Amazon Elastic Compute Cloud

Microsoft Windows ガイド

API Version 2013-08-15

アマゾン ウェブ サービス

Amazon Elastic Compute Cloud: Microsoft Windows ガイド

アマゾン ウェブ サービス

Copyright © 2013 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

The following are trademarks of Amazon Web Services, Inc.: Amazon, Amazon Web Services Design, AWS, Amazon CloudFront, Cloudfront, Amazon DevPay, DynamoDB, ElastiCache, Amazon EC2, Amazon Elastic Compute Cloud, Amazon Glacier, Kindle, Kindle Fire, AWS Marketplace Design, Mechanical Turk, Amazon Redshift, Amazon Route 53, Amazon S3, Amazon VPC. In addition, Amazon.com graphics, logos, page headers, button icons, scripts, and service names are trademarks, or trade dress of Amazon in the U.S. and/or other countries. Amazon's trademarks and trade dress may not be used in connection with any product or service that is not Amazon's, in any manner that is likely to cause confusion among customers, or in any manner that disparages or discredits Amazon.

All other trademarks not owned by Amazon are the property of their respective owners, who may or may not be affiliated with, connected to, or sponsored by Amazon.

Amazon Elastic Compute Cloud Microsoft Windows Guide

Amazon Elastic Compute Cloud（Amazon EC2）は、ユーザーがソフトウェアシステムを構築しホストするための、コンピューティング能力を自在に拡張および縮小できるウェブサービスです（実際には Amazon のデータセンター内のサーバーインスタンス）。Amazon EC2 では、AWS マネジメントコンソール、API アクション、またはコマンドラインインターフェイスを使って、インフラストラクチャリソースにアクセスできます。このガイドを使って、Windows オペレーティングシステムで Amazon EC2 を使い始めましょう。

知りたい情報

知りたい情報	関連トピック
Amazon EC2 の概要を知る	Amazon EC2 とは (p. 3)
Amazon EC2 をすぐに使い始める	&EC2 Windows インスタンスの使用開始 (p. 8)
Amazon EC2 インスタンス上で WordPress をセットアップする	&EC2 インスタンスでの WordPress ブログのデプロイ (p. 19)
EC2 との相互動作についての基本概念を学ぶ	Amazon EC2 インフラストラクチャ (p. 22)
自分の Amazon EC2 インスタンスへのアクセスを制御する	Amazon EC2 のリソースに対するアクセスの制御 (p. 28)
Windows AMI の詳しい使用方法を見る	Windows Amazon マシンイメージ（AMI）(p. 34)
Amazon EC2 で Microsoft システムセンター 2012 向けの AWS マネジメントパックを使用する	Microsoft System Center Operations Manager 向け AWS マネジメントパック (p. 63)
セカンダリプライベート IP アドレスを認識するようにWindows インスタンスを設定する	VPC での Windows インスタンスのセカンダリプライベート IP アドレスの設定 (p. 92)
Amazon EC2 を使って HPC クラスタをセットアップする	Amazon EC2 での Windows HPC クラスターのセットアップ (p. 97)

知りたい情報	関連トピック
コマンドラインツールで作業を開始する	Windows への Amazon EC2 コマンドラインインターフェイスツールのインストール (p. 107)
Windows Server インスタンスで診断プログラムを実行する	AWS Diagnostics for Microsoft Windows Server (p. 114)

その他のリソース

Amazon EC2 に関するさらに詳しい情報は、次の表を参考にしてください。

実行する操作	関連するセクション
製品の全体的な概要と料金に関する情報	Amazon EC2 製品ページ
AWS ウェブアプリケーションホスティングのセットアップ方法	Getting Started Guide AWS Web Application Hosting for Microsoft Windows
Amazon EC2 の詳しい使用方法を見る	Amazon Elastic Compute Cloud User Guide
Amazon EC2 にプログラム的にアクセスするために利用できるライブラリ	Available Libraries
Amazon EC2 API を使い始める	Making API Requests

Amazon EC2 とは

Topics

概要

Amazon Elastic Compute Cloud (Amazon EC2) はアマゾン ウェブ サービス (AWS) の 1 つで、これを使用すると、インターネット全体のサーバー、ソフトウェア、およびストレージリソースにセルフサービスの形でアクセスできます。Amazon EC2 を使うことは基本的に、仮想サーバーやストレージデバイスで構成されるインフラストラクチャを時間単位でレンタルすることです。これらの仮想サーバーを使って、いつでも、必要な時間だけ、あらゆる合法的な目的のために、アプリケーションをインストール、実行、処理することができます。要件が満たされた後は、インフラストラクチャの利用を完全に終了することもできますし、または後で必要になる時まで能力を縮小してメンテナンスモードにしておくこともできます。料金は使用した分だけ支払えばよく、基本料金はありません。Amazon EC2 を使えば、高価なハードウェアを購入する必要がなく、トラフィックが少ないときや処理能力が低くてよいときにも無駄が生じません。

Amazon EC2 の詳細

Windows 環境での Amazon EC2 の動作Amazon EC2 は Amazon マシン画像 (AMI) と呼ばれるテンプレートを提供します。ここには、オペレーティングシステム、アプリケーションサーバー、アプリケーションなどのソフトウェアがすでに構成されています。ユーザーはこれらのテンプレートを使用して、AMI の実行コピーであるサーバーのインスタンスを起動します。インスタンスを起動した後は、物理的なサーバーと同じように使用できます。また、1 つの AMI から複数のインスタンスを起動して、各インスタンス上に同じ構成を複製することもできます。

Amazon は、Windows プラットフォームに固有のソフトウェア構成を含むさまざまな AMI を公開しています。さらに、AWS 開発者コミュニティのメンバーが独自のカスタム AMI を作成してきました。Amazon またはその他の信頼できるソース提供者が作成した Windows AMI を使えば十分な場合もあり

ます。そしてインスタンスを起動するたびにスクリプトを実行して必要なデータやソフトウェアを提供するよう Windows インスタンスをカスタマイズすることができます。また、プリインストールされたアプリケーションや構成済みのアプリケーションを使ってカスタム Windows AMI を作成することもできます。これらの AMI はすばやく効率的に起動できるため、ライブデプロイメントに使用できます。Amazon Windows AMI の詳細については、Windows Amazon マシンイメージ (AMI) (p. 34) を参照してください。AMI とインスタンスの使用方法については、&EC2 Windows インスタンスの使用開始 (p. 8) を参照してください。

Windows Server と Amazon EC2 Windows インスタンスの違い

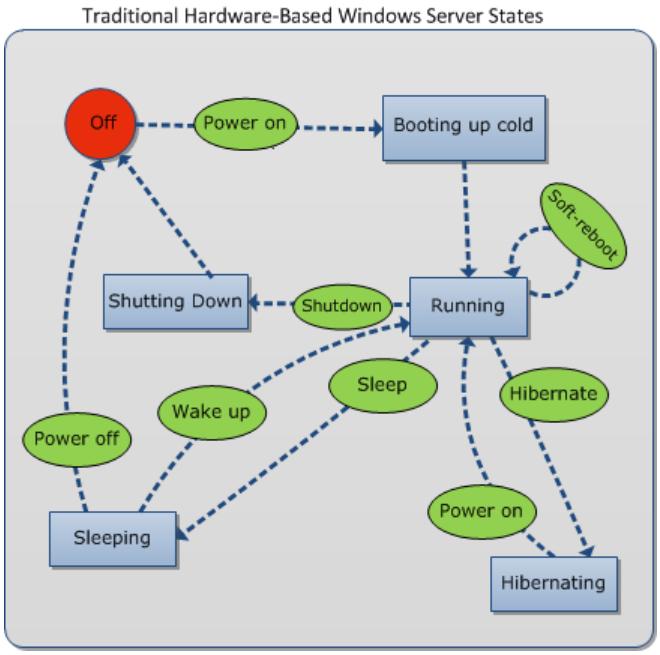
Amazon EC2 インフラストラクチャは、インターネット経由でアクセスできる仮想サーバーで構成されています。これらは通常、クラウドサーバーと呼ばれます。Amazon EC2 を使用することで、高価なハードウェアを購入して維持する必要がなくなります。しかし Amazon EC2 Windows インスタンスを起動する前に、クラウドサーバー上で実行されるアプリケーションのアーキテクチャが、ハードウェア上で実行される従来のアプリケーションモデルのアーキテクチャと大きく異なることを理解しておく必要があります。クラウドサーバー上でアプリケーションを実装するには、設計プロセスに大幅な変更が必要です。

以下の表に、Windows Server と Amazon EC2 Windows インスタンスの重要な相違点についてまとめます。

Amazon EC2 Windows インスタンス	Windows Server
需要に応じてデプロイおよび終了するよう設計されている。	いったんセットアップすると容易に廃棄できない。
リソースと性能の拡張や縮小が容易である。	リソースと性能には物理的な限界がある。
インフラストラクチャの利用に対して料金を支払う。インスタンスを終了すると課金も停止する。	インフラストラクチャに対して支払いを行うため、使っても使わなくても価格は同じ。
機器を設置するための空間を必要とせず、定期的なメンテナンスも不要。	機器を設置するための空間と、定期的なメンテナンスが必要。

Amazon EC2 Windows インスタンスは、起動後は、従来のハードウェアベースの Windows Server とほぼ同じように動作します。例えば、Windows Server と Amazon EC2 インスタンスのどちらも、ウェブアプリケーションの実行、バッチ処理、大規模な計算能力を必要とするアプリケーションの管理などを実行できます。しかし、サーバーハードウェアモデルとクラウドコンピュートモデルの間には大きな違いがあります。Amazon EC2 インスタンスが実行される方法は、従来の Windows Server の実行方法とは異なります。

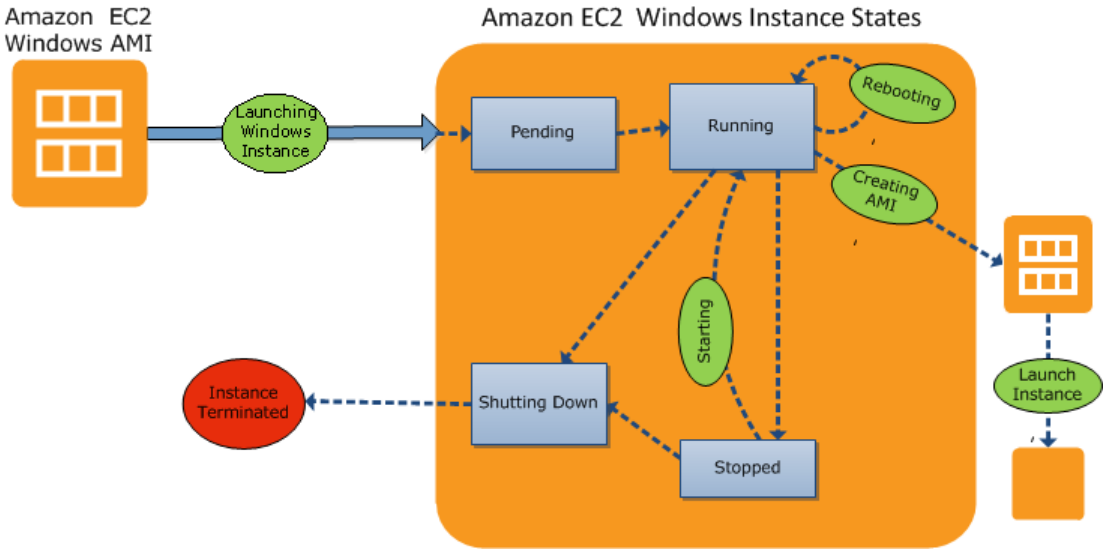
従来の Windows Server は、起動からシャットダウンまでにいくつかの段階を通過します (下の図を参照) 。



従来のハードウェアベースの Windows Server の起動は、電源ボタンを押すところから始まります。これは*コールドブート*と呼ばれます。サーバーが起動して動作し始めた後は、シャットダウンするまで実行し続けるか、一定期間スリープ状態にしておくか、またはハイバネーション状態にしておくか選択することができます。ハイバネーション中とスリープ中は、サーバーはオフになっています。Windows Server の電源をオンにすることで、このような状態から実行状態に戻すことができます。しかし、サーバーの電源をオフにした場合は、もう一度実行状態に戻すにはコールドブート以外に方法がありません。

従来の Windows Server は、電源オフになっている間、関連付けられたリソースは変更されることなく、電源を切った時点の状態を保持します。ハードドライブに保存した情報はその場所に留まり、必要なときにはいつでもアクセスできます。

Amazon EC2 Windows インスタンスは、多くの点で従来のハードウェアベースのサーバーと似ています。下の図と上の図を比較してみてください。



Amazon EC2 Windows インスタンスは、インスタンスの起動によって開始します。次に、登録を実行するため、一時的に保留状態になります。その後実行状態に移行し、そこではインスタンスを再起動、停止、または再開することができます。Windows インスタンスは、インスタンスを終了するためのシャットダウン手順を開始するまで、実行状態を維持します。Amazon EC2 Windows インスタンスが実行状態のときに、インスタンスのイメージを作成して追加のインスタンスを起動することもできます。この機能により、需要に応じてインフラストラクチャをスケーリングできます。

Note

Amazon EC2 Windows インスタンスを終了すると、そのインフラストラクチャは利用できなくなります。同じインフラストラクチャを使用するには、新たにインスタンスを起動する必要があります。

Amazon EC2 インスタンスの実行中または停止中は、Amazon EC2 インスタンスとそれに付随するリソースを制御できます。インスタンスを終了した後は、同じ構成で別のインスタンスを起動することもできますし、別の要件を満たすために別の構成でインスタンスを起動することもできます。

Amazon EC2 Windows インスタンスで実行するアプリケーションの設計

先のセクションで述べた相違点は、Amazon EC2 Windows インスタンスで実行させるアプリケーションを設計する際に、非常に重要になります。

Amazon EC2 用に作成されたアプリケーションは、基盤となるコンピューティングインフラストラクチャを、オンデマンドベースで使用します。ジョブを実行するために必要なリソース (ストレージや計算処理など) をオンデマンドで使用し、処理が終了したらリソースを解放します。また多くの場合、ジョブの終了後はアプリケーション自体も廃棄されます。アプリケーションは、稼働中にはリソース要件に応じて弾力的に拡張または縮小します。Amazon EC2 インスタンス上で実行されるアプリケーションは、インフラストラクチャにエラーが発生しても、自由に各種コンポーネントを終了して再作成することができます。

Windows アプリケーションを Amazon EC2 で実行する前提で設計すれば、需要の変化に応じてコンピューティングおよびストレージリソースを迅速にデプロイ/縮小させることも可能です。

Amazon EC2 Windows インスタンスの実行には、従来の Windows Server なら必要とするような、ハードウェア、ソフトウェア、ストレージといったシステムパッケージ一式を調達する必要がありません。代わりに、Windows アプリケーションの拡張性と全体的なパフォーマンスを向上させるために、各種のクラウドリソースをどのように使用するかに専念できます。

Amazon EC2 ではエラーや機能停止に備えた設計がアーキテクチャーに不可欠となります。拡張性と冗長性のある他のあらゆるシステムと同様、このシステムのアーキテクチャでも、計算処理、ネットワーク、およびストレージのエラーに備えなければなりません。各種のエラーを処理するためのメカニズムをアプリケーションに組み込んでおく必要があります。鍵となるのは、密に結合されておらず、非同期的に相互動作し、お互いを独立した拡張性を持つブラックボックスとして扱う、個々のコンポーネントで構成されたモジュラーシステムの構築です。これにより、いずれかのコンポーネントにエラーが発生したりビジー状態になったりしても、現在のシステムを維持したままそのコンポーネントの別のインスタンスを起動することができます。

もう 1 つ、エラーに強い設計に重要なのは、アプリケーションを地理的に分散することです。アプリケーションを地理的に分散したリージョンに複製しておくことで、システムの可用性を向上させることができます。詳しくはリージョンとアベイラビリティーゾーンの使用を参照してください。

Amazon EC2 インフラストラクチャはプログラム可能で、デプロイ作業の自動化、ソフトウェアとアプリケーションのインストールと設定、仮想サーバーのブートストラップなどをスクリプトを使って実行できます。

Amazon EC2 Windows インスタンス上で実行するアプリケーションアーキテクチャーのすべてのレイヤにおいてセキュリティを実装する必要があります。重要なデータや機密データを Amazon EC2 環境に保存することについて懸念がある場合は、アップロード前に暗号化してください。Amazon EC2 では、ファイルの暗号化はオペレーティングシステムにより異なります。

Amazon EC2 の課金方式

Amazon EC2 は利用した分だけ料金を支払うシステムで、基本料金はありません。料金は大まかに以下のように分類されます。

- インスタンス使用

  Important

  インスタンスの起動と同時に課金が始まります。使用料は、インスタンスの実行時間 (アイドル時間も含む) に対して請求されます。

- データ転送
- ストレージ

料金の内訳と価格については、Amazon EC2 料金表ページを参照してください。サンプル環境で調達コストを試算するには、AWS エコノミクスセンターの Amazon EC2 コスト比較計算ツールをご利用ください。

請求書を確認するには、AWS アカウントアクティビティページを参照してください。

Windows ユーザーのためのヒントとコツ

このセクションでは、Amazon EC2 を使用する上でのヒントとコツを紹介します。

- Internet Explorer で最良の結果を得るには、最新バージョンを実行してください。
- RDP セッションを開いたときにドメインのプロンプトが表示されたら (例: ユーザー名が [IP-1024BB\Administrator] と表示されます)、[Remote Desktop] ダイアログボックスで [Options] をクリックし、[Administrator] の前にあるテキストを削除してください。
- インスタンスに接続する最も簡単な方法は、EC2コンソールでインスタンスを右クリックし、[Connect] をクリックすることです。

  インスタンスのパブリック DNS 名は変わることがあります (インスタンスが再起動された場合など)。キャッシュされた RDP セッションを使っていてインスタンスに接続できない場合、これが原因の可能性があります。コンソールを使って接続した場合は、DNS パブリック名が自動的に取得されるため、最新の DNS パブリック名が使用されます。
- キーペアなしでインスタンスを起動しないでください。キーペアがないと、インスタンスに接続できません。
- インスタンスを起動して接続したら、次の 2 つのことを実行してください。
  1. インスタンスにログインして管理者パスワードを変更する。
  2. ログインしたままで、管理者権限を持つユーザーアカウントをもうひとつ作成する。
     このアカウントは、元の管理者パスワードを忘れた場合や、元の管理者アカウントで問題が発生した場合に使用することができます。

&EC2 Windows インスタンスの使用開始

このチュートリアルでは、Amazon Elastic Compute Cloud (Amazon EC2) Windows インスタンスの使用を開始するための手順を説明します。ほとんどの操作は、ポイントアンドクリック型のウェブベースのインターフェイス、AWS マネジメントコンソールで行います。また、開始手順を紹介した短い動画「Getting Started with Amazon EC2: Launching a Windows Instance」も用意されています。

この Amazon EC2 チュートリアルを完了するには

1. EC2 のサインアップ (p. 8)
2. Windows インスタンスを起動する (p. 9)
3. Windows インスタンスへの接続 (p. 12)
4. (オプション) Elastic IP アドレスの作成 (p. 13)
5. (オプション) インスタンスを監視する CloudWatch アラームの作成 (p. 14)
6. クリーンアップ (p. 17)

チュートリアルを完了したら、クリーンアップを行う必要があります。

EC2 のサインアップ

アマゾン ウェブ サービス (AWS) にサインアップすると、AWS アカウントは、Amazon EC2 を含む &AWS のすべてのサービスに自動的に登録されます。料金が発生するのは、実際に使用したサービスの分のみです。

Amazon EC2 については、お客様が利用された分のみのお支払いとなります。AWS の新規のお客様の場合、Amazon EC2 を無料で使い始めることができます。詳細については、「AWS 無料使用範囲」を参照してください。

AWS アカウントを作成するには

1. Go to http://aws.amazon.com, and then click Sign Up.
2. Follow the on-screen instructions.

Part of the sign-up procedure involves receiving a phone call and entering a PIN using the phone keypad.

Windows インスタンスを起動する

AWS マネジメントコンソールを使用して Windows インスタンスを起動する手順を以下に示します。インスタンスとは、AWS クラウドにある仮想サーバーです。Amazon EC2 を使用して、インスタンスで実行されるオペレーティングシステムとアプリケーションをセットアップし、設定することができます。

Important

AWS にサインアップすると、AWS 無料使用範囲 を利用して、Amazon EC2 を無料で使い始めることができます。AWS アカウントを作成したのが過去 12 か月以内で、Amazon EC2 と Amazon EBS の無料使用範囲を使い切っていない場合、無料使用範囲内で利用できるオプションを選択することで、このチュートリアルでは一切費用がかかりません。それ以外の場合、インスタンスを起動してから終了する (このチュートリアルの最後のタスク) までの間、アイドル時間を含めて、標準の Amazon EC2 利用料が発生します。無料使用範囲外でこのチュートリアルを完了するためにかかる費用は小額です。

インスタンスを起動するには

1. Sign in to the AWS Management Console and open the Amazon EC2 console at https://console.aws.amazon.com/ec2/.
2. ナビゲーションバーで、インスタンスを起動するリージョンを選択します。このチュートリアルでは、デフォルトのリージョンを使用します。通常はこの選択が重要になります。Amazon EC2 リソースにはリージョンを超えて共有できるものと、できないものがあるからです。例えば既存の Amazon EBS ボリュームにインスタンスを接続するには、そのインスタンスのリージョンにボリュームと同じものを選択する必要があります。

3. コンソールダッシュボードで、[Launch Instance] をクリックします。
4. [Create a New Instance] ページで、[Quick Launch Wizard] をクリックします (このウィザードでは、自動でさまざまな構成設定が選択されるので、すぐに始められます)。

5. （任意）[Name Your Instance] に、わかりやすいインスタンス名を入力します（複数のインスタンスを実行する場合は、コンソール上で区別しやすい名前にすると便利です）。
6. [Choose a Key Pair] では、作成済みの既存のキーペアを選択するか、新しいキーペアを作成します。ここではキーペアを新しく作成することにします。

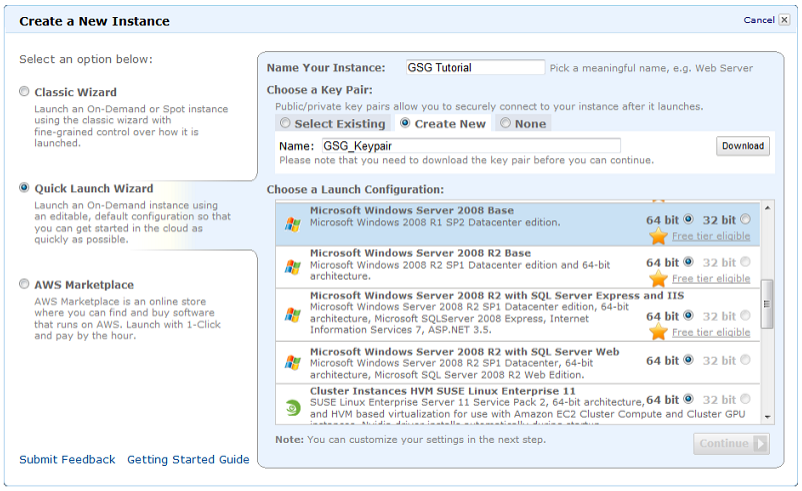
   

   Important

   決して [None] オプションを選択しないでください。キーペアなしでインスタンスを起動すると、そのインスタンスに接続できなくなります。

   a. [Create New] をクリックします。
   b. キーペアの名前を入力してから [Download] をクリックします。プライベートキーの内容は、インスタンスが起動した後、そのインスタンスに接続するときに必要です。キーペアのプライベート部分は、アマゾン ウェブ サービスでは保持されません。
   c. プライベートキーをお使いのコンピュータの安全な場所に保存します。このキーはインスタンスへの接続に必要になるので、場所をメモしておいてください。

7. Quick Launch Wizard により、[Choose a Launch Configuration] に、インスタンスのテンプレートとして機能する基本的な Amazon マシンイメージ（AMI）の一覧が表示されます。AMI には、ウェブサーバー、データベースサーバーなど、サーバーの新しいインスタンスを作成するために必要なものがすべて含まれます。このチュートリアルでは、Microsoft Windows Server 2008 R2 の 64 ビットバージョンを選択します。設定に星印が付いているので、これは無料使用範囲で使えます。



8. [Continue] をクリックして、インスタンスの設定を表示し、カスタマイズします。
9. [Security Details] の [Security Group] に、ウィザードが自動選択したセキュリティグループが表示されます。

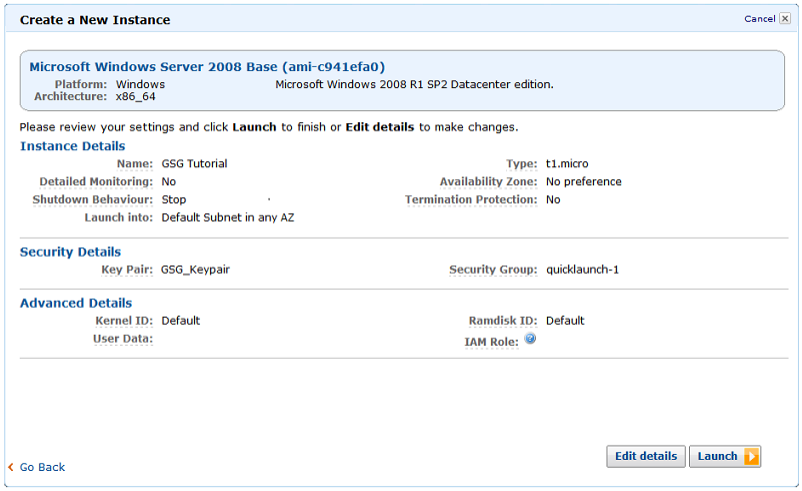
   セキュリティグループとは、インスタンスのファイアウォールルールを定義するものです。このルールでは、どの着信ネットワークトラフィックをインスタンスに配信するかを指定します。他のトラフィックはすべて無視されます。

   Amazon EC2 を使うのが初めてで、セキュリティグループをまだセットアップしていない場合は、セキュリティグループが自動的に定義されます。名前と説明は、[Edit details] ボタンをクリックすると変更できます。セキュリティグループには、インスタンスへの接続を可能にする基本のファイアウォールルールが含まれます。Windows インスタンスの場合、ポート 3389 で リモートデスクトッププロトコル（RDP）を介して接続します。

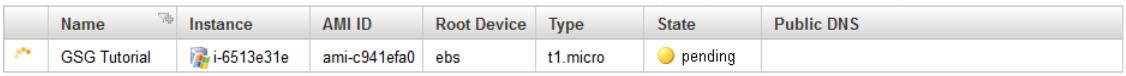
以前に Amazon EC2 を使用したことがある場合は、Windows インスタンス用の既存のセキュリティグループが自動的に検出されます。

Caution

作成するセキュリティグループでは、すべての IP アドレスが指定ポート（例: RDP）経由でインスタンスへアクセスすることが許可されます。今回はチュートリアルで短時間使うだけなので、このままでもかまいませんが、本番環境では安全ではありません。本稼働の場合は、特定の IP アドレスまたは IP アドレス範囲のみにインスタンスへのアクセスを許可してください。

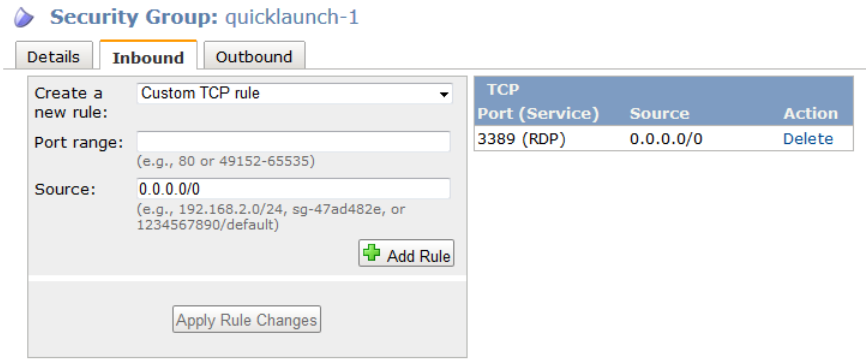


10. `t1.micro` インスタンスが Windows Server 2008 AMI を使用して起動するように設定されていることを確認してから、[Launch] をクリックしてインスタンスを起動します。
11. インスタンスを起動することを知らせる確認ページが表示されます。[Close] をクリックして確認ページを閉じ、コンソールに戻ります。
12. ナビゲーションペインの [Instances] をクリックして起動のステータスを表示します。インスタンスはすぐに起動します。インスタンスを起動した直後のステータスは `pending` です。インスタンスが開始されると、ステータスは `running` に変わり、インスタンスはパブリック DNS 名を取得します（[Public DNS] 列が非表示の場合、[Instances] ページの右上隅にある [Show/Hide] アイコンをクリックして [Public DNS] を選択します）。



	Name	Instance	AMI ID	Root Device	Type	State	Public DNS
	GSG Tutorial	i-6513e31e	ami-c941efa0	ebs	t1.micro	pending	

13. インスタンスのパブリック DNS 名を記録します。この名前は、次の手順で必要になります。
14. （オプション）インスタンスを起動すると、セキュリティグループのルールを表示できます。[Instances] ページからインスタンスを選択します。[Description] タブで [Security Groups] を見つけて [view rules] をクリックします。



このセキュリティグループに設定されたルールは 1 つだけで、任意の IP ソースからポート 3389 への RDP トラフィックを許可しています。IIS と SQL を実行する Windows インスタンスを起動すると、Quick Launch Wizard は、HTTP 用のポート 80 (IIS 用) と MS SQL 用のポート 1433 へのトラフィックを許可するルールを追加した、新しいセキュリティグループを作成します。

Windows インスタンスへの接続

Windows インスタンスに接続するには、まず初期の管理者パスワードを取得します。次に、リモートデスクトップを使用してインスタンスに接続するときに、このパスワードを指定します。

Note

Windows インスタンスでは、一度に 2 つの同時リモート接続しか許可されていません。3 つめの接続を行おうとすると、エラーが発生します。詳細については、「1 つの接続で実行可能な同時リモート接続数を構成する」を参照してください。

Windows インスタンスに接続するには

1. EC2 コンソールでは、インスタンスを選択し、[Actions] をクリックしてから、[Connect] をクリックします。
2. [Console Connect] ダイアログボックスで、[Retrieve Password] をクリックします (インスタンスが起動してからパスワードが使用できるようになるまで数分かかります) 。
3. [Browse] をクリックし、インスタンスの起動時に作成したプライベートキーファイルを探します。ファイルを選択して [OK] をクリックすると、ファイルの内容がすべて [Private Key contents] ボックスにコピーされます。
4. [Decrypt Password] をクリックします。インスタンスのデフォルトの管理者パスワードが [Console Connect] ダイアログボックスに、前に表示されていた [Retrieve Password] リンクと置き換わる形で表示されます。
5. デフォルトの管理者パスワードを記録するか、クリップボードにコピーします。このパスワードはインスタンスに接続するのに必要です。
6. [Download shortcut file] をクリックします。ブラウザによって .rdp ファイルを開くか、保存するよう求められます。どちらでもかまいません。終了したら、[Close] をクリックして [Console Connect] ダイアログボックスを閉じます。
7. .rdp ファイルを開いた場合は、[リモートデスクトップ接続] ダイアログボックスが表示されます。.rdp ファイルを保存した場合は、ダウンロードしたディレクトリまで移動して .rdp ファイルをダブルクリックすると、このダイアログボックスが表示されます。リモート接続の発行元が不明であるという警告が表示されることがあります。[接続] をクリックしてインスタンスに接続します。セキュリティ証明書を認証できなかったという警告が表示されることがあります。[Yes] をクリックして続行します。

8. ログイン画面が表示されたら、ユーザー名 `Administrator` を使用してインスタンスにログインします。パスワードは、ステップ 7 で記録またはコピーしたデフォルトの管理者パスワードを使用します。

次のことをお勧めします。

- 管理者パスワードをデフォルト以外の値に変更します。パスワードの変更は、そのインスタンスにログインした状態で行います。これは通常の Windows Server と同じです。
- 管理者権限を持つユーザーアカウントをもう 1 つインスタンスに作成します。これは管理者パスワードを忘れた場合や、管理者アカウントで問題が発生した場合の安全策です。

Elastic IP アドレスの作成

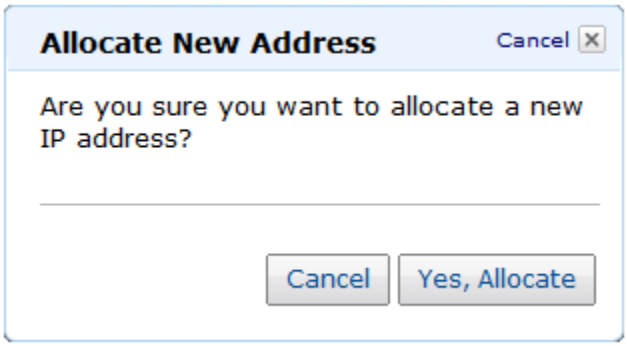
デフォルトでは、EC2-Classic またはデフォルト VPC で起動されたすべての Amazon EC2 インスタンスに対して、起動時に 2 つの IP アドレスが割り当てられます。1 つはプライベートアドレス (RFC 1918) で、もう 1 つはネットワークアドレス変換 (NAT) によってプライベート IP アドレスにマップされるパブリックアドレスです。デフォルト以外の VPC で起動されたインスタンスは、デフォルトではパブリック IP アドレスが割り当てられませんが、起動時に割り当てることができます。

インスタンスに接続するには、パブリック IP アドレスに関連付けられたパブリック DNS 名を使用します。ただし、この名前は固定ではなく、インスタンスが停止して再起動されたときなどに変更されることがあります。接続先として固定アドレスが必要な場合は、Elastic IP アドレスを使用できます。

Elastic IP アドレスは、動的クラウドコンピューティングのために考案された、静的 IP アドレスです。また Elastic IP アドレスは、特定のインスタンスではなく、お使いのアカウントに関連付けられます。アカウントに関連付けた Elastic IP アドレスは、明示的に解放するまでアカウントに関連付けられたままになります。従来の静的 IP アドレスと異なるのは、Elastic IP アドレスでは、インスタンスやアベイラビリティーゾーンに障害が発生しても、パブリック IP アドレスをアカウント内の任意のインスタンスに高速リマッピングすることで、障害をカバーできる点です。

Elastic IP アドレスを Windows インスタンスに割り当てるには

1. ナビゲーションペインで [Elastic IPs] をクリックします。
2. [Allocate New Address] をクリックします。
3. [Allocate New Address] ダイアログボックスで、[Yes, Allocate] をクリックします。



4. 作成した Elastic IP アドレスを選択し、[Associate Address] をクリックします。
5. [Associate Address] ダイアログボックスの [Instance] リストからインスタンスを選択し、[Yes, Associate] をクリックします。

インスタンスを監視する CloudWatch アラームの作成

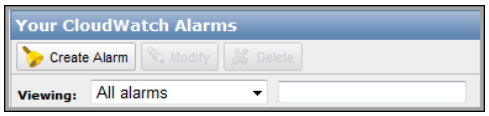
Amazon CloudWatch を使うと、インスタンスを様々な面から監視し、指定した基準でアラームを送信するよう設定できます。例えば、インスタンスの CPU 使用率が 70% を超えたときにメールで知らせるアラームを設定できます。

ここではインスタンスを起動したばかりなので、CPU 使用率がこのしきい値を超えることはまず考えられません。代わりに、CPU の使用率が 70% を *下回った* 状態が 5 分間継続したら E メールで知らせる CloudWatch アラームを設定してみましょう。

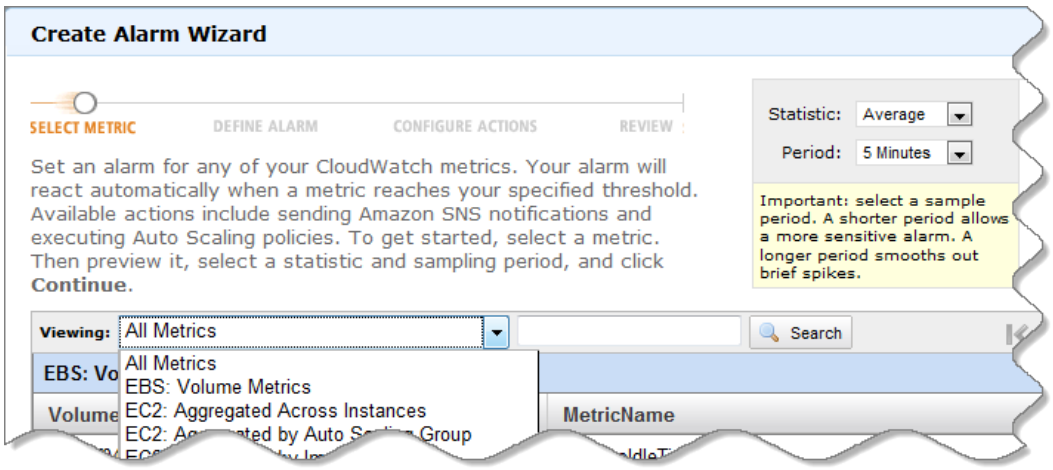
[Create Alarm Wizard] の手順に従ってアラームを作成します。

Create Alarm Wizard を開くには

1. Open the Amazon CloudWatch console at https://console.aws.amazon.com/cloudwatch/.
2. ナビゲーションペインの [Alarms] をクリックします。
3. [CloudWatch Alarms] ページで [Create Alarm] をクリックします。



4. [Create Alarm Wizard] の [SELECT METRIC] ページが開きます。



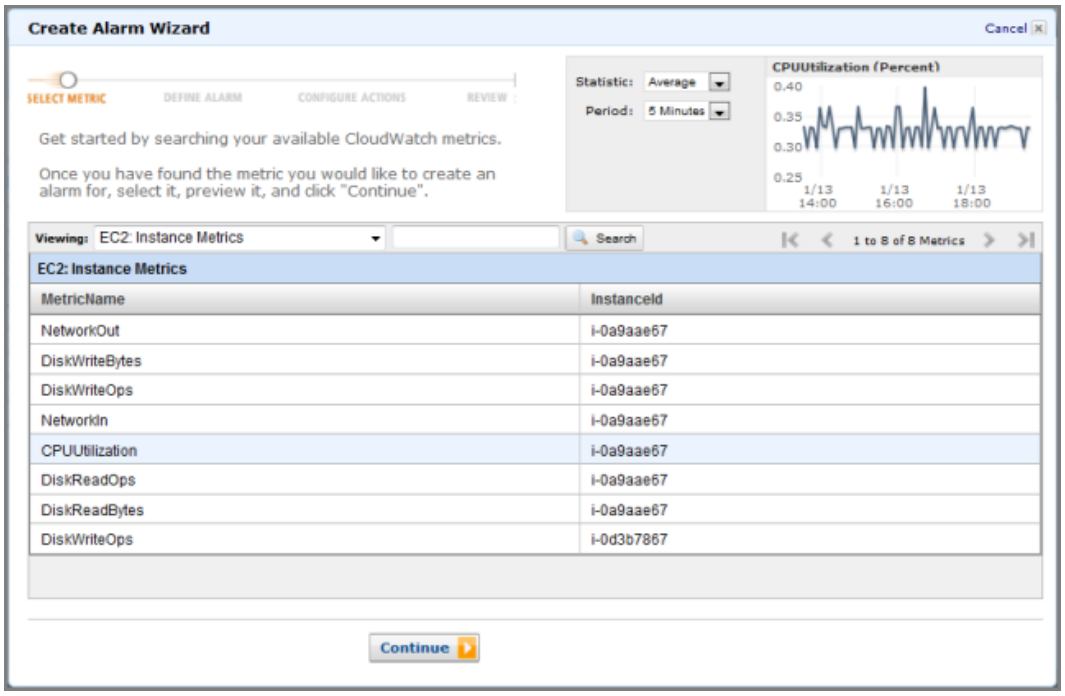
アラームの基準を選択するには

1. [Create Alarm Wizard] の [SELECT METRIC] ページで、[Viewing] リストから [EC2: Instance Metrics] を選択します。

   各インスタンスで利用可能な基準が [EC2 Instance Metrics] ペインに表示されます。
2. インスタンスを ID で探して [CPUUtilization] がある行を選択します。

   1 インスタンスの平均 `CPUUtilization` を示すグラフが [SELECT METRICS] ページの右上に表示されます。

3. [Statistic] リストから [Average] を選択します。
4. [Period] リストから、期間 (**5　** など) を選択します。
5. [Continue] をクリックします。

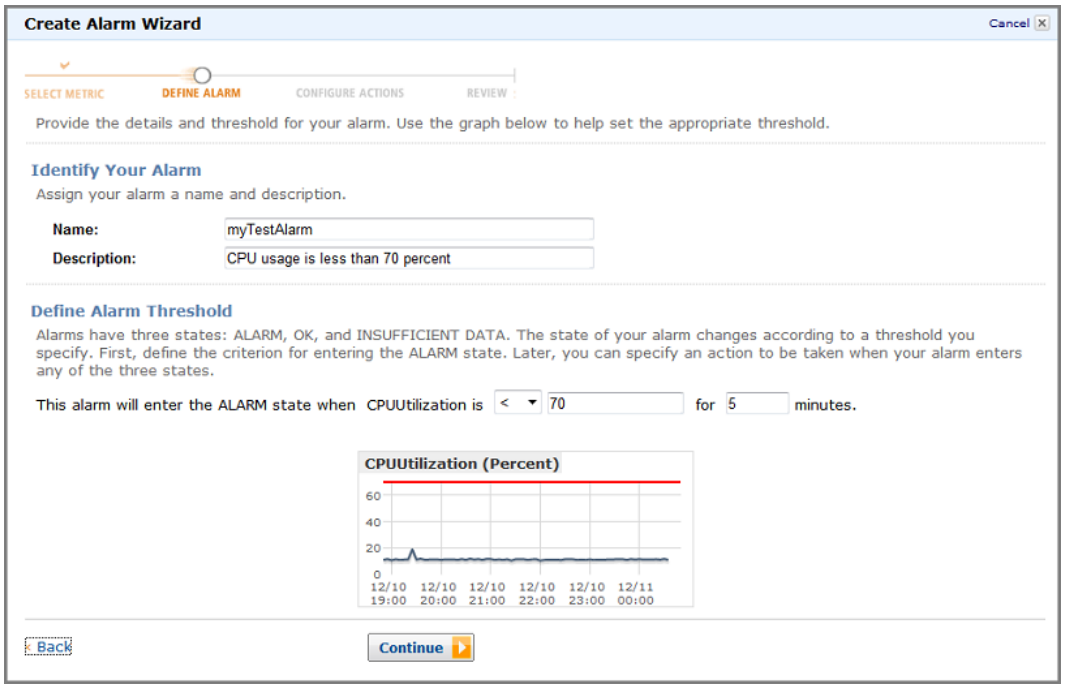


6. [Create Alarm Wizard] の [DEFINE ALARM] ページが開きます。

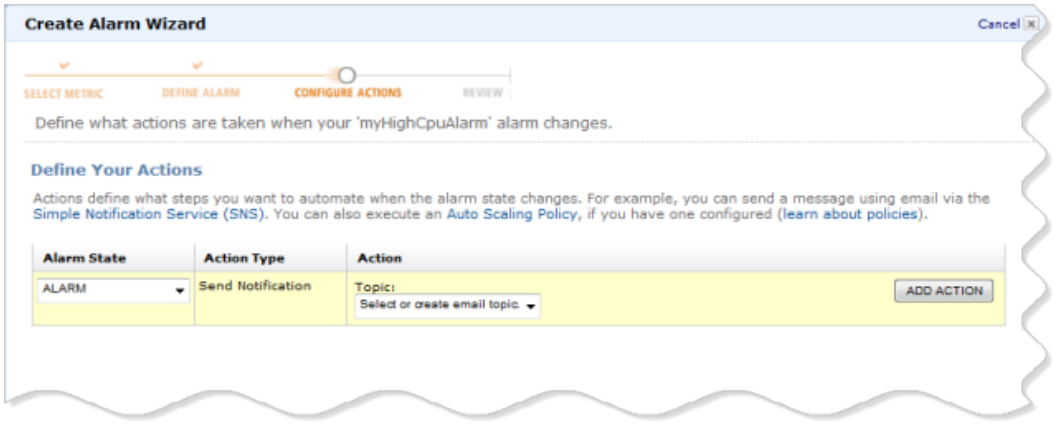
アラームの名前、説明、しきい値を定義するには

1. [DEFINE ALARM] ページで、[Name] フィールドに、アラームの名前 (例: **myTestAlarm**) を入力します。
2. [Description] フィールドには、アラームの説明 (例: **CPU　　70　　**) を入力します。
3. [Define Alarm Threshold] リストで < を選択します。
4. 最初の[**Define Alarm Threshold**] フィールドに 70 と入力し、2 番目のフィールドに **5** と入力します。

   ページ上に、しきい値が図で示されます。
5. [Continue] をクリックします。



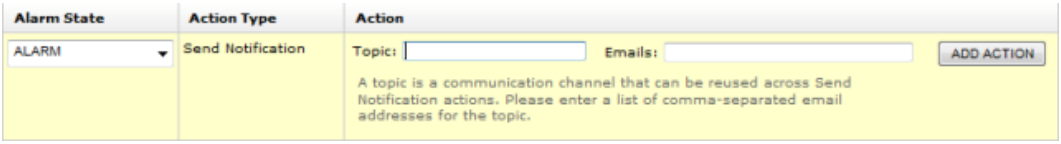
6. [Create Alarm Wizard] の [CONFIGURE ACTIONS] ページが開きます。



アラームを E メールで通知するよう設定するには

1. [CONFIGURE ACTIONS] ページで、[Alarm State] リストから [ALARM] を選択します。
2. [Topic] リストから [Create Email Topic] を選択します。

   [Topic] リストの代わりに [Topic] と [Emails] という 2 つの新しいフィールドが表示されます。
3. [Topic] フィールドに、この Amazon Simple Notification Service (Amazon SNS) トピックを説明するような名前 (例: **`myTestAlarm`**) を入力します。
4. [Emails] フィールドに、アラームが ALARM 状態になったとき通知を送信する宛先 E メールアドレスを入力します。アドレスを複数入力する場合はコンマで区切ります。



5. [ADD ACTION] をクリックします。

   アクションが保存され、[ADD ACTION] ボタンが [REMOVE] ボタンに変わります。
6. [Continue] をクリックします。
7. [Create Alarm Wizard] の [REVIEW] ページが開きます。

これでアラームの定義と動作設定は完了です。次に設定を確認してアラームを作成します。

アラーム設定を確認してアラームを作成するには

1. [Create Alarm Wizard] の [REVIEW] ページに表示されるアラーム設定を確認します。

   設定に変更を加えるには、[Edit Definition]、[Edit Metric]、または [Edit Actions] リンクを使用します。
2. アラームの作成を完了するには、[Create Alarm] をクリックします。

   確認のためのウィンドウが表示されます。
3. [Close] をクリックします。

   これでアラームが作成されました。指定したメールアドレスに、通知のオプトイン確認ページへのリンクを記載した通知メールが送信されます。オプトインすると、次にインスタンスが 70% 以下の CPU 使用率で 5 分間以上実行されたときにメールで通知されます。

クリーンアップ

チュートリアルは完了したので、作成したリソースをクリーンアップできます。必要に応じてインスタンスをカスタマイズして、使い続けることもできます。

Important

無料使用範囲を利用中でない場合は、インスタンスの起動直後から料金が発生し、インスタンスが実行中の間は、アイドル状態であっても、その時間分の料金が請求されます。

インスタンスが不要になった場合は、次のリソースをクリーンアップする必要があります。

- Amazon CloudWatch アラーム
- Elastic IP アドレス
- インスタンス

CloudWatch アラームを削除するには

1. Open the Amazon CloudWatch console at https://console.aws.amazon.com/cloudwatch/.
2. ナビゲーションペインの [Alarms] をクリックします。
3. 作成したアラームを選択し、右クリックして [Delete] をクリックします。

Elastic IP アドレスを作成した場合は、インスタンスとの関連付けを解除して開放する必要があります。

Important

Elastic IP アドレスを解放しないと、使用していないことに対して課金されます。

Elastic IP アドレスの関連付けを解除して解放するには

1. Open the Amazon EC2 console at https://console.aws.amazon.com/ec2/.
2. ナビゲーションペインで [Elastic IPs] をクリックします。
3. Elastic IP アドレスを選択して、[Disassociate] をクリックします。
4. プロンプトが表示されたら、[Yes, Disassociate] をクリックします。
5. Elastic IP アドレスを再度選択して、[Release] をクリックします。
6. プロンプトが表示されたら、[Yes, Release] をクリックします。

インスタンスを終了するということは、実質的には、そのインスタンスを削除するということです。いったん終了したインスタンスに再接続することはできません。

無料使用範囲外でインスタンスを起動した場合は、インスタンスのステータスが `shutting down` または `terminated` に変わるとインスタンスの課金が停止します。

インスタンスを終了するには

1. [Instances] ページのインスタンスリストから、終了するインスタンスを探します。
2. インスタンスを右クリックしてから、[Terminate] をクリックします。
3. 確認を求められたら、[Yes, Terminate] をクリックします。

&EC2 インスタンスでの WordPress ブログのデプロイ

このセクションでは、EC2 Windows インスタンスで WordPress ウェブサイトを作成しデプロイする手順を説明します。

Topics

前提条件

作業を始める前に、次のことを実行してください。

- Microsoft Windows Server 2008 R2 と Internet Information Services (IIS) がプリインストールされた AMI から Amazon EC2 インスタンスを起動します。EC2 インスタンスの起動方法についての詳細は、「&EC2 Windows インスタンスの使用開始 (p. 8)」を参照してください。
- AWS 無料使用範囲 (利用できる場合) を使用して、12 か月間無料で使用できる EC2 Windows *t1.micro* インスタンスを起動して使用します。AWS 無料使用範囲では、新しいアプリケーションの起動や既存のアプリケーションのテストが可能で、単に AWS を実際に使ってみてどのようなものであるか確認するために使用することができます。無料使用範囲の利用資格と特長の詳細については、AWS 無料使用範囲製品ページを参照してください。

  Important

  通常のインスタンスを起動していて、そのまま WordPress ウェブサイトをデプロイすると、終了するまで、そのインスタンスの標準の Amazon EC2 使用料が発生します。Amazon EC2 の使用料については、Amazon EC2 製品ページを参照してください。

- EC2 インスタンスを起動するセキュリティグループで、着信トラフィックに対してポート 80 (HTTP) と 3389 (RDP) を開放していることを確認します。ポート 80 は、インスタンスの外部にあるコンピュータが HTTP で接続することを許可します。ポート 80 が開放されていないと、インスタンスの

外部から WordPress サイトにアクセスできません。ポート 3389 は、リモートデスクトッププロトコルを使用してインスタンスに接続できるようにします。

- EC2 インスタンスに接続します。

Microsoft Web 配置ツールのインストール

この手順では、Microsoft IIS Web 配置ツールを使用してサーバーに WordPress をインストールして設定します。Web配置ツールを使用すると、ウェブアプリケーションやウェブサイトをIISサーバーに簡単にデプロイできます。詳細については、http://www.iis.net/downloads/microsoft/web-deploy を参照してください。

1. 前提条件 (p. 19) の条件を満たしていることを確認します。
2. Internet Explorer セキュリティ強化の構成を無効にします。
   a. EC2 インスタンスで [Start] をクリックし、[Administrative Tools] をポイントし、[Server Manager] をクリックします。
   b. [Security Information] ペインで [Configure IE ESC] をクリックします。
   c. [Administrators] の下にある [Off] をクリックして [OK] をクリックします。
   d. [Server Manager] ウィンドウを閉じます。
3. EC2 インスタンスで Internet Explorer を開き、`http://www.iis.net/download/webdeploy` に移動 ます。
4. Web 配置ツールの最新バージョンをダウンロードしてインストールします。

WordPress のインストール

Web 配置ツールをインストールできたので、これを使用して WordPress をサーバーにインストールして設定します。

WordPress をインストールするには

1. Web Platform Installer を開いて [Applications] をクリックします。
2. WordPress を選択して、[Add] をクリックし、次に [Installl] をクリックします。
3. [Prerequisites] ページで、使用するデータベースとして MySQL を選択します。MySQL データベースの管理者パスワードとして好きなパスワードを [Password] ボックスと [Confirm Password] ボックスに入力し、[Continue] をクリックします。
4. インストールするサードパーティアプリケーション、Microsoft 製品、コンポーネントのリストを確認して、[I Accept] をクリックします。Web Platform Installer がソフトウェアのインストールを終えると、新しいサイトを設定するよう求めるプロンプトが表示されます。
5. ['WordPress' application name:] ボックスのデフォルトのアプリケーションを消去して、空のままにします。他のボックスのデフォルト情報はそのままにして、[Continue] をクリックします。
6. [Yes] をクリックして、フォルダの内容が上書きされることを承認して、ウィザードを終了します。
7. WordPress の [Welcome] ページで次の情報を入力して [Install WordPress] をクリックします。
   - [Site Title] – 自分のサイトのタイトルを入力します。
   - [Username] – `admin` のままにします。
   - [Password, twice] – サイトのパスワードです。同じパスワードを 2 つ目のボックスに入力します。
   - [Your E-mail] – 自分の E メールアドレスを入力します。

- [Privacy] – 検索エンジンにサイトのインデックス作成を許可する場合は、このボックスをチェックします。

8. [Log In] をクリックします。
9. [Log In] ページで [Username] に `admin` を、[Password] に先ほど入力したサイトのパスワードを入力します。

最初のブログ記事の作成

これで、新しい WordPress サイトに最初のブログ記事を作成できるようになりました。

最初のブログ記事を作成するには

1. `http://localhost/wp-admin` にアクセスして WordPress ダッシュボードを開きます。認証情報を求められたら、[Username] に `admin` を、[Password] にサイトのパスワードを入力します。
2. [QuickPress]ボックスに以下の情報を入力します。
   - [Title] – `My First Post`
   - [Content] – `This is my first post`
3. [Publish] をクリックし、localhost にブログを公開します。通知が表示されるので、投稿を表示するか編集するかを選びます。
4. [View post] をクリックして投稿を表示します。

WordPress サイトの公開

これで WordPress ブログを localhost に表示できるようになりました。次はこのウェブサイトを EC2 インスタンスにデフォルトサイトとして公開し、他のユーザーが閲覧できるようにします。以下に、WordPress の設定を変更して、localhost の代わりに EC2 インスタンスをポイントする手順を説明します。

WordPress サイトのデフォルト設定を指定するには

1. `http://localhost/wp-admin` にアクセスして WordPress ダッシュボードを開きます。認証情報を求められたら、[Username] に `admin` を、[Password] にサイトのパスワードを入力します。
2. [Dashboard] ペインで、[Settings] をクリックします。
3. [General Settings] ページで次の情報を入力して [Save Changes] をクリックします。
   - [WordPress address （URL）] – EC2 インスタンスのパブリック DNS アドレスです。例えば、`http://ec2-203-0-113-25.compute-1.amazonaws.com` のような URL になります。
   - [Site address （URL）] – [WordPress address (URL)] に設定したものと同じ、EC2 インスタンスのパブリック DNS アドレスにします。
4. 新しいサイトを表示するには、WordPress をホストしている EC2 インスタンス以外のコンピュータでブラウザを開き、ウェブアドレスフィールドに EC2 インスタンスのパブリック DNS アドレスを入力します。WordPress サイトが表示されます。

これで、EC2 インスタンス上に WordPress サイトをデプロイできました。

Amazon EC2 インフラストラクチャ

Amazon EC2 の利用にあたっては、Amazon EC2 インフラストラクチャのコンポーネント、および従来のデータセンターとの類似点や相違点を知っておく必要があります。このセクションでは、Amazon EC2 のコンポーネントについて簡単に説明します。

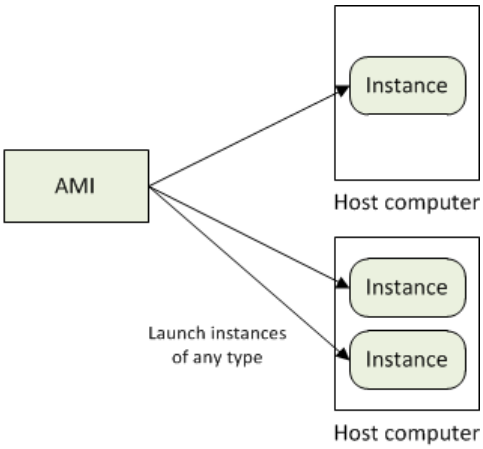
Topics

Amazon マシンイメージとインスタンス

Amazon マシンイメージ *(AMI)* は、ソフトウェア構成 (オペレーティングシステム、アプリケーションサーバー、アプリケーションなど) を記録したテンプレートです。AMI から *インスタンス*を起動します。インスタンスは AMI のコピーであり、クラウド内で仮想サーバーとして実行されます。

Amazon は、よく使用されるソフトウェア構成の AMI を数多く公開しています。加えて、AWS 開発者コミュニティのメンバーによって作成された、独自のカスタム AMI もあります。お客様自身でカスタム AMI を作成することもできます。必要なものがすべて含まれた新しいインスタンスを、すばやく簡単に起動できるようになります。例えば、ウェブサイトまたはウェブサービスに使用する場合は、AMI に含まれるものとして、ウェブサーバー、関連する静的コンテンツ、動的ページ用のコードが考えられます。この AMI からインスタンスを起動すると、ウェブサーバーが起動し、アプリケーションはリクエストを受け付け可能な状態になります。

1 つの AMI から、複数の異なるタイプのインスタンスを起動することもできます。*インスタンスタイプ*とは本質的に、インスタンスに使用されるホストコンピュータのハードウェアを決定するものです。インスタンスタイプごとに、コンピューティングとメモリの装備が異なります。インスタンスタイプを選択するときは、そのインスタンス上で実行するアプリケーションやソフトウェアに必要なメモリ量とコンピューティング能力を考慮してください。詳細については、「利用可能なインスタンスタイプ」を参照してください。次の図に示すように、1 つの AMI から複数のインスタンスを起動できます。



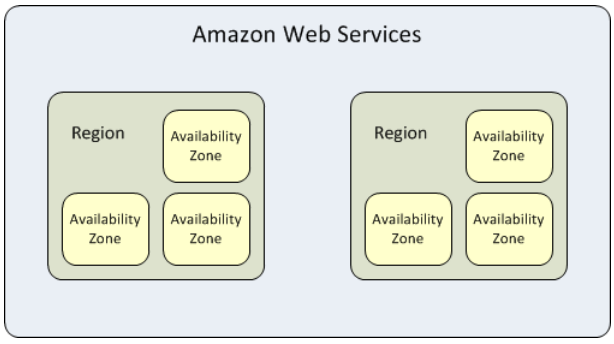
Windows インスタンスは、停止または終了させるか、エラーが発生するまで実行を続けます。インスタンスがエラーで終了した場合は、元の AMI から新しいインスタンスを起動できます。

Windows AMI とインスタンスについて詳しくは、Windows Amazon マシンイメージ (AMI) (p. 34) および Windows インスタンスタイプを参照してください。

リージョンとアベイラビリティーゾーン

Amazon は世界各地 (北米、欧州、アジアなど) にデータセンターを設置しています。これに対応して、Amazon EC2 はさまざまな *リージョン*でご利用いただけるようになっています。インスタンスをそれぞれ別のリージョンで起動すると、アプリケーションを顧客に近い場所で実行する、あるいは法規制などの要件に適合させるという要望を満たすことができます。Amazon EC2 の利用料金はリージョンによって異なります (リージョン別の料金については Amazon EC2 料金表を参照してください)。

各リージョンは、複数のそれぞれ独立した場所で構成されており、これらの場所を *アベイラビリティーゾーン*といいます。各アベイラビリティーゾーンは、他のアベイラビリティーゾーンにおける障害の影響を受けないように設計されています。また、同じリージョン内の他のゾーンには、低コスト、低レイテンシーでネットワーク接続できるようになっています。独立した利用可能ゾーンで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することができます。



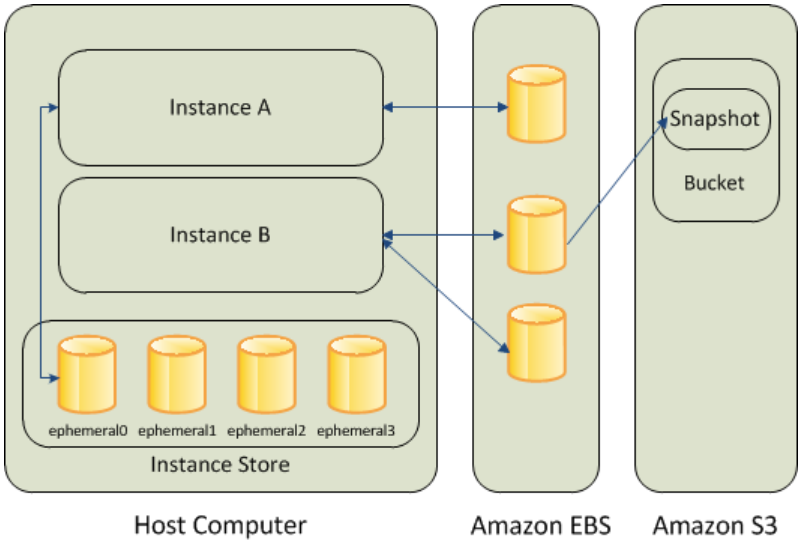
利用可能なリージョンとアベイラビリティーゾーンについて詳しくは、リージョンとアベイラビリティーゾーンの使用を参照してください。

ストレージ

Amazon EC2 を使っていると、保存したいデータが出てくる場合があります。Amazon EC2 では以下のストレージオプションを提供しています。

- Amazon Elastic Block Store (Amazon EBS)
- Amazon EC2 インスタンスストア
- Amazon Simple Storage Service (Amazon S3)

各ストレージタイプの関係を下の図に示します。

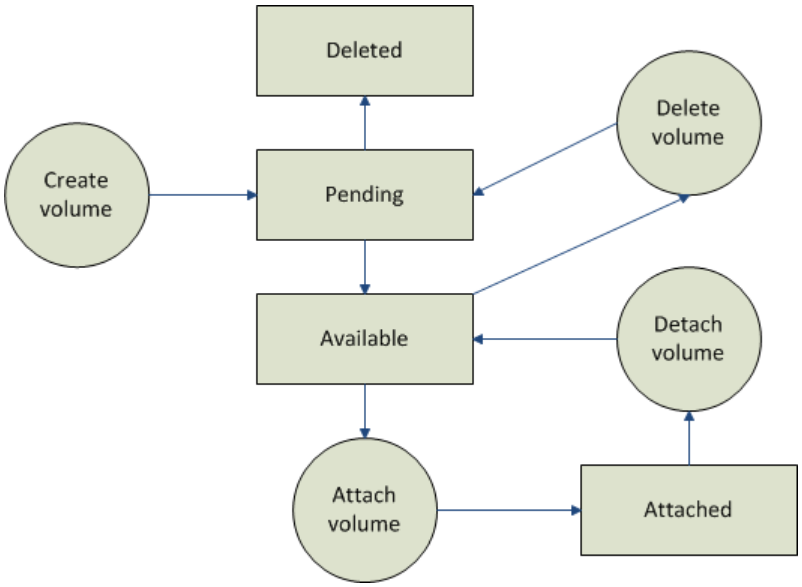


Amazon EBS ボリューム

Amazon EBS ボリュームは、ほとんどのユースケースにお勧めのストレージオプションです。Amazon EBS は、インスタンスに永続的なブロックレベルのストレージを提供します。Amazon EBS ボリュームは基本的に、実行するインスタンスにアタッチできる、物理的なハードディスクのようなものです。

Amazon EBS が特に適しているのは、アプリケーションでデータベースやファイルシステムを必要としている場合や、ブロックレベルの Raw ストレージにアクセスする場合です。

上の図に示したように、同じインスタンスに複数のボリュームをアタッチできます。また、データのバックアップコピーを保持するために、EBS ボリュームのスナップショットを作成して Amazon S3 に保存することができます。スナップショットから新しい Amazon EBS ボリュームを作成して他のインスタンスにアタッチすることもできます。また、インスタンスからボリュームをデタッチして別のインスタンスにアタッチすることもできます。次の図は、EBS ボリュームのライフサイクルを表したものです。



Amazon EBS ボリュームについて詳しくは、Amazon Elastic Block Store を参照してください。

インスタンスストア

マイクロインスタンスを除き、すべてのインスタンスタイプにインスタンスストアがあります。これは、インスタンスで利用できるブロックレベルの一時的なストレージです。これは、ホストコンピュータに物理的にアタッチされているストレージです。インスタンスストアボリューム上のデータは、関連付けられたインスタンスが停止または終了すると消滅します。インスタンスストアボリュームの詳細については、「Amazon EC2 インスタンスストレージ」を参照してください。

インスタンスストアは、安価な一時ストレージを提供するオプションです。データを永久保存する必要がない場合には、インスタンスストアボリュームが便利です。

Amazon S3

Amazon S3 はインターネット用のストレージサービスです。無制限の量のデータをウェブ上のどこからでも保管し取得することができる、シンプルなウェブサービスインターフェイスを提供します。Amazon S3 について詳しくは、Amazon S3 の製品ページを参照してください。

ルートデバイスストレージ

Amazon EC2 インスタンスを起動するとき、ルートデバイスにはそのインスタンスの起動に使用されるイメージが格納されています。

すべての AMI は、*Amazon EBS-backed* (AMI からインスタンスを起動するときのルートデバイスは Amazon EBS ボリュームである) と *Instance-store backed* (AMI からインスタンスを起動するときのルートデバイスは、Amazon S3 に格納されているテンプレートから作成されたインスタンスストアボリュームである) のいずれかに分類されます。

AMI の説明には、ルートデバイスの種類 (`ebs` か `instance store`か) も明記されています。この違いが重要なのは、AMI のタイプによって、実行できる機能が大きく異なるからです。違いについての詳細は Windows AMI のルートデバイスストレージ (p. 35) を参照してください。

ネットワークとセキュリティ

インスタンスは、EC2-Classic または EC2-VPC のいずれかのプラットフォームで起動できます。EC2-Classic で起動したインスタンスには、パブリック IP アドレスが割り当てられます。EC2-VPC で起動したインスタンスには、デフォルトの VPC で起動した場合にのみ、パブリック IP アドレスが割り当てられます。デフォルト以外の VPC で起動したインスタンスには、起動時にパブリック IP アドレスを割り当てる必要があります。EC2-Classic と EC2-VPC の詳細については、「*Amazon Elastic Compute Cloud User Guide*」の「サポートされているプラットフォーム」を参照してください。

インスタンスは、制御不能な理由により エラーを起こしたり終了したりすることがあります。インスタンスが異常終了したために別のインスタンスを起動した場合は、そのインスタンスのパブリック IP アドレスは元の IP アドレスとは異なります。ただし、お使いのアプリケーションで固定 IP アドレスが必要な場合は、Amazon EC2 から *Elastic IP アドレス*が提供されます。詳細については、「*Amazon Elastic Compute Cloud User Guide*」の「Using Instance IP Addresses」を参照してください。

インスタンスにアクセスできるユーザーの制御には *セキュリティグループ*を使います。これは、着信ネットワークファイアウォール (どのプロトコル、ポート、ソース IP 範囲にインスタンスへの到達を許可するかを指定する) に似ています。複数のセキュリティグループを作成してそれぞれに異なるルールを割り当てることもできます。その後で、各インスタンスを 1 つまたは複数のセキュリティグループに割り当てます。どのトラフィックが各インスタンスに到達できるかは、ルールを使用して決定されます。セキュリティグループを設定するときに、特定の IP アドレスや特定のセキュリティグループのみにインスタンスへのアクセスを許可するように指定することができます。セキュリティグループについては、Amazon EC2 セキュリティグループ (p. 29) を参照してください。

モニタリング、自動スケーリング、負荷分散

AWS が提供する機能でどのようなことが実現できるかを次の表にまとめました。

タスク	関連するガイド
インスタンスと Amazon EBS ボリュームの基本的な統計情報を監視します。	Amazon CloudWatch Developer Guide
Amazon EBS のキャパシティを、定義された条件に合わせて自動的に拡張または縮小します。	Auto Scaling Developer Guide
アプリケーションへのトラフィックを、複数の Amazon EC2 インスタンスに自動的に分散します。	Elastic Load Balancing Developer Guide

AWS Identity and Access Management

Amazon EC2 integrates with AWS Identity and Access Management (IAM), a service that enables you to do the following:

- Create users and groups under your AWS account
- Easily share your AWS resources between the users in your AWS account
- Assign unique security credentials to each user
- Control each user's access to services and resources
- Get a single bill for all users in your AWS account

With Amazon EC2, you can use IAM to control which users in your AWS account can create AMIs or launch instances.

For more information about IAM, see the following:

- Identity and Access Management (IAM)
- IAM Getting Started Guide
- Using IAM

利用可能な EC2 インターフェイス

AWS では、EC2 にアクセスするためのインターフェイスを各種提供しています。

AWS マネジメントコンソール

AWS マネジメントコンソールはウェブベースのシンプルな GUI です。コンソールの使い方については、&EC2 Windows インスタンスの使用開始 (p. 8) を参照してください。

コマンドラインツール (API ツール)

EC2 API をラップする Java ベースのコマンドラインクライアントです。詳しくは Windows への Amazon EC2 コマンドラインインターフェイスツールのインストール (p. 107) および Amazon Elastic Compute Cloud Command Line Reference を参照してください。

プログラミングインターフェイス

次の表は、プログラムを使用して Amazon EC2 にアクセスするときに使用できるリソースの一覧です。

リソース	説明
AWS SDK	AWS SDK には、サンプルコード、ライブラリ、ツール、ドキュメント、テンプレートが収録されています。 AWS SDK をダウンロードするには、AWS Software Development Kit (SDK) のページにアクセスしてください。
ライブラリ	開発者が独自のライブラリを提供することができます。このようなライブラリは、下記の AWS 開発者センターにあります。 • Java 開発者センター • モバイル開発者センター • PHP 開発者センター • Python 開発者センター • Ruby 開発者センター • Windows & .NET 開発者センター
Amazon EC2 API	コードから直接 Amazon EC2 API を呼び出すこともできます。 詳しくは「Making API Requests」および Amazon Elastic Compute Cloud API Reference を参照してください。

Amazon EC2 のリソースに対するアクセスの制御

Amazon EC2 の機能を使用すると、AWS のリソースや他のサービスにアクセスしたり、AWS マネジメントコンソール、コマンドラインインターフェイス (CLI) ツール、API を使用したりできます。

Topics

セキュリティ認証情報

目的	使用するもの
インスタンスに接続する	キーペア (管理者パスワードを復号化するために使用します)
Amazon EC2 コンソールを使用する	E メールアドレスとパスワード
&EC2 CLI を使用する	アクセスキー
&EC2 API を使用する	アクセスキー
AMI または EBS スナップショットを共有する	AWS アカウント ID (ハイフンなし)
Windows AMI をバンドルして Amazon S3 にアップロードする	アクセスキー
インスタンスが Amazon S3 などの他のサービスを使うことを許可する	アクセスキー (インスタンスにあります)

詳細については、「AWS Security Credentials」を参照してください。

AWS Identity and Access Management （ IAM ）

IAM の機能を使用すると、AWS アカウントのセキュリティ認証情報を共有せずに、他のユーザー、サービス、およびアプリケーションが Amazon EC2 のリソースを使用できるようになります。Amazon EC2 のリソースの完全使用または制限付き使用のどちらを許可するか選択できます。

詳細については、「*Amazon Elastic Compute Cloud User Guide*」の「Controlling Access」を参照してください。

Amazon EC2 セキュリティグループ

*セキュリティグループ*は、1 つ以上のインスタンスに到達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。インスタンスを起動する際、1 つまたは複数のセキュリティグループに割り当てることができます。セキュリティグループのそれぞれに、そのインスタンスへのトラフィックを制御するルールを追加できます。セキュリティグループルールはいつでも変更できます。新しいルールは、そのセキュリティグループが割り当てられているインスタンスすべてに自動的に適用されます。

セキュリティグループの詳細については、「*Amazon Elastic Compute Cloud User Guide*」の「EC2 Security Groups」を参照してください。

アクセスを特定の IP アドレス範囲に制限する

セキュリティグループルール作成時のデフォルトのソースは `0.0.0.0/0` です。このデフォルトでは、あらゆる IP アドレスからインスタンスへの接続が可能です。ウェブサーバーでは、だれでもウェブページを閲覧できるよう、この設定を使用するとよいでしょう。しかし RDP アクセスについては、インスタンスへのアクセスを許可する相手を制御する必要があります。このため、セキュリティグループルールを使用して、アクセスを 1 つの IP アドレスまたは IP アドレスの範囲に制限してください。サービスを使用して、ローカルコンピュータのパブリック IP アドレスを取得できます。IP アドレスを提供するサービスを検索するには、検索フレーズ「what is my IP address」を使用します。ISP 経由で、またはファイアウォールの内側から静的な IP アドレスなしで接続している場合は、クライアントコンピュータで使用されている IP アドレスの範囲を見つける必要があります。

アクセスを特定のセキュリティグループに制限する

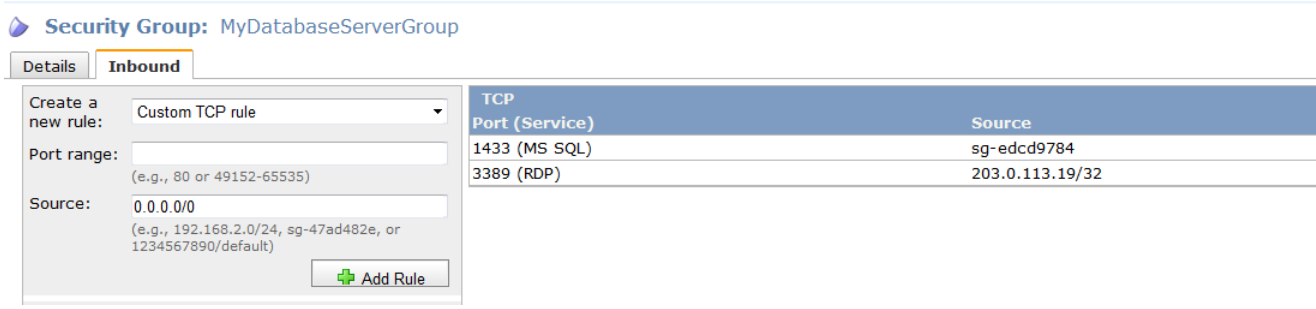
セキュリティグループルールの作成時に、セキュリティグループをソースとして指定できます。たとえば、アプリケーションが次の 2 つのインスタンスを使用するものとします。

- IIS を実行するウェブサーバー
- SQL Server を実行するデータベースサーバー

データベースサーバーに接続できるソースは、セキュリティグループ [`sg-edcd9784`] 内で起動されたウェブサーバーだけにしたいとします。

そこで、データベースサーバーインスタンス用のセキュリティグループを作成する際、ポート 1433 （MS SQL ）を開くルールを追加し、ソースを [`sg-edcd9784`] に指定します。これでデータベースサーバーは、セキュリティグループ [`sg-edcd9784`] のメンバーからの MS SQL トラフィックのみを受け付けるようになります。この例では、ウェブサーバーを実行しているインスタンスのみが、このポートのデータベースインスタンスに接続できます。

当社のデータベースサーバーに対し、静的 IP アドレス `203.0.113.19` を持つクライアントコンピュータだけが、RDP を使って接続できるようにする場合を考えてみましょう。IP アドレスを `203.0.113.19/32` と指定することができます。この CIDR ブロックでは IPv4 アドレス範囲全体が使用されるので、単一のホストを示すことになります。



Windows Server インスタンスのパスワード

Windows インスタンスに接続する際、インスタンスにアクセスする権限を持つユーザーアカウントの名前とそのパスワードを指定する必要があります。インスタンスに初めて接続するときは、管理者アカウントとデフォルトの管理者パスワードを指定します。管理者パスワードはデフォルト値以外に変更すること、またそのインスタンスに管理者権限を持つユーザーアカウントをもう 1 つ作成することをお勧めします。

Windows Server インスタンスの管理者アカウントのパスワードを忘れた場合や、パスワードの期限が切れた場合は、Amazon EC2 設定サービスを使用してパスワードをリセットすることができます。

Important

ローカル管理者アカウントを無効にしている場合、この方法でパスワードをリセットすることはできません。

このセクションでは、パスワードをリセットする必要があるインスタンスを*リセットインスタンス*と呼びます。

パスワードをリセットするための前提条件

Amazon EC2 設定サービスを使用して Windows Server インスタンスのパスワードをリセットするには、次の前提条件を満たす必要があります。

- Amazon EC2 設定サービスが、パスワードをリセットするインスタンスにインストールされている。このサービスはデフォルトで、すべての Amazon Windows AMI で有効になっています。ダウンロードすることもできます。詳細については、「EC2Config の最新バージョンのインストール (p. 48)」を参照してください。
- パスワードをリセットする必要があるインスタンスと同じアベイラビリティーゾーンに実行中の Windows Server 2003 インスタンスがあり、そのインスタンスにログインすることができる。以下の手順では、このインスタンスを*リカバリインスタンス*と呼びます。Windows Server 2003 インスタンスをお勧めするのは、起動ファイルを変更しようとしない、旧式のブートローダーを使用するためです。

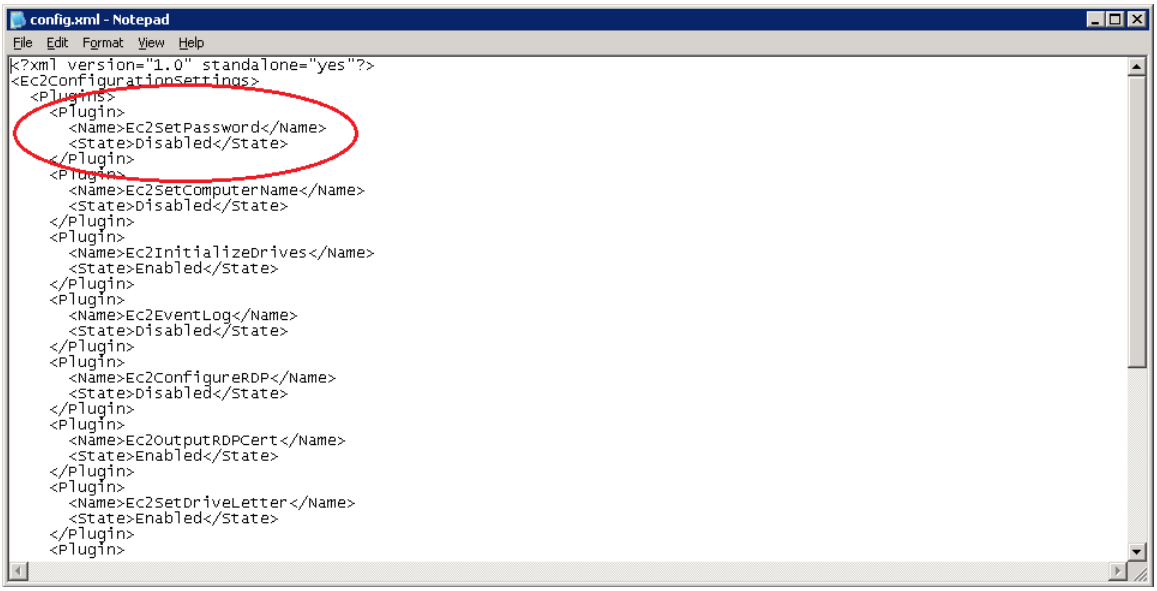
Windows Server インスタンスでの理者パスワードのリセット

リセットインスタンスの起動ボリュームにある設定ファイルを変更すると、Amazon EC2 設定サービスを使用して管理者パスワードをリセットできます。ただし、このファイルは現在ルートボリュームでないボリュームでのみ変更できます。このため、まずリセットインスタンスからルートボリュームをデ

タッチして、それをリカバリインスタンスにアタッチし、設定ファイルを変更してからルートボリュームをリセットインスタンスに再アタッチする必要があります。

管理者パスワードをリセットするには

1. Sign in to the AWS Management Console and open the Amazon EC2 console at https://console.aws.amazon.com/ec2/.
2. リセットインスタンスを停止する
   a. [Navigation] ペインの [Instances] をクリックします。
   b. リセットインスタンスを右クリックして [Stop] をクリックします。
   c. [Stop Instances] ダイアログボックスで、[Yes, Stop] をクリックします。インスタンスが停止したら、次の手順に進みます。
3. ルートボリュームをデタッチする
   a. [Navigation] ペインの [Volumes] をクリックします。
   b. ボリュームのリストでリセットインスタンスのルートボリュームを右クリックし、次に [Detach Volume] をクリックします。ボリュームのステータスが [available] に変わったら、次のステップに進みます。
4. リカバリインスタンスにボリュームをアタッチする
   a. ボリュームを右クリックして [Attach Volume] をクリックします。
   b. [Attach Volume] ダイアログボックスの [Instances] リストで、リカバリインスタンスを選択します。
   c. [Device] ボックスに **xvdf** と入力し（まだ存在しない場合）、[Yes, Attach] をクリックします。
   d. リカバリインスタンスにログインし、ボリュームをオンラインに設定します。詳細については、「Make the Volume Available on Windows」を参照してください。
5. リセットボリュームの設定ファイルを変更する
   a. リカバリインスタンスで Notepad などのテキストエディタを使って、ボリュームから `\Program Files\Amazon\Ec2ConfigService\Settings\config.xml` ファイルを開きます。
   b. ファイルの先頭の、<Plugin>、<Name>Ec2SetPassword</Name> の下で、<State>Disabled</State> を <State>Enabled</State> に変更し、ファイルを保存します。



6. ボリュームをリカバリインスタンスからデタッチします。

   a. リカバリインスタンスで、ボリュームをオフラインに設定します。
   b. [Navigation] ペインの [Volumes] をクリックします。
   c. ボリュームのリストでボリュームを右クリックし、[Detach Volume] をクリックします。ボリュームのステータスが [available] に変わったら、次のステップに進みます。

7. リセットインスタンスにボリュームを再アタッチする

   a. ボリュームを右クリックして [Attach Volume] をクリックします。
   b. [Attach Volume] ダイアログボックスの [Instances] ドロップダウンリストで、ボリュームを選択します。
   c. [Device] ボックスで、`/dev/sda1` と入力し、[Yes, Attach] をクリックします。

8. リセットインスタンスを再起動する

   a. [Navigation] ペインの [Instances] をクリックします。
   b. リセットインスタンスを右クリックして [Start] をクリックします。
   c. [Start Instances] ダイアログボックスで、[Yes, Start] をクリックします。

      Important

      インスタンスに新しい IP IP アドレスと DNS 名が割り当てられます。

   d. 新しい DNS 名でリモートデスクトッププロトコル接続を更新します。

9. 新しいデフォルトパスワードを取得する

   a. [Navigation] ペインの [Instances] をクリックします。
   b. リセットインスタンスを右クリックし、[Get Windows Password] をクリックします。

c. [Retrieve Default Windows Administrator Password] ダイアログボックスで、[Browse] をクリックし、適切なプライベートキー（.pem）ファイルを選択します。

d. [Decrypt Password] をクリックし、復号化されたパスワードを使用してリセットインスタンスに管理者として ログインします。

Windows Amazon マシンイメージ（ AMI ）

Windows Amazon マシンイメージ（ AMI ）とは、Amazon EC2 Windows インスタンスを起動するために必要なすべての情報が入ったテンプレートです。いわば Windows Server とその他サーバーの実行に必要なソフトウェアを含むブートパーティションのスナップショットです。AMI は Windows インスタンスを起動するときに指定します。インスタンスとはクラウドで実行する仮想サーバーです。

AWS Windows AMI の詳細については、Amazon Windows AMI の基本 (p. 34)を参照してください。

AWS マネジメントコンソールを使うと、指定した条件を満たす Windows AMI を検索できます。例えば、AWS が提供する Windows AMI だけを表示したり、EC2 コミュニティが提供する Windows AMI だけを表示したりできます。Windows AMI の選択について詳しくは、Windows AMI の選択 (p. 38) を参照してください。

パブリック AMI の中に、ニーズに適合するものがあるかもしれません。パブリック AMI をカスタマイズして、自分で利用するために新しい AMI として保存することもできます。詳細については、独自の Windows AMI の作成 (p. 49) を参照してください。

新しく作成した AMI は、非公開にして自分だけで利用することもできますし、他の AWS アカウントを指定して共有することもできます。または、カスタマイズした AMIを公開して Amazon EC2 コミュニティで利用できるようにすることも可能です。安全かつセキュアで有用なパブリック AMI を作成するのはごく簡単で、いくつかのシンプルなガイドラインに従うだけです。共有 AMI の作成と使用について詳しくは、共有 Windows AMI (p. 54) を参照してください。

有料の AMI もあります。サードパーティから購入したり、Red Hat などの企業とサービス契約を結ぶことで入手したりする AMI です。AMI を他の開発者に販売することをお考えの場合は、Amazon DevPay を参照してくだ い。独自の AMI を作成して他の Amazon EC2 ユーザーに販売することもできます。有料の AMI の使用方法や販売方法の詳細については、有料 Windows AMI (p. 60) を参照してください。

AMI には分類や管理の目的で、任意のタグを付けられます。詳しくは、Using Tags を参照してください。

Amazon Windows AMI の基本

アマゾン ウェブ サービス（ AWS ）では、Windows プラットフォーム用にソフトウェア設定済みのパブリック AMI を各種提供しています。これを利用すれば、すぐに Amazon EC2 を使ったアプリケーション構築とデプロイを始められます。まず要件に適合する AMI を選び、次にその AMI を使って イン

スタンスを起動します。インスタンスには、他の Windows サーバーと同様に、リモートデスクトップ接続を使って接続します。

AWS では現在、以下のバージョンの Windows をベースにした AMI を公開しています。

- Microsoft Windows Server 2012 (64 ビット)
- Microsoft Windows Server 2008 R2 (64 ビット)
- Microsoft Windows Server 2008 (64 ビット)
- Microsoft Windows Server 2008 (32 ビット)
- Microsoft Windows Server 2003 (64 ビット)
- Microsoft Windows Server 2003 (32 ビット)

また AWS は、SQL Server、SQL Server Express、Internet Information Services (IIS)、ASP.NET を含む、一般的に利用可能な AMI も提供しているため、すぐに使い始めたい方に有用です。これら AMI の 1 つまたは複数を使ってアプリケーションをデプロイできます。例えば、SQL Server Express、IIS、および ASP.NET を含む AWS Windows AMI を使って起動したインスタンスで、ウェブと ASP.NET アプリケーションを実行できます。SQL Server を含む AWS Windows AMI からインスタンスを起動すれば、インスタンスをデータベースサーバーとして稼働させることもできます。または、基本的な Windows AMI からインスタンスを起動し、必要なソフトウェアとアプリケーションをインストールしてカスタマイズし、カスタマイズしたインスタンスを AMI として登録することもできます。そのカスタマイズした AMI を使用すれば、指定したソフトウェアとアプリケーションを備えたインスタンスを起動できます。

AWS Windows AMI は年に数回更新されます。AWS AMI を更新する際、それまでの AMI を停止し、新しい AMI と AMI ID に差し替えます。更新後の AMI を探すには、ID ではなく名前を使います。通常、AMI 名の基本構造は同じで、最後に新しい日付が追加されます。クエリまたはスクリプトを使用して AMI を名前で検索し、正しい AMI であることを確認してから、インスタンスを起動することができます。

パブリック AMI には、AWS が提供するものの他に、AWS 開発者コミュニティが公開した AMI もあります。AWS または信頼できるソースが提供する Windows AMI のみを使用することをお勧めします。

AWS が承認する Microsoft Windows AMI のリストは、Amazon マシンイメージ (AMI) にアクセスして、プラットフォームとして Windows を選択してください。表示されたリストから任意の AMI をクリックすると、その詳細が表示されます。

Windows AMI のルートデバイスストレージ

Amazon EC2 Windows インスタンスを起動する AMI には、Instance store-backed (インスタンスストアに格納されたもの) と、Amazon Elastic Block Store-backed (Amazon EBS に格納されたもの) があります。このセクションでは、この 2 種類の AMI の違いについて説明します。AMI を選択する前に、この相違点を考慮することは重要です。

instance store-backed AMI から起動したインスタンスは、インスタンスストアボリュームをルートデバイスとして使います。instance store-backed AMI のルートデバイスボリュームのイメージは、まず Amazon S3 に格納されます。instance store-backed AMI を使ってインスタンスを起動すると、ルートデバイスのイメージは Amazon S3 からインスタンスのルートパーティションにコピーされます。以降はこのルートデバイスボリュームを使用してインスタンスを起動します。

Amazon EBS-backed AMI から起動したインスタンスは、Amazon EBS ボリュームをルートデバイスとして使用します。Amazon EBS-backed AMI のルートデバイスボリュームは、Amazon EBS スナップショットです。Amazon EBS-backed AMI を使ってインスタンスを起動すると、その EBS スナップショットからルート EBS ボリュームが作成され、インスタンスにアタッチされます。以降はこのルートデバイスボリュームを使用してインスタンスを起動します。

Amazon EC2 コンソールの navigation ペインで [AMIs] を選択すると、[Root Device Type] 列に、各 AMI が Instance store-backed（`instance-store`）か EBS-backed（`ebs`）かのどちらであるか表示されます。

次の表では、instance store-backed AMI と Amazon EBS-backed AMI の違いをまとめています。

特徴	Amazon EBS-Backed	Amazon Instance Store-Backed
起動時間	通常 1 分以内	通常 5 分以内
サイズ制限	1 TiB	10 GiB
ルートデバイス	Amazon EBS ボリューム	インスタンスストアボリューム
データの永続性	インスタンスにエラーが発生した後も、インスタンスの終了後もデータを保持できる	インスタンスを終了するまでデータを保持できる
アップグレード	インスタンスの停止中に、インスタンスタイプ、カーネル、RAM ディスク、およびユーザーデータが変更可能	インスタンスの属性は、インスタンスを終了するまで固定
料金	インスタンスの使用、Amazon EBS ボリュームの使用、Amazon EBS スナップショット（AMI ストレージ）	インスタンスの使用と Amazon S3（AMI ストレージ）
停止状態	停止状態（インスタンスは実行されていないが、Amazon EBS に保持されている）にできる	停止状態にできない

AWS Windows AMI の構成

AWS Windows AMI の構成は、Microsoft が提供するメディアによってインストールされる Windows Server とできるだけ同じに合わせてあります。ただし、インストールのデフォルト設定にいくつかの違いがあります。Amazon EC2 Windows AMI には、EC2Config サービスと呼ばれる追加のサービスがインストールされています。

EC2Config サービスはローカルシステムアカウントで動作し、主に初期セットアップで使用されます。EC2Config は、インスタンスの起動時に以下のタスクを実行します。

- ホスト名をプライベート DNS 名に設定する
- 管理者アカウントにランダムな初期パスワードを生成し設定する
- インスタンスにアタッチされたすべてのドライブを初期化しフォーマットする
- リモートデスクトップ用のホスト証明書を生成しインストールする
- インスタンスのクロックをタイムサーバーと同期する

Windows インスタンスを初期構成で起動後、AMI をカスタマイズして独自の AMI を作成するプロセスの中で、EC2Config サービスを使って構成設定を変更できます。カスタマイズした AMI から起動したインスタンスは新しい構成で起動します。EC2Config サービスのバイナリ、および新規 Windows AMI の設定に必要なその他のツールは、32 ビットインスタンスでは `%ProgramFiles%\Amazon` ディレクトリ、64 ビットインスタンスでは `%ProgramFiles(x86)%\Amazon` ディレクトリに含まれています。詳細については、「独自の Windows AMI の作成 (p. 49)」を参照してください。

Xen ドライバ

AWS Windows AMI には、Xen 仮想ハードウェアにアクセスできるようにするためのドライバー式が含まれています。Amazon EC2 は、これらのドライバを使用して、インスタンスストアと Amazon Elastic Block Store（Amazon EBS）ボリュームをデバイスにマップします。

RedHat ドライバ用のソースファイルは、32 ビットインスタンスでは `%ProgramFiles%\RedHat` ディレクトリ、64 ビットインスタンスでは `%ProgramFiles(x86)%\RedHat` ディレクトリにあります。RedHat Paravirtualized ネットワークドライバである `rhelnet` と、RedHat SCSI ミニポートドライバである `rhelscsi` の 2 つがあります。

Citrix ドライバは、32 ビットインスタンスでは `%ProgramFiles%\Citrix` ディレクトリ、64 ビットインスタンスでは `%ProgramFiles(x86)%\Citrix` ディレクトリにあります。

Citrix に付属するドライバコンポーネントは、`HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\services` にあります。具体的には次のとおりです。

- xenevtchn
- xeniface
- xennet
- xennet6
- xensvc
- xenvbd
- xenvif

Citrix には、Windows サービスとして実行する、 XenGuestAgent というドライバコンポーネントも付属しています。これは、起動時の時刻の同期、API からのシャットダウンや再起動の各イベントなどのタスクを処理します。管理サービスは、コマンドラインで「`services.msc`」と入力することでアクセスできます。

既存の AMI の RedHat ドライバを Citrix ドライバにアップグレードする方法の詳細については、Windows AMI での PV ドライバのアップグレード (p. 119) を参照してください。

インスタンスを最新の状態に維持

Windows インスタンスは、初回起動時に、その時点で最新のセキュリティアップデートをすべて適用した状態になっています。しかし、いったんインスタンスを起動したら、AMI 作成時点より後に公開されたアップデートも含め、以降のアップデートを実施するのはユーザーの責任です。Microsoft のアップデートは、Windows Update サービスまたはインスタンスで提供されている Automatic Updates ツールを使用してデプロイできます。また、自身でデプロイしたサードパーティソフトウェアも、それぞれのソフトウェアに適したメカニズムを使ってすべて最新の状態に維持する必要があります。Windows インスタンスを起動するたびに、最初に Windows Update サービスを実行することをお勧めします。

Note

アップデートを適用した後、Amazon EC2 Windows インスタンスを再起動します。instance store-backed インスタンスでも、Amazon EBS-backed インスタンスでも、再起動は同じように動作します。詳細については、「Windows AMI のルートデバイスストレージ (p. 35)」を参照してください。

サポート

基本の AWS Windows AMI のインストールと使用に関するサポートは、AWS プレミアムサポートの加入者向けサービスとなっています。詳細については、AWS サポートを参照してください。

AWS Windows AMI に関する質問がありましたら、Amazon EC2 フォーラムに投稿することをお勧めします。

不具合はプレミアムサポートまたは Amazon EC2 フォーラムにご報告ください。

Windows AMI の選択

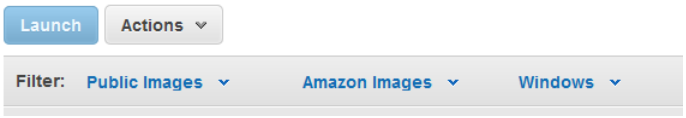
Amazon マシンイメージ (AMI) は、Amazon EC2 の基本的な構成要素です。EC2; を使って何かをするには、まず最初に AMI を選択する必要があります。AMI は AWS または Amazon EC2 コミュニティによって提供されます。独自の AMI を作成することもできますが、独自の AMI を作成するには、基本の AMI の 1 つを使用する必要があります。

AMI を選定し終えたら、その AMI ID を記録しておきます。AMI ID はインスタンスを起動して接続するときに使います。インスタンスの起動について詳しくは、Windows インスタンスを起動する (p. 9) を参照してください。Windows インスタンスへの接続について詳しくは、Windows インスタンスへの接続 (p. 12) を参照してください。

AWS マネジメントコンソールの使用

利用可能な AMI の一覧を表示するには

1. Sign in to the AWS Management Console and open the Amazon EC2 console at https://console.aws.amazon.com/ec2/.
2. ナビゲーションペインの [AMIs] をクリックします。
3. (オプション) [Filter] オプションを使用して、表示された AMI の一覧を操作できます。例えば、Amazon が提供するすべての Windows AMI を表示するには、[Public Images]、[Amazon Images] を選択し、[Filter] リストから [Windows] を選択します。



4. AMI の [Go to Details Page] (虫眼鏡) をクリックすると、別画面にその AMI のプロパティが表示されます。

   AMI を選択するときは、その AMI が instance store-backed か Amazon EBS-backed かの違いが重要になります。使用目的に合わせて AMI タイプを選んでください。詳細については、「Windows AMI のルートデバイスストレージ (p. 35)」を参照してください。

コマンドラインツールの使用

Amazon EC2 では、Amazon EC2 Query API をラップする Java ベースのコマンドラインクライアントを提供しています。このセクションのサンプルコマンドを試すには、コマンドラインツールをインストールする必要があります。コマンドラインツールのインストールについて詳しくは、Windows への Amazon EC2 コマンドラインインターフェイスツールのインストール (p. 107) を参照してください。

目的に合った AMI を見つけるには

- ec2-describe-images コマンドを使うと、求める条件に合う AMI の一覧を表示できます。

  以下のコマンドでは、AWS 所有のすべての Windows AMI を表示します。この出力例は、AWS Windows AMI の一部です。

```
C:\> ec2-describe-images -o amazon --filter "platform=windows"

IMAGE ami-c941efa0 amazon/Windows_Server-2008-SP2-English-64Bit-Base-
2013.05.15  amazon  available  public  x86_64
machine  windows  ebs  hvm xen
BLOCKDEVICEMAPPING  EBS /dev/sda1  snap-b81a74c9  30  true  standard
IMAGE ami-2b41ef42 amazon/Windows_Server-2008-R2_SP1-English-64Bit-Base-
2013.05.15  amazon  available  public  x86_64
machine  windows  ebs  hvm  xen
BLOCKDEVICEMAPPING  EBS /dev/sda1  snap-f00e6081  30  true  standard
IMAGE ami-b340eeda amazon/Windows_Server-2008-R2_SP1-English-64Bit-
SQL_2008_R2_SP1_Express-2012.07.11  amazon  available  public  x86_64
machine  windows  ebs  hvm  xen
BLOCKDEVICEMAPPING  EBS /dev/sda1  snap-0e2d437f  30  true  standard
IMAGE   ami-a8e705c1    ec2-paid-ibm-images/ibm-infosphere-is-winclient.mani
fest.xml    amazon  available public  [devpay: EC129708}
i386 machine windows instance-store  hvm  xen
IMAGE   ami-df20c3b6    ec2-public-windows-images/Server2003r2-i386-Win-
v1.07.manifest.xml      amazon  available  public  i386
machine  windows  instance-store  hvm  xen
IMAGE   ami-dd20c3b4    ec2-public-windows-images/Server2003r2-x86_64-Win-
v1.07.manifest.xml    amazon  available  public  x86_64
machine  windows  instance-store  hvm  xen
```

Tip

一覧をフィルタリングして、指定した条件に合う AMI だけを返すようにもできます。結果をフィルタリングする方法については、Amazon Elastic Compute Cloud Command Line Reference の *ec2-de cribe-images* を参照してください。

EC2Config サービスを使用した Windows インスタンスの設定

AWS Windows AMI には、追加サービスの EC2Config サービスがアマゾン ウェブ サービスによってインストールされています。使用は任意ですが、このサービスは他の手段では利用できない高度な機能を提供します。このサービスは LocalSystem アカウントで動作し、インスタンスでタスクを実行します。バイナリおよびその他のファイルは `%ProgramFiles%\Amazon\EC2ConfigService` ディレクトリに含まれます。

EC2Config サービスはインスタンスが起動したときに開始します。インスタンスの初回起動時、およびユーザーがインスタンスを停止し再開するたびに、タスクを実行します。オンデマンドでタスクを実行させることもできます。タスクには自動的に有効化されるものもありますが、手動で有効化しなければならないものもあります。EC2Config は設定ファイル群を使って操作を制御します。設定ファイル群に変更を加えるには、GUI ツールを使うか、XML ファイルを直接編集します。

EC2Config サービスは Sysprep という Microsoft ツールを実行します。このツールを利用すると、再利用可能なカスタマイズされた Windows イメージを作成できます。Sysprep の詳細については、Sysprep Technical Reference を参照してください。

EC2Config は Sysprep を呼び出すときに、`EC2ConfigService\Settings` にある設定ファイル郡を使って実行する操作を決定します。このファイル群を編集するには、[Ec2 Service Properties] ダイアログボックスで間接的に行うか、XML エディタまたはテキストエディタで直接行います。ただし一部の高度な設定は [Ec2 Service Properties] ダイアログボックスに表示されないため、変更するにはファイルを直接編集する必要があります。

インスタンスの設定を更新した後で、そのインスタンスから AMI を作成した場合、その AMI から起動されるすべてのインスタンスには、更新後の新しい設定が適用されます。AMI の作成の詳細については、Amazon EBS-Backed Windows AMI の作成 (p. 50) を参照してください。

Topics

EC2Config のタスクの概要

EC2Config は、インスタンスの初回起動時に複数の初期起動タスクを実行し、その後、それらを無効にします。これらのタスクを再び実行するには、明示的に有効化した後でインスタンスをシャットダウンするか、手動で Sysprep を実行する必要があります。初回起動時のタスクには以下のものがあります。

- コンピュータ名を（プライベート DNS 名と一致するように）設定する
- 管理者アカウントに、ランダムに生成した暗号化パスワードを設定する
- リモートデスクトップに使用されるホスト証明書を生成しインストールする
- オペレーティングシステムパーティションを動的に拡張する
- 指定されたユーザーデータ（インストールされている場合は CloudInit.NET も）を実行する

EC2Config は、インスタンスが起動するたびに次のタスクを実行します。

- Windows ライセンス認証状況を確認し、必要に応じてライセンス認証を行う
- key management server（KMS）を設定し、Windows のライセンス認証を行う
- Amazon EBS ボリュームおよびインスタンスストアボリュームをフォーマットおよびマウントし、ボリューム名をドライブ文字にマップする
- インスタンスのクロックをタイムサーバーと同期する
- トラブルシューティングに役立つよう、イベントログのエントリをコンソールに出力する
- コンソールに Windows の準備が完了した旨の通知を出力する
- デスクトップ背景に壁紙情報を表示する
- 複数の NIC がアタッチされているとき、プライマリネットワークアダプタにカスタムルートを追加して、IP アドレス 169.254.169.250、169.254.169.251、および 169.254.169.254 を有効にする。これらのアドレスは Windows ライセンス認証が使用し、またユーザーがインスタンスのメタデータにアクセスする際にも使用します。

インスタンスの実行中、ユーザーは EC2Config にリクエストを送信して以下のタスクをオンデマンドで実行させることができます。

- Sysprep を実行し、インスタンスをシャットダウンして、ユーザーがそこから AMI を作成できるようにする詳細については、「Amazon EBS-Backed Windows AMI の作成 (p. 50)」を参照してください。

Ec2 Service Properties

[Ec2 Service Properties] ダイアログボックスを使って各種設定を有効化または無効化する手順を以下に説明します。

[Ec2 Service Properties] ダイアログボックスを使用して設定を変更するには

1. Windows インスタンスを起動して接続します。
2. [Start] メニューから [All Programs] をクリックし、次に [EC2ConfigService Settings] をクリックします。
3. [Ec2 Service Properties] ダイアログボックスの [General] タブで、次の設定の有効/無効を切り替えることができます。

   [Set Computer Name]
   インスタンスのホスト名を、インスタンスの IP アドレスに基づく一意の名前に設定し、起動後に一度再起動します。ホスト名を自分で指定する場合や、既存のホスト名を変更しない場合は、この設定を有効化しないでください。

   [User Data]
   ユーザーデータの実行により、初回起動時にスクリプトをインスタンスのメタデータに挿入できます。インスタンスから、http://169.254.169.254/latest/user-data/ でユーザーデータを読むことができます。この情報はインスタンスが存続する限り変更されません。インスタンスの停止時、起動時も保持され、インスタンスが終了するまで削除されません。

   大きいスクリプトを使用する場合は、ユーザーデータを使ってスクリプトをダウンロードして実行することをお勧めします。

   EC2Config がユーザーデータを実行するためには、次のいずれかの特殊タグでスクリプト行を囲む必要があります。

   `<script></script>`
   cmd.exe プロンプトで実行できる任意のコマンドを実行します。

   例: `<script>dir > c:\test.log</script>`

   `<powershell></powershell>`
   PowerShell プロンプトで実行できる任意のコマンドを実行します。

   AWS Tools for Windows PowerShell を含む AMI を使用する場合、これらのコマンドレットも使用できます。インスタンスの起動時に IAM 役割を指定した場合は、コマンドレットに認証情報を指定する必要はありません。インスタンスで実行するアプリケーションは役割の認証情報を使って、Amazon S3 バケットなどの AWS リソースにアクセスできるからです。

   例:
   ```
   <powershell>Read-S3Object -BucketName myS3Bucket -Key
   myFolder/myFile.zip -File c:\destinationFile.zip</powershell>
   ```

   `script` タグと `powershell` タグが両方存在する場合、タグの順番に関わらず、まずバッチスクリプトが実行され、次に PowerShell スクリプトが実行されます。

   EC2Config では、ユーザーデータが base64 エンコーディングを利用できる必要があります。ユーザーデータが base64 エンコーディングで利用できない場合、EC2Config は `script` タグ、または `powershell` タグを見つけられず実行できなかったというエラーをログに出力します。エンコーディングが正しくない場合は、次の例を参考に、PowerShell を使用してエンコーディングを設定できます。

```
$UserData = [System.Convert]::ToBase64String([System.Text.Encoding]::AS
CII.GetBytes($Script))
```

初回起動

デフォルトでは、すべての Amazon AMI で初回起動時のユーザーデータの実行が有効になっています。EC2Config で [Shutdown with Sysprep] をクリックすると、[User Data] チェックボックスの設定に関わらず、ユーザーデータの実行は再度有効になります。

ランダムのパスワードが生成された場合のみ、ローカル管理者ユーザーのもとでユーザーデータの実行が実施されます。これは、EC2Config がパスワードを生成し、認証情報の認識は短期間 (コンソールに送信するまで) しか行われないためです。EC2Config はパスワードの変更を記録したり、追跡したりしないので、ユーザーがランダムのパスワードを生成しない場合、ユーザーデータの実行は EC2Config サービスアカウントによって行われます。

2 回目以降の起動

ユーザーデータプラグインは、初回起動後に自動的に無効になるため、再起動時にユーザーデータを保持するには、次のいずれかのアクションを実行する必要があります。

- schtasks.exe /Create を使用してシステムの起動時に実行するスケジュールタスクをプログラム的に作成し、そのスケジュールタスクで `C:\Program Files\Amazon\Ec2ConfigServer\Scripts\UserScript.ps1` にあるユーザーデータスクリプト(または他のスクリプト) を実行させる。
- 次のようなスクリプトを使って、`Settings.xml` のユーザーデータプラグインをプログラムで再度有効にする

```
<powershell>
$EC2SettingsFile="C:\Program Files\Amazon\Ec2ConfigService\Settings\Con
fig.xml"
$xml = [xml](get-content $EC2SettingsFile)
$xmlElement = $xml.get_DocumentElement()
$xmlElementToModify = $xmlElement.Plugins

foreach ($element in $xmlElementToModify.Plugin)
{
    if ($element.name -eq "Ec2SetPassword")
    {
        $element.State="Enabled"
    }
    elseif ($element.name -eq "Ec2HandleUserData")
    {
        $element.State="Enabled"
    }
}
$xml.Save($EC2SettingsFile)
</powershell>
```

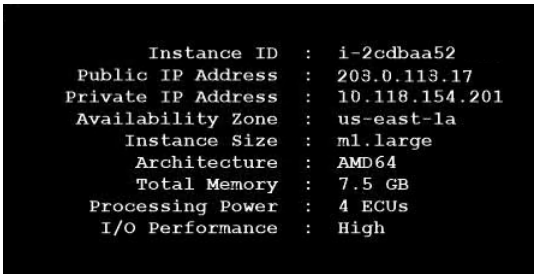
- EC2Config バージョン 2.1.10 から、<persist>true</persist> を使って、ユーザーデータの実行後にプラグインを再度有効にすることができるようになりました。

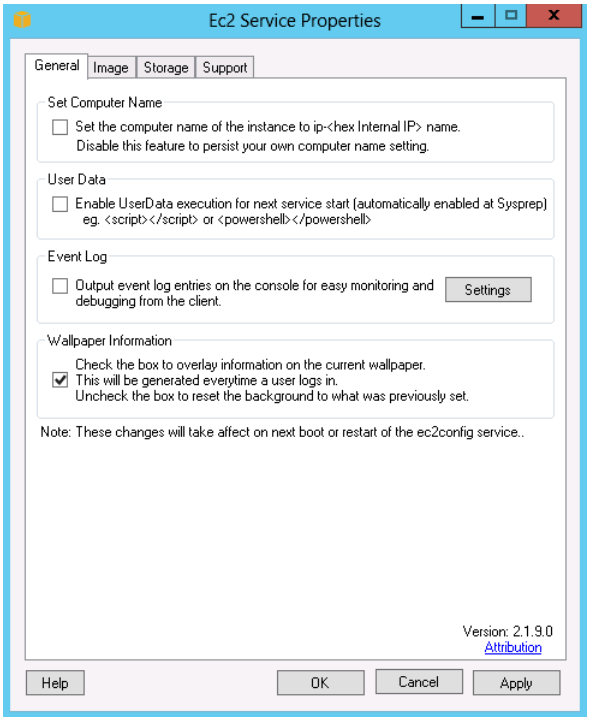
[Event Log]
起動中にイベントログのエントリをコンソールに表示する機能を有効にして、モニタリングやデバッグを容易にします。

[Settings] をクリックすると、コンソールに出力するログエントリにフィルタを指定できます。デフォルトでは、System イベントログから 3 つの最新エラーエントリがコンソールに出力されます。

[Wallpaper Information]
デスクトップの背景にシステム情報を表示する機能を有効化します。デスクトップの背景に表示される情報は、設定ファイル `EC2ConfigService\Settings\WallpaperSettings.xml` で制御します。

以下はデスクトップの背景に表示される情報のサンプルです。





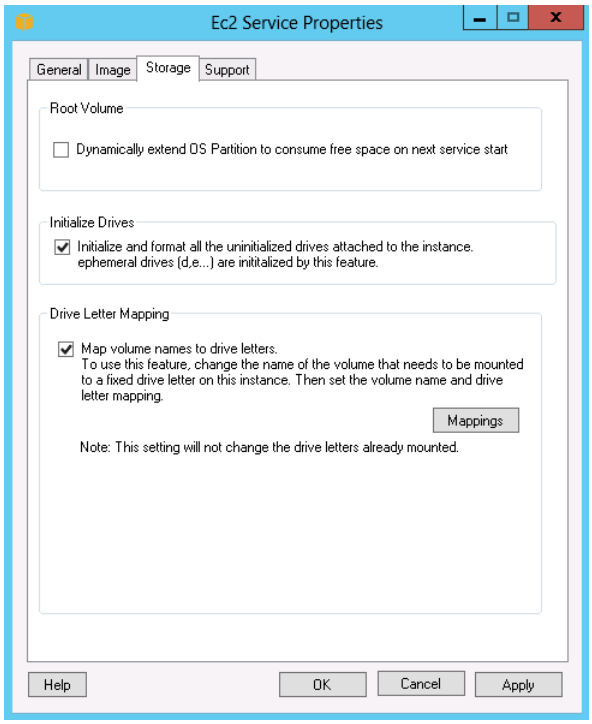
4. [Storage] タブをクリックします。有効化または無効化できる設定は以下のとおりです。

[Root Volume]
ディスク 0/ボリューム 0 が未使用の領域を含むように、動的に拡張します。独自のサイズを指定したルートデバイスボリュームからインスタンスを起動するときに便利です。

[Initialize Drives]
インスタンスの起動中に、インスタンスにアタッチされた全インスタンスストアボリュームのフォーマットとマウントを行います。

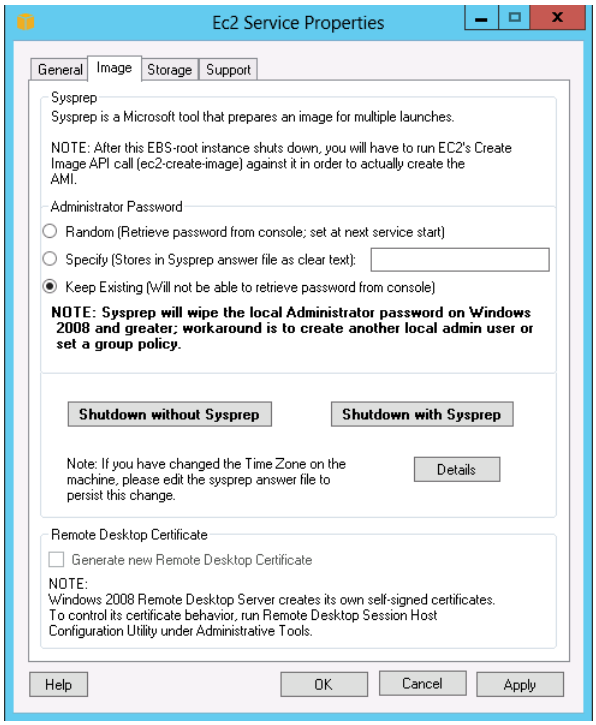
[Drive Letter Mapping]
デフォルトではシステムが、インスタンスにアタッチされたボリュームにドライブ文字をマッピングします。システムは任意のドライブ文字を選択できます。ボリュームのドライブ文字を選択するには、[Mappings] をクリックします。[DriveLetterSetting] ダイアログボックスで、[Volume Name] と [Drive Letter] の値をボリュームごとに指定した後、[OK] をクリックします。既に使用されているドライブ文字との衝突を避けるために、アルファベットの後ろの方（Z:、Y:、など）を選択することをお勧めします。

ドライブ文字のマッピングを指定して、指定したボリューム名の 1 つと同じラベルを持つボリュームをアタッチすると、EC2Config は指定したドライブ文字をボリュームに自動的に割り当てます。ただし、ドライブ文字がすでに使用されている場合、ドライブ文字のマッピングは失敗します。EC2Config は、ドライブ文字のマッピングを指定したときにマウント済みだったボリュームのドライブ文字を変更しません。



5. 設定をいったん保存して後で作業を再開するには、[OK] をクリックして [Ec2 Service Properties] ダイアログボックスを閉じます。

   インスタンスのカスタマイズが完了しており、そのインスタンスから AMI を作成する準備ができている場合は、[Image] タブをクリックします。Administrator パスワードのオプションを選択してから、[Shutdown with Sysprep] または [Shutdown without Sysprep] をクリックします。EC2Config は、選択したパスワードオプションに基づいて設定ファイルを編集します。



Sysprep を実行しインスタンスをシャットダウンするかどうか確認を求められたら [Yes] をクリックします。これにより、EC2Config が Sysprep を実行します。次に、ユーザーは自動的にログオフさせられ、インスタンスがシャットダウンします。Amazon EC2 コンソールの [Instances] ページを見ると、インスタンスの状態が `running` から `stopping` に、そして `stopped` へ変わるのがわかります。この時点で、インスタンスから AMI を安全に作成できます。

コマンドラインから Sysprep ツールを手動で呼び出すには、次のコマンドを使います。

```
%ProgramFiles%\Amazon\Ec2ConfigService\ec2config.exe -sysprep
```

ただし、この操作は慎重に行ってください。`Ec2ConfigService\Settings` フォルダで指定した XML ファイルオプションが正しくないと、インスタンスに接続できなくなる場合があります。設定ファイルについて詳しくは、EC2Config の設定ファイル群 (p. 45) を参照してください。コマンドラインから Sysprep を設定して実行する例については、`EC2ConfigService\Scripts\InstallUpdates.ps1` を参照してください。

EC2Config の設定ファイル群

以下の設定ファイルは `Ec2ConfigService\Settings` ディレクトリにあり、編集して設定を変更できます。

- `ActivationSettings.xml` – key management server （ KMS ）を使 た製品ライセンス認証を制御します。
- `BundleConfig.xml` – EC2Config が AMI 作成のためにインスタンスを準備する方法を制御します。
- `Config.xml` – 主要な設定を制御します。
- `DriveLetterConfig.xml` – ドライブ文字のマッピングを制御します。
- `EventLogConfig.xml` – インスタンスの起動中、コンソールに表示されるイベントログ情報を制御します。
- `WallpaperSettings.xml` – デスクトップの背景に表示される情報を制御します。

これらのファイルの設定は、EC2Config サービスの操作をコントロールします。

ActivationSettings.xml

- `SetAutodiscover` – KMS を自動検出するかどうかを示します。
- `TargetKMSServer` – KMS のプライベート IP アドレスです。KMS はお使いのインスタンスと同じリージョンに存在する必要があります。
- `DiscoverFromZone` – 指定された DNS ゾーンから KMS サーバーを探します。
- `ReadFromUserData` – UserData から KMS サーバーを取得します。
- `LegacySearchZones` – 指定された DNS ゾーンから KMS サーバーを探します。
- `DoActivate` – このセクションで指定された設定を用いてライセンス認証を試みます。この値は `true` または `false` となります。
- `LogResultToConsole` – 結果をコンソールに出力します。

BundleConfig.xml

- `AutoSysprep` – Sysprep を自動で使用するかどうかを示します。Sysprep を使用には、値を `Yes` に変更します。
- `SetRDPCertificate` – Windows 2003 インスタンスで実行しているリモートデスクトップサーバーに自己署名証明書を設定します。これにより、RDP で安全にインスタンスに接続できます。新しいインスタンスに証明書が必要な場合は、値を `Yes` に変更します。

  この設定は、Windows Server 2008 および Windows Server 2012 のインスタンスでは使用されません。これらのインスタンス自身が証明書を生成できるからです。
- `SetPasswordAfterSysprep` – 新しく起動したインスタンスにランダムなパスワードを設定し、ユーザー起動キーで暗号化し、暗号化されたパスワードをコンソールに出力します。新しいインスタンスに暗号化されたランダムなパスワードを自動的に設定しない場合は、この設定の値を `No` に変更します。

Config.xml

プラグイン

- `Ec2SetPassword` – 暗号化されたランダムなパスワードを、インスタンスを起動するたびに新しく生成します。この機能は、最初の起動以後デフォルトで無効化されますので、同じインスタンスを再起動してもユーザーが設定したパスワードが変更されることはありません。引き続きインスタンスを起動するたびにパスワードを生成するには、この設定を `Enabled` に変更します。

  インスタンスから AMI を作成する予定がある場合、この設定は重要になります。
- `Ec2SetComputerName` – インスタンスのホスト名を、インスタンスの IP アドレスに基づく一意の名前に設定した後、インスタンスを再起動します。ホスト名を自分で指定したい場合や、既存のホスト名を変更したくない場合は、この設定を無効化する必要があります。
- `Ec2InitializeDrives` – 起動時にすべてのインスタンスストアボリュームの初期化とフォーマットを行います。この機能はデフォルトで有効になっており、各インスタンスストアボリュームを初期化した後、ドライブ D:/、ドライブ E:/、と順次マウントしていきます。インスタンスストアボリュームの詳細については、Amazon Elastic Compute Cloud User Guide の Amazon EC2 Instance Store を参照してください。
- `Ec2EventLog` – コンソールにイベントログのエントリを表示します。デフォルトでは、System イベントログから 3 つの最新エラーエントリが表示されます。表示するイベントログのエントリを指定するには、`EC2ConfigService\Settings` ディレクトリにある `EventLogConfig.xml` ファイルを編集します。このファイル内の設定について詳しくは、MSDN ライブラリの Eventlog Key を参照してください。
- `Ec2ConfigureRDP` – ユーザーがリモートデスクトップを使ってインスタンスに安全にアクセスできるよう、自己署名証明書を設定します。この機能は、Windows Server 2008 および Windows Server

2012 のインスタンスでは無効化されています。これらのインスタンス自身が証明書を生成できるからです。

- `Ec2OutputRDPCert` – ユーザーがサムプリントと照合できるよう、リモートデスクトップの証明書情報をコンソールに表示します。
- `Ec2SetDriveLetter` – ユーザーが定義した設定に基づき、ドライブ文字をマウントされたボリュームに割り当てます。デフォルトでは、Amazon EBS ボリュームがインスタンスにアタッチされている場合、ドライブ文字を使ってそのインスタンスにマウントできます。ドライブ文字のマッピングを指定するには、`EC2ConfigService\Settings` ディレクトリにある `DriveLetterConfig.xml` ファイルを編集します。
- `Ec2WindowsActivate` – DNS サフィックス一覧から適切な KMS エントリを探すかどうかを示します。適切な KMS エントリが見つかると、プラグインはリクエストに応答した最初のサーバーをユーザーのアクティベーションサーバーとして設定します。Windows Server 2008 R2 以降の Windows Server は、サフィックス一覧から自動的にエントリを探すことができます。Windows Server 2008 R2 および Windows Server 2012 では、プラグインによってこの検索が手動で実行されます。

  KMS 設定を変更するには、 ActivationSettings.xml`EC2ConfigService\Settings` ファイルを編集します。
- `Ec2DynamicBootVolumeSize`ディスク 0/ボリューム 0 が未使用の領域を含むように拡張します。
- `Ec2HandleUserData` – Sysprep が実行された後初めてインスタンスが起動するときに、ユーザーが作成したスクリプトを作成し実行します。script タグでラップされたコマンドはバッチファイルに保存され、PowerShell タグでラップされたコマンドは .ps1 ファイルに保存されます。

グローバル設定

- `ManageShutdown` – Sysprep の実行中に、instance store-backed AMI から起動したインスタンスが終了しないようにします。
- `SetDnsSuffixList` – ネットワークアダプタの DNS サフィックスを Amazon EC2 に設定します。これにより、完全修飾ドメイン名がなくても、Amazon EC2 で実行中のサーバーの DNS 解決が可能になります。
- `WaitForMetaDataAvailable` – メタデータにアクセスできるようになり、ネットワークが利用可能になるまで、EC2Config サービスが起動処理を続行しないようにします。これにより、EC2Config はアクティベーションのメタデータや他のプラグインから情報を取得できるようになります。
- `ShouldAddRoutes` – 複数の NIC がアタッチされているとき、プライマリネットワークアダプタにカスタムルートを追加して、IP アドレス 169.254.169.250、169.254.169.251、および 169.254.169.254 を有効にします。これらのアドレスは Windows ライセンス認証が使用し、またユーザーがインスタンスのメタデータにアクセスする際にも使用します。
- `RemoveCredentialsfromSyspreponStartup` – 次回のサービスの開始時に `Sysprep.xml` から管理者パスワードを削除します。パスワードを保存しておくためには、この設定を編集します。

DriveLetterConfig.xml

- `DriveLetterMapping` – ドライブ文字のマッピングを設定します。ドライブ文字のマッピングを生成するには、以下の XML を構成します。

```
<?xml version="1.0" standalone="yes"?>
<DriveLetterMapping>
  <Mapping>
    <VolumeName></VolumeName>
    <DriveLetter></DriveLetter>
  </Mapping>
  . . .
  <Mapping>
    <VolumeName></VolumeName>
```

```
    <DriveLetter></DriveLetter>
  </Mapping>
</DriveLetterMapping>
```

- `VolumeName` – ボリュームラベル。例: `My Volume`
- `DriveLetter` – ドライブ文字。例: `X:`

EventLogConfig.xml

- `Category` – 監視するイベントログキー。
- `ErrorType` – イベントの種類 (例えばエラー、警告、情報など) 。
- `NumEntries` – このカテゴリに格納されるイベントの数。
- `LastMessageTime` – 同じメッセージが何度もプッシュされることを防ぐため、サービスがメッセージをプッシュするたびにこの値が更新されます。
- `AppName` – イベントログを記録したイベントソースまたはアプリケーション。

WallpaperSettings.xml

- `Instance ID` – インスタンスの ID を表示します。
- `Public IP Address` – インスタンスのパブリック IP アドレスを表示します。
- `Private IP Address` – インスタンスのプライベート IP アドレスを表示します。
- `Availability Zone` – インスタンスが実行されているアベイラビリティーゾーンを表示します。
- `Instance Size` – インスタンスのタイプを表示します。
- `Architecture` – `PROCESSOR_ARCHITECTURE` 環境変数の設定を表示します。
- `AddMemory` – システムメモリを表示します。単位は GB です。
- `AddECU` – 処理能力を表示します。単位は ECU です。
- `AddIO` – I/O 性能を表示します。

EC2Config の最新バージョンのインストール

EC2Config サービスはデフォルトで各 AWS Windows AMI に含まれています。当社が更新バージョンをリリースする際、すべての AWS Windows AMIs を最新バージョンに更新します。しかし、ユーザーが所有する Windows AMI やインスタンスについては、更新作業はユーザー自身が行う必要があります。

EC2Config の更新通知は、Amazon EC2 forum でご確認ください。

お使いの Windows AMI に含まれる EC2Config のバージョンを確認するには

1. AMI からインスタンスを起動して接続します。
2. コントロールパネルから [Program and Features] を選択します。
3. インストールされたプログラムのリストで `Ec2ConfigService` を探します。バージョン番号は [Version] 列に表示されています。

EC2Config の最新バージョンをインストールするには

1. Amazon Windows EC2Config Service に移動します。
2. [Download] をクリックします。
3. ファイルをダウンロードして解凍します。

4. `EC2Install.exe` を実行します。 セットアッププログラムによってサービスが停止され、アンインストールと新しいバージョンの再インストールが行われます。
5. インスタンスを再起動します。
6. インスタンスに接続し、サービス管理ツールを実行して、`EC2Config service` の状態が `Started` になっていることを確認します。

各バージョンの変更点の詳細については、ダウンロードページの [What's New] セクションを参照してください。

EC2Config の停止、削除、アンインストール

EC2Config は他の通常のサービスと同じように管理できます。

更新した設定をインスタンスに適用するには、サービスをいったん停止してから再起動します。EC2Config を手動でインストールするには、サービスをいったん停止する必要があります。

EC2Config サービスを停止するには

1. Windows インスタンスを起動して接続します。
2. [Start] メニューで [Administrative Tools] にカーソルを重ね、[Services] をクリックします。
3. サービス一覧から EC2Config を右クリックして [Stop] を選択します。

構成設定を更新したり独自 AMI を作成したりする必要がなければ、このサービスは削除できます。サービスを削除するとレジストリのサブキーも削除されます。

EC2Config サービスを削除するには

1. コマンドプロンプトウィンドウを起動します。
2. 次のコマンドを実行します。

```
sc delete ec2config
```

構成設定を更新したり独自 AMI を作成したりする必要がなければ、EC2Config はアンインストールできます。サービスをアンインストールすると、ファイル、レジストリのサブキー、サービスへのショートカット (ある場合) が削除されます。

EC2Config をアンインストールするには

1. Windows インスタンスを起動して接続します。
2. [Start] メニューから [Control Panel] をクリックします。
3. [Programs and Features] をダブルクリックします。
4. プログラム一覧から EC2ConfigService を選択して、[Uninstall] をクリックします。

独自の Windows AMI の作成

Windows インスタンスは、接続後は、他の Windows Server とまったく同じように扱えます。Windows インスタンスを利用するには複数の方法があります。

- インスタンスをそのまま使って、ユーザーが求めるタスクを実行し、タスクが終了したらインスタンスを停止または終了する。

- 実行したいタスクと期間に応じて、ソフトウェア、アプリケーション、および追加ストレージをインスタンスに追加してカスタマイズする。例えば、ある Windows AMI をベースにして、Microsoft Visual Studio Team Foundation Server をインストールし、追加ストレージとして Amazon EBS ボリュームを複数アタッチすることができます (インスタンスは、instance store-backed か Amazon EBS-backed かに関わらず、ソフトウェアおよびアプリケーションのインストール後に再起動できます)。
- カスタマイズしたインスタンスから独自 AMI を作成する。カスタマイズした AMI から複数のインスタンスを起動することもできます。

Windows インスタンスの起動、接続、使用方法の詳細については、Amazon EC2 インスタンスを参照してください。

独自 AMI を作成する前に、ベースとするインスタンスの構成を変更することもできます。新しい AMI から起動されたインスタンスにはすべて、新しい構成が適用されます。Amazon EC2 Windows インスタンスには、EC2Config という構成ツールが付属しています。このツールを使ってインスタンスを構成します。EC2Config サービスの使い方の詳細については、EC2Config サービスを使用した Windows インスタンスの設定 (p. 39) を参照してください。

AMI の作成方法は、ルートストレージデバイスに何を選択するかで異なります。AMI には、Amazon EBS-backed AMI と、Amazon EC2 instance store-backed AMI があります。Amazon EBS-backed AMI と Amazon EC2 instance store-backed AMI では、AMI のサイズ制限やストレージ、データの永続性などに大きな違いがあります。この 2 つの AMI の相違点の詳細については、Windows AMI のルートデバイスストレージ (p. 35) を参照してください。

Amazon EBS-backed Windows AMI の作成方法について詳しくは、Amazon EBS-Backed Windows AMI の作成 (p. 50) を参照してください。instance store-backed Windows AMI の作成方法について詳しくは、Instance Store-Backed Windows AMI の作成; (p. 52) を参照してください。

Amazon EBS-Backed Windows AMI の作成

Amazon EBS-backed Windows AMI を作成する手順は簡単です。インスタンスを起動してカスタマイズしてから、AMI を作成します。

instance store-backed AMI の作成手順はこれとは異なります。詳細については、「Instance Store-Backed Windows AMI の作成; (p. 52)」を参照してください。

Amazon EBS-backed AMI の作成準備をするには

1. Amazon EC2 コンソール (https://console.aws.amazon.com/ec2/) を開きます。
2. ナビゲーションペインの [AMIs] をクリックします。作成する AMI に似ている Amazon EBS-backed AMI を選択します。Amazon EBS-backed Windows AMI を表示するには、[Filter] リストから次のオプションを順番に選択します: [Public Images]、[EBS Images]、[Windows]。

   作成する AMI と同じバージョンの Windows Server を使うパブリック AMI であれば、どれを選択してもかまいません。ただし、必ず Amazon EBS-backed AMI を選択してください。instance store-backed AMI は選択しないでください。
3. [Launch] をクリックして、選択した Amazon EBS-backed AMI のインスタンスを起動します。デフォルト値をそのまま使ってウィザードを完了します。

   AWS Management Console を使って Windows インスタンスを起動する方法について詳しくは、Windows インスタンスを起動する (p. 9) を参照してください。
4. インスタンスを実行したまま接続して、カスタマイズします。例えば、インスタンスに次のような変更を加えることができます。

   a. ソフトウェアやアプリケーションをインストールする。
   b. データをコピーする。

c. 起動時間を短縮するために一時ファイルの消去、ハードディスクのデフラグ、占有領域の開放処理を行う。
d. 新規ユーザーアカウントを作成し、Administrators グループに追加する。
e. EC2Config を使って構成設定を変更する。詳細については、「EC2Config サービスを使用した Windows インスタンスの設定 (p. 39)」を参照してください。

AWS Management Console を使った Windows インスタンスへの接続方法について詳しくは、Windows インスタンスへの接続 (p. 12)を参照してください。

5. インスタンスの準備が終わったら、AMI を作成する前に、データの整合性を確保するため、いったんインスタンスを停止することをお勧めします。EC2Config を使ってインスタンスを停止させていない場合は、以下の手順でインスタンスを停止します。

a. 実行中のインスタンスを右クリックして [Stop Instance] を選択します。
b. 確認ダイアログボックスで [Yes, Stop Instance] をクリックします。

インスタンスのカスタマイズが終了したので、Windows AMI の作成に入ります。AWS Management Console を使って AMI を作成する手順を以下に説明します。代わりにコマンドラインツールを使って AMI を作成する方法の詳細は、ec2-create-image を参照してください。

Amazon EBS-backed AMI を作成するには

1. Amazon EC2 コンソールの [Instances] ページで、インスタンスを右クリックして [Create Image (EBS AMI)] を選択します。

   [Create Image] ダイアログボックスが開きます。
2. イメージにつける一意の名前と説明を入力します。説明の入力は任意です (最長 255 文字) 。
3. Amazon EBS ボリュームを追加するには [EBS Volumes] をクリックします。各ボリュームについて必要な情報を入力したら [Add] をクリックします。

   作成した AMI からインスタンスを起動すると、ここで追加したボリュームは自動的にそのインスタンスにアタッチされます。空のボリュームはフォーマットしてマウントする必要があります。スナップショットベースのボリュームはマウントする必要があります。
4. インスタンスストアボリュームを追加するには [Instance Store Volumes] をクリックします。インスタンスストアボリュームとデバイス名を選択して [Add] をクリックします。

   その後新しい AMI からインスタンスを起動すると、追加されたボリュームは自動的に初期化されてマウントされます。これらのボリュームには、AMI の作成に使用された実行中のインスタンスのインスタンスストアボリュームのデータは含まれません。
5. [Yes, Create] をクリックすると AMI の作成を開始します。
6. [AMIs] ページに移動して、AMI の状態を確認します。現在作成中の AMI は状態が `pending` です。

   AMI の作成プロセスは完了までに数分かかります。プロセスが完了すると、AMI の状態が `available` に変わります。
7. [Snapshots] ページに移動して、新しい AMI 用に作成されたスナップショットを表示します。この AMI から起動するすべてのインスタンスは、ルートデバイスボリュームとしてこのスナップショットを使用します。

これで AMI とスナップショットの作成が終わりました。この 2 つは削除されない限り、ユーザーの AMI アカウントの課金対象です。不要になった AMI とスナップショットは、コンソールから次の手順で削除できます。

AMI およびスナップショットを削除するには

1. [AMIs] ページに移動します。AMI を選択し、[Actions] をクリックして [Deregister] を選択します。確認を求められたら、[Continue] をクリックします。
2. [Snapshots] ページに移動し、スナップショットを右クリックして、[Delete Snapshot] を選択します。確認メッセージが表示されたら、[Yes, Delete] をクリックします。

あるいは ec2-deregister コマンドで AMI を削除し、ec2-delete-snapshot コマンドでスナップショットを削除することもできます。

Instance Store-Backed Windows AMI の作成;

ここでは Instance Store-Backed Windows AMI を作成する手順を説明します。まずインスタンスを起動してカスタマイズし、次にイメージをバンドルし、最後にそのイメージを登録します。

Amazon EBS-backed Windows AMI の作成手順は、これとは異なります。詳細については、「Amazon EBS-Backed Windows AMI の作成 (p. 50)」を参照してください。

Instance Store-Backed Windows AMI の概要

instance Store-Backed AMI から起動されたインスタンスは、インスタンスストアボリュームをルートデバイスボリュームとして使います。instance store-backed AMI のルートデバイスボリュームのイメージは、まず Amazon S3 に格納されます。instance store-backed AMI を使ってインスタンスを起動すると、ルートデバイスボリュームのイメージは Amazon S3 からインスタンスのルートパーティションにコピーされます。以降はこのルートデバイスボリュームを使用してインスタンスを起動します。

instance store-backed AMI を作成したら、Amazon S3 にアップロードする必要があります。Amazon S3 はデータオブジェクトをバケットに保存します。これは、概念的にはディレクトリに似たものです。バケットにはグローバルに一意な名前があり、それぞれ固有の AWS アカウントによって所有されています。

バンドルの手順

バンドルは次の手順で行います。

- イメージを圧縮し、帯域幅とストレージの占有量を最小限にします。
- 圧縮イメージを暗号化して署名し、機密性を確保し作成者本人のものと判るようにします。
- 暗号化したイメージを、アップロードしやすい大きさに分割します。
- Sysprep を実行してコンピュータ固有の情報 (例えば MAC アドレスやコンピュータ名) を削除し、Windows イメージの仮想化に備えます。
- イメージの分割ファイルとそのチェックサムの一覧を含むマニフェストファイルを作成します。
- AMI のコンポーネントをすべて Amazon S3 バケットにアップロードします。このバケットは、バンドルリクエストを行ったときに指定したものです。

ストレージボリューム

instance store-backed AMI を作成するときは、インスタンスのストレージについて、以下の点に特に留意してください。

- 作成した AMI から新たにインスタンスを起動すると、ルートデバイスボリューム (C:) が自動的にアタッチされます。他のインスタンスストアボリュームにあるデータはすべて、インスタンスがバンドルされると削除されます。
- ルートデバイスボリューム以外のインスタンスストアボリューム (例えば D:) は一時的なもので、長期保存の必要がないデータの格納にだけ使用するようにします。

- instance store-backed インスタンスに Amazon EBS ボリュームを追加することもできます。Amazon EBS ボリュームは Amazon S3 バケット内に格納されるため、インスタンスがバンドルされてもそのまま残ります。そのため、永続的に保存する必要のあるデータはすべて、インスタンスストアボリュームではなく、&EBS ボリュームに保存することをお勧めします。

Amazon EC2 ストレージオプションの詳細については、ストレージを参照してください。

Instance Store-Backed Windows AMI の作成前の準備

AMI を作成するには、まずベースとなるインスタンスの準備から始めます。インスタンスに必要なデータやソフトウェアを入れてカスタマイズします。こうして作成した AMI からインスタンスを起動すれば、必要なものはすべて揃っていることになります。

Instance Store-Backed Windows AMI の作成準備をするには;

1. Amazon EC2 コンソール (https://console.aws.amazon.com/ec2/) を開きます。
2. ナビゲーションペインの [AMIs] をクリックします。作成する AMI に似ている instance store-backed AMI を選択します。instance store-backed Windows AMI を表示するには、[Filter] リストから次のオプションを順番に選択します: [Public Images]、[Instance Store Images]、[Windows]。

   作成する AMI と同じバージョンの Windows Server を使うパブリック AMI であれば、どれを選択してもかまいません。ただし、必ず instance store-backed AMI を選択してください。Amazon EBS-backed AMI は選択しないでください。
3. [Launch] をクリックして、選択した instance store-backed AMI のインスタンスを起動します。デフォルト値をそのまま使ってウィザードを完了します。

   AWS Management Console を使って Windows インスタンスを起動する方法について詳しくは、Windows インスタンスを起動する (p. 9) を参照してください。
4. インスタンスを実行したまま接続して、カスタマイズします。例えば、インスタンスに次のような変更を加えることができます。

   a. ソフトウェアやアプリケーションをインストールする。
   b. データをコピーする。
   c. 起動時間を短縮するために一時ファイルの消去、ハードディスクのデフラグ、占有領域の開放処理を行う。
   d. 新規ユーザーアカウントを作成し、Administrators グループに追加する。
   e. EC2Config を使って構成設定を変更する。詳細については、「EC2Config サービスを使用した Windows インスタンスの設定 (p. 39)」を参照してください。

   AWS Management Console を使った Windows インスタンスへの接続方法について詳しくは、Windows インスタンスへの接続 (p. 12)を参照してください。

Instance Store-Backed Windows AMI のバンドル

インスタンスのカスタマイズが終わったので、そのインスタンスをバンドルして AMI を作成できるようになりました。AWS Management Console を使って AMI をバンドルする手順を以下に説明します。代わりにコマンドラインツールを使って AMI をバンドルする方法の詳細については、ec2-bundle-instance を参照してください。

Amazon EC2 instance store-backed AMI をバンドルするには

1. 作成する AMI に使う Amazon S3 バケットについて、既存のバケットを使うか、新しく作成するかを決めます。新しい Amazon S3 バケットを作成する手順は以下のとおりです。

   a. Amazon S3 コンソール (https://console.aws.amazon.com/s3) を開きます。
   b. [Create Bucket] をクリックします。
   c. バケットの名前を指定して [Create] をクリックします。

2. Amazon EC2 コンソール (https://console.aws.amazon.com/ec2/) を開きます。
3. インスタンスを右クリックして [Bundle Instance (instance store AMI)] を選択します。

   [Bundle Instance] ダイアログボックスが開きます。
4. 必要な情報を入力したら、[Bundle] をクリックします。

   a. [Amazon S3 Bucket Name] で自分が所有する S3 バケットのいずれかの名前を指定します。
   b. [Amazon S3 Key Name] でバンドル処理で生成するファイルのプレフィックスを指定します。

   [Bundle Instance] ダイアログボックスに、インスタンスをバンドルするリクエストが成功したことを知らせるメッセージと、バンドルタスクの ID が表示されます。

   Amazon EC2 はインスタンスをシャットダウンしてそれをバンドルし、ユーザーが指定した Amazon S3 バケットに新規イメージを格納します。
5. バンドルタスクの状態を見るには、[Bundle Instance] ダイアログボックスで [View Bundling Tasks] をクリックします。[Close] をクリックしてダイアログボックスを閉じます。

   バンドルタスクの進行につれて、状態は `waiting-for-shutdown`、`bundling`、そして `storing` に移行します。バンドルタスクが正常に完了できない場合、状態は `failed` になります。

Instance Store-Backed Windows AMI の登録

最後に、Amazon EC2 がイメージを見つけてそこからインスタンスを起動できるように、バンドルしたイメージを登録する必要があります。

AWS Management Console を使って AMI を登録する手順を以下に説明します。代わりにコマンドラインツールを使って AMI を登録する方法の詳細については、ec2-register を参照してください。

1. Amazon EC2 コンソール (https://console.aws.amazon.com/ec2/) を開きます。
2. ナビゲーションペインの [AMIs] をクリックします。デフォルトでは、ユーザーが所有する AMI がコンソールに表示されます。
3. 新しくバンドルした AMI を選択し、[Actions] をクリックして [Register New AMI] を選択します。
4. [Register Image] ダイアログボックスの [AMI Manifest Path] に入力して [Register] をクリックします。

これで新しい AMI が Amazon S3 に格納されました。以後 AMI を登録解除して削除するまでの間、ユーザーの AWS アカウントに継続して料金が発生します。

Amazon S3 に格納したソースイメージに変更を加えた場合、いったん登録解除してから再登録しないと、変更が有効になりません。

共有 Windows AMI

共有 Windows AMI とは、開発者が構築し、他の AWS 開発者が使用できるように公開している Windows AMI です。公開されている共有 AMI を使うこともできますし、独自の AMI を作成して共有することもできます。安全かつセキュアで有用なパブリック AMI を作成するのはごく簡単です。

共有用の Windows AMIs の作成

このガイドラインは、使い勝手を向上させ、利用者のインスタンスのセキュリティ脆弱性を抑え、作成者を守るためのものです。

共有 Windows AMI を作成する際は、以下のガイドラインに従ってください。

1. 指示に従って Windows インスタンスを起動し接続します。
2. 共有するソフトウェアとアプリケーションをインストールしてインスタンスをカスタマイズします。安全に共有できる AMI を作成するため、以下を実行します。
   - バンドルの前に必ずシェル履歴を削除してください。シェル履歴には機密情報が含まれることがあります。
   - キーペアなどのインスタンス証明書情報を保存したことがある場合は、削除するか、AMI に含まれない場所に移動してください。
   - 管理者パスワードとその他のアカウントのパスワードが共有に適した値に設定されていることを確認してください。これらのパスワードは、共有 AMI を起動するユーザーはだれでも使用することができます。
   - 共有したくないパスワードはすべて削除します。
   - AMI を公開する前に必ずテストしてください。
3. Sysprep を実行してインスタンスを準備し、新たにインスタンスを起動するたびに新しいパスワードを生成する機能を有効化してください。インスタンスがシャットダウンします。
4. インスタンスのイメージを作成します。

AMI の共有

Amazon EC2 では、ご自分の AMI を他の AWS アカウントと共有できます。このセクションでは、Amazon EC2 コマンドラインツールを使って AMI を共有する方法を説明します。

Note

先に進む前に、必ず 共有用の Windows AMIs の作成 (p. 55) にある AMI 共有のセキュリティガイドラインをお読みください。

AMI には `launchPermission` というプロパティがあり、所有者以外にその AMI からインスタンスを起動できる AWS アカウントを制御します。AMI の `launchPermission` プロパティを変更することで、すべての AWS アカウントにその AMI の起動を許可する (AMI をパブリックにする) ことも、指定したアカウントだけにその AMI の起動を許可することもできます。

`launchPermission` プロパティは、アカウントと起動グループのリストです。リストの項目を追加または削除することで、起動許可を指定できます。特定のアカウントにのみ起動許可を与えるには、AWS アカウント ID を追加または削除します。現在使用可能な起動グループは `all` グループのみで、これを選択すると AMI が公開されます。このセクションのこれ以降の部分では、起動グループのことを単に「グループ」と呼びます。起動グループはセキュリティグループとは違いますので混同しないよう注意してください。AMI にはパブリックの起動許可と明示的な起動許可の両方を指定できます。

Note

ご自分の AMI が他の AWS アカウントから起動されても、その使用料が所有者に請求されることはありません。AMI を実際に起動したアカウントに課金されます。

AMI の公開

AMI を公開にするには

- AMI の launchPermission に all グループを追加します。

```
C:\>  ec2-modify-image-attribute <ami_id> --launch-permission -a all
```

<*ami_id*> パラメータは AMI.の ID です。

この例では、ami-2bb65342 AMI を公開します。

```
C:\>  ec2-modify-image-attribute ami-2bb65342  --launch-permission -a all
launchPermission        ami-2bb65342    ADD     group   all
```

AMI の起動許可を確認するには

- 次のコマンドを入力します。<*ami_id*> は AMI の ID です。

```
C:\> ec2-describe-image-attribute <ami_id> -l
```

この例では、ami-2bb65342 AMI の起動許可を表示します。

```
C:\> ec2-describe-image-attribute ami-2bb65342 -l
launchPermission        ami-2bb65342    group   all
```

AMI を再び非公開にするには

- 起動許可から all グループを削除します。<*ami_id*> は AMI の ID です。

```
C:\> ec2-modify-image-attribute <ami_id> -l -r all
```

これを実行しても、AMI の明示的な起動許可や AMI から実行中のインスタンスには影響しません。

この例は ami-2bb65342 AMI の起動許可から all グループを削除して非公開にします。

```
C:\> ec2-modify-image-attribute ami-2bb65342 -l -r all
launchPermission        ami-2bb65342    REMOVE  group   all
```

特定の AWS アカウントとの AMI の共有

AMI を公開せず、特定の AWS アカウントとだけ共有することもできます。必要なのはアカウント ID のみです。

明示的な起動許可を与えるには

- 次のコマンドを入力します。

```
C:\> ec2-modify-image-attribute <ami_id> -l -a <user_id>
```

<ami_id> は AMI の ID で、 *<user_id>* はアカウント ID です。ハイフンは付けません。

次の例は、ID 111122223333 の AWS アカウントに対し、ami-2bb65342 AMI の起動許可を与えます。

```
C:\> ec2-modify-image-attribute ami-2bb65342 -l -a 111122223333
launchPermission        ami-2bb65342    ADD     userId  111122223333
```

アカウントに与えた起動許可を取り消すには

- 次のコマンドを入力します。

```
C:\> ec2-modify-image-attribute <ami_id> -l -r <user_id>
```

<ami_id> は AMI の ID で、 *<user_id>* はアカウント ID です。ハイフンは付けません。

次の例は、ID 111122223333 の AWS アカウントに与えた ami-2bb65342 AMI の起動許可を取り消します。

```
C:\> ec2-modify-image-attribute ami-2bb65342 -l -r 111122223333
launchPermission        ami-2bb65342    REMOVE  userId  111122223333
```

すべての起動許可を取り消すには

- すべてのパブリックおよび明示的な起動許可を取り消すには、次のコマンドを入力します。

```
C:\> ec2-reset-image-attribute <ami_id> -l
```

<ami_id> は AMI の ID です。

次の例は、ami-2bb65342 AMI へのパブリックおよび明示的な起動許可をすべて取り消します。

```
C:\> ec2-reset-image-attribute ami-2bb65342 -l
launchPermission        ami-2bb65342    RESET
```

Note

AMI の所有者は常にその AMI へのアクセス権を持っており、このコマンドの影響を受けません。

共有 AMI の公開

共有 AMI を作成した後、関連情報を Amazon EC2 リソースセンターで公開することもできます。

AMI を公開するには

1. AMI を Amazon Web Services リソースセンターの Public AMIs フォルダに投稿し、以下の情報を入力します。

   - AMI ID
   - AMI 名（Amazon EBS-backed AMI の場合）または AMI マニフェスト（Amazon EC2 instance store-backed AMI の場合）
   - 公開者
   - 公開者の URL
   - OS / ディストリビューション
   - 主な特徴
   - 説明
   - デーモン / サービス
   - リリースノート

2. 以下の情報をコピーしてドキュメントにペーストすることもできます。HTML 編集モードで実行する必要があります。

```
<strong>&AMI; ID: </strong>[ami-id]<br />
<strong>AMI Manifest: </strong>[myawsbucket/image.manifest.xml]<br />
<h2>About this &AMI;</h2>
<ul>

    <li>Published by [Publisher] (<a href="http://www.mysite.com">[ht
tp://www.mysite.com]</a>).<br />
            </li>
    <li>[Key Features] <br />
            </li>
    <li>[Description]</li>
    <li>This image contains the following daemons / services:
    <ul>

        <li>[Daemon 1]</li>
        <li>[Daemon 2]</li>
    </ul>
            </li>
</ul>
<h2><strong>What&#39;s New?</strong></h2>The following changes were made on
 [Date].<br />
<ul>

    <li>[Release Notes 1]</li>
</ul>
<span style="font-size: x-small; font-family: courier new,couri
er">     - [Note 1]</span><br />
<span style="font-size: x-small; font-family: courier new,couri
er">     - [Note 2]</span><br />
<span style="font-size: x-small; font-family: courier new,couri
er">     - [Note 3]</span><br />
<ul>
```

公開元を明らかにする

現在 AMI はアカウント ID で表されるため、その共有 AMI を提供しているのがどのような相手なのか簡単に知る方法はありません。

そこで、公開した AMI の説明とその AMI ID を Amazon EC2 開発者フォーラムに投稿することをお勧めします。こうすることで、新しい共有 AMI を試してみたいユーザーが情報を入手しやすくなります。また AMI を Amazon マシンイメージ (AMI) ページに投稿することもできます。

共有 Windows AMI の使用

このセクションでは、共有 AMI の見つけ方と安全な使い方を説明します。Amazon EC2 をごく簡単に利用する方法の 1 つは、必要な構成を備えた共有 AMI を見つけ、カスタマイズして使うことです。

共有 AMI の検索

共有 AMI を検索するには

- `ec2-describe-images` コマンド (または短縮形の `ec2dim` コマンド) に、結果をフィルタするフラグを付けて実行します。

  フラグを使って結果をフィルタする方法を以下に例で示します。

  - 次のコマンドは、すべてのパブリック AMI の一覧を表示します。*`-x all`* フラグは、すべての AWS アカウントから実行可能な AMI (つまり、パブリック起動許可が設定された AMI) を示します。これには自分が作成してパブリック起動許可を与えた AMI も含みます。

    ```
    C:\> ec2dim -x all
    ```

  - 次のコマンドは、自分が明示的な起動許可を持っている AMI の一覧を表示します。自分が所有する AMI は除外されます。

    ```
    C:\> ec2dim -x self
    ```

  - 次のコマンドは、 Amazon が所有する AMI の一覧を表示します。.

    ```
    C:\> ec2dim -o amazon
    ```

  - 次のコマンドは、指定した AWS アカウントが所有する AMI の一覧を表示します。

    ```
    C:\> ec2dim -o <target_uid>
    ```

    <target_uid> は求める AMI を所有するアカウント ID です。

  フラグの詳細と、フラグを使って結果をフィルタする方法の詳細は、*Amazon Elastic Compute Cloud Command Line Reference* の ec2-describe-images を参照してください。

共有 AMI の安全な利用

AMI はご自身の責任で起動してください。当社は Amazon EC2 ユーザー間で共有される AMI の完全性や安全性を保証しません。したがって、共有 AMI を取り扱う際は、ご自分のデータセンターに外部のコードをデプロイするとき同様、充分な注意を払ってください。

できれば、信頼できるソース (信頼できるウェブサイトや Amazon EC2 ユーザー) から AMI ID を入手してください。AMI の提供元が不明な場合は、起動する前に AWS フォーラムでその AMI に関するコメントがないかどうか探してみることをお勧めします。また、共有 AMI についての疑問や意見があるときは、AWS フォーラムに投稿してください。

Amazon のパブリックイメージにはエイリアスの所有者が割り当てられており、[userId] フィールドに `amazon` と表示されます。そのため Amazon のパブリックイメージは簡単に見分けられます。

Note

ユーザーは AMI の所有者にエイリアスを使えません。

Windows インスタンスの起動、接続、使用方法の詳細については、インスタンスの使用 を参照してください。

有料 Windows AMI

このセクションでは、有料 AMI の見つけ方と起動方法、サポート製品コードがある場合のインスタンス起動方法を説明します。有料 AMI とは他の開発者から購入できる AMI です。

Amazon EC2 は Amazon DevPay と統合されているので、開発者は、自身が開発した AMI を他の Amazon EC2 ユーザーに有償で提供したり、インスタンスにサポートを提供したりすることができます。Amazon DevPay の詳細については、Amazon DevPay Developer Guide を参照してください。

Note

Amazon DevPay のすべての有料 AMI は Amazon インスタンスストアに格納されます。現時点では、AWS Marketplace は有料 Windows AMI をサポートしていません。

有料 AMI の検索

どのような有料 AMI があるかを知る方法はいくつかあります。Amazon EC2 リソースセンターやフォーラムで情報を探すことができます。または、開発者から直接有料 AMI に関する情報が提供される場合もあります。

また、`ec2-describe-images` コマンドを使ってイメージの説明を表示すれば、その AMI が有料 AMI かどうかが判ります。このコマンドを実行すると、その AMI に関連付けられた製品コードが表示されます (次の例を参照) 。AMI が有料 AMI の場合、製品コードがあります。それ以外の場合はありません。その後 Amazon EC2 リソースセンターやフォーラムで調べれば、その有料 AMI に関する詳細情報や、利用するためのサインアップ窓口が判る場合もあります。

Note

有料 AMI は起動する前にサインアップする必要があります。

AMI が有料かどうかを調べるには

- 次のコマンドを入力します。

```
C:\> ec2-describe-images <ami_id>
```

<ami_id> は AMI ID です。

このコマンドは、指定した AMI に関連する様々なフィールドを返します。出力内に製品コード (例: D6F6052A)) があった場合、その AMI は有料 AMI です。

この例は、有料 AMI の詳細情報を呼び出す ec2-describe-images コールを示します。製品コードは ACD42B6F です。

```
C:\> ec2-describe-images ami-a5bf59cc
IMAGE   ami-a5bf59cc    cloudmin-2.6-paid/image.manifest.xml    541491349868
    available public   ACD42B6F        i386    machine
   instance-store
```

有料 AMI の購入

有料 AMI を起動するには、事前に (購入) サインアップする必要があります。

通常はその有料 AMI の販売者が AMI に関する情報や価格、購入サイトへのリンクを提供しています。リンクをクリックすると、AWS にログインするよう促されます。その後、有料 AMI の価格が表示され、AMI の購入を確認するメッセージが表示されます。

Important

有料 AMI は Amazon EC2 リザーブドインスタンスの割引対象になりません。つまり、リザーブドインスタンスを購入しても、有料 AMI から起動したら、リザーブドインスタンスによる割引料金にはならないということです。有料 AMI の販売者が指定した価格を常に支払う必要があります。リザーブドインスタンスの詳細については、On-Demand and Reserved Instances を参照してください。

有料 AMI の起動

このセクションでは有料 AMI の起動方法と、サポート製品コードがある場合のインスタンス起動方法を説明します。

有料 AMI を購入したら、そこからインスタンスを起動できます。有料 AMI の起動は、他の AMI の起動と同じです。追加パラメータは必要ありません。インスタンスは AMI の所有者が設定した料金に基づいて課金されます。

有料 AMI を起動するには

- 次のコマンドを入力します。

```
C:\> ec2-run-instances <ami_id>
```

<ami_id> は AMI ID です。

この例は ami-2bb65342 AMI を起動するためのコマンドを示します。

```
C:\> ec2-run-instances ami-2bb65342
RESERVATION r-a034c7c9 111122223333 default
INSTANCE i-31a7425a ami-2bb65342 pending 0 m1.small 2010-03-19T13:59:03+0000
 us-east-1a aki-94c527fd ari-96c527ff monitoring-disabled  ebs
```

Note

有料 AMI の所有者は、特定のインスタンスが自分の有料 AMI から起動されたかどうかを確認できます。

有料サポートの利用

有料 AMI には、開発者がソフトウェア (または派生した AMI) のサポートを提供できる機能もあります。開発者は、ユーザーがサインアップして使用できるサポート製品を提供することもできます。このモデルでは、開発者はユーザーに商品を提供します。商品にサインアップすると、開発者はユーザーに商品の製品コードを渡します。ユーザーはこのコードを自分の AMI に関連付ける必要があります。これにより、開発者は、ユーザーのインスタンスがサポート対象であることを確認できます。また、ユーザーが製品からインスタンスを実行すると、その製品について開発者が定めた規定に従って課金されます。

Important

Amazon EC2 リザーブドインスタンスを購入しても、サポート製品コード付き AMI では使えません。つまり、いったん自分の AMI に製品コードを関連付けると、その AMI からの起動にはリザーブドインスタンス割引価格が適用されません。常に、サポート製品の販売者が指定した価格を支払うことになります。リザーブドインスタンスの詳細については、On-Demand and Reserved Instances を参照してください。

自分の AMI に製品コードを関連付けるには

- ec2-modify-image-attribute コマンドを入力します。

```
C:\>  ec2-modify-image-attribute <ami_id> --product-code <product_code>
```

<ami_id> は AMI ID、*<product_code>* は製品コードです。

Important

一度設定した製品コード属性を変更したり削除したりすることはできません。

有料 AMI を起動する際、ec2-run-instances に追加パラメータは不要です。インスタンスは AMI 所有者が設定した料金に従って課金されます。

次のコマンドは、有料 AMI *ami-2bb65342* を起動します。

```
C:\> ec2-run-instances ami-2bb65342
RESERVATION r-a034c7c9 111122223333 default
INSTANCE i-31a7425a ami-2bb65342 pending 0 m1.small 2010-03-19T13:59:03+0000
us-east-1a aki-94c527fd ari-96c527ff monitoring-disabled  ebs
```

有料およびサポート対象 AMI の請求

毎月末に、その月に有料またはサポート対象 AMI を使用したことによるクレジットカードへの請求金額を記載した E メールが届きます。これは通常の Amazon EC2 使用料金とは別に請求されます。

有料 およびサポート対象 AMI の使用方法について詳しくは、Amazon Payments サインインページにアクセスしてください。

Microsoft System Center Operations Manager 向け AWS マネジメントパック

アマゾン ウェブ サービス (AWS) のインフラストラクチャとアプリケーションサービスを利用すると、エンタープライズアプリケーションやビッグデータのプロジェクトから、ソーシャルゲームやモバイルアプリに至るまで、ほぼあらゆるものをクラウドで構築、運用できます。Microsoft System Center Operations Manager 向け AWS マネジメントパック は、AWS で実行中のアプリケーションの可用性とパフォーマンスを監視する機能を備えています。

AWS マネジメントパックは Amazon EC2 インスタンスと、インスタンスで実行している Microsoft Windows または Linux オペレーティングシステムをリンクします。AWS マネジメントパックは Microsoft System Center Operations Manager 向けの拡張パックです。データセンターの指定されたコンピュータ (監視ノードと呼びます) と Amazon ウェブ サービス API を使用して、AWS リソースに関する情報をリモートで検出して収集します。Operations Manager 監視の追加ウィザードを実行して、AWS マネジメントパックが AWS リソースに関する情報を検出できるように設定します。詳細については、「ステップ 1: AWS マネジメントパックのインストール (p. 66)」を参照してください。

AWS マネジメントパックを使用すると、次の AWS リソースを監視できます。

- Amazon Elastic Compute Cloud （ Amazon EC2) インスタンス
- Amazon Elastic Block Store （ Amazon EBS) ボリューム
- Elastic Load Balancing
- AWS Elastic Beanstalk
- AWS CloudFormation スタック
- Auto Scaling グループおよびアベイラビリティーゾーン

AWS マネジメントパックは、Amazon CloudWatch メトリックスとアラームを使って AWS リソースを監視します。Amazon CloudWatch メトリックスは Microsoft System Center にパフォーマンスカウンタとして表示され、Amazon CloudWatch アラームは警告として表示されます。

システム要件

AWS マネジメントパックをダウンロードする前に、お使いのシステムが次の要件を満たすことを確認してください。

- System Center Operations Manager 2007 R2、または、System Center Operations Manager 2012 SP1
- System Center 2012 の場合、アマゾン ウェブ サービス マネジメントパックには Microsoft.Unix.Library MP バージョン 7.3.2026.0 以降が必要です。
- System Center 2007 R2 の場合、AWS マネジメントパックには Microsoft.Unix.Library MP バージョン 6.1.7000.256 以降が必要です。
- System Center Operations Manager 2012 の場合、Cumulative Update 1 以降が必要です。更新ファイルは、少なくとも、アマゾン ウェブ サービスモニタリングに追加されている管理サーバー、監視ノードを実行しているエージェント、Amazon MP の監視対象のエージェントにデプロイする必要があります。アマゾン ウェブ サービスモニタリングに追加されているすべてのコンピュータで、一般に公開されている最新の Operations Manager 更新を実行することをお勧めします。

前提条件

AWS マネジメントパックをダウンロードする前に、お使いのシステムが次の前提条件を満たしていることを確認してください。

- 監視ノードとして指定したデータセンターのエージェント管理のコンピュータで、エージェントプロキシのオプション、[このエージェントがプロキシとして動作し、他のコンピュータ上の管理オブジェクトを検出できるようにする] が有効になっている。
- 監視ノードのアクションアカウントが、監視ノードに対してローカル管理者権限を持っている。
- 監視ノードがインターネットに接続できる（AWS API コールをするため）。
- Microsoft .NET Framework バージョン 3.5.1 以降が監視ノードにインストールされている。
- AWS アカウントで Amazon CloudWatch サービスが有効になっている。
- 管理する Amazon EC2 インスタンスが Microsoft System Center（Amazon EC2 インスタンスと、そこで実行している Windows または Linux オペレーティングシステムをリンクするためのOperations Manager エージェント）を実行している。この機能を使用する場合、エージェントがデプロイされ、実行中で、データセンターの管理サーバーと通信できることを確認する必要があります。

AWS マネジメントパックのダウンロード

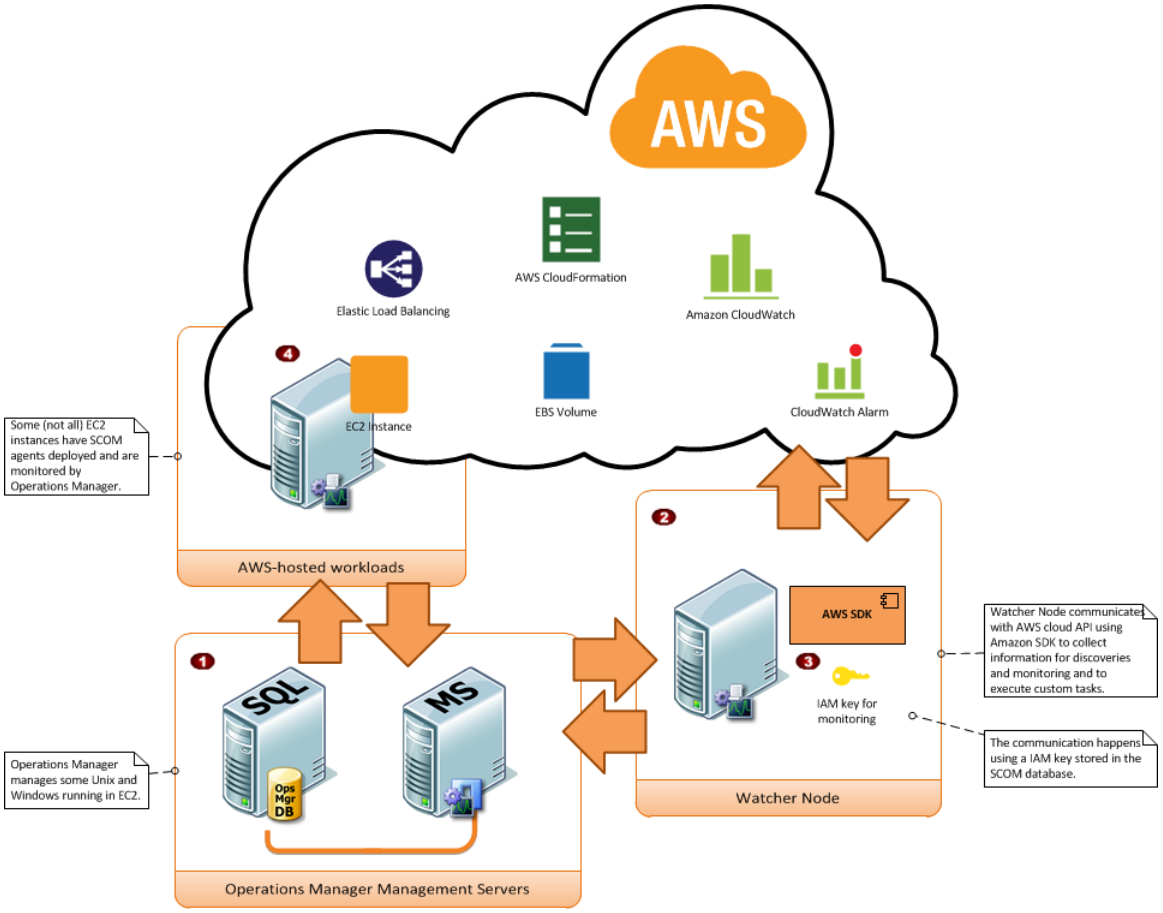
AWS リソースを監視するには、AWS マネジメントパックをダウンロードする必要があります。AWS マネジメントパックは無料です。監視する AWS リソース（例: Amazon EC2 インスタンス、Elastic Load Balancing、Amazon CloudWatch メトリックスとアラーム）についてだけ料金をお支払いください。

AWS マネジメントパックをダウンロードするには

1. AWS Management Pack for Microsoft System Center ウェブサイトで、[Download AWS MP for SCOM 2007 R2] または [Download AWS MP for SCOM 2012] をクリックします。
2. プロンプトが表示されたら、Amazon.AmazonWebServices.mpb または AWS MP Setup.msi ファイルをコンピュータに保存します。

AWS マネジメントパックをデプロイするには

次の手順に従って AWS マネジメントパックをインポートしてデプロイする前に、AWS リソースの監視に関連するさまざまなコンポーネントを確認してください。また、監視ノードとして使用するコンピュータを決定する必要があります。AWS リソースの監視に必要な AWS 認証情報についても決定する必要があります。主なコンポーネントを次の図に示します。



項目	コンポーネント	説明
1	Operations Manager インフラストラクチャ	Microsoft SQL Server と Microsoft Active Directory ドメインなどの、1 つ以上の管理サーバーとその依存関係。これらのサーバーは、オンプレミスまたは AWS クラウドのいずれかにデプロイでき、どちらのケースもサポートされています。
2	監視ノード	AWS SDK for .NET を使用して AWS と通信するために使用する、エージェント管理のコンピュータ。オンプレミスまたは AWS クラウドのいずれかにデプロイできますが、エージェント管理のコンピュータであり、インターネット接続が可能である必要があります。1 つの AWS アカウントを監視できる監視ノードは 1 台だけです。ただし、複数の AWS アカウントの監視に同じ監視ノードを使用することができます。

項目	コンポーネント	説明
3	AWS 認証情報	監視ノードが AWS API コールを行うために使用するアクセスキー ID とシークレットアクセスキー。AWS マネジメントパックの設定時にこれらの認証情報を指定する必要があります。リードオンリー権限の IAM ユーザーを作成してその認証情報を使用することをお勧めします。IAM ユーザーの作成方法について詳しくは、*Using IAM* の Adding a New User to Your AWS Account を参照してください。
4	Amazon EC2 インスタンス	AWS クラウドで実行する仮想コンピュータAmazon EC2 インスタンスには、Operations Manager エージェントがインストールされている場合とインストールされていない場合があります。Operations Manager エージェントをインストールすると、インスタンスの状態とは別にオペレーティングシステムやアプリケーションの状態を見ることができるので、状態をより詳細に把握できます。

Topics

ステップ 1: AWS マネジメントパックのインストール

AWS マネジメントパックをダウンロードしたら、それをインポートして、1 つまたは複数の AWS アカウントを監視するように設定する必要があります。

System Center 2012 向け AWS マネジメントパックをインストールするには

1. Microsoft System Center Operations Manager Operations Console の [Go] メニューで [Administration] をクリックします。
2. [Management Packs] を右クリックし、次に [Import Management Packs] をクリックします。
3. Import Management Packs Wizard で [Add] をクリックし、次に [Add from disk] をクリックします。
4. [Select Management Packs to import] ダイアログボックスで [Amazon.AmazonWebServices.mpb] をクリックして、ダウンロード先のディレクトリを指定し、[Open] をクリックします。
5. [Select Management Packs] ページに、インポートに選択した AWS マネジメントパックが表示されます。[Import] をクリックします。

   Note

   [import] をクリックしても、[Import] でErrorアイコンが表示されている管理パックはインポートされません。

6. [Import Management Packs] ページが開いて、管理パックの進行状況を表示します。インポート処理中に問題が起きた場合は、リストから管理パックを選択すると詳細なステータスが表示されます。[Close] をクリックします。

System Center 2007 R2 向け AWS マネジメントパックをインストールするには

System Center 2007 の場合、管理パックは Microsoft System Installer ファイル、AWS_MP_Setup.msi として配布されます。これには、監視ノードと System Center Operations Manager のルート管理サーバーに必要な各種 DLL、オペレーションコンソール、および Amazon.AmazonWebServices.mp ファイルが含まれます。

Note

ルート管理サーバー、オペレーションコンソール、AWS 監視ノードが別々のコンピュータにある場合、それぞれのコンピュータでインストーラを実行する必要があります。

1. AWS_MP_Setup.msi ファイルを実行します。
2. [Welcome to the Amazon Web Services Management Pack Setup Wizard] 画面で [Next] をクリックします。
3. [End-User License Agreement] 画面で、使用許諾書を読み、[I accept the terms in the License Agreement] チェックボックスをクリックしてから [Next] をクリックします。
4. [Custom Setup] 画面で、インストールする機能を選択し、[Next] をクリックします。

   - [Operations Console Component] をクリックすると、Amazon.AmazonWebServices.UI.Pages.dll ライブラリをインストールして、グローバルアセンブリキャッシュ (GAC) に登録し、AWS マネジメントパックファイル、Amazon.AmazonWebServices.mp をインストールします。
   - [Root Management Server] をクリックすると、Amazon.AmazonWebServices.Modules.dll ライブラリをインストールして、GAC に登録します。
   - [AWS Watcher Node] をクリックすると、Amazon.AmazonWebServices.Modules.dll ライブラリをインストールして、GAC に登録し、AWS SDK for .NET (AWSSDK.dll) を GAC にインストールします。

5. [Ready to install Amazon Web Services Management Pack] 画面で [Install] をクリックします。
6. [Completed the Amazon Web Services Management Pack Setup Wizard] 画面で、[Finish] をクリックします。

Note

必要な DLL がコピーされて GAC に登録されます。管理パックファイル (*.mp) は、オペレーションコンソールを実行しているコンピュータの Program Files (x86)/Amazon Web Services Management Pack フォルダにコピーされます。他の管理パックと同様に、管理パックを SCOM 2007 R2 SP1 に手動でインポートする必要があります。

7. Operations Consoleの [Go] メニューで[Administration] をクリックします。
8. [Administration] ナビゲーションペインで [Administration] を右クリックして、[Import Management Packs] をクリックします。
9. [Import Management Packs] ウィザードで、[Add] をクリックして、[Add from disk] をクリックします。
10. [Select Management Packs to import] ダイアログボックスで、ディレクトリを、管理パックファイルが保存されている `C:\Program Files (x86)\Amazon Web Services Management Pack` に変更し、[Amazon.AmazonWebServices.mp] をクリックして、次に [Open] をクリックします。
11. [Select Management Packs] ページの [Import list] で [Amazon Web Services] を選択し 、[Install] をクリックします。

Note

[Install] をクリックしても、[Import list] でErrorアイコンが表示されている管理パックはインストールされません。

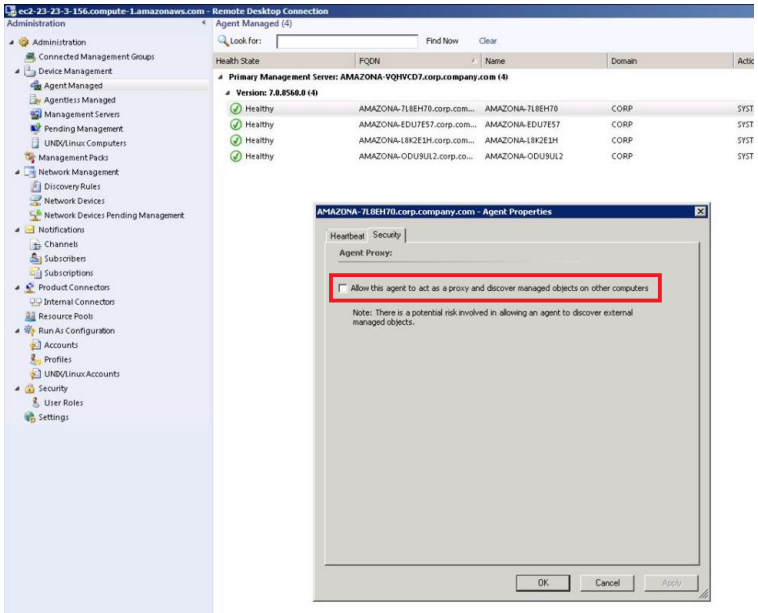
12. [Import Management Packs] ページが開いて、管理パックの進行状況を表示します。インポート処理中に問題が起きた場合は、リストから管理パックを選択すると詳細なステータスが表示されます。[Close] をクリックします。

ステップ 2: 監視ノードの設定

監視ノードは、監視ノードコンピュータ外で検出を実行するので、監視ノードでプロキシエージェントオプションを有効にしておく必要があります。プロキシエージェントにより、検出機能が他のコンピュータのオブジェクトを操作できるようになります。

プロキシエージェントを有効にするには

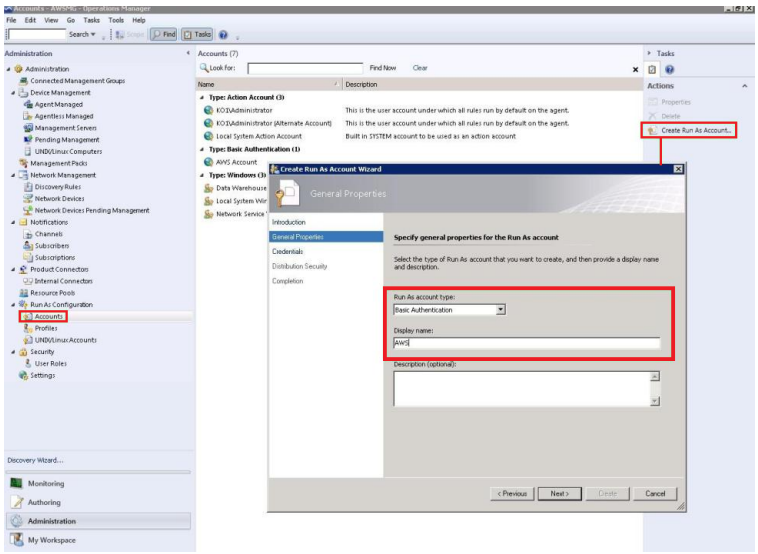
1. Microsoft System Center Operations Manager Operations Console の [Go] メニューで [Administration] をクリックします。
2. [Administration] ワークスペースの [Device Management] で [Agent Managed] をクリックします。
3. [エージェントで管理] 項目のリストで監視ノードを右クリックし、[Properties] をクリックします。
4. [Agent Properties] ダイアログボックスの [Security] タブで、[Allow this agent to act as proxy and discover managed objects on other computers] チェックボックスを選択してから [OK] をクリックします。



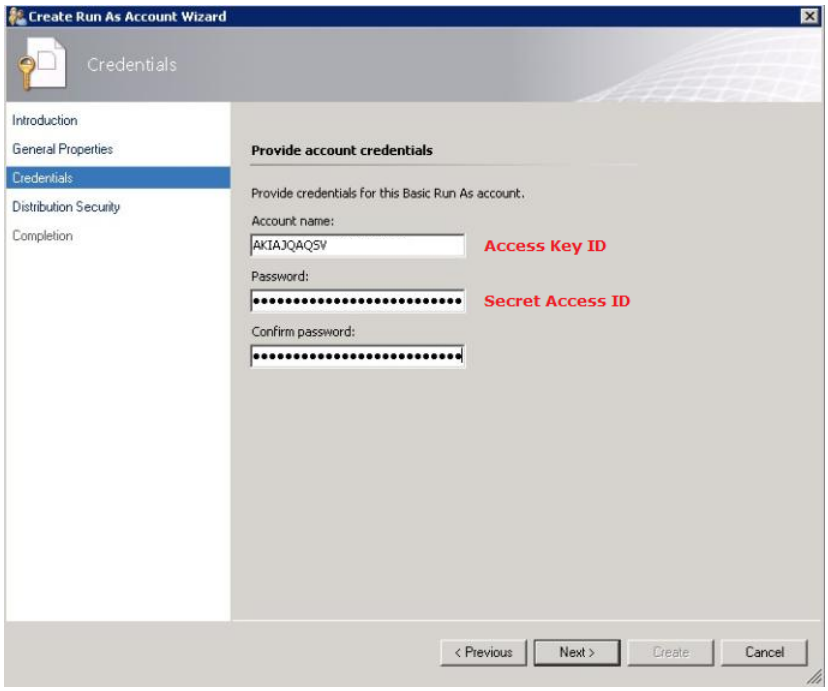
ステップ 3: AWS 実行アカウントを作成する

AWS 実行アカウントを作成するには

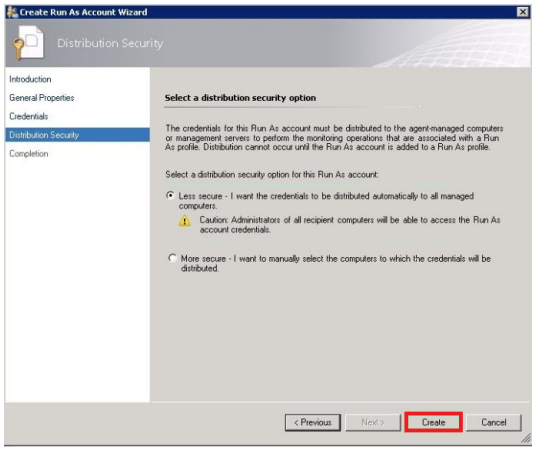
1. Microsoft System Center Operations Manager Operations Console の [Go] メニューで [Administration] をクリックします。
2. [Administration] ワークスペースで [Run As Configuration] ノードを展開し、[Accounts] を選択します。
3. [Accounts] ペインを右クリックし、[Create Run As Account] をクリックします。
4. [Create Run As Account Wizard] の [General Properties] ページで、[Run As account type] ドロップダウンリストから [Basic Authentication] を選択します。.



5. [Display name] ボックスに、表 名 (例: 「John I M Acc unt) を入力し、[Description] ボックスに説明を入力します。



6. [Next] をクリックし、 Credentials] ページの [Account name] ボックスにアクセスキー ID を入力し、[Password] ボックスにシークレットアクセスキーを入力します。
7. [Next] をクリックし、[Distribution Security] ページで [More secure - I want to manually select the computers to which the credentials will be distributed] を選択します。



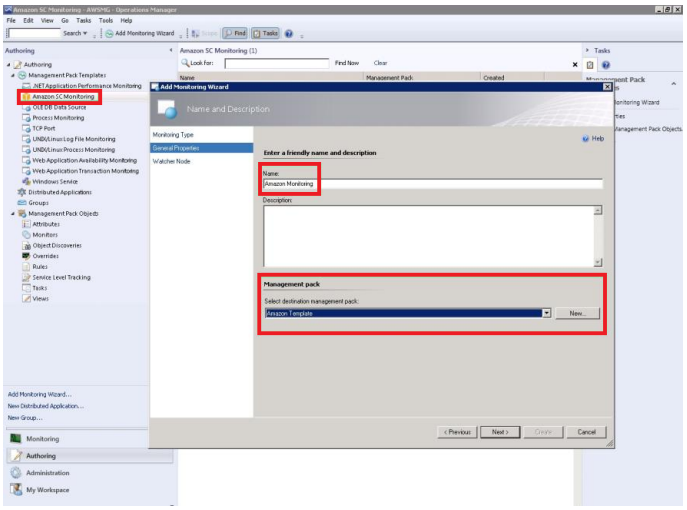
8. [Create] をクリックし、次に [Close] をクリックすると実行アカウントの作成は完了です。

ステップ 4: 監視の追加ウィザードを実行する

AWS マネジメントパックに特定の AWS アカウントを監視するように設定するには、監視の追加ウィザードを使用します。これは、オペレーションコンソールの [作成] ワークスペースから利用できます。これにより、監視する AWS アカウントの設定を含む新しいマネジメントパックが作成されます。AWS アカウントを監視対象に追加するたびにこのウィザードを実行する必要があります。つまり、2 つの AWS アカウントを監視する場合は、ウィザードを 2 回実行する必要があります。

監視の追加ウィザードを実行するには

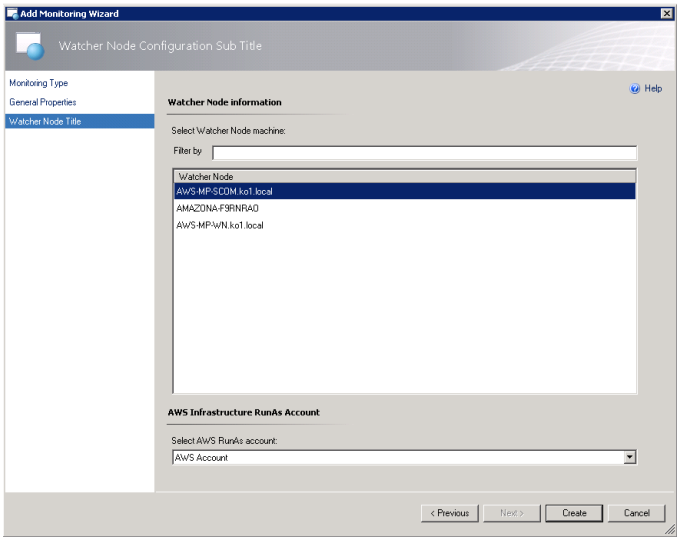
1. Microsoft System Center Operations Manager Operations Console の [Go] メニューで [Authoring] をクリックします。
2. [Authoring] ワークスペースで [Management Pack Templates] ノードを展開して [Amazon Web Services] を右クリックし、[Add Monitoring Wizard] をクリックします。
3. [Add Monitoring Wizard] の [Select the monitoring type list] リストで [Amazon Web Services] を選択し、[Next] をクリックします。
4. [General Properties] ページの [Name] ボックスに名前 (例:「John AWS Resources」) を入力し、[Description] ボックスに説明を入力します。
5. [Select destination management pack] ドロップダウンリストから設定を保存したい場所の既存の管理パックを選択 (または、[New] をクリックして新規に作成) し、[Next] をクリックします。



Note

デフォルトでは、管理パックオブジェクトの作成、ルールまたは監視を無効化、またはオーバーライドの作成を行うと、Operations Manager は設定をデフォルトの管理パックに保存します。ベストプラクティスとして、カスタマイズされた設定をデフォルトの管理パックに保存するのではなく、カスタマイズするシールドされた管理パックごとに管理パックを作成することをお勧めします。

6. [Watcher Node Configuration] ページの [Watcher Node] リストで、エージェントに管理されたコンピュータを監視ノードとして選択します。



7. [Select AWS Run As account] ドロップダウンリストで、前の手順で作成した実行アカウントを選択し、[Create] をクリックします。
8. AWS マネジメントパックは設定が完了すると、まず監視ノードを検出します。監視ノードが正常に検出されたことを確認するには、オペレーションコンソールの [Monitoring] ワークスペースに移動します。新しく Amazon Web Services フォルダが表示され、その下に Amazon Watcher Nodes サブフォルダが表示されているはずです。このサブフォルダは監視ノードを表示します。AWS マネジメントパックは、アマゾン ウェブ サービスへの監視ノードの接続を自動的に確認して監視します。監視ノードが検出されると、このリストに表示されます。監視ノードの準備ができたら、状態が [健全] に変わります。

   Note

   アマゾン ウェブ サービスとの接続を確立するために、AWS マネジメントパックは、AWS SDK for .NET、モジュール、スクリプトを監視ノードにデプロイする必要があります。これには 10 分ほどかかります。監視ノードが表示されない、あるいは状態が [Not Monitored] と表示される場合、インターネット接続と IAM の権限をよく確認してください。詳細については、「AWS マネジメントパックのトラブルシューティング (p. 86)」を参照してください。

9. 監視ノードが検出されると、依存関係の検出が実行され、オペレーションコンソールの [監視] ワークスペースに AWS リソースが表示されます。

   Note

   AWS リソースの検出は 20 分以内に完了するはずですが、Operations Manager 環境、AWS 環境、管理サーバーの負荷、監視ノードの負荷によっては、さらに時間がかかる場合もあります。詳細については、「AWS マネジメントパックのトラブルシューティング (p. 86)」を参照してください。

AWS マネジメントパックの使用

このセクションでは、AWS マネジメントパックのビューとタスクを使用して AWS リソースとメトリックスの状態を監視する方法と、コンテキストを意識したタスクの実行方法について説明します。

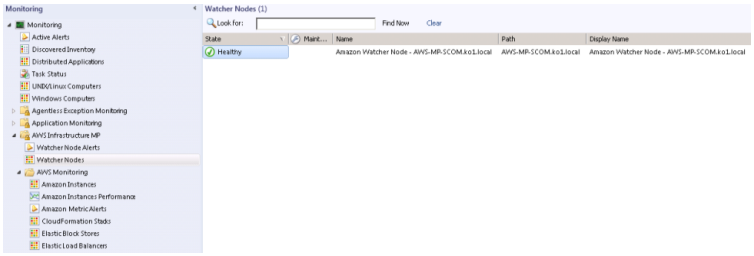
Topics

ビュー

AWS マネジメントパックには次のビューが用意されていて、オペレーションコンソールの [監視] ワークスペースに表示されます。

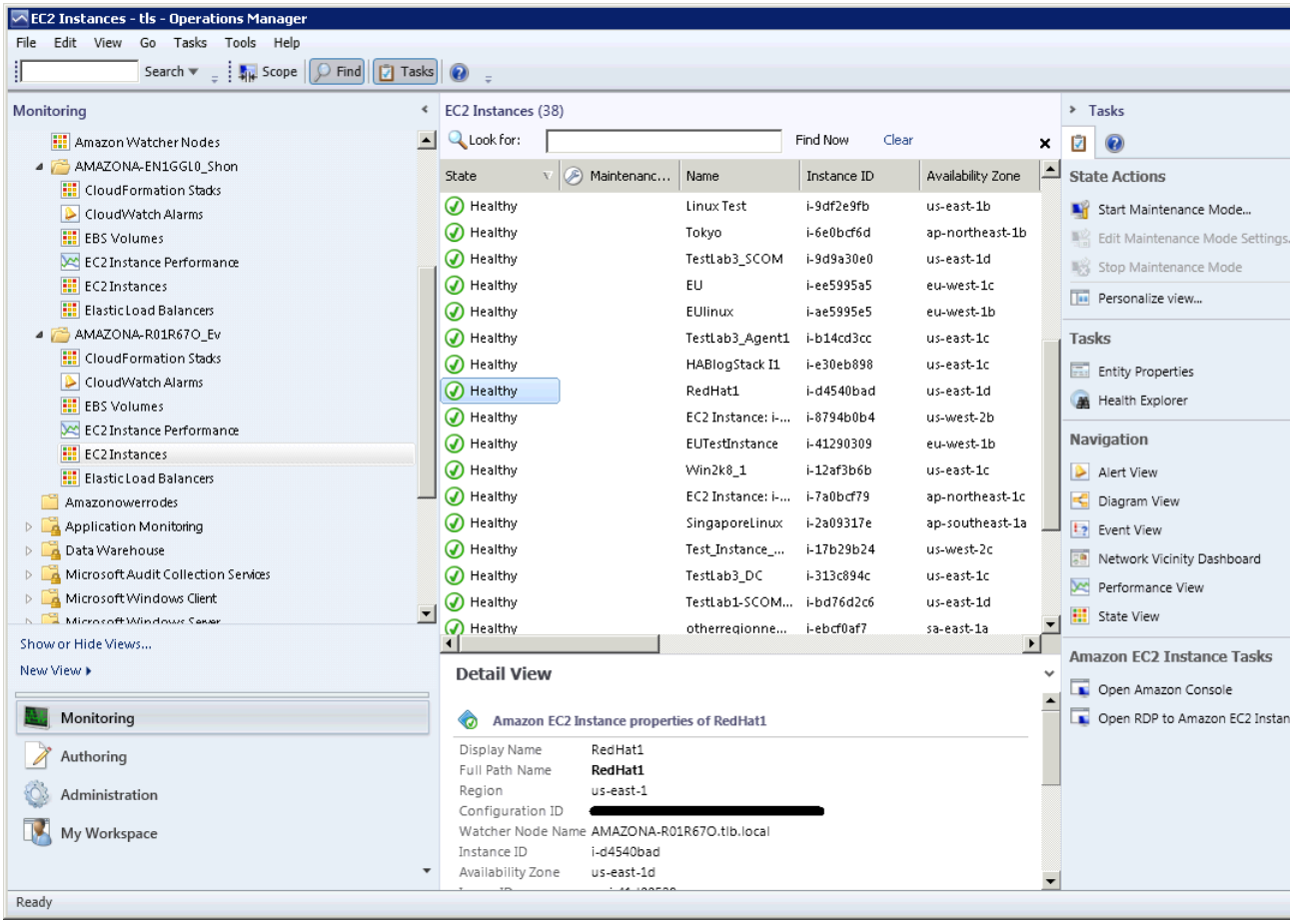
- Watcher Nodes State View

  監視対象の全 AWS アカウントの監視ノードの状態を表示します。「健全」な状態とは、監視ノードが正しく設定され、AWS と通信できる状態を指します。



- EC2 Instances State View

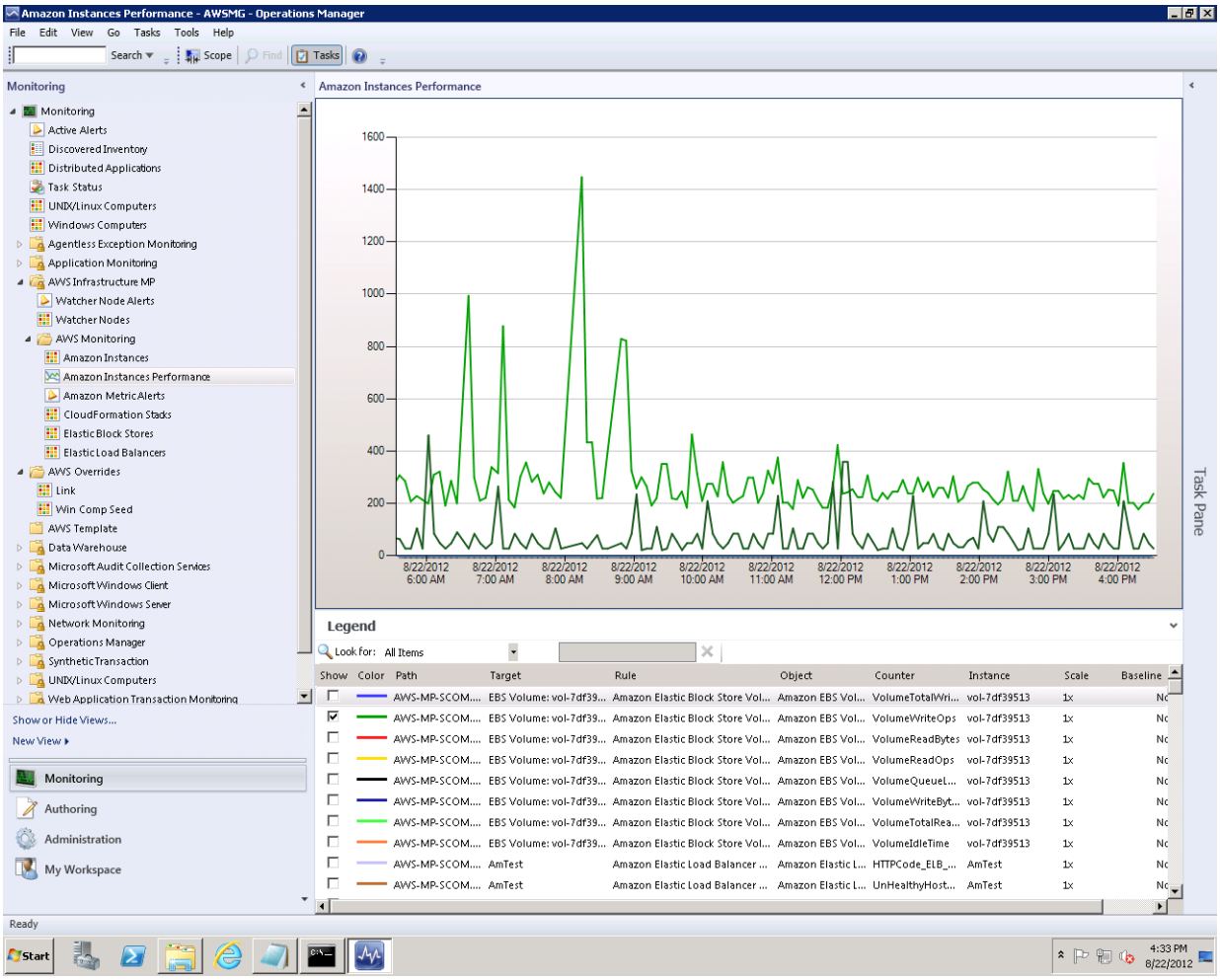
  特定の AWS アカウントの、すべての Amazon EC2 インスタンス (全アベイラビリティーゾーンおよびリージョン) の状態を表示します。Amazon Virtual Private Cloud (VPC) で実行中の Amazon EC2 インスタンスもビューに表示されます。AWS マネジメントパックは Amazon EC2 タグを取得するので、これらのタグを使用してリストを検索したりフィルタリングしたりできます。[Windows Computer] 列と [UNIX/Linux Computer] 列により、Operations Manager エージェントが Amazon EC2 インスタンス内で実行しているかどうかがわかります。



- AWS Performance View

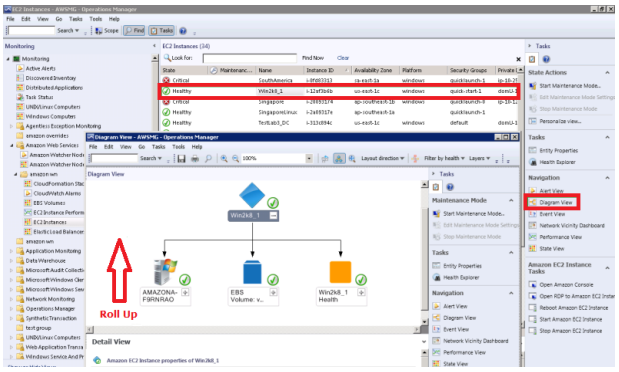
  Amazon EC2、Amazon EBS、Elastic Load Balancing のデフォルト Amazon CloudWatch メトリックスを表示します。これらのメトリックスの詳細については、*Amazon CloudWatch Developer Guide* の CloudWatch Metrics, Namespaces, and Dimensions Reference を参照してください。

  次の図に例を示します。



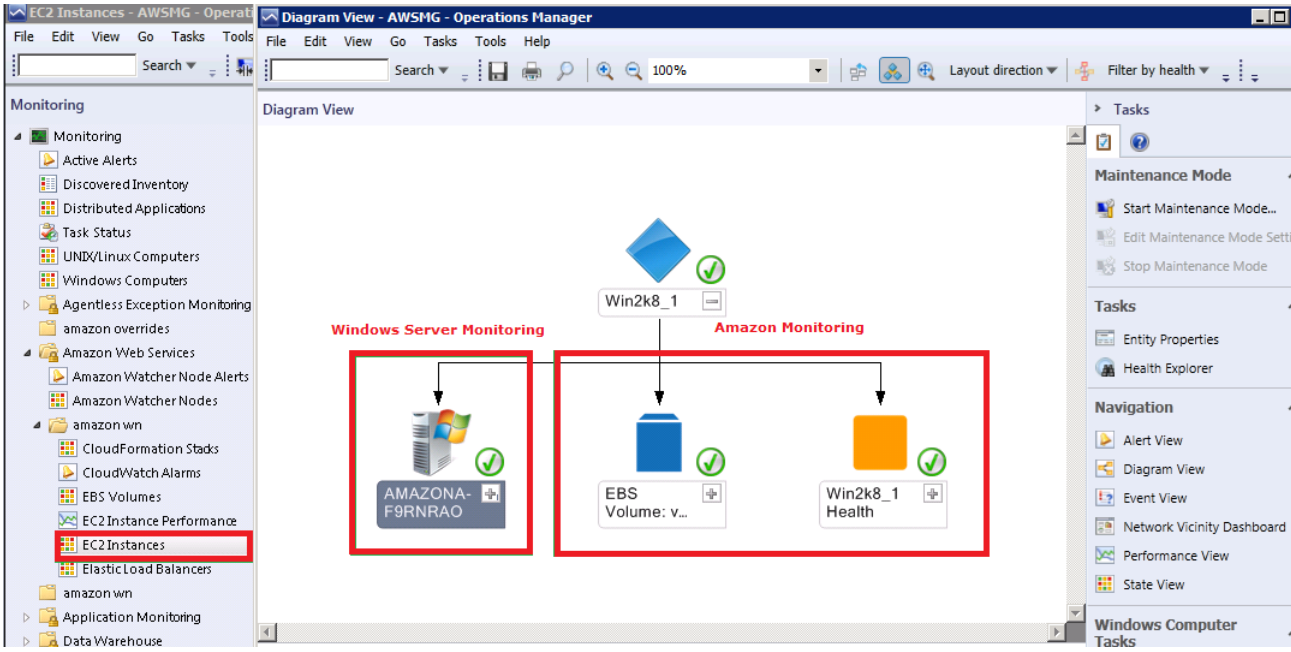
- Instance Diagram View

  Amazon EC2 インスタンスと他のコンポーネントの関係を表示します。

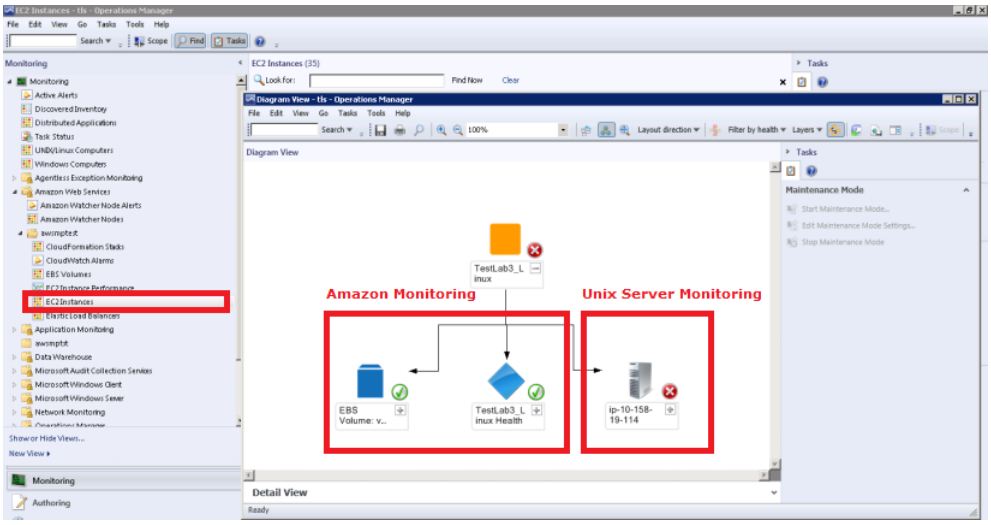


  Amazon EC2 インスタンスとそのオペレーティングシステムの関係が確立すると、ダイアグラムビューには自動的にオペレーティングシステムとその配下のコンポーネントが表示されます。

  以下の図は、Windows を実行する Amazon EC2 インスタンスの例を示します。

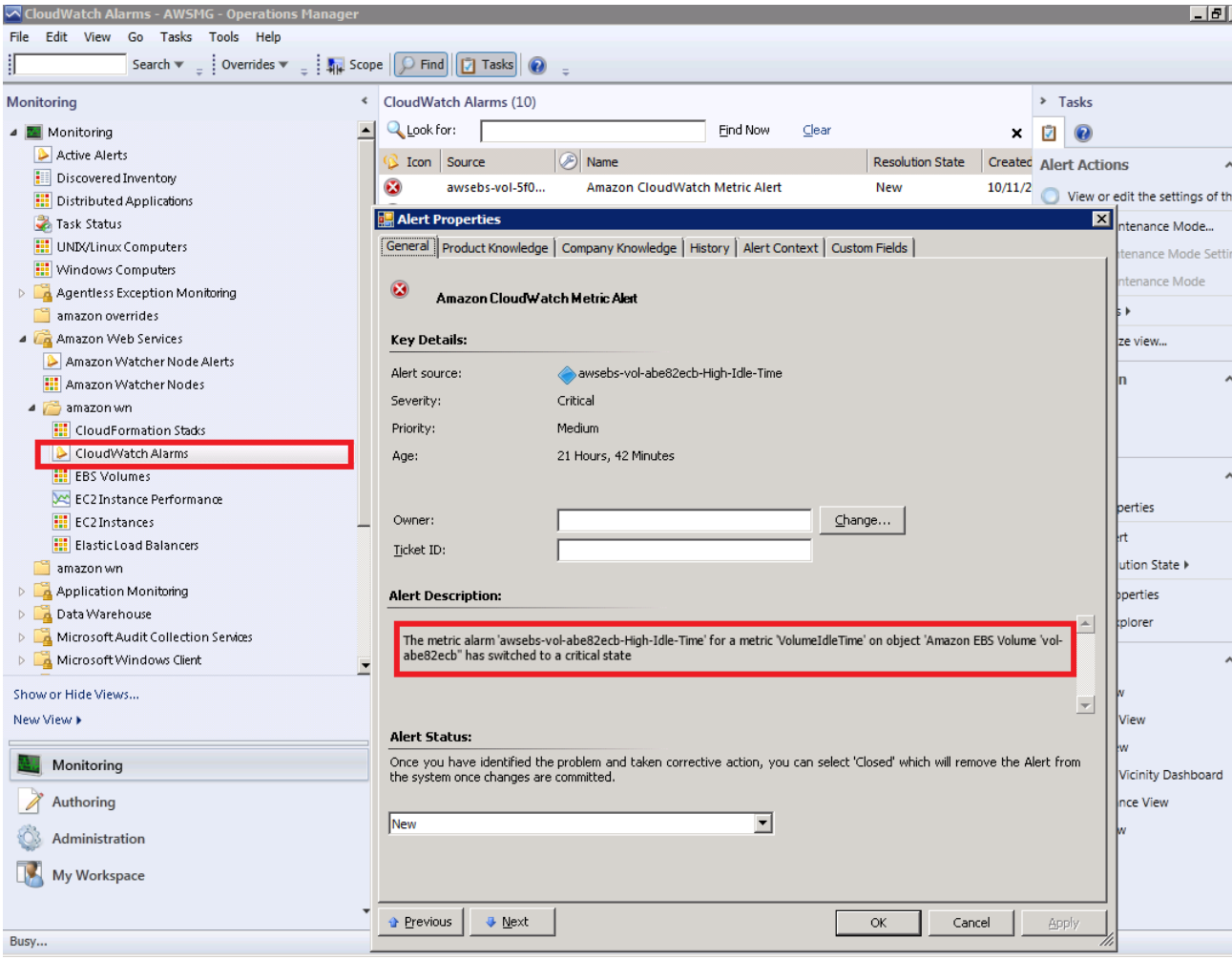


以下の図は、UNIX を実行する Amazon EC2 インスタンスの例を示します。



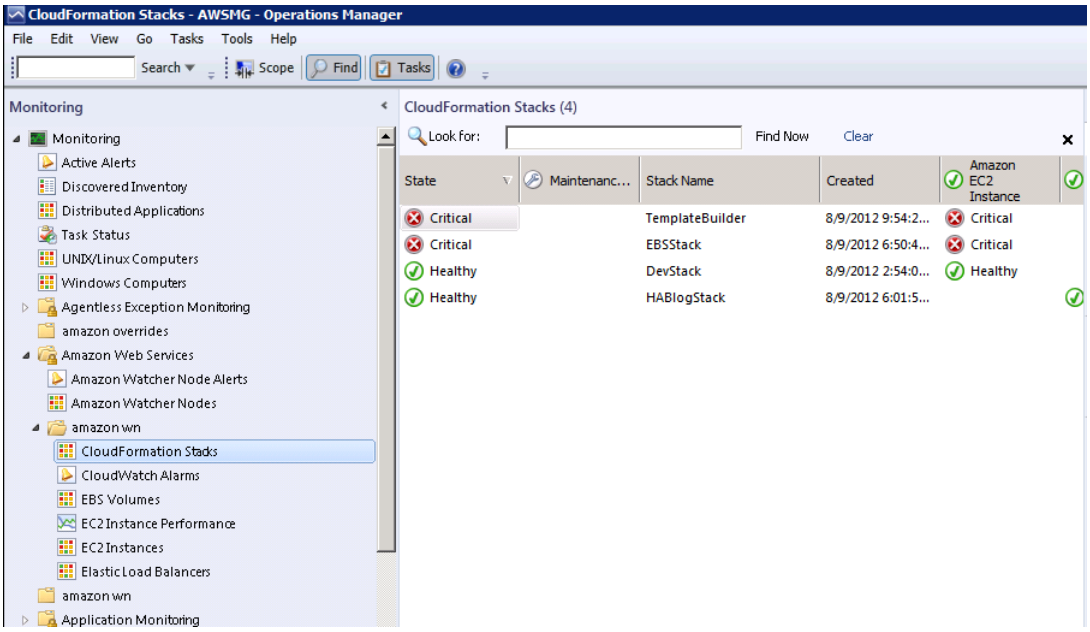
- AWS Alerts View

  見つかった AWS リソースに関連する Amazon CloudWatch アラームを表示します。



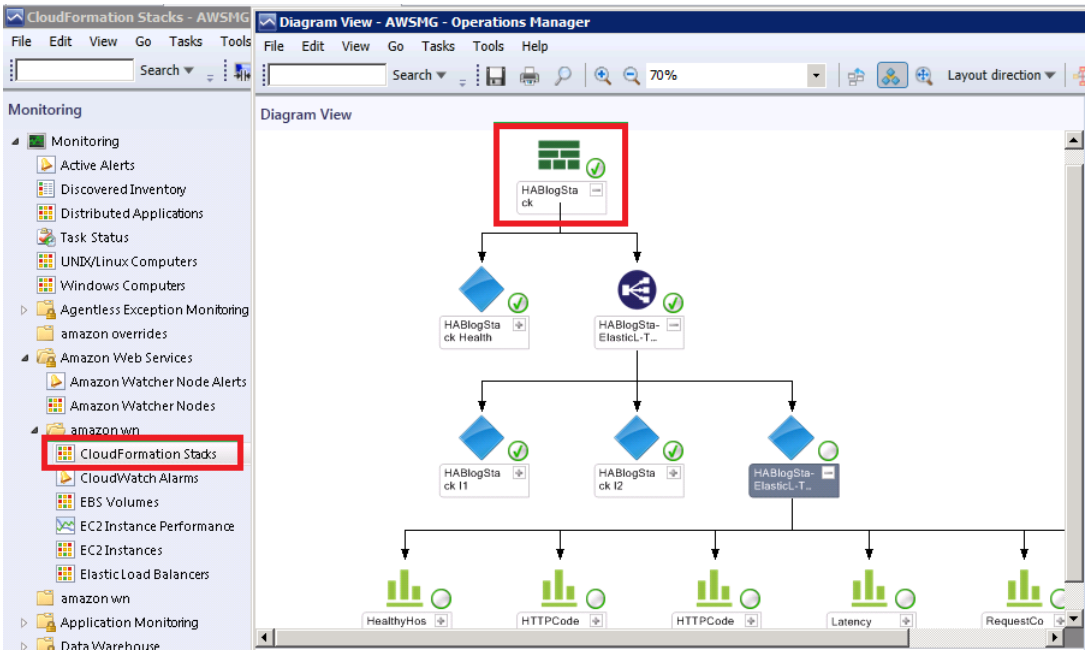
- CloudFormation Stacks State View

  特定の AWS アカウントの、すべての AWS CloudFormation スタック（全リージョン）のヘルス状態を表示します。



- CloudFormation Stack Diagram View

  AWS CloudFormation スタックと他のコンポーネントの関係を表示します。AWS CloudFormation スタックには Amazon EC2 または Elastic Load Balancing のリソースが含まれる場合があります。次の図に例を示します。

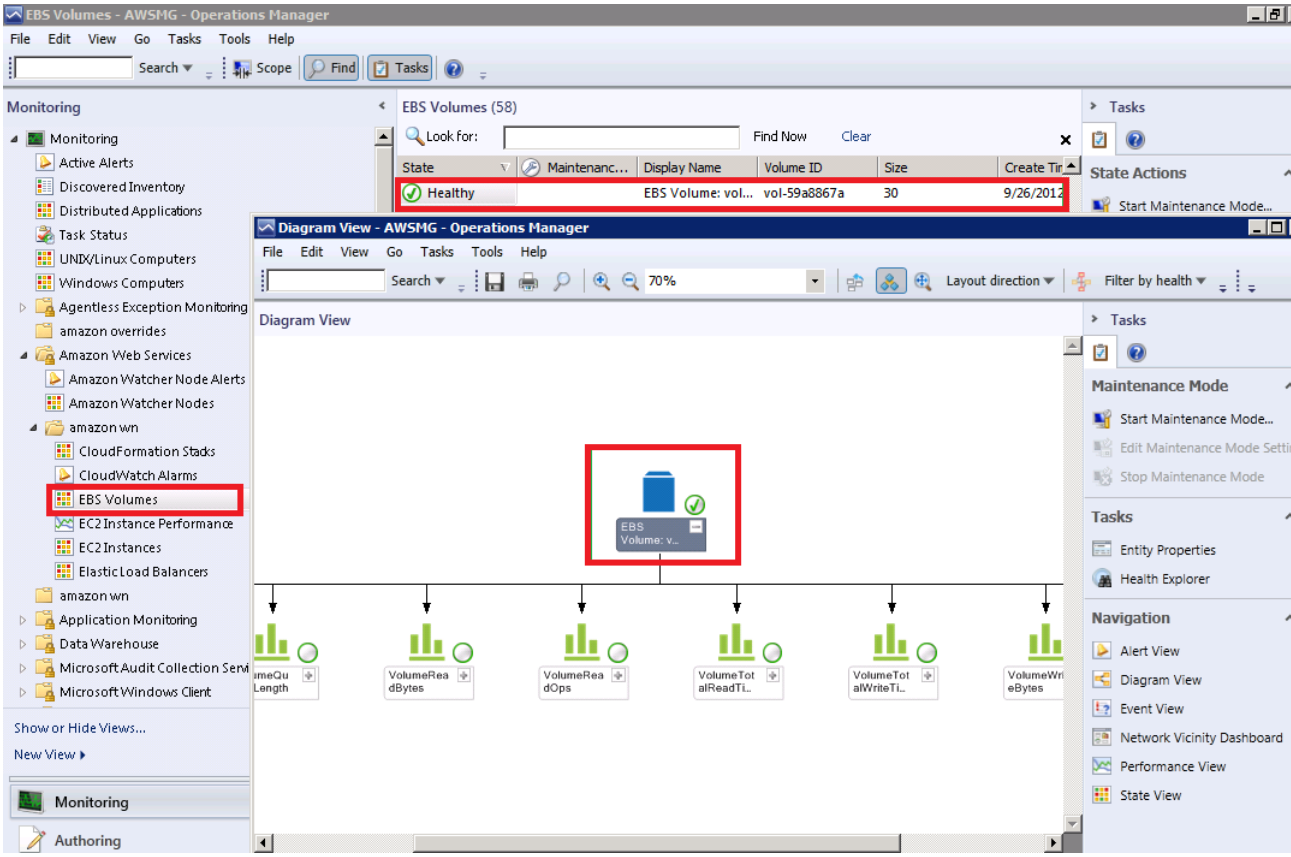


- EBS Volumes State View

  特定の AWS アカウントの、すべての Amazon EBS ボリューム (全アベイラビリティーゾーンおよびリージョン) のヘルス状態を表示します。

- EBS Volume Diagram View

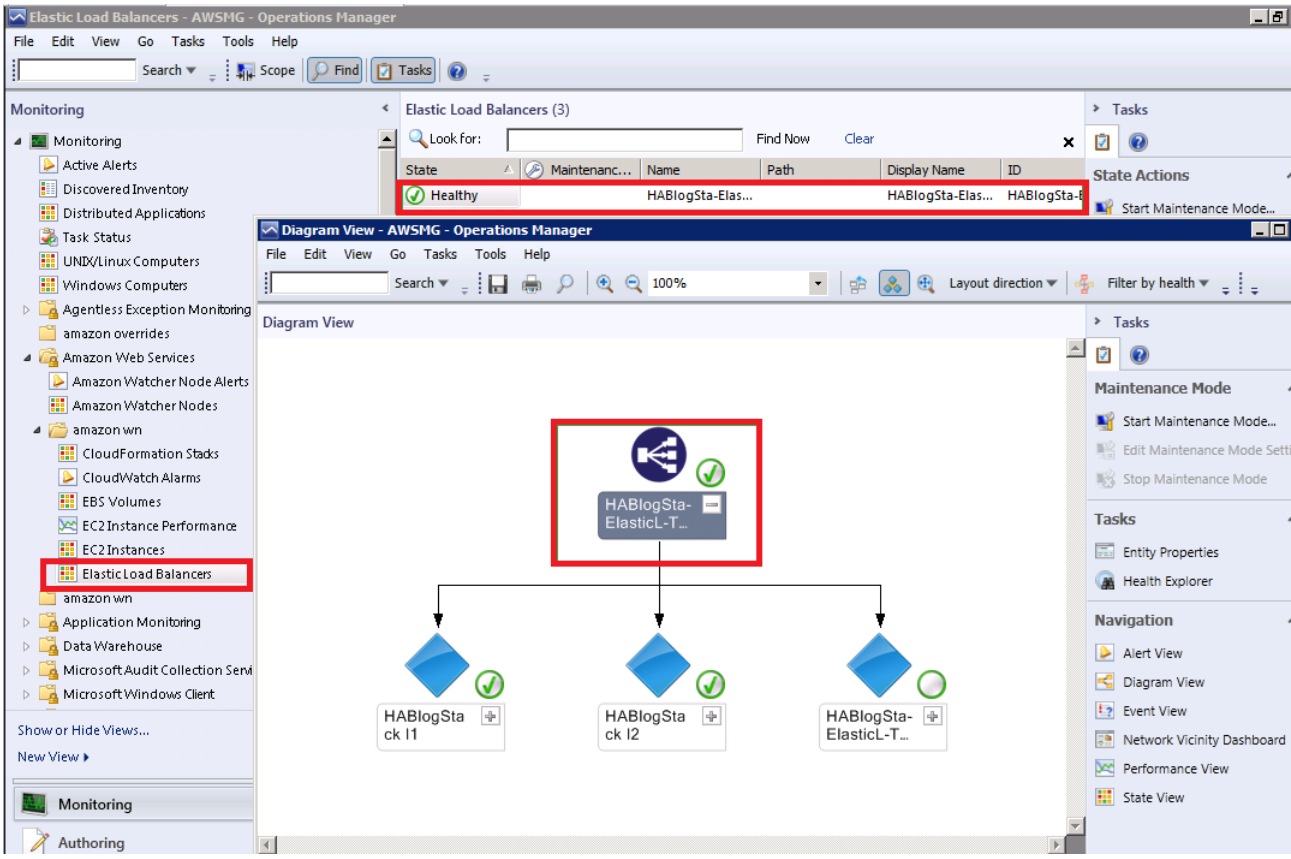
Amazon EBS ボリュームとデフォルトの Amazon CloudWatch メトリックスを表示します。Amazon CloudWatch メトリックスが 「Not Monitored」と表示される場合、Amazon CloudWatch メトリックに少なくとも 1 つの Amazon CloudWatch アラームが定義されていることを確認してください。次の図に例を示します。



- Elastic Load Balancers State View

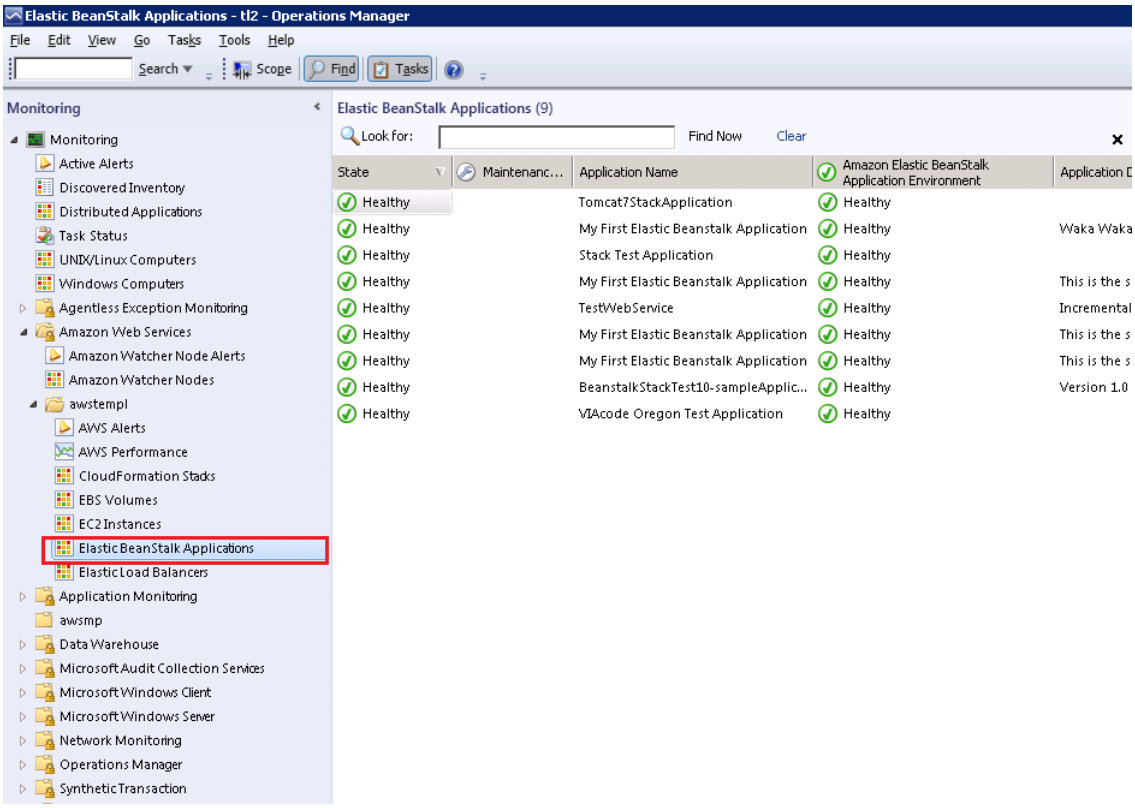
  特定の AWS アカウントの、すべての Load Balancing (全アベイラビリティーゾーン) のヘルス状態を表示します。
- Elastic Load Balancing ダイアグラムビュー

  Elastic Load Balancing と他のコンポーネントの関係を表示します。次の図に例を示します。



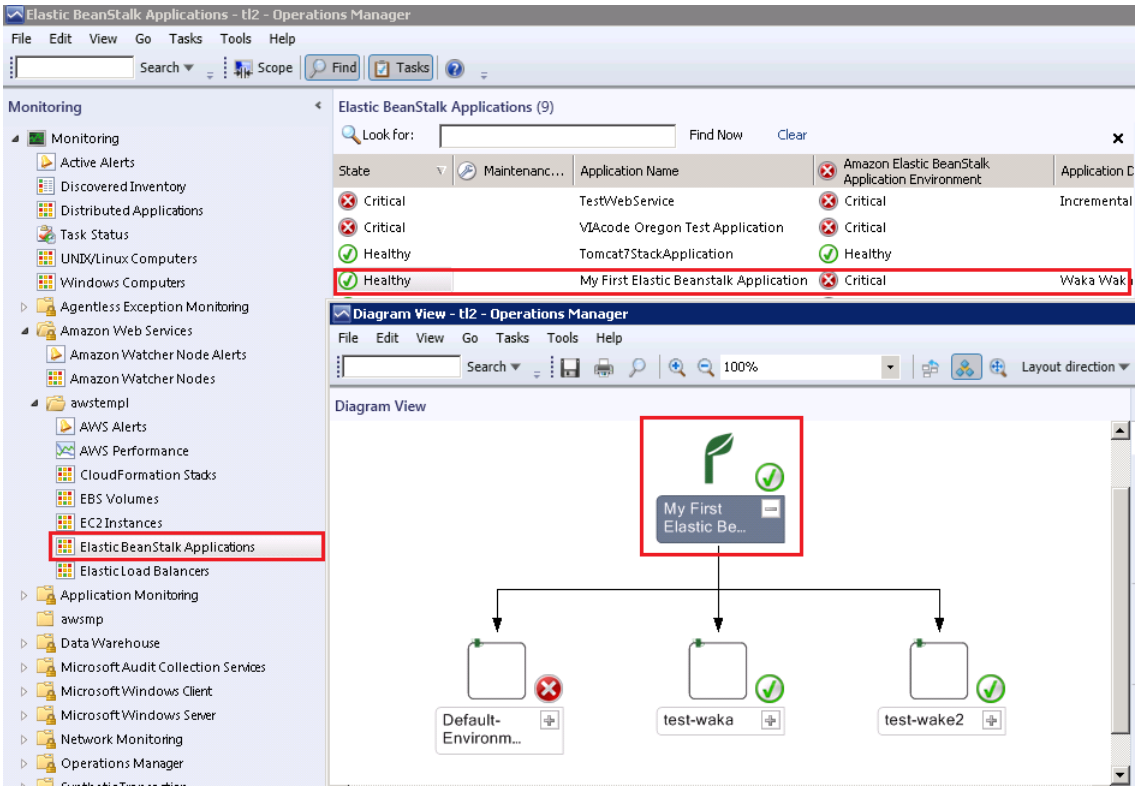
- AWS Elastic Beanstalk Application State View

  検出されたすべての AWS Elastic Beanstalk アプリケーションの状態を表示します。



- AWS Elastic Beanstalk Application Diagram View

  AWS Elastic Beanstalk アプリケーション、アプリケーション環境、アプリケーション設定、および
  アプリケーションリソースオブジェクトを表示します。



タスク

AWS マネジメントパックを使用して、Amazon EC2 インスタンスに対して多くのタスクを実行できます。

Amazon EC2 インスタンスのタスク

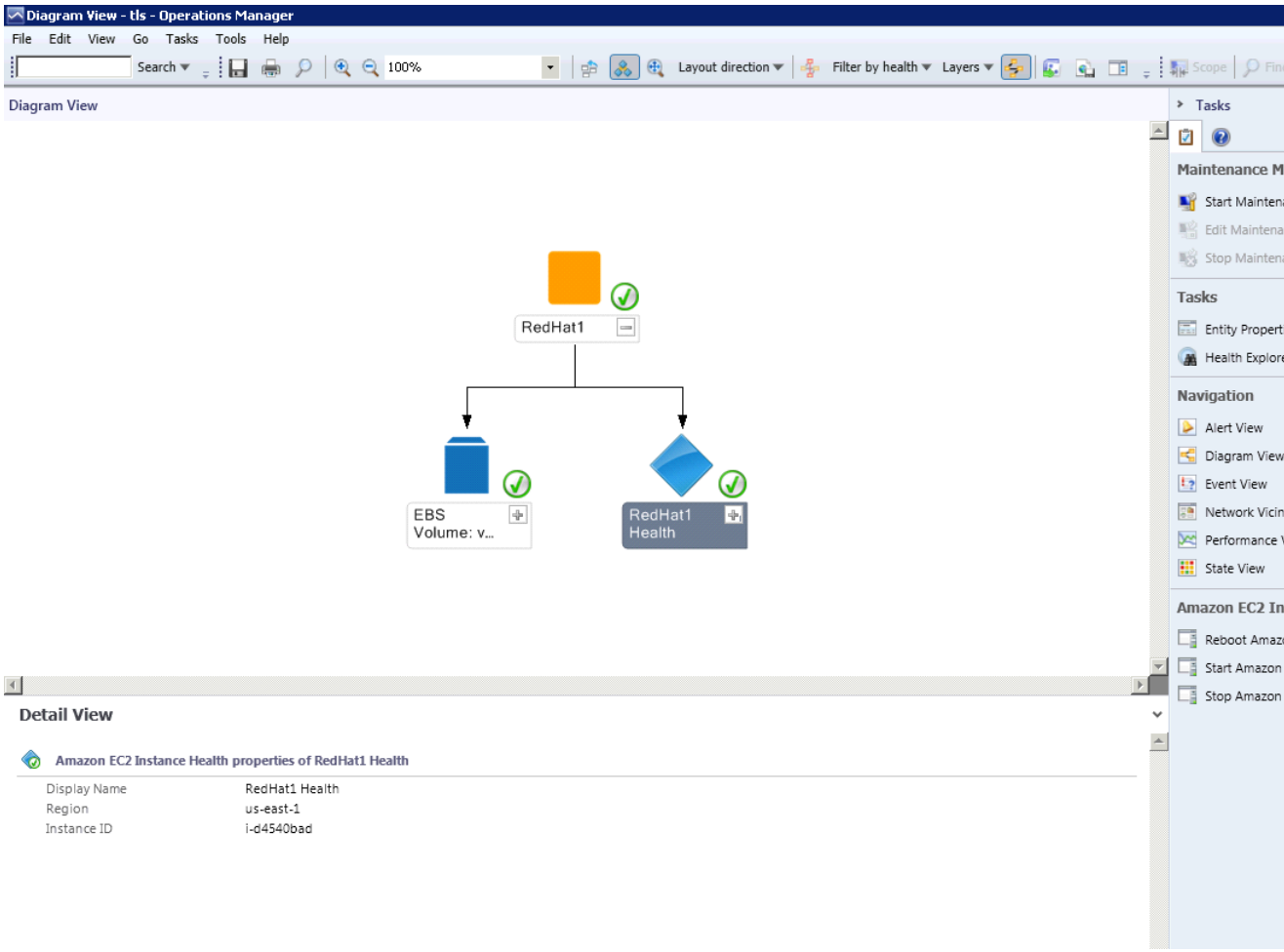
EC2 インスタンス状態ビューで Amazon EC2 インスタンスを選択すると、インスタンスのヘルスタスクを実行できます。

- AWS マネジメントコンソールに接続する: ウェブブラウザでAWS マネジメントコンソールを起動します。
- Amazon EC2 インスタンスに対して RDP を開く: RDP 接続を、選択した Amazon EC2 Windows インスタンスに開きます。

Amazon EC2 インスタンスのヘルスタスク:

次のタスクは、ダイアグラムビューで Amazon EC2 インスタンスヘルスエンティティを選択すると実行できるようになります。

- Amazon EC2 インスタンスを再起動する: Amazon EC2 インスタンスをリモートで再起動します。
- Amazon EC2 インスタンスを開始する: 停止している Amazon EC2 インスタンスをリモートで開始します。
- Amazon EC2 インスタンスを停止する: 実行中の Amazon EC2 インスタンスをリモートで停止します。



AWS マネジメントパックの理解

AWS マネジメントパックの検出(オブジェクトおよび関係) とヘルスモデルについて、次のセクションで説明します。

検出

AWS マネジメントパックは次のオブジェクトを検出します。

- Amazon EC2 インスタンス
- Amazon EBS ボリューム
- Elastic Load Balancing
- AWS CloudFormation スタック
- Amazon CloudWatch メトリックス (検出した Amazon EC2、Amazon EBS、Elastic Load Balancing リソースのデフォルトのメトリックス)
- Amazon CloudWatch アラーム (検出されたメトリックスに定義されたもの)
- AWS Elastic Beanstalk アプリケーション
- Auto Scaling グループおよびアベイラビリティーゾーン

Amazon CloudWatch メトリックスの検出では、次のガイドラインが適用されます。

- ダイアグラムビューの Amazon CloudWatch メトリックスでは、メトリックスに Amazon CloudWatch アラームが定義されていない場合「Not Monitored」と表示されます。
- デフォルトの Amazon CloudWatch メトリックスだけが Operations Manager に表示されます。カスタム Amazon CloudWatch メトリックスは Operations Manager に表示されません。
- AWS CloudFormation スタックにはデフォルトの Amazon CloudWatch メトリックスがありません。
- 停止された Amazon EC2 インスタンスや使用されていない Amazon EBS ボリュームはデフォルトの Amazon CloudWatch メトリックスのデータを生成しません。
- Amazon EC2 インスタンスが開始してから、Operations Manager に Amazon CloudWatch メトリックスが表示されるまでに 30 分ほどかかります。
- Amazon CloudWatch は、監視データを 2 週間保持します。AWS リソースが終了しても監視データは保持されます。このデータは Operations Manager に表示されます。

AWS マネジメントパックは次の関係を検出します。

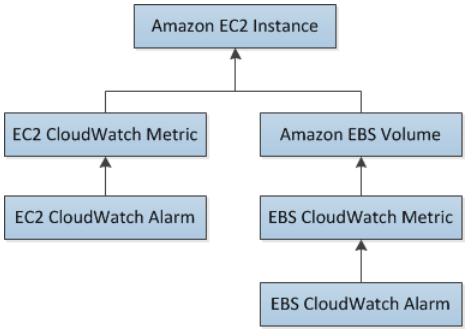
- AWS CloudFormation スタックとその Elastic Load Balancing リソースまたは Amazon EC2 リソース
- Elastic Load Balancing ロードバランサーとその Amazon EC2 インスタンス
- Amazon EC2 インスタンスとその Amazon EBS ボリューム
- Amazon EC2 インスタンスとその Windows/Linux オペレーティングシステム
- AWS Elastic Beanstalk アプリケーションとその環境、設定、リソース

AWS マネジメントパックは Amazon EC2 インスタンスとそこで実行されるオペレーティングシステムの関係を自動的に検出します。この関係を検出するには、Operations Manager エージェントが Amazon EC2 インスタンスにインストール、設定されており、対応するオペレーティングシステム管理パックが Operations Manager にインポートされている必要があります。

これらの検出の詳細、検出の順序、デフォルトの時間間隔については、検出 (p. 87) を参照してください。

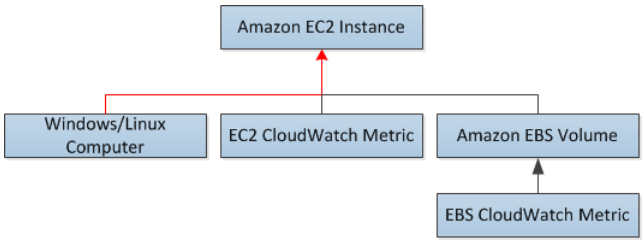
ヘルスモデル

次の図は、AWS マネジメントパックでヘルス状態がロールアップする方法を表しています。



Amazon CloudWatch アラームのヘルス状態は対応する Amazon CloudWatch メトリックスにロールアップします。Amazon EC2 の Amazon CloudWatch メトリックスはそのヘルス状態を Amazon EC2 インスタンスにロールアップします。同様に、Amazon EBS の Amazon CloudWatch メトリックスはヘルス状態を Amazon EBS ボリュームにロールアップします。Amazon EC2 インスタンスによって使用される Amazon EBS ボリュームのヘルス状態を Amazon EC2 インスタンスにロールアップします。

Amazon EC2 インスタンスとそのオペレーティングシステムの関係が検出されると、オペレーティングシステムのヘルス状態が Amazon EC2 インスタンスにロールアップします。

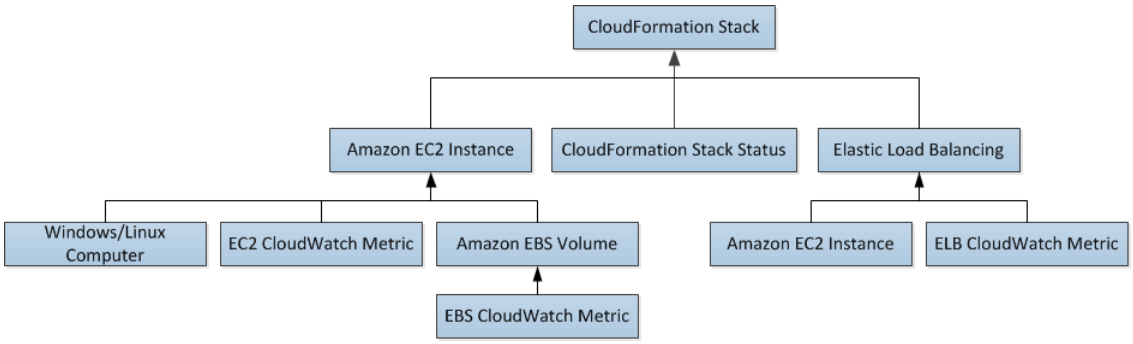


AWS CloudFormation スタックのヘルス状態は、AWS CloudFormation スタック自身とそのリソース（Elastic Load Balancing ロードバランサーと Amazon EC2 インスタンス）の状態に依存します。

次の表は、AWS CloudFormation スタックの状態とそのヘルス状態の対応を示しています。

ヘルス状態	AWS CloudFormation スタックの状態	コメント
エラー	CREATE_FAILED DELETE_IN_PROGRESS DELETE_FAILED UPDATE_ROLLBACK_FAILED	ほとんどの場合使用できる
警告	UPDATE_ROLLBACK_IN_PROGRESS UPDATE_ROLLBACK_COMPLETE_CLEANUP_IN_PROGRESS UPDATE_ROLLBACK_COMPLETE	問題発生後に復旧中
健全	CREATE_COMPLETE UPDATE_IN_PROGRESS UPDATE_COMPLETE_CLEANUP_IN_PROGRESS UPDATE_COMPLETE	使用できる

AWS CloudFormation スタックのヘルス状態の完全なロールアップモデルは以下のとおりです。



AWS マネジメントパックのカスタマイズ

このセクションでは、AWS マネジメントパックのカスタマイズ方法について説明します。

AWS マネジメントパックで実装される検出、監視、ルールの時間間隔の設定を変更できます。デフォルトの時間間隔の詳細については、検出、監視、ルール、イベント (p. 86) を参照してください。AWS マネジメントパックは、オペレーションコンソールの [作成] ワークスペースでデフォルトを上書きすることによりカスタマイズできます。

上書きする方法の詳細については、 *Microsoft TechNet* ウェブサイトのターゲット設定と上書きによる監視の調整を参照してください。

カスタムのルールや監視の作成方法の詳細については、 *Microsoft TechNet* ウェブサイトのSystem Center 2012 の運用マネージャーの作成、または System Center Operations Manager 2007 R2 Management Pack Authoring Guide を参照してください。

AWS マネジメントパックのトラブルシューティング

このセクションでは、トラブルシューティングのヒントをご紹介します。

- システムセンター 2012 – Operations Manager の最新の更新プログラムロールアップがインストールされていることを確認します。
- AWS マネジメントパックには、更新プログラムのロールアップ 1 以上が必要です。
- AWS マネジメントパックをインポート後、監視の追加ウィザードを実行して AWS マネジメントパックを設定してあることを確認します。詳細については、「ステップ 1: AWS マネジメントパックのインストール (p. 66)」を参照してください。
- 十分な時間 (10 ~ 20 分) をかけて AWS リソースを検出します。
- 監視ノードが適切に設定されていることを確認します。
  - プロキシエージェントが有効になっている。詳細については、「ステップ 2: 監視ノードの設定 (p. 68)」を参照してください。
  - 監視ノードがインターネットに接続できる。
  - 監視ノードのアクションアカウントにローカル管理者権限がある。
  - 監視ノードに .NET framework 3.5.1. 以降がインストールしてある。
- 監視ノードが健全で、すべてのアラートが解消済みであることを確認します。詳細については、「ビュー (p. 73)」を参照してください。
- AWS 実行アカウントが有効であることを確認します。
  - アクセスキー ID とシークレットアクセスキーの値が正しい。
  - アクセスキーがアクティブである (AWS マネジメントコンソールの [My Account] -> [Security Credentials] ページで確認) 。
  - IAM ユーザーにリードオンリー以上のアクセス権限がある。
    - Amazon CloudWatch メトリックスが「Not Monitored」と表示される場合、Amazon CloudWatch メトリックスに少なくとも 1 つの Amazon CloudWatch アラームが定義されていることを確認してください。
    - トラブルシューティングをさらに続けるには、イベントログの情報を使用します。
    - 管理サーバーおよび監視ノードの Operations Manager イベントログを確認してください。詳細については、イベント (p. 90)AWS マネジ ントパックが Operations Managerイベントログに書き出したすべてのイベントのリストを参照してください。

検出、監視、ルール、イベント

このトピックでは、AWS マネジメントパックによって実行される検出、監視、ルールについて取り上げます。また、AWS マネジメントパックが管理サーバーおよび監視ノードについて Operations Manager イベントログに書き出すイベントのリストも取り上げます。

検出

検出は、AWS マネジメントパックによって監視される AWS リソースです。

検出	実行場所	時間間隔 (秒)
監視ノード検出 ルート管理サーバーを対象にして、監視ノードオブジェクトを作成します。	管理サーバー	14400
UNIX および Windows コンピュータ検出 Amazon EC2 インスタンスで実行中の UNIX および Windows コンピュータを検出します。その結果、コンピュータでシンプルな URL クエリスクリプトが実行されて、Amazon EC2 インスタンスオブジェクトと Windows および UNIX コンピュータのリンクに使用できる Amazon EC2 インスタンス ID を特定します。この検出により、AmazonComputerLink オブジェクトのプロパティが追加されます。	UNIX / Windows コンピュータ	14400
Amazon EC2 インスタンスと Windows または UNIX コンピュータの関係の検出 Amazon EC2 インスタンスと Windows または UNIX コンピュータの関係を検出します。	管理サーバー	14400

検出	実行場所	時間間隔（秒）
AWS Elastic Beanstalk 検出 AWS Elastic Beanstalk と環境、リソース、設定の関係を検出します。	監視ノード	14400

監視

監視は AWS リソースのヘルス状態を測定するために使用されます。

監視	実行場所	時間間隔（秒）
AWS CloudFormation スタックの状態	監視ノード	900
Amazon CloudWatch メトリックスアラーム	監視ノード	900
Amazon EBS ボリュームの状態	監視ノード	900
Amazon EC2 インスタンスの状態	監視ノード	900
Amazon EC2 インスタンスのシステムの状態	監視ノード	900
監視ノードと Amazon Cloud の接続性	監視ノード	900

ルール

ルールは（Amazon CloudWatch メトリックスに基づいて）アラートを作成し、分析と報告を行うためにデータを収集します。

ルール	実行場所	時間間隔（秒）
AWS リソース検出ルール 監視ノードを対象に、AWS API を使用して AWS リソースのオブジェクトを検出します。検出の対象となる AWS リソースは、Amazon EC2 インスタンス、Amazon Elastic Block Store ボリューム、Elastic Load Balancing、AWS CloudFormation スタックです。この検出には Amazon CloudWatch メトリックスやアラームの検出は含まれません。この検出が完了すると、「Not Monitored」状態にある AWS リソースのオブジェクトが表示されます。	監視ノード	14400
Amazon CloudWatch メトリックスとアラームの検出ルール 既に検出された AWS リソースのオブジェクトを対象とし、デフォルトの Amazon CloudWatch メトリックスとアラーム（メトリックスに関連付けられたものがある場合）を検出します。	監視ノード	14400
Amazon Elastic Block Store ボリュームのパフォーマンスメトリックスデータ収集ルール	監視ノード	900

ルール	実行場所	時間間隔 (秒)
Amazon EC2 インスタンスのパフォーマンスメトリックスデータ収集ルール	監視ノード	900
Elastic Load Balancing ロードバランシングのパフォーマンスメトリックスデータ収集ルール	監視ノード	900

イベント

イベントは、監視対象のリソースに関連するアクティビティを報告します。イベントは、Operations Manager イベントログに書き出されます。

イベント ID	説明
4101	Amazon EC2 インスタンスの検出 (通常検出) が終了しました
4102	Elastic Load Balancing メトリックスの検出、 Amazon EBS ボリュームメトリックスの検出、 Amazon EC2 インスタンスメトリックスの検出が終了しました
4103	Amazon CloudWatch メトリックスアラームの検出が終了しました
4104	Amazon Windows コンピュータの検出が終了しました
4105	Amazon メトリックスアラームの収集が終了しました
4106	EC2 インスタンスとコンピュータの関係の検出が終了しました
4107	AWS CloudFormation スタックの状態の収集が終了しました
4108	監視ノードの可用性状態の収集が終了しました
4109	Amazon メトリックス収集ルールが終了しました
4110	Amazon インスタンスの状態を変更するタスクが終了しました
4111	EC2 インスタンスの状態の監視ステートが終了しました
4112	Amazon EBS ボリュームの状態の監視ステートが終了しました
4113	Amazon EC2 インスタンスのスケジュールイベントの監視ステートを算出しました
4114	Amazon EBS スケジュールイベントの監視ステートを算出しました
4115	AWS Elastic Beanstalk 検出が終了しました
4116	AWS Elastic Beanstalk 環境状態ステートを算出しました
4117	AWS Elastic Beanstalk 環境の運用ステートを算出しました

イベント ID	説明
4118	AWS Elastic Beanstalk 環境の構成ステートを算出しました

VPC での Windows インスタンスのセカンダリプライベート IP アドレスの設定

EC2-VPC では、インスタンスに複数のプライベート IP アドレスを指定できます。VPC でインスタンスにセカンダリプライベート IP アドレスを割り当てたら、セカンダリプライベート IP アドレスを認識するようにインスタンスのオペレーティングシステムを設定する必要があります。

セカンダリプライベート IP アドレスを認識するように Windows インスタンスのオペレーティングシステムを設定するには、次の手順を実行します。

- ステップ 1: Windows インスタンスで静的 IP アドレス指定を設定する (p. 93)
- ステップ 2: Windows インスタンスにセカンダリプライベート IP アドレスを設定する (p. 94)
- ステップ3: セカンダリプライベート IP アドレスを使用するようにアプリケーションを設定する (p. 95)

Note

この手順は Windows Server 2008 R2 を対象にしています。Windows インスタンスのオペレーティングシステムによって、手順が異なる場合があります。

前提条件

- ベストプラクティスとして、最新の AMI を使用して Windows インスタンスを起動します。以前の Windows AMI を使用している場合は、次のページに示されているマイクロソフトの最新修正プログラムが適用されていることを確認してください: http://support.microsoft.com/kb/2582281
- VPC でインスタンスを起動した後、セカンダリプライベート IP アドレスを追加します。詳細については、「*Amazon Elastic Compute Cloud User Guide*」の「Multiple IP Addresses」を参照してください。
- この手順を実行した後でインターネットからウェブサイトへの要求を許可するには、Elastic IP アドレスを設定し、それをセカンダリプライベート IP アドレスに関連付ける必要があります。詳細については、「*Amazon Elastic Compute Cloud User Guide*」の「Assigning a Elastic IP Address to the Secondary Private IP Address」を参照してください。

ステップ 1: Windows インスタンスで静的 IP アドレス指定を設定する

Windows インスタンスが複数の IP アドレスを使用するには、インスタンスが DHCP サーバーではなく、静的 IP アドレス指定を使用するように設定する必要があります。

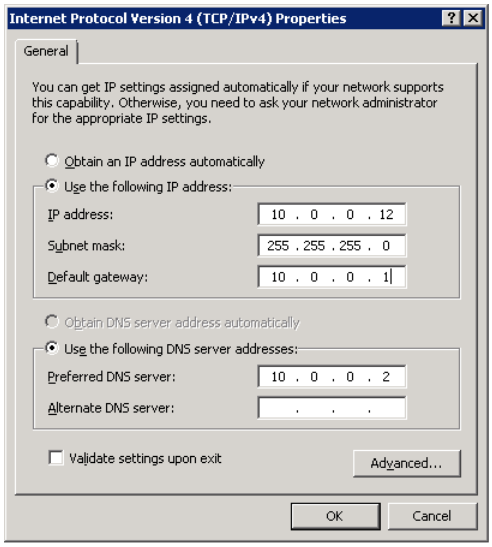
Important

インスタンスで静的 IP アドレス指定を設定する場合、IP アドレスは、AWS コンソール、CLI、または API で割り当てた IP アドレスと正確に一致している必要があります。これらの IP アドレスを誤って入力すると、インスタンスは到達不能になる可能性があります。

インスタンスで DHCP の使用から静的アドレス指定に変換する間、Windows インスタンスへの RDP 接続が数秒間失われます。インスタンスは以前と同じ IP アドレス情報を保持していますが、この情報は静的であり、DHCP によって管理されていません。

Windows インスタンスで静的 IP アドレス指定を設定するには

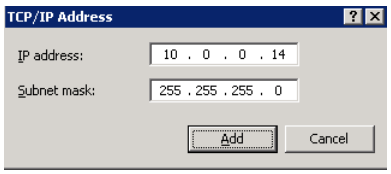
1. インスタンスに接続します。
2. [Start] ボタンをクリックし、[Control Panel] をクリックします。
3. [Network and Internet] をクリックし、[Network and Sharing Center] をクリックします。
4. ネットワークインターフェイス (ローカルエリア接続) をクリックします。
5. [Properties] をクリックします。
6. [Internet Protocol Version 4 (TCP/IPv4)] をクリックし、[Properties] をクリックします。
7. [Properties] ダイアログボックスで、[Use the following IP address]。
8. [Start] ボタンをクリックします。[Search] ボックスに `cmd` と入力し、Enter キーを押します。コマンドプロンプトウィンドウが開きます。
9. コマンドプロンプトで、ipconfig /all コマンドを入力します。
10. ネットワークインターフェイスの現在の IPv4 アドレス、デフォルトゲートウェイ、および DNS サーバーをメモします。
11. [Internet Protocol Version 4 (TCP/IPv4) Properties] ダイアログボックスで、[IP address] ボックスの [Use the following IP address] の下に、コマンドプロンプトウィンドウに表示された IPv4 アドレスを入力します。
12. [Subnet mask] ボックスに、コマンドプロンプトウィンドウに表示されたサブネットマスクを入力します。
13. [Default gateway] ボックスに、コマンドプロンプトウィンドウに表示されたデフォルトゲートウェイの IP アドレスを入力し、[OK] をクリックします。



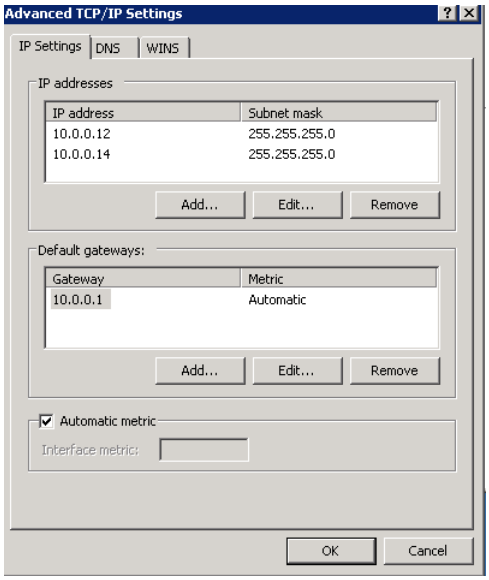
ステップ 2: Windows インスタンスにセカンダリプライベート IP アドレスを設定する

Windows インスタンスにセカンダリ IP アドレスを設定するには

1. Open the Amazon EC2 console at https://console.aws.amazon.com/ec2/.
2. [Navigation] ペインの [Instances] をクリックします。
3. インスタンスを選択します。
4. 下部のペインにある [Description] タブで、セカンダリプライベート IP アドレスをメモします。
5. インスタンスに接続します。
6. Windows インスタンスで、[Start] ボタンをクリックし、[Control Panel] をクリックします。
7. [Network and Internet] をクリックし、[Network and Sharing Center] をクリックします。
8. ネットワークインターフェイス (ローカルエリア接続) をクリックします。
9. [Propierties] をクリックします。
10. [Local Area Connection Properties] ページで、[Internet Protocol Version 4 (TCP/IPv4)] をクリックし、[Properties] をクリックして、[Advanced] をクリックします。
11. [Add] をクリックします。
12. [TCP/IP Address] ダイアログボックスの [IP address] ボックスにセカンダリプライベート IP アドレスを入力します。[Subnet mask] ボックスに、ステップ 1: Windows インスタンスで静的 IP アドレス指定を設定する (p. 93) でプライマリプライベート IP アドレス用に入力したものと同じサブネットマスクを入力し、[Add] をクリックします。



13. IP アドレス設定を確認して、 [OK] をクリックします。



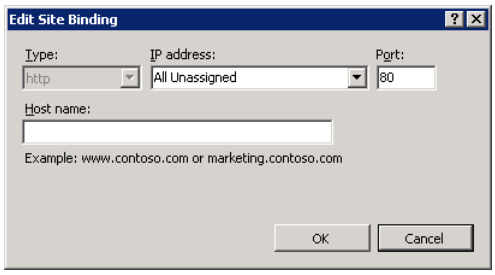
14. もう一度 [OK] をクリックし、[Close] をクリックします。
15. オペレーティングシステムにセカンダリプライベート IP アドレスが追加されたことを確認するには、コマンドプロンプトで、 ipconfig /all というコマンドを実行します。

ステップ3: セカンダリプライベート IP アドレスを使用するようにアプリケーションを設定する

セカンダリプライベート IP アドレスを使用するように任意のアプリケーションを設定できます。たとえば、インスタンスが IIS でウェブサイトを実行している場合、セカンダリプライベート IP アドレスを使用するように IIS を設定できます。

セカンダリプライベート IP アドレスを使用するように IIS を設定するには

1. インスタンスに接続します。
2. インターネットインフォメーションサービス (IIS) マネージャーを開きます。
3. [Connections] ペインで、[Sites] を展開します。
4. ウェブサイトを右クリックし、[Edit Bindings] をクリックします。
5. [Site Bindings] ダイアログボックスの [Type] で、[http] をクリックし、[Edit] をクリックします。
6. [Edit Site Binding] ダイアログボックスの [IP address] ボックスで、セカンダリプライベート IP アドレスをクリックします (デフォルトでは、各ウェブサイトはすべての IP アドレスからの HTTP 要求を受け付けます) 。



7. [OK] をクリックし、[Close] をクリックします。

Amazon EC2 での Windows HPC クラスターのセットアップ

このセクションでは、Amazon Elastic Compute Cloud (Amazon EC2) インスタンスだけを使ってスケーラブルな高性能コンピューティング (HPC) クラスターを起動する手順を説明します。Windows HPC クラスタでは、Active Directory ドメインコントローラと DNS サーバー、ヘッドノード、1 つまたは複数のコンピュートノードが必要です。このセクションで説明する手順に従うと、これらすべてのコンポーネントを構成し、Windows HPC クラスタを起動することができます。ハイパフォーマンスコンピューティングについて詳しくは、「AWS のハイパフォーマンスコンピューティング (HPC)」を参照してください。

Amazon EC2 で Windows HPC クラスターをセットアップする手順

前提条件

Windows HPC クラスター用のインスタンスを設定する前に、必ず以下の条件を満たしていることを確認してください。

- AWS アカウントをまだ開いていない場合は開きます。
- 目的のリージョンで設定を始める前に、Amazon EC2 料金表ページを開き、ドロップダウンリストからそのリージョンを選択して、該当するリージョンでクラスターコンピューティングインスタンスが利用できることを確認します。
- Amazon EC2 コマンドラインツールをインストールします。詳しくは Windows への Amazon EC2 コマンドラインインターフェイスツールのインストール (p. 107) を参照してください。
- オプションで、 HPC Pack 2008 R2 をダウンロードすることもできます。もしくは後で AMI インスタンスに直接 HPC Pack 2008 R2 Express をダウンロードします。

タスク 1: Active Directory ドメインコントローラーをセットアップする

Active Directory ドメインコントローラーは、HPC 環境で認証とリソース中央管理を行うもので、インストールに必要です。Active Directory をセットアップするには次の 3 つの手順を実行します。

1. Active Directory 用のセキュリティグループを作成する。
2. ドメインコントローラー用のインスタンスを起動する。
3. HPC クラスター用にドメインコントローラーを設定する。

Active Directory 用のセキュリティグループのセットアップ

セキュリティグループスクリプト create-AD-sec-groups.bat を実行して、ドメインコントローラー用およびドメインメンバー用のルールを作成します。コマンドラインツールをインストールしていない場合は、Windows Server 2008/Windows Server 2008 R2 のポート要件に合わせたセキュリティグループを手動で作成します。詳細については、 Microsoft ウェブサイトで「ファイアウォールのドメインとの信頼関係を構成する方法」を参照してください。

Active Directory に必要なセキュリティグループを作成するには

1. テキストエディタを使用して create_AD_security.bat (p. 104) の内容をコピーし、ファイルを `create-AD-sec-groups.bat` という名前で、Amazon EC2 コマンドラインツールで設定した、Amazon Web Services への接続元として使用するコンピュータに保存します。
2. ローカル管理者としてファイルを実行します。
3. AWS Management Console にログインして、SG - Domain Controller と SG - Domain Member の 2 つのセキュリティグループが表示されることを確認します。

ドメインコントローラー用のインスタンスの起動

ドメインコントローラーを設定するには、AWS からインスタンスを起動し、そのインスタンスを HPC クラスターのドメインコントローラーに指定します。

ドメインコントローラー用のインスタンスを起動するには

1. Microsoft Windows Server 2008 R2 Base から、[”Domain Controller”]という名前の m1.large Amazon EC2 インスタンスタイプ (または用途に応じた別のインスタンスタイプ) を起動し、[”SG - Domain Controller”]セキュリティグループに割り当てます。
2. Elastic IP アドレスを作成し、この IP アドレスを Domain Controller インスタンスに関連付けます。
   a. ナビゲーションペインの[”Elastic IPs”]をクリックします。
   b. [Allocate New Address] をクリックします。
   c. [”Allocate New Address”]ダイアログボックス内で、[”Yes Allocate”]をクリックします。
   d. 作成した Elastic IP アドレスを選択し、[Associate Address] をクリックします。
   e. [”Associate Address”]ダイアログボックスの[”Instance”]ドロップダウンリストで、ドメインコントローラーインスタンスを選択し、[”Yes Associate”]をクリックします。

HPC クラスター用のドメインコントローラーの設定

次に、作成したインスタンスにログインし、そのインスタンスを HPC クラスターのドメインコントローラーとして設定します。

インスタンスをドメインコントローラーとして設定するには

1. インスタンスに接続します。
2. ["Server Manager"]を開き、Active Directory Domain Services の役割を追加します。
3. Server Manager を使用するか、または DCPromo.exe を実行して、サーバーをドメインコントローラに昇格させます。
4. 新しいフォレスト内に新しいドメインを作成します。
5. 完全修飾ドメイン名 (FQDN) として hpc.local と入力します。
6. Forest Functional Level として["Windows Server 2008 R2"]を選択します。
7. DNS Server オプションが選択されていることを確認し、["Next"]をクリックします。
8. ["Yes, the computer will use an IP address automatically assigned by a DHCP server (not recommended)"]を選択します。
9. 警告メッセージが表示されたら、["Yes"]をクリックして続行します。
10. ウィザードを完了して["Reboot on Completion"]を選択します。
11. インスタンスに hpc.local\administrator としてログインします。
12. ドメインユーザー hpc.local\hpcuser を作成します。

タスク 2: ヘッドノードを設定する

HPC クライアントはすべてヘッドノードに接続します。ヘッドノードは、スケジュールされたジョブを実行できるようにします。ヘッドノードは次の手順で設定します:

1. HPC クラスター用のセキュリティグループを作成します。
2. ヘッドノード用のインスタンスを起動します。
3. HPC Pack をインストールします。
4. クラスターを設定します。

HPC クラスター用のセキュリティグループの作成

セキュリティグループスクリプト create-HPC-sec-group.bat を実行して、[SG - Windows HPC Cluster] という名前のセキュリティグループを、HPC クラスターノードのルールで作成します。コマンドラインツールをインストールしていない場合は、手動でセキュリティグループを作成し、HPC クラスターメンバーがそのセキュリティグループ内だけで通信するようポート要件を設定します。詳細については、 Microsoft のウェブサイトで「Windows ファイアウォール」を参照してください。

HPC クラスターに必要なセキュリティグループを作成するには

1. テキストエディタを使用して create-HPC-sec-group.bat (p. 105) の内容をコピーし、ファイルを `create-HPC-sec-group.bat` という名前で、EC2 コマンドラインツールで設定した、Amazon Web Services への接続元として使用するコンピュータに保存します。
2. ローカル管理者としてファイルを実行します。

3. AWS Management Console にログインし、セキュリティグループ SG - Windows HPC Cluster が表示されることを確認します。

HPC ヘッドノード用のインスタンスの起動

ヘッドノードを設定するために、AWS からクラスターインスタンスを起動して、そのインスタンスを hpc.local のドメインメンバーに設定し、必要なユーザーアカウントを設定します。

ヘッドノード用のインスタンスを設定するには

1. [Microsoft Windows 2008 R2 64-bit for Cluster Instances] から [HPC-Head] という名前のインスタンスを起動し、そのインスタンスを [SG - Windows HPC Cluster] および [SG - Domain Member] の両方のセキュリティグループに割り当てます。
2. インスタンスにログインし、IPConfig /all を使用して、[HPC-Head] から既存の DNS サーバーアドレスを取得します。
3. [HPC-Head] NIC の TCP/IPv4 プロパティを、プライマリ DNS として [Domain Controller] Elastic IP アドレスを含めるように更新し、前の手順の追加 DNS IP アドレスを指定します。
4. hpc.local\administrator 証明書 (ドメイン管理者アカウント) を使って、コンピュータを hpc.local ドメインに参加させます。
5. hpc.local\hpcuser をローカル管理者として追加します。証明書を求められたら、hpc.local\administrator を指定し、再起動します。
6. hpc.local\hpcuser として [HPC-Head] にログインし直します。

HPC Pack のインストール

このセクションでは、HPC Pack のダウンロードとインストールの方法を説明します。

HPC Pack をインストールするには

1. hpc.local\hpcuser アカウントを使って [HPC-Head] インスタンスに接続します。
2. [Server Manager] を使って、Administrators の Internet Explorer セキュリティ強化の構成 (IE ESC) をオフにします。
   a. [Server Manager] の [Security Information] で、[Configure IE ESC] をクリックします。
   b. Administrators の IE ESC をオフにします。
3. [HPC-Head] に HPC Pack 2008 R2 Express をインストールします。
   a. 次のページから、HPC Pack 2008 R2 Express を [HPC-Head] にダウンロードします: http://go.microsoft.com/fwlink/?LinkID=198084
   b. フォルダにファイルを抽出し、フォルダを開いて、[setup.exe] をダブルクリックします。
   c. [”HPC Pack 2008 R2 Express”]を選択し、[”Next”]をクリックします。
   d. ライセンス契約に同意する場合は同意するオプションを選択し、[”Next”]をクリックします。
   e. インストールのページで[”Create a new HPC cluster by creating a head node”]を選択し、[”Next”]をクリックします。
   f. デフォルト設定のままですべてのデータベースをヘッドノードにインストールし、[”Next”]をクリックします。
   g. ウィザードを終了します。

HPC クラスターでのヘッドノードの設定

このセクションでは、HPC クラスターにヘッドノードを設定する方法を説明します。

HPC クラスターにヘッドノードを設定するには

1. [HPC Cluster Manager] を開始します。
2. [”Deployment To-Do List”]内で、[”Configure your network”]を選択します。
   a. ウィザードでデフォルトオプション (5) を選択し、[”Next”]をクリックします。
   b. すべての画面でデフォルトを受け入れ、サーバーの更新方法とカスタマーフィードバックへの参加方法を選択して、ウィザードを終了します。
   c. [”Configure”]をクリックします。
3. [”Provide Network Credentials”]を選択し、hpc.local\hpcuser 証明書を指定します。
4. [”Configure the naming of new nodes”]を選択し、[”OK”]をクリックします。
5. [”Create a node template”]を選択します。
   a. [”Compute node template”]を選択し、[”Next”]をクリックします。
   b. [”Without operating system”]を選択し、デフォルト設定のまま続行します。
   c. [Create] をクリックします。

タスク 3: コンピュートノードをセットアップする

コンピュートノードのセットアップは、次の手順で行います。

1. コンピュートノード用のインスタンスを起動します。
2. インスタンスに HPC Pack をインストールします。
3. クラスターにコンピュートノードを追加します。

HPC コンピュートノード用のインスタンスの起動

コンピュートノードを設定するために、AWS からクラスターインスタンスを起動し、そのインスタンスを hpc.local のドメインメンバーに設定して、必要なユーザーアカウントを設定します。

コンピュートノード用のインスタンスを設定するには

1. [Microsoft Windows 2008 R2 64-bit for Cluster Instances] から [HPC-Compute] という名前のインスタンスを起動し、そのインスタンスを [SG - Windows HPC Cluster] および [SG - Domain Member] の両方のセキュリティグループに割り当てます。
2. インスタンスにログインし、[IPConfig /all] を使用して、HPC-Compute から既存の DNS サーバーアドレスを取得します。
3. [HPC-Compute] NIC の TCP/IPv4 プロパティを、プライマリ DNS として Domain Controller Elastic IP アドレスを含めるように変更し、前の手順の追加の DNS IP アドレスを指定します。
4. hpc.local\administrator 証明書 (ドメイン管理者アカウント) を使って、コンピュータを hpc.local ドメインに参加させます。

5. hpc.local\hpcuser をローカル管理者として追加します。証明書を求められたら、hpc.local\administrator を指定し、再起動します。
6. hpc.local\hpcuser として [HPC-Compute] にログインし直します。

コンピュートノードへの HPC Pack のインストール

このセクションでは、HPC クラスターのコンピュートノードに HPC パックをダウンロードしてインストールする方法を説明します。

コンピュートノードに HPC Pack をインストールするには

1. hpc.local\hpcuser アカウントを使って [HPC-Compute] インスタンスに接続します。
2. [Server Manager] を使って、Administrators の Internet Explorer セキュリティ強化の構成（IE ESC）をオフにします。
   a. [Server Manager] の [Security Information] で、[Configure IE ESC] をクリックします。
   b. Administrators の IE ESC をオフにします。
3. [HPC-Compute] に HPC Pack 2008 R2 Express をインストールします。
   a. 次のページから、HPC Pack 2008 R2 Express を [HPC-Compute] にダウンロードします: http://go.microsoft.com/fwlink/?LinkID=198084
   b. フォルダにファイルを抽出し、フォルダを開いて、[setup.exe] をダブルクリックします。
   c. [”HPC Pack 2008 R2 Express”]を選択し、[”Next”]をクリックします。
   d. ライセンス契約に同意する場合は同意するオプションを選択し、[”Next”]をクリックします。
   e. インストールのページで[”Join an existing HPC cluster by creating a new compute node”]を選択し、[”Next”]をクリックします。
   f. [HPC-Head] インスタンスのコンピュータ名 FQDN を指定し、デフォルトを選択します。
   g. ウィザードを終了します。

HPC クラスターへのコンピュートノードの追加

クラスター設定を完了するために、ヘッドノードから、コンピュートノードをクラスターに追加します。

クラスターにコンピュートノードを追加するには

1. hpc.local\hpcuser として [HPC-Head] にログインします。
2. [HPC-Head] で、[HPC Cluster Manager] を開きます。
3. 左下のペインで[”Node Management”]を選択します。
4. [”Unapproved”]バケット内にコンピュートノードが表示されたら、リストされているノードを右クリックして[”Add Node”]を選択します。
   a. [”Add compute nodes or broker nodes that have already been configured”]を選択します。
   b. ノードの横にあるチェックボックスを選択して、[”Add”]をクリックします。

5. ノードを右クリックして[”Bring Online”]をクリックします。

タスク 4: HPC コンピュートノードをスケーリングする (オプション)

コンピュートノードをスケーリングするには

1. hpc.local\hpcuser として [HPC-Compute] にログインします。
2. HP Pack 2008 R2 Express インストールパッケージからローカルにダウンロードしたファイルをすべて削除します。(すでにセットアップを実行してイメージ上にこれらのファイルが作成されているため、AMI 用にクローンを保持する必要はありません。)
3. `C:\Program Files\Amazon\Ec2ConfigService` からファイル sysprep2008.xml を開きます。
4. `<settings pass="specialize">` の末尾に以下のセクションを追加します。このとき、`hpc.local`、`password`、および `hpcuser` をお使いの環境に合わせて適切な内容に置き換えてください。

```
 <component name="Microsoft-Windows-UnattendedJoin" processorArchitec
ture="amd64" publicKeyToken="31bf3856ad364e35"
language="neutral" versionScope="nonSxS" xmlns:wcm="http://schemas.mi
crosoft.com/WMIConfig/2002/State"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
            <Identification>
                <UnsecureJoin>false</UnsecureJoin>
               <Credentials>
                    <Domain>hpc.local</Domain>
                   <Password>Password</Password>
                    <Username>hpcuser</Username>
                </Credentials>
                <JoinDomain>hpc.local</JoinDomain>
            </Identification>
        </component>
```

5. sysprep2008.xml を保存します。
6. [”Start”]をクリックし、[”All Programs”]をポイントして[”EC2ConfigService Settings”]をクリックします。
   a. [”General”]タブをクリックし、[”Set Computer Name”]チェックボックスをオフにします。
   b. [”Bundle”]タブをクリックし、[”Run Sysprep and Shutdown Now”]をクリックします。
7. Sign in to the AWS Management Console and open the Amazon EC2 console at https://console.aws.amazon.com/ec2/.
8. [”Navigation”]内で、[”Instances”]をクリックします。
9. インスタンスのステータスが [Stopped] になるまで待ちます。
10. インスタンスを右クリックし、[”Create Image (EBS AMI)”]を選択します。
11. イメージ名とイメージの説明を入力し、[”Create This Image”]をクリックすると、インスタンスから AMI が作成されます。
12. シャットダウンされた元の [HPC-Compute] ノードを開始します。

13. hpc.local\hpcuser アカウントを使ってヘッドノードに接続します。
14. [HPC Cluster Manager] から、エラー状態で表示されるようになった古いノードを削除します。
15. AWS Management Console の[”Navigation”]で[”AMIs”]をクリックします。
16. 作成した AMI を使ってクラスタに追加のノードを作成します。

これで、作成した AMI から、追加のコンピュートノードをいくつでも起動できるようになりました。ノードは自動的にドメインに追加されますが、クラスターへの追加は手動で行う必要があります。ヘッドノードを使って [HPC Cluster Manager] から設定済みノードとして追加した後、オンラインにします。

Lizard パフォーマンス測定アプリケーションの実行

必要に応じて Lizard アプリケーションを実行します。これはお使いの HPC クラスターの演算性能と効率を測定するものです。これを行うには、 http://www.microsoft.com/download/en/details.aspx?id=8433 から lizard_x64.msi インストーラをダウンロードして、hpc.local\hpcuser としてヘッドノード上で直接実行します。

create_AD_security.bat

以下の .bat ファイルは、Active Directory 環境用に 2 つのセキュリティグループを作成します。1 つは Active Directory ドメインコントローラー用、もう 1 つは Active Directory ドメインメンバーサーバー用です。

```
set DC="SG - Domain Controller"
set DM="SG - Domain Member"


:: ============================================================================
:: Creates Security groups Prior to Adding Rules
:: ============================================================================

call ec2addgrp %DM% -d "Active Directory Domain Member"
call ec2addgrp %DC% -d "Active Directory Domain Controller"


:: ============================================================================
:: Security group for Domain Controller
:: ============================================================================

:: For LDAP and related services. Details at link below
:: http://support.microsoft.com/kb/179442
call ec2auth %DC% -o %DM% -P UDP -p 123
call ec2auth %DC% -o %DM% -P TCP -p 135
call ec2auth %DC% -o %DM% -P UDP -p 138
call ec2auth %DC% -o %DM% -P TCP -p "49152-65535"
call ec2auth %DC% -o %DM% -P TCP -p 389
call ec2auth %DC% -o %DM% -P UDP -p 389
call ec2auth %DC% -o %DM% -P TCP -p 636
```

```
call ec2auth %DC% -o %DM% -P TCP -p 3268
call ec2auth %DC% -o %DM% -P TCP -p 3269
call ec2auth %DC% -o %DM% -P TCP -p 53
call ec2auth %DC% -o %DM% -P UDP -p 53
call ec2auth %DC% -o %DM% -P TCP -p 88
call ec2auth %DC% -o %DM% -P UDP -p 88
call ec2auth %DC% -o %DM% -P TCP -p 445
call ec2auth %DC% -o %DM% -P UDP -p 445

:: For ICMP as required by Active Directory
call ec2auth %DC% -P ICMP -t -1:-1

:: For Elastic IP to communicate with DNS
call ec2auth %DC% -s 0.0.0.0/0 -P UDP -p 53

:: For RDP for connecting to desktop remotely
call ec2auth %DC% -P TCP -p 3389

:: ============================================================================
:: Security group for Domain Member
:: ============================================================================

:: For LDAP and related services. Details at link below
:: http://support.microsoft.com/kb/179442

call ec2auth %DM% -o %DC% -P TCP -p "49152-65535"
call ec2auth %DM% -o %DC% -P UDP -p "49152-65535"
call ec2auth %DM% -o %DC% -P TCP -p 53
call ec2auth %DM% -o %DC% -P UDP -p 53
```

create-HPC-sec-group.bat

以下の .bat ファイルは HPC クラスターノード用に 1 つのセキュリティグループを作成します。この bat ファイルを、Amazon Web Services への接続に使用するクライアントコンピュータから実行します。

```
set HPC="SG - Windows HPC Cluster"

:: ============================================================================
:: Creates Security groups Prior to Adding Rules
:: ============================================================================


call ec2addgrp %HPC% -d "Windows HPC Server 2008 R2 Cluster Nodes"


:: ============================================================================
:: Security group for Windows HPC Cluster
:: ============================================================================


:: For HPC related services. Details at link below
```

```
:: http://technet.microsoft.com/en-us/library/ff919486(WS.10).aspx#BKMK_Firewall
call ec2auth %HPC% -o %HPC% -P TCP -p 80
call ec2auth %HPC% -o %HPC% -P TCP -p 443
call ec2auth %HPC% -o %HPC% -P TCP -p 1856
call ec2auth %HPC% -o %HPC% -P TCP -p 5800
call ec2auth %HPC% -o %HPC% -P TCP -p 5801
call ec2auth %HPC% -o %HPC% -P TCP -p 5969
call ec2auth %HPC% -o %HPC% -P TCP -p 5970
call ec2auth %HPC% -o %HPC% -P TCP -p 5974
call ec2auth %HPC% -o %HPC% -P TCP -p 5999
call ec2auth %HPC% -o %HPC% -P TCP -p 6729
call ec2auth %HPC% -o %HPC% -P TCP -p 6730
call ec2auth %HPC% -o %HPC% -P TCP -p 7997
call ec2auth %HPC% -o %HPC% -P TCP -p 8677
call ec2auth %HPC% -o %HPC% -P TCP -p 9087
call ec2auth %HPC% -o %HPC% -P TCP -p 9090
call ec2auth %HPC% -o %HPC% -P TCP -p 9091
call ec2auth %HPC% -o %HPC% -P TCP -p 9092
call ec2auth %HPC% -o %HPC% -P TCP -p "9100-9163"
call ec2auth %HPC% -o %HPC% -P TCP -p "9200-9263"
call ec2auth %HPC% -o %HPC% -P TCP -p 9794
call ec2auth %HPC% -o %HPC% -P TCP -p 9892
call ec2auth %HPC% -o %HPC% -P TCP -p 9893
call ec2auth %HPC% -o %HPC% -P UDP -p 9893

:: For HPC related services, these are NOT in the first table but are there in
 the third table at link below
:: http://technet.microsoft.com/en-us/library/ff919486(WS.10).aspx#BKMK_Firewall
call ec2auth %HPC% -o %HPC% -P TCP -p 6498
call ec2auth %HPC% -o %HPC% -P TCP -p 7998
call ec2auth %HPC% -o %HPC% -P TCP -p 8050
call ec2auth %HPC% -o %HPC% -P TCP -p 5051

:: For RDP for connecting to desktop remotely
call ec2auth %HPC% -P TCP -p 3389
```

Windows への Amazon EC2 コマンドラインインターフェイスツールのインストール

Amazon EC2 コマンドラインインターフェイスツール（*CLI ツール*とも呼ばれる）では、Amazon EC2 API アクションをラップします。このツールは Java で作成されており、Windows と Linux/UNIX/Mac OSX 用のシェルスクリプトが含まれています。

Amazon EC2 CLI ツールを使用するには、まずダウンロードし、AWS アカウントを使用するように設定する必要があります。お使いのコンピューターまたは Amazon EC2 インスタンスでツールを設定できます。

以下のタスクを完了して、Amazon EC2 環境を設定します。

1. CLI ツールをダウンロードする (p. 108)
2. JAVA_HOME 環境変数を設定する (p. 108)
3. EC_HOME 環境変数を設定する (p. 109)
4. 環境変数 AWS_ACCESS_KEY および AWS_SECRET_KEY を設定する (p. 110)
5. （オプション）リージョンを設定する (p. 111)
6. (オプション)プロキシを使用する (p. 112)
7. リモートデスクトップをダウンロードする (p. 113)

Note

ここで説明する手順は、Windows 7 クライアントを対象としています。他のバージョンの Windows を使用している場合、一部のタスクで必要な操作が異なることがあります。

タスク 1: コマンドラインインターフェイスツール (CLI ツール) をダウンロードする

CLI ツールは Amazon EC2 CLI Tools から ZIP ファイルとして入手できます。ツールは Java で作成されており、Windows と Linux/UNIX/Mac OSX 用のシェルスクリプトが含まれています。この ZIP ファイルは自己完結型で、インストールの必要はありません。ファイルをダウンロードして解凍するだけです。

タスク 2: JAVA_HOME 環境変数を設定する

Amazon EC2 CLI ツールには Java が必要です。ツールは、`JAVA_HOME` 環境変数を読み込んで Java ランタイムを探します。この環境変数には、`bin` というサブディレクトリを含むディレクトリのフルパスを指定する必要があります。bin ディレクトリには、インストールした Java 実行可能ファイル (`java.exe`) が保存されています。

コンピュータまたはインスタンスの **`JAVA_HOME`** 環境変数を設定するには

1. Java 1.6 以降がインストールされていない場合は、ダウンロードしてインストールしてください。JRE または JDK のどちらでもかまいません。各種プラットフォーム用 JRE を参照およびダウンロードするには、 無料 Java のダウンロードにアクセスしてください。
2. `JAVA_HOME` に Java のホームディレクトリのフルパスを設定します。例えば、Java 実行可能ファイルが `C:\Program Files (x86)\Java\jre7\bin` にある場合、`JAVA_HOME` に `C:\Program Files (x86)\Java\jre7` を設定します。

   Important

   これらの手順を行っても、現在開いているコマンドプロンプトウィンドウの環境変数は更新されません。手順を完了してからコマンドプロンプトウィンドウを開くと、更新分が反映されます。環境が適切に設定されたことを確認するために新しいコマンドプロンプトウィンドウを開く必要があるのは、このためです。

   a. [Start] ボタンをクリックして [Computer] を右クリックし、[Properties] をクリックします。
   b. [”Advanced system settings”]をクリックします。
   c. [”Environment Variables”]をクリックします。
   d. [”System variables”]の下にある[”New”]をクリックします。
   e. [”Variable name”]に **`JAVA_HOME`** と入力します。
   f. [Variable value] に Java ホームディレクトリへのパスを入力します(例えば、C:\Program Files (x86)\Java\jre7)。

      Important

      `JAVA_HOME` には、bin ディレクトリを含めないでください。これはよくある間違いです。bin を含めると、CLI ツールが動作しません。

   g. [OK] をクリックします。

3. 新しいコマンドプロンプトウィンドウを開き、このコマンドを使用して、`JAVA_HOME` 設定を確認します。

```
C:\> "%JAVA_HOME%"\bin\java -version
```

環境変数が正しく設定してあれば、出力は次のようになります。

```
java version "1.7.0_05"
Java(TM) SE Runtime Environment (build 1.7.0_05-b05)
Java HotSpot(TM) Client VM (build 23.1-b03, mixed mode, sharing)
```

このように出力されない場合は、JAVA_HOME の設定を確認し、誤りを修正して、新しいコマンドプロンプトウィンドウを開き、再度コマンドを実行します。

4. Java 実行可能ファイルを含む bin ディレクトリを、パスに追加します。追加する場所は他のバージョンの Java の前です。

   a. [System variables] の [Path] を選択し、[Edit] をクリックします。

   b. [Variable values] フィールドで、他のバージョンの Java の前に **;%JAVA_HOME%\bin;** を追加します。

5. 新しいコマンドプロンプトウィンドウを開き、このコマンドを使用して、Path 環境変数が更新されたかどうか確認します。

```
C:\> java -version
```

前と同じ出力が表示されるはずです。表示されない場合は、Path の設定を確認して誤りを修正し、新しいコマンドプロンプトウィンドウを開いて再度コマンドを実行します。

タスク 3: EC2_HOME 環境変数を設定する

Amazon EC2 CLI ツールは、サポートライブラリの場所を特定するために EC2_HOME 環境変数を読み取ります。この環境変数に、CLI ツールを解凍した場所へのパスを設定する必要があります。このディレクトリは、ec2-api-tools-*w.x.y.z* という名前です。この *w*、*x*、*y*、および *z* は、バージョン番号の構成要素です。このディレクトリには、bin と lib というサブディレクトリがあります。

コンピュータまたはインスタンスの EC2_HOME 環境変数を設定するには

1. EC2_HOME に、CLI ツールの解凍先であるディレクトリのパスを指定します。

   Important

   これらの手順を行っても、現在開いているコマンドプロンプトウィンドウの環境変数は更新されません。手順を完了してからコマンドプロンプトウィンドウを開くと、更新分が反映されます。環境が適切に設定されたことを確認するために新しいコマンドプロンプトウィンドウを開く必要があるのは、このためです。

   a. [Start] ボタンをクリックして [Computer] を右クリックし、[Properties] をクリックします。

   b. [”Advanced system settings”]をクリックします。

   c. [”Environment Variables”]をクリックします。

   d. [”System variables”]の下にある[”New”]をクリックします。

   e. [”Variable name”]に **EC2_HOME** と入力します。

f. [Variable value]に、CLI ツールのインストール先であるディレクトリへのパスを入力します。例えば、**C:\AWS\EC2\ec2-api-tools-1.6.7.2** と入力します。

2. 新しいコマンドプロンプトウィンドウを開き、このコマンドを使って、EC2_HOME の設定を確認します。

```
C:\> dir "%EC2_HOME%"
```

環境変数が正しく設定されていれば、ディレクトリの一覧出力が表示されます。「ファイルが見つかりません」エラーになった場合は、EC2_HOME の設定を確認し、誤りを修正して、新しいコマンドプロンプトウィンドウを開き、再度コマンドを実行します。

3. ツールの bin ディレクトリを、システムの Path 環境変数に追加します。ガイドのこれ以降の部分では、この作業を完了していることを前提しています。

Path を変更する手順は次のとおりです。

a. [System variables] の [Path] を選択し、[Path] をクリックします。

b. [”Variable values”]に、**;%EC2_HOME%\bin** を追加します。

タスク 4: 環境変数 AWS_ACCESS_KEY および AWS_SECRET_KEY を設定する

アクセスキーによって Amazon EC2 CLI に対してお客様を特定します。アクセスキーには、アクセスキー ID とシークレットアクセスキーの 2 種類があります。アクセスキーを作成した際に安全な場所にアクセスキーを保管する必要があります。Your Security Credentials ページからアクセスキー ID を取得できますが、シークレットアクセスキーを取得することはできません。そのため、シークレットアクセスキーが分からない場合、CLI ツールを使用する前に新しいアクセスキーを作成する必要があります。

コマンドを発行するたびに、--aws-access-key および --aws-secret-key (または -O および -W) オプションを使用してアクセスキーを指定する必要があります。また、以下の環境変数を使用してアクセスキーを保管する方が容易な場合があります。

- AWS_ACCESS_KEY – お客様のアクセスキー ID
- AWS_SECRET_KEY – お客様のシークレットアクセスキー

これらの環境変数は、正しく設定しておけば必要なオプションのデフォルト値として使われるので、コマンドラインへの入力を省略できます。

以下の手順では、アクセスキーを指定する環境変数の作成方法を説明します。

コンピュータまたはインスタンスの環境変数を設定するには

1. [Start] ボタンをクリックして [Computer] を右クリックし、[Properties] をクリックします。
2. [Advanced system settings] をクリックします。
3. [Environment Variables] をクリックします。
4. [System variables] の下にある [New] をクリックします。
5. [Variable name] に [**AWS_ACCESS_KEY**] と入力します。

6. [Variable value] で、アクセスキー ID を指定します。
7. [System variables] の下にある [New] をクリックします。
8. [Variable name] に [**AWS_SECRET_KEY**] と入力します。
9. [Variable value] で、シークレットアクセスキーを指定します。

すべての環境変数が正しく設定されたことを確認するには、新しいコマンドプロンプトウィンドウを開いて次のコマンドを実行します。

```
C:\> ec2-describe-regions
```

環境変数が正しく設定されていれば、次のような出力が表示されます。

```
REGION  us-east-1       ec2.us-east-1.amazonaws.com
REGION  eu-west-1       ec2.eu-west-1.amazonaws.com
REGION  sa-east-1       ec2.sa-east-1.amazonaws.com
REGION  ap-northeast-1  ec2.ap-northeast-1.amazonaws.com
REGION  us-west-2       ec2.us-west-2.amazonaws.com
REGION  us-west-1       ec2.us-west-1.amazonaws.com
REGION  ap-southeast-1  ec2.ap-southeast-1.amazonaws.com
```

「内部コマンドまたは外部コマンドとして認識されていません」というようなエラーが発生した場合は、Path の設定を確認し、誤りを修正して、新しいコマンドプロンプトウィンドウを開き、再度コマンドを実行します。

「必須オプション -O がない」というようなエラーが発生した場合は、AWS_ACCESS_KEY の設定を確認し、誤りを修正して、新しいコマンドプロンプトウィンドウを開き、再度コマンドを実行します。

「-W がない」というようなエラーが発生した場合は、AWS_SECRET_KEY の設定を確認し、誤りを修正して、新しいコマンドプロンプトウィンドウを開き、再度コマンドを実行します。

タスク 5: リージョンを設定する (オプション)

Amazon EC2 CLI ツールは、デフォルトではサービスエンドポイント URL ec2.us-east-1.amazonaws.com で指定された us-east-1 リージョンを使用します。お使いのインスタンスが別のリージョンにある場合は、その場所を指定する必要があります。例えばインスタンスが欧州にある場合は、--region eu-west-1 オプションを使用するか、EC2_URL 環境変数を設定することで、eu-west-1 リージョンを指定します。

このセクションでは、サービスエンドポイント URL を変更することで別のリージョンを指定する方法を説明します。

別のリージョンを指定するには(コンピュータまたはインスタンス)

1. 使用可能なリージョンは、「*Amazon Web Services General Reference*」の「Regions and Endpoints」を参照してください。
2. サービスエンドポイントを変更するには、EC2_URL 環境変数を設定します。

   次の例では、EC2_URL を設定します。

   a. [Start] ボタンをクリックして [Computer] を右クリックし、[Properties] をクリックします。
   b. [”Advanced system setting”]をクリックします。
   c. [”Environment Variables”]をクリックします。

d. [”System variables”]の下にある[”New”]をクリックします。
e. [Variable name] に **EC2_URL** と入力します。
f. [Variable value] に **https://ec2 eu-west-1.amazonaws.com**.

タスク 6: プロキシを使用する (オプション)

CLI ツールをインストールしたコンピューターがプロキシサーバーの使用を必要とする場合、EC2_JVM_ARGS 環境変数でプロキシサーバーを使用するために CLI ツールに通知する必要があります。

次の表に、EC2_JVM_ARGS 変数に対して設定できるプロキシ設定のプロパティを示します。必要なプロパティは、使用しているプロキシサーバーの種類によって異なります。たとえば、http.proxyDomain と http.proxyWorkstation のプロパティは Windows NTLM プロキシでのみ使われます。

プロパティ	説明
https.proxyHost	HTTPS プロキシホスト。EC2_URL が HTTPS ホストを指定した場合に使用します。
https.proxyPort	HTTPS プロキシポート。EC2_URL が HTTPS ホストを指定した場合に使用します。
http.proxyHost	HTTP プロキシホスト。EC2_URL が HTTP ホストを指定した場合に使用します。
http.proxyPort	HTTP プロキシポート。EC2_URL が HTTP ホストを指定した場合に使用します。
http.proxyDomain	プロキシドメイン (HTTPS および HTTP)
http.proxyWorkstation	プロキシワークステーション (HTTPS および HTTP)
http.proxyUser	プロキシユーザー名 (HTTPS および HTTP)
http.proxyPass	プロキシパスワード (HTTPS および HTTP)

コンピュータまたはインスタンスの EC2_JVM_ARGS 環境変数を設定するには

1. [Start] ボタンをクリックして [Computer] を右クリックし、[Properties] をクリックします。
2. [Advanced system settings] をクリックします。
3. [Environment Variables] をクリックします。
4. [System variables] の下にある [New] をクリックします。
5. [Variable name] に **EC2_JVM_ARGS** と入力します。
6. [Variable value] に、プロキシ設定プロパティを指定します。例:
   -Dhttps.proxyHost=my.proxy.com -Dhttps.proxyPort=8080 。

タスク 7: リモートデスクトップをダウンロードする

Windows インスタンスに接続するには、リモートデスクトップクライアントが必要です。最新バージョンのほとんどの Windows には、リモートデスクトップクライアントがすでに含まれます。リモートデスクトップクライアントがあるかどうかを確認するには、コマンドプロンプトウィンドウを開き、mstsc と入力します。このコマンドでリモートデスクトップ接続ウィンドウが表示された場合、クライアントは設定されています。表示されない場合は、 Microsoft Windows のホームページにアクセスし、リモートデスクトップ接続のダウンロードを検索してください。

これで、コマンドプロンプトウィンドウから Amazon EC2 を使用できるようになりました。

AWS Diagnostics for Microsoft Windows Server

AWS Diagnostics for Microsoft Windows Server は使いやすくシンプルなツールで、Amazon EC2 Windows Server インスタンスで実行して、起こりうる問題を診断してトラブルシューティングするためのものです。ログファイルを収集して問題を解決するだけでなく、問題がありそうな部分をプロアクティブに検索することができる、たいへん便利なツールです。例えば、アプリケーションに影響を与える可能性のある Windows ファイアウォールと AWS セキュリティグループ間の設定の問題を診断するために使用できます。他のインスタンスから EBS ブートボリュームを調べて、そのボリューム上の Windows Server インスタンスをトラブルシューティングするために必要なログを収集することもできます。

ユースケースとして、Key Management Service（KMS）アクティベーションの問題の診断が挙げられます。DNS サーバーを変更し、ドメインにインスタンスを追加した場合、またはサーバー時間が同期していない場合、KMS アクティベーションが失敗することがあります。この場合、自分で構成設定を調べて問題をデバッグするのではなく、AWS Diagnostics for Microsoft Windows Server ツールを実行して、問題解決に必要な情報を入手します。

別のユースケースとしては、Amazon EC2 セキュリティグループと Windows ファイアウォール間のルールの不一致の問題があります。セキュリティグループに関する AWS ユーザー認証情報を提供すると、AWS Diagnostics for Microsoft Windows Server ツールは、セキュリティグループにリストされたポートが Windows ファイアウォールを通過できるように許可されているかどうかを検証できます。自分でファイアウォールルールをセキュリティグループルールとつき合わせて調べる必要がなくなります。

AWS Diagnostics for Microsoft Windows Server ツールは無料で利用でき、https://s3.amazonaws.com/ec2-downloads-windows/AWSDiagnostics/AWSDiagnostics.zip からダウンロードしてインストールできます。

AWS Diagnostics for Microsoft Windows Server は 2 種類のモジュールで構成されます。データ収集モジュールはさまざまなソースからデータを収集し、データ分析モジュールは収集されたデータを、一連の定義済みルールと照らし合わせて解析し、問題を識別して、解決方法を提案します。

AWS Diagnostics for Microsoft Windows Server ツールは、EC2 インスタンスで実行する Windows Server でのみ利用できます。ツールを起動すると、EC2 インスタンスで実行しているかどうかをツールが確認します。確認に失敗すると、メッセージボックスに「`EC2InstanceCheckFailed`」エラーメッセージが表示されます。

分析ルール

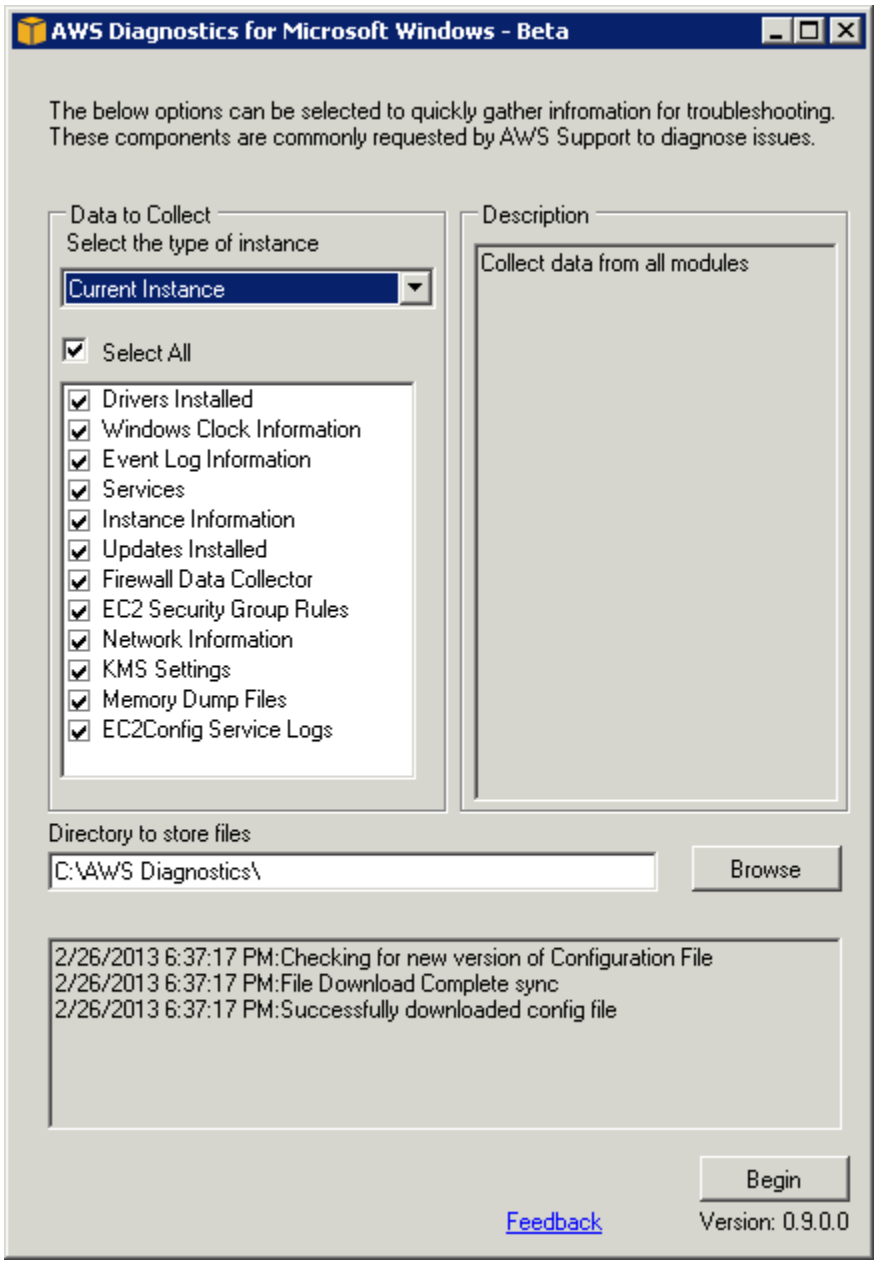
AWS Diagnostics for Microsoft Windows Server は次の分析ルールが用意されています。

- アクティベーションステータスと KMS 設定のチェック
- メタデータと KMS アクセスに対する適切なルートテーブルエントリーのチェック
- Amazon EC2 セキュリティグループルールと Windows ファイアウォールルールの比較
- PV ドライバ (RedHat または Citrix) のバージョンチェック
- `RealTimeIsUniversal`レジストリキーが設定されているかどうかのチェック
- 複数の NIC を使用している場合のデフォルトゲートウェイ設定のチェック
- ミニダンプファイルのコードのバグチェック

問題が見つからなかった場合でも、ツールによって収集されたデータが役立つ場合があります。ツールによって作成されたデータファイルを調べて問題がないか確認したり、ファイルを AWS プレミアムサポートに送信してサポートケースの解決を助けることができます。

現在のインスタンスの分析

現在のインスタンスを分析するには、AWS Diagnostics for Microsoft Windows Server ツールを実行して、[Current Instance] をインスタンスの種類に選択します。メインウィンドウの [Data to Collect] セクションで、AWS Diagnostics for Microsoft Windows Server が収集するデータを指定します。



データ	説明
インストールされたドライバ	インスタンスにインストールされているすべてのドライバに関する情報を収集します。
Windows クロック情報	インスタンスの現在時刻とタイムゾーンの情報を収集します。
イベントログ情報	イベントログから、メッセージ (重大、エラー、警告) を収集します。
サービス	インスタンスにインストールされているサービスに関する情報を収集します。

データ	説明
インスタンス情報	メタデータとローカル環境変数から情報を収集します。
インストールされた更新ファイル	インスタンスにインストールされている更新ファイルに関する情報を収集します。
ファイアウォールデータ	Windows ファイアウォール設定に関する情報を収集します。
EC2 セキュリティグループルール	インスタンスに関連付けられた Amazon EC2 セキュリティグループのルールに関する情報を収集します。
ネットワーク情報	インスタンスのルートテーブルと IP アドレス情報を収集します。
KMS 設定	Key Management Service 設定を収集します。
メモリダンプファイル	インスタンスに存在するメモリダンプファイルを収集します。
EC2Config サービスログ	EC2Config サービスによって生成されたログファイルを収集します。

オフラインインスタンスからのデータ収集

[Offline Instance] オプションは、EC2 Windows Server インスタンスを起動できない、またはインスタンスが正常に機能しないために AWS Diagnostics for Microsoft Windows Server を実行できない場合に、そのインスタンスの問題をデバッグする場合に便利です。このような場合、EBS ブートボリュームをそのインスタンスからデタッチし、別の EC2 Windows Server インスタンスにアタッチします。

オフラインインスタンからデータを収集するには

1. 問題のあるインスタンスを、まだ停止していない場合は停止します。
2. EBS ブートボリュームを問題のあるインスタンスからデタッチします。
3. EBS ブートボリュームを、AWS Diagnostics for Microsoft Windows Server がインストールされている別の正常な Windows Server インスタンスにアタッチします。
4. 正常なインスタンスでボリュームをマウントしてドライブ文字を割り当てます (例: 「F:」) 。
5. 正常なインスタンスで AWS Diagnostics for Microsoft Windows Server ツールを実行して、[Offline Instance] オプションを選択します。
6. さきほどマウントしたボリュームのドライブ文字 (例: 「F:」) を選択します。
7. [Begin] をクリックします。

AWS Diagnostics for Microsoft Windows Server ツールはボリュームをスキャンして、ボリューム上のログファイルに基づいてトラブルシューティング情報を収集します。オフラインインスタンスの場合、収集するデータは固定です。また、データの分析は行われません。

データファイルストレージ

AWS Diagnostics for Microsoft Windows Server ツールはデフォルトで、ツールを起動したディレクトリにデータファイルを保存します。AWS Diagnostics for Microsoft Windows Server ツールが収集した

データファイルを保存する場所を変更することができます。選択したディレクトリに、「DataCollected」という名前のディレクトリが作成されます (まだ作成されていない場合) 。アプリケーションを実行するたびに、現在の日付と時刻が名前に付いたディレクトリが個別に作成されます。各データ収集モジュールは、そのデータセットの情報を含む XML ファイルを作成します。生成されたすべてのデータファイルのコピーを含む ZIP ファイルアーカイブも作成されます。このアーカイブは、必要に応じて AWS プレミアムサポートのエンジニアに提供することができます。

Windows AMI での PV ドライバのアップグレード

Amazon Windows AMI には、Xen 仮想ハードウェアにアクセスできるようにするためのドライバー式が含まれています。Amazon EC2 は、これらのドライバを使用して、インスタンスストアと Amazon Elastic Block Store (Amazon EBS) ボリュームをデバイスにマップします。

Windows AMI が RedHat ドライバを使用している場合、Citrix ドライバにアップグレードすることができます。すでに Citrix ドライバを使用している場合は、Citrix paravirtualized (PV) ゲストエージェントドライバにアップグレードできます。

Topics

Windows Server 2003 インスタンスのアップグレード

このセクションでは、Windows Server 2003 インスタンスで RedHat ドライバを Citrix ドライバにアップグレードする方法を説明します。

ドライバをアップグレードする前に、次のことを実行してください。

- インスタンスにある重要な情報のバックアップを作成するか、インスタンスから AMI を作成します。AMI の作成の詳細については、Amazon EBS-Backed Windows AMI の作成 (p. 50)を参照してください。AMI を作成する場合は、事前に次のことを実行してください。
  - EC2Config サービスで Sysprep ツールを有効にしていないことを確認する。
  - パスワードを書き留める。
  - イーサネットアダプタを DHCP に設定する。
- Amazon Windows EC2Config サービス にアクセスして最新バージョンの EC2Config をインストールします。EC2Config サービスの詳細については、EC2Config サービスを使用した Windows インスタンスの設定 (p. 39) を参照してください。

Windows Server 2003 AMI をアップグレードするには

1. インスタンスに接続してローカル管理者としてログインします。インスタンスへの接続の詳細については、「Windows インスタンスへの接続」を参照してください。
2. インスタンスから、Amazon EC2 Windows Paravirtual Driver Upgrade Script にアクセスして Citrix アップブレードパッケージをダウンロードします。
3. アップグレードパッケージを好きな場所に抽出します。
4. Upgrade.bat ファイルをダブルクリックします。セキュリティ警告が表示された場合は、[実行] をクリックします。
5. [Upgrade Drivers] ダイアログボックスの情報を確認して、アップグレードを開始する場合は、[Yes] をクリックします。
6. [Red Hat Paravirtualized Xen Drivers for Windows (R) uninstaller] ダイアログボックスで [Yes] をクリックして RedHat ソフトウェアを削除します。インスタンスが再起動します。

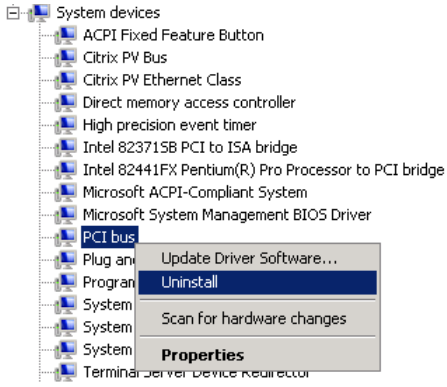
   

   Note

   アンインストーラのダイアログボックスが表示されない場合は、Windows タスクバーの [Red Hat Paravirtualiz...] をクリックします。

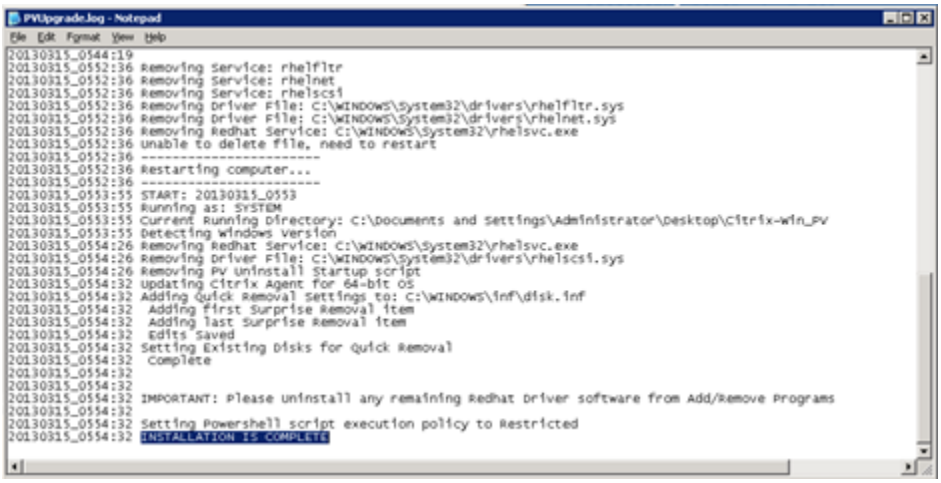
7. インスタンスが再起動して使用できる状態にあることを確認します。
   a. EC2 コンソールを開きます。
   b. [Instances] でインスタンスを右クリックし、[Get System Log] を選択します。
   c. ログメッセージの最後を確認します。「`Windows is Ready to use`」と表示されているはずです。
8. インスタンスに接続してローカル管理者としてログインします。PowerShell、RedHat アンインストーラ、PVUpgrade.log、Windows のデバイス マネージャという 4 つのアプリケーションを開いてアップグレード処理を続けます。
9. PCI BUS をアンインストールします。
   a. [Device Manager] ウィンドウで、[System devices] を展開し、[PCI bus] を右クリックして [Uninstall] を選択します。

   b. [Confirm Device Removal] ダイアログボックスで [OK] をクリックします。
   c. すぐにインスタンスを再起動しないので、[System Settings Change] ダイアログボックスで [No] をクリックします。

d. Device Managerを閉じます。アップグレードスクリプトによって、インスタンスが再起動します。

10. ステップ 7 の手順を繰り返して、インスタンスが利用できる状態であることを確認します。確認できたら、管理者としてログインします。
11. インストールが完了したことを確認します。先ほどファイルを抽出した Citrix-WIN_PV フォルダに移動して、PVUpgrade.log ファイルを開き、「INSTALLATION IS COMPLETE」という文字列を確認します。



Windows Server 2008 および Windows Server 2008 R2 インスタンスのアップグレード

このセクションでは、Windows Server 2008 または Windows Server 2008 R2 インスタンスで RedHat ドライバを Citrix ドライバにアップグレードする方法を説明します。

ドライバをアップグレードする前に、次のことを実行してください。

- Amazon Windows EC2Config サービス にアクセスして最新バージョンの EC2Config をインストールします。EC2Config サービスの詳細については、EC2Config サービスを使用した Windows インスタンスの設定 (p. 39) を参照してください。
- インスタンスにある重要な情報のバックアップを作成するか、インスタンスから AMI を作成します。AMI の作成の詳細については、Amazon EBS-Backed Windows AMI の作成 (p. 50)を参照してください。AMI を作成する場合は、事前に次のことを実行してください。
  - EC2Config サービスで Sysprep ツールを有効にしていないことを確認する。
  - パスワードを書き留める。
  - イーサネットアダプタを DHCP に設定する。

Windows Server 2008 または Windows Server 2008 R2 AMI をアップグレードするには

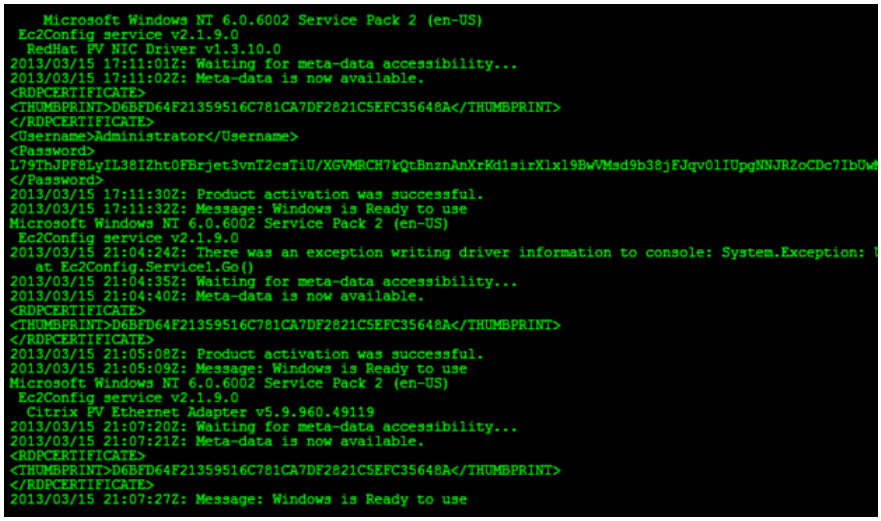
1. インスタンスに接続してローカル管理者としてログインします。インスタンスへの接続の詳細については、「Windows インスタンスへの接続」を参照してください。
2. インスタンスから、Amazon EC2 Windows Paravirtual Driver Upgrade Script にアクセスして Citrix アップブレードパッケージをダウンロードします。

3. アップグレードパッケージを好きな場所に抽出します。
4. Upgrade.bat ファイルをダブルクリックします。セキュリティ警告が表示された場合は、[Run] をクリックします。
5. [Upgrade Drivers] ダイアログボックスの内容を確認して、アップグレードを開始する場合は、[Yes] をクリックします。
6. [Red Hat Paravirtualized Xen Drivers for Windows (R) uninstaller] ダイアログボックスで [Yes] をクリックして RedHat ソフトウェアを削除します。インスタンスが再起動します。

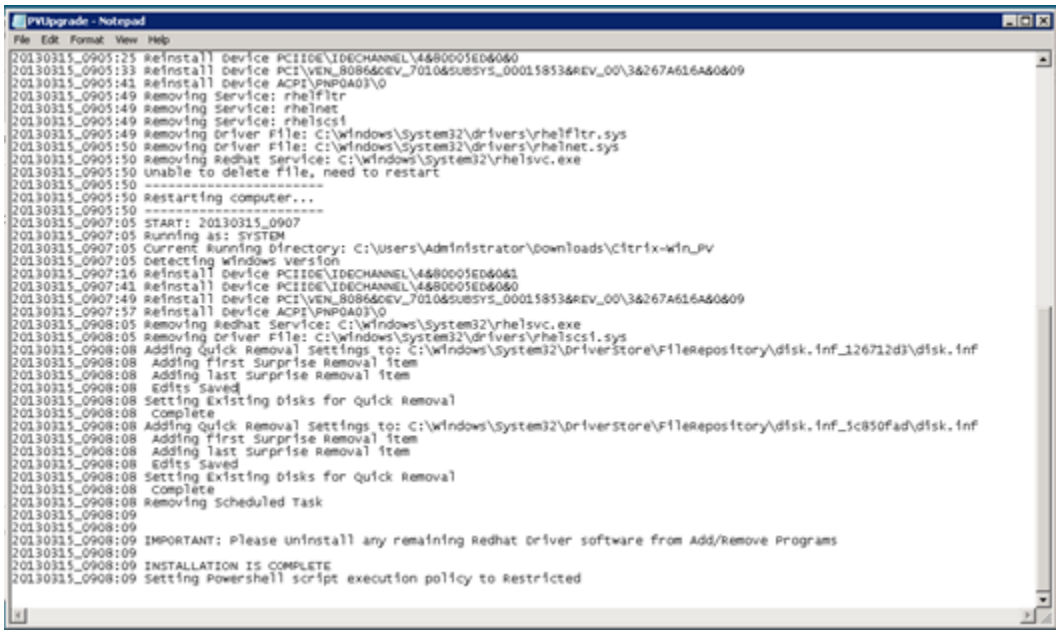
   Note

   アンインストーラのダイアログボックスが表示されない場合は、Windows タスクバーの [Red Hat Paravirtualiz…] をクリックします。

7. インスタンスが再起動して使用できる状態にあることを確認します。
   a. EC2 コンソールを開きます。
   b. [Instances] でインスタンスを右クリックし、[Get System Log] を選択します。
   c. アップグレード操作では、サーバーが 3 ~ 4 回再起動します。何度再起動されたかは、ログで「`Windows is Ready to use`」が表示された回数で確認できます。



8. インスタンスに接続してローカル管理者としてログインします。
9. [Red Hat Paravirtualized Xen Drivers for Windows (R) uninstaller] ダイアログボックスを閉じます。
10. インストールが完了したことを確認します。先ほどファイルを抽出した Citrix-WIN_PV フォルダに移動して、PVUpgrade.log ファイルを開き、「`INSTALLATION IS COMPLETE`」という文字列を確認します。



Citrix PV ゲストエージェントドライバのアップグレード

Windows サーバーで Citrix ドライバを使用している場合、Citrix PV ゲストエージェントドライバにアップグレードできます。このドライバは Windows サービスの 1 つとして実行し、起動時の時刻の同期や、API からのシャットダウンイベントや再起動イベントなどのタスクを処理します。このアップグレードパッケージは Windows Server 2012 を含む Windows Server のすべてのバージョンで実行できます。

ドライバのアップグレードを始める前に、インスタンスにある重要な情報のバックアップを作成するか、インスタンスから AMI を作成します。AMI の作成の詳細については、Amazon EBS-Backed Windows AMI の作成 (p. 50)を参照してください。AMI を作成する場合は、事前に次のことを実行してください。

- EC2Config サービスで Sysprep ツールを有効にしていないことを確認する。
- パスワードを書き留める。
- イーサネットアダプタを DHCP に設定する。

Citrix PV ゲストエージェントドライバをアップグレードするには

1. インスタンスに接続してローカル管理者としてログインします。インスタンスへの接続の詳細については、「Windows インスタンスへの接続」を参照してください。
2. インスタンスから、Amazon EC2 Windows Paravirtual Driver Upgrade Script にアクセスして Citrix アップブレードパッケージをダウンロードします。
3. アップグレードパッケージを好きな場所に抽出します。
4. Upgrade.bat ファイルをダブルクリックします。セキュリティ警告が表示された場合は、[実行] をクリックします。
5. [Upgrade Drivers] ダイアログボックスの内容を確認して、アップグレードを開始する場合は、[Yes] をクリックします。

6. アップグレードが完了すると、PVUpgrade.log ファイルが開きます。「UPGRADE IS COMPLETE」という文字列が含まれているはずです。
7. インスタンスを再起動します。

ドキュメント履歴

次の表に、*Amazon Elastic Compute Cloud Microsoft Windows Guide* に追加する重要項目を示します。このガイドのアップデートにより、お客様からいただいたフィードバックも反映されました。

変更点	説明	リリース日
AWS マネジメントパック用に新しいセクションが追加されました。	AWS マネジメントパックは Amazon EC2 インスタンスと、インスタンスで実行している Microsoft Windows または Linux オペレーティングシステムをリンクします。AWS マネジメントパックは Microsoft System Center Operations Manager 向けの拡張パックです。詳細については、「Microsoft System Center Operations Manager 向け AWS マネジメントパック (p. 63)」を参照してください。	2013 年 5 月 8 日
内容の追加	トピック「Windows AMI での PV ドライバのアップグレード (p. 119)」では Windows AMI. 内の paravirtualized (PV) ドライバをアップグレードする方法を説明します。	2013 年 3 月
内容の追加	トピック「AWS Diagnostics for Microsoft Windows Server (p. 114)」では、AWS Diagnostics for Microsoft Windows Server を使用した診断方法やトラブルシューティング方法を説明します。	2013 年 3 月
内容の追加	トピック「&EC2 Windows インスタンスの使用開始 (p. 8)」では、最初の Windows インスタンスを起動してそのインスタンスに接続する方法を詳しく説明します。トピック「Amazon EC2 のリソースに対するアクセスの制御 (p. 28)」では、インスタンスへのアクセスをコントロールする方法の概要を説明します。トピック「&EC2 インスタンスでの WordPress ブログのデプロイ (p. 19)」では、WordPress ブログを Amazon EC2 インスタンス上で作成してデプロイする方法を説明します。	2011 年 12 月
内容の追加	トピック「Amazon EC2 での Windows HPC クラスターのセットアップ (p. 97)」では、Windows HPC クラスタを Amazon Elastic Compute Cloud 上で設定する方法を説明します。	2011 年 11 月

変更点	説明	リリース日
	このガイドには、Amazon EC2 Windows インスタンスの使用に関する情報が記載されています。Windows インスタンスの基本的なインフラストラクチャコンポーネントについては、「Amazon EC2 とは (p. 3)」を参照してください。Windows AMI の使用については、「Windows Amazon マシンイメージ (AMI) (p. 34)」を参照してください。コマンドラインインターフェイスのセットアップ方法については、「Windows への Amazon EC2 コマンドラインインターフェイスツールのインストール (p. 107)」を参照してください。	2011 年 9 月