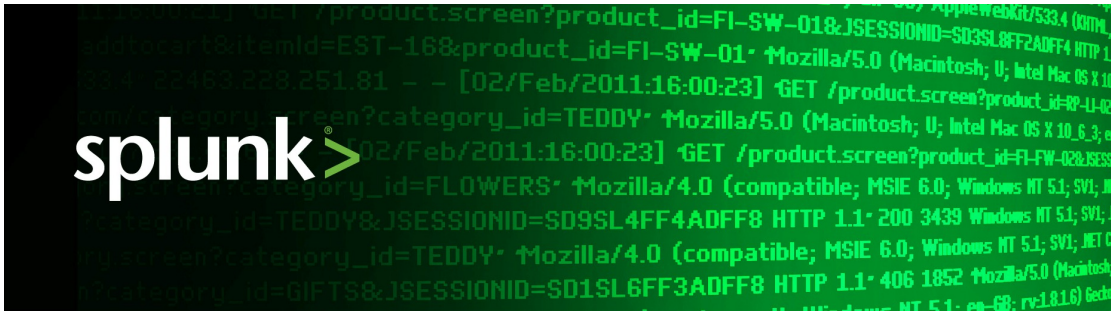


Splunk® Enterprise 7.1.0

搜索手册

生成时间：2018 年 4 月 18 日下午 3:21

Table of Contents

搜索概述

搜索入门

该手册介绍**搜索和报表应用**以及如何使用 Splunk 搜索处理语言 (**SPL**)。

搜索和报表应用简称为搜索应用，是您在 Splunk 部署中导航数据的主要方式。搜索应用由一个基于网络的界面 (Splunk Web)、一个命令行界面 (CLI) 和 Splunk SPL 组成。

从这里开始

如果您之前未使用过 Splunk 搜索，了解相关信息的最佳方式是从搜索教程开始。搜索教程介绍了搜索和报表应用，并逐步指导您添加数据、搜索数据及构建简单报表和仪表板。

搜索教程为您了解 Splunk 搜索提供良好基础。

在自己的环境中进行入门学习

在完成搜索教程的学习后，您应该会了解可以探索的数据类型，Splunk 软件索引数据的方式，以及 Splunk 知识对象等。

以下是可以查看的资源：

- 将数据上载到 Splunk 部署。请参阅《数据导入手册》。
- 了解索引如何工作。请参阅《管理索引器和索引器群集》手册。
- 了解字段和知识对象，如主机、来源类型和事件类型。请参阅《知识管理器手册》。

有效使用搜索应用

当然，您还需要了解如何有效地使用搜索应用，而这正是本手册的重点所在。本手册包含有关如何搜索数据的详细信息。

搜索应用基本技巧

- 浏览 Splunk Web
- 使用搜索应用
- 搜索类型
- 命令类型

详细的搜索信息

- 检索事件
- 指定时间范围
- 优化搜索
- 创建表格和图表
- 评估和操作字段
- 计算统计信息和高级统计
- 事件分组和相关性
- 管理搜索任务

搜索命令参考

如果您对组成 Splunk SPL 的搜索命令和参数目录感兴趣，请参阅*搜索参考*。

分布式搜索

如果您正在使用 Splunk Enterprise，**分布式搜索**提供了一种调整部署的方式，即将搜索管理和显示层从索引和搜索检索层分离。有关分布式搜索的介绍，请参阅*《分布式搜索手册》*。

另请参阅

浏览 Splunk Web
使用 Splunk 搜索

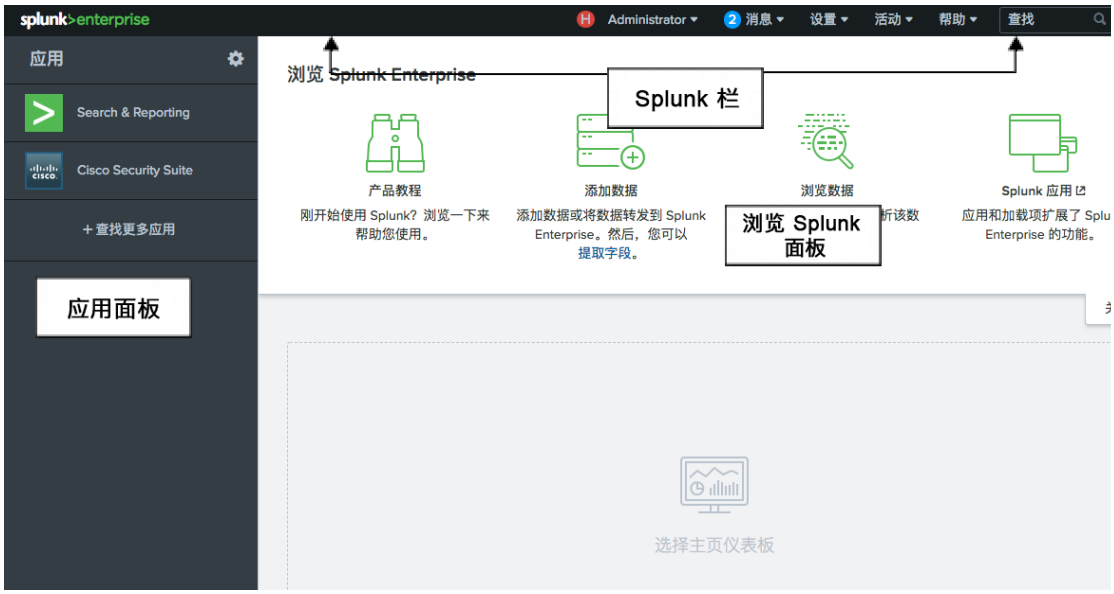
浏览 Splunk Web

本主题介绍在 Splunk 的网络浏览器界面 **Splunk Web** 中的不同视图之间导航。

关于 Splunk Home

Splunk Home 是 Splunk 部署中数据和应用的交互门户。当您首次登录 Splunk 部署时，会进入 Splunk Home。此页面会显示您的所有应用。

Splunk Home 的主要部分包括导航栏、应用菜单、浏览面板和一个自定义的默认仪表板（这里未显示）。



您可以对自己的 Splunk 帐户进行配置，以便在启动 Splunk 时显示其他视图而非 Splunk Home，例如**搜索和报表**应用的“搜索”或“数据透视表”视图。

应用面板

“应用”面板列出您有权查看的 Splunk 实例上安装的应用。从该列表中选择应用并打开它。**搜索和报表**应用常被简称为 Splunk 搜索。如果您有多个应用，可以在工作区内对其执行拖放来进行重新排列。

您可启用下列操作。

- 单击齿轮图标查看并管理安装在 Splunk 部署中的应用。

- 单击加号图标以浏览更多要安装的应用。

浏览面板

浏览面板上的选项可帮助您开始操作。单击相应图标可打开**添加数据**视图、浏览新的应用、打开用户文档或打开 Splunk Answers。

主页仪表板

主页仪表板位于浏览面板下方。当您第一次打开 Splunk Home 时，不会出现默认的仪表板。

单击标记为**选择主页仪表板**的区域，并选择一个默认仪表板。

如果您以前未使用过 Splunk 软件，在您创建并保存一些搜索之前，先不要选择默认仪表板。您可能想创建自己的仪表板，并将它用作默认仪表板。

有关仪表板的更多信息，请参阅*《仪表板和可视化》*手册。

有关 Splunk 栏

使用 Splunk 栏导航 Splunk Web。您可以使用 Splunk 栏在应用之间切换、添加数据、管理设置和编辑 Splunk 配置、查看系统级别消息、监视搜索任务和告警活动以及获取 Splunk 软件使用帮助。

其他视图中的 Splunk 栏，如**搜索和报表**应用的“搜索”视图，也包括 Splunk 徽标旁边的**应用**菜单。使用**应用**菜单在已在计算机上安装的 Splunk 应用之间快速切换。

返回到 Splunk Home

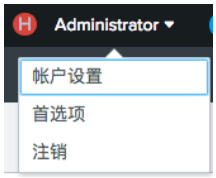
单击导航栏上的 Splunk 徽标，可从 Splunk Web 中的任何其他视图回到 Splunk Home。

“设置”菜单

“设置”菜单列出了与知识对象、分布式环境设置、系统和许可、数据和验证设置有关的配置页面。如果您看不到其中某些选项，那么您无权查看或编辑它们。

帐户菜单

使用**帐户**菜单以编辑帐户设置，或登出此 Splunk 安装。这里的**用户菜单**称为“管理员”，因为这是新安装的默认用户名。可以通过选择**编辑帐户**和更改**全名**来更改此显示名称。您还可以编辑其他设置，包括：时区设置、此帐户的默认应用和帐户密码。



"消息"菜单

所有系统级错误消息都在**消息**菜单中列出。当有可查看的新消息时，**消息**菜单旁边将显示数字通知。通知显示为一个数字，代表您收到的消息数量。

"活动"菜单

"活动"菜单列出了到任务和触发的告警视图的快捷键。

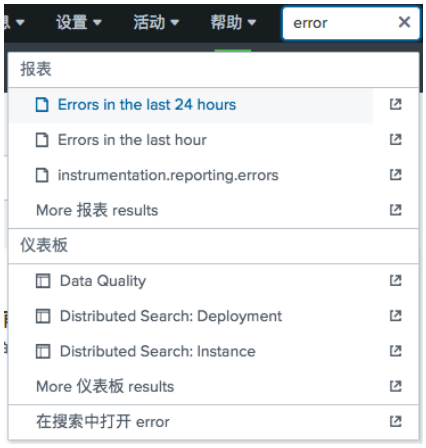
- 单击**任务**可打开搜索任务管理器窗口，您可以通过该窗口查看和管理当前运行的搜索。
- 单击**触发的告警**可查看已触发的计划的告警。

帮助

单击"**帮助**"可查看指向视频教程、Splunk Answers、Splunk 支持门户和在线文档的链接。

查找

使用**查找**在 Splunk 部署内搜索对象。**查找**执行的是在已保存对象中 ID、标签和描述不区分大小写的匹配。例如，如果您键入 **error**，它会返回包含术语 "error" 的所有保存的对象。



这些已保存对象包括**报表**和**仪表板**。结果显示在列表中，通过所存在的类别进行分隔。

通过单击**打开搜索中的错误**，您还可在**搜索报表**应用中运行搜索**错误**。

另请参阅

使用 Splunk 搜索

关于搜索语言

Splunk **搜索处理语言** (SPL) 包括所有搜索命令及其函数、参数和子句。搜索命令会指示 Splunk 软件如何处理从索引中检索到的事件。例如，您需要使用某一命令筛选不需要的信息、提取更多的信息、评估新字段、计算统计信息、重新排序结果或创建图表。

有些搜索命令拥有与其关联的函数和参数。您可以利用这些函数及其参数来指定命令如何操作结果和命令操作哪些字段。例如，可使用函数来设置图表中的数据格式、描述要计算的统计信息类型以及指定要评估的字段。有些命令还使用子句来指定对搜索结果的分组方式。

要熟悉 SPL，请阅读本手册中的以下主题：

- 搜索类型

- 命令类型
- 使用 Splunk 搜索

有关 SPL 语法的更多信息，请参阅*《搜索参考》*中的“了解 SPL 语法”。

有关函数的信息，请参阅

- *搜索参考*中的评估函数。
- *《搜索参考》*中的“统计和图表函数”

搜索类型

当您进行搜索时，您将开始识别模式和更多可作为可搜索字段的信息。您可在为新数据创建索引时对 Splunk 软件进行配置以使其识别这些新字段，也可以在搜索期间创建新字段。无论您了解哪些内容，都可以在事件数据中使用、添加和编辑关于该知识的字段、事件和交易。捕获此类知识将有助于您构建更有效的搜索并生成更详细的报表。

在深入研究搜索的语言和语法之前，您应该先问自己要试图实现什么目标。通常，在将数据导入到 Splunk 部署之后，您会希望：

- 进行调查，以了解有关刚刚创建索引的数据的详细信息或找出问题的根源。
- 将搜索结果汇总为报表（表格形式或其他可视化形式均可）。

鉴于此，您可能会听说过我们提到以下两种搜索类型：原始事件搜索和转换搜索。

原始事件搜索

原始事件搜索是只从一个或多个索引中检索事件，通常适用于需要分析问题的搜索。此类搜索的示例包括：检查错误代码、关联事件、调查安全问题和分析故障。这些搜索通常并不包含搜索命令（`search` 本身除外），而且结果通常是一个原始事件的列表。

- 请阅读有关原始事件搜索的更多信息，从“关于检索事件”开始。

转换搜索

转换搜索是对一组结果执行某种类型的统计计算的搜索。在此类搜索中，首先从索引中检索事件，然后将这些事件传递给一个或多个搜索命令。这些搜索始终要求为必填字段，并且至少为统计命令组中的一个统计命令。示例包括：获取错误事件的每日数量、计算特定用户的登录次数或计算字段值的 95%。

- 请阅读“关于搜索处理语言语法”中有关搜索结构的更多信息。
- 请阅读“关于子搜索”，了解使用子搜索筛选结果的更多信息。
- 请阅读有关转换搜索和命令的更多信息，从“关于转换命令和搜索”开始。

信息密度

无论是要检索原始事件还是构建报表，都应当同时考虑是对*稀疏*信息还是对*密集*信息运行搜索：

- **稀疏搜索**是指查找单个事件或不经常出现在大型数据集中的事件。您可能听过这些搜索被称为“大海捞针”或“罕见术语”搜索。此类搜索示例包括：搜索特定和唯一的 IP 地址或错误代码。
- **密集搜索**是指对许多事件执行扫描并制作报表。此类搜索示例包括：计算所发生错误的数量或从特定主机中查找所有事件。

请参阅*《容量规划手册》*中的“搜索类型如何影响 Splunk Enterprise 性能”。

命令类型

在您了解 Splunk SPL 时，可能会听到用于描述搜索命令类型的术语：流、生成和转换。此主题说明这些术语的含义并列出属于每种类别的命令。

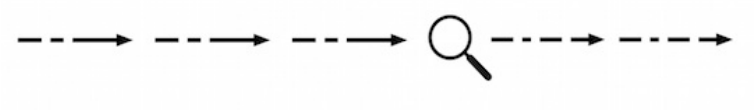
几乎所有搜索命令都可分为四大类：可分配流命令、集中流命令、转换命令和生成命令。这些分类并不互相排斥。有些命令只适用于一类命令。`stats` 命令是仅适用于转换分类的命令。其他命令可以适用于多个分类。例如，某个命令可以是流命令，同时也可以是生成命令。

还有一些命令不属于这些类别。这些命令在本主题末尾进行介绍。

若需每种类型的完整命令列表，请参阅*《搜索参考》*中的“命令类型”。

流命令和非流命令

流命令作用于搜索返回的每个事件。本质上是，一个事件进来，然后一个事件（或没有事件）出去。



例如，`eval` 命令可以创建一个名为 `full_name` 的新字段，以串联的方式包含 `first_name` 字段值、一个空格和 `last_name` 字段值。

```
... | eval full_name = first_name." ".last_name
```

`eval` 命令评估每个事件，而不会考虑其他事件。

非流命令在作用于整个事件集之前，要求必须收到所有索引器的事件。有些转换命令是非流命令。有些其他命令不是转换命令，但还是非流命令。有时这些都称为基于事件的非流命令。

例如，在 `sort` 命令能开始排序事件之前，`sort` 命令要求必须收到整个事件集。非流命令的其他示例包括 `dedup`、`stats` 和 `top`。

非流命令将整个事件集强制推到搜索头。这要求大量的数据移动并会失去并发性。

有关如何减少非流命令成本的信息，请参阅本手册中的"编写更好的搜索"。

处理属性

下表介绍了某些类型命令之间的处理差异。

	可分配流	集中流	基于事件的非流	转换
可运行于索引器	Y	N	N	N
可于最终输入前输出	Y	Y	N	N
若输入为事件则输出事件	Y	Y	Y	N

运行命令时，会根据命令类型输出事件或结果。例如，运行 `sort` 命令时，输入为事件，输出是按照您指定的顺序排序的事件。但是，转换命令不输出事件。转换命令输出结果。例如，`stats` 命令将输出计算结果表格。用于计算这些结果的事件不再可用。运行转换命令后，您无法运行将事件当做输入的命令。

基于事件的非流命令是不适用于四大类命令类型的命令。`stats` 命令是基于事件的非流命令。请参阅"其他命令"。

可分配流

作用于搜索返回的每个事件的流命令。对于可分配流命令，事件顺序无关紧要。可分配流命令是指可运行于索引器上的命令，这种方式可提高处理时间。搜索中的其他命令确定可分配流命令是否在索引器上运行：

- 如果可分配流命令之前的所有命令都可在索引器上运行，则此可分配流命令就在索引器上运行。
- 如果可分配流命令之前的命令中有一个必须在搜索头上运行，则此搜索中的其他命令都**必须**在搜索头上运行。当搜索处理移动到搜索头时，则无法移动回索引器。

可分配流命令可并行应用于索引数据的子集。例如，`rex` 是一个流命令。它会在搜索时间提取字段并将字段添加到事件。

部分常用可分配流命令有：eval、fields、makemv、rename、regex、replace、strcat 和 where。

若需可分配流命令的完整列表，请参阅*《搜索参考》*中的"流命令"。

集中流

对于集中流命令，事件顺序很重要。集中流命令将转换应用到搜索返回的每个事件。但与可分配流命令不同的是，集中流命令仅作用于搜索头。您可能还听说过描述这些命令的"状态流"这一术语。

集中流命令包括：head、streamstats、dedup 的一些模式以及群集的一些模式。

转换

转换命令会对搜索结果进行排序并生成数据表格。该命令将每个事件的指定单元格值"转换"成可供 Splunk 软件用于统计的数字值。转换命令不是流命令。如果要将搜索结果数据转换为可视化需要的数据结构（如柱形图、条形图、

折线图、面积图和饼图），则需要使用转换命令。

转换命令包括：chart、timechart、stats、top、rare、contingency、highlight、typer 和 addtotals（用于计算列总计时，非行总计）。

有关转换命令和他们在创建统计表格和图表可视化中所扮演角色的更多信息，请参阅本手册中的"关于转换命令和搜索"。

若需转换命令的完整列表，请参阅*《搜索参考》*中的"转换命令"。

生成

生成命令从索引获取信息，且不会经过任何转换。生成命令为事件生成（可分配或集中）或报表生成。大部分报表生成的命令也都是集中命令。结果会以列表或表格的形式返回，具体取决于命令类型。

生成命令不要求也不需要输入。通常在搜索开始时通过前导管道符调用生成命令。即，不可存在通过管道符传递给生成命令的搜索。例外情况是 `search` 命令，因为搜索命令在搜索开始时为隐式，无需调用。

生成命令的示例包括：dbinspect、datamodel、inputcsv、metadata、pivot、search 和 tstats

若需生成命令的完整列表，请参阅*《搜索参考》*中的"生成命令"。

安排

安排命令是控制如何处理搜索某些方面的命令。此命令不会直接影响搜索及的最终结果。例如，您可以对搜索应用安排命令，以启用或禁用可有助于快速完成整个搜索的搜索优化。

安排命令示例包括重新分布和 noop。

其他命令

还有一些命令不属于这些类别。一些命令仅适用于特定情况下的分类。

这些命令既不属于转换、可分配，也不属于流命令：sort、eventstats、群集的一些模式、dedup 和 fillnull。

使用 Splunk Web、CLI 或 REST API 搜索

您可以通过 Splunk Web 和 Splunk REST API 执行搜索。如果您使用的是 Splunk Enterprise，也可以通过命令行界面 (CLI) 运行搜索。究竟哪个工具最合适有时取决于要搜索的内容。

如果您需要能在单个搜索中一起搜索 Splunk Enterprise 和 Splunk Cloud 部署，则必须配置混合搜索。请参阅 *Splunk Cloud 用户手册*中的"配置混合搜索"。

使用 Splunk Web 搜索

使用 Splunk Web 进行搜索时，您使用的是搜索应用，并可通过选择搜索模式（快速、详细、智能）来控制搜索体验。Splunk 软件会根据所选的模式自动发现并提取非默认字段，以事件列表或表格形式返回结果，并运行必要的计算以生成事件时间线。计算事件时间线需要大量的成本，因为这需要创建数据桶并将事件和字段的统计信息保留在 dispatch 目录中，以便当用户单击时间线上的某个栏位时，即可使用这些统计信息。

- 请阅读本手册中的如何"设置搜索模式以调整搜索体验"，了解更多信息。

使用 CLI 或 REST API 搜索

通过命令行界面 (CLI) 运行搜索或使用 REST API 中的搜索任务端点创建搜索时，此搜索将直接转到 Splunk 搜索引擎，而不经过 **Splunk Web**。这些搜索的完成速度要比 Splunk Web 中的快很多，因为 Splunk 软件不计算或生成事件时间线。而是将搜索结果显示为一个原始事件列表或一个表格，具体取决于搜索类型。

- 请阅读*《搜索参考》*中的"关于 CLI 搜索"以了解更多信息。
- 请阅读*《REST API 参考手册》*中的"使用 REST API 创建搜索"。

使用搜索应用

关于搜索应用

搜索和报表应用，简称为*搜索应用*，是一个可用于搜索数据并创建相应报表的应用程序。

本主题介绍搜索应用所包含的视图和元素。

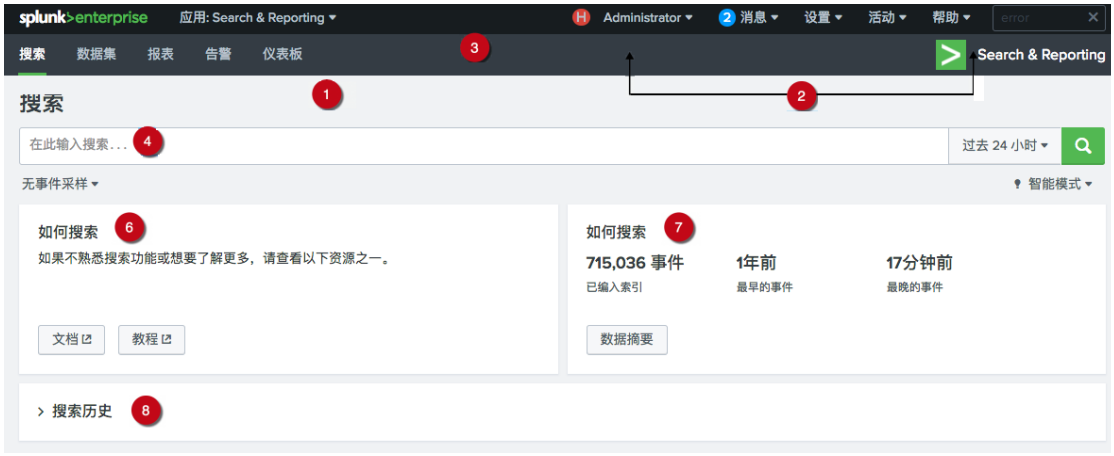
打开搜索应用

1.从 Splunk Home 上，在**应用**面板中单击**搜索和报表**。这一操作会打开搜索和报表应用中的“搜索摘要”视图。

搜索摘要视图

在您运行搜索之前，搜索摘要视图中会显示以下元素：应用栏、搜索栏、时间范围挑选器、**如何搜索**面板、**搜索内容**面板，以及**搜索历史**。

有些是您在其他视图中可以看到的常规元素。面板还包括搜索摘要视图特定的元素，这些元素位于搜索栏下方：**如何搜索**面板、**搜索内容**面板及**搜索历史**面板。



数字	元素	描述
1	**应用程序菜单**	在已安装的 Splunk 应用程序间进行切换。当前列出“搜索和报表”应用程序。此菜单位于 Splunk 栏中。
2	**Splunk 栏**	编辑 Splunk 配置，查看系统级别的消息，获取产品使用帮助。
3	**应用栏**	在当前应用程序中不同视图间进行导航。搜索和报表应用中的视图有：搜索、数据集、报表、告警和仪表板。
4	**搜索栏**	指定搜索条件。
5	**时间范围挑选器**	指定搜索时间段，例如过去 30 分钟或昨天。默认时间范围为**过去 24 小时**。
6	**如何搜索**	包含至*搜索手册*和*搜索教程*的链接。
7	**搜索内容**	显示此 Splunk 实例上已上载的、您有权查看的数据的摘要。
8	**搜索历史**	可查看已运行搜索的列表。在您运行首次搜索后，即会显示搜索历史。

数据摘要

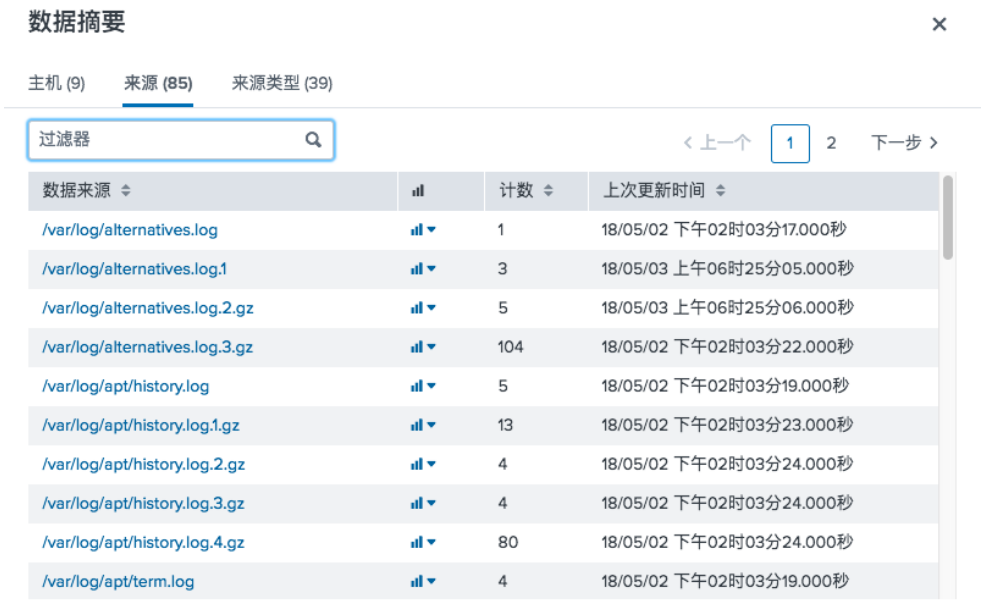
数据摘要对话框显示三个选项卡：“主机”、“来源”、“来源类型”。这些选项卡代表您数据中的可搜索字段。

Host

事件的**主机**是事件来源的网站计算机的主机名、IP 地址或完全限定域名。在分布式环境中，您可以使用主机字段从特定计算机中搜索数据。

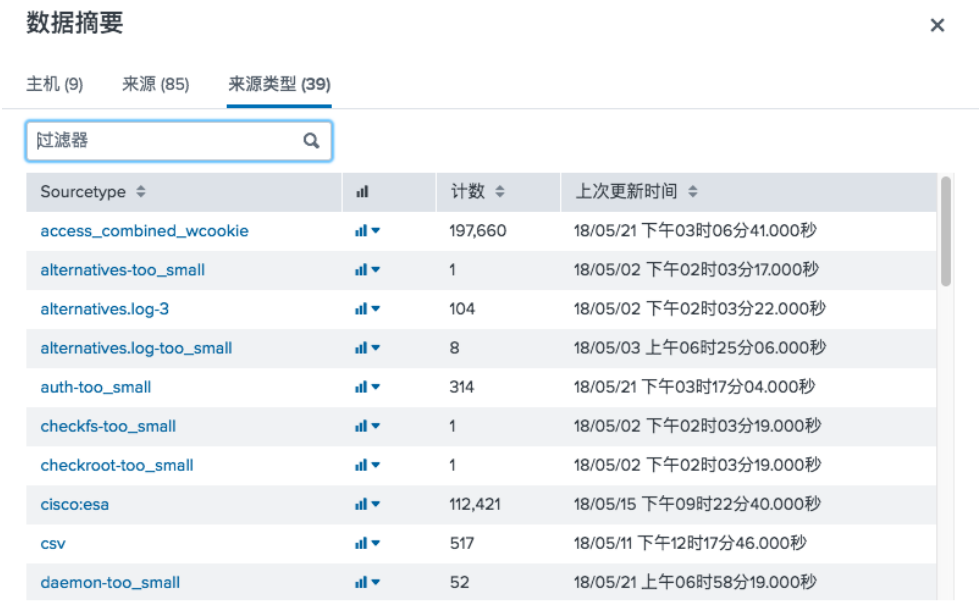
数据来源

事件的**数据来源**是文件或目录路径、网络端口或生成事件的脚本。



来源类型

事件的**来源类型**告诉您数据是哪种类型的数据，通常基于数据的格式设定方式。这种分类允许您跨多个数据来源和主机搜索相同类型的数据。



在本例中，来源类型为：

- **access_combined_wcookie:** Apache Web 服务器日志
- **secure:** 安全服务器日志
- **vendor_sales:** 全球销售供应商

若需了解分配给数据的来源类型，请参阅*数据导入*手册中的“来源类型为何重要”。

新搜索视图

在您运行搜索之后或当您单击**搜索**选项卡以开始新搜索时，会打开**新搜索**视图。应用栏、搜索栏和时间范围挑选器在此视图中仍然可用。另外，此视图还包含了许多其他元素：搜索操作按钮和搜索模式选择器、事件计数、任务状态栏以及“事件”、“模式”、“统计”和“可视化”选项卡。

您可在搜索栏中键入 `index=_internal` 并按 **Enter** 键，以查看来自 Splunk 实例内部日志文件的事件。

如果您按照*搜索教程*中的步骤将数据导入了 Splunk 部署，则可在搜索栏中键入 `buttercupgames` 并按 **Enter** 键，以便在您的事件中搜索关键字 "buttercupgames"。

在此视图中，也可使用应用栏、搜索栏和时间范围挑选器。**新搜索**视图包含许多其他元素，例如搜索操作按钮、搜索模式选择器、事件计数、任务状态栏以及“事件”、“模式”、“统计”和“可视化”选项卡。

应用栏

使用应用栏在搜索和报表应用的不同视图之间导航：搜索、数据透视表、报表、告警和仪表板。以下手册专门介绍了这些其他操作。

- *告警手册*
- *仪表板和可视化*
- *数据透视表手册*
- *报表手册*

搜索栏

使用搜索栏可在 Splunk Web 中指定搜索标准。输入搜索字符串并按 **Enter**，或单击搜索栏右侧的搜索图标。

时间范围挑选器

时间是您指定的单个最重要搜索参数。

使用时间范围挑选器可检索特定时间段内的事件。对于**实时搜索**，可以指定检索事件的窗口。对于**历史搜索**，可以通过指定相对时间范围（15 分钟以前、昨天等等）对搜索加以限制。您还可以使用具体日期和时间范围对搜索加以限制。时间范围挑选器提供了许多可供选择的预设时间范围，但您也可以键入自定义时间范围。

有关更多信息，请参阅“关于涉及时间的搜索”。

时间线

时间线是在结果中每个时间点上发生的事件数的虚拟表示。时间线中的高值或低值表示活动高峰或服务器停机。时间线选项位于时间线上方。您可放大、缩小及更改图表刻度。

当您单击时间线上的点或使用一个时间线选项，则时间线显示会根据搜索返回的事件更改。新搜索没有运行。

搜索操作

您可以执行的搜索操作有很多，包括执行搜索任务、保存、共享、导出及打印搜索结果等。

有关更多信息，请参阅：

- 对正在运行的搜索执行操作
- 关于任务和任务管理
- 导出搜索结果

搜索模式

您可以使用搜索模式选择器来根据您的需求提供搜索体验。模式包括智能（默认值）、快速和详细。

有关更多信息，请参阅“搜索模式”。

字段边栏

事件列表的左侧为字段边栏。当检索匹配搜索的事件时，事件中的“字段”边栏会显示**已选字段**和**感兴趣的字段**。这些是 Splunk 软件从您的数据中提取的字段。

首次运行搜索时，**已选字段**列表中显示默认字段：host、source 和 sourcetype。每个事件中都会显示默认字段。

感兴趣的字段是出现在至少 20% 的事件中的字段。

字段名称旁边是出现字段名称的事件数。单击任一字段名称显示有关该字段的更多信息。

搜索剖析

搜索由用管道符（|）分隔的一系列命令组成。每个管道符之后的第一个用空白分隔的字符串用于控制所使用的命令。每个命令的其余文本会以特定于给定的命令方式进行处理。

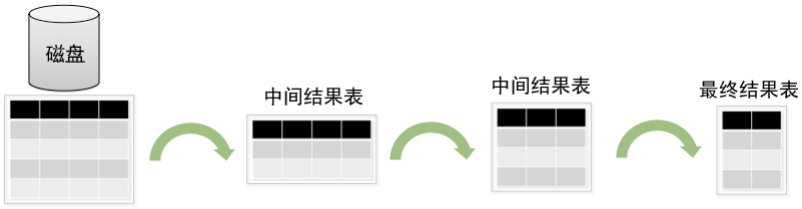
此主题介绍 Splunk 搜索剖析，以及字段和字段值的每个命令和语法规则所共享的一些语法规则。

搜索剖析

它可帮助您以表格形式直观地显示所有索引的数据，以便更好地了解搜索命令操作数据的方法。每个搜索命令都会重新定义表格的形状。

例如，让我们来看看下面的搜索。

```
sourcetype=syslog ERROR | top user | fields - percent
```



磁盘表示您所有的索引数据。磁盘是一个有固定大小的表，其中列代表字段，行代表事件。第一个中间结果表显示的行较少，表示从索引中检索到的与搜索术语 "sourcetype=syslog ERROR" 匹配的事件的子集。第二个中间结果表显示的列较少，表示最高命令（即 "top user"）的结果，此命令将事件汇总为一个包含前 10 位用户的列表并显示用户、计数和百分比。然后，"fields - percent" 会删除显示百分比的列，因此，剩下的最终结果表就变得更小了。

关于搜索管道

“搜索管道”是指 Splunk 搜索的结构，在此结构中，多个连续命令通过管道符 "|" 链接在一起。管道符指示 Splunk 软件使用一个命令（位于管道符左侧）的输出或结果作为下一个命令（位于管道符右侧）的输入。这样，您就能够沿着管道限制或增加每一步的数据，直至获得所需的结果。

Splunk 搜索从管道开头的搜索术语开始执行。这些搜索术语包括关键字、短语、布尔表达式、键/值对等，用于指定要从索引中检索的事件。请参阅“关于检索事件”。

之后可以使用管道符将检索的事件以输入形式传递搜索命令。搜索命令会指示 Splunk 软件在从索引中检索到事件之后如何处理事件。例如，您可能使用命令筛选不需要的信息、提取更多的信息、评估新字段、计算统计信息、重新排序结果或创建图表。有些命令拥有与其关联的函数和参数。您可以利用这些函数及其参数来指定命令如何操作结果以及要操作哪些字段；例如，如何创建图表、要计算何种类型的统计信息以及要评估哪些字段。有些命令还允许您使用子句来指定对搜索结果的分组方式。

- 有关使用搜索命令可以完成的工作的更多信息，请参阅“关于搜索处理语言”。
- 在*《搜索参考》*中，关于搜索命令列表，请参阅“命令快速参考”和关于语法和使用信息的单独搜索命令参考主题。

引号和转义字符

通常，需要在含有空格、逗号、管道符、引号或方括号的短语和字段值两边加上引号。引号必须平衡，起始引号后必须跟随非转义结束引号。例如：

- 诸如 `error | stats count` 之类的搜索将查找含有字符串错误的事件的数量。
- 诸如 `... | search "error | stats count"` 之类的搜索将按顺序返回含有文字字符串 "error"、管道符、stats 和 count 的原始事件。

另外，如果您不想搜索关键字和短语的默认含义（例如布尔运算符和字段/值对），请在这些关键字和短语两侧加上引号。例如：

- 搜索不表示布尔运算符的关键字 AND：`error "AND"`
- 搜索此字段/值短语：`error "startswith=foo"`

反斜杠字符 (\) 用于转义引号、管道符及其本身。仍可在引号内部扩展反斜杠转义序列。例如：

- 如果在搜索中使用 \| 序列，则会将管道符发送到命令，而不是用管道符来拆分各个命令。
- \" 序列将向命令发送引号字符，例如，搜索引号字符或使用 rex 将引号字符插入字段中。
- \\ 序列可在命令中用作反斜杠字符。

如果反斜杠序列无法为 Splunk 软件识别，就无法修改。

- 例如，搜索字符串中的 \s 可以 \s 形式发送给命令，因为 \s 属于未知的转义序列。
- 但是，在搜索字符串中，\\s 可以 \s 形式发送给命令，因为 \\ 属于已知的转义序列，它将被转换为 \。

使用反斜杠转义字符时不能搜索星号 *。Splunk 软件将星号字符视为主分割符。因此，星号永远不会出现在索引中。如果您要搜索星号字符，需要在您的数据上运行后期筛选 regex 搜索：

```
index=_internal | regex ".*\*.*"
```

有关主分割符的更多信息，请参阅《数据导入手册》中的“事件处理概述”。

示例

示例 1： `myfield` 字段由值 `6` 创建。

```
... | eval myfield="6"
```

示例 2： `myfield` 字段由值 `"` 创建。

```
... | eval myfield="\""
```

示例 3： `myfield` 字段由值 `\` 创建。

```
... | eval myfield="\\"
```

示例 4： 由于引号不平衡，此搜索会产生错误。

```
... | eval myfield="\"
```

字段

流经 Splunk 搜索管道的事件和结果以字段集合的形式存在。字段基本上来自 Splunk 索引（如表示事件时间的 _time、表示文件名的数据来源数据等），字段也可能源自搜索时间的广泛数据来源（如事件类型、标记、使用 `rex` 命令提取的正则表达式，来自 `stats` 命令的总数等）。

对于给定事件，给定字段名称可能存在，也可能不存在。如果存在，它可能包含单个值或多个值。每个值都是一个文本字符串。值的长度可能为正值（字符串或文本）或为零（空字符串或 ""）。

例如，某些数字是包含该数字的字符串。例如，含有数字值 10 的字段包含字符 1 和 0："10"。从值中获取数字的命令会自动在内部将这些值转换为用于计算的数字。

空值字段
: 对于特定结果或事件，不存在空值字段。在同一搜索中，其他事件或结果的此字段可能具有值。例如，`fillnull` 命令将向事件或结果添加字段和默认值，这些事件或结果缺少搜索中其他事件或结果所包含的字段。

空字段
: 空字段是包含单个值（即空字符串）的字段的缩略名。

空值
: 空字符串或 "" 的值。也可将其描述为长度为零的字符串。

多值字段
: 具有多个值的字段。所有非空值字段都包含一个有序的字符串列表。通常情况下，这是含有一个值的列表。如果列表中包含多个条目，我们称其为多值字段。请参阅*《搜索手册》*中的“操作和评估多值字段”。

协助构建搜索

Splunk 搜索处理语言 (SPL) 包含可用于构建搜索的命令和函数。所有这些命令和函数都在*《搜索参考》*中有所记录。

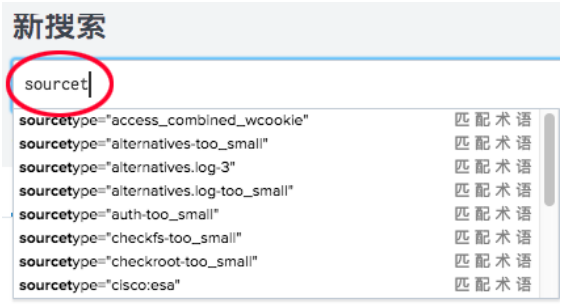
Splunk Web 有几个可帮您构建和分析搜索的内置功能。

- 搜索助理模式
- 语法突出显示
- 自动格式搜索语法
- 为搜索行编号
- 快捷方式

本主题讨论搜索助理的使用。有关语法突出显示、自动格式、行号和快捷方式的信息，请参阅“协助读取搜索”。

使用搜索助理构建搜索

当您在搜索栏中键入几个字母或术语时，搜索助理会将与这些输入内容相匹配的术语和搜索罗列出来。



匹配术语是基于根据您的数据建立了索引的术语。**匹配搜索**是基于您最近的搜索。

此列表会随着您的输入不断更新。

若要将此列表的某个项目添加到您的搜索标准中，可以单击此项目，或使用箭头键突出显示此项目，然后按 **Enter**。

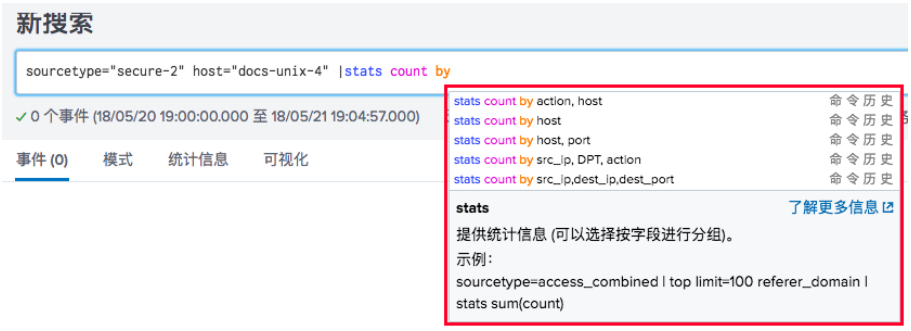
搜索助理模式

搜索助理有三种模式：完整、紧凑和无。默认是紧凑模式。

紧凑模式

在紧凑模式下，搜索助理会随着您的输入列出匹配的术语和搜索。当您键入一个管道符（|）表示您想使用命令时，则会显示 SPL 命令列表。您可以键入一个字母以跳至此列表中以该字母开头的部分。例如，如果您输入字母 **s**，则此列表会显示所有以字母 **s** 开头的命令。

当您输入命令时，会显示一个列表，列出命令历史和匹配搜索。最初，命令历史会显示一些命令示例。随着您在搜索中开始使用命令，命令历史会显示您对此命令的使用情况，而不再显示示例。



在列表下方会给出此命令的简要描述和示例。**了解更多**链接会在新窗口中打开*《搜索参考》*，并显示此命令的记录。

要通过键盘访问**了解更多**链接，请使用箭头键突出显示命令或属性名称。按**选项卡**突出显示**了解更多**链接，然后按 **Enter** 激活链接。

如果您在命令之后输入了一些内容，搜索助理会显示与您的输入相匹配的所有命令参数或历史。

搜索助理也可以显示参数所需的数据类型。在搜索栏中键入参数。如果等于符号（=）是参数语法的一部分，则应将其包含在内。在以下示例中，搜索助理显示 `countfield` 参数必须要有一个 <string> 值。

完整模式

在完整模式下，搜索助理会随着您的输入列出匹配的术语和搜索，同时给出术语在您的索引数据中出现的次数计数。此计数告诉您，如果您要搜索此术语的话会返回多少搜索结果。如果此列表中未包含某术语或短语，则表示您的索引数据中不包含此术语。

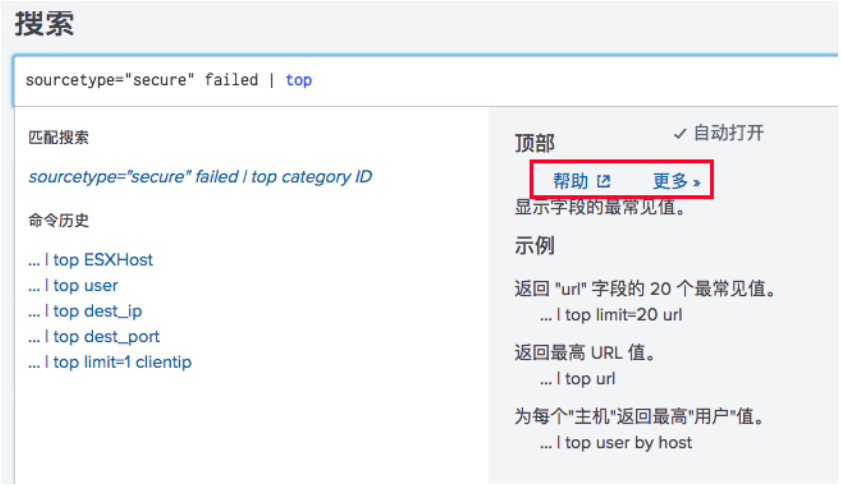
完整模式也会在“如何搜索”部分给出建议，说明可以通过哪些方式检索事件和使用搜索命令。

当您在搜索栏中输入命令时，会显示匹配术语和搜索的列表，而不是**命令历史**列表。

若要将命令历史列表中的某个项目添加到您的搜索标准中，可以单击此项目，或使用箭头键突出显示此项目，然后按**Enter**。

搜索助理会显示此命令的简要描述并提供一些示例。在命令描述旁边会有两个链接：帮助和更多。

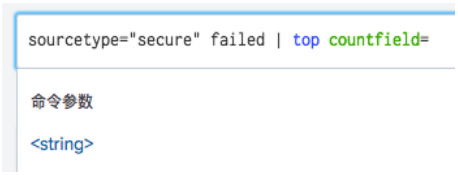
- **帮助**链接会在新窗口中打开*搜索参考*，并显示此命令的记录。
- **更多**链接会针对显示在屏幕上的命令提供更多扩展信息。



当点击**更多**链接时，会显示几个新部分。**详细信息**部分提供了此命令的具体描述。**语法**部分给出此命令的基本语法。**相关命令**部分列出与您输入的命令相关的其他命令。如果此命令的语法很复杂，单击其语法旁边的**更多**链接展开其语法。

如果您在命令之后输入了一些内容，搜索助理会显示与您的输入相匹配的所有命令参数或历史。

搜索助理可显示参数所需的数据类型。在搜索栏中键入参数。包含等于符号（=），如果它是参数语法的一部分。在以下示例中，搜索助理显示 `countfield` 参数必须要有一个 <string> 值。



无模式

您可将模式更改为**无**模式来关闭搜索助理。

更改搜索助理模式

搜索助理的默认模式是紧凑模式。您可以更改搜索助理模式或在构建搜索时暂时隐藏搜索助理。

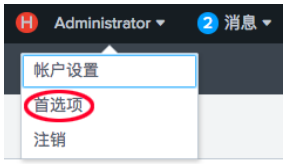
当更改搜索助理模式时，此更改仅会对您的用户帐户产生影响。

前提条件

如果搜索栏包含尚未运行的搜索，则在更改搜索助理模式之前要先运行此搜索。否则，在您更改模式之后此搜索会丢失。运行此搜索可将其添加到搜索历史中。在更改模式之后，您可以通过搜索历史访问此搜索。

步骤

1. 在 Splunk 栏中，选择 **[*用户帐户名*] > 首选项**。



2. 单击 **SPL 编辑器**。
3. 验证**高级编辑器**是否打开。
4. 对于搜索助理，单击您想要使用的模式：**完整**、**紧凑**或**无**。

5. 单击**应用**。

隐藏和显示搜索助理

默认情况下，当您在搜索栏中进行输入时搜索助理即会打开。您可以关闭或隐藏搜索助理。

关闭搜索助理

要关闭搜索助理，请将搜索助理模式更改为**无**。

隐藏搜索助理

隐藏搜索助理选项取决于您使用的模式。

紧凑模式
: 您无法隐藏搜索助理。您只能关闭搜索助理。

完整模式
: 要隐藏“完整”模式中的搜索助理，您可以关闭**自动打开**功能并折叠搜索助理下拉列表。

1. 在搜索助理窗口中，单击**自动打开**。这一操作会移除**自动打开**旁边的勾选标记。
2. 单击“搜索”栏右侧的折叠和展开按钮隐藏搜索助理。

   搜索助理保持隐藏状态，直到您使用展开按钮重新显示搜索助理。请参阅本主题中的“取消隐藏搜索助理窗口”。

当您取消勾选**自动打开**并单击折叠按钮时，将隐藏搜索助理，即使您已开始新搜索或关闭并重新打开 Splunk Web。搜索助理会保持隐藏状态，直到您取消隐藏。

取消隐藏搜索助理

如果搜索助理隐藏，请单击搜索栏右侧的展开按钮，然后单击**自动打开**。

如果以上步骤没有取消隐藏搜索助理窗口，则要么是搜索助理关闭了，要么是您所输入搜索栏的内容没有任何相关协助内容。

要重新开启搜索助理，则需要将搜索助理模式改为**紧凑**或**完整**。

为所有用户更改默认搜索助理模式

个别用户可以更改自己的默认搜索助理模式。也可以为所有用户全局更改默认搜索助理模式。

前提条件

- 只有具有文件系统访问权限的用户，如系统管理员才能为所有用户更改默认搜索助理模式。
- 请参阅*《管理员手册》*中的“如何编辑配置文件”了解具体步骤。

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。在本地目录进行更改。

步骤

1. 打开搜索应用的本地 `user-prefs.conf.spec.in` 文件。例如，`$SPLUNK_HOME/etc/apps/<app_name>/local`。
2. 在 **[general]** 段落中，通过选择一种其他模式值更改搜索助理模式。选择**完整模式**、**紧凑模式**或**无模式**。例如：`search_assistant=full`。
3. 重新启动 Splunk 实例。

协助搜索的阅读

搜索字符串可能很长，不容易读取。搜索栏包含一些功能，可协助您阅读、解析或解释 Splunk 搜索处理语言 (SPL) 语法。语法突出显示功能会以不同的颜色显示不同的 SPL 部分。语法高亮显示有两种不同的颜色主题。

除了颜色主题，您还可以使用自动格式和行号功能读取搜索。可以使用键盘快捷方式查找搜索语法中的信息。

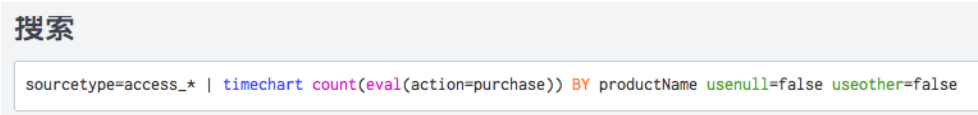
语法突出显示

有了语法突出显示功能，SPL 命令、参数、函数和关键字都有各自的颜色代码，简化了搜索的读取。

考虑以下搜索。

```
sourcetype=access_* | timechart count(eval(action=purchase)) BY productName usenull=false useother=false
```

当语法突出显示功能打开时，此搜索很容易读取。语法突出显示会以不同的颜色显示命令、参数、函数和关键字。下图表示带语法突出功能的搜索字符串。



语法验证

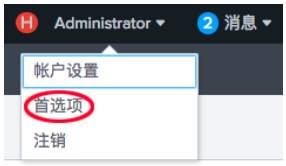
如果命令、参数、函数或布尔运算符的拼写或大小写不正确，则术语不会以相应颜色突出显示。不以颜色显示表示告警：语法不正确。

如果您为参数指定了的数据类型不正确，则值为红色。例如，`top` 命令的 `limit` 参数需要为整数。如果您输入 `...|top limit=false`，则术语 `false` 会以红色突出显示，因为它不是一个整数。

关闭语法突出显示

您可以通过将颜色主题更改为**白底黑字**来关闭语法突出显示。对于无法区分不同颜色的人，这个功能很有用。

1. 在 Splunk 栏中，选择 ***[用户帐户名]*** > **首选项**。



2. 单击 **SPL 编辑器**。
3. 在**主题**选项卡中，单击**白底黑字**。

4. 单击**应用**。

颜色主题

您可以通过指定颜色主题改变搜索条件外观。有几种主题可供选择。

主题名称	描述	注释
白底黑字	白色背景。黑色文本。无其他颜色。	对于无法区分不同颜色的人，这个功能很有用。
淡色主题	白色背景。黑色文本。命令、参数、函数、关键字修饰符和布尔运算符颜色。	默认主题
深色主题	黑色背景。淡灰文本。命令、参数、函数、关键字修饰符和布尔运算符颜色。	

颜色代码

搜索语法中所使用的颜色代码由执行的颜色主题确定。默认主题为**淡色**主题。下表描述了**淡色**主题和**深色**主题的颜色代码。

语法组件	颜色	示例
命令	蓝色	...timechart

命令参数	绿色	...timechart usenull=false
函数	粉色	...timechart count
关键字编辑器和布尔运算符	橙色	...timechart count BY productName

下图显示了以**深色**主题突出显示的语法。

更改颜色主题

您可以使用帐户菜单更改“搜索”栏中的颜色主题。

1. 在 Splunk 栏中，选择 **[*用户帐户名*] > 首选项**。
2. 单击 **SPL 编辑器**。
3. 在**主题**选项卡中，选择您想要使用的颜色主题。
4. 单击**应用**。

自动格式搜索语法

构建搜索之后，您可以设置 Splunk 软件将搜索语法格式设为键入的格式。自动格式使您的搜索可读性更高。每个管道部分都会分析为单独一行。每个子搜索都会缩进。

下图显示了关闭自动格式之后，搜索如何出现在“搜索”栏中。

打开自动格式之后，按照下图所示分析相同搜索。

触发自动格式的字符

字符	自动格式
管道（|）	将管道放在新行中，以划分搜索条件中的每个新管道部分。
左方括号（[）	将左方括号（代表子搜索的开始）放在新行中并缩进几个空格。

如果管道或左括号在带引号的字符串里面，则表示没有触发自动格式。

打开搜索自动格式

默认关闭搜索语法的自动格式。您可以在“设置”对话框中打开搜索语法自动格式。

更改“设置”对话框中的选项仅针对您的设置做出更改。不会影响其他用户的设置。

1. 在 Splunk 栏中，选择 **[*用户帐户名*] > 首选项**。
2. 单击 **SPL 编辑器**。
3. 在**常规**选项卡中，单击**搜索自动格式**。

4. 单击**应用**。

对您键入“搜索”栏的新搜索应用自动格式。如果“搜索”栏中已经有搜索，请使用“搜索”栏快捷键对该搜索应用自动格式。

为什么我的搜索不能自动格式化？

自动格式功能仅对您在“搜索”栏中输入的搜索有效。如果您将搜索粘贴到“搜索”栏或从**搜索历史**中选择一个搜索，那么即使自动搜索功能已打开，该搜索也不会自动设定格式。

要对您粘贴到“搜索”栏或从搜索历史中选择的搜索应用自动格式，请使用以下键盘快捷方式将自动格式应用到搜索。

- 对于 Linux 或 Windows 系统，用 **Ctrl + **
- 对于 Mac OSX，用 **Command + **

为搜索行编号

要使搜索读取更容易，您可以在“搜索”栏显示行号。以下图像显示了行号和已打开的自动格式。

打开行号

默认情况下，行号处于关闭状态。您可以在“首选项”对话框中打开行号。

1. 在 Splunk 栏中，选择 **[*用户帐户名*] > 首选项**。
2. 单击 **SPL 编辑器**。
3. 在**常规**选项卡中，单击**行号**。
4. 单击**应用**。

更改**首选项**对话框中的选项仅针对您的设置做出更改。不会影响其他用户的首选项设置。请参阅“为所有用户更改默认搜索首选项”。

搜索栏中的一排并非一行

行编号功能只适用于行号。搜索栏中的一排不一定是一行。搜索栏中可能有一个很长的行跨多排，但是仍只是一行。

例如，如果您将一个很长的搜索复制到“搜索”栏，但复制之后的格式并非多行，那么该搜索会跨多排，但是只有一个

行号。

您可以使用以下方法在“搜索”栏中创建行。

- 打开“搜索”自动格式功能，输入管道符或左括号。
- 使用“搜索”栏快捷方式自动设置当前搜索格式。
- 按 **Shift + Enter** 拆分光标中的活动行。按 **Enter** 不会在“搜索”栏中创建新行。

搜索栏快捷方式

在搜索栏中，您可以使用键盘快捷方式来协助您编写、阅读和解析搜索标准。

让搜索更易于阅读

长搜索可能较难于读取。例如，以下搜索使用了多个命令并在搜索结果中包含了多个重命名列。

```
sourcetype=access_* status=200 | stats count AS views count(eval(action="addtocart")) AS addtocart
count(eval(action="purchase")) AS purchases by productName | eval viewsToPurchases=(purchases/views)*100 | eval
cartToPurchases=(purchases/addtocart)*100 | table productName views addtocart purchases viewsToPurchases
cartToPurchases | rename productName AS "Product Name", views AS "Views", addtocart as "Adds To Cart", purchases AS
"Purchases"
```

以下图像显示了此搜索显示在搜索栏中的样子。

您可以使用键盘快捷方式，将每个管道部分单独成为一排并进行解析。每个子搜索都会缩进。使用这些键盘快捷键不需要打开自动格式功能。

- 对于 Linux 或 Windows 系统，用 **Ctrl + **
- 对于 Mac OSX，用 **Command + **

以下图像显示了使用此快捷方式后的结果。

您还可以使用 **Shift + Enter** 强制创建新行。请参阅“排和字快捷方式”。

扩展搜索

对于较长的搜索或包含**搜索宏**或**已保存搜索**的搜索，可能很难在“搜索”栏中看到完整的搜索。

您可以使用键盘快捷方式看到完整搜索内容，在“搜索”页面使用“搜索”栏中的 **Command-Shift-E (Mac OSX)** 或 **Control-Shift-E（Linux 或 Windows）**。这会打开显示扩展搜索字符串的预览，包括所有搜索宏和保存的搜索。如果已打开语法突出显示或行编号，这些功能也会出现在预览中。

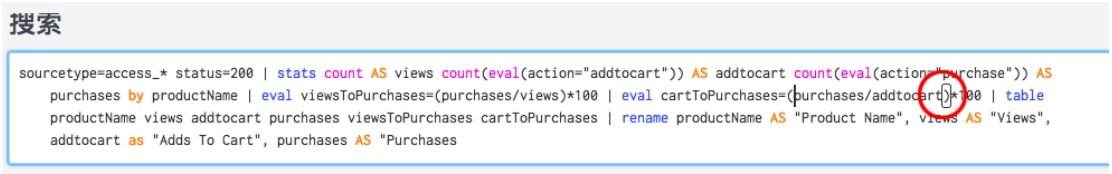
您可以复制预览窗口中的部分搜索。您还可以单击预览窗口中的**在搜索中打开**以在新窗口中运行搜索。请参阅“预览搜索”。

突出显示搜索术语

- 要在搜索中突出显示所有出现该词的地方，请双击该单词。

找到所匹配的括号

- 将光标放在左括号或右括号之后。所匹配的括号即会突出显示出来。



撤销或重做快捷方式

使用以下键盘快捷方式撤销和重做搜索栏中的操作。

操作	Linux 或 Windows	Mac OSX
撤销前一操作。	Ctrl + Z	Command + Z
重做前一操作。	Ctrl + Y 或 Ctrl + Shift + Z	Command + Y 或 Command + Shift + Z

搜索助理窗口快捷方式

当搜索助理处于紧凑模式时，可使用键盘快捷方式选取列表中的项目，以及关闭和重新打开搜索助理窗口。

操作	Linux 或 Windows	Mac OSX
将光标移到搜索助理窗口内。	向下箭头键	向下箭头键
关闭搜索助理窗口。	ESC	ESC
重新打开搜索助理窗口。	Ctrl + 空格	Control + 空格
选择搜索助理窗口中的一个项目并将其插入到搜索栏中。	使用向上箭头和向下箭头键突出显示该项目，然后按 **Enter**。	使用向上箭头和向下箭头键突出显示该项目，然后按 **Enter**。
在搜索助理窗口中**了解更多**链接和列表之间切换。	Tab	Tab

查找和替换快捷方式

使用以下键盘快捷方式查找和替换“搜索”栏中的术语。

操作	Linux 或 Windows	Mac OSX
查找术语。	Ctrl + F	Command + F
查找和替换术语。	Ctrl + H	Command + Option + F

排和字快捷方式

行和排的差异对于了解何时使用键盘快捷方式来操作搜索栏搜索条件中的行或排非常重要。

- 长搜索在搜索栏中会显示为多行。
- 如果搜索未进行分析，整个搜索是一排。
- 如果该搜索已进行分析，则每个管道部分和子搜索都将位于单独的行中，此时一行就等于一排。

操作	Linux 或 Windows	Mac OSX
拆分光标处的活动行。	Shift + Enter	Shift + Enter
删除活动排。如果此搜索是包含多行的一排且没有通过解析分为多个单独的排，则会删除整个搜索。	Ctrl + D	Command + D
复制活动行，并将其粘贴到活动行下方。	Alt + Shift + 向下箭头	Command + Option + 向下箭头
复制活动行，并将其粘贴到活动行上方。	Alt + Shift + 向上箭头	Command + Option + 向上箭头
将活动行向下移一行。	Alt + 向下箭头	Option + 向下箭头
将活动行向上移一行。	Alt + 向上箭头	Option + 向上箭头
将从光标处到行末端的搜索标准删除。	Alt + Delete	Control + K
将从光标处到行起始处的搜索标准删除。	Alt + Backspace	Command + Delete
删除光标右侧的字或空格。	Ctrl + Delete	Alt + Delete
删除光标左侧的字或空格。	Ctrl + Backspace	Option + Delete

为所有用户更改默认搜索首选项

个别用户可以更改自己的语法突出显示、自动格式和行编号功能的默认“搜索”首选项。

也可以为所有用户全局更改默认“搜索”首选项。

前提条件

- 只有具有文件系统访问权限的用户，如系统管理员才能为所有用户更改默认搜索首选项。如果您使用的是 Splunk Cloud 并想更改 Splunk 系统的默认“搜索”设置，请打开“支持提交问题”。
- 请参阅*《管理员手册》*中的“如何编辑配置文件”了解具体步骤。

不要更改或复制`default`目录中的配置文件。`default`目录中的文件必须保持原样并位于其原始位置。在`local`目录进行更改。

步骤

1. 打开搜索应用的本地 `user-prefs.conf.spec.in` 文件。例如，`$SPLUNK_HOME/etc/apps/<app_name>/local`。
2. 您可以在 [general] 段落中更改下表列出的设置。

功能	属性语法	默认设置
语法突出显示	search_syntax_highlighting = <boolean>	true
自动格式	search_auto_format = <boolean>	false
行编号	search_line_numbers = <boolean>	false

3. 重新启动 Splunk 实例。

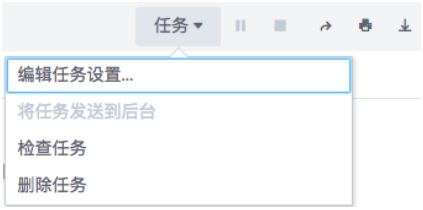
搜索操作

Splunk 软件提供了一组控件，可用于管理“进行中”搜索并创建报表和仪表板。

控制搜索任务的进度

在启动搜索后，您可以在不离开“搜索”视图的情况下访问和管理与搜索**任务**有关的信息。

1. 如果您的搜索处于运行中、暂停或完成状态，即可在搜索操作组中单击**任务**。



2. 从列表中选择选项。
   - **编辑任务设置。**打开“任务设置”对话框。您可在此对话框中更改任务的读取权限、延长任务的使用期限，以及获取任务的 URL。您可使用此 URL 与其他人共享任务或在 Web 浏览器中为任务添加一个书签。
   - **将任务发送到后台运行。**在后台运行此任务。如果搜索任务完成得很慢，使用此选项。这一操作使您可以处理其他活动，包括运行一个新搜索任务。
   - **检查任务。**打开“搜索任务查看器”窗口，并显示此搜索任务的相关信息和指标。您可以在搜索运行时或搜索完成之后选择此操作。有关更多信息，请参阅“查看搜索任务属性”。
   - **删除任务。**删除当前任务，即使此任务处于运行中、暂停或完成状态。删除任务后，仍可将搜索保存为报表。

有关更多信息，请参阅“关于任务和任务管理”。

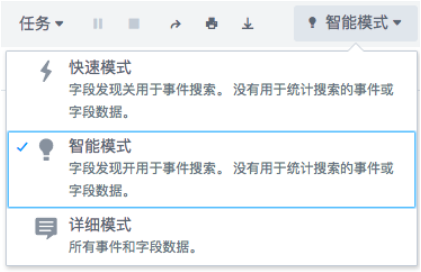
更改搜索模式

搜索模式决定搜索体验。默认搜索模式为智能模式。

快速模式
: 通过减少搜索返回的事件信息量提高搜索的速度。

详细模式
: 返回尽可能多的事件信息。

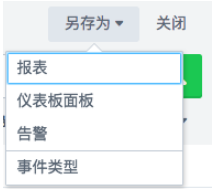
智能模式
: 自动根据所运行的搜索类型，在快速模式和详细模式之间切换搜索行为。



有关更多信息，请参阅本手册中的“搜索”模式。

保存结果

另存为菜单列出了用于将搜索结果另存为报表、仪表板面板、告警和事件类型的选项。



报表
: 将搜索另存为**报表**，以供以后再使用。您可以在“报表”页面中再次运行此报表。从应用栏中可访问“报表”页面。请阅读*《报表手册》*中有关如何“创建和编辑报表”的更多信息。

仪表板面板
: 根据搜索生成仪表板**面板**并将其添加到新的或现有仪表板中。要了解更多信息，请参阅*《仪表板和可视化》*手册中的“仪表板概览”。

告警
: 根据搜索定义**告警**。告警在后台运行报表（按**计划**或**实时**）。当搜索返回符合您在告警定义中所设置条件的结果时，即会触发告警。有关更多信息，请参阅*《告警手册》*。

事件类型
: 将拥有共同特性的事件归为一类。如果搜索不包括**管道符**或**子搜索**，您可以使用此选项将搜索另存为一种事件类型。有关更多信息，请参阅*《知识管理器》*手册中的“关于事件类型”和“在 Splunk Web 中定义事件类型”。

其他搜索操作

位于任务搜索控件与搜索模式选择器之间的是三个按钮，分别用于**共享**、**导出**和**打印**搜索结果。

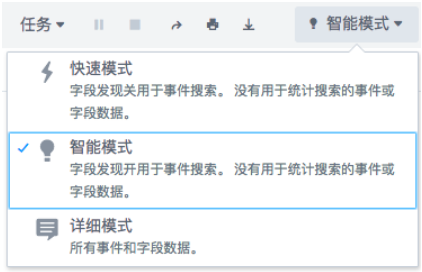
- 单击**共享**可共享任务。选择此按钮，任务的生命周期将延长至 7 天，读取权限将设置为“每个人”。有关任务的更多信息，请参见本手册中的“关于任务和任务管理”。
- 单击**导出**可导出结果。可以选择以 CSV、原始事件、XML 或 JSON 形式输出并指定要导出的结果数。
- 单击**打印**可将结果发送到已配置的打印机。

此外，可使用**另存为**菜单旁边的**关闭**按钮取消搜索并返回到 Splunk Home。

搜索模式

您可以使用搜索模式选择器来根据您的需求提供搜索体验。

搜索模式选择器位于搜索栏的右侧。模式包括智能、快速和详细。默认是智能模式。



根据您设置的模式，您可以查看搜索可用的所有数据（但需要花费较长的搜索时间），也可以通过特定方式加速并简化搜索。

快速和详细模式代表搜索模式范围的两端。默认的智能模式会根据所运行的搜索类型在快速模式和详细模式之间进行切换。当您首次运行保存的搜索时，该搜索将在智能模式下运行。

使用快速模式

快速模式以搜索性能为优先，不会返回非必要的字段或事件数据。这意味着搜索返回的是重要和所需的数据。

- **禁用字段发现。字段发现**是一个进程，Splunk 软件通过此进程来提取除了**默认字段**（如 `host`、`source` 和 `sourcetype`）之外的其他字段。Splunk 软件仅会返回默认字段和完成搜索所必须的字段的信息。如果您搜索的是特定字段，则这些字段会提取出来。
- **仅在运行报表搜索时以报表结果表形式或可视化方式描述搜索结果**。报表搜索是指包含**转换命令**的搜索。在快速模式下，您将只会看到事件列表和*不*包含转换命令的搜索的事件时间线。

有关启用或禁用字段发现后 Splunk 软件相关操作的详细信息，请参阅*知识管理员手册*中的“Splunk Enterprise 何时提取字段”。

使用详细模式

详细模式返回的是所有可能的字段和事件数据，即使这表示搜索需要较长的时间才能完成，即使搜索包含报表命令。

- **尽可能发现所有字段。**这包括默认字段、自动搜索时间字段提取以及所有用户定义的索引时间和搜索时间字段提取。发现的字段显示在“事件”结果选项卡中左侧字段边栏。
- **返回结果的事件列表视图并生成搜索时间线。**如果搜索包含报表命令，它还会生成报表表格和可视化。

如果您将一个转换搜索放在一起但不完全确定需要为哪些字段制作报表，或者您需要验证您为其建立摘要的事件是否正确，则可以使用详细模式。

在详细模式下运行报表时，报表无法受益于报表加速功能。如果为报表启用报表加速，并且此后搜索运行的速度更快，请注意，如果将搜索的模式切换为详细，搜索将以较慢的非加速速度运行。

报表加速设计用于与超出 10 万个事件和利用**转换命令**的缓慢完成搜索结合使用。有关更多信息，请参阅*报表手册*中的“加速报表”。

使用智能模式

所有报表在首次创建时，就在智能模式，即默认搜索模式下运行。智能模式设计为所运行的任何搜索或报表提供最佳结果。如果您搜索事件，可获得所需的全部事件信息。如果您运行的是转换搜索，Splunk 软件会首先考虑速度，然后再考虑完整性，并会直接显示报表结果表格或可视化。

如果运行的智能模式搜索*不*包括转换命令，则搜索行为会与在详细模式下的行为相同。

- **尽可能发现所有字段。**
- **生成完整的事件列表和事件时间线。**将不显示任何事件表或可视化，因为您需要使用转换命令来实现此目的。

如果运行的智能模式搜索包括转换命令，则搜索行为会与在快速模式下的行为相同。

- **禁用字段发现。**
- **不会花时间生成事件列表和事件时间线**，并将您直接带到报表结果表或可视化。

有关转换命令和转换搜索的更多信息，请参阅*搜索手册*中的"关于报表命令"。

搜索历史

查看搜索历史有几种方法。

搜索摘要视图中的搜索历史

完整的搜索历史会显示在搜索摘要视图底部。

使用**搜索历史**面板查看之前运行的搜索并进行交互。

单击**扩展搜索历史**以查看您的搜索历史。搜索历史将显示为含以下各列的表格：

搜索
: 包含搜索字符串，以纯文本显示，以便复制内容。默认情况下，搜索历史报表截取搜索字符串，以编写在单独一行。对于更长的搜索字符串，您可以单击搜索字符串左边的扩展图标，以显示完整的搜索字符串。

操作
: 包含操作，**添加到搜索**。单击**添加到搜索**以使用所选历史搜索内容替代搜索栏内容。您可以使用键盘快捷键显示新浏览器选项卡中的搜索。
Windows：使用 CTRL+ 单击**添加到搜索**
Mac：使用 Command + 单击**添加到搜索**

上次运行
: 包含上次运行搜索的日期和时间。

筛选以查找搜索

您可以筛选搜索历史以快速查找您正在查找的搜索。您可以按关键字或时间筛选。

- 将关键字键入**筛选**文本框以查找包含该关键字的历史搜索。例如，键入 **sourcetype=access_*** 查找包含该条件的搜索。
- 基于上次运行的搜索，从时间过滤器列表中选择。选择**今天**、**过去 7 天**或**最后 30 天**。要查看完整的搜索历史，请选择**无时间过滤器**。

对搜索历史进行排序

在搜索历史表中，单击**搜索**栏标题对按搜索条件对搜索进行字母排序。单击**上次运行**列标题，以按搜索运行的日期对搜索进行排序。您可以再次单击列标题按升序或降序对列表进行排序。

更改页面显示

默认情况下，搜索历史会显示最近的 20 个搜索。后续页面显示旧搜索。单击**每页 20** 以更改搜索历史列表中每页出现的搜索数。选择每页显示 10 条、20 条或 50 条搜索。

使用搜索助理的搜索历史

此外，键入搜索栏中的搜索条件之后，搜索助理会将历史中的搜索显示为可能和您键入的条件**匹配的搜索**。

另请参阅

浏览 Splunk Web

搜索入门

搜索命令入门

在搜索管道的开头，`search` 命令是隐含的，尽管您没有显式指定该命令。如果您键入

```
host=webserver*
```

就会好像您键入的是

```
search host=webserver*
```

使用关键字、短语、字段、布尔表达式和比较表达式来准确指定您想要从 Splunk 索引检索的事件。

具体信息请参阅：

- 通配符
- 布尔表达式
- 字段表达式
- NOT 和 != 之间的区别
- 使用 CASE 和 TERM 匹配短语
- SPL 和正则表达式

关键字和短语

默认情况下，当您使用关键字和短语进行搜索时，Splunk 软件会通过匹配数据中的原始事件字段 `_raw` 来检索事件。开始添加搜索调节器（如 `_time` 和 `tag` 等字段）时，还可对 `_raw` 字段中提取的信息进行匹配。

搜索包含关键字和带引号的短语（或任何非搜索调节器内容）的字符串时，Splunk 软件会在 `_raw` 字段中搜索匹配的事件或结果。下面是关键字和短语的一些示例：

```
web
```

```
error
```

```
web error
```

```
"web error"
```

请注意，搜索加有引号的短语 "web error" 与不加引号时的搜索不同。搜索 `web error` 时，Splunk 软件会返回包含 "web" 和 "error" 的事件。搜索 "web error" 时，Splunk 软件只返回包含短语 "web error" 的事件。

另请参阅

- 了解 SPL 语法
- 关于检索事件
- 关于搜索时间范围
- 优化的快速提示

通配符

使用星号通配符（*）字符可匹配字符串中任意数量的字符。如果您指定的星号没有其他条件，那么您要求与所有字符匹配。检索所有事件，直到达到最大限制。当 `*` 作为搜索字符串的一部分时，将基于该字符串生成匹配。例如：

- `my*` 和 `myhost1`、`myhost.ny.mydomain.com`、`myeventtype` 等匹配。
- `*host` 和 `myhost`、`yourhost` 等匹配。
- `*host*` 和 `host1`、`myhost3`、`yourhost27.yourdomain.com` 等匹配。

对于想要检索的事件而言，搜索术语越特定，越有机会匹配到事件。例如，搜索 `access denied` 将始终优于搜索 `denied`。如果 90% 的事件含有单词 `error`，仅 5% 的事件含有单词 `sshd`，而您想要查找的事件要求同时包含这两个单词，则在搜索中包括 `sshd` 将更为有效。

何时避免使用通配符

有几种情况应该要避免使用通配符。

避免在字符串中间使用通配符

单词或字符串中间的通配符可能造成不一致。如果字符串包含标点符号，如下划线 _ 或短划线 - 字符，这一点尤为重要。

例如，假设您有以下产品 ID 列表。

```
DB-SG-G01
DC-SG-G02
MB-AG-G07
MB-AG-T01
SC-MG-G01
SF-BVS-G01
SG-SH-G05
WC-SH-A02
WC-SH-G04
```

您以字母 S 开始，以 G01 结尾创建搜索，查找所有产品 ID。

```
...productID=S*G01 ...
```

由于产品 ID 包含标点符号，搜索结果可能会因为包含标点符号的数据索引方式而异。

如果产品 ID 比较简短，请在搜索中指定精确的产品 ID。例如：

```
...productID=SC-MG-G01 OR productID=SF-BVS-G01 ...
```

如果产品 ID 较长，请使用查找。请参阅“关于查找和工作流操作”。

避免使用通配符以匹配标点符号

标点符号是非数字或字母的字符。如果您想要匹配包含标点符号的部分字符串，指定含有您正在搜索的标点符号的每个字符串。

例如，您在事件的 `uri_path` 字段中有以下值。

```
/cart.do
/cart/error.do
/cart/success.do
/category.screen
/oldlink
/product.screen
/productscreen.html
/show.do
/stuff/logo.ico
```

您想要匹配以 `/cart` 开始的每个 `uri_path`。问题是该路径包含正斜线 (/) 字符和句号 (.) 字符。直接在搜索条件中指定标点符号，而不是为标点符号指定通配符，如 `/cart*`。例如，指定 `/cart.do OR /cart/error.do OR /cart/success.do`。

前缀通配符可能产生性能问题

如果您在字符串开头使用通配符，可能造成性能退化。

布尔表达式

Splunk 搜索处理语言 (SPL) 支持布尔运算符：`AND`、`OR` 和 `NOT`。

运算符必须大写。

两个词之间始终隐含有 AND 运算符，即：`web error` 与 `web AND error` 相同。所以有明确的理由将其包括在内，否则您不需要指定 AND 运算符。

NOT 运算符仅适用于紧随 NOT 之后的术语。要运用于多个术语，则必须用括号把这些术语括起来。

包含搜索通常将优于排除搜索。搜索 "access denied" 在生成结果时要快于搜索 NOT "access granted"。

评估顺序

Splunk 软件按照以下顺序评估布尔表达式：

1. 括号中的表达式。

2. `NOT` 子句。

3. OR 子句。

4. AND 子句。

示例

以下示例显示了 Splunk 软件如何处理布尔表达式。

考虑以下搜索。

```
A=1 AND B=2 OR C=3
```

这和指定 `A=1 B=2 OR C=3` 的相同

如果在指定值时没有使用括号，此搜索会处理为

```
A=1 AND ( B=2 OR C=3 )
```

以下是另一个示例。

```
error NOT 403 OR 404
```

没有括号，此搜索会处理为：

- 搜索包含 "error" 字符串但不包括关键字 403 的任何事件
- 搜索包含 "error" 字符串和 404 的任何事件

可使用括号将布尔表达式分组。例如：

```
error NOT (403 OR 404)
```

```
(A=1 AND B=2 ) OR C=3
```

字段表达式

添加数据时，Splunk 软件将提取成对的信息，并将它们另存为字段。某些字段是所有事件的公用字段，而其他字段则不是公用字段。在搜索术语中使用字段可提高匹配到特定事件的机率。

如果要在 Web 访问日志中搜索特定 HTTP 状态错误，请不要搜索 "web error 404"，您可使用字段来搜索：

```
status=404
```

请参阅“使用字段检索事件”。

使用比较运算符匹配字段值

您可使用比较运算符匹配某个特定值或一系列字段值。

运算符	示例	结果
=	field=foo	精确匹配 "foo" 的多值字段值。
!=	field!=foo	不匹配 "foo" 的多值字段值。
<	field<x	小于 x 的数字字段值。
>	field>x	大于 x 的数字字段值。
<=	field<=x	小于等于 x 的数字字段值。
>=	field>=x	大于等于 x 的数字字段值。

例如，要找到 delay 字段大于 10 的事件：

```
delay > 10
```

NOT 和 != 之间的区别

当您想从搜索中排除结果时，可以使用 NOT 运算符或 != 字段表达式。但是，这两种方式返回的结果有一个显著差异。

假设您有以下字段：

- fieldA

- fieldB
- fieldC

这些字段分别有 3 个不同的值。例如，fieldA 有 value1、value2 和 value3。

用 != 搜索

如果您搜索 `fieldB!=value3`，则此搜索**返回的值仅**为非 value3 的其他 fieldB 值：

- fieldB=value1，fieldB=value2

如果 fieldB 不存在，则不会返回任何值。

用 NOT 搜索

如果您搜索 `NOT fieldB=value3`，则此搜索**会返回所有值，只排除** fieldB=value3 的值：

- fieldA=value1，fieldA=value2，fieldA=value3
- fieldB=value1，fieldB=value2
- fieldC=value1，fieldC=value2，fieldC=value3

如果 fieldB 不存在，则 `NOT fieldB=value3` 返回：

- fieldA=value1，fieldA=value2，fieldA=value3
- fieldC=value1，fieldC=value2，fieldC=value3

使用 CASE() 和 TERM() 匹配短语

如果要搜索 Splunk 索引中的特定术语或短语，请使用 CASE() 或 TERM() 指令对整个术语进行精确的匹配。

- CASE：搜索术语和字段值的匹配项，区分大小写。
- TERM：将括号中的内容匹配为索引中的一个术语，即使其中包含通常被识别为次要分隔符的字符（如句点或下划线）。

当搜索含有次要分隔符的术语时，默认会被视为一个短语：它搜索子术语（即两个次要分隔符之间的术语）的组合项，并对结果进行后期过滤。例如，当您搜索 IP 地址 127.0.0.1 时，Splunk 软件会搜索：`127 AND 0 AND 1`

如果这些子术语的组合项很常见，即使整个术语本身并不常见，搜索效率也不会很高。

如果搜索 TERM(127.0.0.1)，Splunk 软件会将 IP 地址视为单个术语，然后在原始数据中寻找匹配项。

TERM 比较适用于术语包含次要分隔符并且被主分隔符（如空格或逗号）限制的情况。实际上，TERM 不适用于不被主分隔符限制的术语。以下示例说明了这一点。

有关 Splunk 软件如何将事件划分成若干个可搜索段的更多信息，请阅读*《数据导入》*手册中的“关于分段”。

示例

TERM(127.0.0.1) 适用于类似如下所示的原始数据：

127.0.0.1 - admin

但是，它不适用于类似如下所示的数据：

ip=127.0.0.1 - user=admin

这是因为 "=" 是次分割符，而事件的 IP 地址部分的索引为：ip, 127, 0, 1, ip=127.0.0.1

如果您的数据如下所示：

ip 127.0.0.1 - user admin

TERM(user admin) 无法返回结果。空格是主分割符，且短语 "user admin" 不会作为单个术语建立索引。

SPL 和正则表达式

Splunk 正则表达式是 PCRE（Perl 兼容正则表达式）。

您可以与 `rex` 和 `regex` 命令一起使用正则表达式。您也可以与 `match` 和 `replace` 等评估函数一起使用正则表达式。

以下为在 Splunk 搜索中使用正则表达式前应该了解的几点事项。

管道符

正则表达式中使用管道符 (|) 来指定 OR 条件。例如，A 或 B 的表达式为 A | B。

因为管道符在 SPL 中用于分隔命令，所以您必须将使用管道符的正则表达式用引号引起来。例如：

```
...|regex "expression | with pipe"
```

SPL 会将其解释为查找文本 "expression" 或 "with pipe" 的搜索。

反斜杠字符

正则表达式中的反斜杠字符 (\ ) 用于“转义”特定字符。例如。正则表达式中的句点字符用于匹配除了换行符之外的任意字符。如果您想匹配一个句点字符，则必须在正则表达式中使用 `\.` 以转义此句点字符。

Splunk SPL 使用星号 (*) 作为通配符。反斜杠不能用于转义搜索字符串中的星号。

如果一个正则表达式包含双反斜杠而包含此正则表达式的搜索遇到了双反斜杠，例如文件路径如 `c:\\temp` 中的情况，则此搜索将第一个反斜杠解释为正则表达式的转义符。此文件路径会解释为 `c:\temp`，即删除了其中一个反斜杠。

您必须通过在文件路径的根部分指定 **4** 个连续反斜杠的方式，同时转义文件路径中的两个反斜杠字符。例如：`c:\\\\temp`。对于较长的文件路径，如 `c:\\temp\example`，您可以在搜索字符串中的正则表达式中指定 `c:\\\\temp\\example`。

正则表达式的更多信息

有关更多信息：

- 请参阅“使用正则表达式提取字段”
- 请参阅*《知识管理器手册》*中的“有关 Splunk 正则表达式”。

优化搜索

关于搜索优化

搜索优化是一项尽可能提高搜索运行效率的技术。

如未优化，搜索运行时间通常更久，检索的索引数据量比实际需求更大，低效率地使用更多的内存和网络资源。将这些问题乘以数百或数千次搜索，最终导致系统缓慢或迟缓。

您可以遵循一系列基本原则来优化搜索。

- 仅检索所需数据
- 尽可能少移动数据
- 尽可能多并行化处理工作
- 设置合适的时间窗口

要实施搜索优化原则，请使用以下技术。

- 初始搜索中尽可能多地过滤
- 仅连接和查找所需数据
- 对最少的可能事件进行评估
- 在搜索条件中，尽可能晚地移动将数据带到搜索头中的命令

索引和搜索

运行搜索时，Splunk 软件会使用索引文件中的信息，以识别从磁盘中检索哪些事件。从磁盘检索事件数量越少，搜索运行越快。

您如何构建搜索对从磁盘检索的事件数量有重大影响。

索引数据之后，会按照时间将数据处理成事件。经处理的数据由几个文件构成：

- 压缩形式的原始数据（**原始数据**）
- 指向原始数据的索引（**索引文件**，也称为 **tsidx 文件**）。
- 一些元数据文件

这些文件将写入磁盘并驻留在一组按时间组织的目录（称为**数据桶**）中。

有效使用索引

限制从磁盘提取的数据量的一种方式是将数据划分为单独的索引。如果您很少一次跨多种数据类型执行搜索，将不同类型的数据划分为单独的索引。然后将搜索限制在特定的索引。例如，将 Web 访问数据存储在一个索引中，将防火墙数据存储在另一个索引中。建议对稀疏数据使用单独索引的方法，因为稀疏数据可能存在于大量不相关数据中。

- 请参阅*《管理索引器和索引器群集》*手册中的“如何设置多个索引”。
- 请参阅“从索引中检索事件”

两种搜索说明

一些经常使用的搜索会不必要地消耗大量系统资源。您将了解如何只需优化一个搜索就能节省重要的系统资源。

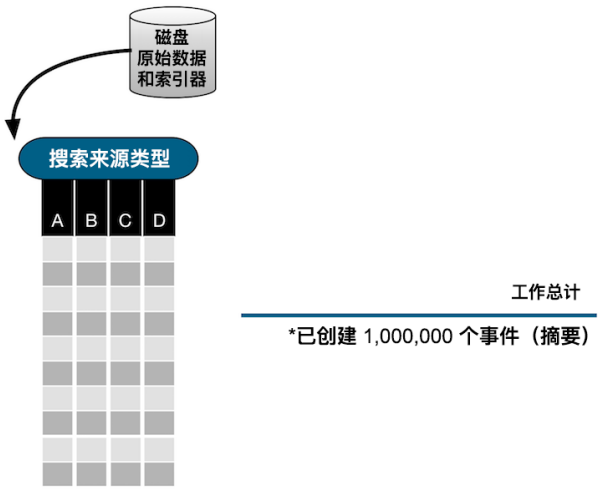
经常使用的搜索

一个经常使用的搜索包含查找、评估和另一个搜索。例如：

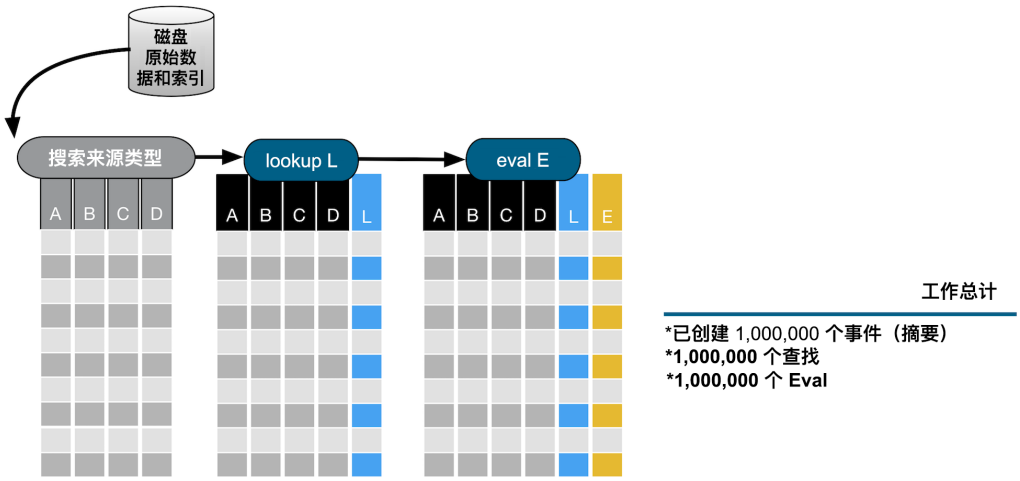
```
sourcetype=my_source | lookup my_lookup_file D OUTPUT L | eval E=L/T | search A=25 L>100 E>50
```

以下图表以简单直观的呈现方式显示本搜索。

运行搜索时，将访问索引并根据来源类型提取 100 万个事件。



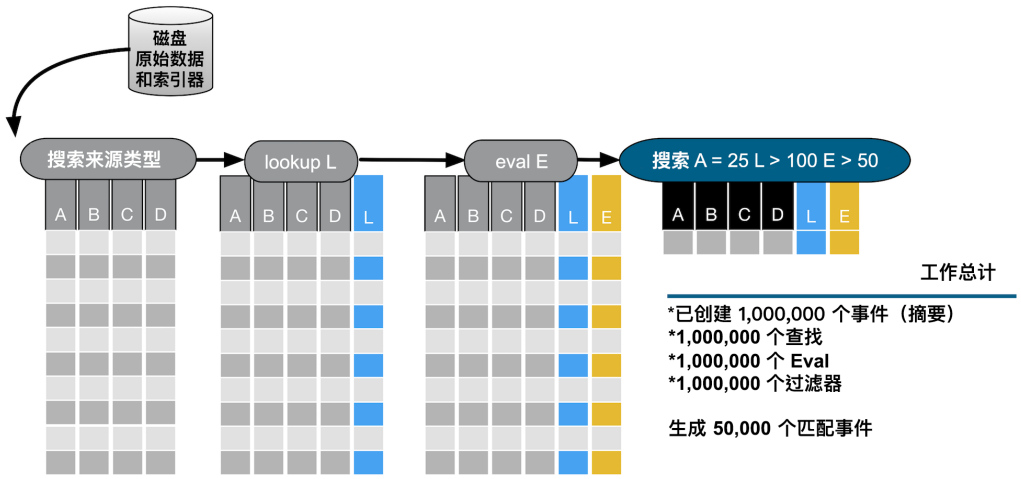
在下一步搜索过程中，在全部 100 万个事件上运行 `lookup` 和 `eval` 命令。`lookup` 和 `eval` 命令都将列添加到事件，如下图所示。



最后，第二个搜索命令是针对 A、L 和 E 列运行的。

- 对 A 列而言，搜索正在查找等于 25 的值。
- 对于添加为 `lookup` 命令结果的 L 列，搜索将查找大于 100 的值。
- 对于添加为 `eval` 命令结果的 E 列，搜索将查找大于 50 的值。

识别出和 A、L 和 E 列条件匹配的事件并返回和搜索条件匹配的 50,000 个事件。下图显示了完整过程以及该无效搜索中所涉及的资源成本。



优化搜索

您可以在搜索过程中通过将一些组件从第二个 `search` 移动到较早位置来优化整个搜索。

移动第一个管道之前的 `A=25` 条件将过滤之前的事件并减少访问索引的时间。提取的事件量是 300,000。和原始搜索相比，减少了 700,000。针对 300,000 个事件而不是 100 万个事件执行 `lookup`。

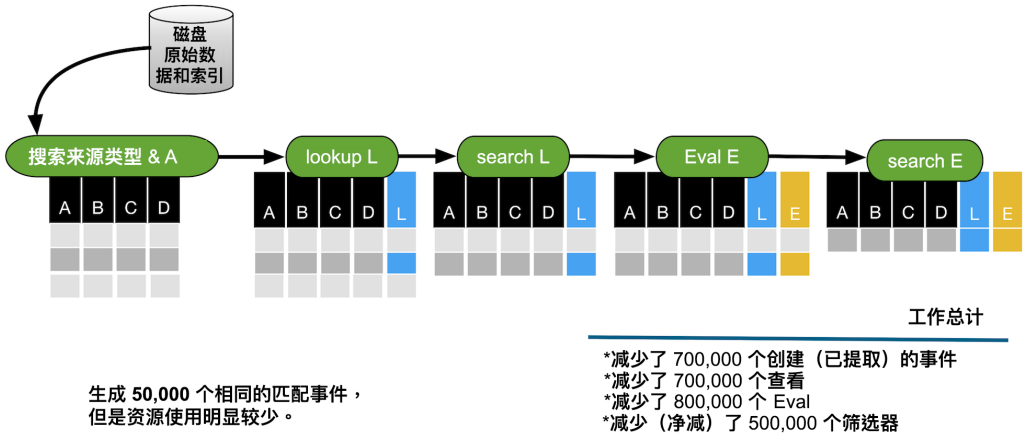
在 `lookup` 进一步过滤事件之后立即移动条件 `L>100`，减少 100,000 返回的事件数量。针对 200,000 个事件而不是 100 万个事件执行 `eval`。

条件 `E>50` 取决于 `eval` 命令的而结果并无法移动。结果和原始搜索相同。返回 50,000 个事件，但对资源的影响非常小。

这就是优化搜索。

```
sourcetype=my_source A=25 | lookup my_lookup_file L OUTPUTNEW L | search L>100 | eval E=L/T | search E>50
```

下图显示重新安排搜索条件的影响。



另请参阅

- 优化的快速提示
- 编写更好的搜索
- 内置优化

优化的快速提示

快速搜索的关键是尽量将要从磁盘中提取的数据量限制在绝对最小值。尽早在搜索中筛选数据，这样可将需要处理的数据量减至最低。

限制从磁盘提取的数据量

通过尽可能具体地设置狭窄的时间窗口和检索必要事件的最小数量来限制从磁盘范围检索的数据量的技术。

缩小时间窗口

限制从磁盘提取的数据量的最有效方式之一是限制时间范围。使用时间范围挑选器，或在搜索中指定时间调节器，以识别搜索所需的最小时间窗口。

如果您只需要查看过去一小时的数据，请不要使用默认的**过去 24 小时**时间范围。

- 请参阅"选择要应用于搜索的时间范围"
- 请参阅"在搜索中指定时间调节器"

如果您必须使用宽泛的时间范围（如**过去一周**或**所有时间**），那么请使用其他技术限制从该磁盘检索的数据量。

指定索引、数据来源或来源类型

了解如何组织数据对优化搜索而言很重要。花时间了解哪些索引包含数据、数据来源或来源类型。了解数据相关信息有助于您缩小搜索范围。

1. 运行以下搜索：

   ```
   source=*
   ```

   此搜索并非一个优化搜索，但您可借此机会了解所访问的数据。

2. 在**选择字段**列表中，单击各类型字段并查看主机、数据来源和源类型值。
3. 在**感兴趣的字段**列表中，单击**索引**字段。查看您已访问的索引名称。

尽量在搜索中指定索引、数据来源或来源类型。为数据建立索引时，Splunk 软件会自动使用许多字段标记每个事件。索引、数据来源或来源类型字段将作为默认字段自动添加到每个事件。*默认字段*是 Splunk 软件在搜索时间在事件数据中识别的索引字段。主机、数据来源和来源类型字段描述事件产生位置。

特定

尽可能在搜索中使用特定术语。尽量避免使用通配符。

例如，无需使用关键字通配符，而是：

```
*error
```

使用特定关键字：

```
fatal_error
```

以下是另一个示例。

无需使用字段值通配符，而是：

```
status=404 OR status=5*
```

指定各值：

```
status=404 OR status=500 OR status=503
```

将来源类型或索引和一个或多个字段-值对结合。例如：

```
sourcetype=access_* status=200 action=purchase
```

此搜索只检索您的 Web 访问日志中事件。将通配符 access_* 用于字段值以匹配任何 Apache Web 访问来源类型。来源类型可能为 `access_common`、`access_combined` 或 `access_combined_wcookie`。两个特定字段值对包括在搜索 `status=200` 和 `action=purchase` 中。

限制检索的事件数量

您可以指定通过 `head` 命令检索的事件数量。`head` 命令仅检索历史搜索的最近 N 个事件或实时搜索的前 N 个已捕获事件。

以下几种情况中限制检索的事件数量很有用：

- 正在创建搜索并想要确定正在检索的事件是否正确。
- 您只需要搜索事件的子集或示例集

例如：

```
sourcetype=access_* | head 1000 ...
```

避免使用 NOT 表达式

跟踪 NOT 表达式所用的资源比您指定要查找的内容更多。尽量避免使用 NOT 表达式。例如，无需使用 NOT 字符串或 != 表达式，例如：

```
(NOT host=d NOT host=e)
```

或

```
(host!=d AND host!=e)
```

使用您正在搜索的特定术语：

```
(host=a OR host=b OR host=c).
```

要了解更多信息，请参阅"NOT 和 != 之间的区别"。

尽快过滤

在计算前尽快筛选出结果。您可以使用字段值对和命令过滤结果。

使用第一管道前面的字段值对

为字段值对建立索引。在第一管道前指定字段值对是过滤事件的有效方式。

例如，在以下搜索中，术语 `status=404` 在一个单独搜索中：

```
ERROR | search status=404
```

将术语 `status=404` 移到第一个管道符前：

```
ERROR status=404
```

以下是另一个示例。

第二个搜索包括术语 `clientip="10.0.0.0/8"`。没有理由等待过滤该术语。

```
ERROR | stats sum(bytes) as sum by clientip | search sum >1048576 AND clientip="10.0.0.0/8"
```

移动术语 `clientip="10.0.0.0/8"` 以过滤出 `stats` 命令前面的所有其他 clientip 地址。

```
ERROR clientip="10.0.0.0/8" | stats sum(bytes) by clientip | search sum > 1048576
```

计算命令前使用筛选命令

在命令执行计算（如 `eval`）之前使用筛选命令（如 `where`）。

例如，此搜索的 `eval` 命令后有一个 `where` 命令。运行 `where` 命令之前，搜索不需要 `eval` 命令结果。

```
field1=value | eval KB=bytes/1024 | where field2=field3
```

移动 `where` 命令以在处理 `eval` 命令前过滤结果：

```
field1=value | where field2=field3 | eval KB=bytes/1024
```

过滤搜索结果中不必要的字段

您可以使用 `fields` 之类的命令将不必要的字段从搜索结果中移除。

尽量推迟使用非流命令

尽量推迟使用搜索中的推迟命令，如 `dedup`、`sort` 和 `stats`。这些命令被称为非流命令。在能够运行这些命令之前，必须返回整个结果集。例如，除非所有结果均可用，否则不能对结果进行排序。

关于流命令和非流命令之间的区别的说明，请参阅"命令类型"。

关于按类型划分的命令列表，请参阅*搜索参考*中的"命令类型"。

搜索优化的其他技术

有几种其他技术可以用于优化搜索。

- 使用仪表板中的后期处理搜索。请参阅*仪表板和可视化*中的“搜索驱动仪表板和表单”。
- 使用**摘要索引**、**报表加速**和**数据模型加速**功能。
- 使用**快速模式**，通过减少搜索返回的事件数据来加快搜索速度。请参阅“搜索模式”。

另请参阅

- 关于搜索优化
- 编写更好的搜索
- 内置优化

编写更好的搜索

本主题介绍了搜索缓慢的一些原因，以及可帮助您编写运行效率更高的搜索的一些指南。影响搜索速度的因素有很多，包括：

- 您正在搜索的数据量
- 搜索的构建方式
- 并发搜索的数量

要优化搜索运行速度，请尽量减少每个搜索组件所需的处理时间。

明确您的搜索类型

对于优化搜索的建议会因所运行搜索的类型和所搜索数据的特性而异。基于您想要完成的目标，搜索分为两种类型。搜索旨在检索事件或生成汇总或组织数据的报表。

请参阅“搜索类型”。

检索事件的搜索

原始事件搜索会从 Splunk 索引中检索没有对检索到的事件进行任何其他处理的事件。从索引中检索事件时，请使用有关要检索事件的特定信息。您可以使用事件特有的关键字和字段-值对来实现此目的。

如果您要检索的事件常常出现在数据集中，此搜索称为*密集搜索*。如果您要检索的事件很少出现在数据集中，此搜索称为*稀疏搜索*。对大量的数据进行稀疏搜索所需的时间要比对相同数据集执行密集搜索所需的时间长。

请参阅“使用字段检索事件”。

生成报表的搜索

报表生成的搜索或转换的搜索会在从索引中检索到事件之后对事件执行其他处理。这种处理可以包括筛选、转换以及对结果集使用一个或多个统计函数的其他运算。由于这种处理发生在内存中，因此您检索事件时限制性和特定性越高，搜索速度就会越快。

请参阅“有关转换命令和搜索”。

命令类型和并行处理

某些命令可以处理流中的事件。一个事件进来，然后一个事件（或没有事件）出去。这些都称为**流命令**。流命令示例如下：`where`、`eval`、`lookup` 和 `search`。

其他命令在命令完成前需要来自所有索引器的全部事件。这些都称为**非流命令**。非流命令示例如下：`stats`、`sort`、`dedup`、`top` 和 `append`。

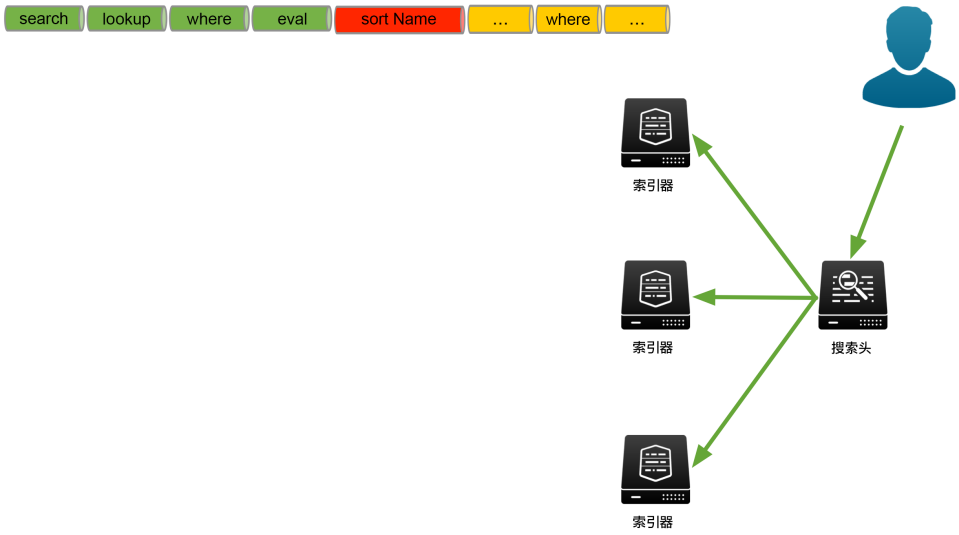
所有数据可用时，只能运行非流命令。要处理非流命令，需将来自索引器的所有搜索结果发送到搜索头。发送之后，所有进一步处理必须由搜索头执行，而不是在索引器上并行处理。

并行处理示例

搜索中的早期非流命令会减少并行处理。

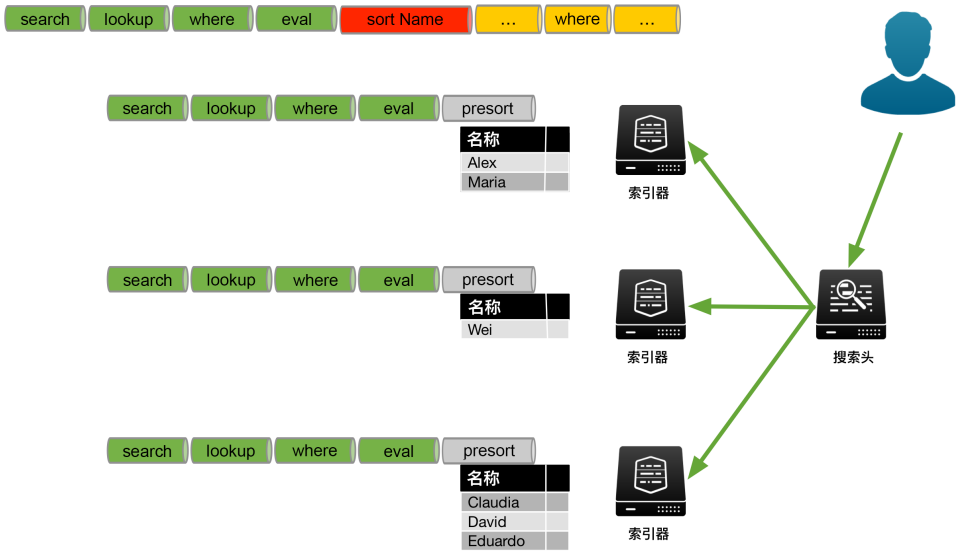
例如，下图显示了用户运行的搜索。该搜索以 `search` 命令开始，该命令隐藏为“搜索”栏中的第一个命令。然后使用 `lookup`、`where` 和 `eval` 命令继续搜索。然后搜索会基于 `Name` 字段包含 `sort`，紧接着使用另外一个 `where` 命令。

搜索会发送到搜索头并分发给索引器，以处理和索引器一样多的搜索内容。



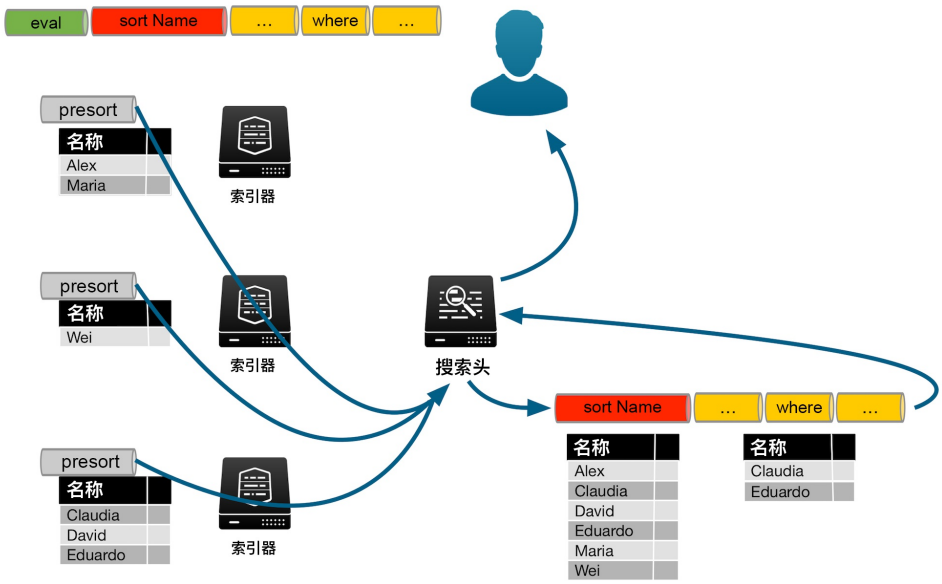
对于各索引器上的事件，索引器会处理搜索，除非索引器遇到非流命令。在此示例中，索引器会通过 `eval` 命令处理搜索。要执行 `sort`，所有结果必须发送到搜索头进行处理。

但是，各索引器上的结果可以用索引器进行排序。这称为*预排序*。在本示例中，排序在 `Name` 字段中。下图显示了第一个索引器返回了名字：Alex 和 Maria。第二个索引器返回了姓名 Wei。第三个索引器返回了姓名 Claudia、David 和 Eduardo。



要返回按姓名排序的完整结果列表，必须将和搜索条件匹配的所有事件发送到搜索头。所有结果都发送到搜索头上之后，剩余搜索必须在搜索头中处理。在此示例中，`sort` 和任何剩余命令都在搜索头中处理。

下图显示每个索引器已根据 `Name` 字段预排序结果。结果已发送到搜索头，且附加在一起。搜索头之后会按照正确的顺序对整个列表进行排序。搜索头会处理搜索中的剩余命令，以产生最终结果。在此示例中，包含第二个 `where` 命令。将最终结果返回给用户。



在索引器上运行部分或全部搜索之后，搜索会并行处理且搜索性能会大幅提高。

要优化搜索，尽量推迟将非流命令放到搜索字符串中。

调整搜索提示

大多数情况下搜索较慢，因为您从索引中检索事件的查询较为复杂。例如，如果搜索中包含非常大的 OR 列表、复杂子搜索（细分至 OR 列表中）和短语类型搜索，处理过程将需要较长时间。此部分介绍调整搜索的一些提示，使搜索效率更高。

在具有高基和大量不常见或独特值的一组字段值中使用 BY 子句执行统计需要大量内存。一种可能的补救方法是减少和 `tstats` 命令一起使用的 `chunk_size` 设置的值。另外，减少 BY 子句必须处理的不同值的数量也有用。

将搜索限制在特定的索引

如果您很少一次跨多种数据类型执行搜索，将不同类型的数据划分为单独的索引。然后将搜索限制在特定的索引。例如，将 Web 访问数据存储在一个索引中，将防火墙数据存储在另一个索引中。建议对稀疏数据使用这一方法，因为稀疏数据可能存在于大量不相关数据中。

请参阅*管理索引器和索引器群集*中的“创建自定义索引”和本手册中的“从索引中检索事件”。

有效使用字段

使用已经提取的字段（索引字段），而不是在搜索时间提取字段，则使用字段搜索的速度会更快。有关索引字段和默认字段的更多信息，请参阅*《知识管理器手册》*中的“关于字段”。

使用索引和默认字段

尽可能在任何时候利用索引字段和默认字段来帮助您有效搜索或筛选数据。索引时，Splunk 软件会提取一组每个事件通用的默认字段。这些字段包括 `host`、`source` 以及 `sourcetype`。尽早在搜索中使用这些字段来筛选数据，这样，可将需要处理的数据量减至最低。

例如，如果您要构建有关 Web 访问错误的报表，请在报表命令之前先搜索这些特定错误：

```
sourcetype=access_* (status=4* OR status=5*) | stats count by status
```

用 <field>::<value> 指定索引字段

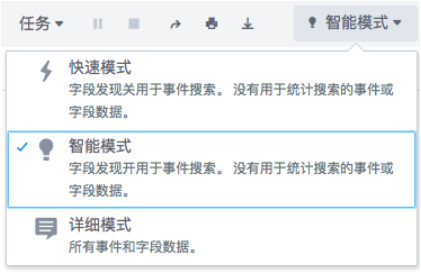
对于从结构性数据（如 CSV 文件和 JSON 数据来源）索引的字段，您还可以运行高效的搜索。进行此操作时，请使用双冒号替代等号，如：`<field>::<value>`。

虽然此语法亦可用于搜索默认和自定义索引字段，但是搜索结构性数据中索引的字段时表现最佳。您无法将其在**搜索时间**字段中搜索。

可以禁用字段发现来改进搜索性能

如果搜索不需要其他字段，可对**搜索模式**进行设置以禁用字段发现，从而改善时间线视图的搜索性能，或者使用

`fields` 命令仅指定要在结果中查看的字段。



但是，禁用字段发现会仅禁用自动**字段提取**，但会提取执行搜索所需的字段，如您明确要搜索的字段和**默认字段**（如 `_time`、`host`、`source` 和 `sourcetype`）。搜索运行速度会更快，因为 Splunk 软件不再尝试从事件中提取所有可能的字段。

默认情况下，**搜索模式**设置为**智能**。如果您使用报表命令运行搜索，但不知道数据中存在哪些字段，而且认为自己可能需要借助这些字段在某种程度上缩小搜索范围，请将搜索模式设置为**详细**。

请参阅“设置搜索模式”。

另请参阅*搜索参考*中的字段命令相关主题。

汇总数据

在非常大的数据集中执行搜索可能需要很多的时间。如果您需要定期生成有关大量数据的报表，可使用摘要索引预先计算出报表中最常使用的值。对保存的搜索进行计划以收集指标，并生成汇总数据（而不是原始数据）的报表。

请参阅“使用摘要索引提高报表效率”。

使用搜索任务查看器

您可以使用**搜索任务查看器**这个工具来解决搜索性能问题，同时还能决定搜索在哪个阶段花费的时间最多。它会对搜索行为进行详细分析，从而帮助了解有关知识对象（如事件类型、标记、查找和搜索内的其他组件）执行成本的信息。

请参阅本手册中的“查看搜索任务属性”。

另请参阅

- 关于搜索优化
- 优化的快速提示
- 内置优化

内置优化

Splunk 软件包括内置优化，可以分析和处理您的搜索以实现效率最大化。

这些优化的一个目标是尽快筛选结果。筛选会减少搜索需要处理的数据量。尽早筛选可避免对不属于最终搜索结果的事件进行不必要的查找和评估计算。

还有一个目标是尽可能多地在索引器中并行处理。内置优化可对搜索处理进行重新排序，这样就可以在将搜索结果发送到搜索头进行最终处理之前，尽可能多地在索引器上并行运行命令。您可以使用“任务查看器”看到重新排序的搜索。请参阅“分析搜索优化”。

断言优化

作为过滤器进行断言，以在处理搜索时移除不必要的事件。在搜索管道中应用断言的时间越早，系统资源的使用效率越高。如果尽早应用断言，获取事件时，仅会获取和断言匹配的事件。如果搜索过程中应用相同断言较晚，那么获取事件超出需求时会造成资源浪费。

有几项注重于断言的内置优化：

- 断言合并
- 断言下推
- 断言拆分

断言类型

断言类型分为简单断言和复杂断言两种。

- 简单断言是表单 `field = value` 的单一条件。
- 复杂断言是几个结合（AND 和 OR）和分离 (NOT) 的组合。例如，`Field1 = Value1 OR Field2 = Value2 AND Field3 = Value3`。

可合并或拆分复杂断言以优化搜索。

在 Splunk SPL 中，有两个执行基于断言的筛选命令，`where` 和 `search`。

使用 `where` 命令进行筛选的示例为：

```
index="_internal" | where bytes > 10
```

使用 `search` 命令进行筛选的示例为：

```
index="_internal" | search bytes > 10
```

基于搜索的断言是基于 where 的断言的子集。换句话说，基于 where 的断言可替换基于搜索的断言。但是基于搜索的断言不能替换基于 where 的断言。

断言合并

断言合并优化需要两个断言并将其合并为一个。例如，考虑以下搜索：

```
index=main | search a > 10 AND fieldA = "New"
```

本示例中有两个搜索命令。在每个搜索开始时隐含的管道前 `search` 命令和第一管道后的显式 `search` 命令。通过断言合并优化，第二个 `search` 命令中的断言会和第一个 `search` 命令中的断言合并。例如：

```
index=main AND a > 10 AND fieldA = "New"
```

某些情况下，使用 `where` 命令的断言也可以合并。例如，考虑以下搜索：

```
index=main | where fieldA = "New"
```

通过断言合并优化，`where` 命令中指定的断言会和 `search` 命令中指定的断言合并。

```
index=main AND fieldA = "New"
```

字段提取和断言合并问题

如果正则表达式模式提取的是子标记，那么**内联字段提取**需要特殊处理。`fields.conf` 文件中的字段必须设为 `indexed=false`。请参阅*《知识管理器手册》*中的“内联字段提取配置示例”。

考虑以下样本事件：

```
Mon Apr 17 16:08:16 2017 host=10.10.1.1  Login  name=John  SUCCESS FRANCE
```

您可以创建一个使用以下正则表达式的名为 `country` 的已提取的字段：

```
SUCCESS\s+?\S{3}
```

- `SUCCESS` 字面上匹配字符 SUCCESS 并区分大小写。
- `\s` 匹配任何空格字符（空格、选项卡、新行）。
- `+?` 匹配一次和无限次之间的量词，匹配模式的最少量。
- `\S` 匹配任何非空白字符。
- `{3}` 是一个量词，精确匹配 3 个非空格字符。

对于示例事件，以下正则表达式会提取单词 FRANCE 的前三个字符或 `FRA`。提取 `FRA` 是索引的术语 FRANCE 的子标记。

当您使用提取的字段进行搜索时，例如：

```
index=main | search country=FRA
```

已使用断言合并优化程序优化搜索：

```
index=main country=FRA
```

但是搜索未返回结果，因为 `FRA` 不是索引的一部分。`FRANCE` 是索引的术语。

要解决这个问题，您必须将以下设置添加到 `fields.conf` 文件中：

```
[country]
indexed=false
```

您还可以禁用内置优化。请参阅“优化设置”。

断言下推

断言下推优化会分析搜索并对搜索处理进行重新排序，这样能够尽早处理断言。

简单断言下推示例

您可执行以下搜索：

```
sourcetype=access* (status=401 OR status=403) | lookup usertogroup user OUTPUT group | where
src_category="email_server"
```

搜索的 `sourcetype=access* (status=401 OR status=403)` 部分检索了 50,000 个事件。在全部 50,000 个事件中执行 `lookup`。然后应用 `where` 命令，过滤不符合 `src_category="email_server"` 条件的事件。结果是丢弃了 45,000 个事件，5,000 个事件返回到搜索结果中。

如果在 `lookup` 前应用使用 `where` 命令的搜索条件，则会从结果中过滤更多事件。执行 `lookup` 之前仅从磁盘检索了 5,000 个事件。

通过断言下推，重新排序搜索以进行处理。在第一管道之前通过移动搜索条件 `src_category="email_server"` 取消 `where` 命令。

```
sourcetype=access* (status=401 OR status=403) src_category="email_server" | lookup usertogroup user OUTPUT group
```

复杂断言下推示例

考虑以下搜索片段：

```
... | eval x=if(isnull(x) OR x==""), "missing", x) | where x = "hello"
```

在这种情况下，字段无效或为空时 `eval` 命令正在分配默认值。这是常用信息模型 (CIM) 数据模型中的常用模式。

内置优化会在处理搜索前重新组织搜索条件。在执行 `eval` 命令之前已移动 `where` 命令。

```
... | where x = "hello" | eval x=if(isnull(x) OR x==""), "missing", x)
```

断言拆分

断言拆分是将一个断言划分或拆分成更小断言的操作。然后，断言拆分优化程序可尽可能将较小的断言移动到搜索较早的位置。

考虑以下搜索：

```
index="_internal" component = "SearchProcess" | eval a = (x + y) | where a > 200 AND x < 10
```

因为 `field a` 生成作为 `eval` 命令的一部分，所以无法在搜索中较早处理。但是，`field x` 在事件中且可较早进行处理。断言拆分优化划分搜索 `where` 部分的组件。对搜索重新排序以较早处理符合条件的组件。

```
index="_internal" component = "SearchProcess" | where x < 10 | eval a = (x + y) | where a > 200
```

在执行 `eval` 命令之前已移动 `where` 命令 `x < 10` 部分。此次移动减少了 `eval` 命令必须要处理的结果数量。

投影清除

此优化会分析搜索并确定搜索中指定的任何生成字段是否会用于产生搜索结果。如果生成的字段识别为可清除，则运行优化的搜索版本。搜索语法保持不变。

例如，考虑以下搜索：

```
index=_internal | eval c = x * y / z | stats count BY a, b
```

`eval c = x * y / z` 生成的字段 `c` 不用于 `stats` 计算。可将 `c` 字段从搜索中清除。

搜索语法保持不变：

```
index=_internal | eval c = x * y / z | stats count BY a, b
```

但是现在运行的优化搜索是：

```
index=_internal | stats count BY a, b
```

以下是另一个示例：

```
| tstats count FROM datamodel=internal_audit_logs
```

对于未构建数据模型的数据桶来说，这将扩展以下回退搜索：

```
| search ((index=* OR index=_*) index=_audit)
| eval nodename = "Audit"
| eval is_searches=if(searchmatch("(action=search NOT dmauditsearch)"),1,0),
  is_not_searches=1-is_searches, is_modify=if(searchmatch("(action=edit_user
  OR action=edit_roles OR action=update)"),1,0), is_not_modify=1-is_modify
| eval nodename = if(nodename == "Audit"
  AND searchmatch("(action=search NOT dmauditsearch)"), mvappend(nodename,
  "Audit.searches"), nodename)
| eval is_realtime=case(is_realtime == 0, "false", is_realtime == 1, "true",
  is_realtime == "N/A", "false"), search_id=replace(search_id,"'",""),
  search=replace(search,"'",""), search_type=case((id LIKE "DM_%"
  OR savedsearch_name LIKE "_ACCELERATE_DM%"), "dm_acceleration",
  search_id LIKE "scheduler%", "scheduled", search_id LIKE "rt%", "realtime",
  search_id LIKE "subsearch%", "subsearch", (search_id LIKE "SummaryDirector%"
  OR search_id LIKE "summarize_SummaryDirector%"), "summary_director",
  1=1, "adhoc")
| rename is_realtime AS Audit.searches.is_realtime
  search_id AS Audit.searches.search_id
  search AS Audit.searches.search
  search_type AS Audit.searches.search_type
| eval nodename = if(nodename == "Audit" AND searchmatch("(action=edit_user
  OR action=edit_roles OR action=update)"), mvappend(nodename,
  "Audit.modify"), nodename)
| rename action AS Audit.action info AS Audit.info object AS Audit.object
  operation AS Audit.operation path AS Audit.path user AS Audit.user
  exec_time AS Audit.exec_time result_count AS Audit.result_count
  savedsearch_name AS Audit.savedsearch_name
  scan_count AS Audit.scan_count total_run_time AS Audit.total_run_time
  is_searches AS Audit.is_searches is_not_searches AS Audit.is_not_searches
  is_modify AS Audit.is_modify is_not_modify AS Audit.is_not_modify
| addinfo type=count label=prereport_events
| stats count
```

该搜索可优化为此语法：

```
| search ((index=* OR index=_*) index=_audit) | stats count
```

清除不必要的生成字段（或投影）可提高搜索性能。

事件类型和标记优化

事件类型和标记的搜索处理时间会非常长。Splunk Enterprise Security 尤其如此，其中有很多定义的事件类型和标记。系统中定义的事件类型和标记越多，注释成本越高。

内置事件类型和标记优化将应用于转换搜索和数据模型加速搜索。

转换搜索

在未执行内置优化的情况下运行转换搜索，会将所有事件类型和标记从配置系统上传到搜索处理器。无论搜索是否需要事件类型和标记，都会上传。搜索处理尝试对各事件应用事件类型和标记。

例如，考虑以下搜索：

```
index=_internal | search tag=foo | stats count by host
```

此搜索仅需要标记 `foo`。但是所有事件类型和标记都会上传，不需要优化。内置优化可通过分析搜索和仅上传需要的事件类型和标记解决此问题。如果不需要事件类型或标记，则不会上传，这样可提高搜索性能。

对于转换搜索，此优化通过 `limits.conf` 文件（位于 `[search_optimization::required_field_values]` 段落中）中的几项设置控制。这些设置默认为打开。除非您需要对这些问题进行故障排除，否则不需要更改这些设置。

数据模型加速搜索

事件类型和标签优化可解决的另一个问题是专门针对数据模型加速。通过 `datamodels.conf` 文件中的配置设置，您可以指定想要和数据模型加速搜索结合使用的标记列表。您可以使用 `tags_whitelist` 设置在 `[dm.name]` 段落下指定列表。

有关 `tags_whitelist` 设置的详细信息，包括使用情况示例，请参阅*知识管理员手册*中的"设置标记白名单以优化数据模型性能"。

分析搜索优化

您可以使用"任务查看器"分析内置优化的影响。

确定搜索处理时间

您可使用"任务查看器"确定内置搜索优化是否有助于您更快完成搜索。

1. 运行搜索。
2. 在搜索操作按钮中，选择**任务 > 检查任务**。
3. 在"任务查看器"窗口中，查看窗口顶部的消息。消息类似于"搜索已结束，通过在 X 秒内扫描了 X 个事件，返回了 X 个结果"。记住完成搜索的秒数。
4. 关闭"任务查看器"窗口。
5. 使用搜索栏将 `|noop search_optimization=false` 添加到搜索末端。这将关闭对本次搜索的内置优化。
6. 运行该搜索。
7. 查看任务并将"任务查看器"窗口顶部的消息和之前的消息进行比较。此消息指定了未经内置优化的搜索处理需要多少秒。

查看优化搜索

您可以将原始搜索和打开内置优化时创建的优化搜索进行比较。

1. 运行搜索。
2. 在搜索操作按钮中，选择**任务**、**检查任务**。
3. 在"任务查看器"窗口中，扩展**搜索任务属性**部分并滚动到 **normalizedSearch** 属性。该属性显示了基于原始搜索创建的内部搜索。
4. 滚动到 **optimizedSearch** 属性。该属性显示了应用内置优化时基于 **normalizedSearch** 创建的搜索。

优化设置

默认启用内置优化。

针对特定搜索关闭优化

极少数情况下，内置在搜索处理器中的优化可能无法正确优化搜索。如遇这些情况，您可以通过关闭该特定搜索的搜索优化解决问题。

您可以通过关闭搜索优化确定意外或有限的搜索结果是否是由搜索优化导致的。

使用 `noop` 命令关闭特定搜索的内置优化。

将 `noop search_optimization=false` 添加到搜索末端。例如：

```
| datamodel Authentication Successful_Authentication search | where Authentication.user = "fred" | noop
search_optimization=false
```

关闭所有优化

您可以为所有用户关闭所有内置优化。

前提条件

- 只有具有文件系统访问权限的用户，如系统管理员才能关闭内置优化。
- 请参阅*《管理员手册》*中的"如何编辑配置文件"了解具体步骤。

步骤

1. 打开搜索应用的本地 `limits.conf` 文件。例如，`$SPLUNK_HOME/etc/apps/<app_name>/local`。

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。

2. 在 `[search_optimization]` 段落中，将 `enabled` 设为 `false`。

另请参阅

- 关于优化
- 优化的快速提示
- 编写更好的搜索

搜索标准化

当您在一串搜索字符串中使用 `search` 或 `where` 命令时，**SPL** 处理器可能会对命令后面的表达式语句重新排序以进行标准化处理。SPL 处理器应用两种正常化逻辑以搜索字符串：断言切换和断言排序。

有关断言和基于断言的搜索优化的更多信息，请参阅“内置优化”。

使用**任务查看器**查看搜索标准化和优化结果。请参阅“分析搜索优化”。

搜索标准化的益处

执行搜索标准化之后，某些搜索效果会更好。例如，断言合并优化不能合并将字段值放在字段名称前面的 `where` 语句。但是，如果 SPL 处理器会对这些语句应用断言切换标准化，这样字段名称会在字段值前面，断言合并优化就可以合并该语句。

请参阅“断言合并”。

断言切换标准化

在断言切换标准化的情况下，SPL 处理器会会采用有字段-值对的 `where` 语句，其中字段值在字段名称前面，处理器会对该等语句进行切换，这样字段名称会在前面。

例如，在此搜索中，字段值在字段前面：

```
index=main | where "error"=status
```

进行标准化后，字段名称和字段值位置切换：

```
index=main | where (status == "error")
```

仅当 SPL 处理器可区分字段名称和字段值的情况下，断言切换标准化才有用。SPL 处理器会将用括号括起来的数字字段值和字符串字段值放在操作符的右侧。SPL 处理器会尽可能切换值包含函数的值-字段组合，如 `value()=field`。

SPL 处理器不会对布尔值、时间和 IPv4 字段应用断言切换标准化。例如，对于类似 `true=purchased` 的布尔值-字段对，SPL 处理器无法区分是 `true` 还是 `purchased` 是字段名称。

断言排序标准化

在断言排序标准化情况下，SPL 处理器会使用字典顺序排序逻辑，确保 `search` 表达式和 `where` 语句始终以相同的方式排序。

搜索命令的断言排序

当您在字符串中使用 `search` 命令时，SPL 处理器会对命令后面的任意布尔表达式应用断言排序标准化。

例如，以下三种搜索使用带有布尔表达式的 `search` 命令。这些搜索看起来不同，但产生的结果是一样的：

```
| search ( z OR y AND d AND c AND b AND a )
```

```
| search ( d AND z OR y AND c AND b AND a )
```

```
| search ( d AND ( z OR y ) AND ( c AND b AND a ) )
```

标准化后，这些字符串会重新排序，这样可以共享以下表单：

```
| search ((y OR z) a b c d)
```

Where 命令的断言排序

当您在字符串中使用 `where` 命令时，SPL 处理器会对命令后面的任意布尔语句或算术语句应用断言排序标准化。

例如，这些 `where` 语句的数学表达式会产生相同的结果，但是排序不同：

```
| where x = (d+(c-a)+c*b)*b

| where b*(d+(c-a)+c*b) = x

| where ((b*c)+d+(c-a))*b = x
```

标准化后，这些 `where` 语句共享以下表单：

```
| where (x == ((((b * c) + (c - a)) + d) * b))
```

将断言切换和断言排序组合在一起的示例

以下示例将断言切换和断言分配组合在一起。进行标准化之前，您可以使用以下 `where` 语句：

```
| where status="error" OR code=500

| where "error"=status OR code=500

| where 500=code OR "error"=status
```

标准化后，这些 `where` 语句共享以下表单：

```
| where ((code == 500) OR (status == "error"))
```

禁用搜索标准化

如果您出于搜索性能考虑，将按特定顺序放置 `search` 表达式和 `where` 语句，您可能想要禁用搜索标准化。可通过 `limits.conf` 中的单独设置控制断言切换标准化和断言排序标准化。您可以禁用一种标准化，启用其他的。

前提条件

- 只有具有文件系统访问权限的用户，如系统管理员才能禁用搜索标准化。
- 请参阅*《管理员手册》*中的"如何编辑配置文件"了解具体步骤。

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。

禁用断言切换标准化

1. 打开搜索应用的本地 `limits.conf` 文件。例如，`$SPLUNK_HOME/etc/apps/<app_name>/local`。
2. 在 `[search_optimization::search_flip_normalization]` 段落中，设置 `enabled=false`。

禁用断言排序标准化

1. 打开搜索应用的本地 `limits.conf` 文件。例如，`$SPLUNK_HOME/etc/apps/<app_name>/local`。
2. 在 `[search_optimization::search_sort_normalization]` 段落中，设置 `enabled=false`。

检索事件

关于检索事件

您运行搜索，即是要在事件数据中找到与搜索术语匹配的片段。这些搜索术语为关键字、短语、布尔表达式、字段名称和值对等。它们用于指定您想要从索引中检索到的事件。请阅读搜索命令入门，了解如何有效使用搜索命令。

您的事件数据可能被划分到不同的索引和分布式搜索节点中。请参阅“从索引中检索事件”，了解如何跨多个索引和服务器搜索的更多信息。

事件按反向时间顺序从索引中检索而来。默认情况下，搜索结果按从最新到最旧进行排序。如果按时间过滤，检索事件的速度会更快一些，无论您是使用时间线在事件群集上缩小还是将时间范围应用到搜索本身。有关详细信息，请参阅“如何使用时间线调查事件”和“关于搜索中的时间范围”。

事件、事件数据和字段

在将数据添加到 Splunk 索引后，会用短语*事件数据*来代表这些数据。事件是指活动的单个记录或此事件数据的实例。例如，事件可能是日志文件中的单个日志项。由于 Splunk 软件按事件的时间信息分隔各个事件，因此事件之间的区分以时间戳为准。

下面是一个示例事件：

```
172.26.34.223 - - [01/Jul/2005:12:05:27 -0700] "GET /trade/app?action=logout HTTP/1.1" 200 2953
```

事件包含成对的信息或字段。在添加数据且为数据建立索引后，Splunk 软件会自动为您提取一些有用的字段，如事件的源主机以及事件的数据源类型。

使用字段检索事件

字段是**事件数据**中的可搜索名称/值对。所有字段都具有名称，因此可使用名称搜索相应字段。使用字段表达式搜索比仅使用关键字和带引号的短语搜索更精确（从而更高效）。

让我们看一下下面的搜索：

```
host=webserver
```

此搜索中，`host=webserver` 指示要搜索 `host` 字段值为 `webserver` 的事件。当您运行此搜索时，它不会检索带不同 `host` 字段值的事件，也不会检索字段值为 `webserver` 的其他事件。这意味着，与仅在搜索栏中搜索 `webserver` 时的结果相比，此搜索会返回一组更具体的搜索结果。

有关更多信息，请阅读《知识管理器》手册中的“关于字段”。

索引时间和搜索时间字段

在 Splunk 软件处理事件数据时，它会先后在索引时间和搜索时间从该数据中提取并定义字段。

请参阅《*管理索引器和群集*》手册中的“索引时间对比搜索时间”。

索引时间的字段提取

在**索引时间**内，Splunk 软件会提取一小组默认字段。该组字段包含**默认字段**、自定义索引字段，以及从结构性数据索引的字段。

默认字段存在于所有事件中。三个重要的默认字段是主机、数据来源和数据来源类型。这些字段会说明事件的数据来源。其他默认字段包括日期时间字段，这些字段为事件时间戳提供了额外的可搜索粒度。Splunk 软件还会自动添加分类为内部字段的默认字段。

自定义索引字段已针对索引时间提取进行手动配置。请参阅*《数据导入》*手册中的“在索引时间创建自定义字段”。

最后，当 Splunk 软件索引结构性数据时，会为所找到的字段创建索引时间字段提取。结构性数据的一些示例包括：

- 逗号分隔值文件 (CSV)
- 制表符分隔值文件 (TSV)
- 管道符分隔值文件
- JavaScript 对象符号 (JSON) 数据来源

当搜索默认字段值和自定义索引字段值时，可以使用标准的 `<field>=<value>` 语法。此语法匹配默认字段、自定义索引字段和搜索时字段。

然而，如果对于在索引时从结构性数据中提取的字段进行特殊搜索，您的搜索效率可以更高，但您必须将等号替换为

双冒号，如下：

```
<field>::<value>
```

此语法在搜索从结构性数据中建立索引的字段时表现最佳。然而，您还能将其用于搜索默认和自定义索引字段。您无法将其在**搜索时间**字段中搜索。

关于从结构性数据文件中提取字段的更多信息，请参阅*《数据导入》*手册中的“从具有标头的文件中提取数据”。

搜索时间的字段提取

在搜索时间，Splunk 软件会提取额外字段，具体取决于其**搜索模式**设置，以及对于所运行搜索的类型该设置是否启用字段发现。

搜索示例

示例 1：在所有 "corp" 服务器上搜索用户 "strawsky" 访问的事件。随后，报告了 20 个最近发生的事件。

```
host=corp* eventtype=access user=strawsky
```

在本示例中，`host` 是默认字段，而 `eventtype` 和 `user` 是可能自动提取的或者您已定义的其他字段。

通常，事件类型是一个用户定义字段，可让您对事件进行分类以简化搜索。您可以将搜索保存为事件类型，并使用 `eventtype` 字段快速检索这些事件。有关更多信息，请阅读《知识管理器》手册中的“关于事件类型”。

示例 2：搜索来自来源 "/var/www/log/php_error.log" 的事件。

```
source="/var/www/log/php_error.log"
```

事件的来源是事件来源的文件、流或其他输入的名称。

示例 3：搜索具有 Apache Web 访问来源类型的所有事件。

```
sourcetype="access_*"
```

事件的来源类型是事件来源的数据导入的格式。在此搜索中，使用通配符与以 "access_" 开头的任何 Apache Web 访问日志匹配。这包括 access_common 和 access_combined（您可能还会看到 access_combined_wcookie）。

示例 4：搜索从不同 CSV 文件中索引的信息，以获得基于普莱诺员工的列表。

```
employee_office::Plano
```

您已经索引员工记录的若干 CSV 文件。每个 CSV 文件共享相同的字段。您想要从关联德州普莱诺办公室的文件中搜索员工。

本例使用 `<field>::<value>` 语法来查找在索引时间内提取的 CSV 文件中的字段。虽然此语法亦可处理其他类型的索引时间字段，但是当处理从索引的结构性数据中提取的字段时表现最佳。无法查找搜索时间提取的字段。

示例 5：在 corp1 中搜索超过 4 行的事件，并忽略包含术语 400 的事件。

```
host=corp1 linecount>4 NOT 400
```

您可以使用比较表达式来匹配字段/值对。含有 "=" 和 "!=" 的比较表达式适用于所有字段/值对。含有 < > <= >= 的比较表达式仅适用于具有数字值的字段。本示例指定搜索行数为 4 行以上的事件，`linecount>4`。

示例 6：使用布尔 "NOT" 的搜索与使用比较运算符 "!=" 的搜索不同。以下搜索将返回 `field` 为未定义（或空值）的事件。

```
NOT field="value"
```

以下搜索将仅返回其中存在 `field`，并且没有值 "value" 的事件。

```
field!="value"
```

如果所涉及的值为通配符 "*"，则 `NOT field=*` 将返回字段为空/未定义的事件，而 `field!=*` 不会返回任何事件。

关于字段的更多信息

本主题只介绍一些使用字段的搜索。

- 您可以将搜索限制于特定索引；在分布式拓扑结构中，则可限制于特定搜索节点。
- 在《知识管理器手册》中可以看到“使用默认字段”的更多搜索示例。

当您开始使用 Splunk 搜索语言为数据建立摘要和将数据转换为报表时，字段就变得更为重要。有关更多信息，请阅

读“关于报表命令”。

事件示例

默认情况下，Splunk 搜索会检索所有事件。但是，在某些情况下，您可能想检索事件的示例集，而非整个事件集。您可能想使用事件示例的原因有以下几种。

- 执行快速搜索，以确保返回正确事件
- 判断一个大数据集的特性，而无须处理每个事件
- 测试数据选择、格式设置、计算和其他搜索组件都正常工作

对于大部分搜索，事件示例都可大幅提高搜索性能，而不会弱化其功能。

事件示例比

示例比指任何事件被包含于示例结果集中的可能性。计算该比值的公式为 `1/sample_ratio_value`。

例如，如果示例比值为 100，则每个事件有 1/100 的机率可能被包含于结果集中。每个事件的选择与所有其他事件的选择无关。很可能前 100 个事件中有很多事件都被包含进去，也可能一个都没包含。

如果未使用示例的事件匹配了 1,000,000 个事件，使用示例比值 100 会使搜索仅返回约 10,000 个事件。

如果您要示例搜索返回多次结果，则返回结果的确切数量由一个二项式分布建模，其中 n=1000000 且 p=0.01。此分布看起来像是一个普通分布，其中 mean=10000、标准偏差 (stdev)=99.5。

在 Splunk Web 中，您指定的示例比必须是一个大于 1 的正整数。要在 Splunk Web 中禁用示例，将此比值设为 1。

设置默认示例比

在 Splunk Enterprise 中，通过编辑 `ui-prefs.conf` 文件设置默认示例比。示例比必须是正整数。

在 Splunk Cloud 中，要更改默认示例比，请向 Splunk 支持提交问题。

事件示例如何工作

事件示例默认为不启用。当您运行一个搜索时，它会返回匹配您标准的所有事件。当您指定了一个比值时，示例会作用于活跃的搜索窗口。当您将搜索保存为报表或仪表板面板后，示例仍然有效。

在指定比值时，此值会覆盖 Splunk 部署所配置的默认值，并在下次修改之前一直有效。

如果您打开一个新搜索窗口，事件示例则不会继续保持启用状态。但是，您所使用的最后一个自定义比值会出现在**示例**下拉列表中。

事件示例需避免使用的命令和函数

通常，使用 `transaction`、`stats` 或 `streamstats` 命令的搜索并不适合采用示例。

当您使用示例事件集计算统计信息时，统计值可能并不准确。要确定正确的统计值，必须对事件示例返回的值进行放大。而这种放大也只能给出一个近似正确的值。

例如，您通过启用了事件示例的以下搜索来创建报表。

```
... | stats sum(x)
```

由于您启用了事件示例，返回的结果并非所有事件的完整总和，而只是示例事件集的总和。如果示例比是 100，真正的总和约为此搜索返回值的 100 倍。

会遇到这种问题的统计计算有 `count`、`sum` 和 `sumsq`。

当启用了事件示例时，其他难于解释的统计信息包括：

- `distinct_count`
- `earliest`
- `latest`
- `max`
- `min`

指定示例比

指定示例比的操作会激活搜索的事件示例。

1. 在 Splunk Web 中的搜索栏下方，单击**无事件示例**。

2.您可使用其中一个默认比值或指定一个自定义比值。

a. 若要使用其中一个默认比值，单击**示例**下拉列表中的所需比值。
b. 若要指定一个自定义比值，单击**自定义**并键入所需比值。然后单击**应用**。该比值必须为大于 1 的正整数。

事件示例的标志

在搜索和报表应用中有几个标志表示事件示例已启用。在运行搜索后，**示例**下拉列表出现在事件计数行。**示例**下拉列表的标签指定了应用于此搜索的比值。另外，如果应用了示例值，**任务**下拉列表会指定应用于此搜索的比值。

带有报表与仪表板面板的事件示例

您可以将使用事件示例的搜索保存为报表或仪表板面板。使用**另存为**下拉列表保存搜索。

在将搜索保存为报表后，此报表在运行时会应用其示例比。

在将搜索保存为仪表板面板后，此面板由内联搜索操纵。当刷新仪表板时，会使用和内联搜索一起保存的示例比。

如果您打开一个报表并将其添加到仪表板面板中，可以指定如何操纵此面板。可以指定此面板由此报表所基于的内联搜索操纵。也可以指定此面板由报表本身操纵。

由报表操纵的面板
　　当您在简单 XML 中查看面板数据来源时，不会有任何迹象表明此报表使用了事件示例。

由内联搜索操纵的面板
　　当您在简单 XML 中查看面板数据来源时，如果基本搜索使用了事件示例，则会有 <sampleRatio> 条目。例如：

```
<event>
  <title>sample events</title>
  <search>
     <query>buttercupgames</query>
     <earliest>@d</earliest>
     <latest>now</latest>
     <sampleRatio>500</sampleRatio>
  </search>
</event>
```

加速的报表
　　您无法加速基于事件示例搜索的报表。请参阅*《报表手册》*中的“加速报表”。

从索引中检索事件

您始终能够创建新索引和管理数据的存储位置。另外，当将数据拆分到不同的索引时，可以使用 `index` 字段一次搜索多个索引。

指定一个或多个索引进行搜索

Splunk 管理员能够设置用户搜索的默认索引。基于角色和权限，用户可能有权访问一个或多个索引。例如，该用户可能只能搜索主要索引或所有公共索引。然后，该用户可指定搜索这些索引的子集（单个索引或多个索引）。有关设置用户和角色的更多信息，请参阅*《确保 Splunk Enterprise 安全》*中的“关于用户和角色”。

有关管理索引和设置多个索引的更多信息，请参阅*《管理索引器和群集》*手册中的“关于管理索引”。

通过 Splunk Web 控制索引访问权限

1.导航到**管理器 > 访问控制 > 角色**。

2.选择已为用户分配的角色。

　　在接下来的屏幕底部您将看到索引控件。

3.控制特定角色有权访问的索引以及默认搜索索引。

语法

可按照指定字段名称和值的相同方式来指定要搜索的不同索引。在下例中，字段名称为 `index`，字段值为特定索引的名称：

```
index=<indexname>
```

您可使用 * 通配符指定多组索引；例如，如果想要同时搜索 "mail" 和 "main" 索引，可搜索：

```
index=mai*
```

还可使用括号将不同的搜索划分到特定索引中。有关详细信息，请参阅示例 3。

注意：在搜索栏中键入 "index=" 时，键盘缓冲表示基于您的角色和权限设置，您可以搜索的所有索引。

示例

示例 1：搜索所有公共索引。

```
index=*
```

示例 2：搜索所有索引：公共和内部。

```
index=* OR index=_*
```

示例 3：在不同的索引中划分不同的搜索；在本示例中，您要搜索三个不同的索引：main、_internal 和 mail。您想要查看全部三个索引中匹配 "error" 的事件，还有与 main 中的 "warn" 或 mail 中的 "failed" 匹配的错误。

```
(index=main (error OR warn)) OR (index=_internal error) OR (index=mail (error OR failed))
```

示例 4：在不同的分布式 Splunk 服务器中搜索多个索引。

```
(splunk_server=local index=main 404 ip=10.0.0.0/16) OR (splunk_server=remote index=mail user=admin)
```

未找到您要寻找的事件？

当您添加一个输入时，该输入将相对于您所在的应用程序进行添加。某些应用程序会写入输入数据至自身的特定索引（例如，Splunk App for Unix and Linux 使用 'os' 索引）。

在一个或多个分布式搜索节点中搜索

从搜索头执行分布式搜索时，默认情况下，可将您的搜索限制为特定搜索节点（也称为“索引器节点”）并限定为在您的已保存和已计划搜索中执行。Splunk 搜索节点的名称将另存为 "splunk_server" 字段中的值。有关分布式搜索的更多信息，请参阅《分布式搜索手册》中的“关于分布式搜索”。

如果未指定搜索节点，您的搜索将访问您有权访问的所有搜索节点。您可以访问的默认节点是由与您的配置文件相关联的角色和权限所控制，并由您的 Splunk 管理员设置。有关更多信息，请参阅《确保 Splunk Enterprise 安全》中的“关于用户和角色”。

默认情况下，当某些搜索节点存在高延迟现象，且您不想搜索它们时，将搜索限制为特定节点的功能将会非常有用。如果指定一个或多个节点，它们将是包括在搜索中的唯一服务器。

可按照指定其他字段名称和值的相同方式来指定要搜索的不同节点。在此状况下，字段名称为 "splunk_server"，字段值为特定分布式节点的名称：

```
splunk_server=<peer_name>
```

注意：您可使用值 "local" 来代表您要从中搜索的 Splunk 实例；换言之，即搜索头本身。

```
splunk_server=local
```

请记住，**字段名称区分大小写**；如果大小写不匹配，Splunk 不会识别出字段名称。

示例

示例 1：从指定搜索节点返回结果。

```
error (splunk_server=NYsplunk OR splunk_server=CAsplunk) NOT splunk_server=TXsplunk
```

示例 2：在分布式搜索节点 "foo" 或 "bar" 上搜索不同的索引。

```
(splunk_server=foo index=main 404 ip=10.0.0.0/16) OR (splunk_server=bar index=mail user=admin)
```

分类和分组类似事件

事件与事件类型并不相同。事件是数据的一个实例，例如，一个日志条目。事件类型是用于标记和分组事件的分类。

事件的匹配的事件类型的名称在事件的多值字段（名为 `eventtype`）中设置。您可按照搜索任意字段值那样来搜索这

些事件组（例如，SSH 登录）。

本主题介绍如何分类事件（将搜索另存为事件类型）及如何搜索添加标记的字段。有关事件、Splunk 识别事件的方式以及 Splunk 在建立索引时对事件所进行的具体处理操作的更多信息，请参阅*《数据导入》*手册中的“事件处理概述”主题。

重要提示：您不能将搜索管道另存为事件类型；即，将搜索另存为事件类型时，它不能包括搜索命令。

将搜索另存为新事件类型

搜索事件数据时，实际上您是在筛选所有不需要的事件。搜索结果是具有共同特性的事件，您可以为它们指定一个共同的名称。

例如，如果您经常搜索不同主机上的登录失败操作，则可将这些事件保存为一个事件类型，并命名为 `failed_login`：

```
"failed login" OR "FAILED LOGIN" OR "Authentication failure" OR "Failed to authenticate user"
```

将此搜索另存为事件类型：

1. 单击**另存为**并选择**事件类型**。
2. 在**另存为事件类型**窗口中，为您的搜索类型键入一个**名称**。
   在本例中，该名称为 **failed_login**。

   您还可以在**标记**字段中添加应该应用于事件类型的一系列标记。有关标记的更多信息，请参阅以下**使用标记分组和查找类似事件**。
3. 单击**保存**保存事件类型名称。

现在，您可以通过在搜索条件中指定事件类型，按照搜索任意字段的相同方式来快速搜索与此事件类型匹配的所有事件。

例如，您可能对在特定主机上查找失败的登录感兴趣。搜索可能如下所示：

```
host=target eventtype=failed_login
```

或者，您可能想要调查可疑用户的活动。搜索可能如下所示：

```
user=suspicious eventtype=failed_login
```

使用类型学习程序发现新事件类型

将任何搜索传递到 `typelearner` 命令可显示事件类型的建议。默认情况下，`typelearner` 将比较从搜索产生的事件的标点符号，从而将具有类似标点符号和术语的那些事件分组到一起。

您可为 Splunk 软件指定另一个不同字段来分组事件；`typelearner` 的工作方式与所有字段都相同。结果是搜索结果中具有共同的字段和短语的一组事件。

有关更多信息和示例，请参阅搜索命令参考中的“类型学习程序”。

使用标记分组和查找类似事件

在您的数据中，可能具有包含相关字段值的事件组。为帮助您更有效地搜索这些字段组，可为其字段值分配标记。可为任何提取的字段（包括事件类型、主机、数据来源或来源类型）分配一个或多个标记。

事件类型可有一个或多个标记与之关联。将搜索另存为事件类型时可在事件类型管理器中添加这些标记（位于**管理器 > 事件类型**）。在此窗口的事件类型列表中，选择您要编辑的事件类型。

将标记添加到事件类型后，可按照搜索任意标记的相同方式来搜索它们。假定您已将对防火墙事件的搜索另存为事件类型 `firewall_allowed`，然后将对登录事件的搜索另存为事件类型 `login_successful`。如果将这两种事件类型均标记为*允许*，则可通过使用以下搜索来检索任一事件类型的所有事件：

```
tag::eventtype="allow"
```

可为字段/值对设置标记。还可为字段名设置别名。请参阅 Splunk Web 中的“搜索和创建字段别名”中的“为字段值对设置标记”。

搜索加标记的字段值

可使用两种方法来搜索标记。如果要搜索与任何字段中的值相关联的标记，可使用以下语法：

```
tag=<tagname>
```

或者，如果要搜索与特定字段中的值相关联的标记，可使用以下语法：

```
tag::<field>=<tagname>
```

使用通配符搜索标记

搜索关键字和字段值（包括事件类型和标记）时，可使用星号 (*) 通配符。

例如，如果具有不同类型 IP 地址的多个事件类型标记（如 `IP-src` 和 `IP-dst`），可使用以下语法来搜索所有这些事件：

```
tag::eventtype=IP-*
```

如果想要找到标记包含 "local" 的所有主机，可搜索标记：

```
tag::host=*local*
```

另外，如果想要搜索事件类型无标记的事件，可搜索布尔表达式：

```
NOT tag::eventtype=*
```

使用时间线调查事件

时间线是在每个时间点上发生的搜索结果中事件数的虚拟表示。时间线显示事件随时间的分布情况。

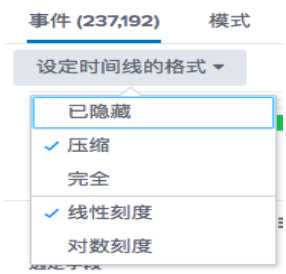
当您使用时间线调查事件时，您没有运行新搜索。您在筛选现有的搜索结果。

可使用时间线来突出显示事件的模式或群集，或者调查事件活动中的高值（活动高峰）和低值（可能出现的服务器停机）。将鼠标放在在一个栏位上方可查看事件计数。单击某个栏位可钻取到该时间范围。

更改时间线格式

时间线位于事件列表上方“事件”选项卡内。它显示搜索运行时间范围内事件的计数。这里，时间线显示**所有时间**内的 web 访问事件。

格式选项位于**格式时间线**菜单中：



可隐藏时间线或显示时间线“紧凑”或“完整”视图。您还可以在线性刻度或对数刻度（对数）之间切换时间线刻度。

选择**完整**后，时间线视图比较高，以容纳轴上的标签。计数在 Y 轴上，时间在 X 轴上。

缩小和放大来调查事件

时间线上方是缩放选项。默认情况下，时间线是放大的。下图显示“完整”视图中显示的放大时间线。可以使用**缩小**选项。

时间线图例

时间线图例位于时间线右上角。图例表示时间线的刻度。例如**每列 1 小时**表示每列都显示此小时内的事件计数。放大和缩小可更改时间刻度。

缩小

单击**缩小**，图例指示每列现在代表**每列 1 天**而不是一小时。

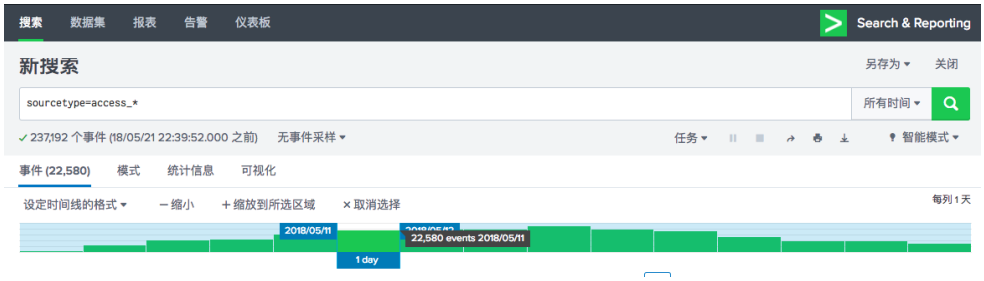
缩小不仅会更改时间线，还会更改时间范围挑选器中的值。

重置缩放

要重置缩放或缩小，请更改时间范围挑选器中的值。例如，如果您使用**所有时间**搜索，然后放大，请选择时间范围挑选器中的**所有时间**，以返回值原始时间线时间刻度。

缩放到所选区域

移动鼠标并选择时间线上的栏位后，时间线上方的**缩放到所选区域**或**取消选择**选项可用。



移动鼠标，点击最高栏或将鼠标移到时间线中的一组栏位。事件列表会更新，只显示在该选定时间范围内发生的事件。时间范围挑选器也将更新到所选时间范围。可通过单击**取消选择**取消此选择。

选择时间线上的一组栏然后单击**缩放到所选区域**之后，会将您的搜索结果过滤为仅显示所选时间范围。时间线和事件列表将更新，以显示所选区域结果。

与所选栏对应的日期和时间以及该时间范围内的事件数量反映在搜索栏正下方信息中。

缩放到所选时间范围后，不能**取消选择**。但是，您可以再次**缩小**或更改时间范围挑选器中的时间。

钻取事件详情

运行在**事件**选项卡中返回事件的搜索后，单击这些事件中的一部分，以运行使用您所选事件详细信息的不同类型的辅助**钻取**搜索。

在**事件**选项卡中，您单击事件的这些部分后，将能够运行钻取搜索：

- **字段值**。从索引时间和搜索时间的事件中**提取**字段。请参阅*《知识管理器手册》*中的“关于字段”。
- **标记**。一个**标记**和一个或多个字段-值对关联。一个字段-值对可以与多个标记相关联。请参阅*知识管理员手册*中的“关于标记和别名”。
- **段**（可以是连接的段字符串）。**段**是事件的可搜索部分。请参阅*《数据导入手册》*中的“关于事件分段”，了解配置和创建段的方法。
- **时间戳**。**时间戳**是事件中的日期和时间信息。

钻取搜索动作

钻取搜索可为字段、标记和事件段执行下列动作。

钻取搜索动作	描述	结果
添加到搜索	使原始搜索字符串更侧重于所选事件详细信息并运行搜索。Splunk 软件丢弃搜索字符串中的**转换命令**，以及这些命令之后的所有内容。	与原始搜索返回的数据集类似的数据集，经筛选，仅包括具有所选字段、标记或段的事件。
从搜索中排除	从原始搜索中排除所选事件详细信息，并运行搜索。Splunk 软件丢弃搜索字符串中的转换命令，以及这些命令之后的所有内容。	与来自原始搜索的数据集类似的数据集，经筛选，仅包括没有所选字段、标记或段的事件。
从搜索中移除	运行一个与原始搜索一样的新搜索，只是新搜索不再搜索高亮显示的字段值、标记或段。Splunk 软件丢弃搜索字符串中的转换命令，以及这些命令之后的所有内容。	限制性不如原始搜索返回的数据集那么高的新数据集。它不再排除不带高亮显示的字段值、标记或段的事件。
新搜	执行仅侧重所选字段、标记或段的新搜索。	包含带所选字段、标记或段的所有事件的新数据集

索		新数据集。

钻取搜索动作示例

这些示例将使用《搜索教程》中的示例数据，但应与任意格式的 Apache Web 访问日志结合使用。要在您自己的 Splunk 实例上尝试这些示例，您必须下载样本数据，然后按照说明**将教程数据导入 Splunk**。运行搜索时使用时间范围**过去 7 天**。

添加到搜索

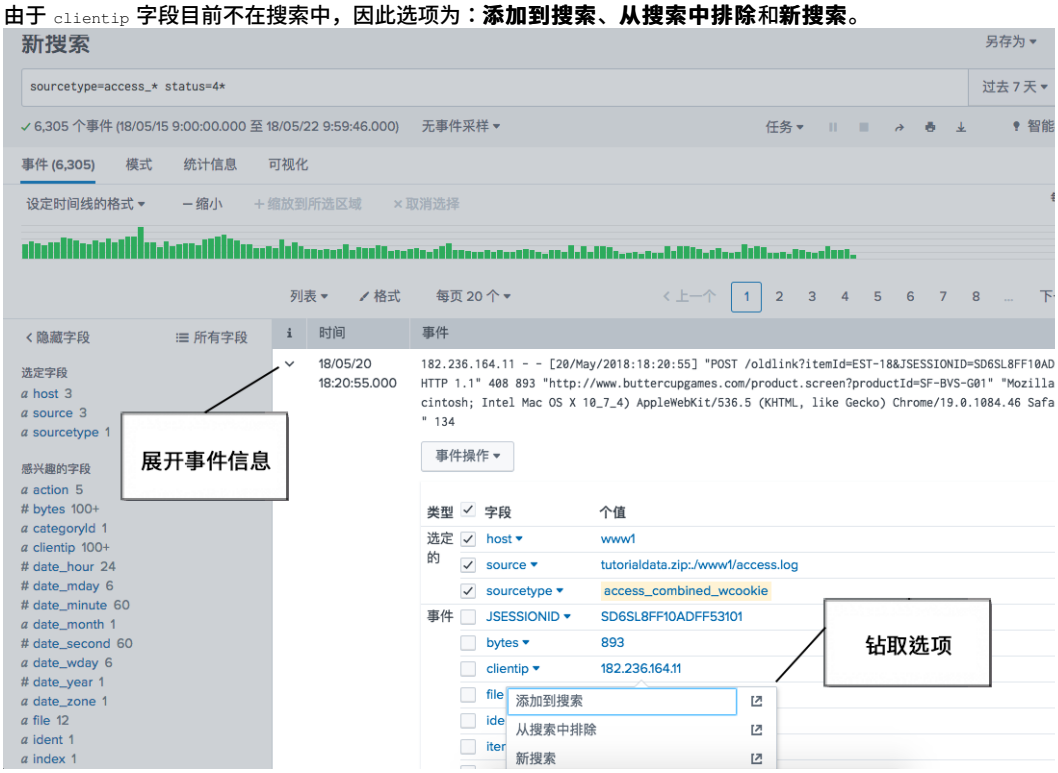
1. 运行可在**事件**选项卡中返回事件列表的搜索或报表。
   如果搜索包含**转换命令**，将**搜索模式**设置为**详细**。

   让我们从搜索 Apache Web 访问日志文件和查找以 4 开头（例如 404）的 HTTP 错误的基本搜索开始。

   ```
   sourcetype=access_* status=4*
   ```

2. 在搜索结果中，扩展第一个时间的事件信息。
3. 选择 `clientip` 字段值。在本例中，值为 `182.236.164.11`。

   由于 `clientip` 字段目前不在搜索中，因此选项为：**添加到搜索**、**从搜索中排除**和**新搜索**。



   搜索字符串中显式包含的字段值、标记或段会以黄色高亮显示在事件信息中。
4. 单击**添加到搜索**。将字段-值对添加到搜索。使用相同的时间范围运行搜索。例如：

   ```
   sourcetype=access_* status=4* clientip="182.236.164.11"
   ```

从搜索中移除

您可以使用**从搜索中移除**钻取选项移除特定条件。您无法移除包含通配符的条件。

继续使用同一个搜索，将 `clientip` 从搜索中移除。

1. 要将条件从搜索中移除，扩展第一个搜索结果的事件信息。
2. 选择 `clientip` 字段值。钻取选项为**从搜索中移除**和**新搜索**。

3. 单击**从搜索中移除**。使用相同的时间范围更新和运行搜索。例如：

```
sourcetype=access_* status=4*
```

假设原始搜索如下所示：

```
sourcetype=access_* status=404
```

扩展包含 `status` 条件的任何事件信息。单击状态字段的值 `404`。从钻取选项中，选择**从搜索中移除**。

从搜索中排除

您可以使用**从搜索中扩展**钻取选项从搜索中排除特定条件。

1. 扩展包含您想要排除条件的搜索结果的事件信息。
   假设原始搜索如下所示：

   ```
   sourcetype=access_* status=4*
   ```

   您想要排除所有**查看**操作。

2. 关于 `action` 字段，选择**查看**值然后单击**从搜索中排除**。使用**不等于**语法将字段-值对添加到搜索。使用相同的时间范围运行搜索。例如：

   ```
   sourcetype=access_* status=4* action!=view
   ```

选择段

在单击段之前，选择事件段或所连接的段集。Splunk 软件可以识别出黄色高亮显示的段选择。

常见值

如果您选择的值在为其字段找到的前十个值中，**添加到搜索**和**从搜索中排除**选项将显示选项名称下方的事件数量。

- **添加到搜索**下方的事件数量表示包含该字段值的事件数量。
- **从搜索中排除**下方的事件数量表示不包含该字段值的事件数量。

在新选项卡中运行钻取搜索

选择一个钻取选项之后，搜索会在当前选项卡中运行并取代当前搜索。

但是，您可以在新选项卡中运行钻取搜索，并保持当前搜索结果不变。要在新选项卡中运行钻取搜索，单击选项的**在新选项卡中打开**图标。

事件时间戳钻取搜索

事件时间戳钻取搜索可帮助您找出事件相关性，并执行根源分析。

单击事件**时间戳**并运行可返回与该事件时间上相近的事件的辅助搜索。

当您打开事件时，您也可以单击 `_time` 字段以运行此类钻取搜索。

此搜索的控制称为 **_time accelerator**。请参阅本手册中“用时间查找附近事件”，了解如何使用 _time accelerator 的详细信息。

通过“模式”选型卡识别事件模式

搜索结果中的事件可以分组到**事件模式**中。属于同一事件模式的事件拥有相同的特性，通常可由特定搜索字串返回。事件模式分析对返回不同范围事件的搜索有帮助，因为它可以快速显示出搜索结果数据集中最常见类型的事件。

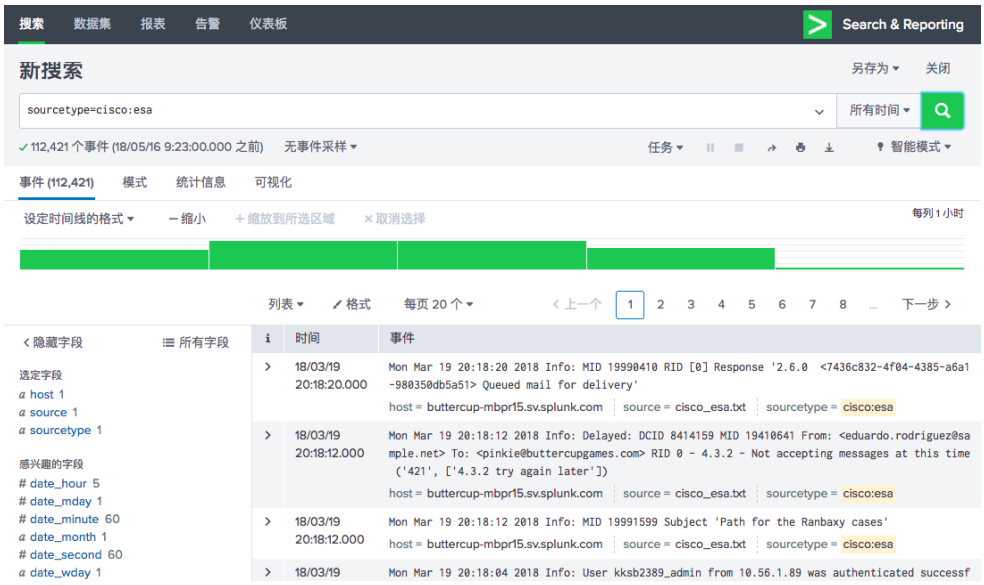
模式选项卡简化了事件模式识别过程。单击模式选项卡，可查看搜索返回的事件组中最常见模式的列表。每个模式都代表着拥有类似结构的一组事件。

单击模式，以：

- 查看符合模式的结果中事件的大致数量。
- 查看返回这种模式事件的搜索。
- 如果可行，将模式搜索保存为事件类型。并非所有事件模式都可保存为事件类型。
- 根据模式创建告警。例如，您可以创建告警，在特定模式很少增加或减少时触发。

事件模式示例

使用 `sourcetype=cisco:esa` 的搜索会在所有时间内运行并返回 112,421 个事件。



此模式是基于 50,000 个事件示例。

要查看事件列表中的所有模式，请单击**模式**选项卡。最初结果识别了 45 个模式。您可以将较小的滑块移动到更大的滑块，以更改事件模式的范围。如果您将滑块拖到**较大**侧，将返回 14 个模式。

威胁级别事件模式是最常见的事件模式。一些所列模式在此数据集中相对少见，可能不容易在事件选项卡中找到它们。模式选项卡有助于查看这些事件模式，并在需要情况下，将其保存为事件类型。

模式选项卡如何运作

单击“模式”选项卡后，Splunk 软件对截止此时接收到的搜索结果子集运行辅助搜索。此搜索任务将分析这些结果，并派生出这些结果中最常见的事件模式。然后，它将以降序顺序（从最普遍到最不普遍）列出这些模式。其中可能不包括基于非常小的事件组的离群模式，因为它们的统计意义不大。

原始数据集包含大量不同事件模式时，辅助搜索将需要很长时间才能完成。例如，某些搜索返回的数据集包含 500 多种模式，其中大多数模式代表了非常小的事件集合。识别这些模式的算法经过特别设计，以避免针对小模式进行这

些复杂步骤，但也尽量做到准确。

当**搜索模式**设置为**详细**时，模式选项卡仅接受**转换搜索**。模式选项卡无法找到其他搜索模式下的**实时搜索**适用的模式。

事件模式关键字

模式选项卡通过一个或多个关键字的存在与否来定义模式。如果为某模式识别的关键字添加到原始搜索，或从原始搜索中排除，搜索将返回符合该模式的事件。模式中的关键字以绿色文本形式，在模式列表中显示。事件列表中不会标识排除的关键字。

在前面的事件模式示例中，威胁级别事件模式的关键字为“威胁”，这意味着返回符合模式事件的搜索形式为：

```
sourcetype=cisco_esa threat
```

如果此事件模式也通过排除关键字“已验证”进行识别，返回符合模式事件的搜索形式为：

```
sourcetype=cisco_esa threat ( NOT verified )
```

要查看与模式相关联的所有关键字，单击模式。

使用模式选项卡

1. 从搜索视图中运行可返回 5,000 多个结果的搜索。

能够返回 5,000 多个结果的搜索可产生可靠的模式。

2. 单击模式选项卡。

您无需等待搜索完成，但通过最终完成的搜索结果可产生更准确的模式列表。

3. （可选）如果模式过多或过少，或如果未看到期望的模式，可移动滑块。

将滑块拖到**较大**端，将运行辅助搜索任务，合并某些模式，从而在每个模式组中产生代表更多事件的事件模式，以及更多种类的事件。

将滑块滑动到**较小**端，将运行增加结果粒度的辅助搜索任务。它找出的事件模式代表较少数量的事件。

4. （可选）单击模式，查看该模式的信息。

估计事件是数据集中由原始搜索返回、符合事件模式的事件的估计数量。在此示例中，原始搜索有 112,000 个事件。此模式约占事件总数中的 10.84%（12,200 个）。

包含的关键字可以识别能够添加到基本搜索以返回模式的关键字。如果模式选项卡识别出应从基本搜索中排除的关键字，它们将在**排除的关键字**部分中显示。

可在**搜索**下查看返回符合事件模式的搜索。

5. （可选）在模式信息区域，单击**查看事件**以运行**搜索**下显示的搜索。

搜素时，此搜索使用的时间范围与原始搜索的时间范围相同。

6. （可选）在模式信息区域，单击**另存为事件类型**，以将搜索保存为**事件类型**。

另存为事件类型仅对基于特定搜索（不含管道符和附加搜索命令）的事件模式可用。请参阅本手册中的“关于事件类型”部分。

7. （可选）在模式信息区域，单击**创建告警**，以基于模式创建一个**告警**。

例如，创建一个当事件模式频率高于或低于阈值时触发的计划的告警。如果知道符合某事件模式的事件出现的频率较稳定，例如约为每小时 100 个事件，可将告警设置为按小时计划运行，在返回 150 或更多事件时触

发。请参阅*《告警手册》*。

模式选项卡中的数字

辅助搜索完成后，模式选项卡显示一条消息，表明它获得显示结果所分析的事件数量。

模式选项卡分析原始搜索返回事件总数的一部分。这部分中事件数量最大值为 50,000。此最大值将降低模式选项卡辅助搜索的处理时间。如果原始搜索返回事件数量低于 50,000，辅助搜索将最多分析原始搜索时间跨度内每个时间线栏的 1000 个事件。例如，如果原始搜索跨 14 个时间线栏，辅助搜索将分析 14,000 个事件，以获得其模式列表。

可通过更新 `max_events`（位于 `limits.conf` 中）来控制辅助搜索分析的事件总数。默认设置为 `50000`。请勿更改此值。数量低于 50,000 将降低事件的准确性。数量高于 50,000 将增加辅助搜索所需的处理时间。

模式信息区域提供的预计事件数量不适用于模式选项卡辅助搜索中分析的事件数量。它适用于原始搜索返回的事件总数。如果事件模式预计表示 7,350 个事件，原始搜索返回 265,000 个事件，模式则占搜索返回事件的 2.7%。

限制模式选项卡使用

默认情况下，所有角色（包括用户角色）都可以使用模式选项卡。要限制模式选项卡的使用，移除 `pattern_detect` **操作**。没有此操作的角色运行搜索后，不会显示模式选项卡选项。

有关功能的更多信息，请参阅*《确保 Splunk 安全》*手册中的"关于使用功能定义角色"。

预览事件

默认情况下，当您在分布式环境中运行一个搜索时，会在所有搜索节点都开始返回所指定时间范围内的事件数据时才会显示搜索结果。在包含大量节点的分布式环境中或当部分节点速度很慢时，搜索结果的显示可能会有所延迟。

事件预览模式会在事件返回后即显示此事件，而不用等所有事件都返回了才显示搜索结果。此模式显示内存中尚未提交的事件。

使用预览模式的限制

启用事件预览模式时，事件查看器中有一些限制。

您无法展开事件查看器来查看某个事件的具体信息，除非搜索的所有事件均已返回。将鼠标移到信息图标上方时，会出现一个消息，告诉您事件预览模式已启用。

随着结果的不断返回和列出在事件查看器中，事件的顺序会改变。当新结果会添加到事件查看器时，这些事件会按正确的时间顺序插入到相应位置。

事件查看器提供一个选项，用于决定以列表、表格还是原始事件的形式显示事件。当事件查看器设置为**表格**且事件预览模式已启用时，在此搜索完成之前您无法排列事件列表的顺序。

启用事件预览模式

若想更快的查看事件，可以在搜索应用中启用事件预览模式。

启用事件预览的操作会应用于搜索应用的所有用户，而不仅仅只应用于您自己。

前提条件

- 只有具有文件系统访问权限的用户，如系统管理员才能启用事件预览模式。
- 请参阅*《管理员手册》*中的"如何编辑配置文件"了解具体步骤。

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。在`本地`目录进行更改。

步骤

1. 打开搜索应用的本地 `limits.conf` 文件。例如，`$SPLUNK_HOME/etc/apps/<app_name>/local`。
2. 在 [search] 段落下，将 `timeline_events_preview` 设置为 `true`。

如果您使用的是 Splunk Cloud 并想启用预览模式，请向 Splunk 支持提交问题。

指定时间范围

关于涉及时间的搜索

如何使用时间戳

时间戳处理是**事件处理**中的一个关键步骤。Splunk 软件使用**时间戳**执行下列操作：

- 按时间关联**事件**
- 在 Splunk Web 中创建时间线直方图
- 为搜索设置时间范围

Splunk 软件在**索引时间**为事件添加时间戳。软件使用在原始事件数据中找到的信息自动分配时间戳值。请参阅*数据导入*中的“时间戳分配如何工作”。

时间戳格式

Splunk 软件有时表示 UNIX 时间。以此方式表示的时间会显示为一系列数字，如 `1518632124`。您可以使用任何 UNIX 时间转换器将 UNIX 时间转换为 GMT 时间或当地时间。

指定较短的时间范围

开始新搜索时，默认时间范围是**过去 24 小时**。此范围可避免运行搜索事件过于宽泛，从而避免浪费系统资源，以及生成超出实际需求的结果。

无论您是运行新搜索、报表还是创建仪表板，将事件范围缩小至您真正需要的日期或时间至关重要。

时间对于确定发生错误的内容也同样重要。即使不是确切知道发生了什么，您通常也会知道何时发生了错误。查看大约与出错在同一时间发生的事件有助于关联结果并找出问题的根源。

此部分讨论如何使用时间缩小搜索范围以及如何按时间对您搜索中的事件进行分组。

- 选择要应用于搜索的时间范围
- 在搜索条件中指定时间调节器
- 在实时搜索中指定时间窗口
- 使用时间查找附近事件

选择要应用于搜索的时间范围

使用**时间范围挑选器**对搜索设置时间限制。可以使用预设时间范围、创建自定义时间范围，以及基于日期指定时间范围或日期和时间的方式对搜索进行限制，也可以使用时间范围挑选器的高级功能。这些选项会在下面的部分中进行介绍。

注意：如果您所在的时区与服务器时区不同，基于时间的搜索将使用建立数据索引的 Splunk 实例中的事件时间戳。

从预设时间范围列表中选择

时间范围挑选器包括了许多内置时间范围选项，这些选项已在文件 `times.conf` 中进行了定义。可以从“实时”窗口、“相对”时间范围的列表中选择，并在“所有时间”内搜索。

定义自定义相对时间范围

使用“相对”时间范围选项为与“现在”或“当前小时开始”相关的搜索指定自定义时间范围。您可以从时间范围单元列表中选择：秒前、分钟前等等。

默认情况下，“最早”设为**未对齐到**，“最晚”设为**现在**。如果您为**最早**或**最晚**指定对齐到选项，时间范围将对齐到您选定的时间期间的开头。例如，如果您选择**天前**，最早时间将对齐到**今天开始**值。

字段下方的预览栏更新为您选择的时间范围。

要了解关于相对时间范围的更多信息，请参阅“在搜索中指定时间调节器”。

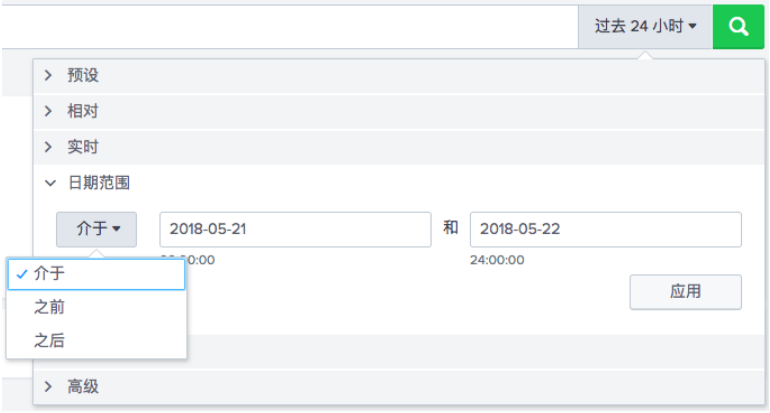
定义自定义实时时间范围

您可使用“实时”选项为**实时搜索**指定自定义最早时间。由于此时间范围是针对实时搜索的，因此最晚时间是不相关的。

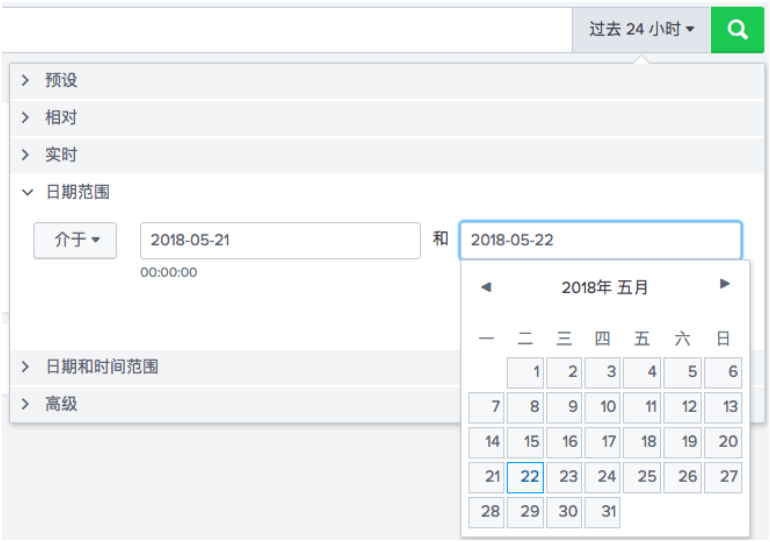
有关实时搜索的时间范围的更多信息，请参阅“在搜索中指定实时时间范围窗口”。

定义自定义日期范围

使用“日期范围”选项在搜索中指定自定义日历日期。您可以在以下选项中选择来返回事件：**介于**开始和结束日期之间，某个日期**之前**和某个日期**之后**。

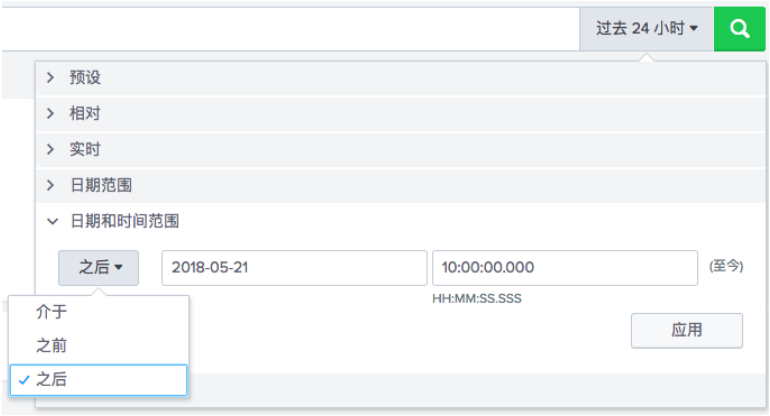


对于上述字段，您可以将日期键入文本框，或从日历中选择日期。



定义自定义日期和时间范围

使用“日期和时间范围”选项为搜索的开始和结束指定日历日期和时间。



可以将日期键入文本框，或从日历中选择日期。

使用高级时间范围选项

使用“高级”选项指定最早和最晚搜索时间。可以通过 Unix 时间或相关时间符号（如 `-3d@d`）填写时间。您键入的 UNIX 时间值将转换为本地时间。

您指定的 UNIX 时间或相对时间以时间戳的形式显示在文本字段下方，这样您可以验证您输入的内容。

自定义预设时间范围列表

Splunk Web 时间范围挑选器中的**预设**列表中显示的一组时间范围可以进行自定义。可以基于已有时间范围创建一个新的时间范围，也可以隐藏时间范围。

基于已有时间范围创建一个新的时间范围

创建新时间范围最简单的方式是使用已有时间范围作为基础来创建新的。例如，相对时间范围列表包含一个名为**过去 15 分钟**的时间范围。您想创建一个过去 30 分钟的时间范围。首先可以复制**过去 15 分钟**的时间范围。在复制时，将**最早**设置从 -15min 改为 -30min。

1. 从**设置**菜单，在知识列表下方选择**用户界面**。
2. 在用户界面窗口中，选择**时间范围**。
3. 找到要使用的时间范围。
4. 在操作列中，单击**复制**。
5. 随后会显示此时间范围规格的副本。针对此时间范围规格进行所需修改，然后单击**保存**。

新时间范围出现在预设菜单的相对列表中。

创建新预设时间范围

您可以为预设菜单创建新的时间范围。例如，您要创建一个时间范围，用于显示昨天从 12:00 到 15:00 这几个小时内的搜索。您需要在最早和最晚字段中指定相对时间。在**最早**字段中，指定 `-1d@d+12h`。在**最晚**字段中，指定 `-1d@d+15h`。

1. 从**设置**菜单，在知识列表下方选择**用户界面**。
2. 在用户界面窗口中，选择**时间范围**。
3. 单击**新建**。
4. 完成“添加新建”窗口中的字段，然后单击**保存**。

新时间范围出现在预设菜单的相对列表中。

隐藏预设列表中的时间范围

1. 从**设置**菜单，在知识列表下方选择**用户界面**。
2. 在用户界面窗口中，选择**时间范围**。
3. 找到要隐藏的时间范围。在状态列中，单击**禁用**。

为 API 或 CLI 设置默认时间范围

如果您想要为 REST API 端点或为命令行界面 (CLI) 指定时间范围，您可以在 `times.conf` 文件中手动设置时间范围。

前提条件

- 只有具有文件系统访问权限的用户，如系统管理员才能在 `times.conf` 文件中手动更改时间范围。
- 请参阅*《管理员手册》*中的“如何编辑配置文件”了解具体步骤。

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。在`本地`目录进行更改。

步骤

1. 打开搜索应用的本地 `times.conf` 文件。例如，`$SPLUNK_HOME/etc/apps/<app_name>/local`。
2. 为要指定的时间范围创建一个段落。若需相关示例，请参阅*《管理员手册》*中的 times.conf 参考。

如果您使用的是 Splunk Cloud 并想隐藏一个时间范围或创建一个新时间范围，请向 Splunk 支持提交问题。

更改默认时间范围

“搜索和报表”应用中的临时搜索默认时间范围设置为**过去 24 小时**。管理员可以全局设置默认时间范围，涵盖所有应用。请参阅*《管理员手册》*中的“更改默认值”。

在搜索中指定时间调节器

搜索或保存搜索时，可使用以下时间调节器指定绝对和相对时间范围：

```
earliest=<time_modifier>
latest=<time_modifier>
```

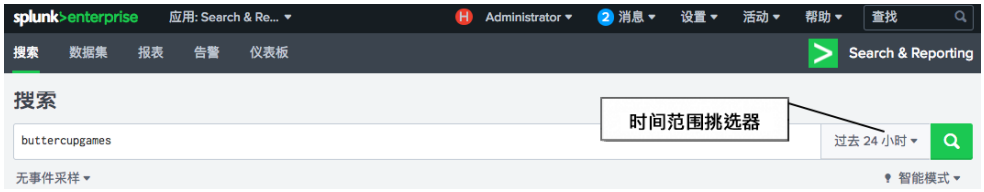
绝对时间范围使用具体的日期和时间，例如从 2017 年 11 月 1 日上午 12 点到 2017 年 11 月 13 日上午 12 点。

相对时间范围取决于搜索的运行时间。例如，相对时间范围 -60m 表示“60 分钟前”。如果现在时间是下午 3 点，则此搜索会返回前 60 分钟，即今天下午 2 点到 3 点之间的事件。

当前时间指**现在**。

时间调节器和时间范围挑选器

在搜索栏或保存的搜索中指定的时间范围会覆盖在时间范围挑选器中所选的时间范围。



时间范围和子搜索

您在搜索中直接指定的时间范围仅适用于主搜索。在主搜索中指定的时间范围不适用于子搜索。同样地，在子搜索中指定的时间范围仅适用于该子搜索。该时间范围不适用于主搜索或任何其他子搜索。

从时间范围挑选器中选择的时间范围会适用于主搜索或子搜索。

指定绝对时间范围

对于精确的时间范围，时间调节器的语法为 `%m/%d/%Y:%H:%M:%S`。例如，以下搜索指定了时间范围，从 2017 年 10 月 19 日上午 12 点到 2017 年 10 月 27 日上午 12 点。

```
earliest=10/19/2017:0:0:0 latest=10/27/2017:0:0:0
```

如果仅指定 `earliest` 时间调节器，默认设置会将 `latest` 设置为当前时间（**现在**）。如果指定了 `latest` 时间调节器，那么必须指定 `earliest` 时间。

指定相对时间范围

您可以在搜索中使用表示时间量的字符串来定义相对时间。语法是一个整数加一个时间单位。

1. 在字符串的最前面加上减号 (-) 或加号 (+) 来表示时间量往前或往后的偏移。

2. 用一个数字加一个时间单位来指定时间量。指定一个时间量时，数字将被隐含。例如，`s` 等同于 `1s`，`m` 等同于 `1m`，诸如此类。下表列出了所支持的时间单位。

时间范围	有效值
秒	s, sec, secs, second, seconds
分	m, min, minute, minutes
时	h, hr, hrs, hour, hours
天	d, day, days
周	w, week, weeks
月	mon, month, months
季	q, qtr, qtrs, quarter, quarters
年	y, yr, yrs, year, years

指定相对时间时，可以使用 `Now` 来代表当前时间。

对齐到时间的相对时间调节器

有了相对时间，您可以指定**对齐到**时间，此时间指相对于相对时间的偏移量。对齐到时间单位针对您指定的时间量向下舍入到最早或最晚时间。要这样做，用 "@" 字符将时间量与“对齐到”时间单位分开。

对齐到时间单位的语法为 `[+|-]<time_integer><time_unit>@<time_unit>`。

当对齐到最早或最晚时间时，Splunk 软件始终向后对齐或向下舍入到最晚时间，而不晚于指定时间。例如，当前时间是 15:45:00，那么对齐到 `earliest=-h@h`。时间调节器对齐到 14:00。

也可以将相对时间调节器定义为仅使用对齐到时间单位。例如，要对齐到一周中的特定一天，使用 @w0 代表星期日、@w1 代表星期一等等。对于星期日，可以指定 w0 或 w7。

如果没有指定对齐到时间单位，搜索则会使用秒作为对齐到时间单位。

对齐到选项在很多情况下都非常有用。例如，如果您想要搜索前一个月的事件，则指定 `earliest=-mon@mon latest=@mon`。在此示例中，起始时间为上月月首，结束时间为当月月首。

相对时间和相对对齐到时间的差别

4 月 28 日，您决定在 14:05 运行搜索。

- 如果指定 `earliest=-2d`，则此搜索会正好追溯到两天前，即于 4 月 26 日下午 14:05 开始运行。
- 如果指定 `earliest=-2d@d`，则此搜索会追溯到两天前，并对齐到那天的开始时间。搜索从 4 月 26 日 00:00 开始查找事件。

特殊时间单位

以下缩写为预留给特殊情况下的时间单位和对齐时间偏移。

时间单位	描述
`earliest=1`	如果想要搜索自 UTC epoch 时间起的事件，使用 `earliest=1`。（搜索字符串中的 `earliest=0` 表示搜索中没有使用时间。） 当 `earliest=1` 且 `latest=now` 或 `latest=<a large number>` 时，搜索将针对**所有时间**运行。差别在于： • 指定 `latest=now()`（此为默认值）不会返回将来事件。 • 指定 `latest=<a big number>` 会返回将来事件，即包含晚于当前时间 `now()` 的时间戳的事件。
`latest=now()`	指定搜索在当前时间开始或结束。
`@q, @qtr, or @quarter`	指定对齐到最近季度的开始日期：1 月 1 日、4 月 1 日、7 月 1 日或 10 月 1 日。
`w0, w1, w2, w3, w4, w5, w6, and w7`	指定“对齐到”**星期几**；其中 w0 为星期日、w1 为星期一，以此类推。当对齐到周时（`@w` 或 `@week`），相当于对齐到星期日或 `@w0`。您可以使用 w0 或 w7 表示星期日。

相对时间调节器示例

在这些例子中，当前时间是 2017 年 2 月 5 日星期三下午 1:37:05。另请注意，24 小时通常是 1 天，但是由于夏令时间界限的原因，有时并不是 1 天。

时间修饰符	描述	生成的时间	等效调节器
now	现在、当前时间	2017 年 2 月 5 日，星期三，下午 01:37:05	now
-60m	60 分钟前	2017 年 2 月 5 日，星期三，下午 12:37:05	-60m@s
-1h@h	1 小时前，对齐到小时	2017 年 2 月 5 日，星期三，12:00:00	
-1d@d	昨天	2017 年 2 月 4 日，星期二，上午 12:00:00	
-24h	24 小时前（昨天）	2017 年 2 月 4 日，星期二，下午 01:37:05	-24h@s
-7d@d	7 天前、1 周前的今天	2017 年 1 月 28 日，星期三，上午 12:00:00	
-7d@m	7 天前，对齐到分钟界限	2017 年 1 月 28 日，星期三，下午 01:37:05	
@w0	本周起始日	2017 年 2 月 2 日，星期日，上午 12:00:00	
+1d@d	明天	2017 年 2 月 6 日，星期四，上午 12:00:00	
+24h	从现在起的 24 小时、明天	2017 年 2 月 6 日，星期四，下午 01:37:05	+24h@s

链接相对时间偏移的示例

您还可以指定“与对齐到时间之间的偏移”或者与时间修饰符结合使用以获得更具体的相对时间定义。

时间修饰符	描述	生成的时间
@d-2h	对齐到今天的起始钟点（上午 12 点），并从该时间中减去 2 小	昨晚下午 10 点

@d-2h	时。	昨晚下午 10 点。
-mon@mon+7d	一个月前，对齐到当月第一天的午夜，然后加上 7 天。	上个月 8 号（上午 12 点）。

使用相对时间调节器搜索的示例

示例 1：从本周起始日到搜索的当前时间（现在）的 Web 访问错误。

```
eventtype=webaccess error earliest=@w0
```

此搜索将返回从本周星期日上午 12:00 开始到当前时间的匹配的事件。当然，这意味着如果您在星期一中午运行此搜索，则只能看到 36 小时数据的事件。

示例 2：当前工作周（星期一到星期五）的 Web 访问错误。

```
eventtype=webaccess error earliest=@w1 latest=+7d@w6
```

此搜索将返回从本周星期一上午 12:00 开始到本周星期五下午 11:59 结束的匹配的事件。

如果您在星期一中午运行此搜索，则只能看到 12 小时数据的事件。然而，如果在星期五运行此搜索，您将看到从本周起始日到星期五当前时间的事件。但是，时间线将显示完整工作周。

示例 3：上一完整工作周的 Web 访问错误。

```
eventtype=webaccess error earliest=-7d@w1 latest=@w6
```

此搜索将返回从上个星期一上午 12:00 开始到上个星期五下午 11:59 结束的匹配的事件。

为实时搜索指定时间范围

搜索运行时设置历史搜索的时间范围。对于实时搜索，时间范围边界不断更新，默认情况下，结果从启动搜索时开始累计。您还可以指定一个用于表示时间滑动窗口的范围，例如，最后 30 秒。如果指定滑动窗口，Splunk 软件将采用该时间量来累计数据。例如，如果滑动窗口为 5 分钟，则直到第一个 5 分钟过去，您才能看到数据。您可以覆盖此行为，这样，Splunk 软件将于在标准实时搜索模式下运行之前，以历史数据回填初始窗口。请参阅“实时回填”。

实时调节器语法

要实时运行搜索，可以从时间范围的范围挑选器列表中的预定义实时时间范围窗口中进行选择，也可以使用**自定义时间...** 并选择**实时**指定自定义实时窗口。

实时搜索时间范围所遵循的语法与历史搜索相同，不同之处在于，您需要在相对时间说明符的前面加上 "rt"，即 rt<time_modifier>：`rt[+|-]<time_integer><time_unit>@<time_unit>`。请参阅“在搜索中指定时间调节器”。

这些值不适用于搜索字符串。如果您是 Splunk Enterprise 管理员，可以在编辑 times.conf 文件时使用这些值（用于向时间范围挑选器添加选项），或者在保存的搜索对话框中使用它们来指定最早/最晚时间范围，也可以在使用 REST API 访问 Splunk 搜索引擎时使用。

将时间范围与实时搜索一起使用时，在最后一秒钟发生的部分事件可能不会显示。这属于预期行为，原因是事件中的时间戳与事件到达时间存在一定的延迟。由于时间范围是相对于事件中的时间戳而言，而不是相对于事件的到达时间，因此，在该时窗后到达的事件将不会显示。

“所有时间”的实时搜索

在设置时窗（30 秒、1 分钟、2 小时）中发生的实时搜索与设置为“所有时间”的实时搜索之间存在着微小的差异。

- **在“窗口式”实时搜索中，**如果搜索中的事件不在窗口范围内，这些事件会消失，而对于时间早于搜索任务创建时间的那些事件，如果发生，便会显示在窗口中。
- **在“所有时间”实时搜索中，**窗口将涵盖所有事件，因此，一旦事件显示在窗口中便不会消失，不过，对于时间早于搜索任务创建时间的那些事件，如果发生，也会显示在窗口中。
- 相比之下，**历史搜索**事件永远也不会从您正在搜索的已设定时间范围内消失，且最新事件始终早于任务创建时间（但所含事件时间戳为将来日期的搜索除外）。

实时回填

对于实时窗口式搜索，可用历史数据回填初始窗口。它以单个搜索的形式运行，仅分为两个阶段：首先，搜索历史数据用以回填事件；然后，执行标准实时搜索。实时回填将确保过去时间段中，实际可视化和统计指标上使用数据填充的实时仪表板从一开始便是精确的。

可在 `limits.conf` 文件的 [realtime] 段落中启用实时回填：

```
[realtime]

default_backfill = <bool>
* Specifies if windowed real-time searches should backfill events
* Defaults to true
```

使用时间查找附近事件

Splunk Web 时间线和搜索的时间范围都基于事件**时间戳**。

搜索错误或解决问题时，查看大约在同一时间发生的事件可帮助关联结果并找出问题的根源。本主题介绍如何使用事件的时间戳和使用时间线搜索周围事件。

使用时间加速器

`_time` 字段表示事件的时间戳。运行搜索检索事件时，**时间**列下面将列出每个事件的时间戳。

可以单击事件的时间戳，并且打开包含控件的对话框，控件名称为 **_time accelerator**。使用 _time accelerator 运行可按时间检索与该事件相近事件的新搜索。

您可以搜索事件时间前后发生的所有事件。加速器为**该时间之前**、**该时间之后**以及**此时**。此外，您可以搜索附近的事件。例如，您可以搜索大于 30 秒、小于 1 小时、大于或小于 5 小时等等。

使用时间线

时间线是 Splunk 搜索在所选时间范围内返回的事件数量直方图。时间范围分为多个小的时间间隔（例如秒、分钟、小时或天），每个间隔的事件计数以列的形式显示。

在出现了与您的搜索匹配的事件时，时间线上的每列的位置对应于一个实例。如果在某个时间段没有列，则未找到事件。列越高，表示在该时间发生的事件越多。

事件数量中的高峰或沿时间线无事件，可表示您想要调查的时间段。

时间线拥有**钻取**功能，与表格和图表钻取类似。单击时间线中某列后，搜索结果将更新，以仅显示此列表示的事件。如果双击某列，将在此列所代表的时间范围内重新运行搜索。然后，您可以搜索此时间范围的所有周围事件。

子搜索

关于子搜索

使用子搜索

子搜索包含在主要搜索或外部搜索内。如果某搜索中包括子搜索，则先运行子搜索。

主要搜索中的子搜索必须用方括号括起来。

考虑以下搜索。

```
sourcetype=access_* status=200 action=purchase [search sourcetype=access_* status=200 action=purchase | top limit=1
clientip | table clientip] | stats count, dc(productId), values(productId) by clientip
```

搜索中用方括号括起的部分就是子搜索。

```
[search sourcetype=access_* status=200 action=purchase | top limit=1 clientip | table clientip]
```

子搜索中的第一个命令必须为**生成命令**，如 `search`、`eventcount` 或 `tstats`。若需生成命令的列表，请参阅*《搜索参考》*中的“命令类型”。

子搜索如何工作

子搜索可以查找之后添加到主要搜索中作为条件或参数的单条信息。因为您查找的单条信息是动态的，所以用到子搜索。每次运行子搜索时，单条信息都可能会变化。

例如，您想要返回前一小时内最活跃的主机的所有事件。最活跃的主机可能随时变化。在可以返回主机事件之前，您需要识别最活跃的主机。

将此搜索分为两部分。

- 前一小时内最活跃的主机。这是子搜索。
- 来自该主机的事件。这是主要搜索。

您可以运行两个搜索以获取事件列表。以下搜索可以识别前一小时内最活跃的主机。

```
sourcetype=syslog earliest=-1h | top limit=1 host | fields host
```

该搜索只会返回一个主机值。假设结果是名为 `crashy` 的主机。要返回主机 `crashy` 的所有事件，您需要运行第二个搜索。

```
sourcetype=syslog host=crashy
```

运行两个搜索的缺点在于您无法设置报表和仪表板面板以自动运行。您必须运行第一个搜索以识别所需信息，然后再用该信息运行第二个搜索。

您可以将这两个搜索结合成一个搜索（含子搜索）。

```
sourcetype=syslog [search sourcetype=syslog earliest=-1h | top limit=1 host | fields + host]
```

子搜索用方括号括起来并首先运行。本示例中的子搜索可以识别前一小时内最活跃的主机。然后将子搜索结果用作主要搜索的条件。主要搜索会返回主机事件。

时间范围和子搜索

您在搜索中直接指定的时间范围仅适用于主搜索。在主搜索中指定的时间范围不适用于子搜索。同样地，在子搜索中指定的时间范围仅适用于该子搜索。该时间范围不适用于主搜索或任何其他子搜索。

从时间范围挑选器中选择的时间范围会适用于主搜索或子搜索。

何时使用子搜索

子搜索主要用于两个目的：

- 使用另一个搜索的输出参数化一个搜索。以上描述的有关搜索前一小时内最活跃的主机的搜索是一个使用子搜索的示例。
- 运行单独的搜索，但使用 `append` 命令将输出添加至第一个搜索。

只有在正在尝试执行的显式操作涉及搜索，而不涉及数据转换的情况下，方可使用子搜索。例如，您不能将子搜索用于 "`sourcetype=top | multikv`"，因为 `multikv` 命令不能使用子搜索作为参数。特定命令（如 `append` 和 `join`）可以接受

子搜索作为参数。

搜索字符串中的多个子搜索

您可以在搜索中使用多个子搜索。

如果搜索有一组嵌套子搜索，首先运行最里面的子搜索，然后运行外层的子搜索，紧接着运行最外层的子搜索，最后运行主要搜索。

例如，您有以下搜索。

```
index=* OR index=_* | [search index=* | stats count by component | [search index=* | stats count by user | [search index=* | stats by ipaddress]]]
```

搜索顺序处理为：

1. `search index=* | stats by ipaddress`
2. `search index=* | stats count by user`
3. `search index=* | stats count by component`
4. （一个隐含的搜索命令）`index=* OR index=_*`（和嵌套式子搜索结果）

以下是另一个示例。

```
index=foo error [ search index=bar baz [search index=* | stats count by user | search count>100] | stats count by host ]
```

当您发现频繁使用子搜索时，一定要分析搜索语法。通常可能是重写搜索而忽略了子搜索。

如果搜索时按序排列而非嵌套式，则首选运行最左边的子搜索或搜索开头部分。然后逐步向右，或到搜索末尾，再运行下一个子搜索。运行完所有子搜索之后再运行主要搜索。

例如，您有以下搜索。

```
index=* OR index=_* | [search index=* | stats count by customerID] | [search index=* | stats by productName]
```

搜索顺序处理为：

1. `search index=* | stats by customerID`
2. `search index=* | stats count by productName`
3. （一个隐含的搜索命令）`index=* OR index=_*` （和子搜索结果）

子搜索示例

这些事例显示了用和不用子搜索搜索数据之间的不同。

> 这些示例将使用《搜索教程》中的示例数据，但应与任意格式的 Apache Web 访问日志结合使用。要在您自己的 Splunk 实例上尝试此示例，您必须下载样本数据，然后按照说明**将教程数据导入 Splunk**。运行搜索时使用时间范围**所有时间**。

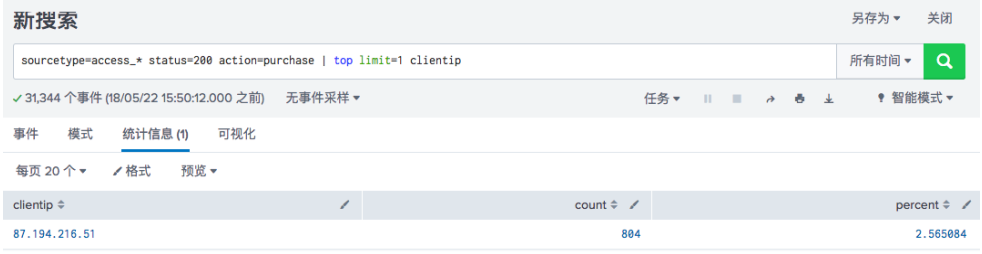
示例 1：在没有子搜索的情况下，找出最常光顾的顾客购买什么

您想要找找看 Buttercup Games 网上商店光顾频率最高的一位顾客，以及该顾客的购买情况。使用 `top` 命令返回最常光顾顾客。

1. 要查找访问在线商店次数最多的顾客，使用此搜索。

   ```
   sourcetype=access_* status=200 action=purchase | top limit=1 clientip
   ```

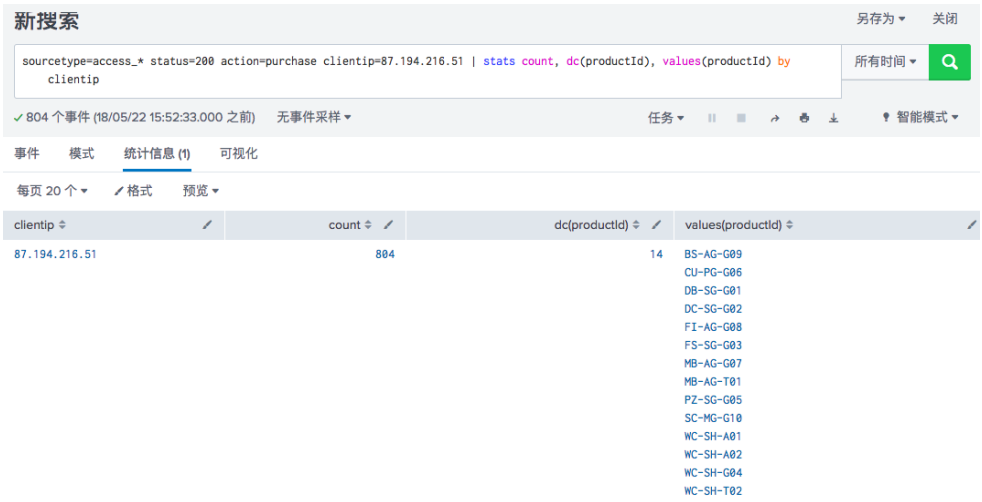
   `limit=1` 参数指定返回 1 个值。`clientip` 参数指定要返回的字段

   该搜索会返回一个 `clientip` 值 87.194.216.51，您将使用此值识别 VIP 客户。

2. 现在，您需要运行另一个搜索，以确定此 VIP 客户购买了多少种不同产品。使用 `stats` 命令统计此 VIP 客户购买的数量。

```
sourcetype=access_* status=200 action=purchase clientip=87.194.216.51 | stats count, dc(productId), values(productId) by clientip
```



   此搜索使用了 `count()` 函数，以返回 VIP 顾客的购买总数。`dc()` 函数即 distinct_count 函数。使用此函数可统计顾客购买的不同或每种产品的数量。`values` 函数用于将不同的产品 ID 显示为多值字段。

此方法的缺点在于每次要构建此表格时您就必须运行两个搜索。买得最多的顾客不可能在任何给定时间范围内都是同一人。

示例 2：使用子搜索，找出最常光顾的顾客购买什么

让我们从第一个要求开始，识别单个 Buttercup Games 在线商店中最常光顾的顾客。

1. 将下列搜索复制粘贴到搜索栏并运行此搜索。确保时间范围设为**所有时间**。

   ```
   sourcetype=access_* status=200 action=purchase | top limit=1 clientip | table clientip
   ```

   此搜索会返回最常光顾的顾客的 clientip `clientip=87.194.216.51`。这里的搜索基本上与步骤 1 中的示例 1 相同。不同在于最后的管道命令 `| table clientip`，该命令在表中显示 clientip 信息。

   要知道顾客购买了什么东西，使用相同数据运行搜索。您可提供最常光顾的顾客搜索结果作为购买搜索条件之一。

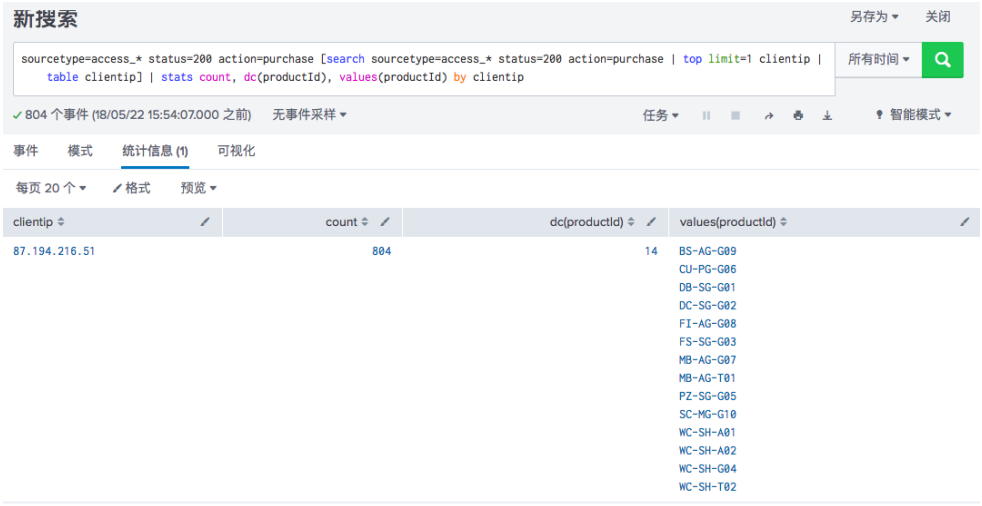
   最常光顾的顾客搜索会成为购买搜索的**子搜索**。购买搜索指**外部**搜索或主要搜索。由于您在搜索相同的数据，外部搜索的开始和子搜索的开始相同。

   子搜索用方括号 [ ] 括起，分析搜索时将首先处理子搜索。

2. 将下列搜索复制粘贴到搜索栏并运行此搜索。

   ```
   sourcetype=access_* status=200 action=purchase [search sourcetype=access_* status=200 action=purchase | top limit=1 clientip | table clientip] | stats count, dc(productId), values(productId) by clientip
   ```

   由于 `top` 命令还返回 **count** 和 **percent** 字段，因此 `table` 命令用于只保留 `clientip` 值。



这些结果应当与示例 1 中的两个搜索结果匹配，前提是您在同一时间范围内运行此搜索。如果您更改了时间范围，则可能会看到不同的结果，因为买得最多的顾客将有所不同。

此子搜索的性能取决于非重复 IP 地址匹配 `status=200 action=purchase` 的数量。如果有成千上万的非重复 IP 地址，则 `top` 命令必须在 top 1 返回之前追踪所有地址，这将影响性能。默认情况下，子搜索最多返回 10,000 个结果，运行时间最多 60 秒。在大型生产环境中，本例中的子搜索可能在完成之前已超时。最佳选择是重写查询，以限制子搜索必须处理的事件数量。或者，您可以增加最大结果数和最大运行时间参数。

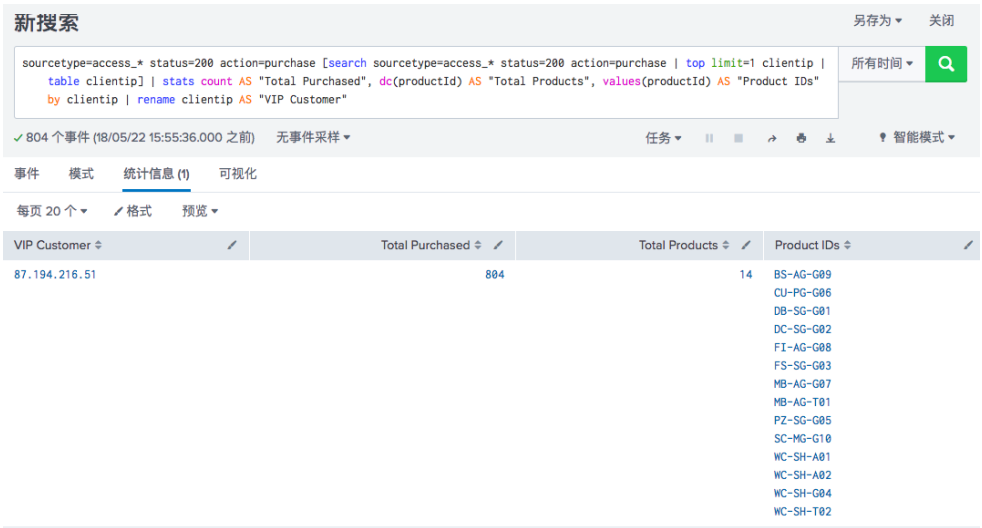
您可以通过对列进行重命名，使信息更容易理解。

柱形图	Rename
count	购买总数
dc(productId)	总产品
values(productId)	产品 ID
clientip	VIP 客户

通过对搜索中的字段使用 AS 操作符对列进行重命名。如果您想要使用的重命名包含空格，则必须用引号将新名括起。

3. 要对字段进行重命名，将下列搜索复制粘贴到搜索栏并运行此搜索。

```
sourcetype=access_* status=200 action=purchase [search sourcetype=access_* status=200 action=purchase | top limit=1 clientip | table clientip] | stats count AS "Total Purchased", dc(productId) AS "Total Products", values(productId) AS "Product IDs" by clientip | rename clientip AS "VIP Customer"
```



子搜索性能注意事项

如果搜索需要返回大量结果，则可能会导致子搜索性能大幅降低。

思考以下搜索。

```
sourcetype=access_* status=200 action=purchase [search sourcetype=access_* status=200 action=purchase | top limit=1 clientip | table clientip] | stats count, dc(productId), values(productId) by clientip
```

此子搜索的性能取决于非重复 IP 地址匹配 `status=200 action=purchase` 的数量。如果有成千上万的非重复 IP 地址，则 `top` 命令必须在影响性能的 top 1 返回之前追踪所有地址。

另外，子搜索最多返回 10,000 个结果，运行时间最多 60 秒。在大型生产环境中，本例中的子搜索很可能在完成之前已经超时。

您可使用几种方式来控制结果：

- 试着重写查询，以限制子搜索必须处理的事件数量。
- 您可以更改 format 命令在搜索内内联操作的结果数，方法是 `format` 命令附加到子搜索的末尾。

  ```
  ...| format maxresults = <integer>
  ```

  有关更多信息，请参阅*《搜索参考》*中的 `format` 命令。

如果您使用的是 Splunk Enterprise，也可以通过编辑 `limits.conf` 文件中的设置来控制子搜索。请参阅"如何编辑配置文件"。编辑运行时间和返回结果最大数的设置。

[subsearch]
maxout = <integer>

- 要从子搜索中返回的最大结果数量。
- 此数值不能大于或等于 10500。
- 默认为 10000。

maxtime = <integer>

- 子搜索完成之前的最大运行秒数。
- 默认为 60。

ttl = <integer>

- 缓存给定子搜索的结果的时间，以秒为单位。
- 请勿将此值设置为 120 秒以下。
- 默认为 300。

运行搜索后，可单击**任务**菜单并选择*检查任务*，打开"搜索任务查看器"。向下滚动至 `remoteSearch` 组件，您可以看到从子搜索中返回的实际查询。有关更多信息，请参阅本手册中的"查看搜索任务属性"。

子搜索命令的输出设置

默认情况下，子搜索返回的最大结果数为 10,000。您会发现实际的输出结果数量有所变化，因为每个命令都可以在调用子搜索时更改 `maxout` 的默认值。另外，默认值适用于要扩展到搜索表达式的子搜索，但是某些命令（如 join、append 和 appendcols）并非如此。

- 例如，如果指定了 `maxresultrows` 参数，则 `append` 命令可以覆盖默认最大值，除非您将 `maxout` 指定为 `append` 命令的参数。
- `join` 命令的输出限制由 `limits.conf` 文件中的 `subsearch_maxout` 控制。此输出限制默认为 50,000 个事件。

问答

有什么问题吗？请访问 Splunk Answers，查看 Splunk 社区有哪些与使用子搜索相关的问题和答案。

使用子搜索关联事件

子搜索可获取一个搜索的结果并将其用于另一个搜索，从而启用类似连续状态的数据分析。您可以使用子搜索关联数据并在整个事件集中评估事件，包括不同索引中的数据或分布式环境中的 Splunk Enterprise 服务器。

例如，您可能有两个或更多个索引用于不同的应用程序日志。这些日志中的事件数据可能共享至少一个通用字段。您可以使用该字段的值在一个索引中基于不在另一个索引中的某个值来搜索事件：

```
sourcetype=some_sourcetype NOT [search sourcetype=another_sourcetype | fields field_val]
```

注意：这相当于 SQL "NOT IN" 功能：

SELECT * from some_table

```
WHERE field_value
NOT IN (SELECT field_value FROM another_table)
```

更改子搜索结果的格式

当使用子搜索时，`format` 命令会隐式应用于子搜索结果。`format` 命令会将子搜索结果更改为单个线性搜索字符串。这用于要将返回字段中的值传入主要搜索的情况。

如果子搜索返回了一个表格，例如：

```
           | field1  | field2  |
            -------------------
event/row1 | val1_1  | val1_2  |
event/row2 | val2_1  | val2_2  |
```

`format` 命令返回：

```
(field1=val1_1 AND field2=val1_2) OR (field1=val2_1 AND field2=val2_2)
```

有关更多信息，请参阅“格式命令”。

格式特例

`format` 命令执行的格式有几个特例。

- 所有内部字段（以前导下划线 (_) 字符开头的字段）将被忽略，并且不再设置为线性搜索字符串格式。
- 如果字段名称是 `search` 或 `query`，字段值将直接在重新设置格式的搜索字符串中呈现。

搜索和查询字段

您可以将字段重命名为 `search` 或 `query` 以更改子搜索结果格式。将字段重命名为 `search` 或 `query` 是一种特殊使用情况。将字段重命名为其他名称后，子搜索将返回您指定的新字段名称。

使用搜索字段名称

当需要附加一些静态数据或对子搜索中的数据进行评估时，请使用 `search` 字段名称和 `format` 命令。然后，您可以将数据传递到主要搜索。例如，您可以将搜索结果中的第二个字段重命名为 `search`，如下表所示：

```
           | field1  | search  |
            -------------------
event/row1 | val1_1  | val1_2  |
event/row2 | val2_1  | val2_2  |
```

然后使用 `format` 命令返回：

```
(field1=val1_1 AND val1_2) OR (field1=val2_1 AND val2_2)
```

而不是

```
(field1=val1_1 AND field2=val1_2) OR (field1=val2_1 AND field2=val2_2)
```

对于多值字段，当您使用 `search` 字段名称时，此字段的第一个值将用作实际的搜索术语。

使用查询字段名称

当您想从子搜索中返回字段而非字段名称时，请使用 `query` 字段名称。

`query` 字段名称和使用 `format` 命令类似。而不是将字段和值对传递到主要搜索，如：

```
(field1=val1_1 AND field2=val1_2) OR (field1=val2_1 AND field2=val2_2)
```

使用 `query` 字段名称只传递值：

```
(val1_1 AND val1_2) OR (val2_1 AND val2_2)
```

示例

以下搜索将查询和名称标记或字段值关联的 `clID` 字段中的值。然后，使用 clID 值来搜索多个数据来源。

```
index=myindex [search index=myindex host=myhost MyName | top limit=1 clID | fields clID ]
```

子搜索返回字段和值的格式为：`( (clID="0050834ja") )`

要仅返回值 `0050834ja`，在子搜索中将 `clID` 字段重命名为 `search`：例如：

```
index=myindex [search index=myindex host=myhost MyName | top limit=1 clID | fields clID | rename clID as search ]
```

如果字段是已命名的 `search` 或 `query`，则字段名称将被丢弃，并且子搜索末尾的隐式 `| format` 命令将仅返回值。

如果您返回多个值，如指定 `...| top limit=3`，则子搜索将使用值之间的布尔值或运算符返回每个值。例如，如果之前的搜索示例使用 `...| top limit=3`，则从子搜索返回的值为 `( ( value1 ) OR ( value2 ) OR ( value3 ) )`。

创建统计性表格和图表可视化

有关转换命令和搜索

要创建图表可视化，您的搜索必须将事件数据转换为统计数据表。图表和其他类型数据可视化需要这些统计数据表。本节介绍如何使用**转换命令**来转换事件数据。

本节介绍转换命令的主要类别，并提供关于如何在搜索中使用这些命令的示例。

转换命令

主要转换命令有：

- `chart`：创建图表，可显示您想要绘制的任何**系列**的数据。图表的 X 轴可决定要跟踪的字段。
- `timechart`：用于创建"时段趋势"报表，这意味着 `_time` 始终为 X 轴。
- `top`：生成用于显示字段最常用值的图表。
- `rare`：创建用于显示字段最不常用值的图表。
- `stats`：生成用于显示摘要统计信息的报表。

要了解更多信息，请参阅*《搜索参考》*中的"转换命令"。

注意：如您在以下示例中所见，您始终将转换命令放在搜索命令之后，中间用**管道符**（|）连接。

`chart`、`timechart` 和 `stats` 命令全部设计用于与统计函数结合使用。可用统计函数包括：

- 计数、非重复计数
- 平均值、中值、模式
- 最小值、最大值、范围、百分比
- 标准偏差、方差
- 总和
- 最早出现、最晚出现

有关统计参数的更多信息，请参阅*《搜索参考》*中的"统计和图表函数"。部分统计函数仅适用于 `timechart` 命令。

注意：所有使用转换命令的搜索都将生成特定数据结构。不同图表类型要求按照特殊方式设置这些数据结构。例如，并非所有可用于生成条形图、柱形图、折线图和面积图的搜索也可用于生成饼图。要了解更多信息，请阅读*《仪表板和可视化》*手册中的"可视化的数据结构要求"。

表格、图表和报表示例

以下示例使用转换命令创建表格、图表和报表：

- 创建基于时间的图表
- 创建不（一定）基于时间的图表
- 创建用于显示摘要统计信息的报表
- 构建多数据系列图表

实时报表

您可使用实时搜索来实时计算大量传入数据流的相关指标，而不必使用摘要索引。不过，由于您要基于实时连续数据流生成报表，因此，当事件流入时，时间线将更新，且您只能在预览模式下查看表格或图表。另外，某些搜索命令将更适用于实时状态下使用（例如，streamstats 和 rtorder）。请参阅"关于实时搜索和报表"。

另请参阅

命令类型
搜索类型

创建基于时间的图表

本主题介绍使用 timechart 命令创建基于时间的报表。

timechart 命令

timechart 命令可生成摘要统计信息表。该表随后可设置为图表可视化格式，在该可视化中，数据绘制于始终为时间字段的 X 轴上。使用 timechart 命令显示随时间变化的统计趋势。您还可以选择按另一字段将数据拆分为图表中的单独系列。Timechart 可视化通常为折线图、面积图或柱形图。

如果您使用了 `timechart` 命令，则 X 轴表示时间。Y 轴可以是任何其他的字段值、计数值，或字段值的统计计算。

有关更多信息，请参阅*《仪表板和可视化》*手册中的“可视化的数据结构要求”。

示例

示例 1：此报表使用内部 Splunk 日志数据来直观显示 Splunk 随时间处理的平均索引吞吐量（索引 kbps）。此信息由处理器分隔或拆分：

```
index=_internal "group=thruput" | timechart avg(instantaneous_eps) by processor
```

另请参阅

构建多数据系列图表

创建不（一定）基于时间的图表

本主题介绍使用**转换命令** chart 创建非基于时间的报表。

chart 命令

chart 命令以支持数据系列可视化的数据结构，即图表（例如柱形图、折线图、面积图和饼图）的形式返回结果。

与 `timechart` 命令不同（此命令使用 `_time` **默认字段**作为 X 轴），用 `chart` 命令创建的图表使用任意字段作为 X 轴。借助 chart 命令，可使用 `over` 关键字来确定将作为 X 轴的字段。

示例

示例 1：使用 Web 访问数据向您显示独特访客在每个工作日的平均计数。

```
sourcetype=access_* | chart avg(clientip) over date_wday
```

您可以选择按另一个字段拆分数据，这意味着“拆分依据”字段的每个非重复值都是图表中的一个单独系列。如果您的搜索包括 "split by" 子句，请将 `over` 子句放在 "split by" 子句之前。

以下报表将生成一个图表，其中显示每个 `clientip` 在给定时间范围内处理的 KB 总和，按 `host` 拆分。完成后的图表会显示 Y 轴为 `bytes` 值，而 X 轴为 `clientip`。延迟值由主机分隔。运行此搜索后，将报表的格式设置为堆叠条形图。

```
sourcetype=access_* | chart sum(bytes) over clientip by host
```

示例 2：创建一个堆叠条形图，用于拆分送至服务器的 http 和 https 请求。

为此，首先创建 `ssl_type`，即搜索时间**字段提取**，其中包含入站端口号或传入 URL 请求（假定已记录这些内容）。完成后的搜索如下所示：

```
sourcetype=access_* | chart count over ssl_type
```

运行搜索后，将结果的格式设置为堆叠条形图。

直观显示高低字段值

本主题介绍如何使用**转换命令** top 和 rare 创建显示最常用和最不常用值的图表。

top 和 rare 命令

top 命令在返回事件中返回指定字段的最常用值。rare 命令在返回事件中返回指定字段的最不常用值。两个命令的语法相同。如果不指定限制，`top` 或 `rare` 中显示的默认值数量为 10。

示例

示例 1：生成的报表将对防火墙信息分类，列出系统所使用的前 100 个目标端口：

```
sourcetype=firewall | top limit=100 dst_port
```

示例 2：生成显示拒绝数量最少的数据来源端口的报表。

```
sourcetype=firewall action=Deny | rare src_port
```

一个更为复杂的 top 命令示例

假定您要为监视系统中的告警日志建立索引，并且您有两个字段：

- msg 为消息，例如 CPU at 100%。
- mc_host 为生成消息的主机，如 log01。

如何才能获得显示 top msg 及发送这些消息的 mc_host 值的报表，并能产生一个类似如下的表格：

mc_host 发送的消息
CPU 使用率 100%
log01
log02
log03
日志文件告警
host02
host56
host11

要实现这一点，设置一个搜索，查找 top message（每个 mc_host 的，使用 limit=1 仅返回一条），然后 sort（以降序按消息 count 排序）：

```
sourcetype=alert_log | top 1 msg by mc_host | sort count
```

创建用于显示摘要统计信息的报表

本主题介绍使用 stats 和 eventstats **转换命令**创建用于显示某字段相关摘要统计信息的报表。

stats 和 eventstats 命令

eventstats 命令的工作方式与 stats 命令的完全相同，只是每个事件中嵌入了命令的聚合结果，且仅限于与每个事件有关的聚合。

使用 split-by 子句

为充分利用 stats 命令，您需要使用一个 "split by" 子句。例如，以下报表不会提供大量信息：

```
sourcetype=access_combined | stats avg(kbps)
```

它为您提供所有事件 kbps 的平均值（sourcetype 为 access_combined），即单个值。生成的柱形图仅包含一列。

但如果用“拆分依据”字段分隔该报表，Splunk 软件将生成按该字段细分统计信息的报表。以下报表将生成一个柱形图，它对 access_combined 日志排序，以获得按主机区分的平均吞吐量 (kbps)：

```
sourcetype=access_combined | stats avg(kbps) by host
```

示例

示例 1：创建报表，使其按降序显示 Splunk 进程的 CPU 使用率：

```
index=_internal "group=pipeline" | stats sum(cpu_seconds) by processor | sort sum(cpu_seconds) desc
```

示例 2：创建报表，显示由主机分隔、sourcetype 为 access_combined 的所有事件的平均 kbps。

为 eventstats 结果指定字段名称，具体方式为添加 as 参数。因此，上文第一个示例可使用 "avgkbps" 作为包含 eventstats avg(kbps) 运算结果的新字段的名称来重述：

```
sourcetype=access_combined | eventstats avg(kbps) as avgkbps by host
```

运行此组命令时，Splunk 软件将一个新字段 avgkbps 添加到每个 sourcetype=access_combined 事件（包含 kbps 字段的事件）。avgkbps 的值是该事件的平均 kbps。

在搜索结果中查找关联、统计相关性和差异

本主题介绍使用**转换命令**查找搜索结果中各字段值之间的关联、相似度和差异。

associate 命令

associate 命令确定通过字段/字段值对与彼此相关联的事件。例如，如果一个事件的 referer_domain 为

"http://www.google.com/"，另一事件的 `referer_domain` 为同一 URL 值，则这两个事件彼此关联。

使用 *supcnt*、*supfreq* 和 *improv* 参数"调节"通过 `associate` 命令获得的结果。有关这些参数的更多信息，请参阅相关命令参考主题。

示例：搜索 Web access sourcetype，并确定至少共享三个字段/字段值对关联的事件：

```
sourcetype=access* | associate supcnt=3
```

correlate 命令

correlate 命令计算字段之间的统计相关性。它使用 `cocur` 运算计算两个字段存在于同一组结果中时间的百分比。

*示例：*搜索所有 `eventtype=goodaccess` 的事件，并计算所有那些字段之间的同现关联。

```
eventtype=goodaccess | correlate type=cocur
```

diff 命令

使用 diff 命令比较两个搜索结果之间的差异。默认情况下，它将比较所选搜索结果的原始文本，除非您使用 *attribute* 参数集中在特定字段属性上。

示例：比较搜索中返回的第 44 个和第 45 个事件的 IP 地址。

```
eventtype=goodaccess | diff pos1=44 pos2=45 attribute=ip
```

构建多数据系列图表

Splunk **转换命令**不支持直接在图表（或时间表）中定义多个数据**系列**。不过，通过使用 `stats` 和 `xyseries` 命令组合，仍可实现此目的。

`chart` 和 `timechart` 命令均返回用于绘图的列表数据，其中 X 轴分别为某个任意字段或 `_time`。当这些命令与拆分依据字段一起使用时，输出为一个表格，其中每一列代表该拆分依据字段的一个非重复值。

与此相反，`stats` 命令生成的表格中，每一行代表分组依据字段值的单个唯一组合。然后，您可使用 `xyseries` 命令重新定义用于绘图的数据系列。

大多数情况下，您可使用 "...| stats n by x,y | xyseries x y n" 模拟 "...| chart n by x,y" 的结果。（对于结果的 `timechart` 当量，x = `_time`。）

方案

假定您想要基于应用程序服务器群集中的数据生成报表。从每台服务器收集的事件包含诸如活动会话计数、来自上次更新处理的请求等信息，并放置在 `applications_servers` 索引中。您想要在同一张时间表上显示每个服务器实例及每个实例的会话数，以便能够比较会话和负载的分布情况。

理想情况下，您需要能够运行时间表报表，如：

```
index=application_servers | timechart sum(handledRequests) avg(sessions) by source
```

但时间表不支持多个数据系列；因此，您需要改为运行以下类似搜索：

```
index=application_servers | bin _time | stats sum(handledRequests) as hRs, avg(sessions) as ssns by _time,source |
eval s1="handledReqs sessions" | makemv s1 | mvexpand s1 | eval yval=case(s1=="handledReqs",hRs,s1=="sessions",ssns)
| eval series=source+":"+s1 | xyseries _time,series,yval
```

走查

```
... | bin _time
```

在使用 `stats` 命令之前，您要做的第一件事情是按时间分隔事件。

```
... | stats sum(handledRequests) as hRs, avg(sessions) as ssns by _time,source
```

`stats` 命令用于计算每个数据来源值的统计信息：`handledRequests` 的值总和重命名为 `hRs`；`sessions` 的平均数重命名为 `ssns`。

```
... | eval s1="handledRequests sessions" | makemv s1 | mvexpand s1
```

它使用 `eval` 命令将单值字段 "s1" 添加到从 stats 命令获得的每个结果。然后，makemv 命令将 s1 转换为一个多值字段，其中第一个值为 "handledRequests"，第二个值为 "sessions"。接下来，mvexpand 为 s1 的每个值创建单独系列。

```
... | eval yval=case(s1=="handledRequests",hRs,s1=="sessions",ssns)
```

它使用 eval 命令定义一个新字段 yval，然后基于它的匹配情况为其分配值。因此，如果 s1 的值为 "handledRequests"，yval 会被分配的值为 "hRs"。如果 s1 的值为 "sessions"，yval 会被分配的值为 "ssns"。

```
... | eval series=source+":"+s1
```

它使用 eval 命令定义一个新字段 series，用于连接主机与 s1 字段的值。

```
... | xyseries _time,series,yval
```

最后，xyseries 命令使用 _time 作为 X 轴，yval 作为 Y 轴来定义一个图表，且数据由 series 定义。

比较多天中每小时的总和

`timechart command` 生成的图表可以显示随时间而变化的趋势。它对于所能执行的操作有严格的限制。有时您应该使用 `chart command` 命令，该命令可提供更多灵活性。

此示例说明如何使用 `chart` 来比较几天中收集的值。此方法不适用于 `timechart`

方案

这两种搜索基本相同。它们均显示在 24 小时期间内 `P` 字段的每小时总和。唯一区别是其中一个搜索会涵盖过去 10 天的这段期间，而另一个搜索会涵盖过去 9 天的这段期间：

搜索 1：

```
earliest=-10d latest=-9d | timechart span="1h" sum(P)
```

搜索 2：

```
earliest=-9d latest=-8d | timechart span="1h" sum(P)
```

创建一个柱形图，其中合并这两个搜索的结果，因此，您能够看到 10 天前下午 3 点的 `P` 总和以及 9 天前下午 3 点的 `P` 总和。

解决方案

使用 chart 命令设置包括这两天的搜索。然后为搜索结果中所找到的每个不同 `date_hour` 和 `date_wday` 组合创建一个“P 的总和”列。

完成后的搜索如下所示：

```
earliest=-10d latest=-8d | chart sum(P) by date_hour date_wday
```

这会生成一个含有 24 个时隙的图表，每个时隙代表一天当中的一个小时。每个时隙包含两列，使您能够比较报表的时间范围所涵盖的两天间每小时的总和。

有关报表搜索入门及构建方式，请参阅*《搜索手册》*中的“使用报表命令”。

有关 `chart>` 和 `timechart` 函数的更多信息，请参阅*《搜索参考》*中的“统计和图表函数”。

钻取表格和图表

运行搜索后，您可以运行不同类型的辅助**钻取**搜索。钻取搜索选项取决于您单击的元素类型。

在**统计**选项卡中，您可以在单击字段值或计算搜索结果后运行钻取搜索。

关于钻取字段-值对的信息，请参阅“钻取事件详细信息”。

您还可以单击图表和可视化元素以运行钻取搜索。关于更多信息，请参阅*《仪表板和可视化》*中的“使用钻取保障仪表板的交互性”。

在数据透视表中打开非转换搜索以创建表格和图表

不含**转换命令**的搜索将返回事件列表，可在**事件**选项卡中查看；您也可以使用**模式**选项卡查看这些事件中的主导模式。但是，这些非转换搜索不会返回统计表格形式的结果。没有统计表格的情况下，Splunk 软件无法创建图表或其他可视化。这意味着运行非转换搜索后，**统计**或**可视化**选项卡中不会显示结果。

如果运行非转换搜索并想基于此搜索创建表格或图表，转到**统计**或**可视化**选项卡，并在数据透视表中打开搜索。

在**数据透视表编辑器**中，无需编辑原始搜索即可构建表格和图表。当使用数据透视表构建器优化可视化时，会按要求重新运行基本搜索，以便您可以查看所做更改会带来的影响。

当您将在数据透视表构建器中创建的可视化另存为报表或仪表板面板时，系统会创建相应的**数据模型**。此数据模型是保存的报表或仪表板面板的基础。数据模型定义报表或仪表板面板中涉及的基本搜索和字段。没有数据模型，您将无法重新运行报表或查看保存的面板。

在数据透视表中打开搜索

1. 在搜索视图中，运行非转换搜索。例如：

   ```
   sourcetype=access_* status=200 action=purchase
   ```

2. 转到**统计**或**可视化**选项卡并单击**数据透视表**。
3. 选择您想用于在数据透视表构建器中构建数据透视表表格或图表的字段组。

   每个选项显示它所代表的字段数量（用括号表示）：

   所有字段提供搜索发现的所有字段。

   选择的字段仅提供针对**字段**选项卡标识为**选择的字段**的字段。如果在数据透视表中打开搜索，未进行更改或未选择字段，**选择的字段**选项将提供默认的选择字段：`host`、`source` 和 `sourcetype`。要使用一组不同字段构建数据透视表表格或图表，可先转到**字段**选项卡，在“选择的字段”列表中选择字段，然后再转到**统计**或**可视化**选项卡，在数据透视表中打开搜索。

   借助**字段至少具有**，您可以为字段集设置覆盖阈值。例如，要处理适用于大多数事件的字段，可设置较高阈值，例如 70%。数据透视表中得到的字段集仅包含搜索返回的 70%（或更多）的事件中存在的字段。

4. 单击**确定**以转到**数据透视表编辑器**。
5. 构建数据透视表表格或图表。

   数据透视表元素类型中的**属性**列表（过滤器、拆分行、拆分列以及列值）包含您在步骤 3 中选择的字段集。

   注意：如果未保存表格或图表就转到“数据透视表编辑器”之外的地方，所做的工作将丢失。

   要保存所进行的工作，请参阅下一子主题“保存完成的数据透视表表格或图表”。

在数据透视表编辑器中时，可单击**在搜索中打开**，以在搜索界面中打开数据透视表搜索并对该搜索进行编辑。此操作会将您带出“数据透视表编辑器”，并阻止您保存您所创建的任何数据透视表表格或图表（参阅下一子主题）。有关使用“数据透视表编辑器”设计表格的详细信息，请参阅“使用数据透视表编辑器设计数据透视表表格”。有关使用“数据透视表编辑器”设计图表和其他可视化的详细信息，请参阅“使用数据透视表编辑器设计数据透视表图表和可视化”。

保存完成的数据透视表表格或图表

可以将数据透视表编辑器中的表格或图表另存为报表或仪表板面板。但是，Splunk 软件还必须创建数据模型，以支持保存的报表或面板。保存数据模型后，此模型需要访问报表或面板。

1. 在“数据透视表编辑器”中，单击**另存为**并选择*报表*或*仪表板面板*。

   根据所选择的内容，将显示“另存为报表”或“另存为仪表板面板”对话框。

2. 在**另存为**对话框中，显示您所保存报表或仪表板面板相关信息。

   有关这些字段的详细信息，请参阅保存报表或保存仪表板面板相关文档。

3. 在**另存为**对话框中，为将支持报表或仪表板面板的数据模型键入**模型名称**和**模型 ID**。

   如果您的角色拥有管理员级别的能力，可通过此流程管理 Splunk Enterprise 创建的模型。

4. 单击**保存**可保存报表或仪表板面板，并创建数据模型。

   单击按钮查看新报表或仪表板面板，或通过单击模型名称转到新数据模型。

   单击数据模型名称可转到数据模型构建器。在数据模型构建器中，可以更改与模型相关的字段，以及向模型添加数据模型数据集。

关于通过此方法创建的数据集的权限

新创建的数据模型为专用，仅供其创建者查看和使用。仅具有管理员或高级用户角色的用户（或拥有同等权限的角色）才能够共享数据模型。如果未共享数据模型，则也无法共享利用此数据模型创建的报表或仪表板面板。此外，数据模型只有在共享后才能加速。

如果创建的报表或仪表板面板仅供自己使用，则无需进行其他操作。如果希望其他用户能够访问报表或仪表板面板且您拥有相应权限，请共享相关数据模型；或请拥有管理员级别权限的用户帮您将其共享。

有关数据模型权限的更多信息，请参阅*《知识管理器手册》*中的“管理数据模型”。

实时搜索和报表

关于实时搜索和报表

使用**实时搜索**和报表，您可以在事件**建立索引**之前搜索事件以及在事件流入时预览报表。

- 您可以基于在后台持续运行的实时搜索设计**告警**。与基于**计划报表**的告警相比，此类**实时告警**在提供通知时将更为及时。有关更多信息，请参阅*《告警手册》*。
- 您也可以在**仪表板**中显示实时搜索结果和报表。关于更多信息，请参阅*《仪表板和可视化》*中的“仪表板概览”。

并发的实时搜索的数量可以显著地影响索引性能。要减少对索引器的影响，可以启用索引实时搜索。

注意：默认情况下，只有具有“管理员”角色的用户才能运行和保存实时搜索。有关管理角色及向用户分配角色的更多信息，请参阅*确保 Splunk Enterprise 安全*中的“添加和编辑角色”。

实时搜索机制

实时搜索会在事件到达建立索引时即对他们进行扫描。启动实时搜索之后，Splunk 软件会扫描传入事件。扫描会查找包含索引时间字段的事件，包含该字段表明事件*可*与搜索匹配。

当实时搜索运行时，软件会根据您的搜索条件定期评估已扫描事件，以在为搜索定义的滑动**时间范围窗口**找到实际匹配。当搜索以更快或更慢的速率发现匹配的事件时，匹配的事件的数量会随时间上下波动。如果您要在 Splunk Web 中运行搜索，搜索时间线还将显示搜索在所选时间范围内返回的匹配的事件。

下面是实时搜索的一个示例，时间范围为 1 分钟。在获取以下屏幕截图时，搜索自启动以来已扫描总计 501 个事件。匹配的事件计数 333 代表在过去一分钟内确定与搜索条件匹配的事件数。下一分钟，此数字将在 312 和 357 之间波动。如果该数字显著上升或下降，可能表示发生了需要您仔细查看的事情。

如您所见，最新事件位于时间线右侧。事件会随着时间向左移动，直到离开左侧，从时间范围窗口中完全消失。

实时搜索应继续运行，除非您或其他用户停止搜索或删除搜索任务。实时搜索不得因任何其他原因而“超时”。如果您的事件停止，可能是出现了与性能相关的问题（请参阅“预期性能和已知限制”）。

实时搜索可利用所有搜索功能，其中包括如查找、交易等高级功能。还有与实时搜索结合使用的特定搜索命令，如 `streamstats` 和 `rtorder`。

索引实时搜索

建立事件索引后启用要运行的实时搜索可极大提高索引性能。有大量并发实时搜索时，这一点尤其如此。要减少对索引器的影响，可以启用索引实时搜索。它将运行诸如**历史搜索**等搜索，但还将随着磁盘上出现新事件，不断对其进行更新。

不需要实时到秒的准确度时，使用索引实时搜索。

前提条件

- 只有具有文件系统访问权限的用户，如系统管理员才能启用索引实时搜索。
- 请参阅*《管理员手册》*中的"如何编辑配置文件"了解具体步骤。

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。在本地目录进行更改。

步骤

1. 打开搜索应用的本地 `limits.conf` 文件。例如，`$SPLUNK_HOME/etc/apps/<app_name>/local`。
2. 在 `[realtime]` 段落中，将 `indexed_realtime_use_by_default` 设为 `true`。

如果您使用的是 Splunk Cloud 并想修改索引实时搜索的默认值，请打开支持票证。

关于同步延迟时间

*索引*实时搜索返回的结果将始终落后于实时搜索。构建在索引实时搜索中的是同步延迟。同步延迟是一种预防措施，这样不会缺失数据。

索引数据不一定以索引数据时的顺序显示在磁盘上，因为：

- 使用多个线程同时索引。
- 同步延迟对您操作系统上的索引数据进行排序

实时索引必须记住为时间范围窗口的当前迭代返回的最新索引事件。该事件被用作时间范围窗口下一次迭代的起点。如果不强制执行同步延迟，则最新事件之前的某些事件可能无法搜索。时间范围窗口迭代期间不会返回这些事件，且永远不会返回。随着索引和系统上传的增加，未返回的事件的可能性也会增加。

您可通过 `indexed_realtime_disk_sync_delay = <int>` 设置控制同步延迟落后时间的秒数。默认情况下，延迟设为 60 秒。

默认的 60 秒相当保守。对于大部分系统来说，30 秒的延迟可能会运行成功。对于您的系统和使用情况而言，如果可以接受索引实时搜索缺失一些事件，你可以将设置非常低或 0 同步延迟。但是，这样如果有事件缺失，您可能无法收到消息，除了和所有事件匹配的搜索。

其他索引实时设置

以下是您可用于配置索引实时搜索行为的其他设置，包括：

- indexed_realtime_default_span
- indexed_realtime_maximum_span
- indexed_realtime_cluster_update_interval

这些设置在 limits.conf.spec 文件中有介绍。

Splunk Web 中的实时搜索和报表

Splunk Web 中的实时搜索

运行**实时搜索**的方式与运行**历史搜索**完全相同。不过，由于您要搜索实时且连续的数据流，因此，当事件流入时，时间线将更新，且您只能在预览模式下查看报表。另外，相对于历史搜索，某些搜索命令将更适用于实时搜索。例如，streamstats 和 rtorder 均设计用于实时搜索。

要在 Splunk Web 中启动一个实时搜索，请使用时间范围菜单选择一个预设的实时**时间范围**，如 **30 秒**或 **1 分钟**。您还可以指定滑动时间范围窗口以应用到实时搜索。

如果您有 Apache Web 访问数据，运行以下搜索以在 Web 流量事件流入时查看它们。

```
sourcetype=access_*
```

从输入管道流入的原始事件没有时序性。您可使用 `rtorder` 命令缓冲实时搜索中的事件，然后按照时间顺序的升序发出这些事件。

以下示例保留 Web 流量事件最后 5 分钟的缓冲区，一旦这些事件超出 5 分钟时，立即按升序时间顺序发出事件。如果某事件在 5 分钟后已发出，则将丢弃超过 5 分钟的新接收事件。

```
sourcetype=access_* | rtorder discard=t buffer_span=5m
```

实时搜索依赖于事件流。因此，使用任何其他前导搜索命令将无法运行实时搜索，如 `| metadata`（不生成事件）或 `|`

`inputcsv`（仅在文件中读取）。另外，如果您试图将搜索结果发送到 `| outputcsv`，则直到完成实时搜索，才会向 CSV 文件写入。

Splunk Web 中的实时报表

运行报表以预览用于访问大多数 Web 页面的 IP 地址。在此示例中，`top` 命令将返回一个三列的表格：clientip、count 和 percent。当数据流入时，表格将以新值更新。

```
sourcetype=access_* | top clientip
```

对于每个 Web 流量事件，添加一个 `count` 字段以代表到目前为止所看到的事件数（但计数中不包括当前事件）。

```
sourcetype=access_* | streamstats count current=false
```

您还可以钻取到实时报表。但实时钻取并不衍生其他实时搜索。相反，当您将钻取到已检索并建立索引的事件时，它将衍生历史搜索。关于更多信息，请参阅*《仪表板和可视化》*中的“使用钻取保障仪表板交互性”。

CLI 中的实时搜索和报表

要在 CLI 中运行实时搜索，请将命令 "search" 替换为 "rtsearch"：

```
./splunk rtsearch 'eventtype=pageview'
```

使用 `highlight` 命令可强调搜索结果中的术语。以下示例将突出显示 pageview 事件中的 "GET"：

```
./splunk rtsearch 'eventtype=pageview | highlight GET'
```

默认情况下，搜索结果已启用换行。使用 `-wrap` 选项可关闭换行：

```
./splunk rtsearch 'eventtype=pageview' -wrap 0
```

CLI 中的实时报表还将在预览模式下显示并在数据流入时更新。

```
./splunk rtsearch 'error | top clientip'
```

使用 `-preview` 选项可停止结果预览：

```
./splunk rtsearch 'error | top clientip' -preview false
```

如果关闭预览，仍可通过 Splunk Web 中的“任务”页面管理（保存、暂停、完成或删除）搜索。完成搜索后，将显示报表表格。有关更多信息，请参阅本手册中的“使用任务页面管理任务”。

要运行窗口式实时搜索，请使用 `earliest_time` 和 `latest_time` 参数。

```
rtsearch 'index=_internal' -earliest_time 'rt-30s' -latest_time 'rt+30s'
```

注意：只能在 API 级别设置实时搜索，因此，如果您尝试在搜索字符串内指定时间范围调节器，搜索将不会运行。应在 REST API 中将 `earliest_time` 和 `latest_time` 参数设置为同名参数。

通过访问 CLI 帮助参考，可查看所有 CLI 命令。有关更多信息，请参阅本手册中的“使用 CLI 获取帮助”。

实时搜索和报表的预期性能和已知限制

实时搜索匹配的是已到达端口但尚未保存到磁盘上的事件。事件到达的速率和匹配的数量决定了占用多少内存以及对索引速率的影响。

索引吞吐量

只要索引器当前未负载过重且没有很多并发的实时搜索，Splunk 软件的性能便应该是可接受的。不过，如果您拥有大量并发实时搜索，则实时搜索将对大容量环境和网络负载的性能产生重大影响。

规划实时搜索时，应考虑它将如何影响以下两项的性能：

- 必须对实时事件进行流处理的**搜索节点**。
- 必须处理聚合实时事件流的**搜索头**。

在搜索节点上完成的任务越多，需要在搜索头上完成的任务就越少，反之亦然。搜索节点对于整体系统功能非常重要，因此，您不需要让其负担过多的实时事件筛选任务。不过，如果搜索节点根本未过滤任何实时事件，则向搜索头发送所有实时事件所需的处理能力和带宽将需要巨大的成本，尤其是在并发运行多个实时搜索时。

在搜索头无法与搜索节点保持联系的情况下，索引处理器上的队列将停止标记搜索事件。但这些事件具有序号，可用于通知从搜索注意事项中忽略事件的时间和数量。

并发实时和历史搜索

您可在硬件的限制内并发运行实时和历史搜索。针对相同或不同用户的单独搜索，则无限制。

并发实时搜索

运行多个实时搜索将会对索引容量造成负面影响。实时搜索功能已进行优化，可对稀疏（或罕见术语）搜索进行实时告警，并将牺牲索引容量来改善延迟和可靠性。

索引实时搜索

并发的实时搜索的数量可以显著地影响索引性能。要减少对索引器的影响，可以启用索引实时搜索。这将基本上运行诸如历史搜索等搜索，但还将随着磁盘上出现新事件，不断对其进行更新。

请阅读“关于实时搜索和报表”中关于如何启用索引实时搜索的更多信息。

实时搜索窗口

窗口式实时搜索比非窗口式实时搜索占用的资源更多。管理和预览窗口内容所需的操作可能会导致窗口式实时搜索无法跟上高速索引。如果您的窗口式搜索未显示预期数量的事件，请尝试非窗口式搜索。如果您只关心事件计数，请尝试在搜索中使用 "timechart count"。

阅读有关如何“为实时搜索指定时间范围”的更多信息。

如何限制实时搜索的使用情况

由于过度使用实时搜索会产生性能成本，因此，有必要限制实时搜索的使用情况。

限制实时搜索的选项如下：

- 在索引器级别禁用实时搜索，方法是编辑特定索引的 `indexes.conf`。
- 禁用特定角色和用户的实时搜索。
- 编辑 `limits.conf` 以减少可在任意给定时间并发运行的实时搜索的数量。
- 编辑 `limits.conf` 以限制索引器对实时搜索的支持。

如果您使用的是 Splunk Cloud 并想限制实时搜索，请向 Splunk 支持提交问题。

在 indexes.conf 中禁用实时搜索

在索引器上执行实时搜索可能需要占用大量资源。如果想要禁用索引器上的实时搜索，可编辑 `[default]` 设置（位于索引器的 `indexes.conf` 中）。

```
[default]
enableRealtimeSearch = <bool>
```

注意：与多个索引器相连的*搜索头*仍能从启用实时搜索的索引器中获得实时搜索结果。

禁用用户或角色的实时搜索

在 Splunk Web 中，可访问**管理器 > 访问控制**并将实时搜索这项**操作**映射到具体用户或角色。默认情况下，已将 **rtsearch** 操作分配给管理员和超级用户角色，而未分配给普通用户角色。无 rtsearch 操作的角色将无法在相应的搜索头上运行实时搜索，无论该搜索头连接到哪些索引器。

对实时搜索设置搜索限制

可使用 `[search]` 段落（位于 `limits.conf`）来更改可在系统上并发运行的最大实时搜索数。

```
[search]
max_rt_search_multiplier = <decimal number>
realtime_buffer = <int>
```

`max_rt_search_multiplier`

- 与最大历史搜索数相乘来确定最大并发实时搜索数的数字。默认为 1。
- **注意：**最大实时搜索数的计算方式为：`max_rt_searches = max_rt_search_multiplier x max_hist_searches`

`realtime_buffer`

- 为 UI 中的实时搜索保留的最大可访问事件数。必须大于等于 1。默认为 10000。
- 一旦达到此限制，实时缓冲区会充当循环缓冲区。

对实时搜索设置索引器限制

可使用 `[realtime]` 段落（位于 `limits.conf`）来更改实时搜索索引器支持的默认设置。**可通过 REST API 参数为单个搜索覆盖这些选项。**

```
[realtime]
queue_size = <int>
blocking = [0|1]
max_blocking_secs = <int>
indexfilter = [0|1]
```

`queue_size = <int>`

- 每个实时搜索的队列大小。必须大于 0。
- 默认为 10000。

`blocking =[0|1]`

- 指定如果队列已满，索引器是否应阻止。
- 默认为 false (0)。

`max_blocking_secs = <int>`

- 队列已满时要阻止的最长时间。如果 `blocking = false`，则此选项没有意义。
- 如果设置为 0，表示“无限制”。
- 默认为 60。

`indexfilter = [0|1]`

- 指定索引器是否应预先过滤事件以提高效率。
- 默认为 true (1)。

评估和操作字段

关于评估和操作字段

本部分介绍可用于评估新字段、操作现有字段、添加新字段来丰富事件以及分析多值字段的搜索命令。

- 评估新字段的核心在于 eval 命令及其函数。与 stats 命令不同，stats 命令允许根据事件中的字段计算统计信息，而 eval 允许使用现有字段和任意表达式创建新字段。eval 命令具有许多函数。请阅读“使用 eval 命令和函数”中有关这些函数的更多信息。
- 您可以在搜索时间轻松地在您的数据中增加更多信息。请阅读有关如何“使用查找从外部查找表中添加字段”的更多信息。
- Splunk 搜索语言允许您使用各种搜索命令以不同方式提取字段。
- 您的事件可能包含具有多个值的字段。Splunk 搜索语言包括与多值字段配合使用的各种搜索命令和函数。请阅读关于如何“操作和评估多值字段”的更多信息。

使用 eval 命令和函数

利用 eval 命令可以设计任意表达式，使这些表达式使用自动提取的字段创建一个新字段，新字段获取的值为该表达式的计算结果。eval 命令用途广泛，非常有用。一些 eval 表达式相对较为简单，但是这些表达式通常是非常复杂的。

本主题介绍如何使用 eval 命令和评估函数。

eval 表达式的类型

eval 表达式是代表目标字段值的文字、字段、运算符以及函数的组合。表达式可以包括数学运算、字符串连接、比较表达式、布尔表达式或对 eval 函数之一的调用。Eval 表达式需要字段的值对运算类型有效。

例如，除了加法之外，如果值不是数字，则算术运算将不会生成有效的结果。对于加法，如果两个操作数都是字符串，eval 可以连接这两个操作数。如果使用 '.' 对值进行连接，则无论这两个值实际是何种类型，eval 都会将其视为字符串。

示例 1：结合使用 eval 表达式和 stats 函数。

搜索所有索引并统计 `status` 字段值为 `404` 的事件数。将结果重命名为被称为 `count_status` 的字段，按来源类型组织结果。

```
index=* | stats count(eval(status="404")) as count_status by sourcetype
```

示例 2：定义值为两个圆的面积总和的字段。

使用 `eval` 命令定义值为两个圆 A 和 B 的面积总和的字段。

```
... | eval sum_of_areas = pi() * pow(radius_a, 2) + pi() * pow(radius_b, 2)
```

圆的面积为 πr^2，其中 r 为半径。对于圆 A 和 B，半径分别为 radius_a 和 radius_b。此 eval 表达式使用 `pi` 和 `pow` 函数计算每个圆的面积，然后将它们加在一起，并将结果保存在名为 `sum_of_areas` 的字段中。

示例 3：定义使用城市和州/省字段的位置字段

利用 `eval` 命令定义使用城市和州/省字段的位置字段。例如，如果 city=Philadelphia、state=PA，则 location="Philadelphia, PA"。

```
... | eval location=city.", ".state
```

这个 eval 表达式是简单的字符串连接。

示例 4：使用 eval 函数对电子邮件来源进行分类

此示例使用样本电子邮件数据。您可以将 `sourcetype=cisco:esa` 替换为 `sourcetype` 值并将 `mailfrom` 字段替换为您数据的电子邮件地址字段名，从而实现对任何电子邮件数据运行此搜索。例如，电子邮件可以是 `To`、`From` 或 `Cc`）。

此示例将根据电子邮件地址的域对电子邮件的来源进行分类。将 .com、.net 和 .org 地址视为 **local**，而将任何其他地址视为 **abroad**。域名有很多。当然，.com、.net 或 .org 以外的域不一定属于 **abroad**。这只是一个示例。

本搜索中的 `eval` 命令包含多个以逗号隔开的表达式。

```
sourcetype="cisco:esa" mailfrom=*| eval accountname=split(mailfrom,"@"), from_domain=mvindex(accountname,-1),
location=if(match(from_domain, "[^\n\r\s]+\.(com|net|org)"), "local", "abroad") | stats count BY location
```

此搜索的前半部分与之前的示例相似。`split()` 函数用于拆分 `mailfrom` 字段中的电子邮件地址。`mvindex` 函数将 `from_domain` 定义为 `mailfrom` 字段 `@` 符号之后的部分。

然后，使用 `if()` 和 `match()` 函数。

- 如果 `from_domain` 值以 `.com, .net., or .org` 结尾，则为 `location` 字段分配值 `local`。
- 如果 `from_domain` 不匹配，则为 `location` 分配 `abroad`。

接下来，通过管道符将 `eval` 结果传递给 `stats` 命令，以统计每个 `location` 值的结果数。

“统计”选项卡中显示的结果如下：

location	count
abroad	3543
local	14136

注意：此示例只是说明如何使用 `match()` 函数。如果您要对事件进行分类并快速查找这些事件，更好的方法是使用事件类型。请阅读*《知识管理器手册》*中的**关于事件类型**。

定义已计算字段

如果您发现您定期使用特定的 eval 表达式，则可以考虑将该字段定义为已计算字段。这样做意味着，当编写搜索时，您可以将 eval 表达式去掉，并按照其他提取的字段的引用方式来引用该字段。运行搜索后，将在搜索时间提取字段并将其添加到将字段包含在 eval 表达式的事件中。

请阅读《知识管理器手册》中有关如何在“定义已计算字段”中对此进行配置的更多内容。

使用查找从查找表中添加字段

您可以使事件中的字段与外部来源（如查找表）中的字段相匹配，并利用这些匹配将更多信息嵌入事件之中。

查找表可以是一个静态 CSV 文件，一个 KV 存储集合，也可以是 Python 脚本的输出。还可以用搜索结果填充 CSV 文件或 KV 存储集合，然后将该文件设置为一个查找表。有关字段查找相关详细信息，请参阅*《知识管理器手册》*中的“配置 CSV 和外部查找”以及“配置 KV 存储查找”。

在配置字段查找之后，可以使用 `lookup` 命令从搜索应用对其进行调用。

示例：假设有一个名为 `dnslookup` 的字段查找，它引用一个 Python 脚本，该脚本执行 DNS 和反向 DNS 查找，并接受主机名或 IP 地址作为参数 - 您可使用查询命令使事件中的主机名称值与表格中的主机名称值相匹配，然后将相应的 IP 地址添加到事件中。

```
... | lookup dnslookup host OUTPUT ip
```

有关使用 Splunk 脚本 `external_lookup.py` 的更广泛示例，请参阅 Splunk 博客中的“对主机条目进行反向 DNS 查找”。

使用搜索命令提取字段

可使用搜索命令以不同方式提取字段。

- 在 Perl 正则表达式中，`rex` 命令用命名组执行字段提取。
- `extract`（或 `kv`，代表“键/值”）命令使用默认模式显式提取字段值对。
- `multikv` 命令可从多行、表格形式的事件中提取字段和值对。
- `spath` 命令在诸如 XML 和 JSON 等已构建事件数据上提取字段和值对。
- `xmlkv` 和 `xpath` 命令从 XML 格式的事件数据中提取字段和值对。
- `kvform` 命令可根据预定义的表单模板提取字段和值对。

在 Splunk Web 中，您可以在**设置 > 字段 > 字段提取**页面中定义字段提取。

以下部分介绍了如何使用正则表达式和命令提取字段。请参阅*《知识管理器手册》*中的“关于字段”。

使用正则表达式提取字段

在 Perl 正则表达式（包含在搜索条件里面）中，`rex` 命令使用命名组执行字段提取。`rex` 命令会将原始事件的段与正则表达式进行匹配，并将这些匹配值保存到一个字段中。

在此示例中，字符串 `From:` 和 `To:` 之后出现的值分别保存到 **from** 和 **to** 字段中。

```
... | rex field=_raw "From: (?<from>.*) To: (?<to>.*)"
```

如果原始事件包含 `From: Susan To: Bob`，搜索会提取字段名称和值对：**from**=*Susan* 和 **to**=*Bob*。

有关正则表达式语法和用法的入门资料，请参阅 www.regular-expressions.info。以下是有用的第三方工具，可用于编写和测试正则表达式：

- regex101
- RegExr
- Debuggex

从 .conf 文件提取字段

使用 `extract` 命令，以在搜索结果集中强制进行字段值提取。如果使用 `extract` 命令时未指定任何参数，则会使用 `props.conf` 文件中的字段提取段落来提取字段。您可以使用 `extract` 命令测试任何手动添加到 .conf 文件的字段提取。

从表格形式的事件中提取字段

使用 `multikv` 命令可强制从多行、表格形式的事件中提取字段值。`multikv` 命令会针对每个表格行创建一个新事件，并从表格标题中派生字段名称。

从 XML 格式的事件中提取字段

`xmlkv` 命令可强制从事件数据 XML 格式的标记中（如网页中的交易）提取字段值。

从 XML 和 JSON 文档中提取字段

`spath` 命令从结构化数据格式（如 XML 和 JSON）中提取信息并将这些提取的值储存在字段中。

根据表单模板从事件中提取字段

`kvform` 命令会根据预定义并存储在 `$SPLUNK_HOME/etc/system/local/` 或您自己的自定义应用程序目录 `$SPLUNK_HOME/etc/apps/` 中的表单模板，从事件中提取字段值对。例如，如果是 `form=sales_order`，则搜索会查找 `sales_order.form`，还会将所有处理的事件与该表单进行匹配以提取值。

如果您使用的是 Splunk Cloud 并想将表单模板用于字段提取，请向 Splunk 支持提交问题。

评估和操作多值字段

关于多值字段

多值字段会在**搜索时间**进行分析，以便您可以在搜索管道中处理这些值。用于处理多值字段的搜索命令包括 makemv、mvcombine、mvexpand 和 nomv。eval 和 where 命令支持可用于多值字段的 `mvcount()`, `mvfilter()`, `mvindex()`, and `mvjoin()` 等函数。请查看*搜索参考*中的评估函数和此主题中的示例。

如果您使用的是 Splunk Enterprise，可在 `fields.conf` 文件中配置多值字段，以指定 Splunk 软件如何检测单一提取的字段值中的多个字段值。编辑 `fields.conf`（位于 `$SPLUNK_HOME/etc/system/local/`，或您自己的自定义应用程序目录 `$SPLUNK_HOME/etc/apps/`）。有关如何执行此操作的更多信息，请参阅*知识管理员手册*中的“通过 fields.conf 配置多值字段提取”。

如果您使用的是 Splunk Cloud 并想配置多值字段，请向 Splunk 支持提交问题。

如果搜索生成了结果，如表格，则此结果会写入 `results.csv.gz` 文件。`results.csv.gz` 文件的内容包括以 "__mv_" 开头的字段。这些字段仅供内部使用，且用于编码多值字段。

评估多值字段

多值字段的一个较为常见的示例是电子邮件地址字段，该字段通常会在单个 sendmail 事件中出现两次或三次--一次用于发件人，另一次用于收件人列表，第三次可能会出现在抄送地址列表中。

计算字段中的值数量

使用 `mvcount()` 函数可计算某单值或多值字段中的值数量。

在本示例中，mvcount() 返回 To、From 和 Cc 字段中的电子邮件地址数量，并将该地址保存在指定的 "_count" 字段中。

```
eventtype="sendmail" | eval To_count=mvcount(split(To,"@"))-1 | eval From_count=mvcount(From) | eval Cc_count=
mvcount(split(Cc,"@"))-1
```

此搜索采用 `To` 字段中的值，并使用 `split` 函数拆分带 @ 标记的电子邮件地址。`split` 函数还可用于 `Cc` 字段实现相同目的。

如果 `From` 字段中仅存在一个电子邮件地址（可能正如您的预期），`mvcount(From)` 会返回 1。另外，如果没有 Cc 地址，则事件可能不存在 Cc 字段。在这种情况下，`mvcount(cc)` 会返回空值。

从多值字段中筛选值

使用 `mvfilter()` 函数可使用任意布尔表达式筛选多值字段。`mvfilter` 函数一次只处理一个字段。

在本例中，mvfilter() 会保留字段 `email` 以 `.net` 或 `.org` 结尾的所有值。

```
eventtype="sendmail" | eval email=mvfilter(match(email, "\.net$") OR match(email, "\.org$"))
```

注意：本例还使用 `match()` 函数将用引号定义的模式与 `email` 的值进行比较。请参阅*搜索参考*中的评估函数。

从多值字段中返回值的子集

使用 `mvindex()` 函数可引用多值字段中的某个特定值或值的子集。由于索引编号是从 0 开始，因此如果您要引用某个字段的第三个值，应将其指定为 2。

在本示例中，mvindex() 返回发件人所发送每封电子邮件的 "To" 字段中的第一个电子邮件地址：

```
eventtype="sendmail" from=Sender@* | eval to_first=mvindex(to,0)
```

如果您要查看发件人每次写入的前 3 个电子邮件地址，请使用如下搜索。

```
eventtype="sendmail" from=Sender@* | eval top_three=mvindex(to,0,2)
```

在本示例中，`top_three` 本身是一个多值字段。

操作多值字段

使用 nomv 将多值字段转换为单个值

可使用 `nomv` 命令将指定的多值字段的值转换为单个值。`nomv` 命令会覆盖在 `fields.conf` 文件中设置的多值字段配置。

在本 sendmail 事件示例中，您希望将 senders 字段的值合并为单个值。

```
eventtype="sendmail" | nomv senders
```

使用 makemv 分隔多值字段

可使用 `makemv` 命令将多值字段分隔成多个单值字段。在本 sendmail 搜索结果示例中，您希望将发送者字段的值分隔成多个字段值。

```
eventtype="sendmail" | makemv delim="," senders
```

在分隔字段值后，可通过其他命令发送这些值。例如，可以显示前几位发件人。

```
eventtype="sendmail" | makemv delim="," senders | top senders
```

使用 mvexpand 根据多值字段创建多个事件

可使用 `mvexpand` 命令将多值字段的值扩展为单独的事件，这些事件分别对应于多值字段的每个值。在本例中，针对多值字段 "foo" 的每个值创建新的事件。

```
... | mvexpand foo
```

使用 mvcombine 从类似事件中创建多值字段

用 ":" 分隔符合并 "foo" 的值。

```
... | mvcombine delim=":" foo
```

另请参阅

*知识管理员手册*中的“通过 fields.conf 配置多值字段提取”。

计算统计信息

关于计算统计信息

本部分介绍如何计算事件摘要统计信息。在您考虑使用 Splunk 的搜索处理语言 (SPL) 计算统计信息时，您可能首先想到 `stats` 命令。stats 命令会生成以表格形式显示摘要统计信息的报表。另外，您可以使用 `chart` 和 `timechart` 命令创建摘要统计信息的图表可视化，使用 `geostats` 命令为包括地理位置字段的事件创建摘要统计信息的地图可视化。

`stats`、`chart` 和 `timechart` 命令（及其相关的命令 `eventstats`、`geostats` 和 `streamstats`）设计为与统计函数结合使用。有关使用上述命令和函数的搜索的示例，请阅读“使用 stats 命令和函数”。

后面的主题将讨论如何：

- “将 stats 与 eval 表达式和函数配合使用”来计算统计信息。
- “将迷你图添加到报表表格”。

“高级”统计部分包含与检测异常、查找和删除离群值、检测模式和时间系列预测相关的话题。

使用 stats 命令和函数

本主题介绍如何将统计函数与**转换命令** `chart`、`timechart`、`stats`、`eventstats` 和 `streamstats` 结合使用。

- 有关 stat 命令和语法的更多信息，请参阅*《搜索参考》*中的 stat 命令。
- 有关 stats 函数的列表，请参阅*《搜索参考》*中的“统计和图表函数”。

关于 stats 命令和函数

`stats`、`streamstats` 和 `eventstats` 命令都可用于计算搜索结果摘要统计信息或从索引中检索的事件的摘要统计信息。`stats` 命令对搜索结果进行整体计算。`streamstats` 命令在每次看到事件时，以流化方式为每个事件计算统计信息。`eventstats` 命令计算所有搜索结果的统计信息，并将聚合内联添加到与其相关的每个事件中。有关这些命令差别的详细信息，请参阅下一部分。

`chart command` 以支持可视化的数据结构，即以图表（例如柱形图、折线图、面积图和饼图）的形式返回结果。图表的 X 轴可决定要跟踪的字段。`timechart command` 以时间系列格式图表返回结果，其中数据绘制于始终为时间字段的 X 轴上。请参阅《数据可视化手册》的“可视化参考”，了解有关可视化功能和选项的更多信息。

`stats`、`chart` 和 `timechart` 命令（及其相关的命令 `eventstats` 和 `streamstats`）设计为与统计函数结合使用。统计函数列表允许您对字段出现次数进行计数，并计算字段值的总和、平均值、范围等。

有关统计函数列表及其用法的更多信息，请参阅*搜索参考*中的“统计和图表函数”。

Stats、eventstats 和 streamstats

`eventstats` 和 `streamstats` 命令是 `stats` 命令的变体。

`stats` 命令对搜索结果进行整体处理，并仅返回您指定的字段。例如，下列搜索将返回一个含两列（10 行）的表。

```
sourcetype=access_* | head 10 | stats sum(bytes) as ASumOfBytes by clientip
```

`ASumOfBytes` 和 `clientip` 字段是 stats 命令后面可以使用的唯一字段。例如，以下搜索将在 `bytes` 列中返回空单元格，因为它不是结果字段。

```
sourcetype=access_* | head 10 | stats sum(bytes) as ASumOfBytes by clientip | table bytes, ASumOfBytes, clientip
```

要在结果中看到除 `ASumOfBytes` 和 `clientip` 之外的其他字段，需要将它们包含在 stats 命令中。同时，如果想要对原始事件中的任何原始字段进行计算，则需要在 stats 命令前进行。

`eventstats` 命令与 `stats` 命令计算相同的统计信息，但它还会将结果聚合到原始数据。运行下列搜索时，将返回事件列表，而不是结果表，因为 eventstats 命令不会更改原始数据。

```
sourcetype=access_* | head 10 | eventstats sum(bytes) as ASumOfBytes by clientip
```

可使用 `table` 命令将结果的格式设置为显示所需字段的表格。现在您还可在结果中查看 `bytes`（或原始事件中任意原始字段）的值。

```
sourcetype=access_* | head 10 | eventstats sum(bytes) as ASumOfBytes by clientip | table bytes, ASumOfBytes,
clientip
```

`streamstats` 命令还可将计算的统计信息聚合到原始事件，但仅在可以看到事件时进行聚合。要演示这一点，将 `_time` 字段包含在较早的搜索中，并使用 `streamstats`。

```
sourcetype=access_* | head 10 | sort _time | streamstats sum(bytes) as ASumOfBytes by clientip | table _time,
clientip, bytes, ASumOfBytes
```

此搜索将基于所看到的时间，计算每个事件的总和，而不计算每个 `clientip`（由 `stats` 和 `eventstats` 返回）的总和。`streamstats` 命令非常适用于报告已知时间范围的事件。

示例

示例 1

本示例创建一天中每小时新用户联机数量的图表。

```
... | sort _time | streamstats dc(userid) as dcusers | delta dcusers as deltadcusers | timechart sum(deltadcusers)
```

`dc`（或 `distinct_count`）函数返回 `userid` 唯一值计数，并对结果字段 `dcusers` 进行重命名。

如果不重命名函数，例如，“dc(userid) 重命名为 dcusers”，生成的计算将自动保存到函数调用，例如 "dc(userid)"。

`delta` 命令用于查找当前和之前 dcusers 值之间的差别。随后，将绘制此 delta 总和随时间的变化情况。

示例 2

本示例计算某个字段的中值，然后将字段的值小于中值的事件的计数制成图表。

```
... | eventstats median(bytes) as medbytes | eval snap=if(bytes>=medbytes, bytes, "smaller") | timechart count by
snap
```

使用 Eventstats 计算先前搜索的所有字节值的中值。

示例 3

本示例计算已计算字段的标准偏差和方差。

```
sourcetype=log4j ERROR earliest=-7d@d latest=@d | eval warns=errorGroup+"-"+errorNum | stats count as
Date_Warns_Count by date_mday,warns | stats stdev(Date_Warns_Count), var(Date_Warns_Count) by warns
```

此搜索返回过去 7 天的错误，并从已提取的字段 errorGroup 和 errorNum 创建新字段、警告。stats 命令使用了两次。首先，它会计算每天的单日警告数量。然后，它会计算该警告计数的标准偏差和方差。

示例 4

可将计算的字段用作搜索的过滤参数。

```
sourcetype=access_* | eval URILen = len(useragent) | eventstats avg(URILen) as AvgURILen, stdev(URILen) as
StdDevURILen| where URILen > AvgURILen+(2*StdDevURILen) | chart count by URILen span=10 cont=true
```

在此示例中，eventstats 用于计算 `useragent` 中 URI 长度的平均和标准偏差。然后，这些数字将用作检索事件的过滤器。

将 stats 与 eval 表达式和函数配合使用

本主题介绍如何在 stats 计算中使用 eval 表达式和函数。

- 有关 eval 命令和语法的更多信息，请参阅*搜索参考*中的 eval 命令。
- 有关 eval 函数的列表，请参阅*《搜索参考》*中“评估函数”。
- 另外，您也可以阅读本手册另一个部分有关“使用 eval 命令评估和操作字段”中的更多内容。

示例 1：匹配事件的非重复计数

本例统计出现错误的 IP 地址的个数。这类似于搜索事件，过滤特定错误代码，然后使用 stats 命令来统计 IP 地址的个数。

```
status=404 | stats dc(ip)
```

使用 eval 表达式来执行此任务的最佳方法为：

```
status=404 | stats dc(eval(if(status=404, ip, NULL))) AS dc_ip
```

示例 2：对字段分类和计数

此示例使用样本电子邮件数据。您可以将 `sourcetype=cisco:esa` 替换为 `sourcetype` 值并将 `mailfrom` 字段替换为您数据的电子邮件地址字段名，从而实现对任何电子邮件数据运行此搜索。例如，电子邮件可以是 `To`、`From` 或 `Cc`）。

查找组织内的电子邮件有多少是来自 .com、.net、.org 或其他顶级域。

本搜索中的 `eval` 命令包含两个以逗号隔开的表达式。

```
sourcetype="cisco:esa" mailfrom=* | eval accountname=split(mailfrom,"@"), from_domain=mvindex(accountname,-1) | stats count(eval(match(from_domain, "[^\n\r\s]+\.com"))) AS ".com", count(eval(match(from_domain, "[^\n\r\s]+\.net"))) AS ".net", count(eval(match(from_domain, "[^\n\r\s]+\.org"))) AS ".org", count(eval(NOT match(from_domain, "[^\n\r\s]+\.(com|net|org)"))) AS "other"
```

- 此搜索的第一部分使用 `eval` 命令拆分 `mailfrom` 字段中的电子邮件地址。`from_domain` 被定义为 `@` 符号之后的 `mailfrom` 字段的一部分。
  - 使用 `split()` 函数把 `mailfrom` 字段拆分为一个名为 `accountname` 的多值字段。`accountname` 的第一个值是 "@" 符号前的所有内容，第二个值则为该符号后的所有内容。
  - `mvindex()` 函数用于将多值字段 `accountname` 中的 `from_domain` 设置为第二个值。
- 然后，通过管道符将结果传递给 `stats` 命令。`count()` 函数用于计算 `eval` 表达式的结果数。
- `eval` 使用 `match()` 函数将 `from_domain` 与查找域名中各种后缀的正则表达式进行比较。如果 `from_domain` 的值与正则表达式匹配，则更新 `count`，且是为每个后缀更新，包括 `.com`、`.net` 和 `.org`。其他域名后缀将按 `other` 统计。

“统计”选项卡中显示的结果如下：

.com	.net	.org	其他
4246	9890	0	3543

将迷你图添加到搜索结果

如果要使用 `stats` 和 `chart` 搜索，可通过向结果表添加迷你图来增加搜索的有用性和总体信息密度。迷你图是内联图表，它们显示在搜索结果的表单元格内，用于显示与每一行的主关键字相关联的基于时间趋势。

例如，假定您将此搜索设置为针对**过去 15 分钟**发生的事件来运行：

```
index=_internal | chart count by sourcetype
```

此搜索将返回一个两列的结果表，其中显示在过去 15 分钟内索引为 `_internal` 的来源类型的事件计数。第一列列出每个 `sourcetype`（在过去 1 小时的 `_internal` 索引事件集合中找到的）；这是该表格的主键。第二列 `count` 显示每个来源类型的事件计数：

sourcetype	count
scheduler	18
splunk_archiver-2	18
splunk_web_access	223
splunk_web_service	226
splunkd	73486
splunkd_access	1745
splunkd_ui_access	5925

可将迷你图添加到此搜索的结果中，方法是向搜索本身添加 `sparkline` 函数：

```
index=_internal | chart sparkline count by sourcetype
```

由此生成的表格与上一个表格几乎完全相同，只不过在当前的表格中，每一行都有一个迷你图，显示列出的每个来源类型在过去 15 分钟内事件计数的趋势。

sourcetype	sparkline	count
splunk_web_access		8
splunk_web_service		6
splunkd		2421
splunkd_access		46
splunkd_ui_access		232

现在，您能够轻松地看到以前可能无法看到的数据模式。在 15 分钟时间跨度内大约四分之三的位置处，一些搜索活动在大多数的 `index=_internal` 来源类型都明显造成了波形。而 `splunkd` 几乎类似于一个正常的心跳在整个时间的跨度内正常运行的图表状况。

表格中的每个迷你图显示的信息与该迷你图中表示的其他事件相关，但与其他迷你图无关。一个迷你图中的峰值不一定与另一个迷你图中的峰值相等。

将迷你图与 stats 和 chart 命令结合使用

应始终将迷你图功能与 `chart` 和 `stats` 搜索结合使用，因为它是这两个搜索命令的函数。它本身不是命令。迷你图的功能对于这两个搜索命令是相同的。

迷你图本身不可用作仪表板图表可视化，但可以将仪表板面板设置为表格可视化，从而显示迷你图。有关更多信息，请参阅《Splunk 数据可视化手册》中的“可视化参考”主题，

有关 `chart` 和 `stats` 命令的更多信息（包括有关 `sparkline` 函数语法的详细信息），请参阅*搜索参考*中的 "chart" 和 "stats"。

示例：Stats、迷你图和地震数据

以下是 `stats` 搜索的一些示例，这些搜索使用迷你图提供有关地震数据的附加信息。

> 此示例使用从 USGS 地震网站下载的近期地震数据。该数据是一个逗号分隔的 ASCII 文本文件，其中包含每次记录的地震的震级 (mag)、坐标（经度、纬度）、区域（地点）等。
>
> 您可以从 **USGS 地震源**下载当前 CSV 文件并作为输入添加。

比方说，您要使用 USGS 地震数据显示过去一个月内地震最为频繁的位置，一列显示每个位置的平均地震强度。您可以使用如下搜索：

```
source="all_month.csv" | stats sparkline count, avg(mag) by locationSource | sort count
```

此搜索返回以下表格，其中的迷你图说明了地震最为频繁的各位置在过去一个月内的地震次数：

locationSource	sparkline	count	avg(mag)
bgs		1	2.7
ld		2	2.275
ott		9	2.0777777777777775
se		10	2.015
ismp		17	2.025882352941177
nm		25	1.8228000000000002
tul		28	2.7178571428571425
uu		197	1.1888324873096445
mb		202	0.7736138613861386
uw		222	1.1463963963963963
pr		269	2.7057249070631966
hv		340	1.6103235294117646
nn		858	0.6322843822843826
us		988	4.123380566801619
ci		1319	0.9805761940864288
nc		1994	0.9795634721525331
ak		3015	1.5647097844112765

您马上就会发现不同位置之间的地震分布情况的差异。

您可以单击 sourceLocation 查看位置计算中包含的实际事件。

对于 **avg(mag)** 列，您可以使用“格式”图标更改该列中的数字格式。

只需将鼠标悬停在迷你图上，即可获得特定区域的最小和最大计数；在本示例中，您可以发现，在阿拉斯加南部为期 7 天的时间内，每天最少发生一次地震，最多一天发生了 6 次地震。

但是，如果您希望迷你图不仅要表示地震次数，还要表示给每个区域带来影响的地震的平均震级，该怎么办？换言之，如何通过迷你折线图表示图表的每个“时间数据桶”（段）的平均地震震级？

请尝试如下所示的搜索：

```
source="all_month.csv"| stats sparkline(avg(mag),6h) as magnitude_trend count, avg(mag) by locationSource | sort -
```

```
count
```

此搜索针对每个区域生成一个迷你图，其中显示迷你图每一段中包含的地震事件的平均地震震级。在 **sort** 后指定一个短划线（-），则将对结果进行降序排列。

但是此搜索不只完成了这些。它还要求迷你图将自身分割成若干较小的时间部分。在前一表格中，迷你图按日期进行分割，因此迷你图中的每个数据点表示了整个 24 小时期间内的事件数量。这就是为什么这些迷你图如此简短。

将 `6h` 添加到搜索语言中可覆盖此默认设置，并显示以六小时为单位拆分成离散数据块的迷你图，以便于按选定的时间范围查看事件的分布情况。

此搜索还会将迷你图列重命名为 **magnitude_trend**，以便于理解。

locationSource	magnitude_trend	count	avg(mag)
ak		3015	1.56
nc		1994	0.98
ci		1319	0.98
us		988	4.12
nn		858	0.63
hv		340	1.61
pr		269	2.71
uw		222	1.15
mb		202	0.77
uu		197	1.19

现在您可以看出 **tul** 位置的地震传播比之前搜索中显示的更加均匀。

高级统计

关于高级统计

前一部分介绍了如何使用 `stats` 和 `eval` 计算基本统计，以及如何创建迷你图。

本部分我们将讨论如何检测数据中的异常。内容包括查找离群值以识别数据中的异常或峰值。您可能想删除对计算或为数据绘制图表的方式造成不必要影响的离群值。您可以检测数据中的模式，基于事件之间的相似性来分组事件。如果您所监视的事件有某种模式和相关性，则可以使用它们预测将来的活动。了解此情况后，您就可以基于特定阈值主动发出告警，并执行 "what-if" 分析，对不同的情况进行比较。所有这些都可通过高级统计实现，而且还可实现更多。

- 高级统计命令
- 关于异常检测
- 查找和移除离群值
- 检测异常
- 检测模式
- 关于时间系列预测
- 机器学习工具套件

另请参阅

使用 stats 命令和函数

将 stats 与 eval 表达式和函数配合使用

将迷你图添加到报表表格

高级统计命令

现在，让我们来了解可用于执行高级统计的命令富组。以下表格按类别将这些命令进行了组织。

查找异常事件的命令

以下命令用于查找您的数据中的异常事件。您可搜索不常见或无关事件和字段，或将相似事件组成群集。

命令	描述
`analyzefields`	分析数值字段，以确定其预知另一个离散字段的能力。
`anomalies`	计算事件的 "unexpectedness" 分数。
`anomalousvalue`	查找并汇总不规则或不常见的搜索结果。
`anomalydetection`	计算每个事件的可能性并检测特别小的可能性。
`cluster`	将相似事件分成一组。
`kmeans`	将事件分区为 k 群集，每个群集由其平均值定义。每个事件归属于平均值最接近的群集。
`outlier`	移除无关的数字值。
`rare`	显示最不常用的字段值。

预测和趋势命令

以下命令预知未来值并计算可用于创建可视化的趋势线。

命令	描述
`predict`	预测时间系列的未来值。
`trendline`	计算字段的移动平均值。
`x11`	删除重复模式，以查找时间系列的趋势。

用于报表、图表和地图的命令

以下命令可用于构建**转换搜索**。以下命令返回图表和其他类型数据可视化所需的统计数据表。

命令	描述
addtotals	计算每个搜索结果的所有数字字段值的总和。
contingency	构建应变表，即两个字段值的同现矩阵。
correlate	计算不同字段之间的相关性。
eval	计算算术、字符串或布尔表达式。将值放入新字段中。
eventstats	向所有搜索结果添加摘要统计信息。
geostats	生成统计信息，以显示地理数据并在地图上创建数据摘要。
outlier	移除无关的数字值。
rare	显示最不常用的字段值。
stats	计算结果集的聚合统计信息，比如 average、count、和 sum。
streamstats	将以前事件相关的摘要统计信息添加到每个搜索结果中。
timechart	创建时间系列图和相应的统计信息表。请参阅*《搜索参考》*中的“统计和图表函数”。
trendline	计算字段的移动平均值。

关于异常检测

本部分介绍异常检测。有关检测异常、查找和删除离群值、检测模式和时间系列预测相关主题的完整列表，请参阅本手册中的“关于高级统计”。

异常检测简介

异常指系统预期行为中的偏差。异常可以是：

- 单个事件
- 一系列事件
- 一系列交易
- 复杂模式

异常检测的常用案例示例包括：

产业	使用案例示例
IT	识别来自 IP 地址范围的分布式拒绝服务 (DDoS) 攻击。
营销	很少但高价值客户购买模式。
产品	很少，或以前未知的方式且此方式使用可产生更好结果，或比已知方式更有效地产生结果的产品。
安全	快速交易。检测正在执行的交易何时由某用户执行时比其他用户更快。这可能表示一个自动程序或尝试探测安全措施的操作。

有效异常检测

要执行有效异常检测，将所有数据放在同一个地方。如果没有将机器和业务数据放在同一个地方，则无法执行综合分析。

开始跟踪 IT 和业务性能指标。另外，创建显示系统当前状态的基本数据图像。

另请参阅

关于高级统计、查找和删除离群值、检测异常

查找和移除离群值

本部分介绍离群值。有关检测异常、检测模式和时间系列预测相关主题的完整列表，请参阅本手册中的“关于高级统计”。

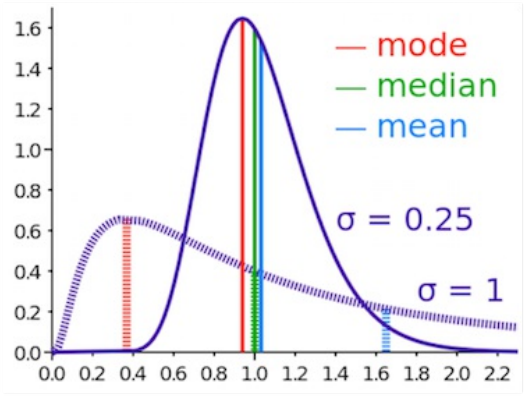
什么是离群值？

离群值是指远离数据点典型分布的数据点。

在某些情况下，您可能仅想识别离群值。在另一些情况下，您可能想删除离群值，以便它们不会影响统计结果的准确性或防止在图表中显示数据时产生问题。

统计离群值

*统计离群值*是指远离一定程度中心度的数据点。典型程度的中心度包括平均值、中值和模式。平均值指平均所得的值。中值指所有值排序后位于中间的值。模式指最常用的值。



中心度程度	Splunk 中心度命令
平均值	`stats avg(field)`
中值	`stats median(field)`
模式	`stats mod(field)`

统计离群值有多种不同的类型。离群值可以是远离平均值、远离中值或相对于模式较不常用的值。离群值也可以是远离数据点典型范围的数据点。

识别离群值

您可使用几种方式来识别离群值。一般来说，这些方式均涉及一定程度中心度的计算及随后的离群值识别。

在某些情况下，当您发现异常时就会识别出离群值。例如，当您为数据绘制图表时发现轴线不准确。离群值可能是主要原因。

将这些统计离群值作为风向标。如果发现统计离群值较往日更多，则这一现象本身就是一个异常。

要计算中心度程度，可使用以下命令。

中心度程度	Splunk 中心度命令
标准偏差	`stats stdev(field)`
四分位和百分位	`stats perc75(field) perc25(field)`
前 3 个最常用的值	`top 3 field`

在许多情况下，您需要其他命令来计算所查找的信息。以下内容给出了部分示例。

计算可接受范围的上限值和下限值以识别离群值

以下示例以来自 `quote.csv` 文件的前 500 个事件为例。`streamstats` 和一个包含 100 个事件的移动窗口用于计算平均

和标准偏差。平均和标准偏差与 eval 命令一起用于计算上限值和下限值。将 stdev 乘以 2，以此方式在计算中添加敏感性。eval 命令使用这些限值来识别离群值。随后，离群值会以降序排列出来。

```
| inputlookup quote.csv | head 500 | eval _time=(round(strptime(time, "%Y-%m-%d %H:%M:%SZ"))) | streamstats window=100 avg("price") as avg stdev("price") as stdev | eval lowerBound=(avg-stdev*2) | eval upperBound=(avg+stdev*2) | eval isOutlier=if('price' < lowerBound OR 'price' > upperBound, 1, 0) | fields "_time", "symbol", "sourcetype", "time", "price", "lowerBound", "upperBound", "isOutlier" | sort - isOutlier
```

使用四分位差 (IQR) 识别离群值

以下示例以来自 quote.csv 文件的前 500 个事件为例。eventstats 命令用于计算中值、第 25 个百分位 (p25) 和第 75 个百分位 (p75)。eval 命令通过减去百分位从而计算出四分位差 (IQR)。中值和四分位差 (IQR) 与 eval 命令一起用于计算上限值和下限值。将 IQR 乘以 20，以此方式在计算中添加敏感性。eval 命令使用这些限值来识别离群值。随后，离群值会以降序排列出来。

```
| inputlookup quote.csv | head 500 | eval _time=(round(strptime(time, "%Y-%m-%d %H:%M:%SZ"))) | eventstats median("price") as median p25("price") as p25 p75("price") as p75 | eval IQR=(p75-p25) | eval lowerBound=(median-IQR*20) | eval upperBound=(median+IQR*20) | eval isOutlier=if('price' < lowerBound OR 'price' > upperBound, 1, 0) | fields "_time", "symbol", "sourcetype", "time", "price", "lowerBound", "upperBound", "isOutlier" | sort - isOutlier
```

删除图表中的离群值

您可以使用离群值命令来移除搜索结果中的无关数字值。

也可以选择移除或转换带离群值的事件。remove 选项用于移除这些事件。transform 选项用于截去离群值的无关值以使其达到阈值。该阈值由 param 选项指定。

如果某个值落在 param 阈值乘四分位差 (IQR) 所得值之外，则此值即被视为离群值。param 的默认值为 2.5。

创建网络服务器事件的图表，转换无关值

对于网络服务器事件的 timechart，转换无关的平均 CPU 值。

```
host=''web_server'' 404 | timechart avg(cpu_seconds) by host | outlier action=transform
```

移除影响图表中 Y 轴显示的离群值

有时当您创建一个图表时，少量值非常远离此图表的其他值以致于呈现出来后为不可见。您可以移除这些离群值，以使图表值均可见。

```
index=_internal source=*access* | timechart span=1h max(bytes) | fillnull | outlier
```

在所有交易中使用三西格玛规则以删除离群值

此示例使用 eventstats 命令来计算平均和标准偏差。然后会计算三西格玛限值。where 命令对搜索结果进行筛选。只会返回持续时间小于三西格玛限值的事件。

```
... | eval durationMins = (duration/60) | eventstats avg(durationMins) as Avrg, stdev(durationMins) as StDev | eval threeSigmaLimit = (Avrg + (StDev * 3)) | where durationMins < threeSigmaLimit
```

管理离群值告警

在设置告警时，重要的是查看您已经设置的离群值阈值。

* 如果此阈值太低，则会因非关键离群值返回过多告警
* 如果此阈值太高，则不会返回足够的告警，而且可能无法识别离群值

通常，数据中是离群值的事件比例很小。如果您每天有 1,000,000 个事件且 5% 的事件是离群值，除非指定限制，否则设置一个告警会触发 50,000 个告警操作。*限制*会抑制在给定时间范围内具有相同字段值的多余告警。

例如，您的搜索每分钟平均返回 100 个事件。您只希望在事件状态为 404 时产生告警。那么可以设置告警，使其每 60 秒执行一次告警操作，而不要在 60 秒的时间窗内每出现一个状态为 404 的事件就发出一次告警。

设置告警限制

设置限制是设置告警的一部分。

1. 确定是离群值的事件的百分比。
2. 在**设置**下方，选择**搜索、报表和告警**。
3. 在“计划和告警”下方，单击**计划此搜索的时间**复选框。屏幕会展开并显示计划和告警选项。

4.指定告警条件和模式。

5.勾选**限制**复选框，并指定限制的有限期限。

6.指定告警操作。

7.单击**保存**。

另请参阅

关于高级统计
SPL 命令：outlier、stats、top

检测异常

查找数据中的峰值

您想识别数据中的峰值。峰值可以说明在哪里出现了表示某个指标突然大幅上升或下降的巅峰值（或谷底值）。峰值的种类有很多。流量峰值、销量峰值、返回量峰值、数据库负载峰值。不管您感兴趣的峰值是哪一种，您要进行监视并可能要采取特定行动以消除这种峰值。

移动趋势线有助于峰值的查看。运行一个后跟趋势线命令的搜索，且该命令使用待创建趋势线的字段。

例如，对于 Web 访问数据，您可以为 `bytes` 字段的平均值绘制图表。

```
sourcetype=access* | timechart avg(bytes) as avg_bytes
```

若要在图表中添加另一行或栏系列对应 `bytes` 字段的前 5 个值的简单移动平均线 (sma)，使用以下命令：

```
... | trendline sma5(avg_bytes) as moving_avg_bytes
```

如果要清楚识别峰值，可能需要添加峰值的额外系列。以下搜索添加了名为 "spike" 的字段，说明字节平均数何时超过移动平均数的两倍。

```
... | eval spike=if(avg_bytes > 2 * moving_avg_bytes, 10000, 0)
```

这里的 10000 是任意值，您应选择与您的数据相关且可突显峰值的值。将 Y 轴的格式更改为对数刻度也会有所帮助。

总而言之，此搜索如下：

```
sourcetype=access* | timechart avg(bytes) as avg_bytes | trendline sma5(avg_bytes) as moving_avg_bytes | eval
spike=if(avg_bytes > 2 * moving_avg_bytes, 10000, 0)
```

此搜索使用前 5 个结果 (sma5) 的简单移动平均值。探索不同的简单移动平均值以确定用于识别峰值的最佳简单移动平均值。

趋势线命令也支持指数移动平均线 (ema) 和加权移动平均线 (wma)。

或者，您也可以绕过所有的图表绘制并将 `eval` 命令替换为 `where` 命令以对结果进行筛选。

```
... | where avg_bytes > 2 * moving_avg_bytes
```

通过查看表格视图或设置告警，您可得知 `avg_bytes` 何时出现峰值。

另请参阅

关于高级统计、eval、趋势线

检测模式

本部分介绍检测数据中的模式。有关检测异常、查找和删除离群值和时间系列预测相关主题的完整列表，请参阅本手册中的“关于高级统计”。

检测事件中的模式

群集命令是可用于检测事件中模式的强大命令。此命令可根据事件彼此之间的相似程度对事件进行分组。除非您指定其他字段，否则 `cluster` 命令基于 `_raw` 字段的内容来分组事件。

当您使用 `cluster` 命令时，会在每个事件后面附加两个新字段。

- `cluster_count` 指属于此群集的事件数量，即群集大小。
- `cluster_label` 指定事件所属的群集。例如，如果搜索返回 10 个群集，则会用 1 到 10 对这些群集进行标记。

异常可能出现在小型或大型事件组（或群集）中。小型组可能包含某用户的 1 或 2 个登录事件。大型事件可能是，比如成千上万个类似事件的 DDoS 攻击。

合理使用群集命令参数

- 使用 `labelonly=true` 参数返回所有事件。如果使用 `labelonly=false`（默认值），则每个群集只会返回一个事件。
- 使用 `showcount=true` 参数以便将 `cluster_count` 字段添加到所有事件中。如果 `showcount=false`（默认值），则事件数量不会添加到事件中。
- 阈值参数 `t` 调整群集敏感性。阈值越小，则群集数量越少。

可与群集命令一起使用的其他命令

- 在 cluster_label 列使用 dedup 命令以查看每个群集内最新加入的事件。
- 要分组事件并使结果更易于阅读，在群集列中使用 sort 命令。基于群集数排序 cluster_count 列。
  - 对于小型事件组，以升序对 `cluster_count` 列进行排序。
  - 对于大型事件组，以降序对 `cluster_count` 列进行排序。
  - 以升序对 `cluster_label` 列进行排序。群集标签为数值。以升序排序会按标签以数字顺序对事件进行组织。

返回每个群集中最新的 3 个事件

以下搜索使用了 `sales_entries.log` 文件中的 CustomerID。设置 `showcount=true` 可确保所有事件均获得 `cluster_count`。群集阈值设为 0.7。设置 `labelonly=true` 会返回传入事件。`dedup` 命令用于查看每个群集中最新的 3 个事件。结果以降序排序以分组事件。

```
source="/opt/log/ecommsv1/sales_entries.log" CustomerID | cluster showcount=true t=0.7 labelonly=true | table _time,
cluster_count, cluster_label, _raw | dedup 3 cluster_label | sort -cluster_count, cluster_label, - _time
```

如果没有设置 `labelonly=true`，则每个群集只会返回一个事件。

另请参阅

关于高级统计、dedup、群集、排序

关于时间系列预测

有几种方式可用于预测时间系列数据。例如：

- 用于确定虚拟环境的硬件要求和预测能量消耗的容量规划
- 预测收入和其他业务指标
- 加强对关键组件的监视，从而检测到系统故障，并在出现服务中断之前加以避免

您可以使用报表和仪表板在活动进行时对其进行监视，然后深入查看事件，并进行根源分析，以了解出现问题的原因。如果您所监视的事件有某种模式和相关性，则可以使用它们预测将来的活动。了解此情况后，您就可以基于特定阈值主动发出告警，并执行 "what-if" 分析，对不同的情况进行比较。

用于时间系列预测的命令

Splunk 搜索语言包括两个预测命令：predict 和 x11。

- predict 命令允许使用不同的预测算法来预测单值和多值字段将来的值。
- x11 命令（名称取自 X11 算法）删除字段中的季节波动，以呈现基本数据系列中真正的趋势。

预测算法

可以从使用 `predict` 命令的以下算法中选择：LL、LLP、LLT、LLB 和 LLP5。以上每个算法都是 Kalman 过滤器的变体。

算法选项	算法名称	描述
LL	局部等级	这是一个没有趋势和季节性的单变量模型。需要最少 2 个数据点。
LLP	季节性局部等级	这是一个具有季节性的单变量模型。时间系列的周期将自动计算。需要的最小数据点数为周期的两倍。
LLT	局部等级	这是一个具有趋势但没有季节性的单变量模型。需要最少 3 个数据点。

	趋势	
LLB	双变量局部等级	这是一个没有趋势和季节性的双变量模型。需要最少 2 个数据点。LLB 使用一组数据进行另一组的预测。例如，假定它使用数据集 Y 进行数据集 X 的预测。如果 holdback=10，则 LLB 算法将使用 Y 的最后 10 个数据点预测 X 的最后 10 个数据点。
LLP5		为其预测合并 LLT 和 LLP 模型。

有关更多信息，请参阅*《搜索参考》*中的 "predict 命令"。

用 x11 命令预测季节性

时间系列数据的季节性组件要么是加法，要么是乘法。这点反映为您可以使用 x11 命令计算的两种类型的季节性：`add()`（适用于加法）和 `mult()`（适用于乘法）。

如何确定对数据进行哪种类型的季节性调整？说明加法和乘法季节性组件之间差异的最好方法是使用示例：每年鲜花的销售额将在一年的某些日子达到高峰，例如情人节和母亲节。

在情人节期间，玫瑰的销售额可能每年增长 X 美元。此金额独立于系列的正常水平，您可以将 X 美元添加到每年情人节的预测，使此时间系列成为加法季节性调整的候选调整。在加法的季节性调整中，通过加上或减去代表绝对量（值与该季节的正常值相差的量）的数量，调整时间系列的每个值。

或者，在乘法的季节性组件中，季节性影响以百分比表示。季节性差异的绝对幅度在系列随时间增长时增加。例如，在情人节期间销售的玫瑰数量增加 40% 或者以系数 1.4 增长。在玫瑰的销量总体走弱时，情人节销售的绝对（美元）增额也会相对走弱。但是，百分比将保持不变。如果玫瑰的销量走强，则绝对（美元）增额将按比例提高。在乘法季节性调整中，将时间系列的每个值除以一个数量（该数量表示与正常销量的百分比或通常在该季节才会出现的系数）来消除此模式。

在图表上绘制时，这两种类型的季节性组件将呈现不同的特性：

- 加法季节性系列呈现稳定的季节性波动，与系列的整体水平无关。
- 乘法季节性系列呈现不同规模的季节性波动（取决于系列的整体水平）。

有关更多信息，请参阅*搜索参考*中的“x11 命令”。

另请参阅

关于高级统计、预测、x11

机器学习

本部分描述了 Splunk 机器学习工具套件。有关检测异常、查找和删除离群值、检测模式和时间系列预测相关主题的完整列表，请参阅“关于高级统计”。

Splunk 机器学习工具套件

Splunk 机器学习工具套件应用程序提供了新 SPL 命令、自定义可视化、助理和示例，可用于探索各种机器学习理念。

每个助理都包括带数据集的端到端示例，以及可将可视化和 SPL 命令应用到您个人数据中的功能。您可以查看助理面板和底层代码，了解其工作原理。有关信息，请参阅《Splunk 机器学习工具套件用户指南》。

事件分组和相关性

关于事件分组和相关性

事件相关性是在来自多个来源的数据中查找看似不相关事件间的关系，以回答此类问题：“特定系列事件发生的时间间隔为多久？”或“交易完成需要的总时间为多少？”

Splunk 软件支持使用时间和地理位置、交易、子搜索、字段查找和连接实现事件的关联性。

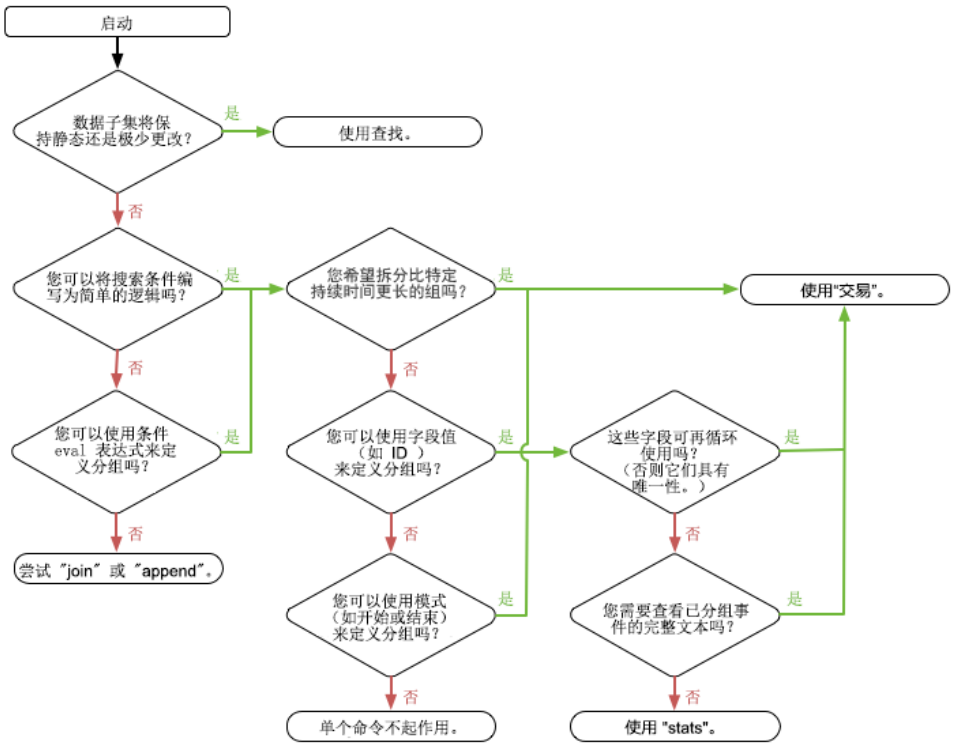
- 根据事件的时间距离或地理位置确定关系。在可能需要查看发生于给定时间段或位置的所有或任意事件子集的安全性或操作调查中，请使用此相关性。
- 跟踪一系列可能来自单独 IT 系统和数据源，以及单个交易的相关事件。确定完成交易花费的时间量，以及单个交易中的事件数量。
- 使用子搜索获取一个搜索的结果，并将其用于另一个搜索中。创建条件式搜索，使得在该搜索中仅当子搜索满足特定阈值时才可以查看搜索结果。
- 通过查看将数据与外部来源关联。
- 使用类似 SQL 的内部和外部连接，根据一个或多个通用字段将两个完全不同的数据集链接起来。

本章介绍关联或分组事件的三种方法：

- 使用时间确定事件之间的关系
- 使用子搜索关联事件
- 使用交易确定和分组相关事件

还可以使用字段查找和搜索语言的其他功能。根据搜索条件和想要定义分组的方式，可能可以使用搜索命令，如 append、associate、contingency、join 或 stats。有时没有可使用的单个命令。

如果不清楚从哪开始，下面的流程图可帮您确定使用查找、定义交易还是试用另一搜索命令来定义事件分组。



大多数情况下，使用 stats 命令或 transaction 命令可完成更多操作；相对于使用 `join` 和 `append` 命令，更建议您使用 stats 命令或 transaction 命令。有关何时使用 `stats` 和 `transaction` 的更多信息，请参阅本章节的后续主题“关于交易”。还可以参阅本手册的“计算统计信息”这一章中有关 stats 命令的更多信息。

注意：该图信息由 Nick Mealy 提供。

使用时间确定事件之间的关系

时间对于确定发生错误的内容至关重要 - 您通常知道何时发生了错误。Splunk 软件可用于确定事件中的基准模式或趋势，并将其与当前活动进行比较。

您可以通过运行一系列基于时间的搜索调查并确定异常活动，然后使用时间线深入分析特定的时间段。查看大约在同一时间发生的事件可帮助关联结果并找出问题的根源。

阅读本手册中有关如何“使用时间线调查事件”的更多信息。

关于交易

交易是任意一组一定时间跨度内概念相关的事件，例如，与单个客户在线预定某个酒店房间相关的一系列事件，或者与防火墙入侵事件相关的一系列事件。**交易类型**是配置的交易，保存为字段并与 `transaction` 命令结合使用。任意数量的数据来源可以在多个日志项内生成交易。

交易搜索

交易搜索适用于观察在多个已记录事件内执行任何物理事件的情况。使用 transaction 命令定义交易或覆盖 `transactiontypes.conf` 中指定的交易选项。

交易搜索的常见用法是，将多个事件分组为单个元事件，以表示单个物理事件。例如，**内存不足问题**可能触发记录若干数据库事件，这些事件可以全部组织在一起，形成一个交易。

要了解更多信息，请参阅本手册中的“确定事件并将其分组为交易”。

使用 stats 而不用 transaction

stats 命令和 transaction 命令在允许您根据字段值聚合单个事件方面具有相似性。

`stats` 命令表示计算按一个或多个字段分组的事件的统计信息并丢弃事件（除非正在使用 eventstats 或 streamstats）。另一方面，除第一个和最后一个事件间的持续时间以及事件计数，`transaction` 命令不计算分组事件的统计信息。此外，它将保留原始事件和原始事件的其他字段值，允许您使用更为复杂的条件对事件进行分组，例如按时间跨度或延迟限制分组，需要术语定义组的开始和结束。

`transaction` 命令在两种情况下最为有用：

1.来自一个或多个字段的唯一 ID 不足以识别两个交易的差异。即重新使用标记的情况，例如由 cookie 或客户端 IP 标记的 Web 会话。这种情况下，时间跨度或暂停也用于将数据分段到交易中。其他情况下，重新使用标记时，例如在 DHCP 日志中，特定消息将可能识别交易的开始和末尾。

2.需要查看组合事件的原始文本，而不是事件组成字段的分析。

其他情况下通常使用 `stats` 命令较好，因为该命令执行效率更高，尤其是在分布式环境中。通常事件中存在唯一 ID，而且可使用 `stats`。

例如，要计算由唯一 ID `trade_id` 标记的交易持续时间统计信息，下列搜索将产生相同答案：

```
... | transaction trade_id | chart count by duration span=log2
```

和

```
... | stats range(_time) as duration by trade_id | chart count by duration span=log2
```

但如果 `trade_id` 值被重用，而每个交易以相同文本结束，例如 "END"，则唯一的解决方案是使用以下 `transaction` 搜索：

```
... | transaction trade_id endswith=END | chart count by duration span=log2
```

另一方面，如果 `trade_id` 值被重用，但不在 10 分钟的期限内，则解决方案是使用以下 `transaction` 搜索：

```
... | transaction trade_id maxpause=10m | chart count by duration span=log2
```

有关“关于事件分组和相关性”的详细信息，请参阅本手册之前的章节。

交易和宏搜索

交易和宏搜索是一个功能非常强大的组合，可以替代交易搜索。创建交易搜索，然后使用 `$field$` 对其进行保存以允许替代。

有关如何使用宏搜索和交易的示例，请参阅*《知识管理器手册》*中的“定义和使用搜索宏”。

确定事件并将其分组为交易

您可以搜索相关事件，并将其分组为一个单个事件，此事件称为*交易*（有时称为会话）。

交易可以包括：

- 来自相同数据来源和相同主机的不同事件。
- 来自相同主机的不同数据来源的不同事件。
- 来自不同主机和不同数据来源的类似事件。

在 Splunk Web 或 CLI 中使用 `transaction` 命令搜索交易。`transaction` 命令会生成可在报表中使用的事件分组。要使用 `transaction`，请调用交易类型（通过 transactiontypes.conf 配置的类型），或者通过设置 `transaction` 命令的搜索选项定义搜索中的交易约束。

交易搜索选项

在搜索时间返回的交易包含每个事件的原始文本、共享的事件类型以及字段值。交易还包含其他数据，它们存储在以下字段中：`duration` 和 `transactiontype`。

- **`duration`** 包含交易的持续时间（交易的第一个事件与最后一个事件的时间戳之间的差异）。
- **`transactiontype`** 是交易的名称（根据交易的段落名称在 `transactiontypes.conf` 中定义）。

可以将 `transaction` 添加到任何搜索中。要获得最佳搜索性能，请精心创建搜索，然后将其发送到交易命令。有关更多信息，请参阅《搜索参考》手册中 `transaction` 命令的相关主题。

在 `transaction` 命令后加上以下选项。**注意：**有些 `transaction` 选项不能与其他选项结合使用。

`name=<transaction-name>`

- 在 `transactiontypes.conf` 中指定段落的名称。使用此选项调用已经配置供重新使用的**交易类型**。如果提供了其他参数，则这些参数会覆盖在交易规则中为相同参数指定的值。例如，如果您调用的交易规则 `web_purchase` 使用 `maxevents=10` 配置，但您想使用 `maxevents` 的其他值运行它，则将 `maxevents` 添加到具有所需值的搜索字符串：

```
sourcetype=access_* | transaction name=web_purchase maxevents=5
```

[field-list]

- 这是一个用逗号分隔的字段列表，例如 `...| transaction host,cookie`
- 如果设置此选项，则所有事件都必须具有相同的字段，才会被视为属于相同交易。
- 具有通用字段名称但值不同的事件将不分在同一组。
  - 例如，如果您添加了 `...| transaction host`，则含有 `host=mylaptop` 的搜索结果决不能与含有 `host=myserver` 的搜索结果处于相同交易之中。
  - 不具有 `host` 值的搜索结果可以与具有 `host=mylaptop` 的结果处于相同交易之中。

`match=closest`

- 指定要用于交易定义的匹配类型。
- 当前唯一受支持的值为 closest。

`maxspan=[<integer>s | m | h | d]`

- 设置一个交易的最大持续时间。
- 可以为秒、分钟、小时或天数。
  - 例如：5s、6m、12h 或 30d。
- 默认为 `maxspan=-1`，适用于“所有时间”的时间范围。

`maxpause=[<integer> s|m|h|d]`

- 指定各交易之间的最长暂停时间。
- 交易内各事件之间的暂停时间不得超过 maxpause。
- 如果为负值，将禁用 maxspause 约束。
- 默认为 `maxpause=-1`。

`startswith=<string>`

- 搜索或 eval 筛选表达式，如果某事件满足条件，则标志新交易的开始。
- 例如：
  - `startswith="login"`
  - `startswith=(username=foobar)`
  - `startswith=eval(speed_field < max_speed_field)`
  - `startswith=eval(speed_field < max_speed_field/12)`
- 默认为 `""`。

`endswith=<transam-filter-string>`

- 搜索或 eval 筛选表达式，如果某事件满足条件，则标志交易的结束。
- 例如：
  - `endswith="logout"`
  - `endswith=(username=foobar)`
  - `endswith=eval(speed_field < max_speed_field)`
  - `endswith=eval(speed_field < max_speed_field/12)`
- 默认为 `""`。

对于 `startswith` 和 `endswith`，`<transam-filter-string>` 使用以下语法进行定义：`"<search-expression>" | (<quoted-search-expression>) | eval(<eval-expression>`

- `<search-expression>` 是一个不包含引号的有效搜索表达式。
- `<quoted-search-expression>` 是一个包含引号的有效搜索表达式。
- `<eval-expression>` 是一个求值结果为布尔值的有效 eval 表达式。

示例：

- 搜索表达式：`(name="foo bar")`
- 搜索表达式：`"user=mildred"`
- 搜索表达式：`("search literal")`
- eval 布尔表达式：`eval(distance/time < max_speed)`

交易搜索示例

运行搜索，以将单个用户（或客户端 IP 地址）在时间范围内查看的所有网页分为一组。

该搜索从访问日志中获取事件，然后使用共享相同 `clientip` 值和彼此 5 分钟内发生的事件（在 3 小时时间跨度内）创建一个交易。

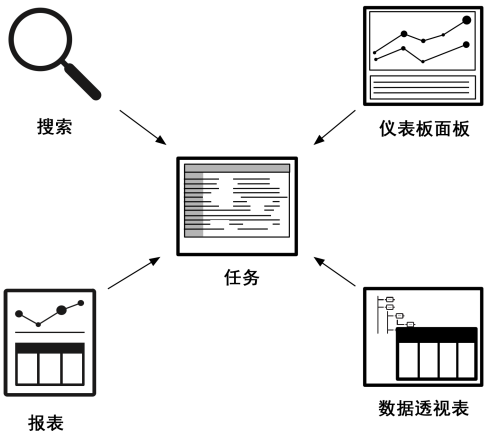
```
sourcetype=access_combined | transaction clientip maxpause=5m maxspan=3h
```

有关更多示例，请参考《搜索参考手册》中的交易命令主题。

管理任务

关于任务和任务管理

每次运行搜索、创建数据透视表、打开报表或加载仪表板面板时，Splunk 软件都会在系统中创建一个**任务**。当您运行搜索时，即是在创建一个**临时搜索**。数据透视表、报表和面板由**保存的搜索**提供数据。



任务是指跟踪临时搜索或保存的搜索的相关信息的过程。所跟踪的信息包括任务的所有者、运行该任务的应用、返回的事件数，以及任务的运行时间。

每个任务过程都会创建一个**搜索项目**。此项目包含临时搜索或保存的搜索运行时所返回的结果和关联元数据。

检查任务和管理任务

有几种方式可用于查看任务的相关信息。您可以检查任务，或者也可以管理任务。

搜索任务查看器
: 使用“搜索任务查看器”查看当前任务的相关信息，如任务执行成本和搜索任务属性。请参阅“查看搜索任务属性”。

任务管理器页面
: 使用任务管理器页面查看近期任务的相关信息。如果您具有管理员角色或带同等权限的角色，则可以管理由其他用户运行的搜索任务。请参阅“管理搜索任务”。

任务菜单

在运行搜索或在 Splunk Web 中打开报表后，您可以在不离开“搜索”页面的情况下访问和管理搜索任务的相关信息。当搜索处于运行中、暂停或完成状态时，可单击**任务**并从中选择可用选项。

- **编辑任务设置。**选择此项以打开“任务设置”对话框。您可在此对话框中更改任务的读取权限、延长任务的生存期，以及获取任务的 URL。您可使用此 URL 与其他人共享任务或在 Web 浏览器中为此任务创建一个书签。
- **将任务发送到后台运行。**如果搜索任务的完成进度很慢且您想处理其他 Splunk 活动（包括运行新的搜索任务），则可以选择此选项。该任务会继续在后台运行。
- **检查任务。**打开一个单独的窗口并通过“搜索任务查看器”显示搜索任务的信息和指标。
- **删除任务。**使用此选项可删除当前处于运行中、暂停或完成状态的任务。删除任务后，仍可将搜索保存为报表。

编辑搜索任务设置

搜索任务在运行中、暂停或完成后，可以打开“任务设置”对话框。只需单击**任务**并选择**编辑任务设置**。

共享任务

有几种方式可以将任务共享给其他 Splunk 用户。您可以更改任务权限或发送任务的链接。如果您需要其他用户也能看到任务返回的结果，此方法非常有效。请参阅“共享和导出任务”。

任务生存期

当运行新搜索时，系统会将任务保留一段时间，这段时间称为**生存期**。默认生存期为 10 分钟。生存期从任务开始运行的时间点开始计算。请参阅本手册中的“延长任务的生存期”。

管理长时间运行的任务

有时搜索任务的运行时间很长。您可能想要编辑搜索以更改搜索条件，或者想要暂停搜索或在后台运行搜索。

自动暂停长时间运行的搜索

要处理无意间启动的、长期运行的搜索任务，您可以自动暂停此任务。此功能默认情况下只能通过单击摘要仪表板来启用，可用于处理用户错误地启动了“所有时间”搜索的情况。

为特定搜索视图启用了自动暂停时，搜索视图在搜索期间将包含一个自动暂停倒计时字段。如果达到搜索时间限制，系统会显示一个信息窗口，通知用户搜索已被暂停。它会为用户提供恢复搜索或结束搜索选项。默认情况下，自动暂停前的限制是 30 秒。

当计算机进入睡眠模式或休眠模式时管理任务

当从不属于 Splunk 服务器的计算机的 Splunk Web 中运行搜索，且计算机更改为睡眠模式或休眠模式时，基本搜索过程会停止。Splunk 软件会将更改解释为睡眠模式或休眠模式，就好像运行软件的浏览器选项卡已关闭且不再可用。

要避免这个问题，请使用以下一种技术：

- 将任务发送到后台运行。计算机进入睡眠模式或休眠模式时，任务继续在后台运行。从**任务**菜单中，选择**将任务发送到后台**。
- 保存和计划搜索。搜索独立于用于创建搜索的计算机运行。您需要决定是否想要将搜索保存和计划为报表、仪表板或告警。请参阅“保存搜索和计划搜索”。
- 共享任务。任务的生存期将自动延长至 7 天，读取权限将设置为“每个人”。请参阅“共享任务和导出结果”。
- 在计算机进入睡眠模式或休眠模式之前更改计算机上的设置延长时间。

管理任务

具有管理员角色或同等权限角色的用户，可以限制给定用户可以运行的任务数，以及其任务项目可以占用的空间量。

您必须为角色定义所需的限制，并为此角色分配用户。通过为系统中每个用户指定特定角色，您可应用高粒度级别。

编辑搜索限制设置

要针对角色编辑其搜索限制设置：

1. 在 Splunk Web 中，前往**设置 > 访问控制 > 角色**。在 Splunk Enterprise 中，您可以在 `authorize.conf` 文件中手动编辑搜索限制，如**手动编辑搜索限制**中所述。

您可以通过命令行管理运行中任务。有关更多信息，请参阅"从操作系统管理搜索任务"。

手动编辑搜索限制

1. 查看*《管理员手册》*中的 authorize.conf.example 文件的内容。此示例说明了您可能会使用的部分属性。
2. 创建配置文件。

范围	描述
系统范围	在系统的本地目录中创建 `authorize.conf` 文件。系统本地目录的位置是 `$SPLUNK_HOME/etc/system/local`。
应用程序特定	在应用程序的本地目录中创建 `authorize.conf` 文件。应用程序本地目录的位置是 `$SPLUNK_HOME/etc/apps/<app_name>/local`。

3. 编辑本地 `authorize.conf` 文件。要对用户可以运行的任务进行限制，将以下信息添加到此文件中：
    1. 为要创建的角色添加一个段落。使用格式 `[role_<roleName>]`。角色名称必须是小写字符。例如，`[role_ninja]`。
    2. 可选。添加 `importRoles` 属性。导入角色也会导入此角色的其他方面，如此角色可以搜索的索引。例如，`importRoles = user`。
    3. 添加 `srchDiskQuota` 属性和值。这是此角色用户的搜索任务可以占用的最大磁盘空间量 (MB)。默认值为 100MB。例如，`srchDiskQuota = 500`。
    4. 添加 `srchJobsQuota` 属性和值。这是此角色用户可以运行的最大并发搜索数量。默认值为 3。`srchJobsQuota = 10`。
    5. 可选。添加 `rtsearch` 属性指定是否授权用户运行**实时搜索**。如果您授权用户运行实时搜索，则也应指定 `rtSrchJobsQuota` 属性。

- 有关属性描述和默认值的相关信息，请参阅*《管理员手册》*中 `authorize.conf` 文件的 role_name 段落。
- 有关角色的详细信息，请参阅*《确保 Splunk Enterprise 安全》*手册中的"添加和编辑角色"。

另请参阅

- *《报表手册》*中的"创建和编辑报表"

延长任务的生存期

当运行一个新搜索任务时，系统会保留此任务一段时间，这段时间称为**生存期**。在生存期内，您可以访问该任务并查看任务返回的数据。如果在指定生存期内都没有用户访问此任务，则该任务会过期并从系统中删除。

生存期设置有两种：10 分钟和 7 天。生存期从任务开始运行的时间点开始计算。

任务的默认生存期

搜索任务的默认生存期取决于此搜索任务是**计划的搜索**还是非计划搜索项目。

例如，基于内联搜索的仪表板面板使用非计划的搜索。基于报表的面板使用保存的搜索。保存的搜索可以是计划的，也可以是非计划的。

非计划搜索的默认生存期

当运行一个**临时搜索**且此搜索会自行完成或结束时，生成的搜索任务的默认生存期为 10 分钟。其他知识对象，如实时告警和基于使用非计划搜索的内联搜索的面板，具有相同的默认生存期。

计划搜索的默认生存期

计划搜索会按固定间隔启动搜索任务。默认情况下，这些任务的保留时间为计划的搜索的间隔时间乘以二。例如，如果搜索每 6 小时运行一次，则生成的任务将在 12 小时后到期。

生存期自动延长

当您访问一个活跃任务时，例如查看搜索任务的结果，则其生存期会重置。无论任务生存期是 10 分钟，还是 7

天，都会进行重置。以下几个示例对此进行了介绍。

- 如果生存期设置为 10 分钟且您从上午 11 点开始运行此搜索任务，则此任务的生存期会设置为上午 11:10 终止。如果您在上午 11:07 再次运行此任务，则此任务的生存期会重置为上午 11:17 终止。
- 如果您将某个任务的生存期设置为 7 天，然后在 4 天后访问此任务，则此任务的生存期会重置为从当前日期和时间开始的 7 天后再到期。

更改当前任务的生存期

在搜索应用中可以更改当前临时搜索任务的生存期设置。

1. 选择**任务**下拉列表。
2. 选择**编辑任务设置**以显示“任务设置”。
3. 将**生存期**设置为 10 分钟或 7 天。

更改活跃任务的生存期

对于之前运行的计划搜索或非计划搜索所生成的任务，也可以更改其生存期。但只能更改活跃搜索任务的生存期。请参阅“管理搜索任务”。

过期的任务

当任务生存期结束后，此任务即会过期并从系统中删除。

有可能在您查看任务列表时某任务将要过期。当您尝试延长过期任务的生存期时，将出现一条消息，说明该任务不复存在。您无法延长过期任务的生存期。

更改默认生存期

您可以更改非计划搜索和计划搜索所生成任务的默认生存期。

更改非计划搜索的默认生存期

在 Splunk Enterprise 中，可以更改非计划搜索所生成任务的默认生存期。

前提条件

- 只有具有文件系统访问权限的用户，如系统管理员才能更改默认生存期值。
- 请参阅*《管理员手册》*中的“如何编辑配置文件”了解具体步骤。

不要更改或复制default目录中的配置文件。default目录中的文件必须保持原样并位于其原始位置。在local目录进行更改。

1. 打开搜索应用的本地 `limits.conf` 文件。例如，`$SPLUNK_HOME/etc/apps/<app_name>/local`。
2. 在 `[search]` 段落中，将 `default_save_ttl` 值更改为所需的数字。TTL 是 "time to live" 的缩写，表示持续时间。

如果您使用的是 Splunk Cloud 并想更改非计划搜索所生成任务的默认生存期，请向 Splunk 支持提交问题。

更改计划搜索的默认生存期

在 Splunk Enterprise 中，您可以更改特定计划的搜索所生成任务的默认生存期。

前提条件

- 只有具有文件系统访问权限的用户，如系统管理员才能更改默认生存期值。
- 请参阅*《管理员手册》*中的“如何编辑配置文件”了解具体步骤。

不要更改或复制default目录中的配置文件。default目录中的文件必须保持原样并位于其原始位置。在local目录进行更改。

1. 打开本地 `savedsearches.conf` 文件。例如，`$SPLUNK_HOME/etc/apps/<app_name>/local`。
2. 找到计划的搜索，并将 `dispatch.ttl` 设置更改为不同的间隔倍数。

如果您使用的是 Splunk Cloud 并想更改计划搜索所生成任务的默认生存期，请向 Splunk 支持提交问题。

共享任务和导出结果

您可以将任务共享给其他 Splunk 用户，或将事件数据导出进行归档或在第三方图表应用程序中使用。

与他人共享任务

共享任务时，您共享的是搜索某次运行时所生成的结果。

有几种方式可以将特定任务共享给其他 Splunk 用户。您可以更改搜索任务的权限以将此任务共享给其他用户。也可以将搜索任务的 URL 发送给其他 Splunk 用户，以此方式共享此任务。

您只可以更改当前任务的权限或发送当前任务的链接。

更改任务权限

您可以通过更改任务权限的方式共享任务。所有任务都默认为**专用**。

1. 从**任务**菜单中，选择**编辑任务设置**以显示“任务设置”对话框。
2. 将**读取权限**的设置更改为**每个人**。
3. 单击**保存**

共享任务 URL

通过向其他 Splunk 用户发送任务的链接，可以将任务共享给他们。如果您需要其他用户也能看到任务返回的结果，此方法非常有效。

当您要向用户发送链接时，该用户必须具有使用任务所属应用程序的权限。

确定要用哪种方式获取任务链接。可以使用**共享**图标或**任务**菜单。

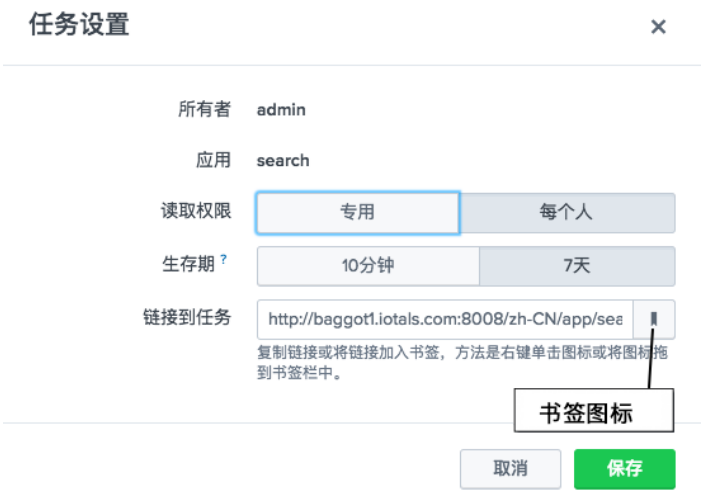
1. 要使用“共享”图标：

   1. 单击此图标。 “共享”图标是其中一种搜索操作图标。
   2. 在**链接到任务**文本框中，复制 URL 并将此链接发送给要共享任务结果的用户。

      此任务的权限自动更改为**每个人**，且其生存期自动延长为 **7 天**。

- 要使用**任务**菜单：

1. 从**任务**菜单中，选择**编辑任务设置**以显示“任务设置”对话框。
2. 将**读取权限**的设置更改为**每个人**。如果任务的权限设置为**专用**，其他用户则无法通过链接访问此任务。
3. 将**生存期**设置更改为 **7 天**。
4. 复制任务的链接并将此链接发送给要共享任务结果的用户。



您也可以使用书签图标将保存此链接以供个人使用。书签图标显示在“任务设置”对话框和“共享作业”对话框中。您可以单击书签图标并将其拖到网页浏览器的书签栏。

将任务结果导出为文件

任务结果可以以多种格式导出，如 CSV、JSON、PDF、原始事件和 XML。导出后即可将此文件存档，或在第三方图表应用程序中使用。格式选项取决于所操作任务项目的类型。

- 如果搜索会生成计算的数据且此数据会显示在“统计”选项卡中，则不可使用原始事件格式导出。
- 如果搜索是保存的搜索，如报表，则可用 PDF 格式导出。

导出文件保存在浏览器或操作系统的默认下载目录中。

您可使用几种方式来导出搜索结果，其中包括 Splunk Web、CLI、SDK 和 REST。有些方式的速度较快，而另一些则适用于非常大的事件集。

有关导出方式的完整列表和特定步骤的链接，请参阅“导出搜索结果”。

管理搜索任务

可以使用“任务”页面来查看和管理您所拥有的任何**任务**。

如果您具有管理角色或同等权限的角色，则可以管理您的 Splunk 实现中所有用户所运行的搜索任务。

打开“任务”页面

- 在 Splunk Web 中，选择**活动 > 任务**以查看您的任务列表。这将打开**任务**页面。

“任务”页面显示了不同类型搜索任务的列表。

- 由最近手动运行的临时搜索或数据透视表所生成的任务。
- 在加载仪表板或打开报表时所运行的搜索的任务。
- 计划搜索的任务。

刷新任务列表

“任务”页面的任务列表不会自动刷新。

- 在您打开“任务”页面之后创建的任务不会自动显示，除非您重新加载“任务”页面。
- 如果在您打开“任务”页面时有任务过期，此任务仍会显示在“任务”页面列表中，但无法查看任务结果。

重新加载即可刷新“任务”页面。

任务操作

可以使用“操作”列执行任务操作。

使用**任务**下拉列表编辑任务设置、延长任务生存期、检查任务或删除任务。

使用操作图标暂停、停止、共享或导出任务。

要针对多个任务执行这些操作，选择这些任务并单击**编辑已选项**。然后选择要执行的操作。

查看和比较任务

可以查看您最近派遣的或保存以备后续查看的任务列表，并用它来比较任务统计信息，如运行时间、匹配的事件总数和大小等。

活跃任务计数

在“任务”页面的右上角有一个计数，表示列表中所列任务的总数。

此计数代表您打开“任务”页面时的任务数。如果有任务在“任务”页面打开后过期，此任务计数不会自动刷新。

排序任务列表

默认情况下，任务列表按**创建时间**列进行排序。

您可以按任意列在列标题处所显示的排序按钮，对列表进行排序。例如，可以按任务过期时间或按任务所有者对列表进行排序。

- 单击列标题一次即可以升序对列表进行排序。再单击一次即可以降序对列表进行排序。

筛选任务列表

任务列表可以按应用程序、所有者和状态进行筛选。

- 在**筛选**框中，键入搜索条件中出现的术语或表达式来筛选列表。

例如，可以在“筛选”框中指定 `diskUsage`、`EMBED AND diskUsage=8*` 或 `label=EMBED AND diskUsage=8*`。

查看任务搜索结果

可以查看在“任务”页面列出的搜索结果。

1. 单击搜索链接可查看与特定任务相关的结果。
   - 对于临时搜索，此链接显示的是搜索条件。
   - 对于保存的搜索，此链接显示的是报表、仪表板面板或数据透视表的名称。

   结果会显示在搜索应用视图中。

查看执行中任务的进度

可以查看计划搜索、实时搜索和长期运行的历史搜索所派遣的任务。

使用**状态**列查看执行中任务的进度。**状态**列显示的是已处理事件的百分比。当前任务的状态为**运行中**。正在后台运行的任务的状态为**后台运行**。

更改每页显示的任务计数

每页列表中所显示的任务数可以更改。默认是每页显示 10 个任务。在窗口的右侧，您可以设置每页显示 10 个、20 个或 50 个任务。

检查任务

通过检查任务可以进一步了解搜索在做什么，以及 Splunk 软件把大部分时间花在什么地方。

使用“搜索任务查看器”查看当前任务的相关信息，如任务执行成本和搜索任务属性。

1. 在特定任务的**操作**列中，选择**任务**。
2. 选择**检查任务**。

有关“搜索任务查看器”使用的更多信息，请参阅“查看搜索任务属性”。

延长搜索任务的生存期

从“任务”页面中，有几种方式可以更改任务的生存期。要了解特定任务类型生存期的详细信息，请参阅“延长任务生存期”。

快速延长任务的生存期

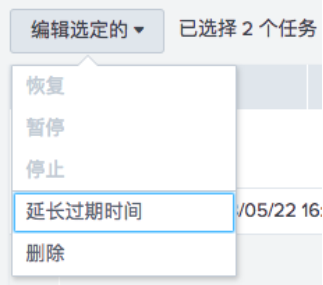
您可以快速延长任务的生存期。

1. 在特定任务的**操作**列中，选择**任务**。
2. 选择**延长任务过期时间**。

延长多个任务的生存期

您可以同时延长多个任务的生存期。

1. 选择要延长生存期的所有任务。
2. 在任务列表上方，单击**编辑已选项**。
3. 选择**延长过期时间**。



选择生存期要延长的时间段

用此方式选择延长的时间段。

1. 在特定任务的**操作**列中，选择**任务**。
2. 选择**编辑任务设置**。
3. 选择 10 分钟或 7 天。
4. 单击**保存**。

共享任务

您可以更改任务权限或共享任务的 URL，以此方式共享任务。

更改任务权限

您可以通过更改任务权限的方式共享任务。默认情况下，所有任务都设置为“专用”。

1. 从**任务**菜单中，选择**编辑任务设置**以显示“任务设置”对话框。
2. 将**读取权限**的设置更改为**每个人**。
3. 单击**保存**。

共享任务 URL

通过向其他 Splunk 用户发送任务的链接，可以将任务共享给他们。

您向其发送链接的用户也必须具有使用任务所属应用程序的权限。

1. 在**操作**列中，单击待共享任务的“共享”图标。
2. 在“共享任务”对话框中，复制**链接到任务**中的 URL。
3. 将链接发送给要共享任务结果的用户。

此任务的权限更改为**每个人**，且其生存期延长为**7 天**。

通过**任务设置**窗口也可以获取此 URL。如果您用此方式共享任务，则必须将**读取权限**更改为**每个人**，且**生存期**更改为 **7 天**。

导出任务

任务的事件数据可以以多种格式导出，如 CSV、JSON、PDF、原始事件和 XML。导出后即可将此文件存档，或在第三方图表应用程序中使用。格式选项取决于所操作任务项目的类型。

1. 在**操作**列中，单击待导出事件数据的“导出”图标。
2. 单击**格式**并选择导出搜索结果时所需的格式。
3. 在**文件名**字段中，键入待保存事件数据的导出文件的名称。
4. 在**结果数**字段中，指定要导出的结果数量。
5. 单击**导出**将任务事件保存到导出文件中。

导出文件保存在浏览器或操作系统的默认下载目录中。

注意：如果搜索返回大量结果，则可以通过搜索应用访问所有结果。但是，完整的结果集可能并不会保存在**搜索任务项目**中。当导出搜索结果时，导出过程是基于搜索任务项目，而非搜索应用中的结果。如果项目不包含完整的结果集，“导出结果”对话框的底部会显示一条消息，告诉您 Splunk 软件会在结果导出前重新运行此搜索。

有关更多信息，请参阅“导出搜索结果”。

如果“导出”图标不可见，则是由系统管理员隐藏了起来以防止导出数据。

当导出大量数据时扩展会话超时

若使用“导出”按钮尝试导出大量数据时，此会话可能会超时。

具有管理员角色或同等权限角色的用户可以采用以下步骤来延长会话超时限制。

1. 单击**设置 > 服务器设置 > 常规设置**。
2. 在 **Splunk Web** 部分，增加“会话超时”字段中的数字。
3. 单击**保存**。

增加超时设置能够使 Splunk Web 有更多的时间将浏览器和 Splunk Web 之间的连接建立起来。

删除任务

您可以从任务列表中删除一个或多个任务。

删除单个任务

您可以从任务列表中删除一个任务。

1. 在特定任务的**操作**列中，选择**任务**。
2. 选择**删除任务**。

删除多个任务

您可以一次删除多个任务。

1. 选择要删除的所有任务。
2. 在任务列表上方，单击**编辑已选项**。
3. 选择**删除**。

查看搜索任务属性

搜索任务查看器工具可让您进一步了解搜索任务，并了解 Splunk 软件大部分时间都花在什么地方。

本主题介绍如何使用“搜索任务查看器”进行搜索**任务**性能问题的故障排除，并了解搜索中的事件类型、标记和查找等知识对象的行为。有关更多信息，请参阅本手册中的“管理搜索任务”。

打开“搜索任务查看器”

只要搜索任务尚未过期（即搜索项目仍存在），您就可以通过“搜索任务查看器”查看搜索任务。访问“搜索任务查看器”时搜索并非一定要处于运行中。

1. 运行该搜索。

2. 从**任务**菜单中，选择**检查任务**。

“搜索任务查看器”在一个单独的窗口中打开。

查看搜索任务的属性

如果您有任务项目的搜索 ID (SID)，您可以使用 URL 来检查搜索任务项目。您可以在“任务管理器”中查找搜索的 SID（单击右上角的**任务**链接），也可以在 Splunk 的 dispatch 目录 `$SPLUNK_HOME/var/run/splunk/dispatch` 中找到。有关“任务管理器”的更多信息，请参阅本手册中的“管理搜索任务”。

如果您查看“搜索任务查看器”窗口的 URI 路径，您将会在字符串的末尾看到类似如下所示的内容：

```
.../manager/search/job_inspector?sid=1299600721.22
```

`sid` 是 SID 号。`namespace` 是 SID 所关联的应用名称。在本例中，SID 是 1299600721.22。

在 URI 路径的 `sid=` 后面键入搜索项目的 SID，然后按 **Enter**。只要您具有查看搜索所需的所有权权限，您就能够检查该搜索。

现在，您看到了什么？

“搜索任务查看器”显示的内容

“搜索任务查看器”窗口的顶部会显示一条信息消息。此消息取决于任务是处于暂停、运行中还是已完成状态。例如，如果任务已完成，则此消息会告诉您它找到的结果数量及完成搜索所花费的时间。如果有错误消息，则也会显示在窗口的顶部。

“搜索任务查看器”显示的关键信息是执行成本和搜索任务属性。

执行成本

执行成本部分列出了有关搜索组件以及每个组件对整体搜索性能的影响程度的信息。

搜索任务属性
: 搜索任务属性部分列出了任务的其他特性。

执行成本

执行成本部分给出的信息可用于搜索效率问题的故障排除。您可以缩小影响搜索性能的处理组件范围。此部分包含用于处理搜索的搜索处理组件的相关信息。

- 组件持续时间（以秒为单位）。
- 每个组件在搜索运行期间被调用的次数。
- 每个组件的输入和输出事件数量。

“搜索任务查看器”按字母顺序列出组件。可见的组件数量取决于所运行的搜索。

以下表格描述了典型关键字搜索中每个搜索命令和分布式搜索组件的重要意义。

搜索命令的执行成本

通常，对于属于搜索任务的每个命令，有一个参数 `command.<command_name>`。这些参数的值表示处理每个 `<command_name>` 所用的时间。例如，如果使用表格命令，您将看到 `command.table`。

搜索命令组件名称	描述
`command.search`	Splunk 软件识别包含匹配搜索的索引字段的事件后，会分析事件以识别哪些事件符合其他搜索条件。这些是并发操作，不是连续操作。 • `command.search.index` - 指示其在 TSIDX 文件中查找要在原始数据中读取的位置需要的时间。这是从基本搜索标记中识别要检索的事件需要的时间。 • `command.search.rawdata` - 指示其从原始数据文件读取实际事件需要的时间。 • `command.search.typer` - 指示其为事件分配事件类型需要的时间。 • `command.search.kv` - 指示其对事件应用字段提取需要的时间。 • `command.search.fieldalias` - 指示其根据 `props.conf` 重命名字段需要的时间。 • `command.search.lookups` - 指示其根据现有字段创建新字段（执行字段查找）需要的时间。 • `command.search.filter` - 指示其筛选出不匹配事件（例如，字段和短语）需要的时间。 • `command.search.tags` - 指示其为事件分配标记需要的时间。

使用的命令类型与您要查看的“调用”、“输入计数”和“输出计数”数字之间存在一定的关系。对于生成事件的搜索，您希望输入计数为 0 且输出计数为某一数量的事件 (X)。如果搜索既是一个生成搜索，又是一个筛选搜索，筛选搜索将具有输入（等于生成搜索的输出，即 X），且 output=X。因而总计数为 input=X，output=2*X，调用计数也会加倍。

已派遣搜索的执行成本

分布式搜索组件名称	描述
`dispatch.check_disk_usage`	检查此任务的磁盘使用情况所用的时间。
`dispatch.createdSearchResultInfrastructure`	为每个对等节点创建和设置收集器的时间，以及对每个对等节点执行 HTTP POST 的时间。
`dispatch.earliest_time`	指定搜索的最早时间。可以是相对时间或绝对时间。默认是空字符串。
`dispatch.emit_prereport_files`	当运行**转换搜索**时，Splunk Enterprise 直至搜索完成，才能计算报表的统计结果。在它从搜索对等节点获取事件之后 (`dispatch.fetch`)，它会将结果写入本地文件。`dispatch.emit_prereport_files` 提供 Splunk Enterprise 将转换搜索结果写入这些本地文件所需要的时间。
`dispatch.evaluate`	分析搜索，并设置运行搜索所需的数据结构所用的时间。该组件还包括评估和运行子搜索所用的时间。这是对使用的每个搜索命令进一步拆分而来的一个组件。通常，`dispatch.evaluate.<command_name>` 会指出分析和评估 `<command_name>` 参数所用的时间。例如，`dispatch.evaluate.search` 指示评估和分析 `search` 命令参数所用的时间。
	搜索头等待或从搜索节点获取事件需要的时间。`dispatch.fetch` 值与

dispatch.fetch	command.search 值不同。command.search 值包括所有索引器所花的时间，该时间可大于实际搜索时间。如果您只有一个节点，则 dispatch.fetch 和 command.search 值类似。在分布式环境中，搜索不同，这些值也会有很大不同。
dispatch.preview	生成预览结果需要的时间。
dispatch.process_remote_timeline	对搜索节点生成的时间线信息进行解码需要的时间。
dispatch.reduce	减少中间报表输出所用的时间。
dispatch.stream.local	搜索头在搜索的流部分花费的时间。
dispatch.stream.remote	在跨所有节点聚合的、分布式搜索环境中执行远程搜索需要的时间。另外，通过下列项目指示在每个远程搜索节点上执行远程搜索需要的时间：dispatch.stream.remote.<search_peer_name>。output_count 在此情况下表示发送的字节，而不是事件。
dispatch.timeline	生成时间线和字段边栏信息需要的时间。
dispatch.writeStatus	定期更新任务 dispatch 目录中的 status.csv 和 info.csv 所用的时间。
startup.handoff	从单独搜索过程的分叉到分叉的搜索过程的有用工作开始之间经过的时间。换句话说，它是建立搜索设备所需要的近似时间。这是所有涉及的对等节点之间的累计值。如果这需要很长时间，则可以作为带有 .conf 文件或 dispatch 目录的 I/O 问题的指示。

搜索任务属性

搜索任务属性字段提供有关搜索任务的信息。**搜索任务属性**字段按字母顺序列出。

参数名称	描述
cursorTime	最早时间，之后将不扫描任何事件。可用于指示进度。请参阅 doneProgress 的描述。
delegate	用于保存的搜索，指定用户已启动的任务。默认为 scheduler。
diskUsage	所使用的磁盘空间总量（以字节为单位）。
dispatchState	搜索的状态。可以为 QUEUED、PARSING、RUNNING、PAUSED、FINALIZING、FAILED 或 DONE 中的任何一个。
doneProgress	介于 0 和 1.0 之间的数字，指示近似的搜索进度。 doneProgress = (latestTime - cursorTime) / (latestTime - earliestTime)
dropCount	仅限于实时搜索，可能因 rt_queue_size（默认为 100,000）而丢弃的事件的数量。
earliestTime	搜索任务配置为启动的最早时间。可用于指示进度。请参阅 doneProgress 的描述。
eai:acl	描述应用程序和用户级权限。例如，应用是否全局共享？哪些用户可以运行或查看搜索？
eventAvailableCount	可用于导出的事件的数量。
eventCount	搜索返回的事件的数量。换言之，这是与搜索术语实际匹配的已扫描事件（由 scanCount 表示）的子集。
eventFieldCount	在搜索结果中发现的字段的数量。
eventIsStreaming	指示该搜索的事件是否正在进行流处理。
eventIsTruncated	指示搜索的事件是否尚未存储，从而导致无法从事件端点获得搜索。
eventSearch	在任何转换命令之前的整个搜索的子集。时间线和事件端点表示此搜索部分的结果。
eventSorting	指示该搜索的事件是否已排序，以及按何种顺序排序。asc = 升序；desc = 降序；none = 未排序
isBatchMode	指示搜索是否在批处理模式下运行。这仅适用于包括转换命令的搜索。
isDone	指示搜索是否已完成。
isFailed	指示执行搜索时是否出现了致命错误。例如，搜索字符串具有无效的语法。

isFinalized	指示搜索是否已完成（在完成之前停止）。
isPaused	指示搜索是否已暂停。
isPreviewEnabled	指示预览是否已启用。
isRealTimeSearch	指示搜索是否为实时搜索。
isRemoteTimeline	指示是否启用了远程时间线功能。
isSaved	指示搜索任务已保存，搜索项目从最后一次查看或改动任务起在磁盘上保存 7 天。添加或编辑 default_save_ttl 值（位于 limits.conf 中）以覆盖默认值 7 天。
isSavedSearch	指示这是否是一个使用计划程序运行的保存的搜索。
isTimeCursored	指定 cursorTime 是否可信任。如果第一个命令是搜索，则此参数一般设置为 true。
isZombie	指示运行搜索的进程是否处于不可用状态，但搜索尚未完成。
keywords	该搜索使用的所有正面关键字。正面关键字是指不存在于 NOT 子句中的关键字。
label	为该搜索创建的自定义名称。
latestTime	搜索任务配置为启动的最晚时间。可用于指示进度。请参阅 doneProgress 的描述。
numPreviews	目前已为该搜索任务生成的预览的数量。
messages	错误和调试消息。
optimizedSearch	已运行搜索的重构语法。内置优化程序会分析您的搜索并重构搜索语法（如可行），以提高搜索性能。您运行的搜索在**搜索**任务属性中显示。
performance	这是**执行成本**的另一种表示形式。
remoteSearch	发送到每个搜索节点的搜索字符串。
reportSearch	如使用报告命令，则为报告进行搜索。
request	搜索向 splunkd 发送的 GET 参数。
resultCount	搜索返回的结果总数。
resultIsStreaming	指示搜索的最终结果是否可使用流获得（例如，无转换操作）。
resultPreviewCount	最新预览结果中的结果行数量。
runDuration	完成搜索需要的时间（以秒为单位）。
scanCount	已扫描或从磁盘读取的事件的数量。
search	搜索字符串。
searchCanBeEventType	如果搜索可以保存为一个事件类型，此值为 1；否则为 0。 只有基本搜索（无子搜索或管道）可保存为事件类型。
searchProviders	已联系到的所有搜索节点的列表。
sid	搜索的 ID 号。
statusBuckets	最大时间线数据桶数量。
ttl	存活时间，或在搜索任务完成之后到搜索任务过期之前的一段时间。
其他信息	指向关于搜索的进一步信息的链接。这些链接可能不是始终可用。 • 时间线 • 字段摘要 • search.log

注意：进行搜索性能问题的故障排除时，了解 scanCount 和 resultCount 成本之间的差异是很重要的。对于密集搜索，scanCount 和 resultCount 是类似的 (scanCount = resultCount)；而对于稀疏搜索，scanCount 则比结果计数大得多 (scanCount >> resultCount)。使用 resultCount/time 速率衡量的搜索性能应该不会太高，但使用 scanCount/time 时搜索性能应该会很高。通常，scanCount/second 事件速率应该停留在 10k 和 20k 个事件/秒之间，性能才会被视为良好。

调试消息

配置“搜索任务查看器”，当您的搜索中出现错误时，可以显示调试消息。例如，当您的结果中有字段缺失时，调试消息可以向您发出警告。

在搜索完成之后，“搜索任务查看器”将调试消息显示在“搜索任务查看器”窗口的顶部。

默认情况下，“搜索任务查看器”会隐藏调试消息。要配置显示它们，打开 `limits.conf` 并将 `infocsv_log_level` 参数（位于 `[search_info]` 段落中）设置为 `INFO`。

```
[search_info]
infocsv_log_level = INFO
```

搜索任务查看器输出示例

下面是 dedup 搜索在所有时间内运行的执行成本的示例：

```
* | dedup punct
```

执行成本面板的搜索命令组件可能如下所示：

执行成本

持续时间 (秒)		Component	调用	输入计数	输出计数
	0.417	command.dedup	110	33,945	10,341
	0.109	command.fields	109	1,189,258	1,189,258
	10.683	command.prededup	109	1,189,258	33,945
	92.205	command.search	109	-	1,189,258
	0.935	command.search.index	104	-	-
	0.135	command.search.fieldalias	102	1,189,258	1,189,258
	0.113	command.search.calcfields	102	1,189,258	1,189,258
	0	command.search.index.usec_1_8	4	-	-
	0	command.search.index.usec_512_4096	1	-	-
	0	command.search.index.usec_64_512	99	-	-
	0	command.search.index.usec_8_64	90	-	-
	36.721	command.search.typer	109	1,189,258	1,189,258
	27.841	command.search.kv	102	-	-
	17.764	command.search.rawdata	102	-	-
	4.226	command.search.lookups	102	1,189,258	1,189,258

command.search 组件及其下的所有内容向您提供搜索的 `search` 命令部分的性能影响，即管道符前面的所有内容。

`command.prededup` 提供在将 `search` 命令的结果传递给 `dedup` 命令之前对这些结果进行处理的性能影响。

- `command.prededup` 的输入计数和 `command.search` 的输出计数匹配。
- `command.dedup` 的输入计数和 `command.prededup` 的输出计数匹配。

在此情况下，`command.prededup` 的“输出”计数应与搜索完成时返回的事件数量匹配。此为**搜索任务属性**下 `resultCount` 的值。

问答

有什么问题吗？请访问 Splunk Answers，查看 Splunk 社区有哪些与使用“搜索任务查看器”相关的问题和答案。

Dispatch 目录和搜索项目

您所运行的每个搜索或告警都会创建一个必须保存到磁盘上的**搜索项目**。项目所存储的目录位于 `dispatch` 目录下。每个**搜索任务**都有一个搜索特定目录。任务到期时，会删除搜索特定目录。

有关搜索任务的信息，请参阅“关于任务和任务管理”。

Dispatch 目录位置

`dispatch` 目录储存了搜索运行的节点中的项目。这些节点包括搜索头、搜索节点和独立 Splunk Enterprise 实例。`dispatch` 目录的路径是 `$SPLUNK_HOME/var/run/splunk/dispatch`。

Dispatch 目录内容

在 `dispatch` 目录中，会针对每个搜索或告警创建一个搜索特定目录。每个搜索特定目录中有一个包含搜索结果的

CSV 文件、一个包含搜索执行详细信息的 `search.log` 和其他文件。这些是 0 字节文件。

查看 Dispatch 目录内容

您可以通过命令行窗口或 UI 窗口（如 Windows Explorer 或查找器）列出搜索特定目录。

例如，要查看命令行窗口中的列表，更改为 Dispatch 目录然后列出该目录中的内容。以下列表包含临时、实时以及计划的搜索特定目录。

```
# cd $SPLUNK_HOME/var/run/splunk/dispatch
# ls
1346978195.13
rt_scheduler__admin__search__RMD51cfb077d0798f99a_at_1469464020_37.0
1469483311.272
1469483309.269
1469483310.27
scheduler__nobody_c3BsdW5rX2FyY2hpdmVy__RMD5473cbac83d6c9db7_at_1469503020_53
admin__admin__search__count_1347454406.2
rt_1347456938.31
1347457148.46
subsearch_1347457148.46_1347457148.1
```

搜索特定目录内容包括文件和子目录。以下示例显示了名为 `1346978195.13` 的搜索特定目录的内容。

```
#ls 1346978195.13/
args.txt
audited
buckets/
events/
generate_preview
info.csv
metadata.csv
peers.csv
request.csv
results.csv.gz
runtime.csv
search.log
status.csv
timeline.csv
```

窗口用户应使用 **dir** 替代命令行窗口或者 Windows Explorer 中的 **ls**，以查看 `dispatch` 目录内容。

搜索特定目录文件描述

出现在搜索特定目录中的文件或子目录取决于您运行的搜索类型。下表列出了可能出现在搜索特定目录中的文件和子目录。

文件名称	内容
args.txt	传递给搜索过程的参数。
alive.token	搜索过程的状态。指定搜索是活动还是非活动状态。
audited	表示事件已通过审计签名的标记。
buckets	包含每个数据桶的字段选取器统计信息的子目录。数据桶是特别在搜索直方图 UI 中可见的数据块。与索引数据桶无关。
custom_prop.csv	包含自定义任务属性，即可添加到搜索任务的任意键值，可后续检索，并通常由 UI 显示目标使用。
events	包含用于生成搜索结果的事件的子目录。
generate_preview	表示此搜索已请求预览的标记。该文件主要用于 Splunk Web 搜索。
info.csv	包括最早时间、最晚时间和结果计数详细的搜索信息列表。
metadata.csv	搜索所有者和角色。
peers.csv	要求运行此搜索的节点列表。

pipeline_sets	索引器所运行的管道集数量。默认值为 1。
remote_events 或 events_num_num.csv.gz	用于 remote-timeline 优化，这样就可以对使用 `status_buckets>0` 运行的报告搜索进行 MapReduce。
request.csv	来自请求的搜索参数列表，包括搜索的字段和文本。
results.csv.gz	包括搜索结果的归档。
rtwindow.czv.gz	当窗口中的事件多于内存可存放的数量时，最近实时窗口的事件。默认限制是 50K。
runtime.csv	暂停并取消设置。
search.log	搜索过程的日志。
sort...	排序临时文件，通过 `sort` 命令用于大型搜索。
srtmpfile...	一般搜索临时文件，由未给临时文件命名的实用工具使用。
status.csv	搜索的当前状态，如是否仍在运行中。搜索状态可以是以下各项中的任一项：QUEUED、PARSING、RUNNING、PAUSED、FINALIZING、FAILED 和 DONE。
timeline.csv	每个时间线数据桶的事件计数。

Dispatch 目录也包含临时数据模型加速摘要。这些与持续数据模型加速摘要不同，是存放在索引级别的。

Dispatch 目录命名约定

根据搜索类型命名 Dispatch 目录中的搜索特定目录名称。对于已保存的搜索和计划搜索，搜索特定目录名称由以下条件决定。

- 搜索名称少于 20 个字符且仅包含 ASCII 字母数字字符。搜索特定目录名称包括搜索名称。
- 搜索名称等于或多于 20 个字符且包含非字母数字字符。名称会使用哈希值。这是为了确保可在文件系统中创建由搜索 ID 命名的搜索特定目录。

包含多个子搜索的搜索可能超出了分发目录名称的最大长度。如果超出了最大长度，则搜索失败。

搜索类型	命名约定	示例
本地临时搜索	搜索的 UNIX 时间。	临时搜索。`1347457078.35` 实时临时搜索。`rt_1347456938.31` 使用子搜索的临时搜索，该子搜索会创建两个分发目录。`1347457148.46` `subsearch_1347457148.46_1347457148.1`

保存的搜索	请求搜索的用户、运行搜索所用的用户上下文身份、搜索来自的应用、搜索字符串、UNIX 时间。	“count” - 由**管理员**运行，在用户上下文**管理员**中，保存在应用**搜索**中
		`admin__admin__search__count_1347454406.2`
		“过去 24 小时内的错误” - 由**某人**运行，在用户上下文**某人**中，保存在应用**搜索**中
		`somebody__somebody__search_RMD5473cbac83d6c9db7_1347455134.20`

计划的搜索	请求搜索的用户、运行搜索所用的用户上下文身份、搜索来自的应用、搜索字符串、UNIX 时间以及添加在最后以防名称冲突的内部 ID。	“foo” - 由**计划程序**运行，无用户上下文，保存在应用 **unix** 中
		`scheduler__nobody__unix__foo_at_1347457380_051d958b8354c580`
		“foo2” - 搜索头 **sh01** 中的远程节点搜索，有**管理员**用户上下文，由**计划程序**运行，保存在应用**搜索**中
		`remote_sh01_scheduler__admin__search__foo2_at_1347457920_79152a9a8bf33e5e`

远程搜索	来自远程节点的搜索以单词“远程”开头。	“foo2” - 搜索头 **sh01** 中的远程节点搜索 `remote_sh01_scheduler__admin__search__foo2_at_1347457920_79152a9a8bf33e5e`

实时搜索	实时搜索以字母 "rt" 开头。	临时实时搜索 `rt_1347456938.31`
复制的搜索	搜索是以 "rsa_" 开头的复制搜索结果 (artifact)。将完成的搜索复制到最初运行搜索的搜索头以外的搜索头时，这些 SID 会出现在搜索头群集中。	
复制计划的搜索	"Rsa_scheduler_" 是计划程序分发的复制搜索结果的前缀。	搜索 “foo” 由 **rsa_scheduler** 运行，无用户上下文，保存在应用 **unix** 中。`rsa_scheduler_nobody_unix_foo_at_1502989576_051d958b8354c580`
报表加速搜索	这些是探测搜索，由数据模型加速创建，用于检索所有对等节点的加速百分比。	`SummaryDirector_1503528948.24878_D12411CE-A361-4F75-B90A-28AFDA88151B`

Dispatch 目录维护

Dispatch 目录获取程序每 30 秒会遍历一次所有项目。它会基于项目的最后一次访问时间及其配置的持续时间 (TTL) 或生存期来删除已过期的项目。

有关使用 Splunk Web 更改搜索项目的默认生存期的信息，请参阅“延长任务生存期”。

Dispatch 目录中的搜索项目生存期

默认生存期值取决于搜索类型。搜索完成后，生存期开始倒计时。下表按搜索类型列出了默认生存期值。

搜索类型	默认生存期
手动运行保存的搜索或临时搜索	10 分钟
来自对等节点的远程搜索	10 分钟
计划的搜索	生存期会因所选告警操作（若有）而异。如果告警有多个操作，则生存期最长的操作成为搜索项目的生存期。若没有操作，值由 `savedsearches.conf` 文件中的 `dispatch.ttl` 属性确定。`dispatch.ttl` 属性的默认值是计划周期的两倍 (2x)。 告警操作确定计划的搜索的默认生存期。 • 电子邮件、RSS 和追踪告警操作的默认生存期是 24 小时。 • 脚本告警操作的默认生存期是 10 分钟。 • 摘要索引和填充查找告警操作的默认生存期是 2 分钟。
显示数据来源的计划的搜索	30 秒
子搜索	5 分钟 子搜索会生成两个搜索特定目录。现有一个子搜索的搜索特定目录和一个搜索（该搜索使用子搜索）的搜索特定目录。这些目录具有不同的生存期值。

更改搜索项目生存期

有多种方式可用于更改搜索项目生存期。修改默认搜索行为会影响没有应用其他生存期值或 TTL 的搜索。

请参阅“如何编辑配置文件”。

搜索行为类型	过程
全局搜索行为	在 limits.conf 文件中，设置 [搜索] 段落中的 ttl 或 remote_ttl，或 [子搜索] 段落中的 ttl。 修改 limits.conf 可提供搜索的默认值，以便影响没有应用其他生存期值的搜索。
搜索特定行为	在 savedsearches.conf 文件中，为单个搜索设置 dispatch.ttl。 或在 Splunk Web 中保存搜索时单独为其设置一个值。 这一操作会覆盖默认搜索行为。
带告警操作的搜索	在 alert_actions.conf 中，设置一个 ttl 值以指定当特定告警操作触发时搜索项目的最小生存期。 这一操作会覆盖应用于搜索的所有较短生存期。

基于目录存在的时间清除 Dispatch 目录

随着添加到 Dispatch 目录的项目越来越多，项目量可能对搜索性能或出现在 UI 中的警告产生负面影响。警告阈值是基于 limits.conf 文件中的 dispatch_dir_warning_size 属性。

dispatch_dir_warning_size 属性的默认值是 5000。

您可以将搜索特定目录从 Dispatch 目录移动到另一个 Dispatch 目录、目标和目录。您可以使用 clean-dispatch 命令移动搜索特定目录。您必须指定一个晚于搜索特定目录最新修改时间的时间。目标目录必须与 Dispatch 目录位于同一文件系统。

运行命令 $SPLUNK_HOME/bin/splunk clean-dispatch help，了解如何使用 clean-dispatch 命令。

有关清除 Dispatch 目录的更多信息，请参阅*《故障排除手册》*中的“搜索任务过多”。

限制搜索进程内存使用率

Splunk 软件可配置为，当搜索任务进程所使用的常驻内存量超过所配置的阈值时，自动终止该搜索任务进程。

当出现以下情况时，您可能需要使用此功能：

- 您想主动避免失控搜索导致一个或多个搜索节点瘫痪的情形。
- 您已经遇到此情形，不想它再次发生。
- 在分布式管理控制台中，**搜索活动：实例**视图显示一个或多个所消耗的物理内存已达危险值的搜索。您可以在**内存消耗排名前 10 的搜索**面板中查看此信息。

如果您使用的是 Splunk Cloud 并想调整此阈值，则必须向 Splunk 支持提交问题，因为您没有访问 limits.conf 文件的权限。

这个阈值是用来做什么的？

启用此阈值可限制每个搜索进程可使用的最大内存。超过内存大小阈值的搜索进程会被自动杀掉，以减少损失。

此阈值使用由平台检测记录的进程资源使用率信息。因此，此功能仅在 *nix、Solaris 和 Windows 平台中可用。

- 请参阅*《REST API 参考手册》*中的“自检端点描述”。
- 请参阅*《故障排除手册》*中的“有关平台检测框架”。

搜索内存会进行定期检测，因此短暂的尖峰值可能会超过配置的限值。

此功能已融入 Dispatch 目录获取程序中。因此，当获取程序组件停止响应时也会导致搜索内存检测频率中的停滞。

启用搜索进程内存阈值

搜索进程内存跟踪默认是禁用的。

1.请参阅*《管理员手册》*中的“如何编辑配置文件”。

2.打开 `limits.conf` 文件。

3.在 [搜索] 段落中，将 `enable_memory_tracker` 属性的设置更改为 `true`。

4.查看和调整内存限值。

此限值可设置为一个绝对值或所识别的系统最大值的百分比，分别使用 `search_process_memory_usage_threshold` 或 `search_process_memory_usage_percentage_threshold`。通常搜索的这两种值都会检测，并应用较低的值。请参阅**《管理员手册》**中的 limits.conf.spec。

5.要启用配置更改，必须重启 Splunk Enterprise。

阈值活动记录在哪里？

如果阈值导致搜索进程在搜索头上终止，则会插入一个错误至搜索项目文件 `info.csv`。如果搜索是通过 Splunk Web 运行的，在 Splunk Web 中也会显示此错误消息。此错误消息会说明此进程已被终止，并且会指明限值设置和具体值。

如果阈值导致搜索进程在搜索节点上终止，则会在 StreamedSearch 目录中的 `splunkd.log` 文件中记录一个 WARN 消息。

在两种情况下都会在 DispatchReaper 目录的 `splunkd.log` 文件中记录一个 WARN 消息。

此消息类似于：

```
Forcefully terminated search process with sid=... since
its \[relative physical or physical] memory usage (... \[MB or %])
has exceeded the \[relative physical or physical] memory
threshold specified in limits.conf/...setting name.. (...setting value...)
```

从操作系统管理 Splunk Enterprise 任务

如果您在 Microsoft Windows 或 *nix 上有 Splunk Enterprise，您可以从操作系统管理搜索任务，如本主题所述。有关如何在 Splunk Web 中管理搜索任务的信息，请参阅本手册的“管理搜索任务”。

在 *nix 中管理任务

当搜索任务运行时，它会在操作系统中将自己显示成名为 `splunkd search` 的进程。您可以在 OS 命令行管理该任务的基本进程。

要查看该任务的进程及其参数，请键入：

```
> top
> c
```

此时会显示所有正在运行的进程以及这些进程的全部参数。

键入 `ps -ef | grep "search"` 会将此列表中的所有 Splunk 搜索进程隔离。类似于以下样式：

```
[pie@fflanda ~]$ ps -ef | grep 'search'
530369338 71126 59262   0 11:19AM ??         0:01.65 [splunkd pid=59261] search --id=rt_1344449989.64 --
maxbuckets=300 --ttl=600 --maxout=10000 --maxtime=0 --lookups=1 --reduce_freq=10 --rf=* --user=admin --pro --
roles=admin:power:user  AhjH8o/Render TERM_PROGRAM_VERSION=303.2
530369338 71127 71126   0 11:19AM ??         0:00.00 [splunkd pid=59261] search --id=rt_1344449989.64 --
maxbuckets=300 --ttl=600 --maxout=10000 --maxtime=0 --lookups=1 --reduce_freq=10 --rf=* --user=admin --pro --roles=
```

每个搜索任务有两个进程；第二个进程是“辅助”进程，由 `splunkd` 进程使用，以便根据需要执行进一步的工作。主要任务将使用系统资源。如果终止了主进程，辅助进程也会自行终止。

进程信息包括：

- 搜索字符串 (search=)
- 该任务的任务 ID (id=)
- 任务的项目（任务生成的输出）在磁盘上保留并处于可用状态的 ttl 或时间长度 (ttl=)
- 运行任务的用户 (user=)
- 该用户所属的角色 (roles=)

在任务运行期间，其数据将被写入 `$SPLUNK_HOME/var/run/splunk/dispatch/<job_id>/` Scheduled jobs（计划的已保存搜索），包括作为目录名称一部分保存的搜索名称。

进程的 `ttl` 值决定了数据将在该位置保留多长时间，甚至在您终止一个任务后。在 OS 中终止一个任务时，如果还需

要删除其项目，则可能需要在终止任务之前查看其任务 ID。

在 Windows 中管理任务

在 Windows 中，每个搜索也同样作为独立进程运行。Windows 没有与 *nix `top` 命令等效的命令行命令，但它提供了几种方法可用于查看正在执行的搜索任务的命令行参数。

- 使用 Process Explorer 实用工具 (http://technet.microsoft.com/en-us/scriptcenter/dd742419.aspx) 查找执行搜索的进程的命令行。
- 在 Powershell (http://technet.microsoft.com/en-us/scriptcenter/dd742419.aspx) 命令行环境中使用 `TASKLIST` 和 `Get-WMIObj` 命令来获取搜索任务的 `ProcessID` 和 `CommandLine` 参数。

在搜索运行时，用于此搜索的数据会写入 `%SPLUNK_HOME\var\run\splunk\dispatch\<epoch_time_at_start_of_search>.<number_separator>` 目录中。保存的搜索将写入到相似的目录中，其命名约定为 "`admin__admin__search_`"，并且除 UNIX 时间外还带有一个随机生成的哈希数字。

使用文件系统管理任务

您可以在一个任务的项目目录中创建和删除文件，以此方式来管理此任务：

- 要取消一个任务，应进入该任务的项目目录创建一个 "cancel" 文件。
- 要保留该任务的项目（并忽略其 ttl 设置），应创建一个 "save" 文件。
- 要暂停一个任务，应创建一个 "pause" 文件，要取消暂停，应删除此 "pause" 文件。

保存和计划搜索

保存搜索

创建搜索时，有几种保存搜索的方式供您选择。在“搜索”应用中，这些选项位于**另存为**下拉菜单中。

另存为选项	描述	更多信息
报表	当创建希望再次运行的搜索时，您可以将它另存为报表。	请参阅*《报表手册》*中的“创建和编辑报表”
仪表板面板	您还可以将搜索另存为仪表板面板。仪表板可以有一个或多个面板，可以显示表格或地理可视化中的搜索结果。	请参阅*《仪表板和可视化》*手册中的“入门”。
告警	有些搜索会及时提供您想要收到的信息。您可以将搜索另存为告警。告警是一项保存的搜索根据搜索结果触发的操作。该操作可能会发送电子邮件或运行脚本。	请参阅*《告警手册》*中的“关于告警”。
事件类型	您可以将搜索另存为事件类型。事件类型是一种分类系统，旨在帮助您理解数据。事件类型可让您通过大量数据进行筛选，查找类似的模式，并创建告警和报表。	请参阅*《知识管理器手册》*中的“关于事件类型”。

另请参阅

计划搜索

计划搜索

您可以**计划搜索**以定期运行。

选项	描述	更多信息
报表	将搜索另存为报表之后，您可以将该报表转换为计划的报表。计划的报表是以计划的时间间隔运行的报表，每次运行时都将触发一个操作。计划的报表可以触发两种操作：发送电子邮件和运行脚本。	请参阅*《报表手册》*中的“计划报表”。
仪表板面板	有几个创建计划的报表的选项： • 您可以基于计划的报表创建一个仪表板面板。 • 当您将**临时搜索**另存为仪表板面板之后，面板将搜索作为内联搜索进行引用。您可以编辑仪表板面板，以将搜索转换为报表，然后计划该报表。	请参阅*《仪表板和可视化》*手册中的“使用仪表板面板”。
告警	您可以创建计划的告警以定期搜索事件。您可以通过配置计划、触发条件和限制来自定义告警。	请参阅*《告警手册》*中的“创建计划的告警”。

另请参阅

保存搜索

导出搜索结果

导出搜索结果

您可以导出 Splunk 部署中的搜索结果，并将数据转发到第三方系统，如本主题所述。

可用导出方法是什么？

Splunk 平台提供了若干导出方法：

- 使用 Splunk Web 导出数据
- 使用 CLI 导出数据
- 使用 SDK 导出数据
- 使用 REST API 导出数据
- dump 搜索命令
- 数据转发

Splunk 应用

- 部署和使用 Splunk App for CEF
- 部署和使用 Splunk DB Connect
- Hadoop Connect
- 安装并使用带有 Microsoft Excel 的 Splunk ODBC 驱动程序
- 安装并使用带有 MicroStrategy 的 Splunk ODBC 驱动程序
- 安装并使用带有 Tableau 的 Splunk ODBC 驱动程序

导出选项

所选择的导出方式取决于涉及的数据量和交互性的水平。比如，通过 Splunk Web 的单个按需搜索导出可能适合于低数据量导出。相反，如果您想设置更高数据量的计划性导出，则 SDK 和 REST 选项工作效果最佳。

对于大型导出，搜索数据检索的最稳妥方法是命令行界面 (CLI)。您可以从 CLI 使用多种 Splunk SDK 对外部应用程序的搜索进行定制。来自 CLI 的 REST API 也能工作，但建议只用于内部使用。

相对专业知识水平，Splunk Web 和 CLI 方法比 SDK 和 REST API 更容易访问，因为后者在使用软件开发工具包或 REST API 端点工作时需要之前的经验。

方法	数据量	交互	注释
Splunk Web	低	按需，交互	易于获得按需导出
CLI	中	按需，低交互	易于获得按需导出
其余	高	自动，用于计算机对计算机最佳	SDK 下面工作
SDK	高	自动，用于计算机对计算机最佳	用于自动时最佳

支持的导出格式

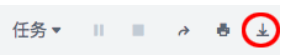
可以使用下列格式导出 Splunk 数据：

- 原始事件（用于结果为原始事件和非已计算字段的搜索结果）
- CSV
- JSON
- XML
- PDF（用于保存的搜索，使用 Splunk Web）

使用 Splunk Web 导出数据

您可以将搜索、报表或数据透视表任务的事件数据以多种格式导出。导出后即可将此文件存档，或在第三方图表应用程序中使用。

1. 在运行搜索、报表或数据透视表之后，单击“导出”按钮。“导出”按钮是其中一种搜索操作按钮。



如果此按钮不可见，则是由系统管理员隐藏了起来以防止导出数据。

使用“导出结果”窗口为导出文件指定格式和名称：

在导出结果前，有时必须重新运行搜索。请参阅“导出时触发搜索”以重新运行。

2. 单击**格式**并选择导出搜索结果时所需的格式。
   所支持的格式取决于所操作任务项目的类型。

格式	临时搜索	保存的搜索	注释
CSV	X	X	
JSON	X	X	
PDF		X	如果搜索是保存的搜索，如报表，则可用 PDF 格式导出。
原始事件	X	X	如果搜索会生成计算的数据且此数据会显示在“统计”选项卡中，则不可使用原始事件格式导出。
XML	X	X	

3. 可选。在**文件名**字段中，可键入待保存事件数据的导出文件的名称。如果未指定文件名，则会用搜索任务 ID 作为文件名来创建文件。搜索任务 ID 是搜索运行时的 UNIX 时间。例如，`1463687468_7.csv`。
4. 可选。在**结果数**字段中，可指定要导出的结果数量。如果未指定数量，则会导出所有事件。例如，如果在**结果数**字段中指定 500，则仅会导出搜索返回的前 500 个结果。
5. 单击**导出**将任务事件保存到导出文件中。

此文件保存在浏览器或操作系统的默认下载目录中。例如，对于大部分 Windows 和 Mac OS X 用户来说，导出文件位于默认**下载**目录中。对于 Linux，查看 XDG 配置文件来确认下载目录。

导出时触发搜索以重新运行

如果搜索返回大量结果，则可能并非所有结果都将存储在**搜索任务项目**中。

当导出搜索结果时，导出过程是基于搜索任务项目，而非搜索应用中的结果。如果项目不包含完整的结果集，“导出结果”对话框的底部会显示一条消息，告诉您 Splunk 软件会在结果导出前重新运行此搜索。

搜索头判定无法检索任务项目中的所有事件时重新运行搜索。搜索头按照以下逻辑决定何时重新运行搜索：

- 如果搜索不是一个报表，且以下情况真实存在。
  - 搜索未完成
  - 搜索正在使用远程时间线
  - 搜索头判定搜索未保留所有事件

当导出大量数据时扩展会话超时

若使用“导出”按钮导出大量数据时，此会话可能会在完成前出现超时。会话超时限制可以延长。

1. 单击**设置 > 服务器设置 > 常规设置**。
2. 在 **Splunk Web** 部分，增加“会话超时”字段中的数字。
3. 单击**保存**。

增加超时设置以便为浏览器和 Splunk Web 之间连接的建立提供更多时间。

转发数据到第三方系统

所导出的数据可以转发给第三方系统。

- 若需简要概述，请参阅本手册中的“转发数据到第三方系统”。
- 若需详情信息，请参阅*《转发数据》*中的“转发数据到第三方系统”。

通过报表将结果发送给相关方

您可以为报表添加计划使其定期运行，并且通过电子邮件将结果发送到项目相关方。电子邮件可以在邮件中以表格的形式列出报表结果，也可以 CSV 或 PDF 附件的方式发送报表结果。电子邮件还可以包含 Splunk Enterprise 报表结果的链接。请参阅*《报表手册》*中的“计划报表”。

使用 CLI 导出数据

命令行界面 (CLI) 对于脚本来说非常简单，能够处理自动操作，可以使大量数据的处理速度比 Splunk Web 还要快，效率更高。要通过 CLI 访问 Splunk Enterprise，您需要 Splunk Enterprise 服务器的 shell 访问权限，或者访问远程 Splunk 服务器正确端口的权限。如果您使用的是 Splunk Cloud，则没有 Splunk Cloud 部署的 shell 访问权限，因此无法使用 CLI 来导出数据。

使用 CLI 的导出数据语法如下所示：

```
splunk search [eventdata] -preview 0 -maxout 0 -output [rawdata|json|csv|xml] > [myfilename.log] ...
```

默认情况下，您最多可以导出 100 个事件。要增加此数量，请使用 `-maxout` 参数。例如，当您包含 `-maxout 300000` 时，您可以导出 300,000 个事件。将 `-maxout` 设置为 0 以导出无限数量的事件。

有关 Splunk Enterprise CLI 的更多信息，请参阅*《管理员手册》*中的“关于 CLI”。

CLI 输出命令示例

此 CLI 示例从 `_internal` 索引获取搜索字符串指定的时间范围内发生的事件，并且以原始数据格式将其中 200,000 个事件输出至 `test123.dmp` 文件。

```
splunk search "index=_internal earliest=09/14/2015:23:59:00 latest=09/16/2015:01:00:00 " -output rawdata -maxout
200000 > c:/test123.dmp
```

使用 Splunk REST API 导出数据

使用 Splunk REST API 访问来自命令行或 Web 浏览器的数据。

适用于 Splunk Cloud 部署的 REST API 访问

如果您有一个自动服务的 Splunk Cloud 部署并且想使用 Splunk REST API，则要向 Splunk 支持提交问题，请求启用 API。更多信息，请参见*《REST API 教程》*中的“将 REST API 用于 Splunk Cloud”。

导出数据

运行搜索任务开始导出数据以生成结果。然后您可以将此搜索结果数据导出到一个文件中。

1. 在 `/services/search/jobs/` 使用 POST 操作运行搜索任务。如果您正在使用自定义时间范围，使用 POST 请求跳过它。

   ```
   curl -k -u admin:changeme \
        https://localhost:8089/services/search/jobs/ -d search="search sourcetype=access_* earliest=-7d"
   ```
2. 为搜索获取搜索任务 ID (SID)。`/jobs` 端点返回包括 `<sid>` 的 XML 响应或搜索任务 ID。

   ```
   <?xml version='1.0' encoding='UTF-8'?>
   <response>
     <sid>1423855196.339</sid>
   </response>
   ```

   您也可以通过在**搜索任务查看器**中查看任务，以获取搜索任务 ID。导航至**活动 > 任务**以打开任务管理器，查找您刚运行的搜索任务，然后单击**检查**。“搜索任务查看器”在一个单独的窗口中打开。
3. 在 `/results` 端点上使用 GET 请求将搜索结果导出到一个文件。确保在 GET 请求过程中执行以下操作：
   - **确定对象端点。**
     要在您的应用程序中，为您的用户查看当前可用对象端点列表，请导航至 `https://localhost:8089/servicesNS/<user>/<app>/`。
     例如：
     `https://localhost:8089/servicesNS/admin/search/saved/searches/`
   - **确定搜索任务用户和应用程序。**
     以下示例将 `<user>` 定义为 `admin`，并且将 `<app>` 定义为 `search`。
   - **确定导出格式**
     使用 `output_mode` 参数指定以下一种可用导出格式。格式名称小写，如图所示。
     `atom | csv | json | json_cols | json_rows | raw | xml`

     此示例将搜索结果导出到 JSON 文件。

```
curl -u admin:changeme \
     -k https://localhost:8089/servicesNS/admin/search/jobs/1423855196.339/results/ \
     --get -d output_mode=json -d count=5
```

另请参阅

有关 `/jobs` 和 `/export` 端点的更多信息，请参见《REST API 参考》中的以下信息。

- search/jobs
- search/jobs/export

请参阅*《REST API 教程》*中的“使用 REST API 创建搜索”。

使用 Splunk SDK 导出数据

Splunk 软件开发工具包 (SDK) 使软件开发人员可以使用常见编程语言创建 Splunk 应用。Splunk SDK 让您可使用第三方报表工具和门户集成 Splunk 部署，将搜索结果包含在应用程序中，以及提取大量数据以用于归档。要使用 Splunk SDK，您应该很熟悉 SDK 知识和开发能力。

Splunk 为 Python、Java、JavaScript 和 C# 提供 SDK。当您在这些 SDK 中运行导出搜索时，会立刻运行搜索，此操作不会为搜索创建任务，并立刻开始启动流结果。

Splunk SDK 建立在 Splunk REST API 顶部。为 REST API 端点提供更简单的界面。由于代码行数较少，因此您可以编写应用程序，这样您就可以：

- 创建和运行身份验证搜索
- 添加数据
- 为数据建立索引
- 管理搜索任务
- 配置 Splunk

有关 Splunk SDK 的更多信息，请阅读 Splunk 开发人员门户的“Splunk SDK 概述”。

使用 Python SDK 导出数据

适用于 Python 的 Splunk SDK 能够让您编写与 Splunk 部署交互的 Python 应用程序。使用 Python SDK 导出的搜索能够在历史模式和实时模式中运行。立刻开始启动，并且立即获取流结果，您可以将他们集成到 Python 应用程序。

使用 Python SDK 执行导出搜索。

1. 设置您要搜索的参数。如下示例将参数设置为最后一个小时内 `splunklib` 的导出搜索。

```
import splunklib.client as client
import splunklib.results as results
```

2. 必要时更改或获取这些值。

```
HOST = "localhost"
PORT = 8089
USERNAME = "admin"
PASSWORD = "changeme"
```

3. 运行正常模式搜索。

```
service = client.connect(
    host=HOST
    port=PORT,
    username=USERNAME,
    password=PASSWORD)

rr = results.ResultsReader(service.jobs.export("search index=_internal earliest=-1h | head 5"))
```

4. 使用 ResultsReader 获得结果并显示。

```
for result in rr:
    if isinstance(result, results.Message):
```

```
        # Diagnostic messages might be returned in the results
        print '%s: %s' % (result.type, result.message)
    elif isinstance(result, dict):
        # Normal events are returned as dicts
        print result
assert rr.is_preview == False
```

使用 Java SDK 导出数据

当使用 Java 时，Java SDK 能够执行和导出搜索。

要使用 Java SDK 执行导出搜索，请使用 CLI 在 `/splunk-sdk-java` 目录中运行以下示例：

```
java -jar dist/examples/export.jar main --username="admin" --password="changeme"
```

“导出” 应用程序将“主”索引导出到已保存至当前工作目录的 `export.out`。如果您要再次运行此应用程序，请在重试之前删除 `export.out`。如果您不进行此操作，则会得到一个错误讯息。

如下为 Java SDK 的其他 CLI 示例。显示如何包含搜索查询，以及将输出格式更改为 JSON。

```
java -jar dist/examples/export.jar main --search="search sourcetype=access_*" json
```

使用 JavaScript 导出来导出数据

Javascript 导出端点能够在 Javascript 结构中导出 Splunk 数据。尽管 Splunk Javascript SDK 目前还不支持 Javascript 导出端点，您可以使用节点 javascript (.js) 应用程序请求来导出数据。

要使用 Javascript 导出端点执行导出搜索：

1.加载请求模块。将请求设计为 http/https 调用的最简单方法。

```
var request = require('request');
```

2.调用 **get** 以发布 GET 请求。输入以下参数：

- `strictSSL` - 当设置为 false 时，`strictSSL` 会阻止请求验证 Splunk 部署返回的服务器证书，因为它在默认情况下并不是一个有效的证书。
- `uri` - 随附导出端点路径，提供 Splunk 主机的 `uri`。JSON 响应在查询字符串中指定。
- `qs` - 设置 `qs` 以提供搜索参数。通过这种方式，您不必将 URI 编码传入搜索字符串。

```
request.get(
    {
        strictSSL: false,
        uri: 'https://localhost:8089/servicesNS/admin/search/search/jobs/
              export?output_mode=json',
        qs: {
            search: 'search index=_internal'
        }
    }
)
```

3.调用 `auth` 以使用 HTTP 基本身份验证，并传递您的 Splunk 用户名和密码。

```
.auth('admin', 'changeme', false)
```

4.将结果通过管道符传递给 `stdout`。

```
.pipe(process.stdout);
```

使用 C# SDK 导出数据

使用 C# SDK 导出搜索立刻异步运行，不会为搜索创建任务，但立刻开始启动流结果。当导出大量的历史或实时数据时，C# SDK 将非常有用。

要使用 C# SDK 执行导出搜索：

1.使用 StreamReader 创建预览搜索。

```
SearchPreviewStream searchPreviewStream;
```

2.导出搜索结果预览。

```
using (searchPreviewStream = service.ExportSearchPreviewsAsync("search index=_internal | head 100").Result)
{
    int previewNumber = 0;
```

3.通过每个搜索结果预览枚举。

```
    foreach (var searchPreview in searchPreviewStream.ToEnumerable())
    {
        Console.WriteLine("Preview {0:D8}: {1}", ++previewNumber, searchPreview.IsFinal ? "final" : "partial");
        int recordNumber = 0;

        foreach (var result in searchPreview.Results)
        {
            Console.WriteLine(string.Format("{0:D8}: {1}", ++recordNumber, result));
        }
    }
}
```

使用转储命令导出数据

您可以使用 `dump` 搜索命令将大量事件集导出到本地磁盘。此命令可以与 CLI、Splunk SDK 和 Splunk Web 一起使用。

`dump` 命令的基本语法是：

```
dump basefilename=<string> [rollsize=<number>] [compress=<number>] [format=<string>] [fields=<comma-delimited-string>]
```

`<format>` 是创建中 dump 文件的数据格式。可用格式选项有：`raw`、`csv`、`tsv`、`xml` 和 `json`。

有关搜索示例和必选及可选参数的完整解释，请参阅*《搜索参考》*中的 `dump` 命令。

转发数据到第三方系统

Splunk 软件可通过以下方式将数据转发给第三方系统：

- 通过纯 TCP 套接字
- 打包到标准 syslog 中

要将数据转发给第三方系统，必须编辑 `outputs.conf`、`props.conf` 和 `transforms.conf` 文件以配置重型转发器。此导出方式类似将您的数据路由到其他 Splunk 部署。您可以按主机、数据来源或数据来源类型筛选数据。

请参阅*《转发数据》*手册中的"转发数据到第三方系统"。

编写自定义搜索命令

关于编写自定义搜索命令

您可以自定义内置命令或针对自定义处理或计算编写自己的搜索命令，以此方式扩展 Splunk 搜索处理语言 (SPL)。

如果您使用的是 Splunk Cloud，则没有访问 Splunk 部署中文件系统的权限。如果要创建自定义搜索命令，请向 Splunk 支持提交问题。

下表描述了可用于创建自定义搜索命令的协议、格式和 SDK。

支持的协议	描述	支持的可执行格式	SDK
自定义搜索命令协议，版本 2	用于为大范围的平台和可执行格式创建自定义命令。	.bat、.cmd、.exe、.js、.pl、.py、.sh	适用于 Python 的 Splunk SDK
自定义搜索命令协议，版本 1	与适用于 Python 的 Splunk SDK 一起创建适用于 Python 的自定义命令。 仅与 Intersplunk.py SDK 一起支持现有自定义命令。	.pl、.py	适用于 Python 的 Splunk SDK Intersplunk.py

使用版本 2 自定义搜索命令协议的自定义搜索命令，可以多种编程语言实现。这些自定义命令甚至可实现为平台特定的二进制文件。

相比之下，使用版本 1 协议的自定义搜索命令仅能在 Python 中实现。使用版本 1 协议的自定义命令只能使用包含在 Splunk 软件中的 Python 注释器来运行。

关于 SDK

使用适用于 Python 的 Splunk SDK 创建自定义搜索命令。适用于 Python 的 Splunk SDK 包含若干模版，可用于构建新的自定义搜索命令。

Intersplunk.py 是旧版的 SDK，只能用于支持用版本 1 协议构建的现有自定义搜索命令。Intersplunk.py SDK 不应用于新的自定义搜索命令。

关于协议

版本 2 协议

使用版本 2 自定义搜索命令协议有两大重要好处。

- 通过版本 2 协议，外部程序通过一次调用处理整个 Splunk 搜索结果集。整个搜索只会调用外部程序一次，大幅减少了运行时开销。
- 和版本 1 协议相比，版本 2 协议要求的配置属性较少。
- 支持非 Python 自定义搜索命令，例如，C++、Go、Java 和 JavaScript (Node.js)。
- 支持平台特定的可执行文件和二进制文件。可以用编译语言，如 C++，在多个平台上编写自定义搜索命令。

版本 1 协议

版本 1 自定义搜索命令协议分组处理事件，每组 50,000 个事件。对于大型结果集，外部程序要调用和终止多次。

另请参阅

- 编写自定义搜索命令
- 自定义命令的安全责任

编写自定义搜索命令

自定义搜索命令是一个可执行文件，可读入和写出数据。它可以是一个 Python 脚本、C++ 程序或其他可执行二进制文件。为简化描述，此类文件在本文档中称为**可执行文件**。

搜索命令可执行要求

搜索命令可执行文件应该位于适当目录中，且其名称符合某些基本规则。

可执行文件位置

搜索命令可执行文件必须位于相应的 $SPLUNK_HOME/etc/apps/<app_name>/bin/ 目录。大多数 Splunk Enterprise 随附的可执行文件都与搜索和报表应用相关联，并且存储于 $SPLUNK_HOME/etc/apps/search/bin 中。请参阅此目录中的可执行文件查看相关示例。

如果您使用的是 Splunk Cloud 并想创建自定义搜索命令，则必须向 Splunk 支持提交问题。您没有访问 Splunk Cloud 部署中文件系统的权限。

可执行文件名称
: 可执行文件的名称应与您要在 Splunk 搜索中调用的命令名称一致，且符合命名约定 <command_name>.py。搜索命令名称只能包含字母数字（a-z 和 0-9）字符。新命令应拥有唯一的名称。其名称不可与任何内置或已有自定义命令一致。

通过适用于 Python 的 Splunk SDK 构建命令可执行文件

通过版本 2 协议，您可以使用外部搜索可执行文件的平台特定版本。请参阅“为自定义搜索命令选择位置”。

注意：有关适用于 Python 的 Splunk SDK 的处理输入、输出和错误等信息，请参阅“Splunk 开发门户”中的“如何通过适用于 Python 的 Splunk SDK 创建自定义搜索命令”。

在构建搜索命令可执行文件之后，必须将自定义命令添加到 Splunk 部署。

通过 Intersplunk.py 文件构建命令可执行文件

作为构建可执行文件的一部分，您需要指定如何处理输入、发送输出和处理错误。

处理输入

通过 Intersplunk.py 文件，可执行文件的输入格式应采用纯 CSV 或 Intersplunk 格式，即标头部分的格式先依照空行格式再依照纯 CSV 体格式。

解释可执行文件输入最简单的方式是使用 splunk.Intersplunk.readResults，它采用 3 个可选参数并返回字典列表（表示输入事件的列表）。可选参数为 input_buf、settings 和 has_header。

inputbuf
: inputbuf 参数指定从何处读取输入。如果参数设为 None（默认值），则会从 sys.stdin 读取输入。

settings
: settings 参数应该是一个字典，其中储存了在输入标头中发现的所有信息。此参数的默认值是 None，表示没有记录任何设置

has_header
: has_header 参数指定是否使用输入标头。此参数的默认值为 True。

要表示您的可执行文件要使用标头，使用 enableheader 属性。enableheader 属性的默认值为 True，表示输入将包含标头部分，并且您正使用 Intersplunk 格式。

如果可执行文件在输入中不需要标头部分（enableheader 设为 False），您可以直接使用 Python csv 模块来读取输入。例如：

```
导入 csv

r = csv.reader(sys.stdin)
for l in r:
...
```

使用 Python CSV 的好处是，您可以随时中断 for 循环，并且只有在那个点迭代输入中的行才会读入内存中。在某些使用情况中，这会带来更好的性能。

发送输出

您也可以使用 Intersplunk.py 文件来构建可执行文件的输出。splunk.Intersplunk.generateErrorResults 采用字符串并将正确的错误输出写入 sys.stdout。splunk.Intersplunk.outputResults 采用字典对象列表并将相应的 CSV 输出写入 sys.stdout。

要输出数据，请添加以下内容：

```
splunk.Intersplunk.outputResults(results)
```

可执行文件的输出应该为纯 CSV。对于错误条件，只返回 CSV，其中包含单个 "ERROR" 列和一个含有消息内容的行（标题行除外）。

处理错误

如果可执行文件不具有属性 supports_getinfo = true，则传递到可执行文件的参数（位于 sys.argv 中）与用于在搜

索语言中调用您的自定义命令的参数相同。supports_getinfo 属性表示可执行文件的第一个参数是 __GETINFO__ or __EXECUTE__。这样，您就能够在分析时使用命令参数调用可执行文件来检查语法错误，然后再执行任意搜索。如果此时存在任何错误，任何实际执行的搜索查询都将中断。如果使用 __GETINFO__ 进行调用，您还可以根据参数动态指定可执行文件的属性（例如，流或非流）。

如果您的可执行文件将属性 supports_getinfo 设置为 'true'，则应该首先进行类似如下所示的调用：

(isgetinfo, sys.argv) = splunk.Intersplunk.isGetInfo(sys.argv)

此调用将从 sys.argv 中去除第一个参数，并检查您是在 GETINFO 模式还是 EXECUTE 模式中。如果目前正处于 GETINFO 模式中，则可执行文件应使用 splunk.Intersplunk.outputInfo() 返回可执行文件的属性；或者，如果参数无效，返回 splunk.Intersplunk.parseError()。

outputInfo() 及其参数的定义如下所示：

def outputInfo(streaming, generating, retevs, reqsop, preop, timeorder=False)

也可以在 commands.conf 文件中设置这些属性。请参阅“如何编辑配置文件”。

另请参阅

- 要确定执行命令的位置，请参阅“为自定义搜索命令选择位置”。
- 要了解所支持的协议和 SDK，请参阅“关于编写自定义搜索命令”。
- 要了解关于使用自定义搜索命令的最佳安全实践，请参阅“自定义命令的安全责任”。
- 有关自定义搜索命令参数的信息，请参阅 commands.conf。

为自定义搜索命令选择位置

在创建自定义搜索命令时，必须更新**本地**目录中的 commands.conf 文件。

如果您使用的是 Splunk Cloud，则没有访问 Splunk Cloud 部署中文件系统的权限。您必须向 Splunk 支持提交问题才能将自定义搜索命令添加到您的部署中。

找到正确的 commands.conf 文件

默认目录，$SPLUNK_HOME/etc/system/default，包含预配置版本的配置文件。**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。

相反，您需要确定要存放自定义搜索命令的本地目录。选择正确的位置是必不可少的。

1. 确定命令范围。

范围	描述
应用程序特定的自定义命令	将应用特定的命令添加到应用程序的本地目录中的 commands.conf 文件。应用程序本地目录的位置是 $SPLUNK_HOME/etc/apps/<app_name>/local。
系统范围的自定义命令	将系统范围的命令添加到系统本地目录中的 commands.conf 文件。系统本地目录的位置是 $SPLUNK_HOME/etc/system/local。

2. 确定 commands.conf 文件是否已经存在于您想存放的本地目录中。如果此目录中没有该文件，则在此目录中创建一个空 commands.conf 文件。不要从默认目录复制 commands.conf 文件。

决定要将可执行文件放在哪里

您也需要决定将自定义命令可执行文件放在哪里。Splunk 软件预期会从所有相应的应用程序目录中查找可执行文件。在大多数情况中，应将可执行文件放置于应用命名空间内。

下表说明了可执行文件应存放的位置，基于包含相应自定义命令段落的 commands.conf 文件的位置。

Commands.conf 文件位置	所需脚本文件位置
$SPLUNK_HOME/etc/apps/<app_name>/local	$SPLUNK_HOME/etc/apps/<app_name>/bin 如果您的命令是针对特定平台的，则位置为： $SPLUNK_HOME/etc/apps/<app_name>/<PLATFORM>/bin/
$SPLUNK_HOME/etc/system/local	$SPLUNK_HOME/etc/system/bin

但有一个例外。要使用外部进程运行可执行文件，则不用将可执行文件放置于应用的 bin 目录中。相反，必须在 .path 文件中指定可执行文件的位置。.path 文件必须存放在应用的其中一个 bin 目录。请参阅“使用外部程序处理命

令可执行文件”。

Splunk 软件如何查找您的自定义命令

在相应的本地 `commands.conf` 文件中添加一个段落即可注册自定义搜索命令。

例如，要将自定义命令 "fizbin" 添加到您的部署中，应将以下段落添加到 `commands.conf` 文件中。

```
[fizbin]
chunked = true
```

若需了解添加段落的详细描述，请参阅主题“将自定义命令添加到 Splunk 部署”。但是，在实际将段落添加到 `commands.conf` 文件之前必须先了解软件是如何定位自定义命令可执行文件的。

Splunk 软件会搜索两个位置来查找运行自定义搜索命令所需的可执行文件：

- 平台特定应用程序 `bin` 目录，

```
$SPLUNK_HOME/etc/apps/<app_name>/<PLATFORM>/bin/
```

- 默认应用 `bin` 目录，`$SPLUNK_HOME/etc/apps/<app_name>/bin/`

平台特定的自定义命令

下表列出了所支持的平台特定的 `bin` 目录和所搜索的文件扩展名。

平台架构	目录	文件扩展名
Linux 64 位 x86_64	`linux_x86_64/bin`	.sh、.py、.js 和无扩展名
Linux 32 位 x86	`linux_x86/bin`	.sh、.py、.js 和无扩展名
Mac OS X 64 位 x86_64	`darwin_x86_64/bin`	.sh、.py、.js 和无扩展名
Windows 64 位 x86_64	`windows_x86_64/bin`	.bat、.cmd、.py、.js、.exe
Windows 64 位 x86_64	`windows_x86_64/bin`	.bat、.cmd、.py、.js、.exe

例如，如果在 Linux 64 位 Splunk 实例上使用 `fizbin` 命令，则会搜索以下路径：

```
$SPLUNK_HOME/etc/apps/<app_name>/linux_x86_64/bin/fizbin.sh
$SPLUNK_HOME/etc/apps/<app_name>/linux_x86_64/bin/fizbin.py
$SPLUNK_HOME/etc/apps/<app_name>/linux_x86_64/bin/fizbin.js
$SPLUNK_HOME/etc/apps/<app_name>/linux_x86_64/bin/fizbin
$SPLUNK_HOME/etc/apps/<app_name>/bin/fizbin.sh
$SPLUNK_HOME/etc/apps/<app_name>/bin/fizbin.py
$SPLUNK_HOME/etc/apps/<app_name>/bin/fizbin.js
$SPLUNK_HOME/etc/apps/<app_name>/bin/fizbin
```

Splunk 软件在找到与命令名字相同的文件时会停止搜索，在本例中即为 `fizbin`。

最好在默认应用程序的 `bin` 目录中包含一个可执行文件的平台中立版本，`$SPLUNK_HOME/etc/apps/<app_name>/bin/`。如果有人使用的平台并非由您提供实现而又想在此平台上运行您的自定义命令可执行文件，则这一作法会非常有用。

您也可以显式指定 Splunk 软件应查找的可执行文件，只需在 `commands.conf` 文件中指定 `filename` 属性即可。例如，假设 `fizbin` 命令在 `commands.conf` 文件中定义如下：

```
[fizbin]
chunked = true
filename = fizbin.py
```

在本示例中，Splunk 软件不会尝试猜测文件扩展名，而仅会在 Python 可执行文件应该存放的位置去搜索 `fizbin.py` 文件。

```
$SPLUNK_HOME/etc/apps/<app_name>/linux_x86_64/bin/fizbin.py
$SPLUNK_HOME/etc/apps/<app_name>/bin/fizbin.py
```

处理文件扩展名

在找到自定义命令可执行文件后，Splunk 软件会查找文件扩展名以确定如何运行您的命令。

文件扩展名	操作
`.py`	使用 Splunk 软件中包含的 Python 注释器 `$SPLUNK_HOME/bin/python` 来运行您的命令。
`.js`	使用 Splunk 软件中包含的 Node.js 运行时 `$SPLUNK_HOME/bin/node` 来运行您的命令。
可执行文件没有扩展名，或者未能识别文件扩展名	Splunk 软件尝试不使用注释器而直接运行可执行文件。在基于 UNIX 的平台上，这意味着可执行文件必须拥有可执行位集。

指定命令参数

通过将 `command.arg.<N>` 属性添加到 `commands.conf` 文件段落的方式指定要使用的命令行参数。例如，如果要将诸如 `--verbose` 的标记传递给 `fizbin.py` 可执行文件，可将以下属性添加到 `commands.conf` 文件段落：

```
[fizbin]
chunked = true
filename = fizbin.py
command.arg.1 = --verbose
```

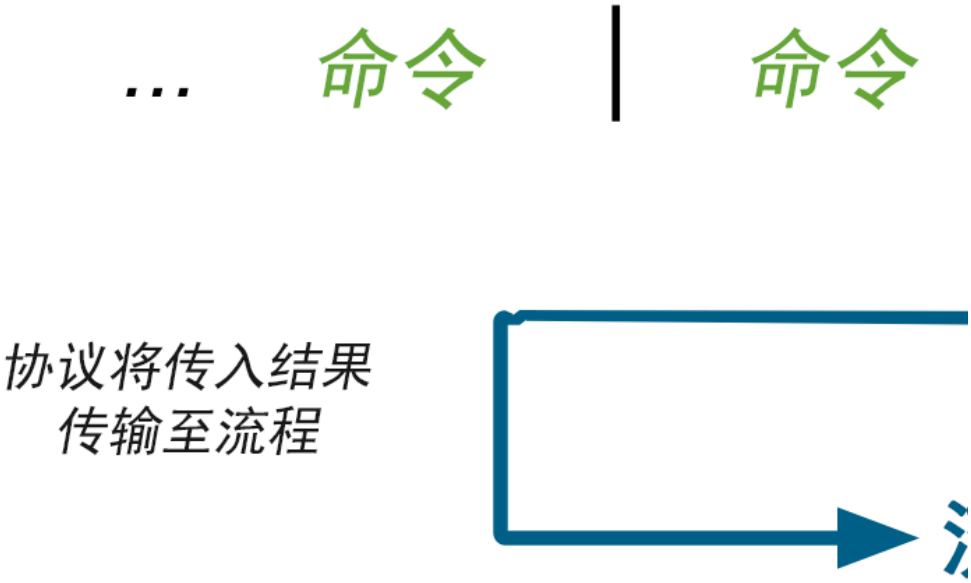
您可以指定任意数量的 `command.arg.<N>` 参数。例如：

```
[fizbin]
chunked = true
filename = java.path   #See the next section for filename examples
command.arg.1 = fizbin.jar
command.arg.2 = -classpath
command.arg.3 = <CLASSPATH>
```

参数的最后一段必须为数字。参数会以数字顺序送出进行处理。跳过的所有数字都会忽略。这些参数中的环境变量，如 `$SPLUNK_HOME`，会进行替换。

使用外部程序处理命令可执行文件

搜索的处理方式是一次一个命令。前一个命令的结果会发送给下一个命令。当搜索到达自定义命令时，搜索会使用协议将前一个命令的结果发送给单独的进程。这一单独的进程可以是内置进程，也可以是外部进程。



Splunk 软件包含一个 Python 注释器和一个 JavaScript 运行时环境。默认情况下，如果您的自定义命令可执行文件是一个 Python 脚本或 JavaScript 文件，则此命令可执行文件会在 Splunk 软件中包含的相应可执行文件处理器上运行。

如果您的可执行文件不是 Python 脚本或 JavaScript 文件，或者您想使用您自己系统中的可执行文件处理器，则必须指定您要用来处理可执行文件的外部程序的位置。

Java 示例

例如，您想要使用 Java 文件运行自定义搜索。Splunk 软件不会包含 Java 运行时间环境 (JRE)。您需要为 JRE 指定路径。

1. 创建 `.path` 文件，如 `$SPLUNK_HOME/etc/apps/<app_name>/bin/java.path`。`.path` 文件必须存放在应用程序的其中一个 `bin` 目录。
2. 在 `.path` 文件中，为 Java 运行时间环境 (JRE) 指定路径。例如，`/usr/bin/java`。
3. 在 `commands.conf` 文件中，指定 `filename` 和 `command.arg.N` 属性来定义您的命令。`filename` 属性不支持绝对路径。以下示例显示了 `fizbin` 命令的段落。

```
[fizbin]
chunked = true
filename = java.path
command.arg.1 = fizbin.jar
command.arg.2 = -classpath
command.arg.3 = <CLASSPATH>
```

在本示例中，Splunk 软件搜索 `java.path` 文件。

所有指定的环境变量，如 `$JAVA_HOME`，在 `.path` 文件中都会进行替换。

Python 示例

例如，您想使用您自己操作系统中的 Python 注释器而非 Splunk 软件所包含的 Python 注释器。

1. 创建一个 `.path` 文件，如 $SPLUNK_HOME/etc/apps/<app_name>/bin/system_python.path。`.path` 文件必须存放在应用的其中一个 `bin` 目录。

2. 在 .path 文件中，为 Python 注释器指定路径。例如，/usr/bin/python。
3. 在 commands.conf 文件中，指定 filename 和 command.arg.1 属性来定义您的命令。filename 属性不支持绝对路径。以下示例显示了 fizbin 命令的段落。

```
[fizbin]
chunked = true
filename = system_python.path
command.arg.1 = fizbin.py
```

在本示例中，Splunk 软件搜索 system_python.path 文件。

所有指定的环境变量，如 $PYTHON_PATH，在 .path 文件中都会进行替换。

另请参阅

将自定义命令添加到 Splunk 部署

将自定义命令添加到 Splunk 部署

必须将自定义命令添加到相应的 commands.conf 配置文件中。

前提条件

请参阅以下主题。

- 编写自定义搜索命令
- 为自定义搜索命令选择位置

如果您使用的是 Splunk Cloud，则没有访问 Splunk Cloud 部署中文件系统的权限。您必须向 Splunk 支持提交问题才能将自定义搜索命令添加到您的部署中。

将自定义命令添加到您的部署所涉及的任务包括：

1. 创建或编辑**本地**目录中的 commands.conf 文件。
2. 在描述此命令的 commands.conf 文件中添加一个新段落。
3. 重新启动 Splunk Enterprise。

添加新段落至本地 commands.conf 文件

编辑本地 commands.conf 文件，并为此命令添加一个段落。

commands.conf 中的每个段落代表一个特定搜索命令的配置。以下示例显示了可启用您自定义命令脚本的段落：

```
[<stanza_name>]
chunked=true
filename = <string>
```

stanza_name 是在搜索中用来调用命令的关键字。stanza_name 也是搜索命令的名称。搜索命令名称必须为小写，且只能包含字母数字（a-z 和 0-9）字符。命令名称必须唯一。stanza_name 不能与其他自定义或内置命令相同。

chunked=true 属性指定此命令使用版本 2 协议。

filename 属性指定自定义命令脚本的名称。filename 属性也指定自定义命令脚本的位置。

例如，要创建自定义命令 "fizbin"，您要在 commands.conf 文件中创建一个段落。

```
[fizbin]
chunked = true
filename = fizbin.py
```

其他可用于描述自定义命令的属性会在本主题的后续章节中进行介绍。

描述命令（版本 2 协议）

版本 2 的自定义搜索命令协议动态判断命令是生成命令、流命令还是生成事件的命令。

另外，它会一直发送验证标记给使用协议的搜索命令。

下表介绍了可用协议指定的属性。

属性	描述
command.arg.<N>	在调用自定义搜索命令脚本时可使用的额外命令行参数。环境变量，如 $SPLUNK_HOME，会进行替换。
filename	使用自定义搜索命令时会运行的脚本的名称。
is_risky	当用户单击链接或键入 URL 将搜索加载到 Splunk Web 中时，如果搜索包含风险性命令，则会发出一则警告。而当用户创建临时搜索时，该警告不会出现。如果您的自定义搜索命令有风险，指定此属性。内置风险性命令的示例包括 `delete` 和 `dump`。要确定您的自定义命令是否具有风险，请参阅*《确保 Splunk Enterprise 安全》*手册中的"风险性命令的防护"。
maxchunksize	外部命令可生成的数据块的最大尺寸，元数据的尺寸加上正文尺寸。如果命令试图生成的数据块大于 maxchunksiz 值，此命令会被终止。
maxwait	自定义搜索命令在生成输出之前可以暂停的最大秒数。

请参阅*《管理员手册》*中的 commands.conf.spec 主题，了解更多关于这些配置属性的信息。

描述命令（版本 1 协议）

您可通过版本 1 协议用于描述自定义命令的部分属性可指定命令类型。

1. 您需要了解命令类型之间的不同。所有搜索命令分为四大类：
   - 可分配流
   - 集中流
   - 生成
   - 转换

   有关命令类型的综合描述，请参阅本手册中的"命令类型"。若需每种类型的内置命令完整列表，请参阅*《搜索参考》*中的"命令类型"。
2. 在 commands.conf 文件中描述自定义搜索命令的类型。
   1. 在 commands.conf 文件中指定 `streaming` 或 `generating` 参数。使用这些属性指定命令是生成命令、流命令还是生成事件的命令。
   2. 也可以通过 `retainevents` 参数指定自定义命令是保留还是转换事件。

有关可配置设置的列表，请参阅*《管理员手册》*中的 commands.conf 参考文件。例如：

generating = [true|false|stream]

- 指定命令是否会生成新事件。
- 如果流化，您的命令会生成新事件 (generating = true)，并且为流命令 (streaming = true)。
- 默认为 false。

streaming = [true|false]

- 指定命令是否为流式命令。
- 默认为 false。

如果自定义搜索命令保留或转换事件，则要包含 `retainevents` 属性：

retainsevents = [true|false]

- 如果命令要保留事件，则指定 'true'，类似 `sort`、`dedup` 或 `cluster` 命令。如果命令要转换事件，则指定 'false'，类似 `stats` 命令。
- 默认为 false。

在段落中您只可以为自定义搜索命令指定少数几个属性。

有关配置属性的完整列表，请参阅*《管理员手册》*中的 commands.conf.spec 主题。

重新启动 Splunk Enterprise

将自定义命令添加到相应 `commands.conf` 文件后，您必须重新启动 Splunk Enterprise。

对自定义命令脚本或 `commands.conf` 文件中现有命令的参数所做的更改不需要重新启动。

另请参阅

控制自定义命令和脚本的访问权限

控制自定义命令和脚本的访问权限

在编写好脚本并将其添加到 commands.conf 之后就可以使用了。

默认情况下，所有角色都有访问 commands.conf 的权限，但只有管理员有写入的权限。这意味着所有角色都可以运行 commands.conf 中列出的命令，除非个别命令的访问控制进行了显式更改。如果您想限制特定角色或用户才能使用某个命令，则必须修改此命令的访问控制。

更改自定义命令的权限

您可以通过**设置**菜单或编辑 default.meta.conf 文件来修改访问控制。

在 Splunk Web 中更改权限

可使用**设置**菜单按用户角色更改某个命令的访问控制。

1.在 Splunk Web 中，选择**设置**，**高级搜索**。

2.单击**搜索命令**。

3.在此搜索命令的**共享**列下，单击**权限**。
这一操作将打开所选搜索命令的**权限**视图。通过此页面指定：

- 此命令应显示于当前应用还是所有应用。
- 哪些角色具有读取和写入此命令的权限。

4.不要忘记保存更改！

在 default.meta.conf 文件中更改权限

您可以通过 $SPLUNK_HOME/etc/apps/<app_name>/metadata/ 目录中的 default.meta.conf 文件来更改某个命令的访问控制。

以下示例显示的是 commands.conf 的默认访问权限，以及输入命令的访问权限（只有管理员才能运行此命令）。

```
[commands]
access = read : [ * ], write : [ admin ]
export = system

[commands/input]
access = read : [ admin ], write : [ admin ]
```

在命令脚本文件中更改访问控制

在命令脚本文件中可以更改访问控制限制。$SPLUNK_HOME/etc/apps/<app_name>/metadata/default.meta 文件中的 [searchscripts] 段落对这些控制进行了定义。默认情况下，这些文件对所有角色和应用都可见，但只有具有文件系统访问的用户，如系统管理员才能编辑文件。

```
[searchscripts]
access = read : [ * ], write : [ admin ]
export = system
```

使用 export = system 属性使文件对系统中的所有应用均可用。在上面的示例中，commands.conf 和 [searchscripts] 的访问权限是全局性的。如果 [searchscripts] 下面没有出现全局导出，则 commands.conf 文件中的脚本配置在所有应用中均可见，但脚本文件本身却不是。

不具有 UI 的应用中的自定义命令应导出到系统，因为无法在本地上下文中运行此命令。

禁用自定义命令

您可以使用**设置**菜单禁用搜索命令，使其无法在应用中运行

1.在 Splunk Web 中，选择**设置**，**高级搜索**。

2.单击**搜索命令**。

搜索命令页面显示一个列出命令的表格和命令相关应用及所有者的信息，并提供选项用于限制权限和禁用命令。

注意：此表格只列出以 Python 编写的搜索命令。

3.在搜索命令的**状态**列下，单击**禁用**。

窗口顶部会出现一个消息横幅，确认此命令已在应用中禁用。

另请参阅

- 自定义命令的安全责任
- *《管理员手册》*中的 default.meta.conf 文件

自定义搜索命令示例

本主题只适用于 **Intersplunk.py 文件**和版本 1 协议。

对于版本 2 协议示例，请参阅"如何在 dev.splunk.com 上使用适用于 Python 的 Splunk SDK 创建自定义搜索命令"。该页面上有几个示例：

- 起始示例
- 基本示例
- 形状示例

此外，Python 还有一些其他的 Splunk SDK 示例。

以下是一个名为 `shape` 的自定义搜索命令示例。`shape` 命令基于事件行计数（高或低）和行长度（窄、宽和非常宽）以及行是否缩进来分类事件。

添加 Python 脚本

将脚本 `shape.py` 添加到相应的应用目录 `$SPLUNK_HOME/etc/apps/<app_name>/bin/` ：

```
import splunk.Intersplunk
def getShape(text):
      description = []
      linecount = text.count("\n") + 1
      if linecount > 10:
          description.append("tall")
      elif linecount > 1:
          description.append("short")
      avglinelen = len(text) / linecount
      if avglinelen > 500:
          description.append("very_wide")
      elif avglinelen > 200:
          description.append("wide")
      elif avglinelen < 80:
          description.append("thin")
      if text.find("\n ") >= 0 or text.find("\n\t") >= 0:
          description.append("indented")
      if len(description) == 0:
          return "normal"
      return "_".join(description)
# get the previous search results
 results,unused1,unused2 = splunk.Intersplunk.getOrganizedResults()
 # for each results, add a 'shape' attribute, calculated from the raw event text
 for result in results:
      result["shape"] = getShape(result["_raw"])
 # output results
 splunk.Intersplunk.outputResults(results)
```

编辑配置文件

编辑位于应用本地目录中的以下配置文件，例如 `$SPLUNK_HOME/etc/app/<app_name>/local`。

1. 在 `commands.conf` 文件中，添加以下段落：
   [shape]
   filename = shape.py
2. 在 `authorize.conf` 文件中，添加以下两个段落：
   [capability::run_script_shape]
   [role_admin]
   run_script_shape= enabled
3. 重新启动 Splunk Enterprise。

运行命令

以下示例说明如何通过 CLI 运行搜索。您也可以在 Splunk Web 中运行命令。

显示多行事件的顶部*形状*：

```
$ splunk search "linecount>1 | shape | top shape"
```

搜索结果以表格形式返回。

```
shape                       count     percent

tall_indented               43        43.000000
short_indented              29        29.000000
tall_thin_indented          15        15.000000
short_thin_indented         10        10.000000
short_thin                  3         3.000000
```

自定义命令的安全责任

作为自定义搜索命令的作者，您有责任在开发自定义搜索命令时遵循最佳安全实践。本主题包含有关编写高质量安全自定义搜索命令的信息。

在 Splunk 软件部署的索引器和搜索头上，自定义搜索命令的运行需要与 splunkd 一样的权限。自定义搜索命令无法在 Sandbox 中运行。所以，您的部署的安全性取决于您的自定义搜索命令的安全性。

验证搜索结果

首先，您应该将来自搜索结果的数据视为不可信的用户输入。搜索结果可能包含任意字符串。如果您不经过转义或不经过程序内部的验证而直接使用这些字符串，则可能会不小心引入安全漏洞。

例如，如果将来自搜索结果且未经验证的字段作为参数传递给 shell 命令，未转义的分号可能会使恶意人员能够在 Splunk 部署中运行任意程序。作为搜索命令的作者，您有责任验证输入，并避免代码插入和路径遍历漏洞，如：

- 代码插入漏洞，如 SQL 插入
- 路径遍历漏洞，如允许用户数据引用任意文件系统路径

使用基于角色的访问控制

默认情况下，所有登录的用户都可以使用自定义搜索命令。按用户角色限制搜索命令的访问权限是一个比较好的做法。

有些搜索命令的编写可能旨在执行有权限的维护操作，如要“清理数据库缓存”的命令。只有具有特殊权限的用户可以使用这些命令。

您可以使用基于角色的访问控制功能来限制搜索命令的运行权限。可以将权限限制给特定用户或角色。

例如，您有一个名为 "launchmissiles" 的自定义搜索命令，而且您希望只有“管理员”角色能使用这个命令。此时，您应执行以下步骤：

1. 在应用程序的 `metadata` 目录中创建 `default.meta` 文件。
2. 在 `default.meta` 文件中，添加以下内容：

   ```
   [commands/launchmissiles]
   access = read : [ admin ], write : [ admin ]
   ```

   `read` 权限指定了谁可以运行自定义搜索命令。以上示例将 `read` 权限限制给 `admin` 角色。您可以将访问权限指定给其他角色，如“高级用户”角色，或您在 `authorize.conf` 文件中定义的新角色。

   将 `write` 权限仅设置给“管理员”角色。其他角色没有 `write` 权限。

请参阅*《确保 Splunk Enterprise 安全》*中的“关于用户和角色”。

避免使用本地文件系统

尽量避免访问或修改文件系统。您编写的所有访问文件系统的代码都可能包含错误，而这些错误会使恶意人员能够泄露 Splunk 部署的数据。错误可使用户从不允许用户搜索的索引中读取数据。

当试图在搜索头群集环境中部署自定义搜索命令时，对文件系统的未受约束的访问可能导致严重问题。例如，如果直接在自定义搜索命令中编辑 `.conf` 文件，这些更改不会复制到搜索头群集的其他成员中。您应始终使用 Splunk REST API 来与 `.conf` 文件或其他知识对象进行交互。如果要永久允许来自自定义搜索命令的数据，使用 REST API 写入 KV 存储或写入 `.conf` 文件。

临时目录

如果必须要使用文件系统，例如要从内存溢出数据至磁盘，则在搜索的 `dispatch` 目录下创建一个临时目录。使用安全目录创建方法，如 Python 中的 `tempfile.mkdtemp()` 函数。Splunk 软件会在搜索过期后自动清除 `dispatch` 目录。

- 对于使用版本 2 协议的自定义搜索命令，Dispatch 目录位于 `dispatch_dir` 属性中，而此属性属于与 Splunk 软件的 `getinfo` 操作一起发送的 `searchinfo` JSON 对象。
- 不要使用 `/tmp`、`$TMPDIR` 或其他类似策略来查找临时目录并写入文件。记住，Windows 中路径的最大长度为 260 个字符。

搜索示例和走查

本部分包含哪些内容？

本部分包括从 Splunk Answers 和字段中收集的实用搜索示例的走查。

- 添加注释到搜索
- 计算动态字段的大小

添加注释到搜索

在搜索字符串中添加注释最灵活的方式是使用内置注释宏。可以在搜索字符串中和单个命令字符串中多次使用宏。搜索的注释不会影响搜索性能。

默认情况下，注释宏仅在“搜索”应用中共享。

使用注释宏

可以使用注释宏在搜索字符串的任何位置添加注释。注释的语法为 `` `comment("comment text")` ``。

示例

```
`comment("THIS IS A COMMENT")`

`comment("This part of the search returns only one value")`
```

注释的开头和结尾是左引号、重音符字符。

添加多个注释到搜索

以下搜索示例根据深度对最近发生的地震进行分类。

```
source=usgs
| eval Description=case(depth<=70, "Shallow", depth>70 AND depth<=300,  "Mid",
  depth>300, "Deep")
| stats count min(mag) max(mag) BY Description
```

如果添加内联注释，则搜索会更易于理解。这是具有多个注释的同一搜索，添加这些注释用以解释搜索各部分。

```
source=usgs `comment("source is the us geological service (usgs)")`
| eval Description=case(depth<=70, "Shallow", depth>70 AND depth<=300, "Mid",
  depth>300, "Deep")
  `comment("Creates field Description. Case function specifies earthquake
  depths, returns Description values - Shallow, Mid, Deep.")`
| stats count min(mag) max(mag) `comment("Counts earthquakes, displays min
  and max magnitudes")` BY Description
```

注释时考虑使用大写字母便于查找。这是带有大写字母注释的同一搜索。

```
source=usgs `comment("SOURCE IS THE US GEOLOGICAL SERVICE (USGS)")`
| eval Description=case(depth<=70, "Shallow", depth>70 AND depth<=300, "Mid",
  depth>300, "Deep")
  `comment("CREATES FIELD DESCRIPTION. CASE FUNCTION SPECIFIES EARTHQUAKE DEPTHS, RETURNS DESCRIPTION VALUES -
SHALLOW, MID, DEEP.")`
| stats count min(mag) max(mag) `comment("COUNTS EARTHQUAKES, DISPLAYS MIN AND MAX MAGNITUDES")` BY Description
```

使用注释排查搜索中的问题

以下搜索示例旨在返回单个索引的字节。但是，此搜索的 `stats` 命令中有一个错误字段 <split-by clause>。

```
index=_internal source=*license* type=usage | stats sum(b) BY index
```

您可以注释搜索的部分以协助识别问题。另一个选项是以详细模式运行此搜索。在此搜索中，`stats` 部分进行了注释。

```
index=_internal source=*license* type=usage `comment("| stats sum(b) BY index")`
```

结果显示了正确的字段名称。您需要将字段名指定为 **idx**，而非 **index**。

```
index=_internal source=*license* type=usage | stats sum(b) BY idx
```

（感谢 Splunk 用户 Runals 提供此示例。）

计算动态字段的大小

此搜索可确定事件中哪些字段占用的磁盘空间最大，无需事先了解字段名称和事件数量。

方案

```
index=_internal earliest=-15m latest=now
| fieldsummary
| rex field=values max_match=0 "value\":\"(?<values>[^\"]*)\","
| mvexpand values
| eval bytes=len(values)
| rex field=field
"^(?!date|punct|host|hostip|index|linecount|source|sourcetype|timeendpos|timestartpos|splunk_server)(?
<FieldName>.*)"
| stats count sum(bytes) as SumOfBytesInField values(values) as Values max(bytes) as MaxFieldLengthInBytes by
FieldName
| rename count as NumOfValuesPerField
| eventstats sum(NumOfValuesPerField) as TotalEvents sum(SumOfBytesInField) as TotalBytes
| eval PercentOfTotalEvents=round(NumOfValuesPerField/TotalEvents*100,2)
| eval PercentOfTotalBytes=round(SumOfBytesInField/TotalBytes*100,2)
| eval ConsumedMB=SumOfBytesInField/1024/1024
| eval TotalMB=TotalBytes/1024/1024
| table FieldName NumOfValuesPerField SumOfBytesInField ConsumedMB PercentOfTotalBytes PercentOfTotalEvents
| addcoltotals labelfield=FieldName label=Totals
| sort - PercentOfTotalEvents
```

“统计”选项卡中显示的结果如下：

FieldName	NumValuesPerField	SumOfBytesInField	ConsumedMB	PercentOfTotal
Totals	1802	45700	0.0436	99.87
cumulative_hits	100	587	0.0006	1.28
eps	100	1862	0.0018	4.07
kb	100	1159	0.0011	2.54
kbps	100	1881	0.0018	4.12
req_time	100	3000	0.0029	6.56
uri	100	10559	0.0101	23.11
uri_query	100	3532	0.0034	7.73
message	96	11633	0.0111	25.46
avg_age	76	280	0.0012	2.80
ev	62	140	0.0001	0.31
average_kbps	59	1071	0.0010	2.34

走查

让我们介绍搜索的各个部分。

1. 本示例首先介绍一个搜索，检索过去 15 分钟内 `index=_internal` 中的所有事件。

   ```
   index=_internal earliest=-15m latest=now
   ```

   注意：可以将此替换为任何搜索字符串和时间范围。

2. 接下来，添加 fieldsummary 命令以创建以前检索的事件中所有字段的摘要。

   ```
   | fieldsummary
   ```

“统计”选项卡中显示的结果如下：

字段	count	distinct_count	is_exact	max	mean	min	numeric_count	stdev	values
abandoned_channels	29	1	1	0.0	0.00	0.0	29	0.00	[{"value":"0
active	29	1	1	0.0	0.00	0.0	29	0.00	[{"value":"0
active_hist_searches	31	2	1	1.0	0.13	0.0	31	0.34	[{"value":"0 {"value":"1"
average_kbps	87	59	1	0.3	0.21	0.0	87	0.15	[{"value":"0 {"value":"0. {"value":"0. {"value":"0. {"value":"0.

3. 使用正则表达式将每个字段的值提取到一个被称为 **values** 的多值字段并进行扩展。以字节为单位计算每个值的长度。

```
| rex field=values max_match=0 "value\":\"(?<values>[^\"]*)\","
| mvexpand values
| eval bytes=len(values)
```

4. 使用另一个正则表达式提取字段的值，但有一些例外。

```
| rex field=field
"^(?!date|punct|host|hostip|index|linecount|source|sourcetype|timeendpos|timestartpos|splunk_server)(?
<FieldName>.*)"
```

5. `stats` 命令用于使用 stats 函数执行多个计算，包括字节计数和总和 (SumOfBytesInField)。值函数用于以多值条目（值）返回 `values` 字段的所有唯一值。最大函数可计算最大字段长度，单位为字节 (MaxFieldLengthInBytes)。根据字段名称组织这些结果。

```
| stats count sum(bytes) as SumOfBytesInField values(values) as Values max(bytes) as MaxFieldLengthInBytes by
FieldName
| rename count as NumOfValuesPerField
```

6. `eventstats` 命令用于计算多个总和，每个字段中的值数量 (TotalEvents) 以及每个字段中的字节总和 (Total Bytes)。

```
| eventstats sum(NumOfValuesPerField) as TotalEvents sum(SumOfBytesInField) as TotalBytes
```

7. 运行多个 `eval` 计算事件总数百分比、字节总数百分比、消耗的兆字节和兆字节总计。

```
| eval PercentOfTotalEvents=round(NumOfValuesPerField/TotalEvents*100,2)
| eval PercentOfTotalBytes=round(SumOfBytesInField/TotalBytes*100,2)
| eval ConsumedMB=SumOfBytesInField/1024/1024
| eval TotalMB=TotalBytes/1024/1024
```

8. `table` 命令用于显示特定的字段集。`addfoltotals` 命令用于计算每列的总计。`sort` 命令用于根据 PercentageOfTotalEvents 字段对列表进行降序排列。

```
| table FieldName NumberOfValuesPerField SumOfBytesInField ConsumedMB PercentageOfTotalBytes
PercentageOfTotalEvents
| addcoltotals labelfield=FieldName label=Totals
| sort - PercentageOfTotalEvents
```

“统计”选项卡中显示的结果如下：

FieldName	NumValuesPerField	SumOfBytesInField	ConsumedMB	PercentOfTotal
Totals	1802	45700	0.0436	99.87
cumulative_hits	100	587	0.0006	1.28
eps	100	1862	0.0018	4.07
kb	100	1159	0.0011	2.54
kbps	100	1881	0.0018	4.12

req_time	100	3000	0.0029	6.56
uri	100	10559	0.0101	23.11
uri_query	100	3532	0.0034	7.73
message	96	11633	0.0111	25.46
avg_age	76	280	0.0012	2.80
ev	62	140	0.0001	0.31
average_kbps	59	1071	0.0010	2.34