



# Data Collector Reference - サービス

## Data Infrastructure Insights

NetApp  
December 19, 2024

This PDF was generated from [https://docs.netapp.com/ja-jp/data-infrastructure-insights/task\\_config\\_telegraf\\_node.html](https://docs.netapp.com/ja-jp/data-infrastructure-insights/task_config_telegraf_node.html) on December 19, 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	3
Apache Data Collector	5
総領事データ収集	8
Couchbase Data Collector	9
CouchDB データコレクタ	11
Docker Data Collector	13
Elasticsearch Data Collector	20
Flink Data Collector の使用	22
Hadoop Data Collector	27
HAProxy Data Collector	33
JVM Data Collector ( JVM データ収集)	39
Kafka Data Collector の利用	44
Kibana データコレクタ	47
Kubernetes Monitoring Operatorのインストールと設定	49
Memcached Data Collector	66
MongoDB データコレクタ	69
MySQL データコレクタ	71
netstat Data Collector の場合	74
nginx データコレクタ	75
PostgreSQL データコレクタ	78
Puppet Agent データコレクタ	80
Redis Data Collector の場合	82

# 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 デバイスのパスタイプ	ノード IP ノード名ノード OS モード	空き inode 空き inode 使用済み合計使用済み合計 inode 数
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 時間進行中の合計 IOPS 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）加重 IO 時間合計書き込みバイト数（1 秒あたり）書き込み時間合計書き込み時間（1 秒あたり）現在のディスクキュー長書き込み時間読み取り時間（I/O 時間）
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 使用率ユーザー 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 」：
ノードネットワーク	ネットワークインターフェイスのノード UUID	ノード名ノード IP ノード OS	送信された受信バイト数 パケットを送信送信パケットを破棄しました送信エラーパケット受信破棄されたパケット受信エラーパケット受信したパケット送信しました

## セットアップ

セットアップとトラブルシューティングの情報については、ページを参照し["エージェントの設定"](#)をご覧ください。

## 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. コレクション用エージェント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用のエージェントをインストールする場合は、[Show Instructions]をクリックして手順を展開し["エージェントのインストール"](#)ます。
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。



## ActiveMQ Configuration

Gathers ActiveMQ 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 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-activemq.conf file.

```
[[inputs.activemq]]
  ## Required ActiveMQ Endpoint, port
  ## USER-ACTION: Provide address of ActiveMQ, HTTP port for ActiveMQ
  server = "<INSERT_ACTIVEMQ_ADDRESS>"
  port = <INSERT_ACTIVEMQ_PORT>
```

- 2 Replace <INSERT\_ACTIVEMQ\_ADDRESS> with the applicable ActiveMQ server address. Please specify a real machine address, and refrain from using a loopback address.
- 3 Replace <INSERT\_ACTIVEMQ\_PORT> with the applicable ActiveMQ server HTTP port.
- 4 Replace <INSERT\_ACTIVEMQ\_USERNAME> and <INSERT\_ACTIVEMQ\_PASSWORD> with the applicable ActiveMQ credentials.
- 5 Modify 'webadmin' if needed (if ActiveMQ server changes web admin root path).
- 6 Modify 'Namespace' if needed for server disambiguation (to avoid name clashes).
- 7 Restart the Telegraf service.

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

## セットアップ

詳細については、"[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	消費者数キューカウント キューサイズ
ActiveMQ サブスクライバ	クライアント ID 接続 ID ポートサーバの名前空間	は、アクティブなデステイネーションノード名ノード IP ノード UUID ノード OS セレクタサブスクリプションです	Dequeue Count Depatched Count DeQueue Size Dequeue Count Pending Queue Size (デキューカウント ディスパッチキューカウント キューサイズ)
ActiveMQ トピック	トピックポートサーバの名前空間	ノード名ノード IP ノード UUID ノード OS	消費者数キューカウント キューカウントサイズ

## トラブルシューティング

詳細については、ページを参照して"[サポート](#)"ください。

## Apache Data Collector

このデータコレクタを使用すると、テナント上の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](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. コレクション用エージェント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用のエージェントをインストールする場合は、[Show Instructions]をクリックして手順を展開し"[エージェントのインストール](#)"ます。
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。



## Apache Configuration

Gathers Apache metrics.

---

### What Operating System or Platform Are You Using? [Need Help?](#)

Ubuntu & Debian

### 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 Ensure that the Apache HTTP Server system you're going to gather metrics on has the 'mod\_status' module enabled and exposed. 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-apache.conf file.

```
[[inputs.apache]]
## An array of URLs to gather from, must be directed at the machine
## readable version of the mod_status page including the auto query string.
## USER-ACTION: Provide address of apache server, port for apache server, confirm path for
server-status.
## Please provide a real machine IP address and refrain from using a loopback address.

```
- 3 Replace <INSERT\_APACHE\_ADDRESS> with the applicable Apache server address. Please specify a real machine address, and refrain from using a loopback address.
- 4 Replace <INSERT\_APACHE\_PORT> with the applicable Apache server port.
- 5 Modify the '/server-status' path in accordance to the Apache server configuration.
- 6 Restart the Telegraf service.

```
systemctl restart telegraf
```

## セットアップ

Telegraf の Apache の HTTP Server 用プラグインは 'mod\_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
```

「MOD\_STATUS」モジュールの詳細な手順については、["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 Port Parent Server Config Generation Parent Server MPM Generation Server Uptime が停止しています	ビジーワーカーバイト / 要求バイト / 秒 CPU 子 システム CPU 子ユーザ CPU 負荷 CPU システム CPU ユーザ非同期接続ク ローズ非同期接続キーブ アライブ非同期接続書き 込み接続維持アイドル接 続書き込み接続書き込み 接続要求ごとの合計時間 アイドルワーカー負荷平 均 (過去 1 分) 負荷平均 (過去 15 分) プロセス リクエスト数 / 秒合計ア クセス時間合計合計 KBytes スコアボードクロ ーディングスコアボード DNS ルックアップスコア ボード仕上げスコアボー ドアイドルクリーンアッ プスコアボードキーブア ライブスコアボードログ スコアボードオープンス コアボードスコアボード 閲覧スコアボードスコア ボード送信スコアボード 開始スコアボード待機中

## トラブルシューティング

詳細については、ページを参照して["サポート"](#)ください。

## 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。

収集用のエージェントを設定していない場合は、テナントで要求され["エージェントをインストールします"](#)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

詳細については、を参照して["総領事からのご説明"](#)ください。

## 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	ネームスペースチェック ID サービスノード	ノード IP ノード OS ノード UUID ノードネームサ ービス名チェックネーム サービス ID ステータス	重大な合格警告

## トラブルシューティング

詳細については、ページを参照して["サポート"](#)ください。

# 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. コレクション用エージェント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用のエージェントをインストールする場合は、[Show Instructions]をクリックして手順を展開し["エージェントのインストール"](#)ます。
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。



## Couchbase Configuration

Gathers Couchbase 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 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-couchbase.conf file.

```
## Read metrics from one or many couchbase clusters
[[inputs.couchbase]]
  ## specify servers via a url matching:
  ## [protocol://][:password]@address[:port]
  ## e.g.
  ## http://username:password@127.0.0.1:8090
```

- 2 Replace <INSERT\_USERNAME> and <INSERT\_PASSWORD> with couchbase server account credentials.
- 3 Replace <INSERT\_COUCHBASE\_ADDRESS> with the applicable Couchbase address. Please specify a real machine address, and refrain from using a loopback address.
- 4 Replace <INSERT\_COUCHBASE\_PORT> with the applicable Couchbase port.
- 5 Restart the Telegraf service.

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

## セットアップ

詳細については、を参照して"[Couchbase ドキュメント](#)"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 」：
Couchbase Node	ネームスペースクラスタの Couchbase ノードホスト名	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	Data Used Data fetches Disk Used Item Count Memory Used Operations per Second Quota Used (使用データフェッチディスクの使用アイテム数 1 秒あたりのメモリ使用量)

## トラブルシューティング

詳細については、ページを参照して["サポート"](#)ください。

## 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. コレクション用エージェント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用のエージェントをインストールする場合は、[Show Instructions]をクリックして手順を展開し["エージェントのインストール"](#)ます。
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。



## CouchDB Configuration

Gathers CouchDB metrics.

### What Operating System or Platform Are You Using?

[Need Help?](#)

RHEL & CentOS

### 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 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-couchdb.conf file.

```
## Read CouchDB Stats from one or more servers
[[inputs.couchdb]]
  ## Works with CouchDB stats endpoints out of the box
  ## Multiple Hosts from which to read CouchDB stats:
  ## USER-ACTION: Provide comma-separated list of couchdb IP(s) and port(s).
```

- 2 Replace <INSERT\_COUCHDB\_ADDRESS> with the applicable CouchDB address. Please specify a real machine address, and refrain from using a loopback address.
- 3 Replace <INSERT\_COUCHDB\_PORT> with the applicable CouchDB port.
- 4 Modify the URL if CouchDB monitoring is exposed at different path
- 5 Restart the Telegraf service.

```
systemctl restart telegraf
```

## セットアップ

詳細については、[を参照して"CouchDB のドキュメント"](#)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 」：
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 データベース書き込み データベースオープン OS ファイル最大要求時間 最小要求時間 httpd 要求メソッドのコピー httpd 要求メソッドの削除 httpd 要求メソッドの削除 httpd 要求メソッドの取得 httpd 要求メソッドのヘッド httpd 要求メソッドの POST httpd 要求メソッドのステータスコード 200 201 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

詳細については、ページを参照して["サポート"](#)ください。

## 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。

収集用のエージェントを設定していない場合は、テナントで要求され["エージェントをインストールします"](#)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。



## Docker Configuration

Gathers Docker metrics.

### What Operating System or Platform Are You Using?

[Need Help?](#)

RHEL & CentOS

### 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 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-docker.conf` file.

```
[[inputs.docker]]
  ## Docker Endpoint
  ## To use TCP, set endpoint = "tcp://[ip]:[port]". By default, Docker uses port 2375 for
  unencrypted and 2376 for encrypted
  ## To use environment variables (ie, docker-machine), set endpoint = "ENV"
```

- 2 Replace `<INSERT_DOCKER_ENDPOINT>` with the applicable Docker endpoint.
- 3 Modify 'Namespace' if needed for server disambiguation (to avoid name clashes).
- 4 Restart the Telegraf service.

```
systemctl restart telegraf
```

## セットアップ

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

Telegraf 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	ノード名ノード IP ノード UUID ノード OS Kubernetes クラスタ Docker バージョンユニット	メモリコンテナコンテナ一時停止コンテナコンテナ停止 CPU 実行ルーチンイメージリスナーイベント使用ファイル記述子データ使用可能データ合計データ使用メタデータ使用可能メタデータ使用ブルックサイズ合計

オブジェクト：	識別子：	属性：	「 dataPoints 」：
Docker コンテナ	名前空間コンテナ名 Docker Engine	<p>Kubernetes コンテナコンテナハッシュ Kubernetes コンテナポート  Kubernetes コンテナ再起動カウント Kubernetes コンテナ終了メッセージパス Kubernetes コンテナ終了メッセージポリシー  Kubernetes ポッド終了猶予期間コンテナイメージコンテナ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  Kubernetes コンテナ  Kubernetes Kubernetes  Kubernetes Kubernetes  Kubernetes タイプ  Kubernetes ポッド名  Kubernetes ポッドネームスペース Kubernetes ポッドポッド 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 の確認 Kubernetes IO 構成ソース OpenShift IO  SCC Kubernetes 概要  Kubernetes Kubernetes 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ライセンススキーマ構築日スキーマネームスキーマ 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メンテナ顧客ポッド  Kubernetes 状態セットポッド名テナント  WebConsole アーキテクチャ信頼できるソース URL ビルド日付 RH ビルドホスト RH コンポーネント配布範囲インストールリリース実行概要 VCS タイプベンダーバージョン健全性ステータスコンテナ ID</p>	<p>Memory Active  Anonymous Memory アクティブな匿名メモリキャッシュメモリ階層制限メモリ非アクティブな匿名メモリ非アクティブメモリメモリメモリメモリメモリ最大使用メモリページフォールトメモリページメモリページメモリ内でページアウトされたメモリ常駐セットサイズメモリ常駐セット全体アクティブ 匿名メモリアクティブファイルメモリ合計キャッシュメモリ非アクティブ匿名メモリ合計非アクティブファイルメモリ合計マップファイルメモリ総ページフォールトメモリ総ページ主要エラーメモリ合計ページインメモリ総ページアウトメモリ総常駐セットサイズメモリ合計常駐セットサイズヒュージメモリ合計  Unevictable Memory  Unevictable Memory  Usage Memory Usage  Percent Exit Code OOM killed PID Started at Failing Streak  (Unevictable メモリ使用率終了コード OOM がエラー終了時に PID を終了しました</p>

オブジェクト：	識別子：	属性：	「 dataPoints 」：
Docker コンテナブロック IO	名前空間コンテナ名 Device Docker Engine	Kubernetes コンテナコンテナハッシュ Kubernetes コンテナポート Kubernetes コンテナ再起動カウント Kubernetes コンテナ終了メッセージパス Kubernetes コンテナ終了メッセージポリシー Kubernetes ポッド終了猶予期間 コンテナイメージ コンテナステータス コンテナバージョン ノード名 Kubernetes コンテナログパス Kubernetes Kubernetes コンテナ Kubernetes Kubernetes Kubernetes タイプ Kubernetes ポッド名 Kubernetes ポッドネームスペース Kubernetes ポッドポッド 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 構成表示 Kubernetes 構成ソース OpenShift SCC Kubernetes 概要 Kubernetes Kubernetes 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 Service Schema Build Date スキーマ ネーム スキーマ ベンダー カスタマー ポッド Kubernetes 静的状態 設定 ポッド名 テナント WebConsole 構築日 ライセンス ベンダー アーキテクチャ 信頼できる ソース URL RH ビルド ホスト RH コンポーネント 配布範囲 インストール メンテナリ リース 実行概要 削除 VCS リファレンス バージョン スキーマ URL スキーマ バージョン コンテナ ID	IO サービス バイト再帰的 非同期 IO サービス バイト再帰的 IO サービス バイト再帰的 IO サービス バイト再帰的 な総 IO サービス バイト再帰的 IO サービス 対象再帰的 IO サービス 対象再帰的 IO サービス 対象再帰的 IO サービス 対象の再帰的 な総 IO サービス 対象 IO

オブジェクト：	識別子：	属性：	「 dataPoints 」：
Docker コンテナネットワーク	名前空間コンテナネットワーク Docker エンジン	コンテナイメージコンテナステータスコンテナノード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 dropped RX bytes RX Errors RX Packets TX dropped TX Bytes TX Errors TX パケット



## 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。	Tegraf エージェントログを調べて、次のエラーが報告されているかどうかを確認します。プラグインのエラー [input.docdocker ]: Docker デーモンソケットに接続しようとしたときに許可が拒否されました。接続されている場合は、事前に指定した Docker Unix ソケットへの Telegraf エージェントアクセスを提供するために必要な手順を実行してください。

詳細については、ページを参照して["サポート"](#)ください。

## 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. コレクション用エージェント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用のエージェントをインストールする場合は、[Show Instructions]をクリックして手順を展開し["エージェントのインストール"](#)ます。
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。



## Elasticsearch Configuration

Gathers Elasticsearch metrics.

### What Operating System or Platform Are You Using?

[Need Help?](#)

Ubuntu & Debian

### 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 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-elasticsearch.conf file.

```
[[inputs.elasticsearch]]
  ## USER-ACTION: Provide comma-separated list of Elasticsearch servers.
  ## Note that for scenarios in which metrics from multiple Elasticsearch clusters are being
  ## sent to Cloud Insights, the Elasticsearch cluster names must be unique.
  ## Please specify actual machine IP address, and refrain from using a loopback address
```

- 2 Replace <INSERT\_ELASTICSEARCH\_ADDRESS> with the applicable Elasticsearch address. Please specify a real machine address, and refrain from using a loopback address.
- 3 Replace <INSERT\_ELASTICSEARCH\_PORT> with the applicable Elasticsearch port.
- 4 Restart the Telegraf service.

```
systemctl restart telegraf
```

## セットアップ

詳細については、[を参照して"Elasticsearch のドキュメント"](#)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：
Elasticsearch クラスタ	ネームスペースクラスタ	ノード IP ノード名クラスタステータス

オブジェクト：	識別子：	属性：
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 ノードです	ゾーン ID

## トラブルシューティング

詳細については、ページを参照して["サポート"](#)ください。

## 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. コレクション用エージェント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用のエージェントをインストールする場合は、[Show Instructions]をクリックして手順を展開し["エージェントのインストール"](#)ます。
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 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  
  ## USER-ACTION: Provide address(es) of flink Task 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

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。テストされた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する必要があります。

### 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する必要があります。

## 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	ネットワーク使用可能なメモリセグメントネットワーク合計メモリセグメントガベージコレクション PS MarkSweep 回数ガベージコレクション PS MarkSweep 時間ガベージコレクション PS スカベンジコレクション PS スカベンジカウントガベージコレクション PS スカベンジカウント時間ヒープメモリのコミットヒープメモリ最大ヒープメモリ使用スレッド数デーモンスレッド数スレッド数開始済みの合計
フリックジョブ	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 の最後のチェックポイント外部パス再起動時間	ダウンタイムフルリスタート最後のチェックポイント位置合わせのバッファ最後のチェックポイント期間最後のチェックポイントサイズ完了したチェックポイント数失敗したチェックポイント数アップタイムの進行中のチェックポイント数
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 Count ガベージコレクション PS MarkSweep Time ガベージコレクション PS スカベンジコレクション PS スカベンジコレクション時間ヒープメモリコミットヒープメモリ最大ヒープメモリ使用数登録済みタスクマネージャ数実行中タスクスロット使用可能なタスクスロット合計スレッド数デーモンスレッド数 最大スレッド数スレッド数の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	サーバーノード名サブタ スク名サブタスクインデ ックスタスク試行情 ID タス ク試行情番号タスク名タス クマネージャ ID ノード IP 現在の入力ウォーターマ ーク	キュー内のプール使用バッ ファプール使用バッファ アプール使用バッファ アウトプール使用バッファ ローカル数秒あたりのロー カル数カウント数バッ ファローカル秒あたりの ローカル数カウント数バッ ファリモート秒あたりの リモート数バッファ数の リモート数カウント数バッ ファ数リモートのバッ ファ数 / 秒数バッファ数 リモートのバッファ数 Second Rate Number Buffers Out Number Buffers Out Number Buffers Out per Second Count Number Buffer Out per Second Count Local Number In Local Number Count Number Number Bytes in Local per Second Rate Number バイト in Local Number Number Number Number Number Number Number Number Number バイト in Remote Number Number バイト in Remote Number Number バイト数 1 秒あたりのレ ート数 1 秒あたりのバ イト数 1 秒あたりのバ イト数 1 秒あたりのレ コード数

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 のタスク ID	サーバーノード名ジョブ 名オペレータ名サブタス 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	現在の入力ウォーターマ ーク現在の出力ウォーター マーク数レコード / 秒 数カウント数レコード / 秒あたりのレコード数レ コード数 1 秒あたりのレ コード数レコード数秒あ たりのレコード数遅延レ コード数割り当て済みパ ーティション数消費済み 率コミット平均コミット レイテンシ 最大コミット 率コミット失敗したコミ ット成功した接続クロー ズ率接続数接続作成率平 均フェッチレイテンシ最 大フェッチ速度平均フェ 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 ハートビート率受信バイ トレート I/O 平均時間 ( ns ) IO 待機率 I/O 待機 時間平均 ( ns ) 加入時間 平均最終ハートビート時 間平均ネットワーク IO レ ート発信バイトレートレ コード消費率レコード要 求あたりの最大レコード 遅延平均要求レート最大 応答速度選択レート同期 速度同期時間平均ハート ビート応答 Time Max Join Time Max Sync Time Max の最大同期時間

## トラブルシューティング

詳細については、ページを参照して["サポート"](#)ください。

## 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。

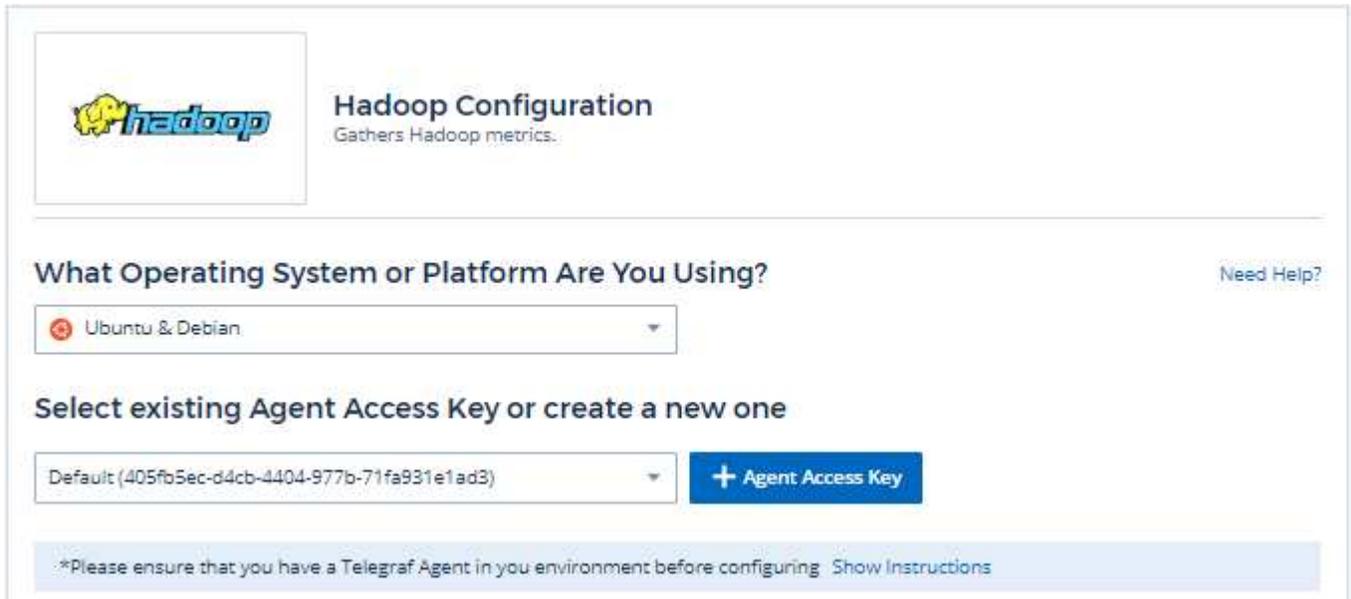
### インストール

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。

- コレクション用エージェント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用のエージェントをインストールする場合は、[*Show Instructions*]をクリックして手順を展開し"[エージェントのインストール](#)"ます。
- 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。
- 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。



 **Hadoop Configuration**  
Gathers Hadoop metrics.

**What Operating System or Platform Are You Using?** [Need Help?](#)

Ubuntu & Debian

**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 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  
  ## Please specify real machine address and refrain from using a loopback address
```

- 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

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。テストされた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、<hadoop\_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：
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
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
Hadoop ResourceManager	クラスタネームスペースサーバ	ノード名ノード IP
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
Hadoop NameNode	クラスタネームスペースサーバ	ノード名ノード IP トランザクション ID 前回の書き込み時間最後にロードしたあとの編集 HA 状態ファイルシステム状態ブロックプール ID クラスタ ID コンパイル情報別バージョン数
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

## トラブルシューティング

詳細については、ページを参照して["サポート"](#)ください。

## 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. コレクション用エージェント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用のエージェントをインストールする場合は、[*Show Instructions*]をクリックして手順を展開し"[エージェントのインストール](#)"ます。
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。



## HAProxy Configuration

Gathers HAProxy metrics.

### What Operating System or Platform Are You Using?

[Need Help?](#)

Ubuntu & Debian

### 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 Ensure that the HAProxy system you're going to gather metrics on has 'stats enable' option. 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-haproxy.conf` file.

```
# Read metrics of HAProxy, via socket or HTTP stats page
[[inputs.haproxy]]
  ## An array of address to gather stats about. Specify an ip on hostname
  ## with optional port. ie localhost, 10.10.3.33:1936, etc.
  ## Make sure you specify the complete path to the stats endpoint
  ## <url> for the endpoint. ie http://10.10.3.33:1936/haproxy?stats
```

- 3 Replace `<INSERT_HAPROXY_ADDRESS>` with the applicable HAProxy server address. Please specify a real machine address, and refrain from using a loopback address.
- 4 Replace `<INSERT_HAPROXY_PORT>` with the applicable HAProxy server port.
- 5 Modify the `'haproxy?stats'` path in accordance to the HAProxy server configuration.
- 6 Modify `'username'` and `'password'` in accordance to the HAProxy server configuration (if credentials are required).
- 7 Modify `'Namespace'` if needed for server disambiguation (to avoid name clashes).
- 8 Restart the Telegraf service.

```
systemctl restart telegraf
```

## セットアップ

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
```

完全で最新の手順については、を参照して["HAProxy のドキュメント"](#)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 」：
HAProxy フロントエンド	ネームスペースアドレス 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	バイト数キャッシュヒットキャッシュ参照キャッシュルックアップ圧縮バイトをバイパス圧縮バイトをバイパス圧縮した圧縮バイトをアウト圧縮応答を転送接続レート接続レート最大接続数セキュリティから拒否された接続ルール要求による拒否された総接続数セキュリティの懸念による拒否された応答セッションルール要求エラー応答 1xx 応答 2xx 応答 3xx 応答 4xx 応答 5xx 応答その他の要求代行受信セッションレート最大要求レート最大要求数最大要求数最大セッション総セッション数合計要求の書き換え
HAProxy サーバ	ネームスペースアドレス プロキシサーバ	ノード IP ノード名のチェック終了時間のチェック構成のチェック健全性のチェック構成のチェック構成のチェックステータスのチェックプロキシ ID の最終変更時刻最終セッションタイムモードプロセス ID サーバー ID のステータスの重み	アクティブサーババックアップサーババイト数 ( バイト ) チェックアウトダウンクライアント接続の失敗平均ダウンタイム合計拒否応答接続エラー応答 1xx 応答 2xx 応答 4xx 応答 4xx 応答その他サーバ選択キュー現在キュー最大キュー平均セッション数 / 1 秒あたりの 2 番目のセッション最大接続再使用応答時間平均セッション最大サーバ転送セッションの中止合計セッション時間平均要求再ディスパッチ要求再書き込み要求の再実行

オブジェクト：	識別子：	属性：	「 dataPoints 」：
HAProxy バックエンド	ネームスペースアドレス プロキシ	ノード IP ノード名プロキシ ID 最終変更時刻最終セッション時間モードプロセス ID サーバー ID セッション制限ステータス重み	アクティブサーババックアップサーババイト数キャッシュヒットキャッシュ検索数キャッシュ検索数チェックダウンクライアントが圧縮バイトをバイパス圧縮バイト数をバイパス圧縮バイト数を無視圧縮応答をアウト接続接続平均ダウンタイム時間セキュリティの懸念応答拒否された応答接続エラー応答 1xx 応答 2xx 応答 3xx 応答 4xx 応答 5xx 応答その他のサーバ選択キュー現在キュー最大キュー 1 秒あたりのセッション最大キュー平均時間セッション 1 秒あたりの最大要求接続再使用応答時間平均セッション最大サーバ転送総セッション数総セッション時間平均要求再ディスパッチ要求再試行要求 書き換え

## トラブルシューティング

詳細については、ページを参照して["サポート"](#)ください。

## 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. コレクション用エージェント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用のエージェントをインストールする場合は、[Show Instructions]をクリックして手順を展開し["エージェントのインストール"](#)ます。
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。



## Java Configuration

Gathers JVM metrics.

---

### What Operating System or Platform Are You Using? [Need Help?](#)

RHEL & CentOS

### 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 your environment before configuring. [Show Instructions](#)

### Follow Configuration Steps [Need Help?](#)

- 1 Install Jolokia on your JVMs. 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-jvm.conf file.

```
# Read JMX metrics through Jolokia
[[inputs.jolokia2_agent]]
  # USER-ACTION: Provide address(es) of JVM, port for jolokia, add one URL for each JVM in
  your cluster
  # Please specify actual machine IP address, and refrain from using a loopback address (i.e.
  127.0.0.1 or 0.0.0.0)
```
- 3 Replace <INSERT\_JVM\_ADDRESS> with the applicable JVM address. Please specify a real machine address, and refrain from using a loopback address.
- 4 Replace <INSERT\_JOLOKIA\_PORT> with the applicable JVM jolokia port.
- 5 Modify 'Namespace' if needed for server disambiguation (to avoid name clashes).
- 6 Restart the Telegraf service.

```
systemctl restart telegraf
```

## セットアップ

詳細については、を["JVMのマニュアル"](#)参照してください。

## 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 」：
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書ベンダーランタイム仕様バージョンアップタイム VM 名ランタイム 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	<p>クラス読み込み済みのクラス読み込み済みの合計クラス読み込み済みメモリヒープコミット済みメモリヒープ使用済み最大メモリヒープ使用済みメモリ非ヒープ初期化メモリ非ヒープ最大メモリ非ヒープ使用メモリオブジェクト保留中のファイナライズ OS プロセッサ利用可能な OS コミット済み仮想メモリサイズ OS フリー 物理メモリサイズ 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 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 負荷 OS システム負荷平均 OS 合計物理メモリサイズ OS 合計スワップスペースサイズ スレッドデーモンカウントスレッドピーク数 スレッド数スレッド総数開始数ガベージコレクタコピー収集数ガベージコレクタコピー収集時間ガベージコレクタマークスイープコレクション数ガベージコレクタマークスイープコレクション時間ガベージコレクタ G1 旧世代コレクション時間コレクタガベージ G1 ヤング生成コレクション数ガベージコレクタ G1 ヤング生成コレクション時間ガベージコレクタ同時マークスイープコレクション数ガベージコレクタ同時マークスイープコレクション時間ガベージコレクタパラレルコレクション数ガベージコレクタパラレルスカベンジマークスイープコレクション数ガベージコレクタパラレルスカベンジコレクション数ガベージコレクタパラレルスカベンジ 収集時間</p>

## トラブルシューティング

詳細については、ページを参照して["サポート"](#)ください。

# 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. コレクション用エージェント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用のエージェントをインストールする場合は、[*Show Instructions*]をクリックして手順を展開し["エージェントのインストール"](#)ます。
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。



## Kafka Configuration

Gathers Kafka 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 Kafka brokers. 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-kafka.conf file.

```
# Read JMX metrics through Jolokia
[[inputs.jolokia2_agent]]
  ## USER-ACTION: Provide address(es) of kafka broker(s), port for jolokia, add one URL for
  ## each broker in your cluster
  ## Please specify actual machine IP address, and refrain from using a loopback address (i.e.
  ## 127.0.0.1)
```

- 3 Replace <INSERT\_KAFKA\_BROKER\_ADDRESS> with the applicable Kafka broker address. Please specify a real machine address, and refrain from using a loopback address.
- 4 Replace <INSERT\_JOLOKIA\_PORT> with the applicable Kafka broker jolokia port.
- 5 Modify 'Namespace' if needed for server disambiguation (to avoid name clashes).
- 6 Modify 'Cluster' if needed for Kafka cluster designation.
- 7 Restart the Telegraf service.

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

## セットアップ

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

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が必要です"[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.password -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：
Kafka Broker	クラスタネームスペースのブローカー	ノード名ノード IP

## トラブルシューティング

詳細については、ページを参照して["サポート"](#)ください。

# 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. コレクション用エージェント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用のエージェントをインストールする場合は、[*Show Instructions*]をクリックして手順を展開し["エージェントのインストール"](#)ます。
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。



## Kibana Configuration

Gathers Kibana metrics.

### What Operating System or Platform Are You Using?

[Need Help?](#)

Ubuntu & Debian

### 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 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-kibana.conf` file.

```
[[inputs.kibana]]
  ## specify a list of one or more Kibana servers
  ## USER-ACTION: Provide address of kibana server(s), port(s) for kibana server
  ## Please specify actual machine IP address, and refrain from using a loopback address (i.e.
  localhost or 127.0.0.1).
```

- 2 Replace `<INSERT_KIBANA_ADDRESS>` with the applicable Kibana server address. Please specify a real machine address, and refrain from using a loopback address.
- 3 Replace `<INSERT_KIBANA_PORT>` with the applicable Kibana server port.
- 4 Replace `'username'` and `'password'` with the applicable Kibana server authentication credentials as needed, and uncomment the lines.
- 5 Modify `'Namespace'` if needed for server disambiguation (to avoid name clashes).
- 6 Restart the Telegraf service.

```
systemctl restart telegraf
```

## セットアップ

詳細については、を参照して"[Kibana のドキュメンテーション](#)"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」：
キバナ	ネームスペースアドレス	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使用要求 / 秒応答時間 平均応答時間最大アップ タイム

## トラブルシューティング

詳細については、ページを参照して["サポート"](#)ください。

# 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

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

## 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
- 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。以前のKubernetes Operatorの場合は[アップグレード](#)、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、「[について](#)」を参照してください[プロキシを設定します](#)。

カスタムリポジトリがある場合は、を参照してください [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。

NetApp / Observability / Collectors

Cluster Name ↑	Status	Operator Version	Network Performance and Map	Change Analysis	
au-pod	Outdated	1.1540.0	1.347.0	1.162.0	⋮
jks-troublemaker	Latest	1.1579.0	N/A	1.201.0	⋮
oom-test	Outdated	1.1555.0	N/A	1.161.0	⋮ Modify Deployment

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。

 **kubernetes**  
Kubernetes

## Modify Deployment

### Cluster Information

Kubernetes Cluster  
ci-demo-01

Network Performance and Map  
Enabled - Online

Event Logs  
Enabled - Online

Change Analysis  
Enabled - Online

### Deployment Options

[Need Help?](#)

- Network Performance and Map
- Event Logs
- Change Analysis

Cancel

Complete Modification

## 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（名前スペースがdefault\_netapp-monitoring\_でない場合は、適切な名前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：

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

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（名前スペースがデフォルトのNetApp監視機能でない場合は、適切な名前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（名前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、適切な名前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, アンインストール

>>既存の演算子。

\* <<installing-the-kubernetes-monitoring-operator, インストール

>>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。
- カスタムリポジトリを使用している場合は、使用していることを確認して**最新のコンテナイメージを取得します**ください。

## 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。演算子の最新バージョンについては、[のリストを参照してください"使用可能な設定"](#)。

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、テナント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/ 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_port>
```

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>/ci-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/\\_](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

イレベルを実装する場合があります。

## 公差と接続 (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["塗料および耐性"](#)をご覧ください。

に戻ります。"[\\* NetApp Kubernetes監視オペレータのインストール\\*ページ](#)"

## 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\_file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 Monitoring Operatorイメージシグネチャの確認

オペレータ用のイメージと、展開するすべての関連イメージは、NetAppによって署名されています。インストール前にcosignツールを使用してイメージを手動で検証するか、Kubernetesアドミッションコントローラを設定できます。詳細については、[を参照してください](#)"Kubernetes のドキュメント"。

イメージシグネチャの検証に使用する公開キーは、Monitoring Operatorインストールタイルの\_オプションで使用できます。オペレータイメージをプライベートリポジトリにアップロード> Image Signature Public Key\_

画像折丁を手動で確認するには、次の手順に従います。

1. 画像プルスニペットをコピーして実行する
2. プロンプトが表示されたら、リポジトリパスワードをコピーして入力します。
3. イメージ署名公開キーを保存します（この例ではdii-image-signing.pub）。
4. コサインを使用して画像を確認します。次のcosignの使用例を参照してください。

```
$ cosign verify --key dii-image-signing.pub --insecure-ignore-sct
--insecure-ignore-tlog <repository>/<image>:<tag>
Verification for <repository>/<image>:<tag> --
The following checks were performed on each of these signatures:
- The cosign claims were validated
- The signatures were verified against the specified public key
[{"critical":{"identity":{"docker-
reference":"<repository>/<image>"},"image":{"docker-manifest-
digest":"sha256:<hash>"},"type":"cosign container image
signature"},"optional":null}]
```

## 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 Tegra エージェントをアンインストールし、最新の Tegra 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。

<p>問題</p>	<p>次の操作を実行します</p>
<p>E0901 15:21:39.962145 1  reflecto.r.go:178]k8s.io/kube-state-  metrics/internal/store/builder.go:352:  List*v1.MutatingWebhookConfiguration:サーバはリク  エストされたリソースE0901 15:21:43.168161を見つ  けることができませんでした。</p>	<p>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、次の  Leubectl version_ kbe-state-metrics バージョンを取得  します。 <i>kubectll</i> デプロイ <i>/kube-state-metrics -o</i>  <i>jsonpath='{.image}'</i> これらのメッセージが発生しない  ようにするには、 kube-state-metrics デプロイを修正  して、次の Leases 設定を具体的に無効にしてくださ  い。 <i>_hookates_web_volumeconfigurations</i>  <i>resources= 証明リクエスト ,</i>  <i>configmaps,cronjobs,demonsets,horizontalscalers,ingl</i>  <i>eers,jobs,limitrange,scapers,networkpolicies ,</i>  <i>nodes,persistentvolumes,persistentvolumesalims,pers</i>  <i>istentvolumes,podeters,</i>  <i>replicaSets,replicaSets,replicationcontrollers</i>  <i>,residetodポッド</i>  <i>,residetappeditors,appers,uns,uns,uns,uns,sets,uns,u</i>  <i>ns,uns,uns,uns,sets,uns,sets,uns,sets,uns,uns,sets,u</i>  <i>ns,uns,sets,uns,uns,uns,wodecodeclieticecodetics,set</i>  <i>s,sets,sets,sets,uns,sets,uns,uns,sets,sets,sets,un</i> 検  証する Web フック設定 ' ボリュームの添付ファイル</p>
<p>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ために、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-172-31-39-47 telegraf [1827] : time="2021-  10-11T14 : 23 : 41Z" level=error msg="failed to  create cache  directory./etc/telegraf/.cache/snowflake、err:mkdir  /etc/telegraf/.ca che: permission denied.ignored \n"  func="gosnowflake.(*defaultLogger).Errorf"  file="log.go:120" Oct 11 14:23:41 ip-172-31-39-47  telegraf [1827]無視されました。open  /etc/telegraf/.cache/snowflake/ocsp_response_cache.j  son:該当するファイルまたはディレクトリがありませ  ん\n" func="gosnowflake.(*defaultLogger).Errorf"  file="log.go:120" Oct 11 14:23:41 ip-172-31-39-47  telegrZ [1827]: 2021-T1114:114:114Telegraf 1.19.3 を  起動しています</p>	<p>これは問題と呼ばれています。"<a href="#">この GitHub の記事</a>"  詳細については、を参照してください。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。</p>
<p>Kubernetes で、Telegraf ポッドが次のエラーを報告  しています。 "Error in processing mountstats info:  failed to open mountstats file: /hostfs  /proc/1/mountstats 、 error: open /hostfs  /proc/1/mountstats : permission denied"</p>	<p>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、を参照して"<a href="#">OpenShift の手順</a>"ください。</p>

問題	次の操作を実行します
<p>Kubernetes で、Telegraf ReplicaSet ポッドから次のエラーが報告されています。[プラグインの inputs.prometheus] エラー： Could not load keypair /etc/Kubernetes /pki/ etcd/server.crt : /etc/Kubernetes /pki/ etcd/server.key : open /etc/Kubernetes /pki/ etcd/server.key : 特定のディレクトリまたは crt ファイルをロードできませんでした</p>	<p>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 を編集して RS テレグラムを編集し、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。</p>
<p>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？</p>	<p>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。<a href="#">アンインストール</a>以前の監視演算子: kubectl delete agent-monitoring-cr-n NetApp kubectl delete ns NetApp -monitoring kubectl delete crd agents.monitoring.com kubectl delete clusterrole agent-manager-role agent-proxy-role agent-metrics-reader kubectl delete clusterrolebinding agent-manager-manager-rolebinding agent-manager-manager-rolebinding NetApp NetApp <a href="#">インストール</a> モニタリングオペレータの最新バージョン。</p>
<p>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。</p>	<p>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 &lt;name-space&gt; edit agent 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 (should show Pod Security Policy removed) kubectl get all -n &lt;namespace&gt;</p>
<p>grep -i psp (should show that nothing is found)</p>	<p>「ImagePullBackoff」エラーが発生しました</p>
<p>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発生することがあります。<a href="#">詳細はこちら</a> カスタム/プライベートリポジトリの構成について</p>	<p>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。</p>

問題	次の操作を実行します
<p>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。</p> <pre> kubect1 -n netapp-monitoring get all kubect1 -n netapp-monitoring describe all kubect1 -n netapp-monitoring logs &lt;monitoring-operator-pod&gt; --all -containers=true kubect1 -n netapp-monitoring logs &lt;telegraf-pod&gt; --all -containers=true </pre>	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
<p>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。以下を試してみてください:•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てください。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</p>	PodはOperator名前空間（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
<p>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。</p>	Operator名前空間の一部のnet-observerポッドがPending状態です
<p>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。</p>	Kubernetes監視演算子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。[inputs.prometheus]プラグインエラー:\ <a href="http://kube-state-metricsへのHTTPリクエストの作成エラー">http://kube-state-metricsへのHTTPリクエストの作成エラー</a> 。 <namespace>.svc.cluster.local : 8080/metrics : get\ <a href="http://kube-state-metrics">http://kube-state-metrics</a> <namespace>.svc.cluster.local : 808080/metrics : dial tcp : lookup kube-state-metrics .<namespace>.svc.svc.cluster.local tc.local

問題	次の操作を実行します
<p>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。</p>	<p>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。</p>
<p>Kubernetesのバージョンを確認します（例：kubect1 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。</p>	<p>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。</p>
<p>_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<a href="#">"設定オプション"</a>ください。...spec:...telegraf:...  <code>-name : docker run-mode : -</code>            DaemonSet 置換： <code>-key</code>  <code>: docker_unix_sock_placeholder 値 : unix</code>  <code>:/run/docker.sock.....</code></p>	<p>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。[agent]出力への書き込み中にエラーが発生しました。http：Post "\https://&lt;tenant_url&gt;/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)</p>
<p>_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<a href="#">"設定オプション"</a>ください。</p>	<p>一部のイベントログの_involvedobject_dataが見つかりません。</p>
<p>上記の手順を実行していることを確認して<a href="#">"権限"</a>ください。</p>	<p>2つの監視オペレータポッド（netapp-ci-monitoring-operator-pod &lt;pod&gt;とmonitoring-operator-pod）が実行されているのはなぜ&lt;pod&gt;ですか？</p>
<p>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<a href="#">古いオペレータを削除します</a>。、とが必要です。<a href="#">新しいものを取り付ける</a></p>	<p>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。</p>
<p>event-exporterポッドの名前を取得します。</p> <pre style="border: 1px solid #ccc; padding: 10px; margin-top: 10px;">`kubect1 -n netapp-monitoring get pods</pre>	<p>grep event-exporter</p>

問題	次の操作を実行します
awk '{print \$1}'	<pre>sed 's/event-exporter./event-exporter/'</pre> <p>「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 <code>kubectl -n netapp-monitoring edit agent</code>、前の手順で見つけた適切なイベントエクスポートポッド名を反映するように <code>log_file</code> の値を設定します。具体的には、<code>log_file</code> は「<code>/var/log/containers/netapp-ci-event-exporter.log</code>」または「<code>/var/log/containers/event-exporter</code>」のいずれかに設定する必要があります。</p> <pre>.... <b>fluent-bit:</b> ... - name: event-exporter-ci <b>substitutions:</b> - key: LOG_FILE <b>values:</b> - /var/log/containers/netapp-ci-event-exporter.log ... ....</pre> <p>あるいは、1つはまた、<a href="#">再インストール</a>エージェントをすることができます<a href="#">アンインストール</a>。</p>
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。	CPUやメモリの制限を必要に応じて増やすには、Kubernetes Monitoring Operatorを参照して <a href="#">"設定オプション"</a> ください。
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はステータスに <a href="#">"切断"</a> あります。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。
Kubernetesクラスタの[Log Analysis]で、「Event discarded as being older than maxEventAgeSeconds」というメッセージが確認されています。	Operator_agentconfiguration_を変更し、 <code>event-exporter-maxEventAgeSeconds</code> (60秒)、 <code>event-exporter-kubeQPS</code> (100)、および <code>event-exporter-kubeBurst</code> (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、ページを参照して <a href="#">"設定オプション"</a> ください。

問題	次の操作を実行します
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。	<p>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。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せん。_unprotected_configuration_オプションの詳細については、ページを参照してください"<a href="#">設定オプション</a>"。</p>
Telegrafから次のような警告メッセージが表示されません。[inputs.diskio]「vdc」のディスク名を収集できません：/dev/vdcの読み取り中にエラーが発生しました：該当するファイルまたはディレクトリがありません_	<p>Kubernetes Monitoring Operatorの場合、これらの警告メッセージは問題なく無視してかまいません。または、AgentConfigurationでtelegrafセクションを編集し、_runDsPrivileged_をtrueに設定します。詳細については、を参照して"<a href="#">オペレータの設定オプション</a>"ください。</p>

<p>問題</p> <p>Fluent-bitポッドが次のエラーで失敗しています。[2024/10/16 14:16:23][error][src/fluent-bit/plugins/in_tail/tail_fs_inotify.c:360 errno=24]開いているファイルが多すぎます[2024/10/16 14:16:23][error] failed initialize initialization failed.</p>	<p>次の操作を実行します</p> <p>クラスタの_fsnotify_settingsを変更してみます。</p> <pre>sudo sysctl fs.inotify.max_user_instances (take note of setting)  sudo sysctl fs.inotify.max_user_instances=&lt;something larger than current setting&gt;  sudo sysctl fs.inotify.max_user_watches (take note of setting)  sudo sysctl fs.inotify.max_user_watches=&lt;something larger than current setting&gt;</pre> <p>Fluent-bitを再起動します。</p> <p>注：これらの設定をノードの再起動後も維持するには、<code>/etc/sysctl.conf</code>に次の行を追加する必要があります。</p> <pre>fs.inotify.max_user_instances=&lt;something larger than current setting&gt; fs.inotify.max_user_watches=&lt;something larger than current setting&gt;</pre>
--	---

詳細については、のページまたはを["Data Collector サポートマトリックス"](#)参照して["サポート"](#)ください。

## 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。



## セットアップ

詳細については、を参照して["Memcached Wiki"](#)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 」：
Memcached	ネームスペースサーバ	ノード IP ノード名	接続処理認証要求の受け入れ失敗した認証バイト読み取り (1 秒あたり) バイト書き込み (1 秒あたり) CAS Badval CAS Hits CAS Mises Flush Reqs (1 秒あたり) GET Reqs (1 秒あたり) Touch Reqs (1 秒あたり) Touch Reqs (1 秒あたり) Connection Wrqs (1 秒あたり) 接続構造未完了接続現在の保存済みアイテムの数減少リクエストヒット数 (1 秒あたり) 削除リクエストヒット数 (1 秒あたり) 削除リクエストミス回数 (1 秒あたり) 削除リクエスト期限削除有効な削除アイテム期限切れアイテム取得ヒット数 (1 秒あたり) 取得ミス回数 (1 秒あたり) Used Hash Bytes Hash (使用済みハッシュバイトハッシュ) が拡張されています。要求ヒット (1 秒あたり) が含まれています。サーバ要求ミス (1 秒あたり) サーバ最大バイトリッスン無効な再要求されたワーカースレッド数再要求された数再要求されたワーカースレッド数合計開かれた接続数保存された

## トラブルシューティング

詳細については、ページを参照して["サポート"](#)ください。

# MongoDB データコレクタ

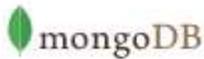
Data Infrastructure Insightsは、このデータコレクタを使用してMongoDBから指標を収集します。

## 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. コレクション用エージェント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用のエージェントをインストールする場合は、[Show Instructions]をクリックして手順を展開し"[エージェントのインストール](#)"ます。
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。



## MongoDB Configuration

Gathers MongoDB metrics.

### What Operating System or Platform Are You Using?

[Need Help?](#)

RHEL & CentOS

### 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 Open mongod.conf. Locate the line beginning with "bindIp", and append the address of the node on which the Telegraf agent resides. After saving the change, restart the MongoDB server.
- 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-mongodb.conf file.

```
[[inputs.mongodb]]
  ## An array of URLs of the form:
  ## "mongodb://" [user ":" pass "@"] host [ ":" port]
  ## For example:
  ## mongodb://user:auth_key@10.10.3.30:27017,
  ## mongodb://10.10.0.0:27017
```

- 3 Replace <INSERT\_MONGODB\_ADDRESS> with the applicable MongoDB server address. Please specify a real machine address, and refrain from using a loopback address.
- 4 Replace <INSERT\_MONGODB\_PORT> with the applicable MongoDB port.
- 5 Restart the Telegraf service.

```
systemctl restart telegraf
```

## セットアップ

詳細については、を参照して"[MongoDB のドキュメント](#)"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 」：
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

情報はページから見つけることができ["サポート"](#)ます。

## 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. コレクション用エージェント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用のエージェントをインストールする場合は、[Show Instructions]をクリックして手順を展開し["エージェントのインストール"](#)ます。
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。



## MySQL Configuration

Gathers MySQL 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 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-mysql.conf file.

```
[[inputs.mysql]]
  ## USER-ACTION: Provide comma-separated list of MySQL credentials, IP(s), and port(s)
  ## e.g. servers = ["user:passwd@tcp(127.0.0.1:3306)?tls=false"]
  ## Please specify actual machine IP address, and refrain from using a loopback address
  (i.e. localhost or 127.0.0.1).
```

- 2 Review and verify the contents of the configuration file.
- 3 Replace <INSERT\_USERNAME> and <INSERT\_PASSWORD> with the applicable MySQL credentials.
- 4 Replace <INSERT\_PROTOCOL> with the applicable MySQL connection protocol. The typical protocol is tcp.
- 5 Replace <INSERT\_MYSQL\_ADDRESS> with the applicable MySQL server address. Please specify a real machine address, and refrain from using a loopback address.
- 6 Replace <INSERT\_MYSQL\_PORT> with the applicable MySQL server port. The typical port is 3306.
- 7 Modify the 'tls' parameter in accordance to the MySQL server configuration.
- 8 Restart the Telegraf service.

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

## セットアップ

詳細については、を参照して"MySQL のドキュメント"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」：
MySQL	名前空間 MySQL サーバ	ノード IP ノード名	中断されたクライアント (1秒あたり) 中止され た接続 (1秒あたり) RX バイト (1秒あたり) TX バイト (1秒あたり) コ マンド管理者 (1秒あた り) コマンド Alter Event コマンド Alter Function コ マンド Alter Instance コマ ンド Alter 手順コマンド Alter 表コマンド Alter 表 領域コマンド Alter User コマンド Alter User コマ ンド Alter To Keycache コ マンド BEGIN コマンド Binlog コマンド手順コマ ンド Change Master コマ ンド Change Repl Filter コマンド Check コマンド チェックサムコマンドコ ミットコマンドデータベ ースの作成コマンド関数 の作成コマンドインデッ クスの作成コマンド手順 コマンドの作成トリガコ マンドの作成トリガコマ ンドの作成 UDF コマンド ユーザーコマンドの作成 ビューコマンドの作成 SQL 接続エラー作成した tmp ディスクテーブル遅 延エラーフラッシュコマ ンドの受け入れ ハンドラ Commit InnoDB Buffer Pool Bytes Data Key Blocks Not flushed Key Read Requests Key Write Requests Key Writes Max Execution Time Exceeded Max Used Connections Performance Schema Accounts Lost Prepared Stmt Count Qcache Free Blocks Queries 質問フル 結合選択範囲選択範囲チ ェックを選択します [テ ーブルロックを即時スキ ャン]を選択します

## トラブルシューティング

詳細については、ページを参照して["サポート"](#)ください。

## 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指標を収集しません。

### 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. コレクション用エージェント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用のエージェントをインストールする場合は、[*Show Instructions*]をクリックして手順を展開し["エージェントのインストール"](#)ます。
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。

## netstat

### Netstat Configuration

Gathers netstat metrics of the host where telegraf agent is installed.

---

#### 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

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-netstat.conf file.

```
# Read TCP metrics such as established, time wait and sockets counts.
[[inputs.netstat]]
# no configuration
[inputs.netstat.tags]
  CloudInsights = "true"
```
- 2

Restart the Telegraf service.

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

## セットアップ

### 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

詳細については、ページを参照して["サポート"](#)ください。

## 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. コレクション用エージェント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用のエージェントをインストールする場合は、[Show Instructions]をクリックして手順を展開し"[エージェントのインストール](#)"ます。
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。

**NGINX** Nginx Configuration  
Gathers Nginx metrics.

What Operating System or Platform Are You Using? [Need Help?](#)

Ubuntu & Debian

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 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/nginx_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 {
        stub_status on;
    }
}
```

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

nginxの指標を収集するには、nginxを有効にする必要があります"[HTTP\\_STUB\\_STATE\\_MODULE](#)"です。

詳細については、を参照して"[nginx のドキュメント](#)"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 」：
nginx	ネームスペースサーバ	ノード IP ノード名ポート	アクティブな処理済み読み取り要求の書き込み待機を受け入れます

## トラブルシューティング

詳細については、ページを参照して["サポート"](#)ください。

## 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. コレクション用エージェント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用のエージェントをインストールする場合は、[Show Instructions]をクリックして手順を展開し["エージェントのインストール"](#)ます。
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。



## PostgreSQL Configuration

Gathers PostgreSQL metrics.

### What Operating System or Platform Are You Using?

[Need Help?](#)

RHEL & CentOS

### 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 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-postgresql.conf file.

```
[[inputs.postgresql]]
# USER-ACTION: Provide credentials for access, address of PostgreSQL server, port for
PostgreSQL server, one DB for access
address = "postgres://<INSERT_USERNAME>:<INSERT_PASSWORD>@<INSERT_POSTGRESQL_ADDRESS>:
<INSERT_POSTGRESQL_PORT>/<INSERT_DB>"
```

- 2 Replace <INSERT\_USERNAME> and <INSERT\_PASSWORD> with the applicable PostgreSQL credentials.
- 3 Replace <INSERT\_POSTGRESQL\_ADDRESS> with the applicable PostgreSQL address. Please specify a real machine address, and refrain from using a loopback address.
- 4 Replace <INSERT\_POSTGRESQL\_PORT> with the applicable PostgreSQL port.
- 5 Replace <INSERT\_DB> with the applicable PostgreSQL database.
- 6 Modify 'Namespace' if needed for server disambiguation (to avoid name clashes).
- 7 Restart the Telegraf service.

```
systemctl restart telegraf
```

## セットアップ

詳細については、[を参照して"PostgreSQL のドキュメント"](#)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」：
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	バッファ割り当てバックエンドバッファ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ンチェックポイント同期時刻チェックポイント書き込み時間チェックポイント要求チェックポイント時間最大書き込みクリーン
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	ブロック読み取り時間ブロック書き込み時間ブロックヒットブロック競合の読み取りクライアント数一時ファイル数一時ファイル数行削除された行取得された行数返された行更新されたトランザクションコミット済みトランザクションロールバック

## トラブルシューティング

詳細については、ページを参照して["サポート"](#)ください。

## 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. コレクション用エージェント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用のエージェントをインストールする場合は、[Show Instructions]をクリックして手順を展開し["エージェントのインストール"](#)ます。
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。

**Puppet Agent Configuration**  
Gathers Puppet agent 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 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-puppetagent.conf file.  

```
## Reads last_run_summary.yaml file and converts to measurements
[[inputs.puppetagent]]
  ## Location of puppet last run summary file
  ## USER-ACTION: Modify the location if last_run_summary.yaml is on different path
  location = "/var/lib/puppet/state/last_run_summary.yaml"
```
- 2 Modify 'location' if last\_run\_summary.yaml is on different path
- 3 Modify 'Namespace' if needed for puppet agent disambiguation (to avoid name clashes).
- 4 Restart the Telegraf service.  

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

## セットアップ

詳細については、"[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 ージョン構成文字列バー ジョン Puppet	変更合計イベント失敗イ ベント成功イベント成功 イベントリソース合計変 更リソースリソース再起 動失敗リソース再起動リ ソース再起動スケジュー ルリソーススキップリソ ース合計時間アンカー時 間設定取得時間 cron タイ ム実行時間ファイルタイ ムファイルタイムファイ ルタイムファイルタイム ファイルタイムスケジュー ールタイムサービスタイ ムスケジュール時間設定 キー合計 時間ユーザー
--------------	--------------------	---	---

## トラブルシューティング

詳細については、ページを参照して["サポート"](#)ください。

## 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. コレクション用エージェント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用のエージェントをインストールする場合は、[Show Instructions]をクリックして手順を展開し["エージェントのインストール"](#)ます。
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。



## Redis Configuration

Gathers Redis 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 Configure Redis to accept connections from the address of the node on which the Telegraf agent resides. Open the Redis configuration file.

```
vi /etc/redis.conf
```

- 2 Locate the line that begins with 'bind 127.0.0.1', and append the address of the node on which the Telegraf agent resides

```
bind 127.0.0.1 <NODE_IP_ADDRESS>
```

- 3 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-redis.conf file.

```
# Read metrics from one or many redis servers
[[inputs.redis]]
  ## specify servers via a url matching:
  ## [protocol://][:password]@address[:port]
  ## e.g.
  ## http://192.168.1.100:6379
```

- 4 Replace <INSERT\_REDIS\_ADDRESS> with the applicable Redis address. Please specify a real machine address, and refrain from using a loopback address.

- 5 Replace <INSERT\_REDIS\_PORT> with the applicable Redis port.

- 6 Restart the Telegraf service.

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

## セットアップ

詳細については、を参照して"[Redis 文書](#)"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」：
Redis	ネームスペースサーバ		

## トラブルシューティング

詳細については、ページを参照して["サポート"](#)ください。

## 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权を有し、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权については、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。