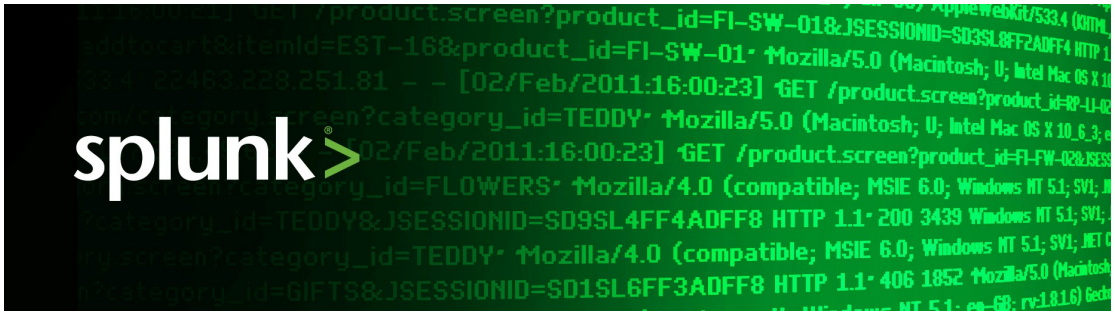


Splunk® Enterprise 7.2.0

指标

生成时间：2018 年 10 月 17 日，上午 8:55

Table of Contents

指标简介

指标概述

指标是一个适合于系统管理员和 IT 工具工程师使用的功能，主要用来实时收集、调查、监视和共享技术基础设施、安全系统和业务应用程序中的指标。

Splunk 平台中的指标使用自定义索引类型，该类型索引针对指标存储和检索而进行优化。要使用指标，包含了 `mstats` 命令以应用数字聚合（如平均、总和、百分比等），将来自不同数据来源的问题隔离并关联起来。

什么是指标？

指标是一组测量值，包含时间戳、指标名称、值和维度。

时间戳
: 表示何时进行指标测量。

指标名称
: 使用用圆点隔开的层次结构引用名称空间（例如，nginx.upstream.responses.5xx）。任何字符串均可用作指标名称。我们建议指标名称中只包含小写字母、数字、下划线和圆点。圆点用于隔开名称空间分段，以创建指标层次结构。

值
: 数字数据点代表指标值（如计数），或者特定时间精度的计算值（如前一分钟响应时间指标的百分位）。

维度
: 提供指标元数据。例如：

    区域：us-east-1、us-west-1、us-west-2、us-central1
    实例类型：t2.medium、t2.large、m3.large、n1-highcpu-2
    技术：nginx、redis、tomcat

您可以将指标名称视为您正在测量的内容，而维度是您可用来过滤结果或对其进行分组的分类。

以下为生成指标的系统示例：

- IT 基础设施，如主机、网络和设备
- 系统组件，如网络服务器和数据库
- 应用程序特定指标，如测量功能性能的计时器
- SaaS
- 传感器，如 IoT

Splunk 平台中的指标是什么？

指标是 Splunk 平台的一项功能。指标包括：

- 指标集合框架，可用于收集和获取大容量指标测量值，通常使用圆点符号隔开从代理和 API 获取的指标名称和指标层次结构，如 collectd.host.docker_stats.app.task.cpu.system。
- 支持收集现有的线路指标协议，如 collectd、StatsD 和 DogStatsD 的方法。
- 通用转发器 (UF) 和重型转发器 (HWF) 的可使用的框架，用于收集指标并安全有效地将指标负载转发到独立的指标索引或指标索引群集。
- 指标引入管道，支持针对结构化良好的指标负载协议在索引时间应用弹出和转换。
- 特定于指标的数据目录，提供浏览和过滤大量数据名称和维度的方法。

有关 Splunk 平台中的指标数据示例，请参阅“数据导入”。

指标入门

Splunk 平台从不同的数据来源收集指标，并将此数据存储到新的索引类型中，该索引类型是针对引入和检索数据进行优化的。

Splunk 平台本机支持以下指标收集工具：

- 收集代理，一个带 write_HTTP 插件的基于 Unix 的守护进程。Collectd 支持 100 多个前端插件。
- StatsD 线路协议，被大范围的客户端库和其他开源工具使用。

这两个工具都属于轻量型工具，易于使用，且有一个较大的支持社区。如果您想要收集应用程序和系统的性能指标，请查看这些工具确定其是否适合您的环境。

如果您更喜欢使用不同的指标收集工具，则可以使用 Splunk 平台通过手动配置来收集和分析数据。

指标数据格式

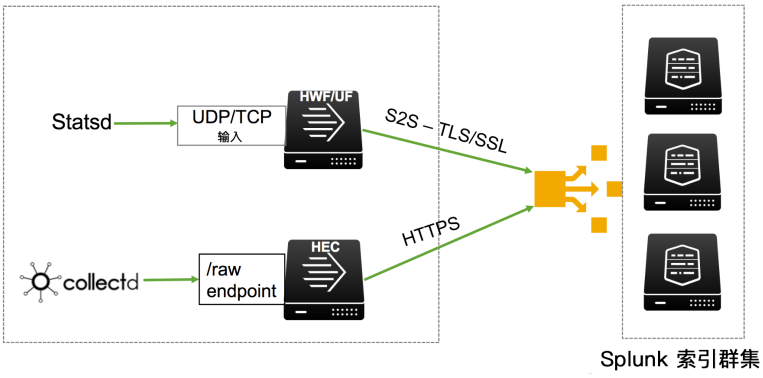
指标数据使用含以下字段的特定格式。

字段	必填	可写入或内部	描述	示例
metric_name	X	可写入	指标名称。	os.cpu.user
_time	X	可写入	UNIX 时间表示法中的指标时间戳。	2017 年 8 月 14 日，17:12:39.000
_value	X	可写入	指标的数字值。此字段为 64 位浮点数，精确到小数点 15 到 17 位。	42.12345
<dimension 0> ... <dimension *n*>	X	可写入	任意数量的字段，表示如何拆分指标。	ip
_dims	X	内部	自动生成的内部字段，包含指标事件中所有维度的名称。此字段用于返回指标索引中独特的维度名称列表。	_dims::ip
metric_type		可写入	指标类型。只支持 "g" (gauge)。	g
source		内部	指标数据来源。	udp:8125
host	X	内部	原始主机。Splunk 软件中的标准字段。	server007
index	X	内部	指标索引名称。Splunk 软件中的标准字段。	metricsindex
sourcetype	X	内部	指标的数字结构。Splunk 软件中的标准字段。	statsd

支持的线路协议

Splunk 平台中的指标本机支持以下指标线路协议：

- 通过 UDP/TCP 的普通 StatsD
- 通过 UDP/TCP 的带维度的 StatsD 扩展名
- 使用 HTTP 事件收集器 (HEC) 的 HTTPS 上的 Collectd



有关数据导入的详细信息，请参见“从 StatsD 导入指标”和“从 collectd 导入指标”。

要支持其他线路指标协议，您可以使用自定义转换，以将来自其他工具的指标数据导入 Splunk 平台。请参阅“从其他客户端获取指标”了解详细信息。

指标来源类型

Splunk 平台包括以下预先训练的来源类型，以支持最受支持的线路指标协议：

来源类型名称	描述
statsd	使用针对普通 StatsD 和带维度的 StatsD 扩展的指标线路协议支持数据。
collectd_http	使用针对 collectd 的指标线路协议支持数据。
metrics_csv	支持 CSV 格式的数据。有关用法的详细信息，请参阅“从其他来源导入指标”。

指标索引

要尽可能有效地存储和分析指标数据，指标数据要存储在只针对指标的新型索引。指标索引只用于指标数据。不能将事件索引转换为指标索引，反之亦然。

有关更多信息，请参阅*《管理索引器和索引器群集》*手册中的“创建指标索引”。

有关如何测量指标数据的信息，请参阅*《管理员手册》*中的“Splunk Enterprise 许可如何工作”。

默认指标索引

您可以向用户角色分配默认指标索引。请参阅*《确保 Splunk 安全》*中的“通过 Splunk Web 添加和编辑角色”。

通过指标命令，如 `mcatalog` 或 `mstats` 运行搜索且未按特定索引筛选搜索时，搜索将自动搜索分配给您的角色的默认索引。如果运行不按特定指标筛选的指标搜索并且您的角色没有分配默认指标索引，指标搜索将运行于空数据集。

搜索和带指标的 CLI 命令

要在指标索引中分析数据和枚举项目，请使用 `mstats` 和 `mcatalog` 搜索命令。和事件结合使用的其他搜索命令不会和指标结合使用。例如，`delete` 命令不会和指标结合使用。有关搜索指标索引的更多信息，请参阅“搜索和监视指标”。

管理 CLI 命令并非都和指标结合使用。使用 `-datatype metric` 参数时，您可以将 `add index` 和 `list index` 命令和指标结合使用。请参阅*《管理索引器和索引器群集》*手册中的“创建指标索引”。

指标数据导入

从 StatsD 导入指标

StatsD 是一个在 Node.js 平台上运行的网络守护进程，通过 UDP 或 TCP 发送指标。有关 StatsD 的概述，请参阅 Code as Craft 网站上的“测量所有，量化一切”。

StatsD 有几个实现，其中一些会以不同的方式对维度进行编码。Splunk 平台本机支持以下几种格式：

普通的 StatsD 线路指标协议：

```
<metric_name>:<_value>|<metric_type>
```

示例指标：

```
performance.os.disk:1099511627776|g
```

带维度扩展名的 StatsD 线路指标协议：

```
<metric_name>:<_value>|<metric_type>|#dim1:valueX,dim2:valueY
```

示例指标：

```
performance.os.disk:1099511627776|g|#region:us-west-1,datacenter:us-west-
1a,rack:63,os:Ubuntu16.10,arch:x64,team:LON,service:6,service_version:0,service_environment:test,
path:/dev/sda1,fstype:ext3
```

有关指标名称和维度格式的更多信息，请参阅“指标最佳实践”。

使用其他 StatsD 格式

如果您使用 StatsD 实现，该实现使用来自 Splunk 本机支持的维度格式，例如将维度嵌入指标名称，您仍可在 Splunk 平台中使用指标。但是，您需要自定义 Splunk 配置文件，以指定如何从格式中提取维度。

还可以使用 StatsD 收集指标，但是使用 collectd 通过 HTTP 将数据发送到 Splunk 平台。这种方法的好处在于 collectd 可以将指标数据中的维度格式规范化。请参阅“从 collectd 导入指标”了解更多信息。

为 StatsD 数据设置数据导入

在您配置 StatsD 协议中要发送数据的数据来源之后，请在 Splunk 平台中创建一个 UDP 或 TCP 数据导入，以在打开的端口侦听 StatsD 数据。

1. 在 Splunk Web 中前往**设置 > 数据导入**。
2. 在**本地输入**中，单击 **UDP** 或 **TCP** 旁边的**新增**，具体取决于您想要创建何种类型的输入。

   使用 UDP 端口获取指标数据时，您无法使用并行引入或多管道设置功能。

3. 针对**端口**，请输入您针对 StatsD 正在使用的端口号。
4. 单击**下一步**。
5. 单击**选择来源类型**，然后选择**指标 > statsd**。
6. 关于**索引**，请选择现有的指标索引。或者，单击**创建新索引**来创建一个索引。
   如果您选择创建一个索引，请在**新索引**对话框中：
   1. 输入**索引名称**。用户定义的索引名称只能由数字、小写字母、下划线和连字符组成。索引名称不能以下划线或连字符开头。
   2. 请单击**指标**选择**索引数据类型**。
   3. 需要时，配置其他索引属性。
   4. 单击**保存**。
7. 单击**查看**，然后单击**提交**。

提取格式不受支持的 StatsD 维度

许多 StatsD 客户端都将维度名称嵌入指标名称。例如，假设您的 StatsD 客户端使用以下线路指标协议格式，Splunk 平台本机不支持此格式：

```
<dimension>.<metric_name>:<value>|<metric_type>
```

以下是使用这种不支持的格式返回的指标示例：

```
10.1.1.198.cpu.percent:75|g
```

指标字段值应为：

```
metric_name=cpu.percent
```

```
_value=75
metric_type=g
```

提取的维度应为：

```
ip=10.1.1.198
```

要创建正确的结果，您必须编辑 Splunk 配置文件或使用 REST API 创建自定义来源类型，指定如何从此指标数据中提取维度。

通过编辑配置文件配置维度提取

1. 为 StatsD 指标数据定义自定义来源类型。
   1. 在文本编辑器中，从您要使用的位置的本地目录中打开 props.conf 配置文件，如搜索和报告应用 ($SPLUNK_HOME/etc/apps/search/local/) 或系统 ($SPLUNK_HOME/etc/system/local)。如果此位置中不存在 props.conf 文件，请创建文本文件并将其保存到该位置。
   2. 将段落添加到 props.conf 文件中，如下所示：

      ```
      # props.conf

      [<metrics_sourcetype_name>]
      METRICS_PROTOCOL = statsd
      STATSD-DIM-TRANSFORMS = <statsd_dim_stanza_name1>,<statsd_dim_stanza_name2>...
      ```
      - *metrics_sourcetype_name*：您的自定义指标来源类型名称。
      - *statsd_dim_stanza_name*：以逗号隔开的转换段落名称列表，该名称指定了如何提取维度。如果只有一个段落用于来源类型，且如果转换段落名称和 *metrics_sourcetype_name* 一样，此 STATSD-DIM-TRANSFORMS 设置可忽略。

- 定义一个或多个正则表达式从 *metric_name* 提取维度。

1. 在文本编辑器中，从您要使用的位置的本地目录中打开 transforms.conf 配置文件，如搜索和报告应用 ($SPLUNK_HOME/etc/apps/search/local/) 或系统 ($SPLUNK_HOME/etc/system/local)。如果此位置中不存在 transforms.conf 文件，请创建文本文件并将其保存到该位置。
2. 将段落添加到各正则表达式中，如下所示：

   ```
   # transforms.conf

   [statsd-dims:<unique_transforms_stanza_name>]
   REGEX = <regular expression>
   REMOVE_DIMS_FROM_METRIC_NAME = <Boolean>
   ```

   - *unique_transforms_stanza_name*：此段落的唯一名称。
   - REGEX = *<regular expression>*：定义如何从 StatsD 指标数据匹配和提取维度的正则表达式。Splunk 平台支持命名的捕获组提取格式 `(?<dim1>group)(?<dim2>group)...`，为提取的相应值提供维度名称。
   - REMOVE_DIMS_FROM_METRIC_NAME = *<Boolean>*：指定 StatsD 用圆点隔开的名称分段中不匹配的分段是否用作 *metric_name*。

     如果为 `true`，将维度值从测量值中移除，未匹配部分成为 *metric_name*。默认值为 true。

     如果为 `false`，提取的维度值包括在 *metric_name* 中。

     例如，维度测量名称为 "x.y.z"。正则表达式匹配 "y" 和 "z"。如果 REMOVE_DIMS_FROM_METRIC_NAME 是 `true`，则 *metric_name* 是 "x"。如果为 `false`，则 *metric_name* 是 "x.y.z"。

- 如为 StatsD 数据设置数据导入中所述，为此来源类型创建数据导入，然后选择自定义来源类型。

有关编辑这些配置文件的更多信息，请参阅*《管理员》*手册中的“关于配置文件，props.conf 和 transforms.conf”。

配置维度提取示例

假设您有 StatsD 指标数据，例如：

```
data=mem.percent.used.10.2.3.4.windows:33|g
```

您需要提取 "ipv4" 和 "os" 维度。

如果您定义两个正则表达式，一个针对 "ipv4"，另一个针对 "os"，您可以将以下段落添加到配置文件中：

```
# props.conf.example

[my_custom_metrics_sourcetype]
```

```
METRICS_PROTOCOL = statsd
STATSD-DIM-TRANSFORMS = regex_stanza1, regex_stanza2

# transforms.conf.example

[statsd-dims:regex_stanza1]
REGEX = (?<ipv4>\d{1,3}.\d{1,3}.\d{1,3}.\d{1,3})
REMOVE_DIMS_FROM_METRIC_NAME = true
[statsd-dims:regex_stanza2]
REGEX = \S+\.(?<os>\w+):
REMOVE_DIMS_FROM_METRIC_NAME = true
```

现在假设您可以使用单一正则表达式完成这个提取。在此情况下，您可以将以下段落添加到配置文件中：

```
# props.conf.example

[my_custom_metrics_sourcetype]
METRICS_PROTOCOL = statsd

# transforms.conf.example

[statsd-dims:my_custom_metrics_sourcetype]
REGEX = (?<ipv4>\d{1,3}.\d{1,3}.\d{1,3}.\d{1,3})\.(?<os>\w+):
REMOVE_DIMS_FROM_METRIC_NAME = true
```

注意，当只有单一正则表达式用于来源类型时，不需要 **props.conf** 配置文件中的 STATSD-DIM-TRANSFORMS 设置。

使用 REST API 为 StatsD 配置维度提取

1. 使用 /services/saved/sourcetypes REST 端点为 StatsD 指标数据定义自定义来源类型：

   ```
   https://<host>:<mPort>/services/saved/sourcetypes   \
   -d "name=<metrics_sourcetype_name>&METRICS_PROTOCOL=statsd&STATSD-DIM-
   TRANSFORMS=<statsd_dim_stanza_name>&SHOULD_LINEMERGE=false&ANNOTATE_PUNCT=false&ADD_EXTRA_TIME_FIELDS=false&DAT
   ```

   - *metrics_sourcetype_name*：您的自定义指标来源类型名称。
   - *statsd_dim_stanza_name*：转换段落名称列表，该段落名称指定了如何提取维度。如果只有一个段落用于来源类型，且如果转换段落名称和 *metrics_sourcetype_name* 一样，此 STATSD-DIM-TRANSFORMS 设置可忽略。

   例如，输入以下命令：

   ```
   curl -k -u admin:changeme https://localhost:8089/services/saved/sourcetypes   \
   -d "name=statsd_custom&METRICS_PROTOCOL=statsd&STATSD-DIM-TRANSFORMS=statsd-
   ex&SHOULD_LINEMERGE=false&ANNOTATE_PUNCT=false&ADD_EXTRA_TIME_FIELDS=false&DATETIME_CONFIG=CURRENT&pulldown_typ
   ```

2. 创建一个或多个正则表达式，使用 /data/transforms/statsdextractions REST 端点从 *metric_name* 中提取维度：

   ```
   https://<host>:<mPort>/services/data/transforms/statsdextractions \
   -d "name=<unique_transforms_stanza_name>&REGEX=<regular expression>&REMOVE_DIMS_FROM_METRIC_NAME=<Boolean>"
   ```

   - *unique_transforms_stanza_name*：此段落的唯一名称。
   - REGEX = <regular expression>：定义如何从 StatsD 指标数据匹配和提取维度的正则表达式。Splunk 平台支持命名的捕获组提取格式 `(?<dim1>group)(?<dim2>group)...`，为提取的相应值提供维度名称。
   - REMOVE_DIMS_FROM_METRIC_NAME = <Boolean>：指定 StatsD 用圆点隔开的名称分段中不匹配的分段是否用作 *metric_name*。

     如果为 `true`，将维度值从测量值中删除，未匹配部分成为 *metric_name*。默认值为 true。

     如果为 `false`，提取的维度值包括在 *metric_name* 中。

     例如，维度测量名称为 "x.y.z"。正则表达式匹配 "y" 和 "z"。如果 REMOVE_DIMS_FROM_METRIC_NAME 是 `true`，则 *metric_name* 是 "x"。如果为 `false`，则 *metric_name* 是 "x.y.z"。

   例如，输入以下命令：

   ```
   curl -k -u admin:changeme https://localhost:8089/services/data/transforms/statsdextractions \
   -d "name=statsd-ex&REGEX=\.(?<hostname>\S%2B?)\.(?
   <ip>\d{1,3}\.\d{1,3}\.\d{1,3}\.\d{1,3})&REMOVE_DIMS_FROM_METRIC_NAME=true"
   ```

3. 重新加载指标处理器，以使用 /admin/metrics-reload/_reload REST 端点加载配置更改：

```
https://<host>:<mPort>/services/admin/metrics-reload/_reload
```

例如，输入以下命令：

```
curl -k -u admin:changeme \
https://localhost:8089/services/admin/metrics-reload/_reload
```

4. 如为 StatsD 数据设置数据导入中所述，为此来源类型创建数据导入，然后选择自定义来源类型。

有关使用 Splunk REST API 的更多信息，请参见《*REST API 参考手册*》中的“使用 REST API 参考，/data/transforms/statsdextractions 和 /admin/metrics-reload/_reload”。

从 collectd 导入指标

Collectd 是一个开源守护进程，从各种数据来源中收集性能指标。Collectd 通过 collectd write_http 插件使用 HTTP 事件收集器 (HEC) 将指标数据发送到 Splunk 平台中的数据导入。

要使用 collectd 发送指标，请执行以下操作：

1. 配置 HTTP 事件收集器 (HEC) 数据导入。
2. 安装 collectd。
3. 配置 collectd。
4. 启动 collectd。

配置 HTTP 事件收集器 (HEC) 数据导入

HTTP 事件收集器 (HEC) 是一个端点，该端点可让您使用 HTTP 或安全 HTTP (HTTPS) 协议将应用程序事件发送至您的 Splunk 平台部署。设置 collectd 之前配置此数据导入，因为您配置 collectd 时需要使用数据导入详细信息。

1. 在 Splunk Web 中单击**设置 > 数据导入**。
2. 在**本地输入**中，单击 **HTTP 事件收集器**。
3. 确认 HEC 是否启用。
    1. 单击**全局设置**。
    2. 对于**所有标记**，如果尚未选择**启用**按钮，则单击此按钮。
    3. 注意 **HTTP 端口号**值，您需要用此值来配置 collectd。
    4. 单击**保存**。
4. 单击**新标记**配置 HEC 标记发送数据。
5. 在**选择数据来源**页面，请针对**名称**输入标记名称，如 "collectd token"。
6. 请勿选择其他选项。
7. 单击**下一步**。
8. 在**输入设置**页面，单击**选择**选择**来源类型**。
9. 单击**选择来源类型**，然后选择**指标 > collectd_http**。
10. 在**默认索引**旁边，选择指标索引或单击**创建新索引**创建一个索引。
    如果您选择创建一个索引，请在**新索引**对话框中：
    1. 输入**索引名称**。用户定义的索引名称只能由数字、小写字母、下划线和连字符组成。索引名称不能以下划线或连字符开头。
    2. 请单击**指标**选择**索引数据类型**。
    3. 需要时，配置其他索引属性。
    4. 单击**保存**。
11. 单击**查看**，然后单击**提交**。
12. 复制显示的**标记值**，您需要用此值配置 collectd。

直接将 collectd 事件添加到指标索引

要测试您的数据导入，您可以用 `/collector/raw` REST API 端点直接将 collectd 事件发送到指标索引，这样可接受 collectd JSON 格式的数据。您的指标索引已分配给 HEC 数据导入，该数据导入有其唯一的 HEC 标记，且来源类型为 "collectd_http"。

以下示例显示了 curl 命令，该命令将 collectd 事件发送到与 HEC 标记相关的索引：

```
curl -k https://localhost:8088/services/collector/raw?sourcetype=collectd_http   \
-H "Authorization: Splunk <HEC_token>"                                           \
-d
'[{"values":[164.9196798931339196],"dstypes":["derive"],"dsnames":["value"],"time":1505356687.894,"interval":10.000,"
```

您可以验证 HEC 数据导入是否正常，方法是将时间范围设置为“所有时间”的情况下，使用 `mcatalog` 运行搜索以列出所有指标名称，例如：

```
| mcatalog values(metric_name) WHERE index=<your_metrics_index> AND
metric_name=protocols.protocol_counter.InOctets.value
```

或者使用 Metrics Catalog REST 端点列出指标名称：

```
curl -k -u <admin:passwd> "https://localhost:8089/services/catalog/metricstore/metrics?earliest=0"
```

请参阅*《导入数据》*中的“关于 Splunk HTTP 事件收集器”和“为 HTTP 事件收集器格式化事件”。请参阅*《搜索参考》*手册中的 mstats 和 mcatalog。请参阅*《REST API 参考手册》*中的“Metrics Catalog 端点描述，/collector 和 /collector/raw”。

安装 collectd

在您想要收集指标的系统中，将 collectd 安装到计算机中。

1. 前往 collectd 网站中的“初始步骤”页面。
2. 遵照操作系统中的安装说明，安装 collectd 版本 5.6 或更高版本。

配置 collectd

Collectd 服务器是一个可选的守护进程，可用于聚合来自不同输入和一对多 collectd 客户端的指标。

通过配置 collectd.conf 配置文件中的插件配置 collectd 客户端。collectd.conf 文件的位置取决于操作系统。更多信息，请参阅 collectd 网站“初始步骤”页面中的“配置”。

write_http 插件

write_http 插件需要来自 HEC 数据导入的以下字段：

字段名称	描述	语法	
URL	提交值的 URL。此 URL 包括您的 Splunk 主机（IP 地址、主机名或负载均衡器名称）和 HTTP 端口号。	URL "https://<Splunk_host>:<HTTP_port>/services/collector/raw"	URL "https://10.66.104.1
标头	添加到请求的 HTTP 标头。	标头 "Authorization: Splunk <HEC_token>"	标头 "Authorization 9a3c-273e3a75aa2
格式	数据格式。	格式 "JSON"	格式 "JSON"

启用和配置插件

通过取消注释插件的 `LoadPlugin` 语句启用以下各插件，然后按照说明配置插件。大部分插件用于收集基本的 OS 级指标。调试时需要**日志文件**插件。您可以根据自身需求配置其他插件。

您可能需要单独安装一些插件，取决于您的安装方法和操作系统。有关详细信息，请参阅 collectd 网站。

插件	建议配置

cpu	`LoadPlugin cpu`
	`<Plugin cpu>`
	`  ReportByCpu true`
	`</Plugin>`

界面	`LoadPlugin interface` 使用默认配置。

负载	`LoadPlugin load`
	`<Plugin load>`
	`  ReportRelative true`
	`</Plugin>`

日志文件	`LoadPlugin logfile`
	`<Plugin logfile>`
	`  LogLevel info`
	`  File STDOUT`
	`  Timestamp true`
	`  PrintSeverity false`
	`</Plugin>`

内存	`LoadPlugin memory`
	`<Plugin memory>`
	`  ValuesAbsolute true`
	`  ValuesPercentage true`
	`</Plugin>`

网络	`LoadPlugin network` 仅当 collectd 客户端和 connectd 服务器不在同一台机器上时启用此插件，然后使用默认配置。
syslog	`LoadPlugin syslog` 使用默认配置。

write_http	您需要来自 HEC 数据导入的值配置此插件。 `LoadPlugin write_http`
	`<Plugin write_http>`
	`  <Node "node1">`
	`  URL "https://<Splunk_host>:<HTTP_port>/services/collector/raw"`
	`  Header "Authorization: Splunk <HEC_token>"`
	`  Format "JSON"`
	`  VerifyPeer false`
	`  VerifyHost false`
	`  Metrics true`
	`  StoreRates true`
	`  </Node>`
	`</Plugin>`

启动 collectd

要启动 collectd，请遵循 collectd 网站上的初始步骤页面上的“启动守护进程”下的指令。

必须安装 collectd.conf 文件中所有启用插件的模块。显示任何缺失的模块错误。有关可用 collectd 插件的更多信息，请参阅 collectd Wiki 网站中的“插件表”。

根据您的操作系统安装模块。例如，在 Linux 系统上，您必须安装 collectd-write_http.x86_64 才能使用 **write_http** 插件。

提示：

- 关于故障排除，请参阅通过**日志文件**插件启用的 collectd 日志文件获取详细信息。
- 使用**日志文件**插件中的 `File` 设置写入特定的文件而不是标准输出。例如：

```
<Plugin logfile>
    LogLevel info
    File "/var/log/collectd.log"
    Timestamp true
    PrintSeverity false
</Plugin>
```

- 如果您正在 Linux 中安装 collectd，您可以使用 yum 列出可用模块。例如，使用此 CLI 命令：

```
yum list | grep collectd
```

- 在 collectd.conf 文件中，将 `FQDNLookup` 设置设为 `false` 为域名呈现一个友好名称。

从其他来源导入指标

如果您要收集非本机支持的来源指标，您仍可以将此指标数据添加到指标索引。

从 CSV 格式的文件导入指标

如果您的指标数据是 CSV 格式，请使用 `metrics_csv` 预先培训的来源类型。

CSV 文件必须具有以 `metric_timestamp`、`metric_name` 和 `_value` 字段开头的标题。所有其他字段均按**维度**处理。

字段名称	必填	描述	示例
`metric_timestamp`	**X**	Epoch 时间（自 1970 年 1 月 1 日以来经历的时间），以毫秒为单位。	1504907933.000
`metric_name`	**X**	使用圆点字符串表示法的指标名称。	os.cpu.percent
`_value`	**X**	数字值。	42.12345
维度		所有其他字段均按维度处理。	ip

要添加 CSV 数据到指标索引，请用以下格式创建数据导入：

- 来源类型：**指标 > metrics_csv**
- 索引：指标索引

请参阅*《数据导入》*手册中的"监视文件和目录"以及*《管理索引器和索引器群集》*手册中的"创建指标索引"。

CSV 文件指标输入示例

这是一个针对指标进行合理格式化的 CSV 文件示例。表格的前三列为必填字段，`metric_timestamp`、`metric_name` 和 `_value`。第四列 `process_object_guid` 是维度。

```
"metric_timestamp","metric_name","_value","process_object_guid"
"1509997011","process.cpu.avg","2563454144","dbd1414b-378e-48bd-9735-bc2bab1e58fa"
"1509997011","process.cpu.min","2563454144","dbd1414b-378e-48bd-9735-bc2bab1e58fa"
"1509997011","process.cpu.max","2563454144","dbd1414b-378e-48bd-9735-bc2bab1e58fa"
"1509997011","process.cpu.last","2563454144","dbd1414b-378e-48bd-9735-bc2bab1e58fa"
"1509997011","process.ram.avg","2563454144","dbd1414b-378e-48bd-9735-bc2bab1e58fa"
"1509997011","process.ram.min","2563454144","dbd1414b-378e-48bd-9735-bc2bab1e58fa"
"1509997011","process.ram.max","2563454144","dbd1414b-378e-48bd-9735-bc2bab1e58fa"
"1509997011","process.ram.last","2563454144","dbd1414b-378e-48bd-9735-bc2bab1e58fa"
"1509997011","process.disk.avg","38750","dbd1414b-378e-48bd-9735-bc2bab1e58fa"
"1509997011","process.disk.min","38750","dbd1414b-378e-48bd-9735-bc2bab1e58fa"
"1509997011","process.disk.max","38750","dbd1414b-378e-48bd-9735-bc2bab1e58fa"
"1509997011","process.disk.last","38750","dbd1414b-378e-48bd-9735-bc2bab1e58fa"
```

要将此指标数据导入系统，创建使用预置 `metrics_csv` 来源类型、并能将指标数据移动到指标索引的输入。

设置 `metrics_csv` 输入后，通用转发器上将存在下列 `inputs.conf` 配置：

```
#inputs.conf

[monitor:///opt/metrics_data]
index = metrics
sourcetype = metrics_csv
```

通用转发器监视 CSV 数据并将其发送到指标索引器。设置 `metrics_csv` 输入后，指标索引器上将存在下列 `indexes.conf` 配置：

```
#indexes.conf

[metrics]
homePath = $SPLUNK_DB/metrics/db
coldPath = $SPLUNK_DB/metrics/colddb
thawedPath = $SPLUNK_DB/metrics/thaweddb
datatype = metric
maxTotalDataSizeMB = 512000
```

通过 TCP/UDP 从客户端导入指标

您可以通过手动配置数据来源类型，然后定义正则表达式指定 Splunk 软件应如何提取需要的指标字段，将来自非本机支持的客户端的指标数据添加到指标索引。请参阅“指标数据格式”。

例如，假设您正在使用 Graphite。Graphite 的纯文本协议格式是：

```
<metric path><metric value><metric timestamp>
```

示例指标可能是：

```
510fcbb8f755.sda2.diskio.read_time 250 1487747370
```

要为这些指标建立索引，请编辑 Splunk 配置文件以手动指定如何提取字段。

通过编辑配置文件配置字段提取

1. 为指标数据定义自定义来源类型。
    1. 在文本编辑器中，从您要使用的位置的本地目录中打开 props.conf 配置文件，如搜索和报告应用 ($SPLUNK_HOME/etc/apps/search/local/) 或系统 ($SPLUNK_HOME/etc/system/local)。如果此位置中不存在 props.conf 文件，请创建文本文件并将其保存到该位置。
    2. 将段落添加到 props.conf 文件中，如下所示：

```
# props.conf

[<metrics_sourcetype_name>]
TIME_PREFIX = <regular expression>
TIME_FORMAT = <strptime-style format>
TRANSFORMS-<class> = <transform_stanza_name>
NO_BINARY_CHECK = true
SHOULD_LINEMERGE = false
pulldown_type = 1
category = Metrics
```

    - *Metrics_sourcetype_name* 您的自定义指标来源类型名称。
    - TIME_PREFIX = *正则表达式*：正则表达式表示时间戳的位置。
    - TIME_FORMAT = *strptime-style 格式*：Strptime 格式字符串，用于提取日期。有关 strptime 的更多信息，请参阅*《数据导入》*手册中的“配置时间戳识别”。
    - TRANSFORMS-<*class*> = <*transform_stanza_name*>: *类*是识别要提取的字段命名空间的唯一文字字符串。*Transform_stanza_name* 是表示如何提取字段的 transforms.conf 中的段落名称。

- 为每个要提取的指标字段定义正则表达式。

1. 在文本编辑器中，从您要使用的位置的本地目录中打开 transforms.conf 配置文件，如搜索和报告应用 ($SPLUNK_HOME/etc/apps/search/local/) 或系统 ($SPLUNK_HOME/etc/system/local)。如果此位置中不存在 transforms.conf 文件，请创建文本文件并将其保存到该位置。
2. 将段落添加到各正则表达式中，如下所示：

```
# transforms.conf

[<transform_stanza_name>]
REGEX = <regular expression>
FORMAT = <string>
WRITE_META = true
```

- *transform_stanza_name*：此段落的唯一名称。
- REGEX = <*regular expression*>：定义如何从此指标数据中匹配和提取指标字段的正则表达式。
- FORMAT = <*string*>：指定指标事件格式的字符串。

- 如“为 StatsD 数据设置数据导入”中所述，为此来源类型创建数据导入，然后选择自定义来源类型。

有关编辑这些配置文件的更多信息，请参阅*《管理员》*手册中的“关于配置文件,props.conf 和 transforms.conf”。

配置字段提取示例

此示例显示如何创建自定义来源类型和正则表达式以从 Graphite 指标数据中提取字段。

```
# props.conf.example

[graphite_plaintext]
TIME_PREFIX = \s(\d{0,10})$
TIME_FORMAT =  %s
NO_BINARY_CHECK = true
```

```
SHOULD_LINEMERGE = false
pulldown_type = 1
TRANSFORMS-graphite-host = graphite_host
TRANSFORMS-graphite-metricname = graphite_metric_name
TRANSFORMS-graphite-metricvalue = graphite_metric_value
category = Metrics

# transforms.conf.example

[graphite_host]
REGEX = ^(\S[^\.]+)
FORMAT = host::$1
DEST_KEY = MetaData:Host
[graphite_metric_name]
REGEX = \.(\S+)
FORMAT = metric_name::graphite.$1
WRITE_META = true
[graphite_metric_value]
REGEX = \w+\s+(\d+.?\d+)\s+
FORMAT = _value::$1
WRITE_META = true
```

通过 HTTP 或 HTTPS 从客户端导入指标

如果您想要通过 HTTP 或 HTTPS 从非本机支持的客户端发送 JSON 格式的指标数据到指标索引，请使用 HTTP 事件收集器 (HEC) 和 `/collector` REST API 端点。

为 HEC 创建数据导入和标记

1. 在 Splunk Web 中单击**设置 > 数据导入**。
2. 在**本地输入**中，单击 **HTTP 事件收集器**。
3. 确认 HEC 是否启用。
    1. 单击**全局设置**。
    2. 对于**所有标记**，如果尚未选择**启用**按钮，则单击此按钮。
    3. 单击**保存**。
4. 单击**新标记**配置 HEC 标记发送数据。
5. 在**选择来源**页面，请输入标记名称，例如“指标标记”，选择**名称**。
6. 请勿选择其他选项。
7. 单击**下一步**。
8. 在**输入设置**页面，单击**新建**选择**来源类型**。
9. 在**来源类型**中，输入新来源类型名称。
10. 对于**来源类型类别**，请选择**指标**。
11. 还可在**来源类型描述**中输入描述。
12. 在**默认索引**旁边，选择指标索引或单击**创建新索引**创建一个索引。
    如果您选择创建一个索引，请在**新索引**对话框中：
    1. 输入**索引名称**。
    2. 请单击**指标**选择**索引数据类型**。
    3. 需要时，配置其他索引属性。
    4. 单击**保存**。
13. 单击**查看**，然后单击**提交**。
14. 复制显示的**标记值**。发送数据需要此 HEC 标记。

请参阅“用 Splunk 开发人员门户中 HTTP 事件收集器导入数据”。

用 HTTP 将数据发送到指标索引

使用 `/collector` REST API 端点和 HEC 标记直接将数据发送到指标索引，如下所示：

```
http://<Splunk_host>:<HTTP_port>/services/collector \
-H "Authorization: Splunk <HEC_token>"                                    \
-d "<metrics_data>"
```

您需要提供以下值：

- Splunk 主机（IP 地址、主机或负载均衡器名称）
- HTTP 端口号
- HEC 标记值
- 需要将 "event" 字段设为 "metric" 的指标数据。

有关 HEC 的更多信息，请参阅“用 Splunk 开发人员门户中的 HTTP 事件收集器和事件换行导入数据”。

有关 `/collector` 端点的更多信息，请参阅*《REST API 参考手册》*中的 /collector。

使用 HEC 发送指标示例

以下示例显示了发送指标测量值到指标索引的命令，值如下所示：

- Splunk 主机："localhost"
- HTTP 端口号："8088"
- HEC 标记值："b0221cd8-c4b4-465a-9a3c-273e3a75aa29"

```
curl -k https://localhost:8088/services/collector                    \
-H "Authorization: Splunk b0221cd8-c4b4-465a-9a3c-273e3a75aa29"       \
-d '{"time": 1486683865.000,"event":"metric","source":"disk","host":"host_99","fields":{"region":"us-west-
1","datacenter":"us-west-
1a","rack":"63","os":"Ubuntu16.10","arch":"x64","team":"LON","service":"6","service_version":"0","service_environment
```

将日志数据转换为指标

日志到指标的转换概述

指标通常存在于非结构化或半结构化日志数据中。Splunk 平台可自动将日志数据转换为指标数据点，然后将该数据插入您指定的指标索引。当您将日志数据引入 Splunk 平台部署时，或者您使用 `mcollect` 或 `meventcollect` 命令在日志数据上运行搜索时，平台可进行转换。

此功能遵循 Splunk 平台的旧功能，允许在引入时和搜索时提取事件中的字段。当您设置日志到指标的转换时，您可查看从非结构化事件中提取的字段-值对，并将捕获的数字字段识别为测量字段。Splunk 平台会为事件中的每个测量字段生成独立的、唯一指标数据点。

您可选择识别已提取字段用于 Splunk 平台加入黑名单，这样这些字段不会在指标数据点中显示。

通过 Splunk 平台将您尚未识别为测量或黑名单字段的已提取字段作为维度添加至指标数据点。相同事件中生成的所有指标数据点共享相同的维度字段-值对。

日志事件

这里有两种包含指标数据的日志事件：

_time	事件
08-05-2017 20:26:29.073 -0700	INFO 指标 - group=queue、name=aeq、max_size_kb=500、current_size_kb=300、current_size=53、largest_size=65、smallest_size=5
08-05-2017 20:26:29.075 -0700	INFO 指标 - group=queue、name=indexqueue、max_size_kb=500、current_size_kb=200、current_size=55、largest_size=85、smallest_size=0

Splunk 平台会运行一个流程，从事件中提取字段-值对并显示下表。

_time	组	name	max_size_kb	current_size_kb	current_size	largest_size
08-05-2017 20:26:29.073 -0700	队列	aeq	500	300	53	65
08-05-2017 20:26:29.075 -0700	队列	indexqueue	500	200	55	85

如果您将这些事件中的测量字段识别为 `max_size_kb`、`current_size_kb`、`current_size`、`largest_size` 和 `smallest_size`，Splunk 平台会为每个字段提取生成唯一的指标数据点。下表有 10 个指标数据点，分别用于两个原始日志事件中的各测量字段-值对。

_time	metric_name	_value	组	name
08-05-2017 20:26:29.073 -0700	max_size_kb	500	队列	aeq
08-05-2017 20:26:29.073 -0700	current_size_kb	300	队列	aeq
08-05-2017 20:26:29.073 -0700	current_size	53	队列	aeq
08-05-2017 20:26:29.073 -0700	largest_size	65	队列	aeq
08-05-2017 20:26:29.073 -0700	smallest_size	5	队列	aeq
08-05-2017 20:26:29.075 -0700	max_size_kb	500	队列	indexqueue
08-05-2017 20:26:29.075 -0700	current_size_kb	200	队列	indexqueue
08-05-2017 20:26:29.075 -0700	current_size	55	队列	indexqueue
08-05-2017 20:26:29.075 -0700	largest_size	85	队列	indexqueue
08-05-2017 20:26:29.075 -0700	smallest_size	0	队列	indexqueue

日志到指标的指标数据点的分析

每个指标数据点都包含一个 _time 字段、metric_name 字段和 _value 字段。指标数据点可有一个或多个维度字段。在“指标概述”中了解有关指标数据点的更多信息。

下表说明了日志到指标的转换流程如何派生每个指标数据点字段的值：

指标字段	示例值	源值
_time	08-05-2017 20:26:29.075 -0700	使用初始事件中的 _time 值。如果单个事件生成多个指标数据点，则这些数据点共享同一 _time 值。
metric_name	largest_size	使用为指标数据点提供 _value 的测量字段名称。在此情况下，测量字段命名为 largest_size。
_value	85	使用指标数据点依据的测量字段的值。在此情况下，指标数据点基于 largest_size=85。
维度字段	group=queue, name=indexqueue	日志事件中的不能识别为测量字段或黑名单字段的任何字段（除 _time 和 metric_name 之外）会变为维度字段。相同日志事件中生成的所有指标数据点共享相同的时间戳和维度字段-值对。

通过 Splunk Web 设置基本的引入时日志到指标的转换

当被引入的日志中的所有事件共享相同的字段时，使用 Splunk Web 设置引入时将日志转换为指标数据点。

Splunk Web 设置日志到指标的转换流程分为两个阶段：

1. 您可以在设置中的“来源类型”列表页面创建“日志到指标”类别的新来源类型。
2. 当您创建或编辑输入时，将“日志到指标”来源类型和适当的日志数据导入关联起来。

更多信息，请参阅“在 Splunk Web 中设置引入时日志到指标的转换”。

用 props.conf 和 transforms.conf 创建复杂的引入时将日志转换为指标

当引入日志中的事件有不同的测量字段集时，在 transforms.conf 和 props.conf 中手动创建配置，以在引入时将日志转换为指标。例如，您可以设计按共享字段值对事件进行排序的配置，然后将特定的日志到指标的转换规则应用于各事件组。

更多信息，请参阅“用配置文件设置引入时日志到指标的转换”。

在 Splunk Web 中设置引入时日志到指标的转换

您可以通过 Splunk Web 设置引入时日志到指标的转换。如果您希望 Splunk 平台保留特定指标索引中转换产生的指标数据点，您可能想要在引入时进行日志到指标的转换。

完成以下两个任务以设置引入时日志到指标的转换：

- 在 "Log to Metrics" 类别中创建来源类型。
- 将此来源类型应用到日志数据导入。

要使用此功能，您的角色必须有 edit_metric_schema 功能。如果您的角色没有此功能，您需要通过 Splunk Web 设置引入时将日志转换为指标，请联系 Splunk 管理员。

了解您的数据

创建 "Log to Metrics" 来源类型要求您对想要转换为指标数据点的日志数据有一定了解。您需要了解日志数据中的字段，和这些字段适合的类别。

字段类别	描述
测量	数值变为唯一的指标数据点的字段。
维度	提供指标数据点的其他元数据的字段。Splunk 平台将从日志事件中提取的、您尚未识别为测量或黑名单字段的所有字段作为维度。事件中生成的所有指标数据点都会共享维度字段-值对。
列入黑名单的字段	日志事件中的字段，该字段未显示在该事件生成的指标数据点中。对于指标数据点集合而言不重要的高基数字段适合加入字段黑名单。

例如，假设您有一个带时间戳和以下五个字段的事件：max_kb、min_kb、server_model、group 和 division。如果您将 max_kb 和 min_kb 识别为测量，将 group 和 division 识别为黑名单字段，Splunk 平台将产生两个指标数据点，分别用

于各测量字段。指标数据点将共享 `server_model` 作为维度字段。

创建 "Log to Metrics" 来源类型

您可以使用设置中的“来源类型”列表页面在 "Log to Metrics" 类别中创建来源类型。

前提条件

- 请参阅“日志到指标的转换概述”
- 关于“来源类型”列表页面的完整概述和添加新来源类型的流程，请参阅*数据导入*中的“管理来源类型”。

步骤

1. 选择**设置 > 来源类型**打开“来源类型”列表页面。
2. 单击**新来源类型**打开“创建来源类型”对话框。
3. 输入新来源类型的**名称**。
4. （可选）输入新来源类型的来源类型**描述**。必要时，选择不同的**目标应用**。
5. 选择**类别 > Log to Metrics**。
6. 为您的数据选择合适的**索引提取**。

   例如，如果您使用结构化 CSV 或 JSON 格式的数据，请选择 **csv** 或 **json**，视情况而定。如果您的数据在技术上是非结构化的，但是事件是字段-值对的字符串，请使用 **field_extraction**。
7. （可选）必要时，更改**事件换行**、**时间戳**和**高级**选项卡上的日志数据设置。
8. 单击**指标**选项卡显示 "Log to Metrics" 来源类型设置。

文本框标签	可选？	描述
测量	否	输入一个或多个数字测量字段名称，用逗号隔开。
黑名单	是	输入一个或多个与此来源类型相关联的日志事件生成的指标数据点中您想要列入黑名单的维度字段名称，用逗号隔开。您可能想要将对于指标集合来说不必要的高基数维度字段列入黑名单。

9. 单击**保存**。

为上载文件或目录中的数据应用 "Log to Metrics" 来源类型

当您在 "Log to Metrics" 类别中创建来源类型之后，您可使用“添加数据”工作流的“设置来源类型”步骤，为将单一文件指定为数据来源的数据导入应用来源类型。当您将 Log to Metric 类别来源类型设为该等输入后，**指标**下拉选项卡将显示在“设置来源类型”页面的左窗格中。使用此选项卡输入或更新测量列表，并将来源类型维度列入黑名单。

*数据导入*中详细介绍了“添加数据”工作流。

前提条件

- 请参阅“日志到指标功能概述”。
- 请参阅“创建 'Log to Metrics' 来源类型”。
- 请参阅*数据导入*中的“使用 Splunk Web 监视文件和目录”，查看将单一文件指定为数据来源的输入的“添加数据”工作流。
- 请参阅*数据导入*中的“设置来源类型”页，查看“添加数据”工作流的“设置来源类型”步骤概述。

步骤

1. 按照“添加数据”工作流说明上载或监视文件或目录，直到您打开“选择来源类型”页面。
2. 在“选择来源类型”页面，选择**来源类型 > Log to Metrics**并从列表中选择适当的来源类型。
   选择“Log to Metrics”来源类型只有，右侧预览面板不会填充指标数据的预览。您可以看到其他来源类型的预览。
3. （可选）打开**事件换行**、**时间戳**和**高级**下拉选项卡，必要时为数据导入更新设置。
4. （可选）打开**指标**下拉选项卡，在**测量**和**黑名单**文本框中输入或更新字段列表。**测量**需要至少一个字段。

文本框标签	描述
测量	预览此文本框中用逗号隔开的数字测量字段名称列表，必要时进行更新。为和此来源类型相关的日志事件中的每个测量字段-值对创建唯一的指标数据点。
黑名单	此文本框可包含用逗号隔开的您想要列入黑名单的维度字段列表，该字段来自与此来源类型相关联的日志事件生成的指标数据点。您可能想要将对于指标集合来说不必要的高基数维度字段列入黑名单。

5. 单击**下一步**继续数据导入的“添加数据”工作流。

用配置文件设置引入时日志到指标的转换

如果您可以访问部署的 props.conf 和 transforms.conf，您可手动配置比您可以用 Splunk Web 设置的更复杂的日志到指标的转换。例如，您可以设计可处理日志的数据到指标的转换，其中并非所有的事件都具有相同的测量和维度字段集。

要配置日志到指标的转换，您需要将段落添加到 props.conf 和 transforms.conf 文件。

1. 首先在 transforms.conf 文件中创建 [metric-schema] 段落，配置测量和黑名单维度。
2. 然后在 props.conf 来源类型段落中引用这些配置。

关于引用时将日志转换为指标数据点的概述，请参阅"日志到指标的转换概述"。

Transforms.conf 配置

基本的 transforms.conf 配置授予您和定义日志到指标的转换的 Splunk Web 方法相同的日志到指标的功能。这些配置允许您为所有事件具有相同的测量和维度字段的日志创建日志到指标的配置。

在 transforms.conf 中，您需要设置 [metric-schema] 段落，识别测量和黑名单维度列表。此配置语法如下所示：

```
[metric-schema:<unique_transforms_stanza_name>]
METRIC-SCHEMA-MEASURES = <measure_field1>, <measure_field2>,...
METRIC-SCHEMA-BLACKLIST-DIMS = <dimension_field1>, <dimension_field2>,...
```

METRIC-SCHEMA-MEASURES 和 METRIC-SCHEMA-BLACKLIST-DIMS 设置确定与段落相关的各日志事件如何转换为多个指标数据点。

设置语法	描述	是否必需？
METRIC-SCHEMA-MEASURES = <measure_field1>, <measure_field2>,...	提供测量字段的列表。Splunk 平台会为和 [metric-schema] 段落相关联的事件中的每个测量字段-值对生成独立的指标数据点。出现这种情况，测量字段名称会变成指标数据点的 metric_name 值，测量值会变成指标数据点的 _value 值。	是
METRIC-SCHEMA-BLACKLIST-DIMS = <dimension_field1>, <dimension_field2>,...	提供和名单维度字段的列表。这些字段不应显示为和 [metric-schema] 段落相关联的事件生成的指标数据点中的维度。您可能想要将对于指标集合来说不必要的高基数维度字段列入黑名单。	否

未识别为 METRIC-SCHEMA-MEASURES 测量字段或 METRIC-SCHEMA-BLACKLIST-DIMS 和黑名单维度字段的所有字段均在指标数据点中显示为维度。事件中的维度字段-值对被该事件产生的所有指标数据字段共享。

将日志到指标设置应用于日志中的特定事件

[metric-schema] 段落设置的辩题允许您根据该日志中所有事件共享的字段值，创建针对特定日志事件组的 METRIC-SCHEMA-MEASURES 和 METRIC-SCHEMA-BLACKLIST-DIMS 设置。相关语法如下所示：

```
[metric-schema:<unique_transforms_stanza_name>]
METRIC-SCHEMA-MEASURES-<unique_metric_name_prefix> = <measure_field1>, <measure_field2>,...
METRIC-SCHEMA-BLACKLIST-DIMS-<unique_metric_name_prefix> = <dimension_field1>, <dimension_field2>,...
```

<unique_metric_name_prefix> 必须和 metric_name 字段值匹配，该字段值由和 [metric-schema] 段落相关联的所有事件共享。metric_name 字段值应和 [metric-schema] 段落中的不同事件类型相对应。

如果日志事件尚未共享 metric_name 字段，有几种方式可将该字段添加到事件中。例如，您可以：

- 创建命名为 metric_name 的索引时字段提取。
- 使用 INGEST_EVAL 设置在引入时将共享的字段重新命名为 metric_name。

请参阅"目标日志到指标的转换示例"查看在引入时使用 INGEST_EVAL 将共享 group 字段重命名为 metric_name 的示例。

如果配置正确，METRIC-SCHEMA-MEASURES-<unique_metric_name_prefix> 设置会产生带遵循以下语法的 metric_name 值的指标数据点：<unique_metric_name_prefix>.<measure_field_name>。

始终结合使用 METRIC-SCHEMA-BLACKLIST-DIMS-<unique_metric_name_prefix> 设置和相应的 METRIC-SCHEMA-MEASURES-<unique_metric_name_prefix> 设置。

您不能在同一个段落中使用 METRIC-SCHEMA-MEASURES 和 METRIC-SCHEMA-MEASURES-<unique_metric_name_prefix>。您也不能在同一个段落中使用 METRIC-SCHEMA-BLACKLIST-DIMS 和 METRIC-SCHEMA-BLACKLIST-DIMS-<unique_metric_name_prefix>。

Props.conf 配置

创建 transforms.conf 配置之后，您需要将该等配置和 props.conf 中的来源类型关联起来。您可将配置添加到具有 METRIC-SCHEMA-TRANSFORMS 设置的来源类型段落中来执行此操作。此设置具有以下语法：

```
[ <sourcetype> ]
METRIC-SCHEMA-TRANSFORMS = <metric-schema:stanza_name>[,<metric-schema:stanza_name>]...
```

将日志到指标转换段落的名称放在配置的 <stanza_name> 部分。这样会将日志到指标转换段落和来源类型相关的日志事件关联起来。

日志到指标的转换设置的操作顺序

Splunk 会在基本的 METRIC-SCHEMA-MEASURES 和 METRIC-SCHEMA-BLACKLIST-DIMS 设置之前处理所有的 METRIC-SCHEMA-MEASURES-<unique_metric_name_prefix> 和 METRIC-SCHEMA-BLACKLIST-DIMS-<unique_metric_name_prefix> 设置。

换句话说，Splunk 平台会在处理事件不可知的日志到指标的设置之前，处理所有的针对事件的日志到指标的设置。这样后一组设置可处理 <unique_metric_name_prefix> 设置不处理的剩余事件。

目标日志到指标的转换示例

此处为事件集合。请注意：有两个具有不同测量和维度字段集的事件架构。事件共享 group 字段，group 值协调到两个事件架构中。

_time	事件
08-05-2017 20:26:29.073 -0700	INFO 指标 - group=queue、location=sf、corp=splunk、name=udp_queue、max_size_kb=0、current_size_kb=0、current_size=0、largest_size=0、smallest_size=0
08-05-2017 20:26:29.073 -0700	INFO 指标 - group=queue、location=sf、corp=splunk、name=aggqueue、max_size_kb=1024、current_size_kb=1、current_size=5、largest_size=35、smallest_size=0
08-05-2017 20:26:29.073 -0700	INFO 指标 - group=queue、location=sf、corp=splunk、name=auditqueue、max_size_kb=500、current_size_kb=0、current_size=0、largest_size=1、smallest_size=0
08-05-2017 20:26:29.075 -0700	INFO 指标 - group=pipeline、name=indexerpipe、processor=indexin、cpu_seconds=0、executes=171、cumulative_hits=2214401
08-05-2017 20:26:29.075 -0700	INFO 指标 - group=pipeline、name=indexerpipe、processor=index_thruput、cpu_seconds=0、executes=171、cumulative_hits=2214401
08-05-2017 20:26:29.075 -0700	INFO 指标 - group=pipeline、name=indexerpipe、processor=indexandforward、cpu_seconds=0、executes=171、cumulative_hits=2214401

检查完这些事件之后，您可决定是否需要在 transforms.conf 和 props.conf 中定义执行以下任务的配置集：

- 使用 INGEST_EVAL 在引入时将 group 字段的名称更改为 metric_name。
- 为 metric_name=queue 事件和 metric_name=pipeline 事件提供单独的日志到指标设置。
- 将 metric_name=queue 指标数据点中的 location 和 corp 字段列入黑名单。
- 将日志到指标设置和具有 metrics_log 来源类型的事件关联起来。

这些配置可能如下所示：

transforms.conf

```
[eval_pipeline]
INGEST_EVAL = metric_name=group

[metric-schema:extract_metrics]
METRIC-SCHEMA-MEASURES-queue = max_size_kb,current_size_kb,current_size,largest_size,smallest_size
METRIC-SCHEMA-BLACKLIST-DIMS-queue = location,corp
METRIC-SCHEMA-MEASURES-pipeline=cpu_seconds,executes,cumulative_hits
```

props.conf

```
[metrics_log]
TRANSFORMS-metricslog = eval_pipeline
METRICS-SCHEMA-TRANSFORMS = metric-schema:extract_metrics
```

以下示例为这些配置能够使 Splunk 平台从这些事件中生成的指标数据点：

_time	metric_name	_value	name	processor
08-05-2017 20:26:29.073 -0700	queue.max_size_kb	1024	aggqueue	
08-05-2017 20:26:29.073 -0700	queue.current_size_kb	1	aggqueue	
08-05-2017 20:26:29.073 -0700	queue.current_size	5	aggqueue	
08-05-2017 20:26:29.073 -0700	queue.largest_size	35	aggqueue	
08-05-2017 20:26:29.073 -0700	queue.smallest_size	0	aggqueue	
08-05-2017 20:26:29.075 -0700	pipeline.cpu_seconds	0	indexerpipe	indexin
08-05-2017 20:26:29.075 -0700	pipeline.executes	171	indexerpipe	indexin
08-05-2017 20:26:29.075 -0700	pipeline.cumulative_hits	2214401	indexerpipe	indexin

与指标结合使用

搜索和监视指标

要分析指标索引中的数据，请使用报表命令 `mstats`。您可以使用 `mstats` 应用指标聚合，以将来自不同数据来源的问题隔离并关联起来。请参阅*《搜索参考》*手册中的 mstats。

要枚举指标名称、维度和值，请使用内部搜索命令 `mcatalog`。请参阅*《搜索参考》*手册中的 mcatalog。

其他搜索命令不适用于指标索引。

请注意以下差别：

- 您无法为单独的指标事件搜索指标数据。
- 您不能将自动查找与指标数据结合使用。这是因为自动查找应用于单个事件，而指标将进行聚合分析。
- 您不能执行搜索-时间提取。
- 您可以用与自定义索引字段等效的字段丰富指标，这些字段被视为维度。
- 您可以将预留的字段，如 "source"、"sourcetype" 或 "host"，用作维度。但是，如果提取的维度名称为预留名称，则该名称应附前缀 "extracted_" 避免名称冲突。例如，如果维度名称为 "host"，请搜索 "extracted_host" 进行查找。
- 以连字符 (_) 开始的维度未建立索引，因此这些维度是不可搜索的。

搜索示例

要列出超出 10 秒间隔的指标名称计数：

```
| mstats count where metric_name=* span=10s BY metric_name
```

要对维度进行一次简单计数：

```
| mstats count where index=mymetricsdata metric_name=aws.ec2.CPUUtilization
```

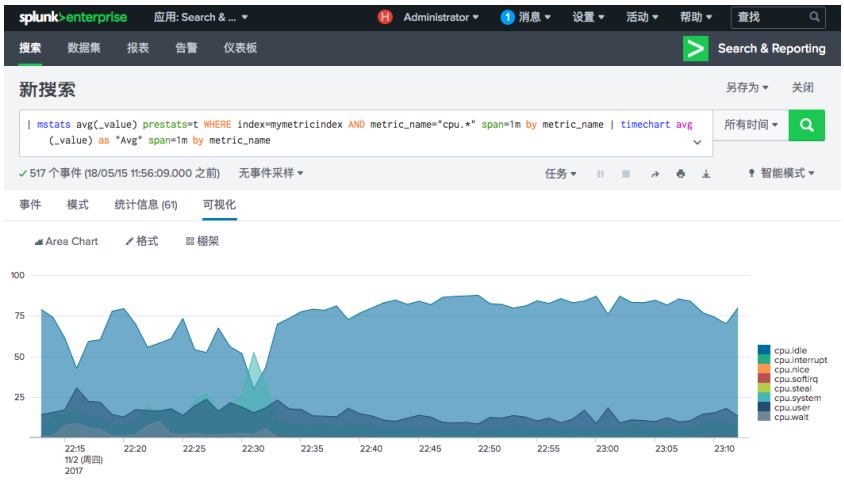
要计算每 30 秒间隔的测量平均值：

```
| mstats avg(_value) WHERE index=mymetricdata AND metric_name=aws.ec2.CPUUtilization span=30s
```

您还可以显示图表中的结果。以下示例使用了通配符搜索和分组依据：

```
| mstats avg(_value) prestats=t WHERE index=mymetricindex AND metric_name="cpu.*" span=1m by metric_name | timechart
avg(_value) as "Avg" span=1m by metric_name
```

此类搜索可用于堆叠不同的 CPU 指标，总计达 100%。



本搜索显示了使用 EVAL 语句的示例：

```
| mstats avg(_value) as "Avg" WHERE metric_name="memory.free.value" span=5s | eval mem_gb = Avg / 1024 / 1024 / 1024
| timechart max("mem_gb") span=5s
```

要列出所有指标索引中所有指标名称：

```
| mcatalog values(metric_name)
```

要列出所有指标索引中所有维度：

```
| mcatalog values(_dims)
```

使用 REST API 列出指标数据

您还可以使用 Metrics Catalog REST API 端点来枚举指标数据：

- 使用 `GET /services/catalog/metricstore/metrics` 端点列出指标名称。
- 使用 `GET /services/catalog/metricstore/dimensions` 端点列出维度名称。
- 使用 `GET /services/catalog/metricstore/dimensions/{dimension-name}/values` 端点列出给定维度值。

您还可以使用带这些端点的过滤器通过索引、维度和维度值限制结果。

请参阅*《REST API 参考手册》*中的“Metrics Catalog 端点描述”。

指标索引性能

本主题汇总了指标索引性能结果。

磁盘上的大小

用支持的指标来源类型（`collectd_http`、`statsd`、`metrics_csv`）引入典型的指标负载，指标索引所需磁盘存储空间比在事件索引中存储相同的负载要少占用约 50% 的空间。

吞吐量

在决定是否通过另行添加索引器来横向扩展时，请考虑以下事项。

使用带 HTTP 事件收集器 (HEC) 输入的 `collectd_http` 来源类型，测试最大获取吞吐量每秒稳定在 55,000 个事件左右，没有额外搜索负载的情况下，每秒大约是 58,000 个事件。

- 默认批处理大小为每批 5,000 个事件。在 100 到 5,000 个事件的不同批次大小之间没有观察到明显的获取性能差别。
- 为这些测试启用 `keep-alive` 设置。
- 典型的事件大小约为 214 字节。

使用带 UDP 输入的 `statsd` 来源类型，吞吐量会有很大不同，具体取决于其他网络活动。针对 UDP 输入，如果指标是 collected，我们建议尽量使用**通用转发器**。

速度

为运行指标查询考虑以下测试结果。此测试使用了来自 1,000 个主机的指标，指标索引中的事件总数为 60 亿个事件，其中查询具有代表性，且没有使用名称为 `metric_name` 的通配符。

时间范围	事件	查询速度
1 小时	3500 万	< 0.1s
1 天	8.5 亿	~3-5s
1 周	60 亿	~20-22s

请参阅*《容量规划手册》*。

指标的最佳实践

以下是在 Splunk 平台中和指标结合使用时的最佳方式：

基数问题

随着指定索引和数据桶中存储的指标时间序列基数的增加，指标搜索性能会降低。换句话说，随着指标数据中设置的唯一维度集的增加，指标搜索的速度会降低。以下策略可帮助您减少指标索引和数据桶中的时间序列基数。

- **删除数据中不必要的维度。**重点删除具有各种唯一值的维度，如用户 ID 或手机号码。
- **使用较大的数据桶大小。**这样可帮助您减少每个指标数据点的开销。例如，您可尝试将数据桶大小调整为 10GB。
- **跨多个索引器拆分指标数据。**执行此操作时，请按照相对搜索域对索引进行分区。将经常一起搜索的数据保留在同一索引中。例如，如果很少一起搜索 IT 基础设施指标数据和销售/营销指标，您可以将 IT 基础设施指标数据保留在一个索引中，将销售/营销指标保留在另一个索引中。

高结果行基数也会降低搜索性能。您可通过提高时间数据桶 `span` 减少返回的行数来尝试减少这种情况。您还可缩短搜索的整体时间范围。

具有维度扩展名的 StatsD 格式

如果你正在索引 StatsD 格式的数据，请使用具有维度扩展名的 StatsD 格式提高性能：`cpu.idle:0.5|g|#host:some-hostsplunk.com,app:some-app`

使用这种格式而不是将维度和指标名称结合在一起的纯 StatsD 格式：`cpu.idle.some-hostsplunk.com.some-app`

其他最佳实践

- 指标的 `_value` 字段应为 "Double" 类型而不是 "String" 类型，避免导致索引效率低下。
- 对于 Metrics Catalog 端点的 REST 调用的更快响应时间，请在适用的情况下使用限制时间窗口。默认情况下，仅搜索过去 24 小时的数据。请参阅*《REST API 参考手册》*中的“Metrics Catalog 端点描述”。
- 确保维度名称不以下划线 (_) 开头。这些维度不会被索引。