
Amazon CloudWatch

入門ガイド

API Version 2013-09-06



Amazon Web Services

Amazon CloudWatch: 入門ガイド

Amazon Web Services

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 CloudWatch の使用を開始する	1
CloudWatch にサインアップする	2
コマンドラインインターフェイスをセットアップする	3
メトリックスを CloudWatch にパブリッシュする	7
CloudWatch から統計情報を取得する	10
コンソールでグラフを表示する	11
次のステップ	14
CloudWatch にアクセスするその他の方法	14
引き続きコマンドラインインターフェイスを使用する	15
コンソールを使用する	15
既存のライブラリを使用する	15
コードで直接ウェブサービス API を呼び出す	15
CloudWatch に関するその他の情報	15
EC2 インスタンスのモニタリング	15
アラームを作成する	16
フィードバックをお寄せください	17
文書履歴	18
本ガイドについて	19

Amazon CloudWatch の使用を開始する

Amazon CloudWatch は、メトリックスの収集、表示、分析のためのウェブサービスです。Amazon CloudWatch の使用を開始するには、次の図に示すタスクを実行します。CloudWatch コマンドラインインターフェイス (CLI) と AWS マネジメントコンソール (ポイントアンドクリック方式のウェブベースインターフェイス) の両方を使用します。



このガイドでは、例えば CLI のセットアップ、メトリックスのパブリッシュ、統計情報の取得、コンソールでのグラフの表示など、Amazon CloudWatch の基本的な手順を詳しく説明します。

CloudWatch にサインアップする



すでに AWS アカウントをお持ちの場合は、次の手順に進んでください。まだ AWS アカウントをお持ちでない場合は、ここで説明する手順に従ってアカウントを作成してください。



Note

アカウントを作成すると、そのアカウントで AWS のすべてのサービスへのサインアップが自動的に行われます。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。

サインアップして 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.

これで、コマンドラインツールをダウンロードできる状態になりました。

コマンドラインインターフェイスを セットアップする



Topics

- [コマンドラインツールをインストールする \(p. 3\)](#)
- [認証情報に対して CLI を設定する \(p. 5\)](#)
- [設定をテストする \(p. 5\)](#)

コマンドラインツールをインストールする

このガイドで説明するコマンドを使用するには、コマンドラインツールをインストールして設定する必要があります。

Amazon CloudWatch コマンドラインツールをインストールするには

1. [Amazon CloudWatch ツールのページ](#)からコマンドラインツールをダウンロードします。
2. CloudWatch アーカイブを、ワークステーション上の任意の場所に保存して圧縮を解除します。
3. コマンド `java -version` を入力して、ワークステーションにインストールされている Java のバージョンを特定します。1.5 以降のバージョンがインストールされていない場合は、新しいバージョンを <http://www.java.com> からダウンロードしてインストールしてください。
4. `JAVA_HOME` 環境変数を、Java のインストール先を指すように設定します。

次に示すのは、Linux/UNIX でこの環境変数を設定する方法の例です。

```
$ export JAVA_HOME=<PATH>
```

次に示すのは、Windows での構文の例です。

```
C:\> set JAVA_HOME=<PATH>
```

5. 次のとおりに PATH 環境変数に追加します。

Linux/UNIX では、PATH を更新する方法は次のとおりです。

```
$ export PATH=$PATH:$JAVA_HOME/bin
```



Note

export コマンドが適用されるのは、現在のシェルセッションのみです。環境変数を永続的に作成または更新するには、このコマンドを起動スクリプトに入れてください。例えば、Bash シェルを使用する場合は、コマンドを ~/.bashrc または /etc/profile ファイルに含めることができます。

Windows の場合、構文はわずかに異なります。

```
C:\> set PATH=%PATH%;%JAVA_HOME%\bin
```



Note

Windows の環境変数は、コマンドウィンドウを閉じたときにリセットされます。永続的に設定するには、setx コマンドを使用します。

6. AWS_CLOUDWATCH_HOME 環境変数を、Amazon CloudWatch アーカイブの圧縮を解除した場所を指すように設定します。

Linux/UNIX では、この環境変数を設定する方法は次のとおりです。

```
$ export AWS_CLOUDWATCH_HOME=<path-to-tools>
```

Windows の場合、構文はわずかに異なります。

```
C:\> set AWS_CLOUDWATCH_HOME=<path-to-tools>
```

7. 次のとおりに PATH 環境変数に追加します。

Linux/UNIX では、PATH を更新する方法は次のとおりです。

```
$ export PATH=$PATH:$AWS_CLOUDWATCH_HOME/bin
```

Windows の場合、構文はわずかに異なります。

```
C:\> set PATH=%PATH%;%AWS_CLOUDWATCH_HOME%\bin
```

認証情報に対して CLI を設定する

CloudWatch の CLI を使用する前に、AWS の認証情報を提供する必要があります。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-credential-file` パラメータを使って証明書を指定することもできますし、またはローカルシステム上の証明書をポイントする環境変数を設定しておくこともできます。環境変数が正しく設定されていれば、コマンドを発行するたびに `--aws-credential-file` パラメータを指定する必要はありません。以下の手順では、証明書ファイルおよび対応する `AWS_CREDENTIAL_FILE` 環境変数を作成する方法を説明しています。

CLI に対してセキュリティ認証情報を設定するには

1. アクセスキー ID とシークレットアクセスキーを、CloudWatch の CLI をインストールしたフォルダにある `credential-file-path.template` という名前のファイルに追加します。
2. ファイル名を変更して、ワークステーション上の任意の場所に保存します。

- Linux/UNIX を使用している場合は、次のとおりにファイルのアクセス許可を設定します。

```
$ chmod 600 <credential file name>
```

- Windows では、ファイルのアクセス許可を変更する必要はありません。

3. `AWS_CREDENTIAL_FILE` 環境変数を、前のステップで作成した認証情報ファイルの完全修飾パスに設定します。

- Linux/UNIX では、この環境変数を設定する方法は次のとおりです。

```
$ export AWS_CREDENTIAL_FILE=<path-to-file>
```

- Windows の場合、この環境変数は以下のように設定します。

```
C:\> set AWS_CREDENTIAL_FILE=<path-to-file>
```

設定をテストする

Amazon CloudWatch のインストールと設定をテストするには

1. ワークステーションで、新しいコマンドプロンプトを開きます。
2. コマンド `mon-cmd` を入力します。
3. 次のような出力が表示されます。

Command Name	Description
-----	-----
mon-delete-alarms	Delete alarms
mon-describe-alarm-history	Describe alarm history
mon-describe-alarms	Describe alarms fully.
mon-describe-alarms-for-metric	Describe all alarms ... a single metric
mon-disable-alarm-actions	Disable all actions for a given alarm
mon-enable-alarm-actions	Enable all actions for a given alarm
mon-get-stats	Get metric statistics
mon-list-metrics	List user's metrics
mon-put-data	Put metric data
mon-put-metric-alarm	Create new alarm or update existing one
mon-set-alarm-state	Manually set the state of an alarm
mon-version	Prints the version ... tool and the API

For help on a specific command, type '<commandname> --help'

これで、Amazon CloudWatch コマンドラインツールのインストールと設定は完了です。

メトリックスを CloudWatch にパブリッシュする



Topics

- [データ構成を定義する \(p. 7\)](#)
- [メトリックスを Amazon CloudWatch に追加する \(p. 8\)](#)

Amazon CloudWatch CLI のインストールが完了したので、メトリックスを CloudWatch にパブリッシュできる状態になりました。ここでは、コマンドラインツールを使用して、*GetStarted* という架空のアプリケーションのメトリックスを 1 つだけパブリッシュします。

データ構成を定義する

次に示す例では、アプリケーションのリクエストレイテンシーをトラッキングするデータポイントをパブリッシュします。メトリックスの名前と名前空間は、わかりやすいものを選択します。この例では、メトリックスに *RequestLatency* という名前を付けて、すべてのデータポイントを *GetStarted* という名前空間に入れます。

パブリッシュするデータポイントは多数あります。これらの集合が、3 時間分のレイテンシーデータを表します。生データは、3 時間にわたって読み取られた 15 個のリクエストレイテンシー読み取り値で構成されています。読み取り値の単位はミリ秒です。

- 1 時間目: 87、51、125、235
- 2 時間目: 121、113、189、65、89
- 3 時間目: 100、47、133、98、100、328

データを CloudWatch にパブリッシュするときは、単一のデータポイントとしてパブリッシュすることも、複数のデータポイントを集約したセット (統計セット) としてパブリッシュすることもできます。ここでは、1 時間目のデータポイントをそれぞれ単一データポイントとしてパブリッシュします。

時間	生データ
1	87
1	51
1	125
1	235

2 時間目と 3 時間目のデータについては、データポイントを時間ごとに集約し、統計セットとしてパブリッシュします。



Note

メトリックス集約の最小単位は 1 分間です。

データポイントを集約して統計セットとして CloudWatch にパブリッシュするときに、4 つの事前定義キー (Sum、Minimum、Maximum、SampleCount) を指定できます。キーの値を次の表に示します。

時間	生データ	Sum	Minimum	Maximum	SampleCount
2	121, 113, 189, 65, 89	577	65	189	5
3	100, 47, 133, 98, 100, 328	806	47	328	6

メトリックスを Amazon CloudWatch に追加する

データ構成の定義が完了すると、データの追加を開始できる状態になります。



Note

`mon-put-data` コマンドを使用するときは、過去 2 週間以内の日付範囲を使用する必要があります。現時点では、データポイントを削除する機能はありません。CloudWatch によって自動的に、`timestamp` から 2 週間以上経過しているデータポイントが削除されます。

`mon-put-data` コマンドのパラメータの多くに短縮形があり、すばやくデータを入力できるようになっています。次の表は、関係する `mon-put-data` のパラメータの短縮形をまとめたものです。これらの短縮形を使用すると、データ入力が簡単になります。

パラメータ	短縮形
<code>--metric-name</code>	<code>-m</code>
<code>--namespace</code>	<code>-n</code>
<code>--timestamp</code>	<code>-t</code>

パラメータ	短縮形
<code>--statisticValues</code>	<code>-s</code>
<code>--unit</code>	<code>-u</code>
<code>--value</code>	<code>-v</code>

データポイントを CloudWatch にパブリッシュするには

1. コマンドプロンプトを開いて、次に示すコマンドを入力します。ただし、タイムスタンプは、2時間前の時刻を UTC (Universal Coordinated Time) で表したタイムスタンプで置き換えてください。

```
mon-put-data -m RequestLatency -n "GetStarted" -t 2010-10-29T20:30:00Z -v 87 -u Milliseconds  
mon-put-data -m RequestLatency -n "GetStarted" -t 2010-10-29T20:30:00Z -v 51 -u Milliseconds  
mon-put-data -m RequestLatency -n "GetStarted" -t 2010-10-29T20:30:00Z -v 125 -u Milliseconds  
mon-put-data -m RequestLatency -n "GetStarted" -t 2010-10-29T20:30:00Z -v 235 -u Milliseconds
```

CloudWatch から応答が返されるのは、コマンドを実行できなかったときだけです。

2. 2番目のデータポイントを入力します。ただし、今回のタイムスタンプは、最初のタイムスタンプの1時間後を UTC (Universal Coordinated Time) で表したものを使用します。

```
mon-put-data -m RequestLatency -n "GetStarted" -t 2010-10-29T21:30:00Z -s "Sum=577,Minimum=65,Maximum=189,SampleCount=5" -u Milliseconds
```

3. 最後のデータポイントを入力します。ただし、今回はタイムスタンプを省略します。デフォルトである現在の時刻が使用されるようにするためです。

```
mon-put-data -m RequestLatency -n "GetStarted" -s "Sum=806,Minimum=47,Maximum=328,SampleCount=6" -u Milliseconds
```

`mon-put-data` コマンドには、他にも多数のオプションがあります。詳細については、『[Amazon CloudWatch Developer Guide](#)』の「`mon-put-data`」を参照してください。

メトリックスを追加した後は、統計情報を取得できます。

CloudWatch から統計情報を取得する



Amazon CloudWatch へのメトリックスのパブリッシュが完了したので、そのメトリックスに基づく統計情報を取得できる状態になりました。次に示す例では、`--metric-name`、`--namespace`、`--statistics` だけが必須のパラメータです。

コマンドラインツールを使用して統計情報を取得するには

- `--start-time` には、過去の時刻を指定します。パブリッシュしたタイムスタンプのうち最も早いものよりも前の時刻を指定してください。`--end-time` パラメータは省略できます。このパラメータのデフォルト値は現在の時刻であるからです。`--headers` パラメータを指定すると、表の見出しが表示されます。

```
mon-get-stats -n GetStarted -m RequestLatency -s "Average" --start-time
2010-10-29T00:00:00Z --headers
```

CloudWatch からは、次の情報が返されます。

Time	Average	Unit
2010-10-29 20:30:00	124.5	Milliseconds
2010-10-29 21:30:00	115.4	Milliseconds
2010-10-29 22:17:00	134.33333333333334	Milliseconds

統計情報を図として見るには、AWS マネジメントコンソールを使用してグラフを表示します。

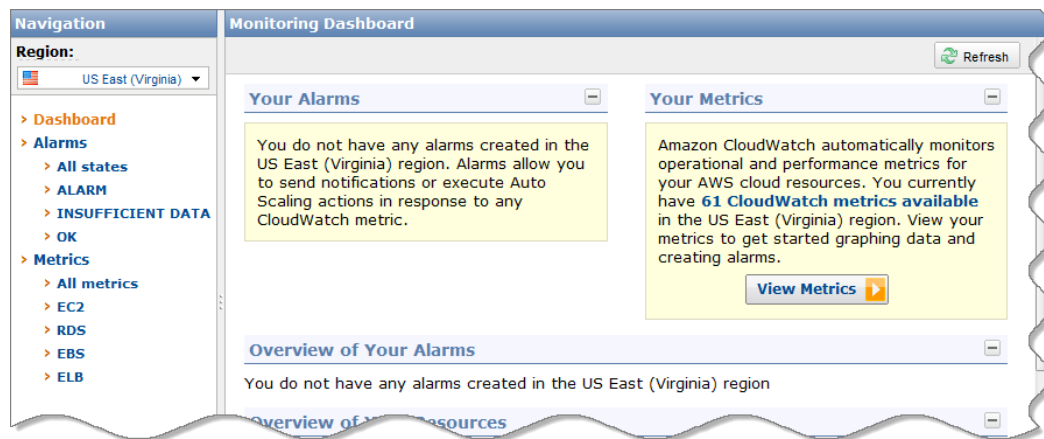
コンソールでグラフを表示する



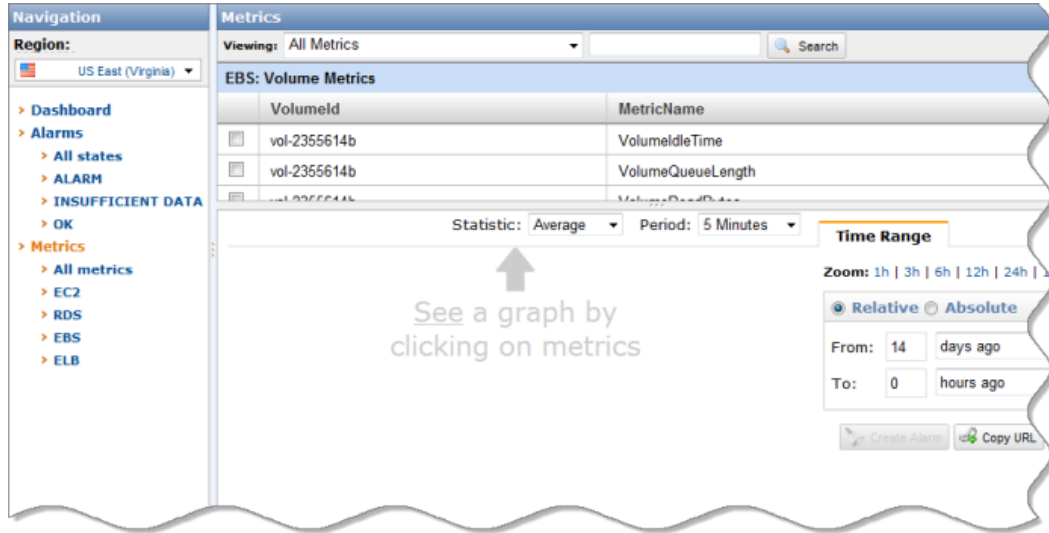
AWS マネジメントコンソールで統計グラフを見ることができます。

統計情報のグラフをコンソールで表示するには

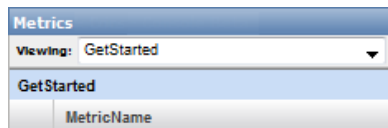
1. Sign in to the AWS Management Console and open the Amazon CloudWatch console at <https://console.aws.amazon.com/cloudwatch/>.



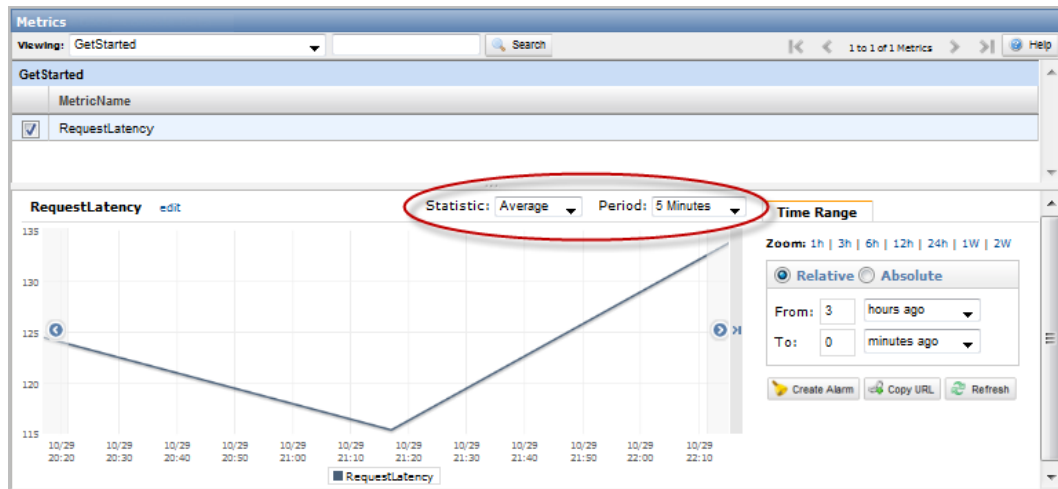
2. [Navigation] ペインの [Metrics] をクリックします。[Metrics] が右のペインで開きます。



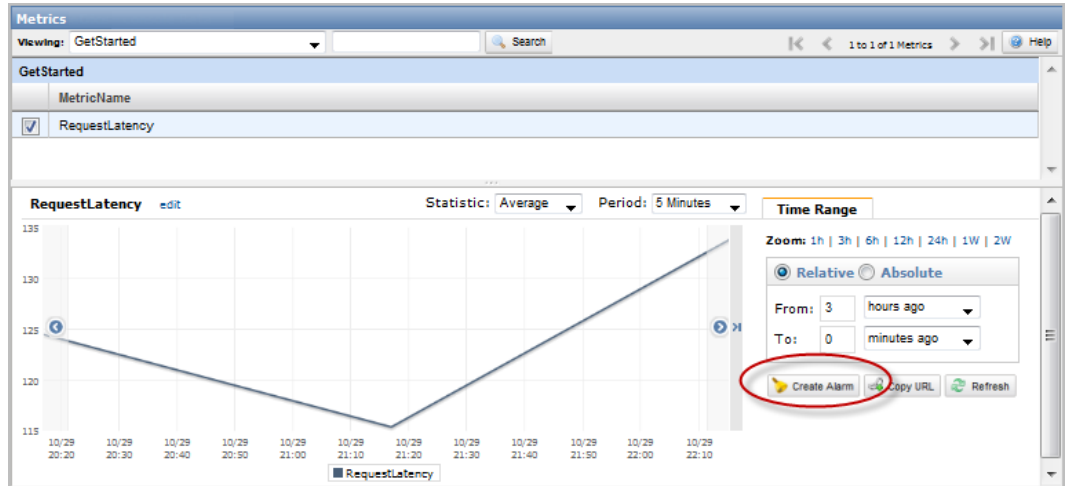
3. [Metrics] ペインで、[Viewing] ドロップダウンリストボックスから [GetStarted] を選択します。



4. [Metrics] ペインで、RequestLatency というメトリックス名の横のチェックボックスをオンにします。グラフの表示が更新されて、次の画面キャプチャのようになります。



5. グラフを変更するには、[Statistic] と [Period] のドロップダウンリストボックスで別の値を選択します。
6. このメトリックスに対するアラームを作成することもできます。作成するには、[Create Alarm] をクリックします。



次のステップ

Topics

- [AWS Account and Security Credentials \(p. 14\)](#)
- [CloudWatch にアクセスするその他の方法 \(p. 14\)](#)
- [CloudWatch に関するその他の情報 \(p. 15\)](#)
- [Amazon CloudWatch のリソース \(p. 16\)](#)

Amazon CloudWatch には、このガイドで取り上げたもの以外にも多数の機能があります。例えば、独自のアラームの作成や、Auto Scaling および Simple Notification Service (SNS) との統合です。このセクションでは、Amazon CloudWatch の理解を深め、さらに活用していただけるように、追加のリソースへのリンクを示します。

AWS Account and Security Credentials

So far you signed up for the service, got an AWS account and security credentials, and then completed a short exercise covering the essential product functions. Now that you're finished with the exercise, we recommend that you check with an administrator or coworker in your organization to determine if he or she already has an AWS account and security credentials for you to use in future interactions with AWS.

If you're an account owner or administrator and want to know more about AWS Identity and Access Management, go to the product description at <http://aws.amazon.com/iam> or to the technical documentation at [Using IAM](#).

CloudWatch にアクセスするその他の方法

この入門ガイドでは、CLI を使用してメトリックスをパブリッシュし、統計情報を取得する方法を説明しています。引き続き CloudWatch を CLI 経由で使用することも、他のインターフェイスを試すこともできます。

引き続きコマンドラインインターフェイスを使用する

コマンドラインインターフェイスには、メトリックスのパブリッシュと統計情報の取得の他に、アラーム作成の機能もあります。このコマンドラインツールを利用すると、API のコーディングやライブラリの使用は不要で、すぐに Amazon CloudWatch のすべての機能を実行できます。

コンソールを使用する

AWS マネジメントコンソールには、パブリッシュしたメトリックスに関する統計情報の表示の他にも多数の機能があります。CloudWatch をコンソール経由で使用する方法の詳細については、『[Amazon CloudWatch Developer Guide](#)』をご覧ください。このコンソールには便利なオンラインヘルプもあり、コンソールの [Help] ボタンをクリックするだけで参照できます。

既存のライブラリを使用する

CloudWatch をプログラムインターフェイス経由で使用できるように、次に示す言語のライブラリとソースが用意されています。

- [Java](#)
- [PHP](#)
- [Python](#)
- [Ruby](#)
- [Windows および .NET](#)

すべての言語のライブラリとサンプルコードについては、[Amazon EC2 のサンプルコード & ライブラリ](#)のページにアクセスしてください。

コードで直接ウェブサービス API を呼び出す

作成するコードの中で直接 Amazon CloudWatch Query API を呼び出す場合は、『[Amazon CloudWatch Developer Guide](#)』を参照してください。このガイドでは、API リクエストの作成と認証の方法、および CloudWatch を API 経由で使用する方法について説明します。すべての API アクションの説明については、『[Amazon CloudWatch API Reference](#)』を参照してください。

CloudWatch に関するその他の情報

このセクションでは、CloudWatch のその他の機能の一覧と、詳細情報を得られる場所を示します。

EC2 インスタンスのモニタリング

モニタリングは、EC2 インスタンスを作成した直後に有効化することも、後で有効化することもできます。モニタリングを有効化した後、しばらくすると、インスタンスに関する統計情報を CLI、Query API、または AWS マネジメントコンソールを使用して取得できるようになります。詳細については、『[Amazon CloudWatch Developer Guide](#)』を参照してください。

アラームを作成する

アラームを作成すると、EC2 インスタンスから収集された特定のメトリックスを監視することができます。メトリックスがしきい値を超えたときは、アラームによって電子メールメッセージを Simple Notification Service (SNS) 経由で送信することや、Auto Scaling アクティビティをトリガーすることができます。詳細については、『[Amazon CloudWatch Developer Guide](#)』を参照してください。

Amazon CloudWatch のリソース

Amazon CloudWatch を利用する際に役立つ関連リソースを次の表にまとめました。

リソース	説明
Amazon CloudWatch のよくある技術的な質問	Amazon RDS について開発者からよく寄せられる上位 20 の質問です。
リリースノート	リリースノートには、最新リリースの概要が記載されています。新機能、解決された問題、既知の問題が具体的に記載されています。
AWS 開発者リソースセンター	ドキュメンテーション、コード例、リリースノートをはじめとする、AWS ベースの革新的なアプリケーション開発に役立つさまざまな情報が収められた、中心的起点となるリソースセンターです。
AWS マネジメントコンソール	コンソールでは、プログラミングを行うことなく、Amazon CloudWatch とその他の AWS 製品のほとんどの機能を実行できます。
Amazon CloudWatch ディスカッションフォーラム	Amazon CloudWatch に関連する技術的な質問を提起するための、開発者向けコミュニティベースフォーラムです。
AWS サポートセンター	当社の開発者フォーラム、技術上のよくある質問、サービスステータスページ、Premium Support へのアクセスを含む、AWS テクニカルサポートのホームページです。
AWS プレミアムサポート	AWS プレミアムサポートに関する情報のメインのウェブページです。マンツーマンの、対応の迅速なサポートチャネルで、AWS インフラストラクチャサービスでお客様がアプリケーションを構築して実行する手助けを行っています。
Amazon CloudWatch 製品情報	Amazon CloudWatch に関する情報のメインのウェブページです。
お問い合わせ	AWS の支払、アカウント設定その他に関する連絡先です。

フィードバックをお寄せください

お客様からのフィードバックは、AWSのドキュメントをさらに有益で使いやすくするうえで重要です。Amazon CloudWatch を使い始めるにあたってお気づきになったことを、[こちらのアンケート](#)からお知らせください。

ご協力ありがとうございます。

文書履歴

この文書は、Amazon CloudWatch の 2013-09-06 リリースに関連付けられています。本ガイドの最終更新日は 31 October 2013 です。

次の表は、『*Amazon CloudWatch Getting Started Guide*』の前回リリース以降の重要な変更点をまとめたものです。

変更点	説明	リリース日
コンテンツの削除	Publish Metrics to CloudWatch (p. 7) のセクションから検証のステップを削除しました。検証は必要ないからです。CloudWatch コマンドラインインターフェイスからは、メトリックスを作成できない場合に応答が返されます。	2011 年 6 月 6 日
パブリックリリース	これは『 <i>Amazon CloudWatch Getting Started Guide</i> 』の最初のリリースです。	2011 年 5 月 10 日

本ガイドについて

本ガイドは *Amazon CloudWatch Getting Started Guide* です。最終更新日は October 31, 2013 です。

Amazon CloudWatch は、このガイドの中で「CloudWatch」と表記されることもありますが、その場合もすべての著作権と法的保護が適用されます。