



Splunk[®] Enterprise 8.2.0

指标

生成时间：2021 年 5 月 24 日，14:33

Table of Contents

指标简介	3
指标概述	3
指标入门	4
指标数据导入	7
从 StatsD 导入指标	7
配置特殊 StatsD 输入自定义	8
通过 REST API 端点配置 StatsD 维度提取	11
从 collectd 导入指标	12
从其他来源导入指标	15
将日志数据转换为指标	20
将事件日志转换为指标数据点	20
在 Splunk Web 中设置引入时日志到指标的转换	21
使用配置文件设置引入时 log-to-metrics 转换	23
汇总指标数据	29
汇总指标数据以加快搜索性能并增加存储容量	29
使用 Splunk Web 新建和编辑指标汇总策略	31
通过 REST API 创建和维护指标汇总策略	32
使用配置文件管理指标汇总策略	33
与指标结合使用	36
将 Analytics Workspace 中指标可视化	36
搜索和监视指标	36
对指标时间序列执行统计计算	37
调查计数器指标	38
使用直方图指标	41
指标索引性能	44
指标的最佳实践	44

指标简介

指标概述

指标是一个适合于系统管理员和 IT 及服务工程师使用的功能，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。

在 Splunk 平台中，您使用指标索引来存储指标数据。该索引类型针对指标数据的存储和检索进行了优化。

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。您可以在这些指标索引的指标数据点上运行指标特定的命令，如 `mstats`、`mcatalog` 和 `msearch`。例如，`mstats` 命令允许您将聚合函数，如平均值、总和、计数和速率等应用于这些数据点，帮助您将来自不同数据来源的问题隔离并关联起来。

从 Splunk 平台版本 8.0.0 开始，指标索引和搜索区分大小写。因此，诸如 `mstats` 和 `msearch` 的指标搜索命令会将以下内容视为三个不同的指标：`cap.gear`、`CAP.GEAR` 和 `Cap.Gear`。

什么是指标数据点？

指标是某个具体时间点的单个测量值。如果您将该测量值与时间戳和一个或多个维度结合在一起，您会得到一个指标数据点。单个指标数据点只有一个时间戳，但可以包含多个测量值和多个维度。

- timestamp**
表示数据点中测量值的获取时间。
- metric_name**
您正在测量的东西。使用用圆点隔开的层次结构引用命名空间（例如，`spl.mlog.per_index_thruput.ev`）。您可以使用任何字符串作为指标名称。指标名称可以包含字母、数字、下划线、点和其他符号（保留术语 "metric_name" 除外）。指标名称使用点将其命名空间分成多个段。您可以借助点来创建指标的层次结构。
- numeric_value**
一个数字（整数或双精度浮点数），表示给定时间点的指标值，例如计数。
- measurement**
`metric_name` 和相应的 `numeric_value` 字段值结合。测量值始终遵循以下语法：`metric_name:<metric_name>=<numeric_value>`。
例如：`metric_name:cpu.idle=15` 或 `metric_name:io.util=10.232`。

维度
提供有关测量值的其他信息的元数据字段。维度提供您可以用于筛选指标数据点或对其进行分组的类别。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`

所有指标数据点均具有以下三个默认维度：`host`、`source` 和 `sourcetype`。Splunk 软件为指标数据点建立索引时会为其添加这几个维度。即使指标数据点没有其他任何维度，仍可以按这些默认维度对其进行过滤或分组。

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
- 软件即服务（SaaS）系统
- 传感器，例如物联网（IoT）功能

什么是指标时间序列？

指标时间序列是测量相同内容且具有相同维度集的指标数据点集。以下是三个来自指标时间系列的指标数据点。注意：每个指标数据点都有针对 `max.size.kb`、`current.size.kb` 和 `current.size` 指标的测量值，且他们共享相同的维度字段值组合。

_time	metric_name:max.size.kb	metric_name:current.size.kb	metric_name:current.size	组	name
08-05-2019 16:26:42.025 -0700	500	300	53	队列	azd
08-05-2019 16:26:41.055 -0700	345	245	43	队列	azd
08-05-2019 16:26:40.025 -0700	280	180	33	队列	azd

00-00-2017 16:26:40.023 -0700	334	124	39	队列	azd
-------------------------------------	-----	-----	----	----	-----

请参阅“对指标时间序列执行统计计算”，了解有关指标时间序列以及如何在 `mstats` 搜索中使用 `_timeseries` 字段的更多信息。

Splunk 平台为指标数据提供哪些功能？

Splunk 平台提供了一个全面的指标解决方案，从一端的指标数据摄取、索引和转换，到另一端的指标搜索、分析和报告。

指标数据导入

Splunk 平台利用代理和 API 的指标集合框架收集和引入大容量指标。支持行指标协议，如 `collectd` 和 `StatsD`。通用转发器和重型转发器可以使用这个集合框架引入指标数据并安全转发到独立的指标索引或指标索引群集。请参阅“指标数据导入”。

在建立索引时将事件数据转换为指标数据

指标引入管道符可以在索引时转换您的数据，这样符合结构化良好的指标协议。您还可以使用日志到指标功能在引入和索引事件数据时，将事件数据转换为指标数据。请参阅“将事件日志转换为指标数据点”。

在搜索时将事件数据转换为指标数据

使用 `mcollect` 和 `meventcollect` 命令可以在搜索时将事件数据搜索的结果或流事件转换为指标数据点。请参阅 `mcollect` 和 `meventcollect` 命令的相关主题。

搜索和报告指标数据

指标特定的 `mstats`、`msearch` 和 `mcatalog` 命令可用于筛选、聚合指标数据，并针对这些数据建立报表。请参阅“搜索和监视指标”。

可视化和分析指标趋势

`Analytics Workspace` 让您可以轻松监视和分析指标数据的趋势，而无需使用 Splunk 搜索处理语言（SPL）。使用它来创建交互式图表、可视化指标数据相关性并将新建另存为图表或仪表板。请参阅《*Analytics Workspace*》手册中的“关于 `Analytics Workspace`”。

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。

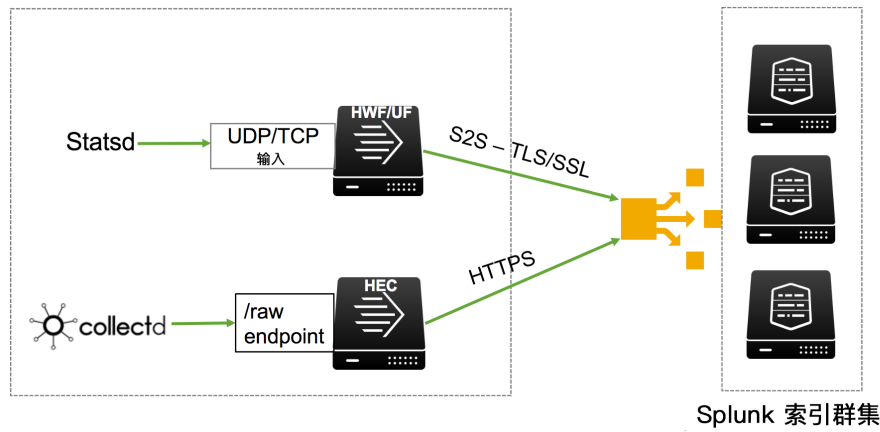
字段	必填	可写入或内部	描述	示例
<code>_time</code>	X	可写入	UNIX 时间表示法中的指标时间戳。	2017-08-14 17:12:39.000
<code>metric_name:<metric_name></code>	X	可写入	指标名称。始终有一个数值。这是 64 位浮点数，精确到小数点 15 到 17 位。	<code>metric_name:os.cpu.user=42.12345</code>
<code><dimension 0> ... <dimension n></code>	X	可写入	任意数量维度字段，表示如何拆分指标。	<code>ip</code>
<code>_dims</code>	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。	<code>_dims::ip</code>
<code>source</code>		内部	指标数据来源。	<code>udp:8125</code>
<code>host</code>	X	内部	原始主机。Splunk 软件中的标准字段。	<code>server007</code>
<code>index</code>	X	内部	指标索引名称。Splunk 软件中的标准字段。	<code>metricsindex</code>
<code>sourcetype</code>	X	内部	指标的数字结构。Splunk 软件中的标准字段。	<code>statsd</code>

如果 `metric_name` 字段为空或完全由空格组成，则 Splunk 平台无法索引包含这类字段的指标数据点。

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索引类型。与事件索引中的事件相比，指标索引存储指标数据点采用的格式可以提供更快的搜索性能和更有效的数据存储。

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。

您可以在默认的 `_metrics` 索引中监视内部 Splunk 指标。这是 `_internal` 事件索引的指标类似结构。

如果您使用的是 Splunk Enterprise，请参阅《管理索引器和索引器群集》手册中的“创建指标索引”。

如果您使用的是 Splunk Cloud，请参阅《Splunk Cloud 用户手册》中的“管理 Splunk Cloud 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指标数据和枚举项目，请使用 `mstats` 和 `mcatalog` 搜索命令。
- `msearch` 命令允许您查看单个指标数据点，无需聚合。
- `mcollect` 和 `meventcollect` 命令会在搜索时将事件日志数据转换为指标数据点。

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 有几种指标协议格式，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：

- 基本 StatsD 数据行指标协议，其包括 `metric_name`、`_value` 和 `metric_type`。
- 扩展 StatsD 数据行指标协议，添加样本率和维度。

Splunk 支持 StatsD 指标数据点的两个 `metric_type` 值：`g`，用于仪表指标，以及用于计数器指标的 `c`。

为了易于使用，默认情况下，Splunk 平台会将 StatsD 数据转换为单测量值指标数据点，其中每个指标数据点针对指标名称有一个键值对，针对指标测量值有另一个键值对。如果您需要 Splunk 软件将 StatsD 数据转换为支持每个数据点有多个指标测量值的指标数据点格式，请在 `props.conf` 中将 `STATSD_EMIT_SINGLE_MEASUREMENT_FORMAT=false` 添加到指标源类型的段落中。

有关更多信息，请参阅“创建特殊 StatsD 输入自定义”。

基本的 StatsD 指标协议

基本的 StatsD 数据行指标协议只有三个字段：`metric_name`、指标 `_value` 和 `metric_type`。

语法

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

示例指标

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

扩展 StatsD 指标协议

扩展 StatsD 数据行指标协议支持维度和样本率。样本率只适用于计数器指标，表示他们具有 `c` 的 `metric_type`。对于

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”。

语法

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

示例仪表指标

仪表是一个表示可以任意波动的单个数字值的指标。例如，您可以使用仪表表示当前运行搜索任务的数量，或者服务器房间内的温度。

```
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/sdal,fstype:ext3
```

示例计数器指标，经 Splunk 平台处理后

计数器指标计算事件的发生次数。其值只能增加或重设为零。例如，您可以使用计数器表示服务的请求、完成的任务或者错误的数量。有关计数器指标的更多信息，请参阅“调查计数器指标”。

以下为经过 Splunk 平台处理的计数器指标示例。

```
event.login:6|c|@0.5|#region:west,dc:west-1,ip:10.1.1.1,host:valis1.buttercupgames.com,app:zoolu
```

请注意：此计数器指标的样本率为 0.5。这意味着 StatsD 客户端仅在 50% 的时间内对此计数器指标进行采样。Splunk 平台通过将指标值乘以 1/0.5 或 2 来调整此值。这意味着从 StatsD 客户端发送的原始指标如下所示：

```
event.login:3|c|@0.5|#region:west,dc:west-1,ip:10.1.1.1,host:valis1.buttercupgames.com,app:zoolu
```

注意：原始指标事件有一个数字值 3。

关于样本率

当为特定计数器指标生成大量数据点时，Splunk 平台聚合这些数据点会非常占用资源。StatsD 客户端通过实施采样率来管理，以减少将其发送到 Splunk 平台的网络流量。

StatsD 客户端将 `sample_rate` 值放入计数器指标数据点，以向 Splunk 平台指示其采用的实际下采样百分比。Splunk 平台通过将下采样计数器指标的值乘以 $1/\text{sample_rate}$ 来响应此情况。

例如，假设您有名为 `event.login` 的计数器指标，`sample_rate` 为 0.1。这表示只有 10% 的 `event.login` 数据点从 StatsD 客户端传递到 Splunk 平台实施。Splunk 平台将 `event.login` 值乘以 $1/0.1$ 或 10 以调整丢失的数据点。因此，如果您的 Splunk 平台实施接收值为 2 的 `event.login` 数据点，那么该实施会将该值更改为 20。

Splunk 平台为不在 0 和 1 范围内的 `sample_rate` 值传递警告消息。`sample_rate` 的默认设置是 1。

使用其他 StatsD 格式

如果您使用 StatsD 实现，该实现使用来自 Splunk 平台本机支持维度的其他不同格式，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**。

配置特殊 StatsD 输入自定义

当使用 Splunk Web 界面设置新的 StatsD 指标数据导入时，如“从 StatsD 导入指标”中所述，您可能不需要对该导入进行任何其他配置。但是，有些 StatsD 输入用例要求您对本地的 `props.conf` 和 `transforms.conf` 文件进行特殊的手动自定义。

您需要创建或更新配置文件，以使 Splunk 部署能够：

- 将引入的 StatsD 指标数据转换为多测量值指标数据点。默认情况下，Splunk 软件会将引入的 StatsD 指标数据转换为单测量值指标数据点。
- 从 StatsD 数据的指标名称中提取维度。

前提条件 以下前提条件适用于本主题中介绍的所有操作过程。

- 只有具有文件系统访问权限的用户，如系统管理员，才能使用配置文件设计这些 StatsD 输入自定义。
- 请参阅 Splunk Enterprise 《*管理员手册*》中的“如何编辑配置文件”了解具体步骤。
- 您可以有几个具有相同名称的配置文件，分散在默认目录、本地目录和应用目录中。请参阅 Splunk Enterprise 《*管理员手册*》中“在何处放置（或查找）已修改的配置文件”。
- 请参阅《*分布式搜索*》中的“使用 `Deployer` 分布应用和配置更新”，了解具体操作。

不要更改或复制默认目录中的配置文件。默认目录中的文件必须保持原样并位于其原始位置。更改本地目录中的文件。

通过 StatsD 数据导入生成多测量值指标数据点

默认情况下，Splunk 软件会将 StatsD 指标数据转换为单测量值指标数据点。在单测量值数据点格式下，每个指标数据点针对指标名称有一个键值对，针对相应的指标测量值有另一个键值对。指标数据点中的其余字段为维度。

另一方面，多测量值指标数据点可以在指标数据点中有一个或多个指标测量值。每个测量值都遵循以下语法：metric_name:<metric_name>=<numeric_value>。指标数据点中的每个测量值都共享该指标数据点中的维度。

对于 StatsD 指标数据引入，首选单测量值指标数据点格式，因为很多 StatsD 客户端会将维度名称嵌入指标名称中。发生这种情况时，为单测量值指标数据点创建维度提取配置会更加容易。如果您决定让 StatsD 输入生成多测量值数据点，则必须了解，这会让通过 metric_name 进行的维度提取比其他方式更加困难。

有关使用 props.conf 和 transforms.conf 文件配置 StatsD 维度提取的更多信息，请参阅本主题中的“为不受支持的 StatsD 格式配置维度提取”。

如果您希望 StatsD 输入生成多测量值指标数据点，则需要将 props.conf 的 StatsD 输入源类型对应的段落中添加 STATSD_EMIT_SINGLE_MEASUREMENT_FORMAT = false。如果指标数据还没有源类型，则必须创建一个自定义源类型。

步骤

1. 通过打开您要使用的位置的 props.conf 配置文件开始定义自定义源类型，例如搜索和报表应用（\$SPLUNK_HOME/etc/apps/search/local/）或系统中的位置（\$SPLUNK_HOME/etc/system/local/）。如果此位置中不存在 props.conf 文件，请创建一个文本文件并将其以 props.conf 文件名保存到该位置。
2. 将段落添加到 props.conf 文件中，如下所示：

```
[<custom_metrics_source_type_name>]
METRICS_PROTOCOL = STATSD
STATSD_EMIT_SINGLE_MEASUREMENT_FORMAT = false
```

- custom_metrics_source_type_name：您的自定义指标来源类型名称。
 - METRICS_PROTOCOL：识别指标输入的传入指标数据所用的指标协议。这里选择了 STATSD，因为它是 StatsD 指标输入。
 - STATSD_EMIT_SINGLE_MEASUREMENT_FORMAT：控制 StatsD 处理器是生成单测量值还是多测量值指标数据点。将其设置为 false 以生成多测量值数据点。仅当 METRICS_PROTOCOL 设置为 STATSD 时，此设置才有效。
- 如果 props.conf 有更改的话，请将更改部署到索引器。

如果您在索引器前采用了重型转发器，则 props.conf 处理会发生在转发器上，而非索引器上。因此，您必须将 props.conf 更改部署到重型转发器中。

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

为不受支持的 StatsD 格式配置维度提取

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
```

Splunk 软件处理了该指标数据并对其进行字段提取之后，如果您使用的是单测量值指标数据点格式，则提取的指标名称和测量值应如下所示：

```
metric_name=cpu.percent _value=75
```

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。这需要两个过程：

- 在 transforms.conf 中，定义维度提取配置。
- 在 props.conf 中，为 StatsD 数据创建自定义源类型。

以下过程用于从单测量值指标数据点的 metric_name 字段中提取维度。它们不适用于设置为生成多测量值指标数据点的 StatsD 导入。

为 StatsD 指标数据定义维度提取配置的步骤

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，例如搜索和报表应用（\$SPLUNK_HOME/etc/apps/search/local/）或系统中的位置（\$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。
- 在 transforms.conf 文件中，为每个维度提取配置添加一个段落，如下所示：

```
[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。StatsD 维度提取配置的段落名称必须以 statsd-dims: 为前缀。
- 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。

设置为 true 时，Splunk 软件会从测量值中删除维度值，而不匹配的部分则成为 metric_name。默认设置为 true。设置为 false 时，此设置将提取的维度值保留在 metric_name 中。

例如，指标测量值名称为 x.y.z。正则表达式匹配 y 和 z。当 REMOVE_DIMS_FROM_METRIC_NAME 为 true 时，metric_name 为 x。当它为 false 时，metric_name 为 x.y.z。

- 将更改保存到 transforms.conf 文件中。
- 如果 props.conf 和 transforms.conf 有更改的话，请将更改部署到索引器。

如果您在索引器前采用了重型转发器，则 props.conf 和 transforms.conf 处理会发生在转发器上，而非索引器上。因此，您必须将 props.conf 更改部署到重型转发器中。

为 StatsD 指标数据定义自定义来源类型的步骤

- 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，例如搜索和报表应用（\$SPLUNK_HOME/etc/apps/search/local/）或系统中的位置（\$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。
- 将段落添加到 props.conf 文件中，如下所示：

```
[<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。
 - METRICS_PROTOCOL：识别指标输入的传入指标数据所用的指标协议。这里选择了 STATS，因为它是 StatsD 指标输入。
 - STATSD-DIM-TRANSFORMS：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，且如果相关 transforms.conf 段落名称和 metrics_sourcetype_name 一样，此 STATSD-DIM-TRANSFORMS 设置可忽略。
- 将更改保存到 props.conf 文件中。
 - 如[为 StatsD 数据设置数据导入](#)中所述，为此来源类型新建数据导入，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：

```
# 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
```

```
# props.conf.example
```

```
[my_custom_metrics_sourcetype]
METRICS_PROTOCOL = statsd
STATSD-DIM-TRANSFORMS = regex_stanza1, regex_stanza2
```

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：

```
# 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
```

```
# props.conf.example
```

```
[my_custom_metrics_sourcetype]
METRICS_PROTOCOL = statsd
```

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。

通过 REST API 端点配置 StatsD 维度提取

如果您使用 Splunk Cloud 或根本没有 Splunk 文件系统的访问权限，则可以通过手动调用 REST API 端点来为不受支持的 StatsD 格式配置维度提取。如果您的 StatsD 客户端在指标名称中嵌入了维度，则可能需要配置维度提取。

前提条件

- 阅读“为不受支持的 StatsD 格式配置维度提取”的第一部分，简要了解为什么可能需要进行维度提取。
- 请参阅《REST API 参考手册》中的以下主题，以更好地了解此过程中讨论的 Splunk REST API 系统和 REST 端点。
 - 使用 REST API 参考
 - `/data/transforms/statsdextractions`
 - `/admin/metrics-reload/_reload`
- 如果您使用的是 Splunk Cloud，请参阅《REST API 教程》中的“Splunk Cloud REST API 的访问要求和限制”。

步骤

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>:<port>/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&DATETIME_CONFIG=CURRENT&pulldown_type=true&category=Metrics"`
 - `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_type=true&category=Metrics"
```

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>:<port>/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>\$%2B?)\.(?<ip>\d{1,3}\.\d{1,3}\.\d{1,3}\.\d{1,3})&REMOVE_DIMS_FROM  
METRIC_NAME=true"
```

- 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
```

- 如 [为 StatsD 数据设置数据导入](#) 中所述，为此来源类型新建数据导入，然后选择自定义来源类型。

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：

- [配置 HTTP 事件收集器（HEC）数据导入](#)。
- [安装 collectd](#)。
- [配置 collectd](#)。
- [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。

- 在 Splunk Web 中单击 **设置 > 数据导入**。
- 在本地输入中，单击 **HTTP 事件收集器**。
- 确认 HEC 是否启用。
 - 单击 **全局设置**。
 - 对于 **所有标记**，如果尚未选择 **启用** 按钮，则单击此按钮。
 - 注意 **HTTP 端口号** 值，您需要用此值来配置 collectd。
 - 单击 **保存**。
- 单击 **新标记配置** HEC 标记发送数据。
- 在 **选择数据来源** 页面，请针对名称输入标记名称，如 "collectd token"。
- 请勿选择其他选项。
- 单击 **下一步**。
- 在 **输入设置** 页面，单击 **选择来源类型**。
- 单击 **选择来源类型**，然后选择 **指标 > collectd_http**。
- 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：

 - 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。
 - 请单击 **指标选择索引数据类型**。
 - 需要时，配置其他索引属性。
 - 单击 **保存**。
- 单击 **查看**，然后单击 **提交**。
- 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, "host": "collectd", "plugin": "protocols", "plugin_instance": "IpExt", "type": "protocol_counter", "type_instance": "InOctets"}]}'
```

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"
```

关于使用 HEC 的更多信息，请参阅《导入数据》中的以下主题：

- 在 Splunk Web 中设置并使用 HTTP 事件收集器
- 为 HTTP 事件收集器设置事件格式
- 将指标发送到指标索引

请参阅《搜索参考》手册中的 mstats 和 mcatalog。

请参阅《REST API 参考手册》中的以下主题：

- Metrics Catalog 端点描述
- /collector
- /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	示例
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://10.66.104.127:8088/services/collector/raw"
标头	添加到请求的 HTTP 标头。	标头 "Authorization: Splunk <HEC_token>"	标头 "Authorization: Splunk b0221cd8-c4b4-465a-9a3c-273e3a75aa29"

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	<pre>LoadPlugin cpu <Plugin cpu> ReportByCpu true </Plugin></pre>
界面	<pre>LoadPlugin interface</pre> <p>使用默认配置。</p>
负载	<pre>LoadPlugin load <Plugin load> ReportRelative true </Plugin></pre>
日志文件	<pre>LoadPlugin logfile <Plugin logfile> LogLevel info File STDOUT Timestamp true PrintSeverity false </Plugin></pre>
内存	<pre>LoadPlugin memory <Plugin memory> ValuesAbsolute true ValuesPercentage true </Plugin></pre>
网络	<pre>LoadPlugin network</pre> <p>仅当 <code>collectd</code> 客户端和 <code>connectd</code> 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。</p>
syslog	<pre>LoadPlugin syslog</pre> <p>使用默认配置。</p>
write_http	<p>您需要来自 HEC 数据导入的值配置此插件。LoadPlugin write_http</p> <pre><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></pre>

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

当您使用 CSV 文件作为指标数据的输入时，有两种可接受的 CSV 文件格式。您选择哪种格式取决于您想要 Splunk 软件如何索引 CSV 文件中的信息。让索引使每个数据点就有多个测量值，还是使各数据点只有一个测量值？

使用可以包含多个测量值的指标数据点更有效。当您通过这种方式索引指标数据时，您可以降低数据存储成本，可以从提高搜索性能中获益。

设置指标 CSV 来源类型和数据输入

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。它可以处理两种 CSV 指标格式。

创建数据输入以将您的 CSV 数据添加到指标索引中。输入使用预置的 `metrics_csv` 来源类型。数据输入应：

- 来源类型：指标 > `metrics_csv`
- 索引：指标索引

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

```
#inputs.conf

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

您还应在指标索引器上有以下 `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
```

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”。

为多测量值指标数据点设置 CSV 文件格式

当您为多测量值指标数据点设置 CSV 文件格式时，第一列标题是 `_time`，指标时间戳。这是必填字段。

接下来是每个指标测量值的一个或多个列标题。各测量列标题遵循以下语法：`metric_name:<metric_name>`。

Splunk 软件将非时间戳或测量值的其他列视为维度。

CSV 表每行都是单独的指标数据点。

字段名称	必填	描述	示例
_time	是	指标时间戳。格式是 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:<metric_name>	是	特定指标的测量值，由 <metric_name> 指定，如 metric_name:os.cpu.idle 或 metric_name:max.size.kpbs。这些值始终是数值。	13.34
维度	否	所有其他字段均按维度处理。	对于名为 ip 的维度，值是 192.0.2.1。

这是一个针对多测量值指标数据点设置格式的 CSV 文件示例。第一列是指标时间戳 _time。中间三列是测量值。最后两列是维度。

```
"_time","metric_name:cpu.usr","metric_name:cpu.sys","metric_name:cpu.idle","dc","host"
"1562020701",11.12,12.23,13.34,"east","east.splunk.com"
"1562020702",21.12,22.33,23.34,"west","west.splunk.com"
```

此 CSV 文件示例包含以下部分中单测量值指标数据点的示例 CSV 文件相同的信息。但是，因为为此信息使用了两个数据点而不是六个，索引时占用的空间更少。

为单测量值指标数据点设置 CSV 文件格式

当您为单测量值指标数据点设置 CSV 文件格式时，前三列是单测量值指标数据点的必填字段：

- metric_timestamp
- metric_name
- _value。

所有其他列均按维度处理。

在引入和索引过程中，metric_name 和 _value 测量值将合并为 metric_name:<metric_name>=<numeric_value> 格式。

字段名称	必填	描述	示例
metric_timestamp	是	时间戳格式是 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	是	使用圆点字符串表示法的指标名称。	os.cpu.percent
_value	是	数值与 metric_name 相关。	42.12345
维度	否	所有其他字段均按维度处理。	ip

这是一个针对单测量值指标数据点设置格式的 CSV 文件示例。表格的前三列是单测量值指标数据点的必填字段。所有其他列均是维度。此 CSV 文件的 dc 和 host 作为维度。

```
"metric_timestamp","metric_name","_value","dc","host"
"1562020701","cpu.usr",11.12,"east","east.splunk.com"
"1562020701","cpu.sys",12.23,"east","east.splunk.com"
"1562020701","cpu.idle",13.34,"east","east.splunk.com"
"1562020702","cpu.usr",21.12,"west","west.splunk.com"
"1562020702","cpu.sys",22.33,"west","west.splunk.com"
"1562020702","cpu.idle",23.34,"west","west.splunk.com"
```

如果将此示例与多测量值指标数据点的示例进行比较，您可以看到单指标格式将如何占用磁盘上的更多空间。此表包含了和多测量值表格相同的信息。但是，此表使用六个数据点，而多测量值表格只使用了两个。

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>*：类时唯一的文字字符串，识别要提取的字段的名空间。*transform_stanza_name* 是表示如何提取字段的 transforms.conf 中段落名称。

• 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，为此来源类型新建数据导入，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《数据导入》中的“在 Splunk Web 中设置并使用 HTTP 事件收集器”。

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《数据导入》中的“在 Splunk Web 中设置并使用 HTTP 事件收集器”和“为 HTTP 事件收集器设置事件格式”。

有关 /collector 端点参考，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数据点到指标索引的命令，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_1.splunk.com","fields":{"region":"us-west -
1","datacenter":"dc1","rack":"63","os":"Ubuntu16.10","arch":"x64","team":"LON","service":"6","service
_version":"0","service_environment":"test","path":"/dev/sda1","fstype":"ext3","metric_name:cpu.usr":
11.12,"metric_name:cpu.sys": 12.23, "metric_name:cpu.idle": 13.34}}'
```

此指标数据点的测量值显示在 JSON blob 的结尾处。遵循使用 "metric_name:<metric_name>:<numeric_value>" 语法的多指标格式。

多指标 JSON 格式

8.0.0 之前的 Splunk 平台版本使用 JSON 格式，该格式一个 JSON 对象只支持一个指标测量值。这会产生一次只包含一个测量值的指标数据点。

Splunk 平台版本 8.0.0 支持 JSON 格式，允许各 JSON 对象包含多个指标的测量值。这些 JSON 对象会生成多测量值指标数据点。多测量值指标数据点占用的磁盘空间更小，可以提高搜索性能。

以下是多个指标格式的 JSON 对象。

```
{
  "time": 1486683865,
  "source": "metrics",
  "sourcetype": "perflog",
  "host": "host_1.splunk.com",
  "fields": {
    "region": "us-west-1",
    "datacenter": "dc2",
    "rack": "63",
    "os": "Ubuntu16.10",
    "arch": "x64",
    "team": "LON",
    "service": "6",
    "service_version": "0",
    "service_environment": "test",
    "path": "/dev/sda1",
    "fstype": "ext3",
    "metric_name:cpu.usr": 11.12,
    "metric_name:cpu.sys": 12.23,
    "metric_name:cpu.idle": 13.34
  }
}
```

将日志数据转换为指标

将事件日志转换为指标数据点

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。当您把日志数据引入 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，并识别搜索头可以将其数字值转换为测量值的字段。

您可选择识别已提取字段以便 Splunk 平台排除这些字段，这样这些字段就不会在指标数据点中显示。

搜索头将您尚未识别为测量或排除字段的已提取字段作为维度添加至指标数据点。

某些日志到指标功能扩展，如创建可自动将数字字段处理为测量的日志到指标配置，只能通过手动配置文件编辑或 REST API 操作管理。

将事件转换为指标数据点的好处

如果您发现从实际的角度出发将事件转换为指标数据点很有意义，那么您可能就会想这样做。与事件索引中的事件相比，指标索引存储指标数据点采用的格式可以提供更快的搜索性能和更有效的数据存储。

此外，如果您使用的是 Splunk Enterprise，则从日志到指标的转换可能会在许可证配额方面带来一些好处。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”。

将事件转换为具有多个测量值的指标数据点

这里有两种包含指标数据的日志事件。这些事件都有 `internaldata` 来源类型。

_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
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

在您设置日志到指标配置之后，Splunk 平台会运行从具有 `internaldata` 来源类型的事件中提取字段-值对的程序。会将数字字段转换为遵循以下语法的测量值字段：`metric_name:<metric name>=<value>`。会将剩余的字段（`group` 和 `name`）视为维度。

_time	组	name	metric_name:max_size_kb	metric_name:current_size_kb	metric_name:current_s
08-05-2017 20:26:29.073 - 0700	队列	aeq	500	300	53
08-05-2017 20:26:29.075 - 0700	队列	indexqueue	500	200	55

日志到指标的指标数据点的分析

各指标数据点包含 `_time` 字段和一个或多个测量值字段。指标数据点也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	源值
_time	08-05-2017 20:26:29.075 - 0700	使用初始事件中的 <code>_time</code> 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 <code>_time</code> 值。
测量值字段	metric_name:current_size=53	通过下列语法将带有数字值的字段转换为测量值字段： <code>metric_name:<metric_name>=<numeric_value></code> 。

metric_name	current_size	使用为测量值提供 numeric_value 的字段名称。在此情况下，测量值基于 current_size=53。
numeric_value	53	使用测量值依据的数字字段的值。在此情况下，测量值基于 current_size=53。
维度字段	group=queue, name=indexqueue	日志事件中的不能识别为测量字段或排除字段的任何字段（除 _time 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。

如果 metric_name 字段为空或完全由空格组成，则 Splunk 平台无法索引包含这类字段的指标数据点。

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”。

永远不会转换为指标测量值的数字字段

某些数字字段名称是保留的。Splunk 软件无法将带有这些名称的索引字段转换为指标测量值。如果您将这些名称用于索引的测量值字段，则应安排在它们进行日志到指标处理之前，对它们进行重命名。这种重命名需要更改您的 transforms.conf 配置。

以下是保留的字段名称的列表：

- _event_status
- _indextime
- _subsecond
- _value
- date_hour
- date_mday
- date_minute
- date_month
- date_second
- date_wday
- date_year
- date_zone
- linecount
- timeendpos
- timestartpos
- metric_timestamp
- punct
- time
- timestamp

== 日志到指标许可费用

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	为特定指标提供数值的字段。单个的指标数据点可以包含多个测量值。
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排除字段的所有字段作为维度。单个的指标数据点可以包含多个维度。
排除字段	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排除。

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 识别为测量值，将 group 和 division 识别为排除字段，Splunk 平台将产生一个指标数据点，将 metric_name:max_kb 和 metric_name:min_kb 作为测量值，将 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“将事件日志转换为指标数据点”
- 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，如果您正在使用结构化的 CSV- 或 JSON- 格式的数据，则选择 csv 或 json（如适用）。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。

如果选择 field_extraction，则 Splunk 软件会自动将 WRITE_META=true 添加到 transforms.conf 段落中以进行字段提取。请参阅《数据导入》中的“Splunk 如何构建索引字段”。

7. （可选）必要时，更改事件执行、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	<p>从与所选来源类型关联的事件数据中输入一个或多个数值测量值字段名称，以逗号隔开。Splunk 会用 metric_name:<metric_name>::<numeric_value> 语法将各列出的字段转换为测量值，然后将这些测量值放入已完成的指标数据点。</p> <p>您可以使用通配符 (*) 匹配事件数据中的多个数值测量值字段。例如，如果您的事件包含 max_size_kb、min_size_kb 和 current_size_kb，您可以将 *_size_kb 包含在维度字段名称集中。这样会将三个字段都添加到测量值集中。</p> <p>或者，如果您想要 Splunk 软件将事件数据中的所有数值字段视为测量值，只需在测量字段中输入 _ALLNUMS_。</p> <p>如果您想要 Splunk 软件将事件数据中的所有数值字段视为测量值，在测量字段中输入 NUMS_EXCEPT_。之后是空格，然后是一个逗号分隔的数字字段列表，这些字段来自您不希望作为指标提取的事件数据。这些字段将被提取为维度。</p>
白名单	是	<p>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包含的维度字段名称，用逗号隔开。排除所有其他维度字段。如果事件数据中的大多数字段是基数很高的字段或者不是您的指标必需的，您可能想要设置包含字段的小列表。</p> <p>使用通配符 (*) 匹配事件数据中的多个维度字段值。例如，如果您的事件数据包含 customer_id、employee_id 和</p>

		consultant_id 作为维度，而且您想要包含所有这些维度，则可以将 <code>*_id</code> 添加到维度字段名称集中。这会将所有三个维度都添加到包含列表中。
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排除的维度字段名称，用逗号隔开。包含所有其他维度字段。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排除。 使用通配符 (*) 匹配事件数据中的多个维度字段值。

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	<p>查看文本框中的条目，必要时更新。可以包含用逗号隔开的数值测量字段列表，这些字段来自和所选来源类型匹配的事件数据。Splunk 会用 <code>metric_name:<metric_name>::<numeric_value></code> 语法将各列出的字段转换为测量值，然后将这些测量值放入已完成的指标数据点。</p> <p>您可以使用通配符 (*) 匹配事件数据中的多个数值测量值字段。例如，如果您的事件包含 <code>max_size_kb</code>、<code>min_size_kb</code> 和 <code>current_size_kb</code>，您可以将 <code>*_size_kb</code> 包含在维度字段名称集中。这样会将三个字段都添加到测量值集中。</p> <p>可以只包含术语 <code>_ALLNUMS_</code>。这会告诉 Splunk 平台将您事件数据中的所有数值字段转换为测量值。</p> <p>或者可以包含术语 <code>_NUMS_EXCEPT_</code>，后面是一个空格，以及用逗号隔开的数值测量字段列表。这会告诉 Splunk 平台将您事件数据中的所有数值字段转换为测量值，列出的字段除外，列出的字段会另行提取为维度。</p>
白名单	<p>此文本框可包含用逗号隔开的您特别想要包含的维度字段列表，这些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。不在此列表中的所有维度字段均被排除。如果事件数据中的大多数数字段是基数很高的字段或者不是您的指标必需的，您可能想要设置包含列表。然后，您可以只保留对您重要的那些字段，而忽略其余字段。</p> <p>使用通配符 (*) 匹配事件数据中的多个维度字段值。例如，如果您的事件数据包含维度 <code>customer_id</code>、<code>employee_id</code> 和 <code>consultant_id</code>，并且黑名单文本框中设置了 <code>*_id</code>，则从日志到指标转换生成的指标数据点中仅包含这三个维度。</p>
黑名单	<p>此文本框可包含用逗号隔开的您特别想要排除的维度字段列表，这些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。不在此列表中的所有维度字段都包括在内。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排除。</p> <p>使用通配符 (*) 匹配事件数据中的多个维度字段值。</p>

5. 单击下一步继续数据导入的“添加数据”工作流。

使用配置文件设置引入时 log-to-metrics 转换

如果您有部署的 `props.conf` 和 `transforms.conf` 文件的访问权限，您可以手动配置 log-to-metrics 转换，这比使用 Splunk

Web 设置更为复杂。例如，您可以设计可处理日志的 log-to-metrics 转换，其中并非所有事件都具有相同的指标和维度字段集。

要配置 log-to-metrics 转换，您需要将段落添加到 props.conf 和 transforms.conf 文件。

1. 在 transforms.conf 中指定 log-to-metrics 转换的方案。
2. 在 props.conf 中配置 log-to-metrics 设置。

关于引用时将日志转换为指标数据点的概述，请参阅“将事件日志转换为指标数据点”。

Splunk Web 中尚未提供日志到指标功能扩展

当您通过直接编辑配置文件管理日志到指标处理时，您可以利用 Splunk Web 中尚未提供的可选功能扩展。

扩展功能	在 Splunk Web 中	通过配置文件编辑	设置
根据共享字段的值对事件进行排序，并将不同的日志到指标转换规则对应到各事件组。	日志到指标转换规则应用于属于所选来源类型的所有事件。	如果属于所选的来源类型的所有的事件都共享 metric_name 字段，您可以为具有该字段特定值的事件设计日志到指标规则。	METRIC-SCHEMA-MEASURES- <unique_metric_name_prefix>、METRIC-SCHEMA-BLACKLIST-DIMS-<unique_metric_name_prefix> 和 METRIC-SCHEMA-WHITELIST-DIMS-<unique_metric_name_prefix>。

转发器注意事项

处理 log-to-metrics 转换时，您正在使用的转发器类型以及正在引入的数据类型需要具有 log-to-metrics 配置的 transforms.conf 和 props.conf 文件的特定索引器版本和位置。

以下配置适用，无论涉及的数据是结构化还是非结构化。结构化数据包括类似 CSV 和 JSON 的格式。有关更多信息，请参阅“为日志数据来源设置字段提取”。

数据类型	转发器版本和类型	所需的索引器版本	Log-to-metrics 配置文件的位置
结构化	7. 2. 4 或更高版本（但低于 8. x）的通用转发器	7. x 或更高版本	通用转发器
非结构化	任何低于 8. x 的通用转发器版本	7. 3. x 或更高版本	索引器
结构化	8. x 通用转发器	8. x	通用转发器
非结构化	8. x 通用转发器	8. x	索引器
结构化	7. 3. x 重型转发器	7. x 或更高版本	重型转发器
非结构化	7. 3. x 重型转发器	7. x 或更高版本	重型转发器
结构化	8. x 重型转发器	8. x	重型转发器
非结构化	8. x 重型转发器	8. x	重型转发器

在 transforms.conf 中指定 log-to-metrics 转换的方案

使用 transforms.conf 文件中的配置识别日志中哪些事件包含您想要提取的指标数据点，然后指定从该日志事件中提取指标的方式。

1. 识别日志中的哪些事件包含您想要提取的指标数据点，然后将在配置中应用相关设置。
 - 将 log-to-metrics 设置应用于日志中的所有事件。
 - 将 log-to-metrics 设置应用于日志中的特定事件。
2. 指定从日志事件中提取测量值的方式。
3. 确定将哪些事件字段转换为指标维度。

Log-to-metrics 指标方案设置参考

指标方案设置决定与段落相关的日志事件如何转换为指标数据点。此表介绍在 transforms.conf 中配置 log-to-metrics 时适用于可用设置的语法：

指标方案设置语法	描述	是否必
----------	----	-----

		需?
METRIC-SCHEMA-MEASURES = (_ALLNUMS_ (_NUMS_EXCEPT_)? <field1>, <field2>,...)	识别如何将事件中将数字字段提取为与这些事件对应的指标数据点中的测量。您可以识别应转换为测量的数字字段，或者您可以将事件中的所有数字字段处理为测量。	是
METRIC-SCHEMA-BLACKLIST-DIMS = <dimension_field1>, <dimension_field2>,...	识别排除的维度字段。这些字段在您不能在指标数据点（这些数据点从与 [metric-schema] 段落相关的事件生成）中显示为维度的事件数据中。如果事件数据中的一些字段是基数很高且不是指标集合必需的维度字段，您可能想要设置黑名单。	否
METRIC-SCHEMA-WHITELIST-DIMS = <dimension_field1>,<dimension_field2>,...	识别白名单维度字段。这些字段在您必需在指标数据点（这些数据点从与 [metric-schema] 段落相关的事件生成）中显示为维度的事件数据中。如果事件数据中的大多数字段是基数很高的字段或者不是您的指标必需的，您可能想要设置白名单。	否

将 *log-to-metrics* 设置应用于日志中的所有事件

使用 METRIC-SCHEMA-MEASURES 设置将日志到指标处理应用于日志中的所有事件。您可以选择使用 METRIC-SCHEMA-BLACKLIST-DIMS 和 METRIC-SCHEMA-WHITELIST-DIMS 设置从结果指标数据点中筛选不必要的维度字段。

此配置语法如下所示：

```
[metric-schema:<unique_transforms_stanza_name>
METRIC-SCHEMA-MEASURES = ( _ALLNUMS_ | ( _NUMS_EXCEPT_ )? <field1>, <field2>,... )
METRIC-SCHEMA-BLACKLIST-DIMS = <dimension_field1>, <dimension_field2>,...
METRIC-SCHEMA-WHITELIST-DIMS = <dimension_field1>,<dimension_field2>,...
```

使用您从“指定从日志事件中提取指标的方式”中选择的特定设置替换 (_ALLNUMS_ | (_NUMS_EXCEPT_)? <field1>, <field2>,...)。

将 *log-to-metrics* 设置应用于日志中的特定事件

使用 METRIC-SCHEMA-MEASURES-<unique_metric_name_prefix> 设置将日志到指标处理应用于日志中的特定事件。它将根据日志中所有事件共享的字段值锁定事件的一个子集。

选择使用 METRIC-SCHEMA-BLACKLIST-DIMS-<unique_metric_name_prefix> 和 METRIC-SCHEMA-WHITELIST-DIMS-<unique_metric_name_prefix> 参数从结果指标数据点中筛选不必要的维度字段。

此配置语法如下所示：

```
[metric-schema:<unique_transforms_stanza_name>]
METRIC-SCHEMA-MEASURES-<unique_metric_name_prefix> = ( _ALLNUMS_ | ( _NUMS_EXCEPT_ )? <field1>, <field2>,... )
METRIC-SCHEMA-BLACKLIST-DIMS-<unique_metric_name_prefix> = <dimension_field1>, <dimension_field2>,...
METRIC-SCHEMA-WHITELIST-DIMS-<unique_metric_name_prefix> = <dimension_field1>,<dimension_field2>,...
```

使用您从“指定从日志事件中提取指标的方式”中选择的特定设置替换 (_ALLNUMS_ | (_NUMS_EXCEPT_)? <field1>, <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 字段的值必须和 metric-schema 段落中表示的不同事件类型对应。

如果日志事件尚未共享 metric_name 字段，可通过以下几种方式将其添加到事件中：

- 新建命名为 metric_name 的索引时字段提取。
- 使用 INGEST_EVAL 设置在引入时将 metric_name 字段添加到事件。关于说明如何配置的示例，请参阅“针对性的 log-to-metrics 转换示例”。

如果配置正确，METRIC-SCHEMA-MEASURES-<unique_metric_name_prefix> 设置可生成带遵循以下语法的参数的指标数据点：metric_name:<unique_metric_name_prefix>.<measure_field_name>=<numeric_measure_field_value>。

始终将 METRIC-SCHEMA-BLACKLIST-DIMS-<unique_metric_name_prefix> 和 METRIC-SCHEMA-WHITELIST-DIMS-<unique_metric_name_prefix> 设置和相应的 METRIC-SCHEMA-MEASURES-<unique_metric_name_prefix> 设置结合使用。

指定从日志事件中提取测量值的方式

有几种选项可以从日志事件中提取测量值：

- 您可以提取事件中的所有指标字段作为测量值。

- 您可以使用一些排除项提取数字字段作为测量值，或排除特定字段使其不会被提取为测量值。
- 您可以提取特定的字段作为测量值，或者将提取作为测量值的特定字段列入白名单。

无论是为日志中的所有事件应用 `log-to-metrics` 设置，还是仅对日志中的特定事件应用这些设置，都可以使用这些选项。

提取测量值的方法	描述	语法示例	带数字和非数字值的字段
提取所有数字字段作为测量值。	使用 <code>_ALLNUMS_</code> 设置来设定 <code>[metric-schema]</code> 段落。	<code>[metric-schema:<unique_transforms_stanza_name>] METRIC-SCHEMA-MEASURES = _ALLNUMS_</code>	<code>_ALLNUMS_</code> 设置会提取数字值作为字段的测量值。由于是非数字值，因此相同的字段还可用作维度字段。如果您想要该字段仅用作测量值，将其排除在维度字段之外。请参阅“确定将哪些事件字段转换为指标维度”。
用一些排除项提取数字字段作为测量值。	使用 <code>_NUMS_EXCEPT_</code> 设置来设定 <code>[metric-schema]</code> 段落，以定义您不希望提取为测量值的字段的黑名单。 <code>_NUMS_EXCEPT_</code> 和函数设置的字段名称之间必须有空格。	<code>[metric-schema:<unique_transforms_stanza_name>] METRIC-SCHEMA-MEASURES = _NUMS_EXCEPT_ <measure_field1>, <measure_field2>,...</code>	<code>_NUMS_EXCEPT_</code> 设置会提取数字值作为字段的测量值。如果您想要同时拥有数字字段和非数字字段的字段仅作为维度字段，使用 <code>_NUMS_EXCEPT_</code> 设置将其排除在外，不得提取为测量值。
提取特定的字段作为测量值。	在 <code>transforms.conf</code> 中，设置 <code>[metric-schema]</code> 段落，该段落标识包含测量值的字段列表，以仅将这些字段提取为测量值。	<code>[metric-schema:<unique_transforms_stanza_name>] METRIC-SCHEMA-MEASURES = <measure_field1>, <measure_field2>,...</code>	如果您使用此设置指定同时具有数字和非数字值的字段，那么会将数字值提取作为测量值，而忽略非数字值。该字段不会用作具有非数字值的维度字段。

您可以使用通配符匹配数据中的多个类似数字字段。例如，假设您的事件数据包含以下数字字段，`max_size_kb`、`min_size_kb` 和 `current_size_kb`。您可以设置 `*_size_kb` 的 `<measure_field>` 值，将测量列表中的三个数字字段都包含在内，无需单独列出各字段。

确定将哪些事件字段转换为指标维度

`METRIC-SCHEMA-MEASURES` 设置没有表示为测量的任何事件字段可以显示为事件生成的指标数据点中的维度。

但是，你可以选择使用 `METRIC-SCHEMA-BLACKLIST-DIMS` 和 `METRIC-SCHEMA-WHITELIST-DIMS` 设置筛选指标数据点中的维度。

- 如果您为 `METRIC-SCHEMA-BLACKLIST-DIMS` 提供事件字段列表，搜索头会将所有未列出的非测量字段转换为指标数据点维度。
- 如果您为 `METRIC-SCHEMA-WHITELIST-DIMS` 提供事件字段列表，该列表中的字段只是搜索头将其转换为指标数据点维度的字段。这会忽略所有其他非测量字段

此配置语法如下所示：

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

当指标方案具有为 `METRIC-SCHEMA-BLACKLIST-DIMS` 和 `METRIC-SCHEMA-WHITELIST-DIMS` 定义的字段时，搜索处理器使用以下评估逻辑：

- 如果维度列在 `BLACKLIST` 中，即使也会显示在 `WHITELIST` 中，它不会显示在结果指标数据点中。
- 如果维度没有列在 `WHITELIST` 中，即使也不会显示在 `BLACKLIST` 中，它不会显示在结果指标数据点中。

您可以使用通配符匹配数据中的多个类似维度字段。例如，假设您的事件数据包含以下维度，`customer_id`、`employee_id` 和 `consultant_id`。您可以设置 `*_id` 的 `<dimension_name>` 值，将维度字段列表中的三个维度都包含在内，无需单独列出。

如果您的字段同时包含数字值和非数字值，您正在使用 `_ALLNUMS_` 设置或 `_NUMS_EXCEPT_` 设置，通过设置将字段提取为测量值，并因非数字值的缘故重新将其提取为维度。如果您想要该字段仅用作测量值，将其排除在维度字段之外。

关于从非结构化数据中提取新字段或重命名字段

“日志到指标”功能的原理是：检查 meta 中的索引字段并将这些字段转换为指标测量值，或将其作为维度或排除在维度之外。如果您使用结构化数据索引字段提取方式（例如 CSV 或 JSON 提取），则除了在此提供的说明之外，您无需进行其他设置。但是，如果要设计一个更复杂的配置，而且此配置涉及从非结构化数据中提取字段或重命名现有字段，则需要确保这些字段已建立索引。要确保这一点，您可以在该提取的 transforms.conf 段落中设置 WRITE_META=true。

请参阅《数据导入》中的“Splunk 如何构建索引字段”。

在 props.conf 中配置 log-to-metrics 设置

在 transforms.conf 为来源类型配置指标方案之后，在 props.conf 中完成 log-to-metrics 设置配置。在 props.conf 文件中配置 log-to-metrics 设置：

- 1. 参考 transforms.conf 中的指标方案。
- 2. 为日志数据来源设置字段提取。

参考 transforms.conf 中的指标方案。

要将 log-to-metrics 方案和特定的日志来源类型关联，参考 props.conf 中日志来源类型段落中的 transforms.conf 配置。使用 METRIC-SCHEMA-TRANSFORMS 设置，该设置具有以下语法：

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

在 METRIC-SCHEMA-TRANSFORMS 设置的 <stanza_name> 部分中键入 log-to-metrics 转换段落的名称。

为日志数据来源设置字段提取

要使用 log-to-metrics 配置，您必须设计可从您的日志数据中提取字段的配置。您使用的配置取决于数据是否结构化。

如果您的日志输入时结构化的格式（如 CSV 文件或 JSON），将 INDEXED_EXTRactions 设置添加到 props.conf 段落中。请参阅数据导入中的“从结构化数据文件中提取文件”。

如果从技术层面来说，您的日志数据时非结构化的，但是被组织成可以轻松提取的字段-值对，将 TRANSFORMS-
<class>=field_extraction 添加到该段落。上述参考的是 transforms.conf 中的 [field_extraction] 段落，该段落默认包括在 Splunk 平台中。[field_extraction] 段落使用简单的正则表达式从日志数据中提取字段-值对。

Log-to-metrics 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。

针对性的 log-to-metrics 转换示例

当一个日志来源类型包含具有不同测量集和维度字段的多个事件方案时，使用针对性的 log-to-metrics 转换。以下事件集合示例包含两个事件方案。这些事件共享一个 group 字段，并且 group 的值识别两个事件方案。

_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	INFO 指标 - group=pipeline、name=indexerpipe、processor=index_thruput、cpu_seconds=0、executes=171、

20:26:29.075 -0700	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

检查这些事件后，您决定您需要定义 transforms.conf 和 props.conf 中的一组配置，执行以下任务：

- 设置 TRANSFORMS-<class>=field_extraction 在引入时从日志行中提取字段-值对。
- 在引入时使用 INGEST_EVAL 将 metric_name 添加到带 group 字段的每个事件中。新的 metric_name 字段获得与 group 字段相同的值。
- 为 metric_name=queue 事件和 metric_name=pipeline 事件提供单独的 log-to-metrics 设置。从 metric_name=queue 事件中提取所有的数字字段作为测量值。
- 排除 metric_name=queue 指标数据点的维度中的 group、location 和 corp 字段。排除 metric_name=pipeline 事件的维度中的 group 字段。
- 将日志到指标设置和具有 metrics_log 来源类型的事件关联起来。

这些配置如下所示：

transforms.conf

```
[eval_pipeline]
INGEST_EVAL = metric_name=group

[metric-schema:extract_metrics]
METRIC-SCHEMA-MEASURES-queue=_ALLNUMS_
METRIC-SCHEMA-BLACKLIST-DIMS-queue=group,location,corp
METRIC-SCHEMA-MEASURES-pipeline=cpu_seconds,executes,cumulative_hits
METRIC-SCHEMA-BLACKLIST-DIMS-pipeline=group
```

props.conf

```
[metrics_log]
TRANSFORMS-fieldvalue=field_extraction
TRANSFORMS-metricslog=eval_pipeline
METRIC-SCHEMA-TRANSFORMS=metric-schema:extract_metrics
```

这些配置新建的指标数据点如以下示例所示：

_time	测量	维度
08-05-2017 20:26:29.073 -0700	metric_name:queue.max_size_kb=0, metric_name:queue.current_size_kb=0, metric_name:queue.current_size=0, metric_name:queue.largest_size=0, metric_name:queue.smallest_size=0	name=udp_queue
08-05-2017 20:26:29.073 -0700	metric_name:queue.max_size_kb=1024, metric_name:queue.current_size_kb=1, metric_name:queue.current_size=5, metric_name:queue.largest_size=35, metric_name:queue.smallest_size=0	name=aggqueue
08-05-2017 20:26:29.073 -0700	metric_name:queue.max_size_kb=500, metric_name:queue.current_size_kb=0, metric_name:queue.current_size=0, metric_name:queue.largest_size=1, metric_name:queue.smallest_size=0	name=auditqueue
08-05-2017 20:26:29.075 -0700	metric_name:pipeline.cpu_seconds=0, metric_name:pipeline.executes=171, metric_name:pipeline.cumulative_hits=2214401	name=indexerpipe, processor=indexin
08-05-2017 20:26:29.075 -0700	metric_name:pipeline.cpu_seconds=0, metric_name:pipeline.executes=171, metric_name:pipeline.cumulative_hits=2214401,	name=indexerpipe, processor=index_thruput
08-05-2017 20:26:29.075 -0700	metric_name:pipeline.cpu_seconds=0, metric_name:pipeline.executes=171, metric_name:pipeline.cumulative_hits=2214401,	name=indexerpipe, processor=indexandforward

汇总指标数据

汇总指标数据以加快搜索性能并增加存储容量

如果您有快速为大量的唯一指标数据点建立索引的高容量指标，您可能会关心历史指标数据存储容量以及跨大数据集搜索的低性能等问题。

指标汇总策略可以帮您解决这些问题。您可将指标汇总策略应用于具有大容量指标的指标索引。指标汇总策略会在这些索引上为指标的聚合和汇总设置规则。生成的**指标汇总摘要**是在一个或多个目标指标索引中新建的。汇总摘要包含指标数据点，这些数据点是来源索引中原始指标数据点的聚合。摘要指标占据磁盘空间更小，搜索速度比原始指标更快。

您可以通过 Splunk Web 新建指标汇总策略，方法是在 `metric_rollups.conf` 中添加或更新配置，以及使用 `catalog/metricstore/rollup` REST API 端点。

某些指标汇总功能扩展，如为汇总策略定义多个默认聚合函数的功能，只能通过手动配置文件编辑或 REST API 操作管理。

请参阅以下主题：

- 使用 Splunk Web 新建和编辑指标汇总策略
- 通过 REST API 创建和维护指标汇总策略
- 使用配置文件管理指标汇总策略

指标汇总策略的索引前提条件

如果您想要定义指标汇总策略，您必须识别来源指标索引以及一个或多个目标指标索引。来源索引会保留您想指标汇总策略汇总的原始指标。目标索引就是存储汇总摘要的位置。

如果摘要有空间，您可以指定来源索引为目标索引。但是，将来源索引和目标索引放在同一个设备上可能会降低您从该功能获得更多数据存储优势的能力。

如果指标汇总策略的目标索引不存在，您必须新建索引。请参阅《*管理索引器和索引器群集*》中的“新建指标索引”。

使用具有分布式搜索的指标汇总摘要

填充汇总摘要的后台搜索在搜索头上操作。这意味着它们要求在搜索头上可以发现来源索引和目标索引。如果您使用分布式搜索，您的索引都在索引器层上，并且在搜索头上无法发现。

您可以通过在搜索头层上新建独立来源和目标索引来解决这个问题。只要独立索引具有和索引器层上的实际索引相同的名称，Splunk 软件会将您为独立索引新建的任何汇总策略应用于实际索引。

如果您使用分布式搜索，您还需要让搜索头上的独立索引将其摘要数据转发到索引器上的实际索引。您可以通过设置使用白名单筛选所有其他索引的搜索头上的通用转发器配置来操作。这会启用该索引将指标汇总摘要数据转发到索引器层上的实际目标指标索引。请参阅以下主题：

- 最佳做法：《*分布式搜索*》中的“将搜索头数据转发到索引器层”。
- 按转发数据中的目标索引筛选数据。

关于 `_metrics_rollup` 索引

`_metrics_rollup` 索引是一个内部索引，旨在供监视控制台使用。只有在启用了监视控制台以及在监视控制台应用上下文中配置的指标汇总策略时，数据才会流向该索引。

要了解如何使数据流向 `_metrics_rollup` 索引，请参阅《*监视 Splunk Enterprise*》中的“资源使用情况：CPU 使用率”，然后在“历史平均 CPU 使用率”仪表板中找到介绍相关内容的子章节。

指标汇总策略的剖析

如果您有包含大容量指标的来源指标索引，您可以为该指标新建指标汇总策略。来源指标索引必须在搜索头上可发现。请参阅“指标汇总策略的索引前提条件”。

指标汇总策略要求

指标汇总策略至少决定：

- 在来源索引中为原始指标新建多少汇总摘要。
- 摘要存储在哪些目标索引中。

- 为汇总摘要生成聚合指标数据点的计划搜索的期限。
- 用于原始指标摘要的默认聚合函数。

下表定义指标汇总策略的所需组件。

项目	描述
一个或多个汇总摘要定义	汇总摘要定义决定搜索头新建汇总摘要的位置和方式。
默认的聚合函数	这是搜索头用于其生成汇总摘要时聚合来源索引中的指标数据点的默认函数。如果您没有定义聚合函数或者您通过 Splunk Web 新建指标汇总策略，搜索头使用 avg 函数。其他符合条件的函数是 count、max、median、min、perc<int> 和 sum。

各汇总摘要定义会进一步分为两个部分：目标指标索引名称和时间跨度。

组件	描述
目标指标索引名称	这是将新建指标汇总摘要的索引。如果目标指标索引不存在，您必须新建一个。该索引必须在搜索头上可发现。请参阅“指标汇总策略的索引前提条件”。
时间跨度	这会设置生成指标汇总摘要的计划搜索的期限。期限必须用相对时间语法表示，如 1h 表示一小时，或 20m 表示二十分钟。如果您设置的时间跨度小于 60 秒，可能会遇到搜索并发问题。

指标汇总策略选项

指标汇总策略可以选择包括维度筛选器以及一个或多个特例规则。下表介绍这些可选组件。

项目	描述
维度筛选器	您可以指示一组必须按策略生成的摘要中的汇总指标必须包含或排除的维度。包含的维度只是来自来源指标数据点的汇总指标中的维度。排除的维度只是不在汇总指标中显示的来源指标数据点的维度。
聚合特例规则	为需要与汇总策略中大部分指标不同的聚合函数的指标新建特例规则。例如，如果您的默认聚合是 <avg>，您可能有特定的度量标准，应该使用 count 或 perc<int> 等函数进行聚合。其他符合条件的函数是 max、median、min 和 sum。

如何生成指标汇总摘要

指标汇总摘要是根据可以包含多个子搜索的单个已保存搜索的结果构建的。此搜索在由汇总摘要定义的时间跨度组件决定的计划上运行。它使用默认的聚合函数或可能为特定指标定义的任何特例聚合函数聚合原始指标数据点集。如果已针对摘要策略进行定义，则搜索会将任何不在维度筛选器中的维度删除。

摘要新建的搜索为来源索引中具有相同聚合函数和维度集的一组指标衍生单独的子搜索。

搜索头为由摘要新建的搜索生成的聚合指标数据点指定新的指标名称。新的指标名称符合以下命名约定：<raw_metric_name>_mrollup_<aggregate_function>_<timespan_in_seconds>。

摘要新建的搜索还可将三个新字段添加到各汇总指标数据点。

字段名称	描述
rollup_source_index	来源索引的名称
rollup_span	生成此指标数据点所属汇总摘要的计划搜索的期限
rollup_aggregate	新建此聚合数据点使用的函数

指标汇总摘要生成示例

假设您在名为 HomeIndex 的来源索引上有指标汇总策略。此指标汇总策略的详细信息如下所示：

- 它具有将其目标索引命名为 SumIndex 的汇总摘要定义，并提供 1h 作为其背景计划搜索的期限。
- 是通过 Splunk Web 新建的，因此使用 <avg> 作为其默认的聚合方法。
- 具有以下仅包括三种维度的维度筛选器：ip、app 和 region。这意味着该策略仅汇总包含这些维度中的一个或多个的 HomeIndex 中的指标，并且该策略会删除其为汇总摘要新建的聚合指标数据点中的所有其他维度。

- 它具有名为 `Metric_C` 的指标的例外规则。此规则表示当搜索头为此指标新建汇总指标数据点时，将使用 `max` 函数聚合。

保存此策略之后，摘要新建的搜索开始按小时计划在后台运行。搜索运行时，会在将维度包含在维度集中的 `HomeIndex` 上为各指标衍生子搜索。这些子搜索在每次运行时会生成单个聚合指标数据点。这表示，如果符合资格的指标在过去一小时内 `HomeIndex` 上为 75 个数据点建立索引，那么会根据汇总搜索任务将这 75 个数据点聚合到单个指标数据点中。

所有这些聚合指标数据点将存储在 `SumIndex` 中。各数据点是符合资格的 `HomeIndex` 指标过去一小时内生成的指标数据点聚合。后台搜索为 `SumIndex` 摘要指标数据点提供反映其来源的新指标名称，但也明确将其标识为汇总指标数据点。

要继续示例，假设在 `HomeIndex` 上，您有三个指标：`metric_A`、`metric_B` 和 `metric_C`。他们有不同的维度组合，`metric_C` 具有例外规则，要求其指标数据点的聚合方式与其他数据点不同。下表根据这些指标包含的维度介绍这些指标、用于聚合的函数以及指定的汇总指标数据点的 `metric_name`。

来源索引上的 <code>metric_name</code>	包括 IP 维 度？	包括应用维 度？	包括区域维 度？	聚合函数	目标索引上的 <code>metric_name</code>
<code>metric_A</code>	是	是	是	avg（默认）	<code>metric_A_mrollup_avg_3600s</code>
<code>metric_B</code>	否	否	否	n/a	未汇总，因为缺乏所需维度。
<code>metric_C</code>	是	否	是	max（特例规则）	<code>metric_C_mrollup_max_3600s</code>

只要指标数据点都共享相同的所含维度组合，汇总摘要搜索会汇总指标数据点。在上一个示例中，汇总了 `metric_C` 的所有数据点，因为它们都有 IP 和区域。但是，如果有些数据点属于具有包含维度的指标，而另一些数据点属于缺乏包含维度的指标，那么不会汇总该指标的数据点。

稍后，您可以搜索 `SumIndex`，方法和您当前搜索 `HomeIndex` 的方法完全一样。您可以在更长的时间段内更快地搜索，因为搜索是在较小的指标数据点集上运行的，这些指标数据点只有一到三个维度字段。

您还可以使指标在 `SumIndex` 中的存储时间比您可能将相应指标存储在 `HomeIndex` 上的时间更长，因为它们占用的磁盘空间更少。

使用 Splunk Web 新建和编辑指标汇总策略

本主题向您显示如何使用 Splunk Web 新建或编辑指标汇总策略。

通过 Splunk Web 创建的所有指标汇总策略都是在“搜索和报表”应用的上下文中创建的。

如果要为其他应用中的数据创建指标汇总策略，则需要通过 REST API 调用或直接编辑配置文件的方式来实现。有关更多信息，请参阅：


- 通过 REST API 创建和维护指标汇总策略
- 使用配置文件管理指标汇总策略

为指标索引新建指标汇总策略

前提条件

- 请参阅“汇总指标数据以加快搜索性能并增加存储容量”获取指标汇总策略的概念性介绍。
- 指标汇总策略需要有来源指标索引和一个或多个目标指标索引。这些索引必须在搜索头上可发现。请参阅“指标汇总策略的索引前提条件”。
- 要使用 Splunk Web 新建指标汇总策略，您的角色必须有 `list_metrics_catalog` 和 `edit_metrics_rollup` 功能。请参阅《确保 Splunk Enterprise 安全》中的“关于定义带功能的角色”。

步骤

- 选择 **设置 > 索引** 打开“索引”列表页面。
- 找到您想要定义指标汇总策略的指标索引，然后单击 **编辑** 链接。没有汇总策略的指标索引有一个看起来像测量尺的图标：。
- 向下滚动至 **编辑** 对话框的底部。在 **汇总策略** 下，单击 **新建策略**。
- 定义汇总摘要。选择目标索引和时间范围。

设置	描述
目标索引	这是将存储汇总摘要的指标索引。下拉框仅显示指标索引。
时间范围	此设置提供使用聚合指标数据点填充汇总摘要的搜索期限。


- （可选）单击**添加其他摘要**以添加其他汇总摘要。
- （可选）定义**维度筛选器**。
选择**包含的维度**或**排除的维度**。然后单击**维度**字段选择一个或多个维度。维度列表仅限于过去 24 小时内由来源索引建立索引的维度。

设置	描述
包含的维度	选择此选项表示列出的维度只是来源指标中应包含在指标汇总策略生成的汇总指标中的维度。此外，不会汇总没有这些维度的来源索引中的指标。
排除的维度	选择此选项表示指标汇总策略生成的汇总指标将有列出的维度以外的来源指标中的维度。不会汇总仅有一些排除的维度组合的来源指标。

- （可选）单击**添加例外规则**定义例外规则。
例外规则允许您覆盖特定指标的默认聚合函数。指标汇总策略可以有多个例外规则。

设置	描述
例外指标	从默认指标中选择需要不同聚合函数的指标。列表仅显示过去 24 小时内由来源索引建立索引的指标。
聚合	为指标选择备用聚合函数。

- （可选）单击**一般策略**返回一般策略设置。
- 单击**新建策略**保存新策略。
如果您正在编辑策略，单击**编辑策略**保存您的更改。

在“索引”列表页面中，具有指标汇总策略的指标有一个图标，看起来像是被挤在一起的两个方板，好像中间压缩着一个物体： 

更改默认聚合

当您通过 Splunk Web 创建指标汇总策略时，他们使用 avg 作为默认聚合函数。摘要创建的搜索会将此默认聚合函数应用于其在来源指标索引中发现的指标，保存那些有定义的特例规则的指标。

您无法通过 UI 更改此默认聚合函数，但是如果您有 metric_rollups.conf 的访问权限，您可更改特定指标汇总策略的默认聚合函数。请参阅“通过配置文件管理指标汇总策略”。

通过 REST API 创建和维护指标汇总策略

当遇到在 Splunk Web 中可通过指标汇总策略实施的操作限制时，而且如果您无权访问 Splunk 实施的 metric_rollups.conf 文件，则可以通过手动调用 /catalog/metricstore/rollup REST 端点来创建、更新和删除指标汇总策略。

以下前提条件适用于本主题中的所有操作过程。

前提条件

- 有关指标汇总摘要和管理其创建的策略的概述，请参阅“汇总指标数据以加快搜索性能并增加存储容量”。
- 有关 Splunk 平台 REST API 概念的广泛概述，请参阅《REST API 用户手册》。
- 《REST API 参考手册》中记录了与指标汇总策略相关的端点：
 - /catalog/metricstore/rollup - 用于创建新的指标汇总策略
 - /catalog/metricstore/rollup/{index} - 用于更新或删除现有的指标汇总策略。

为指标汇总策略提供 Splunk Web 中不可用的功能

通过 REST 调用创建或更新指标汇总策略时，可以把它们作为 REST 调用中的 POST 请求参数添加到 /catalog/metricstore/rollup 或 /catalog/metricstore/rollup/{index} 端点中，从而为它们提供 Splunk Web 中不可用的可选功能。

可选功能	POST 请求参数	描述	相对于 Splunk Web 实现了哪些改进
提供多种汇总功能	default_agg	聚合函数列表，用 # 字符分隔开来。提供聚合函数集，汇总搜索使用该函数集聚合来源指标索引中的指标数据点，以生成汇总摘要。default_agg 参数可由 metric_overrides 参数针对特定指标进行覆盖。	在 Splunk Web 中创建的指标汇总策略仅限于用 avg 来聚合指标。
指标筛选	metric_list 和 metric_list_type	这些参数结合起来可在搜索头创建一个过滤器，通过该过滤器让特定指标汇总到摘要中，同时过滤其他指标。您可以指定一组指标名称，然后决定应该将所有指标还是那些指标纳入汇总摘要之内或是排除在汇总摘要之外。	在 Splunk Web 中，您无法设计从结果汇总摘要中筛选特定指标的汇总策略。

指标名称 规则的多个聚合函数	metric_overrides	您可以定义为指标指定两个或以上备用聚合函数的排除规则。使用此参数可以覆盖一个或多个指标的默认聚合。每个指标覆盖配对一个带一个或多个聚合函数（由 # 个字符分隔）的指标名称。	在 Splunk Web 中，您只能为一个指标指定一个备用聚合函数。
-------------------	------------------	--	------------------------------------

为“搜索和报表”以外的应用创建指标汇总策略

默认情况下，通过 Splunk Web 创建的指标汇总策略都是在“搜索和报表”应用的上下文中创建的。但是，通过 REST API 调用，您可以在您喜欢的任何应用的上下文中创建指标汇总策略。

如果要这么做，请使用 /catalog/metricstore/rollup 端点，并将应用文件夹名称嵌入端点 URL 中。遵循以下语法：

https://localhost:8089/servicesNS/nobody/<app-name>/catalog/metricstore/rollup

这将在所指定的应用的上下文中生成指标汇总策略。如果没有添加应用名称，则 Splunk 平台默认会使用 search（表示“搜索和报表”应用）。

您无法为指标汇总策略指定特定所有者。如果您尝试指定一个，则系统会将其所有权重置为 "nobody"，表示所有用户共同拥有。

以下是一个 REST 调用的示例，该调用为 Buttercup Games 应用创建一个指标汇总策略。该指标汇总策略在名为 index_s 的源索引上为数据建立摘要，并将其放在两个目标索引的两个摘要中。

```
curl -k -u admin:changeme https://localhost:8089/servicesNS/nobody/buttercupgames/catalog/metricstore/rollup -d name=index_s -d default_agg=avg#max -d dimension_list="app,region" -d dimension_list_type=included -d metric_overrides="foo2|count#avg,foo1|min#avg" -d summaries="1h|index_d_1h,1d|index_d_1d" -d metric_list="foo3,foo4" -d metric_list_type=excluded
```

有关此 REST 调用中的 POST 请求参数的详细信息，请参阅《REST API 参考手册》中有关 catalog/metricstore/rollup 端点的条目。

使用配置文件管理指标汇总策略

如果您有部署的配置文件的访问权限，则可以手动为您的来源指标索引配置指标汇总策略。

请参阅“汇总指标数据以加快搜索性能并增加存储容量”获取指标汇总策略的概念性介绍。

在新建指标汇总策略配置之前，您应已识别或新建来源指标索引以及一个或多个目标指标索引。这些索引必须在搜索头上可发现。如果您使用分布式搜索，您必须新建独立索引并设置数据转发以启用指标汇总策略。

请参阅“指标汇总策略的索引前提条件”。

创建共享给“搜索和报表”以外的其他应用的指标汇总策略

默认情况下，Splunk 软件会为您通过 Splunk Web 创建的指标汇总策略提供“搜索和报表”应用的上下文。如果您手动配置指标汇总策略，则可以为其他应用创建指标汇总策略，具体方式为：将 metric_rollups.conf 文件添加到该应用的 etc/apps/<app-name>/local 目录中，然后在该文件中放入此汇总策略的配置。这种文件放置方式会创建一个指标汇总策略，该策略的拥有者为 "nobody"，即由应用的所有用户共同拥有。

您无法创建私有或由特定用户拥有的指标汇总策略。Splunk 软件将忽略在 etc/users/<user-name>/<app-name> 中创建的指标汇总策略配置。

某个应用专有的汇总策略会生成计划搜索，这些搜索会将其汇总摘要填充到 etc/apps/<app-name>/local 路径下的 savedsearches.conf 文件中。这些已保存搜索的应用上下文包含在其配置段落名称中，该名称是其对象名称的两倍。这些搜索的名称遵循以下语法：_ss_mrollup_<source_index>__<target_index>_<app_name>。

Splunk Web 中尚未提供的指标汇总功能扩展

当您通过直接编辑配置文件管理指标汇总策略时，您可以利用 Splunk Web 中尚未提供的可选功能扩展。

扩展功能	在 Splunk Web 中	通过配置文件编辑	设置
更改默认聚合功能	新汇总策略的默认聚合功能被改为 avg。无法在 Splunk Web 中更改此设置。	您可以将默认聚合函数改为一个或多个聚合函数。搜索头会汇总您指定的各函数的聚合指标。	defaultAggregation
指标筛选	您无法设计从结果汇总摘要中筛选特定指标的汇总策略。	您可以识别一组指标，这些指标专门包含在汇总策略生成的摘要中或从中排除。	metricList 和 metricListType

指标排除规则的多个聚合函数	您只能为一个指标指定一个备用聚合函数。	您可以定义为指标指定两个或以上备用聚合函数的排除规则。	aggregation.<metric_name>
---------------	---------------------	-----------------------------	---------------------------

您还可以在指标汇总端点上通过 REST API 操作管理扩展的功能。请参阅《REST API 参考手册》中的“Metrics Catalog 端点描述”。

在 metric_rollups.conf 中指定指标汇总策略段落

要配置指标汇总策略，您需要将段落添加到 metric_rollups.conf 文件。

指标汇总策略段落的配置语法如下所示：

```
[index:<Metric Index Name>]
defaultAggregation = <'#' separated list of aggregation functions>
rollup.<summary number>.rollupIndex = <string Index name>
rollup.<summary number>.span = <time range string>
metricList = <comma-separated list of metrics>
metricListType = <excluded/included>
dimensionList = <comma-separated list of dimensions>
dimensionListType = <excluded/included>
aggregation.<metric_name> = <'#' separated list of aggregation functions>
```

下表定义了这些设置。这解释了需要哪些设置，哪些是可选的。

设置	值	是否必需？	描述	默认值
[index:<Metric Index Name>]	来源指标索引的名称。	是	这是段落标题。为指标汇总策略所属的来源指标索引命名。	n/a
defaultAggregation	聚合函数列表，用 # 字符分隔开来。	是	此设置提供聚合来源指标索引中的指标数据点时，汇总搜索使用的聚合函数集以汇总摘要。defaultAggregation 可由 aggregation.<metric_name> 设置针对特定指标进行覆盖。此设置支持以下函数：avg、count、max、median、min、perc<int> 和 sum。	avg
rollup.<summary number>.rollupIndex	汇总的目标索引的名称。	是	此设置是汇总摘要定义的一半。两个汇总摘要定义的一半都应有相同的 <summary number>。要使之生效，指标汇总策略段落必须至少包含一个完整的汇总摘要定义。<string Index name> 是存储摘要所在的目标指标索引的名称。需要此设置。请勿留空。	摘要数量是 1，标题的字符串索引名称是 Metric Index Name
rollup.<summary number>.span	时间范围字符串。	是	此设置是汇总摘要定义的一半。两个汇总摘要定义的一半都应有相同的 <summary number>。指标汇总策略可以有多个汇总摘要定义。要使之生效，指标汇总策略段落必须至少包含一个完整的汇总摘要定义。<time range string> 是计划搜索的期限，该搜索用属于来源索引中指标聚合的汇总指标数据点填充汇总摘要。此设置有一个下边界，受 limits.conf（该文件的默认设置是 300 秒或五分钟）中的 minspanallowed 设置的限制。需要此设置。请勿留空。	summary number = 1，时间范围字符串为 1h
metricList	用逗号隔开的指标名称列表。	否	将此设置和 metricListType 结合使用，在搜索头创建筛选器，允许汇总特定指标，其他的不行。所有列出的指标均应在来源指标索引中有指标数据点。	空字符串
metricListType	[included excluded]	否	将此设置和 metricList 设置结合使用，在搜索头创建筛选器，允许汇总特定指标，其他的不行。当您把 metricListType 设为 excluded 时，除 metricsList 中的指标外，搜索头会汇总来源索引中的所有可用指标。当您把 metricListType 设为 included 时，搜索头只会汇总 metricsList 中的指标，筛选来源索引中的所有其他指标。	excluded
dimensionList	逗号分隔的维度列表。	否	将此设置和 dimensionListType 结合使用，在搜索头创建筛选器，允许汇总特定维度，其他的不行。所有列出的维度均应显示在来源索引的指标数据点中。	空字符串

dimensionListType	[included excluded]	否	将此设置和 dimensionList 设置结合使用，在搜索头创建筛选器，允许汇总特定维度，其他的不行。当您将 dimensionListType 设为 excluded，汇总策略生成的汇总指标将包括除 dimensionList 中的维度之外的所有可用维度。当您将 dimensionListType 设为 included，汇总策略生成的汇总指标将筛选除 dimensionList 中的维度之外的所有可用维度。	excluded
aggregation.<list of aggregation functions>	聚合函数 列表，用 # 字符分 隔开来。	否	使用此可选设置为来源指标索引中的特定 metric_name 提供例外规则。例外规则为 metric_name 定义了一组单独的聚合函数。 使用此属性为该 metric name 指定其他聚合函数。指标汇总策略可以有多个例外规则，只要这些规则各自用于不同的 metric name。请勿设置将使用相同聚合函数集的例外规则设为 defaultAggregation 设置。此设置支持以下函数：avg、count、max、median、min、perc<int> 和 sum。	空字符串

更改汇总摘要搜索的最小允许跨度

rollup.<summary number>.span 设置有一个较低的边界，由 limits.conf 中 [rollup] 段落的 minspanallowed 限制确定。minspanallowed 默认设为 300 秒或 5 分钟。如果您为低于 minspanallowed 的汇总摘要搜索提供跨度，您将看到一条错误消息。

此限制是为了防止您以可能导致搜索并发问题的频率设置汇总摘要搜索，这样，该运行的计划搜索将无法运行，因为同时运行了太多搜索。但是，如果您需要更改此限制，您可以进行更改。请勿将 minspanallowed 值设置低于 60 秒。

与指标结合使用

将 Analytics Workspace 中指标可视化

在 Analytics Workspace 中新建指标数据的交互式图表。您可以监视和分析指标，无需使用 Splunk Analytics Workspace 中的 Splunk 搜索处理语言 (SPL)。

Analytics Workspace 功能和操作

使用 Analytics Workspace 在指标上执行以下分析功能和操作：

- 新建图标帮助您可视化数据中的关联性。
- 将数据点聚合为有意义的值。
- 转换图表的时间范围以比较早期时期的指标。
- 按指定的维度拆分指标。
- 筛选指标以包括或排除特定的值。

您还可以将工作区图表另存为 Splunk 平台中的告警和仪表板。

Enterprise 7.3 及更高版本中包含 Splunk Analytics Workspace。要访问 Splunk Metrics Workspace：

1. 打开“搜索和报表”应用。
2. 单击“搜索和报表”栏上的指标选项卡。

有关 Analytics Workspace 的更多信息，请参阅《*Analytics Workspace*》手册中的“关于 Analytics Workspace”。

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`。

要以较小的规模在单个指标数据点上进行搜索，而又不涉及 `mstats` 聚合，请使用 `mpreview` 命令。`mpreview` 命令是一种工具，用于导入指标数据、排除指标数据中的问题，以及探索指标索引。请参阅《*搜索参考*》手册中的 `mpreview`。

要在搜索时将日志事件转换为指标数据点并将这些指标数据点写入指标索引，使用 `mcollect` 或 `meventcollect` 命令。请参阅《*搜索参考*》手册中的 `mcollect` 和 `meventcollect`。

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。

从 Splunk 平台版本 8.0.0 开始，指标索引和搜索区分大小写。因此，举例来说，指标搜索命令会将以下内容视为三个不同的指标：cap.gear、CAP.GEAR 和 Cap.Gear。

搜索示例

要列出所有指标索引中所有指标名称：

```
| mcatalog values(metric_name) WHERE index=*
```

要列出所有指标索引中所有维度：

```
| mcatalog values(_dims) WHERE index=*
```

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
```

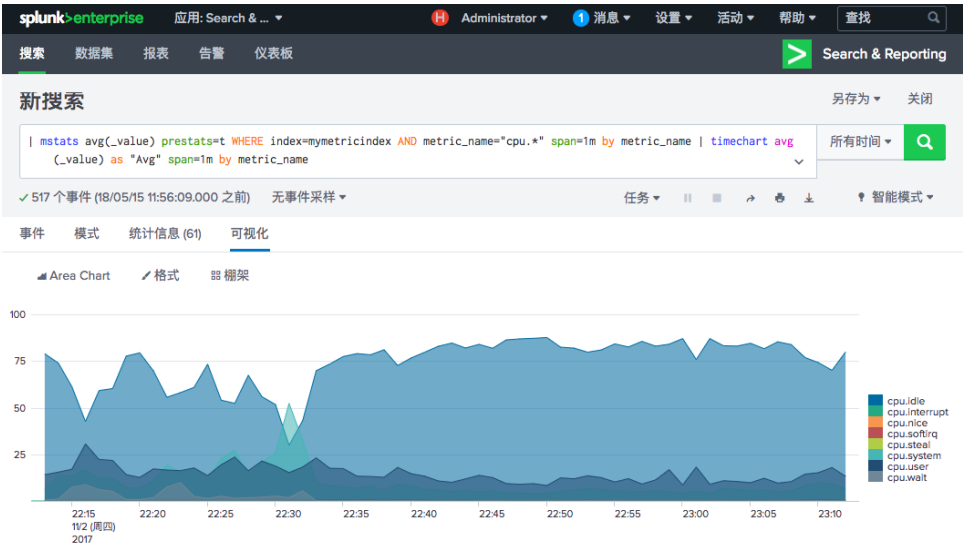
要计算每 30 秒间隔的测量平均值：

```
| 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
```

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”。

对指标时间序列执行统计计算

指标时间序列是一组指标数据点，这些数据点共享一个指标的唯一组合和一组维度字段-值对。

例如，假设您有一个名为 miles.driven 的指标。此指标表示各种赛车的里程表读数。miles.driven 的指标数据点包括以下维度：vehicle_type、engine_type、vehicle_number 和 driver_name。

下表显示了按 _time 排序的一组指标数据点。您可以看到，对于 miles.driven 指标它们分成了两个不同的指标时间序列：

_time	metric_name:miles.driven	vehicle_type	engine_type	vehicle_number	driver_name
01-05-2020 16:26:42.025 -0700	134.0643	Ferrari	F136	011	LanaR
01-05-2020 16:26:41.834 -0700	128.4515	Ferrari	F136	009	RavenM

01-05-2020 16:26:41.655 -0700	133.7509	Ferrari	F136	011	LanaR
01-05-2020 16:26:41.007 -0700	127.8861	Ferrari	F136	009	RavenM
01-05-2020 16:26:40.623 -0700	127.1277	Ferrari	F136	009	RavenM
01-05-2020 16:26:40.014 -0700	133.2482	Ferrari	F136	011	LanaR

此指标数据点表中的两个指标时间序列都将 `Ferrari` 作为车辆类型，将 `F136` 作为 `engine_type`，但是它们具有不同的 `vehicle_number` 和 `vehicle_driver` 值。`vehicle_number=009` 和 `driver_name=RavenM` 的指标数据点组成一个不同的指标时间序列。`vehicle_number=011` 和 `driver_name=LanaR` 的指标数据点组成另一个不同的指标时间序列。

正如不同的 `car_number` 和 `driver_name` 值所示，此样本中的指标数据点来自约在同一时间行驶的两辆不同的汽车。如果要获取 `miles.driven` 指标的平均 `rate(X)`，则没有必要计算所有这六个指标数据点的平均速率。相反，获取按指标时间序列分组的平均速率，这样就不会把汽车混在一起。

如果您在搜索中调出与特定指标相关的所有维度，则可以对与该指标相关的时间序列执行统计计算。但是这种方法很难实行，特别是对于涉及大量维度的指标来说更是如此。

```
| mstats avg(miles.driven) BY vehicle_type engine_type vehicle_number driver_name
```

特殊的 `_timeseries` 字段替换了那些可能很长的维度列表。将其与 `mstats` 结合使用，以计算每个时间序列的统计信息。例如，此搜索会检索样本中的两个时间序列的平均 `miles.driven`：

```
| mstats avg(miles.driven) BY _timeseries
```

有关更多信息，请参阅《搜索参考》中的 `mstats`。

`_timeseries` 是一个内部字段

`_timeseries` 是一个内部字段，未显示在 Splunk Web 界面中。如果想让它显示在结果，则需要执行 `rename` 命令以便将 `_timeseries` 显示为 `timeseries` 或 `time_series`。

```
| mstats avg(miles.driven) BY _timeseries | rename _timeseries AS timeseries
```

当由 `mstats` 以外的命令处理 `_timeseries` 的值时，将 `_timeseries` 与 `group-by` 字段组合使用

`_timeseries` 是 JSON 格式的字段。因此，如果需要通过其他非 `mstats` 命令（例如 `stats`）处理其值，则可以将其与另一个 `group-by` 字段组合使用。这种方式最适用于所有结果共享同一个指标时间序列的情况。

以下搜索使用 `mstats` 来计算与 `miles.driven` 指标相关的时间序列的速率。然后，它使用 `stats` 来计算每个速度的总和。

```
mstats rate(miles.driven) as driven BY vehicle_number, _timeseries | stats sum(rate(miles.driven)) BY vehicle_number
```

您可以使用 `rate_sum(X)` 函数简化此示例搜索。

请参阅《搜索参考》中的“时间函数”。

调查计数器指标

计数器指标是最常见的指标类型之一。计数器指标值总是在更改时增加，除非在重新启动时将其重置为零。换句话说，该值是单调增加的。

您可以使用计数器指标计数。汽车里程表提供了计数器指标的简单示例。里程表表示汽车已行驶的英里数。里程表值从不减少，除非重设为零。

计数器指标一般用来统计事件数量。例如，大多数网络指标都涉及事件计数，无论您是在谈论网站访问、网络接口错误、发送或接收的软件包还是磁盘操作。

定期计数器和累计计数器

计数器指标有两种类型：定期计数器和累计计数器。下表介绍这些指标类型、列出相关的指标协议，并列出您用于查询指标类型的关键 SPL。

计数器指标类型	描述	指标线路协议	用于查询的 SPL
定期	每次向服务器发送测量值时，客户端都会将计数器的值重置为零，这意味着每个数据点都是独立的。	StatsD、collected ABSOLUTE、collected DERIVE (storerates=true)	使用 mstats、stats、带 sum(x) 的 tstats 或带 per_*(x) 的 timechart。
累计	只有在重置服务时，计数器的值才会重置为零。将各新值添加到上一个值中。您可以比较两个测量值来获取累计率。	collected COUNTER、collected DERIVE (storerates=false)	如果 Splunk 平台版本是 7.2.x 或更高版本，使用带 rate(x) 的 mstats。如果 Splunk 平台版本是 7.0.x 或 7.1.x，使用带 latest(x) 和 eval 的 streamstats。

定期计数器总计

由于每次指标客户端将定期计数器发送到 Splunk 平台时定期计数器重置为零的方式，这些定期计数器会报告为一系列独立的测量值。要查看这些测量值如何用作计数器，您可以运行 mstats、stats 或 tstats 搜索，这些搜索使用 sum(x) 函数将这些测量值聚合在一起。或者，您可以运行使用其中一种 per_*(x) 函数将这些测量值聚合在一起的 timechart 搜索。

获取累计计数器的计数率

追踪累计计数器指标的人员通常会发现随着时间的推移计数率是一个比计数更有趣的测量值。计数率会告诉您指标活动何时加速或减慢，这可能是某些指标的重要信息。

您确定计数器速率的方式很大程度上取决于您的 Splunk 平台实施版本。如果您正在使用 7.0.x 或 7.1.x，您将 streamstats 与 latest(x) 和 eval 结合使用以返回累计计数器的速率。如果您的 Splunk 平台实施版本是 7.2.x 或更高版本，您可将 mstats 和 rate(x) 函数结合使用以获取计数器速率。

两种获取计数器速率的方法返回的结果略有不同。发生这种情况是因为它们比较的计数值集不同。

速率确定方法	速率计算中使用的计数值差异	示例
streamstats, latest(x) 函数, 和 eval	使用前一时间跨度内最新事件的计数值与当前时间跨度内最新事件的计数值之间的差值	如果您的时间跨度为 1h，则要获得 2 P.M. 的速率，您可能会获得 1 P.M. - 2 P.M. 时间跨度的最新事件，并将其与 2 P.M. - 3 P.M. 时间跨度的最新事件进行比较。
mstats 使用 rate(x) 函数	使用时间跨度内最早事件的计数值与同一时间跨度内最新事件的计数值之间的差值。	如果您的时间跨度为 1h，要获得 2 P.M. 的速率，您可能会从 1 P.M. - 2 P.M. 时间跨度获得最早事件，并将其与 1 P.M. - 2 P.M. 时间跨度的最新事件进行比较。

为计数器速率搜索构建 SPL 时，确保您没有混淆计数器指标。如果您需要报告多个计数器指标，使用 BY 子句进行分离。您还应设置 name=indexerpipe processor=index_thruput 以关注一个特定的计数器指标。

使用 streamstats、latest(x) 和 eval 返回计数器速率

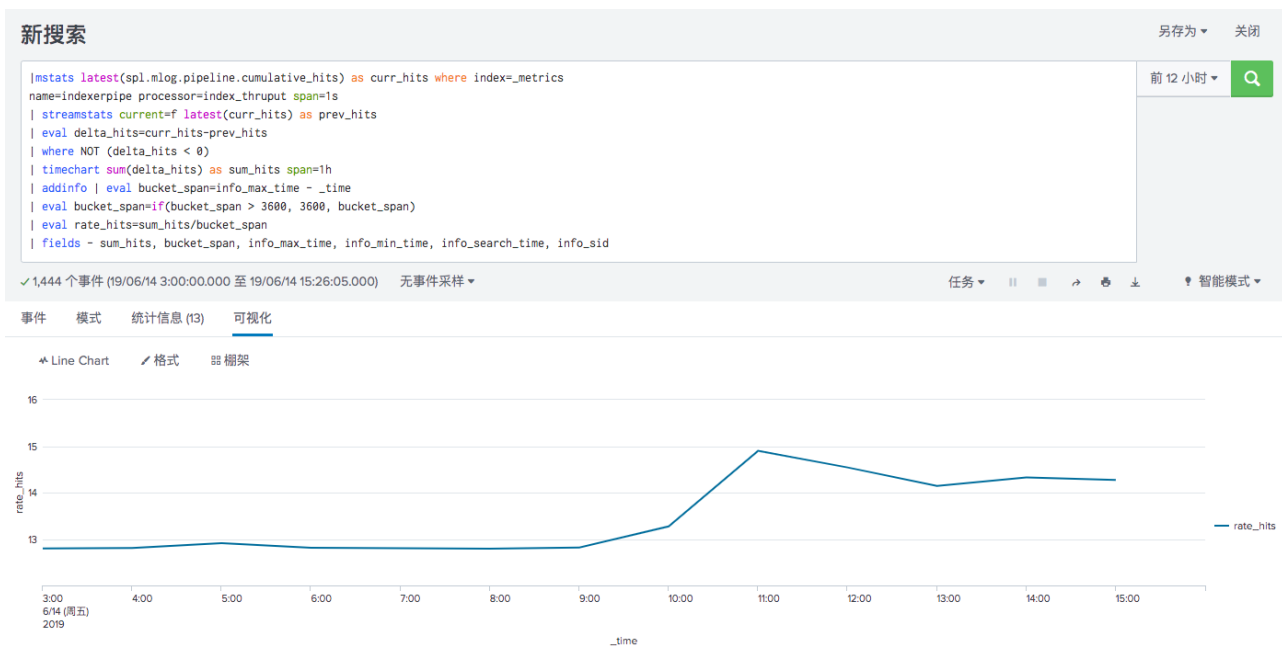
如果您的 Splunk 版本是 7.0.x 或 7.1.x，或者 rate(x) 函数不合适，使用 streamstats、latest(x) 函数和 eval。例如，如果您不能指望每个时间跨度有两个指标数据点，那么您可能会坚持使用 streamstats。

如果您使用此方法，确保设置 current=f 以强制搜索使用上一个时间跨度中的最新值。

以下为计数器速率搜索示例，使用 streamstats、latest(x) 和 eval 进行计算：

```
| mstats latest(pipeline.cumulative_hits) as curr_hits where index=_metrics name=indexerpipe processor=index_thruput span=1s |
streamstats current=f latest(curr_hits) as prev_hits | eval delta_hits=curr_hits-prev_hits | where NOT (delta_hits < 0) |
timechart sum(delta_hits) as sum_hits span=1h | addinfo | eval bucket_span=info_max_time - _time | eval
bucket_span=if(bucket_span > 3600, 3600, bucket_span) | eval rate_hits=sum_hits/bucket_span | fields - sum_hits, bucket_span,
info_max_time, info_min_time, info_search_time, info_sid
```

这是此搜索返回的折线图的示例。



走查

以下为示例搜索的逐步走查。

1. 使用 `mstats`、`streamstats` 和 `eval` 组合获得每秒的 `delta` 计数。

```
| mstats latest(pipeline.cumulative_hits) as curr_hits where index=_metrics name=indexerpipe processor=index_thruput
span=1s
| streamstats current=f latest(curr_hits) as prev_hits
| eval delta_hits=curr_hits-prev_hits
| where NOT (delta_hits < 0)
```

注意：`streamstats` 使用 `current=f`。这会强制搜索使用上一个时间跨度中的最新值。

2. 计算每小时 `delta` 计数的总和。

```
| timechart sum(delta_hits) as sum_hits span=1h
```

3. 计算数据桶的时间跨度。应为 1h，除非是最新的数据桶，这样可能不满 1h。

```
| addinfo | eval bucket_span=info_max_time - _time
| eval bucket_span=if(bucket_span > 3600, 3600, bucket_span)
```

4. 最后，用以下函数 `rate = delta_count/time_range` 计算速率。

```
| eval rate_hits=sum_hits/bucket_span
| fields - sum_hits, bucket_span, info_max_time, info_min_time, info_search_time, info_sid
```

将 `mstats` 和 `rate(x)` 函数结合使用以返回计数器速率

如果您正在使用 Splunk 平台 7.2.x 或更高版本，将 `mstats` 和 `rate(x)` 函数结合使用以确定计数器速率。

要使用 `mstats` 和 `rate(x)` 获取适当的速率测量值，您需要在搜索中每个时间跨度至少有两个计数器事件。Splunk 平台使用这两个值之间的差异来确定实际速率。如果您无法保证每个时间跨度有两个指标数据点，您可能会改用 `streamstats` 方法。

`rate(x)` 函数使用以下计算衍生其值：

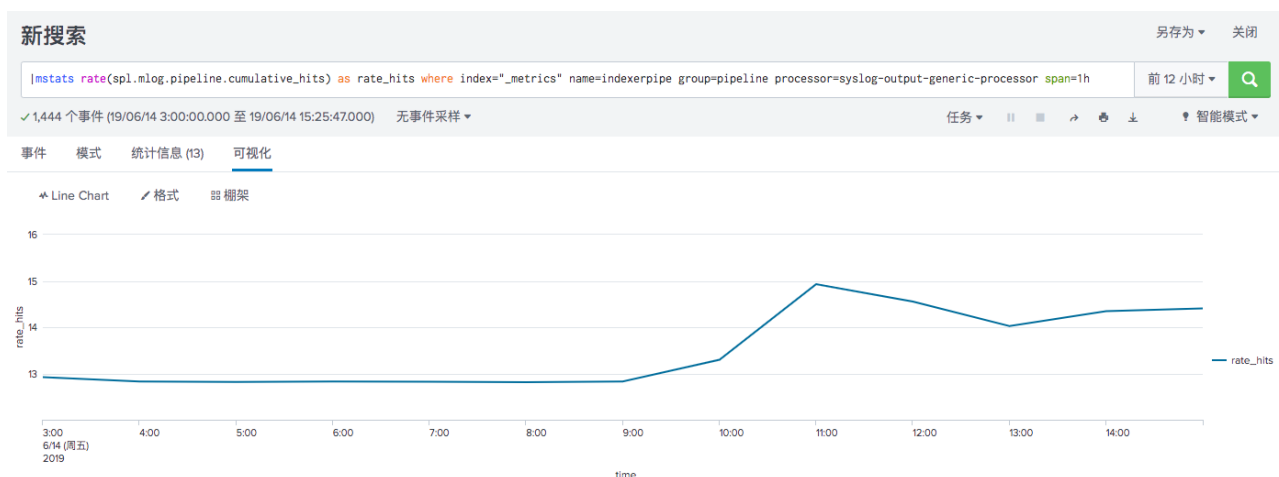
$$(\text{latest}(\langle \text{counter_field} \rangle) - \text{earliest}(\langle \text{counter_field} \rangle)) / (\text{latest_time}(\langle \text{counter_field} \rangle) - \text{earliest_time}(\langle \text{counter_field} \rangle))$$

请参阅《搜索参考》中的“时间函数”获取有关这些函数的更多信息。

以下为计数器速率搜索示例，使用 `mstats` 和 `rate(x)` 获取计数器速率：

```
| mstats rate(pipeline.cumulative_hits) as rate_hits where index=_metrics name=indexerpipe processor=index_thruput span=1h
```

这是此搜索返回的折线图的示例。



针对累计计数器指标计算平均速率和合计速率

使用 `rate_avg(X)` 和 `rate_sum(X)` 函数可得出累计计数器指标的平均速率和合计速率。这些函数都考虑了**指标时间序列**，以提高计算的准确性。这些函数首先计算指标的速率，并按指标时间序列分组。然后，它们会生成这些指标时间序列的平均值或合计值，具体取决于您使用的是函数。

这些函数接收使用了 `_timeseries` 字段的相对复杂的搜索，例如：

```
| mstats rate(spl.mlog.thruput.thruput.total_k_processed) where index=_metrics BY _timeseries | spath input=_timeseries | stats sum(rate(spl.mlog.thruput.thruput.total_k_processed)) span=1h
```

并将其转换为较简单的搜索，例如：

```
| mstats rate_sum(spl.mlog.thruput.thruput.total_k_processed) where index=_metrics span=1h
```

`rate_avg(X)` 和 `rate_sum(X)` 函数的另一个好处是，即使每个时间跨度的每个指标时间序列只有一个指标数据点，也能够计算出速率。这些函数可以跨时间段提取数据以计算速率。

有关指标时间序列和 `_timeseries` 字段的更多信息，请参阅“对指标时间序列执行统计计算”。

有关 `rate_avg(X)` 和 `rate_sum(X)` 函数的更多信息，请参阅《搜索参考》中的“时间函数”。

使用直方图指标

直方图是复杂的指标数据类型。直方图数据点定义不同大小的数据桶集。其指标回答以下关于指定时间内跨这些数据桶的测量值分布的问题，如请求持续时间或响应大小。

- 指标有多少测量值少于或等于各数据桶上限值？例如，在指定时间内，在 0.1 秒或更少的请求持续时间内，有多少记录的请求持续时间测量值在数据桶范围内？在 0.5 秒或更少的请求持续时间内，有多少记录的请求持续时间测量值在数据桶范围内？以此类推。
- 为指标记录的所有测量值总和是多少？
- 为指标记录的所有测量值完整计数是多少？

有很多方式可以处理这种情况：

- 对于请求持续时间这样的指标而言，您可以使用请求持续时间测量的计数和测量值的总和获取指定时间范围内平均请求持续时间。
- 您可以设计告警监视特定数据桶的测量计数。例如，您可以设置告警，如果 300 毫秒内服务的请求数量低于请求总数的 95%，告警会通知您。
- 您可以预估百分比值，如您已服务 95% 的请求内的请求持续时间。

索引前后的直方图指标

要在 Splunk 平台中使用直方图指标，您需要从 Prometheus 或使用 HTTP 事件收集器或 DSP 的类似指标监视客户端引入直方图格式的指标数据点。

直方图数据点结构由几个单独的指标数据点组成。在您的 Splunk 平台实施引入数据之前，处理 HTTP 请求持续时间（以秒为单位）的原始直方图数据点应像下方所示：

http_req_dur_sec_bucket{le="0.05",server="ronnie",endpoint="/"} 24054 1568073334000
http_req_dur_sec_bucket{le="0.1",server="ronnie",endpoint="/"} 33444 1568073334000
http_req_dur_sec_bucket{le="0.2",server="ronnie",endpoint="/"} 100392 1568073334000
http_req_dur_sec_bucket{le="0.5",server="ronnie",endpoint="/"} 129389 1568073334000
http_req_dur_sec_bucket{le="1",server="ronnie",endpoint="/"} 133988 1568073334000
http_req_dur_sec_bucket{le="+Inf",server="ronnie",endpoint="/"} 144320 1568073334000
http_req_dur_sec_sum{server="ronnie",endpoint="/"} 53423 1568073334000
http_req_dur_sec_count{server="ronnie",endpoint="/"} 144320 1568073334000

直方图数据点的每一行都有此格式：

<metric_name>{<dim0>=<dim_value0>,<dim1>=<dim_value1>,...,<dimN>=<dim_valueN>} <_value><timestamp>

引入后，直方图数据点如下所示：

metric_name: http_req_dur_sec_bucket	metric_name: http_req_dur_sec_sum	metric_name: http_req_dur_sec_count	le	服务器	数据点	时间戳（单位为秒）
24054			0.05	ronnie	/	1568073334000
33444			0.1	ronnie	/	1568073334000
100392			0.2	ronnie	/	1568073334000
129389			0.5	ronnie	/	1568073334000
133988			1	ronnie	/	1568073334000
144320			+Inf	ronnie	/	1568073334000
	53423			ronnie	/	1568073334000
		144320		ronnie	/	1568073334000

此表中的各单个指标是整个直方图指标数据点的组件。

直方图指标数据点的分析

各直方图数据点由三种类型的指标组成。每种指标类型都表示关于某个特定指标（自直方图的时间戳以来）所记录的一些测量。各指标也是一个累计计数器指标的示例，表示值未随时间减少。有关计数器指标的更多信息，请参阅“调查计数器指标”。

指标类型	定义	示例
<metric_name>_bucket	提供直方图数据桶的测量值计数。该字段的值与 le 维度（该维度设置数据桶的上边界值）的值相对应。	在我们的示例指标数据点中，24,054 个 http_req_dur_sec 测量值小于或等于 0.05 秒（自直方图数据点的时间戳）。同时，33,444 个 http_req_dur_sec 测量值小于或等于 0.1 秒。
<metric_name>_sum	提供此直方图数据点中捕获的所有 <metric_name> 测量值的总和。	在我们的示例指标数据点中，此指标数据点中包含的所有 http_req_dur_sec 测量值的总和是 53,243 秒。
<metric_name>_count	提供此数据点中捕获的所有测量值的计数。	我们的示例直方图数据点表示 144,320 个 http_req_dur_sec 测量值的分布。

数据桶计数指标和数据桶边界维度

直方图数据点中通常有几种数据桶计数指标。他们形成一个数据桶边界（由 le 维度定义）越来越大的数据桶的同时测量值计数顺序。第一个数据桶计数只表示值相对较小的测量值。第二个数据桶计数表示第一个计数的测量值加上值稍微大些的测量值。

这个序列一直持续到最后的数据桶计数，和 +Inf 的 le 维度值对应。+Inf 是 "Infinite" 的缩写。这表示最后的数据桶会捕获之前的数据桶捕获的所有测量值，以及超出之前的数据桶的任何测量值。+Inf <metric_name> bucket 的计数应相当于 <metric_name>_count 字段值。两个字段都提供了自直方图时间戳以来按直方图数据点分类的所有测量值。

根据设计，<metric_name>_sum 和 <metric_name>_sum 字段没有 le 值。

Prometheus 客户端要求数据桶边界维度命名为 le（字段名是“小于或等于”的首字母缩写），但是 Splunk 软件非常灵活，如

果 `le` 不适合您的需求，可以使用其他名称。

在搜索中使用直方图指标

因为直方图指标数据包含互联的计数器指标集，您可以将 `rate(x)` 函数和 `mstats` 结合使用以显示指定时间跨度中的数据桶分布。

`_timeseries` 字段也十分重要。该字段允许您在 `rate(x)` 计算之后的命令中按各种维度字段进行分组。这允许您能够执行与 Prometheus 客户端允许的计算类似的计算，当没有显式 `by` 语句时，每个类似于 `stats` 的操作都会隐式执行类似于 `by _timeseries` 的操作。请参阅“对指标时间序列执行统计计算”。

测量值的计数和总和

每个直方图数据点都有一个所有测量值（自直方图时间戳以来）的计数以及所有这些测量值的值的总和。您可以使用此信息计算指定时间内的平均测量值。在以下示例中，我们用它计算过去五分钟内的平均请求。

```
| mstats rate(http_req_dur_sec_sum) as sum_req_duration where index="metrics" AND earliest="-5m" by _timeseries | appendcols [
| mstats rate(http_req_dur_secs_count) as num_req where index="metrics" AND earliest="-5m" by _timeseries ] | eval
avg_req_duration=sum_req_duration / num_req | fields avg_req_duration
```

不良服务率的告警

您可使用直方图指标设计告警，当您的 http 请求服务率低于特定阈值时，触发告警。例如，假设您的服务水平协议是在 300 毫秒内服务 95% 的 http 请求。您可以配置直方图，数据桶上限是 0.3 秒。以下搜索根据过去 5 分钟内的作业计算 300 毫秒内服务的相对请求数量。您可以在告警定义中使用此服务，当作业的 `percent_requests_served` 低于 95 时触发告警

```
| mstats rate(http_req_dur_sec_bucket) as bkt_req_per_sec where index="metrics" AND le=0.3 AND earliest="-5m" by _timeseries,
job | stats sum(bkt_req_per_sec) as sum_bkt_req_per_sec by job | appendcols [ | mstats rate(http_req_dur_sec_count) as
req_per_sec where index="metrics" AND earliest="-5m" by _timeseries, job | stats sum(req_per_sec) as sum_req_per_sec by job ] |
eval percent_requests_served=sum_bkt_req_per_sec / sum_req_per_sec | fields job, percent_requests_served
```

粗略估计 Apdex 评分

Apdex 评分通过计算满意性能测量值和不满意性能测量值的比率，提供应用程序性能的客户满意度的数字测量。您可使用直方图指标粗略估计 Apdex 评分。

假设您想要针对 `http_req_dur_sec` 直方图进行估算。首先将数据桶目标请求持续时间配置为上限。然后将另一数据桶允许的请求持续时间配置为上限，通常是目标请求持续时间的 4 倍。例如，如果目标请求时间是 300 毫秒，那么允许的请求持续时间是 1.2 秒。

以下表达式给出了过去 5 分钟内各作业的 Apdex 评分：

```
| mstats rate(http_req_dur_sec_bucket) as bkt_req_per_sec_0.3 where index="metrics" AND le=0.3 AND earliest="-5m" by
_timeseries, job | stats sum(bkt_req_per_sec_0.3) as sum_bkt_req_per_sec_0.3 by job | appendcols [ | mstats
rate(http_req_dur_sec_bucket) as bkt_req_per_sec_1.2 where index="metrics" AND le=1.2 AND earliest="-5m" by _timeseries, job |
stats sum(bkt_req_per_sec_1.2) as sum_bkt_req_per_sec_1.2 by job ] | appendcols [ | mstats rate(http_req_dur_sec_count) as
req_per_sec where index="metrics" AND earliest="-5m" by _timeseries, job | stats sum(req_per_sec) as sum_req_per_sec by job ] |
eval apdex_score=(sum_bkt_req_per_sec_0.3 + sum_bkt_req_per_sec_1.2) / 2 / sum_req_per_sec | fields job, apdex_score
```

此搜索将两个数据桶的总和分开，因为直方图数据桶是累计的。`le=0.3` 数据桶包含在 `le=1.2` 数据桶中。除以 2 就可以了。
计算不会和传统的 Apdex 评分完全匹配，因为其中包含了计算令人满意和允许的部分中的错误。

用 histperc 宏计算百分比值

Histperc 宏允许您计算直方图指标的百分位数值。该宏计算数据桶边界和计数器的增长率，并根据直方图边界之间的某些线性内插预估和指定百分位数相关的值。

假设您有一个名为 `http_req_dur_sec` 的直方图宏，它以秒为单位提供 HTTP 请求持续时间测量的分布。您可以使用 `histperc` 宏计算请求持续时间，在此期间您服务了 95% 请求，也称为请求服务的 P95 值。

要这样操作，您要设置 `histperc` 宏。Histperc 宏获取了四个参数，最后一个是可选的。

```
histperc(<perc>, <rate_field>, <bucket_upper_boundary_dimension> [, <groupby-fields>])
```

参数	描述	是否必需？
----	----	-------

<i>perc</i>	所需的百分比值。该值必须介于 0.0 到 1.0 之间。	是
<i>rate_field</i>	包含 <code>mstats rate(x)</code> 命令输出的字段名称。直方图宏使用此输出生成某段时间的直方图分布。	是
<i>bucket_upper_boundary_dimension</i>	表示直方图数据结构中包含的完整的数据桶上边界的维度名称。Prometheus 指标使用 <code>le</code> ，表示“小于或等于”。	是
<i>groupby-fields</i>	百分位数计算期间要按其分组的一个或多个维度。字段列表必须用引号括起来，并用逗号隔开。	否

Histperc 的三个参数版本在设置中的搜索宏页面上以 `histperc(3)` 形式列出。带 `groupby-field` 参数的四个参数版本在设置中的搜索宏页面上以 `histperc(4)` 形式列出。

Histperc 宏示例

此搜索会计算您已服务了 99% 请求的 HTTP 请求持续时间。它会按 `_time` 对结果进行分组以绘制表格。

```
| mstats rate(http_req_dur_sec_bucket) as requests_per_sec where index="metrics" by _timeseries, le span=5m | stats
sum(requests_per_sec) as total_requests_per_sec by _time, le | `histperc(0.99, total_requests_per_sec, le, _time)`
```

关于这些示例

这些示例是基于 Prometheus（一个开源的指标监视和报警系统）用于说明他们支持的直方图指标类型的示例。

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

如果您正在对 StatsD 格式的数据建立索引，使用带有维度扩展的 StatsD 格式以获得更好的性能：`cpu.idle:0.5|g|#host:some-hostsplunk.com,app:some-app`

使用带维度扩展的格式，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。