Data Collector Reference - サービス

Data Infrastructure Insights

NetApp
September 25, 2024

This PDF was generated from https://docs.netapp.com/ja-jp/cloudinsights/task_config_telegraf_node.html on September 25, 2024. Always check docs.netapp.com for the latest.

目次

Data Collector Reference - サービス . . . 1
ノードデータ収集 . . . 1
ActiveMQ データコレクタ . . . 4
Apache Data Collector . . . 6
総領事データ収集 . . . 10
Couchbase Data Collector . . . 10
CouchDB データコレクタ . . . 13
Docker Data Collector . . . 15
Elasticsearch Data Collector . . . 23
Flink Data Collector の使用 . . . 28
Hadoop Data Collector . . . 35
HAProxy Data Collector . . . 46
JVM Data Collector （ JVM データ収集） . . . 54
Kafka Data Collector の利用 . . . 58
Kibana データコレクタ . . . 63
Kubernetes Monitoring Operatorのインストールと設定 . . . 65
Memcached Data Collector . . . 84
MongoDB データコレクタ . . . 88
MySQL データコレクタ . . . 90
netstat Data Collector の場合 . . . 95
nginx データコレクタ . . . 97
PostgreSQL データコレクタ . . . 99
Puppet Agent データコレクタ . . . 101
Redis Data Collector の場合 . . . 104

Data Collector Reference - サービス

ノードデータ収集

Data Infrastructure Insightsは、エージェントをインストールしたノードから指標を収集します。

インストール

1. [Observability]>[Collectors]*で、オペレーティングシステム/プラットフォームを選択します。統合データコレクタ（ Kubernetes 、 Docker 、 Apache など）をインストールすると、ノードのデータ収集も設定されることに注意してください。
2. 指示に従って、エージェントを設定します。手順は、データの収集に使用するオペレーティングシステムまたはプラットフォームのタイプによって異なります。

オブジェクトとカウンタ

次のオブジェクトとそのカウンタがノードの指標として収集されます。

オブジェクト：	識別子：	属性：	「 **dataPoints** 」：
ノードファイルシステム	ノード UUID デバイス パス を入力します	ノードIP ノード名 ノードOS モード	無料 空きinode inodeの合計 使用されているinode合計 使用済み合計 使用済み
ノードディスク	ノード UUID ディスク	ノードIP ノード名 ノードOS	IO時間の合計 IOPSを実行中です 読み取りバイト数（1秒あたり） 読み取り時間合計 読み取り数（1秒あたり） Weighted IO Time Totalの略 書き込みバイト数（1秒あたり） 書き込み時間合計 1秒あたりの書き込み数 現在のディスクキューの長さ 書き込み時間 読み取り時間 IO時間

オブジェクト：	識別子：	属性：	「 **dataPoints** 」：
ノードCPU	ノード UUID CPU	ノードIP ノード名 ノードOS	システムCPU使用率 ユーザCPU使用率 アイドルCPU使用率 プロセッサCPU使用率 割り込みCPU使用率 DPC CPU使用率

オブジェクト：	識別子：	属性：	「 **dataPoints** 」：
ノード	ノード UUID	ノードIP ノード名 ノードOS	カーネル起動時間 カーネルコンテキストスイッチ（1秒あたり） カーネルエントロピーが利用可能です カーネル割り込み（1秒あたり） カーネルプロセスのフォーク（1秒あたり） メモリがアクティブです 使用可能なメモリの合計 使用可能なメモリ メモリがバッファされました メモリキャッシュ メモリコミット制限 メモリはとしてコミットされます メモリが汚れています メモリ空き メモリの空き容量が大きい Memory High Totalの略 メモリのページサイズが大きすぎます メモリ巨大なページ無料 Memory Huge Pages Totalの略 メモリが不足しています Memory Low Totalの略 メモリマップ済み Memory Page Tablesの略 メモリ共有 メモリスラブ メモリスワップキャッシュ メモリスワップフリー メモリスワップの合計 メモリ合計 使用メモリの合計 使用されているメモリ メモリvmallocチャンク メモリvmallocの合計 メモリvmallocが使用されています メモリが配線されています メモリライトバック合計 Memory Writeback tmp（ メモリライトバックtmp メモリキャッシュの障害 メモリ要求ゼロエラー メモリページ障害 メモリページ

オブジェクト：	識別子：	属性：	「 **dataPoints** 」：
ノードネットワーク	Network Interface の略 ノード UUID	ノード名 ノードIP ノードOS	受信したバイト数 送信されたバイト数 送信されたパケットが破棄されました Packets Outboud Errors（パケット送信エラー） 受信したパケットは破棄されました パケット受信エラー 受信したパケット 送信されたパケット

セットアップ（ **Setup** ）

セットアップおよびトラブルシューティングの情報は、にあります "エージェントの設定" ページ

ActiveMQ データコレクタ

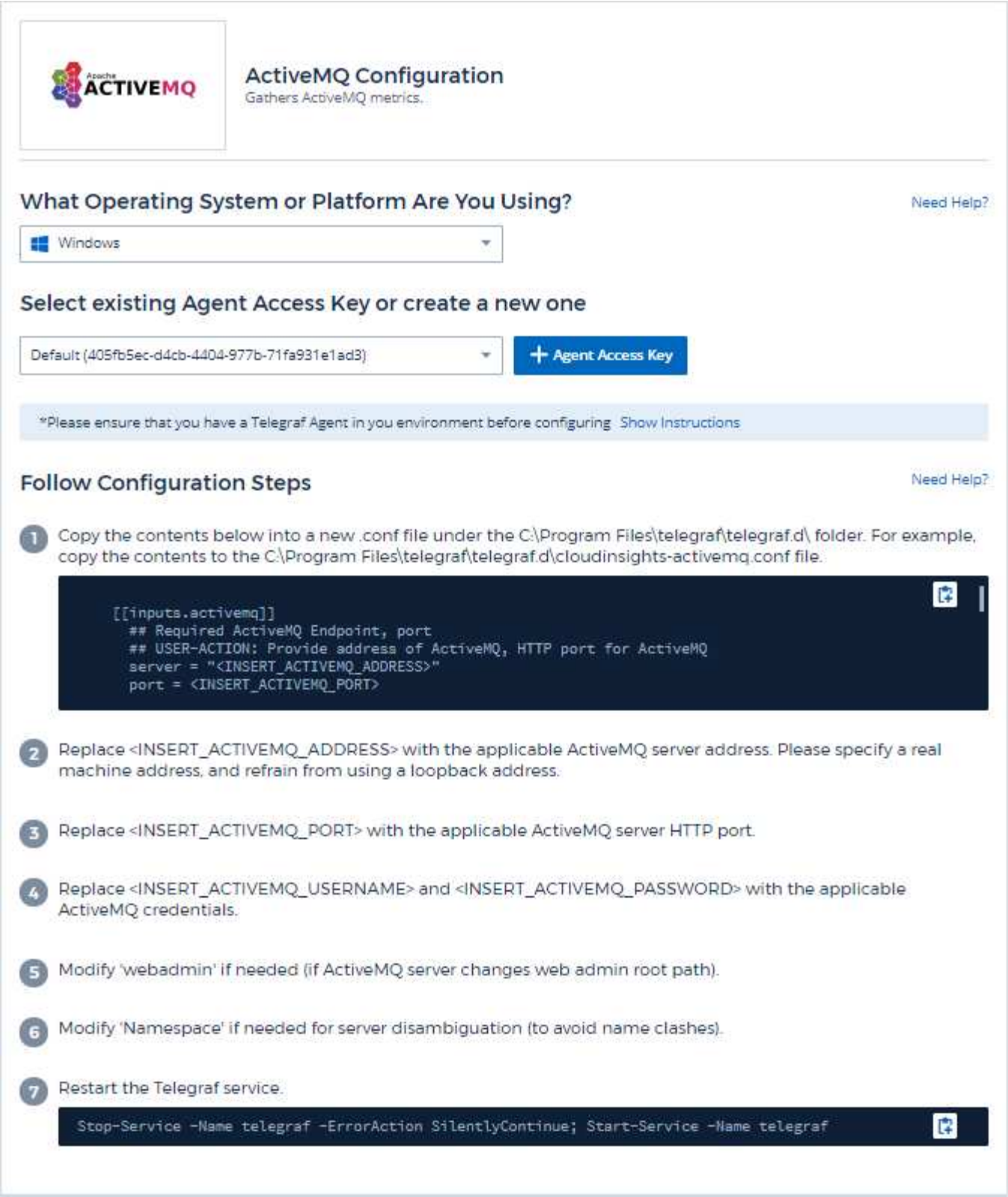
Data Infrastructure Insightsは、このデータコレクタを使用してActiveMQから指標を収集します。

インストール

1. [Observability]>[Collectors]で、+ Data Collector *をクリックします。[ActiveMQ]を選択します。

   Tegraf エージェントがインストールされているオペレーティングシステムまたはプラットフォームを選択します。

2. Agent for Collection をまだインストールしていない場合、または別のオペレーティングシステムまたはプラットフォームに Agent をインストールする場合は、 *Show Instructions* をクリックしてを展開します "エージェントのインストール" 手順
3. このデータコレクタで使用するエージェントアクセスキーを選択します。[**+Agent Access Key**] ボタンをクリックすると、新しいエージェントアクセスキーを追加できます。ベストプラクティス：別のエージェントアクセスキーを使用するのは、たとえば OS/ プラットフォーム別にデータコレクタをグループ化する場合だけです。
4. 設定手順に従ってデータコレクタを設定します。手順は、データの収集に使用するオペレーティングシステムまたはプラットフォームのタイプによって異なります。



セットアップ（ **Setup** ）

情報は、に記載されています "ActiveMQ のドキュメント"

オブジェクトとカウンタ

次のオブジェクトとそのカウンタが収集されます。

オブジェクト：	識別子：	属性：	「 **dataPoints** 」：
ActiveMQ キュー	ネームスペース キュー ポート サーバ	ノード名 ノードIP ノード UUID	消費者数 デキューカウント enqueueカウント キューサイズ
ActiveMQ サブスクライバ	クライアント ID 接続ID ポート サーバ ネームスペース	はアクティブです 宛先 ノード名 ノードIP ノード UUID ノードOS セレクタ サブスクリプション。	デキューカウント ディスパッチ数 ディスパッチキューサイズ enqueueカウント 保留中のキューサイズ
ActiveMQ トピック	トピック ポート サーバ ネームスペース	ノード名 ノードIP ノード UUID ノードOS	消費者数 デキューカウント enqueueカウント サイズ

トラブルシューティング

追加情報はから入手できます "サポート" ページ

Apache Data Collector

このデータコレクタを使用すると、環境内の Apache サーバからデータを収集できます。

前提条件

- Apache HTTP Server がセットアップされ、適切に実行されている必要があります
- エージェントのホスト / VM に対する sudo 権限または管理者権限が必要です
- 通常、 apache_mod_status_module は、 Apache サーバの「 /server-status?auto 」場所にページを公開するように設定されています。使用可能なすべてのフィールドを収集するには、 *ExtendedStatus* オプションを有効にする必要があります。サーバの設定方法については、 Apache モジュールのドキュメントを参照してください。 https://httpd.apache.org/docs/2.4/mod/mod_status.html#enable

インストール

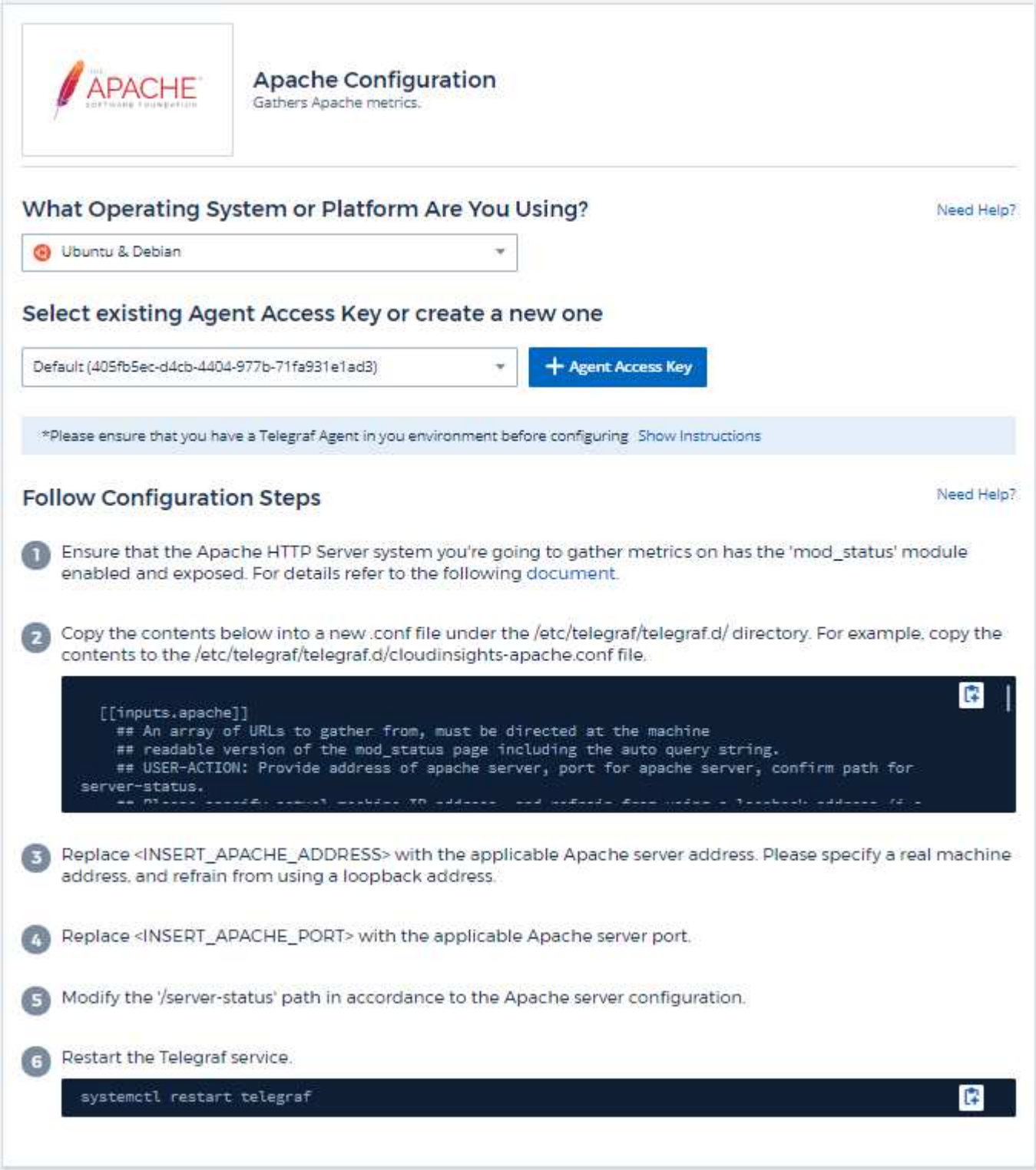
1. [Observability]>[Collectors]で、+ Data Collector *をクリックします。Apacheを選択します。

   Tegraf エージェントがインストールされているオペレーティングシステムまたはプラットフォームを選択します。

2. Agent for Collection をまだインストールしていない場合、または別のオペレーティングシステムまたはプラットフォームに Agent をインストールする場合は、 *Show Instructions* をクリックしてを展開します "エ

ージェントのインストール" 手順

3. このデータコレクタで使用するエージェントアクセスキーを選択します。[**+Agent Access Key**] ボタンをクリックすると、新しいエージェントアクセスキーを追加できます。ベストプラクティス：別のエージェントアクセスキーを使用するのは、たとえば OS/ プラットフォーム別にデータコレクタをグループ化する場合だけです。
4. 設定手順に従ってデータコレクタを設定します。手順は、データの収集に使用するオペレーティングシステムまたはプラットフォームのタイプによって異なります。



セットアップ（ **Setup** ）

Telegraf の Apache の HTTP Server 用プラグインは 'OD _status' モジュールを有効にする必要があります。これを有効にすると、 Apache の HTTP サーバは、ブラウザで表示したり、 Apache の HTTP サーバ設定の状態を抽出するためにスクレイピングされた HTML エンドポイントを公開します。

互換性：

構成は Apache の HTTP Server バージョン 2.4.38 に対して開発されました。

mod_status を有効にします。

'OD_status' モジュールを有効にして公開するには、次の 2 つの手順を実行します。

- イネーブルモジュール
- モジュールから統計情報を公開しています

イネーブルモジュール：

モジュールのロードは '/usr/local/apache/conf/httpd.conf ' の下の config ファイルによって制御されます。構成ファイルを編集し、次の行のコメントを解除します。

```
LoadModule status_module modules/mod_status.so
Include conf/extra/httpd-info.conf
```

モジュールからの統計情報の公開：

'OD_status' の公開は '/usr/local/apache2/conf/extra /httpd-info.conf' の下の config ファイルによって制御されます。設定ファイルに次のものがあることを確認してください (少なくとも、他のディレクティブが存在することを確認してください) 。

```
# Allow server status reports generated by mod_status,
# with the URL of http://servername/server-status
<Location /server-status>
    SetHandler server-status
</Location>

#
# ExtendedStatus controls whether Apache will generate "full" status
# information (ExtendedStatus On) or just basic information
(ExtendedStatus
# Off) when the "server-status" handler is called. The default is Off.
#
ExtendedStatus On
```

'OD_status' モジュールの詳細な手順については、を参照してください "Apache のドキュメント"

オブジェクトとカウンタ

次のオブジェクトとそのカウンタが収集されます。

オブジェクト：	識別子：	属性：	「 **dataPoints** 」：
Apache	ネームスペース サーバ	ノードIP ノード名 ポート 親サーバー構成の生成 親サーバーのMPM生成 サーバの稼働時間 が停止しています	多忙な労働者 要求あたりのバイト数 1秒あたりのバイト数 CPU Children Systemの略 CPU Children Userの略 CPU負荷 CPUシステム CPUユーザ 非同期接続が終了しています 非同期接続のキープアライブ 非同期接続の書き込み接続合計 リクエストごとの期間 アイドル状態の従業員 負荷平均（過去1m） 負荷平均（過去15m） 負荷平均（過去5m） プロセス 1秒あたりの要求数 合計アクセス数 合計期間 合計KB スコアボード終了 スコアボードのDNSルックアップ スコアボードの仕上げ スコアボードアイドルクリーンアップ スコアボードキープアライブ スコアボードログ スコアボードが開きます スコアボードの読み取り スコアボード送信 スコアボードが開始されました スコアボード待機中

トラブルシューティング

追加情報はから入手できます "サポート" ページ

総領事データ収集

Data Infrastructure Insightsは、このデータコレクタを使用してConsulから指標を収集します。

インストール

1. [Observability]>[Collectors]で、+ Data Collector *をクリックします。Consulを選択します。

   収集用にエージェントを設定していない場合は、にプロンプトが表示されます "エージェントをインストールします" お客様の環境で実現します。

   エージェントがすでに設定されている場合は、適切な OS またはプラットフォームを選択し、 [* Continue（続行）] をクリックします。

2. [Consul Configuration] 画面の指示に従って、データコレクタを設定します。手順は、データの収集に使用するオペレーティングシステムまたはプラットフォームのタイプによって異なります。

セットアップ（ **Setup** ）

情報は、に記載されています "総領事からのご説明"。

接続のためのオブジェクトとカウンタ

次のオブジェクトとそのカウンタが収集されます。

オブジェクト：	識別子：	属性：	「 **dataPoints** 」：
総領事	ネームスペース IDを確認します サービスノード	ノードIP ノードOS ノード UUID ノード名 サービス名 [名前（ Name ）] を サービス ID ステータス	重要 パス 警告

トラブルシューティング

追加情報はから入手できます "サポート" ページ

Couchbase Data Collector

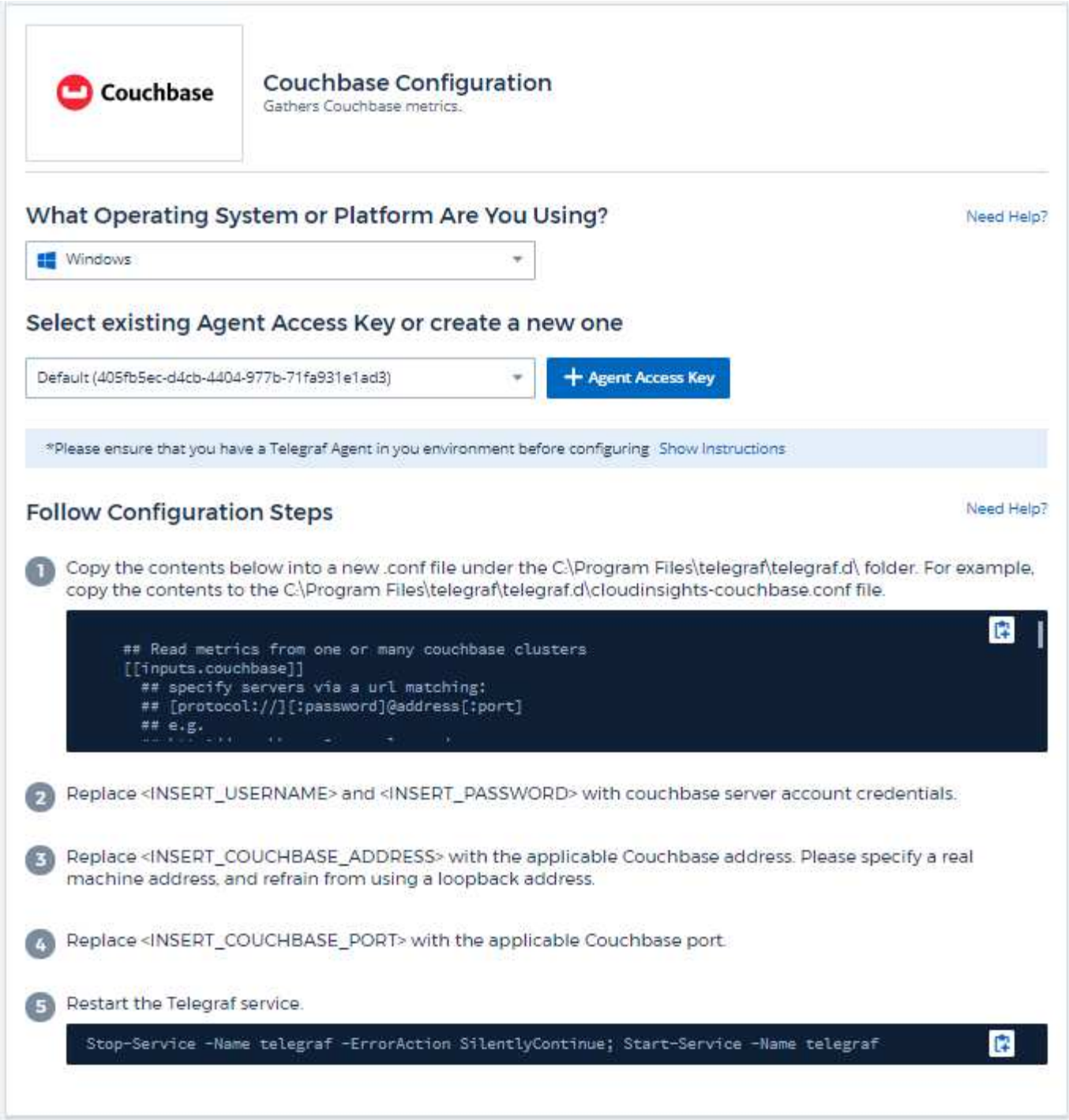
Data Infrastructure Insightsは、このデータコレクタを使用してCouchbaseから指標データを収集します。

インストール

1. [Observability]>[Collectors]で、+ Data Collector *をクリックします。[Couchbase]を選択します。

Tegraf エージェントがインストールされているオペレーティングシステムまたはプラットフォームを選択します。

2. Agent for Collection をまだインストールしていない場合、または別のオペレーティングシステムまたはプラットフォームに Agent をインストールする場合は、 *Show Instructions* をクリックしてを展開します "エージェントのインストール" 手順
3. このデータコレクタで使用するエージェントアクセスキーを選択します。[**+Agent Access Key**] ボタンをクリックすると、新しいエージェントアクセスキーを追加できます。ベストプラクティス：別のエージェントアクセスキーを使用するのは、たとえば OS/ プラットフォーム別にデータコレクタをグループ化する場合だけです。
4. 設定手順に従ってデータコレクタを設定します。手順は、データの収集に使用するオペレーティングシステムまたはプラットフォームのタイプによって異なります。



セットアップ（ **Setup** ）

情報は、に記載されています "Couchbase ドキュメント"。

オブジェクトとカウンタ

次のオブジェクトとそのカウンタが収集されます。

オブジェクト：	識別子：	属性：	「 **dataPoints** 」：
Couchbase Node	ネームスペース クラスタ Couchbaseノードのホスト名	ノード名 ノードIP	メモリ空き メモリ合計
Couchbase バケット	ネームスペース バケット クラスタ	ノード名 ノードIP	使用済みデータ データフェッチ 使用されているディスク アイテム数 使用されているメモリ 1秒あたりの処理数 使用済みクォータ

トラブルシューティング

追加情報はから入手できます "サポート" ページ

CouchDB データコレクタ

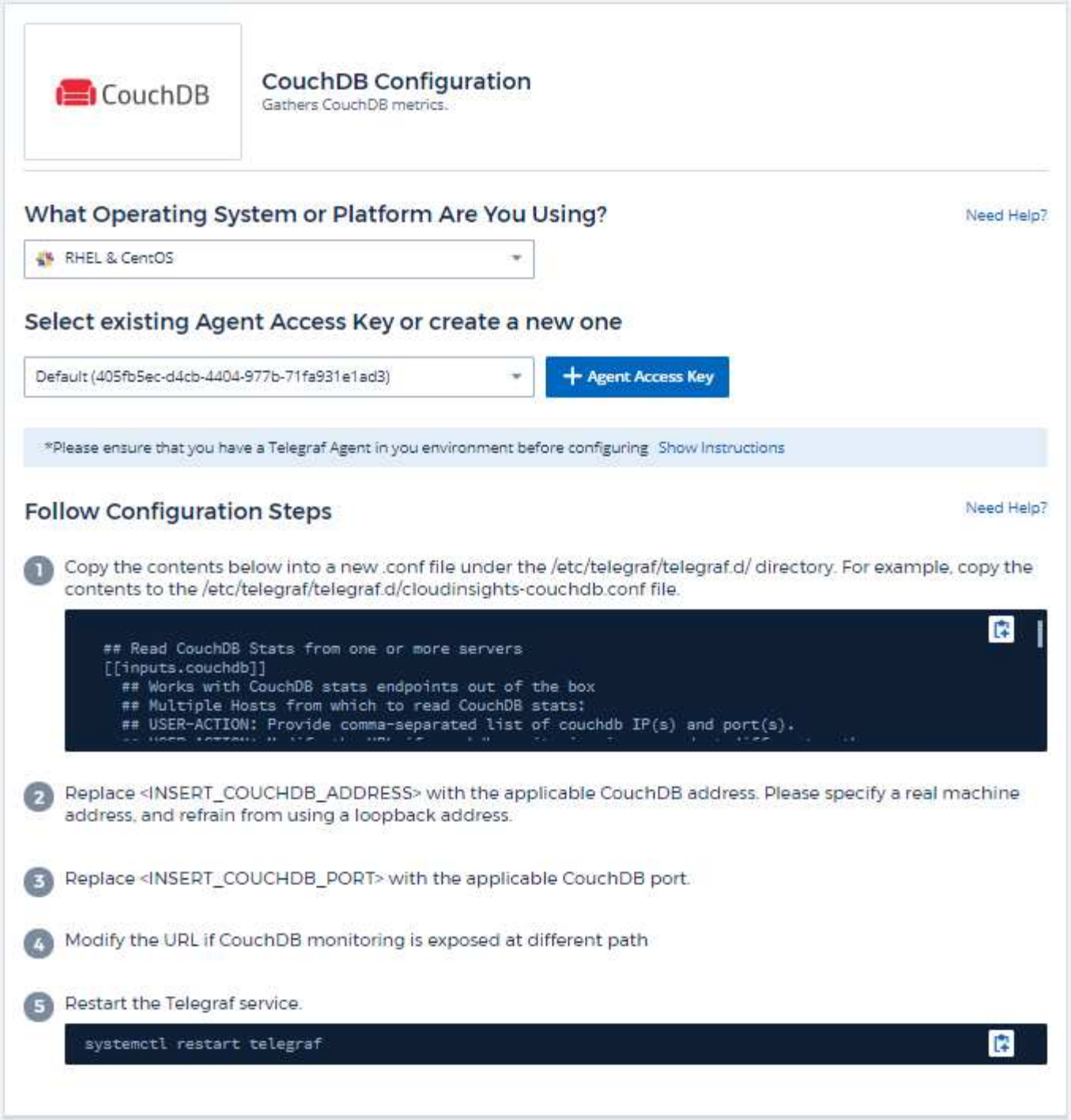
Data Infrastructure Insightsは、このデータコレクタを使用してCouchDBから指標データを収集します。

インストール

1. [Observability]>[Collectors]で、+ Data Collector *をクリックします。[CouchDB]を選択します。

   Tegraf エージェントがインストールされているオペレーティングシステムまたはプラットフォームを選択します。

2. Agent for Collection をまだインストールしていない場合、または別のオペレーティングシステムまたはプラットフォームに Agent をインストールする場合は、 *Show Instructions* をクリックしてを展開します "エージェントのインストール" 手順
3. このデータコレクタで使用するエージェントアクセスキーを選択します。[**+Agent Access Key**] ボタンをクリックすると、新しいエージェントアクセスキーを追加できます。ベストプラクティス：別のエージェントアクセスキーを使用するのは、たとえば OS/ プラットフォーム別にデータコレクタをグループ化する場合だけです。
4. 設定手順に従ってデータコレクタを設定します。手順は、データの収集に使用するオペレーティングシステムまたはプラットフォームのタイプによって異なります。



セットアップ（ **Setup** ）

情報は、に記載されています "CouchDB のドキュメント"。

オブジェクトとカウンタ

次のオブジェクトとそのカウンタが収集されます。

オブジェクト：	識別子：	属性：	「 **dataPoints** 」：
CouchDB	ネームスペース サーバ	ノード名 ノードIP	認証キャッシュヒット 認証キャッシュミス データベースの読み取り データベースへの書き込み データベースが開きます OSファイルを開きます 最大要求時間 最小要求時間 httpdリクエストメソッドコピー httpdリクエストメソッド削除 httpdリクエストメソッドGET httpdリクエストメソッドヘッド httpdリクエストメソッドPost httpdリクエストメソッドPUT ステータスコード200 ステータスコード201 ステータスコード202 ステータスコード301 ステータスコード304. ステータスコード400 ステータスコード401 ステータスコード403 ステータスコード404 ステータスコード405 ステータスコード409 ステータスコード412 ステータスコード500

トラブルシューティング

追加情報はから入手できます "サポート" ページ

Docker Data Collector

Data Infrastructure Insightsは、このデータコレクタを使用してDockerから指標を収集します。

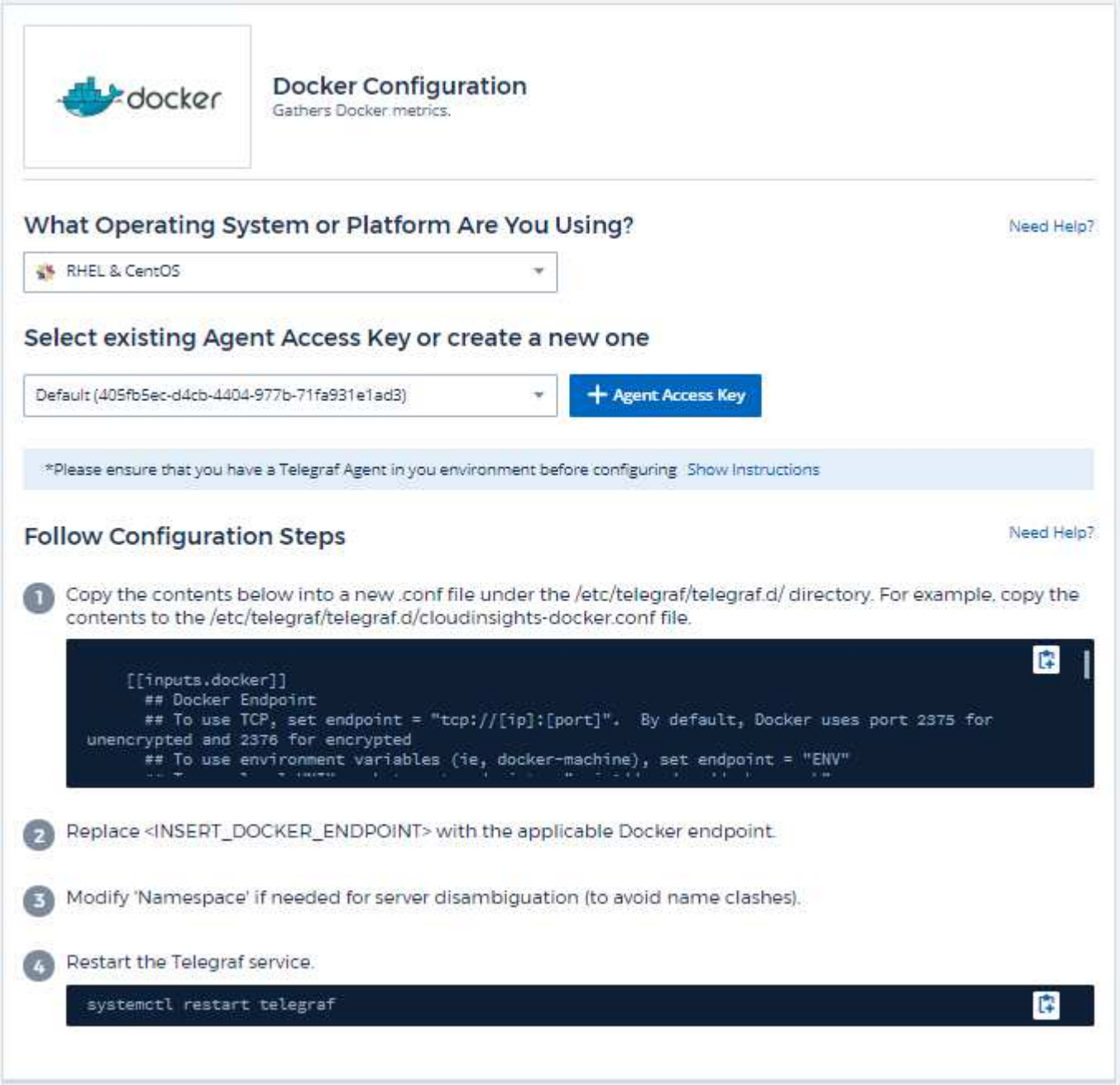
インストール

1. [Observability]>[Collectors]で、+ Data Collector *をクリックします。[Docker]を選択します。

   収集用にエージェントを設定していない場合は、にプロンプトが表示されます "エージェントをインストールします" お客様の環境で実現します。

エージェントがすでに設定されている場合は、適切な OS またはプラットフォームを選択し、 [* Continue（続行）] をクリックします。

2. Docker Configuration 画面の指示に従って、データコレクタを設定します。手順は、データの収集に使用するオペレーティングシステムまたはプラットフォームのタイプによって異なります。



セットアップ（ **Setup** ）

Telegraf Docker 用入力プラグインは、指定された UNIX ソケットまたは TCP エンドポイントを介してメトリックを収集します。

互換性

Docker バージョン 1.12.6 に対して構成が開発されました。

セットアップ中です

UNIX ソケット経由で **Docker** にアクセスする

Tegraf エージェントが BareMetal 上で実行されている場合は、次のコマンドを実行して、テレグラフ UNIX ユーザを Docker UNIX グループに追加します。

```
 sudo usermod -aG docker telegraf
Telegraf エージェントが Kubernetes
ポッド内で実行されている場合は、ソケットをポッドにボリュームとしてマッピングし、そのボリュームを /var/run/docker.sock にマウントすることで、 Docker Unix
ソケットを公開します。　たとえば、 PodSpec に次の情報を追加します。
```

```
volumes:
...
- name: docker-sock
hostPath:
path: /var/run/docker.sock
type: File
```

次に、次の項目をコンテナに追加します。

```
volumeMounts:
...
- name: docker-sock
mountPath: /var/run/docker.sock
```

Kubernetesプラットフォーム用に提供されているData Infrastructure Insightsインストーラがこのマッピングを自動的に処理します。

TCP エンドポイントを介して **Docker** にアクセスする

デフォルトでは、 Docker は暗号化されていないアクセスにポート 2375 を使用し、暗号化されたアクセスにポート 2376 を使用します。

オブジェクトとカウンタ

次のオブジェクトとそのカウンタが収集されます。

オブジェクト：	識別子：	属性：	「 **dataPoints** 」：
Docker Engine の略	ネームスペース Docker Engine の略	ノード名 ノードIP ノード UUID ノードOS Kubernetesクラスタ Dockerバージョン 単位	メモリ コンテナ コンテナが一時停止しました 実行中のコンテナ コンテナが停止しました CPU ルーチンを実行します イメージ リスナーイベント 使用されているファイル記述子 使用可能なデータ データ合計 使用済みデータ メタデータが使用可能です メタデータ合計 使用されているメタデータ プールのブロックサイズ

オブジェクト：	識別子：	属性：	「 **dataPoints** 」：
Docker コンテナ	ネームスペース コンテナ名 Docker Engine の略	Kubernetesコンテナハッシュ Kubernetesコンテナポート Kubernetesコンテナの再起動数 Kubernetes Container Termination Message Pathの略 Kubernetes Container Termination Message Policyの略 Kubernetesポッド終了の猶予期間 コンテナイメージ コンテナのステータス コンテナバージョン ノード名 Kubernetesコンテナログパス Kubernetesコンテナ名 Kubernetes Dockerタイプ Kubernetesポッド名 Kubernetesポッドネームスペース KubernetesポッドUID Kubernetesサンドボックス ID ノードIP ノード UUID Dockerバージョン Kubernetes IO設定を確認しました Kubernetes IO構成ソース OpenShift IO SCC Kubernetes概要の略 Kubernetesの表示名 OpenShiftタグ Komposeサービス ポッドテンプレートハッシュ コントローラリビジョンハッシュ ポッドテンプレート生成 使用許諾 スキーマビルド日 スキーマライセンス スキーマ名 スキーマURL スキーマVCS URL スキーマベンダー スキーマバージョン スキーマスキーマバージョン	Memory Active Anonymousの略 メモリアクティブファイル メモリキャッシュ メモリ階層の制限 メモリ非アクティブ匿名 メモリ非アクティブファイル メモリ制限 Memory Mapped Fileの略 Memory Max Usageの略 メモリページ障害 メモリページの重大な障害 メモリがページインされました メモリがページアウトされました メモリ常駐設定サイズ メモリ常駐セットサイズが大きすぎます Memory Total Active Anonymousの略 Memory Total Active File の略 メモリ合計キャッシュ Memory Total Inactive Anonymousの略 Memory Total Inactive File の略 Memory Total Mapped File の略 Memory Total Page Fault の略 Memory Total Page Major Faultの略 ページインされたメモリの合計 ページアウトされたメモリの合計 Memory Total Resident Set Sizeの略 メモリ合計常駐セットサイズが大きすぎます Memory Total Unevictable の略 Memory Unevictable（アクセス不能メモリ）の略 メモリ使用量 メモリ使用率 終了コード ウームは殺された ピッド

オブジェクト：	識別子：	属性：	「 **dataPoints** 」：
Docker コンテナブロック IO	ネームスペース コンテナ名 デバイス Docker Engine の略	Kubernetesコンテナハッシュ Kubernetesコンテナポート Kubernetesコンテナの再起動数 Kubernetes Container Termination Message Pathの略 Kubernetes Container Termination Message Policyの略 Kubernetesポッド終了の猶予期間 コンテナイメージ コンテナのステータス コンテナバージョン ノード名 Kubernetesコンテナログパス Kubernetesコンテナ名 Kubernetes Dockerタイプ Kubernetesポッド名 Kubernetesポッドネームスペース KubernetesポッドUID KubernetesサンドボックスID ノードIP ノード UUID Dockerバージョン Kubernetes Configを確認 Kubernetes構成ソース OpenShift SCC Kubernetes概要の略 Kubernetesの表示名 OpenShiftタグ スキーマスキーマバージョン ポッドテンプレートハッシュ コントローラリビジョンハッシュ ポッドテンプレート生成 Komposeサービス スキーマビルド日 スキーマライセンス スキーマ名 スキーマベンダー 顧客ポッド Kubernetes StatefulSetポッド名 テナント WebConsoleの略	IO Service Bytes Recursive Asyncの略 IO Service Bytes Recursive Readの略 IO Service Bytes Recursive Syncの略 IOサービスバイト数の再帰的合計 IO Service Bytes Recursive Writeの略 IO Serviced Recursive Asyncの略 IO Serviced Recursive Readの略 IO Serviced Recursive Syncの略 IOサービス再帰合計 IO Serviced Recursive Writeの略

オブジェクト：	識別子：	属性：	「 **dataPoints** 」：
Docker コンテナネットワーク	ネームスペース コンテナ名 ネットワーク Docker Engine の略	コンテナイメージ コンテナのステータス コンテナバージョン ノード名 ノードIP ノード UUID ノードOS Kubernetesクラスタ Dockerバージョン コンテナID	RXがドロップされました Rxバイト Rxエラー Rxパケット 送信がドロップされました 送信バイト数 TXエラー 送信パケット数

オブジェクト：	識別子：	属性：	「 **dataPoints** 」：
Docker コンテナの CPU	ネームスペース コンテナ名 CPU Docker Engine の略	Kubernetesコンテナハッシュ Kubernetesコンテナポート Kubernetesコンテナの再起動数 Kubernetes Container Termination Message Pathの略 Kubernetes Container Termination Message Policyの略 Kubernetesポッド終了の猶予期間 Kubernetes Configを確認 Kubernetes構成ソース OpenShift SCC コンテナイメージ コンテナのステータス コンテナバージョン ノード名 Kubernetesコンテナログパス Kubernetesコンテナの名前 Kubernetes Dockerタイプ Kubernetesポッド名 Kubernetesポッドネームスペース KubernetesポッドUID Kubernetesサンドボックス ID ノードIP ノード UUID ノードOS Kubernetesクラスタ Dockerバージョン Kubernetes概要の略 Kubernetesの表示名 OpenShiftタグ スキーマバージョン ポッドテンプレートハッシュ コントローラリビジョンハッシュ ポッドテンプレート生成 Komposeサービス スキーマビルド日 スキーマライセンス スキーマ名 スキーマベンダー 顧客ポッド Kubernetes StatefulSetポッド名	スロットリング期間 調整された期間の調整 スロットル調整時間 カーネルモードでの使用 ユーザーモードでの使用 方法 使用率 使用システム 使用量の合計

トラブルシューティング

問題	次の操作を実行します
設定ページの手順に従っても、Data Infrastructure InsightsにDocker指標が表示されません。	Telegrafエージェントログで、次のエラーが報告されているかどうかを確認します。 来い! プラグイン[inputs.docker]のエラー：Dockerデーモンソケットに接続しようとしているときに権限が拒否されました もしそうなら、上記のようにTelegrafエージェントがDocker Unixソケットにアクセスできるようにするために必要な手順を実行します。

追加情報はから入手できます "サポート" ページ

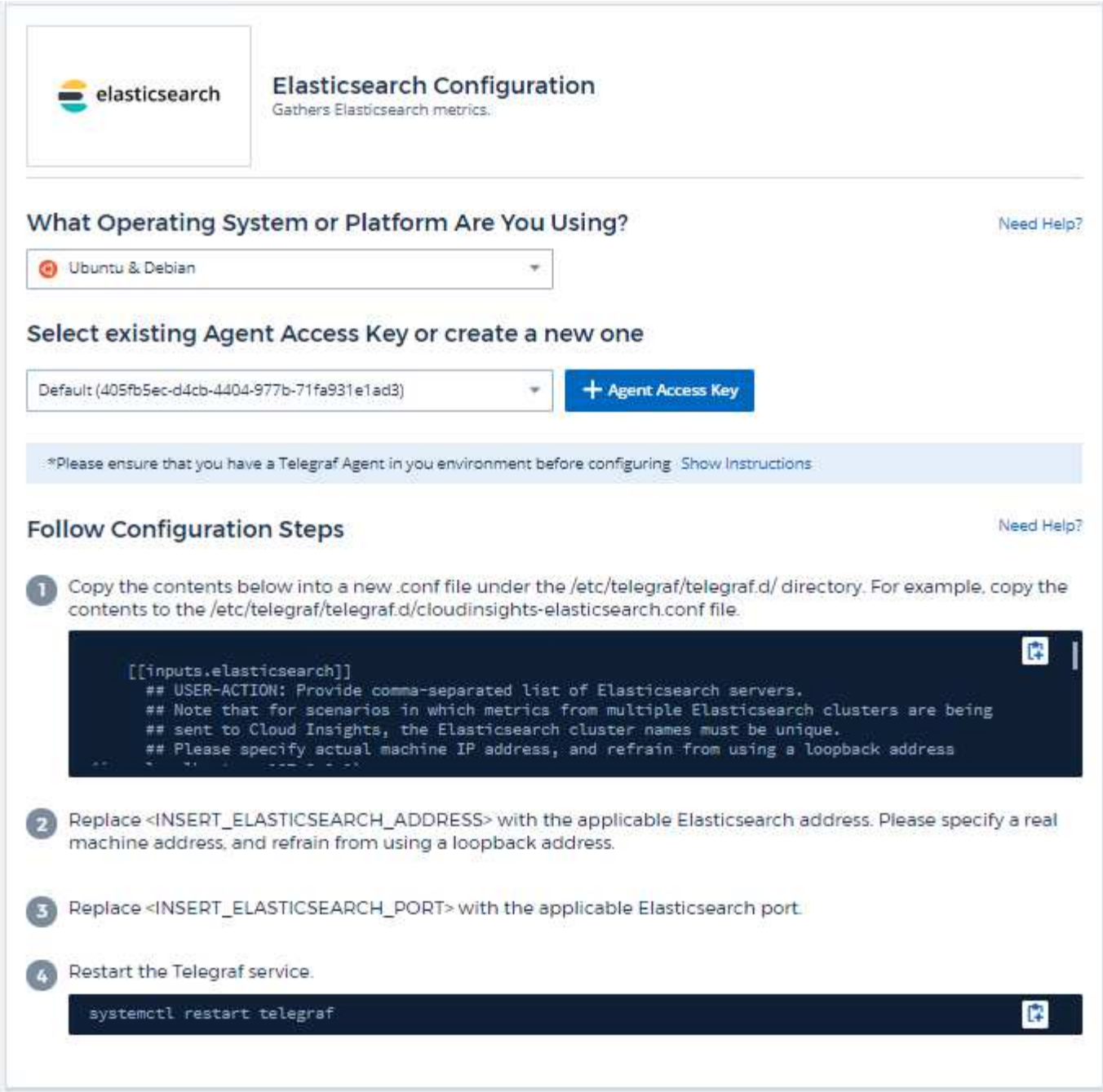
Elasticsearch Data Collector

Data Infrastructure Insightsは、このデータコレクタを使用してElasticsearchから指標を収集します。

1. [Observability]>[Collectors]で、+ Data Collector *をクリックします。[Elasticsearch]を選択します。

   Tegraf エージェントがインストールされているオペレーティングシステムまたはプラットフォームを選択します。

2. Agent for Collection をまだインストールしていない場合、または別のオペレーティングシステムまたはプラットフォームに Agent をインストールする場合は、 *Show Instructions* をクリックしてを展開します "エージェントのインストール" 手順
3. このデータコレクタで使用するエージェントアクセスキーを選択します。[**+Agent Access Key**] ボタンをクリックすると、新しいエージェントアクセスキーを追加できます。ベストプラクティス：別のエージェントアクセスキーを使用するのは、たとえば OS/ プラットフォーム別にデータコレクタをグループ化する場合だけです。
4. 設定手順に従ってデータコレクタを設定します。手順は、データの収集に使用するオペレーティングシステムまたはプラットフォームのタイプによって異なります。



セットアップ（ **Setup** ）

情報は、に記載されています "Elasticsearch のドキュメント"。

オブジェクトとカウンタ

次のオブジェクトとそのカウンタが収集されます。

オブジェクト：	識別子：	属性：	「 **dataPoints** 」：
Elasticsearch クラスタ	ネームスペース クラスタ	ノードIP ノード名 クラスタのステータス	マスターノード数 合計ノード数 ファイルシステムの使用可能データ（バイト） ファイルシステムの空き容量（バイト） ファイルシステムデータ合計（バイト） JVMスレッド OS割り当て済みのプロセッサ OS利用可能なプロセッサ OSメモリ空き容量（バイト） OSメモリフリー OSメモリ合計（バイト） 使用済みOSメモリ（バイト） 使用されているOSメモリ プロセスCPU インデックス完了サイズ（バイト） インデックス数 インデックスドキュメント数 インデックスドキュメントが削除されました Indicesフィールドデータの削除 インデックスフィールドデータメモリサイズ（バイト） インデックスクエリキャッシュ数 インデックスキャッシュサイズ Indices Segments Countの略 インデックスセグメントドキュメント値メモリ（バイト） インデックスシャードインデックスプライマリー平均 インデックスシャードインデックスプライマリー最大 Indices shards Index Primaries Min インデックスシャードインデックスレプリケーション平均 インデックスシャードインデックスレプリケーシ

オブジェクト：	識別子：	属性：	「 **dataPoints** 」：
Elasticsearch ノード	ネームスペース クラスタ ESノードID ESノードIP ESノード	ゾーン ID	機械学習が有効になりました 機械学習メモリ 機械学習の最大オープンジョブ数 X-Packがインストールされています ブレーカーアカウンティング推定サイズ（バイト） ブレーカーアカウンティング制限サイズ（バイト） ブレーカー会計オーバーヘッド ブレーカー会計が作動しました ブレーカーフィールドデータ推定サイズ（バイト） ブレーカーフィールドのデータ制限サイズ（バイト） ブレーカフィールドデータオーバーヘッド ブレーカーフィールドデータがトリップしました ブレーカの飛行中の推定サイズ（バイト） ブレーカインフライト制限サイズ（バイト） ブレーカインフライトオーバーヘッド ブレーカインフライトが停止しました ブレーカ親推定サイズ（バイト） ブレーカ親制限サイズ（バイト） Breakers親の頭上 ブレーカの親がトリップしました ブレーカー要求推定サイズ（バイト） ブレーカー要求制限サイズ（バイト） ブレーカー要求オーバーヘッド ブレーカー要求が作動しました ファイルシステムの使用可能データ（バイト） ファイルシステムの空き容量（バイト）

トラブルシューティング

追加情報はから入手できます "サポート" ページ

Flink Data Collector の使用

Data Infrastructure Insightsは、このデータコレクタを使用してFlinkから指標を収集します。

インストール

1. [Observability]>[Collectors]で、+ Data Collector *をクリックします。[Flink]を選択します。

   Tegraf エージェントがインストールされているオペレーティングシステムまたはプラットフォームを選択します。

2. Agent for Collection をまだインストールしていない場合、または別のオペレーティングシステムまたはプラットフォームに Agent をインストールする場合は、 *Show Instructions* をクリックしてを展開します "エージェントのインストール" 手順

3. このデータコレクタで使用するエージェントアクセスキーを選択します。[**+Agent Access Key**] ボタンをクリックすると、新しいエージェントアクセスキーを追加できます。ベストプラクティス：別のエージェントアクセスキーを使用するのは、たとえば OS/ プラットフォーム別にデータコレクタをグループ化する場合だけです。

4. 設定手順に従ってデータコレクタを設定します。手順は、データの収集に使用するオペレーティングシステムまたはプラットフォームのタイプによって異なります。

Flink

Flink Configuration

Gathers Flink metrics.

What Operating System or Platform Are You Using?

Need Help?

Windows

Select existing Agent Access Key or create a new one

Default (405fb5ec-d4cb-4404-977b-71fa931e1ad3)

+ Agent Access Key

*Please ensure that you have a Telegraf Agent in you environment before configuring Show Instructions

Follow Configuration Steps

Need Help?

1. Install Jolokia on your Flink JobManager(s) and Flink Task Manager(s). For details refer to the following document.
2. Copy the contents below into a new .conf file under the C:\Program Files\telegraf\telegraf.d\ folder. For example, copy the contents to the C:\Program Files\telegraf\telegraf.d\cloudinsights-flink.conf file.

```
## ##########################
## JobManager
## ##########################
[[inputs.jolokia2_agent]]
  ## USER-ACTION: Provide address(es) of flink Job Manager(s), port for jolokia, add one URL
```

3. Replace <INSERT_FLINK_JOBMANAGER_ADDRESS> with the applicable Flink Job Manager address(es). Please specify a real machine address, and refrain from using a loopback address.
4. Replace <INSERT_FLINK_TASKMANAGER_ADDRESS> with the applicable Flink Task Manager address(es). Please specify a real machine address, and refrain from using a loopback address.
5. Replace <INSERT_JOLOKIA_PORT> with the applicable jolokia port.
6. Modify 'Namespace' if needed for server disambiguation (to avoid name clashes).
7. Modify 'Cluster' if needed for Flink cluster designation.
8. Restart the Telegraf service.

```
Stop-Service -Name telegraf -ErrorAction SilentlyContinue; Start-Service -Name telegraf
```

セットアップ（ **Setup** ）

フルリンク展開では、次のコンポーネントが使用されます。

JobManager ： Flink プライマリシステム。一連の TaskManager を調整しますハイアベイラビリティ設定では、システムに複数の JobManager が存在します。
TaskManager ： Flink 演算子が実行される場所です。
Flink プラグインは、テレグラムの Jolokia プラグインに基づいています。すべての Flink コンポーネントから情報を収集するための要件など、 JMX はすべてのコンポーネントで Jolokia を介して設定および公開する必要があります。

互換性

Flink バージョン 1.7.0 に対して設定が開発されました。

セットアップ中です

Jolokia エージェント JAR

個々のコンポーネントについては、 Jolokia エージェント jar ファイルのバージョンをダウンロードする必要があります。テスト対象のバージョンは、でした "Jolokia 代理店 1.6.0"。

以下の手順では、ダウンロードした jar ファイル（ jolokia-jvm-1.6.0-agent.jar ）が「 /opt/Flink/lib/' 」の下に配置されると想定しています。

JobManager

JobManager で Jolokia API を公開するように設定するには、ノードで次の環境変数を設定して JobManager を再起動します。

```
 export FLINK_ENV_JAVA_OPTS="-javaagent:/opt/flink/lib/jolokia-jvm-1.6.0-
agent.jar=port=8778,host=0.0.0.0"
Jolokia （ 8778 ）には別のポートを選択できます。Jolokia をロックする内部 IP
を持っている場合は、「 catch all 」 0.0.0.0 を自分の IP
で置き換えることができます。この IP
には、テレグラムプラグインからアクセスできる必要があります。
```

TaskManager の略

Jolokia API を公開するように TaskManager を設定するには ' ノードに次の環境変数を設定し 'TaskManager を再起動します

```
 export FLINK_ENV_JAVA_OPTS="-javaagent:/opt/flink/lib/jolokia-jvm-1.6.0-
agent.jar=port=8778,host=0.0.0.0"
Jolokia （ 8778 ）には別のポートを選択できます。Jolokia をロックする内部 IP
を持っている場合は、「 catch all 」 0.0.0.0 を自分の IP
で置き換えることができます。この IP
には、テレグラムプラグインからアクセスできる必要があります。
```

オブジェクトとカウンタ

次のオブジェクトとそのカウンタが収集されます。

オブジェクト：	識別子：	属性：	「 **dataPoints** 」：
Flink タスクマネージャ	クラスタ ネームスペース サーバ	ノード名 タスクマネージャID ノードIP	Network Available Memory Segmentsの略 Network Total Memory Segmentsの略 ガベージコレクションPS MarkSweep数 ガベージコレクションPS MarkSweep Timeの略 ガベージコレクションPS スカベンジ数 ガベージコレクションPS スカベンジ時間 ヒープメモリがコミットされました ヒープメモリの初期化 ヒープメモリ最大 ヒープメモリ使用済み スレッドカウントデーモン スレッド数のピーク スレッド数 スレッド数合計が開始されました
フリンクジョブ	クラスタ ネームスペース サーバ ジョブ ID	ノード名 ジョブ名 ノードIP Last Checkpoint External Pathの略 再起動時間	ダウンタイム 完全に再起動します 前回のチェックポイントアライメントがバッファリングされました 前回のチェックポイント期間 前回のチェックポイントサイズ 完了したチェックポイントの数 失敗したチェックポイントの数 進行中のチェックポイントの数 チェックポイントの数 アップタイム

オブジェクト：	識別子：	属性：	「 **dataPoints** 」：
Flink ジョブマネージャー	クラスタ ネームスペース サーバ	ノード名 ノードIP	ガベージコレクションPS MarkSweep数 ガベージコレクションPS MarkSweep Timeの略 ガベージコレクションPS スカベンジ数 ガベージコレクションPS スカベンジ時間 ヒープメモリがコミットされました ヒープメモリの初期化 ヒープメモリ最大 ヒープメモリ使用済み 登録されているタスクマネージャの数 実行中のジョブの数 使用可能なタスクスロット タスクスロットの合計 スレッドカウントデーモン スレッド数のピーク スレッド数 スレッド数合計が開始されました

オブジェクト：	識別子：	属性：	「 **dataPoints** 」：
Flink タスク	クラスタ ネームスペース ジョブ ID タスク ID	サーバ ノード名 ジョブ名 サブタスクインデックス タスク試行ID タスク試行番号 タスク名 タスクマネージャID ノードIP Current Input Watermark の略	Pool Usageのバッファ Buffers in Queue Length の略 Buffers Out Pool Usageの略 バッファアウトキュー長 ローカルのバッファ数 Number Buffers in Local Per Secondカウント ローカル/秒レートのバッファ数 リモートのNumber Buffers（バッファ数） Number Buffers in Remote Per Second（リモート/秒）カウント Number Buffers in Remote Per Second Rate（リモート/秒レート） Number Buffers Outの略 Number Buffers Out Per Secondカウント Number Buffers Out Per Second Rateの略 ローカルのバイト数 1秒あたりのローカルバイト数 ローカル/秒レートのバイト数 リモートのバイト数 1秒あたりのリモートバイト数 Remote Per Second Rate のバイト数 送信されたバイト数 Number Bytes Out Per Second Count（1秒 1秒あたりの送信バイト数レート のレコード数 1秒あたりのレコード数 1秒あたりのレコード数 レコード数が出ている Number Records Out Per Second Countの略 Number Records Out Per Second Rateの略

オブジェクト：	識別子：	属性：	「 **dataPoints** 」：
Flink タスクオペレータ	クラスタ ネームスペース ジョブ ID オペレータID タスク ID	サーバ ノード名 ジョブ名 演算子名 サブタスクインデックス タスク試行ID タスク試行番号 タスク名 タスクマネージャID ノードIP	Current Input Watermark の略 Current Output Watermark の略 のレコード数 1秒あたりのレコード数 1秒あたりのレコード数 レコード数が出ている Number Records Out Per Second Countの略 Number Records Out Per Second Rateの略 遅延レコード数がドロップされました 割り当て済みパーティション Bytes Consumed Rate コミットレイテンシの平均 コミットレイテンシ最大 コミット率 コミットに失敗しました コミットに成功しました 接続完了率 接続数 接続作成レート カウント フェッチレイテンシの平均 フェッチレイテンシの最大値 フェッチレート 取得サイズ平均 フェッチサイズ最大 フェッチスロットル時間平均 フェッチスロットル時間最大 ハートビートレート 受信バイトレート IO比率 IO時間の平均（ns） IO待機比率 IO待機時間の平均（ns） 参加率 ジョイン時間平均 前回のハートビート前 Network IO Rateの略 Outgoing Byte Rateの略 レコード消費率 最大遅延レコード リクエストあたりのレコード平均 リクエスト率

トラブルシューティング

追加情報はから入手できます "サポート" ページ

Hadoop Data Collector

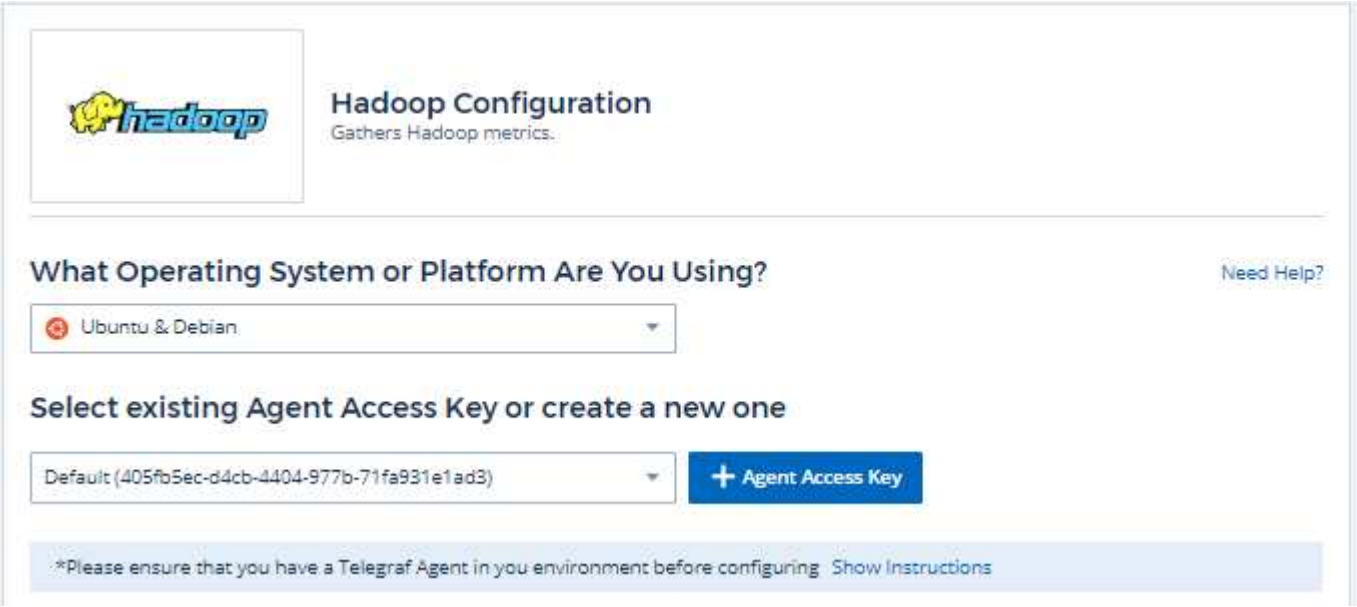
Data Infrastructure Insightsは、このデータコレクタを使用してHadoopから指標を収集します。

インストール

1. [Observability]>[Collectors]で、+ Data Collector *をクリックします。Hadoopを選択します。

   Tegraf エージェントがインストールされているオペレーティングシステムまたはプラットフォームを選択します。

2. Agent for Collection をまだインストールしていない場合、または別のオペレーティングシステムまたはプラットフォームに Agent をインストールする場合は、 *Show Instructions* をクリックしてを展開します "エージェントのインストール" 手順
3. このデータコレクタで使用するエージェントアクセスキーを選択します。[**+Agent Access Key**] ボタンをクリックすると、新しいエージェントアクセスキーを追加できます。ベストプラクティス：別のエージェントアクセスキーを使用するのは、たとえば OS/ プラットフォーム別にデータコレクタをグループ化する場合だけです。
4. 設定手順に従ってデータコレクタを設定します。手順は、データの収集に使用するオペレーティングシステムまたはプラットフォームのタイプによって異なります。



Follow Configuration Steps

Need Help?

1. Install Jolokia on your Hadoop NameNode, Secondary NameNode, DataNode(s), ResourceManager, NodeManager(s) and JobHistoryServer. For details refer to the following document.

2. Copy the contents below into a new .conf file under the /etc/telegraf/telegraf.d/ directory. For example, copy the contents to the /etc/telegraf/telegraf.d/cloudinsights-hadoop.conf file.

```
################
# NAMENODE     #
################
[[inputs.jolokia2_agent]]
  ## USER-ACTION: Provide address(es) of Hadoop NameNode, port for jolokia
```

3. Replace <INSERT_HADOOP_NAMENODE_ADDRESS> with the applicable Hadoop NameNode address. Please specify a real machine address, and refrain from using a loopback address. Replace corresponding <INSERT_JOLOKIA_PORT> with the NameNode's assigned Jolokia port.

4. Replace <INSERT_HADOOP_SECONDARYNAMENODE_ADDRESS> with the applicable Hadoop Secondary NameNode address. Please specify a real machine address, and refrain from using a loopback address. Replace corresponding <INSERT_JOLOKIA_PORT> with the Secondary NameNode's assigned Jolokia port.

5. Replace <INSERT_HADOOP_DATANODE_ADDRESS> with the applicable Hadoop DataNode address(es). Please specify a real machine address, and refrain from using a loopback address. Replace corresponding <INSERT_JOLOKIA_PORT> with the DataNode's assigned Jolokia port.

6. Replace <INSERT_HADOOP_RESOURCEMANAGER_ADDRESS> with the applicable Hadoop ResourceManager address. Please specify a real machine address, and refrain from using a loopback address. Replace corresponding <INSERT_JOLOKIA_PORT> with the ResourceManager's assigned Jolokia port.

7. Replace <INSERT_HADOOP_NODEMANAGER_ADDRESS> with the applicable Hadoop NodeManager address(es). Please specify a real machine address, and refrain from using a loopback address. Replace corresponding <INSERT_JOLOKIA_PORT> with the NodeManager's assigned Jolokia port.

8. Replace <INSERT_HADOOP_JOBHISTORYSERVER_ADDRESS> with the applicable Hadoop Job History Server address. Please specify a real machine address, and refrain from using a loopback address. Replace corresponding <INSERT_JOLOKIA_PORT> with the Job History Server's assigned Jolokia port.

9. Modify 'Namespace' if needed for server disambiguation (to avoid name clashes).

10. Modify 'Cluster' if needed for Hadoop cluster designation.

11. Restart the Telegraf service.

```
systemctl restart telegraf
```

セットアップ（ **Setup** ）

Hadoop を完全に導入するには、次のコンポーネントが必要です。

- NameNode ： Hadoop 分散型ファイルシステム（ HDFS ）プライマリシステム。一連の DataNode を調

整します。

- セカンダリ NameNode ：メイン NameNode のウォームフェイルオーバー。Hadoop では、 NameNode への昇格は自動的には行われません。セカンダリ NameNode は、必要に応じてプロモート可能な状態にするために、 NameNode から情報を収集します。
- DataNode ：データの実際の所有者。
- ResourceManager ：コンピューティングのプライマリシステム（ yarn ）。一連の NodeManager を調整します。
- NodeManager ：コンピューティング用のリソース。アプリケーションを実行するための実際の場所。
- JobHistoryServer ：ジョブ履歴に関連するすべての要求の処理を担当します。

Hadoop プラグインは、テレグラムの Jolokia プラグインに基づいています。すべての Hadoop コンポーネントから情報を収集するための要件など、 JMX はすべてのコンポーネントで Jolokia 経由で設定および公開する必要があります。

互換性

構成は Hadoop バージョン 2.9.2 に対して開発されました。

セットアップ中です

Jolokia エージェント JAR

個々のコンポーネントについては、 Jolokia エージェント jar ファイルのバージョンをダウンロードする必要があります。テスト対象のバージョンは、でした "Jolokia 代理店 1.6.0"。

以下の手順では、ダウンロードした jar ファイル（ jolokia-jvm-1.6.0-agent.jar ）が「 /opt/hadoop /lib/' 」の下に配置されると想定しています。

NameNode

Jolokia API が公開されるように NameNode を設定するには、 <hadoop home>/etc/hadoop /hadoop -env.sh で次のセットアップを行います。

```
export HADOOP_NAMENODE_OPTS="$HADOOP_NAMENODE_OPTS
-javaagent:/opt/hadoop/lib/jolokia-jvm-1.6.0
-agent.jar=port=7800,host=0.0.0.0 -Dcom.sun.management.jmxremote
-Dcom.sun.management.jmxremote.port=8000
-Dcom.sun.management.jmxremote.ssl=false
-Dcom.sun.management.jmxremote.password.file=$HADOOP_HOME/conf/jmxremote.p
assword"
You can choose a different port for JMX (8000 above) and Jolokia (7800).
If you have an internal IP to lock Jolokia onto you can replace the "catch
all" 0.0.0.0 by your own IP. Notice this IP needs to be accessible from
the telegraf plugin. You can use the option '-
Dcom.sun.management.jmxremote.authenticate=false' if you don't want to
authenticate. Use at your own risk.
```

セカンダリ NameNode

セカンダリ NameNode で Jolokia API を公開するように設定するには、 <hadoop home>/etc/hadoop /hadoop -env.sh で次のように設定します。

```
export HADOOP_SECONDARYNAMENODE_OPTS="$HADOOP_SECONDARYNAMENODE_OPTS
-javaagent:/opt/hadoop/lib/jolokia-jvm-1.6.0
-agent.jar=port=7802,host=0.0.0.0 -Dcom.sun.management.jmxremote
-Dcom.sun.management.jmxremote.port=8002
-Dcom.sun.management.jmxremote.ssl=false
-Dcom.sun.management.jmxremote.password.file=$HADOOP_HOME/conf/jmxremote.p
assword"
You can choose a different port for JMX (8002 above) and Jolokia (7802).
If you have an internal IP to lock Jolokia onto you can replace the "catch
all" 0.0.0.0 by your own IP. Notice this IP needs to be accessible from
the telegraf plugin. You can use the option '-
Dcom.sun.management.jmxremote.authenticate=false' if you don't want to
authenticate. Use at your own risk.
```

DataNode

Jolokia API が公開されるように DataNode を設定するには、 <hopp_home>/etc/hadoop /hadoop -env.sh に以下のセットアップを行います。

```
export HADOOP_DATANODE_OPTS="$HADOOP_DATANODE_OPTS
-javaagent:/opt/hadoop/lib/jolokia-jvm-1.6.0
-agent.jar=port=7801,host=0.0.0.0 -Dcom.sun.management.jmxremote
-Dcom.sun.management.jmxremote.port=8001
-Dcom.sun.management.jmxremote.ssl=false
-Dcom.sun.management.jmxremote.password.file=$HADOOP_HOME/conf/jmxremote.p
assword"
You can choose a different port for JMX (8001 above) and Jolokia (7801).
If you have an internal IP to lock Jolokia onto you can replace the "catch
all" 0.0.0.0 by your own IP. Notice this IP needs to be accessible from
the telegraf plugin. You can use the option '-
Dcom.sun.management.jmxremote.authenticate=false' if you don't want to
authenticate. Use at your own risk.
```

ResourceManager の略

ResourceManager で Jolokia API を公開するように設定するには、 <hadoop HOME>//etc/hadoop /hadoop -env.sh で次の設定を行うことができます。

```
export YARN_RESOURCEMANAGER_OPTS="$YARN_RESOURCEMANAGER_OPTS
-javaagent:/opt/hadoop/lib/jolokia-jvm-1.6.0
-agent.jar=port=7803,host=0.0.0.0 -Dcom.sun.management.jmxremote
-Dcom.sun.management.jmxremote.port=8003
-Dcom.sun.management.jmxremote.ssl=false
-Dcom.sun.management.jmxremote.password.file=$HADOOP_HOME/conf/jmxremote.p
assword"
You can choose a different port for JMX (8003 above) and Jolokia (7803).
If you have an internal IP to lock Jolokia onto you can replace the "catch
all" 0.0.0.0 by your own IP. Notice this IP needs to be accessible from
the telegraf plugin. You can use the option '-
Dcom.sun.management.jmxremote.authenticate=false' if you don't want to
authenticate. Use at your own risk.
```

NodeManager

Jolokia API を公開するように NodeManager を設定するには、 <hadoop home>/etc/hadoop /hadoop -env.sh で次の設定を行うことができます。

```
export YARN_NODEMANAGER_OPTS="$YARN_NODEMANAGER_OPTS
-javaagent:/opt/hadoop/lib/jolokia-jvm-1.6.0
-agent.jar=port=7804,host=0.0.0.0 -Dcom.sun.management.jmxremote
-Dcom.sun.management.jmxremote.port=8004
-Dcom.sun.management.jmxremote.ssl=false
-Dcom.sun.management.jmxremote.password.file=$HADOOP_HOME/conf/jmxremote.p
assword"
You can choose a different port for JMX (8004 above) and Jolokia (7804).
If you have an internal IP to lock Jolokia onto you can replace the "catch
all" 0.0.0.0 by your own IP. Notice this IP needs to be accessible from
the telegraf plugin. You can use the option '-
Dcom.sun.management.jmxremote.authenticate=false' if you don't want to
authenticate. Use at your own risk.
```

JobHistoryServer

JobHistoryServer で Jolokia API を公開するように設定するには、 <hadoop home>/etc/hadoop /hadoop -env.sh で次の設定を行うことができます。

```
export HADOOP_JOB_HISTORYSERVER_OPTS="$HADOOP_JOB_HISTORYSERVER_OPTS
-javaagent:/opt/hadoop/lib/jolokia-jvm-1.6.0
-agent.jar=port=7805,host=0.0.0.0 -Dcom.sun.management.jmxremote
-Dcom.sun.management.jmxremote.port=8005
-Dcom.sun.management.jmxremote.password.file=$HADOOP_HOME/conf/jmxremote.p
assword"
You can choose a different port for JMX (8005 above) and Jolokia (7805).
If you have an internal IP to lock Jolokia onto you can replace the "catch
all" 0.0.0.0 by your own IP. Notice this IP needs to be accessible from
the telegraf plugin. You can use the option '-
Dcom.sun.management.jmxremote.authenticate=false' if you don't want to
authenticate. Use at your own risk.
```

オブジェクトとカウンタ

次のオブジェクトとそのカウンタが収集されます。

オブジェクト：	識別子：	属性：	「 **dataPoints** 」：
Hadoop セカンダリ NameNode	クラスタ ネームスペース サーバ	ノード名 ノードIP コンパイル情報 バージョン	GCカウント GCコピー数 GCマークスイープコンパクトカウント GC番号情報のしきい値を超えました GC番号警告しきい値を超えました GC時間 GCコピー時間 GCマークスイープコンパクト時間 GC合計エクストラスリープ時間 エラー数をログに記録します ログ致命的数 ログ情報数 警告数をログに記録します メモリヒープがコミットされました メモリヒープ最大 使用されているメモリヒープ メモリ最大 メモリのヒープがコミットされていません メモリ非ヒープ最大 メモリのヒープが使用されていません ブロックされたスレッド スレッド新規 スレッド実行可能 スレッドが終了しました スレッドの待機時間 待機中のスレッド

オブジェクト：	識別子：	属性：	「 **dataPoints** 」：
Hadoop ノードマネージャ	クラスタ ネームスペース サーバ	ノード名 ノードIP	割り当て済みのコンテナ メモリ割り当て Memory Allocated Oportunisticの略 仮想コア割り当てオポチュニスティック 仮想コアが割り当てられました 使用可能なメモリ 使用可能な仮想コア ディレクトリがローカルではありません ディレクトリの不正なログ クリーニング前のキャッシュサイズ コンテナ起動時間平均時間 Container Launch Duration Number of Operations（コンテナ起動時間） コンテナが完了しました コンテナが失敗しました コンテナの初期化 コンテナを強制終了します コンテナをリリース コンテナの再構築 障害発生時にコンテナがロールバックされました 実行中のコンテナ ディスク使用率が良好なローカルディレクトリ Disk Utilization Good Log Directoriesの略 バイトがプライベート削除されました Bytes Deleted Public コンテナはOpportunityを実行しています 削除されたバイト数の合計 接続をシャッフルします 出力バイトをシャッフルします 出力のシャッフルに失敗しました シャッフル出力OK GCカウント GCコピー数 GCマークスイープコンパクトカウント GC番号情報のしきい値を

オブジェクト：	識別子：	属性：	「 **dataPoints** 」：
Hadoop ResourceManager	クラスタ ネームスペース サーバ	ノード名 ノードIP	ApplicationMaster起動遅延平均 ApplicationMaster起動遅延番号 ApplicationMaster登録遅延平均 ApplicationMaster Register Delay Numberの略 NodeManagerアクティブ番号 NodeManagerの運用停止番号 NodeManagerの運用停止番号 NodeManagerの番号が失われました NodeManagerがリブートしました NodeManagerシャットダウン番号 NodeManagerの正常な番号 NodeManagerのメモリ制限 NodeManager仮想コア数の制限 使用済み容量 アクティブアプリケーション アクティブユーザー 割り当て済みアグリゲートコンテナ アグリゲートコンテナがプリエンプトされました アグリゲートコンテナが解放されました アグリゲートメモリの秒数がプリエンプトされました 割り当て済みアグリゲートノードのローカルコンテナ アグリゲートオフスイッチコンテナの割り当て済み アグリゲートのAckローカルコンテナの割り当て済み容量 アグリゲート仮想コア（秒）がプリエンプトされました 割り当て済みのコンテナ 割り当てられたメモリ

オブジェクト：	識別子：	属性：	「 **dataPoints** 」：
Hadoop DataNode	クラスタ ネームスペース サーバ	ノード名 ノードIP クラスタID バージョン	トランシーバ数 送信が進行中です キャッシュ容量 使用されているキャッシュ 容量 使用されるDFS 推定損失合計容量 前回のボリューム障害率 キャッシュされた番号をブロックする ブロック番号のキャッシュに失敗しました ブロック番号のキャッシュ解除に失敗しました ボリューム番号に失敗しました 残り容量 GCカウント GCコピー数 GCマークスイープコンパクトカウント GC番号情報のしきい値を超えました GC番号警告しきい値を超えました GC時間 GCコピー時間 GCマークスイープコンパクト時間 GC合計エクストラスリープ時間 エラー数をログに記録します ログ致命的数 ログ情報数 警告数をログに記録します メモリヒープがコミットされました メモリヒープ最大 使用されているメモリヒープ メモリ最大 メモリのヒープがコミットされていません メモリ非ヒープ最大 メモリのヒープが使用されていません ブロックされたスレッド スレッド新規 スレッド実行可能 スレッドが終了しました スレッドの待機時間

オブジェクト：	識別子：	属性：	「 **dataPoints** 」：
Hadoop NameNode	クラスタ ネームスペース サーバ	ノード名 ノードIP 最後に書き込まれたトランザクションID 最後にロードされた編集からの時間 HAの状態 File System Stateの略 ブロックプールID クラスタID コンパイル情報 個別のバージョン数 バージョン	ブロック容量 合計ブロック数 合計容量 使用済み容量 Capacity Used Non DFSの略 ブロックが壊れています 推定損失合計容量 超過をブロックします ハートビートの期限が切れました ファイル合計 File System Lock Queue Lengthの略 ブロックが見つかりません 第1因子のレプリケーションが欠落しているブロック クライアントがアクティブです データノードが故障しています 稼働を停止しているデータノードが故障しています ライブを運用停止するデータノード データノードの運用停止 暗号化ゾーン番号 データノードがメンテナンスに移行しています 作成中のファイル メンテナンス中のデータノードが停止しました データノードはメンテナンス中です Data Nodes Liveの略 ストレージが古い レプリケーション保留タイムアウト データノードメッセージが保留中です 削除を保留中のブロック レプリケーションを保留中のブロック ミスレプリケートされたブロックが延期されました スケジュールされたレプリケーションをブロックします Snapshot スナップショット可能な

オブジェクト：	識別子：	属性：	「 **dataPoints** 」：
Hadoop ジョブ履歴サーバ	クラスタ ネームスペース サーバ	ノード名 ノードIP	GCカウント GCコピー数 GCマークスイープコンパクトカウント GC番号情報のしきい値を超えました GC番号警告しきい値を超えました GC時間 GCコピー時間 GCマークスイープコンパクト時間 GC合計エクストラスリープ時間 エラー数をログに記録します ログ致命的数 ログ情報数 警告数をログに記録します メモリヒープがコミットされました メモリヒープ最大 使用されているメモリヒープ メモリ最大 メモリのヒープがコミットされていません メモリ非ヒープ最大 メモリのヒープが使用されていません ブロックされたスレッド スレッド新規 スレッド実行可能 スレッドが終了しました スレッドの待機時間 待機中のスレッド

トラブルシューティング

追加情報はから入手できます "サポート" ページ

HAProxy Data Collector

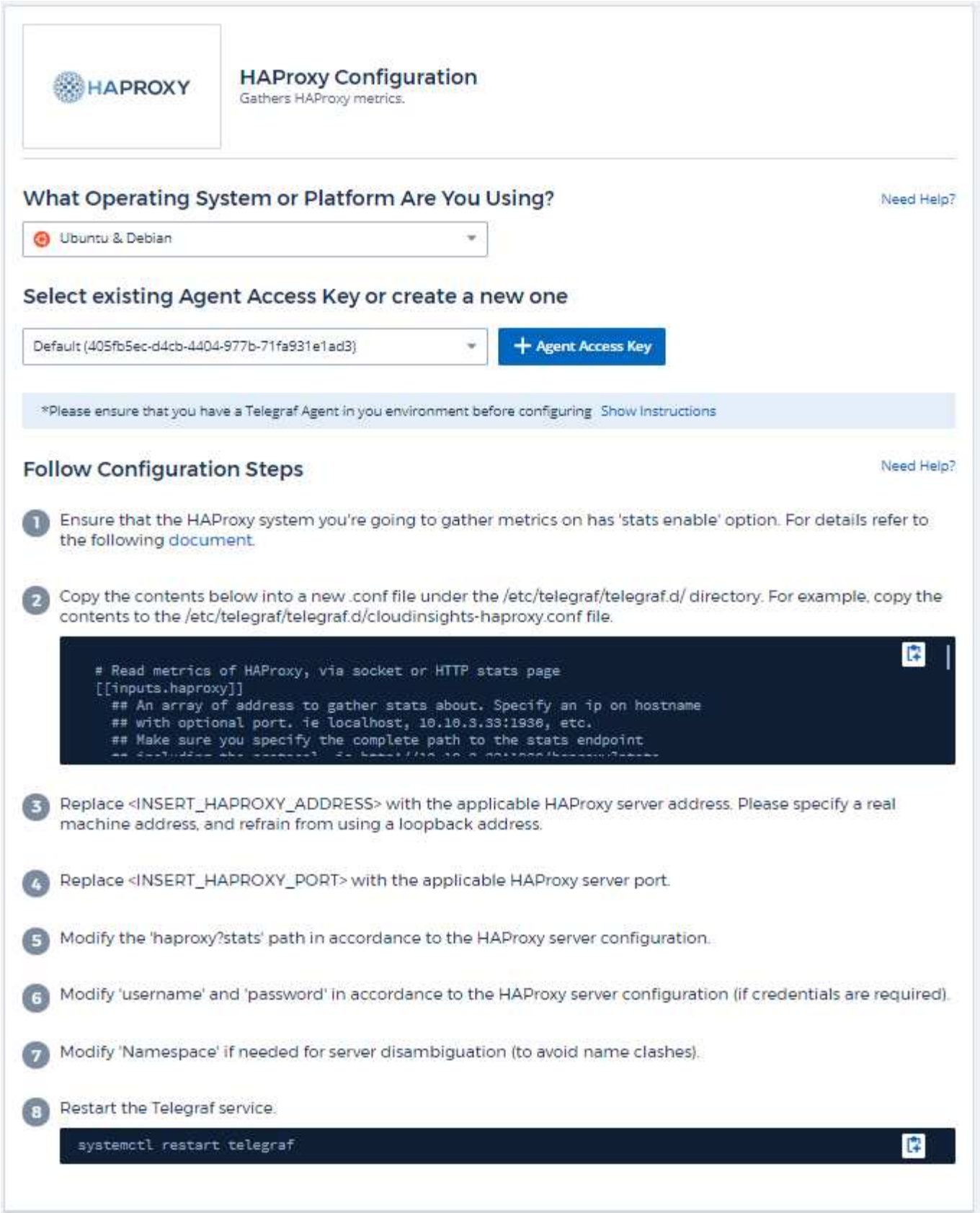
Data Infrastructure Insightsは、このデータコレクタを使用してHAProxyから指標を収集します。

インストール

1. [Observability]>[Collectors]で、+ Data Collector *をクリックします。[HAProxy]を選択します。

Tegraf エージェントがインストールされているオペレーティングシステムまたはプラットフォームを選択します。

2. Agent for Collection をまだインストールしていない場合、または別のオペレーティングシステムまたはプラットフォームに Agent をインストールする場合は、 *Show Instructions* をクリックしてを展開します "エージェントのインストール" 手順
3. このデータコレクタで使用するエージェントアクセスキーを選択します。[**+Agent Access Key**] ボタンをクリックすると、新しいエージェントアクセスキーを追加できます。ベストプラクティス：別のエージェントアクセスキーを使用するのは、たとえば OS/ プラットフォーム別にデータコレクタをグループ化する場合だけです。
4. 設定手順に従ってデータコレクタを設定します。手順は、データの収集に使用するオペレーティングシステムまたはプラットフォームのタイプによって異なります。



セットアップ（ **Setup** ）

Telegraf の HAProxy 用プラグインは、 HAProxy Stats の有効化に依存しています。これは HAProxy に組み込まれている構成ですが、すぐに有効にすることはできません。有効にすると 'HAProxy はブラウザで表示でき

る HTML エンドポイントを公開したり ' すべての HAProxy 構成のステータスを抽出するためにスクレイピングしたりできます

互換性：

構成は HAProxy バージョン 1.9.4 に対して開発されました。

設定：

統計情報を有効にするには、 haproxy 設定ファイルを編集し、「金庫」セクションの後に次の行を追加します。この行には、ユーザー自身のユーザー名とパスワード、および / または haproxy URL を使用します。

```
stats enable
stats auth myuser:mypassword
stats uri  /haproxy?stats
```

次に、 stats を有効にした単純な構成ファイルの例を示します。

```
global
  daemon
  maxconn 256

defaults
  mode http
  stats enable
  stats uri /haproxy?stats
  stats auth myuser:mypassword
  timeout connect 5000ms
  timeout client 50000ms
  timeout server 50000ms

frontend http-in
  bind *:80
  default_backend servers

frontend http-in9080
  bind *:9080
  default_backend servers_2

backend servers
  server server1 10.128.0.55:8080 check ssl verify none
  server server2 10.128.0.56:8080 check ssl verify none

backend servers_2
  server server3 10.128.0.57:8080 check ssl verify none
  server server4 10.128.0.58:8080 check ssl verify none
```

最新の手順の詳細については、を参照してください "HAProxy のドキュメント"。

オブジェクトとカウンタ

次のオブジェクトとそのカウンタが収集されます。

オブジェクト：	識別子：	属性：	「 **dataPoints** 」：
HAProxy フロントエンド	ネームスペース 住所 プロキシ	ノードIP ノード名 プロキシID モード プロセスID セッションレート制限 サーバID セッション制限 ステータス	受信バイト数 バイトアウト キャッシュヒット キャッシュルックアップ 圧縮バイトがバイパスされました 圧縮バイト数 圧縮バイトが送信されました 圧縮応答 接続レート 接続レート最大 接続合計 接続ルールによって拒否された要求 セキュリティ上の懸念により拒否されたリクエスト セキュリティ上の懸念により拒否された応答 セッションルールによって拒否された要求 エラーを要求します 応答1xx 応答は2xx 応答は3xx 応答は4xx 応答は5xx 応答その他 要求が傍受されました セッションレート セッションレート最大 リクエスト率 要求レート最大 リクエストの合計 セッション セッションの最大数 セッションの合計 書き換えを要求します

オブジェクト：	識別子：	属性：	「 **dataPoints** 」：
HAProxy サーバ	ネームスペース 住所 プロキシ サーバ	ノードIP ノード名 完了までの時間を確認します フォール設定を確認します 健全性の値を確認します 立ち上がり設定を確認します ステータスを確認します プロキシID 最終変更時刻 前回のセッション時間 モード プロセスID サーバID ステータス 重量	アクティブサーバ バックアップサーバ 受信バイト数 バイトアウト チェックダウン チェックに失敗しました クライアントが中止されました 接続 接続平均時間 ダウンタイムの合計 拒否された応答 接続エラー 応答エラー 応答1xx 応答は2xx 応答は3xx 応答は4xx 応答は5xx 応答その他 サーバ選択合計 キューの現在 キューの最大数 キュー平均時間 1秒あたりのセッション数 1秒あたりのセッションの最大数 接続の再利用 応答時間平均 セッション セッションの最大数 サーバ転送が中止されます セッションの合計 セッション合計時間平均 再ディスパッチを要求します 再試行を要求します 書き換えを要求します

オブジェクト：	識別子：	属性：	「 **dataPoints** 」：
HAProxy バックエンド	ネームスペース 住所 プロキシ	ノードIP ノード名 プロキシID 最終変更時刻 前回のセッション時間 モード プロセスID サーバID セッション制限 ステータス 重量	アクティブサーバ バックアップサーバ 受信バイト数 バイトアウト キャッシュヒット キャッシュルックアップ チェックダウン クライアントが中止されました 圧縮バイトがバイパスされました 圧縮バイト数 圧縮バイトが送信されました 圧縮応答 接続 接続平均時間 ダウンタイムの合計 セキュリティ上の懸念により拒否されたリクエスト セキュリティ上の懸念により拒否された応答 接続エラー 応答エラー 応答1xx 応答は2xx 応答は3xx 応答は4xx 応答は5xx 応答その他 サーバ選択合計 キューの現在 キューの最大数 キュー平均時間 1秒あたりのセッション数 1秒あたりのセッションの最大数 リクエストの合計 接続の再利用 応答時間平均 セッション セッションの最大数 サーバ転送が中止されます セッションの合計 セッション合計時間平均 再ディスパッチを要求します 再試行を要求します 書き換えを要求します

トラブルシューティング

追加情報はから入手できます "サポート" ページ

JVM Data Collector （ JVM データ収集）

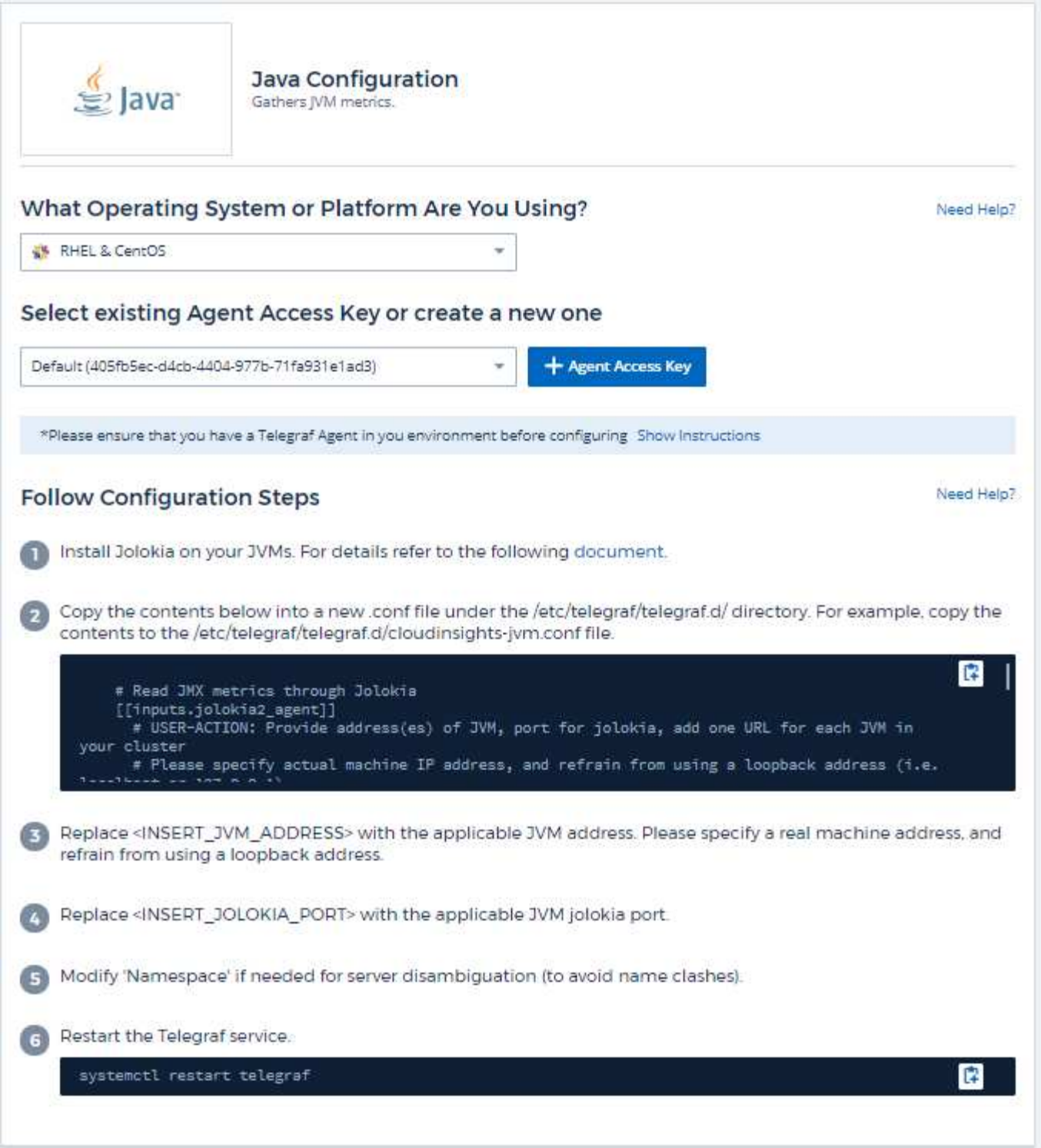
Data Infrastructure Insightsは、このデータコレクタを使用してJVMから指標を収集します。

インストール

1. [Observability]>[Collectors]で、+ Data Collector *をクリックします。JVMを選択します。

   Tegraf エージェントがインストールされているオペレーティングシステムまたはプラットフォームを選択します。

2. Agent for Collection をまだインストールしていない場合、または別のオペレーティングシステムまたはプラットフォームに Agent をインストールする場合は、 *Show Instructions* をクリックしてを展開します "エージェントのインストール" 手順
3. このデータコレクタで使用するエージェントアクセスキーを選択します。[**+Agent Access Key**] ボタンをクリックすると、新しいエージェントアクセスキーを追加できます。ベストプラクティス：別のエージェントアクセスキーを使用するのは、たとえば OS/ プラットフォーム別にデータコレクタをグループ化する場合だけです。
4. 設定手順に従ってデータコレクタを設定します。手順は、データの収集に使用するオペレーティングシステムまたはプラットフォームのタイプによって異なります。



セットアップ（ **Setup** ）

情報については、を参照してください "JVMのドキュメント"。

オブジェクトとカウンタ

次のオブジェクトとそのカウンタが収集されます。

オブジェクト：	識別子：	属性：	「 **dataPoints** 」：
JVM	ネームスペース JVM	OSアーキテクチャ OS名 OSバージョン ランタイム仕様 ランタイム仕様ベンダー ランタイム仕様バージョン アップタイム ランタイムVM名 ランタイムVMベンダー ランタイムVMバージョン ノード名 ノードIP	クラスが読み込まれました クラスロード合計 クラスがアンロードされました メモリヒープがコミットされました メモリヒープ初期化 使用メモリヒープの最大値 使用されているメモリヒープ メモリのヒープがコミットされていません メモリ非ヒープ初期化 メモリ非ヒープ最大 メモリのヒープが使用されていません Memory Objects Pending Finalizationの略 OSプロセッサが使用可能です OS Committed Virtual Memory Sizeの略 OS Free Physical Memory Sizeの略 OS Free Swap Space Size（OS空きスワップスペースサイズ） OS最大ファイル記述子数 OS Open File Descriptors Count（OSオープンファイル記述子数） OSプロセッサCPU負荷 OSプロセッサCPU時間 OSシステムCPU負荷 OSシステム負荷平均 OS合計物理メモリサイズ [OS Total Swap Space Size]をクリックします スレッドデーモン数 スレッドのピーク数 スレッド数 スレッド合計開始数 ガベージコレクタコピーコレクション数 ガベージコレクタのコピー収集時間 ガベージコレクタマーク-スイープ収集数 ガベージコレクタマークスイープ収集時間 ガベージコレクタG1旧世代コレクション数

トラブルシューティング

追加情報はから入手できます "サポート" ページ

Kafka Data Collector の利用

Data Infrastructure Insightsは、このデータコレクタを使用してKafkaから指標データを収集します。

インストール

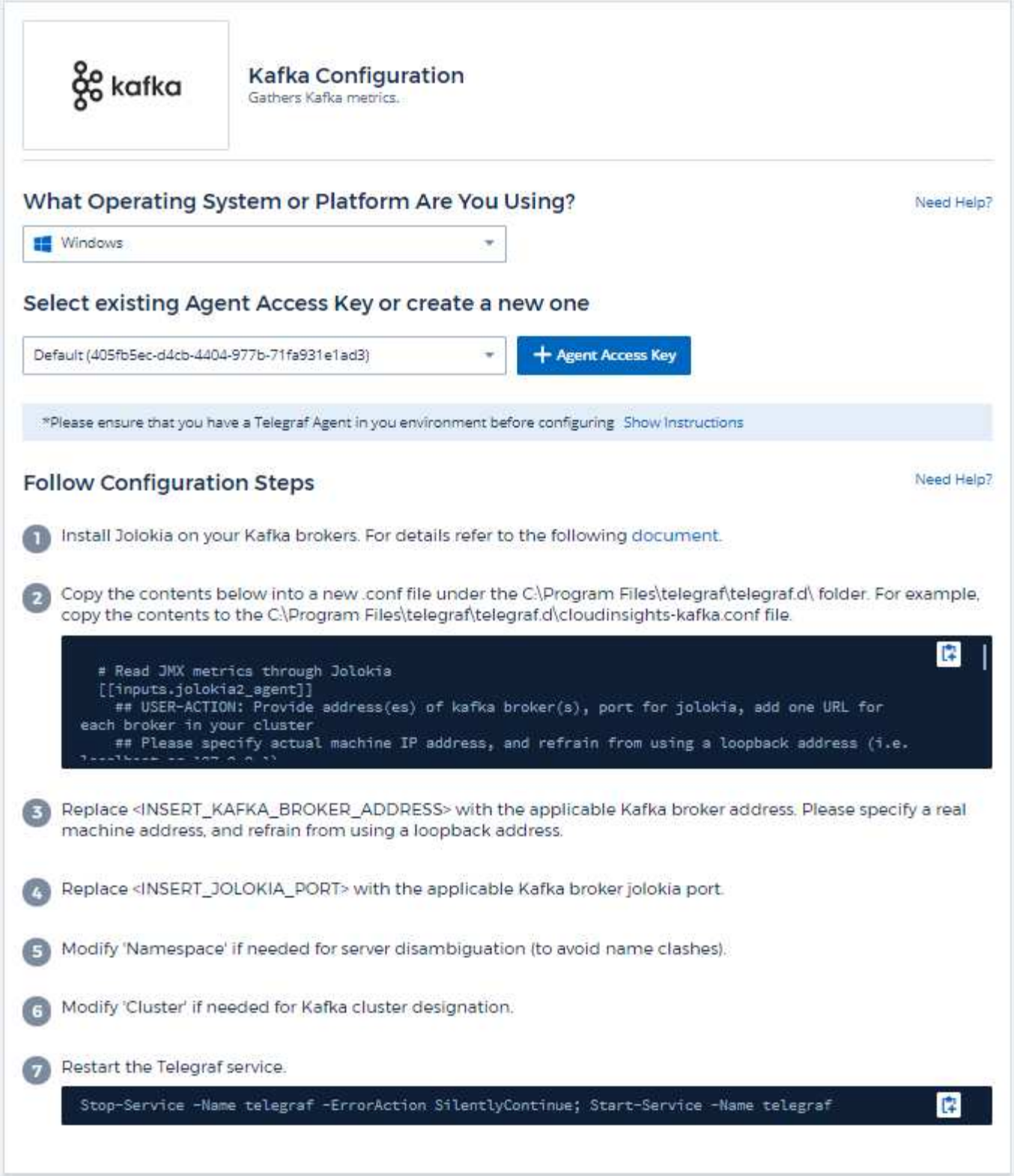
1. [Observability]>[Collectors]で、+ Data Collector *をクリックします。Kafkaを選択します。

   Tegraf エージェントがインストールされているオペレーティングシステムまたはプラットフォームを選択します。

2. Agent for Collection をまだインストールしていない場合、または別のオペレーティングシステムまたはプラットフォームに Agent をインストールする場合は、 *Show Instructions* をクリックしてを展開します "エージェントのインストール" 手順

3. このデータコレクタで使用するエージェントアクセスキーを選択します。[**+Agent Access Key**] ボタンをクリックすると、新しいエージェントアクセスキーを追加できます。ベストプラクティス：別のエージェントアクセスキーを使用するのは、たとえば OS/ プラットフォーム別にデータコレクタをグループ化する場合だけです。

4. 設定手順に従ってデータコレクタを設定します。手順は、データの収集に使用するオペレーティングシステムまたはプラットフォームのタイプによって異なります。



セットアップ（ **Setup** ）

Kafka プラグインは、テレグラムの Jolokia プラグインに基づいています。すべての Kafka ブローカーから情報を収集する必要があるため、 JMX はすべてのコンポーネントで Jolokia 経由で設定および公開する必要があります。

互換性

Kafka バージョン 0.11.0.2 の構成向けに開発されました。

セットアップ中です

以下の手順はすべて、 Kafka のインストール場所が「 /opt/Kafka 」であることを前提としています。以下の手順を使用して、設置場所を変更できます。

Jolokia エージェント JAR

Jolokia エージェント jar ファイルのバージョン "ダウンロードしました"。テスト対象のバージョンは Jolokia エージェント 1.6.0 でした。

以下の手順では、ダウンロードした jar ファイル（ jolokia-jvm-1.6.0-agent.jar ）が「 /opt/Kafka/libs/' 」の下に配置されると想定しています。

Kafka Brokers のようになります

Kafka Brokers で Jolokia API を公開するように設定するには、「 Kafka -run-class.sh 」コールの直前に、<Kafka_home>/bin/Kafka-server-start.sh に次の項目を追加します。

```
export JMX_PORT=9999
export RMI_HOSTNAME=`hostname -I`
export KAFKA_JMX_OPTS="-javaagent:/opt/kafka/libs/jolokia-jvm-1.6.0-
agent.jar=port=8778,host=0.0.0.0
-Dcom.sun.management.jmxremote.password.file=/opt/kafka/config/jmxremote.p
assword -Dcom.sun.management.jmxremote.ssl=false
-Djava.rmi.server.hostname=$RMI_HOSTNAME
-Dcom.sun.management.jmxremote.rmi.port=$JMX_PORT"
```

上記の例では 'hostname-i' を使用して 'RMI_HOSTNAME' 環境変数を設定しています。複数の IP マシンでは、 RMI 接続に使用する IP を収集するために、これを調整する必要があります。

JMX (9999 以上) とジョロキア (8778) には別のポートを選択できますJolokia をロックする内部 IP を持っている場合は、「 catch all 」 0.0.0.0 を自分の IP で置き換えることができます。この IP には、テレグラムプラグインからアクセスできる必要があります。認証を行わない場合は、オプション「 -Dcom.sun.management.jmxremote.authenticate=false' 」を使用できます。自己責任で使用してください。

オブジェクトとカウンタ

次のオブジェクトとそのカウンタが収集されます。

オブジェクト：	識別子：	属性：	「 **dataPoints** 」：
Kafka Broker	クラスタ ネームスペース ブローカー	ノード名 ノードIP	レプリカマネージャフェッチャー最大遅延 Zookeeperクライアント接続 ZooKeeperクライアント接続（15mレート） ZooKeeperクライアント接続（5mレート） ZooKeeperクライアント接続（平均速度） ZooKeeperクライアント接続（1mレート） レプリカマネージャのパーティション数 スレッドカウントデーモン スレッド数のピーク 現在のスレッド数 スレッド数合計が開始されました オフラインパーティション Produce Requests Total Time（50パーセンタイル値） Produce Requests Total Time（75パーセンタイル値） Produce Requests Total Time（95パーセンタイル値） Produce Requests Total Time（98パーセンタイル値） Produce Requests Total Time（999パーセンタイル値） Produce Requests Total Time（99パーセンタイル値） Produce Requests Total Timeの略 Produce Requests Total Time Max Produce Requests Total Time Meanの略 Produce Requests Total Time Min Produce Requests Total Time stddev レプリカマネージャISRが縮小されます レプリカ・マネージャのISRが縮小（15mレート

トラブルシューティング

追加情報はから入手できます "サポート" ページ

Kibana データコレクタ

Data Infrastructure Insightsでは、このデータコレクタを使用してKibanaから指標データを収集します。

インストール

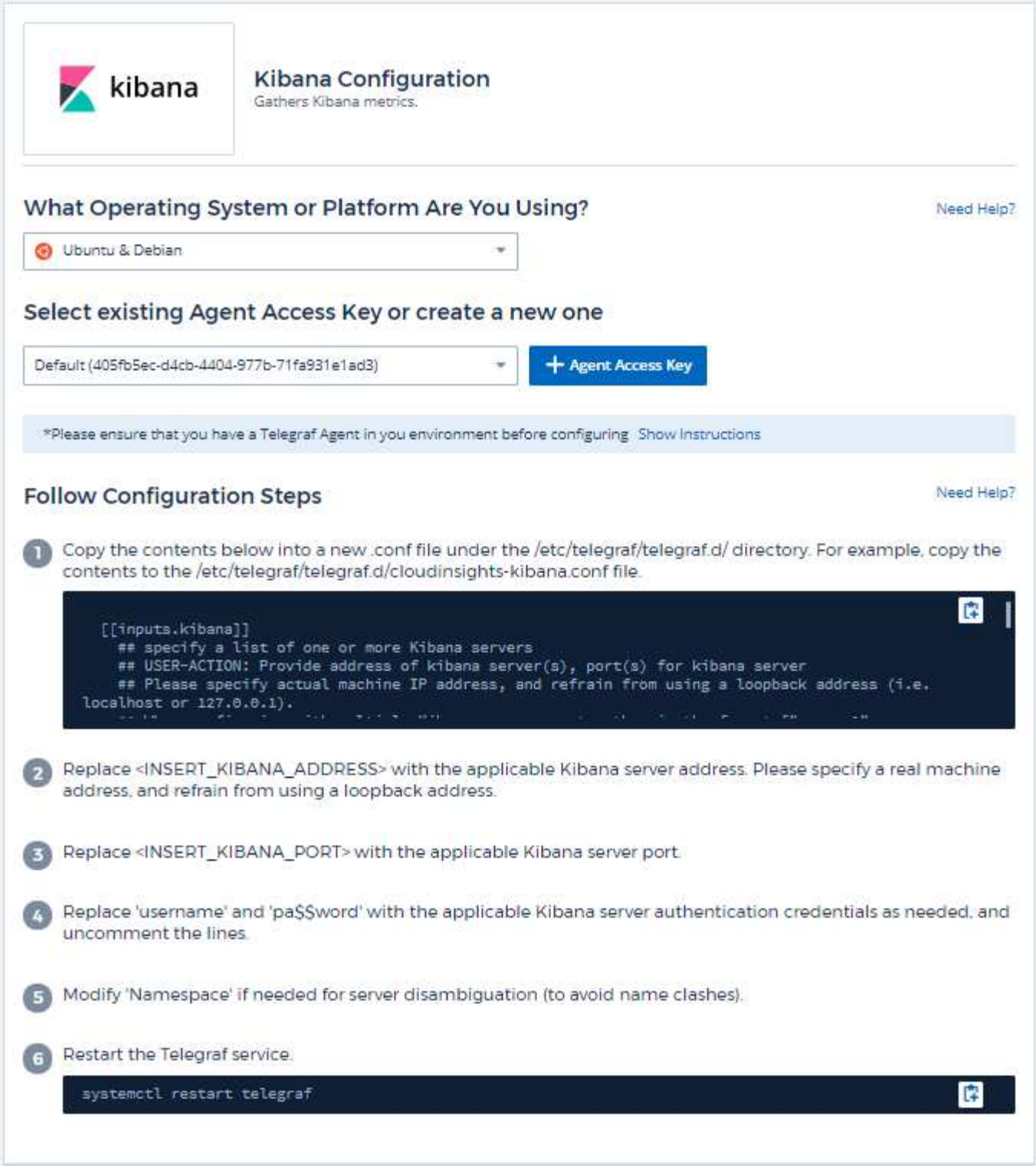
1. [Observability]>[Collectors]で、+ Data Collector *をクリックします。「きばな」を選択します。

   Tegraf エージェントがインストールされているオペレーティングシステムまたはプラットフォームを選択します。

2. Agent for Collection をまだインストールしていない場合、または別のオペレーティングシステムまたはプラットフォームに Agent をインストールする場合は、 *Show Instructions* をクリックしてを展開します "エージェントのインストール" 手順

3. このデータコレクタで使用するエージェントアクセスキーを選択します。[**+Agent Access Key**] ボタンをクリックすると、新しいエージェントアクセスキーを追加できます。ベストプラクティス：別のエージェントアクセスキーを使用するのは、たとえば OS/ プラットフォーム別にデータコレクタをグループ化する場合だけです。

4. 設定手順に従ってデータコレクタを設定します。手順は、データの収集に使用するオペレーティングシステムまたはプラットフォームのタイプによって異なります。



セットアップ（ **Setup** ）

情報は、に記載されています "Kibana のドキュメンテーション"。

オブジェクトとカウンタ

次のオブジェクトとそのカウンタが収集されます。

オブジェクト：	識別子：	属性：	「 **dataPoints** 」：
キバナ	ネームスペース 住所	ノードIP ノード名 バージョン ステータス	同時接続 ヒープ最大 ヒープが使用されました 1秒あたりの要求数 応答時間平均 最大応答時間 アップタイム

トラブルシューティング

追加情報はから入手できます "サポート" ページ

Kubernetes Monitoring Operatorのインストールと設定

Data Infrastructure Insightsは、Kubernetesコレクション向けの「Kubernetes Monitoring Operator」を提供しています。新しいオペレータを導入するには、* Kubernetes > Collectors >+ Kubernetes Collector *に移動します。

Kubernetes Monitoring Operatorをインストールする前に

を参照してください "前提条件" Kubernetes Monitoring Operatorをインストールまたはアップグレードする前のドキュメント。

Kubernetes Monitoring Operatorのインストール

kubernetes

Kubernetes

Deploy NetApp Monitoring Operator

Quickly install and configure a Kubernetes Operator to send cluster information to Cloud Insights.

Select existing API Access Token or create a new one

KEY2024 (...vw6NdM) | + API Access Token | Production Best Practices

Installation Instructions

Need Help?

Please review the pre-requisites for installing the NetApp Kubernetes Monitoring Operator.
To update an existing operator installation please follow these steps.

1 Define Kubernetes cluster name and namespace

Provide the Kubernetes cluster name and specify a namespace for deploying the monitoring components.

Cluster	Namespace
clustername	netapp-monitoring

2 Download the operator YAML files

Execute the following download command in a *bash* prompt.

Copy Download Command Snippet

Reveal Download Command Snippet

This snippet includes a unique access key that is valid for 24 hours.

3 Optional: Upload the operator images to your private repository

By default, the operator pulls container images from the Cloud Insights repository. To use a private repository, download the required images using the Image Pull command. Then upload them to your private repository maintaining the same tags and directory structure.
Finally, update the image paths in operator-deployment.yaml and the docker repository settings in operator-config.yaml.
For more information review the documentation.

Copy Image Pull Snippet

Reveal Image Pull Snippet

Copy Repository Password

Reveal Repository Password

This password is valid for 24 hours.

4 Optional: Review available configuration options

Configure custom options such as proxy and private repository settings. Review the instructions and available options.

5 Deploy the operator (create new or upgrade existing)

Execute the *kubectl* snippet to apply the following operator YAML files.

- operator-setup.yaml - Create the operator's dependencies.
- operator-secrets.yaml - Create secrets holding your API key.
- operator-deployment.yaml, operator-cr.yaml - Deploy the NetApp Kubernetes Monitoring Operator.
- operator-config.yaml - Apply the configuration settings if not already present.

Copy kubectl Apply Snippet

Reveal kubectl Apply Snippet

After deploying the operator, **delete or securely store operator-secrets.yaml.**

6 Next

KubernetesにKubernetes Monitoring Operatorエージェントをインストールする手順は次のとおりです。

1. 一意のクラスタ名およびネームスペースを入力してください。実行中の場合 をアップグレードして 以前のKubernetes Operatorで、同じクラスタ名とネームスペースを使用します。
2. これらを入力すると、ダウンロードコマンドスニペットをクリップボードにコピーできます。
3. スニペットを a_bash_window に貼り付け、実行します。Operatorインストールファイルがダウンロードされます。スニペットには固有のキーがあり、24時間有効です。
4. カスタムリポジトリまたはプライベートリポジトリがある場合は、オプションのImage Pullスニペットをコピーし、a_bash_shellに貼り付けて実行します。画像がプルされたら、プライベートリポジトリにコピーします。必ず同じタグとフォルダ構造を維持してください。_operator-deployment.yaml_のパスと_operator-config.yaml_のDockerリポジトリ設定を更新します。
5. 必要に応じて、プロキシやプライベートリポジトリの設定など、使用可能な設定オプションを確認します。あなたはについてもっと読むことができます "設定オプション"。
6. 準備ができたら、kubectl Applyスニペットをコピーしてダウンロードし、実行してOperatorをデプロイします。
7. インストールが自動的に開始されます。完了したら、[*Next*]ボタンをクリックします。
8. インストールが完了したら、[*Next*]ボタンをクリックします。また、_operator-secrets.yaml_fileを削除するか、安全に保存してください。

プロキシを使用している場合は、 プロキシを設定します。

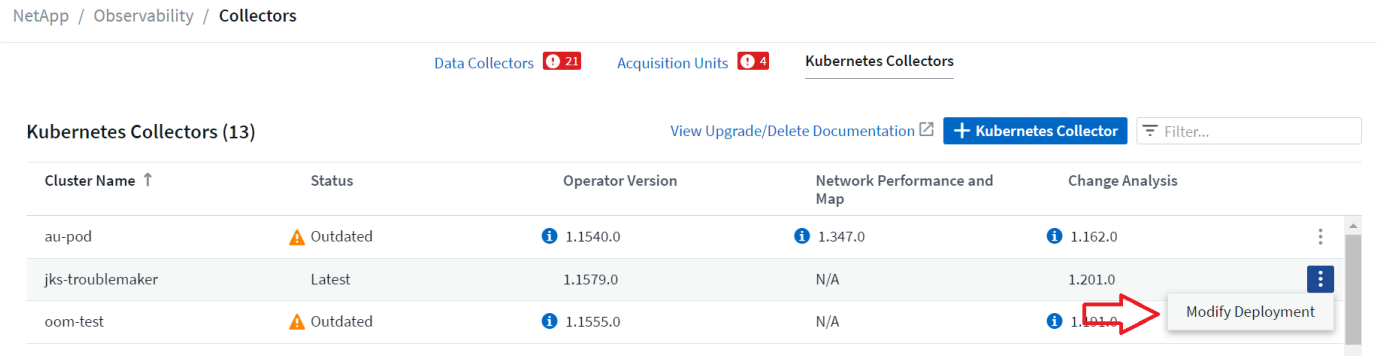
カスタムリポジトリをお持ちの場合は、 カスタム/プライベートDockerリポジトリを使用する。

Kubernetes監視コンポーネント

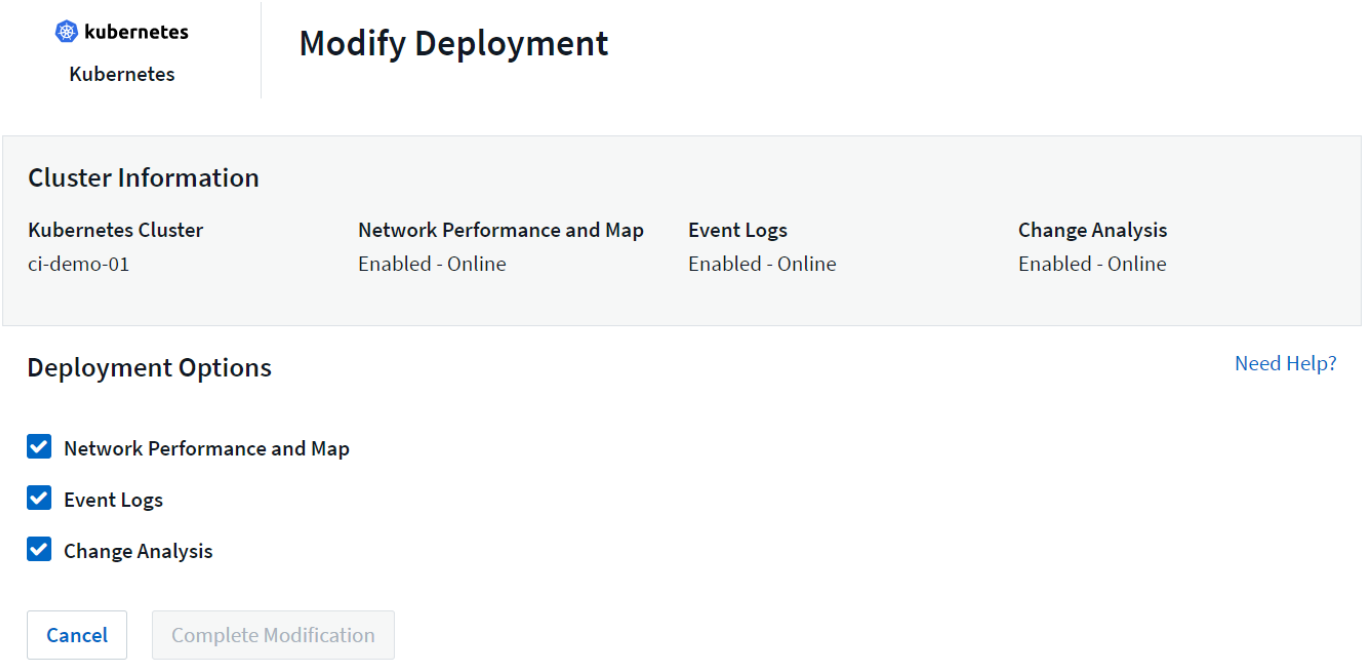
Data Infrastructure Insights Kubernetes Monitoringは、次の4つの監視コンポーネントで構成されます。

- クラスタ指標
- ネットワークパフォーマンスとマップ（オプション）
- イベントログ（オプション）
- 変更分析（オプション）

上記のオプションコンポーネントは、各Kubernetesコレクタに対してデフォルトで有効になっています。特定のコレクタ用のコンポーネントが必要ないと判断した場合は、* Kubernetes > Collectors *に移動し、画面右側のコレクタの「three dots」メニューから_Modify Deployment_を選択して無効にできます。



画面には各コンポーネントの現在の状態が表示され、必要に応じてそのコレクタのコンポーネントを無効または有効にすることができます。



をアップグレードして

最新の**Kubernetes Monitoring Operator**へのアップグレード

既存のOperatorにAgentConfigurationが存在するかどうかを確認します（ネームスペースがdefault_netapp-monitoring_でない場合は、適切なネームスペースに置き換えてください）。

```
 kubectl -n netapp-monitoring get agentconfiguration netapp-monitoring-configuration
AgentConfigurationが存在する場合：
```

- をインストールします 既存の演算子の上にある最新の演算子。
  - 確認してください 最新のコンテナイメージを取得します カスタムリポジトリを使用している場合。

AgentConfigurationが存在しない場合は、次の手順を実行します。

- クラスタ名がData Infrastructure Insightsで認識される名前であることをメモします（ネームスペースがデフォルトのNetApp監視機能でない場合は、適切なネームスペースで置き換えてください）。

```
 kubectl -n netapp-monitoring get agent -o jsonpath='{.items[0].spec.cluster-name}'
* 既存のOperatorのバックアップを作成します（ネームスペースがデフォルトのネットアップ監視機能になっていない場合は、適切なネームスペースで置き換えてください）。
```

```
 kubectl -n netapp-monitoring get agent -o yaml > agent_backup.yaml
* <<to-remove-the-kubernetes-monitoring-operator,をアンインストールします>> 既存の演算子。
* <<installing-the-kubernetes-monitoring-operator,をインストールします>> 最新の演算子。
```

  - 同じクラスタ名を使用してください。
  - 最新のOperator YAMLファイルをダウンロードしたら、展開する前に、agent_backup.yamlにあるカスタマイズをダウンロードしたoperator-config.yamlに移植します。
  - 確認してください 最新のコンテナイメージを取得します カスタムリポジトリを使用している場合。

Kubernetes Monitoring Operatorの停止と起動

Kubernetes Monitoring Operatorを停止するには：

```
 kubectl -n netapp-monitoring scale deploy monitoring-operator --replicas=0
Kubernetes Monitoring Operatorを起動するには：
```

```
kubectl -n netapp-monitoring scale deploy monitoring-operator --replicas=1
```

アンインストール中です

Kubernetes Monitoring Operatorを削除するには

Kubernetes Monitoring Operatorのデフォルトのネームスペースは「netapp-monitoring」です。 独自のネームスペースを設定した場合は、それらのネームスペースと、以降のすべてのコマンドおよびファイルを置き換えます。

新しいバージョンの監視オペレータは、次のコマンドを使用してアンインストールできます。

```
kubectl -n <NAMESPACE> delete agent -l installed-by=nkmo-<NAMESPACE>
kubectl -n <NAMESPACE> delete
clusterrole,clusterrolebinding,crd,svc,deploy,role,rolebinding,secret,sa
-l installed-by=nkmo-<NAMESPACE>
```

監視オペレータが専用のネームスペースに配置されている場合は、ネームスペースを削除します。

```
 kubectl delete ns <NAMESPACE>
```

最初のコマンドが「リソースが見つかりません」を返した場合は、次の手順に従って古いバージョンの監視オペレータをアンインストールします。

次の各コマンドを順番に実行します。現在のインストール状況によっては、これらのコマンドの一部で「オブジェクトが見つかりません」というメッセージが返される場合があります。これらのメッセージは無視してかまいません。

```
kubectl -n <NAMESPACE> delete agent agent-monitoring-netapp
kubectl delete crd agents.monitoring.netapp.com
kubectl -n <NAMESPACE> delete role agent-leader-election-role
kubectl delete clusterrole agent-manager-role agent-proxy-role agent-
metrics-reader <NAMESPACE>-agent-manager-role <NAMESPACE>-agent-proxy-role
<NAMESPACE>-cluster-role-privileged
kubectl delete clusterrolebinding agent-manager-rolebinding agent-proxy-
rolebinding agent-cluster-admin-rolebinding <NAMESPACE>-agent-manager-
rolebinding <NAMESPACE>-agent-proxy-rolebinding <NAMESPACE>-cluster-role-
binding-privileged
kubectl delete <NAMESPACE>-psp-nkmo
kubectl delete ns <NAMESPACE>
```

セキュリティコンテキスト制約が事前に作成されている場合は、次の手順を実行します。

```
kubectl delete scc telegraf-hostaccess
```

Kubeステートメトリックについて

NetApp Kubernetes Monitoring Operatorは、他のインスタンスとの競合を回避するために独自のkube-state-metricsをインストールします。

Kube-State-Metricsの詳細については、を参照してください。 "このページです"。

オペレータの設定/カスタマイズ

これらのセクションでは、オペレータ設定のカスタマイズ、プロキシの操作、カスタムまたはプライベートDockerリポジトリの使用、OpenShiftの操作について説明します。

設定オプション

最も一般的に変更される設定は、_AgentConfiguration_customリソースで構成できます。オペレータを配備する前に、_operator-config.yaml_fileを編集して、このリソースを編集できます。このファイルには、コメントアウトされた設定例が含まれています。のリストを参照してください "使用可能な設定" 演算子の最新バージョン。

オペレータが配備された後で、次のコマンドを使用してこのリソースを編集することもできます。

```
kubectl -n netapp-monitoring edit AgentConfiguration
```

展開したオペレータのバージョンが`AgentConfiguration`をサポートしているかどうかを確認するには、次のコマンドを実行します。

```
kubectl get crd agentconfigurations.monitoring.netapp.com
```

「`Error from server (NotFound)`」というメッセージが表示された場合は、`AgentConfiguration`を使用する前にオペレータをアップグレードする必要があります。

プロキシサポートを設定しています

Kubernetes Monitoring Operatorをインストールするために、環境内でプロキシを使用できる場所は2つあります。同じプロキシシステムでも、別のプロキシシステムでもかまいません。

- インストールコードスニペット（「curl」を使用）の実行中に、スニペットが実行されるシステムをData Infrastructure Insights環境に接続するために必要なプロキシ
- ターゲットのKubernetesクラスタがData Infrastructure Insights環境と通信するために必要なプロキシ

これらのいずれかまたは両方にプロキシを使用する場合、Kubernetes Operating Monitorをインストールするには、まず、Data Infrastructure Insights環境との通信が良好になるようにプロキシが設定されていることを確認する必要があります。プロキシがあり、OperatorをインストールするサーバI/ VMからData Infrastructure Insightsにアクセスできる場合は、プロキシが適切に設定されている可能性があります。

Kubernetes Operating Monitorのインストールに使用するプロキシについては、Operatorをインストールする前に、_http_proxy/https_proxy_environment変数を設定します。一部のプロキシ環境では'_no_proxy環境変数も設定する必要があります

変数を設定するには、Kubernetes Monitoring Operatorをインストールする前に、システム*で次の手順を実行します。

1. 現在のユーザの *https_proxy* 変数と _http_proxy_environment 変数を設定します。
   a. セットアップするプロキシに認証（ユーザ名/パスワード）がない場合は、次のコマンドを実行します。

```
 export https_proxy=<proxy_server>:<proxy_port>
.. セットアップするプロキシに認証（ユーザ名
/パスワード）が設定されている場合は、次のコマンドを実行します。
```

```
export
http_proxy=<proxy_username>:<proxy_password>@<proxy_server>:<proxy_po
rt>
```

KubernetesクラスタがData Infrastructure Insights環境と通信するために使用するプロキシの場合は、以下の手順をすべて読んでからKubernetes Monitoring Operatorをインストールします。

Kubernetes Monitoring Operatorをデプロイする前に、operator-config.yamlのAgentConfigurationのプロキシセクションを設定します。

```
agent:
  ...
  proxy:
    server: <server for proxy>
    port: <port for proxy>
    username: <username for proxy>
    password: <password for proxy>

    # In the noproxy section, enter a comma-separated list of
    # IP addresses and/or resolvable hostnames that should bypass
    # the proxy
    noproxy: <comma separated list>

    isTelegrafProxyEnabled: true
    isFluentbitProxyEnabled: <true or false> # true if Events Log enabled
    isCollectorsProxyEnabled: <true or false> # true if Network
Performance and Map enabled
    isAuProxyEnabled: <true or false> # true if AU enabled
  ...
...
```

カスタムまたはプライベートの**Docker**リポジトリを使用する

Kubernetes監視オペレータは、デフォルトで、Data Infrastructure Insightsリポジトリからコンテナイメージを取得します。監視のターゲットとして使用されているKubernetesクラスタがあり、そのクラスタがカスタムまたはプライベートのDockerリポジトリまたはコンテナレジストリからコンテナイメージのみをプルするように構成されている場合は、Kubernetes Monitoring Operatorが必要とするコンテナへのアクセスを設定する必要があります。

NetApp Monitoring Operatorのインストールタイルから[Image Pull Snippet]を実行します。このコマンドを実行すると、Data Infrastructure Insightsリポジトリにログインし、オペレータが必要とするすべてのイメージを取得して、Data Infrastructure Insightsリポジトリからログアウトします。プロンプトが表示されたら、指定したリポジトリの一時パスワードを入力します。このコマンドは、オプション機能を含む、オペレータが使用するすべてのイメージをダウンロードします。これらの画像がどの機能に使用されるかについては、以下を参照してください。

Core Operator Functionality and Kubernetes Monitoringの略

- ネットアップによる監視
- ci-kube-rbac-proxy
- CI-KSM
- CI-テレグラフ
- distroless-root-user

イベントログ

- CI-fluent-bit
- ci-kubernetes-event-exporter

ネットワークのパフォーマンスとマップ

- ci-net-observerの略

社内のポリシーに従って、オペレータ用の Docker イメージをプライベート / ローカル / エンタープライズ Docker リポジトリにプッシュします。リポジトリ内のこれらのイメージへのイメージタグとディレクトリパスが、Data Infrastructure Insightsリポジトリ内のイメージタグとディレクトリパスと一致していることを確認します。

operator-deployment.yamlでmonitoring-operatorデプロイメントを編集し、プライベートDockerリポジトリを使用するようにすべてのイメージ参照を変更します。

```
image: <docker repo of the enterprise/corp docker repo>/kube-rbac-
proxy:<ci-kube-rbac-proxy version>
image: <docker repo of the enterprise/corp docker repo>/netapp-
monitoring:<version>
```

operator-config.yamlのAgentConfigurationを編集して、新しいDockerリポジトリの場所を反映します。プライベートリポジトリ用に新しいimagePullSecretを作成します。詳細については、_ https://kubernetes.io/docs/tasks/configure-pod-container/pull-image-private-registry/_を参照してください

```
agent:
  ...
  # An optional docker registry where you want docker images to be pulled
from as compared to CI's docker registry
  # Please see documentation link here:
xref:{relative_path}task_config_telegraf_agent_k8s.html#using-a-custom-or-
private-docker-repository
  dockerRepo: your.docker.repo/long/path/to/test
  # Optional: A docker image pull secret that maybe needed for your
private docker registry
  dockerImagePullSecret: docker-secret-name
```

OpenShift の手順

OpenShift 4.6以降で実行している場合は、_runPrivileged_settingを有効にするには、_operator-config.yaml_でAgentConfigurationを編集する必要があります。

```
# Set runPrivileged to true SELinux is enabled on your kubernetes nodes
runPrivileged: true
```

OpenShiftは、一部のKubernetesコンポーネントへのアクセスをブロックする可能性のある追加のセキュリテ

ィレベルを実装する場合があります。

秘密に関する注意事項

Kubernetes Monitoring Operatorのシークレットをクラスタ全体で表示する権限を削除するには、インストール前に_operator-setup.yaml_fileから次のリソースを削除します。

```
ClusterRole/netapp-ci-<namespace>-agent-secret-clusterrole
ClusterRoleBinding/netapp-ci-<namespace>-agent-secret-clusterrolebinding
```

アップグレードの場合は、クラスタからリソースも削除します。

```
kubectl delete ClusterRole/netapp-ci-<namespace>-agent-secret-clusterrole
kubectl delete ClusterRoleBinding/netapp-ci-<namespace>-agent-secret-
clusterrolebinding
```

変更分析が有効になっている場合は、_AgentConfiguration_or_operator -config.yaml_を変更して、変更管理セクションのコメントを解除し、変更管理セクションの下に_kindsToIgnoreFromWatch：'"secrets"'_を含めます。この行の一重引用符と二重引用符の存在と位置に注意してください。

```
# change-management:
  ...
  # # A comma separated list of kinds to ignore from watching from the
default set of kinds watched by the collector
  # # Each kind will have to be prefixed by its apigroup
  # # Example: '"networking.k8s.io.networkpolicies,batch.jobs",
"authorization.k8s.io.subjectaccessreviews"'
  kindsToIgnoreFromWatch: '"secrets"'
  ...
```

Kubernetes のチェックサムの検証

Data Infrastructure Insightsエージェントインストーラは整合性チェックを実行しますが、ダウンロードしたアーティファクトをインストールまたは適用する前に独自の検証を実行することもできます。デフォルトのダウンロードおよびインストールではなく、ダウンロードのみの操作を実行するには、 UI から取得したエージェントインストールコマンドを編集し、末尾の「インストール」オプションを削除します。

次の手順を実行します。

1. 指示に従ってエージェントインストーラスニペットをコピーします。
2. スニペットをコマンドウィンドウに貼り付ける代わりに、テキストエディタに貼り付けます。
3. コマンドから末尾の「--install」を削除します。
4. コマンド全体をテキストエディタからコピーします。

5. 次に、コマンドウィンドウ（作業ディレクトリ内）に貼り付けて実行します。
   - Download and install （デフォルト）：

```
 installerName=cloudinsights-rhel_centos.sh … && sudo -E -H
./$installerName --download --install
** ダウンロードのみ：
```

```
installerName=cloudinsights-rhel_centos.sh … && sudo -E -H
./$installerName --download
```

download-onlyコマンドを実行すると、必要なすべてのアーティファクトがData Infrastructure Insightsから作業ディレクトリにダウンロードされます。アーティファクトには次のものがありますが、これらに限定することはできません。

- インストールスクリプト
- 環境ファイル
- YAMLファイル
- 署名済みチェックサムファイル（ SHA256 署名）
- 署名の検証に使用する PEM ファイル（ NetApp_cert.pem ）

インストールスクリプト、環境ファイル、 YAML ファイルは、目視検査を使用して検証できます。

PEM ファイルは、フィンガープリントが次のようになっていることを確認することで検証できます。

```
 1A918038E8E127BB5C87A202DF173B97A05B4996
具体的には、
```

```
 openssl x509 -fingerprint -sha1 -noout -inform pem -in netapp_cert.pem
署名済みチェックサムファイルは、 PEM ファイルを使用して確認できます。
```

```
 openssl smime -verify -in sha256.signed -CAfile netapp_cert.pem -purpose
any
すべてのアーティファクトが正常に検証されたら、次のコマンドを実行してエージェントのインストールを開始できます。
```

```
sudo -E -H ./<installation_script_name> --install
```

公差と接線（**Tolerations and Taints**）

NetApp-ci-telegraf-ds_、*NetApp-CI-fluent-bit-ds*、および_NetApp-CI-net-observer-l4-DS_DaemonSetsは、すべてのノードのデータを正しく収集するために、クラスタ内のすべてのノードでポッドをスケジュールする必要があります。オペレータは、いくつかの既知の*テイント*に耐えられるように設定されています。ノードにカスタムのtaintsを設定して、すべてのノードでポッドが実行されないようにしている場合は、それらのtaintsに* toleration *を作成できます "（*AgentConfiguration*）をクリックします"。クラスタ内のすべてのノードにカスタムテイントを適用した場合は、オペレータの導入に必要な許容範囲を追加して、オペレータポッドをスケジュールおよび実行できるようにする必要があります。

Kubernetesの詳細はこちらをご覧ください "塗料および耐性"。

に戻ります "* NetApp Kubernetes監視オペレータのインストール*ページ"

トラブルシューティング

Kubernetes Monitoring Operatorの設定で問題が発生した場合に試すべきこと：

問題	次の操作を実行します
Kubernetes 永続ボリュームと対応するバックエンドストレージデバイスの間にハイパーリンク / 接続がありません。My Kubernetes Persistent Volume がストレージサーバのホスト名を使用して設定されます。	手順に従って既存の Tegraf エージェントをアンインストールし、最新の Tegraf エージェントを再インストールします。Telegrafバージョン2.0以降を使用しており、KubernetesクラスタストレージがData Infrastructure Insightsによってアクティブに監視されている必要があります。

問題	次の操作を実行します
ログに次のようなメッセージが表示されます。 E0901 15:21:39.962145 1 reflector.go:178]k8s.io/kube-state-metrics/internal/store/builder.go:352:リストに失敗しました*v1.MutatingWebhookConfiguration:サーバーは要求されたリソースを見つけることができませんでした E0901 15:21:43.168161 1 reflector.go:178]k8s.io/kube-state-metrics/internal/store/builder.go:352:リストに失敗しました*v1 Lease:サーバーは要求されたリソースを見つけることができませんでした(GET leases.coordination.k8s.io) など	これらのメッセージは、1.20より前のバージョンのKubernetesでkube-state-metricsバージョン2.0.0以上を実行している場合に発生する可能性があります。 Kubernetesのバージョンを取得するには： *kubectl*バージョン kube-state-metricsバージョンを取得するには、次の手順を実行します。 *kubectl get deploy/kube-state-metrics -o jsonpath='｛..image｝'* これらのメッセージが発生しないように、ユーザはkube-state-metrics展開を変更して、次のリースを無効にすることができます。 *mutatingwebhookconfigurations* 検証webhookconfigurations_ *volumeattachments*リソース 具体的には、次のCLI引数を使用できます。 resources=certificatesigningrequests, configmaps, cronjobs, daemonsets, deployments, endpoints, horizontalpodautoscalers, ingresses, jobs, limitranges, namespaces, networkpolicies, poddisruptionbudgets, pods, ReplicaSets, replicationcontrollasses, resourcequotases, secrets, resourcequotases, secrets, services, storage, storefuls. デフォルトのリソースリストは次のとおりです。 certificatesigningrequests, configmaps, cronjobs, daemonsets, deployments, endpoints, horizontalpododautoscalers, ingresses, jobs, leases, limitranges, mutatingwebhookconfiguration, namespaces, networkpersistentvolumes, poddisruptionbudgets, pers, persistentresets, pondsets, podsets, postresets, replicassess, replicastess, replicatess, replicastorets, replicast 検証Webhook構成’ボリューム添付ファイル"\y"ケンショウ:Webhookコウセイ’ボリュームアタッチメント

問題	次の操作を実行します
Telegrafから次のようなエラーメッセージが表示されますが、Telegrafは起動して実行されます。 10月11日14:23:41 IP-172-31-39-47 systemd[1]: InfluxDBにメトリックを報告するプラグイン駆動のサーバーエージェントを起動しました。 10 月 11 日 14 ： 23 ： 41 IP-172-41-39-47 テレグラム [1827] ： time="2021 - 10-11T14 ： 23 ： 41Z" level= error msg=" キャッシュディレクトリの作成に失敗しました。/etc/telegraf/.cache/snowflake 、err:mkdir /etc/telegraf/.ca CHE：権限が拒否されました。無視\n" func="gosnowflake.(*defaultLogger).Errorf" file="log.go:120" 10月11日14：23：41 IP-172-31-39-47 telegraf [1827] ：time="2021-10-11T14：23：41Z" level=error msg="failed to open.無視されます。/etc/telegraf/.cache/snowflake/ocsp_response_cache.jsonを開きます。no such ファイルまたはディレクトリ\n" func="gosnowflake.(*defaultLogger).Errorf" file="log.go:120" 10月11日14:23:41 IP-172-31-39-47 telegraf [1827]: 2021-10-11T14:23:41Z I! Telegraf 1.19.3 を起動しています	これは問題と呼ばれています。 を参照してください "この GitHub の記事" 詳細：Tegraf が起動して動作している限り、ユーザはこのエラーメッセージを無視できます。
Kubernetesで、Telegrafポッドが次のエラーを報告しています。 "mountstats情報の処理中にエラーが発生しました ：mountstatsファイルを開けませんでした ：/hostfs/proc/1/mountstats、エラー ：open/hostfs/proc/1/mountstats：権限が拒否されました"	SELinuxを有効にして強制すると、TelegrafポッドがKubernetesノードの/proc/1/mountstatsファイルにアクセスできなくなる可能性があります。この制限を克服するには、agentconfigurationを編集し、runPrivileged設定を有効にします。詳細については、 "OpenShift の手順"。
Kubernetesで、Telegraf ReplicaSetポッドが次のエラーを報告しています。 [プラグインのinputs.prometheus]エラー：keypair /etc/kubernetes/pki/etcd/server.crtをロードできませんでした：/etc/kubernetes/pki/etcd/server.key：open /etc/kubernetes/pki/etcd/server.crt：該当するファイルまたはディレクトリはありません	Telegraf ReplicaSet ポッドは、マスターまたは etcd 用に指定されたノード上で実行することを目的としています。これらのノードのいずれかで ReplicaSet ポッドが実行されていない場合は、これらのエラーが発生します。マスター / etcd ノードに汚染があるかどうかを確認します。その場合は、 Telegraf ReplicaSet 、テレグラム af-RS に必要な忍容を追加します。 たとえば、ReplicaSet… kubectl edit rs telegraf-rs 仕様に適切な公差を追加します。次に、 ReplicaSet ポッドを再起動します。

問題	次の操作を実行します
PSP/PSA環境があります。これはモニタリングオペレータに影響しますか？	KubernetesクラスタがPod Security Policy（PSP）またはPod Security Admission（PSA）を使用して実行されている場合は、最新のKubernetes Monitoring Operatorにアップグレードする必要があります。PSP/PSAをサポートしている現在のオペレータにアップグレードするには、次の手順に従います。 1. をアンインストールします 以前の監視オペレータ： kubectl delete agent agent-monitoring-netapp-n netapp-monitoring kubectlによってネットアップによる監視が削除されます kubectlはCRD agents.monitoring.netapp.comを削除します kubectl delete clusterrole agent-manager-role agent-proxy-role agent-metrics-reader kubectl delete clusterrolebinding agent-manager-rolebinding agent-proxy-rolebinding agent-cluster-admin-rolebinding 2. をインストールします モニタリングオペレータの最新バージョン。
Operatorを展開しようとして問題が発生しましたが、PSP/PSAを使用しています。	1.次のコマンドを使用してエージェントを編集します。 kubectl -n <name-space>編集エージェント 2.「security-policy-enabled」を「false」に設定します。これにより、PodセキュリティポリシーとPodセキュリティアドミッションが無効になり、オペレータが展開できるようになります。次のコマンドを使用して確認します。 kubectl get psp（Pod Security Policy removedを表示する必要があります） kubectl get all -n <namespace>
grep -i psp (should show that nothing is found)	「ImagePullBackoff」エラーが発生しました
これらのエラーは、カスタムまたはプライベートのDockerリポジトリがあり、Kubernetes Monitoring Operatorを適切に認識するように設定していない場合に表示されることがあります。 詳細はこちら カスタム/プライベートリポジトリの設定について	監視オペレータの配置に問題 を使用していますが、現在のドキュメントでは解決できません。

問題	次の操作を実行します

次のコマンドの出力をキャプチャまたはメモし、テクニカルサポートチームに連絡します。
 	
`kubectl -n netapp-monitoring get all`
`kubectl -n netapp-monitoring describe all`
`kubectl -n netapp-monitoring logs <monitoring-operator-pod> --all-containers=true`
`kubectl -n netapp-monitoring logs <telegraf-pod> --all-containers=true`
	Operator名前空間のNet-Observer（ワークロードマップ）ポッドがCrashLoopBackOffにある

これらのポッドは、Network ObservabilityのWorkload Mapデータコレクタに対応しています。以下をお試しください。 •いずれかのポッドのログをチェックして、カーネルの最小バージョンを確認します。例： ----- {"ci-tenant-id"："your-tenant-id"、"collector-cluster"："your-k8s-cluster-name"、"environment"："prod"、"level"："error"、"msg"："検証に失敗しました。理由：カーネルバージョン3.10.0が最小カーネルバージョン4.18.0よりも小さい、"time"："2022-11-09T08:23:08Z"} ----- •Net-Observerポッドを使用するには、Linuxカーネルのバージョンが4.18.0以上である必要があります。「uname -r」コマンドを使用してカーネルのバージョンを確認し、4.18.0以上であることを確認します	PodはOperatorネームスペース（デフォルト：netapp-monitoring）で実行されているが、Queriesのワークロードマップまたは Kubernetes指標のデータがUIに表示されない
K8Sクラスタのノードの時間設定を確認します。監査およびデータレポートを正確に作成するには、Network Time Protocol（NTP；ネットワークタイムプロトコル）またはSimple Network Time Protocol（SNTP；簡易ネットワークタイムプロトコル）を使用してAgentマシンの時刻を同期することを強く推奨します。	Operator名前空間の一部のnet-observerポッドがPending状態です

問題	次の操作を実行します
net-observerはデーモンセットであり、Kubernetesクラスタの各ノードでポッドを実行します。 •保留状態のポッドをメモし、CPUまたはメモリのリソース問題が発生しているかどうかを確認します。必要なメモリとCPUがノードにあることを確認します。	Kubernetes Monitoring Operatorをインストールした直後にログに次のようなメッセージが表示されます。 [プラグインのinputs.prometheus]エラー：\http://kube-state-metricsへの要求エラー。<namespace>.svc.cluster.local：8080/metrics：get\ http://kube-state-metrics <namespace>.svc.cluster.local：8080/metrics：dial tcp：lookup kube-state-metrics 。<namespace>.svc.cluster.local：該当するホストはありません。
このメッセージが表示されるのは、通常、_KSM_PODが起動する前に、新しいオペレータがインストールされ、_テレ グラム-RS_PODが稼働している場合のみです。これらのメッセージは、すべてのポッドが実行されると停止します。	クラスタに存在するKubernetes CronJobsについて収集された指標が表示されません。
Kubernetesのバージョンを確認します（`kubectl version`）。 v1.20.x以下の場合、これは想定される制限です。 Kubernetes Monitoring Operatorで導入されたkube-state-metricsリリースでは、v1.cronjobのみがサポートされます。 Kubernetes 1.20.x以前では、cronjobリソースはv1beta.cronjobにあります。その結果、kube-state-metricsはcronjobリソースを見つけることができません。	オペレータのインストール後、telegraf-DSポッドがCrashLoopBackOffに入り、PODログに「su：Authentication failure」と表示されます。
_AgentConfiguration_のtelegrafセクションを編集し、set_dockerMetricCollectionEnabled_をfalseに設定します。詳細については、オペレータのを参照して"設定オプション"ください。注: Data Infrastructure Insights Federal Editionを使用している場合、_su_の使用が制限されているユーザーはDockerメトリクスを収集できません。Dockerソケットにアクセスするには、telegrafコンテナをrootとして実行するか、_su_を使用してtelegrafユーザーをDockerグループに追加する必要があるためです。Dockerメトリック収集と_su_の使用はデフォルトで有効になっています。両方を無効にするには、_AgentConfiguration_fileの_telegraf.docker_entryを削除します。…spec:…telegraf:… -name：docker run-mode： -DaemonSet 置換： -key：docker_unix_sock_placeholder 値：unix：//run/docker.sock……	Telegrafログに次のようなエラーメッセージが繰り返し表示されます。 来い! [agent]出力への書き込み中にエラーが発生しました。http：Post "\https：//<tenant_url>/rest/v1/lake/ingest/influxdb"：context deadline exceeded (Client. ヘッダー待機中にタイムアウトを超過しました）
_AgentConfiguration_およびincrease_outputTimeout_のtelegrafセクションを10秒に編集します。詳細については、オペレータのを参照してください "設定オプション"。	一部のイベントログの_involvedobject_dataが見つかりません。
次の手順を実行していることを確認してください： "権限" 上記のセクション。	2つの監視オペレータポッド（netapp-ci-monitoring-operator-pod <pod>とmonitoring-operator-pod）が実行されているのはなぜ<pod>ですか？

問題	次の操作を実行します
2023年10月12日付けで、Data Infrastructure Insightsは、ユーザへのサービス向上のためにオペレータをリファクタリングしました。これらの変更を完全に採用するには古いオペレータを削除します。、とが必要です。新しいものを取り付ける	Kubernetesイベントが予期せずData Infrastructure Insightsに報告されなくなりました。
event-exporterポッドの名前を取得します。 `kubectl -n netapp-monitoring get pods`	grep event-exporter
awk '{print $1}'	sed 's/event-exporter./event-exporter/'` 「netapp-ci-event-exporter」または「event-exporter」のいずれかにする必要があります。 次に、監視エージェントを編集します。 `kubectl -n netapp-monitoring edit agent`をクリックし、log_fileの値を設定して、前の手順で見つけた適切なイベントエクスポータポッド名を反映します。 具体的には、log_fileは「/var/log/containers/netapp-ci-event-exporter.log」または「/var/log/containers/event-exporter 。**log**」のいずれかに設定する必要があります。 **….** **fluent-bit:** **…** **- name: event-exporter-ci** **substitutions:** **- key: LOG_FILE** **values:** **- /var/log/containers/netapp-ci-event-exporter**.log … …. 別の方法として、 をアンインストールします および 再インストール エージェント。
リソースが不足しているため、Kubernetes Monitoring Operatorによってデプロイされたポッドがクラッシュしています。	Kubernetes Monitoring Operatorを参照 "設定オプション" 必要に応じてCPUやメモリの制限を増やします。
イメージがないか無効な設定が原因で、netapp-ci-kube-state-metricsポッドが起動しないか準備完了状態になりました。これでStatefulSetが停止し、設定の変更がnetapp-ci-kube-state-metricsポッドに適用されなくなりました。	StatefulSetは "切断" 状態。設定の問題を修正したら、netapp-ci-kube-state-metricsポッドをバウンスします。
NetApp-ci-kube-state-metricsポッドがKubernetes Operatorのアップグレード実行後に起動せず、ErrImagePullがスローされる（イメージをプルできない）。	ポッドを手動でリセットしてみてください。

問題	次の操作を実行します
Kubernetesクラスタの[Log Analysis]で、「Event discarded as being older then maxEventAgeSeconds」というメッセージが確認されています。	Operator_agentconfiguration_を変更し、*event-exporter-maxEventAgeSeconds*（60秒）、*event-exporter-kubeQPS*（100）、および_event-exporter-kubeBurst_（500）を増やします。これらの設定オプションの詳細については、を参照してください。 "設定オプション" ページ
Telegrafが警告するか、ロック可能なメモリが不足しているためにクラッシュします。	基盤となるオペレーティングシステム/ノードでTelegrafのロック可能メモリの制限を増やしてみてください。制限値を増やすことができない場合は'NKMOエージェントの構成を変更して'_unprotected_to_true_に設定します これにより、Telegrafはロックされたメモリページを予約しないように指示します。復号化されたシークレットがディスクにスワップアウトされる可能性があるため、セキュリティリスクが発生する可能性がありますが、ロックされたメモリを予約できない環境では実行できます。_unprotected_configurationオプションの詳細については、 "設定オプション" ページ

追加情報はから入手できます "サポート" ページまたはを参照してください "Data Collector サポートマトリックス"。

Memcached Data Collector

Data Infrastructure Insightsは、このデータコレクタを使用してMemcachedから指標を収集します。

インストール

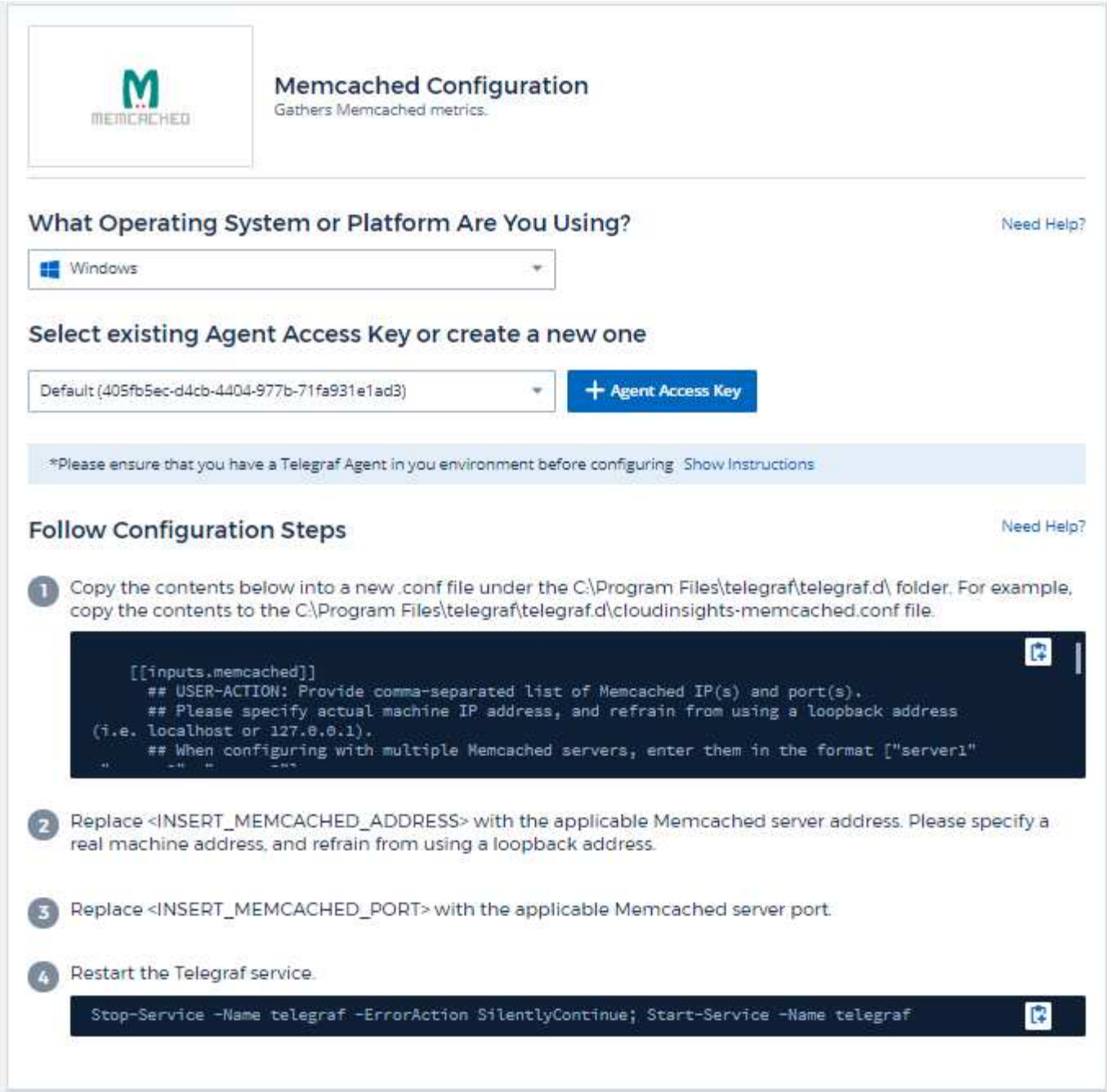
1. [Observability]>[Collectors]で、+ Data Collector *をクリックします。Memcachedを選択します。

   Tegraf エージェントがインストールされているオペレーティングシステムまたはプラットフォームを選択します。

2. Agent for Collection をまだインストールしていない場合、または別のオペレーティングシステムまたはプラットフォームに Agent をインストールする場合は、 *Show Instructions* をクリックしてを展開します "エージェントのインストール" 手順

3. このデータコレクタで使用するエージェントアクセスキーを選択します。[**+Agent Access Key**] ボタンをクリックすると、新しいエージェントアクセスキーを追加できます。ベストプラクティス：別のエージェントアクセスキーを使用するのは、たとえば OS/ プラットフォーム別にデータコレクタをグループ化する場合だけです。

4. 設定手順に従ってデータコレクタを設定します。手順は、データの収集に使用するオペレーティングシステムまたはプラットフォームのタイプによって異なります。



セットアップ（ **Setup** ）

情報は、に記載されています "Memcached Wiki"。

オブジェクトとカウンタ

次のオブジェクトとそのカウンタが収集されます。

オブジェクト：	識別子：	属性：	「 **dataPoints** 」：
Memcached	ネームスペース サーバ	ノードIP ノード名	接続を受け入れています 認証要求を処理しました 認証に失敗しました 使用済みバイト数 読み取りバイト数（1秒あたり） 書き込みバイト数（1秒あたり） キャス・バヴァル CASヒット CASミス フラッシュ要求（1秒あたり） 要求の取得（1秒あたり） 要求の設定（1秒あたり） タッチ要求（1秒あたり） 接続歩留まり（1秒あたり） 接続構造 接続を開きます 現在保存されているアイテム Decr Requests Hits（1秒あたり） Decr Requests Misses（1秒あたり） Delete Requests Hits（1秒あたり） Delete Requests Misses（1秒あたり） 削除されたアイテム 有効な削除 期限切れのアイテム GETヒット数（1秒あたり） Get Misses（1秒あたり） 使用されているハッシュバイト数 ハッシュが拡張されています ハッシュパワーレベル Incr Requests Hits（1秒あたり） Incr Requests Misses（1秒あたり） サーバ最大バイト数 リッスン無効番号 再生されました ワーカースレッド数 オープンされている接続の総数 保存されている合計アイテム数 [ヒット]に触れます

トラブルシューティング

追加情報はから入手できます "サポート" ページ

MongoDB データコレクタ

Data Infrastructure Insightsは、このデータコレクタを使用してMongoDBから指標を収集します。

インストール

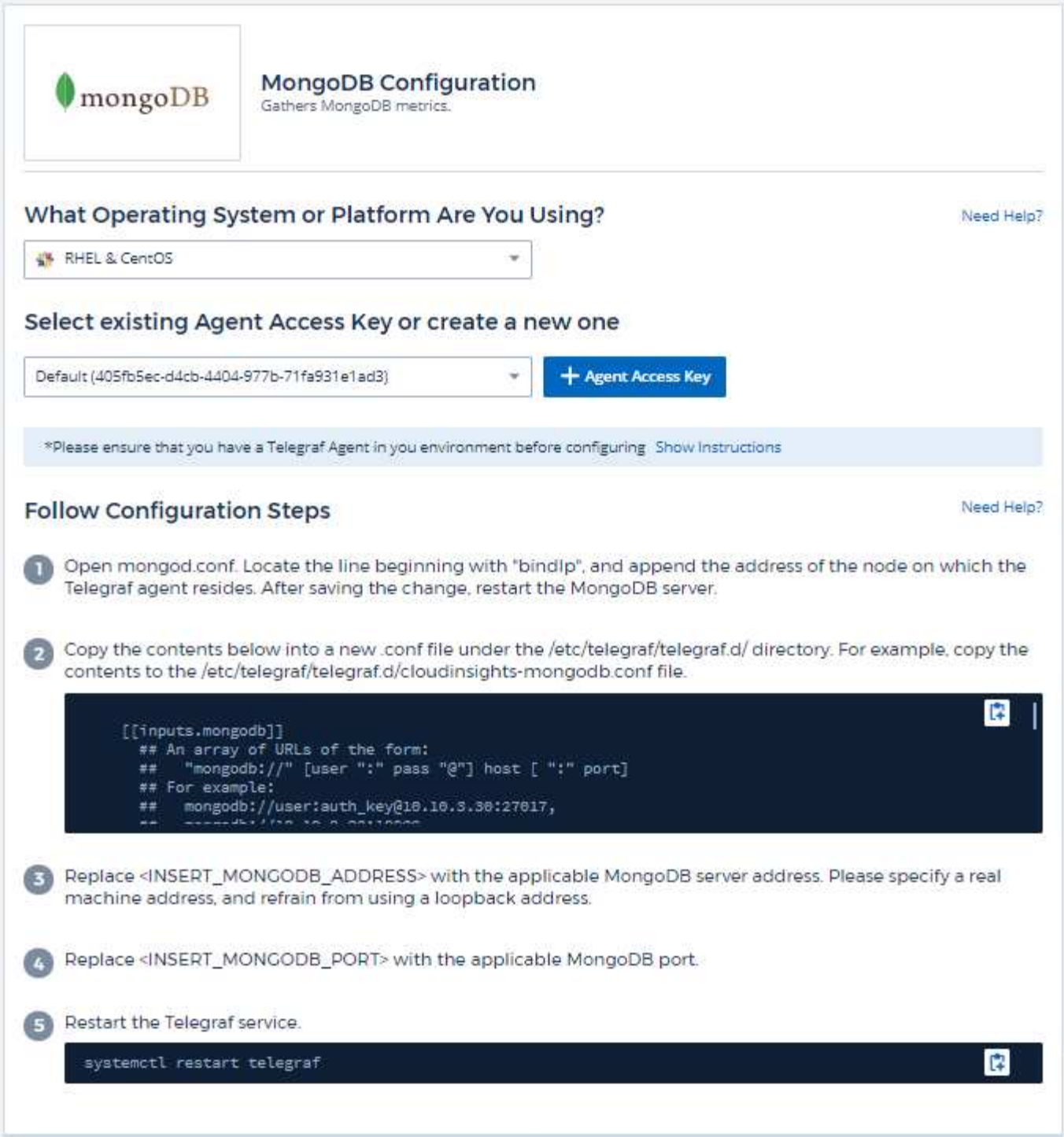
1. [Observability]>[Collectors]で、+ Data Collector *をクリックします。[MongoDB]を選択します。

   Tegraf エージェントがインストールされているオペレーティングシステムまたはプラットフォームを選択します。

2. Agent for Collection をまだインストールしていない場合、または別のオペレーティングシステムまたはプラットフォームに Agent をインストールする場合は、 *Show Instructions* をクリックしてを展開します "エージェントのインストール" 手順

3. このデータコレクタで使用するエージェントアクセスキーを選択します。[**+Agent Access Key**] ボタンをクリックすると、新しいエージェントアクセスキーを追加できます。ベストプラクティス：別のエージェントアクセスキーを使用するのは、たとえば OS/ プラットフォーム別にデータコレクタをグループ化する場合だけです。

4. 設定手順に従ってデータコレクタを設定します。手順は、データの収集に使用するオペレーティングシステムまたはプラットフォームのタイプによって異なります。



セットアップ（ **Setup** ）

情報は、に記載されています "MongoDB のドキュメント"。

オブジェクトとカウンタ

次のオブジェクトとそのカウンタが収集されます。

オブジェクト：	識別子：	属性：	「 **dataPoints** 」：
MongoDB	ネームスペース ホスト名		
MongoDB データベース	ネームスペース ホスト名 データベース名		

トラブルシューティング

情報はから入手できます "サポート" ページ

MySQL データコレクタ

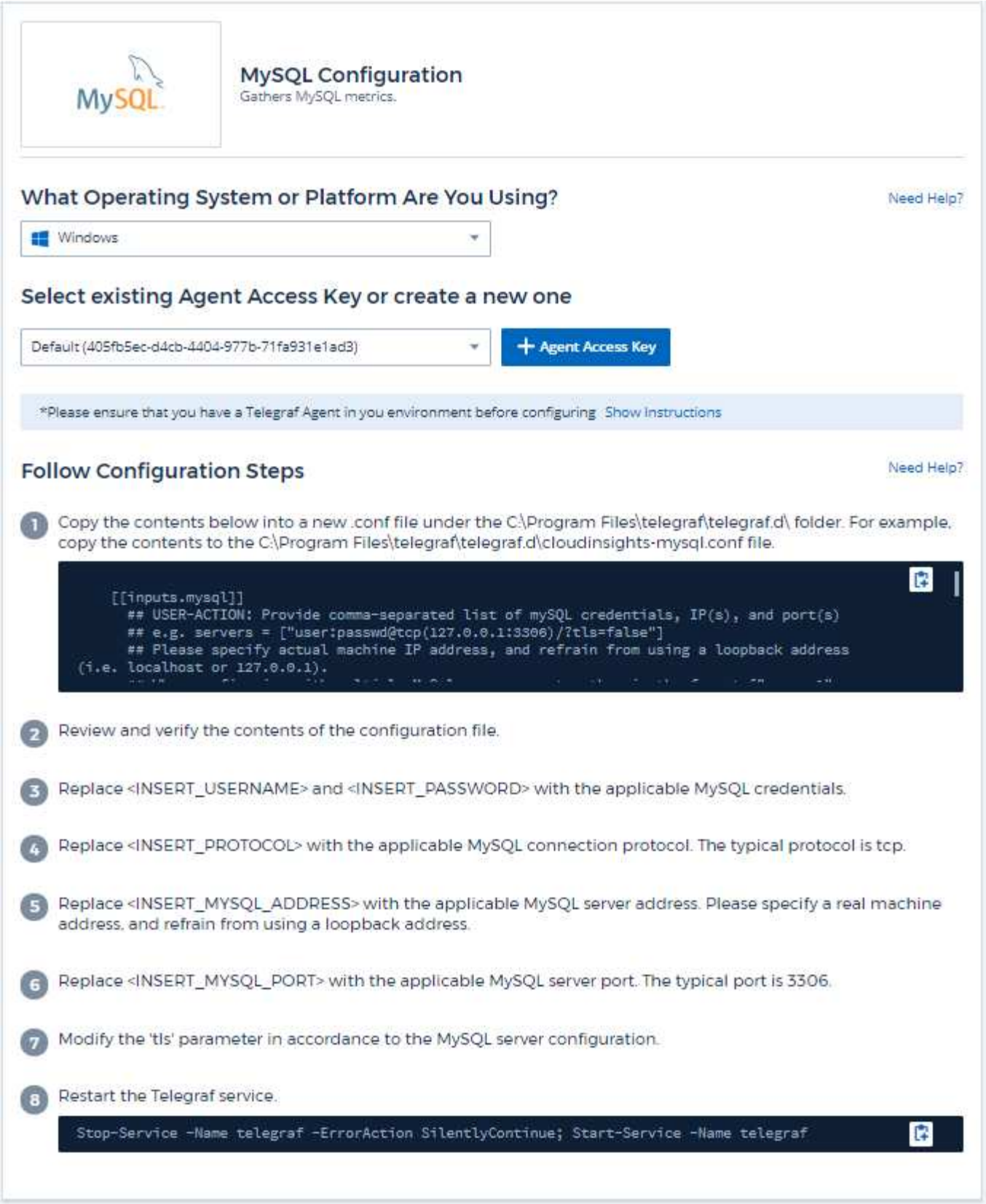
Data Infrastructure Insightsは、このデータコレクタを使用してMySQLから指標を収集します。

インストール

1. [Observability]>[Collectors]で、+ Data Collector *をクリックします。MySQLを選択します。

   Tegraf エージェントがインストールされているオペレーティングシステムまたはプラットフォームを選択します。

2. Agent for Collection をまだインストールしていない場合、または別のオペレーティングシステムまたはプラットフォームに Agent をインストールする場合は、 *Show Instructions* をクリックしてを展開します "エージェントのインストール" 手順
3. このデータコレクタで使用するエージェントアクセスキーを選択します。[**+Agent Access Key**] ボタンをクリックすると、新しいエージェントアクセスキーを追加できます。ベストプラクティス：別のエージェントアクセスキーを使用するのは、たとえば OS/ プラットフォーム別にデータコレクタをグループ化する場合だけです。
4. 設定手順に従ってデータコレクタを設定します。手順は、データの収集に使用するオペレーティングシステムまたはプラットフォームのタイプによって異なります。



セットアップ（ **Setup** ）

情報は、に記載されています "MySQL のドキュメント"。

オブジェクトとカウンタ

次のオブジェクトとそのカウンタが収集されます。

オブジェクト：	識別子：	属性：	「 **dataPoints** 」：
MySQL	ネームスペース MySQLサーバ	ノードIP ノード名	中止されたクライアント数（1秒あたり） 中断された接続数（1秒あたり） 受信バイト数（1秒あたり） 転送バイト数（1秒あたり） 管理コマンド数（1秒あたり） コマンドはイベントを変更します コマンドは機能を変更します コマンドはインスタンスを変更します コマンドは手順を変更します コマンドはサーバーを変更します コマンドはテーブルを変更します コマンドはテーブルスペースを変更します コマンドはユーザーを変更します コマンド解析（Commands Analyze コマンドはキーキャッシュに割り当てます コマンドが開始されます コマンドBinlog コマンド手順を呼び出します コマンド変更DB コマンドマスターを変更します [コマンド][リプレゼンテーションフィルタを変更] コマンドチェック コマンドチェックサム コマンドCommit コマンドCreate DB コマンドイベントを作成します コマンド機能を作成します コマンドインデックスを作成します コマンドCreate手順 コマンドサーバーを作成します コマンドテーブルを作成します

トラブルシューティング

追加情報はから入手できます "サポート" ページ

netstat Data Collector の場合

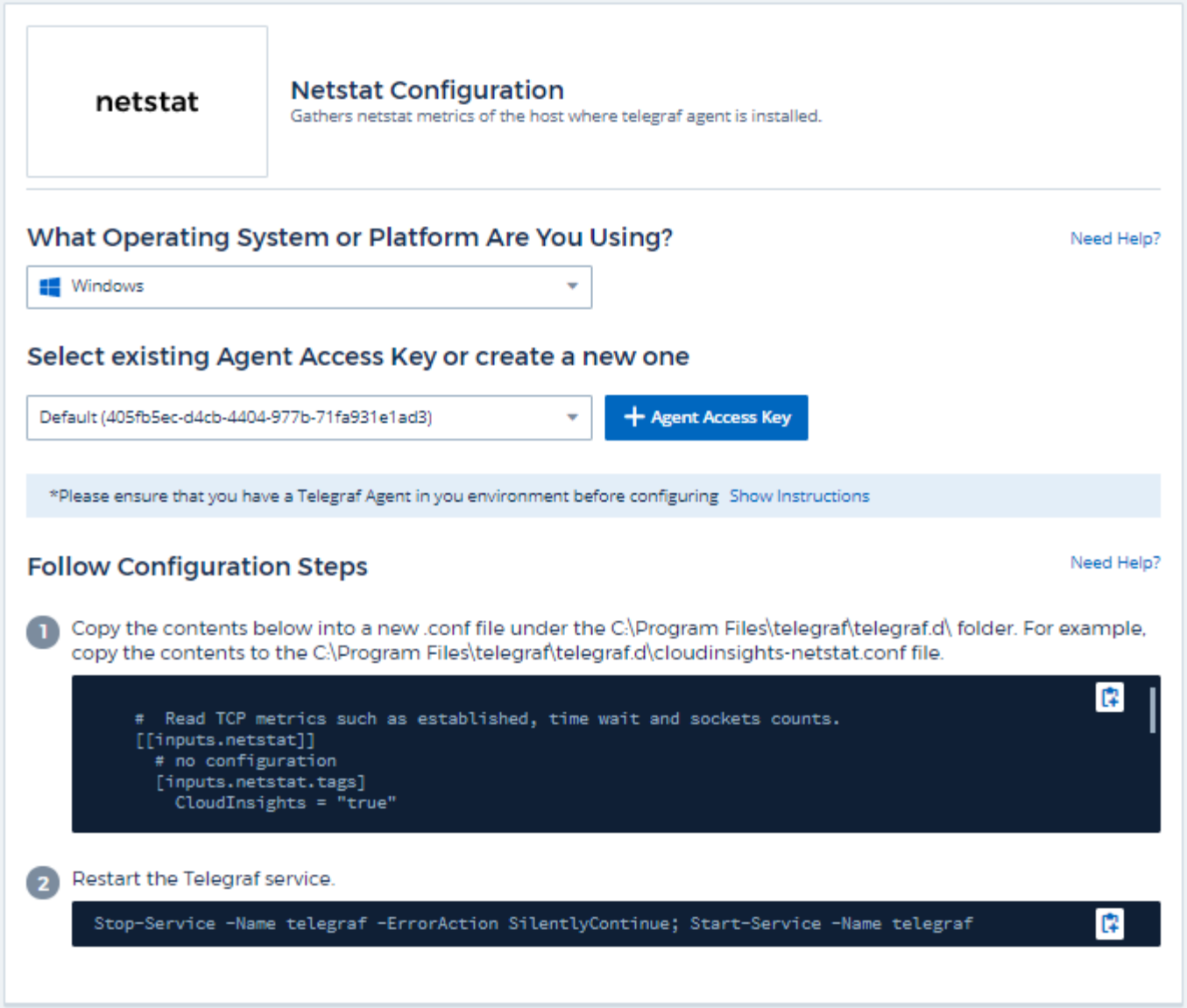
Data Infrastructure Insightsは、このデータコレクタを使用してnetstat指標を収集します。

インストール

1. [Observability]>[Collectors]で、+ Data Collector *をクリックします。netstatを選択します。

   Tegraf エージェントがインストールされているオペレーティングシステムまたはプラットフォームを選択します。

2. Agent for Collection をまだインストールしていない場合、または別のオペレーティングシステムまたはプラットフォームに Agent をインストールする場合は、 *Show Instructions* をクリックしてを展開します "エージェントのインストール" 手順
3. このデータコレクタで使用するエージェントアクセスキーを選択します。[**+Agent Access Key**] ボタンをクリックすると、新しいエージェントアクセスキーを追加できます。ベストプラクティス：別のエージェントアクセスキーを使用するのは、たとえば OS/ プラットフォーム別にデータコレクタをグループ化する場合だけです。
4. 設定手順に従ってデータコレクタを設定します。手順は、データの収集に使用するオペレーティングシステムまたはプラットフォームのタイプによって異なります。



セットアップ（ **Setup** ）

オブジェクトとカウンタ

次のオブジェクトとそのカウンタが収集されます。

オブジェクト：	識別子：	属性：	「 dataPoints 」：
netstat	ノード UUID	ノードIP ノード名	

トラブルシューティング

追加情報はから入手できます "サポート" ページ

nginx データコレクタ

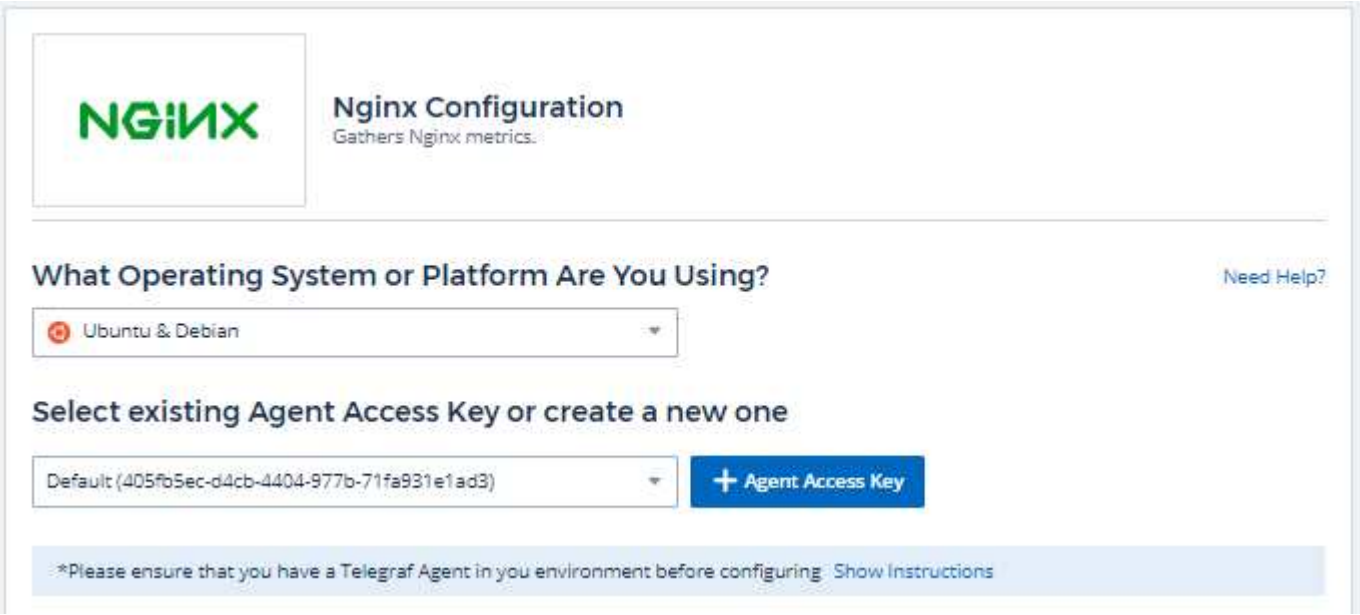
Data Infrastructure Insightsは、このデータコレクタを使用してNginxから指標を収集します。

インストール

1. [Observability]>[Collectors]で、+ Data Collector *をクリックします。[Nginx]を選択します。

   Tegraf エージェントがインストールされているオペレーティングシステムまたはプラットフォームを選択します。

2. Agent for Collection をまだインストールしていない場合、または別のオペレーティングシステムまたはプラットフォームに Agent をインストールする場合は、 *Show Instructions* をクリックしてを展開します "エージェントのインストール" 手順
3. このデータコレクタで使用するエージェントアクセスキーを選択します。[**+Agent Access Key**] ボタンをクリックすると、新しいエージェントアクセスキーを追加できます。ベストプラクティス：別のエージェントアクセスキーを使用するのは、たとえば OS/ プラットフォーム別にデータコレクタをグループ化する場合だけです。
4. 設定手順に従ってデータコレクタを設定します。手順は、データの収集に使用するオペレーティングシステムまたはプラットフォームのタイプによって異なります。



Follow Configuration Steps

Need Help?

1. If you already have a URL enabled to provide Nginx metrics, go directly to the plugin configuration.
2. Nginx metrics are available through a status page when the HTTP stub status module is enabled.Refer to the below link for verifying/enabling http_stub_status_module.

```
http://nginx.org/en/docs/http/ngx_http_stub_status_module.html
```

3. After verifying the module is enabled ,modify the Nginx configuration to set up a locally-accessible URL for the status page:

```
server {
   listen   <PORT NUMBER>;
   Please specify actual machine IP address, and refrain from using a loopback address (i.e.
localhost or 127.0.0.1)
   server_name  <IP ADDRESS>;
   location /nginx_status {
```

4. Reload the configuration:

```
nginx -s reload
```

5. Copy the contents below into a new .conf file under the /etc/telegraf/telegraf.d/ directory. For example, copy the contents to the /etc/telegraf/telegraf.d/cloudinsights-nginx.conf file.

```
    [[inputs.nginx]]
      ## USER-ACTION: Provide Nginx status url
      ## Please specify actual machine IP address where nginx_status is enabled, and refrain from
using a loopback address (i.e. localhost or 127.0.0.1).
      ## When configuring with multiple Nginx servers, enter them in the format ["url1" ,"url2",
```

6. Replace <INSERT_NGINX_ADDRESS> with the applicable Nginx address. Please specify a real machine address, and refrain from using a loopback address.
7. Replace <INSERT_NGINX_PORT> with the applicable Nginx port.
8. Restart the Telegraf service.

```
systemctl restart telegraf
```

セットアップ（ **Setup** ）

nginx メトリックの収集では、 Nginx が必要です "HTTP_STUB_STATE_MODULE" を有効にします。

追加情報は、にあります "nginx のドキュメント"。

オブジェクトとカウンタ

次のオブジェクトとそのカウンタが収集されます。

オブジェクト：	識別子：	属性：	「 **dataPoints** 」：
nginx	ネームスペース サーバ	ノードIP ノード名 ポート	受け入れます アクティブ 処理されます 読書 リクエスト 待機中です 書くこと

トラブルシューティング

追加情報はから入手できます "サポート" ページ

PostgreSQL データコレクタ

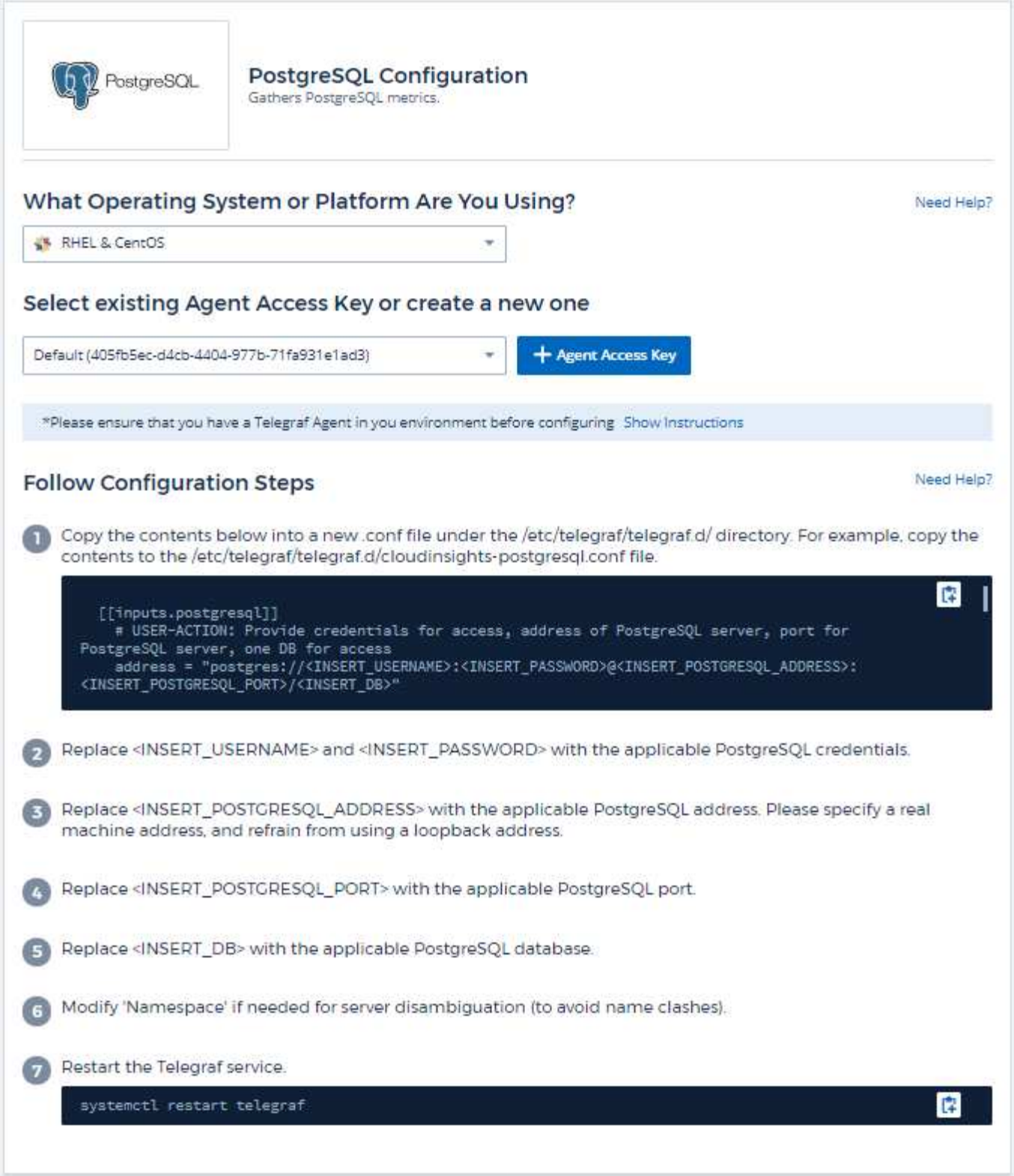
Data Infrastructure Insightsは、このデータコレクタを使用してPostgreSQLから指標を収集します。

インストール

1. [Observability]>[Collectors]で、+ Data Collector *をクリックします。PostgreSQLを選択します。

   Tegraf エージェントがインストールされているオペレーティングシステムまたはプラットフォームを選択します。

2. Agent for Collection をまだインストールしていない場合、または別のオペレーティングシステムまたはプラットフォームに Agent をインストールする場合は、 *Show Instructions* をクリックしてを展開します "エージェントのインストール" 手順
3. このデータコレクタで使用するエージェントアクセスキーを選択します。[**+Agent Access Key**] ボタンをクリックすると、新しいエージェントアクセスキーを追加できます。ベストプラクティス：別のエージェントアクセスキーを使用するのは、たとえば OS/ プラットフォーム別にデータコレクタをグループ化する場合だけです。
4. 設定手順に従ってデータコレクタを設定します。手順は、データの収集に使用するオペレーティングシステムまたはプラットフォームのタイプによって異なります。



セットアップ（ **Setup** ）

情報は、に記載されています "PostgreSQL のドキュメント"。

オブジェクトとカウンタ

次のオブジェクトとそのカウンタが収集されます。

オブジェクト：	識別子：	属性：	「 **dataPoints** 」：
PostgreSQL サーバ	ネームスペース データベース サーバ	ノード名 ノードIP	割り当てられたバッファ バッファバックエンド バッファーバックエンド ファイル同期 バッファチェックポイント バッファクリーニング Checkpoints Sync Timeの略 Checkpoints Write Timeの略 チェックポイントリクエスト チェックポイントに時間が設定されました MAX Written Clean
PostgreSQL データベース	ネームスペース データベース サーバ	データベースOID ノード名 ノードIP	Blocks Read Timeの略 Blocks Write Timeの略 ブロックヒット 読み取りをブロックします コンフリクト デッドロック クライアント番号 一時ファイルのバイト数 一時ファイル番号 行を削除しました 行がフェッチされました 挿入された行 返される行数 行を更新しました コミットされたトランザクション ロールバックされたトランザクション

トラブルシューティング

追加情報はから入手できます "サポート" ページ

Puppet Agent データコレクタ

Data Infrastructure Insightsは、このデータコレクタを使用してPuppet Agentから指標を収集します。

インストール

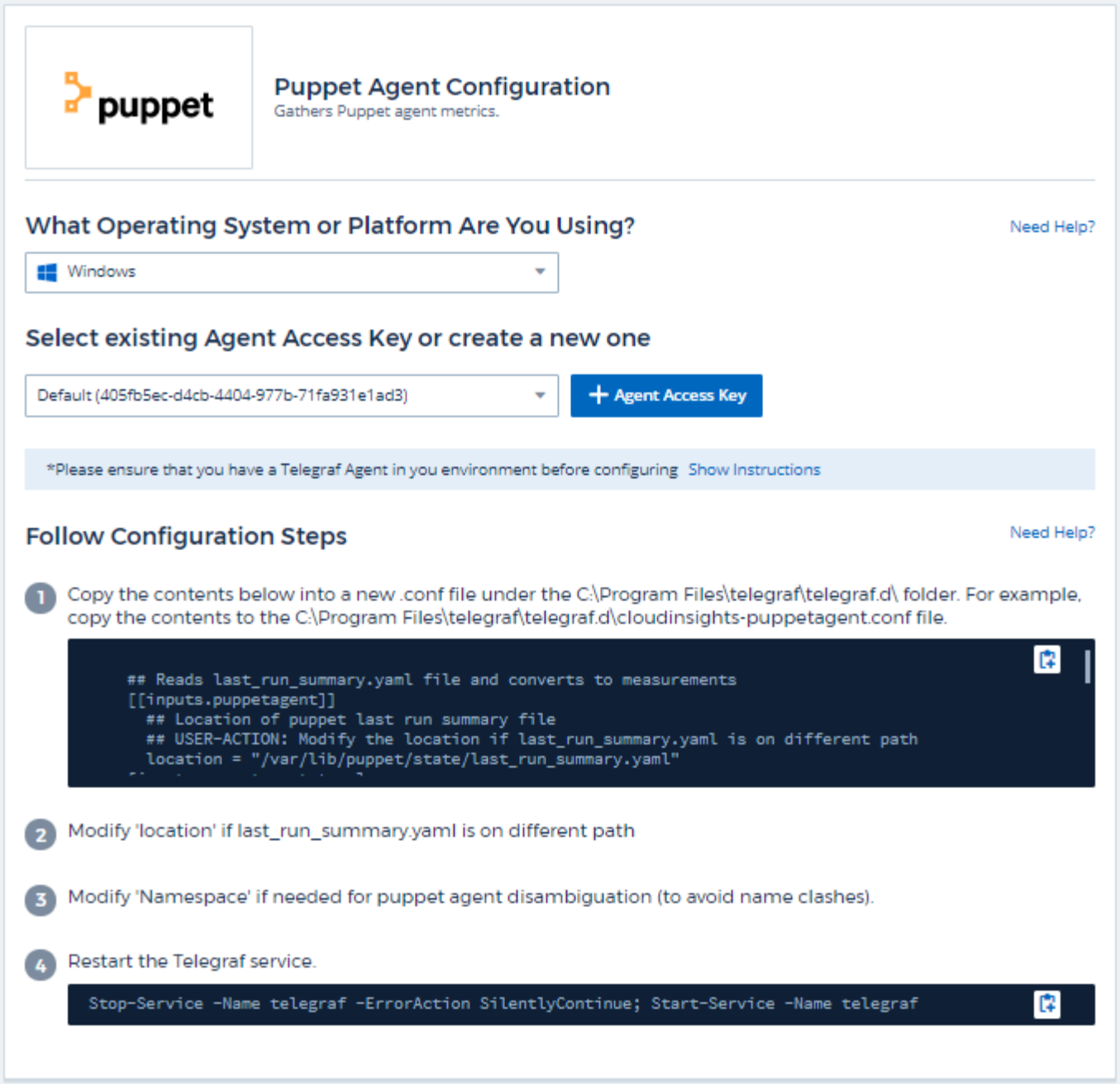
1. [Observability]>[Collectors]で、+ Data Collector *をクリックします。[Puppet]を選択します。

   Tegraf エージェントがインストールされているオペレーティングシステムまたはプラットフォームを選択します。

2. Agent for Collection をまだインストールしていない場合、または別のオペレーティングシステムまたはプラットフォームに Agent をインストールする場合は、 *Show Instructions* をクリックしてを展開します "エージェントのインストール" 手順

3. このデータコレクタで使用するエージェントアクセスキーを選択します。[**+Agent Access Key**] ボタンをクリックすると、新しいエージェントアクセスキーを追加できます。ベストプラクティス：別のエージェントアクセスキーを使用するのは、たとえば OS/ プラットフォーム別にデータコレクタをグループ化する場合だけです。

4. 設定手順に従ってデータコレクタを設定します。手順は、データの収集に使用するオペレーティングシステムまたはプラットフォームのタイプによって異なります。



セットアップ（ **Setup** ）

情報は、に記載されています "Puppet のドキュメント"

オブジェクトとカウンタ

次のオブジェクトとそのカウンタが収集されます。

オブジェクト：	識別子：	属性：	「 dataPoints 」：

Puppet Agent	ネームスペース ノード UUID	ノード名 場所 ノードIP バージョン設定文字列 バージョンPuppet	合計を変更します イベント失敗 イベント成功 イベント合計 リソースが変更されました リソースに失敗しました リソースを再起動できませんでした リソースOutofsync リソースが再起動しました リソースがスケジュールされました リソースがスキップされました リソース合計 タイムアンカー Time Configretrievalの略 タイムcron タイム・エグゼクティブ 時間ファイル（Time File） Time Filebucketの略 タイムラストラン タイムパッケージ タイムスケジュール タイムサービス 時間Sshauthorizedキー 合計時間 タイムユーザー

トラブルシューティング

追加情報はから入手できます "サポート" ページ

Redis Data Collector の場合

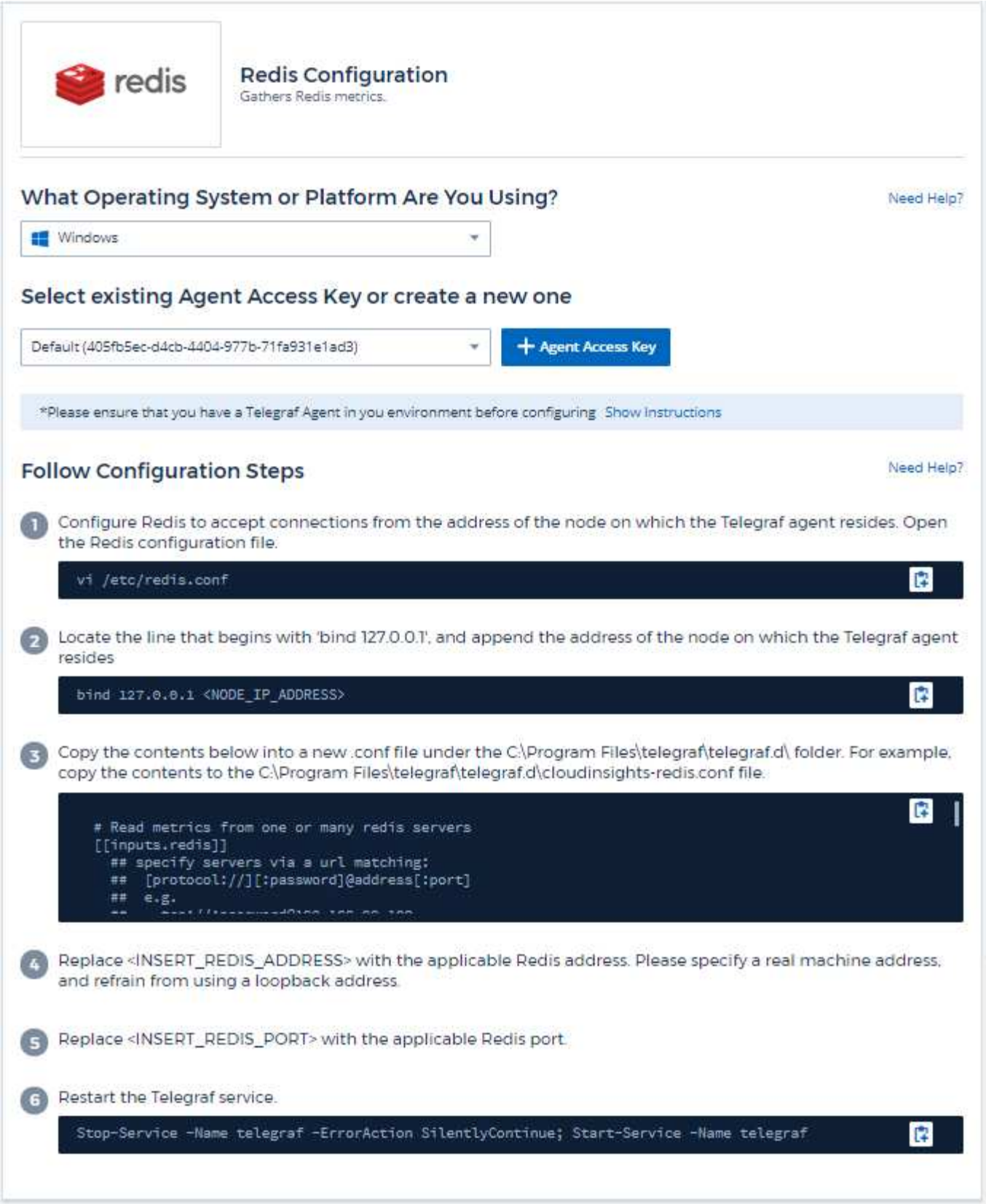
Data Infrastructure Insightsは、このデータコレクタを使用してRedisから指標を収集します。Redis は、データベース、キャッシュ、メッセージブローカーとして使用されるオープンソースのインメモリデータ構造ストアで、文字列、ハッシュ、リスト、セットなどのデータ構造をサポートしています。

インストール

1. [Observability]>[Collectors]で、+ Data Collector *をクリックします。Redisを選択します。

   Tegraf エージェントがインストールされているオペレーティングシステムまたはプラットフォームを選択します。

2. Agent for Collection をまだインストールしていない場合、または別のオペレーティングシステムまたはプラットフォームに Agent をインストールする場合は、 *Show Instructions* をクリックしてを展開します "エージェントのインストール" 手順
3. このデータコレクタで使用するエージェントアクセスキーを選択します。[**+Agent Access Key**] ボタンをクリックすると、新しいエージェントアクセスキーを追加できます。ベストプラクティス：別のエージェントアクセスキーを使用するのは、たとえば OS/ プラットフォーム別にデータコレクタをグループ化する場合だけです。
4. 設定手順に従ってデータコレクタを設定します。手順は、データの収集に使用するオペレーティングシステムまたはプラットフォームのタイプによって異なります。



セットアップ（ **Setup** ）

情報は、に記載されています "Redis 文書"。

オブジェクトとカウンタ

次のオブジェクトとそのカウンタが収集されます。

オブジェクト：	識別子：	属性：	「 **dataPoints** 」：
Redis	ネームスペース サーバ		

トラブルシューティング

追加情報はから入手できます "サポート" ページ

著作権に関する情報

Copyright © 2024 NetApp, Inc. All Rights Reserved. Printed in the U.S.このドキュメントは著作権によって保護されています。著作権所有者の書面による事前承諾がある場合を除き、画像媒体、電子媒体、および写真複写、記録媒体、テープ媒体、電子検索システムへの組み込みを含む機械媒体など、いかなる形式および方法による複製も禁止します。

ネットアップの著作物から派生したソフトウェアは、次に示す使用許諾条項および免責条項の対象となります。

このソフトウェアは、ネットアップによって「現状のまま」提供されています。ネットアップは明示的な保証、または商品性および特定目的に対する適合性の暗示的保証を含み、かつこれに限定されないいかなる暗示的な保証も行いません。ネットアップは、代替品または代替サービスの調達、使用不能、データ損失、利益損失、業務中断を含み、かつこれに限定されない、このソフトウェアの使用により生じたすべての直接的損害、間接的損害、偶発的損害、特別損害、懲罰的損害、必然的損害の発生に対して、損失の発生の可能性が通知されていたとしても、その発生理由、根拠とする責任論、契約の有無、厳格責任、不法行為（過失またはそうでない場合を含む）にかかわらず、一切の責任を負いません。

ネットアップは、ここに記載されているすべての製品に対する変更を随時、予告なく行う権利を保有します。ネットアップによる明示的な書面による合意がある場合を除き、ここに記載されている製品の使用により生じる責任および義務に対して、ネットアップは責任を負いません。この製品の使用または購入は、ネットアップの特許権、商標権、または他の知的所有権に基づくライセンスの供与とはみなされません。

このマニュアルに記載されている製品は、1つ以上の米国特許、その他の国の特許、および出願中の特許によって保護されている場合があります。

権利の制限について：政府による使用、複製、開示は、DFARS 252.227-7013（2014年2月）およびFAR 5252.227-19（2007年12月）のRights in Technical Data -Noncommercial Items（技術データ - 非商用品目に関する諸権利）条項の(b)(3)項、に規定された制限が適用されます。

本書に含まれるデータは商用製品および / または商用サービス（FAR 2.101の定義に基づく）に関係し、データの所有権はNetApp, Inc.にあります。本契約に基づき提供されるすべてのネットアップの技術データおよびコンピュータ ソフトウェアは、商用目的であり、私費のみで開発されたものです。米国政府は本データに対し、非独占的かつ移転およびサブライセンス不可で、全世界を対象とする取り消し不能の制限付き使用権を有し、本データの提供の根拠となった米国政府契約に関連し、当該契約の裏付けとする場合にのみ本データを使用できます。前述の場合を除き、NetApp, Inc.の書面による許可を事前に得ることなく、本データを使用、開示、転載、改変するほか、上演または展示することはできません。国防総省にかかる米国政府のデータ使用権については、DFARS 252.227-7015(b)項（2014年2月）で定められた権利のみが認められます。

商標に関する情報

NetApp、NetAppのロゴ、http://www.netapp.com/TMに記載されているマークは、NetApp, Inc.の商標です。その他の会社名と製品名は、それを所有する各社の商標である場合があります。