



資料收集器參考 - 服務

Data Infrastructure Insights

NetApp
February 11, 2026

目錄

資料收集器參考 - 服務	1
節點資料收集	1
安裝	1
物體和計數器	1
設定	2
ActiveMQ資料收集器	2
安裝	2
設定	3
物體和計數器	3
故障排除	3
Apache 資料收集器	3
安裝	3
設定	4
物體和計數器	5
故障排除	5
Consul 資料收集器	5
安裝	5
設定	5
領事的物件和計數器	5
故障排除	6
Couchbase資料收集器	6
安裝	6
設定	6
物體和計數器	6
故障排除	6
CouchDB資料收集器	7
安裝	7
設定	7
物體和計數器	7
故障排除	7
Docker 資料收集器	8
安裝	8
設定	8
物體和計數器	9
故障排除	12
Elasticsearch 資料收集器	12
設定	13
物體和計數器	13
故障排除	13

Flink 資料收集器	13
安裝	13
設定	14
物體和計數器	14
故障排除	16
Hadoop資料收集器	16
安裝	16
設定	16
物體和計數器	20
故障排除	20
HAProxy 資料收集器	20
安裝	20
設定	21
物體和計數器	22
故障排除	24
JVM資料收集器	24
安裝	24
設定	24
物體和計數器	24
故障排除	25
Kafka 資料收集器	25
安裝	25
設定	26
物體和計數器	27
故障排除	27
Kibana 資料收集器	27
安裝	27
設定	27
物體和計數器	27
故障排除	27
Kubernetes 監控 Operator 安裝與設定	28
安裝 Kubernetes Monitoring Operator 之前	28
安裝 Kubernetes 監控操作員	28
Kubernetes 監控元件	30
升級到最新的 Kubernetes Monitoring Operator	30
停止並啟動 Kubernetes 監控操作員	32
解除安裝	32
關於 Kube-state-metrics	33
配置/自訂操作員	33
關於秘密的說明	37
驗證 Kubernetes 監控 Operator 鏡像簽名	38

故障排除	38
Memcached 資料收集器	45
安裝	45
設定	46
物體和計數器	46
故障排除	47
MongoDB 資料收集器	47
安裝	47
設定	48
物體和計數器	48
故障排除	49
MySQL資料收集器	49
安裝	49
設定	50
物體和計數器	51
故障排除	51
Netstat資料收集器	51
安裝	52
設定	52
物體和計數器	52
故障排除	53
Nginx 資料收集器	53
安裝	53
設定	54
物體和計數器	54
故障排除	55
PostgreSQL 資料收集器	55
安裝	55
設定	56
物體和計數器	57
故障排除	57
Puppet代理程式資料收集器	57
安裝	57
設定	58
物體和計數器	58
故障排除	59
Redis 資料收集器	59
安裝	59
設定	60
物體和計數器	61
故障排除	61

資料收集器參考 - 服務

節點資料收集

Data Infrastructure Insights從安裝代理程式的節點收集指標。

安裝

1. 從*可觀察性 > 收集器*中選擇一個作業系統/平台。請注意，安裝任何整合資料收集器（Kubernetes、Docker、Apache 等）也將配置節點資料收集。
2. 按照說明配置代理。說明根據您用於收集資料的作業系統或平台的類型而有所不同。

物體和計數器

以下物件及其計數器作為節點指標進行收集：

目的：	標識符：	屬性：	數據點：
節點檔案系統	節點 UUID 設備路徑類型	節點IP 節點名稱 節點OS 模式	空間 Inode 空間 Inode 已使用 Inode 總數 已使用 Inode 總數 已使用 Inode 總數
節點磁碟	節點 UUID 磁碟	節點 IP 節點名稱 節點作業系統	IO 時間總 IOPS 進行中讀取位元組數（每秒）讀取時間總讀取次數（每秒）加權 IO 時間總寫入位元組數（每秒）寫入時間總寫入次數（每秒）當前磁碟佇列長度寫入時間讀取時間 IO 時間
節點CPU	節點 UUID CPU	節點 IP 節點名稱 節點作業系統	系統 CPU 使用率 使用者 CPU 使用率 閒置 CPU 使用率 處理器 CPU 使用率 中斷 CPU 使用率 DPC CPU 使用率

目的：	標識符：	屬性：	數據點：
節點	節點UUID	節點 IP 節點名稱 節點作業系統	核心啟動時間內核上下文切換（每秒）內核可用內核中斷（每秒）內核程序分叉（每秒）活動記憶體可用記憶體總可用記憶體緩衝記憶體快取記憶體提交限制已提交記憶體記憶體高可用記憶體高總記憶體大頁大小記憶體大頁可用記憶體大頁總記憶體低可用記憶體低總記憶體對頁數記憶體共用記憶體尺寸已用記憶體綁定記憶體寫回總記憶體寫回臨時記憶體快取故障記憶體需求零故障記憶體頁故障記憶體頁記憶體非分頁記憶體分頁記憶體快取核心記憶體待機快取正常記憶體待機快取保留記憶體轉換故障行程阻斷行程系統用戶數量 系統調用
節點網路	網路介面節點 UUID	節點名稱 節點 IP 節點 OS	接收位元組數 傳送位元組數 傳出資料包數 丟棄傳出資料包數 錯誤資料包數 接收資料包數 丟棄接收資料包數 錯誤資料包數 傳送資料包數

設定

設定和故障排除資訊可以在["配置代理"](#)頁。

ActiveMQ資料收集器

Data Infrastructure Insights使用此資料收集器從 ActiveMQ 收集指標。

安裝

1. 從*可觀察性 > 收集器*中，按一下*+資料收集器*。選擇 ActiveMQ。

選擇安裝 Telegraf 代理程式的作業系統或平台。

2. 如果您尚未安裝用於收集的代理，或者您希望為不同的作業系統或平台安裝代理，請按一下「顯示說明」以展開["代理安裝"](#)指示。
3. 選擇用於此資料收集器的代理存取金鑰。您可以透過點擊 + 代理存取金鑰 按鈕新增新的代理存取金鑰。最佳實務：只有當您想要對資料收集器進行分組（例如按作業系統/平台）時才使用不同的代理存取金鑰。
4. 依照設定步驟配置資料收集器。說明根據您用於收集資料的作業系統或平台的類型而有所不同。

設定

資訊可以在 "[ActiveMQ 文檔](#)"

物體和計數器

收集以下物件及其計數器：

目的：	標識符：	屬性：	數據點：
ActiveMQ 佇列	命名空間佇列連接埠伺服器	節點名稱節點IP節點UUID	消費者數量 出隊數量 入隊數量 隊列大小
ActiveMQ 訂閱者	客戶端 ID 連接 ID 連接埠 伺服器命名空間	是否處於活動狀態目標節點名稱節點 IP 節點 UUID 節點 OS 選擇器訂閱	出隊計數 已調度計數 已調度隊列大小 入隊計數 待處理隊列大小
ActiveMQ 主題	主題連接埠伺服器命名空間	節點名稱節點 IP 節點 UUID 節點 OS	消費者數量 出隊數量 入隊數量 大小

故障排除

更多資訊可從"[支援](#)"頁。

Apache 資料收集器

此資料收集器允許從租戶上的 Apache 伺服器收集資料。

先決條件

- 您必須設定並正確執行 Apache HTTP 伺服器
- 您必須在代理主機/虛擬機器上擁有 sudo 或管理員權限
- 通常，Apache `mod_status` 模組會配置為在 Apache 伺服器的「`/server-status?auto`」位置顯示一個頁面。必須啟用 `_ExtendedStatus_` 選項才能收集所有可用欄位。有關如何設定伺服器的信息，請參閱 Apache 模組文件：https://httpd.apache.org/docs/2.4/mod/mod_status.html#enable

安裝

1. 從*可觀察性 > 收集器*中，按一下*+資料收集器*。選擇 Apache。
選擇安裝 Telegraf 代理程式的作業系統或平台。
2. 如果您尚未安裝用於收集的代理，或者您希望為不同的作業系統或平台安裝代理，請按一下「顯示說明」以展開"[代理安裝](#)"指示。
3. 選擇用於此資料收集器的代理存取金鑰。您可以透過點擊 + 代理存取金鑰 按鈕新增新的代理存取金鑰。最佳實務：只有當您想要對資料收集器進行分組（例如按作業系統/平台）時才使用不同的代理存取金鑰。
4. 依照設定步驟配置資料收集器。說明根據您用於收集資料的作業系統或平台的類型而有所不同。

設定

Telegraf 的 Apache HTTP 伺服器外掛程式依賴「mod_status」模組的啟用。啟用此功能後，Apache 的 HTTP 伺服器將公開一個 HTML 端點，您可以在瀏覽器上查看該端點，也可以抓取該端點以提取所有 Apache 的 HTTP 伺服器配置的狀態。

相容性:

配置是針對 Apache 的 HTTP 伺服器版本 2.4.38 開發的。

啟用 **mod_status** :

啟用和公開“mod_status”模組涉及兩個步驟：

- 啟用模組
- 從模組中公開統計數據

啟用模組：

模組的載入由“/usr/local/apache/conf/httpd.conf”下的設定檔控制。編輯設定檔並取消註解以下行：

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

從模組公開統計資料：

'mod_status' 的公開由 '/usr/local/apache2/conf/extra/httpd-info.conf' 下的設定檔控制。確保該設定檔中包含以下內容（至少其他指令也會存在）：

```
# 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
```

有關“mod_status”模組的詳細說明，請參閱["Apache 文件"](#)

物體和計數器

收集以下物件及其計數器：

目的：	標識符：	屬性：	數據點：
阿帕契	命名空間伺服器	節點 IP 節點名稱 連接埠 父伺服器設定產生 父伺服器 MPM 產生伺服器正常運作時間正在停止	繁忙工作者 每個請求的位元組數 每秒位元組數 CPU 子進程 系統 CPU 子進程 使用者 CPU 負載 CPU 系統 CPU 用戶 非同步連接 關閉非同步連接 保持活動 異步連接 寫入連接 每個請求的總持續時間 空閒工作者 平均負載（最近 1 分鐘） 平均負載（最近 15 分鐘） 持續時間總數 5 分鐘） 關閉記分板 DNS 尋找記分板 完成記分板 空閒清理記分板 保持活動記分板 記錄記分板 開啟記分板 讀取記分板 發送記分板 啟動記分板 等待

故障排除

更多資訊可從["支援"](#)頁。

Consul 資料收集器

Data Infrastructure Insights使用此資料收集器從 Consul 收集指標。

安裝

1. 從*可觀察性 > 收集器*中，按一下*+資料收集器*。選擇 Consul。

如果您尚未設定用於收集的代理，系統會提示您["安裝代理"](#)您的租戶。

如果您已經配置了代理，請選擇適當的作業系統或平台，然後按一下「繼續」。

2. 請依照 Consul 設定畫面中的指示配置資料收集器。說明根據您用於收集資料的作業系統或平台的類型而有所不同。

設定

資訊可以在["Consul 文件"](#)。

領事的物件和計數器

收集以下物件及其計數器：

目的：	標識符：	屬性：	數據點：
領事	命名空間檢查ID服務節點	節點 IP 節點 OS 節點 UUID 節點名稱 服務名稱 檢查名稱 服務 ID 狀態	關鍵超車警告

故障排除

更多資訊可從["支援"](#)頁。

Couchbase資料收集器

Data Infrastructure Insights使用此資料收集器從 Couchbase 收集指標。

安裝

1. 從*可觀察性 > 收集器*中，按一下*+資料收集器*。選擇 Couchbase。
 - 選擇安裝 Telegraf 代理程式的作業系統或平台。
2. 如果您尚未安裝用於收集的代理，或者您希望為不同的作業系統或平台安裝代理，請按一下「顯示說明」以展開["代理安裝"](#)指示。
3. 選擇用於此資料收集器的代理存取金鑰。您可以透過點擊 + 代理存取金鑰 按鈕新增新的代理存取金鑰。最佳實務：只有當您想要對資料收集器進行分組（例如按作業系統/平台）時才使用不同的代理存取金鑰。
4. 依照設定步驟配置資料收集器。說明根據您用於收集資料的作業系統或平台的類型而有所不同。

[Couchbase 配置]

設定

資訊可以在["Couchbase 文檔"](#)。

物體和計數器

收集以下物件及其計數器：

目的：	標識符：	屬性：	數據點：
Couchbase 節點	命名空間叢集 Couchbase 節點主機名	節點名稱 節點IP	記憶體可用記憶體總計
Couchbase 桶	命名空間 儲存桶 集群	節點名稱 節點IP	已用資料 資料擷取 已使用 磁碟 專案計數 已用記憶體 每秒運算元 已用配額

故障排除

更多資訊可從["支援"](#)頁。

CouchDB資料收集器

Data Infrastructure Insights使用此資料收集器從 CouchDB 收集指標。

安裝

1. 從*可觀察性 > 收集器*中，按一下*+資料收集器*。選擇 CouchDB。
選擇安裝 Telegraf 代理程式的作業系統或平台。
2. 如果您尚未安裝用於收集的代理，或者您希望為不同的作業系統或平台安裝代理，請按一下「顯示說明」以展開"[代理安裝](#)"指示。
3. 選擇用於此資料收集器的代理存取金鑰。您可以透過點擊 + 代理存取金鑰 按鈕新增新的代理存取金鑰。最佳實務：只有當您想要對資料收集器進行分組（例如按作業系統/平台）時才使用不同的代理存取金鑰。
4. 依照設定步驟配置資料收集器。說明根據您用於收集資料的作業系統或平台的類型而有所不同。

[CouchDB 配置]

設定

資訊可以在"[CouchDB 文件](#)"。

物體和計數器

收集以下物件及其計數器：

目的：	標識符：	屬性：	數據點：
CouchDB	命名空間伺服器	節點名稱 節點IP	驗證快取命中 驗證快取未命中 資料庫讀取 資料庫寫入 開啟資料庫 開啟作業系統檔案 最大請求時間 最小請求時間 Httpd 請求方法 複製 Httpd 請求方法 刪除 Httpd 請求方法 取得 Httpd 請求方法 頭部 Httpd 請求方法 發布 Httpd 請求方法 取得 Httpd 請求方法 頭部 Httpd 請求方法 發布 Httpd 請求方法 取得 Httpd 請求方法 頭部 Httpd 請求方法 取得 Httpd 請求方法 頭 20120 狀態狀態代碼 304 狀態代碼 400 狀態代碼 401 狀態代碼 403 狀態代碼 404 狀態代碼 405 狀態代碼 409 狀態代碼 412 狀態代碼 500

故障排除

更多資訊可從"[支援](#)"頁。

Docker 資料收集器

Data Infrastructure Insights使用此資料收集器從 Docker 收集指標。

安裝

1. 從*可觀察性 > 收集器*中，按一下*+資料收集器*。選擇 Docker。

如果您尚未設定用於收集的代理，系統會提示您"安裝代理"您的租戶。

如果您已經配置了代理，請選擇適當的作業系統或平台，然後按一下「繼續」。

2. 請依照 Docker 配置畫面中的說明配置資料收集器。說明根據您用於收集資料的作業系統或平台的類型而有所不同。

[Docker 配置]

設定

Docker 的 Telegraf 輸入插件透過指定的 UNIX 套接字或 TCP 端點收集指標。

相容性

配置是針對 Docker 版本 1.12.6 開發的。

設定

透過 UNIX 套接字存取 Docker

如果 Telegraf 代理程式在裸機上執行，請透過執行以下命令將 telegraf Unix 使用者新增至 docker Unix 群組：

```
sudo usermod -aG docker telegraf
```

如果 Telegraf 代理程式在 Kubernetes pod 中執行，則透過將套接字作為磁碟區對應到 pod 中，然後將該磁碟區掛載到 /var/run/docker.sock 來公開 Docker Unix 套接字。例如，將以下內容新增至 PodSpec：

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

然後，將以下內容新增至容器：

```

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

```

請注意，為 Kubernetes 平台提供的Data Infrastructure Insights安裝程式會自動處理此對應。

透過 TCP 端點存取 Docker

預設情況下，Docker 使用連接埠 2375 進行未加密訪問，使用連接埠 2376 進行加密訪問。

物體和計數器

收集以下物件及其計數器：

目的：	標識符：	屬性：	數據點：
Docker 引擎	命名空間 Docker 引擎	節點名稱 節點 IP 節點 UUID 節點 OS Kubernetes 叢集 Docker 版本 單元	記憶體容器 已暫停的容器 正在運行的容器 已停止的 容器 CPU Go 例程 映像 偵聽器事件 已使用的檔案 描述符 可用資料 已使用的 資料總量 可用元資料 元資 料 已使用的總元資料 池 區塊大小

目的：	標識符：	屬性：	數據點：
Docker容器	命名空間 容器名稱 Docker引擎	Kubernetes 容器雜湊 Kubernetes 容器連接埠 Kubernetes 容器重開機計數 Kubernetes 容器終止訊息路徑 Kubernetes 容器終止訊息策略 Kubernetes Pod 終止寬限期 容器映像 容器狀態 容器版本 節點名稱 Kubernetes 容器終止站 Kubernet 容器名稱 Knetuber 容器名稱 Knetnetfnet net1沙盒 ID 節點 IP 節點 UUID Docker 版本 Kubernetes IO 配置 可見 Kubernetes IO 配置 來源 OpenShift IO SCC Kubernetes 描述 Kubernetes 顯示名稱 OpenShift 標籤 Kompose 服務 Pod 模板 雜湊 控制器 修訂 StatefulSet Pod 名稱 租用戶 Web 控制台 架構 權威來源 URL 建置日期 RH 建置主機 RH 元件分佈 範圍 安裝 發佈 運作 摘要 解除安裝 VCS 引用 VCS 類型 供應商 版本 健康狀況 容器 ID	記憶體活動 匿名記憶體 活動檔案 記憶體快取 記憶體層級限制 記憶體非活動 匿名記憶體非活動 檔案 記憶體限制 記憶體對應 檔案 記憶體最大使用量 記憶體頁面錯誤 記憶體頁面嚴重錯誤 記憶體分頁傳入 記憶體分頁傳出 記憶體駐留 集大小 記憶體駐留 集大小 巨大 記憶體總活動 匿名記憶體總活動 檔案 記憶體總快取 記憶體總非活動 匿名記憶體總非活動 檔案 記憶體總映射 檔案 記憶體總頁面嚴重錯誤 記憶體總分頁傳入 記憶體總分頁傳出 記憶體總駐留 集大小 記憶體總駐留 集大小 巨大 記憶體總不可清除 記憶體不可清除 記憶體使用量 記憶體使用率 退出程式碼 OOM 終止 PID 啟動於 失敗 連勝

目的：	標識符：	屬性：	數據點：
Docker 容器區塊 IO	命名空間容器名稱裝置Docker引擎	Kubernetes 容器雜湊 Kubernetes 容器連接埠 Kubernetes 容器重開機計數 Kubernetes 容器終止訊息路徑 Kubernetes 容器終止訊息策略 Kubernetes Pod 終止寬限期 容器映像 容器狀態 容器版本 節點名稱 Kubernetes 容器終止站 Kubernet 容器名稱 Knetuber 容器名稱 Knetnetfnet net1沙盒 ID 節點 IP 節點 UUID Docker 版本 Kubernetes 配置可見 Kubernetes 配置來源 OpenShift SCC Kubernetes 說明 Kubernetes 顯示名稱 OpenShift 標籤 架構 架構 版本 Pod 範本雜湊 控制器 修訂雜湊 Pod 範本產生 Kompose 服務服務架構權威來源 URL RH 建置主機 RH 元件分佈範圍 安裝 維護者 發佈 運行摘要 解除安裝 VCS 引用 VCS 類型 版本 架構 URL 架構 VCS URL 架構 版本 容器 ID	IO 服務位元組數 遞歸異步 IO 服務位元組數 遞歸讀取 IO 服務位元組數 遞歸同步 IO 服務位元組數 遞歸總計 IO 服務位元組數 遞歸寫入 IO 服務 遞歸異步 IO 服務 遞歸 IO 服務位元組數 遞歸寫入 IO 服務 遞歸
Docker容器網路	命名空間容器名稱網路Docker引擎	容器映像 容器狀態 容器版本 節點名稱 節點IP 節點UUID 節點OS K8s叢集 Docker版本 容器ID	RX 丟棄 RX 位元組數 RX 錯誤數 RX 封包數 TX 丟棄 TX 位元組數 TX 錯誤數 TX 封包數

目的：	標識符：	屬性：	數據點：
Docker 容器 CPU	命名空間容器名稱CPU Docker引擎	Kubernetes 容器雜湊 Kubernetes 容器連接埠 Kubernetes 容器重新啟動 計數 Kubernetes 容器終止 訊息路徑 Kubernetes 容器 終止訊息策略 Kubernetes Pod 終止寬限期 Kubernetes 設定可見 Kubernetes 設定來源 OpenShift SCC 容器映像 容器狀態 容器版本 節點配 置可見 Kubernetes 設定來 源 OpenShift SCC 容器映 像 netuber 容器版本 節點 代碼Kubernetes Pod 命名 空間 Kubernetes Pod UID Kubernetes 沙盒 ID 節點 IP 節點 UUID 節點作業系 統 Kubernetes 叢集 Docker 版本 Kubernetes 說明 Kubernetes 顯示名稱 OpenShift 標籤 架構版本 Pod 範本雜湊 控制器修訂 版 Pod設定 客戶端架構 Kompose 服務授權服務授 權控制台許可證供應商架 構權威來源 URL RH 建置 主機 RH 元件分佈範圍安 裝維護者發佈運行摘要卸 載 VCS 引用 VCS 類型版 本架構 URL 架構 VCS URL 架構版本容器 ID	限制週期限限制週期限 限制時間核心模式使用 率使用者模式使用率使用 率百分比系統使用率總計

故障排除

問題：	試試一下：
在按照配置頁面上的說明操作後，我沒有在Data Infrastructure Insights中看到我的 Docker 指標。	檢查 Telegraf 代理程式日誌，看它是否報告以下錯誤： E! 插件 [inputs.docker] 中的錯誤：嘗試連接到 Docker 守護程式套接字時權限被拒絕如果確實如此，請採取必要的步驟，以便按照上面指定的方式為 Telegraf 代理提供對 Docker Unix 套接字的訪問權限。

更多資訊可從["支援"](#)頁。

Elasticsearch 資料收集器

Data Infrastructure Insights使用此資料收集器從 Elasticsearch 收集指標。

1. 從*可觀察性 > 收集器*中，按一下*+資料收集器*。選擇 Elasticsearch。

選擇安裝 Telegraf 代理程式的作業系統或平台。

2. 如果您尚未安裝用於收集的代理，或者您希望為不同的作業系統或平台安裝代理，請按一下「顯示說明」以展開"代理安裝"指示。
3. 選擇用於此資料收集器的代理存取金鑰。您可以透過點擊 + 代理存取金鑰 按鈕新增新的代理存取金鑰。最佳實務：只有當您想要對資料收集器進行分組（例如按作業系統/平台）時才使用不同的代理存取金鑰。
4. 依照設定步驟配置資料收集器。說明根據您用於收集資料的作業系統或平台的類型而有所不同。

[Elasticsearch 配置]

設定

資訊可以在"Elasticsearch 文件"。

物體和計數器

收集以下物件及其計數器：

目的：	標識符：	屬性：
Elasticsearch 集群	命名空間集群	節點IP 節點名稱 叢集狀態
Elasticsearch 節點	命名空間 叢集 ES 節點 ID ES 節點 IP ES 節點	區域 ID

故障排除

更多資訊可從"支援"頁。

Flink 資料收集器

Data Infrastructure Insights使用此資料收集器從 Flink 收集指標。

安裝

1. 從*可觀察性 > 收集器*中，按一下*+資料收集器*。選擇 Flink。
選擇安裝 Telegraf 代理程式的作業系統或平台。
2. 如果您尚未安裝用於收集的代理，或者您希望為不同的作業系統或平台安裝代理，請按一下「顯示說明」以展開"代理安裝"指示。
3. 選擇用於此資料收集器的代理存取金鑰。您可以透過點擊 + 代理存取金鑰 按鈕新增新的代理存取金鑰。最佳實務：只有當您想要對資料收集器進行分組（例如按作業系統/平台）時才使用不同的代理存取金鑰。
4. 依照設定步驟配置資料收集器。說明根據您用於收集資料的作業系統或平台的類型而有所不同。

[Flink 配置]

設定

完整的 Flink 部署涉及以下元件：

JobManager：Flink 主系統。協調一系列 TaskManager。在高可用性設定中，系統將有多個 JobManager。
TaskManager：這是執行 Flink 運算子的地方。Flink 插件基於 telegraf 的 Jolokia 插件。由於需要從所有 Flink 組件收集信息，因此需要在所有組件上配置並透過 Jolokia 公開 JMX。

相容性

配置是針對 Flink 版本 1.7.0 開發的。

設定

鬼椒特效罐

對於所有單一元件，必須下載 Jolokia 代理程式 jar 檔案的版本。測試的版本是"[Jolokia代理 1.6.0](#)"。

以下說明假設下載的 jar 檔案 (jolokia-jvm-1.6.0-agent.jar) 位於「/opt/flink/lib/」位置下。

作業管理器

若要設定 JobManager 以公開 Jolokia API，您可以在節點上設定以下環境變量，然後重新啟動 JobManager：

```
export FLINK_ENV_JAVA_OPTS="-javaagent:/opt/flink/lib/jolokia-jvm-1.6.0-agent.jar=port=8778,host=0.0.0.0"
```

您可以為 Jolokia (8778) 選擇不同的連接埠。如果您有一個內部 IP 來鎖定 Jolokia，您可以用自己的 IP 替換「catch all」 0.0.0.0。請注意，此 IP 需要可從 telegraf 插件存取。

工作管理員

若要設定 TaskManager 以公開 Jolokia API，您可以在節點上設定下列環境變量，然後重新啟動 TaskManager：

```
export FLINK_ENV_JAVA_OPTS="-javaagent:/opt/flink/lib/jolokia-jvm-1.6.0-agent.jar=port=8778,host=0.0.0.0"
```

您可以為 Jolokia (8778) 選擇不同的連接埠。如果您有一個內部 IP 來鎖定 Jolokia，您可以用自己的 IP 替換「catch all」 0.0.0.0。請注意，此 IP 需要可從 telegraf 插件存取。

物體和計數器

收集以下物件及其計數器：

目的：	標識符：	屬性：	數據點：
Flink 任務管理器	叢集命名空間伺服器	節點名稱 任務管理器ID 節點IP	網路可用記憶體段 網路總記憶體段 垃圾收集 PS 標記清除計數 垃圾收集 PS 標記清除時間 垃圾收集 PS 清除計數 垃圾收集 PS 清除時間 堆內存 已提交堆內存 初始化堆內存 最大堆內存 已用線程數 守護程序線程數 峰值線程數 已啟動線程數
Flink 作業	叢集命名空間伺服器作業 ID	節點名稱 作業名稱 節點 IP 上次檢查點 外部路徑 重新啟動時間	停機時間 完全重啟 上次檢查點對齊 緩衝的上次檢查點持續時間 上次檢查點大小 已完成檢查點的數量 失敗的檢查點的數量 正在進行的檢查點的數量 檢查點數量 正常運行時間
Flink 作業管理器	叢集命名空間伺服器	節點名稱 節點IP	垃圾收集 PS 標記清除計數 垃圾收集 PS 標記清除時間 垃圾收集 PS 清除計數 垃圾收集 PS 清除時間 堆內存 已提交堆內存 初始化堆內存 最大堆內存 已使用已註冊任務管理器數量 正在運行的作業數量 可用的任務槽數量 任務槽總數 守護線程數 峰值線程數 已啟動線程數
Flink 任務	叢集命名空間作業 ID 任務 ID	伺服器節點名稱 作業名稱 子任務索引 任務嘗試ID 任務嘗試次數 任務名稱 任務管理器ID 節點IP 目前輸入 浮水印	緩衝區入池使用情況 緩衝區入佇列長度 緩衝區出池使用情況 緩衝區出佇列長度 本地緩衝區數量 本地每秒緩衝區入計數 本地每秒緩衝區入計數速率 遠端緩衝區數量 遠端每秒緩衝區入計數 遠端每秒緩衝區入計數速率 緩衝區出數量 緩衝區出計數 每秒緩衝區出計數速率 本地元組數 本地每秒位元組數 本地每秒位元組數 本地每秒位元組數速率 遠端位元組數 遠端每秒位元組數 計數 遠端每秒位元組數速率 出位元組數 每秒位元組數計數 每秒出位元組數速率 記錄入數 每秒記錄入數計數 每秒記錄入數速率 記錄出數 每秒記錄出數

目的：	標識符：	屬性：	數據點：
Flink 任務運算符	叢集命名空間作業 ID 操作員 ID 任務 ID	伺服器節點名稱作業名稱 操作員名稱子任務索引 任務嘗試ID任務嘗試次數 任務名稱任務管理器ID節 點IP	目前輸入浮水印 目前輸出 浮水印 輸入記錄數 每秒輸 入記錄數 計數 每秒輸入記 錄數速率 輸出記錄數 每秒 輸出記錄數 計數 每秒輸出 記錄數速率 延遲記錄數 已 分配分區 消耗位元組數 速 率 提交延遲 平均提交延遲 最大提交速率 提交失敗 提 交成功 獲取速率 獲取延遲 取得速率 取得速率限制連 接時間值最大心跳速率 傳 入位元組速率 IO 比率 平 均 IO 時間 (奈秒) IO 等 待比率 平均 IO 等待時間 (奈秒) 加入速率 加入時 間 上次心跳前的平均時間 網路 IO 速率 傳出位元組 速率 消耗記錄數 速率 記 錄延遲同步後的最大速度 要求最大值

故障排除

更多資訊可從["支援"](#)頁。

Hadoop資料收集器

Data Infrastructure Insights使用此資料收集器從 Hadoop 收集指標。

安裝

1. 從*可觀察性 > 收集器*中，按一下*+資料收集器*。選擇 Hadoop。
 - 選擇安裝 Telegraf 代理程式的作業系統或平台。
2. 如果您尚未安裝用於收集的代理，或者您希望為不同的作業系統或平台安裝代理，請按一下「顯示說明」以展開["代理安裝"](#)指示。
3. 選擇用於此資料收集器的代理存取金鑰。您可以透過點擊 + 代理存取金鑰 按鈕新增新的代理存取金鑰。最佳實務：只有當您想要對資料收集器進行分組（例如按作業系統/平台）時才使用不同的代理存取金鑰。
4. 依照設定步驟配置資料收集器。說明根據您用於收集資料的作業系統或平台的類型而有所不同。

[Hadoop配置] [Hadoop配置]

設定

完整的 Hadoop 部署涉及以下元件：

- NameNode：Hadoop 分散式檔案系統 (HDFS) 主系統。協調一系列 DataNode。
- 輔助 NameNode：主 NameNode 的溫故障轉移。在 Hadoop 中，向 NameNode 的提升不會自動發生。輔助 NameNode 收集來自 NameNode 的訊息，以便在需要時進行提升。
- DataNode：資料的實際所有者。
- ResourceManager：計算主系統 (Yarn)。協調一系列 NodeManager。
- NodeManager：用於計算的資源。應用程式運行的實際位置。
- JobHistoryServer：負責服務所有與作業歷史相關的請求。

Hadoop 插件基於 telegraf 的 Jolokia 插件。由於需要從所有 Hadoop 元件收集信息，因此需要在所有元件上配置並透過 Jolokia 公開 JMX。

相容性

配置是針對 Hadoop 版本 2.9.2 開發的。

設定

鬼椒特效罐

對於所有單一元件，必須下載 Jolokia 代理程式 jar 檔案的版本。測試的版本是 "[Jolokia 代理 1.6.0](#)"。

以下說明假設下載的 jar 檔案 (jolokia-jvm-1.6.0-agent.jar) 位於 「/opt/hadoop/lib/」 位置下。

名稱節點

若要設定 NameNode 以公開 Jolokia API，您可以在 <HADOOP_HOME>/etc/hadoop/hadoop-env.sh 中設定以下內容：

```
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.
```

輔助 NameNode

若要設定輔助 NameNode 以公開 Jolokia API，您可以在 <HADOOP_HOME>/etc/hadoop/hadoop-env.sh 中設定以下內容：

```
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.

資料節點

若要設定 DataNode 以公開 Jolokia API，您可以在 <HADOOP_HOME>/etc/hadoop/hadoop-env.sh 中設定以下內容：

```
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.

資源管理器

若要設定 ResourceManager 以公開 Jolokia API，您可以在 <HADOOP_HOME>/etc/hadoop/hadoop-env.sh 中設定以下內容：

```
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.

節點管理器

若要設定 NodeManagers 以公開 Jolokia API，您可以在 <HADOOP_HOME>/etc/hadoop/hadoop-env.sh 中設定以下內容：

```
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.

作業歷史伺服器

若要設定 JobHistoryServer 以公開 Jolokia API，您可以在 <HADOOP_HOME>/etc/hadoop/hadoop-env.sh 中設定以下內容：

```
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.

物體和計數器

收集以下物件及其計數器：

目的：	標識符：	屬性：
Hadoop 輔助 NameNode	叢集命名空間伺服器	節點名稱 節點IP 編譯資訊 版本
Hadoop 節點管理器	叢集命名空間伺服器	節點名稱 節點IP
Hadoop資源管理器	叢集命名空間伺服器	節點名稱 節點IP
Hadoop 資料節點	叢集命名空間伺服器	節點名稱 節點 IP 叢集 ID 版本
Hadoop 名稱節點	叢集命名空間伺服器	節點名稱節點 IP 事務 ID 上次寫入時間自上次載入編輯 HA 狀態檔案系統狀態區塊池 ID 叢集 ID 編譯資訊不同版本計數版本
Hadoop 作業歷史伺服器	叢集命名空間伺服器	節點名稱 節點IP

故障排除

更多資訊可從["支援"](#)頁。

HAProxy 資料收集器

Data Infrastructure Insights使用此資料收集器從 HAProxy 收集指標。

安裝

1. 從*可觀察性 > 收集器*中，按一下*+資料收集器*。選擇 HAProxy。

選擇安裝 Telegraf 代理程式的作業系統或平台。

2. 如果您尚未安裝用於收集的代理，或者您希望為不同的作業系統或平台安裝代理，請按一下「顯示說明」以展開["代理安裝"](#)指示。

3. 選擇用於此資料收集器的代理存取金鑰。您可以透過點擊 + 代理存取金鑰 按鈕新增新的代理存取金鑰。最佳實務：只有當您想要對資料收集器進行分組（例如按作業系統/平台）時才使用不同的代理存取金鑰。
4. 依照設定步驟配置資料收集器。說明根據您用於收集資料的作業系統或平台的類型而有所不同。

[HAProxy 配置]

設定

Telegraf 的 HAProxy 外掛程式依賴 HAProxy Stats 的支援。這是 HAProxy 內建的配置，但它並非開箱即用。啟用後，HAProxy 將公開一個 HTML 端點，您可以在瀏覽器上查看該端點，也可以抓取該端點以提取所有 HAProxy 配置的狀態。

相容性:

配置是針對 HAProxy 版本 1.9.4 開發的。

設定:

要啟用統計訊息，請編輯您的 haproxy 設定檔並在「預設」部分後添加以下行，使用您自己的使用者/密碼和/或 haproxy URL：

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

以下是啟用統計資訊的簡化範例設定檔：

```
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
```

有關完整和最新的說明，請參閱["HAProxy 文檔"](#)。

物體和計數器

收集以下物件及其計數器：

目的：	標識符：	屬性：	數據點：
HAProxy 前端	命名空間地址代理	節點 IP 節點名稱 代理 ID 模式 進程 ID 會話速率限制 伺服器 ID 會話限制狀態	傳入位元組數 傳出位元組數 快取命中數 快取查找數 壓縮 繞過的位元組數 傳入 壓縮 傳出位元組數 快取命中數 快取查找數 壓縮 繞過的位元組數 傳入 壓縮 傳出位元組數 壓縮回應數 連線速率 連線速率 最大連線數 連線規則拒絕的請求總數 安全性考慮拒絕的請求數為 5 其他請求 攔截的會話速率 會話速率 最大請求速率 請求速率 最大請求速率 總會話數 會話數 最大會話數 總請求數 重寫
HAProxy 伺服器	命名空間位址代理伺服器	節點IP 節點名稱 檢查完成時間 檢查下降配置 檢查健康值 檢查上升配置 檢查狀態 代理ID 上次更改時間 上次會話時間 模式 進程ID 伺服器ID 狀態 權重	活動伺服器 備份伺服器 輸入位元組數 輸出位元組數 檢查停機數 檢查失敗數 用戶端中止連線數 連線平均時間 停機時間 拒絕回應總數 連線錯誤數 回應錯誤數 回應 1xx 數 回應 2xx 數 回應 3xx 數 工作階段 4xx 數位回應 5 數 佇列時間佇列 其他數位伺服器最大連線重用數 回應時間 平均會話數 最大會話數 伺服器傳輸中止數 會話總數 會話總數 總時間 平均請求數 重新分派數 請求數 重試數 請求數 重寫數
HAProxy 後端	命名空間地址代理	節點IP 節點名稱 代理ID 上次更改時間 上次會話時間 模式 進程ID 伺服器ID 會話限制 狀態 權重	活動伺服器 備份伺服器 傳入位元組數 傳出位元組數 快取命中數 快取查找數 檢查失敗數 用戶端中止數 壓縮 繞過的位元組數 壓縮傳入位元組數 壓縮傳出位元組數 壓縮回應數 連線數 連線平均時間 停機時間 因安全問題拒絕回應數 回應數 5xx 回應數 選定的其他伺服器 總計 佇列數 目前 佇列數 最大佇列數 平均時間 每秒會話數 每秒會話數 最大請求數 總連線重用數 回應時間 平均會話數 最大會話數 伺服器傳送中止數 會話數 會話總數 時間 平均請求數 重新分派數 最大會話數 伺服器傳送中止數 會話數 會話總數

故障排除

更多資訊可從["支援"](#)頁。

JVM資料收集器

Data Infrastructure Insights使用此資料收集器從 JVM 收集指標。

安裝

1. 從*可觀察性 > 收集器*中，按一下*+資料收集器*。選擇 JVM。
選擇安裝 Telegraf 代理程式的作業系統或平台。
2. 如果您尚未安裝用於收集的代理，或者您希望為不同的作業系統或平台安裝代理，請按一下「顯示說明」以展開["代理安裝"](#)指示。
3. 選擇用於此資料收集器的代理存取金鑰。您可以透過點擊 + 代理存取金鑰 按鈕新增新的代理存取金鑰。最佳實務：只有當您想要對資料收集器進行分組（例如按作業系統/平台）時才使用不同的代理存取金鑰。
4. 依照設定步驟配置資料收集器。說明根據您用於收集資料的作業系統或平台的類型而有所不同。

[JVM 配置]

設定

資訊可以在["JVM 文檔"](#)。

物體和計數器

收集以下物件及其計數器：

目的：	標識符：	屬性：	數據點：
JVM	命名空間 JVM	作業系統架構 作業系統名稱 作業系統版本 執行階段規格 執行階段規格供應商 執行階段規格版本 正常運作時間 執行階段虛擬機器名稱 執行階段虛擬機器供應商 執行階段虛擬機器版本 節點名稱 節點 IP	載入的類別 已載入的類別卸載的類別總數 記憶體 堆提交的記憶體 堆初始化記憶體 堆已用最大記憶體 堆已用記憶體 非堆提交記憶體 非堆初始化記憶體 非堆最大記憶體 非堆疊已用記憶體 磁碟機台已用記憶體 作業系統可用處理器 作業系統已提交記憶體大小 作業系統作業系統作業系統負載程式 作業系統交換時數 作業系統系統平均負載 作業系統總實體記憶體大小 作業系統總交換空間大小 執行緒守護程式數 執行緒峰值數 執行緒數 執行緒總數 已啟動執行緒數 垃圾收集器複製收集數 垃圾收集器複製收集時間 垃圾回收器標記-清除收集數 垃圾收集器標記-清除收集時間 垃圾收集器 G1 老數代垃圾回收數 收集數個垃圾代數數 垃圾收集數垃圾收集器並發標記-清除收集計數 垃圾收集器並發標記-清除收集時間 垃圾收集器並行收集計數 垃圾收集器並行收集時間 垃圾收集器並行清除標記-清除收集計數 垃圾收集器並行清除標記-清除收集時間 垃圾收集器並行清除收集計數 垃圾收集器並行清除收集時間

故障排除

更多資訊可從["支援"](#)頁。

Kafka 資料收集器

Data Infrastructure Insights使用此數據收集器從 Kafka 收集指標。

安裝

1. 從*可觀察性 > 收集器*中，按一下*+資料收集器*。選擇 Kafka。

選擇安裝 Telegraf 代理程式的作業系統或平台。

2. 如果您尚未安裝用於收集的代理，或者您希望為不同的作業系統或平台安裝代理，請按一下「顯示說明」以展開“代理安裝”指示。
3. 選擇用於此資料收集器的代理存取金鑰。您可以透過點擊 + 代理存取金鑰 按鈕新增新的代理存取金鑰。最佳實務：只有當您想要對資料收集器進行分組（例如按作業系統/平台）時才使用不同的代理存取金鑰。
4. 依照設定步驟配置資料收集器。說明根據您用於收集資料的作業系統或平台的類型而有所不同。

[Kafka 配置]

設定

Kafka 插件基於 telegraf 的 Jolokia 插件。由於需要從所有 Kafka 代理程式收集信息，因此需要在所有元件上配置並透過 Jolokia 公開 JMX。

相容性

配置是針對 Kafka 版本 0.11.0.2 開發的。

設定

以下所有說明均假設 kafka 的安裝位置是「/opt/kafka」。您可以調整以下說明以反映您的安裝位置。

鬼椒特效罐

Jolokia 代理 jar 檔案必須是“下載”。測試的版本是 Jolokia 代理 1.6.0。

以下說明假設下載的 jar 檔案 (jolokia-jvm-1.6.0-agent.jar) 位於「/opt/kafka/libs/」位置下。

Kafka 經紀人

要設定 Kafka Brokers 以公開 Jolokia API，您可以在 <KAFKA_HOME>/bin/kafka-server-start.sh 中的「kafka-run-class.sh」呼叫之前新增以下內容：

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

請注意，上面的範例使用“hostname -I”來設定“RMI_HOSTNAME”環境變數。在多個 IP 機器中，需要進行調整以收集您關心的 RMI 連線 IP。

您可以為 JMX (9999 以上) 和 Jolokia (8778) 選擇不同的連接埠。如果您有一個內部 IP 來鎖定 Jolokia，您可以用自己的 IP 替換「catch all」 0.0.0.0。請注意，此 IP 需要可從 telegraf 插件存取。如果您不想進行身份驗證，可以使用選項「-Dcom.sun.management.jmxremote.authenticate=false」。使用風險自負。

物體和計數器

收集以下物件及其計數器：

目的：	標識符：	屬性：
卡夫卡經紀人	叢集命名空間代理	節點名稱 節點IP

故障排除

更多資訊可從["支援"](#)頁。

Kibana 資料收集器

Data Infrastructure Insights使用此資料收集器從 Kibana 收集指標。

安裝

1. 從*可觀察性 > 收集器*中，按一下*+資料收集器*。選擇 Kibana。
選擇安裝 Telegraf 代理程式的作業系統或平台。
2. 如果您尚未安裝用於收集的代理，或者您希望為不同的作業系統或平台安裝代理，請按一下「顯示說明」以展開["代理安裝"](#)指示。
3. 選擇用於此資料收集器的代理存取金鑰。您可以透過點擊 + 代理存取金鑰 按鈕新增新的代理存取金鑰。最佳實務：只有當您想要對資料收集器進行分組（例如按作業系統/平台）時才使用不同的代理存取金鑰。
4. 依照設定步驟配置資料收集器。說明根據您用於收集資料的作業系統或平台的類型而有所不同。

[Kibana 配置]

設定

資訊可以在["Kibana 文檔"](#)。

物體和計數器

收集以下物件及其計數器：

目的：	標識符：	屬性：	數據點：
基巴納	命名空間位址	節點IP 節點名稱 版本狀態	並發連線數 堆最大堆使用量 每秒請求數 回應時間 平均回應時間 最大正常運作時間

故障排除

更多資訊可從["支援"](#)頁。

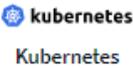
Kubernetes 監控 Operator 安裝與設定

Data Infrastructure Insights 為 Kubernetes 集合提供了 **Kubernetes Monitoring Operator**。導覽至 **Kubernetes > Collectors > +Kubernetes Collector** 來部署新的操作員。

安裝 Kubernetes Monitoring Operator 之前

查看“先決條件”在安裝或升級 Kubernetes Monitoring Operator 之前，請先閱讀文件。

安裝 Kubernetes 監控操作員



Deploy NetApp Monitoring Operator

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

Select existing API Access Token or create a new one

KEY2024 (...vw6NdM) + API Access Token [Production Best Practices ?](#)

Installation Instructions [Need Help?](#)

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

- 1 Define Kubernetes cluster name and namespace**

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

Cluster	Namespace
<input type="text" value="clustername"/>	<input type="text" value="netapp-monitoring"/>
- 2 Download the operator YAML files**

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

[Copy Download Command Snippet](#)

[Reveal Download Command Snippet](#)

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

3 Optional: Upload the operator images to your private repository

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

Copy Image Pull Snippet

Reveal Image Pull Snippet

Copy Repository Password

Reveal Repository Password

This password is valid for 24 hours.

4 Optional: Review available configuration options

Configure custom options such as proxy and private repository settings. Review the [instructions and available options](#).

5 Deploy the operator (create new or upgrade existing)

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

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

Copy kubectl Apply Snippet

Reveal kubectl Apply Snippet

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

6 Next

在 **Kubernetes** 上安裝 **Kubernetes Monitoring Operator** 代理程式的步驟：

1. 輸入唯一的叢集名稱和命名空間。如果你升級與先前的 Kubernetes Operator 一樣，使用相同的叢集名稱和命名空間。
2. 輸入這些內容後，您可以將下載命令片段複製到剪貼簿。
3. 將程式碼片段貼到 `bash` 視窗並執行。將下載 Operator 安裝檔。請注意，該程式碼片段具有唯一密鑰，並且有效期為 24 小時。
4. 如果您有自訂或私人儲存庫，請複製可選的 Image Pull 程式碼片段，將其貼上到 `bash shell` 中並執行它。提取圖像後，將其複製到您的私人儲存庫。確保保持相同的標籤和資料夾結構。更新 `_operator-deployment.yaml_` 中的路徑以及 `_operator-config.yaml_` 中的 docker 儲存庫設定。
5. 如果需要，請查看可用的配置選項，例如代理或私人儲存庫設定。您可以閱讀更多關於"[配置選項](#)"。
6. 準備好後，透過複製 `kubectl Apply` 程式碼片段、下載並執行它來部署 Operator。
7. 安裝將自動進行。完成後，按一下“下一步”按鈕。
8. 安裝完成後，按一下“下一步”按鈕。請確保刪除或安全儲存 `operator-secrets.yaml` 檔案。

如果您有自訂儲存庫，請閱讀[使用自訂/私人 docker 倉庫](#)。

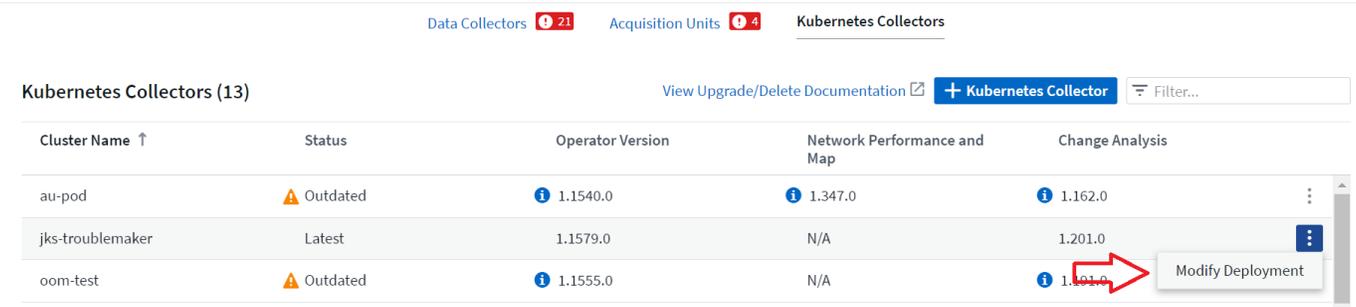
Kubernetes 監控元件

Data Infrastructure Insights Kubernetes 監控由四個監控元件組成：

- 集群指標
- 網路效能和地圖（可選）
- 事件日誌（可選）
- 變化分析（可選）

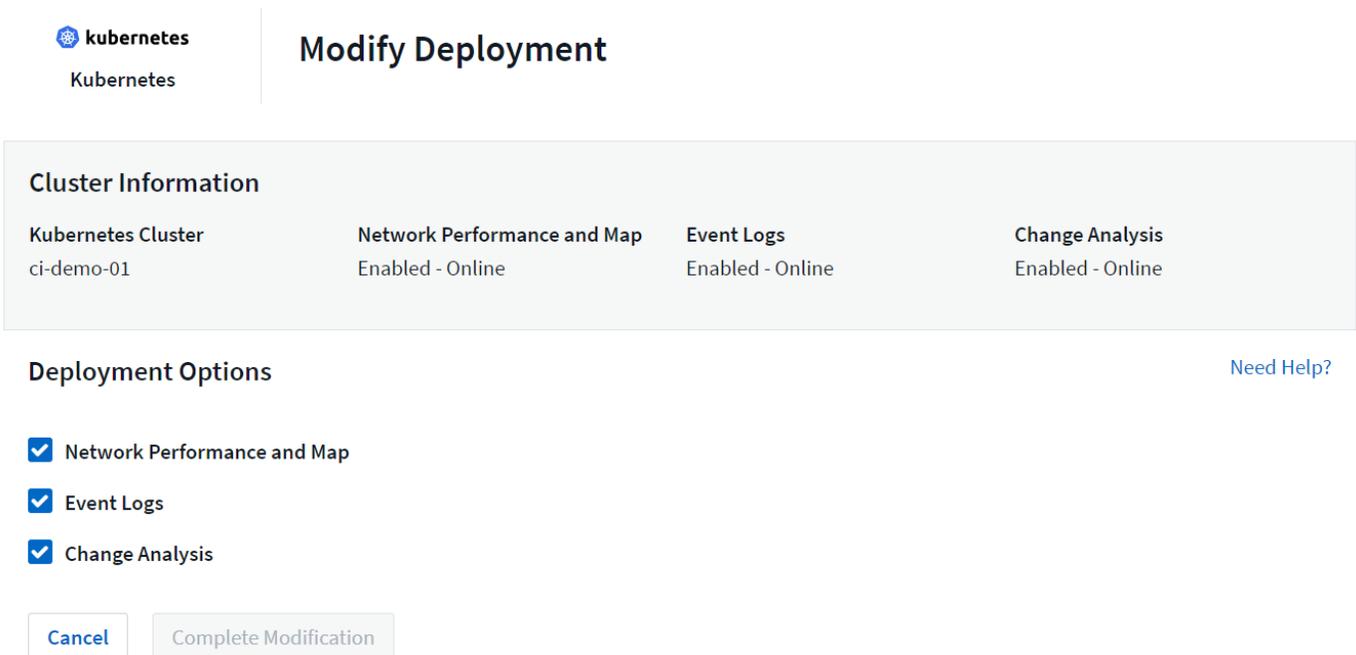
預設情況下，每個 Kubernetes 收集器都會啟用上述選用元件；如果您決定不需要特定收集器的元件，則可以透過導覽至 **Kubernetes > Collectors** 並從螢幕右側收集器的「三個點」選單中選擇 **修改部署** 來停用它。

NetApp / Observability / Collectors



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

螢幕顯示每個元件的目前狀態，並允許您根據需要停用或啟用該收集器的元件。



kubernetes
Kubernetes

Modify Deployment

Cluster Information

Kubernetes Cluster	Network Performance and Map	Event Logs	Change Analysis
ci-demo-01	Enabled - Online	Enabled - Online	Enabled - Online

Deployment Options [Need Help?](#)

- Network Performance and Map
- Event Logs
- Change Analysis

[Cancel](#) [Complete Modification](#)

升級到最新的 Kubernetes Monitoring Operator

DII 按鈕升級

您可以透過 DII Kubernetes Collectors 頁面升級 Kubernetes Monitoring Operator。按一下要升級的叢集旁的選

單，然後選擇“升級”。操作員將驗證影像簽名，對目前安裝進行快照並執行升級。幾分鐘內，您將看到操作員狀態從「升級進行中」進展到「最新」。如果遇到錯誤，您可以選擇錯誤狀態以了解更多詳細信息，並參考下面的按鈕升級故障排除表。

使用私有儲存庫進行按鈕升級

如果您的操作員配置為使用私人儲存庫，請確保運行操作員所需的所有映像及其簽名均可在您的儲存庫中取得。如果在升級過程中遇到缺少圖像的錯誤，只需將它們新增至儲存庫並重試升級。若要將圖像簽名上傳到您的儲存庫，請使用以下 `cosign` 工具，確保上傳 3 下指定的所有圖像的簽名可選：將操作員圖像上傳到您的私人儲存庫 > 圖像拉取片段

```
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>
```

回滾到之前運行的版本

如果您使用按鈕升級功能進行升級，並且在升級後七天內遇到當前版本的操作員的任何困難，則可以使用升級過程中建立的快照降級到先前執行的版本。按一下要回滾的叢集旁邊的選單，然後選擇「回滾」。

手動升級

確定現有 Operator 是否存在 *AgentConfiguration*（如果您的命名空間不是預設的 *netapp-monitoring*，請替換為對應的命名空間）：

```
kubectl -n netapp-monitoring get agentconfiguration netapp-ci-monitoring-configuration
如果存在 _AgentConfiguration_：
```

- [安裝](#)最新的 Operator 優於現有的 Operator。
 - 確保您[拉取最新的容器鏡像](#)如果您使用自訂儲存庫。

如果 *AgentConfiguration* 不存在：

- 記下 Data Infrastructure Insights 識別的叢集名稱（如果您的命名空間不是預設的 *netapp-monitoring*，請替換為適當的命名空間）：

```
kubectl -n netapp-monitoring get agent -o
jsonpath='{.items[0].spec.cluster-name}'
* 建立現有 Operator 的備份（如果您的命名空間不是預設的 netapp-
monitoring，請替換為適當的命名空間）：
```

```
kubectl -n netapp-monitoring get agent -o yaml > agent_backup.yaml
* <<to-remove-the-kubernetes-monitoring-operator,解除安裝>>現有的營運商。
* <<installing-the-kubernetes-monitoring-operator,安裝>>最新的操作員。
```

- 使用相同的叢集名稱。
- 下載最新的 Operator YAML 檔案後，在部署之前，將 *agent_backup.yaml* 中找到的任何自訂項目移植到下載的 *operator-config.yaml*。
- 確保您[拉取最新的容器鏡像](#)如果您使用自訂儲存庫。

停止並啟動 Kubernetes 監控操作員

要停止 Kubernetes Monitoring Operator：

```
kubectl -n netapp-monitoring scale deploy monitoring-operator
--replicas=0
```

啟動 Kubernetes Monitoring Operator：

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

解除安裝

刪除 Kubernetes Monitoring Operator

請注意，Kubernetes Monitoring Operator 的預設命名空間是「netapp-monitoring」。如果您設定了自己的命名空間，請在這些命令和所有後續命令和檔案中取代該命名空間。

可以使用以下命令卸載較新版本的監控操作員：

```
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>
```

如果監控操作員部署在其自己的專用命名空間中，請刪除該命名空間：

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

注意：如果第一個命令返回“未找到資源”，請使用以下說明卸載舊版本的監控操作員。

按順序執行以下每個命令。根據您目前的安裝，其中一些命令可能會傳回「未找到對象」訊息。您可以安全地忽略這些訊息。

```
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>
```

如果先前建立了安全上下文約束：

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

關於 **Kube-state-metrics**

NetApp Kubernetes Monitoring Operator 安裝自己的 kube-state-metrics 以避免與任何其他實例發生衝突。

有關 Kube-State-Metrics 的信息，請參閱["本頁"](#)。

配置/自訂操作員

這些部分包含有關自訂操作員配置、使用代理程式、使用自訂或私人 docker 儲存庫或使用 OpenShift 的資訊。

配置選項

最常修改的設定可以在 `_AgentConfiguration_` 自訂資源中配置。您可以在部署操作員之前透過編輯 `operator-config.yaml` 檔案來編輯此資源。該文件包含已註解掉的設定範例。查看列表["可用設定"](#)以取得最新版本的操作員。

您也可以部署操作員後使用以下命令編輯此資源：

```
kubectl -n netapp-monitoring edit AgentConfiguration
若要確定已部署的 operator 版本是否支援 _AgentConfiguration_，請執行下列命令：
```

```
kubectl get crd agentconfigurations.monitoring.netapp.com
如果您看到「伺服器錯誤（未找到）」訊息，則必須先升級您的操作員才能使用
AgentConfiguration。
```

配置代理支援

您可以在租戶的兩個地方使用代理程式來安裝 Kubernetes Monitoring Operator。這些可能是相同或獨立的代理系統：

- 執行安裝程式碼片段（使用“curl”）期間需要代理，以將執行程式碼片段的系統連接到您的Data Infrastructure Insights環境
- 目標 Kubernetes 叢集與您的Data Infrastructure Insights環境通訊所需的代理

如果您對其中一個或兩個都使用代理，為了安裝 Kubernetes 操作監視器，您必須先確保您的代理程式配置為允許與您的Data Infrastructure Insights環境進行良好的通訊。如果您有代理並且可以從您希望安裝 Operator 的伺服器/VM 存取Data Infrastructure Insights，那麼您的代理可能配置正確。

對於用於安裝 Kubernetes Operating Monitor 的代理，在安裝 Operator 之前，請設定 `http_proxy/https_proxy` 環境變數。對於某些代理環境，您可能還需要設定 `no_proxy environment` 變數。

若要設定變量，請在安裝 Kubernetes Monitoring Operator*之前*在系統上執行下列步驟：

1. 為目前使用者設定 `https_proxy` 和/或 `http_proxy` 環境變數：
 - a. 如果正在設定的代理程式沒有身份驗證（使用者名稱/密碼），請執行以下命令：

```
export https_proxy=<proxy_server>:<proxy_port>
.. 如果正在設定的代理確實具有身份驗證（使用者名稱/密碼），請執行以下命令：
```

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

對於用於 Kubernetes 叢集與Data Infrastructure Insights環境通訊的代理，請在閱讀所有這些說明後安裝 Kubernetes 監控操作員。

在部署 Kubernetes Monitoring Operator 之前，請先設定 `operator-config.yaml` 中 `AgentConfiguration` 的 `proxy` 區段。

```

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
  ...
  ...

```

使用自訂或私有的 **Docker** 倉庫

預設情況下，Kubernetes Monitoring Operator 將從Data Infrastructure Insights儲存庫中提取容器映像。如果您有 Kubernetes 叢集作為監控目標，且該叢集配置為僅從自訂或私人 Docker 儲存庫或容器註冊表中提取容器映像，則必須配置對 Kubernetes 監控操作員所需容器的存取權。

從NetApp Monitoring Operator 安裝圖塊運行「Image Pull Snippet」。此命令將登入Data Infrastructure Insights儲存庫，為操作員提取所有影像依賴項，並登出Data Infrastructure Insights儲存庫。出現提示時，輸入提供的儲存庫臨時密碼。此命令下載操作員使用的所有影像，包括選用功能。請參閱下文以了解這些圖像的用途。

核心 Operator 功能和 Kubernetes 監控

- netapp-監控
- ci-kube-rbac-代理
- ci-ksm
- 西電訊報
- distroless-root 用戶

事件日誌

- ci-fluent-bit
- ci-kubernetes-事件導出器

網路效能和地圖

- ci-net-觀察者

根據您的公司政策將操作員 docker 映像推送到您的私人/本地/企業 docker 儲存庫。確保儲存庫中這些圖像的圖像標籤和目錄路徑與Data Infrastructure Insights儲存庫中的一致。

編輯 `operator-deployment.yaml` 中的 `monitoring-operator` 部署，並修改所有映像引用以使用您的私人 Docker 儲存庫。

```
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>
```

編輯 `operator-config.yaml` 中的 `AgentConfiguration`，以反映新的 docker repo 位置。為您的私人儲存庫建立新的 `imagePullSecret`，更多詳細資訊請參閱 <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
```

長期密碼的 API 存取權杖

某些環境（例如 Proxy 儲存庫）需要 Data Infrastructure Insights docker 儲存庫的長期密碼。安裝時在 UI 中提供的密碼僅有效 24 小時。您可以使用 API Access Token 作為 docker 儲存庫密碼，而不使用該密碼。只要 API Access Token 有效，此密碼就會一直有效。您可以為此特定目的產生新的 API Access Token，或使用現有的 API Access Token。

"[在此閱讀](#)" 以取得建立新 API 存取權杖的說明。

若要從下載的 `operator-secrets.yaml` 檔案中提取現有的 API 存取令牌，使用者可以執行下列命令：

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

若要從正在執行的 operator 安裝中提取現有的 API 存取令牌，使用者可以執行以下命令：

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

OpenShift 說明

如果您執行的是 OpenShift 4.6 或更高版本，則必須編輯 `operator-config.yaml` 中的 `AgentConfiguration` 以啟用 `runPrivileged` 設定：

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

Openshift 可能會實施額外的安全級別，從而阻止對某些 Kubernetes 元件的存取。

容忍度和污點

`netapp-ci-tegraf-ds`、`netapp-ci-fluent-bit-ds` 和 `netapp-ci-net-observer-l4-ds` DaemonSet 必須在叢集中的每個節點上安排一個 pod，以便正確收集所有節點上的資料。操作員已配置為容忍一些眾所周知的*污點*。如果您在節點上配置了任何自訂污點，從而阻止 Pod 在每個節點上運行，則可以為這些污點建立 *容忍度*[在 `AgentConfiguration` 中](#)。如果您已將自訂污點套用至叢集中的所有節點，則您也必須向操作員部署新增必要的容忍度，以允許調度和執行操作員 pod。

了解有關 Kubernetes 的更多信息["污點和容忍度"](#)。

[返回** NetApp Kubernetes 監控操作員安裝* 頁面"](#)

關於秘密的說明

若要刪除 Kubernetes Monitoring Operator 查看叢集範圍機密的權限，請在安裝之前從 `operator-setup.yaml` 檔案中刪除下列資源：

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

如果這是升級，也請從叢集中刪除資源：

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

如果啟用了變更分析，請修改 `AgentConfiguration` 或 `operator-config.yaml` 以取消註解變更管理部分，並在變更管理部分下包含 `kindsToIgnoreFromWatch: "secrets"`。請注意此行中單引號和雙引號的存在和位置。

```
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"'
  ...
```

驗證 Kubernetes 監控 Operator 鏡像簽名

操作員的映像及其部署的所有相關映像均由NetApp簽署。您可以在安裝前使用 `cosign` 工具手動驗證映像，或設定 Kubernetes 准入控制器。如欲了解更多詳情，請參閱["Kubernetes 文檔"](#)。

用於驗證鏡像簽名的公鑰可在「監控操作員」安裝磁貼中找到，位於「可選：將操作員鏡像上傳到您的私人儲存庫 > 鏡像簽署公鑰」下

若要手動驗證影像簽名，請執行下列步驟：

1. 複製並運行圖像拉取片段
2. 出現提示時複製並輸入儲存庫密碼
3. 儲存圖片簽署公鑰（範例中為 `dii-image-signing.pub`）
4. 使用 `cosign` 驗證映像。請參閱以下 `cosign` 用法範例

```
$ 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}]
```

故障排除

如果您在設定 Kubernetes Monitoring Operator 時遇到問題，請嘗試以下操作：

<p>問題：</p>	<p>試試一下：</p>
<p>我沒有看到我的 Kubernetes 持久捲和相應的後端儲存裝置之間的超連結/連接。我的 Kubernetes 持久性磁碟區是使用儲存伺服器的主機名稱配置的。</p>	<p>請依照步驟卸載現有的 Telegraf 代理，然後重新安裝最新的 Telegraf 代理程式。您必須使用 Telegraf 2.0 或更高版本，並且您的 Kubernetes 叢集儲存必須由 Data Infrastructure Insights 主動監控。</p>
<p>我在日誌中看到類似以下內容的訊息：E0901 15:21:39.962145 1 reflector.go:178] k8s.io/kube-state-metrics/internal/store/builder.go:352: 無法列出*v1.MutatingWebhooks/internal/store/builder.go:352: 無法列出*v1.MutatingWebhookConfigurationWeb 1 reflector.go:178] k8s.io/kube-state-metrics/internal/store/builder.go:352: 無法列出*v1.Lease: 伺服器找不到要求的資源 (取得leases.coordination.k8s.io) 等。</p>	<p>如果您執行 kube-state-metrics 版本 2.0.0 或更高版本且 Kubernetes 版本低於 1.20，則可能會出現這些訊息。取得 Kubernetes 版本：<code>kubectl version</code> 取得 kube-state-metrics 版本：<code>kubectl get deploy/kube-state-metrics -o jsonpath='{..image}'</code> 為了防止這些訊息，使用者可以修改其 kube-state-metrics 部署以停用下列租約</p> <p>：<code>podDisruptions, mutatingWebhooks, mutatingWebhookConfigurations, volumeAttachments, resources</code> 更具體地說，他們可以使用以下 CLI 參數</p> <p>：<code>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</code> 預設資源清單為：「<code>certificatesigningrequests、configmaps、cronjobs、daemonsets、deployments、endpoints、horizontalpodautoscalers、ingresses、jobs、leases、limitranges、mutatingwebhookconfigurations、namespaces、networkpolicies、nodes、persistentvolumeclaims、persistentvolumes、poddisruptionbudgets、pods、replicasets、replicationcontrollers、resourcequotas、secrets、services、statefulsets、storageclasses、validatingwebhookconfigurations、volumeattachments</code>」</p>
<p>我看到 Telegraf 發出類似以下內容的錯誤訊息，但 Telegraf 確實啟動並運行：10月11日 14:23:41 ip-172-31-39-47 systemd[1]: 已啟動用於將指標報告到 InfluxDB 的插件驅動的伺服器代理。10月11日 14:23:41 ip-172-31-39-47 telegraf[1827]: time="2021-10-11T14:23:41Z" level=error msg="無法建立快取目錄。 /etc/telegraf/.cache/snowflake，錯誤：mkdir /etc/telegraf/.cache：權限被拒絕。 ignored\n" func="gosnowflake.(*defaultLogger).Errorf" file="log.go:120" 10月11日 14:23:41 ip-172-31-39-47 telegraf[1827]: time="2021-10-11T14:23:41Z" level=ignore msg="開啟 /etc/telegraf/.cache/snowflake/ocsp_response_cache.json：沒有這樣的檔案或目錄\n" func="gosnowflake.(*defaultLogger).Errorf" file="log.go:120" 10月11日 14:23:41 ip-172-31-39-47 telegraf[1827]: 2021-10-11T14:23:41Z I I 啟動 Telegraf 1.19.3</p>	<p>這是一個已知問題。參考這篇 GitHub 文章了解更多詳情。只要 Telegraf 正常運行，用戶就可以忽略這些錯誤訊息。</p>

<p>問題：</p>	<p>試試一下：</p>
<p>在 Kubernetes 上，我的 Telegraf pod 會報告以下錯誤： “處理 mountstats 訊息時出錯：無法開啟 mountstats 檔案：/hostfs/proc/1/mountstats，錯誤：開啟 /hostfs/proc/1/mountstats：權限被拒絕”</p>	<p>如果啟用並強制執行 SELinux，則可能會阻止 Telegraf pod 存取 Kubernetes 節點上的 /proc/1/mountstats 檔案。若要克服此限制，請編輯代理程式配置並啟用 runPrivileged 設定。有關更多詳細信息，請參閱 OpenShift 說明。</p>
<p>在 Kubernetes 上，我的 Telegraf ReplicaSet pod 會報告以下錯誤： [inputs.prometheus] 外掛程式錯誤：無法載入金鑰對 /etc/kubernetes/pki/etcd/server.crt ：/etc/kubernetes/pki/etcd/server.key:開啟 /etc/k11/kubernetes/pki/etcd/server.key:開啟 /etc/k1/1/p1/F11572 目錄：</p>	<p>Telegraf ReplicaSet pod 旨在指定為主節點或 etcd 的節點上運作。如果 ReplicaSet pod 沒有在其中一個節點上執行，您將會收到這些錯誤。檢查您的 master/etcd 節點是否有污點。如果確實如此，請在 Telegraf ReplicaSet、telegraf-rs 添加必要的容忍度。例如，編輯 ReplicaSet...kubect edit rs telegraf-rs...並將適當的容忍度加入規範。然後，重新啟動 ReplicaSet pod。</p>
<p>我有一個 PSP/PSA 環境。這會影響我的監控操作員嗎？</p>	<p>如果您的 Kubernetes 叢集正在執行 Pod 安全性原則 (PSP) 或 Pod 安全存取 (PSA)，則必須升級至最新的 Kubernetes 監控操作員。請依照以下步驟升級至支援 PSP/PSA 的目前業者： 1.解除安裝上一個監控操作員： 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-roeteing.netapp clusterrolebinding agent-manager-rolebinding agent-proxy-rolebinding agent-cluster-admin-rolebinding 2.安裝監控操作員的最新版本。</p>
<p>我在嘗試部署操作員時遇到了問題，並且我正在使用 PSP/PSA。</p>	<p>1.使用以下命令編輯代理程式： kubectl -n <namespace> edit agent 2.將“security-policy-enabled”標記為“false”。這將停用 Pod 安全性原則和 Pod 安全准入並允許操作員部署。使用以下命令確認： kubectl get psp (應該顯示 Pod 安全性策略已刪除) kubectl get all -n <namespace></p>
<p>grep -i psp (應該顯示未找到任何內容)</p>	<p>出現“ImagePullBackoff”錯誤</p>
<p>如果您有自訂或私有的 docker 儲存庫，並且尚未配置 Kubernetes Monitoring Operator 以正確識別它，則可能會看到這些錯誤。閱讀更多關於自訂/私人 repo 的配置。</p>	<p>我的監控操作員部署出現了問題，目前文件無法幫助我解決該問題。</p>

<p>問題：</p> <p>擷取或以其他方式記錄以下命令的輸出，並聯絡技術支援團隊。</p> <pre> kubect1 -n netapp-monitoring get all kubect1 -n netapp-monitoring describe all kubect1 -n netapp-monitoring logs <monitoring-operator-pod> --all -containers=true kubect1 -n netapp-monitoring logs <telegraf-pod> --all -containers=true </pre>	<p>試試一下：</p> <p>Operator 命名空間中的 net-observer（工作負載圖） pod 處於 CrashLoopBackOff 狀態</p>
<p>這些 pod 對應於網路可觀測性的工作負載圖資料收集器。嘗試以下操作：</p> <ul style="list-style-type: none"> • 檢查其中一個 pod 的日誌以確認最低核心版本。例如： <pre> --- {ci-tenant-id": "your-tenant-id", "collector-cluster": "your-k8s-cluster-name", "environment": "prod", "level": "error", "msg": "驗證失敗。原因：核心版本 3.10.0 低於最低核心版本 4.18.0", "time": "2022-11-09T08:23:08Z"} --- </pre> • Net-observer pods 要求 Linux 核心版本至少為 4.18.0。使用命令“uname -r”檢查內核版本，並確保它們 >= 4.18.0 	<p>Pod 在 Operator 命名空間（預設值：netapp-monitoring）中運行，但 UI 中未顯示工作負載圖或查詢中的 Kubernetes 指標的數據</p>
<p>檢查K8S叢集節點上的時間設定。為了準確的稽核和數據報告，強烈建議使用網路時間協定 (NTP) 或簡單網路時間協定 (SNTP) 同步代理機器上的時間。</p>	<p>Operator 命名空間中的部分 net-observer pod 處於 Pending 狀態</p>
<p>Net-observer是一個DaemonSet，在k8s叢集的每個Node中執行一個pod。注意處於待處理狀態的 pod，並檢查它是否遇到 CPU 或記憶體資源問題。確保節點中具有所需的記憶體和 CPU。</p>	<p>安裝 Kubernetes Monitoring Operator 後，我立即在日誌中看到以下內容：[inputs.prometheus] 插件錯誤：向 http://kube-state-metrics.<namespace>.svc.cluster.local:8080/metrics 發出 HTTP 請求時出錯：獲取\http://kube-state-metrics.<namespace>.svc.cluster.local:8080/metrics：撥號 tcp：尋找 kube-state-metrics.<namespace>.svc.cluster.local：沒有這樣的主機</p>
<p>通常僅在安裝新操作員且 telegraf-rs pod 在 ksm pod 啟動之前啟動時才會看到此訊息。一旦所有 pod 都運行起來，這些訊息就會停止。</p>	<p>我確實沒有看到針對我的叢集中存在的 Kubernetes CronJobs 收集任何指標。</p>
<p>驗證你的 Kubernetes 版本（即 kubect1 version）。如果是 v1.20.x 或更低版本，這是一個預期的限制。使用 Kubernetes Monitoring Operator 部署的 kube-state-metrics 版本僅支援 v1.CronJob。在 Kubernetes 1.20.x 及以下版本中，CronJob 資源位於 v1beta.CronJob。因此，kube-state-metrics 無法找到 CronJob 資源。</p>	<p>安裝操作員後，telegraf-ds pod 進入 CrashLoopBackOff，pod 日誌顯示「su: Authentication failed」。</p>

<p>問題：</p> <p>編輯 <i>AgentConfiguration</i> 中的 <i>telegraf</i> 區段，並將 <i>dockerMetricCollectionEnabled</i> 設為 <i>false</i>。如需更多詳細資訊，請參閱 operator 的“配置選項”。... spec: ... telegraf: ...- name: docker run-mode: - DaemonSet substitutions: - key: DOCKER_UNIX_SOCKET_PLACEHOLDER value: unix:///run/docker.sock</p>	<p>試試一下：</p> <p>我在 Telegraf 日誌中看到類似以下內容的重複錯誤訊息：E! [agent] 寫入輸出時發生錯誤。http：發佈「\https://<tenant_url>/rest/v1/lake/ingest/influxdb」：超出上下文截止時間（等待標頭時超出 Client.Timeout）</p>
<p>編輯 <i>_AgentConfiguration_</i> 中的 <i>telegraf</i> 部分，並將 <i>_outputTimeout_</i> 增加到 10 秒。欲了解更多詳情，請參閱運營商的“配置選項”。</p>	<p>我缺少某些事件日誌的 <i>_involvedobject_</i> 資料。</p>
<p>確保您已按照“權限”上面的部分。</p>	<p>為什麼我看到兩個監控操作員 pod 正在運行，一個名為 <i>netapp-ci-monitoring-operator-<pod></i>，另一個名為 <i>monitoring-operator-<pod></i>？</p>
<p>自 2023 年 10 月 12 日起，Data Infrastructure Insights 已重構了 Operator，以便更好地服務我們的用戶；為了完全採用這些更改，您必須刪除舊的操作員和安裝新的。</p>	<p>我的 kubernetes 事件意外停止向 Data Infrastructure Insights 回報。</p>
<p>檢索事件導出器 pod 的名稱：</p> <pre> `kubectl -n netapp-monitoring get pods ` </pre>	<p>grep event-exporter</p>
<pre> awk '{print \$1}' </pre>	<pre> sed 's/event-exporter./event-exporter/' </pre> <p>它應該是“netapp-ci-event-exporter”或“event-exporter”。接下來編輯監控代理 <code>kubectl -n netapp-monitoring edit agent</code>，並設定 <code>LOG_FILE</code> 的值以反映上一個步驟中找到的適當的事件導出器 pod 名稱。更具體地說，<code>LOG_FILE</code> 應該設定為“<code>/var/log/containers/netapp-ci-event-exporter.log</code>”或“<code>/var/log/containers/event-exporter*.log</code>”</p> <pre> fluent-bit: ... - name: event-exporter-ci substitutions: - key: LOG_FILE values: - /var/log/containers/netapp-ci-event-exporter*.log </pre> <p>或者，也可以解除安裝和重新安裝代理人。</p>
<p>我看到 Kubernetes Monitoring Operator 部署的 pod 由於資源不足而崩潰。</p>	<p>參考 Kubernetes Monitoring Operator “配置選項”根據需要增加 CPU 和/或記憶體限制。</p>

問題：	試試一下：
缺少影像或配置無效導致 netapp-ci-kube-state-metrics pod 無法啟動或準備就緒。現在 StatefulSet 卡住了，配置更改沒有應用到 netapp-ci-kube-state-metrics pod。	StatefulSet 位於"破碎的"狀態。修復所有配置問題後，反彈 netapp-ci-kube-state-metrics pod。
執行 Kubernetes Operator 升級後，netapp-ci-kube-state-metrics pod 無法啟動，拋出 ErrImagePull（無法拉取影像）。	嘗試手動重置 pod。
在日誌分析下，我的 Kubernetes 叢集中觀察到「事件因超過 maxEventAgeSeconds 而被丟棄」訊息。	修改 Operator <i>agentconfiguration</i> ，將 <i>event-exporter-maxEventAgeSeconds</i> （即增加到 60s）、 <i>event-exporter-kubeQPS</i> （即增加到 100）和 <i>event-exporter-kubeBurst</i> （即增加到 500）增加。有關這些配置選項的更多詳細信息，請參閱"配置選項"頁。
Telegraf 因可鎖定記憶體不足而發出警告或崩潰。	嘗試增加底層作業系統/節點中 Telegraf 可鎖定記憶體的限制。如果無法增加限制，請修改 NKMO 代理程式配置並將 <i>unprotected</i> 設為 <i>true</i> 。這將指示 Telegraf 不要嘗試保留鎖定的記憶體頁面。雖然這可能會帶來安全風險，因為解密的秘密可能會被交換到磁碟，但它允許在無法保留鎖定記憶體的環境中執行。有關 <i>unprotected</i> 配置選項的更多詳細信息，請參閱"配置選項"頁。
我看到來自 Telegraf 的類似以下內容的警告訊息： W! [inputs.diskio] 無法收集「vdc」的磁碟名稱：讀取 /dev/vdc 時發生錯誤：沒有此檔案或目錄	對於 Kubernetes Monitoring Operator 而言，這些警告訊息是良性的，可以安全地忽略。或者，在 AgentConfiguration 中編輯 telegraf 區段，並將 <i>runDsPrivileged</i> 設為 <i>true</i> 。如需更多詳細資訊，請參閱 "操作員的配置選項"。

<p>問題：</p> <p>我的 fluent-bit pod 出現以下錯誤：[2024/10/16 14:16:23] [錯誤] [/src/fluent-bit/plugins/in_tail/tail_fs_inotify.c:360 errno=24] 開啟的檔案太多 [2024/10/16 14:16:23] [錯誤] [引擎] 輸入初始化失敗</p>	<p>試試一下：</p> <p>嘗試更改叢集中的 <i>fsnotify</i> 設定：</p> <pre> 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> </pre> <p>重新啟動 Fluent-bit。</p> <p>注意：為了讓這些設定在節點重新啟動後仍然有效，您需要在 <code>/etc/sysctl.conf</code> 中新增以下幾行</p> <pre> fs.inotify.max_user_instances=<something larger than current setting> fs.inotify.max_user_watches=<something larger than current setting> </pre>
<p>telegraf DS pods 報告與 kubernetes 輸入外掛程式相關的錯誤，由於無法驗證 TLS 憑證而無法發出 HTTP 要求。例如：E! [inputs.kubernetes] 外掛程式錯誤：向下列物件發出 HTTP 請求時發生錯誤"<a "="" &lt;kubelet_ip&gt;="" &lt;kubelet_ip&gt;;:10250="" >https:="" :="" a>tls：無法驗證證書：x509：無法驗證="" a>得到"<a="" class="bare" href="https://&lt;kubelet_IP&gt;;:10250/stats/summary" ip="" p="" san<="" stats="" summary":<="" 的證書，因為它不包含任何=""> </p>	<p>如果 kubelet 使用自簽名證書，和/或指定的證書未在證書 <code>_Subject Alternative Name_</code> 清單中包含 <code><kubelet_IP></code>，則會發生這種情況。為了解決這個問題，使用者可以修改"代理配置"，並將 <code>telegraf.insecureK8sSkipVerify</code> 設為 <code>true</code>。這將配置 telegraf 輸入插件以跳過驗證。或者，使用者可以設定 kubelet "伺服器 TLS Bootstrap"，這將觸發來自「<code>certificates.k8s.io</code>」API 的憑證請求。</p>

<p>問題：</p> <p>我在 Fluent-bit Pod 中收到以下錯誤，且 Pod 無法啟動：2026/01/12 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</p>	<p>試試一下：</p> <p>確保資料庫檔案所在的主機目錄具有適當的讀取 / 寫入權限。更具體地說、主機目錄應授予非 root 使用者讀取 / 寫入權限。預設的資料庫檔案位置為 /var/log/、除非被 fluent-bit-dbFile <i>agentconfiguration</i> 選項覆寫。如果啟用 SELinux、請嘗試將 fluent-bit-seLinuxOptionsType agentconfiguration 選項設為 'spc_t'</p>
--	--

更多資訊可從["支援"](#)頁面或在["數據收集器支援矩陣"](#)。

Memcached 資料收集器

Data Infrastructure Insights使用此數據收集器從 Memcached 收集指標。

安裝

1. 從*可觀察性 > 收集器*中，按一下*+資料收集器*。選擇 Memcached。

選擇安裝 Telegraf 代理程式的作業系統或平台。

2. 如果您尚未安裝用於收集的代理，或者您希望為不同的作業系統或平台安裝代理，請按一下「顯示說明」以展開["代理安裝"](#)指示。
3. 選擇用於此資料收集器的代理存取金鑰。您可以透過點擊 + 代理存取金鑰 按鈕新增新的代理存取金鑰。最佳實務：只有當您想要對資料收集器進行分組（例如按作業系統/平台）時才使用不同的代理存取金鑰。
4. 依照設定步驟配置資料收集器。說明根據您用於收集資料的作業系統或平台的類型而有所不同。



Memcached Configuration

Gathers Memcached metrics.

What Operating System or Platform Are You Using?

[Need Help?](#)

Windows

Select existing Agent Access Key or create a new one

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

+ Agent Access Key

*Please ensure that you have a Telegraf Agent in you environment before configuring. [Show Instructions](#)

Follow Configuration Steps

[Need Help?](#)

- 1 Copy the contents below into a new .conf file under the C:\Program Files\telegraf\telegraf.d\ folder. For example, copy the contents to the C:\Program Files\telegraf\telegraf.d\cloudinsights-memcached.conf file.

```
[[inputs.memcached]]
  ## USER-ACTION: Provide comma-separated list of Memcached IP(s) and port(s).
  ## Please specify actual machine IP address, and refrain from using a loopback address
  ## (i.e. localhost or 127.0.0.1).
  ## When configuring with multiple Memcached servers, enter them in the format ["server1"
```

- 2 Replace <INSERT_MEMCACHED_ADDRESS> with the applicable Memcached server address. Please specify a real machine address, and refrain from using a loopback address.
- 3 Replace <INSERT_MEMCACHED_PORT> with the applicable Memcached server port.
- 4 Restart the Telegraf service.

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

設定

資訊可以在"[Memcached 維基百科](#)"。

物體和計數器

收集以下物件及其計數器：

目的：	標識符：	屬性：	數據點：
Memcached	命名空間伺服器	節點IP 節點名稱	正在接受的連線已處理的驗證請求失敗的身份驗證已使用的位元組數讀取的位元組數（每秒）寫入的位元組數（每秒）CAS BadvalCAS 命中CAS未命中刷新請求（每秒）獲取請求（每秒）設定請求（每秒）觸摸請求（每秒）連接產量（每秒）連接結構打開的連接當前存儲的項目遞減請求命中數（每秒）遞減請求未命中數（每秒）刪除請求命中數（每秒）刪除請求未命中數（每秒）驅逐的項目有效驅逐過期的項目獲取命中數（每秒）獲取未命中數（每秒）已使用的哈希字節數哈希正在擴展哈希功率級別遞增請求命中數（每秒）遞增請求未命中數（每秒）伺服器最大字節數偵聽已禁用已回收工作線程數總計打開的連接數儲存的項目總數 觸控命中數 觸碰未命中數 伺服器正常運作時間

故障排除

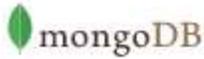
更多資訊可從["支援"](#)頁。

MongoDB 資料收集器

Data Infrastructure Insights使用此資料收集器從 MongoDB 收集指標。

安裝

1. 從*可觀察性 > 收集器*中，按一下*+資料收集器*。選擇 MongoDB。
- 選擇安裝 Telegraf 代理程式的作業系統或平台。
2. 如果您尚未安裝用於收集的代理，或者您希望為不同的作業系統或平台安裝代理，請按一下「顯示說明」以展開["代理安裝"](#)指示。
3. 選擇用於此資料收集器的代理存取金鑰。您可以透過點擊 + 代理存取金鑰 按鈕新增新的代理存取金鑰。最佳實務：只有當您想要對資料收集器進行分組（例如按作業系統/平台）時才使用不同的代理存取金鑰。
4. 依照設定步驟配置資料收集器。說明根據您用於收集資料的作業系統或平台的類型而有所不同。



MongoDB Configuration

Gathers MongoDB metrics.

What Operating System or Platform Are You Using?

[Need Help?](#)

RHEL & CentOS

Select existing Agent Access Key or create a new one

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

[+ Agent Access Key](#)

*Please ensure that you have a Telegraf Agent in you environment before configuring. [Show Instructions](#)

Follow Configuration Steps

[Need Help?](#)

- 1 Open mongod.conf. Locate the line beginning with "bindIp", and append the address of the node on which the Telegraf agent resides. After saving the change, restart the MongoDB server.
- 2 Copy the contents below into a new .conf file under the /etc/telegraf/telegraf.d/ directory. For example, copy the contents to the /etc/telegraf/telegraf.d/cloudinsights-mongodb.conf file.

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

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

```
systemctl restart telegraf
```

設定

資訊可以在"[MongoDB 文檔](#)"。

物體和計數器

收集以下物件及其計數器：

目的：	標識符：	屬性：	數據點：
MongoDB	命名空間主機名		
MongoDB 資料庫	命名空間 主機名稱 資料庫名稱		

故障排除

資訊可以從["支援"](#)頁。

MySQL資料收集器

Data Infrastructure Insights使用此資料收集器從 MySQL 收集指標。

安裝

1. 從*可觀察性 > 收集器*中，按一下*+資料收集器*。選擇 MySQL。
- 選擇安裝 Telegraf 代理程式的作業系統或平台。
2. 如果您尚未安裝用於收集的代理，或者您希望為不同的作業系統或平台安裝代理，請按一下「顯示說明」以展開["代理安裝"](#)指示。
 3. 選擇用於此資料收集器的代理存取金鑰。您可以透過點擊 + 代理存取金鑰 按鈕新增新的代理存取金鑰。最佳實務：只有當您想要對資料收集器進行分組（例如按作業系統/平台）時才使用不同的代理存取金鑰。
 4. 依照設定步驟配置資料收集器。說明根據您用於收集資料的作業系統或平台的類型而有所不同。



MySQL Configuration

Gathers MySQL metrics.

What Operating System or Platform Are You Using?

[Need Help?](#)

Windows

Select existing Agent Access Key or create a new one

Default (405fb5ec-d4cb-4404-977b-71fa931e1ad3) [+ Agent Access Key](#)

*Please ensure that you have a Telegraf Agent in you environment before configuring. [Show Instructions](#)

Follow Configuration Steps

[Need Help?](#)

- 1 Copy the contents below into a new .conf file under the C:\Program Files\telegraf\telegraf.d\ folder. For example, copy the contents to the C:\Program Files\telegraf\telegraf.d\cloudinsights-mysql.conf file.

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

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

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

設定

資訊可以在"MySQL 文檔"。

物體和計數器

收集以下物件及其計數器：

目的：	標識符：	屬性：	數據點：
MySQL	命名空間 MySQL 伺服器	節點IP 節點名稱	中止的客戶端 (每秒) 中止的連線 (每秒) RX 位元組 (每秒) TX 位元組 (每秒) 命令管理 (每秒) 命令修改事件命令修改函數命令修改實例命令修改過程命令修改伺服器命令修改表命令修改表空間命令修改用戶命令分析命令分配給密鑰快取命令開始命令 Binlog密鑰快取命令開始命令 命令調用過程命令更改數據庫命令更改主命令更改複製過濾器命令檢查命令校驗和命令提交命令創建數據庫命令創建事件命令創建函數命令創建索引命令創建過程命令創建服務器命令創建表命令創建觸發器命令創建 UDF 命令創建用戶命令創建視圖命令釋放 SQL 連接錯誤接受已創建的臨時磁盤表延遲錯誤刷新命令處理程序提交 Innodb緩衝池位元組資料未刷新的密鑰區塊密鑰讀取請求密鑰寫入請求密鑰寫入最大執行次數超出時間最大使用連接打開檔案效能模式帳戶丟失準備好的語句計數 Qcache 空閒區塊查詢問題選擇完全連接選擇全範圍連接選擇範圍檢查選擇掃描表鎖立即

故障排除

更多資訊可從["支援"](#)頁。

Netstat資料收集器

Data Infrastructure Insights使用此資料收集器來收集 Netstat 指標。

安裝

1. 從*可觀察性 > 收集器*中，按一下*+資料收集器*。選擇 Netstat。
選擇安裝 Telegraf 代理程式的作業系統或平台。
2. 如果您尚未安裝用於收集的代理，或者您希望為不同的作業系統或平台安裝代理，請按一下「顯示說明」以展開"代理安裝"指示。
3. 選擇用於此資料收集器的代理存取金鑰。您可以透過點擊 + 代理存取金鑰 按鈕新增新的代理存取金鑰。最佳實務：只有當您想要對資料收集器進行分組（例如按作業系統/平台）時才使用不同的代理存取金鑰。
4. 依照設定步驟配置資料收集器。說明根據您用於收集資料的作業系統或平台的類型而有所不同。



Netstat Configuration

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

What Operating System or Platform Are You Using?

[Need Help?](#)

Windows

Select existing Agent Access Key or create a new one

Default (405fb5ec-d4cb-4404-977b-71fa931e1ad3) [+ Agent Access Key](#)

*Please ensure that you have a Telegraf Agent in you environment before configuring. [Show Instructions](#)

Follow Configuration Steps

[Need Help?](#)

- 1 Copy the contents below into a new .conf file under the C:\Program Files\telegraf\telegraf.d\ folder. For example, copy the contents to the C:\Program Files\telegraf\telegraf.d\cloudinsights-netstat.conf file.

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

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

設定

物體和計數器

收集以下物件及其計數器：

目的：	標識符：	屬性：	數據點：
網路狀態	節點UUID	節點IP 節點名稱	

故障排除

更多資訊可從["支援"](#)頁。

Nginx 資料收集器

Data Infrastructure Insights使用此資料收集器從 Nginx 收集指標。

安裝

1. 從*可觀察性 > 收集器*中，按一下*+資料收集器*。選擇 Nginx。
 - 選擇安裝 Telegraf 代理程式的作業系統或平台。
2. 如果您尚未安裝用於收集的代理，或者您希望為不同的作業系統或平台安裝代理，請按一下「顯示說明」以展開["代理安裝"](#)指示。
3. 選擇用於此資料收集器的代理存取金鑰。您可以透過點擊 + 代理存取金鑰 按鈕新增新的代理存取金鑰。最佳實務：只有當您想要對資料收集器進行分組（例如按作業系統/平台）時才使用不同的代理存取金鑰。
4. 依照設定步驟配置資料收集器。說明根據您用於收集資料的作業系統或平台的類型而有所不同。



Nginx Configuration

Gathers Nginx metrics.

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

⊕ Ubuntu & Debian

Select existing Agent Access Key or create a new one

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

+ Agent Access Key

*Please ensure that you have a Telegraf Agent in you environment before configuring. [Show Instructions](#)

Follow Configuration Steps

[Need Help?](#)

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

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

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

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

- 4 Reload the configuration:

```
nginx -s reload
```

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

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

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

```
systemctl restart telegraf
```

設定

Nginx 指標收集需要 Nginx "[http_stub_status_module](#)" 被啟用。

更多資訊請參閱 "[Nginx 文檔](#)"。

物體和計數器

收集以下物件及其計數器：

目的：	標識符：	屬性：	數據點：
Nginx	命名空間伺服器	節點 IP 節點名稱 連接埠	接受活動已處理讀取請求 等待寫入

故障排除

更多資訊可從["支援"](#)頁。

PostgreSQL 資料收集器

Data Infrastructure Insights使用此資料收集器從 PostgreSQL 收集指標。

安裝

1. 從*可觀察性 > 收集器*中，按一下*+資料收集器*。選擇 PostgreSQL。
- 選擇安裝 Telegraf 代理程式的作業系統或平台。
2. 如果您尚未安裝用於收集的代理，或者您希望為不同的作業系統或平台安裝代理，請按一下「顯示說明」以展開["代理安裝"](#)指示。
3. 選擇用於此資料收集器的代理存取金鑰。您可以透過點擊 + 代理存取金鑰 按鈕新增新的代理存取金鑰。最佳實務：只有當您想要對資料收集器進行分組（例如按作業系統/平台）時才使用不同的代理存取金鑰。
4. 依照設定步驟配置資料收集器。說明根據您用於收集資料的作業系統或平台的類型而有所不同。



PostgreSQL Configuration

Gathers PostgreSQL metrics.

What Operating System or Platform Are You Using?

[Need Help?](#)

RHEL & CentOS

Select existing Agent Access Key or create a new one

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

+ Agent Access Key

*Please ensure that you have a Telegraf Agent in you environment before configuring [Show Instructions](#)

Follow Configuration Steps

[Need Help?](#)

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

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

- 2 Replace <INSERT_USERNAME> and <INSERT_PASSWORD> with the applicable PostgreSQL credentials.
- 3 Replace <INSERT_POSTGRESQL_ADDRESS> with the applicable PostgreSQL address. Please specify a real machine address, and refrain from using a loopback address.
- 4 Replace <INSERT_POSTGRESQL_PORT> with the applicable PostgreSQL port.
- 5 Replace <INSERT_DB> with the applicable PostgreSQL database.
- 6 Modify 'Namespace' if needed for server disambiguation (to avoid name clashes).
- 7 Restart the Telegraf service.

```
systemctl restart telegraf
```

設定

資訊可以在"PostgreSQL 文檔"。

物體和計數器

收集以下物件及其計數器：

目的：	標識符：	屬性：	數據點：
PostgreSQL 伺服器	命名空間資料庫伺服器	節點名稱 節點IP	緩衝區 已指派緩衝區 後端緩衝區 後端檔案同步緩衝區 檢查點緩衝區 清理檢查點 同步時間檢查點 寫入時間檢查點 請求檢查點 定時最大寫入清理
PostgreSQL 資料庫	命名空間資料庫伺服器	資料庫 OID 節點名稱 節點IP	區塊讀取時間區塊寫入時間區塊命中區塊讀取衝突死鎖客戶端數量臨時檔案位元組臨時檔案數量行刪除行取得行插入行返回行更新交易提交事務回滾

故障排除

更多資訊可從["支援"](#)頁。

Puppet代理程式資料收集器

Data Infrastructure Insights使用此資料收集器從 Puppet Agent 收集指標。

安裝

1. 從*可觀察性 > 收集器*中，按一下*+資料收集器*。選擇 Puppet。
選擇安裝 Telegraf 代理程式的作業系統或平台。
2. 如果您尚未安裝用於收集的代理，或者您希望為不同的作業系統或平台安裝代理，請按一下「顯示說明」以展開["代理安裝"](#)指示。
3. 選擇用於此資料收集器的代理存取金鑰。您可以透過點擊 + 代理存取金鑰 按鈕新增新的代理存取金鑰。最佳實務：只有當您想要對資料收集器進行分組（例如按作業系統/平台）時才使用不同的代理存取金鑰。
4. 依照設定步驟配置資料收集器。說明根據您用於收集資料的作業系統或平台的類型而有所不同。



Puppet Agent Configuration

Gathers Puppet agent metrics.

What Operating System or Platform Are You Using?

[Need Help?](#)

Windows

Select existing Agent Access Key or create a new one

Default (405fb5ec-d4cb-4404-977b-71fa931e1ad3) [+ Agent Access Key](#)

*Please ensure that you have a Telegraf Agent in you environment before configuring [Show Instructions](#)

Follow Configuration Steps

[Need Help?](#)

- 1 Copy the contents below into a new .conf file under the C:\Program Files\telegraf\telegraf.d\ folder. For example, copy the contents to the C:\Program Files\telegraf\telegraf.d\cloudinsights-puppetagent.conf file.

```
## Reads last_run_summary.yaml file and converts to measurements
[[inputs.puppetagent]]
  ## Location of puppet last run summary file
  ## USER-ACTION: Modify the location if last_run_summary.yaml is on different path
  location = "/var/lib/puppet/state/last_run_summary.yaml"
```

- 2 Modify 'location' if last_run_summary.yaml is on different path
- 3 Modify 'Namespace' if needed for puppet agent disambiguation (to avoid name clashes).
- 4 Restart the Telegraf service.

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

設定

資訊可以在 "[Puppet 文檔](#)"

物體和計數器

收集以下物件及其計數器：

目的：	標識符：	屬性：	數據點：
-----	------	-----	------

傀儡代理	命名空間節點 UUID	節點名稱 位置 節點 IP 版本 設定字串版本 Puppet	變更總數 事件數 失敗事件數 成功事件數 總資源數 已變更資源數 失敗資源數 重新啟動失敗資源數 不同步資源數 重新啟動資源數 已排程資源數 跳過資源數 總時間 錨點時間 設定檢索時間 Cron 時間 執行時間 時間資料桶時間 時間執行時間 包時間 排程時間 服務時間 Suthorizedized 的時間資料庫時間
------	-------------	--------------------------------	--

故障排除

更多資訊可從["支援"](#)頁。

Redis 資料收集器

Data Infrastructure Insights使用此資料收集器從 Redis 收集指標。Redis 是一個開源的記憶體資料結構存儲，用作資料庫、快取和訊息代理，支援以下資料結構：字串、哈希、列表、集合等。

安裝

1. 從*可觀察性 > 收集器*中，按一下*+資料收集器*。選擇 Redis。
 - 選擇安裝 Telegraf 代理程式的作業系統或平台。
2. 如果您尚未安裝用於收集的代理，或者您希望為不同的作業系統或平台安裝代理，請按一下「顯示說明」以展開["代理安裝"](#)指示。
3. 選擇用於此資料收集器的代理存取金鑰。您可以透過點擊 + 代理存取金鑰 按鈕新增新的代理存取金鑰。最佳實務：只有當您想要對資料收集器進行分組（例如按作業系統/平台）時才使用不同的代理存取金鑰。
4. 依照設定步驟配置資料收集器。說明根據您用於收集資料的作業系統或平台的類型而有所不同。



Redis Configuration

Gathers Redis metrics.

What Operating System or Platform Are You Using?

[Need Help?](#)

Windows

Select existing Agent Access Key or create a new one

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

+ Agent Access Key

*Please ensure that you have a Telegraf Agent in you environment before configuring. [Show Instructions](#)

Follow Configuration Steps

[Need Help?](#)

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

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

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

```
bind 127.0.0.1 <NODE_IP_ADDRESS>
```

- 3 Copy the contents below into a new .conf file under the C:\Program Files\telegraf\telegraf.d\ folder. For example, copy the contents to the C:\Program Files\telegraf\telegraf.d\cloudinsights-redis.conf file.

```
# Read metrics from one or many redis servers
[[inputs.redis]]
  ## specify servers via a url matching:
  ## [protocol://][:password]@address[:port]
  ## e.g.
  ## http://username:password@127.0.0.1:6379
```

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

- 5 Replace <INSERT_REDIS_PORT> with the applicable Redis port.

- 6 Restart the Telegraf service.

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

設定

資訊可以在"Redis 文件"。

物體和計數器

收集以下物件及其計數器：

目的：	標識符：	屬性：	數據點：
Redis	命名空間伺服器		

故障排除

更多資訊可從["支援"](#)頁。

版權資訊

Copyright © 2026 NetApp, Inc. 版權所有。台灣印製。非經版權所有人事先書面同意，不得將本受版權保護文件的任何部分以任何形式或任何方法（圖形、電子或機械）重製，包括影印、錄影、錄音或儲存至電子檢索系統中。

由 NetApp 版權資料衍伸之軟體必須遵守下列授權和免責聲明：

此軟體以 NETAPP「原樣」提供，不含任何明示或暗示的擔保，包括但不限於有關適售性或特定目的適用性之擔保，特此聲明。於任何情況下，就任何已造成或基於任何理論上責任之直接性、間接性、附隨性、特殊性、懲罰性或衍生性損害（包括但不限於替代商品或服務之採購；使用、資料或利潤上的損失；或企業營運中斷），無論是在使用此軟體時以任何方式所產生的契約、嚴格責任或侵權行為（包括疏忽或其他）等方面，NetApp 概不負責，即使已被告知有前述損害存在之可能性亦然。

NetApp 保留隨時變更本文所述之任何產品的權利，恕不另行通知。NetApp 不承擔因使用本文所述之產品而產生的責任或義務，除非明確經過 NetApp 書面同意。使用或購買此產品並不會在依據任何專利權、商標權或任何其他 NetApp 智慧財產權的情況下轉讓授權。

本手冊所述之產品受到一項（含）以上的美國專利、國外專利或申請中專利所保障。

有限權利說明：政府機關的使用、複製或公開揭露須受 DFARS 252.227-7013（2014 年 2 月）和 FAR 52.227-19（2007 年 12 月）中的「技術資料權利 - 非商業項目」條款 (b)(3) 小段所述之限制。

此處所含屬於商業產品和 / 或商業服務（如 FAR 2.101 所定義）的資料均為 NetApp, Inc. 所有。根據本協議提供的所有 NetApp 技術資料和電腦軟體皆屬於商業性質，並且完全由私人出資開發。美國政府對於該資料具有非專屬、非轉讓、非轉授權、全球性、有限且不可撤銷的使用權限，僅限於美國政府為傳輸此資料所訂合約所允許之範圍，並基於履行該合約之目的方可使用。除非本文另有規定，否則未經 NetApp Inc. 事前書面許可，不得逕行使用、揭露、重製、修改、履行或展示該資料。美國政府授予國防部之許可權利，僅適用於 DFARS 條款 252.227-7015(b)（2014 年 2 月）所述權利。

商標資訊

NETAPP、NETAPP 標誌及 <http://www.netapp.com/TM> 所列之標章均為 NetApp, Inc. 的商標。文中所涉及的所有其他公司或產品名稱，均為其各自所有者的商標，不得侵犯。