Datensammlerreferenz – Dienste

Data Infrastructure Insights

NetApp
February 11, 2026

Inhalt

Datensammlerreferenz – Dienste

Knotendatenerfassung

Data Infrastructure Insights sammelt Metriken von 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 beliebigen Integrationsdatensammlers (Kubernetes, Docker, Apache usw.) auch die Knotendatenerfassung konfiguriert wird.
2. Befolgen Sie die Anweisungen zum Konfigurieren des Agenten. Die Anweisungen variieren je nach Art des Betriebssystems oder der Plattform, die Sie zum Sammeln von Daten verwenden.

Objekte und Zähler

Die folgenden Objekte und ihre Zähler werden als Knotenmetriken erfasst:

Objekt:	Kennungen:	Merkmale:	Datenpunkte:
Knotendateisystem	Knoten-UUID-Gerätepfadtyp	Knoten-IP Knotenname Knoten-Betriebssystemmodus	Freie Inodes Freie Inodes Gesamtanzahl der verwendeten Inodes Gesamtanzahl der verwendeten Inodes Gesamtanzahl der verwendeten Inodes
Knotendatenträger	Knoten-UUID-Datenträger	Knoten-IP Knotenname Knoten-Betriebssystem	IO-Zeit Laufende IOPS Gesamtanzahl gelesener Bytes (pro Sek.) Lesezeit Gesamtanzahl gelesener Bytes (pro Sek.) Gewichtete IO-Zeit Gesamtanzahl geschriebener Bytes (pro Sek.) Schreibzeit Gesamtanzahl geschriebener Bytes (pro Sek.) Aktuelle Warteschlangenlänge der Festplatte Schreibzeit Lesezeit IO-Zeit
Knoten-CPU	Knoten-UUID-CPU	Knoten-IP Knotenname Knoten-Betriebssystem	System-CPU-Auslastung Benutzer-CPU-Auslastung Leerlauf-CPU-Auslastung Prozessor-CPU-Auslastung Interrupt-CPU-Auslastung DPC-CPU-Auslastung

Objekt:	Kennungen:	Merkmale:	Datenpunkte:
Node	Knoten-UUID	Knoten-IP Knotenname Knoten-Betriebssystem	Kernel-Bootzeit Kernel-Kontextwechsel (pro Sek.) Verfügbare Kernel-Entropie Kernel-Interrupts (pro Sek.) Abgespaltene Kernel-Prozesse (pro Sek.) Aktiver Speicher Verfügbarer Gesamtspeicher Verfügbarer Speicher Gepufferter Speicher Zwischengespeicherter Speicher Commit-Limit Als Speicher festgeschriebener Speicher Dirty Memory Freier Speicher hoch Freier Speicher hoch Gesamtspeicher Größe großer Seiten Speicher Große Seiten Freier Speicher Große Seiten Gesamtspeicher niedrig Freier Speicher niedrig Gesamtspeicher Zugeordneter Speicher Seitentabellen Speicher Gemeinsam genutzter Speicher Slab-Speicher Swap Zwischengespeicherter Speicher Swap Freier Speicher Swap Gesamtspeicher Gesamt verwendeter Speicher Gesamt verwendeter Speicher Speicher Vmalloc-Chunk-Speicher Vmalloc Gesamtspeicher Vmalloc Verwendeter Speicher Verdrahteter Speicher Writeback Gesamter Speicher Writeback Temporärer Speicher Cache-Fehler Speicherbedarf Null-Fehler Speicherseitenfehler Speicherseiten Speicher Nicht ausgelagerter Speicher Ausgelagerter Speicher Cache-Kernspeicher Standby-Cache Normaler Speicher Standby-Cache

Objekt:	Kennungen:	Merkmale:	Datenpunkte:
Knotennetzwerk	Netzwerkschnittstellenknoten-UUID	Knotenname Knoten-IP Knoten-Betriebssystem	Empfangene Bytes Gesendete Bytes Ausgehende Pakete Verworfene Pakete Ausgehende Fehler Empfangene Pakete Verworfene Pakete Empfangene Fehler Empfangene Pakete Gesendete Pakete

Aufstellen

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

ActiveMQ-Datensammler

Data Infrastructure Insights verwendet diesen Datensammler, um Metriken von 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 der der Telegraf-Agent installiert ist.

2. Wenn Sie noch keinen Agenten für die Sammlung installiert haben oder einen Agenten für ein anderes Betriebssystem oder eine andere Plattform installieren möchten, klicken Sie auf *Anweisungen anzeigen*, um das"Agenteninstallation" Anweisungen.
3. Wählen Sie den Agent-Zugriffsschlüssel zur Verwendung mit diesem Datensammler aus. Sie können einen neuen Agentenzugriffsschlüssel hinzufügen, indem Sie auf die Schaltfläche **+ Agentenzugriffsschlüssel** klicken. Best Practice: Verwenden Sie nur dann einen anderen Agent-Zugriffsschlüssel, wenn Sie Datensammler beispielsweise nach Betriebssystem/Plattform gruppieren möchten.
4. Befolgen Sie die Konfigurationsschritte, um den Datensammler zu konfigurieren. Die Anweisungen variieren je nach Art des Betriebssystems oder der Plattform, die Sie zum Sammeln von Daten verwenden.

[ActiveMQ-Konfiguration]

Aufstellen

Informationen finden Sie im "ActiveMQ-Dokumentation"

Objekte und Zähler

Die folgenden Objekte und ihre Zähler werden gesammelt:

Objekt:	Kennungen:	Merkmale:	Datenpunkte:
ActiveMQ-Warteschlange	Namespace-Warteschlangen-Port-Server	Knotenname Knoten-IP Knoten-UUID	Anzahl der Verbraucher Anzahl der Dequeues Anzahl der Enqueues Warteschlangengröße
ActiveMQ-Abonnent	Client-ID Verbindungs-ID Port Server-Namespace	Ist aktiv Zielknotenname Knoten-IP Knoten-UUID Knoten-Betriebssystem-Selektor Abonnement	Anzahl der aus der Warteschlange entfernten Aufgaben Anzahl der versendeten Aufgaben Warteschlangengröße Anzahl der in die Warteschlange entfernten Aufgaben Größe der ausstehenden Warteschlange
ActiveMQ-Thema	Thema Port Server Namespace	Knotenname Knoten-IP Knoten-UUID Knoten-Betriebssystem	Anzahl der Verbraucher Anzahl der Dequeues Anzahl der Enqueues Anzahl der Größe

Fehlerbehebung

Weitere Informationen finden Sie in der"Support" Seite.

Apache-Datensammler

Dieser Datensammler ermöglicht das Sammeln von Daten von Apache-Servern auf Ihrem Mandanten.

Voraussetzungen

- Sie müssen Ihren Apache HTTP-Server eingerichtet und ordnungsgemäß ausgeführt haben
- Sie müssen über Sudo- oder Administratorberechtigungen auf Ihrem Agent-Host/Ihrer VM verfügen
- Normalerweise ist das Apache-Modul *mod_status* 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 zum Konfigurieren Ihres Servers finden Sie in der Apache-Moduldokumentation: 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 der der Telegraf-Agent installiert ist.

2. Wenn Sie noch keinen Agenten für die Sammlung installiert haben oder einen Agenten für ein anderes Betriebssystem oder eine andere Plattform installieren möchten, klicken Sie auf *Anweisungen anzeigen*, um das"Agenteninstallation" Anweisungen.
3. Wählen Sie den Agent-Zugriffsschlüssel zur Verwendung mit diesem Datensammler aus. Sie können einen neuen Agentenzugriffsschlüssel hinzufügen, indem Sie auf die Schaltfläche **+ Agentenzugriffsschlüssel** klicken. Best Practice: Verwenden Sie nur dann einen anderen Agent-Zugriffsschlüssel, wenn Sie

Datensammler beispielsweise nach Betriebssystem/Plattform gruppieren möchten.

4. Befolgen Sie die Konfigurationsschritte, um den Datensammler zu konfigurieren. Die Anweisungen variieren je nach Art des Betriebssystems oder der Plattform, die Sie zum Sammeln von Daten verwenden.

[Apache-Konfiguration]

Aufstellen

Das Plugin von Telegraf für den HTTP-Server von Apache erfordert die Aktivierung des Moduls „mod_status“. Wenn dies aktiviert ist, stellt der HTTP-Server von Apache einen HTML-Endpunkt bereit, der in Ihrem Browser angezeigt oder zum Extrahieren des Status der gesamten HTTP-Serverkonfiguration von Apache verwendet werden kann.

Kompatibilität:

Die Konfiguration wurde für Apaches HTTP-Server Version 2.4.38 entwickelt.

Aktivieren von mod_status:

Das Aktivieren und Freigeben der „mod_status“-Module umfasst zwei Schritte:

- Aktivierungsmodul
- Statistiken aus dem Modul anzeigen

Aktivierungsmodul:

Das Laden der Module wird durch die Konfigurationsdatei unter „/usr/local/apache/conf/httpd.conf“ gesteuert. Bearbeiten Sie die Konfigurationsdatei und entfernen Sie die Kommentarzeichen aus den folgenden Zeilen:

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

Statistiken aus dem Modul anzeigen:

Die Offenlegung von „mod_status“ wird durch die Konfigurationsdatei unter „/usr/local/apache2/conf/extra/httpd-info.conf“ gesteuert. Stellen Sie sicher, dass die Konfigurationsdatei Folgendes enthält (zumindest andere Anweisungen werden dort vorhanden sein):

```
# 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

Die folgenden Objekte und ihre Zähler werden gesammelt:

Objekt:	Kennungen:	Merkmale:	Datenpunkte:
Apache	Namespace-Server	Knoten-IP Knotenname Port Übergeordneter Server Konfigurationsgenerierung Übergeordneter Server MPM-Generierung Server-Betriebszeit wird gestoppt	Beschäftigte Worker Bytes pro Anfrage Bytes pro Sekunde CPU-Kinder System-CPU-Kinder Benutzer-CPU-Last CPU-System-CPU-Benutzer Asynchrone Verbindungen Schließende asynchrone Verbindungen Asynchrone Keep-Alive-Verbindungen Schreibende Verbindungen Gesamtdauer pro Anfrage Inaktive Worker Durchschnittliche Auslastung (letzte 1 Min.) Durchschnittliche Auslastung (letzte 15 Min.) Durchschnittliche Auslastung (letzte 5 Min.) Prozesse Anfragen pro Sekunde Gesamtzugriffe Gesamtdauer Gesamt-KByte Anzeigetafel Anzeigetafel wird geschlossen Anzeigetafel DNS-Lookups Anzeigetafel wird beendet Anzeigetafel Anzeigetafel Leerlaufbereinigung Anzeigetafel Keep-Alive Anzeigetafel wird protokolliert Anzeigetafel geöffnet Anzeigetafel liest Anzeigetafel sendet Anzeigetafel startet Anzeigetafel wartet

Fehlerbehebung

Weitere Informationen finden Sie in der"Support" Seite.

Konsul-Datensammler

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

Installation

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

Wenn Sie keinen Agenten für die Sammlung konfiguriert haben, werden Sie aufgefordert,"Installieren eines Agenten" auf Ihrem Mieter.

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 Consul-Konfigurationsbildschirm, um den Datensammler zu konfigurieren. Die Anweisungen variieren je nach Art des Betriebssystems oder der Plattform, die Sie zum Sammeln von Daten verwenden.

Aufstellen

Informationen finden Sie im"Konsuldokumentation" .

Objekte und Zähler für Konsul

Die folgenden Objekte und ihre Zähler werden gesammelt:

Objekt:	Kennungen:	Merkmale:	Datenpunkte:
Konsul	Namespace-Check-ID-Dienstknoten	Knoten-IP Knoten-Betriebssystem Knoten-UUID Knotenname Dienstname Name prüfen Dienst-ID Status	Warnung vor kritischem Überholen

Fehlerbehebung

Weitere Informationen finden Sie in der"Support" Seite.

Couchbase-Datensammler

Data Infrastructure Insights verwendet diesen Datensammler, um Metriken von Couchbase zu sammeln.

Installation

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

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

2. Wenn Sie noch keinen Agenten für die Sammlung installiert haben oder einen Agenten für ein anderes Betriebssystem oder eine andere Plattform installieren möchten, klicken Sie auf *Anweisungen anzeigen*, um das"Agenteninstallation" Anweisungen.
3. Wählen Sie den Agent-Zugriffsschlüssel zur Verwendung mit diesem Datensammler aus. Sie können einen neuen Agentenzugriffsschlüssel hinzufügen, indem Sie auf die Schaltfläche **+ Agentenzugriffsschlüssel** klicken. Best Practice: Verwenden Sie nur dann einen anderen Agent-Zugriffsschlüssel, wenn Sie Datensammler beispielsweise nach Betriebssystem/Plattform gruppieren möchten.
4. Befolgen Sie die Konfigurationsschritte, um den Datensammler zu konfigurieren. Die Anweisungen variieren je nach Art des Betriebssystems oder der Plattform, die Sie zum Sammeln von Daten verwenden.

[Couchbase-Konfiguration]

Aufstellen

Informationen finden Sie im"Couchbase-Dokumentation" .

Objekte und Zähler

Die folgenden Objekte und ihre Zähler werden gesammelt:

Objekt:	Kennungen:	Merkmale:	Datenpunkte:
Couchbase-Knoten	Namespace-Cluster Couchbase-Knoten-Hostname	Knotenname Knoten-IP	Speicher Freier Speicher Gesamt
Couchbase-Eimer	Namespace-Bucket-Cluster	Knotenname Knoten-IP	Verwendete Daten Datenabrufe Verwendete Festplatte Anzahl der Elemente Verwendeter Speicher Vorgänge pro Sekunde Verwendetes Kontingent

Fehlerbehebung

Weitere Informationen finden Sie in der"Support" Seite.

CouchDB-Datensammler

Data Infrastructure Insights verwendet 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 der der Telegraf-Agent installiert ist.

2. Wenn Sie noch keinen Agenten für die Sammlung installiert haben oder einen Agenten für ein anderes Betriebssystem oder eine andere Plattform installieren möchten, klicken Sie auf *Anweisungen anzeigen*, um das"Agenteninstallation" Anweisungen.
3. Wählen Sie den Agent-Zugriffsschlüssel zur Verwendung mit diesem Datensammler aus. Sie können einen neuen Agentenzugriffsschlüssel hinzufügen, indem Sie auf die Schaltfläche **+ Agentenzugriffsschlüssel** klicken. Best Practice: Verwenden Sie nur dann einen anderen Agent-Zugriffsschlüssel, wenn Sie Datensammler beispielsweise nach Betriebssystem/Plattform gruppieren möchten.
4. Befolgen Sie die Konfigurationsschritte, um den Datensammler zu konfigurieren. Die Anweisungen variieren je nach Art des Betriebssystems oder der Plattform, die Sie zum Sammeln von Daten verwenden.

[CouchDB-Konfiguration]

Aufstellen

Informationen finden Sie im"CouchDB-Dokumentation" .

Objekte und Zähler

Die folgenden Objekte und ihre Zähler werden gesammelt:

Objekt:	Kennungen:	Merkmale:	Datenpunkte:
CouchDB	Namespace-Server	Knotenname Knoten-IP	Authentifizierungscache-Treffer Authentifizierungscache-Fehler Datenbank-Lesevorgänge Datenbank-Schreibvorgänge Datenbanken öffnen Betriebssystemdateien öffnen Max. Anforderungszeit Min. Anforderungszeit HTTP-Anforderungsmethoden HTTP-Anforderungsmethoden kopieren HTTP-Anforderungsmethoden löschen HTTP-Anforderungsmethoden abrufen HTTP-Anforderungsmethoden headen HTTP-Anforderungsmethoden posten HTTP-Anforderungsmethoden setzen Statuscodes 200 Statuscodes 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 in der"Support" Seite.

Docker-Datensammler

Data Infrastructure Insights verwendet diesen Datensammler, um Metriken von Docker zu sammeln.

Installation

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

   Wenn Sie keinen Agenten für die Sammlung konfiguriert haben, werden Sie aufgefordert,"Installieren eines Agenten" auf Ihrem Mieter.

   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 Docker-Konfigurationsbildschirm, um den Datensammler zu konfigurieren. Die Anweisungen variieren je nach Art des Betriebssystems oder der Plattform, die Sie zum Sammeln von Daten verwenden.

[Docker-Konfiguration]

Aufstellen

Das Telegraf-Eingabe-Plugin für Docker sammelt Metriken über einen angegebenen UNIX-Socket oder einen TCP-Endpunkt.

Kompatibilität

Die Konfiguration wurde für Docker Version 1.12.6 entwickelt.

Einrichten

Zugriff auf Docker über einen UNIX-Socket

Wenn der Telegraf-Agent auf Baremetal ausgeführt wird, 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 frei, indem Sie den Socket als Volume in den Pod
einbinden und dieses Volume dann in /var/run/docker.sock mounten.  Fügen
Sie der PodSpec beispielsweise 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 für die Kubernetes-Plattform bereitgestellte Data Infrastructure Insights -Installationsprogramm diese Zuordnung automatisch vornimmt.

Zugriff auf Docker über einen TCP-Endpunkt

Standardmäßig verwendet Docker Port 2375 für unverschlüsselten Zugriff und Port 2376 für verschlüsselten Zugriff.

Objekte und Zähler

Die folgenden Objekte und ihre Zähler werden gesammelt:

Objekt:	Kennungen:	Merkmale:	Datenpunkte:
Docker-Engine	Namespace Docker Engine	Knotenname Knoten-IP Knoten-UUID Knoten-Betriebssystem Kubernetes-Cluster Docker-Version Einheit	Speichercontainer Container Angehaltene Container Laufende Container Gestoppte CPUs Go-Routinen Bilder Listener Verwendete Ereignisse Datei-Deskriptoren Daten Verfügbare Daten Gesamt verwendete Daten Metadaten Verfügbare Metadaten Gesamt verwendete Metadaten Pool-Blockgröße

Objekt:	Kennungen:	Merkmale:	Datenpunkte:
Docker-Container	Namespace Containername Docker Engine	Kubernetes-Container-Hash Kubernetes-Container-Ports Kubernetes-Container-Neustart-Zähler Kubernetes-Container-Beendigungsnachrichtenpfad Kubernetes-Container-Beendigungsnachrichtenrichtlinie Kubernetes-Pod-Beendigungsfrist Container-Image Container-Status Container-Version Knotenname Kubernetes-Container-Protokollpfad Kubernetes-Container-Name Kubernetes-Docker-Typ Kubernetes-Pod-Name Kubernetes-Pod-Namespace Kubernetes-Pod-UID Kubernetes-Sandbox-ID Knoten-IP Knoten-UUID Docker-Version Kubernetes-E/A-Konfiguration gesehen Kubernetes-E/A-Konfigurationsquelle OpenShift-E/A-SCC Kubernetes-Beschreibung Kubernetes-Anzeigename OpenShift-Tags Kompose-Service Pod-Vorlagen-Hash Controller-Revisions-Hash Pod-Vorlagengenerierung Lizenzschema Erstellungsdatum Schema-Lizenz Schemaname Schema-URL Schema-VCS-URL Schemaanbieter Schemaversion Schema Schemaversionsbetreuer Kunde Pod Kubernetes StatefulSet Pod-Name Mandant Webkonsolen-Architektur Autoritative Quell-URL Erstellungsdatum RH-Build-Host RH-Komponente Verteilungsbereich Installation Release-	Aktiver anonymer Speicher Aktiver Dateispeicher Cache-Speicher Hierarchische Grenze Inaktiver anonymer Speicher Inaktive Dateispeichergrenze Zugeordneter Dateispeicher Maximale Speicherauslastung Speicherseitenfehler Schwerwiegender Seitenfehler Eingelagerter Speicher Ausgelagerter Speicher Ausgelagerter Speicher Größe des residenten Satzes Riesiger residenter Satz Größe des residenten Satzes Gesamter aktiver anonymer Speicher Gesamter aktiver Dateispeicher Gesamter Cache-Speicher Gesamter inaktiver anonymer Speicher Gesamter inaktiver Dateispeicher Gesamter zugeordneter Dateispeicher Gesamter Seitenfehler Speicher Gesamter schwerwiegender Seitenfehler Gesamter ausgelagerter Speicher Gesamter ausgelagerter Speicher Gesamte Größe des residenten Satzes Gesamter residenter Satz Größe des residenten Satzes Gesamter nicht auslagerbarer Speicher Nicht auslagerbarer Speichernutzung Speichernutzung in Prozent Exit-Code OOM beendet PID gestartet bei Failing Streak

Objekt:	Kennungen:	Merkmale:	Datenpunkte:
Docker-Container-Block-IO	Namespace Containername Gerät Docker Engine	Kubernetes-Container-Hash Kubernetes-Container-Ports Anzahl der Neustarts von Kubernetes-Containern Pfad der Kubernetes-Container-Beendigungsnachrichten Richtlinie für Kubernetes-Container-Beendigungsnachrichten Karenzzeit für Kubernetes-Pod-Beendigung Container-Image Container-Status Container-Version Knotenname Kubernetes-Container-Protokollpfad Kubernetes-Container-Name Kubernetes-Docker-Typ Kubernetes-Pod-Name Kubernetes-Pod-Namespace Kubernetes-Pod-UID Kubernetes-Sandbox-ID Knoten-IP Knoten-UUID Docker-Version Gesehene Kubernetes-Konfiguration Kubernetes-Konfigurationsquelle OpenShift SCC Kubernetes-Beschreibung Kubernetes-Anzeigename OpenShift-Tags Schema Schemaversion Pod-Vorlagen-Hash Controller-Revisions-Hash Pod-Vorlagengenerierung Kompose-Dienst Schema-Erstellungsdatum Schema-Lizenz Schemaname Schemaanbieter Kunden-Pod Kubernetes-StatefulSet Pod-Name Mandant-Webkonsole Erstellungsdatum Lizenzanbieter Architektur Autoritative Quell-URL RH-Build-Host RH-Komponente Verteilungsbereich Installationsbetreuer Release-Ausführungszusammenfa	IO-Service-Bytes rekursiv Asynchron IO-Service-Bytes rekursiv Lesen IO-Service-Bytes rekursiv Sync IO-Service-Bytes rekursiv gesamt IO-Service-Bytes rekursiv Schreiben IO bedient rekursiv Asynchron IO bedient rekursiv Lesen IO bedient rekursiv Sync IO bedient rekursiv gesamt IO bedient rekursiv Schreiben

Objekt:	Kennungen:	Merkmale:	Datenpunkte:
Docker-Container-Netzwerk	Namespace Containername Netzwerk Docker Engine	Container-Image Container-Status Container-Version Knotenname Knoten-IP Knoten-UUID Knoten-Betriebssystem K8s-Cluster Docker-Version Container-ID	RX Verloren RX Bytes RX Fehler RX Pakete TX Verloren TX Bytes TX Fehler TX Pakete

Objekt:	Kennungen:	Merkmale:	Datenpunkte:
Docker-Container-CPU	Namespace Containername CPU Docker Engine	Kubernetes-Container-Hash Kubernetes-Container-Ports Kubernetes-Container-Neustart-Zähler Kubernetes-Container-Beendigungsnachrichtenpfad Kubernetes-Container-Beendigungsnachrichtenrichtlinie Kubernetes-Pod-Beendigungsfrist Kubernetes-Konfiguration gesehen Kubernetes-Konfigurationsquelle OpenShift SCC-Container-Image Containerstatus Containerversion Knotenname Kubernetes-Container-Protokollpfad Kubernetes-Containername Kubernetes-Docker-Typ Kubernetes-Pod-Name Kubernetes-Pod-Namespace Kubernetes-Pod-UID Kubernetes-Sandbox-ID Knoten-IP Knoten-UUID Knoten-Betriebssystem Kubernetes-Cluster Docker-Version Kubernetes-Beschreibung Kubernetes-Anzeigename OpenShift-Tags Schemaversion Pod-Vorlagen-Hash Controller-Revisions-Hash Pod-Vorlagengenerierung Kompose-Dienst Schema-Erstellungsdatum Schema-Lizenz Schemaname Schemaanbieter Kunden-Pod Kubernetes-StatefulSet Pod-Name Mandant-Webkonsole Erstellungsdatum Lizenzanbieter Architektur Autoritative Quell-URL RH-Build-Host RH-Komponente Verteilungsbereich Installationsbetreuer Release-	Drosselungsperioden Drosselung Drosselungsperioden Drosselungszeit Drosselungszeit Nutzung im Kernelmodus Nutzung im Benutzermodus Nutzung in Prozent Nutzung Systemnutzung gesamt

Fehlerbehebung

Problem:	Versuchen Sie Folgendes:
Ich sehe meine Docker-Metriken nicht in Data Infrastructure Insights, nachdem ich die Anweisungen auf der Konfigurationsseite befolgt habe.	Überprüfen Sie die Protokolle des Telegraf-Agenten, um zu sehen, ob der folgende Fehler gemeldet wird: E! Fehler im Plugin [inputs.docker]: Beim Versuch, eine Verbindung zum Docker-Daemon-Socket herzustellen, wurde die Berechtigung verweigert. Wenn dies der Fall ist, ergreifen Sie die erforderlichen Maßnahmen, um dem Telegraf-Agenten Zugriff auf den Docker-Unix-Socket zu gewähren, wie oben angegeben.

Weitere Informationen finden Sie in der"Support" Seite.

Elasticsearch-Datensammler

Data Infrastructure Insights verwendet diesen Datensammler, um Metriken von Elasticsearch zu sammeln.

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

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

2. Wenn Sie noch keinen Agenten für die Sammlung installiert haben oder einen Agenten für ein anderes Betriebssystem oder eine andere Plattform installieren möchten, klicken Sie auf *Anweisungen anzeigen*, um das"Agenteninstallation" Anweisungen.
3. Wählen Sie den Agent-Zugriffsschlüssel zur Verwendung mit diesem Datensammler aus. Sie können einen neuen Agentenzugriffsschlüssel hinzufügen, indem Sie auf die Schaltfläche **+ Agentenzugriffsschlüssel** klicken. Best Practice: Verwenden Sie nur dann einen anderen Agent-Zugriffsschlüssel, wenn Sie Datensammler beispielsweise nach Betriebssystem/Plattform gruppieren möchten.
4. Befolgen Sie die Konfigurationsschritte, um den Datensammler zu konfigurieren. Die Anweisungen variieren je nach Art des Betriebssystems oder der Plattform, die Sie zum Sammeln von Daten verwenden.

[Elasticsearch-Konfiguration]

Aufstellen

Informationen finden Sie im"Elasticsearch-Dokumentation" .

Objekte und Zähler

Die folgenden Objekte und ihre Zähler werden gesammelt:

Objekt:	Kennungen:	Merkmale:
Elasticsearch-Cluster	Namespace-Cluster	Knoten-IP Knotenname Clusterstatus
Elasticsearch-Knoten	Namespace-Cluster ES-Knoten-ID ES-Knoten-IP ES-Knoten	Zonen-ID

Fehlerbehebung

Weitere Informationen finden Sie in der"Support" Seite.

Flink-Datensammler

Data Infrastructure Insights verwendet diesen Datensammler, um Metriken von Flink zu sammeln.

Installation

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

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

2. Wenn Sie noch keinen Agenten für die Sammlung installiert haben oder einen Agenten für ein anderes Betriebssystem oder eine andere Plattform installieren möchten, klicken Sie auf *Anweisungen anzeigen*, um das"Agenteninstallation" Anweisungen.
3. Wählen Sie den Agent-Zugriffsschlüssel zur Verwendung mit diesem Datensammler aus. Sie können einen neuen Agentenzugriffsschlüssel hinzufügen, indem Sie auf die Schaltfläche **+ Agentenzugriffsschlüssel** klicken. Best Practice: Verwenden Sie nur dann einen anderen Agent-Zugriffsschlüssel, wenn Sie Datensammler beispielsweise nach Betriebssystem/Plattform gruppieren möchten.
4. Befolgen Sie die Konfigurationsschritte, um den Datensammler zu konfigurieren. Die Anweisungen variieren je nach Art des Betriebssystems oder der Plattform, die Sie zum Sammeln von Daten verwenden.

[Flink-Konfiguration]

Aufstellen

Eine vollständige Flink-Bereitstellung umfasst die folgenden Komponenten:

JobManager: Das Flink-Primärsystem. Koordiniert eine Reihe von TaskManagern. In einer Hochverfügbarkeitskonfiguration verfügt das System über mehr als einen JobManager. TaskManager: Hier werden Flink-Operatoren ausgeführt. Das Flink-Plugin basiert auf dem Jolokia-Plugin von Telegraf. Um beispielsweise Informationen von allen Flink-Komponenten zu sammeln, muss JMX auf allen Komponenten konfiguriert und über Jolokia verfügbar gemacht werden.

Kompatibilität

Die Konfiguration wurde für Flink Version 1.7.0 entwickelt.

Einrichten

Jolokia Agent Jar

Für alle Einzelkomponenten muss eine Version der Jolokia-Agent-JAR-Datei heruntergeladen werden. Die getestete Version war"Jolokia-Agent 1.6.0" .

Die folgenden Anweisungen gehen davon aus, dass die heruntergeladene JAR-Datei (jolokia-jvm-1.6.0-agent.jar) am Speicherort „/opt/flink/lib/“ abgelegt ist.

JobManager

Um JobManager so zu konfigurieren, dass die Jolokia-API verfügbar gemacht 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 für Jolokia einen anderen Port (8778) wählen.  Wenn Sie eine
interne IP haben, auf die Sie Jolokia sperren möchten, können Sie die
„Catch-All"-IP 0.0.0.0 durch Ihre eigene IP ersetzen.  Beachten Sie, dass
diese IP vom Telegraf-Plugin aus zugänglich sein muss.
```

Task-Manager

Um TaskManager so zu konfigurieren, dass sie die Jolokia-API verfügbar machen, 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 für Jolokia einen anderen Port (8778) wählen.  Wenn Sie eine
interne IP haben, auf die Sie Jolokia sperren möchten, können Sie die
„Catch-All"-IP 0.0.0.0 durch Ihre eigene IP ersetzen.  Beachten Sie, dass
diese IP vom Telegraf-Plugin aus zugänglich sein muss.
```

Objekte und Zähler

Die folgenden Objekte und ihre Zähler werden gesammelt:

Objekt:	Kennungen:	Merkmale:	Datenpunkte:
Flink Task-Manager	Cluster-Namespace-Server	Knotenname Task-Manager-ID Knoten-IP	Verfügbare Netzwerkspeichersegmente Netzwerkspeichersegmente gesamt Garbage Collection PS MarkSweep-Anzahl Garbage Collection PS MarkSweep-Zeit Garbage Collection PS Scavenge-Anzahl Garbage Collection PS Scavenge-Zeit Heap-Speicher zugesichert Heap-Speicher initialisieren Heap-Speicher max. Verwendeter Heap-Speicher Thread-Anzahl Daemon-Threads Spitzen-Thread-Anzahl Thread-Anzahl insgesamt gestartet
Flink Job	Job-ID des Cluster-Namespace-Servers	Knotenname Jobname Knoten-IP Letzter Prüfpunkt Externer Pfad Neustartzeit	Ausfallzeit Vollständige Neustarts Ausrichtung des letzten Prüfpunkts Gepuffert Dauer des letzten Prüfpunkts Größe des letzten Prüfpunkts Anzahl abgeschlossener Prüfpunkte Anzahl fehlgeschlagener Prüfpunkte Anzahl laufender Prüfpunkte Anzahl Prüfpunkte Betriebszeit

Objekt:	Kennungen:	Merkmale:	Datenpunkte:
Flink Job Manager	Cluster-Namespace-Server	Knotenname Knoten-IP	Garbage Collection PS MarkSweep-Anzahl Garbage Collection PS MarkSweep-Zeit Garbage Collection PS Scavenge-Anzahl Garbage Collection PS Scavenge-Zeit Heap-Speicher zugesichert Heap-Speicher initialisieren Heap-Speicher max. verwendeter Heap-Speicher Anzahl registrierter Task-Manager Anzahl laufender Jobs Verfügbare Task-Slots Gesamt-Thread-Anzahl Daemon-Thread-Anzahl Spitzen-Thread-Anzahl Threads insgesamt gestartet

Objekt:	Kennungen:	Merkmale:	Datenpunkte:
Flink-Aufgabe	Cluster-Namespace Job-ID Aufgaben-ID	Serverknotenname Jobname Sub-Task-Index Task-Versuchs-ID Task-Versuchsnummer Taskname Task-Manager-ID Knoten-IP Aktuelles Eingabewasserzeichen	Puffer im Pool, Nutzung, Puffer in Warteschlangenlänge, Puffer aus, Pool, Nutzung, Puffer aus, Warteschlangenlänge, Anzahl, Puffer lokal, Anzahl, Puffer lokal pro Sekunde, Anzahl, Puffer lokal pro Sekunde, Rate, Anzahl, Puffer extern, Anzahl, Puffer extern pro Sekunde, Anzahl, Puffer extern pro Sekunde, Anzahl, Puffer extern pro Sekunde, Rate, Anzahl, Puffer extern, Anzahl, Puffer extern pro Sekunde, Anzahl, Puffer extern pro Sekunde, Rate, Anzahl, Bytes lokal, Anzahl, Bytes lokal pro Sekunde, Anzahl, Bytes lokal pro Sekunde, Rate, Anzahl, Bytes lokal pro Sekunde, Rate, Anzahl, Bytes extern, Anzahl, Bytes extern pro Sekunde, Anzahl, Bytes extern pro Sekunde, Rate, Anzahl, Bytes extern pro Sekunde, Anzahl, Bytes extern pro Sekunde, Rate, Anzahl, Datensätze extern, Anzahl, Datensätze extern pro Sekunde, Rate, Anzahl, Datensätze extern, Anzahl, Datensätze extern pro Sekunde, Rate, Anzahl, Datensätze extern pro Sekunde

Objekt:	Kennungen:	Merkmale:	Datenpunkte:
Flink-Task-Operator	Cluster-Namespace Job-ID Operator-ID Aufgaben-ID	Serverknotenname Jobname Operatorname Sub-Task-Index Task-Versuchs-ID Task-Versuchsnummer Taskname Task-Manager-ID Knoten-IP	Aktuelles Eingabe-Wasserzeichen Aktuelles Ausgabe-Wasserzeichen Anzahl Datensätze (ein) Anzahl Datensätze (ein) pro Sekunde Anzahl Datensätze (ein) pro Sekunde Rate Anzahl Datensätze (aus) Anzahl Datensätze (aus) pro Sekunde Anzahl Datensätze (aus) pro Sekunde Rate Anzahl verspätet gelöschte Datensätze Zugewiesene Partitionen Verbrauchte Byte-Rate Commit-Latenz (Durchschnitt) Commit-Latenz (Maximal) Commit-Rate Fehlgeschlagene Commits Erfolgreiche Commits Verbindungsabschlussrate Anzahl Verbindungen Anzahl Verbindungserstellungsrate Anzahl Abruflatenz (Durchschnitt) Max. Abruflatenz Abrufrate (Durchschnitt) Abrufgröße (Maximal) Abrufgröße (Maximal) Abrufdrosselzeit (Durchschnitt) Max. Abrufdrosselzeit Heartbeat-Rate Eingehende Byte-Rate IO-Verhältnis IO-Zeit (Durchschnitt) (ns) IO-Warteverhältnis IO-Wartezeit (Durchschnitt) Join-Rate durchschnittliche Join-Zeit vor dem letzten Heartbeat Netzwerk-IO-Rate Ausgehende Byte-Rate Verbrauchte Datensätze Rate der Datensätze Verzögerung (Maximal) Datensätze pro Anfrage durchschnittliche Anfragerate Anfragegröße durchschnittliche Anfragegröße (Maximal) Antwortrate Auswahlrate

Fehlerbehebung

Weitere Informationen finden Sie in der"Support" Seite.

Hadoop-Datensammler

Data Infrastructure Insights verwendet diesen Datensammler, um Metriken von Hadoop zu sammeln.

Installation

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

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

2. Wenn Sie noch keinen Agenten für die Sammlung installiert haben oder einen Agenten für ein anderes Betriebssystem oder eine andere Plattform installieren möchten, klicken Sie auf *Anweisungen anzeigen*, um das"Agenteninstallation" Anweisungen.
3. Wählen Sie den Agent-Zugriffsschlüssel zur Verwendung mit diesem Datensammler aus. Sie können einen neuen Agentenzugriffsschlüssel hinzufügen, indem Sie auf die Schaltfläche **+ Agentenzugriffsschlüssel** klicken. Best Practice: Verwenden Sie nur dann einen anderen Agent-Zugriffsschlüssel, wenn Sie Datensammler beispielsweise nach Betriebssystem/Plattform gruppieren möchten.
4. Befolgen Sie die Konfigurationsschritte, um den Datensammler zu konfigurieren. Die Anweisungen variieren je nach Art des Betriebssystems oder der Plattform, die Sie zum Sammeln von Daten verwenden.

[Hadoop-Konfiguration] [Hadoop-Konfiguration]

Aufstellen

Eine vollständige Hadoop-Bereitstellung umfasst die folgenden Komponenten:

- NameNode: Das primäre System des Hadoop Distributed File System (HDFS). Koordiniert eine Reihe von DataNodes.
- Sekundärer NameNode: ein Warm-Failover für den Haupt-NameNode. In Hadoop erfolgt die Beförderung zum NameNode nicht automatisch. Der sekundäre NameNode sammelt Informationen vom NameNode, um bei Bedarf für die Beförderung bereit zu sein.
- DataNode: Tatsächlicher Eigentümer der Daten.
- ResourceManager: Das primäre Rechensystem (Yarn). Koordiniert eine Reihe von NodeManagern.
- NodeManager: Die Ressource für die Berechnung. Tatsächlicher Speicherort für die Ausführung von Anwendungen.
- JobHistoryServer: Verantwortlich für die Bearbeitung aller Anfragen zum Jobverlauf.

Das Hadoop-Plugin basiert auf dem Jolokia-Plugin von Telegraf. Um beispielsweise Informationen von allen Hadoop-Komponenten zu sammeln, muss JMX auf allen Komponenten konfiguriert und über Jolokia verfügbar gemacht werden.

Kompatibilität

Die Konfiguration wurde für Hadoop Version 2.9.2 entwickelt.

Einrichten

Jolokia Agent Jar

Für alle Einzelkomponenten muss eine Version der Jolokia-Agent-JAR-Datei heruntergeladen werden. Die getestete Version war"Jolokia-Agent 1.6.0" .

Die folgenden Anweisungen gehen davon aus, dass die heruntergeladene JAR-Datei (jolokia-jvm-1.6.0-agent.jar) am Speicherort „/opt/hadoop/lib/“ abgelegt ist.

NameNode

Um NameNode für die Bereitstellung der Jolokia-API zu konfigurieren, können Sie Folgendes in <HADOOP_HOME>/etc/hadoop/hadoop-env.sh 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 für die Bereitstellung der Jolokia-API zu konfigurieren, 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.
```

Datenknoten

Um die DataNodes so zu konfigurieren, dass sie die Jolokia-API verfügbar machen, 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.
```

Ressourcenmanager

Um den ResourceManager für die Bereitstellung der Jolokia-API zu konfigurieren, 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.
```

Knotenmanager

Um die NodeManager so zu konfigurieren, dass sie die Jolokia-API verfügbar machen, 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.
```

JobHistoryServer

Um den JobHistoryServer für die Bereitstellung der Jolokia-API zu konfigurieren, 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

Die folgenden Objekte und ihre Zähler werden gesammelt:

Objekt:	Kennungen:	Merkmale:
Sekundärer NameNode von Hadoop	Cluster-Namespace-Server	Knotenname Knoten-IP Kompilierungsinformationen Version
Hadoop NodeManager	Cluster-Namespace-Server	Knotenname Knoten-IP
Hadoop-Ressourcenmanager	Cluster-Namespace-Server	Knotenname Knoten-IP
Hadoop-Datenknoten	Cluster-Namespace-Server	Knotenname Knoten-IP Cluster-ID Version

Objekt:	Kennungen:	Merkmale:
Hadoop-Namensknoten	Cluster-Namespace-Server	Knotenname Knoten-IP Transaktions-ID Zuletzt geschrieben Zeit seit dem letzten Laden Bearbeitungen HA-Status Dateisystemstatus Blockpool-ID Cluster-ID Kompilierungsinformationen Eindeutige Versionsanzahl Version
Hadoop JobHistoryServer	Cluster-Namespace-Server	Knotenname Knoten-IP

Fehlerbehebung

Weitere Informationen finden Sie in der"Support" Seite.

HAProxy-Datensammler

Data Infrastructure Insights verwendet diesen Datensammler, um Metriken 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 der der Telegraf-Agent installiert ist.

2. Wenn Sie noch keinen Agenten für die Sammlung installiert haben oder einen Agenten für ein anderes Betriebssystem oder eine andere Plattform installieren möchten, klicken Sie auf *Anweisungen anzeigen*, um das"Agenteninstallation" Anweisungen.
3. Wählen Sie den Agent-Zugriffsschlüssel zur Verwendung mit diesem Datensammler aus. Sie können einen neuen Agentenzugriffsschlüssel hinzufügen, indem Sie auf die Schaltfläche **+ Agentenzugriffsschlüssel** klicken. Best Practice: Verwenden Sie nur dann einen anderen Agent-Zugriffsschlüssel, wenn Sie Datensammler beispielsweise nach Betriebssystem/Plattform gruppieren möchten.
4. Befolgen Sie die Konfigurationsschritte, um den Datensammler zu konfigurieren. Die Anweisungen variieren je nach Art des Betriebssystems oder der Plattform, die Sie zum Sammeln von Daten verwenden.

[HAProxy-Konfiguration]

Aufstellen

Das Plugin von Telegraf für HAProxy basiert auf der Aktivierung von HAProxy Stats. Dies ist eine in HAProxy integrierte Konfiguration, die jedoch nicht standardmäßig aktiviert ist. Wenn aktiviert, stellt HAProxy einen HTML-Endpunkt bereit, der in Ihrem Browser angezeigt oder zum Extrahieren des Status aller HAProxy-Konfigurationen abgerufen werden kann.

Kompatibilität:

Die Konfiguration wurde für HAProxy Version 1.9.4 entwickelt.

Einrichten:

Um Statistiken zu aktivieren, bearbeiten Sie Ihre Haproxy-Konfigurationsdatei und fügen Sie nach dem Abschnitt „Standardeinstellungen“ die folgenden Zeilen hinzu. Verwenden Sie dabei Ihren eigenen Benutzer/Ihr eigenes Passwort und/oder Ihre Haproxy-URL:

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

Nachfolgend sehen Sie eine vereinfachte Beispielkonfigurationsdatei 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

Die folgenden Objekte und ihre Zähler werden gesammelt:

Objekt:	Kennungen:	Merkmale:	Datenpunkte:
HAProxy Frontend	Namespace-Adressproxy	Knoten-IP Knotenname Proxy-ID Modus Prozess-ID Sitzungsratenbegrenzung Server-ID Sitzungsbegrenzungsstatus	Bytes Eingehend Bytes Ausgehend Cache-Treffer Cache-Suchen Komprimierung Bytes Umgangene Komprimierung Bytes Eingehend Komprimierung Bytes Ausgehend Komprimierung Antworten Verbindungsrate Verbindungsrate Max. Verbindungen Gesamt Durch Verbindungsregel abgelehnte Anfragen Durch Sicherheitsbedenken abgelehnte Anfragen Antworten Durch Sicherheitsbedenken abgelehnte Anfragen Durch Sitzungsregel abgelehnte Anfragen Fehler bei Anfragen Antworten 1xx Antworten 2xx Antworten 3xx Antworten 4xx Antworten 5xx Antworten Andere Anfragen Abgefangene Anfragen Sitzungsrate Sitzungsrate Max. Anfragen Rate Anfragen Rate Max. Anfragen Gesamtanzahl Sitzungen Sitzungen Max. Sitzungen Gesamtanzahl Anfragen Neuschreibungen

Objekt:	Kennungen:	Merkmale:	Datenpunkte:
HAProxy Server	Namespace-Adress-Proxy-Server	Knoten-IP Knotenname Zeit bis zum Abschluss der Prüfung Fallkonfiguration prüfen Gesundheitswert prüfen Anstiegskonfiguration prüfen Status prüfen Proxy-ID Zeit der letzten Änderung Zeit der letzten Sitzung Modus Prozess-ID Server-ID Status Gewicht	Aktive Server Backup-Server Bytes rein Bytes raus Check-Downs Check-Fehlschläge Client-Abbrüche Verbindungen Durchschnittliche Verbindungsdauer Ausfallzeit Gesamt abgelehnte Antworten Verbindungsfehler Antwortfehler Antworten 1xx Antworten 2xx Antworten 3xx Antworten 4xx Antworten 5xx Antworten Andere ausgewählte Server Gesamtwarteschlange Aktuelle Warteschlange Max. Durchschnittliche Warteschlangendauer Sitzungen pro Sekunde Sitzungen pro Sekunde Max. Verbindungswiederverwendung Antwortzeit Durchschnittliche Sitzungen Sitzungen Max. Server-Übertragungsabbrüche Sitzungen Gesamtsitzungen Gesamtzeit Durchschnittliche Anfragen Neuzuweisungen Anfragen Wiederholungsversuche Anfragen Neuschreibungen

Objekt:	Kennungen:	Merkmale:	Datenpunkte:
HAProxy-Backend	Namespace-Adressproxy	Knoten-IP Knotenname Proxy-ID Letzte Änderung Zeit der letzten Sitzung Modus Prozess-ID Server-ID Sitzungslimit Status Gewicht	Aktive Server Backup-Server Bytes In Bytes Out Cache-Treffer Cache-Lookups Check-Downs Client-Abbrüche Komprimierung Bytes Umgangene Komprimierung Bytes In Komprimierung Bytes Out Komprimierung Antworten Verbindungen Durchschnittliche Verbindungsausfallzeit Anfragen insgesamt aufgrund von Sicherheitsbedenken abgelehnt Antworten aufgrund von Sicherheitsbedenken abgelehnt Verbindungsfehler Antwortfehler Antworten 1xx Antworten 2xx Antworten 3xx Antworten 4xx Antworten 5xx Antworten Andere ausgewählte Server Gesamtwarteschlange Aktuelle Warteschlange Max. Durchschnittliche Warteschlangenzeit Sitzungen pro Sekunde Sitzungen pro Sekunde Max. Anfragen Gesamte Verbindungswiederverwendung Antwortzeit Durchschnittliche Sitzungen Sitzungen Max. Serverübertragungsabbrüche Sitzungen Gesamtsitzungen Gesamtzeit Durchschnittliche Anfragen Neuzuweisungen Anfragen Wiederholungsversuche Anfragen Neuschreibungen

Fehlerbehebung

Weitere Informationen finden Sie in der"Support" Seite.

JVM-Datensammler

Data Infrastructure Insights verwendet diesen Datensammler, um Metriken von JVM zu sammeln.

Installation

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

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

2. Wenn Sie noch keinen Agenten für die Sammlung installiert haben oder einen Agenten für ein anderes Betriebssystem oder eine andere Plattform installieren möchten, klicken Sie auf *Anweisungen anzeigen*, um das"Agenteninstallation" Anweisungen.
3. Wählen Sie den Agent-Zugriffsschlüssel zur Verwendung mit diesem Datensammler aus. Sie können einen neuen Agentenzugriffsschlüssel hinzufügen, indem Sie auf die Schaltfläche **+ Agentenzugriffsschlüssel** klicken. Best Practice: Verwenden Sie nur dann einen anderen Agent-Zugriffsschlüssel, wenn Sie Datensammler beispielsweise nach Betriebssystem/Plattform gruppieren möchten.
4. Befolgen Sie die Konfigurationsschritte, um den Datensammler zu konfigurieren. Die Anweisungen variieren je nach Art des Betriebssystems oder der Plattform, die Sie zum Sammeln von Daten verwenden.

[JVM-Konfiguration]

Aufstellen

Informationen finden Sie in"JVM-Dokumentation" .

Objekte und Zähler

Die folgenden Objekte und ihre Zähler werden gesammelt:

Objekt:	Kennungen:	Merkmale:	Datenpunkte:
JVM	Namespace JVM	Betriebssystemarchitektur Betriebssystemname Betriebssystemversion Laufzeitspezifikation Laufzeitspezifikationsanbieter Laufzeitspezifikationsversion Betriebszeit Laufzeit-VM-Name Laufzeit-VM-Anbieter Laufzeit-VM-Version Knotenname Knoten-IP	Geladene Klasse, geladene Klasse, gesamt, entladene Klasse, Speicher-Heap, festgeschriebener Speicher, Heap-Init-Speicher, verwendeter Heap, max. Speicher, verwendeter Heap, nicht festgeschriebener Speicher, nicht festgeschriebener Speicher, nicht festgeschriebener Speicher, nicht festgeschriebener Speicher, nicht festgeschriebener Speicher, max. nicht festgeschriebener Speicher, verwendeter Speicher, ausstehende Finalisierung, verfügbare OS-Prozessoren, Größe des virtuellen OS-Speichers, Größe des freien OS-physischen Speichers, Größe des freien OS-Swap-Speichers, maximale OS-Dateideskriptoranzahl, Anzahl der geöffneten OS-Dateideskriptoren, CPU-Auslastung des OS-Prozessors, CPU-Zeit des OS-Prozessors, CPU-Auslastung des OS-Systems, durchschnittliche OS-Systemauslastung, gesamt physische OS-Speichergröße, Größe des OS-Swap-Speichers insgesamt, Anzahl der Thread-Daemons, Anzahl der Thread-Spitzen, Anzahl der Threads, Anzahl der gestarteten Threads, Anzahl der Garbage Collector-Kopien, Zeit der Garbage Collector-Kopien, Anzahl der Garbage Collector-

Fehlerbehebung

Weitere Informationen finden Sie in der"Support" Seite.

Kafka-Datensammler

Data Infrastructure Insights verwendet diesen Datensammler, um Metriken von 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 der der Telegraf-Agent installiert ist.

2. Wenn Sie noch keinen Agenten für die Sammlung installiert haben oder einen Agenten für ein anderes Betriebssystem oder eine andere Plattform installieren möchten, klicken Sie auf *Anweisungen anzeigen*, um das"Agenteninstallation" Anweisungen.
3. Wählen Sie den Agent-Zugriffsschlüssel zur Verwendung mit diesem Datensammler aus. Sie können einen neuen Agentenzugriffsschlüssel hinzufügen, indem Sie auf die Schaltfläche **+ Agentenzugriffsschlüssel** klicken. Best Practice: Verwenden Sie nur dann einen anderen Agent-Zugriffsschlüssel, wenn Sie Datensammler beispielsweise nach Betriebssystem/Plattform gruppieren möchten.
4. Befolgen Sie die Konfigurationsschritte, um den Datensammler zu konfigurieren. Die Anweisungen variieren je nach Art des Betriebssystems oder der Plattform, die Sie zum Sammeln von Daten verwenden.

[Kafka-Konfiguration]

Aufstellen

Das Kafka-Plugin basiert auf dem Jolokia-Plugin von Telegraf. Um beispielsweise Informationen von allen Kafka-Brokern zu sammeln, muss JMX auf allen Komponenten konfiguriert und über Jolokia verfügbar gemacht werden.

Kompatibilität

Die Konfiguration wurde für Kafka Version 0.11.0.2 entwickelt.

Einrichten

Bei allen folgenden Anweisungen wird davon ausgegangen, dass Ihr Installationsort für Kafka „/opt/kafka“ ist. Sie können die folgenden Anweisungen an Ihren Installationsort anpassen.

Jolokia Agent Jar

Eine Version, die die Jolokia-Agent-JAR-Datei haben muss"heruntergeladen" . Die getestete Version war Jolokia Agent 1.6.0.

Die folgenden Anweisungen gehen davon aus, dass die heruntergeladene JAR-Datei (jolokia-jvm-1.6.0-agent.jar) am Speicherort „/opt/kafka/libs/“ abgelegt ist.

Kafka-Broker

Um Kafka Brokers so zu konfigurieren, dass die Jolokia-API verfügbar gemacht wird, können Sie Folgendes in <KAFKA_HOME>/bin/kafka-server-start.sh direkt vor dem Aufruf von „kafka-run-class.sh“ 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 im obigen Beispiel „hostname -I“ verwendet wird, um die Umgebungsvariable „RMI_HOSTNAME“ einzurichten. Bei Maschinen mit mehreren IP-Adressen muss dies angepasst werden, um die IP-Adresse zu erfassen, die für RMI-Verbindungen von Bedeutung ist.

Sie können für JMX (9999 oben) und Jolokia (8778) einen anderen Port wählen. Wenn Sie eine interne IP haben, auf die Sie Jolokia sperren möchten, können Sie die „Catch-All“-IP 0.0.0.0 durch Ihre eigene IP ersetzen. Beachten Sie, dass diese IP vom Telegraf-Plugin aus zugänglich sein muss. Sie können die Option „-Dcom.sun.management.jmxremote.authenticate=false“ verwenden, wenn Sie keine Authentifizierung wünschen. Die Nutzung erfolgt auf eigene Gefahr.

Objekte und Zähler

Die folgenden Objekte und ihre Zähler werden gesammelt:

Objekt:	Kennungen:	Merkmale:
Kafka Broker	Cluster-Namespace-Broker	Knotenname Knoten-IP

Fehlerbehebung

Weitere Informationen finden Sie in der"Support" Seite.

Kibana-Datensammler

Data Infrastructure Insights verwendet diesen Datensammler, um Metriken 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 der der Telegraf-Agent installiert ist.

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

3. Wählen Sie den Agent-Zugriffsschlüssel zur Verwendung mit diesem Datensammler aus. Sie können einen neuen Agentenzugriffsschlüssel hinzufügen, indem Sie auf die Schaltfläche **+ Agentenzugriffsschlüssel** klicken. Best Practice: Verwenden Sie nur dann einen anderen Agent-Zugriffsschlüssel, wenn Sie Datensammler beispielsweise nach Betriebssystem/Plattform gruppieren möchten.
4. Befolgen Sie die Konfigurationsschritte, um den Datensammler zu konfigurieren. Die Anweisungen variieren je nach Art des Betriebssystems oder der Plattform, die Sie zum Sammeln von Daten verwenden.

[Kibana-Konfiguration]

Aufstellen

Informationen finden Sie im"Kibana-Dokumentation" .

Objekte und Zähler

Die folgenden Objekte und ihre Zähler werden gesammelt:

Objekt:	Kennungen:	Merkmale:	Datenpunkte:
Kibana	Namespace-Adresse	Knoten-IP Knotenname Version Status	Gleichzeitige Verbindungen Heap Max Verwendeter Heap Anfragen pro Sekunde Antwortzeit Durchschnittliche Antwortzeit Max. Betriebszeit

Fehlerbehebung

Weitere Informationen finden Sie in der"Support" Seite.

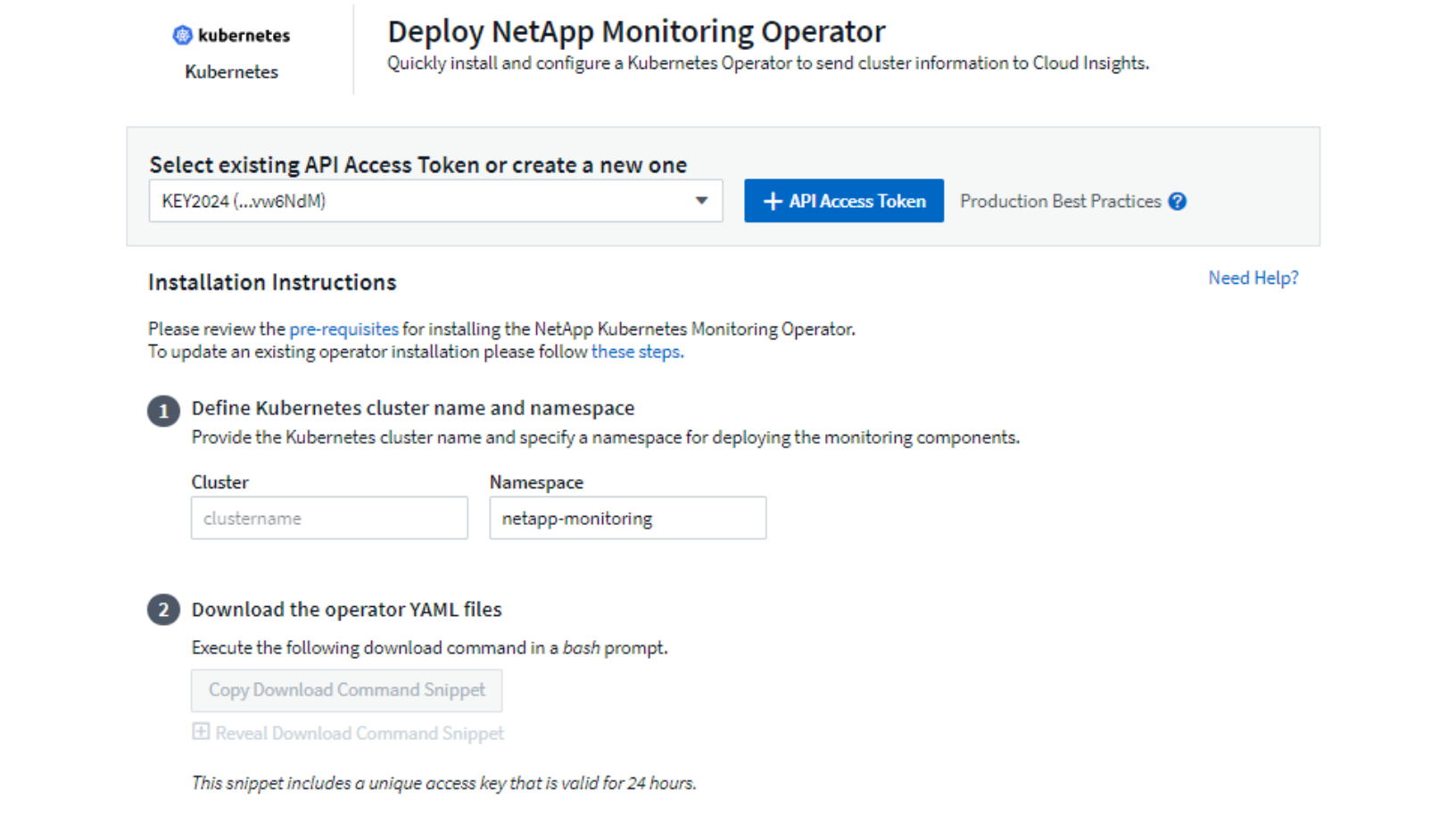
Installation und Konfiguration des Kubernetes Monitoring Operators

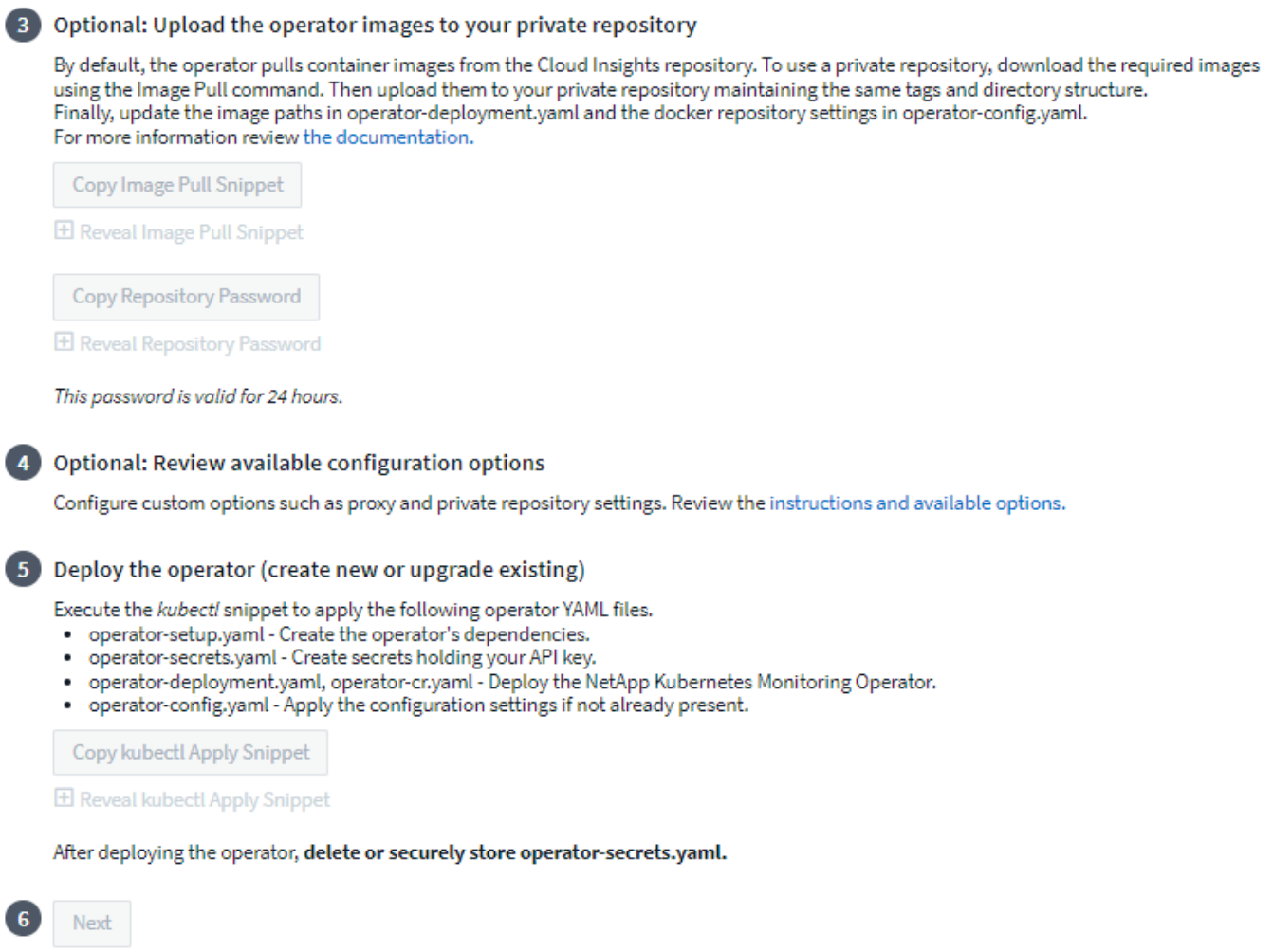
Data Infrastructure Insights bietet den **Kubernetes Monitoring Operator** für die Kubernetes-Sammlung. Navigieren Sie zu **Kubernetes > Collectors > +Kubernetes Collector**, um einen neuen Operator bereitzustellen.

Vor der Installation des Kubernetes Monitoring Operator

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

Installieren des Kubernetes Monitoring Operators


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
clustername
Namespace
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-Agenten auf Kubernetes:

1. Geben Sie einen eindeutigen Clusternamen und Namespace ein. Wenn SieUpgrade von einem vorherigen Kubernetes-Operator, verwenden Sie denselben Clusternamen und Namespace.
2. Sobald diese eingegeben sind, können Sie den Download-Befehlsausschnitt in die Zwischenablage kopieren.
3. Fügen Sie den Snippet in ein *Bash*-Fenster ein und führen Sie ihn aus. Die Operator-Installationsdateien werden heruntergeladen. Beachten Sie, dass das Snippet einen eindeutigen Schlüssel hat und 24 Stunden gültig ist.
4. Wenn Sie ein benutzerdefiniertes oder privates Repository haben, kopieren Sie den optionalen Image Pull-Ausschnitt, fügen Sie ihn in eine *Bash*-Shell ein und führen Sie ihn aus. Sobald die Bilder abgerufen wurden, kopieren Sie sie in Ihr privates Repository. Achten Sie darauf, dieselben Tags und dieselbe Ordnerstruktur beizubehalten. Aktualisieren Sie die Pfade in *operator-deployment.yaml* sowie die Docker-Repository-Einstellungen in *operator-config.yaml*.
5. Überprüfen Sie bei Bedarf die verfügbaren Konfigurationsoptionen wie Proxy- oder private Repository-Einstellungen. Weitere Informationen finden Sie unter"Konfigurationsoptionen" .
6. Wenn Sie bereit sind, stellen Sie den Operator bereit, indem Sie das kubectl Apply-Snippet kopieren, herunterladen und ausführen.
7. Die Installation erfolgt automatisch. Wenn der Vorgang abgeschlossen ist, klicken Sie auf die Schaltfläche *Weiter*.

8. Wenn die Installation abgeschlossen ist, klicken Sie auf die Schaltfläche *Weiter*. Denken Sie daran, auch die Datei *operator-secrets.yaml* zu löschen oder sicher zu speichern.

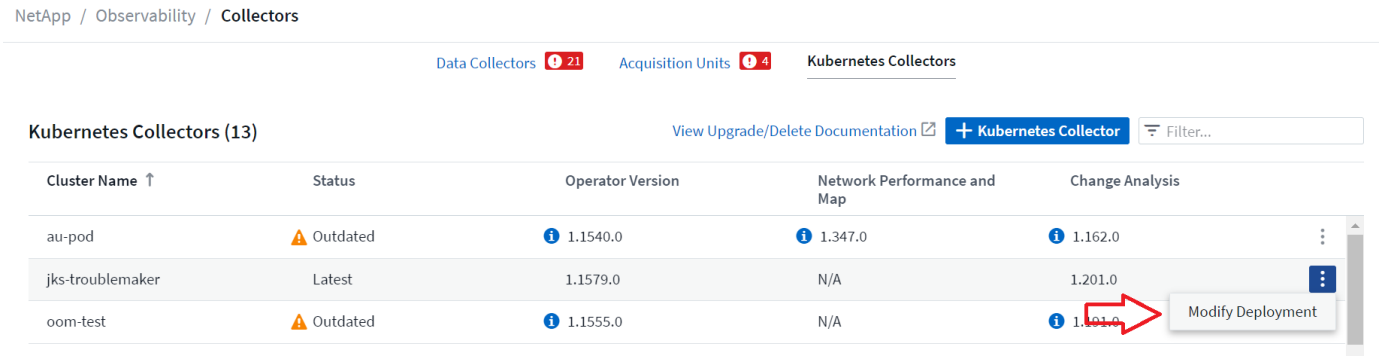
Wenn Sie ein benutzerdefiniertes Repository haben, lesen Sie überVerwenden eines benutzerdefinierten/privaten Docker-Repositorys .

Kubernetes-Überwachungskomponenten

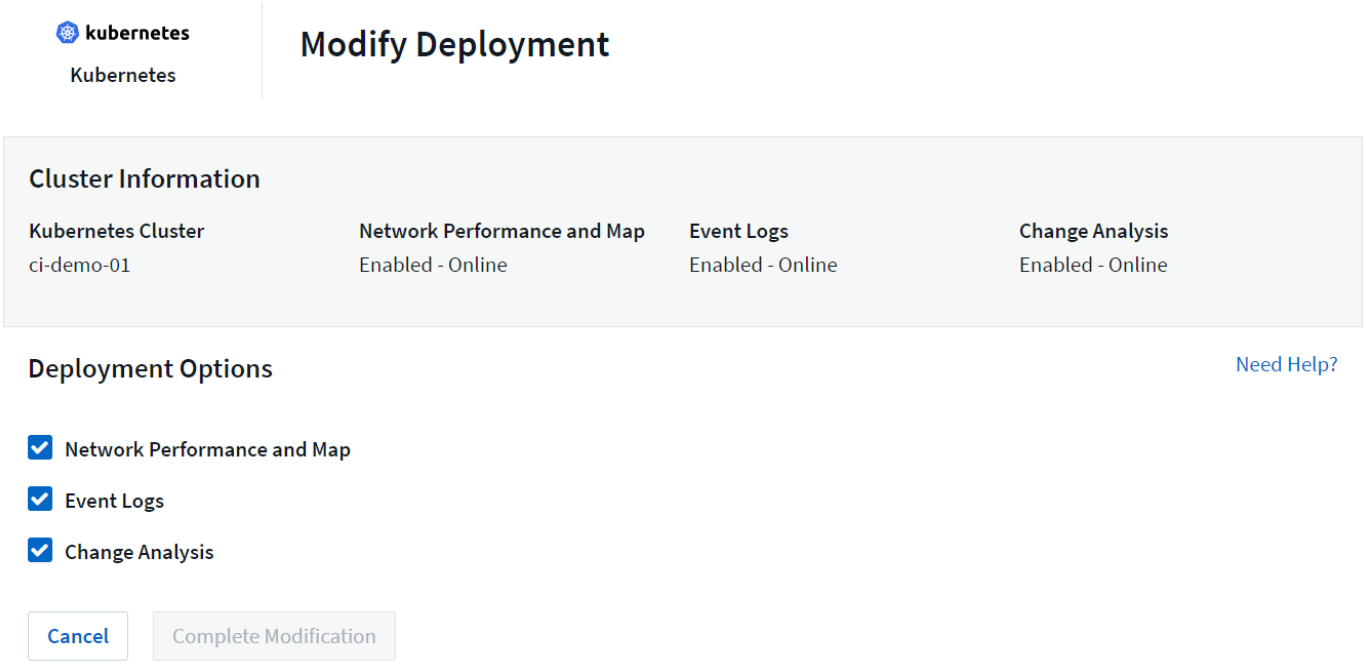
Data Infrastructure Insights Kubernetes Monitoring besteht aus vier Überwachungskomponenten:

- Clustermetriken
- Netzwerkleistung und Karte (optional)
- Ereignisprotokolle (optional)
- Änderungsanalyse (optional)

Die oben genannten optionalen Komponenten sind standardmäßig für jeden Kubernetes-Collector aktiviert. Wenn Sie entscheiden, dass Sie eine Komponente für einen bestimmten Collector nicht benötigen, können Sie sie deaktivieren, indem Sie zu **Kubernetes > Collectors** navigieren und im Drei-Punkte-Menü des Collectors auf der rechten Bildschirmseite *Bereitstellung ändern* auswählen.



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



Upgrade auf den neuesten Kubernetes Monitoring Operator

DII-Druckknopf-Upgrades

Sie können den Kubernetes Monitoring Operator über die DII Kubernetes Collectors-Seite aktualisieren. Klicken Sie auf das Menü neben dem Cluster, den Sie aktualisieren möchten, und wählen Sie *Upgrade*. Der Betreiber überprüft die Bildsignaturen, erstellt einen Snapshot Ihrer aktuellen Installation und führt das Upgrade durch. Innerhalb weniger Minuten sollte der Status des Operators von „Upgrade läuft“ bis „Neueste“ fortschreiten. Wenn ein Fehler auftritt, können Sie für weitere Einzelheiten den Fehlerstatus auswählen und die Tabelle zur Fehlerbehebung bei Push-Button-Upgrades weiter unten zu Rate ziehen.

Push-Button-Upgrades mit privaten Repositories

Wenn Ihr Operator für die Verwendung eines privaten Repositorys konfiguriert ist, stellen Sie bitte sicher, dass alle zum Ausführen des Operators erforderlichen Bilder und deren Signaturen in Ihrem Repository verfügbar sind. Wenn während des Upgrade-Vorgangs ein Fehler aufgrund fehlender Bilder auftritt, fügen Sie diese einfach zu Ihrem Repository hinzu und versuchen Sie das Upgrade erneut. Um die Bildsignaturen in Ihr Repository hochzuladen, verwenden Sie bitte das Cosign-Tool wie folgt und stellen Sie sicher, dass Sie Signaturen für alle unter 3 angegebenen Bilder hochladen. Optional: Laden Sie die Operatorbilder in Ihr privates Repository hoch > Image Pull Snippet

```
cosign copy example.com/src:v1 example.com/dest:v1
#Example
cosign copy <DII container registry>/netapp-monitoring:<image version>
<private repository>/netapp-monitoring:<image version>
```

Rollback auf eine zuvor ausgeführte Version

Wenn Sie das Upgrade mithilfe der Funktion „Upgrade per Knopfdruck“ durchgeführt haben und innerhalb von sieben Tagen nach dem Upgrade Probleme mit der aktuellen Version des Operators auftreten, können Sie mithilfe des während des Upgrade-Vorgangs erstellten Snapshots ein Downgrade auf die zuvor ausgeführte

Version durchführen. Klicken Sie auf das Menü neben dem Cluster, für den Sie ein Rollback durchführen möchten, und wählen Sie *Rollback* aus.

Manuelle Upgrades

Bestimmen Sie, ob eine *AgentConfiguration* mit dem vorhandenen Operator existiert (wenn Ihr Namespace nicht der Standard-*netapp-monitoring* ist, ersetzen Sie ihn durch den entsprechenden Namespace):

```
 kubectl -n netapp-monitoring get agentconfiguration netapp-ci-monitoring-
configuration
Wenn eine _AgentConfiguration_ existiert:
```

- Installierender neueste Operator über den vorhandenen Operator.
  - Stellen Sie sicher, dass SieAbrufen der neuesten Container-Images wenn Sie ein benutzerdefiniertes Repository verwenden.

Falls die *AgentConfiguration* nicht existiert:

- Notieren Sie sich den von Data Infrastructure Insights erkannten Clusternamen (wenn Ihr Namespace nicht der Standardnamespace „netapp-monitoring“ ist, ersetzen Sie ihn durch den entsprechenden Namespace):

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

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

  - Verwenden Sie denselben Clusternamen.
  - Nach dem Herunterladen der neuesten Operator-YAML-Dateien portieren Sie alle in *agent_backup.yaml* gefundenen Anpassungen in die heruntergeladene *operator-config.yaml*, bevor Sie bereitstellen.
  - Stellen Sie sicher, dass SieAbrufen der neuesten Container-Images wenn Sie ein benutzerdefiniertes Repository verwenden.

Stoppen und Starten des Kubernetes-Überwachungsoperators

So stoppen 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

So entfernen Sie den Kubernetes Monitoring Operator

Beachten Sie, dass der Standardnamespace 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 Monitoring-Operators 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>
Hinweis: Wenn der erste Befehl „Keine Ressourcen gefunden“ zurückgibt,
befolgen Sie die folgenden Anweisungen, um ältere Versionen des
Überwachungsoperators zu deinstallieren.
```

Führen Sie die folgenden Befehle der Reihe nach aus. Abhängig von Ihrer aktuellen Installation können einige dieser Befehle die Meldung „Objekt nicht gefunden“ zurückgeben. Diese Nachrichten können bedenkenlos 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 zu Kube-State-Metrics finden Sie unter"diese Seite" .

Konfigurieren/Anpassen des Operators

Diese Abschnitte enthalten Informationen zum Anpassen Ihrer Operatorkonfiguration, zum Arbeiten mit Proxy, zum Verwenden eines benutzerdefinierten oder privaten Docker-Repositorys oder zum Arbeiten mit OpenShift.

Konfigurationsoptionen

Die am häufigsten geänderten Einstellungen können in der benutzerdefinierten Ressource *AgentConfiguration* konfiguriert werden. Sie können diese Ressource vor der Bereitstellung des Operators bearbeiten, indem Sie die Datei *operator-config.yaml* bearbeiten. Diese Datei enthält auskommentierte Beispiele für Einstellungen. Siehe die Liste der"Verfügbare Einstellungen" für die neueste 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 Ihre 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 (Nicht gefunden)“ angezeigt wird, muss Ihr Operator aktualisiert werden, bevor Sie die Agentenkonfiguration verwenden können.
```

Konfigurieren der Proxy-Unterstützung

Es gibt zwei Stellen, an denen Sie einen Proxy auf Ihrem Mandanten verwenden können, um den Kubernetes Monitoring Operator zu installieren. Dabei kann es sich um dasselbe oder um separate Proxy-Systeme handeln:

- Proxy, der während der Ausführung des Installationscode-Snippets (mit „curl“) benötigt wird, um das System, auf dem das Snippet ausgeführt wird, mit Ihrer Data Infrastructure Insights -Umgebung zu verbinden
- Proxy, der vom Ziel-Kubernetes-Cluster zur Kommunikation mit Ihrer Data Infrastructure Insights -Umgebung benötigt wird

Wenn Sie für einen oder beide einen Proxy 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/der Sie den Operator installieren möchten, auf Data Infrastructure Insights zugreifen können, ist Ihr Proxy wahrscheinlich richtig konfiguriert.

Legen Sie für den Proxy, der zur Installation des Kubernetes Operating Monitor verwendet wird, vor der Installation des Operators die Umgebungsvariablen *http_proxy/https_proxy* fest. Für einige Proxy-Umgebungen müssen Sie möglicherweise auch die Umgebungsvariable *no_proxy* festlegen.

Um die Variable(n) festzulegen, führen Sie **vor** der Installation des Kubernetes Monitoring Operator die folgenden Schritte auf Ihrem System aus:

1. Legen Sie die Umgebungsvariable(n) *https_proxy* und/oder *http_proxy* für den aktuellen Benutzer fest:
   a. Wenn der einzurichtende Proxy keine Authentifizierung (Benutzername/Passwort) hat, führen Sie den folgenden Befehl aus:

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

      ```
      export
      http_proxy=<proxy_username>:<proxy_password>@<proxy_server>:<proxy_port>
      ```

Damit der für Ihren Kubernetes-Cluster verwendete Proxy mit Ihrer Data Infrastructure Insights -Umgebung kommunizieren kann, installieren Sie nach dem Lesen aller dieser Anweisungen den Kubernetes Monitoring Operator.

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 ruft der Kubernetes Monitoring Operator Container-Images aus dem Data Infrastructure Insights Repository ab. Wenn Sie einen Kubernetes-Cluster als Ziel für die Überwachung verwenden und dieser Cluster so konfiguriert ist, dass er nur Container-Images aus einem benutzerdefinierten oder privaten Docker-Repository oder Container-Register abruft, müssen Sie den Zugriff auf die vom Kubernetes Monitoring Operator benötigten Container konfigurieren.

Führen Sie das „Image Pull Snippet“ aus der Installationskachel des NetApp Monitoring Operator aus. Mit diesem Befehl melden Sie sich beim Data Infrastructure Insights -Repository an, rufen alle Bildabhängigkeiten für den Operator ab und melden sich vom Data Infrastructure Insights -Repository ab. Geben Sie bei der entsprechenden Aufforderung das bereitgestellte temporäre Repository-Passwort ein. Dieser Befehl lädt alle vom Bediener verwendeten Bilder herunter, auch für optionale Funktionen. Unten sehen Sie, für welche Funktionen diese Bilder verwendet werden.

Kernoperator-Funktionalität und Kubernetes-Überwachung

- NetApp-Überwachung
- ci-kube-rbac-proxy
- ci-ksm
- ci-telegraf
- Distroless-Root-Benutzer

Ereignisprotokoll

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

Netzwerkleistung und Karte

- ci-net-observer

Übertragen Sie das Operator-Docker-Image gemäß Ihren Unternehmensrichtlinien in Ihr privates/lokales/Unternehmens-Docker-Repository. Stellen Sie sicher, dass die Bild-Tags und Verzeichnispfade zu diesen Bildern in Ihrem Repository mit denen im Data Infrastructure Insights -Repository übereinstimmen.

Bearbeiten Sie die Bereitstellung des Überwachungsoperators in operator-deployment.yaml und ändern Sie alle Bildreferenzen, 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 den neuen Speicherort des Docker-Repositorys anzugeben. 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
```

API access token für langfristige Passwörter

In manchen Umgebungen (z. B. Proxy-Repositories) sind Langzeitpasswörter für das Data Infrastructure Insights docker repository erforderlich. Das bei der Installation in der Benutzeroberfläche angegebene Passwort ist nur 24 Stunden gültig. Stattdessen kann man ein API Access Token als Passwort für das docker repository verwenden. Dieses Passwort ist so lange gültig, wie das API Access Token gültig ist. Man kann ein neues API Access Token speziell für diesen Zweck generieren oder ein bereits vorhandenes verwenden.

"Hier lesen" Anweisungen zum Erstellen eines neuen API Access Token.

Um ein vorhandenes API Access Token aus einer heruntergeladenen *operator-secrets.yaml* Datei zu extrahieren, können Benutzer Folgendes ausführen:

```
grep '\.dockerconfigjson' operator-secrets.yaml |sed 's/.*\.dockerconfigjson:
//g' |base64 -d |jq
```

Um ein vorhandenes API Access Token aus einer laufenden Operator-Installation zu extrahieren, können Benutzer Folgendes ausführen:

```
kubectl -n netapp-monitoring get secret netapp-ci-docker -o
jsonpath='{.data.\.dockerconfigjson}' |base64 -d |jq
```

OpenShift-Anweisungen

Wenn Sie OpenShift 4.6 oder höher verwenden, 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 implementiert möglicherweise eine zusätzliche Sicherheitsebene, die den Zugriff auf einige Kubernetes-Komponenten blockieren kann.

Toleranzen und Makel

Die DaemonSets *netapp-ci-telegraf-ds*, *netapp-ci-fluent-bit-ds* und *netapp-ci-net-observer-l4-ds* müssen auf jedem Knoten in Ihrem Cluster einen Pod planen, um Daten auf allen Knoten korrekt zu erfassen. Der Operator wurde so konfiguriert, dass er einige bekannte **Verunreinigungen** toleriert. Wenn Sie benutzerdefinierte Taints auf Ihren Knoten konfiguriert haben und dadurch verhindern, dass Pods auf jedem Knoten ausgeführt werden, können Sie eine **Toleranz** für diese Taints erstellen."in der *AgentConfiguration*" . Wenn Sie benutzerdefinierte Taints auf alle Knoten in Ihrem Cluster angewendet haben, müssen Sie der Operatorbereitstellung auch die erforderlichen Toleranzen hinzufügen, damit der Operator-Pod geplant und ausgeführt werden kann.

Mehr über Kubernetes erfahren"Makel und Duldungen" .

Zurück zum"* Seite „NetApp Kubernetes Monitoring Operator Installation“*"

Eine Anmerkung zu Geheimnissen

Um dem Kubernetes Monitoring Operator die Berechtigung zum Anzeigen von Geheimnissen im gesamten Cluster zu entziehen, löschen Sie vor der Installation die folgenden Ressourcen aus der Datei *operator-setup.yaml*:

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

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 Datei *AgentConfiguration* oder *operator-config.yaml*, um den Abschnitt zur Änderungsverwaltung zu kommentieren und *kindsToIgnoreFromWatch: '"secrets"'* in den Abschnitt zur Änderungsverwaltung aufzunehmen. Beachten Sie 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 Bildsignaturen des Kubernetes-Überwachungsoperators

Das Image für den Operator und alle zugehörigen Images, die er bereitstellt, sind von NetApp signiert. Sie können die Images vor der Installation manuell mit dem Cosign-Tool überprüfen oder einen Kubernetes-Zulassungscontroller konfigurieren. Weitere Einzelheiten finden Sie in der"Kubernetes-Dokumentation" .

Der öffentliche Schlüssel, der zum Überprüfen der Bildsignaturen verwendet wird, ist in der Installationskachel des Überwachungsoperators unter *Optional: Laden Sie die Operatorbilder in Ihr privates Repository hoch > Öffentlicher Schlüssel der Bildsignatur* verfügbar.

Um eine Bildsignatur manuell zu überprüfen, führen Sie die folgenden Schritte aus:

1. Kopieren und führen Sie das Image Pull Snippet aus
2. Kopieren Sie das Repository-Passwort und geben Sie es ein, wenn Sie dazu aufgefordert werden.
3. Speichern Sie den öffentlichen Schlüssel der Bildsignatur (dii-image-signing.pub im Beispiel).
4. Überprüfen Sie die Bilder mit Cosign. Siehe 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

Wenn beim Einrichten des Kubernetes Monitoring Operators Probleme auftreten, können Sie Folgendes versuchen:

Problem:	Versuchen Sie Folgendes:
Ich sehe keinen Hyperlink/keine Verbindung zwischen meinem Kubernetes Persistent Volume und dem entsprechenden Back-End-Speichergerät. Mein Kubernetes Persistent Volume wird mit dem Hostnamen des Speicherservers konfiguriert.	Befolgen Sie die Schritte zum Deinstallieren des vorhandenen Telegraf-Agenten und installieren Sie anschließend den neuesten Telegraf-Agenten neu. Sie müssen Telegraf Version 2.0 oder höher verwenden und Ihr Kubernetes-Clusterspeicher muss aktiv von Data Infrastructure Insights überwacht werden.

Problem:	Versuchen Sie Folgendes:
Ich sehe in den Protokollen Meldungen, die den folgenden ähneln: E0901 15:21:39.962145 1 reflector.go:178] k8s.io/kube-state-metrics/internal/store/builder.go:352: *v1.MutatingWebhookConfiguration konnte nicht aufgelistet werden: Der Server konnte die angeforderte Ressource nicht finden. E0901 15:21:43.168161 1 reflector.go:178] k8s.io/kube-state-metrics/internal/store/builder.go:352: *v1.Lease konnte nicht aufgelistet werden: Der Server konnte die angeforderte Ressource nicht finden (get leases.coordination.k8s.io) usw.	Diese Meldungen können auftreten, wenn Sie kube-state-metrics Version 2.0.0 oder höher mit Kubernetes-Versionen unter 1.20 ausführen. So erhalten Sie die Kubernetes-Version: *kubectl version* So erhalten Sie die kube-state-metrics-Version: *kubectl get deploy/kube-state-metrics -o jsonpath='{..image}'* Um diese Meldungen zu verhindern, können Benutzer ihre kube-state-metrics-Bereitstellung ändern, um die folgenden Leases zu deaktivieren: *mutatingwebhookconfigurations validatingwebhookconfigurations volumeattachments resources* Genauer gesagt können sie das folgende CLI-Argument verwenden: resources=certificatesigningrequests,configmaps,cronjobs,daemonsets, deployments,endpoints,horizontalpodautoscalers,ingresses,jobs,limitranges, namespaces,networkpolicies,nodes,persistentvolumeclaims,persistentvolumes, poddisruptionbudgets,pods,replicasets,replicationcontrollers,resourcequotas, secrets,services,statefulsets,storageclasses Die Standardressourcenliste ist: „Zertifikatsignaturanforderungen, Konfigurationszuordnungen, Cronjobs, Daemonsets, Bereitstellungen, Endpunkte, horizontale Pod-Autoskalierer, Ingresses, Jobs, Leases, Grenzwertbereiche, mutierende Webhookkonfigurationen, Namespaces, Netzwerkrichtlinien, Knoten, persistente Volumeansprüche, persistente Volumes, Pod-Unterbrechungsbudgets, Pods, Replikatsets, Replikationscontroller, Ressourcenkontingente, Geheimnisse, Dienste, Statefulsets, Speicherklassen, validierende Webhookkonfigurationen, Volumeanhänge“

Problem:	Versuchen Sie Folgendes:
Ich sehe Fehlermeldungen von Telegraf, die den folgenden ähneln, aber Telegraf wird gestartet und ausgeführt: 11. Okt. 14:23:41 ip-172-31-39-47 systemd[1]: Der Plugin-gesteuerte Server-Agent zum Melden von Metriken in InfluxDB wurde gestartet. 11. Okt. 14:23:41 ip-172-31-39-47 telegraf[1827]: time="2021-10-11T14:23:41Z" level=error msg="Cache-Verzeichnis konnte nicht erstellt werden. /etc/telegraf/.cache/snowflake, err: mkdir /etc/telegraf/.ca che: Zugriff verweigert. Ignoriert\n" func="gosnowflake.(*defaultLogger).Errorf" file="log.go:120" 11. Okt. 14:23:41 ip-172-31-39-47 telegraf[1827]: time="2021-10-11T14:23:41Z" level=error msg="Öffnen fehlgeschlagen. Ignoriert. Öffnen Sie /etc/telegraf/.cache/snowflake/ocsp_response_cache.json: keine solche Datei oder kein solches Verzeichnis\n" func="gosnowflake.(*defaultLogger).Errorf" file="log.go:120" 11. Okt. 14:23:41 ip-172-31-39-47 telegraf[1827]: 2021-10-11T14:23:41Z I! Telegraf 1.19.3 wird gestartet	Dies ist ein bekanntes Problem. Siehe"Dieser GitHub-Artikel" für weitere Details. Solange Telegraf läuft, können Benutzer diese Fehlermeldungen ignorieren.
Auf Kubernetes melden meine Telegraf-Pods den folgenden Fehler: „Fehler beim Verarbeiten der Mountstats-Informationen: Mountstats-Datei konnte nicht geöffnet werden: /hostfs/proc/1/mountstats, Fehler: Öffnen von /hostfs/proc/1/mountstats: Berechtigung verweigert“	Wenn SELinux aktiviert ist und erzwungen wird, verhindert es wahrscheinlich, dass die Telegraf-Pods auf die Datei /proc/1/mountstats auf dem Kubernetes-Knoten zugreifen können. Um diese Einschränkung zu umgehen, bearbeiten Sie die Agentenkonfiguration und aktivieren Sie die Einstellung „runPrivileged“. Weitere Einzelheiten finden Sie in den OpenShift-Anweisungen.
Auf Kubernetes meldet mein Telegraf ReplicaSet-Pod den folgenden Fehler: [inputs.prometheus] Fehler im Plugin: Schlüsselpaar /etc/kubernetes/pki/etcd/server.crt konnte nicht geladen werden:/etc/kubernetes/pki/etcd/server.key: öffne /etc/kubernetes/pki/etcd/server.crt: keine solche Datei oder kein solches Verzeichnis	Der Telegraf ReplicaSet-Pod soll auf einem Knoten ausgeführt werden, der als Master oder für etcd bestimmt ist. Wenn der ReplicaSet-Pod auf einem dieser Knoten nicht ausgeführt wird, werden diese Fehler angezeigt. Überprüfen Sie, ob Ihre Master-/etcd-Knoten Verunreinigungen aufweisen. Wenn dies der Fall ist, fügen Sie die erforderlichen Toleranzen zum Telegraf ReplicaSet, telegraf-rs, hinzu. Bearbeiten Sie beispielsweise das ReplicaSet … kubectl edit rs telegraf-rs … und fügen Sie der Spezifikation die entsprechenden Toleranzen hinzu. Starten Sie dann den ReplicaSet-Pod neu.

Problem:	Versuchen Sie Folgendes:
Ich habe eine PSP/PSA-Umgebung. Betrifft dies meinen Überwachungsbetreiber?	Wenn Ihr Kubernetes-Cluster mit Pod Security Policy (PSP) oder Pod Security Admission (PSA) ausgeführt wird, müssen Sie auf den neuesten Kubernetes Monitoring Operator aktualisieren. Befolgen Sie diese Schritte, um auf den aktuellen Operator mit Unterstützung für PSP/PSA zu aktualisieren: 1. Deinstallieren der vorherige Überwachungsoperator: kubectl delete agent 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-cluster-admin-rolebinding 2. Installieren die neueste Version des Überwachungsoperators.
Beim Versuch, den Operator bereitzustellen, sind mir Probleme begegnet, und ich verwende PSP/PSA.	1. Bearbeiten Sie den Agenten mit dem folgenden Befehl: kubectl -n <name-space> edit agent 2. Markieren Sie „security-policy-enabled“ als „false“. Dadurch werden die Pod-Sicherheitsrichtlinien und die Pod-Sicherheitszulassung deaktiviert und dem Operator die Bereitstellung ermöglicht. Bestätigen Sie mit den folgenden Befehlen: kubectl get psp (sollte anzeigen, dass die Pod-Sicherheitsrichtlinie entfernt wurde) kubectl get all -n <namespace>
grep -i psp (sollte anzeigen, dass nichts gefunden wurde)	„ImagePullBackoff“-Fehler aufgetreten
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 es ordnungsgemäß erkannt wird. Mehr lesen Informationen zur Konfiguration für benutzerdefinierte/private Repos.	Ich habe ein Problem mit der Bereitstellung meines Überwachungsoperators und die aktuelle Dokumentation hilft mir nicht bei der Lösung.

Problem:	Versuchen Sie Folgendes:

Erfassen oder notieren Sie die Ausgabe der folgenden Befehle und wenden Sie sich an das technische Supportteam.
 	
`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-Pods (Workload Map) im Operator-Namespace befinden sich in CrashLoopBackOff

Diese Pods entsprechen dem Workload Map-Datensammler für die Netzwerkbeobachtung. Versuchen Sie Folgendes: • Überprüfen Sie die Protokolle eines der Pods, um die Mindestkernelversion zu bestätigen. Beispiel: ---- {"ci-tenant-id":"Ihre Mandanten-ID","collector-cluster":"Ihr K8S-Clustername","environment":"prod","level":"error","msg":"Validierung fehlgeschlagen. Grund: Kernelversion 3.10.0 ist niedriger als die Mindestkernelversion 4.18.0","time":"2022-11-09T08:23:08Z"} ---- • Net-Observer-Pods erfordern mindestens die Linux-Kernelversion 4.18.0. Überprüfen Sie die Kernelversion mit dem Befehl „uname -r“ und stellen Sie sicher, dass sie >= 4.18.0 ist	Pods werden im Operator-Namespace ausgeführt (Standard: Netapp-Monitoring), aber in der Benutzeroberfläche werden keine Daten für die Workload-Map 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-Computer mithilfe des Network Time Protocol (NTP) oder Simple Network Time Protocol (SNTP) zu synchronisieren.	Einige der Net-Observer-Pods im Operator-Namespace befinden sich im Status „Ausstehend“
Net-Observer ist ein DaemonSet 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 die CPU oder den Speicher vorliegt. Stellen Sie sicher, dass im Knoten genügend Speicher und CPU verfügbar sind.	Unmittelbar nach der Installation des Kubernetes Monitoring Operator wird mir in meinen Protokollen Folgendes angezeigt: [inputs.prometheus] Fehler im Plug-In: Fehler beim Senden der 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: no such host

Problem:	Versuchen Sie Folgendes:
Diese Meldung wird normalerweise nur angezeigt, wenn ein neuer Operator installiert wird und der *telegraf-rs*-Pod vor dem *ksm*-Pod aktiv ist. Diese Nachrichten sollten aufhören, sobald alle Pods ausgeführt werden.	Ich sehe keine erfassten Metriken für die in meinem Cluster vorhandenen Kubernetes-CronJobs.
Überprüfen Sie Ihre Kubernetes-Version (d. h. `kubectl version`). Wenn es sich um v1.20.x oder niedriger handelt, ist dies eine erwartete Einschränkung. Die mit dem Kubernetes Monitoring Operator bereitgestellte Version von kube-state-metrics unterstützt nur v1.CronJob. Bei Kubernetes 1.20.x und darunter befindet sich die CronJob-Ressource unter v1beta.CronJob. Aus diesem Grund kann kube-state-metrics die CronJob-Ressource nicht finden.	Nach der Installation des Operators treten die Telegraf-DS-Pods in CrashLoopBackOff ein und die Pod-Protokolle zeigen „su: Authentifizierungsfehler“ an.
Bearbeiten Sie den telegraf-Abschnitt in *AgentConfiguration* und setzen Sie *dockerMetricCollectionEnabled* auf false. Für weitere Details siehe die "Konfigurationsoptionen" des Operators. … spec: … telegraf: … - name: docker run-mode: - DaemonSet substitutions: - key: DOCKER_UNIX_SOCK_PLACEHOLDER value: unix:///run/docker.sock … …	In meinen Telegraf-Protokollen werden immer wieder Fehlermeldungen angezeigt, die den folgenden ähneln: E! [Agent] Fehler beim Schreiben in outputs.http: Post "https://<tenant_url>/rest/v1/lake/ingest/influxdb": Kontextfrist überschritten (Client.Timeout beim Warten auf Header überschritten)
Bearbeiten Sie den Telegraf-Abschnitt in *AgentConfiguration* und erhöhen Sie *outputTimeout* auf 10 s. Weitere Einzelheiten finden Sie in der Betriebsanleitung des Betreibers."Konfigurationsoptionen" .	Mir fehlen *involvedobject*-Daten für einige Ereignisprotokolle.
Stellen Sie sicher, dass Sie die Schritte in der"Berechtigungen" Abschnitt oben.	Warum werden zwei Überwachungsoperator-Pods ausgeführt, einer mit dem Namen netapp-ci-monitoring-operator-<pod> und der andere mit dem Namen monitoring-operator-<pod>?
Ab dem 12. Oktober 2023 hat Data Infrastructure Insights den Operator überarbeitet, um unseren Benutzern einen besseren Service zu bieten. Damit diese Änderungen vollständig übernommen werden können, müssen SieEntfernen Sie den alten Operator Undinstallieren Sie die neue .	Meine Kubernetes-Ereignisse wurden unerwartet nicht mehr an Data Infrastructure Insights gemeldet.
Rufen Sie den Namen des Event-Exporter-Pods ab: `` `kubectl -n netapp-monitoring get pods ``	grep event-exporter

Problem:	Versuchen Sie Folgendes:
awk '{print $1}'	sed 's/event-exporter./event-exporter/'` Es sollte entweder „netapp-ci-event-exporter“ oder „event-exporter“ sein. Bearbeiten Sie als Nächstes den Überwachungsagenten `kubectl -n netapp-monitoring edit agent` und legen Sie den Wert für LOG_FILE so fest, dass er den entsprechenden Event-Exporter-Pod-Namen widerspiegelt, der im vorherigen Schritt gefunden wurde. Genauer gesagt sollte LOG_FILE entweder auf "/var/log/containers/netapp-ci-event-exporter.log" oder "/var/log/containers/event-exporter*.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 auchdeinstallieren UndNeuinstallation der Agent.
Ich sehe, dass vom Kubernetes Monitoring Operator bereitgestellte Pods aufgrund unzureichender Ressourcen abstürzen.	Siehe den Kubernetes Monitoring Operator"Konfigurationsoptionen" um die CPU- und/oder Speichergrenzen nach Bedarf zu erhöhen.
Ein fehlendes Image oder eine ungültige Konfiguration führte dazu, dass die netapp-ci-kube-state-metrics-Pods nicht gestartet werden konnten oder nicht bereit waren. Jetzt steckt das StatefulSet fest und Konfigurationsänderungen werden nicht auf die Netapp-CI-Kube-State-Metrics-Pods angewendet.	Das StatefulSet ist in einem"gebrochen" Zustand. Nachdem Sie alle Konfigurationsprobleme behoben haben, führen Sie einen Bounce der Netapp-CI-Kube-State-Metrics-Pods durch.
netapp-ci-kube-state-metrics-Pods können nach der Ausführung eines Kubernetes Operator-Upgrades nicht gestartet werden und lösen ErrImagePull aus (das Abrufen des Images schlägt fehl).	Versuchen Sie, die Pods manuell zurückzusetzen.
Bei der Protokollanalyse werden für meinen Kubernetes-Cluster Meldungen vom Typ „Ereignis verworfen, da es älter ist als maxEventAgeSeconds“ beobachtet.	Ändern Sie die Operator-Agentenkonfiguration und erhöhen Sie *event-exporter-maxEventAgeSeconds* (z. B. auf 60 s), *event-exporter-kubeQPS* (z. B. auf 100) und *event-exporter-kubeBurst* (z. B. auf 500). Weitere Einzelheiten zu diesen Konfigurationsoptionen finden Sie im"Konfigurationsoptionen" Seite.

Problem:	Versuchen Sie Folgendes:
Telegraf warnt vor unzureichendem sperrbaren Speicher oder stürzt ab.	Versuchen Sie, das Limit des sperrbaren Speichers für Telegraf im zugrunde liegenden Betriebssystem/Knoten zu erhöhen. Wenn eine Erhöhung des Limits keine Option ist, ändern Sie die NKMO-Agentenkonfiguration und setzen Sie *unprotected* auf *true*. Dadurch wird Telegraf angewiesen, keinen Versuch zu unternehmen, gesperrte Speicherseiten zu reservieren. Dies kann zwar ein Sicherheitsrisiko darstellen, da entschlüsselte Geheimnisse möglicherweise auf die Festplatte ausgelagert werden, ermöglicht jedoch die Ausführung in Umgebungen, in denen die Reservierung gesperrten Speichers nicht möglich ist. Weitere Informationen zu den *ungeschützten* Konfigurationsoptionen finden Sie im"Konfigurationsoptionen" Seite.
Ich sehe Warnmeldungen von Telegraf, die etwa wie folgt aussehen: *W! [inputs.diskio] Der Datenträgername für „vdc“ konnte nicht ermittelt werden: Fehler beim Lesen von /dev/vdc: keine solche Datei oder kein solches Verzeichnis*	Für den Kubernetes Monitoring Operator sind diese Warnmeldungen harmlos und können sicher ignoriert werden. Alternativ bearbeiten Sie den telegraf-Abschnitt in AgentConfiguration und setzen Sie *runDsPrivileged* auf true. Weitere Einzelheiten finden Sie unter "Konfigurationsoptionen des Betreibers".

Problem:	Versuchen Sie Folgendes:

Mein Fluent-Bit-Pod schlägt mit den folgenden Fehlern fehl: [2024/10/16 14:16:23] [Fehler] [/src/fluent-bit/plugins/in_tail/tail_fs_inotify.c:360 errno=24] Zu viele offene Dateien [2024/10/16 14:16:23] [Fehler] Initialisierung der Eingabe tail.0 fehlgeschlagen [2024/10/16 14:16:23] [Fehler] [Engine] Initialisierung der Eingabe fehlgeschlagen	Versuchen Sie, Ihre *fsnotify*-Einstellungen in Ihrem Cluster zu ändern:
	`sudo sysctl fs.inotify.max_user_instances (take note of setting)`
	`sudo sysctl fs.inotify.max_user_instances=<something larger than current setting>`
	`sudo sysctl fs.inotify.max_user_watches (take note of setting)`
	`sudo sysctl fs.inotify.max_user_watches=<something larger than current setting>`
	Starten Sie Fluent-bit neu.
	Hinweis: Um diese Einstellungen auch nach einem Neustart des Knotens dauerhaft zu halten, müssen Sie die folgenden Zeilen in */etc/sysctl.conf* einfügen.
	`fs.inotify.max_user_instances=<something larger than current setting>`
	`fs.inotify.max_user_watches=<something larger than current setting>`

Problem:	Versuchen Sie Folgendes:
Die Telegraf DS-Pods melden Fehler im Zusammenhang mit dem Kubernetes-Eingabe-Plugin, das keine HTTP-Anfragen stellen kann, da das TLS-Zertifikat nicht validiert werden kann. Zum Beispiel: E! [inputs.kubernetes] Fehler im Plugin: Fehler beim Senden einer HTTP-Anfrage an"<a href="https://<kubelet_IP>:10250/stats/summary": " class="bare">https://<kubelet_IP>:10250/stats/summary": Erhalten"<a href="https://<kubelet_IP>:10250/stats/summary": " class="bare">https://<kubelet_IP>:10250/stats/summary": tls: Zertifikat konnte nicht überprüft werden: x509: Zertifikat für <kubelet_IP> kann nicht validiert werden, da es keine IP-SANs enthält	Dies tritt auf, wenn das Kubelet selbstsignierte Zertifikate verwendet und/oder das angegebene Zertifikat die <kubelet_IP> nicht in der Liste „Subject Alternative Name“ des Zertifikats enthält. Um dieses Problem zu lösen, kann der Benutzer die"Agentenkonfiguration" , und setzen Sie *telegraf:insecureK8sSkipVerify* auf *true*. Dadurch wird das Telegraf-Eingabe-Plugin so konfiguriert, dass die Überprüfung übersprungen wird. Alternativ kann der Benutzer das Kubelet konfigurieren für"serverTLSBootstrap" , wodurch eine Zertifikatsanforderung von der API „certificates.k8s.io“ ausgelöst wird.
Ich erhalte den folgenden Fehler in den Fluent-bit-Pods und der Pod kann nicht gestartet werden: 026/01/12 20:20:32] [error] [sqldb] error=unable to open database file [2026/01/12 20:20:32] [error] [input:tail:tail.0] db: could not create 'in_tail_files' table [2026/01/12 20:20:32] [error] [input:tail:tail.0] could not open/create database [2026/01/12 20:20:32] [error] failed initialize input tail.0 [2026/01/12 20:20:32] [error] [engine] input initialization failed	Stellen Sie sicher, dass das Hostverzeichnis, in dem sich die DB-Datei befindet, die richtigen lesen/schreiben-Berechtigungen hat. Genauer gesagt sollte das Hostverzeichnis lesen/schreiben-Berechtigungen für Nicht-Root-Benutzer gewähren. Der Standardpfad für die DB-Datei ist /var/log/, sofern er nicht durch die fluent-bit-dbFile *agentconfiguration* -Option überschrieben wird. Wenn SELinux aktiviert ist, versuchen Sie, die fluent-bit-seLinuxOptionsType agentconfiguration-Option auf 'spc_t' zu setzen.

Weitere Informationen finden Sie in der"Support" Seite oder in der"Datensammler-Supportmatrix" .

Memcached-Datensammler

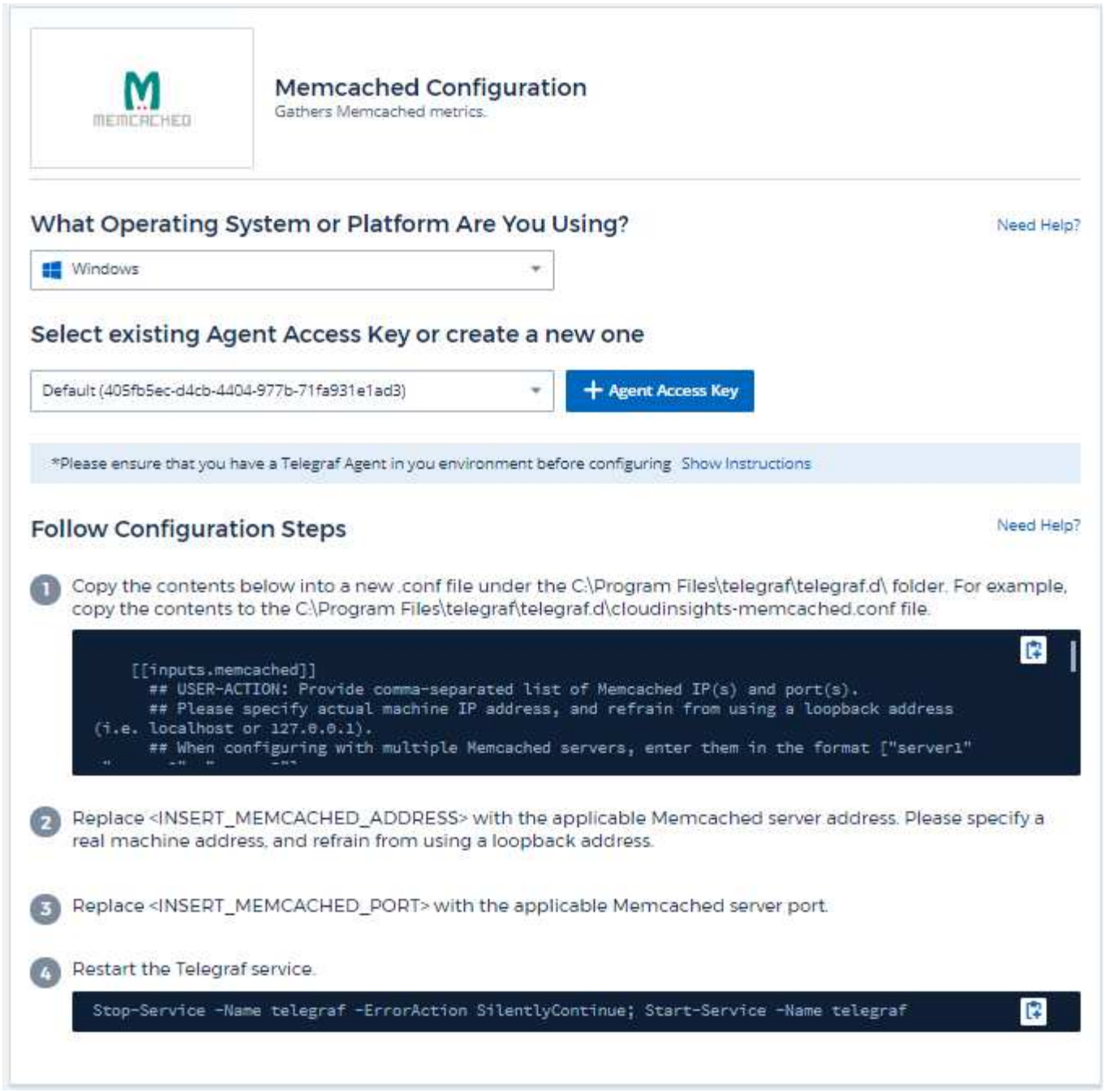
Data Infrastructure Insights verwendet diesen Datensammler, um Metriken von Memcached zu sammeln.

Installation

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

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

2. Wenn Sie noch keinen Agenten für die Sammlung installiert haben oder einen Agenten für ein anderes Betriebssystem oder eine andere Plattform installieren möchten, klicken Sie auf *Anweisungen anzeigen*, um das"Agenteninstallation" Anweisungen.
3. Wählen Sie den Agent-Zugriffsschlüssel zur Verwendung mit diesem Datensammler aus. Sie können einen neuen Agentenzugriffsschlüssel hinzufügen, indem Sie auf die Schaltfläche **+ Agentenzugriffsschlüssel** klicken. Best Practice: Verwenden Sie nur dann einen anderen Agent-Zugriffsschlüssel, wenn Sie Datensammler beispielsweise nach Betriebssystem/Plattform gruppieren möchten.
4. Befolgen Sie die Konfigurationsschritte, um den Datensammler zu konfigurieren. Die Anweisungen variieren je nach Art des Betriebssystems oder der Plattform, die Sie zum Sammeln von Daten verwenden.



Aufstellen

Informationen finden Sie im"Memcached-Wiki" .

Objekte und Zähler

Die folgenden Objekte und ihre Zähler werden gesammelt:

Objekt:	Kennungen:	Merkmale:	Datenpunkte:
Memcached	Namespace-Server	Knoten-IP Knotenname	Verbindungen annehmen Bearbeitete Authentifizierungsanfragen Fehlgeschlagene Authentifizierungen Verwendete Bytes Gelesene Bytes (pro Sek.) Geschriebene Bytes (pro Sek.) CAS Badval CAS-Treffer CAS-Fehlschläge Flush-Anforderungen (pro Sek.) Get-Anforderungen (pro Sek.) Set-Anforderungen (pro Sek.) Touch-Anforderungen (pro Sek.) Verbindungserträge (pro Sek.) Verbindungsstrukturen Offene Verbindungen Aktuell gespeicherte Elemente Decr-Anforderungen Treffer (pro Sek.) Decr-Anforderungen Fehlschläge (pro Sek.) Löschanforderungen Treffer (pro Sek.) Löschanforderungen Fehlschläge (pro Sek.) Ausgewiesene Elemente Gültige Ausweisungen Abgelaufene Elemente Get-Treffer (pro Sek.) Get-Fehlschläge (pro Sek.) Verwendete Hash-Bytes Hash wird erweitert Hash-Leistungsstufe Incr-Anforderungen Treffer (pro Sek.) Incr-Anforderungen Fehlschläge (pro Sek.) Server Max. Bytes Listen Deaktiviert Anzahl zurückgeforderte Worker-Threads Anzahl insgesamt geöffnete Verbindungen

Fehlerbehebung

Weitere Informationen finden Sie in der"Support" Seite.

MongoDB-Datensammler

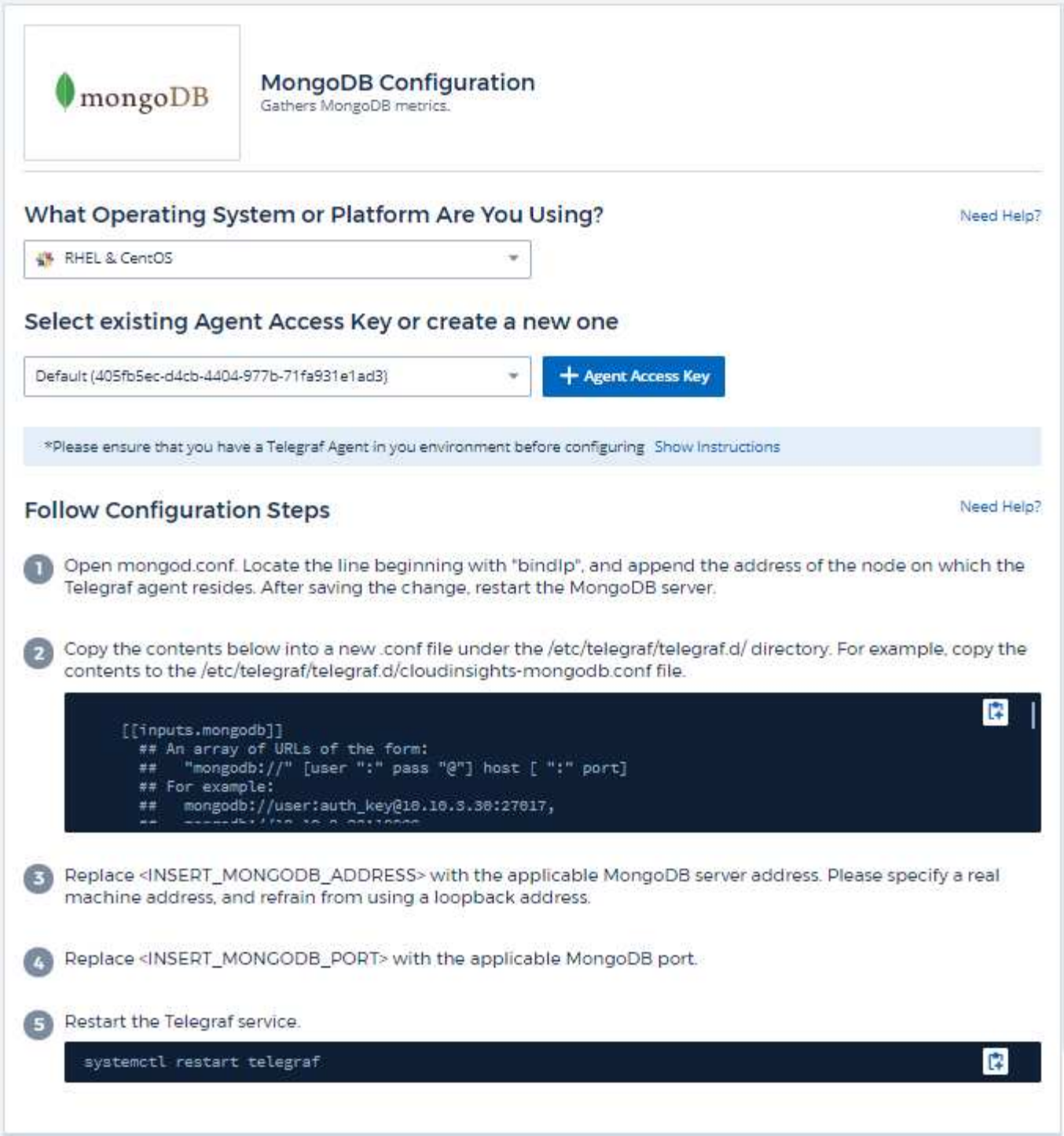
Data Infrastructure Insights verwendet diesen Datensammler, um Metriken von MongoDB zu sammeln.

Installation

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

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

2. Wenn Sie noch keinen Agenten für die Sammlung installiert haben oder einen Agenten für ein anderes Betriebssystem oder eine andere Plattform installieren möchten, klicken Sie auf *Anweisungen anzeigen*, um das"Agenteninstallation" Anweisungen.
3. Wählen Sie den Agent-Zugriffsschlüssel zur Verwendung mit diesem Datensammler aus. Sie können einen neuen Agentenzugriffsschlüssel hinzufügen, indem Sie auf die Schaltfläche **+ Agentenzugriffsschlüssel** klicken. Best Practice: Verwenden Sie nur dann einen anderen Agent-Zugriffsschlüssel, wenn Sie Datensammler beispielsweise nach Betriebssystem/Plattform gruppieren möchten.
4. Befolgen Sie die Konfigurationsschritte, um den Datensammler zu konfigurieren. Die Anweisungen variieren je nach Art des Betriebssystems oder der Plattform, die Sie zum Sammeln von Daten verwenden.



Aufstellen

Informationen finden Sie im"MongoDB-Dokumentation" .

Objekte und Zähler

Die folgenden Objekte und ihre Zähler werden gesammelt:

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

Objekt:	Kennungen:	Merkmale:	Datenpunkte:
MongoDB-Datenbank	Namespace Hostname Datenbankname		

Fehlerbehebung

Informationen finden Sie im"Support" Seite.

MySQL-Datensammler

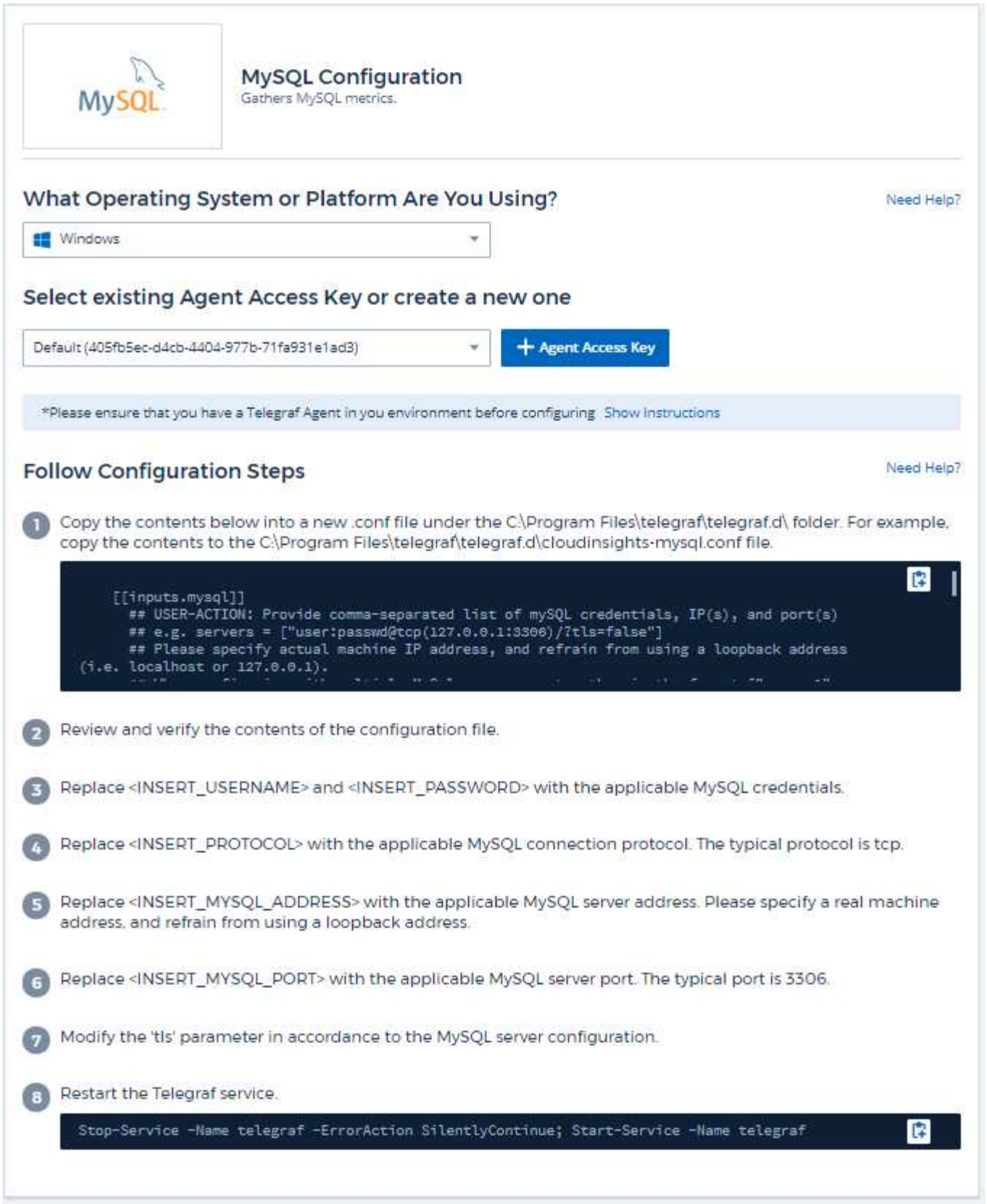
Data Infrastructure Insights verwendet diesen Datensammler, um Metriken aus MySQL zu sammeln.

Installation

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

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

2. Wenn Sie noch keinen Agenten für die Sammlung installiert haben oder einen Agenten für ein anderes Betriebssystem oder eine andere Plattform installieren möchten, klicken Sie auf *Anweisungen anzeigen*, um das"Agenteninstallation" Anweisungen.
3. Wählen Sie den Agent-Zugriffsschlüssel zur Verwendung mit diesem Datensammler aus. Sie können einen neuen Agentenzugriffsschlüssel hinzufügen, indem Sie auf die Schaltfläche **+ Agentenzugriffsschlüssel** klicken. Best Practice: Verwenden Sie nur dann einen anderen Agent-Zugriffsschlüssel, wenn Sie Datensammler beispielsweise nach Betriebssystem/Plattform gruppieren möchten.
4. Befolgen Sie die Konfigurationsschritte, um den Datensammler zu konfigurieren. Die Anweisungen variieren je nach Art des Betriebssystems oder der Plattform, die Sie zum Sammeln von Daten verwenden.



Aufstellen

Informationen finden Sie im"MySQL-Dokumentation" .

Objekte und Zähler

Die folgenden Objekte und ihre Zähler werden gesammelt:

Objekt:	Kennungen:	Merkmale:	Datenpunkte:
MySQL	Namespace MySQL-Server	Knoten-IP Knotenname	Abgebrochene Clients (pro Sek.) Abgebrochene Verbindungen (pro Sek.) RX-Bytes (pro Sek.) TX-Bytes (pro Sek.) Befehle Admin (pro Sek.) Befehle Ereignis ändern Befehle Funktion ändern Befehle Instanz ändern Befehle Prozedur ändern Befehle Server ändern Befehle Tabelle ändern Befehle Tablespace ändern Befehle Benutzer ändern Befehle Analysieren Befehle Zu Keycache zuweisen Befehle Beginnen Befehle Binlog Befehle Prozedur aufrufen Befehle DB ändern Befehle Master ändern Befehle Repl ändern Filterbefehle Prüfen Befehle Prüfsummenbefehle Commit Befehle DB erstellen Befehle Ereignis erstellen Befehle Funktion erstellen Befehle Index erstellen Befehle Prozedur erstellen Befehle Server erstellen Befehle Tabelle erstellen Befehle Trigger erstellen Befehle UDF erstellen Befehle Benutzer erstellen Befehle Ansicht erstellen Befehle Dealloc SQL-Verbindungsfehler Akzeptieren Erstellte temporäre Datenträgertabellen Verzögerte Fehler Befehle leeren Handler Commit InnoDB-Pufferpool Bytes Daten Schlüsselblöcke nicht geleert Schlüsselleseanforderungen Schlüsselschreibanforderungen Schlüsselschreibvorgänge Max. Ausführungszeit überschritten Max. verwendete Verbindungen Geöffnete Dateien

Fehlerbehebung

Weitere Informationen finden Sie in der"Support" Seite.

Netstat-Datensammler

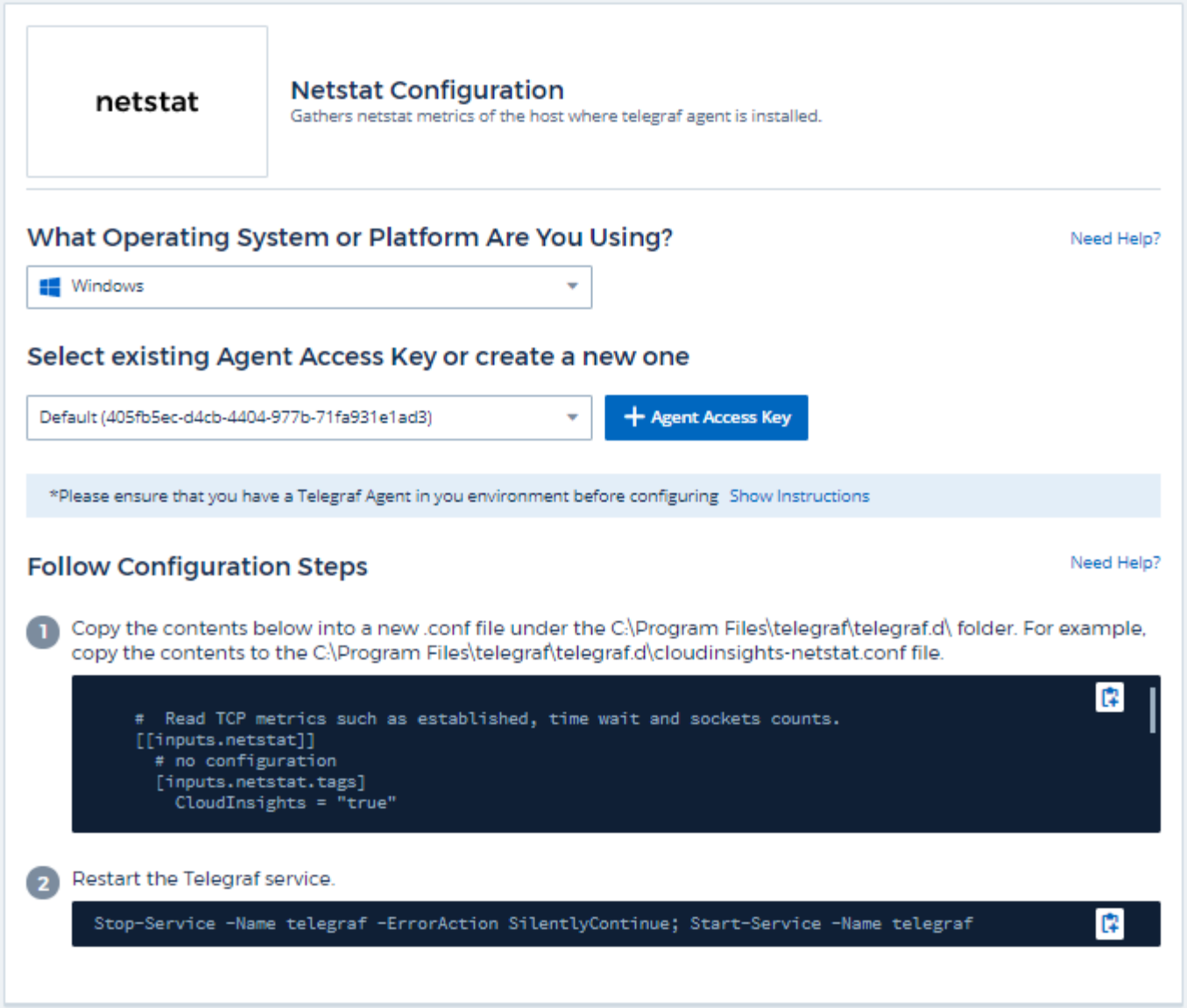
Data Infrastructure Insights verwendet diesen Datensammler, um Netstat-Metriken zu erfassen.

Installation

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

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

2. Wenn Sie noch keinen Agenten für die Sammlung installiert haben oder einen Agenten für ein anderes Betriebssystem oder eine andere Plattform installieren möchten, klicken Sie auf *Anweisungen anzeigen*, um das"Agenteninstallation" Anweisungen.
3. Wählen Sie den Agent-Zugriffsschlüssel zur Verwendung mit diesem Datensammler aus. Sie können einen neuen Agentenzugriffsschlüssel hinzufügen, indem Sie auf die Schaltfläche **+ Agentenzugriffsschlüssel** klicken. Best Practice: Verwenden Sie nur dann einen anderen Agent-Zugriffsschlüssel, wenn Sie Datensammler beispielsweise nach Betriebssystem/Plattform gruppieren möchten.
4. Befolgen Sie die Konfigurationsschritte, um den Datensammler zu konfigurieren. Die Anweisungen variieren je nach Art des Betriebssystems oder der Plattform, die Sie zum Sammeln von Daten verwenden.



Aufstellen

Objekte und Zähler

Die folgenden Objekte und ihre Zähler werden gesammelt:

Objekt:	Kennungen:	Merkmale:	Datenpunkte:
Netstat	Knoten-UUID	Knoten-IP Knotenname	

Fehlerbehebung

Weitere Informationen finden Sie in der"Support" Seite.

Nginx-Datensammler

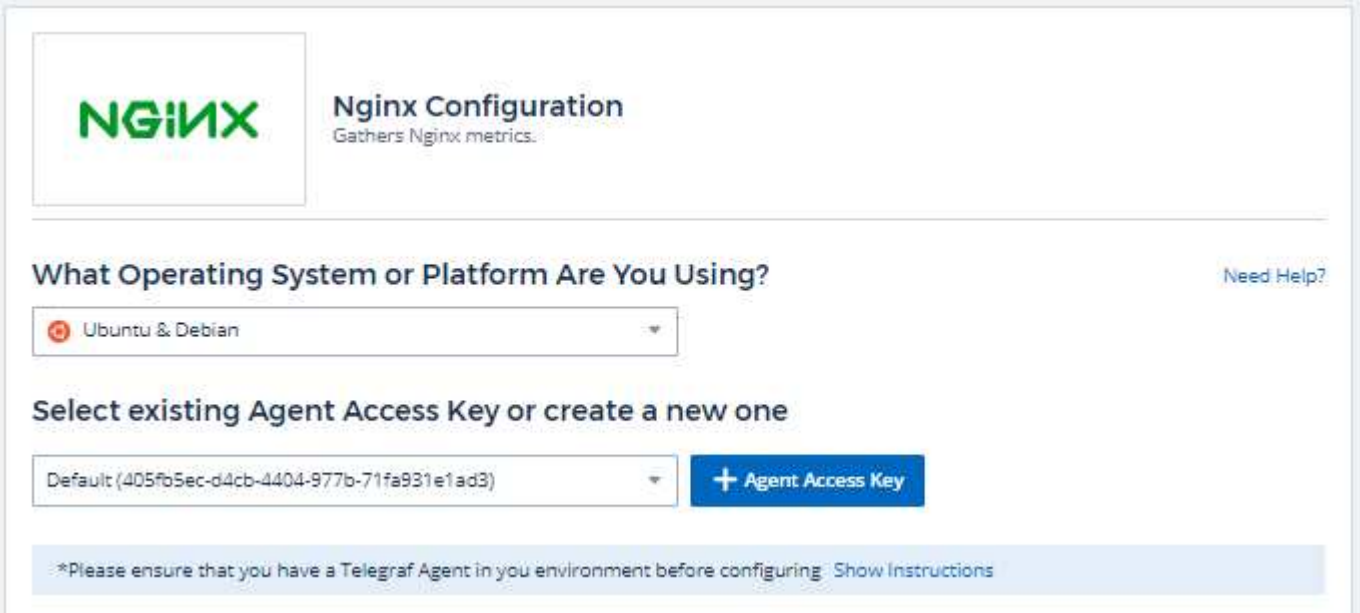
Data Infrastructure Insights verwendet diesen Datensammler, um Metriken von Nginx zu sammeln.

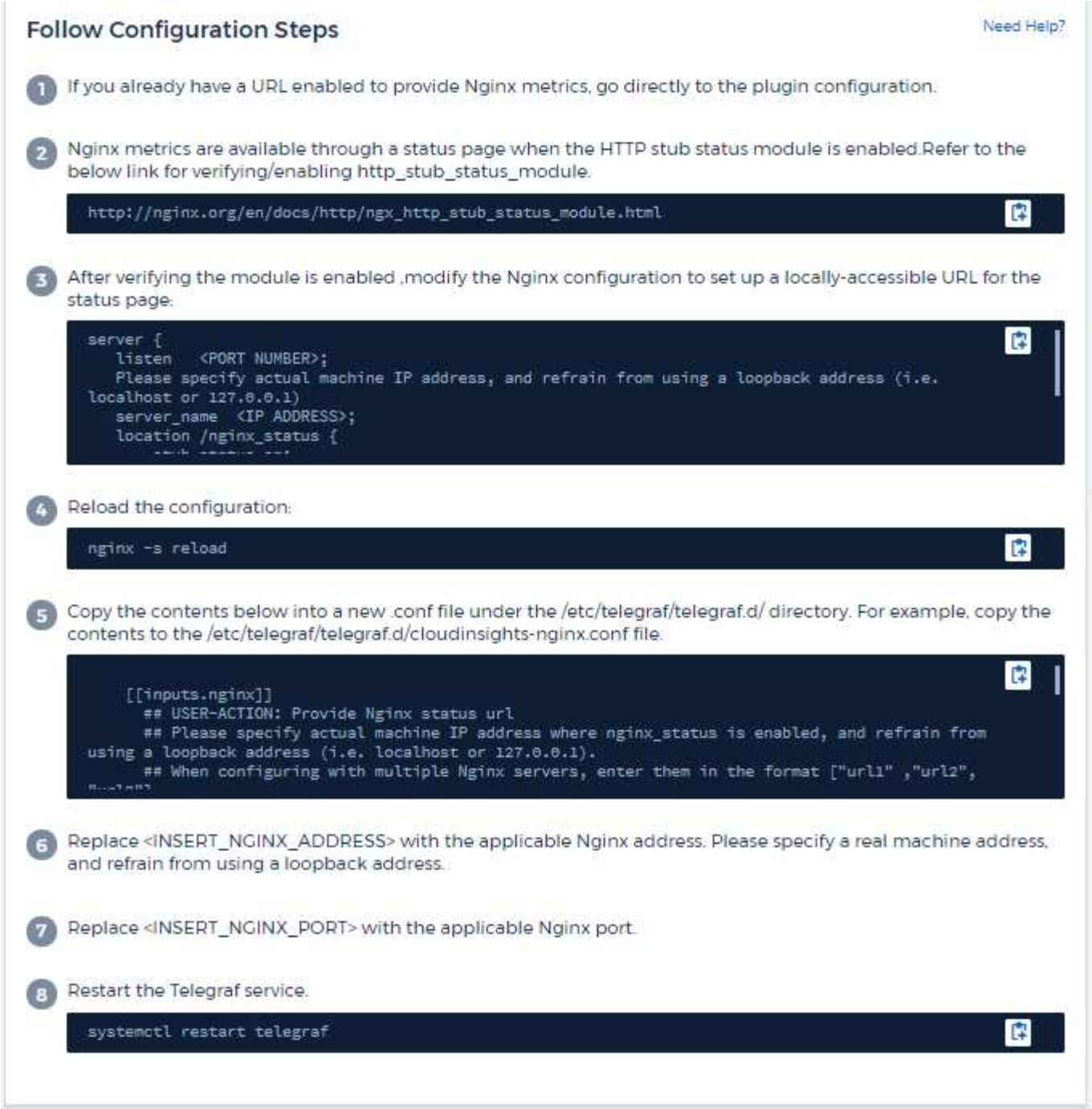
Installation

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

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

2. Wenn Sie noch keinen Agenten für die Sammlung installiert haben oder einen Agenten für ein anderes Betriebssystem oder eine andere Plattform installieren möchten, klicken Sie auf *Anweisungen anzeigen*, um das"Agenteninstallation" Anweisungen.
3. Wählen Sie den Agent-Zugriffsschlüssel zur Verwendung mit diesem Datensammler aus. Sie können einen neuen Agentenzugriffsschlüssel hinzufügen, indem Sie auf die Schaltfläche **+ Agentenzugriffsschlüssel** klicken. Best Practice: Verwenden Sie nur dann einen anderen Agent-Zugriffsschlüssel, wenn Sie Datensammler beispielsweise nach Betriebssystem/Plattform gruppieren möchten.
4. Befolgen Sie die Konfigurationsschritte, um den Datensammler zu konfigurieren. Die Anweisungen variieren je nach Art des Betriebssystems oder der Plattform, die Sie zum Sammeln von Daten verwenden.





Aufstellen

Die Nginx-Metriksammlung erfordert, dass Nginx"http_stub_status_module" aktiviert werden.

Weitere Informationen finden Sie in der"Nginx-Dokumentation" .

Objekte und Zähler

Die folgenden Objekte und ihre Zähler werden gesammelt:

Objekt:	Kennungen:	Merkmale:	Datenpunkte:
Nginx	Namespace-Server	Knoten-IP Knotenname Port	Akzeptiert Aktive bearbeitete Leseanfragen Wartende Schreibanfragen

Fehlerbehebung

Weitere Informationen finden Sie in der"Support" Seite.

PostgreSQL-Datensammler

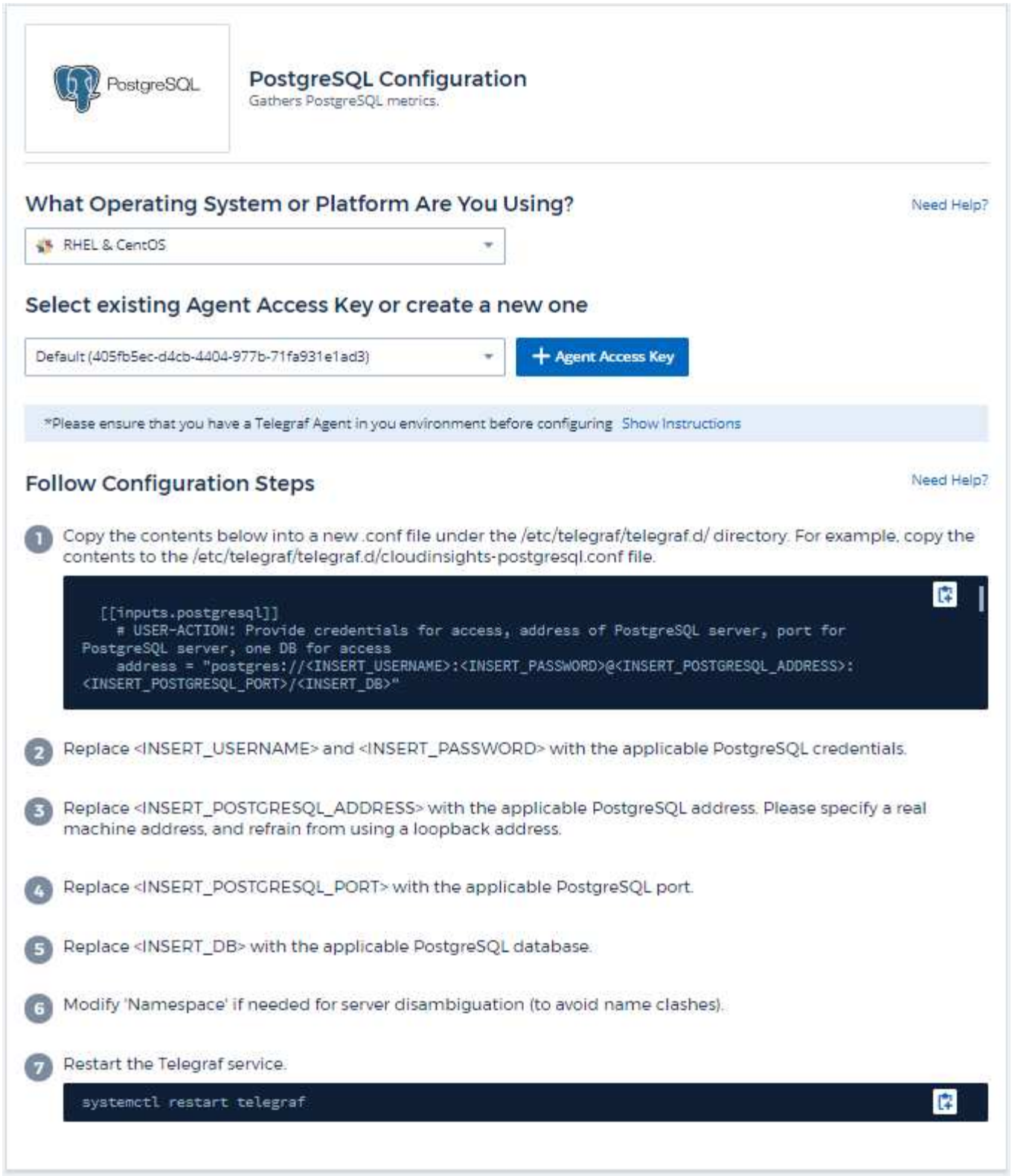
Data Infrastructure Insights verwendet diesen Datensammler, um Metriken von PostgreSQL zu sammeln.

Installation

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

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

2. Wenn Sie noch keinen Agenten für die Sammlung installiert haben oder einen Agenten für ein anderes Betriebssystem oder eine andere Plattform installieren möchten, klicken Sie auf *Anweisungen anzeigen*, um das"Agenteninstallation" Anweisungen.
3. Wählen Sie den Agent-Zugriffsschlüssel zur Verwendung mit diesem Datensammler aus. Sie können einen neuen Agentenzugriffsschlüssel hinzufügen, indem Sie auf die Schaltfläche **+ Agentenzugriffsschlüssel** klicken. Best Practice: Verwenden Sie nur dann einen anderen Agent-Zugriffsschlüssel, wenn Sie Datensammler beispielsweise nach Betriebssystem/Plattform gruppieren möchten.
4. Befolgen Sie die Konfigurationsschritte, um den Datensammler zu konfigurieren. Die Anweisungen variieren je nach Art des Betriebssystems oder der Plattform, die Sie zum Sammeln von Daten verwenden.



Aufstellen

Informationen finden Sie im"PostgreSQL-Dokumentation" .

Objekte und Zähler

Die folgenden Objekte und ihre Zähler werden gesammelt:

Objekt:	Kennungen:	Merkmale:	Datenpunkte:
PostgreSQL-Server	Namespace-Datenbankserver	Knotenname Knoten-IP	Puffer Zugewiesene Puffer Backend-Puffer Backend-Dateisynchronisierungspuffer Checkpoint-Puffer Saubere Checkpoints Synchronisierungszeit Checkpoints Schreibzeit Checkpoints Anfragen Checkpoints Zeitlich begrenzt Max Geschrieben Sauber
PostgreSQL-Datenbank	Namespace-Datenbankserver	Datenbank-OID Knotenname Knoten-IP	Blöcke Lesezeit Blöcke Schreibzeit Blöcke Treffer Blöcke Lesevorgänge Konflikte Deadlocks Client Anzahl Temp. Dateien Bytes Temp. Dateien Anzahl Zeilen Gelöschte Zeilen Abgerufene Zeilen Eingefügte Zeilen Zurückgegebene Zeilen Aktualisierte Transaktionen Festgeschriebene Transaktionen Zurückgesetzte Transaktionen

Fehlerbehebung

Weitere Informationen finden Sie in der"Support" Seite.

Puppet Agent-Datensammler

Data Infrastructure Insights verwendet diesen Datensammler, um Metriken von Puppet Agent zu sammeln.

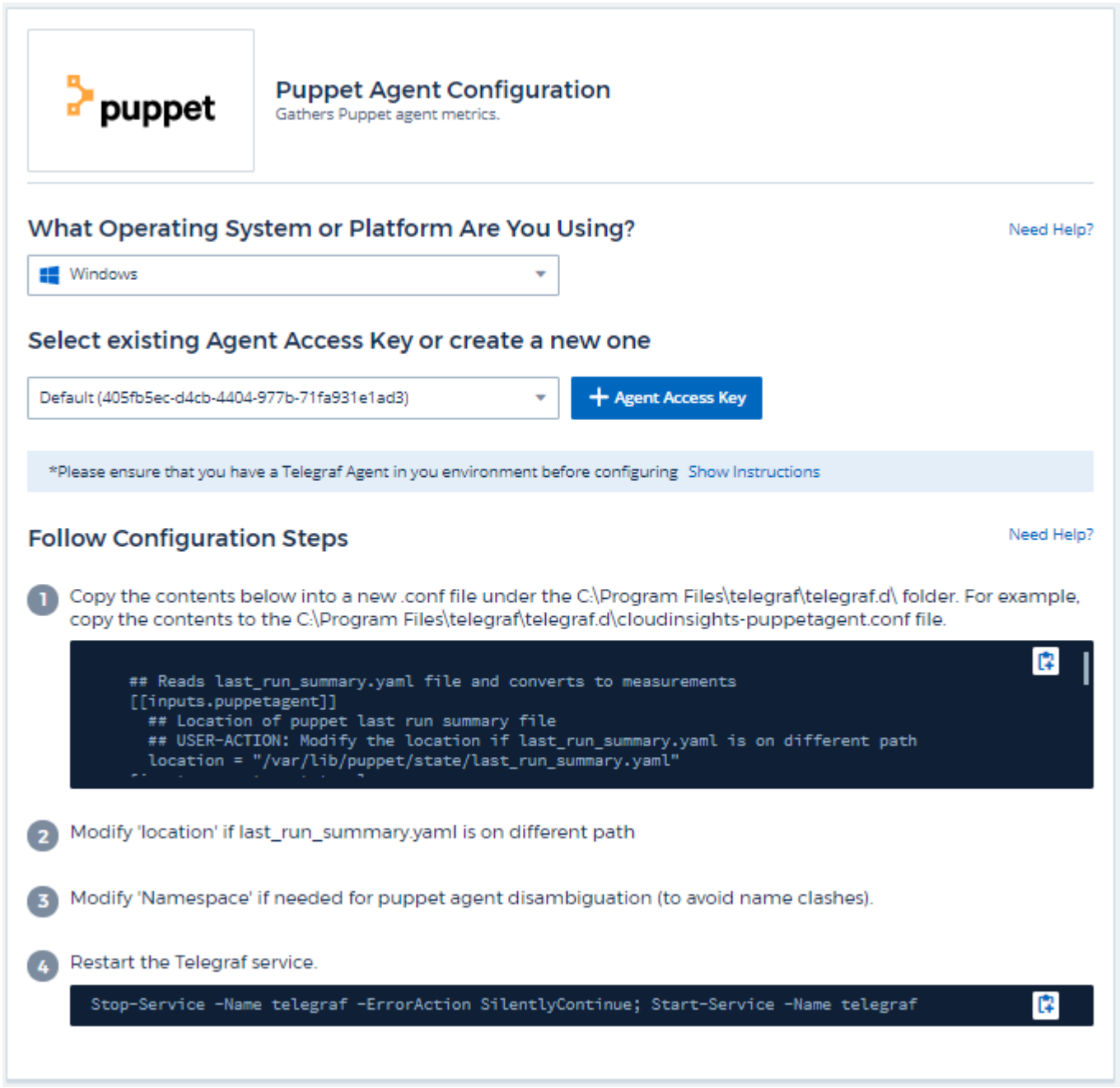
Installation

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

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

2. Wenn Sie noch keinen Agenten für die Sammlung installiert haben oder einen Agenten für ein anderes Betriebssystem oder eine andere Plattform installieren möchten, klicken Sie auf *Anweisungen anzeigen*, um das"Agenteninstallation" Anweisungen.
3. Wählen Sie den Agent-Zugriffsschlüssel zur Verwendung mit diesem Datensammler aus. Sie können einen neuen Agentenzugriffsschlüssel hinzufügen, indem Sie auf die Schaltfläche **+ Agentenzugriffsschlüssel** klicken. Best Practice: Verwenden Sie nur dann einen anderen Agent-Zugriffsschlüssel, wenn Sie Datensammler beispielsweise nach Betriebssystem/Plattform gruppieren möchten.

4. Befolgen Sie die Konfigurationsschritte, um den Datensammler zu konfigurieren. Die Anweisungen variieren je nach Art des Betriebssystems oder der Plattform, die Sie zum Sammeln von Daten verwenden.



Aufstellen

Informationen finden Sie im "Puppet-Dokumentation"

Objekte und Zähler

Die folgenden Objekte und ihre Zähler werden gesammelt:

Objekt:	Kennungen:	Merkmale:	Datenpunkte:

Puppenagent	Namespace-Knoten-UUID	Knotenname Standort Knoten-IP-Version Konfigurationszeichenfolge-Version Puppet	Änderungen Gesamtereignisse Fehlerereignisse Erfolgsereignisse Gesamtressourcen Geänderte Ressourcen Fehlgeschlagene Ressourcen Neustart fehlgeschlagene Ressourcen Nicht synchronisierte Ressourcen Neu gestartete Ressourcen Geplante Ressourcen Übersprungene Ressourcen Gesamtzeit Ankerzeit Konfigurationsabrufzeit Cron-Zeit Ausführungszeit Dateizeit Filebucket-Zeit Lastrun-Zeit Paketzeit Zeitplanzeit Servicezeit Sshauthorizedkey-Zeit Gesamtzeit Benutzer

Fehlerbehebung

Weitere Informationen finden Sie in der"Support" Seite.

Redis-Datensammler

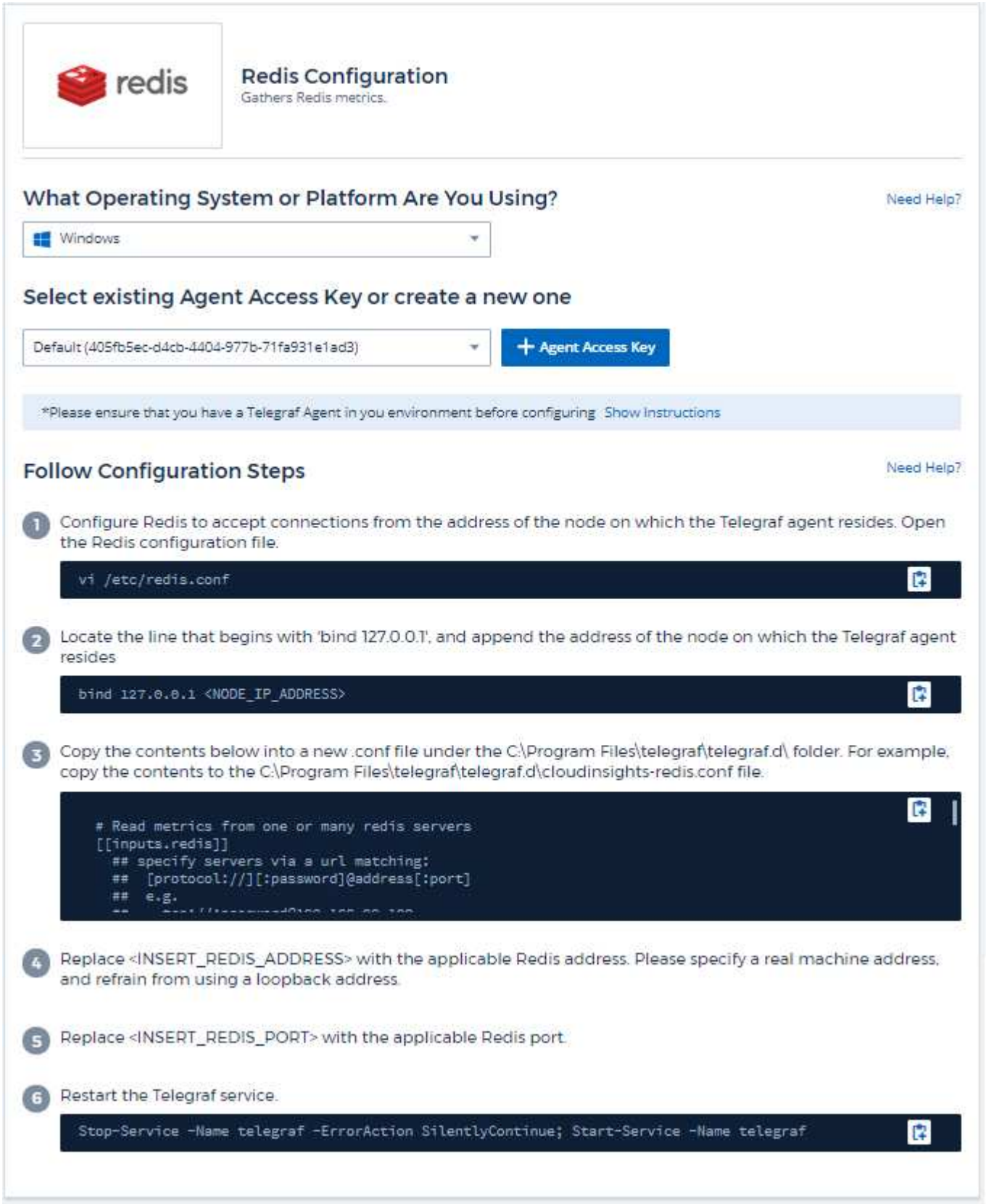
Data Infrastructure Insights verwendet diesen Datensammler, um Metriken von Redis zu sammeln. Redis ist ein Open-Source-Datenstrukturspeicher im Arbeitsspeicher, der als Datenbank, Cache und Nachrichtenbroker verwendet wird und die folgenden Datenstrukturen unterstützt: Zeichenfolgen, Hashes, Listen, Sets und mehr.

Installation

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

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

2. Wenn Sie noch keinen Agenten für die Sammlung installiert haben oder einen Agenten für ein anderes Betriebssystem oder eine andere Plattform installieren möchten, klicken Sie auf *Anweisungen anzeigen*, um das"Agenteninstallation" Anweisungen.
3. Wählen Sie den Agent-Zugriffsschlüssel zur Verwendung mit diesem Datensammler aus. Sie können einen neuen Agentenzugriffsschlüssel hinzufügen, indem Sie auf die Schaltfläche **+ Agentenzugriffsschlüssel** klicken. Best Practice: Verwenden Sie nur dann einen anderen Agent-Zugriffsschlüssel, wenn Sie Datensammler beispielsweise nach Betriebssystem/Plattform gruppieren möchten.
4. Befolgen Sie die Konfigurationsschritte, um den Datensammler zu konfigurieren. Die Anweisungen variieren je nach Art des Betriebssystems oder der Plattform, die Sie zum Sammeln von Daten verwenden.



Aufstellen

Informationen finden Sie im"Redis-Dokumentation" .

Objekte und Zähler

Die folgenden Objekte und ihre Zähler werden gesammelt:

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

Fehlerbehebung

Weitere Informationen finden Sie in der"Support" Seite.