



# 《数据收集器参考—服务》

## Data Infrastructure Insights

NetApp  
August 19, 2025

This PDF was generated from [https://docs.netapp.com/zh-cn/data-infrastructure-insights/task\\_config\\_telegraf\\_node.html](https://docs.netapp.com/zh-cn/data-infrastructure-insights/task_config_telegraf_node.html) on August 19, 2025. Always check [docs.netapp.com](https://docs.netapp.com) for the latest.

# 目录

《数据收集器参考—服务》	1
节点数据收集	1
安装	1
对象和计数器	1
设置	2
ActiveMQ 数据收集器	2
安装	3
设置	3
对象和计数器	3
故障排除	3
Apache Data Collector	3
安装	4
设置	4
对象和计数器	5
故障排除	5
Consul 数据收集器	5
安装	6
设置	6
用于转换的对象和计数器	6
故障排除	6
Couchbase 数据收集器	6
安装	6
设置	7
对象和计数器	7
故障排除	7
CouchDB Data Collector	7
安装	7
设置	7
对象和计数器	7
故障排除	8
Docker Data Collector	8
安装	8
设置	8
对象和计数器	9
故障排除	12
Elasticsearch Data Collector	13
设置	13
对象和计数器	13
故障排除	13

Flink 数据收集器	13
安装	13
设置	14
对象和计数器	15
故障排除	17
Hadoop Data Collector	17
安装	17
设置	18
对象和计数器	21
故障排除	22
HAProxy 数据收集器	22
安装	22
设置	22
对象和计数器	23
故障排除	25
JVM 数据收集器	25
安装	25
设置	25
对象和计数器	25
故障排除	26
Kafka 数据收集器	26
安装	27
设置	27
对象和计数器	28
故障排除	28
Kibana 数据收集器	28
安装	28
设置	28
对象和计数器	28
故障排除	29
Kubnetes Monitoring Operator 安装和配置	29
在安装Kubnetes Monitoring Operator之前	29
安装Kubnetes Monitoring Operator	29
Kubelnetes监控组件	31
升级到最新的Kubnetes Monitoring Operator	31
停止和启动Kubnetes Monitoring Operator	33
正在卸载	33
关于Kube-state-metrics	34
配置/自定义操作员	34
关于安全的注意事项	38
验证Kubnetes监控操作员图像签名	38

故障排除	39
内存缓存的 Data Collector	44
安装	44
设置	45
对象和计数器	46
故障排除	46
MongoDB 数据收集器	46
安装	46
设置	47
对象和计数器	48
故障排除	48
MySQL Data Collector	48
安装	48
设置	49
对象和计数器	50
故障排除	50
Netstat 数据收集器	50
安装	51
设置	51
对象和计数器	51
故障排除	52
nginx 数据收集器	52
安装	52
设置	53
对象和计数器	53
故障排除	54
PostgreSQL 数据收集器	54
安装	54
设置	55
对象和计数器	56
故障排除	56
Puppet Agent 数据收集器	56
安装	56
设置	57
对象和计数器	57
故障排除	58
Redis Data Collector	58
安装	58
设置	59
对象和计数器	60
故障排除	60

# 《数据收集器参考—服务》

## 节点数据收集

Data Infrastructure Insight从安装代理的节点收集指标。

### 安装

1. 从\*Observability > Collectors\*中，选择操作系统/平台。请注意，安装任何集成数据收集器（Kubernetes，Docker，Apache 等）也会配置节点数据收集。
2. 按照说明配置代理。这些说明因用于收集数据的操作系统或平台的类型而异。

### 对象和计数器

以下对象及其计数器将作为节点指标收集：

对象：	标识符：	属性：	数据点：
节点文件系统	节点 UUID 设备路径类型	节点 IP 节点名称节点操作系统模式	可用索引节点可用索引节点已用总索引节点已用总容量
节点磁盘	节点 UUID 磁盘	节点 IP 节点名称节点操作系统	IO 时间进行中的总 IOPS 读取字节（每秒） 读取时间 总读取时间（每秒） 加权 IO 时间 总写入字节（每秒） 写入时间 总写入时间 当前磁盘队列长度 写入时间 读取时间 IO 时间
节点CPU	节点 UUID CPU	节点 IP 节点名称节点操作系统	系统 CPU 使用情况 用户 CPU 使用情况 空闲 CPU 使用情况 处理器 CPU 使用情况 中断 CPU 使用情况 DPC CPU 使用情况

对象：	标识符：	属性：	数据点：
节点	节点UUID	节点 IP 节点名称节点操作系统	内核启动时间 内核上下文切换 (每秒) 内核 Entropy 可用内核中断 (每秒) 内核进程 forked (每秒) 内存活动 内存可用 总内存可用 内存缓存 内存缓存 内存 提交限制 已提交 内存已用 尽内存可用 内存高可用 内存大总内存 大页面大小 内存大页面 可用 内存大页面 总内存低 可用 内存低 总内存 映射 内存 页面 表 内存 共享 内存 Slab 内存 交换 缓存 可用 内存 交换 总内存 已用 总内存 已用 总内存 已用 内存 Vmalloc 区块 内存 Vmalloc 总内存 Vmalloc 已用 内存 已用 内存 已线 内存 回写 总内存 回写 温度 内存 缓存 故障 内存 需求 零 故障 内存 页面 故障 内存 页面 内存 页面 内存 页面 非 分页 内存 分页 内存 缓存 核心 内存 备用 缓存 正常 内存 备用 缓存 预留 内存 过渡 故障 进程 被阻止 的进程 停止 进程 转储 进程 运行 进程 休眠 进程 已停止 进程 总进程 线程 进程 未知 进程 Zombie 处理器 队列 长度 空闲 交换 总已用 交换 总容量 Swap Used Swap in Swap out System Uptime System NumCPU System NumUsers System 调用
节点网络	网络接口节点 UUID	节点名称节点 IP 节点操作系统	字节 接收 字节 发送 数据包 传出 丢弃 的数据包 传出 错误 数据包 接收 丢弃 的数据包 接收 错误 数据包 接收 发送 的数据包

## 设置

有关设置和故障排除的信息、请参见["配置代理"](#)页面。

## ActiveMQ 数据收集器

Data Infrastructure Insight 使用此数据收集器从 ActiveMQ 力象限中收集指标。

## 安装

- 从\*Observability > Collectors\*中，单击\*+Data Collector\*。选择Active魔力象限。

选择安装了 Telegraf 代理的操作系统或平台。

- 如果尚未安装用于收集的代理，或者要为其他操作系统或平台安装代理，请单击 Show Instructions 展开说明。["代理安装"](#)
- 选择要用于此数据收集器的代理访问密钥。您可以通过单击 \* + 代理访问密钥 \* 按钮来添加新的代理访问密钥。最佳实践：仅当您要按操作系统 / 平台对数据收集器进行分组时，才使用其他代理访问密钥。
- 按照配置步骤配置数据收集器。这些说明因用于收集数据的操作系统或平台的类型而异。

[ActiveMQ 配置]

## 设置

有关信息、请参见 ["ActiveMQ 文档"](#)

## 对象和计数器

将收集以下对象及其计数器：

对象：	标识符：	属性：	数据点：
ActiveMQ 队列	命名空间队列端口服务器	节点名称节点 IP 节点 UUID	使用者计数重复计数排队计数队列大小
ActiveMQ 订阅者	客户端 ID 连接 ID 端口服务器命名空间	是否为活动目标节点名称节点 IP 节点 UUID 节点操作系统选择器订阅	已发送的重复计数已发送的计数已发送的队列大小 Enqueue 计数待定队列大小
ActiveMQ 主题	主题端口服务器命名空间	节点名称节点 IP 节点 UUID 节点操作系统	使用者计数重复计数排队计数大小

## 故障排除

有关其他信息、请参见["支持"](#)页面。

# Apache Data Collector

此数据收集器允许从租户上的Apache服务器收集数据。

## 前提条件

- 您必须已设置 Apache HTTP 服务器并正确运行
- 您必须对代理主机 /VM 具有 sudo 或管理员权限
- 通常，Apache *mod\_status* 模块会配置为在 Apache 服务器的 /server-status ? auto 位置公开页面。要收集所有可用字段，必须启用 *ExtendedStatus* 选项。有关如何配置服务器的信息，请参见Apache模块文档：[https://httpd.apache.org/docs/2.4/mod/mod\\_status.html#enable](https://httpd.apache.org/docs/2.4/mod/mod_status.html#enable)

## 安装

- 从“Observability > Collectors”中，单击“+Data Collector”。选择Apache。

选择安装了 Telegraf 代理的操作系统或平台。

- 如果尚未安装用于收集的代理，或者要为其他操作系统或平台安装代理，请单击 Show Instructions 展开说明。[“代理安装”](#)
- 选择要用于此数据收集器的代理访问密钥。您可以通过单击 “+ 代理访问密钥” 按钮来添加新的代理访问密钥。最佳实践：仅当您要按操作系统 / 平台对数据收集器进行分组时，才使用其他代理访问密钥。
- 按照配置步骤配置数据收集器。这些说明因用于收集数据的操作系统或平台的类型而异。

## [Apache 配置]

## 设置

Telegraf 用于 Apache HTTP 服务器的插件依赖于 "mod\_status" 模块的启用。启用此选项后，Apache 的 HTTP 服务器将显示一个 HTML 端点，您可以在浏览器上查看此端点，也可以对其进行擦除，以便提取所有 Apache 的 HTTP 服务器配置的状态。

兼容性：

此配置是根据 Apache 的 HTTP 服务器版本 2.4.38 开发的。

正在启用 **mod\_status**：

启用和公开 "mod\_status" 模块涉及两个步骤：

- 正在启用模块
- 从模块中公开统计信息

启用模块：

模块的加载由 "/usr/local/apache/conf/httpd.conf" 下的配置文件控制。编辑此配置文件并取消注释以下行：

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

从模块公开统计信息：

"od\_status" 的公开由 "/usr/local/apache2/conf/ext/httpd-info.conf" 下的配置文件控制。请确保此配置文件中包含以下内容（至少还有其他指令）：

```

# 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

```

有关"mod\_status"模块的详细说明、请参见["Apache 文档"](#)

## 对象和计数器

将收集以下对象及其计数器：

对象：	标识符：	属性：	数据点：
Apache	命名空间服务器	节点 IP 节点名称端口父服务器配置生成父服务器 MPM 生成服务器正常运行时间正在停止	每秒繁忙工作负载字节每个请求字节每个 CPU 子系统 CPU 子用户 CPU 负载 CPU 系统 CPU 用户异步连接关闭异步连接保持活动异步连接写入连接每个请求的总持续时间空闲工作负载平均值（最后 1 米）负载平均值（最后 15 米）负载平均值（最后 5 米）每秒处理请求总访问总持续时间总计 KBytes 记分板关闭记分板 DNS 查找记分板完成记分板闲置清理记分板保持活动记分板开放式记分板读取记分板发送记分板开始等待记分板

## 故障排除

有关其他信息、请参见["支持"](#)页面。

## Consul 数据收集器

Data Infrastructure Insight 可使用此数据收集器从整合服务收集指标。

## 安装

- 从\*Observability > Collectors\*中，单击\*+Data Collector\*。选择"执行"。

如果尚未配置用于收集的Agent、系统将提示您在租户上配置["安装代理"](#)。

如果已配置代理，请选择相应的操作系统或平台，然后单击 \* 继续 \*。

- 按照 Consul Configuration 屏幕中的说明配置数据收集器。这些说明因用于收集数据的操作系统或平台的类型而异。

## 设置

有关信息，请参见["Consul 文档"](#)。

## 用于转换的对象和计数器

将收集以下对象及其计数器：

对象：	标识符：	属性：	数据点：
Consul	命名空间检查 ID 服务节点	节点 IP 节点操作系统节点 UUID 节点名称服务名称检查名称服务 ID 状态	严重通过警告

## 故障排除

有关其他信息、请参见["支持"](#)页面。

## Couchbase 数据收集器

Data Infrastructure Insight 使用此数据收集器从Cucchbase收集指标。

## 安装

- 从\*Observability > Collectors\*中，单击\*+Data Collector\*。选择Couchbase。

选择安装了 Telegraf 代理的操作系统或平台。

- 如果尚未安装用于收集的代理，或者要为其他操作系统或平台安装代理，请单击 Show Instructions 展开说明。["代理安装"](#)
- 选择要用于此数据收集器的代理访问密钥。您可以通过单击 \* + 代理访问密钥 \* 按钮来添加新的代理访问密钥。最佳实践：仅当您要按操作系统 / 平台对数据收集器进行分组时，才使用其他代理访问密钥。
- 按照配置步骤配置数据收集器。这些说明因用于收集数据的操作系统或平台的类型而异。

[Couchbase 配置]

## 设置

有关信息，请参见["Couchbase 文档"](#)。

## 对象和计数器

将收集以下对象及其计数器：

对象：	标识符：	属性：	数据点：
Couchbase 节点	命名空间集群 Couchbase 节点主机名	节点名称节点 IP	内存可用总内存
Couchbase 存储分段	命名空间分段集群	节点名称节点 IP	已用数据提取磁盘已用项 目计数已用内存操作 / 已 用配额

## 故障排除

有关其他信息、请参见["支持"](#)页面。

# CouchDB Data Collector

Data Infrastructure Insight 使用此数据收集器从 CouchDB 收集指标。

## 安装

1. 从 \*Observability > Collectors\* 中，单击 \*+Data Collector\*。选择 CouchDB。

选择安装了 Telegraf 代理的操作系统或平台。

2. 如果尚未安装用于收集的代理，或者要为其他操作系统或平台安装代理，请单击 Show Instructions 展开说明。["代理安装"](#)
3. 选择要用于此数据收集器的代理访问密钥。您可以通过单击 \* + 代理访问密钥 \* 按钮来添加新的代理访问密钥。最佳实践：仅当您要按操作系统 / 平台对数据收集器进行分组时，才使用其他代理访问密钥。
4. 按照配置步骤配置数据收集器。这些说明因用于收集数据的操作系统或平台的类型而异。

## [CouchDB 配置]

## 设置

有关信息，请参见["CouchDB 文档"](#)。

## 对象和计数器

将收集以下对象及其计数器：

对象：	标识符：	属性：	数据点：
CouchDB	命名空间服务器	节点名称节点 IP	身份验证缓存命中身份验证缓存未命中数据库读取数据库写入数据库打开打开操作系统文件最大请求时间最小请求时间 httpd 请求方法复制 httpd 请求方法删除 httpd 请求方法获取 httpd 请求方法头 httpd 请求方法 POST httpd 请求方法 PUT 状态代码 200 状态代码 201 状态代码 202 状态代码 301 状态代码 304 状态代码 400 状态代码 401 状态代码 403 状态代码 404 状态代码 405 状态代码 409 状态代码 4.12 状态代码 500

## 故障排除

有关其他信息、请参见["支持"](#)页面。

## Docker Data Collector

Data Infrastructure Insight 使用此数据收集器从 Docker 收集指标。

### 安装

- 从 \*Observability > Collectors\* 中，单击 \*+Data Collector\*。选择 Docker。

如果尚未配置用于收集的 Agent、系统将提示您在租户上配置["安装代理"](#)。

如果已配置代理，请选择相应的操作系统或平台，然后单击 \* 继续 \*。

- 按照 Docker 配置屏幕中的说明配置数据收集器。这些说明因用于收集数据的操作系统或平台的类型而异。

### [Docker 配置]

### 设置

适用于 Docker 的 Telegraf 输入插件通过指定的 UNIX 套接字或 TCP 端点收集指标。

### 兼容性

配置是根据 Docker 1.12.6 版开发的。

### 设置

## 通过 UNIX 套接字访问 Docker

如果 Telegraf 代理正在 BareMetal 上运行, 请运行以下命令将此 Telegraf Unix 用户添加到 Docker Unix 组:

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

如果 Telegraf 代理在 Kubernetes Pod 中运行, 请将此套接字映射到 Pod 中作为卷, 然后将此卷挂载到 /var/run/docker.sock , 以公开 Docker Unix 套接字。例如, 将以下内容添加到 PodSpec 中:

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

然后, 将以下内容添加到容器中:

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

请注意、为Kubernetes平台提供的Data Infrastructure Insight安装程序会自动执行此映射。

## 通过 TCP 端点访问 Docker

默认情况下, Docker 使用端口 2375 进行未加密访问, 而使用端口 2376 进行加密访问。

## 对象和计数器

将收集以下对象及其计数器:

对象:	标识符:	属性:	数据点:
Docker 引擎	命名空间 Docker 引擎	节点名称 节点 IP 节点 UUID 操作系统 Kubernetes 集群 Docker 版本 单元	Memory Containers Paused Containers Running Containers Stopped CPU Go 例程 图像侦听器事件 已用文件描述符 数据可用 数据总已用 数据可用元 数据已用元 数据总元 数据池块大小

对象：	标识符：	属性：	数据点：
Docker 容器	命名空间容器名称 Docker 引擎	Kubernetes 容器哈希 Kubernetes 容器端口 Kubernetes 容器重新启动 计数 Kubernetes 容器终止 消息路径 Kubernetes 容器 终止消息策略 Kubernetes Pod 终止宽限期容器映像 容器状态容器版本节点名 称 Kubernetes 容器日志路 径 Kubernetes 容器名称 Kubernetes Docker 类型 Kubernetes Pod 名称 Kubernetes Pod 名称 Kubernetes Pod 命名空间 POD UID Kubernetes 沙 盒 ID 节点 IP 节点 UUID Docker 版本 Kubernetes IO 配置已查看 Kubernetes IO 配置源 OpenShift IO SCC Kubernetes 问题描述 Kubernetes 显示名称 OpenShift 标记 Kompose Service Pod 模板哈希控制 器修订哈希模板生成许可 证架构构建日期架构许可 证架构名称架构 URL 架构 VCS URL 架构供应商架构 版本架构版本维护人员客 户 Pod Kubernetes StatusSet Pod Name Tenant WEBCONSOLE Architecture 权威源 URL 构建日期 RH 构建主机 RH 组件分发范围安装发行运 行摘要卸载 VCS 参考 VCS 类型供应商版本运行 状况容器 ID	内存活动匿名内存活动文 件内存缓存内存分层限制 内存非活动匿名内存非活 动文件内存限制内存映射 的文件内存最大使用量内 存页面故障内存页面内存 中的主要故障内存分页内 存驻留设置内存驻留设置 大小内存设置大小巨大内 存总活动量 匿名内存总活 动文件内存总缓存总非活 动匿名内存总非活动文件 内存总映射文件内存总页 面故障内存总页面主要故 障内存总分页内存总分页 内存总分页内存总驻留设 置大小内存总驻留设置大 小内存总内存总内存总内 存总内存总内存总内存总 内存总容量 Unevictable Memory Unevictable Memory Usage Memory Usage Percent Exit Code OOM Ked PID 在错误的数 据流时启动

对象：	标识符：	属性：	数据点：
Docker 容器块 IO	命名空间容器名称设备 Docker 引擎	Kubernetes 容器哈希 Kubernetes 容器端口 Kubernetes 容器重新启动 计数 Kubernetes 容器终止 消息路径 Kubernetes 容器 终止消息策略 Kubernetes Pod 终止宽限期容器映像 容器状态容器版本节点名 称 Kubernetes 容器日志路 径 Kubernetes 容器名称 Kubernetes Docker 类型 Kubernetes Pod 名称 Kubernetes Pod 名称 Kubernetes Pod 命名空间 POD UID Kubernetes 沙 盒 ID 节点 IP 节点 UUID Docker 版本 Kubernetes 配置已查看 Kubernetes 配 置源 OpenShift SCC Kubernetes 问题描述 Kubernetes 显示名称 OpenShift 标记架构版本 Pod 模板哈希控制器版本 哈希模板生成科姆服务架 构构建日期架构架构许 可证名称架构供应商客户 POD Kubernetes StatusSet Pod Name Tenant WEBCONSOLE Build Date License Vendor Architecture Authoritative Source URL RH Build Host RH Component Distribution Scope Install Maintainer Release Run Summary 卸载 VCS 参考 VCS 类型版本架构 URL 架构 VCS URL 架构版本 容器 ID	IO 服务字节数递归异步 IO 服务字节数递归读取 IO 服 务字节数递归同步 IO 服务 字节数递归总 IO 服务字节 数递归写入 IO 服务递归异 步异步 IO 服务递归读取 IO 服务的递归同步 IO 服 务的递归总 IO 服务的递归 写入
Docker 容器网络	命名空间容器名称网络 Docker 引擎	容器映像容器状态容器版 本节点名称节点 IP 节点 UUID 节点操作系统 K8s 集群 Docker 版本容器 ID	Rx 丢弃的 Rx 字节接收错 误接收数据包 Tx 丢弃的 TX 字节 Tx 错误 Tx 数据 包

对象：	标识符：	属性：	数据点：
Docker 容器 CPU	命名空间容器名称 CPU Docker 引擎	Kubernetes 容器哈希 Kubernetes 容器端口 Kubernetes 容器重新启动 计数 Kubernetes 容器终止 消息路径 Kubernetes 容器 终止消息策略 Kubernetes Pod 终止宽限期 Kubernetes 配置源 OpenShift SCC 容器映像 容器状态容器版本节点名 称 Kubernetes 容器日志路 径 Kubernetes 容器名称 Docker Kubernetes 容器 名称 键入 Kubernetes Pod 名称 Kubernetes Pod 命名 空间 Kubernetes Pod UID Kubernetes 沙盒 ID 节点 IP 节点 UUID 节点操作系 统 Kubernetes 集群 Docker 版本 Kubernetes 问题描述 Kubernetes 显示 名称 OpenShift 标记架构 版本模板哈希控制器版本 哈希 Pod 模板生成工具服 务架构构建日期架构许可 证架构名称 架构供应商客 户 Pod Kubernetes StatusSet Pod Name Tenant WEBCONSOLE Build Date License Vendor Architecture Authoritative Source URL RH Build Host RH Component Distribution Scope Install Maintainer Release Run Summary 卸载 VCS 参考 VCS Type 版本架构 URL 架构 VCS URL 架构 URL 架构版本容器 ID	限制期限限制限制期限限 制内核模式下的限制时间 使用在用户模式下的使用 量使用量百分比使用量系 统使用量总数

## 故障排除

问题：	请尝试以下操作：
按照配置页面上的说明操作后、我在Data Infrastructure Insight中看不到我的Docker指标。	检查 Telegraf 代理日志，查看它是否报告以下错误： e ! 插件 [inputs.docker] 中的错误：尝试连接到 Docker 守护进程套接字时被拒绝获得权限如果确实如此，请按照上述说明执行必要的步骤为 Telegraf 代理提供对 Docker Unix 套接字的访问权限。

有关其他信息、请参见["支持"](#)页面。

# Elasticsearch Data Collector

Data Infrastructure Insight 可使用此数据收集器从 Elasticsearch 中收集指标。

1. 从 \*Observability > Collectors\* 中，单击 \*+Data Collector\*。选择 EIASCHEARSEarch。  
选择安装了 Telegraf 代理的操作系统或平台。
2. 如果尚未安装用于收集的代理，或者要为其他操作系统或平台安装代理，请单击 Show Instructions 展开说明。[“代理安装”](#)
3. 选择要用于此数据收集器的代理访问密钥。您可以通过单击 \* + 代理访问密钥 \* 按钮来添加新的代理访问密钥。最佳实践：仅当您要按操作系统 / 平台对数据收集器进行分组时，才使用其他代理访问密钥。
4. 按照配置步骤配置数据收集器。这些说明因用于收集数据的操作系统或平台的类型而异。

[Elasticsearch 配置]

## 设置

有关信息，请参见[“Elasticsearch 文档”](#)。

## 对象和计数器

将收集以下对象及其计数器：

对象：	标识符：	属性：
Elasticsearch 集群	命名空间集群	节点 IP 节点名称集群状态
Elasticsearch 节点	命名空间集群 ES 节点 ID ES 节点 IP ES 节点	分区 ID

## 故障排除

有关其他信息，请参见[“支持”页面](#)。

# Flink 数据收集器

Data Infrastructure Insight 使用此数据收集器从 Flink 收集指标。

## 安装

1. 从 \*Observability > Collectors\* 中，单击 \*+Data Collector\*。选择“链接”。  
选择安装了 Telegraf 代理的操作系统或平台。
2. 如果尚未安装用于收集的代理，或者要为其他操作系统或平台安装代理，请单击 Show Instructions 展开说明。[“代理安装”](#)
3. 选择要用于此数据收集器的代理访问密钥。您可以通过单击 \* + 代理访问密钥 \* 按钮来添加新的代理访问密钥。最佳实践：仅当您要按操作系统 / 平台对数据收集器进行分组时，才使用其他代理访问密钥。

4. 按照配置步骤配置数据收集器。这些说明因用于收集数据的操作系统或平台的类型而异。

## [Flink 配置]

### 设置

完整的 Flink 部署包括以下组件：

JobManager： Flink 主系统。协调一系列 TaskManager。在高可用性设置中，系统将具有多个 JobManager。  
TaskManager：执行 Flink 运算符的位置。Flink 插件基于此电报的 JOLokia 插件。例如，需要从所有 Flink 组件收集信息，因此需要在所有组件上通过 Jallokia 配置并公开 JMX。

### 兼容性

此配置是根据 Flink 1.0.0 版开发的。

### 设置

#### JOLokia Agent Jar

对于所有单个组件，必须下载一个版本的 jarokia 代理 JAR 文件。测试的版本为["JOLokia 代理 1.6.0"](#)。

以下说明假定已下载的 JAR 文件（jolokia-jvm-1.6.0-agent.jar）位于位置 "/opt/Flink/lib/" 下。

#### JobManager

要将 JobManager 配置为公开此 Jobokia API，您可以在节点上设置以下环境变量，然后重新启动 JobManager：

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

您可以为朱洛基亚选择其他端口（8778）。如果您有一个内部 IP 来锁定 Jallokia，则可以将 "catch all" 0.0.0.0 替换为您自己的 IP。请注意，此 IP 需要可从电报插件访问。

#### TaskManager

要配置 TaskManager 以公开此 JOLokia API，您可以在节点上设置以下环境变量，然后重新启动 TaskManager：

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

您可以为朱洛基亚选择其他端口（8778）。如果您有一个内部 IP 来锁定 Jallokia，则可以将 "catch all" 0.0.0.0 替换为您自己的 IP。请注意，此 IP 需要可从电报插件访问。

## 对象和计数器

将收集以下对象及其计数器：

对象：	标识符：	属性：	数据点：
Flink 任务管理器	集群命名空间服务器	节点名称任务管理器 ID 节点 IP	网络可用内存段网络总内存段垃圾收集 PS 标记扫描计数垃圾收集 PS 标记扫描时间垃圾收集 PS Scavenge 计数垃圾收集 PS Scavenge 时间堆内存已提交堆内存堆内存堆内存最大堆内存已使用的线程数守护进程线程数峰值线程数线程数计数 已启动总数
Flink 作业	集群命名空间服务器作业 ID	节点名称作业名称节点 IP 最后检查点外部路径重新启动时间	停机完全重新启动上次检查点对齐缓冲上次检查点持续时间上次检查点大小已完成检查点数量失败检查点数量进行中的检查点数量检查点正常运行时间数量
Flink 作业管理器	集群命名空间服务器	节点名称节点 IP	垃圾收集 PS 标记扫描计数垃圾收集 PS 标记扫描时间垃圾收集 PS Scavenge 计数垃圾收集 PS Scavenge 时间堆内存已提交堆内存 Init 堆内存最大堆内存已用数量已注册任务管理器数量正在运行作业任务插槽可用任务插槽总数线程计数守护进程线程计数 已启动的线程计数峰值总线程数

对象：	标识符：	属性：	数据点：
Flink 任务	集群命名空间作业 ID 任务 ID	服务器节点名称作业名称子任务索引任务尝试 ID 任务尝试次数任务名称任务管理器 ID 节点 IP 当前输入水印	队列长度缓冲区中的池使用量缓冲区输出池使用量缓冲区输出本地每秒数量缓冲区中的队列长度数字缓冲区计数本地每秒速率缓冲区中的本地每秒速率缓冲区数量缓冲区远程数量缓冲区中的远程数量缓冲区每秒计数远程数量缓冲区数量缓冲区 第二速率数字缓冲区输出数量缓冲区每秒输出数量缓冲区每秒输出速率字节数以本机数字字节数以本机每秒为单位的字节数以本机每秒为单位的字节数远程数字字节数远程每秒字节数字节数远程中的字节数 每秒速率字节数字节数每秒字节数计数字字节数每秒字节数速率中的字节数在数量记录中的每秒计数数量记录在每秒速率中的记录数量记录输出每秒的记录数量在数量中的记录每秒计数的字节数在每秒的速率中的记录数量

对象：	标识符：	属性：	数据点：
Flink 任务操作员	集群命名空间作业 ID 操作员 ID 任务 ID	服务器节点名称作业名称操作员名称子任务索引任务尝试 ID 任务尝试次数任务名称任务管理器 ID 节点 IP	当前输入水印当前输出水印数量记录在数量记录中每秒记录数记录每秒数量记录每秒比率数记录输出数量记录每秒注销数量记录每秒注销数量记录每秒比率数延迟记录已丢弃已分配分区字节数已消耗比率提交延迟平均提交延迟最大提交速率提交失败提交成功连接关闭速率连接计数连接创建速率计数提取延迟平均提取延迟最大提取速率提取大小平均提取大小最大提取节流时间平均提取节流时间最大检测信号速率传入字节速率 IO 比率 IO 时间平均 IO 时间 ( ns ) IO 等待比率 IO 等待时间平均 ( ns ) 加入速率加入时间平均上次检测信号前网络 IO 速率传出字节速率记录已用速率记录滞后每个请求的最大记录平均请求速率请求大小请求大小最大响应速率选择速率同步速率同步时间平均检测信号响应时间最长加入时间最长同步时间最长

## 故障排除

有关其他信息、请参见["支持"](#)页面。

## Hadoop Data Collector

Data Infrastructure Insight 使用此数据收集器从 Hadoop 收集指标。

## 安装

- 从 \*Observability > Collectors\* 中，单击 \*+Data Collector\*。选择 Hadoop。

选择安装了 Telegraf 代理的操作系统或平台。

- 如果尚未安装用于收集的代理，或者要为其他操作系统或平台安装代理，请单击 Show Instructions 展开说明。["代理安装"](#)
- 选择要用于此数据收集器的代理访问密钥。您可以通过单击 \* + 代理访问密钥 \* 按钮来添加新的代理访问密钥。最佳实践：仅当您要按操作系统 / 平台对数据收集器进行分组时，才使用其他代理访问密钥。

4. 按照配置步骤配置数据收集器。这些说明因用于收集数据的操作系统或平台的类型而异。

## [Hadoop 配置] [Hadoop 配置]

### 设置

完整的 Hadoop 部署包括以下组件：

- NameNode：Hadoop 分布式文件系统（HDFS）主系统。协调一系列 DataNode。
- Secondary NameNode：主 NameNode 的热故障转移。在 Hadoop 中，不会自动升级到 NameNode。二级 NameNode 从 NameNode 收集信息，以便在需要时可以进行升级。
- DataNode：数据的实际所有者。
- ResourceManager：计算主系统（Yarn）。协调一系列 NodeManager。
- NodeManager：计算资源。运行应用程序的实际位置。
- JobHistoryServer：负责处理所有与作业历史记录相关的请求。

Hadoop 插件基于此电报的 JOLokia 插件。例如，需要从所有 Hadoop 组件收集信息，因此需要在所有组件上通过 Jallokia 配置并公开 JMX。

### 兼容性

此配置是根据 Hadoop 2.9.2 版开发的。

### 设置

#### JOLokia Agent Jar

对于所有单个组件，必须下载一个版本的 jarokia 代理 JAR 文件。测试的版本为 "[JOLokia 代理 1.6.0](#)"。

以下说明假定已下载的 JAR 文件（jolokia-jvm-1.6.0-agent.jar）位于位置 "/op/hadoop/lib/" 下。

#### NameNode

要配置 NameNode 以公开此 JOLokia API，您可以在 <Hadoot\_home>/etc/Hadoop/Hadoop-env.sh 中设置以下内容：

```
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.

## 二级 NameNode

要配置二级 NameNode 以公开此 JOLokia API，您可以在 <Hadoot\_home>/etc/Hadoop/Hadoop-env.sh 中设置以下内容：

```
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.

## DataNode

要将 DataNode 配置为公开此 Jolokia API，您可以在 <Hadoot\_home>/etc/Hadoop/Hadoop-env.sh 中设置以下内容：

```
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.

## ResourceManager

要将 ResourceManager 配置为公开此 Jolokia API，您可以在 <Hadoot\_home>/etc/Hadoop/Hadoop-env.sh 中设置以下内容：

```
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.

## NodeManager

要配置 NodeManager 以公开此 JOLOkia API，您可以在 <Hadoot\_home>/etc/Hadoop/Hadoop-env.sh 中设置以下内容：

```

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.password"

```

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

要将 JobHistoryServer 配置为公开 Joyokia API , 您可以在 <Hadoot\_home>/etc/Hadoop/Hadoop-env.sh 中设置以下内容：

```

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.password"

```

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.

## 对象和计数器

将收集以下对象及其计数器：

对象：	标识符：	属性：
Hadoop 二级 NameNode	集群命名空间服务器	节点名称节点 IP 编译信息版本
Hadoop NodeManager	集群命名空间服务器	节点名称节点 IP
Hadoop ResourceManager	集群命名空间服务器	节点名称节点 IP
Hadoop DataNode	集群命名空间服务器	节点名称节点 IP 集群 ID 版本

对象：	标识符：	属性：
Hadoop NameNode	集群命名空间服务器	节点名称节点 IP 事务 ID 上次加载后的最后写入时间编辑 HA 状态文件系统状态块池 ID 集群 ID 编译信息不同版本计数版本
Hadoop JobHistoryServer	集群命名空间服务器	节点名称节点 IP

## 故障排除

有关其他信息、请参见["支持"](#)页面。

## HAProxy 数据收集器

Data Infrastructure Insight 使用此数据收集器从 HAProxy 收集指标。

### 安装

1. 从 \*Observability > Collectors\* 中，单击 \*+Data Collector\*。选择 HAProxy。

选择安装了 Telegraf 代理的操作系统或平台。

2. 如果尚未安装用于收集的代理，或者要为其他操作系统或平台安装代理，请单击 Show Instructions 展开说明。["代理安装"](#)
3. 选择要用于此数据收集器的代理访问密钥。您可以通过单击 \* + 代理访问密钥 \* 按钮来添加新的代理访问密钥。最佳实践：仅当您要按操作系统 / 平台对数据收集器进行分组时，才使用其他代理访问密钥。
4. 按照配置步骤配置数据收集器。这些说明因用于收集数据的操作系统或平台的类型而异。

### [HAProxy 配置]

### 设置

Telegraf 的 HAProxy 插件依赖于启用 HAProxy 统计信息。这是一种内置在 HAProxy 中的配置，但它不会开箱即用。启用后，HAProxy 将显示一个 HTML 端点，您可以在浏览器上查看此端点或对其进行擦除，以便提取所有 HAProxy 配置的状态。

### 兼容性：

配置是根据 HAProxy 1.0.4 版开发的。

### 设置：

要启用统计信息，请编辑您的 `haproxy` 配置文件，并使用您自己的用户 / 密码和 / 或 `haproxy` URL 在 " 存储 " 部分后面添加以下行：

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

下面是一个简化的配置文件示例，该配置文件已启用统计信息：

```
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
```

有关完整的最新说明，请参阅["HAProxy 文档"](#)。

## 对象和计数器

将收集以下对象及其计数器：

对象：	标识符：	属性：	数据点：
HAProxy 前端	命名空间地址代理	节点 IP 节点名称代理 ID 模式进程 ID 会话速率限制 服务器 ID 会话限制状态	字节（以字节为单位）传出缓存命中缓存查找压缩字节绕过压缩字节输出压缩响应连接速率连接速率最大连接被连接拒绝的请求总数安全问题拒绝的连接规则请求有关安全问题拒绝的响应有关会话规则请求错误响应 1xx 拒绝的请求 响应 2xx 响应 3xx 响应 4xx 响应 5xx 响应其他请求截获的会话速率会话速率最大请求速率最大请求总会话会话最大会话请求重写总数
HAProxy 服务器	命名空间地址代理服务器	节点 IP 节点名称检查完成时间检查秋季配置检查运行状况值检查上升配置检查状态代理 ID 上次更改时间上次会话时间模式进程 ID 服务器 ID 状态权重	Active Server 备份服务器字节（以字节为单位）输出检查下载检查失败客户端中止连接连接平均时间停机总拒绝响应连接错误响应响应错误响应 1xx 响应 2xx 响应 3xx 响应 5xx 响应其他服务器选择的总队列当前队列最大队列平均时间会话数 每秒第二个会话最大连接重用响应时间平均会话会话最大服务器传输中止会话总会话总时间平均请求重新派单请求重写请求
HAProxy 后端	命名空间地址代理	节点 IP 节点名称代理 ID 上次更改时间上次会话时间模式进程 ID 服务器 ID 会话限制状态权重	Active Server 备份服务器字节（以字节为单位）传出缓存命中缓存查找检查下载客户端中止压缩字节绕过压缩字节（以压缩字节为单位）输出压缩响应连接平均时间停机安全问题拒绝的总请求响应安全问题连接错误响应错误响应 1xx 响应 2xx 响应 3xx 响应 4xx 响应 5xx 响应其他服务器选择的总队列当前队列最大队列平均每秒会话时间最大请求总数连接重复使用响应时间平均会话会话最大服务器传输中止会话总数会话总时间平均请求重新发送请求请求请求重试请求 重写

## 故障排除

有关其他信息、请参见["支持"](#)页面。

## JVM 数据收集器

Data Infrastructure Insight 使用此数据收集器从JVM收集指标。

### 安装

1. 从\*Observability > Collectors\*中，单击\*+Data Collector\*。选择JVM。

选择安装了 Telegraf 代理的操作系统或平台。

2. 如果尚未安装用于收集的代理，或者要为其他操作系统或平台安装代理，请单击 Show Instructions 展开说明。["代理安装"](#)
3. 选择要用于此数据收集器的代理访问密钥。您可以通过单击 \* + 代理访问密钥 \* 按钮来添加新的代理访问密钥。最佳实践：仅当您要按操作系统 / 平台对数据收集器进行分组时，才使用其他代理访问密钥。
4. 按照配置步骤配置数据收集器。这些说明因用于收集数据的操作系统或平台的类型而异。

### [JVM配置]

### 设置

有关信息，请参见["JVM文档"](#)。

### 对象和计数器

将收集以下对象及其计数器：

对象：	标识符：	属性：	数据点：
JVM	命名空间JVM	操作系统架构 操作系统名称 操作系统版本 运行时规范 运行时规范供应商 运行时规范版本 正常运行时 VM 名称 运行时 VM 供应商 运行时 VM 版本 节点名称 节点 IP	类已加载 类已卸载 总类已提交 内存堆已使用 最大内存堆已使用 内存非堆已提交 内存非堆初始化 最大内存未堆已使用 对象待定最终确定 操作系统处理器可用 操作系统已提交 虚拟内存大小 操作系统可用物理内存大小 操作系统可用交换空间大小 操作系统最大文件描述符计数 操作系统打开文件描述符计数 操作系统操作符计数 操作系统处理器 CPU 负载 操作系统处理器 CPU 时间 操作系统 CPU 负载 操作系统系统负载 平均操作系统总物理内存大小 操作系统总交换空间大小 线程守护进程 计数线程峰值计数 线程计数 线程已开始 总数垃圾 收集器副本 收集计数 垃圾收集器 副本收集时间 垃圾收集器 标记 - 扫描 收集计数 垃圾收集器 标记 - 扫描 收集时间 垃圾收集器 G1 旧一代 收集时间 垃圾收集器 G1 年轻一代 收集计数 垃圾收集器 G1 年轻一代 收集时间 垃圾收集器并发 标记 - 扫描 收集计数 垃圾收集器并发 标记 - 扫描 收集时间 垃圾收集器并行 收集计数 垃圾收集器并行 收集时间 垃圾收集器并行 收集计数 垃圾收集器并行 收集标记 - 扫描 收集计数 垃圾收集器并行 收集标记 - 扫描 收集时间 垃圾收集器并行 收集收集器 收集计数 垃圾收集器并行 收集时间

故障排除

有关其他信息、请参见“[支持](#)”页面。

# Kafka 数据收集器

Data Infrastructure Insight 使用此数据收集器从 Kafka 收集指标。

## 安装

- 从“Observability > Collectors”中，单击“+Data Collector”。选择Kafka。

选择安装了 Telegraf 代理的操作系统或平台。

- 如果尚未安装用于收集的代理，或者要为其他操作系统或平台安装代理，请单击 Show Instructions 展开说明。[“代理安装”](#)
- 选择要用于此数据收集器的代理访问密钥。您可以通过单击 “+ 代理访问密钥” 按钮来添加新的代理访问密钥。最佳实践：仅当您要按操作系统 / 平台对数据收集器进行分组时，才使用其他代理访问密钥。
- 按照配置步骤配置数据收集器。这些说明因用于收集数据的操作系统或平台的类型而异。

### [Kafka 配置]

## 设置

Kafka 插件基于此电报的 JOLokia 插件。例如，需要从所有 Kafka 代理收集信息，因此需要在所有组件上通过 Jallokia 配置和公开 JMX。

### 兼容性

此配置是根据 Kafka 0.11.0.2 版开发的。

### 设置

以下所有说明均假定您的 Kafka 安装位置为 "/opt/Kafka"。您可以根据安装位置调整以下说明。

#### JOLokia Agent Jar

Jolokia代理jar文件的版本必须为[“已下载”](#)。测试的版本为 JOLokia 代理 1.6.0。

以下说明假定下载的 JAR 文件（jolokia-jvm-1.6.0-agent.jar）位于 "/opt/Kafka/libs/" 位置下。

#### Kafka 代理

要配置 Kafka 代理以公开此 JOLokia API，您可以在调用 "Kafka-run-class.sh" 之前的 <Kafka\_home>/bin/Kafka-server-start.sh 中添加以下内容：

```
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.p
assword -Dcom.sun.management.jmxremote.ssl=false
-Djava.rmi.server.hostname=$RMI_HOSTNAME
-Dcom.sun.management.jmxremote.rmi.port=$JMX_PORT"
```

请注意，上述示例使用 "hostname -i" 设置 "RMI\_HOSTNAME" 环境变量。在多个 IP 计算机中，需要对其进行

调整，以收集您关心的用于 RMI 连接的 IP。

您可以为 JMX（上面为 9999）和朱洛基亚（8778）选择其他端口。如果您有一个内部 IP 来锁定 Jallokia，则可以将 "catch all" 0.0.0.0 替换为您自己的 IP。请注意，此 IP 需要可从电报插件访问。如果您不想进行身份验证，可以使用选项 -Dcom.sun.management.jmxremote.authenticate=false'。使用风险自行承担。

## 对象和计数器

将收集以下对象及其计数器：

对象：	标识符：	属性：
Kafka Broker	集群命名空间代理	节点名称节点 IP

## 故障排除

有关其他信息、请参见["支持"](#)页面。

# Kibana 数据收集器

Data Infrastructure Insight 使用此数据收集器从基巴纳收集指标。

## 安装

1. 从\*Observability > Collectors\*中，单击\*+Data Collector\*。选择“Kibana”。

选择安装了 Telegraf 代理的操作系统或平台。

2. 如果尚未安装用于收集的代理，或者要为其他操作系统或平台安装代理，请单击 Show Instructions 展开说明。["代理安装"](#)
3. 选择要用于此数据收集器的代理访问密钥。您可以通过单击 \* + 代理访问密钥 \* 按钮来添加新的代理访问密钥。最佳实践：仅当您要按操作系统 / 平台对数据收集器进行分组时，才使用其他代理访问密钥。
4. 按照配置步骤配置数据收集器。这些说明因用于收集数据的操作系统或平台的类型而异。

## [Kibana 配置]

## 设置

有关信息，请参见["Kibana 文档"](#)。

## 对象和计数器

将收集以下对象及其计数器：

对象：	标识符：	属性：	数据点：
Kibana	命名空间地址	节点 IP 节点名称版本状态	并发连接堆最大堆已用请求 / 秒响应时间平均响应时间最长正常运行时间

## 故障排除

有关其他信息、请参见["支持"](#)页面。

# Kubernetes Monitoring Operator安装和配置

Data Infrastructure Insight为Kubernetes集合提供了\* Kubernetes Monitoring Operator\*。导航到\*Kubernetes >收集器>+Kubernetes Collector\*以部署新操作员。

## 在安装Kubernetes Monitoring Operator之前

在安装或升级Kubornetes Monitoring Operator之前、请参见相关["前提条件"](#)文档。

## 安装Kubernetes Monitoring Operator

The screenshot shows the 'Deploy NetApp Monitoring Operator' page. At the top, there's a 'kubernetes' icon and the word 'Kubernetes'. Below that, the title 'Deploy NetApp Monitoring Operator' and a subtitle 'Quickly install and configure a Kubernetes Operator to send cluster information to Cloud Insights.' To the right of the title is a 'Need Help?' link.

In the center, there's a section titled 'Select existing API Access Token or create a new one'. It contains a dropdown menu with the value 'KEY2024 (...vw6NdM)' and a button '+ API Access Token'. To the right of the dropdown is a 'Production Best Practices' link with a question mark icon.

Below this is an 'Installation Instructions' section. It contains a note: 'Please review the [pre-requisites](#) for installing the NetApp Kubernetes Monitoring Operator. To update an existing operator installation please follow [these steps](#).'. To the right is another 'Need Help?' link.

Step 1: 'Define Kubernetes cluster name and namespace'. It says 'Provide the Kubernetes cluster name and specify a namespace for deploying the monitoring components.' There are two input fields: 'Cluster' (containing 'clustername') and 'Namespace' (containing 'netapp-monitoring').

Step 2: 'Download the operator YAML files'. It says 'Execute the following download command in a *bash* prompt.' Below this are two buttons: 'Copy Download Command Snippet' and 'Reveal Download Command Snippet'.

A note at the bottom states: '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KubeNet上安装Kubernetes Monitoring Operator代理的步骤：

1. 输入唯一的集群名称和命名空间。如果您[正在升级](#)来自先前的Kubernetes Operator、请使用相同的集群名称和命名空间。
2. 输入这些代码后、您可以将Download Command代码录复制到剪贴板。
3. 将此代码片段粘贴到 `bash` 窗口中并执行。此时将下载Operator安装文件。请注意、此代码片段具有唯一的密钥、有效期为24小时。
4. 如果您有自定义或私有存储库、请复制可选的映像提取代码段、将其粘贴到`_bash_ shell`中并执行该代码段。提取映像后、将其复制到您的私有存储库。请务必保持相同的标记和文件夹结构。更新`_operator-DEPRAYAML_`中的路径以及`_operator-config.yaml_`中的Docker存储库设置。
5. 如果需要、请查看可用的配置选项、例如代理或专用存储库设置。您可以阅读有关的更多信息["配置选项"](#)。
6. 准备好后、请通过复制kubec临时 应用的小程序来部署Operator、然后下载并执行该操作。
7. 安装将自动进行。完成后、单击`_Next_`按钮。
8. 安装完成后、单击`_Next_`按钮。同时、请务必删除或安全地存储`_operator-秘密.yaml`文件。

如果您有自定义存储库、请阅读有关[使用自定义/私有Docker存储库](#)。

## Kubernetes监控组件

Data Infrastructure Insight Kubernetes监控由四个监控组件组成：

- 集群指标
- 网络性能和映射(可选)
- 事件日志(可选)
- 变更分析(可选)

默认情况下、每个Kubernetes收集器都会启用上述可选组件；如果您确定某个特定收集器不需要某个组件、则可以通过导航到\* Kubernetes > Collectors \*并从屏幕右侧收集器的“三个点”菜单中选择\_Modify Deployment\_来禁用此组件。

NetApp / Observability / Collectors

Data Collectors (0 2)   Acquisition Units (0 4)   Kubernetes Collectors

Kubernetes Collectors (13)   View Upgrade/Delete Documentation   + Kubernetes Collector   Filter...

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	1.101.0	⋮

此屏幕将显示每个组件的当前状态，并允许您根据需要为该收集器禁用或启用组件。

**kubernetes**    Kubernetes

### Modify Deployment

**Cluster Information**

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

**Deployment Options**

Network Performance and Map  
 Event Logs  
 Change Analysis

Need Help?    [Cancel](#)    [Complete Modification](#)

升级到最新的Kubernetes Monitoring Operator

## DII按钮升级

您可以通过DII Kubnetes收集器页面升级Kubnetes Monitoring Operator。单击要升级的集群旁边的菜单、然后选择\_Upgrade\_。操作员将验证映像签名、为当前安装创建快照并执行升级。在几分钟内、您将看到操作员状态进度、直到Upgrade in Progress to latest。如果遇到错误、您可以选择错误状态以了解更多详细信息、并参阅下面的按钮升级故障排除表。

### 使用私有存储库进行按钮式升级

如果您的操作员已配置为使用专用存储库、请确保运行操作员所需的所有映像及其签名均可在您的存储库中使用。如果在升级过程中遇到缺少映像的错误、只需将其添加到存储库中、然后重试升级即可。要将图像签名上传到您的存储库、请按如下所示使用联合签名工具、确保为3可选：将操作员图像上传到您的私有存储库>图像提取片段中指定的所有图像上传签名

```
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>
```

### 正在回滚到先前运行的版本

如果您使用按钮升级功能进行了升级、并且在升级后七天内遇到与当前操作员版本相关的任何问题、则可以使用升级过程中创建的快照降级到先前运行的版本。单击要回滚的集群旁边的菜单、然后选择\_Rollback\_。

## 手动升级

确定现有Operator是否存在AgentConfiguration (如果您的命名空间不是默认的\_NetApp-monitoring \_、请替换相应的命名空间):

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

- 安装现有运算符上的最新运算符。
  - 如果您使用的是自定义存储库、请确保您[提取最新的容器映像](#)使用的是。

如果AgentConfiguration不存在：

- 记下数据基础架构洞察力可识别的集群名称(如果您的命名空间不是默认的NetApp监控、请替换相应的命名空间):

```
kubectl -n netapp-monitoring get agent -o
jsonpath='{.items[0].spec.cluster-name}'
* 为现有Operator创建备份(如果您的命名空间不是默认的NetApp
监控、请替换相应的命名空间):
```

```
kubectl -n netapp-monitoring get agent -o yaml > agent_backup.yaml
* <<to-remove-the-kubernetes-monitoring-operator, 卸载>>现有操作员。
* <<installing-the-kubernetes-monitoring-operator, 安装>>最新运算符。
```

- 请使用相同的集群名称。
- 下载最新的Operator YAML文件后、在部署之前、将在agent\_backup.yaml中找到的所有自定义设置移植到下载的operator-config.yaml。
- 如果您使用的是自定义存储库、请确保您[提取最新的容器映像](#)使用的是。

## 停止和启动Kubnetes Monitoring Operator

要停止Kubnetes Monitoring Operator：

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

要启动Kubnetes Monitoring Operator：

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

## 正在卸载

### 删除Kubnetes Monitoring Operator

请注意、Kubernetes Monitoring Operator的默认命名空间为"netp-monitoring"。如果您已设置自己的命名空间，[请在这些命令和所有后续命令和文件中替换该命名空间](#)。

可以使用以下命令卸载较新版本的监控操作员：

```
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>
```

如果监控操作员部署在自己的专用命名空间中、请删除此命名空间：

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

**注：如果第一个命令返回“No Resources Found”(未找到资源)**，[请按照以下说明卸载较旧版本的监控操作员](#)。

按顺序执行以下每个命令。根据您当前的安装、其中某些命令可能会返回"object not found (未找到对象)"消息。可以安全地忽略这些消息。

```
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>
```

如果以前创建了安全上下文约束：

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

## 关于Kube-state-metrics

NetApp Kubernetes监控操作员会安装自己的Kube-state-metrics，以避免与任何其他实例发生冲突。

有关Kube-State-Metrics的信息，请参见[此页面](#)。

## 配置/自定义操作员

这些部分包含有关自定义操作员配置、使用代理、使用自定义或私有Docker存储库或使用OpenShift的信息。

### 配置选项

最常修改的设置可以在`_AgentConfiguration_`自定义资源中进行配置。您可以通过编辑`_operator-config.yaml`文件来在部署操作员之前编辑此资源。此文件包含注释掉的设置示例。有关操作符的最新版本，请参见列表[可用设置](#)。

您也可以在部署操作员后使用以下命令编辑此资源：

```
kubectl -n netapp-monitoring edit AgentConfiguration
要确定您部署的操作员版本是否支持AgentConfiguration，请运行以下命令：
```

```
kubectl get crd agentconfigurations.monitoring.netapp.com
如果您看到“Error from server (NotFound)”
消息，则必须先升级操作员，然后才能使用AgentConfiguration。
```

## 配置代理支持

您可以在两个位置使用租户上的代理来安装Kubnetes Monitoring Operator。这些代理系统可以是相同的、也可以是单独的：

- 在执行安装代码段(使用"curl")期间需要代理、以便将执行此代码段的系统连接到Data Infrastructure Insight环境
- 目标Kubnetes集群与Data Infrastructure Insight环境通信所需的代理

如果您对其中一个或这两个环境使用代理、则要安装Kubornetes Operating Monitor、必须首先确保您的代理已配置为能够与Data Infrastructure Insight环境进行良好的通信。如果您有一个代理、并且可以从要安装Operator的服务器/VM访问Data Infrastructure Insight、则您的代理可能已正确配置。

对于用于安装Kubersnetes Operating Monitor的代理、在安装Operator之前、请设置`_http_proxy/https_proxy_Environment`变量。对于某些代理环境、您可能还需要设置`_no_proxy environment_`变量。

要设置变量，请在\*安装Kubernetes Monitoring Operator之前\*在系统上执行以下步骤：

1. 为当前用户设置`https_proxy`和 / 或`http_proxy`环境变量：

- a. 如果要设置的代理没有身份验证(用户名/密码)、请运行以下命令：

```
export https_proxy=<proxy_server>:<proxy_port>
.. 如果要设置的代理具有身份验证(用户名/密码)、请运行以下命令：
```

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

要使Kubennetes集群所使用的代理与Data Infrastructure Insight环境进行通信、请在阅读所有这些说明后安装Kubennetes Monitoring Operator。

在部署Kubernetes Monitoring Operator之前、请在operator-config.yaml中配置AgentConfiguration的代理部分。

```

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
  ...
  ...

```

## 使用自定义或专用Docker存储库

默认情况下、Kubornetes监控操作员将从Data Infrastructure Insight存储库中提取容器映像。如果您将某一个Kubornetes集群用作监控目标、并且该集群配置为仅从自定义或私有Docker存储库或容器注册表中提取容器映像、则必须配置对Kubornetes监控操作员所需容器的访问权限。

从NetApp Monitoring Operator安装磁贴运行"Image Pull Snippet"。此命令将登录到Data Infrastructure Insight存储库、提取操作员的所有映像依赖关系、然后从Data Infrastructure Insight存储库中注销。出现提示时、输入提供的存储库临时密码。此命令可下载操作员使用的所有映像、包括可选功能的映像。请参见以下内容、了解这些图像用于哪些功能。

## 核心操作员功能和Kubornetes监控

- NetApp监控
- CI-KKube-RBAC-代理
- CI-KSM
- CI-(国际通信)
- distroless root用户

## 事件日志

- CI-流畅位
- CI-Kuber-netes-event-exporter

## 网络性能和映射

- CI-net-observer

根据您的企业策略，将操作员 Docker 映像推送到您的私有 / 本地 / 企业 Docker 存储库。确保存储库中这些映像的映像标记和目录路径与 Data Infrastructure Insight 存储库中的映像标记和目录路径一致。

在operator-DEPLOYAML中编辑monitor-operator部署、并修改所有映像引用以使用私有Docker存储库。

```
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>
```

编辑operator-config.yaml中的AgentConfiguration以反映新的Docker repo位置。为私有存储库创建新的imagePullSecret,有关更多详细信息，请参见<https://kubernetes.io/docs/tasks/configure-pod-container/pull-image-private-registry/>

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

## OpenShift 说明

如果您运行的是OpenShift 4.6或更高版本，则必须在\_operator-config.yaml中编辑AgentConfiguration以启用\_run特权\_设置：

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

OpenShift可以实施更高的安全级别、从而可能阻止对某些Kubernetes组件的访问。

## 容差和污物

*netapp-CI-tentlaf-ds\_*、*netapp-CI-fluent-bit-ds* 和 *netapp-CI-net-oboder-L4-DS* DemonSets 必须在集群中的每个节点上计划一个POD、以便正确收集所有节点上的数据。操作器已配置为允许某些众所周知的\*污染\*。如果您在节点上配置了任何自定义污染，从而阻止Pod在每个节点上运行，则可以为这些污染创建\*toleration\*“在\_AgentConfiguration 中”。如果已将自定义污染应用于集群中的所有节点，则还必须向操作员部署添加必要的容错值、以便可以计划和执行操作员POD。

了解有关Kubernetes的更多信息["损害和公差"](#)。

返回到["NetApp Kubernetes监控操作员安装"页面](#)

## 关于安全的注意事项

要删除Kubernetes Monitoring Operator在集群范围内查看机密的权限、请在安装之前从`_operator-setup.yaml`文件中删除以下资源：

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

如果是升级、请同时从集群中删除资源：

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

如果启用了"变更分析"、请修改`_AgentConfiguration_或_operator-config.yaml`以取消注释`change-management`部分、并在`change-management`部分下包括`_kindsToIgnoreFromWatch: "secrets"__`。记下此行中单引号和双引号的存在和位置。

```
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"'  
  ...
```

## 验证Kubernetes监控操作员图像签名

操作员的映像及其部署的所有相关映像均由NetApp签名。您可以在安装之前使用联合签名工具手动验证映像、也可以配置Kubernetes接入控制器。有关详细信息，请参见["Kubernetes 文档"](#)。

用于验证图像签名的公共密钥可在"Monitoring Operator"安装磁贴中的`_可选：将操作员图像上传到您的私有存储库>图像签名公共密钥_`下找到

要手动验证映像签名、请执行以下步骤：

1. 复制并运行映像提取片段
2. 根据提示复制并输入存储库密码

- ### 3. 存储图像签名公共密钥(示例中为dll-image-signing.pub)

4. 使用联合签名验证图像。请参见以下联合签名用法示例

```
$ 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

在设置Kubernetes Monitoring Operator时遇到问题时、请尝试以下操作：

<p>问题：</p> <p>我看到来自Telegraf的错误消息如下所示、但Telegraf确实启动并运行：10月11日14: 23: 41 IP-172-31-39-47 systemd[1]: 启动插件驱动的服务器代理、以便向InfluxDB报告指标。10月11日14: 23: 41 IP-172-31-39-47电话[1827]: time="2021-10-11T14: 23: 41Z" level = error msg="failed to create cache directory。/etc/trendelaf/.cache/snowsclap, err: mkdir /etc/trendelaf/.cache: permission denied. ignored\n" func="gosnowscale.(defaultLogg).Errorf"  <b>file="log.go:120" OCT11 14: 23: 41 IP-172-31-39-47 trendelaf[1827]: time="msg-10:23=11Z"</b>打开错误。已忽略。打开/etc/trendelaf/.cache/snowsclap/ocsp_response_cache.json：无此文件或目录\n" func="gosnowsclap. ( defaultLogger).Errorf"  <b>file="log.go:120" OCT11 14: 23: 41 IP-172-31-39-47 trendelaf[1827]: 2021-10-11T14: 23: 41Z</b>启动 Telegraf 1.19.3</p>	<p>请尝试以下操作：</p> <p>这是一个已知的问题描述。<a href="#">"此 GitHub 文章"</a>有关详细信息、请参见。只要 Telegraf 启动并运行，用户就可以忽略这些错误消息。</p>
<p>在 Kubernetes 上，我的 Telegraf Pod 报告以下错误：" 处理 mountstats 信息时出错：无法打开 mountstats 文件：/hostfs/proc/1/mountstats，错误：open /hostfs/proc/1/mountstats：权限被拒绝 "</p>	<p>如果启用并强制实施 SELinux，则可能会阻止 Telegraf Pod 访问 Kubernetes 节点上的 /proc/1/mountstats 文件。要克服此限制，请编辑代理配置并启用 run 特权设置。有关更多详细信息，请参阅 OpenShift 说明。</p>
<p>在 Kubernetes 上，我的 Telegraf ReplicaSet Pod 报告以下错误：inputs.prometheus] 插件错误：无法加载密钥类型 /etc/Kubernetes，PKI/etcd/server.crt：/etc/Kubernetes，crt/etcd/server.key：打开 /etc/Kubernetes，pki/etcd/server.key：open /etc/Kubernetes，pki/etcd/server.key：no 此类文件或目录</p>	<p>Telegraf ReplicaSet Pod 应在指定为主节点或 etcd 节点上运行。如果 ReplicaSet Pod 未在其中一个节点上运行，您将收到这些错误。检查您的主 /etcd 节点是否具有此类节点的影响。如果是，请将必要的容错添加到 Telegraf ReplicaSet，即 Teleaf-RS 中。例如，编辑 ReplicaSet... kubectl edit RS ceaf-rs ... 并将适当的容错添加到规范中。然后，重新启动 ReplicaSet Pod。</p>
<p>我使用的是PSP/PSA环境。这是否会影响我的监控操作员？</p>	<p>如果您的 Kubornetes 集群运行的是 Pod 安全策略 (PSP) 或 Pod 安全准入 (PSA)，则必须升级到最新的 Kubornetes Monitoring Operator。按照以下步骤升级到支持 PSP/PSA 的当前 Operator：</p> <ol style="list-style-type: none"> <li>卸载先前的监控运算符：kubectl delete agent agent-monitoring- ngubect NetApp delete ns NetApp 监控 kubectl delete crd agents.monitoring.kubec.com kubectl delete NetApp NetApp delete-manager-roole agent-proxy-roxy-role-metric-reactor kubectl delete cluster cluster-manager-rolebingagent-proxy-rolebingagent-rolebingagent-roleb 安装最新版本的监控操作符。</li> </ol>
<p>我在尝试部署操作员时遇到问题，并且我正在使用 PSP/PSA。</p>	<p>1. 使用以下命令编辑代理：kubectl -n &lt;name-space&gt; edit agent 2. 将 "securtion-policy-enabled" 标记为 "false"。这将禁用 Pod 安全策略和 Pod 安全准入，并允许操作员进行部署。使用以下命令进行确认：kubectl get PSP (应显示 Pod Security Policy Removed) kubectl get all -n &lt;namespace&gt; grep -i PSP (应显示未找到任何内容)</p>

问题:	请尝试以下操作:
出现"ImagePullBackoff"错误  我正在部署监控操作员问题描述、而当前文档对我的解决没有帮助。	<p>如果您具有自定义或专用Docker存储库、但尚未将Kubernetes Monitoring Operator配置为正确识别它、则可能会出现这些错误。<a href="#">阅读更多内容</a>关于为自定义/专用repo配置。</p> <p>捕获或记下以下命令的输出、然后联系技术支持团队。</p> <pre>kubectl -n netapp-monitoring get all kubectl -n netapp-monitoring describe all kubectl -n netapp-monitoring logs &lt;monitoring-operator-pod&gt; --all -containers=true kubectl -n netapp-monitoring logs &lt;telegraf-pod&gt; --all -containers=true</pre>
Operator命名空间中的Net-Observer (Workload Map) Pod位于CrashLoopBackOff中	<p>这些Pod对应于用于网络可观察性的工作负载映射数据收集器。请尝试以下操作:</p> <ul style="list-style-type: none"> <li>•检查其中一个Pod的日志以确认最低内核版本。例如: --- {"ci租户id": "Your-en租户id"、"cCollector cluster-cluster-": "Your -K8s-cluster-name"、"景 置": "prod"、"level": "error"、"msg": "验证失败。原因: 内核版本3.10.0低于最低内核版本4.18.0"、"time": "2022-11-09T08: 23: 08Z"}</li> <li>•Net-observer Pod要求Linux内核版本至少为4.18.0。使用命令"uname -r"检查内核版本、并确保它们&gt;= 4.18.0</li> </ul>
Pod正在Operator命名空间中运行(默认值: netapp-monitoring)、但在查询中、UI中不会显示工作负载映射或KubeNet指标的任何数据	检查K8S集群节点上的时间设置。为了准确地进行审核和数据报告、强烈建议使用网络时间协议(NTP)或简单网络时间协议(SNTP)同步Agent计算机上的时间。
Operator命名空间中的某些Net-observer Pod处于Pending状态	Net-observer是一个DaemonSet、在K8s集群的每个节点上运行一个POD。•记下处于“待定”状态的POD，并检查它是否遇到了CPU或内存的资源问题描述。确保节点中具有所需的内存和CPU。
安装Kubernetes监控操作员后、我的日志inputs.prometheus]中立即显示以下内容: [HTTP错误插件: 向\TCP.SVC.CLUSTER-LOCAL:8080/METRICS发出HTTP://KUBE-STATE-METRICS <NAMESPACE>请求时出错: GET \TCP.SVC.CLUSTER-LOCAL:8080/METRICS <a href="http://KUBE-STATE-METRICS">HTTP://KUBE-STATE-METRICS</a> : 拨打<NAMESPACE><NAMESPACE>: LOOKUP Kube-state-metrics.tcp.svc.cluster-local: NO SUCH HOST 主机	通常、只有在安装了新操作员且_craaf-RS_ POD 在_KSM_ POD启动之前启动时、才会显示此消息。所有Pod运行后、这些消息应停止。

问题：	请尝试以下操作：
我没有看到为集群中的Kubernetes CronJobs收集任何指标。	验证您的Kubernetes版本(即 kubectl version)。如果是v1.20.x或更低版本、则这是预期的限制。随Kubernetes Monitoring Operator部署的Kube-state-metrics版本仅支持v1.cronjob.对于Kubernetes 1.2.x及更低版本、cronJob资源位于v1beta.cronJob。因此、Kube-state-metrics找不到cronJob资源。
安装操作员后、该特拉夫DS Pod进入CrashLoopBackOff、并且POD日志指示"su : authentication failure"(su：身份验证失败)。	编辑_AgentConfiguration_中的"特拉夫"部分、并将_dockerMetricCollectionEnabled"设置为false。有关详细信息，请参阅操作员的 <a href="#">"配置选项"</a> 。规范：.. -名称： Docker run-mode： - DemonSet 替换 项： -关键字： Docker _UNIS_sdoc_s占位符 值： UNIX： //run/Docker。sk.....
我在Telegraf日志中看到重复出现以下错误消息：E! [agent]写入至Outputs.http: POST "\https : //lace/rest/v1/lace/ingest/影响xdb"时出错：超过上下文截止时间(<tenant_url>。等待标头时超时)	编辑_AgentConfiguration_中的"特拉夫"部分、并将_outputTimeout_增加到10秒。有关详细信息，请参阅操作员的 <a href="#">"配置选项"</a> 。
我缺少一些事件日志的_volvedobject_数据。	确保已按照上述部分中的步骤进行操作 <a href="#">"权限"</a> 。
为什么我看到两个监控操作员Pod正在运行、一个名为NetApp-CI-monitoring operator-Pod <pod>、另一个名为monitoring operator-Pod? <pod>	自2023年10月12日起、Data Infrastructure Insight对运营者进行了重构、以更好地为用户服务；要完全采用这些变更 <a href="#">删除旧运算符</a> 、您必须和 <a href="#">安装新的</a> 。
我的Kubernetes事件意外停止向Data Infrastructure Insight报告。	检索事件导出器Pod的名称：  <pre>`kubectl -n netapp-monitoring get pods`</pre>
grep event-exporter	awk '{print \$1}'
sed 's/event-exporter./event-exporter/' 此名称应为"NetApp-CI-event-exporter "或"event-exporter。接下来，编辑监控代理 kubectl -n netapp-monitoring edit agent，并设置log_file的值，以反映在上一步中找到的相应事件导出器POD名称。更具体地说、log_file应设置为"/var/log/containers/NetApp-CI-event-exporter.log"或"/var/log/containers/event-exporter 。log"  .... <b>fluent-bit:</b> ... <b>- name: event-exporter-ci</b> <b>substitutions:</b> <b>- key: LOG_FILE</b> <b>values:</b> <b>- /var/log/containers/netapp-ci-event-exporter.log</b> ... .... 或者、也可以 <a href="#">卸载 重新安装</a> 选择代理。	我发现Kubeneses监控操作员部署的POD因资源不足而崩溃。

问题：	请尝试以下操作：
有关根据需要增加CPU和/或内存限制的信息、请参见Kubornetes Monitoring Operator"配置选项"。	缺少映像或配置无效会导致NetApp-CI-Kube-state-metrics Pod无法启动或准备就绪。现在、StatefulSet停止运行、并且配置更改未应用于NetApp-CI-KUE-STATE-MERics Pod。
"状态集"处于"已损坏"状态。修复任何配置问题后、退回NetApp-CI-Kube-state-metrics Pod。	运行Kubelnetes Operator升级后、netapo-CI-Kube-state-metrics Pod无法启动、引发ErrImagePull (无法提取映像)。
尝试手动重置Pod。	在日志分析下、我的Kubernetes集群显示"Event Discarded as older than maxEventAgeSeconds"消息。
修改Operator <i>agentconfiguration</i> _并将_ <i>event-exporter maxEventAgeSeconds</i> (例如、60秒)、 <i>event-exporter kubeQPS</i> (例如、100)和_ <i>event-exporter kubeBurst</i> _(例如、500)增加到。有关这些配置选项的更多详细信息、请参见 <a href="#">"配置选项"</a> 页面。	Telegraf会发出警告、指出可锁定内存不足或崩溃。
尝试增加底层操作系统/节点中Telegraf可锁定内存的限制。如果不能增加限制、请修改nLMO代理配置并 将_ <i>Unprotoed</i> _设置为_true。这将指示Telegraf不尝试预留锁定的内存页。由于解密的机密可能会交换到磁盘、因此这可能会带来安全风险、但它允许在无法预留锁定内存的环境中执行。有关_ <i>UnprotoECE</i> _配置选项的更多详细信息、请参阅 <a href="#">"配置选项"</a> 页面。	我看到Telegraf发出的警告消息如下所示：W ! [/Inputs.diskio]无法收集"vdc"的磁盘名称：读取/dev/vdc时出错：没有此文件或目录
对于Kubnetes监控操作员、这些警告消息不会产生负面影响、可以放心地忽略。或者、也可以编辑AgentConfiguration中的" <i>tendraf</i> "部分、并将_ <i>runDs</i> 专用_ <i>设置为true</i> 。有关详细信息，请参阅 <a href="#">"操作员配置选项"</a> 。	我的流畅位POD出现故障、并出现以下错误： [2024/10/16 14: 16: 23][error][src/fluent-bit/plugins/in_outle/Tail_fs_inoTIFY.c: 360 errno=24] 打开的文件过多[2024/10/16 14: 16: 23][error] failed initiation input. 0 [2024/10/16: 16: 23] input[引擎初始化失败][错误]

问题:	请尝试以下操作:
尝试更改集群中的_fsNOTES_设置:	<p>该报电DS Pod报告错误、因为Kubernetes输入插件由于无法验证TLS证书而无法发出HTTP请求。例如：E! [Inputs.Kubernetes]插件中的错误：发出获取TLS的"<a href="https://&lt;kubelet_IP&gt;:10250/stats/summary">https://&lt;kubelet_IP&gt;:10250/stats/summary</a>": " class="bare"&gt;<a href="https://&lt;kubelet_IP&gt;:10250/stats/summary">https://&lt;kubelet_IP&gt;:10250/stats/summary</a>":&lt;/a&gt;HTTP请求时出错"&lt;a href="https://&lt;kubelet_IP&gt;:10250/stats/summary"&gt;https://&lt;kubelet_IP&gt;:10250/stats/summary": " class="bare"&gt;<a href="https://&lt;kubelet_IP&gt;:10250/stats/summary">https://&lt;kubelet_IP&gt;:10250/stats/summary</a>":&lt;/a&gt;：无法验证证书：X509：无法验证&lt;kubelet_IP&gt;证书、因为它不包含任何IP SAN</p> <pre> sudo sysctl fs.inotify.max_user_instances (take note of setting)  sudo sysctl fs.inotify.max_user_instances=&lt;something larger than current setting&gt;  sudo sysctl fs.inotify.max_user_watches (take note of setting)  sudo sysctl fs.inotify.max_user_watches=&lt;something larger than current setting&gt; </pre> <p>重新启动Fluent位。</p> <p>注意：要使这些设置在节点重新启动后保持不变、您需要在<code>/etc/sysctl.conf</code>中放置以下行</p> <pre> fs.inotify.max_user_instances=&lt;something larger than current setting&gt; fs.inotify.max_user_watches=&lt;something larger than current setting&gt; </pre>

有关其他信息，请参见["支持"](#)页面或["数据收集器支持列表"](#)。

## 内存缓存的 Data Collector

Data Infrastructure Insight使用此数据收集器从Memcached...收集指标。

### 安装

1. 从\*Observability > Collectors\*中，单击\*+Data Collector\*。选择Memcached.

选择安装了 Telegraf 代理的操作系统或平台。

- 如果尚未安装用于收集的代理，或者要为其他操作系统或平台安装代理，请单击 [Show Instructions](#) 展开说明。["代理安装"](#)
- 选择要用于此数据收集器的代理访问密钥。您可以通过单击 \* + 代理访问密钥 \* 按钮来添加新的代理访问密钥。最佳实践：仅当您要按操作系统 / 平台对数据收集器进行分组时，才使用其他代理访问密钥。
- 按照配置步骤配置数据收集器。这些说明因用于收集数据的操作系统或平台的类型而异。

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

Follow Configuration Steps [Need Help?](#)

- 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"]
## " " " "
```

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

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

## 设置

有关信息，请参见["memcached wiki"](#)。

## 对象和计数器

将收集以下对象及其计数器：

对象：	标识符：	属性：	数据点：
已缓存内存	命名空间服务器	节点 IP 节点名称	接受连接已处理身份验证 请求失败身份验证字节数 已用字节数读取（每秒） 写入的字节数（每秒） CAS Badval CAS 命中 CAS 未命中刷新 Reqs（ 每秒）获取 Reqs（每秒） 设置 Reqs（每秒）触 摸 Reqs（每秒）连接生 成率（每秒）连接结构打 开连接当前已存储项数 据删除请求命中（每秒）数 据删除请求未命中（每秒） 删除请求命中（每秒） 删除请求未命中（每秒） 项目已逐出有效的逐出已 过期项目获取命中（每秒） 获取未命中（每秒）已 用哈希字节哈希正在扩展 哈希功率级别增量请求命 中数（每秒）增量请求未 命中数（每秒）服务器最 大字节侦听已禁用已回收 的工作线程数已打开连接 总数已存储的项目数触摸 未命中服务器正常运行时 间

## 故障排除

有关其他信息、请参见["支持"](#)页面。

## MongoDB 数据收集器

Data Infrastructure Insight 使用此数据收集器从 MongoDB 收集指标。

### 安装

1. 从 \*Observability > Collectors\* 中，单击 \*+Data Collector\*。选择 MongoDB。

选择安装了 Telegraf 代理的操作系统或平台。

2. 如果尚未安装用于收集的代理，或者要为其他操作系统或平台安装代理，请单击 Show Instructions 展开说明。["代理安装"](#)
3. 选择要用于此数据收集器的代理访问密钥。您可以通过单击 \* + 代理访问密钥 \* 按钮来添加新的代理访问密

钥。最佳实践：仅当您要按操作系统 / 平台对数据收集器进行分组时，才使用其他代理访问密钥。

- 按照配置步骤配置数据收集器。这些说明因用于收集数据的操作系统或平台的类型而异。

 MongoDB  
Gathers MongoDB metrics.

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

RHEL & CentOS

### Select existing Agent Access Key or create a new one

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

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

### Follow Configuration Steps

Need Help?

- 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.
- 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@10.0.3.38:27017,  
##   ...  
##   ...
```
- Replace <INSERT\_MONGODB\_ADDRESS> with the applicable MongoDB server address. Please specify a real machine address, and refrain from using a loopback address.
- Replace <INSERT\_MONGODB\_PORT> with the applicable MongoDB port.
- Restart the Telegraf service.  

```
systemctl restart telegraf
```

## 设置

有关信息，请参见["MongoDB 文档"](#)。

## 对象和计数器

将收集以下对象及其计数器：

对象：	标识符：	属性：	数据点：
MongoDB	命名空间主机名		
MongoDB 数据库	命名空间主机名数据库名称		

## 故障排除

可从页面中找到相关信息["支持"](#)。

## MySQL Data Collector

Data Infrastructure Insight 使用此数据收集器从 MySQL 收集指标。

## 安装

1. 从 \*Observability > Collectors\* 中，单击 \*+Data Collector\*。选择 MySQL。

选择安装了 Telegraf 代理的操作系统或平台。

2. 如果尚未安装用于收集的代理，或者要为其他操作系统或平台安装代理，请单击 Show Instructions 展开说明。["代理安装"](#)
3. 选择要用于此数据收集器的代理访问密钥。您可以通过单击 \* + 代理访问密钥 \* 按钮来添加新的代理访问密钥。最佳实践：仅当您要按操作系统 / 平台对数据收集器进行分组时，才使用其他代理访问密钥。
4. 按照配置步骤配置数据收集器。这些说明因用于收集数据的操作系统或平台的类型而异。



## 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 your 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
```



## 设置

有关信息，请参见["MySQL 文档"](#)。

## 对象和计数器

将收集以下对象及其计数器：

对象：	标识符：	属性：	数据点：
MySQL	命名空间 MySQL 服务器	节点 IP 节点名称	已中止的客户端 (每秒) 已中止连接 (每秒) 的接收字节 (每秒) 的传输字节 (每秒) 命令管理员 (每秒) 命令更改事件命令 更改功能命令更改实例命令 更改操作步骤命令 更改服务器命令 更改表命令 更改表空间命令 更改用户命令 命令分析命令分配给密钥缓存命令 开始命令 二级日志命令 命令调用操作步骤命令 更改数据库命令 更改主命令 更改复制筛选器命令 检查命令 校验和命令 提交命令 创建数据库命令 创建事件命令 创建功能命令 创建索引命令 创建操作步骤命令 创建服务器命令 创建表命令 创建触发器命令 创建 UDF 命令 创建用户命令 创建查看命令 分配 SQL 连接 错误接受创建的 tmp 磁盘 表延迟错误刷新命令 处理程序提交 InnoDB 缓冲区 池字节数据 密钥块未刷新 密钥读取请求 密钥写入请求 密钥写入最大执行时间 超过最大已用连接数 打开文件 性能架构 帐户丢失 已准备的临时计数 Qcache 可用块 查询问题 选择完全 加入选择 完整范围 加入选择 范围检查 选择扫描 表锁定 立即

## 故障排除

有关其他信息、请参见["支持"](#)页面。

## Netstat 数据收集器

Data Infrastructure Insight 使用此数据收集器收集 Netstat 指标。

## 安装

- 从\*Observability > Collectors\*中，单击\*+Data Collector\*。选择Netstat。

选择安装了 Telegraf 代理的操作系统或平台。

- 如果尚未安装用于收集的代理，或者要为其他操作系统或平台安装代理，请单击 Show Instructions 展开说明。[“代理安装”](#)
- 选择要用于此数据收集器的代理访问密钥。您可以通过单击 \* + 代理访问密钥 \* 按钮来添加新的代理访问密钥。最佳实践：仅当您要按操作系统 / 平台对数据收集器进行分组时，才使用其他代理访问密钥。
- 按照配置步骤配置数据收集器。这些说明因用于收集数据的操作系统或平台的类型而异。

The screenshot shows the configuration page for the Netstat collector. It includes sections for selecting the operating system (Windows), choosing an agent access key (Default or Create New), and step-by-step instructions for configuration.

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

Follow Configuration Steps [Need Help?](#)

- 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"
```
- Restart the Telegraf service.

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

## 设置

### 对象和计数器

将收集以下对象及其计数器：

对象：	标识符：	属性：	数据点：
netstat	节点UUID	节点 IP 节点名称	

## 故障排除

有关其他信息、请参见["支持"](#)页面。

## nginx 数据收集器

Data Infrastructure Insight使用此数据收集器从Nginx收集指标。

### 安装

1. 从\*Observability > Collectors\*中，单击\*+Data Collector\*。选择"NGINX"。

选择安装了 Telegraf 代理的操作系统或平台。

2. 如果尚未安装用于收集的代理，或者要为其他操作系统或平台安装代理，请单击>Show Instructions\_展开说明。["代理安装"](#)
3. 选择要用于此数据收集器的代理访问密钥。您可以通过单击 \* + 代理访问密钥 \* 按钮来添加新的代理访问密钥。最佳实践：仅当您要按操作系统 / 平台对数据收集器进行分组时，才使用其他代理访问密钥。
4. 按照配置步骤配置数据收集器。这些说明因用于收集数据的操作系统或平台的类型而异。

Nginx Configuration  
Gathers Nginx metrics.

What Operating System or Platform Are You Using?

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 your 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/ngx\\_http\\_stub\\_status\\_module.html](http://nginx.org/en/docs/http/ngx_http_stub_status_module.html)

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

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

- 4 Reload the configuration:

```
nginx -s reload
```

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

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

- 6 Replace `<INSERT_NGINX_ADDRESS>` with the applicable Nginx address. Please specify a real machine address, and refrain from using a loopback address.

- 7 Replace `<INSERT_NGINX_PORT>` with the applicable Nginx port.

- 8 Restart the Telegraf service.

```
systemctl restart telegraf
```

## 设置

要收集Nginx指标，需要启用Nginx "`http_stub_status_module`"。

有关其他信息，请参见 "[nginx 文档](#)"。

## 对象和计数器

将收集以下对象及其计数器：

对象：	标识符：	属性：	数据点：
nginx	命名空间服务器	节点 IP 节点名称端口	接受正在进行的已处理读取请求，正在等待写入

## 故障排除

有关其他信息、请参见["支持"](#)页面。

## PostgreSQL 数据收集器

Data Infrastructure Insight 使用此数据收集器从 PostgreSQL 收集指标。

### 安装

1. 从 \*Observability > Collectors\* 中，单击 \*+Data Collector\*。选择 PostgreSQL。

选择安装了 Telegraf 代理的操作系统或平台。

2. 如果尚未安装用于收集的代理，或者要为其他操作系统或平台安装代理，请单击 Show Instructions 展开说明。["代理安装"](#)
3. 选择要用于此数据收集器的代理访问密钥。您可以通过单击 \* + 代理访问密钥 \* 按钮来添加新的代理访问密钥。最佳实践：仅当您要按操作系统 / 平台对数据收集器进行分组时，才使用其他代理访问密钥。
4. 按照配置步骤配置数据收集器。这些说明因用于收集数据的操作系统或平台的类型而异。



PostgreSQL

## PostgreSQL Configuration

Gathers PostgreSQL metrics.

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

[Need Help?](#)

RHEL &amp; CentOS



### Select existing Agent Access Key or create a new one

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

[+ Agent Access Key](#)

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

### Follow Configuration Steps

[Need Help?](#)

- 1 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
```



## 设置

有关信息，请参见["PostgreSQL 文档"](#)。

## 对象和计数器

将收集以下对象及其计数器：

对象：	标识符：	属性：	数据点：
PostgreSQL 服务器	命名空间数据库服务器	节点名称节点 IP	缓冲区已分配缓冲区后端缓冲区后端文件同步缓冲区检查点缓冲区清理检查点同步时间检查点写入时间检查点请求检查点已写入时间最大值
PostgreSQL 数据库	命名空间数据库服务器	数据库 OID 节点名称节点 IP	块读取时间块写入时间块命中块读取冲突死锁客户端编号临时文件字节临时文件编号行删除行提取行插入行返回行更新已提交事务已回滚

## 故障排除

有关其他信息、请参见["支持"](#)页面。

## Puppet Agent 数据收集器

Data Infrastructure Insight 使用此数据收集器从 Puppet Agent 收集指标。

### 安装

1. 从 \*Observability > Collectors\* 中，单击 \*+Data Collector\*。选择 Puppet。

选择安装了 Telegraf 代理的操作系统或平台。

2. 如果尚未安装用于收集的代理，或者要为其他操作系统或平台安装代理，请单击 Show Instructions 展开说明。["代理安装"](#)
3. 选择要用于此数据收集器的代理访问密钥。您可以通过单击 \* + 代理访问密钥 \* 按钮来添加新的代理访问密钥。最佳实践：仅当您要按操作系统 / 平台对数据收集器进行分组时，才使用其他代理访问密钥。
4. 按照配置步骤配置数据收集器。这些说明因用于收集数据的操作系统或平台的类型而异。



## 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 your 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
```

## 设置

有关信息，请参见 ["Puppet 文档"](#)

## 对象和计数器

将收集以下对象及其计数器：

对象：	标识符：	属性：	数据点：
-----	------	-----	------

Puppet 代理	命名空间节点 UUID	节点名称位置节点 IP 版本 配置字符串版本 Puppet	更改总数事件失败事件成功事件总资源更改资源失败资源无法重新启动资源异步资源重新启动资源已计划资源已跳过资源总时间锁定时间配置检索时间Cron Time Exec 时间文件分段时间 lastrun 时间包时间计划时间服务时间Sshauthorizedkey 时间总计 时间用户
-----------	-------------	----------------------------------	---

## 故障排除

有关其他信息、请参见["支持"](#)页面。

## Redis Data Collector

Data Infrastructure Insight 使用此数据收集器从 Redis 收集指标。Redis 是一种开源内存数据结构存储，用作数据库，缓存和消息代理，支持以下数据结构：字符串，哈希，列表，集等。

## 安装

1. 从\*Observability > Collectors\*中，单击\*+Data Collector\*。选择Redis。

选择安装了 Telegraf 代理的操作系统或平台。

2. 如果尚未安装用于收集的代理，或者要为其他操作系统或平台安装代理，请单击 Show Instructions 展开说明。["代理安装"](#)
3. 选择要用于此数据收集器的代理访问密钥。您可以通过单击 \* + 代理访问密钥 \* 按钮来添加新的代理访问密钥。最佳实践：仅当您要按操作系统 / 平台对数据收集器进行分组时，才使用其他代理访问密钥。
4. 按照配置步骤配置数据收集器。这些说明因用于收集数据的操作系统或平台的类型而异。



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

### Follow Configuration Steps

[Need Help?](#)

- 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
```



- 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>
```



- 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.
  ## -----
```



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

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

- Restart the Telegraf service.

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



## 设置

有关信息，请参见["Redis 文档"](#)。

## 对象和计数器

将收集以下对象及其计数器：

对象：	标识符：	属性：	数据点：
Redis	命名空间服务器		

## 故障排除

有关其他信息、请参见["支持"](#)页面。

## 版权信息

版权所有 © 2025 NetApp, Inc.。保留所有权利。中国印刷。未经版权所有者事先书面许可，本文档中受版权保护的任何部分不得以任何形式或通过任何手段（图片、电子或机械方式，包括影印、录音、录像或存储在电子检索系统中）进行复制。

从受版权保护的 NetApp 资料派生的软件受以下许可和免责声明的约束：

本软件由 NetApp 按“原样”提供，不含任何明示或暗示担保，包括但不限于适销性以及针对特定用途的适用性的隐含担保，特此声明不承担任何责任。在任何情况下，对于因使用本软件而以任何方式造成的任何直接性、间接性、偶然性、特殊性、惩罚性或后果性损失（包括但不限于购买替代商品或服务；使用、数据或利润方面的损失；或者业务中断），无论原因如何以及基于何种责任理论，无论出于合同、严格责任或侵权行为（包括疏忽或其他行为），NetApp 均不承担责任，即使已被告知存在上述损失的可能性。

NetApp 保留在不另行通知的情况下随时对本文档所述的任何产品进行更改的权利。除非 NetApp 以书面形式明确同意，否则 NetApp 不承担因使用本文档所述产品而产生的任何责任或义务。使用或购买本产品不表示获得 NetApp 的任何专利权、商标权或任何其他知识产权许可。

本手册中描述的产品可能受一项或多项美国专利、外国专利或正在申请的专利的保护。

有限权利说明：政府使用、复制或公开本文档受 DFARS 252.227-7013（2014 年 2 月）和 FAR 52.227-19（2007 年 12 月）中“技术数据权利 — 非商用”条款第 (b)(3) 条规定的限制条件的约束。

本文档中所含数据与商业产品和/或商业服务（定义见 FAR 2.101）相关，属于 NetApp, Inc. 的专有信息。根据本协议提供的所有 NetApp 技术数据和计算机软件具有商业性质，并完全由私人出资开发。美国政府对这些数据的使用权具有非排他性、全球性、受限且不可撤销的许可，该许可既不可转让，也不可再许可，但仅限在与交付数据所依据的美国政府合同有关且受合同支持的情况下使用。除本文档规定的情形外，未经 NetApp, Inc. 事先书面批准，不得使用、披露、复制、修改、操作或显示这些数据。美国政府对国防部的授权仅限于 DFARS 的第 252.227-7015(b)（2014 年 2 月）条款中明确的权利。

## 商标信息

NetApp、NetApp 标识和 <http://www.netapp.com/TM> 上所列的商标是 NetApp, Inc. 的商标。其他公司和产品名称可能是其各自所有者的商标。