Data Collector Reference - Dienste

Data Infrastructure Insights

NetApp
December 19, 2024

Inhalt

Data Collector Reference - Dienste

Erfassung Von Node-Daten

Data Infrastructure Insights sammelt Kennzahlen aus dem Knoten, auf dem Sie einen Agenten installieren.

Installation

1. Wählen Sie unter **Observability > Collectors** ein Betriebssystem/eine Plattform aus. Beachten Sie, dass durch die Installation eines Datensammlers für die Integration (Kubernetes, Docker, Apache usw.) auch die Erfassung von Node-Daten konfiguriert wird.
2. Befolgen Sie die Anweisungen, um den Agenten zu konfigurieren. Die Anweisungen hängen vom Betriebssystem oder der Plattform ab, die Sie zur Datenerfassung verwenden.

Objekte und Zähler

Die folgenden Objekte und ihre Zähler werden als Node-Kennzahlen erfasst:

Objekt:	Kennungen:	Attribute:	Datenpunkte:
Knoten Dateisystem	Node-UUID-Gerätetyp	Node-IP Node-Name Node OS-Modus	Freie Inodes Free Inodes Total Inodes Used Total Used Total Used Total Used
Node-Festplatte	Node-UUID-Festplatte	Node-IP Node-Name Node OS	I/O-Zeit insgesamt IOPS in Bearbeitung Lesen von Bytes (pro s) Lesezeit insgesamt Lesevorgänge (pro s) gewichtete I/O-Zeit insgesamt Schreibbyte (pro s) Schreibzeit Gesamtzahl Schreibvorgänge (pro s) Aktuelle Festplattenwarteschlange Länge Schreibzeit I/O-Zeit
Node-CPU	Node-UUID-CPU	Node-IP Node-Name Node OS	System CPU Usage User CPU Usage Idle CPU Usage Prozessor CPU Usage Interrupt CPU Usage DPC CPU Usage

Objekt:	Kennungen:	Attribute:	Datenpunkte:
Knoten	Node-UUID	Node-IP Node-Name Node OS	Kernel Boot Time Kernel Context Switches (per sec) Kernel Entropy Available Kernel Interrupts (per sec) Kernel processes Forked (per sec) Arbeitsspeicher Aktiver Speicher Verfügbar Gesamter Verfügbarer Speicher Gepufferter Speicher Im Cache Speicherlimit Speicher Speicher Bereitgestellt Als Speicher Schmutziger Speicher Freier Speicher Hoher Freier Speicher Hoher Gesamtspeicher Riesige Seitengröße Speicher Riesige Seiten Freier Speicher Riesige Seiten Gesamt Speicher Niedriger Freier Speicher Niedriger Speicher Gemappter Speicher Seitentabellen Speicher Gemeinsam Genutzter Speicher Slab Speicher Austausch Gecachten Speicher Austausch Freier Speicher Austausch Gesamt Speicher Verwendeter Gesamt-Speicher Verwendeter Speicher Vmalloc Chunk Speicher Vmalloc Gesamt-Speicher Vmalloc Verwendeter Speicher Wired Memory Writeback Total Memory Writeback Tmp Speicher Cache Fehler Speicheranforderung Null Fehler Speicherseiten Fehler Speicherseiten Fehler Speicherseiten-Speicher-Seiten-Speicher Nicht Gepageter Speicher Paged Memory Cache Core Memory Standby Cache Normaler Speicher Standby Cache Reserve Memory Transition Fehler Prozesse Blockierte Prozesse Dead Processes

Objekt:	Kennungen:	Attribute:	Datenpunkte:
Node-Netzwerk	UUID der Netzwerkschnittstelle-Node	Node Name Node-IP-Node-OS	Bytes Empfangene Bytes Gesendete Pakete Ausgehende Pakete Ausgehende Pakete Ausgehende Pakete Ausgehende Pakete Paketfehler Empfangen Pakete Empfangene Fehler Pakete Empfangene Pakete Empfangene Pakete Empfangen Pakete

Einrichtung

Informationen zur Einrichtung und Fehlerbehebung finden Sie auf der "Konfigurieren eines Agenten" Seite.

ActiveMQ Data Collector

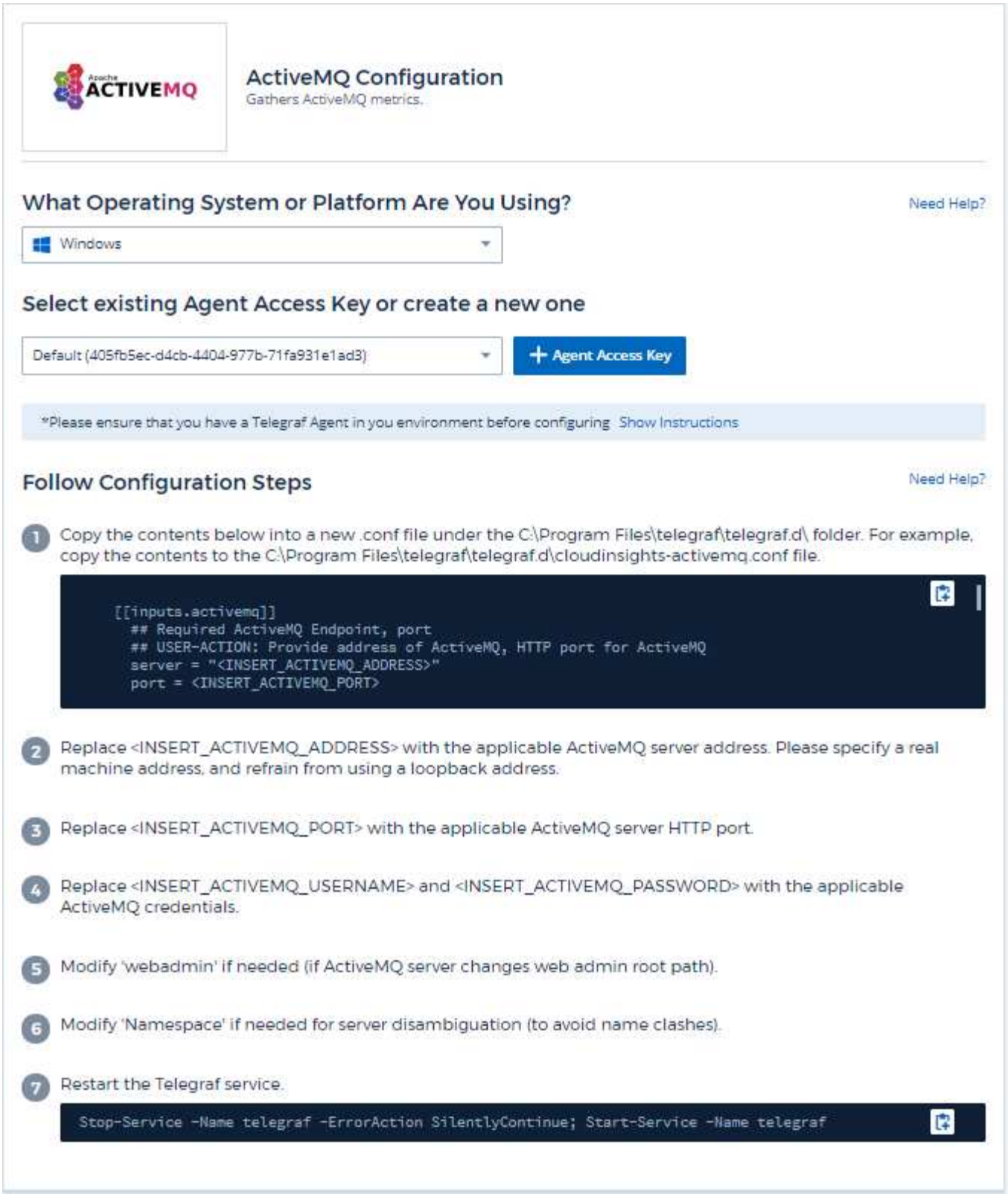
Data Infrastructure Insights verwendet diesen Datensammler, um Kennzahlen aus ActiveMQ zu sammeln.

Installation

1. Klicken Sie unter **Observability > Collectors** auf **+Data Collector**. Wählen Sie ActiveMQ.

   Wählen Sie das Betriebssystem oder die Plattform aus, auf dem der Telegraf-Agent installiert ist.

2. Wenn Sie noch keinen Agenten zur Sammlung installiert haben oder einen Agenten für ein anderes Betriebssystem oder eine andere Plattform installieren möchten, klicken Sie auf *Anweisungen anzeigen*, um die Anweisungen zu erweitern"Agenten-Installation".
3. Wählen Sie den Agent-Zugriffsschlüssel für diesen Datensammler aus. Sie können einen neuen Agent-Zugriffsschlüssel hinzufügen, indem Sie auf die Schaltfläche **+ Agent Access Key** klicken. Best Practice: Verwenden Sie einen anderen Agent-Zugriffsschlüssel nur, wenn Sie Datensammler gruppieren möchten, zum Beispiel nach Betriebssystem/Plattform.
4. Befolgen Sie die Konfigurationsschritte, um den Datensammler zu konfigurieren. Die Anweisungen hängen vom Betriebssystem oder der Plattform ab, die Sie zur Datenerfassung verwenden.



Einrichtung

Informationen finden Sie im "ActiveMQ-Dokumentation"

Objekte und Zähler

Folgende Objekte und ihre Zähler werden gesammelt:

Objekt:	Kennungen:	Attribute:	Datenpunkte:
ActiveMQ-Warteschlange	Namespace Queue Port Server	Node Name Node-IP-Node-UUID	Anzahl Der Warteschlange Anzahl Der Kunden Anzahl Der Ausgleiche Anzahl Warteschlange Größe
ActiveMQ-Abonnenten	Namespace für Client-ID-Verbindungs-ID-Port-Server	Ist Active Destination Node Name Node IP Node UUID Node OS Selector Subscription	Anzahl Der Entsandten Absendete Warteschlange Anzahl Der Abgesandten Warteschlange Größe Anzahl Der Warteschlange Anzahl Der Ausstehenden Warteschlange Größe
ActiveMQ-Thema	Thema Port Server Namespace	Node Name Node-IP-Node-UUID-Node-OS	Anzahl Der Ausgleichen Anzahl Der Verbraucher Größe Der Anzahl Der Warteschlangen

Fehlerbehebung

Weitere Informationen finden Sie auf der "Support" Seite.

Apache Data Collector

Dieser Datensammler ermöglicht die Erfassung von Daten von Apache-Servern auf Ihrem Mandanten.

Voraussetzungen

- Sie müssen Ihren Apache HTTP Server einrichten und ordnungsgemäß ausführen lassen
- Sie müssen über sudo- oder Administratorberechtigungen auf Ihrem Agent-Host/VM verfügen
- In der Regel ist das Apache *mod_Status*-Modul so konfiguriert, dass eine Seite am Speicherort '/Server-Status?Auto' des Apache-Servers angezeigt wird. Die Option *ExtendedStatus* muss aktiviert sein, um alle verfügbaren Felder zu erfassen. Informationen zur Konfiguration Ihres Servers finden Sie in der Dokumentation zum Apache-Modul: https://httpd.apache.org/docs/2.4/mod/mod_status.html#enable

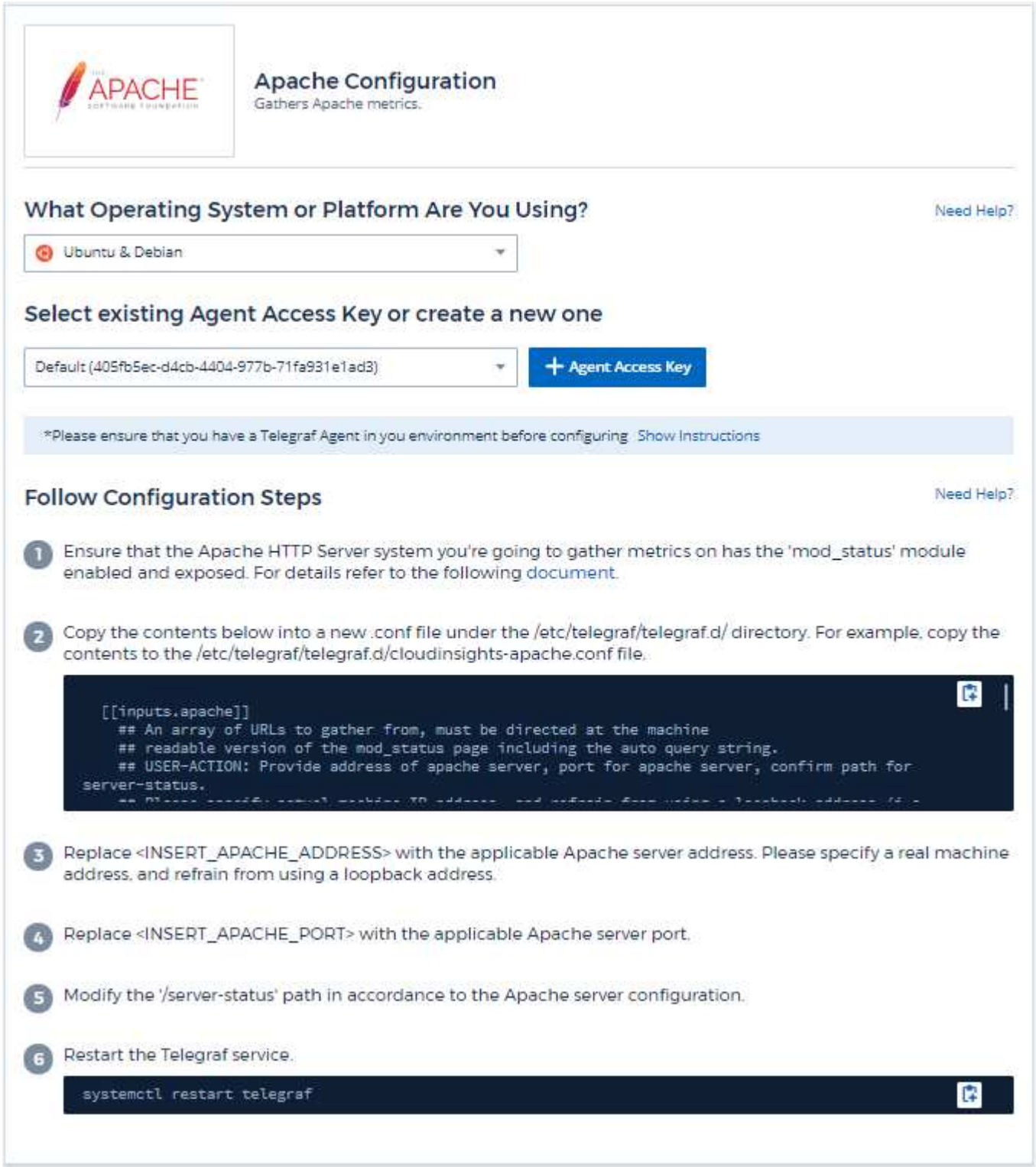
Installation

1. Klicken Sie unter **Observability > Collectors** auf **+Data Collector**. Wählen Sie Apache.

   Wählen Sie das Betriebssystem oder die Plattform aus, auf dem der Telegraf-Agent installiert ist.

2. Wenn Sie noch keinen Agenten zur Sammlung installiert haben oder einen Agenten für ein anderes Betriebssystem oder eine andere Plattform installieren möchten, klicken Sie auf *Anweisungen anzeigen*, um die Anweisungen zu erweitern"Agenten-Installation".

3. Wählen Sie den Agent-Zugriffsschlüssel für diesen Datensammler aus. Sie können einen neuen Agent-Zugriffsschlüssel hinzufügen, indem Sie auf die Schaltfläche **+ Agent Access Key** klicken. Best Practice: Verwenden Sie einen anderen Agent-Zugriffsschlüssel nur, wenn Sie Datensammler gruppieren möchten, zum Beispiel nach Betriebssystem/Plattform.
4. Befolgen Sie die Konfigurationsschritte, um den Datensammler zu konfigurieren. Die Anweisungen hängen vom Betriebssystem oder der Plattform ab, die Sie zur Datenerfassung verwenden.



Einrichtung

Das Telegraf-Plugin für Apache's HTTP Server setzt auf das 'mod_Status'-Modul, um aktiviert zu werden. Wenn diese Option aktiviert ist, wird Apache HTTP Server einen HTML-Endpunkt anzeigen, der in Ihrem Browser angezeigt oder für die Extraktion des Status aller Apache HTTP Server-Konfigurationen gepatzt werden kann.

Kompatibilität:

Die Konfiguration wurde gegen Apache HTTP Server Version 2.4.38 entwickelt.

Aktivieren von mod_Status:

Das Aktivieren und Bereitstellen der 'mod_Status'-Module umfasst zwei Schritte:

- Modul wird aktivieren
- Legen Sie Statistiken aus dem Modul fest

Modul aktivieren:

Das Laden von Modulen wird durch die Konfigurationsdatei unter '/usr/local/apache/conf/httpd.conf' gesteuert. Bearbeiten Sie die config-Datei und heben Sie die folgenden Zeilen aus:

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

Statistiken aus dem Modul offenlegen:

Die Offenlegung von 'mod_Status' wird durch die Konfigurationsdatei unter '/usr/local/apache2/conf/extra/httpd-info.conf' gesteuert. Stellen Sie sicher, dass Sie in dieser Konfigurationsdatei Folgendes haben (mindestens sind weitere Richtlinien vorhanden):

```
# 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
```

Ausführliche Anweisungen zum Modul 'mod_Status' finden Sie im "Apache-Dokumentation"

Objekte und Zähler

Folgende Objekte und ihre Zähler werden gesammelt:

Objekt:	Kennungen:	Attribute:	Datenpunkte:
Apache	Namespace-Server	Node-IP-Knotenname-Port-Parent Server-Konfiguration der übergeordnete Server-Generation der MPM-Generation wird angehalten	Beschäftigte Arbeiter Bytes pro Anfrage Bytes pro Sekunde CPU Kinder System CPU Kinder Benutzer CPU Last CPU System CPU System CPU Benutzer asynchrone Verbindungen Schließen Asynchronous Connections am Leben Asynchronous Connections Writing connections Total Duration per Request Idle Workers Load Average (Last 1m) Load Average (Last 15m) Load Average (Last Average (Last 5m) Prozesse Anfragen pro Sekunde Gesamtzugriff Gesamtdauer Gesamtdauer KBytes Scoreboard schließen Scoreboard DNS Lookups Scoreboard abschließen Scoreboard-Idle Cleanup Scoreboard halten am Leben Scoreboard Logging Scoreboard öffnen Scoreboard lesen Scoreboard senden Scoreboard Starting Scoreboard warten

Fehlerbehebung

Weitere Informationen finden Sie auf der "Support" Seite.

Consul Data Collector

Data Infrastructure Insights verwendet diesen Datensammler, um Metriken von Consul zu erfassen.

Installation

1. Klicken Sie unter **Observability > Collectors** auf **+Data Collector**. Wählen Sie Konsul.

Wenn Sie keinen Agenten für die Sammlung konfiguriert haben, werden Sie auf Ihrem Mandanten zu aufgefordert"Installieren Sie einen Agenten".

Wenn Sie bereits einen Agenten konfiguriert haben, wählen Sie das entsprechende Betriebssystem oder die entsprechende Plattform aus, und klicken Sie auf **Weiter**.

2. Befolgen Sie die Anweisungen auf dem Bildschirm Consul Configuration, um den Datensammler zu konfigurieren. Die Anweisungen hängen vom Betriebssystem oder der Plattform ab, die Sie zur Datenerfassung verwenden.

Einrichtung

Informationen finden Sie im "Dokumentation für Konsul".

Objekte und Zähler für Konsul

Folgende Objekte und ihre Zähler werden gesammelt:

Objekt:	Kennungen:	Attribute:	Datenpunkte:
Konsul	Namespace-ID-Service-Node prüfen	Node-IP Node OS Node UUID Node Name Service Name Check Name Service Service ID Status	Warnung Bei Kritischem Durchgang

Fehlerbehebung

Weitere Informationen finden Sie auf der "Support" Seite.

Couchbase Data Collector

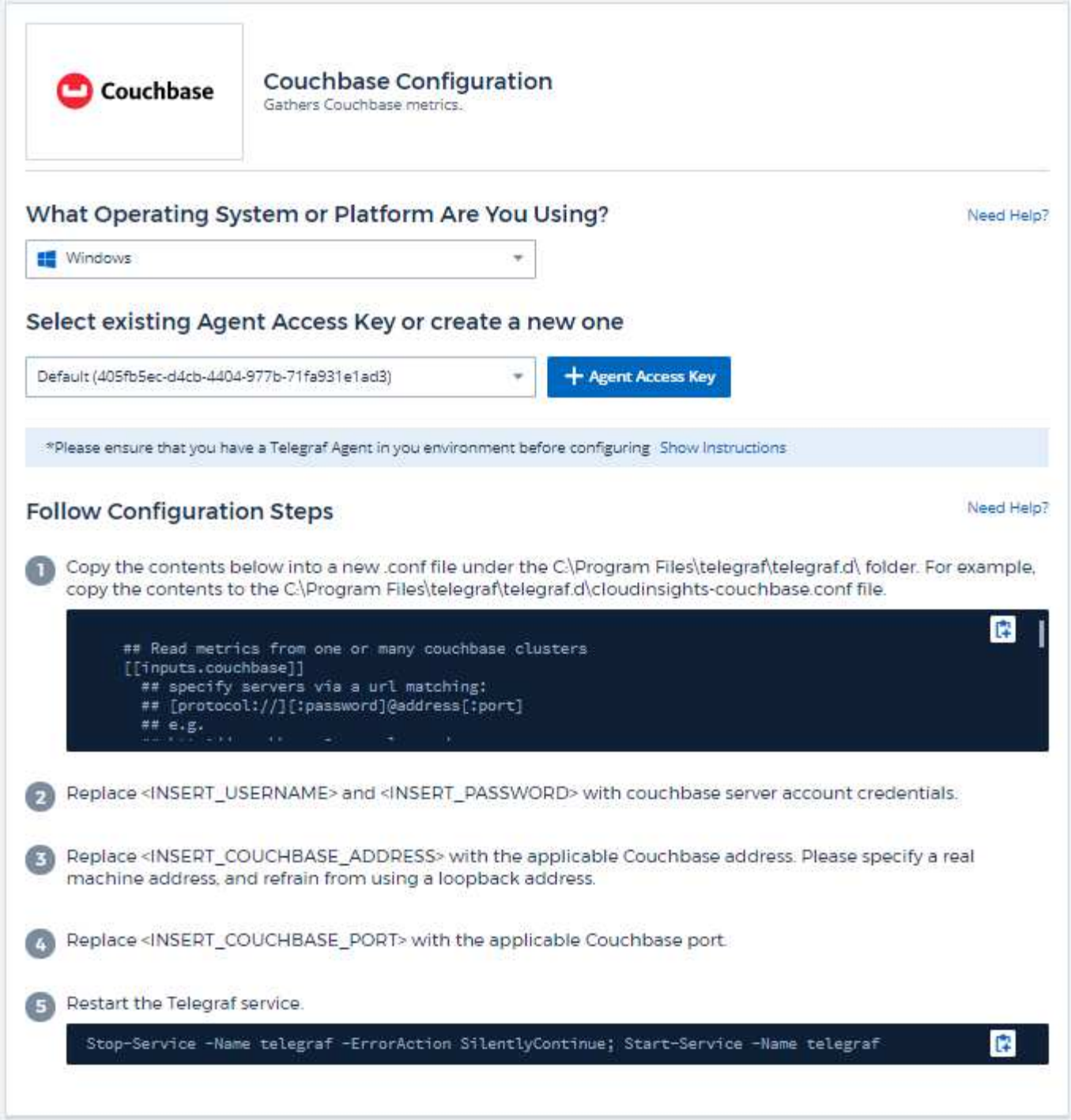
Data Infrastructure Insights nutzt diesen Datensammler zur Erfassung von Kennzahlen aus Couchbase.

Installation

1. Klicken Sie unter **Observability > Collectors** auf **+Data Collector**. Wählen Sie Couchbase.

   Wählen Sie das Betriebssystem oder die Plattform aus, auf dem der Telegraf-Agent installiert ist.

2. Wenn Sie noch keinen Agenten zur Sammlung installiert haben oder einen Agenten für ein anderes Betriebssystem oder eine andere Plattform installieren möchten, klicken Sie auf *Anweisungen anzeigen*, um die Anweisungen zu erweitern"Agenten-Installation".
3. Wählen Sie den Agent-Zugriffsschlüssel für diesen Datensammler aus. Sie können einen neuen Agent-Zugriffsschlüssel hinzufügen, indem Sie auf die Schaltfläche **+ Agent Access Key** klicken. Best Practice: Verwenden Sie einen anderen Agent-Zugriffsschlüssel nur, wenn Sie Datensammler gruppieren möchten, zum Beispiel nach Betriebssystem/Plattform.
4. Befolgen Sie die Konfigurationsschritte, um den Datensammler zu konfigurieren. Die Anweisungen hängen vom Betriebssystem oder der Plattform ab, die Sie zur Datenerfassung verwenden.



Einrichtung

Informationen finden Sie im "Couchbase Dokumentation".

Objekte und Zähler

Folgende Objekte und ihre Zähler werden gesammelt:

Objekt:	Kennungen:	Attribute:	Datenpunkte:
Couchbase Node	Namespace Cluster Couchbase Node-Hostname	Node Name Node-IP	Speicher Insgesamt
Couchbase Bucket	Namespace-Bucket-Cluster	Node Name Node-IP	Daten Verwendete Daten Abrufen Verwendete Elemente Anzahl Verwendete Elemente Speicher Verwendete Operationen Pro Sekunde Kontingent Verwendet

Fehlerbehebung

Weitere Informationen finden Sie auf der "Support" Seite.

CouchDB Data Collector

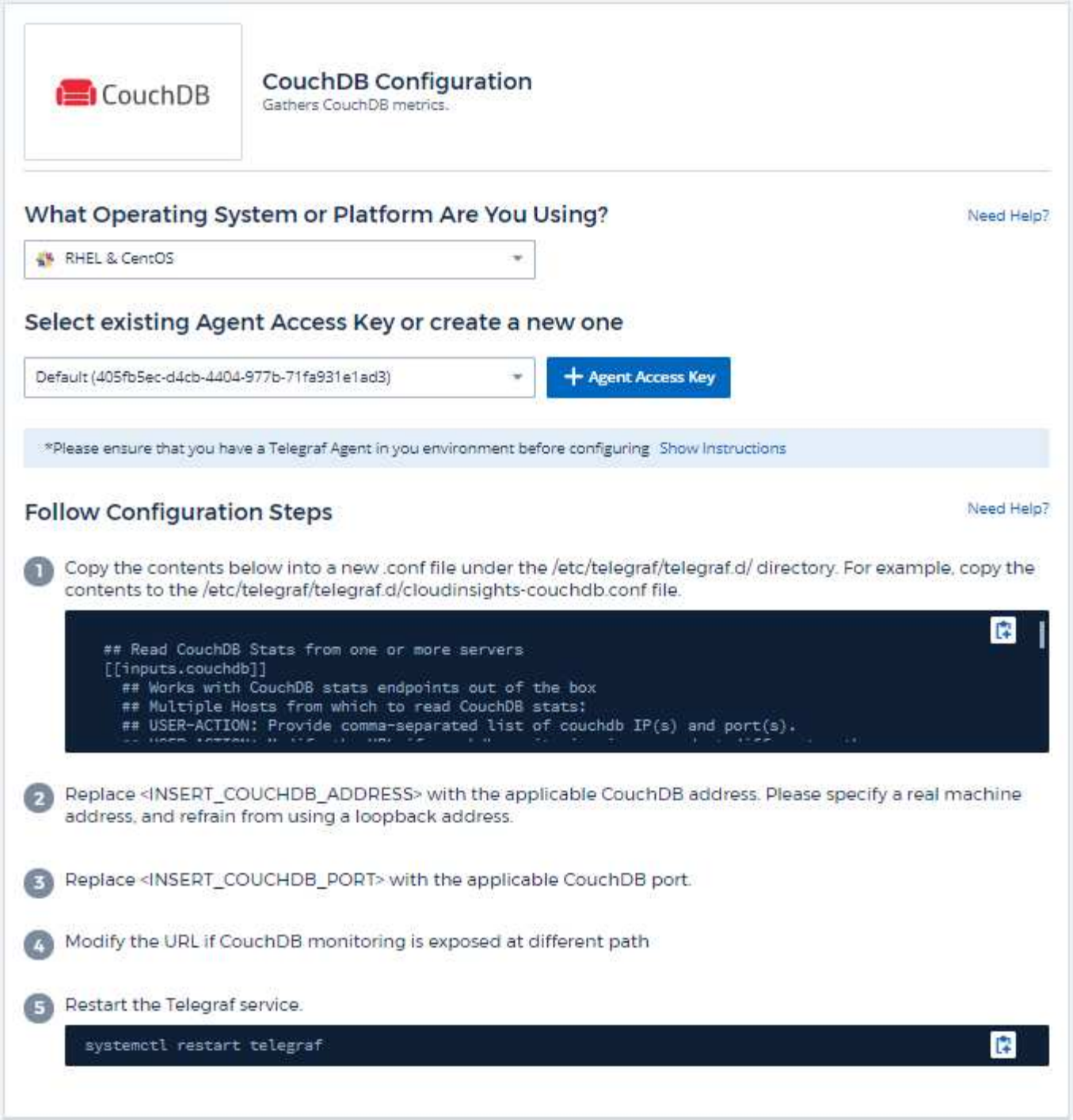
Data Infrastructure Insights nutzt diesen Datensammler, um Metriken von CouchDB zu sammeln.

Installation

1. Klicken Sie unter **Observability > Collectors** auf **+Data Collector**. Wählen Sie CouchDB.

   Wählen Sie das Betriebssystem oder die Plattform aus, auf dem der Telegraf-Agent installiert ist.

2. Wenn Sie noch keinen Agenten zur Sammlung installiert haben oder einen Agenten für ein anderes Betriebssystem oder eine andere Plattform installieren möchten, klicken Sie auf *Anweisungen anzeigen*, um die Anweisungen zu erweitern"Agenten-Installation".
3. Wählen Sie den Agent-Zugriffsschlüssel für diesen Datensammler aus. Sie können einen neuen Agent-Zugriffsschlüssel hinzufügen, indem Sie auf die Schaltfläche **+ Agent Access Key** klicken. Best Practice: Verwenden Sie einen anderen Agent-Zugriffsschlüssel nur, wenn Sie Datensammler gruppieren möchten, zum Beispiel nach Betriebssystem/Plattform.
4. Befolgen Sie die Konfigurationsschritte, um den Datensammler zu konfigurieren. Die Anweisungen hängen vom Betriebssystem oder der Plattform ab, die Sie zur Datenerfassung verwenden.



Einrichtung

Informationen finden Sie im "CouchDB-Dokumentation".

Objekte und Zähler

Folgende Objekte und ihre Zähler werden gesammelt:

Objekt:	Kennungen:	Attribute:	Datenpunkte:
CouchDB	Namespace-Server	Node Name Node-IP	Authentifizierung Cache Treffer Authentifizierung Cache Fräulein Datenbank liest Datenbank schreibt Datenbanken Open OS Files Max Anfragenzeit Min Anfragestzeit httpd Request Methoden httpd Request Methoden httpd Request löschen httpd Request Methods Get httpd Request Methods Head httpd Request Methods Post httpd Request Methods Put Status Codes 200 Status Codes 201 Statuscodes 202 Statuscodes 301 Statuscodes 304 Statuscodes 400 Statuscodes 401 Statuscodes 403 Statuscodes 404 Statuscodes 405 Statuscodes 409 Statuscodes 412 Statuscodes 500

Fehlerbehebung

Weitere Informationen finden Sie auf der "Support" Seite.

Docker Data Collector

Data Infrastructure Insights verwendet diesen Datensammler, um Kennzahlen aus Docker zu sammeln.

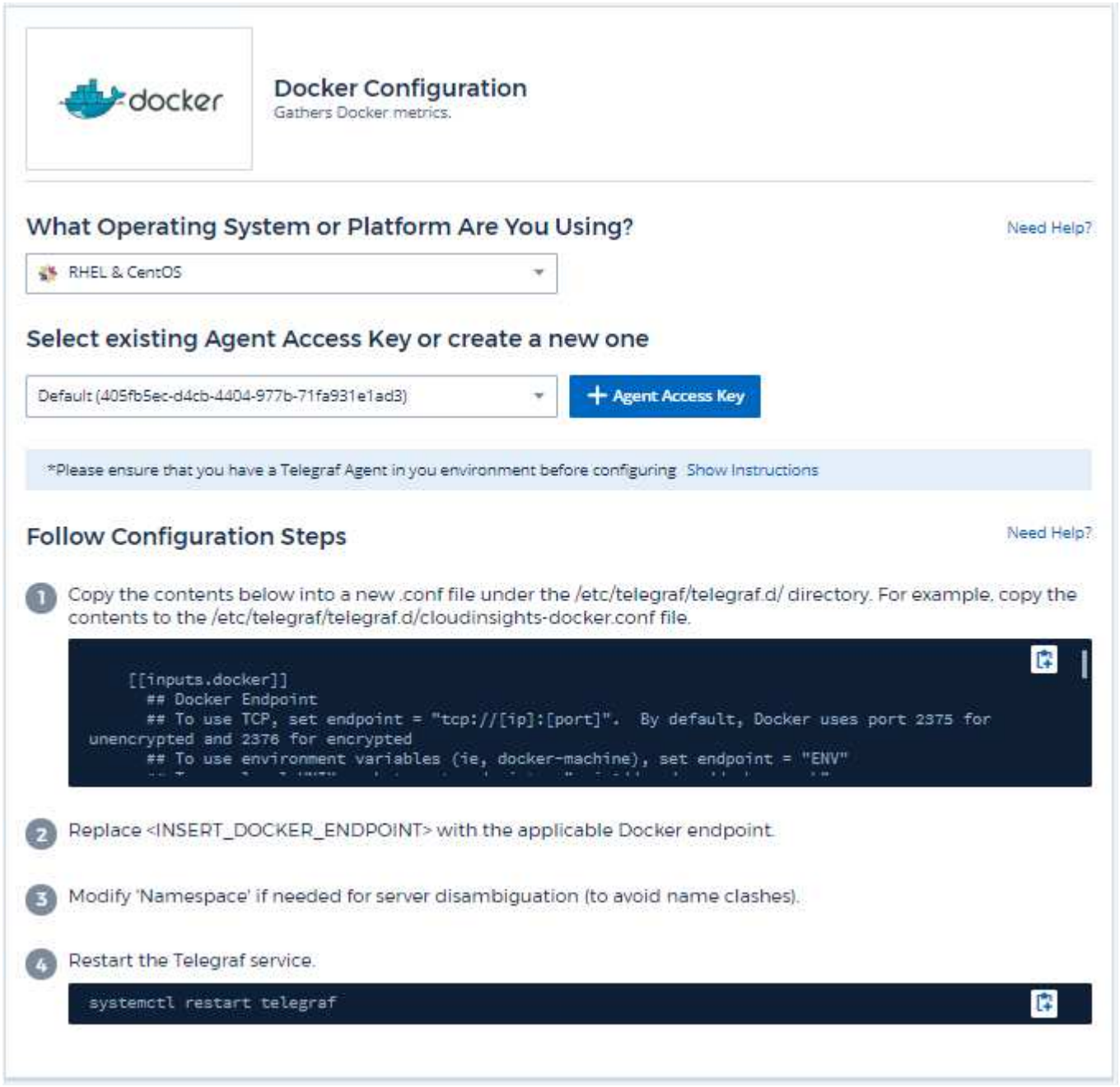
Installation

1. Klicken Sie unter **Observability > Collectors** auf **+Data Collector**. Für Docker:

   Wenn Sie keinen Agenten für die Sammlung konfiguriert haben, werden Sie auf Ihrem Mandanten zu aufgefordert"Installieren Sie einen Agenten".

   Wenn Sie bereits einen Agenten konfiguriert haben, wählen Sie das entsprechende Betriebssystem oder die entsprechende Plattform aus, und klicken Sie auf **Weiter**.

2. Befolgen Sie die Anweisungen im Bildschirm Docker-Konfiguration, um den Datensammler zu konfigurieren. Die Anweisungen hängen vom Betriebssystem oder der Plattform ab, die Sie zur Datenerfassung verwenden.



Einrichtung

Das Telegraf-Input-Plug-in für Docker erfasst Kennzahlen über einen bestimmten UNIX-Socket oder einen TCP-Endpunkt.

Kompatibilität

Die Konfiguration wurde mit Docker Version 1.12.6 entwickelt.

Einrichtung

Zugriff auf Docker über einen UNIX-Socket

Wenn der Telegraf-Agent auf bareMetal läuft, fügen Sie den telegraf Unix-Benutzer zur Docker Unix-Gruppe hinzu, indem Sie Folgendes ausführen:

```
 sudo usermod -aG docker telegraf
Wenn der Telegraf-Agent in einem Kubernetes Pod ausgeführt wird, legen Sie
den Docker Unix-Socket offen, indem Sie den Socket als Volume in den POD
einbilden und das Volume dann in /var/run/docker.sock mounten. Fügen Sie
zum Beispiel der PodSpec Folgendes hinzu:
```

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

Fügen Sie dann dem Container Folgendes hinzu:

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

Beachten Sie, dass das Installationsprogramm von Data Infrastructure Insights für die Kubernetes-Plattform diese Zuordnung automatisch übernimmt.

Zugriff auf Docker über einen TCP-Endpunkt

Docker verwendet standardmäßig Port 2375 für unverschlüsselte Zugriffe und Port 2376 für verschlüsselten Zugriff.

Objekte und Zähler

Folgende Objekte und ihre Zähler werden gesammelt:

Objekt:	Kennungen:	Attribute:	Datenpunkte:
Docker Engine	Docker Engine Für Namespace	Node Name Node-IP-Node-UUID Node OS Kubernetes Cluster Docker-Versionseinheit	Speichercontainer Container verwendete Container ausgeführt Container gestoppt CPUs Gehroutinen Bilder Listener Ereignisse verwendete Datei Deskriptoren Daten verfügbar Daten insgesamt verwendete Metadaten Verfügbare Metadaten insgesamt verwendete Pool Blocksize

Objekt:	Kennungen:	Attribute:	Datenpunkte:
Docker Container	Namespace Container-Name Docker Engine	Kubernetes-Container-Hash Kubernetes-Container-Ports Kubernetes-Container Restart Anzahl Kubernetes-Container-Ende Meldungspfad Kubernetes Container-Beendigung Meldungsrichtlinie Kubernetes Pod Kulanzzeit Container-Image Container-Status Container-Version Node-Name Kubernetes Container-Log-Pfad Kubernetes Container-Name Kubernetes Docker-Typ Kubernetes Pod Name Kubernetes Namespace Kubernetes Pod UID Kubernetes Sandbox ID Node IP Node UUID Docker Version Kubernetes IO Config Kubernetes IO-Konfiguration gesehen Kubernetes IO-Konfiguration Quelle OpenShift IO SCC Kubernetes Beschreibung Kubernetes Anzeigename OpenShift Tags Kompose Service Pod Vorlage Hash Controller Revision Hash Pod Vorlage Erstellung Lizenz Schema Build Date Schema Lizenz Schema Name Schema URL Schema VCS URL Schema Vendor Schema Version Schema Schema Schema Version Maintainer Customer Pod Kubernetes StatefulSet Pod Name Tenant WebConsole Architektur autoritäre Quelle URL Build Datum RH Build Host RH Component Distribution Scope Installation Release Run Zusammenfassung Uninstall Ref Type Vendor Version Health Status	Speicher Aktiv Anonymer Speicher Aktiv Speicher Cache Hierarchischer Grenzwert Speicher Inaktiver Anonymer Speicher Inaktiver Speicher Speicherlimit Arbeitsspeicher Gemappter Speicher Max Nutzung Speicherseitenfehler Speicherseite Hauptfehler Speicher Im Speicher Ausgepeitet Speicher Resident Set Größe Speicher Resident Set Größe Riesige Speicher Gesamt Aktiv Anonymer Speicher Gesamt Active File Memory Gesamt Cache Speicher Inaktiver Anonymer Speicher Gesamt Inaktiver Speicher Gesamt Mapped File Memory Total Page Fault Memory Total Page Major Fehler Memory Total Paged In Memory Total Paged Out Memory Total Resident Set Größe Speicher Gesamt Resident Set Größe Riesige Speicher Gesamt Nicht entfernen Speicher nicht entfernen Speichernutzung Speichernutzung Prozent Exit Code OOM tötete PID bei fehlender Streak gestartet

Objekt:	Kennungen:	Attribute:	Datenpunkte:
Docker Container Block IO	Namespace Container Name Device Docker Engine	Kubernetes-Container-Hash Kubernetes-Container-Ports Kubernetes-Container Restart Anzahl Kubernetes-Container-Ende Meldungspfad Kubernetes Container-Beendigung Meldungsrichtlinie Kubernetes Pod Kulanzzeit Container-Image Container-Status Container-Version Node-Name Kubernetes Container-Log-Pfad Kubernetes Container-Name Kubernetes Docker-Typ Kubernetes Pod Name Kubernetes Namespace Kubernetes Pod UID Kubernetes Sandbox ID Node IP Node UUID Docker Version Kubernetes Config Kubernetes Config gesehen Kubernetes Config Quelle OpenShift SCC Kubernetes Beschreibung Kubernetes Anzeigename OpenShift Tags Schema Schema Version Pod Template Hash Controller Revision Hash Pod Template Generation Kompose Service Schema Build Date Schema Lizenz Schema Name Schema Vendor Customer Pod Kubernetes StatprofSet Pod Name Tenant WebConsole Build Date License Vendor Architecture authorited Source URL RH Build Host RH Component Distribution Scope Install Maintainer Release Run Summary Uninstall VCS Ref VCS Typ Version Schema URL Schema VCS Schema Version Container ID	IO Service Bytes rekursiv Async IO Service Bytes rekursiv IO lesen Service Bytes rekursiv Sync IO Service Bytes rekursiv IO Service Bytes rekursiv Schreib IO Serviced rekursive Async E/A Serviced rekursive Read IO Serviced rekursive Sync IO Serviced rekursive Total IO Serviced rekursive Write

Objekt:	Kennungen:	Attribute:	Datenpunkte:
Docker Container Network	Namespace Container Name Network Docker Engine	Container Image Container Status Container Version Node Name Node IP Node UUID Node OS K8s Cluster Docker Version Container ID	RX-reduzierte RX-Bytes RX-Fehler RX-Pakete TX reduzierte TX-Bytes TX-Fehler TX-Pakete

Objekt:	Kennungen:	Attribute:	Datenpunkte:
Docker Container-CPU	Namespace Container Name CPU Docker Engine	Kubernetes-Container-Hash Kubernetes-Container-Ports Kubernetes-Container Restart Anzahl Kubernetes-Container-Ende Meldungspfad Kubernetes Container-Beendigung Meldungsrichtlinie Kubernetes Pod Kulanzzeit Kubernetes-Konfiguration Kubernetes-Konfiguration Kubernetes-KonfigurationSCC-Container-Image Container-Status Container-Version Node-Name Kubernetes Container-Log-Pfad Kubernetes-Container-Name Kubernetes Docker Typ Kubernetes Pod Name Kubernetes Pod Namespace Kubernetes Pod UID Kubernetes Sandbox ID Node IP Node UUID Node OS Kubernetes Cluster Docker Version Kubernetes Beschreibung Kubernetes Anzeigename OpenShift Tags Schema Version Pod Template Hash Controller Revision Hash Pod Template Generation Kompose Service Schema Build Date Schema License Schema Name Schema Hersteller-Pod Kubernetes StatprofSet Pod Name Tenant WebConsole Build Date License Vendor Architecture authorited Source URL RH Build Host RH Component Distribution Scope Install Maintainer Release Run Summary Uninstall VCS Ref VCS Typ Version Schema URL Schema VCS Schema VCS URL Schema Version Container ID	Drosselungszeiträume Drosselung Gedrosselte Perioden Drosselung Gedrosselte Zeitnutzung Im Kernel-Modus Nutzung Im Benutzermodus Auslastung Prozent Nutzung Des Systems Gesamt

Fehlerbehebung

Problem:	Versuchen Sie dies:
Ich sehe meine Docker-Kennzahlen in Data Infrastructure Insights nicht, nachdem ich die Anweisungen auf der Konfigurationsseite befolgt habe.	Prüfen Sie die Telegraf-Agentenprotokolle, um zu sehen, ob es folgenden Fehler meldet: E! Fehler im Plugin [inputs.docker]: Berechtigung verweigert beim Versuch, eine Verbindung zum Docker Daemon-Socket herzustellen.Falls dies der Fall ist, ergreifen Sie die erforderlichen Schritte, um den Telegraf-Agent-Zugriff auf den Docker Unix-Sockel wie oben angegeben zu ermöglichen.

Weitere Informationen finden Sie auf der "Support" Seite.

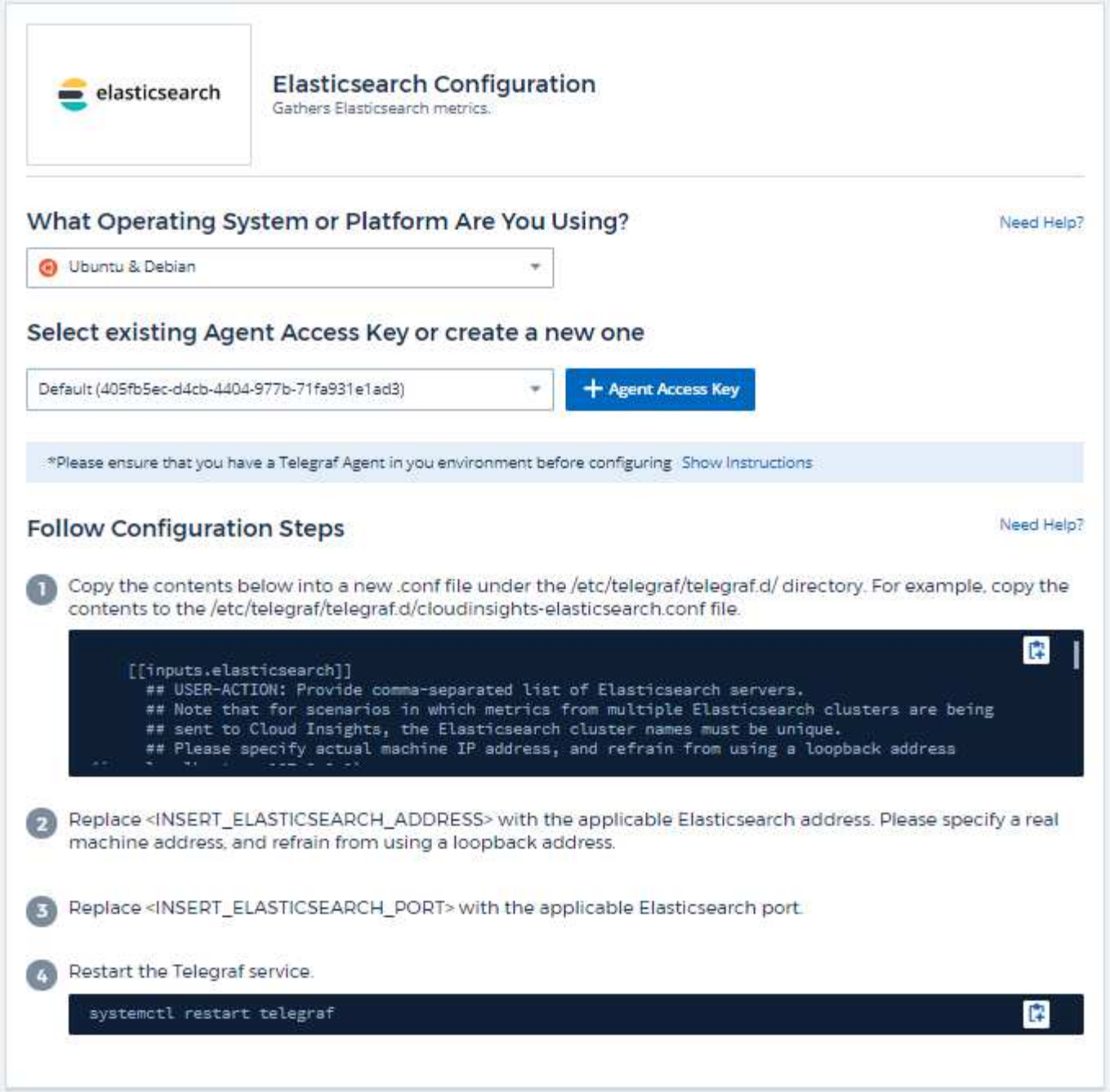
Elasticsearch Data Collector

Data Infrastructure Insights verwendet diesen Datensammler zum Erfassen von Kennzahlen aus Elasticsearch.

1. Klicken Sie unter **Observability > Collectors** auf **+Data Collector**. Wählen Sie Elasticsearch.

   Wählen Sie das Betriebssystem oder die Plattform aus, auf dem der Telegraf-Agent installiert ist.

2. Wenn Sie noch keinen Agenten zur Sammlung installiert haben oder einen Agenten für ein anderes Betriebssystem oder eine andere Plattform installieren möchten, klicken Sie auf *Anweisungen anzeigen*, um die Anweisungen zu erweitern"Agenten-Installation".
3. Wählen Sie den Agent-Zugriffsschlüssel für diesen Datensammler aus. Sie können einen neuen Agent-Zugriffsschlüssel hinzufügen, indem Sie auf die Schaltfläche **+ Agent Access Key** klicken. Best Practice: Verwenden Sie einen anderen Agent-Zugriffsschlüssel nur, wenn Sie Datensammler gruppieren möchten, zum Beispiel nach Betriebssystem/Plattform.
4. Befolgen Sie die Konfigurationsschritte, um den Datensammler zu konfigurieren. Die Anweisungen hängen vom Betriebssystem oder der Plattform ab, die Sie zur Datenerfassung verwenden.



Einrichtung

Informationen finden Sie im "Elasticsearch-Dokumentation".

Objekte und Zähler

Folgende Objekte und ihre Zähler werden gesammelt:

Objekt:	Kennungen:	Attribute:
Elasticsearch-Cluster	Namespace-Cluster	Node-IP Node-Name Cluster-Status

Objekt:	Kennungen:	Attribute:
Elasticsearch-Node	Namespace Cluster es Node ID es Node IP es Node	Zone-ID

Fehlerbehebung

Weitere Informationen finden Sie auf der "Support" Seite.

Flik Data Collector

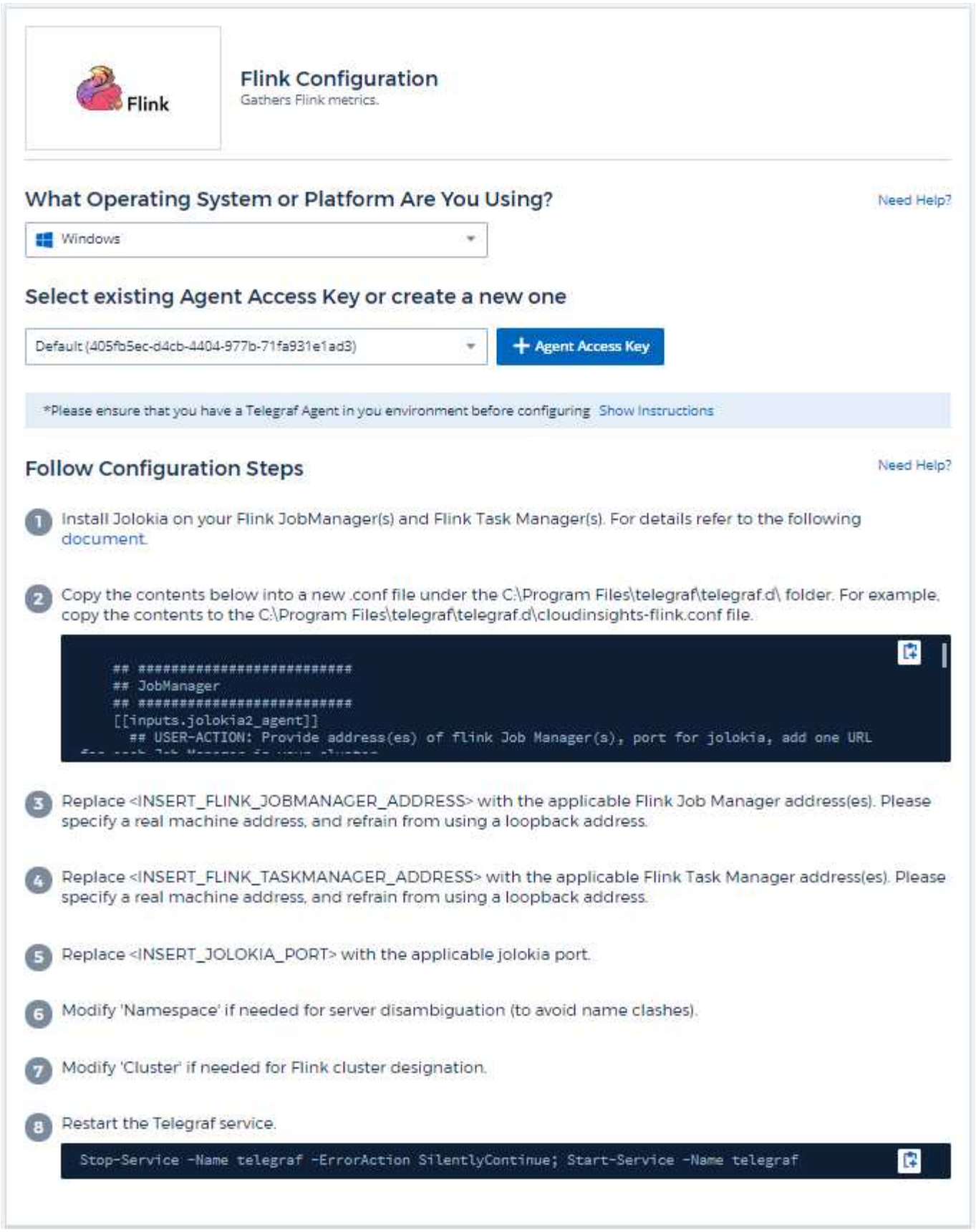
Data Infrastructure Insights verwendet diesen Datensammler, um Kennzahlen von Flink zu erfassen.

Installation

1. Klicken Sie unter **Observability > Collectors** auf **+Data Collector**. Wählen Sie „Flink“.

   Wählen Sie das Betriebssystem oder die Plattform aus, auf dem der Telegraf-Agent installiert ist.

2. Wenn Sie noch keinen Agenten zur Sammlung installiert haben oder einen Agenten für ein anderes Betriebssystem oder eine andere Plattform installieren möchten, klicken Sie auf *Anweisungen anzeigen*, um die Anweisungen zu erweitern"Agenten-Installation".
3. Wählen Sie den Agent-Zugriffsschlüssel für diesen Datensammler aus. Sie können einen neuen Agent-Zugriffsschlüssel hinzufügen, indem Sie auf die Schaltfläche **+ Agent Access Key** klicken. Best Practice: Verwenden Sie einen anderen Agent-Zugriffsschlüssel nur, wenn Sie Datensammler gruppieren möchten, zum Beispiel nach Betriebssystem/Plattform.
4. Befolgen Sie die Konfigurationsschritte, um den Datensammler zu konfigurieren. Die Anweisungen hängen vom Betriebssystem oder der Plattform ab, die Sie zur Datenerfassung verwenden.



Einrichtung

Eine vollständige Flink-Implementierung umfasst die folgenden Komponenten:

JobManager: Das Primärsystem Flik. Koordiniert eine Reihe von TaskManagers. In einer Konfiguration mit hoher Verfügbarkeit verfügt das System über mehr als einen JobManager. Taskmanager: Hier werden Flik-Operatoren ausgeführt. Das Flink Plugin basiert auf dem telegraf Jolokia Plugin. Als Voraussetzung für die Erfassung von Informationen aus allen Flik-Komponenten muss JMX auf allen Komponenten konfiguriert und über Jolokia freigelegt werden.

Kompatibilität

Die Konfiguration wurde gegen die Version 1.7 von Flink entwickelt.

Einrichtung

Jolokia Agent Jar

Für alle einzelnen Komponenten muss eine Version der Jolokia Agent JAR-Datei heruntergeladen werden. Die Version wurde getestet gegen war "Jolokia Agent 1.6.0".

Anweisungen unten gehen davon aus, dass die heruntergeladene JAR-Datei (jolokia-jvm-1.6.0-Agent.jar) unter dem Speicherort '/opt/flink/lib/' platziert wird.

JobManager

Um JobManager so zu konfigurieren, dass die Jolokia API freigegeben wird, können Sie die folgende Umgebungsvariable auf Ihren Knoten einrichten und dann den JobManager neu starten:

```
 export FLINK_ENV_JAVA_OPTS="-javaagent:/opt/flink/lib/jolokia-jvm-1.6.0-
agent.jar=port=8778,host=0.0.0.0"
Sie können einen anderen Port für Jolokia (8778) wählen. Wenn Sie eine
interne IP haben, um Jolokia zu sperren, können Sie die „Catch all“
0.0.0.0 durch Ihre eigene IP ersetzen. Beachten Sie, dass diese IP über
das telegraf-Plugin zugänglich sein muss.
```

Taskmanager

So konfigurieren Sie TaskManager(s), um die Jolokia-API zu öffnen, können Sie die folgende Umgebungsvariable auf Ihren Knoten einrichten und dann den TaskManager neu starten:

```
 export FLINK_ENV_JAVA_OPTS="-javaagent:/opt/flink/lib/jolokia-jvm-1.6.0-
agent.jar=port=8778,host=0.0.0.0"
Sie können einen anderen Port für Jolokia (8778) wählen. Wenn Sie eine
interne IP haben, um Jolokia zu sperren, können Sie die „Catch all“
0.0.0.0 durch Ihre eigene IP ersetzen. Beachten Sie, dass diese IP über
das telegraf-Plugin zugänglich sein muss.
```

Objekte und Zähler

Folgende Objekte und ihre Zähler werden gesammelt:

Objekt:	Kennungen:	Attribute:	Datenpunkte:
Flik Task Manager	Cluster Namespace-Server	Node Name Task-Manager-ID-Knoten-IP	Netzwerk verfügbar Speichersegmente Netzwerk Speichersegmente Speichersegmente Garbage Collection PS MarkSweep Count Garbage Collection PS MarkSweep Time Garbage Collection PS Scavenge Count Garbage Collection PS Scavenge Time Heap Memory Comstived Heap Memory Init Heap Memory Max Heap Memory Used Thread Count Daemon Thread Count Thread Count Spitzenanzahl Thread Count Thread Count Insgesamt Gestartet
Druckauftrag Einflken	Job-ID des Cluster-Namespace-Servers	Node Name Job Name Node-IP Letzte Checkpoint External Path-Neustartzeit	Ausfall Vollneustarts Last Checkpoint Alignment Buffered Last Checkpoint Duration Last Checkpoint Size Anzahl der abgeschlossenen Checkpoints Anzahl der fehlgeschlagenen Checkpoints Anzahl der laufenden Checkpoints Anzahl der Kontrollpunkte Betriebszeit

Objekt:	Kennungen:	Attribute:	Datenpunkte:
Flik Job Manager	Cluster Namespace-Server	Node Name Node-IP	Garbage Collection PS MarkSweep Count Garbage Collection PS MarkSweep Time Garbage Collection PS Scavenge Count Garbage Collection PS Scravenge Time Heap Memory Comstived Heap Memory Init Heap Memory Max Heap Memory Used Number Registrierte Task-Manager Anzahl laufende Jobs Taskleisten verfügbare Task-Steckplätze Gesamt-Thread-Anzahl Daemon-Thread-Anzahl Maximale Anzahl Der Threads Anzahl Der Threads Insgesamt Begonnen

Objekt:	Kennungen:	Attribute:	Datenpunkte:
Flik-Aufgabe	Cluster Namespace Job-ID Task-ID	Server Node Name Job Name Sub Task-Index Task-Versuch-ID Task-Versuch Nummer Task-Name Task-Manager-ID Knoten-IP Aktuelle Eingabe-Wasserzeichen	Puffer in Pool Nutzung Buffers in Warteschlange Länge Buffer Out Pool Nutzung Buffer Out Queue Länge Anzahl Puffer in Lokale Anzahl Buffers in Local per Second Anzahl Puffer in Local per second Rate Anzahl Puffer in Remote Number Buffers in Remote per second Anzahl Puffer in Remote per second Anzahl der Puffer in Remote per Anzahl Der Auspuffer Anzahl Der Auspuffer Pro Sekunde Anzahl Auspuffer Pro Sekunde Anzahl Bytes Pro Sekunde Anzahl Bytes In Lokale Anzahl Bytes Pro Sekunde Anzahl Bytes In Lokal Pro Sekunde Anzahl Bytes In Lokal Pro Sekunde Anzahl Bytes In Remote Number Bytes In Remote Per Second Anzahl Bytes In Remote Pro Sekunde Rate Anzahl Bytes Out Anzahl Bytes Out Pro Sekunde Anzahl Bytes Out Pro Sekunde Anzahl Datensätze In Number Datensätze In Per Second Anzahl Datensätze Pro Sekunde Anzahl Datensätze Pro Sekunde Anzahl Datensätze Pro Sekunde Anzahl Datensätze Aus Anzahl Datensätze Pro Sekunde Anzahl Datensätze Aus Pro Sekunde

Objekt:	Kennungen:	Attribute:	Datenpunkte:
Flik Task Operator	Cluster Namespace Job-ID Operator-ID Task-ID	Server Node Name Job Name Operator Name Sub Task-Index Task-Versuch-ID Task-Versuch-Nummer Task-Name Task-Manager-ID-Knoten-IP	Aktuelle Eingabe Watermark Current Output Watermark Number Records In Number Records In Per Second Count Anzahl Datensätze In Pro Sekunde Anzahl Datensätze Pro Sekunde Anzahl Datensätze Aus Anzahl Datensätze Pro Sekunde Anzahl Anzahl Datensätze Aus Pro Sekunde Anzahl Verspätete Datensätze Verworfen Zugewiesene Partitionen Bytes Verbrauchte Rate Commit Latenz Durchschn. Commit-Latenz Max. Commit Rate Commits faciert fehlgeschlagene Verbindungen Close Rate Verbindungsanzahl Verbindungserzeugung Rate Anzahl Abholen Latenz durchschn. Abholen Max. Abholen Rate Abholen Größe Max. Abholen Drosselzeit durchschn. Abrufdauer Max. Heartbeat Rate Incoming Byte Rate I/O-Zeit durchschn. (Ns) I/O Wartezeit I/O Wartezeit durchschn. (Ns) Verbindungsrate Verbindungszeit durchschn. Letzter Heartbeat ago Netzwerk-I/O-Rate ausgehende Byte-Datensätze verbrauchte Rate Datensätze lag max. Datensätze pro Anforderung durchschn. Anfragemgröße Durchschnittl. Anfragengröße max. Ansprechrate Wählen Rate Synchronisierungszeit durchschn. Heartbeat Antwort Zeit Max. Verbindungszeit Max. Synchronisierungszeit

Fehlerbehebung

Weitere Informationen finden Sie auf der "Support" Seite.

Hadoop Data Collector

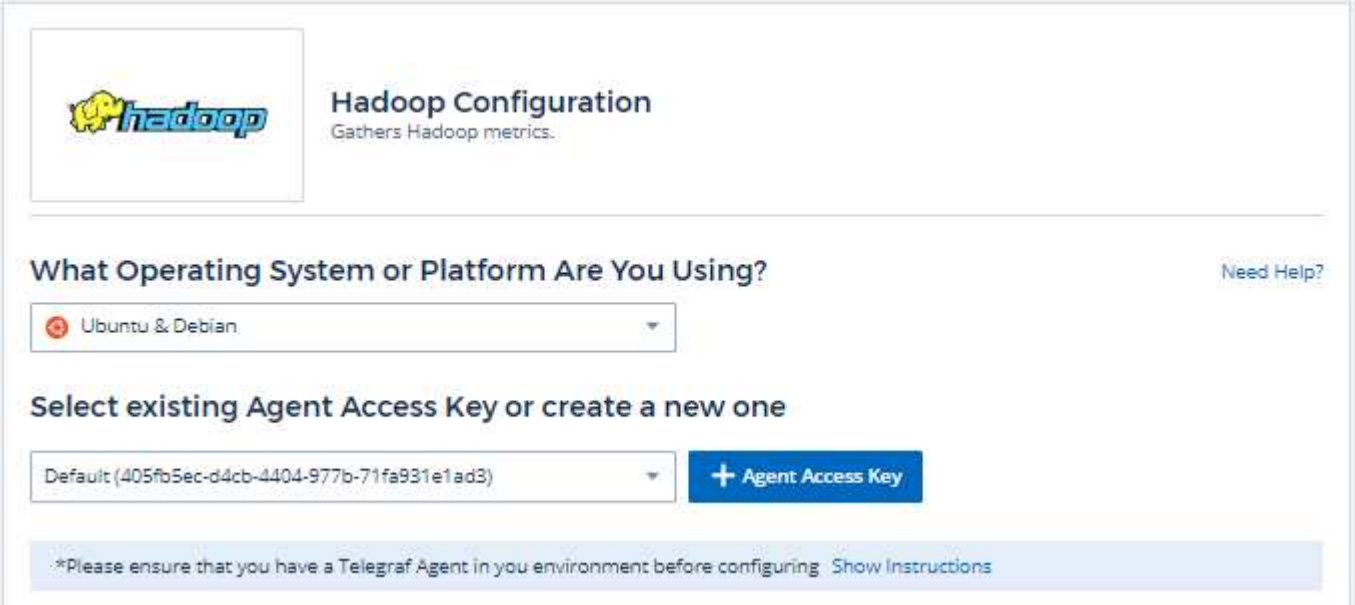
Data Infrastructure Insights verwendet diesen Datensammler, um Kennzahlen aus Hadoop zu sammeln.

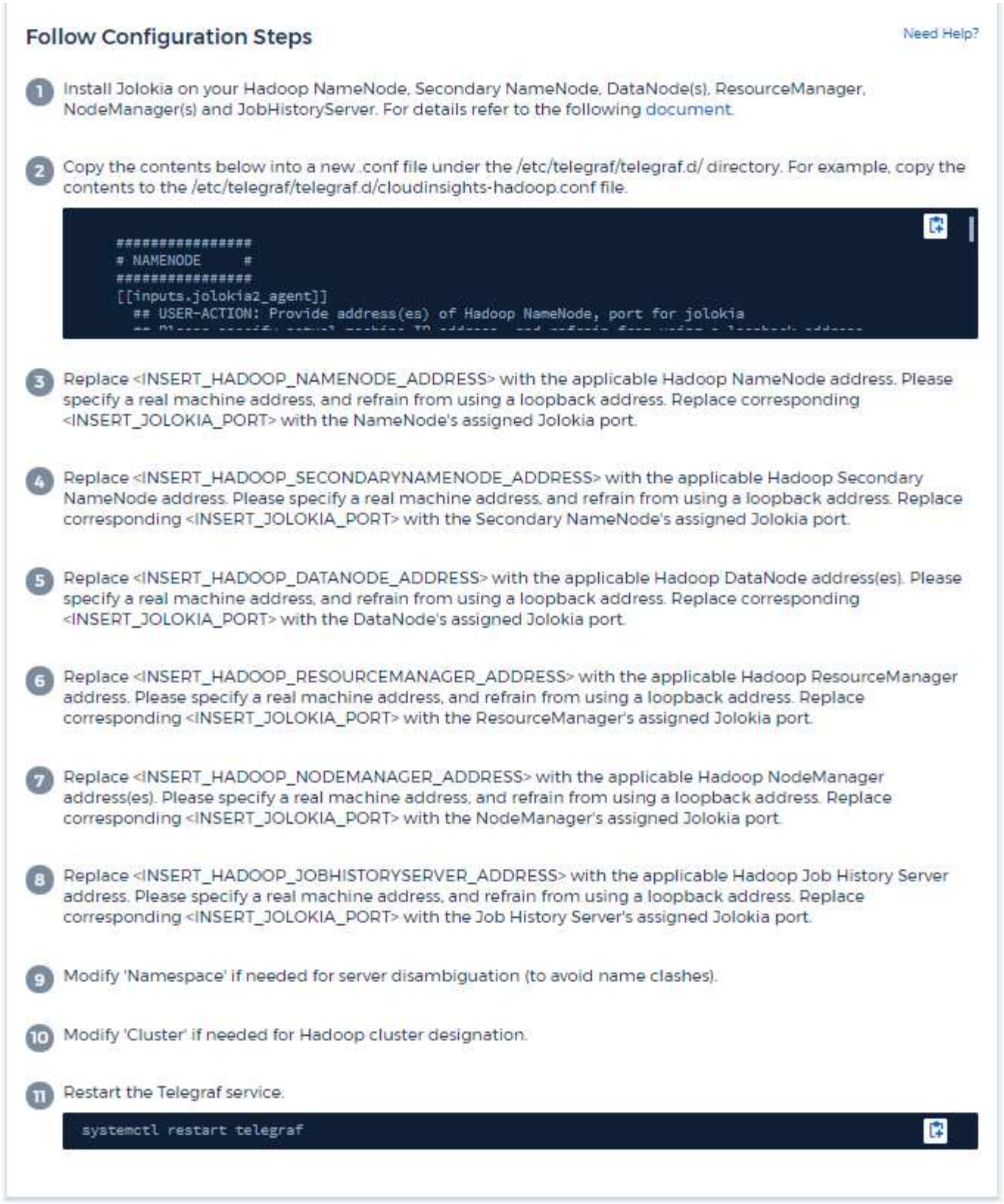
Installation

1. Klicken Sie unter **Observability > Collectors** auf **+Data Collector**. Für Hadoop.

   Wählen Sie das Betriebssystem oder die Plattform aus, auf dem der Telegraf-Agent installiert ist.

2. Wenn Sie noch keinen Agenten zur Sammlung installiert haben oder einen Agenten für ein anderes Betriebssystem oder eine andere Plattform installieren möchten, klicken Sie auf *Anweisungen anzeigen*, um die Anweisungen zu erweitern"Agenten-Installation".
3. Wählen Sie den Agent-Zugriffsschlüssel für diesen Datensammler aus. Sie können einen neuen Agent-Zugriffsschlüssel hinzufügen, indem Sie auf die Schaltfläche **+ Agent Access Key** klicken. Best Practice: Verwenden Sie einen anderen Agent-Zugriffsschlüssel nur, wenn Sie Datensammler gruppieren möchten, zum Beispiel nach Betriebssystem/Plattform.
4. Befolgen Sie die Konfigurationsschritte, um den Datensammler zu konfigurieren. Die Anweisungen hängen vom Betriebssystem oder der Plattform ab, die Sie zur Datenerfassung verwenden.





Einrichtung

Eine vollständige Hadoop Implementierung umfasst die folgenden Komponenten:

- NameNode: Das primäre System Hadoop Distributed File System (HDFS) Koordiniert eine Reihe von DataNodes.

- Sekundärer NameNode: Ein warmer Failover für den NameNode. In Hadoop erfolgt die Heraufstufung auf NameNode nicht automatisch. Secondary NameNode sammelt Informationen von NameNode, damit sie bei Bedarf heraufgestuft werden können.
- DataNode: Tatsächlicher Eigentümer von Daten.
- ResourceManager: Das primäre Computersystem (Yarn). Koordiniert eine Reihe von NodeManagers.
- NodeManager: Die Ressource für Computing. Aktueller Speicherort für das Ausführen von Anwendungen.
- JobHistorieServer: Verantwortlich für die Bearbeitung aller Anfragen im Zusammenhang mit der Jobhistorie.

Das Hadoop Plugin basiert auf dem telegraf Jolokia Plugin. Um Informationen aus allen Hadoop Komponenten zu sammeln, muss JMX auf allen Komponenten konfiguriert und zugänglich gemacht werden.

Kompatibilität

Die Konfiguration wurde mit Hadoop Version 2.9 entwickelt.

Einrichtung

Jolokia Agent Jar

Für alle einzelnen Komponenten muss eine Version der Jolokia Agent JAR-Datei heruntergeladen werden. Die Version wurde getestet gegen war "Jolokia Agent 1.6.0".

Die nachfolgende Anleitung setzt voraus, dass die heruntergeladene JAR-Datei (jolokia-jvm-1.6.0-Agent.jar) unter der Adresse '/opt/hadoop/lib/' abgelegt wird.

NameNode

Um NameNode zu konfigurieren, um die Jolokia API freizugeben, können Sie unter <HADOOP_HOME>/etc/hadoop/hadoop-env.sh Folgendes einrichten:

```
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.
```

Sekundärer NameNode

Um den sekundären NameNode zu konfigurieren, um die Jolokia API freizugeben, können Sie Folgendes in <HADOOP_HOME>/etc/hadoop/hadoop-env.sh einrichten:

```
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

Um die DataNodes so zu konfigurieren, dass sie die Jolokia API aussetzen, können Sie Folgendes in <HADOOP_HOME>/etc/hadoop/hadoop-env.sh einrichten:

```
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

Um den ResourceManager so zu konfigurieren, dass die Jolokia API zur Verfügung gestellt wird, können Sie Folgendes in <HADOOP_HOME>/etc/hadoop/hadoop-env.sh einrichten:

```
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

Um die NodeManagers so zu konfigurieren, dass sie die Jolokia API aussetzen, können Sie Folgendes in <HADOOP_HOME>/etc/hadoop/hadoop-env.sh einrichten:

```
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.
```

JobGeschichteServer

Um den JobHistorieServer so zu konfigurieren, dass die Jolokia API zur Verfügung gestellt wird, können Sie Folgendes in <HADOOP_HOME>/etc/hadoop/hadoop-env.sh einrichten:

```
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.
```

Objekte und Zähler

Folgende Objekte und ihre Zähler werden gesammelt:

Objekt:	Kennungen:	Attribute:
Sekundärer Hadoop NameNode	Cluster Namespace-Server	Node Name Node IP Compile Info Version
Hadoop NodeManager	Cluster Namespace-Server	Node Name Node-IP
Hadoop ResourceManager	Cluster Namespace-Server	Node Name Node-IP
Hadoop DataNode	Cluster Namespace-Server	Node Name Node-IP Cluster-ID-Version
Hadoop NameNode	Cluster Namespace-Server	Node Name Node IP Transaktions-ID Letzte geschriebene Zeit seit Letzte geladen Edits HA State File System Status Block Pool ID Cluster ID Compile Info unterschiedliche Version Anzahl Version
Hadoop JobGeschichteServer	Cluster Namespace-Server	Node Name Node-IP

Fehlerbehebung

Weitere Informationen finden Sie auf der "Support" Seite.

HAProxy Data Collector

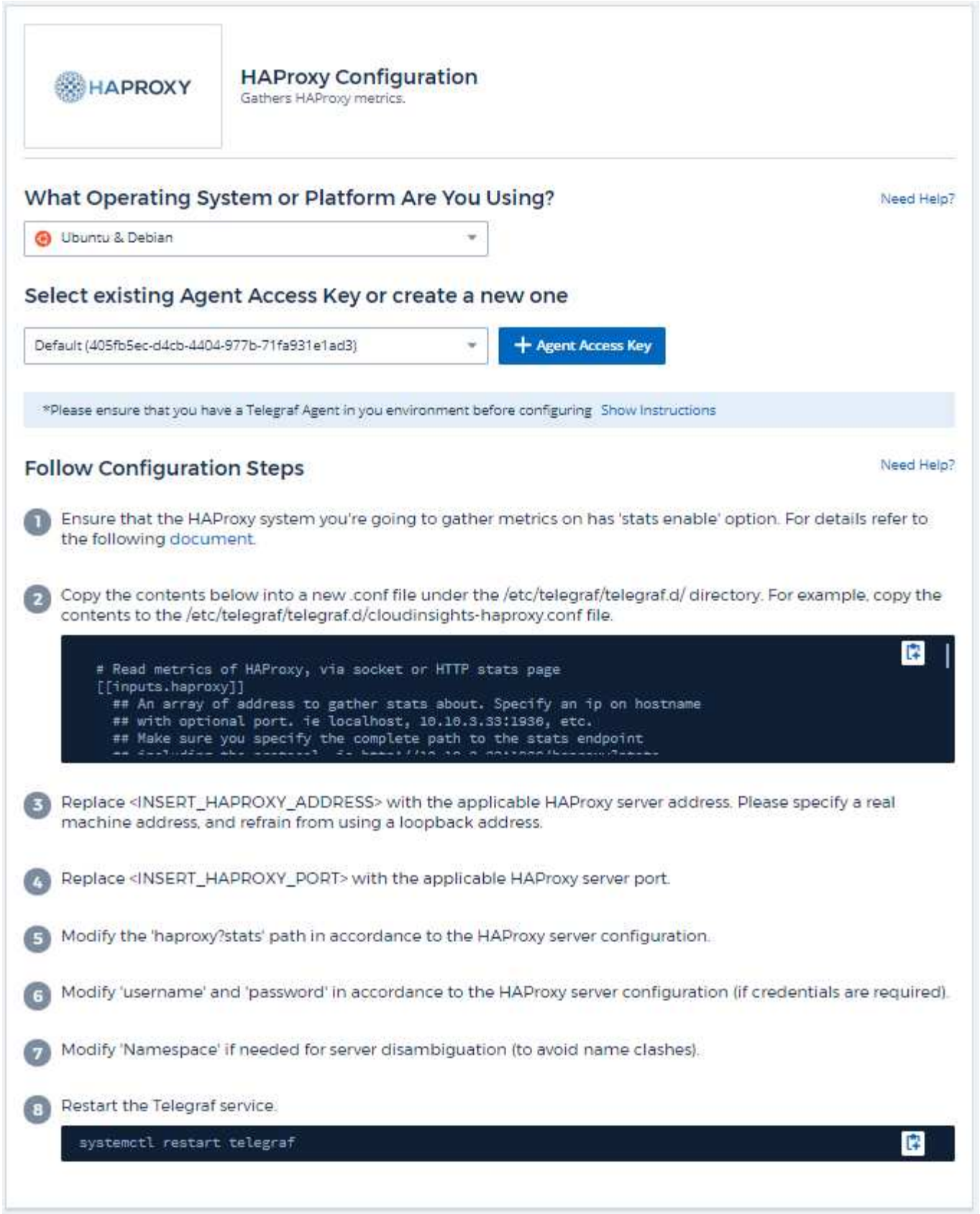
Data Infrastructure Insights verwendet diesen Datensammler, um Kennzahlen von HAProxy zu sammeln.

Installation

1. Klicken Sie unter **Observability > Collectors** auf **+Data Collector**. Wählen Sie HAProxy.

   Wählen Sie das Betriebssystem oder die Plattform aus, auf dem der Telegraf-Agent installiert ist.

2. Wenn Sie noch keinen Agenten zur Sammlung installiert haben oder einen Agenten für ein anderes Betriebssystem oder eine andere Plattform installieren möchten, klicken Sie auf *Anweisungen anzeigen*, um die Anweisungen zu erweitern"Agenten-Installation".
3. Wählen Sie den Agent-Zugriffsschlüssel für diesen Datensammler aus. Sie können einen neuen Agent-Zugriffsschlüssel hinzufügen, indem Sie auf die Schaltfläche **+ Agent Access Key** klicken. Best Practice: Verwenden Sie einen anderen Agent-Zugriffsschlüssel nur, wenn Sie Datensammler gruppieren möchten, zum Beispiel nach Betriebssystem/Plattform.
4. Befolgen Sie die Konfigurationsschritte, um den Datensammler zu konfigurieren. Die Anweisungen hängen vom Betriebssystem oder der Plattform ab, die Sie zur Datenerfassung verwenden.



Einrichtung

Telegraf's Plugin für HAProxy setzt auf HAProxy Stats Aktivierung. Diese Konfiguration ist in HAProxy integriert, ist jedoch nicht sofort aktiviert. Wenn HAProxy aktiviert ist, wird ein HTML-Endpunkt angezeigt, der

in Ihrem Browser angezeigt werden kann oder für die Extraktion des Status aller HAProxy-Konfigurationen abgekratzt werden kann.

Kompatibilität:

Die Konfiguration wurde gegen HAProxy-Version 1.9.4 entwickelt.

Einrichtung:

Um Statistiken zu aktivieren, bearbeiten Sie Ihre haproxy-Konfigurationsdatei und fügen Sie nach dem Abschnitt 'Standards' die folgenden Zeilen hinzu: Verwenden Sie Ihren eigenen Benutzer/Ihr Passwort und/oder die haproxy-URL:

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

Im Folgenden finden Sie eine vereinfachte Beispiel-Konfigurationsdatei mit aktivierten Statistiken:

```
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
```

Vollständige und aktuelle Anweisungen finden Sie im "HAProxy-Dokumentation".

Objekte und Zähler

Folgende Objekte und ihre Zähler werden gesammelt:

Objekt:	Kennungen:	Attribute:	Datenpunkte:
HAProxy Frontend	Namespace-Adressenproproxy	Node-IP-Knotenname Proxy-ID-Modus Prozess-id Sitzungen Ratenlimit Server-id Sitzungen Limit Status	Bytes in Bytes Out Cache Hits Cache Lookups Komprimierung Bytes umgangen Komprimierung Bytes in Komprimierung Bytes Out Komprimierung Reaktionen Verbindungsrate Verbindungsrate Max Verbindungen insgesamt Anträge, die von der Verbindung abgelehnt werden Rule Requests verweigert durch Sicherheitsbedenken Antworten verweigert durch Sicherheitsbedenken Anfragen abgelehnt durch Session Rule Requests erfragt Fehler Antworten 1xx Antworten 2xx Antworten 3xx Antworten 4xx Antworten 5xx Antworten andere Anfragen Abfangen Sitzungen Rate Sitzungen Max Anfragen Rate Max Anfragen Rate Max Anforderungen Total Sessions Sitzungen Max Sitzungen Antworten Neuschreibung Total Requests

Objekt:	Kennungen:	Attribute:	Datenpunkte:
HAProxy-Server	Namespace-Adresse-Proxy-Server	Node-IP-Knotenname Check Time to Finish Check Fall Configuration Check Health Value Check RISE Configuration Check Status Proxy ID Last Change Time Last Session Time Mode Process id Server Status Weight	Aktive Server Backup Server Bytes in Bytes Out Downs Check Downs Check Fails Client abgebrochen Verbindungen Verbindung Verbindung Durchschnittliche Zeit Ausfallzeit Gesamt Denied Responses Verbindungsfehler Antwort 1xx Antworten 2xx Antworten 3xx Antworten 4xx Antworten 5xx Antworten anderer Server ausgewählt Total Queue Current Queue Max. Durchschnittliche Zeit Sitzungen pro Zweite Sitzungen pro Sekunde Max. Wiederverwendbarkeit der Verbindung Reaktionszeit Durchschnittliche Sitzungen Sitzungen Max Server Transfer bricht Sitzungen gesamte Sitzungen Gesamtzeit Durchschnittliche Anforderungen Redispatches Anfragen Wiederholungen Anfragen Neuschreibung Anfragen

Objekt:	Kennungen:	Attribute:	Datenpunkte:
HAProxy-Back-End	Namespace-Adressenproproxy	Node-IP-Node-Name Proxy-ID Letzte Änderung Zeit Letzte Sitzung Zeitmodus Prozess-id Server-id Sitzungen Limit Status Gewicht	Aktive Server Backup Server Bytes in Bytes Out Cache Aufrufe Cache Lookups überprüfen Downs Client abbricht Komprimierung Bytes umgangen Komprimierung Bytes in Komprimierung Bytes out Komprimierungsantworten Verbindung Durchschnittliche Zeit Ausfallzeit Total Requests verweigert durch Sicherheitsbedenken Antworten verweigert durch Sicherheit Bedenken Verbindungsfehler Antworten Reaktion 1xx Antworten 2xx Antworten 3xx Antworten 4xx Antworten 5xx Antworten anderer Server ausgewählt Total Queue Current Queue Max. Warteschlange Durchschnittliche Zeit Sitzungen pro Sekunde Sitzungen pro Sekunde Max. Anfragen Gesamt Verbindungswiederverwendung Reaktionszeit Durchschnittliche Sitzungen Sitzungen Max. Serverübertragung Abtreibungen Sitzungen Gesamtzeit Durchschnittliche Anfragen Neuzuweisen Wiederholungsanfragen Wiederholungsanfragen Wiederholungsanfragen Wiederholungsanfragen Anträge Neu Schreiben

Fehlerbehebung

Weitere Informationen finden Sie auf der "Support" Seite.

JVM Data Collector

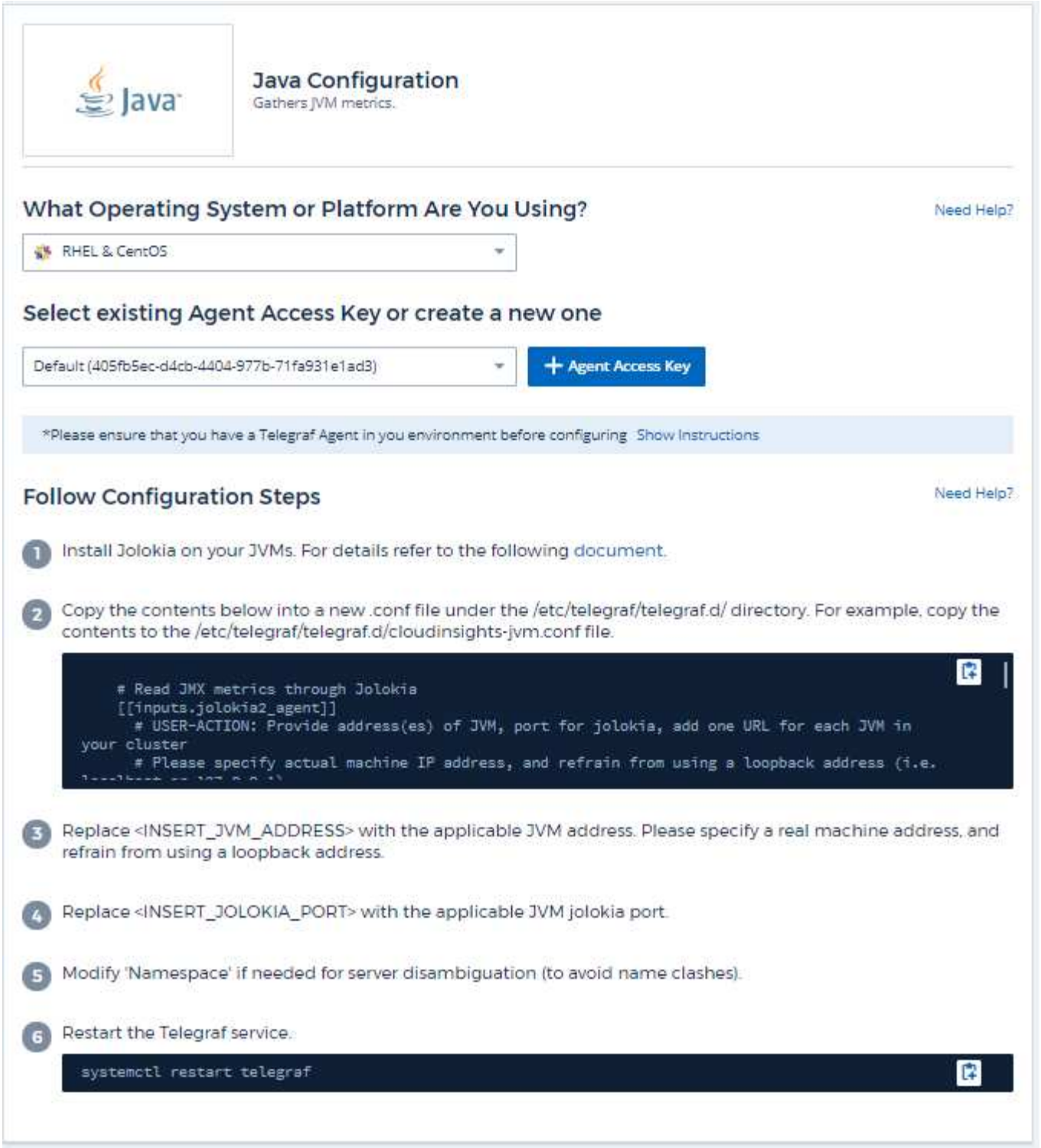
Data Infrastructure Insights verwendet diesen Datensammler zur Erfassung von Kennzahlen aus JVM.

Installation

1. Klicken Sie unter **Observability > Collectors** auf **+Data Collector**. Wählen Sie JVM.

   Wählen Sie das Betriebssystem oder die Plattform aus, auf dem der Telegraf-Agent installiert ist.

2. Wenn Sie noch keinen Agenten zur Sammlung installiert haben oder einen Agenten für ein anderes Betriebssystem oder eine andere Plattform installieren möchten, klicken Sie auf *Anweisungen anzeigen*, um die Anweisungen zu erweitern"Agenten-Installation".
3. Wählen Sie den Agent-Zugriffsschlüssel für diesen Datensammler aus. Sie können einen neuen Agent-Zugriffsschlüssel hinzufügen, indem Sie auf die Schaltfläche **+ Agent Access Key** klicken. Best Practice: Verwenden Sie einen anderen Agent-Zugriffsschlüssel nur, wenn Sie Datensammler gruppieren möchten, zum Beispiel nach Betriebssystem/Plattform.
4. Befolgen Sie die Konfigurationsschritte, um den Datensammler zu konfigurieren. Die Anweisungen hängen vom Betriebssystem oder der Plattform ab, die Sie zur Datenerfassung verwenden.



Einrichtung

Informationen finden Sie unter "JVM-Dokumentation".

Objekte und Zähler

Folgende Objekte und ihre Zähler werden gesammelt:

Objekt:	Kennungen:	Attribute:	Datenpunkte:
JVM	Namespace-JVM	OS Architektur OS Name OS Version Laufzeit Spezifikation Laufzeit Spezifikation Hersteller Laufzeit Spezifikation Version Uptime Laufzeit VM Name Laufzeit VM Anbieter Laufzeit VM Version Node Name Node IP	Class Loaded Class Loaded Class Memory Unloaded Memory Heap Init Memory Heap used Max Memory Heap Used Memory Non Heap Innit Memory Non Heap Max Memory nicht Heap Used Memory Objects Ausstehende Fertigstellung von Betriebssystemprozessoren verfügbar Betriebssystem engagierte virtuelle Speichergröße OS Kostenlos Physikalische Speichergröße OS Freier Swap Speicherplatz Größe OS Max Datei Descriptor Anzahl OS Open File Descriptors Anzahl Betriebssystem Prozessor CPU Load OS CPU Time OS System CPU Load OS System Load Average OS Gesamt Physical Memory Size OS Gesamt Swap Space Size Thread Daemon Anzahl der Threads Spitzenanzahl Thread Count Thread Total Started Count Garbage Collector Copy Collection Count Garbage Collector Copy Collection Time Garbage Collector Sammlung von Mark-Sweep Sammlungszeit Zeitabfälle Collector G1 Sammlung der Alten Generation Speicherbage Collector G1 Zeitabbage der Jungen Generation Sammlungszähler Garbage Collector G1 Young Generation Collection Time Garbage Collector Zeitabfälle Sammlung der aktuellen Mark-Sweep Sammlung Zeitgarage Collector Parallel Collection Count Garbage Collector Parallel

Fehlerbehebung

Weitere Informationen finden Sie auf der "Support" Seite.

Kafka Data Collector

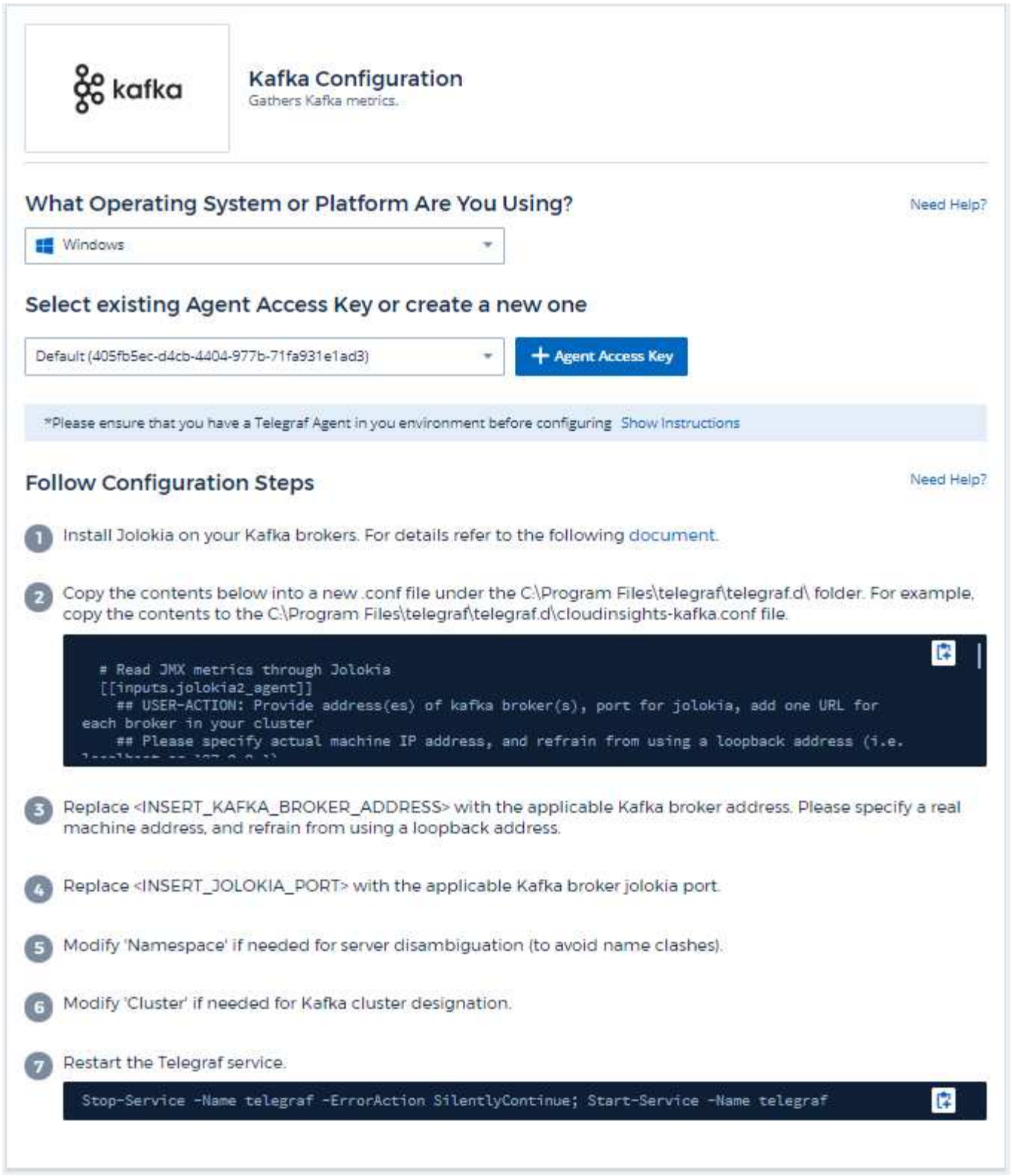
Data Infrastructure Insights nutzt diesen Datensammler, um Kennzahlen aus Kafka zu sammeln.

Installation

1. Klicken Sie unter **Observability > Collectors** auf **+Data Collector**. Wählen Sie Kafka.

   Wählen Sie das Betriebssystem oder die Plattform aus, auf dem der Telegraf-Agent installiert ist.

2. Wenn Sie noch keinen Agenten zur Sammlung installiert haben oder einen Agenten für ein anderes Betriebssystem oder eine andere Plattform installieren möchten, klicken Sie auf *Anweisungen anzeigen*, um die Anweisungen zu erweitern"Agenten-Installation".
3. Wählen Sie den Agent-Zugriffsschlüssel für diesen Datensammler aus. Sie können einen neuen Agent-Zugriffsschlüssel hinzufügen, indem Sie auf die Schaltfläche **+ Agent Access Key** klicken. Best Practice: Verwenden Sie einen anderen Agent-Zugriffsschlüssel nur, wenn Sie Datensammler gruppieren möchten, zum Beispiel nach Betriebssystem/Plattform.
4. Befolgen Sie die Konfigurationsschritte, um den Datensammler zu konfigurieren. Die Anweisungen hängen vom Betriebssystem oder der Plattform ab, die Sie zur Datenerfassung verwenden.



Einrichtung

Das Kafka Plugin basiert auf dem telegraf's Jolokia Plugin. Um Informationen aus allen Kafka-Brokern zu sammeln, muss JMX über Jolokia auf allen Komponenten konfiguriert und zugänglich gemacht werden.

Kompatibilität

Konfiguration wurde gegen Kafka Version 0.11.0 entwickelt.

Einrichtung

Alle Anweisungen unten Nehmen wir an, dass Ihr Installationsort für kafka '/opt/kafka' ist. Sie können die nachfolgenden Anweisungen an Ihren Installationsort anpassen.

Jolokia Agent Jar

Eine Version die Jolokia Agent jar-Datei muss sein"Heruntergeladen". Die gegen die Version getestete war Jolokia Agent 1.6.0.

Anweisungen unten gehen davon aus, dass die heruntergeladene JAR-Datei (jolokia-jvm-1.6.0-Agent.jar) unter dem Speicherort '/opt/kafka/libs/' abgelegt wird.

Kafka Brokers

Um Kafka Brokers so zu konfigurieren, dass sie die Jolokia API aussetzen, können Sie in <KAFKA_HOME>/bin/kafka-Server-Start.sh kurz vor dem Anruf „kafka-run-class.sh“ Folgendes hinzufügen:

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

Beachten Sie, dass das obige Beispiel 'Hostname -i' verwendet, um die Umgebungsvariable 'RMI_HOSTNAME' einzurichten. In mehreren IP-Maschinen muss dies optimiert werden, um die IP, die Sie für RMI-Verbindungen interessieren, zu erfassen.

Sie können einen anderen Port für JMX (9999 oben) und Jolokia (8778) wählen. Wenn Sie eine interne IP haben, um Jolokia zu sperren, können Sie die „Catch all“ 0.0.0.0 durch Ihre eigene IP ersetzen. Beachten Sie, dass diese IP über das telegraf-Plugin zugänglich sein muss. Sie können die Option '-Dcom.sun.management.jmxremote.authenticate=false' verwenden, wenn Sie nicht authentifizieren möchten. Nutzung auf eigenes Risiko.

Objekte und Zähler

Folgende Objekte und ihre Zähler werden gesammelt:

Objekt:	Kennungen:	Attribute:
Kafka Broker	Cluster Namespace Broker	Node Name Node-IP

Fehlerbehebung

Weitere Informationen finden Sie auf der "Support" Seite.

Kibana Data Collector

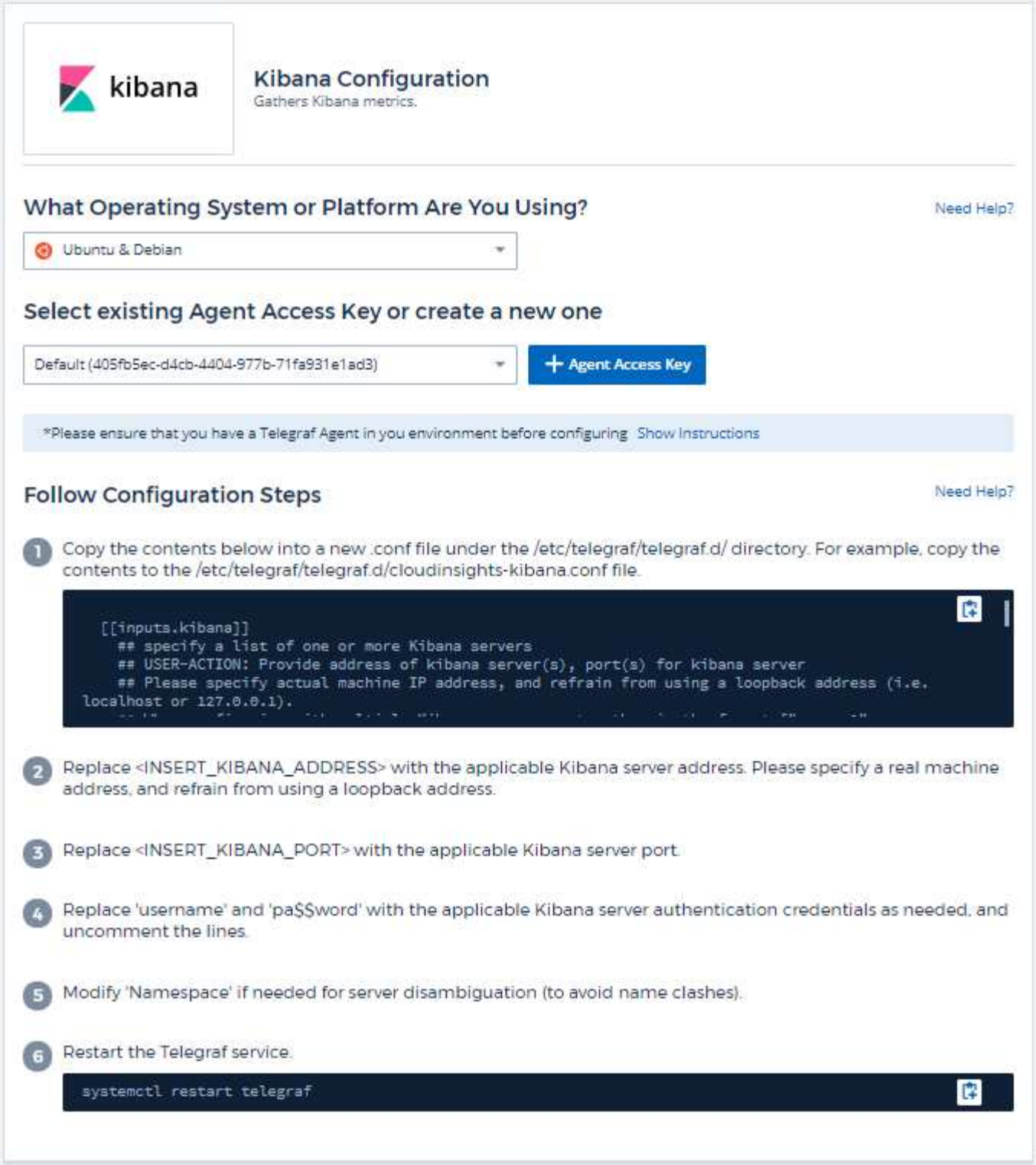
Data Infrastructure Insights verwendet diesen Datensammler, um Kennzahlen von Kibana zu sammeln.

Installation

1. Klicken Sie unter **Observability > Collectors** auf **+Data Collector**. Wählen Sie Kibana.

   Wählen Sie das Betriebssystem oder die Plattform aus, auf dem der Telegraf-Agent installiert ist.

2. Wenn Sie noch keinen Agenten zur Sammlung installiert haben oder einen Agenten für ein anderes Betriebssystem oder eine andere Plattform installieren möchten, klicken Sie auf *Anweisungen anzeigen*, um die Anweisungen zu erweitern"Agenten-Installation".
3. Wählen Sie den Agent-Zugriffsschlüssel für diesen Datensammler aus. Sie können einen neuen Agent-Zugriffsschlüssel hinzufügen, indem Sie auf die Schaltfläche **+ Agent Access Key** klicken. Best Practice: Verwenden Sie einen anderen Agent-Zugriffsschlüssel nur, wenn Sie Datensammler gruppieren möchten, zum Beispiel nach Betriebssystem/Plattform.
4. Befolgen Sie die Konfigurationsschritte, um den Datensammler zu konfigurieren. Die Anweisungen hängen vom Betriebssystem oder der Plattform ab, die Sie zur Datenerfassung verwenden.



Einrichtung

Informationen finden Sie im "Kibana Dokumentation".

Objekte und Zähler

Folgende Objekte und ihre Zähler werden gesammelt:

Objekt:	Kennungen:	Attribute:	Datenpunkte:
Kibana	Namespace-Adresse	Versionsstatus des Node-IP-Node-Namens	Gleichzeitige Verbindungen Heap Max Heap verwendete Anforderungen pro Sekunde Antwortzeit Max. Betriebszeit

Fehlerbehebung

Weitere Informationen finden Sie auf der "Support" Seite.

Installation und Konfiguration des Kubernetes Monitoring Operator

Data Infrastructure Insights bietet den **Kubernetes Monitoring Operator** für die Kubernetes-Sammlung an. Navigieren Sie zu **Kubernetes > Collectors > +Kubernetes Collector**, um einen neuen Operator bereitzustellen.

Bevor Sie den Kubernetes Monitoring Operator installieren

Lesen Sie die "Voraussetzungen" Dokumentation, bevor Sie den Kubernetes Monitoring Operator installieren oder aktualisieren.

Installieren des Kubernetes Monitoring Operator

kubernetes

Kubernetes

Deploy NetApp Monitoring Operator

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

Select existing API Access Token or create a new one

KEY2024 (...vw6NdM)

+ API Access Token

Production Best Practices

Installation Instructions

Need Help?

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

1 Define Kubernetes cluster name and namespace

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

Cluster	Namespace
clustername	netapp-monitoring

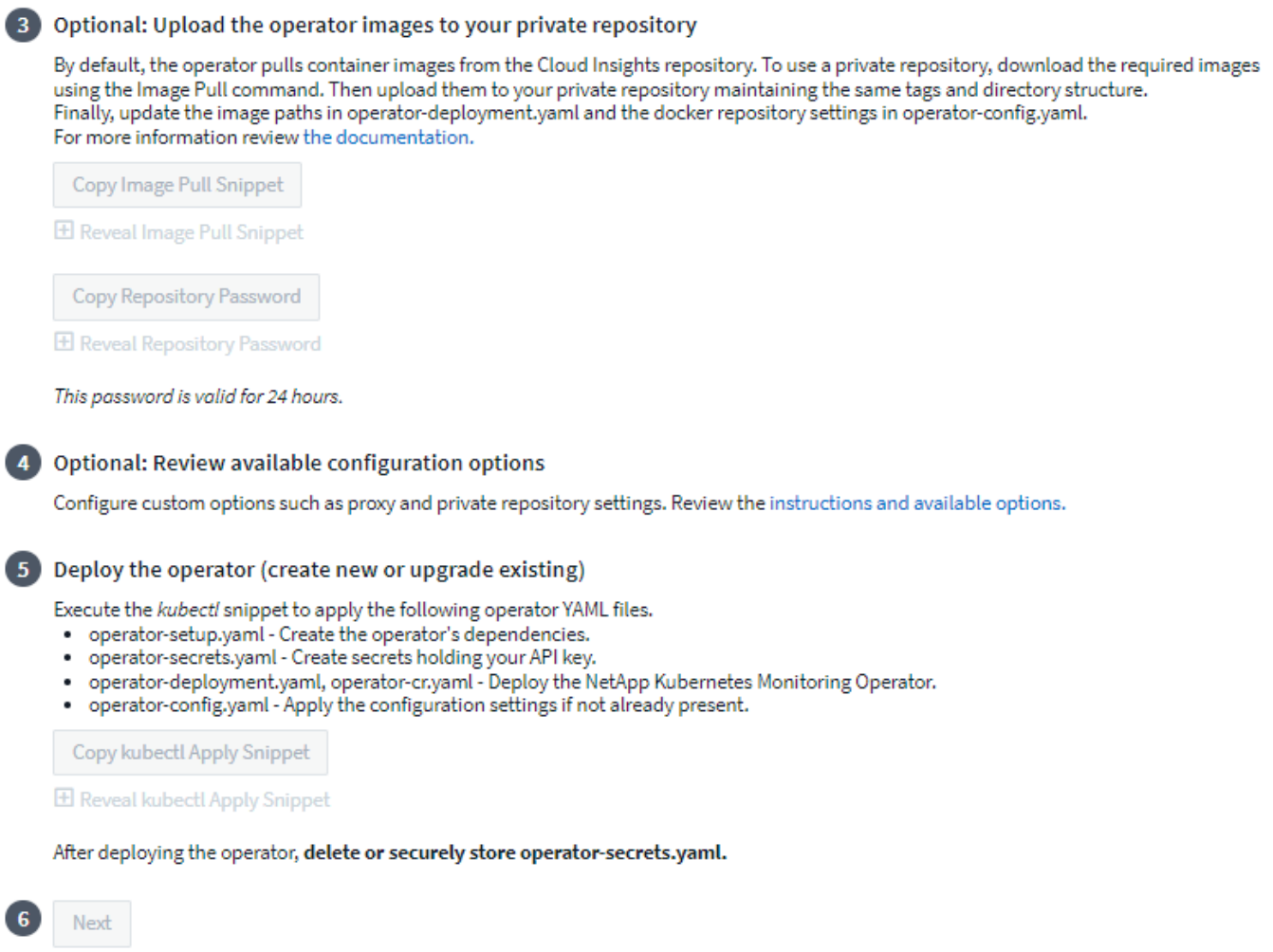
2 Download the operator YAML files

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

Copy Download Command Snippet

Reveal Download Command Snippet

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



Schritte zum Installieren des Kubernetes Monitoring Operator Agent auf Kubernetes:

1. Geben Sie einen eindeutigen Cluster-Namen und einen eindeutigen Namespace ein. Wenn Sie von einem früheren Kubernetes-Operator stammenAktualisierung, verwenden Sie den gleichen Cluster-Namen und den gleichen Namespace.
2. Sobald diese eingegeben wurden, können Sie den Download-Befehl-Snippet in die Zwischenablage kopieren.
3. Fügen Sie das Snippet in ein *bash* Fenster ein und führen Sie es aus. Die Installationsdateien des Bedieners werden heruntergeladen. Beachten Sie, dass das Snippet einen eindeutigen Schlüssel hat und für 24 Stunden gültig ist.
4. Wenn Sie ein benutzerdefiniertes oder privates Repository haben, kopieren Sie das optionale Bild-Pull-Snippet, fügen Sie es in eine *bash*-Shell ein und führen Sie es aus. Nachdem die Bilder gezogen wurden, kopieren Sie sie in Ihr privates Repository. Stellen Sie sicher, dass Sie dieselben Tags und Ordnerstrukturen beibehalten. Aktualisieren Sie die Pfade in *Operator-Deployment.yaml* sowie die Einstellungen des Docker-Repository in *Operator-config.yaml*.
5. Prüfen Sie bei Bedarf die verfügbaren Konfigurationsoptionen, z. B. Proxy- oder private Repository-Einstellungen. Lesen Sie mehr über "Konfigurationsoptionen".
6. Wenn Sie bereit sind, stellen Sie den Operator bereit, indem Sie den kubectl Apply-Snippet kopieren, herunterladen und ausführen.
7. Die Installation wird automatisch ausgeführt. Klicken Sie anschließend auf die Schaltfläche „*Next*“.

8. Wenn die Installation abgeschlossen ist, klicken Sie auf die Schaltfläche „*Next*“. Achten Sie darauf, auch die Datei *Operator-Secrets.yaml* zu löschen oder sicher zu speichern.

Wenn Sie einen Proxy verwenden, lesen Sie über Proxy wird konfiguriert.

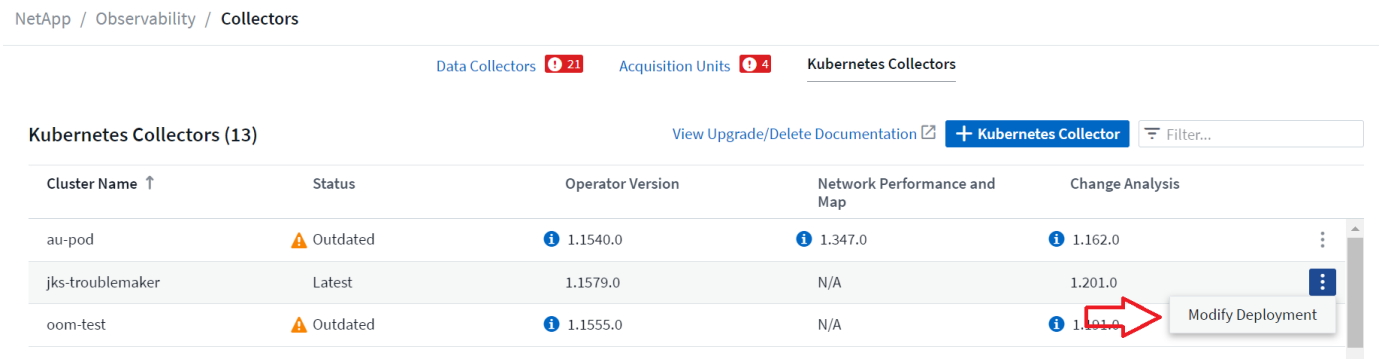
Wenn Sie ein benutzerdefiniertes Repository haben, lesen Sie über Ein benutzerdefiniertes/privates Docker-Repository verwenden.

Kubernetes-Monitoring-Komponenten

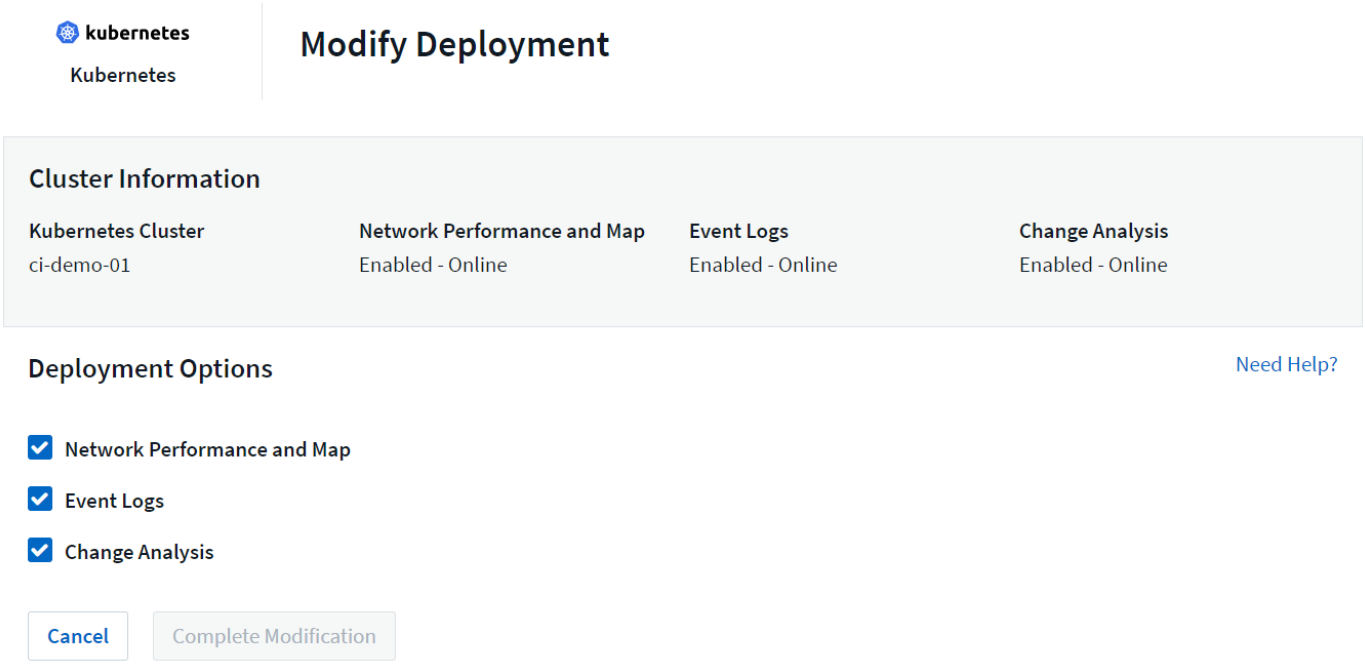
Data Infrastructure Insights Kubernetes Monitoring besteht aus vier Monitoring-Komponenten:

- Cluster-Kennzahlen
- Netzwerkleistung und -Zuordnung (optional)
- Ereignisprotokolle (optional)
- Änderungsanalyse (optional)

Die oben aufgeführten optionalen Komponenten sind standardmäßig für jeden Kubernetes-Collector aktiviert. Wenn Sie sich entscheiden, keine Komponente für einen bestimmten Collector zu benötigen, können Sie sie deaktivieren, indem Sie zu **Kubernetes > Collectors** navigieren und im Collector-Menü „drei Punkte“ rechts auf dem Bildschirm *Modify Deployment* auswählen.



Der Bildschirm zeigt den aktuellen Status jeder Komponente an und ermöglicht es Ihnen, Komponenten für diesen Collector nach Bedarf zu deaktivieren oder zu aktivieren.



Upgrade auf den neuesten Kubernetes Monitoring Operator

Ermitteln Sie, ob eine AgentConfiguration bei dem vorhandenen Operator vorhanden ist (wenn Ihr Namespace nicht der Standardwert *netapp-Monitoring* ist, ersetzen Sie den entsprechenden Namespace):

```
 kubectl -n netapp-monitoring get agentconfiguration netapp-monitoring-
configuration
Wenn eine AgentConfiguration vorhanden ist:
```

- Installieren Der letzte Operator über den vorhandenen Operator.
  - Stellen Sie sicher, dass Die neuesten Container-Bilder werden angezeigtSie ein benutzerdefiniertes Repository verwenden.

Wenn AgentConfiguration nicht vorhanden ist:

- Notieren Sie sich den von Data Infrastructure Insights erkannten Cluster-Namen (wenn Ihr Namespace nicht das standardmäßige NetApp-Monitoring ist, ersetzen Sie den entsprechenden Namespace):

```
 kubectl -n netapp-monitoring get agent -o
jsonpath='{.items[0].spec.cluster-name}'
* Erstellen Sie eine Sicherung des bestehenden Operators (wenn Ihr
Namespace nicht der Standard-netapp-Überwachung ist, ersetzen Sie den
entsprechenden Namespace):
```

```
 kubectl -n netapp-monitoring get agent -o yaml > agent_backup.yaml
* <<to-remove-the-kubernetes-monitoring-operator,Deinstallieren>> Der
vorhandene Operator.
* <<installing-the-kubernetes-monitoring-operator,Installieren>> Der
neueste Bediener.
```

- Verwenden Sie denselben Cluster-Namen.
- Nachdem Sie die neuesten Operator YAML-Dateien heruntergeladen haben, können Sie alle in Agent_Backup.yaml gefundenen Anpassungen vor der Bereitstellung an den heruntergeladenen Operator-config.yaml übertragen.
- Stellen Sie sicher, dass Die neuesten Container-Bilder werden angezeigtSie ein benutzerdefiniertes Repository verwenden.

Anhalten und Starten des Kubernetes Monitoring Operator

So beenden Sie den Kubernetes Monitoring Operator:

```
 kubectl -n netapp-monitoring scale deploy monitoring-operator
--replicas=0
So starten Sie den Kubernetes Monitoring Operator:
```

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

Deinstallation

Um den Kubernetes Monitoring Operator zu entfernen

Beachten Sie, dass der Standard-Namespace für den Kubernetes Monitoring Operator „netapp-Monitoring“ ist. Wenn Sie Ihren eigenen Namespace festgelegt haben, ersetzen Sie diesen Namespace in diesen und allen nachfolgenden Befehlen und Dateien.

Neuere Versionen des Überwachungsoperators können mit den folgenden Befehlen deinstalliert werden:

```
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>
```

Wenn der Überwachungsoperator in seinem eigenen dedizierten Namespace bereitgestellt wurde, löschen Sie den Namespace:

```
 kubectl delete ns <NAMESPACE>
Wenn der erste Befehl „Keine Ressourcen gefunden“ zurückgibt, verwenden
Sie die folgenden Anweisungen, um ältere Versionen des
Überwachungsoperators zu deinstallieren.
```

Führen Sie jeden der folgenden Befehle in der Reihenfolge aus. Abhängig von Ihrer aktuellen Installation können einige dieser Befehle Nachrichten ‘object not found’ zurückgeben. Diese Meldungen können sicher ignoriert werden.

```
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>
```

Wenn zuvor eine Sicherheitskontextbeschränkung erstellt wurde:

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

Über Kube-State-Metrics

Der NetApp Kubernetes Monitoring Operator installiert seine eigenen kube-State-Metriken, um Konflikte mit anderen Instanzen zu vermeiden.

Informationen über Kube-State-Metrics finden Sie unter "Auf dieser Seite".

Konfigurieren/Anpassen des Bedieners

Diese Abschnitte enthalten Informationen zur Anpassung Ihrer Bedienerkonfiguration, zur Arbeit mit Proxy, zur Verwendung eines benutzerdefinierten oder privaten Docker-Repositorys oder zur Arbeit mit OpenShift.

Konfigurationsoptionen

Die am häufigsten geänderten Einstellungen können in der benutzerdefinierten Ressource *AgentConfiguration* konfiguriert werden. Sie können diese Ressource bearbeiten, bevor Sie den Operator bereitstellen, indem Sie die Datei *Operator-config.yaml* bearbeiten. Diese Datei enthält kommentierte Beispiele für Einstellungen. In der Liste "Verfügbare Einstellungen"finden Sie die aktuellste Version des Operators.

Sie können diese Ressource auch bearbeiten, nachdem der Operator bereitgestellt wurde, indem Sie den

folgenden Befehl verwenden:

```
 kubectl -n netapp-monitoring edit AgentConfiguration
Um festzustellen, ob die bereitgestellte Version des Operators
AgentConfiguration unterstützt, führen Sie den folgenden Befehl aus:
```

```
 kubectl get crd agentconfigurations.monitoring.netapp.com
Wenn die Meldung „Fehler vom Server (notfound)“ angezeigt wird, muss Ihr
Bediener aktualisiert werden, bevor Sie die AgentConfiguration verwenden
können.
```

Proxy-Unterstützung Wird Konfiguriert

An zwei Stellen können Sie einen Proxy für Ihren Mandanten verwenden, um den Kubernetes Monitoring Operator zu installieren. Es kann sich um dieselben oder separate Proxy-Systeme handelt:

- Proxy wird während der Ausführung des Installationscode-Snippets (mit „Curl“) benötigt, um das System zu verbinden, auf dem das Snippet ausgeführt wird, mit Ihrer Data Infrastructure Insights-Umgebung
- Der vom Kubernetes Ziel-Cluster benötigte Proxy für die Kommunikation mit der Insights Umgebung Ihrer Dateninfrastruktur ist erforderlich

Wenn Sie einen Proxy für eine oder beide dieser Optionen verwenden, müssen Sie zur Installation des Kubernetes Operating Monitor zunächst sicherstellen, dass Ihr Proxy so konfiguriert ist, dass eine gute Kommunikation mit Ihrer Data Infrastructure Insights-Umgebung möglich ist. Wenn Sie über einen Proxy verfügen und von dem Server/der VM, von dem aus Sie den Operator installieren möchten, auf Data Infrastructure Insights zugreifen können, ist Ihr Proxy wahrscheinlich richtig konfiguriert.

Für den Proxy, der zur Installation des Kubernetes Operating Monitor verwendet wird, legen Sie vor der Installation des Operators die Umgebungsvariablen *http_Proxy/https_Proxy* fest. In einigen Proxy-Umgebungen müssen Sie möglicherweise auch die Variable *no_Proxy Environment* festlegen.

Um die Variablen festzulegen, führen Sie die folgenden Schritte auf Ihrem System aus * bevor* den Kubernetes Monitoring Operator installiert:

1. Legen Sie die Umgebungsvariable *https_Proxy* und/oder *http_Proxy* für den aktuellen Benutzer fest:
   a. Wenn der Proxy, der eingerichtet wird, keine Authentifizierung (Benutzername/Passwort) aufweist, führen Sie den folgenden Befehl aus:

```
 export https_proxy=<proxy_server>:<proxy_port>
.. Wenn der Proxy, der eingerichtet wird, über Authentifizierung
(Benutzername/Passwort) verfügt, führen Sie folgenden Befehl aus:
```

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

Wenn der Proxy, der für das Kubernetes-Cluster zur Kommunikation mit der Insights Umgebung Ihrer Dateninfrastruktur verwendet wird, verwendet wird, installieren Sie den Kubernetes Monitoring Operator, nachdem Sie alle diese Anweisungen gelesen haben.

Konfigurieren Sie den Proxy-Abschnitt von AgentConfiguration in Operator-config.yaml, bevor Sie den Kubernetes Monitoring Operator bereitstellen.

```
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
  ...
...
```

Verwenden eines benutzerdefinierten oder privaten Docker Repositorys

Standardmäßig zieht der Kubernetes Monitoring Operator Container-Images aus dem Repository Data Infrastructure Insights. Wenn Sie ein Kubernetes-Cluster als Ziel für das Monitoring verwenden und der Cluster so konfiguriert ist, dass er nur Container-Images aus einem benutzerdefinierten oder privaten Docker-Repository oder der Container-Registrierung zieht, müssen Sie den Zugriff auf die Container konfigurieren, die vom Kubernetes Monitoring Operator benötigt werden.

Führen Sie das „Image Pull Snippet“ aus der NetApp Monitoring Operator Installationskachel aus. Dieser Befehl meldet sich beim Repository Data Infrastructure Insights an, zieht alle Image-Abhängigkeiten für den Operator ab und meldet sich vom Repository Data Infrastructure Insights ab. Wenn Sie dazu aufgefordert werden, geben Sie das angegebene temporäre Repository-Passwort ein. Mit diesem Befehl werden alle vom Bediener verwendeten Bilder heruntergeladen, einschließlich optionaler Funktionen. Nachfolgend sehen Sie, für welche Funktionen diese Bilder verwendet werden.

Core Operator-Funktionalität und Kubernetes Monitoring

- netapp Monitoring
- ci-kube-rbac-Proxy
- ci-ksm

- ci-telegraf
- Distroless-root-user

Ereignisprotokoll

- ci-Fluent-Bit
- ci-kubernetes-Event-Exporteur

Netzwerkleistung und -Zuordnung

- ci-Netz-Beobachter

Übertragen Sie das Operator-Docker-Image gemäß Ihren Unternehmensrichtlinien in das private/lokale/unternehmenseigene Docker-Repository. Stellen Sie sicher, dass die Bild-Tags und Verzeichnispfade zu diesen Images in Ihrem Repository mit denen im Data Infrastructure Insights Repository übereinstimmen.

Bearbeiten Sie die Bereitstellung des Monitoring-Operators in Operator-Deployment.yaml, und ändern Sie alle Bildverweise, um Ihr privates Docker-Repository zu verwenden.

```
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>
```

Bearbeiten Sie die AgentConfiguration in Operator-config.yaml, um die neue Position des Docker-Repo zu berücksichtigen. Erstellen Sie ein neues imagePullSecret für Ihr privates Repository. Weitere Informationen finden Sie unter *https://kubernetes.io/docs/tasks/configure-pod-container/pull-image-private-registry/*

```
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-Anweisungen

Wenn Sie OpenShift 4.6 oder höher ausführen, müssen Sie die AgentConfiguration in *Operator-config.yaml* bearbeiten, um die Einstellung *runPrivileged* zu aktivieren:

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

OpenShift kann zusätzliche Sicherheitsstufen implementieren, die den Zugriff auf einige Kubernetes-Komponenten blockieren könnten.

Toleranzen und Verfleckungen

Die DemonSets *netapp-CI-telegraf-ds*, *netapp-CI-Fluent-Bit-ds* und *netapp-CI-net-Observer-l4-ds* müssen für jeden Node im Cluster einen Pod planen, damit Daten auf allen Nodes korrekt erfasst werden. Der Operator wurde so konfiguriert, dass er einige bekannte **Fehler** toleriert. Wenn Sie auf Ihren Knoten benutzerdefinierte Taints konfiguriert haben und damit verhindern, dass Pods auf jedem Knoten ausgeführt werden, können Sie für diese Taints eine **Toleration** erstellen"In der *AgentConfiguration*". Wenn Sie auf alle Nodes im Cluster benutzerdefinierte Taints angewendet haben, müssen Sie der Operator-Bereitstellung auch die erforderlichen Toleranzen hinzufügen, damit der Operator-Pod geplant und ausgeführt werden kann.

Erfahren Sie mehr über Kubernetes "Tönungen und Tolerationen".

Kehren Sie zum zurück **"NetApp Kubernetes Monitoring Operator Installation** Seite"

Ein Hinweis über Geheimnisse

Um die Berechtigung für den Kubernetes Monitoring Operator zum Anzeigen der geheimen Daten im gesamten Cluster zu entfernen, löschen Sie vor der Installation die folgenden Ressourcen aus der Datei *Operator-Setup.yaml*:

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

Wenn es sich um ein Upgrade handelt, löschen Sie auch die Ressourcen aus Ihrem Cluster:

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

Wenn die Änderungsanalyse aktiviert ist, ändern Sie die Optionen *AgentConfiguration* oder *Operator-config.yaml*, um den Änderungsmanagementabschnitt zu entkommentieren und *kindsToIgnoreFromWatch: '"Secrets"* im Bereich Change-Management aufzunehmen. Notieren Sie sich das Vorhandensein und die Position von einfachen und doppelten Anführungszeichen in dieser Zeile.

```
# 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"'
  ...
```

Überprüfen Der Signaturen Der Kubernetes Monitoring Operator Images

Das Bild für den Betreiber und alle damit verbundenen Bilder werden von NetApp signiert. Sie können die Images vor der Installation mit dem cosign-Tool manuell überprüfen oder einen Kubernetes-Aufnahme-Controller konfigurieren. Weitere Informationen finden Sie im "Kubernetes-Dokumentation".

Der öffentliche Schlüssel, der zur Überprüfung der Bildsignaturen verwendet wird, ist in der Kachel Monitoring Operator install unter *Optional: Laden Sie die Operatorbilder in Ihr privates Repository > Image Signature Public Key*

So überprüfen Sie eine Bildsignatur manuell:

1. Kopieren Sie das Bild-Pull-Snippet, und führen Sie es aus
2. Kopieren Sie das Repository-Kennwort, und geben Sie es ein, wenn Sie dazu aufgefordert werden
3. Speichern Sie den Public Key der Bildsignatur (im Beispiel dii-image-signing.Pub).
4. Überprüfen Sie die Bilder mit cosign. Beachten Sie das folgende Beispiel für die Verwendung von Cosign

```
$ 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}]
```

Fehlerbehebung

Bei Problemen beim Einrichten des Kubernetes Monitoring Operator sollten Sie Folgendes versuchen:

Problem:	Versuchen Sie dies:
Ich sehe keinen Hyperlink/Verbindung zwischen meinem Kubernetes Persistent Volume und dem entsprechenden Back-End Storage-Gerät. Mein Kubernetes Persistent Volume wird mit dem Hostnamen des Storage-Servers konfiguriert.	Befolgen Sie die Schritte, um den bestehenden Telegraf-Agent zu deinstallieren, und installieren Sie dann den neuesten Telegraf-Agent erneut. Sie müssen Telegraf Version 2.0 oder höher verwenden. Der Kubernetes-Cluster-Storage muss aktiv durch Data Infrastructure Insights überwacht werden.
Ich sehe Meldungen in den Protokollen, die folgende ähneln: E0901 15 352:21:39.962145 1 Reflektor.go:178] k8s.io/kube-State-metrics/internal/Store/Builder.go:352: Fehler beim Auflisten *v1.MutatingWebhookKonfiguration: Der Server konnte die angeforderte Ressource E0901 15:21:43.168161 1 Reflector.go:178] k8s.io/kube-Builder nicht finden	Diese Nachrichten können auftreten, wenn Sie kube-State-Metrics Version 2.0.0 oder höher mit Kubernetes-Versionen unter 1.20 ausführen. Um die Kubernetes-Version zu erhalten: *Kubectl Version* um die kube-State-metrics-Version zu erhalten: *Kubectl get Deploy/kube-State-metrics -o jsonpath='{..image}'* um zu verhindern, dass diese Nachrichten passieren, können Benutzer ihre kube-State-Metrics-Implementierung ändern, um die folgenden Elemente zu deaktivieren: _Mutingwebhookkonfigurationen___volumehaWeitere Resources=certificationesigningrequests,configmaps, cronjobs,dämsets, Bereitstellungen,Endpunkte,HorizontalpodAutoscaler, nesresses,Jobs,Begrenzungsbereiche,Namensräume, Netzwerkrichtlinien,Knoten,Persistenz,stagemasnesmases,nesmasnesmases,nesmasnesmasnesmasnesnesmasnesequets,ndecoses,nescontascrises,nesequequequesefises,nesequequesequesefiscones,mases,nesequidatequesequesefiscones,nesequesequesefiscrises,nesequesequesefiscones,nesefisconesefisconmases,mases,nesequesequesefiscones,necequesequeseques Validatingwebhookkonfigurationen, Volumeanhänge“
Ich sehe Fehlermeldungen von Telegraf ähnlich wie die folgenden, aber Telegraf startet und läuft: Okt 11 14:23:41 ip-172-31-39-47 systemd[1]: Startete den Plugin-getriebenen Server Agent für das Reporting von Metriken in InfluxDB. Okt 11 14:23:41 ip-172-31-39-47 telegraf[1827]: Time=„2021-10-11T14:23:41Z“ Level=error msg=„konnte kein Cache-Verzeichnis erstellen. /Etc/telegraf/.Cache/snowflake, err: Mkdir /etc/telegraf/.ca che: Berechtigung verweigert. Ignored\n" func=„gosnowflake.(*defaultLogger).Errorf“ file=„log.go:1827 23“ Okt 31 2021:39-47 10 ip-172-11 14-23:41 telegraf[120]: Time=„41-11TZ Fehler“:41T14=. Ignored. Open /etc/telegraf/.Cache/snowflake/ocsp_response_Cache .json: No such file or Directory\n" func=„gosnowflake.(*defaultLogger).Errorf“ file=„log.go:23“ Okt 2021:10 ip-1827-31-39-47 telegraf[172]: 11 14-23:41-11T11T14:120:41Z I! Telegraf 1.19.3 Starten	Dies ist ein bekanntes Problem. "Dieser GitHub-Artikel"Weitere Informationen finden Sie unter. Solange Telegraf läuft, können Benutzer diese Fehlermeldungen ignorieren.

Problem:	Versuchen Sie dies:
Auf Kubernetes meldet mein Telegraf pod(s) den folgenden Fehler: „Fehler in der Verarbeitung von mountstats-Infos: Habe mountstats-Datei nicht geöffnet: /Hostfs/proc/1/mountstats, Fehler: Open /hostfs/proc/1/mountstats: Permission dementied“	Wenn SELinux aktiviert und durchgesetzt wird, wird wahrscheinlich verhindert, dass die Telegraf PODs auf die Datei /proc/1/mountstats auf dem Kubernetes-Knoten zugreifen. Um diese Einschränkung zu überwinden, bearbeiten Sie die Agentkonfiguration und aktivieren Sie die runPrivileged-Einstellung. Weitere Informationen finden Sie im "OpenShift-Anweisungen".
Auf Kubernetes meldet mein Telegraf ReplicaSet POD den folgenden Fehler: [inputs.prometheus] Fehler im Plugin: Konnte keine keypair /etc/kubernetes/pki/etcd/Server.crt:/etc/kubernetes/pki /etcd/Server.key: Öffnen /etc/kubernetes/pki/etcd/Server.crt: Keine solche Datei oder Verzeichnis	Der Pod Telegraf ReplicaSet soll auf einem Knoten ausgeführt werden, der als Master oder für etc bestimmt ist. Wenn der ReplicaSet-Pod auf einem dieser Knoten nicht ausgeführt wird, werden diese Fehler angezeigt. Überprüfen Sie, ob Ihre Master/etcd-Knoten eine Tönungswalle haben. Fügen Sie in diesem Fall die erforderlichen Verträgungen in das Telegraf ReplicaSet, telegraf-rs ein. Bearbeiten Sie zum Beispiel die Datei ReplicaSet… kubectl edit rs telegraf-rs …und fügen Sie die entsprechenden Verträgungen der Spezifikation hinzu. Starten Sie anschließend den Pod ReplicaSet neu.
Ich habe eine PSP/PSA Umgebung. Hat dies Auswirkungen auf meinen Überwachungsperator?	Wenn Ihr Kubernetes-Cluster mit Pod-Sicherheitsrichtlinie (PSP) oder Pod Security Admission (PSA) ausgeführt wird, müssen Sie ein Upgrade auf den aktuellen Kubernetes Monitoring Operator durchführen. Gehen Sie wie folgt vor, um auf den aktuellen Operator mit Unterstützung für PSP/PSA zu aktualisieren: 1. Deinstallieren Der bisherige Monitoring-Operator: Kubectl delete Agent-Monitoring-NetApp -n NetApp-Monitoring kubectl delete ns NetApp-Monitoring kubectl delete crd Agents.Monitoring.NetApp.com kubectl delete clusterrole Agent-Manager-role Agent-Proxy-role Agent-metrics-reader kubectl delete clusterrolebinding Agent-Manager-rolebinding Agent-Proxy-rolebinding Agent-rolebinding Agent-Cluster-admin-rolebinding 2. Installieren Die neueste Version des Überwachungsbedieners.
Ich habe Probleme beim Versuch, den Operator bereitzustellen, und ich habe PSP/PSA in Gebrauch.	1. Bearbeiten Sie den Agenten mit folgendem Befehl: Kubectl -n <name-space> edit Agent 2. Markieren Sie „Sicherheitspolitik aktiviert“ als „falsch“. Dadurch werden Pod-Sicherheitsrichtlinien und Pod-Sicherheitszulassung deaktiviert und der Bediener kann die Bereitstellung durchführen. Bestätigung mit den folgenden Befehlen: Kubectl get psp (sollte Pod Security Policy entfernt zeigen) kubectl get all -n <Namespace> grep -i psp (sollte zeigen, dass nichts gefunden wird)

Problem:	Versuchen Sie dies:
„ImagePullBackoff“-Fehler erkannt	Diese Fehler können auftreten, wenn Sie über ein benutzerdefiniertes oder privates Docker-Repository verfügen und den Kubernetes Monitoring Operator noch nicht so konfiguriert haben, dass er es richtig erkennt. Weitere Informationen Info über die Konfiguration für benutzerdefinierte/private Repo.

Ich habe ein Problem mit der Installation meines Monitoring-Bedieners, und die aktuelle Dokumentation hilft mir nicht, es zu lösen.	Erfassen oder notieren Sie die Ausgabe der folgenden Befehle, und wenden Sie sich an den technischen Support.
	`kubectl -n netapp-monitoring get all`
	`kubectl -n netapp-monitoring describe all`
	`kubectl -n netapp-monitoring logs <monitoring-operator-pod> --all-containers=true`
	`kubectl -n netapp-monitoring logs <telegraf-pod> --all-containers=true`

NET-Observer (Workload Map)-Pods im Operator Namespace befinden sich in CrashLoopBackOff	Diese Pods entsprechen dem Workload Map-Datensammler für Network Observability. Versuchen Sie Folgendes: • Überprüfen Sie die Protokolle eines der Pods, um die minimale Kernel-Version zu bestätigen. Beispiel: --- {"CI-Tenant-id":"your-Tenant-id","Collector-Cluster":"your-k8s-Cluster-Name","Environment":"prod","Level":"error","msg":"failed in validation. Grund: Kernel-Version 3.10.0 ist kleiner als die minimale Kernel-Version von 4.18.0","Time":"2022-11-09T08:23:08Z"} ---- • Net-Observer-Pods erfordern die Linux-Kernel-Version mindestens 4.18.0. Überprüfen Sie die Kernel-Version mit dem Befehl „uname -r“ und stellen Sie sicher, dass sie >= 4.18.0 sind
Pods werden im Operator Namespace ausgeführt (Standard: netapp-Monitoring), es werden jedoch keine Daten in der UI für die Workload-Zuordnung oder Kubernetes-Metriken in Abfragen angezeigt	Überprüfen Sie die Zeiteinstellung auf den Knoten des K8S-Clusters. Für eine genaue Prüfung und Datenberichterstattung wird dringend empfohlen, die Zeit auf dem Agent-Rechner mit Network Time Protocol (NTP) oder Simple Network Time Protocol (SNTP) zu synchronisieren.

Problem:	Versuchen Sie dies:
Einige der Net-Observer-Pods im Namespace Operator befinden sich im Status „Ausstehend“	NET-Observer ist ein DemonSet und führt in jedem Knoten des K8s-Clusters einen Pod aus. • Beachten Sie den Pod, der sich im Status „Ausstehend“ befindet, und prüfen Sie, ob ein Ressourcenproblem für CPU oder Speicher vorliegt. Stellen Sie sicher, dass der erforderliche Arbeitsspeicher und die erforderliche CPU im Knoten verfügbar sind.
Ich sehe Folgendes in meinen Protokollen sofort nach der Installation des Kubernetes Monitoring Operators: [inputs.prometheus] Fehler im Plugin: Fehler beim Erstellen einer HTTP-Anforderung an http://kube-state-metrics.<namespace>.svc.Cluster.local:8080/metrics: Get http://kube-state-metrics.<namespace>.svc.Cluster.local:8080/metrics: Dial tcp: Lookup kube-State-metrics.<namespace>.svc.Cluster.local: Kein solcher Host	Diese Meldung wird normalerweise nur angezeigt, wenn ein neuer Operator installiert ist und der Pod „*telegraf-rs*“ vor dem Einschalten des Pod „*ksm*“ steht. Diese Meldungen sollten beendet werden, sobald alle Pods ausgeführt werden.
Ich sehe keine Kennzahlen für die Kubernetes-Kronjobs, die in meinem Cluster vorhanden sind, erfasst.	Überprüfen Sie Ihre Kubernetes-Version (d. h. `kubectl version`). Wenn es v1.20.x oder niedriger ist, ist dies eine erwartete Einschränkung. Die mit dem Kubernetes Monitoring Operator implementierte Version von kube-State-Metrics unterstützt nur v1.cronjob. Bei Kubernetes 1.20.x und niedriger befindet sich die Ressource cronjob unter v1beta.cronjob. Daher können kube-State-Metriken die Ressource cronjob nicht finden.
Nach der Installation des Bedieners geben die telegraf-ds-Pods CrashLoopBackOff ein und die POD-Protokolle zeigen „su: Authentication failure“ an.	Bearbeiten Sie den Abschnitt telegraf in *AgentConfiguration*, und setzen Sie *dockerMetricCollectionEnabled* auf false. Weitere Informationen finden Sie im "Konfigurationsoptionen". … Spec: … telegraf: … - Name: docker Run-Mode: - DemonSet Ersetzungen: - Schlüssel: DOCKER_UNIX_SOCK_PLACEHOLDER Wert: unix:///run/Docker.Sock … …
Ich sehe wiederholte Fehlermeldungen wie die folgenden in meinen Telegraf-Logs: E! [Agent] Fehler beim Schreiben in Outputs.http: Post "https://<tenant_url>/Rest/v1/Lake/ingest/influxdb": Kontext-Deadline überschritten (Client. Zeitüberschreitung beim Warten auf Header überschritten)	Bearbeiten Sie den Abschnitt telegraf in *AgentConfiguration*, und erhöhen Sie *outputTimeout* auf 10s. Weitere Informationen finden Sie im "Konfigurationsoptionen".
Ich vermisse *involvedobject* Daten für einige Event Logs.	Stellen Sie sicher, dass Sie die Schritte im Abschnitt oben befolgt haben"Berechtigungen".

Problem:	Versuchen Sie dies:
Wieso werden zwei Monitoring Operator Pods ausgeführt, einer mit dem Namen netapp-CI-Monitoring-Operator-<pod> und der andere mit dem Namen Monitoring-Operator-<pod>?	Seit dem 12. Oktober 2023 hat Data Infrastructure Insights den Betreiber refaktorisiert, um unseren Benutzern besser dienen zu können. Damit diese Änderungen vollständig umgesetzt werden, müssen Sie Entfernen Sie den alten Bedienerund Installieren Sie den neuen.
Meine kubernetes-Ereignisse haben unerwartet aufgehört, Daten bei Infrastruktur-Insights zu melden.	Rufen Sie den Namen des POD für den Event-Exporter ab: `kubectl -n netapp-monitoring get pods`
grep event-exporter	awk '{print $1}'
sed 's/event-exporter./event-exporter/'` Es sollte entweder „netapp-CI-Event-Exporteur“ oder „Event-Exporteur“ sein. Bearbeiten Sie anschließend den Überwachungsagenten `kubectl -n netapp-monitoring edit agent`und legen Sie den Wert für LOG_FILE so fest, dass der entsprechende POD-Name des Ereignisexporteurs im vorherigen Schritt angezeigt wird. Genauer gesagt sollte LOG_FILE auf "/var/log/Containers/netapp-CI-Event-exporteur.log" oder "/var/log/Containers/Event-exporteur*.log" gesetzt werden …. fluent-bit: … - name: event-exporter-ci substitutions: - key: LOG_FILE values: - /var/log/containers/netapp-ci-event-exporter*.log … …. Alternativ kann man auch Deinstallieren und Neu installieren den Agenten.	Ich sehe POD(s), die vom Kubernetes-Monitoring-Operator bereitgestellt werden, aufgrund unzureichender Ressourcen.
Informationen zum Erhöhen der CPU- und/oder Speichergrenzen finden Sie im Kubernetes Monitoring Operator"Konfigurationsoptionen".	Durch ein fehlendes Image oder eine ungültige Konfiguration wurden die netapp-CI-kube-State-metrics Pods nicht gestartet oder nicht einsatzbereit gemacht. Jetzt bleibt StatefulSet stecken und Konfigurationsänderungen werden nicht auf die Pods mit den netapp-CI-kube-State-Metriken angewendet.

Problem:	Versuchen Sie dies:
StatefulSet befindet sich in einem "Defekt" Status. Nachdem Sie Konfigurationsprobleme behoben haben, springen die netapp-CI-kube-State-metrics-Pods an.	Pods mit netapp-CI-kube-Status-Metriken können nicht gestartet werden, nachdem ein Kubernetes Operator Upgrade ausgeführt wurde. Es wird ErrImagePull geworfen (es konnte nicht das Image entfernt werden).
Versuchen Sie, die Pods manuell zurückzusetzen.	„Event disorded as being alder then maxEventAgeSeconds“ Meldungen werden für meinen Kubernetes Cluster unter Log Analysis beobachtet.
Ändern Sie den Operator *agentkonfiguration*, und erhöhen Sie die Erweiterung *Event-exporteur-maxEventAgeSeconds* (d. h. auf 60s), *Event-exporteur-kubeQPS* (d. h. auf 100) und *Event-exporteur-kubeBurst* (d. h. auf 500). Weitere Informationen zu diesen Konfigurationsoptionen finden Sie auf der "Konfigurationsoptionen" Seite.	Telegraf warnt vor unzureichenden, abschließbaren Speichern oder stürzt ab.
Versuchen Sie, die Grenze des abschließbaren Speichers für Telegraf im zugrunde liegenden Betriebssystem/Knoten zu erhöhen. Wenn eine Erhöhung des Limits keine Option ist, ändern Sie die NKMO-Agentkonfiguration und setzen Sie *Unprotected* auf *true*. Dadurch wird Telegraf angewiesen, keine gesperrten Speicherseiten zu reservieren. Dies kann zwar ein Sicherheitsrisiko darstellen, da entschlüsselte Geheimnisse möglicherweise auf die Festplatte ausgetauscht werden, ermöglicht aber die Ausführung in Umgebungen, in denen das Reservieren von gesperrtem Speicher nicht möglich ist. Weitere Informationen zu den Konfigurationsoptionen *Unprotected* finden Sie auf der "Konfigurationsoptionen" Seite.	Ich sehe Warnhinweise von Telegraf wie folgt: *W! [Inputs.diskio] der Datenträgername für „vdc“ kann nicht erfasst werden: Fehler beim Lesen von /dev/vdc: Keine Datei oder Verzeichnis*
Für den Kubernetes Monitoring Operator sind diese Warnmeldungen gutartig und können sicher ignoriert werden. Alternativ können Sie den telegraf-Abschnitt in AgentConfiguration bearbeiten und *runDsPrivileged* auf true setzen. Weitere Informationen finden Sie im "Konfigurationsoptionen des Bedieners".	Mein Fluent-Bit-Pod schlägt mit den folgenden Fehlern fehl: [2024/10/16 14:16:23] [error] [/src/Fluent-Bit/Plugins/in_tail/tail_fs_inotify.c:360 errno=10/16 14] zu viele geöffnete Dateien [16/23:16:23] [error] initialisieren des Input tail.0 [2024/24:2024:10/16 14] [error] die Eingabe-Initialisierung ist fehlgeschlagen

Weitere Informationen finden Sie auf der "Support" Seite oder im "Data Collector Supportmatrix".

Memcached Data Collector

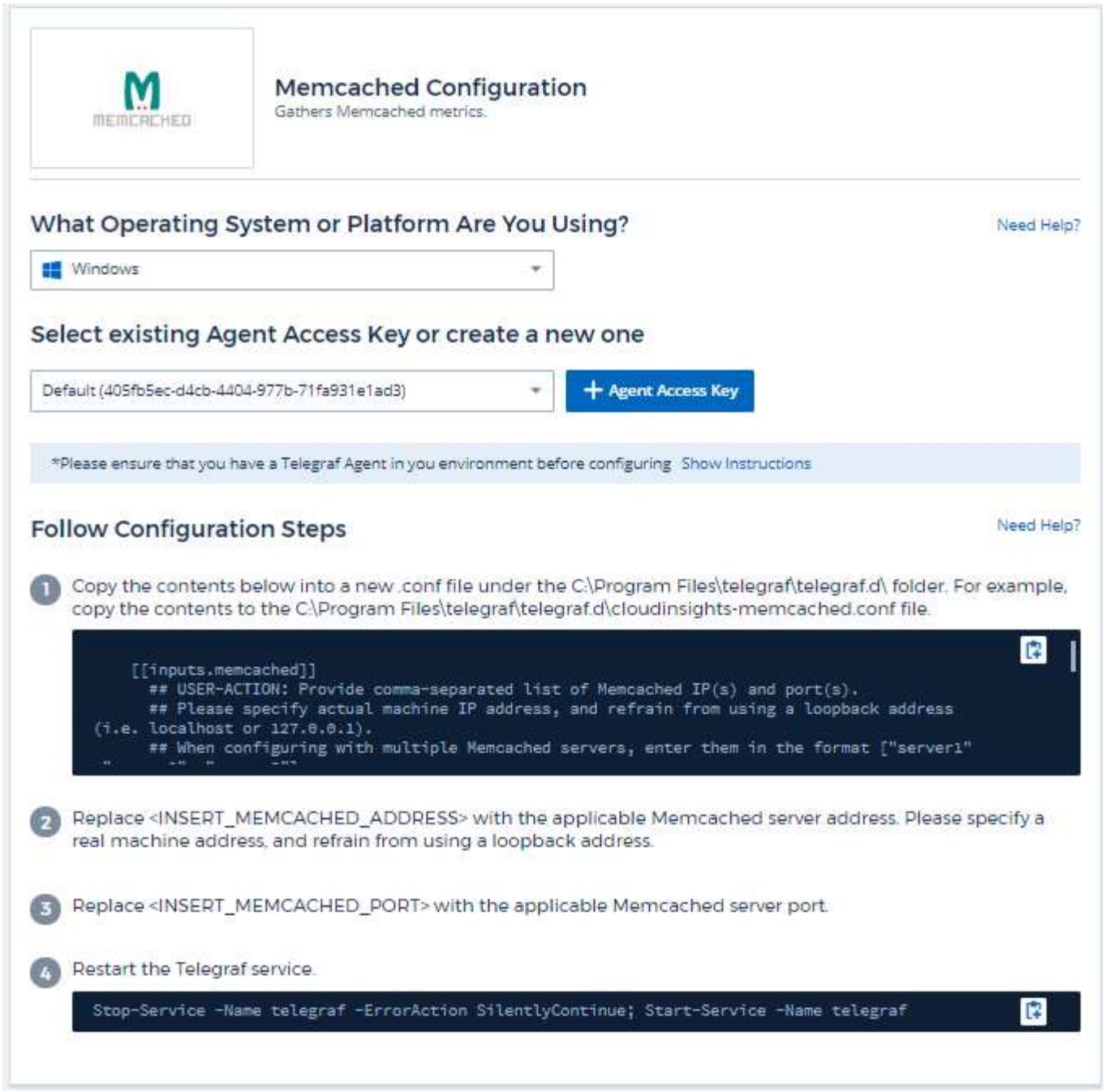
Data Infrastructure Insights nutzt diesen Datensammler, um Kennzahlen aus Memcached zu sammeln.

Installation

1. Klicken Sie unter **Observability > Collectors** auf **+Data Collector**. Wählen Sie Memcached.

   Wählen Sie das Betriebssystem oder die Plattform aus, auf dem der Telegraf-Agent installiert ist.

2. Wenn Sie noch keinen Agenten zur Sammlung installiert haben oder einen Agenten für ein anderes Betriebssystem oder eine andere Plattform installieren möchten, klicken Sie auf *Anweisungen anzeigen*, um die Anweisungen zu erweitern"Agenten-Installation".
3. Wählen Sie den Agent-Zugriffsschlüssel für diesen Datensammler aus. Sie können einen neuen Agent-Zugriffsschlüssel hinzufügen, indem Sie auf die Schaltfläche **+ Agent Access Key** klicken. Best Practice: Verwenden Sie einen anderen Agent-Zugriffsschlüssel nur, wenn Sie Datensammler gruppieren möchten, zum Beispiel nach Betriebssystem/Plattform.
4. Befolgen Sie die Konfigurationsschritte, um den Datensammler zu konfigurieren. Die Anweisungen hängen vom Betriebssystem oder der Plattform ab, die Sie zur Datenerfassung verwenden.



Einrichtung

Informationen finden Sie im "Wiki mit Memcached".

Objekte und Zähler

Folgende Objekte und ihre Zähler werden gesammelt:

Objekt:	Kennungen:	Attribute:	Datenpunkte:
Gememcachte	Namespace-Server	Node-IP-Node-Name	Akzeptieren von Verbindungen verarbeitet Authentifizierungsanforderungen fehlgeschlagene Authentifizierungen verwendete Bytes (pro Sekunde) geschriebene Bytes (pro Sek.) CAS Badval CAS Hits CAS Misses Flush Reqs (pro Sek.) Get Reqs (pro Sek.) Set Reqs (pro Sek.) Touch Reqs (pro Sek.) Verbindungserträge (pro Sek.) Verbindungsstrukturen Verbindungen öffnen Aktuelle gespeicherte Objekte Decr fordert Zugriffe (pro Sek.) Decr fordert Fehlschläge (pro Sek.) Löschen von Anfragen Treffer (pro Sek.) Löschen von Anfragen Fehlschläge (pro Sek.) entfernte Objekte gültige Abtreibungen abgelaufene Objekte Get Hits (pro Sek.) Get Misses (pro Sek.) Gebrauchte Hash Bytes Hash-Bytes erweitert Hash Power Level Inc. Hash Power Level Inc. Zugriffe (pro Sek.) Infr Anfragen Misses (pro Sek.) Server Max Bytes anhören deaktiviert Num zurückgewonnener Mitarbeiter Threads Anzahl geöffnete Verbindungen Gesamtzahl der gespeicherten Elemente Touch Hits Touch Misses Server Uptime

Fehlerbehebung

Weitere Informationen finden Sie auf der "Support" Seite.

MongoDB Data Collector

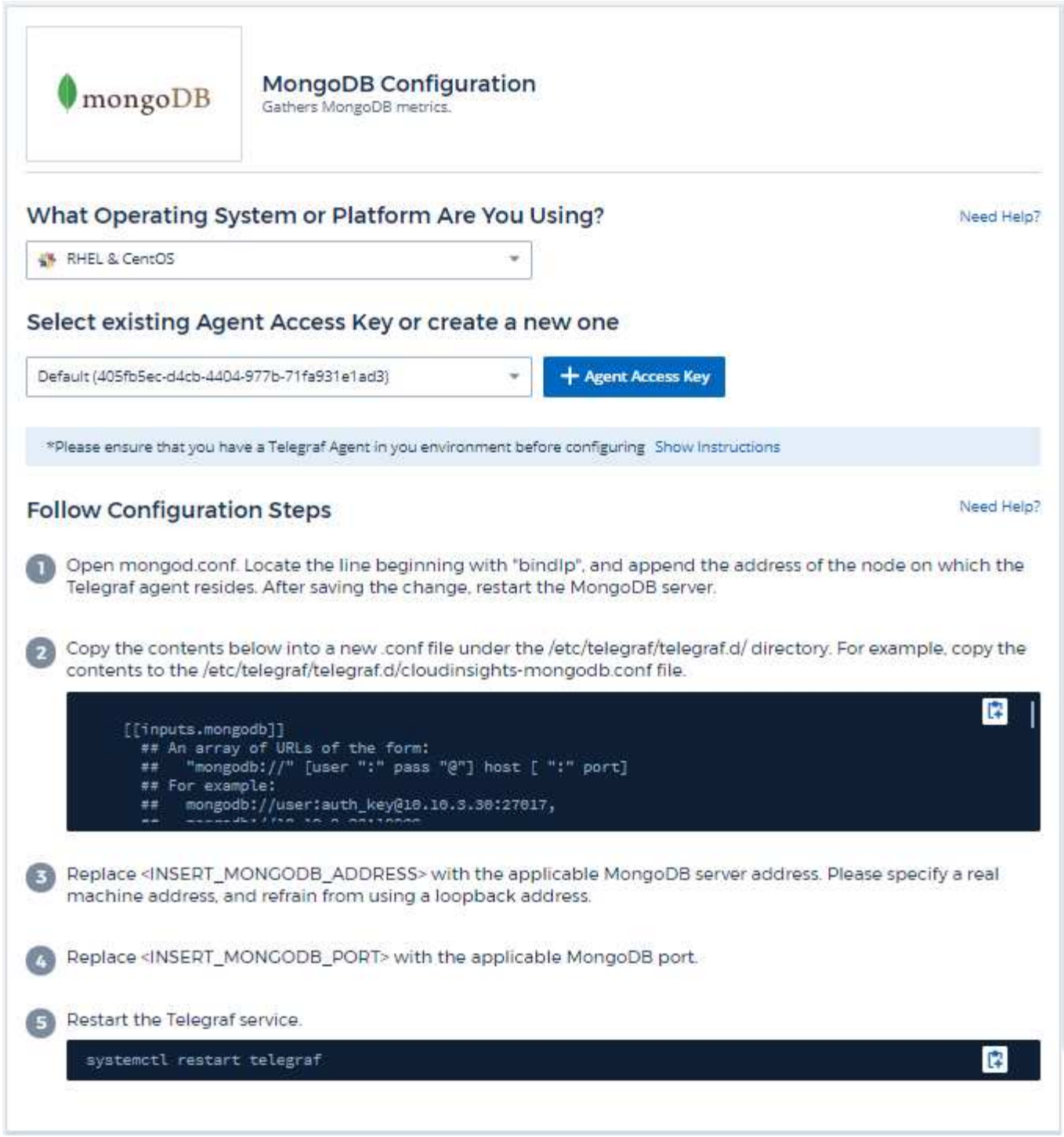
Data Infrastructure Insights verwendet diesen Datensammler, um Kennzahlen von MongoDB zu sammeln.

Installation

1. Klicken Sie unter **Observability > Collectors** auf **+Data Collector**. Wählen Sie MongoDB.

   Wählen Sie das Betriebssystem oder die Plattform aus, auf dem der Telegraf-Agent installiert ist.

2. Wenn Sie noch keinen Agenten zur Sammlung installiert haben oder einen Agenten für ein anderes Betriebssystem oder eine andere Plattform installieren möchten, klicken Sie auf *Anweisungen anzeigen*, um die Anweisungen zu erweitern"Agenten-Installation".
3. Wählen Sie den Agent-Zugriffsschlüssel für diesen Datensammler aus. Sie können einen neuen Agent-Zugriffsschlüssel hinzufügen, indem Sie auf die Schaltfläche **+ Agent Access Key** klicken. Best Practice: Verwenden Sie einen anderen Agent-Zugriffsschlüssel nur, wenn Sie Datensammler gruppieren möchten, zum Beispiel nach Betriebssystem/Plattform.
4. Befolgen Sie die Konfigurationsschritte, um den Datensammler zu konfigurieren. Die Anweisungen hängen vom Betriebssystem oder der Plattform ab, die Sie zur Datenerfassung verwenden.



Einrichtung

Informationen finden Sie im "MongoDB Dokumentation".

Objekte und Zähler

Folgende Objekte und ihre Zähler werden gesammelt:

Objekt:	Kennungen:	Attribute:	Datenpunkte:
MongoDB	Namespace-Hostname		

Objekt:	Kennungen:	Attribute:	Datenpunkte:
MongoDB Datenbank	Name der Namespace-Hostname-Datenbank		

Fehlerbehebung

Informationen finden Sie auf der "Support" Seite.

MySQL Data Collector

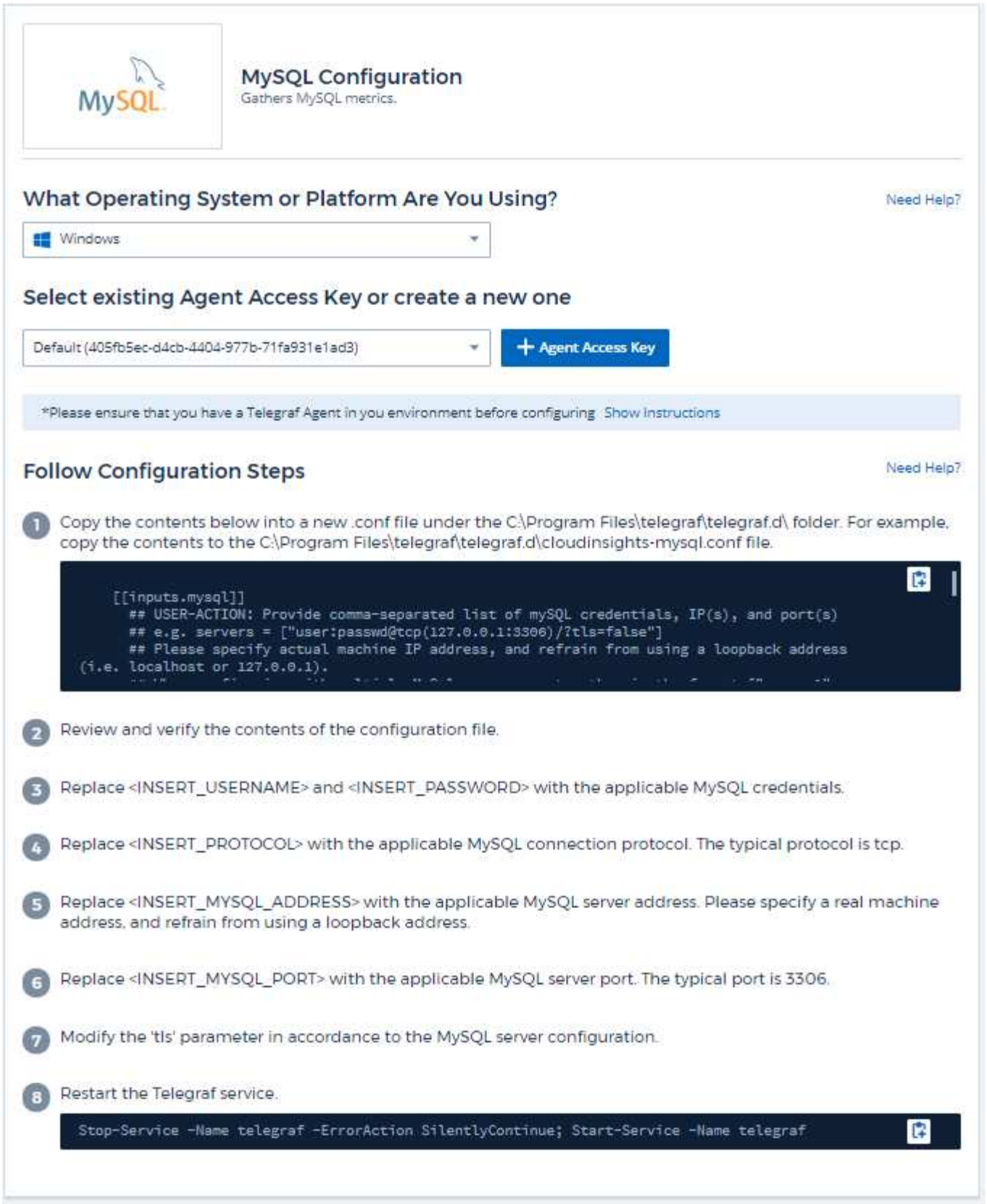
Data Infrastructure Insights verwendet diesen Datensammler, um Kennzahlen aus MySQL zu sammeln.

Installation

1. Klicken Sie unter **Observability > Collectors** auf **+Data Collector**. Wählen Sie MySQL.

   Wählen Sie das Betriebssystem oder die Plattform aus, auf dem der Telegraf-Agent installiert ist.

2. Wenn Sie noch keinen Agenten zur Sammlung installiert haben oder einen Agenten für ein anderes Betriebssystem oder eine andere Plattform installieren möchten, klicken Sie auf *Anweisungen anzeigen*, um die Anweisungen zu erweitern"Agenten-Installation".
3. Wählen Sie den Agent-Zugriffsschlüssel für diesen Datensammler aus. Sie können einen neuen Agent-Zugriffsschlüssel hinzufügen, indem Sie auf die Schaltfläche **+ Agent Access Key** klicken. Best Practice: Verwenden Sie einen anderen Agent-Zugriffsschlüssel nur, wenn Sie Datensammler gruppieren möchten, zum Beispiel nach Betriebssystem/Plattform.
4. Befolgen Sie die Konfigurationsschritte, um den Datensammler zu konfigurieren. Die Anweisungen hängen vom Betriebssystem oder der Plattform ab, die Sie zur Datenerfassung verwenden.



Einrichtung

Informationen finden Sie im "MySQL-Dokumentation".

Objekte und Zähler

Folgende Objekte und ihre Zähler werden gesammelt:

Objekt:	Kennungen:	Attribute:	Datenpunkte:
MySQL	Namespace für MySQL Server	Node-IP-Node-Name	Abgebrochene Clients (pro s) abgebrochene Verbindungen (pro s) RX Byte (pro s) TX Bytes (pro Sek.) Befehle Admin (pro Sek.) Befehle Alter Ereignisbefehle Alter Funktion Befehle Alter Instanz Befehle Alter Prozedur Befehle Alter Server Befehle Alter Tabelle Befehle Alter Tablespace Befehle Alter Benutzer Befehle Analyse Befehle Zuweisen zu Keycache-Befehlen Begin-Befehle Binlog-Befehle Aufruf Procedure-Befehle DB-Befehle Change Master befiehlt Change Repl Filter Befehle Check Commands Prüfsummenbefehle Befehle Commit-Befehle DB-Befehle erstellen Ereignisbefehle erstellen Befehle erstellen Index-Befehle erstellen Maßnahmen-Befehle erstellen Serverbefehle erstellen Trigger-Befehle erstellen UDF-Befehle erstellen Benutzerbefehle erstellen Befehle anzeigen erstellen Dealloc SQL-Verbindungsfehler akzeptieren erstellte tmp-Disk-Tabellen verzögerte Fehler Flush-Befehle Handler Commit Innodb Buffer Pool Bytes Daten Schlüsselblöcke Nicht Gespült Schlüssel Leseanforderungen Schlüssel Schreib Schlüssel Schreibvorgänge Max Ausführungszeit Überschritten Max Verwendete Verbindungen Open Files Performance Schema Konten Lost Prepared Stmt Count Qcache Freie Blöcke

Fehlerbehebung

Weitere Informationen finden Sie auf der "Support" Seite.

Netstat Data Collector

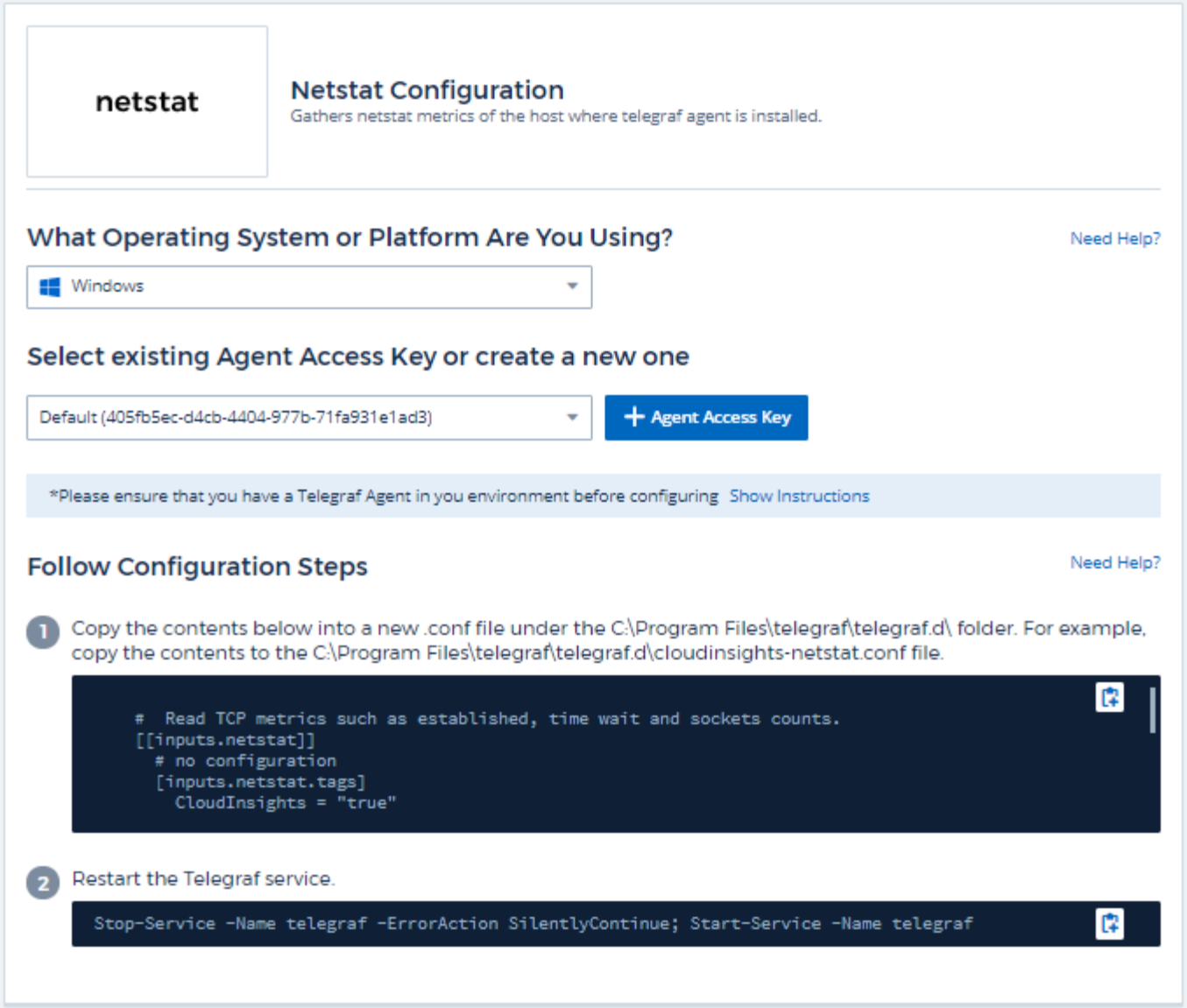
Data Infrastructure Insights verwendet diesen Datensammler, um netstat-Metriken zu erfassen.

Installation

1. Klicken Sie unter **Observability > Collectors** auf **+Data Collector**. Wählen Sie Netstat.

   Wählen Sie das Betriebssystem oder die Plattform aus, auf dem der Telegraf-Agent installiert ist.

2. Wenn Sie noch keinen Agenten zur Sammlung installiert haben oder einen Agenten für ein anderes Betriebssystem oder eine andere Plattform installieren möchten, klicken Sie auf *Anweisungen anzeigen*, um die Anweisungen zu erweitern"Agenten-Installation".
3. Wählen Sie den Agent-Zugriffsschlüssel für diesen Datensammler aus. Sie können einen neuen Agent-Zugriffsschlüssel hinzufügen, indem Sie auf die Schaltfläche **+ Agent Access Key** klicken. Best Practice: Verwenden Sie einen anderen Agent-Zugriffsschlüssel nur, wenn Sie Datensammler gruppieren möchten, zum Beispiel nach Betriebssystem/Plattform.
4. Befolgen Sie die Konfigurationsschritte, um den Datensammler zu konfigurieren. Die Anweisungen hängen vom Betriebssystem oder der Plattform ab, die Sie zur Datenerfassung verwenden.



Einrichtung

Objekte und Zähler

Folgende Objekte und ihre Zähler werden gesammelt:

Objekt:	Kennungen:	Attribute:	Datenpunkte:
Netstat	Node-UUID	Node-IP-Node-Name	

Fehlerbehebung

Weitere Informationen finden Sie auf der "Support" Seite.

Nginx Data Collector

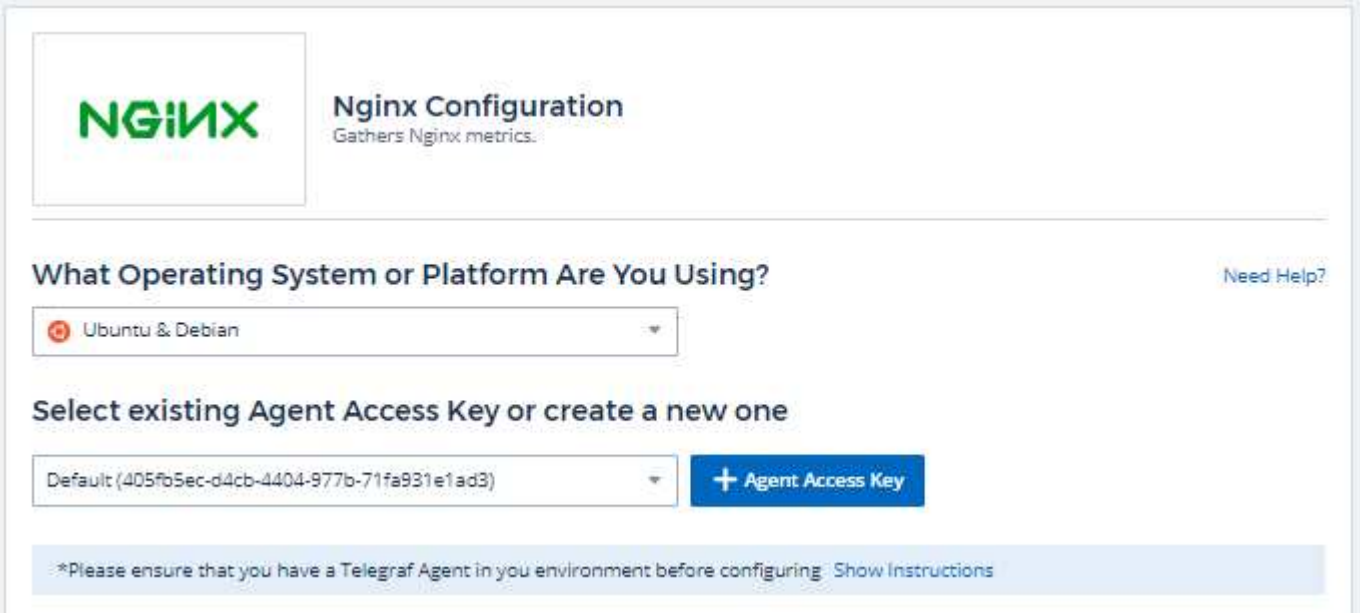
Data Infrastructure Insights verwendet diesen Datensammler, um Kennzahlen von Nginx zu sammeln.

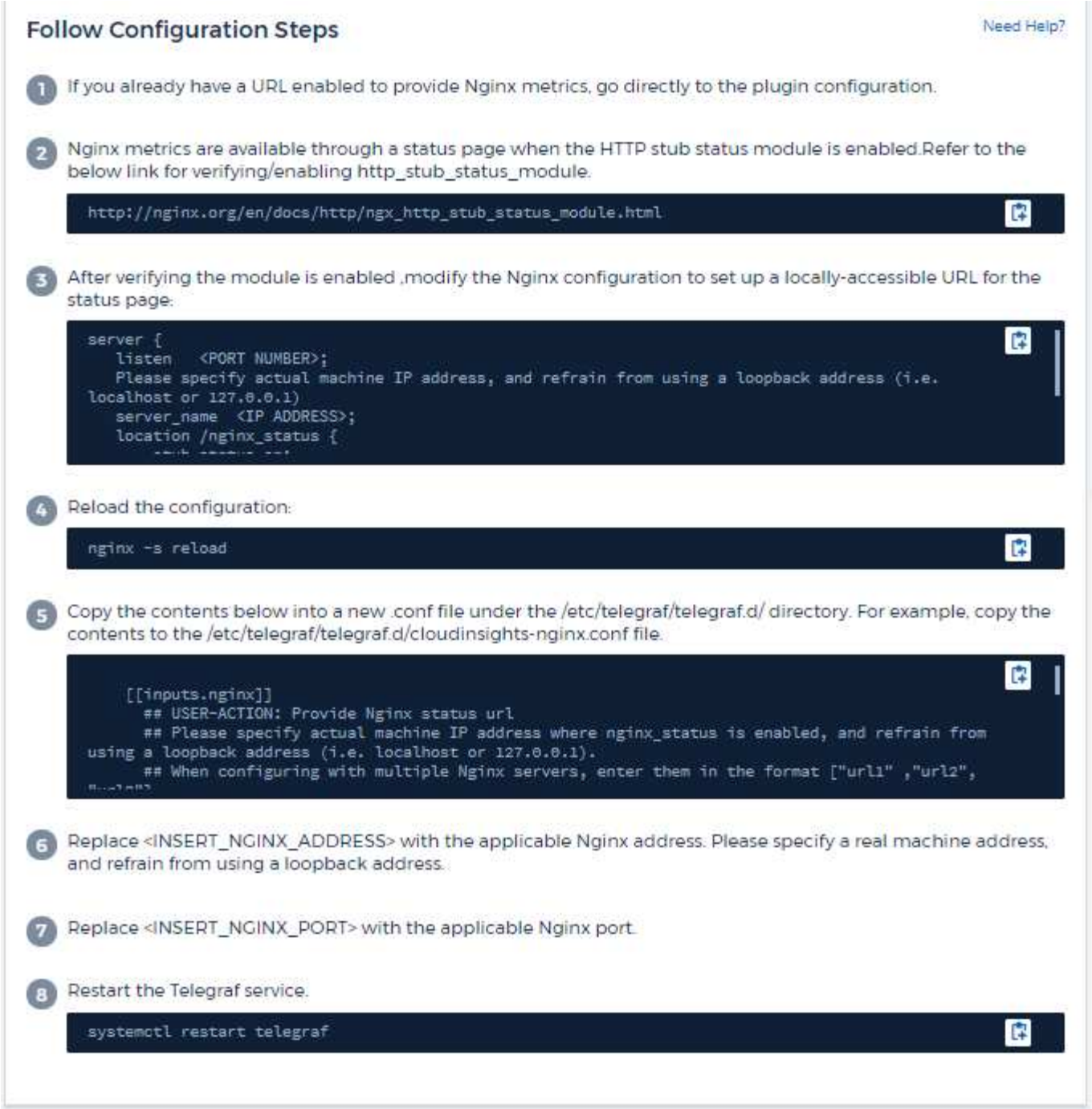
Installation

1. Klicken Sie unter **Observability > Collectors** auf **+Data Collector**. Wählen Sie Nginx.

   Wählen Sie das Betriebssystem oder die Plattform aus, auf dem der Telegraf-Agent installiert ist.

2. Wenn Sie noch keinen Agenten zur Sammlung installiert haben oder einen Agenten für ein anderes Betriebssystem oder eine andere Plattform installieren möchten, klicken Sie auf *Anweisungen anzeigen*, um die Anweisungen zu erweitern"Agenten-Installation".
3. Wählen Sie den Agent-Zugriffsschlüssel für diesen Datensammler aus. Sie können einen neuen Agent-Zugriffsschlüssel hinzufügen, indem Sie auf die Schaltfläche **+ Agent Access Key** klicken. Best Practice: Verwenden Sie einen anderen Agent-Zugriffsschlüssel nur, wenn Sie Datensammler gruppieren möchten, zum Beispiel nach Betriebssystem/Plattform.
4. Befolgen Sie die Konfigurationsschritte, um den Datensammler zu konfigurieren. Die Anweisungen hängen vom Betriebssystem oder der Plattform ab, die Sie zur Datenerfassung verwenden.





Einrichtung

Für die Nginx-Metrik muss Nginx "http_stub_Status_Module"aktiviert sein.

Weitere Informationen finden Sie im "Nginx-Dokumentation".

Objekte und Zähler

Folgende Objekte und ihre Zähler werden gesammelt:

Objekt:	Kennungen:	Attribute:	Datenpunkte:
Nginx	Namespace-Server	Node-IP-Node-Name-Port	Akzeptiert Aktive Bearbeitet Leseanforderungen, Die Auf Das Schreiben Warten

Fehlerbehebung

Weitere Informationen finden Sie auf der "Support" Seite.

PostgreSQL Data Collector

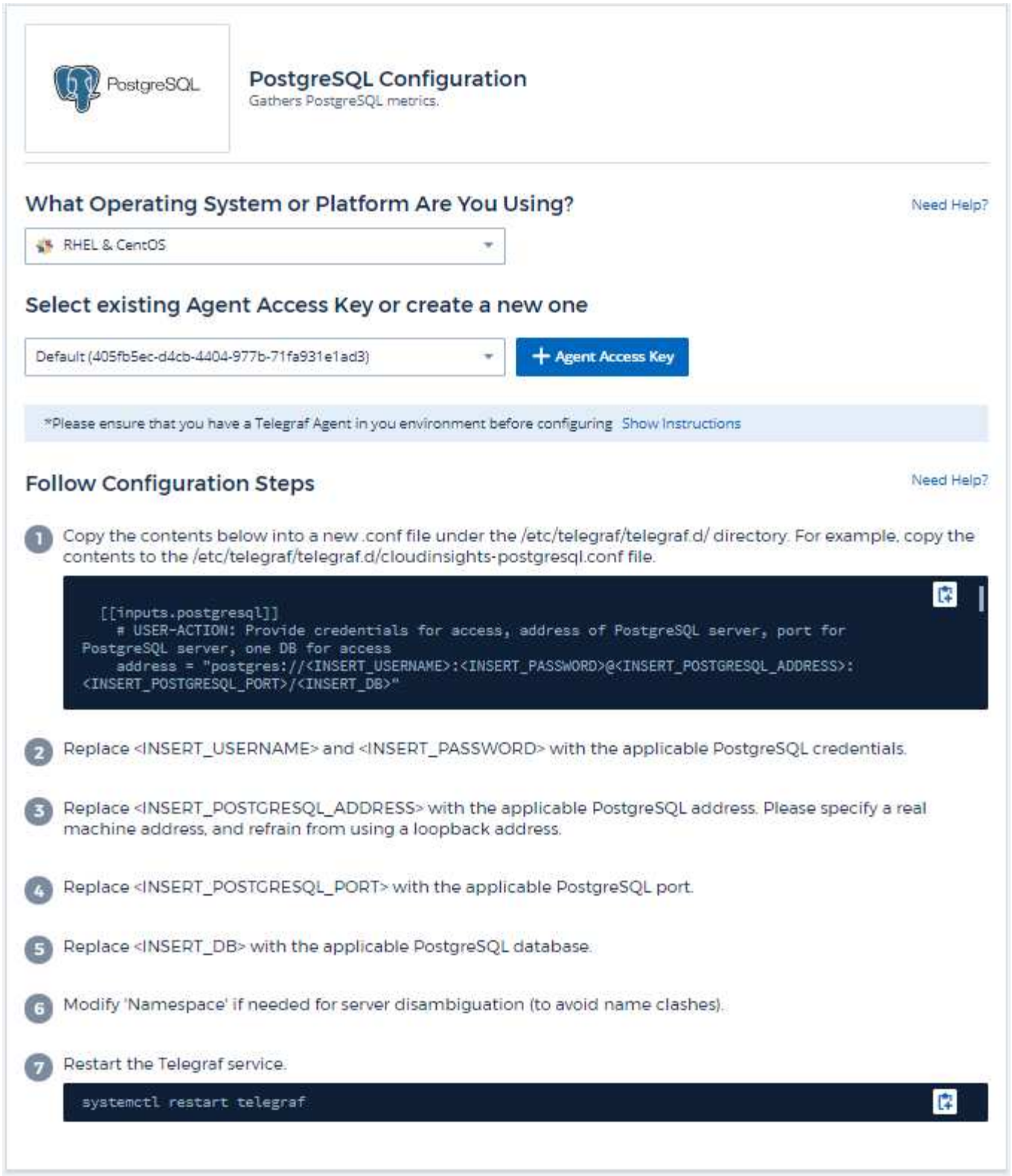
Data Infrastructure Insights nutzt diesen Datensammler, um Metriken aus PostgreSQL zu sammeln.

Installation

1. Klicken Sie unter **Observability > Collectors** auf **+Data Collector**. Wählen Sie PostgreSQL.

   Wählen Sie das Betriebssystem oder die Plattform aus, auf dem der Telegraf-Agent installiert ist.

2. Wenn Sie noch keinen Agenten zur Sammlung installiert haben oder einen Agenten für ein anderes Betriebssystem oder eine andere Plattform installieren möchten, klicken Sie auf *Anweisungen anzeigen*, um die Anweisungen zu erweitern"Agenten-Installation".
3. Wählen Sie den Agent-Zugriffsschlüssel für diesen Datensammler aus. Sie können einen neuen Agent-Zugriffsschlüssel hinzufügen, indem Sie auf die Schaltfläche **+ Agent Access Key** klicken. Best Practice: Verwenden Sie einen anderen Agent-Zugriffsschlüssel nur, wenn Sie Datensammler gruppieren möchten, zum Beispiel nach Betriebssystem/Plattform.
4. Befolgen Sie die Konfigurationsschritte, um den Datensammler zu konfigurieren. Die Anweisungen hängen vom Betriebssystem oder der Plattform ab, die Sie zur Datenerfassung verwenden.



Einrichtung

Informationen finden Sie im "PostgreSQL-Dokumentation".

Objekte und Zähler

Folgende Objekte und ihre Zähler werden gesammelt:

Objekt:	Kennungen:	Attribute:	Datenpunkte:
PostgreSQL Server	Namespace-Datenbankserver	Node Name Node-IP	Puffer Zugeordnete Buffers Back-End-Puffer Dateisynchronisation Buffers Checkpoint Puffer Clean Checkpoints Sync Time Checkpoints Write Time Checkpoints Requests Checkpoints Timed Max Geschrieben Sauber
PostgreSQL Datenbank	Namespace-Datenbankserver	Datenbank OID Node Name Node IP	Blöcke Lesezeit Blöcke Write Time Blocks Treffer Blöcke Liest Konflikte Deadlocks Client-Nummer Temp-Dateien Bytes Temp-Dateien Anzahl Zeilen Gelöschte Zeilen Abgeholt Zeilen Zeilenanzahl Zeilenanzahl Zeilenanzahl Zeilenumeinfügen Letzte Transaktionen Letzte Transaktionen Übertragen Rollbacks

Fehlerbehebung

Weitere Informationen finden Sie auf der "Support" Seite.

Puppet Agent Data Collector

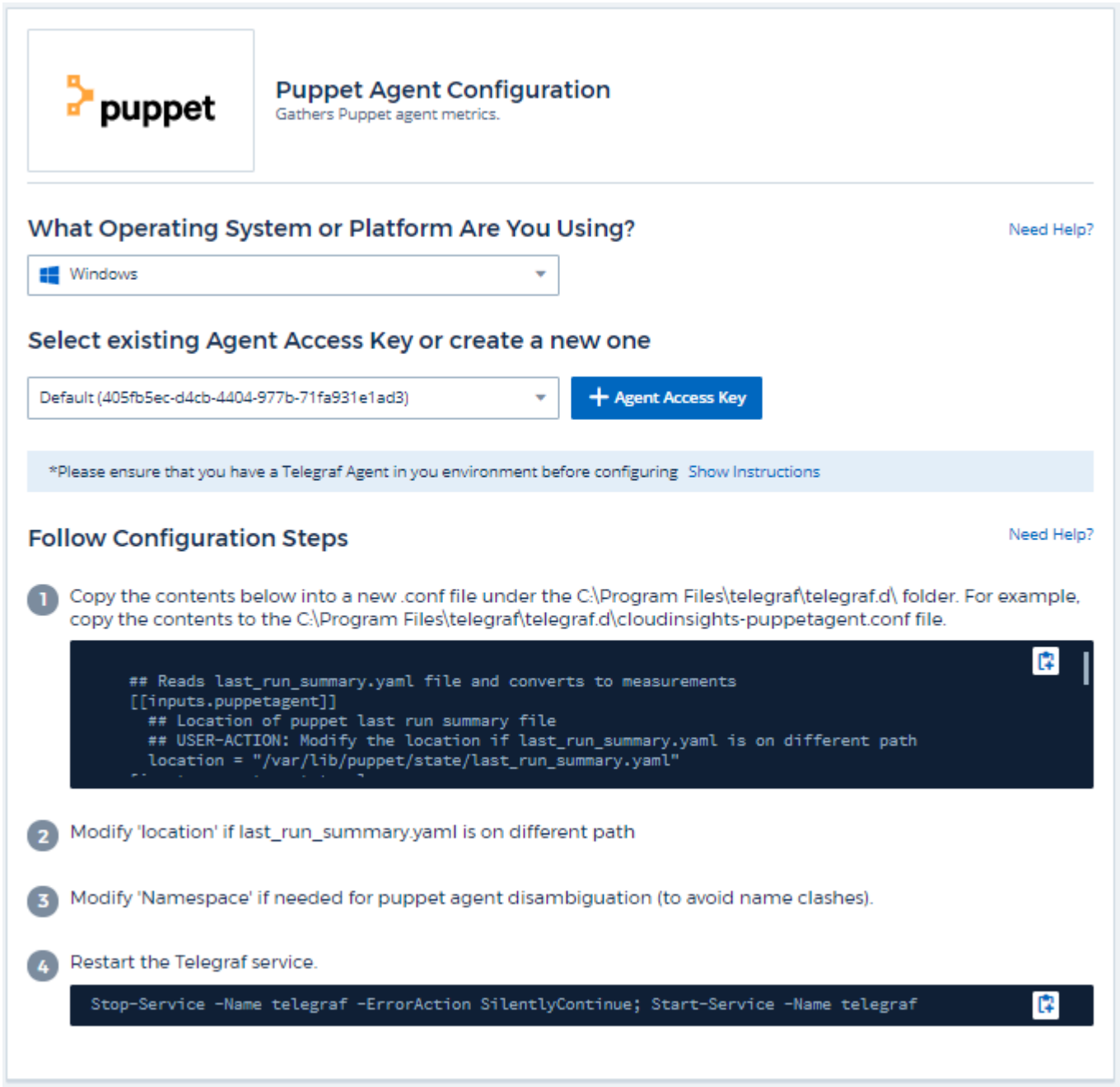
Data Infrastructure Insights nutzt diesen Datensammler, um Kennzahlen von Puppet Agent zu sammeln.

Installation

1. Klicken Sie unter **Observability > Collectors** auf **+Data Collector**. Wählen Sie Puppet.

   Wählen Sie das Betriebssystem oder die Plattform aus, auf dem der Telegraf-Agent installiert ist.

2. Wenn Sie noch keinen Agenten zur Sammlung installiert haben oder einen Agenten für ein anderes Betriebssystem oder eine andere Plattform installieren möchten, klicken Sie auf *Anweisungen anzeigen*, um die Anweisungen zu erweitern"Agenten-Installation".
3. Wählen Sie den Agent-Zugriffsschlüssel für diesen Datensammler aus. Sie können einen neuen Agent-Zugriffsschlüssel hinzufügen, indem Sie auf die Schaltfläche **+ Agent Access Key** klicken. Best Practice: Verwenden Sie einen anderen Agent-Zugriffsschlüssel nur, wenn Sie Datensammler gruppieren möchten, zum Beispiel nach Betriebssystem/Plattform.
4. Befolgen Sie die Konfigurationsschritte, um den Datensammler zu konfigurieren. Die Anweisungen hängen vom Betriebssystem oder der Plattform ab, die Sie zur Datenerfassung verwenden.



Einrichtung

Informationen finden Sie im "Puppet-Dokumentation"

Objekte und Zähler

Folgende Objekte und ihre Zähler werden gesammelt:

Objekt:	Kennungen:	Attribute:	Datenpunkte:

Puppet Agent	Namespace-Node-UUID	Node Name Ort Node-IP-Version Konfigstring Version Puppet	Änderungen Total Events Failure Ereignisse Success Events Summe Ressourcen Geänderte Ressourcen Fehlgeschlagen Ressourcen Konnten Nicht Neu Starten Ressourcen Outofsync Ressourcen Neustart Ressourcen Geplante Ressourcen Übersprungene Ressourcen Gesamtzeit Ankerzeit Abruf Configtime Cron Time Exec Time File Time Filebucket Time Lastrun Time Package Time Zeitplanzeit Service Time Sshauorizedkey Time Total Time User

Fehlerbehebung

Weitere Informationen finden Sie auf der "Support" Seite.

Redis Data Collector

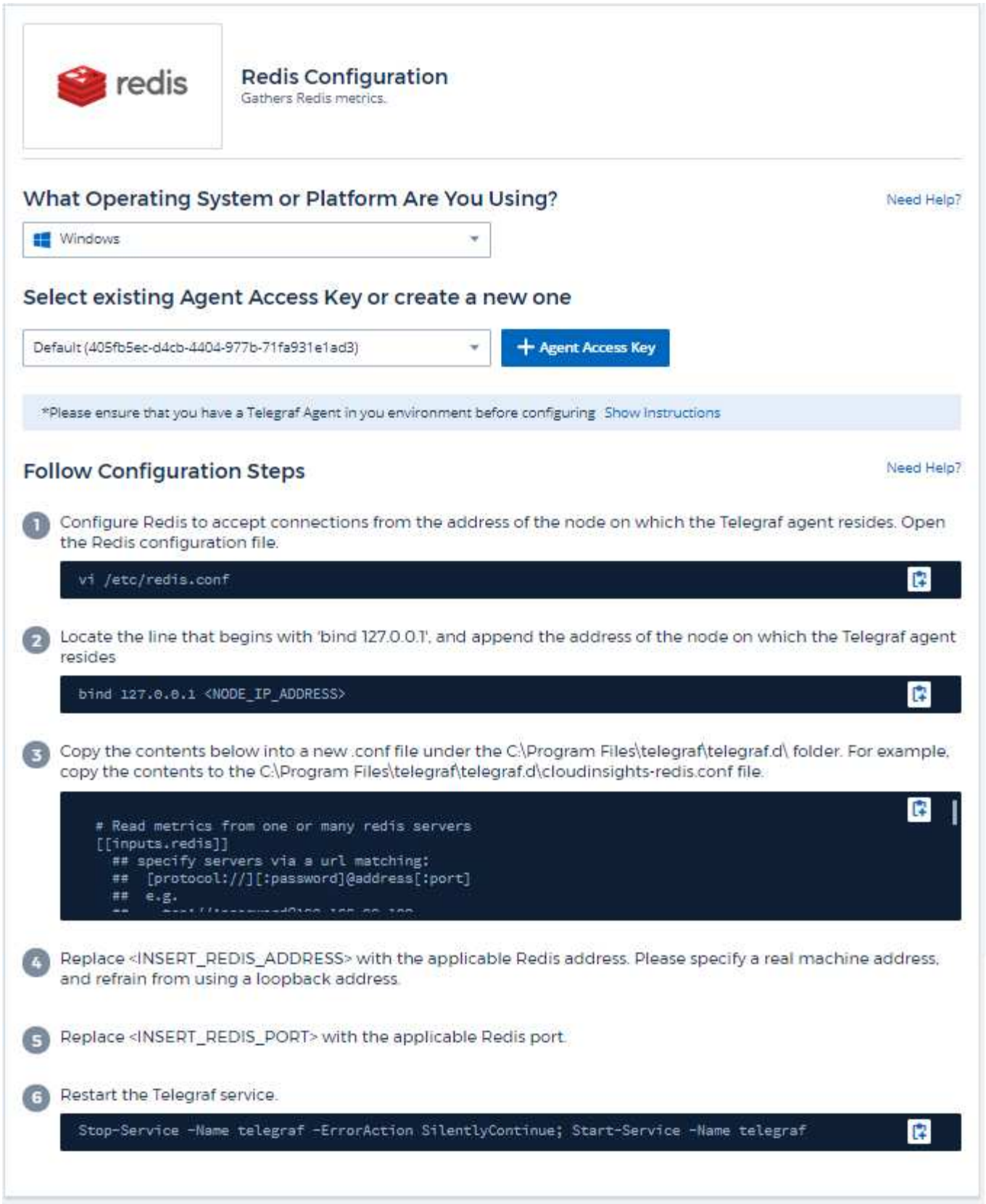
Data Infrastructure Insights nutzt diesen Datensammler, um Kennzahlen von Redis zu sammeln. Redis ist ein Open Source, in-Memory Data Structure Store, der als Datenbank-, Cache- und Nachrichten-Broker verwendet wird und die folgenden Datenstrukturen unterstützt: Strings, Hash-Funktionen, Listen, Sätze und mehr.

Installation

1. Klicken Sie unter **Observability > Collectors** auf **+Data Collector**. Wählen Sie „Redis“.

   Wählen Sie das Betriebssystem oder die Plattform aus, auf dem der Telegraf-Agent installiert ist.

2. Wenn Sie noch keinen Agenten zur Sammlung installiert haben oder einen Agenten für ein anderes Betriebssystem oder eine andere Plattform installieren möchten, klicken Sie auf *Anweisungen anzeigen*, um die Anweisungen zu erweitern"Agenten-Installation".
3. Wählen Sie den Agent-Zugriffsschlüssel für diesen Datensammler aus. Sie können einen neuen Agent-Zugriffsschlüssel hinzufügen, indem Sie auf die Schaltfläche **+ Agent Access Key** klicken. Best Practice: Verwenden Sie einen anderen Agent-Zugriffsschlüssel nur, wenn Sie Datensammler gruppieren möchten, zum Beispiel nach Betriebssystem/Plattform.
4. Befolgen Sie die Konfigurationsschritte, um den Datensammler zu konfigurieren. Die Anweisungen hängen vom Betriebssystem oder der Plattform ab, die Sie zur Datenerfassung verwenden.



Einrichtung

Informationen finden Sie im "Redis-Dokumentation".

Objekte und Zähler

Folgende Objekte und ihre Zähler werden gesammelt:

Objekt:	Kennungen:	Attribute:	Datenpunkte:
Redis	Namespace-Server		

Fehlerbehebung

Weitere Informationen finden Sie auf der "Support" Seite.