

BPAユーザガイドDistributed Tracing Framework v5.1

- [はじめに](#)
 - [ジプキン](#)
 - [@opentelemetry/api](#)
- [主なコンポーネント](#)
- [前提条件](#)
 - [ネットワーク要件](#)
- [Dockerによる導入の構成](#)
 - [モニタリングプロファイルの有効化](#)
 - [Zipkinコンテナの確認](#)
- [サービスでのトレースの有効化](#)
 - [構成の読み込み階層](#)
 - [ポッド固有の設定](#)
 - [使用例](#)
 - [サンプル シナリオ](#)
 - [グローバル設定](#)
 - [使用例](#)
 - [サンプル シナリオ](#)
 - [フォールバック設定](#)
 - [使用例](#)
 - [トレース設定の設定](#)
 - [構成のプロパティ](#)
 - [設定変更の適用](#)
 - [Helm Upgradeコマンド](#)
- [KubernetesおよびHelmの導入](#)
 - [値の監視の有効化](#)
 - [Zipkinサービスの展開](#)
 - [導入の検証](#)
 - [サービスステータスの確認](#)
- [アクセスとモニタリング](#)
 - [Zipkinダッシュボードへの1ノードアクセス](#)
 - [3ノードKubernetesのZipkinダッシュボードへのアクセス](#)
 - [Zipkin UIの使用](#)
- [トレーサとスパン](#)

はじめに

ジプキン

Zipkinは、サービスアーキテクチャの遅延問題のトラブルシューティングに必要なタイミングデータの収集を支援する分散型トレースシステムです。このガイドでは、Zipkinを展開し、BPAプラットフォームサービス間でトレースを有効にする手順について説明します。

@opentelemetry/api

@opentelemetry/apiパッケージは、Node.jsのOpenTelemetryのコアAPIです。トレース、スパン、およびコンテキスト伝達を作成および管理するためのインターフェイスとクラスを提供します。このパッケージを使用すると、開発者はトレースやメトリックなどのテレメトリデータを収集するためにアプリケーションをインストルメント化し、Zipkinなどのバックエンドにエクスポートして分析できます。

主なコンポーネント

Zipkinの主なコンポーネントは次のとおりです。

- Zipkin Server : セントラルトレースコレクタおよびユーザインターフェイス(UI)
- OpenTelemetry : トレース用のインストルメンテーションライブラリ
- Elasticsearch : トレースデータのストレージバックエンド
- Secure Sockets Layer(SSL)およびTransport Layer Security(TLS) : セキュアな通信を提供します。

前提条件

ネットワーク要件

Zipkinを配備してトレースを有効にするには、次のネットワーク要件が必要です。

- ポート9412:Zipkin Hypertext Transfer Protocol Secure(HTTPS)APIエンドポイント
- ポート9411:Zipkin Hypertext Transfer Protocol(HTTP)管理エンドポイント
- ポート9200:Elasticsearchクラスタへのアクセス

Dockerによる導入の構成

モニタリングプロファイルの有効化

1. 次のパスからBPA 1ノードDockerを起動します。

```
cd /opt/bpa/bpa-{build_version}/scripts
```

2. 次のコマンドを使用して、モニタリングオプションを指定してbashスクリプトを実行します。

```
./startbpa.1node.sh monitoring
```

Zipkinコンテナの確認

1. コンテナのステータスを確認するには、次のコマンドを実行します。

```
docker ps | grep tracers-zipkin
```

2. Zipkinコンテナのログを表示するには、次のコマンドを実行します。

```
docker logs tracers-zipkin -f
```

サービスでのトレースの有効化

BPAアプリケーションは、「tracingConfig.json」ファイルを使用した柔軟なトレース設定をサポートしています。システムは、さまざまな導入シナリオに対応するために、3つのレベルの優先順位を持つ階層型構成の読み込みメカニズムを実装しています。

構成の読み込み階層

アプリケーションは、優先順位に基づいて次の順序でトレース設定をロードします。

ポッド固有の設定

パス:bpa-helm-chart/charts/<サービス名>/public_conf

使用例

- 個々のポッド固有のトレース設定
- 特定のサービスまたはポッドに対して異なるトレース設定が可能
- 手動配置が必要：ユーザは、このファイルをポッドのフォルダパスに手動で配置する必要があります：bpa-helm-chart/charts/<サービス名>/public_conf

サンプル シナリオ

- 重要なサービスのみの詳細トレースを有効にする
- マイクロサービスごとにサンプリングレートが異なる
- ポッドのトラブルシューティングのためのデバッグ固有のトレース

グローバル設定

パス:bpa-helm-chart/bpa/conf/common/globals/tracingConfig.json

使用例

- すべてのコンテナとポッドのグローバルトレース設定
- 一元化されたトレース管理
- グローバルな影響：このファイルに対する変更は、導入環境内のすべてのポッドに影響します。

サンプル シナリオ

- BPA展開全体でトレースを有効または無効にする
- 本番環境に対して統一されたトレーシング・ポリシーを設定
- コンプライアンスとモニタリングの要件の一元化

フォールバック設定

パス：../conf/tracingConfig.json (フォールバック)

使用例

- アプリケーションにバンドルされているデフォルトのトレース設定
- 開発環境の既定値
- トレース設定の欠落が原因でアプリケーションが失敗しないようにします。

コンフィギュレーションファイルの形式が正しくない、または読み取り不能である場合、システムは次の処理を行います。

- エラーを記録します。
- トレースエラーを回避するため、既定値は{enable: false}です
- アプリケーションの起動を継続

トレース設定の設定

ポッドのフォルダパスにある次のコンテンツを使用して、「tracingConfig.json」ファイルを作成または更新します。

bpa-helm-chart/charts/<サービス名>/public_conf/

例:

- トレースを有効にするには

```
{  
    "enable": true  
}
```

- トレースを無効にするには :

```
{  
    "enable": false  
}
```

構成のプロパティ

- enable (boolean) (必須) : アプリケーションのトレースを有効にするか無効にするかを制御します
- true : 分散トレースを有効にします。
 - アプリケーションはトレースヘルパーを初期化します
 - トレースが収集され、設定されたバックエンドに送信されます。
 - パフォーマンスモニタリングおよびデバッグ機能の有効化
- false : 分散トレースを無効にします。
 - アプリケーションのパフォーマンスに対するトレースのオーバーヘッドなし
 - トレースデータの収集または送信なし
 - トレース機能なしでアプリケーションを実行

設定変更の適用

Helm Upgradeコマンド

「tracingConfig.json」ファイルを適切な場所に配置した後、Helm upgradeコマンドを使用して変更を適用します。

```
helm upgrade bpa-rel --namespace bpa-ns /opt/bpa-helm-chart
```

コマンドの内訳 :

- helm upgrade : 既存のHelmリリースを更新します。
- bpa-rel:BPA Helmリリースの名前
- --namespace bpa-ns: BPAがデプロイされるKubernetes名前空間を指定します。
- /opt/bpa-helm-chart: BPA Helmグラフディレクトリへのパス

KubernetesおよびHelmの導入

値の監視の有効化

次のヘルム・チャート・パスで、「values.yaml」ファイルのヘルム値を編集して監視を有効にします :

```
yaml
global:
```

```
enableMonitoring: true
```

Zipkinサービスの展開

次のHelmコマンドを使用して、Zipkinサービスを配備します。

```
helm install bpa-rel --create-namespace --namespace bpa-ns
```

導入の検証

展開を確認するには、次のコマンドを使用します。

```
kubectl get pods -n bpa-ns | grep tracers-zipkin  
kubectl get svc -n bpa-ns | grep tracers-zipkin
```

サービスステータスの確認

ポッドログを表示するには、次のコマンドを使用します。

```
kubectl logs -n bpa-ns deployment/tracers-zipkin -f
```

サービスエンドポイントを確認するには、次のコマンドを使用します。

```
kubectl describe svc tracers-zipkin -n bpa-ns
```

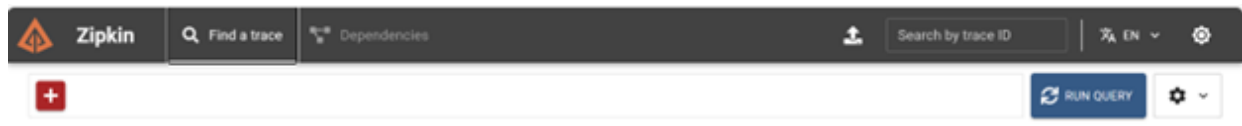
アクセスとモニタリング

Zipkinダッシュボードへの1ノードアクセス

URL:<https://<サーバIP>:9412/zipkin/>

3ノードKubernetesのZipkinダッシュボードへのアクセス

URL:<https://<cluster-ip>:30900/zipkin/>



Search Traces

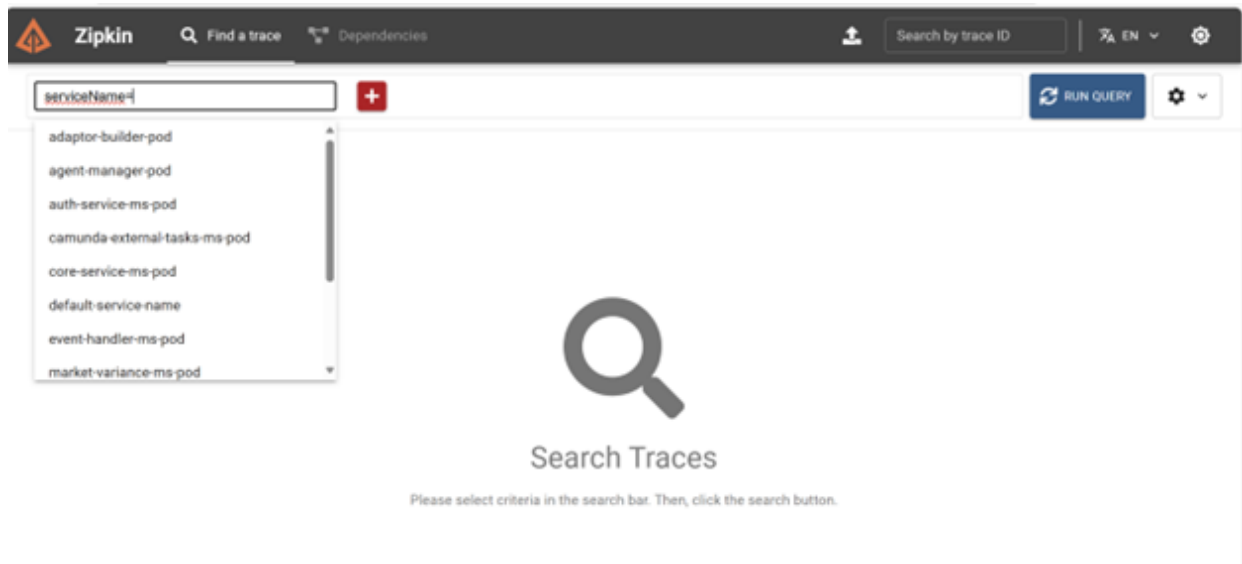
Please select criteria in the search bar. Then, click the search button.

Zipkinダッシュボード

Zipkin UIの使用

Zipkinダッシュボードには、トレースを検索して視覚化するためのUIが用意されています。主なコンポーネントは次のとおりです。

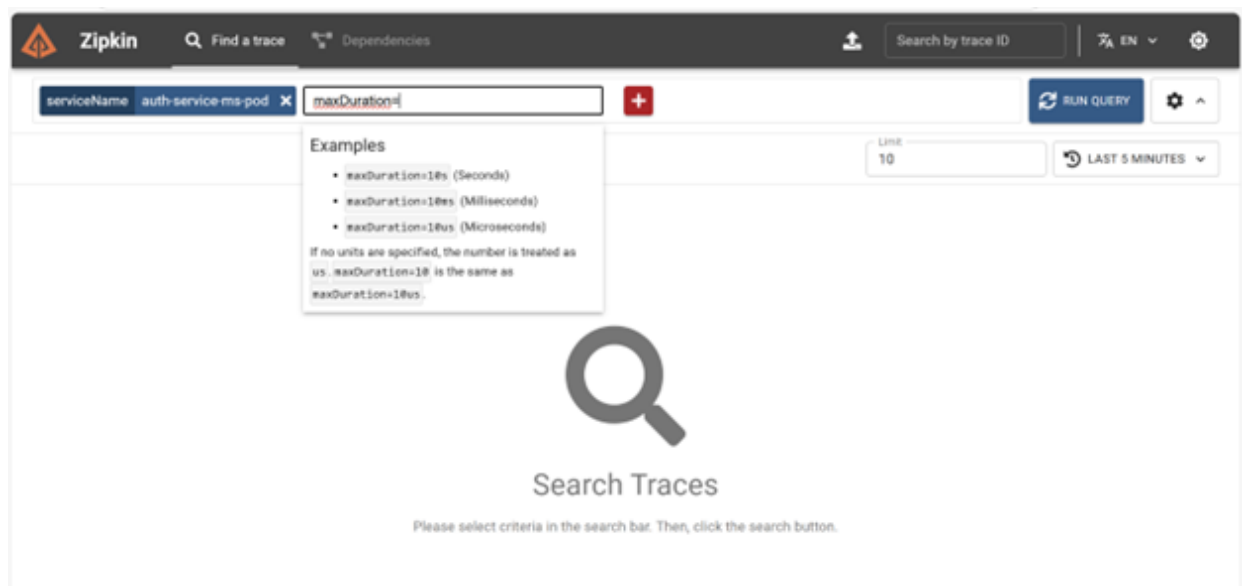
- サービス名のドロップダウン：サービス名でトレースをフィルタリングします。



Service Name

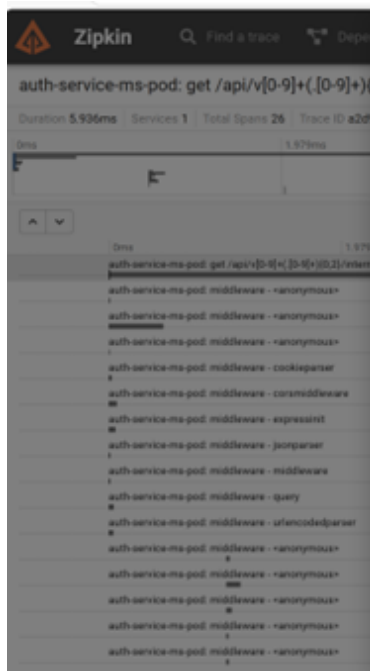
Service Name

- 時間範囲セクタ：トレース検索の時間ウィンドウを定義します。



時間範囲セクタ

- トレースリスト：検索条件に一致するトレースのリストを表示します



Span ID	Service name	Span name	Start time	Duration
48cfe1ec3faa26	auth-service-ms-pod	get /api/v[0-9]+([0-9]+)([0-9]+)	07/18 08:37:45.107	5.936ms
0eabf964630c0f2	auth-service-ms-pod	middleware - <anonymo..	07/18 08:37:45.107	12.000µs
45990fb36337736	auth-service-ms-pod	middleware - <anonymo..	07/18 08:37:45.107	467.000µs
533ab670a89b1b0	auth-service-ms-pod	middleware - <anonymo..	07/18 08:37:45.107	8.000µs
2a28cd577647910f	auth-service-ms-pod	middleware - cookieparser	07/18 08:37:45.107	26.000µs
c83796ebd2f1b956	auth-service-ms-pod	middleware - corsmiddle..	07/18 08:37:45.107	70.000µs
0c71a399b1a749b4	auth-service-ms-pod	middleware - expressint	07/18 08:37:45.107	57.000µs
da6798c5d3aa9444	auth-service-ms-pod	middleware - jsonparser	07/18 08:37:45.107	15.000µs
50a2619ac283995	auth-service-ms-pod	middleware - middleware	07/18 08:37:45.107	15.000µs
c75eab2679b57c	auth-service-ms-pod	middleware - query	07/18 08:37:45.107	43.000µs
b697d958232a6fc	auth-service-ms-pod	middleware - urlencoded..	07/18 08:37:45.107	40.000µs
0893daa4fb0a89ef	auth-service-ms-pod	middleware - <anonymo..	07/18 08:37:45.108	31.000µs
4a94cb7194b66807	auth-service-ms-pod	middleware - <anonymo..	07/18 08:37:45.108	121.000µs
0b1a7396d43013	auth-service-ms-pod	middleware - <anonymo..	07/18 08:37:45.108	50.000µs
989da40cc86036e0	auth-service-ms-pod	middleware - <anonymo..	07/18 08:37:45.108	19.000µs
b05ed4820ca2288	auth-service-ms-pod	middleware - <anonymo..	07/18 08:37:45.108	24.000µs
c038a16b2833b0	auth-service-ms-pod	middleware - <anonymo..	07/18 08:37:45.108	53.000µs
7a19ee5d040e2179	auth-service-ms-pod	middleware - initialize	07/18 08:37:45.108	74.000µs
a20a9a96a2650c47	auth-service-ms-pod	router - /about/cisco-bp..	07/18 08:37:45.108	3.000µs
c0e72796d4f56b7a	auth-service-ms-pod	router - /about/cisco-bp..	07/18 08:37:45.108	4.000µs

スパンの詳細

トレースとスパン

トレースは、スパンの作成と管理を行います。トレースは、特定のサービスまたはアプリケーションコンポーネントに関連付けられます。スパンは、トレース内の作業または操作の単一の単位を表します。各スパンには、操作名、開始時刻と終了時刻、属性、他のスパンとの親子関係などの情報が含まれます。

翻訳について

シスコは世界中のユーザにそれぞれの言語でサポート コンテンツを提供するために、機械と人による翻訳を組み合わせて、本ドキュメントを翻訳しています。ただし、最高度の機械翻訳であっても、専門家による翻訳のような正確性は確保されません。シスコは、これら翻訳の正確性について法的責任を負いません。原典である英語版（リンクからアクセス可能）もあわせて参照することを推奨します。