
> 2021 年 1 月 12 日以降に作成された新しい Ubuntu インスタンスは、Lightsail コンソールで作成されたときにデフォルトでIPv6有効になっています。インスタンスの作成時に がデフォルトで有効になっている場合でも、インスタンスIPv6で を設定するにはIPv6、このガイドの次のステップを完了する必要があります。

Ubuntu 16 インスタンスIPv6で を設定する

Lightsail の Ubuntu 16 インスタンスIPv6で 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 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 インスタンスを見つけ、ブラウザベースのSSHクライアントアイコンを選択して を使用して接続しますSSH。



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、レスポンスにはリスト表示されません。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-[redacted]:~$ 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、この例`scope global`に示すように、レスポンスに と表示されます。ここで停止する必要があります。インスタンスはIPv6アドレスを認識するように既に設定されているため、この手順のステップ 4 ~ 9 を実行する必要はありません。

```
ubuntu@ip-[redacted]:~$ 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 2600:1f13:[redacted]:ed2c:91e2/128 scope global
       valid_lft forever preferred_lft forever
    inet6 fe80::af:[redacted]:16bf/64 scope link
       valid_lft forever preferred_lft forever
```

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。

```
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

# The loopback network interface
auto lo
iface lo inet loopback

# Source interfaces
# Please check /etc/network/interfaces.d before changing this file
# as interfaces may have been defined in /etc/network/interfaces.d
# See LP: #1262951
source /etc/network/interfaces.d/*.cfg

iface eth0 inet6 dhcp
```

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。

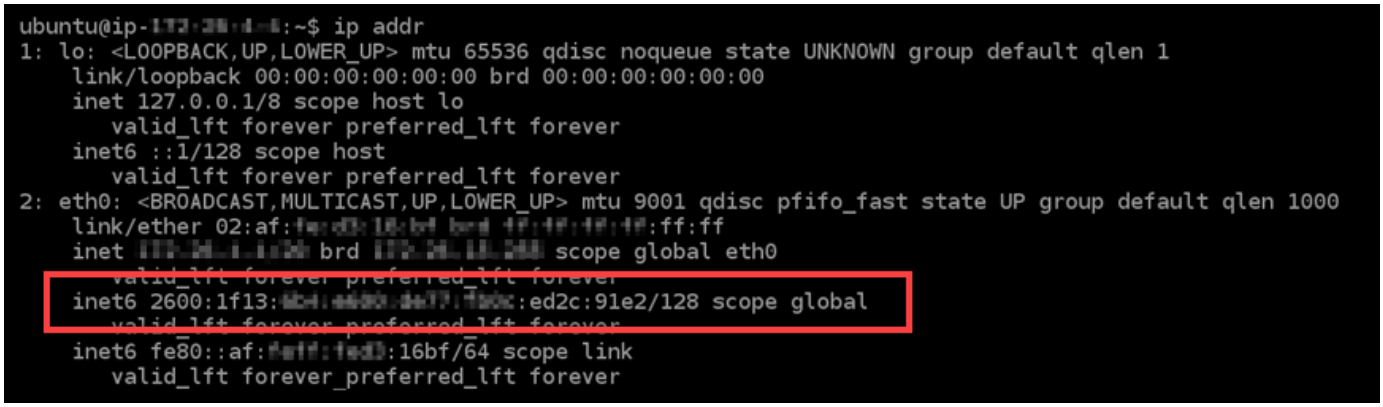
```
sudo service networking restart
```

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アドレスを認識できるように、さらに数分かかる場合があります。

9. 次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アドレスが認識され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、この例scope globalに示すように、 のラベルが付いたレスポンスにそのアドレスが表示されます。



Lightsail インスタンスの設計図IPv6で を設定する方法については、 step-by-step 「」の手順に従ってください。

このガイドでは、、DebiancPanel、、Nginx、Plesk GitLab、Ubuntu 16 など、さまざまなインスタンスブループリントについて説明します。手順にはSSH、 経由でインスタンスに接続する、ネットワーク設定ファイルを変更する、サービスを再起動する、インスタンスが割り当てられたIPv6アドレスを認識することを確認することが含まれます。このガイドに従うことで、Lightsail インスタンスが IPv4と IPv6 アドレスの両方を利用するように適切に設定され、より良い接続が可能になり、インターネットの将来に向けてアプリケーションを準備することができます。

Lightsail オペレーション AWS CLI の をセットアップする

AWS Command Line Interface （ AWS CLI) は、上級ユーザーとデベロッパーがターミナル (Linux および Unix) またはコマンドプロンプト (Windows) にコマンドを入力して Amazon Lightsail サービスを制御できるようにするツールです。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 () を使用して Lightsail を制御することもできますAPI。

Lightsail では、ローカルデスクトップ AWS CLI に をインストールするか、Lightsail インスタンスにインストールできます。

の詳細については AWS CLI、「 AWS Command Line Interface ユーザーガイド」を参照してください。Amazon Lightsail コマンドは、 AWS CLI コマンドリファレンス にあります。

- をローカルデスクトップ AWS CLI にインストールするには、 AWS Command Line Interface ドキュメントの「 AWS CLIのインストール」を参照してください。
- Ubuntu ベースの Lightsail インスタンス AWS CLI に をインストールするには、インスタンスに接続し、 と入力しますsudo apt-get -y install awscli。

> **Note**
>
> は、Amazon Linux Lightsail インスタンスに既にインストール AWS CLI されている必要があります。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。

をインストールしたら AWS CLI、アクセスキーを取得し、それらを使用する AWS CLI ように を設定する必要があります。詳細については、「Lightsail APIまたは を使用するためのアクセスキーを作成する AWS Command Line Interface」を参照してください。

Lightsail APIと のアクセスキーを生成する AWS CLI

Lightsail APIまたは AWS Command Line Interface （ AWS CLI) を使用するには、新しい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。次の手順を使用してキーを作成し、Lightsail を呼び出す AWS CLI ように を設定しますAPI。

ステップ 1: 新規アクセスキーを作成する

AWS Identity and Access Management （ IAM) コンソールで新しいアクセスキーを作成できます。

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 アクションへのフルアクセスまたは特定のアクセス権限が必要です。

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: を設定する AWS CLI

をインストールしていない場合は AWS CLI、今すぐインストールできます。「AWS Command Line Interfaceのインストール」を参照してください。をインストールしたら AWS CLI、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 AWS リージョン がある 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。

オプションの使用の詳細については、「 AWS CLI リファレンス」の AWS CLI `--region`「一般的なオプション」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

- のインストール SDK
- Amazon Lightsail で動作する AWS Command Line Interface ように を設定する
- APIドキュメントを読む

Lightsail LAMP インスタンスに PHP アプリケーションをデプロイする

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利用できます。

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礎が整います。

目次

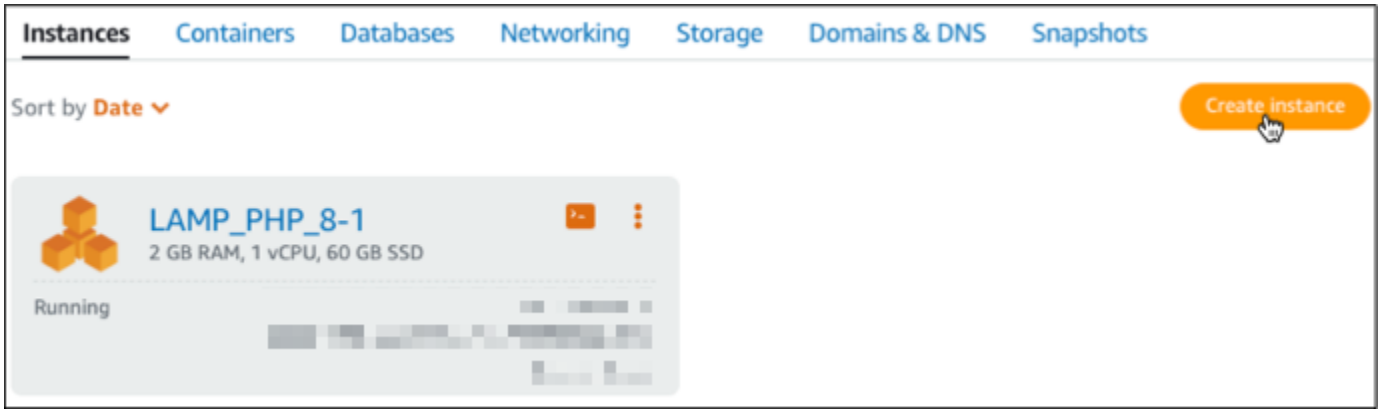
ステップ 1: AWS にサインアップ

このチュートリアルには 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。

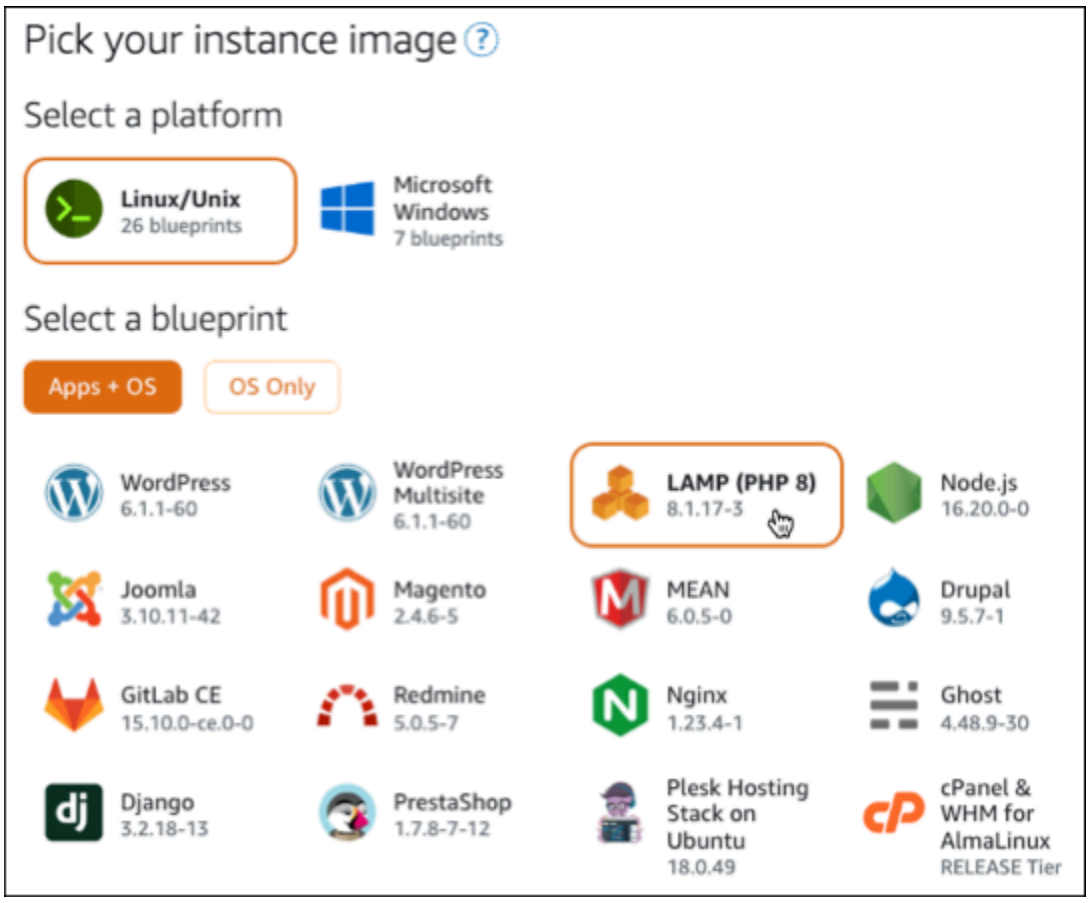


3. インスタンスの AWS リージョン とアベイラビリティーゾーンを選択します。

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。5 USD Lightsail プランを 1 か月間 (最大 750 時間) 無料で試すことができます。は 1 か月分の無料 AWS クレジットをアカウントに付与します。

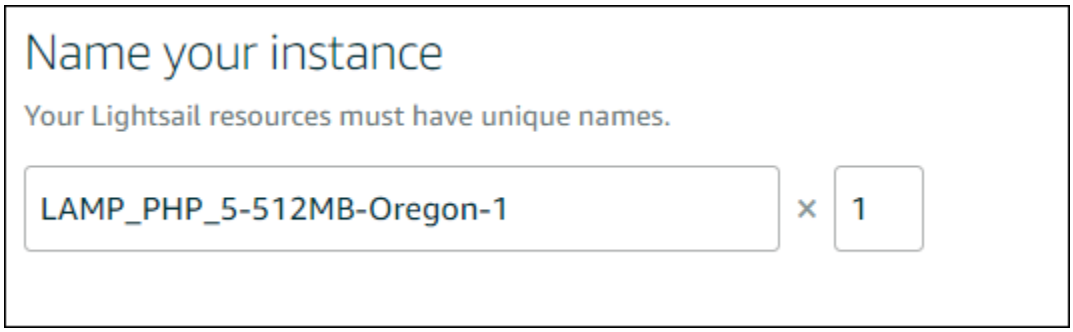
   > **Note**
   >
   > AWS 無料利用枠の一部として、一部のインスタンスバンドルで Amazon Lightsail を無料で使い始めることができます。詳細については、Amazon Lightsail の料金」ページのAWS 「無料利用枠」を参照してください。

6. インスタンスの名前を入力します。

   リソース名:

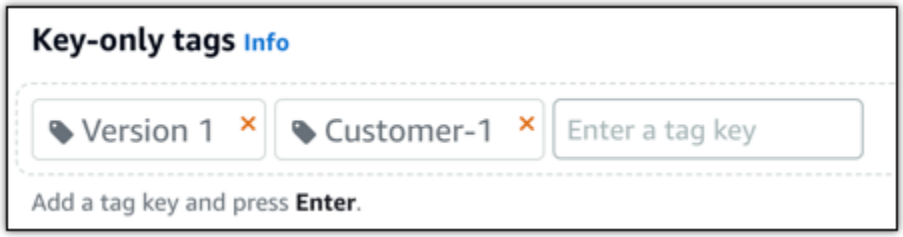
   - Lightsail アカウントの各 AWS リージョン 内で一意である必要があります。

- 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。



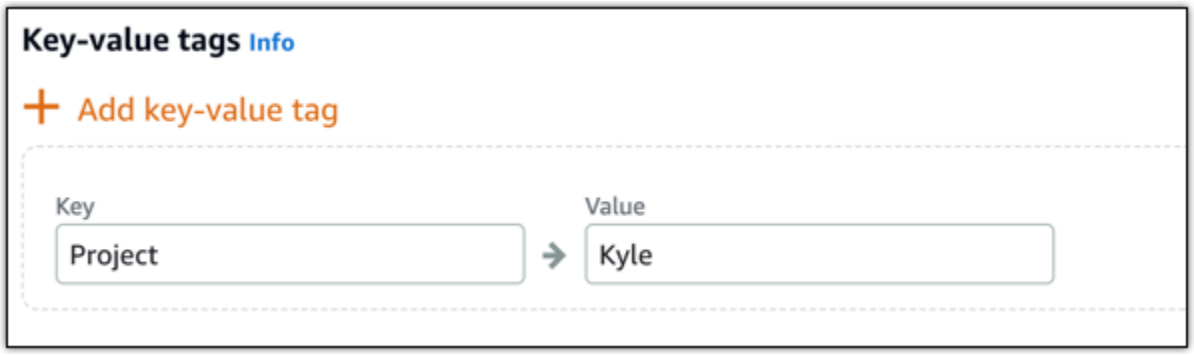
7. 以下のいずれかのオプションを選択して、インスタンスにタグを追加します。

   - [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。



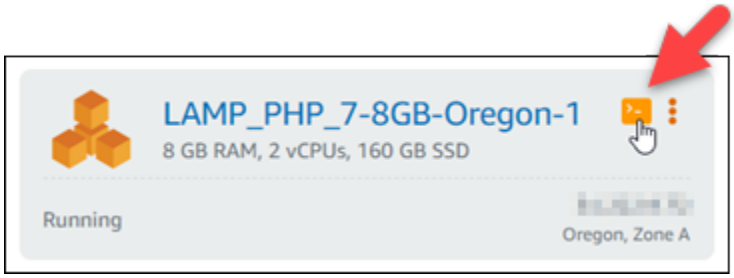
> **Note**
>
> 「キーのみ」のタグと「キーバリュー」のタグの詳細については、「タグ」を参照してください。

8. [インスタンスの作成] を選択します。

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取得します。詳細については、Amazon Lightsail」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。

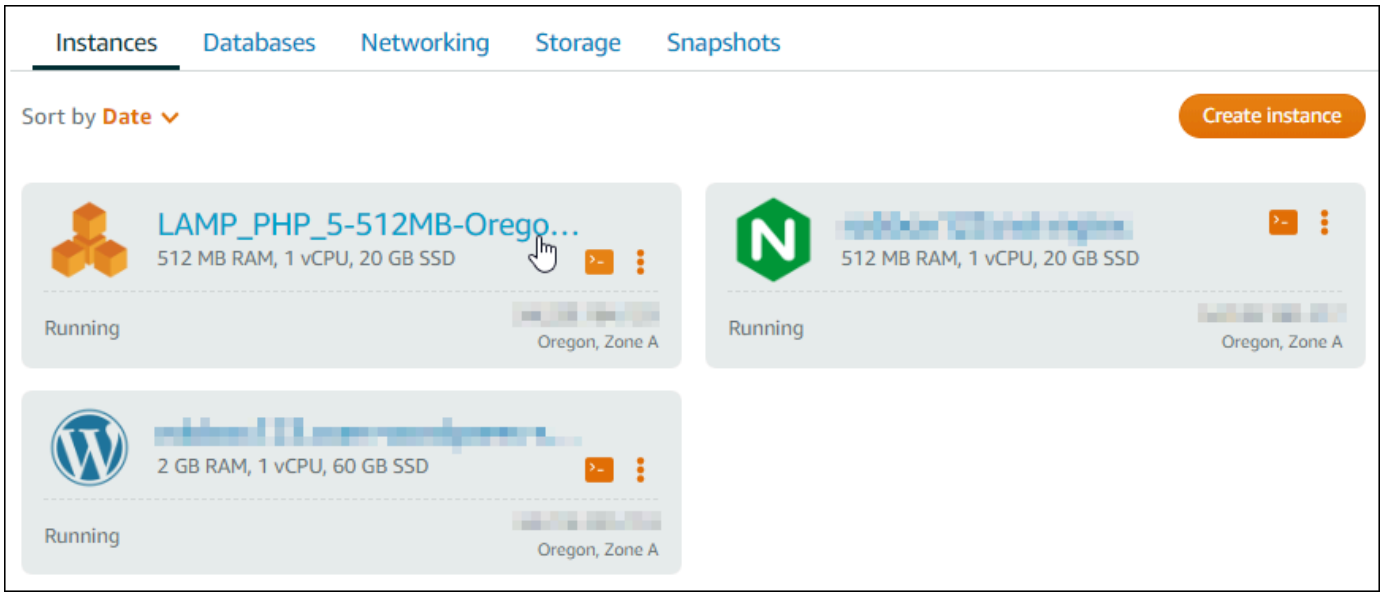
Bitnami アプリケーションは、モジュールのインストーラを使用してインストールすることもできます。Bitnami ウェブサイトから WordPress、Drupal、Magento、Moodle などのアプリケーションをダウンロードし、サーバーの機能を拡張します。Bitnami アプリケーションのインストールの詳細については、Bitnami ドキュメントの「開始方法」を参照してください。

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

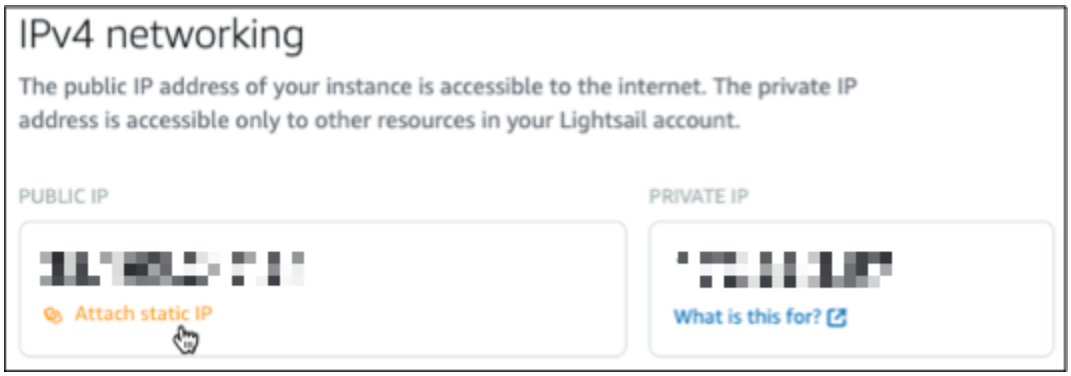
LAMP インスタンスのデフォルトのパブリック IP は、インスタンスを停止して開始すると変わります。インスタンスにアタッチした静的 IP アドレスは、インスタンスを停止して開始しても変わりません。

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。



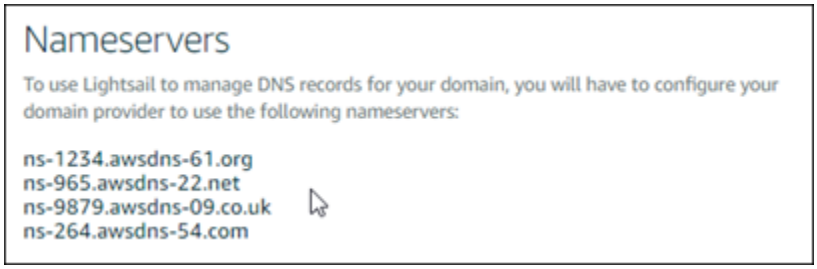
3. 静的 IP に名前を付け、[作成してアタッチ] を選択します。

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より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 に転送します。



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ステップを以下に示します。

- 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

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、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されている MariaDB データベースに保存されます。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 の LAMP インスタンスから Amazon RDS の Aurora マネージドデータベースにアプリケーションデータベースを接続する方法を示します。

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 LAMP インスタンスを作成し、そのインスタンスで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、「Amazon VPC ピアリングをセットアップして Lightsail の外部の AWS リソースと連携させる」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。

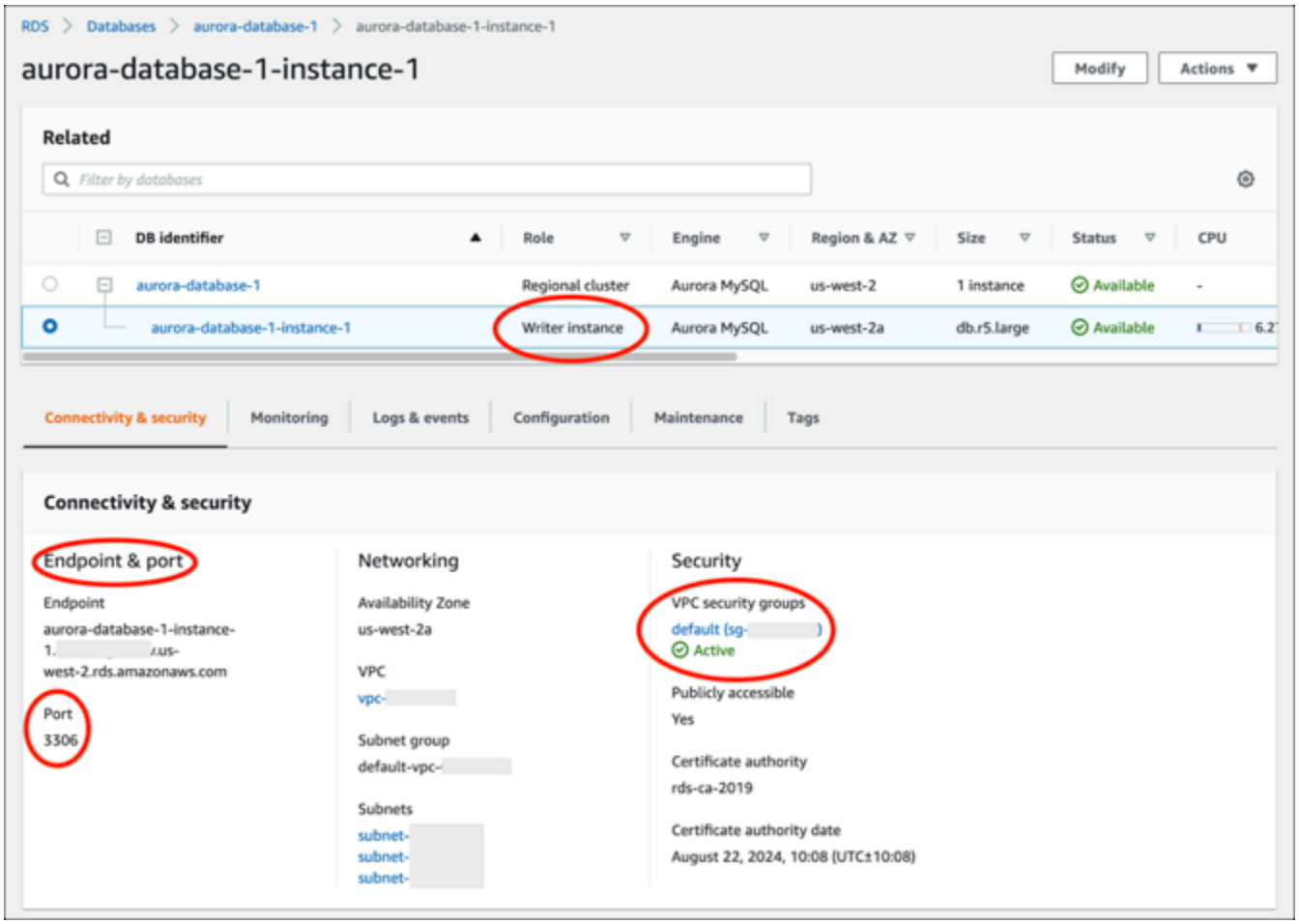
ステップ 2: Aurora データベースのセキュリティグループを設定する

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。

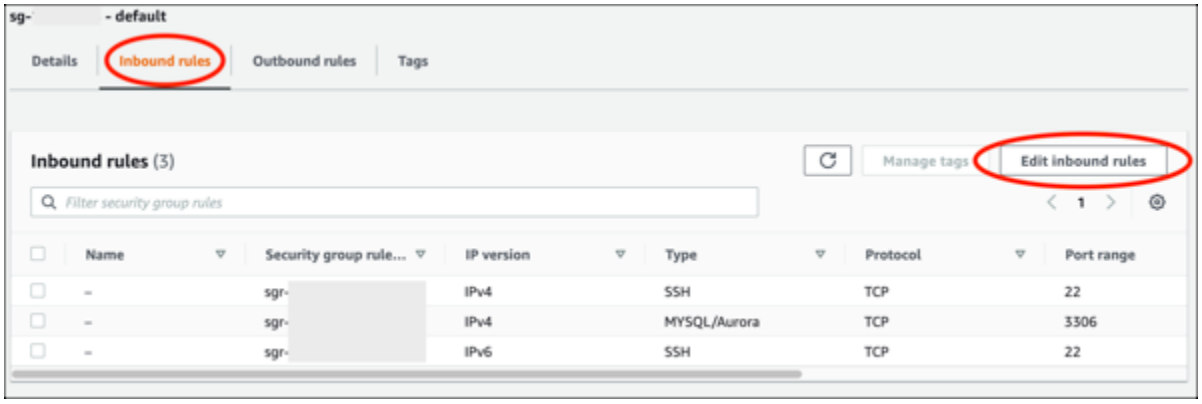
LAMP インスタンスが Aurora データベースへの接続を確立できるよう、以下の手順を完了してセキュリティグループを設定します。

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。これらは、後でデータベースに接続するように 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。

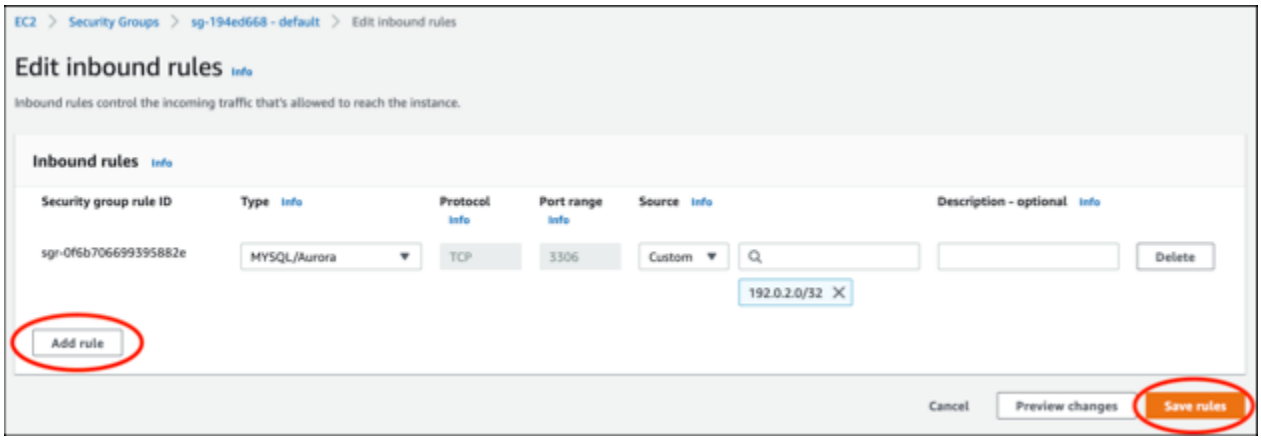


7. Aurora データベースのセキュリティグループが選択されていることを確認します。
8. [Inbound rules] (インバウンドルール) タブを開きます。
9. [Edit inbound rules] (インバウンドルールの編集) を選択します。



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。

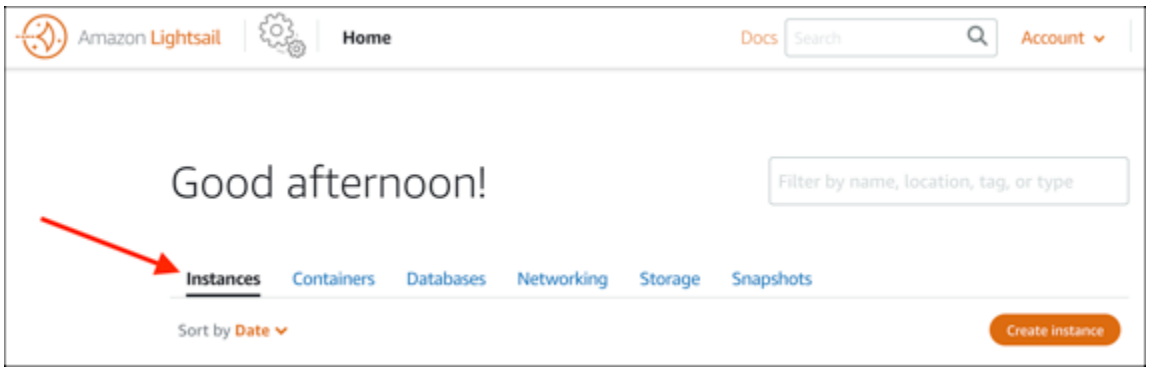
    - デフォルトの MySQL ポート 3306 を使用する場合は、[Type] (タイプ) ドロップダウンメニューから [MySQL/Aurora] を選択します。
    - データベースのカスタムポートを使用する場合は、[Type] (タイプ) ドロップダウンメニューから [Custom TCP] (カスタム TCP) を選択し、[Port Range] (ポート範囲) テキストボックスにポート番号を入力します。
12. [Source] (ソース) テキストボックスに LAMP インスタンスのプライベート IP アドレスを追加します。IP アドレスは、 CIDR 表記で入力する必要があります (/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

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. 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。コマンドで、 を Aurora データベースのエンドポイントアドレス*DatabaseEndpoint*に置き換え、*Port* をデータベースのポートに置き換えます。を、データベースの作成時に入力したユーザーの名前*MyUserName*に置き換えます。

```
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)]>
```

このレスポンス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、Lightsail インスタンスのプライベート IP アドレスが接続できるようにデータベースのセキュリティグループを設定する必要があります。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';
```

Lightsail で 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利用できます。

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礎が整います。

目次

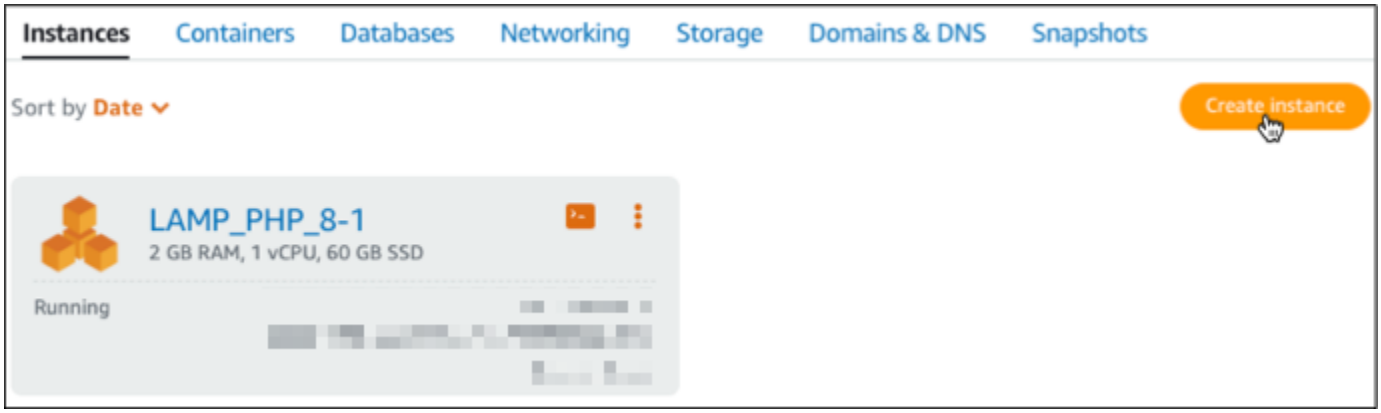
ステップ 1: AWS にサインアップ

このチュートリアルには 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: Lightsail で Windows Server 2016 インスタンスを作成する

Windows Server 2016 インスタンスを Lightsail で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。

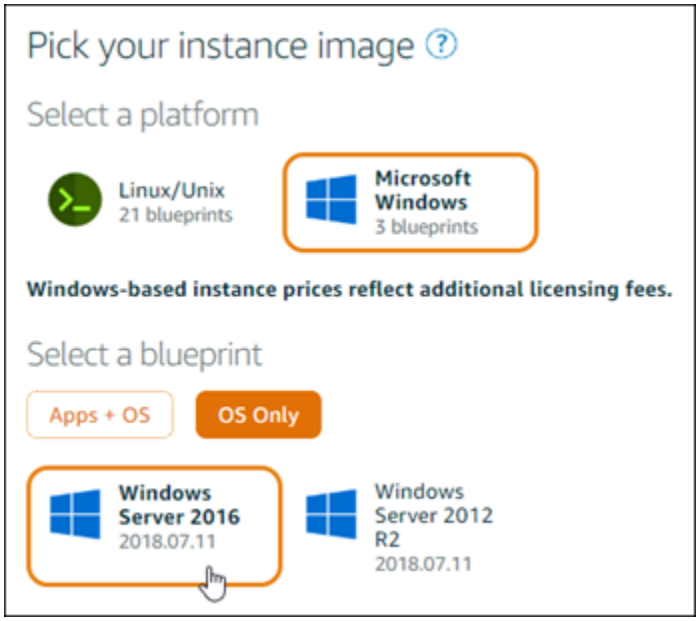


3. インスタンスの AWS リージョン とアベイラビリティーゾーンを選択します。

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 [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。

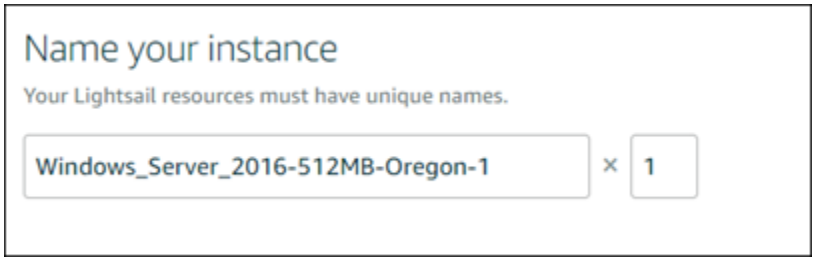
   プランには、低額で予測可能なコスト、マシン設定 (RAM、SSD、vCPU)、およびデータ転送枠が含まれます。9.50 USD Lightsail プランを 1 か月間 (最大 750 時間) 無料で試すことができます。 は 1 か月分の無料 AWS クレジットをアカウントに付与します。

   > **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 の料金」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。

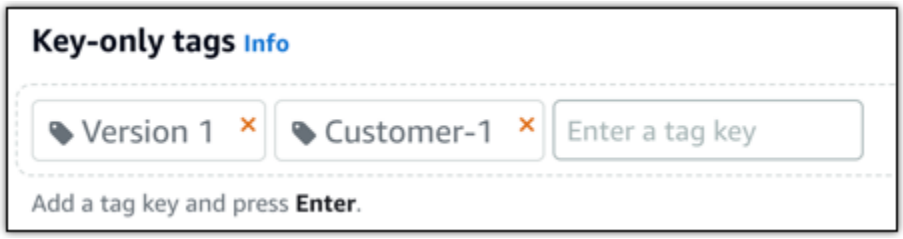
   リソース名:

   - 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。



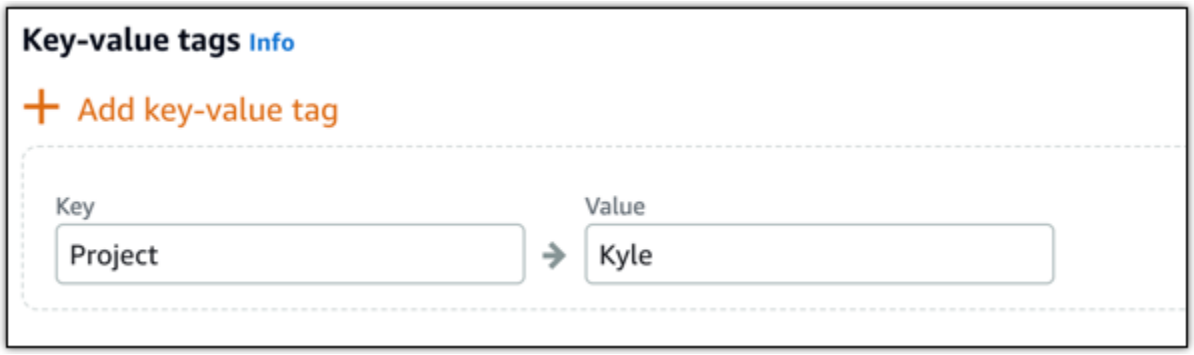
7. 以下のいずれかのオプションを選択して、インスタンスにタグを追加します。

   - [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。



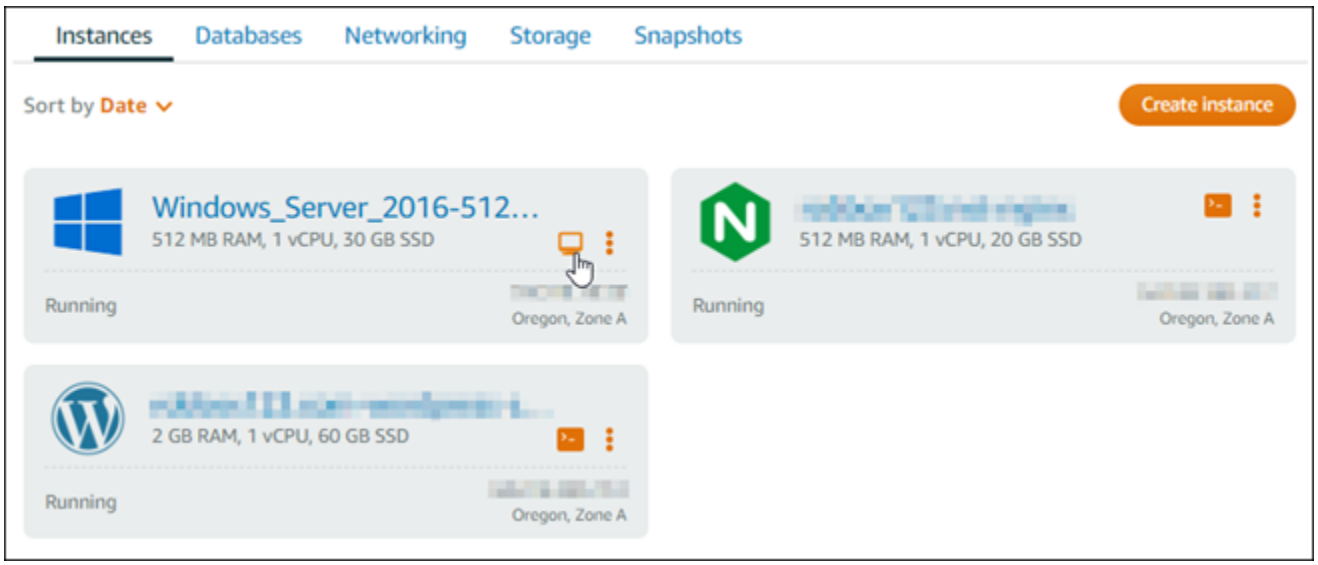
> **Note**
>
> 「キーのみ」のタグと「キーバリュー」のタグの詳細については、「タグ」を参照してください。

8. [インスタンスの作成] を選択します。

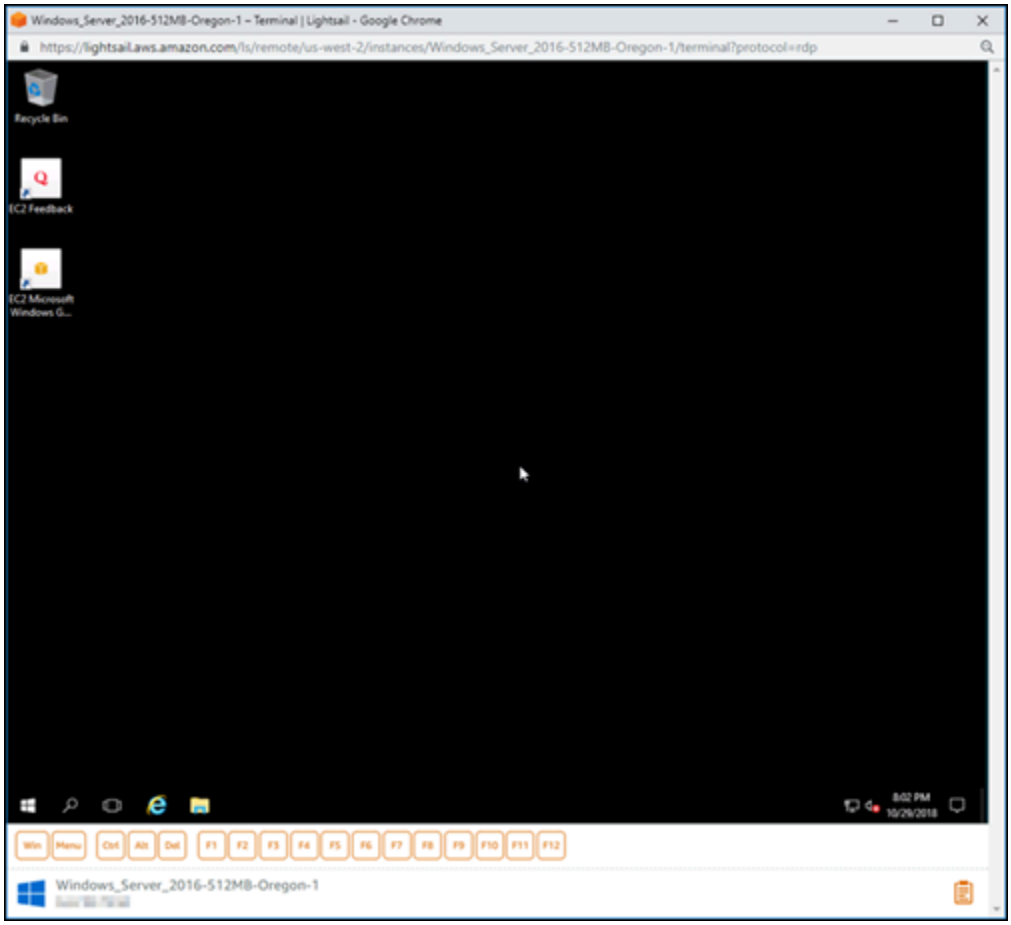
ステップ 3: RDP 経由で Windows Server 2016 インスタンスに接続する

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。

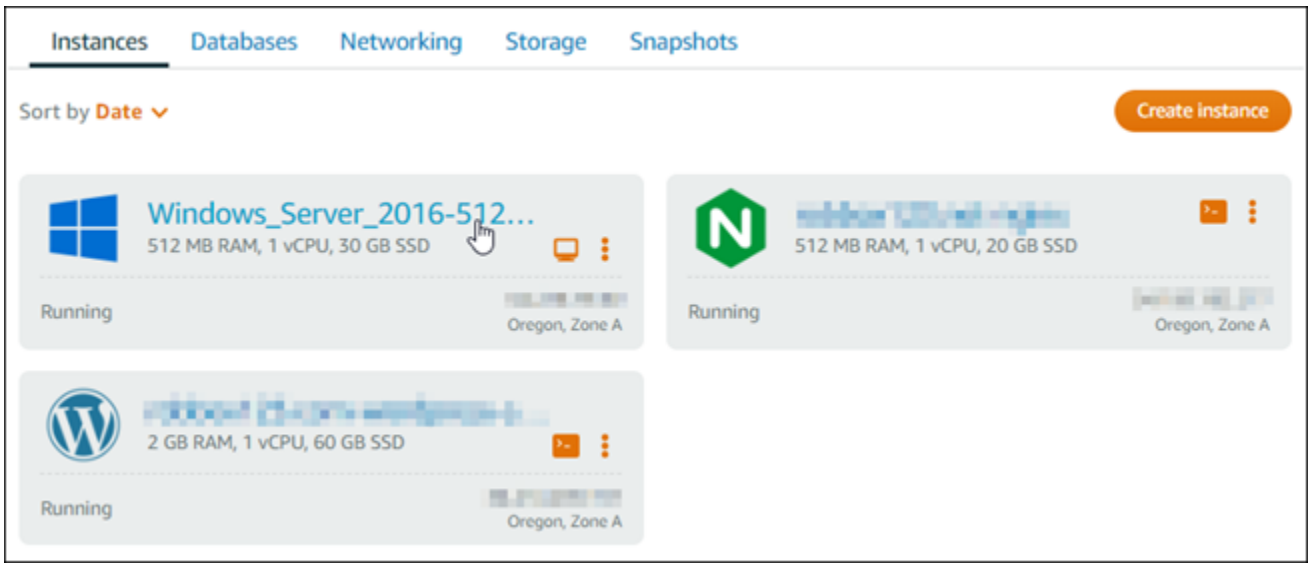


ステップ 4: 静的 IP アドレスを作成して Windows Server 2016 インスタンスにアタッチする

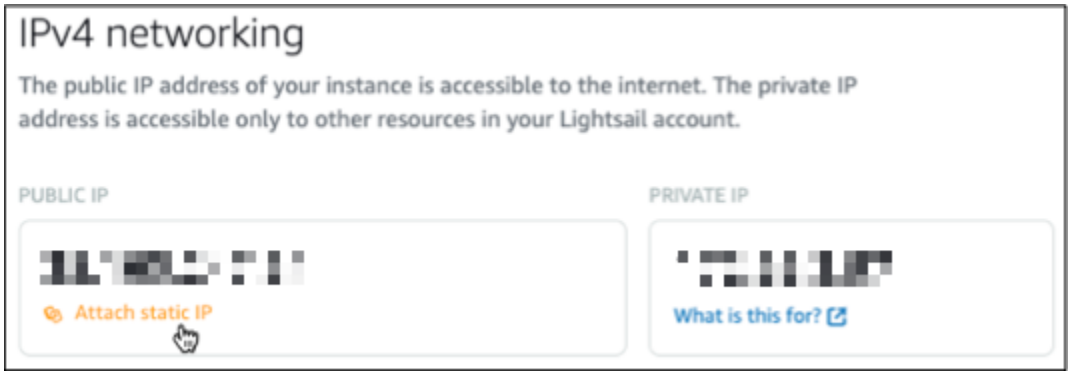
Windows Server 2016 インスタンスのデフォルトのパブリック IP は、インスタンスを停止して開始すると変わります。インスタンスにアタッチした静的 IP アドレスは、インスタンスを停止して開始しても変わりません。

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。

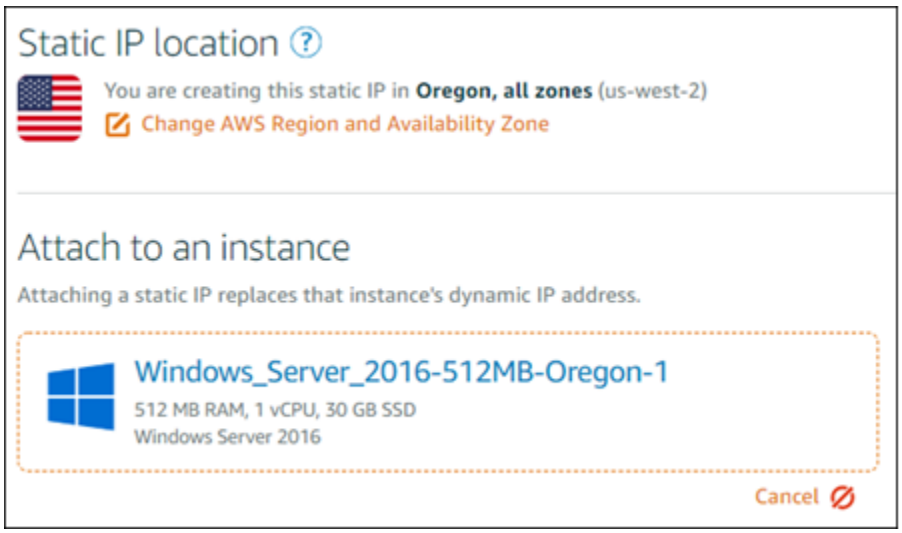
1. Lightsail ホームページのインスタンスタブで、実行中の Windows Server 2016 インスタンスを選択します。



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:

   - 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。

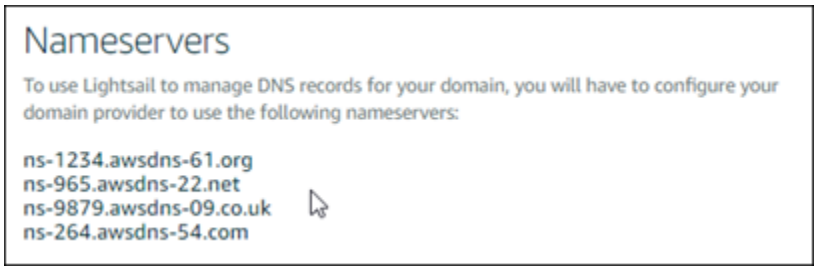
5. [Create(作成)] を選択します。

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 に転送します。

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の手順をいくつか紹介します。

- 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

で Lightsail APIアクティビティをモニタリングする AWS CloudTrail

Amazon Lightsail は と統合されています。これは AWS CloudTrail、Lightsail のユーザー、ロール、または サービスによって実行されたアクションを記録する AWS サービスです。Lightsail のすべてのAPI呼び出しをイベントとして CloudTrail キャプチャします。キャプチャされた呼び出しには、Lightsail コンソールからの呼び出しと Lightsail APIオペレーションへのコード呼び出しが含まれます。証跡を作成する場合は、Lightsail の CloudTrail イベントなど、Amazon S3 バケットへのイベントの継続的な配信を有効にすることができます。証跡を設定しない場合でも、 CloudTrail コンソールのイベント履歴 で最新のイベントを表示できます。によって収集された情報を使用して CloudTrail、Lightsail に対するリクエスト、リクエスト元の IP アドレス、リクエスト者、リクエスト日時などの詳細を確認できます。

の詳細については CloudTrail、「 AWS CloudTrail ユーザーガイド」を参照してください。

の Lightsail 情報 CloudTrail

CloudTrail AWS アカウントを作成すると、 がアカウントで有効になります。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 CloudTrail イベント履歴 の他の AWS サービスイベントとともに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、「イベント履歴を使用した CloudTrail イベントの表示」を参照してください。

Lightsail のイベントなど、 AWS アカウント内のイベントの継続的な記録については、証跡を作成します。証跡により CloudTrail 、 はログファイルを Amazon S3 バケットに配信できます。デフォルトでは、コンソールで証跡を作成すると、証跡はすべてのAWSリージョンに適用されます。証

跡は、 AWS パーティション内のすべてのリージョンからのイベントをログに記録し、指定した Amazon S3 バケットにログファイルを配信します。さらに、 CloudTrail ログで収集されたイベントデータをさらに分析し、それに基づいて行動するように他の AWS 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:

- 証跡の作成のための概要
- CloudTrail サポートされているサービスと統合
- の Amazon SNS Notifications の設定 CloudTrail
- 複数のリージョンからの CloudTrail ログファイルの受信と複数のアカウントからのログファイルの受信 CloudTrail

すべての Lightsail アクションは によってログに記録 CloudTrail され、Amazon Lightsail API リファレンス に文書化されます。例えば、、、および RebootInstanceセクションを呼び出すAttachStaticIpとGetInstance、 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、誰がリクエストを生成したかという情報が含まれます。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または AWS Identity and Access Management （ IAM) ユーザー認証情報のどちらを使用して行われたか。
- リクエストがロールまたはフェデレーション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 サービスによって行われたかどうか。

詳細については、「 CloudTrail userIdentity要素」を参照してください。

Lightsail ログファイルエントリについて

証跡は、指定した Amazon S3 バケットにイベントをログファイルとして配信できるようにする設定です。 CloudTrail ログファイルには 1 つ以上のログエントリが含まれます。イベントは任意の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、リクエストパラメータなどに関する情報が含まれます。 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。

Lightsail の問題をトラブルシューティングするための HAR ファイルの作成

Amazon Lightsail コンソールまたは Lightsail Virtual Private Server (VPS) で問題が発生した場合は、ウェブブラウザから HAR ファイルを送信するように求められる AWS Support ことがあります。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 ファイルでは、 AWS Support がこれらの問題を調査またはレプリケート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、 step-by-step 次の手順に従います。

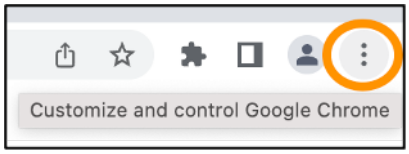
目次

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。

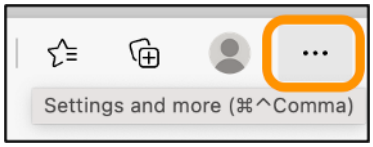


2. [More tools] (その他のツール) で、[Developer tools] (デベロッパーツール) を選択します。
3. ブラウザで DevTools を開き、ネットワークパネルを選択します。
4. [Preserve log] (ログを保持) チェックボックスを選択します。
5. 現在のネットワークリクエストをすべてクリアするには、[Clear] (クリア) を選択します。
6. 直面している問題を再現します
7. で DevTools、ネットワークリクエストのコンテキスト (右クリック) メニューを開きます。
8. [Save all as HAR with content] (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。

詳細については、Google Developers ウェブサイトの「Chrome を開く DevTools」および「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)

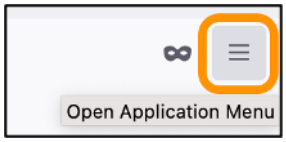
1. ブラウザの右上にある [設定など] を選択します。



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 を開き、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. で DevTools、ネットワークリクエストのコンテキスト (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 で Prometheus を使用してシステムリソースとアプリケーションをモニタリングする

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 インスタンスにノードエクスポーターを使用して Prometheus をインストールおよび設定する方法を示します。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

Amazon Lightsail インスタンスに Prometheus をインストールする前に、次の操作を行う必要があります。

- Lightsail で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」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。

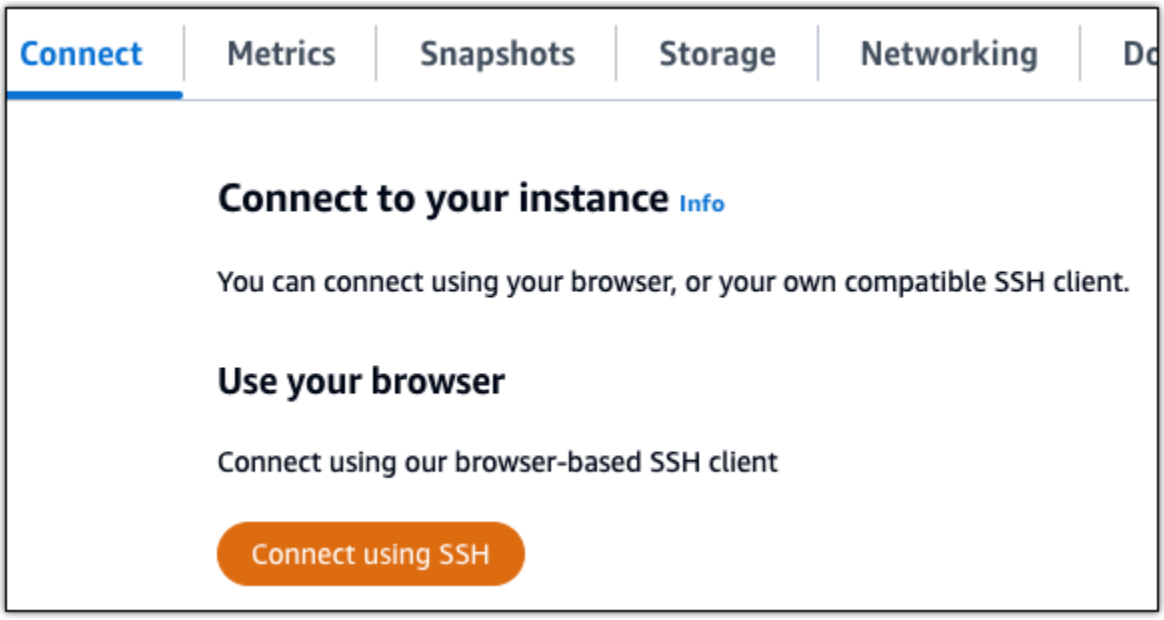
ステップ 2: Lightsail インスタンスにユーザーとローカルシステムディレクトリを追加する

SSH を使用して Lightsail インスタンスに接続し、ユーザーとシステムディレクトリを追加するには、次の手順を実行します。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
```

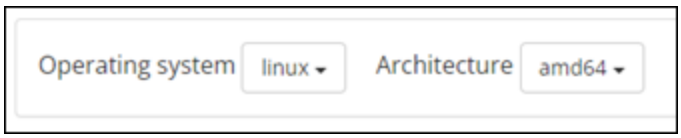
```
sudo chown prometheus:prometheus /etc/prometheus
```

```
sudo chown prometheus:prometheus /var/lib/prometheus
```

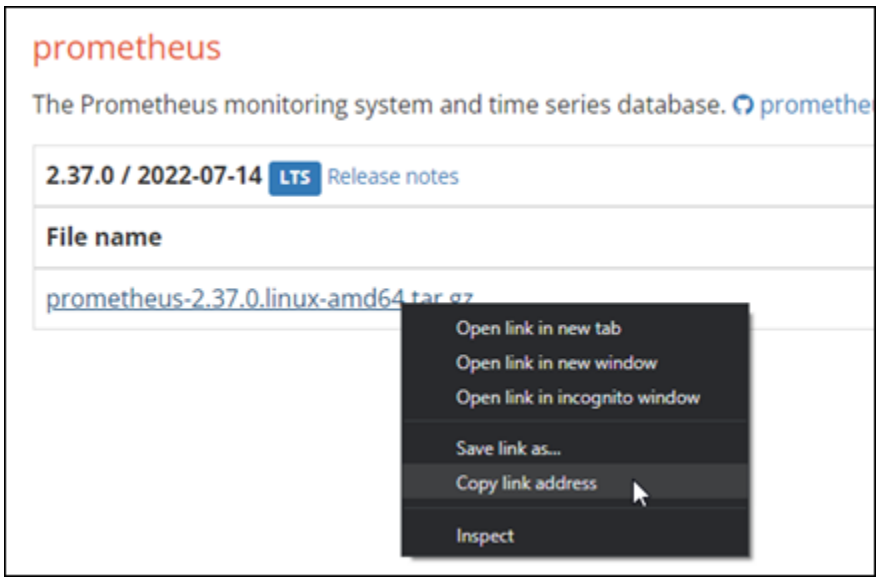
ステップ 3: Prometheus バイナリパッケージをダウンロードする

次の手順を実行して、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
```

を、この手順の前半でコピーしたアドレス*prometheus-download-address*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。

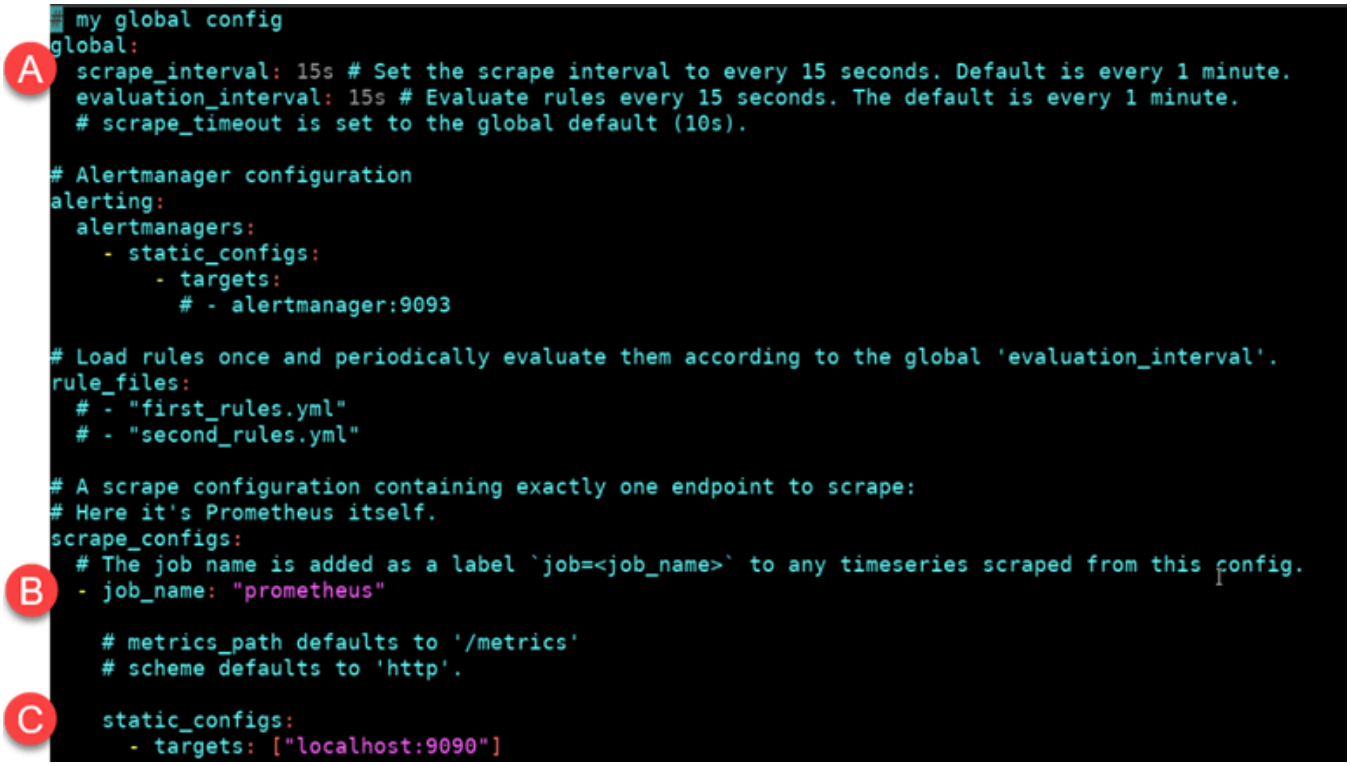
```
sudo cp /etc/prometheus/prometheus.yml /etc/prometheus/prometheus.yml.backup
```

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。




```
# my global config
global:
  scrape_interval: 15s # Set the scrape interval to every 15 seconds. Default is every 1 minute.
  evaluation_interval: 15s # Evaluate rules every 15 seconds. The default is every 1 minute.
  # scrape_timeout is set to the global default (10s).

# Alertmanager configuration
alerting:
  alertmanagers:
    - static_configs:
        - targets:
          # - alertmanager:9093

# Load rules once and periodically evaluate them according to the global 'evaluation_interval'.
rule_files:
  # - "first_rules.yml"
  # - "second_rules.yml"

# A scrape configuration containing exactly one endpoint to scrape:
# Here it's Prometheus itself.
scrape_configs:
  # The job name is added as a label `job=<job_name>` to any timeseries scraped from this config.
  - job_name: "prometheus"

    # metrics_path defaults to '/metrics'
    # scheme defaults to 'http'.

    static_configs:
      - targets: ["localhost:9090"]
```

> **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
```

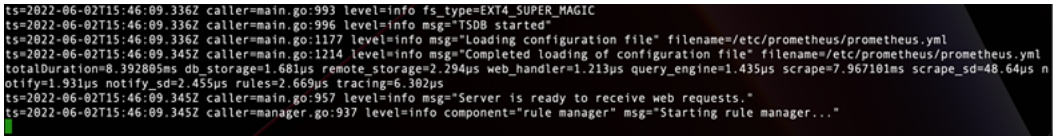
ステップ 5: Prometheus をスタートする

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
```

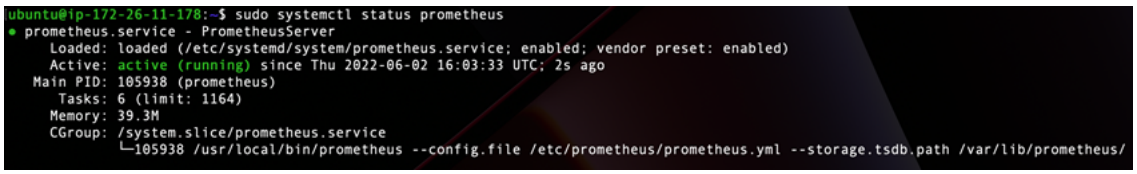
8. 次のコマンドを入力して Prometheus を再起動します。

```
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。

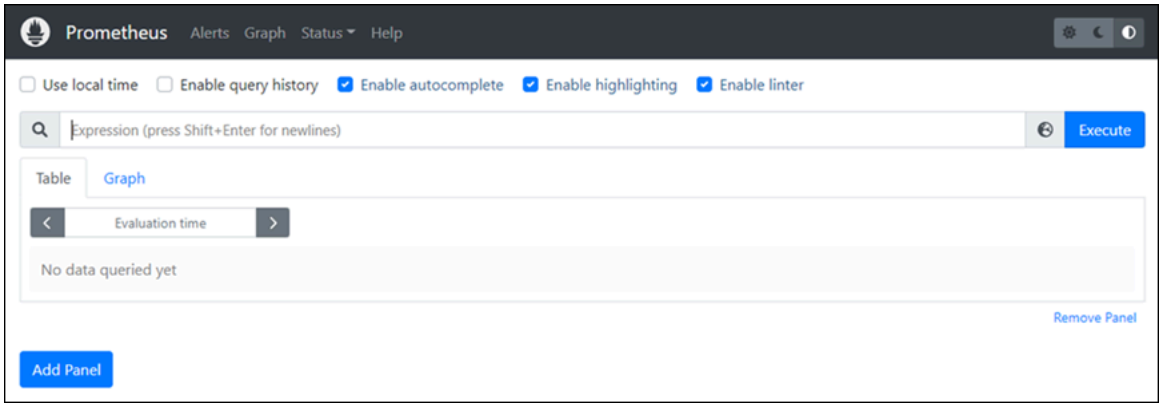
11. 次のコマンドを入力して、インスタンスが起動されたときに Prometheus が起動できるようにします。

```
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
```

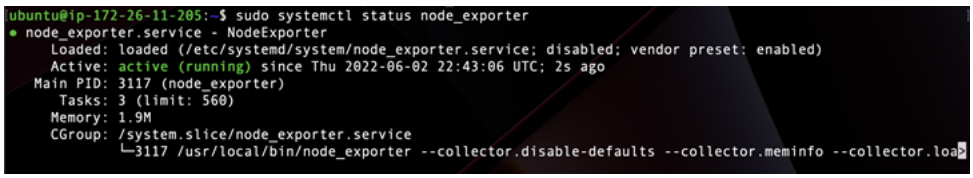
7. 次のコマンドを入力して node_exporter サービスを開始します。

```
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。

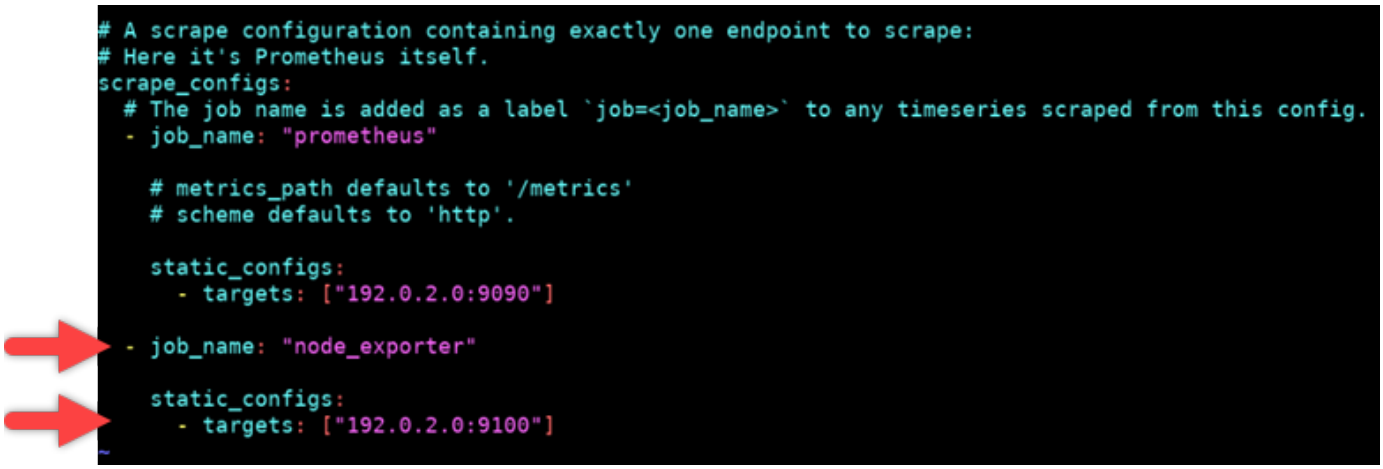
1. SSH を使用して Lightsail インスタンスに接続します。
2. 次のコマンドを入力して、 Vim を使用し、prometheus.yml ファイルを開きます。

```
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。
- の設定と同様にprometheusjob_name、*<ip_addr>* を Lightsail インスタンスにアタッチされている静的 IP アドレスに置き換えます。

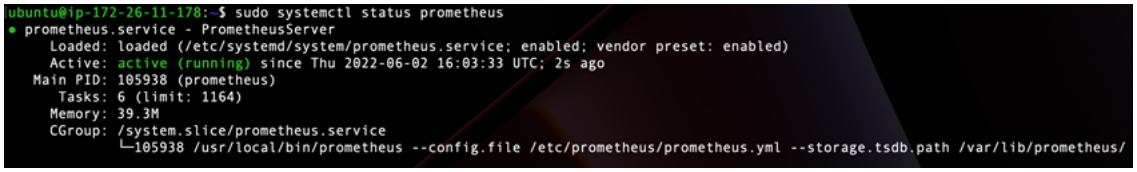
5. [Esc] キーを押して挿入モードを終了し、[:wq!] と入力して変更内容を保存して Vim を終了します。
6. 以下のコマンドを入力して Prometheus サービスを再起動し、設定ファイルへの変更を確定します。

```
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
```

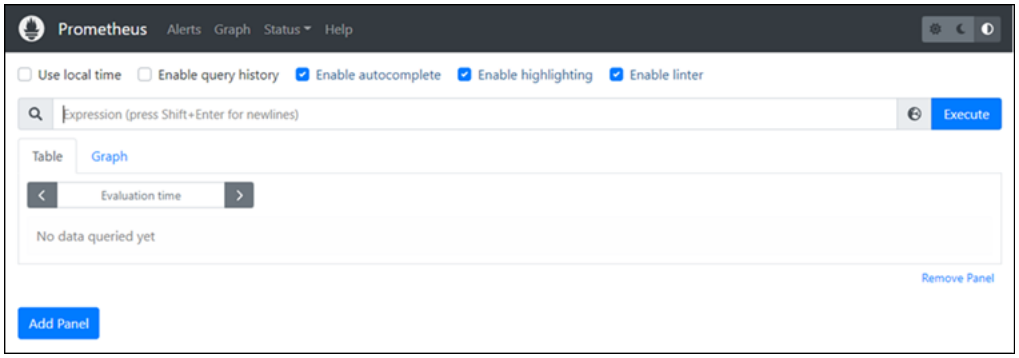
サービスが正常に再起動された場合は、次のような出力が表示されます。



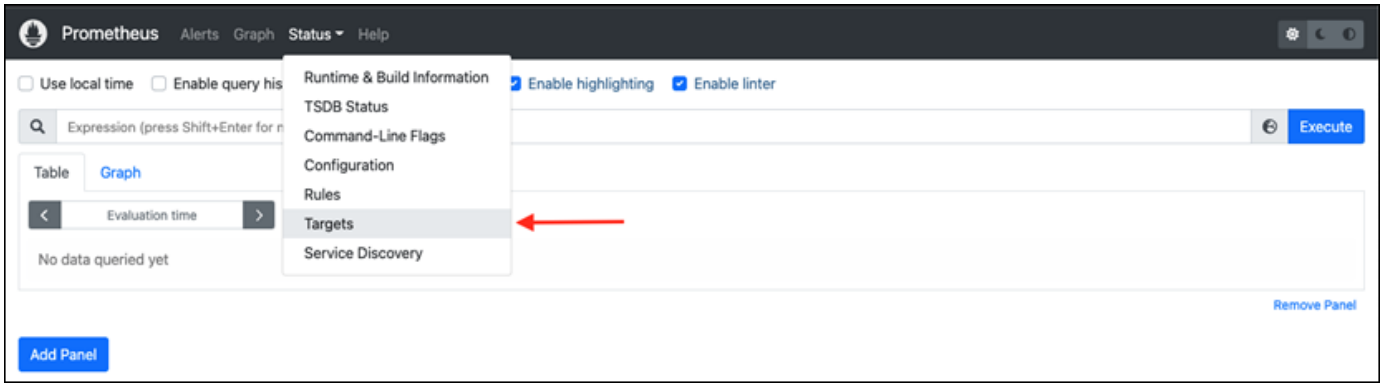
8. [Q] を押して、ステータスコマンドを終了します。
9. ローカルコンピュータで Web ブラウザを開き、次の Web アドレスにアクセスして Prometheus 管理インターフェイスを表示します。

```
http:<ip_addr>:9090
```

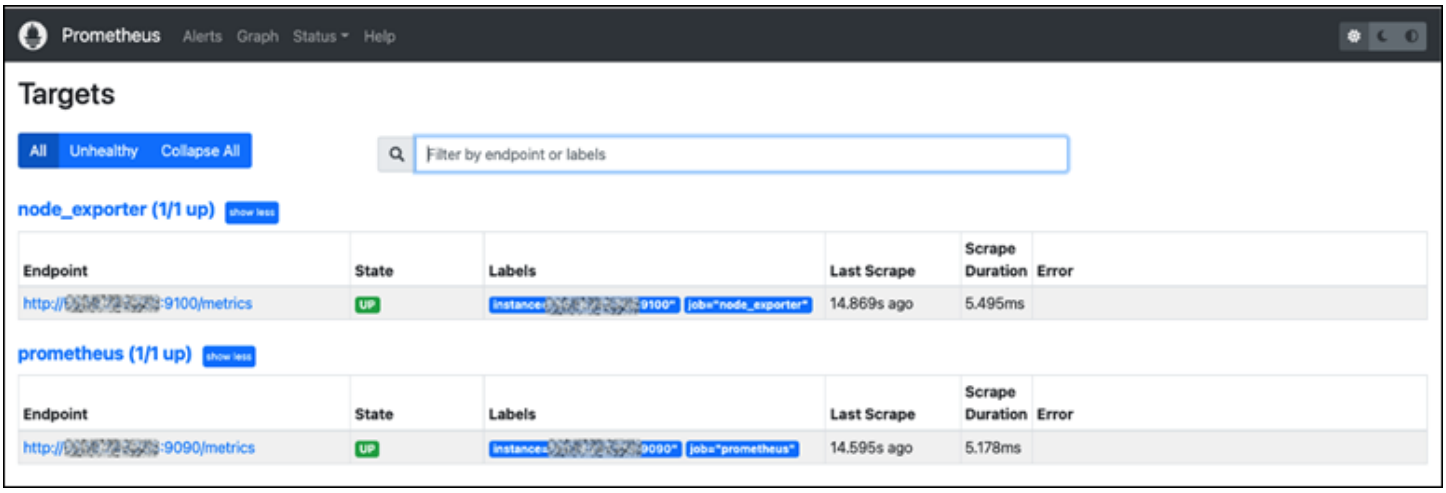
<ip_addr> を Lightsail インスタンスの静的 IP アドレスに置き換えます。次の例に示すようなダッシュボードが表示されます。



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。

scp を使用して Lightsail 上の Linux インスタンス間でファイルを転送する

Linux の Secure Copy (scp) コマンドを使用して、ローカルコンピュータから Linux または Unix インスタンスに、および Amazon Lightsail 内のあるインスタンスから別のインスタンスにファイルを転送します。scp コマンドの詳細については、man7 ウェブサイトの「scp(1) — Linux 手動ページ」を参照してください。

このチュートリアルでは、ある Lightsail インスタンスから別のインスタンスにファイルをコピーする手順について説明します。

内容

前提条件

- 両方のインスタンスのパブリック IP アドレスを持つ 2 つの Lightsail インスタンスが実行されています。インスタンスのパブリック IP アドレスを取得するには。Lightsail コンソール にサインインし、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。
- SSH キーペアを使用して両方のインスタンスにアクセスできます。詳細については、「Linux インスタンスに接続する」を参照してください。

ステップ 1: プライベートキー (.pem) ファイルをローカルコンピュータに保存する

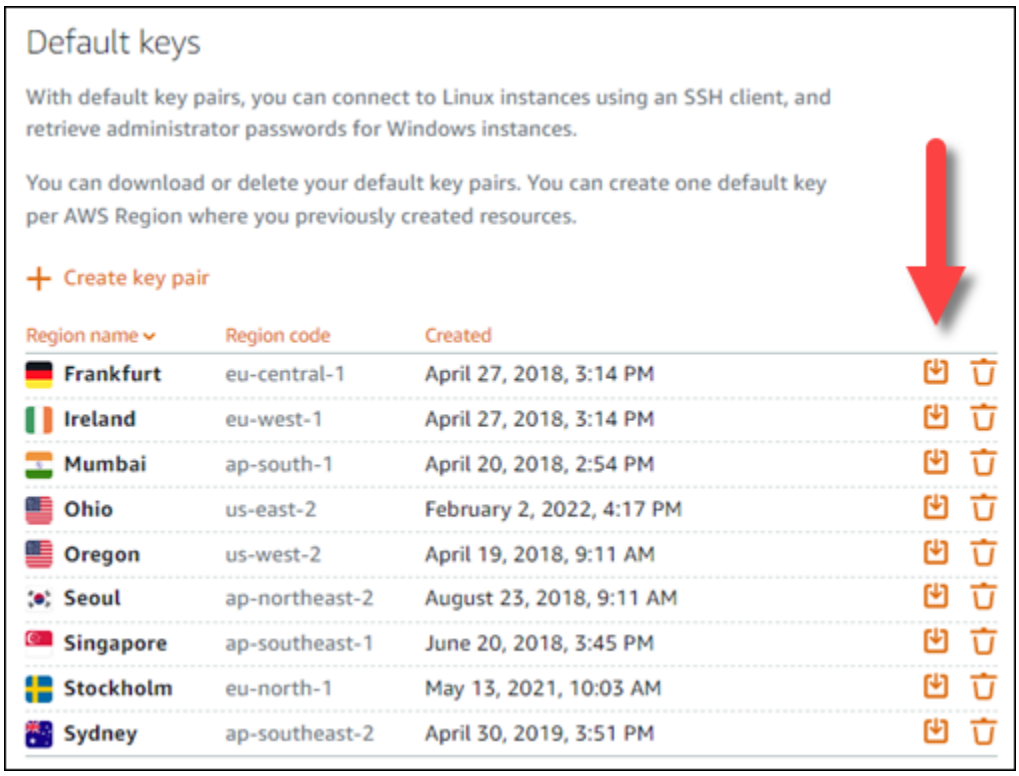
プライベートキー (.pem) ファイルをローカルコンピュータに保存するには、次のステップを実行します。ターゲットインスタンスのプライベートキーファイルは、あるインスタンスから別のインスタンスにファイルを安全に転送するために使用されます。同じ 内のインスタンス間でファイルをコピーするには AWS リージョン、そのリージョンのデフォルトキーを使用します。異なるリージョンのインスタンス間でファイルをコピーするには、ターゲットインスタンスがあるリージョンのデフォ

ルトキーを使用します。キーペアの詳細については、「」を参照してくださいSSH インスタンスへの接続。

> **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見つけて、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すると、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 scp を使用してインスタンスにファイルを転送することはできません。

プライベートキー (.pem) をローカルコンピュータに保存するには

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バーでユーザー名を選択し、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. ファイルを転送するインスタンス AWS リージョン がある 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。

   ダウンロードしたキーを、ユーザーのホームディレクトリの「Keys」フォルダなど、すべてのSSHキーを保存するディレクトリに移動することもできます。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。

ステップ 2: プライベートキーのアクセス許可を変更する

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。

プライベートキーファイルのアクセス許可を変更するには

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* 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
```

ステップ 3: プライベートキーをインスタンスに転送する

次の手順では、ローカルコンピュータから scp コマンドを実行して、プライベートキーをソースインスタンスに転送します。

scp を使用してコンピュータからソースインスタンスにプライベートキーを転送するには

1. コンピュータ上のプライベートキーファイルの場所と、インスタンスの送信先パスを決定します。次の例では、プライベートキーファイルの名前は です。*private-key.pem*、ソースインスタンスのユーザー名は です *ec2-user*、ソースインスタンスのIPv4アドレスは です *public-ipv4-address*、ソースインスタンスのIPv6アドレスは です。*public-ipv6-address*。 の *destination-path/* は、プライベートキーを転送するソースインスタンス上の場所です。

   > **Note**
   >
   > 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、無料BSD、オープンSUSEインスタンス: `ec2-user`
   > - 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 および WHM インスタンス: `centos`

- (IPv4) プライベートキーファイルをインスタンスに転送するには、コンピュータから次のコマンドを入力します。

```
scp -i /path/private-key.pem /path/private-key.pem ec2-user@public-ipv4-address:path/
```

- (IPv6) インスタンスに IPv6 アドレスしかない場合にプライベートキーファイルをインスタンスに転送するには、コンピュータから次のコマンドを入力します。IPv6 アドレスは角括弧 ([ ]) で囲む必要があります。括弧 () はエスケープする必要があります\。

```
scp -i /path/private-key.pem /path/private-key.pem ec2-user@\[public-ipv6-address\]:path/
```

2. を使用してインスタンスに接続していない場合はSSH、次のようなレスポンスが表示されます。

```
The authenticity of host 'ec2-198-51-100-1.compute-1.amazonaws.com (10.254.142.33)'
can't be established.
RSA key fingerprint is 1f:51:ae:28:bf:89:e9:d8:1f:25:5d:37:2d:7d:b8:ca:9f:f5:f1:6f.
Are you sure you want to continue connecting (yes/no)?
```

yes と入力します。

3. 転送が成功した場合、レスポンスは以下のようになります。

```
Warning: Permanently added 'ec2-198-51-100-1.compute-1.amazonaws.com' (RSA)
to the list of known hosts.
private-key.pem                                 100%   480     24.4KB/s   00:00
```

プライベートキーをソースインスタンスに転送したので、ターゲットインスタンスに安全に接続してファイルを転送することができます。次のステップに進み、その方法を確認してください。

ステップ 4: Lightsail Linux インスタンスと Unix インスタンス間でファイルを安全に転送する

次の手順では、あるインスタンス (ソースインスタンス) から scp コマンドを実行して、ファイルを別のインスタンス (ターゲットインスタンス) に転送します。

scp を使用してインスタンス間でファイルを転送するには

1. を使用してソースインスタンスに接続しますSSH。接続するには、ローカルコンピュータのターミナルプログラムを使用するか、Lightsail でブラウザベースのSSHクライアントを使用します。詳細については、「Linux インスタンスに接続する」を参照してください。
2. ソースインスタンス 上のファイルの場所と、ターゲットインスタンス 上の宛先パスを決定します。次の例では、プライベートキーファイルの名前は です。*private-key.pem*インスタンスのユーザー名は です。*ec2-user*、インスタンスのIPv4アドレスは です。*public-ipv4-address*、インスタンスのIPv6アドレスは です。*public-ipv6-address*。 の *destination-path/* は、ファイルを転送するターゲットインスタンス上の場所です。
   - (IPv4) ソースインスタンス からターゲットインスタンス にファイルを転送するには、ソースインスタンス から次のコマンドを入力します。

     ```
     scp -i /path/private-key.pem /path/my-file.txt ec2-user@public-ipv4-address:destination-path/
     ```

   - (IPv6) ソースインスタンス からターゲットインスタンス にファイルを転送するには、ソースインスタンス から次のコマンドを入力します。IPv6 アドレスは角括弧 ([ ]) で囲む必要があります。括弧 () はエスケープする必要があります\。

     ```
     scp -i /path/private-key.pem /path/my-file.txt ec2-user@\[public-ipv6-address\]:destination-path/
     ```

3. を使用してターゲットインスタンスにまだ接続していない場合はSSH、次のようなレスポンスが表示されます。

   ```
   The authenticity of host 'ec2-198-51-100-1.compute-1.amazonaws.com (10.254.142.33)'
   can't be established.
   RSA key fingerprint is 1f:51:ae:28:bf:89:e9:d8:1f:25:5d:37:2d:7d:b8:ca:9f:f5:f1:6f.
   Are you sure you want to continue connecting (yes/no)?
   ```

   yes と入力します。
4. 転送が成功した場合、レスポンスは以下のようになります。

   ```
   Warning: Permanently added 'ec2-198-51-100-1.compute-1.amazonaws.com' (RSA)
   to the list of known hosts.
   my-file.txt                                   100%   480     24.4KB/s   00:00
   ```

Lightsail を他の AWS サービスとVPCピアリングと統合する

Amazon Lightsail は、Amazon EC2や AWS などのサービスセット AWS Identity and Access Management を使用して、使用開始を容易にします。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。

Lightsail リソースは、Amazon VPCピアリングを介して他の AWS サービスと統合できます。VPC ピアリングをセットアップする方法について説明します。

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、クラウドでサイズ変更可能なコンピューティングキャパシティ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を使用すると、最小限の摩擦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 は、新しいサーバーインスタンスを取得して起動するのに必要な時間を数分にEC2短縮するため、コンピューティング要件の変化に応じて容量をすばやくスケールアップおよびスケールダウンできます。Amazon は、実際に使用した容量に対してのみ料金を支払うことができるようにすることで、コンピューティングの経済性EC2を変更します。Amazon EC2 は、障害耐性の高いアプリケーションを構築し、一般的な障害シナリオから分離するためのツールをデベロッパーに提供します。

Amazon の詳細をご覧くださいEC2。

Amazon VPC

Amazon Virtual Private Cloud (Amazon VPC) では、 クラウドの論理的に分離されたセクションを AWS プロビジョニングできます。ここでは、定義した仮想ネットワークで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 のネットワーク設定は簡単にカスタマイズできますVPC。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、複数のセキュリティレイヤーを活用して、各サブネットの Amazon EC2インスタンスへのアクセスを制御できます。

さらに、企業のデータセンターと の間にハードウェア仮想プライベートネットワーク (VPN) 接続を作成しVPC、企業のデータセンターの拡張として AWS クラウドを活用できます。

Amazon の詳細をご覧くださいVPC。

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。他のサービスから自動的にトリガーしたり、ウェブAWSやモバイルアプリから直接呼び出すようにコードを設定できます。

の詳細については、「」を参照してください AWS Lambda。

Amazon API Gateway

Amazon API Gateway は、開発者があらゆる規模APIsで簡単に作成、公開、保守、モニタリング、保護できるフルマネージドサービスです。を数回クリックするだけで AWS Management Console、アプリケーションAPIがバックエンドサービスからデータ、ビジネスロジック、または機能にアクセスするための「フロントドア」として機能する を作成できます。これには、Amazon で実行されているワークロードEC2、Lambda で実行されているコード、または任意のウェブアプリケーションが含まれます。Amazon API Gateway は、最大数十万の同時API通話の受け入れと処理に関連するすべてのタスクを処理します。これには、トラフィック管理、認可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 には最低料金やスタートアップコストはありません。料金は、受信したAPI通話と転送されたデータの量に対してのみ発生します。

Amazon API Gateway の詳細をご覧ください。

データベース

Amazon DynamoDB

Amazon DynamoDB は、あらゆる規模で一貫した 1 SQL桁のミリ秒単位のレイテンシーを必要とするすべてのアプリケーション向けの高速で柔軟なデータベースなしサービスです。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) を使用すると、クラウドでリレーショナルデータベースを簡単にセットアップ、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には、Amazon Aurora、Postgre、My、MariaDBSQL、Oracle、Microsoft SQL Server などSQL、使い慣れた 6 つのデータベースエンジンが用意されています。

Amazon の詳細をご覧くださいRDS。

Amazon Aurora

Amazon Aurora は、My SQL互換のリレーショナルデータベースエンジンで、ハイエンドの商用データベースのスピードと可用性を、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と組み合わせます。Aurora は、商用データベースのセキュリティ、可用性、信頼性を 10 分の 1 のコストで備えており、MySQL よりも最大 5 倍優れたパフォーマンスを提供します。

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インスタンス間のトラフィックの単純な負荷分散に最適です。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、同じ Amazon EC2インスタンス上の複数のポート間で負荷分散したりできます。

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 Elastic Load Balancing サービスの負荷分散オプションであり、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または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 サービスを使用すると、AWSクラウド内のリアルタイムのストリーミングデータを簡単に操作できます。Amazon Kinesis サービスには、大量のストリーミングデータを に簡単にロードする Amazon Data FirehoseAWS、標準 でストリーミングデータを分析する Amazon Managed Service for Apache FlinkSQL、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 などがあります。

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、費用対効果の高い方法で処理できるマネージド Hadoop フレームワークを提供します。また、Amazon で Apache Spark、HBase、Presto、Flink などの一般的な分散フレームワークを実行したりEMR、Amazon S3 や DynamoDB などの他のAWSデータストアのデータとやり取りしたりすることもできます。

Amazon は、ログ分析、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など、幅広いビッグデータユースケースをEMR安全かつ確実に処理します。

Amazon の詳細をご覧くださいEMR。

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IAM、Amazon EC2や などの他の AWS のサービス、クラウドデータ移行サービス、および初期または継続的なデータ取り込みのためのゲートウェイと組み合わせて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インスタンスで使用する永続的なブロックストレージボリュームを提供します。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、コンポーネントの障害から保護され、高可用性と耐久性を実現します。Amazon EBS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提供します。Amazon を使用するとEBS、プロビジョニングした分だけ低価格で済むため、数分以内に使用量をスケールアップまたはスケールダウンできます。

Amazon の詳細をご覧くださいEBS。

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、 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AWS。 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、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。を使用すると CloudWatch 、システム全体のリソース使用率、アプリケーションのパフォーマンス、運用状態を可視化できます。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、Node.jsPHP、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、 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 easy-to-use するためのサービスですIIS。

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、アプリケーションを強化するAWSリソースを完全に制御し、基盤となるリソースにいつでもアクセスできます。

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、Docker コンテナをサポートする非常にスケーラブルで高性能なコンテナ管理サービスで、Amazon EC2インスタンスのマネージド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 では、独自のクラスター管理インフラストラクチャをインストール、運用、スケーリングする必要がありECSません。簡易API呼び出しを使用すると、Docker 対応アプリケーションを起動および停止し、クラスターの完全な状態をクエリし、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、使い慣れた多くの機能にアクセスできます。Amazon を使用してECS、リソースのニーズと可用性の要件に基づいて、クラスター全体のコンテナの配置をスケジュールできます。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 の詳細をご覧くださいECS。

セキュリティとユーザーサインイン

AWS Identity and Access Management (IAM)

IAM では、ユーザーの AWS サービスとリソースへのアクセスを安全に制御できます。を使用するとIAM、AWSユーザーとグループを作成および管理し、アクセス許可を使用してAWSリソースへのアクセスを許可または拒否できます。

の詳細については、「」を参照してくださいIAM。

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 では、Facebook、Twitter、Amazon などのソーシャル ID プロバイダー、SAMLID ソリューション、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オプションもあります。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、企業が安全で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ようにするフルマネージド型のソースコントロールサービスです。 AWS CodeCommit は、独自のソース管理システムを運用したり、インフラストラクチャのスケーリングを心配したりする必要性を排除します。を使用して AWS CodeCommit 、ソースコードからバイナリまで、あらゆるものを安全に保存でき、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、スケーラブルでフル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では、クラウドアプリケーションのコンポーネントを簡単かつ費用対効果の高い方法で切り離すことができます。Amazon を使用するとSQS、メッセージが失われたり、他のサービスが常に利用可能になったりすることなく、任意の量のデータを送信できます。Amazon SQS には、高いスループットと at-least-once 処理機能を備えた標準キュー、および FIFO (先入れ先出し) 配信と 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 を使用するとSQS、可用性の高いメッセージングクラスターを運用およびスケーリングする管理上の負担を軽減しながら、使用する分だけ低価格で支払うことができます。

Amazon の詳細をご覧くださいSQS。

Amazon SNS

Amazon Simple Notification Service (Amazon SNS) は、高速で柔軟、フルマネージド型のプッシュ通知サービスで、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 では、モバイルデバイスのユーザーや E メールの受信者にプッ

シュ通知を送信したり、他の分散サービスにメッセージを送信したりすることが簡単で費用対効果が高いです。

Amazon ではSNS、Baidu Cloud Push を使用して、Apple Push Notification Service (APNS)、Google Cloud Messaging (GCM)、Fire OS、Windows デバイス、および中国の Android デバイスに通知を送信できます。Amazon を使用してSNS、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。

これらのエンドポイント以外にも、Amazon SNSは 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することもできます。

Amazon の詳細をご覧くださいSNS。

Amazon SES

Amazon Simple Email Service (Amazon SES) は、Amazon.com が独自の顧客基盤を提供するために開発した、信頼性が高くスケーラブルなインフラストラクチャ上に構築された費用対効果の高い E メールサービスです。Amazon ではSES、最低限のコミットメントなしで 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 の詳細をご覧くださいSES。

ワークフロー

Amazon Simple Workflow Service (Amazon SWF)

Amazon SWF は、デベロッパーが並列またはシーケンシャルステップを持つバックグラウンドジョブを構築、実行、スケーリングするのに役立ちます。Amazon はSWF、クラウド内のフルマネージド型のステートトラッカーおよび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 がお手伝いします。

Amazon の詳細をご覧くださいSWF。

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することができます。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、携帯電話などの大規模市場向け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 AppStream は、クラウドからアプリケーションをストリーミングSDKするための を提供しています。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、ビジネスニーズを満たす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。

で Lightsail リソースを作成する AWS CloudFormation

Amazon Lightsail は AWS CloudFormation、 AWS リソースとインフラストラクチャの作成と管理に費やす時間を短縮できるように、リソースのモデル化とセットアップに役立つサービスである 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) を記述するテンプレートを作成し、それらのリソースを AWS CloudFormation プロビジョニングして設定します。

を使用すると AWS CloudFormation、テンプレートを再利用して Lightsail リソースを一貫して繰り返しセットアップできます。リソースを 1 回記述し、複数の AWS アカウント およびリージョンで同じリソースを何度もプロビジョニングします。

Lightsail と AWS CloudFormation 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 を理解する必要があります。テンプレートは、JSON や 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は、 AWS CloudFormation スタックでプロビジョニングするリソースを記述します。JSON または YAML に慣れていない場合は、 AWS CloudFormation デザイナー を使用してテンプレートの使用を開始 AWS CloudFormation できます。詳細については、「 ユーザーガイド」の AWS CloudFormation 「デザイナーとは」を参照してください。 AWS CloudFormation

Lightsail は、AWS でのインスタンスとディスクの作成をサポートしています AWS CloudFormation。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。

の詳細 AWS CloudFormation

の詳細については AWS CloudFormation、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

アプリデプロイ用の Lightsail リソースを詳しく見る

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 による Amazon Lightsail インスタンスのヘルスのモニタリング

  2022 年 3 月 30 日 – Datadog による 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とコストの管理にどのように役立つかについて説明します。
- Amazon Lightsail AWS の使用に関する調査のために Galaxy をセットアップする方法

  2022 年 1 月 13 日 – 科学ワークフロー、データ統合、デジタル保存プラットフォームである Galaxy を Lightsail にデプロイします。
- 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使用量のモニタリング

  2021 年 6 月 14 日 – モニタリング、アラーム、通知 CloudWatch のためにメモリ使用量を Amazon に送信するように 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された 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して Lightsail にデプロイ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する方法。

- Amazon Lightsail コンテナを使用した WordPress ウェブサイトの起動

  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 ウェブサイトを起動します。

- Lightsail コンテナ: クラウドでコンテナを簡単に実行する方法

  2020 年 11 月 13 日 – コンテナベースのワークロードを Lightsail にデプロイします。

- 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 で本番環境を設定し、Lightsail からその環境にウェブサービスを移行します。

- Amazon Lightsail インスタンスで実行する 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によるウェブサイトのパフォーマンスの向上

  2020 年 7 月 23 日 – 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 WordPress。

- Amazon Lightsail インスタンスのシステムパフォーマンスを積極的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による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 および 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 – 変更を CodePipeline にプッシュするたびに、 と CodeDeploy連携し、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 GitHub。

- 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シンプルな Node.js ウェブアプリケーションをロードバランシングする方法。

- Amazon Lightsail で Ghost を使用して写真日記を作成する

2020 年 3 月 23 日 – Ghost on Lightsail 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 に高可用性 WordPress サイトをデプロイする、パート 1: で高可用性 Lightsail データベースを実装する WordPress

2019 年 10 月 22 日 – Lightsail で高可用性 WordPressサイトを構築する、パート 1。

- Amazon Lightsail に高可用性 WordPress サイトをデプロイする、パート 2: で Amazon S3 を使用してメディアファイルを安全に配信 WordPress する Amazon Lightsail

2019 年 10 月 31 日 – Lightsail で高可用性 WordPressサイトを構築する、パート 2。

- Amazon Lightsail での高可用性 WordPress サイトのデプロイ、パート 3: Amazon を使用したセキュリティとパフォーマンスの向上 CloudFront

2019 年 11 月 7 日 – Lightsail で高可用性 WordPressサイトを構築する、パート 3。

- Amazon Lightsail での高可用性 WordPress サイトのデプロイ、パート 4: Lightsail ロードバランサーによるパフォーマンスとスケーラビリティの向上 Amazon Lightsail

2019 年 11 月 14 日 – Lightsail で高可用性 WordPress サイトを構築する、パート 4。

- Amazon Lightsail を使用してサービスとしてのプラットフォームを構築する

2019 年 10 月 8 日 – Lightsail にポケットプラットフォームを組み立てます。

- Amazon Lightsail を使用した Nginx ベースの HTTP/HTTPS ロードバランサーのデプロイ

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。

- を初めて AWS クラウド使用する場合 Amazon Lightsail が役立つ

2019 年 3 月 27 日 – Amazon Lightsail の開始方法。

- 新規 — Amazon Lightsail 用の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: より多くのインスタンスサイズと料金削減

  2018 年 8 月 23 日 – Lightsail インスタンスの概要。

- Amazon Lightsail : の能力 AWS、VPS のシンプルさ

  2016 年 11 月 30 日 – Lightsail のローンチに関するお知らせ。

チュートリアル

上位 5 位の実践チュートリアル:

1. ロードバランシングされた WordPress ウェブサイトを作成する

   2021 年 9 月 8 日 – Lightsail で高可用性 WordPressウェブサイトを起動します。

2. Amazon Lightsail による WordPress ウェブサイトの移行と管理

   2021 年 2 月 22 日 – シーウマソフトウェアを使用してウェブサイトの WordPressクローンを Lightsail に起動します。

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 および 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、cPanel インスタンスと WHM インスタンスが Lightsail で起動および実行され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 を設定および設定する方法

  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。

- 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 に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 バケットとディストリビューションに接続する

  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 PrestaShop 認定ブループリントを使用して Lightsail インスタンスを起動して設定します。

- Amazon Lightsail で Amazon EFS を使用する方法 Amazon Lightsail

  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 – Lightsail コンテナで Flask アプリケーションを提供する方法について説明します。

- Amazon Lightsail を使用した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で動作するように 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。

- 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 インストールされた でウェブサイトを起動して実行します。

- 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 で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

  202July14 日 – PrestaShop 認定 Bitnami ブループリントを使用して Lightsail インスタンスを起動し、設定します。

- Amazon Lightsail に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 に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 チュートリアルの概要

  2020 年 8 月 31 日 – Lightsail でクラウドジャーニーを始めましょう。

- Amazon Lightsail : の使用を開始する最も簡単な方法 AWS

  2020 年 3 月 20 日 – Lightsail は、 の使用を開始する最も簡単な方法です AWS。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 でのマルチサイトの設定 WordPress

  2019 年 1 月 15 日 – Lightsail でマルチサイトインスタンスを設定する WordPress方法について説明します。

- Lightsail の管理

  2018 年 10 月 9 日 – Lightsail の主な機能を簡単にご覧ください。

- Amazon Lightsail に 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 に WordPress インスタンスをデプロイする

2018 年 6 月 5 日 – Lightsail に WordPress インスタンスをデプロイします。

Lightsail の請求と使用状況の詳細を表示する

Amazon Lightsail の請求は、Amazon Web Services (AWS) の請求を通じて処理されます。Lightsail の請求書を表示するには、AWS Billing and Cost Management ダッシュボード に移動するか、Lightsail コンソールの上部のナビゲーションバーで請求を選択します。料金の詳細については、「Lightsail の料金」ページを参照してください。

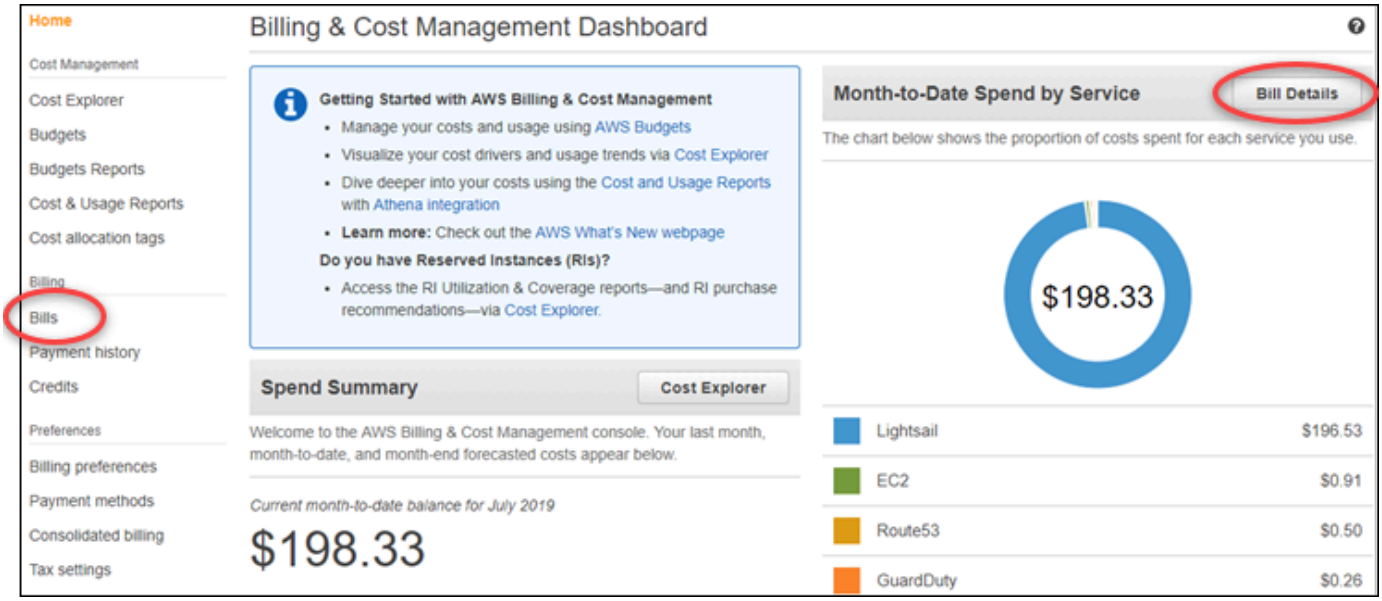
Lightsail の請求書の詳細を表示する

毎月の Lightsail 請求書の詳細な内訳を表示するには：

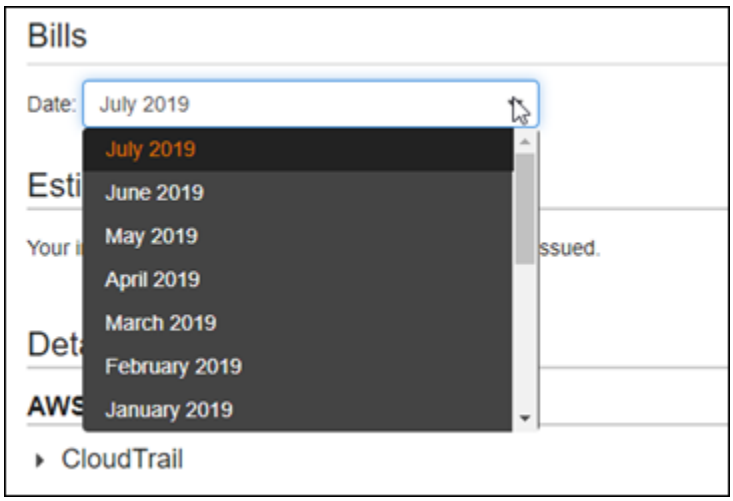
1. AWS Billing and Cost Management ダッシュボードにサインインします。

   請求ダッシュボードのホームページには、請求の概要 month-to-date が表示されます。

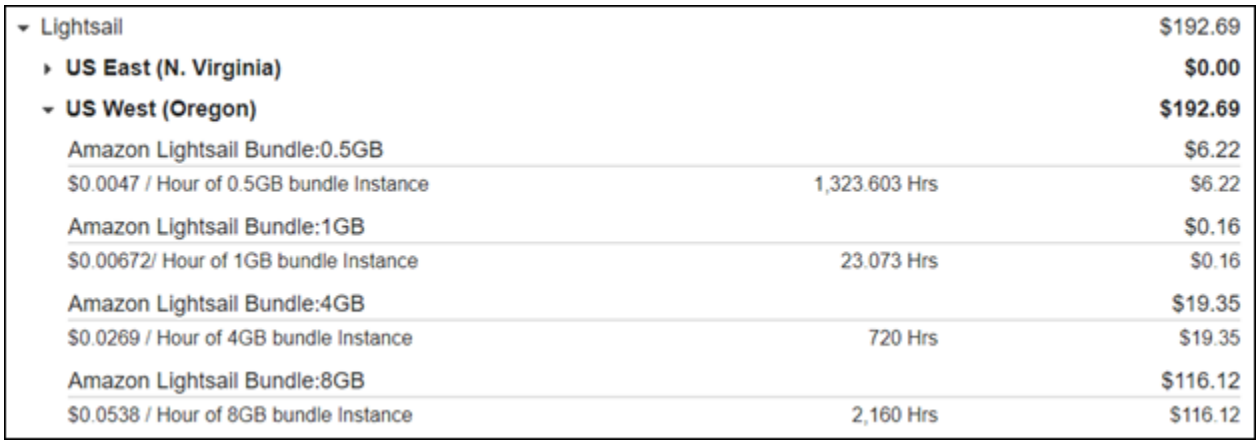
2. 月額の料金の詳細バージョンを表示するには、ダッシュボードのホームページで [料金明細] を選択するか、左側のナビゲーションペインで [請求] を選択します。



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. 請求書ページを下にスクロールし、Lightsail 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

次のリストでは、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について説明します。これらの使用タイプは、Lightsail リソースの月額請求の料金を特定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: 使用された 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。例えば、4GB のメ

モリが指定されている場合、1 か月USDあたり 24 USD 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。例えば、4GB のメモリが指定されている場合、1 か月USDあたり 44 USD 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。例えば、4GB のメモリが指定されている場合、60 USDUSD/月の標準データベースプランの請求時間が表示されます。
- Amazon Lightsail RelationalDatabase:SizeGB (高可用性): 使用された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。例えば、4GB のメモリが指定されている場合、120 USDUSD/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： 使用されたブロックストレージディスクの量 (1 か月あたりのギガバイト単位) 。
- Amazon Lightsail DNS- クエリ： その月のDNSクエリの数 (カウント) 。
- Amazon Lightsail Load Balancer : 使用された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 (1 か月あたりのギガバイト単位) 。
- Amazon Lightsail Region-UnusedStaticIP: アタッチされていない静的な の量 IPs (時間単位) 。
- Amazon Lightsail RegionTotalDataXfer-In-Bytes: 転送されたデータの合計量 (ギガバイト単位) 。
- Amazon Lightsail RegionTotalDataXfer-Out-Bytes: 転送されたデータの合計量 (ギガバイト単位) 。
- Amazon Lightsail Region-DataXfer-Out-Overage-Bytes: インターネットまたはパブリックに転送されIPs、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量を超えるデータの量 (ギガバイト単位) 。

請求のリージョンコード

Lightsail の請求および使用状況レポートは、コードと略語を使用します。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 でよくある質問への回答を取得する

このセクションでは、Lightsail に関連する一般的な質問と回答を、以下のカテゴリ別にまとめています。

トピック

Lightsail に関するよくある質問への詳細な回答については、各カテゴリに記載されているリンクを参照してください。

Lightsail とそのグローバル可用性について説明します。

Amazon Lightsail とは

Amazon Lightsail は、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ソリューションを必要とするデベロッパー、中小企業、学生、その他のユーザー AWS にとって、 の

使用を開始する最も簡単な方法です。Lightsail は、デベロッパーにコンピューティング、ストレージ、ネットワーク容量を提供します。Lightsail には、仮想マシン、コンテナ、データベース、、ロードバランサーCDN、DNS管理など、プロジェクトをすばやく起動するために必要なものがすべて含まれており、月額料金が予測可能です。

Lightsail で何ができますか？

アプリケーションを簡単にデプロイおよび管理するためのすべてを含む事前設定済みの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とオペレーティングシステムのセキュリティとヘルスが Lightsail によって管理されるデータベースを作成したりできます。Lightsail は、数十個以下のインスタンスを必要とするプロジェクトや、シンプルな管理インターフェイスを好むデベロッパーに最適です。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 コマースサイトの実行などがあります。プロジェクトが大きくなるにつれて、インスタンスでロードバランサーとアタッチされたブロックストレージを使用して、冗長性と稼働時間を向上させ、多数の他の AWS サービスにアクセスして新しい機能を追加できます。

Lightsail は を提供していますAPIか？

はい。Lightsail コンソールで行ったことはすべて、公開されている によってバックアップされますAPI。Lightsail と をインストールCLIして使用する方法について説明しますAPI。

Lightsail にサインアップするにはどうすればよいですか？

Lightsail の使用を開始するには、開始してログインを選択します。Amazon Web Services アカウントを使用して Lightsail にアクセスします。まだ持っていない場合は、作成するよう求められます。

Lightsail AWS リージョン はどの で利用できますか？

Lightsail は現在、次の で利用できます AWS リージョン。

AWS リージョン

- 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、AWS リージョン Lightsail の「」および「アベイラビリティーゾーン」を参照してください。

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 サービスクォータとは

引き上げることができるクォータを含む最新の Lightsail サービスクォータについては、「」の「Lightsail サービスクォータ」を参照してくださいAWS 全般のリファレンス。サービスクォータを引き上げるには、 でケースを開きますAWS Support。

より詳細なヘルプを得るにはどうすればよいですか?

Lightsail のコンテキスト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。ヘルプパネルを開くには、Lightsail コンソールの右上隅にあるヘルプパネルアイコン i を選択します。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。 https://lightsail.aws.amazon.com/ls/docs/overviewまたAPI、Lightsail または を使用する場合 AWS CLI、Lightsail にはサポートされているすべてのプログ

ラミング言語の完全なAPIリファレンスがあります。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 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については、Lightsail のドキュメントとサポートフォーラム を参照してください。

さらに、 は、個々のニーズを満たすための有料プランの配列 AWS Support を提供します。

請求とアカウント管理

Lightsail プランの料金はいくらですか？

Lightsail プランはオンデマンドの時間料金で請求されるため、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、固定時間料金が最大月額プランコストまで請求されます。最も安価な Lightsail プランは、1 時間USDあたり 0.0067 USD (1 か月USDあたり 5 USD) 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USDあたり 0.0127 USD (1 か月USDあたり 9.50 USD) から開始します。

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、その月の 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、按分計算されたコストのみが請求されます。例えば、最も安価な Lightsail インスタンスプランを 1 か月に 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。既存または新規の AWS いずれのお客様でも、5 ドルの USD Lightsail プランは 750 時間無料で利用できます。また、9.50 USD の Windows プランを使用して、Windows Server ライセンスを含む Lightsail USD プランを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。例えば、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。無料トライアルオファーは、Lightsail の使用にサインアップしてから最初の暦月内の使用にのみ適用されます。アカウントが組織にリンクされている場合 (AWSOrganizations の下) 、組織内の 1 つのアカウントのみが AWS 無料利用枠 オファーの恩恵を受けることができます。

> **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 を無料で使い始めることができます。詳細については、Amazon Lightsail の料金」ページのAWS 「無料利用枠」を参照してください。

Lightsail 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起動されたときに開始されます。

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 の使用を開始した新規または既存の AWS 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のコストはいくらですか？

Lightsail マネージドデータベースには 4 つのプランサイズがあり、40 1GB のSSDストレージと 100 GB のデータ転送許容量を持つ 1 GB RAM データベースインスタンスの場合、1 か月USDあたり 15 USD から始まります。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 の新規お客様は、15 USD の Lightsail USD プランの 1 か月分が無料になります。

Lightsail ブロックストレージのコストはいくらですか？

Lightsail ブロックストレージの料金は、1 GB USD あたり 1 か月あたり 0.10 USD です。

Lightsail ロードバランサーのコストはいくらですか？

Lightsail ロードバランサーの料金は 1 か月USDあたり 18 USD です。

証明書管理の課金対象を教えてください。

Lightsail 証明書と証明書管理は、Lightsail ロードバランサーの使用で無料です。

Lightsail 静的IPv4アドレスのコストはいくらですか？

Lightsail インスタンスにアタッチされた静的 IP アドレスに関連するコストはありません。静的 は のみのインスタンスにIPv6アタッチIPs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するのに役立つように努めているため、インスタンスにアタッチIPsされていない静的 に対して 1 時間USD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) のディストリビューションプランには、データ転送許容量が含まれています。

Lightsail インスタンスの場合、インスタンスへ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OUT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 アドレスを使用する AWS リソースへの超過データ転送に対してのみ課金されます。Lightsail インスタンスへの超過データ転送に対しては課金されません。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、Lightsail インスタンスへのデータ転送 IN と Lightsail インスタンスOUTからのデータ転送の両方が、データ転送許容量を超えて無料です。

Lightsail マネージドデータベースOUTの場合、データ転送のみが許容量に対してカウントされます。データ転送許容量を超えた場合、Lightsail マネージドデータベースOUTからインターネットへのデータ転送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の場合、ディストリビューションからのすべてのデータ転送は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。

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。インスタンスOUTのデータ転送 IN と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OUT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 アドレスを使用する AWS リソースへの超過データ転送に対してのみ課金されます。Lightsail インスタンスへの超過データ転送に対しては課金されません (例 1 を参照) 。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。データ転送許容量は、リージョン内の同じバンドル (bundleId) のインスタンスについて集計されます (例 2 と例 3 を参照) 。データ転送許容量は、同じサイズの IPv4 および IPv6 インスタンスに対しても集計されます

(例 4 を参照) 。インスタンスを削除して新しいインスタンスを作成しても、データ転送許容量はリセットされません (例 5 を参照) 。

Lightsail バンドルの詳細については、「Amazon Lightsail APIリファレンス」の「バンドル」を参照してください。 Amazon Lightsail

- 例 1 – 1 か月USDあたり 5 USD のインスタンスバンドル (bundleId nano_3_0) があり、1 か月あたり 1 TB のデータ転送許容量があります。500 GB のデータをインターネットに送信 (データ転送 OUT) し、400 GB のデータをインスタンスに送信 (データ転送 IN) すると、900 GB の 1 TB の許容量を消費したことになります。さらに 200 GB のデータをインターネットに送信すると、許容量を 100 GB 超過し、100 GB のデータ転送OUT超過料金が請求されます。次に 200 GB のデータをインスタンスに送信する場合、超過に対して課金されません。
- 例 2 – リージョンで 1 か月USDあたり 2 つの 5 USD のインスタンスバンドル (bundleId nano_3_0) があり、それぞれに 1 か月あたり 1 TB のデータ転送許容量がある場合、合計 2 TB のデータ転送許容量が得られます。1 つ目のインスタンスで 1.5 TB のデータをインターネットに送信し、2 つ目のインスタンスで 100 GB のデータをインターネットに送信する場合でも、合計許容量である 2 TB で 400 GB となり、データ転送OUT超過料金は請求されません。
- 例 3 – インスタンスバンドルの 2 つのセットを作成します。米国西部 (bundleIdオレゴンnano_3_0) リージョンでは、1 か月USDあたり 2 USD のインスタンスバンドル () で A を設定し、1 か月USDあたり 3 USD のインスタンスバンドル () bundleId micro_3_0で B を設定します。これにより、集合 A に対して 2 TB のデータ転送許容量、集合 B に対して 6 TB のデータ転送許容量が得られます。集合 A インスタンスを介してインターネットに 3 TB のデータを、集合 B インスタンスを介してインターネットに 4 TB のデータを転送した場合、集合 A インスタンスのデータ転送許容量を超え、1 TB のデータ転送OUT超過料金が請求されます。セット B インスタンスの許容量は 2 TB です。
- 例 4 – 請求月の最初の 20 日以内に、1 か月USDあたり 3.50 USD のIPv6インスタンスバンドル (bundleId nano_ipv6_3_0) の 1 TB データ転送許容量の合計 600 GB を消費しました。インスタンスのネットワークタイプを 21 日にデュアルスタック (bundleId月USD額 5 USD nano_3_0の料金) に切り替えることにしました。その月のデータ転送使用率はリセットされず、600 GB のままになり、400 GB の許容量は残ります。請求月の残りの期間に 500 GB のデータをインターネットに送信すると、100 GB のデータ転送OUT超過料金が発生します。
- 例 5 – インスタンスバンドル (bundleId nano_3_0) は 1 か月USDあたり 3 USD で、それぞれ 1 TB のデータ転送許容量があります。請求月内に合計 3 TB のデータ転送許容量の 1 TB を消費し、残りのデータ転送許容量の 2 TB を残したとします。すべてのインスタンスを削除し、同じ請求月内に同じリージョンに同じバンドル (bundleId nano_3_0) の 3 つの新しいインスタンスを作成する場合、データ転送使用率は引き続き 1 TB になり、残りのデータ転送許容量は引き続き 2 TB に

なります。データ転送OUT超過料金が発生する前は、同じ月内にインスタンスを介してさらに 2 TB のデータを転送できます。

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。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。これは、ロードバランサーの背後ではない Lightsail インスタンスのデータ転送許容量にインターネットに出入りするトラフィックがカウントされるのと同じです。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。インスタンスがプランのデータ転送許容量を超える場合は、使用したデータ転送 (インターネットOUTへのデータ転送のみ) の GB ごとに超過料金が課金されます。

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。Lightsail インスタンスOUT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のデータ転送、またはプライベート IP アドレスが使用されている場合は同じリージョンの AWS 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月間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、Lightsail インスタンスOUTからインターネット、別の AWS リージョン 、または同じリージョンの AWS リソースへのデータ転送に対して課金されます。無料利用枠を超えるこれらのタイプのデータ転送の料金は、次のとおりです。

- 米国東部 (オハイオ) (us-east-2): 0.09 USDUSD/GB
- 米国東部 (バージニア北部) (us-east-1): 0.09 USDUSD/GB
- 米国西部 (オレゴン) (us-west-2): 0.09 USDUSD/GB
- アジアパシフィック (ムンバイ) (ap-south-1): 0.13 USDUSD/GB
- アジアパシフィック (ソウル) (ap-northeast-2): 0.13 USDUSD/GB
- アジアパシフィック (シンガポール) (ap-southeast-1): 0.12 USDUSD/GB

- アジアパシフィック (シドニー) (ap-southeast-2): 0.17 USDUSD/GB
- アジアパシフィック (東京) (ap-northeast-1): 0.14 USDUSD/GB
- カナダ (中部) (ca-central-1): 0.09 USDUSD/GB
- 欧州 (フランクフルト) (eu-central-1): 0.09 USDUSD/GB
- 欧州 (アイルランド) (eu-west-1): 0.09 USDUSD/GB
- 欧州 (ロンドン) (eu-west-2): 0.09 USDUSD/GB
- 欧州 (パリ) (eu-west-3): 0.09 USDUSD/GB
- 欧州 (ストックホルム) (eu-north-1): 0.09 USDUSD/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ディストリビューション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に対して課金されますOUT。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ます。

- アジアパシフィック: 0.13 USDUSD/GB
- カナダ: 0.09 USDUSD/GB
- 欧州: 0.09 USDUSD/GB
- インド: 0.13 USDUSD/GB
- 日本: 0.14 USDUSD/GB
- 中東: 0.11 USDUSD/GB
- 南アフリカ: 0.11 USDUSD/GB
- 南米: 0.11 USDUSD/GB
- 米国: $0.09 USD/GB

インスタンスのデータ転送許容量は によってどのように異なりますか AWS リージョン？

Lightsail インスタンスのリージョンデータ転送許容量は、Amazon Lightsail の料金表に記載されています。許容量は、アジアパシフィック (ムンバイおよびシドニー) リージョンを除き AWS リージョ

ン、すべての で同じです。ムンバイおよびシドニーリージョンのプランには、他のリージョンのデータ転送許容量の半分が含まれます。

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 で同じです AWS リージョン。

Lightsail ドメインのコストはいくらですか？

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管理のコストはいくらですか？

DNS Lightsail 内では 管理は無料です。最大 6 つのDNSゾーンと、DNSゾーンごとに必要な数のレコードを作成できます。また、ゾーンに対する毎月のDNSクエリ許容量は 300 万です。1 か月で最初の 300 万件のクエリを超えると、100 万件のDNSクエリUSDあたり 0.40 USD が課金されます。

Lightsail スナップショットの料金は？

Lightsail スナップショット (手動および自動) の保存には USD1 か月あたり 0.05 USD かかります。つまり、28 GB の容量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、それを 1 か月間保持すると、その月USDに対して 1.40 USD の料金が発生します。

同じインスタンスの連続する複数のスナップショットを作成すると、Lightsail は自動的にスナップショットのコストを最適化します。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。上記の例では、インスタンスデータが 2 GB しか変更されない場合、2 番目のインスタンススナップショットのコストは 1 か月USDあたり 0.10 USD にすぎません。

AWS アカウントを管理するにはどうすればよいですか？

Lightsail は AWS のサービスであり、 AWS 信頼され、実績のあるクラウドインフラストラクチャ上で実行されます。Lightsail と にログインするには、同じ AWS アカウントと認証情報を使用します AWS Management Console。

AWS 請求情報AWS とコスト管理コンソール から、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 アカウントを管理できます。

Lightsail の法的利用規約は何ですか？

Lightsail は Amazon ウェブサービスであるため、Lightsail を使用するには、まず AWS カスタマーアグリーメント および 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ライセンス契約も適用され、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。

請求情報とコスト管理コンソールから AWS 請求の支払いと管理を行うことができます。 はほとんどの主要なクレジットカード AWS を受け入れます。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。

ブロックストレージ (ディスク)

Lightsail ブロックストレージで何ができますか？

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 インスタンスにアタッチ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 () を使用しますSSD。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のバランスをとり、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スループットを必要とするアプリケーション、または MongoDB や 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ご利用のお客様は、Lightsail の代わりに GP2または プロビジョンドIOPSSSDストレージEC2で Amazon 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 Lightsail プランに含まれているストレージとどのように異なりますか？

Lightsail プランに含まれてい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?

アタッチされた各ディスクは最大 16 TB で、Lightsail アカウントのアタッチされたブロックストレージの合計量は 20 TB を超えることはできません。

Lightsail インスタンスごとにアタッチできるディスクの数

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を提供しますか ?

はい。データの安全性を維持するために、Lightsail がユーザーに代わって管理するキーを使用して、Lightsail にアタッチされたすべてのディスクとディスクスナップショットがデフォルトで保管時に暗号化されます。Lightsail は、Lightsail インスタンスとアタッチされたディスク間を移動するデータの暗号化も提供します。

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、データの定期的なバックアップが可能です。

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。

証明書

Lightsail でプロビジョニングされた証明書の使用方法

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、検証済みトラフィックを安全な AWS 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 終了を提供します。Lightsail 証明書は、個々の Lightsail インスタンスではなく、Lightsail ロードバランサーでのみ使用できます。

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、アクセス権を持っていることを検証して、身分証明書を提供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 は検証するドメインのDNSゾーンにCNAMEレコードを追加するよう促すプロンプトを表示します。現在DNSゾーンを管理している場所、つまり Lightsail DNS管理プロバイダーまたは外部DNSホスティングプロバイダーにCNAMEレコードを追加するのに 72 時間かかります。

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、ユーザーまたは組織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 HTTPS対応のロードバランサーを使用することはできません。

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 は、SSL/TLS 証明書のマネージド更新を提供します。つまり、Lightsail は有効期限が切れる前に証明書を自動的に更新しようとします。ユーザーによるアクションは必要ありません。Lightsail 証明書を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 が提供する証明書をダウンロードできますか ?

いいえ。Lightsail 証明書は Lightsail アカウントにバインドされ、Lightsail の外部で削除して使用することはできません。

連絡先とモニタリング通知

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、指定したアドレスに送信される E 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することができます。E メールとSMSテキストメッセージで通知を受け取るには、リソースをモニタリングする各 AWS リージョン で、E 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。

連絡先はいくつ追加できますか?

リソースをモニタリングする各 AWS リージョン に、1 つの E メールアドレスと 1 つの携帯電話番号を追加できます。SMS テキストメッセージは、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、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。

コンテナサービス

Lightsail コンテナサービスで何ができますか？

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できます。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。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) を指定するだけです。Lightsail は、基盤となるインフラストラクチャを管理することなく、コンテナサービスの実行を処理します。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方法を教えてください。

Amazon ECR Public Registry などのオンラインパブリックレジストリ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構築して、 を使用して簡単なステップで Lightsail にプッシュすることもできます 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、パブリックコンテナレジストリのみが Lightsail コンテナサービスで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、プライベートにしておくこともできます。

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のHTTPSエンドポイントを 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。Lightsail のコンテナサービスには、カスタムドメイン名を持つ SSL/TLS 証明書を作成してアタッチできます。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DNSの が Lightsail DNS ゾーンを使用している場合は、ドメインの頂点 (`example.com`) またはサブドメイン (`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。または、ALIASレコードの追加をサポートするDNSホスティングプロバイダーを使用して、ドメインの頂点 (`example.com`) 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料金はいくらですか？

Lightsail コンテナサービスはオンデマンドの時間料金で請求されるため、使用した分のみお支払いいただきます。使用する Lightsail コンテナサービスごとに、月額サービス料金の上限まで固定時間料金が課金されます。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USDあたり 0.0094 USD (1 か月USDあたり 7 USD) から開始されます。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、実行中または無効の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すると、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コストが請求されます。例えば、1 か月に 100 時間、Micro のパワーと 1 のスケールで Lightsail コンテナサービスを使用すると、1.34 USD (0.0134 USD*100) 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、サービスの IN と の両方OUTのデータ転送にカウントされます。クォータを超えると、パブリック IP アドレスを使用する場合、Lightsail コンテナサービスOUTからインターネット、別の AWS リージョン 、または同じリージョンの AWS リソースへのデータ転送に対して課金されます。無料利用枠を超えるこれらのタイプのデータ転送の料金は、次のとおりです。

毎月のデータ転送クォータを超えた場合の料金

- 米国東部 (オハイオ) (us-east-2): 0.09 USDUSD/GB
- 米国東部 (バージニア北部) (us-east-1): 0.09 USDUSD/GB
- 米国西部 (オレゴン) (us-west-2): 0.09 USDUSD/GB
- アジアパシフィック (ムンバイ) (ap-south-1): 0.13 USDUSD/GB
- アジアパシフィック (ソウル) (ap-northeast-2): 0.13 USDUSD/GB
- アジアパシフィック (シンガポール) (ap-southeast-1): 0.12 USDUSD/GB
- アジアパシフィック (シドニー) (ap-southeast-2): 0.17 USDUSD/GB
- アジアパシフィック (東京) (ap-northeast-1): 0.14 USDUSD/GB
- カナダ (中部) (ca-central-1): 0.09 USDUSD/GB
- 欧州 (フランクフルト) (eu-central-1): 0.09 USDUSD/GB
- 欧州 (アイルランド) (eu-west-1): 0.09 USDUSD/GB
- 欧州 (ロンドン) (eu-west-2): 0.09 USDUSD/GB
- 欧州 (パリ) (eu-west-3): 0.09 USDUSD/GB
- 欧州 (ストックホルム) (eu-north-1): 0.09 USDUSD/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になり、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」を返します。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。サービスに関連付けられたHTTPSパブリックエンドポイントアドレス、コンテナイメージ、デプロイ履歴、ログも完全に削除されます。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 ?

現在、コンテナサービスは Lightsail CDNディストリビューションのオリジンとして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 にリダイレクトするように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HTTPS ?

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HTTPS、すべてのHTTPリクエストを 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のメトリクスを提供します。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 をサポートしていますIPv6か ?

Lightsail コンテナサービスHTTPSエンドポイントは、 IPv4と の両方をサポートしますIPv6。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。

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を使用すると、Amazon のグローバル配信ネットワークに保存して提供することで、Lightsail リソースでホストされるコンテンツの配信を簡単に高速化できます CloudFront。ディストリビューションは、シンプルなSSL証明書の作成とホスティングを提供することで、ウェブサイトがHTTPSトラフィックをサポートできるように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を処理するのに役立ちます。Lightsail のすべての機能と同様に、数回クリックするだけでセットアップを完了でき、月額料金がシンプルになります。

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を使用すると、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、Lightsail 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、静的IPv4アドレスを Lightsail インスタンスにアタッチする必要がありますか ?

はい。オリジンとして指定されたインスタンスには静的IPv4アドレスを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 をサポートしていませんIPv6。

WordPress ウェブサイトで Lightsail ディストリビューションを設定する方法を教えてください。

ディストリビューションを作成し、オリジンとして WordPress インスタンスを選択し、プランを選択すると、すべて設定済みです。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 WordPress 設定を自動的に設定します。

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て、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を使用すると、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、アタッチできます。

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が有効な一意の Amazon CloudFront ドメイン名で作成されます。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。詳細については、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 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HTTPS？

Lightsail ディストリビューションは、すべてのHTTPリクエストを に自動的にリダイレクトHTTPSして、コンテンツが安全に配信されるようにします。

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ポイントするには、apex ドメインをディストリビューションのデフォルトドメインにマッピングする ALIASレコードをドメインのドメインネームシステム (DNS) に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の違いは何ですか？

データ転送 IN と はインスタンスのデータ転送クォータにOUTカウントされますが、オリジンとビューワーOUTへのデータ転送のみがディストリビューションのクォータにカウントされます。さらに、ディストリビューションのクォータOUTを超えるすべてのデータ転送には超過料金が課金されますが、一部のタイプのデータ転送OUTはインスタンスで無料です。最後に、Lightsail ディストリビューションは異なるリージョンの超過モデルを使用しますが、レートの大部分はインスタンスの超過に対して請求されるものと同じです。

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は、ディストリビューションが受信したリクエストの合計数、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の量、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メトリクスを提供します。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で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 CloudFront トリビューションはいつ使用すべきですか？

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しているユーザー専用に設計されています。で別のサービスを使用してウェブサイトやアプリケーションを AWS ホストしている場合、複雑な設定ニーズがある場合、または 1 秒あたりのリクエスト数が多いか、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 を使用することをお勧めします 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 に移動できますか CloudFront？

はい。Amazon で同様の設定の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 CloudFront。Lightsail 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 CloudFront ビューションで設定することもできます。ディストリビューションを に移動するには、次の手順を実行します CloudFront。

Lightsail ディストリビューションを に移動する方法 CloudFront

- ディストリビューションのオリジンとして設定された 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 にエクスポートしEC2、Amazon 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EC2。詳細については、「Amazon へのスナップショットのエクスポートEC2」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構成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 デベロッパーガイド」の「ディストリビューションの作成」を参照してください。
- ディストリビューションにアタッチする証明書を AWS Certificate Manager c (ACM) で作成します CloudFront。詳細については、「 ユーザーガイド」の「パブリック証明書のリクエストACM」を参照してください。
- 作成したACM証明書を使用するように CloudFront 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の使用方法

Lightsail CDNディストリビューションは、固定価格のデータ転送バンドルを使用して作成され、サービスの使用コストをシンプルかつ予測可能にします。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 CDNディストリビューションは をサポートしていますIPv6か ？

すべての Lightsail CDNディストリビューションはデフォルトでIPv6有効になっています。ディストリビューションホスト名は、 IPv4と IPv6 アドレスの両方に解決されます。IPv6 は、 CDNの管理ページのネットワークタブのトグルを使用して無効にできます。

Lightsail CDNディストリビューションを操作するには、オリジンIPv6を有効にする必要がありますか？

いいえ。 CDN ディストリビューションは と の両方のIPv6IPv4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IPv4するときにシームレスに に変換します。したがって、ディストリビューションの背後にあるオリジンは、デュアルスタックでも、 IPv4のみでもかまいません。

データベース

Lightsail マネージドデータベースとは

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とオペレーティングシステムのセキュリティとヘルスを維持するので、インフラストラクチャ管理に関する深い専門知識なしで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には、計画に一定の量のメモリ、コンピューティング能力、およびSSDベース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、または既存のオンプレミスまたはホストされたデータベースから Lightsail に移行することで実行できます。

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ルするときにデータを同期させる必要があるCMSs、 WordPress や最も一般的なステートフルアプリケーションに特に役立ちます。マネージドデータベースは、Lightsail ロードバランサーおよび 2 つ以上の Lightsail インスタンスと組み合わせて、強力でスケーリングされた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できるため、アプリケーションの稼働時間を確保できます。

Lightsail が管理する機能

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 日間のポイントインタイム復元を可能にし、データ損失やコンポーネントの障害から保護します。Lightsail は、セキュリティを強化するために保管中および転送中のデータを自動的に暗号化し、データベースへの簡単で安全な接続のため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 は設定されたメンテナンスウィンドウ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 がサポートするデータベースの種類とバージョン

Lightsail マネージドデータベースは、My SQLおよび Postgre の最新バージョンをサポートしていますSQL。現在、これらのバージョンは MySQL 5.7、MySQL 8.0、PostgreSQL 9、PostgreSQL 10、PostgreSQL 11、PostgreSQL 12 です。Lightsail は、メジャーバージョンオプションごとに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、標準および高可用性プランで 4 つのサイズ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、標準および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、データをスタンバイデータベースに同期的にレプリケートし、インフラストラクチャに障害が発生した場合やメンテナンス中にスタンバイデータベースにフェイルオーバーすることで、データベースに冗長性と耐久性を追加します。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/メンテナンス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をスケールアップするには、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します。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、「Lightsail でのスナップショットからのデータベースの作成」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にはどうすればよいですか？

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、すべての自動バックアップレコードが削除され、 point-in-time 復元できなくなります。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 マネージドデータベースの手動スナップショットは、データベース管理ページから作成できます。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、USD月額 0.05 USD で請求されます。

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 マネージド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。データベースの削除中、Lightsail には、データの偶発的な損失を防ぐために、必要に応じて手動スナップショットを作成するワンクリックオプションが用意されています。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。

インスタンスを異なるアベイラビリティーゾーン AWS リージョン 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接続できますか？

異なる で実行されているインスタンスで Lightsail マネージドデータベースを使用することはできません AWS リージョン。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 マネージドデータベースにデータをロードするにはどうすればよいですか？

Lightsail マネージド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

データベースに接続し、標準のSQLクライアントアプリケーションを使用して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名前など、データベースのデータベース管理画面で接続データを確認できます。詳細については、「Amazon Lightsail で MySQL データベースに接続する」または「Postgre データベースに接続する」を参照してください。 SQL Amazon Lightsail

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そのデータベースの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口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 マネージドデータベースを自分の AWS 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にはどうすればよいですか？

Lightsail マネージドデータベースをEC2インスタンスに接続するには、パブリックインターネット経由で接続します。すべての AWS サービスへの接続では、データベースのデータ転送許容量が消費され、パブリックインターネット経由でのデータ転送許容量を超える AWS サービスへのデータ出力には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インスタンス間のVPCピア接続を使用することはできません。

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ます。これにより、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データベースを保護します。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 マネージドデータベースで使用され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、パブリックモードですべての Lightsail マネージドデータベースの MySQL のポート 3306 を開きます。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対してのみ開かれます。

Lightsail マネージドデータベースサービスは をサポートしていますIPv6か？

Lightsail マネージドデータベースは をサポートしていませんIPv6。

ドメイン

Lightsail ドメインで何ができますか？

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 (TLDs) を使用できますか？

Lightsail は Amazon Route 53 TLDsと同じ汎用 を使用します。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。地理的ドメインは、Route 53 を使用して登録された後、Lightsail コンソールで使用できます。Lightsail がサポートTLDsする の詳細については、Amazon Route 53」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DNSのサービスにできますか？

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DNSの管理を Lightsail に移管できます。詳細については、「ドメインのDNSレコードを管理するDNSゾーンを作成する」を参照してください。

Lightsail でのドメイン登録を開始するにはどうすればよいですか？

Lightsail にログインしたら、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 でドメインを登録するタイミング

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 に移管できますか？

ドメインは Route 53 に移管できます。ドメインの移管が完了すると、ドメインは Lightsail コンソールで使用可能になります。詳細については、Amazon Route 53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 にドメインを登録したら、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 () にポイントできますCDN。

Lightsail リソースを Amazon Elastic Compute Cloud (Amazon EC2) にエクスポートする

Amazon へのエクスポートとは EC2

Amazon へのエクスポートEC2は、Amazon で Lightsail インスタンスのコピーを作成できる機能ですEC2。Amazon にエクスポートするとEC2、Amazon EC2が提供する幅広いインスタンスタイプ、設定、料金モデルから選択でき、ネットワーク、ストレージ、コンピューティング環境をより細かく制御できます。

Amazon にエクスポートする理由 EC2

Lightsail では、バンドルされた予測可能な低価格で、さまざまなクラウドベースのアプリケーションを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 にエクスポートEC2すると、より多くのCPUパワー、メモリ、ネットワーク機能を備えた仮想マシンから、 と を備えた特殊なインスタンスや高速化されたインスタンスまで、幅広いインスタンスタイプでアプリケーションを実行できますFPGAsGPUs。さらに、Amazon EC2は自動管理とセットアップを減らし、 などのクラウド環境の設定方法をより細かく制御できますVPC。

Amazon へのエクスポートはどのようにEC2機能しますか？

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 に慣れているお客様はEC2、Amazon EC2作成ウィザードまたは API を使用して、既存の または EBSボリュームから新しい Amazon EC2インスタンスEC2AMIまたは Amazon EBSボリュームを作成できます。または、Lightsail はガイド付き Lightsail コンソールエクスペリエンスも提供して、新しいEC2インスタンスを簡単に作成できるようにします。

> **Note**
>
> cPanel & WHM (CentOS 7) インスタンスのスナップショットは Amazon にエクスポートできませんEC2。

どのように請求されますか?

Amazon へのエクスポートEC2機能は無料です。手動スナップショットを Amazon にエクスポートするとEC2、Lightsail 手動スナップショットに加えて、個別に Amazon EC2イメージの料金が請求されます。起動した新しい Amazon EC2インスタンスはEC2、Amazon EBSストレージボリュームやデータ転送 (Amazon ストレージボリュームを含む) からも課金されます。新しいインスタンスとリソースのEC2料金の詳細については、Amazon の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る Lightsail リソースは、削除されるまで通常の料金で請求されます。

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 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、マネージド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 から Amazon EBSボリュームとしてリハイドレートできますAPI。

Lightsail のどのリソースをエクスポートできますか ?

Lightsail の Amazon へのエクスポートEC2機能は、Linux および Windows インスタンススナップショットの Amazon 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EC2。また、Amazon へのブロックストレージディスク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EBS。現在、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 IPsおよび DNSレコードのエクスポートはサポートされていません。さらに、EC2現時点では、Django、Ghost、および cPanel & WHMインスタンスのスナップショットを Amazon にエクスポートすることはできません。

インスタンス

Lightsail インスタンスとは

Lightsail インスタンスは、 に存在する仮想プライベートサーバー (VPS) です AWS クラウド。Lightsail インスタンスを使用して、データの保存、コードの実行、ウェブベースのアプリケーションまたはウェブサイトの構築を行います。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、他の AWS リソースに接続したりできます。Lightsail コンソールから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

Lightsail プランには、バンドルとも呼ばれ、固定量のメモリ (RAM) とコンピューティング ()、 SSD ベースのストレージ (ディスクvCPUs) 、および無料のデータ転送許容量を持つ仮想サーバーが含まれています。Lightsail プランは、静的IPv4アドレスとDNS管理も提供します。Lightsail プランは時間単位のオンデマンドで請求されるため、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きに自動的にインストールされる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多数用意されています。アプリケーションテンプレートには、 WordPress、 WordPress Multisite、 cPanel &WHM PrestaShop、Django、Drupal、Ghost、Joomla!、Magento、Redmine、、Nginx (LEMP) LAMP、MEAN、Node.js などがあります。

ブラウザSSH内または独自の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できます。

Lightsail で使用できるオペレーティングシステム

Lightsail は現在、 AlmaLinux OS 9、Amazon Linux 2、Amazon Linux 2023、CentOS、Debian、Free、BSDOpen、Ubuntu の 7 つの Linux または Unix のようなディストリビューションと、2016SUSE、2019、2022 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、自分のライセンスが必要ですか？

Lightsail で使用できるすべてのインスタンスブループリントには、 cPanel & 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、Amazon Lightsail WHMの「クイックスタートガイド： cPanel &Amazon Lightsail」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、独自のライセンス () を持ち込む必要はありませんBYOL。

Lightsail インスタンスを作成する方法

Lightsail にログインしたら、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 を使用してインスタンスAPI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 インスタンスの動作

Lightsail インスタンスは、ウェブサーバー、デベロッパー環境、小規模なデータベースのユースケース AWS 向けに によって特別に設計されています。このようなワークロードは、 をCPU頻繁にまたは一貫して使用しませんが、パフォーマンスバーストが必要になる場合があります。Lightsail は、ベースラインレベルのパフォーマンスを提供するバーストパフォーマンスインスタンスCPUを使用し、ベースラインを超えてバーストする追加機能を提供します。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。

ビデオエンコーディングやアプリケーションなどのアプリケーションに、一貫して高いCPUパフォーマンスで高度に設定可能な環境やインスタンスが必要な場合はHPC、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は、サステナブルゾーンとバーストゾーンが表示されます。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。バーストゾーンで動作している間、インスタンスはより多くの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」を参照してください。

インスタンスのCPU使用率が持続可能なゾーンからバーストゾーンにまたがったときに通知されるメトリクスアラームを追加します。詳細については、Amazon Lightsail でのインスタンスメトリ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にはどうすればよいですか ?

Lightsail は、ブラウザからインスタンスのターミナルへのワンクリックの安全な接続を提供し、Linux/Unix ベースのインスタンスSSHへのアクセスと Windows ベースのインスタンスRDPへのアクセスをサポートします。ワンクリック接続を使用するには、インスタンス管理画面を起動し、 を使用してSSH接続 または を使用して接続 RDPを選択すると、新しいブラウザウィンドウが開き、インスタンスに自動的に接続します。

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する場合、Lightsail はSSH キーの保存および管理作業を行い、SSHクライアントで使用す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 を使用してインスタンスの手動スナップショットAPI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 AWS アカウントの他のリソースに接続するにはどうすればよいですか ?

Lightsail インスタンスは、VPCピアリングを使用して AWS 、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「ピアリングを有効にする」を選択すると、Lightsail がユーザーに代わって作業を行います。VPC ピアリングを有効にすると、プライベート VPCを使用してデフォルトの Amazon の他の AWS リソースに対処できますIPs。こちらから手順を確認いただけます。

> **Note**
>
> Lightsail とのVPCピアリングが機能するには、 AWS アカウントにデフォルトの Amazon VPCを設定する必要があります。2013 年 12 月より前に作成された AWS アカウントにはデフォルトの がなくVPC、設定する必要があります。デフォルト の設定の詳細については、VPChttps://docs.aws.amazon.com/AmazonVPC/latest/UserGuide/default-vpc.html「」を参照してください。

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パブリックIPv4アドレスが解放されるため、停止および起動後に同じ IP を

保持する必要があるインスタンスには静的IPv4アドレス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たパブリックIPv6アドレスは、インスタンスが停止および起動されても変更されないことに注意してください。

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。そのインスタンスで静的IPv4アドレスを使用していた場合、静的IPv4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。

ロードバランサー

Lightsail ロードバランサーで何ができますか？

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間でトラフィックを分散し、正常なターゲットインスタンスのみにトラフィックを向けることで、インスタンスの問題やデータセンターの停止が原因でアプリケーションがダウン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、ピーク負荷時に訪問者に良好なパフォーマンスを維持することもできます。

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、保守の複雑さを排除します。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。

異なる AWS リージョン 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ロードバランサーを使用できますか？

ロードバランサーは、異なる 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、手動で調整することなく、アプリケーションへのトラフィックの急増を処理するように自動的にスケーリングされます。アプリケーションで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され、引き続き効率的にトラフィックを Lightsail インスタンスに転送します。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、一貫して非常に大量のトラフィックが発生するアプリケーションでは、パフォーマンスが低下したり、スロットリングが発生したりする可能性があります。アプリケーションが一貫して 5 GB/時間を超えるデータを管理する場合、または一貫して多数の接続 (>400k の新しい接続/時間、>15k のアクティブな同時接続) を持つことが予想される場合は、代わりに Application Load Balancing EC2で Amazon を使用することをお勧めします。

Lightsail ロードバランサーはターゲットインスタンスにトラフィックをどのよう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 はウェブアプリケーションのルートにヘルスチェックリクエストを送信します。Lightsail から ping へのパス (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到達できる場合、Lightsail はそこにトラフィックを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 アカウントインスタンスのクォータまで、ロードバランサーには必要な数のターゲットインスタンスを追加できます。

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することができます。

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すると、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実行され、通常の Lightsail インスタンスとして Lightsail コンソールに表示されます。ロードバランサーを削除した後、レコードを更新して、以前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DNS誘導する必要がある可能性があることに注意してください。

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、データ整合性のために訪問者が同じターゲットインスタンスに到達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、 および HTTPS接続をサポートします。

Lightsail ロードバランサーは をサポートしていますIPv6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、デフォルトでデュアルスタックモードで動作します (つまり、 IPv4と IPv6 プロトコルの両方でクライアントトラフィックを受け入れます) 。IPv6 この日付より前に作成されたロードバランサーでは、ロードバランサーの管理ページの Networking タブのトグルを使用して を有効にできます。IPv6 このトグルを使用して、どのロードバランサーでも無効にできます。

IPv6 有効なロードバランサーを使用するには、ロードバランサーの背後にあるインスタンスIPv6を有効にする必要がありますか？

いいえ。ロードバランサーは IPv4 と の両方のIPv6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IPv4するときにシームレスに に変換します。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、デュアルスタックでも、 IPv4のみでもかまいません。

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、またはブロックストレージディスクの point-in-time 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。または、インスタンスとディスクで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、および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 で Linux/Unix インスタンスの日次スナップショットをスケジュールする方法です。1 日のうちの時間を選択でき、Lightsail は選択した時刻に毎日自動的にスナップショットを作成し、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にタグを付けたり、Amazon に直接エクスポートしたりすることはできませんEC2。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、Lightsail コンソール、Lightsail または を使用して Linux APIまたは Unix インスタンスで有効にできます。また AWS CLI、Lightsail APIまたは 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」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、最も古いスナップショットを削除して最新のスナップショットに置き換えます。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」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、Linux または Unix インスタンスの作成 AWS CLI 時、またはインスタンスの実行後に有効にすることができます。

自動スナップショットは、ディスクの作成時または作成後に有効にすることもできます。ただし、Lightsail API、、または を使用してのみ実行できます AWS CLI。

詳細については、Amazon Lightsail」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」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?

Lightsail アカウントに保存されているスナップショットに対してのみ料金が発生します。Lightsail スナップショット (手動および自動) の保存には USD1 か月あたり 0.05 USD 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 は毎日の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取得を再開し、最も古いスナップショットを削除して最新のスナップショットに置き換えます。

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」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。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 にエクスポートEC2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、「Amazon へのスナップショットのエクスポートEC2」を参照してください。

リソースヘルスメトリクスとアラーム

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の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含まれます。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、E メールアドレスに送信される E 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にすることができます。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。

ネットワーク

Lightsail で IP アドレスを使用する方法

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 (IPv62021 年 1 月 12 日より前に作成されたインスタンスでは、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 インスタンスと AWS リソース間で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 を使用して、登録済みドメイン名やローカルコンピュータからの または 接続など、インターネットSSHRDPからインスタンスに接続できます。また、静的IPv4アドレスをインスタンスにアタッチすることもできます。これにより、パブリックIPv4アドレスは、インスタンスが停止および開始されても変更されないIPv4

アドレスに置き換えられます。IPv6 インスタンスに割り当てられた アドレスは、インスタンスが削除されるか、インスタンスIPv6で を無効にすることでIPv6手動で解放されるまで変更されません。

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 および IPv6) と IPv6のみの設定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パブリック IP アドレスです。インスタンスに静的IPv4アドレスを割り当てて、そのパブリック を置き換えることができますIPv4。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。このようにして、インスタンスを置き換えるたびに新しい IP アドレスを指すように外部システム (DNSレコードなど) を再設定する必要はありません。Lightsail は現在、 IPsの静的 IPv4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IPv6で を無効にすることでIPv6手動で解放されるまで変更されません。

インスタンスにアタッチIPsできる静的な の数

インスタンスにアタッチできる静的 IP は一度に 1 つだけです。

DNS レコードとは

DNS は、 のような人間が読める名前`www.example.com`を、コンピュータ`192.0.2.1`が相互に接続するために使用するような英数字の IP アドレスに変換する、グローバルに分散されたサービスです。Lightsail を使用すると、 などの登録済みドメイン名`photos.example.com`を Lightsail インスタンスIPsのパブリックに簡単にマッピングできます。このようにして、ユーザーが のような人間が読める名前`example.com`をブラウザに入力すると、Lightsail はユーザーを誘導するインスタンスの IP にアドレスを自動的に変換します。これらの翻訳はそれぞれ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 または任意の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、インスタンスのどのポートにさまざまなタイプのトラフィックで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。

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?

イメージ、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、数回クリックするだけで Lightsail CDNディストリビューションに関連付けることができるため、世界中の視聴者へのコンテンツの配信を迅速かつ簡単に高速化できます。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、 AWS リージョン Lightsail が利用可能なすべての に 3 つの異なる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。

オブジェクトストレージでのデータ転送許容量の仕組みを教えてください。

Lightsail オブジェクトストレージとの間でデータを転送することで、データ転送許容量を消費できます。ただし、以下を除きます。

- 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のデータ転送
- Lightsail オブジェクトストレージから同じ 内の別の Lightsail リソースに転送されるデータ AWS リージョン （ 別の AWS アカウントのリソースを含むが、同じ 内 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るデータ

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、毎月の AWS 請求サイクル内で 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、 () を使用して AWS Command Line Interface Lightsail オブジェクトストレージの使用を開始することもできますAWS CLI。

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、アクセス許可タブで、アクセスキーを作成するか、Lightsail インスタンスにロールを割り当て、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。

他の AWS アカウントとバケットを共有するにはどうすればよいですか?

Lightsail では、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 AWS アカウント ID を使用して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簡単になります。 AWS アカウント ID を指定すると、そのアカウントはバケットへの読み取り専用アクセス権を持ちます。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、数回クリックするだけで Lightsail CDNディストリビューションに関連付けることができるため、世界中の視聴者へのコンテンツの配信を迅速かつ簡単に高速化できます。そのためには、Lightsail CDNディストリビューションを作成し、Lightsail バケットを Lightsail CDNディストリビューションのオリジンとして選択するだけです。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 でのバケットのメトリクスの表示」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。

Lightsail のタグ

タグとは

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であるため、Lightsail コンソールでリソースをフィルタリングするための「キーのみ」タグを作成できます。

Lightsail でタグを使用する方法

タグを使用すると、Lightsail コンソールと でリソースをグループ化およびフィルタリングしAPI、請求のコストを追跡および整理し、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規制できます。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、請求コンソールで「コスト配分タグ」を作成して、さまざまなプロジェクトまたはユーザー間でコストを追跡して割り当てます。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。
- アクセスの管理 – AWS アカウントへのアクセス権を持つユーザーが、 AWS Identity and Access Management ポリシーを使用して Lightsail リソースを編集、作成、削除する方法を制御できま

す。これにより、Lightsail リソースへのフルアクセスを他のユーザーに許可することなく、より簡単に他のユーザーとコラボレーションできます。

Lightsail でのタグの使用の詳細については、「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、次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ますが、Lightsail API、、または を使用してスナップショット AWS CLI 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ときに、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または を使用してスナップショット AWS CLI 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?

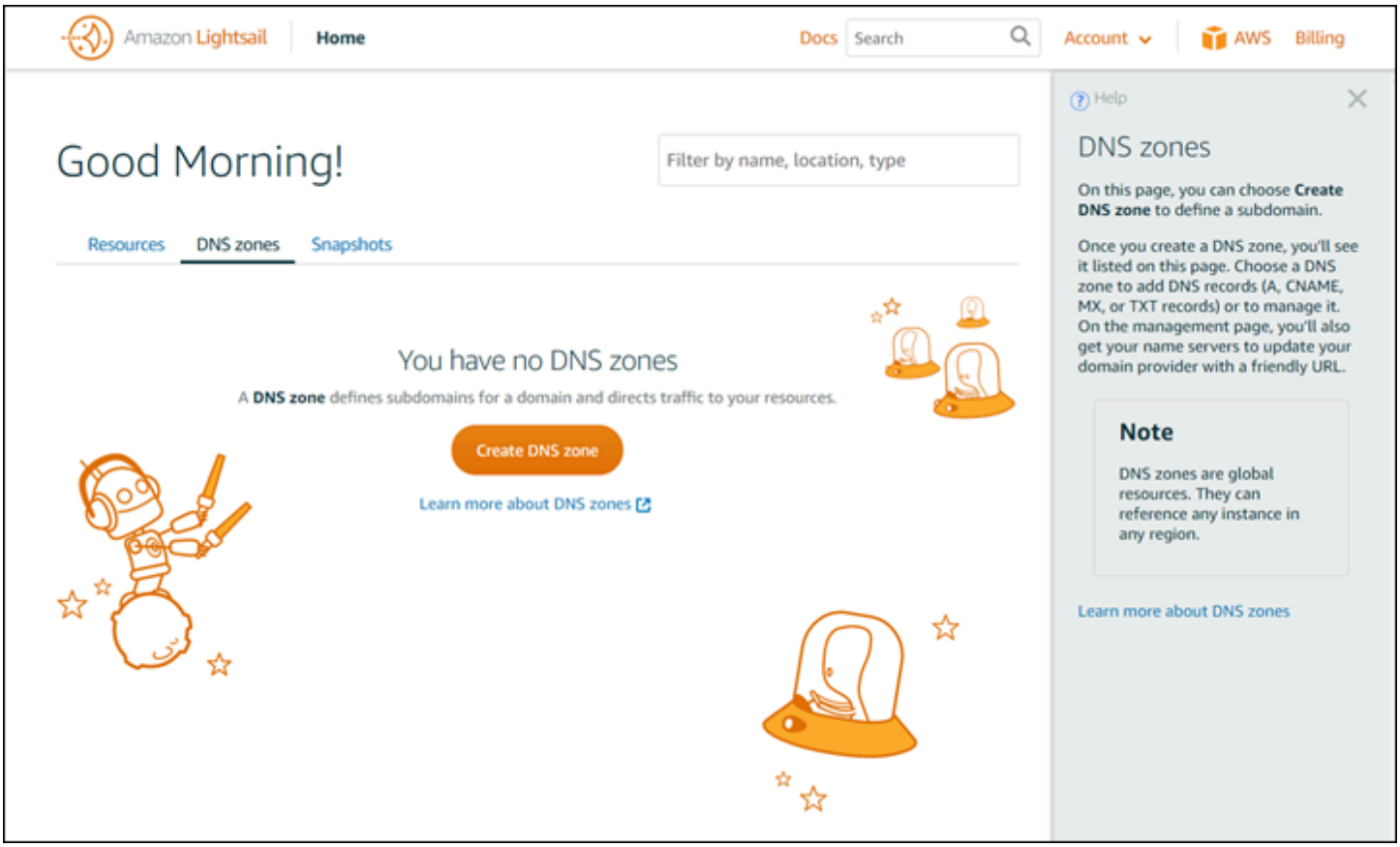
Lightsail タグはキーと値のペアであり、インスタンスなどのリソースをさまざまなカテゴリ (project:Blog、project:Game、project:Test など) に整理できます。これにより、リソースの整理、請求レポート、アクセス管理などのすべてのユースケースを全面的に管理できます。Lightsail コンソールでは、リソースにキーのみのタグを付けて、コンソールですばやくフィルタリングすることもできます。

Lightsail に役立つリソースを見つける

Amazon Lightsail では、いくつかの方法でヘルプを見つけることができます。

コンテキスト依存のヘルプパネル

Lightsail では、コンソールの各ページにコンテキストに応じたヘルプパネルがあり、そのページに固有のヒントや情報が追加されています。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。



ユーザーガイドについて

Amazon Lightsail ユーザーガイドには、Lightsail での作業に役立つハウツートピックと概念的な概要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

各ページの上部にある検索ボックスを使用して、Lightsail の任意のページからドキュメント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。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、フィードバックの提供を選択し、フィードバックを送信して提案を行うことができます。

Lightsail CLIと の使用 API

AWS Command Line Interface （ AWS CLI) または Lightsail を使用して、Lightsail リソース RESTAPIを作成、読み取り、更新、削除できます。に加えてRESTAPI、Java、Ruby、 JavaScript (Node.js)、Go、、Python、 などPHP、SDK複数の言語の もあります。NET (C#)、および C++。Lightsail の詳細についてはAPI、「Lightsail APIリファレンス」を参照してください。

> **Note**
>
> Lightsail 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API。Lightsail を使用するためのアクセスキーの設定について説明しますAPI。

AWS CLI は、Lightsail リソースを使用する際に役立ちます。でAWS AWS CLI、「」と入力`aws lightsail help`するだけで、使用可能なコマンドについて学習できます。特定のCLIコマンドに関するヘルプについては、コマンド名を入力し、その後`help`にコマンド名を入力して、そのパラメータと例外の詳細を確認します。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 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AWS。

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。