



# Splunk<sup>®</sup> Enterprise 6.5.0

## 管理员手册

生成时间：2016 年 9 月 26 日，下午 10:19

# Table of Contents

<b>欢迎使用 Splunk Enterprise 管理</b>	<b>6</b>
如何使用本手册	6
Splunk 平台管理：更多内容	6
Splunk 平台管理员的其他手册	8
Windows 管理员简介	9
关于 Splunk Free	10
在 *nix 和 Windows 上运行 Splunk 的差异	11
配置 Splunk 的方法	12
<b>在 Windows 上充分利用 Splunk Enterprise</b>	<b>13</b>
在 Windows 上部署 Splunk	13
优化 Splunk 以获得高性能	14
在系统映像上加入 Splunk	15
将通用转发器集成到系统映像中	16
将完整 Splunk 集成到系统映像中	16
<b>使用 Splunk Web 管理 Splunk Enterprise</b>	<b>17</b>
启动 Splunk Web	17
Splunk Web 管理任务	17
Splunk Enterprise 默认仪表板	18
自定义 Splunk Web 横幅消息	19
配合使用 Splunk Web 与代理服务器	19
<b>使用配置文件管理 Splunk Enterprise</b>	<b>19</b>
关于配置文件	19
配置文件目录	20
配置文件结构	21
配置文件优先顺序	22
单个 props.conf 文件中的属性优先顺序	26
如何编辑配置文件	27
更改配置文件之后何时重新启动 Splunk Enterprise	28
配置文件列表	30
配置参数和数据管道	31
备份配置信息	34
检查 Splunk 软件文件完整性	34
<b>使用命令行界面 (CLI) 管理 Splunk Enterprise</b>	<b>35</b>
关于 CLI	35
获取 CLI 相关帮助	36
CLI 管理命令	39
使用 CLI 来管理远程 Splunk Enterprise 实例	43
自定义 CLI 登录横幅	44
<b>启动 Splunk Enterprise 并执行初始任务</b>	<b>45</b>
启动和停止 Splunk Enterprise	45
配置 Splunk 在开机时启动	47
安装您的许可证	48

更改默认值	49
将 Splunk 绑定到某个 IP	53
配置 Splunk 以使用 IPv6	53
确保配置安全	55
共享性能数据	55
<b>配置 Splunk 许可证</b>	<b>59</b>
Splunk Enterprise 许可授权如何运作	59
Splunk 软件许可证类型	60
组、堆叠、池和其他术语	62
安装许可证	63
配置许可证主服务器	64
配置许可证从服务器	65
创建或编辑许可证池	65
向许可证池添加索引器	67
从 CLI 管理许可证	67
<b>管理 Splunk 许可证</b>	<b>69</b>
管理许可证	69
关于许可证违规	69
交换许可证主服务器	71
<b>许可证使用情况报表视图</b>	<b>71</b>
关于 Splunk Enterprise 的许可证使用情况报表视图	71
使用“许可证使用情况报表视图”	73
<b>管理应用键值存储</b>	<b>74</b>
有关应用键值存储	74
重新同步 KV 存储	75
备份 KV 存储	75
<b>认识 Splunk 应用</b>	<b>76</b>
应用和加载项	76
搜索和报表应用	77
配置在应用中打开的 Splunk Web	77
从哪里获得更多应用和加载项	78
应用部署概述	79
应用架构和对象所有权	80
管理应用和加载项对象	81
管理应用和加载项配置及属性	82
<b>认识 Hunk</b>	<b>83</b>
认识 Hunk	83
<b>管理用户</b>	<b>84</b>
关于用户和角色	84
配置用户语言和区域设置	85
配置用户会话超时	86
<b>配置文件参考</b>	<b>86</b>
alert_actions.conf	86

app.conf	94
audit.conf	99
authentication.conf	101
authorize.conf	118
collections.conf	128
commands.conf	130
crawl.conf	135
datamodels.conf	137
datatypesbnf.conf	140
default.meta.conf	141
default-mode.conf	142
deployment.conf	143
deploymentclient.conf	143
distsearch.conf	147
eventdiscoverer.conf	154
event_renderers.conf	155
eventtypes.conf	157
fields.conf	158
indexes.conf	160
inputs.conf	184
instance.cfg.conf	223
limits.conf	224
literals.conf	262
macros.conf	263
multikv.conf	265
outputs.conf	267
passwords.conf	281
pdf_server.conf	282
procmon-filters.conf	285
props.conf	286
pubsub.conf	307
restmap.conf	309
savedsearches.conf	315
searchbnf.conf	329
segmenters.conf	331
server.conf	333
serverclass.conf	377
serverclass.seed.xml.conf	384
setup.xml.conf	386
source-classifier.conf	389
sourcetypes.conf	389
splunk-launch.conf	391
tags.conf	393
telemetry.conf	394
times.conf	396
transactiontypes.conf	398
transforms.conf	401
ui-prefs.conf	411
ui-tour.conf	413
user-prefs.conf	415
user-seed.conf	417
viewstates.conf	418
visualizations.conf	420

web.conf	421
wmi.conf	437
workflow_actions.conf	441

# 欢迎使用 Splunk Enterprise 管理

## 如何使用本手册

本手册提供不同 Splunk 管理方法的相关信息。它还为您介绍 Windows 和 \*nix 的一些初始管理任务。

**注意：**除非另有说明，否则本手册中的任务和过程对 Windows 和 \*nix 操作系统均适用。

有关 Splunk 管理过程更多内容的概述，包括本手册中未介绍的任务（如设置用户或数据以及安全性配置），请参阅本手册中的[“Splunk 管理：更多内容”](#)。

有关可供 Splunk 用户使用的其他手册的列表和简单描述，请参阅[“Splunk 管理员的其他手册”](#)。

### 使用《管理员手册》可以执行的操作

任务：	查看此处：
启动 Splunk 并进行一些初始配置	开始使用 Splunk 所需执行的全部操作，从启动 Splunk 和安装许可证到将 Splunk 绑定至 IP。有关更多信息，请参阅： <a href="#">“如何开始”</a> 。
使用 Splunk Web 配置和管理 Splunk	Splunk Web 的概述以及如何使用它来管理 Splunk。有关更多信息，请参阅 <a href="#">“使用 Splunk Web”</a> 。
使用配置文件配置和管理 Splunk	在何处能够找到配置文件、如何创建和编辑它们，以及关于文件优先顺序的一些重要内容。请参阅 <a href="#">“关于配置文件”</a> 开始操作。
使用 Splunk 命令行界面 (CLI) 配置和管理 Splunk	如何使用命令行界面配置 Splunk 的概述。有关更多信息，请参阅 <a href="#">“关于 CLI”</a> 。
在 Windows 上优化 Splunk	使用 Splunk 时您应了解的一些 Windows 特定的事项，包括最佳部署的一些提示以及使用系统映像的相关信息。有关更多信息，请参阅 <a href="#">“Windows 管理员简介”</a> 。
了解 Splunk 许可证	<a href="#">安装许可证</a> ，然后转到此处了解有关 Splunk 许可证的所有必要信息： <a href="#">“管理 Splunk 许可证”</a> 以获得更多信息。
熟悉 Splunk 应用	Splunk 应用的简介和概述以及如何将其集成到 Splunk 配置中。有关更多信息，请参阅 <a href="#">“认识 Splunk 应用”</a> 。
管理用户设置	<a href="#">管理用户</a> 这一章向您介绍如何管理用户设置。  有关创建用户的更多信息，请参阅《确保 Splunk Enterprise 安全》手册中用户和基于角色的访问控制。

## Splunk 平台管理：更多内容

[《管理员手册》](#)提供初始管理任务以及可用于管理 Splunk 软件的不同方法的相关信息。有关如何使用《管理员手册》的更多特定概述，请参阅[如何使用本手册](#)。

下面是初始配置之后您可能要执行的管理任务以及在何处了解更多信息。

任务：	查看此处：
执行备份	<a href="#">备份配置信息</a> 备份索引数据 设置退休和归档策略
定义告警	<a href="#">《告警手册》</a>
管理搜索任务	管理搜索任务

有关更多管理帮助的信息，请参阅下方介绍的手册。

### 安装和升级 Splunk Enterprise

《安装手册》介绍如何安装和升级 Splunk Enterprise。要获得特定任务相关信息，先从此处开始。

任务：	查看此处：
了解安装要求	计划您的安装

预估硬件容量需求	预估硬件需求
安装 Splunk	在 Windows 上安装 Splunk Enterprise 在 Unix、Linux 或 MacOS 上安装 Splunk Enterprise
升级 Splunk Enterprise	从早期版本升级

## 数据导入

“数据导入”为您提供数据导入的信息：如何使用来自外部来源的数据，以及如何增强数据价值。

任务：	查看此处：
了解如何使用外部数据	如何将数据导入 Splunk
配置文件和目录输入	获取文件和目录的数据
配置网络输入	获取网络事件
配置 Windows 输入	获取 Windows 数据
配置其他输入	其他数据导入方式
增强您的数据值	配置事件处理 配置时间戳 配置索引字段提取 配置主机值 配置来源类型 管理事件分段 使用查找和工作流动作
查看您的数据在建立索引后的显示效果	预览您的数据
过程改善	改善数据输入过程

## 管理索引和索引器

“管理索引器和群集”告诉您如何配置索引。它还介绍了如何管理维护索引的组件：索引器和索引器群集。

任务：	查看此处：
了解索引	索引概述
管理索引	管理索引
管理索引存储	管理索引存储
备份索引	备份索引数据
归档索引	设置退休和归档策略
了解群集和索引复制	关于群集和索引复制
部署群集	部署群集
配置群集	配置群集
管理群集	管理群集
了解群集架构	群集如何工作

## 调整 Splunk 平台部署

《分布式部署手册》介绍如何跨多个组件（例如，转发器、索引器和搜索头）来分布 Splunk 平台功能。关联手册详细介绍分布式组件：

- 《转发数据手册》介绍转发器。
- 《分布式搜索手册》介绍搜索头。
- 《更新 Splunk 组件手册》阐述如何使用部署服务器和转发器管理来管理您的部署。

任务：	查看此处：
了解分布式 Splunk 平台部署	调整部署规模
针对 Splunk 平台部署执行容量规划	预估硬件需求
了解如何转发数据	转发数据

跨多个索引器进行分布式搜索	跨多个索引器搜索
更新部署	在您的环境中部署配置更新

## 确保 Splunk Enterprise 安全

“确保 Splunk 安全”介绍如何确保您的 Splunk Enterprise 部署的安全。

任务：	查看此处：
验证用户和编辑角色	用户和基于角色的访问控制
使用 SSL 确保数据安全	安全验证和加密
审计 Splunk 软件	审计系统活动
与 Splunk 软件配合使用单一登录 (SSO)	配置单点登录
将 Splunk 软件与 LDAP 配合使用	设置使用 LDAP 进行的用户验证

## Splunk 软件故障排除

《故障排除手册》提供有关 Splunk 平台故障排除的总体指导。此外，在其他手册的相关主题中还提供了针对特定问题的故障排除信息。

任务：	查看此处：
了解 Splunk 平台故障排除工具	初始步骤
了解 Splunk 日志文件	Splunk 日志文件
使用 Splunk 支持	联系 Splunk 支持
解决常见问题	一些常用方案

## 参考和其他信息

Splunk 文档会包含一些有用的参考，以及其他一些可能对 Splunk 软件管理员有帮助的信息来源。

参考：	查看此处：
配置文件参考	《管理员手册》中的 <a href="#">配置文件参考</a>
REST API 参考	《REST API 参考手册》
CLI 帮助	在 Splunk Enterprise 的安装实例中提供。有关如何调用它的详细信息，请阅读《管理员手册》中的 <a href="#">获取 CLI 相关帮助</a> 。
版本信息	发行说明
管理 Splunk 平台知识对象的相关信息	知识管理器手册

## Splunk 平台管理员的其他手册

该《[管理员手册](#)》是专为 Splunk Enterprise 管理员提供重要信息和程序的几本书籍中的其中一本。但它仅介绍您使用 Splunk Enterprise 可以执行的一些基本操作。

如果您需要配置、运行和维护 Splunk Enterprise，将其作为服务提供给您自己或其他用户使用，请先阅读这本手册。然后，您可以阅读其他手册以获得有关 Splunk Enterprise 管理特定领域的详细信息。

手册	涵盖内容	重要主题领域
数据导入	指定数据导入和改善 Splunk 软件数据处理方式	如何将数据导入 Splunk 配置事件处理 预览您的数据
管理索引器和群集	管理 Splunk 索引器和索引器群集	关于索引和索引器 管理索引 备份和归档您的索引 关于群集和索引复制 部署群集
分布式部署	扩展部署以适应您所在企业的需求。	分布式 Splunk 概述



<b>转发数据</b>	将数据转发到 Splunk 中。	<b>转发数据</b>
<b>分布式搜索</b>	使用搜索头跨多个索引器进行分布式搜索。	<b>跨多个索引器搜索</b>
<b>更新 Splunk 组件</b>	使用部署服务器和转发器管理来更新 Splunk 组件，如转发器和索引器。	<b>在您的环境中部署更新</b>
<b>确保 Splunk 安全</b>	数据安全和用户验证	<b>用户验证和角色使用 SSL 加密和验证审计</b>
<b>监视 Splunk Enterprise</b>	用包含的仪表板和告警监视 Splunk Enterprise 部署并排除故障	<b>关于监视控制台</b>
<b>故障排除</b>	解决问题	<b>初始步骤 Splunk 日志文件 一些常用方案</b>
<b>安装</b>	安装和升级 Splunk	<b>系统要求 分步安装程序 从早期版本升级</b>

主题“[学习管理 Splunk](#)”会提供有关从哪里阅读特定管理任务的更详细指导信息。

## Splunk 管理员感兴趣的其他书籍

除了介绍主要管理任务的手册以外，您有时可能需要阅读其他手册，具体取决于 Splunk Enterprise 安装大小和您的职责范围。这些是 Splunk Enterprise 核心文档集的其他手册：

- **搜索教程。**该手册向您介绍如何使用 Splunk 进行搜索。
- **知识管理器。**该手册会介绍如何管理 Splunk 知识对象，例如事件类型、标记、查找、字段提取、工作流动作、保存的搜索和视图。
- **告警。**该手册介绍 Splunk 的告警和监视功能。
- **数据可视化。**该手册介绍 Splunk 提供的一系列可视化。
- **搜索手册。**该手册告诉您如何搜索和使用 Splunk 搜索语言。
- **搜索参考。**该参考包含 Splunk 搜索命令的详细目录。
- **开发用于 Splunk Web 的视图和应用。**该手册介绍如何使用高级 XML 来开发视图和应用。它还包含其他开发人员主题，例如自定义脚本和扩展 Splunk。
- **REST API 参考。**该手册提供了所有可公开访问的 REST API 端点信息。
- **发行说明。**在这里可以找到有关新功能、已知问题和已修复问题的信息。

## 更详细的 Splunk 文档

要获得全部 Splunk Enterprise 文档链接，包括以上列出的手册，请访问：[Splunk Enterprise 文档](#)。

要访问所有 Splunk 文档，包括应用的手册，请前往此页面：[欢迎使用 Splunk 文档](#)。

## 制作 PDF

如果您需要本手册的 PDF 版本，请单击本页左侧目录下方的红色链接[将管理员手册下载为 PDF](#)。即会为您动态生成手册的 PDF 版本。您可以将其保存或打印出来以便之后阅读。

## Windows 管理员简介

欢迎使用！

Splunk 是一款功能强大、高效的工具，它可以帮助 Windows 管理员解决在 Windows 网络上出现的问题。它提供了“即装即用”的功能集，使其成为 Windows 管理员工具箱内的秘密武器。它可以通过添加应用来增加自身功能，因此使其具有更高的可扩展性。此外，它还拥有一个不断扩大、兴旺的用户社区。

## Windows 用户如何使用本手册

本手册包括多个主题，以帮助您试验、学习、部署和充分利用 Splunk。

除非另外指定，否则本手册中的信息对 Windows 和 \*nix 用户均有所帮助。如果您不熟悉 Windows 或 \*nix 操作命令，强烈建议您查阅在[\\*nix 和 Windows 上运行 Splunk 的差异](#)。

在“在 Windows 上充分利用 Splunk”一章中，我们还提供了一些额外信息。本章适用于 Windows 用户，可帮助您充分利用 Splunk，其中包括下列信息。

在[Windows 上部署 Splunk](#)提供专用于 Windows 用户的一些注意事项和准备工作。本主题在计划部署时使用。

[优化 Splunk 以获得高性能](#) 介绍在部署期间或部署完成之后确保 Splunk 能够在 Windows 部署上正常运行的方法。

[在系统映像上加入 Splunk](#) 帮助您使 Splunk 成为每个 Windows 系统映像或安装过程的一部分。在此处，您可以找到用于将 Splunk 和 Splunk 转发器安装到系统映像上的任务。

## 相关信息

以下是其他 Splunk 手册中所涉及的一些其他 Windows 主题：

- 有关所有已安装的 Splunk for Windows 服务的概述（来自《安装手册》）
- Splunk 可以监视什么（来自《数据导入手册》）
- 有关确定如何监视远程 Windows 数据的注意事项（来自《数据导入手册》）。阅读该主题以获得有关如何从多个计算机远程获取数据的重要信息。
- 合并来自多个主机的数据（来自《通用转发器手册》）

其他有用的信息：

- 我的数据位于何处？（来自《数据导入手册》）
- 使用 Splunk 的命令行界面 (CLI)（来自《数据导入手册》）
- 来源、来源类型和字段（来自《数据导入手册》）
- 字段和字段提取（来自《知识管理器手册》）
- 实时搜索（来自《用户手册》）
- 保存的搜索（来自《用户手册》）
- 仪表板创建（来自《用户手册》）

## 当您需要帮助时

如果您打算深入了解 Splunk 知识，我们提供有大量的教育课程。

如果您在使用过程中遭遇困难，Splunk 拥有庞大免费的支持基础设施向您提供帮助：

- Splunk Answers。
- Splunk 维基社区。
- Splunk Internet Relay Chat (IRC) 通道 (EFNet #splunk)。（需要 IRC 客户端）

如果您的问题仍然未能解决，您可以联系 Splunk 的支持团队。在“联系支持”页面上会提供具体的做法。

**注意：**社区级别以上的支持级别需要具备 Enterprise 许可证。要获得此许可证，您需要联系我们的销售团队。

## 关于 Splunk Free

Splunk Free 是完全免费的 Splunk 版本。Free 许可证使您可以每天最多对 500 MB 数据建立索引而不会过期。

500 MB 的限制指每天您可以添加（我们称之为索引）的新数据，但您可以每天添加数据，想要存储多少就存储多少。例如，您可以每天添加 500 MB 数据，并逐渐地在 Splunk Enterprise 中存储 10 TB 数据。

如果您每天需要多于 500 MB 的数据，则需要购买 Enterprise 许可证。有关许可授权的更多信息，请参阅 [Splunk 许可授权如何运作](#)。

Splunk Free 通过追踪 [许可证违规](#) 来规范您的许可证使用情况。如果您在 30 天周期内有 3 次超过 500 MB/天，Splunk Free 会继续对您的数据建立索引，但搜索功能会禁用，直到您在 30 天周期内得到的警告次数下降到 3 次或更少。

## Splunk Free 是否适合您？

Splunk Free 专门供个人用户对 IT 数据执行特殊搜索和可视化。您可以持续使用 Splunk Free 来对少量数据（< 500 MB/天）建立索引。此外，您可以使用它来短期批量加载和分析大型数据集 - Splunk Free 使您可以在 30 天周期内最多批量加载 3 次大型数据集。这对于大型数据集的取证分析非常有用。

## Splunk Free 中包含什么？

Splunk Free 是一款单用户产品。它支持 Splunk Enterprise 的所有功能，例外情况如下：

- 分布式搜索配置（包括搜索头群集化）不可用。
- 不能以 TCP/HTTP 格式转发。这意味着您可以向其他 Splunk 平台实例转发数据，但不能向非 Splunk 软件转发。
- 部署管理功能不可用。
- 告警（监视）不可用。
- 索引器群集化不可用。
- 报表加速摘要不可用。
- 尽管 Splunk Free 实例可以用作转发器（转发到 Splunk Enterprise 索引器），但它不能作为部署服务器的客户端。

- 当使用 Splunk Free 时无验证或用户以及角色管理。也就是说：
  - 不存在登录机制。不会提示您输入用户名/密码，通过命令行或浏览器可以访问和控制 Splunk Free 的所有方面。
  - 所有访问均被视为与管理用户同等。只有一个角色（管理员），并且无法配置。您无法添加更多角色或创建用户帐户。
  - 在运行搜索时将针对所有公共索引 'index=\*'。
  - 不支持如用户配额、每个搜索时间范围最大值和搜索过滤条件等搜索限制。
  - 功能系统被禁用。为所有访问 Splunk Free 的用户启用全部可用操作。

## 从 Enterprise Trial 许可证切换到 Free

当您首次下载并安装 Splunk 时，您将自动采用 Enterprise Trial 许可证。您可以继续使用 Enterprise Trial 许可证，直到它过期，也可以立即切换到 Free 许可证，这都取决于您的需求。

### 切换到 Free 应了解的注意事项

Splunk Enterprise Trial 为您提供了许多在 Splunk Free 中不可用的功能。当您切换到 Free 时，**应注意以下方面**：

- 您创建的用户帐户或角色将不再起作用。
- 任何连接到实例的用户将自动以 admin 身份登录。您将可以看到更新检查，但不再显示登录屏幕。
- 任何由 admin 以外用户创建且未全局共享的知识对象（如事件类型、交易或来源类型定义）将不再可用。如果您需要在切换到 Splunk Free 之后继续使用这些知识对象，可采取以下做法之一：
  - 在切换之前使用 Splunk Web 将它们提升为全局可用。请参阅[管理应用和加载项对象](#)。
  - 手动编辑它们所在的配置文件以提升它们。请参阅[应用架构和对象所有权](#)。
- 您所定义的任何告警将不再触发。您将**不再从 Splunk 软件接收告警**您仍然可以计划搜索运行以达到仪表板和摘要索引的目的。
- outputs.conf 中以 TCP 或 HTTP 格式转发到第三方应用程序的配置将停止工作。

如果在使用 Enterprise Trial 许可证时尝试在 Splunk Web 中进行任何上述配置，您将收到上述 Splunk Free 限制。

## 如何切换到 Splunk Free？

如果您目前有 Splunk Enterprise（试用版或非试用版），您可以等待 Enterprise 许可证到期，也可以随时切换到 Free 许可证。要切换到 Free 许可证：

1. 以具有管理员权限的用户身份登录到 Splunk Web，然后导航至**设置 > 许可授权**。
2. 单击页面顶部的**更改许可证组**。



3. 选择 **Free 许可证**，然后单击**保存**。
4. 将提示您重新启动。

## 在 \*nix 和 Windows 上运行 Splunk 的差异

本主题将阐明当 Splunk 运行的状况下，您在 \*nix 和 Windows 操作系统上会遇到的功能差异。这里不会深入探讨到技术性的比较，也不会鼓吹或偏爱任何操作系统，而是旨在说明在不同操作系统特定的 Splunk 手册页面上为何内容有所不同。

### 路径

在 \*nix 操作系统处理文件和目录的方式上的主要差异，就是在路径名中用于分隔文件或目录的斜线类型有所不同。  
\*nix 系统使用正斜线 ("/")。另一方面，Windows 使用反斜线 ("\")。

\*nix 路径示例：

```
/opt/splunk/bin/splunkd
```

Windows 路径示例：

```
C:\Program Files\Splunk\bin\splunkd.exe
```

## 环境变量

另一项差异是这两种操作系统在环境变量表示上有所不同。两种操作系统均采用各自的方法暂时存储在一个或多个环境变量中的数据。在 \*nix 系统上，这是通过在环境变量名称之前加上美元符号("\$") 来表示，例如：

```
# SPLUNK_HOME=/opt/splunk; export $SPLUNK_HOME
```

在 Windows 上则稍微有点不同——您需要使用百分号("%") 指定环境变量。根据您使用的环境变量类型，您可能需要将一个或两个百分号放在环境变量名称之前或名称的两侧。

```
> set SPLUNK_HOME="C:\Program Files\Splunk"
> echo %SPLUNK_HOME%
C:\Program Files\Splunk
>
```

要在 Windows 环境中设置 %SPLUNK\_HOME% 变量，您可以采用以下两种方法之一：

- 编辑位于 %SPLUNK\_HOME%\etc 中的 `splunk-launch.conf`。
- 通过访问“环境变量”窗口来设置变量。打开“资源管理器”窗口，在左窗格中右键单击“我的电脑”，然后从显示的窗口中选择“属性”。在出现“系统属性”窗口之后，选择“高级”选项卡，然后单击标签窗口底部的“环境变量”按钮。

## 配置文件

Splunk Enterprise 与使用 ASCII/UTF-8 字符集编码的配置文件结合使用。在 Windows 中编辑配置文件时，将文本编辑器配置为利用此编码写文件。在一些 Windows 版本中，UTF-8 不是默认字符集编码。请参阅[如何编辑配置文件](#)。

## 配置 Splunk 的方法

Splunk 在一组**配置文件**中维护配置信息。您可以使用以下任一（或全部）方法配置 Splunk：

- 使用 Splunk Web。
- 使用 Splunk 的命令行界面 (CLI) 命令。
- 直接编辑 Splunk 的配置文件。
- 使用通过 Splunk REST API 来更新配置的应用设置屏幕。

上述所有方法都更改基本配置文件的内容。在不同情况下，您可能发现不同的便利方法。

### 使用 Splunk Web

您可以在 Splunk Web 中执行大多数常用配置任务。默认情况下，Splunk Web 在安装该 Web 的主机的端口 8000 上运行：

- 如果在本地计算机上运行 Splunk，则访问 Splunk Web 的 URL 是 `http://localhost:8000`。
- 如果在远程计算机上运行 Splunk，则访问 Splunk Web 的 URL 是 `http://<hostname>:8000`，其中 <hostname> 是运行 Splunk 的计算机的名称。

管理菜单可在 Splunk Web 菜单栏中的**设置**下找到。Splunk 文档集中介绍的大多数任务均针对 Splunk Web。有关 Splunk Web 的更多信息，请参阅[认识 Splunk Web](#)。

### 编辑配置文件

大多数 Splunk 配置信息存储在 .conf 文件中。这些文件位于 Splunk 安装目录（在文档中通常称为 \$SPLUNK\_HOME）下的 `/etc/system` 下。在大多数情况下，可将这些文件复制到本地目录并使用首选的文件编辑器对其进行更改。

在开始编辑配置文件之前，请阅读[“关于配置文件”](#)。

### 使用 Splunk CLI

许多配置选项可通过 CLI 提供。这些选项记录在本手册的 CLI 章节中。也可以在 Splunk 运行时使用 `help` 命令来获取 CLI 帮助参考：

`./splunk help`

有关 CLI 的更多信息，请参阅本手册中的“关于 CLI”。如果您不熟悉 CLI 命令或在 Windows 环境下工作，也应查阅[在 \\*nix 和 Windows 上运行 Splunk 的差异](#)。

## 应用的设置屏幕

开发人员可以创建应用的设置屏幕，允许用户设置此应用的配置，而不必直接编辑配置文件。利用设置屏幕，可以更为轻松地将应用分布到不同的环境，或者根据特定使用情况自定义应用。

设置屏幕使用 Splunk 的 REST API 来管理应用的配置文件。

有关设置屏幕的更多信息，请参阅 Splunk 开发人员门户中的“为 Splunk 应用创建设置页面”。

## 管理分布式环境

Splunk 部署服务器为分布式环境提供集中管理和配置。您可以使用它将配置文件集或其他内容部署到覆盖整个企业的 Splunk 实例组。

有关管理部署的信息，请参阅“更新 Splunk 组件”手册。

# 在 Windows 上充分利用 Splunk Enterprise

## 在 Windows 上部署 Splunk

您可以通过多种方法来将 Splunk 集成到您的 Windows 环境中。本主题介绍其中一些方案，并提供了有关如何确保 Splunk for Windows 部署适应您的企业环境的指南。

本主题更多侧重于在 Windows 环境中部署 Splunk，即使您将其集成到 Windows 企业环境中，Splunk 本身也具有需要您注意的分布式部署操作。《分布式部署手册》包含有关在多台计算机上分散运行 Splunk 服务的大量信息。

当大规模地在 Windows 上部署 Splunk 时，您可以完全依赖您自己的部署实用工具（如 System Center Configuration Manager 或 Tivoli/BigFix）来在企业内的计算机上部署 Splunk 及其配置。或者，您也可以将 Splunk 集成到系统映像中，然后通过 Splunk 的部署服务器来部署 Splunk 配置和应用。

## 概念

当您部署 Splunk 到您的 Windows 网络中时，它会捕获来自计算机的数据并集中存储。一旦数据就位之后，您就可以针对已建立索引的数据来执行搜索和创建报告与仪表板等。对于系统管理员而言，更重要的是，当数据到达时 Splunk 可以发送告警以通知您发生了什么。

在典型的部署中，您可以将某些硬件专门用于 Splunk 建立索引，然后使用通用转发器与 Windows Management Instrumentation (WMI) 的组合集合来自企业内其他计算机的数据。

## 注意事项

在 Windows 企业环境中部署 Splunk 需要大量的规划步骤。

首先，您必须对您的企业环境进行清点，从物理网络开始，然后确定该网络上不同计算机的单独配置情况。其中包括但不限于：

- 统计您的环境中的计算机数量，并确定其中需要安装 Splunk 的子集。这将定义您的 Splunk 拓扑结构的初始框架。
- 计算您的网络带宽，包括主要站点和任何远程或外部站点的带宽。这将确定您需要在哪里安装您的主 Splunk 实例，以及在哪里和如何使用 Splunk 转发器。
- 评估您的网络当前的运行状况，尤其是网络分隔区域。确保您的边缘路由器和交换机工作正常，以便设定部署期间和之后的网络性能基准。

然后，您必须在开始部署之前回答众多问题，其中包括：

- **在您的计算机上哪些数据需要建立索引？您需要针对其中哪部分数据执行搜索、报告或告警？**这可能是最需要认真考虑的重要因素。通过回答这些问题，您将确定如何解决其他需要注意的每个问题。它可以确定在哪里安装 Splunk，以及您在这些安装中使用哪种类型的 Splunk。它还可以确定 Splunk 可能使用多少计算能力和网络带宽。
- **网络布局如何？所有外部站点的链路配置如何？这些链路使用哪种安全性？**充分了解您的网络拓扑结构，有助于确定您需要在哪些计算机上安装 Splunk，以及从网络的立场决定应该要在这些计算机上安装哪种类型的 Splunk（索引器或转发器）。

对于 LAN 或 WAN 链路较为薄弱的站点，有必要考虑在不同站点之间传输多大的 Splunk 数据量。例如，如果您具



有轴辐式网络，即一个中心站点连接到多个分支站点，则最好在分支站点的计算机上部署转发器，以向每个分支站点内的一个中间转发器发送数据。然后，中间转发器会将数据发回到中央站点。这与让分支站点内所有计算机都向中央站点的索引器发送数据的做法相比成本更低。

如果外部站点具有文件、打印或数据库服务，则您还需要考虑这些流量。

- **您 Active Directory (AD) 是如何配置的？** 在您的域控制器 (DC) 上操作主机角色是如何定义的？所有域控制器是否位于中央站点，或者是否有控制器位于卫星站点？如果您的 AD 为分布式结构，您的桥头服务器是否配置正确？您的站点间拓扑生成器 (ISTG) 角色的服务器是否工作正常？如果您在运行 Windows Server 2008 R2，那么在分支站点内是否设有只读域控制器 (RODC)？如果有，则需要考虑 AD 复制流量以及 Splunk 和其他网络流量的影响。
- **您的网络中的服务器还扮演其他哪些角色？** Splunk 索引器需要资源来保持高性能运行，如果与其他耗费资源型应用程序或服务（如 Microsoft Exchange、SQL Server 乃至 Active Directory 本身）共享服务器，则在这些计算机上 Splunk 可能会出现性能问题。有关与 Splunk 索引器共享服务器资源的其他信息，请参阅《容量规划手册》中的“针对 Splunk Enterprise 容量规划的介绍”。
- **您如何向您的用户告知部署情况？** Splunk 安装意味着环境的改变。根据 Splunk 的部署方式，某些计算机将会安装新的软件。用户可能会误将这些新安装的软件与他们在其计算机上已知的问题或速度变慢相关联。您应保持在任何变动时通知您的用户，以减少部署相关的支持请求。

## 准备将 Splunk 部署在 Windows 上

如何将 Splunk 部署到您的当前环境中，取决于您对 Splunk 的需求（并与可用计算资源保持平衡）、您的物理和网络布局，以及您的企业基础设施。由于并不存在一种特定的 Splunk 部署方法，因此也没有可供遵循的分步说明。不过，您可以遵循一些通用指南。

要成功地部署 Splunk，您需要：

- **准备您的网络。** 在将 Splunk 集成到您的环境中之前：
  - 确保您的网络工作正常，所有开关、路由器和线缆配置正确。
  - 交换任何损坏或故障设备。
  - 确保所有虚拟 LAN (VLAN) 设置正确。
  - 测试网络吞吐量，尤其是网络链路薄弱的站点之间的网络吞吐量。
- **准备您的 Active Directory。** 尽管 AD 并不是运行 Splunk 所必需的，但最好在部署之前确保其工作正常。其中包括但不限于：
  - 确定您的所有域控制器，以及它们可能执行的操作主机角色。如果在分支站点内设有 RODC，则应确保它们与操作主机 DC 之间具有目前最快的连接。
  - 确保 AD 复制工作正常，并且所有站点链路具有一个包含全局目录副本的 DC。
  - 如果您的林划分为多个站点，则应确保您的 ISTG 角色服务器工作正常，或者在您的站点内至少分配两个桥头服务器（主服务器和备份服务器）。
  - 确保您的 DNS 基础设施工作正常。

您可能需要将 DC 放在您的网络的不同子网上，并在部署过程中根据需要捕获灵活单一主机操作 (FSMO 或操作主机) 角色，以确保最高的 AD 操作和复制性能。

- **定义您的 Splunk 部署。** 在您的 Windows 网络准备就绪之后，接下来您必需决定将 Splunk 部署到网络中的哪个位置。考虑以下方面：
  - 确定您需要 Splunk 在每台计算机上为其建立索引的数据集，以及您是否需要 Splunk 针对任何收集到的数据发出告警。
  - 如可行，在每个网路段中专门指定一台或多台计算机来处理 Splunk 索引操作。有关分布式 Splunk 部署的容量规划的其他信息，请阅读《容量规划手册》中的“针对 Splunk Enterprise 容量规划的介绍”。
  - 不要在运行耗费资源型服务（例如：AD，尤其是执行 FSMO 角色的 DC；任何版本的 Exchange、SQL Server；Hyper-V 或 VMWare 等计算机可视化产品）的计算机上安装整个 Splunk。相反，应使用通用转发器，或通过 WMI 连接到这些计算机。
  - 如果您正在运行 Windows Server 2008/2008 R2 Core，当您在这些计算机上安装 Splunk 时，请记住您将无法在 Splunk Web 上通过 GUI 来做出更改。
  - 适当安排您的 Splunk 布局，以确保尽量使用最小资源网络，尤其是对于较为薄弱的 WAN 链路。通用转发器可以显著地减少在网络上发送的 Splunk 相关流量。
- **将您的部署计划告知您的用户。** 在整个过程中对您的用户进行部署状态的建议是非常重要的。这将显著地减少您将来收到的支持请求数量。

## 优化 Splunk 以获得高性能

与许多服务一样，在 Windows 上的 Splunk 也需要适当的维护，以便保持高性能运行。本主题介绍您可以在部署期间或之后采用哪些方法来确保在 Windows 上的 Splunk 能够正常运行。

要确保 Splunk 高性能运行：

- **指定一台或多台计算机来运行 Splunk。** Splunk 具有水平或横向伸缩性。这意味着通过增加运行 Splunk 的计算机台数，而不是增加单台计算机的资源，可以获得更高的性能。如可行，应拆分建立索引和搜索活动并

在多台计算机上运行，在这些计算机上只运行主要的 Splunk 服务。除了通用转发器之外，如果在运行与其他服务共享的服务器上运行 Splunk，则性能会降低。

- **针对您的 Splunk 索引使用专用高速磁盘。**用于 Splunk 建立索引的系统可用磁盘越快，Splunk 的运行速度也越快。如可行，使用主轴转速超过 10,000 RPM 的磁盘。如需针对 Splunk 使用冗余存储，使用基于硬件的 RAID 1+0（也称为 RAID 10）。它提供了速度和冗余性之间的最佳平衡。不建议使用通过 Windows 磁盘管理工具提供的基于软件的 RAID 配置。
- **不允许防毒程序扫描用于运行 Splunk 作业的磁盘。**当防毒文件系统驱动器对访问文件执行病毒扫描时，性能会显著降低，尤其是当 Splunk 内部老化最近建立索引的数据时。如果您必须在运行 Splunk 的服务器上使用防毒程序，则必须确保所有 Splunk 目录和程序被排除在访问文件时的扫描之外。
- **如可用，使用多个索引。**将由 Splunk 建立索引的数据分散到不同索引中。将所有数据发送到默认索引可能会导致您的系统出现 I/O 瓶颈问题。应根据情况配置您的索引，以使其尽量指向系统中的不同物理卷。有关如何配置索引的更多信息，请参阅本手册中的“配置您的索引”。
- **不要将您的索引存储在操作系统所在的同一物理磁盘或分区上。**不建议将存储有 Windows 操作系统目录（%WINDIR%）或其交换文件的磁盘用于 Splunk 数据存储。应将您的 Splunk 索引存储在系统中的其他磁盘上。

有关索引存储方式的更多信息，包括数据库数据桶类型以及 Splunk 如何存储与老化这些项目的信息，请阅读本手册中的“Splunk 如何存储索引”。

- **不要将您的 Splunk 索引的热/温数据桶存储在网络卷上。**网络延迟将导致性能显著降低。应采用高速、本地磁盘来存储您的 Splunk 索引的热/温数据桶。对于冷/冻结的索引数据桶，您可以指定网络共享，例如分布式文件系统 (DFS) 卷或网络文件系统 (NFS) 装入点，但应注意，如果搜索包含那些存储在冷数据桶中的数据，则速度会变慢。
- **保持您的 Splunk 索引器的磁盘可用性、带宽和可用空间。**确保用于存储 Splunk 索引的磁盘卷始终具有 20% 或以上的可用空间。磁盘性能随可用空间的减少而成比例降低，因为这时磁盘寻道时间会增加。这会影响到 Splunk 建立数据索引的速度，并决定了搜索结果、报告和告警的返回速度。在默认的 Splunk 安装中，用于包含您索引的驱动器必须至少有 5,000 MB（约 5 GB）的可用磁盘空间，否则建立索引操作将暂停。

## 在系统映像上加入 Splunk

本主题介绍有关使 Splunk 成为每个 Windows 系统映像或安装过程的一部分的概念。它还引导您完成大致的集成过程，不论您使用何种映像工具。

- 有关将 Windows 数据导入 Splunk 的更多信息，请阅读《数据导入手册》中的“关于 Windows 数据和 Splunk”。
- 有关分布式 Splunk 部署的信息，请阅读《分布式部署手册》中的“分布式概述”。这也是您了解如何实施 Splunk 部署的必读内容（不论您采用哪种操作系统）。您还可以从中读到有关 Splunk 分布式部署操作的信息。
- 有关如何规划大规模 Splunk 部署的信息，请阅读《容量规划手册》中的“针对 Splunk Enterprise 容量规划的介绍”和本手册中的“[在 Windows 上部署 Splunk](#)”。

## 在 Windows 上的系统集成概念

将 Splunk 集成到 Windows 系统映像中的主要目的在于确保当在企业中启用计算机时 Splunk 能够立即可用。这样，您无需在计算机启用之后安装和配置 Splunk。

在此方案中，当 Windows 系统启动时，它将在引导后立即启动 Splunk。然后，根据所安装 Splunk 实例的类型和配置，Splunk 要么集合该计算机的数据并转发给某个索引器（多数情况下），要么开始对那些从其他 Windows 计算机转发过来的数据建立索引。

系统管理员还可以配置 Splunk 实例与某个部署服务器进行通信，以便执行后续配置和更新管理。

在许多典型环境中，在 Windows 计算机上的通用转发器会向某个中央索引器或一组索引器发送数据，然后根据您的特定需求，针对这些数据执行搜索、报告和告警。

## 系统集成注意事项

将 Splunk 集成到您的 Windows 系统映像中需要细致规划。

在多数情况下，集成到 Windows 系统映像中的首选 Splunk 组件是通用转发器。通用转发器专门设计用于分享在执行其他角色的计算机上的资源，它能够以非常低的成本来执行索引器可执行的大量工作。您还可以通过 Splunk 的部署服务器或某个企业内配置管理器来修改转发器的配置，而无需使用 Splunk Web 来做出更改。

在某些情况下，您可能需要将整个 Splunk 实例集成到系统映像中。这种做法的场合和时机是否适当，取决于您的特定需求和资源可用性。

Splunk 不建议您在执行任何其他角色的服务器的系统映像中包含 Splunk 的完整版本，除非您需要的是索引器而不是转发器的操作。在企业中安装多个索引器并不会带来额外的索引能力或速度，相反可能带来意外的结果。

在将 Splunk 集成到系统映像中之前，请考虑：

- **您需要 Splunk 来建立索引的数据量，以及 Splunk 需要将这些数据发送到何处（如适用）。**这将直接用于磁盘空间计算，应当是最重要的考虑因素。
- **要在映像或计算机上安装的 Splunk 实例类型。**在执行其他职责的工作站或服务器上安装通用转发器具有显著的优点，但在某些情况下可能不太适合。
- **在安装映像的计算机上可用的系统资源。**在每个映像系统上有多少可用磁盘空间、RAM 和 CPU 资源？它是否支持安装 Splunk？
- **您的网络的资源需求。**不论您是使用 WMI 来将其连接到远程计算机以集合数据，或是在每台计算机上安装转发器并向索引器发送数据，Splunk 都需要网络资源。
- **在映像中所安装的其他程序的系统要求。**如果 Splunk 与另一个服务器共享资源，则它可能会占用其他程序的可用资源。考虑您是否应在运行完整 Splunk 实例的工作站或服务器上安装其他程序。在此类情况下，通用转发器更能胜任，因为它采用轻型设计。
- **安装映像的计算机在您的环境中扮演的角色。**将成为只诸如 Office 这类生产力应用程序运行的工作站吗？或者，它将成为您的 Active Directory 林的操作主机域控制器？

## 将 Splunk 集成到系统映像中

在您确定上述检查表中问题的答案之后，下一步是将 Splunk 集成到您的系统映像中。这里列出了大致的步骤，因此您可以使用您喜好的系统映像或配置工具来完成该任务。

从下列系统集成选项中选择一项：

- [将通用转发器集成到系统映像中](#)
- [将 Splunk 的完整版本集成到系统映像中](#)

## 将通用转发器集成到系统映像中

本主题介绍将 Splunk 通用转发器集成到 Windows 系统映像中的程序。有关将 Splunk Enterprise 集成到映像中的其他信息，请参阅[将 Splunk Enterprise 集成到系统映像中](#)。

1. 使用基准计算机，根据您的需要安装并配置 Windows，包括安装任何所需的 Windows 功能、服务包和其他组件。
2. 安装并配置所需的应用程序，同时兼顾 Splunk 的系统和硬件容量需求。
3. 通过命令行安装和配置通用转发器。进行安装时，至少要提供 `LAUNCHSPLUNK=0` 命令行标记。
4. 继续到安装的图形部分，选择所需的输入、部署服务器和/或转发器目标。
5. 在您完成安装之后，打开命令提示符或 PowerShell 窗口。
6. 从提示符下，编辑那些无法在安装程序中配置的附加配置文件。
7. 从提示符编辑配置文件后，改为通用转发器 `bin` 目录。
8. 运行 `./splunk clone-prep-clear-config`
9. 关闭命令提示符或 PowerShell 窗口。
10. 在“服务控制面板”中，将启动类型设为“Automatic”，配置 `splunkd` 服务自动启动。
11. 使用诸如 Windows System Image Manager (WSIM) 等工具来准备系统映像以便加入域。Microsoft 建议使用 SYSPREP 或 WSIM 以在复制之前更改计算机安全标识符 (SID)，这与使用第三方工具（如 Ghost Walker 或 NTSID）的做法相反。
12. 在您配置好用于创建映像的系统之后，重新启动计算机，并使用您喜好的映像工具来制作映像副本。

接下来您就可以部署映像了。

## 将完整 Splunk 集成到系统映像中

本主题介绍将 Splunk 的完整版本集成到 Windows 系统映像中的程序。有关将 Splunk 集成到映像中的其他信息，请参阅本手册中的[“在系统映像上加入 Splunk”](#)。

要将 Splunk 的完整版本集成到系统映像中：

1. 使用基准计算机，根据您的需要安装并配置 Windows，包括安装任何所需的 Windows 功能、修补程序和其他组件。
  2. 安装并配置任何所需的应用程序，同时兼顾 Splunk 的系统和硬件容量需求。
  3. 安装和配置 Splunk。
- 重要提示：** 您可以使用 GUI 安装程序来进行安装，但从命令行中安装软件包时可以使用更多选项。
4. 在您配置 Splunk 输入之后，打开命令提示符。
  5. 从该提示符下，切换到 `%SPLUNK_HOME%\bin` 目录并发出 `.\splunk stop` 来停止 Splunk
  6. 发出 `.\splunk clean eventdata` 以清除任何事件数据。
  7. 关闭命令提示符窗口。



8. 确保将 `splunkd` 和 `splunkweb` 服务均设为自动启动。为此，您需要在“服务控制面板”中将其启动类型设为 'Automatic'。

9. 使用诸如 SYSPREP（适用于 Windows XP 和 Windows Server 2003/2003 R2）和/或 Windows 系统映像管理器 (WSIM)（对于 Windows Vista、Windows 7 和 Windows Server 2008/2008 R2）等工具来准备系统映像以便加入域。

**注意：**Microsoft 建议使用 SYSPREP 和 WSIM 以在复制之前更改计算机安全标识符 (SID)，这与使用第三方工具（如 Ghost Walker 或 NTSID）的做法相反。

10. 在您配置好用于创建映像的系统之后，重新启动计算机，并使用您喜爱的映像工具来制作映像副本。

接下来您就可以部署映像了。

## 使用 Splunk Web 管理 Splunk Enterprise

### 启动 Splunk Web

Splunk 运行后，您就可以启动 Web 界面 **Splunk Web** 了。要了解更多 Splunk Web 相关信息，请参阅：

- [Splunk Web 管理任务](#)
- 导航 Splunk Web
- 使用 Splunk 搜索

要启动 Splunk Web，导航到：

`http://mysplunkhost:<port>`

使用您在安装期间选择的主机和端口。

当您使用 Enterprise 许可证首次登录 Splunk 时，默认的登录信息为：

**用户名 - admin**  
**密码 - changeme**

**注意：**使用免费许可证的 Splunk 不具有访问控制，因此不会提示您输入登录信息。

**注意：**从 Splunk 版本 4.1.4 起，您无法从远程浏览器访问 Splunk Free，除非您已编辑 `$SPLUNK_HOME/etc/local/server.conf` 并将 `allowRemoteLogin` 设置为 `Always`。如果您在运行 Splunk Enterprise，则默认情况下禁止管理员用户远程登录（设为 `requireSetPassword`），除非您更改了默认密码。

### Splunk Web 管理任务

**Splunk Web** 是 Splunk 的基于浏览器的界面。下面是您在 Splunk Web 中可以执行的几项操作：

- 配置数据导入
- 搜索数据和报表并可视化结果
- 调查问题
- 在本机或通过 LDAP 策略管理用户
- Splunk 部署故障排除
- 管理群集和对等节点

参考系统要求以获得所支持的操作系统和浏览器列表。

### Splunk 设置菜单

Splunk Web 提供了一个方便的界面来管理大多数 Splunk 的操作。大多数功能均可通过单击菜单中的**设置**来访问。从此处，您可以：

#### 管理数据

在**设置 > 数据**下，您可以执行以下操作：

- **数据输入**使您可以查看数据类型的列表并配置这些数据类型。要添加输入，请单击“数据输入”页面中的**添加数据**按钮。有关如何添加数据的更多信息，请参阅《*数据导入*》手册。
- **转发和接收**使您可以设置转发器和接收器。有关设置转发和接收的更多信息，请参阅《*转发数据*》手册。
- **索引**使您可以添加、禁用和启用索引。
- **报表加速摘要**带您搜索和报表应用，以使您审阅现有的报表摘要。有关创建报表摘要的更多信息，请参阅《*知识管理器手册*》。

## 管理用户和用户验证

通过导航到**设置 > 用户和验证 > 访问控制**，您可以执行以下操作：

- 创建和管理用户
- 定义和分配角色
- 设置 LDAP 验证策略

有关使用用户和验证的更多信息，请参阅《确保 Splunk 安全》手册。

## 使用应用

要查看您已安装的应用，请选择菜单栏中的**应用**。

在此页面上，您可以从已经安装且当前可用的应用列表中选择一个应用。您还可以从此处访问下列菜单选项：

- **查找更多应用**使您可以搜索并安装其他应用。
- **管理应用**使您可以管理现有应用。

您还可以从“主页”页面访问所有应用。

有关应用的更多信息，请参阅“开发用于 Splunk Web 的视图和应用”。

## 管理系统的各个方面

**设置 > 系统**下的选项允许您执行以下操作：

- **服务器设置**使您可以管理 Splunk 设置，如端口、主机名、索引路径、电子邮件服务器以及系统日志和部署客户端信息。有关使用 Splunk Web 配置和管理分布式环境的更多信息，请参阅《更新 Splunk 组件》手册。
- **服务器控件**使您可以重新启动 Splunk。
- **许可授权**使您可以管理 Splunk 许可证并延长其有效期。

# Splunk Enterprise 默认仪表板

Splunk Enterprise 附带了一系列有用的仪表板。它们有助于您排除系统故障和进行搜索，还能帮助您了解您需要如何设计您自己的仪表板和视图。

## 活动仪表板

您可以通过单击页面顶部附近的用户栏中的**活动 > 系统活动**找到下列仪表板。

**注意：**这些仪表板仅对具备管理员角色权限的用户可见。请参阅《确保 Splunk Enterprise 安全》中的“添加和管理用户”。有关设置仪表板权限的信息，请参阅《知识管理器手册》。



- **搜索活动**-该仪表板集合提供有关您的 Splunk 实例搜索活动的速览信息。您可以了解到搜索何时运行、它们对系统产生的负载量、哪些搜索最常用、哪些搜索视图和仪表板用得最多等等。提供的仪表板如下：
  - 搜索活动概述
  - 搜索详细信息
  - 搜索用户活动
- **服务器活动** - 该仪表板的集合提供有关 Splunkd 和 Splunk Web 性能的指标信息，而且便于进行故障排除。您可以看到所报告的错误数量、最近错误列表、时间戳问题和未处理的异常状况列表、显示近期浏览器使用情况的图表等等。提供的仪表板如下：
  - 内部消息和错误
  - 许可证使用情况
- **计划程序活动**-该仪表板的集合允许您深入了解搜索计划程序的工作情况，确保及时运行特殊和计划的搜索。
  - 计划程序活动概述
  - 按用户或应用的计划程序活动
  - 依据保存搜索的计划程序活动
  - 计划程序错误

## 摘要仪表板

“摘要”仪表板是您在进入“搜索和报表”应用之后首先看到的内容。它提供了一个搜索栏和时间范围挑选器，您可以用它们来输入并运行您的初始搜索。

当您某个输入添加到 Splunk 时，该输入将与您当前使用的应用建立关联。某些应用（例如 \*nix 和 Windows 应用）会将输入数据写入到特定索引（对于 \*nix 和 Windows，此为 **os** 索引）。如果您查看摘要仪表板，但看不到那些您确定其位于 Splunk 中的数据，请确保您在正确的索引上进行查看。

您可能需要将某个应用使用的索引添加到您当前角色的默认索引列表。有关角色的更多信息，请参阅《确保 Splunk 安全》中关于角色的此主题。有关摘要仪表板的更多信息，请参阅搜索教程。

## 自定义 Splunk Web 横幅消息

您可以添加和编辑 Splunk Web 中消息菜单下显示的通知。

您需要管理员或系统用户级别权限来添加或编辑通知。

要更改或编辑通知：

1. 选择**设置 > 用户界面**。
2. 单击**新建**以创建新消息，或单击**公告消息**并选择您想要编辑的消息。
3. 编辑现有消息文本，或为新消息指定一个名称和消息文本。
4. 单击**保存**。用户访问菜单中的消息时，消息立刻会出现。

## 配合使用 Splunk Web 与代理服务器

当 Splunk Web 位于代理服务器之后时，用户使用访问 Splunk 网站的 Splunk Web 链接时可能会遇到问题。例如，一些 Splunk Web 页面直接链接到 Splunk 应用的下载站点，许多“了解更多信息”链接将使你访问在线文档。

要解决此问题，只需设置 HTTP\_PROXY 环境变量。要获得永久性结果，您可以在 `splunk-launch.conf` 配置文件中指定设置，该文件位于 `$SPLUNK_HOME/etc/`（\*nix 系统）和 `%SPLUNK_HOME%\etc\`（Windows 系统）中。

**注意：**“应用管理器”不支持用于代理服务器，如果您使用带有 Splunk Web 的代理服务器，您必须手动下载和更新应用。

在 `splunk-launch.conf` 中，添加以下属性/值对：

```
HTTP_PROXY = <IP address or host name>:<port number>
```

例如：

```
HTTP_PROXY = 10.1.8.11:8787
```

**重要提示：**如果代理服务器仅处理 HTTPS 请求，则必须使用以下属性/值对：

```
HTTPS_PROXY = <IP address or host name>:<port number>
```

例如：

```
HTTPS_PROXY = 10.1.8.11:8888
```

## 使用配置文件管理 Splunk Enterprise

### 关于配置文件

Splunk Enterprise 的配置信息都存储在**配置文件**中。这些文件由 `.conf` 扩展名标识，其中保存配置不同方面的信息。这些方面包括：

- 系统设置
- 验证和授权信息
- 索引映射和设置
- 部署和群集配置
- 知识对象和已保存的搜索

有关配置文件的列表以及每个文件所涵盖内容的概述，请参阅本手册中的[“配置文件列表”](#)。

大多数配置文件均附带了 `$SPLUNK_HOME/etc/system/default/` 目录中的 Splunk 软件。

## 使用 Splunk Web 来管理配置文件

在 Splunk Web 中更改配置时，此更改会写入到该设置的配置文件副本中。Splunk 软件创建此配置文件的副本（如果不存在）、将更改写入到此副本，并将其添加到 `$SPLUNK_HOME/etc/...` 下的某个目录中。新文件所添加到的目录取决于多个因素，这将在本手册的[“配置文件目录”](#)中加以讨论。最常见的目录为 `$SPLUNK_HOME/etc/system/local`，在本示例中使用此目录。

如果您在 Splunk Web 中添加一个新索引，软件会执行下列操作：

1. 检查文件的副本是否存在。
2. 如果无副本存在，软件会创建 `indexes.conf` 的副本并将其添加到 `$SPLUNK_HOME/etc/system/local` 之类的目录中。
3. 将更改写入到 `indexes.conf` 的副本中。
4. 在 `$SPLUNK_HOME/etc/system/default` 中保留未更改的默认文件。

## 直接编辑配置文件

尽管可以从 Splunk Web 进行许多配置，但您也可以针对任一设置直接编辑配置文件。对于一些 Splunk Web 不支持的高级自定义，请直接编辑配置文件。

**注意：**与在 Splunk Web 中进行更改相比，编辑配置文件需要重新启动的频率更高。请参阅本手册中的[更改配置文件之后何时重新启动 Splunk](#)。

**重要提示：**不要更改或复制默认目录中的配置文件。默认文件必须保持原样并位于其原始位置。要更改特定配置文件的设置，必须先是非默认目录中创建新版本文件，然后添加想要更改的设置。有关您可以编辑配置文件的目录位置相关信息，请参阅[配置文件目录](#)。开始创建文件新版本时，以空文件开始。不要以默认目录副本开始。

在您更改任何配置文件之前：

- 了解默认配置文件如何工作以及您编辑的副本应置于何处。请参阅本手册中的[“配置文件目录”](#)。
- 了解构成配置文件的段落结构以及如何设置要编辑的属性。请参阅本手册中的[“配置文件结构”](#)。
- 了解不同目录中同一配置文件的不同副本如何分层放置，以便您知道您副本的最佳放置位置。请参阅本手册中的[“配置文件优先顺序”](#)。

熟悉配置文件内容和目录结构并了解如何利用 Splunk Enterprise 的配置文件优先顺序后，请参阅[“如何编辑配置文件”](#)以了解如何安全地修改文件。

## 配置文件目录

单个 Splunk 实例通常有配置文件的多个版本，分散在几个目录中。您可以有几个具有相同名称的配置文件，分散在 `default`、`local` 和 `app` 目录中。这样可以形成层级，允许 Splunk 基于因素（例如当前用户和当前应用）决定配置优先级。

要了解有关 Splunk 如何评定配置优先级的更多信息，请参阅[“配置文件优先顺序”](#)。

**注意：**给定配置文件最准确的设置列表在该配置文件的 `.spec` 文件中。`.spec` 和 `.example` 文件的最新版本在[“配置文件参考”](#)或 `$SPLUNK_HOME/etc/system/README` 中。

## 关于默认文件

*“上述所有目录您都可以随意更改，除了 `/default` - 不要尝试编辑任何内容”*

-- duckfez, 2010

默认目录包含预配置版本的配置文件。默认目录位置是 `$SPLUNK_HOME/etc/system/default`。

**重要提示：**不要更改或复制默认目录中的配置文件。默认文件必须保持原样并位于其原始位置。Splunk Enterprise 升级过程将覆盖默认目录，因此默认目录下所做的所有更改在升级过程中将丢失。非默认配置目录下所做的更改，例如 `$SPLUNK_HOME/etc/system/local` 或 `$SPLUNK_HOME/etc/apps/<app_name>/local` 升级后仍存在。

要更改特定配置文件的属性值，必须先是非默认目录中创建新版本文件，然后在其中修改值。非默认目录中的值优先于默认目录中的值。

**注意：**开始创建新版本文件时，以空文件开始，仅添加需要更改的属性。不要以默认目录副本开始。如果您将整个默认文件复制到优先级较高的位置，则将来 Splunk Enterprise 升级时默认值的任意更改将不会生效，因为复制的文件中的值将覆盖默认文件中更新的值。

## 在何处放置（或查找）已修改的配置文件

您可以将配置文件的几个版本分层堆叠在一起，Splunk 按照“[配置文件优先顺序](#)”中介绍的分层堆叠方案使用不同的属性值。

切勿在默认目录中编辑文件。而是在配置目录之一中创建和/或编辑文件，例如 `$SPLUNK_HOME/etc/system/local`。升级过程中不会覆盖这些目录。

对于大多数部署，您可以使用 `$SPLUNK_HOME/etc/system/local` 目录进行配置更改。然而，在某些情况下，您可能希望使用其他目录中的文件。下面是 `$SPLUNK_HOME/etc` 中的配置目录结构：

- `$SPLUNK_HOME/etc/system/local`
  - 基于网站范围的本地更改存储在此目录中；例如，想要可供所有应用使用的设置。如果此目录中尚不存在您正在查找的配置文件，请创建该文件并授予其写入权限。
- `$SPLUNK_HOME/etc/slave-apps/[[_cluster]<app_name>]/[local|default]`
  - **仅供群集对等节点使用。**
  - `$SPLUNK_HOME/etc/slave-apps` 下的子目录包含所有对等节点通用的配置文件。
  - **不要更改群集对等节点本身的这些子目录的内容。**而是使用群集主节点来将任何新的或已修改的文件分发这些子目录。
  - `_cluster` 目录包含的配置文件虽然并不属于真正应用中的一部分，但是在所有对等节点之间必须保持一致。`indexes.conf` 文件就是一个典型示例。
  - 有关更多信息，请参阅《管理索引器和群集手册》中的“更新通用对等节点配置”。
- `$SPLUNK_HOME/etc/apps/<app_name>/[local|default]`
  - 如果进行配置更改时您在应用中，则设置会写入应用的 `/local` 目录中的配置文件。例如，在默认 Splunk 搜索应用中对搜索时间设置所做的编辑会写入：`$SPLUNK_HOME/etc/apps/search/local/`
  - 如果您想要编辑配置文件，但使更改仅应用于某一应用，则将配置文件复制到该应用的 `/local` 目录（具有写入权限）中，然后在此目录中进行更改。
- `$SPLUNK_HOME/etc/users`
  - 用户特定的配置更改写入此处。
- `$SPLUNK_HOME/etc/system/README`
  - 此目录包含支持参考文档。对于大多数配置文件，有两个参考文件：`.spec` 和 `.example`；例如 `inputs.conf.spec` 和 `inputs.conf.example`。`.spec` 文件指定语法，包括可用属性和变量列表。`.example` 文件包含实际用法示例。

## 配置文件结构

编辑配置文件之前，您自己应熟悉这些文件的结构。

### 段落

配置文件由一个或多个段落或章节组成。每个段落的开头是段落标题，以方括号括起。此标题标识该段落中所保留的设置。每个设置均为指定特定配置设置的属性值对。

例如，`inputs.conf` 提供 `[SSL]`，其中包括服务器证书和密码的设置（以及其他设置）：

```
[SSL]
serverCert = <pathname>
password = <password>
```

其中一些属性为必填，而另一些属性为可选，具体取决于段落类型。

### 设置新段落

编辑配置文件时，您可能要更改默认段落（如上所述），也可能需要添加全新的段落。

基本模式如下：

```
[stanza1_header]
<attribute1> = <val1>
# comment
<attribute2> = <val2>
...

[stanza2_header]
<attribute1> = <val1>
<attribute2> = <val2>
...
```

**重要提示：**属性区分大小写。例如 `sourcetype = my_app` 与 `SOURCETYPE = my_app` 不同。其中一个起作用，另一个不起作用。

## 段落范围

配置文件常常会有不同范围的段落，更为具体详细的段落优先。例如，假设此 `outputs.conf` 配置文件示例，用于配置转发器：

```
[tcpout]
indexAndForward=true
compressed=true

[tcpout:my_indexersA]
autoLB=true
compressed=false
server=mysplunk_indexer1:9997, mysplunk_indexer2:9997

[tcpout:my_indexersB]
autoLB=true
server=mysplunk_indexer3:9997, mysplunk_indexer4:9997
```

请注意，本示例文件有两个段落层级：

- 全局 `[tcpout]`，包含影响所有 tcp 转发的设置。
- 两个 `[tcpout:<target_list>]` 段落，其设置仅影响在每个目标组中定义的索引器。

`compressed` 在 `[tcpout:my_indexersA]` 中的设置覆盖该属性在 `[tcpout]` 中的设置，*仅适用于 `my_indexersA` 目标组中的索引器*。

有关转发器和 `outputs.conf` 的更多信息，请参阅“使用 `outputs.conf` 配置转发器”。

## 配置文件优先顺序

有关配置文件的更多信息，请阅读“关于配置文件”。

Splunk 软件使用**配置文件**确定其行为的几乎各个方面。Splunk 平台部署可以具有同一配置文件的多个副本。这些文件副本通常分层放置在影响用户、**应用**或系统整体的目录中。

编辑配置文件时，了解 Splunk 软件如何评估这些文件以及这些文件的优先顺序非常重要。

合并更改时，Splunk 软件对配置文件执行下列操作：

- 它按照基于位置的优先顺序方案合并来自文件的所有副本的设置。
- 当不同副本有冲突的属性值时（也就是说，不同副本将相同属性设置为不同值时），它使用具有最高优先级的文件的值。
- 它按照配置文件在其目录结构中的位置来确定优先级（依据文件是在系统目录、应用目录还是用户目录中）。确定应用目录集合中的优先级时，Splunk 使用 ASCII 排序顺序。应用目录中名称包含 "A" 的文件比应用目录中名称包含 "B" 的文件的优先级高，以此类推。

**注意：**除了解决一个文件的多个副本之间的配置设置之外，Splunk 软件有时还需要解决单个文件中的设置。有关 Splunk 软件如何在单个 `props.conf` 文件中确定优先顺序的信息，请参阅“单个 `props.conf` 文件中的属性优先顺序”。

### 关于配置文件上下文

如何确定优先顺序取决于文件的上下文。

#### **应用或用户上下文与全局上下文**

要确定配置文件的多个副本的优先级，Splunk 软件首先确定目录方案。

Splunk 软件使用两个主要的目录优先顺序方案。

- 应用或用户：一些活动，如搜索，发生在应用或用户上下文中。应用和用户上下文对于搜索时间处理非常重要，其中某些知识对象或动作可能仅对特定应用中的特定用户有效。
- 全局：建立索引等活动发生在全局上下文中。它们独立于应用或用户。例如，确定监视行为的配置文件发生在应用或用户上下文外，但其本质上属于全局上下文。

#### **群集对等节点配置上下文**

对于群集对等节点全局配置，还有扩展的优先顺序。这是因为一些配置文件（例如 `indexes.conf`）必须在所有对等节

点之间都相同。

要保持文件一致，需从群集主节点对其进行管理，这样文件便分布至对等节点，以便所有对等节点都包含相同的文件版本。这些文件在群集对等节点的配置中具有最高优先级，这将在下一节中介绍。

有关如何跨对等节点分布配置的更多信息，请参阅《管理索引器和群集》手册中的“更新通用对等节点配置”。

## Splunk 如何确定优先顺序

本小节将对优先顺序和上下文的概念进行介绍。对于按目录名称排序的列表，请参阅将在本主题稍后部分介绍的“目录优先顺序摘要”。

### 全局上下文内的优先顺序：

当上下文为全局时（即，其中没有应用或用户上下文），目录优先级按以下顺序下降：

- 1.System local 目录 -- 最高优先级
- 2.App local 目录
- 3.App default 目录
- 4.System default 目录 -- 最低优先级

使用全局配置时（例如 `inputs.conf`），Splunk 首先使用来自 `system/local` 中文件副本的属性。然后查找 `app` 目录中文件的副本，追加在其中找到的属性，但忽略已在 `system/local` 中发现的属性。最后，对于在系统级或应用级都未显示分配的属性，为其分配 `system/default` 目录中文件的默认值。

**注意：**群集对等节点具有扩展的优先顺序，下一节将对此进行介绍。

### 群集对等节点的优先顺序

对于群集对等节点，全局上下文会考虑一些额外的对等节点特定 ("slave-app") 目录。这些目录包含一些在所有对等节点之间都相同的应用和配置。以下是扩展的群集对等节点优先顺序：

- 1.Slave-app local 目录（仅适用于群集对等节点）-- 最高优先级
- 2.System local 目录
- 3.App local 目录
- 4.Slave-app default 目录（仅适用于群集对等节点）
- 5.App default 目录
- 6.System default 目录 -- 最低优先级

带有群集对等节点，并对所有对等节点通用的自定义设置（slave-app local 目录中的那些设置）具有最高优先级。

### 应用或用户上下文内的优先顺序

有应用或用户上下文时，目录优先级为从 user 降至 app 降至 system：

- 1.当前用户的 User 目录 -- 最高优先级
- 2.当前运行的应用的 App 目录（local，后跟 default）
- 3.所有其他应用的 App 目录（local，后跟 default）-- 仅适用于导出设置
- 4.System 目录（local，后跟 default）-- 最低优先级

例如，`savedsearches.conf` 中的属性可在所有三个层级设置：用户级、应用级和系统级。Splunk 始终使用用户级属性的值（如果有），优先于在应用级或系统级设置的相同属性的值。

### 应用目录名称如何影响优先顺序

**注意：**就大多数实际用途而言，此小节中的信息可能无关紧要，但如果您需要强制按某一顺序进行评估或者故障排除，则此信息非常有用。

确定应用目录集合中的优先级时，Splunk 使用 ASCII 排序顺序。应用目录中名称包含 "A" 的文件比应用目录中名称包含 "B" 的文件的优先级高，以此类推。此外，以大写字母开始的所有应用的优先级将高于以小写字母开头的的所有应用，这是由 ASCII 排序顺序决定的。（例如，"A" 的优先级高于 "Z"，但 "Z" 的优先级高于 "a"。）

此外，数字表示的目录的优先级比字母表示的目录的优先级高，按词典顺序进行评估，而不是按数字顺序进行评估。例如，按降序排列的优先顺序：

```
$SPLUNK_HOME/etc/apps/myapp1
$SPLUNK_HOME/etc/apps/myapp10
$SPLUNK_HOME/etc/apps/myapp2
$SPLUNK_HOME/etc/apps/myapp20
...
$SPLUNK_HOME/etc/apps/myappApple
```

```
$SPLUNK_HOME/etc/apps/myappBanana
$SPLUNK_HOME/etc/apps/myappZabaglione
...
$SPLUNK_HOME/etc/apps/myappapple
$SPLUNK_HOME/etc/apps/myappbanana
$SPLUNK_HOME/etc/apps/myappzabaglione
...
```

**注意：**在应用或用户上下文中确定优先顺序时，目前正在运行的应用的目录优先于所有其他应用的目录，与目录的命名方式无关。此外，其他应用方面的检查应当仅针对于导出设置。

### 目录优先顺序摘要

总而言之，目录优先级顺序从最高到最低如下：

#### 全局上下文：

```
$SPLUNK_HOME/etc/system/local/*

$SPLUNK_HOME/etc/apps/A/local/* ... $SPLUNK_HOME/etc/apps/z/local/*

$SPLUNK_HOME/etc/apps/A/default/* ... $SPLUNK_HOME/etc/apps/z/default/*

$SPLUNK_HOME/etc/system/default/*
```

#### 全局上下文 - 仅供群集对等节点使用：

```
$SPLUNK_HOME/etc/slave-apps/A/local/* ... $SPLUNK_HOME/etc/slave-apps/z/local/*

$SPLUNK_HOME/etc/system/local/*

$SPLUNK_HOME/etc/apps/A/local/* ... $SPLUNK_HOME/etc/apps/z/local/*

$SPLUNK_HOME/etc/slave-apps/A/default/* ... $SPLUNK_HOME/etc/slave-apps/z/default/*

$SPLUNK_HOME/etc/apps/A/default/* ... $SPLUNK_HOME/etc/apps/z/default/*

$SPLUNK_HOME/etc/system/default/*
```

**重要提示：**在 `slave-apps/[local|default]` 目录中，特殊 `_cluster` 子目录的优先级要高于以小写字母开头的所有应用子目录（如 `anApp`）。但是，该子目录的优先级要低于以大写字母开头的所有应用（如 `AnApp`）。这是由下划线（"\_"）字符在 ASCII 排序顺序中的位置决定的。

#### 应用或用户上下文：

```
$SPLUNK_HOME/etc/users/*

$SPLUNK_HOME/etc/apps/Current_running_app/local/*

$SPLUNK_HOME/etc/apps/Current_running_app/default/*

$SPLUNK_HOME/etc/apps/A/local/*, $SPLUNK_HOME/etc/apps/A/default/*, ... $SPLUNK_HOME/etc/apps/z/local/*,
$SPLUNK_HOME/etc/apps/z/default/* (but see note below)

$SPLUNK_HOME/etc/system/local/*

$SPLUNK_HOME/etc/system/default/*
```

**重要提示：**在应用或用户上下文中，目前正在运行的应用的所有配置文件优先于所有其他应用的文件。此规则也适用于该应用的 `local` 和 `default` 目录。因此，如果当前上下文是应用 C，Splunk 会先评估 `$SPLUNK_HOME/etc/apps/C/local/*` 和 `$SPLUNK_HOME/etc/apps/C/default/*`，然后再评估其他应用的 `local` 或 `default` 目录。此外，只有当其他应用的配置数据已通过该应用的 `default.meta` 文件全局导出时，Splunk 软件才会查看这些数据。有关更多信息，请参阅 Splunk 开发人员门户中的“为 Splunk 应用中的对象设置权限”。

同理，请注意，只有当特定用户登录或执行搜索时，才会评估 `/etc/users/`。

### 属性优先顺序工作原理示例



本属性优先顺序示例使用 `props.conf`。 `props.conf` 文件比较特殊，因为其上下文可以是全局，也可以是应用或用户，具体取决于 Splunk 评估文件的时机。Splunk 同时在索引时间（全局）和搜索时间（应用或用户）评估 `props.conf`

假设 `$SPLUNK_HOME/etc/system/local/props.conf` 包含以下段落：

```
[source::/opt/Locke/Logs/error*]
sourcetype = fatal-error
```

而 `$SPLUNK_HOME/etc/apps/t2rss/local/props.conf` 包含同一段落的另一个版本：

```
[source::/opt/Locke/Logs/error*]
sourcetype = t2rss-error
SHOULD_LINEMERGE = True
BREAK_ONLY_BEFORE_DATE = True
```

始终应用 `t2rss` 中的行合并属性分配，因为这些属性分配仅出现在该文件的该版本中。但是与 `sourcetype` 属性有冲突。在 `/system/local` 版本中，`sourcetype` 的值为 "fatal-error"。在 `/apps/t2rss/local` 版本中，属性值为 "t2rss-error"。

由于这是在索引时间应用的 `sourcetype` 分配，Splunk 使用全局上下文确定目录优先顺序。在全局上下文中，Splunk 将最高优先级指定给 `system/local` 中的属性分配。因此，`sourcetype` 属性指定值 "fatal-error"。

该文件最终的内部合并版本如下：

```
[source::/opt/Locke/Logs/error*]
sourcetype = fatal-error
SHOULD_LINEMERGE = True
BREAK_ONLY_BEFORE_DATE = True
```

## 配置文件及其上下文列表

如上所述，Splunk 根据配置文件所运行的上下文（全局或应用或用户）来确定文件的评估方式。通常而言，影响数据输入、建立索引或部署活动的文件为全局；影响搜索活动的文件通常有应用或用户上下文。

`props.conf` 和 `transforms.conf` 文件在应用或用户或全局上下文中都可以评估，具体情况取决于 Splunk 是在索引时间还是在搜索时间使用这两个文件。

### 全局配置文件

```
admon.conf
authentication.conf
authorize.conf
crawl.conf
deploymentclient.conf
distsearch.conf
indexes.conf
inputs.conf
outputs.conf
pdf_server.conf
procmonfilters.conf
props.conf -- global and app/user context
pubsub.conf
regmonfilters.conf
report_server.conf
restmap.conf
searchbnf.conf
segmenters.conf
server.conf
serverclass.conf
serverclass.seed.xml.conf
source-classifier.conf
sourcetypes.conf
sysmon.conf
tenants.conf
transforms.conf -- global and app/user context
user-seed.conf -- special case: Must be located in /system/default
web.conf
```

wmi.conf

## 应用或用户配置文件

```
alert_actions.conf
app.conf
audit.conf
commands.conf
eventdiscoverer.conf
event_renderers.conf
eventtypes.conf
fields.conf
limits.conf
literals.conf
macros.conf
multikv.conf
props.conf -- global and app/user context
savedsearches.conf
tags.conf
times.conf
transactiontypes.conf
transforms.conf -- global and app/user context
user-prefs.conf
workflow_actions.conf
```

## 配置优先顺序和其他问题的故障排除

Splunk 的配置文件系统支持位于许多不同位置的许多重叠配置文件。实现这种程度灵活性的代价是判定您的 Splunk 安装中使用哪个配置选项的哪个值有时非常复杂。如果您想查找在给定情况下会使用什么配置设置的一些提示，请阅读《故障排除手册》中的“使用 btool 排除配置故障”。

## 单个 props.conf 文件中的属性优先顺序

除了要了解[属性优先顺序在文件之间如何工作](#)之外，您有时还需要考虑在单个 [props.conf](#) 文件中的属性优先级。

### 影响同一目标的一组段落中的优先顺序

当两个或多个段落指定影响同一项目的行为时，按段落的 ASCII 顺序评估项目。例如，假设您在 `props.conf` 中指定以下段落：

```
[source:.../bar/baz]
attr = val1

[source:.../bar/*]
attr = val2
```

将使用 `attr` 在第二个段落中的值，因为该值的路径在 ASCII 顺序中较高，因此相对来说优先。

### 覆盖 props.conf 中的默认属性优先级

有方法覆盖 `props.conf` 中的默认 ASCII 优先级。使用 `priority` 键为给定段落指定较高或较低的优先级。

例如，假设有以下来源：

```
source::az
```

和以下模式：

```
[source:...a...]
sourcetype = a

[source:...z...]
sourcetype = z
```

在本案例中，默认行为是由模式 "source:...a..." 提供的设置优先于由 "source:...z..." 提供的设置。因此，`sourcetype` 的值为 "a"。

要覆盖此默认 ASCII 顺序，使用 `priority` 键：

```
[source:...a...]  
sourcetype = a  
priority = 5  
  
[source:...z...]  
sourcetype = z  
priority = 10
```

为第二个段落分配更高的优先级导致 `sourcetype` 的值为 "z"。

还需要考虑另一个属性优先顺序问题。默认情况下，与字符串逐字匹配的段落（“字面匹配段落”）优先于正则表达式模式匹配段落。这是由于其 `priority` 键的默认值：

- 0 是模式匹配段落的默认值。
- 100 是字面匹配段落的默认值。

因此，字面匹配段落始终优先于模式匹配段落，除非通过显式设置 `priority` 键来更改行为。

您可以使用 `priority` 键来解决相同类型的模式之间的冲突，例如 `sourcetype` 模式或 `host` 模式。但是，`priority` 键不影响规范类型之间的优先顺序。例如，`source` 模式优先于 `host` 和 `sourcetype` 模式，而不管 `priority` 键值。

## 具有多个属性分配的事件的优先顺序

`props.conf` 文件设置属性，以按 `host`、`source` 或 `sourcetype`（有时还有事件类型）处理单个事件。因此，一个事件的默认字段 `host`、`source` 或 `sourcetype` 有可能相同属性设置不同值。优先顺序如下：

- `source`
- `host`
- `sourcetype`

您可能想要覆盖默认 `props.conf` 设置。例如，假设您要跟踪 `mylogfile.xml`，默认标记为 `sourcetype = xml_file`。此配置只要更改就会重新索引整个文件，即使您手动指定另一个 `sourcetype`，因为 `property` 通过 `source` 来设置。要覆盖此设置，通过 `source` 添加显式配置：

```
[source::/var/log/mylogfile.xml]  
CHECK_METHOD = endpoint_md5
```

## 如何编辑配置文件

编辑配置文件之前，请确保您熟悉以下内容：

- 要了解默认配置文件所在位置以及您编辑的配置文件应置于何处，请参阅[配置文件目录](#)。
- 要了解文件结构以及如何设置要编辑的属性，请参阅[配置文件结构](#)。
- 要了解如何分层放置和组合跨多个目录的配置文件，请参阅[配置文件优先顺序](#)。

## 自定义配置文件

要在配置文件中自定义属性，请在本地目录或应用目录下创建新的文件，以相同的名字命名，然后您可以将您需要自定义的特定属性添加到本地配置文件。

1. 确定配置文件是否已经存在于首选目录，如 `$SPLUNK_HOME/etc/system/local`。请参阅本手册中的[配置文件优先顺序](#)。
2. 如果文件已经在首选目录中存在，编辑现有文件。若不存在，在首选目录中创建文件。请勿将默认配置文件内容复制到首选目录的文件中。这是为了确保任何 Splunk 软件升级合理地更新默认值。
3. 只将需要自定义的段落和属性添加到本地文件。

## 清除属性

您可以清除任何一个属性，方法是将属性值设置为空值。例如：

```
forwardedindex.0.whitelist =
```

这样可覆盖该属性以前保存的所有值，包括在默认文件中设置的值，从而使系统认为值已完全取消设置。

## 插入注释

您可以在配置文件中插入注释。方法是使用 `#` 符号：

```
# This stanza forwards some log files.  
[monitor:///var/log]
```

**重要提示：**从左侧开始注释。不要将注释与段落或属性放在同一行上：

```
[monitor:///var/log]    # This is a really bad place to put your comment.
```

例如，对于以下属性

```
a_setting = 5    #5 is the best number
```

这会将 a\_setting 属性设置为 "5 #5 is the best number"，从而造成意外的结果。

## 在 Windows 和其他非 UTF-8 操作系统上创建和编辑配置文件

Splunk 平台使用 ASCII/UTF-8 编码的配置文件。对于 UTF-8 不是默认字符集的操作系统（例如 Windows），要配置文本编辑器以写入该格式的文件。

## 更改配置文件之后何时重新启动 Splunk Enterprise

通过配置文件对 Splunk Enterprise 进行更改时，您可能需要重新启动 Splunk Enterprise 以使更改生效。

**注意：**在 Splunk Web 中进行的更改需要重新启动的可能性很小。这是因为 Splunk Web 自动更新基本配置文件并将更改通知运行中的 Splunk 实例 (splunkd)。

本主题提供指导原则来帮助您确定是否在更改后重新启动。更改是否需要重新启动取决于多个因素，本主题并不提供明确结论。始终查阅配置文件或其参考主题，了解特定更改是否需要重新启动。有关配置文件的完整列表以及每个文件所涵盖内容的概述，请参阅本手册中的[配置文件列表](#)。

### 重新启动转发器的时间

如果您更改重型转发器的配置文件，则必须重新启动转发器，但不必重新启动接收索引器。如果更改是已经配置为更改后重新启动的已部署应用的一部分，转发器则会自动重新启动。

### 重新启动 Splunkweb 的时间

必须重新启动 Splunkweb，才能对 Splunk Web 访问启用或禁用 SSL。

### 重新启动 Splunkd 的时间

正常情况下，在作出下列几种更改后重新启动 Splunkd。

#### 索引更改

- 索引时间字段提取
- 时间戳属性

**注意：**通过 Splunk Web 和 CLI 进行影响索引建立的设置更改时，这些设置不需要重新启动而且会立即生效。

请参阅《[管理索引器和索引器群集](#)》中的“更新通用对等节点配置和应用”。

#### 用户和角色更改

在配置文件中进行的任何用户和角色更改均需重新启动，包括：

- LDAP 配置（如果您在 Splunk Web 中进行这些更改，则可重新加载更改而不必重新启动。）
- 密码更改
- 角色功能更改
- Splunk Enterprise 本机验证更改，如用户到角色的映射。

#### 系统更改

影响系统设置或服务器状态的所有更改均需重新启动，比如：

- 许可授权更改
- Web 服务器配置更新
- 常规索引器设置（最小可用磁盘空间、默认服务器名称等）更改
- 常规设置（如端口设置）更改。请参阅《[管理索引器和索引器群集](#)》中的“确定哪个 indexes.conf 变更需要重启”。
- 更改转发器的输出设置

- 更改 Splunk Enterprise 实例操作系统中的时区（Splunk Enterprise 在启动时从基本操作系统检索其本地时区）
- 创建搜索头池
- 安装某些应用可能需要重新启动。查阅您要安装每个应用的文档。

## 不需要重新启动的 Splunk Enterprise 更改

应用于搜索时间处理的设置会立即生效，而且不需要重新启动。这是因为搜索在重新加载配置的单独进程中运行。例如，每次搜索都会重新读取查找表、标记和事件类型。

这包括（但不限于）下列更改：

- 查找表
- 字段提取
- 知识对象
- 标记
- 事件类型

包含搜索时间操作的文件包括（但不限于）：

- macros.conf
- props.conf
- transforms.conf
- savedsearches.conf（如果更改创建了一个端点，则必须重新启动。）

要查看您的端点类型，在浏览器中输入以下链接：

<http://yoursplunkserver:8000/en-GB/debug/refresh>

另外，只要您的索引器正在从转发器接收数据，索引时间弹出和转换不需要重启。也就是说：

- 在索引器上更改 props.conf 和 transforms.conf 不需要重新启动。
- 在索引器群集中，对等节点从主节点接收更改时，props.conf 和 transforms.conf 更改会自动重新加载。
- 在非群集索引器上，props.conf 和 transforms.conf 更改需要重新加载。
- 无论在群集索引器还是非群集索引器上，conf 文件重新加载后，更改在转发器 auto-LB 时间段后生效。

## 如何重新加载文件

要重新加载 transforms.conf：

<http://yoursplunkserver:8000/en-us/debug/refresh?entity=admin/transforms-lookup>  
for new lookup file definitions that reside within transforms.conf

<http://yoursplunkserver:8000/en-us/debug/refresh?entity=admin/transforms-extract>  
for new field transforms/extractions that reside within transforms.conf

要重新加载 authentication.conf，使用 Splunk Web。转到设置 > 访问控制 > 验证方法，然后单击重新加载验证配置按钮。此操作刷新验证缓存，但不断开当前用户连接。

## 索引器群集重新启动

要了解索引器群集重新启动、使用滚动重新启动的方法和时间等相关信息，请参阅《管理索引器和索引器群集》中的“重新启动整个索引器群集或单个对等节点”。

## 使用案例

在复杂情况下，重新启动 Splunk Enterprise 最为保险。下面是一些方案的示例，其中您能够（或不能够）避免重新启动。

### 方案：在 props.conf 和 search.conf 中编辑搜索或索引时间转换

是否重新启动取决于更改是否与索引时间设置或搜索时间设置有关。索引时间设置包括：

- 换行
- 时间戳分析

搜索时间设置主要涉及字段提取和创建，不需要重新启动。任何索引时间的更改仍需要重新启动。因此，例如：

1. 如果 props.conf 和 transforms.conf 被配置为在索引上进行搜索时间转换，则您无需做任何操作。对于任何搜索时间的更改，每次您运行搜索时 Splunk 会重新加载 props.conf 和 transforms.conf。

2. 如果在重型转发器上搜索时间发生更改，您必须重新启动该转发器。如果更改是已部署应用（配置为更改后重新启动）的一部分，则它会自动重新启动。

3. 如果它是索引器上的索引时间转换，您必须重新启动索引器以添加更改。

**方案：编辑 `savedsearches.conf` 和新搜索以创建 REST 端点**

您必须重新启动索引器以集成新端点。

## 配置文件列表

下面是与每个配置文件相关联可用的一些规范和示例文件的列表。一些配置文件没有规范或示例文件；在编辑这些没有随附规范和示例文件的配置文件之前，请联系支持部门。

**重要提示：**不要在 `$SPLUNK_HOME/etc/system/default/` 中编辑任何配置文件的默认副本。请参阅[如何编辑配置文件](#)。

文件	用途
<code>alert_actions.conf</code>	创建告警。
<code>app.conf</code>	配置应用属性
<code>audit.conf</code>	配置审计和事件哈希。本功能在该版本中不可用。
<code>authentication.conf</code>	在 Splunk 的内置验证或 LDAP 之间切换，以及配置 LDAP。
<code>authorize.conf</code>	配置角色，包括具体访问控制。
<code>collections.conf</code>	为应用配置 KV 存储集合。
<code>commands.conf</code>	将搜索命令连接到自定义搜索脚本。
<code>crawl.conf</code>	配置 <code>crawl</code> 查找新数据源。
<code>datamodels.conf</code>	用于配置数据模型的属性/值对。
<code>default.meta.conf</code>	为 Splunk 应用对象设置权限。
<code>deploymentclient.conf</code>	指定部署服务器的客户端的行为。
<code>distsearch.conf</code>	指定分布式搜索的行为。
<code>event_renderers.conf</code>	配置 <code>event-rendering</code> 属性。
<code>eventtypes.conf</code>	创建事件类型定义。
<code>fields.conf</code>	创建多值字段，并为索引字段添加搜索功能。
<code>indexes.conf</code>	管理和配置索引设置。
<code>inputs.conf</code>	设置数据输入。
<a href="#">instance.cfg.conf</a>	指定和管理 Splunk 特定实例的设置。这非常方便，例如，在标识转发器进行内部搜索时。
<code>limits.conf</code>	为搜索命令设置各种限制（例如最大结果大小或并发实时搜索）。
<code>literals.conf</code>	自定义在 Splunk Web 中显示的文本，例如搜索错误字符串。
<code>macros.conf</code>	在设置中定义搜索宏。
<code>multikv.conf</code>	为类似表的事件（ <code>ps</code> 、 <code>netstat</code> 、 <code>ls</code> ）配置提取规则。
<code>outputs.conf</code>	设置转发行为。
<code>passwords.conf</code>	为应用保留凭证信息。
<code>procmon-filters.conf</code>	监视 Windows 进程数据。
<code>props.conf</code>	设置索引属性配置，包括时区偏移、自定义来源类型规则和模式冲突优先级。同时，还会将转换映射到事件属性。
<code>pubsub.conf</code>	定义部署服务器的自定义客户端。
<code>restmap.conf</code>	创建自定义 REST 端点。
<code>savedsearches.conf</code>	定义普通报表、计划的报表和告警。

searchbnf.conf	配置搜索助理。
segmenters.conf	配置分段。
server.conf	为 Splunk 的后端（Splunkd 与 Splunk Web 之间的通信）启用 SSL，并指定证书位置。
serverclass.conf	定义与部署服务器一起使用的部署服务器类。
serverclass.seed.xml.conf	配置如何在启动时为装有应用的部署客户端播种。
source-classifier.conf	创建来源类型（例如敏感数据）时的忽略条件。
sourcetypes.conf	用于存储来源类型学习规则的机器生成的文件。
tags.conf	配置字段的标记。
<a href="#">telemetry.conf</a>	启用应用以收集关于应用使用情况和其他属性的遥测数据。
times.conf	定义在搜索应用中使用的自定义时间范围。
transactiontypes.conf	为交易搜索添加附加交易类型。
transforms.conf	配置对数据导入执行的正则表达式转换。在带有 props.conf 的串联中使用。
ui-prefs.conf	为视图更改 UI 首选项。包括为时间范围挑选器更改默认最早及最晚数值。
user-seed.conf	设置默认用户和密码。
<a href="#">visualizations.conf</a>	列出应用可以为系统提供的可视化。
<a href="#">viewstates.conf</a>	使用此文件在 Splunk 中设置 IU 视图（如图表）。
web.conf	配置 Splunk Web，启用 HTTPS。
wmi.conf	设置 Windows management instrumentation (WMI) 输入。
workflow_actions.conf	配置工作流动作。

## 配置参数和数据管道

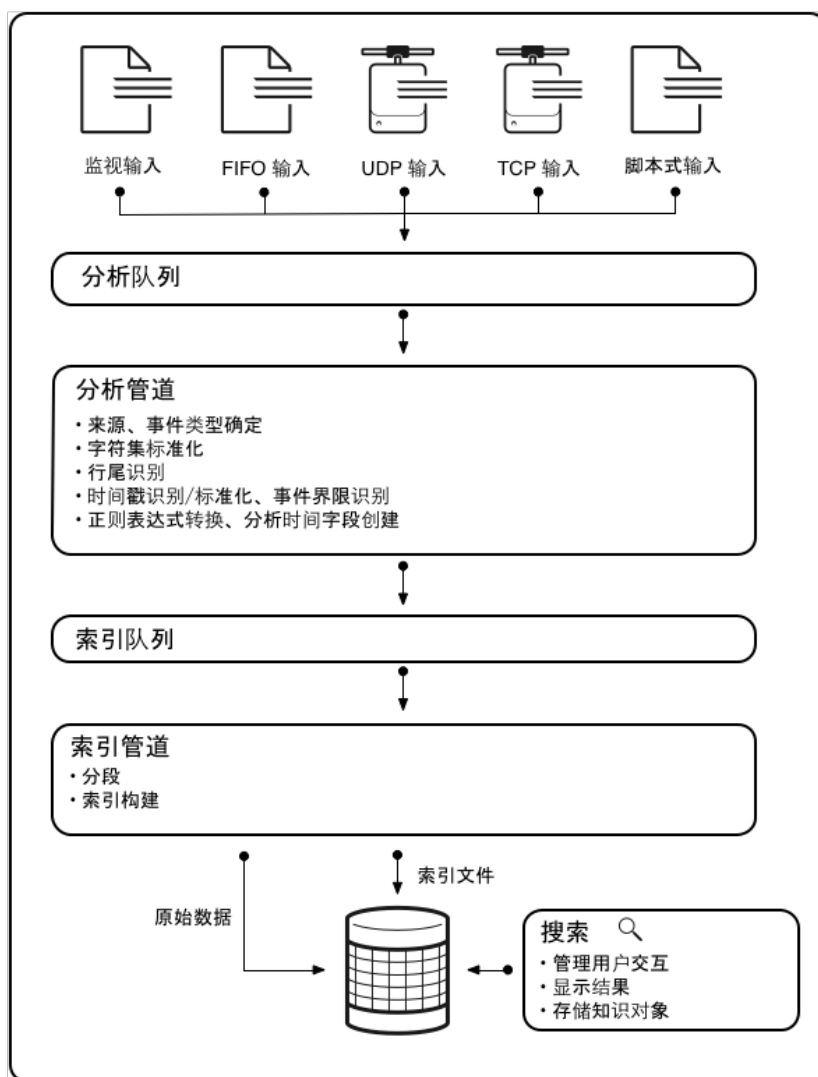
数据在从原始输入转换为可搜索事件过程中经历了几个阶段。此过程称为**数据管道**，由以下四个阶段构成：

- 输入
- 分析
- 索引
- 搜索

数据管道的每个阶段依赖于不同的配置文件参数。知道某一特定参数作用于哪个阶段可让您确定在 Splunk 部署拓扑中需要设置该参数的位置。

### 数据管道的外观

下图概述了数据管道：



在《分布式部署手册》中的“数据如何通过 Splunk：数据管道”中详细介绍了数据管道。

## Splunk Enterprise 组件如何与管道的阶段相关联

一个或多个 Splunk Enterprise 组件可以执行每个管道阶段。例如，通用转发器、重型转发器或索引器可以执行输入阶段。

数据仅在每个阶段经历一次，因此，每个配置仅属于一个组件，特别是处理该阶段的部署中的第一个组件。例如，假定您通过一组通用转发器让数据进入到系统，通用转发器转发数据到中间重型转发器，然后重型转发器又向上转发数据到索引器。在这种情况下，该数据的输入阶段发生在通用转发器中，且分析阶段发生在重型转发器中。

数据管道阶段	可以执行该角色的组件
输入	索引器 通用转发器 重型转发器
分析	索引器 重型转发器 轻型/通用转发器（仅与 INDEXED_EXTRactions 属性结合使用）
索引	索引器
搜索	索引器 搜索头

在哪里配置参数取决于特定部署中的组件。例如，在大多数情况中，您可以在索引器中设置分析参数。但是，如何您让重型转发器向索引器提供数据，则在重型转发器中设置分析参数。同样，您在搜索头（如果有）中设置搜索参数。但是，如果您没有部署专用搜索头，则在索引器中设置搜索参数。



有关更多信息，请参阅《分布式部署手册》中的“组件和数据管道”。

## 配置参数如何与管道的阶段相关联

下面的配置参数与使用这些参数的管道阶段的对应表并不详尽。结合此信息以及您对特定部署中哪个 Splunk 组件执行各个阶段的了解，您可以确定在何处配置每项设置。

例如，如果您要使用通用转发器来获取输入，则需要对转发器配置 `inputs.conf` 参数。但是，如果您的索引器直接获取网络输入，则需要对索引器配置这些网络相关的 `inputs.conf` 参数。

如下阶段中的下列项目按照 Splunk 应用的顺序列出（也就是说 `LINE_BREAKER` 的发生先于 `TRUNCATE`）。

### 输入阶段

- `inputs.conf`
- `props.conf`
  - `CHARSET`
  - `NO_BINARY_CHECK`
  - `CHECK_METHOD`
  - `CHECK_FOR_HEADER`
  - `PREFIX_SOURCETYPE`
  - `sourcetype`
- `wmi.conf`
- `regmon-filters.conf`

### 结构化分析阶段

- `props.conf`
  - `INDEXED_EXTRactions` 和所有其他结构化数据标头提取

### 分析阶段

- `props.conf`
  - `LINE_BREAKER`、`TRUNCATE`、`SHOULD_LINEMERGE`、`BREAK_ONLY_BEFORE_DATE` 和所有其他行合并设置
  - `TIME_PREFIX`、`TIME_FORMAT`、`DATETIME_CONFIG` (`datetime.xml`)、`Tz` 和所有其他时间提取设置和规则
  - `TRANSFORMS` 其中包括按事件队列筛选、按事件索引分配、按事件路由
  - `SEDCMD`
  - `MORE_THAN`、`LESS_THAN`
- `transforms.conf`
  - 由在 `props.conf` 中的子句 `TRANSFORMS` 引用的段落
  - `LOOKAHEAD`、`DEST_KEY`、`WRITE_META`、`DEFAULT_VALUE`、`REPEAT_MATCH`

### 索引阶段

- `props.conf`
  - `SEGMENTATION`
- `indexes.conf`
- `segmenters.conf`

### 搜索阶段

- `props.conf`
  - `EXTRACT`
  - `REPORT`
  - `LOOKUP`
  - `KV_MODE`
  - `FIELDALIAS`
  - `EVAL`
  - `rename`
- `transforms.conf`
  - 由在 `props.conf` 中的子句 `REPORT` 引用的段落
  - `filename`、`external_cmd` 和所有其他查找相关设置
  - `FIELDS`、`DELIMS`
  - `MV_ADD`
- 查找文件夹中的查找文件
- `bin` 文件夹中的搜索和查找脚本
- 搜索命令和查找脚本
- `savedsearches.conf`
- `eventtypes.conf`
- `tags.conf`
- `commands.conf`

- alert\_actions.conf
- macros.conf
- fields.conf
- transactiontypes.conf
- multikv.conf

### 其他配置设置

有一些设置在分布式 Splunk 环境中无法正常工作。有可能出现异常，其中包括：

- props.conf
  - CHECK\_FOR\_HEADER、LEARN\_MODEL、maxDist。这些设置是在分析阶段创建的，但需要将生成的配置移动到搜索阶段配置位置。

## 备份配置信息

Splunk 的所有配置信息都包含在**配置文件**中。要备份这组配置文件，可以创建 `$SPLUNK_HOME/etc/` 的归档或副本。此目录及其子目录包含 Splunk 安装的所有默认和自定义设置以及所有应用，包括保存的搜索、用户帐户、标记、自定义来源类型名称和其他配置信息。

将此目录复制到要恢复的新 Splunk 实例。执行此操作时不需要停止 Splunk。

有关配置文件的更多信息，请阅读[“关于配置文件”](#)。

### 备份群集主节点

如果您正在使用**索引复制**，则可以备份主节点的静态配置。这是配置备用主节点（可以在主要主节点发生故障时进行接管）时的特定使用。有关详细信息，请参阅《管理索引器和群集手册》中的“配置主节点”。

## 检查 Splunk 软件文件完整性

Splunk 软件随附的大多数文件都不应该被最终用户或管理员修改。但是，很多用户会不小心修改这些文件。比如，有人会在默认目录里编辑配置文件，或者文件可能因为硬件问题、文件系统问题、损坏的安装或错误脚本等问题遭到破坏。

Splunk 软件实例文件内容被无效修改时，文件验证能够识别。您可以手动运行此检查，启动时它也会自动运行。

### 手动运行此检查

您可能想要在以下任何一种情况下手动运行完整性检查：

- 升级之后遇到问题。
- 有可疑的存储系统问题症状出现。
- 怀疑存在或者希望防止对默认 conf 文件进行常见的错误编辑。
- 作为例行系统检查的一部分。请参见《*监视 Splunk Enterprise*》手册里的“自定义运行状况检查”。

用默认设置手动运行检查，从安装目录键入 `./splunk validate files`。您可以用两个控件手动运行完整性检查。

- 您可以通过 `-manifest` 指定描述正确文件内容的文件。您可能想要通过此操作检查拙劣的升级之后来自先前安装的旧清单，以验证文件只是过于陈旧。您可以使用任何有效的清单文件。清单文件位于新下载的 Splunk Enterprise 安装目录下。
- 您可以通过使用 `-type conf` 仅对以 `.conf` 结尾的文件进行测试。这是启动时检查打印至终端的消息组。

### 自动验证选项

检查在启动时分两部分运行。

首先，作为 `splunkd` 开始前传输前检查的一部分，检查仅迅速验证默认 conf 文件并将消息写入您的终端。

接下来，`splunkd` 启动后，检查会验证 Splunk Enterprise 随附的所有文件（默认 conf 文件、库、二进制文件、数据文件等）。这一更为完整的检查会将结果写到 `splunkd.log` 和 Splunk Web 中的公告消息系统。您可以在 `limits.conf` 中进行配置。

`limits.conf` 第二部分检查选项包括以下内容：

- 运行和日志
- 运行、日志和将信息输出至 Splunk Web
- 禁用

请参阅 [limits.conf.spec](#)。

阅读安装提供的所有文件对 I/O 性能有一些影响。如果您需要在行中多次重新启动 Splunk 软件，您可能希望暂时

禁用此项检查来提高 I/O 性能。

文件通过对比安装目录的清单文件进行验证。如果文件被删除或者修改，检查就无法正确进行。

监视控制台运行状况检查交互

监视控制台运行状况检查查询 `server/status/installed-file-integrity` 端点。启动时完整性检查运行时，此端点由结果填充。请参阅《REST API 参考手册》中的 `server/status/installed-file-integrity`。

如果 Splunk Enterprise 启动，同时禁用 `limits.conf` 中的完整性检查，则 REST 文件完整性信息将不可用。此外，手动运行不更新结果。

请参阅《监视 Splunk Enterprise》中的“自定义运行状况检查”。

使用命令行界面 (CLI) 管理 Splunk Enterprise

关于 CLI

您可以使用 Splunk 平台的命令行界面 (CLI) 来监视、配置和执行搜索。该产品提供有 CLI 帮助说明，您可以通过终端或 shell 界面进行访问。本主题讨论如何访问此信息。

访问 CLI

Splunk 平台 CLI 命令位于 `$SPLUNK_HOME/bin` (或 Windows 主机的 `%SPLUNK_HOME%\bin`。)

您可以通过点击设置 > 服务器设置 > 常规设置进入 Splunk Web，找到实例中的 Splunk 安装路径。

要访问 Splunk 平台 CLI，您需要：

- Shell 提示符、命令提示符或 PowerShell 会话
- 访问 Splunk 平台实例或转发器或
- 对远程 Splunk 平台实例相应端口的访问权限。

CLI 帮助文档

如果您具有管理员权限，您不但可以使用 CLI 来执行搜索，还可以使用它来配置和监视您的 Splunk 实例。用于配置和监视 Splunk 的 CLI 命令并不是搜索命令。搜索命令是 `search` 和 `dispatch` CLI 命令的参数。某些命令需要您使用用户名和密码进行验证，或者指定目标 Splunk 服务器。

您可以使用以下命令查找 CLI 的帮助信息：

UNIX	Windows
<code>./splunk help</code>	<code>./splunk help</code>

有关如何访问特定 CLI 命令或任务帮助的更多信息，请参阅本手册中的[“获取 CLI 相关帮助”](#)和[“CLI 管理命令”](#)。

在 \*nix 上使用 CLI

如果您具有管理员或 root 权限，您可以将您的 Splunk 平台安装的顶级目录 `$SPLUNK_HOME/bin` 添加到 shell 路径中，以简化 CLI 访问。

此示例适用于在默认位置安装 Splunk Enterprise 的 Linux/BSD/Solaris 用户：

```
# export SPLUNK_HOME=/opt/splunk
# export PATH=$SPLUNK_HOME/bin:$PATH
```

此示例适用于在默认位置安装 Splunk Enterprise 的 Mac 用户：

```
# export SPLUNK_HOME=/Applications/Splunk
# export PATH=$SPLUNK_HOME/bin:$PATH
```

现在，您可以使用如下方式来调用 CLI 命令：

```
./splunk <command>
```

当在 CLI 会话中操作时，要设置 `$SPLUNK_HOME` 环境变量：

- 在 \*nix 中：`source /opt/splunk/bin/setSplunkEnv`
- 在 Windows 中：`splunk.exe envvars > setSplunkEnv.bat & setSplunkEnv.bat`

### Mac OS X 需要提升权限来访问系统文件或目录。

Mac OS X 需要超级用户权限以运行那些访问系统文件或目录的任何命令。请使用 **sudo** 或 "su -"（在新 shell 中作为 root）来运行 CLI 命令。建议使用 **sudo**。（用户 "root" 默认情况下未启用，但任何管理员用户都可以使用 **sudo**。）

## 在 Windows 上使用 CLI

要在 Windows 的 Splunk Enterprise 中运行 CLI 命令，以管理员身份使用 PowerShell 或命令提示符。

1. 以管理员身份打开 PowerShell 窗口或命令提示符。
2. 更改至 Splunk Enterprise `bin` 目录。
3. 运行 Splunk 命令，方法是键入 `splunk`，然后是子命令和任何所需参数。

```
C:\Program Files\Splunk\bin> splunk status
splunkd is running.
splunk helpers are running.
```

您可以从 CLI 中运行诸多命令，并执行多个任务。关于使用 CLI 的帮助，请参阅[获取 CLI 相关帮助](#)。

## 在 Windows 上设置 Splunk 环境变量

在 Windows 上使用 CLI 无需设置 Splunk 环境变量。如果您要使用环境变量来运行 CLI 命令，必须手动设置变量，因为 Windows 不会默认设置变量。

### 暂时设置 Splunk 环境变量

1. 打开 PowerShell 窗口或命令提示符。
2. 从 PowerShell 窗口或命令提示符键入下列命令来临时设置环境变量，或使用“计算机属性”中的“环境变量”对话框来永久设置变量。

PowerShell	命令提示符
<code>\$splunk_home=C:\Program Files\Splunk</code>	<code>set SPLUNK_HOME="C:\Program Files\Splunk"</code>
3. 使用变量运行 Splunk 命令。	
PowerShell	命令提示符
<code>\$splunk_home\bin\splunk status</code>	<code>%SPLUNK_HOME%\bin\splunk add forward-server 192.168.1.100:9997 -auth admin:changeme</code>

### 永久设置 Splunk 环境变量

完成这一步骤后，如果您不更改或删除变量条目，Windows 会针对变量使用您所设置的值。

要永久设置环境变量，请参阅“在 MS TechNet 上添加或更改环境变量”。

## 问答

有什么问题吗？请访问 [Splunk Answers](#) 以查看在 Splunk 社区中围绕使用 CLI 有哪些问题和解答。

## 获取 CLI 相关帮助

本主题介绍如何访问 Splunk 内置的 CLI 帮助参考，其中包含有关 CLI 命令以及如何使用它们的信息。本主题还简要介绍了通用参数，这些参数可以用于任何 CLI 命令。

### 访问 CLI 帮助参考

如果您需要查找某个 CLI 命令或其语法信息，可使用 Splunk 内置的 CLI 帮助参考。

首先，您可以使用 `help` 命令来访问默认的帮助信息：

```
./splunk help
```

此命令将返回一个对象的列表，以帮助您访问更多特定的 CLI 帮助主题，例如，管理命令、群集、转发、许可授权、搜索等。

通用参数

某些命令需要您使用用户名和密码进行验证，或者指定目标主机或应用。对于这些命令，您可以使用以下通用参数之一：auth、app 或 uri。

```
./splunk [command] [object] [-parameter <value> | <value>]... [-app] [-owner] [-uri] [-auth]
```

参数	描述
app	指定要运行命令的应用或命名空间；对于搜索，默认为搜索应用。
auth	指定登录凭据来执行要求登录时所需的命令。
owner	指定与对象关联的 owner/user 上下文；如果未指定，则默认为当前登录的用户。
uri	在任何指定的（远程）Splunk 服务器上执行命令。

app

在 CLI 中，app 是一个适用于许多命令的对象，例如 create app 或 enable app。不过，如果您希望对特定应用运行该命令，您也可以将其作为参数添加到某个 CLI 命令中。

语法：

```
./splunk command object [-parameter value]... -app appname
```

例如，当您在 CLI 中运行搜索时，则默认为搜索应用。如果想要在另一应用中运行搜索：

```
./splunk search "eventtype=error | stats count by source" -deattach f -preview t -app unix
```

auth

如果 CLI 命令要求验证，Splunk 将提示您提供用户名和密码。您还可以使用 -auth 标记来与命令一起传递这些信息。当您需要运行某个要求不同于当前登录用户的执行权限的命令时，auth 参数也非常有用。

注意：auth 必须作为在 CLI 命令的参数中最后指定的参数。

语法：

```
./splunk command object [-parameter value]... -auth username:password
```

uri

如果您要在远程 Splunk 服务器上运行命令，可使用 -uri 标记来指定目标主机。

语法：

```
./splunk command object [-parameter value]... -uri specified-server
```

通过以下格式来指定目标 Splunk 服务器：

```
[http|https]://name_of_server:management_port
```

可以为 name\_of\_server 指定 IP 地址。支持 IPv4 和 IPv6 格式；例如，specified-server 可以使用：127.0.0.1:80 或 "[2001:db8::1]:80"。默认情况下，splunkd 仅在 IPv4 上侦听。要启用 IPv6 支持，请参阅[“配置 Splunk 以使用 IPv6”](#)中的说明。

示例：以下示例从远程 "splunkserver" 的端口 8089 返回搜索结果。

```
./splunk search "host=fflanda error 404 *.gif" -auth admin -uri https://splunkserver:8089
```

有关您可以在远程服务器上运行的 CLI 命令的更多信息，请参阅本章中的[下一个主题](#)。

有用的帮助主题

在运行默认的 Splunk CLI 帮助时，您会看到列出的以下对象。

## CLI 管理命令

可以使用 CLI 执行各种管理功能，例如，添加或编辑输入、更新配置设置和搜索。如果想要查看 CLI 管理命令类型的列表，请键入：

```
./splunk help commands
```

这些命令将在本手册下一个主题[“CLI 管理命令”](#)中进行更详细的介绍。

## 群集的 CLI 帮助

索引复制（也称为群集）是一项由索引器群集组成的 Splunk 功能，这些索引器已配置为复制数据以实现若干目标：数据可用性、数据保真度、灾难容错和获得改进的搜索性能。

您可以使用 CLI 来查看和编辑群集主节点或群集对等节点上的群集配置。要获得与群集相关的命令和参数的列表，请键入：

```
./splunk help clustering
```

有关更多信息，请参阅《[管理索引器和群集](#)》手册中的“使用 CLI 配置群集”。

## Splunk 控制的 CLI 帮助

可使用 CLI 来启动、停止和重新启动 Splunk 服务器 (`splunkd`) 和 web (`splunkweb`) 进程，或者检查相关进程是否正在运行。要获得控制的列表，请键入：

```
./splunk help controls
```

有关更多信息，请参阅《[管理员手册](#)》中的[“启动和停止 Splunk”](#)。

## 数据管理的 CLI 帮助

向 Splunk 添加数据时，Splunk 将对数据进行处理并将其存储在索引中。默认情况下，您提供给 Splunk 的数据会存储在 **main** 索引中，但您可以使用 CLI 为 Splunk 创建和指定其他索引，以用于其他数据导入。要查看用于管理索引和数据存储区的对象和命令的列表，请键入：

```
./splunk help datastore  
./splunk help index
```

有关更多信息，请参阅《[管理索引器和群集](#)》手册中的“关于管理索引”、“创建自定义索引”和“从 Splunk 删除索引和数据”。

## 分布式搜索部署的 CLI 帮助

可使用 CLI 查看和管理分布式搜索配置。要获得对象和命令的列表，请键入：

```
./splunk help distributed
```

有关分布式搜索的信息，请阅读《[分布式搜索](#)》手册中的“关于分布式搜索”。

## 转发和接收的 CLI 帮助

Splunk 部署可包括数十个或数百个将数据转发到一个或多个接收器的转发器。可使用 CLI 查看和管理数据转发配置。要获得转发对象和命令的列表，请键入：

```
./splunk help forwarding
```

有关更多信息，请参阅《[转发数据](#)》手册中的“关于转发和接收”。

## 搜索和实时搜索的 CLI 帮助

您还可以使用 CLI 来运行历史搜索和实时搜索。可以使用以下命令访问 Splunk 搜索和实时搜索的帮助页面：

```
./splunk help search  
./splunk help rtsearch
```

此外，还可以使用 `search-commands`、`search-fields` 和 `search-modifiers` 对象来访问各自的帮助说明和语法：

```
./splunk help search-commands  
./splunk help search-fields  
./splunk help search-modifiers
```

**注意：**Splunk CLI 将空格解释为换行。对于包含多个单词的主题名称，应在单词之间使用短划线。

要了解有关使用 CLI 搜索数据的更多信息，请参阅《搜索参考手册》中的“关于 CLI 搜索”和“CLI 搜索语法”以及《搜索手册》中的“CLI 中的实时搜索和报表”。

## CLI 管理命令

本主题将介绍 CLI 管理命令，也就是用于管理和配置 Splunk 服务器和分布式部署的命令。

有关访问 CLI 的信息以及 CLI 帮助中所含内容的信息，请参阅上一主题[“获取 CLI 相关帮助”](#)。如果您要查找如何从 CLI 运行搜索的详细信息，请参阅《搜索参考手册》中的“关于 CLI 搜索”。

您的 Splunk 角色配置决定了您可以执行哪些操作（命令）。多数操作需要您具备 Splunk 管理员身份。有关设置和管理 Splunk 用户与角色的更多信息，请参阅《管理员手册》中的[“关于用户和角色”](#)主题。

### Splunk CLI 命令语法

CLI 命令的一般语法为：

```
./splunk <command> [<object>] [--<parameter>] <value>...
```

请注意以下事项：

- 某些命令不需要对象或参数。
- 某些命令具有可由其值单独指定的默认参数。

### 命令、对象和示例

**命令**是您可以执行的操作。您在**对象**上执行操作。

命令	对象	示例
add	exec, forward-server, index, licenser-pools, licenses, master, monitor, oneshot, saved-search, search-server, tcp, udp, user	<b>1.向来源 /var/log 中添加监视目录和文件输入。</b>  ./splunk add monitor /var/log/
		<b>2.向实例（搜索头在其间进行搜索）列表中添加另一个主节点。</b>  ./splunk add cluster-master https://127.0.0.1:8089 -secret testsecret -multisite false'
anonymize	来源	<b>1.取代位于 /tmp/messages 的文件中的标识数据，如用户名和 IP 地址。</b>  ./splunk anonymize file -source /tmp/messages
		<b>2.使用 name-terms（包含常见英文名列表的文件）使 Mynames.txt 匿名。</b>  ./splunk anonymize file -source /tmp/messages -name_terms \$SPLUNK_HOME/bin/Mynames.txt
apply	cluster-bundle	<b>1.在对等节点上激活已验证的软件包。</b>  ./splunk apply cluster-bundle
		<b>2.Skip-validation 是一个可选参数，可将其配置来跳</b>

		<p>过在主节点和对等节点上的软件包验证。</p> <pre>./splunk apply cluster-bundle --skip-validation&lt;/code&gt;</pre>
clean	all, eventdata, globaldata, inputdata, userdata, kvstore	<p><b>1.从 Splunk 安装中删除数据。</b> eventdata 索引为原始日志文件的导出事件。</p> <pre>./splunk clean eventdata</pre> <p><b>2.globaldata 指主机标记和来源类型别名。</b></p> <pre>./splunk clean globaldata</pre>
cmd	btool, classify, locktest, locktool, parsetest, pcregextest, regextest, searchtest, signtool, walklex	<p><b>1.运行带有各种环境变量设置的 splunk btool inputs list 命令字符串。运行 splunk envvars 以查看设置了哪些环境变量。</b></p> <pre>./splunk cmd btool inputs list</pre> <p><b>2.显示 bin 目录的内容。</b></p> <pre>./splunk cmd /bin/ls</pre>
Create	app	<p><b>1.根据模板构建 myNewApp。</b></p> <pre>./splunk create app myNewApp -template sample_app</pre>
createssl	无	
diag	无	
disable	app, boot-start, deploy-client, deploy-server, dist-search, index, listen, local-index, maintenance-mode, perfmon, webserver, web-ssl, wmi	<p><b>1.在索引器群集中对等节点上禁用维护模式。必须在主节点上调用。</b></p> <pre>'./splunk disable maintenance-mode'</pre> <p><b>2.禁用 logs1 集合。</b></p> <pre>./splunk disable eventlog logs1</pre>
display	app, boot-start, deploy-client, deploy-server, dist-search, jobs, listen, local-index	<p><b>1.为所有应用显示状态信息，如启用/禁用。</b></p> <pre>./splunk display app</pre> <p><b>2.为 unix 应用显示状态信息。</b></p> <pre>./splunk display app unix</pre>
edit	app, cluster-config, shcluster-config, exec, index, licenser-localslave, licenser-groups, monitor, saved-search, search-server, tcp, udp, user	<p><b>1.编辑当前的群集配置。</b></p> <pre>./splunk edit cluster-config -mode slave -site site2</pre> <p><b>2.编辑 /var/log 中的监视目</b></p>



		<p>录输入，并且只从该文件的结尾读取。</p> <pre>./splunk edit monitor /var/log -follow-only true</pre>
enable	app, boot-start, deploy-client, deploy-server, dist-search, index, listen, local-index, maintenance-mode, perfmon, webserver, web-ssl, wmi	<p><b>1. 在索引器群集中对等节点上设置维护模式。必须在主节点上调用。</b></p> <pre>'./splunk enable maintenance-mode'</pre> <p><b>2. 启用 coll 集合。</b></p> <pre>./splunk enable perfmon coll</pre>
export	eventdata, user data	<p><b>1. 将数据从 Splunk 服务器导出到</b> /tmp/apache_raw_404_logs 中。</p> <pre>./splunk export eventdata - index my_apache_data -dir /tmp/apache_raw_404_logs - host localhost -terms "404 html"</pre>
fsck	repair, scan, clear-bloomfilter	
help	无	
import	userdata	<p><b>1. 从目录 /tmp/export.dat 导入用户帐户数据。</b></p> <pre>./splunk import userdata - dir /tmp/export.dat</pre>
install	app	<p><b>1. 从 foo.tar 安装应用到本地 Splunk 服务器。</b></p> <pre>./splunk install app foo.tar</pre> <p><b>2. 从 foo.tgz 安装应用到本地 Splunk 服务器。</b></p> <pre>./splunk install app foo.tgz</pre>
list	cluster-buckets, cluster-config, cluster-generation, cluster-peers, deploy-clients, excess-buckets, exec, forward-server, index, inputstatus, licenser-groups, licenser-localslave, licenser-messages, licenser-pools, licenser-slaves, licenser-stacks, licenses, jobs, master-info, monitor, peer-info, peer-buckets, perfmon, saved-search, search-server, tcp, udp, user, wmi	<p><b>1. 列出所有活跃的监视目录和文件输入。这显示当前或最近被 Splunkd 监视到发生更改的文件和目录。</b></p> <pre>./splunk list monitor</pre> <p><b>2. 列出所有堆叠间的所有许可证。</b></p> <pre>./splunk list licenses</pre>
login,logout	无	
offline	无	<p><b>1. 用于以不影响现有搜索的方式关闭对等节点。主节点为数据桶重新安排主要对等节点，并在设置了 enforce-counts 标记的情况下修复群集状态。</b></p> <pre>./splunk offline</pre>

		<p><b>2. 因为使用了 <code>--enforce-counts</code> 标记，在该对等节点停止之前该群集已完全修复。</b></p> <pre>./splunk offline --enforce-counts</pre>
package	app	<p><b>1. 打包 stubby 应用并返回它的 uri。</b></p> <pre>./splunk package app stubby</pre>
rebuild	无	
refresh	deploy-clients	
reload	ad, auth, deploy-server, index, listen, monitor, registry, script, tcp, udp, perfmon, wmi	<p><b>1. 重新加载部署服务器，整个加载或通过服务器类加载。</b></p> <pre>./splunk reload deploy-server</pre> <p><b>2. 重新加载 my_serverclass。</b></p> <pre>./splunk reload deploy-server -class my_serverclass</pre>
remove	app, cluster-peers, excess-buckets, exec, forward-server, index, jobs, licenser-pools, licenses, monitor, saved-search, search-server, tcp, udp, user	<p><b>1. 从实例（搜索头在其间进行搜索）列表中移除群集主节点。使用 testsecret 作为 secret/pass4SymmKey。</b></p> <pre>'./splunk remove cluster-master https://127.0.0.1:8089 -secret testsecret'</pre> <p><b>2. 移除 Unix 应用。</b></p> <pre>./splunk remove app unix</pre>
rolling-restart	cluster-peers, shcluster-members	
rtsearch	app, batch, detach, earliest_time, header, id, index_earliest, index_latest, max_time, maxout, output, preview, rt_id, timeout, uri, wrap	<p><b>1. 对于单独的行运行实时搜索，以避免自动换行。</b></p> <pre>./splunk rtsearch 'error' -wrap false</pre> <p><b>2. 运行实时搜索。正如使用传统的搜索命令一样使用 rtsearch。</b></p> <pre>./splunk rtsearch 'eventtype=webaccess error   top clientip'</pre>
search	app, batch, detach, earliest_time, header, id, index_earliest, index_latest, latest_time, max_time, maxout, output, preview, timeout, uri, wrap	<p><b>1. 使用通配符作为搜索对象。触发异步搜索并显示搜索的任务 id 和 ttl。</b></p> <pre>./splunk search '*' -detach true</pre> <p><b>2. 使用 eventtype=webaccess</b></p>

		<p><code>error</code> 作为搜索对象。行的长度超过终端宽度时不进行自动换行。</p> <pre>./splunk search 'eventtype=webaccess error' -wrap 0</pre>
set	datastore-dir, deploy-poll, default-hostname, default-index, minfreemb, servername, server-type, splunkd-port, web-port, kvstore-port	<p><b>1. 设置强制索引准备位。</b></p> <pre>./splunk set indexing-ready</pre> <p><b>2. 设置 <code>bologna:1234</code> 作为部署服务器以向其轮询更新。</b></p> <pre>./splunk set deploy-poll bologna:1234</pre>
show	config, cluster-bundle-status, datastore-dir, deploy-poll, default-hostname, default-index, jobs, minfreemb, servername, splunkd-port, web-port, kvstore-port	<p><b>1. 显示当前日志级别。</b></p> <pre>./splunk show log-level</pre> <p><b>2. 显示 Splunk Enterprise 被配置为向哪个部署服务器轮询。</b></p> <pre>./splunk show deploy-poll</pre>
spool	无	
start,stop,restart	splunkd, splunkweb	
status	splunkd, splunkweb	
validate	index	<p><b>1. 使用 <code>main</code> 作为索引来验证。验证 <code>indexes.conf</code> 中指定的索引路径。</b></p> <pre>./splunk validate index main</pre>
version	无	

## 使用 CLI 导出搜索结果

您可以使用 CLI 导出大量的搜索结果。有关如何使用 CLI 导出搜索结果的信息，以及有关 Splunk Enterprise 提供的其他导出方法的信息，请参阅《搜索手册》中的“导出搜索结果”。

## CLI 故障排除

Splunk 的 CLI 还包含可帮助对 Splunk 问题进行故障排除的工具。可以使用 Splunk CLI 命令 `cmd` 调用这些工具：

```
./splunk cmd <tool>
```

要获得 CLI 实用工具的列表，请参阅《故障排除手册》中的“提供支持的命令行工具”。

## 使用 CLI 来管理远程 Splunk Enterprise 实例

您可以对任何 CLI 命令使用 `uri` 参数，以便将此命令发往另一个 Splunk Enterprise 实例，并在您的本地服务器上查看结果。

本主题介绍：

- 有关使用 `uri` 参数的语法。
- 您无法远程使用的 CLI 命令。

**注意：**默认情况下禁止管理员用户的远程 CLI 访问，除非您已更改了默认密码。

## 启用远程访问

如果您在使用 Splunk Free（无登录凭据），则默认情况下禁止远程访问，除非您编辑 `$SPLUNK_HOME/etc/system/local/server.conf` 并设置以下值：

```
allowRemoteLogin=always
```

**注意：** `add oneshot` 命令工作于本地实例，但不能远程使用。

有关编辑配置文件的更多信息，请参阅本手册中的[关于配置文件](#)。

## 将 CLI 命令发往远程服务器

对任何 CLI 命令使用 `uri` 参数的一般语法为：

```
./splunk command object [-parameter <value>]... -uri <specified-server>
```

`uri` 值 `specified-server` 的格式为：

```
[http|https]://name_of_server:management_port
```

此外，`name_of_server` 可以是远程 Splunk Enterprise 实例的完全解析域名或 IP 地址。

**重要提示：** 该 `uri` 值是您在远程 Splunk Enterprise 实例的 `web.conf` 中定义的 `mgmtHostPort` 值。有关更多信息，请参阅本手册中的[web.conf 参考](#)。

有关 CLI 的一般信息，请参阅本手册中的[关于 CLI](#) 和[获取 CLI 相关帮助](#)。

### 搜索远程实例

以下示例从远程 "splunkserver" 返回搜索结果。

```
./splunk search "host=fflanda error 404 *.gif" -uri https://splunkserver:8089
```

有关使用 CLI 来执行搜索语法的详细信息，请参阅《[搜索参考手册](#)》中的“关于 CLI 搜索”。

### 查看远程实例上已安装的应用

以下示例会返回在远程 "splunkserver" 上所安装应用的列表。

```
./splunk display app -uri https://splunkserver:8089
```

## 更改您的默认 URI 值

您可以使用 `SPLUNK_URI` 环境变量来设置默认 URI 值。如果您将此值更改为远程服务器的 URI，则在每次要访问远程服务器时无需包含 `uri` 参数。

要更改 `SPLUNK_URI` 的值，键入：

```
$ export SPLUNK_URI=[http|https]://name_of_server:management_port # For Unix shells
C:\> set SPLUNK_URI=[http|https]://name_of_server:management_port # For Windows shell
```

对于上述示例，您可以键入以下命令来更改 `SPLUNK_URI` 的值：

```
$ export SPLUNK_URI=https://splunkserver:8089
```

## 您无法远程运行的 CLI 命令

除了控制服务器的命令之外，您可以远程运行所有其他 CLI 命令。这些服务器控制命令包括：

- start, stop, restart
- status, version

通过访问 CLI 帮助参考，可查看所有 CLI 命令。请参阅本手册中的[获取 CLI 相关帮助](#)。

## 自定义 CLI 登录横幅

如果您提供 CLI 对数据的访问权限，则可能需要自定义登录横幅以通知您的用户监视情况、其法律义务和误用处

罚。也可以为您的 CLI 登录添加其他安全性（采用基本验证的形式）。

要创建自定义登录横幅并添加基本验证，请将下列段落添加到本地 `server.conf` 文件中：

```
[httpServer]
cliLoginBanner = <string>
allowBasicAuth = true|false
basicAuthRealm = <string>
```

- 对于 `cliLoginBanner = <string>`

创建系统在提示输入验证凭据之前希望您的用户在 Splunk CLI 中看到的消息，如访问策略信息。默认值是无消息。

要创建多行横幅，请将各行置于逗号分隔列表中，每一行都用双引号引起来。例如：

```
cliLoginBanner="Line 1","Line 2","Line 3"
```

要在横幅文本中包括双引号，请在行中使用两个引号。例如：

```
cliLoginBanner="This is a line that "contains quote characters"!"
```

- 对于 `allowBasicAuth = true|false`：

如果您希望除了 Splunk 的现有 (`authtoken`) 验证外，客户端还需使用 HTTP Basic 验证对 Splunk 服务器进行已验证的请求，则将此值设置为 `true`。这对允许有规划地访问 REST 端点以及从 Web 浏览器访问 REST API 都非常有用。UI 或 CLI 则不需要此设置。默认值为 `true`。

- 对于 `basicAuthRealm = <string>`：

如果您已启用 `allowBasicAuth`，请使用此属性添加在提示凭据时可显示在 Web 浏览器中的文本字符串。您可以显示描述服务器和/或访问策略的短消息。默认情况下，显示的文本为 `"/splunk"`。

## 启动 Splunk Enterprise 并执行初始任务

### 启动和停止 Splunk Enterprise

本主题提供有关启动和停止 Splunk Enterprise 的简要说明。

#### 在 Windows 上启动 Splunk Enterprise

在 Windows 上，Splunk Enterprise 默认安装在 `C:\Program Files\Splunk` 中。Splunk 文档中的许多示例都使用 `$SPLUNK_HOME` 来表示 Splunk 安装目录。如果您将 Splunk Enterprise 安装在默认目录中，则可将字符串 `$SPLUNK_HOME`（在 Windows 上为 `%SPLUNK_HOME%`）替换为 `C:\Program Files\Splunk`。

Splunk Enterprise 安装了两个服务：`splunkd` 和 `splunkweb`。正常操作时，仅 `splunkd` 运行，处理所有 Splunk Enterprise 操作，包括 Splunk Web 界面。要更改这种情况，您需要将 Splunk Enterprise 置于旧模式中。请参阅“[在旧模式中，在 Windows 上启动 Splunk Enterprise](#)。”

在 Windows 上，您可以通过以下方法之一来启动和停止 Splunk：

**1. 通过 Windows 服务控制面板（从 `Start -> Control Panel -> Administrative Tools -> Services` 访问）来启动和停止 Splunk Enterprise 进程**

- 服务器守护程序和 Web 界面：`splunkd`
- Web 界面（仅在旧模式中）：`splunkweb`。在正常操作时，此服务启动，并在收到启动请求时立即退出。

**2. 从命令提示符下，使用 `NET START <service>` 或 `NET STOP <service>` 命令来启动和停止 Splunk Enterprise 服务：**

- 服务器守护程序和 Web 界面：`splunkd`
- Web 界面（仅在旧模式中）：`splunkweb`。在正常操作时，此服务启动，并在收到启动请求时立即退出。

**3. 前往 `%SPLUNK_HOME%\bin` 并键入以下命令，即可同时启动、停止或重新启动这两个进程：**

```
> splunk [start|stop|restart]
```

#### 在旧模式中，在 Windows 上启动 Splunk Enterprise

如果您要在旧模式中运行 Splunk Enterprise，其中 `splunkd` 和 `splunkweb` 都运行，则您必须更改配置参数。

**重要提示：切勿永久在传统模式下运行 Splunk Web。** 使用传统模式临时解决由用户界面与主 `splunkd` 服务的新集成引入的问题。一旦您纠正此问题，尽快将 Splunk Web 返回到正常模式。

要将 Splunk Enterprise 置于旧模式：

1.从命令提示符，转到 `%SPLUNK_HOME%\etc\system\local`。

2.编辑 `%SPLUNK_HOME%\etc\system\local\web.conf`，如果不存在，则在 `%SPLUNK_HOME%\etc\system\local` 中创建以 `web.conf` 命名的新文件。请参阅[如何编辑配置文件](#)。

3.在 `web.conf` 中，将 `appserverPorts` 和 `httpport` 属性设置如下：

```
[settings]
appServerPorts = 0
httpport = 8000
```

4.保存文件并将其关闭。

5.重新启动 Splunk Enterprise。`splunkd` 和 `splunkweb` 服务启动并保持运行。

6.通过浏览 `http://<server name>:<httpport>` 和输入凭据，登录 Splunk Enterprise。

要恢复正常的 Splunk Enterprise 操作：编辑 `%SPLUNK_HOME%\etc\system\local\web.conf` 来删除 `appServerPorts` 和 `httpport` 属性。

## 在 UNIX 上启动 Splunk Enterprise

Splunk Enterprise 安装时，有一个进程在 \*nix，`splunkd` 上。正常操作时，仅 `splunkd` 运行，处理所有 Splunk Enterprise 操作，包括 Splunk Web 界面。要更改这种情况，您需要将 Splunk Enterprise 置于旧模式中。请参阅[在旧模式中，在 Unix 上启动 Splunk Enterprise](#)。”

### 启动 Splunk Enterprise

在 Splunk Enterprise 服务器主机上，从 shell 提示符运行以下命令：

```
# splunk start
```

**注意：**如果您已将 Splunk Enterprise 配置为在开机时启动，则您应该使用服务命令启动 Splunk Enterprise。这将确保在 `init.d` 脚本中配置的用户可启动软件。

```
# service splunk start
```

这将启动 `splunkd`（索引器和 Splunk Web 界面）。

要分别启动它们，键入：

```
# splunk start splunkd
```

或

（仅在旧模式中）`# splunk start splunkweb`

**注意：**如果 `startwebserver` 属性已禁用，或 `appServerPorts` 属性在 `web.conf` 中设置为任何非 0 的值，则手动启动 `splunkweb` 不会进行任何操作。`splunkweb` 过程将不会在这两者中的任一情况中启动。请参阅[在旧模式中，在 Unix 上启动 Splunk Enterprise](#)。”

要重新启动 Splunk Enterprise（`splunkd` 或 `splunkweb`），键入：

```
# splunk restart
```

```
# splunk restart splunkd
```

（仅在旧模式中）`# splunk restart splunkweb`

### 在旧模式中，在 Unix 上启动 Splunk Enterprise

如果您要在 `splunkd` 和 `splunkweb` 都运行的情况下运行 Splunk Enterprise，则您必须将 Splunk Enterprise 置于旧模式中。

要将 Splunk Enterprise 置于旧模式：

1.从 shell 提示符，转到 `$SPLUNK_HOME/etc/system/default`。

2.创建 `web.conf` 的副本并将其放入 `$SPLUNK_HOME/etc/system/local` 中。

3.在 `$SPLUNK_HOME/etc/system/local` 中编辑 `web.conf`。

4.在 `web.conf` 中，将 `appserverPorts` 和 `httpport` 属性设置如下：

```
[settings]
appServerPorts = 0
httpport = 8000
```

5. 保存文件并将其关闭。

6. 重启启动 Splunk Enterprise（请参阅“在 Unix 上启动 Splunk Enterprise”）。splunkd 和 splunkweb 服务启动并保持运行。

7. 通过浏览 `http://<server name>:<httpport>` 和输入凭据，登录 Splunk Enterprise。

要恢复正常的 Splunk Enterprise 操作：编辑 `%SPLUNK_HOME%\etc\system\local\web.conf` 并删除 `appServerPorts` 和 `httpport` 属性。

## 停止 Splunk Enterprise

要关闭 Splunk Enterprise，运行以下命令：

```
# splunk stop
```

要分别停止 splunkd 和 Splunk Web，键入：

```
# splunk stop splunkd
```

或

（仅在旧模式中）`# splunk stop splunkweb`

## 检查 Splunk 是否正在运行

要检查 Splunk Enterprise 是否正在运行，在服务器主机上，从 shell 提示符下键入以下命令：

```
# splunk status
```

您应当可以看到以下输出：

```
splunkd is running (PID: 3162).
splunk helpers are running (PIDs: 3164).
```

如果 Splunk Enterprise 在旧模式中运行，则您将在输出中看到额外一行：

```
splunkweb is running (PID: 3216).
```

**注意：**在 Unix 系统上，您必须以运行 Splunk Enterprise 的用户身份登录以运行 `splunk status` 命令。其他用户无法读取所需文件以正确地报告状态。

如果 `splunk status` 决定服务运行，则将返回状态代码 0 或成功。如果 `splunk status` 决定服务不运行，则将返回针对非运行服务 Linux Standard Base 值 3。其他值可能会显示 `splunk status` 出现错误。

您还可以使用 `ps` 来检查正在运行的 Splunk Enterprise 进程：

```
# ps aux | grep splunk | grep -v grep
```

Solaris 用户应使用 `ps` 的 `-ef` 参数，而非 `aux`：

```
# ps -ef | grep splunk | grep -v grep
```

## 从 Splunk Web 重新启动 Splunk Enterprise

您还可以从 Splunk Web 重新启动 Splunk：

1. 导航到系统 > 服务器控件。

2. 单击重新启动 Splunk。

这将重新启动 splunkd 和 splunkweb 过程（仅在旧模式中）。

## 配置 Splunk 在开机时启动

在 Windows 上，Splunk 会默认为在开机时启动。要禁用此特性，请参阅本主题末尾的“在 Windows 上禁用开机

时启动”。

在 \*nix 平台上，您需要配置 Splunk 以在开机时启动。

## 在 \*nix 平台上启用开机时启动

Splunk 提供了一个工具，它可以更新您的系统启动配置，以便使 Splunk 能够在系统启动时启动。该工具会创建适当的 `init` 脚本（或做出类似的配置更改，具体取决于您的操作系统）。

以 root 用户身份运行：

```
$SPLUNK_HOME/bin/splunk enable boot-start
```

如果不以根用户身份启动 Splunk，您可以传递 `-user` 参数以指定以哪个用户身份启动 Splunk。例如，如果要以用户 bob 的身份运行 Splunk，则作为根用户，您可以运行：

```
$SPLUNK_HOME/bin/splunk enable boot-start -user bob
```

如果您打算禁止 Splunk 在开机时启动，则运行：

```
$SPLUNK_HOME/bin/splunk disable boot-start
```

在 `$SPLUNK_HOME/etc/init.d/README` 中提供了更多相关信息，您也可以在命令行中键入 `help boot-start`。

## Mac 用户注意事项

Splunk 会自动在目录中创建一个脚本和配置文件：`/System/Library/StartupItems`。此脚本在系统开机时运行，并且在系统关机时自动停止 Splunk。

**注意：**如果您在使用 Mac OS，您**必须**具备 root 级权限（或使用 **sudo**）。您需要管理员权限以使用 **sudo**。

**示例：**

要在 Mac OS 上于系统开机时启用 Splunk，请使用：

仅 CLI：

```
./splunk enable boot-start
```

CLI 和 **sudo**：

```
sudo ./splunk enable boot-start
```

## 在 Windows 上禁用开机时启动

默认情况下，Splunk 会在您启动 Windows 计算机时自动启动。您可以配置 Splunk 进程（`splunkd` 和 `splunkweb`）以从 Windows 服务控制面板中手动启动它们。

## 安装您的许可证

首次下载 Splunk 时，将要求您注册。

通过注册，您可以获得一份临时（60 天）的 Enterprise Trial 许可证，此许可证允许您每天创建的最大索引量为 500 MB。该许可证会包含在您的下载中。

Enterprise 许可证启用以下功能：

- 多个用户帐户和访问控制。
- 分布式搜索和数据路由。
- 部署管理。

有关 Splunk 许可授权的更多信息，请参阅本手册中的[Splunk 许可授权如何运作](#)。

## 从哪里获得新许可证？

当您请求新的许可证时，您将通过来自 Splunk 的电子邮件收到许可证。您还可以从您的 `splunk.com` 中的“我的订单”页面获得新的许可证。

要通过 Splunk Web 来安装和更新您的许可证，请导航到**设置 > 许可授权**，并遵循[这些说明](#)。



## 更改默认值

在您开始针对您的环境来配置 Splunk Enterprise 之前，请查看下列默认设置，以确定是否需要更改某些设置。

### 设置或更改环境变量

您可以通过在操作系统上设置环境变量以更改 Splunk Enterprise 的启动方式。

在 \*nix 上，使用 `setenv` 或 `export` 命令来设置特定变量。例如：

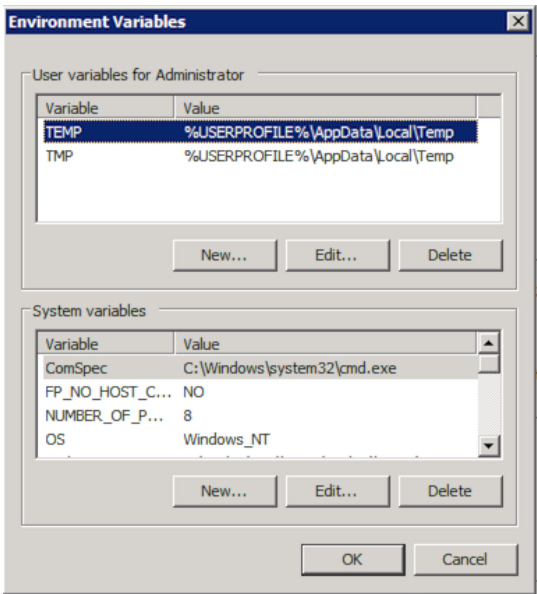
```
# export SPLUNK_HOME = /opt/splunk02/splunk
```

如果您想要永久地设置环境，编辑相应的 shell 初始化文件，并为您希望 Splunk Enterprise 启动时使用的变量添加条目。

在 Windows 上，在命令提示符或 PowerShell 窗口中使用 `set` 环境变量：

```
C:\> set SPLUNK_HOME = "C:\Program Files\Splunk"
```

如果您想要永久地设置环境，使用“环境变量”窗口以添加条目到“用户变量”列表中。



有几个可用的环境变量：

环境变量	用途
SPLUNK_HOME	Splunk Enterprise 安装目录的完全限定路径。
SPLUNK_DB	目录（包含 Splunk Enterprise 索引目录）的完全限定路径。
SPLUNK_BINDIP	Splunk Enterprise 应该在启动时绑定以接受连接的系统 IP 地址。当主机有超过一个活动的 IP 地址时非常有用。
SPLUNK_IGNORE_SELINUX	指示 Splunk Enterprise 在启用了 SELinux 的 Linux 主机上运行时尝试启动。默认情况下，当 Splunk Enterprise 检测到 SELinux 处于活跃状态时会立即退出。该变量使此检查无效，并能用于已配置 SELinux 的方案中以允许 Splunk Enterprise 运行。
SPLUNK_OS_USER	指示 Splunk Enterprise 假定您指定的用户凭据，无论您作为哪个用户启动它。例如，如果您在系统中指定用户 'splunk' 并作为 root 用户启动 Splunk Enterprise，它采用 'splunk' 用户的权限，而且任何由这些过程写入的文件都将属于 'splunk' 用户所有。
SPLUNK_SERVER_NAME	splunkd 服务（在 Windows 上）或过程（在 *nix 上）的名称。切勿设置该变量，除非您知道您在做什么。
SPLUNK_WEB_NAME	splunkweb 服务（在 Windows 上）或过程（在 *nix 上）的名称。切勿设置该变量，除非您知道您在做什么。

您也可以通过编辑 `splunk-launch.conf`（在一些情况下，编辑 `web.conf`）来为每个实例编辑这些环境变量。当您在主机上运行超过一个 Splunk 实例时，这会有所帮助。请参阅“[splunk-launch.conf](#)”。

## 更改 admin 默认密码

使用 Enterprise 许可证的 Splunk 具有默认的管理帐户和密码，admin/changeme。Splunk 强烈建议您更改默认的管理帐户和密码以保证系统安全。您的密码应该不宜过于简单，并且需要遵照一般密码设置最佳实践：

- 使用单词、数字、符号及大小写字母组合。
- 密码要尽量复杂，尽可能长。我们建议密码长度至少为 10 个字符。
- 不要使用诸如您的生日、社保号、电话号码或家人姓名等作为密码，因为可能不是很可靠。
- 不要使用能够在词典中找到的单词。
- 不要使用您在其他地方使用或之前使用过的密码。

### 使用 Splunk Web

要更改管理员默认密码：

1. 以管理员用户身份登录 Splunk Web。
2. 单击界面右上方的**设置**。
3. 在屏幕的“用户和验证”部分中单击**访问控制**。
4. 单击**用户**。
5. 单击 **admin** 用户。
6. 更新密码，并单击**保存**。

### 使用 Splunk CLI

Splunk CLI 命令为：

```
splunk edit user
```

**重要提示：**您必须首先使用现有密码进行验证，然后才能更改密码。通过 CLI 或使用 `-auth` 参数登录 Splunk。例如，该命令将管理员密码从 `changeme` 更改为 `foo`：

```
splunk edit user admin -password foo -role admin -auth admin:changeme
```

**注意：**在 \*nix 操作系统上，shell 会将某些特殊字符解释为命令指令。您必须对这些字符进行转义，方法是分别在各字符之前加上 `\` 或者将密码用单引号 (') 括起。例如：

```
splunk edit user admin -password 'FFL14io!23ur$' -role admin -auth admin:changeme
```

或

```
splunk edit user admin -password FFL14io!23ur\$ -role admin -auth admin:changeme
```

在 Windows 上，使用脱字符 (^) 对保留的 shell 字符进行转义或者将密码用双引号 (") 括起。例如：

```
splunk edit user admin -password "FFL14io!23ur>" -role admin -auth admin:changeme
```

或

```
splunk edit user admin -password FFL14io!23ur^> -role admin -auth admin:changeme
```

**注意：**您还可以跨服务器一次重置所有密码。有关过程，请参阅“跨多个服务器部署密码”。

## 更改网络端口

Splunk 在安装时配置两个端口：

- **HTTP/HTTPS 端口。**该端口为 Splunk Web 提供套接字。默认端口为 8000。
- **管理端口。**该端口用于与 `splunkd` 守护程序通信。Splunk Web 在此端口上与 `splunkd` 通信。此外，命令行界面和任何来自其他服务器的分布式连接也均采用此端口。默认端口为 8089。

**重要提示：**在安装期间，您可能会将这些端口设置为默认值以外的其他值。

**注意：**对于从转发器接收数据的 Splunk 实例，必须配置另外一个端口，即接收器端口。它们使用此端口来侦听来

自转发器的传入数据。在安装期间不进行此配置。默认接收器端口是 9997。有关更多信息，请参阅《转发数据》手册中的“启用接收器”。

### 使用 *Splunk Web*

要将这些端口从安装时的设置值更改为其他值：

1. 以管理员用户身份登录 Splunk Web。
2. 单击界面右上方的**设置**。
3. 在屏幕的“系统”部分中单击**服务器设置**链接。
4. 单击**常规设置**。
5. 更改**管理端口**或 **Web 端口**的值，并单击**保存**。

### 使用 *Splunk CLI*

要通过 Splunk CLI 来更改端口设置，请使用 CLI 命令 `set`。例如，该命令将 Splunk Web 端口设为 9000：

```
splunk set web-port 9000
```

该命令将 splunkd 端口设为 9089：

```
splunk set splunkd-port 9089
```

## 更改默认 Splunk 服务器名称

Splunk 服务器名称设置同时控制在 Splunk Web 中显示的名称和在分布式设置中发送给其他 Splunk 服务器的名称。

默认名称来自 DNS 或 Splunk 服务器主机的 IP 地址。

### 使用 *Splunk Web*

要更改 Splunk 服务器名称：

1. 以管理员用户身份登录 Splunk Web。
2. 单击界面右上方的**设置**。
3. 在屏幕的“系统”部分中单击**服务器设置**链接。
4. 单击**常规设置**。
5. 更改 **Splunk 服务器名称**的值，并单击**保存**。

### 使用 *Splunk CLI*

要通过 CLI 来更改服务器名称，使用 `set servername` 命令。例如，该命令将服务器名称设置为 `foo`：

```
splunk set servername foo
```

## 更改数据存储区位置

数据存储区是 Splunk 服务器用于存储所有索引数据的顶级目录。

**注意：**如果您更改此目录，服务器将不会迁移旧的数据存储区文件。相反，它会从新的位置重新开始。

要将您的数据迁移到另一个目录，请遵循“移动索引”中的说明。

### 使用 *Splunk Web*

要更改数据存储区位置：

1. 以管理员用户身份登录 Splunk Web。
2. 单击界面右上方的**设置**。
3. 在屏幕的“系统”部分中单击**系统设置**链接。

4.单击常规设置。

5.更改索引路径中的路径，并单击保存。

6.使用 CLI 重新启动 Splunk。导航到 `$SPLUNK_HOME/bin/` (\*nix) 或 `%SPLUNK_HOME%\bin` (Windows)，然后运行以下命令：

```
splunk restart
```

**重要提示：**不要使用“设置”中的重新启动功能。这不会达到更改索引目录的预期效果。您必须从 CLI 重新启动。

### 使用 Splunk CLI

要通过 CLI 来更改数据存储区目录，使用 `set datastore-dir` 命令。例如，该命令将数据存储区目录设置为 `/var/splunk/`：

```
splunk set datastore-dir /var/splunk/
```

### 设置最小可用磁盘空间

最小可用磁盘空间设置会控制在 Splunk 停止建立索引之前，数据存储区位置中磁盘空间不足的最低情况。

如果可用磁盘空间增加，则 Splunk 会恢复建立索引的状态。

### 使用 Splunk Web

要设置最小可用磁盘空间：

- 1.以管理员用户身份登录 Splunk Web。
- 2.单击界面右上方的设置。
- 3.在屏幕的“系统”部分中单击系统设置链接。
- 4.单击常规设置。
- 5.更改可用磁盘空间低于此值时暂停建立索引的值，并单击保存。

### 使用 Splunk CLI

要通过 CLI 更改最小可用空间值，使用 `set minfreemb` 命令。例如，该命令将最小可用空间设置为 2000 MB：

```
splunk set minfreemb 2000
```

### 设置默认的时间范围

“搜索和报表”应用中的特殊搜索默认时间范围设置为**所有时间**。管理员可全局设置所有应用的默认时间范围。设置存储在 `[general_default]` 段落的 `SPLUNK_HOME/etc/apps/user-prefs/local/user-prefs.conf` 文件中。

此设置适用于 Splunk Apps 的所有搜索页面，而非仅“搜索和报表”应用。此设置适用于所有用户角色。

**注意：**此设置不适用于仪表板。

### 使用 Splunk Web

- 1.以管理员用户身份登录 Splunk Web。
- 2.单击设置。
- 3.在“系统”部分，单击服务器设置。
- 4.单击搜索首选项。
- 5.在默认搜索时间范围下拉菜单中，选择要使用的时间并单击保存。

### ui\_prefs.conf 文件中的时间范围设置

对于特定应用程序或用户，您可能在 `ui-prefs.conf` 文件中已有时间范围设置。`ui-prefs.conf` 文件中的设置优先于使用 Splunk Web 对全局默认时间范围所做的所有设置。

但如果想要所有应用程序和用户都使用全局默认时间范围，可考虑移除 `ui-prefs.conf` 文件中的设置。

## 其他默认设置

在 Splunk Web 设置的“常规设置”屏幕上还有其他一些您可能想要更改的默认设置。请仔细浏览屏幕以查看这些选项。

## 另请参阅

关于配置文件

`user-prefs.conf`

`ui-prefs.conf`

## 将 Splunk 绑定到某个 IP

您可以强制 Splunk 将其端口绑定到某个特定的 IP 地址。默认情况下，Splunk 会绑定到 IP 地址 0.0.0.0，即所有可用的 IP 地址。

更改 Splunk 的绑定 IP 仅适用于 Splunk 守护程序 (splunkd)，其侦听端口为：

- TCP 端口 8089（默认）
- 针对以下功能配置的任何端口：
  - SplunkTCP 输入
  - TCP 或 UDP 输入

要将 Splunk Web 进程 (splunkweb) 绑定到特定 IP，请使用 `web.conf` 中的 `server.socket_host` 设置。

## 暂时

要暂时更改，请在启动 Splunk 之前设置环境变量 `SPLUNK_BINDIP=<ipaddress>`。

## 永久

如果您要对工作环境做出永久性更改，请修改 `$SPLUNK_HOME/etc/splunk-launch.conf` 以包括 `SPLUNK_BINDIP` 属性和 `<ipaddress>` 值。例如，要将 Splunk 端口绑定到 127.0.0.1（仅本地环回），`splunk-launch.conf` 应为：

```
# Modify the following line to suit the location of your Splunk install.
# If unset, Splunk will use the parent of the directory this configuration
# file was found in
#
# SPLUNK_HOME=/opt/splunk
SPLUNK_BINDIP=127.0.0.1
```

**重要提示：** `web.conf` 中 `mgmtHostPort` 属性具有默认值 127.0.0.1:8089。因此，如果您将 `SPLUNK_BINDIP` 更改为 127.0.0.1 以外的任何值，则还必须更改 `mgmtHostPort` 以使用同一 IP 地址。例如，如果您在 `splunk-launch.conf` 中做出此改动：

```
SPLUNK_BINDIP=10.10.10.1
```

您还必须在 `web.conf` 中做出此改动（假设管理端口为 8089）：

```
mgmtHostPort=10.10.10.1:8089
```

请查看 `web.conf` 以了解有关 `mgmtHostPort` 属性的更多信息。

## IPv6 注意事项

从版本 4.3 起，`web.conf` `mgmtHostPort` 设置已扩展为可以接受使用方括号括起的 IPv6 地址。因此，如果您将 `splunkd` 配置为仅在 IPv6 上侦听（通过本手册中的“[配置 Splunk 以使用 IPv6](#)”所描述的 `server.conf` 相关设置），则必须将其从 127.0.0.1:8089 更改为 `:::1:8089`。

## 配置 Splunk 以使用 IPv6

本主题介绍 Splunk 对 IPv6 的支持，以及如何配置它。在执行本主题中的程序之前，您可能需要：

- 阅读本手册中的“[关于配置文件](#)”，以了解 Splunk 配置文件如何工作
- 阅读《数据导入》手册中的“从 TCP 和 UDP 端口获取数据”
- 阅读本手册中的“[server.conf](#)”，以了解在 `server.conf` 配置文件中可用的选项参考。

- 阅读本手册中的 ["inputs.conf"](#)，以了解在 `inputs.conf` 配置文件中可用的选项参考。

从版本 4.3 起，Splunk 开始支持 IPv6。用户可以通过 IPv6 网络连接到 Splunk Web、使用 CLI 并转发数据。

## IPv6 平台支持

所有 Splunk 支持的操作系统平台（请参阅《安装手册》中的“支持的操作系统”）都会支持使用 IPv6 配置，但以下操作系统除外：

- HPUX PA-RISC
- Solaris 8 和 9
- AIX

## 配置 Splunk 以在 IPv6 网络上侦听

在配置 Splunk 以在 IPv6 网络上侦听时，您可以选择下列选项。您可以将 Splunk 配置为：

- 仅连接到 IPv6 地址，然后忽略来自 DNS 的所有 IPv4 结果
- 连接到 IPv4 和 IPv6 地址，且
  - 首先尝试 IPv6 地址
  - 首先尝试 IPv4 地址
- 仅连接到 IPv4 地址，然后忽略来自 DNS 的所有 IPv6 结果

要配置 Splunk 以在 IPv6 上侦听：编辑位于 `$SPLUNK_HOME/etc/system/local` 中的 `server.conf` 的副本，添加以下内容：

```
listenOnIPv6=[yes|no|only]
```

- `yes` 表示 `splunkd` 将侦听来自 IPv6 和 IPv4 的连接。
- `no` 表示 `splunkd` 将只在 IPv4 上侦听，**此为默认设置**。
- `only` 表示 Splunk 将只在 IPv6 上侦听传入连接。

```
connectUsingIpVersion=[4-first|6-first|4-only|6-only|auto]
```

- `4-first` 表示 `splunkd` 将首先尝试连接到 IPv4 地址，如果失败，则尝试连接到 IPv6。
- `6-first` 与 `4-first` 相反。这是大多数启用 IPv6 的客户端应用（如 Web 浏览器）采取的策略，但可能在 IPv6 部署的早期阶段较不可靠。
- `4-only` 表示 `splunkd` 将忽略来自 DNS 的任何 IPv6 结果。
- `6-only` 表示 `splunkd` 将忽略来自 DNS 的任何 IPv4 结果。
- `auto` 表示 `splunkd` 将根据 `listenOnIPv6` 设置选取合理的策略。**此为默认值**。
  - 如果 `splunkd` 只在 IPv4 上侦听，则其行为与您指定 `4-only` 时的情况一致。
  - 如果 `splunkd` 只在 IPv6 上侦听，则其行为与您指定 `6-only` 时的情况一致。
  - 如果 `splunkd` 同时在二者上侦听，则其行为与您指定 `6-first` 时的情况一致。

**重要提示：**这些设置只会影响 DNS 查找。例如，设置 `connectUsingIpVersion = 6-first` 将不会导致采用显式 IPv4 地址（如 `"server=10.1.2.3:9001"`）的段落无法工作。

## 如果您只有少量输入，并且不打算针对整个部署启用 IPv6

如果您在 IPv6 上只有少量数据来源，并且不希望为您的整个 Splunk 部署启用它，您可以将上述的 `listenOnIPv6` 设置添加到 `inputs.conf` 中的任何 `[udp]`、`[tcp]`、`[tcp-ssl]`、`[splunktcp]` 或 `[splunktcp-ssl]` 段落。这将覆盖对应输入的 `server.conf` 中相同名称的设置。

## 在 IPv6 上转发数据

您的 Splunk 转发器可以在 IPv6 上转发数据。在 `outputs.conf` 中支持以下设置：

- 在 `[tcpout]` 段落中的 `server` 设置可以包含标准 `[host]:port` 格式中的 IPv6 地址。
- `[tcpout-server]` 段落可以接受标准 `[host]:port` 格式中的 IPv6 地址。
- 在 `[syslog]` 段落中的 `server` 设置可以包含标准 `[host]:port` 格式中的 IPv6 地址。

## 针对 IPv6 的分布式搜索配置

您的 Splunk 分布式搜索部署可以使用 IPv6。在 `distsearch.conf` 中支持以下设置：

- `servers` 设置可以包含标准 `[host]:port` 格式中的 IPv6 地址。
- 不过，`heartbeatMcastAddr` 并未经过更新以支持 IPv6 地址。该设置已在 Splunk 4.3 中被弃用，并将从未来的产品版本中删除。

## 在 IPv6 上访问 Splunk Web

如果您的网络策略允许或需要来自 Web 浏览器的 IPv6 连接，您可以配置 `splunkweb` 服务以使其行为不同于 `splunkd`。从版本 4.3 起，`web.conf` 开始支持 `listenOnIPv6` 设置。该设置的行为与其在 `server.conf` 中（如上所述）

完全相同，但仅适用于 Splunk Web。

现有的 `web.conf` `mgmtHostPort` 设置已扩展为可以接受使用方括号括起的 IPv6 地址。因此，如果您将 `splunkd` 配置为仅在 IPv6 上侦听（通过上述 `server.conf` 中的设置），则必须将其从 `127.0.0.1:8089` 更改为 `:::1:8089`。

## Splunk CLI 和 IPv6

Splunk CLI 可以在 IPv6 上与 `splunkd` 通信。如果您在 `web.conf` 中设置 `mgmtHostPort`，定义了 `$SPLUNK_URI` 环境变量，或使用 `-uri` 命令行选项，则可以做到这一点。当使用 `-uri` 选项时，请确保使用方括号将 IPv6 IP 地址括起，并用引号将整个地址和端口引起来，例如：`-uri "[2001:db8::1]:80"`。

## IPv6 和 SSO

如果您在 IPv6 上使用 SSO，则不必对 `trustedIP` 属性使用方括号，如下例所示。这同时适用于 `web.conf` 和 `server.conf`。

在下面的 `web.conf` 示例中，`mgmtHostPort` 属性使用方括号，但 `trustedIP` 属性未使用：

```
[settings]
mgmtHostPort = :::1:8089
startwebserver = 1
listenOnIPv6=yes
trustedIP=2620:70:8000:c205:250:56ff:fe92:1c7,::1,2620:70:8000:c205::129
SSOMode = strict
remoteUser = X-Remote-User
tools.proxy.on = true
```

有关 SSO 的更多信息，请参阅《确保 Splunk Enterprise 安全》手册中的“配置单一登录”。

## 确保配置安全

如果您的配置尚不安全，是时候确保 Splunk 和您的数据安全了。采取适当的步骤以确保 Splunk 减少攻击面并缓解大多数漏洞的风险和影响。

安装后应采取的一些重要措施：

- 设置用户和角色。您可以使用 Splunk 本机验证配置用户和/或使用 LDAP 管理用户。请参阅“关于用户验证”。
- 设置证书验证 (SSL)。Splunk 随附了一系列默认证书，这些证书应被替换以确保验证安全。我们提供添加 SSL 加密和验证以及配置安全验证的指导原则和进一步说明。

《确保 Splunk Enterprise 安全》手册提供可确保 Splunk 安全的方法的更多相关信息。包括检查表以对您的配置进行强化。有关更多信息，请参阅“确保 Splunk Enterprise 安全”。

## 共享性能数据

您可以选择自动与 Splunk, Inc. 共享许可证使用情况及部署性能的特定相关数据。Splunk 根据这些数据来决定未来的产品开发，不会将您的信息共享给第三方。

### 选择加入或退出

对于以下两种数据类型，您可以选择发送两种、发送其中一种或不发送任何一种：

- **许可证使用情况数据**，描述您已激活的许可证，以及您索引的数据数量，
- **匿名使用情况数据**，与部署性能相关。

您以管理员或等同于管理员的身份在搜索头上首次运行 Splunk Web 时，会显示一种模式。

- 单击**跳过**，为执行这一操作的用户永久抑制模式。使用此选项将决定留给其他管理员。
- 单击**确定**为所有用户永久抑制此模式。

您也可以通过导航到**设置 > 工具**随时选择加入或者退出。

如果您选择退出，收集您系统上数据的搜索将不再运行，不会发送任何数据。

启用或禁用工具的功能由 `edit_telemetry_settings` 操作控制。

### 收集的数据

对于任何一种数据，您都可以在 Splunk Web 中查看发送了哪些数据。

1. 导航到**设置 > 工具**。
2. 在相关数据类别（“匿名使用数据”或“许可证使用数据”）下，单击**查看日志**。
3. 单击**查看数据**。

本数据日志只有在集合首次运行后才可见（请参阅[功能空间](#)）。如果要在生产环境选择加入前查看将发送哪些数据，您可以在沙盒环境下选择加入。

匿名使用数据不与客户帐户绑定，并且仅在聚合分析时使用。请注意收集时，匿名使用数据不加密。收到的数据会安全地存储在本地服务器中，严格限制为仅有聚合分析时才能访问。收集的许可证 ID 仅以下两种情况使用：验证数据是否从有效的 Splunk 产品中接收，协助分析不同 Splunk 产品如何在用户人数之间部署。

以下表格介绍您选择加入两个程序后将收集的数据。数据格式为 JSON，并带有名为“组件”的字段标记。

描述	组件	注意
已激活的许可证组和子组	licensing.stack	
许可证堆叠配额总计、许可证池消耗总计、许可证堆叠类型	licensing.stack	
许可证池配额、许可证池消耗	licensing.stack	
许可证 ID	licensing.stack	始终发送，但仅针对选择加入许可证使用报告的用户保留。
索引器群集中节点数量、索引器群集的复制因子和搜索因子	deployment.clustering.indexer	
GUID、主机、根据类型（虚拟/实体）分类的内核数、CPU 架构、内存大小、存储容量（分区）、OS/版本、Splunk 版本	deployment.node	针对每个索引器或搜索头
主机数量、Splunk 实例数量、OS/版本、CPU 架构、Splunk 软件版本、转发量分布	deployment.forwarders	针对转发器
核心利用率、存储利用率、内存使用、索引吞吐量、搜索延迟	deployment.node performance.indexing performance.search	
索引量、事件数量、主机数量、来源类型名称	usage.indexing.sourcetype	
活跃用户的数量	usage.users.active	
每种搜索的数量、并发搜索的分布	usage.search.type usage.search.concurrent	
应用名称、页面名称、局域设置、用户数量、页面加载数量	usage.app.page	

## 数据样本

单击**扩展**查看收集的数据样本。

组件	数据类别	示例
deployment.clustering.indexer	群集配置	<pre>{   "host": "docteam-unix-5",   "summaryReplication": true,   "siteReplicationFactor": null,   "enabled": true,   "multiSite": false,   "searchFactor": 2,   "siteSearchFactor": null,   "timezone": "-0700",   "replicationFactor": 3 }</pre>
		<pre>{   "hosts": 168,   "instances": 497, }</pre>



deployment.forwarders	转发器架构、转发量	<pre> "architecture": "x86_64", "os": "Linux", "splunkVersion": "6.5.0", "type": "uf", "kb": {   "min": 389,   "max": 2291497,   "total": 189124803,   "p10": 40960,   "p20": 139264,   "p30": 216064,   "p40": 269312,   "p50": 318157,   "p60": 345088,   "p70": 393216,   "p80": 489472,   "p90": 781312 } </pre>
deployment.node	主机架构、利用率	<pre> {   "guid": "123309CB-ABCD-4BB9-9B6A-185316600F23",   "host": "docteam-unix-3",   "os": "Linux",   "osExt": "Linux",   "osVersion": "3.10.0-123.el7.x86_64",   "splunkVersion": "6.5.0",   "cpu": {     "coreCount": 2,     "utilization": {       "min": 0.01,       "p10": 0.01,       "p20": 0.01,       "p30": 0.01,       "p40": 0.01,       "p50": 0.02,       "p60": 0.02,       "p70": 0.03,       "p80": 0.03,       "p90": 0.05,       "max": 0.44     },     "virtualCoreCount": 2,     "architecture": "x86_64"   },   "memory": {     "utilization": {       "min": 0.26,       "max": 0.34,       "p10": 0.27,       "p20": 0.28,       "p30": 0.28,       "p40": 0.28,       "p50": 0.29,       "p60": 0.29,       "p70": 0.29,       "p80": 0.3,       "p90": 0.31     },     "capacity": 3977003401   },   "disk": {     "fileSystem": "xfs",     "capacity": 124014034944,     "utilization": 0.12   } } </pre>
		<pre> {   "type": "download-trial",   "guid": "4F735357-F278-4AD2-BBAB-139A85A75DBB",   "product": "enterprise", </pre>

licensing.stack	许可证配额及消耗	<pre> "name": "download-trial", "licenseIDs": [   "553A0D4F-3B7B-4AD5-B241-89B94386A07F" ], "quota": 524288000, "pools": [   {     "quota": 524288000,     "consumption": 304049405   } ], "consumption": 304049405, "subgroup": "Production", "host": "docteam-unix-9" } </pre>
performance.indexing	索引吞吐量和索引量	<pre> {   "host": "docteam-unix-5",   "thruput": {     "min": 412,     "max": 9225,     "total": 42980219,     "p10": 413,     "p20": 413,     "p30": 431,     "p40": 450,     "p50": 474,     "p60": 488,     "p70": 488,     "p80": 488,     "p90": 518   } } </pre>
performance.search	搜索运行时间统计信息	<pre> {   "latency": {     "min": 0.01,     "max": 1.33,     "p10": 0.02,     "p20": 0.02,     "p30": 0.05,     "p40": 0.16,     "p50": 0.17,     "p60": 0.2,     "p70": 0.26,     "p80": 0.34,     "p90": 0.8   } } </pre>
usage.app.page	应用页用户和视图	<pre> {   "app": "search",   "locale": "en-US",   "occurrences": 1,   "page": "datasets",   "users": 1 } </pre>
usage.indexing.sourcetype	通过来源类型索引	<pre> {   "name": "vendor_sales",   "bytes": 2026348,   "events": 30245,   "hosts": 1 } </pre>
	搜索并发	<pre> {   "host": "docteam-unix-5"   "searches": {     "min": 1,     "max": 11,     "p10": 1,     "p20": 1,     "p30": 1, </pre>

usage.search.concurrent	技术文档	<pre>         "p40": 1,         "p50": 1,         "p60": 1,         "p70": 1,         "p80": 2,         "p90": 3       }     } </pre>
usage.search.type	按类型搜索	<pre> {   "ad-hoc": 1428,   "scheduled": 225 } </pre>
usage.users.active	活跃用户	<pre> {   "active": 23 } </pre>

## 未被收集的数据

以下类型的数据未收集：

- 用户名或密码。
- 插入 Splunk 平台实例的索引数据。

## 为什么要发送许可证使用情况数据

有些许可证项目需要您提供许可证使用情况报告。最简单易行的方式是选择自动将信息发送至 Splunk。

如果您不选择加入自动许可证数据共享，仍可以手动发送这些数据。在 Splunk Web 中导航至**设置 > 工具**并遵循说明将数据导出到本地目录。

## 功能空间

每天上午 3:05 开始汇总并发送数据。

### 关于搜索

如果您选择加入，则 Splunk Enterprise 部署中的一个实例将通过特殊搜索收集数据。所有搜索按序列运行，从上午 3:05 开始。所有搜索由脚本式输入触发。请参阅“配置计划报表的优先级”。

### 运行搜索的节点

您的部署中仅有一个节点运行搜索以收集使用情况数据。具体哪个实例取决于您的部署细节：

- 在索引群集环境下，搜索在群集主节点上运行。
- 如果启用的是搜索头群集，而不是索引器群集，则搜索将通过搜索头管理员运行。
- 如果您的部署没有使用群集，则搜索通过搜索头运行。

### 关于内部日志文件

如果您启用了许可证使用情况报告，则产品工具首次运行时会在 `$SPLUNK_HOME/var/log/splunk` 创建一个新文件。文件名为 `license_usage_summary.log`，文件大小不超过 25MB。文件将索引至新的内部索引 `_telemetry`。索引默认保留两年，大小不超过 256MB。

搜索运行后，数据打包发送至 Splunk, Inc.。

应用驻留在 `$SPLUNK_HOME/etc/apps/splunk_instrumentation` 的文件系统。

# 配置 Splunk 许可证

## Splunk Enterprise 许可授权如何运作

Splunk Enterprise 从您指定的来源获取数据并加以处理，以便您进行分析。我们将此过程称为建立索引。有关准确的建立索引过程的信息，请参阅《数据导入手册》中的“Splunk 软件如何处理您的数据”。

Splunk Enterprise 许可证规定了您在每个日历日（从前一天午夜到当天午夜，以许可证主服务器上的时钟为准）可以建立索引的数据量。

在您的 Splunk Enterprise 基础设施中，任何执行索引操作的主机都必须获得相应的许可授权。您可以使用本地安

装的许可证来运行独立的索引器，也可以将某个 Splunk Enterprise 实例配置为**许可证主服务器**，并从其他被配置为**许可证从服务器**的索引器中建立**许可证池**，并从中选取。

除了索引量之外，访问某些 Splunk Enterprise 功能也需要 Enterprise 许可证。有关不同许可证类型的更多信息，请阅读[Splunk 许可证类型](#)。

## 关于许可证主服务器和许可证从服务器之间的连接

在配置了许可证主服务器实例，并为其添加了许可证从服务器之后，许可证从服务器每分钟会向许可证主服务器告知一次其使用情况。如果由于某种原因无法连接到许可证主服务器，许可证从服务器会启动 72 小时计时器。如果许可证从服务器连续 72 小时无法与许可证主服务器通信，则许可证从服务器上的搜索操作将被阻止（仍然继续执行索引操作）。用户将无法搜索许可证从服务器上的索引数据，除非可以再次连接到许可证主服务器。

## Splunk Enterprise 许可证生命周期

当您首次安装所下载的 Splunk Enterprise 副本时，该实例使用 Enterprise Trial 许可证，试用期为 60 天。该许可证允许您在 60 天的试用期内使用 Splunk Enterprise 的所有功能，每天最多可以建立 500 MB 的数据索引。

一旦 60 天试用期满后（并且您没有购买并安装 [Enterprise 许可证](#)），您可以选择切换到 Splunk Free。Splunk Free 包括 [Splunk Enterprise 的功能子集](#)，专门用于独立部署的使用和短期取证调查。它允许您每天最多对 500 MB 数据建立索引而不会过期。

**重要提示：** Splunk Free 不包括验证、计划搜索或告警功能。这意味着任何用户都可以访问您的安装（通过 Splunk Web 或 CLI）而无需提供凭据。此外，计划的已保存搜索或告警将不再会被触发。

如果您希望在 60 天试用期满后继续使用 Splunk Enterprise 功能，则必须购买 Enterprise 许可证。请联系 Splunk 销售代表以了解更多信息。

在您购买并下载 Enterprise 许可证之后，您可以在您的实例上安装它，然后就可以访问 Splunk Enterprise 功能了。请阅读本手册中的“[Splunk 许可证类型](#)”以了解 Enterprise 功能相关信息。

有关升级现有许可证的信息，请参阅《[安装手册](#)》中的“迁移到新的 Splunk Enterprise 许可证”。

## Splunk 软件许可证类型

每个 Splunk 软件实例都需要一个许可证。Splunk 许可证指定了给定的 Splunk 平台实例可以建立索引的数据量以及您有权访问的功能。本主题介绍不同的许可证类型及相关选项。

通常有以下几种许可证类型：

- Enterprise 许可证启用所有 Enterprise 功能，例如验证和分布式搜索。自 Splunk Enterprise 6.5.0 起，新的 Enterprise 许可证为非强制许可证。
- Free 许可证允许建立有限的索引量且可永久使用，但禁用验证。
- 转发器许可证允许您转发数据和启用验证，但不允许对数据建立索引。
- Beta 许可证通常启用 Enterprise 功能，但仅限于 Splunk Beta 版本。
- 高级应用许可证与 Enterprise 或 Cloud 许可证结合使用以访问应用的功能。

本主题中还介绍部署包括分布式搜索或索引器群集时的许可注意事项。

有关升级 4.2 之前版本的许可证信息，请参阅《[安装手册](#)》中的“迁移到新的 Splunk Enterprise 许可证”。

## Splunk Enterprise 许可证

Splunk Enterprise 是标准 Splunk 软件许可证。它允许您使用所有 Splunk Enterprise 功能，包括验证、分布式搜索、部署管理、告警计划和基于角色的访问控制。Enterprise 许可证需要购买，它在索引量上没有限制。请联系 Splunk 销售代表以获取更多信息。

以下列出了其他类型的 Enterprise 许可证，它们均包含相同的功能：

### **非强制许可证**

如果您的许可证主服务器正在运行 Splunk Enterprise 6.5.0 或更新的版本，则可以使用非强制 Enterprise 许可证。这一新许可证类型允许用户继续搜索，即使您在 30 天窗口收到五次警告。您的许可证主服务器仍然自认为属于违规，但搜索不会受阻。

### **Enterprise Trial 许可证**

当您首次下载 Splunk 时，您将被要求进行注册。通过注册，您可以获得一份 Enterprise **Trial** 许可证，在此许可证下您每天可以建立的最大索引量为 500 MB。从您开始使用 Splunk 算起，Enterprise Trial 许可证将在 60 天后失效。如果您采用 Enterprise Trial 许可证来运行 Splunk，那么在您的许可证失效之后，Splunk 将要求您切换到 [Splunk Free 许可证](#)。

在您安装 Splunk 软件之后，您可以选择使用 Enterprise Trial 许可证来运行 Splunk（直到该许可证到期），购买 Enterprise 许可证或[切换到 Free 许可证](#)（该许可证已包括在内）。

**注意：**Enterprise Trial 许可证有时也称为 "download-trial"。

### **Sales Trial 许可证**

如果您使用 Splunk Sales，您可以请求试用大小和持续时间不同的多个 Enterprise 许可证。从您开始使用 Splunk 算起，Enterprise Trial 许可证将在 60 天后失效。如果您准备试用大规模的部署，并要求较长的持续时间，或者想要在试用期间对较大数据量建立索引，您可以直接与 Splunk Sales 或您的销售代表联系，提出您的请求。

### **开发/测试许可证**

用特定的许可证程序可以获取开发/测试许可证，以在沙盘环境下操作 Splunk 软件。如果部署正在使用开发/测试许可证，则所有用户都能在 Splunk Web 导航栏左边看见开发/测试戳。

**警告：**开发/测试许可证未通过 Enterprise 许可证堆叠。如果您用 Enterprise 许可证安装开发/测试许可证，Enterprise 许可证文件将会被覆盖。

### **Free 许可证**

Free 许可证包括每天 500 MB 的索引量，免费（供您使用），没有截止日期。

以下功能在使用 Enterprise 许可证时可用，但在 **Splunk Free** 中禁用：

- 多个用户帐户和基于角色的访问控制
- 分布式搜索
- 以 TCP/HTTP 格式转发（您可以向其他 Splunk 软件实例转发数据，但不能向非 Splunk 软件实例转发）
- 部署管理（包括客户端）
- 告警/监视
- 验证和用户管理，包括本机验证、LDAP 和脚本式验证。
  - 不存在登录机制。不会提示您输入用户名/密码，通过命令行或浏览器可以访问和控制 Splunk 软件的所有方面。
  - 您无法添加更多角色或创建用户帐户。
  - 在运行搜索时将针对所有公共索引 'index=\*'，且不支持如用户配额、最大每个搜索时间范围和搜索过滤条件的搜索限制。
  - 功能系统被禁用，为访问 Splunk 软件的所有用户启用全部操作。

请参阅[有关 Splunk Free 的更多信息](#)。

### **对比许可证功能**

请参阅本表对比主要许可证类型。

行为或功能	Enterprise 6.5.0 之前的版本	非强制 Enterprise	开发测试 Enterprise	开发测试个性化	Enterprise Trial	Free
违规时阻止搜索	是	否	各有不同	各有不同	是	是
在警告或违规时进行内部记录并将信息显示在 Splunk Web	是	是	是	是	是	是
具有其他许可证	是	是	否	否	是	是
Enterprise 完整功能集	是	是	是	否	是	否

### **转发器许可证**

本许可证允许转发无限量数据（但不允许建立索引），还在实例上启用安全性，以使用户必须提供用户名和密码才能访问。（Free 许可证也可用于转发无限量数据，但没有安全性。）

转发器许可证包括在 Splunk 中；不需要单独购买。

Splunk 提供几个转发器选项：

- **通用转发器**自动启用/应用许可证；安装后不需要再执行其他步骤。
- 轻型转发器使用相同的许可证，但必须通过手动更改为转发器许可证组来启用。

- 重型转发器也必须手动转换为转发器许可证组。如果要执行建立索引，应向实例授予对 Enterprise 许可证堆叠的访问权限。阅读本手册中的“[组、堆叠、池和其他术语](#)”以获得有关 Splunk 许可证术语的详细信息。

## Beta 许可证

Splunk 的 Beta 版本需要不同的许可证，此许可证与其他 Splunk 版本不兼容。此外，如果您要评估 Beta 版本的 Splunk，使用 Free 或 Enterprise 许可证时该版本不会运行。Beta 许可证通常启用 Enterprise 功能，但这些功能仅限在 Beta 版本中使用。如果您要评估 Beta 版本的 Splunk，该版本将附带自己的许可证。

## 搜索头的许可证（用于分布式搜索）

**搜索头**是 Splunk 实例，将搜索分布到其他 Splunk 索引器。虽然搜索头通常不在本地对任何数据建立索引，但您仍想要使用许可证限制对搜索头的访问。

没有特别用于搜索头的特殊许可证类型，也就是说，没有“搜索头许可证”。但是，**您必须有 Enterprise 许可证来配置搜索头**。Splunk 建议您将搜索头添加到 Enterprise 许可证池，即使您不需要索引任何数据。阅读“[组、堆叠、池和其他术语](#)”以及“[创建或编辑许可证池](#)”。

**注意：**如果您的现有搜索头已安装 4.2 之前版本的转发器许可证，则升级后不读取该转发器许可证。

## 索引器群集节点的许可证（用于索引复制）

如同任意 Splunk 部署，您的许可要求由索引器处理的数据量来驱动。请与您的 Splunk 销售代表联系，购买更多的许可量。

只有几个特定于索引复制的许可证问题：

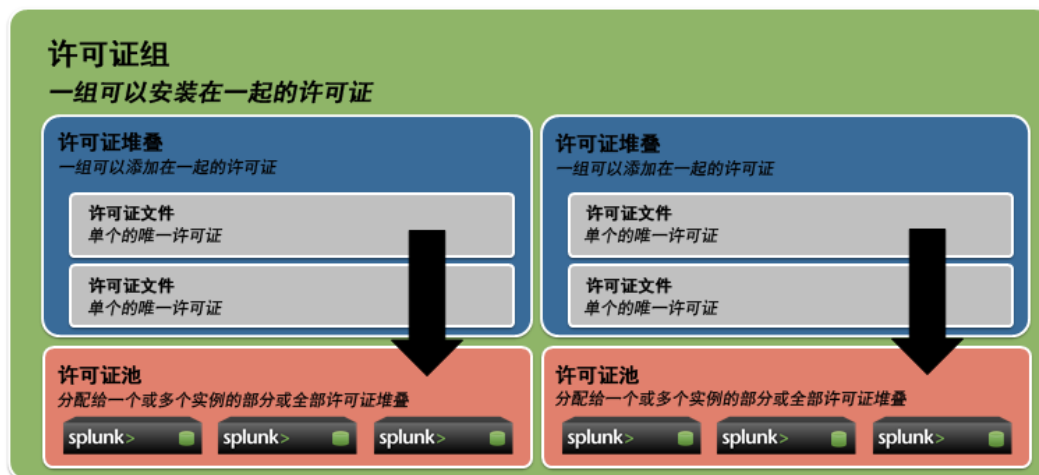
- 所有群集节点（包括主节点、对等节点和搜索头）都需要在 Enterprise 许可证池中，即使不需要它们对任何数据建立索引。
- 群集节点必须共享相同的许可授权配置。
- 只有传入数据用于计算许可证的数据量，复制的数据不计算在内。
- 使用 Free 许可证时不能使用索引复制。

阅读《[管理索引器和群集手册](#)》中有关“系统要求和其他部署注意事项”的更多信息。

## 组、堆叠、池和其他术语

您可以将兼容 Splunk Enterprise 许可证聚集成可用许可量的堆叠，然后定义使用从给定堆叠许可量的索引器池。

**Splunk Free 用户：**此功能仅与 Enterprise 许可证相关。如果您运行 Splunk Free 的独立实例，则不需要组、池和堆叠。



## 堆叠

可以将某些 Splunk 许可证类型聚集在一起或堆叠在一起，使其可用许可量是单个许可证的许可量的总和。

这意味着，随着时间的推移，您可以在需要时增加您的索引量容量，而不必换掉许可证。相反，您只需购买更多的容量，然后将附加容量添加到适当的堆叠。

- [Enterprise 许可证](#)和 [Sales Trial 许可证](#)可以堆叠在一起，并且可以互相堆叠。
- 标准 Splunk Enterprise 下载软件包附带的 Enterprise **trial** 许可证不得包含在堆叠中。Enterprise Trial 许可证专门供独立使用，它自成一组。除非您安装了 Enterprise 或 Sales Trial 许可证，否则您将无

法创建堆叠或定义池以供其他索引器使用。

- [Splunk Free 许可证](#)不能与其他许可证堆叠，包括 Splunk Free 许可证。
- [转发器许可证](#)不能与其他许可证堆叠，包括转发器许可证。
- 测试/开发许可证不能与其他许可证堆叠，包括 Enterprise 许可证。如果您同时安装 Enterprise 许可证和测试/开发许可证，Enterprise 许可证会被删除。

## 组

**许可证组**包含一个或多个堆叠。堆叠只能是一个组的成员，而且每个 Splunk 安装中只能有一个组处于“活动”状态。特别是，这意味着，给定许可证主服务器每次只能管理一个组类型的许可证池。这些组为：

- Enterprise/Sales Trial 组 -- 此组允许堆叠购买的 Enterprise 许可证和 Sales Trial 许可证（此类许可证是具有设定到期日期的 Enterprise 许可证，与下载的 Enterprise Trial 不同）。
- Enterprise Trial 组 -- 此组是首次安装新 Splunk 平台实例时的默认组。您不能将多个 Enterprise Trial 许可证组合成堆叠，然后从该堆叠创建池。**如果您切换到其他组，则不能再切换回 Enterprise Trial 组。**
- Free 组 -- 此组用于 Splunk Free 的安装。当 Enterprise Trial 许可证在 60 天后到期时，该 Splunk 实例转换为 Free 组。您不能将多个 Splunk Free 许可证组合成堆叠，然后从该堆叠创建池。
- Forwarder 组 -- 此组用于将 Splunk 实例配置为通用转发器或轻型转发器。这些转发器类型不执行任何索引建立，因此实际上并不通过管理器中的许可页面进行管理，但确实属于许可证组。如果您将某一 Splunk 实例的许可证组更改为 Forwarder 组，则假设该 Splunk 实例配置为转发器，不对任何数据建立索引。有关更多信息，请参阅[转发器](#)和[“转发器许可证”](#)。

## 子组

子组可以拥有其中一个值，包括开发测试或生产。您不能用两个不同的子组堆叠许可证。

Splunk Enterprise 6.5.0 对子组有所介绍。没有子组的许可证，如在 Splunk Enterprise 6.5.0 之前发布的许可证，可将子组视为“生产”。

## 池

您可以定义许可量的池（其许可量取自给定许可证的[许可证堆叠](#)），同时指定其他索引 Splunk 实例作为该池的成员，用于使用和跟踪许可量。

许可证池由 Splunk 的一个[许可证主服务器](#)和零个或多个[许可证从服务器](#)实例组成，这些实例配置为使用已设置许可证或[许可证堆叠](#)的许可量。

## 许可证从服务器

许可证从服务器是一个或多个许可证池的成员。许可证从服务器对许可量的访问权限由许可证主服务器控制。

## 许可证主服务器

一个许可证主服务器控制一个或多个许可证从服务器。您可以从许可证主服务器上定义池、添加许可容量和管理许可证从服务器。

# 安装许可证

本主题介绍如何安装新的许可证。您可以在 Splunk 平台[许可证主服务器](#)上安装多份许可证。**注意：**如果您用 Enterprise 许可证安装开发/测试许可证，Enterprise 许可证文件将会被覆盖。

在您继续之前，您可能需要阅读这些主题：

- 阅读 [Splunk 许可授权如何工作](#)以获得有关 Splunk 许可授权的简介。
- 阅读 [Splunk 软件许可证类型](#)以对比许可证类型，并了解哪些许可证可以合并，哪些不可以。
- 阅读[组、堆叠、池和其他术语](#)以获得有关 Splunk 许可证术语的更多信息。

## 添加新许可证

要添加新许可证：

1. 导航到设置 > 许可证。
2. 单击添加许可证。



3. 单击**选择文件**并浏览您的许可证文件，然后选择该文件，或者单击**直接复制和粘贴许可证 XML...**并将许可证文件的文本粘贴到所提供的字段中。

4. 单击**安装**。如果这是您安装的第一份 Enterprise 许可证，则必须重新启动 Splunk Enterprise。您的许可证现已安装就绪。

## 配置许可证主服务器

本主题介绍如何将 Splunk 实例配置为**许可证主服务器**。在您继续之前，您可能需要阅读这些主题：

- 阅读本手册中的“[Splunk 许可授权如何工作](#)”以获得有关 Splunk 许可授权的简介。
- 阅读本手册中的“[组、堆叠、池和其他术语](#)”以获得有关 Splunk 许可证术语的详细信息。

### 许可证主服务器种类

许可证主服务器有两种基本类型：

- **独立许可证主服务器**
  - 如果您具有单个 Splunk 索引器，并希望管理其许可证，则您可以将其作为许可证主服务器来运行，并在其上面安装一份或多份 Enterprise 许可证，这样它可以将自身作为从许可证服务器来进行管理。
  - 当您首次下载并安装 Splunk Enterprise 时，它含有一份为期 60 天的 500 MB Enterprise Trial 许可证。该实例会自动配置为独立许可证主服务器，您无法针对此类许可证创建池或定义任何**许可证从服务器**。如果您想要创建**一个或多个堆叠或池**，然后为其分配多个索引器，则必须购买并安装 [Enterprise 许可证](#)。要安装许可证，请遵循本手册中“[安装许可证](#)”的说明。
- **中央许可证主服务器**
  - 如果您有多个索引器，并想要从中央位置管理对所购买的许可容量的访问权限，则可配置中央许可证主服务器，然后将这些索引器添加为**许可证从服务器**。
  - 如果许可证主服务器也是索引器，则该服务器也是自己的许可证主服务器，但 Splunk 建议，如果您有**搜索头**，可将其指派为许可证主服务器。
  - 如果您的环境规模很大，其中配有多个搜索头，则您可能想要一些或所有不是许可证主服务器的搜索头将搜索分布到许可证主服务器，有以下两个原因：
    - 您可以对照许可证日志运行搜索。
    - 如果在搜索头上出现不寻常的情况（例如，您的许可证有时间限制，将在 5 天后到期），则在运行搜索时可在搜索头上看到这一情况，作为信息消息的一部分附加到搜索结果。

### 许可证主服务器和从服务器兼容性

许可证主服务器必须和从服务器用相同的版本或用更高的版本。

许可证主服务器版本	可兼容的许可证从服务器版本
6.1.x	5.X、6.0.x 和 6.1.x
6.2.x	5.x、6.0.x、6.1.x、6.2.x
6.3.x	5.x、6.0.x、6.1.x、6.2.x、6.3.x
6.4.x	5.x、6.0.x、6.1.x、6.2.x、6.3.x、6.4.x

### 配置中央许可证主服务器

默认情况下，Splunk 的独立实例是自己的许可证主服务器。要配置中央许可证主服务器，[安装一个或多个 Enterprise 许可证](#)。

安装 Enterprise 许可证之后，您可以[创建一个或多个堆叠和池](#)，用于访问安装的许可证，然后从许可证主服务器[管理许可证](#)。



## 配置许可证从服务器

本主题介绍如何将 Splunk 索引器配置为**许可证从服务器**。在您继续之前，您可能需要阅读这些主题：

- 阅读本手册中的“[Splunk 许可授权如何工作](#)”以获得有关 Splunk 许可授权的简介。
- 阅读本手册中的“[组、堆叠、池和其他术语](#)”以获得有关 Splunk 许可证术语的详细信息。
- 有关设置许可证主服务器的说明，请参阅本手册中的“[配置许可证主服务器](#)”。
- 有关从命令行执行这些任务的帮助，请参阅本手册中的“[从 CLI 管理许可证](#)”。

1. 在想要配置为许可证从服务器的索引器上，登录 Splunk Web，然后导航到**设置 > 许可授权**。

2. 单击**更改为从服务器**。



3. 将单选按钮从**将此 Splunk 实例 <此索引器> 指定为许可证主服务器**切换到**将其他 Splunk 实例指定为许可证主服务器**。

4. 指定此许可证从服务器应报告的许可证主服务器。您必须提供 IP 地址或主机名和 **Splunk 管理端口**（默认为 8089）。

**注意：**可以按 IPv4 或 IPv6 格式指定 IP 地址。有关 IPv6 支持的详细信息，请参阅本手册中的“[配置 Splunk 以使用 IPv6](#)”。

5. 单击**保存**。如果此实例还未安装 Enterprise 许可证，则必须重新启动 Splunk。此索引器现在已配置为许可证从服务器。

要切换回去，导航到**设置 > 许可授权**，然后单击**切换到本地主服务器**。如果此实例还未安装 Enterprise 许可证，要使此更改生效，必须重新启动 Splunk。

## 创建或编辑许可证池

本主题介绍如何从一个或多个已安装许可证创建许可证池，以及如何编辑现有许可证池。在您继续之前，您可能需要阅读这些主题：

- 阅读本手册中的“[Splunk 许可授权如何工作](#)”以获得有关 Splunk 许可授权的简介。
- 阅读本手册中的“[组、堆叠、池和其他术语](#)”以获得有关 Splunk 许可证术语的详细信息。
- 阅读“[安装许可证](#)”，了解有关安装许可证的更多信息。
- 有关从命令行执行这些任务的帮助，请参阅本手册中的“[从 CLI 管理许可证](#)”。

当您首次下载并安装 Splunk 时，它含有一份为期 60 天的 500 MB Enterprise Trial 许可证。该 Splunk 实例会自动配置为**独立许可证主服务器**，您无法针对此类许可证创建池或定义任何**许可证从服务器**。如果您想要创建**一个或多个堆叠或池**，然后为其分配多个索引器，则必须购买并安装 [Enterprise 许可证](#)。

在**设置 > 许可授权**的下例中，全新的 Splunk 安装上刚刚安装了 100 MB Enterprise 许可证：

## 授权

此服务器充当独立许可证服务器
 [更改为从服务器](#)

### Trial 许可证组

[更改许可证组](#)

此服务器配置为使用 Trial 许可证组中的许可证

[添加许可证](#)
[使用情况报表](#)

#### 告警

许可告警通知您索引警告过多以及许可配置不正确。 [了解更多信息](#)

当前的

- 无许可告警

永久的

- 无许可违规

### Splunk Enterprise 堆叠

[了解更多信息](#)

• `auto_generated_pool_enterprise` 当前是默认许可证池。可以通过将服务器索引器指向此计算机上的 `splunkd` 端口来将它们自动添加到该池中。

许可证	数量	过期时间	状态
Splunk Enterprise	100 MB	Jan 18, 2038 7:14:07 PM	有效
每日有效数量	100 MB		

池	索引器	今天使用的数量	
auto_generated_pool_enterprise *		0 MB / 100 MB	<a href="#">编辑</a>   <a href="#">删除</a>

今天没有索引器报告到该池

[+ 添加池](#)

当您在全新的 Splunk 服务器上安装 Enterprise 许可证时，Splunk 会自动从中创建 Enterprise 许可证堆叠（也称为“Splunk Enterprise 堆叠”），并为其定义默认的许可证池（称为 `auto_generated_pool_enterprise`）。

该默认池的默认配置会将任何连接到此许可证主服务器的许可证从服务器添加到池中。您可以编辑该池以更改此配置，为其添加更多的索引器，或从此堆叠创建新的许可证池。

## 要编辑现有的许可证池

1. 在要编辑的许可证池的旁边，单击 **编辑**。这将显示“编辑许可证池”页面。
2. 根据需要，更改分配或改变[如何允许索引器访问此池](#)。您还可以更改池的描述，但不能更改池的名称。
3. 单击 **提交**。

## 要创建新许可证池

**重要提示：**要想能够从默认 Enterprise 堆叠创建新许可证池，必须使一些索引量可用，您可以编辑 `auto_generated_pool_enterprise` 池并减少其分配，也可以整个删除池。单击池名称旁的 **删除** 来删除池。

1. 单击页面底部的 [+ 添加池](#)。这将显示“创建新许可证池”页面。
2. 指定池的名称，也可以指定池的描述。
3. 设置该池的分配。分配是指整个堆叠的许可量中可供属于该池的索引器使用的许可量。分配可以是特定值，也可以是堆叠中的整个可用索引量（只要没有分配给任何其他池）。
4. 指定索引器如何访问该池。这些选项为：
  - 环境中所有已配置为许可证从服务器的索引器都能连接到此许可证池并使用其中的许可证分配。
  - 只有您指定的索引器才能连接到此池并使用其中的许可证分配。
5. 要允许从池绘制特定的索引器，请单击“可用”索引器列表中索引器名称旁边的加号，以将其移动到“关联的”索引

66

器列表中。

## 向许可证池添加索引器

本主题介绍如何向现有许可证池添加索引器。在您继续之前，您可能需要阅读这些主题：

- 阅读本手册中的“[Splunk 许可授权如何工作](#)”以获得有关 Splunk 许可授权的简介。
- 阅读本手册中的“[组、堆叠、池和其他术语](#)”以获得有关 Splunk 许可证术语的详细信息。

### 索引器如何访问许可证池

对许可证池的堆叠的访问由该池的许可证主服务器来控制。通过指定许可证主服务器的 URI 和管理端口，可以将池配置为仅允许访问特定索引器，也可以将池配置为允许访问与其连接的所有索引器。

### 添加特定索引器

按照以下两个基本步骤将特定索引器访问权限授予给定许可证池的堆叠：

1. 将索引器配置为许可证从服务器，并为其提供许可证主服务器的 URI 和管理端口。为此，应遵循本手册的“[配置许可证从服务器](#)”中的说明。
2. 在许可证管理器上配置池，以接受来自该索引器的访问。为此，遵循“[创建或编辑许可证池](#)”中的说明以编辑许可证池，选择单选按钮选项以仅允许访问特定索引器，然后在“可用”索引器列表中单击索引器名称旁边的加号，以便将其移动到“关联的”索引器列表中。

### 添加任何连接的索引器

遵循以下步骤，使所有连接到此许可证主服务器的索引器均有权访问给定许可证池的堆叠：

1. 将索引器配置为许可证从服务器，并为其提供许可证主服务器的 URI 和管理端口。为此，应遵循本手册的“[配置许可证从服务器](#)”中的说明。
2. 在许可证主服务器上配置池，以接受来自任何索引器的访问。为此，遵循“[创建或编辑许可证池](#)”中的说明以编辑许可证池，然后选择单选按钮选项以允许从任何连接的索引器进行访问。

## 从 CLI 管理许可证

本主题介绍如何使用 Splunk CLI 监视和管理 Splunk 许可证。在您继续之前，请阅读这些主题：

- 阅读本手册中的“[Splunk 许可授权如何工作](#)”以获得有关 Splunk 许可授权的简介。
- 阅读本手册中的“[组、堆叠、池和其他术语](#)”以获得有关 Splunk 许可证术语的详细信息。

本主题仅涵盖与 Splunk 的许可证相关对象交互时可以使用的 CLI 命令。其中一些命令还有您可以为每个对象指定的必填参数和可选参数。有关完整的语法和用法示例，请参阅 Splunk 的 CLI 帮助。

- 有关使用 Splunk 命令行界面的简介，请阅读本手册中的“[关于 CLI](#)”。

有关通过 Splunk 的 REST API 管理许可证的信息，请参阅《REST API 参考手册》中的“许可证”。

### CLI 许可证命令和对象

使用 Splunk CLI，您可以添加、编辑、列出和删除许可证和许可证相关对象。可用命令为：

命令	对象	描述
add	licenses, licenser-pools	将许可证或许可证池添加到许可证堆叠。仅当您拥有 Enterprise 许可证时，此命令才可用。
edit	licenser-localslave, licenser-pools	编辑许可证堆叠中本地许可证从服务器节点或许可证池的属性。仅当您拥有 Enterprise 许可证时，此命令才可用。
list	licenser-groups, licenser-localslave, licenser-messages, licenser-pools, licenser-slaves, licenser-stacks, licenses	根据指定的许可证相关对象，列出该对象或该对象的成员的属性。
remove	licenser-pools, licenses	从许可证堆叠中删除许可证或许可证池。

许可证相关对象为：

对象	描述
----	----

licenser-groups	您可以切换到的不同许可证组。
licenser-localslave	本地索引器的配置。
licenser-messages	关于您的许可证状态的告警或警告。
licenser-pools	池或虚拟许可证。一个堆叠可以分割成各种池，多个从服务器共享每个池的配额。
licenser-slaves	已联系到主服务器的所有从服务器。
licenser-stacks	此对象代表许可证堆叠。堆叠包含相同类型的许可证，可累加。
licenses	此 Splunk 实例的所有许可证。

## 常用许可证相关任务

下面给出常用许可证相关任务的示例。

### 管理许可证

要将新许可证添加到许可证堆叠，指定许可证文件的路径：

```
./splunk add licenses /opt/splunk/etc/licenses/enterprise/enterprise.lic
```

要列出许可证堆叠中的所有许可证：

```
./splunk list licenses
```

List 还显示每个许可证的属性，包括它启用的功能 (features)、它属于的许可证组和堆叠 (group\_id, stack\_id)、它允许的索引配额 (quota) 以及对每个许可证均唯一的许可证密钥 (license\_hash)。

如果许可证期满，您还可以从许可证堆叠中删除它。要从许可证堆叠中删除许可证，指定许可证的哈希值：

```
./splunk remove licenses EM+S8VetLnQEblF+5Gwx9rR4M4Y91AkIE=781882C56833F36D
```

### 管理许可证池

您可以从许可证堆叠中的一个或多个许可证创建许可证池（如果您有 [Enterprise 许可证](#)）。基本上，一个许可证堆叠可以划分为多个许可证池。每个池可以拥有多个从许可证服务器，它们共享该池的配额。

要查看所有许可证堆叠中的所有许可证池：

```
./splunk list licensor-pools
```

要将某个许可证池添加到堆叠，您需要：命名该池，指定您要添加到的堆叠，并指定要分配到该池的索引量：

```
./splunk add licensor-pools pool01 -quota 10mb -slaves guid1,guid2 -stack_id enterprise
```

您还可以指定该池和作为其成员的许可证从服务器的描述（均为可选）。

您可以编辑许可证池的描述、索引配额和许可证从服务器：

```
./splunk edit licensor-pools pool01 -description "Test" -quota 15mb -slaves guid3,guid4 -append_slaves true
```

这主要是添加对池的描述（“Test”），将配额从 10MB 更改为 15MB，向该池添加许可证从服务器 guid3 和 guid4 池（而不是覆盖或取代 guid1 和 guid2）。

要从堆叠中删除许可证池，应指定名称：

```
./splunk remove licensor-pools pool01
```

### 管理许可证从服务器

许可证从服务器是一个或多个许可证池的成员。许可证从服务器对许可量的访问权限由许可证主服务器控制。

要列出所有联系过许可证主服务器的许可证从服务器：

```
./splunk list licensor-slaves
```

要列出本地从许可证服务器的所有属性：

```
./splunk list licenser-localslave
```

要添加许可证从服务器，应编辑该本地许可证从服务器节点的属性（指定 splunkd 许可证主服务器实例的 URI 或 'self'）：

```
./splunk edit licenser-localslave -master_uri 'https://master:port'
```

### 监视许可证状态

您可以使用 list 命令来查看有关许可证状态的消息（告警或警告）。

```
./splunk list licenser-messages
```

## 管理 Splunk 许可证

### 管理许可证

本主题介绍如何管理 Splunk Enterprise 许可证。在您继续之前，您可能需要阅读这些主题：

- 阅读本手册中的 [Splunk 许可授权如何工作](#)。
- 阅读本手册中的[组、堆叠、池和其他术语](#)。
- 有关从命令行执行其中一些任务的帮助，请参阅本手册中的[从 CLI 管理许可证](#)。

有关升级现有许可证的信息，请参阅《[安装手册](#)》中的“迁移到新的 Splunk 许可证”。

### 删除许可证

如果某一许可证已到期，您可以删除它。要删除一个或多个许可证：

1. 在许可证主服务器上，导航到**系统 > 许可授权**。

许可证	数量	过期时间	状态	
Dev_Splunk_Enterprise	1 MB	2011-5-24上午 3:30:06	有效	<a href="#">删除</a>
Dev_Splunk_Enterprise	5 MB	2011-5-24上午 5:54:50	有效	<a href="#">删除</a>
每日有效数量	6 MB			

2. 单击您要删除的许可证旁边的**删除**。

3. 再次单击**删除**确认。

**注意：**您不能删除许可证主服务器上的许可证列表中的最后一个许可证。

### 查看许可证使用情况

您可以通过“许可证使用情况报告视图”监视您的部署中的许可证使用情况。访问**系统 > 许可授权 > 使用情况报表**中的视图。请阅读下一章中的[许可证使用情况报告视图](#)以获得更多信息。

### 关于许可证违规

本主题介绍许可证违规问题以及这些问题如何发生和如何解决。在您继续之前，您可能需要阅读这些主题：

- 阅读 [Splunk 软件许可证类型](#)以了解关于新的非限制许可证的信息。
- 阅读 [Splunk Enterprise 许可授权如何工作](#)以获得有关 Splunk Enterprise 许可授权的简介。

### 许可证违规和警告是什么？

当您超过您的许可证所允许的最大索引量时，将发生警告和违规。

如果您在任意一个日历日超过您的日许可量，您将收到**违规警告**。如果您在过去的 30 天内，强制 Enterprise 许可证收到 5 次或更多警告，Free 许可证收到 3 次警告，您的许可证即出现**违规问题**。除非您正在使用 Splunk Enterprise 6.5.0 或更新的非强制许可证，否则搜索会被禁用以避免攻击**池**。只要所有池的许可证使用总量不超过许可证主服务器的许可证配额总数，其他池便仍可搜索。

如果您在先前的 30 天内收到的警告次数小于 5 次 (Enterprise) 或 3 次 (Free)，或者如果您应用临时重置许可证

（仅可用于 Enterprise），则恢复搜索功能。要获得重置许可证，请与您的销售代表联系。请参阅[安装许可证](#)。

从 Splunk Enterprise 6.5.0 开始，Enterprise 客户可以要求使用非强制许可证。您超过许可证配额或许可证违规时，许可证发出告警，但不会禁用搜索。即便在违规期间，搜索保留启用功能。有关详细信息，请参阅[Splunk 软件许可证类型](#)。

**注意：**即使发生许可证违规，**摘要索引量**也不用于计算许可证，因为摘要索引如任何其他非内部搜索行为一样停止下来。内部索引（如 `_internal` 和 `_introspection`）不用于计算许可证索引量。

如果您收到许可证警告，直到午夜前（按许可证主服务器上的时间计算），您都可解决此问题，否则警告将计入过去 30 天内的警告总数中。

在许可证违规期间：

- **Splunk 软件不停止对数据建立索引。**
- 如果您正在使用 6.5.0 之前的许可证，则违规时 Splunk 软件会阻止搜索。
- 如果您正在使用新的非强制许可证，即便您许可证违规，搜索仍旧继续。
- 禁用对 `_internal` 索引进行的搜索。这意味着，您可访问“监视控制台”或针对 `_internal` 运行搜索以诊断许可证问题。

## 许可证警告的结构

如果某一池中的索引器超过分配给该池的许可量，则会在 Splunk Web 上任何一页看到消息里的消息。

单击消息里的链接可转到**设置 > 许可授权**，相应警告显示在该页面的**告警**部分下。单击警告以获得有关该警告的更多信息。

发生违规行为时，在许可证从服务器上也显示类似的消息。

以下列出生成许可告警的一些条件：

- 当某一从服务器孤立存在时，将显示告警（短暂显示，可在午夜之前修复）。
- 当某一池已用尽时，将显示告警（短暂显示，可在午夜之前修复）。
- 当某一堆叠已用尽时，将显示告警（短暂显示，可在午夜之前修复）。
- 警告发送给一个或多个从服务器时，显示告警。只要在之前 30 天之内警告有效，告警仍然显示。

## 关于许可证主服务器和许可证从服务器之间的连接

在配置了许可证主服务器实例，并为其添加了许可证从服务器之后，许可证从服务器每分钟会向许可证主服务器告知一次其使用情况。如果许可证主服务器发生故障或由于某种原因无法连接到许可证主服务器，许可证从服务器会启动 72 小时计时器。如果许可证从服务器连续 72 小时无法与许可证主服务器通信，则许可证从服务器上的搜索操作将被阻止（仍然继续执行索引操作）。用户将无法搜索许可证从服务器上的索引数据，除非可以再次连接到许可证主服务器。

要了解是否有许可证从服务器不能连接许可证主服务器，可在 `splunkd.log` 中查找包含 `failed to transfer rows` 的事件或在 `_internal` 索引中搜索。

## 如何避免许可证违规

要避免许可证违规，监视您的许可证使用情况，同时确保您有足够的许可量来支持。如果您没有足够的许可量，则需要增加许可证或减少索引量。

分布式管理控制台包含您可以启用的告警，包括监视许可证使用情况的告警。请参阅《*监视 Splunk Enterprise*》中的“平台告警”。

使用**许可证使用情况**报告来查看有关您部署中故障排除索引量的详细信息。请阅读下一章中的[许可证使用情况报表视图](#)以获得相关信息。

## 纠正许可证警告

如果 Splunk 软件要求您在午夜之前纠正许可证警告，那么很可能您已经超出了当天的配额。这称为“软警告”。每日许可证配额在午夜（此时软警告成为“硬警告”）重置。您必须在此之前解决这个问题，并且还要确保明天不会超过此配额。

对数据创建索引之后，便无法取消数据的索引使您的许可证有回旋余地。您需要以下面这些方式之一来获得额外的许可证空间：

- 购买更大的许可证。
- 重新排列许可证池（如果某个池存在多余的许可证空间）。
- 如果您的许可证主服务器正在运行 Splunk Enterprise 6.5.0 或者之后的版本，请求非强制 Enterprise 许可证。

如果您不能采用其中任何一种方式，则通过使用更少的许可证以防止未来出现警告。请参阅[许可证使用情况报表视图](#)以了解哪些数据源对您的配额有最大贡献。



确定产生数据问题的主要原因后，您可以决定是否需要它发出的所有数据。如果不需要，请阅读《转发数据》手册里的“路由和过滤数据”。

## 问答

有什么问题吗？请访问 [Splunk Answers](#) 以查看在 [Splunk 社区](#) 中对于许可证违规有哪些相关的问题和解答。

## 交换许可证主服务器

执行此步骤之前您应已配置许可证池。如果要将许可证从服务器中的一台服务器变成池的许可证主服务器，该如何操作？

本主题将提供相关的操作步骤。总的来说，您先将一台从服务器提升为主服务器。然后将旧的主服务器降级为从服务器。详细信息如下。

1. 从许可授权池中删除新的许可证主服务器，然后将其设置为主服务器。
  - 登录到许可证从服务器（该服务器将成为新的主服务器）。
  - 导航到 **设置 > 许可证**。
  - 按照提示 [将其配置为新的许可证主服务器](#)。
  - 重新启动 Splunk。
2. 在新的许可证主服务器中，[添加许可证密钥](#)。检查许可证密钥是否与旧的许可证主服务器相匹配。
3. 使池中的其他许可证从服务器指向新的许可证主服务器。
  - 在每台从服务器上，导航到 **设置 > 许可授权**。
  - 更改许可证主服务器 URI 以指向新的许可证主服务器，然后单击 **保存**。
  - 在刚刚更新了其条目的许可证从服务器上重新启动 Splunk。
4. 检查其中一台许可证从服务器是否已连接到新的许可证主服务器。
5. 将旧的许可证主服务器降级到从服务器：
  - 在旧的许可证主服务器上导航到 **设置 > 许可授权 > 更改为从服务器**。
  - 忽略重新启动提示。
  - 在“更改为从服务器”屏幕中，通过单击“**指定其他 Splunk Enterprise 实例作为许可证主服务器**”使新的从服务器指向新的许可证主服务器。
6. 在新的许可证从服务器上，停止 Splunk Enterprise 并删除 `/opt/splunk/etc/licenses/enterprise/` 文件夹下的旧许可证文件。（如若不然，许可证将出现重复，系统会发出错误和/或警告信息。）
7. 在新的许可证从服务器上，启动 Splunk Enterprise 并确认它已连接到新的许可证主服务器。

## 许可证使用情况报表视图

### 关于 Splunk Enterprise 的许可证使用情况报表视图

#### 许可证使用情况报表视图介绍

许可证使用情况报表视图 (LURV) 是针对与 Splunk 许可容量和索引量有关的问题的合并资源。您可以直接从 Splunk 许可授权页看到每天的索引量、任何许可证警告和附带有多个报告选项的过去 30 天许可证使用情况的视图。

LURV 可显示许可证池的许可证使用情况详细信息。此仪表板从逻辑上分为两部分：一部分在当前滚动窗口中显示当日的许可证使用情况信息以及任何警告信息；另一部分显示过去 30 天内的历史许可证使用情况。

对于 LURV 中的每个面板，您可以单击面板左下角的“在搜索中打开”与搜索交互。

#### 访问许可证使用情况报表视图

在 **设置 > 许可授权 > 使用情况报表** 中找到 LURV。



在部署的许可证主服务器上访问 LURV。（如果部署中只有一个实例，则此实例就是其自己的许可证主服务器。）

## “今天”选项卡

第一次进入 LURV 时，会在“今天”选项卡下看到五个面板。这些面板显示了尚未结束的当天的许可证使用情况状态以及警告。无论许可证主服务器设置在哪个时区，对于许可证，每天的结束时间都是午夜。

“今天”选项卡中的所有面板都查询 Splunk REST API。

### 今天的许可证使用情况面板

此面板可评估当天的许可证使用情况以及跨所有池的每日许可证总配额。

### 每个池面板的今天许可证使用情况

此面板可显示每个池的许可证使用情况以及每个池的每日许可证配额。

### 今天每个池面板使用的每日许可证配额的百分比

此面板可显示已由每个池编入索引的每日许可证配额的百分比。百分比以对数标度进行显示。

### 池使用情况警告面板

此面板可显示每个池在过去 30 天中（或自应用了最后一个许可证重设密钥以来）所收到的软警告和硬警告。请阅读本手册中的[关于许可证违规](#)，以了解有关软警告和硬警告以及许可证违规的详细信息。

### 从服务器使用情况警告面板

对于每台许可证从服务器，此面板显示：警告数目、池成员资格以及从服务器是否违规。

## “前 30 天”选项卡

单击“前 30 天”选项卡即可看到五个以上面板和几个下拉选项。

这些面板中的所有可视化元素限制了显示出来的主机、来源、来源类型、索引、池（拆分所依据的任何字段）的数量。如果其中任何一个字段有 10 个以上不同值，第 10 个之后的值将被标记为“其他”。我们已使用 `timechart` 将所显示出来的值的最大数量设置为 10。希望这在大多数时候能给您提供足够多的信息，不会令可视化元素难以读懂。

这些面板所使用的数据都是使用 `license_usage.log, type=RolloverSummary`（每日总计）集合来的。如果您的许可证主服务器中的时间达到本地午夜，它不会为当天生成 `RolloverSummary` 事件，您不会在这些面板中看到当天的数据。

### 拆分依据：无拆分依据、索引器、池

这三个拆分依据选项不需另行说明。请阅读本手册前面章节中的[向许可证池添加索引器和许可证池](#)。

### 拆分依据：来源、来源类型、主机、索引

关于这四个拆分依据字段有两件事您应该了解：报表加速和压缩。

#### 报表加速

按数据来源、来源类型和主机拆分使用 `license_usage.log type=Usage`，可以每隔一分钟提供一次实时使用情况统计数据。我们建议在您的许可证主服务器上加速支配这些拆分依据选项的报表。（如果没有加速，搜索会相当慢，因为它在 30 天的数据（以每分钟一个事件的速率生成）中搜索 -- 事件数量很大！）



默认情况下对于此报表，加速处于禁用状态。要加速报表，请单击在选择其中一个拆分依据值时显示在信息消息中的链接。也可以在**设置 > 搜索和报表 > 许可证使用情况数据立方体**中找到加速工作流。请阅读《报表手册》中的“加速报表”。

请注意，在首次选择报表加速后，它可能会花费长达 10 分钟的时间来开始。然后 Splunk 将花费一些时间来构建加速摘要 -- 一般为几分钟到几十分钟，具体取决于编入摘要的数据量。只有在加速完成构建后，对于这些拆分依据选项才会有更快速的性能。

但在第一次运行加速后，后续报表将依据已有内容构建，以将报表保持在最新状态（因而报表的速度会加快）。您应只在第一次启用报表加速时会等待较长时间。

**重要提示：**请仅在许可证主服务器上启用报表加速。

使用 `auto_summarize` 在 `savedsearches.conf` 中配置加速的运行频率。默认每隔 10 分钟运行一次。请频繁运行，以保持工作负荷小而稳定。我们在 3 分钟处每 10 分钟设置一个 cron。可以在 `auto_summarize.cron_schedule` 中配置此设置。

## 压缩

每个索引器会定期向许可证管理器报告已编入索引的数据状态：按来源、来源类型、主机和索引划分。如果不同（来源、来源类型、主机、索引）元组的数量超过 `squash_threshold`，Splunk 会压缩 `{host, source}` 值，仅报告按 `{sourcetype, index}` 的划分。这可防止内存和 `license_usage.log` 行快速增长。

由于对其他字段的压缩，仅按来源类型和索引的拆分可确保获得完整的报表（每字节）。如果这两个字段有许多不同值，则对于按来源和主机的拆分，不保证一定能获得完整的报表。Splunk 会报告编入索引的完整数量，而不是名称。所以您会失去粒度（也就是不知道是谁消耗了这个数量），但仍知道所消耗的数量为多少。

可以在 `server.conf` 的 `[license]` 段落中使用 `squash_threshold` 设置配置压缩（但要慎重）。可以增加此值，但这样做可能会使用大量内存，所以请在更改前咨询 Splunk 支持工程师。

如果发生了压缩，LURV 始终会通知您（在 UI 中使用警告消息）。

如果您发现您需要粒度信息，则可从 `metrics.log` 使用 `per_host_thruput` 获得该信息。

## 平均每日数据量的前 5 个值

对于按已从“拆分依据”菜单中选择的字段拆分的任何项目，“前 5 个”面板可显示其平均和最大每日使用量的前五个值。

请注意，它选择的是前五个平均值（而不是峰值）。因此，例如，假设您有不止五个来源类型。来源类型 F 通常比其他值小很多但是有一个简短峰值。来源类型 F 的**最大**每日使用量很高，但是它的**平均**使用量可能仍然很低（因为它所有天的使用量都很低，这降低了它的平均使用量）。由于此面板选择前五个**平均**值，来源类型 F 可能仍不会显示在此视图中。

## 使用 LURV

请阅读下一个主题中有关[基于 LURV 面板配置告警](#)的提示。

## 使用“许可证使用情况报表视图”

本主题与使用许可证使用情况报表视图 (LURV) 有关。要了解此视图，请阅读前面的主题[“关于 Splunk 的许可证使用情况报表视图”](#)。

## 设置告警

您可以将任何 LURV 面板变成一个告警。例如，假如您要针对许可证使用情况达到配额的 80% 设置一个告警。

从**今天使用的每日许可证配额的百分比**面板开始。单击面板左下角的“在搜索中打开”。附加

```
| where '% used' > 80
```

然后选择**另存为 > 告警**，并遵循告警向导执行操作。

Splunk Enterprise 附带一些您可以启用的预配置告警。请参阅《[分布式管理控制台手册](#)》中的“平台告警”。

## LURV 故障排除：30 天面板上没有结果

许可证使用情况报表视图的“最近 30 天”视图面板上缺少结果表示搜索时在其上查看页面的许可证主服务器实例无法从其自身 `$SPLUNK_HOME/var/log/splunk/license_usage.log` 文件中查找到的事件。

这通常由以下原因之一引起：

- **许可证主服务器**配置为将事件转发到索引器（请阅读《[分布式搜索手册](#)》中有关此最佳做法的更多信息），但

未配置为搜索头。通过将所有索引器添加到被许可证主服务器转发事件作为搜索节点的部分中，这很容易解决。

- 许可证主服务器不从它自己的 `$SPLUNK_HOME/var/log/splunk` 目录中读取事件（因此也不为其建立索引）。如果 `[monitor://$SPLUNK_HOME/var/log/splunk]` 默认数据输入由于某些原因被禁用，便会出现此情况。

如果您的许可证主服务器午夜时关闭，则您的数据可能会出现间隙。

## 管理应用键值存储

### 有关应用键值存储

应用键值存储（或 KV 存储）提供在 Splunk 应用内保存和检索数据的方法，从而允许您管理和维护应用程序的状态。

以下是 Splunk 应用可能使用 KV 存储的一些方法：

- 跟踪事件查看系统（将问题从一个用户移动到另一个用户）中的工作流。
- 保留用户提供的环境资产列表。
- 控制任务队列。
- 当用户与应用进行交互时，通过存储用户或应用程序状态管理 UI 会话。
- 存储用户元数据。
- 通过 Splunk 或外部数据存储从搜索查询缓存结果。
- 为模块化输入存储检查点数据。

有关使用 KV 存储的信息，请参阅 Splunk 应用开发者的应用键值存储文档。

### KV 存储如何使用您的部署

KV 存储将数据存储为集合中的键值对。主要概念如下：

- **集合**是数据的容器，与数据库表类似。集合存在于给定应用的上下文中。
- **记录**包含数据的每个条目，与数据库表中的行类似。
- **字段**对应键名称，与数据库表中的列类似。字段将数据值包含为 JSON 文件。尽管不需要，您还可强制限定字段值的数据类型（数量、布尔、时间和字符串）。
- **\_key**为包含每个记录的唯一 ID 的预留字段。如果您未显式指定 key 值，则应用会自动生成一个值。
- **\_user**为包含每个记录的用户 ID 的预留字段。此字段不可以被覆盖。
- **加速**通过使包含加速字段的搜索更快的返回来改进搜索性能。加速在易于遍历的表单中存储集合数据的一小部分。

KV 存储驻留在搜索头的文件。

在搜索头群集中，如果任何节点收到写入，则 KV 存储会将写入委派到 **KV 存储管理员**。但是，KV 存储保持本地读取。

### 系统要求

KV 存储在所有 Splunk Enterprise 64 位构建中可用并被支持。它在 32 位 Splunk Enterprise 构建中不可用。KV 存储在通用转发器中也不可用。请参阅 Splunk Enterprise 系统要求。

默认情况下，KV 存储使用端口 8191。您可以更改 `server.conf` 的 `[kvstore]` 段落中的端口号。有关 Splunk Enterprise 使用的其他端口的信息，请参阅《分布式搜索手册》中的“搜索头群集的系统要求和其他部署注意事项”。

有关您可以在 KV 存储中更改的其他配置的信息，请参阅 [server.conf.spec](#) 中的“KV 存储配置”部分。

### 关于 Splunk FIPS

要通过 KV 存储使用 FIPS，请参阅 [server.conf.spec](#) 中的“KV 存储配置”部分。

如果未启用 Splunk FIPS，这些设置将被忽略。

如果您启用了 FIPS 但未提供需要的设置（`caCertFile`、`sslKeysPath` 和 `sslKeysPassword`），KV 存储不会运行。在 `splunkd.log` 中和在执行 `splunk start` 的控制台上查找错误消息。

### 决定您的应用是否使用 KV 存储

默认情况下，KV 存储在 Splunk Enterprise 6.2+ 上启用。

使用 KV 存储的应用通常具有定义在 `$SPLUNK_HOME/etc/apps/<app name>/default` 中的 `collections.conf`。另外，`transforms.conf` 将引用具有 `external_type = kvstore` 的集合

## 使用 KV 存储

要使用 KV 存储：

1. 使用配置文件或 REST API 创建集合并选择定义具有数据类型的字段列表。
2. 使用搜索查找命令和 Splunk REST API 执行创建-读取-更新-删除 (CRUD) 操作。
3. 使用 REST API 管理集合。

## 在 Splunk Enterprise 部署上监视其影响

您可通过监视控制台上的两个视图监视 KV 存储性能。一个视图可供查看整个部署（请参阅《*监视 Splunk Enterprise*》中的 KV 存储：部署）。另一个视图提供有关每个搜索头上 KV 存储操作的信息（请参阅《*监视 Splunk Enterprise*》中的 KV 存储：实例）。

## 重新同步 KV 存储

KV 存储成员用所有的写入操作转换数据失败时，KV 存储成员可能是旧的。要解决这个问题，您必须重新同步成员。

### 辨别旧的 KV 存储成员

检查 REST 的自检 `/serverstatus` 端点以识别旧 KV 存储成员。您可以使用 cURL 为端点请求 GET。

```
curl -k -u user:pass https://<host>:<mPort>/services/server/introspection/kvstore/serverstatus
```

如果端点返回自检数据失败，那么成员是旧的。

更多关于 REST API 的信息，请参阅《*REST API 用户手册*》中的“基本概念”。

有关自检端点的更多信息，请阅读《*REST API 参考手册*》中的“自检端点描述”。

### 重新同步旧 KV 存储成员

如果超过一半的成员是旧的，请重新创建群集。请参阅“备份 KV 存储”。

如果旧成员少于一半，请逐个重新同步每个 KV 存储成员。

1. 停止含有旧的 KV 存储成员的搜索头。
2. 运行命令 `splunk clean kvstore --local`。
3. 重新启动搜索头。

## 备份 KV 存储

本主题介绍如何安全地备份和恢复 KV 存储。

### 备份 KV 存储

在执行这些步骤之前，确保熟悉标准的备份和存储工具以及您组织使用的程序。

1. 要备份 KV 存储数据，首先关闭将从中备份 KV 存储的 Splunk 实例。
2. 备份 `server.conf` 文件中 `[kvstore]` 段落的 `dbPath` 参数中指定的路径下的所有文件。
3. 在单一节点上，备份 `$SPLUNK_DB` 路径中找到的 `kvstore` 文件夹。默认情况下，路径为 `/var/lib/splunk/kvstore`。

如果使用搜索头群集，备份任何群集成员上的 KV 存储数据。

### 恢复 KV 存储数据

**注意：**为了成功恢复 KV 存储数据，KV 存储集合 `collections.conf` 必须已经存在于 KV 存储将恢复到的 Splunk 实例上。如果您在恢复 KV 存储数据之后创建集合 `collections.conf`，则 KV 存储数据将会丢失。

要将 KV 存储数据恢复到与备份它的相同的搜索头群集，在每个群集成员上恢复 `kvstore` 文件夹。例如，在有三个成员的搜索头群集上：

1. 从搜索头群集的一个成员上备份 KV 存储数据。
2. 停止每个群集成员。
3. 恢复备份的 KV 存储数据文件夹到每个群集成员上。

4. 启动每个群集成员。

### 将 KV 存储数据恢复到搜索头群集中添加的新成员上

将 KV 存储数据恢复到新成员上，并添加新成员到群集中。例如，在有三个成员的搜索头群集上：

1. 从搜索头群集的一个成员上备份 KV 存储数据。
2. 在您想添加到搜索头群集的搜索头上：
  - a. 添加成员到群集。请参阅《分布式搜索》手册中的“添加群集成员”。
  - b. 停止成员。
  - c. 恢复 KV 存储数据。
  - d. 启动新成员。

### 从原有的搜索头群集恢复 KV 存储数据到新的搜索头群集

**注意：**该程序假定您正在使用新的 Splunk Enterprise 实例创建新的搜索头群集。

1. 从当前（原有）搜索头群集的一个搜索头上备份 KV 存储数据。
2. 要在新的搜索头群集上恢复 KV 存储数据，搜索头群集必须使用一个成员进行初始化，并在启动一个成员之前恢复 KV 存储数据文件夹，然后添加余下的搜索头到搜索头群集环境。本示例使用一个原有的三节点搜索头群集环境和一个新的三节点搜索头群集环境：
  - 从原有搜索头群集的一个搜索头上备份数据。
  - 在搜索头中，将位于新的搜索头群集环境中。
  - 使用您正恢复的 KV 存储数据相同的集合名称创建 KV 存储集合。
  - 通过 `replication_factor=1` 初始化搜索头群集
  - 停止 Splunk 实例并恢复 KV 存储数据。
  - 清理 KV 存储群集。这会从之前的群集上移除群集信息：

```
splunk clean kvstore -cluster
```

- 只需通过一个搜索头启动 Splunk 实例和 bootstrap。
- 在 KV 存储已恢复到搜索头（将位于新的搜索头群集环境中）之后，现在您可以向其中添加其他的新搜索头群集成员。
- 完成之后，转到并更改每个搜索头上的 `replication_factor` 为所需的复制因子数，然后执行滚动重新启动。

## 认识 Splunk 应用

### 应用和加载项

用户经常咨询应用和加载项的定义，以便确定它们彼此之间的区别。没有明确的标准来统一区分应用和加载项。两者都是您安装在 Splunk Enterprise 实例上的配置打包集，并使实例更容易与其他技术或供应商集成，或向它们插入数据。

- **应用**一般提供广泛的用户界面，使您能利用您的数据工作，而且它们经常使用一个或多个加载项以插入不同类型的数据。
- **加载项**一般启用 Splunk Enterprise 或 Splunk 应用以插入或映射特定类型的数据。

对于“管理员”用户，应用和加载项功能两者作为帮助您获得 Splunk Enterprise 中的数据然后有效使用这些数据的工具，它们的区别造成的影响应该很小。对于应用开发人员，区别造成的影响更大：关于开发应用的指南，请参阅 [dev.splunk.com](http://dev.splunk.com)。

### 应用

**应用**是运行在 Splunk Enterprise 上的应用程序。即装即用，Splunk Enterprise 包括一个使您能利用您的数据的默认基本应用：[搜索和报表应用](#)。对于基本应用不能解决的用例，您可以在 Splunk Enterprise 实例上安装很多其他应用，有些是免费的，有些是付费的。示例包括 Splunk App for Microsoft Exchange、Splunk App for Enterprise Security 和 Splunk DB Connect。一个应用可以使用一个或多个加载项，以方便它集合或映射特定类型的数据。

### 加载项

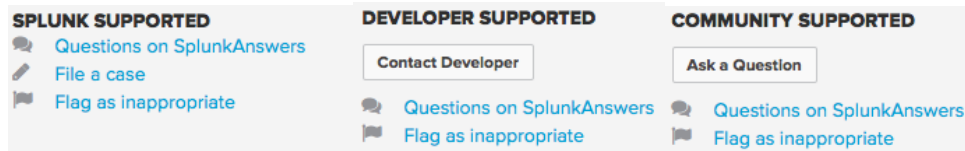
**加载项**运行在 Splunk Enterprise 上为应用提供特定的功能，如导入数据、映射数据或提供**已保存的搜索**和宏。示例包括 Splunk Add-on for Checkpoint OPSEC LEA、Splunk Add-on for Box 和 Splunk Add-on for

McAfee。

## 应用和加载项支持和认证

任何人都可以为 Splunk 软件开发应用或加载项。Splunk 和我们社区的成员创建应用和加载项，并通过 Splunkbase（在线应用市场）与 Splunk 软件的其他用户共享它们。Splunk 不支持 Splunkbase 上的所有应用和加载项。Splunkbase 中的标签指示每个应用或加载项由谁支持。

- Splunk 支持团队仅接受 Splunkbase 上显示 **Splunk 支持** 标签的应用和加载项相关的案例，并回答其相关问题。
- 一些开发人员会支持他们自己的应用和加载项。这些应用和加载项在 Splunkbase 上显示**开发人员支持** 标签。
- Splunk 开发社区支持在 Splunkbase 上显示**社区支持** 标签的应用和加载项。



此外，应用开发人员可以为他们的应用或加载项获得 Splunk 认证。这意味着 Splunk 已经检查了应用或加载项，并发现它符合 Splunk 开发的最佳做法。但是，认证并不意味着 Splunk 支持应用或加载项。例如，发布在 Splunkbase 上并通过 Splunk 认证的由社区开发人员创建的加载项不被 Splunk 支持。在 Splunkbase 上查找 **Splunk 支持** 标签以确定 Splunk 是否支持该应用或加载项。

## 搜索和报表应用

首次安装并登录 Splunk 时，您将进入 Splunk 主页。此主页将显示已为您预先安装的应用中的“单击应用”。



默认情况下，Splunk 提供“搜索和报表”应用。此界面提供了 Splunk 的核心功能，设计旨在用于一般用途。当您第一次登录并提供搜索字段时，此应用显示在主页顶部，您可以立即开始使用它。

进入到“搜索和报表”应用（通过运行搜索或单击主页中的应用）后，可以使用菜单栏选项选择下列项目：

- **搜索**：搜索索引。有关更多信息，请参阅搜索教程中的“使用 Splunk 搜索”。
- **数据透视图**：使用数据模型为数据快速设计并生成表格、图表和可视化元素。有关更多信息，请参阅《数据透视图手册》。
- **报表**：将搜索变成报表。有关更多信息，请参阅搜索教程中的“保存和共享报表”。
- **告警**：为 Splunk 搜索和报表设置告警。有关更多信息，请参阅《告警手册》。
- **仪表板**：利用预定义的仪表板或自己创建。请参阅《仪表板和可视化手册》。

## 配置在应用中打开的 Splunk Web

可以配置 Splunk Web，以使其在所选定应用而不是 Splunk 主页中打开。可以将 Splunk Web 设置为对于所有用户均在特定应用中打开，也可以根据特定用户匹配应用。

### 为单一用户跳过 Splunk 主页

您可以对 Splunk Web 进行配置，这样在用户登录时，可以直接进入您选择的应用，而不是 Splunk 主页。

使“搜索”应用成为用户的默认登录应用：

### 1. 在该用户的本地目录中创建名为 `user-prefs.conf` 的文件：

```
etc/users/<user>/user-prefs/local/user-prefs.conf
```

- 对于 `admin` 用户，该文件应位于：

```
etc/users/admin/user-prefs/local/user-prefs.conf
```

- 对于 `test` 用户，它应位于：

```
etc/users/test/user-prefs/local/user-prefs.conf
```

### 2. 在 `user-prefs.conf` 文件中加入下列行：

```
default_namespace = search
```

## 为所有用户跳过 Splunk 主页

您可以为所有用户指定当它们登录后进入的默认应用。例如，如果要将“搜索”应用作为全局默认应用，则应编辑 `$SPLUNK_HOME/etc/apps/user-prefs/local/user-prefs.conf` 并指定：

```
[general_default]
default_namespace = search
```

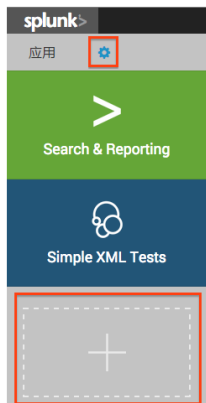
**注意：**如果用户无权访问“搜索”应用，则会显示出错。

## 从哪里获得更多应用和加载项

您可以在 Splunkbase 上找到新的应用和加载项：<https://splunkbase.splunk.com/>。您也可以在 Splunk Enterprise 主页上浏览新应用。

### 如果您已连接到 Internet

如果 Splunk Enterprise 服务器或客户端计算机已连接到 Internet，则可以从主页导航到应用浏览器。



- 您可以单击您上一次安装的应用下方的 + 符号以直接转到应用浏览器。
- 您也可以单击**应用**旁边的齿轮以转到应用管理器页面。单击**浏览更多应用**以转到应用浏览器。

**重要提示：**如果 Splunk Web 位于代理服务器之后，您可能在访问 Splunkbase 时遇到问题。要解决此问题，您需要遵照“指定代理服务器”中的说明来设置 `HTTP_PROXY` 环境变量。

### 如果您未连接到 Internet

如果您的 Splunk Enterprise 服务器和客户端未连接到 Internet，则必须首先从 Splunkbase 下载应用，然后将其复制到您的服务器上：

1. 从具有 Internet 连接的计算机上，浏览 Splunkbase 以找到所需的应用或加载项。
2. 下载应用或加载项。
3. 下载之后，将其复制到您的 Splunk Enterprise 服务器上。
4. 将其放入您的 `$SPLUNK_HOME/etc/apps` 目录中。

5. 通过诸如 `tar -xvf`（在 \*nix 上）或 WinZip（在 Windows 上）等工具来解压您的应用或加载项。请注意，虽然 Splunk 应用和加载项使用 .SPL 扩展名进行打包，但其内部仍为 tar 和 gzip 格式。您可以需要强制您的工具识别此扩展名。

6. 您可能需要重新启动 Splunk Enterprise，这取决于应用或加载项的内容。

7. 现在，您的应用或加载项已安装就绪。如果它具有 Web UI 组件，您还可以从 Splunk 主页上使用它。

## 应用部署概述

本主题提供关于您可用于在常见 Splunk 软件环境中部署 Splunk 应用和加载项的方法概述。

有关应用和加载项部署的更多详细信息，请参阅特定的 Splunk 应用文档，或参阅《Splunk 加载项》手册中的“将 Splunk 加载项安装于何处”。

### 前提条件

您必须具有一个现成的 Splunk 平台部署，在其上安装 Splunk 应用和加载项。

### 部署方法

有几种方法可将应用和加载项部署到 Splunk 平台上。要使用正确的部署方法取决于特定 Splunk 软件部署的以下特性：

- 部署架构（单实例或分布式）
- 群集类型（搜索头群集和/或索引器群集）
- 位置（本地或在 Splunk Cloud 中）

### 部署架构

有两种基本的 Splunk Enterprise 部署架构：

- **单实例部署**：在单实例部署中，一个 Splunk Enterprise 实例既用作搜索头，又用作索引器。
- **分布式部署**：分布式部署会包括多个 Splunk Enterprise 组件，其中包含搜索头、索引器和转发器。请参阅《分布式部署手册》中的“使用 Splunk Enterprise 组件调整部署规模”。分布式部署也包括标准独立组件和/或群集组件，其中包含搜索头群集、索引器群集和多站点群集。请参阅《分布式部署手册》中的“分布式 Splunk Enterprise 概述”。

#### 单实例部署

要在单实例上部署应用，从 **Splunkbase** 下载应用到本地主机，然后使用 **Splunk Web** 安装应用。

一些应用目前不支持通过 Splunk Web 安装。确保在安装之前查看特定应用的安装说明。

#### 分布式部署

您可以使用以下方法在分布式环境中部署应用：

- 使用 Splunk Web 在每个组件上手动安装应用，或通过命令行手动安装应用。
- 使用**部署服务器**安装应用。部署服务器自动分发新的应用、应用更新和某些配置更新到搜索头、索引器和转发器上。请参阅《更新 Splunk Enterprise 实例》中的“关于部署服务器和转发器管理”。

或者，您可以使用第三方配置管理工具来部署应用，例如：

- Chef
- Puppet
- Salt
- Windows 配置工具

大多数情况下，您必须将 Splunk 应用安装在搜索头、索引器和转发器上。要确定您必须将应用安装在哪个 Splunk Enterprise 组件上，请参阅特定应用的安装说明。

### 将应用部署到群集

Splunk 分布式部署包括以下这些群集类型：

- **搜索头群集**
- **索引器群集**

您可以使用**配置软件包**方法将应用部署到索引器和搜索头群集成员上。



## 搜索头群集

要将应用部署到搜索头群集，您必须使用 **Deployer**。Deployer 是将应用和配置更新分发给搜索头群集成员的 Splunk Enterprise 实例。Deployer 不能是搜索头群集的成员，而且必须在搜索头群集之外运行。请参阅《分布式搜索》手册中的“使用 deployer 分布应用和配置更新”。

**警告：**切勿从 deployer 之外的任何实例上将配置软件包部署到搜索头群集。如果您在非-deployer 实例（例如群集成员）上运行 `apply schcluster-bundles` 命令，该命令会删除所有现有的应用和所有搜索头群集成员上用户生成的内容！

## 索引器群集

要将应用部署到索引器群集中的对等节点（索引器）上，首先您必须将应用放在索引器群集主节点上适当的位置，然后使用配置软件包方法来将应用分发到对等节点。您可以使用 Splunk Web 或 CLI 将配置软件包应用到对等节点。有关更多信息，请参阅《管理索引器和索引器群集》中的“更新通用对等节点配置和应用”。

当您无法使用部署服务器将应用部署到对等节点时，您可以使用它来分发应用到索引器群集主节点。有关更多信息，请参阅《管理索引器和索引器群集》中的“使用部署服务器分发应用到主节点”。

## 部署应用到 Splunk Cloud

如果您想要部署应用或加载项到 Splunk Cloud，请联系 Splunk 支持以获得指导。支持团队可以部署应用或加载项到部署组件（Splunk Cloud 用户不能访问）上。

## 部署加载项到 Splunk Light

您可以安装并启用选择有限的加载项，在 Splunk Light 实例上配置新的数据输入。请参阅 Splunk Light 《入门手册》中的“配置加载项来添加数据”。

# 应用架构和对象所有权

应用通常由 Splunk **知识对象** 构建而成。Splunk 知识包括诸如保存的搜索、事件类型、标记（用于增强您的 Splunk 数据，方便您找到所需信息的项目）等对象。

**注意：**偶尔您可能也会将对象保存到加载项，但这并不常见。应用和加载项都存储在应用目录中。在很少的情况下您需要将对象保存到加载项，您可按照本主题中为应用介绍的管理方式来管理加载项。

任何登录到 Splunk Web 的用户都可以在所使用的应用下创建知识对象并将其保存到相应的用户目录下（假设具有足够权限）。这是一种默认行为 - 当用户保存某个对象时，该对象会进入当前所运行的应用下相应的用户目录中。用户目录位于 `$SPLUNK_HOME/etc/users/<user_name>/<app_name>/local` 中。一旦用户在该应用中保存了对象之后，这一对象仅对此用户（当其使用该应用时）可用，除非用户采取以下做法之一：

- 提升此对象，以使其对有权访问该应用的所有用户可用
- 将此对象限定于特定角色或用户（仍然在该应用上下文中）
- 使此对象对所有应用、加载项和用户全局可用（除非您明确将其限定于特定角色/用户）

**注意：**用户必须对应用或加载项具有写入权限，方能将对象提升到该级别。

## 提升和共享 Splunk 知识

用户可以通过“权限”对话框与其他用户共享他们的 Splunk 知识对象。这意味着对应用或加载项具有读取权限的用户可以看到共享的对象并使用它们。例如，如果某个用户共享了一个保存的搜索，则其他用户可以看到此搜索，但必须处于创建搜索所在的应用中。因此，如果您在应用 "Fflanda" 中创建了一个保存的搜索并共享，则 Fflanda 的其他用户可以看到保存的搜索，前提是他们对 Fflanda 具有读取权限。

具有写入权限的用户可以将其对象提升到应用级别。这意味着对象将从其用户目录复制到应用的目录 - 从：

```
$SPLUNK_HOME/etc/users/<user_name>/<app_name>/local/
```

至：

```
$SPLUNK_HOME/etc/apps/<app_name>/local/
```

只有那些在应用中具有写入权限的用户可以执行此操作。

## 使 Splunk 知识对象全局可用

最后，在提升对象时，用户可以决定他们是否希望自己的对象全局可用，这意味着所有的应用都能够看到它。同样，用户必须对原始应用具有写入权限。最方便的做法是在 Splunk Web 中完成此操作，不过您也可以直接将相关对象移动到所需目录中。

要将应用 D 中属于用户 C 的对象 A（在 B.conf 中定义）全局可用：



1. 将定义对象 A 的段落从 `$SPLUNK_HOME/etc/users/C/D/B.conf` 移动到 `$SPLUNK_HOME/etc/apps/D/local/B.conf`。

2. 在该应用的 `local.meta` 文件中，向对象 A 的段落部分添加设置 `export = system`。如果此对象的段落不存在，您可以添加相应的段落。

例如，要提升由用户 `fflanda` 在 \*Nix 应用中创建的事件类型 `rhallen` 以使其全局可用：

1. 将 `[rhallen]` 段落从 `$SPLUNK_HOME/etc/users/fflanda/unix/local/eventtypes.conf` 移动到 `$SPLUNK_HOME/etc/apps/unix/local/eventtypes.conf`。

2. 添加以下段落：

```
[eventtypes/rhallen]
export = system
```

至 `$SPLUNK_HOME/etc/apps/unix/metadata/local.meta`。

**注意：**如果您是从“搜索”应用中共享事件类型，则无需将 `export = system` 设置添加到 `local.meta`，因为默认情况下该应用会全局导出其所有事件。

### 这适用于哪些对象？

这里讨论的知识对象仅限于那些受访问控制机制影响的对象。这些对象也称为应用级别对象，可以通过在“用户”菜单栏中选择 **应用 > 管理应用** 来查看它们。所有用户都可以使用此页面来管理他们创建和共享的任何对象。这些对象包括：

- 保存的搜索和报表
- 事件类型
- 视图和仪表板
- 字段提取

还有一些仅具有管理员权限（或对特定对象具有读取/写入权限）用户可用的系统级别对象。这些对象包括：

- 用户
- 角色
- 验证
- 分布式搜索
- 输入
- 输出
- 部署
- 许可证
- 服务器设置（例如：主机名、端口等）

**重要提示：**如果您添加了一个输入，Splunk 会将该输入添加到属于您当前所用应用的 `inputs.conf` 副本中。这意味着如果您直接从“搜索”导航到您的应用，则您的输入将被添加到 `$SPLUNK_HOME/etc/apps/search/local/inputs.conf`，而这可能并不是您希望的行为。

## 应用配置和知识优先顺序

当您向 Splunk 添加知识时，它是在您当前所在的应用上下文中添加的。当 Splunk 评估配置和知识时，它会按照特定的优先顺序来评估它们，因此您可以控制在哪种上下文中使用哪些知识定义与配置。参阅“关于配置文件”以了解有关 Splunk 配置文件和优先顺序的更多信息。

## 管理应用和加载项对象

当 Splunk 用户创建**应用**或**加载项**时，将创建一个构成应用或加载项的对象集合。这些对象可以包括**视图**、**命令**、**导航项目**、**事件类型**、**保存的搜索**、**报表**等等。其中每个对象均具有关联的权限，以确定谁能够查看或改变它们。默认情况下，管理员用户有**权限**改变 Splunk 系统中的所有对象。

参阅这些主题以了解更多信息：

- 有关应用和加载项的概述，请参阅本手册中的[“应用和加载项是什么？”](#)。
- 有关应用和加载项权限的更多信息，请参阅本手册中的[“应用架构和对象所有权”](#)。
- 要了解有关如何创建您自己的应用和加载项的更多信息，请参阅《开发用于 Splunk Web 的视图和应用手册》。


## 在 Splunk Web 中查看和管理应用或加载项对象

要使用 Splunk Web 来查看您的 Splunk 部署中的对象，您可以采用以下方法：

- 要一次查看您的系统中所有应用/加载项的对象：**设置 > 所有配置**。
- 要查看所有保存的搜索和报表对象：**设置 > 搜索和报表**。

- 要查看所有事件类型：**设置 > 事件类型**。
- 要查看所有字段提取：**设置 > 字段**。

您可以：

- 在任何具有**排序箭头**  的页面上查看和操作对象
- 使用**应用上下文**栏，可以对视图进行过滤以只查看那些来自给定应用或加载项的对象、特定用户拥有的对象或包含特定字符串的对象。

使用应用上下文栏上的“搜索”字段在相关字段中搜索字符串。默认情况下，Splunk 会在所有可用字段中搜索字符串。要在特定字段中执行搜索，应指定相应的字段。支持通配符。

**注意：**有关搜索命令页面上各个搜索命令的信息，请参阅《[搜索参考手册](#)》。

## 在 CLI 中更新应用或加载项

要使用 CLI 来更新您的 Splunk 实例上的现有应用：

```
./splunk install app <app_package_filename> -update 1 -auth <username>:<password>
```

Splunk 会根据从安装软件包中找到的信息来更新应用或加载项。

## 使用 CLI 禁用应用或加载项

要通过 CLI 来禁用应用：

```
./splunk disable app [app_name] -auth <username>:<password>
```

**注意：**如果您正在运行 Splunk Free，则无需提供用户名和密码。

## 卸载应用或加载项

要从 Splunk 安装中删除已安装的应用：

**1.**（可选）删除该应用或加载项的索引数据。通常，Splunk 不会从已删除的应用或加载项访问索引数据。不过，您可以使用 Splunk CLI 的 clean 命令来从应用中删除索引数据，然后再删除该应用。参阅“使用 CLI 命令来删除索引数据”。

**2.**删除应用及其目录。应位于 `$SPLUNK_HOME/etc/apps/<appname>` 中。您可在 CLI 中运行以下命令：

```
./splunk remove app [appname] -auth <username>:<password>
```

**3.**您可能需要删除为您的应用或加载项创建的用户特定目录，做法是删除这里查找到的文件（如果有）：`$SPLUNK_HOME/splunk/etc/users/*/<appname>`

**4.**重新启动 Splunk。

## 管理应用和加载项配置及属性

您可以从“应用”菜单中管理安装在 Splunk Enterprise 实例中的应用的配置与属性。单击“用户”栏中的**应用**以选择一个已安装的应用或管理一个应用。从“管理应用”页面上，您可以：

- 编辑应用或加载项的权限
- 启用或禁用应用或加载项
- 执行诸如启动应用、编辑属性和查看应用对象等操作。

## 编辑应用和加载项属性

您对配置和属性的编辑，取决于您是应用的所有者还是用户。

选择**应用 > 管理应用**，然后为要编辑的应用或加载项单击**编辑属性**。您可以对该 Splunk Enterprise 实例中安装的应用进行以下编辑。

- **名称**：更改应用或加载项在 Splunk Web 中的显示名称。
- **更新检查**：默认情况下，更新检查处于启用状态。您可以禁用更新检查和覆盖默认设置。有关详细信息，请参阅下面的[检查应用或加载项更新](#)。
- **可见**：带有视图的应用应是可见的。通常没有视图的加载项应禁用可见属性。
- **上传资产**：使用此字段选择可通过应用或加载项访问的本地文件资产文件，例如 HTML、JavaScript 或 CSS 文件。从此面板一次只能上传一个文件。

关于应用和加载项的配置和属性详情，请参阅“Splunk 开发人员门户”中的“开发 Splunk 应用”。

## 检查更新

您可以配置 Splunk Enterprise 是否在 Splunkbase 中检查应用或加载项更新。默认情况下，启用检查更新。您可以禁用应用更新检查：在**设置 > 应用 > 编辑属性**中编辑此属性。

但是，如果从 Splunk Web 无法访问此属性，您还可以手动编辑应用 `app.conf` 文件来禁用更新检查。在 `$SPLUNK_HOME/etc/apps/<app_name>/local/app.conf` 中创建或编辑以下段落以禁用更新检查：

```
[package]
check_for_updates = 0
```

**注意**：编辑 `app.conf` 的本地版本，而非默认版本。这样可避免用下一个应用更新覆盖您的设置。

# 认识 Hunk

## 认识 Hunk

Hunk 允许您将远程 HDFS 数据存储区配置为虚拟索引器，以便 Splunk 可在本机上报告驻留在 Hadoop 上的数据。虚拟索引配置正确后，您可以报告和可视化驻留在远程 Hadoop 数据存储区上的数据。以下链接指向 Hunk 用户手册中的主题。

## Hunk 手册

简介

- 认识 Hunk
- Hunk 的新功能
- 常见问题
- 了解更多和获取帮助

Hunk 概念

- 关于虚拟索引
- 关于流资源库
- Splunk 如何返回 Hadoop 数据报表
- 关于传递验证

安装 Hunk

- 关于安装和配置 Hunk
- 系统和软件要求
- 下载和安装 Splunk

- 升级 Hunk
- 启动 Splunk
- 授权 Hunk
- Hunk 和 Splunk 配合使用
- 卸载 Hunk
- 使用 Hunk Amazon Machine Image 安装 Hunk

#### 使用配置文件管理 Hunk

- 设置 Splunk 搜索头实例
- 在配置文件中设置提供程序和虚拟索引
- 设置流库
- 添加来源类型
- 管理 Hive 数据
- 配置 Hive 预处理器
- 为 Parquet 配置 Hive 预处理器
- 配置报表加速
- 配置传递验证
- 配置 Kerberos 验证

#### 管理 Hunk 用户界面

- 关于 Hunk 用户界面
- 添加或编辑 HDFS 提供程序
- 添加或编辑虚拟索引
- 设置传递验证

#### 搜索虚拟索引

- 在虚拟索引上使用搜索命令
- 使用报表加速

#### 参考

- Hunk 故障排除
- 性能最佳实践
- 提供程序配置变量
- YARN 必需的配置变量

#### REST API 参考

- 提供程序
- 索引

#### 发行说明

- Hunk 发行说明

### 教程

- 欢迎使用 Hunk 教程

## 管理用户

### 关于用户和角色

您可以创建具有密码的用户，然后将这些用户分配给您已创建的**角色**。Splunk Enterprise Free 不支持用户验证。

Splunk Enterprise 附带单个默认用户（**管理员**用户）。管理员用户的默认密码为 **changeme**。正如该密码所暗示，您应在安装软件后立即更改此密码。

#### 创建用户

Splunk Enterprise 支持三种类型的验证系统，并且在《*确保 Splunk Enterprise 安全*》手册中有所说明。

- **本机验证。**有关更多信息，请参阅“设置带有 Splunk Enterprise 本机验证的用户验证”。
- **LDAP。**Splunk 支持使用其内部验证服务或您的现有 LDAP 服务器进行验证。有关更多信息，请参阅“设置使用 LDAP 进行的用户验证”。
- **脚本式验证 API。**使用脚本式验证将 Splunk 本机验证与外部验证系统（如 RADIUS 或 PAM）连接起来。

有关更多信息，请参阅“设置使用外部系统进行的用户验证”。

## 关于角色

用户会被分配给角色。角色包含一组**操作**。这些功能规定了角色可以执行哪些操作。例如，操作决定是否允许具有特定角色的人员添加输入或编辑保存的搜索。各种功能已在《*确保 Splunk Enterprise 安全*》手册中的“关于使用功能定义角色”中列出。

默认情况下，Splunk Enterprise 具有以下预定义角色：

- 管理员--该角色被分配有最多的操作。
- 高级用户--该角色可以编辑所有共享对象（保存的搜索等），以及告警、标记事件或其他类似任务。
- 普通用户--该角色可以创建并编辑自己的已保存搜索，运行搜索，编辑其首选项，创建并编辑事件类型和其他类似任务。
- 可以删除--此角色允许用户按关键字删除。在使用删除搜索运算符时，才需要此功能。

**注意**不要编辑预定义角色，而是应该创建继承内置角色属性的自定义角色，并按照要求修改自定义角色。

有关角色和如何将用户分配给角色的详细信息，请参阅《*确保 Splunk Enterprise 安全*》手册中的“用户和基于角色的访问控制”一章。

## 查找现有用户和角色

要在 Splunk Web 中查找某个现有用户或角色，可通过选择**设置 > 访问控制**，使用“访问控制”部分中的“用户或角色”页面顶部的搜索栏。支持通配符。默认情况下，Splunk Enterprise 在所有您输入字符串的可用字段中搜索。要在特定字段中执行搜索，应指定相应的字段。例如，要只搜索电子邮件地址，可键入 email=<电子邮件地址或地址片段>:，要只搜索“全名”字段，可键入 realname=<姓名或姓名片段>。要搜索给定角色中的用户，应使用 "roles="。



## 配置用户语言和区域设置

当用户登录时，Splunk 会自动使用在用户浏览器中所设置的语言。要切换语言，请更改浏览器的区域设置。区域设置配置随不同浏览器而有所不同。

Splunk 会检测区域配置字符串。区域配置字符串包含两个组件：语言指示符和本地化区域指示符。它通常表示为两个小写字母和两个大写字母，中间以下划线相连。例如 "en\_US" 表示美国英语，"en\_GB" 表示英国英语。

用户的区域设置还会影响日期、时间和数字的格式，因为不同国家或地区在这些方面具有不同的格式标准。

Splunk 针对以下区域设置提供内置支持：

```
de_DE
en_GB
en_US
fr_FR
it_IT
ja_JP
ko_KR
zh_CN
zh_TW
```

如果您要针对其他语言添加本地化支持，请参阅《开发人员手册》中的“翻译 Splunk”以获得相关指南。然后，您需要告诉您的用户在其浏览器中指定适当的区域设置。

## 浏览器区域设置如何影响时间戳格式

默认情况下，根据浏览器区域设置来确定 Splunk 中的时间戳格式。如果浏览器被配置为美国英语，则时间戳将采用美国的日期格式：MM/DD/YYYY:HH:MM:SS。如果浏览器配置为英国英语，则时间戳将采用欧洲日期格式：DD/MM/YYYY:HH:MM:SS。

有关时间戳格式的更多信息，请参阅《数据导入手册》中的“配置时间戳识别”。

## 覆盖浏览器区域设置

通过修改您用来访问 Splunk 的 URL，您可以更改 Splunk 针对给定会话所采用的区域设置。Splunk URL 会遵循格式 http://host:port/locale/...。例如，当您访问并登录 Splunk 时，对于美国英语，URL 会显示为 http://hostname:8000/en-US/account/login。要使用英国英语设置，您可以将区域设置字符串更改为 http://hostname:8000/en-GB/account/login。然后，该会话将以英国英语格式显示时间戳，并且在此期间接受这种格式

的时间戳。

如果请求 Splunk 界面不支持尚未本地化的区域设置，则会显示消息：`Invalid language Specified`。

参阅《开发人员手册》中的“翻译 Splunk”以获得有关本地化 Splunk 的更多信息。

## 配置用户会话超时

在 Splunk 用户会话超时之前经历的时间取决于以下三项超时设置之间的相互作用：

- `splunkweb` 会话超时。
- `splunkd` 会话超时。
- 浏览器会话超时。

`splunkweb` 和 `splunkd` 超时决定了浏览器与 Splunk 间相互作用的最大空闲时间。浏览器会话超时决定了用户与浏览器间相互作用的最大空闲时间。

`splunkweb` 和 `splunkd` 超时通常具有相同的值，因为它们通过同一字段进行设置。在 Splunk Web 中设置此超时：

1. 单击 Splunk Web 右上角的**设置**。
2. 在“系统”下，单击**服务器设置**。
3. 单击**常规设置**。
4. 在**会话超时**字段中，输入超时值。
5. 单击**保存**。

这将同时为 `splunkweb` 和 `splunkd` 设置用户会话超时值。它们的初始值均为 60 分钟。如果您通过 Splunk Web 来更改超时值，则它们的值始终保持相同。

如果由于某些原因，您需要为 `splunkweb` 和 `splunkd` 设置不同的超时值，那么您可以编辑其基本配置文件 `web.conf` (`tools.sessions.timeout` 属性) 和 `server.conf` (`sessionTimeout` 属性) 来做到这一点。就实际应用而言，没有理由为它们指定不同的超时值。在任何情况下，如果用户正在使用 SplunkWeb (`splunkweb`) 来访问 Splunk 实例 (`splunkd`)，那么这两个超时属性中以值较小的为准。因此，如果 `web.conf` 中的 `tools.sessions.timeout` 值为 "90" (分钟)，`server.conf` 中的 `sessionTimeout` 值为 "1h" (1 小时；60 分钟)，则会话将在 60 分钟后超时。

除了设置 `splunkweb/splunkd` 会话值之外，您还可以通过编辑 `web.conf` 中的 `ui_inactivity_timeout` 值来指定用户浏览器会话的超时。Splunk 浏览器会话将在达到该值时超时。默认值为 60 分钟。如果 `ui_inactivity_timeout` 被设为小于 1，则不会发生超时 -- 只要浏览器处于开启状态，会话就不会超时。

只有当浏览器会话达到其超时值之后，`splunkweb/splunkd` 会话超时才会开始倒计时。因此，要确定在超时之前要经历多长时间，应将 `ui_inactivity_timeout` 值加上 `splunkweb` 和 `splunkd` 超时值中较小的一个。例如，假设以下条件：

- `splunkweb` 超时：15m
- `splunkd` 超时：20m
- 浏览器 (`ui_inactivity_timeout`) 超时：10m

则用户会话的活动状态将保持 25m (15m+10m)。如果连续 25 分钟无任何活动，则用户将被提示重新登录。

**注意：**如果更改了超时值（通过 Splunk Web 或配置文件），则必须重新启动 Splunk 以使更改生效。

## 配置文件参考

### alert\_actions.conf

以下为 `alert_actions.conf` 的规范和示例文件。

#### alert\_actions.conf.spec

```
# Version 6.5.0
#
# This file contains possible attributes and values for configuring global
# saved search actions in alert_actions.conf. Saved searches are configured
# in savedsearches.conf.
#
# There is an alert_actions.conf in $SPLUNK_HOME/etc/system/default/.
# To set custom configurations, place an alert_actions.conf in
```

```
# $SPLUNK_HOME/etc/system/local/. For examples, see
# alert_actions.conf.example. You must restart Splunk to enable
# configurations.
#
# To learn more about configuration files (including precedence) please see
# the documentation located at
# http://docs.splunk.com/Documentation/Splunk/latest/Admin/Aboutconfigurationfiles
```

## 全局设置

```
# GLOBAL SETTINGS
# Use the [default] stanza to define any global settings.
# * You can also define global settings outside of any stanza, at the top
#   of the file.
# * Each conf file should have at most one default stanza. If there are
#   multiple default stanzas, attributes are combined. In the case of
#   multiple definitions of the same attribute, the last definition in the
#   file wins.
# * If an attribute is defined at both the global level and in a specific
#   stanza, the value in the specific stanza takes precedence.

maxresults = <integer>
* Set the global maximum number of search results sent via alerts.
* Defaults to 100.

hostname = [protocol]<host>[:<port>]
* Sets the hostname used in the web link (url) sent in alerts.
* This value accepts two forms.
  * hostname
    examples: splunkserver, splunkserver.example.com
  * protocol://hostname:port
    examples: http://splunkserver:8000, https://splunkserver.example.com:443
* When this value is a simple hostname, the protocol and port which
  are configured within splunk are used to construct the base of
  the url.
* When this value begins with 'http://', it is used verbatim.
  NOTE: This means the correct port must be specified if it is not
  the default port for http or https.
* This is useful in cases when the Splunk server is not aware of
  how to construct an externally referenceable url, such as SSO
  environments, other proxies, or when the Splunk server hostname
  is not generally resolvable.
* Defaults to current hostname provided by the operating system,
  or if that fails, "localhost".
* When set to empty, default behavior is used.

ttl      = <integer>[p]
* Optional argument specifying the minimum time to live (in seconds)
  of the search artifacts, if this action is triggered.
* If p follows integer, then integer is the number of scheduled periods.
* If no actions are triggered, the artifacts will have their ttl determined
  by the "dispatch.ttl" attribute in savedsearches.conf.
* Defaults to 10p
* Defaults to 86400 (24 hours)   for: email, rss
* Defaults to   600 (10 minutes) for: script
* Defaults to   120 (2 minutes)  for: summary_index, populate_lookup

maxtime = <integer>[m|s|h|d]
* The maximum amount of time that the execution of an action is allowed to
  take before the action is aborted.
* Use the d, h, m and s suffixes to define the period of time:
  d = day, h = hour, m = minute and s = second.
  For example: 5d means 5 days.
* Defaults to 5m for everything except rss.
* Defaults to 1m for rss.

track_alert = [1|0]
* Indicates whether the execution of this action signifies a trackable alert.
* Defaults to 0 (false).
```

```

command = <string>
* The search command (or pipeline) which is responsible for executing
  the action.
* Generally the command is a template search pipeline which is realized
  with values from the saved search - to reference saved search
  field values wrap them in dollar signs ($).
* For example, to reference the savedsearch name use $name$. To
  reference the search, use $search$

is_custom = [1|0]
* Specifies whether the alert action is based on the custom alert
  actions framework and is supposed to be listed in the search UI.

payload_format = [xml|json]
* Configure the format the alert script receives the configuration via
  STDIN.
* Defaults to "xml"

label = <string>
* For custom alert actions: Define the label shown in the UI. If not
  specified, the stanza name will be used instead.

description = <string>
* For custom alert actions: Define the description shown in the UI.

icon_path = <string>
* For custom alert actions: Define the icon shown in the UI for the alert
  action. The path refers to appserver/static within the app where the
  alert action is defined in.

alert.execute.cmd = <string>
* For custom alert actions: Explicitly specify the command to be executed
  when the alert action is triggered. This refers to a binary or script
  in the bin folder of the app the alert action is defined in, or to a
  path pointer file, also located in the bin folder.
* If a path pointer file (*.path) is specified, the contents of the file
  is read and the result is used as the command to be executed.
  Environment variables in the path pointer file are substituted.
* If a python (*.py) script is specified it will be prefixed with the
  bundled python interpreter.

alert.execute.cmd.arg.<n> = <string>
* Provide additional arguments to the alert action execution command.
  Environment variables are substituted.

#####
# EMAIL: these settings are prefaced by the [email] stanza name
#####

```

## **[email]**

```

[email]
* Set email notification options under this stanza name.
* Follow this stanza name with any number of the following
  attribute/value pairs.
* If you do not specify an entry for each attribute, Splunk will
  use the default value.

```

```

from = <string>
* Email address from which the alert originates.
* Defaults to splunk@$LOCALHOST.

```

```

to = <string>
* The To email address receiving the alert.

```

```

cc = <string>
* Any cc email addresses receiving the alert.

```

```

bcc = <string>
* Any bcc email addresses receiving the alert.

```



```

message.report = <string>
* Specify a custom email message for scheduled reports.
* Includes the ability to reference attributes from the result,
  saved search, or job

message.alert = <string>
* Specify a custom email message for alerts.
* Includes the ability to reference attributes from result,
  saved search, or job

subject = <string>
* Specify an alternate email subject if useNSSubject is false.
* Defaults to SplunkAlert-<savedsearchname>.

subject.alert = <string>
* Specify an alternate email subject for an alert.
* Defaults to SplunkAlert-<savedsearchname>.

subject.report = <string>
* Specify an alternate email subject for a scheduled report.
* Defaults to SplunkReport-<savedsearchname>.

useNSSubject = [1|0]
* Specify whether to use the namespaced subject (i.e subject.report) or
  subject.

footer.text = <string>
* Specify an alternate email footer.
* Defaults to "If you believe you've received this email in error, please see your Splunk
administrator.\r\n\r\nsplunk > the engine for machine data."

format = [table|raw|csv]
* Specify the format of inline results in the email.
* Accepted values: table, raw, and csv.
* Previously accepted values plain and html are no longer respected
  and equate to table.
* To make emails plain or html use the content_type attribute.

include.results_link = [1|0]
* Specify whether to include a link to the results.

include.search = [1|0]
* Specify whether to include the search that caused an email to be sent.

include.trigger = [1|0]
* Specify whether to show the trigger condition that caused the alert to
  fire.

include.trigger_time = [1|0]
* Specify whether to show the time that the alert was fired.

include.view_link = [1|0]
* Specify whether to show the title and a link to enable the user to edit
  the saved search.

content_type = [html|plain]
* Specify the content type of the email.
  * plain sends email as plain text
  * html sends email as a multipart email that include both text and html.

sendresults = [1|0]
* Specify whether the search results are included in the email. The
  results can be attached or inline, see inline (action.email.inline)
* Defaults to 0 (false).

inline = [1|0]
* Specify whether the search results are contained in the body of the alert
  email.
* If the events are not sent inline, they are attached as a csv text.
* Defaults to 0 (false).

priority = [1|2|3|4|5]

```

- \* Set the priority of the email as it appears in the email client.
- \* Value mapping: 1 highest, 2 high, 3 normal, 4 low, 5 lowest.
- \* Defaults to 3.

mailserver = <host>[:<port>]

- \* You must have a Simple Mail Transfer Protocol (SMTP) server available to send email. This is not included with Splunk.
- \* Specifies the SMTP mail server to use when sending emails.
- \* <host> can be either the hostname or the IP address.
- \* Optionally, specify the SMTP <port> that Splunk should connect to.
- \* When the "use\_ssl" attribute (see below) is set to 1 (true), you must specify both <host> and <port>.
- (Example: "example.com:465")
- \* Defaults to \$LOCALHOST:25.

use\_ssl = [1|0]

- \* Whether to use SSL when communicating with the SMTP server.
- \* When set to 1 (true), you must also specify both the server name or IP address and the TCP port in the "mailserver" attribute.
- \* Defaults to 0 (false).

use\_tls = [1|0]

- \* Specify whether to use TLS (transport layer security) when communicating with the SMTP server (starttls)
- \* Defaults to 0 (false).

auth\_username = <string>

- \* The username to use when authenticating with the SMTP server. If this is not defined or is set to an empty string, no authentication is attempted.
- NOTE: your SMTP server might reject unauthenticated emails.
- \* Defaults to empty string.

auth\_password = <password>

- \* The password to use when authenticating with the SMTP server.
- Normally this value will be set when editing the email settings, however you can set a clear text password here and it will be encrypted on the next Splunk restart.
- \* Defaults to empty string.

sendpdf = [1|0]

- \* Specify whether to create and send the results as a PDF.
- \* Defaults to 0 (false).

sendcsv = [1|0]

- \* Specify whether to create and send the results as a csv file.
- \* Defaults to 0 (false).

pdfview = <string>

- \* Name of view to send as a PDF

reportPaperSize = [letter|legal|ledger|a2|a3|a4|a5]

- \* Default paper size for PDFs
- \* Accepted values: letter, legal, ledger, a2, a3, a4, a5
- \* Defaults to "letter".

reportPaperOrientation = [portrait|landscape]

- \* Paper orientation: portrait or landscape
- \* Defaults to "portrait".

reportIncludeSplunkLogo = [1|0]

- \* Specify whether to include a Splunk logo in Integrated PDF Rendering
- \* Defaults to 1 (true)

reportCIDFontList = <string>

- \* Specify the set (and load order) of CID fonts for handling Simplified Chinese(gb), Traditional Chinese(cns), Japanese(jp), and Korean(kor) in Integrated PDF Rendering.
- \* Specify in a space-separated list
- \* If multiple fonts provide a glyph for a given character code, the glyph from the first font specified in the list will be used
- \* To skip loading any CID fonts, specify the empty string
- \* Defaults to "gb cns jp kor"

```

reportFileName = <string>
    * Specify the name of attached pdf or csv
    * Defaults to "$name$-time:%Y-%m-%d$"

width_sort_columns = <bool>
    * Whether columns should be sorted from least wide to most wide left to right.
    * Valid only if format=text
    * Defaults to true

preprocess_results = <search-string>
    * Supply a search string to Splunk to preprocess results before emailing
    them. Usually the preprocessing consists of filtering out unwanted
    internal fields.
    * Defaults to empty string (no preprocessing)

pdf.footer_enabled = [1 or 0]
    * Set whether or not to display footer on PDF.
    * Defaults to 1.

pdf.header_enabled = [1 or 0]
    * Set whether or not to display header on PDF.
    * Defaults to 1.

pdf.logo_path = <string>
    * Define pdf logo by syntax <app>:<path-to-image>
    * If set, PDF will be rendered with this logo instead of Splunk one.
    * If not set, Splunk logo will be used by default
    * Logo will be read from $SPLUNK_HOME/etc/apps/<app>/appserver/static/<path-to-image> if <app> is provided.
    * Current app will be used if <app> is not provided.

pdf.header_left = [logo|title|description|timestamp|pagination|none]
    * Set which element will be displayed on the left side of header.
    * Nothing will be display if this option is not been set or set to none
    * Defaults to None, nothing will be displayed on this position.

pdf.header_center = [logo|title|description|timestamp|pagination|none]
    * Set which element will be displayed on the center of header.
    * Nothing will be display if this option is not been set or set to none
    * Defaults to description

pdf.header_right = [logo|title|description|timestamp|pagination|none]
    * Set which element will be displayed on the right side of header.
    * Nothing will be display if this option is not been set or set to none
    * Defaults to None, nothing will be displayed on this position.

pdf.footer_left = [logo|title|description|timestamp|pagination|none]
    * Set which element will be displayed on the left side of footer.
    * Nothing will be display if this option is not been set or set to none
    * Defaults to logo

pdf.footer_center = [logo|title|description|timestamp|pagination|none]
    * Set which element will be displayed on the center of footer.
    * Nothing will be display if this option is not been set or set to none
    * Defaults to title

pdf.footer_right = [logo|title|description|timestamp|pagination|none]
    * Set which element will be displayed on the right side of footer.
    * Nothing will be display if this option is not been set or set to none
    * Defaults to timestamp,pagination

pdf.html_image_rendering = <bool>
    * Whether images in HTML should be rendered.
    * If enabling rendering images in HTML breaks the pdf for whatever reason,
    * it could be disabled by setting this flag to False,
    * so the old HTML rendering will be used.
    * Defaults to True.

sslVersions = <versions_list>
    * Comma-separated list of SSL versions to support.
    * The versions available are "ssl3", "tls1.0", "tls1.1", and "tls1.2".
    * The special version "*" selects all supported versions. The version "tls"

```

- selects all versions tls1.0 or newer.
- \* If a version is prefixed with "-" it is removed from the list.
- \* SSLv2 is always disabled; "-ssl2" is accepted in the version list but does nothing.
- \* When configured in FIPS mode, ssl3 is always disabled regardless of this configuration.
- \* Defaults to "\*",-ssl2" (anything newer than SSLv2).

sslVerifyServerCert = true|false

- \* If this is set to true, you should make sure that the server that is being connected to is a valid one (authenticated). Both the common name and the alternate name of the server are then checked for a match if they are specified in this configuration file. A certificate is considered verified if either is matched.
- \* If this is set to true, make sure 'server.conf/[sslConfig]/sslRootCAPath' has been set correctly.
- \* Default is false.

sslCommonNameToCheck = <commonName1>, <commonName2>, ...

- \* Optional. Defaults to no common name checking.
- \* Check the common name of the server's certificate against this list of names.
- \* 'sslVerifyServerCert' must be set to true for this setting to work.

sslAltNameToCheck = <alternateName1>, <alternateName2>, ...

- \* Optional. Defaults to no alternate name checking.
- \* Check the alternate name of the server's certificate against this list of names.
- \* If there is no match, assume that Splunk is not authenticated against this server.
- \* 'sslVerifyServerCert' must be set to true for this setting to work.

cipherSuite = <cipher suite string>

- \* If set, Splunk uses the specified cipher string for the communication with the SMTP server.
- \* If not set, Splunk uses the default cipher string provided by OpenSSL.
- \* This is used to ensure that the client does not make connections using weak encryption protocols.
- \* Default is 'TLSv1+HIGH:TLSv1.2+HIGH:@STRENGTH'.

```
#####
# RSS: these settings are prefaced by the [rss] stanza
#####
```

## **[rss]**

[rss]

- \* Set RSS notification options under this stanza name.
- \* Follow this stanza name with any number of the following attribute/value pairs.
- \* If you do not specify an entry for each attribute, Splunk will use the default value.

items\_count = <number>

- \* Number of saved RSS feeds.
- \* Cannot be more than maxresults (in the global settings).
- \* Defaults to 30.

```
#####
# script: Used to configure any scripts that the alert triggers.
#####
```

## **[script]**

[script]

filename = <string>

- \* The filename, with no path, of the script to trigger.
- \* The script should be located in: \$SPLUNK\_HOME/bin/scripts/
- \* For system shell scripts on Unix, or .bat or .cmd on windows, there are no further requirements.
- \* For other types of scripts, the first line should begin with a #! marker, followed by a path to the interpreter that will run the script.
- \* Example: #!C:\Python27\python.exe

\* Defaults to empty string.

```
#####
# summary_index: these settings are prefaced by the [summary_index] stanza
#####
```

### **[summary\_index]**

[summary\_index]

inline = [1|0]

\* Specifies whether the summary index search command will run as part of the scheduled search or as a follow-on action. This is useful when the results of the scheduled search are expected to be large.

\* Defaults to 1 (true).

\_name = <string>

\* The name of the summary index where Splunk will write the events.

\* Defaults to "summary".

```
#####
# populate_lookup: these settings are prefaced by the [populate_lookup] stanza
#####
```

### **[populate\_lookup]**

[populate\_lookup]

dest = <string>

\* Name of the lookup table to populate (stanza name in transforms.conf) or the lookup file path to where you want the data written. If a path is specified it MUST be relative to \$SPLUNK\_HOME and a valid lookups directory.

For example: "etc/system/lookups/<file-name>" or

"etc/apps/<app>/lookups/<file-name>"

\* The user executing this action MUST have write permissions to the app for this action to work properly.

## **alert\_actions.conf.example**

# Version 6.5.0

#

# This is an example alert\_actions.conf. Use this file to configure alert actions for saved searches.

#

# To use one or more of these configurations, copy the configuration block into alert\_actions.conf in \$SPLUNK\_HOME/etc/system/local/. You must restart Splunk to enable configurations.

#

# To learn more about configuration files (including precedence) please see the documentation located at

# <http://docs.splunk.com/Documentation/Splunk/latest/Admin/Aboutconfigurationfiles>

[email]

# keep the search artifacts around for 24 hours

ttl = 86400

# if no @ is found in the address the hostname of the current machine is appended  
from = splunk

format = table

inline = false

sendresults = true

hostname = CanAccessFromTheWorld.com

command = sendemail "to=\$action.email.to\$" "server=\$action.email.mailserver(default=localhost)\$"

```

"from=$action.email.from(default=splunk@localhost)$" "subject=$action.email.subject(recurse=yes)$"
"format=$action.email.format(default=csv)$" "sssummary=Saved Search [$name]: $counttype($results.count)"
"sslink=$results.url$" "ssquery=$search$" "ssname=$name$" "inline=$action.email.inline(default=False)$"
"sendresults=$action.email.sendresults(default=False)$" "sendpdf=$action.email.sendpdf(default=False)$"
"pdfview=$action.email.pdfview$" "searchid=$search_id$" "graceful=$graceful(default=True)$"
maxinputs="$maxinputs(default=1000)$" maxtime="$action.email.maxtime(default=5m)$"
_validate-1 = action.email.sendresults, validate( is_bool('action.email.sendresults'), "Value of argument
'action.email.sendresults' must be a boolean")

use_tls = 1
sslVersions = tls1.2
sslVerifyServerCert = true
sslCommonNameToCheck = host1, host2

[rss]
# at most 30 items in the feed
items_count=30

# keep the search artifacts around for 24 hours
ttl = 86400

command = createrss "path=$name.xml" "name=$name$" "link=$results.url$" "descr=Alert trigger: $name$,
results.count=$results.count$ " "count=30" "graceful=$graceful(default=1)$"
maxtime="$action.rss.maxtime(default=1m)$"

[summary_index]
# don't need the artifacts anytime after they're in the summary index
ttl = 120

# make sure the following keys are not added to marker (command, ttl, maxresults, _)
command = summaryindex addtime=true index="$action.summary_index._name(required=yes)$" file="$name$_$#random$.stash"
name="$name$" marker="$action.summary_index*{format=$KEY=\\\\"$VAL\\",
key_regex="action.summary_index.(?!(:command|maxresults|ttl|(:_.*))$)(.*)"$

[custom_action]
# flag the action as custom alert action
is_custom = 1

# configure appearance in the UI
label = Custom Alert Action
description = Triggers a custom alert action
icon_path = custom_alert.png

# override default script execution
# java.path is a path pointer file in <app>/bin pointing to the actual java executable
alert.execute.cmd = java.path
alert.execute.cmd.arg.1 = -jar
alert.execute.cmd.arg.2 = $SPLUNK_HOME/etc/apps/myapp/bin/custom.jar
alert.execute.cmd.arg.3 = --execute

```

## app.conf

以下为 app.conf 的规范和示例文件。

### app.conf.spec

```

# Version 6.5.0
#
# This file maintains the state of a given app in Splunk Enterprise. It may also be used
# to customize certain aspects of an app.
#
# There is no global, default app.conf. Instead, an app.conf may exist in each
# app in Splunk Enterprise.
#
# You must restart Splunk Enterprise to reload manual changes to app.conf.
#
# To learn more about configuration files (including precedence) please see the
# documentation located at
# http://docs.splunk.com/Documentation/Splunk/latest/Admin/Aboutconfigurationfiles

```

```
#
# Settings for how an app appears in Launcher (and online on Splunkbase)
#

[launcher]

[launcher]
# global setting

remote_tab = <bool>
* Set whether the Launcher interface will connect to apps.splunk.com.
* This setting only applies to the Launcher app and should be not set in any
  other app
* Defaults to true.

# per-application settings

version = <version string>
* Version numbers are a number followed by a sequence of dots and numbers.
* Version numbers for releases should use three digits.
* Pre-release versions can append a single-word suffix like "beta" or "preview."
* Pre-release designations should use lower case and no spaces.
* Examples:
  * 1.2.0
  * 3.2.1
  * 11.0.34
  * 2.0beta
  * 1.3beta2
  * 1.0preview

description = <string>
* Short explanatory string displayed underneath the app's title in Launcher.
* Descriptions should be 200 characters or less because most users won't read
  long descriptions!

author = <name>
* For apps you intend to post to Splunkbase, enter the username of your
  splunk.com account.
* For internal-use-only apps, include your full name and/or contact info
  (e.g. email).

# Your app can include an icon which will show up next to your app in Launcher
# and on Splunkbase. You can also include a screenshot, which will show up on
# Splunkbase when the user views info about your app before downloading it.
# Icons are recommended, although not required.
# Screenshots are optional.
#
# There is no setting in app.conf for these images. Instead, icon and
# screenshot images should be placed in the appserver/static dir of
# your app. They will automatically be detected by Launcher and Splunkbase.
#
# For example:
#
#   <app_directory>/appserver/static/appIcon.png    (the capital "I" is required!)
#   <app_directory>/appserver/static/screenshot.png
#
# An icon image must be a 36px by 36px PNG file.
# An app screenshot must be 623px by 350px PNG file.

#
# [package] defines upgrade-related metadata, and will be
# used in future versions of Splunk Enterprise to streamline app upgrades.
#
```

## **[package]**

```
[package]
id = <appid>
* id should be omitted for internal-use-only apps which are not intended to be
```

```

    uploaded to Splunkbase
* id is required for all new apps uploaded to Splunkbase. Future versions of
  Splunk Enterprise will use appid to correlate locally-installed apps and the
  same app on Splunkbase (e.g. to notify users about app updates)
* id must be the same as the folder name in which your app lives in
  $SPLUNK_HOME/etc/apps
* id must adhere to cross-platform folder-name restrictions:
  * must contain only letters, numbers, "." (dot), and "_" (underscore) characters
  * must not end with a dot character
  * must not be any of the following names: CON, PRN, AUX, NUL,
    COM1, COM2, COM3, COM4, COM5, COM6, COM7, COM8, COM9,
    LPT1, LPT2, LPT3, LPT4, LPT5, LPT6, LPT7, LPT8, LPT9

check_for_updates = <bool>
* Set whether Splunk Enterprise should check Splunkbase for updates to this app.
* Defaults to true.

```

```

#
# Set install settings for this app
#

```

## **[install]**

```

[install]
state = disabled | enabled
* Set whether app is disabled or enabled.
* If an app is disabled, its configs are ignored.
* Defaults to enabled.

state_change_requires_restart = true | false
* Set whether changing an app's state ALWAYS requires a restart of Splunk Enterprise.
* State changes include enabling or disabling an app.
* When set to true, changing an app's state always requires a restart.
* When set to false, modifying an app's state may or may not require a restart
  depending on what the app contains. This setting cannot be used to avoid all
  restart requirements!
* Defaults to false.

is_configured = true | false
* Stores indication of whether the application's custom setup has been performed
* Defaults to false

build = <integer>
* Required.
* Must be a positive integer.
* Increment this whenever you change files in appserver/static.
* Every release must change both "version" and "build" settings.
* Ensures browsers don't use cached copies of old static files
  in new versions of your app.
* Build is a single integer, unlike version which can be a complex string
  like 1.5.18.

allows_disable = true | false
* Set whether an app allows itself to be disabled.
* Defaults to true.

install_source_checksum = <string>
* Records a checksum of the tarball from which a given app was installed.
* Splunk Enterprise will automatically populate this value upon install.
* You should *not* set this value explicitly within your app!

#
# Handle reloading of custom .conf files (4.2+ versions only)
#

```

## **[triggers]**

```

[triggers]

```



```
reload.<conf_file_name> = [ simple | rest_endpoints | access_endpoints <handler_url> | http_get <handler_url> |
http_post <handler_url> ]
```

- \* Splunk Enterprise will reload app configuration after every app-state change: install, update, enable, and disable.
- \* If your app does not use a custom config file (e.g. myconffile.conf) then it won't need a [triggers] stanza, because \$SPLUNK\_HOME/etc/system/default/app.conf already includes a [triggers] stanza which automatically reloads config files normally used by Splunk Enterprise.
- \* If your app uses a custom config file (e.g. myconffile.conf) and you want to avoid unnecessary Splunk Enterprise restarts, you'll need to add a reload value in the [triggers] stanza.
- \* If you don't include [triggers] settings and your app uses a custom config file, a Splunk Enterprise restart will be required after every state change.
- \* Specifying "simple" implies that Splunk Enterprise will take no special action to reload your custom conf file.
- \* Specify "access\_endpoints" and a URL to a REST endpoint, and Splunk Enterprise will call its \_reload() method at every app state change.
- \* Specify "http\_get" and a URL to a REST endpoint, and Splunk Enterprise will simulate an HTTP GET request against this URL at every app state change.
- \* Specify "http\_post" and a URL to a REST endpoint, and Splunk Enterprise will simulate an HTTP POST request against this URL at every app state change.
- \* "rest\_endpoints" is reserved for Splunk Enterprise internal use for reloading restmap.conf.

\* Examples:

## **[triggers]**

```
[triggers]
    * Do not force a restart of Splunk Enterprise for state changes of MyApp
    * Do not run special code to tell MyApp to reload myconffile.conf
    * Apps with custom config files will usually pick this option
    reload.myconffile = simple

    * Do not force a restart of Splunk Enterprise for state changes of MyApp.
    * Splunk Enterprise calls the /admin/myendpoint/_reload method in my custom EAI handler.
    * Use this advanced option only if MyApp requires custom code to reload its configuration when its state
changes
    reload.myotherconffile = access_endpoints /admin/myendpoint

#
# Set UI-specific settings for this app
#
```

## **[ui]**

```
[ui]
is_visible = true | false
* Indicates if this app should be visible/navigable as a UI app
* Apps require at least 1 view to be available from the UI

show_in_nav = true | false
* Indicates if this app should be shown in global app dropdown

is_manageable = true | false
* Support for this setting has been removed. It no longer has any effect.

label = <string>
* Defines the name of the app shown in the Splunk Enterprise GUI and Launcher
* Recommended length between 5 and 80 characters.
* Must not include "Splunk For" prefix.
* Label is required.
* Examples of good labels:
    IMAP Monitor
    SQL Server Integration Services
    FISMA Compliance

docs_section_override = <string>
* Defines override for auto-generated app-specific documentation links
* If not specified, app-specific documentation link will
```

```

include [<app-name>:<app-version>]
* If specified, app-specific documentation link will
include [<docs_section_override>]
* This only applies to apps with documentation on the Splunk documentation site

attribution_link = <string>
* URL that users can visit to find third-party software credits and attributions for assets the app uses.
* External links must start with http:// or https://.
* Values that do not start with http:// or https:// will be interpreted as Quickdraw "location" strings
* and translated to internal documentation references.

setup_view = <string>
* Optional setting
* Defines custom setup view found within /data/ui/views REST endpoint
* If not specified, default to setup.xml

#
# Credential-verification scripting (4.2+ versions only)
# Credential entries are superseded by passwords.conf from 6.3 onwards.
# While the entries here are still honored post-6.3, updates to these will occur in passwords.conf which will shadow
any values present here.
#

```

### **[credentials\_settings]**

```

[credentials_settings]
verify_script = <string>
* Optional setting.
* Command line to invoke to verify credentials used for this app.
* For scripts, the command line should include both the interpreter and the
  script for it to run.
  * Example: "$SPLUNK_HOME/bin/python" "$SPLUNK_HOME/etc/apps/<myapp>/bin/$MY_SCRIPT"
* The invoked program is communicated with over standard in / standard out via
  the same protocol as splunk scripted auth.
* Paths incorporating variable expansion or explicit spaces must be quoted.
  * For example, a path including $SPLUNK_HOME should be quoted, as likely
    will expand to C:\Program Files\Splunk

```

### **[credential:<realm>:<username>]**

```

[credential:<realm>:<username>]
password = <password>
* Password that corresponds to the given username for the given realm.
  Note that realm is optional
* The password can be in clear text, however when saved from splunkd the
  password will always be encrypted

# diag app extensions, 6.4+ only

```

### **[diag]**

```

[diag]
extension_script = <filename>
* Setting this variable declares that this app will put additional information
  into the troubleshooting & support oriented output of the 'splunk diag'
  command.
* Must be a python script.
* Must be a simple filename, with no directory separators.
* The script must exist in the 'bin' sub-directory in the app.
* Full discussion of the interface is located on the Developer portal.
  See http://dev.splunk.com/view/SP-CAA8E8H
* Defaults to unset, no app-specific data collection will occur.

data_limit = <positive integer>[b|kb|MB|GB]
* Defines a soft-ceiling for the amount of uncompressed data that should be
  added to the diag by the app extension.
* Large diags damage the main functionality of the tool by creating data blobs
  too large to copy around or upload.

```

- \* Use this setting to ensure that your extension script does not accidentally produce far too much data.
- \* Once data produced by this app extension reaches the limit, diag will not add any further files on behalf of the extension.
- \* After diag has finished adding a file which goes over this limit, all further files will not be added.
- \* Must be a positive number followed by a size suffix.
  - \* Valid suffixes: b: bytes, kb: kilobytes, mb: megabytes, gb: gigabytes
  - \* Suffixes are case insensitive.
- \* Defaults to 100MB.

# Other diag settings

default\_gather\_lookups = <filename> [, <filename> ...]

- \* Setting this variable declares that the app contains lookups which should always be gathered by diag (by default).
- \* Essentially, if there are lookups which are useful for troubleshooting an app, and will never contain sensitive (user) data, they can be added to this list, and they will appear in generated diags for use when troubleshooting the app from customer diags.
- \* Any files in lookup dirs which are not listed here are not gathered by default; this can be overridden with the diag flag --include-lookups
- \* This setting is new in Splunk Enterprise/Light version 6.5. Older versions gather all lookups by default.
- \* This does not override the size-ceiling on files in etc. Large lookups will still be excluded, unless the etc-filesize-limit is raised/disabled.
- \* This controls only files in the same app directory as this conf file. For example, if you have an app directory in etc/slave-apps (index clustering), this setting must appear in etc/slave-apps/appname/default/app.conf or local/app.conf
- \* Additional lists can be created with default\_gather\_lookups-classname = ...
- \* Defaults to unset.

## app.conf.example

```
# Version 6.5.0
#
# The following are example app.conf configurations. Configure properties for
# your custom application.
#
# There is NO DEFAULT app.conf.
#
# To use one or more of these configurations, copy the configuration block into
# app.conf in $SPLUNK_HOME/etc/system/local/. You must restart Splunk to
# enable configurations.
#
# To learn more about configuration files (including precedence) please see the
# documentation located at
# http://docs.splunk.com/Documentation/Splunk/latest/Admin/Aboutconfigurationfiles

[launcher]
author=<author of app>
description=<textual description of app>
version=<version of app>
```

## audit.conf

以下为 audit.conf 的规范和示例文件。

### audit.conf.spec

```
# Version 6.5.0
#
# This file contains possible attributes and values you can use to configure
# auditing and event signing in audit.conf.
```

```
#
# There is NO DEFAULT audit.conf. To set custom configurations, place an
# audit.conf in $SPLUNK_HOME/etc/system/local/. For examples, see
# audit.conf.example. You must restart Splunk to enable configurations.
#
# To learn more about configuration files (including precedence) please see the
# documentation located at
# http://docs.splunk.com/Documentation/Splunk/latest/Admin/Aboutconfigurationfiles
```

## 全局设置

```
# GLOBAL SETTINGS
# Use the [default] stanza to define any global settings.
# * You can also define global settings outside of any stanza, at the top of the file.
# * Each conf file should have at most one default stanza. If there are
# multiple default stanzas, attributes are combined. In the case of multiple
# definitions of the same attribute, the last definition in the file wins.
# * If an attribute is defined at both the global level and in a specific
# stanza, the value in the specific stanza takes precedence.

#####
# KEYS: specify your public and private keys for encryption.
#####
```

## [auditTrail]

```
[auditTrail]
* This stanza turns on cryptographic signing for audit trail events (set in inputs.conf).
* You must have a private key to encrypt the signatures and a public key to
  decrypt them.

privateKey= <path>
* The path to the file containing the private key.
* Generate your own keys using openssl in $SPLUNK_HOME/bin/.
* If not present, a default key will be generated one time and placed at
  $SPLUNK_HOME/etc/auth/audit/private.pem

publicKey= <path>
* The path to the file containing the public key.
* Generate your own keys using openssl in $SPLUNK_HOME/bin/.
* If not present, a default key will be generated one time and placed at
  $SPLUNK_HOME/etc/auth/audit/public.pem

queueing=[true|false]
* Turn off sending audit events to the indexQueue -- tail the audit events
  instead.
* If this is set to 'false', you MUST add an inputs.conf stanza to tail the
  audit log in order to have the events reach your index.
* Defaults to true.
```

## audit.conf.example

```
# Version 6.5.0
#
# This is an example audit.conf. Use this file to configure auditing.
#
# There is NO DEFAULT audit.conf.
#
# To use one or more of these configurations, copy the configuration block into
# audit.conf in $SPLUNK_HOME/etc/system/local/. You must restart Splunk to
# enable configurations.
#
# To learn more about configuration files (including precedence) please see the
# documentation located at
# http://docs.splunk.com/Documentation/Splunk/latest/Admin/Aboutconfigurationfiles
```

```
[auditTrail]
privateKey=/some/path/to/your/private/key/private_key.pem
publicKey=/some/path/to/your/public/key/public_key.pem

# If this stanza exists, audit trail events will be cryptographically signed.
# You must have a private key to encrypt the signatures and a public key to decrypt them.
# Generate your own keys using openssl in $SPLUNK_HOME/bin/.
```

## authentication.conf

以下为 authentication.conf 的规范和示例文件。

### authentication.conf.spec

```
# Version 6.5.0
#
# This file contains possible attributes and values for configuring
# authentication via authentication.conf.
#
# There is an authentication.conf in $SPLUNK_HOME/etc/system/default/. To
# set custom configurations, place an authentication.conf in
# $SPLUNK_HOME/etc/system/local/. For examples, see
# authentication.conf.example. You must restart Splunk to enable
# configurations.
#
# To learn more about configuration files (including precedence) please see
# the documentation located at
# http://docs.splunk.com/Documentation/Splunk/latest/Admin/Aboutconfigurationfiles
```

### 全局设置

```
# GLOBAL SETTINGS
# Use the [default] stanza to define any global settings.
# * You can also define global settings outside of any stanza, at the top
#   of the file.
# * Each conf file should have at most one default stanza. If there are
#   multiple default stanzas, attributes are combined. In the case of
#   multiple definitions of the same attribute, the last definition in the
#   file wins.
# * If an attribute is defined at both the global level and in a specific
#   stanza, the value in the specific stanza takes precedence.

[authentication]
* Follow this stanza name with any number of the following attribute/value
  pairs.

authType = [Splunk|LDAP|Scripted|SAML|ProxySSO]
* Specify which authentication system to use.
* Supported values: Splunk, LDAP, Scripted, SAML, ProxySSO.
* Defaults to Splunk.

authSettings = <authSettings-key>,<authSettings-key>,...
* Key to look up the specific configurations of chosen authentication
  system.
* <authSettings-key> is the name of a stanza header that specifies
  attributes for scripted authentication, SAML, ProxySSO and for an LDAP
  strategy. Those stanzas are defined below.
* For LDAP, specify the LDAP strategy name(s) here. If you want Splunk to
  query multiple LDAP servers, enter a comma-separated list of all
  strategies. Each strategy must be defined in its own stanza. The order in
  which you specify the strategy names will be the order Splunk uses to
  query their servers when looking for a user.
* For scripted authentication, <authSettings-key> should be a single
  stanza name.

passwordHashAlgorithm = [SHA512-crypt|SHA256-crypt|SHA512-crypt-<num_rounds>|SHA256-crypt-<num_rounds>|MD5-crypt]
* For the default "Splunk" authType, this controls how hashed passwords are
```

stored in the `$SPLUNK_HOME/etc/passwd` file.

- \* "MD5-crypt" is an algorithm originally developed for FreeBSD in the early 1990's which became a widely used standard among UNIX machines. It was also used by Splunk up through the 5.0.x releases. MD5-crypt runs the salted password through a sequence of 1000 MD5 operations.
- \* "SHA256-crypt" and "SHA512-crypt" are newer versions that use 5000 rounds of the SHA256 or SHA512 hash functions. This is slower than MD5-crypt and therefore more resistant to dictionary attacks. SHA512-crypt is used for system passwords on many versions of Linux.
- \* These SHA-based algorithm can optionally be followed by a number of rounds to use. For example, "SHA512-crypt-10000" will use twice as many rounds of hashing as the default implementation. The number of rounds must be at least 1000.

If you specify a very large number of rounds (i.e. more than 20x the default value of 5000), splunkd may become unresponsive and connections to splunkd (from splunkweb or CLI) will time out.

- \* This setting only affects new password settings (either when a user is added or a user's password is changed). Existing passwords will continue to work but retain their previous hashing algorithm.
- \* The default is "SHA512-crypt".

`externalTwoFactorAuthVendor = <string>`

- \* OPTIONAL.
- \* A valid Multifactor vendor string will enable Multifactor authentication and loads support for the corresponding vendor if supported by Splunk.
- \* Empty string will disable Multifactor authentication in splunk.
- \* Currently splunk supports duo as a Multifactor authentication vendor.

`externalTwoFactorAuthSettings = <externalTwoFactorAuthSettings-key>`

- \* OPTIONAL.
- \* Key to look up the specific configuration of chosen Multifactor authentication vendor.

## LDAP 设置

```
#####
# LDAP settings
#####LDAP settings

[<authSettings-key>]
* Follow this stanza name with the attribute/value pairs listed below.
* For multiple strategies, you will need to specify multiple instances of
  this stanza, each with its own stanza name and a separate set of
  attributes.
* The <authSettings-key> must be one of the values listed in the
  authSettings attribute, specified above in the [authentication] stanza.
```

`host = <string>`

- \* REQUIRED
- \* This is the hostname of LDAP server.
- \* Be sure that your Splunk server can resolve the host name.

`SSLEnabled = [0|1]`

- \* OPTIONAL
- \* Defaults to disabled (0)
- \* See the file `$SPLUNK_HOME/etc/openldap/openldap.conf` for SSL LDAP settings

`port = <integer>`

- \* OPTIONAL
- \* This is the port that Splunk should use to connect to your LDAP server.
- \* Defaults to port 389 for non-SSL and port 636 for SSL

`bindDN = <string>`

- \* OPTIONAL, leave this blank to retrieve your LDAP entries using
 anonymous bind (must be supported by the LDAP server)
- \* Distinguished name of the user that will be retrieving the LDAP entries
- \* This user must have read access to all LDAP users and groups you wish to
 use in Splunk.

`bindDNpassword = <password>`

- \* OPTIONAL, leave this blank if anonymous bind is sufficient
- \* Password for the bindDN user.

userBaseDN = <string>

- \* REQUIRED
- \* This is the distinguished names of LDAP entries whose subtrees contain the users
- \* Enter a ';' delimited list to search multiple trees.

userBaseFilter = <string>

- \* OPTIONAL
- \* This is the LDAP search filter you wish to use when searching for users.
- \* Highly recommended, especially when there are many entries in your LDAP user subtrees
- \* When used properly, search filters can significantly speed up LDAP queries
- \* Example that matches users in the IT or HR department:
  - \* userBaseFilter = (|(department=IT)(department=HR))
  - \* See RFC 2254 for more detailed information on search filter syntax
- \* This defaults to no filtering.

userNameAttribute = <string>

- \* REQUIRED
- \* This is the user entry attribute whose value is the username.
- \* NOTE: This attribute should use case insensitive matching for its values, and the values should not contain whitespace
  - \* Usernames are case insensitive in Splunk
- \* In Active Directory, this is 'sAMAccountName'
- \* A typical attribute for this is 'uid'

realNameAttribute = <string>

- \* REQUIRED
- \* This is the user entry attribute whose value is their real name (human readable).
- \* A typical attribute for this is 'cn'

emailAttribute = <string>

- \* OPTIONAL
- \* This is the user entry attribute whose value is their email address.
- \* Defaults to 'mail'

groupMappingAttribute = <string>

- \* OPTIONAL
- \* This is the user entry attribute whose value is used by group entries to declare membership.
- \* Groups are often mapped with user DN, so this defaults to 'dn'
- \* Set this if groups are mapped using a different attribute
  - \* Usually only needed for OpenLDAP servers.
  - \* A typical attribute used to map users to groups is 'uid'
    - \* For example, assume a group declares that one of its members is 'splunkuser'
    - \* This implies that every user with 'uid' value 'splunkuser' will be mapped to that group

groupBaseDN = [<string>;<string>;...]

- \* REQUIRED
- \* This is the distinguished names of LDAP entries whose subtrees contain the groups.
- \* Enter a ';' delimited list to search multiple trees.
- \* If your LDAP environment does not have group entries, there is a configuration that can treat each user as its own group
  - \* Set groupBaseDN to the same as userBaseDN, which means you will search for groups in the same place as users
  - \* Next, set the groupMemberAttribute and groupMappingAttribute to the same attribute as userNameAttribute
    - \* This means the entry, when treated as a group, will use the username value as its only member
  - \* For clarity, you should probably also set groupNameAttribute to the same value as userNameAttribute as well

groupBaseFilter = <string>

- \* OPTIONAL
- \* The LDAP search filter Splunk uses when searching for static groups
- \* Like userBaseFilter, this is highly recommended to speed up LDAP queries

- \* See RFC 2254 for more information
- \* This defaults to no filtering

dynamicGroupFilter = <string>

- \* OPTIONAL
- \* The LDAP search filter Splunk uses when searching for dynamic groups
- \* Only configure this if you intend to retrieve dynamic groups on your LDAP server
- \* Example: '(objectclass=groupOfURLs)'

dynamicMemberAttribute = <string>

- \* OPTIONAL
- \* Only configure this if you intend to retrieve dynamic groups on your LDAP server
- \* This is REQUIRED if you want to retrieve dynamic groups
- \* This attribute contains the LDAP URL needed to retrieve members dynamically
- \* Example: 'memberURL'

groupNameAttribute = <string>

- \* REQUIRED
- \* This is the group entry attribute whose value stores the group name.
- \* A typical attribute for this is 'cn' (common name)
- \* Recall that if you are configuring LDAP to treat user entries as their own group, user entries must have this attribute

groupMemberAttribute = <string>

- \* REQUIRED
- \* This is the group entry attribute whose values are the groups members
- \* Typical attributes for this are 'member' and 'memberUid'
- \* For example, consider the groupMappingAttribute example above using groupMemberAttribute 'member'
- \* To declare 'splunkuser' as a group member, its attribute 'member' must have the value 'splunkuser'

nestedGroups = <bool>

- \* OPTIONAL
- \* Controls whether Splunk will expand nested groups using the 'memberof' extension.
- \* Set to 1 if you have nested groups you want to expand and the 'memberof' extension on your LDAP server.

charset = <string>

- \* OPTIONAL
- \* ONLY set this for an LDAP setup that returns non-UTF-8 encoded data. LDAP is supposed to always return UTF-8 encoded data (See RFC 2251), but some tools incorrectly return other encodings.
- \* Follows the same format as CHARSET in props.conf (see props.conf.spec)
- \* An example value would be "latin-1"

anonymous\_referrals = <bool>

- \* OPTIONAL
- \* Set this to 0 to turn off referral chasing
- \* Set this to 1 to turn on anonymous referral chasing
- \* IMPORTANT: We only chase referrals using anonymous bind. We do NOT support rebinding using credentials.
- \* If you do not need referral support, we recommend setting this to 0
- \* If you wish to make referrals work, set this to 1 and ensure your server allows anonymous searching
- \* Defaults to 1

sizelimit = <integer>

- \* OPTIONAL
- \* Limits the amount of entries we request in LDAP search
- \* IMPORTANT: The max entries returned is still subject to the maximum imposed by your LDAP server
- \* Example: If you set this to 5000 and the server limits it to 1000, you'll still only get 1000 entries back
- \* Defaults to 1000

timelimit = <integer>

- \* OPTIONAL
- \* Limits the amount of time in seconds we will wait for an LDAP search request to complete



- \* If your searches finish quickly, you should lower this value from the default
- \* Defaults to 15

network\_timeout = <integer>

- \* OPTIONAL
- \* Limits the amount of time a socket will poll a connection without activity
- \* This is useful for determining if your LDAP server cannot be reached
- \* IMPORTANT: As a connection could be waiting for search results, this value must be higher than 'timelimit'
- \* Like 'timelimit', if you have a fast connection to your LDAP server, we recommend lowering this value
- \* Defaults to 20

## 映射角色

```
#####
# Map roles
#####Map roles

[roleMap_<authSettings-key>]
* The mapping of Splunk roles to LDAP groups for the LDAP strategy specified
  by <authSettings-key>
* IMPORTANT: this role mapping ONLY applies to the specified strategy.
* Follow this stanza name with several Role-to-Group(s) mappings as defined
  below.
* Note: Importing groups for the same user from different strategies is not
  supported.

<Splunk RoleName> = <LDAP group string>
* Maps a Splunk role (from authorize.conf) to LDAP groups
* This LDAP group list is semicolon delimited (no spaces).
* List several of these attribute value pairs to map several Splunk roles to
  LDAP Groups
```

## 脚本式验证

```
#####
# Scripted authentication
#####Scripted authentication

[<authSettings-key>]
* Follow this stanza name with the following attribute/value pairs:

scriptPath = <string>
* REQUIRED
* This is the full path to the script, including the path to the program
  that runs it (python)
* For example: "$SPLUNK_HOME/bin/python" "$SPLUNK_HOME/etc/system/bin/$MY_SCRIPT"
* Note: If a path contains spaces, it must be quoted. The example above
  handles the case where SPLUNK_HOME contains a space

scriptSearchFilters = [1|0]
* OPTIONAL - Only set this to 1 to call the script to add search filters.
* 0 disables (default)

[cacheTiming]
* Use these settings to adjust how long Splunk will use the answers returned
  from script functions before calling them again.

userLoginTTL = <time range string>
* Timeout for the userLogin script function.
* These return values are cached on a per-user basis.
* The default is '0' (no caching)

getUserInfoTTL = <time range string>
* Timeout for the getUserInfo script function.
* These return values are cached on a per-user basis.
```

```

* The default is '10s'

getUsersTTL = <time range string>
* Timeout for the getUsers script function.
* There is only one global getUsers cache (it is not tied to a
  specific user).
* The default is '10s'

* All timeouts can be expressed in seconds or as a search-like time range
* Examples include '30' (30 seconds), '2mins' (2 minutes), '24h' (24 hours), etc.
* You can opt to use no caching for a particular function by setting the
  value to '0'
  * Be aware that this can severely hinder performance as a result of heavy
    script invocation
* Choosing the correct values for cache timing involves a tradeoff between
  new information latency and general performance
  * High values yield better performance from calling the script less, but
    introduces a latency in picking up changes
  * Low values will pick up changes in your external auth system more
    quickly, but may slow down performance due to increased script
    invocations

```

## Splunk 验证模式设置

```

#####
# Settings for Splunk Authentication mode
#####Settings for Splunk Authentication mode

[splunk_auth]
* Settings for Splunk's internal authentication system.

minPasswordLength = <positive integer>
* Specifies the minimum permitted password length in characters when
  passwords are set or modified.
* This setting is optional.
* If 0, there is no required minimum. In other words there is no constraint.
* Password modification attempts which do not meet this requirement will be
  explicitly rejected. Defaults to 0 (disabled).

```

## SAML 设置

```

#####
# SAML settings
#####SAML settings

[<saml-authSettings-key>]
* Follow this stanza name with the attribute/value pairs listed below.
* The <authSettings-key> must be one of the values listed in the
  authSettings attribute, specified above in the [authentication] stanza.

fqdn = <string>
* OPTIONAL
* The fully qualified domain name where this splunk instance is running.
* If this value is not specified, Splunk will default to the value specified
  in server.conf.
* If this value is specified and 'http://' or 'https://' prefix is not
  present, splunk will use the ssl setting for splunkweb.
* Splunk will use this information to populate the 'assertionConsumerServiceUrl'.

redirectPort = <port number>
* OPTIONAL
* The port where SAML responses will be sent. Typically, this is the
  web port.
* If internal port redirection is needed, set this port and the
  'assertionconsumerServiceUrl' in the AuthNRequest will contain this port
  instead of the splunkweb port.

```

\* To prevent any port information to be appended in the 'assertionConsumerServiceUrl' attribute, set this to 0.

idpSSOUrl = <url>

- \* REQUIRED
- \* The protocol endpoint on the IDP (Identity Provider) where the AuthNRequests should be sent.
- \* SAML requests will fail if this information is missing.

idpAttributeQueryUrl = <url>

- \* OPTIONAL
- \* The protocol endpoint on the IDP (Identity Provider) where the attribute query requests should be sent.
- \* Attribute queries can be used to get the latest 'role' information, if there is support for Attribute queries on the IDP.
- \* When this setting is absent, Splunk will cache the role information from the saml assertion and use it to run saved searches.

idpCertPath = <Pathname>

- \* OPTIONAL
- \* This setting is required if 'signedAssertion' is set to true.
- \* This value is relative to \$SPLUNK\_HOME/etc/auth/idpCerts.
- \* The value for this setting can be the name of the certificate file or a directory.
- \* If it is empty, Splunk will automatically verify with certificates in all subdirectories present in \$SPLUNK\_HOME/etc/auth/idpCerts.
- \* If the saml response is to be verified with a IDP (Identity Provider) certificate that is self signed, then this setting holds the filename of the certificate.
- \* If the saml response is to be verified with a certificate that is a part of a certificate chain(root, intermediate(s), leaf), create a subdirectory and place the certificate chain as files in the subdirectory.
- \* If there are multiple end certificates, create a subdirectory such that, one subdirectory holds one certificate chain.
- \* If multiple such certificate chains are present, the assertion is considered verified, if validation succeeds with any certificate chain.
- \* The file names within a certificate chain should be such that root certificate is alphabetically before the intermediate which is alphabetically before of the end cert.  
ex. cert\_1.pem has the root, cert\_2.pem has the first intermediate cert, cert\_3.pem has the second intermediate certificate and cert\_4.pem has the end certificate.

idpSLOUrl = <url>

- \* OPTIONAL
- \* The protocol endpoint on the IDP (Identity Provider) where a SP (Service Provider) initiated Single logout request should be sent.

errorUrl = <url>

- \* OPTIONAL
- \* The url to be displayed for a SAML error. Errors may be due to erroneous or incomplete configuration in either the IDP or Splunk. This url can be absolute or relative. Absolute url should follow pattern <protocol>://[<host> e.g. https://www.external-site.com. Relative urls should start with '/'. A relative url will show up as an internal link of the splunk instance, e.g. https://splunkhost:port/relativeUrlWithSlash

errorUrlLabel = <string>

- \* OPTIONAL
- \* Label or title of the content pointed to by errorUrl.

entityId = <string>

- \* REQUIRED
- \* The entity id for SP connection as configured on the IDP.

signAuthnRequest = [ true | false ]

- \* OPTIONAL
- \* This tells Splunk whether to sign AuthNRequests.
- \* Defaults to true.

signedAssertion = [true|false]

- \* OPTIONAL
- \* This tells Splunk if the SAML assertion has been signed by the IDP
- \* If set to false, Splunk will not verify the signature of the assertion using the certificate of the IDP.
- \* Currently, we accept only signed assertions.

- \* Defaults to true.

attributeQuerySoapPassword = <password>

- \* OPTIONAL
- \* This setting is required if 'attributeQueryUrl' is specified.
- \* Attribute query requests are made using SOAP using basic authentication
- \* The password to be used when making an attribute query request.
- \* This string will obfuscated upon splunkd startup.

attributeQuerySoapUsername = <string>

- \* OPTIONAL
- \* This setting is required if 'attributeQueryUrl' is specified.
- \* Attribute Query requests are made using SOAP using basic authentication
- \* The username to be used when making an attribute query request.

attributeQueryRequestSigned = [ true | false ]

- \* OPTIONAL
- \* Specifies whether to sign attribute query requests.
- \* Defaults to true

attributeQueryResponseSigned = [ true | false ]

- \* OPTIONAL
- \* Specifies whether attribute query responses are signed.
- \* If set to false, Splunk will not verify the signature in the response using the certificate of the IDP.
- \* Defaults to true.

redirectAfterLogoutToUrl = <url>

- \* OPTIONAL
- \* The user will be redirected to this url after logging out of Splunk.
- \* If this is not specified and a idpSLO is also missing, the user will be redirected to splunk.com after logout.

defaultRoleIfMissing = <splunk role>

- \* OPTIONAL
- \* If the IDP does not return any AD groups or splunk roles as a part of the assertion, we will use this value if provided.

skipAttributeQueryRequestForUsers = <comma separated list of users>

- \* OPTIONAL
- \* To skip attribute query requests being sent to the IDP for certain users, add them here.
- \* By default, attribute query requests will be skipped for local users.
- \* For non-local users, use this in conjunction with 'defaultRoleIfMissing'.

maxAttributeQueryThreads = <int>

- \* OPTIONAL
- \* Defaults to 2, max is 10
- \* Number of threads to use to make attribute query requests.
- \* Changes to this will require a restart to take effect.

maxAttributeQueryQueueSize = <int>

- \* OPTIONAL
- \* Defaults to 50
- \* The number of attribute query requests to queue, set to 0 for infinite size.
- \* Changes to this will require a restart to take effect.

attributeQueryTTL = <ttl in seconds>

- \* OPTIONAL
- \* Determines the time for which Splunk will cache the user and role information.
- \* Once the ttl expires, Splunk will make an attribute query request to retrieve the role information.
- \* Default ttl if not specified, is 3600 seconds.

allowSslCompression = [ true | false ]

- \* OPTIONAL
- \* If set to true, the server will allow clients to negotiate SSL-layer data compression.
- \* If not set, defaults to the setting in server.conf.

```

cipherSuite = <cipher suite string>
* OPTIONAL
* If set, Splunk uses the specified cipher string for the HTTP server.
* If not set, defaults to the setting in server.conf.
* Attribute query requests might fail if the IDP requires a relaxed
  ciphersuite.
* Use "openssl s_client -cipher 'TLSv1+HIGH:@STRENGTH' -host <IDP host> -port 443"
  to determine if splunk can connect to the IDP

sslVersions = <versions_list>
* OPTIONAL
* Comma-separated list of SSL versions to support.
* The versions available are "ssl3", "tls1.0", "tls1.1", and "tls1.2"
* If not set, defaults to the setting in server.conf.

sslCommonNameToCheck = <commonName>
* OPTIONAL
* If this value is set, and 'sslVerifyServerCert' is set to true,
  splunkd will limit most outbound HTTPS connections to hosts which use
  a cert with this common name.
* If not set, Splunk uses the setting specified in server.conf.

sslAltNameToCheck = <alternateName1>, <alternateName2>, ...
* OPTIONAL
* If this value is set, and 'sslVerifyServerCert' is set to true,
  splunkd will also be willing to verify certificates which have a so-called
  "Subject Alternate Name" that matches any of the alternate names in this
  list.
* If not set, Splunk uses the setting specified in server.conf.

ecdhCurveName = <string>
* DEPRECATED; use 'ecdhCurves' instead.
* ECDH curve to use for ECDH key negotiation.
* If not set, Splunk uses the setting specified in server.conf.

ecdhCurves = <comma separated list of ec curves>
* ECDH curves to use for ECDH key negotiation.
* The curves should be specified in the order of preference.
* The client sends these curves as a part of Client Hello.
* The server supports only the curves specified in the list.
* We only support named curves specified by their SHORT names.
  (see struct ASN1_OBJECT in asn1.h)
* The list of valid named curves by their short/long names can be obtained
  by executing this command:
  $SPLUNK_HOME/bin/splunk cmd openssl ecparam -list_curves
* Default is empty string.
* e.g. ecdhCurves = prime256v1,secp384r1,secp521r1
* If not set, Splunk uses the setting specified in server.conf.

clientCert = <path>
* Full path to the client certificate PEM format file.
* Certificates are auto-generated upon first starting Splunk.
* You may replace the auto-generated certificate with your own.
* Default is $SPLUNK_HOME/etc/auth/server.pem.
* If not set, Splunk uses the setting specified in
  server.conf/[sslConfig]/serverCert.

sslKeysfile = <filename>
* DEPRECATED; use 'clientCert' instead.
* File is in the directory specified by 'caPath' (see below).
* Default is server.pem.

sslPassword = <password>
* Optional server certificate password.
* If unset, Splunk uses the setting specified in server.conf.
* Default is password.

sslKeysfilePassword = <password>
* DEPRECATED; use 'sslPassword' instead.

caCertFile = <filename>
* OPTIONAL

```

- \* Public key of the signing authority.
- \* Default is cacert.pem.
- \* If not set, Splunk uses the setting specified in server.conf.

caPath = <path>

- \* DEPRECATED; use absolute paths for all certificate files.
- \* If certificate files given by other settings in this stanza are not absolute paths, then they will be relative to this path.
- \* Default is \$SPLUNK\_HOME/etc/auth.

sslVerifyServerCert = <bool>

- \* OPTIONAL
- \* Used by distributed search: when making a search request to another server in the search cluster.
- \* If not set, Splunk uses the setting specified in server.conf.

blacklistedAutoMappedRoles = <comma separated list of roles>

- \* OPTIONAL
- \* Comma separated list of splunk roles that should be blacklisted from being auto-mapped by splunk from the IDP Response.

blacklistedUsers = <comma separated list of user names>

- \* OPTIONAL
- \* Comma separated list of user names from the IDP response to be blacklisted by splunk platform.

nameIdFormat = <string>

- \* OPTIONAL
- \* If supported by IDP, while making SAML Authentication request this value can be used to specify the format of the Subject returned in SAML Assertion.

ssoBinding = <string>

- \* OPTIONAL
- \* This is the binding that will be used when making a SP-initiated saml request.
- \* Acceptable options are 'HTTPPost' and 'HTTPRedirect'
- \* Defaults to 'HTTPPost'
- \* This binding must match the one configured on the IDP.

sloBinding = <string>

- \* OPTIONAL
- \* This is the binding that will be used when making a logout request or sending a logout response to complete the logout workflow.
- \* Acceptable options are 'HTTPPost' and 'HTTPRedirect'
- \* Defaults to 'HTTPPost'
- \* This binding must match the one configured on the IDP.

signatureAlgorithm = RSA-SHA1 | RSA-SHA256

- \* OPTIONAL
- \* Defaults to RSA-SHA1.
- \* This setting is applicable only for redirect binding.
- \* RSA-SHA1 corresponds to 'http://www.w3.org/2000/09/xmldsig#rsa-sha1'.
- \* RSA-SHA256 corresponds to 'http://www.w3.org/2001/04/xmldsig-more#rsa-sha256'.
- \* Specifies the signature algorithm that will be used for a SP-initiated saml request, when 'signedAuthnRequest' is set to true.
- \* This will be sent as a part of 'sigAlg'.

## 映射角色

```
#####
# Map roles
#####Map roles

[roleMap_<saml-authSettings-key>]
* The mapping of Splunk roles to SAML groups for the SAML stanza specified by <authSettings-key>
* If a SAML group is not explicitly mapped to a Splunk role, but has same name as a valid Splunk role then for ease of configuration, it is auto-mapped to that Splunk role.
* Follow this stanza name with several Role-to-Group(s) mappings as defined
```

below.

```
<Splunk RoleName> = <SAML group string>
* Maps a Splunk role (from authorize.conf) to SAML groups
* This SAML group list is semicolon delimited (no spaces).
* List several of these attribute value pairs to map several Splunk roles to SAML Groups.
* If role mapping is not specified, Splunk expects Splunk roles in the assertion and attribute query response returned from the IDP.
```

## **SAML 用户角色映射**

```
#####
# SAML User Roles Map
#####SAML User Roles Map

[userToRoleMap_<saml-authSettings-key>]
* The mapping of SAML user to Splunk roles for the SAML stanza specified by <authSettings-key>
* Follow this stanza name with several User-to-Role(s) mappings as defined below.
* The stanza is used only when the IDP does not support Attribute Query Request

<SAML User> = <Splunk Roles string>
* Maps a SAML user to Splunk role (from authorize.conf)
* This Splunk Role list is semicolon delimited (no spaces).
```

## **验证响应属性映射**

```
#####
# Authentication Response Attribute Map
#####Authentication Response Attribute Map

[authenticationResponseAttrMap_SAML]
* Splunk expects email, real name and roles to be returned as SAML Attributes in SAML assertion. This stanza can be used to map attribute names to what Splunk expects. These are optional settings and are only needed for certain IDPs.

role = <string>
* OPTIONAL
* Attribute name to be used as role in SAML Assertion.
* Default is "role"

realName = <string>
* OPTIONAL
* Attribute name to be used as realName in SAML Assertion.
* Default is "realName"

mail = <string>
* OPTIONAL
* Attribute name to be used as email in SAML Assertion.
* Default is "mail"
```

## **代理 SSO 模式设置**

```
#####
# Settings for Proxy SSO mode
#####Settings for Proxy SSO mode

[roleMap_proxySSO]

* The mapping of Splunk roles to groups passed in headers from proxy server.
* If a group is not explicitly mapped to a Splunk role, but has same name as a valid Splunk role then for ease of configuration, it is
```

auto-mapped to that Splunk role.

- \* Follow this stanza name with several Role-to-Group(s) mappings as defined below.

```
<Splunk RoleName> = <Group string>
```

- \* Maps a Splunk role (from authorize.conf) to groups
- \* This group list is semicolon delimited (no spaces).
- \* List several of these attribute value pairs to map several Splunk roles to Groups
- \* If role mapping is not specified, user is logged in with default User role.

```
[userToRoleMap_proxySSO]
```

- \* The mapping of ProxySSO user to Splunk roles
- \* Follow this stanza name with several User-to-Role(s) mappings as defined below.

```
<ProxySSO User> = <Splunk Roles string>
```

- \* Maps a ProxySSO user to Splunk role (from authorize.conf)
- \* This Splunk Role list is semicolon delimited (no spaces).

```
[proxysso-authsettings-key]
```

- \* Follow this stanza name with the attribute/value pairs listed below.

```
defaultRoleIfMissing = <splunk role>
```

- \* OPTIONAL
- \* If splunk roles cannot be determined based on role mapping, use default configured
- \* splunk role.

```
blacklistedAutoMappedRoles = <comma separated list of roles>
```

- \* OPTIONAL
- \* Comma separated list of splunk roles that should be blacklisted from being auto-mapped by splunk from the proxy server headers.

```
blacklistedUsers = <comma separated list of user names>
```

- \* OPTIONAL
- \* Comma separated list of user names from the proxy server headers to be blacklisted by splunk platform.

## 密钥存储

```
#####
# Secret Storage
#####Secret Storage
```

```
[secrets]
```

```
disabled = <bool>
```

- \* Toggles integration with platform-provided secret storage facilities.
- \* Defaults to false if Common Criteria mode is enabled.
- \* Defaults to true if Common Criteria mode is disabled.
- \* NOTE: Splunk plans to submit Splunk Enterprise for Common Criteria evaluation. Splunk does not support using the product in Common Criteria mode until it has been certified by NIAP. See the "Securing Splunk Enterprise" manual for information on the status of Common Criteria certification.

```
filename = <filename>
```

- \* Designates a Python script that integrates with platform-provided secret storage facilities, like the GNOME keyring.
- \* <filename> should be the name of a Python script located in one of the following directories:
  - \$SPLUNK\_HOME/etc/apps/\*/bin
  - \$SPLUNK\_HOME/etc/system/bin
  - \$SPLUNK\_HOME/etc/searchscripts
- \* <filename> should be a pure basename; it should contain no path separators.
- \* <filename> should end with a .py file extension.

```
namespace = <string>
```

- \* Use an instance-specific string as a namespace within secret storage.
- \* When using the GNOME keyring, this namespace is used as a keyring name.



- \* If multiple Splunk instances must store separate sets of secrets within the same storage backend, this value should be customized to be unique for each Splunk instance.
- \* Defaults to "splunk".

## Duo MFA 供应商设置

```
#####
# Duo MFA vendor settings
#####Duo MFA vendor settings
[<duo-externalTwoFactorAuthSettings-key>]
* <duo-externalTwoFactorAuthSettings-key> must be the value listed in the
  externalTwoFactorAuthSettings attribute, specified above in the [authentication]
  stanza.
* This stanza contains Duo specific Multifactor authentication settings and will be
  activated only when externalTwoFactorAuthVendor is Duo.
* All the below attributes except appSecretKey would be provided by Duo.

apiHostname = <string>
* REQUIRED
* Duo's API endpoint which performs the actual Multifactor authentication.
* e.g. apiHostname = api-xyz.duosecurity.com

integrationKey = <string>
* REQUIRED
* Duo's integration key for splunk. Must be of size = 20.
* Integration key will be obfuscated before being saved here for security.

secretKey = <string>
* REQUIRED
* Duo's secret key for splunk. Must be of size = 40.
* Secret key will be obfuscated before being saved here for security.

appSecretKey = <string>
* REQUIRED
* Splunk application specific secret key which should be random and locally generated.
* Must be atleast of size = 40 or longer.
* This secret key would not be shared with Duo.
* Application secret key will be obfuscated before being saved here for security.

failOpen = <bool>
* OPTIONAL
* Defaults to false if not set.
* If set to true, Splunk will bypass Duo Multifactor Authentication when the service is
  unavailable.

timeout = <int>
* OPTIONAL
* It determines the connection timeout in seconds for the outbound duo HTTPS connection.
* If not set, Splunk will use its default HTTPS connection timeout which is 12 seconds.

sslVersions = <versions_list>
* OPTIONAL
* Comma-separated list of SSL versions to support for incoming connections.
* The versions available are "ssl3", "tls1.0", "tls1.1", and "tls1.2".
* If not set, Splunk uses the sslVersions provided in server.conf

cipherSuite = <cipher suite string>
* OPTIONAL
* If set, Splunk uses the specified cipher string for the HTTP server.
* If not set, Splunk uses the cipher string provided in server.conf

ecdhCurves = <comma separated list of ec curves>
* OPTIONAL
* ECDH curves to use for ECDH key negotiation.
* If not set, Splunk uses the ecdh curve names provided in server.conf

sslVerifyServerCert = <bool>
* OPTIONAL
* Defaults to false if not set.
```

- \* If this is set to true, you should make sure that the server that is being connected to is a valid one (authenticated). Both the common name and the alternate name of the server are then checked for a match if they are specified in this configuration file. A certificate is considered verified if either is matched.

```
sslCommonNameToCheck = <commonName1>, <commonName2>, ...
```

- \* OPTIONAL
- \* Not set by default.
- \* If this value is set, Splunk will limit outbound duo HTTPS connections to host which use a cert with one of the listed common names.
- \* sslVerifyServerCert must be set to true for this setting to work.

```
sslAltNameToCheck = <alternateName1>, <alternateName2>, ...
```

- \* OPTIONAL
- \* Not set by default.
- \* If this value is set, Splunk will limit outbound duo HTTPS connections to host which use a cert with one of the listed alternate names.
- \* sslVerifyServerCert must be set to true for this setting to work.

```
sslRootCAPath = <path>
```

- \* OPTIONAL
- \* Not set by default.
- \* The <path> must refer to full path of a PEM format file containing one or more root CA certificates concatenated together.
- \* This Root CA must match the CA in the certificate chain of the SSL certificate returned by duo server.

```
useClientSSLCompression = <bool>
```

- \* OPTIONAL
- \* If set to true on client side, compression is enabled between the server and client as long as the server also supports it.
- \* If not set, Splunk uses the client SSL compression setting provided in server.conf

## authentication.conf.example

```
# Version 6.5.0
#
# This is an example authentication.conf. authentication.conf is used to
# configure LDAP, Scripted, SAML and Proxy SSO authentication in addition
# to Splunk's native authentication.
#
# To use one of these configurations, copy the configuration block into
# authentication.conf in $SPLUNK_HOME/etc/system/local/. You must reload
# auth in manager or restart Splunk to enable configurations.
#
# To learn more about configuration files (including precedence) please see
# the documentation located at
# http://docs.splunk.com/Documentation/Splunk/latest/Admin/Aboutconfigurationfiles

##### Use just Splunk's built-in authentication (default):
[authentication]
authType = Splunk

##### LDAP examples

#### Basic LDAP configuration example
[authentication]
authType = LDAP
authSettings = ldaphost

[ldaphost]
host = ldaphost.domain.com
port = 389
SSEnabled = 0
bindDN = cn=Directory Manager
bindDNpassword = password
userBaseDN = ou=People,dc=splunk,dc=com
```

```

userBaseFilter = (objectclass=splunkusers)
groupBaseDN = ou=Groups,dc=splunk,dc=com
groupBaseFilter = (objectclass=splunkgroups)
userNameAttribute = uid
realNameAttribute = givenName
groupMappingAttribute = dn
groupMemberAttribute = uniqueMember
groupNameAttribute = cn
timelimit = 10
network_timeout = 15

# This stanza maps roles you have created in authorize.conf to LDAP Groups
[roleMap_ldaphost]
admin = SplunkAdmins

#### Example using the same server as 'ldaphost', but treating each user as
#### their own group
[authentication]
authType = LDAP
authSettings = ldaphost_usergroups

[ldaphost_usergroups]
host = ldaphost.domain.com
port = 389
SSLEnabled = 0
bindDN = cn=Directory Manager
bindDNpassword = password
userBaseDN = ou=People,dc=splunk,dc=com
userBaseFilter = (objectclass=splunkusers)
groupBaseDN = ou=People,dc=splunk,dc=com
groupBaseFilter = (objectclass=splunkusers)
userNameAttribute = uid
realNameAttribute = givenName
groupMappingAttribute = uid
groupMemberAttribute = uid
groupNameAttribute = uid
timelimit = 10
network_timeout = 15

[roleMap_ldaphost_usergroups]
admin = admin_user1;admin_user2;admin_user3;admin_user4
power = power_user1;power_user2
user = user1;user2;user3

#### Sample Configuration for Active Directory (AD)
[authentication]
authSettings = AD
authType = LDAP

[AD]
SSLEnabled = 1
bindDN = ldap_bind@splunksupport.kom
bindDNpassword = ldap_bind_user_password
groupBaseDN = CN=Groups,DC=splunksupport,DC=com
groupBaseFilter =
groupMappingAttribute = dn
groupMemberAttribute = member
groupNameAttribute = cn
host = ADbogus.splunksupport.kom
port = 636
realNameAttribute = cn
userBaseDN = CN=Users,DC=splunksupport,DC=com
userBaseFilter =
userNameAttribute = sAMAccountName
timelimit = 15
network_timeout = 20
anonymous_referrals = 0

[roleMap_AD]
admin = SplunkAdmins
power = SplunkPowerUsers
user = SplunkUsers

```

```

#### Sample Configuration for Sun LDAP Server
[authentication]
authSettings = SunLDAP
authType = LDAP

[SunLDAP]
SSLEnabled = 0
bindDN = cn=Directory Manager
bindDNpassword = Directory_Manager_Password
groupBaseDN = ou=Groups,dc=splunksupport,dc=com
groupBaseFilter =
groupMappingAttribute = dn
groupMemberAttribute = uniqueMember
groupNameAttribute = cn
host = ldapbogus.splunksupport.com
port = 389
realNameAttribute = givenName
userBaseDN = ou=People,dc=splunksupport,dc=com
userBaseFilter =
userNameAttribute = uid
timelimit = 5
network_timeout = 8

[roleMap_SunLDAP]
admin = SplunkAdmins
power = SplunkPowerUsers
user = SplunkUsers

#### Sample Configuration for OpenLDAP
[authentication]
authSettings = OpenLDAP
authType = LDAP

[OpenLDAP]
bindDN = uid=directory_bind,cn=users,dc=osx,dc=company,dc=com
bindDNpassword = directory_bind_account_password
groupBaseFilter =
groupNameAttribute = cn
SSLEnabled = 0
port = 389
userBaseDN = cn=users,dc=osx,dc=company,dc=com
host = hostname_OR_IP
userBaseFilter =
userNameAttribute = uid
groupMappingAttribute = uid
groupBaseDN = dc=osx,dc=company,dc=com
groupMemberAttribute = memberUid
realNameAttribute = cn
timelimit = 5
network_timeout = 8
dynamicGroupFilter = (objectclass=groupOfURLs)
dynamicMemberAttribute = memberURL
nestedGroups = 1

[roleMap_OpenLDAP]
admin = SplunkAdmins
power = SplunkPowerUsers
user = SplunkUsers

##### Scripted Auth examples

#### The following example is for RADIUS authentication:
[authentication]
authType = Scripted
authSettings = script

[script]
scriptPath = "$SPLUNK_HOME/bin/python" "$SPLUNK_HOME/share/splunk/authScriptSamples/radiusScripted.py"

# Cache results for 1 second per call

```

```

[cacheTiming]
userLoginTTL      = 1
getUserInfoTTL    = 1
getUsersTTL       = 1

#### The following example works with PAM authentication:
[authentication]
authType = Scripted
authSettings = script

[script]
scriptPath = "$SPLUNK_HOME/bin/python" "$SPLUNK_HOME/share/splunk/authScriptSamples/pamScripted.py"

# Cache results for different times per function
[cacheTiming]
userLoginTTL      = 30s
getUserInfoTTL    = 1min
getUsersTTL       = 5mins

##### SAML auth example

[authentication]
authSettings = samlv2
authType = SAML

[samlv2]
attributeQuerySoapPassword = changeme
attributeQuerySoapUsername = test
entityId = test-splunk
idpAttributeQueryUrl = https://exsso/idp/attrsvc.ssaml2
idpCertPath = /home/splunk/etc/auth/idp.crt
idpSSOUrl = https://exsso/idp/SSO.saml2
idpSLOUrl = https://exsso/idp/SLO.saml2
signAuthnRequest = true
signedAssertion = true
attributeQueryRequestSigned = true
attributeQueryResponseSigned = true
redirectPort = 9332
cipherSuite = TLSv1 MEDIUM:@STRENGTH
nameIdFormat = urn:oasis:names:tc:SAML:1.1:nameid-format:emailAddress

[roleMap_SAML]
admin = SplunkAdmins
power = SplunkPowerUsers
user = all

[userToRoleMap_SAML]
samluser = user

[authenticationResponseAttrMap_SAML]
role = "http://schemas.microsoft.com/ws/2008/06/identity/claims/groups"
mail = "http://schemas.xmlsoap.org/ws/2005/05/identity/claims/emailaddress"
realName = "http://schemas.microsoft.com/identity/claims/displayname"

# Multifactor authentication example
[authentication]
externalTwoFactorAuthVendor = duo
externalTwoFactorAuthSettings = duo-mfa

# Duo specific authentication setting example
[duo-mfa]
apiHostname = api-xyz.duosecurity.com
appSecretKey = mustBeARandomStringOfSize40OrLonger
integrationKey = mustBeADuoProvidedStringOfSize20
secretKey = mustBeADuoProvidedStringOfSize40

##### Proxy SSO auth example

[authentication]
authSettings = my_proxy

```

```

authType = ProxySSO

[my_proxy]
blacklistedUsers = user1,user2
blacklistedAutoMappedRoles = admin
defaultRoleIfMissing = user

[roleMap_proxySSO]
admin = group1;group2
user = group1;group3

[userToRoleMap_proxySSO]
proxy_user1 = user
proxy_user2 = power;can_delete

```

## authorize.conf

以下为 authorize.conf 的规范和示例文件。

### authorize.conf.spec

```

# Version 6.5.0
#
# This file contains possible attribute/value pairs for creating roles in
# authorize.conf. You can configure roles and granular access controls by
# creating your own authorize.conf.
#
# There is an authorize.conf in $SPLUNK_HOME/etc/system/default/. To set
# custom configurations, place an authorize.conf in
# $SPLUNK_HOME/etc/system/local/. For examples, see authorize.conf.example.
# You must restart Splunk to enable configurations.
#
# To learn more about configuration files (including precedence) please see
# the documentation located at
# http://docs.splunk.com/Documentation/Splunk/latest/Admin/Aboutconfigurationfiles

```

### 全局设置

```

# GLOBAL SETTINGS
# Use the [default] stanza to define any global settings.
# * You can also define global settings outside of any stanza, at the top
#   of the file.
# * Each conf file should have at most one default stanza. If there are
#   multiple default stanzas, attributes are combined. In the case of
#   multiple definitions of the same attribute, the last definition in
#   the file wins.
# * If an attribute is defined at both the global level and in a specific
#   stanza, the value in the specific stanza takes precedence.

```

### [default]

```

[default]
srchFilterSelecting = <boolean>
* Determine's whether roles' search filters will be used for selecting or
  eliminating during role inheritance.
* Selecting will join the search filters with an OR when combining the
  filters.
* Eliminating will join the search filters with an AND when combining the
  filters.
  * All roles will default to true (in other words, selecting).
* Example:
  * role1 srchFilter = sourcetype!=ex1 with selecting=true
  * role2 srchFilter = sourcetype=ex2 with selecting = false
  * role3 srchFilter = sourcetype!=ex3 AND index=main with selecting = true
  * role3 inherits from role2 and role 2 inherits from role1
  * Resulting srchFilter = ((sourcetype!=ex1) OR (sourcetype!=ex3 AND index=main)) AND ((sourcetype=ex2))

```

## **[capability::<capability>]**

```
[capability::<capability>]
```

- \* DO NOT edit, remove, or add capability stanzas. The existing capabilities are the full set of Splunk system capabilities.
- \* Splunk adds all of its capabilities this way
- \* For the default list of capabilities and assignments, see `authorize.conf` under the 'default' directory
- \* Descriptions of specific capabilities are listed below.

## **[role\_<roleName>]**

```
[role_<roleName>]
```

<capability> = <enabled>

- \* A capability that is enabled for this role.
- \* You can list many of these.
- \* Note that 'enabled' is the only accepted value here, as capabilities are disabled by default.
- \* Roles inherit all capabilities from imported roles, and inherited capabilities cannot be disabled.
- \* Role names cannot have uppercase characters. User names, however, are case-insensitive.

importRoles = <string>

- \* Semicolon delimited list of other roles and their associated capabilities that should be imported.
- \* Importing other roles also imports the other aspects of that role, such as allowed indexes to search.
- \* By default a role imports no other roles.

grantableRoles = <string>

- \* Semicolon delimited list of roles that can be granted when `edit_user` capability is present.
- \* By default, a role with `edit_user` capability can create/edit a user and assign any role to them. But when `grantableRoles` is present, the roles that can be assigned will be restricted to the ones provided.
- \* For a role that has no `edit_user` capability, `grantableRoles` has no effect.
- \* Defaults to not present.
- \* Example: `grantableRoles = role1;role2;role3`

srchFilter = <string>

- \* Semicolon delimited list of search filters for this Role.
- \* By default we perform no search filtering.
- \* To override any search filters from imported roles, set this to '\*', as the 'admin' role does.

srchTimeWin = <number>

- \* Maximum time span of a search, in seconds.
  - \* This time window limit is applied backwards from the latest time specified in a search.
- \* By default, searches are not limited to any specific time window.
- \* To override any search time windows from imported roles, set this to '0' (infinite), as the 'admin' role does.
- \* -1 is a special value that implies no search window has been set for this role
  - \* This is equivalent to not setting `srchTimeWin` at all, which means it can be easily overridden by an imported role

srchDiskQuota = <number>

- \* Maximum amount of disk space (MB) that can be used by search jobs of a user that belongs to this role
- \* Defaults to '100', for 100 MB.

srchJobsQuota = <number>

- \* Maximum number of concurrently running historical searches a member of this role can have.
- \* This excludes real-time searches, see `rtSrchJobsQuota`.
- \* Defaults to 3.

```

rtSrchJobsQuota = <number>
* Maximum number of concurrently running real-time searches a member of this
  role can have.
* Defaults to 6.

srchMaxTime = <number><unit>
* Maximum amount of time that searches of users from this role will be
  allowed to run.
* Once the search has been ran for this amount of time it will be auto
  finalized, If the role
* Inherits from other roles, the maximum srchMaxTime value specified in the
  included roles.
* This maximum does not apply to real-time searches.
* Examples: 1h, 10m, 2hours, 2h, 2hrs, 100s
* Defaults to 100days

srchIndexesDefault = <string>
* Semicolon delimited list of indexes to search when no index is specified
* These indexes can be wildcarded, with the exception that '*' does not
  match internal indexes
* To match internal indexes, start with '_'. All internal indexes are
  represented by '_'
* Defaults to none, but the UI will automatically populate this with 'main'
  in manager

srchIndexesAllowed = <string>
* Semicolon delimited list of indexes this role is allowed to search
* Follows the same wildcarding semantics as srchIndexesDefault
* Defaults to none, but the UI will automatically populate this with '*' in
  manager

deleteIndexesAllowed = <string>
* Semicolon delimited list of indexes this role is allowed to delete
* This setting must be used in conjunction with the delete_by_keyword
  capability
* Follows the same wildcarding semantics as srchIndexesDefault
* Defaults to none

cumulativeSrchJobsQuota = <number>
* Maximum number of concurrently running historical searches in total
  across all members of this role
* Requires enable_cumulative_quota = true in limits.conf to take effect.
* If a user belongs to multiple roles, the user's searches count against the role with
  the largest cumulative search quota. Once the quota for that role is consumed, the
  user's searches count against the role with the next largest quota, and so on.
* In search head clustering environments, this setting takes effect on a per-member basis.
  There is no cluster-wide accounting.

cumulativeRTSrchJobsQuota = <number>
* Maximum number of concurrently running real-time searches in total
  across all members of this role
* Requires enable_cumulative_quota = true in limits.conf to take effect.
* If a user belongs to multiple roles, the user's searches count against the role with
  the largest cumulative search quota. Once the quota for that role is consumed, the
  user's searches count against the role with the next largest quota, and so on.
* In search head clustering environments, this setting takes effect on a per-member basis.
  There is no cluster-wide accounting.

### Descriptions of Splunk system capabilities

```

### **[capability::accelerate\_datamodel]**

```

[capability::accelerate_datamodel]
* Required to accelerate a datamodel.

```

### **[capability::admin\_all\_objects]**

```

[capability::admin_all_objects]
* A role with this capability has access to objects in the system (user
  objects, search jobs, etc.)

```



- \* This bypasses any ACL restrictions (similar to root access in a \*nix environment)
- \* We check this capability when accessing manager pages and objects

### **[capability::change\_authentication]**

[capability::change\_authentication]

- \* Required to change authentication settings through the various authentication endpoints.
- \* Also controls whether authentication can be reloaded

### **[capability::change\_own\_password]**

[capability::change\_own\_password]

- \* Self explanatory. Some auth systems prefer to have passwords be immutable for some users.

### **[capability::list\_storage\_passwords]**

[capability::list\_storage\_passwords]

- \* Controls access to the /storage/passwords endpoint. Users with this capability can perform GETs. Note that the admin\_all\_objects capability is required to perform POSTs to the /storage/passwords endpoint.

### **[capability::delete\_by\_keyword]**

[capability::delete\_by\_keyword]

- \* Required to use the 'delete' search operator. Note that this does not actually delete the raw data on disk.
- \* Delete merely masks the data (via the index) from showing up in search results.

### **[capability::edit\_deployment\_client]**

[capability::edit\_deployment\_client]

- \* Self explanatory. The deployment client admin endpoint requires this cap for edit.

### **[capability::list\_deployment\_client]**

[capability::list\_deployment\_client]

- \* Self explanatory.

### **[capability::edit\_deployment\_server]**

[capability::edit\_deployment\_server]

- \* Self explanatory. The deployment server admin endpoint requires this cap for edit.
- \* Required to change/create remote inputs that get pushed to the forwarders.

### **[capability::list\_deployment\_server]**

[capability::list\_deployment\_server]

- \* Self explanatory.

### **[capability::edit\_dist\_peer]**

[capability::edit\_dist\_peer]

- \* Required to add and edit peers for distributed search.

### **[capability::edit\_forwarders]**

[capability::edit\_forwarders]  
\* Required to edit settings for forwarding data.  
\* Used by TCP and Syslog output admin handlers  
\* Includes settings for SSL, backoff schemes, etc.

### **[capability::edit\_httpauths]**

[capability::edit\_httpauths]  
\* Required to edit and end user sessions through the httpauth-tokens endpoint

### **[capability::edit\_indexer\_cluster]**

[capability::edit\_indexer\_cluster]  
\* Required to edit or manage indexer cluster.

### **[capability::edit\_input\_defaults]**

[capability::edit\_input\_defaults]  
\* Required to change the default hostname for input data in the server settings endpoint.

### **[capability::edit\_monitor]**

[capability::edit\_monitor]  
\* Required to add inputs and edit settings for monitoring files.  
\* Used by the standard inputs endpoint as well as the one-shot input endpoint.

### **[capability::edit\_modinput\_winhostmon]**

[capability::edit\_modinput\_winhostmon]  
\* Required to add and edit inputs for monitoring Windows host data.

### **[capability::edit\_modinput\_winnetmon]**

[capability::edit\_modinput\_winnetmon]  
\* Required to add and edit inputs for monitoring Windows network data.

### **[capability::edit\_modinput\_winprintmon]**

[capability::edit\_modinput\_winprintmon]  
\* Required to add and edit inputs for monitoring Windows printer data.

### **[capability::edit\_modinput\_perfmon]**

[capability::edit\_modinput\_perfmon]  
\* Required to add and edit inputs for monitoring Windows performance.

### **[capability::edit\_modinput\_admon]**

[capability::edit\_modinput\_admon]  
\* Required to add and edit inputs for monitoring Splunk's Active Directory.

### **[capability::edit\_roles]**

[capability::edit\_roles]  
\* Required to edit roles as well as change the mappings from users to roles.  
\* Used by both the users and roles endpoint.

### **[capability::edit\_roles\_grantable]**

[capability::edit\_roles\_grantable]

- \* Restrictive version of the edit\_roles capability. Only allows creation of roles with subset of the capabilities that the current user has as part of its grantable\_roles. only works in conjunction with edit\_user and grantableRoles

### **[capability::edit\_scripted]**

[capability::edit\_scripted]

- \* Required to create and edit scripted inputs.

### **[capability::edit\_search\_server]**

[capability::edit\_search\_server]

- \* Required to edit general distributed search settings like timeouts, heartbeats, and blacklists

### **[capability::list\_introspection]**

[capability::list\_introspection]

- \* Required to read introspection settings and statistics for indexers, search, processors, queues, etc.
- \* Does not permit editing introspection settings.

### **[capability::list\_settings]**

[capability::list\_settings]

- \* Required to list general server and introspection settings such as the server name, log levels, etc.

### **[capability::edit\_server]**

[capability::edit\_server]

- \* Required to edit general server and introspection settings such as the server name, log levels, etc.
- \* Inherits ability to read general server and introspection settings.

### **[capability::edit\_search\_head\_clustering]**

[capability::edit\_search\_head\_clustering]

- \* Required to edit and manage search head clustering.

### **[capability::edit\_search\_scheduler]**

[capability::edit\_search\_scheduler]

- \* Required to disable/enable the search scheduler.

### **[capability::edit\_search\_schedule\_priority]**

[capability::edit\_search\_schedule\_priority]

- \* Required to give a search a higher-than-normal schedule priority.

### **[capability::edit\_search\_schedule\_window]**

[capability::edit\_search\_schedule\_window]

- \* Required to give a search a non-automatic (or no) schedule window.

### **[capability::list\_search\_scheduler]**

[capability::list\_search\_scheduler]  
\* Required to display search scheduler settings.

### **[capability::edit\_sourcetypes]**

[capability::edit\_sourcetypes]  
\* Required to create and edit sourcetypes.

### **[capability::edit\_splunktcp]**

[capability::edit\_splunktcp]  
\* Required to change settings for receiving TCP input from another Splunk instance.

### **[capability::edit\_splunktcp\_ssl]**

[capability::edit\_splunktcp\_ssl]  
\* Required to list or edit any SSL specific settings for Splunk TCP input.

### **[capability::edit\_splunktcp\_token]**

[capability::edit\_splunktcp\_token]  
\* Required to list or edit splunktcp\_tokens which can be used on a receiving system to only accept data from forwarders that have been configured with same token.

### **[capability::edit\_tcp]**

[capability::edit\_tcp]  
\* Required to change settings for receiving general TCP inputs.

### **[capability::edit\_udp]**

[capability::edit\_udp]  
\* Required to change settings for UDP inputs.

### **[capability::edit\_telemetry\_settings]**

[capability::edit\_telemetry\_settings]  
\* Required to change settings to opt-in and send telemetry data.

### **[capability::edit\_token\_http]**

[capability::edit\_token\_http]  
\* Required to create, edit, display and remove settings for HTTP token input.

### **[capability::edit\_user]**

[capability::edit\_user]  
\* Required to create, edit, or remove users.  
\* Note that Splunk users may edit certain aspects of their information without this capability.  
\* Also required to manage certificates for distributed search.

### **[capability::edit\_view\_html]**

[capability::edit\_view\_html]  
\* Required to create, edit, or otherwise modify HTML-based views.

### **[capability::edit\_web\_settings]**

[capability::edit\_web\_settings]  
\* Required to change the settings for web.conf through the system settings endpoint.

### **[capability::get\_diag]**

[capability::get\_diag]  
\* Required to use the /streams/diag endpoint to get remote diag from an instance

### **[capability::get\_metadata]**

[capability::get\_metadata]  
\* Required to use the 'metadata' search processor.

### **[capability::get\_typeahead]**

[capability::get\_typeahead]  
\* Required for typeahead. This includes the typeahead endpoint and the 'typeahead' search processor.

### **[capability::input\_file]**

[capability::input\_file]  
\* Required for inputcsv (except for dispatch=t mode) and inputlookup

### **[capability::indexes\_edit]**

[capability::indexes\_edit]  
\* Required to change any index settings like file size and memory limits.

### **[capability::license\_tab]**

[capability::license\_tab]  
\* Required to access and change the license. (Deprecated)

### **[capability::license\_edit]**

[capability::license\_edit]  
\* Required to access and change the license.

### **[capability::license\_view\_warnings]**

[capability::license\_view\_warnings]  
\* Required to view license warnings on the system banner

### **[capability::list\_forwarders]**

[capability::list\_forwarders]  
\* Required to show settings for forwarding data.  
\* Used by TCP and Syslog output admin handlers.

### **[capability::list\_httpauths]**

[capability::list\_httpauths]  
\* Required to list user sessions through the httpauth-tokens endpoint.

### **[capability::list\_indexer\_cluster]**

[capability::list\_indexer\_cluster]  
\* Required to list indexer cluster objects like buckets, peers etc.

### **[capability::list\_inputs]**

[capability::list\_inputs]  
\* Required to view the list of various inputs.  
\* This includes input from files, TCP, UDP, Scripts, etc.

### **[capability::list\_search\_head\_clustering]**

[capability::list\_search\_head\_clustering]  
\* Required to list search head clustering objects like artifacts, delegated jobs, members, captain, etc.

### **[capability::output\_file]**

[capability::output\_file]  
\* Required for outputcsv (except for dispatch=t mode) and outputlookup

### **[capability::request\_remote\_tok]**

[capability::request\_remote\_tok]  
\* Required to get a remote authentication token.  
\* Used for distributing search to old 4.0.x Splunk instances.  
\* Also used for some distributed peer management and bundle replication.

### **[capability::rest\_apps\_management]**

[capability::rest\_apps\_management]  
\* Required to edit settings for entries and categories in the python remote apps handler.  
\* See restmap.conf for more information

### **[capability::rest\_apps\_view]**

[capability::rest\_apps\_view]  
\* Required to list various properties in the python remote apps handler.  
\* See restmap.conf for more info

### **[capability::rest\_properties\_get]**

[capability::rest\_properties\_get]  
\* Required to get information from the services/properties endpoint.

### **[capability::rest\_properties\_set]**

[capability::rest\_properties\_set]  
\* Required to edit the services/properties endpoint.

### **[capability::restart\_splunkd]**

[capability::restart\_splunkd]  
\* Required to restart Splunk through the server control handler.

### **[capability::rtsearch]**

[capability::rtsearch]  
\* Required to run a realtime search.

### **[capability::run\_debug\_commands]**

[capability::run\_debug\_commands]  
\* Required to run debugging commands like 'summarize'

### **[capability::schedule\_search]**

[capability::schedule\_search]  
\* Required to schedule saved searches.

### **[capability::schedule\_rtsearch]**

[capability::schedule\_rtsearch]  
\* Required to schedule real time saved searches. Note that scheduled\_search capability is also required to be enabled

### **[capability::search]**

[capability::search]  
\* Self explanatory - required to run a search.

### **[capability::use\_file\_operator]**

[capability::use\_file\_operator]  
\* Required to use the 'file' search operator.

### **[capability::accelerate\_search]**

[capability::accelerate\_search]  
\* Required to save an accelerated search  
\* All users have this capability by default

### **[capability::web\_debug]**

[capability::web\_debug]  
\* Required to access /\_bump and /debug/\*\* web debug endpoints

### **[capability::edit\_server\_crl]**

[capability::edit\_server\_crl]  
\* Required to reload CRL information within Splunk

### **[capability::search\_process\_config\_refresh]**

[capability::search\_process\_config\_refresh]  
\* Required to use the "refresh search-process-config" CLI command, which manually flushes idle search processes.

### **[capability::extra\_x509\_validation]**

[capability::extra\_x509\_validation]  
\* Required to perform additional X509 validation through the /server/security/extra-x509-validation.

## **authorize.conf.example**

```
# Version 6.5.0
#
# This is an example authorize.conf. Use this file to configure roles and
# capabilities.
#
# To use one or more of these configurations, copy the configuration block
# into authorize.conf in $SPLUNK_HOME/etc/system/local/. You must reload
# auth or restart Splunk to enable configurations.
#
# To learn more about configuration files (including precedence) please see
# the documentation located at
# http://docs.splunk.com/Documentation/Splunk/latest/Admin/Aboutconfigurationfiles

[role_ninja]
rtsearch = enabled
importRoles = user
srchFilter = host=foo
srchIndexesAllowed = *
srchIndexesDefault = mail;main
srchJobsQuota = 8
rtSrchJobsQuota = 8
srchDiskQuota = 500

# This creates the role 'ninja', which inherits capabilities from the 'user'
# role. ninja has almost the same capabilities as power, except cannot
# schedule searches.
#
# The search filter limits ninja to searching on host=foo.
#
# ninja is allowed to search all public indexes (those that do not start
# with underscore), and will search the indexes mail and main if no index is
# specified in the search.
#
# ninja is allowed to run 8 search jobs and 8 real time search jobs
# concurrently (these counts are independent).
#
# ninja is allowed to take up 500 megabytes total on disk for all their jobs.
```

## collections.conf

以下为 collections.conf 的规范和示例文件。

### collections.conf.spec

```
# Version 6.5.0
#
# This file configures the KV Store collections for a given app in Splunk.
#
# To learn more about configuration files (including precedence) please see
# the documentation located at
# http://docs.splunk.com/Documentation/Splunk/latest/Admin/Aboutconfigurationfiles
```

#### [<collection-name>]

```
[<collection-name>]
enforceTypes = true|false
* Indicates whether to enforce data types when inserting data into the
  collection.
* When set to true, invalid insert operations fail.
* When set to false, invalid insert operations drop only the invalid field.
* Defaults to false.

field.<name> = number|bool|string|time
* Field type for a field called <name>.
* If the data type is not provided, it is inferred from the provided JSON
  data type.
```



```

accelerated_fields.<name> = <json>
* Acceleration definition for an acceleration called <name>.
* Must be a valid JSON document (invalid JSON is ignored).
* Example: 'acceleration.foo={"a":1, "b":-1}' is a compound acceleration
  that first sorts 'a' in ascending order and then 'b' in descending order.
* If multiple accelerations with the same definition are in the same
  collection, the duplicates are skipped.
* If the data within a field is too large for acceleration, you will see a
  warning when you try to create an accelerated field and the acceleration
  will not be created.
* An acceleration is always created on the _key.
* The order of accelerations is important. For example, an acceleration of
  { "a":1, "b":1 } speeds queries on "a" and "a" + "b", but not on "b"
  alone.
* Multiple separate accelerations also speed up queries. For example,
  separate accelerations { "a": 1 } and { "b": 1 } will speed up queries on
  "a" + "b", but not as well as a combined acceleration { "a":1, "b":1 }.
* Defaults to nothing (no acceleration).

profilingEnabled = true|false
* Indicates whether to enable logging of slow-running operations, as defined
  in 'profilingThresholdMs'.
* Defaults to false.

profilingThresholdMs = <zero or positive integer>
* The threshold for logging a slow-running operation, in milliseconds.
* When set to 0, all operations are logged.
* This setting is only used when 'profilingEnabled' is true.
* This setting impacts the performance of the collection.
* Defaults to 1000.

replicate = true|false
* Indicates whether to replicate this collection on indexers. When false,
  this collection is not replicated, and lookups that depend on this
  collection will not be available (although if you run a lookup command
  with 'local=true', local lookups will still be available). When true,
  this collection is replicated on indexers.
* Defaults to false.

replication_dump_strategy = one_file|auto
* Indicates how to store dump files. When set to one_file, dump files are
  stored in a single file. When set to auto, dumps are stored in multiple
  files when the size of the collection exceeds the value of
  'replication_dump_maximum_file_size'.
* Defaults to auto.

replication_dump_maximum_file_size = <unsigned integer>
* Specifies the maximum file size (in KB) for each dump file when
  'replication_dump_strategy=auto'.
* If this value is larger than 'concerningReplicatedFileSize', which is set
  in distsearch.conf, the value of 'concerningReplicatedFileSize' will be
  used instead.
* KV Store does not pre-calculate the size of the records that will be written
  to disk, so the size of the resulting files can be affected by the
  'max_rows_in_memory_per_dump' setting from 'limits.conf'.
* Defaults to 10240KB.

type = internal_cache|undefined
* Indicates the type of data that this collection holds.
* When set to 'internal_cache', changing the configuration of the current
  instance between search head cluster, search head pool, or standalone
  will erase the data in the collection.
* Defaults to 'undefined'.
* For internal use only.

```

## collections.conf.example

```

# Version 6.5.0
#

```

```
# The following is an example collections.conf configuration.
#
# To use one or more of these configurations, copy the configuration block
# into collections.conf in $SPLUNK_HOME/etc/system/local/. You must restart
# Splunk to enable configurations.
#
# To learn more about configuration files (including precedence) please see
# the documentation located at
# http://docs.splunk.com/Documentation/Splunk/latest/Admin/Aboutconfigurationfiles

[mycollection]

field.foo = number
field.bar = string
accelerated_fields.myacceleration = {"foo": 1, "bar": -1}
```

## commands.conf

以下为 commands.conf 的规范和示例文件。

### commands.conf.spec

```
# Version 6.5.0
#
# This file contains possible attribute/value pairs for creating search
# commands for any custom search scripts created. Add your custom search
# script to $SPLUNK_HOME/etc/searchscripts/ or
# $SPLUNK_HOME/etc/apps/MY_APP/bin/. For the latter, put a custom
# commands.conf in $SPLUNK_HOME/etc/apps/MY_APP. For the former, put your
# custom commands.conf in $SPLUNK_HOME/etc/system/local/.

# There is a commands.conf in $SPLUNK_HOME/etc/system/default/. For examples,
# see commands.conf.example. You must restart Splunk to enable configurations.

# To learn more about configuration files (including precedence) please see the
# documentation located at
# http://docs.splunk.com/Documentation/Splunk/latest/Admin/Aboutconfigurationfiles
```

### 全局设置

```
# GLOBAL SETTINGS
# Use the [default] stanza to define any global settings.
# * You can also define global settings outside of any stanza, at the top of
# the file.
# * Each conf file should have at most one default stanza. If there are
# multiple default stanzas, attributes are combined. In the case of
# multiple definitions of the same attribute, the last definition in the
# file wins.
# * If an attribute is defined at both the global level and in a specific
# stanza, the value in the specific stanza takes precedence.
```

### [<STANZA\_NAME>]

```
[<STANZA_NAME>]
* Each stanza represents a search command; the command is the stanza name.
* The stanza name invokes the command in the search language.
* Set the following attributes/values for the command. Otherwise, Splunk uses
the defaults.
* If the filename attribute is not specified, Splunk searches for an
external program by appending extensions (e.g. ".py", ".pl") to the
stanza name.
* If chunked = true, in addition to ".py" and ".pl" as above, Splunk
searches using the extensions ".exe", ".bat", ".cmd", ".sh", ".js",
and no extension (to find extensionless binaries).
* See the filename attribute for more information about how Splunk
searches for external programs.
```

```

type = <string>
* Type of script: python, perl
* Defaults to python.

filename = <string>
* Optionally specify the program to be executed when the search command is used.
* Splunk looks for the given filename in the app's bin directory.
* The filename attribute can not reference any file outside of the app's bin directory.
* If the filename ends in ".py", Splunk's python interpreter is used
  to invoke the external script.
* If chunked = true, Splunk looks for the given filename in
  $SPLUNK_HOME/etc/apps/MY_APP/<PLATFORM>/bin before searching
  $SPLUNK_HOME/etc/apps/MY_APP/bin, where <PLATFORM> is one of
  "linux_x86_64", "linux_x86", "windows_x86_64", "windows_x86",
  "darwin_x86_64" (depending on the platform on which Splunk is
  running on).
* If chunked = true and if a path pointer file (*.path) is specified,
  the contents of the file are read and the result is used as the
  command to be run. Environment variables in the path pointer
  file are substituted. Path pointer files can be used to reference
  system binaries (e.g. /usr/bin/python).

command.arg.<N> = <string>
* Additional command-line arguments to use when invoking this
  program. Environment variables will be substituted (e.g. $SPLUNK_HOME).
* Only available if chunked = true.

local = [true|false]
* If true, specifies that the command should be run on the search head only
* Defaults to false

perf_warn_limit = <integer>
* Issue a performance warning message if more than this many input events are
  passed to this external command (0 = never)
* Defaults to 0 (disabled)

streaming = [true|false]
* Specify whether the command is streamable.
* Defaults to false.

maxinputs = <integer>
* Maximum number of events that can be passed to the command for each
  invocation.
* This limit cannot exceed the value of maxresultrows in limits.conf.
* 0 for no limit.
* Defaults to 50000.

passauth = [true|false]
* If set to true, splunkd passes several authentication-related facts
  at the start of input, as part of the header (see enableheader).
* The following headers are sent
  * authString: pseudo-xml string that resembles
    <auth><userId>username</userId><username>username</username><authToken>auth_token</authToken></auth>
    where the username is passed twice, and the authToken may be used
    to contact splunkd during the script run.
  * sessionKey: the session key again.
  * owner: the user portion of the search context
  * namespace: the app portion of the search context
* Requires enableheader = true; if enableheader = false, this flag will
  be treated as false as well.
* Defaults to false.
* If chunked = true, this attribute is ignored. An authentication
  token is always passed to commands using the chunked custom search
  command protocol.

run_in_preview = [true|false]
* Specify whether to run this command if generating results just for preview
  rather than final output.
* Defaults to true

enableheader = [true|false]

```

- \* Indicate whether or not your script is expecting header information or not.
- \* Currently, the only thing in the header information is an auth token.
- \* If set to true it will expect as input a head section + '\n' then the csv input
- \* NOTE: Should be set to true if you use splunk.Intersplunk
- \* Defaults to true.

retainsevents = [true|false]

- \* Specify whether the command retains events (the way the sort/dedup/cluster commands do) or whether it transforms them (the way the stats command does).
- \* Defaults to false.

generating = [true|false]

- \* Specify whether your command generates new events. If no events are passed to the command, will it generate events?
- \* Defaults to false.

generates\_timeorder = [true|false]

- \* If generating = true, does command generate events in descending time order (latest first)
- \* Defaults to false.

overrides\_timeorder = [true|false]

- \* If generating = false and streaming=true, does command change the order of events with respect to time?
- \* Defaults to false.

requires\_preop = [true|false]

- \* Specify whether the command sequence specified by the 'streaming\_preop' key is required for proper execution or is it an optimization only
- \* Default is false (streaming\_preop not required)

streaming\_preop = <string>

- \* A string that denotes the requested pre-streaming search string.

required\_fields = <string>

- \* A comma separated list of fields that this command may use.
- \* Informs previous commands that they should retain/extract these fields if possible. No error is generated if a field specified is missing.
- \* Defaults to '\*'

supports\_multivalues = [true|false]

- \* Specify whether the command supports multivalues.
- \* If true, multivalues will be treated as python lists of strings, instead of a flat string (when using Intersplunk to interpret stdin/stdout).
- \* If the list only contains one element, the value of that element will be returned, rather than a list (for example, isinstance(val, basestring) == True).

supports\_getinfo = [true|false]

- \* Specifies whether the command supports dynamic probing for settings (first argument invoked == \_\_GETINFO\_\_ or \_\_EXECUTE\_\_).

supports\_rawargs = [true|false]

- \* Specifies whether the command supports raw arguments being passed to it or if it prefers parsed arguments (where quotes are stripped).
- \* If unspecified, the default is false

undo\_scheduler\_escaping = [true|false]

- \* Specifies whether the commands raw arguments need to be unescaped.
- \* This is particularly applies to the commands being invoked by the scheduler.
- \* This applies only if the command supports raw arguments(supports\_rawargs).
- \* If unspecified, the default is false

requires\_srinfo = [true|false]

- \* Specifies if the command requires information stored in SearchResultsInfo.
- \* If true, requires that enableheader be set to true, and the full pathname of the info file (a csv file) will be emitted in the header under the key 'infoPath'
- \* If unspecified, the default is false

needs\_empty\_results = [true|false]

- \* Specifies whether or not this search command needs to be called with intermediate empty search results
- \* If unspecified, the default is true

changes\_colorder = [true|false]

- \* Specify whether the script output should be used to change the column ordering of the fields.
- \* Default is true

outputheader = <true/false>

- \* If set to true, output of script should be a header section + blank line + csv output
- \* If false, script output should be pure csv only
- \* Default is false

clear\_required\_fields = [true|false]

- \* If true, required\_fields represents the \*only\* fields required.
- \* If false, required\_fields are additive to any fields that may be required by subsequent commands.
- \* In most cases, false is appropriate for streaming commands and true for reporting commands
- \* Default is false

stderr\_dest = [log|message|none]

- \* What do to with the stderr output from the script
- \* 'log' means to write the output to the job's search.log.
- \* 'message' means to write each line as an search info message. The message level can be set to adding that level (in ALL CAPS) to the start of the line, e.g. "WARN my warning message."
- \* 'none' means to discard the stderr output
- \* Defaults to log

is\_order\_sensitive = [true|false]

- \* Specify whether the command requires ordered input.
- \* Defaults to false.

is\_risky = [true|false]

- \* Searches using Splunk Web are flagged to warn users when they unknowingly run a search that contains commands that might be a security risk. This warning appears when users click a link or type a URL that loads a search that contains risky commands. This warning does not appear when users create ad hoc searches.
- \* This flag is used to determine whether the command is risky.
- \* Defaults to false.
- \* - Specific commands that ship with the product have their own defaults

chunked = [true|false]

- \* If true, this command supports the new "chunked" custom search command protocol.
- \* If true, the only other commands.conf attributes supported are is\_risky, maxwait, maxchunksize, filename, and command.arg.<N>.
- \* If false, this command uses the legacy custom search command protocol supported by Intersplunk.py.
- \* Default is false

maxwait = <integer>

- \* Only available if chunked = true.
- \* Not supported in Windows.
- \* The value of maxwait is the maximum number of seconds the custom search command can pause before producing output.
- \* If set to 0, the command can pause forever.
- \* Default is 0

maxchunksize = <integer>

- \* Only available if chunked = true.
- \* The value of maxchunksize is maximum size chunk (size of metadata plus size of body) the external command may produce. If the command tries to produce a larger chunk, the command is terminated.
- \* If set to 0, the command may send any size chunk.
- \* Default is 0

## commands.conf.example

```
# Version 6.5.0
#
# Configuration for external search commands
#

#####
# defaults for all external commands, exceptions are below in individual
# stanzas

# type of script: 'python', 'perl'
TYPE = python
# default FILENAME would be <stanza-name>.py for python, <stanza-name>.pl for
# perl and <stanza-name> otherwise

# is command streamable?
STREAMING = false

# maximum data that can be passed to command (0 = no limit)
MAXINPUTS = 50000

# end defaults
#####

[crawl]
FILENAME = crawl.py

[createrss]
FILENAME = creators.py

[diff]
FILENAME = diff.py

[gentimes]
FILENAME = gentimes.py

[head]
FILENAME = head.py

[loglady]
FILENAME = loglady.py

[marklar]
FILENAME = marklar.py

[runshellscript]
FILENAME = runshellscript.py

[sendemail]
FILENAME = sendemail.py

[translate]
FILENAME = translate.py

[transpose]
FILENAME = transpose.py

[uniq]
FILENAME = uniq.py

[windbag]
filename = windbag.py
supports_multivalues = true

[xmlkv]
FILENAME = xmlkv.py

[xmlunescape]
FILENAME = xmlunescape.py
```

## crawl.conf

以下为 crawl.conf 的规范和示例文件。

### crawl.conf.spec

```
# Version 6.5.0
#
# This file contains possible attribute/value pairs for configuring crawl.
#
# There is a crawl.conf in $SPLUNK_HOME/etc/system/default/. To set custom
# configurations, place a crawl.conf in $SPLUNK_HOME/etc/system/local/. For
# help, see crawl.conf.example. You must restart Splunk to enable
# configurations.
#
# To learn more about configuration files (including precedence) please see the
# documentation located at
# http://docs.splunk.com/Documentation/Splunk/latest/Admin/Aboutconfigurationfiles
#
# Set of attribute-values used by crawl.
#
# If attribute, ends in _list, the form is:
#
#     attr = val, val, val, etc.
#
# The space after the comma is necessary, so that "," can be used, as in
# BAD_FILE_PATTERNS's use of "*",v"
```

### 全局设置

```
# GLOBAL SETTINGS
# Use the [default] stanza to define any global settings.
# * You can also define global settings outside of any stanza, at the top of
#   the file.
# * Each conf file should have at most one default stanza. If there are
#   multiple default stanzas, attributes are combined. In the case of
#   multiple definitions of the same attribute, the last definition in the
#   file wins.
# * If an attribute is defined at both the global level and in a specific
#   stanza, the value in the specific stanza takes precedence.
```

### [default]

```
[default]
```

### [files]

```
[files]
* Sets file crawler-specific attributes under this stanza header.
* Follow this stanza name with any of the following attributes.

root = <semi-colon separate list of directories>
* Set a list of directories this crawler should search through.
* Defaults to /;/Library/Logs

bad_directories_list = <comma-separated list of bad directories>
* List any directories you don't want to crawl.
* Defaults to:
    bin, sbin, boot, mnt, proc, tmp, temp, dev, initrd, help, driver, drivers, share, bak, old, lib, include,
    doc, docs, man, html, images, tests, js, dtd, org, com, net, class, java, resource, locale, static, testing, src,
    sys, icons, css, dist, cache, users, system, resources, examples, gdm, manual, spool, lock, kerberos, .thumbnails,
    libs, old, manuals, splunk, splunkpreview, mail, resources, documentation, applications, library, network,
    automount, mount, cores, lost\+found, fonts, extensions, components, printers, caches, findlogs, music, volumes,
    libexec
```

```

bad_extensions_list = <comma-separated list of file extensions to skip>
* List any file extensions and crawl will skip files that end in those extensions.
* Defaults to:
    Ot, a, adb, ads, ali, am, asa, asm, asp, au, bak, bas, bat, bmp, c, cache, cc, cg, cgi, class, clp, com,
    conf, config, cpp, cs, css, csv, cxx, dat, doc, dot, dvi, dylib, ec, elc, eps, exe, f, f77, f90, for, ftn, gif, h,
    hh, hlp, hpp, hqx, hs, htm, html, hxx, icns, ico, ics, in, inc, jar, java, jin, jpeg, jpg, js, jsp, kml, la, lai,
    lhs, lib, license, lo, m, m4, mcp, mid, mp3, mpg, msf, nib, nsmmap, o, obj, odt, ogg, old, ook, opt, os, os2, pal,
    pbm, pdf, pdf, pem, pgm, php, php3, php4, pl, plex, plist, plo, plx, pm, png, po, pod, ppd, ppm, ppt, prc, presets,
    ps, psd, psym, py, pyc, pyd, pyw, rast, rb, rc, rde, rdf, rdr, res, rgb, ro, rsrc, s, sgml, sh, shtml, so, soap,
    sql, ss, stg, strings, tcl, tdt, template, tif, tiff, tk, uue, v, vhd, wsd1, xbm, xlb, xls, xlw, xml, xsd, xsl,
    xslt, jame, d, ac, properties, pid, del, lock, md5, rpm, pp, deb, iso, vim, lng, list

bad_file_matches_list = <comma-separated list of regex>
* Crawl applies the specified regex and skips files that match the patterns.
* There is an implied "$" (end of file name) after each pattern.
* Defaults to:
    *~, *#, *,v, *readme*, *install, (/|^).*, *passwd*, *example*, *makefile, core.*

packed_extensions_list = <comma-separated list of extensions>
* Specify extensions of compressed files to exclude.
* Defaults to:
    bz, bz2, tbz, tbz2, Z, gz, tgz, tar, zip

collapse_threshold = <integer>
* Specify the minimum number of files a source must have to be considered a
    directory.
* Defaults to 1000.

days_sizek_pairs_list = <comma-separated hyphenated pairs of integers>
* Specify a comma-separated list of age (days) and size (kb) pairs to constrain
    what files are crawled.
* For example: days_sizek_pairs_list = 7-0, 30-1000 tells Splunk to crawl only
    files last modified within 7 days and at least 0kb in size, or modified
    within the last 30 days and at least 1000kb in size.
* Defaults to 30-0.

big_dir_filecount = <integer>
* Skip directories with files above <integer>
* Defaults to 10000.

index = <$INDEX>
* Specify index to add crawled files to.
* Defaults to main.

max_badfiles_per_dir = <integer>
* Specify how far to crawl into a directory for files.
* Crawl excludes a directory if it doesn't find valid files within the
    specified max_badfiles_per_dir.
* Defaults to 100.

```

## **[network]**

```

[network]
* Sets network crawler-specific attributes under this stanza header.
* Follow this stanza name with any of the following attributes.

host = <host or ip>
* default host to use as a starting point for crawling a network
* Defaults to 'localhost'.

subnet = <int>
* default number of bits to use in the subnet mask. Given a host with IP
    123.123.123.123, a subnet value of 32, would scan only that host, and a value
    or 24 would scan 123.123.123.*.
* Defaults to 32.

```

## **crawl.conf.example**

```
# Version 6.5.0
```



```
#
# The following are example crawl.conf configurations. Configure properties for
# crawl.
#
# To use one or more of these configurations, copy the configuration block into
# crawl.conf in $SPLUNK_HOME/etc/system/local/. You must restart Splunk to
# enable configurations.
#
# To learn more about configuration files (including precedence) please see the
# documentation located at
# http://docs.splunk.com/Documentation/Splunk/latest/Admin/Aboutconfigurationfiles

[files]
bad_directories_list= bin, sbin, boot, mnt, proc, tmp, temp, home, mail, .thumbnails, cache, old
bad_extensions_list= mp3, mpg, jpeg, jpg, m4, mcp, mid
bad_file_matches_list= *example*, *makefile, core.*
packed_extensions_list= gz, tgz, tar, zip
collapse_threshold= 10
days_sizek_pairs_list= 3-0,7-1000, 30-10000
big_dir_filecount= 100
index=main
max_badfiles_per_dir=100

[network]
host = myserver
subnet = 24
```

## datamodels.conf

以下为 datamodels.conf 的规范和示例文件。

### datamodels.conf.spec

```
# Version 6.5.0
#
# This file contains possible attribute/value pairs for configuring
# data models. To configure a datamodel for an app, put your custom
# datamodels.conf in $SPLUNK_HOME/etc/apps/MY_APP/local/
#
# For examples, see datamodels.conf.example. You must restart Splunk to
# enable configurations.
#
# To learn more about configuration files (including precedence) please see
# the documentation located at
# http://docs.splunk.com/Documentation/Splunk/latest/Admin/Aboutconfigurationfiles
```

### 全局设置

```
# GLOBAL SETTINGS
# Use the [default] stanza to define any global settings.
# * You can also define global settings outside of any stanza, at the top
# of the file.
# * Each conf file should have at most one default stanza. If there are
# multiple default stanzas, attributes are combined. In the case of
# multiple definitions of the same attribute, the last definition in the
# file wins.
# * If an attribute is defined at both the global level and in a specific
# stanza, the value in the specific stanza takes precedence.
```

### [<datamodel\_name>]

```
[<datamodel_name>]
* Each stanza represents a data model. The data model name is the stanza name.
```

```

acceleration = <bool>
* Set acceleration to true to enable automatic acceleration of this data model.
* Automatic acceleration creates auxiliary column stores for the fields
  and values in the events for this datamodel on a per-bucket basis.
* These column stores take additional space on disk, so be sure you have the
  proper amount of disk space. Additional space required depends on the
  number of events, fields, and distinct field values in the data.
* The Splunk software creates and maintains these column stores on a schedule
  you can specify with 'acceleration.cron_schedule.' You can query
  them with the 'tstats' command.

acceleration.earliest_time = <relative-time-str>
* Specifies how far back in time the Splunk software should keep these column
  stores (and create if acceleration.backfill_time is not set).
* Specified by a relative time string. For example, '-7d' means 'accelerate
  data within the last 7 days.'
* Defaults to an empty string, meaning 'keep these stores for all time.'

acceleration.backfill_time = <relative-time-str>
* ADVANCED: Specifies how far back in time the Splunk software should create
  its column stores.
* ONLY set this parameter if you want to backfill less data than the
  retention period set by 'acceleration.earliest_time'. You may want to use
  this parameter to limit your time window for column store creation in a large
  environment where initial creation of a large set of column stores is an
  expensive operation.
* WARNING: Do not set 'acceleration.backfill_time' to a
  narrow time window. If one of your indexers is down for a period longer
  than this backfill time, you may miss accelerating a window of your incoming
  data.
* MUST be set to a more recent time than 'acceleration.earliest_time'. For
  example, if you set 'acceleration.earliest_time' to '-1y' to retain your
  column stores for a one year window, you could set 'acceleration.backfill_time'
  to '-20d' to create column stores that only cover the last 20 days. However,
  you cannot set 'acceleration.backfill_time' to '-2y', because that goes
  farther back in time than the 'acceleration.earliest_time' setting of '-1y'.
* Defaults to empty string (unset). When 'acceleration.backfill_time' is unset,
  the Splunk software always backfills fully to 'acceleration.earliest_time.'

acceleration.max_time = <unsigned int>
* The maximum amount of time that the column store creation search is
  allowed to run (in seconds).
* Note that this is an approximate time, as the 'summarize' search only
  finishes on clean bucket boundaries to avoid wasted work.
* Defaults to: 3600
* An 'acceleration.max_time' setting of '0' indicates that there is no time
  limit.

acceleration.cron_schedule = <cron-string>
* Cron schedule to be used to probe/generate the column stores for this
  data model.
* Defaults to: */5 * * * *

acceleration.manual_rebuilds = <bool>
* ADVANCED: When set to 'true,' this setting prevents outdated summaries from
  being rebuilt by the 'summarize' command.
* Normally, during the creation phase, the 'summarize' command automatically
  rebuilds summaries that are considered to be out-of-date, such as when the
  configuration backing the data model changes.
* The Splunk software considers a summary to be outdated when:
  * The data model search stored in its metadata no longer matches its current
    data model search.
  * The search stored in its metadata cannot be parsed.
  * A lookup table associated with the data model is altered.
* NOTE: If the Splunk software finds a partial summary be outdated, it always
  rebuilds that summary so that a bucket summary only has results corresponding to
  one datamodel search.
* Defaults to: false

acceleration.max_concurrent = <unsigned int>
* The maximum number of concurrent acceleration instances for this data
  model that the scheduler is allowed to run.

```

```

* Defaults to: 2

acceleration.schedule_priority = default | higher | highest
* Raises the scheduling priority of a search:
+ "default": No scheduling priority increase.
+ "higher": Scheduling priority is higher than other data model searches.
+ "highest": Scheduling priority is higher than other searches regardless of
  scheduling tier except real-time-scheduled searches with priority = highest
  always have priority over all other searches.
+ Hence, the high-to-low order (where RTSS = real-time-scheduled search, CSS
  = continuous-scheduled search, DMAS = data-model-accelerated search, d =
  default, h = higher, H = highest) is:
    RTSS(H) > DMAS(H) > CSS(H)
    > RTSS(h) > RTSS(d) > CSS(h) > CSS(d)
    > DMAS(h) > DMAS(d)
* The scheduler honors a non-default priority only when the search owner has
  the 'edit_search_schedule_priority' capability.
* Defaults to: default
* WARNING: Having too many searches with a non-default priority will impede the
  ability of the scheduler to minimize search starvation. Use this setting
  only for mission-critical searches.

acceleration.hunk.compression_codec = <string>
* Applicable only to Hunk Data models. Specifies the compression codec to
  be used for the accelerated orc/parquet files.

acceleration.hunk.dfs_block_size = <unsigned int>
* Applicable only to Hunk data models. Specifies the block size in bytes for
  the compression files.

acceleration.hunk.file_format = <string>
* Applicable only to Hunk data models. Valid options are "orc" and "parquet"

#***** Dataset Related Attributes *****
# These attributes affect your interactions with datasets in Splunk Web and should
# not be changed under normal conditions. Do not modify them unless you are sure you
# know what you are doing.

dataset.description = <string>
* User-entered description of the dataset entity.

dataset.type = [datamodel|table]
* The type of dataset:
+ "datamodel": An individual data model dataset.
+ "table": A special root data model dataset with a search where the dataset is
  defined by the dataset.commands attribute.
* Default: datamodel

dataset.commands = [<object>(<object>)*]
* When the dataset.type = "table" this stringified JSON payload is created by the
  table editor and defines the dataset.

dataset.fields = [<string>(<string>)*]
* Automatically generated JSON payload when dataset.type = "table" and the root
  data model dataset's search is updated.

dataset.display.diversity = [latest|random|diverse|rare]
* The user-selected diversity for previewing events contained by the dataset:
+ "latest": search a subset of the latest events
+ "random": search a random sampling of events
+ "diverse": search a diverse sampling of events
+ "rare": search a rare sampling of events based on clustering
* Default: latest

dataset.display.sample_ratio = <int>
* The integer value used to calculate the sample ratio for the dataset diversity.
  The formula is 1 / <int>.
* The sample ratio specifies the likelihood of any event being included in the
  sample.
* For example, if sample_ratio = 500 each event has a 1/500 chance of being
  included in the sample result set.

```

```

* Default: 1

dataset.display.limiting = <int>
* The limit of events to search over when previewing the dataset.
* Default: 100000

dataset.display.currentCommand = <int>
* The currently selected command the user is on while editing the dataset.

dataset.display.mode = [table|datasummary]
* The type of preview to use when editing the dataset:
+ "table": show individual events/results as rows.
+ "datasummary": show field values as columns.
* Default: table

dataset.display.datasummary.earliestTime = <time-str>
* The earliest time used for the search that powers the datasummary view of
the dataset.

dataset.display.datasummary.latestTime = <time-str>
* The latest time used for the search that powers the datasummary view of
the dataset.

```

## datamodels.conf.example

```

# Version 6.5.0
#
# Configuration for example datamodels
#

# An example of accelerating data for the 'mymodel' datamodel for the
# past five days, generating and checking the column stores every 10 minutes
[mymodel]
acceleration = true
acceleration.earliest_time = -5d
acceleration.cron_schedule = */10 * * * *
acceleration.hunk.compression_codec = snappy
acceleration.hunk.dfs_block_size = 134217728
acceleration.hunk.file_format = orc

```

## datatypesbnf.conf

以下为 datatypesbnf.conf 的规范和示例文件。

### datatypesbnf.conf.spec

```

# Version 6.5.0
#
# This file effects how the search assistant (typeahead) shows the syntax for
# search commands

```

#### [<syntax-type>]

```

[<syntax-type>]
* The name of the syntax type you're configuring.
* Follow this field name with one syntax= definition.
* Syntax type can only contain a-z, and -, but cannot begin with -

syntax = <string>
* The syntax for you syntax type.
* Should correspond to a regular expression describing the term.
* Can also be a <field> or other similar value.

```

### datatypesbnf.conf.example

No example

## default.meta.conf

以下为 default.meta.conf 的规范和示例文件。

### default.meta.conf.spec

```
# Version 6.5.0
#
#
# *.meta files contain ownership information, access controls, and export
# settings for Splunk objects like saved searches, event types, and views.
# Each app has its own default.meta file.

# Interaction of ACLs across app-level, category level, and specific object
# configuration:
* To access/use an object, users must have read access to:
  * the app containing the object
  * the generic category within the app (eg [views])
  * the object itself
* If any layer does not permit read access, the object will not be accessible.

* To update/modify an object, such as to edit a saved search, users must have:
  * read and write access to the object
  * read access to the app, to locate the object
  * read access to the generic category within the app (eg. [savedsearches])
* If object does not permit write access to the user, the object will not be
  modifiable.
* If any layer does not permit read access to the user, the object will not be
  accessible in order to modify

* In order to add or remove objects from an app, users must have:
  * write access to the app
* If users do not have write access to the app, an attempt to add or remove an
  object will fail.

* Objects that are exported to other apps or to system context have no change
  to their accessibility rules. Users must still have read access to the
  containing app, category, and object, despite the export.

# Set access controls on the app containing this metadata file.
[]
access = read : [ * ], write : [ admin, power ]
* Allow all users to read this app's contents. Unless overridden by other
  metadata, allow only admin and power users to share objects into this app.

# Set access controls on this app's views.
```

### [views]

```
[views]
access = read : [ * ], write : [ admin ]
* Allow all users to read this app's views. Allow only admin users to create,
  remove, share, or unshare views in this app.

# Set access controls on a specific view in this app.
```

### [views/index\_status]

```
[views/index_status]
access = read : [ admin ], write : [ admin ]
* Allow only admin users to read or modify this view.

# Make this view available in all apps.
export = system
```

```
* To make this view available only in this app, set 'export = none' instead.
owner = admin
* Set admin as the owner of this view.
```

## default.meta.conf.example

```
# Version 6.5.0
#
# This file contains example patterns for the metadata files default.meta and
# local.meta
#
# This example would make all of the objects in an app globally accessible to
# all apps
[]
export=system
```

## default-mode.conf

以下为 default-mode.conf 的规范和示例文件。

### default-mode.conf.spec

```
# Version 6.5.0
#
# This file documents the syntax of default-mode.conf for comprehension and
# troubleshooting purposes.

# default-mode.conf is a file that exists primarily for Splunk Support and
# Services to configure splunk.

# CAVEATS:

# DO NOT make changes to default-mode.conf without coordinating with Splunk
# Support or Services. End-user changes to default-mode.conf are not
# supported.
#
# default-mode.conf *will* be removed in a future version of Splunk, along
# with the entire configuration scheme that it affects. Any settings present
# in default-mode.conf files will be completely ignored at this point.
#
# Any number of seemingly reasonable configurations in default-mode.conf
# might fail to work, behave bizarrely, corrupt your data, iron your cat,
# cause unexpected rashes, or order unwanted food delivery to your house.
# Changes here alter the way that pieces of code will communicate which are
# only intended to be used in a specific configuration.

# INFORMATION:

# The main value of this spec file is to assist in reading these files for
# troubleshooting purposes. default-mode.conf was originally intended to
# provide a way to describe the alternate setups used by the Splunk Light
# Forwarder and Splunk Universal Forwarder.

# The only reasonable action is to re-enable input pipelines that are
# disabled by default in those forwarder configurations. However, keep the
# prior caveats in mind. Any future means of enabling inputs will have a
# different form when this mechanism is removed.
```

```
# SYNTAX:
```

**[pipeline:<string>]**

```
[pipeline:<string>]
disabled = true | false
```

```
disabled_processors = <string>
```

### **[pipeline:<string>]**

```
[pipeline:<string>]
```

- \* Refers to a particular Splunkd pipeline.
- \* The set of named pipelines is a splunk-internal design. That does not mean that the Splunk design is a secret, but it means it is not external for the purposes of configuration.
- \* Useful information on the data processing system of splunk can be found in the external documentation, for example <http://docs.splunk.com/Documentation/Splunk/latest/Deploy/Datapipeline>

```
disabled = true | false
```

- \* If set to true on a specific pipeline, the pipeline will not be loaded in the system.

```
disabled_processors = <processor1>, <processor2>
```

- \* Processors which normally would be loaded in this pipeline are not loaded if they appear in this list
- \* The set of named processors is again a splunk-internal design component.

## **default-mode.conf.example**

No example

## **deployment.conf**

以下为 deployment.conf 的规范和示例文件。

### **deployment.conf.spec**

```
# Version 6.5.0
#
# *** REMOVED; NO LONGER USED ***
#
#
# This configuration file has been replaced by:
# 1.) deploymentclient.conf - for configuring Deployment Clients.
# 2.) serverclass.conf - for Deployment Server server class configuration.
#
#
# Compatibility:
# Splunk 4.x Deployment Server is NOT compatible with Splunk 3.x Deployment Clients.
#
```

## **deployment.conf.example**

No example

## **deploymentclient.conf**

以下为 deploymentclient.conf 的规范和示例文件。

### **deploymentclient.conf.spec**

```
# Version 6.5.0
#
# This file contains possible attributes and values for configuring a
# deployment client to receive content (apps and configurations) from a
# deployment server.
#
```

```
# To customize the way a deployment client behaves, place a
# deploymentclient.conf in $SPLUNK_HOME/etc/system/local/ on that Splunk
# instance. Configure what apps or configuration content is deployed to a
# given deployment client in serverclass.conf. Refer to
# serverclass.conf.spec and serverclass.conf.example for more information.
#
# You must restart Splunk for changes to this configuration file to take
# effect.
#
# To learn more about configuration files (including precedence) please see
# the documentation located at
# http://docs.splunk.com/Documentation/Splunk/latest/Admin/Aboutconfigurationfiles

*****
# Configure a Splunk deployment client.
#
# Note: At a minimum the [deployment-client] stanza is required in
# deploymentclient.conf for deployment client to be enabled.
*****
```

## 全局设置

```
# GLOBAL SETTINGS
# Use the [default] stanza to define any global settings.
# * You can also define global settings outside of any stanza, at the top
#   of the file.
# * Each conf file should have at most one default stanza. If there are
#   multiple default stanzas, attributes are combined. In the case of
#   multiple definitions of the same attribute, the last definition in the
#   file wins.
# * If an attribute is defined at both the global level and in a specific
#   stanza, the value in the specific stanza takes precedence.
```

## [deployment-client]

```
[deployment-client]
disabled = [false|true]
* Defaults to false
* Enable/Disable deployment client.

clientName = deploymentClient
* Defaults to deploymentClient.
* A name that the deployment server can filter on.
* Takes precedence over DNS names.

workingDir = $SPLUNK_HOME/var/run
* Temporary folder used by the deploymentClient to download apps and
  configuration content.

repositoryLocation = $SPLUNK_HOME/etc/apps
* The location into which content is installed after being downloaded from a
  deployment server.
* Apps and configuration content must be installed into the default location
  ($SPLUNK_HOME/etc/apps) or it will not be recognized by
  the Splunk instance on the deployment client.
  * Note: Apps and configuration content to be deployed may be located in
    an alternate location on the deployment server. Set both
    repositoryLocation and serverRepositoryLocationPolicy explicitly to
    ensure that the content is installed into the correct location
    ($SPLUNK_HOME/etc/apps) on the deployment client
  * The deployment client uses the 'serverRepositoryLocationPolicy'
    defined below to determine which value of repositoryLocation to use.

serverRepositoryLocationPolicy = [acceptSplunkHome|acceptAlways|rejectAlways]
* Defaults to acceptSplunkHome.
* acceptSplunkHome - accept the repositoryLocation supplied by the
  deployment server, only if it is rooted by
  $SPLUNK_HOME.
* acceptAlways - always accept the repositoryLocation supplied by the
```



```

        deployment server.
* rejectAlways - reject the server supplied value and use the
        repositoryLocation specified in the local
        deploymentclient.conf.

endpoint=$deploymentServerUri$/services/streams/deployment?name=$serverClassName:$appName$
* The HTTP endpoint from which content should be downloaded.
* Note: The deployment server may specify a different endpoint from which to
  download each set of content (individual apps, etc).
* The deployment client will use the serverEndpointPolicy defined below to
  determine which value to use.
* $deploymentServerUri$ will resolve to targetUri defined in the
  [target-broker] stanza below.
* $serverClassName$ and $appName$ mean what they say.

serverEndpointPolicy = [acceptAlways|rejectAlways]
* defaults to acceptAlways
* acceptAlways - always accept the endpoint supplied by the server.
* rejectAlways - reject the endpoint supplied by the server. Always use the
  'endpoint' definition above.

phoneHomeIntervalInSecs = <number in seconds>
* Defaults to 60.
* Fractional seconds are allowed.
* This determines how frequently this deployment client should check for new
  content.

handshakeRetryIntervalInSecs = <number in seconds>
* Defaults to one fifth of phoneHomeIntervalInSecs
* Fractional seconds are allowed.
* This sets the handshake retry frequency.
* Could be used to tune the initial connection rate on a new server

handshakeReplySubscriptionRetry = <integer>
* Defaults to 10
* If splunk is unable to complete the handshake, it will retry subscribing to
  the handshake channel after this many handshake attempts

appEventsResyncIntervalInSecs = <number in seconds>
* Defaults to 10*phoneHomeIntervalInSecs
* Fractional seconds are allowed.
* This sets the interval at which the client reports back its app state to the server.

# Advanced!
# You should use this property only when you have a hierarchical deployment
# server installation, and have a Splunk instance that behaves as both a
# DeploymentClient and a DeploymentServer.

# NOTE: hierarchical deployment servers are not a currently recommended
# configuration. Splunk has seen problems in the field that have not yet
# been resolved with this type of configuration.

reloadDSOnAppInstall = [false|true]
* Defaults to false
* Setting this flag to true will cause the deploymentServer on this Splunk
  instance to be reloaded whenever an app is installed by this
  deploymentClient.

sslVersions = <versions_list>
* Comma-separated list of SSL versions to connect to the specified Deployment Server
* The versions available are "ssl3", "tls1.0", "tls1.1", and "tls1.2".
* The special version "*" selects all supported versions. The version "tls"
  selects all versions tls1.0 or newer.
* If a version is prefixed with "-" it is removed from the list.
* SSLv2 is always disabled; "-ssl2" is accepted in the version list but does nothing.
* When configured in FIPS mode, ssl3 is always disabled regardless
  of this configuration.
* Defaults to sslVersions value in server.conf [sslConfig] stanza.

sslVerifyServerCert = <bool>
* If this is set to true, Splunk verifies that the Deployment Server (specified in 'targetUri')
  being connected to is a valid one (authenticated). Both the common

```

```

    name and the alternate name of the server are then checked for a
    match if they are specified in 'sslCommonNameToCheck' and 'sslAltNameToCheck'.
    A certificate is considered verified if either is matched.
* Defaults to sslVerifyServerCert value in server.conf [sslConfig] stanza.

caCertFile = <path>
* Full path to a CA (Certificate Authority) certificate(s) PEM format file.
* The <path> must refer to a PEM format file containing one or more root CA
  certificates concatenated together.
* Used for validating SSL certificate from Deployment Server
* Defaults to caCertFile value in server.conf [sslConfig] stanza.

sslCommonNameToCheck = <commonName1>, <commonName2>, ...
* If this value is set, and 'sslVerifyServerCert' is set to true,
  splunkd checks the common name(s) of the certificate presented by
  the Deployment Server (specified in 'targetUri') against this list of common names.
* Defaults to sslCommonNameToCheck value in server.conf [sslConfig] stanza.

sslAltNameToCheck = <alternateName1>, <alternateName2>, ...
* If this value is set, and 'sslVerifyServerCert' is set to true,
  splunkd checks the alternate name(s) of the certificate presented by
  the Deployment Server (specified in 'targetUri') against this list of subject alternate names.
* Defaults to sslAltNameToCheck value in server.conf [sslConfig] stanza.

cipherSuite = <cipher suite string>
* If set, uses the specified cipher string for making outbound HTTPS connection.

ecdhCurves = <comma separated list of ec curves>
* ECDH curves to use for ECDH key negotiation.
* The curves should be specified in the order of preference.
* The client sends these curves as a part of Client Hello.
* We only support named curves specified by their SHORT names.
  (see struct ASN1_OBJECT in asn1.h)
* The list of valid named curves by their short/long names can be obtained
  by executing this command:
  $SPLUNK_HOME/bin/splunk cmd openssl ecparam -list_curves
* Default is empty string.
* e.g. ecdhCurves = prime256v1,secp384r1,secp521r1

# The following stanza specifies deployment server connection information

```

### **[target-broker:deploymentServer]**

```

[target-broker:deploymentServer]
targetUri= <deploymentServer>:<mgmtPort>
* URI of the deployment server.

phoneHomeIntervalInSecs = <nonnegative number>
* see phoneHomeIntervalInSecs above

```

## **deploymentclient.conf.example**

```

# Version 6.5.0
#
# Example 1
# Deployment client receives apps and places them into the same
# repositoryLocation (locally, relative to $SPLUNK_HOME) as it picked them
# up from. This is typically $SPLUNK_HOME/etc/apps. There
# is nothing in [deployment-client] because the deployment client is not
# overriding the value set on the deployment server side.

[deployment-client]

[target-broker:deploymentServer]
targetUri= deploymentserver.splunk.mycompany.com:8089

# Example 2
# Deployment server keeps apps to be deployed in a non-standard location on

```

```

# the server side (perhaps for organization purposes).
# Deployment client receives apps and places them in the standard location.
# Note: Apps deployed to any location other than
# $SPLUNK_HOME/etc/apps on the deployment client side will
# not be recognized and run.
# This configuration rejects any location specified by the deployment server
# and replaces it with the standard client-side location.

[deployment-client]
serverRepositoryLocationPolicy = rejectAlways
repositoryLocation = $SPLUNK_HOME/etc/apps

[target-broker:deploymentServer]
targetUri= deploymentserver.splunk.mycompany.com:8089


# Example 3
# Deployment client should get apps from an HTTP server that is different
# from the one specified by the deployment server.

[deployment-client]
serverEndpointPolicy = rejectAlways
endpoint = http://apache.mycompany.server:8080/$serverClassName/$appName$.tar

[target-broker:deploymentServer]
targetUri= deploymentserver.splunk.mycompany.com:8089


# Example 4
# Deployment client should get apps from a location on the file system and
# not from a location specified by the deployment server

[deployment-client]
serverEndpointPolicy = rejectAlways
endpoint = file://<some_mount_point>/$serverClassName/$appName$.tar

[target-broker:deploymentServer]
targetUri= deploymentserver.splunk.mycompany.com:8089
handshakeRetryIntervalInSecs=20


# Example 5
# Deployment client should phonehome server for app updates quicker
# Deployment client should only send back appEvents once a day

[deployment-client]
phoneHomeIntervalInSecs=30
appEventsResyncIntervalInSecs=86400

[target-broker:deploymentServer]
targetUri= deploymentserver.splunk.mycompany.com:8089

```

## distsearch.conf

以下为 distsearch.conf 的规范和示例文件。

### distsearch.conf.spec

```

# Version 6.5.0
#
# This file contains possible attributes and values you can use to configure
# distributed search.
#
# To set custom configurations, place a distsearch.conf in
# $SPLUNK_HOME/etc/system/local/. For examples, see distsearch.conf.example.
# You must restart Splunk to enable configurations.
#
# To learn more about configuration files (including precedence) please see the
# documentation located at
# http://docs.splunk.com/Documentation/Splunk/latest/Admin/Aboutconfigurationfiles

```

```
#
# These attributes are all configured on the search head, with the exception of
# the optional attributes listed under the SEARCH HEAD BUNDLE MOUNTING OPTIONS
# heading, which are configured on the search peers.
```

## 全局设置

```
# GLOBAL SETTINGS
# Use the [default] stanza to define any global settings.
# * You can also define global settings outside of any stanza, at the top of
#   the file.
# * Each conf file should have at most one default stanza. If there are
#   multiple default stanzas, attributes are combined. In the case of
#   multiple definitions of the same attribute, the last definition in the
#   file wins.
# * If an attribute is defined at both the global level and in a specific
#   stanza, the value in the specific stanza takes precedence.

[distributedSearch]
* Set distributed search configuration options under this stanza name.
* Follow this stanza name with any number of the following attribute/value
  pairs.
* If you do not set any attribute, Splunk uses the default value (if there
  is one listed).

disabled = [true|false]
* Toggle distributed search off (true) and on (false).
* Defaults to false (your distributed search stanza is enabled by default).

heartbeatMcastAddr = <IP address>
* This setting is deprecated

heartbeatPort = <port>
* This setting is deprecated

ttl = <integer>
* This setting is deprecated

heartbeatFrequency = <int, in seconds>
* This setting is deprecated

statusTimeout = <int, in seconds>
* Set connection timeout when gathering a search peer's basic
  info (/services/server/info).
* Note: Read/write timeouts are automatically set to twice this value.
* Defaults to 10.

removedTimedOutServers = [true|false]
* This setting is no longer supported, and will be ignored.

checkTimedOutServersFrequency = <integer, in seconds>
* This setting is no longer supported, and will be ignored.

autoAddServers = [true|false]
* This setting is deprecated

bestEffortSearch = [true|false]
* Whether to remove a peer from search when it does not have any of our
  bundles.
* If set to true searches will never block on bundle replication, even when a
  peer is first added - the peers that don't have any common bundles will
  simply not be searched.
* Defaults to false

skipOurselves = [true|false]
* This setting is deprecated

servers = <comma separated list of servers>
* Initial list of servers.
* Each member of this list must be a valid uri in the format of scheme://hostname:port
```

```

disabled_servers = <comma separated list of servers>
* A list of disabled search peers. Peers in this list are not monitored or searched.
* Each member of this list must be a valid uri in the format of scheme://hostname:port

quarantined_servers = <comma separated list of servers>
* A list of quarantined search peers.
* Each member of this list must be a valid uri in the format of scheme://hostname:port
* The admin may quarantine peers that seem unhealthy and are degrading search
  performance of the whole deployment.
* Quarantined peers are monitored but not searched by default.
* A user may use the splunk_server arguments to target a search to quarantined peers
  at the risk of slowing the search.
* When a peer is quarantined, running realtime searches will NOT be restarted. Running
  realtime searches will continue to return results from the quarantined peers. Any
  realtime searches started after the peer has been quarantined will not contact the peer.
* Whenever a quarantined peer is excluded from search, appropriate warnings will be displayed
  in the search.log and Job Inspector

shareBundles = [true|false]
* Indicates whether this server will use bundle replication to share search
  time configuration with search peers.
* If set to false, the search head assumes that all the search peers can access
  the correct bundles via share head storage and have configured the options listed
  under "SEARCH HEAD BUNDLE MOUNTING OPTIONS".
* Defaults to true.

useSHPBundleReplication = <bool>|always
* Relevant only in search head pooling environments. Whether the search heads
  in the pool should compete with each other to decide which one should handle
  the bundle replication (every time bundle replication needs to happen) or
  whether each of them should individually replicate the bundles.
* When set to always and bundle mounting is being used then use the search head
  pool guid rather than each individual server name to identify bundles (and
  search heads to the remote peers).
* Defaults to true

trySSLFirst = <bool>
* This setting is no longer supported, and will be ignored.

peerResolutionThreads = <int>
* This setting is no longer supported, and will be ignored.

defaultUriScheme = [http|https]
* When a new peer is added without specifying a scheme for the uri to its management
  port we will use this scheme by default.
* Defaults to https

serverTimeout = <int, in seconds>
* REMOVED, this setting is now ignored and has been replaced by
  connectionTimeout, sendTimeout, receiveTimeout

connectionTimeout = <int, in seconds>
* Amount of time in seconds to use as a timeout during search peer connection
  establishment.

sendTimeout = <int, in seconds>
* Amount of time in seconds to use as a timeout while trying to write/send data
  to a search peer.

receiveTimeout = <int, in seconds>
* Amount of time in seconds to use as a timeout while trying to read/receive
  data from a search peer.

authTokenConnectionTimeout = <number, in seconds>
* Maximum number of seconds to connect to a remote search peer, when getting
  its auth token
* Fractional seconds are allowed
* Default is 5

authTokenSendTimeout = <number, in seconds>
* Maximum number of seconds to send a request to the remote peer, when getting

```

```

    its auth token
* Fractional seconds are allowed
* Default is 10

authTokenReceiveTimeout = <number, in seconds>
* Maximum number of seconds to receive a response from a remote peer, when
  getting its auth token
* Fractional seconds are allowed
* Default is 10

#*****

```

## 分布式搜索关键对生成选项

```

# DISTRIBUTED SEARCH KEY PAIR GENERATION OPTIONS
#*****

[tokenExchKeys]

certDir = <directory>
* This directory contains the local Splunk instance's distributed search key
  pair.
* This directory also contains the public keys of servers that distribute
  searches to this Splunk instance.

publicKey = <filename>
* Name of public key file for this Splunk instance.

privateKey = <filename>
* Name of private key file for this Splunk instance.

genKeyScript = <command>
* Command used to generate the two files above.

#*****

```

## 复制设置选项

```

# REPLICATION SETTING OPTIONS
#*****

[replicationSettings]

connectionTimeout = <int, in seconds>
* The maximum number of seconds to wait before timing out on initial connection
  to a peer.

sendRcvTimeout = <int, in seconds>
* The maximum number of seconds to wait for the sending of a full replication
  to a peer.

replicationThreads = <int>
* The maximum number of threads to use when performing bundle replication to peers.
* Must be a positive number
* Defaults to 5.

maxMemoryBundleSize = <int>
* The maximum size (in MB) of bundles to hold in memory. If the bundle is
  larger than this the bundles will be read and encoded on the fly for each
  peer the replication is taking place.
* Defaults to 10

maxBundleSize = <int>
* The maximum size (in MB) of the bundle for which replication can occur. If
  the bundle is larger than this bundle replication will not occur and an
  error message will be logged.
* Defaults to: 1024 (1GB)

concerningReplicatedFileSize = <int>
* Any individual file within a bundle that is larger than this value (in MB)

```

```

    will trigger a splunkd.log message.
* Where possible, avoid replicating such files, e.g. by customizing your blacklists.
* Defaults to: 50

excludeReplicatedLookupSize = <int>
* Any lookup file larger than this value (in MB) will be excluded from the knowledge bundle that the search head
replicates to its search peers.
* When this value is set to 0, this feature is disabled.
* Defaults to 0

allowStreamUpload = auto | true | false
* Whether to enable streaming bundle replication for peers.
* If set to auto, streaming bundle replication will be used when connecting to
peers with a complete implementation of this feature (Splunk 6.0 or higher).
* If set to true, streaming bundle replication will be used when connecting to
peers with a complete or experimental implementation of this feature (Splunk
4.2.3 or higher).
* If set to false, streaming bundle replication will never be used.
    Whatever the value of this setting, streaming bundle replication will not be
    used for peers that completely lack support for this feature.
* Defaults to: auto

allowSkipEncoding = <bool>
* Whether to avoid URL-encoding bundle data on upload.
* Defaults to: true

allowDeltaUpload = <bool>
* Whether to enable delta-based bundle replication.
* Defaults to: true

sanitizeMetaFiles = <bool>
* Whether to sanitize or filter *.meta files before replication.
* This feature can be used to avoid unnecessary replications triggered by
writes to *.meta files that have no real effect on search behavior.
* The types of stanzas that "survive" filtering are configured via the
replicationSettings:refineConf stanza.
* The filtering process removes comments and cosmetic whitespace.
* Defaults to: true

[replicationSettings:refineConf]

replicate.<conf_file_name> = <bool>
* Controls whether Splunk replicates a particular type of *.conf file, along
with any associated permissions in *.meta files.
* These settings on their own do not cause files to be replicated. A file must
still be whitelisted (via replicationWhitelist) to be eligible for inclusion
via these settings.
* In a sense, these settings constitute another level of filtering that applies
specifically to *.conf files and stanzas with *.meta files.
* Defaults to: false

#*****

```

## 复制白名单选项

```

# REPLICATION WHITELIST OPTIONS
#*****

[replicationWhitelist]

<name> = <whitelist_pattern>
* Controls Splunk's search-time conf replication from search heads to search
nodes.
* Only files that match a whitelist entry will be replicated.
* Conversely, files which are not matched by any whitelist will not be
replicated.
* Only files located under $SPLUNK_HOME/etc will ever be replicated in this
way.
* The regex will be matched against the filename, relative to $SPLUNK_HOME/etc.
    Example: for a file "$SPLUNK_HOME/etc/apps/fancy_app/default/inputs.conf"
            this whitelist should match "apps/fancy_app/default/inputs.conf"

```

```

* Similarly, the etc/system files are available as system/...
  user-specific files are available as users/username/appname/...
* The 'name' element is generally just descriptive, with one exception:
  if <name> begins with "refine.", files whitelisted by the given pattern will
  also go through another level of filtering configured in the
  replicationSettings:refineConf stanza.
* The whitelist_pattern is the Splunk-style pattern matching, which is
  primarily regex-based with special local behavior for '...' and '*'.
  * ... matches anything, while * matches anything besides directory separators.
    See props.conf.spec for more detail on these.
  * Note '.' will match a literal dot, not any character.
* Note that these lists are applied globally across all conf data, not to any
  particular app, regardless of where they are defined. Be careful to pull in
  only your intended files.

```

```

#*****

```

## 复制黑名单选项

```

# REPLICATION BLACKLIST OPTIONS
#*****

[replicationBlacklist]

<name> = <blacklist_pattern>
* All comments from the replication whitelist notes above also apply here.
* Replication blacklist takes precedence over the whitelist, meaning that a
  file that matches both the whitelist and the blacklist will NOT be
  replicated.
* This can be used to prevent unwanted bundle replication in two common
  scenarios:
  * Very large files, which part of an app may not want to be replicated,
    especially if they are not needed on search nodes.
  * Frequently updated files (for example, some lookups) will trigger
    retransmission of all search head data.
* Note that these lists are applied globally across all conf data. Especially
  for blacklisting, be careful to constrain your blacklist to match only data
  your application will not need.

```

```

#*****

```

## 捆绑包执行者白名单选项

```

# BUNDLE ENFORCER WHITELIST OPTIONS
#*****

[bundleEnforcerWhitelist]

<name> = <whitelist_pattern>
* Peers uses this to make sure knowledge bundle sent by search heads and
  masters do not contain alien files.
* If this stanza is empty, the receiver accepts the bundle unless it contains
  files matching the rules specified in [bundleEnforcerBlacklist]. Hence, if
  both [bundleEnforcerWhitelist] and [bundleEnforcerBlacklist] are empty (which
  is the default), then the receiver accepts all bundles.
* If this stanza is not empty, the receiver accepts the bundle only if it
  contains only files that match the rules specified here but not those in
  [bundleEnforcerBlacklist].
* All rules are regexs.
* This stanza is empty by default.

```

```

#*****

```

## 捆绑包执行者黑名单选项

```

# BUNDLE ENFORCER BLACKLIST OPTIONS
#*****

```



```
[bundleEnforcerBlacklist]

<name> = <blacklist_pattern>
* Peers uses this to make sure knowledge bundle sent by search heads and
  masters do not contain alien files.
* This list overrides [bundleEnforceWhitelist] above. That means the receiver
  rejects (i.e. removes) the bundle if it contains any file that matches the
  rules specified here even if that file is allowed by [bundleEnforcerWhitelist].
* If this stanza is empty, then only [bundleEnforcerWhitelist] matters.
* This stanza is empty by default.

#*****
```

## 搜索头捆绑包安装选项

```
# SEARCH HEAD BUNDLE MOUNTING OPTIONS
# You set these attributes on the search peers only, and only if you also set
# shareBundles=false in [distributedSearch] on the search head. Use them to
# achieve replication-less bundle access. The search peers use a shared storage
# mountpoint to access the search head bundles ($SPLUNK_HOME/etc).
#*****

[searchhead:<searchhead-splunk-server-name>]
* <searchhead-splunk-server-name> is the name of the related searchhead
  installation.
* This setting is located in server.conf, serverName = <name>

mounted_bundles = [true|false]
* Determines whether the bundles belong to the search head specified in the
  stanza name are mounted.
* You must set this to "true" to use mounted bundles.
* Default is "false".

bundles_location = <path_to_bundles>
* The path to where the search head's bundles are mounted. This must be the
  mountpoint on the search peer, not on the search head. This should point to
  a directory that is equivalent to $SPLUNK_HOME/etc/. It must contain at least
  the following subdirectories: system, apps, users.

#*****
```

## 分布式搜索组定义

```
# DISTRIBUTED SEARCH GROUP DEFINITIONS
# These are the definitions of the distributed search groups. A search group is
# a set of search peers as identified by thier host:management-port. A search
# may be directed to a search group using the splunk_server_group argument.The
# search will be dispatched to only the members of the group.
#*****

[distributedSearch:<splunk-server-group-name>]
* <splunk-server-group-name> is the name of the splunk-server-group that is
  defined in this stanza

servers = <comma separated list of servers>
* List of search peers that are members of this group. Comma serparated list
  of peer identifiers i.e. hostname:port

default = [true|false]
* Will set this as the default group of peers against which all searches are
  run unless a server-group is not explicitly specified.
```

## distsearch.conf.example

```
# Version 6.5.0
#
```

```

# These are example configurations for distsearch.conf. Use this file to
# configure distributed search. For all available attribute/value pairs, see
# distsearch.conf.spec.
#
# There is NO DEFAULT distsearch.conf.
#
# To use one or more of these configurations, copy the configuration block into
# distsearch.conf in $SPLUNK_HOME/etc/system/local/. You must restart Splunk
# to enable configurations.
#
# To learn more about configuration files (including precedence) please see the
# documentation located at
# http://docs.splunk.com/Documentation/Splunk/latest/Admin/Aboutconfigurationfiles

[distributedSearch]
servers = https://192.168.1.1:8059,https://192.168.1.2:8059

# This entry distributes searches to 192.168.1.1:8059,192.168.1.2:8059.
# These machines will be contacted on port 8059 using https
# Attributes not set here will use the defaults listed in distsearch.conf.spec.

# this stanza controls the timing settings for connecting to a remote peer and
# the send timeout
[replicationSettings]
connectionTimeout = 10
sendRcvTimeout = 60

# this stanza controls what files are replicated to the other peer each is a
# regex
[replicationWhitelist]
allConf = *.conf

# Mounted bundles example.
# This example shows two distsearch.conf configurations, one for the search
# head and another for each of the search head's search peers. It shows only
# the attributes necessary to implement mounted bundles.

# On a search head whose Splunk server name is "searcher01":
[distributedSearch]
...
shareBundles = false

# On each search peer:
[searchhead:searcher01]
mounted_bundles = true
bundles_location = /opt/shared_bundles/searcher01

```

## eventdiscoverer.conf

以下为 eventdiscoverer.conf 的规范和示例文件。

### eventdiscoverer.conf.spec

```

# Version 6.5.0

# This file contains possible attributes and values you can use to configure
# event discovery through the search command "typelearner."
#
# There is an eventdiscoverer.conf in $SPLUNK_HOME/etc/system/default/. To set
# custom configurations, place an eventdiscoverer.conf in
# $SPLUNK_HOME/etc/system/local/. For examples, see
# eventdiscoverer.conf.example. You must restart Splunk to enable
# configurations.
#
# To learn more about configuration files (including precedence) please see the
# documentation located at
# http://docs.splunk.com/Documentation/Splunk/latest/Admin/Aboutconfigurationfiles

```

## 全局设置

```
# GLOBAL SETTINGS
# Use the [default] stanza to define any global settings.
# * You can also define global settings outside of any stanza, at the top of
#   the file.
# * Each conf file should have at most one default stanza. If there are
#   multiple default stanzas, attributes are combined. In the case of
#   multiple definitions of the same attribute, the last definition in the
#   file wins.
# * If an attribute is defined at both the global level and in a specific
#   stanza, the value in the specific stanza takes precedence.

ignored_keywords = <comma-separated list of terms>
* If you find that event types have terms you do not want considered (for
  example, "mylaptopname"), add that term to this list.
* Terms in this list are never considered for defining an event type.
* For more details, refer to $SPLUNK_HOME/etc/system/default/eventdiscoverer.conf).
* Default = "sun, mon, tue,..."

ignored_fields = <comma-separated list of fields>
* Similar to ignored_keywords, except these are fields as defined in Splunk
  instead of terms.
* Defaults include time-related fields that would not be useful for defining an
  event type.

important_keywords = <comma-separated list of terms>
* When there are multiple possible phrases for generating an eventtype search,
  those phrases with important_keyword terms are favored. For example,
  "fatal error" would be preferred over "last message repeated", as "fatal" is
  an important keyword.
* Default = "abort, abstract, accept,..."
* For the full default setting, see $SPLUNK_HOME/etc/system/default/eventdiscoverer.conf.
```

## eventdiscoverer.conf.example

```
# Version 6.5.0
#
# This is an example eventdiscoverer.conf. These settings are used to control
# the discovery of common eventtypes used by the typelearner search command.
#
# To use one or more of these configurations, copy the configuration block into
# eventdiscoverer.conf in $SPLUNK_HOME/etc/system/local/. You must restart
# Splunk to enable configurations.
#
# To learn more about configuration files (including precedence) please see the
# documentation located at
# http://docs.splunk.com/Documentation/Splunk/latest/Admin/Aboutconfigurationfiles

# Terms in this list are never considered for defining an eventtype.
ignored_keywords = foo, bar, application, kate, charlie

# Fields in this list are never considered for defining an eventtype.
ignored_fields = pid, others, directory
```

## event\_renderers.conf

以下为 event\_renderers.conf 的规范和示例文件。

### event\_renderers.conf.spec

```
# Version 6.5.0
#
# This file contains possible attribute/value pairs for configuring event rendering properties.
#
```

```
# Beginning with version 6.0, Splunk Enterprise does not support the
# customization of event displays using event renderers.
#
# There is an event_renderers.conf in $SPLUNK_HOME/etc/system/default/. To set custom configurations,
# place an event_renderers.conf in $SPLUNK_HOME/etc/system/local/, or your own custom app directory.
#
# To learn more about configuration files (including precedence) please see the documentation
# located at http://docs.splunk.com/Documentation/Splunk/latest/Admin/Aboutconfigurationfiles
```

## 全局设置

```
# GLOBAL SETTINGS
# Use the [default] stanza to define any global settings.
#
# * You can also define global settings outside of any stanza, at the top of the file.
#
# * Each conf file should have at most one default stanza. If there are multiple default
#   stanzas, attributes are combined. In the case of multiple definitions of the same
#   attribute, the last definition in the file wins.
#
# * If an attribute is defined at both the global level and in a specific stanza, the
#   value in the specific stanza takes precedence.
```

### [<name>]

```
[<name>]
* Stanza name. This name must be unique.

eventtype = <event type>
* Specify event type name from eventtypes.conf.

priority = <positive integer>
* Highest number wins!!

template = <valid Mako template>
* Any template from the $APP/appserver/event_renderers directory.

css_class = <css class name suffix to apply to the parent event element class attribute>
* This can be any valid css class value.
* The value is appended to a standard suffix string of "splEvent-". A css_class value of foo would
  result in the parent element of the event having an html attribute class with a value of splEvent-foo
  (for example, class="splEvent-foo"). You can externalize your css style rules for this in
  $APP/appserver/static/application.css. For example, to make the text red you would add to
  application.css:splEvent-foo { color:red; }
```

## event\_renderers.conf.example

```
# Version 6.5.0
# DO NOT EDIT THIS FILE!
# Please make all changes to files in $SPLUNK_HOME/etc/system/local.
# To make changes, copy the section/stanza you want to change from $SPLUNK_HOME/etc/system/default
# into ../local and edit there.
#
# This file contains mappings between Splunk eventtypes and event renderers.
#
# Beginning with version 6.0, Splunk Enterprise does not support the
# customization of event displays using event renderers.
#

[event_renderer_1]
eventtype = hawaiian_type
priority = 1
css_class = EventRenderer1

[event_renderer_2]
eventtype = french_food_type
priority = 1
template = event_renderer2.html
css_class = EventRenderer2
```

```
[event_renderer_3]
eventtype = japan_type
priority = 1
css_class = EventRenderer3
```

## eventtypes.conf

以下为 eventtypes.conf 的规范和示例文件。

### eventtypes.conf.spec

```
# Version 6.5.0
#
# This file contains all possible attributes and value pairs for an
# eventtypes.conf file. Use this file to configure event types and their
# properties. You can also pipe any search to the "typelearner" command to
# create event types. Event types created this way will be written to
# $SPLUNK_HOME/etc/system/local/eventtypes.conf.
#
# There is an eventtypes.conf in $SPLUNK_HOME/etc/system/default/. To set
# custom configurations, place an eventtypes.conf in
# $SPLUNK_HOME/etc/system/local/. For examples, see eventtypes.conf.example.
# You must restart Splunk to enable configurations.
#
# To learn more about configuration files (including precedence) please see
# the documentation located at
# http://docs.splunk.com/Documentation/Splunk/latest/Admin/Aboutconfigurationfiles
```

### 全局设置

```
# GLOBAL SETTINGS
# Use the [default] stanza to define any global settings.
# * You can also define global settings outside of any stanza, at the top
#   of the file.
# * Each conf file should have at most one default stanza. If there are
#   multiple default stanzas, attributes are combined. In the case of
#   multiple definitions of the same attribute, the last definition in the
#   file wins.
# * If an attribute is defined at both the global level and in a specific
#   stanza, the value in the specific stanza takes precedence.
```

### [<\$EVENTTYPE>]

```
[<$EVENTTYPE>]
* Header for the event type
* $EVENTTYPE is the name of your event type.
* You can have any number of event types, each represented by a stanza and
  any number of the following attribute/value pairs.
* NOTE: If the name of the event type includes field names surrounded by the
  percent character (for example "%$FIELD%") then the value of $FIELD is
  substituted into the event type name for that event. For example, an
  event type with the header [cisco-%code%] that has "code=432" becomes
  labeled "cisco-432".

disabled = [1|0]
* Toggle event type on or off.
* Set to 1 to disable.

search = <string>
* Search terms for this event type.
* For example: error OR warn.
* NOTE: You cannot base an event type on:
  * A search that includes a pipe operator (a "|" character).
  * A subsearch (a search pipeline enclosed in square brackets).
  * A search referencing a report. This is a best practice. Any report that is referenced by an
```

event type can later be updated in a way that makes it invalid as an event type. For example, a report that is updated to include transforming commands cannot be used as the definition for an event type. You have more control over your event type if you define it with the same search string as the report.

```
priority = <integer, 1 through 10>
* Value used to determine the order in which the matching eventtypes of an
  event are displayed.
* 1 is the highest priority and 10 is the lowest priority.

description = <string>
* Optional human-readable description of this saved search.

tags = <string>
* DEPRECATED - see tags.conf.spec

color = <string>
* color for this event type.
* Supported colors: none, et_blue, et_green, et_magenta, et_orange,
  et_purple, et_red, et_sky, et_teal, et_yellow
```

## eventtypes.conf.example

```
# Version 6.5.0
#
# This file contains an example eventtypes.conf. Use this file to configure custom eventtypes.
#
# To use one or more of these configurations, copy the configuration block into eventtypes.conf
# in $SPLUNK_HOME/etc/system/local/. You must restart Splunk to enable configurations.
#
# To learn more about configuration files (including precedence) please see the documentation
# located at http://docs.splunk.com/Documentation/Splunk/latest/Admin/Aboutconfigurationfiles
#

# The following example makes an eventtype called "error" based on the search "error OR fatal."

[error]
search = error OR fatal


# The following example makes an eventtype template because it includes a field name
# surrounded by the percent character (in this case "%code%").
# The value of "%code%" is substituted into the event type name for that event.
# For example, if the following example event type is instantiated on an event that has a
# "code=432," it becomes "cisco-432".

[cisco-%code%]
search = cisco
```

## fields.conf

以下为 fields.conf 的规范和示例文件。

### fields.conf.spec

```
# Version 6.5.0
#
# This file contains possible attribute and value pairs for:
# * Telling Splunk how to handle multi-value fields.
# * Distinguishing indexed and extracted fields.
# * Improving search performance by telling the search processor how to
  handle field values.

# Use this file if you are creating a field at index time (not advised).
#
```

```
# There is a fields.conf in $SPLUNK_HOME/etc/system/default/. To set custom
# configurations, place a fields.conf in $SPLUNK_HOME/etc/system/local/. For
# examples, see fields.conf.example. You must restart Splunk to enable
# configurations.
#
# To learn more about configuration files (including precedence) please see the
# documentation located at
# http://docs.splunk.com/Documentation/Splunk/latest/Admin/Aboutconfigurationfiles
```

## 全局设置

```
# GLOBAL SETTINGS
# Use the [default] stanza to define any global settings.
# * You can also define global settings outside of any stanza, at the top of
#   the file.
# * Each conf file should have at most one default stanza. If there are
#   multiple default stanzas, attributes are combined. In the case of
#   multiple definitions of the same attribute, the last definition in the
#   file wins.
# * If an attribute is defined at both the global level and in a specific
#   stanza, the value in the specific stanza takes precedence.
```

### [<field name>]

```
[<field name>]
* Name of the field you're configuring.
* Follow this stanza name with any number of the following attribute/value
  pairs.
* Field names can only contain a-z, A-Z, 0-9, and _, but cannot begin with a
  number or _

# TOKENIZER indicates that your configured field's value is a smaller part of a
# token. For example, your field's value is "123" but it occurs as "foo123" in
# your event.
TOKENIZER = <regular expression>
* Use this setting to configure multivalue fields (refer to the online
  documentation for multivalue fields).
* A regular expression that indicates how the field can take on multiple values
  at the same time.
* If empty, the field can only take on a single value.
* Otherwise, the first group is taken from each match to form the set of
  values.
* This setting is used by the "search" and "where" commands, the summary and
  XML outputs of the asynchronous search API, and by the top, timeline and
  stats commands.
* Tokenization of indexed fields (INDEXED = true) is not supported so this
  attribute is ignored for indexed fields.
* Default to empty.

INDEXED = [true|false]
* Indicate whether a field is indexed or not.
* Set to true if the field is indexed.
* Set to false for fields extracted at search time (the majority of fields).
* Defaults to false.

INDEXED_VALUE = [true|false|<sed-cmd>|<simple-substitution-string>]
* Set this to true if the value is in the raw text of the event.
* Set this to false if the value is not in the raw text of the event.
* Setting this to true expands any search for key=value into a search of
  value AND key=value (since value is indexed).
* For advanced customization, this setting supports sed style substitution.
  For example, 'INDEXED_VALUE=s/foo/bar/g' would take the value of the field,
  replace all instances of 'foo' with 'bar,' and use that new value as the
  value to search in the index.
* This setting also supports a simple substitution based on looking for the
  literal string '<VALUE>' (including the '<' and '>' characters).
  For example, 'INDEXED_VALUE=source:*<VALUE>*' would take a search for
  'myfield=myvalue' and search for 'source:*myvalue*' in the index as a
  single term.
```

\* For both substitution constructs, if the resulting string starts with a '[', Splunk interprets the string as a Splunk LISPY expression. For example, 'INDEXED\_VALUE=[OR <VALUE> source::\*<VALUE>]' would turn 'myfield=myvalue' into applying the LISPY expression '[OR myvalue source::\*myvalue]' (meaning it matches either 'myvalue' or 'source::\*myvalue' terms).

\* Defaults to true.

\* NOTE: You only need to set indexed\_value if indexed = false.

## fields.conf.example

```
# Version 6.5.0
#
# This file contains an example fields.conf. Use this file to configure
# dynamic field extractions.
#
# To use one or more of these configurations, copy the configuration block into
# fields.conf in $SPLUNK_HOME/etc/system/local/. You must restart Splunk to
# enable configurations.
#
# To learn more about configuration files (including precedence) please see the
# documentation located at
# http://docs.splunk.com/Documentation/Splunk/latest/Admin/Aboutconfigurationfiles
#
# These tokenizers result in the values of To, From and Cc treated as a list,
# where each list element is an email address found in the raw string of data.

[To]
TOKENIZER = (\w[\w\.\-]*@[\w\.\-]*\w)

[From]
TOKENIZER = (\w[\w\.\-]*@[\w\.\-]*\w)

[Cc]
TOKENIZER = (\w[\w\.\-]*@[\w\.\-]*\w)
```

## indexes.conf

以下为 indexes.conf 的规范和示例文件。

### indexes.conf.spec

```
# Version 6.5.0
#
# This file contains all possible options for an indexes.conf file. Use
# this file to configure Splunk's indexes and their properties.
#
# There is an indexes.conf in $SPLUNK_HOME/etc/system/default/. To set
# custom configurations, place an indexes.conf in
# $SPLUNK_HOME/etc/system/local/. For examples, see indexes.conf.example.
# You must restart Splunk to enable configurations.
#
# To learn more about configuration files (including precedence) please see
# the documentation located at
# http://docs.splunk.com/Documentation/Splunk/latest/Admin/Aboutconfigurationfiles
#
# CAUTION: You can drastically affect your Splunk installation by changing
# these settings. Consult technical support
# (http://www.splunk.com/page/submit_issue) if you are not sure how to
# configure this file.
#
```

### 全局设置

```
# GLOBAL SETTINGS
# Use the [default] stanza to define any global settings.
# * You can also define global settings outside of any stanza, at the top
```



```

# of the file.
# * Each conf file should have at most one default stanza. If there are
# multiple default stanzas, attributes are combined. In the case of
# multiple definitions of the same attribute, the last definition in the
# file wins.
# * If an attribute is defined at both the global level and in a specific
# stanza, the value in the specific stanza takes precedence.

sync = <nonnegative integer>
* The index processor syncs events every <integer> number of events.
* Set to 0 to disable.
* Highest legal value is 32767
* Defaults to 0.

defaultDatabase = <index name>
* If no index is specified during search, Splunk searches the default index.
* The specified index displays as the default in Splunk Manager settings.
* Defaults to "main".

queryLanguageDefinition = <path to file>
* DO NOT EDIT THIS SETTING. SERIOUSLY.
* The path to the search language definition file.
* Defaults to $SPLUNK_HOME/etc/searchLanguage.xml.

lastChanceIndex = <index name>
* Gives ability to define a last chance index for events destined for
non-existent indexes.
* If an event arrives whose index destination key points to an index that is
not configured (such as when using index=<index name> in the input stanza or
by a setting in a transform), it will route that event to the index specified
by this setting. The index destination key of that event will be overwritten
with the specified index name before routing.
* <index name> must name an existing enabled index. Splunk will not start if
this is not the case.
* If this setting is not defined or is empty, it will drop such events.
* If set to "default", then the default index specified by the
"defaultDatabase" will be used as a last chance index.
* Defaults to empty.

memPoolMB = <positive integer>|auto
* Determines how much memory is given to the indexer memory pool. This
restricts the number of outstanding events in the indexer at any given
time.
* Must be greater than 0; maximum value is 1048576 (which corresponds to 1 TB)
* Setting this too high can lead to splunkd memory usage going up
substantially.
* Setting this too low can degrade splunkd indexing performance.
* Setting this to "auto" or an invalid value will cause Splunk to autotune
this parameter.
* Defaults to "auto".
  * The values derived when "auto" is seen are as follows:
    * System Memory Available less than ... | memPoolMB
      1 GB | 64 MB
      2 GB | 128 MB
      8 GB | 128 MB
      8 GB or higher | 512 MB
* Only set this value if you are an expert user or have been advised to by
Splunk Support.
* CARELESSNESS IN SETTING THIS MAY LEAD TO PERMANENT BRAIN DAMAGE OR
LOSS OF JOB.

indexThreads = <nonnegative integer>|auto
* Determines the number of threads to use for indexing.
* Must be at least 1 and no more than 16.
* This value should not be set higher than the number of processor cores in
the box.
* If splunkd is also doing parsing and aggregation, the number should be set
lower than the total number of processors minus two.
* Setting this to "auto" or an invalid value will cause Splunk to autotune
this parameter.
* Only set this value if you are an expert user or have been advised to by
Splunk Support.

```

\* CARELESSNESS IN SETTING THIS MAY LEAD TO PERMANENT BRAIN DAMAGE OR LOSS OF JOB.

\* Defaults to "auto".

rtRouterThreads = 0|1

\* Set this to 1 if you expect to use non-indexed real time searches regularly. Index throughput drops rapidly if there are a handful of these running concurrently on the system.

\* If you are not sure what "indexed vs non-indexed" real time searches are, see README of indexed\_realtime\* settings in limits.conf

\* NOTE: This is not a boolean value, only 0 or 1 is accepted. In the future, we may allow more than a single thread, but current implementation only allows one to create a single thread per pipeline set

rtRouterQueueSize = <positive integer>

\* Defaults to 10000

\* This setting is only relevant if rtRouterThreads != 0

\* This queue sits between the indexer pipeline set thread (producer) and the rtRouterThread

\* Changing the size of this queue may impact real time search performance

assureUTF8 = true|false

\* Verifies that all data retrieved from the index is proper by validating all the byte strings.

\* This does not ensure all data will be emitted, but can be a workaround if an index is corrupted in such a way that the text inside it is no longer valid utf8.

\* Will degrade indexing performance when enabled (set to true).

\* Can only be set globally, by specifying in the [default] stanza.

\* Defaults to false.

enableRealtimeSearch = true|false

\* Enables real-time searches.

\* Defaults to true.

suppressBannerList = <comma-separated list of strings>

\* suppresses index missing warning banner messages for specified indexes

\* Defaults to empty

maxRunningProcessGroups = <positive integer>

\* splunkd fires off helper child processes like splunk-optimize, recover-metadate, etc. This param limits how many child processes can be running at any given time.

\* This maximum applies to entire splunkd, not per index. If you have N indexes, there will be at most maxRunningProcessGroups child processes, not N\*maxRunningProcessGroups

\* Must maintain maxRunningProcessGroupsLowPriority < maxRunningProcessGroups

\* This is an advanced parameter; do NOT set unless instructed by Splunk Support

\* Highest legal value is 4294967295

\* Defaults to 8 (note: up until 5.0 it defaulted to 20)

maxRunningProcessGroupsLowPriority = <positive integer>

\* Of the maxRunningProcessGroups (q.v.) helper child processes, at most maxRunningProcessGroupsLowPriority may be low-priority (e.g. fsck) ones.

\* This maximum applies to entire splunkd, not per index. If you have N indexes, there will be at most maxRunningProcessGroupsLowPriority low-priority child processes, not N\*maxRunningProcessGroupsLowPriority

\* Must maintain maxRunningProcessGroupsLowPriority < maxRunningProcessGroups

\* This is an advanced parameter; do NOT set unless instructed by Splunk Support

\* Highest legal value is 4294967295

\* Defaults to 1

bucketRebuildMemoryHint = <positive integer>[KB|MB|GB]|auto

\* Suggestion for the bucket rebuild process for the size (bytes) of tsidx file it will try to build.

\* Larger files use more memory in rebuild, but rebuild will fail if there is not enough.

\* Smaller files make the rebuild take longer during the final optimize step.

\* Note: this value is not a hard limit on either rebuild memory usage or tsidx size.

\* This is an advanced parameter, do NOT set this unless instructed by Splunk Support.

- \* Defaults to "auto", which varies by the amount of physical RAM on the host
  - \* less than 2GB RAM = 67108864 (64MB) tsidx
  - \* 2GB to 8GB RAM = 134217728 (128MB) tsidx
  - \* more than 8GB RAM = 268435456 (256MB) tsidx
- \* If not "auto", then must be 16MB-1GB.
- \* Value may be specified using a size suffix: "16777216" or "16MB" are equivalent.
- \* Inappropriate use of this parameter will cause splunkd to not start if rebuild is required.
- \* Highest legal value (in bytes) is 4294967295

inPlaceUpdates = true|false

- \* If true, metadata updates are written to the .data files directly
- \* If false, metadata updates are written to a temporary file and then moved into place
- \* Intended for advanced debugging of metadata issues
- \* Setting this parameter to false (to use a temporary file) will impact indexing performance, particularly with large numbers of hosts, sources, or sourcetypes (~1 million, across all indexes.)
- \* This is an advanced parameter; do NOT set unless instructed by Splunk Support
- \* Defaults to true

serviceOnlyAsNeeded = true|false

- \* Causes index service (housekeeping tasks) overhead to be incurred only after index activity.
- \* Indexer module problems may be easier to diagnose when this optimization is disabled (set to false).
- \* Defaults to true.

serviceSubtaskTimingPeriod = <positive integer>

- \* Subtasks of indexer service task will be timed on every Nth execution, where N = value of this parameter, in seconds.
- \* Smaller values will give greater accuracy; larger values will lessen timer overhead.
- \* Timer measurements will be found in metrics.log, marked "group=subtask\_seconds, task=indexer\_service"
- \* Highest legal value is 4294967295
- \* We strongly suggest value of this parameter divide evenly into value of 'rotatePeriodInSecs' parameter.
- \* Defaults to 30

processTrackerServiceInterval = <nonnegative integer>

- \* Controls how often, in seconds, indexer checks status of the child OS processes it had launched to see if it can launch new processes for queued requests.
- \* If set to 0, indexer will check child process status every second.
- \* Highest legal value is 4294967295
- \* Defaults to 15

maxBucketSizeCacheEntries = <nonnegative integer>

- \* This value is not longer needed and its value is ignored.

tsidxStatsHomePath = <path on server>

- \* An absolute path that specifies where Splunk creates namespace data with 'tscollect' command
- \* If the directory does not exist, we attempt to create it.
- \* Optional. If this is unspecified, we default to the 'tsidxstats' directory under \$SPLUNK\_DB

hotBucketTimeRefreshInterval = <positive integer>

- \* Controls how often each index refreshes the available hot bucket times used by the indexes REST endpoint.
- \* Refresh will occur every N times service is performed for each index.
  - \* For busy indexes, this is a multiple of seconds.
  - \* For idle indexes, this is a multiple of the second-long-periods in which data is received.
- \* This tunable is only intended to relax the frequency of these refreshes in the unexpected case that it adversely affects performance in unusual production scenarios.
- \* This time is tracked on a per-index basis, and thus can be adjusted on a per-index basis if needed.

\* If, for some reason, you want have the index information refreshed with every service (and accept minor performance overhead), you can use the value 1.

\* Defaults to 10 (services).

\*\*\*\*\*

## 每个索引选项

```
# PER INDEX OPTIONS
# These options may be set under an [<index>] entry.
#
# Index names must consist of only numbers, letters, periods, underscores,
# and hyphens.
#*****

disabled = true|false
* Toggles your index entry off and on.
* Set to true to disable an index.
* Defaults to false.

deleted = true
* If present, means that this index has been marked for deletion: if splunkd
  is running, deletion is in progress; if splunkd is stopped, deletion will
  re-commence on startup.
* Normally absent, hence no default.
* Do NOT manually set, clear, or modify value of this parameter.
* Seriously: LEAVE THIS PARAMETER ALONE.

homePath = <path on index server>
* An absolute path that contains the hotdb and warmdb for the index.
* Splunkd keeps a file handle open for warmdb at all times.
* May contain a volume reference (see volume section below).
* CAUTION: Path MUST be writable.
* Required. Splunk will not start if an index lacks a valid homePath.
* Must restart splunkd after changing this parameter; index reload will not
  suffice.

coldPath = <path on index server>
* An absolute path that contains the colddb for the index.
* Cold databases are opened as needed when searching.
* May contain a volume reference (see volume section below).
* CAUTION: Path MUST be writable.
* Required. Splunk will not start if an index lacks a valid coldPath.
* Must restart splunkd after changing this parameter; index reload will not
  suffice.

thawedPath = <path on index server>
* An absolute path that contains the thawed (resurrected) databases for the
  index.
* May NOT contain a volume reference.
* Required. Splunk will not start if an index lacks a valid thawedPath.
* Must restart splunkd after changing this parameter; index reload will not
  suffice.

bloomHomePath = <path on index server>
* Location where the bloomfilter files for the index are stored.
* If specified, MUST be defined in terms of a volume definition (see volume
  section below)
* If bloomHomePath is not specified, bloomfilter files for index will be
  stored inline, inside bucket directories.
* CAUTION: Path must be writable.
* Must restart splunkd after changing this parameter; index reload will not
  suffice.

createBloomfilter = true|false
* Controls whether to create bloomfilter files for the index.
* TRUE: bloomfilter files will be created. FALSE: not created.
* Defaults to true.

summaryHomePath = <path on index server>
```

- \* An absolute path where transparent summarization results for data in this index should be stored. Must be different for each index and may be on any disk drive.
- \* May contain a volume reference (see volume section below).
- \* Volume reference must be used if data retention based on data size is desired.
- \* If not specified, Splunk will use a directory 'summary' in the same location as homePath
  - \* For example, if homePath is "/opt/splunk/var/lib/splunk/index1/db", then summaryHomePath would be "/opt/splunk/var/lib/splunk/index1/summary".
- \* CAUTION: Path must be writable.
- \* Must restart splunkd after changing this parameter; index reload will not suffice.
- \* Defaults to unset.

tstatsHomePath = <path on index server>

- \* Required.
- \* Location where datamodel acceleration TSIDX data for this index should be stored
- \* MUST be defined in terms of a volume definition (see volume section below)
- \* Must restart splunkd after changing this parameter; index reload will not suffice.
- \* CAUTION: Path must be writable.
- \* Defaults to volume:\_splunk\_summaries/\$\_index\_name/datamodel\_summary, where \$\_index\_name is runtime-expanded to the name of the index

maxBloomBackfillBucketAge = <nonnegative integer>[smhd]|infinite

- \* If a (warm or cold) bloomfilter-less bucket is older than this, Splunk will not create a bloomfilter for that bucket.
- \* When set to 0, bloomfilters are never backfilled
- \* When set to "infinite", bloomfilters are always backfilled
- \* NB that if createBloomfilter=false, bloomfilters are never backfilled regardless of the value of this parameter
- \* Highest legal value in computed seconds is 2 billion, or 2000000000, which is approximately 68 years.
- \* Defaults to 30d.

enableOnlineBucketRepair = true|false

- \* Controls asynchronous "online fsck" bucket repair, which runs concurrently with Splunk
- \* When enabled, you do not have to wait until buckets are repaired, to start Splunk
- \* When enabled, you might observe a slight performance degradation
- \* Defaults to true.

enableDataIntegrityControl = true|false

- \* If set to true, hashes are computed on the rawdata slices and stored for future data integrity checks
- \* If set to false, no hashes are computed on the rawdata slices
- \* It has a global default value of false

# The following options can be set either per index or globally (as defaults # for all indexes). Defaults set globally are overridden if set on a # per-index basis.

maxWarmDBCount = <nonnegative integer>

- \* The maximum number of warm buckets.
- \* Warm buckets are located in the <homePath> for the index.
- \* If set to zero, Splunk will not retain any warm buckets (will roll them to cold as soon as it can)
- \* Highest legal value is 4294967295
- \* Defaults to 300.

maxTotalDataSizeMB = <nonnegative integer>

- \* The maximum size of an index (in MB).
- \* If an index grows larger than the maximum size, the oldest data is frozen.
- \* This parameter only applies to hot, warm, and cold buckets. It does not apply to thawed buckets.
- \* Highest legal value is 4294967295
- \* Defaults to 500000.

rotatePeriodInSecs = <positive integer>

- \* Controls the service period (in seconds): how often splunkd performs certain housekeeping tasks. Among these tasks are:
  - \* Check if a new hotdb needs to be created.
  - \* Check if there are any cold DBs that should be frozen.
  - \* Check whether buckets need to be moved out of hot and cold DBs, due to respective size constraints (i.e., homePath.maxDataSizeMB and coldPath.maxDataSizeMB)
- \* This value becomes the default value of the rotatePeriodInSecs attribute for all volumes (see rotatePeriodInSecs in the Volumes section)
- \* Highest legal value is 4294967295
- \* Defaults to 60.

frozenTimePeriodInSecs = <nonnegative integer>

- \* Number of seconds after which indexed data rolls to frozen.
- \* If you do not specify a coldToFrozenScript, data is deleted when rolled to frozen.
- \* IMPORTANT: Every event in the DB must be older than frozenTimePeriodInSecs before it will roll. Then, the DB will be frozen the next time splunkd checks (based on rotatePeriodInSecs attribute).
- \* Highest legal value is 4294967295
- \* Defaults to 188697600 (6 years).

warmToColdScript = <script path>

- \* Specifies a script to run when moving data from warm to cold.
- \* This attribute is supported for backwards compatibility with versions older than 4.0. Migrating data across filesystems is now handled natively by splunkd.
- \* If you specify a script here, the script becomes responsible for moving the event data, and Splunk-native data migration will not be used.
- \* The script must accept two arguments:
  - \* First: the warm directory (bucket) to be rolled to cold.
  - \* Second: the destination in the cold path.
- \* Searches and other activities are paused while the script is running.
- \* Contact Splunk Support ([http://www.splunk.com/page/submit\\_issue](http://www.splunk.com/page/submit_issue)) if you need help configuring this setting.
- \* The script must be in \$SPLUNK\_HOME/bin or a subdirectory thereof.
- \* Defaults to empty.

coldToFrozenScript = [path to script interpreter] <path to script>

- \* Specifies a script to run when data will leave the splunk index system.
  - \* Essentially, this implements any archival tasks before the data is deleted out of its default location.
- \* Add "\$DIR" (quotes included) to this setting on Windows (see below for details).
- \* Script Requirements:
  - \* The script must accept one argument:
    - \* An absolute path to the bucket directory to archive.
  - \* Your script should work reliably.
    - \* If your script returns success (0), Splunk will complete deleting the directory from the managed index location.
    - \* If your script return failure (non-zero), Splunk will leave the bucket in the index, and try calling your script again several minutes later.
    - \* If your script continues to return failure, this will eventually cause the index to grow to maximum configured size, or fill the disk.
  - \* Your script should complete in a reasonable amount of time.
    - \* If the script stalls indefinitely, it will occupy slots.
    - \* This script should not run for long as it would occupy resources which will affect indexing.
- \* If the string \$DIR is present in this setting, it will be expanded to the absolute path to the directory.
- \* If \$DIR is not present, the directory will be added to the end of the invocation line of the script.
- \* This is important for Windows.
  - \* For historical reasons, the entire string is broken up by shell-pattern expansion rules.
  - \* Since windows paths frequently include spaces, and the windows shell breaks on space, the quotes are needed for the script to understand the directory.
- \* If your script can be run directly on your platform, you can specify just the script.
  - \* Examples of this are:
    - \* .bat and .cmd files on Windows

- \* scripts set executable on UNIX with a `#!/` shebang line pointing to a valid interpreter.
- \* You can also specify an explicit path to an interpreter and the script.
  - \* Example: `/path/to/my/installation/of/python.exe path/to/my/script.py`
- \* Splunk ships with an example archiving script in that you SHOULD NOT USE `$SPLUNK_HOME/bin` called `coldToFrozenExample.py`
- \* DO NOT USE the example for production use, because:
  - \* 1 - It will be overwritten on upgrade.
  - \* 2 - You should be implementing whatever requirements you need in a script of your creation. If you have no such requirements, use `coldToFrozenDir`
- \* Example configuration:
  - \* If you create a script in `bin/` called `our_archival_script.py`, you could use:
    - UNIX:
 

```
coldToFrozenScript = "$SPLUNK_HOME/bin/python" "$SPLUNK_HOME/bin/our_archival_script.py"
```
    - Windows:
 

```
coldToFrozenScript = "$SPLUNK_HOME/bin/python" "$SPLUNK_HOME/bin/our_archival_script.py" "%DIR%"
```
- \* The example script handles data created by different versions of splunk differently. Specifically data from before 4.2 and after are handled differently. See "Freezing and Thawing" below:
- \* The script must be in `$SPLUNK_HOME/bin` or a subdirectory thereof.

`coldToFrozenDir` = <path to frozen archive>

- \* An alternative to a `coldToFrozen` script - simply specify a destination path for the frozen archive
- \* Splunk will automatically put frozen buckets in this directory
- \* For information on how buckets created by different versions are handled, see "Freezing and Thawing" below.
- \* If both `coldToFrozenDir` and `coldToFrozenScript` are specified, `coldToFrozenDir` will take precedence
- \* Must restart `splunkd` after changing this parameter; index reload will not suffice.
- \* May NOT contain a volume reference.

# Freezing and Thawing (this should move to web docs)

4.2 and later data:

- \* To archive: remove files except for the `rawdata` directory, since `rawdata` contains all the facts in the bucket.
- \* To restore: run `splunk rebuild <bucket_dir>` on the archived bucket, then atomically move the bucket to thawed for that index

4.1 and earlier data:

- \* To archive: gzip the `.tsidx` files, as they are highly compressible but cannot be recreated
- \* To restore: unpack the `tsidx` files within the bucket, then atomically move the bucket to thawed for that index

`compressRawdata` = `true|false`

- \* This parameter is ignored. The `splunkd` process always compresses raw data.

`maxConcurrentOptimizes` = <nonnegative integer>

- \* The number of concurrent optimize processes that can run against the hot DB.
- \* This number should be increased if:
  - \* There are always many small `tsidx` files in the hot DB.
  - \* After rolling, there are many `tsidx` files in warm or cold DB.
- \* Must restart `splunkd` after changing this parameter; index reload will not suffice.
- \* Highest legal value is 4294967295
- \* Defaults to 6

`maxDataSize` = <positive integer>|`auto`|`auto_high_volume`

- \* The maximum size in MB for a hot DB to reach before a roll to warm is triggered.
- \* Specifying `"auto"` or `"auto_high_volume"` will cause Splunk to autotune this parameter (recommended).
- \* You should use `"auto_high_volume"` for high-volume indexes (such as the main index); otherwise, use `"auto"`. A "high volume index" would typically be considered one that gets over 10GB of data per day.
- \* Defaults to `"auto"`, which sets the size to 750MB.
- \* `"auto_high_volume"` sets the size to 10GB on 64-bit, and 1GB on 32-bit systems.

- \* Although the maximum value you can set this is 1048576 MB, which corresponds to 1 TB, a reasonable number ranges anywhere from 100 to 50000. Before proceeding with any higher value, please seek approval of Splunk Support.
- \* If you specify an invalid number or string, maxDataSize will be auto tuned.
- \* NOTE: The maximum size of your warm buckets may slightly exceed 'maxDataSize', due to post-processing and timing issues with the rolling policy.

rawFileSizeBytes = <positive integer>

- \* Deprecated in version 4.2 and later. We will ignore this value.
- \* Rawdata chunks are no longer stored in individual files.
- \* If you really need to optimize the new rawdata chunks (highly unlikely), edit rawChunkSizeBytes

rawChunkSizeBytes = <positive integer>

- \* Target uncompressed size in bytes for individual raw slice in the rawdata journal of the index.
- \* If 0 is specified, rawChunkSizeBytes will be set to the default value.
- \* NOTE: rawChunkSizeBytes only specifies a target chunk size. The actual chunk size may be slightly larger by an amount proportional to an individual event size.
- \* WARNING: This is an advanced parameter. Only change it if you are instructed to do so by Splunk Support.
- \* Must restart splunkd after changing this parameter; index reload will not suffice.
- \* Highest legal value is 18446744073709551615
- \* Defaults to 131072 (128KB).

minRawFileSyncSecs = <nonnegative decimal>|disable

- \* How frequently we force a filesystem sync while compressing journal slices. During this interval, uncompressed slices are left on disk even after they are compressed. Then we force a filesystem sync of the compressed journal and remove the accumulated uncompressed files.
- \* If 0 is specified, we force a filesystem sync after every slice completes compressing.
- \* Specifying "disable" disables syncing entirely: uncompressed slices are removed as soon as compression is complete
- \* Some filesystems are very inefficient at performing sync operations, so only enable this if you are sure it is needed
- \* Must restart splunkd after changing this parameter; index reload will not suffice.
- \* No exponent may follow the decimal.
- \* Highest legal value is 18446744073709551615
- \* Defaults to "disable".

maxMemMB = <nonnegative integer>

- \* The amount of memory to allocate for indexing.
- \* This amount of memory will be allocated PER INDEX THREAD, or, if indexThreads is set to 0, once per index.
- \* IMPORTANT: Calculate this number carefully. splunkd will crash if you set this number higher than the amount of memory available.
- \* The default is recommended for all environments.
- \* Highest legal value is 4294967295
- \* Defaults to 5.

maxHotSpanSecs = <positive integer>

- \* Upper bound of timespan of hot/warm buckets in seconds.
- \* NOTE: If you set this too small, you can get an explosion of hot/warm buckets in the filesystem.
- \* If you set this parameter to less than 3600, it will be automatically reset to 3600, which will then activate snapping behavior (see below).
- \* This is an advanced parameter that should be set with care and understanding of the characteristics of your data.
- \* If set to 3600 (1 hour), or 86400 (1 day), becomes also the lower bound of hot bucket timespans. Further, snapping behavior (i.e. ohSnap) is activated, whereby hot bucket boundaries will be set at exactly the hour or day mark, relative to local midnight.
- \* Highest legal value is 4294967295
- \* Defaults to 7776000 seconds (90 days).
- \* Note that this limit will be applied per ingestion pipeline. For more



information about multiple ingestion pipelines see `parallelIngestionPipelines` in `server.conf.spec` file.

- \* With N parallel ingestion pipelines, each ingestion pipeline will write to and manage its own set of hot buckets, without taking into account the state of hot buckets managed by other ingestion pipelines. Each ingestion pipeline will independently apply this setting only to its own set of hot buckets.

`maxHotIdleSecs` = <nonnegative integer>

- \* Provides a ceiling for buckets to stay in hot status without receiving any data.
- \* If a hot bucket receives no data for more than `maxHotIdleSecs` seconds, Splunk rolls it to warm.
- \* This setting operates independently of `maxHotBuckets`, which can also cause hot buckets to roll.
- \* A value of 0 turns off the idle check (equivalent to infinite idle time).
- \* Highest legal value is 4294967295
- \* Defaults to 0.

`maxHotBuckets` = <positive integer>

- \* Maximum hot buckets that can exist per index.
- \* When `maxHotBuckets` is exceeded, Splunk rolls the least recently used (LRU) hot bucket to warm.
- \* Both normal hot buckets and quarantined hot buckets count towards this total.
- \* This setting operates independently of `maxHotIdleSecs`, which can also cause hot buckets to roll.
- \* Highest legal value is 4294967295
- \* Defaults to 3.
- \* Note that this limit will be applied per ingestion pipeline. For more information about multiple ingestion pipelines see `parallelIngestionPipelines` in `server.conf.spec` file.
- \* With N parallel ingestion pipelines the maximum number of hot buckets across all of the ingestion pipelines will be  $N * \text{maxHotBuckets}$  but `maxHotBuckets` for each ingestion pipeline. Each ingestion pipeline will independently write to and manage up to `maxHotBuckets` number of hot buckets. As a consequence of this, when multiple ingestion pipelines are used, there may be multiple (dependent on number of ingestion pipelines configured) hot buckets with events with overlapping time ranges.

`minHotIdleSecsBeforeForceRoll` = <nonnegative integer>|auto

- \* When there are no existing hot buckets that can fit new events because of their timestamps and the constraints on the index (refer to `maxHotBuckets`, `maxHotSpanSecs` and `quarantinePastSecs`), if any hot bucket has been idle (i.e. not receiving any data) for `minHotIdleSecsBeforeForceRoll` number of seconds, a new bucket will be created to receive these new events and the idle bucket will be rolled to warm.
- \* If no hot bucket has been idle for `minHotIdleSecsBeforeForceRoll` number of seconds, or if `minHotIdleSecsBeforeForceRoll` has been set to zero, then a best fit bucket will be chosen for these new events from the existing set of hot buckets.
- \* This setting operates independently of `maxHotIdleSecs`, which causes hot buckets to roll after they have been idle for `maxHotIdleSecs` number of seconds, \*regardless\* of whether new events can fit into the existing hot buckets or not due to an event timestamp. `minHotIdleSecsBeforeForceRoll`, on the other hand, controls a hot bucket roll \*only\* under the circumstances when the timestamp of a new event cannot fit into the existing hot buckets given the other parameter constraints on the system (parameters such as `maxHotBuckets`, `maxHotSpanSecs` and `quarantinePastSecs`).
- \* auto: Specifying "auto" will cause Splunk to autotune this parameter (recommended). The value begins at 600 seconds but automatically adjusts upwards for optimal performance. Specifically, the value will increase when a hot bucket rolls due to idle time with a significantly smaller size than `maxDataSize`. As a consequence, the outcome may be fewer buckets, though these buckets may span wider earliest-latest time ranges of events.
- \* 0: A value of 0 turns off the idle check (equivalent to infinite idle time). Setting this to zero means that we will never roll a hot bucket for the reason that an event cannot fit into an existing hot bucket due to the constraints of other parameters. Instead, we will find a best fitting bucket to accommodate that event.
- \* Highest legal value is 4294967295.
- \* NOTE: If you set this configuration, there is a chance that this could lead to frequent hot bucket rolls depending on the value. If your index contains a large number of buckets whose size-on-disk falls considerably short of the

size specified in `maxDataSize`, and if the reason for the roll of these buckets is due to "caller=lru", then setting the parameter value to a larger value or to zero may reduce the frequency of hot bucket rolls (see AUTO above). You may check `splunkd.log` for a similar message below for rolls due to this setting.

```
INFO HotBucketRoller - finished moving hot to warm bid=_internal~0~97597E05-7156-43E5-85B1-B0751462D16B
idx=_internal from=hot_v1_0 to=db_1462477093_1462477093_0 size=40960 caller=lru maxHotBuckets=3, count=4 hot
buckets,evicting_count=1 LRU hots
```

- \* Defaults to "auto".

`quarantinePastSecs` = <positive integer>

- \* Events with timestamp of `quarantinePastSecs` older than "now" will be dropped into quarantine bucket.
- \* This is a mechanism to prevent the main hot buckets from being polluted with fringe events.
- \* Highest legal value is 4294967295
- \* Defaults to 77760000 (900 days).

`quarantineFutureSecs` = <positive integer>

- \* Events with timestamp of `quarantineFutureSecs` newer than "now" will be dropped into quarantine bucket.
- \* This is a mechanism to prevent main hot buckets from being polluted with fringe events.
- \* Highest legal value is 4294967295
- \* Defaults to 2592000 (30 days).

`maxMetaEntries` = <nonnegative integer>

- \* Sets the maximum number of unique lines in .data files in a bucket, which may help to reduce memory consumption
- \* If exceeded, a hot bucket is rolled to prevent further increase
- \* If your buckets are rolling due to `Strings.data` hitting this limit, the culprit may be the 'punct' field in your data. If you do not use punct, it may be best to simply disable this (see `props.conf.spec`)
- \* NOTE: since at least 5.0.x, large strings.data from punct will be rare.
- \* There is a delta between when maximum is exceeded and bucket is rolled.
- \* This means a bucket may end up with epsilon more lines than specified, but this is not a major concern unless excess is significant
- \* If set to 0, this setting is ignored (it is treated as infinite)
- \* Highest legal value is 4294967295

`syncMeta` = true|false

- \* When "true", a sync operation is called before file descriptor is closed on metadata file updates.
- \* This functionality was introduced to improve integrity of metadata files, especially in regards to operating system crashes/machine failures.
- \* NOTE: Do not change this parameter without the input of a Splunk support professional.
- \* Must restart `splunkd` after changing this parameter; index reload will not suffice.
- \* Defaults to true.

`serviceMetaPeriod` = <positive integer>

- \* Defines how frequently metadata is synced to disk, in seconds.
- \* Defaults to 25 (seconds).
- \* You may want to set this to a higher value if the sum of your metadata file sizes is larger than many tens of megabytes, to avoid the hit on I/O in the indexing fast path.
- \* Highest legal value is 4294967295

`partialServiceMetaPeriod` = <positive integer>

- \* Related to `serviceMetaPeriod`. If set, it enables metadata sync every <integer> seconds, but only for records where the sync can be done efficiently in-place, without requiring a full re-write of the metadata file. Records that require full re-write will be synced at `serviceMetaPeriod`.
- \* <integer> specifies how frequently it should sync. Zero means that this feature is turned off and `serviceMetaPeriod` is the only time when metadata sync happens.
- \* If the value of `partialServiceMetaPeriod` is greater than `serviceMetaPeriod`, this setting will have no effect.
- \* By default it is turned off (zero).
- \* This parameter is ignored if `serviceOnlyAsNeeded` = true (the default).
- \* Highest legal value is 4294967295

throttleCheckPeriod = <positive integer>

- \* Defines how frequently Splunk checks for index throttling condition, in seconds.
- \* NOTE: Do not change this parameter without the input of a Splunk Support professional.
- \* Highest legal value is 4294967295
- \* Defaults to 15

maxTimeUnreplicatedWithAcks = <nonnegative decimal>

- \* Important if you have enabled indexer acknowledgements (ack) on forwarders and have replication enabled (via Index Clustering)
- \* This parameter puts an upper limit on how long events can sit unacknowledged in a raw slice
- \* To disable this, you can set to 0, but this is NOT recommended!!!
- \* NOTE: This is an advanced parameter; make sure you understand the settings on all your forwarders before changing this. This number should not exceed ack timeout configured on any forwarders, and should indeed be set to at most half of the minimum value of that timeout. You can find this setting in outputs.conf readTimeout setting, under the tcpout stanza.
- \* Highest legal value is 2147483647
- \* Defaults to 60 (seconds)

maxTimeUnreplicatedNoAcks = <nonnegative decimal>

- \* Important only if replication is enabled for this index, otherwise ignored
- \* This parameter puts an upper limit on how long an event can sit in raw slice.
- \* If there are any ack'd events sharing this raw slice, this parameter will not apply (maxTimeUnreplicatedWithAcks will be used instead)
- \* Highest legal value is 2147483647
- \* To disable this, you can set to 0; please be careful and understand the consequences before changing this parameter
- \* Defaults to 60 (seconds)

isReadOnly = true|false

- \* Set to true to make an index read-only.
- \* If true, no new events can be added to the index, but the index is still searchable.
- \* Must restart splunkd after changing this parameter; index reload will not suffice.
- \* Defaults to false.

homePath.maxDataSizeMB = <nonnegative integer>

- \* Specifies the maximum size of homePath (which contains hot and warm buckets).
- \* If this size is exceeded, Splunk will move buckets with the oldest value of latest time (for a given bucket) into the cold DB until homePath is below the maximum size.
- \* If this attribute is missing or set to 0, Splunk will not constrain the size of homePath.
- \* Highest legal value is 4294967295
- \* Defaults to 0.

coldPath.maxDataSizeMB = <nonnegative integer>

- \* Specifies the maximum size of coldPath (which contains cold buckets).
- \* If this size is exceeded, Splunk will freeze buckets with the oldest value of latest time (for a given bucket) until coldPath is below the maximum size.
- \* If this attribute is missing or set to 0, Splunk will not constrain size of coldPath
- \* If we freeze buckets due to enforcement of this policy parameter, and coldToFrozenScript and/or coldToFrozenDir archiving parameters are also set on the index, these parameters \*will\* take into effect
- \* Highest legal value is 4294967295
- \* Defaults to 0.

disableGlobalMetadata = true|false

- \* NOTE: This option was introduced in 4.3.3, but as of 5.0 it is obsolete and ignored if set.
- \* It used to disable writing to the global metadata. In 5.0 global metadata was removed.

```

repFactor = <nonnegative integer>|auto
* Only relevant if this instance is a clustering slave (but see note about
  "auto" below).
* See server.conf spec for details on clustering configuration.
* Value of 0 turns off replication for this index.
* If set to "auto", slave will use whatever value the master has.
* Highest legal value is 4294967295
* Defaults to 0.

minStreamGroupQueueSize = <nonnegative integer>
* Minimum size of the queue that stores events in memory before committing
  them to a tsidx file. As Splunk operates, it continually adjusts this
  size internally. Splunk could decide to use a small queue size and thus
  generate tiny tsidx files under certain unusual circumstances, such as
  file system errors. The danger of a very low minimum is that it can
  generate very tiny tsidx files with one or very few events, making it
  impossible for splunk-optimize to catch up and optimize the tsidx files
  into reasonably sized files.
* Defaults to 2000.
* Only set this value if you have been advised to by Splunk Support.
* Highest legal value is 4294967295

streamingTargetTsidxSyncPeriodMsec = <nonnegative integer>
* Period we force sync tsidx files on streaming targets. This setting is
  needed for multi-site clustering where streaming targets may be primary.
* if set to 0, we never sync (equivalent to infinity)

journalCompression = gzip|lz4
* Select compression algorithm for rawdata journal file
* Defaults to gzip

enableTsidxReduction = true|false
* By enabling this setting, you turn on the tsidx reduction capability. This causes the
  indexer to reduce the tsidx files of buckets, when the buckets reach the age specified
  by timePeriodInSecBeforeTsidxReduction.
* Defaults to false.

suspendHotRollByDeleteQuery = true|false
* When the "delete" search command is run, all buckets containing data to be deleted are
  marked for updating of their metadata files. The indexer normally first rolls any hot buckets,
  as rolling must precede the metadata file updates.
* When suspendHotRollByDeleteQuery is set to true, the rolling of hot buckets for the "delete"
  command is suspended. The hot buckets, although marked, do not roll immediately, but instead
  wait to roll in response to the same circumstances operative for any other hot buckets; for
  example, due to reaching a limit set by maxHotBuckets, maxDataSize, etc. When these hot buckets
  finally roll, their metadata files are then updated.
* Defaults to false

tsidxReductionCheckPeriodInSec = <positive integer>
* Time period between service runs to reduce the tsidx files for any buckets that have
  reached the age specified by timePeriodInSecBeforeTsidxReduction.
* Defaults to 600 (seconds).

timePeriodInSecBeforeTsidxReduction = <positive integer>
* Age at which buckets become eligible for tsidx reduction.
  The bucket age is the difference between the current time
  and the timestamp of the bucket's latest event.
* Defaults to 604800 (seconds).

*****

```

## 每个提供程序系列选项

```

# PER PROVIDER FAMILY OPTIONS
# A provider family is a way of collecting properties that are common to
# multiple providers. There are no properties that can only be used in a
# provider family, and not in a provider. If the same property is specified
# in a family, and in a provider belonging to that family, then the latter
# value "wins".
#

```

```
# All family stanzas begin with "provider-family:". For example:
# [provider-family:family_name]
# vix.mode=stream
# vix.command = java
# vix.command.arg.1 = -Xmx512m
# ...
#*****
#*****
```

## 每个提供程序选项

```
# PER PROVIDER OPTIONS
# These options affect External Resource Providers. All provider stanzas
# begin with "provider:". For example:
# [provider:provider_name]
# vix.family = hadoop
# vix.env.JAVA_HOME = /path/to/java/home
# vix.env.HADOOP_HOME = /path/to/hadoop/client/libraries
#
# Each virtual index must reference a provider.
#*****
vix.family = <family>
* A provider family to which this provider belongs.
* The only family available by default is "hadoop". Others may be added.

vix.mode = stream|report
* Usually specified at the family level.
* Typically should be "stream". In general, do not use "report" without
  consulting Splunk Support.

vix.command = <command>
* The command to be used to launch an external process for searches on this
  provider.
* Usually specified at the family level.

vix.command.arg.<N> = <argument>
* The Nth argument to the command specified by vix.command.
* Usually specified at the family level, but frequently overridden at the
  provider level, for example to change the jars used depending on the
  version of Hadoop to which a provider connects.

vix.<property name> = <property value>
* All such properties will be made available as "configuration properties" to
  search processes on this provider.
* For example, if this provider is in the Hadoop family, the configuration
  property "mapreduce.foo = bar" can be made available to the Hadoop
  via the property "vix.mapreduce.foo = bar".

vix.env.<env var name> = <env var variable>
* Will create an environment variable available to search processes on this
  provider.
* For example, to set the JAVA_HOME variable to "/path/java" for search
  processes on this provider, use "vix.env.JAVA_HOME = /path/java".

#*****
# PER PROVIDER OPTIONS -- HADOOP
# These options are specific to ERPs with the Hadoop family.
# NOTE: Many of these properties specify behavior if the property is not
# set. However, default values set in system/default/indexes.conf
# take precedence over the "unset" behavior.
#*****

vix.javaprops.<JVM system property name> = <value>
* All such properties will be used as Java system properties.
* For example, to specify a Kerberos realm (say "foo.com") as a Java
  system property, use the property
  "vix.javaprops.java.security.krb5.realm = foo.com".

vix.mapred.job.tracker = <logical name or server:port>
* In high-availability mode, use the logical name of the Job Tracker.
```

- \* Otherwise, should be set to server:port for the single Job Tracker.
- \* Note: this property is passed straight to Hadoop. Not all such properties are documented here.

vix.fs.default.name = <logical name or hdfs://server:port>

- \* In high-availability mode, use the logical name for a list of Name Nodes.
- \* Otherwise, use the URL for the single Name Node.
- \* Note: this property is passed straight to Hadoop. Not all such properties are documented here.

vix.splunk.setup.onsearch = true|false

- \* Whether to perform setup (install & bundle replication) on search.
- \* Defaults to false.

vix.splunk.setup.package = current|<path to file>

- \* Splunk .tgz package to install and use on data nodes (in vix.splunk.home.datanode).
- \* Uses the current install if set to value 'current' (without quotes).

vix.splunk.home.datanode = <path to dir>

- \* Path to where splunk should be installed on datanodes/tasktrackers, i.e. SPLUNK\_HOME.
- \* Required.

vix.splunk.home.hdfs = <path to dir>

- \* Scratch space for this Splunk instance on HDFS
- \* Required.

vix.splunk.search.debug = true|false

- \* Whether to run searches against this index in debug mode. In debug mode, additional information is logged to search.log.
- \* Optional. Defaults to false.

vix.splunk.search.recordreader = <list of classes>

- \* Comma separated list of data preprocessing classes.
- \* Each such class must extend BaseSplunkRecordReader and return data to be consumed by Splunk as the value.

vix.splunk.search.splitter = <class name>

- \* Set to override the class used to generate splits for MR jobs.
- \* Classes must implement com.splunk.mr.input.SplitGenerator.
- \* Unqualified classes will be assumed to be in the package com.splunk.mr.input.
- \* To search Parquet files, use ParquetSplitGenerator.
- \* To search Hive files, use HiveSplitGenerator.

vix.splunk.search.mr.threads = <postive integer>

- \* Number of threads to use when reading map results from HDFS
- \* Numbers less than 1 will be treated as 1.
- \* Numbers greater than 50 will be treated as 50.
- \* If not set, defaults to 10.

vix.splunk.search.mr.maxsplits = <positive integer>

- \* Maximum number of splits in an MR job.
- \* If not set, defaults to 10000.

vix.splunk.search.mr.minsplits = <positive integer>

- \* Number of splits for first MR job associated with a given search.
- \* If not set, defaults to 100.

vix.splunk.search.mr.splits.multiplier = <decimal greater than or equal to 1.0>

- \* Factor by which the number of splits is increased in consecutive MR jobs for a given search, up to the value of maxsplits.
- \* If not set, defaults to 10.

vix.splunk.search.mr.poll = <positive integer>

- \* Polling period for job status, in milliseconds.
- \* If not set, defaults to 1000 (ie. 1 second).

vix.splunk.search.mr.mapper.output.replication = <positive integer>

- \* Replication level for mapper output.
- \* Defaults to 3.

```

vix.splunk.search.mr.mapper.output.gzlevel = <integer between 0 and 9, inclusive>
* The compression level used for the mapper output.
* Defaults to 2.

vix.splunk.search.mixedmode = true|false
* Whether mixed mode execution is enabled.
* Defaults to true.

vix.splunk.search.mixedmode.maxstream = <nonnegative integer>
* Max # of bytes to stream during mixed mode.
* Value = 0 means there's no stream limit.
* Will stop streaming after the first split that took the value over the limit.
* If not set, defaults to 10 GB.

vix.splunk.jars = <list of paths>
* Comma delimited list of Splunk dirs/jars to add to the classpath in the
  Search Head and MR.

vix.env.HUNK_THIRDPARTY_JARS = <list of paths>
* Comma delimited list of 3rd-party dirs/jars to add to the classpath in the
  Search Head and MR.

vix.splunk.impersonation = true|false
* Enable/disable user impersonation.

vix.splunk.setup.bundle.replication = <positive integer>
* Set custom replication factor for bundles on HDFS.
* Must be an integer between 1 and 32767.
* Increasing this setting may help performance on large clusters by decreasing
  the average access time for a bundle across Task Nodes.
* Optional. If not set, the default replication factor for the file-system
  will apply.

vix.splunk.setup.bundle.max.inactive.wait = <positive integer>
* A positive integer represent a time interval in seconds.
* Defaults to 5.
* While a task waits for a bundle being replicated to the same node by another
  task, if the bundle file is not modified for this amount of time, the task
  will begin its own replication attempt.

vix.splunk.setup.bundle.poll.interval = <positive integer>
* A positive number, representing a time interval in milliseconds.
* Defaults to 100.
* While a task waits for a bundle to be installed by another task on the same
  node, it will check once per interval whether that installation is complete.

vix.splunk.setup.bundle.setup.timelimit = <positive integer>
* A positive number, representing a time duration in milliseconds.
* Defaults to 20,000 (i.e. 20 seconds).
* A task will wait this long for a bundle to be installed before it quits.

vix.splunk.setup.package.replication = true|false
* Set custom replication factor for the Splunk package on HDFS. This is the
  package set in the property vix.splunk.setup.package.
* Must be an integer between 1 and 32767.
* Increasing this setting may help performance on large clusters by decreasing
  the average access time for the package across Task Nodes.
* Optional. If not set, the default replication factor for the file-system
  will apply.

vix.splunk.setup.package.max.inactive.wait = <positive integer>
* A positive integer represent a time interval in seconds.
* Defaults to 5.
* While a task waits for a Splunk package being replicated to the same node by
  another task, if the package file is not modified for this amount of time,
  the task will begin its own replication attempt.

vix.splunk.setup.package.poll.interval = <positive integer>
* A positive number, representing a time interval in milliseconds.
* Defaults to 100.
* While a task waits for a Splunk package to be installed by another task on
  the same node, it will check once per interval whether that installation is

```

```

complete.

vix.splunk.setup.package.setup.timelimit = <positive integer>
* A positive number, representing a time duration in milliseconds.
* Defaults to 20,000 (i.e. 20 seconds).
* A task will wait this long for a Splunk package to be installed before it quits.

vix.splunk.search.column.filter = true|false
* Enables/disables column filtering. When enabled, Hunk will trim columns that
  are not necessary to a query on the Task Node, before returning the results
  to the search process.
* Should normally increase performance, but does have its own small overhead.
* Works with these formats: CSV, Avro, Parquet, Hive.
* If not set, defaults to true.

#
# Kerberos properties
#

vix.kerberos.principal = <kerberos principal name>
* Specifies principal for Kerberos authentication.
* Should be used with vix.kerberos.keytab and either
  1) vix.javaprops.java.security.krb5.realm and
     vix.javaprops.java.security.krb5.kdc, or
  2) security.krb5.conf

vix.kerberos.keytab = <kerberos keytab path>
* Specifies path to keytab for Kerberos authentication.
* See usage note with vix.kerberos.principal.

#
# The following properties affect the SplunkMR heartbeat mechanism. If this
# mechanism is turned on, the SplunkMR instance on the Search Head updates a
# heartbeat file on HDFS. Any MR job spawned by report or mix-mode searches
# checks the heartbeat file. If it is not updated for a certain time, it will
# consider SplunkMR to be dead and kill itself.
#

vix.splunk.heartbeat = true|false
* Turn on/off heartbeat update on search head, and checking on MR side.
* If not set, defaults to true.

vix.splunk.heartbeat.path = <path on HDFS>
* Path to heartbeat file.
* If not set, defaults to <vix.splunk.home.hdfs>/dispatch/<sid>/

vix.splunk.heartbeat.interval = <positive integer>
* Frequency with which the Heartbeat will be updated on the Search Head.
* Unit is millisecond.
* Default value is 6 seconds (6000).
* Minimum value is 1000. Smaller values will cause an exception to be thrown.

vix.splunk.heartbeat.threshold = <positive integer>
* The number of times the MR job will detect a missing heartbeat update before
  it considers SplunkMR dead and kills itself.
* Default value is 10.

## The following sections are specific to data input types.

#
# Sequence file
#

vix.splunk.search.recordreader.sequence.ignore.key = true|false
* When reading sequence files, if this key is enabled, events will be expected
  to only include a value. Otherwise, the expected representation is
  key+"\t"+value.
* Defaults to true.

#
# Avro

```



```

#

vix.splunk.search.recordreader.avro.regex = <regex>
* Regex that files must match in order to be considered avro files.
* Optional. Defaults to \.avro$

#
# Parquet
#

vix.splunk.search.splitter.parquet.simplifyresult = true|false
* If enabled, field names for map and list type fields will be simplified by
  dropping intermediate "map" or "element" subfield names. Otherwise, a field
  name will match parquet schema completely.
* Defaults to true.

#
# Hive
#

vix.splunk.search.splitter.hive.ppd = true|false
* Enable or disable Hive ORC Predicate Push Down.
* If enabled, ORC PPD will be applied whenever possible to prune unnecessary
  data as early as possible to optimize the search.
* If not set, defaults to true.

vix.splunk.search.splitter.hive.fileformat = textfile|sequencefile|rcfile|orc
* Format of the Hive data files in this provider.
* If not set, defaults to "textfile".

vix.splunk.search.splitter.hive.dbname = <DB name>
* Name of Hive database to be accessed by this provider.
* Optional. If not set, defaults to "default".

vix.splunk.search.splitter.hive.tablename = <table name>
* Table accessed by this provider.
* Required property.

vix.splunk.search.splitter.hive.columnnames = <list of column names>
* Comma-separated list of file names.
* Required if using Hive, not using metastore.

vix.splunk.search.splitter.hive.columntypes = string:float:int # COLON separated list of column types, required
* Colon-separated list of column- types.
* Required if using Hive, not using metastore.

vix.splunk.search.splitter.hive.serde = <SerDe class>
* Fully-qualified class name of SerDe.
* Required if using Hive, not using metastore, and if specified in creation of Hive table.

vix.splunk.search.splitter.hive.serde.properties = <list of key-value pairs>
* Comma-separated list of "key=value" pairs.
* Required if using Hive, not using metastore, and if specified in creation of Hive table.

vix.splunk.search.splitter.hive.fileformat.inputformat = <InputFormat class>
* Fully-qualified class name of an InputFormat to be used with Hive table data.

vix.splunk.search.splitter.hive.rowformat.fields.terminated = <delimiter>
* Will be set as the Hive SerDe property "field.delim".
* Optional.

vix.splunk.search.splitter.hive.rowformat.escaped = <escape char>
* Will be set as the Hive SerDe property "escape.delim".
* Optional.

vix.splunk.search.splitter.hive.rowformat.lines.terminated = <delimiter>
* Will be set as the Hive SerDe property "line.delim".
* Optional.

vix.splunk.search.splitter.hive.rowformat.mapkeys.terminated = <delimiter>
* Will be set as the Hive SerDe property "mapkey.delim".
* Optional.

```

```
vix.splunk.search.splitter.hive.rowformat.collectionitems.terminated = <delimiter>
* Will be set as the Hive SerDe property "collection.delim".
* Optional.

#
# Archiving
#

vix.output.buckets.max.network.bandwidth = 0|<bits per second>
* Throttles network bandwidth to <bits per second>
* Defaults to 0, meaning no throttling.
* Set at provider level. Applied to all virtual indexes using a provider with this setting.

*****
```

## 每个虚拟索引选项

```
# PER VIRTUAL INDEX OPTIONS
# These options affect virtual indexes. Like indexes, these options may
# be set under an [<virtual-index>] entry.
#
# Virtual index names have the same constraints as normal index names.
#
# Each virtual index must reference a provider. I.e:
# [virtual_index_name]
# vix.provider = <provider_name>
#
# All configuration keys starting with "vix." will be passed to the
# external resource provider (ERP).
*****

vix.provider = <provider_name>
* Name of the external resource provider to use for this virtual index.

*****
# PER VIRTUAL INDEX OPTIONS -- HADOOP
# These options are specific to ERPs with the Hadoop family.
*****

#
# The vix.input.* configurations are grouped by an id.
# Inputs configured via the UI always use '1' as the id.
# In this spec we'll use 'x' as the id.
#

vix.input.x.path = <path>
* Path in a hadoop filesystem (usually HDFS or S3).
* May contain wildcards.
* Checks the path for data recursively when ending with '...'
* Can extract fields with ${field}. I.e: "/data/${server}/...", where server
  will be extracted.
* May start with a schema.
  * The schema of the path specifies which hadoop filesystem implementation to
    use. Examples:
    * hdfs://foo:1234/path, will use a HDFS filesystem implementation
    * s3a://s3-bucket/path, will use a S3 filesystem implementation

vix.input.x.accept = <regex>
* Specifies a whitelist regex.
* Only files within the location given by matching vix.input.x.path, whose
  paths match this regex, will be searched.

vix.input.x.ignore = <regex>
* Specifies a blacklist regex.
* Searches will ignore paths matching this regex.
* These matches take precedence over vix.input.x.accept matches.

vix.input.x.required.fields = <comma separated list of fields>
* Fields that will be kept in search results even if the field is not required by the search
```

```

# Earliest time extractions - For all 'et' settings, there's an equivalent 'lt' setting.
vix.input.x.et.regex = <regex>
* Regex extracting earliest time from vix.input.x.path

vix.input.x.et.format = <java.text.SimpleDateFormat date pattern>
* Format of the extracted earliest time.
* See documentation for java.text.SimpleDateFormat

vix.input.x.et.offset = <seconds>
* Offset in seconds to add to the extracted earliest time.

vix.input.x.et.timezone = <java.util.SimpleTimeZone timezone id>
* Timezone in which to interpret the extracted earliest time.
* Examples: "America/Los_Angeles" or "GMT-8:00"

vix.input.x.et.value = mtime|<epoch time in milliseconds>
* Sets the earliest time for this virtual index.
* Can be used instead of extracting times from the path via vix.input.x.et.regex
* When set to "mtime", uses the file modification time as the earliest time.

# Latest time extractions - See "Earliest time extractions"

vix.input.x.lt.regex = <regex>
* Latest time equivalent of vix.input.x.et.regex

vix.input.x.lt.format = <java.text.SimpleDateFormat date pattern>
* Latest time equivalent of vix.input.x.et.format

vix.input.x.lt.offset = <seconds>
* Latest time equivalent of vix.input.x.et.offset

vix.input.x.lt.timezone = <java.util.SimpleTimeZone timezone id>
* Latest time equivalent of vix.input.x.et.timezone

vix.input.x.lt.value = <mod time>
* Latest time equivalent of vix.input.x.et.value

#
# Archiving
#

vix.output.buckets.path = <hadoop path>
* Path to a hadoop filesystem where buckets will be archived

vix.output.buckets.older.than = <seconds>
* Buckets must be this old before they will be archived.
* A bucket's age is determined by the the earliest _time field of any event in
  the bucket.

vix.output.buckets.from.indexes = <comma separated list of splunk indexes>
* List of (non-virtual) indexes that will get archived to this (virtual) index.

vix.unified.search.cutoff_sec = <seconds>
* Window length before present time that configures where events are retrieved
  for unified search
* Events from now to now-cutoff_sec will be retrieved from the splunk index
  and events older than cutoff_sec will be retrieved from the archive index

*****
# PER VIRTUAL INDEX OR PROVIDER OPTIONS -- HADOOP
# These options can be set at either the virtual index level or provider
# level, for the Hadoop ERP.
#
# Options set at the virtual index level take precedence over options set
# at the provider level.
#
# Virtual index level prefix:
# vix.input.<input_id>.<option_suffix>
#
# Provider level prefix:
# vix.splunk.search.<option_suffix>
*****

```

```

# The following options are just defined by their <option_suffix>

#
# Record reader options
#

recordreader.<name>.<conf_key> = <conf_value>
* Sets a configuration key for a RecordReader with <name> to <conf_value>

recordreader.<name>.regex = <regex>
* Regex specifying which files this RecordReader can be used for.

recordreader.journal.buffer.size = <bytes>
* Buffer size used by the journal record reader

recordreader.csv.dialect = default|excel|excel-tab|tsv
* Set the csv dialect for csv files
* A csv dialect differs on delimiter_char, quote_char and escape_char.
* Here is a list of how the different dialects are defined in order delim,
  quote, and escape:
  * default    = , " \
  * excel      = , " "
  * excel-tab  = \t " "
  * tsv        = \t " \

#
# Splitter options
#

splitter.<name>.<conf_key> = <conf_value>
* Sets a configuration key for a split generator with <name> to <conf_value>

splitter.file.split.minsize = <bytes>
* Minimum size in bytes for file splits.
* Defaults to 1.

splitter.file.split.maxsize = <bytes>
* Maximum size in bytes for file splits.
* Defaults to Long.MAX_VALUE.

*****
# Volume settings. This section describes settings that affect the volume-
# optional and volume-mandatory parameters only.
#
# All volume stanzas begin with "volume:". For example:
# [volume:volume_name]
# path = /foo/bar
#
# These volume stanzas can then be referenced by individual index
# parameters, e.g. homePath or coldPath. To refer to a volume stanza, use
# the "volume:" prefix. For example, to set a cold DB to the example stanza
# above, in index "hiro", use:
# [hiro]
# coldPath = volume:volume_name/baz
# This will cause the cold DB files to be placed under /foo/bar/baz. If the
# volume spec is not followed by a path
# (e.g. "coldPath=volume:volume_name"), then the cold path would be
# composed by appending the index name to the volume name ("/foo/bar/hiro").
#
# Note: thawedPath may not be defined in terms of a volume.
# Thawed allocations are manually controlled by Splunk administrators,
# typically in recovery or archival/review scenarios, and should not
# trigger changes in space automatically used by normal index activity.
*****

path = <path on server>
* Required.
* Points to the location on the file system where all databases that use
  this volume will reside. You must make sure that this location does not
  overlap with that of any other volume or index database.

```

```

maxVolumeDataSizeMB = <positive integer>
* Optional.
* If set, this attribute limits the total size of all databases that reside
  on this volume to the maximum size specified, in MB. Note that this it
  will act only on those indexes which reference this volume, not on the
  total size of the path set in the path attribute of this volume.
* If the size is exceeded, Splunk will remove buckets with the oldest value
  of latest time (for a given bucket) across all indexes in the volume,
  until the volume is below the maximum size. This is the trim operation.
  Note that this can cause buckets to be chilled [moved to cold] directly
  from a hot DB, if those buckets happen to have the least value of
  latest-time (LT) across all indexes in the volume.
* Highest legal value is 4294967295, lowest legal value is 1.

rotatePeriodInSecs = <nonnegative integer>
* Optional.
* Specifies period of trim operation for this volume.
* If not set, the value of global rotatePeriodInSecs attribute is inherited.
* Highest legal value is 4294967295

```

## indexes.conf.example

```

# Version 6.5.0
#
# This file contains an example indexes.conf. Use this file to configure
# indexing properties.
#
# To use one or more of these configurations, copy the configuration block
# into indexes.conf in $SPLUNK_HOME/etc/system/local/. You must restart
# Splunk to enable configurations.
#
# To learn more about configuration files (including precedence) please see
# the documentation located at
# http://docs.splunk.com/Documentation/Splunk/latest/Admin/Aboutconfigurationfiles
#
# The following example defines a new high-volume index, called "hatch", and
# sets this to be the default index for both incoming data and search.
#
# Note that you may want to adjust the indexes that your roles have access
# to when creating indexes (in authorize.conf)

defaultDatabase = hatch

[hatch]

homePath    = $SPLUNK_DB/hatchdb/db
coldPath    = $SPLUNK_DB/hatchdb/colddb
thawedPath  = $SPLUNK_DB/hatchdb/thaweddb
maxDataSize = 10000
maxHotBuckets = 10

# The following example changes the default amount of space used on a
# per-index basis.

[default]
maxTotalDataSizeMB = 650000

# The following example changes the time data is kept around by default.
# It also sets an export script. NOTE: You must edit this script to set
# export location before running it.

[default]
maxWarmDBCount = 200
frozenTimePeriodInSecs = 432000
rotatePeriodInSecs = 30
coldToFrozenScript = "$SPLUNK_HOME/bin/python" "$SPLUNK_HOME/bin/myColdToFrozenScript.py"

```

```

# This example freezes buckets on the same schedule, but lets Splunk do the
# freezing process as opposed to a script
[default]
maxWarmDBCount = 200
frozenTimePeriodInSecs = 432000
rotatePeriodInSecs = 30
coldToFrozenDir = "$SPLUNK_HOME/myfrozenarchive"

### This example demonstrates the use of volumes ###

# volume definitions; prefixed with "volume:"

[volume:hot1]
path = /mnt/fast_disk
maxVolumeDataSizeMB = 100000

[volume:cold1]
path = /mnt/big_disk
# maxVolumeDataSizeMB not specified: no data size limitation on top of the
# existing ones

[volume:cold2]
path = /mnt/big_disk2
maxVolumeDataSizeMB = 1000000

# index definitions

[idx1]
homePath = volume:hot1/idx1
coldPath = volume:cold1/idx1

# thawedPath must be specified, and cannot use volume: syntax
# choose a location convenient for reconstitution from archive goals
# For many sites, this may never be used.
thawedPath = $SPLUNK_DB/idx1/thaweddb

[idx2]
# note that the specific indexes must take care to avoid collisions
homePath = volume:hot1/idx2
coldPath = volume:cold2/idx2
thawedPath = $SPLUNK_DB/idx2/thaweddb

[idx3]
homePath = volume:hot1/idx3
coldPath = volume:cold2/idx3
thawedPath = $SPLUNK_DB/idx3/thaweddb

### Indexes may be allocated space in effective groups by sharing volumes ###

# perhaps we only want to keep 100GB of summary data and other
# low-volume information
[volume:small_indexes]
path = /mnt/splunk_indexes
maxVolumeDataSizeMB = 100000

# and this is our main event series, allowing 50 terabytes
[volume:large_indexes]
path = /mnt/splunk_indexes
maxVolumeDataSizeMB = 50000000

# summary and rare_data together will be limited to 100GB
[summary]
homePath=volume:small_indexes/summary/db
coldPath=volume:small_indexes/summary/colddb
thawedPath=$SPLUNK_DB/summary/thaweddb
# low-volume indexes probably don't want a lot of hot buckets
maxHotBuckets = 2
# if the volume is quite low, and you have data sunset goals you may
# want to have smaller buckets
maxDataSize = 500

```

```

[rare_data]
homePath=volume:small_indexes/rare_data/db
coldPath=volume:small_indexes/rare_data/coldddb
thawedPath=$SPLUNK_DB/rare_data/thaweddb
maxHotBuckets = 2

# main, and any other large volume indexes you add sharing large_indexes
# will be together be constrained to 50TB, separately from the 100GB of
# the small_indexes
[main]
homePath=volume:large_indexes/main/db
coldPath=volume:large_indexes/main/coldddb
thawedPath=$SPLUNK_DB/main/thaweddb
# large buckets and more hot buckets are desirable for higher volume
# indexes, and ones where the variations in the timestream of events is
# hard to predict.
maxDataSize = auto_high_volume
maxHotBuckets = 10

[idx1_large_vol]
homePath=volume:large_indexes/idx1_large_vol/db
coldPath=volume:large_indexes/idx1_large_vol/coldddb
homePath=$SPLUNK_DB/idx1_large/thaweddb
# this index will exceed the default of .5TB requiring a change to maxTotalDataSizeMB
maxTotalDataSizeMB = 750000
maxDataSize = auto_high_volume
maxHotBuckets = 10
# but the data will only be retained for about 30 days
frozenTimePeriodInSecs = 2592000

### This example demonstrates database size constraining ###

# In this example per-database constraint is combined with volumes. While a
# central volume setting makes it easy to manage data size across multiple
# indexes, there is a concern that bursts of data in one index may
# significantly displace data from others. The homePath.maxDataSizeMB setting
# can be used to assure that no index will ever take more than certain size,
# therefore alleviating the concern.

# global settings

# will be inherited by all indexes: no database will exceed 1TB
homePath.maxDataSizeMB = 1000000

# volumes

[volume:caliente]
path = /mnt/fast_disk
maxVolumeDataSizeMB = 100000

[volume:frio]
path = /mnt/big_disk
maxVolumeDataSizeMB = 1000000

# and this is our main event series, allowing about 50 terabytes
[volume:large_indexes]
path = /mnt/splunk_indexes
maxVolumeDataSizeMB = 50000000

# indexes

[i1]
homePath = volume:caliente/i1
# homePath.maxDataSizeMB is inherited
coldPath = volume:frio/i1
# coldPath.maxDataSizeMB not specified: no limit - old-style behavior

thawedPath = $SPLUNK_DB/i1/thaweddb

[i2]
homePath = volume:caliente/i2

```

```

# overrides the default maxDataSize
homePath.maxDataSizeMB = 1000
coldPath = volume:frio/i2
# limits the cold DB's
coldPath.maxDataSizeMB = 10000
thawedPath = $SPLUNK_DB/i2/thaweddb

[i3]
homePath = /old/style/path
homePath.maxDataSizeMB = 1000
coldPath = volume:frio/i3
coldPath.maxDataSizeMB = 10000
thawedPath = $SPLUNK_DB/i3/thaweddb

# main, and any other large volume indexes you add sharing large_indexes
# will together be constrained to 50TB, separately from the rest of
# the indexes
[main]
homePath=volume:large_indexes/main/db
coldPath=volume:large_indexes/main/colddb
thawedPath=$SPLUNK_DB/main/thaweddb
# large buckets and more hot buckets are desirable for higher volume indexes
maxDataSize = auto_high_volume
maxHotBuckets = 10

```

## inputs.conf

以下为 inputs.conf 的规范和示例文件。

### inputs.conf.spec

```

# Version 6.5.0

# This file contains possible settings you can use to configure inputs,
# distributed inputs such as forwarders, and file system monitoring in
# inputs.conf.
#
# There is an inputs.conf in $SPLUNK_HOME/etc/system/default/. To set custom
# configurations, place an inputs.conf in $SPLUNK_HOME/etc/system/local/. For
# examples, see inputs.conf.example. You must restart Splunk to enable new
# configurations.
#
# To learn more about configuration files (including precedence), see the
# documentation located at
# http://docs.splunk.com/Documentation/Splunk/latest/Admin/Aboutconfigurationfiles
#

```

### 全局设置

```

# GLOBAL SETTINGS
# Use the [default] stanza to define any global settings.
# * You can also define global settings outside of any stanza, at the top of
# the file.
# * Each conf file should have at most one default stanza. If there are
# multiple default stanzas, settings are combined. In the case of
# multiple definitions of the same setting, the last definition in the
# file wins.
# * If an setting is defined at both the global level and in a specific
# stanza, the value in the specific stanza takes precedence.

#*****
# GENERAL SETTINGS:
# The following settings are valid for all input types (except file system
# change monitor, which is described in a separate section in this file).
# You must first enter a stanza header in square brackets, specifying the input
# type. See further down in this file for examples.

```



```
# Then, use any of the following settings.
#*****

host = <string>
* Sets the host key/field to a static value for this stanza.
* Primarily used to control the host field, which the input applies to events
  that come in through this input stanza.
* Detail: Sets the host key initial value. The input uses this key during
  parsing/indexing, in particular to set the host field. It also uses this
  field at search time.
* As a convenience, the input prepends the chosen string with 'host::'.
* WARNING: Do not put the <string> value in quotes. Use host=foo, not host="foo".
* If set to '$decideOnStartup', will be interpreted as hostname of executing
  machine; this will occur on each splunkd startup.
* If you run multiple instances of the software on the same system (hardware
  or virtual machine), choose unique values for 'host' to differentiate
  your data, e.g. myhost-sh-1 or myhost-idx-2.
* The literal default conf value is $decideOnStartup, but at installation
  time, the setup logic adds the local hostname as determined by DNS to the
  $SPLUNK_HOME/etc/system/local/inputs.conf default stanza, which is the
  effective default value.

index = <string>
* Sets the index to store events from this input.
* Primarily used to specify the index to store events coming in via this input
  stanza.
* Detail: Sets the index key's initial value. The key is used when selecting an
  index to store the events.
* Defaults to "main" (or whatever you have set as your default index).

source = <string>
* Sets the source key/field for events from this input.
* NOTE: Overriding the source key is generally not recommended. Typically, the
  input layer will provide a more accurate string to aid problem
  analysis and investigation, accurately recording the file from which the data
  was retrieved. Please consider use of source types, tagging, and search
  wildcards before overriding this value.
* Detail: Sets the source key's initial value. The key is used during
  parsing/indexing, in particular to set the source field during
  indexing. It is also the source field used at search time.
* As a convenience, the chosen string is prepended with 'source::'.
* WARNING: Do not quote the <string> value: source=foo, not source="foo".
* Defaults to the input file path.

sourcetype = <string>
* Sets the sourcetype key/field for events from this input.
* Primarily used to explicitly declare the source type for this data, as
  opposed to allowing it to be determined via automated methods. This is
  typically important both for searchability and for applying the relevant
  configuration for this type of data during parsing and indexing.
* Detail: Sets the sourcetype key's initial value. The key is used during
  parsing/indexing, in particular to set the source type field during
  indexing. It is also the source type field used at search time.
* As a convenience, the chosen string is prepended with 'sourcetype::'.
* WARNING: Do not quote the <string> value: sourcetype=foo, not sourcetype="foo".
* If unset, Splunk picks a source type based on various aspects of the data.
  There is no hard-coded default.

queue = [parsingQueue|indexQueue]
* Specifies where the input processor should deposit the events it reads.
* Set queue to "parsingQueue" to apply props.conf and other parsing rules to
  your data. For more information about props.conf and rules for timestamping
  and linebreaking, refer to props.conf and the online documentation at
  http://docs.splunk.com/Documentation.
* Set queue to "indexQueue" to send your data directly into the index.
* Defaults to parsingQueue.

# Pipeline Key defaulting.

* Pipeline keys in general can be defaulted in inputs stanzas.
* The list of user-available modifiable pipeline keys is described in
  transforms.conf.spec; see transforms.conf.spec for further information on
```

these keys.

- \* The currently-defined keys which are available literally in inputs stanzas are as follows:

```
queue = <value>
_raw  = <value>
_meta = <value>
_time = <value>
```

- \* Inputs have special support for mapping host, source, sourcetype, and index to their metadata names such as host -> Metadata:Host
- \* Defaulting these values is not recommended, and is generally only useful as a workaround to other product issues.
- \* Defaulting these keys in most cases will override the default behavior of input processors; but this behavior is not guaranteed in all cases.
- \* Values defaulted here, as with all values provided by inputs, can be altered by transforms at parse-time.

```
# *****
# This section contains options for routing data using inputs.conf rather than
# outputs.conf.
# Note concerning routing via inputs.conf:
# This is a simplified set of routing options you can use as data comes in.
# For more flexible options or details on configuring required or optional
# settings, see outputs.conf.spec.
```

```
_TCP_ROUTING = <tcpout_group_name>,<tcpout_group_name>,<tcpout_group_name>, ...
```

- \* Comma-separated list of tcpout group names.
- \* Using this, you can selectively forward the data to specific indexer(s).
- \* Specify the tcpout group the forwarder should use when forwarding the data. The tcpout group names are defined in outputs.conf with [tcpout:<tcpout\_group\_name>].
- \* Defaults to groups specified in "defaultGroup" in [tcpout] stanza in outputs.conf.
- \* To forward data from the "\_internal" index, \_TCP\_ROUTING must explicitly be set to either "\*" or a specific splunktcp target group.

```
_SYSLOG_ROUTING = <syslog_group_name>,<syslog_group_name>,<syslog_group_name>, ...
```

- \* Comma-separated list of syslog group names.
- \* Using this, you can selectively forward the data to specific destinations as syslog events.
- \* Specify the syslog group to use when forwarding the data. The syslog group names are defined in outputs.conf with [syslog:<syslog\_group\_name>].
- \* Defaults to groups present in "defaultGroup" in [syslog] stanza in outputs.conf.
- \* The destination host must be configured in outputs.conf, using "server=[<ip>|<servername>]:<port>".

```
_INDEX_AND_FORWARD_ROUTING = <string>
```

- \* Only has effect if using selectiveIndexing feature in outputs.conf.
- \* If set for any input stanza, should cause all data coming from that input stanza to be labeled with this setting.
- \* When selectiveIndexing is in use on a forwarder:
  - \* data without this label will not be indexed by that forwarder.
  - \* data with this label will be indexed in addition to any forwarding.
- \* This setting does not actually cause data to be forwarded or not forwarded in any way, nor does it control where the data is forwarded in multiple-forward path cases.
- \* Defaults to not present.

## 黑名单

```
#*****
# Blacklist
#*****Blacklist
```

```
[blacklist:<path>]
```

- \* Protect files on the file system from being indexed or previewed.
- \* The input treats a file as blacklisted if the file starts with any of the defined blacklisted <paths>.

- \* The preview endpoint will return an error when asked to preview a blacklisted file.
- \* The oneshot endpoint and command will also return an error.
- \* When a blacklisted file is monitored (monitor:// or batch://), filestatus endpoint will show an error.
- \* For fschange with the 'sendFullEvent' option enabled, contents of blacklisted files will not be indexed.

### 后跟有效的输入类型和特定于输入的设置：

```
#*****
# Valid input types follow, along with their input-specific settings:
#*****Valid input types follow, along with their input-specific settings:
```

### 监视器：

```
#*****
# MONITOR:
#*****MONITOR:

[monitor://<path>]
* This directs a file monitor input to watch all files in <path>.
* <path> can be an entire directory or a single file.
* You must specify the input type and then the path, so put three slashes in
  your path if you are starting at the root on *nix systems (to include the
  slash that indicates an absolute path).

# Additional settings:

host_regex = <regular expression>
* If specified, <regular expression> extracts host from the path to the file
  for each input file.
  * Detail: This feature examines the source key; if source is set
    explicitly in the stanza, that string will be matched, not the original
    filename.
* Specifically, the first group of the regex is used as the host.
* If the regex fails to match, the default "host =" setting is used.
* If host_regex and host_segment are both set, the input ignores host_regex.
* Defaults to unset.

host_segment = <integer>
* If set to N, the Nth "/"-separated segment of the path is set as host. If
  host_segment=3, for example, the third segment is used.
* If the value is not an integer or is less than 1, the default "host ="
  setting is used.
* Defaults to unset.

whitelist = <regular expression>
* If set, files from this input are monitored only if their path matches the
  specified regex.
* Takes precedence over the deprecated _whitelist setting, which functions
  the same way.

blacklist = <regular expression>
* If set, files from this input are NOT monitored if their path matches the
  specified regex.
* Takes precedence over the deprecated _blacklist setting, which functions
  the same way.

Note concerning wildcards and monitor:
* You can use wildcards to specify your input path for monitored input. Use
  "..." for recursive directory matching and "*" for wildcard matching in a
  single directory segment.
* "..." recurses through directories. This means that /foo/.../bar will match
  foo/bar, foo/1/bar, foo/1/2/bar, etc.
* You can use multiple "..." specifications in a single input path. For
  example: /foo/.../bar/...
```

- \* The asterisk (\*) matches anything in a single path segment; unlike "...", it does not recurse. For example, /foo/\*/bar matches the files /foo/bar, /foo/1/bar, /foo/2/bar, etc. However, it does not match /foo/1/2/bar. A second example: /foo/m\*r/bar matches /foo/mr/bar, /foo/mir/bar, /foo/moor/bar, etc.
- \* You can combine "\*" and "." as needed: foo/.../bar/\* matches any file in the bar directory within the specified path.

crcSalt = <string>

- \* Use this setting to force the input to consume files that have matching CRCs (cyclic redundancy checks).
  - \* (The input only performs CRC checks against, by default, the first 256 bytes of a file. This behavior prevents the input from indexing the same file twice, even though you may have renamed it -- as, for example, with rolling log files. However, because the CRC is based on only the first few lines of the file, it is possible for legitimately different files to have matching CRCs, particularly if they have identical headers.)
- \* If set, <string> is added to the CRC.
- \* If set to the literal string <SOURCE> (including the angle brackets), the full directory path to the source file is added to the CRC. This ensures that each file being monitored has a unique CRC. When crcSalt is invoked, it is usually set to <SOURCE>.
- \* Be cautious about using this setting with rolling log files; it could lead to the log file being re-indexed after it has rolled.
- \* In many situations, initCrcLength can be used to achieve the same goals.
- \* Defaults to empty.

initCrcLength = <integer>

- \* This setting adjusts how much of a file the input reads before trying to identify whether it is a file that has already been seen. You might want to adjust this if you have many files with common headers (comment headers, long CSV headers, etc) and recurring filenames.
- \* CAUTION: Improper use of this setting will cause data to be re-indexed. You might want to consult with Splunk Support before adjusting this value - the default is fine for most installations.
- \* Defaults to 256 (bytes).
- \* Must be in the range 256-1048576.

ignoreOlderThan = <nonnegative integer>[s|m|h|d]

- \* The monitor input will compare the modification time on files it encounters with the current time. If the time elapsed since the modification time is greater than this setting, it will be placed on the ignore list.
- \* Files placed on the ignore list will not be checked again for any reason until the Splunk software restarts, or the file monitoring subsystem is reconfigured. This is true even if the file becomes newer again at a later time.
  - \* Reconfigurations occur when changes are made to monitor or batch inputs via the UI or command line.
- \* Use IgnoreOlderThan to increase file monitoring performance when monitoring a directory hierarchy containing many unchanging older files, and when removing or blacklisting those files from the monitoring location is not a reasonable option.
- \* Do NOT select a time that files you want to read could reach in age, even temporarily. Take potential downtime into consideration!
  - \* Suggested value: 14d, which means 2 weeks
  - \* For example, a time window in significant numbers of days or small numbers of weeks are probably reasonable choices.
  - \* If you need a time window in small numbers of days or hours, there are other approaches to consider for performant monitoring beyond the scope of this one setting.
- \* NOTE: Most modern Windows file access APIs do not update file modification time while the file is open and being actively written to. Windows delays updating modification time until the file is closed. Therefore you might have to choose a larger time window on Windows hosts where files may be open for long time periods.
- \* Value must be: <number><unit>. For example, "7d" indicates one week.
- \* Valid units are "d" (days), "h" (hours), "m" (minutes), and "s" (seconds).
- \* Defaults to unset, meaning there is no threshold and no files are ignored for modification time reasons.

followTail = [0|1]

- \* WARNING: Use of followTail should be considered an advanced administrative action.
- \* Treat this setting as an 'action':
  - \* Enable this setting and start the Splunk software.
  - \* Wait enough time for the input to identify the related files.
  - \* Disable the setting and restart.
- \* DO NOT leave followTail enabled in an ongoing fashion.
- \* Do not use followTail for rolling log files (log files that get renamed as they age), or files whose names or paths vary.
- \* You can use this to force the input to skip past all current data for a given stanza.
  - \* In more detail: this is intended to mean that if you start the monitor with a stanza configured this way, all data in the file at the time it is first encountered will not be read. Only data that arrives after the first encounter time will be read.
  - \* This can be used to "skip over" data from old log files, or old portions of log files, to get started on current data right away.
- \* If set to 1, monitoring starts at the end of the file (like tail -f).
- \* If set to 0, monitoring starts at the beginning of the file.
- \* Defaults to 0.

alwaysOpenFile = [0|1]

- \* Opens a file to check whether it has already been indexed, by skipping the modification time/size checks.
- \* Only useful for files that do not update modification time or size.
- \* Only known to be needed when monitoring files on Windows, mostly for Internet Information Server logs.
- \* This flag should only be used as a last resort, as it increases load and slows down indexing.
- \* Defaults to 0.

time\_before\_close = <integer>

- \* Modification time delta required before the file monitor can close a file on EOF.
- \* Tells the system not to close files that have been updated in past <integer> seconds.
- \* Defaults to 3.

multiline\_event\_extra\_waittime = [true|false]

- \* By default, the file monitor sends an event delimiter when:
  - \* It reaches EOF of a file it monitors and
  - \* The last character it reads is a newline.
- \* In some cases, it takes time for all lines of a multiple-line event to arrive.
- \* Set to true to delay sending an event delimiter until the time that the file monitor closes the file, as defined by the 'time\_before\_close' setting, to allow all event lines to arrive.
- \* Defaults to false.

recursive = [true|false]

- \* If false, the input will not monitor sub-directories that it finds within a monitored directory.
- \* Defaults to true.

followSymlink = [true|false]

- \* Whether or not to follow any symbolic links within a monitored directory.
- \* If set to false, the input ignores symbolic links found within a monitored directory.
- \* If set to true, the input follows symbolic links and monitor files at the symbolic link destination.
- \* Additionally, any whitelists or blacklists that the input stanza defines also apply to files at the symbolic link's destination.
- \* Defaults to true.

\_whitelist = ...

- \* This setting is deprecated.
- \* It is still honored, unless the 'whitelist' setting also exists.

\_blacklist = ...

- \* This setting is deprecated.
- \* It is still honored, unless the 'blacklist' setting also exists.

```
# dedicatedFD = ...
* This setting has been removed. It is no longer needed.
```

## 批次 (Splunk Web 中的“上传文件”) :

```
#*****
# BATCH  ("Upload a file" in Splunk Web):
#*****BATCH  ("Upload a file" in Splunk Web):

NOTE: Batch should only be used for large archives of historic data. If you
want to continuously monitor a directory or index small archives, use 'monitor'
(see above). 'batch' reads in the file and indexes it, and then deletes the
file on disk.

[batch://<path>]
* A one-time, destructive input of files in <path>.
* For continuous, non-destructive inputs of files, use 'monitor' instead.

# Additional settings:

move_policy = sinkhole
* IMPORTANT: This setting is required. You *must* include
  "move_policy = sinkhole" when you define batch inputs.
* This setting causes the input to load the file destructively.
* Do not use the 'batch' input type for files you do not want to delete after
  indexing.
* The "move_policy" setting exists for historical reasons, but remains as an
  explicit double check. As an administrator you must very explicitly declare
  that you want the data in the monitored directory (and its sub-directories) to
  be deleted after being read and indexed.

host_regex = see MONITOR, above.
host_segment = see MONITOR, above.
crcSalt = see MONITOR, above.

# IMPORTANT: 'batch' inputs do not use the following setting:
# source = <string>

followSymlink = [true|false]
* Works similarly to the same setting for monitor, but does not delete files
  after following a symbolic link out of the monitored directory.

# The following settings work identically as for [monitor::] stanzas,
# documented above
host_regex = <regular expression>
host_segment = <integer>
crcSalt = <string>
recursive = [true|false]
whitelist = <regular expression>
blacklist = <regular expression>
initCrcLength = <integer>
```

## TCP :

```
#*****
# TCP:
#*****TCP:

[tcp://<remote server>:<port>]
* Configures the input to listen on a specific TCP network port.
* If a <remote server> makes a connection to this instance, this stanza is
  used to configure the input.
* If you do not specify <remote server>, this stanza matches all connections
  on the specified port.
* Generates events with source set to tcp:portnumber, for example: tcp:514
* If you do not specify a sourcetype, generates events with sourcetype
  set to tcp-raw.
```

```
# Additional settings:

connection_host = [ip|dns|none]
* "ip" sets the host to the IP address of the system sending the data.
* "dns" sets the host to the reverse DNS entry for the IP address of the system
  sending the data.
* "none" leaves the host as specified in inputs.conf, typically the splunk
  system hostname.
* Defaults to "dns".

queueSize = <integer>[KB|MB|GB]
* The maximum size of the in-memory input queue.
* Defaults to 500KB.

persistentQueueSize = <integer>[KB|MB|GB|TB]
* Maximum size of the persistent queue file.
* Defaults to 0 (no persistent queue).
* If set to some value other than 0, persistentQueueSize must be larger than
  the in-memory queue size (as defined by the 'queueSize' setting in
  inputs.conf or 'maxSize' settings in [queue] stanzas in server.conf).
* Persistent queues can help prevent loss of transient data. For information on
  persistent queues and how the 'queueSize' and 'persistentQueueSize' settings
  interact, see the online documentation.
* Defaults to 0 (no persistent queue).

requireHeader = <bool>
* Require a header be present at the beginning of every stream.
* This header may be used to override indexing settings.
* Defaults to false.

listenOnIPv6 = <no | yes | only>
* Select whether the input listens on IPv4, IPv6, or both
* Set this to 'yes' to listen on both IPv4 and IPv6 protocols.
* Set to 'only' to listen on only the IPv6 protocol.
* If not present, the input uses the setting in the [general] stanza
  of server.conf.

acceptFrom = <network_acl> ...
* Lists a set of networks or addresses to accept connections from.
* Separate multiple rules with commas or spaces.
* Each rule can be in one of the following formats:
  1. A single IPv4 or IPv6 address (examples: "10.1.2.3", "fe80::4a3")
  2. A Classless Inter-Domain Routing (CIDR) block of addresses
    (examples: "10/8", "fe80:1234/32")
  3. A DNS name, possibly with a '*' used as a wildcard
    (examples: "myhost.example.com", "*.splunk.com")
  4. A single '*' which matches anything
* You can also prefix an entry with '!' to cause the rule to reject the
  connection. The input applies rules in order, and uses the first one that
  matches.
  For example, "!10.1/16, *" allows connections from everywhere except
  the 10.1.*.* network.
* Defaults to "*" (accept from anywhere)

rawTcpDoneTimeout = <seconds>
* Specifies timeout value for sending Done-key.
* If a connection over this port remains idle for more than
  'rawTcpDoneTimeout' seconds after receiving data, it adds a Done-key. This
  declares that the last event has been completely received.
* Defaults to 10 seconds.

[tcp:<port>]
* Configures the input listen on the specified TCP network port.
* This stanza is similar to [tcp://<remote server>:<port>], but listens for
  connections to the specified port from any host.
* Generates events with a source of tcp:<port>.
* If you do not specify a sourcetype, generates events with a source type of
  tcp-raw.
* This stanza supports the following settings:

connection_host = [ip|dns|none]
queueSize = <integer>[KB|MB|GB]
```

```

persistentQueueSize = <integer>[KB|MB|GB|TB]
requireHeader = <bool>
listenOnIPv6 = <no | yes | only>
acceptFrom = <network_acl> ...
rawTcpDoneTimeout = <seconds>

```

## 数据分布：

```

#*****
# Data distribution:
#*****Data distribution:

# Global settings for splunktcp. Used on the receiving side for data forwarded
# from a forwarder.

[splunktcp]
route = [has_key|absent_key:<key>:<queueName>;...]
* Settings for the light forwarder.
* The receiver sets these parameters automatically -- you DO NOT need to set
  them.
* The property route is composed of rules delimited by ';' (semicolon).
* The receiver checks each incoming data payload via cooked tcp port against
  the route rules.
* If a matching rule is found, the receiver sends the payload to the specified
  <queueName>.
* If no matching rule is found, the receiver sends the payload to the default
  queue specified by any queue= for this stanza. If no queue= key is set in
  the stanza or globally, the events will be sent to the parsingQueue.

enableS2SHeartbeat = [true|false]
* This specifies the global keepalive setting for all splunktcp ports.
* This option is used to detect forwarders which might have become unavailable
  due to network, firewall, or other problems.
* The receiver monitors each connection for presence of heartbeat, and if the
  heartbeat is not seen for s2sHeartbeatTimeout seconds, it closes the
  connection.
* Defaults to true (heartbeat monitoring enabled).

s2sHeartbeatTimeout = <seconds>
* This specifies the global timeout value for monitoring heartbeats.
* The receiver closes a forwarder connection if it does not receive
  a heartbeat for 's2sHeartbeatTimeout' seconds.
* Defaults to 600 seconds (10 minutes).

inputShutdownTimeout = <seconds>
* Used during shutdown to minimize data loss when forwarders are connected to a
  receiver.
* During shutdown, the tcp input processor waits for the specified number of
  seconds and then closes any remaining open connections. If, however, all
  connections close before the end of the timeout period, shutdown proceeds
  immediately, without waiting for the timeout.

stopAcceptorAfterQBlock = <seconds>
* Specifies the time, in seconds, to wait before closing the splunktcp port.
* If the receiver is unable to insert received data into the configured queue
  for more than the specified number of seconds, it closes the splunktcp port.
* This action prevents forwarders from establishing new connections to this
  receiver.
* Forwarders that have an existing connection will notice the port is closed
  upon test-connections and move to other receivers.
* Once the queue unblocks, and TCP Input can continue processing data, the
  receiver starts listening on the port again.
* This setting should not be adjusted lightly as extreme values can interact
  poorly with other defaults.
* Defaults to 300 (5 minutes).

listenOnIPv6 = no|yes|only
* Select whether this receiver listens on IPv4, IPv6, or both protocols.
* Set this to 'yes' to listen on both IPv4 and IPv6 protocols.
* Set to 'only' to listen on only the IPv6 protocol.

```



\* If not present, the input uses the setting in the [general] stanza of server.conf.

acceptFrom = <network\_acl> ...

- \* Lists a set of networks or IP addresses from which to accept connections.
- \* Specify multiple rules with commas or spaces.
- \* Each rule can be in the following forms:
  1. A single IPv4 or IPv6 address (examples: "10.1.2.3", "fe80::4a3")
  2. A CIDR block of addresses (examples: "10/8", "fe80:1234/32")
  3. A DNS name, possibly with a '\*' used as a wildcard (examples: "myhost.example.com", "\*.splunk.com")
  4. A single '\*', which matches anything.
- \* You can also prefix an entry with '!' to cause the rule to reject the connection. The input applies rules in order, and uses the first one that matches. For example, "!10.1/16, \*" allows connections from everywhere except the 10.1.\*.\* network.
- \* Defaults to "\*" (accept from anywhere)

negotiateNewProtocol = [true|false]

- \* If set to true, lets forwarders that connect to this indexer (or specific port) send data using the new forwarder protocol.
- \* If set to false, denies the use of the new forwarder protocol during connection negotiation.
- \* Defaults to true.

concurrentChannelLimit = <unsigned integer>

- \* Each forwarder that connects to this indexer may use up to <concurrentChannelLimit> unique channel codes.
- \* In other words, each forwarder may have up to <concurrentChannelLimit> channels in flight concurrently.
- \* The receiver closes a forwarder connection if a forwarder attempts to exceed this value.
- \* This setting only applies when the new forwarder protocol is in use.
- \* Defaults to 300.

# Forwarder-specific settings for splunktcp.

[splunktcp://[<remote server>]:<port>]

- \* Receivers use this input stanza.
- \* This is the same as the [tcp://] stanza, except the remote server is assumed to be a Splunk instance, most likely a forwarder.
- \* <remote server> is optional. If you specify it, the receiver only listen for data from <remote server>.
- \* Use of <remote server> is not recommended. Use the 'acceptFrom' setting, which supersedes this setting.

connection\_host = [ip|dns|none]

- \* For splunktcp, the host or connection\_host will be used if the remote Splunk instance does not set a host, or if the host is set to "<host>::<localhost>".
- \* "ip" sets the host to the IP address of the system sending the data.
- \* "dns" sets the host to the reverse DNS entry for IP address of the system sending the data.
- \* "none" leaves the host as specified in inputs.conf, typically the splunk system hostname.
- \* Defaults to "ip".

compressed = [true|false]

- \* Specifies whether the receiver receives compressed data.
- \* Applies to non-SSL receiving only. There is no compression setting required for SSL.
- \* If set to true, the forwarder port(s) should also have compression turned on; otherwise, the receiver rejects the connection.
- \* Defaults to false.

enableS2SHeartbeat = [true|false]

- \* This specifies the keepalive setting for the splunktcp port.
- \* This option is used to detect forwarders which might have become unavailable due to network, firewall, or other problems.
- \* The receiver monitors the connection for presence of heartbeat, and if it does not see the heartbeat s2sHeartbeatTimeout seconds, it closes the connection.
- \* This overrides the default value specified at the global [splunktcp] stanza.

```

* Defaults to true (heartbeat monitoring enabled).

s2sHeartbeatTimeout = <seconds>
* This specifies the timeout value for monitoring heartbeats.
* The receiver closes the forwarder connection if it does not see a heartbeat
  for 's2sHeartbeatTimeout' seconds.
* This overrides the default value specified at the global [splunktcp] stanza.
* Defaults to 600 seconds (10 minutes).

queueSize = <integer>[KB|MB|GB]
* The maximum size of the in-memory input queue.
* Defaults to 500KB.

negotiateNewProtocol = [true|false]
* See the description for [splunktcp].

concurrentChannelLimit = <unsigned integer>
* See the description for [splunktcp].

[splunktcp:<port>]
* This input stanza is the same as [splunktcp://[<remote server>]:<port>], but
  does not have a remote server restriction.
* Please see documentation for [splunktcp://[<remote server>]:<port>] for
  following supported settings:

connection_host = [ip|dns|none]
compressed = [true|false]
enableS2SHeartbeat = [true|false]
s2sHeartbeatTimeout = <seconds>
queueSize = <integer>[KB|MB|GB]
negotiateNewProtocol = [true|false]
concurrentChannelLimit = <unsigned integer>

# Access control settings.
[splunktcp:token://<token name>]
* This stanza is optional.
* Use this stanza to specify forwarders from which to accept data.
* You must configure a token on the receiver, then configure the same
  token on forwarders.
* The receiver discards data from forwarders that do not have the
  token configured.
* This setting is enabled for all receiving ports.

token = <string>
* Value of token.

# SSL settings for data distribution:

[splunktcp-ssl:<port>]
* Use this stanza type if you are receiving encrypted, parsed data from a
  forwarder.
* Set <port> to the port on which the forwarder sends the encrypted data.
* Forwarder settings are set in outputs.conf on the forwarder.
* Compression for SSL is enabled by default. On the forwarder you can still
  specify compression with the 'useClientSSLCompression' setting in
  outputs.conf.
* The 'compressed' setting is used for non-SSL connections. However, if you
  still specify 'compressed' for SSL, ensure that the 'compressed' setting is
  the same as on the forwarder, as splunktcp protocol expects the same
  'compressed' setting from forwarders.

connection_host = [ip|dns|none]
* For splunktcp, the host or connection_host will be used if the remote Splunk
  instance does not set a host, or if the host is set to "<host>::<localhost>".
* "ip" sets the host to the IP address of the system sending the data.
* "dns" sets the host to the reverse DNS entry for IP address of the system
  sending the data.
* "none" leaves the host as specified in inputs.conf, typically the splunk
  system hostname.
* Defaults to "ip".

compressed = [true|false]

```

```

* See comments for [splunktcp:<port>].

enableS2SHeartbeat = true|false
* See comments for [splunktcp:<port>].

s2sHeartbeatTimeout = <seconds>
* See comments for [splunktcp:<port>].

listenOnIPv6 = no|yes|only
* Select whether this receiver listens on IPv4, IPv6, or both protocols.
* Set this to 'yes' to listen on both IPv4 and IPv6 protocols.
* Set to 'only' to listen on only the IPv6 protocol.
* If not present, the input uses the setting in the [general] stanza
  of server.conf.

acceptFrom = <network_acl> ...
* Lists a set of networks or IP addresses from which to accept connections.
* Specify multiple rules with commas or spaces.
* Each rule can be in the following forms:
  1. A single IPv4 or IPv6 address (examples: "10.1.2.3", "fe80::4a3")
  2. A CIDR block of addresses (examples: "10/8", "fe80:1234/32")
  3. A DNS name, possibly with a '*' used as a wildcard (examples:
    "myhost.example.com", "*.splunk.com")
  4. A single '*', which matches anything.
* You can also prefix an entry with '!' to cause the rule to reject the
  connection. The input applies rules in order, and uses the first one that
  matches. For example, "!10.1/16, *" allows connections from everywhere except
  the 10.1.*.* network.
* Defaults to "*" (accept from anywhere)

negotiateNewProtocol = [true|false]
* See comments for [splunktcp].

concurrentChannelLimit = <unsigned integer>
* See comments for [splunktcp].

# To specify global ssl settings, that are applicable for all ports, add the
# settings to the SSL stanza.
# Specify any ssl setting that deviates from the global setting here.
# For a detailed description of each ssl setting, refer to the [SSL] stanza.

serverCert = <path>
sslPassword = <password>
rootCA = <path>
requireClientCert = <bool>
sslVersions = <string>
cipherSuite = <cipher suite string>
ecdhCurves = <comma separated list of ec curves>
dhFile = <path>
allowSslRenegotiation = true|false
sslQuietShutdown = [true|false]
sslCommonNameToCheck = <commonName1>, <commonName2>, ...
sslAltNameToCheck = <alternateName1>, <alternateName2>, ...

[tcp-ssl:<port>]
* Use this stanza type if you are receiving encrypted, unparsed data from a
  forwarder or third-party system.
* Set <port> to the port on which the forwarder/third-party system is sending
  unparsed, encrypted data.

listenOnIPv6 = <no | yes | only>
* Select whether the receiver listens on IPv4, IPv6, or both protocols.
* Set this to 'yes' to listen on both IPv4 and IPv6 protocols.
* Set to 'only' to listen on only the IPv6 protocol.
* If not present, the receiver uses the setting in the [general] stanza
  of server.conf.

acceptFrom = <network_acl> ...
* Lists a set of networks or IP addresses from which to accept connections.
* Specify multiple rules with commas or spaces.
* Each rule can be in the following forms:
  1. A single IPv4 or IPv6 address (examples: "10.1.2.3", "fe80::4a3")

```

- 2. A CIDR block of addresses (examples: "10/8", "fe80:1234/32")
- 3. A DNS name, possibly with a '\*' used as a wildcard (examples: "myhost.example.com", "\*.splunk.com")
- 4. A single '\*', which matches anything.

- \* You can also prefix an entry with '!' to cause the rule to reject the connection. The input applies rules in order, and uses the first one that matches. For example, "!10.1/16, \*" allows connections from everywhere except the 10.1.\* network.
- \* Defaults to "\*" (accept from anywhere)

[SSL]

- \* Set the following specifications for receiving Secure Sockets Layer (SSL) communication underneath this stanza name.

serverCert = <path>

- \* The full path to the server certificate Privacy-Enhanced Mail (PEM) format file.
- \* PEM is the most common text-based storage format for SSL certificate files.
- \* There is no default.

sslPassword = <password>

- \* Server certificate password, if any.
- \* Initially set to plain-text password.
- \* Upon first use, the input encrypts and rewrites the password to \$SPLUNK\_HOME/etc/system/local/inputs.conf.

password = <password>

- \* This setting is DEPRECATED.
- \* Do not use this setting. Use the 'sslPassword' setting instead.

rootCA = <path>

- \* This setting is DEPRECATED.
- \* Do not use this setting. Use 'server.conf/[sslConfig]/sslRootCAPath' instead.
- \* Used only if 'sslRootCAPath' is unset.

requireClientCert = <bool>

- \* Determines whether a client must present an SSL certificate to authenticate.
- \* Full path to the root CA (Certificate Authority) certificate store.
- \* The <path> must refer to a PEM format file containing one or more root CA certificates concatenated together.
- \* Defaults to false.

sslVersions = <string>

- \* A comma-separated list of SSL versions to support.
- \* The versions available are "ssl3", "tls1.0", "tls1.1", and "tls1.2"
- \* The special version "\*" selects all supported versions. The version "tls" selects all versions "tls1.0" or newer.
- \* To remove a version from the list, prefix it with "-".
- \* SSLv2 is always disabled. You can specify "-ssl2" in the version list, but doing so has no effect.
- \* When configured in Federal Information Processing Standard (FIPS) mode, the "ssl3" version is always disabled, regardless of this configuration.
- \* Defaults to "\*, -ssl2". (anything newer than SSLv2)

supportSSLV3Only = <bool>

- \* This setting is DEPRECATED.
- \* SSLv2 is now always disabled.
- \* Use the "sslVersions" setting to set the list of supported SSL versions.

cipherSuite = <cipher suite string>

- \* If set, uses the specified cipher string for the input processors.
- \* If not set, the default cipher string is used.
- \* Provided by OpenSSL. This is used to ensure that the server does not accept connections using weak encryption protocols.
- \* Must specify 'dhFile' to enable any Diffie-Hellman ciphers.

ecdhCurveName = <string>

- \* This setting is DEPRECATED.
- \* Use the 'ecdhCurves' setting instead.
- \* This setting specifies the Elliptic Curve Diffie-Hellman (ECDH) curve to use for ECDH key negotiation.
- \* Splunk only supports named curves that have been specified by their SHORT name.

- \* The list of valid named curves by their short/long names can be obtained by executing this command:  
`$$SPLUNK_HOME/bin/splunk cmd openssl ecparam -list_curves`
- \* Default is empty string.

`ecdhCurves = <comma separated list of ec curves>`

- \* ECDH curves to use for ECDH key negotiation.
- \* The curves should be specified in the order of preference.
- \* The client sends these curves as a part of Client Hello.
- \* The server supports only the curves specified in the list.
- \* Splunk only supports named curves that have been specified by their SHORT names.  
 (see struct ASN1\_OBJECT in asn1.h)
- \* The list of valid named curves by their short/long names can be obtained by executing this command:  
`$$SPLUNK_HOME/bin/splunk cmd openssl ecparam -list_curves`
- \* Default is empty string.
- \* Example setting: `ecdhCurves = prime256v1,secp384r1,secp521r1`

`dhFile = <path>`

- \* Full path to the Diffie-Hellman parameter file.
- \* DH group size should be no less than 2048 bits.
- \* This file is required in order to enable any Diffie-Hellman ciphers.
- \* Not set by default.

`dhfile = <path>`

- \* This setting is DEPRECATED.
- \* Use the 'dhFile' setting instead.

`allowSslRenegotiation = true|false`

- \* In the SSL protocol, a client may request renegotiation of the connection settings from time to time.
- \* Setting this to false causes the server to reject all renegotiation attempts, which breaks the connection.
- \* This limits the amount of CPU a single TCP connection can use, but it can cause connectivity problems, especially for long-lived connections.
- \* Defaults to true.

`sslQuietShutdown = [true|false]`

- \* Enables quiet shutdown mode in SSL.
- \* Defaults to false.

`sslCommonNameToCheck = <commonName1>, <commonName2>, ...`

- \* Check the common name of the client's certificate against this list of names.
- \* If there is no match, assume that the Splunk instance is not authenticated against this server.
- \* This setting is optional.
- \* Defaults to no common name checking.
- \* `requireClientCert` must be set to true for this setting to work.

`sslAltNameToCheck = <alternateName1>, <alternateName2>, ...`

- \* Check the alternate name of the client certificate against this list of names.
- \* If there is no match, assume that the Splunk instance is not authenticated against this server.
- \* This setting is optional.
- \* Defaults to no alternate name checking.
- \* For this setting to work, the 'requireClientCert' setting must be set to true.

## **UDP :**

```
#*****
# UDP:
#*****UDP:
```

`[udp://<remote server>:<port>]`

- \* Similar to the `[tcp://]` stanza, except that this stanza causes the Splunk instance to listen on a UDP port.
- \* Only one stanza per port number is currently supported.
- \* Configures the instance to listen on a specific port.
- \* If you specify `<remote server>`, the specified port only accepts data

```

    from that host.
* If <remote server> is empty - [udp://<port>] - the port accepts data sent
  from any host.
* The use of <remote server> is not recommended. Use the 'acceptFrom'
  setting, which supersedes this setting.
* Generates events with source set to udp:portnumber, for example: udp:514
* If you do not specify a sourcetype, generates events with sourcetype set
  to udp:portnumber.

# Additional settings:

connection_host = [ip|dns|none]
* "ip" sets the host to the IP address of the system sending the data.
* "dns" sets the host to the reverse DNS entry for IP address of the system
  sending the data.
* "none" leaves the host as specified in inputs.conf, typically the splunk
  system hostname.
* Defaults to "ip".

_rcvbuf = <integer>
* Specifies the receive buffer for the UDP port (in bytes).
* If you set the value to 0 or a negative number, the input ignores the value.
* Note: If the default value is too large for an OS, the instance tries to set
  the value to 1572864/2. If that value is also too large, the instance
  retries with 1572864/(2*2). It continues to retry by halving the value until
  it succeeds.
* Defaults to 1,572,864.

no_priority_stripping = [true|false]
* Setting for receiving syslog data.
* If you set this setting to true, the instance does NOT strip the <priority>
  syslog field from received events.
* NOTE: Do NOT set this setting if you want to strip <priority>.
* Default is false.

no_appending_timestamp = [true|false]
* Whether or not to append a timestamp and host to received events.
* If you set this setting to true, the instance does NOT append a timestamp
  and host to received events.
* NOTE: Do NOT set this setting if you want to append timestamp and host
  to received events.
* Default is false.

queueSize = <integer>[KB|MB|GB]
* Maximum size of the in-memory input queue.
* Defaults to 500KB.

persistentQueueSize = <integer>[KB|MB|GB|TB]
* Maximum size of the persistent queue file.
* Defaults to 0 (no persistent queue).
* If set to some value other than 0, persistentQueueSize must be larger than
  the in-memory queue size (as defined by the 'queueSize' setting in
  inputs.conf or 'maxSize' settings in [queue] stanzas in server.conf).
* Persistent queues can help prevent loss of transient data. For information on
  persistent queues and how the 'queueSize' and 'persistentQueueSize' settings
  interact, see the online documentation.

listenOnIPv6 = <no | yes | only>
* Select whether the instance listens on the IPv4, IPv6, or both protocols.
* Set this to 'yes' to listen on both IPv4 and IPv6 protocols.
* Set to 'only' to listen on only the IPv6 protocol.
* If not present, the input uses the setting in the [general] stanza
  of server.conf.

acceptFrom = <network_acl> ...
* Lists a set of networks or IP addresses from which to accept connections.
* Specify multiple rules with commas or spaces.
* Each rule can be in the following forms:
  1. A single IPv4 or IPv6 address (examples: "10.1.2.3", "fe80::4a3")
  2. A CIDR block of addresses (examples: "10/8", "fe80:1234/32")
  3. A DNS name, possibly with a '*' used as a wildcard (examples:
    "myhost.example.com", "*.splunk.com")

```

4. A single '\*', which matches anything.

- \* You can also prefix an entry with '!' to cause the rule to reject the connection. The input applies rules in order, and uses the first one that matches.

For example, "!10.1/16, \*" allows connections from everywhere except the 10.1.\*.\* network.

- \* Defaults to "\*" (accept from anywhere)

```
[udp:<port>]
```

- \* This input stanza is the same as [udp://<remote server>:<port>], but does not have a <remote server> restriction.
- \* See the documentation for [udp://<remote server>:<port>] to configure supported settings:

```
connection_host = [ip|dns|none]
_rcvbuf = <integer>
no_priority_stripping = [true|false]
no_appending_timestamp = [true|false]
queueSize = <integer>[KB|MB|GB]
persistentQueueSize = <integer>[KB|MB|GB|TB]
listenOnIPv6 = <no | yes | only>
acceptFrom = <network_acl> ...
```

## **FIFO (先入先出队列)**

```
#*****
# FIFO (First In, First Out queue):
#*****FIFO (First In, First Out queue):
```

```
[fifo://<path>]
```

- \* This stanza configures the monitoring of a FIFO at the specified path.

```
queueSize = <integer>[KB|MB|GB]
```

- \* Maximum size of the in-memory input queue.
- \* Defaults to 500KB.

```
persistentQueueSize = <integer>[KB|MB|GB|TB]
```

- \* Maximum size of the persistent queue file.
- \* Defaults to 0 (no persistent queue).
- \* If set to some value other than 0, persistentQueueSize must be larger than the in-memory queue size (as defined by the 'queueSize' setting in inputs.conf or 'maxSize' settings in [queue] stanzas in server.conf).
- \* Persistent queues can help prevent loss of transient data. For information on persistent queues and how the 'queueSize' and 'persistentQueueSize' settings interact, see the online documentation.

## **脚本式输入：**

```
#*****
# Scripted Input:
#*****Scripted Input:
```

```
[script://<cmd>]
```

- \* Runs <cmd> at a configured interval (see below) and indexes the output that <cmd> returns.
- \* The <cmd> must reside in one of the following directories:
  - \* \$SPLUNK\_HOME/etc/system/bin/
  - \* \$SPLUNK\_HOME/etc/apps/\$YOUR\_APP/bin/
  - \* \$SPLUNK\_HOME/bin/scripts/
- \* The path to <cmd> can be an absolute path, make use of an environment variable such as \$SPLUNK\_HOME, or use the special pattern of an initial '.' as the first directory to indicate a location inside the current app.
- \* The '.' specification must be followed by a platform-specific directory separator.
  - \* For example, on UNIX:

```
[script://./bin/my_script.sh]
```
  - Or on Windows:

```
[script://.\bin\my_program.exe]
```

This '.' pattern is strongly recommended for app developers, and necessary for operation in search head pooling environments.

- \* <cmd> can also be a path to a file that ends with a ".path" suffix. A file with this suffix is a special type of pointer file that points to a command to be run. Although the pointer file is bound by the same location restrictions mentioned above, the command referenced inside it can reside anywhere on the file system. The .path file must contain exactly one line: the path to the command to run, optionally followed by command-line arguments. The file can contain additional empty lines and lines that begin with '#'. The input ignores these lines.

interval = [<number>|<cron schedule>]

- \* How often to run the specified command (in seconds), or a valid cron schedule.
- \* NOTE: when you specify a cron schedule, the input does not run the script on start-up.
- \* If you specify the interval as a number, it may have a fractional component; e.g., 3.14
- \* The cron implementation for data inputs does not currently support names of months or days.
- \* Defaults to 60.0 seconds.
- \* The special value 0 forces this scripted input to be run continuously; that is, as soon as the script exits, the input restarts it.
- \* The special value -1 causes the scripted input to run once on start-up.

passAuth = <username>

- \* User to run the script as.
- \* If you provide a username, the instance generates an auth token for that user and passes it to the script via stdin.

queueSize = <integer>[KB|MB|GB]

- \* Maximum size of the in-memory input queue.
- \* Defaults to 500KB.

persistentQueueSize = <integer>[KB|MB|GB|TB]

- \* Maximum size of the persistent queue file.
- \* Defaults to 0 (no persistent queue).
- \* If set to some value other than 0, persistentQueueSize must be larger than the in-memory queue size (as defined by the 'queueSize' setting in inputs.conf or 'maxSize' settings in [queue] stanzas in server.conf).
- \* Persistent queues can help prevent loss of transient data. For information on persistent queues and how the 'queueSize' and 'persistentQueueSize' settings interact, see the online documentation.

index = <index name>

- \* The index where the input sends the data.
- \* Note: this parameter will be passed as a command-line argument to <cmd> in the format: -index <index name>.
- \* If the script does not need the index info, it can ignore this argument.
- \* If you do not specify an index, the script uses the default index.

send\_index\_as\_argument\_for\_path = [true|false]

- \* Whether or not to pass the index as an argument when specified for stanzas that begin with 'script:/'
- \* When you set this setting to true, the script passes the argument as '-index <index name>'.
- \* To avoid passing the index as a command line argument, set this to false.
- \* Defaults to true.

start\_by\_shell = [true|false]

- \* Whether or not to run the specified command through the operating system shell or command prompt.
- \* If you set this setting to true, the host operating system runs the specified command through the OS shell ("/bin/sh -c" on UNIX, "cmd.exe /c" on Windows.)
- \* If you set the setting to false, the input runs the program directly without attempting to expand shell metacharacters.
- \* On Unix hosts, defaults to true.
- \* On Windows hosts defaults to false.
- \* You might want to explicitly set the setting to false for scripts that you know do not need UNIX shell metacharacter expansion. This is a Splunk best practice.



## 文件系统更改监视器 (fschange monitor)

```
#####
# File system change monitor (fschange monitor)
#####File system change monitor (fschange monitor)
#
# The file system change monitor has been deprecated as of Splunk Enterprise
# version 5.0 and might be removed in a future version of the product.
#
# You cannot simultaneously monitor a directory with both the 'fschange'
# and 'monitor' stanza types.

[fschange:<path>]
* Monitors changes (such as additions, updates, and deletions) to this
  directory and any of its sub-directories.
* <path> is the direct path. Do not preface it with '/' like with
  other inputs.
* Sends an event for every change.

# Additional settings:
# NOTE: The 'fschange' stanza type does not use the same settings as
# other input types. It uses only the following settings:

index = <index name>
* The index where the input sends the data.
* Defaults to _audit, unless you either do not set the 'signedaudit'
  setting, or set 'signedaudit' to false.
* If you set 'signedaudit' to false, events go into the default index.

signedaudit = [true|false]
* Whether or not to send cryptographically signed add/update/delete events.
* If you set this setting to true, the input does the following to
  events that it generates:
  * Puts the events in the _audit index.
  * Sets the event sourcetype to 'audittrail'
* If you set the setting to false, the input:
  * Places events in the default index.
  * Sets the sourcetype to whatever you specify (or "fs_notification"
    by default).
* You must set 'signedaudit' to false if you want to set the index for
  fschange events.
* You must also enable auditing in audit.conf.
* Defaults to false.

filters = <filter1>,<filter2>,...
* Each filter is applied left to right for each file or directory
  found during the monitor poll cycle.
* See the "File System Monitoring Filters" section below for help
  on how to define a fschange filter.

recurse = [true|false]
* Whether or not the fschange input should look through all sub-directories
  for changes to files in a directory.
* If you set this setting to true, the input recurses through
  sub-directories within the directory specified in [fschange].
* Defaults to true.

followLinks = [true|false]
* Whether or not the fschange input should follow any symbolic
  links it encounters.
* If you set this setting to true, the input follows symbolic links.
* Do not set this setting to true unless you can confirm that
  doing so will not create a file system loop (For example, in
  Directory A, symbolic link B points back to Directory A.)
* Defaults to false.

pollPeriod = <integer>
* How often, in seconds, to check a directory for changes.
* Defaults to 3600 seconds (1 hour).
```

```

hashMaxSize = <integer>
* Calculate a SHA256 hash for every file that is less than or equal to
  <integer> bytes.
* The input uses this hash as an additional method for detecting changes to the
  file/directory.
* Defaults to -1 (disabled).

fullEvent = [true|false]
* Whether or not to send the full event if the input detects an add or
  update change.
* Set to true to send the full event if an add or update change is detected.
* Further qualified by the 'sendEventMaxSize' setting.
* Defaults to false.

sendEventMaxSize = <integer>
* Limits the size of event data that the fschange input sends.
* Only send the full event if the size of the event is less than or equal to
  <integer> bytes.
* This limits the size of indexed file data.
* Defaults to -1, which is unlimited.

sourcetype = <string>
* Set the source type for events from this input.
* The input automatically prepends "sourcetype=" to <string>.
* Defaults to "audittrail" if you set the 'signedaudit' setting to true.
* Defaults to "fs_notification" if you set the 'signedaudit' setting to false.

host = <string>
* Set the host name for events from this input.
* Defaults to whatever host sent the event.

filesPerDelay = <integer>
* The number of files that the fschange input processes between processing
  delays, as specified by the 'delayInMills' setting.
* After a delay of 'delayInMills' milliseconds, the fschange input processes
  <integer> files, then waits 'delayInMills' milliseconds again before
  repeating this process.
* This is used to throttle file system monitoring so it consumes less CPU.
* Defaults to 10.

delayInMills = <integer>
* The delay, in milliseconds, that the fschange input waits prior to
  processing 'filesPerDelay' files.
* After a delay of 'delayInMills' milliseconds, the fschange input processes
  <integer> files, then waits 'delayInMills' milliseconds again before
  repeating this process.
* This is used to throttle file system monitoring so it consumes less CPU.
* Defaults to 100.

```

## 文件系统监视过滤器：

```

#*****
# File system monitoring filters:
#*****File system monitoring filters:

[filter:<filtertype>:<filtername>]
* Defines a filter of type <filtertype> and names it <filtername>.
* <filtertype>:
  * Filter types are either 'blacklist' or 'whitelist.'
  * A whitelist filter processes all file names that match the
    regular expression list that you define within the stanza.
  * A blacklist filter skips all file names that match the
    regular expression list.
* <filtername>
  * The fschange input uses filter names that you specify with
    the 'filters' setting for a given fschange stanza.
  * You can specify multiple filters buy separating them with commas.

```

```

regex<integer> = <regex>
* Blacklist and whitelist filters can include a set of regular expressions.
* The name of each regex MUST be 'regex<integer>', where <integer>
  starts at 1 and increments.
* The input applies each regular expression in numeric order:
  regex1=<regex>
  regex2=<regex>
  ...

```

## **http: (HTTP 事件收集器)**

```

#*****
# http: (HTTP Event Collector)
#*****http: (HTTP Event Collector)

# Global settings for the HTTP Event Collector (HEC) Input.

[http]
port = <number>
* The event collector data endpoint server port.
* Defaults to 8088.

disabled = [0|1]
* Whether or not the event collector input is active.
* Set this setting to 1 to disable the input, and 0 to enable it.
* Defaults to 1 (disabled).

outputgroup = <string>
* The name of the output group that the event collector forwards data to.
* Defaults to empty string.

useDeploymentServer = [0|1]
* Whether or not the event collector input should write its configuration to
  a deployment server repository.
* When you set this setting to 1 (enabled), the input writes its
  configuration to the directory that you specify with the
  'repositoryLocation' setting in serverclass.conf.
* You must copy the full contents of the splunk_httpinput app directory
  to this directory for the configuration to work.
* When disabled, the input writes its configuration to
  $SPLUNK_HOME/etc/apps by default.
* Defaults to 0 (disabled).

index = <string>
* The default index to use.
* Defaults to the "default" index.

sourcetype = <string>
* The default source type for the events.
* If you do not specify a sourcetype, the input does not set a sourcetype
  for events it generates.

enableSSL = [0|1]
* Whether or not to use SSL for the event collector endpoint server.
* HEC shares SSL settings with the Splunk management server and cannot have
  'enableSSL' set to true when the Splunk management server has SSL disabled.
* Defaults to 0 (enabled).

dedicatedIoThreads = <number>
* Defines the number of dedicated input/output threads in the event collector
  input.
* Defaults to 0 (The input uses a single thread).

maxSockets = <int>
* The number of simultaneous HTTP connections that the event collector input
  accepts simultaneously.
* Set this setting to constrain resource usage.
* If you set this setting to 0, the input automatically sets it to
  one third of the maximum allowable open files on the host.
* If this number is less than 50, the input sets it to 50. If this number

```

is greater than 400000, the input sets it to 400000.

- \* If this number is negative, the input does not enforce a limit on connections.
- \* Defaults to 0.

maxThreads = <int>

- \* The number of threads that can be used by active HTTP transactions.
- \* Set this to constrain resource usage.
- \* If you set this setting to 0, the input automatically sets the limit to one third of the maximum allowable threads on the host.
- \* If this number is less than 20, the input sets it to 20. If this number is greater than 150000, the input sets it to 150000.
- \* If the 'maxSockets' setting has a positive value and 'maxThreads' is greater than 'maxSockets', then the input sets 'maxThreads' to be equal to 'maxSockets'.
- \* If set to a negative number, the input does not enforce a limit on threads.
- \* Defaults to 0.

serverCert = <path>

- \* The full path to the server certificate PEM format file.
- \* The same file may also contain a private key.
- \* Default is \$SPLUNK\_HOME/etc/auth/server.pem.
- \* The Splunk software automatically generates certificates when it first starts.
- \* You may replace the auto-generated certificate with your own certificate.

sslKeysfile = <filename>

- \* This setting is DEPRECATED.
- \* Use the 'serverCert' setting instead.
- \* File is in the directory specified by 'caPath' (see below).
- \* Defaults to server.pem.

sslPassword = <password>

- \* The server certificate password.
- \* Initially set to plain-text password.
- \* Upon first use, it will be encrypted and rewritten.
- \* Defaults to "password".

sslKeysfilePassword = <password>

- \* This setting is DEPRECATED.
- \* Use the 'sslPassword' setting instead.

caCertFile = <filename>

- \* This setting is DEPRECATED.
- \* Use the 'server.conf/[sslConfig]/sslRootCAPath' setting instead.
- \* Used only if you do not set the 'sslRootCAPath' setting.
- \* Specifies the file name (relative to 'caPath') of the CA (Certificate Authority) certificate PEM format file containing one or more certificates concatenated together.
- \* Defaults to cacert.pem.

caPath = <path>

- \* This setting is DEPRECATED.
- \* Use absolute paths for all certificate files.
- \* If certificate files given by other settings in this stanza are not absolute paths, then they will be relative to this path.
- \* Defaults to \$SPLUNK\_HOME/etc/auth.

sslVersions = <versions\_list>

- \* A comma-separated list of SSL versions to support.
- \* The versions available are "ssl3", "tls1.0", "tls1.1", and "tls1.2"
- \* The special version "\*" selects all supported versions. The version "tls" selects all versions "tls1.0" or newer.
- \* To remove a version from the list, prefix it with "-".
- \* SSLv2 is always disabled. You can specify "-ssl2" in the version list, but doing so has no effect.
- \* When configured in Federal Information Processing Standard (FIPS) mode, the "ssl3" version is always disabled, regardless of this configuration.
- \* Defaults to "\*, -ssl2". (anything newer than SSLv2)

cipherSuite = <cipher suite string>

- \* The cipher string to use for the HTTP server.

- \* Use this setting to ensure that the server does not accept connections using weak encryption protocols.
- \* If you set this setting, the input uses the specified cipher string for the HTTP server.
- \* If you do not set the setting, the input uses the default cipher string that OpenSSL provides.

listenOnIPv6 = no|yes|only

- \* Select whether this input listens on IPv4, IPv6, or both.
- \* Set this to 'yes' to listen on both IPv4 and IPv6 protocols.
- \* Set to 'only' to listen on only the IPv6 protocol.
- \* If not present, the input uses the setting in the [general] stanza of server.conf.

acceptFrom = <network\_acl> ...

- \* Lists a set of networks or IP addresses from which to accept connections.
- \* Specify multiple rules with commas or spaces.
- \* Each rule can be in the following forms:
  1. A single IPv4 or IPv6 address (examples: "10.1.2.3", "fe80::4a3")
  2. A CIDR block of addresses (examples: "10/8", "fe80:1234/32")
  3. A DNS name, possibly with a '\*' used as a wildcard (examples: "myhost.example.com", "\*.splunk.com")
  4. A single '\*', which matches anything.
- \* You can also prefix an entry with '!' to cause the rule to reject the connection. The input applies rules in order, and uses the first one that matches. For example, "!10.1/16, \*" allows connections from everywhere except the 10.1.\*.\* network.
- \* Defaults to "\*" (accept from anywhere)

requireClientCert = <bool>

- \* Requires that any client connecting to the HEC port has a certificate that can be validated by the certificate authority specified in the 'caCertFile' setting.
- \* Defaults to false.

ecdhCurveName = <string>

- \* This setting is DEPRECATED.
- \* Use the 'ecdhCurves' setting instead.
- \* This setting specifies the ECDH curve to use for ECDH key negotiation.
- \* Splunk only supports named curves that have been specified by their SHORT name.
- \* The list of valid named curves by their short/long names can be obtained by executing this command:  
\$SPLUNK\_HOME/bin/splunk cmd openssl ecparam -list\_curves
- \* Default is empty string.

ecdhCurves = <comma separated list of ec curves>

- \* ECDH curves to use for ECDH key negotiation.
- \* The curves should be specified in the order of preference.
- \* The client sends these curves as a part of Client Hello.
- \* The server supports only the curves specified in the list.
- \* Splunk only supports named curves that have been specified by their SHORT names. (see struct ASN1\_OBJECT in asn1.h)
- \* The list of valid named curves by their short/long names can be obtained by executing this command:  
\$SPLUNK\_HOME/bin/splunk cmd openssl ecparam -list\_curves
- \* Default is empty string.
- \* Example setting: ecdhCurves = prime256v1,secp384r1,secp521r1

crossOriginSharingPolicy = <origin\_acl> ...

- \* List of the HTTP Origins for which to return Access-Control-Allow-\* (CORS) headers.
- \* These headers tell browsers that we trust web applications at those sites to make requests to the REST interface.
- \* The origin is passed as a URL without a path component (for example "https://app.example.com:8000").
- \* This setting can take a list of acceptable origins, separated by spaces and/or commas.
- \* Each origin can also contain wildcards for any part. Examples:
  - \*://app.example.com:\* (either HTTP or HTTPS on any port)
  - https://\*.example.com (any host under example.com, including example.com itself).
- \* An address can be prefixed with a '!' to negate the match, with the first matching origin taking precedence. For example,

```

"!*://evil.example.com:* *://*.example.com:*" to not avoid
matching one host in a domain.
* A single "*" can also be used to match all origins.
* By default, the list is empty.

forceHttp10 = auto|never|always
* Whether or not the REST HTTP server forces clients that connect
  to it to use the HTTP 1.0 specification for web communications.
* When set to "always", the REST HTTP server does not use some
  HTTP 1.1 features such as persistent connections or chunked
  transfer encoding.
* When set to "auto" it does this only if the client did not send
  a User-Agent header, or if the user agent is known to have bugs
  in its support of HTTP/1.1.
* When set to "never" it always allows HTTP 1.1, even to
  clients it suspects may be buggy.
* Defaults to "auto".

sslCommonNameToCheck = <commonName1>, <commonName2>, ...
* If you set this setting and also set 'requireClientCert' to true,
  splunkd limits most inbound HTTPS connections to hosts that use
  a cert with one of the listed common names.
* The most important scenario is distributed search.
* This feature does not work with the deployment server and client
  communication over SSL.
* This setting is optional.
* Defaults to no common name checking.

sslAltNameToCheck = <alternateName1>, <alternateName2>, ...
* If you set this setting and also set 'requireClientCert' to true,
  splunkd can verify certificates that have a so-called
  "Subject Alternate Name" that matches any of the alternate
  names in this list.
* Subject Alternate Names are effectively extended descriptive
  fields in SSL certs beyond the commonName. A common practice for
  HTTPS certs is to use these values to store additional valid
  hostnames or domains where the cert should be considered valid.
* Accepts a comma-separated list of Subject Alternate Names to consider
  valid.
* Items in this list are never validated against the SSL Common Name.
* This feature does not work with the deployment server and client
  communication over SSL.
* Optional. Defaults to no alternate name checking

sendStrictTransportSecurityHeader = true|false
* If set to true, the REST interface sends a "Strict-Transport-Security"
  header with all responses to requests made over SSL.
* This can help avoid a client being tricked later by a Man-In-The-Middle
  attack to accept a non-SSL request. However, this requires a commitment that
  no non-SSL web hosts will ever be run on this hostname on any port. For
  example, if Splunk Web is in default non-SSL mode this can break the
  ability of browser to connect to it. Enable with caution.
* Defaults to false.

allowSslCompression = true|false
* If set to true, the server will allow clients to negotiate
  SSL-layer data compression.
* Defaults to true.

allowSslRenegotiation = true|false
* In the SSL protocol, a client may request renegotiation of the connection
  settings from time to time.
* Setting this to false causes the server to reject all renegotiation
  attempts, which breaks the connection.
* This limits the amount of CPU a single TCP connection can use, but it can
  cause connectivity problems, especially for long-lived connections.
* Defaults to true.

ackIdleCleanup = true|false
* If set to true, the server removes the ACK channels that are idle
  for 'maxIdleTime' seconds.
* Default to false.

```

```

maxIdleTime = <int>
* The maximum number of seconds the ACK channels are idle before they are
  removed.
* Defaults to 600 seconds.

channel_cookie = <string>
* The name of the cookie to use when sending data with a specified channel ID.
* The value of the cookie will be the channel sent. For example, if you have
  set 'channel_cookie=foo' and sent a request with channel ID set to 'bar',
  then you will have a cookie in the response with the value 'foo=bar'.
* If no channel ID is present in the request, then no cookie will be returned.
* This setting is to be used for load balancers (for example, AWS ELB) that can
  only provide sticky sessions on cookie values and not general header values.
* If no value is set (the default), then no cookie will be returned.
* Defaults to the empty string (no cookie).

```

## HTTP 事件收集器 (HEC) - 每个标记的本地段落

```

#*****
# HTTP Event Collector (HEC) - Local stanza for each token
#*****HTTP Event Collector (HEC) - Local stanza for each token

[http://name]

token = <string>
* The value of the HEC token.

disabled = [0|1]
* Whether or not this token is active.
* Defaults to 0 (enabled).

description = <string>
* A human-readable description of this token.
* Defaults to empty string.

indexes = <string>
* The indexes the event for this token can go to.
* If you do not specify this value, the index list is empty, and any index
  can be used.

index = <string>
* The default index to use for this token.
* Defaults to the default index.

sourcetype = <string>
* The default sourcetype to use if it is not specified in an event.
* Defaults to empty string.

outputgroup = <string>
* The name of the forwarding output group to send data to.
* Defaults to empty string.

queueSize = <integer>[KB|MB|GB]
* The maximum size of the in-memory input queue.
* Defaults to 500KB.

persistentQueueSize = <integer>[KB|MB|GB|TB]
* Maximum size of the persistent queue file.
* Defaults to 0 (no persistent queue).
* If set to some value other than 0, persistentQueueSize must be larger than
  the in-memory queue size (as defined by the 'queueSize' setting in
  inputs.conf or 'maxSize' settings in [queue] stanzas in server.conf).
* Persistent queues can help prevent loss of transient data. For information on
  persistent queues and how the 'queueSize' and 'persistentQueueSize' settings
  interact, see the online documentation.

connection_host = [ip|dns|proxied_ip|none]
* Specify the host if an event doesn't have host set.
* "ip" sets the host to the IP address of the system sending the data.

```

- \* "dns" sets the host to the reverse DNS entry for IP address of the system sending the data.
- \* "proxied\_ip" checks whether an X-Forwarded-For header was sent (presumably by a proxy server) and if so, sets the host to that value. Otherwise, the IP address of the system sending the data is used.
- \* "none" leaves the host as specified in the HTTP header.

useACK = [true|false]

- \* When set to true, acknowledgment (ACK) is enabled. Events in a request will be tracked until they are indexed. An events status (indexed or not) can be queried from the ACK endpoint with the ID for the request.
- \* When set to false, acknowledgment is not enabled.
- \* This setting can be set at the stanza level.
- \* Defaults to false.

## WINDOWS 输入：

```

#*****
# WINDOWS INPUTS:
#*****WINDOWS INPUTS:

* Windows platform specific input processor.
# *****

# Splunk on Windows ships with several Windows-only inputs. They are
# defined in the default inputs.conf.

* Use the "disabled=" setting to enable/disable any of them.
* A short summary of the inputs follows:
  * Perfmon: Monitors Windows performance counters, objects, and instances.
  * WinRegMon: Tracks and report any changes that occur in the
    local system Registry.
  * ADMon: Indexes existing Active Directory (AD) objects and listens for AD
    changes.
  * WMI: Retrieves event logs remotely and locally through the Windows
    Management Instrumentation subsystem. It can also gather performance
    data remotely, as well as receive various system notifications. See
    wmi.conf.spec for information on how to configure this input.

#*****

# The following Windows input specifications are for parsing on non-Windows
# platforms.
#*****

```

## 性能监视器

```

#*****
# Performance Monitor
#*****Performance Monitor

[perfmon://<name>]

* This section explains possible settings for configuring
  the Windows Performance Monitor input.
* Each perfmon:// stanza represents an individually configured performance
  monitoring input. If you configure the input through Splunk Web, then the
  value of "<NAME>" matches what was specified there. While you can add
  performance monitor inputs manually, Splunk recommends that you use Splunk
  Web to configure them, because it is easy to mistype the values for
  Performance Monitor objects, counters and instances.
* Note: The perfmon stanza is for local systems ONLY. To define performance
  monitor inputs for remote machines, use wmi.conf.

object = <string>

* This is a valid Performance Monitor object as defined within Performance
  Monitor (for example, "Process," "Server," "PhysicalDisk.")
* You can specify a single valid Performance Monitor object or use a
  regular expression (regex) to specify multiple objects.
* This setting is required, and the input will not run if the setting is

```



not present.

- \* There is no default.

counters = <semicolon-separated strings>

- \* This can be a single counter, or multiple valid Performance Monitor counters.
- \* This setting is required, and the input will not run if the setting is not present.
- \* '\*' is equivalent to all available counters for a given Performance Monitor object.
- \* There is no default.

instances = <semicolon-separated strings>

- \* This can be a single instance, or multiple valid Performance Monitor instances.
- \* '\*' is equivalent to all available instances for a given Performance Monitor counter.
- \* If applicable instances are available for a counter and this setting is not present, then the input logs data for all available instances (this is the same as setting 'instances = \*').
- \* If there are no applicable instances for a counter, then this setting can be safely omitted.
- \* There is no default.

interval = <integer>

- \* How often, in seconds, to poll for new data.
- \* This setting is required, and the input will not run if the setting is not present.
- \* The recommended setting depends on the Performance Monitor object, counter(s) and instance(s) that you define in the input, and how much performance data you require.
- \* Objects with numerous instantaneous or per-second counters, such as "Memory," "Processor" and "PhysicalDisk" should have shorter interval times specified (anywhere from 1-3 seconds).
- \* Less volatile counters such as "Terminal Services", "Paging File", and "Print Queue" can have longer times configured.
- \* Default is 300 seconds.

mode = [single|multikv]

- \* Specifies how the performance monitor input prints events.
- \* Set to 'single' to print each event individually, or 'multikv' to print events in multikv (formatted multiple key-value pair) format.
- \* Defaults to single.

samplingInterval = <sampling interval in ms>

- \* Advanced setting.
- \* How often, in milliseconds, to poll for new data.
- \* Enables high-frequency performance sampling. The input collects performance data every sampling interval. It then reports averaged data and other statistics at every interval.
- \* The minimum legal value is 100, and the maximum legal value must be less than what the 'interval' setting to.
- \* If not specified, high-frequency sampling does not take place.
- \* Defaults to not specified (disabled).

stats = <average;count;dev;min;max>

- \* Advanced setting.
- \* Reports statistics for high-frequency performance sampling.
- \* Acceptable values are: average, count, dev, min, max.
- \* You can specify multiple values by separating them with semicolons.
- \* If not specified, the input does not produce high-frequency sampling statistics.
- \* Defaults to not specified (disabled).

disabled = [0|1]

- \* Specifies whether or not the input is enabled.
- \* 1 to disable the input, 0 to enable it.
- \* Defaults to 0 (enabled).

index = <string>

- \* Specifies the index that this input should send the data to.

- \* This setting is optional.
- \* If no value is present, defaults to the default index.

showZeroValue = [0|1]

- \* Specifies whether or not zero value event data should be collected.
- \* Set to 1 to capture zero value event data, and 0 to ignore such data.
- \* Defaults to 0 (ignore zero value event data)

useEnglishOnly = [true|false]

- \* Controls which Windows Performance Monitor API the input uses.
- \* If true, the input uses PdhAddEnglishCounter() to add the counter string. This ensures that counters display in English regardless of the Windows host locale.
- \* If false, the input uses PdhAddCounter() to add the counter string.
- \* Note: if you set this setting to true, the 'object' setting does not accept a regular expression as a value on hosts that have a non-English locale.
- \* Defaults to false.

formatString = <double format specifier>

- \* Controls the print format for double-precision statistic counters.
- \* Do not use quotes when specifying this string.
- \* Defaults to "%.20g" (without quotes).

###

# Direct Access File Monitor (does not use file handles)

# For Windows systems only.

###

[MonitorNoHandle://<path>]

- \* This input intercepts file writes to the specific file.
- \* <path> must be a fully qualified path name to a specific file. Wildcards and directories are not accepted.
- \* You can specify more than one stanza of this type.

disabled = [0|1]

- \* Whether or not the input is enabled.
- \* Defaults to 0 (enabled).

index = <string>

- \* Specifies the index that this input should send the data to.
- \* This setting is optional.
- \* Defaults to the default index.

## Windows 事件日志输入

\*\*\*\*\*

# Windows Event Log Monitor

\*\*\*\*\*Windows Event Log Monitor

[WinEventLog://<name>]

- \* This section explains possible settings for configuring the Windows Event Log monitor.
- \* Each WinEventLog:// stanza represents an individually configured WinEventLog monitoring input. If you you configure the input through Splunk Web, the value of "<NAME>" matches what was specified there. While you can add event log monitor inputs manually, Splunk recommends that you use Splunk Web to configure Windows event log monitor inputs because it is easy to mistype the values for event log channels.
- \* Note: The WinEventLog stanza is for local systems only. To define event log monitor inputs for remote machines, use wmi.conf.

start\_from = <string>

- \* How the input should chronologically read the Event Log channels.
- \* If you set this setting to 'oldest', the input reads Windows event logs from oldest to newest.
- \* If you set this setting to 'newest' the input reads Windows event logs in reverse, from newest to oldest. Once the input consumes the backlog of

events, it stops.

- \* Do not set this setting to 'newest' and at the same time set the 'current\_only' setting to 1. This results in the input not collecting any events because you instructed it to read existing events from oldest to newest and read only incoming events concurrently (A logically impossible combination.)
- \* Defaults to oldest.

current\_only = [0|1]

- \* Whether or not to acquire only events that arrive while the instance is running.
- \* If you set this setting to 1, the input only acquires events that arrive while the instance runs and the input is enabled. The input does not read data which was stored in the Windows Event Log while the instance was not running. This means that there will be gaps in the data if you restart the instance or experiences downtime.
- \* If you set the setting to 0, the input first gets all existing events already stored in the log that have higher event IDs (have arrived more recently) than the most recent events acquired. The input then monitors events that arrive in real time.
- \* Do not set this setting to 1 and at the same time set the 'start\_from' setting to 'newest'. This results in the input not collecting any events because you instructed it to read existing events from oldest to newest and read only incoming events concurrently (A logically impossible combination.)
- \* Defaults to 0 (false), gathering stored events first before monitoring live events.

batch\_size = <integer>

- \* How many Windows Event Log items to read per request.
- \* If troubleshooting identifies that the Event Log input is a bottleneck in acquiring data, increasing this value can help.
- \* NOTE: Splunk Support has seen cases where large values can result in a stall in the Event Log subsystem.
- \* If you increase this value significantly, monitor closely for trouble.
- \* In local testing and in customer acceptance testing, 10 worked well for both throughput and reliability.
- \* The default value is 10.

checkpointInterval = <integer>

- \* How often, in seconds, that the Windows Event Log input saves a checkpoint.
- \* Checkpoints store the eventID of acquired events. This lets the input continue monitoring at the correct event after a shutdown or outage.
- \* The default value is 5.

disabled = [0|1]

- \* Whether or not the input is enabled.
- \* Set to 1 to disable the input, and 0 to enable it.
- \* The default is 0 (enabled).

evt\_resolve\_ad\_obj = [1|0]

- \* How the input should interact with Active Directory while indexing Windows Event Log events.
- \* If you set this setting to 1, the input resolves the Active Directory Security Identifier (SID) objects to their canonical names for a specific Windows Event Log channel.
- \* If you enable the setting, the rate at which the input reads events on high-traffic Event Log channels can decrease. Latency can also increase during event acquisition. This is due to the overhead involved in performing AD translations.
- \* When you set this setting to 1, you can optionally specify the domain controller name or dns name of the domain to bind to with the 'evt\_dc\_name' setting. The input connects to that domain controller to resolve the AD objects.
- \* If you set this setting to 0, the input does not attempt any resolution.
- \* Defaults to 0 (disabled) for all channels.

evt\_dc\_name = <string>

- \* Which Active Directory domain controller to bind to for AD object resolution.
- \* If you prefix a dollar sign to a value (for example, \$my\_domain\_controller), the input interprets the value as an environment variable. If the

environment variable has not been defined on the host, it is the same as if the value is blank.

- \* This setting is optional.
- \* This setting can be set to the NetBIOS name of the domain controller or the fully-qualified DNS name of the domain controller. Either name type can, optionally, be preceded by two backslash characters. The following examples represent correctly formatted domain controller names:
  - \* "FTW-DC-01"
  - \* "\\FTW-DC-01"
  - \* "FTW-DC-01.splunk.com"
  - \* "\\FTW-DC-01.splunk.com"
  - \* \$my\_domain\_controller

evt\_dns\_name = <string>

- \* The fully-qualified DNS name of the domain that the input should bind to for AD object resolution.
- \* This setting is optional.

evt\_resolve\_ad\_ds =[auto|PDC]

- \* How the input should choose the domain controller to bind for AD resolution.
- \* This setting is optional.
- \* If set to PDC, the input only contacts the primary domain controller to resolve AD objects.
- \* If set to auto, the input lets Windows chose the best domain controller.
- \* If you set the 'evt\_dc\_name' setting, the input ignores this setting.
- \* Defaults to 'auto' (let Windows determine the domain controller to use.)

evt\_ad\_cache\_disabled = [0|1]

- \* Enables or disables the AD object cache.
- \* Defaults to 0.

evt\_ad\_cache\_exp = <time in seconds>

- \* The expiration time, in seconds, for AD object cache entries.
- \* This setting is optional.
- \* The minimum allowed value is 10 and the maximum allowed value is 31536000.
- \* Defaults to 3600.

evt\_ad\_cache\_exp\_neg = <time in seconds>

- \* The expiration time, in seconds, for negative AD object cache entries.
- \* This setting is optional.
- \* The minimum allowed value is 10 and the maximum allowed value is 31536000.
- \* Defaults to 10.

evt\_ad\_cache\_max\_entries = <number of entries>

- \* The maximum number of AD object cache entries.
- \* This setting is optional.
- \* The minimum allowed value is 10 and the maximum allowed value is 40000.
- \* Defaults to 1000.

evt\_sid\_cache\_disabled = [0|1]

- \* Enables or disables account Security IDentifier (SID) cache.
- \* This setting is global. It affects all Windows Event Log stanzas.
- \* Defaults to 0.

evt\_sid\_cache\_exp = <time in seconds>

- \* The expiration time for account SID cache entries.
- \* This setting is optional.
- \* This setting is global. It affects all Windows Event Log stanzas.
- \* The minimum allowed value is 10 and the maximum allowed value is 31536000.
- \* Defaults to 3600.

evt\_sid\_cache\_exp\_neg = <time in seconds>

- \* The expiration time for negative account SID cache entries.
- \* This setting is optional.
- \* This setting is global. It affects all Windows Event Log stanzas.
- \* The minimum allowed value is 10 and the maximum allowed value is 31536000.
- \* Defaults to 10.

evt\_sid\_cache\_max\_entries = <number of entries>

- \* The maximum number of account SID cache entries.

- \* This setting is optional.
- \* This setting is global. It affects all Windows Event Log stanzas.
- \* The minimum allowed value is 10 and the maximum allowed value is 40000.
- \* Defaults to 10.

index = <string>

- \* Specifies the index that this input should send the data to.
- \* This setting is optional.
- \* If no value is present, defaults to the default index.

# Event Log filtering

#

# Filtering at the input layer is desirable to reduce the total

# processing load in network transfer and computation on the Splunk

# nodes that acquire and processing Event Log data.

whitelist = <list of eventIDs> | key=regex [key=regex]

blacklist = <list of eventIDs> | key=regex [key=regex]

whitelist1 = <list of eventIDs> | key=regex [key=regex]

whitelist2 = <list of eventIDs> | key=regex [key=regex]

whitelist3 = <list of eventIDs> | key=regex [key=regex]

whitelist4 = <list of eventIDs> | key=regex [key=regex]

whitelist5 = <list of eventIDs> | key=regex [key=regex]

whitelist6 = <list of eventIDs> | key=regex [key=regex]

whitelist7 = <list of eventIDs> | key=regex [key=regex]

whitelist8 = <list of eventIDs> | key=regex [key=regex]

whitelist9 = <list of eventIDs> | key=regex [key=regex]

blacklist1 = <list of eventIDs> | key=regex [key=regex]

blacklist2 = <list of eventIDs> | key=regex [key=regex]

blacklist3 = <list of eventIDs> | key=regex [key=regex]

blacklist4 = <list of eventIDs> | key=regex [key=regex]

blacklist5 = <list of eventIDs> | key=regex [key=regex]

blacklist6 = <list of eventIDs> | key=regex [key=regex]

blacklist7 = <list of eventIDs> | key=regex [key=regex]

blacklist8 = <list of eventIDs> | key=regex [key=regex]

blacklist9 = <list of eventIDs> | key=regex [key=regex]

- \* These settings are optional.
- \* Both numbered and unnumbered whitelists and blacklists support two formats:
  - \* A comma-separated list of event IDs.
  - \* A list of key=regular expression pairs.
  - \* You cannot combine these formats. You can use either format on a specific line.
- \* Numbered whitelist settings are permitted from 1 to 9, so whitelist1 through whitelist9 and blacklist1 through blacklist9 are supported.
- \* If no whitelist or blacklist rules are present, the input reads all events.

## 事件日志白名单和黑名单样式

```
###
# Event Log whitelist and blacklist formats
####Event Log whitelist and blacklist formats

* Event ID list format:
  * A comma-separated list of terms.
  * Terms may be a single event ID (e.g. 6) or range of event IDs (e.g. 100-200)
  * Example: 4,5,7,100-200
    * This applies to events with IDs 4, 5, 7, or any event ID between 100
      and 200, inclusive.
  * The event ID list format provides no additional functionality over the
    key=regex format, but can be easier to understand:
    List format:      4,5,7,100-200
    Regex equivalent: EventCode=%^(4|5|7|1..|200)$%

* key=regex format:
  * A whitespace-separated list of Event Log components to match, and
    regular expressions to match against against them.
  * There can be one match expression or multiple expressions per line.
```

- \* The key must belong to the set of valid keys provided below.
- \* The regex consists of a leading delimiter, the regex expression, and a trailing delimiter. Examples: %regex%, \*regex\*, "regex"
- \* When multiple match expressions are present, they are treated as a logical AND. In other words, all expressions must match for the line to apply to the event.
- \* If the value represented by the key does not exist, it is not considered a match, regardless of the regex.
- \* Example:  
whitelist = EventCode=%^200% User=%jrodman%  
Include events only if they have EventCode 200 and relate to User jrodman

# Valid keys for the key=regex format:

- \* The following keys are equivalent to the fields that appear in the text of the acquired events:  
Category, CategoryString, ComputerName, EventCode, EventType, Keywords, LogName, Message, OpCode, RecordNumber, Sid, SidType, SourceName, TaskCategory, Type, User
- \* There are two special keys that do not appear literally in the event.  
\* \$TimeGenerated: The time that the computer generated the event  
\* \$Timestamp: The time that the event was received and recorded by the Event Log service.
- \* The 'EventType' key is only available on Windows Server 2003 / Windows XP and earlier.
- \* The 'Type' key is only available on Windows Server 2008 / Windows Vista and later.
- \* For a detailed definition of these keys, see the online documentation:  
[http://docs.splunk.com/Documentation/Splunk/latest/Data/MonitorWindowsdata#Create\\_advanced\\_filters\\_with\\_.27whitelis](http://docs.splunk.com/Documentation/Splunk/latest/Data/MonitorWindowsdata#Create_advanced_filters_with_.27whitelis)

suppress\_text = [0|1]

- \* Whether or not to include the description of the event text for a given Event Log event.
- \* This setting is optional.
- \* Set this setting to 1 to suppress the inclusion of the event text description.
- \* Set this value to 0 to include the event text description.
- \* Defaults to 0.

renderXml= [true|false]

- \* Whether or not the input returns the event data in XML (eXtensible Markup Language) format or in plain text.
- \* Set this to true to render events in XML.
- \* Set this to false to output events in plain text.
- \* Defaults to false.

## Active Directory 监视器

```
#*****
# Active Directory Monitor
#*****Active Directory Monitor
```

[admon://<name>]

- \* This section explains possible settings for configuring the Active Directory monitor input.
- \* Each admon:// stanza represents an individually configured Active Directory monitoring input. If you configure the input with Splunk Web, then the value of "<NAME>" matches what was specified there. While you can add Active Directory monitor inputs manually, Splunk recommends that you use Splunk Web to configure Active Directory monitor inputs because it is easy to mistype the values for Active Directory monitor objects.

targetDc = <string>

- \* The fully qualified domain name of a valid, network-accessible Active Directory domain controller.
- \* Defaults to the DC that the local host used to connect to AD. The input binds to its root Distinguished Name (DN).

```

startingNode = <string>
* Where in the Active Directory directory tree to start monitoring.
* The user that you configure the Splunk software to run as at
  installation determines where the input starts monitoring.
* If not specified, the input attempts to start at the root of
  the directory tree.

monitorSubtree = [0|1]
* Whether or not to monitor the subtree(s) of a given Active
  Directory tree path.
* Set this to 1 to monitor subtrees of a given directory tree
  path and 0 to monitor only the path itself.
* Defaults to 1 (monitor subtrees of a given directory tree path).

disabled = [0|1]
* Whether or not the input is enabled.
* Set this to 1 to disable the input and 0 to enable it.
* Defaults to 0 (enabled.)

index = <string>
* The index to store incoming data into for this input.
* This setting is optional.
* Defaults to the default index.

printSchema = [0|1]
* Whether or not to print the Active Directory schema.
* Set this to 1 to print the schema and 0 to not print
  the schema.
* Defaults to 1 (print the Active Directory schema).

baseline = [0|1]
* Whether or not to query baseline objects.
* Baseline objects are objects which currently reside in Active Directory.
* Baseline objects also include previously deleted objects.
* Set this to 1 to query baseline objects, and 0 to not query
  baseline objects.
* Defaults to 0 (do not query baseline objects).

```

## **Windows 注册表监视器**

```

###
# Windows Registry Monitor
###Windows Registry Monitor

[WinRegMon://<name>]

* This section explains possible settings for configuring the Windows Registry
  Monitor input.
* Each WinRegMon:// stanza represents an individually configured
  WinRegMon monitoring input.
* If you configure the inputs with Splunk Web, the value of "<NAME>" matches
  what was specified there. While you can add event log monitor inputs
  manually, recommends that you use Splunk Web to configure
  Windows registry monitor inputs because it is easy to mistype the values
  for Registry hives and keys.
* The WinRegMon input is for local systems only.

proc = <string>
* Which processes this input should monitor for Registry access.
* If set, matches against the process name which performed the Registry
  access.
* The input includes events from processes that match the regular expression
  that you specify here.
* The input filters out events for processes that do not match the
  regular expression.
* There is no default.

hive = <string>
* The Registry hive(s) that this input should monitor for Registry access.
* If set, matches against the Registry key that was accessed.

```

- \* The input includes events from Registry hives that match the regular expression that you specify here.
- \* The input filters out events for Registry hives that do not match the regular expression.
- \* There is no default.

type = <string>

- \* A regular expression that specifies the type(s) of Registry event(s) that you want the input to monitor.
- \* There is no default.

baseline = [0|1]

- \* Whether or not the input should get a baseline of Registry events when it starts.
- \* If you set this to 1, the input captures a baseline for the specified hive when it starts for the first time. It then monitors live events.
- \* Defaults to 0 (do not capture a baseline for the specified hive first before monitoring live events).

baseline\_interval = <integer>

- \* Selects how much downtime in continuous registry monitoring should trigger a new baseline for the monitored hive and/or key.
- \* In detail:
  - \* Sets the minimum time interval, in seconds, between baselines.
  - \* At startup, a WinRegMon input will not generate a baseline if less time has passed since the last checkpoint than baseline\_interval chooses.
  - \* In normal operation, checkpoints are updated frequently as data is acquired, so this will cause baselines to occur only when monitoring was not operating for a period of time.
- \* If baseline is set to 0 (disabled), has no effect.
- \* Defaults to 0 (always baseline on startup, if baseline is 1)

disabled = [0|1]

- \* Whether or not the input is enabled.
- \* Set this to 1 to disable the input, or 0 to enable it.
- \* Defaults to 0 (enabled).

index = <string>

- \* The index that this input should send the data to.
- \* This setting is optional.
- \* Defaults to the default index.

## Windows 主机监视

```
###
# Windows Host Monitoring
###Windows Host Monitoring

[WinHostMon://<name>]
```

- \* This section explains possible settings for configuring the Windows host monitor input.
- \* Gathers status information from the local Windows system components as per the type field below.
- \* Each WinHostMon:// stanza represents an WinHostMon monitoring input.
- \* The "<name>" component of the stanza name will be used as the source field on generated events, unless an explicit source setting is added to the stanza. It does not affect what data is collected (see type setting for that).
- \* If you configure the input in Splunk web, the value of "<name>" matches what was specified there.
- \* Note: The WinHostMon input is for local Windows systems only. You can not monitor Windows host information remotely.

type = <semicolon-separated strings>

- \* An expression that specifies the type(s) of host inputs that you want the input to monitor.
- \* Type can be (case insensitive)
   
Computer;Process;Processor;NetworkAdapter;Service;OperatingSystem;Disk;Driver;Roles



```

interval = <integer>
* The interval, in seconds, between when the input runs to gather
  Windows host information and generate events.
* See interval in the Scripted input section for more information.

disabled = [0|1]
* Whether or not the input is enabled.
* Set this to 1 to disable the input, or 0 to enable it.
* Defaults to 0 (enabled).

index = <string>
* The index that this input should send the data to.
* This setting is optional.
* Defaults to the default index.

[WinPrintMon://<name>]

* This section explains possible settings for configuring the Windows print
  monitor input.
* Each WinPrintMon:// stanza represents an WinPrintMon monitoring input.
  The value of "<name>" matches what was specified in Splunk Web.
* Note: The WinPrintMon input is for local Windows systems only.
* The "<name>" component of the stanza name will be used as the source field
  on generated events, unless an explicit source setting is added to the
  stanza. It does not affect what data is collected (see type setting for
  that).

type = <semicolon-separated strings>
* An expression that specifies the type(s) of print inputs
  that you want the input to monitor.
* Type can be (case insensitive)
  Printer;Job;Driver;Port

baseline = [0|1]
* Whether or not to capture a baseline of print objects when the
  input starts for the first time.
* If you set this to 1, the input captures a baseline of
  the current print objects when the input starts for the first time.
* Defaults to 0 (do not capture a baseline.)

disabled = [0|1]
* Whether or not the input is enabled.
* Set to 1 to disable the input, or 0 to enable it.
* Defaults to 0 (enabled).

index = <string>
* The index that this input should send the data to.
* This setting is optional.
* Defaults to the default index.

[WinNetMon://<name>]

* This section explains possible settings for configuring
  a Network Monitor input.
* Each WinNetMon:// stanza represents an individually configured network
  monitoring input. The value of "<name>" matches what was specified
  in Splunk Web. Splunk recommends that you use Splunk Web to
  configure Network Monitor inputs because it is easy to mistype
  the values for Network Monitor objects.

remoteAddress = <regular expression>
* A regular expression that represents the remote IP address of a
  host that is involved in network communication.
* This setting accepts a regular expression that matches against
  IP addresses only, not host names. For example: 192\.163\..*
* The input includes events for remote IP addresses that match
  the regular expression that you specify here.
* The input filters out events for remote IP addresses that do not
  match the regular expression.
* Defaults to unset (including all remote address events).

```

```

process = <regular expression>
* A regular expression that represents the process or application that
  performed a network access.
* The input includes events for processes that match the
  regular expression that you specify here.
* The input filters out events for processes that do not match the
  regular expression.
* Defaults to unset (including all processes and application events).

user = <regular expression>
* A regular expression that represents the Windows user name that
  performed a network access.
* The input includes events for user names that match the
  regular expression that you specify here.
* The input filters out events for user names that do not match the
  regular expression.
* Defaults to unset (including all user name events).

addressFamily = ipv4;ipv6
* Determines the events to include by network address family.
* Setting ipv4 alone will include only TCP/IP v4 packets, while ipv6 alone
  will include only TCP/IP v6 packets.
* To specify both families, separate them with a semicolon.
  For example: ipv4;ipv6
* Defaults to unset (including events with both address families).

packetType = connect;accept;transport.
* Determines the events to include by network packet type.
* To specify multiple packet types, separate them with a semicolon.
  For example: connect;transport
* Defaults to unset (including events with any packet type).

direction = inbound;outbound
* Determines the events to include by network transport direction.
* To specify multiple directions, separate them with a semicolon.
  For example: inbound;outbound
* Defaults to unset (including events with any direction).

protocol = tcp;udp
* Determines the events to include by network protocol.
* To specify multiple protocols, separate them with a semicolon.
  For example: tcp;udp
* For more information about protocols, see
  http://www.ietf.org/rfc/rfc1700.txt
* Defaults to unset (including events with all protocols).

readInterval = <integer>
* How often, in milliseconds, that the input should read the network
  kernel driver for events.
* Advanced option. Use the default value unless there is a problem
  with input performance.
* Set this to adjust the frequency of calls into the network kernel driver.
* Choosing lower values (higher frequencies) can reduce network
  performance, while higher numbers (lower frequencies) can cause event
  loss.
* The minimum allowed value is 10 and the maximum allowed value is 1000.
* Defaults to unset, handled as 100 (msec).

driverBufferSize = <integer>
* The maximum number of packets that the network kernel driver retains
  for retrieval by the input.
* Set to adjust the maximum number of network packets retained in
  the network driver buffer.
* Advanced option. Use the default value unless there is a problem
  with input performance.
* Configuring this setting to lower values can result in event loss, while
  higher values can increase the size of non-paged memory on the host.
* The minimum allowed value is 128 and the maximum allowed value is 32768.
* Defaults to unset, handled as 32768 (packets).

userBufferSize = <integer>
* The maximum size, in megabytes, of the user mode event buffer.

```

- \* Controls amount of packets cached in the the user mode.
- \* Advanced option. Use the default value unless there is a problem with input performance.
- \* Configuring this setting to lower values can result in event loss, while higher values can increase the amount of memory that the network monitor uses.
- \* The minimum allowed value is 20 and the maximum allowed value is 500.
- \* Defaults to unset, handled as 20 (megabytes).

mode = single|multikv

- \* Specifies how the network monitor input generates events.
- \* Set to 'single' to generate one event per packet, or 'multikv' to generate combined events of many packets in multikv format (many packets described in a single table as one event).
- \* Defaults to single.

multikvMaxEventCount = <integer>

- \* The maximum number of packets to combine in multikv format when you set the 'mode' setting to 'multikv'.
- \* Has no effect when 'mode' is set to 'single'.
- \* Advanced option.
- \* The minimum allowed value is 10 and the maximum allowed value is 500.
- \* Defaults to 100.

multikvMaxTimeMs = <integer>

- \* The maximum amount of time, in milliseconds, to accumulate packet data to combine into a large tabular event in multikv format.
- \* Has no effect when 'mode' is set to 'single'.
- \* Advanced option.
- \* The minimum allowed value is 100 and the maximum allowed value is 5000.
- \* Defaults to 1000.

sid\_cache\_disabled = 0|1

- \* Enables or disables account Security IDentifier (SID) cache.
- \* This setting is global. It affects all Windows Network Monitor stanzas.
- \* Defaults to 0.

sid\_cache\_exp = <time in seconds>

- \* The expiration time for account SID cache entries.
- \* This setting is optional.
- \* This setting is global. It affects all Windows Network Monitor stanzas.
- \* The minimum allowed value is 10 and the maximum allowed value is 31536000.
- \* Defaults to 3600.

sid\_cache\_exp\_neg = <time in seconds>

- \* The expiration time for negative account SID cache entries.
- \* This setting is optional.
- \* This setting is global. It affects all Windows Network Monitor stanzas.
- \* The minimum allowed value is 10 and the maximum allowed value is 31536000.
- \* Defaults to 10.

sid\_cache\_max\_entries = <number of entries>

- \* The maximum number of account SID cache entries.
- \* This setting is optional.
- \* This setting is global. It affects all Windows Network Monitor stanzas.
- \* The minimum allowed value is 10 and the maximum allowed value is 40000.
- \* Defaults to 10.

disabled = 0|1

- \* Whether or not the input is enabled.
- \* Defaults to 0 (enabled.)

index = <string>

- \* The index that this input should send the data to.
- \* This setting is optional.
- \* Defaults to the default index.

[powershell://<name>]

- \* Runs Windows PowerShell version 3 commands or scripts.

script = <command>

- \* A PowerShell command-line script or .ps1 script file that the input

```

    should run.
* There is no default.

schedule = [<number>|<cron schedule>]
* How often to run the specified PowerShell command or script.
* You can specify a number in seconds, or provide a valid cron
  schedule.
* Defaults to running the command or script once, at startup.

[powershell2://<name>]
* Runs Windows PowerShell version 2 commands or scripts.

script = <command>
* A PowerShell command-line script or .ps1 script file that the input
  should run.

schedule = <schedule>
* How often to run the specified PowerShell command or script.
* You can provide a valid cron schedule.
* Defaults to running the command or script once, at startup.

```

## inputs.conf.example

```

#   Version 6.5.0
#
# This is an example inputs.conf. Use this file to configure data inputs.
#
# To use one or more of these configurations, copy the configuration block into
# inputs.conf in $SPLUNK_HOME/etc/system/local/. You must restart Splunk to
# enable configurations.
#
# To learn more about configuration files (including precedence) please see the
# documentation located at
# http://docs.splunk.com/Documentation/Splunk/latest/Admin/Aboutconfigurationfiles

# The following configuration reads all the files in the directory /var/log.

[monitor:///var/log]

# The following configuration reads all the files under /var/log/httpd and
# classifies them as sourcetype::access_common.
#
# When checking a file for new data, if the file's modification time is from
# before seven days ago, the file will no longer be checked for changes
# until you restart the software.

[monitor:///var/log/httpd]
sourcetype = access_common
ignoreOlderThan = 7d

# The following configuration reads all the
# files under /mnt/logs. When the path is /mnt/logs/<host>/... it
# sets the hostname (by file) to <host>.

[monitor:///mnt/logs]
host_segment = 3

# The following configuration listens on TCP port 9997 for raw
# data from ANY remote server (not just a Splunk instance). The host of the
# data is set to the IP address of the remote server.

[tcp://:9997]

# The following configuration listens on TCP port 9995 for raw
# data from ANY remote server. The host of the data is set as the host name of

```

```

# the remote server. All data will also be assigned the sourcetype "log4j" and
# the source "tcp:9995".

[tcp://:9995]
connection_host = dns
sourcetype = log4j
source = tcp:9995

# The following configuration listens on TCP port 9995 for raw
# data from 10.1.1.10.
# All data is assigned the host "webhead-1", the sourcetype "access_common" and
# the the source "//10.1.1.10/var/log/apache/access.log".

[tcp://10.1.1.10:9995]
host = webhead-1
sourcetype = access_common
source = //10.1.1.10/var/log/apache/access.log

# The following configuration listens on TCP port 9996 for
# Splunk cooked event data from ANY splunk forwarder.
# The host of the data is set to the host name of the remote server ONLY IF the
# remote data has no host set, or if it is set to "localhost".

[splunktcp://:9996]
connection_host = dns

# The following configuration listens on TCP port 9996 for
# distributed search data from 10.1.1.100. The data is processed the same as
# locally indexed data.

[splunktcp://10.1.1.100:9996]

# The following configuration listens on TCP port 514 for data
# from syslog.corp.company.net. The data is assigned the sourcetype "syslog"
# and the host is set to the host name of the remote server.

[tcp://syslog.corp.company.net:514]
sourcetype = syslog
connection_host = dns

# Following configuration limits the acceptance of data to forwarders
# that have been configured with the token value specified in 'token' field.
# NOTE: The token value is encrypted. The REST endpoint encrypts the token
# while saving it.

[splunkcptoken://tok1]
token = $!$ifQTPTzHD/BA8VgKvVcg0lKQAt3N1C8S/1uK3nAKIE9dd9e9g==

# Set up Secure Sockets Layer (SSL):

[SSL]
serverCert=$SPLUNK_HOME/etc/auth/server.pem
password=password
rootCA=$SPLUNK_HOME/etc/auth/cacert.pem
requireClientCert=false

[splunktcp-ssl:9996]

# Use file system change monitor:

[fschange:/etc/]
fullEvent=true
pollPeriod=60
recurse=true
sendEventMaxSize=100000
index=main

# Monitor the Security Windows Event Log channel, getting the most recent

```

```

# events first, then older, and finally continuing to gather newly arriving events

[WinEventLog://Security]
disabled = 0
start_from = newest
evt_dc_name =
evt_dns_name =
evt_resolve_ad_ds =
evt_resolve_ad_obj = 1
checkpointInterval = 5

# Monitor the ForwardedEvents Windows Event Log channel, only gathering the
# events that arrive after monitoring starts, going forward in time.

[WinEventLog://ForwardedEvents]
disabled = 0
start_from = oldest
current_only = 1
batch_size = 10
checkpointInterval = 5

[tcp://9994]
queueSize=50KB
persistentQueueSize=100MB

# Perfmon: Windows performance monitoring examples

# You must specify the names of objects, counters and instances
# exactly as they are shown in the Performance Monitor application. Splunk Web
# is the recommended interface to use to configure performance monitor inputs.

# These stanzas gather performance data from the local system only.
# Use wmi.conf for performance monitor metrics on remote systems.

# Query the PhysicalDisk performance object and gather disk access data for
# all physical drives installed in the system. Store this data in the
# "perfmon" index.
# Note: If the interval attribute is set to 0, Splunk will reset the interval
# to 1.

[perfmon://LocalPhysicalDisk]
interval = 0
object = PhysicalDisk
counters = Disk Bytes/sec; % Disk Read Time; % Disk Write Time; % Disk Time
instances = *
disabled = 0
index = PerfMon

# Gather common memory statistics using the Memory performance object, every
# 5 seconds. Store the data in the "main" index. Since none of the counters
# specified have applicable instances, the instances attribute is not required.

[perfmon://LocalMainMemory]
interval = 5
object = Memory
counters = Committed Bytes; Available Bytes; % Committed Bytes In Use
disabled = 0
index = main

# Gather data on USB activity levels every 10 seconds. Store this data in the
# default index.

[perfmon://USBChanges]
interval = 10
object = USB
counters = Usb Control Data Bytes/Sec
instances = *
disabled = 0

# Admon: Windows Active Directory monitoring examples

# Monitor the default domain controller (DC) for the domain that the computer

```

```
# running Splunk belongs to. Start monitoring at the root node of Active
# Directory.
[admon://NearestDC]
targetDc =
startingNode =

# Monitor a specific DC, with a specific starting node. Store the events in
# the "admon" Splunk index. Do not print Active Directory schema. Do not
# index baseline events.

[admon://DefaultTargetDC]
targetDc = pri01.eng.ad.splunk.com
startingNode = OU=Computers,DC=eng,DC=ad,DC=splunk,DC=com
index = admon
printSchema = 0
baseline = 0

# Monitor two different DCs with different starting nodes.
[admon://DefaultTargetDC]
targetDc = pri01.eng.ad.splunk.com
startingNode = OU=Computers,DC=eng,DC=ad,DC=splunk,DC=com

[admon://SecondTargetDC]
targetDc = pri02.eng.ad.splunk.com
startingNode = OU=Computers,DC=hr,DC=ad,DC=splunk,DC=com
```

## instance.cfg.conf

以下为 instance.cfg.conf 的规范和示例文件。

### instance.cfg.conf.spec

```
# Version 6.5.0
#
# This file contains the set of attributes and values you can expect to find in
# the SPLUNK_HOME/etc/instance.cfg file; the instance.cfg file is not to be
# modified or removed by user. LEAVE THE instance.cfg FILE ALONE.
#
#
```

### 全局设置

```
# GLOBAL SETTINGS
# The [general] stanza defines global settings.
#
```

#### [general]

```
[general]
guid = <GUID in all-uppercase>
* This setting formerly (before 5.0) belonged in the [general] stanza of
  server.conf file.

* Splunk expects that every Splunk instance will have a unique string for this
  value, independent of all other Splunk instances. By default, Splunk will
  arrange for this without user intervention.

* Currently used by (not exhaustive):
  * Clustering environments, to identify participating nodes.
  * Splunk introspective searches (Splunk on Splunk, Deployment Monitor,
    etc.), to identify forwarders.

* At startup, the following happens:

  * If server.conf has a value of 'guid' AND instance.cfg has no value of
    'guid', then the value will be erased from server.conf and moved to
```

```

instance.cfg file.

* If server.conf has a value of 'guid' AND instance.cfg has a value of
'guid' AND these values are the same, the value is erased from
server.conf file.

* If server.conf has a value of 'guid' AND instance.cfg has a value of 'guid'
AND these values are different, startup halts and error is shown. Operator
must resolve this error. We recommend erasing the value from server.conf
file, and then restarting.

* If you are hitting this error while trying to mass-clone Splunk installs,
please look into the command 'splunk clone-prep-clear-config';
'splunk help' has help.

* See http://www.ietf.org/rfc/rfc4122.txt for how a GUID (a.k.a. UUID) is
constructed.

* The standard regexp to match an all-uppercase GUID is
"[0-9A-F]{8}-[0-9A-F]{4}-[0-9A-F]{4}-[0-9A-F]{4}-[0-9A-F]{12}".

```

## instance.cfg.conf.example

```

# Version 6.5.0
#
# This file contains an example SPLUNK_HOME/etc/instance.cfg file; the
# instance.cfg file is not to be modified or removed by user. LEAVE THE
# instance.cfg FILE ALONE.
#

[general]
guid = B58A86D9-DF3D-4BF8-A426-DB85C231B699

```

## limits.conf

以下为 limits.conf 的规范和示例文件。

### limits.conf.spec

```

# Version 6.5.0
#
# This file contains possible attribute/value pairs for configuring limits for
# search commands.
#
# There is a limits.conf in $SPLUNK_HOME/etc/system/default/. To set custom
# configurations, place a limits.conf in $SPLUNK_HOME/etc/system/local/. For
# examples, see limits.conf.example. You must restart Splunk to enable
# configurations.
#
# To learn more about configuration files (including precedence) please see the
# documentation located at
# http://docs.splunk.com/Documentation/Splunk/latest/Admin/Aboutconfigurationfiles
#

# limits.conf settings and DISTRIBUTED SEARCH
#
# Unlike most settings which affect searches, limits.conf settings are not
# provided by the search head to be used by the search peers. This means
# that if you need to alter search-affecting limits in a distributed
# environment, typically you will need to modify these settings on the
# relevant peers and search head for consistent results.

```

### 全局设置

```

# GLOBAL SETTINGS
# Use the [default] stanza to define any global settings.

```



```
# * You can also define global settings outside of any stanza, at the top of
#   the file.
# * Each conf file should have at most one default stanza. If there are
#   multiple default stanzas, attributes are combined. In the case of
#   multiple definitions of the same attribute, the last definition in the
#   file wins.
# * If an attribute is defined at both the global level and in a specific
#   stanza, the value in the specific stanza takes precedence.

# CAUTION: Do not alter the settings in limits.conf unless you know what you
#           are doing. Improperly configured limits may result in splunkd
#           crashes and/or memory overuse.

* Each stanza controls different parameters of search commands.
```

## **[default]**

```
[default]
max_mem_usage_mb = <non-negative integer>
* Provides a limitation to the amount of RAM a batch of events or results will
  use in the memory of a search process.
* Operates on an estimation of memory use which is not exact.
* The limitation is applied in an unusual way; if the number of results or
  events exceeds maxresults, AND the estimated memory exceeds this limit, the
  data is spilled to disk.
* This means, as a general rule, lower limits will cause a search to use more
  disk I/O and less RAM, and be somewhat slower, but should cause the same
  results to typically come out of the search in the end.
* This limit is applied currently to a number, but not all search processors.
  However, more will likely be added as it proves necessary.
* The number is thus effectively a ceiling on batch size for many components of
  search for all searches run on this system.
* 0 will specify the size to be unbounded. In this case searches may be
  allowed to grow to arbitrary sizes.

* The 'mvexpand' command uses this value in a different way.
  * mvexpand has no combined logic with maxresults
  * If the memory limit is exceeded, output is truncated, not spilled to disk.

* The 'stats' processor uses this value in the following way.
* If the estimated memory usage exceeds the specified limit, the results are spilled to disk
* If '0' is specified, the results are spilled to the disk when the number of results
  exceed the maxresultrows setting.

* This value is not exact. The estimation can deviate by an order of magnitude
  or so to both the smaller and larger sides.
* Defaults to 200 (MB)
```

```
min_batch_size_bytes = <integer>
* Specifies the size of the file/tar after which the file is handled by the
  batch reader instead of the trailing processor.
* Global parameter, cannot be configured per input.
* Note configuring this to a very small value could lead to backing up of jobs
  at the tailing processor.
* Defaults to 20 MB.
```

```
DelayArchiveProcessorShutdown = <bool>
* Specifies whether during splunk shutdown archive processor should finish processing archive file under process.
* If set to false archive processor abandons further processing of archive file and will process again from start
  again.
* If set to true archive processor will complete processing of archive file. Shutdown will be delayed.
* defaults to false
```

## **[searchresults]**

```
[searchresults]
* This stanza controls search results for a variety of Splunk search commands.

maxresultrows = <integer>
* Configures the maximum number of events are generated by search commands
```

which grow the size of your result set (such as multikv) or that create events. Other search commands are explicitly controlled in specific stanzas below.

- \* This limit should not exceed 50000. Setting this limit higher than 50000 causes instability.
- \* Defaults to 50000.

`tocsv_maxretry = <integer>`

- \* Maximum number of times to retry the atomic write operation.
- \* 1 = no retries.
- \* Defaults to 5.

`tocsv_retryperiod_ms = <integer>`

- \* Period of time to wait before each retry.
- \* Defaults to 500.

- \* These setting control logging of error messages to info.csv
- All messages will be logged to search.log regardless of these settings.

`compression_level = <integer>`

- \* Compression level to use when writing search results to .csv.gz files
- \* Defaults to 1

## **[search\_info]**

[search\_info]

- \* This stanza controls logging of messages to the info.csv file
- \* Messages logged to info.csv are available to REST API clients and the Splunk UI, so limiting the messages added to info.csv will mean that these messages will not be available in the UI and/or the REST API.

`max_infocsv_messages = <positive integer>`

- \* If more than max\_infocsv\_messages log entries are generated, additional entries will not be logged in info.csv. All entries will still be logged in search.log.

`infocsv_log_level = [DEBUG|INFO|WARN|ERROR]`

- \* Limits the messages which are added to info.csv to the stated level and above.
- \* For example, if log\_level is WARN, messages of type WARN and higher will be added to info.csv

`show_warn_on_filtered_indexes = <boolean>`

- \* Log warnings if search returns no results because user has no permissions to search on queried indexes

`filteredindexes_log_level = [DEBUG|INFO|WARN|ERROR]`

- \* Log level of messages when search results no results because user has no permissions to search on queries indexes

## **[subsearch]**

[subsearch]

- \* This stanza controls subsearch results.
- \* NOTE: This stanza DOES NOT control subsearch results when a subsearch is called by commands such as join, append, or appendcols.
- \* Read more about subsearches in the online documentation: <http://docs.splunk.com/Documentation/Splunk/latest/Search/Aboutsubsearches>

`maxout = <integer>`

- \* Maximum number of results to return from a subsearch.
- \* This value cannot be greater than or equal to 10500.
- \* Defaults to 10000.

`maxtime = <integer>`

- \* Maximum number of seconds to run a subsearch before finalizing
- \* Defaults to 60.

```
ttl = <integer>
* Time to cache a given subsearch's results, in seconds.
* Do not set this below 120 seconds.
* See definition in [search] ttl for more details on how the ttl is computed
* Defaults to 300.
```

### **[anomalousvalue]**

```
[anomalousvalue]
maxresultrows = <integer>
* Configures the maximum number of events that can be present in memory at one
  time.
* Defaults to searchresults::maxresultsrows (which is by default 50000).

maxvalues = <integer>
* Maximum number of distinct values for a field.
* Defaults to 100000.

maxvaluesize = <integer>
* Maximum size in bytes of any single value (truncated to this size if larger).
* Defaults to 1000.
```

### **[associate]**

```
[associate]
maxfields = <integer>
* Maximum number of fields to analyze.
* Defaults to 10000.

maxvalues = <integer>
* Maximum number of values for any field to keep track of.
* Defaults to 10000.

maxvaluesize = <integer>
* Maximum length of a single value to consider.
* Defaults to 1000.
```

### **[autoregress]**

```
[autoregress]
maxp = <integer>
* Maximum valid period for auto regression
* Defaults to 10000.

maxrange = <integer>
* Maximum magnitude of range for p values when given a range.
* Defaults to 1000.
```

### **[concurrency]**

```
[concurrency]
max_count = <integer>
* Maximum number of detected concurrencies.
* Defaults to 10000000
```

### **[ctable]**

```
[ctable]
* This stanza controls the contingency, ctable, and counttable commands.

maxvalues = <integer>
* Maximum number of columns/rows to generate (the maximum number of distinct
  values for the row field and column field).
* Defaults to 1000.
```

### **[correlate]**

```
[correlate]
maxfields = <integer>
* Maximum number of fields to correlate.
* Defaults to 1000.
```

## **[discretize]**

```
[discretize]
* This stanza set attributes for bin/bucket/discretize.

default_time_bins = <integer>
* When discretizing time for timechart or explicitly via bin, the default bins
  to use if no span or bins is specified.
* Defaults to 100

maxbins = <integer>
* Maximum number of buckets to discretize into.
* If maxbins is not specified or = 0, it defaults to
  searchresults::maxresultrows
* Defaults to 50000.
```

## **[export]**

```
[export]
add_timestamp = <bool>
* Add a epoch time timestamp to JSON streaming output that reflects the time
  the results were generated/retrieved
* Defaults to false

add_offset = <bool>
* Add an offset/row number to JSON streaming output
* Defaults to true
```

## **[extern]**

```
[extern]
perf_warn_limit = <integer>
* Warn when external scripted command is applied to more than this many events
* set to 0 for no message (message is always INFO level)
* Defaults to 10000
```

## **[inputcsv]**

```
[inputcsv]
mkdir_max_retries = <integer>
* Maximum number of retries for creating a tmp directory (with random name as
  subdir of SPLUNK_HOME/var/run/splunk)
* Defaults to 100.
```

## **[indexpreview]**

```
[indexpreview]
max_preview_bytes = <integer>
* Maximum number of bytes to read from each file during preview
* Defaults to 2000000 (2 MB)

max_results_perchunk = <integer>
* Maximum number of results to emit per call to preview data generator
* Defaults to 2500.

soft_preview_queue_size = <integer>
* Loosely-applied maximum on number of preview data objects held in memory
* Defaults to 100.
```

## **[join]**

```
[join]
subsearch_maxout = <integer>
* Maximum result rows in output from subsearch to join against.
* Defaults to 50000.

subsearch_maxtime = <integer>
* Maximum search time (in seconds) before auto-finalization of subsearch.
* Defaults to 60

subsearch_timeout = <integer>
* Maximum time to wait for subsearch to fully finish (in seconds).
* Defaults to 120.
```

## **[kmeans]**

```
[kmeans]
maxdatapoints = <integer>
* Maximum data points to do kmeans clusterings for.
* Defaults to 100000000.

maxkvalue = <integer>
* Maximum number of clusters to attempt to solve for.
* Defaults to 1000.

maxkrange = <integer>
* Maximum number of k values to iterate over when specifying a range.
* Defaults to 100.
```

## **[kv]**

```
[kv]
maxcols = <integer>
* When non-zero, the point at which kv should stop creating new fields.
* Defaults to 512.

limit = <integer>
* Maximum number of keys auto kv can generate.
* Defaults to 100.

maxchars = <integer>
* Truncate _raw to this size and then do auto KV.
* Defaults to 10240 characters.

max_extractor_time = <integer>
* Maximum amount of CPU time, in milliseconds, that a key-value pair extractor
  will be allowed to take before warning. If the extractor exceeds this
  execution time on any event a warning will be issued
* Defaults to 1000.

avg_extractor_time = <integer>
* Maximum amount of CPU time, in milliseconds, that the average (over search
  results) execution time of a key-value pair extractor will be allowed to take
  before warning. Once the average becomes larger than this amount of time a
  warning will be issued
* Defaults to 500
```

## **[lookup]**

```
[lookup]
max_memtable_bytes = <integer>
* Maximum size of static lookup file to use an in-memory index for.
* Defaults to 100000000 in bytes (10MB)
* Lookup files with size above max_memtable_bytes will be indexed on disk
* A large value results in loading large lookup files in memory leading to bigger process memory footprint.
* Caution must be exercised when setting this parameter to arbitrarily high values!
```

```

max_matches = <integer>
* maximum matches for a lookup
* range 1 - 1000
* Defaults to 1000

max_reverse_matches = <integer>
* maximum reverse lookup matches (for search expansion)
* Defaults to 50

batch_index_query = <bool>
* Should non-memory file lookups (files that are too large) use batched queries
  to possibly improve performance?
* Defaults to true

batch_response_limit = <integer>
* When doing batch requests, the maximum number of matches to retrieve
  if more than this limit of matches would otherwise be retrieve, we will fall
  back to non-batch mode matching
* Defaults to 5000000

max_lookup_messages = <positive integer>
* If more than "max_lookup_messages" log entries are generated, additional
  entries will not be logged in info.csv. All entries will still be logged in
  search.log.

```

## **[metrics]**

```

[metrics]
maxseries = <integer>
* The number of series to include in the per_x_thruput reports in metrics.log.
* Defaults to 10.

interval = <integer>
* Number of seconds between logging splunkd metrics to metrics.log.
* Minimum of 10.
* Defaults to 30.

```

## **[metrics:tcpin\_connections]**

```

[metrics:tcpin_connections]
aggregate_metrics = [true|false]
* For each splunktcp connection from forwarder, splunk logs metrics information
  every metrics interval.
* When there are large number of forwarders connected to indexer, the amount of
  information logged can take lot of space in metrics.log. When set to true, it
  will aggregate information across each connection and report only once per
  metrics interval.
* Defaults to false

suppress_derived_info = [true|false]
* For each forwarder connection, _tcp_Bps, _tcp_KBps, _tcp_avg_thruput,
  _tcp_Kprocessed is logged in metrics.log.
* This can be derived from kb. When set to true, the above derived info will
  not be emitted.
* Defaults to true

```

## **[rare]**

```

[rare]
maxresultrows = <integer>
* Maximum number of result rows to create.
* If not specified, defaults to searchresults::maxresultrows
* Defaults to 50000.

maxvalues = <integer>
* Maximum number of distinct field vector values to keep track of.
* Defaults 100000.

```

maxvaluesize = <integer>  
\* Maximum length of a single value to consider.  
\* Defaults to 1000.

## **[restapi]**

[restapi]  
maxresultrows = <integer>  
\* Maximum result rows to be returned by /events or /results getters from REST API.  
\* Defaults to 50000.  
  
time\_format\_reject = <regular expression>  
\* HTTP parameters for time\_format and output\_time\_format which match this regex will be rejected (blacklisted).  
\* The regex will be satisfied by a substring match anywhere in the parameter.  
\* Intended as defense-in-depth against XSS style attacks against browser users by crafting specially encoded URLs for them to access splunkd.  
\* If unset, all parameter strings will be accepted.  
\* To disable this check entirely, set the value to empty.  
# Example of disabling: time\_format\_reject =  
\* Defaults to [<>!], which means that the less-than '<', greater-than '>', and exclamation point '!' are not allowed.  
  
jobscontentmaxcount = <integer>  
\* Maximum length of a property in the contents dictionary of an entry from /jobs getter from REST API  
\* Value of 0 disables truncation  
\* Defaults to 0

## **[search\_metrics]**

[search\_metrics]  
debug\_metrics = <bool>  
\* This indicates whether we should output more detailed search metrics for debugging.  
\* This will do things like break out where the time was spent by peer, and may add additional deeper levels of metrics.  
\* This is NOT related to "metrics.log" but to the "Execution Costs" and "Performance" fields in the Search inspector, or the count\_map in the info.csv file.  
\* Defaults to false

## **[search]**

[search]  
summary\_mode = [all|only|none]  
\* Controls if precomputed summary are to be used if possible?  
\* all: use summary if possible, otherwise use raw data  
\* only: use summary if possible, otherwise do not use any data  
\* none: never use precomputed summary data  
\* Defaults to 'all'  
  
result\_queue\_max\_size = <integer>  
\* Controls the size of the search results queue in dispatch  
\* Default size is set to 100MB  
\* Use caution while playing with this parameter  
  
use\_bloomfilter = <bool>  
\* Control whether to use bloom filters to rule out buckets  
\* Default value set to true  
  
max\_id\_length = <integer>  
\* Maximum length of custom search job id when spawned via REST API arg id=  
  
ttl = <integer>  
\* How long search artifacts should be stored on disk once completed, in seconds. The ttl is computed relative to the modtime of status.csv of the job

if such file exists or the modtime of the search job's artifact directory. If a job is being actively viewed in the Splunk UI then the modtime of status.csv is constantly updated such that the reaper does not remove the job from underneath.

- \* Defaults to 600, which is equivalent to 10 minutes.

failed\_job\_ttl = <integer>

- \* How long search artifacts should be stored on disk once failed, in seconds. The ttl is computed relative to the modtime of status.csv of the job if such file exists or the modtime of the search job's artifact directory. If a job is being actively viewed in the Splunk UI then the modtime of status.csv file is constantly updated such that the reaper does not remove the job from underneath.
- \* Defaults to 86400, which is equivalent to 24 hours.

default\_save\_ttl = <integer>

- \* How long the ttl for a search artifact should be extended in response to the save control action, in second. 0 = indefinitely.
- \* Defaults to 604800 (1 week)

remote\_ttl = <integer>

- \* How long artifacts from searches run in behalf of a search head should be stored on the indexer after completion, in seconds.
- \* Defaults to 600 (10 minutes)

status\_buckets = <integer>

- \* The approximate maximum number buckets to generate and maintain in the timeline.
- \* Defaults to 0, which means do not generate timeline information.

max\_bucket\_bytes = <integer>

- \* This setting has been deprecated and has no effect

max\_count = <integer>

- \* The number of events that can be accessible in any given status bucket (when status\_buckets = 0).
- \* The last accessible event in a call that takes a base and bounds.
- \* Defaults to 500000.
- \* Note: This value does not reflect the number of events displayed on the UI after the search is evaluated/computed.

max\_events\_per\_bucket = <integer>

- \* For searches with status\_buckets>0 this will limit the number of events retrieved per timeline bucket.
- \* Defaults to 1000 in code.

truncate\_report = [1|0]

- \* Specifies whether or not to apply the max\_count limit to report output.
- \* Defaults to false (0).

min\_prefix\_len = <integer>

- \* The minimum length of a prefix before a \* to ask the index about.
- \* Defaults to 1.

cache\_ttl = <integer>

- \* The length of time to persist search cache entries (in seconds).
- \* Defaults to 300.

max\_results\_perchunk = <integer>

- \* Maximum results per call to search (in dispatch), must be less than or equal to maxresultrows.
- \* Defaults to 2500

min\_results\_perchunk = <integer>

- \* Minimum results per call to search (in dispatch), must be less than or equal to max\_results\_perchunk.
- \* Defaults to 100

max\_rawsize\_perchunk = <integer>

- \* Maximum raw size of results per call to search (in dispatch).
- \* 0 = no limit.
- \* Defaults to 100000000 (100MB)
- \* Not affected by chunk\_multiplier

target\_time\_perchunk = <integer>

- \* Target duration of a particular call to fetch search results in ms.



```

* Defaults to 2000

long_search_threshold = <integer>
* Time in seconds until a search is considered "long running".
* Defaults to 2

chunk_multiplier = <integer>
* max_results_perchunk, min_results_perchunk, and target_time_perchunk are
  multiplied by this for a long running search.
* Defaults to 5

min_freq = <number>
* Minimum frequency of a field required for including in the /summary endpoint
  as a fraction (>=0 and <=1).
* Defaults is 0.01 (1%)

reduce_freq = <integer>
* Attempt to reduce intermediate results every how many chunks (0 = never).
* Defaults to 10

reduce_duty_cycle = <number>
* The maximum time to spend doing reduce, as a fraction of total search time
* Must be > 0.0 and < 1.0
* Defaults to 0.25

preview_duty_cycle = <number>
* The maximum time to spend generating previews, as a fraction of total search time
* Must be > 0.0 and < 1.0
* Defaults to 0.25

min_preview_period = <integer>
* This is the minimum time in seconds required between previews, used to limit cases where
  the interval calculated using the preview_duty_cycle parameter is very small, indicating
  that previews should be run frequently.
* Defaults to 1.

max_preview_period = <integer>
* This is the maximum time, in seconds, between previews. Used with the preview interval that
  is calculated with the preview_duty_cycle parameter. '0' indicates unlimited.
* Defaults to 0.

results_queue_min_size = <integer>
* The minimum size for the queue of results that will be kept from peers for
  processing on the search head.
* The queue will be the max of this and the number of peers providing results.
* Defaults to 10

dispatch_quota_retry = <integer>
* The maximum number of times to retry to dispatch a search when the quota has
  been reached.
* Defaults to 4

dispatch_quota_sleep_ms = <integer>
* Milliseconds between retrying to dispatch a search if a quota has been
  reached.
* Retries the given number of times, with each successive wait 2x longer than
  the previous.
* Defaults to 100

base_max_searches = <int>
* A constant to add to the maximum number of searches, computed as a multiplier
  of the CPUs.
* Defaults to 6

max_searches_per_cpu = <int>
* The maximum number of concurrent historical searches per CPU. The system-wide
  limit of historical searches is computed as:
  max_hist_searches = max_searches_per_cpu x number_of_cpus + base_max_searches
* Note: the maximum number of real-time searches is computed as:
  max_rt_searches = max_rt_search_multiplier x max_hist_searches
* Defaults to 1

```

```

max_rt_search_multiplier = <decimal number>
* A number by which the maximum number of historical searches is multiplied to
  determine the maximum number of concurrent real-time searches
* Note: the maximum number of real-time searches is computed as:
  max_rt_searches = max_rt_search_multiplier x max_hist_searches
* Defaults to 1

max_macro_depth = <int>
* Max recursion depth for macros.
* Considered a search exception if macro expansion doesn't stop after this many
  levels.
* Must be greater than or equal to 1.
* Default is 100

max_subsearch_depth = <int>
* max recursion depth for subsearch
* considered a search exception if subsearch doesn't stop after this many levels

realtime_buffer = <int>
* Maximum number of accessible events to keep for real-time searches from
  Splunk Web.
* Acts as circular buffer once this limit is reached
* Must be greater than or equal to 1
* Default is 10000

stack_size = <int>
* The stack size (in bytes) of the thread executing the search.
* Defaults to 4194304 (4 MB)

status_cache_size = <int>
* The number of search job status data splunkd can cache in RAM. This cache
  improves performance of the jobs endpoint
* Defaults to 10000

timeline_freq = <timespan> or <ratio>
* Minimum amount of time between timeline commits.
* If specified as a number < 1 (and > 0), minimum time between commits is
  computed as a ratio of the amount of time that the search has been running.
* defaults to 0 seconds

preview_freq = <timespan> or <ratio>
* Minimum amount of time between results preview updates.
* If specified as a number < 1 (and > 0), minimum time between previews is
  computed as a ratio of the amount of time that the search has been running,
  or as a ratio of the length of the time window for real-time windowed
  searches.
* Defaults to ratio of 0.05

max_combiner_memevents = <int>
* Maximum size of in-memory buffer for search results combiner, in terms of
  number of events.
* Defaults to 50000 events.

replication_period_sec = <int>
* The minimum amount of time in seconds between two successive bundle
  replications.
* Defaults to 60

replication_file_ttl = <int>
* The TTL (in seconds) of bundle replication tarballs, i.e. *.bundle files.
* Defaults to 600 (10m)

sync_bundle_replication = [0|1|auto]
* Flag indicating whether configuration file replication blocks searches or is
  run asynchronously
* When setting this flag to auto Splunk will choose to use asynchronous
  replication if and only if all the peers support async bundle replication,
  otherwise it will fall back into sync replication.
* Defaults to auto

rr_min_sleep_ms = <int>
* Minimum time to sleep when reading results in round-robin mode when no data

```

is available.

- \* Defaults to 10.

`rr_max_sleep_ms = <int>`

- \* Maximum time to sleep when reading results in round-robin mode when no data is available.
- \* Defaults to 1000

`rr_sleep_factor = <int>`

- \* If no data is available even after sleeping, increase the next sleep interval by this factor.
- \* defaults to 2

`fieldstats_update_freq = <number>`

- \* How often to update the field summary statistics, as a ratio to the elapsed run time so far.
- \* Smaller values means update more frequently. 0 means as frequently as possible.
- \* Defaults to 0

`fieldstats_update_maxperiod = <number>`

- \* Maximum period for updating field summary statistics in seconds
- \* 0 means no maximum, completely dictated by `current_run_time`
- \* `fieldstats_update_freq`
- \* Fractional seconds are allowed.
- \* defaults to 60

`timeline_events_preview = <bool>`

- \* Set `timeline_events_preview` to "true" to display events in the Search app as the events are scanned, including events that are in-memory and not yet committed, instead of waiting until all of the events are scanned to see the search results.
- \* When set to "true", you will not be able to expand the event information in the event viewer until events are committed.
- \* When set to "false", events are displayed only after the events are committed (the events are written to the disk).
- \* This setting might increase disk usage to temporarily save uncommitted events while the search is running. Additionally, search performance might be impacted.
- \* Defaults to false.

`remote_timeline = [0|1]`

- \* If true, allows the timeline to be computed remotely to enable better map/reduce scalability.
- \* defaults to true (1).

`remote_timeline_prefetch = <int>`

- \* Each peer should proactively send at most this many full events at the beginning
- \* Defaults to 100.

`remote_timeline_parallel_fetch = <bool>`

- \* Connect to multiple peers at the same time when fetching remote events?
- \* Defaults to true

`remote_timeline_min_peers = <int>`

- \* Minimum search peers for enabling remote computation of timelines.
- \* Defaults to 1 (1).

`remote_timeline_fetchall = [0|1]`

- \* If set to true (1), Splunk fetches all events accessible through the timeline from the remote peers before the job is considered done.
  - \* Fetching of all events may delay the finalization of some searches, typically those running in verbose mode from the main Search view in Splunk Web.
  - \* This potential performance impact can be mitigated by lowering the `max_events_per_bucket` settings.
- \* If set to false (0), the search peers may not ship all matching events to the search-head, particularly if there is a very large number of them.
  - \* Skipping the complete fetching of events back to the search head will result in prompt search finalization.
  - \* Some events may not be available to browse in the UI.
- \* This setting does *not* affect the accuracy of search results computed by reporting searches.
- \* Defaults to true (1).

```

remote_timeline_thread = [0|1]
* If true, uses a separate thread to read the full events from remote peers if
  remote_timeline is used and remote_timeline_fetchall is set to true. (Has no
  effect if remote_timeline or remote_timeline_fetchall is false).
* Defaults to true (1).

remote_timeline_max_count = <int>
* Maximum number of events to be stored per timeline bucket on each search
  peer.
* Defaults to 10000

remote_timeline_max_size_mb = <int>
* Maximum size of disk that remote timeline events should take on each peer
* If limit is reached, a DEBUG message is emitted (and should be visible from
  job inspector/messages
* Defaults to 100

remote_timeline_touchperiod = <number>
* How often to touch remote timeline artifacts to keep them from being deleted
  by the remote peer, while a search is running.
* In seconds, 0 means never. Fractional seconds are allowed.
* Defaults to 300.

remote_timeline_connection_timeout = <int>
* Connection timeout in seconds for fetching events processed by remote peer
  timeliner.
* Defaults to 5.

remote_timeline_send_timeout = <int>
* Send timeout in seconds for fetching events processed by remote peer
  timeliner.
* Defaults to 10.

remote_timeline_receive_timeout = <int>
* Receive timeout in seconds for fetching events processed by remote peer
  timeliner.
* Defaults to 10.

remote_event_download_initialize_pool = <int>
* Size of thread pool responsible for initiating the remote event fetch.
* Defaults to 5.

remote_event_download_finalize_pool = <int>
* Size of thread pool responsible for writing out the full remote events.
* Defaults to 5.

remote_event_download_local_pool = <int>
* Size of thread pool responsible for reading full local events.
* Defaults to 5.

default_allow_queue = [0|1]
* Unless otherwise specified via REST API argument should an async job spawning
  request be queued on quota violation (if not, an http error of server too
  busy is returned)
* Defaults to true (1).

queued_job_check_freq = <number>
* Frequency with which to check queued jobs to see if they can be started, in
  seconds
* Fractional seconds are allowed.
* Defaults to 1.

enable_history = <bool>
* Enable keeping track of searches?
* Defaults to true

max_history_length = <int>
* Max number of searches to store in history (per user/app)
* Defaults to 1000

allow_inexact_metasearch = <bool>
* Should a metasearch that is inexact be allow. If so, an INFO message will be

```

added to the inexact metasearches. If not, a fatal exception will occur at search parsing time.

- \* Defaults to false

indexed\_as\_exact\_metasearch = <bool>

- \* Should we allow a metasearch to treat <field>=<value> the same as <field>::<value> if <field> is an indexed field. Allowing this will allow a larger set of metasearches when allow\_inexact\_metasearch is set to false. However, some of these searches may be inconsistent with the results of doing a normal search.
- \* Defaults to false

dispatch\_dir\_warning\_size = <int>

- \* The number of jobs in the dispatch directory when to issue a bulletin message warning that performance could be impacted
- \* Defaults to 5000

allow\_reuse = <bool>

- \* Allow normally executed historical searches to be implicitly re-used for newer requests if the newer request allows it?
- \* Defaults to true

track\_indeftime\_range = <bool>

- \* Track the \_indeftime range of returned search results?
- \* Defaults to true

reuse\_map\_maxsize = <int>

- \* Maximum number of jobs to store in the reuse map
- \* Defaults to 1000

status\_period\_ms = <int>

- \* The minimum amount of time, in milliseconds, between successive status/info.csv file updates
- \* This ensures search does not spend significant time just updating these files.
  - \* This is typically important for very large number of search peers.
  - \* It could also be important for extremely rapid responses from search peers, when the search peers have very little work to do.
- \* Defaults to 1000 (1 second)

search\_process\_mode = auto | traditional | debug <debugging-command> [debugging-args ...]

- \* Control how search processes are started
- \* When set to "traditional", Splunk initializes each search process completely from scratch
- \* When set to a string beginning with "debug", Splunk routes searches through the given command, allowing the user the to "plug in" debugging tools
  - \* The <debugging-command> must reside in one of
    - \* \$SPLUNK\_HOME/etc/system/bin/
    - \* \$SPLUNK\_HOME/etc/apps/\$YOUR\_APP/bin/
    - \* \$SPLUNK\_HOME/bin/scripts/
  - \* Splunk will pass <debugging-args>, followed by the search command it would normally run, to <debugging-command>
  - \* For example, given:
 

```
search_process_mode = debug $SPLUNK_HOME/bin/scripts/search-debugger.sh 5
```

 Splunk will run a command that looks generally like:
 

```
$SPLUNK_HOME/bin/scripts/search-debugger.sh 5 splunkd search --id=... --maxbuckets=... --ttl=... [...]
```
- \* Defaults to "auto"

max\_searches\_per\_process = <int>

- \* On UNIX we can run more than one search per process; after a search completes its process can wait for another search to be started and let itself be reused
- \* When set to 1 (or 0), we'll never reuse a process
- \* When set to a negative value, we won't limit the number of searches a process can run
- \* When set to a number larger than one, we will let the process run up to that many searches before exiting
- \* Defaults to 500
- \* Has no effect on Windows, or if search\_process\_mode is not "auto"

max\_time\_per\_process = <number>

- \* When running more than one search per process, this limits how much time a process can accumulate running searches before it must exit

- \* When set to a negative value, we won't limit the amount of time a search process can spend running
- \* Defaults to 300.0 (seconds)
- \* Has no effect on Windows, if search\_process\_mode is not "auto", or if max\_searches\_per\_process is set to 0 or 1
- \* NOTE: a search can run longer than this without being terminated, this ONLY prevents that process from being used to run more searches afterwards.

process\_max\_age = <number>

- \* When running more than one search per process, don't reuse a process if it is older than this number of seconds
- \* When set to a negative value, we won't limit the age of a search process
- \* This is different than "max\_time\_per\_process" because it includes time the process spent idle
- \* Defaults to 7200.0 (seconds)
- \* Has no effect on Windows, if search\_process\_mode is not "auto", or if max\_searches\_per\_process is set to 0 or 1
- \* NOTE: a search can run longer than this without being terminated, this ONLY prevents that process from being used to run more searches afterwards.

idle\_process\_reaper\_period = <number>

- \* When allowing more than one search to run per process, we'll periodically check if we have too many idle search processes
- \* Defaults to 30.0 (seconds)
- \* Has no effect on Windows, if search\_process\_mode is not "auto", or if max\_searches\_per\_process is set to 0 or 1

process\_min\_age\_before\_user\_change = <number>

- \* When allowing more than one search to run per process, we'll try to reuse an idle process that last ran a search by the same Splunk user
- \* If no such idle process exists, we'll try using a process from a different user, but only if it has been idle for at least this long
- \* When set to zero, we'll always allow an idle process to be reused by any Splunk user
- \* When set to a negative value, we'll only allow a search process to be used by same Splunk user each time
- \* Defaults to 4.0 (seconds)
- \* Has no effect on Windows, if search\_process\_mode is not "auto", or if max\_searches\_per\_process is set to 0 or 1

launcher\_threads = <int>

- \* When allowing more than one search to run per process, we'll run this many server threads to manage those processes
- \* Defaults to -1 (meaning pick a value automatically)
- \* Has no effect on Windows, if search\_process\_mode is not "auto", or if max\_searches\_per\_process is set to 0 or 1

launcher\_max\_idle\_checks = <int>

- \* When allowing more than one search to run per process, we'll try to find an appropriate idle process to use
- \* This controls how many idle processes we will inspect before giving up and starting a new one
- \* When set to a negative value, we'll inspect every eligible idle process
- \* Defaults to 5
- \* Has no effect on Windows, if search\_process\_mode is not "auto", or if max\_searches\_per\_process is set to 0 or 1

max\_old\_bundle\_idle\_time = <number>

- \* When reaping idle search processes, allow one to be reaped if it is not configured with the most recent configuration bundle, and its bundle hasn't been used in at least this long
- \* When set to a negative value, we won't reap idle processes sooner than normal if they might be using an older configuration bundle
- \* Defaults to 5.0 (seconds)
- \* Has no effect on Windows, if search\_process\_mode is not "auto", or if max\_searches\_per\_process is set to 0 or 1

idle\_process\_cache\_timeout = <number>

- \* When a search process is allowed to run more than one search, it can cache some data between searches
- \* If a search process is idle for this long, take the opportunity to purge some older data from these caches

- \* When set to a negative value, we won't do any purging based on how long the search process is idle
- \* When set to zero, we'll always purge no matter if we're kept idle or not
- \* Defaults to 0.5 (seconds)
- \* Has no effect on Windows, if search\_process\_mode is not "auto", or if max\_searches\_per\_process is set to 0 or 1

idle\_process\_cache\_search\_count = <int>

- \* When a search process is allowed to run more than one search, it can cache some data between searches
- \* If a search process has run this many searches without purging older data from the cache, do it even if the "idle\_process\_cache\_timeout" has not been hit
- \* When set to a negative value, we won't purge no matter how many searches are run
- \* Defaults to 8
- \* Has no effect on Windows, if search\_process\_mode is not "auto", or if max\_searches\_per\_process is set to 0 or 1

idle\_process\_regex\_cache\_hiwater = <int>

- \* When a search process is allowed to run more than one search, it can cache compiled regex artifacts
- \* If that cache grows to larger than this number of entries we'll try purging some older ones
- \* Normally the above "idle\_process\_cache\_\*" settings will take care of keeping the cache a reasonable size. This setting is to prevent the cache from growing extremely large during a single large search
- \* When set to a negative value, we won't purge this cache based on its size
- \* Defaults to 2500
- \* Has no effect on Windows, if search\_process\_mode is not "auto", or if max\_searches\_per\_process is set to 0 or 1

fetch\_remote\_search\_log = [enabled|disabledSavedSearches|disabled]

- \* enabled: all remote search logs will be downloaded barring the oneshot search
- \* disabledSavedSearches: download all remote logs other than saved search logs and oneshot search logs
- \* disabled: irrespective of the search type all remote search log download functionality will be disabled
- \* Defaults to disabledSavedSearches
- \* The previous values:[true|false] are still supported but not recommended for use
- \* The previous value of true maps to the current value of enabled
- \* The previous value of false maps to the current value of disabled

load\_remote\_bundles = <bool>

- \* On a search peer, allow remote (search head) bundles to be loaded in splunkd.
- \* Defaults to false.

use\_dispatchtmp\_dir = <bool>

- \* Whether to use the dispatchtmp directory for temporary search time files (write temporary files to a different directory from a job's dispatch directory).
- \* Temp files would be written to \$SPLUNK\_HOME/var/run/splunk/dispatchtmp/<sid>/
- \* In search head pooling performance can be improved by mounting dispatchtmp to the local file system.
- \* Defaults to true if search head pooling is enabled, false otherwise

check\_splunkd\_period = <number>

- \* Amount of time, in seconds, that determines how frequently the search process (when running a real-time search) checks whether it's parent process (splunkd) is running or not.
- \* Fractional seconds are allowed.
- \* Defaults to 60

allow\_batch\_mode = <bool>

- \* Whether or not to allow the use of batch mode which searches in disk based batches in a time insensitive manner.
- \* In distributed search environments, this setting is used on the search head.
- \* Defaults to true

batch\_search\_max\_index\_values = <int>

- \* When using batch mode this limits the number of event entries read from the index file. These entries are small approximately 72 bytes. However batch

mode is more efficient when it can read more entries at once.

- \* Setting this value to a smaller number can lead to slower search performance.
- \* A balance needs to be struck between more efficient searching in batch mode
- \* and running out of memory on the system with concurrently running searches.
- \* Defaults to 10000000

These settings control the periodicity of retries to search peers in the event of failure. (Connection errors, and others.) The interval exists between failure and first retry, as well as successive retries in the event of further failures.

`batch_retry_min_interval = <int>`

- \* When batch mode attempts to retry the search on a peer that failed wait at least this many seconds
- \* Default to 5

`batch_retry_max_interval = <int>`

- \* When batch mode attempts to retry the search on a peer that failed wait at most this many seconds
- \* Default to 300

`batch_retry_scaling = <double>`

- \* After a retry attempt fails increase the time to wait before trying again by this scaling factor (Value should be > 1.0)
- \* Default 1.5

`batch_wait_after_end = <int>`

- \* Batch mode considers the search ended(finished) when all peers without communication failure have explicitly indicated that they are complete; eg have delivered the complete answer. After the search is at an end, batch mode will continue to retry with lost-connection peers for this many seconds.
- \* Default 900

`batch_search_max_pipeline = <int>`

- \* Controls the number of search pipelines launched at the indexer during batch search.
- \* Default value is set to one pipeline.
- \* Increasing the number of search pipelines should help improve search performance
- \* but there will be an increase in thread and memory usage.

`batch_search_max_results_aggregator_queue_size = <int>`

- \* Controls the size of the search results queue to which all the search pipelines dump the processed search results.
- \* Default size is set to 100MB.
- \* Increasing the size can lead to performance gain where as decreasing can reduce search performance.
- \* Do not set this parameter to zero.

`batch_search_max_serialized_results_queue_size = <int>`

- \* Controls the size of the serialized results queue from which the serialized search results are transmitted.
- \* Default size is set to 100MB.
- \* Increasing the size can lead to performance gain where as decreasing can reduce search performance.
- \* Do not set this parameter to zero.

`write_multifile_results_out = <bool>`

- \* At the end of the search, if results are in multiple files, write out the multiple files to results\_dir directory, under the search results directory.
- \* This will speed up post-processing search, since the results will already be split into appropriate size files.
- \* Default true

`enable_cumulative_quota = <bool>`

- \* Whether to enforce cumulative role based quotas
- \* Default false

`remote_reduce_limit = <unsigned long>`

- \* The number of results processed by a streaming search before we force a reduce
- \* Note: this option applies only if the search is run with --runReduce=true (currently on Hunk does this)
- \* Note: a value of 0 is interpreted as unlimited
- \* Defaults to: 1000000

`max_workers_searchparser = <int>`

- \* The number of worker threads in processing search result when using round



```

    robin policy.
* default 5

max_chunk_queue_size = <int>
* The maximum size of the chunk queue
* default 10000000

max_tolerable_skew = <positive integer>
* Absolute value of the largest timeskew in seconds that we will tolerate
  between the native clock on the searchhead and the native clock on the peer
  (independent of time-zone).
* If this timeskew is exceeded we will log a warning. This estimate is
  approximate and tries to account for network delays.

addpeer_skew_limit = <positive integer>
* Absolute value of the largest time skew in seconds that is allowed when configuring
  a search peer from a search head, independent of time.
* If the difference in time (skew) between the search head and the peer is greater
  than this limit, the search peer will not be added.
* This is only relevant to manually added peers; currently this setting has no effect
  upon index cluster search peers.

unified_search = <bool>
* Turns on/off unified search for hunk archiving, defaults to false if not
  specified.

enable_memory_tracker = <bool>
* If memory tracker is disabled, search won't be terminated even if it exceeds the memory limit.
* Must be set to <true> if you want to enable search_process_memory_usage_threshold or
  search_process_memory_usage_percentage_threshold
* By default false.

search_process_memory_usage_threshold = <double>
* To be active, this setting requires setting: enable_memory_tracker = true
* Signifies the maximum memory in MB the search process can consume in RAM.
* Search processes violating the threshold will be terminated.
* If the value is set to zero, then splunk search processes are allowed
  to grow unbounded in terms of in memory usage.
* The default value is set to 4000MB or 4GB.

search_process_memory_usage_percentage_threshold = <float>
* To be active, this setting requires setting: enable_memory_tracker = true
* Signifies the percentage of the total memory the search process is entitled to consume.
* Any time the search process violates the threshold percentage the process will be brought down.
* If the value is set to zero, then splunk search processes are allowed to grow unbounded
  in terms of percentage memory usage.
* The default value is set to 25%.
* Any number set larger than 100 or less than 0 will be discarded and the default value will be used.

enable_datamodel_meval = <bool>
* Enable concatenation of successively occurring evals into a single
  comma separated eval during generation of datamodel searches.
* default true

do_not_use_summaries = <bool>
* Do not use this setting without working in tandem with Splunk support.
* This setting is a very narrow subset of summary_mode=none. When set to true, this
  setting disables some functionality that is necessary for report acceleration.
  In particular, when set to true, search processes will no longer query the main
  splunkd's /admin/summarization endpoint for report acceleration summary ids.
* In certain narrow use-cases this may improve performance if report acceleration
  (savedsearches.conf:auto_summarize) is not in use by lowering the main splunkd's
  process overhead.
* Defaults to false.

unified_search = <bool>
* Enables the unified search feature.
* Defaults to false.

force_saved_search_dispatch_as_user = <bool>
* Specifies whether to overwrite the 'dispatchAs' value.
* If set to 'true', the 'dispatchAs' value is overwritten by 'user' regardless

```

of the 'user | owner' value in the savedsearches.conf file.

- \* If set to 'false', the value in the savedsearches.conf file is used.
- \* User may want to set this to effectively disable dispatchAs = owner for the entire install, if that more closely aligns with security goals.
- \* Defaults to false.

-- Unsupported [search] settings: --

enable\_status\_cache = <bool>

- \* This is not a user tunable setting. Do not use this setting without working in tandem with Splunk personnel. This setting is not tested at non-default.
- \* This controls whether the status cache is used, which caches information about search jobs (and job artifacts) in memory in main splunkd.
- \* Normally this cacheing is enabled and assists performance. However, when using Search Head Pooling, artifacts in the shared storage location will be changed by other search heads, so this cacheing is disabled.
- \* Explicit requests to jobs endpoints , eg /services/search/jobs/<sid> are always satisfied from disk, regardless of this setting.
- \* Defaults to true; except in Search Head Pooling environments where it defaults to false.

status\_cache\_in\_memory\_ttl = <positive integer>

- \* This setting has no effect unless search head pooling is enabled, AND enable\_status\_cache has been set to true.
- \* This is not a user tunable setting. Do not use this setting without working in tandem with Splunk personnel. This setting is not tested at non-default.
- \* If set, controls the number of milliseconds which a status cache entry may be used before it expires.
- \* Defaults to 60000, or 60 seconds.

## **[realtime]**

[realtime]

# Default options for indexer support of real-time searches

# These can all be overridden for a single search via REST API arguments

local\_connect\_timeout = <int>

- \* Connection timeout for an indexer's search process when connecting to that indexer's splunkd (in seconds)
- \* Defaults to 5

local\_send\_timeout = <int>

- \* Send timeout for an indexer's search process when connecting to that indexer's splunkd (in seconds)
- \* Defaults to 5

local\_receive\_timeout = <int>

- \* Receive timeout for an indexer's search process when connecting to that indexer's splunkd (in seconds)
- \* Defaults to 5

queue\_size = <int>

- \* Size of queue for each real-time search (must be >0).
- \* Defaults to 10000

blocking = [0|1]

- \* Specifies whether the indexer should block if a queue is full.
- \* Defaults to false

max\_blocking\_secs = <int>

- \* Maximum time to block if the queue is full (meaningless if blocking = false)
- \* 0 means no limit
- \* Default to 60

indexfilter = [0|1]

- \* Specifies whether the indexer should prefilter events for efficiency.
- \* Defaults to true (1).

default\_backfill = <bool>

- \* Specifies if windowed real-time searches should backfill events

```

* Defaults to true

enforce_time_order = <bool>
* Specifies if real-time searches should ensure that events are sorted in
  ascending time order (the UI will automatically reverse the order that it
  display events for real-time searches so in effect the latest events will be
  first)
* Defaults to true

disk_usage_update_period = <number>
* Specifies how frequently (in seconds) should the search process estimate the
  artifact disk usage.
* Fractional seconds are allowed.
* Defaults to 10

indexed_realtime_use_by_default = <bool>
* Should we use the indexedRealtime mode by default
* Precedence: SearchHead
* Defaults to false

indexed_realtime_disk_sync_delay = <int>
* After indexing there is a non-deterministic period where the files on disk
  when opened by other programs might not reflect the latest flush to disk,
  particularly when a system is under heavy load.
* This settings controls the number of seconds to wait for disk flushes to
  finish when using indexed/continuous/psuedo realtime search so that we see
  all of the data.
* Precedence: SearchHead overrides Indexers
* Defaults to 60

indexed_realtime_default_span = <int>
* An indexed realtime search is made up of many component historical searches
  that by default will span this many seconds. If a component search is not
  completed in this many seconds the next historical search will span the extra
  seconds. To reduce the overhead of running an indexed realtime search you can
  change this span to delay longer before starting the next component
  historical search.
* Precedence: Indexers
* Defaults to 1

indexed_realtime_maximum_span = <int>
* While running an indexed realtime search, if the component searches regularly
  take longer than indexed_realtime_default_span seconds, then indexed realtime
  search can fall more than indexed_realtime_disk_sync_delay seconds behind
  realtime. Use this setting to set a limit after which we will drop data to
  return back to catch back up to the specified delay from realtime, and only
  search the default span of seconds.
* Precedence: API overrides SearchHead overrides Indexers
* Defaults to 0 (unlimited)

indexed_realtime_cluster_update_interval = <int>
* While running an indexed realtime search, if we are on a cluster we need to
  update the list of allowed primary buckets. This controls the interval that
  we do this. And it must be less than the indexed_realtime_disk_sync_delay. If
  your buckets transition from Brand New to warm in less than this time indexed
  realtime will lose data in a clustered environment.
* Precedence: Indexers
* Default: 30

alerting_period_ms = <int>
* This limits the frequency that we will trigger alerts during a realtime search
* A value of 0 means unlimited and we will trigger an alert for every batch of
  events we read in dense realtime searches with expensive alerts this can
  overwhelm the alerting system.
* Precedence: Searchhead
* Default: 0

```

**[slc]**

```

[slc]
maxclusters = <integer>

```

\* Maximum number of clusters to create.  
\* Defaults to 10000.

### **[findkeywords]**

[findkeywords]  
maxevents = <integer>  
\* Maximum number of events used by the findkeywords command and the Patterns tab.  
\* Defaults to 50000.

### **[sort]**

[sort]  
maxfiles = <integer>  
\* Maximum files to open at once. Multiple passes are made if the number of  
result chunks exceeds this threshold.  
\* Defaults to 64.

### **[stats|sistats]**

[stats|sistats]  
maxmem\_check\_freq = <integer>  
\* How frequently to check to see if we are exceeding the in memory data  
structure size limit as specified by max\_mem\_usage\_mb, in rows  
\* Defaults to 50000 rows  
  
maxresultrows = <integer>  
\* Maximum number of rows allowed in the process memory.  
\* When the search process exceeds max\_mem\_usage\_mb and maxresultrows, data is  
spilled out to the disk  
\* If not specified, defaults to searchresults::maxresultrows (which is by default 50000).  
  
maxvalues = <integer>  
\* Maximum number of values for any field to keep track of.  
\* Defaults to 0 (unlimited).  
  
maxvaluesize = <integer>  
\* Maximum length of a single value to consider.  
\* Defaults to 0 (unlimited).  
  
# rdigest is a data structure used to compute approximate order statistics  
# (such as median and percentiles) using sublinear space.  
  
rdigest\_k = <integer>  
\* rdigest compression factor  
\* Lower values mean more compression  
\* After compression, number of nodes guaranteed to be greater than or equal to  
11 times k.  
\* Defaults to 100, must be greater than or equal to 2  
  
rdigest\_maxnodes = <integer>  
\* Maximum rdigest nodes before automatic compression is triggered.  
\* Defaults to 1, meaning automatically configure based on k value  
  
max\_stream\_window = <integer>  
\* For the streamstats command, the maximum allow window size  
\* Defaults to 10000.  
  
max\_valuemap\_bytes = <integer>  
\* For sistats command, the maximum encoded length of the valuemap, per result  
written out  
\* If limit is exceeded, extra result rows are written out as needed. (0 = no  
limit per row)  
\* Defaults to 100000.  
  
perc\_method = nearest-rank|interpolated  
\* Which method to use for computing percentiles (and medians=50 percentile).  
\* nearest-rank picks the number with 0-based rank R =

```

    floor((percentile/100)*count)
* interpolated means given  $F = (\text{percentile}/100) * (\text{count}-1)$ ,
  pick ranks  $R1 = \text{floor}(F)$  and  $R2 = \text{ceiling}(F)$ .
  Answer =  $(R2 * (F - R1)) + (R1 * (1 - (F - R1)))$ 
* See wikipedia percentile entries on nearest rank and "alternative methods"
* Defaults to interpolated

approx_dc_threshold = <integer>
* When using approximate distinct count (i.e. estdc(<field>) in
  stats/chart/timechart), do not use approximated results if the actual number
  of distinct values is less than this number
* Defaults to 1000

dc_digest_bits = <integer>
*  $2^{\text{<integer>}}$  bytes will be size of digest used for approximating distinct count.
* Defaults to 10 (equivalent to 1KB)
* Must be  $\geq 8$  (128B) and  $\leq 16$  (64KB)

natural_sort_output = <bool>
* Do a natural sort on the output of stats if output size is  $\leq \text{maxresultrows}$ 
* Natural sort means that we sort numbers numerically and non-numbers
  lexicographically
* Defaults to true

list_maxsize = <int>
* Maximum number of list items to emit when using the list() function
  stats/sistats
* Defaults to 100

sparkline_maxsize = <int>
* Maximum number of elements to emit for a sparkline
* Defaults to value of the list_maxsize setting

sparkline_time_steps = <time-step-string>
* Specify a set of time steps in order of decreasing granularity. Use an integer and
  one of the following time units to indicate each step.
** s = seconds
** m = minutes
** h = hours
** d = days
** month
* Defaults to: 1s,5s,10s,30s,1m,5m,10m,30m,1h,1d,1month
* A time step from this list is selected based on the <sparkline_maxsize> setting.
* The lowest <sparkline_time_steps> value that does not exceed the maximum number
  of bins is used.
* Example:
** If you have the following configurations:
** <sparkline_time_steps> = 1s,5s,10s,30s,1m,5m,10m,30m,1h,1d,1month
** <sparkline_maxsize> = 100
** The timespan for 7 days of data is 604,800 seconds.
** Span =  $604,800 / \text{<sparkline_maxsize>}$ .
** If sparkline_maxsize = 100, then span =  $(604,800 / 100) = 6,048 \text{ sec} \approx 1.68 \text{ hours}$ .
** The "1d" time step is used because it is the lowest value that does not exceed
  the maximum number of bins.

default_partitions = <int>
* Number of partitions to split incoming data into for parallel/multithreaded reduce
* Defaults to 1

partitions_limit = <int>
* Maximum number of partitions to split into that can be specified via the
  'partitions' option.
* When exceeded, the number of partitions is reduced to this limit.
* Defaults to 100

```

## **[thruput]**

```

[thruput]
maxKBps = <integer>
* If specified and not zero, this limits the speed through the thruput processor
  in the ingestion pipeline to the specified rate in kilobytes per second.

```

- \* To control the CPU load while indexing, use this to throttle the number of events this indexer processes to the rate (in KBps) you specify.
- \* Note that this limit will be applied per ingestion pipeline. For more information about multiple ingestion pipelines see `parallelIngestionPipelines` in the `server.conf.spec` file.
- \* With N parallel ingestion pipelines the throughput limit across all of the ingestion pipelines will be  $N * \text{maxKBps}$ .

## **[journal\_compression]**

```
[journal_compression]
threads = <integer>
```

- \* Specifies the maximum number of indexer threads which will be work on compressing hot bucket journal data.
- \* Defaults to the number of CPU threads of the host machine
- \* This setting does not typically need to be modified.

## **[top]**

```
[top]
maxresultrows = <integer>
```

- \* Maximum number of result rows to create.
- \* If not specified, defaults to `searchresults::maxresultrows` (usually 50000).

```
maxvalues = <integer>
```

- \* Maximum number of distinct field vector values to keep track of.
- \* Defaults to 100000.

```
maxvaluesize = <integer>
```

- \* Maximum length of a single value to consider.
- \* Defaults to 1000.

## **[summarize]**

```
[summarize]
hot_bucket_min_new_events = <integer>
```

- \* The minimum number of new events that need to be added to the hot bucket (since last summarization) before a new summarization can take place. To disable hot bucket summarization set this value to a \* large positive number.
- \* Defaults to 100000

```
max_hot_bucket_summarization_idle_time = <unsigned int>
```

- \* Maximum amount of time, in seconds, a hot bucket can be idle after which we summarize all the events even if there are not enough events (determined by `hot_bucket_min_new_events`)
- \* Defaults to 900 seconds (or 15 minutes)

```
sleep_seconds = <integer>
```

- \* The amount of time to sleep between polling of summarization complete status.
- \* Default to 5

```
stale_lock_seconds = <integer>
```

- \* The amount of time to have elapse since the mod time of a `.lock` file before summarization considers \* that lock file stale and removes it
- \* Default to 600

```
max_summary_ratio = <float>
```

- \* A number in the [0-1] range that indicates the maximum ratio of summary data / bucket size at which point the summarization of that bucket, for the particular search, will be disabled. Use 0 to disable.
- \* Defaults to 0

```
max_summary_size = <int>
```

- \* Size of summary, in bytes, at which point we'll start applying the `max_summary_ratio`. Use 0 to disable.
- \* Defaults to 0

```
max_time = <int>
```

- \* The maximum amount of time, seconds, that a summary search process is allowed

to run. Use 0 to disable.

- \* Defaults to 0

indextime\_lag = <unsigned int>

- \* The amount of lag time to give indexing to ensure that it has synced any received events to disk. Effectively, the data that has been received in the past indextime\_lag will NOT be summarized.
- \* Do not change this value unless directed by Splunk support.
- \* Defaults to 90

max\_replicated\_hot\_bucket\_idle\_time = <unsigned int>

- \* Maximum amount of time, in seconds, a replicated hot bucket can be idle after which we won't apply indextime\_lag.
- \* This applies to only idle replicated hot buckets. As soon as new events start flowing in we will revert to the default behavior of applying indextime\_lag
- \* Defaults to 3600 seconds

## **[transactions]**

[transactions]

maxopentxn = <integer>

- \* Specifies the maximum number of not yet closed transactions to keep in the open pool before starting to evict transactions.
- \* Defaults to 5000.

maxopenevents = <integer>

- \* Specifies the maximum number of events (which are) part of open transactions before transaction eviction starts happening, using LRU policy.
- \* Defaults to 100000.

## **[inputproc]**

[inputproc]

max\_fd = <integer>

- \* Maximum number of file descriptors that a ingestion pipeline in Splunk will keep open, to capture any trailing data from files that are written to very slowly.
- \* Note that this limit will be applied per ingestion pipeline. For more information about multiple ingestion pipelines see parallelIngestionPipelines in the server.conf.spec file.
- \* With N parallel ingestion pipelines the maximum number of file descriptors that can be open across all of the ingestion pipelines will be N \* max\_fd.
- \* Defaults to 100.

monitornohandle\_max\_heap\_mb = <integer>

- \* Controls the maximum memory used by the Windows-specific modular input MonitorNoHandle.
- \* The memory of this input grows in size when the data being produced by applications writing to monitored files comes in faster than the Splunk system can accept it.
- \* When set to 0, the heap size (memory allocated in the modular input) can grow without limit.
- \* If this size is limited, and the limit is encountered, the input will drop some data to stay within the limit.
- \* Defaults to 0.

time\_before\_close = <integer>

- \* MOVED. This setting is now configured per-input in inputs.conf.
- \* Specifying this setting in limits.conf is DEPRECATED, but for now will override the setting for all monitor inputs.

tailing\_proc\_speed = <integer>

- \* REMOVED. This setting is no longer used.

file\_tracking\_db\_threshold\_mb = <integer>

- \* This setting controls the trigger point at which the file tracking db (also commonly known as the "fishbucket" or btree) rolls over. A new database is created in its place. Writes are targeted at new db. Reads are first targeted at new db, and we fall back to old db for read failures. Any reads served from old db successfully will be written back into new db.

\* MIGRATION NOTE: if this setting doesn't exist, the initialization code in splunkd triggers an automatic migration step that reads in the current value for "maxDataSize" under the "\_thefishbucket" stanza in indexes.conf and writes this value into etc/system/local/limits.conf.

learned\_sourcetypes\_limit = <0 or positive integer>

- \* Limits the number of entries added to the learned app for performance reasons.
- \* If nonzero, limits two properties of data added to the learned app by the file classifier. (Code specific to monitor:: stanzas that auto-determines sourcetypes from content.)
  - \* The number of sourcetypes added to the learned app's props.conf file will be limited to approximately this number.
  - \* The number of file-content fingerprints added to the learned app's sourcetypes.conf file will be limited to approximately this number.
- \* The tracking for uncompressed and compressed files is done separately, so in some cases this value may be exceeded.
- \* This limit is not the recommended solution for auto-identifying sourcetypes. The usual best practices are to set sourcetypes in input stanzas, or alternatively to apply them based on filename pattern in props.conf [source::<pattern>] stanzas.
- \* Defaults to 1000.

## **[scheduler]**

[scheduler]

saved\_searches\_disabled = <bool>

- \* Whether saved search jobs are disabled by the scheduler.
- \* Defaults to false.

max\_searches\_perc = <integer>

- \* The maximum number of searches the scheduler can run, as a percentage of the maximum number of concurrent searches, see [search] max\_searches\_per\_cpu for how to set the system wide maximum number of searches.
- \* Defaults to 50.

max\_searches\_perc.<n> = <integer>

max\_searches\_perc.<n>.when = <cron string>

- \* The same as max\_searches\_perc but the value is applied only when the cron string matches the current time. This allows max\_searches\_perc to have different values at different times of day, week, month, etc.
- \* There may be any number of non-negative <n> that progress from least specific to most specific with increasing <n>.
- \* The scheduler looks in reverse-<n> order looking for the first match.
- \* If either these settings aren't provided at all or no "when" matches the current time, the value falls back to the non-<n> value of max\_searches\_perc.

auto\_summary\_perc = <integer>

- \* The maximum number of concurrent searches to be allocated for auto summarization, as a percentage of the concurrent searches that the scheduler can run.
- \* Auto summary searches include:
  - \* Searches which generate the data for the Report Acceleration feature.
  - \* Searches which generate the data for Data Model acceleration.
- \* Note: user scheduled searches take precedence over auto summary searches.
- \* Defaults to 50.

auto\_summary\_perc.<n> = <integer>

auto\_summary\_perc.<n>.when = <cron string>

- \* The same as auto\_summary\_perc but the value is applied only when the cron string matches the current time. This allows auto\_summary\_perc to have different values at different times of day, week, month, etc.
- \* There may be any number of non-negative <n> that progress from least specific to most specific with increasing <n>.
- \* The scheduler looks in reverse-<n> order looking for the first match.
- \* If either these settings aren't provided at all or no "when" matches the current time, the value falls back to the non-<n> value of auto\_summary\_perc.

priority\_runtime\_factor = <double>

- \* The amount to scale the priority runtime adjustment by.
- \* Every search's priority is made higher (worse) by its typical running time.



Since many searches run in fractions of a second and the priority is integral, adjusting by a raw runtime wouldn't change the result; therefore, it's scaled by this value.

- \* Defaults to 10.

priority\_skipped\_factor = <double>

- \* The amount to scale the skipped adjustment by.
- \* A potential issue with the priority\_runtime\_factor is that now longer-running searches may get starved. To balance this out, make a search's priority lower (better) the more times it's been skipped. Eventually, this adjustment will outweigh any worse priority due to a long runtime. This value controls how quickly this happens.
- \* Defaults to 1.

search\_history\_max\_runtimes = <unsigned int>

- \* The number of runtimes kept for each search.
- \* Used to calculate historical typical runtime during search prioritization.
- \* Defaults to 10.

search\_history\_load\_timeout = <duration-specifier>

- \* The maximum amount of time to defer running continuous scheduled searches while waiting for the KV Store to come up in order to load historical data. This is used to prevent gaps in continuous scheduled searches when splunkd was down.
- \* Use [<int>]<unit> to specify a duration; a missing <int> defaults to 1.
- \* Relevant units are: s, sec, second, secs, seconds, m, min, minute, mins, minutes.
- \* For example: "60s" = 60 seconds, "5m" = 5 minutes.
- \* Defaults to 2m.

max\_continuous\_scheduled\_search\_lookback = <duration-specifier>

- \* The maximum amount of time to run missed continuous scheduled searches for once Splunk comes back up in the event it was down.
- \* Use [<int>]<unit> to specify a duration; a missing <int> defaults to 1.
- \* Relevant units are: m, min, minute, mins, minutes, h, hr, hour, hrs, hours, d, day, days, w, week, weeks, mon, month, months.
- \* For example: "5m" = 5 minutes, "1h" = 1 hour.
- \* A value of 0 means no lookback.
- \* Defaults to 24 hours.

introspection\_lookback = <duration-specifier>

- \* The amount of time to "look back" when reporting introspection statistics.
- \* For example: what is the number of dispatched searches in the last 60 minutes?
- \* Use [<int>]<unit> to specify a duration; a missing <int> defaults to 1.
- \* Relevant units are: m, min, minute, mins, minutes, h, hr, hour, hrs, hours, d, day, days, w, week, weeks.
- \* For example: "5m" = 5 minutes, "1h" = 1 hour.
- \* Defaults to 1 hour.

max\_action\_results = <integer>

- \* The maximum number of results to load when triggering an alert action.
- \* Defaults to 50000

action\_execution\_threads = <integer>

- \* Number of threads to use to execute alert actions, change this number if your alert actions take a long time to execute.
- \* This number is capped at 10.
- \* Defaults to 2

actions\_queue\_size = <integer>

- \* The number of alert notifications to queue before the scheduler starts blocking, set to 0 for infinite size.
- \* Defaults to 100

actions\_queue\_timeout = <integer>

- \* The maximum amount of time, in seconds to block when the action queue size is full.
- \* Defaults to 30

alerts\_max\_count = <integer>

- \* Maximum number of unexpired alerts information to keep for the alerts manager, when this number is reached Splunk will start discarding the oldest

```

    alerts.
* Defaults to 50000

alerts_max_history = <integer>[s|m|h|d]
* Maximum time to search in the past for previously triggered alerts.
* splunkd uses this property to populate the Activity -> Triggered Alerts page at startup.
* Defaults to 7 days.
* Values greater than the default may cause slowdown.

alerts_scoping = host|splunk_server|all
* Determines the scoping to use on the search to populate the triggered alerts
  page. Choosing splunk_server will result in the search query
  using splunk_server=local, host will result in the search query using
  host=<search-head-host-name>, and all will have no scoping added to the
  search query.
* Defaults to splunk_server.

alerts_expire_period = <integer>
* The amount of time between expired alert removal
* This period controls how frequently the alerts list is scanned, the only
  benefit from reducing this is better resolution in the number of alerts fired
  at the savedsearch level.
* Change not recommended.
* Defaults to 120.

persistence_period = <integer>
* The period (in seconds) between scheduler state persistence to disk. The
  scheduler currently persists the suppression and fired-unexpired alerts to
  disk.
* This is relevant only in search head pooling mode.
* Defaults to 30.

max_lock_files = <int>
* The number of most recent lock files to keep around.
* This setting only applies in search head pooling.

max_lock_file_ttl = <int>
* Time (in seconds) that must pass before reaping a stale lock file.
* Only applies in search head pooling.

max_per_result_alerts = <int>
* Maximum number of alerts to trigger for each saved search instance (or
  real-time results preview for RT alerts)
* Only applies in non-digest mode alerting. Use 0 to disable this limit
* Defaults to 500

max_per_result_alerts_time = <int>
* Maximum number of time to spend triggering alerts for each saved search
  instance (or real-time results preview for RT alerts)
* Only applies in non-digest mode alerting. Use 0 to disable this limit.
* Defaults to 300

scheduled_view_timeout = <int>[s|m|h|d]
* The maximum amount of time that a scheduled view (pdf delivery) would be
  allowed to render
* Defaults to 60m

concurrency_message_throttle_time = <int>[s|m|h|d]
* Amount of time controlling throttling between messages warning about scheduler concurrency limits
* Defaults to 10m

shp_dispatch_to_slave = <bool>
* By default the scheduler should distribute jobs throughout the pool.
* Defaults to true

shc_role_quota_enforcement = <bool>
* When this is enabled, the following limits are enforced by the captain for scheduled searches:
  - User role quotas are enforced globally.
    A given role can have (n *number_of_peers) searches running cluster-wide,
    where n is the quota for that role as defined by srchJobsQuota and
    rtSrchJobsQuota on the captain
  - Maximum number of concurrent searches is enforced globally.

```

This is  $(n * \text{number\_of\_peers})$  where  $n$  is the max concurrent searches on the captain (see `max_searches_per_cpu` for a description of how this is computed).

Concurrent searches include both scheduled searches and ad hoc searches.

- \* Scheduled searches will therefore not have an enforcement of either of the above on a per-member basis.
- \* Note that this doesn't control the enforcement of the scheduler quota. For a search head cluster, that is defined as  $(\text{max\_searches\_perc} * \text{number\_of\_peers})$  and is always enforced globally on the captain.
- \* Quota information is conveyed from the members to the captain. Network delays can cause the quota calculation on the captain to vary from the actual values in the members and may cause search limit warnings. This should clear up as the information is synced.
- \* Defaults to false.

`shc_local_quota_check = <bool>`

- \* Enabling this enforces user role quota and maximum number of concurrent searches on a per-member basis.
- \* Cluster-wide scheduler quota is still enforced globally on the captain.
- \* See `shc_role_quota_enforcement` for more details.
- \* Disabling this requires `shc_role_quota_enforcement=true`. Otherwise, all quota checks will be skipped.
- \* Note that disabling this will also disable disk quota checks.
- \* Defaults to true.

## **[auto\_summarizer]**

`[auto_summarizer]`

`cache_timeout = <integer>`

- \* The amount of time, in seconds, to cache auto summary details and search hash codes
- \* Defaults to 600 - 10 minutes

`search_2_hash_cache_timeout = <integer>`

- \* The amount of time, in seconds, to cache search hash codes
- \* Defaults to the value of `cache_timeout` i.e. 600 - 10 minutes

`maintenance_period = <integer>`

- \* The period of time, in seconds, that the auto summarization maintenance happens
- \* Defaults to 1800 (30 minutes)

`allow_event_summarization = <bool>`

- \* Whether auto summarization of searches whose remote part returns events rather than results will be allowed.
- \* Defaults to false

`max_verify_buckets = <int>`

- \* When verifying buckets, stop after verifying this many buckets if no failures have been found
- \* 0 means never
- \* Defaults to 100

`max_verify_ratio = <number>`

- \* Maximum fraction of data in each bucket to verify
- \* Defaults to 0.1 (10%)

`max_verify_bucket_time = <int>`

- \* Maximum time to spend verifying each bucket, in seconds
- \* Defaults to 15 (seconds)

`verify_delete = <bool>`

- \* Should summaries that fail verification be automatically deleted?
- \* Defaults to false

`max_verify_total_time = <int>`

- \* Maximum total time in seconds to spend doing verification, regardless if any buckets have failed or not
- \* Defaults to 0 (no limit)

`max_run_stats = <int>`

- \* Maximum number of summarization run statistics to keep track and expose via

```

REST.
* Defaults to 48

return_actions_with_normalized_ids = [yes|no|fromcontext]
* Report acceleration summaries are stored under a signature/hash which can be
  regular or normalized.
  * Normalization improves the re-use of pre-built summaries but is not
    supported before 5.0. This config will determine the default value of how
    normalization works (regular/normalized)
  * Default value is "fromcontext", which would mean the end points and
    summaries would be operating based on context.
* normalization strategy can also be changed via admin/summarization REST calls
  with the "use_normalization" parameter which can take the values
  "yes"/"no"/"fromcontext"

normalized_summaries = <bool>
* Turn on/off normalization of report acceleration summaries.
* Default = false and will become true in 6.0

detailed_dashboard = <bool>
* Turn on/off the display of both normalized and regular summaries in the
  Report Acceleration summary dashboard and details.
* Default = false

shc_accurate_access_counts = <bool>
* Only relevant if you are using search head clustering
* Turn on/off to make acceleration summary access counts accurate on the
  captain.
* by centralizing the access requests on the captain.
* Default = false

```

## **[show\_source]**

```

[show_source]
max_count = <integer>
* Maximum number of events accessible by show_source.
* The show source command will fail when more than this many events are in the
  same second as the requested event.
* Defaults to 10000

max_timebefore = <timespan>
* Maximum time before requested event to show.
* Defaults to '1day' (86400 seconds)

max_timeafter = <timespan>
* Maximum time after requested event to show.
* Defaults to '1day' (86400 seconds)

distributed = <bool>
* Controls whether we will do a distributed search for show source to get
  events from all servers and indexes
* Turning this off results in better performance for show source, but events
  will only come from the initial server and index
* NOTE: event signing and verification is not supported in distributed mode
* Defaults to true

distributed_search_limit = <unsigned int>
* Sets a limit on the maximum events we will request when doing the search for
  distributed show source
* As this is used for a larger search than the initial non-distributed show
  source, it is larger than max_count
* Splunk will rarely return anywhere near this amount of results, as we will
  prune the excess results
* The point is to ensure the distributed search captures the target event in an
  environment with many events
* Defaults to 30000

```

## **[typeahead]**

```

[typeahead]

```

```

maxcount = <integer>
* Maximum number of typeahead results to find.
* Defaults to 1000

use_cache = [0|1]
* Specifies whether the typeahead cache will be used if use_cache is not
  specified in the command line or endpoint.
* Defaults to true.

fetch_multiplier = <integer>
* A multiplying factor that determines the number of terms to fetch from the
  index, fetch = fetch_multiplier x count.
* Defaults to 50

cache_ttl_sec = <integer>
* How long the typeahead cached results are valid, in seconds.
* Defaults to 300.

min_prefix_length = <integer>
* The minimum string prefix after which to provide typeahead.
* Defaults to 1.

max_concurrent_per_user = <integer>
* The maximum number of concurrent typeahead searches per user. Once this
  maximum is reached only cached typeahead results might be available
* Defaults to 3.

```

## **[typer]**

```

[typer]
maxlen = <int>
* In eventtyping, pay attention to first <int> characters of any attribute
  (such as _raw), including individual tokens. Can be overridden by supplying
  the typer operator with the argument maxlen (for example,
  "|typer maxlen=300").
* Defaults to 10000.

```

## **[authtokens]**

```

[authtokens]
expiration_time = <integer>
* Expiration time of auth tokens in seconds.
* Defaults to 3600

```

## **[sample]**

```

[sample]
maxsamples = <integer>
* Defaults to 10000

maxtotalsamples = <integer>
* Defaults to 100000

```

## **[metadata]**

```

[metadata]
maxresultrows = <integer>
* The maximum number of results in a single chunk fetched by the metadata
  command
* A smaller value will require less memory on the search head in setups with
  large number of peers and many metadata results, though, setting this too
  small will decrease the search performance
* Default is 10000
* Do not change unless instructed to do so by Splunk Support

maxcount = <integer>
* The total number of metadata search results returned by the search head;

```

after the maxcount is reached, any additional metadata results received from the search peers will be ignored (not returned)

- \* A larger number incurs additional memory usage on the search head
- \* Default is 100000

## **[set]**

[set]  
maxresultrows = <integer>

- \* The maximum number of results the set command will use from each resultset to compute the required set operation

## **[input\_channels]**

[input\_channels]  
max\_inactive = <integer>

- \* Internal setting, do not change unless instructed to do so by Splunk Support

lowater\_inactive = <integer>

- \* Internal setting, do not change unless instructed to do so by Splunk Support

inactive\_eligibility\_age\_seconds = <integer>

- \* Internal setting, do not change unless instructed to do so by Splunk Support

## **[ldap]**

[ldap]  
max\_users\_to\_precache = <unsigned integer>

- \* The maximum number of users we will attempt to pre-cache from LDAP after reloading auth
- \* Set this to 0 to turn off pre-caching

allow\_multiple\_matching\_users = <bool>

- \* This controls whether we allow login when we find multiple entries with the same value for the username attribute
- \* When multiple entries are found, we choose the first user DN lexicographically
- \* Setting this to false is more secure as it does not allow any ambiguous login, but users with duplicate entries will not be able to login.
- \* Defaults to true

## **[spath]**

[spath]  
extraction\_cutoff = <integer>

- \* For extract-all spath extraction mode, only apply extraction to the first <integer> number of bytes
- \* Defaults to 5000

extract\_all = <boolean>

- \* Controls whether we respect automatic field extraction when spath is invoked manually.
- \* If true, we extract all fields regardless of settings. If false, we only extract fields used by later search commands.

## **[reversedns]**

[reversedns]  
rdnsMaxDutyCycle = <integer>

- \* Generate diagnostic WARN in splunkd.log if reverse dns lookups are taking more than this percent of time
- \* Range 0-100
- \* Defaults to 10

## **[viewstates]**

```
[viewstates]
enable_reaper = <boolean>
* Controls whether the viewstate reaper runs
* Defaults to true

reaper_freq = <integer>
* Controls how often the viewstate reaper runs
* Defaults to 86400 (1 day)

reaper_soft_warn_level = <integer>
* Controls what the reaper considers an acceptable number of viewstates
* Defaults to 1000

ttl = <integer>
* Controls the age at which a viewstate is considered eligible for reaping
* Defaults to 86400 (1 day)
```

## **[geostats]**

```
[geostats]
maxzoomlevel = <integer>
* Controls the number of zoom levels that geostats will cluster events on

zl_0_gridcell_latspan = <float>
* Controls what is the grid spacing in terms of latitude degrees at the lowest zoom
  level, which is zoom-level 0.
* Grid-spacing at other zoom levels are auto created from this value by reducing by a
  factor of 2 at each zoom-level.

zl_0_gridcell_longspan = <float>
* Controls what is the grid spacing in terms of longitude degrees at the lowest zoom
  level, which is zoom-level 0
* Grid-spacing at other zoom levels are auto created from this value by reducing by a
  factor of 2 at each zoom-level.

filterstrategy = <integer>
* Controls the selection strategy on the geoviz map. Allowed values are 1 and 2.
```

## **[iplocation]**

```
[iplocation]
db_path = <path>
* Absolute path to GeoIP database in MMDB format
* If not set, defaults to database included with splunk
```

## **[tscollect]**

```
[tscollect]
squashcase = <boolean>
* The default value of the 'squashcase' argument if not specified by the command
* Defaults to false

keepresults = <boolean>
* The default value of the 'keepresults' argument if not specified by the command
* Defaults to false

optimize_max_size_mb = <unsigned int>
* The maximum size in megabytes of files to create with optimize
* Specify 0 for no limit (may create very large tsidx files)
* Defaults to 1024
```

## **[tstats]**

```
[tstats]
apply_search_filter = <boolean>
* Controls whether we apply role-based search filters when users run tstats on
  normal index data
```

\* Note: we never apply search filters to data collected with tscollect or datamodel acceleration  
\* Defaults to true

summariesonly = <boolean>

\* The default value of 'summariesonly' arg if not specified by the command  
\* When running tstats on an accelerated datamodel, summariesonly=false implies a mixed mode where we will fall back to search for missing TSIDX data  
\* summariesonly=true overrides this mixed mode to only generate results from TSIDX data, which may be incomplete  
\* Defaults to false

allow\_old\_summaries = <boolean>

\* The default value of 'allow\_old\_summaries' arg if not specified by the command  
\* When running tstats on an accelerated datamodel, allow\_old\_summaries=false ensures we check that the datamodel search in each bucket's summary metadata is considered up to date with the current datamodel search. Only summaries that are considered up to date will be used to deliver results.  
\* The allow\_old\_summaries=true attribute overrides this behavior and will deliver results even from bucket summaries that are considered out of date with the current datamodel.  
\* Defaults to false

chunk\_size = <unsigned int>

\* ADVANCED: The default value of 'chunk\_size' arg if not specified by the command  
\* This argument controls how many events are retrieved at a time within a single TSIDX file when answering queries  
\* Consider lowering this value if tstats queries are using too much memory (cannot be set lower than 10000)  
\* Larger values will tend to cause more memory to be used (per search) and might have performance benefits.  
\* Smaller values will tend to reduce performance and might reduce memory used (per search).  
\* Altering this value without careful measurement is not advised.  
\* Defaults to 10000000

warn\_on\_missing\_summaries = <boolean>

\* ADVANCED: Only meant for debugging summariesonly=true searches on accelerated datamodels.  
\* When true, search will issue a warning for a tstats summariesonly=true search for the following scenarios:  
    a) If there is a non-hot bucket that has no corresponding datamodel acceleration summary whatsoever.  
    b) If the bucket's summary does not match with the current datamodel acceleration search.  
\* Defaults to false

## **[pdf]**

[pdf]

max\_rows\_per\_table = <unsigned int>

\* The maximum number of rows that will be rendered for a table within integrated PDF rendering  
\* Defaults to 1000

render\_endpoint\_timeout = <unsigned int>

\* The number of seconds after which the pdfgen render endpoint will timeout if it has not yet finished rendering the PDF output  
\* Defaults to 3600

## **[kvstore]**

[kvstore]

max\_accelerations\_per\_collection = <unsigned int>

\* The maximum number of accelerations that can be assigned to a single collection  
\* Valid values range from 0 to 50  
\* Defaults to 10

max\_fields\_per\_acceleration = <unsigned int>

\* The maximum number of fields that can be part of a compound acceleration (i.e. an acceleration with multiple keys)



- \* Valid values range from 0 to 50
- \* Defaults to 10

max\_rows\_per\_query = <unsigned int>

- \* The maximum number of rows that will be returned for a single query to a collection.
- \* If the query returns more rows than the specified value, then returned result set will contain the number of rows specified in this value.
- \* Defaults to 50000

max\_queries\_per\_batch = <unsigned int>

- \* The maximum number of queries that can be run in a single batch
- \* Defaults to 1000

max\_size\_per\_result\_mb = <unsigned int>

- \* The maximum size of the result that will be returned for a single query to a collection in MB.
- \* Defaults to 50 MB

max\_size\_per\_batch\_save\_mb = <unsigned int>

- \* The maximum size of a batch save query in MB
- \* Defaults to 50 MB

max\_documents\_per\_batch\_save = <unsigned int>

- \* The maximum number of documents that can be saved in a single batch
- \* Defaults to 1000

max\_size\_per\_batch\_result\_mb = <unsigned int>

- \* The maximum size of the result set from a set of batched queries
- \* Defaults to 100 MB

max\_rows\_in\_memory\_per\_dump = <unsigned int>

- \* The maximum number of rows in memory before flushing it to the CSV projection of KVStore collection.
- \* Defaults to 200

max\_threads\_per\_outputlookup = <unsigned int>

- \* The maximum number of threads to use during outputlookup commands on KVStore
- \* If the value is 0 the thread count will be determined by CPU count
- \* Defaults to 1

## **[http\_input]**

[http\_input]

max\_number\_of\_tokens = <unsigned int>

- \* The maximum number of tokens reported by logging input metrics.
- \* Default to 10000.

metrics\_report\_interval = 60

- \* The interval (in seconds) of logging input metrics report.
- \* Default to 60 (one minute).

max\_content\_length = 1000000

- \* The maximum length of http request content accepted by HTTP Input server.
- \* Default to 1000000 (~ 1MB).

max\_number\_of\_ack\_channel = 1000000

- \* The maximum number of ACK channels accepted by HTTP Event Collector server.
- \* Default to 1000000 (~ 1M).

max\_number\_of\_acked\_requests\_pending\_query = 10000000

- \* The maximum number of ACKed requests pending query on HTTP Event Collector server.
- \* Default to 10000000 (~ 10M).

max\_number\_of\_acked\_requests\_pending\_query\_per\_ack\_channel = 1000000

- \* The maximum number of ACKed requested pending query per ACK channel on HTTP Event Collector server..
- \* Default to 1000000 (~ 1M).

## **[slow\_peer\_disconnect]**

```
[slow_peer_disconnect]
* Settings for the heuristic that will detect and disconnect slow peers towards
  the end of a search that has returned a large volume of data

disabled = <boolean>
* is this feature enabled.
* Defaults to true

batch_search_activation_fraction = <double>
* The fraction of peers that must have completed before we start disconnecting
* This is only applicable to batch search because the slow peers will not hold
  back the fast peers.
* Defaults to 0.9

packets_per_data_point = <unsigned int>
* Rate statistics will be sampled once every packets_per_data_point packets.
* Defaults to 500

sensitivity = <double>
* Sensitivity of the heuristic to newer values. For larger values of sensitivity
  the heuristic will give more weight to newer statistic.
* Defaults to 0.3

grace_period_before_disconnect = <double>
* If the heuristic consistently claims that the peer is slow for at least
  <grace_period_before_disconnect>*life_time_of_collector seconds then only
  will we disconnect the peer
* Defaults to 0.1

threshold_data_volume = <unsigned int>
* The volume of uncompressed data that must have accumulated in KB from
  a peer before we consider them in the heuristic.
* Defaults to 1024

threshold_connection_life_time = <unsigned int>
* All peers will be given an initial grace period of at least these many
  seconds before we consider them in the heuristic.
* Defaults to 60

bound_on_disconnect_threshold_as_fraction_of_mean = <double>
* The maximum value of the threshold data rate we will use to determine if
  a peer is slow. The actual threshold will be computed dynamically at search
  time but will never exceed (100*maximum_threshold_as_fraction_of_mean)% on
  either side of the mean.
* Defaults to 0.2
```

## **[geomfilter]**

```
[geomfilter]
enable_generalization = <boolean>
* Whether or not generalization is applied to polygon boundaries to reduce
  point count for rendering
* Defaults to true

enable_clipping = <boolean>
* Whether or not polygons are clipped to the viewport provided by the render client
* Defaults to true
```

## **[system\_checks]**

```
[system_checks]
insufficient_search_capabilities = enabled | disabled
* Enables/disables automatic daily logging of scheduled searches by users who
  have insufficient capabilities to run them as configured.
* Such searches are those that:
  + Have schedule_priority set to a value other than "default" but the owner
    does not have the edit_search_schedule_priority capability.
  + Have schedule_window set to a value other than "auto" but the owner does
    not have the edit_search_schedule_window capability.
```

- \* This check and any resulting logging occur on system startup and every 24 hours thereafter.
- \* Defaults to enabled.

```
orphan_searches = enabled|disabled
```

- \* Enables/disables automatic UI message notifications to admins for scheduled saved searches with invalid owners.
- \* Scheduled saved searches with invalid owners are considered "orphaned". They cannot be run because Splunk cannot determine the roles to use for the search context.
- \* Typically, this situation occurs when a user creates scheduled searches then departs the organization or company, causing their account to be deactivated.
- \* Currently this check and any resulting notifications occur on system startup and every 24 hours thereafter.
- \* Defaults to enabled.

```
installed_files_integrity = enabled | log_only | disabled
```

- \* Enables/disables automatic verification on every startup that all the files that were installed with the running Splunk version are still the files that should be present.
- \* Effectively this finds cases where files were removed or changed that should not be removed or changed, whether by accident or intent.
- \* The source of truth for the files that should be present is the manifest file in the \$SPLUNK\_HOME directory that comes with the release, so if this file is removed or altered, the check cannot work correctly.
- \* Reading of all the files provided with the install has some I/O cost, though it is paid out over many seconds and should not be severe.
- \* When "enabled", detected problems will cause a message to be posted to the bulletin board (system UI status message).
- \* When "enabled" or "log\_only", detected problems will cause details to be written out to splunkd.log
- \* When "disabled", no check will be attempted or reported.
- \* Defaults to enabled.

```
#####
# Global Optimization Settings
#####
```

### **[search\_optimization]**

```
[search_optimization]
enabled = <bool>
* Enables search optimizations
* Defaults to true
```

```
#####
# Individual optimizers
#####
```

```
#Configuration options for predicate_push optimizations
```

### **[search\_optimization::predicate\_push]**

```
[search_optimization::predicate_push]
enabled = <bool>
* Enables predicate push optimization
* Defaults to true
```

```
#Configuration options for predicate_merge optimizations
```

### **[search\_optimization::predicate\_merge]**

```
[search_optimization::predicate_merge]
enabled = <bool>
* Enables predicate merge optimization
```

\* Defaults to true

## **[mvexpand]**

[mvexpand]

\* This stanza allows for fine tuning of mvexpand search command.

max\_mem\_usage\_mb = <non-negative integer>

\* Overrides the default value for max\_mem\_usage\_mb

\* See definition in [default] max\_mem\_usage\_mb for more details

\* Defaults to 500 (MB)

## **[mvcombine]**

[mvcombine]

\* This stanza allows for fine tuning of mvcombine search command.

max\_mem\_usage\_mb = <non-negative integer>

\* overrides the default value for max\_mem\_usage\_mb

\* See definition in [default] max\_mem\_usage\_mb for more details

\* defaults to 500 (MB)

## **[xyseries]**

[xyseries]

\* This stanza allows for fine tuning of xyseries search command.

max\_mem\_usage\_mb = <non-negative integer>

\* overrides the default value for max\_mem\_usage\_mb

\* See definition in [default] max\_mem\_usage\_mb for more details

## **limits.conf.example**

# Version 6.5.0

# CAUTION: Do not alter the settings in limits.conf unless you know what you are doing.

# Improperly configured limits may result in splunkd crashes and/or memory overuse.

[searchresults]

maxresultrows = 50000

# maximum number of times to try in the atomic write operation (1 = no retries)

tocsv\_maxretry = 5

# retry period is 1/2 second (500 milliseconds)

tocsv\_retryperiod\_ms = 500

[subsearch]

# maximum number of results to return from a subsearch

maxout = 100

# maximum number of seconds to run a subsearch before finalizing

maxtime = 10

# time to cache a given subsearch's results

t11 = 300

[anomalousvalue]

maxresultrows = 50000

# maximum number of distinct values for a field

maxvalues = 100000

# maximum size in bytes of any single value (truncated to this size if larger)

maxvaluesize = 1000

[associate]

maxfields = 10000

maxvalues = 10000

maxvaluesize = 1000

# for the contingency, ctable, and counttable commands

```

[table]
maxvalues = 1000

[correlate]
maxfields = 1000

# for bin/bucket/discretize
[discretize]
maxbins = 50000
# if maxbins not specified or = 0, defaults to searchresults::maxresultrows

[inputcsv]
# maximum number of retries for creating a tmp directory (with random name in
# SPLUNK_HOME/var/run/splunk)
mkdir_max_retries = 100

[kmeans]
maxdatapoints = 100000000

[kv]
# when non-zero, the point at which kv should stop creating new columns
maxcols = 512

[rare]
maxresultrows = 50000
# maximum distinct value vectors to keep track of
maxvalues = 100000
maxvaluesize = 1000

[restapi]
# maximum result rows to be returned by /events or /results getters from REST
# API
maxresultrows = 50000

[search]
# how long searches should be stored on disk once completed
ttl = 86400

# the approximate maximum number of timeline buckets to maintain
status_buckets = 300

# the last accessible event in a call that takes a base and bounds
max_count = 10000

# the minimum length of a prefix before a * to ask the index about
min_prefix_len = 1

# the length of time to persist search cache entries (in seconds)
cache_ttl = 300

[scheduler]

# User default value (needed only if different from system/default value) when
# no max_searches_perc.<n>.when (if any) below matches.
max_searches_perc = 60

# Increase the value between midnight-5AM.
max_searches_perc.0 = 75
max_searches_perc.0.when = * 0-5 * * *

# More specifically, increase it even more on weekends.
max_searches_perc.1 = 85
max_searches_perc.1.when = * 0-5 * * 0,6

[slc]
# maximum number of clusters to create
maxclusters = 10000

[findkeywords]
#events to use in findkeywords command (and patterns UI)
maxevents = 50000

```

```

[stats]
maxresultrows = 50000
maxvalues = 10000
maxvaluesize = 1000

[top]
maxresultrows = 50000
# maximum distinct value vectors to keep track of
maxvalues = 100000
maxvaluesize = 1000

[search_optimization]
enabled = true

[search_optimization::predicate_push]
enabled = true

[search_optimization::predicate_merge]
enabled = true

```

## literals.conf

以下为 literals.conf 的规范和示例文件。

### literals.conf.spec

```

# Version 6.5.0
#
# This file contains attribute/value pairs for configuring externalized strings
# in literals.conf.
#
# There is a literals.conf in $SPLUNK_HOME/etc/system/default/. To set custom
# configurations, place a literals.conf in $SPLUNK_HOME/etc/system/local/. For
# examples, see literals.conf.example. You must restart Splunk to enable
# configurations.
#
# To learn more about configuration files (including precedence) please see the
# documentation located at
# http://docs.splunk.com/Documentation/Splunk/latest/Admin/Aboutconfigurationfiles
#
# For the full list of all literals that can be overridden, check out
# $SPLUNK_HOME/etc/system/default/literals.conf.

#####
#
# CAUTION:
#
# - You can destroy Splunk's performance by editing literals.conf incorrectly.
#
# - Only edit the attribute values (on the right-hand side of the '=').
#   DO NOT edit the attribute names (left-hand side of the '=').
#
# - When strings contain "%s", do not add or remove any occurrences of %s, or
#   reorder their positions.
#
# - When strings contain HTML tags, take special care to make sure that all
#   tags and quoted attributes are properly closed, and that all entities such
#   as & are escaped.
#

```

### literals.conf.example

```
# Version 6.5.0
#
# This file contains an example literals.conf, which is used to
# configure the externalized strings in Splunk.
#
# For the full list of all literals that can be overwritten, consult
# the far longer list in $SPLUNK_HOME/etc/system/default/literals.conf
#

[ui]
PRO_SERVER_LOGIN_HEADER = Login to Splunk (guest/guest)
INSUFFICIENT_DISK_SPACE_ERROR = The server's free disk space is too low. Indexing will temporarily pause until more
disk space becomes available.
SERVER_RESTART_MESSAGE = This Splunk Server's configuration has been changed. The server needs to be restarted by
an administrator.
UNABLE_TO_CONNECT_MESSAGE = Could not connect to splunkd at %s.
```

## macros.conf

以下为 macros.conf 的规范和示例文件。

### macros.conf.spec

```
# Version 6.5.0
#
# This file contains possible attribute/value pairs for search language macros.

# To learn more about configuration files (including precedence) please see the
# documentation located at
# http://docs.splunk.com/Documentation/Splunk/latest/Admin/Aboutconfigurationfiles
```

#### [<STANZA\_NAME>]

```
[<STANZA_NAME>]
* Each stanza represents a search macro that can be referenced in any search.
* The stanza name is the name of the macro if the macro takes no arguments.
  Otherwise, the stanza name is the macro name appended with "<numargs>",
  where <numargs> is the number of arguments that this macro takes.
* Macros can be overloaded. In other words, they can have the same name but a
  different number of arguments. If you have [foobar], [foobar(1)],
  [foobar(2)], etc., they are not the same macro.
* Macros can be used in the search language by enclosing the macro name and any
  argument list within tick marks, for example: `foobar(arg1,arg2)` or `footer`.
* Splunk does not expand macros when they are inside of quoted values, for
  example: "foo`bar`baz".

args = <string>,<string>,...
* A comma-delimited string of argument names.
* Argument names can only contain alphanumeric characters, underscores '_', and
  hyphens '-'.
* If the stanza name indicates that this macro takes no arguments, this
  attribute will be ignored.
* This list cannot contain any repeated elements.

definition = <string>
* The string that the macro will expand to, with the argument substitutions
  made. (The exception is when iseval = true, see below.)
* Arguments to be substituted must be wrapped by dollar signs ($), for example:
  "the last part of this string will be replaced by the value of argument foo $foo$".
* Splunk replaces the $<arg>$ pattern globally in the string, even inside of
  quotes.

validation = <string>
* A validation string that is an 'eval' expression. This expression must
  evaluate to a boolean or a string.
* Use this to verify that the macro's argument values are acceptable.
* If the validation expression is boolean, validation succeeds when it returns
```

true. If it returns false or is NULL, validation fails, and Splunk returns the error message defined by the attribute, `errmsg`.

- \* If the validation expression is not boolean, Splunk expects it to return a string or NULL. If it returns NULL, validation is considered a success. Otherwise, the string returned is the error string.

`errmsg = <string>`

- \* The error message to be displayed if validation is a boolean expression and it does not evaluate to true.

`iseval = <true/false>`

- \* If true, the definition attribute is expected to be an eval expression that returns a string that represents the expansion of this macro.
- \* Defaults to false.

`description = <string>`

- \* OPTIONAL. Simple english description of what the macro does.

## macros.conf.example

```
# Version 6.5.0
#
# Example macros.conf
#

# macro foobar that takes no arguments can be invoked via `foobar`
[foobar]
# the definition of a macro can invoke another macro. nesting can be indefinite
# and cycles will be detected and result in an error
definition = `foobar(foo=defaultfoo)`

# macro foobar that takes one argument, invoked via `foobar(someval)`
[foobar(1)]
args = foo
# note this is definition will include the leading and trailing quotes, i.e.
# something `foobar(someval)`
# would expand to
# something "foo = someval"
definition = "foo = $foo$"

# macro that takes two arguments
# note that macro arguments can be named so this particular macro could be
# invoked equivalently as `foobar(1,2)` `foobar(foo=1,bar=2)` or
# `foobar(bar=2,foo=1)`
[foobar(2)]
args = foo, bar
definition = "foo = $foo$, bar = $bar$"

# macro that takes one argument that does validation
[foovaild(1)]
args = foo
definition = "foovaild = $foo$"
# the validation eval function takes any even number of arguments (>=2) where
# the first argument is a boolean expression, the 2nd a string, the third
# boolean, 4th a string, etc etc etc
validation = validate(foo>15,"foo must be greater than 15",foo<=100,"foo must be <= 100")

# macro showing simple boolean validation, where if foo > bar is not true,
# errmsg is displayed
[foovaild(2)]
args = foo, bar
definition = "foo = $foo$ and bar = $bar$"
validation = foo > bar
errmsg = foo must be greater than bar

# example of an eval-based definition. For example in this case
# `fooeval(10,20)` would get replaced by 10 + 20
[fooeval(2)]
args = foo, bar
```



```
definition = if (bar > 0, "$foo$ + $bar$", "$foo$ - $bar$")
iseval = true
```

## multikv.conf

以下为 multikv.conf 的规范和示例文件。

### multikv.conf.spec

```
# Version 6.5.0
#
# This file contains possible attribute and value pairs for creating multikv
# rules. Multikv is the process of extracting events from table-like events,
# such as the output of top, ps, ls, netstat, etc.
#
# There is NO DEFAULT multikv.conf. To set custom configurations, place a
# multikv.conf in $SPLUNK_HOME/etc/system/local/. For examples, see
# multikv.conf.example. You must restart Splunk to enable configurations.
#
# To learn more about configuration files (including precedence) see the
# documentation located at
# http://docs.splunk.com/Documentation/Splunk/latest/Admin/Aboutconfigurationfiles
#
# NOTE: Only configure multikv.conf if Splunk's default multikv behavior does
# not meet your needs.

# A table-like event includes a table consisting of four sections:
#
```

#### 章节名称 | 描述

```
#-----
# Section Name | Description
#-----Section Name | Description
# pre          | optional: info/description (for example: the system summary output in top)
# header       | optional: if not defined, fields are named Column_N
# body         | required: the body of the table from which child events are constructed
# post         | optional: info/description
#-----

# NOTE: Each section must have a definition and a processing component. See
# below.

[<multikv_config_name>]
* Name of the stanza to use with the multikv search command, for example:
  '| multikv conf=<multikv_config_name> rmorig=f | ....'
* Follow this stanza name with any number of the following attribute/value pairs.
```

#### 章节定义

```
#####
# Section Definition
#####Section Definition
# Define where each section begins and ends.

<Section Name>.start = <regex>
* A line matching this regex denotes the start of this section (inclusive).

OR

<Section Name>.start_offset = <int>
* Line offset from the start of an event or the end of the previous section
  (inclusive).
* Use this if you cannot define a regex for the start of the section.

<Section Name>.member = <regex>
```

```

* A line membership test.
* Member if lines match the regex.

<Section Name>.end = <regex>
* A line matching this regex denotes the end of this section (exclusive).

```

OR

```

<Section Name>.linecount = <int>
* Specify the number of lines in this section.
* Use this if you cannot specify a regex for the end of the section.

```

## 章节处理

```

#####
# Section processing
#####Section processing
# Set processing for each section.

<Section Name>.ignore = [_all_|_none_|_regex_ <regex-list>]
* Determines which member lines will be ignored and not processed further.

<Section Name>.replace = <quoted-str> = <quoted-str>, <quoted-str> = <quoted-str>,...
* List of the form: "toReplace" = "replaceWith".
* Can have any number of quoted string pairs.
* For example: "%" = "_", "#" = "_"

<Section Name>.tokens = [<chopper>|<tokenizer>|<aligner>|<token-list>]
* See below for definitions of each possible token: chopper, tokenizer, aligner,
  token-list.

<chopper> = _chop_, <int-list>
* Transform each string into a list of tokens specified by <int-list>.
* <int-list> is a list of (offset, length) tuples.

<tokenizer> = _tokenize_ <max_tokens (int)> <delims> (<consume-delims>)?
* Tokenize the string using the delim characters.
* This generates at most max_tokens number of tokens.
* Set max_tokens to:
  * -1 for complete tokenization.
  * 0 to inherit from previous section (usually header).
  * A non-zero number for a specific token count.
* If tokenization is limited by the max_tokens, the rest of the string is added
  onto the last token.
* <delims> is a comma-separated list of delimiting chars.
* <consume-delims> - boolean, whether to consume consecutive delimiters. Set to
  false/0 if you want consecutive delimiters to be treated
  as empty values. Defaults to true.

<aligner> = _align_, <header_string>, <side>, <max_width>
* Generates tokens by extracting text aligned to the specified header fields.
* header_string: a complete or partial header field value the columns are aligned with.
* side: either L or R (for left or right align, respectively).
* max_width: the maximum width of the extracted field.
  * Set max_width to -1 for automatic width. This expands the field until any
    of the following delimiters are found: " ", "\t"

<token_list> = _token_list_ <comma-separated list>
* Defines a list of static tokens in a section.
* This is useful for tables with no header, for example: the output of 'ls -lah'
  which misses a header altogether.

```

## multikv.conf.example

```

# Version 6.5.0
#
# This file contains example multi key/value extraction configurations.
#

```

```

# To use one or more of these configurations, copy the configuration block into
# multikv.conf in $SPLUNK_HOME/etc/system/local/. You must restart Splunk to
# enable configurations.
#
# To learn more about configuration files (including precedence) please see the
# documentation located at
# http://docs.splunk.com/Documentation/Splunk/latest/Admin/Aboutconfigurationfiles

# This example breaks up the output from top:

# Sample output:

# Processes: 56 total, 2 running, 54 sleeping... 221 threads 10:14:07
#.....
#
#   PID COMMAND   %CPU TIME      #TH #PTS #MREGS RPRVT RSHRD RSIZE  VSIZE
# 29960 mdimport  0.0% 0:00.29  3    60    50  1.10M  2.55M 3.54M 38.7M
# 29905 pickup   0.0% 0:00.01  1    16    17   164K   832K  764K 26.7M
#....

[top_mkv]
# pre table starts at "Process..." and ends at line containing "PID"
pre.start = "Process"
pre.end = "PID"
pre.ignore = _all_

# specify table header location and processing
header.start = "PID"
header.linecount = 1
header.replace = "%" = "_", "#" = "_"
header.tokens = _tokenize_, -1, " "

# table body ends at the next "Process" line (ie start of another top) tokenize
# and inherit the number of tokens from previous section (header)
body.end = "Process"
body.tokens = _tokenize_, 0, " "

## This example handles the output of 'ls -lah' command:
#
# total 2150528
# drwxr-xr-x 88 john john 2K Jan 30 07:56 .
# drwxr-xr-x 15 john john 510B Jan 30 07:49 ..
# -rw----- 1 john john 2K Jan 28 11:25 .hidden_file
# drwxr-xr-x 20 john john 680B Jan 30 07:49 my_dir
# -r--r--r-- 1 john john 3K Jan 11 09:00 my_file.txt

[ls-lah-cpp]
pre.start = "total"
pre.linecount = 1

# the header is missing, so list the column names
header.tokens = _tokenize_list_, mode, links, user, group, size, date, name

# The ends when we have a line starting with a space
body.end = "^\\s*$"

# This filters so that only lines that contain with .cpp are used
body.member = ".cpp"
# concatenates the date into a single unbreakable item
body.replace = "(\\w{3})\\s+(\\d{1,2})\\s+(\\d{2}:\\d{2})" = "1_2_3"

# ignore dirs
body.ignore = _regex_ "^drwx.*",
body.tokens = _tokenize_, 0, " "

```

## outputs.conf

以下为 outputs.conf 的规范和示例文件。

## outputs.conf.spec

```
# Version 6.5.0
#
# Forwarders require outputs.conf; non-forwarding Splunk instances do not
# use it. It determines how the forwarder sends data to receiving Splunk
# instances, either indexers or other forwarders.
#
# To configure forwarding, create an outputs.conf file in
# $SPLUNK_HOME/etc/system/local/. For examples of its use, see
# outputs.conf.example.
#
# You must restart Splunk to enable configurations.
#
# To learn more about configuration files (including precedence) please see
# the documentation located at
# http://docs.splunk.com/Documentation/Splunk/latest/Admin/Aboutconfigurationfiles
#
# NOTE: To learn more about forwarding, see the documentation at
# http://docs.splunk.com/Documentation/Splunk/latest/Deploy/Aboutforwardingandreceivingdata
```

## 全局设置

```
# GLOBAL SETTINGS
# Use the [default] stanza to define any global settings.
# * You can also define global settings outside of any stanza, at the top
#   of the file.
# * Each conf file should have at most one default stanza. If there are
#   multiple default stanzas, attributes are combined. In the case of
#   multiple definitions of the same attribute, the last definition in the
#   file wins.
# * If an attribute is defined at both the global level and in a specific
#   stanza, the value in the specific stanza takes precedence.

#####
TCP Output stanzas
#####
# There are three levels of TCP Output stanzas:
# * Global: [tcpout]
# * Target group: [tcpout:<target_group>]
# * Single server: [tcpout-server://<ip address>:<port>]
#
# Settings at more specific levels override settings at higher levels. For
# example, an attribute set for a single server overrides the value of that
# attribute, if any, set at that server's target group stanza. See the
# online documentation on configuring forwarders for details.
#
# This spec file first describes the three levels of stanzas (and any
# attributes unique to a particular level). It then describes the optional
# attributes, which can be set at any of the three levels.

#----TCP Output Global Configuration ----
# The global configurations specified here in the [tcpout] stanza can be
# overwritten in stanzas for specific target groups, as described later.
# Note that the defaultGroup and indexAndForward attributes can only be set
# here, at the global level.
#
# Starting with 4.2, the [tcpout] stanza is no longer required.

[tcpout]

defaultGroup = <target_group>, <target_group>, ...
* Comma-separated list of one or more target group names, specified later
  in [tcpout:<target_group>] stanzas.
* The forwarder sends all data to the specified groups.
```

- \* If you don't want to forward data automatically, don't set this attribute.
- \* Can be overridden by an inputs.conf \_TCP\_ROUTING setting, which in turn can be overridden by a props.conf/transforms.conf modifier.
- \* Starting with 4.2, this attribute is no longer required.

indexAndForward = [true|false]

- \* Index all data locally, in addition to forwarding it.
- \* This is known as an "index-and-forward" configuration.
- \* This attribute is only available for heavy forwarders.
- \* This attribute is available only at the top level [tcpout] stanza. It cannot be overridden in a target group.
- \* Defaults to false.

#----Target Group Configuration ----

- # If multiple servers are specified in a target group, the forwarder performs auto load-balancing, sending data alternately to each available server in the group. For example, assuming you have three servers (server1, server2, server3) and autoLBFrequency=30, the forwarder sends all data to server1 for 30 seconds, then it sends all data to server2 for the next 30 seconds, then all data to server3 for the next 30 seconds, finally cycling back to server1.
- #
- # You can have as many target groups as you want.
- # If more than one target group is specified, the forwarder sends all data to each target group.
- # This is known as "cloning" the data.

[tcpout:<target\_group>]

server = [<ip>|<servername>]:<port>, [<ip>|<servername>]:<port>, ...

- \* Required if indexerDiscovery is not set.
- \* Takes a comma separated list of one or more systems to send data to over a tcp socket.
- \* Typically used to specify receiving splunk systems, although it can be used to send data to non-splunk systems (see sendCookedData setting).
- \* For each mentioned system, the following are required:
  - \* IP or servername where one or system is listening.
  - \* Port on which syslog server is listening.

blockWarnThreshold = <integer>

- \* Optional
- \* Default value is 100
- \* Sets the output pipeline send failure count threshold after which a failure message will be displayed as banner on UI
- \* To disable any warnings to be sent to UI on blocked output queue condition, set this to a large value (2 million for example)

indexerDiscovery = <name>

- \* Instructs the forwarder to fetch the list of indexers from the master node specified in the corresponding [indexer\_discovery:<name>] stanza.

token = <string>

- \* Optional
- \* If an access token is configured for receiving Splunk system, that token is populated here. Note that if receiver is configured with an access token and that token is not specified here, then data sent to it will be rejected.

#----Single server configuration ----

- # You can define specific configurations for individual indexers on a server-by-server basis. However, each server must also be part of a target group.

[tcpout-server://<ip address>:<port>]

- \* Optional. There is no requirement to have any tcpout-server stanzas.

## **TCPOUT ATTRIBUTES----**

```
#####
#----TCPOUT ATTRIBUTES----
#####TCPOUT ATTRIBUTES----
# These attributes are optional and can appear in any of the three stanza levels.

[tcput<any of above>]

#----General Settings----

sendCookedData = [true|false]
* If true, events are cooked (have been processed by Splunk).
* If false, events are raw and untouched prior to sending.
* Set to false if you are sending to a third-party system.
* Defaults to true.

heartbeatFrequency = <integer>
* How often (in seconds) to send a heartbeat packet to the receiving server.
* Heartbeats are only sent if sendCookedData=true.
* Defaults to 30 (seconds).

blockOnCloning = [true|false]
* If true, TcpOutputProcessor blocks till at least one of the cloned group
  gets events. This will not drop events when all the cloned groups are
  down.
* If false, TcpOutputProcessor will drop events when all the cloned groups
  are down and queues for the cloned groups are full. When at least one of
  the cloned groups is up and queues are not full, the events are not
  dropped.
* Defaults to true.

# For the following setting see the [tcput:<target_group>] stanza
blockWarnThreshold = <integer>

compressed = [true|false]
* Applies to non-SSL forwarding only. For SSL useClientSSLCompression
  setting is used.
* If true, forwarder sends compressed data.
* If set to true, the receiver port must also have compression turned on (in
  its inputs.conf file).
* Defaults to false.

negotiateNewProtocol = [true|false]
* When setting up a connection to an indexer, try to negotiate the use of
  the new forwarder protocol.
* If set to false, the forwarder will not query the indexer for support for
  the new protocol, and the connection will fall back on the traditional
  protocol.
* Defaults to true.

channelReapInterval = <integer>
* Controls how often, in milliseconds, channel codes are reaped, i.e. made
  available for re-use.
* This value sets the minimum time between reapings; in practice,
  consecutive reapings may be separated by greater
  than <channelReapInterval> milliseconds.
* Defaults to 60000 (1 minute)

channelTTL = <integer>
* Controls how long, in milliseconds, a channel may remain "inactive" before
  it is reaped, i.e. before its code is made available for re-use by a
  different channel.
* Defaults to 300000 (5 minutes)

channelReapLowater = <integer>
* If the number of active channels is above <channelReapLowater>, we reap
  old channels in order to make their channel codes available for re-use.
* If the number of active channels is below <channelReapLowater>, we do not
  reap channels, no matter how old they are.
* This value essentially determines how many active-but-old channels we keep
  "pinned" in memory on both sides of a splunk-to-splunk connection.
* A non-zero value helps ensure that we do not waste network resources by
  "thrashing" channels in the case of a forwarder sending a trickle of data.
```

- \* Defaults to 10.

socksServer = [<ip>|<servername>]:<port>

- \* IP or servername of Socks5 server.
- \* Port on which socks server is listening on. You must specify the port.
- \* Note: Only Socks5 is supported.

socksUsername = <username>

- \* Optional
- \* Socks username to use when authenticating against socks server

socksPassword = <password>

- \* Optional
- \* Socks password to use when authenticating against socks server

socksResolveDNS = <bool>

- \* Optional
- \* If set to true, forwarder will not attempt to resolve indexer's DNS, and
- \* will forward the indexer's DNS as is to let socks server resolve it.

#---Queue Settings---

maxQueueSize = [<integer>|<integer>[KB|MB|GB]|auto]

- \* This attribute sets the maximum size of the forwarder's output queue.
- \* The size can be limited based on the number of entries, or on the total memory used by the items in the queue.
- \* If specified as a lone integer (for example, maxQueueSize=100), maxQueueSize indicates the maximum count of queued items.
- \* If specified as an integer followed by KB, MB, or GB (for example, maxQueueSize=100MB), maxQueueSize indicates the maximum RAM size of all the items in the queue.
- \* If set to auto, chooses a value depending on whether useACK is enabled.
  - \* If useACK=false, uses 500KB
  - \* If useACK=true, uses 7MB
- \* If the useACK setting is enabled, the maximum size of the wait queue is set to to 3x this value.
- \* Although the wait queue and the output queue sizes are both controlled by this attribute, they are separate.
- \* Limiting the queue sizes by quantity is largely historical. However, should you choose to configure queues based on quantity, keep the following in mind:
  - \* Queued items can be events or blocks of data.
    - \* Non-parsing forwarders, such as universal forwarders, will send blocks, which may be up to 64KB.
    - \* Parsing forwarders, such as heavy forwarders, will send events, which will be the size of the events. For some events these are as small as a few hundred bytes. In unusual cases (data dependent), customers may arrange to produce events that are multiple megabytes.
- \* Defaults to auto
  - \* If useACK is enabled, effectively defaults the wait queue to 21MB

dropEventsOnQueueFull = <integer>

- \* If set to a positive number, wait <integer> seconds before throwing out all new events until the output queue has space.
- \* Setting this to -1 or 0 will cause the output queue to block when it gets full, causing further blocking up the processing chain.
- \* If any target group's queue is blocked, no more data will reach any other target group.
- \* Using auto load-balancing is the best way to minimize this condition, because, in that case, multiple receivers must be down (or jammed up) before queue blocking can occur.
- \* Defaults to -1 (do not drop events).
- \* DO NOT SET THIS VALUE TO A POSITIVE INTEGER IF YOU ARE MONITORING FILES!

dropClonedEventsOnQueueFull = <integer>

- \* If set to a positive number, do not block completely, but wait up to <integer> seconds to queue events to a group. If it cannot enqueue to a group for more than <integer> seconds, begin dropping events for the group. It makes sure that at least one group in the cloning configuration will get events. It blocks if event cannot be delivered to any of the cloned groups.
- \* If set to -1, the TcpOutputProcessor will make sure that each group will

```

    get all of the events. If one of the groups is down, then Splunk will
    block everything.
* Defaults to 5.

#----Backoff Settings When Unable To Send Events to Indexer----
# The settings in this section determine forwarding behavior when there are
# repeated failures in sending events to an indexer ("sending failures").

maxFailuresPerInterval = <integer>
* Specifies the maximum number failures allowed per interval before backoff
  takes place. The interval is defined below.
* Defaults to 2.

secsInFailureInterval = <integer>
* Number of seconds in an interval. If the number of write failures exceeds
  maxFailuresPerInterval in the specified secsInFailureInterval seconds, the
  forwarder applies backoff. The backoff time period range is
  1-10 * autoLBFrequency.
* Defaults to 1.

backoffOnFailure = <positive integer>
* Number of seconds a forwarder will wait before attempting another
  connection attempt.
* Defaults to 30

maxConnectionsPerIndexer = <integer>
* Maximum number of allowed connections per indexer. In presence of
  failures, the max number of connection attempt per indexer at any point in
  time.
* Defaults to 2.

connectionTimeout = <integer>
* Time out period if connection establishment does not finish in <integer>
  seconds.
* Defaults to 20 seconds.

readTimeout = <integer>
* Time out period if read from socket does not finish in <integer> seconds.
* This timeout is used to read acknowledgment when indexer acknowledgment is
  used (useACK=true).
* Defaults to 300 seconds.

writeTimeout = <integer>
* Time out period if write on socket does not finish in <integer> seconds.
* Defaults to 300 seconds.

tcpSendBufSz = <integer>
* TCP send buffer size in <integer> bytes.
* Useful to improve throughput with small size events like windows events.
* Only set this value if you are a TCP/IP expert.
* Defaults to system default.

ackTimeoutOnShutdown = <integer>
* Time out period if ACKs not received in <integer> seconds during forwarder shutdown.
* Defaults to 30 seconds.

dnsResolutionInterval = <integer>
* Specifies base time interval in seconds at which indexer dns names will be
  resolved to ip address. This is used to compute runtime
  dnsResolutionInterval as follows:
  runtime interval = dnsResolutionInterval + (number of indexers in server settings - 1)*30.
  DNS resolution interval is extended by 30 second for each additional
  indexer in server setting.
* Defaults to 300 seconds.

forceTimebasedAutoLB = [true|false]
* Will force existing streams to switch to newly elected indexer every
  AutoLB cycle.
* Defaults to false

#----Index Filter Settings.
# These attributes are only applicable under the global [tcpout] stanza.

```



```

# This filter does not work if it is created under any other stanza.
forwardedindex.<n>.whitelist = <regex>
forwardedindex.<n>.blacklist = <regex>
* These filters determine which events get forwarded, based on the indexes
  the events belong are targetting.
* This is an ordered list of whitelists and blacklists, which together
  decide if events should be forwarded to an index.
* The order is determined by <n>. <n> must start at 0 and continue with
  positive integers, in sequence. There cannot be any gaps in the sequence.
  * For example:
      forwardedindex.0.whitelist, forwardedindex.1.blacklist, forwardedindex.2.whitelist, ...
* The filters can start from either whitelist or blacklist. They are tested
  from forwardedindex.0 to forwardedindex.<max>.
* If both forwardedindex.<n>.whitelist and forwardedindex.<n>.blacklist are
  present for the same value of n, then forwardedindex.<n>.whitelist is
  honored. forwardedindex.<n>.blacklist is ignored in this case.
* You should not normally need to change these filters from their default
  settings in $SPLUNK_HOME/system/default/outputs.conf.
* Filtered out events are not indexed if local indexing is not enabled.

forwardedindex.filter.disable = [true|false]
* If true, disables index filtering. Events for all indexes are then
  forwarded.
* Defaults to false.

#----Automatic Load-Balancing
autoLB = true
* Automatic load balancing is the only way to forward data. Round-robin
  method is not supported anymore.
* Defaults to true.

autoLBFrequency = <seconds>
* Every autoLBFrequency seconds, a new indexer is selected randomly from the
  list of indexers provided in the server attribute of the target group
  stanza.
* Defaults to 30 (seconds).

#----SSL Settings----

# To set up SSL on the forwarder, set the following attribute/value pairs.
# If you want to use SSL for authentication, add a stanza for each receiver
# that must be certified.

sslPassword = <password>
* The password associated with the CAcert.
* The default Splunk CAcert uses the password "password".
* There is no default value.

clientCert = <path>
* The full path to the client SSL certificate in PEM format.
* If (and only if) specified, this connection will use SSL.
* There is no default value.

sslCertPath = <path>
* DEPRECATED; use 'clientCert' instead.

cipherSuite = <string>
* If set, uses the specified cipher string for the input processors.
* If not set, the default cipher string provided by OpenSSL is used.
* This is used to ensure that the server does not accept connections using weak
  encryption protocols.

sslCipher = <string>
* DEPRECATED; use 'cipherSuite' instead.

ecdhCurves = <comma separated list of ec curves>
* ECDH curves to use for ECDH key negotiation.
* The curves should be specified in the order of preference.
* The client sends these curves as a part of Client Hello.
* The server supports only the curves specified in the list.
* We only support named curves specified by their SHORT names.
  (see struct ASN1_OBJECT in asn1.h)

```

```

* The list of valid named curves by their short/long names can be obtained
  by executing this command:
  $SSLUNK_HOME/bin/splunk cmd openssl ecparam -list_curves
* Default is empty string.
* e.g. ecdhCurves = prime256v1,secp384r1,secp521r1

sslRootCAPath = <path>
* DEPRECATED; use 'server.conf/[sslConfig]/sslRootCAPath' instead.
* Used only if server.conf's 'sslRootCAPath' is unset.
* Full path to the root CA (Certificate Authority) certificate store.
* The <path> must refer to a PEM format file containing one or more root CA
  certificates concatenated together.
* Default is unset.

sslVerifyServerCert = <bool>
* If true, you must make sure that the server you are connecting to is a
  valid one (authenticated).
* Both the common name and the alternate name of the server are then checked
  for a match.
* Defaults to false.

tlsHostname = <string>
* TLS extension that allows sending an identifier with SSL Client Hello
* Defaults to empty string

sslCommonNameToCheck = <commonName1>, <commonName2>, ...
* Optional. Defaults to no common name checking.
* Check the common name of the server's certificate against this name.
* If there is no match, assume that Splunk is not authenticated against this
  server.
* 'sslVerifyServerCert' must be set to true for this setting to work.

sslAltNameToCheck = <alternateName1>, <alternateName2>, ...
* Optional. Defaults to no alternate name checking.
* Check the alternate name of the server's certificate against this list of names.
* If there is no match, assume that Splunk is not authenticated against this
  server.
* 'sslVerifyServerCert' must be set to true for this setting to work.

useClientSSLCompression = <bool>
* Enables compression on SSL.
* Defaults to value of 'server.conf/[sslConfig]/useClientSSLCompression'.

sslQuietShutdown = <bool>
* Enables quiet shutdown mode in SSL
* Defaults to false

sslVersions = <string>
* Comma-separated list of SSL versions to support
* The versions available are "ssl3", "tls1.0", "tls1.1", and "tls1.2"
* The special version "*" selects all supported versions. The version "tls"
  selects all versions tls1.0 or newer
* If a version is prefixed with "-" it is removed from the list
* SSLv2 is always disabled; "-ssl2" is accepted in the version list but does nothing
* When configured in FIPS mode ssl3 is always disabled regardless of
  this configuration
* Defaults to "*",-ssl2". (anything newer than SSLv2)

#----Indexer Acknowledgment ----
# Indexer acknowledgment ensures that forwarded data is reliably delivered
# to the receiver.
# If the receiver is an indexer, it indicates that the indexer has received
# the data, indexed it, and written it to the file system. If the receiver
# is an intermediate forwarder, it indicates that the intermediate forwarder
# has successfully forwarded the data to the terminating indexer and has
# received acknowledgment from that indexer.

# Important: Indexer acknowledgment is a complex feature that requires
# careful planning. Before using it, read the online topic describing it in
# the Distributed Deployment manual.

useACK = [true|false]

```

- \* When set to true, the forwarder will retain a copy of each sent event, until the receiving system sends an acknowledgement.
- \* The receiver will send an acknowledgement when it has fully handled it (typically written it to disk in indexing)
- \* In the event of receiver misbehavior (acknowledgement is not received), the data will be re-sent to an alternate receiver.
- \* Note: the maximum memory used for the outbound data queues will increase significantly by default (500KB -> 28MB) when useACK is enabled. This is intended for correctness and performance.
- \* When set to false, the forwarder will consider the data fully processed when it finishes writing it to the network socket.
- \* This attribute can be set at the [tcpout] or [tcpout:<target\_group>] stanza levels. You cannot set it for individual servers at the [tcpout-server: ...] stanza level.
- \* Defaults to false.

## Syslog output----

```
#####
#----Syslog output----
#####Syslog output----
# The syslog output processor is not available for universal or light
# forwarders.

# The following configuration is used to send output using syslog:

[syslog]
defaultGroup = <target_group>, <target_group>, ...

# For the following settings see the [syslog:<target_group>] stanza below
type = [tcp|udp]
priority = <priority_value> | NO_PRI
dropEventsOnQueueFull = <integer>
maxEventSize = <integer>

[syslog:<target_group>]

#----REQUIRED SETTINGS----
# Required settings for a syslog output group:

server = [<ip>|<servername>]:<port>
* IP or servername where syslog server is running.
* Port on which server is listening. You must specify the port. Syslog, by
  default, uses 514.

#----OPTIONAL SETTINGS----

# Optional settings for syslog output:

type = [tcp|udp]
* Protocol used.
* Default is udp.

priority = <priority_value> | NO_PRI
* The priority_value should be specified as "<integer>" (an integer surrounded
  by angle brackets). For example, specify a priority of 34 like this: <34>
* The integer must be one to three digits in length.
* The value you enter will appear in the syslog header.
* Mimics the number passed via syslog interface call, documented via man
  syslog.
* The integer can be computed as (<facility> * 8) + <severity>. For example,
  if <facility> is 4 (security/authorization messages) and <severity> is 2
  (critical conditions), the priority will be 34 = (4 * 8) + 2. Set the
  attribute to: <34>
* The table of facility and severity (and their values) can be referenced in
  RFC3164, eg http://www.ietf.org/rfc/rfc3164.txt section 4.1.1
* Defaults to <13>, or a facility of "user" or typically unspecified
  application, and severity of "Notice".
* If you do not wish to add priority, set 'NO_PRI' as priority value.
  * Example: priority = NO_PRI
```

\* The table is reproduced briefly here, some of these are archaic.

Facility:

- 0 kernel messages
- 1 user-level messages
- 2 mail system
- 3 system daemons
- 4 security/authorization messages
- 5 messages generated internally by syslogd
- 6 line printer subsystem
- 7 network news subsystem
- 8 UUCP subsystem
- 9 clock daemon
- 10 security/authorization messages
- 11 FTP daemon
- 12 NTP subsystem
- 13 log audit
- 14 log alert
- 15 clock daemon
- 16 local use 0 (local0)
- 17 local use 1 (local1)
- 18 local use 2 (local2)
- 19 local use 3 (local3)
- 20 local use 4 (local4)
- 21 local use 5 (local5)
- 22 local use 6 (local6)
- 23 local use 7 (local7)

Severity:

- 0 Emergency: system is unusable
- 1 Alert: action must be taken immediately
- 2 Critical: critical conditions
- 3 Error: error conditions
- 4 Warning: warning conditions
- 5 Notice: normal but significant condition
- 6 Informational: informational messages
- 7 Debug: debug-level messages

syslogSourceType = <string>

- \* Specifies an additional rule for handling data, in addition to that provided by the 'syslog' source type.
- \* This string is used as a substring match against the sourcetype key. For example, if the string is set to 'syslog', then all source types containing the string 'syslog' will receive this special treatment.
- \* To match a source type explicitly, use the pattern "sourcetype::sourcetype\_name".
  - \* Example: syslogSourceType = sourcetype::apache\_common
- \* Data which is 'syslog' or matches this setting is assumed to already be in syslog format.
- \* Data which does not match the rules has a header, optionally a timestamp (if defined in 'timestampformat'), and a hostname added to the front of the event. This is how Splunk causes arbitrary log data to match syslog expectations.
- \* Defaults to unset.

timestampformat = <format>

- \* If specified, the formatted timestamps are added to the start of events forwarded to syslog.
- \* As above, this logic is only applied when the data is not syslog, or the syslogSourceType.
- \* If the data is not in syslog-compliant format and timestampformat is not specified, the output produced will not be RFC3164-compliant.
- \* The format is a strftime-style timestamp formatting string. This is the same implementation used in the 'eval' search command, splunk logging, and other places in splunkd.
  - \* For example: %b %e %H:%M:%S for RFC3164-compliant output
  - \* %b - Abbreviated month name (Jan, Feb, ...)
  - \* %e - Day of month
  - \* %H - Hour
  - \* %M - Minute
  - \* %s - Second
- \* For a more exhaustive list of the formatting specifiers, refer to the online documentation.
- \* Note that the string is not quoted.

```

* Defaults to unset, which means that no timestamp will be inserted into the
  front of events.

dropEventsOnQueueFull = <integer>
* If set to a positive number, wait <integer> seconds before throwing out
  all new events until the output queue has space.
* Setting this to -1 or 0 will cause the output queue to block when it gets
  full, causing further blocking up the processing chain.
* If any target group's queue is blocked, no more data will reach any other
  target group.
* Defaults to -1 (do not drop events).

maxEventSize = <integer>
* If specified, sets the maximum size of an event that splunk will transmit.
* All events exceeding this size will be truncated.
* Defaults to 1024 bytes.

#---- Routing Data to Syslog Server ----
# To route data to syslog server:
# 1) Decide which events to route to which servers.
# 2) Edit the props.conf, transforms.conf, and outputs.conf files on the
#    forwarders.

# Edit $SPLUNK_HOME/etc/system/local/props.conf and set a TRANSFORMS-routing
# attribute as shown here:
#
# [<spec>]
# TRANSFORMS-routing=<unique_stanza_name>

* <spec> can be:
  * <sourcetype>, the source type of an event
  * host::<host>, where <host> is the host for an event
  * source::<source>, where <source> is the source for an event

* Use the <unique_stanza_name> when creating your entry in transforms.conf.

# Edit $SPLUNK_HOME/etc/system/local/transforms.conf and set rules to match your props.conf stanza:
#
# [<unique_stanza_name>]
# REGEX=<your_regex>
# DEST_KEY=_SYSLOG_ROUTING
# FORMAT=<unique_group_name>

* <unique_stanza_name> must match the name you created in props.conf.
* Enter the regex rules in <your_regex> to determine which events get
  conditionally routed.
* DEST_KEY should be set to _SYSLOG_ROUTING to send events via SYSLOG.
* Set FORMAT to <unique_group_name>. This should match the syslog group name
  you create in outputs.conf.

```

## ***IndexAndForward Processor-----***

```

#####
#----IndexAndForward Processor-----
#####IndexAndForward Processor-----
# The IndexAndForward processor determines the default behavior for indexing
# data on full Splunk. It has the "index" property, which determines whether
# indexing occurs.
#
# When Splunk is not configured as a forwarder, "index" is set to "true".
# That is, the Splunk instance indexes data by default.
#
# When Splunk is configured as a forwarder, the processor turns "index" to
# "false". That is, the Splunk instance does not index data by default.
#
# The IndexAndForward processor has no effect on the universal forwarder,
# which can never index data.
#
# If the [tcpout] stanza configures the indexAndForward attribute, the value
# of that attribute overrides the default value of "index". However, if you

```

```

# set "index" in the [indexAndForward] stanza, described below, it
# supersedes any value set in [tcpout].

[indexAndForward]
index = [true|false]
* If set to true, data is indexed.
* If set to false, data is not indexed.
* Default depends on whether the Splunk instance is configured as a
  forwarder, modified by any value configured for the indexAndForward
  attribute in [tcpout].

selectiveIndexing = [true|false]
* When index is 'true', all events are indexed. Setting selectiveIndexing to
  'true' allows you to index only specific events that has key
  '_INDEX_AND_FORWARD_ROUTING' set.
* '_INDEX_AND_FORWARD_ROUTING' can be set in inputs.conf as:
  [<input_stanza>]
  _INDEX_AND_FORWARD_ROUTING = local
* Defaults to false.

[indexer_discovery:<name>]

pass4SymmKey = <password>
* Security key shared between indexer_discovery and forwarders.
* If specified here, the same value must also be specified on the master node identified by master_uri.

send_timeout = <seconds>
* Low-level timeout for sending messages to the master node.
* Fractional seconds are allowed.
* Default is 30.

rcv_timeout = <seconds>
* Low-level timeout for receiving messages from the master node.
* Fractional seconds are allowed.
* Default is 30.

cxn_timeout = <seconds>
* Low-level timeout for connecting to the master node.
* Fractional seconds are allowed.
* Default is 30.

master_uri = <uri>
* URI and management port of the cluster master used in indexer discovery.
* Example: https://SplunkMaster01.example.com:8089

```

## outputs.conf.example

```

# Version 6.5.0
#
# This file contains an example outputs.conf. Use this file to configure
# forwarding in a distributed set up.
#
# To use one or more of these configurations, copy the configuration block into
# outputs.conf in $SPLUNK_HOME/etc/system/local/. You must restart Splunk to
# enable configurations.
#
# To learn more about configuration files (including precedence) please see the
# documentation located at
# http://docs.splunk.com/Documentation/Splunk/latest/Admin/Aboutconfigurationfiles

# Specify a target group for an IP:PORT which consists of a single receiver.
# This is the simplest possible configuration; it sends data to the host at
# 10.1.1.197 on port 9997.

[tcpout:group1]
server=10.1.1.197:9997

# Specify a target group for a hostname which consists of a single receiver.

```

```

[tcpout:group2]
server=myhost.Splunk.com:9997

# Specify a target group made up of two receivers. In this case, the data will
# be distributed using AutoLB between these two receivers. You can specify as
# many receivers as you wish here. You can combine host name and IP if you
# wish.
# NOTE: Do not use this configuration with SplunkLightForwarder.

[tcpout:group3]
server=myhost.Splunk.com:9997,10.1.1.197:6666

# You can override any of the global configuration values on a per-target group
# basis. All target groups that do not override a global config will inherit
# the global config.

# Send every event to a receiver at foo.Splunk.com:9997 with a maximum queue
# size of 100,500 events.

[tcpout:group4]
server=foo.Splunk.com:9997
heartbeatFrequency=45
maxQueueSize=100500

# Send data to a receiving system that controls access by tokens.
# NOTE: token value is encrypted. Encryption is done by REST endpoint while saving.
[tcpout:group4]
server=foo.Splunk.com:9997
token=$1$/fRSBT+2APNAyCB7t1cgOyLnAtqAQFC8NI4TGA2wX4JHfN5d9g==

# Clone events to groups indexer1 and indexer2. Also, index all this data
# locally as well.

[tcpout]
indexAndForward=true

[tcpout:indexer1]
server=Y.Y.Y.Y:9997

[tcpout:indexer2]
server=X.X.X.X:6666

# Clone events between two data balanced groups.

[tcpout:indexer1]
server=A.A.A.A:1111, B.B.B.B:2222

[tcpout:indexer2]
server=C.C.C.C:3333, D.D.D.D:4444

# Syslog output configuration
# This example sends only events generated by the splunk daemon to a remote
# syslog host in syslog-compliant format:

[syslog:syslog-out1]
disabled = false
server = X.X.X.X:9099
type = tcp
priority = <34>
timestampformat = %b %e %H:%M:%S

# New in 4.0: Auto Load Balancing
#
# This example balances output between two indexers running on
# 1.2.3.4:4433 and 1.2.4.5:4433.
# To achieve this you'd create a DNS entry for splunkLB pointing
# to the two IP addresses of your indexers:

```

```

#
# $ORIGIN example.com.
# splunkLB A 1.2.3.4
# splunkLB A 1.2.3.5

[tcput]
defaultGroup = lb

[tcput:lb]
server = splunkLB.example.com:4433
autoLB = true

# Alternatively, you can autoLB sans DNS:

[tcput]
defaultGroup = lb

[tcput:lb]
server = 1.2.3.4:4433, 1.2.3.5:4433
autoLB = true

# Compression
#
# This example sends compressed events to the remote indexer.
# NOTE: Compression can be enabled TCP or SSL outputs only.
# The receiver input port should also have compression enabled.

[tcput]
server = splunkServer.example.com:4433
compressed = true

# SSL
#
# This example sends events to an indexer via SSL using splunk's
# self signed cert:

[tcput]
server = splunkServer.example.com:4433
sslPassword = password
sslCertPath = $SPLUNK_HOME/etc/auth/server.pem
sslRootCAPath = $SPLUNK_HOME/etc/auth/ca.pem

#
# The following example shows how to route events to syslog server
# This is similar to tcpout routing, but DEST_KEY is set to _SYSLOG_ROUTING
#

# 1. Edit $SPLUNK_HOME/etc/system/local/props.conf and set a TRANSFORMS-routing
# attribute:
[default]
TRANSFORMS-routing=errorRouting

[syslog]
TRANSFORMS-routing=syslogRouting

# 2. Edit $SPLUNK_HOME/etc/system/local/transforms.conf and set errorRouting
# and syslogRouting rules:
[errorRouting]
REGEX=error
DEST_KEY=_SYSLOG_ROUTING
FORMAT=errorGroup

[syslogRouting]
REGEX=
DEST_KEY=_SYSLOG_ROUTING
FORMAT=syslogGroup

# 3. Edit $SPLUNK_HOME/etc/system/local/outputs.conf and set which syslog
# outputs go to with servers or groups:
[syslog]

```



```

defaultGroup=everythingElseGroup

[syslog:syslogGroup]
server = 10.1.1.197:9997

[syslog:errorGroup]
server=10.1.1.200:9999

[syslog:everythingElseGroup]
server=10.1.1.250:6666

#
# Perform selective indexing and forwarding
#
# With a heavy forwarder only, you can index and store data locally, as well as
# forward the data onwards to a receiving indexer. There are two ways to do
# this:

# 1. In outputs.conf:
[tcpout]
defaultGroup = indexers

[indexAndForward]
index=true
selectiveIndexing=true

[tcpout:indexers]
server = 10.1.1.197:9997, 10.1.1.200:9997

# 2. In inputs.conf, Add _INDEX_AND_FORWARD_ROUTING for any data that you want
#   index locally, and
#   _TCP_ROUTING=<target_group> for data to be forwarded.

[monitor:///var/log/messages/]
_INDEX_AND_FORWARD_ROUTING=local

[monitor:///var/log/httpd/]
_TCP_ROUTING=indexers

```

## passwords.conf

以下为 passwords.conf 的规范和示例文件。

### passwords.conf.spec

```

#   Version 6.5.0
#
# This file maintains the credential information for a given app in Splunk Enterprise.
#
# There is no global, default passwords.conf. Instead, anytime a user creates
# a new user or edit a user onwards hitting the storage endpoint
# will create this passwords.conf file which gets replicated
# in a search head clustering environment.
# Note that passwords.conf is only created from 6.3.0 release.
#
# You must restart Splunk Enterprise to reload manual changes to passwords.conf.
#
# To learn more about configuration files (including precedence) please see the
# documentation located at
# http://docs.splunk.com/Documentation/Splunk/latest/Admin/Aboutconfigurationfiles
# More details for storage endpoint is at
# http://blogs.splunk.com/2011/03/15/storing-encrypted-credentials/

```

**[credential:<realm>:<username>:]**

```

[credential:<realm>:<username>:]
password = <password>

```

- \* Password that corresponds to the given username for the given realm.  
Note that realm is optional
- \* The password can be in clear text, however when saved from splunkd the password will always be encrypted

## passwords.conf.example

```
# Version 6.5.0
#
# The following are example passwords.conf configurations. Configure properties for
# your custom application.
#
# There is NO DEFAULT passwords.conf. The file only gets created once you add/edit
# a credential information via the storage endpoint as follows.
#
# The POST request to add user1 credentials to the storage/password endpoint
# curl -k -u admin:changeme https://localhost:8089/servicesNS/nobody/search/storage/passwords -d name=user1 -d
password=changeme2
#
# The GET request to list all the credentials stored at the storage/passwords endpoint
# curl -k -u admin:changeme https://localhost:8089/services/storage/passwords
#
# To use one or more of these configurations, copy the configuration block into
# passwords.conf in $SPLUNK_HOME/etc/<apps>/local/. You must restart Splunk to
# enable configurations.
#
# To learn more about configuration files (including precedence) please see the
# documentation located at
# http://docs.splunk.com/Documentation/Splunk/latest/Admin/Aboutconfigurationfiles
#

[credential::testuser:]
password = changeme
```

## pdf\_server.conf

以下为 pdf\_server.conf 的规范和示例文件。

### pdf\_server.conf.spec

```
# Version 6.1
#
# This file contains possible attributes and values you can use to configure Splunk's pdf server.
#
# There is a pdf_server.conf in $SPLUNK_HOME/etc/system/default/. To set custom configurations,
# place a pdf_server.conf in $SPLUNK_HOME/etc/system/local/. For examples, see pdf_server.conf.example.
# You must restart the pdf server to enable configurations.
#
# To learn more about configuration files (including precedence) please see the documentation
# located at http://docs.splunk.com/Documentation/Splunk/latest/Admin/Aboutconfigurationfiles

# GLOBAL SETTINGS
# Use the [default] stanza to define any global settings.
#
# * You can also define global settings outside of any stanza, at the top of the file.
#
# * Each conf file should have at most one default stanza. If there are multiple default
#
# stanzas, attributes are combined. In the case of multiple definitions of the same
#
# attribute, the last definition in the file wins.
#
# * If an attribute is defined at both the global level and in a specific stanza, the
#
# value in the specific stanza takes precedence.

[settings]
    * Set general Splunk Web configuration options under this stanza name.
    * Follow this stanza name with any number of the following attribute/value pairs.
    * If you do not specify an entry for each attribute, Splunk will use the default value.

startwebserver = [0|1]
    * Set whether or not to start the server.
```

```

        * 0 disables Splunk Web, 1 enables it.
        * Defaults to 1.

httpport = <port_number>
    * Must be present for the server to start.
    * If omitted or 0 the server will NOT start an http listener.
    * If using SSL, set to the HTTPS port number.
    * Defaults to 9000.

enableSplunkWebSSL = [True|False]
    * Toggle between http or https.
    * Set to true to enable https and SSL.
    * Defaults to False.

privKeyPath = /certs/privkey.pem
caCertPath = /certs/cert.pem
    * Specify paths and names for Web SSL certs.
    * Path is relative to $SPLUNK_HOME/share/splunk.

supportSSLV3Only = [True|False]
    * Allow only SSLv3 connections if true.
    * NOTE: Enabling this may cause some browsers problems.

root_endpoint = <URI_prefix_string>
    * Defines the root URI path on which the appserver will listen.
    * Default setting is '/'.
    * For example: if you want to proxy the splunk UI at http://splunk:8000/splunkui, then set root_endpoint =
/splunkui

static_endpoint = <URI_prefix_string>
    * Path to static content.
    * The path here is automatically appended to root_endpoint defined above.
    * Default is /static.

static_dir = <relative_filesystem_path>
    * The directory that actually holds the static content.
    * This can be an absolute URL if you want to put it elsewhere.
    * Default is share/splunk/search_mrsparkle/exposed.

enable_gzip = [True|False]
    * Determines if web server applies gzip compression to responses.
    * Defaults to True.

#
# cherrypy HTTP server config
#

server.thread_pool = <integer>
    * Specifies the numbers of threads the app server is allowed to maintain.
    * Defaults to 10.

server.socket_host = <ip_address>
    * Host values may be any IPv4 or IPv6 address, or any valid hostname.
    * The string 'localhost' is a synonym for '127.0.0.1' (or '::1', if your hosts file prefers IPv6).
    * The string '0.0.0.0' is a special IPv4 entry meaning "any active interface" (INADDR_ANY), and
    * ':::' is the similar IN6ADDR_ANY for IPv6.
    * The empty string or None are not allowed.
    * Defaults to 0.0.0.0

log.access_file = <filename>
    * Specifies the HTTP access log filename.
    * Stored in default Splunk /var/log directory.
    * Defaults to pdf_access.log

log.error_file = <filename>
    * Specifies the HTTP error log filename.
    * Stored in default Splunk /var/log directory.
    * Defaults to pdf_service.log

log.screen = [True|False]

```

```

    * Indicates if runtime output is displayed inside an interactive tty.
    * Defaults to True

request.show_tracebacks = [True|False]
    * Indicates if an exception traceback is displayed to the user on fatal exceptions.
    * Defaults to True

engine.autoreload_on = [True|False]
    * Indicates if the app server will auto-restart if it detects a python file has changed.
    * Defaults to False

tools.sessions.on = True
    * Indicates if user session support is enabled.
    * Should always be True

tools.sessions.timeout = <integer>
    * Specifies the number of minutes of inactivity before a user session expires.
    * Defaults to 60

response.timeout = <integer>
    * Specifies the number of seconds to wait for the server to complete a response.
    * Some requests such as uploading large files can take a long time.
    * Defaults to 7200

tools.sessions.storage_type = [file]
tools.sessions.storage_path = <filepath>
    * Specifies the session information storage mechanisms.
    * Comment out these two lines to use RAM based sessions instead.
    * Use an absolute path to store sessions outside of the Splunk directory tree.
    * Defaults to storage_type=file, storage_path=var/run/splunk

tools.decode.on = [True|False]
    * Indicates if all strings that come into CherryPy controller methods are decoded as unicode (assumes UTF-8
encoding).
    * WARNING: Disabling this will likely break the application, as all incoming strings are assumed
to be unicode.
    * Defaults to True

tools.encode.on = [True|False]
    * Encodes all controller method response strings into UTF-8 str objects in Python.
    * WARNING: Disabling this will likely cause high byte character encoding to fail.
    * Defaults to True

tools.encode.encoding = <codec>
    * Force all outgoing characters to be encoded into UTF-8.
    * This only works with tools.encode.on set to True.
    * By setting this to utf-8, CherryPy's default behavior of observing the Accept-Charset header
is overwritten and forces utf-8 output. Only change this if you know a particular browser
installation must receive some other character encoding (Latin-1, iso-8859-1, etc.).
    * WARNING: Change this at your own risk.
    * Defaults to utf-8

pid_path = <filepath>
    * Specifies the path to the PID file.
    * Defaults to var/run/splunk/splunkweb.pid.

firefox_cmdline = <cmdline>
    * Specifies additional arguments to pass to Firefox.
    * This should normally not be set.

max_queue = <integer>
    * Specifies the maximum size of the backlog of pending report requests.
    * Once the backlog is reached the server will return an error on receiving additional requests.
    * Defaults to 10.

max_concurrent = <integer>
    * Specifies the maximum number of copies of Firefox that the report server will use concurrently to render
reports.
    * Increase only if the host machine has multiple cores and plenty of spare memory.
    * Defaults to 2.

Xvfb = <path>

```

```

* Pathname to the Xvfb program.
* Defaults to searching the PATH.

xauth = <path>
* Pathname to the xauth program.
* Defaults to searching the PATH.

mcookie = <path>
* Pathname to the mcookie program.
* Defaults to searching the PATH.

appserver_ipaddr = <ip_networks>
* If set, the PDF server will only query Splunk app servers on IP addresses within the IP networks
  specified here.
* Networks can be specified as a prefix (10.1.0.0/16) or using a netmask (10.1.0.0/255.255.0.0).
* IPv6 addresses are also supported.
* Individual IP addresses can also be listed (1.2.3.4).
* Multiple networks should be comma separated.
* Defaults to accepting any IP address.

client_ipaddr = <ip_networks>
* If set, the PDF server will only accept requests from hosts whose IP address falls within the IP
  networks specified here.
* Generally this setting should match the appserver_ipaddr setting.
* Format matches appserver_ipaddr.
* Defaults to accepting any IP address.

screenshot_enabled = [True|False]
* If enabled allows screenshots of the X server to be taken for debugging purposes.
* Enabling this is a potential security hole as anyone on an IP address matching client_ipaddr will be
  able to see reports in progress.
* Defaults to False.

```

## pdf\_server.conf.example

```

# Version 6.1
#
# This is an example pdf_server.conf. Use this file to configure pdf server process settings.
#
# To use one or more of these configurations, copy the configuration block into pdf_server.conf
# in $SPLUNK_HOME/etc/system/local/. You must restart the pdf server to enable configurations.
#
# To learn more about configuration files (including precedence) please see the documentation
# located at http://docs.splunk.com/Documentation/Splunk/latest/Admin/Aboutconfigurationfiles

# This stanza heading must precede any changes.
[settings]

# Change the default port number:
httpport = 12900

# Lock down access to the IP address of specific appservers
# that will utilize the pdf server
appserver_ipaddr = 192.168.3.0/24,192.168.2.2
client_ipaddr = 192.168.3.0/24,192.168.2.2

```

## procmon-filters.conf

以下为 procmon-filters.conf 的规范和示例文件。

### procmon-filters.conf.spec

```

# Version 6.5.0

```

```
#
# *** DEPRECATED ***
#
#
# This file contains potential attribute/value pairs to use when configuring
# Windows registry monitoring. The procmon-filters.conf file contains the
# regular expressions you create to refine and filter the processes you want
# Splunk to monitor. You must restart Splunk to enable configurations.
#
# To learn more about configuration files (including precedence) please see the
# documentation located at
# http://docs.splunk.com/Documentation/Splunk/latest/Admin/Aboutconfigurationfiles

#### find out if this file is still being used.
```

## [<stanza name>]

```
[<stanza name>]
* Name of the filter being defined.

proc = <string>
* Regex specifying process image that you want Splunk to monitor.

type = <string>
* Regex specifying the type(s) of process event that you want Splunk to
  monitor.

hive = <string>
* Not used in this context, but should always have value ".*"
```

## procmon-filters.conf.example

```
# Version 6.5.0
#
# This file contains example registry monitor filters. To create your own
# filter, use the information in procmon-filters.conf.spec.
#
# To use one or more of these configurations, copy the configuration block into
# procmon-filters.conf in $SPLUNK_HOME/etc/system/local/. You must restart
# Splunk to enable configurations.
#
# To learn more about configuration files (including precedence) please see the
# documentation located at
# http://docs.splunk.com/Documentation/Splunk/latest/Admin/Aboutconfigurationfiles

[default]
hive = .*

[not-splunk-optimize]
proc = (?<!splunk-optimize.exe)$
type = create|exit|image
```

## props.conf

以下为 props.conf 的规范和示例文件。

## props.conf.spec

```
#
# This file contains possible attribute/value pairs for configuring Splunk's
# processing properties via props.conf.
#
# Props.conf is commonly used for:
#
# * Configuring linebreaking for multiline events.
# * Setting up character set encoding.
```

```

# * Allowing processing of binary files.
# * Configuring timestamp recognition.
# * Configuring event segmentation.
# * Overriding Splunk's automated host and source type matching. You can use
#   props.conf to:
#   * Configure advanced (regex-based) host and source type overrides.
#   * Override source type matching for data from a particular source.
#   * Set up rule-based source type recognition.
#   * Rename source types.
# * Anonymizing certain types of sensitive incoming data, such as credit
#   card or social security numbers, using sed scripts.
# * Routing specific events to a particular index, when you have multiple
#   indexes.
# * Creating new index-time field extractions, including header-based field
#   extractions.
#   NOTE: We do not recommend adding to the set of fields that are extracted
#         at index time unless it is absolutely necessary because there are
#         negative performance implications.
# * Defining new search-time field extractions. You can define basic
#   search-time field extractions entirely through props.conf. But a
#   transforms.conf component is required if you need to create search-time
#   field extractions that involve one or more of the following:
#   * Reuse of the same field-extracting regular expression across
#     multiple sources, source types, or hosts.
#   * Application of more than one regex to the same source, source type,
#     or host.
#   * Delimiter-based field extractions (they involve field-value pairs
#     that are separated by commas, colons, semicolons, bars, or
#     something similar).
#   * Extraction of multiple values for the same field (multivalued
#     field extraction).
#   * Extraction of fields with names that begin with numbers or
#     underscores.
# * Setting up lookup tables that look up fields from external sources.
# * Creating field aliases.
#
# NOTE: Several of the above actions involve a corresponding transforms.conf
# configuration.
#
# You can find more information on these topics by searching the Splunk
# documentation (http://docs.splunk.com/Documentation/Splunk).
#
# There is a props.conf in $SPLUNK_HOME/etc/system/default/. To set custom
# configurations, place a props.conf in $SPLUNK_HOME/etc/system/local/. For
# help, see props.conf.example.
#
# You can enable configurations changes made to props.conf by typing the
# following search string in Splunk Web:
#
# | extract reload=T
#
# To learn more about configuration files (including precedence) please see
# the documentation located at
# http://docs.splunk.com/Documentation/Splunk/latest/Admin/Aboutconfigurationfiles
#
# For more information about using props.conf in conjunction with
# distributed Splunk deployments, see the Distributed Deployment Manual.

```

## 全局设置

```

# GLOBAL SETTINGS
# Use the [default] stanza to define any global settings.
# * You can also define global settings outside of any stanza, at the top
#   of the file.
# * Each conf file should have at most one default stanza. If there are
#   multiple default stanzas, attributes are combined. In the case of
#   multiple definitions of the same attribute, the last definition in the
#   file wins.
# * If an attribute is defined at both the global level and in a specific
#   stanza, the value in the specific stanza takes precedence.

```

[<spec>]

- \* This stanza enables properties for a given <spec>.
- \* A props.conf file can contain multiple stanzas for any number of different <spec>.
- \* Follow this stanza name with any number of the following attribute/value pairs, as appropriate for what you want to do.
- \* If you do not set an attribute for a given <spec>, the default is used.

<spec> can be:

1. <sourcetype>, the source type of an event.
2. host::<host>, where <host> is the host, or host-matching pattern, for an event.
3. source::<source>, where <source> is the source, or source-matching pattern, for an event.
4. rule::<rule>, where <rule> is a unique name of a source type classification rule.
5. delayedrule::<rule>, where <rule> is a unique name of a delayed source type classification rule.

These are only considered as a last resort before generating a new source type based on the source seen.

**[<spec>] stanza precedence:**

For settings that are specified in multiple categories of matching [<spec>] stanzas, [host::<host>] settings override [<sourcetype>] settings. Additionally, [source::<source>] settings override both [host::<host>] and [<sourcetype>] settings.

**Considerations for Windows file paths:**

When you specify Windows-based file paths as part of a [source::<source>] stanza, you must escape any backslashes contained within the specified file path.

Example: [source::c:\\path\_to\\file.txt]

**[<spec>] stanza patterns:**

When setting a [<spec>] stanza, you can use the following regex-type syntax:

- ... recurses through directories until the match is met or equivalently, matches any number of characters.
- \* matches anything but the path separator 0 or more times. The path separator is '/' on unix, or '\\' on windows. Intended to match a partial or complete directory or filename.
- | is equivalent to 'or'
- ( ) are used to limit scope of |.
- \\ = matches a literal backslash '\\.

Example: [source::....(?!(tar.)(gz|bz2)]

This matches any file ending with '.gz' or '.bz2', provided this is not preceded by 'tar.', so tar.bz2 and tar.gz would not be matched.

**[source::<source>] and [host::<host>] stanza match language:**

Match expressions must match the entire name, not just a substring. If you are familiar with regular expressions, match expressions are based on a full implementation of PCRE with the translation of ..., \* and . Thus . matches a period, \* matches non-directory separators, and ... matches any number of any characters.

For more information see the wildcards section at:  
<http://docs.splunk.com/Documentation/Splunk/latest/Data/Specifyinputpathswithwildcards>

**[<spec>] stanza pattern collisions:**

Suppose the source of a given input matches multiple [source::<source>] patterns. If the [<spec>] stanzas for these patterns each supply distinct settings, Splunk applies all of these settings.



However, suppose two [<spec>] stanzas supply the same setting. In this case, Splunk chooses the value to apply based on the ASCII order of the patterns in question.

For example, take this source:

```
source::az
```

and the following colliding patterns:

```
[source:...a...]  
sourcetype = a
```

```
[source:...z...]  
sourcetype = z
```

In this case, the settings provided by the pattern [source:...a...] take precedence over those provided by [source:...z...], and sourcetype ends up with "a" as its value.

To override this default ASCII ordering, use the priority key:

```
[source:...a...]  
sourcetype = a  
priority = 5
```

```
[source:...z...]  
sourcetype = z  
priority = 10
```

Assigning a higher priority to the second stanza causes sourcetype to have the value "z".

**\*\*Case-sensitivity for [<spec>] stanza matching:\*\***

By default, [source::<source>] and [sourcetype] stanzas match in a case-sensitive manner, while [host::<host>] stanzas match in a case-insensitive manner. This is a convenient default, given that DNS names are case-insensitive.

To force a [host::<host>] stanza to match in a case-sensitive manner use the "(?-i)" option in its pattern.

For example:

```
[host::foo]  
FIELDALIAS-a = a AS one
```

```
[host::(?-i)bar]  
FIELDALIAS-b = b AS two
```

The first stanza will actually apply to events with host values of "FOO" or "Foo" . The second stanza, on the other hand, will not apply to events with host values of "BAR" or "Bar".

**\*\*Building the final [<spec>] stanza:\*\***

The final [<spec>] stanza is built by layering together (1) literal-matching stanzas (stanzas which match the string literally) and (2) any regex-matching stanzas, according to the value of the priority field.

If not specified, the default value of the priority key is:

- \* 0 for pattern-matching stanzas.
- \* 100 for literal-matching stanzas.

NOTE: Setting the priority key to a value greater than 100 causes the pattern-matched [<spec>] stanzas to override the values of the literal-matching [<spec>] stanzas.

The priority key can also be used to resolve collisions between [sourcetype] patterns and [host::<host>] patterns. However, be aware that the priority key does *not* affect precedence across <spec> types. For

example, [<spec>] stanzas with [source::<source>] patterns take priority over stanzas with [host::<host>] and [<sourcetype>] patterns, regardless of their respective priority key values.

```
#*****
# The possible attributes/value pairs for props.conf, and their
# default values, are:
#*****

# International characters and character encoding.

CHARSET = <string>
* When set, Splunk assumes the input from the given [<spec>] is in the
  specified encoding.
* Can only be used as the basis of [<sourcetype>] or [source::<spec>],
  not [host::<spec>].
* A list of valid encodings can be retrieved using the command "iconv -l" on
  most *nix systems.
* If an invalid encoding is specified, a warning is logged during initial
  configuration and further input from that [<spec>] is discarded.
* If the source encoding is valid, but some characters from the [<spec>] are
  not valid in the specified encoding, then the characters are escaped as
  hex (for example, "\xF3").
* When set to "AUTO", Splunk attempts to automatically determine the character encoding and
  convert text from that encoding to UTF-8.
* For a complete list of the character sets Splunk automatically detects,
  see the online documentation.
* This setting applies at input time, when data is first read by Splunk.
  The setting is used on a Splunk system that has configured inputs
  acquiring the data.
* Defaults to ASCII.
```

## 换行

```
#*****
# Line breaking
#*****Line breaking

# Use the following attributes to define the length of a line.

TRUNCATE = <non-negative integer>
* Change the default maximum line length (in bytes).
* Although this is in bytes, line length is rounded down when this would
  otherwise land mid-character for multi-byte characters.
* Set to 0 if you never want truncation (very long lines are, however, often
  a sign of garbage data).
* Defaults to 10000 bytes.

LINE_BREAKER = <regular expression>
* Specifies a regex that determines how the raw text stream is broken into
  initial events, before line merging takes place. (See the SHOULD_LINEMERGE
  attribute, below)
* Defaults to ({\r\n}+), meaning data is broken into an event for each line,
  delimited by any number of carriage return or newline characters.
* The regex must contain a capturing group -- a pair of parentheses which
  defines an identified subcomponent of the match.
* Wherever the regex matches, Splunk considers the start of the first
  capturing group to be the end of the previous event, and considers the end
  of the first capturing group to be the start of the next event.
* The contents of the first capturing group are discarded, and will not be
  present in any event. You are telling Splunk that this text comes between
  lines.
* NOTE: You get a significant boost to processing speed when you use
  LINE_BREAKER to delimit multiline events (as opposed to using
  SHOULD_LINEMERGE to reassemble individual lines into multiline events).
  * When using LINE_BREAKER to delimit events, SHOULD_LINEMERGE should be set
    to false, to ensure no further combination of delimited events occurs.
  * Using LINE_BREAKER to delimit events is discussed in more detail in the web
```

documentation at the following url:  
<http://docs.splunk.com/Documentation/Splunk/latest/Data/indexmulti-lineevents>

**\*\* Special considerations for LINE\_BREAKER with branched expressions \*\***

When using LINE\_BREAKER with completely independent patterns separated by pipes, some special issues come into play.

EG. LINE\_BREAKER = pattern1|pattern2|pattern3

Note, this is not about all forms of alternation, eg there is nothing particular special about

example: LINE\_BREAKER = ([\r\n])+ (one|two|three)

where the top level remains a single expression.

A caution: Relying on these rules is NOT encouraged. Simpler is better, in both regular expressions and the complexity of the behavior they rely on. If possible, it is strongly recommended that you reconstruct your regex to have a leftmost capturing group that always matches.

It may be useful to use non-capturing groups if you need to express a group before the text to discard.

EG. LINE\_BREAKER = (?:one|two)([\r\n]+)

- \* This will match the text one, or two, followed by any amount of newlines or carriage returns. The one-or-two group is non-capturing via the ?: prefix and will be skipped by LINE\_BREAKER.

\* A branched expression can match without the first capturing group matching, so the line breaker behavior becomes more complex.

Rules:

- 1: If the first capturing group is part of a match, it is considered the linebreak, as normal.
- 2: If the first capturing group is not part of a match, the leftmost capturing group which is part of a match will be considered the linebreak.
- 3: If no capturing group is part of the match, the linebreaker will assume that the linebreak is a zero-length break immediately preceding the match.

Example 1: LINE\_BREAKER = end(\n)begin|end2(\n)begin2|begin3

- \* A line ending with 'end' followed a line beginning with 'begin' would match the first branch, and the first capturing group would have a match according to rule 1. That particular newline would become a break between lines.
- \* A line ending with 'end2' followed by a line beginning with 'begin2' would match the second branch and the second capturing group would have a match. That second capturing group would become the linebreak according to rule 2, and the associated newline would become a break between lines.
- \* The text 'begin3' anywhere in the file at all would match the third branch, and there would be no capturing group with a match. A linebreak would be assumed immediately prior to the text 'begin3' so a linebreak would be inserted prior to this text in accordance with rule 3. This means that a linebreak will occur before the text 'begin3' at any point in the text, whether a linebreak character exists or not.

Example 2: Example 1 would probably be better written as follows. This is not equivalent for all possible files, but for most real files would be equivalent.

LINE\_BREAKER = end2?(\n)begin(2|3)?

LINE\_BREAKER\_LOOKBEHIND = <integer>

- \* When there is leftover data from a previous raw chunk, LINE\_BREAKER\_LOOKBEHIND indicates the number of bytes before the end of the raw chunk (with the next chunk concatenated) that Splunk applies the LINE\_BREAKER regex. You may want to increase this value from its default if you are dealing with especially large or multiline events.
- \* Defaults to 100 (bytes).

# Use the following attributes to specify how multiline events are handled.

SHOULD\_LINEMERGE = [true|false]

- \* When set to true, Splunk combines several lines of data into a single

```

multiline event, based on the following configuration attributes.
* Defaults to true.

# When SHOULD_LINEMERGE is set to true, use the following attributes to
# define how Splunk builds multiline events.

BREAK_ONLY_BEFORE_DATE = [true|false]
* When set to true, Splunk creates a new event only if it encounters a new
line with a date.
* Note, when using DATETIME_CONFIG = CURRENT or NONE, this setting is not
meaningful, as timestamps are not identified.
* Defaults to true.

BREAK_ONLY_BEFORE = <regular expression>
* When set, Splunk creates a new event only if it encounters a new line that
matches the regular expression.
* Defaults to empty.

MUST_BREAK_AFTER = <regular expression>
* When set and the regular expression matches the current line, Splunk
creates a new event for the next input line.
* Splunk may still break before the current line if another rule matches.
* Defaults to empty.

MUST_NOT_BREAK_AFTER = <regular expression>
* When set and the current line matches the regular expression, Splunk does
not break on any subsequent lines until the MUST_BREAK_AFTER expression
matches.
* Defaults to empty.

MUST_NOT_BREAK_BEFORE = <regular expression>
* When set and the current line matches the regular expression, Splunk does
not break the last event before the current line.
* Defaults to empty.

MAX_EVENTS = <integer>
* Specifies the maximum number of input lines to add to any event.
* Splunk breaks after the specified number of lines are read.
* Defaults to 256 (lines).

# Use the following attributes to handle better load balancing from UF.
# Please note the EVENT_BREAKER properties are applicable for Splunk Universal
# Forwarder instances only.

EVENT_BREAKER_ENABLE = [true|false]
* When set to true, Splunk will split incoming data with a light-weight
chunked line breaking processor so that data is distributed fairly evenly
amongst multiple indexers. Use this setting on the UF to indicate that
data should be split on event boundaries across indexers especially
for large files.
* Defaults to false

# Use the following to define event boundaries for multi-line events
# For single-line events, the default settings should suffice

EVENT_BREAKER = <regular expression>
* When set, Splunk will use the setting to define an event boundary at the
end of the first matching group instance.

```

## 时间戳提取配置

```

*****
# Timestamp extraction configuration
*****Timestamp extraction configuration

DATETIME_CONFIG = <filename relative to $SPLUNK_HOME>
* Specifies which file configures the timestamp extractor, which identifies
timestamps from the event text.
* This configuration may also be set to "NONE" to prevent the timestamp
extractor from running or "CURRENT" to assign the current system time to

```

each event.

- \* "CURRENT" will set the time of the event to the time that the event was merged from lines, or worded differently, the time it passed through the aggregator processor.
- \* "NONE" will leave the event time set to whatever time was selected by the input layer
- \* For data sent by splunk forwarders over the splunk protocol, the input layer will be the time that was selected on the forwarder by its input behavior (as below).
- \* For file-based inputs (monitor, batch) the time chosen will be the modification timestamp on the file being read.
- \* For other inputs, the time chosen will be the current system time when the event is read from the pipe/socket/etc.
- \* Both "CURRENT" and "NONE" explicitly disable the per-text timestamp identification, so the default event boundary detection (BREAK\_ONLY\_BEFORE\_DATE = true) is likely to not work as desired. When using these settings, use SHOULD\_LINEMERGE and/or the BREAK\_ONLY\_\* , MUST\_BREAK\_\* settings to control event merging.
- \* Defaults to /etc/datetime.xml (for example, \$SPLUNK\_HOME/etc/datetime.xml).

TIME\_PREFIX = <regular expression>

- \* If set, splunk scans the event text for a match for this regex in event text before attempting to extract a timestamp.
- \* The timestamping algorithm only looks for a timestamp in the text following the end of the first regex match.
- \* For example, if TIME\_PREFIX is set to "abc123", only text following the first occurrence of the text abc123 will be used for timestamp extraction.
- \* If the TIME\_PREFIX cannot be found in the event text, timestamp extraction will not occur.
- \* Defaults to empty.

MAX\_TIMESTAMP\_LOOKAHEAD = <integer>

- \* Specifies how far (in characters) into an event Splunk should look for a timestamp.
- \* This constraint to timestamp extraction is applied from the point of the TIME\_PREFIX-set location.
- \* For example, if TIME\_PREFIX positions a location 11 characters into the event, and MAX\_TIMESTAMP\_LOOKAHEAD is set to 10, timestamp extraction will be constrained to characters 11 through 20.
- \* If set to 0, or -1, the length constraint for timestamp recognition is effectively disabled. This can have negative performance implications which scale with the length of input lines (or with event size when LINE\_BREAKER is redefined for event splitting).
- \* Defaults to 150 (characters).

TIME\_FORMAT = <strptime-style format>

- \* Specifies a strptime format string to extract the date.
- \* strptime is an industry standard for designating time formats.
- \* For more information on strptime, see "Configure timestamp recognition" in the online documentation.
- \* TIME\_FORMAT starts reading after the TIME\_PREFIX. If both are specified, the TIME\_PREFIX regex must match up to and including the character before the TIME\_FORMAT date.
- \* For good results, the <strptime-style format> should describe the day of the year and the time of day.
- \* Defaults to empty.

TZ = <timezone identifier>

- \* The algorithm for determining the time zone for a particular event is as follows:
- \* If the event has a timezone in its raw text (for example, UTC, -08:00), use that.
- \* If TZ is set to a valid timezone string, use that.
- \* If the event was forwarded, and the forwarder-indexer connection is using the 6.0+ forwarding protocol, use the timezone provided by the forwarder.
- \* Otherwise, use the timezone of the system that is running splunkd.
- \* Defaults to empty.

TZ\_ALIAS = <key=value>[,<key=value>]...

- \* Provides splunk admin-level control over how timezone strings extracted from events are interpreted.
- \* For example, EST can mean Eastern (US) Standard time, or Eastern

(Australian) Standard time. There are many other three letter timezone acronyms with many expansions.

- \* There is no requirement to use TZ\_ALIAS if the traditional Splunk default mappings for these values have been as expected. For example, EST maps to the Eastern US by default.
- \* Has no effect on TZ value; this only affects timezone strings from event text, either from any configured TIME\_FORMAT, or from pattern-based guess fallback.
- \* The setting is a list of key=value pairs, separated by commas.
  - \* The key is matched against the text of the timezone specifier of the event, and the value is the timezone specifier to use when mapping the timestamp to UTC/GMT.
  - \* The value is another TZ specifier which expresses the desired offset.
  - \* Example: TZ\_ALIAS = EST=GMT+10:00 (See props.conf.example for more/full examples)
- \* Defaults to unset.

MAX\_DAYS\_AGO = <integer>

- \* Specifies the maximum number of days in the past, from the current date as provided by input layer (For e.g. forwarder current time, or modtime for files), that an extracted date can be valid. Splunk still indexes events with dates older than MAX\_DAYS\_AGO with the timestamp of the last acceptable event. If no such acceptable event exists, new events with timestamps older than MAX\_DAYS\_AGO will use the current timestamp.
- \* For example, if MAX\_DAYS\_AGO = 10, Splunk applies the timestamp of the last acceptable event to events with extracted timestamps older than 10 days in the past. If no acceptable event exists, Splunk applies the current timestamp.
- \* Defaults to 2000 (days), maximum 10951.
- \* IMPORTANT: If your data is older than 2000 days, increase this setting.

MAX\_DAYS\_HENCE = <integer>

- \* Specifies the maximum number of days in the future, from the current date as provided by input layer (For e.g. forwarder current time, or modtime for files), that an extracted date can be valid. Splunk still indexes events with dates more than MAX\_DAYS\_HENCE in the future with the timestamp of the last acceptable event. If no such acceptable event exists, new events with timestamps after MAX\_DAYS\_HENCE will use the current timestamp.
- \* For example, if MAX\_DAYS\_HENCE = 3, Splunk applies the timestamp of the last acceptable event to events with extracted timestamps more than 3 days in the future. If no acceptable event exists, Splunk applies the current timestamp.
- \* The default value includes dates from one day in the future.
- \* If your servers have the wrong date set or are in a timezone that is one day ahead, increase this value to at least 3.
- \* Defaults to 2 (days), maximum 10950.
- \* IMPORTANT: False positives are less likely with a tighter window, change with caution.

MAX\_DIFF\_SECS\_AGO = <integer>

- \* This setting prevents Splunk Enterprise from rejecting events with timestamps that are out of order.
- \* Do not use this setting to filter events because Splunk Enterprise uses complicated heuristics for time parsing.
- \* Splunk Enterprise warns you if an event timestamp is more than <integer> seconds BEFORE the previous timestamp and does not have the same time format as the majority of timestamps from the source.
- \* After Splunk Enterprise throws the warning, it only rejects an event if it cannot apply a timestamp to the event (for example, if Splunk cannot recognize the time of the event.)
- \* IMPORTANT: If your timestamps are wildly out of order, consider increasing this value.
- \* Note: if the events contain time but not date (date determined another way, such as from a filename) this check will only consider the hour. (No one second granularity for this purpose.)
- \* Defaults to 3600 (one hour), maximum 2147483646.

MAX\_DIFF\_SECS\_HENCE = <integer>

- \* This setting prevents Splunk Enterprise from rejecting events with timestamps that are out of order.
- \* Do not use this setting to filter events because Splunk Enterprise uses complicated heuristics for time parsing.
- \* Splunk Enterprise warns you if an event timestamp is more than <integer> seconds AFTER the previous timestamp and does not have the same time format

as the majority of timestamps from the source.

- \* After Splunk Enterprise throws the warning, it only rejects an event if it cannot apply a timestamp to the event (for example, if Splunk cannot recognize the time of the event.)
- \* IMPORTANT: If your timestamps are wildly out of order, or you have logs that are written less than once a week, consider increasing this value.
- \* Defaults to 604800 (one week), maximum 2147483646.

## 结构化数据头提取和配置

```
*****
# Structured Data Header Extraction and configuration
*****Structured Data Header Extraction and
configuration

* This feature and all of its settings apply at input time, when data is
  first read by Splunk. The setting is used on a Splunk system that has
  configured inputs acquiring the data.

# Special characters for Structured Data Header Extraction:
# Some unprintable characters can be described with escape sequences. The
# attributes that can use these characters specifically mention that
# capability in their descriptions below.
# \f : form feed      byte: 0x0c
# \s : space          byte: 0x20
# \t : horizontal tab byte: 0x09
# \v : vertical tab   byte: 0x0b

INDEXED_EXTRACTIONS = < CSV|W3C|TSV|PSV|JSON >
* Tells Splunk the type of file and the extraction and/or parsing method
  Splunk should use on the file.
  CSV - Comma separated value format
  TSV - Tab-separated value format
  PSV - pipe "|" separated value format
  W3C - W3C Extended Extended Log File Format
  JSON - JavaScript Object Notation format
* These settings default the values of the remaining settings to the
  appropriate values for these known formats.
* Defaults to unset.

PREAMBLE_REGEX = <regex>
* Some files contain preamble lines. This attribute specifies a regular
  expression which allows Splunk to ignore these preamble lines, based on
  the pattern specified.

FIELD_HEADER_REGEX = <regex>
* A regular expression that specifies a pattern for prefixed headers. Note
  that the actual header starts after the pattern and it is not included in
  the header field.
* This attribute supports the use of the special characters described above.

HEADER_FIELD_LINE_NUMBER = <integer>
* Tells Splunk the line number of the line within the file that contains the
  header fields. If set to 0, Splunk attempts to locate the header fields
  within the file automatically.
* The default value is set to 0.

FIELD_DELIMITER = <character>
* Tells Splunk which character delimits or separates fields in the specified
  file or source.
* This attribute supports the use of the special characters described above.

HEADER_FIELD_DELIMITER = <character>
* Tells Splunk which character delimits or separates header fields in the
  specified file or source.
* This attribute supports the use of the special characters described above.

FIELD_QUOTE = <character>
* Tells Splunk the character to use for quotes in the specified file or
```

```

    source.
* This attribute supports the use of the special characters described above.

HEADER_FIELD_QUOTE = <character>
* Specifies Splunk the character to use for quotes in the header of the
  specified file or source.
* This attribute supports the use of the special characters described above.

TIMESTAMP_FIELDS = [ <string>,..., <string>]
* Some CSV and structured files have their timestamp encompass multiple
  fields in the event separated by delimiters. This attribute tells Splunk to
  specify all such fields which constitute the timestamp in a
  comma-separated fashion.
* If not specified, Splunk tries to automatically extract the timestamp of
  the event.

FIELD_NAMES = [ <string>,..., <string>]
* Some CSV and structured files might have missing headers. This attribute
  tells Splunk to specify the header field names directly.

MISSING_VALUE_REGEX = <regex>
* Tells Splunk the placeholder to use in events where no value is present.

JSON_TRIM_BRACES_IN_ARRAY_NAMES = <bool>
* Tell the json parser not to add the curly braces to array names.
* Note that enabling this will make json indextime extracted array fields names
  inconsistent with spath search processor's naming convention.
* For a json document containing the following array object, with trimming
  enabled a indextime field 'mount_point' will be generated instead of the
  spath consistent field 'mount_point{}'
    "mount_point": ["/disk48", "/disk22"]
* Defaults to false.

```

## 字段提取配置

```

#*****
# Field extraction configuration
#*****Field extraction configuration

NOTE: If this is your first time configuring field extractions in
      props.conf, review the following information first.

There are three different "field extraction types" that you can use to
configure field extractions: TRANSFORMS, REPORT, and EXTRACT. They differ in
two significant ways: 1) whether they create indexed fields (fields
extracted at index time) or extracted fields (fields extracted at search
time), and 2), whether they include a reference to an additional component
called a "field transform," which you define separately in transforms.conf.

**Field extraction configuration: index time versus search time**

Use the TRANSFORMS field extraction type to create index-time field
extractions. Use the REPORT or EXTRACT field extraction types to create
search-time field extractions.

NOTE: Index-time field extractions have performance implications. Creating
      additions to Splunk's default set of indexed fields is ONLY
      recommended in specific circumstances. Whenever possible, extract
      fields only at search time.

There are times when you may find that you need to change or add to your set
of indexed fields. For example, you may have situations where certain
search-time field extractions are noticeably impacting search performance.
This can happen when the value of a search-time extracted field exists
outside of the field more often than not. For example, if you commonly
search a large event set with the expression company_id=1 but the value 1
occurs in many events that do *not* have company_id=1, you may want to add
company_id to the list of fields extracted by Splunk at index time. This is
because at search time, Splunk will want to check each instance of the value
1 to see if it matches company_id, and that kind of thing slows down

```



performance when you have Splunk searching a large set of data.

Conversely, if you commonly search a large event set with expressions like `company_id!=1` or `NOT company_id=1`, and the field `company_id` nearly *\*always\** takes on the value 1, you may want to add `company_id` to the list of fields extracted by Splunk at index time.

For more information about index-time field extraction, search the documentation for "index-time extraction." For more information about search-time field extraction, search the online documentation for "search-time extraction."

**\*\*Field extraction configuration: field transforms vs. "inline" (props.conf only) configs\*\***

The TRANSFORMS and REPORT field extraction types reference an additional component called a field transform, which you define separately in `transforms.conf`. Field transforms contain a field-extracting regular expression and other attributes that govern the way that the transform extracts fields. Field transforms are always created in conjunction with field extraction stanzas in `props.conf`; they do not stand alone.

The EXTRACT field extraction type is considered to be "inline," which means that it does not reference a field transform. It contains the regular expression that Splunk uses to extract fields at search time. You can use EXTRACT to define a field extraction entirely within `props.conf`--no `transforms.conf` component is required.

**\*\*Search-time field extractions: Why use REPORT if EXTRACT will do?\*\***

It's a good question. And much of the time, EXTRACT is all you need for search-time field extraction. But when you build search-time field extractions, there are specific cases that require the use of REPORT and the field transform that it references. Use REPORT if you want to:

- \* Reuse the same field-extracting regular expression across multiple sources, source types, or hosts. If you find yourself using the same regex to extract fields across several different sources, source types, and hosts, set it up as a transform, and then reference it in REPORT extractions in those stanzas. If you need to update the regex you only have to do it in one place. Handy!
- \* Apply more than one field-extracting regular expression to the same source, source type, or host. This can be necessary in cases where the field or fields that you want to extract from a particular source, source type, or host appear in two or more very different event patterns.
- \* Set up delimiter-based field extractions. Useful if your event data presents field-value pairs (or just field values) separated by delimiters such as commas, spaces, bars, and so on.
- \* Configure extractions for multivalued fields. You can have Splunk append additional values to a field as it finds them in the event data.
- \* Extract fields with names beginning with numbers or underscores. Ordinarily, Splunk's key cleaning functionality removes leading numeric characters and underscores from field names. If you need to keep them, configure your field transform to turn key cleaning off.
- \* Manage formatting of extracted fields, in cases where you are extracting multiple fields, or are extracting both the field name and field value.

**\*\*Precedence rules for TRANSFORMS, REPORT, and EXTRACT field extraction types\*\***

- \* For each field extraction, Splunk takes the configuration from the highest precedence configuration stanza (see precedence rules at the beginning of this file).
- \* If a particular field extraction is specified for a source and a source type, the field extraction for source wins out.
- \* Similarly, if a particular field extraction is specified in `../local/` for a `<spec>`, it overrides that field extraction in `../default/`.

TRANSFORMS-<class> = <transform\_stanza\_name>, <transform\_stanza\_name2>,...

\* Used for creating indexed fields (index-time field extractions).

\* <class> is a unique literal string that identifies the namespace of the field you're extracting.

**\*\*Note:\*\*** <class> values do not have to follow field name syntax

restrictions. You can use characters other than a-z, A-Z, and 0-9, and spaces are allowed. <class> values are not subject to key cleaning.

- \* <transform\_stanza\_name> is the name of your stanza from transforms.conf.
- \* Use a comma-separated list to apply multiple transform stanzas to a single TRANSFORMS extraction. Splunk applies them in the list order. For example, this sequence ensures that the [yellow] transform stanza gets applied first, then [blue], and then [red]:
 

```
[source::color_logs]
TRANSFORMS-colorchange = yellow, blue, red
```

REPORT-<class> = <transform\_stanza\_name>, <transform\_stanza\_name2>,...

- \* Used for creating extracted fields (search-time field extractions) that reference one or more transforms.conf stanzas.
- \* <class> is a unique literal string that identifies the namespace of the field you're extracting.
  - \*\*Note:\*\* <class> values do not have to follow field name syntax restrictions. You can use characters other than a-z, A-Z, and 0-9, and spaces are allowed. <class> values are not subject to key cleaning.
- \* <transform\_stanza\_name> is the name of your stanza from transforms.conf.
- \* Use a comma-separated list to apply multiple transform stanzas to a single REPORT extraction.
 

Splunk applies them in the list order. For example, this sequence insures that the [yellow] transform stanza gets applied first, then [blue], and then [red]:

```
[source::color_logs]
REPORT-colorchange = yellow, blue, red
```

EXTRACT-<class> = [<regex>|<regex> in <src\_field>]

- \* Used to create extracted fields (search-time field extractions) that do not reference transforms.conf stanzas.
- \* Performs a regex-based field extraction from the value of the source field.
- \* <class> is a unique literal string that identifies the namespace of the field you're extracting.
  - \*\*Note:\*\* <class> values do not have to follow field name syntax restrictions. You can use characters other than a-z, A-Z, and 0-9, and spaces are allowed. <class> values are not subject to key cleaning.
- \* The <regex> is required to have named capturing groups. When the <regex> matches, the named capturing groups and their values are added to the event.
- \* dotall (?s) and multiline (?m) modifiers are added in front of the regex. So internally, the regex becomes (?ms)<regex>.
- \* Use '<regex> in <src\_field>' to match the regex against the values of a specific field. Otherwise it just matches against \_raw (all raw event data).
- \* NOTE: <src\_field> can only contain alphanumeric characters and underscore (a-z, A-Z, 0-9, and \_).
- \* If your regex needs to end with 'in <string>' where <string> is \*not\* a field name, change the regex to end with '[i]n <string>' to ensure that Splunk doesn't try to match <string> to a field name.

KV\_MODE = [none|auto|auto\_escaped|multi|json|xml]

- \* Used for search-time field extractions only.
- \* Specifies the field/value extraction mode for the data.
- \* Set KV\_MODE to one of the following:
  - \* none: if you want no field/value extraction to take place.
  - \* auto: extracts field/value pairs separated by equal signs.
  - \* auto\_escaped: extracts fields/value pairs separated by equal signs and honors \" and \\ as escaped sequences within quoted values, e.g field="value with \"nested\" quotes"
  - \* multi: invokes the multikv search command to expand a tabular event into multiple events.
  - \* xml : automatically extracts fields from XML data.
  - \* json: automatically extracts fields from JSON data.
- \* Setting to 'none' can ensure that one or more user-created regexes are not overridden by automatic field/value extraction for a particular host, source, or source type, and also increases search performance.
- \* Defaults to auto.
- \* The 'xml' and 'json' modes will not extract any fields when used on data that isn't of the correct format (JSON or XML).

AUTO\_KV\_JSON = [true|false]

- \* Used for search-time field extractions only.
- \* Specifies whether to try json extraction automatically.
- \* Defaults to true.

KV\_TRIM\_SPACES = true|false

- \* Modifies the behavior of KV\_MODE when set to auto, and auto\_escaped.
- \* Traditionally, automatically identified fields have leading and trailing whitespace removed from their values.
  - \* Example event: 2014-04-04 10:10:45 myfield=" apples "
  - would result in a field called 'myfield' with a value of 'apples'.
- \* If this value is set to false, then external whitespace then this outer space is retained.
  - \* Example: 2014-04-04 10:10:45 myfield=" apples "
  - would result in a field called 'myfield' with a value of ' apples '.
- \* The trimming logic applies only to space characters, not tabs, or other whitespace.
- \* NOTE: The Splunk UI currently has limitations with displaying and interactively clicking on fields that have leading or trailing whitespace. Field values with leading or trailing spaces may not look distinct in the event viewer, and clicking on a field value will typically insert the term into the search string without its embedded spaces.
- \* These warts are not specific to this feature. Any such embedded spaces will behave this way.
- \* The Splunk search language and included commands will respect the spaces.
- \* Defaults to true.

CHECK\_FOR\_HEADER = [true|false]

- \* Used for index-time field extractions only.
- \* Set to true to enable header-based field extraction for a file.
- \* If the file has a list of columns and each event contains a field value (without field name), Splunk picks a suitable header line to use to for extracting field names.
- \* If the file has a list of columns and each event contains a field value (without a field name), Splunk picks a suitable header line to use for field extraction.
- \* Can only be used on the basis of [<sourcetype>] or [source::<spec>], not [host::<spec>].
- \* Disabled when LEARN\_SOURCETYPE = false.
- \* Will cause the indexed source type to have an appended numeral; for example, sourcetype-2, sourcetype-3, and so on.
- \* The field names are stored in etc/apps/learned/local/props.conf.
  - \* Because of this, this feature will not work in most environments where the data is forwarded.
- \* This setting applies at input time, when data is first read by Splunk. The setting is used on a Splunk system that has configured inputs acquiring the data.
- \* Defaults to false.

SEDCMD-<class> = <sed script>

- \* Only used at index time.
- \* Commonly used to anonymize incoming data at index time, such as credit card or social security numbers. For more information, search the online documentation for "anonymize data."
- \* Used to specify a sed script which Splunk applies to the \_raw field.
- \* A sed script is a space-separated list of sed commands. Currently the following subset of sed commands is supported:
  - \* replace (s) and character substitution (y).
- \* Syntax:
  - \* replace - s/regex/replacement/flags
  - \* regex is a perl regular expression (optionally containing capturing groups).
  - \* replacement is a string to replace the regex match. Use \n for back references, where "n" is a single digit.
  - \* flags can be either: g to replace all matches, or a number to replace a specified match.
  - \* substitute - y/string1/string2/
  - \* substitutes the string1[i] with string2[i]

FIELDALIAS-<class> = (<orig\_field\_name> AS <new\_field\_name>)+

- \* Use this to apply aliases to a field. The original field is not removed. This just means that the original field can be searched on using any of its aliases.

- \* You can create multiple aliases for the same field.
- \* <orig\_field\_name> is the original name of the field.
- \* <new\_field\_name> is the alias to assign to the field.
- \* You can include multiple field alias renames in the same stanza.
- \* Field aliasing is performed at search time, after field extraction, but before calculated fields (EVAL-\* statements) and lookups.

This means that:

- \* Any field extracted at search time can be aliased.
- \* You can specify a lookup based on a field alias.
- \* You cannot alias a calculated field.

EVAL-<fieldname> = <eval statement>

- \* Use this to automatically run the <eval statement> and assign the value of the output to <fieldname>. This creates a "calculated field."
- \* When multiple EVAL-\* statements are specified, they behave as if they are run in parallel, rather than in any particular sequence.

For example say you have two statements: EVAL-x = y\*2 and EVAL-y=100. In this case, "x" will be assigned the original value of "y \* 2," not the value of "y" after it is set to 100.

- \* Splunk processes calculated fields after field extraction and field aliasing but before lookups. This means that:
  - \* You can use a field alias in the eval statement for a calculated field.
  - \* You cannot use a field added through a lookup in an eval statement for a calculated field.

LOOKUP-<class> = \$TRANSFORM (<match\_field> (AS <match\_field\_in\_event>)?)+ (OUTPUT|OUTPUTNEW (<output\_field> (AS <output\_field\_in\_event>)? )+ )?

- \* At search time, identifies a specific lookup table and describes how that lookup table should be applied to events.
- \* <match\_field> specifies a field in the lookup table to match on.
  - \* By default Splunk looks for a field with that same name in the event to match with (if <match\_field\_in\_event> is not provided)
  - \* You must provide at least one match field. Multiple match fields are allowed.
- \* <output\_field> specifies a field in the lookup entry to copy into each matching event, where it will be in the field <output\_field\_in\_event>.
  - \* If you do not specify an <output\_field\_in\_event> value, Splunk uses <output\_field>.
  - \* A list of output fields is not required.
- \* If they are not provided, all fields in the lookup table except for the match fields (and the timestamp field if it is specified) will be output for each matching event.
- \* If the output field list starts with the keyword "OUTPUTNEW" instead of "OUTPUT", then each outputfield is only written out if it did not previously exist. Otherwise, the output fields are always overridden. Any event that has all of the <match\_field> values but no matching entry in the lookup table clears all of the output fields. NOTE that OUTPUTNEW behavior has changed since 4.1.x (where \*none\* of the output fields were written to if \*any\* of the output fields previously existed).
- \* Splunk processes lookups after it processes field extractions, field aliases, and calculated fields (EVAL-\* statements). This means that you can use extracted fields, aliased fields, and calculated fields to specify lookups. But you can't use fields discovered by lookups in the configurations of extracted fields, aliased fields, or calculated fields.
- \* The LOOKUP- prefix is actually case-insensitive. Acceptable variants include:
 

```
LOOKUP_<class> = [...]
LOOKUP<class>  = [...]
lookup_<class> = [...]
lookup<class>  = [...]
```

## 二进制文件配置

```
#*****
# Binary file configuration
#*****Binary file configuration

NO_BINARY_CHECK = [true|false]
* When set to true, Splunk processes binary files.
* Can only be used on the basis of [<sourcetype>], or [source::<source>],
```

```

    not [host::<host>].
* Defaults to false (binary files are ignored).
* This setting applies at input time, when data is first read by Splunk.
  The setting is used on a Splunk system that has configured inputs
  acquiring the data.

detect_trailing_nulls = [auto|true|false]
* When enabled, Splunk will try to avoid reading in null bytes at the end of
  a file.
* When false, splunk will assume that all the bytes in the file should be
  read and indexed.
* Set this value to false for UTF-16 and other encodings (CHARSET) values
  that can have null bytes as part of the character text.
* Subtleties of 'true' vs 'auto':
  * 'true' is the splunk-on-windows historical behavior of trimming all null
    bytes.
  * 'auto' is currently a synonym for true but will be extended to be
    sensitive to the charset selected (ie quantized for multi-byte
    encodings, and disabled for unsafe variable-width encodings)
* This feature was introduced to work around programs which foolishly
  pre-allocate their log files with nulls and fill in data later. The
  well-known case is Internet Information Server.
* This setting applies at input time, when data is first read by Splunk.
  The setting is used on a Splunk system that has configured inputs
  acquiring the data.
* Defaults to false on *nix, true on windows.

```

## 分段配置

```

#*****
# Segmentation configuration
#*****Segmentation configuration

SEGMENTATION = <segmenter>
* Specifies the segmenter from segmenters.conf to use at index time for the
  host, source, or sourcetype specified by <spec> in the stanza heading.
* Defaults to indexing.

SEGMENTATION-<segment selection> = <segmenter>
* Specifies that Splunk Web should use the specific segmenter (from
  segmenters.conf) for the given <segment selection> choice.
* Default <segment selection> choices are: all, inner, outer, raw. For more
  information see the Admin Manual.
* Do not change the set of default <segment selection> choices, unless you
  have some overriding reason for doing so. In order for a changed set of
  <segment selection> choices to appear in Splunk Web, you will need to edit
  the Splunk Web UI.

```

## 文件校验和配置

```

#*****
# File checksum configuration
#*****File checksum configuration

CHECK_METHOD = [endpoint_md5|entire_md5|modtime]
* Set CHECK_METHOD endpoint_md5 to have Splunk checksum of the first and
  last 256 bytes of a file. When it finds matches, Splunk lists the file as
  already indexed and indexes only new data, or ignores it if there is no
  new data.
* Set CHECK_METHOD = entire_md5 to use the checksum of the entire file.
* Set CHECK_METHOD = modtime to check only the modification time of the
  file.
* Settings other than endpoint_md5 cause Splunk to index the entire file for
  each detected change.
* Important: this option is only valid for [source::<source>] stanzas.
* This setting applies at input time, when data is first read by Splunk.
  The setting is used on a Splunk system that has configured inputs
  acquiring the data.

```

```

* Defaults to endpoint_md5.

initCrcLength = <integer>
* See documentation in inputs.conf.spec.

```

## 小型文件设置

```

#*****
# Small file settings
#*****Small file settings

PREFIX_SOURCETYPE = [true|false]
* NOTE: this attribute is only relevant to the "[too_small]" sourcetype.
* Determines the source types that are given to files smaller than 100
  lines, and are therefore not classifiable.
* PREFIX_SOURCETYPE = false sets the source type to "too_small."
* PREFIX_SOURCETYPE = true sets the source type to "<sourcename>-too_small",
  where "<sourcename>" is a cleaned up version of the filename.
* The advantage of PREFIX_SOURCETYPE = true is that not all small files
  are classified as the same source type, and wildcard searching is often
  effective.
* For example, a Splunk search of "sourcetype=access*" will retrieve
  "access" files as well as "access-too_small" files.
* This setting applies at input time, when data is first read by Splunk.
  The setting is used on a Splunk system that has configured inputs
  acquiring the data.
* Defaults to true.

```

## 来源类型配置

```

#*****
# Sourcetype configuration
#*****Sourcetype configuration

sourcetype = <string>
* Can only be set for a [source:... ] stanza.
* Anything from that <source> is assigned the specified source type.
* Is used by file-based inputs, at input time (when accessing logfiles) such
  as on a forwarder, or indexer monitoring local files.
* sourcetype assignment settings on a system receiving forwarded splunk data
  will not be applied to forwarded data.
* For logfiles read locally, data from logfiles matching <source> is
  assigned the specified source type.
* Defaults to empty.

# The following attribute/value pairs can only be set for a stanza that
# begins with [<sourcetype>]:

rename = <string>
* Renames [<sourcetype>] as <string> at search time
* With renaming, you can search for the [<sourcetype>] with
  sourcetype=<string>
* To search for the original source type without renaming it, use the
  field _sourcetype.
* Data from a a renamed sourcetype will only use the search-time
  configuration for the target sourcetype. Field extractions
  (REPORTS/EXTRACT) for this stanza sourcetype will be ignored.
* Defaults to empty.

invalid_cause = <string>
* Can only be set for a [<sourcetype>] stanza.
* If invalid_cause is set, the Tailing code (which handles uncompressed
  logfiles) will not read the data, but hand it off to other components or
  throw an error.
* Set <string> to "archive" to send the file to the archive processor
  (specified in unarchive_cmd).

```

- \* When set to "winevt", this causes the file to be handed off to the eventlog input processor.
- \* Set to any other string to throw an error in the splunkd.log if you are running Splunklogger in debug mode.
- \* This setting applies at input time, when data is first read by Splunk. The setting is used on a Splunk system that has configured inputs acquiring the data.
- \* Defaults to empty.

is\_valid = [true|false]

- \* Automatically set by invalid\_cause.
- \* This setting applies at input time, when data is first read by Splunk, such as on a forwarder.
- \* This setting applies at input time, when data is first read by Splunk. The setting is used on a Splunk system that has configured inputs acquiring the data.
- \* DO NOT SET THIS.
- \* Defaults to true.

unarchive\_cmd = <string>

- \* Only called if invalid\_cause is set to "archive".
- \* This field is only valid on [source::<source>] stanzas.
- \* <string> specifies the shell command to run to extract an archived source.
- \* Must be a shell command that takes input on stdin and produces output on stdout.
- \* Use \_auto for Splunk's automatic handling of archive files (tar, tar.gz, tgz, tbz, tbz2, zip)
- \* This setting applies at input time, when data is first read by Splunk. The setting is used on a Splunk system that has configured inputs acquiring the data.
- \* Defaults to empty.

unarchive\_sourcetype = <string>

- \* Sets the source type of the contents of the matching archive file. Use this field instead of the sourcetype field to set the source type of archive files that have the following extensions: gz, bz, bz2, Z.
- \* If this field is empty (for a matching archive file props lookup) Splunk strips off the archive file's extension (.gz, bz etc) and lookup another stanza to attempt to determine the sourcetype.
- \* This setting applies at input time, when data is first read by Splunk. The setting is used on a Splunk system that has configured inputs acquiring the data.
- \* Defaults to empty.

LEARN\_SOURCETYPE = [true|false]

- \* Determines whether learning of known or unknown sourcetypes is enabled.
  - \* For known sourcetypes, refer to LEARN\_MODEL.
  - \* For unknown sourcetypes, refer to the rule:: and delayedrule:: configuration (see below).
- \* Setting this field to false disables CHECK\_FOR\_HEADER as well (see above).
- \* This setting applies at input time, when data is first read by Splunk. The setting is used on a Splunk system that has configured inputs acquiring the data.
- \* Defaults to true.

LEARN\_MODEL = [true|false]

- \* For known source types, the file classifier adds a model file to the learned directory.
- \* To disable this behavior for diverse source types (such as sourcecode, where there is no good example to make a sourcetype) set LEARN\_MODEL = false.
- \* This setting applies at input time, when data is first read by Splunk. The setting is used on a Splunk system that has configured inputs acquiring the data.
- \* Defaults to true.

maxDist = <integer>

- \* Determines how different a source type model may be from the current file.
- \* The larger the maxDist value, the more forgiving Splunk will be with differences.
  - \* For example, if the value is very small (for example, 10), then files of the specified sourcetype should not vary much.

- \* A larger value indicates that files of the given source type can vary quite a bit.
- \* If you're finding that a source type model is matching too broadly, reduce its maxDist value by about 100 and try again. If you're finding that a source type model is being too restrictive, increase its maxDist value by about 100 and try again.
- \* This setting applies at input time, when data is first read by Splunk. The setting is used on a Splunk system that has configured inputs acquiring the data.
- \* Defaults to 300.

# rule:: and delayedrule:: configuration

MORE\_THAN<optional\_unique\_value>\_<number> = <regular expression> (empty)  
 LESS\_THAN<optional\_unique\_value>\_<number> = <regular expression> (empty)

- \* These settings apply at input time, when data is first read by Splunk, such as on a forwarder.

An example:

```
[rule::bar_some]
sourcetype = source_with_lots_of_bars
# if more than 80% of lines have "----", but fewer than 70% have "####"
# declare this a "source_with_lots_of_bars"
MORE_THAN_80 = ----
LESS_THAN_70 = ####
```

A rule can have many MORE\_THAN and LESS\_THAN patterns, and all are required for the rule to match.

## 已配置注释处理器

```
*****
# Annotation Processor configured
*****Annotation Processor configured

ANNOTATE_PUNCT = [true|false]
* Determines whether to index a special token starting with "punct:."
  * The "punct:." key contains punctuation in the text of the event.
    It can be useful for finding similar events
  * If it is not useful for your dataset, or if it ends up taking
    too much space in your index it is safe to disable it
* Defaults to true.
```

## 标头处理器配置

```
*****
# Header Processor configuration
*****Header Processor configuration

HEADER_MODE = <empty> | always | firstline | none
* Determines whether to use the inline ***SPLUNK*** directive to rewrite index-time fields.
  * If "always", any line with ***SPLUNK*** can be used to rewrite
    index-time fields.
  * If "firstline", only the first line can be used to rewrite
    index-time fields.
  * If "none", the string ***SPLUNK*** is treated as normal data.
  * If <empty>, scripted inputs take the value "always" and file inputs
    take the value "none".
* This setting applies at input time, when data is first read by Splunk.
  The setting is used on a Splunk system that has configured inputs
  acquiring the data.
* Defaults to <empty>.
```

## 内部设置



```

#####
# Internal settings
#####Internal settings

# NOT YOURS. DO NOT SET.

_actions = <string>
* Internal field used for user-interface control of objects.
* Defaults to "new,edit,delete".

pulldown_type = <bool>
* Internal field used for user-interface control of source types.
* Defaults to empty.

given_type = <string>
* Internal field used by the CHECK_FOR_HEADER feature to remember the
  original sourcetype.
* This setting applies at input time, when data is first read by Splunk.
  The setting is used on a Splunk system that has configured inputs
  acquiring the data.
* Default to unset.

```

## 来源类型类别和描述

```

#####
# Sourcetype Category and Descriptions
#####Sourcetype Category and Descriptions

description = <string>
* Field used to describe the sourcetype. Does not affect indexing behaviour.
* Defaults to unset.

category = <string>
* Field used to classify sourcetypes for organization in the front end. Case
  sensitive. Does not affect indexing behaviour.
* Defaults to unset.

```

## props.conf.example

```

#   Version 6.5.0
#
# The following are example props.conf configurations. Configure properties for
# your data.
#
# To use one or more of these configurations, copy the configuration block into
# props.conf in $SPLUNK_HOME/etc/system/local/. You must restart Splunk to
# enable configurations.
#
# To learn more about configuration files (including precedence) please see the
# documentation located at
# http://docs.splunk.com/Documentation/Splunk/latest/Admin/Aboutconfigurationfiles

#####
# Line merging settings
#####

# The following example linemerges source data into multi-line events for
# apache_error sourcetype.

[apache_error]
SHOULD_LINEMERGE = True

#####

```

```

# Settings for tuning
#####

# The following example limits the amount of characters indexed per event from
# host::small_events.

[host::small_events]
TRUNCATE = 256

# The following example turns off DATETIME_CONFIG (which can speed up indexing)
# from any path that ends in /mylogs/*.log.
#
# In addition, the default splunk behavior of finding event boundaries
# via per-event timestamps can't work with NONE, so we disable
# SHOULD_LINEMERGE, essentially declaring that all events in this file are
# single-line.

[source::.../mylogs/*.log]
DATETIME_CONFIG = NONE
SHOULD_LINEMERGE = false

#####
# Timestamp extraction configuration
#####

# The following example sets Eastern Time Zone if host matches nyc*.

[host::nyc*]
TZ = US/Eastern

# The following example uses a custom datetime.xml that has been created and
# placed in a custom app directory. This sets all events coming in from hosts
# starting with dharma to use this custom file.

[host::dharma*]
DATETIME_CONFIG = <etc/apps/custom_time/datetime.xml>

#####
## Timezone alias configuration
#####

# The following example uses a custom alias to disambiguate the Australian
# meanings of EST/EDT

TZ_ALIAS = EST=GMT+10:00,EDT=GMT+11:00

# The following example gives a sample case wherein, one timezone field is
# being replaced by/interpreted as another.

TZ_ALIAS = EST=AEST,EDT=AEDT

#####
# Transform configuration
#####

# The following example creates a search field for host::foo if tied to a
# stanza in transforms.conf.

[host::foo]
TRANSFORMS-foo=foobar

# The following stanza extracts an ip address from _raw
[my_sourcetype]
EXTRACT-extract_ip = (?<ip>\d{1,3}\.\d{1,3}\.\d{1,3}\.\d{1,3})

# The following example shows how to configure lookup tables
[my_lookuptype]
LOOKUP-foo = mylookuptable userid AS myuserid OUTPUT username AS myusername

```

```

# The following shows how to specify field aliases
FIELDALIAS-foo = user AS myuser id AS myid

#####
# Sourcetype configuration
#####

# The following example sets a sourcetype for the file web_access.log for a
# unix path.

[source:.../web_access.log]
sourcetype = splunk_web_access

# The following example sets a sourcetype for the Windows file iis6.log. Note:
# Backslashes within Windows file paths must be escaped.

[source:...\\iis\\iis6.log]
sourcetype = iis_access

# The following example untars syslog events.

[syslog]
invalid_cause = archive
unarchive_cmd = gzip -cd -

# The following example learns a custom sourcetype and limits the range between
# different examples with a smaller than default maxDist.

[custom_sourcetype]
LEARN_MODEL = true
maxDist = 30

# rule:: and delayedrule:: configuration
# The following examples create sourcetype rules for custom sourcetypes with
# regex.

[rule::bar_some]
sourcetype = source_with_lots_of_bars
MORE_THAN_80 = ----

[delayedrule::baz_some]
sourcetype = my_sourcetype
LESS_THAN_70 = ###

#####
# File configuration
#####

# Binary file configuration
# The following example eats binary files from the sourcetype
# "imported_records".

[imported_records]
NO_BINARY_CHECK = true

# File checksum configuration
# The following example checks the entirety of every file in the web_access dir
# rather than skipping files that appear to be the same.

[source:.../web_access/*]
CHECK_METHOD = entire_md5

```

## pubsub.conf

以下为 pubsub.conf 的规范和示例文件。

## pubsub.conf.spec

```
# Version 6.5.0
#
# This file contains possible attributes and values for configuring a client of
# the PubSub system (broker).
#
# To set custom configurations, place a pubsub.conf in
# $SPLUNK_HOME/etc/system/local/.
# For examples, see pubsub.conf.example. You must restart Splunk to enable
# configurations.
#
# To learn more about configuration files (including precedence) please see the
# documentation located at
# http://docs.splunk.com/Documentation/Splunk/latest/Admin/Aboutconfigurationfiles
```

## 全局设置

```
# GLOBAL SETTINGS
# Use the [default] stanza to define any global settings.
# * You can also define global settings outside of any stanza, at the top of
#   the file.
# * Each conf file should have at most one default stanza. If there are
#   multiple default stanzas, attributes are combined. In the case of
#   multiple definitions of the same attribute, the last definition in the
#   file wins.
# * If an attribute is defined at both the global level and in a specific
#   stanza, the value in the specific stanza takes precedence.

#*****
# Configure the physical location where deploymentServer is running.
# This configuration is used by the clients of the pubsub system.
#*****
```

## [pubsub-server:deploymentServer]

```
[pubsub-server:deploymentServer]
disabled = <false or true>
* defaults to 'false'

targetUri = <IP:Port>|<hostname:Port>|direct
* specify either the url of a remote server in case the broker is remote, or
  just the keyword "direct" when broker is in-process.
* It is usually a good idea to co-locate the broker and the Deployment Server
  on the same Splunk. In such a configuration, all
* deployment clients would have targetUri set to deploymentServer:port.

#*****
# The following section is only relevant to Splunk developers.
#*****

# This "direct" configuration is always available, and cannot be overridden.
```

## [pubsub-server:direct]

```
[pubsub-server:direct]
disabled = false
targetUri = direct
```

## [pubsub-server:<logicalName>]

```
[pubsub-server:<logicalName>]
* It is possible for any Splunk to be a broker. If you have multiple brokers,
```

```

    assign a logicalName that is used by the clients to refer to it.

disabled = <false or true>
* defaults to 'false'

targetUri = <IP:Port>|<hostname:Port>|direct
* The Uri of a Splunk that is being used as a broker.
* The keyword "direct" implies that the client is running on the same Splunk
  instance as the broker.

```

## pubsub.conf.example

```

#   Version 6.5.0

[pubsub-server:deploymentServer]
disabled=false
targetUri=somehost:8089

[pubsub-server:internalbroker]
disabled=false
targetUri=direct

```

## restmap.conf

以下为 restmap.conf 的规范和示例文件。

### restmap.conf.spec

```

#   Version 6.5.0
#
# This file contains possible attribute and value pairs for creating new
# Representational State Transfer (REST) endpoints.
#
# There is a restmap.conf in $SPLUNK_HOME/etc/system/default/. To set custom
# configurations, place a restmap.conf in $SPLUNK_HOME/etc/system/local/. For
# help, see restmap.conf.example. You must restart Splunk to enable
# configurations.
#
# To learn more about configuration files (including precedence) please see the
# documentation located at
# http://docs.splunk.com/Documentation/Splunk/latest/Admin/Aboutconfigurationfiles
#
# NOTE: You must register every REST endpoint via this file to make it
# available.

#####
# Global stanza

[global]
* This stanza sets global configurations for all REST endpoints.
* Follow this stanza name with any number of the following attribute/value
  pairs.

allowGetAuth=[true|false]
* Allow user/password to be passed as a GET parameter to endpoint
  services/auth/login.
* Setting this to true, while convenient, may result in user/password getting
  logged as cleartext in Splunk's logs *and* any proxy servers in between.
* Defaults to false.

allowRestReplay=[true|false]
* POST/PUT/DELETE requests can be replayed on other nodes in the deployment.
* This enables centralized management.
* Turn on or off this feature. You can also control replay at each endpoint
  level. This feature is currently INTERNAL and should not be turned on without
  consulting splunk support.

```

```

* Defaults to false

defaultRestReplayStanza=<string>
* Points to global rest replay configuration stanza.
* Related to allowRestReplay
* Defaults to "restreplayshec"

pythonHandlerPath=<path>
* Path to 'main' python script handler.
* Used by the script handler to determine where the actual 'main' script is
  located.
* Typically, you should not need to change this.
* Defaults to $SPLUNK_HOME/bin/rest_handler.py.

#####
# Applicable to all REST stanzas
# Stanza definitions below may supply additional information for these.
#

[<rest endpoint name>:<endpoint description string>]
match=<path>
* Specify the URI that calls the handler.
* For example if match=/foo, then https://$SERVER:$PORT/services/foo calls this
  handler.
* NOTE: You must start your path with a /.

requireAuthentication=[true|false]
* This optional attribute determines if this endpoint requires authentication.
* Defaults to 'true'.

authKeyStanza=<stanza>
* This optional attribute determines the location of the pass4SymmKey in the
  server.conf to be used for endpoint authentication.
* Defaults to 'general' stanza.
* Only applicable if the requireAuthentication is set true.

restReplay=[true|false]
* This optional attribute enables rest replay on this endpoint group
* Related to allowRestReplay
* This feature is currently INTERNAL and should not be turned on without consulting
  splunk support.
* Defaults to false

restReplayStanza=<string>
* This points to stanza which can override the [global]/defaultRestReplayStanza
  value on a per endpoint/regex basis
* Defaults to empty

capability=<capabilityName>
capability.<post|delete|get|put>=<capabilityName>
* Depending on the HTTP method, check capabilities on the authenticated session user.
* If you use 'capability.post|delete|get|put,' then the associated method is
  checked against the authenticated user's role.
* If you just use 'capability,' then all calls get checked against this
  capability (regardless of the HTTP method).

acceptFrom=<network_acl> ...
* Lists a set of networks or addresses to allow this endpoint to be accessed
  from.
* This shouldn't be confused with the setting of the same name in the
  [httpServer] stanza of server.conf which controls whether a host can
  make HTTP requests at all
* Each rule can be in the following forms:
  1. A single IPv4 or IPv6 address (examples: "10.1.2.3", "fe80::4a3")
  2. A CIDR block of addresses (examples: "10/8", "fe80:1234/32")
  3. A DNS name, possibly with a '*' used as a wildcard (examples:
    "myhost.example.com", "*.splunk.com")
  4. A single '*' which matches anything
* Entries can also be prefixed with '!' to cause the rule to reject the
  connection. Rules are applied in order, and the first one to match is
  used. For example, "!10.1/16, *" will allow connections from everywhere
  except the 10.1.*.* network.

```

```

* Defaults to "" (accept from anywhere)

includeInAccessLog=[true|false]
* If this is set to false, requests to this endpoint will not appear
  in splunkd_access.log
* Defaults to 'true'.

#####
# Per-endpoint stanza
# Specify a handler and other handler-specific settings.
# The handler is responsible for implementing arbitrary namespace underneath
# each REST endpoint.

[script:<uniqueName>]
* NOTE: The uniqueName must be different for each handler.
* Call the specified handler when executing this endpoint.
* The following attribute/value pairs support the script handler.

scripttype=python
* Tell the system what type of script to execute when using this endpoint.
* Defaults to python.
* If set to "persist" it will run the script via a persistent-process that
  uses the protocol from persistconn/appserver.py.

handler=<SCRIPT>.<CLASSNAME>
* The name and class name of the file to execute.
* The file *must* live in an application's bin subdirectory.
* For example, $SPLUNK_HOME/etc/apps/<APPNAME>/bin/TestHandler.py has a class
  called MyHandler (which, in the case of python must be derived from a base
  class called 'splunk.rest.BaseRestHandler'). The tag/value pair for this is:
  "handler=TestHandler.MyHandler".

xsl=<path to XSL transform file>
* Optional.
* Perform an optional XSL transform on data returned from the handler.
* Only use this if the data is XML.
* Does not apply to scripttype=persist.

script=<path to a script executable>
* For scripttype=python this is optional. It allows you to run a script
  which is *not* derived from 'splunk.rest.BaseRestHandler'. This is
  rarely used. Do not use this unless you know what you are doing.
* For scripttype=persist this is the path with is sent to the driver
  to execute. In that case, environment variables are substituted.

script.arg.<N>=<string>
* Only has effect for scripttype=persist.
* List of arguments which are passed to the driver to start the script .
* The script can make use of this information however it wants.
* Environment variables are substituted.

script.param=<string>
* Optional.
* Only has effect for scripttype=persist.
* Free-form argument that is passed to the driver when it starts the
  script.
* The script can make use of this information however it wants.
* Environment variables are substituted.

output_modes=<csv list>
* Specifies which output formats can be requested from this endpoint.
* Valid values are: json, xml.
* Defaults to xml.

passSystemAuth=<bool>
* Specifies whether or not to pass in a system-level authentication token on
  each request.
* Defaults to false.

driver=<path>
* For scripttype=persist, specifies the command to start a persistent
  server for this process.

```

- \* Endpoints that share the same driver configuration can share processes.
- \* Environment variables are substituted.
- \* Defaults to using the persistconn/appserver.py server.

driver.arg.<n> = <string>

- \* For scripttype=persist, specifies the command to start a persistent server for this process.
- \* Environment variables are substituted.
- \* Only takes effect when "driver" is specifically set.

driver.env.<name>=<value>

- \* For scripttype=persist, specifies an environment variable to set when running the driver process.

passConf=<bool>

- \* If set, the script is sent the contents of this configuration stanza as part of the request.
- \* Only has effect for scripttype=persist.
- \* Defaults to true.

passPayload=[true | false | base64]

- \* If set to true, sends the driver the raw, unparsed body of the POST/PUT as a "payload" string.
- \* If set to "base64", the same body is instead base64-encoded and sent as a "payload\_base64" string.
- \* Only has effect for scripttype=persist.
- \* Defaults to false.

passSession=<bool>

- \* If set to true, sends the driver information about the user's session. This includes the user's name, an active authtoken, and other details.
- \* Only has effect for scripttype=persist.
- \* Defaults to true.

passHttpHeaders=<bool>

- \* If set to true, sends the driver the HTTP headers of the request.
- \* Only has effect for scripttype=persist.
- \* Defaults to false.

passHttpCookies=<bool>

- \* If set to true, sends the driver the HTTP cookies of the request.
- \* Only has effect for scripttype=persist.
- \* Defaults to false.

```
#####
# 'admin'
# The built-in handler for the Extensible Administration Interface.
# Exposes the listed EAI handlers at the given URL.
#

[admin:<uniqueName>]

match=<partial URL>
* URL which, when accessed, will display the handlers listed below.

members=<csv list>
* List of handlers to expose at this URL.
* See https://localhost:8089/services/admin for a list of all possible handlers.

#####
# 'admin_external'
# Register Python handlers for the Extensible Administration Interface.
# Handler will be exposed via its "uniqueName".
#

[admin_external:<uniqueName>]

handlertype=<script type>
* Currently only the value 'python' is valid.
```



```

handlerfile=<unique filename>
* Script to execute.
* For bin/myAwesomeAppHandler.py, specify only myAwesomeAppHandler.py.

handlerpersistentmode=[true|false]
* Set to true to run the script in persistent mode and keep the process running
  between requests.

handleractions=<comma separated list>
* List of EAI actions supported by this handler.
* Valid values are: create, edit, list, delete, _reload.

#####
# Validation stanzas
# Add stanzas using the following definition to add arg validation to
# the appropriate EAI handlers.

[validation:<handler-name>]

<field> = <validation-rule>

* <field> is the name of the field whose value would be validated when an
  object is being saved.
* <validation-rule> is an eval expression using the validate() function to
  evaluate arg correctness and return an error message. If you use a boolean
  returning function, a generic message is displayed.
* <handler-name> is the name of the REST endpoint which this stanza applies to
  handler-name is what is used to access the handler via
  /servicesNS/<user>/<app>/admin/<handler-name>.
* For example:
  action.email.sendresult = validate( isbool('action.email.sendresults'), "'action.email.sendresults' must be a
  boolean value").
* NOTE: use ' or $ to enclose field names that contain non alphanumeric characters.

#####
# 'eai'
# Settings to alter the behavior of EAI handlers in various ways.
# These should not need to be edited by users.
#

[eai:<EAI handler name>]

showInDirSvc = [true|false]
* Whether configurations managed by this handler should be enumerated via the
  directory service, used by SplunkWeb's "All Configurations" management page.
  Defaults to false.

desc = <human readable string>
* Allows for renaming the configuration type of these objects when enumerated
  via the directory service.

#####
# Miscellaneous
# The un-described parameters in these stanzas all operate according to the
# descriptions listed under "script:", above.
# These should not need to be edited by users - they are here only to quiet
# down the configuration checker.
#

[input:...]
dynamic = [true|false]
* If set to true, listen on the socket for data.
* If false, data is contained within the request body.
* Defaults to false.

[peerupload:...]
path = <directory path>
* Path to search through to find configuration bundles from search peers.

untar = [true|false]
* Whether or not a file should be untarred once the transfer is complete.

```

```
[restreplayshc]
methods = <comma separated strings>
* REST methods which will be replayed. POST, PUT, DELETE, HEAD, GET are the
  available options

nodelists = <comma separated string>
* strategies for replay. Allowed values are shc, nodes, filternodes
* shc - replay to all other nodes in Search Head Cluster
* nodes - provide raw comma separated URIs in nodes variable
* filternodes - filter out specific nodes. Always applied after other
  strategies

nodes = <comma separated management uris>
* list of specific nodes that you want the REST call to be replayed to

filternodes = <comma separated management uris>
* list of specific nodes that you do not want the REST call to be replayed to

[proxy:appsbrowser]
destination = <splunkbaseAPIURL>
* protocol, subdomain, domain, port, and path of the splunkbase api used to browse apps
* Defaults to https://splunkbase.splunk.com/api
```

## restmap.conf.example

```
# Version 6.5.0
#
# This file contains example REST endpoint configurations.
#
# To use one or more of these configurations, copy the configuration block into
# restmap.conf in $SPLUNK_HOME/etc/system/local/. You must restart Splunk to
# enable configurations.
#
# To learn more about configuration files (including precedence) please see the
# documentation located at
# http://docs.splunk.com/Documentation/Splunk/latest/Admin/Aboutconfigurationfiles

# The following are default REST configurations. To create your own endpoints,
# modify the values by following the spec outlined in restmap.conf.spec.

# //////////////////////////////////////////////////
# global settings
# //////////////////////////////////////////////////

[global]

# indicates if auths are allowed via GET params
allowGetAuth=false

#The default handler (assuming that we have PYTHONPATH set)
pythonHandlerPath=$SPLUNK_HOME/bin/rest_handler.py

# //////////////////////////////////////////////////
# internal C++ handlers
# NOTE: These are internal Splunk-created endpoints. 3rd party developers can
# only use script or search can be used as handlers.
# (Please see restmap.conf.spec for help with configurations.)
# //////////////////////////////////////////////////

[SBA:sba]
match=/properties
capability=get_property_map

[asyncsearch:asyncsearch]
```

```
match=/search
capability=search
```

## savedsearches.conf

以下为 savedsearches.conf 的规范和示例文件。

### savedsearches.conf.spec

```
# Version 6.5.0
#
# This file contains possible attribute/value pairs for saved search entries in
# savedsearches.conf. You can configure saved searches by creating your own
# savedsearches.conf.
#
# There is a default savedsearches.conf in $SPLUNK_HOME/etc/system/default. To
# set custom configurations, place a savedsearches.conf in
# $SPLUNK_HOME/etc/system/local/. For examples, see
# savedsearches.conf.example. You must restart Splunk to enable configurations.
#
# To learn more about configuration files (including precedence) please see the
# documentation located at
# http://docs.splunk.com/Documentation/Splunk/latest/Admin/Aboutconfigurationfiles
```

### 全局设置

```
# GLOBAL SETTINGS
# Use the [default] stanza to define any global settings.
# * You can also define global settings outside of any stanza, at the top of
#   the file.
# * Each conf file should have at most one default stanza. If there are
#   multiple default stanzas, attributes are combined. In the case of multiple
#   definitions of the same attribute, the last definition in the file wins.
# * If an attribute is defined at both the global level and in a specific
#   stanza, the value in the specific stanza takes precedence.
```

### savedsearches.conf 可能的属性/值对有：

```
*****
# The possible attribute/value pairs for savedsearches.conf are:
*****The possible attribute/value pairs for savedsearches.conf are:

[<stanza name>]
* Create a unique stanza name for each saved search.
* Follow the stanza name with any number of the following attribute/value
  pairs.
* If you do not specify an attribute, Splunk uses the default.

disabled = [0|1]
* Disable your search by setting to 1.
* A disabled search cannot run until it is enabled.
* This setting is typically used to keep a scheduled search from running on
  its schedule without deleting the search definition.
* Defaults to 0.

search = <string>
* Actual search terms of the saved search.
* For example, search = index::sampledata http NOT 500.
* Your search can include macro searches for substitution.
* To learn more about creating a macro search, search the documentation for
  "macro search."
* Multi-line search strings currently have some limitations. For example use
  with the search command '|savedsearch' does not currently work with multi-line
  search strings.
```

\* Defaults to empty string.

dispatchAs = [user|owner]

\* When the saved search is dispatched via the "saved/searches/{name}/dispatch" endpoint, this setting controls, what user that search is dispatched as.

\* This setting is only meaningful for shared saved searches.

\* When dispatched as user it will be executed as if the requesting user owned the search.

\* When dispatched as owner it will be executed as if the owner of the search dispatched it no matter what user requested it.

\* If the 'force\_saved\_search\_dispatch\_as\_user' attribute, in the limits.conf file, is set to true then the dispatchAs attribute is reset to 'user' while the saved search is dispatching.

\* Defaults to owner.

## 计划选项

\*\*\*\*\*

# Scheduling options

\*\*\*\*\*Scheduling options

enableSched = [0|1]

\* Set this to 1 to run your search on a schedule.

\* Defaults to 0.

cron\_schedule = <cron string>

\* The cron schedule used to execute this search.

\* For example: \*/5 \* \* \* \* causes the search to execute every 5 minutes.

\* Cron lets you use standard cron notation to define your scheduled search interval.

In particular, cron can accept this type of notation: 00,20,40 \* \* \* \*, which runs the search every hour at hh:00, hh:20, hh:40. Along the same lines, a cron of 03,23,43 \* \* \* \* runs the search every hour at hh:03, hh:23, hh:43.

\* Splunk recommends that you schedule your searches so that they are staggered over time. This reduces system load. Running all of them every 20 minutes (\*/\*20) means they would all launch at hh:00 (20, 40) and might slow your system every 20 minutes.

\* Splunk's cron implementation does not currently support names of months/days.

\* Defaults to empty string.

schedule = <cron-style string>

\* This field is DEPRECATED as of 4.0.

\* For more information, see the pre-4.0 spec file.

\* Use cron\_schedule to define your scheduled search interval.

max\_concurrent = <unsigned int>

\* The maximum number of concurrent instances of this search the scheduler is allowed to run.

\* Defaults to 1.

realtime\_schedule = [0|1]

\* Controls the way the scheduler computes the next execution time of a scheduled search.

\* If this value is set to 1, the scheduler bases its determination of the next scheduled search execution time on the current time.

\* If this value is set to 0, the scheduler bases its determination of the next scheduled search on the last search execution time. This is called continuous scheduling.

\* If set to 1, the scheduler might skip some execution periods to make sure that the scheduler is executing the searches running over the most recent time range.

\* If set to 0, the scheduler never skips scheduled execution periods.

\* However, the execution

of the saved search might fall behind depending on the scheduler's load.

Use continuous scheduling whenever you enable the summary index option.

\* The scheduler tries to execute searches that have realtime\_schedule set to 1 before it executes searches that have continuous scheduling (realtime\_schedule = 0).

```

* Defaults to 1

schedule_priority = default | higher | highest
* Raises scheduling priority of a search:
+ "default": No scheduling priority increase.
+ "higher": Scheduling priority is higher than other searches of the same
  scheduling tier. While there are four tiers of priority for scheduled
  searches, only the following are affected by this property:
  1. Real-Time-Scheduled (realtime_schedule=1).
  2. Continuous-Scheduled (realtime_schedule=0).
+ "highest": Scheduling priority is higher than other searches regardless of
  scheduling tier. However, real-time-scheduled searches with priority =
  highest always have priority over continuous scheduled searches with
  priority = highest.
+ Hence, the high-to-low order (where RTSS = real-time-scheduled search,
  CSS = continuous-scheduled search, d = default, h = higher, H = highest)
  is: RTSS(H) > CSS(H) > RTSS(h) > RTSS(d) > CSS(h) > CSS(d)
* The scheduler honors a non-default priority only when the search owner has
  the 'edit_search_schedule_priority' capability.
* Defaults to "default".
* A non-default priority is mutually exclusive with a non-zero 'schedule_window'
  (see below). If a user specifies both for a scheduled search, the scheduler
  honors the priority only.
* However, if a user specifies both settings for a search, but the search owner
  does not have the 'edit_search_scheduler_priority' capability, then the
  scheduler ignores the priority setting and honors the 'schedule_window'.
* WARNING: Having too many searches with a non-default priority will impede the
  ability of the scheduler to minimize search starvation. Use this setting
  only for mission-critical searches.

schedule_window = <unsigned int> | auto
* When schedule_window is non-zero, it indicates to the scheduler that the
  search does not require a precise start time. This gives the scheduler
  greater flexibility when it prioritizes searches.
* When schedule_window is set to an integer greater than 0, it specifies the
  "window" of time (in minutes) a search may start within.
+ The schedule_window must be shorter than the period of the search.
+ Schedule windows are not recommended for searches that run every minute.
* When set to 0, there is no schedule window. The scheduler starts the search
  as close to its scheduled time as possible.
* When set to "auto," the scheduler calculates the schedule_window value
  automatically.
+ For more information about this calculation, see the search scheduler
  documentation.
* Defaults to 0 for searches that are owned by users with the
  edit_search_schedule_window capability. For such searches, this value can be
  changed.
* Defaults to "auto" for searches that are owned by users that do not have the
  edit_search_schedule_window capability. For such searches, this setting cannot be
  changed.
* A non-zero schedule_window is mutually exclusive with a non-default
  schedule_priority (see schedule_priority for details).

```

## 通知选项

```

#*****
# Notification options
#*****Notification options

counttype = number of events | number of hosts | number of sources | always
* Set the type of count for alerting.
* Used with relation and quantity (below).
* NOTE: If you specify "always," do not set relation or quantity (below).
* Defaults to always.

relation = greater than | less than | equal to | not equal to | drops by | rises by
* Specifies how to compare against counttype.
* Defaults to empty string.

```

```

quantity = <integer>
* Specifies a value for the counttype and relation, to determine the condition
  under which an alert is triggered by a saved search.
* You can think of it as a sentence constructed like this: <counttype> <relation> <quantity>.
* For example, "number of events [is] greater than 10" sends an alert when the
  count of events is larger than by 10.
* For example, "number of events drops by 10%" sends an alert when the count of
  events drops by 10%.
* Defaults to an empty string.

alert_condition = <search string>
* Contains a conditional search that is evaluated against the results of the
  saved search. Alerts are triggered if the specified search yields a
  non-empty search result list.
* NOTE: If you specify an alert_condition, do not set counttype, relation, or
  quantity.
* Defaults to an empty string.

#*****
# generic action settings.
# For a comprehensive list of actions and their arguments, refer to
# alert_actions.conf.
#*****

action.<action_name> = 0 | 1
* Indicates whether the action is enabled or disabled for a particular saved
  search.
* The action_name can be: email | populate_lookup | script | summary_index
* For more about your defined alert actions see alert_actions.conf.
* Defaults to an empty string.

action.<action_name>.<parameter> = <value>
* Overrides an action's parameter (defined in alert_actions.conf) with a new
  <value> for this saved search only.
* Defaults to an empty string.

```

## 电子邮件操作设置

```

#*****
# Settings for email action
#*****Settings for email action

action.email = 0 | 1
* Enables or disables the email action.
* Defaults to 0.

action.email.to = <email list>
* REQUIRED. This setting is not defined in alert_actions.conf.
* Set a comma-delimited list of recipient email addresses.
* Defaults to empty string.

* When configured in Splunk Web, the following email settings
  are written to this conf file only if their values differ
  from settings in alert_actions.conf.

action.email.from = <email address>
* Set an email address to use as the sender's address.
* Defaults to splunk@<LOCALHOST> (or whatever is set in alert_actions.conf).

action.email.subject = <string>
* Set the subject of the email delivered to recipients.
* Defaults to SplunkAlert-<savedsearchname> (or whatever is set
  in alert_actions.conf).

action.email.mailserver = <string>
* Set the address of the MTA server to be used to send the emails.
* Defaults to <LOCALHOST> (or whatever is set in alert_actions.conf).

```

```

action.email.maxresults = <integer>
* Set the maximum number of results to be emailed.
* Any alert-level results threshold greater than this number will be capped at
  this level.
* This value affects all methods of result inclusion by email alert: inline,
  CSV and PDF.
* Note that this setting is affected globally by "maxresults" in the [email]
  stanza of alert_actions.conf.
* Defaults to 10000

action.email.include.results_link = [1|0]
* Specify whether to include a link to search results in the
  alert notification email.
* Defaults to 1 (or whatever is set in alert_actions.conf).

action.email.include.search = [1|0]
* Specify whether to include the query whose results triggered the email.
* Defaults to 0 (or whatever is set in alert_actions.conf).

action.email.include.trigger = [1|0]
* Specify whether to include the alert trigger condition.
* Defaults to 0 (or whatever is set in alert_actions.conf).

action.email.include.trigger_time = [1|0]
* Specify whether to include the alert trigger time.
* Defaults to 0 (or whatever is set in alert_actions.conf).

action.email.include.view_link = [1|0]
* Specify whether to include saved search title and a link for editing
  the saved search.
* Defaults to 1 (or whatever is set in alert_actions.conf).

action.email.inline = [1|0]
* Specify whether to include search results in the body of the
  alert notification email.
* Defaults to 0 (or whatever is set in alert_actions.conf).

action.email.sendcsv = [1|0]
* Specify whether to send results as a CSV file.
* Defaults to 0 (or whatever is set in alert_actions.conf).

action.email.sendpdf = [1|0]
* Specify whether to send results as a PDF file.
* Defaults to 0 (or whatever is set in alert_actions.conf).

action.email.sendresults = [1|0]
* Specify whether to include search results in the
  alert notification email.
* Defaults to 0 (or whatever is set in alert_actions.conf).

```

## 脚本操作设置

```

#*****
# Settings for script action
#*****Settings for script action

action.script = 0 | 1
* Enables or disables the script action.
* 1 to enable, 0 to disable.
* Defaults to 0

action.script.filename = <script filename>
* The filename, with no path, of the shell script to execute.
* The script should be located in: $SPLUNK_HOME/bin/scripts/
* For system shell scripts on Unix, or .bat or .cmd on windows, there
  are no further requirements.
* For other types of scripts, the first line should begin with a #!
  marker, followed by a path to the interpreter that will run the
  script.

```

```
* Example: #!C:\Python27\python.exe
* Defaults to empty string.
```

## 摘要索引操作设置

```
*****
# Settings for summary index action
*****Settings for summary index action

action.summary_index = 0 | 1
* Enables or disables the summary index action.
* Defaults to 0.

action.summary_index._name = <index>
* Specifies the name of the summary index where the results of the scheduled
  search are saved.
* Defaults to summary.

action.summary_index.inline = <bool>
* Determines whether to execute the summary indexing action as part of the
  scheduled search.
* NOTE: This option is considered only if the summary index action is enabled
  and is always executed (in other words, if counttype = always).
* Defaults to true.

action.summary_index.<field> = <string>
* Specifies a field/value pair to add to every event that gets summary indexed
  by this search.
* You can define multiple field/value pairs for a single summary index search.
```

## 查找表填充参数设置

```
*****
# Settings for lookup table population parameters
*****Settings for lookup table population parameters

action.populate_lookup = 0 | 1
* Enables or disables the lookup population action.
* Defaults to 0.

action.populate_lookup.dest = <string>
* Can be one of the following two options:
  * A lookup name from transforms.conf.
  * A path to a lookup .csv file that Splunk should copy the search results to,
    relative to $SPLUNK_HOME.
  * NOTE: This path must point to a .csv file in either of the following
    directories:
    * etc/system/lookups/
    * etc/apps/<app-name>/lookups
  * NOTE: the destination directories of the above files must already exist
* Defaults to empty string.

run_on_startup = true | false
* Toggles whether this search runs when Splunk starts or any edit that changes
  search related args happen (which includes: search and dispatch.* args).
* If set to true the search is ran as soon as possible during startup or after
  edit otherwise the search is ran at the next scheduled time.
* We recommend that you set run_on_startup to true for scheduled searches that
  populate lookup tables or generate artifacts used by dashboards.
* Defaults to false.

run_n_times = <unsigned int>
* Runs this search exactly the given number of times, then never again (until
  Splunk is restarted).
* Defaults to 0 (infinite).
```



## Dispatch 搜索选项

```
#*****
# dispatch search options
#*****dispatch search options

dispatch.ttl = <integer>[p]
* Indicates the time to live (in seconds) for the artifacts of the scheduled
  search, if no actions are triggered.
* If the integer is followed by the letter 'p' Splunk interprets the ttl as a
  multiple of the scheduled search's execution period (e.g. if the search is
  scheduled to run hourly and ttl is set to 2p the ttl of the artifacts will be
  set to 2 hours).
* If an action is triggered Splunk changes the ttl to that action's ttl. If
  multiple actions are triggered, Splunk applies the largest action ttl to the
  artifacts. To set the action's ttl, refer to alert_actions.conf.spec.
* For more info on search's ttl please see limits.conf.spec [search] ttl
* Defaults to 2p (that is, 2 x the period of the scheduled search).

dispatch.buckets = <integer>
* The maximum number of timeline buckets.
* Defaults to 0.

dispatch.max_count = <integer>
* The maximum number of results before finalizing the search.
* Defaults to 500000.

dispatch.max_time = <integer>
* Indicates the maximum amount of time (in seconds) before finalizing the
  search.
* Defaults to 0.

dispatch.lookups = 1| 0
* Enables or disables lookups for this search.
* Defaults to 1.

dispatch.earliest_time = <time-str>
* Specifies the earliest time for this search. Can be a relative or absolute
  time.
* If this value is an absolute time, use the dispatch.time_format to format the
  value.
* Defaults to empty string.

dispatch.latest_time = <time-str>
* Specifies the latest time for this saved search. Can be a relative or
  absolute time.
* If this value is an absolute time, use the dispatch.time_format to format the
  value.
* Defaults to empty string.

dispatch.index_earliest= <time-str>
* Specifies the earliest index time for this search. Can be a relative or
  absolute time.
* If this value is an absolute time, use the dispatch.time_format to format the
  value.
* Defaults to empty string.

dispatch.index_latest= <time-str>
* Specifies the latest index time for this saved search. Can be a relative or
  absolute time.
* If this value is an absolute time, use the dispatch.time_format to format the
  value.
* Defaults to empty string.

dispatch.time_format = <time format str>
* Defines the time format that Splunk uses to specify the earliest and latest
  time.
* Defaults to %FT%T.%Q%:z
```

```

dispatch.spawn_process = 1 | 0
* Specifies whether Splunk spawns a new search process when this saved search
  is executed.
* Default is 1.

dispatch.auto_cancel = <int>
* If specified, the job automatically cancels after this many seconds of
  inactivity. (0 means never auto-cancel)
* Default is 0.

dispatch.auto_pause = <int>
* If specified, the search job pauses after this many seconds of inactivity. (0
  means never auto-pause.)
* To restart a paused search job, specify unpause as an action to POST
  search/jobs/{search_id}/control.
* auto_pause only goes into effect once. Unpausing after auto_pause does not
  put auto_pause into effect again.
* Default is 0.

dispatch.reduce_freq = <int>
* Specifies how frequently Splunk should run the MapReduce reduce phase on
  accumulated map values.
* Defaults to 10.

dispatch.rt_backfill = <bool>
* Specifies whether to do real-time window backfilling for scheduled real time
  searches
* Defaults to false.

dispatch.indexedRealtime = <bool>
* Specifies whether to use indexed-realtime mode when doing realtime searches.
* Default for saved searches is "unset" falling back to limits.conf setting [realtime]
indexed_realtime_use_by_default

dispatch.indexedRealtimeOffset = <int>
* Allows for a per-job override of limits.conf settting [realtime] indexed_realtime_disk_sync_delay
* Default for saved searches is "unset" falling back to limits.conf setting.

dispatch.indexedRealtimeMinSpan = <int>
* Allows for a per-job override of limits.conf settting [realtime] indexed_realtime_default_span
* Default for saved searches is "unset" falling back to limits.conf setting.

dispatch.rt_maximum_span = <int>
* Allows for a per-job override of limits.conf settting [realtime] indexed_realtime_maximum_span
* Default for saved searches is "unset" falling back to limits.conf setting.

dispatch.sample_ratio = <int>
* The integer value used to calculate the sample ratio. The formula is 1 / <int>.
* The sample ratio specifies the likelihood of any event being included in the sample.
* For example, if sample_ratio = 500 each event has a 1/500 chance of being included in the sample result set.
* Defaults to 1.

restart_on_searchpeer_add = 1 | 0
* Specifies whether to restart a real-time search managed by the scheduler when
  a search peer becomes available for this saved search.
* NOTE: The peer can be a newly added peer or a peer that has been down and has
  become available.
* Defaults to 1.

```

## 自动摘要选项

```

#*****
# auto summarization options
#*****auto summarization options
auto_summarize = <bool>
* Whether the scheduler should ensure that the data for this search is
  automatically summarized
* Defaults to false.

```

```

auto_summarize.command = <string>
* A search template to be used to construct the auto summarization for this
  search.
* DO NOT change unless you know what you're doing

auto_summarize.timespan = <time-specifier> (, <time-specifier>)*
* Comma delimited list of time ranges that each summarized chunk should span.
  This comprises the list of available granularity levels for which summaries
  would be available. For example a timechart over the last month whose
  granularity is at the day level should set this to 1d. If you are going to need
  the same data summarized at the hour level because you need to have weekly
  charts then use: 1h;1d

auto_summarize.cron_schedule = <cron-string>
* Cron schedule to be used to probe/generate the summaries for this search

auto_summarize.dispatch.<arg-name> = <string>
* Any dispatch.* options that need to be overridden when running the summary
  search.

auto_summarize.suspend_period      = <time-specifier>
* Amount of time to suspend summarization of this search if the summarization
  is deemed unhelpful
* Defaults to 24h

auto_summarize.max_summary_size    = <unsigned int>
* The minimum summary size when to start testing it's helpfulness
* Defaults to 52428800 (5MB)

auto_summarize.max_summary_ratio   = <positive float>
* The maximum ratio of summary_size/bucket_size when to stop summarization and
  deem it unhelpful for a bucket
* NOTE: the test is only performed if the summary size is larger
  than auto_summarize.max_summary_size
* Defaults to: 0.1

auto_summarize.max_disabled_buckets = <unsigned int>
* The maximum number of buckets with the suspended summarization before the
  summarization search is completely stopped and the summarization of the
  search is suspended for auto_summarize.suspend_period
* Defaults to: 2

auto_summarize.max_time = <unsigned int>
* The maximum amount of time that the summary search is allowed to run. Note
  that this is an approximate time and the summarize search will be stopped at
  clean bucket boundaries.
* Defaults to: 3600

auto_summarize.hash = <string>
auto_summarize.normalized_hash = <string>
* These are auto generated settings.

auto_summarize.max_concurrent = <unsigned int>
* The maximum number of concurrent instances of this auto summarizing search,
  that the scheduler is allowed to run.
* Defaults to: 1

```

## 告警抑制/严重性/失效/追踪/查看设置

```

#*****
# alert suppression/severity/expiration/tracking/viewing settings
#*****alert suppression/severity/expiration/tracking/viewing settings

alert.suppress = 0 | 1
* Specifies whether alert suppression is enabled for this scheduled search.
* Defaults to 0.

alert.suppress.period = <time-specifier>
* Sets the suppression period. Use [number][time-unit] to specify a time.
* For example: 60 = 60 seconds, 1m = 1 minute, 1h = 60 minutes = 1 hour etc

```

```

* Honored if and only if alert.suppress = 1
* Defaults to empty string.

alert.suppress.fields = <comma-delimited-field-list>
* List of fields to use when suppressing per-result alerts. This field *must*
  be specified if the digest mode is disabled and suppression is enabled.
* Defaults to empty string.

alert.severity = <int>
* Sets the alert severity level.
* Valid values are: 1-debug, 2-info, 3-warn, 4-error, 5-severe, 6-fatal
* Defaults to 3.

alert.expires = <time-specifier>
* Sets the period of time to show the alert in the dashboard. Use [number][time-unit]
  to specify a time.
* For example: 60 = 60 seconds, 1m = 1 minute, 1h = 60 minutes = 1 hour etc
* Defaults to 24h.
* This property is valid until splunkd restarts. Restart clears the listing of
  triggered alerts.

alert.digest_mode = true | false
* Specifies whether Splunk applies the alert actions to the entire result set
  or on each individual result.
* Defaults to true.

alert.track = true | false | auto
* Specifies whether to track the actions triggered by this scheduled search.
* auto - determine whether to track or not based on the tracking setting of
  each action, do not track scheduled searches that always trigger actions.
* true - force alert tracking.
* false - disable alert tracking for this search.
* Defaults to auto.

alert.display_view = <string>
* Name of the UI view where the emailed link for per result alerts should point to.
* If not specified, the value of request.ui_dispatch_app will be used, if that
  is missing then "search" will be used
* Defaults to empty string

```

## 特定于 UI 的设置

```

#*****
# UI-specific settings
#*****UI-specific settings

displayview=<string>
* Defines the default UI view name (not label) in which to load the results.
* Accessibility is subject to the user having sufficient permissions.
* Defaults to empty string.

vsid = <string>
* Defines the viewstate id associated with the UI view listed in 'displayview'.
* Must match up to a stanza in viewstates.conf.
* Defaults to empty string.

is_visible = true | false
* Specifies whether this saved search should be listed in the visible saved
  search list.
* Defaults to true.

description = <string>
* Human-readable description of this saved search.
* Defaults to empty string.

request.ui_dispatch_app = <string>
* Specifies a field used by Splunk UI to denote the app this search should be
  dispatched in.
* Defaults to empty string.

```

```
request.ui_dispatch_view = <string>
* Specifies a field used by Splunk UI to denote the view this search should be
  displayed in.
* Defaults to empty string.
```

## 显示格式选项

```
#*****
# Display Formatting Options
#*****Display Formatting Options

# General options
display.general.enablePreview = 0 | 1
display.general.type = [events|statistics|visualizations]
display.general.timeRangePicker.show = 0 | 1
display.general.migratedFromViewState = 0 | 1
display.general.locale = <string>

# Event options
display.events.fields = [<string>(, <string>)*]
display.events.type = [raw|list|table]
display.events.rowNumbers = 0 | 1
display.events.maxLines = <int>
display.events.raw.drilldown = [inner|outer|full|none]
display.events.list.drilldown = [inner|outer|full|none]
display.events.list.wrap = 0 | 1
display.events.table.drilldown = 0 | 1
display.events.table.wrap = 0 | 1

# Statistics options
display.statistics.rowNumbers = 0 | 1
display.statistics.wrap = 0 | 1
display.statistics.overlay = [none|heatmap|highlow]
display.statistics.drilldown = [row|cell|none]
display.statistics.totalsRow = 0 | 1
display.statistics.percentagesRow = 0 | 1
display.statistics.show = 0 | 1

# Visualization options
display.visualizations.show = 0 | 1
display.visualizations.type = [charting|singlevalue|mapping|custom]
display.visualizations.chartHeight = <int>
display.visualizations.charting.chart =
[line|area|column|bar|pie|scatter|bubble|radialGauge|fillerGauge|markerGauge]
display.visualizations.charting.chart.stackMode = [default|stacked|stacked100]
display.visualizations.charting.chart.nullValueMode = [gaps|zero|connect]
display.visualizations.charting.chart.overlayFields = <string>
display.visualizations.charting.drilldown = [all|none]
display.visualizations.charting.chart.style = [minimal|shiny]
display.visualizations.charting.layout.splitSeries = 0 | 1
display.visualizations.charting.layout.splitSeries.allowIndependentYRanges = 0 | 1
display.visualizations.charting.legend.placement = [right|bottom|top|left|none]
display.visualizations.charting.legend.labelStyle.overflowMode = [ellipsisEnd|ellipsisMiddle|ellipsisStart]
display.visualizations.charting.axisTitleX.text = <string>
display.visualizations.charting.axisTitleY.text = <string>
display.visualizations.charting.axisTitleY2.text = <string>
display.visualizations.charting.axisTitleX.visibility = [visible|collapsed]
display.visualizations.charting.axisTitleY.visibility = [visible|collapsed]
display.visualizations.charting.axisTitleY2.visibility = [visible|collapsed]
display.visualizations.charting.axisX.scale = linear|log
display.visualizations.charting.axisY.scale = linear|log
display.visualizations.charting.axisY2.scale = linear|log|inherit
display.visualizations.charting.axisLabelsX.majorLabelStyle.overflowMode = [ellipsisMiddle|ellipsisNone]
display.visualizations.charting.axisLabelsX.majorLabelStyle.rotation = [-90|-45|0|45|90]
display.visualizations.charting.axisLabelsX.majorUnit = <float> | auto
display.visualizations.charting.axisLabelsY.majorUnit = <float> | auto
display.visualizations.charting.axisLabelsY2.majorUnit = <float> | auto
display.visualizations.charting.axisX.minimumNumber = <float> | auto
display.visualizations.charting.axisY.minimumNumber = <float> | auto
display.visualizations.charting.axisY2.minimumNumber = <float> | auto
```

```

display.visualizations.charting.axisX.maximumNumber = <float> | auto
display.visualizations.charting.axisY.maximumNumber = <float> | auto
display.visualizations.charting.axisY2.maximumNumber = <float> | auto
display.visualizations.charting.axisY2.enabled = 0 | 1
display.visualizations.charting.chart.sliceCollapsingThreshold = <float>
display.visualizations.charting.chart.showDataLabels = [all|none|minmax]
display.visualizations.charting.gaugeColors = [<hex>(, <hex>)*]
display.visualizations.charting.chart.rangeValues = [<string>(, <string>)*]
display.visualizations.charting.chart.bubbleMaximumSize = <int>
display.visualizations.charting.chart.bubbleMinimumSize = <int>
display.visualizations.charting.chart.bubbleSizeBy = [area|diameter]
display.visualizations.custom.type = <string>
display.visualizations.custom.height = <int>
display.visualizations.singlevalue.height = <int>
display.visualizations.singlevalue.beforeLabel = <string>
display.visualizations.singlevalue.afterLabel = <string>
display.visualizations.singlevalue.underLabel = <string>
display.visualizations.singlevalue.unit = <string>
display.visualizations.singlevalue.unitPosition = [before|after]
display.visualizations.singlevalue.drilldown = [all|none]
display.visualizations.singlevalue.colorMode = [block|none]
display.visualizations.singlevalue.rangeValues = [<string>(, <string>)*]
display.visualizations.singlevalue.rangeColors = [<string>(, <string>)*]
display.visualizations.singlevalue.trendInterval = <string>
display.visualizations.singlevalue.trendColorInterpretation = [standard|inverse]
display.visualizations.singlevalue.showTrendIndicator = 0 | 1
display.visualizations.singlevalue.showSparkline = 0 | 1
display.visualizations.singlevalue.trendDisplayMode = [percent|absolute]
display.visualizations.singlevalue.colorBy = [value|trend]
display.visualizations.singlevalue.useColors = 0 | 1
display.visualizations.singlevalue.numberPrecision = [0|0.0|0.00|0.000|0.0000]
display.visualizations.singlevalue.useThousandSeparators = 0 | 1
display.visualizations.mapHeight = <int>
display.visualizations.mapping.type = [marker|choropleth]
display.visualizations.mapping.drilldown = [all|none]
display.visualizations.mapping.map.center = (<float>,<float>)
display.visualizations.mapping.map.zoom = <int>
display.visualizations.mapping.map.scrollZoom = 0 | 1
display.visualizations.mapping.map.panning = 0 | 1
display.visualizations.mapping.choroplethLayer.colorMode = [auto|sequential|divergent|categorical]
display.visualizations.mapping.choroplethLayer.maximumColor = <string>
display.visualizations.mapping.choroplethLayer.minimumColor = <string>
display.visualizations.mapping.choroplethLayer.colorBins = <int>
display.visualizations.mapping.choroplethLayer.neutralPoint = <float>
display.visualizations.mapping.choroplethLayer.shapeOpacity = <float>
display.visualizations.mapping.choroplethLayer.showBorder = 0 | 1
display.visualizations.mapping.markerLayer.markerOpacity = <float>
display.visualizations.mapping.markerLayer.markerMinSize = <int>
display.visualizations.mapping.markerLayer.markerMaxSize = <int>
display.visualizations.mapping.data.maxClusters = <int>
display.visualizations.mapping.showTiles = 0 | 1
display.visualizations.mapping.tileLayer.tileOpacity = <float>
display.visualizations.mapping.tileLayer.url = <string>
display.visualizations.mapping.tileLayer.minZoom = <int>
display.visualizations.mapping.tileLayer.maxZoom = <int>

# Patterns options
display.page.search.patterns.sensitivity = <float>

# Page options
display.page.search.mode = [fast|smart|verbose]
display.page.search.timeline.format = [hidden|compact|full]
display.page.search.timeline.scale = [linear|log]
display.page.search.showFields = 0 | 1
display.page.search.tab = [events|statistics|visualizations|patterns]
# Deprecated
display.page.pivot.dataModel = <string>

```

## 表格格式设置

```

#*****
# Table format settings
#*****Table format settings

# Format options
display.statistics.format.<index> = [color|number]
display.statistics.format.<index>.field = <string>
display.statistics.format.<index>.fields = [<string>(, <string>)*]

# Color format options
display.statistics.format.<index>.scale = [category|linear|log|minMidMax|sharedCategory|threshold]
display.statistics.format.<index>.colorPalette = [expression|list|map|minMidMax|sharedList]

# Number format options
display.statistics.format.<index>.precision = <int>
display.statistics.format.<index>.useThousandSeparators = <bool>
display.statistics.format.<index>.unit = <string>
display.statistics.format.<index>.unitPosition = [before|after]

# Scale options for 'category'
display.statistics.format.<index>.scale.categories = [<string>(, <string>)*]

# Scale options for 'log'
display.statistics.format.<index>.scale.base = <int>

# Scale options for 'minMidMax'
display.statistics.format.<index>.scale.minType = [number|percent|percentile]
display.statistics.format.<index>.scale.minValue = <float>
display.statistics.format.<index>.scale.midType = [number|percent|percentile]
display.statistics.format.<index>.scale.midValue = <float>
display.statistics.format.<index>.scale.maxType = [number|percent|percentile]
display.statistics.format.<index>.scale.maxValue = <float>

# Scale options for 'threshold'
display.statistics.format.<index>.scale.thresholds = [<float>(, <float>)*]

# Color palette options for 'expression'
display.statistics.format.<index>.colorPalette.rule = <string>

# Color palette options for 'list'
display.statistics.format.<index>.colorPalette.colors = [<hex>(, <hex>)*]
display.statistics.format.<index>.colorPalette.interpolate = <bool>

# Color palette options for 'map'
display.statistics.format.<index>.colorPalette.colors = {<string>:<hex>(, <string>:<hex>)*}

# Color palette options for 'minMidMax'
display.statistics.format.<index>.colorPalette.minColor = <hex>
display.statistics.format.<index>.colorPalette.midColor = <hex>
display.statistics.format.<index>.colorPalette.maxColor = <hex>

```

## 其他设置

```

#*****
# Other settings
#*****Other settings

embed.enabled = 0 | 1
* Specifies whether a saved search is shared for access with a guestpass.
* Search artifacts of a search can be viewed via a guestpass only if:
  * A token has been generated that is associated with this saved search.
    The token is associated with a particular user and app context.
  * The user to whom the token belongs has permissions to view that search.
  * The saved search has been scheduled and there are artifacts available.
    Only artifacts are available via guestpass: we never dispatch a search.
  * The save search is not disabled, it is scheduled, it is not real-time,
    and it is not an alert.

```

## 弃用设置

```

#*****
# deprecated settings
#*****deprecated settings

sendresults = <bool>
* use action.email.sendresult

action_rss = <bool>
* use action.rss

action_email = <string>
* use action.email and action.email.to

role = <string>
* see saved search permissions

userid = <string>
* see saved search permissions

query = <string>
* use search

nextrun = <int>
* not used anymore, the scheduler maintains this info internally

qualifiedSearch = <string>
* not used anymore, the Splunk software computes this value during runtime

```

## savedsearches.conf.example

```

# Version 6.5.0
#
# This file contains example saved searches and alerts.
#
# To use one or more of these configurations, copy the configuration block into
# savedsearches.conf in $SPLUNK_HOME/etc/system/local/. You must restart Splunk
# to enable configurations.
#
# To learn more about configuration files (including precedence) please see the
# documentation located at
# http://docs.splunk.com/Documentation/Splunk/latest/Admin/Aboutconfigurationfiles

# The following searches are example searches. To create your own search,
# modify the values by following the spec outlined in savedsearches.conf.spec.

[Daily indexing volume by server]
search = index=_internal todaysBytesIndexed LicenseManager-Audit NOT source=*web_service.log NOT
source=*web_access.log | eval Daily
_Indexing_Volume_in_MB = todaysBytesIndexed/1024/1024 | timechart avg(Daily_Indexing_Volume_in_MB) by host
dispatch.earliest_time = -7d

[Errors in the last 24 hours]
search = error OR failed OR severe OR ( sourcetype=access_* ( 404 OR 500 OR 503 ) )
dispatch.earliest_time = -1d

[Errors in the last hour]
search = error OR failed OR severe OR ( sourcetype=access_* ( 404 OR 500 OR 503 ) )
dispatch.earliest_time = -1h

[KB indexed per hour last 24 hours]
search = index=_internal metrics group=per_index_thruput NOT debug NOT sourcetype=splunk_web_access | timechart
fixedrange=t span=1h
sum(kb) | rename sum(kb) as totalKB
dispatch.earliest_time = -1d

[Messages by minute last 3 hours]

```



```
search = index=_internal eps "group=per_source_thruput" NOT filetracker | eval events=eps*kb/kbps | timechart
fixedrange=t span=1m s
um(events) by series
dispatch.earliest_time = -3h

[Splunk errors last 24 hours]
search = index=_internal " error " NOT debug source=*/splunkd.log*
dispatch.earliest_time = -24h
```

## searchbnf.conf

以下为 searchbnf.conf 的规范和示例文件。

### searchbnf.conf.spec

```
# Version 6.5.0
#
#
# This file contain descriptions of stanzas and attribute/value pairs for
# configuring search-assistant via searchbnf.conf
#
# There is a searchbnf.conf in $SPLUNK_HOME/etc/system/default/. It should
# not be modified. If your application has its own custom python search
# commands, your application can include its own searchbnf.conf to describe
# the commands to the search-assistant.
#
# To learn more about configuration files (including precedence) please see
# the documentation located at
# http://docs.splunk.com/Documentation/Splunk/latest/Admin/Aboutconfigurationfiles
```

### 全局设置

```
# GLOBAL SETTINGS
# Use the [default] stanza to define any global settings.
# * You can also define global settings outside of any stanza, at the top
#   of the file.
# * Each conf file should have at most one default stanza. If there are
#   multiple default stanzas, attributes are combined. In the case of
#   multiple definitions of the same attribute, the last definition in the
#   file wins.
# * If an attribute is defined at both the global level and in a specific
#   stanza, the value in the specific stanza takes precedence.
```

### [<search-commandname>-command]

```
[<search-commandname>-command]
* This stanza enables properties for a given <search-command>.
* A searchbnf.conf file can contain multiple stanzas for any number of
  commands. * Follow this stanza name with any number of the following
  attribute/value pairs.
* If you do not set an attribute for a given <spec>, the default is used.
  The default values are empty.
* An example stanza name might be "geocode-command", for a "geocode"
  command.
* Search command stanzas can refer to definitions defined in others stanzas,
  and they do not require "-command", appended to them. For example:
```

### [geocode-command]

```
[geocode-command]
  syntax = geocode <geocode-option>*
  ...
```

## **[geocode-option]**

```
[geocode-option]
  syntax = (maxcount=<int>) | (maxhops=<int>)
  ...

#*****
# The possible attributes/value pairs for searchbnf.conf
#*****

SYNTAX = <string>
* Describes the syntax of the search command. See the head of
  searchbnf.conf for details.
* Required

SIMPLESYNTAX = <string>

* Optional simpler version of the syntax to make it easier to
  understand at the expense of completeness. Typically it removes
  rarely used options or alternate ways of saying the same thing.
* For example, a search command might accept values such as
  "m|min|mins|minute|minutes", but that would unnecessarily
  clutter the syntax description for the user. In this case, the
  simplesyntax can just pick the one (e.g., "minute").

ALIAS = <commands list>
* Alternative names for the search command. This further cleans
  up the syntax so the user does not have to know that
  'savedsearch' can also be called by 'macro' or 'savedsplunk'.

DESCRIPTION = <string>
* Detailed text description of search command. Description can continue on
  the next line if the line ends in "\"
* Required

SHORTDESC = <string>
* A short description of the search command. The full DESCRIPTION
  may take up too much screen real-estate for the search assistant.
* Required

EXAMPLE = <string>
COMMENT = <string>
* 'example' should list out a helpful example of using the search
  command, and 'comment' should describe that example.
* 'example' and 'comment' can be appended with matching indexes to
  allow multiple examples and corresponding comments.
* For example:
  example2 = geocode maxcount=4
  command2 = run geocode on up to four values
  example3 = geocode maxcount=-1
  comment3 = run geocode on all values

USAGE = public|private|deprecated
* Determines if a command is public, private, deprecated. The
  search assistant only operates on public commands.
* Required

TAGS = <tags list>
* List of tags that describe this search command. Used to find
  commands when the user enters a synonym (e.g. "graph" -> "chart")

RELATED = <commands list>
* List of related commands to help user when using one command to
  learn about others.

#*****
# Optional attributes primarily used internally at Splunk
#*****
```

maintainer, appears-in, note, supports-multivalue, optout-in

## searchbnf.conf.example

```
# Version 6.5.0
#
# The following are example stanzas for searchbnf.conf configurations.
#

#####
# selfjoin
#####
[selfjoin-command]
syntax = selfjoin (<selfjoin-options>)* <field-list>
shortdesc = Join results with itself.
description = Join results with itself. Must specify at least one field to join on.
usage = public
example1 = selfjoin id
comment1 = Joins results with itself on 'id' field.
related = join
tags = join combine unite

[selfjoin-options]
syntax = overwrite=<bool> | max=<int> | keepsingle=<int>
description = The selfjoin joins each result with other results that\
  have the same value for the join fields. 'overwrite' controls if\
  fields from these 'other' results should overwrite fields of the\
  result used as the basis for the join (default=true). max indicates\
  the maximum number of 'other' results each main result can join with.\
  (default = 1, 0 means no limit). 'keepsingle' controls whether or not\
  results with a unique value for the join fields (and thus no other\
  results to join with) should be retained. (default = false)
```

## segmenters.conf

以下为 segmenters.conf 的规范和示例文件。

### segmenters.conf.spec

```
# Version 6.5.0
#
# This file contains possible attribute/value pairs for configuring
# segmentation of events in segmenters.conf.
#
# There is a default segmenters.conf in $SPLUNK_HOME/etc/system/default. To set
# custom configurations, place a segmenters.conf in
# $SPLUNK_HOME/etc/system/local/. For examples, see segmenters.conf.example.
# You must restart Splunk to enable configurations.
#
# To learn more about configuration files (including precedence) please see the
# documentation located at
# http://docs.splunk.com/Documentation/Splunk/latest/Admin/Aboutconfigurationfiles
```

### 全局设置

```
# GLOBAL SETTINGS
# Use the [default] stanza to define any global settings.
#
# * You can also define global settings outside of any stanza, at the top of the file.
#
# * Each conf file should have at most one default stanza. If there are multiple default
#   stanzas, attributes are combined. In the case of multiple definitions of the same
#   attribute, the last definition in the file wins.
#
# * If an attribute is defined at both the global level and in a specific stanza, the
```

# value in the specific stanza takes precedence.

## **[<SegmenterName>]**

[<SegmenterName>]

- \* Name your stanza.
- \* Follow this stanza name with any number of the following attribute/value pairs.
- \* If you don't specify an attribute/value pair, Splunk will use the default.

MAJOR = <space separated list of breaking characters>

- \* Set major breakers.
- \* Major breakers are words, phrases or terms in your data that are surrounded by set breaking characters.
- \* By default, major breakers are set to most characters and blank spaces.
- \* Typically, major breakers are single characters.
- \* Please note: \s represents a space; \n, a newline; \r, a carriage return; and \t, a tab.
- \* Default is [ ] < > ( ) { } | ! ; , ' " \* \n \r \s \t & ? + %21 %26 %2526 %3B %7C %20 %2B %3D -- %2520 %5D %5B %3A %0A %2C %28 %29

MINOR = <space separated list of strings>

- \* Set minor breakers.
- \* In addition to the segments specified by the major breakers, for each minor breaker found, Splunk indexes the token from the last major breaker to the current minor breaker and from the last minor breaker to the current minor breaker.
- \* Default is / : = @ . - \$ # % \ \ \_

INTERMEDIATE\_MAJORS = true | false

- \* Set this to "true" if you want an IP address to appear in typeahead as a, a.b, a.b.c, a.b.c.d
- \* The typical performance hit by setting to "true" is 30%.
- \* Default is "false".

FILTER = <regular expression>

- \* If set, segmentation will only take place if the regular expression matches.
- \* Furthermore, segmentation will only take place on the first group of the matching regex.
- \* Default is empty.

LOOKAHEAD = <integer>

- \* Set how far into a given event (in characters) Splunk segments.
- \* LOOKAHEAD applied after any FILTER rules.
- \* To disable segmentation, set to 0.
- \* Defaults to -1 (read the whole event).

MINOR\_LEN = <integer>

- \* Specify how long a minor token can be.
- \* Longer minor tokens are discarded without prejudice.
- \* Defaults to -1.

MAJOR\_LEN = <integer>

- \* Specify how long a major token can be.
- \* Longer major tokens are discarded without prejudice.
- \* Defaults to -1.

MINOR\_COUNT = <integer>

- \* Specify how many minor segments to create per event.
- \* After the specified number of minor tokens have been created, later ones are discarded without prejudice.
- \* Defaults to -1.

MAJOR\_COUNT = <integer>

- \* Specify how many major segments are created per event.
- \* After the specified number of major segments have been created, later ones are discarded without prejudice.
- \* Default to -1.

## segmenters.conf.example

```
# Version 6.5.0
#
# The following are examples of segmentation configurations.
#
# To use one or more of these configurations, copy the configuration block into
# segmenters.conf in $SPLUNK_HOME/etc/system/local/. You must restart Splunk to
# enable configurations.
#
# To learn more about configuration files (including precedence) please see the
# documentation located at
# http://docs.splunk.com/Documentation/Splunk/latest/Admin/Aboutconfigurationfiles

# Example of a segmenter that doesn't index the date as segments in syslog
# data:

[syslog]
FILTER = ^.*?\d\d:\d\d:\d\d\s+\S+\s+(.*)$

# Example of a segmenter that only indexes the first 256b of events:

[limited-reach]
LOOKAHEAD = 256

# Example of a segmenter that only indexes the first line of an event:

[first-line]
FILTER = ^(.*) (\n|$)

# Turn segmentation off completely:

[no-segmentation]
LOOKAHEAD = 0
```

## server.conf

以下为 server.conf 的规范和示例文件。

### server.conf.spec

```
# Version 6.5.0
#
# This file contains the set of attributes and values you can use to
# configure server options in server.conf.
#
# There is a server.conf in $SPLUNK_HOME/etc/system/default/. To set custom
# configurations, place a server.conf in $SPLUNK_HOME/etc/system/local/.
# For examples, see server.conf.example. You must restart Splunk to enable
# configurations.
#
# To learn more about configuration files (including precedence) please see
# the documentation located at
# http://docs.splunk.com/Documentation/Splunk/latest/Admin/Aboutconfigurationfiles
```

### 全局设置

```
# GLOBAL SETTINGS
# Use the [default] stanza to define any global settings.
# * You can also define global settings outside of any stanza, at the top
# of the file.
```

```
# * Each conf file should have at most one default stanza. If there are
# multiple default stanzas, attributes are combined. In the case of
# multiple definitions of the same attribute, the last definition in the
# file wins.
# * If an attribute is defined at both the global level and in a specific
# stanza, the value in the specific stanza takes precedence.
```

## 一般服务器配置

```
#####
# General Server Configuration
#####General Server Configuration
[general]
serverName = <ASCII string>
* The name used to identify this Splunk instance for features such as
  distributed search.
* Defaults to <hostname>-<user running splunk>.
* Shall not be an empty string
* May contain environment variables
* After any environment variables have been expanded, the server name
  (if not an IPv6 address) can only contain letters, numbers, underscores,
  dots, and dashes; and it must start with a letter, number, or an
  underscore.

hostnameOption = <ASCII string>
* The option used to specify the detail in the server name used to identify
  this Splunk instance.
* Can be one of "fullyqualifiedname" , "clustername", "shortname"
* Is applicable to Windows only
* Shall not be an empty string

sessionTimeout = <nonnegative integer>[smhd]
* The amount of time before a user session times out, expressed as a
  search-like time range
* Examples include '24h' (24 hours), '3d' (3 days),
  '7200s' (7200 seconds, or two hours)
* Defaults to '1h' (1 hour)

trustedIP = <IP address>
* All logins from this IP address are trusted, meaning password is no longer
  required
* Only set this if you are using Single Sign On (SSO)

allowRemoteLogin = always|never|requireSetPassword
* Controls remote management by restricting general login. Note that this
  does not apply to trusted SSO logins from trustedIP.
* If 'always', enables authentication so that all remote login attempts are
  allowed.
* If 'never', only local logins to splunkd will be allowed. Note that this
  will still allow remote management through splunkweb if splunkweb is on
  the same server.
* If 'requireSetPassword' (default):
  * In the free license, remote login is disabled.
  * In the pro license, remote login is only disabled for "admin" user if
    default password of "admin" has not been changed.

access_logging_for_phonehome = true|false
* Enables/disables logging to splunkd_access.log for client phonehomes
* defaults to true (logging enabled)

hangup_after_phonehome = true|false
* Controls whether or not the (deployment) server hangs up the connection
  after the phonehome is done.
* By default we use persistent HTTP 1.1 connections with the server to
  handle phonehomes. This may show higher memory usage for a large number of
  clients.
* In case we have more than maximum concurrent tcp connection number of
  deployment clients, persistent connections do not help with the reuse of
  connections anyway, so setting this to false helps bring down memory
  usage.
```

```

* defaults to false (persistent connections for phonehome)

pass4SymmKey = <password>
* Authenticates traffic between:
  * License master and its license slaves.
  * Members of a cluster; see Note 1 below.
  * Deployment server (DS) and its deployment clients (DCs); see Note 2
    below.
* Note 1: Clustering may override the passphrase specified here, in
  the [clustering] stanza. A clustering searchhead connecting to multiple
  masters may further override in the [clustermaster:stanza] stanza.
* Note 2: By default, DS-DCs passphrase auth is disabled. To enable DS-DCs
  passphrase auth, you must *also* add the following line to the
  [broker:broker] stanza in restmap.conf:
    requireAuthentication = true
* In all scenarios, *every* node involved must set the same passphrase in
  the same stanza(s) (i.e. [general] and/or [clustering]); otherwise,
  respective communication (licensing and deployment in case of [general]
  stanza, clustering in case of [clustering] stanza) will not proceed.

listenOnIPv6 = no|yes|only
* By default, splunkd will listen for incoming connections (both REST and
  TCP inputs) using IPv4 only
* To enable IPv6 support in splunkd, set this to 'yes'. splunkd will
  simultaneously listen for connections on both IPv4 and IPv6
* To disable IPv4 entirely, set this to 'only', which will cause splunkd
  to exclusively accept connections over IPv6. You will probably also
  need to change mgmtHostPort in web.conf (use '[:,1]' instead of '127.0.0.1')
* Note that any setting of SPLUNK_BINDIP in your environment or
  splunk-launch.conf will override this value. In that case splunkd will
  listen on the exact address specified.

connectUsingIpVersion = auto|4-first|6-first|4-only|6-only
* When making outbound TCP connections (for forwarding eventdata, making
  distributed search requests, etc) this controls whether the connections
  will be made via IPv4 or IPv6.
* If a host is available over both IPv4 and IPv6 and this is set to
  '4-first', then we will connect over IPv4 first and fallback to IPv6 if
  the connection fails.
* If it is set to '6-first' then splunkd will try IPv6 first and fallback to
  IPv4 on failure
* If this is set to '4-only' then splunkd will only attempt to make
  connections over IPv4.
* Likewise, if this is set to '6-only', then splunkd will only attempt to
  connect to the IPv6 address.
* The default value of 'auto' will select a reasonable value based on
  listenOnIPv6 setting. If that value is set to 'no' it will act like
  '4-only'. If it is set to 'yes' it will act like '6-first' and if it is
  set to 'only' it will act like '6-only'.
* Note that connections to literal addresses are unaffected by this. For
  example, if a forwarder is configured to connect to "10.1.2.3" the
  connection will be made over IPv4 regardless of this setting.

guid = <globally unique identifier for this instance>
* This setting now (as of 5.0) belongs in the [general] stanza of
  SPLUNK_HOME/etc/instance.cfg file; please see specfile of instance.cfg for
  more information.

useHTTPTServerCompression = <bool>
* Whether splunkd HTTP server should support gzip content encoding. For more
  info on how content encoding works, see
  http://www.w3.org/Protocols/rfc2616/rfc2616-sec14.html (section 14.3).
* Defaults to true.

defaultHTTPTServerCompressionLevel = <integer>
* If useHTTPTServerCompression is enabled, this setting controls the
  compression "level" we attempt
* This number must be in the range 1 through 9
* Higher numbers produce smaller compressed results but require more CPU
  usage
* The default value of 6 is appropriate for most environments

```

```
skipHTTPCompressionAcl = <network_acl>
```

- \* Lists a set of networks or addresses to skip compressing data for. These are addresses that are considered so close that network speed is never an issue, so any CPU time spent compressing a response is wasteful.
- \* Note that the server may still respond with compressed data if it already has a compressed version of the data available.
- \* These rules are separated by commas or spaces
- \* Each rule can be in the following forms:
  1. A single IPv4 or IPv6 address (examples: "10.1.2.3", "fe80::4a3")
  2. A CIDR block of addresses (examples: "10/8", "fe80:1234/32")
  3. A DNS name, possibly with a '\*' used as a wildcard (examples: "myhost.example.com", "\*.splunk.com")
  4. A single '\*' which matches anything
- \* Entries can also be prefixed with '!' to negate their meaning.
- \* Defaults to localhost addresses.

```
site = <site-id>
```

- \* Specifies the site that this splunk instance belongs to when multisite is enabled.
- \* Valid values for site-id include site1 to site63

```
useHTTPClientCompression = true|false|on-http|on-https
```

- \* Whether gzip compression should be supported when Splunkd acts as a client (including distributed searches). Note that in order for the content to be compressed, the HTTP server that the client is connecting to should also support compression.
- \* If the connection is being made over https and useClientSSLCompression=true (see below), then setting this option to true would result in double compression work without much compression gain. It is recommended that this value be set to on-http (or to true, and useClientSSLCompression to false).
- \* Defaults to false.

```
embedSecret = <string>
```

- \* When using report embedding, normally the generated URLs can only be used on the search head they were generated on
- \* If "embedSecret" is set, then the token in the URL will be encrypted with this key. Then other search heads with the exact same setting can also use the same URL.
- \* This is needed if you want to use report embedding across multiple nodes on a search head pool.

```
parallelIngestionPipelines = <integer>
```

- \* Data being loaded into splunk, whether for indexing or forwarding, progresses through a series of steps arranged into "pipelines". By setting this to more than one, more processor threads can be set up to perform this work.
- \* Defaults to 1.
- \* NOTE: Be careful when changing this. By increasing the CPU used by data ingestion, less is available for other tasks such as searching. For most installs the default setting is optimal.
- \* NOTE: Please note that enabling multiple ingestion pipelines could change the behaviour of some of the settings in limits.conf file. Each ingestion pipeline will enforce these limits independently.
  1. maxKBps
  2. max\_fd
  3. maxHotBuckets
  4. maxHotSpanSecs

```
instanceType = <string>
```

- \* Should not be modified by users.
- \* Informs components (such as the SplunkWeb Manager section) which environment Splunk is running in, to allow for more customized behaviors.
- \* Defaults to "download", meaning no special behaviors.

```
requireBootPassphrase = <bool>
```

- \* Prompt the user for a boot passphrase when starting Splunk.
- \* Splunk uses this passphrase to grant itself access to platform-provided secret storage facilities, like the GNOME keyring.
- \* For more information about secret storage, see the [secrets] stanza in \$SPLUNK\_HOME/etc/system/README/authentication.conf.spec.
- \* Defaults to true if Common Criteria mode is enabled.



- \* Defaults to false if Common Criteria mode is disabled.
- \* NOTE: Splunk plans to submit Splunk Enterprise for Common Criteria evaluation. Splunk does not support using the product in Common Criteria mode until it has been certified by NIAP. See the "Securing Splunk Enterprise" manual for information on the status of Common Criteria certification.

## 部署配置详情

```
#####
# Deployment Configuration details
#####Deployment Configuration details

[deployment]
pass4SymmKey = <password>
    * Authenticates traffic between Deployment server (DS) and its deployment
      clients (DCs).
    * By default, DS-DCs passphrase auth is disabled. To enable DS-DCs
      passphrase auth, you must *also* add the following line to the
      [broker:broker] stanza in restmap.conf:
          requireAuthentication = true
    * If it is not set in the deployment stanza, the key will be looked in
      the general stanza
```

## SSL 配置详情

```
#####
# SSL Configuration details
#####SSL Configuration details

[sslConfig]
    * Set SSL for communications on Splunk back-end under this stanza name.
    * NOTE: To set SSL (eg HTTPS) for Splunk Web and the browser, use
      web.conf.
    * Follow this stanza name with any number of the following attribute/value
      pairs.
    * If you do not specify an entry for each attribute, Splunk will use the
      default value.

enableSplunkdSSL = <bool>
    * Enables/disables SSL on the splunkd management port (8089) and KV store
      port (8191).
    * Defaults to true.
    * Note: Running splunkd without SSL is not generally recommended.
    * Distributed search will often perform better with SSL enabled.

useClientSSLCompression = <bool>
    * Turns on HTTP client compression.
    * Server-side compression is turned on by default; setting this on the
      client side enables compression between server and client.
    * Enabling this potentially gives you much faster distributed searches
      across multiple Splunk instances.
    * Defaults to true.

useSplunkdClientSSLCompression = <bool>
    * Controls whether SSL compression would be used when splunkd is acting as
      an HTTP client, usually during certificate exchange, bundle replication,
      remote calls etc.
    * NOTE: this setting is effective if, and only if, useClientSSLCompression
      is set to true
    * NOTE: splunkd is not involved in data transfer in distributed search, the
      search in a separate process is.
    * Defaults to true.

sslVersions = <versions_list>
    * Comma-separated list of SSL versions to support for incoming connections.
    * The versions available are "ssl3", "tls1.0", "tls1.1", and "tls1.2".
    * The special version "*" selects all supported versions. The version "tls"
```

selects all versions `tls1.0` or newer.

- \* If a version is prefixed with `"-"` it is removed from the list.
- \* SSLv2 is always disabled; `"-ssl2"` is accepted in the version list but does nothing.
- \* When configured in FIPS mode, `ssl3` is always disabled regardless of this configuration.
- \* Defaults to `"*,-ssl2"` (anything newer than SSLv2).

`sslVersionsForClient = <versions_list>`

- \* Comma-separated list of SSL versions to support for outgoing HTTP connections from `splunkd`. This includes distributed search, deployment client, etc.
- \* This is usually less critical, since SSL/TLS will always pick the highest version both sides support. However, this can be used to prohibit making connections to remote servers that only support older protocols.
- \* The syntax is the same as the `sslVersions` setting above
- \* Note that for forwarder connections, there is a separate `"sslVersions"` setting in `outputs.conf`. For connections to SAML servers, there is a separate `"sslVersions"` setting in `authentication.conf`.
- \* Defaults to `"*,-ssl2"` (anything newer than SSLv2).

`supportSSLV3Only = <bool>`

- \* DEPRECATED. SSLv2 is now always disabled. The exact set of SSL versions allowed is now configurable via the `"sslVersions"` setting above.

`sslVerifyServerCert = <bool>`

- \* Used by distributed search: when making a search request to another server in the search cluster.
- \* Used by distributed deployment clients: when polling a deployment server.
- \* If this is set to true, you should make sure that the server that is being connected to is a valid one (authenticated). Both the common name and the alternate name of the server are then checked for a match if they are specified in this configuration file. A certificate is considered verified if either is matched.
- \* Default is false.

`sslCommonNameToCheck = <commonName1>, <commonName2>, ...`

- \* If this value is set, and `'sslVerifyServerCert'` is set to true, `splunkd` will limit most outbound HTTPS connections to hosts which use a cert with one of the listed common names.
- \* The most important scenario is distributed search.
- \* This feature does not work with the deployment server and client communication over SSL.
- \* Optional. Defaults to no common name checking.

`sslCommonNameList = <commonName1>, <commonName2>, ...`

- \* DEPRECATED; use `'sslCommonNameToCheck'` instead.

`sslAltNameToCheck = <alternateName1>, <alternateName2>, ...`

- \* If this value is set, and `'sslVerifyServerCert'` is set to true, `splunkd` will also be willing to verify certificates which have a so-called "Subject Alternate Name" that matches any of the alternate names in this list.
- \* Subject Alternate Names are effectively extended descriptive fields in SSL certs beyond the commonName. A common practice for HTTPS certs is to use these values to store additional valid hostnames or domains where the cert should be considered valid.
- \* Accepts a comma-separated list of Subject Alternate Names to consider valid.
- \* Items in this list are never validated against the SSL Common Name.
- \* This feature does not work with the deployment server and client communication over SSL.
- \* Optional. Defaults to no alternate name checking

`requireClientCert = <bool>`

- \* Requires that any HTTPS client that connects to `splunkd` internal HTTPS server has a certificate that was signed by a CA (Certificate Authority) specified by `'sslRootCAPath'`.
- \* Used by distributed search: Splunk indexing instances must be authenticated to connect to another splunk indexing instance.
- \* Used by distributed deployment: the deployment server requires that deployment clients are authenticated before allowing them to poll for new configurations/applications.

- \* If true, a client can connect ONLY if a certificate created by our certificate authority was used on that client.
- \* Default is false.

cipherSuite = <cipher suite string>

- \* If set, Splunk uses the specified cipher string for the HTTP server.
- \* If not set, Splunk uses the default cipher string provided by OpenSSL. This is used to ensure that the server does not accept connections using weak encryption protocols.
- \* Must specify 'dhFile' to enable any Diffie-Hellman ciphers.

ecdhCurveName = <string>

- \* DEPRECATED; use 'ecdhCurves' instead.
- \* ECDH curve to use for ECDH key negotiation
- \* We only support named curves specified by their SHORT name.
- \* The list of valid named curves by their short/long names can be obtained by executing this command:  
\$SPLUNK\_HOME/bin/splunk cmd openssl ecparam -list\_curves
- \* Default is empty string.

ecdhCurves = <comma separated list of ec curves>

- \* ECDH curves to use for ECDH key negotiation.
- \* The curves should be specified in the order of preference.
- \* The client sends these curves as a part of Client Hello.
- \* The server supports only the curves specified in the list.
- \* We only support named curves specified by their SHORT names.  
(see struct ASN1\_OBJECT in asn1.h)
- \* The list of valid named curves by their short/long names can be obtained by executing this command:  
\$SPLUNK\_HOME/bin/splunk cmd openssl ecparam -list\_curves
- \* Default is empty string.
- \* e.g. ecdhCurves = prime256v1,secp384r1,secp521r1

serverCert = <path>

- \* Full path to the PEM format server certificate file.
- \* Certificates are auto-generated by splunkd upon starting Splunk.
- \* You may replace the default cert with your own PEM format file.
- \* Default is \$SPLUNK\_HOME/etc/auth/server.pem.

sslKeysfile = <filename>

- \* DEPRECATED; use 'serverCert' instead.
- \* This file is in the directory specified by 'caPath' (see below).
- \* Default is server.pem.

sslPassword = <password>

- \* Server certificate password.
- \* Default is "password".

sslKeysfilePassword = <password>

- \* DEPRECATED; use 'sslPassword' instead.

sslRootCAPath = <path>

- \* Full path to the operating system's root CA (Certificate Authority) certificate store.
- \* The <path> must refer to a PEM format file containing one or more root CA certificates concatenated together.
- \* Required for Common Criteria.
- \* NOTE: Splunk plans to submit Splunk Enterprise for Common Criteria evaluation. Splunk does not support using the product in Common Criteria mode until it has been certified by NIAP. See the "Securing Splunk Enterprise" manual for information on the status of Common Criteria certification.
- \* This setting is not used on Windows.
- \* Default is unset.

caCertFile = <filename>

- \* DEPRECATED; use 'sslRootCAPath' instead.
- \* Used only if 'sslRootCAPath' is unset.
- \* File name (relative to 'caPath') of the CA (Certificate Authority) certificate PEM format file containing one or more certificates concatenated together.
- \* Default is cacert.pem.

```

dhFile = <path>
* PEM format Diffie-Hellman parameter file name.
* DH group size should be no less than 2048bits.
* This file is required in order to enable any Diffie-Hellman ciphers.
* Not set by default.

caPath = <path>
* DEPRECATED; use absolute paths for all certificate files.
* If certificate files given by other settings in this stanza are not absolute
  paths, then they will be relative to this path.
* Default is $SPLUNK_HOME/etc/auth.

certCreateScript = <script name>
* Creation script for generating certs on startup of Splunk.

sendStrictTransportSecurityHeader = <bool>
* If set to true, the REST interface will send a "Strict-Transport-Security"
  header with all responses to requests made over SSL.
* This can help avoid a client being tricked later by a Man-In-The-Middle
  attack to accept a non-SSL request. However, this requires a commitment that
  no non-SSL web hosts will ever be run on this hostname on any port. For
  example, if splunkweb is in default non-SSL mode this can break the
  ability of browser to connect to it. Enable with caution.
* Defaults to false

allowSslCompression = <bool>
* If set to true, the server will allow clients to negotiate
  SSL-layer data compression.
* Defaults to true.

allowSslRenegotiation = <bool>
* In the SSL protocol, a client may request renegotiation of the connection
  settings from time to time.
* Setting this to false causes the server to reject all renegotiation
  attempts, breaking the connection. This limits the amount of CPU a
  single TCP connection can use, but it can cause connectivity problems
  especially for long-lived connections.
* Defaults to true.

```

## **Splunkd HTTP 服务器配置**

```

#####
# Splunkd HTTP server configuration
#####Splunkd HTTP server configuration

[httpServer]
* Set stand-alone HTTP settings for Splunk under this stanza name.
* Follow this stanza name with any number of the following attribute/value
  pairs.
* If you do not specify an entry for each attribute, Splunk uses the default
  value.

atomFeedStylesheet = <string>
* Defines the stylesheet relative URL to apply to default Atom feeds.
* Set to 'none' to stop writing out xsl-stylesheet directive.
* Defaults to /static/atom.xsl.

max-age = <nonnegative integer>
* Set the maximum time (in seconds) to cache a static asset served off of
  the '/static' directory.
* This value is passed along in the 'Cache-Control' HTTP header.
* Defaults to 3600.

follow-symlinks = true|false
* Toggle whether static file handler (serving the '/static' directory)
  follow filesystem symlinks when serving files.
* Defaults to false.

disableDefaultPort = true|false

```

- \* If true, turns off listening on the splunkd management port (8089 by default)
- \* This setting is not recommended:
  - \* This is the general communication path to splunkd. If it is disabled, there is no way to communicate with a running splunk.
  - \* This means many command line splunk invocations cannot function, splunkweb cannot function, the REST interface cannot function, etc.
  - \* If you choose to disable the port anyway, understand that you are selecting reduced Splunk functionality.
- \* Default value is 'false'.

acceptFrom = <network\_acl> ...

- \* Lists a set of networks or addresses to accept data from. These rules are separated by commas or spaces
- \* Each rule can be in the following forms:
  1. A single IPv4 or IPv6 address (examples: "10.1.2.3", "fe80::4a3")
  2. A CIDR block of addresses (examples: "10/8", "fe80:1234/32")
  3. A DNS name, possibly with a '\*' used as a wildcard (examples: "myhost.example.com", "\*.splunk.com")
  4. A single '\*' which matches anything
- \* Entries can also be prefixed with '!' to cause the rule to reject the connection. Rules are applied in order, and the first one to match is used. For example, "!10.1/16, \*" will allow connections from everywhere except the 10.1.\*.\* network.
- \* Defaults to "\*" (accept from anywhere)

streamInWriteTimeout = <positive number>

- \* When uploading data to http server, if http server is unable to write data to receiver for configured streamInWriteTimeout seconds, it aborts write operation.
- \* Defaults to 5 seconds.

max\_content\_length = <int>

- \* Measured in bytes
- \* HTTP requests over this size will be rejected.
- \* Exists to avoid allocating an unreasonable amount of memory from web requests
- \* Defaulted to 838860800 or 800MB
- \* In environments where indexers have enormous amounts of RAM, this number can be reasonably increased to handle large quantities of bundle data.

maxSockets = <int>

- \* The number of simultaneous HTTP connections that Splunk Enterprise accepts simultaneously. You can limit this number to constrain resource usage.
- \* If set to 0, Splunk Enterprise automatically sets it to one third of the maximum allowable open files on the host.
- \* If this number is less than 50, it will be set to 50. If this number is greater than 400000, it will be set to 400000.
- \* If set to a negative number, no limit will be enforced.
- \* Defaults to 0.

maxThreads = <int>

- \* The number of threads that can be used by active HTTP transactions. You can limit this number to constrain resource usage.
- \* If set to 0, Splunk Enterprise automatically sets the limit to one third of the maximum allowable threads on the host.
- \* If this number is less than 20, it will be set to 20. If this number is greater than 150000, it will be set to 150000.
- \* If maxSockets is not negative and maxThreads is greater than maxSockets, then Splunk Enterprise sets maxThreads to be equal to maxSockets.
- \* If set to a negative number, no limit will be enforced.
- \* Defaults to 0.

forceHttp10 = auto|never|always

- \* When set to "always", the REST HTTP server will not use some HTTP 1.1 features such as persistent connections or chunked transfer encoding.
- \* When set to "auto" it will do this only if the client sent no User-Agent header, or if the user agent is known to have bugs in its HTTP/1.1 support.
- \* When set to "never" it always will allow HTTP 1.1, even to

```

clients it suspects may be buggy.
* Defaults to "auto"

crossOriginSharingPolicy = <origin_acl> ...
* List of the HTTP Origins for which to return Access-Control-Allow-* (CORS)
  headers.
* These headers tell browsers that we trust web applications at those sites
  to make requests to the REST interface
* The origin is passed as a URL without a path component (for example
  "https://app.example.com:8000")
* This setting can take a list of acceptable origins, separated
  by spaces and/or commas
* Each origin can also contain wildcards for any part. Examples:
  *://app.example.com:* (either HTTP or HTTPS on any port)
  https://*.example.com (any host under example.com, including example.com itself)
* An address can be prefixed with a '!' to negate the match, with
  the first matching origin taking precedence. For example,
  "!*://evil.example.com:* *://*.example.com:*" to not avoid
  matching one host in a domain
* A single "*" can also be used to match all origins
* By default the list is empty

x_frame_options_sameorigin = true|false
* Adds a X-Frame-Options header set to "SAMEORIGIN" to every response served by splunkd
* Defaults to true

allowEmbedTokenAuth = true|false
* If set to false, splunkd will not allow any access to artifacts
  that previously had been explicitly shared to anonymous users.
* This effectively disables all use of the "embed" feature.
* Defaults to true

cliLoginBanner = <string>
* Sets a message which will be added to the HTTP reply headers
  of requests for authentication, and to the "server/info" endpoint
* This will be printed by the Splunk CLI before it prompts
  for authentication credentials. This can be used to print
  access policy information.
* If this string starts with a '"' character, it is treated as a
  CSV-style list with each line comprising a line of the message.
  For example: "Line 1","Line 2","Line 3"
* Defaults to empty (no message)

allowBasicAuth = true|false
* Allows clients to make authenticated requests to the splunk
  server using "HTTP Basic" authentication in addition to the
  normal "authtoken" system
* This is useful for programmatic access to REST endpoints and
  for accessing the REST API from a web browser. It is not
  required for the UI or CLI.
* Defaults to true

basicAuthRealm = <string>
* When using "HTTP Basic" authentication, the 'realm' is a
  human-readable string describing the server. Typically, a web
  browser will present this string as part of its dialog box when
  asking for the username and password.
* This can be used to display a short message describing the
  server and/or its access policy.
* Defaults to "/splunk"

allowCookieAuth = true|false
* Allows clients to request an HTTP cookie from the /services/server/auth
  endpoint which can then be used to authenticate future requests
* Defaults to true

cookieAuthHttpOnly = true|false
* When using cookie based authentication, mark returned cookies
  with the "httponly" flag to tell the client not to allow javascript
  code to access its value
* Defaults to true
* NOTE: has no effect if allowCookieAuth=false

```

```
cookieAuthSecure = true|false
* When using cookie based authentication, mark returned cookies
  with the "secure" flag to tell the client never to send it over
  an unencrypted HTTP channel
* Defaults to true
* NOTE: has no effect if allowCookieAuth=false OR the splunkd REST
  interface has SSL disabled
```

```
dedicatedIoThreads = <int>
* If set to zero, HTTP I/O will be performed in the same thread
  that accepted the TCP connection.
* If set set to a non-zero value, separate threads will be run
  to handle the HTTP I/O, including SSL encryption.
* Defaults to "0"
* Typically this does not need to be changed. For most usage
  scenarios using the same the thread offers the best performance.
```

## Splunkd HTTP 服务器侦听器配置

```
#####
# Splunkd HTTPServer listener configuration
#####Splunkd HTTPServer listener
configuration

[httpServerListener:<ip>:<port>]
* Enable the splunkd REST HTTP server to listen on an additional port number
  specified by <port>. If a non-empty <ip> is included (for example:
  "[httpServerListener:127.0.0.1:8090]") the listening port will be
  bound only to a specific interface.
* Multiple "httpServerListener" stanzas can be specified to listen on
  more ports.
* Normally, splunkd listens only on the single REST port specified in
  web.conf's "mgmtHostPort" setting, and none of these stanzas need to
  be present. Add these stanzas only if you want the REST HTTP server
  to listen to more than one port.

ssl = <bool>
* Toggle whether this listening ip:port will use SSL or not.
* Default value is 'true'.
* If the main REST port is SSL (the "enableSplunkdSSL" setting in this
  file's [sslConfig] stanza) and this stanza is set to "ssl=false" then
  clients on the local machine such as the CLI may connect to this port.

listenOnIPv6 = no|yes|only
* Toggle whether this listening ip:port will listen on IPv4, IPv6, or both.
* If not present, the setting in the [general] stanza will be used

acceptFrom = <network_acl> ...
* Lists a set of networks or addresses to accept data from. These rules are
  separated by commas or spaces
* Each rule can be in the following forms:
  1. A single IPv4 or IPv6 address (examples: "10.1.2.3", "fe80::4a3")
  2. A CIDR block of addresses (examples: "10/8", "fe80:1234/32")
  3. A DNS name, possibly with a '*' used as a wildcard (examples:
    "myhost.example.com", "*.splunk.com")
  4. A single '*' which matches anything
* Entries can also be prefixed with '!' to cause the rule to reject the
  connection. Rules are applied in order, and the first one to match is
  used. For example, "!10.1/16, *" will allow connections from everywhere
  except the 10.1.*.* network.
* Defaults to the setting in the [httpServer] stanza above
```

## 静态文件处理程序 MIME 类型地图

```
#####
# Static file handler MIME-type map
#####Static file handler MIME-type map
```

```
[mimetype-extension-map]
* Map filename extensions to MIME type for files served from the static file
  handler under this stanza name.

<file-extension> = <MIME-type>
* Instructs the HTTP static file server to mark any files ending
  in 'file-extension' with a header of 'Content-Type: <MIME-type>'.
* Defaults to:
    [mimetype-extension-map]
    gif = image/gif
    htm = text/html
    jpg = image/jpg
    png = image/png
    txt = text/plain
    xml = text/xml
    xsl = text/xml
```

## ***Splunkd\_stderr.log 和 Splunkd\_stdout.log 的日志轮换***

```
#####
# Log rotation of splunkd_stderr.log & splunkd_stdout.log
#####Log rotation of splunkd_stderr.log &
splunkd_stdout.log

# These stanzas apply only on UNIX. splunkd on Windows has no
# stdout.log or stderr.log

[stderr_log_rotation]
* Controls the data retention of the file containing all messages written to
  splunkd's stderr file descriptor (fd 2).
* Typically this is extremely small, or mostly errors and warnings from
  linked libraries.

maxFileSize = <bytes>
* When splunkd_stderr.log grows larger than this value, it will be rotated.
* maxFileSize is expressed in bytes.
* You might want to increase this if you are working on a problem
  that involves large amounts of output to splunkd_stderr.log
* You might want to reduce this to allocate less storage to this log category.
* Defaults to 10000000, which is 10 si-megabytes.

BackupIndex = <non-negative integer>
* How many rolled copies to keep.
  * For example, if this is 2, splunkd_stderr.log.1 and splunkd_stderr.log.2
    may exist. Further rolls will delete the current splunkd_stderr.log.2
* You might want to increase this if you are working on a problem
  that involves large amounts of output to splunkd_stderr.log
* You might want to reduce this to allocate less storage to this log category.
* Defaults to 2.

checkFrequency = <seconds>
* How often to check the size of splunkd_stderr.log
* Larger values may result in larger rolled file sizes but take less resources.
* Smaller values may take more resources but more accurately constrain the
  file size.
* Defaults to 10, meaning 10 seconds.

[stdout_log_rotation]
* Controls the data retention of the file containing all messages written to
  splunkd's stdout file descriptor (fd 1).
* Almost always, there is nothing in this file.

* The same settings exist for this stanza with the same defaults. See above
  for definitions.

maxFileSize = <bytes>
BackupIndex = <non-negative integer>
checkFrequency = <seconds>
```



## 远程应用配置 (如: SplunkBase)

```
#####
# Remote applications configuration (e.g. SplunkBase)
#####Remote applications configuration (e.g.
SplunkBase)

[applicationsManagement]
* Set remote applications settings for Splunk under this stanza name.
* Follow this stanza name with any number of the following attribute/value
  pairs.
* If you do not specify an entry for each attribute, Splunk uses the default
  value.

allowInternetAccess = true|false
* Allow Splunk to access the remote applications repository.

url = <URL>
* Applications repository.
* Defaults to https://apps.splunk.com/api/apps

loginUrl = <URL>
* Applications repository login.
* Defaults to https://apps.splunk.com/api/account:login/

detailsUrl = <URL>
* Base URL for application information, keyed off of app ID.
* Defaults to https://apps.splunk.com/apps/id

useragent = <splunk-version>-<splunk-build-num>-<platform>
* User-agent string to use when contacting applications repository.
* <platform> includes information like operating system and CPU architecture.

updateHost = <URL>
* Host section of URL to check for app updates, e.g. https://apps.splunk.com

updatePath = <URL>
* Path section of URL to check for app updates
  For example: /api/apps:resolve/checkforupgrade

updateTimeout = <time range string>
* The minimum amount of time Splunk will wait between checks for app updates
* Examples include '24h' (24 hours), '3d' (3 days),
  '7200s' (7200 seconds, or two hours)
* Defaults to '24h'

sslVersions = <versions_list>
* Comma-separated list of SSL versions to connect to 'url' (https://apps.splunk.com).
* The versions available are "ssl3", "tls1.0", "tls1.1", and "tls1.2".
* The special version "*" selects all supported versions. The version "tls"
  selects all versions tls1.0 or newer.
* If a version is prefixed with "-" it is removed from the list.
* SSLv2 is always disabled; "-ssl2" is accepted in the version list but does nothing.
* When configured in FIPS mode, ssl3 is always disabled regardless
  of this configuration.
* Defaults to "tls1.2".

sslVerifyServerCert = <bool>
* If this is set to true, Splunk verifies that the remote server (specified in 'url')
  being connected to is a valid one (authenticated). Both the common
  name and the alternate name of the server are then checked for a
  match if they are specified in 'sslCommonNameToCheck' and 'sslAltNameToCheck'.
  A certificate is considered verified if either is matched.
* Default is true.

caCertFile = <path>
* Full path to a CA (Certificate Authority) certificate(s) PEM format file.
* The <path> must refer to a PEM format file containing one or more root CA
  certificates concatenated together.
* Used only if 'sslRootCAPath' is unset.
```

```

* Used for validating SSL certificate from https://apps.splunk.com/

sslCommonNameToCheck = <commonName1>, <commonName2>, ...
* If this value is set, and 'sslVerifyServerCert' is set to true,
  splunkd checks the common name(s) of the certificate presented by
  the remote server (specified in 'url') against this list of common names.
* Defaults to 'apps.splunk.com'

sslCommonNameList = <commonName1>, <commonName2>, ...
* DEPRECATED; use 'sslCommonNameToCheck' instead.

sslAltNameToCheck = <alternateName1>, <alternateName2>, ...
* If this value is set, and 'sslVerifyServerCert' is set to true,
  splunkd checks the alternate name(s) of the certificate presented by
  the remote server (specified in 'url') against this list of subject alternate names.
* Defaults to 'splunkbase.splunk.com, apps.splunk.com'

cipherSuite = <cipher suite string>
* If set, uses the specified cipher string for making outbound HTTPS connection.

ecdhCurves = <comma separated list of ec curves>
* ECDH curves to use for ECDH key negotiation.
* The curves should be specified in the order of preference.
* The client sends these curves as a part of Client Hello.
* We only support named curves specified by their SHORT names.
  (see struct ASN1_OBJECT in asn1.h)
* The list of valid named curves by their short/long names can be obtained
  by executing this command:
  $SPLUNK_HOME/bin/splunk cmd openssl ecparam -list_curves
* Default is empty string.
* e.g. ecdhCurves = prime256v1,secp384r1,secp521r1

```

## 多样配置

```

#####
# Misc. configuration
#####Misc. configuration

[scripts]

initialNumberOfScriptProcesses = <num>
* The number of pre-forked script processes that are launched when the
  system comes up. These scripts are reused when script REST endpoints
  *and* search scripts are executed.
  The idea is to eliminate the performance overhead of launching the script
  interpreter every time it is invoked. These processes are put in a pool.
  If the pool is completely busy when a script gets invoked, a new processes
  is fired up to handle the new invocation - but it disappears when that
  invocation is finished.

```

## (转发器, 非 Splunk 日志文件) 磁盘使用情况设置

```

#####
# Disk usage settings (for the indexer, not for Splunk log files)
#####Disk usage settings (for the indexer,
not for Splunk log files)

[diskUsage]

minFreeSpace = <num>
* Specified in megabytes.
* The default setting is 5000 (approx 5GB)
* Specifies a safe amount of space that must exist for splunkd to continue
  operating.
* Note that this affects search and indexing
* For search:
  * Before attempting to launch a search, splunk will require this amount of

```

```

    free space on the filesystem where the dispatch directory is stored,
    $SPLUNK_HOME/var/run/splunk/dispatch
* Applied similarly to the search quota values in authorize.conf and
  limits.conf.
* For indexing:
  * Periodically, the indexer will check space on all partitions
    that contain splunk indexes as specified by indexes.conf. Indexing
    will be paused and a ui banner + splunkd warning posted to indicate
    need to clear more disk space.

pollingFrequency = <num>
* After every pollingFrequency events indexed, the disk usage is checked.
* The default frequency is every 100000 events.

```

```

pollingTimerFrequency = <num>
* After every pollingTimerFrequency seconds, the disk usage is checked
* The default value is 10 seconds

```

## 队列设置

```

#####
# Queue settings
#####Queue settings
[queue]

maxSize = [<integer>|<integer>[KB|MB|GB]]
* Specifies default capacity of a queue.
* If specified as a lone integer (for example, maxSize=1000), maxSize
  indicates the maximum number of events allowed in the queue.
* If specified as an integer followed by KB, MB, or GB (for example,
  maxSize=100MB), it indicates the maximum RAM allocated for queue.
* The default is 500KB.

cntr_1_lookback_time = [<integer>[s|m]]
* The lookback counters are used to track the size and count (number of
  elements in the queue) variation of the queues using an exponentially
  moving weighted average technique. Both size and count variation
  has 3 sets of counters each. The set of 3 counters is provided to be able
  to track short, medium and long term history of size/count variation. The
  user can customize the value of these counters or lookback time.
* Specifies how far into history should the size/count variation be tracked
  for counter 1.
* It must be an integer followed by [s|m] which stands for seconds and
  minutes respectively.
* The default value for counter 1 is set to 60 seconds.

cntr_2_lookback_time = [<integer>[s|m]]
* See above for explanation and usage of the lookback counter.
* Specifies how far into history should the size/count variation be tracked
  for counter 2.
* The default value for counter 2 is set to 600 seconds.

cntr_3_lookback_time = [<integer>[s|m]]
* See above for explanation and usage of the lookback counter..
* Specifies how far into history should the size/count variation be tracked
  for counter 3.
* The default value for counter 3 is set to 900 seconds.

sampling_interval = [<integer>[s|m]]
* The lookback counters described above collects the size and count
  measurements for the queues. This specifies at what interval the
  measurement collection will happen. Note that for a particular queue all
  the counters sampling interval is same.
* It needs to be specified via an integer followed by [s|m] which stands for
  seconds and minutes respectively.
* The default sampling_interval value is 1 second.

[queue=<queueName>]

maxSize = [<integer>|<integer>[KB|MB|GB]]

```

- \* Specifies the capacity of a queue. It overrides the default capacity specified in [queue].
- \* If specified as a lone integer (for example, maxSize=1000), maxSize indicates the maximum number of events allowed in the queue.
- \* If specified as an integer followed by KB, MB, or GB (for example, maxSize=100MB), it indicates the maximum RAM allocated for queue.
- \* The default is inherited from maxSize value specified in [queue]

cntr\_1\_lookback\_time = [<integer>[s|m]]

- \* Same explanation as mentioned in [queue].
- \* Specifies the lookback time for the specific queue for counter 1.
- \* The default value is inherited from cntr\_1\_lookback\_time value specified in [queue].

cntr\_2\_lookback\_time = [<integer>[s|m]]

- \* Specifies the lookback time for the specific queue for counter 2.
- \* The default value is inherited from cntr\_2\_lookback\_time value specified in [queue].

cntr\_3\_lookback\_time = [<integer>[s|m]]

- \* Specifies the lookback time for the specific queue for counter 3.
- \* The default value is inherited from cntr\_3\_lookback\_time value specified in [queue].

sampling\_interval = [<integer>[s|m]]

- \* Specifies the sampling interval for the specific queue.
- \* The default value is inherited from sampling\_interval value specified in [queue].

## Http 端点的 PubSub 服务器设置。

```
#####
# PubSub server settings for the http endpoint.
#####PubSub server settings for the http
endpoint.
```

[pubsubsvr-http]

disabled = true|false

- \* If disabled, then http endpoint is not registered. Set this value to 'false' to expose PubSub server on http.
- \* Defaults to 'true'

stateIntervalInSecs = <seconds>

- \* The number of seconds before a connection is flushed due to inactivity. The connection is not closed, only messages for that connection are flushed.
- \* Defaults to 300 seconds (5 minutes).

## 一般文件输入设置

```
#####
# General file input settings.
#####General file input settings.
```

[fileInput]

outputQueue = <queue name>

- \* The queue that input methods should send their data to. Most users will not need to change this value.
- \* Defaults to parsingQueue.

## 控制 'splunk diag' 行为的设置，诊断性工具

```
#####
```

```

# Settings controlling the behavior of 'splunk diag', the diagnostic tool
#####Settings controlling the behavior of
'splunk diag', the diagnostic tool

[diag]

# These settings provide defaults for invocations of the splunk diag
# command. Generally these can be further modified by command line flags to
# the diag command.

EXCLUDE=<class> = <glob expression>
* Specifies a glob / shell pattern to be excluded from diags generated on
  this Splunk instance.
  * Example: */etc/secret_app/local/*.conf
* Further excludes can be added at the splunk diag command line, but there
  is no facility to disable configuration-based excludes at the command
  line.
* There is one exclude by default, for the splunk.secret file.

# the following commands can be overridden entirely by their command-line
# equivalents.

components = <comma separated list>
* Specifies which components of the diag should be gathered.
* This allows the disabling and enabling, categorically, of entire portions
  of diag functionality.
* All of these components are further subject to the exclude feature (see
  above), and component-specific filters (see below).
* Currently, with no configuration, all components except 'rest' are enabled
  by default.
* Available components are:
  * index_files    : Files from the index that indicate their health
                    (Hosts|Sources|Sourcetypes.data and bucketManifests).
                    User data is not collected.
  * index_listing  : Directory listings of the index contents are
                    gathered, in order to see filenames, directory names,
                    sizes, timestamps and the like.
  * etc            : The entire contents of the $SPLUNK_HOME/etc
                    directory. In other words, the configuration files.
  * log            : The contents of $SPLUNK_HOME/var/log/...
  * pool           : If search head pooling is enabled, the contents of the
                    pool dir.
  * dispatch       : Search artifacts, without the actual results,
                    In other words var/run/splunk/dispatch, but not the
                    results or events files
  * searchpeers    : Directory listings of knowledge bundles replicated for
                    distributed search
                    In other words: $SPLUNK_HOME/var/run/searchpeers
  * consensus      : Consensus protocol files produced by search head clustering
                    In other words: $SPLUNK_HOME/var/run/splunk/_raft
  * conf_replication_summary : Directory listing of configuration
                    replication summaries produced by search head clustering
                    In other words: $SPLUNK_HOME/var/run/splunk/snapshot
  * rest           : The contents of a variety of splunkd endpoints
                    Includes server status messages (system banners),
                    licenser banners, configured monitor inputs & tailing
                    file status (progress reading input files).
                    * On cluster masters, also gathers master info, fixups,
                      current peer list, clustered index info, current
                      generation, & buckets in bad stats
                    * On cluster slaves, also gathers local buckets & local
                      slave info, and the master information remotely from
                      the configured master.
  * kvstore        : Directory listings of the KV Store data directory
                    contents are gathered, in order to see filenames,
                    directory names, sizes, and timestamps.
  * file_validate  : Produce list of files that were in the install media
                    which have been changed. Generally this should be an
                    empty list.

* The special value 'all' is also supported, enabling everything explicitly.
* Further controlling the components from the command line:

```

- \* The switch --collect replaces this list entirely.
  - \* Example: --collect log,etc
  - This would set the componets to log and etc only, regardless of onfig
- \* The switch --enable adds a specific component to this list.
  - \* Example: --enable pool
  - This would ensure that pool data is collected, regardless of onfig
- \* The switch --disable removes a specific component from this list.
  - \* Example: --disable pool
  - This would ensure that pool data is \*NOT\* collected, regardless of onfig
- \* Currently, the default is to collect all components, save "rest".
- \* In the future there many be additional components which are not in the default set.
  - \* This may occur for new components that are expensive (large and/or slow)
  - \* This may occur for new components that are preceived as sensitive

# Data filters; these further refine what is collected

# most of the existing ones are designed to limit the size and collection time to pleasant values.

# note that most values here use underscores '\_' while the command line uses hyphens '-'

all\_dumps = <bool>

- \* This setting currently is irrelevant on Unix platforms.
- \* Affects the 'log' component of diag. (dumps are written to the log dir on Windows)
- \* Can be overridden with the --all-dumps command line flag.
- \* Normally, Splunk diag will gather only three .DMP (crash dump) files on Windows to limit diag size.
- \* If this is set to true, splunk diag will collect \*all\* .DMP files from the log directory.
- \* Defaults to unset / false (equivalent).

index\_files = [full|manifests]

- \* Selects a detail level for the 'index\_files' component.
- \* Can be overridden with the --index-files command line flag.
- \* 'manifests' limits the index file-content collection to just .bucketManifest files which give some information about Splunks idea of the general state of buckets in an index.
- \* 'full' adds the collection of Hosts.data, Sources.data, and Sourcetypes.data which indicate the breakdown of count of items by those categories per-bucket, and the timespans of those category entries
  - \* 'full' can take quite some time on very large index sizes, especially when slower remote storage is involved.
- \* Defaults to 'manifests'

index\_listing = [full|light]

- \* Selects a detail level for the 'index\_listing' component.
- \* Can be overridden with the --index-listing command line flag.
- \* 'light' gets directory listings (ls, or dir) of the hot/warm and cold container directory locations of the indexes, as well as listings of each hot bucket.
- \* 'full' gets a recursive directory listing of all the contents of every index location, which should mean all contents of all buckets.
  - \* 'full' may take significant time as well with very large bucket counts, espeically on slower storage.
- \* Defaults to 'light'

etc\_filesize\_limit = <non-negative integer in kilobytes>

- \* This filters the 'etc' component
- \* Can be overridden with the --etc-filesize-limit command line flag
- \* This value is specified in kilobytes.
  - \* Example: 2000 - this would be approximately 2MB.
- \* Files in the \$SPLUNK\_HOME/etc directory which are larger than this limit will not be collected in the diag.
- \* Diag will produce a message stating that a file has been skipped for size to the console. (In practice we found these large files are often a surprise to the administrator and indicate problems).
- \* If desired, this filter may be entirely disabled by setting the value

to 0.

- \* Currently, as a special exception, the file \$SPLUNK\_HOME/etc/system/replication/ops.json is permitted to be 10x the size of this limit.
- \* Defaults to 10000 or 10MB.

log\_age = <non-negative integer in days>

- \* This filters the 'log' component
- \* Can be overridden with the --log-age command line flag
- \* This value is specified in days
  - \* Example: 75 - this would be 75 days, or about 2.5 months.
- \* If desired, this filter may be entirely disabled by setting the value to 0.
- \* The idea of this default filter is that data older than this is rarely helpful in troubleshooting cases in any event.
- \* Defaults to 60, or approximately 2 months.

upload\_proto\_host\_port = <protocol://host:port>|disabled

- \* URI base to use for uploading files/diags to Splunk support.
- \* If set to disabled (override in a local/server.conf file), effectively disables diag upload functionality for this Splunk install.
- \* Modification may theoretically permit operations with some forms of proxies, but diag is not specifically designed for such, and support of proxy configurations that do not currently work will be considered an Enhancement Request.
- \* The communication path with api.splunk.com is over a simple but not documented protocol. If for some reason you wish to accept diag uploads into your own systems, it will probably be simpler to run diag and then upload via your own means independently. However if you have business reasons that you want this built-in, get in touch.
- \* Uploading to unencrypted http definitely not recommended.
- \* Defaults to https://api.splunk.com

## 配置许可证池的许可证管理器设置

```
#####
# License manager settings for configuring the license pool(s)
#####License manager settings for configuring
the license pool(s)

[license]
master_uri = [self|<uri>]
* An example of <uri>: <scheme>://<hostname>:<port>

active_group = Enterprise | Trial | Forwarder | Free
# these timeouts only matter if you have a master_uri set to remote master
connection_timeout = 30
* Maximum time (in seconds) to wait before connection to master times out

send_timeout = 30
* Maximum time (in seconds) to wait before sending data to master times out

receive_timeout = 30
* Maximum time (in seconds) to wait before receiving data from master times
  out

squash_threshold = <positive integer>
* Advanced setting. Periodically the indexer must report to license manager
  the data indexed broken down by source, sourcetype, host, and index. If
  the number of distinct (source,sourcetype,host,index) tuples grows over
  the squash_threshold, we squash the (host,source) values and only report a
  breakdown by {sourcetype,index}. This is to prevent explosions in
  memory + license_usage.log lines. Set this only after consulting a Splunk
  Support engineer. This needs to be set on license slaves as well as license
  master.
* Default: 2000

report_interval = <nonnegative integer>[s|m|h]
* Selects a time period for reporting in license usage to the license
  master.
* This value is intended for very large deployments (hundreds of indexers)
  where a large number of indexers may overwhelm the license server.
```

- \* The maximum permitted interval is 1 hour, and the minimum permitted interval is 1 minute.
- \* May be expressed as a positive number of seconds, minutes or hours.
- \* If no time unit is provided, seconds will be assumed.
- \* Defaults to 1 minute, or 1m.

strict\_pool\_quota = <boolean>

- \* Toggles strict pool quota enforcement
- \* If set to true, members of pools will receive warnings for a given day if usage exceeds pool size regardless of whether overall stack quota was exceeded
- \* If set to false, members of pool will only receive warnings if both pool usage exceeds pool size AND overall stack usage exceeds stack size
- \* Defaults to true

pool\_suggestion = <string>

- \* Defaults to empty, which means this feature is disabled
- \* Suggest a pool to the master for this slave.
- \* The master will use this suggestion if the master doesn't have an explicit rule mapping the slave to a given pool (ie...no slave list for the relevant license stack contains this slave explicitly)
- \* If the pool name doesn't match any existing pool, it will be ignored, no error will be generated
- \* This setting is intended to give an alternative management option for pool/slave mappings. When onboarding an indexer, it may be easier to manage the mapping on the indexer itself via this setting rather than having to update server.conf on master for every addition of new indexer
- \* NOTE: If you have multiple stacks and a slave maps to multiple pools, this feature is limited in only allowing a suggestion of a single pool; This is not a common scenario however.

[lmpool:auto\_generated\_pool\_forwarder]

- \* This is the auto generated pool for the forwarder stack

description = <textual description of this license pool>

quota = MAX|<maximum amount allowed by this license>

- \* MAX indicates the total capacity of the license. You may have only 1 pool with MAX size in a stack
- \* The quota can also be specified as a specific size eg. 20MB, 1GB etc

slaves = \*|<slave list>

- \* An asterisk(\*) indicates that any slave can connect to this pool
- \* You can also specify a comma separated slave guid list

stack\_id = forwarder

- \* The stack to which this pool belongs

[lmpool:auto\_generated\_pool\_free]

- \* This is the auto generated pool for the free stack
- \* Field descriptions are the same as that for the "lmpool:auto\_generated\_pool\_forwarder"

[lmpool:auto\_generated\_pool\_enterprise]

- \* This is the auto generated pool for the enterprise stack
- \* Field descriptions are the same as that for the "lmpool:auto\_generated\_pool\_forwarder"

[lmpool:auto\_generated\_pool\_fixed-sourcetype\_<sha256 hash of srctypes>]

- \* This is the auto generated pool for the enterprise fixed srctype stack
- \* Field descriptions are the same as that for the "lmpool:auto\_generated\_pool\_forwarder"

[lmpool:auto\_generated\_pool\_download\_trial]

- \* This is the auto generated pool for the download trial stack
- \* Field descriptions are the same as that for the "lmpool:auto\_generated\_pool\_forwarder"

#####

#

# Search head pooling configuration

#



```

# Changes to a search head's pooling configuration must be made to:
#
#     $SPLUNK_HOME/etc/system/local/server.conf
#
# In other words, you may not deploy the [pooling] stanza via an app, either
# on local disk or on shared storage.
#
# This is because these values are read before the configuration system
# itself has been completely initialized. Take the value of "storage", for
# example. This value cannot be placed within an app on shared storage
# because Splunk must use this value to find shared storage in the first
# place!
#
#####

[pooling]

state = [enabled|disabled]
* Enables or disables search head pooling.
* Defaults to disabled.

storage = <path to shared storage>
* All members of a search head pool must have access to shared storage.
* Splunk will store configurations and search artifacts here.
* On *NIX, this should be an NFS mount.
* On Windows, this should be a UNC path to a Samba/CIFS share.

app_update_triggers = true|false|silent
* Should this search head run update triggers for apps modified by other
  search heads in the pool?
* For more information about update triggers specifically, see the
  [triggers] stanza in $SPLUNK_HOME/etc/system/README/app.conf.spec.
* If set to true, this search head will attempt to reload inputs, indexes,
  custom REST endpoints, etc. stored within apps that are installed,
  updated, enabled, or disabled by other search heads.
* If set to false, this search head will not run any update triggers. Note
  that this search head will still detect configuration changes and app
  state changes made by other search heads. It simply will not reload any
  components within Splunk that might care about those changes, like input
  processors or the HTTP server.
* Setting a value of "silent" is like setting a value of "true", with one
  difference: update triggers will never result in restart banner messages
  or restart warnings in the UI. Any need to restart will instead be
  signaled only by messages in splunkd.log.
* Defaults to true.

lock.timeout = <time range string>
* Timeout for acquiring file-based locks on configuration files.
* Splunk will wait up to this amount of time before aborting a configuration
  write.
* Defaults to '10s' (10 seconds).

lock.logging = true|false
* When acquiring a file-based lock, log information into the locked file.
* This information typically includes:
  * Which host is acquiring the lock
  * What that host intends to do while holding the lock
* There is no maximum filesize or rolling policy for this logging. If you
  enable this setting, you must periodically truncate the locked file
  yourself to prevent unbounded growth.
* The information logged to the locked file is intended for debugging
  purposes only. Splunk makes no guarantees regarding the contents of the
  file. It may, for example, write padding NULs to the file or truncate the
  file at any time.
* Defaults to false.

# The following two intervals interrelate; the longest possible time for a
# state change to travel from one search pool member to the rest should be
# approximately the sum of these two timers.
poll.interval.rebuild = <time range string>
* Rebuild or refresh in-memory configuration data structures at most this
  often.

```

```

* Defaults to '1m' (1 minute).

poll.interval.check = <time range string>
* Check on-disk configuration files for changes at most this often.
* Defaults to '1m' (1 minute).

poll.blacklist.<name> = <regex>
* Do not check configuration files for changes if they match this regular
  expression.
* Example: Do not check vim swap files for changes -- .swp$

```

## 可用性高的群集配置

```

#####
# High availability clustering configuration
#####High availability clustering
configuration

[clustering]

mode = [master|slave|searchhead|disabled]
* Sets operational mode for this cluster node.
* Only one master may exist per cluster.
* Defaults to disabled.

master_uri = [<uri> | clustermaster:stanzaName1, clustermaster:stanzaName2]
* Only valid for mode=slave or searchhead
* URI of the cluster master that this slave or searchhead should connect to.
* An example of <uri>: <scheme>://<hostname>:<port>
* Only for mode=searchhead - If the searchhead is a part of multiple
  clusters, the master uris can be specified by a comma separated list.

advertised_disk_capacity = <integer>
* Acceptable value range is 10 to 100.
* Percentage to use when advertising disk capacity to the cluster master.
  This is useful for modifying weighted load balancing in indexer discovery.
* For example, if you set this attribute to 50 for an indexer with a 500GB disk,
  the indexer will advertise its disk size as 250GB, not 500GB.
* Defaults to 100.

pass4SymmKey = <password>
* Secret shared among the nodes in the cluster to prevent any
  arbitrary node from connecting to the cluster. If a slave or
  searchhead is not configured with the same secret as the master,
  it will not be able to communicate with the master.
* Not set by default.
* If it is not set in the clustering stanza, the key will be looked in the
  general stanza

service_interval = <zero or positive integer>
* Only valid for mode=master
* Specifies, in seconds, how often the master runs its service
  loop. In its service loop, the master checks the state of the
  peers and the buckets in the cluster and also schedules
  corrective action, if possible, for buckets that are not in
  compliance with replication policies.
* Defaults to 0
* A special default value of 0 indicates an auto mode where the service interval
  for the next service call is determined by the time taken by previous call.
  Service interval is bounded by the values 1 and max_auto_service_interval.
  If previous service call takes more than max_auto_service_interval seconds,
  next service interval will be set to max_auto_service_interval seconds.

cnxn_timeout = <seconds>
* Lowlevel timeout for establishing connection between cluster nodes.
* Defaults to 60s.

send_timeout = <seconds>
* Lowlevel timeout for sending data between cluster nodes.

```

- \* Defaults to 60s.

rcv\_timeout = <seconds>

- \* Lowlevel timeout for receiving data between cluster nodes.
- \* Defaults to 60s.

rep\_cxn\_timeout = <seconds>

- \* Lowlevel timeout for establishing connection for replicating data.
- \* Defaults to 5s.

rep\_send\_timeout = <seconds>

- \* Lowlevel timeout for sending replication slice data between cluster nodes.
- \* This is a soft timeout. When this timeout is triggered on source peer, it tries to determine if target is still alive. If it is still alive, it reset the timeout for another rep\_send\_timeout interval and continues. If target has failed or cumulative timeout has exceeded rep\_max\_send\_timeout, replication fails.
- \* Defaults to 5s.

rep\_rcv\_timeout = <seconds>

- \* Lowlevel timeout for receiving acknowledgement data from peers.
- \* This is a soft timeout. When this timeout is triggered on source peer, it tries to determine if target is still alive. If it is still alive, it reset the timeout for another rep\_send\_timeout interval and continues.
- \* If target has failed or cumulative timeout has exceeded rep\_max\_rcv\_timeout, replication fails.
- \* Defaults to 10s.

search\_files\_retry\_timeout = <seconds>

- \* Timeout after which request for search files from a peer is aborted.
- \* To make a bucket searchable, search specific files are copied from another source peer with search files. If search files on source peers are undergoing changes, it asks requesting peer to retry after some time. If cumulative retry period exceeds specified timeout, the requesting peer aborts the request and requests search files from another peer in the cluster that may have search files.
- \* Defaults to 600s.

re\_add\_on\_bucket\_request\_error = true|false

- \* Valid only for mode=slave
- \* If set to true, slave re-add's itself to the cluster master if cluster master returns an error on any bucket request. On re-add, slave updates the master with the latest state of all its buckets.
- \* If set to false, slave doesn't re-add itself to the cluster master. Instead, it updates the master with those buckets that master returned an error.
- \* Defaults to false.

rep\_max\_send\_timeout = <seconds>

- \* Maximum send timeout for sending replication slice data between cluster nodes.
- \* On rep\_send\_timeout source peer determines if total send timeout has exceeded rep\_max\_send\_timeout. If so, replication fails.
- \* If cumulative rep\_send\_timeout exceeds rep\_max\_send\_timeout, replication fails.
- \* Defaults to 600s.

rep\_max\_rcv\_timeout = <seconds>

- \* Maximum cumulative receive timeout for receiving acknowledgement data from peers.
- \* On rep\_rcv\_timeout source peer determines if total receive timeout has exceeded rep\_max\_rcv\_timeout. If so, replication fails.
- \* Defaults to 600s.

multisite = [true|false]

- \* Turns on the multisite feature for this master.
- \* Make sure you set site parameters on the peers when you turn this to true.
- \* Defaults to false.

replication\_factor = <positive integer>

- \* Only valid for mode=master.
- \* Determines how many copies of rawdata are created in the cluster.

- \* Use `site_replication_factor` instead of this in case multisite is turned on.
- \* Must be greater than 0.
- \* Defaults to 3

`site_replication_factor = <comma-separated string>`

- \* Only valid for `mode=master` and is only used if `multisite` is true.
- \* This specifies the per-site replication policy for any given bucket represented as a comma-separated list of per-site entries.
- \* Currently specified globally and applies to buckets in all indexes.
- \* Each entry is of the form `<site-id>:<positive integer>` which represents the number of copies to make in the specified site
- \* Valid site-ids include two mandatory keywords and optionally specific site-ids from `site1` to `site63`
- \* The mandatory keywords are:
  - `origin`: Every bucket has a origin site which is the site of the peer that originally created this bucket. The notion of 'origin' makes it possible to specify a policy that spans across multiple sites without having to enumerate it per-site.
  - `total`: The total number of copies we want for each bucket.
- \* When a site is the origin, it could potentially match both the origin and a specific site term. In that case, the max of the two is used as the count for that site.
- \* The total must be greater than or equal to sum of all the other counts (including origin).
- \* The difference between total and the sum of all the other counts is distributed across the remaining sites.
- \* Example 1: `site_replication_factor = origin:2, total:3`  
Given a cluster of 3 sites, all indexing data, every site has 2 copies of every bucket ingested in that site and one rawdata copy is put in one of the other 2 sites.
- \* Example 2: `site_replication_factor = origin:2, site3:1, total:3`  
Given a cluster of 3 sites, 2 of them indexing data, every bucket has 2 copies in the origin site and one copy in site3. So site3 has one rawdata copy of buckets ingested in both site1 and site2 and those two sites have 2 copies of their own buckets.
- \* Defaults to `origin:2, total:3`

`search_factor = <positive integer>`

- \* Only valid for `mode=master`
- \* Determines how many buckets will have index structures pre-built.
- \* Must be less than or equal to `replication_factor` and greater than 0.
- \* Defaults to 2.

`site_search_factor = <comma-separated string>`

- \* Only valid for `mode=master` and is only used if `multisite` is true.
- \* This specifies the per-site policy for searchable copies for any given bucket represented as a comma-separated list of per-site entries.
- \* This is similar to `site_replication_factor`. Please see that entry for more information on the syntax.
- \* Defaults to `origin:1, total:2`

`available_sites = <comma-separated string>`

- \* Only valid for `mode=master` and is only used if `multisite` is true.
- \* This is a comma-separated list of all the sites in the cluster.
- \* Defaults to an empty string. So if `multisite` is turned on this needs to be explicitly set

`site_mappings = <comma-separated string>`

- \* Only valid for `mode=master`
- \* When you decommission a site, you must update this attribute so that the origin bucket copies on the decommissioned site are mapped to a remaining active site. This attribute maps decommissioned sites to active sites. The bucket copies for which a decommissioned site is the origin site will then be replicated to the active site specified by the mapping.
- \* Used only if `multisite` is true and sites have been decommissioned.
- \* Each comma-separated entry is of the form `<decommissioned_site_id>:<active_site_id>` or `default_mapping:<default_site_id>`.  
`<decommissioned_site_id>` is a decommissioned site and `<active_site_id>` is an existing site, specified in `available_sites`.

For example, if available\_sites=sitel,site2,site3,site4 and you decommission site2, you can map site2 to a remaining site such as site4, like this: site2:site4 .

- \* If a site used in a mapping is later decommissioned, its previous mappings must be remapped to an available site. For instance, if you have the mapping sitel:site2 but site2 is later decommissioned, you can remap both sitel and site2 to an active site3 through the following replacement mappings - sitel:site3,site2:site3 .
- \* Optional entry with syntax default\_mapping:<default\_site\_id> represents the default mapping, for cases where an explicit mapping site is not specified. For example: default\_mapping:site3 maps any decommissioned site to site3, if they are not otherwise explicitly mapped to a site. There can only be one such entry.
- \* Defaults to an empty string.
- \* Example 1: site\_mappings = sitel:site3,default\_mapping:site4.  
The cluster must include site3 and site4 in available\_sites, and sitel must be decommissioned. The origin bucket copies for decommissioned sitel will be mapped to site3. Bucket copies for any other decommissioned sites will be mapped to site4.
- \* Example 2: site\_mappings = site2:site3  
The cluster must include site3 in available\_sites, and site2 must be decommissioned. The origin bucket copies for decommissioned site2 will be mapped to site3. This cluster has no default.
- \* Example 3: site\_mappings = default\_mapping:site5  
The above cluster must include site5 in available\_sites. The origin bucket copies for any decommissioned sites will be mapped onto site5

heartbeat\_timeout = <positive integer>

- \* Only valid for mode=master
- \* Determines when the master considers a slave down. Once a slave is down, the master will initiate fixup steps to replicate buckets from the dead slave to its peers.
- \* Defaults to 60s.

access\_logging\_for\_heartbeats = <bool>

- \* Only valid for mode=master
- \* Enables/disables logging to splunkd\_access.log for peer heartbeats
- \* defaults to false (logging disabled)
- \* NOTE: you do not have to restart master to set this config parameter. Simply run the cli command on master:  
% splunk edit cluster-config -access\_logging\_for\_heartbeats <true|false>

restart\_timeout = <positive integer>

- \* Only valid for mode=master
- \* This is the amount of time the master waits for a peer to come back when the peer is restarted (to avoid the overhead of trying to fixup the buckets that were on the peer).
- \* Note that this only works with the offline command or if the peer is restarted vi the UI.
- \* Defaults to 60s.

quiet\_period = <positive integer>

- \* Only valid for mode=master
- \* This determines the amount of time for which the master is quiet right after it starts. During this period the master does not initiate any action but is instead waiting for the slaves to register themselves. At the end of this time period, it builds its view of the cluster based on the registered information and starts normal processing.
- \* Defaults to 60s.

generation\_poll\_interval = <positive integer>

- \* Only valid if mode=master or mode=searchhead
- \* Determines how often the searchhead polls the master for generation information.
- \* Defaults to 60s.

max\_peer\_build\_load = <integer>

- \* This is the maximum number of concurrent tasks to make buckets searchable that can be assigned to a peer.
- \* Defaults to 2.

max\_peer\_rep\_load = <integer>

- \* This is the maximum number of concurrent non-streaming

replications that a peer can take part in as a target.  
 \* Defaults to 5.

max\_peer\_sum\_rep\_load = <integer>  
 \* This is the maximum number of concurrent summary replications that a peer can take part in as either a target or source.  
 \* Defaults to 5.

max\_replication\_errors = <integer>  
 \* Currently only valid for mode=slave  
 \* This is the maximum number of consecutive replication errors (currently only for hot bucket replication) from a source peer to a specific target peer. Until this limit is reached, the source continues to roll hot buckets on streaming failures to this target. After the limit is reached, the source will no longer roll hot buckets if streaming to this specific target fails. This is reset if at least one successful (hot bucket) replication occurs to this target from this source.  
 \* Defaults to 3.  
 \* The special value of 0 turns off this safeguard; so the source always rolls hot buckets on streaming error to any target.

searchable\_targets = true/false  
 \* Only valid for mode=master  
 \* Tells the master to make some replication targets searchable even while the replication is going on. This only affects hot bucket replication for now.  
 \* Defaults to true

searchable\_target\_sync\_timeout = <integer>  
 \* Only valid for mode=slave  
 \* If a hot bucket replication connection is inactive for this time (in seconds), a searchable target flushes out any pending search related in-memory files.  
 \* Note that regular syncing - when the data is flowing through regularly and the connection is not inactive - happens at a faster rate (default of 5 secs controlled by streamingTargetTsidxSyncPeriodMsec in indexes.conf).  
 \* The special value of 0 turns off this timeout behaviour.  
 \* Defaults to 60 (seconds)

target\_wait\_time = <positive integer>  
 \* Only valid for mode=master.  
 \* Specifies the time that the master waits for the target of a replication to register itself before it services the bucket again and potentially schedules another fixup.  
 \* Defaults to 150s

summary\_wait\_time = <positive integer>  
 \* Only valid for mode=master and summary\_replication=true.  
 \* Specifies the time that the master waits before scheduling fixups for a newly 'done' summary that transitioned from 'hot\_done'. This allows for other copies of the 'hot\_done' summary to also make their transition into 'done', avoiding unnecessary replications.  
 \* Defaults to 660s

commit\_retry\_time = <positive integer>  
 \* Only valid for mode=master  
 \* Specifies the interval after which, if the last generation commit failed, the master forces a retry. A retry is usually automatically kicked off after the appropriate events. This is just a backup to make sure that the master does retry no matter what.  
 \* Defaults to 300s

percent\_peers\_to\_restart = <integer between 0-100>  
 \* Suggested percentage of maximum peers to restart for rolling-restart.  
 \* Actual percentage may vary due to lack of granularity for smaller peer sets.  
 \* Regardless of setting, a minimum of 1 peer will be restarted per round

auto\_rebalance primaries = <bool>  
 \* Only valid for mode=master

- \* Specifies if the master should automatically rebalance bucket primaries on certain triggers. Currently the only defined trigger is when a peer registers with the master. When a peer registers, the master redistributes the bucket primaries so the cluster can make use of any copies in the incoming peer.
- \* Defaults to true.

idle\_connections\_pool\_size = <int>

- \* Only valid for mode=master
- \* Specifies how many idle http(s) connections we should keep alive to reuse. Reusing connections improves the time it takes to send messages to peers in the cluster.
- \* -1 (default) corresponds to "auto", letting the master determine the number of connections to keep around based on the number of peers in the cluster.

use\_batch\_mask\_changes = <bool>

- \* Only valid for mode=master
- \* Specifies if the master should process bucket mask changes in batch or individually one by one.
- \* Defaults to true.
- \* Set to false when there are 6.1 peers in the cluster for backwards compatibility.

service\_jobs\_msec = <positive integer>

- \* Only valid for mode=master
- \* Max time in milliseconds cluster master spends in servicing finished jobs per service call. Increase this if metrics.log has very high current\_size values.
- \* Defaults to 100ms.

summary\_replication = true|false

- \* Only valid for mode=master.
- \* Turns on or off summary replication.
- \* Defaults to false.

rebalance\_threshold = <number between 0.10 and 1.00>

- \* Only valid for mode=master.
- \* During rebalancing buckets amongst the cluster, this threshold is used as a percentage to determine when our cluster is balanced.
- \* 1.00 is 100% indexers fully balanced.

max\_auto\_service\_interval = <positive integer>

- \* Only valid for mode=master
- \* Only valid when service\_interval is in auto mode (i.e service\_interval = 0)
- \* Indicates the maximum value that service interval is bounded by when the service\_interval is in auto mode. If the previous service call took more than max\_auto\_service\_interval seconds, the next service call will run after max\_auto\_service\_interval seconds.
- \* Defaults to 30 seconds.
- \* It is highly recommended that you choose a value that is one-half the smaller of heartbeat\_timeout or restart\_timeout. For example, the default value of 30 is based on the default value of 60 for both heartbeat\_timeout and restart\_timeout.

register\_replication\_address = <IP address, or fully qualified machine/domain name>

- \* Only valid for mode=slave
- \* This is the address on which a slave will be available for accepting replication data. This is useful in the cases where a slave host machine has multiple interfaces and only one of them can be reached by another splunkd instance

register\_forwarder\_address = <IP address, or fully qualified machine/domain name>

- \* Only valid for mode=slave
- \* This is the address on which a slave will be available for accepting data from forwarder. This is useful in the cases where a splunk host machine has multiple interfaces and only one of them can be reached by another splunkd instance.

register\_search\_address = <IP address, or fully qualified machine/domain name>

- \* Only valid for mode=slave
- \* This is the address on which a slave will be available as search head. This is useful in the cases where a splunk host machine has multiple

interfaces and only one of them can be reached by another splunkd instance.

executor\_workers = <positive integer>  
 \* Only valid if mode=master or mode=slave  
 \* Number of threads that can be used by the clustering threadpool.  
 \* Defaults to 10. A value of 0 will default to 1.

manual\_detention = true|false  
 \* Only valid for mode=slave  
 \* Puts this peer node in manual detention.  
 \* Defaults to "false".  
 \* For the current release, this setting is for internal use only.

heartbeat\_period = <non-zero positive integer>  
 \* Only valid for mode=slave  
 \* Controls the frequency the slave attempts to send heartbeats

notify\_scan\_period = <non-zero positive integer>  
 \* Controls the frequency that the indexer scans summary folders for summary updates.  
 \* Only used when summary\_replication is enabled on the Master.  
 \* Defaults to 10 seconds.

enableS2SHeartbeat = true|false  
 \* Only valid for mode=slave  
 \* Splunk will monitor each replication connection for presence of heartbeat, and if the heartbeat is not seen for s2sHeartbeatTimeout seconds, it will close the connection.  
 \* Defaults to true.

s2sHeartbeatTimeout = <seconds>  
 \* This specifies the global timeout value for monitoring heartbeats on replication connections.  
 \* Splunk will close a replication connection if heartbeat is not seen for s2sHeartbeatTimeout seconds.  
 \* Defaults to 600 seconds (10 minutes). Replication source sends heartbeat every 30 second.

throwOnBucketBuildReadError = true|false  
 \* Valid only for mode=slave  
 \* If set to true, index clustering slave throws an exception if it encounters journal read error while building the bucket for a new searchable copy. It also throws all the search & other files generated so far in this particular bucket build.  
 \* If set to false, index clustering slave just logs the error and preserves all the search & other files generated so far & finalizes them as it cannot proceed further with this bucket.  
 \* Defaults to false

cluster\_label = <string>  
 \* This specifies the label of the indexer cluster

[clustermaster:stanzal]  
 \* Only valid for mode=searchhead when the searchhead is a part of multiple clusters.

master\_uri = <uri>  
 \* Only valid for mode=searchhead when present in this stanza.  
 \* URI of the cluster master that this searchhead should connect to.

pass4SymmKey = <password>  
 \* Secret shared among the nodes in the cluster to prevent any arbitrary node from connecting to the cluster. If a searchhead is not configured with the same secret as the master, it will not be able to communicate with the master.  
 \* Not set by default.  
 \* If it is not present here, the key in the clustering stanza will be used. If it is not present in the clustering stanza, the value in the general stanza will be used.

site = <site-id>  
 \* Specifies the site this searchhead belongs to for this particular master when multisite is enabled (see below).  
 \* Valid values for site-id include site1 to site63.



```

multisite = [true|false]
* Turns on the multisite feature for this master_uri for the searchhead.
* Make sure the master has the multisite feature turned on.
* Make sure you specify the site in case this is set to true. If no
  configuration is found in the clustermaster stanza, we default to any
  value for site that might be defined in the [general]
  stanza.
* Defaults to false.

[replication_port://<port>]
# Configure Splunk to listen on a given TCP port for replicated data from
# another cluster member.
# If mode=slave is set in the [clustering] stanza at least one
# replication_port must be configured and not disabled.

disabled = true|false
* Set to true to disable this replication port stanza.
* Defaults to false.

listenOnIPv6 = no|yes|only
* Toggle whether this listening port will listen on IPv4, IPv6, or both.
* If not present, the setting in the [general] stanza will be used.

acceptFrom = <network_acl> ...
* Lists a set of networks or addresses to accept connections from. These
  rules are separated by commas or spaces
* Each rule can be in the following forms:
  1. A single IPv4 or IPv6 address (examples: "10.1.2.3", "fe80::4a3")
  2. A CIDR block of addresses (examples: "10/8", "fe80:1234/32")
  3. A DNS name, possibly with a '*' used as a wildcard (examples:
    "myhost.example.com", "*.splunk.com")
  4. A single '*' which matches anything
* Entries can also be prefixed with '!' to cause the rule to reject the
  connection. Rules are applied in order, and the first one to match is
  used. For example, "!10.1/16, *" will allow connections from everywhere
  except the 10.1.*.* network.
* Defaults to "*" (accept replication data from anywhere)

[replication_port-ssl://<port>]
* This configuration is same as replication_port stanza above but uses SSL.

disabled = true|false
* Set to true to disable this replication port stanza.
* Defaults to false.

listenOnIPv6 = no|yes|only
* Toggle whether this listening port will listen on IPv4, IPv6, or both.
* If not present, the setting in the [general] stanza will be used.

acceptFrom = <network_acl> ...
* This setting is same as setting in replication_port stanza defined above.

serverCert = <path>
* Full path to file containing private key and server certificate.
* The <path> must refer to a PEM format file.
* There is no default value.

sslPassword = <password>
* Server certificate password, if any.
* There is no default value.

password = <password>
* DEPRECATED; use 'sslPassword' instead.

rootCA = <path>
* DEPRECATED; use '[sslConfig]/sslRootCAPath' instead.
* Full path to the root CA (Certificate Authority) certificate store.
* The <path> must refer to a PEM format file containing one or more root CA
  certificates concatenated together.
* Default is unset.

```

```

cipherSuite = <cipher suite string>
* If set, uses the specified cipher string for the SSL connection.
* If not set, uses the default cipher string.
* provided by OpenSSL. This is used to ensure that the server does not
  accept connections using weak encryption protocols.
* Must specify 'dhFile' to enable any Diffie-Hellman ciphers.

sslVersions = <versions_list>
* Comma-separated list of SSL versions to support.
* The versions available are "ssl3", "tls1.0", "tls1.1", and "tls1.2".
* The special version "*" selects all supported versions. The version "tls"
  selects all versions tls1.0 or newer.
* If a version is prefixed with "-" it is removed from the list.
* SSLv2 is always disabled; "-ssl2" is accepted in the version list but does nothing.
* When configured in FIPS mode, ssl3 is always disabled regardless
  of this configuration.
* Defaults to "*, -ssl2" (anything newer than SSLv2).

ecdhCurves = <comma separated list of ec curves>
* ECDH curves to use for ECDH key negotiation.
* The curves should be specified in the order of preference.
* The client sends these curves as a part of Client Hello.
* The server supports only the curves specified in the list.
* We only support named curves specified by their SHORT names.
  (see struct ASN1_OBJECT in asnl.h)
* The list of valid named curves by their short/long names can be obtained
  by executing this command:
  $SPLUNK_HOME/bin/splunk cmd openssl ecparam -list_curves
* Default is empty string.
* e.g. ecdhCurves = prime256v1,secp384r1,secp521r1

dhFile = <path>
* PEM format Diffie-Hellman parameter file name.
* DH group size should be no less than 2048bits.
* This file is required in order to enable any Diffie-Hellman ciphers.
* Not set by default.

dhfile = <path>
* DEPRECATED; use 'dhFile' instead.

supportSSLV3Only = <bool>
* DEPRECATED. SSLv2 is now always disabled. The exact set of SSL versions
  allowed is now configurable via the "sslVersions" setting above.

useSSLCompression = <bool>
* If true, enables SSL compression.
* Defaults to true.

compressed = <bool>
* DEPRECATED; use 'useSSLCompression' instead.
* Used only if 'useSSLCompression' is unset.

requireClientCert = <bool>
* Requires that any peer that connects to replication port has a certificate
  that can be validated by certificate authority specified in rootCA.
* Default is false.

allowSslRenegotiation = <bool>
* In the SSL protocol, a client may request renegotiation of the connection
  settings from time to time.
* Setting this to false causes the server to reject all renegotiation
  attempts, breaking the connection. This limits the amount of CPU a
  single TCP connection can use, but it can cause connectivity problems
  especially for long-lived connections.
* Defaults to true.

sslCommonNameToCheck = <commonName1>, <commonName2>, ...
* Optional. Defaults to no common name checking.
* Check the common name of the client's certificate against this list of names.
* requireClientCert must be set to true for this setting to work.

sslAltNameToCheck = <alternateName1>, <alternateName2>, ...

```

- \* Optional. Defaults to no alternate name checking.
- \* Check the alternate name of the client's certificate against this list of names.
- \* If there is no match, assume that Splunk is not authenticated against this server.
- \* requireClientCert must be set to true for this setting to work.

## 自检设置

```
#####
# Introspection settings
#####Introspection settings

[introspection:generator:disk_objects]
* For 'introspection_generator_addon', packaged with Splunk; provides the
  data ("i-data") consumed, and reported on, by 'introspection_viewer_app'
  (due to ship with a future release).
* This stanza controls the collection of i-data about: indexes; bucket
  superdirectories (homePath, coldPath, ...); volumes; search dispatch
  artifacts.
* On forwarders the collection of index, volumes and dispatch disk objects
  is disabled.

acquireExtra_i_data = true | false
* If true, extra Disk Objects i-data is emitted; you can gain more insight
  into your site, but at the cost of greater resource consumption both
  directly (the collection itself) and indirectly (increased disk and
  bandwidth utilization, to store the produced i-data).
* Please consult documentation for list of regularly emitted Disk Objects
  i-data, and extra Disk Objects i-data, appropriate to your release.
* Defaults to: false.

collectionPeriodInSecs = <positive integer>
* Controls frequency of Disk Objects i-data collection; higher frequency
  (hence, smaller period) gives a more accurate picture, but at the cost of
  greater resource consumption both directly (the collection itself) and
  indirectly (increased disk and bandwidth utilization, to store the
  produced i-data).
* Defaults to: 600 (10 minutes).

[introspection:generator:disk_objects__indexes]
* This stanza controls the collection of i-data about indexes.
* Inherits the values of 'acquireExtra_i_data' and 'collectionPeriodInSecs'
  attributes from the 'introspection:generator:disk_objects' stanza, but
  may be enabled/disabled independently of it.
* This stanza should only be used to force collection of i-data about
  indexes on dedicated forwarders.
* Enabled by default.

[introspection:generator:disk_objects__volumes]
* This stanza controls the collection of i-data about volumes.
* Inherits the values of 'acquireExtra_i_data' and 'collectionPeriodInSecs'
  attributes from the 'introspection:generator:disk_objects' stanza, but
  may be enabled/disabled independently of it.
* This stanza should only be used to force collection of i-data about
  volumes on dedicated forwarders.
* Enabled by default.

[introspection:generator:disk_objects__dispatch]
* This stanza controls the collection of i-data about search dispatch artifacts.
* Inherits the values of 'acquireExtra_i_data' and 'collectionPeriodInSecs'
  attributes from the 'introspection:generator:disk_objects' stanza, but
  may be enabled/disabled independently of it.
* This stanza should only be used to force collection of i-data about
  search dispatch artifacts on dedicated forwarders.
* Enabled by default.

[introspection:generator:disk_objects__fishbucket]
* This stanza controls the collection of i-data about:
  $SPLUNK_DB/fishbucket, where we persist per-input status of file-based
  inputs.
```

- \* Inherits the values of 'acquireExtra\_i\_data' and 'collectionPeriodInSecs' attributes from the 'introspection:generator:disk\_objects' stanza, but may be enabled/disabled independently of it.

[introspection:generator:disk\_objects\_\_bundle\_replication]

- \* This stanza controls the collection of i-data about: bundle replication metrics of distributed search
- \* Inherits the values of 'acquireExtra\_i\_data' and 'collectionPeriodInSecs' attributes from the 'introspection:generator:disk\_objects' stanza, but may be enabled/disabled independently of it.

[introspection:generator:disk\_objects\_\_partitions]

- \* This stanza controls the collection of i-data about: disk partition space utilization.
- \* Inherits the values of 'acquireExtra\_i\_data' and 'collectionPeriodInSecs' attributes from the 'introspection:generator:disk\_objects' stanza, but may be enabled/disabled independently of it.

[introspection:generator:disk\_objects\_\_summaries]

- \* Introspection data about summary disk space usage. Summary disk usage includes both data model and report summaries. The usage is collected per summaryId, locally at each indexer.

disabled = true | false

- \* If not specified, inherits the value from 'introspection:generator:disk\_objects' stanza.

collectionPeriodInSecs = <positive integer>

- \* Controls frequency of Disk Objects - summaries collection; higher frequency (hence, smaller period) gives a more accurate picture, but at the cost of greater resource consumption directly (the summaries collection itself); it is not recommended for a period less than 15 minutes.
- \* If you enable summary collection, the first collection happens 5 minutes after the Splunk instance is started. For every subsequent collection, this setting is honored.
- \* If 'collectionPeriodInSecs' smaller than 5 \* 60, it will be set back to 30 minutes internally.
- \* Set to (N\*300) seconds. Any remainder is ignored.
- \* Defaults to: 1800 (30 minutes).

[introspection:generator:resource\_usage]

- \* For 'introspection\_generator\_addon', packaged with Splunk; provides the data ("i-data") consumed, and reported on, by 'introspection\_viewer\_app' (due to ship with a future release).
- \* "Resource Usage" here refers to: CPU usage; scheduler overhead; main (physical) memory; virtual memory; pager overhead; swap; I/O; process creation (a.k.a. forking); file descriptors; TCP sockets; receive/transmit networking bandwidth.
- \* Resource Usage i-data is collected at both hostwide and per-process levels; the latter, only for processes associated with this SPLUNK\_HOME.
- \* Per-process i-data for Splunk search processes will include additional, search-specific, information.

acquireExtra\_i\_data = true | false

- \* If true, extra Resource Usage i-data is emitted; you can gain more insight into your site, but at the cost of greater resource consumption both directly (the collection itself) and indirectly (increased disk and bandwidth utilization, to store the produced i-data).
- \* Please consult documentation for list of regularly emitted Resource Usage i-data, and extra Resource Usage i-data, appropriate to your release.
- \* Defaults to: false.

collectionPeriodInSecs = <positive integer>

- \* Controls frequency of Resource Usage i-data collection; higher frequency (hence, smaller period) gives a more accurate picture, but at the cost of greater resource consumption both directly (the collection itself) and indirectly (increased disk and bandwidth utilization, to store the produced i-data).
- \* Defaults to: 600 (10 minutes) on UFs, 10 (1/6th of a minute) on non-UFs.

[introspection:generator:resource\_usage\_\_iostats]

- \* This stanza controls the collection of i-data about: IO Statistics data

```

* "IO Statistics" here refers to: read/write requests; read/write sizes;
  io service time; cpu usage during service
* IO Statistics i-data is sampled over the collectionPeriodInSecs
* Does not inherit the value of the 'collectionPeriodInSecs' attribute from the
  'introspection:generator:resource_usage' stanza, and may be enabled/disabled
  independently of it.

collectionPeriodInSecs = <positive integer>
* Controls interval of IO Statistics i-data collection; higher intervals
  gives a more accurate picture, but at the cost of greater resource consumption
  both directly (the collection itself) and indirectly (increased disk and
  bandwidth utilization, to store the produced i-data).
* Defaults to: 60 (1 minute)

[introspection:generator:kvstore]
* For 'introspection_generator_addon', packaged with Splunk
* "KV Store" here refers to: statistics information about KV Store process.

serverStatsCollectionPeriodInSecs = <positive integer>
* Controls frequency of KV Store server status collection
* Defaults to: 27 seconds.

collectionStatsCollectionPeriodInSecs = <positive integer>
* Controls frequency of KV Store db statistics collection
* Defaults to: 600 seconds.

profilingStatsCollectionPeriodInSecs = <positive integer>
* Controls frequency of KV Store profiling data collection
* Defaults to: 5 seconds

rsStatsCollectionPeriodInSecs = <positive integer>
* Controls frequency of KV Store replica set stats collection
* Defaults to: 60 seconds

```

## Splunk 启动用以控制命令的设置

```

#####
# Settings used to control commands started by Splunk
#####Settings used to control commands
started by Splunk

[commands:user_configurable]

prefix = <path>
* All non-internal commands started by splunkd will be prefixed with this
  string, allowing for "jailed" command execution.
* Should be only one word. In other words, commands are supported, but
  commands and arguments are not.
* Applies to commands such as: search scripts, scripted inputs, SSL
  certificate generation scripts. (Any commands that are
  user-configurable).
* Does not apply to trusted/non-configurable command executions, such as:
  splunk search, splunk-optimize, gunzip.
* Default is empty (no prefix).

```

## 搜索头群集配置

```

#####
# search head clustering configuration
#####search head clustering configuration

[shclustering]
disabled = true|false
* Disables or enables search head clustering on this instance.
* Defaults to true; that is, disabled.
* When enabled, the captain needs to be selected via a
  bootstrap mechanism. Once bootstrapped, further captain
  selections are made via a dynamic election mechanism.

```

\* When enabled, you will also need to specify the cluster member's own server address / management uri for identification purpose. This can be done in 2 ways: by specifying the mgmt\_uri attribute individually on each member or by specifying pairs of 'GUID, mgmt-uri' strings in the servers\_list attribute.

mgmt\_uri = [ mgmt-URI ]

\* The management uri is used to identify the cluster member's own address to itself.

\* Either mgmt\_uri or servers\_list is necessary.

\* mgmt\_uri is simpler to author but is unique for each member.

\* servers\_list is more involved, but can be copied as a config string to all members in the cluster.

servers\_list = [ <(GUID, mgmt-uri);>+ ]

\* A semicolon separated list of instance GUIDs and management URIs.

\* Each member will use its GUID to identify its own management URI.

adhoc\_searchhead = <bool>

\* This setting configures a member as an adhoc searchhead; i.e., the member will not run any scheduled jobs.

\* Use the setting captain\_is\_adhoc\_searchhead to reduce compute load on the captain.

\* Defaults to false.

no\_artifact\_replications = <bool>

\* prevent this Search Head Cluster member to be selected as a target for replications.

\* This is an advanced setting, and not to be changed without proper understanding of the implications.

\* Defaults to false

captain\_is\_adhoc\_searchhead = <bool>

\* This setting prohibits the captain from running scheduled jobs. Captain will be dedicated to controlling the activities of the cluster, but can also run adhoc search jobs from clients.

\* Defaults to false.

preferred\_captain = <bool>

\* The cluster tries to assign captaincy to a member with preferred\_captain=true.

\* Note that it is not always possible to assign captaincy to a member with preferred\_captain=true - for example, if none of the preferred members is reachable over the network. In that case, captaincy might remain on a member with preferred\_captain=false.

\* Defaults to true

replication\_factor = <positive integer>

\* Determines how many copies of search artifacts are created in the cluster.

\* This must be set to the same value on all members.

\* Defaults to 3.

pass4SymmKey = <password>

\* Secret shared among the members in the search head cluster to prevent any arbitrary instance from connecting to the cluster.

\* All members must use the same value.

\* If set in the [shclustering] stanza, it takes precedence over any setting in the [general] stanza.

\* Defaults to 'changeme' from the [general] stanza in the default server.conf.

async\_replicate\_on\_proxy = <bool>

\* If the jobs/\${sid}/results REST endpoint had to be proxied to a different member due to missing local replica, this attribute will automatically schedule an async replication to that member when set to true.

\* Default is true.

master\_dump\_service\_periods = <int>

\* If SHPMaster info is switched on in log.cfg, then captain statistics will be dumped in splunkd.log after the specified number of service periods. Purely a debugging aid.

\* Default is 500.

long\_running\_jobs\_poll\_period = <int>

\* Long running delegated jobs will be polled by the captain every

"long\_running\_jobs\_poll\_period" seconds to ascertain whether they are still running, in order to account for potential node/member failure.

- \* Default is 600, i.e. 10 minutes

scheduling\_heuristic = <string>

- \* This setting configures the job distribution heuristic on the captain.
- \* There are currently two supported strategies: 'round\_robin' or 'scheduler\_load\_based'.
- \* Default is 'scheduler\_load\_based'.

id = <GUID>

- \* Unique identifier for this cluster as a whole, shared across all cluster members.
- \* By default, Splunk will arrange for a unique value to be generated and shared across all members.

cxn\_timeout = <seconds>

- \* Low-level timeout for establishing connection between cluster members.
- \* Defaults to 60s.

send\_timeout = <seconds>

- \* Low-level timeout for sending data between search head cluster members.
- \* Defaults to 60s.

rcv\_timeout = <seconds>

- \* Low-level timeout for receiving data between search head cluster members.
- \* Defaults to 60s.

cxn\_timeout\_raft = <seconds>

- \* Low-level timeout for establishing connection between search head cluster members for the raft protocol.
- \* Defaults to 2s.

send\_timeout\_raft = <seconds>

- \* Low-level timeout for sending data between search head cluster members for the raft protocol.
- \* Defaults to 5s.

rcv\_timeout\_raft = <seconds>

- \* Low-level timeout for receiving data between search head cluster members for the raft protocol.
- \* Defaults to 5s.

rep\_cxn\_timeout = <seconds>

- \* Low-level timeout for establishing connection for replicating data.
- \* Defaults to 5s.

rep\_send\_timeout = <seconds>

- \* Low-level timeout for sending replication slice data between cluster members.
- \* This is a soft timeout. When this timeout is triggered on source peer, it tries to determine if target is still alive. If it is still alive, it reset the timeout for another rep\_send\_timeout interval and continues. If target has failed or cumulative timeout has exceeded rep\_max\_send\_timeout, replication fails.
- \* Defaults to 5s.

rep\_rcv\_timeout = <seconds>

- \* Low-level timeout for receiving acknowledgement data from members.
- \* This is a soft timeout. When this timeout is triggered on source member, it tries to determine if target is still alive. If it is still alive, it reset the timeout for another rep\_send\_timeout interval and continues. If target has failed or cumulative timeout has exceeded rep\_max\_rcv\_timeout, replication fails.
- \* Defaults to 10s.

rep\_max\_send\_timeout = <seconds>

- \* Maximum send timeout for sending replication slice data between cluster members.
- \* On rep\_send\_timeout source peer determines if total send timeout has exceeded rep\_max\_send\_timeout. If so, replication fails.

- \* If cumulative rep\_send\_timeout exceeds rep\_max\_send\_timeout, replication fails.
- \* Defaults to 600s.

rep\_max\_rcv\_timeout = <seconds>

- \* Maximum cumulative receive timeout for receiving acknowledgement data from members.
- \* On rep\_rcv\_timeout source member determines if total receive timeout has exceeded rep\_max\_rcv\_timeout. If so, replication fails.
- \* Defaults to 600s.

log\_heartbeat\_append\_entries = <bool>

- \* If true, Splunk will log the the low-level heartbeats between members in splunkd\_access.log . These heartbeats are used to maintain the authority of the captain authority over other members.
- \* Defaults to false.

election\_timeout\_ms = <positive\_integer>

- \* The amount of time that a member will wait before trying to become the captain.
- \* Half of this value is the heartbeat period.
- \* A very low value of election\_timeout\_ms can lead to unnecessary captain elections.
- \* The default is 60000ms, or 1 minute.

election\_timeout\_2\_hb\_ratio = <positive\_integer>

- \* The ratio between the election timeout and the heartbeat time.
- \* A typical ratio between 5 - 20 is desirable. Default is 12 to keep the heartbeat time at 5s.
- \* This ratio determines the number of heartbeat attempts that would fail before a member starts to timeout and tries to become the captain.

heartbeat\_timeout = <positive integer>

- \* Determines when the captain considers a member down. Once a member is down, the captain will initiate fixup steps to replicate artifacts from the dead member to its peers.
- \* Defaults to 60s.

access\_logging\_for\_heartbeats = <bool>

- \* Only valid on captain
- \* Enables/disables logging to splunkd\_access.log for member heartbeats
- \* Defaults to false (logging disabled)
- \* NOTE: you do not have to restart captain to set this config parameter. Simply run the cli command on master:

```
% splunk edit shcluster-config -access_logging_for_heartbeats <true|false>
```

restart\_timeout = <positive integer>

- \* This is the amount of time the captain waits for a member to come back when the instance is restarted (to avoid the overhead of trying to fixup the artifacts that were on the peer).

quiet\_period = <positive integer>

- \* This determines the amount of time for which a newly elected captain waits for members to join. During this period the captain does not initiate any fixups but instead waits for the members to register themselves. Job scheduling and conf replication still happen as usual during this time. At the end of this time period, the captain builds its view of the cluster based on the registered peers and starts normal processing.
- \* Defaults to 60s.

max\_peer\_rep\_load = <integer>

- \* This is the maximum number of concurrent replications that a member can take part in as a target.
- \* Defaults to 5.

target\_wait\_time = <positive integer>

- \* Specifies the time that the captain waits for the target of a replication to register itself before it services the artifact again and potentially schedules another fixup.



\* Defaults to 150s.

percent\_peers\_to\_restart = <integer between 0-100>

\* The percentage of members to restart at one time during rolling restarts.

\* Actual percentage may vary due to lack of granularity for smaller peer sets regardless of setting, a minimum of 1 peer will be restarted per round.

\* Do not set this attribute to a value greater than 20%. Otherwise, issues can arise during the captain election process.

rolling\_restart\_with\_captaincy\_exchange = <bool>

\* If this boolean is turned on, captain will try to exchange captaincy with another node during rolling restart

\* Default = true

\* if you change it to false, captain will restart and captaincy will transfer to some other node

register\_replication\_address = <IP address, or fully qualified machine/domain name>

\* This is the address on which a member will be available for accepting replication data. This is useful in the cases where a member host machine has multiple interfaces and only one of them can be reached by another splunkd instance.

executor\_workers = <positive integer>

\* Number of threads that can be used by the search head clustering threadpool.

\* Defaults to 10. A value of 0 will be interpreted as 1.

heartbeat\_period = <non-zero positive integer>

\* Controls the frequency with which the member attempts to send heartbeats.

enableS2SHeartbeat = true|false

\* Splunk will monitor each replication connection for presence of heartbeat.

If the heartbeat is not seen for s2sHeartbeatTimeout seconds, it will close the connection.

\* Defaults to true.

s2sHeartbeatTimeout = <seconds>

\* This specifies the global timeout value for monitoring heartbeats on replication connections.

\* Splunk will close a replication connection if heartbeat is not seen for s2sHeartbeatTimeout seconds.

\* Replication source sends heartbeat every 30 second.

\* Defaults to 600 seconds (10 minutes).

captain\_uri = [ static-captain-URI ]

\* The management uri of static captain is used to identify the cluster captain for a static captain.

election = <bool>

\* This is used to classify a cluster as static or dynamic (RAFT based).

\* election = false means static captain, which is used for DR situation.

\* election = true means dynamic captain election enabled through RAFT protocol

mode = <member>

\* Accepted values are captain and member, mode is used to identify the function of a node in static search head cluster. Setting mode as captain assumes it to function as both captain and a member.

#proxying related

sid\_proxying = <bool>

\* Enable or disable search artifact proxying. Changing this will impact the proxying of search results, and jobs feed will not be cluster-aware.

\* Only for internal/expert use.

\* Defaults to true.

ss\_proxying = <bool>

\* Enable or disable saved search proxying to captain. Changing this will impact the behavior of Searches and Reports Page.

\* Only for internal/expert use.

\* Defaults to true.

ra\_proxying = <bool>

\* Enable or disable saved report acceleration summaries proxying to captain. Changing this will impact the behavior of report acceleration summaries

page.

- \* Only for internal/expert use.
- \* Defaults to true.

alert\_proxying = <bool>

- \* Enable or disable alerts proxying to captain. Changing this will impact the behavior of alerts, and essentially make them not cluster-aware.
- \* Only for internal/expert use.
- \* Defaults to true.

csv\_journal\_rows\_per\_hb = <int>

- \* Controls how many rows of CSV from the delta-journal are sent per hb
- \* Used for both alerts and suppressions
- \* Do not alter this value without contacting splunk support.
- \* Defaults to 10000

conf\_replication\_period = <int>

- \* Controls how often, in seconds, a cluster member replicates configuration changes.
- \* A value of 0 disables automatic replication of configuration changes.
- \* Defaults to 5

conf\_replication\_max\_pull\_count = <int>

- \* Controls the maximum number of configuration changes a member will replicate from the captain at one time.
- \* A value of 0 disables any size limits.
- \* Defaults to 1000.

conf\_replication\_max\_push\_count = <int>

- \* Controls the maximum number of configuration changes a member will replicate to the captain at one time.
- \* A value of 0 disables any size limits.
- \* Defaults to 100.

conf\_replication\_include.<conf\_file\_name> = <bool>

- \* Controls whether Splunk replicates changes to a particular type of \*.conf file, along with any associated permissions in \*.meta files.
- \* Defaults to false.

conf\_replication\_summary.whitelist.<name> = <whitelist\_pattern>

- \* Whitelist files to be included in configuration replication summaries.

conf\_replication\_summary.blacklist.<name> = <blacklist\_pattern>

- \* Blacklist files to be excluded from configuration replication summaries.

conf\_replication\_summary.concerning\_file\_size = <int>

- \* Any individual file within a configuration replication summary that is larger than this value (in MB) will trigger a splunkd.log warning message.
- \* Defaults to 50.

conf\_replication\_summary.period = <timespan>

- \* Controls how often configuration replication summaries are created.
- \* Defaults to '1m' (1 minute).

conf\_replication\_purge.eligible\_count = <int>

- \* Controls how many configuration changes must be present before any become eligible for purging.
- \* In other words: controls the minimum number of configuration changes Splunk will remember for replication purposes.
- \* Defaults to 20000.

conf\_replication\_purge.eligible\_age = <timespan>

- \* Controls how old a configuration change must be before it is eligible for purging.
- \* Defaults to '1d' (1 day).

conf\_replication\_purge.period = <timespan>

- \* Controls how often configuration changes are purged.
- \* Defaults to '1h' (1 hour).

conf\_deploy\_repository = <path>

- \* Full path to directory containing configurations to deploy to cluster members.

```

conf_deploy_staging = <path>
* Full path to directory where preprocessed configurations may be written
  before being deployed cluster members.

conf_deploy_concerning_file_size = <int>
* Any individual file within <conf_deploy_repository> that is larger than
  this value (in MB) will trigger a splunkd.log warning message.
* Defaults to: 50

conf_deploy_fetch_url = <URL>
* Specifies the location of the deployer from which members fetch the
  configuration bundle.
* This value must be set to a <URL> in order for the configuration bundle to
  be fetched.
* Defaults to empty.

conf_deploy_fetch_mode = auto|replace|none
* Controls configuration bundle fetching behavior when the member starts up.
* When set to "replace", a member checks for a new configuration bundle on
  every startup.
* When set to "none", a member does not fetch the configuration bundle on
  startup.
* Regarding "auto":
  * If no configuration bundle has yet been fetched, "auto" is equivalent
    to "replace".
  * If the configuration bundle has already been fetched, "auto" is
    equivalent to "none".
* Defaults to "replace".

artifact_status_fields = <field> ...
  * Give a comma separated fields to pick up values from status.csv and info.csv for each search artifacts.
  * These fields will be shows in cli/rest endpoint splunk list shcluster-member-artifacts
  * Default values user, app, label

encrypt_fields = <field> ...
  * These are the fields that need to be re-encrypted when Search Head Cluster
    does its own first time run on syncing all members with a new splunk.secret key
  * Give a comma separated fields as a triple elements <conf-file>:<stanza-prefix>:<key elem>
  * For matching all stanzas from a conf, leave the stanza-prefix empty, eg: "server: :pass4SymmKey" matches all
    stanzas with pass4SymmKey as key in server.conf
  * Default values include storage/passwords, secret key for clustering/shclustering, server ssl config

enable_jobs_data_lite = <bool>
*This is for memory reduction on the captain for Search head clustering, leads to lower memory
* in captain while slaves send the artifacts status.csv as a string.
* Default : false

shcluster_label = <string>
* This specifies the label of the search head cluster

retry_autosummarize_or_data_model_acceleration_jobs = <bool>
* Controls whether the captain tries a second time to delegate an
  auto-summarized or data model acceleration job, if the first attempt to
  delegate the job fails.
* Defaults to true.

[replication_port://<port>]
# Configures the member to listen on a given TCP port for replicated data
# from another cluster member.
* At least one replication_port must be configured and not disabled.

disabled = true|false
* Set to true to disable this replication port stanza.
* Defaults to false.

listenOnIPv6 = no|yes|only
* Toggle whether this listening port will listen on IPv4, IPv6, or both.
* If not present, the setting in the [general] stanza will be used.

acceptFrom = <network_acl> ...
* Lists a set of networks or addresses to accept connections from. These

```

rules are separated by commas or spaces.

- \* Each rule can be in the following forms:
  1. A single IPv4 or IPv6 address (examples: "10.1.2.3", "fe80::4a3")
  2. A CIDR block of addresses (examples: "10/8", "fe80:1234/32")
  3. A DNS name, possibly with a '\*' used as a wildcard (examples: "myhost.example.com", "\*.splunk.com")
  4. A single '\*' which matches anything
- \* Entries can also be prefixed with '!' to cause the rule to reject the connection. Rules are applied in order, and the first one to match is used. For example, "!10.1/16, \*" will allow connections from everywhere except the 10.1.\*.\* network.
- \* Defaults to "\*" (accept replication data from anywhere)

[replication\_port-ssl://<port>]

- \* This configuration is same as replication\_port stanza above but uses SSL.

disabled = true|false

- \* Set to true to disable this replication port stanza.
- \* Defaults to false.

listenOnIPv6 = no|yes|only

- \* Toggle whether this listening port will listen on IPv4, IPv6, or both.
- \* If not present, the setting in the [general] stanza will be used.

acceptFrom = <network\_acl> ...

- \* This setting is same as setting in replication\_port stanza defined above.

serverCert = <path>

- \* Full path to file containing private key and server certificate.
- \* The <path> must refer to a PEM format file.
- \* There is no default value.

sslPassword = <password>

- \* Server certificate password, if any.
- \* There is no default value.

password = <password>

- \* DEPRECATED; use 'sslPassword' instead.
- \* Used only if 'sslPassword' is unset.

rootCA = <path>

- \* DEPRECATED; use '[sslConfig]/sslRootCAPath' instead.
- \* Used only if '[sslConfig]/sslRootCAPath' is unset.
- \* Full path to the root CA (Certificate Authority) certificate store.
- \* The <path> must refer to a PEM format file containing one or more root CA certificates concatenated together.
- \* Default is unset.

cipherSuite = <cipher suite string>

- \* If set, uses the specified cipher string for the SSL connection.
- \* If not set, uses the default cipher string.
- \* provided by OpenSSL. This is used to ensure that the server does not accept connections using weak encryption protocols.

supportSSLV3Only = <bool>

- \* DEPRECATED. SSLv2 is now always disabled. The exact set of SSL versions allowed is now configurable via the "sslVersions" setting above.

useSSLCompression = <bool>

- \* If true, enables SSL compression.
- \* Defaults to true.

compressed = <bool>

- \* DEPRECATED; use 'useSSLCompression' instead.
- \* Used only if 'useSSLCompression' is unset.

requireClientCert = <bool>

- \* Requires that any peer that connects to replication port has a certificate that can be validated by certificate authority specified in rootCA.
- \* Default is false.

allowSslRenegotiation = <bool>

- \* In the SSL protocol, a client may request renegotiation of the connection settings from time to time.
- \* Setting this to false causes the server to reject all renegotiation attempts, breaking the connection. This limits the amount of CPU a single TCP connection can use, but it can cause connectivity problems especially for long-lived connections.
- \* Defaults to true.

## KV 存储配置

```
#####
# KV Store configuration
#####KV Store configuration
[kvstore]

disabled = true|false
* Set to true to disable the KV Store process on the current server. To
  completely disable KV Store in a deployment with search head clustering or
  search head pooling, you must also disable KV Store on each individual
  server.
* Defaults to false.

port = <port>
* Port to connect to the KV Store server.
* Defaults to 8191.

replicaset = <replset>
* Replicaset name.
* Defaults to splunkrs.

distributedLookupTimeout = <seconds>
* This setting has been removed, as it is no longer needed

shutdownTimeout = <seconds>
* Time in seconds to wait for a clean shutdown of the KV Store. If this time
  is reached after signaling for a shutdown, KV Store will be terminated
  forcibly.
* Defaults to 100 seconds.

initAttempts = <int>
* The maximum number of attempts to initialize the KV Store when starting
  splunkd.
* Defaults to 300.

replication_host = <host>
* The host name to access the KV Store.
* This setting has no effect on a single Splunk instance.
* When using search head clustering, if the "replication_host" value is not
  set in the [kvstore] stanza, the host you specify for
  "mgmt_uri" in the [shclustering] stanza is used for KV
  Store connection strings and replication.
* In search head pooling, this host value is a requirement for using KV
  Store.
* This is the address on which a kvstore will be available for accepting
  remotely.

verbose = true|false
* Set to true to enable verbose logging.
* Defaults to false.

verboseLevel = <nonnegative integer>
* When verbose logging is enabled specify verbose level for logging
  from 0 to 5, where 5 is the most verbose.
* Defaults to 2.

dbPath = <path>
* Path where KV Store data is stored.
* Changing this directory after initial startup does not move existing data.
  The contents of the directory should be manually moved to the new
  location.
```

\* Defaults to \$SPLUNK\_DB/kvstore.

oplogSize = <int>

\* The size of the replication operation log, in MB, for environments with search head clustering or search head pooling.

In a standalone environment, 20% of this size is used.

\* Defaults to 1000MB (1GB).

\* Once the KV Store has created the oplog for the first time, changing this setting will NOT affect the size of the oplog. A full backup and restart of the KV Store will be required.

\* Do not change this setting without first consulting with Splunk Support.

replicationWriteTimeout = <int>

\* The time to wait, in seconds, for replication to complete while saving KV store operations. When the value is 0, the process never times out.

\* Used for replication environments (search head clustering or search head pooling).

\* Defaults to 1800 seconds (30 minutes).

caCertFile = <path>

\* DEPRECATED; use '[sslConfig]/sslRootCAPath' instead.

\* Used only if 'sslRootCAPath' is unset.

\* Full path to a CA (Certificate Authority) certificate(s) PEM format file.

\* If specified, it will be used in KV Store SSL connections and authentication.

\* Only used when Common Criteria is enabled (SPLUNK\_COMMON\_CRITERIA=1) or FIPS is enabled (i.e. SPLUNK\_FIPS=1).

\* NOTE: Splunk plans to submit Splunk Enterprise for Common Criteria evaluation. Splunk does not support using the product in Common Criteria mode until it has been certified by NIAP. See the "Securing Splunk Enterprise" manual for information on the status of Common Criteria certification.

\* Default is \$SPLUNK\_HOME/etc/auth/cacert.pem

caCertPath = <filepath>

\* DEPRECATED; use '[sslConfig]/sslRootCAPath' instead.

serverCert = <filepath>

\* A certificate file signed by the signing authority specified above by caCertPath.

\* In search head clustering or search head pooling, the certificates at different members must share the same 'subject'.

\* The Distinguished Name (DN) found in the certificate's subject, must specify a non-empty value for at least one of the following attributes: Organization (O), the Organizational Unit (OU) or the Domain Component (DC).

\* Only used when Common Criteria is enabled (SPLUNK\_COMMON\_CRITERIA=1) or FIPS is enabled (i.e. SPLUNK\_FIPS=1).

\* NOTE: Splunk plans to submit Splunk Enterprise for Common Criteria evaluation. Splunk does not support using the product in Common Criteria mode until it has been certified by NIAP. See the "Securing Splunk Enterprise" manual for information on the status of Common Criteria certification.

sslKeysPath = <filepath>

\* DEPRECATED; use 'serverCert' instead.

\* Used only when 'serverCert' is empty.

sslPassword = <password>

\* Password of the private key in the file specified by 'serverCert' above.

\* Must be specified if FIPS is enabled (i.e. SPLUNK\_FIPS=1), otherwise, KV Store will not be available. There is no default value.

\* Only used when Common Criteria is enabled (SPLUNK\_COMMON\_CRITERIA=1) or FIPS is enabled (i.e. SPLUNK\_FIPS=1).

\* NOTE: Splunk plans to submit Splunk Enterprise for Common Criteria evaluation. Splunk does not support using the product in Common Criteria mode until it has been certified by NIAP. See the "Securing Splunk Enterprise" manual for information on the status of Common Criteria certification.

sslKeysPassword = <password>

\* DEPRECATED; use 'sslPassword' instead.

\* Used only when 'sslPassword' is empty.

sslCRLPath = <filepath>

- \* Certificate Revocation List file.
- \* Optional. Defaults to no Revocation List.
- \* Only used when Common Criteria is enabled (SPLUNK\_COMMON\_CRITERIA=1) or FIPS is enabled (i.e. SPLUNK\_FIPS=1).
- \* NOTE: Splunk plans to submit Splunk Enterprise for Common Criteria evaluation. Splunk does not support using the product in Common Criteria mode until it has been certified by NIAP. See the "Securing Splunk Enterprise" manual for information on the status of Common Criteria certification.

modificationsReadIntervalMillisec = <int>

- \* Specifies how often, in milliseconds, to check for modifications to KV Store collections in order to replicate changes for distributed searches.
- \* Defaults to 1000.

modificationsMaxReadSec = <int>

- \* Maximum time interval KVStore can spend while checking for modifications before it produces collection dumps for distributed searches.
- \* Defaults to 30.

[indexer\_discovery]

pass4SymmKey = <password>

- \* Security key shared between master node and forwarders.
- \* If specified here, the same value must also be specified on all forwarders connecting to this master.

polling\_rate = <integer>

- \* A value between 1 to 10. This value affects the forwarder polling frequency to achieve the desired polling rate. The number of connected forwarders is also taken into consideration.
- \* The formula used to determine effective polling interval, in Milliseconds, is: (number\_of\_forwarders/polling\_rate + 30 seconds) \* 1000
- \* Defaults to 10.

indexerWeightByDiskCapacity = <bool>

- \* If set to true, it instructs the forwarders to use weighted load balancing. In weighted load balancing, load balancing is based on the total disk capacity of the target indexers, with the forwarder streaming more data to indexers with larger disks.
- \* The traffic sent to each indexer is based on the ratio of:  $\text{indexer\_disk\_capacity} / \text{total\_disk\_capacity\_of\_indexers\_combined}$
- \* Defaults to false.

## Raft Statemachine 配置

```
#####
# Raft Statemachine configuration
#####Raft Statemachine configuration
[raft_statemachine]

disabled = true|false
* Set to true to disable the raft statemachine.
* This feature require search head clustering to be enabled.
* Any consensus replication among search heads use this feature
* Defaults to true.

replicate_search_peers = true|false
* Add/remove search-server request is applied on all members
  of a search head cluster, when this value to set to true.
* Require a healthy search head cluster with a captain.
```

## server.conf.example

```
# Version 6.5.0
#
```

```

# This file contains an example server.conf. Use this file to configure SSL
# and HTTP server options.
#
# To use one or more of these configurations, copy the configuration block
# into server.conf in $SPLUNK_HOME/etc/system/local/. You must restart
# Splunk to enable configurations.
#
# To learn more about configuration files (including precedence) please see
# the documentation located at
# http://docs.splunk.com/Documentation/Splunk/latest/Admin/Aboutconfigurationfiles

# Allow users 8 hours before they time out
[general]
sessionTimeout=8h
pass4SymmKey = changeme

# Listen on IPv6 in addition to IPv4...
listenOnIPv6 = yes
# ...but make all outgoing TCP connections on IPv4 exclusively
connectUsingIpVersion = 4-only

# Turn on SSL:

[sslConfig]
enableSplunkdSSL = true
useClientSSLCompression = true
sslKeysfile = server.pem
sslKeysfilePassword = password
caCertFile = cacert.pem
caPath = $SPLUNK_HOME/etc/auth
certCreateScript = genMyServerCert.sh

##### SSO Example #####
# This example trusts all logins from the splunk web server and localhost
# Note that a proxy to the splunk web server should exist to enforce
# authentication
[general]
trustedIP = 127.0.0.1

#####

# Set this node to be a cluster master.
#####

[clustering]
mode = master
replication_factor = 3
pass4SymmKey = someSecret
search_factor = 2

#####

# Set this node to be a slave to cluster master "SplunkMaster01" on port
# 8089.
#####

[clustering]
mode = slave
master_uri = https://SplunkMaster01.example.com:8089
pass4SymmKey = someSecret

#####

# Set this node to be a searchhead to cluster master "SplunkMaster01" on
# port 8089.
#####

[clustering]
mode = searchhead
master_uri = https://SplunkMaster01.example.com:8089
pass4SymmKey = someSecret

```



```
#####
# Set this node to be a searchhead to multiple cluster masters -
# "SplunkMaster01" with pass4SymmKey set to 'someSecret and "SplunkMaster02"
# with no pass4SymmKey set here.
#####
[clustering]
mode = searchhead
master_uri = clustermaster:east, clustermaster:west

[clustermaster:east]
master_uri=https://SplunkMaster01.example.com:8089
pass4SymmKey=someSecret

[clustermaster:west]
master_uri=https://SplunkMaster02.example.com:8089

#####
# Open an additional non-SSL HTTP REST port, bound to the localhost
# interface (and therefore not accessible from outside the machine) Local
# REST clients like the CLI can use this to avoid SSL overhead when not
# sending data across the network.
#####
[httpServerListener:127.0.0.1:8090]
ssl = false
```

## serverclass.conf

以下为 serverclass.conf 的规范和示例文件。

### serverclass.conf.spec

```
# Version 6.5.0
#
# This file contains possible attributes and values for defining server
# classes to which deployment clients can belong. These attributes and
# values specify what content a given server class member will receive from
# the deployment server.
#
# For examples, see serverclass.conf.example. You must reload deployment
# server ("splunk reload deploy-server"), or restart splunkd, for changes to
# this file to take effect.
#
# To learn more about configuration files (including precedence) please see
# the documentation located at
# http://docs.splunk.com/Documentation/Splunk/latest/Admin/Aboutconfigurationfiles

#####
# Configure the server classes that are used by a deployment server instance.
#
# Server classes are essentially categories. They use filters to control
# what clients they apply to, contain a set of applications, and may define
# deployment server behavior for the management of those applications. The
# filters can be based on DNS name, IP address, build number of client
# machines, platform, and the so-called clientName. If a target machine
# matches the filter, then the apps and configuration content that make up
# the server class will be deployed to it.

# Property Inheritance
#
# Stanzas in serverclass.conf go from general to more specific, in the
# following order:
# [global] -> [serverClass:<name>] -> [serverClass:<scname>:app:<appname>]
#
# Some properties defined at a general level (say [global]) can be
# overridden by a more specific stanza as it applies to them. All
# overridable properties are marked as such.
```

## 第一层级：全局 #####

```
#####  
##### FIRST LEVEL: global #####  
#####FIRST LEVEL: global #####  
  
# Global stanza that defines properties for all server classes.  
[global]  
  
disabled = true|false  
* Toggles deployment server component off and on.  
* Set to true to disable.  
* Defaults to false.  
  
crossServerChecksum = true|false  
* Ensures that each app will have the same checksum across different deployment  
  servers.  
* Useful if you have multiple deployment servers behind a load-balancer.  
* Defaults to false.  
  
excludeFromUpdate = <path>[,<path>]...  
* Specifies paths to one or more top-level files or directories (and their  
  contents) to exclude from being touched during app update. Note that  
  each comma-separated entry MUST be prefixed by "$app_root$" (otherwise a  
  warning will be generated).  
* Can be overridden at the serverClass level.  
* Can be overridden at the app level.  
* Requires version 6.2.x or higher for both the Deployment Server and Client.  
  
repositoryLocation = <path>  
* The repository of applications on the server machine.  
* Can be overridden at the serverClass level.  
* Defaults to $SPLUNK_HOME/etc/deployment-apps  
  
targetRepositoryLocation = <path>  
* The location on the deployment client where to install the apps defined  
  for this Deployment Server.  
* If this value is unset, or set to empty, the repositoryLocation path is used.  
* Useful only with complex (for example, tiered) deployment strategies.  
* Defaults to $SPLUNK_HOME/etc/apps, the live  
  configuration directory for a Splunk instance.  
  
tmpFolder = <path>  
* Working folder used by deployment server.  
* Defaults to $SPLUNK_HOME/var/run/tmp  
  
continueMatching = true | false  
* Controls how configuration is layered across classes and server-specific  
  settings.  
* If true, configuration lookups continue matching server classes, beyond  
  the first match.  
* If false, only the first match will be used.  
* A serverClass can override this property and stop the matching.  
* Matching is done in the order in which server classes are defined.  
* Can be overridden at the serverClass level.  
* Defaults to true  
  
endpoint = <URL template string>  
* The endpoint from which content can be downloaded by a deployment client.  
  The deployment client knows how to substitute values for variables in the  
  URL.  
* Any custom URL can also be supplied here, as long as it uses the specified  
  variables.  
* Need not be specified unless you have a very specific need, for example:  
  To acquire deployment application files from a third-party Web server, for  
  extremely large environments.  
* Can be overridden at the serverClass level.
```

```

* Defaults to $deploymentServerUri$/services/streams/deployment?name=$serverClassName$:appName$

filterType = whitelist | blacklist
* The whitelist setting indicates a filtering strategy that pulls in a
  subset:
  * Items are not considered to match the stanza by default.
  * Items that match any whitelist entry, and do not match any blacklist
    entry are considered to match the stanza.
  * Items that match any blacklist entry are not considered to match the
    stanza, regardless of whitelist.
* The blacklist setting indicates a filtering strategy that rules out a subset:
  * Items are considered to match the stanza by default.
  * Items that match any blacklist entry, and do not match any whitelist
    entry are considered to not match the stanza.
  * Items that match any whitelist entry are considered to match the
    stanza.
* More briefly:
  * whitelist: default no-match -> whitelists enable -> blacklists disable
  * blacklist: default match -> blacklists disable-> whitelists enable
* Can be overridden at the serverClass level, and the serverClass:app level.
* Defaults to whitelist

whitelist.<n> = <clientName> | <IP address> | <hostname> | <instanceId>
blacklist.<n> = <clientName> | <IP address> | <hostname> | <instanceId>
* 'n' is an unsigned integer. The sequence may start at any value and may be
  non-consecutive.
* The value of this attribute is matched against several things in order:
  * Any clientName specified by the client in its deploymentclient.conf file
  * The IP address of the connected client
  * The hostname of the connected client, as provided by reverse DNS lookup
  * The hostname of the client, as provided by the client
  * For Splunk version > 6.4, the instanceId of the client. This is a GUID
    string, e.g. 'ffe9fe01-a4fb-425e-9f63-56cc274d7f8b'.
* All of these can be used with wildcards. * will match any sequence of
  characters. For example:
  * Match a network range: 10.1.1.*
  * Match a domain: *.splunk.com
* Can be overridden at the serverClass level, and the serverClass:app level.
* There are no whitelist or blacklist entries by default.
* These patterns are PCRE regular expressions, with the following aids for
  easier entry:
  * You can specify simply '.' to mean '\.'
  * You can specify simply '*' to mean '.*'
* Matches are always case-insensitive; you do not need to specify the '(?i)' prefix.

# Note: Overriding one type of filter (whitelist/blacklist) causes the other to
# be overridden (and hence not inherited from parent) too.

# Example with filterType=whitelist:
#   whitelist.0=*.splunk.com
#   blacklist.0=printer.splunk.com
#   blacklist.1=scanner.splunk.com
# This will cause all hosts in splunk.com, except 'printer' and 'scanner', to
# match this server class.

# Example with filterType=blacklist:
#   blacklist.0=*
#   whitelist.0=*.web.splunk.com
#   whitelist.1=*.linux.splunk.com
# This will cause only the 'web' and 'linux' hosts to match the server class.
# No other hosts will match.

# Deployment client machine types (hardware type of respective host machines)
# can also be used to match DCs.
# This filter will be used only if match of a client could not be decided using
# the whitelist/blacklist filters. The value of each machine type is
# designated by the hardware platform itself; a few common ones are:
#   linux-x86_64, windows-intel, linux-i686, freebsd-i386, darwin-i386, sunos-sun4u.
# The method for finding it varies by platform; once a deployment client is
# connected to the DS, however, you can determine the value of DC's machine
# type with this Splunk CLI command on the DS:
#   <code>./splunk list deploy-clients</code>

```

# The <code>utsname</code> values in the output are the respective DCs' machine  
# types.

```
whitelist.from_pathname = <pathname>
blacklist.from_pathname = <pathname>
```

- \* As an alternative to a series of (whitelist|blacklist).<n>, the <clientName>, <IP address>, and <hostname> list can be imported from <pathname> that is either a plain text file or a comma-separated values (CSV) file.
- \* May be used in conjunction with (whitelist|blacklist).select\_field, (whitelist|blacklist).where\_field, and (whitelist|blacklist).where\_equals.
- \* If used by itself, then <pathname> specifies a plain text file where one <clientName>, <IP address>, or <hostname> is given per line.
- \* If used in conjunction with select\_field, where\_field, and where\_equals, then <pathname> specifies a CSV file.
- \* The <pathname> is relative to \$SPLUNK\_HOME.
- \* May also be used in conjunction with (whitelist|blacklist).<n> to specify additional values, but there is no direct relation between them.
- \* At most one from\_pathname may be given per stanza.

```
whitelist.select_field = <field name> | <positive integer>
blacklist.select_field = <field name> | <positive integer>
```

- \* Specifies which field of the CSV file contains the <clientName>, <IP address>, or <hostname> either by field name or number.
- \* If <field name> is given, then the first line of the CSV file MUST be a header line containing the name(s) of all the field(s) and <field name> specifies which field contains the value(s) to be used. Note that field names are case-sensitive.
- \* If <positive integer> is given, then it specifies the column number (starting at 1) of the field that contains the value(s) to be used. In this case, the first line of the CSV file MUST NOT be a header line.
- \* MUST be used in conjunction with (whitelist|blacklist).from\_pathname.
- \* May be used in conjunction with (whitelist|blacklist).where\_field and (whitelist|blacklist).where\_equals.
- \* At most one select\_field may be given per stanza.

```
whitelist.where_field = <field name> | <positive integer>
blacklist.where_field = <field name> | <positive integer>
```

- \* Specifies that only a subset of values are to be selected from (whitelist|blacklist).select\_field.
- \* Specifies which field of the CSV file contains values to be compared against for equality with the (whitelist|blacklist).where\_equals values.
- \* Like (whitelist|blacklist).select\_field, the field may be specified by either name or number. However, select\_field and where\_field MUST be specified the same way, i.e., either BOTH by name or BOTH by number.
- \* MUST be used in conjunction with (whitelist|blacklist).select\_field and (whitelist|blacklist).where\_equals.
- \* At most one where\_field may be given per stanza.

```
whitelist.where_equals = <comma-separated list>
blacklist.where_equals = <comma-separated list>
```

- \* Specifies the value(s) that the value of (whitelist|blacklist).where\_field must equal in order to be selected via (whitelist|blacklist).select\_field.
- \* If more than one value is specified (separated by commas), then the value of (whitelist|blacklist).where\_field may equal ANY ONE of the values.
- \* Each value is a PCRE regular expression with the following aids for easier entry:
  - \* You can specify simply '.' to mean '\.'
  - \* You can specify simply '\*' to mean '.\*'
- \* Matches are always case-insensitive; you do not need to specify the '(?i)' prefix.
- \* MUST be used in conjunction with (whitelist|blacklist).select\_field and (whitelist|blacklist).where\_field.
- \* At most one where\_equals may be given per stanza.

```
machineTypesFilter = <comma-separated list>
```

- \* Not used unless specified.
- \* Boolean OR logic is employed: a match against any element in the list constitutes a match.
- \* This filter is used in boolean AND logic with white/blacklist filters. Only clients which match the white/blacklist AND which match this machineTypesFilter will be included.
- \* In other words, the match is an intersection of the matches for the

```

    white/blacklist and the matches for MachineTypesFilter.
* This filter can be overridden at the serverClass and serverClass:app
  levels.
* These patterns are PCRE regular expressions, with the following aids for
  easier entry:
    * You can specify simply '.' to mean '\.'
    * You can specify simply '*' to mean '.*'
* Matches are always case-insensitive; you do not need to specify the '(?i)'
  prefix.
* Unset by default.

restartSplunkWeb = true | false
* If true, restarts SplunkWeb on the client when a member app or a directly
  configured app is updated.
* Can be overridden at the serverClass level and the serverClass:app level.
* Defaults to false

restartSplunkd = true | false
* If true, restarts splunkd on the client when a member app or a directly
  configured app is updated.
* Can be overridden at the serverClass level and the serverClass:app level.
* Defaults to false

issueReload = true | false
* If true, triggers a reload of internal processors at the client when a
  member app or a directly configured app is updated
* If you don't want to immediately start using an app that is pushed to a
  client, you should set this to false.
* defaults to false

restartIfNeeded = true | false
* This is only valid on forwarders that are newer than 6.4.
* If true and issueReload is also true, then when an updated app is deployed
  to the client, that client will try to reload that app. If it fails, it will
  then restart.
* defaults to false

stateOnClient = enabled | disabled | noop
* If set to "enabled", sets the application state to enabled on the client,
  regardless of state on the deployment server.
* If set to "disabled", set the application state to disabled on the client,
  regardless of state on the deployment server.
* If set to "noop", the state on the client will be the same as on the
  deployment server.
* Can be overridden at the serverClass level and the serverClass:app level.
* Defaults to enabled.

precompressBundles = true | false
* Controls whether the Deployment Server will generate both .bundle and
  .bundle.gz files. The pre-compressed files offer improved performance as
  the DS is not required to compress the bundles on the fly for each client
  that it has to send the bundle to. However, this setting is only
  beneficial if there is no SSL compression in use and the client has
  support for HTTP compression.

* Deployment Server / server.conf
*   allowSslCompression = false
*   useHTTPServerCompression = true
*
* Deployment Client / server.conf
*   useHTTPClientCompression = true
*
* This option is inherited and available upto the serverclass level (not
  app). Apps belonging to server classes that required precompression will
  be compressed, even if they belong to a server class which does not
  require precompression
* Defaults to true

```

## 第二层级：服务器类 #####

```
#####
##### SECOND LEVEL: serverClass #####
#####SECOND LEVEL: serverClass #####

[serverClass:<serverClassName>]
* This stanza defines a server class. A server class is a collection of
  applications; an application may belong to multiple server classes.
* serverClassName is a unique name that is assigned to this server class.
* A server class can override all inheritable properties in the [global] stanza.
* A server class name may only contain: letters, numbers, space, underscore,
  dash, dot, tilde, and the '@' symbol. It is case-sensitive.

# NOTE:
# The keys listed below are all described in detail in the
# [global] section above. They can be used with serverClass stanza to
# override the global setting
continueMatching = true | false
endpoint = <URL template string>
excludeFromUpdate = <path>[,<path>]...
filterType = whitelist | blacklist
whitelist.<n> = <clientName> | <IP address> | <hostname>
blacklist.<n> = <clientName> | <IP address> | <hostname>
machineTypesFilter = <comma-separated list>
restartSplunkWeb = true | false
restartSplunkd = true | false
issueReload = true | false
restartIfNeeded = true | false
stateOnClient = enabled | disabled | noop
repositoryLocation = <path>
```

### 第三层级：应用 #####

```
#####
##### THIRD LEVEL: app #####
#####THIRD LEVEL: app #####

[serverClass:<server class name>:app:<app name>]
* This stanza maps an application (which must already exist in
  repositoryLocation) to the specified server class.
* server class name - the server class to which this content should be
  added.
* app name can be '*' or the name of an app:
  * The value '*' refers to all content in the repositoryLocation, adding
    it to this serverClass. '*' stanza cannot be mixed with named stanzas,
    for a given server class.
  * The name of an app explicitly adds the app to a server class.
    Typically apps are named by the folders that contain them.
  * An application name, if it is not the special '*' sign explained
    directly above, may only contain: letters, numbers, space, underscore,
    dash, dot, tilde, and the '@' symbol. It is case-sensitive.

appFile=<file name>
* In cases where the app name is different from the file or directory name,
  you can use this parameter to specify the file name. Supported formats
  are: directories, .tar files, and .tgz files.

# May override higher-level settings.
issueReload = true | false
restartIfNeeded = true | false
excludeFromUpdate = <path>[,<path>]...
```

### serverclass.conf.example

```
# Version 6.5.0
#
# Example 1
```

```

# Matches all clients and includes all apps in the server class

[global]
whitelist.0=*
# whitelist matches all clients.
[serverClass:AllApps]
[serverClass:AllApps:app:*]
# a server class that encapsulates all apps in the repositoryLocation

# Example 2
# Assign server classes based on dns names.

[global]

[serverClass:AppsForOps]
whitelist.0=*.ops.yourcompany.com
[serverClass:AppsForOps:app:unix]
[serverClass:AppsForOps:app:SplunkLightForwarder]

[serverClass:AppsForDesktops]
filterType=blacklist
# blacklist everybody except the Windows desktop machines.
blacklist.0=*
whitelist.0=*.desktops.yourcompany.com
[serverClass:AppsForDesktops:app:SplunkDesktop]

# Example 3
# Deploy server class based on machine types

[global]

[serverClass:AppsByMachineType]
# Ensure this server class is matched by all clients. It is IMPORTANT to
# have a general filter here, and a more specific filter at the app level.
# An app is matched _only_ if the server class it is contained in was
# successfully matched!
whitelist.0=*

[serverClass:AppsByMachineType:app:SplunkDesktop]
# Deploy this app only to Windows boxes.
machineTypesFilter=windows-*

[serverClass:AppsByMachineType:app:unix]
# Deploy this app only to unix boxes - 32/64 bit.
machineTypesFilter=linux-i686, linux-x86_64

# Example 4
# Specify app update exclusion list.

[global]

# The local/ subdirectory within every app will not be touched upon update.
excludeFromUpdate=$app_root$/local

[serverClass:MyApps]

[serverClass:MyApps:app:SpecialCaseApp]
# For the SpecialCaseApp, both the local/ and lookups/ subdirectories will
# not be touched upon update.
excludeFromUpdate=$app_root$/local,$app_root$/lookups

# Example 5
# Control client reloads/restarts

[global]
restartSplunkd=false
restartSplunkWeb=true

# For this serverclass, we attempt to only reload the configuration files
# within the app, if we fail to reload ie if there's a conf in the app that

```

```

# requires a restart, the admin must restart the instance themselves
[serverClass:ReloadOnly]
issueReload=true

# This is an example of a best effort reloadable serverClass. ie we try to
# reload the app, but if there are files that require a restart, only then
# do we restart
[serverClass:tryReloadThenRestart]
issueReload=true
restartIfNeeded=true

# Example 6a
# Use (whitelist|blacklist) text file import.
[serverClass:MyApps]
whitelist.from_pathname = etc/system/local/clients.txt

# Example 6b
# Use (whitelist|blacklist) CSV file import to read all values from the Client
# field (ignoring all other fields).
[serverClass:MyApps]
whitelist.select_field = Client
whitelist.from_pathname = etc/system/local/clients.csv

# Example 6c
# Use (whitelist|blacklist) CSV file import to read some values from the Client
# field (ignoring all other fields) where ServerType is one of T1, T2, or
# starts with dc.
[serverClass:MyApps]
whitelist.select_field = Client
whitelist.from_pathname = etc/system/local/server_list.csv
whitelist.where_field = ServerType
whitelist.where_equals = T1, T2, dc*

# Example 6d
# Use (whitelist|blacklist) CSV file import to read some values from field 2
# (ignoring all other fields) where field 1 is one of T1, T2, or starts with
# dc.
[serverClass:MyApps]
whitelist.select_field = 2
whitelist.from_pathname = etc/system/local/server_list.csv
whitelist.where_field = 1
whitelist.where_equals = T1, T2, dc*

```

## serverclass.seed.xml.conf

以下为 serverclass.seed.xml.conf 的规范和示例文件。

### serverclass.seed.xml.conf.spec

```

# Version 6.5.0

<!--
# This configuration is used by deploymentClient to seed a Splunk installation with applications, at startup time.
# This file should be located in the workingDir folder defined by deploymentclient.conf.
#
# An interesting fact - the DS -> DC communication on the wire also uses this XML format.
-->
<?xml version="1.0"?>
<deployment name="somename">

    <!--
    # The endpoint from which all apps can be downloaded. This value can be overridden by serviceClass or ap
    declarations below.
    # In addition, deploymentclient.conf can control how this property is used by deploymentClient - see
    deploymentclient.conf.spec.
    -->
    <endpoint>${deploymentServerUri}/services/streams/deployment?name=${serviceClassName}:${appName}</endpoint>

    <!--

```



```

# The location on the deploymentClient where all applications will be installed. This value can be overridden by
serviceClass or
# app declarations below.
# In addition, deploymentclient.conf can control how this property is used by deploymentClient - see
deploymentclient.conf.spec.
-->
<repositoryLocation>${SPLUNK_HOME}/etc/apps</repositoryLocation>

<serviceClass name="serviceClassName">
  <!--
    # The order in which this service class is processed.
  -->
  <order>N</order>

  <!--
    # DeploymentClients can also override these values using serverRepositoryLocationPolicy and
serverEndpointPolicy.
  -->
  <repositoryLocation>${SPLUNK_HOME}/etc/myapps</repositoryLocation>
  <endpoint>splunk.com/spacecake/${serviceClassName}/${appName}.tgz</endpoint>

  <!--
    # Please See serverclass.conf.spec for how these properties are used.
  -->
  <continueMatching>true</continueMatching>
  <restartSplunkWeb>false</restartSplunkWeb>
  <restartSplunkd>false</restartSplunkd>
  <stateOnClient>enabled</stateOnClient>

  <app name="appName1">
    <!--
      # Applications can override the endpoint property.
    -->
    <endpoint>splunk.com/spacecake/${appName}</endpoint>
  </app>
  <app name="appName2"/>

</serviceClass>
</deployment>

```

## serverclass.seed.xml.conf.example

```

<?xml version="1.0" encoding="UTF-8"?>
<deployment name="root">
  <serverClass name="spacecake_apps">
    <app name="app_0">
      <repositoryLocation>${SPLUNK_HOME}/etc/myapps</repositoryLocation>
      <!-- Download app_0 from the given location -->
      <endpoint>splunk.com/spacecake/apps/app_0.tgz</endpoint>
    </app>
    <app name="app_1">
      <repositoryLocation>${SPLUNK_HOME}/etc/myapps</repositoryLocation>
      <!-- Download app_1 from the given location -->
      <endpoint>splunk.com/spacecake/apps/app_1.tgz</endpoint>
    </app>
  </serverClass>
  <serverClass name="foobar_apps">
    <!-- construct url for each location based on the scheme below and download each app -->
    <endpoint>foobar.com:5556/services/streams/deployment?name=${serverClassName}_${appName}.bundle</endpoint>
    <app name="app_0"/>
    <app name="app_1"/>
    <app name="app_2"/>
  </serverClass>
  <serverClass name="local_apps">
    <endpoint>foo</endpoint>
    <app name="app_0">
      <!-- app present in local filesystem -->
      <endpoint>file:/home/johndoe/splunk/ds/service_class_2_app_0.bundle</endpoint>
    </app>
  </serverClass>
</deployment>

```

```

<app name="app_1">
  <!-- app present in local filesystem -->
  <endpoint>file:/home/johndoe/splunk/ds/service_class_2_app_1.bundle</endpoint>
</app>
<app name="app_2">
  <!-- app present in local filesystem -->
  <endpoint>file:/home/johndoe/splunk/ds/service_class_2_app_2.bundle</endpoint>
</app>
</serverClass>
</deployment>

```

## setup.xml.conf

以下为 setup.xml.conf 的规范和示例文件。

### setup.xml.conf.spec

```

#   Version 6.5.0
#
#

```

```

<!--
This file describes the setup XML config and provides some examples.

```

setup.xml provides a Setup Screen that you provide to users to specify configurations for an app. The Setup Screen is available when the user first runs the app or from the Splunk Manager: Splunk > Manager > Apps > Actions > Set up

Place setup.xml in the app's default directory:

```
$SPLUNK_HOME/etc/apps/<app>/default/setup.xml
```

The basic unit of work is an <input>, which is targeted to a triplet (endpoint, entity, field) and other information used to model the data. For example data type, validation information, name/label, etc.

The (endpoint, entity, field attributes) identifies an object where the input is read/written to, for example:

```

endpoint=saved/searches
entity=MySavedSearch
field=cron_schedule

```

The endpoint/entities addressing is relative to the app being configured. Endpoint/entity can be inherited from the outer blocks (see below how blocks work).

Inputs are grouped together within a <block> element:

- (1) blocks provide an iteration concept when the referenced REST entity is a regex
- (2) blocks allow you to group similar configuration items
- (3) blocks can contain <text> elements to provide descriptive text to the user.
- (4) blocks can be used to create a new entry rather than edit an already existing one, set the entity name to "\_new". NOTE: make sure to add the required field 'name' as an input.
- (5) blocks cannot be nested

See examples below.

Block Node attributes:

endpoint - The REST endpoint relative to "https://hostname:port/servicesNS/nobody/<app-name>/" of entities/object the block/input addresses. Generally, an endpoint maps to a Splunk configuration file.

entity - An object at the endpoint. Generally, this maps to a stanza name in a configuration file.  
NOTE: entity names should be URI encoded.

mode - (bulk | iter) used if the entity attribute is a regular expression:

- o iter - (default value for mode) Iterate over all matching entities and provide a separate input field for each.
- o bulk - Update all matching entities with the same value.

NOTE: splunk interprets '\*' as the regex '.\*'

eai\_search - a search to filter entities returned by an endpoint. If not specified the following search is used: eai:acl.app="" OR eai:acl.app="<current-app>" This search matches only objects defined in the app which the setup page is being used for.

NOTE: if objects from another app are allowed to be configured, any changes to those objects will be stored in the current app.

enabled - (true | false | in-windows | in-unix) whether this block is enabled or not

- o true - (default) this block is enabled
- o false - block disabled
- o in-windows - block is enabled only in windows installations
- o in-unix - block is enabled in non-windows installations

Input Node Attributes:

endpoint - see description above (inherited from block)

entity - see description above (inherited from block)

field - <string> the field which is being configured

old\_style\_disable - <bool> whether to perform entity disabling by submitting the edited entity with the following field set: disabled=1. (This is only relevant for inputs whose field=disabled/enabled). Defaults to false.

Nodes within an <input> element can display the name of the entity and field values within the entity on the setup screen. Specify \$name\$ to display the name of the entity. Use \$<field\_name>\$ to specify the value of a specified field.

```
-->

<setup>
<block title="Basic stuff" endpoint="saved/searches/" entity="foobar">
  <text> some description here </text>

  <input field="is_scheduled">
    <label>Enable Schedule for $name$</label>    <!-- this will be rendered as "Enable Schedule for foobar" -->
    <type>bool</type>
  </input>

  <input field="cron_scheduled">
    <label>Cron Schedule</label>
    <type>text</type>
  </input>

  <input field="actions">
    <label>Select Active Actions</label>
    <type>list</type>
  </input>

  <!-- bulk update -->
  <input entity="*" field="is_scheduled" mode="bulk">
    <label>Enable Schedule For All</label>
    <type>bool</type>
  </input>
</block>

<!-- iterative update in this block -->
<block title="Configure search" endpoint="saved/eventtypes/" entity="*" mode="iter">
  <input field="search">
    <label>$name$ search</label>
```

```

        <type>string</type>
    </input>
    <input field="disabled">
        <label>disable $name$</label>
        <type>bool</type>
    </input>
</block>

<block title="Create a new eventtype" endpoint="saved/eventtypes/" entity="_new">
    <input target="name">
        <label>Name</label>
        <type>text</type>
    </input>
    <input target="search">
        <label>Search</label>
        <type>text</type>
    </input>
</block>

<block title="Add Account Info" endpoint="storage/passwords" entity="_new">
    <input field="name">
        <label>Username</label>
        <type>text</type>
    </input>
    <input field="password">
        <label>Password</label>
        <type>password</type>
    </input>
</block>

<!-- example config for "Windows setup" -->
<block title="Collect local event logs" endpoint="admin/win-eventlogs/" eai_search="" >
    <text>
        Splunk for Windows needs at least your local event logs to demonstrate how to search them.
        You can always add more event logs after the initial setup in Splunk Manager.
    </text>

    <input entity="System" field="enabled" old_style_disable="true">
        <label>Enable $name$</label>
        <type>bool</type>
    </input>
    <input entity="Security" field="enabled" old_style_disable="true">
        <label>Enable $name$</label>
        <type>bool</type>
    </input>
    <input entity="Application" field="enabled" old_style_disable="true">
        <label>Enable $name$</label>
        <type>bool</type>
    </input>
</block>

<block title="Monitor Windows update logs" endpoint="data/inputs/monitor">
    <text>
        If you monitor the Windows update flat-file log, Splunk for Windows can show your patch history.
        You can also monitor other logs if you have them, such as IIS or DHCP logs, from Data Inputs in Splunk Manager
    </text>
    <input entity="%24WINDIR%5CWindowsUpdate.log" field="enabled">
        <label>Enable $name$</label>
        <type>bool</type>
    </input>
</block>
</setup>

```

## setup.xml.conf.example

No example

## source-classifier.conf

以下为 source-classifier.conf 的规范和示例文件。

### source-classifier.conf.spec

```
# Version 6.5.0
#
# This file contains all possible options for configuring settings for the
# file classifier in source-classifier.conf.
#
# There is a source-classifier.conf in $SPLUNK_HOME/etc/system/default/ To
# set custom configurations, place a source-classifier.conf in
# $SPLUNK_HOME/etc/system/local/. For examples, see
# source-classifier.conf.example. You must restart Splunk to enable
# configurations.
#
# To learn more about configuration files (including precedence) please see
# the documentation located at
# http://docs.splunk.com/Documentation/Splunk/latest/Admin/Aboutconfigurationfiles

ignored_model_keywords = <space-separated list of terms>
* Terms to ignore when generating a sourcetype model.
* To prevent sourcetype "bundles/learned/*-model.xml" files from containing
  sensitive terms (e.g. "bobsllaptop") that occur very frequently in your
  data files, add those terms to ignored_model_keywords.

ignored_filename_keywords = <space-separated list of terms>
* Terms to ignore when comparing a new sourcename against a known
  sourcename, for the purpose of classifying a source.
```

### source-classifier.conf.example

```
# Version 6.5.0
#
# This file contains an example source-classifier.conf. Use this file to
# configure classification
# of sources into sourcetypes.
#
# To use one or more of these configurations, copy the configuration block
# into source-classifier.conf in $SPLUNK_HOME/etc/system/local/. You must
# restart Splunk to enable configurations.
#
# To learn more about configuration files (including precedence) please see
# the documentation located at
# http://docs.splunk.com/Documentation/Splunk/latest/Admin/Aboutconfigurationfiles

# terms to ignore when generating sourcetype model to prevent model from
# containing servernames
ignored_model_keywords = sun mon tue tues wed thurs fri sat sunday monday tuesday wednesday thursday friday saturday
jan feb mar apr may jun jul aug sep oct nov dec january february march april may june july august september october
november december 2003 2004 2005 2006 2007 2008 2009 am pm ut utc gmt cet cest cetdst met mest metdst mez mesz eet
eest eetdst wet west wetdst msk msd ist jst kst hkt ast adt est edt cst cdt mst mdt pst pdt cast cadt east eadt wast
wadt

# terms to ignore when comparing a sourcename against a known sourcename
ignored_filename_keywords = log logs com common event events little main message messages queue server splunk
```

## sourcetypes.conf

以下为 sourcetypes.conf 的规范和示例文件。

### sourcetypes.conf.spec

```
# Version 6.5.0
#
# NOTE: sourcetypes.conf is a machine-generated file that stores the document
# models used by the file classifier for creating source types.
#
# Generally, you should not edit sourcetypes.conf, as most attributes are
# machine generated. However, there are two attributes which you can change.
#
# There is a sourcetypes.conf in $SPLUNK_HOME/etc/system/default/ To set custom
# configurations, place a sourcetypes.conf in $SPLUNK_HOME/etc/system/local/.
# For examples, see sourcetypes.conf.example. You must restart Splunk to enable
# configurations.
#
# To learn more about configuration files (including precedence) please see the
# documentation located at
# http://docs.splunk.com/Documentation/Splunk/latest/Admin/Aboutconfigurationfiles
```

## 全局设置

```
# GLOBAL SETTINGS
# Use the [default] stanza to define any global settings.
#
# * You can also define global settings outside of any stanza, at the top of
#   the file.
#
# * Each conf file should have at most one default stanza. If there are
#   multiple default stanzas, attributes are combined. In the case of
#   multiple definitions of the same attribute, the last definition in the
#   file wins.
#
# * If an attribute is defined at both the global level and in a specific
#   stanza, the value in the specific stanza takes precedence.
```

```
_sourcetype = <value>
* Specifies the sourcetype for the model.
* Change this to change the model's sourcetype.
* Future sources that match the model will receive a sourcetype of this new
  name.
```

```
_source = <value>
* Specifies the source (filename) for the model.
```

## sourcetypes.conf.example

```
# Version 6.5.0
#
# This file contains an example sourcetypes.conf. Use this file to configure
# sourcetype models.
#
# NOTE: sourcetypes.conf is a machine-generated file that stores the document
# models used by the file classifier for creating source types.
#
# Generally, you should not edit sourcetypes.conf, as most attributes are
# machine generated. However, there are two attributes which you can change.
#
# To use one or more of these configurations, copy the configuration block into
# sourcetypes.conf in $SPLUNK_HOME/etc/system/local/. You must restart Splunk
# to enable configurations.
#
# To learn more about configuration files (including precedence) please see the
# documentation located at
# http://docs.splunk.com/Documentation/Splunk/latest/Admin/Aboutconfigurationfiles
#
# This is an example of a machine-generated sourcetype models for a fictitious
# sourcetype cadcamlog.
#
```

```
[/Users/bob/logs/bnf.x5_Thu_Dec_13_15:59:06_2007_171714722]
_source = /Users/bob/logs/bnf.x5
_sourcetype = cadcamlog
L----- = 0.096899
L-t<_EQ> = 0.016473
```

## splunk-launch.conf

以下为 splunk-launch.conf 的规范和示例文件。

### splunk-launch.conf.spec

```
# Version 6.5.0

# splunk-launch.conf contains values used at startup time, by the splunk
# command and by windows services.
#

# Note: this conf file is different from most splunk conf files. There is
# only one in the whole system, located at
# $SPLUNK_HOME/etc/splunk-launch.conf; further, there are no stanzas,
# explicit or implicit. Finally, any splunk-launch.conf files in
# etc/apps/... or etc/users/... will be ignored.

# Lines beginning with a # are considered comments and are ignored.

#*****
# Environment variables
#
# Primarily, this file simply sets environment variables to be used by
# Splunk programs.
#
# These environment variables are the same type of system environment
# variables that can be set, on unix, using:
#   bourne shells:
#       $ export ENV_VAR=value
#   c-shells:
#       % setenv ENV_VAR value
#
# or at a windows command prompt:
#   C:\> SET ENV_VAR=value
#*****

<environment_variable>=<value>

* Any desired environment variable can be set to any value.
  Whitespace is trimmed from around both the key and value.
* Environment variables set here will be available to all splunk processes,
  barring operating system limitations.

#*****
# Specific Splunk environment settings
#
# These settings are primarily treated as environment variables, though some
# have some additional logic (defaulting).
#
# There is no need to explicitly set any of these values in typical
# environments.
#*****

SPLUNK_HOME=<pathname>

* The comment in the auto-generated splunk-launch.conf is informational, not
  a live setting, and does not need to be uncommented.
* Fully qualified path to the Splunk install directory.
* If unset, Splunk automatically determines the location of SPLUNK_HOME
  based on the location of the splunk CLI executable.
```

- \* Specifically, the parent of the directory containing splunk or splunk.exe
- \* Must be set if Common Criteria mode is enabled.
- \* NOTE: Splunk plans to submit Splunk Enterprise for Common Criteria evaluation. Splunk does not support using the product in Common Criteria mode until it has been certified by NIAP. See the "Securing Splunk Enterprise" manual for information on the status of Common Criteria certification.
- \* Defaults to unset.

SPLUNK\_DB=<pathname>

- \* The comment in the auto-generated splunk-launch.conf is informational, not a live setting, and does not need to be uncommented.
- \* Fully qualified path to the directory containing the splunk index directories.
- \* Primarily used by paths expressed in indexes.conf
- \* The comment in the autogenerated splunk-launch.conf is informational, not a live setting, and does not need to be uncommented.
- \* If unset, becomes \$SPLUNK\_HOME/var/lib/splunk (unix) or %SPLUNK\_HOME%\var\lib\splunk (windows)
- \* Defaults to unset.

SPLUNK\_BINDIP=<ip address>

- \* Specifies an interface that splunkd and splunkweb should bind to, as opposed to binding to the default for the local operating system.
- \* If unset, Splunk makes no specific request to the operating system when binding to ports/opening a listening socket. This means it effectively binds to '\*'; i.e. an unspecified bind. The exact result of this is controlled by operating system behavior and configuration.
- \* NOTE: When using this setting you must update mgmtHostPort in web.conf to match, or the command line and splunkweb will not know how to reach splunkd.
- \* For splunkd, this sets both the management port and the receiving ports (from forwarders).
- \* Useful for a host with multiple IP addresses, either to enable access or restrict access; though firewalling is typically a superior method of restriction.
- \* Overrides the Splunkweb-specific web.conf/[settings]/server.socket\_host param; the latter is preferred when SplunkWeb behavior is the focus.
- \* Defaults to unset.

SPLUNK\_IGNORE\_SELINUX=true

- \* If unset (not present), Splunk on Linux will abort startup if it detects it is running in an SELinux environment. This is because in shipping/distribution-provided SELinux environments, Splunk will not be permitted to work, and Splunk will not be able to identify clearly why.
- \* This setting is useful in environments where you have configured SELinux to enable Splunk to work.
- \* If set to any value, Splunk will launch, despite the presence of SELinux.
- \* Defaults to unset.

SPLUNK\_OS\_USER = <string> | <nonnegative integer>

- \* The OS user whose privileges Splunk will adopt when running, if this parameter is set.
- \* Example: SPLUNK\_OS\_USER=fnietzsche, but a root login is used to start splunkd. Immediately upon starting, splunkd abandons root's privileges, and acquires fnietzsche's privileges; any files created by splunkd (index data, logs, etc.) will be consequently owned by fnietzsche. So when splunkd is started next time by fnietzsche, files will be readable.
- \* When 'splunk enable boot-start -user <U>' is invoked, SPLUNK\_OS\_USER is set to <U> as a side effect.
- \* Under UNIX, username or apposite numeric UID are both acceptable; under Windows, only a username.

\*\*\*\*\*

# Service/server names.

#

# These settings are considered internal, and altering them is not supported.

#

# Under Windows, they influence the expected name of the service; on UNIX they influence the reported name of the appropriate server or daemon process.



```
#
# If you want to run multiple instances of Splunk as *services* under
# Windows, you will need to change the names below for 2nd, 3rd, ...,
# instances. That is because the 1st instance has taken up service names
# 'Splunkd' and 'Splunkweb', and you may not have multiple services with
# same name.
#*****

SPLUNK_SERVER_NAME=<name>
* Names the splunkd server/service.
* Defaults to splunkd (UNIX), or Splunkd (Windows).

SPLUNK_WEB_NAME=<name>
* Names the Python app server / web server/service.
* Defaults to splunkweb (UNIX), or Splunkweb (Windows).
```

## splunk-launch.conf.example

No example

## tags.conf

以下为 tags.conf 的规范和示例文件。

### tags.conf.spec

```
# Version 6.5.0
#
# This file contains possible attribute/value pairs for configuring tags. Set
# any number of tags for indexed or extracted fields.
#
# There is no tags.conf in $SPLUNK_HOME/etc/system/default/. To set custom
# configurations, place a tags.conf in $SPLUNK_HOME/etc/system/local/. For
# help, see tags.conf.example. You must restart Splunk to enable
# configurations.
#
# To learn more about configuration files (including precedence) please see the
# documentation located at
# http://docs.splunk.com/Documentation/Splunk/latest/Admin/Aboutconfigurationfiles
```

### [<fieldname>=<value>]

```
[<fieldname>=<value>]
* The field name and value to which the tags in the stanza
  apply ( eg host=localhost ).
* A tags.conf file can contain multiple stanzas. It is recommended that the
  value be URL encoded to avoid
* config file parsing errors especially if the field value contains the
  following characters: \n, =, []
* Each stanza can refer to only one field=value

<tag1> = <enabled|disabled>
<tag2> = <enabled|disabled>
<tag3> = <enabled|disabled>
* Set whether each <tag> for this specific <fieldname><value> is enabled or
  disabled.
* While you can have multiple tags in a stanza (meaning that multiple tags are
  assigned to the same field/value combination), only one tag is allowed per
  stanza line. In other words, you can't have a list of tags on one line of the
  stanza.

* WARNING: Do not quote the <tag> value: foo=enabled, not "foo"-enabled.
```

### tags.conf.example

```
# Version 6.5.0
#
# This is an example tags.conf. Use this file to define tags for fields.
#
# To use one or more of these configurations, copy the configuration block into
# tags.conf in $SPLUNK_HOME/etc/system/local/. You must restart Splunk to
# enable configurations.
#
# To learn more about configuration files (including precedence) please see the
# documentation located at
# http://docs.splunk.com/Documentation/Splunk/latest/Admin/Aboutconfigurationfiles
#
# This first example presents a situation where the field is "host" and the
# three hostnames for which tags are being defined are "hostswitch,"
# "emailbox," and "devmachine." Each hostname has two tags applied to it, one
# per line. Note also that the "building1" tag has been applied to two hostname
# values (emailbox and devmachine).

[host=hostswitch]
pci = enabled
cardholder-dest = enabled

[host=emailbox]
email = enabled
building1 = enabled

[host=devmachine]
development = enabled
building1 = enabled

[src_ip=192.168.1.1]
firewall = enabled

[seekPtr=1cb58000]
EOF = enabled
NOT_EOF = disabled
```

## telemetry.conf

以下为 telemetry.conf 的规范和示例文件。

### Telemetry.conf. 规范

```
# Version 6.5.0
#
# This file contains possible attributes and values for configuring global
# telemetry settings. Please note that enabling these settings would enable
# apps to collect telemetry data about app usage and other properties.
#
# There is no global, default telemetry.conf. Instead, a telemetry.conf may
# exist in each app in Splunk Enterprise.
#
# To learn more about configuration files (including precedence) please see
# the documentation located at
# http://docs.splunk.com/Documentation/Splunk/latest/Admin/Aboutconfigurationfiles
```

### 全局设置

```
# GLOBAL SETTINGS
# Use the [default] stanza to define any global settings.
# * You can also define global settings outside of any stanza, at the top
#   of the file.
# * Each conf file should have at most one default stanza. If there are
#   multiple default stanzas, attributes are combined. In the case of
#   multiple definitions of the same attribute, the last definition in the
#   file wins.
# * If an attribute is defined at both the global level and in a specific
```

```
# stanza, the value in the specific stanza takes precedence.
```

## **[general]**

```
[general]
sendLicenseUsage = true|false
* Send the licensing usage information of splunk/app to the app owner
* Defaults to false

sendAnonymizedUsage = true|false
* Send the anonymized usage information about various categories like
  infrastructure, utilization etc of splunk/app to the app owner
* Defaults to false

precheckSendAnonymizedUsage = true|false
* Default value for sending anonymized usage in opt in modal
* Defaults to false

precheckSendLicenseUsage = true|false
* Default value for sending license usage in opt in modal
* Defaults to true

showOptInModal = true|false
* Shows the opt in modal. DO NOT SET! When a user opts in, it will
  automatically be set to false to not show the modal again.
* Defaults to true

deprecatedConfig = true|false
* Setting to determine whether the splunk deployment is following
  best practices for the platform as well as the app
* Defaults to false

precheckSendLicenseUsage = true|false
* Default value for sending license usage in opt in modal
* Defaults to true

precheckSendAnonymizedUsage = true|false
* Default value for sending anonymized usage in opt in modal
* Defaults to false

retryTransaction = <string>
* Setting that is created if the telemetry conf updates cannot be delivered to
  the cluster master for the splunk_instrumentation app.
* Defaults to an empty string
```

## **telemetry.conf.示例**

```
# Version 6.5.0
#
# This file contains possible attributes and values for configuring global
# telemetry settings. Please note that enabling these settings would enable
# apps to collect telemetry data about app usage and other properties.
#
# There is no global, default telemetry.conf. Instead, a telemetry.conf may
# exist in each app in Splunk Enterprise.
#
# To learn more about configuration files (including precedence) please see
# the documentation located at
# http://docs.splunk.com/Documentation/Splunk/latest/Admin/Aboutconfigurationfiles

[general]
sendLicenseUsage = false
sendAnonymizedUsage = false
precheckSendAnonymizedUsage = false
precheckSendLicenseUsage = true
showOptInModal = true
deprecatedConfig = false
```

# times.conf

以下为 times.conf 的规范和示例文件。

## times.conf.spec

```
# Version 6.5.0
#
# This file contains possible attribute/value pairs for creating custom time
# ranges.
#
# To set custom configurations, place a times.conf in
# $SPLUNK_HOME/etc/system/local/. For help, see times.conf.example. You
# must restart Splunk to enable configurations.
#
# To learn more about configuration files (including precedence) please see
# the documentation located at
# http://docs.splunk.com/Documentation/Splunk/latest/Admin/Aboutconfigurationfiles
```

### 全局设置

```
# GLOBAL SETTINGS
# Use the [default] stanza to define any global settings.
# * You can also define global settings outside of any stanza, at the top
#   of the file.
# * Each conf file should have at most one default stanza. If there are
#   multiple default stanzas, attributes are combined. In the case of
#   multiple definitions of the same attribute, the last definition in the
#   file wins.
# * If an attribute is defined at both the global level and in a specific
#   stanza, the value in the specific stanza takes precedence.
```

### [<timerange\_name>]

```
[<timerange_name>]
* The token to be used when accessing time ranges via the API or command
  line
* A times.conf file can contain multiple stanzas.

label = <string>
* The textual description used by the UI to reference this time range
* Required

header_label = <string>
* The textual description used by the UI when displaying search results in
  this time range.
* Optional. If omitted, the <timerange_name> is used instead.

earliest_time = <string>
* The string that represents the time of the earliest event to return,
  inclusive.
* The time can be expressed with a relative time identifier or in epoch time.
* Optional. If omitted, no earliest time bound is used.

latest_time = <string>
* The string that represents the time of the latest event to return,
  inclusive.
* The time can be expressed with a relative time identifier or in epoch
  time.
* Optional. If omitted, no latest time bound is used. NOTE: events that
  occur in the future (relative to the server timezone) may be returned.

order = <integer>
* The key on which all custom time ranges are sorted, ascending.
* The default time range selector in the UI will merge and sort all time
  ranges according to the 'order' key, and then alphabetically.
* Optional. Default value is 0.
```

```

sub_menu = <submenu name>
* If present, the time range is to be shown in the given submenu instead
  of in the main menu.
* The value for this key must be the label key of an existing stanza name,
  and that stanza name must have an is_sub_menu = True key
* Optional. If omitted the given time option will display in the main menu.

is_sub_menu = <boolean>
* If True, the given item is only the 'opener' element for a submenu.
* Stanzas containing this key can still be assigned an order value to set
  the placement within the main menu, but can not themselves have
  latest_time nor earliest_time keys.

```

## times.conf.example

```

# Version 6.5.0
#
# This is an example times.conf. Use this file to create custom time ranges
# that can be used while interacting with the search system.
#
# To use one or more of these configurations, copy the configuration block
# into times.conf in $SPLUNK_HOME/etc/system/local/. You must restart Splunk
# to enable configurations.
#
# To learn more about configuration files (including precedence) please see
# the documentation located at
# http://docs.splunk.com/Documentation/Splunk/latest/Admin/Aboutconfigurationfiles
#
# Note: These are examples. Replace the values with your own customizations.

# The stanza name is an alphanumeric string (no spaces) that uniquely
# identifies a time range.
[this_business_week]

# Define the label used in the time range control
label = This business week

# Define the label to be used in display headers. If omitted the 'label' key
# will be used with the first letter lowercased.
header_label = during this business week
earliest_time = +1d@w1
latest_time = +6d@w6

# Define the ordering sequence of this time range. All time ranges are
# sorted numerically, ascending. If the time range is in a sub menu and not
# in the main menu, this will determine the position within the sub menu.
order = 110

# a time range that only has a bound on the earliest time
#
[last_3_hours]
label = Last 3 hours
header_label = in the last 3 hours
earliest_time = -3h
order = 30

# Use epoch time notation to define the time bounds for the Fall Semester
# 2013, where earliest_time is 9/4/13 00:00:00 and latest_time is 12/13/13
# 00:00:00.
#
[Fall_2013]
label = Fall Semester 2013
earliest_time = 1378278000
latest_time = 1386921600

```

```

# two time ranges that should appear in a sub menu instead of in the main
# menu. the order values here determine relative ordering within the
# submenu.
#
[yesterday]
label = Yesterday
earliest_time = -1d@d
latest_time = @d
order = 10
sub_menu = Other options

[day_before_yesterday]
label = Day before yesterday
header_label = from the day before yesterday
earliest_time = -2d@d
latest_time = -1d@d
order = 20
sub_menu = Other options

#
# The sub menu item that should contain the previous two time ranges. The
# order key here determines the submenu opener's placement within the main
# menu.
#
[other]
label = Other options
order = 202

```

## transactiontypes.conf

以下为 transactiontypes.conf 的规范和示例文件。

### transactiontypes.conf.spec

```

# Version 6.5.0
#
# This file contains all possible attributes and value pairs for a
# transactiontypes.conf file. Use this file to configure transaction searches
# and their properties.
#
# There is a transactiontypes.conf in $SPLUNK_HOME/etc/system/default/. To set
# custom configurations, place a transactiontypes.conf in
# $SPLUNK_HOME/etc/system/local/. You must restart Splunk to enable
# configurations.
#
# To learn more about configuration files (including precedence) please see the
# documentation located at
# http://docs.splunk.com/Documentation/Splunk/latest/Admin/Aboutconfigurationfiles

```

### 全局设置

```

# GLOBAL SETTINGS
# Use the [default] stanza to define any global settings.
# * You can also define global settings outside of any stanza, at the top of
# the file.
# * Each conf file should have at most one default stanza. If there are
# multiple default stanzas, attributes are combined. In the case of
# multiple definitions of the same attribute, the last definition in the
# file wins.
# * If an attribute is defined at both the global level and in a specific
# stanza, the value in the specific stanza takes precedence.

[<TRANSACTIONTYPE>]
* Create any number of transaction types, each represented by a stanza name and
any number of the following attribute/value pairs.

```

\* Use the stanza name, [<TRANSACTIONTYPE>], to search for the transaction in Splunk Web.

\* If you do not specify an entry for each of the following attributes, Splunk uses the default value.

maxspan = [<integer> s|m|h|d|-1]

\* Set the maximum time span for the transaction.

\* Can be in seconds, minutes, hours, or days, or -1 for an unlimited timespan.

\* For example: 5s, 6m, 12h or 30d.

\* Defaults to: maxspan=-1

maxpause = [<integer> s|m|h|d|-1]

\* Set the maximum pause between the events in a transaction.

\* Can be in seconds, minutes, hours, or days, or -1 for an unlimited pause.

\* For example: 5s, 6m, 12h or 30d.

\* Defaults to: maxpause=-1

maxevents = <integer>

\* The maximum number of events in a transaction. This constraint is disabled if the value is a negative integer.

\* Defaults to: maxevents=1000

fields = <comma-separated list of fields>

\* If set, each event must have the same field(s) to be considered part of the same transaction.

\* For example: fields=host,cookie

\* Defaults to: ""

connected=[true|false]

\* Relevant only if fields (see above) is not empty. Controls whether an event that is not inconsistent and not consistent with the fields of a transaction opens a new transaction (connected=true) or is added to the transaction.

\* An event can be not inconsistent and not field-consistent if it contains fields required by the transaction but none of these fields has been instantiated in the transaction (by a previous event addition).

\* Defaults to: connected=true

startswith=<transam-filter-string>

\* A search or eval filtering expression which, if satisfied by an event, marks the beginning of a new transaction.

\* For example:

\* startswith="login"

\* startswith=(username=foobar)

\* startswith=eval(speed\_field < max\_speed\_field)

\* startswith=eval(speed\_field < max\_speed\_field/12)

\* Defaults to: ""

endswith=<transam-filter-string>

\* A search or eval filtering expression which, if satisfied by an event, marks the end of a transaction.

\* For example:

\* endswith="logout"

\* endswith=(username=foobar)

\* endswith=eval(speed\_field > max\_speed\_field)

\* endswith=eval(speed\_field > max\_speed\_field/12)

\* Defaults to: ""

\* For startswith/endswith <transam-filter-string> has the following syntax:

\* syntax: "<search-expression>" | (<quoted-search-expression>) | eval(<eval-expression>)

\* Where:

\* <search-expression> is a valid search expression that does not contain quotes

\* <quoted-search-expression> is a valid search expression that contains quotes

\* <eval-expression> is a valid eval expression that evaluates to a boolean. For example, startswith=eval(foo<bar\*2) will match events where foo is less than 2 x bar.

\* Examples:

\* "<search expression>": startswith="foo bar"

\* <quoted-search-expression>: startswith=(name="mildred")

\* <quoted-search-expression>: startswith=("search literal")

\* eval(<eval-expression>): startswith=eval(distance/time < max\_speed)

### memory constraint options ###

```

maxopentxn=<int>
* Specifies the maximum number of not yet closed transactions to keep in the
  open pool. When this limit is surpassed, Splunk begins evicting transactions
  using LRU (least-recently-used memory cache algorithm) policy.
* The default value of this attribute is read from the transactions stanza in
  limits.conf.

maxopenevents=<int>
* Specifies the maximum number of events (can be) part of open transactions.
  When this limit is surpassed, Splunk begins evicting transactions using LRU
  (least-recently-used memory cache algorithm) policy.
* The default value of this attribute is read from the transactions stanza in
  limits.conf.

keepevicted=<bool>
* Whether to output evicted transactions. Evicted transactions can be
  distinguished from non-evicted transactions by checking the value of the
  'evicted' field, which is set to '1' for evicted transactions.
* Defaults to: keepevicted=false

### multivalue rendering options ###

mvlist=<bool>|<field-list>
* Field controlling whether the multivalued fields of the transaction are (1) a
  list of the original events ordered in arrival order or (2) a set of unique
  field values ordered lexicographically. If a comma/space delimited list of
  fields is provided only those fields are rendered as lists
* Defaults to: mvlist=f

delim=<string>
* A string used to delimit the original event values in the transaction event
  fields.
* Defaults to: delim=" "

nullstr=<string>
* The string value to use when rendering missing field values as part of mv
  fields in a transaction.
* This option applies only to fields that are rendered as lists.
* Defaults to: nullstr=NULL

### values only used by the searchtxn search command ###

search=<string>
* A search string used to more efficiently seed transactions of this type.
* The value should be as specific as possible, to limit the number of events
  that must be retrieved to find transactions.
* Example: sourcetype="sendmaill_sendmail"
* Defaults to "*" (all events)

```

## transactiontypes.conf.example

```

# Version 6.5.0
#
# This is an example transactiontypes.conf. Use this file as a template to
# configure transactions types.
#
# To use one or more of these configurations, copy the configuration block into
# transactiontypes.conf in $SPLUNK_HOME/etc/system/local/.
#
# To learn more about configuration files (including precedence) please see the
# documentation located at
# http://docs.splunk.com/Documentation/Splunk/latest/Admin/Aboutconfigurationfiles
#
[default]
maxspan = 5m
maxpause = 2s
match = closest

[purchase]
maxspan = 10m

```



```
maxpause = 5m
fields = userid
```

## transforms.conf

以下为 transforms.conf 的规范和示例文件。

### transforms.conf.spec

```
# Version 6.5.0
#
# This file contains attributes and values that you can use to configure
# data transformations. and event signing in transforms.conf.
#
# Transforms.conf is commonly used for:
# * Configuring regex-based host and source type overrides.
# * Anonymizing certain types of sensitive incoming data, such as credit
#   card or social security numbers.
# * Routing specific events to a particular index, when you have multiple
#   indexes.
# * Creating new index-time field extractions. NOTE: We do not recommend
#   adding to the set of fields that are extracted at index time unless it
#   is absolutely necessary because there are negative performance
#   implications.
# * Creating advanced search-time field extractions that involve one or more
#   of the following:
#   * Reuse of the same field-extracting regular expression across multiple
#     sources, source types, or hosts.
#   * Application of more than one regex to the same source, source type, or
#     host.
#   * Using a regex to extract one or more values from the values of another
#     field.
#   * Delimiter-based field extractions (they involve field-value pairs that
#     are separated by commas, colons, semicolons, bars, or something
#     similar).
#   * Extraction of multiple values for the same field (multivalued field
#     extraction).
#   * Extraction of fields with names that begin with numbers or
#     underscores.
#   * NOTE: Less complex search-time field extractions can be set up
#     entirely in props.conf.
# * Setting up lookup tables that look up fields from external sources.
#
# All of the above actions require corresponding settings in props.conf.
#
# You can find more information on these topics by searching the Splunk
# documentation (http://docs.splunk.com/Documentation)
#
# There is a transforms.conf file in $SPLUNK_HOME/etc/system/default/. To
# set custom configurations, place a transforms.conf
# $SPLUNK_HOME/etc/system/local/. For examples, see the
# transforms.conf.example file.
#
# You can enable configurations changes made to transforms.conf by typing
# the following search string in Splunk Web:
#
# | extract reload=t
#
# To learn more about configuration files (including precedence) please see
# the documentation located at
# http://docs.splunk.com/Documentation/Splunk/latest/Admin/Aboutconfigurationfiles
```

### 全局设置

```
# GLOBAL SETTINGS
# Use the [default] stanza to define any global settings.
```

```
# * You can also define global settings outside of any stanza, at the top
#   of the file.
# * Each conf file should have at most one default stanza. If there are
#   multiple default stanzas, attributes are combined. In the case of
#   multiple definitions of the same attribute, the last definition in the
#   file wins.
# * If an attribute is defined at both the global level and in a specific
#   stanza, the value in the specific stanza takes precedence.
```

[<unique\_transform\_stanza\_name>]

- \* Name your stanza. Use this name when you configure field extractions, lookup tables, and event routing in props.conf. For example, if you are setting up an advanced search-time field extraction, in props.conf you would add REPORT-<class> = <unique\_transform\_stanza\_name> under the [<spec>] stanza that corresponds with a stanza you've created in transforms.conf.
- \* Follow this stanza name with any number of the following attribute/value pairs, as appropriate for what you intend to do with the transform.
- \* If you do not specify an entry for each attribute, Splunk uses the default value.

REGEX = <regular expression>

- \* Enter a regular expression to operate on your data.
- \* NOTE: This attribute is valid for both index-time and search-time field extraction.
- \* REGEX is required for all search-time transforms unless you are setting up a delimiter-based field extraction, in which case you use DELIMS (see the DELIMS attribute description, below).
- \* REGEX is required for all index-time transforms.
- \* REGEX and the FORMAT attribute:
  - \* Name-capturing groups in the REGEX are extracted directly to fields. This means that you do not need to specify the FORMAT attribute for simple field extraction cases (see the description of FORMAT, below).
  - \* If the REGEX extracts both the field name and its corresponding field value, you can use the following special capturing groups if you want to skip specifying the mapping in FORMAT:
    - \_KEY\_<string>, \_VAL\_<string>.
  - \* For example, the following are equivalent:
    - \* Using FORMAT:
      - \* REGEX = ([a-z]+)([a-z]+)
      - \* FORMAT = \$1::\$2
    - \* Without using FORMAT
      - \* REGEX = (?<\_KEY\_1>[a-z]+)(?<\_VAL\_1>[a-z]+)
  - \* When using either of the above formats, in a search-time extraction, the regex will continue to match against the source text, extracting as many fields as can be identified in the source text.
- \* Defaults to an empty string.

FORMAT = <string>

- \* NOTE: This option is valid for both index-time and search-time field extraction. However, FORMAT behaves differently depending on whether the extraction is performed at index time or search time.
- \* This attribute specifies the format of the event, including any field names or values you want to add.
- \* FORMAT for index-time extractions:
  - \* Use \$n (for example \$1, \$2, etc) to specify the output of each REGEX match.
  - \* If REGEX does not have n groups, the matching fails.
  - \* The special identifier \$0 represents what was in the DEST\_KEY before the REGEX was performed.
  - \* At index time only, you can use FORMAT to create concatenated fields:
    - \* Example: FORMAT = ipaddress::\$1.\$2.\$3.\$4
  - \* When you create concatenated fields with FORMAT, "\$" is the only special character. It is treated as a prefix for regex-capturing groups only if it is followed by a number and only if the number applies to an existing capturing group. So if REGEX has only one capturing group and its value is "bar", then:
    - \* "FORMAT = foo\$1" yields "foobar"
    - \* "FORMAT = foo\$bar" yields "foo\$bar"
    - \* "FORMAT = foo\$1234" yields "foo\$1234"
    - \* "FORMAT = foo\$1\$2" yields "foobar\$2"

```

* At index-time, FORMAT defaults to <stanza-name>:::$1
* FORMAT for search-time extractions:
* The format of this field as used during search time extractions is as follows:
* FORMAT = <field-name>:::<field-value> ( <field-name>:::<field-value>)*
  where:
  * field-name = [<string>|<$<extracting-group-number>]
  * field-value = [<string>|<$<extracting-group-number>]
* Search-time extraction examples:
  * 1. FORMAT = first::$1 second::$2 third::other-value
  * 2. FORMAT = $1::$2
* If the key-name of a FORMAT setting is varying, for example $1 in the example 2 just above, then the regex will continue to match against the source key to extract as many matches as are present in the text.
* NOTE: You cannot create concatenated fields with FORMAT at search time. That functionality is only available at index time.
* At search-time, FORMAT defaults to an empty string.

CLONE_SOURCETYPE = <string>
* This name is wrong; a transform with this setting actually clones and modifies events, and assigns the new events the specified sourcetype.

* If CLONE_SOURCETYPE is used as part of a transform, the transform will create a modified duplicate event, for all events that the transform is applied to via normal props.conf rules.
* Use this feature if you need to store both the original and a modified form of the data in your system, or if you want to send the original and a modified form to different outbound systems.
  * A typical example would be to retain sensitive information according to one policy and a version with the sensitive information removed according to another policy. For example, some events may have data that you must retain for 30 days (such as personally identifying information) and only 30 days with restricted access, but you need that event retained without the sensitive data for a longer time with wider access.
* Specifically, for each event handled by this transform, a near-exact copy is made of the original event, and the transformation is applied to the copy. The original event will continue along normal data processing unchanged.
* The <string> used for CLONE_SOURCETYPE selects the sourcetype that will be used for the duplicated events.
* The new sourcetype MUST differ from the the original sourcetype. If the original sourcetype is the same as the target of the CLONE_SOURCETYPE, Splunk will make a best effort to log warnings to splunkd.log, but this setting will be silently ignored at runtime for such cases, causing the transform to be applied to the original event without cloning.
* The duplicated events will receive index-time transformations & sed commands all transforms which match its new host/source/sourcetype.
  * This means that props matching on host or source will incorrectly be applied a second time. (SPL-99120)
* Can only be used as part of of an otherwise-valid index-time transform. For example REGEX is required, there must be a valid target (DEST_KEY or WRITE_META), etc as above.

LOOKAHEAD = <integer>
* NOTE: This option is valid for all index time transforms, such as index-time field creation, or DEST_KEY modifications.
* Optional. Specifies how many characters to search into an event.
* Defaults to 4096.
* You may want to increase this value if you have event line lengths that exceed 4096 characters (before linebreaking).

WRITE_META = [true|false]
* NOTE: This attribute is only valid for index-time field extractions.
* Automatically writes REGEX to metadata.
* Required for all index-time field extractions except for those where DEST_KEY = _meta (see the description of the DEST_KEY attribute, below)
* Use instead of DEST_KEY = _meta.
* Defaults to false.

DEST_KEY = <KEY>
* NOTE: This attribute is only valid for index-time field extractions.

```

- \* Specifies where Splunk stores the expanded FORMAT results in accordance with the REGEX match.
- \* Required for index-time field extractions where WRITE\_META = false or is not set.
- \* For index-time extractions, DEST\_KEY can be set to a number of values mentioned in the KEYS section at the bottom of this file.
  - \* If DEST\_KEY = \_meta (not recommended) you should also add \$0 to the start of your FORMAT attribute. \$0 represents the DEST\_KEY value before Splunk performs the REGEX (in other words, \_meta).
  - \* The \$0 value is in no way derived \*from\* the REGEX match. (It does not represent a captured group.)
- \* KEY names are case-sensitive, and should be used exactly as they appear in the KEYS list at the bottom of this file. (For example, you would say DEST\_KEY = MetaData:Host, \*not\* DEST\_KEY = metadata:host .)

DEFAULT\_VALUE = <string>

- \* NOTE: This attribute is only valid for index-time field extractions.
- \* Optional. Splunk writes the DEFAULT\_VALUE to DEST\_KEY if the REGEX fails.
- \* Defaults to empty.

SOURCE\_KEY = <string>

- \* NOTE: This attribute is valid for both index-time and search-time field extractions.
- \* Optional. Defines the KEY that Splunk applies the REGEX to.
- \* For search time extractions, you can use this attribute to extract one or more values from the values of another field. You can use any field that is available at the time of the execution of this field extraction
- \* For index-time extractions use the KEYS described at the bottom of this file.
  - \* KEYS are case-sensitive, and should be used exactly as they appear in the KEYS list at the bottom of this file. (For example, you would say SOURCE\_KEY = MetaData:Host, \*not\* SOURCE\_KEY = metadata:host .)
- \* If <string> starts with "field:" or "fields:" the meaning is changed. Instead of looking up a KEY, it instead looks up an already indexed field. For example, if a CSV field name "price" was indexed then "SOURCE\_KEY = field:price" causes the REGEX to match against the contents of that field. It's also possible to list multiple fields here with "SOURCE\_KEY = fields:name1,name2,name3" which causes MATCH to be run against a string comprising of all three values, separated by space characters.
- \* SOURCE\_KEY is typically used in conjunction with REPEAT\_MATCH in index-time field transforms.
- \* Defaults to \_raw, which means it is applied to the raw, unprocessed text of all events.

REPEAT\_MATCH = [true|false]

- \* NOTE: This attribute is only valid for index-time field extractions.
- \* Optional. When set to true Splunk runs the REGEX multiple times on the SOURCE\_KEY.
- \* REPEAT\_MATCH starts wherever the last match stopped, and continues until no more matches are found. Useful for situations where an unknown number of REGEX matches are expected per event.
- \* Defaults to false.

DELIMS = <quoted string list>

- \* NOTE: This attribute is only valid for search-time field extractions.
- \* IMPORTANT: If a value may contain an embedded unescaped double quote character, such as "foo"bar", use REGEX, not DELIMS. An escaped double quote (\") is ok.
- \* Optional. Used in place of REGEX when dealing with delimiter-based field extractions, where field values (or field/value pairs) are separated by delimiters such as colons, spaces, line breaks, and so on.
- \* Sets delimiter characters, first to separate data into field/value pairs, and then to separate field from value.
- \* Each individual character in the delimiter string is used as a delimiter to split the event.
- \* Delimiters must be quoted with " " (use \ to escape).
- \* When the event contains full delimiter-separated field/value pairs, you enter two sets of quoted characters for DELIMS:
  - \* The first set of quoted delimiters extracts the field/value pairs.
  - \* The second set of quoted delimiters separates the field name from its corresponding value.

- \* When the event only contains delimiter-separated values (no field names) you use just one set of quoted delimiters to separate the field values. Then you use the `FIELDS` attribute to apply field names to the extracted values (see `FIELDS`, below).
- \* Alternately, Splunk reads even tokens as field names and odd tokens as field values.
- \* Splunk consumes consecutive delimiter characters unless you specify a list of field names.
- \* The following example of `DELIMS` usage applies to an event where field/value pairs are separated by `'|'` symbols and the field names are separated from their corresponding values by `'='` symbols:
 

```
[pipe_eq]
  DELIMS = "|", "="
```
- \* Defaults to `""`.

`FIELDS = <quoted string list>`

- \* NOTE: This attribute is only valid for search-time field extractions.
- \* Used in conjunction with `DELIMS` when you are performing delimiter-based field extraction and only have field values to extract.
- \* `FIELDS` enables you to provide field names for the extracted field values, in list format according to the order in which the values are extracted.
- \* NOTE: If field names contain spaces or commas they must be quoted with `" "` (to escape, use `\`).
- \* The following example is a delimiter-based field extraction where three field values appear in an event. They are separated by a comma and then a space.
 

```
[commalist]
  DELIMS = ", "
  FIELDS = field1, field2, field3
```
- \* Defaults to `""`.

`MV_ADD = [true|false]`

- \* NOTE: This attribute is only valid for search-time field extractions.
- \* Optional. Controls what the extractor does when it finds a field which already exists.
- \* If set to `true`, the extractor makes the field a multivalued field and appends the newly found value, otherwise the newly found value is discarded.
- \* Defaults to `false`

`CLEAN_KEYS = [true|false]`

- \* NOTE: This attribute is only valid for search-time field extractions.
- \* Optional. Controls whether Splunk "cleans" the keys (field names) it extracts at search time.
 

"Key cleaning" is the practice of replacing any non-alphanumeric characters (characters other than those falling between the `a-z`, `A-Z`, or `0-9` ranges) in field names with underscores, as well as the stripping of leading underscores and `0-9` characters from field names.
- \* Add `CLEAN_KEYS = false` to your transform if you need to extract field names that include non-alphanumeric characters, or which begin with underscores or `0-9` characters.
- \* Defaults to `true`.

`KEEP_EMPTY_VALS = [true|false]`

- \* NOTE: This attribute is only valid for search-time field extractions.
- \* Optional. Controls whether Splunk keeps field/value pairs when the value is an empty string.
- \* This option does not apply to field/value pairs that are generated by Splunk's autokv extraction. Autokv ignores field/value pairs with empty values.
- \* Defaults to `false`.

`CAN_OPTIMIZE = [true|false]`

- \* NOTE: This attribute is only valid for search-time field extractions.
- \* Optional. Controls whether Splunk can optimize this extraction out (another way of saying the extraction is disabled).
- \* You might use this if you're running searches under a Search Mode setting that disables field discovery--it ensures that Splunk *\*always\** discovers specific fields.
- \* Splunk only disables an extraction if it can determine that none of the fields identified by the extraction will ever be needed for the successful evaluation of a search.

\* NOTE: This option should be rarely set to false.  
\* Defaults to true.

## 查找表

```
#*****
# Lookup tables
#*****Lookup tables
# NOTE: Lookup tables are used ONLY during search time

filename = <string>
* Name of static lookup file.
* File should be in $SPLUNK_HOME/etc/<app_name>/lookups/ for some <app_name>, or in
  $SPLUNK_HOME/etc/system/lookups/
* If file is in multiple 'lookups' directories, no layering is done.
* Standard conf file precedence is used to disambiguate.
* Defaults to empty string.

collection = <string>
* Name of the collection to use for this lookup.
* Collection should be defined in $SPLUNK_HOME/etc/<app_name>/collections.conf
  for some <app_name>
* If collection is in multiple collections.conf file, no layering is done.
* Standard conf file precedence is used to disambiguate.
* Defaults to empty string (in which case the name of the stanza is used).

max_matches = <integer>
* The maximum number of possible matches for each input lookup value
  (range 1 - 1000).
* If the lookup is non-temporal (not time-bounded, meaning the time_field
  attribute is not specified), Splunk uses the first <integer> entries, in
  file order.
* If the lookup is temporal, Splunk uses the first <integer> entries in
  descending time order. In other words, up <max_matches> lookup entries
  will be allowed to match, and if more than this many the ones nearest to
  the lookup value will be used.
* Default = 1000 if the lookup is not temporal, default = 1 if it is
  temporal.

min_matches = <integer>
* Minimum number of possible matches for each input lookup value.
* Default = 0 for both temporal and non-temporal lookups, which means that
  Splunk outputs nothing if it cannot find any matches.
* However, if min_matches > 0, and Splunk get less than min_matches, then
  Splunk provides the default_match value provided (see below).

default_match = <string>
* If min_matches > 0 and Splunk has less than min_matches for any given
  input, it provides this default_match value one or more times until the
  min_matches threshold is reached.
* Defaults to empty string.

case_sensitive_match = <bool>
* NOTE: This attribute is not valid for KV Store-based lookups.
* If set to false, case insensitive matching will be performed for all
  fields in a lookup table
* Defaults to true (case sensitive matching)

match_type = <string>
* A comma and space-delimited list of <match_type>(<field_name>)
  specification to allow for non-exact matching
* The available match_type values are WILDCARD, CIDR, and EXACT. EXACT is
  the default and does not need to be specified. Only fields that should
  use WILDCARD or CIDR matching should be specified in this list

external_cmd = <string>
* Provides the command and arguments to invoke to perform a lookup. Use this
  for external (or "scripted") lookups, where you interface with with an
  external script rather than a lookup table.
```

- \* This string is parsed like a shell command.
- \* The first argument is expected to be a python script (or executable file) located in \$SPLUNK\_HOME/etc/<app\_name>/bin (or ../etc/searchscripts).
- \* Presence of this field indicates that the lookup is external and command based.
- \* Defaults to empty string.

fields\_list = <string>

- \* A comma- and space-delimited list of all fields that are supported by the external command.

external\_type = [python|executable|kvstore|geo]

- \* Type of external command.
- \* "python" a python script
- \* "executable" a binary executable
- \* "geo" a point-in-polygon lookup
- \* Defaults to "python".

time\_field = <string>

- \* Used for temporal (time bounded) lookups. Specifies the name of the field in the lookup table that represents the timestamp.
- \* Defaults to an empty string, meaning that lookups are not temporal by default.

time\_format = <string>

- \* For temporal lookups this specifies the 'strptime' format of the timestamp field.
- \* You can include subseconds but Splunk will ignore them.
- \* Defaults to %s.%Q or seconds from unix epoch in UTC an optional milliseconds.

max\_offset\_secs = <integer>

- \* For temporal lookups, this is the maximum time (in seconds) that the event timestamp can be later than the lookup entry time for a match to occur.
- \* Default is 2000000000 (no maximum, effectively).

min\_offset\_secs = <integer>

- \* For temporal lookups, this is the minimum time (in seconds) that the event timestamp can be later than the lookup entry timestamp for a match to occur.
- \* Defaults to 0.

batch\_index\_query = <bool>

- \* For large file based lookups, this determines whether queries can be grouped to improve search performance.
- \* Default is unspecified here, but defaults to true (at global level in limits.conf)

allow\_caching = <bool>

- \* Allow output from lookup scripts to be cached
- \* Default is true

max\_ext\_batch = <integer>

- \* The maximum size of external batch (range 1 - 1000).
- \* Only used with kvstore.
- \* Default = 300.

filter = <string>

- \* Filter results from the lookup table before returning data. Create this filter like you would a typical search query using Boolean expressions and/or comparison operators.
- \* For KV Store lookups, filtering is done when data is initially retrieved to improve performance.
- \* For CSV lookups, filtering is done in memory.

feature\_id\_element = <string>

- \* If lookup file is a kmz file, this field can be used to specify the xml path from placemark down to the name of this placemark.
- \* Default = /Placemark/name
- \* ONLY for Kmz files

**鍵：**

```

#*****
# KEYS:
#*****KEYS:
* NOTE: Keys are case-sensitive. Use the following keys exactly as they
      appear.

queue : Specify which queue to send the event to (can be nullQueue, indexQueue).
      * indexQueue is the usual destination for events going through the
        transform-handling processor.
      * nullQueue is a destination which will cause the events to be
        dropped entirely.
_raw  : The raw text of the event.
_meta : A space-separated list of metadata for an event.
_time : The timestamp of the event, in seconds since 1/1/1970 UTC.

MetaData:Host      : The host associated with the event.
                   The value must be prefixed by "host::"

_metaData:Index    : The index where the event should be stored.

MetaData:Source    : The source associated with the event.
                   The value must be prefixed by "source::"

MetaData:Sourcetype : The sourcetype of the event.
                   The value must be prefixed by "sourcetype::"

_TCP_ROUTING      : Comma separated list of tcpout group names (from outputs.conf)
                   Defaults to groups present in 'defaultGroup' for [tcpout].

_SYSLOG_ROUTING    : Comma separated list of syslog-stanza names (from outputs.conf)
                   Defaults to groups present in 'defaultGroup' for [syslog].

* NOTE: Any KEY (field name) prefixed by '_' is not indexed by Splunk, in general.

[accepted_keys]

<name> = <key>

* Modifies Splunk's list of key names it considers valid when automatically
  checking your transforms for use of undocumented SOURCE_KEY or DEST_KEY
  values in index-time transformations.
* By adding entries to [accepted_keys], you can tell Splunk that a key that
  is not documented is a key you intend to work for reasons that are valid
  in your environment / app / etc.
* The 'name' element is simply used to disambiguate entries, similar
  to -class entries in props.conf. The name can be anything of your
  choosing, including a descriptive name for why you use the key.
* The entire stanza defaults to not being present, causing all keys not
  documented just above to be flagged.

```

## transforms.conf.example

```

#   Version 6.5.0
#
# This is an example transforms.conf. Use this file to create regexes and
# rules for transforms. Use this file in tandem with props.conf.
#
# To use one or more of these configurations, copy the configuration block
# into transforms.conf in $SPLUNK_HOME/etc/system/local/. You must restart
# Splunk to enable configurations.
#
# To learn more about configuration files (including precedence) please see
# the documentation located at
# http://docs.splunk.com/Documentation/Splunk/latest/Admin/Aboutconfigurationfiles
#
# Note: These are examples. Replace the values with your own customizations.

# Indexed field:

```



```

[netscreen-error]
REGEX = device_id=\[w+\](?<err_code>[^:]+)
FORMAT = err_code::$1
WRITE_META = true

# Override host:

[hostoverride]
DEST_KEY = MetaData:Host
REGEX = \s(\w*)$
FORMAT = host::$1

# Extracted fields:

[netscreen-error-field]
REGEX = device_id=\[w+\](?<err_code>[^:]+)
FORMAT = err_code::$1

# Static lookup table

[mylookuptable]
filename = mytable.csv

# one to one lookup
# guarantees that we output a single lookup value for each input value, if
# no match exists, we use the value of "default_match", which by default is
# "NONE"
[mylook]
filename = mytable.csv
max_matches = 1
min_matches = 1
default_match = nothing

# Lookup and filter results
[myfilteredlookup]
filename = mytable.csv
filter = id<500 AND color="red"

# external command lookup table

[myexternaltable]
external_cmd = testadapter.py blah
fields_list = foo bar

# Temporal based static lookup table

[staticwtime]
filename = mytable.csv
time_field = timestamp
time_format = %d/%m/%y %H:%M:%S

# Mask sensitive data:

[session-anonymizer]
REGEX = (?m)^(.*)SessionId=\w+(\w{4}[\&"].*)$
FORMAT = $1SessionId=#####$2
DEST_KEY = _raw

# Route to an alternate index:

[AppRedirect]
REGEX = Application
DEST_KEY = _MetaData:Index
FORMAT = Verbose

```

```

# Extract comma-delimited values into fields:

[extract_csv]
DELIMS = ","
FIELDS = "field1", "field2", "field3"

# This example assigns the extracted values from _raw to field1, field2 and
# field3 (in order of extraction). If more than three values are extracted
# the values without a matching field name are ignored.

[pipe_eq]
DELIMS = "|", "="

# The above example extracts key-value pairs which are separated by '|'
# while the key is delimited from value by '='.

[multiple_delims]
DELIMS = ";", "=: "

# The above example extracts key-value pairs which are separated by '|' or
# ';', while the key is delimited from value by '=' or ':'.

##### BASIC MODULAR REGULAR EXPRESSIONS DEFINITION START #####
# When adding a new basic modular regex PLEASE add a comment that lists
# the fields that it extracts (named capturing groups), or whether it
# provides a placeholder for the group name as:
# Extracts: field1, field2...
#

[all_lazy]
REGEX = .*?

[all]
REGEX = .*

[nspaces]
# matches one or more NON space characters
REGEX = \S+

[alphas]
# matches a string containing only letters a-zA-Z
REGEX = [a-zA-Z]+

[alnums]
# matches a string containing letters + digits
REGEX = [a-zA-Z0-9]+

[qstring]
# matches a quoted "string" - extracts an unnamed variable
# name MUST be provided as in [[qstring:name]]
# Extracts: empty-name-group (needs name)
REGEX = "(?<>[^\"]*)"

[sbstring]
# matches a string enclosed in [] - extracts an unnamed variable
# name MUST be provided as in [[sbstring:name]]
# Extracts: empty-name-group (needs name)
REGEX = "\[(?<>[^\]]*)\]"

[digits]
REGEX = \d+

[int]
# matches an integer or a hex number
REGEX = 0x[a-fA-F0-9]+\|\d+

[float]
# matches a float (or an int)
REGEX = \d*\.\d+|[[int]]

```

```

[octet]
# this would match only numbers from 0-255 (one octet in an ip)
REGEX = (?:(?:25[0-5]|2[0-4][0-9]|1[0-9][0-9]|0[0-9][0-9])?)

[ipv4]
# matches a valid IPv4 optionally followed by :port_num the octets in the ip
# would also be validated 0-255 range
# Extracts: ip, port
REGEX = (<?ip>[[octet]](?:\.[[octet]]){3})(?::[int:port])?

[simple_url]
# matches a url of the form proto://domain.tld/uri
# Extracts: url, domain
REGEX = (<?url>\w+:\/\/(?:<domain>[a-zA-Z0-9\-.:~]+)(?:/[^\s"]*)?)

[url]
# matches a url of the form proto://domain.tld/uri
# Extracts: url, proto, domain, uri
REGEX = (<?url>[[alphas:proto]]:\/\/(?:<domain>[a-zA-Z0-9\-.:~]+)(?<uri>/[^\s"]*)?)

[simple_uri]
# matches a uri of the form /path/to/resource?query
# Extracts: uri, uri_path, uri_query
REGEX = (<?uri>(?(<uri_path>[^\s\?"]++)(?:\?(?<uri_query>[^\s"]+))?)

[uri]
# uri = path optionally followed by query [/this/path/file.js?query=part&other=var]
# path = root part followed by file [/root/part/file.part]
# Extracts: uri, uri_path, uri_root, uri_file, uri_query, uri_domain (optional if in proxy mode)
REGEX = (<?uri>(?:\w+:\/\/(?:<uri_domain>[^\s"]++)(?<uri_path>(?(<uri_root>/+([^\s\?;/=]*+/*)*)(<uri_file>[^\s\?;/=]*+)))(?:\?(?<uri_query>[^\s"]+))?)

[hide-ip-address]
# Make a clone of an event with the sourcetype masked_ip_address. The clone
# will be modified; its text changed to mask the ip address.
# The cloned event will be further processed by index-time transforms and
# SEDCMD expressions according to its new sourcetype.
# In most scenarios an additional transform would be used to direct the
# masked_ip_address event to a different index than the original data.
REGEX = ^(.*)src=\d+\.\d+\.\d+\.\d+(.*)$
FORMAT = $1src=XXXXX$2
DEST_KEY = _raw
CLONE_SOURCETYPE = masked_ip_addresses

##### BASIC MODULAR REGULAR EXPRESSIONS DEFINITION END #####

```

## ui-prefs.conf

以下为 ui-prefs.conf 的规范和示例文件。

### ui-prefs.conf.spec

```

# Version 6.5.0
#
# This file contains possible attribute/value pairs for ui preferences for a
# view.
#
# There is a default ui-prefs.conf in $SPLUNK_HOME/etc/system/default. To set
# custom configurations, place a ui-prefs.conf in
# $SPLUNK_HOME/etc/system/local/. To set custom configuration for an app, place
# ui-prefs.conf in $SPLUNK_HOME/etc/apps/<app_name>/local/. For examples, see
# ui-prefs.conf.example. You must restart Splunk to enable configurations.
#
# To learn more about configuration files (including precedence) please see the
# documentation located at
# http://docs.splunk.com/Documentation/Splunk/latest/Admin/Aboutconfigurationfiles

```

## 全局设置

```
# GLOBAL SETTINGS
# Use the [default] stanza to define any global settings.
# * You can also define global settings outside of any stanza, at the top of
#   the file.
# * Each conf file should have at most one default stanza. If there are
#   multiple default stanzas, attributes are combined. In the case of
#   multiple definitions of the same attribute, the last definition in the
#   file wins.
# * If an attribute is defined at both the global level and in a specific
#   stanza, the value in the specific stanza takes precedence.

[<stanza name>]
* Stanza name is the name of the xml view file

dispatch.earliest_time =
dispatch.latest_time =

# Pref only options
display.prefs.autoOpenSearchAssistant = 0 | 1
display.prefs.timeline.height = <string>
display.prefs.timeline.minimized = 0 | 1
display.prefs.timeline.minimalMode = 0 | 1
display.prefs.aclFilter = [none|app|owner]
display.prefs.appFilter = <string>
display.prefs.listMode = [tiles|table]
display.prefs.searchContext = <string>
display.prefs.events.count = [10|20|50]
display.prefs.statistics.count = [10|20|50|100]
display.prefs.fieldCoverage = [0|.01|.50|.90|1]
display.prefs.enableMetaData = 0 | 1
display.prefs.showDataSummary = 0 | 1
display.prefs.customSampleRatio = <int>
display.prefs.showSPL = 0 | 1
display.prefs.livetail = 0 | 1

# count per page for listing pages
countPerPage = [10|20|50]
```

## 显示格式选项

```
#####
# Display Formatting Options
#####Display Formatting Options

# General options
display.general.enablePreview = 0 | 1

# Event options
# TODO: uncomment the fields when we are ready to merge the values
display.events.fields = <string>
display.events.type = [raw|list|table]
display.events.rowNumbers = 0 | 1
display.events.maxLines = [0|5|10|20|50|100|200]
display.events.raw.drilldown = [inner|outer|full|none]
display.events.list.drilldown = [inner|outer|full|none]
display.events.list.wrap = 0 | 1
display.events.table.drilldown = 0 | 1
display.events.table.wrap = 0 | 1

# Statistics options
display.statistics.rowNumbers = 0 | 1
display.statistics.wrap = 0 | 1
display.statistics.drilldown = [row|cell|none]

# Visualization options
display.visualizations.type = [charting|singlevalue]
display.visualizations.custom.type = <string>
```

```

display.visualizations.chartHeight = <int>
display.visualizations.charting.chart = [line|area|column|bar|pie|scatter|radialGauge|fillerGauge|markerGauge]
display.visualizations.charting.chart.style = [minimal|shiny]
display.visualizations.charting.legend.labelStyle.overflowMode = [ellipsisEnd|ellipsisMiddle|ellipsisStart]

# Patterns options
display.page.search.patterns.sensitivity = <float>

# Page options
display.page.search.mode = [fast|smart|verbose]
display.page.search.timeline.format = [hidden|compact|full]
display.page.search.timeline.scale = [linear|log]
display.page.search.showFields = 0 | 1
display.page.home.showGettingStarted = 0 | 1
display.page.search.searchHistoryTimeFilter = [0|@d|-7d@d|-30d@d]

```

## ui-prefs.conf.example

```

# Version 6.5.0
#
# This file contains example of ui preferences for a view.
#
# To use one or more of these configurations, copy the configuration block into
# ui-prefs.conf in $SPLUNK_HOME/etc/system/local/. You must restart Splunk to
# enable configurations.
#
# To learn more about configuration files (including precedence) please see the
# documentation located at
# http://docs.splunk.com/Documentation/Splunk/latest/Admin/Aboutconfigurationfiles
#
# The following ui preferences will default timerange picker on the search page
# from All time to Today We will store this ui-prefs.conf in
# $SPLUNK_HOME/etc/apps/search/local/ to only update search view of search app.
[search]
dispatch.earliest_time = @d
dispatch.latest_time = now

```

## ui-tour.conf

以下为 ui-tour.conf 的规范和示例文件。

### ui-tour.conf.spec

```

# Version 6.5.0
#
# This file contains the tours available for Splunk Onboarding
#
# There is a default ui-tour.conf in $SPLUNK_HOME/etc/system/default.
# To create custom tours, place a ui-tour.conf in
# $SPLUNK_HOME/etc/system/local/. To create custom tours for an app, place
# ui-tour.conf in $SPLUNK_HOME/etc/apps/<app_name>/local/.
#
# To learn more about configuration files (including precedence) see the
# documentation located at
# http://docs.splunk.com/Documentation/Splunk/latest/Admin/Aboutconfigurationfiles
#

```

### 全局设置

```

# GLOBAL SETTINGS
# Use the [default] stanza to define any global settings.
# * You can also define global settings outside of any stanza, at the top of
# the file.
# * This is not a typical conf file for configurations. It is used to set/create
# tours to demonstrate product functionality to users.
# * If an attribute is defined at both the global level and in a specific

```

```
#      stanza, the value in the specific stanza takes precedence.

[<stanza name>]
* Stanza name is the name of the tour

useTour = <string>
* Used to redirect this tour to another when called by Splunk.
* Optional

nextTour = <string>
* String used to determine what tour to start when current tour is finished.
* Optional

intro = <string>
* A custom string used in a modal to describe what tour is about to be taken.
* Optional

type = <image || interactive>
* Can either be "image" or "interactive" to determine what kind of tour it is.
* Required

label = <string>
* The identifying name for this tour used in the tour creation app.
* Optional

tourPage = <string>
* The Splunk view this tour is associated with (only necessary if it is linked to).
* Optional

viewed = <boolean>
* A boolean to determine if this tour has been viewed by a user.
* Set by Splunk
```

## 针对基于图像的浏览

```
#####
## For image based tours
#####For image based tours
# Users can list as many images with captions as they want. Each new image is created by
# incrementing the number.

imageName<int> = <string>
* The name of the image file (example.png)
* Required but Optional only after first is set

imageCaption<int> = <string>
* The caption string for corresponding image
* Optional

imgPath = <string>
* The subdirectory relative to Splunk's 'img' directory in which users put the images.
  This will be appended to the url for image access and not make a server request within Splunk.
  EX) If user puts images in a subdirectory 'foo': imgPath = foo.
  EX) If within an app, imgPath = foo will point to the app's img path of
      appserver/static/img/foo
* Required only if images are not in the main 'img' directory.

context = <system || <specific app name>>
* String consisting of either 'system' or the app name the tour images are to be stored.
* If set to 'system', it will revert to Splunk's native img path.
* Required
```

## 针对交互式浏览

```
#####
## For interactive tours
#####For interactive tours
```

```

# Users can list as many steps with captions as they want. Each new step is created by
# incrementing the number.

urlData = <string>
* String of any querystring variables used with tourPage to create full url executing this tour.
* Optional

stepText<int> = <string>
* The string used in specified step to describe the UI being showcased.
* Required but Optional only after first is set

stepElement<int> = <selector>
* The UI Selector used for highlighting the DOM element for corresponding step.
* Optional

stepPosition<int> = <bottom || right || left || top>
* String that sets the position of the tooltip for corresponding step.
* Optional

stepClickEvent<int> = <click || mousedown || mouseup>
* Sets a specific click event for an element for corresponding step.
* Optional

stepClickElement<int> = <string>
* The UI selector used for a DOM element used in conjunction with click above.
* Optional

```

## ui-tour.conf.example

```

# Version 6.5.0
#
# This file contains the tours available for Splunk Onboarding
#
# To update tours, copy the configuration block into
# ui-tour.conf in $SPLUNK_HOME/etc/system/local/. Restart the Splunk software to
# see the changes.
#
# To learn more about configuration files (including precedence) see the
# documentation located at
# http://docs.splunk.com/Documentation/Splunk/latest/Admin/Aboutconfigurationfiles
#

# Image Tour
[tour-name]
type = image
imageName1 = TourStep1.png
imageCaption1 = This is the first caption
imageName2 = TourStep2.png
imageCaption2 = This is the second caption
imgPath = /testtour
context = system

# Interactive Tour
type = interactive
tourPage = reports
urlData =
label = Interactive Tour Test
stepText1 = Welcome to this test tour
stepText2 = This is the first step in the tour
stepElement2 = .test-selector
stepText3 = This is the second step in the tour
stepElement3 = .test-selector
stepClickEvent3 = mousedown
stepClickElement3 = .test-click-element
viewed = 0

```

## user-prefs.conf

以下为 user-prefs.conf 的规范和示例文件。

## user-prefs.conf.spec

```
# This file describes some of the settings that are used, and
# can be configured on a per-user basis for use by the Splunk Web UI.

# Settings in this file are requested with user and application scope of the
# relevant user, and the user-prefs app.

# Additionally, settings by the same name which are available in the roles
# the user belongs to will be used at lower precedence.

# This means interactive setting of these values will cause the values to be
# updated in
# $SPLUNK_HOME/etc/users/<username>/user-prefs/local/user-prefs.conf where
# <username> is the username for the user altering their preferences.

# It also means that values in another app will never be used unless they
# are exported globally (to system scope) or to the user-prefs app.

# In practice, providing values in other apps isn't very interesting, since
# values from the authorize.conf roles settings are more typically sensible
# ways to defaults for values in user-prefs.
```

### [general]

```
[general]
default_namespace = <app name>
* Specifies the app that the user will see initially upon login to the
  Splunk Web User Interface.
* This uses the "short name" of the app, such as launcher, or search,
  which is synonymous with the app directory name.
* Splunk defaults this to 'launcher' via the default authorize.conf

tz = <timezone>
* Specifies the per-user timezone to use
* If unset, the timezone of the Splunk Server or Search Head is used.
* Only canonical timezone names such as America/Los_Angeles should be
  used (for best results use the Splunk UI).
* Defaults to unset.

lang = <language>
* Specifies the per-user language preference for non-webui operations, where
  multiple tags are separated by commas.
* If unset, English "en-US" will be used when required.
* Only tags used in the "Accept-Language" HTTP header will be allowed, such as
  "en-US" or "fr-FR".
* Fuzzy matching is supported, where "en" will match "en-US".
* Optional quality settings is supported, such as "en-US,en;q=0.8,fr;q=0.6"
* Defaults to unset.

install_source_checksum = <string>
* Records a checksum of the tarball from which a given set of private user
  configurations was installed.
* Analogous to <install_source_checksum> in app.conf.

search_syntax_highlighting = <boolean>
* Highlights different parts of a search string with different colors.
* Defaults to true.

search_assistant = [full|compact|none]
* Specifies the type of search assistant to use when constructing a search.
* Defaults to compact.

infodelivery_enabled = <boolean>
* Enables the info delivery app
* Defaults to true

infodelivery_show_ad_modal = <boolean>
```



```

* Flag to disable/enable the ad modal for info delivery app
* Defaults to true

infodelivery_show_configure_modal = <boolean>
* Flag to disable/enable the configure modal for info delivery
* Defaults to true

datasets:showInstallDialog = <boolean>
* Flag to enable/disable the install dialog for the datasets addon
* Defaults to true

```

### **[default]**

```

[default]
# Additional settings exist, but are entirely UI managed.
<setting> = <value>

```

### **[general\_default]**

```

[general_default]
default_earliest_time = <string>
default_latest_time = <string>
* Sets the global default time range across all apps, users, and roles on the search page.

```

## **user-prefs.conf.example**

```

# Version 6.5.0
#
# This is an example user-prefs.conf. Use this file to configure settings
# on a per-user basis for use by the Splunk Web UI.
#
# To use one or more of these configurations, copy the configuration block
# into user-prefs.conf in $SPLUNK_HOME/etc/system/local/. You must restart
# Splunk to enable configurations.
#
# To learn more about configuration files (including precedence) please see
# the documentation located at
# http://docs.splunk.com/Documentation/Splunk/latest/Admin/Aboutconfigurationfiles
#
# Note: These are examples. Replace the values with your own
# customizations.

# EXAMPLE: Setting the default timezone to GMT for all Power and User role
# members, and setting a different language preference for each.

[role_power]
tz = GMT
lang = en-US

[role_user]
tz = GMT
lang = fr-FR,fr-CA;q=0

```

## **user-seed.conf**

以下为 user-seed.conf 的规范和示例文件。

### **user-seed.conf.spec**

```

# Version 6.5.0
#
# Specification for user-seed.conf. Allows configuration of Splunk's
# initial username and password. Currently, only one user can be configured

```

```
# with user-seed.conf.
#
# Specification for user-seed.conf. Allows configuration of Splunk's initial username and password.
# Currently, only one user can be configured with user-seed.conf.
#
# To override the default username and password, place user-seed.conf in
# $SPLUNK_HOME/etc/system/local. You must restart Splunk to enable configurations.
#
# http://docs.splunk.com/Documentation/Splunk/latest/Admin/Aboutconfigurationfiles
# To learn more about configuration files (including precedence) please see the documentation
# located at http://docs.splunk.com/Documentation/Splunk/latest/Admin/Aboutconfigurationfiles
```

## **[user\_info]**

```
[user_info]
* Default is Admin.

USERNAME = <string>
    * Username you want to associate with a password.
    * Default is Admin.

PASSWORD = <password>
    * Password you wish to set for that user.
    * Default is changeme.
```

## **user-seed.conf.example**

```
# Version 6.5.0
#
# This is an example user-seed.conf. Use this file to create an initial
# This is an example user-seed.conf. Use this file to create an initial login.
#
# NOTE: To change the default start up login and password, this file must be
# NOTE: To change the default start up login and password, this file must be in
# $SPLUNK_HOME/etc/system/default/ prior to starting Splunk for the first time.
#
# To use this configuration, copy the configuration block into
# To use this configuration, copy the configuration block into user-seed.conf
# in $SPLUNK_HOME/etc/system/local/. You must restart Splunk to enable configurations.
#
# To learn more about configuration files (including precedence) please see
# To learn more about configuration files (including precedence) please see the documentation
# located at http://docs.splunk.com/Documentation/Splunk/latest/Admin/Aboutconfigurationfiles

[user_info]
USERNAME = admin
PASSWORD = myowndefaultPass
```

## **viewstates.conf**

以下为 viewstates.conf 的规范和示例文件。

### **viewstates.conf.spec**

```
# Version 6.5.0
#
# This file explains how to format viewstates.
#
# To use this configuration, copy the configuration block into
# viewstates.conf in $SPLUNK_HOME/etc/system/local/. You must restart Splunk
# to enable configurations.
#
# To learn more about configuration files (including precedence) please see
# the documentation located at
# http://docs.splunk.com/Documentation/Splunk/latest/Admin/Aboutconfigurationfiles
```

## 全局设置

```
# GLOBAL SETTINGS
# Use the [default] stanza to define any global settings.
# * You can also define global settings outside of any stanza, at the top
#   of the file.
# * Each conf file should have at most one default stanza. If there are
#   multiple default stanzas, attributes are combined. In the case of
#   multiple definitions of the same attribute, the last definition in the
#   file wins.
# * If an attribute is defined at both the global level and in a specific
#   stanza, the value in the specific stanza takes precedence.
```

### [<view\_name>:<viewstate\_id>]

```
[<view_name>:<viewstate_id>]
* Auto-generated persistence stanza label that corresponds to UI views
* The <view_name> is the URI name (not label) of the view to persist
* if <view_name> = "*", then this viewstate is considered to be 'global'
* The <viewstate_id> is the unique identifier assigned to this set of
  parameters
* <viewstate_id> = '_current' is a reserved name for normal view
  'sticky state'
* <viewstate_id> = '_empty' is a reserved name for no persistence,
  i.e., all defaults

<module_id>.<setting_name> = <string>
* The <module_id> is the runtime id of the UI module requesting persistence
* The <setting_name> is the setting designated by <module_id> to persist
```

## viewstates.conf.example

```
# Version 6.5.0
#
# This is an example viewstates.conf.
#
# To learn more about configuration files (including precedence) please see
# the documentation located at
# http://docs.splunk.com/Documentation/Splunk/latest/Admin/Aboutconfigurationfiles

[charting:g3b5fa71]
ChartTypeFormatter_0_7_0.default = area
Count_0_6_0.count = 10
LegendFormatter_0_13_0.default = right
LineMarkerFormatter_0_10_0.default = false
NullValueFormatter_0_12_0.default = gaps

[*:g3jck9ey]
Count_0_7_1.count = 20
DataOverlay_0_12_0.dataOverlayMode = none
DataOverlay_1_13_0.dataOverlayMode = none
FieldPicker_0_6_1.fields = host sourcetype source date_hour date_mday date_minute date_month
FieldPicker_0_6_1.sidebarDisplay = True
FlashTimeline_0_5_0.annotationSearch = search index=twink
FlashTimeline_0_5_0.enableAnnotations = true
FlashTimeline_0_5_0.minimized = false
MaxLines_0_13_0.maxLines = 10
RowNumbers_0_12_0.displayRowNumbers = true
RowNumbers_1_11_0.displayRowNumbers = true
RowNumbers_2_12_0.displayRowNumbers = true
Segmentation_0_14_0.segmentation = full
SoftWrap_0_11_0.enable = true

[dashboard:_current]
TimeRangePicker_0_1_0.selected = All time
```

# visualizations.conf

以下为 visualizations.conf 的规范和示例文件。

## visualizations.conf.spec

```
# Version 6.5.0
#
# This file contains definitions for visualizations an app makes avialable
# to the system. An app intending to share visualizations with the system
# should include a visualizations.conf in $SPLUNK_HOME/etc/apps/<appname>/default
#
# visualizations.conf should include one stanza for each visualization to be shared
#
# To learn more about configuration files (including precedence) please see
# the documentation located at
# http://docs.splunk.com/Documentation/Splunk/latest/Admin/Aboutconfigurationfiles

#*****
# The possible attribute/value pairs for visualizations.conf are:
#*****
```

### [<stanza name>]

```
[<stanza name>]
* Create a unique stanza name for each visualization. It should match the name
  of the visualization
* Follow the stanza name with any number of the following attribute/value
  pairs.
* If you do not specify an attribute, Splunk uses the default.

allow_user_selection = <bool>
* Optional.
* Whether the visualization should be available for users to select
* Defaults to true

default_height = <int>
* Optional.
* The default height of the visualization in pixels
* Defaults to 250

description = <string>
* Required.
* The short description that will show up in the visualization picker
* Defaults to ""

disabled = <bool>
* Optional.
* Disable the visualization by setting to true.
* If set to true, the visualization is not available anywhere in Splunk
* Defaults to false.

label = <string>
* Required.
* The human-readable label or title of the visualization
* Will be used in dropdowns and lists as the name of the visualization
* Defaults to <app_name>.<viz_name>

search_fragment = <string>
* Required.
* An example part of a search that formats the data correctly for the viz. Typically the last pipe(s) in a search
  query.
* Defaults to ""
```

## visualizations.conf.example

No example

## web.conf

以下为 web.conf 的规范和示例文件。

### web.conf.spec

```
# Version 6.5.0
#
# This file contains possible attributes and values you can use to configure
# Splunk's web interface.
#
# There is a web.conf in $SPLUNK_HOME/etc/system/default/. To set custom
# configurations, place a web.conf in $SPLUNK_HOME/etc/system/local/. For
# examples, see web.conf.example. You must restart Splunk to enable
# configurations.
#
# To learn more about configuration files (including precedence) please see
# the documentation located at
# http://docs.splunk.com/Documentation/Splunk/latest/Admin/Aboutconfigurationfiles

[settings]
* Set general SplunkWeb configuration options under this stanza name.
* Follow this stanza name with any number of the following attribute/value
  pairs.
* If you do not specify an entry for each attribute, Splunk will use the
  default value.

startwebserver = [0 | 1]
* Set whether or not to start SplunkWeb.
* 0 disables SplunkWeb, 1 enables it.
* Defaults to 1.

httpport = <port_number>
* Must be present for SplunkWeb to start.
* If omitted or 0 the server will NOT start an http listener.
* If using SSL, set to the HTTPS port number.
* Defaults to 8000.

mgmtHostPort = <IP:port>
* Location of splunkd.
* Don't include http[s]:// -- just the IP address.
* Defaults to 127.0.0.1:8089.

appServerPorts = <one or more port numbers>
* Port number(s) for the python-based application server to listen on.
  This port is bound only on the loopback interface -- it is not
  exposed to the network at large.
* If set to "0", this will prevent the application server from
  being run from splunkd. Instead, splunkweb will be started as
  a separate python-based service which directly listens to the
  'httpport'. This is how Splunk 6.1.X and earlier behaved.
* Generally, you should only set one port number here. For most
  deployments a single application server won't be a performance
  bottleneck. However you can provide a comma-separated list of
  port numbers here and splunkd will start a load-balanced
  application server on each one.
* It is recommended that this be set to a non-zero value. Setting
  this to "0" should only be done if you experience a compatibility
  problem. The new separate application server configuration is faster
  and supports more configuration options. Also, setting this to
  "0" may cause problems with new functionality, such as using the
  Search Head Clustering feature.
  (see the [shclustering] stanza in server.conf)
* Defaults to 8065.

splunkdConnectionTimeout = <integer>
```

- \* Number of seconds to wait before timing out when communicating with splunkd
- \* Must be at least 30
- \* Values smaller than 30 will be ignored, resulting in the use of the default value
- \* Defaults to 30

enableSplunkWebSSL = [True | False]

- \* Toggle between http or https.
- \* Set to true to enable https and SSL.
- \* Defaults to False.

privKeyPath = <path>

- \* The path to the file containing the web server's SSL certificate's private key.
- \* A relative path is interpreted relative to \$SPLUNK\_HOME and may not refer outside of \$SPLUNK\_HOME (e.g., no ../somewhere).
- \* An absolute path can also be specified to an external key.
- \* See also 'enableSplunkWebSSL' and 'serverCert'.

serverCert = <path>

- \* Full path to the PEM format Splunk web server certificate file.
- \* The file may also contain root and intermediate certificates, if required. They should be listed sequentially in the order:
  - [ Server's SSL certificate ]
  - [ One or more intermediate certificates, if required ]
  - [ Root certificate, if required ]
- \* Default is \$SPLUNK\_HOME/etc/auth/splunkweb/cert.pem.
- \* See also 'enableSplunkWebSSL' and 'privKeyPath'.

caCertPath = <path>

- \* DEPRECATED; use 'serverCert' instead.
- \* A relative path is interpreted relative to \$SPLUNK\_HOME and may not refer outside of \$SPLUNK\_HOME (e.g., no ../somewhere).

requireClientCert = [True | False]

- \* Requires that any HTTPS client that connects to the splunkweb HTTPS server has a certificate that was signed by the CA cert installed on this server.
- \* If true, a client can connect ONLY if a certificate created by our certificate authority was used on that client.
- \* When true, it is mandatory to configure splunkd with same root CA in server.conf. This is needed for internal communication between splunkd and splunkweb.
- \* Defaults to false.

sslCommonNameToCheck = <commonName1>, <commonName2>, ...

- \* Optional. Defaults to no common name checking.
- \* Check the common name of the client's certificate against this list of names.
- \* requireClientCert must be set to true for this setting to work.

sslAltNameToCheck = <alternateName1>, <alternateName2>, ...

- \* If this value is set, and 'requireClientCert' is set to true, splunkweb will verify certificates which have a so-called "Subject Alternate Name" that matches any of the alternate names in this list.
- \* Subject Alternate Names are effectively extended descriptive fields in SSL certs beyond the commonName. A common practice for HTTPS certs is to use these values to store additional valid hostnames or domains where the cert should be considered valid.
- \* Accepts a comma-separated list of Subject Alternate Names to consider valid.
- \* Optional. Defaults to no alternate name checking

serviceFormPostURL = http://docs.splunk.com/Documentation/Splunk

- \* This attribute is deprecated since 5.0.3

userRegistrationURL = https://www.splunk.com/page/sign\_up

updateCheckerBaseURL = http://quickdraw.Splunk.com/js/

docsCheckerBaseURL = http://quickdraw.splunk.com/help

- \* These are various Splunk.com urls that are configurable.
- \* Setting updateCheckerBaseURL to 0 will stop the SplunkWeb from pinging Splunk.com for new versions of itself.

enable\_insecure\_login = [True | False]

- \* Indicates if the GET-based /account/insecurelogin endpoint is enabled
- \* Provides an alternate GET-based authentication mechanism
- \* If True, the /account/insecurelogin?username=USERNAME&password=PASSWD is available
- \* If False, only the main /account/login endpoint is available
- \* Defaults to False

simple\_error\_page = [True | False]

- \* If set to True a simplified error page will be displayed for errors (404, 500, etc.) only containing the status
- \* If set to False a more verbose error page will be displayed containing homelink, message, more\_results\_link, crashes, referrer, debug output and byline
- \* Defaults to False

login\_content = <content\_string>

- \* Add custom content to the login page
- \* Supports any text including html

sslVersions = <list of ssl versions string>

- \* Comma-separated list of SSL versions to support
- \* The versions available are "ssl3", "tls1.0", "tls1.1", and "tls1.2"
- \* The special version "\*" selects all supported versions. The version "tls" selects all versions tls1.0 or newer
- \* If a version is prefixed with "-" it is removed from the list
- \* SSLv2 is always disabled; "-ssl2" is accepted in the version list but does nothing
- \* When appServerPorts=0 only supported values are "all", "ssl3, tls" and "tls"
- \* When configured in FIPS mode ssl3 is always disabled regardless of this configuration
- \* Defaults to "ssl3, tls". (anything newer than SSLv2)
- \* NOTE: this setting only takes effect when appServerPorts is set to a non-zero value

supportSSLV3Only = <bool>

- \* When appServerPorts is set to a non-zero value (the default mode), this setting is DEPRECATED. SSLv2 is now always disabled. The exact set of SSL versions allowed is now configurable via the "sslVersions" setting above.
- \* When appServerPorts is set to zero, this controls whether we disallow SSLv2 connections.

cipherSuite = <cipher suite string>

- \* If set, uses the specified cipher string for the HTTP server.
- \* If not set, uses the default cipher string provided by OpenSSL. This is used to ensure that the server does not accept connections using weak encryption protocols.
- \* Must specify 'dhFile' to enable any Diffie-Hellman ciphers.

ecdhCurves = <comma separated list of ec curves>

- \* ECDH curves to use for ECDH key negotiation.
- \* The curves should be specified in the order of preference.
- \* The client sends these curves as a part of Client Hello.
- \* The server supports only the curves specified in the list.
- \* We only support named curves specified by their SHORT names. (see struct ASN1\_OBJECT in asn1.h)
- \* The list of valid named curves by their short/long names can be obtained by executing this command:  
\$SPLUNK\_HOME/bin/splunk cmd openssl ecparam -list\_curves
- \* Default is empty string.
- \* e.g. ecdhCurves = prime256v1,secp384r1,secp521r1

ecdhCurveName = <string>

- \* DEPRECATED; use 'ecdhCurves' instead.
- \* ECDH curve to use for ECDH key negotiation
- \* We only support named curves specified by their SHORT name. (see struct ASN1\_OBJECT in asn1.h)
- \* The list of valid named curves by their short/long names can be obtained by executing this command:  
\$SPLUNK\_HOME/bin/splunk cmd openssl ecparam -list\_curves
- \* Default is empty string.

dhFile = <path>

- \* The path to the Diffie-Hellman parameter file.

- \* Relative paths are interpreted as relative to \$SPLUNK\_HOME.
- \* Relative paths may not refer outside of \$SPLUNK\_HOME (eg. no ../somewhere).
- \* An absolute path can also be specified to an external key.
- \* Not set by default.

root\_endpoint = <URI\_prefix\_string>

- \* Defines the root URI path on which the appserver will listen
- \* Default setting is '/'
- \* Ex: if you want to proxy the splunk UI at http://splunk:8000/splunkui, then set root\_endpoint = /splunkui

static\_endpoint = <URI\_prefix\_string>

- \* Path to static content
- \* The path here is automatically appended to root\_endpoint defined above
- \* Default is /static

static\_dir = <relative\_filesystem\_path>

- \* The directory that actually holds the static content
- \* This can be an absolute url if you want to put it elsewhere
- \* Default is share/splunk/search\_mrsparkle/exposed

rss\_endpoint = <URI\_prefix\_string>

- \* Path to static rss content
- \* The path here is automatically appended to root\_endpoint defined above
- \* Default is /rss

embed\_uri = <URI>

- \* Optional URI scheme/host/port prefix for embedded content
- \* This presents an optional strategy for exposing embedded shared content that does not require authentication in reverse proxy/SSO environment.
- \* Default is empty and will resolve to the client window.location.protocol + "://" + window.location.host

embed\_footer = <html\_string>

- \* chunk of html to define the footer for an embedded report.
- \* Defaults to "splunk" but can be changed to whatever the user would like.

tools.staticdir.generate\_indexes = [1 | 0]

- \* Indicates if the webserver will serve a directory listing for static directories
- \* Defaults to 0 (false)

template\_dir = <relative\_filesystem\_path>

- \* Base path to mako templates
- \* Defaults to share/splunk/search\_mrsparkle/templates

module\_dir = <relative\_filesystem\_path>

- \* Base path to UI module assets
- \* Defaults to share/splunk/search\_mrsparkle/modules

enable\_gzip = [True | False]

- \* Determines if webserver applies gzip compression to responses
- \* Defaults to True

use\_future\_expires = [True | False]

- \* Determines if the Expires header of /static files is set to a far-future date
- \* Defaults to True

flash\_major\_version = <integer>

flash\_minor\_version = <integer>

flash\_revision\_version = <integer>

- \* Specifies the minimum Flash plugin version requirements
- \* Flash support, broken into three parts.
- \* We currently require a min baseline of Shockwave Flash 9.0 r124

override\_JSON\_MIME\_type\_with\_text\_plain = [True | False]

- \* Specifies whether or not to override the MIME type for JSON data served up by splunkweb endpoints with content-type="text/plain; charset=UTF-8"
- \* If True, splunkweb endpoints (other than proxy) that serve JSON data will serve as "text/plain; charset=UTF-8"
- \* If False, splunkweb endpoints that serve JSON data will serve as "application/json; charset=UTF-8"



```

enable_proxy_write = [True | False]
* Indicates if the /splunkd proxy endpoint allows POST operations
* If True, both GET and POST operations are proxied through to splunkd
* If False, only GET operations are proxied through to splunkd
* Setting this to False will prevent many client-side packages (such as the
  Splunk JavaScript SDK) from working correctly
* Defaults to True

js_logger_mode = [None | Firebug | Server]
* JavaScript Logger mode
* Available modes: None, Firebug, Server
* Mode None: Does not log anything
* Mode Firebug: Use firebug by default if it exists or defer to the older
  less promiscuous version of firebug lite
* Mode Server: Log to a defined server endpoint
* See js/logger.js Splunk.Logger.Mode for mode implementation details and if
  you would like to author your own
* Defaults to None

js_logger_mode_server_end_point = <URI_relative_path>
* Specifies the server endpoint to post javascript log messages
* Used when js_logger_mode = Server
* Defaults to util/log/js

js_logger_mode_server_poll_buffer = <integer>
* Specifies the interval in milliseconds to check, post and cleanse the javascript log buffer
* Defaults to 1000

js_logger_mode_server_max_buffer = <integer>
* Specifies the maximum size threshold to post and cleanse the javascript log buffer
* Defaults to 100

ui_inactivity_timeout = <integer>
* Specifies the length of time lapsed (in minutes) for notification when
  there is no user interface clicking, mouseover, scrolling or resizing.
* Notifies client side pollers to stop, resulting in sessions expiring at
  the tools.sessions.timeout value.
* If less than 1, results in no timeout notification ever being triggered
  (Sessions will stay alive for as long as the browser is open).
* Defaults to 60 minutes

js_no_cache = [True | False]
* Toggle js cache control
* Defaults to False

cacheBytesLimit = <integer>
* When appServerPorts is set to a non-zero value, splunkd can keep a
  small cache of static assets in memory. When the total size of the
  objects in cache grows larger than this, we begin the process of ageing
  entries out.
* Defaults to 4194304 (i.e. 4 Megabytes)
* If set to zero, this cache is completely disabled

cacheEntriesLimit = <integer>
* When appServerPorts is set to a non-zero value, splunkd can keep a
  small cache of static assets in memory. When the number of the objects
  in cache grows larger than this, we begin the process of ageing
  entries out.
* Defaults to 16384
* If set to zero, this cache is completely disabled

staticCompressionLevel = <integer>
* When appServerPorts is set to a non-zero value, splunkd can keep a
  small cache of static assets in memory. These are stored
  compressed and can usually be served directly to the web browser
  in compressed format.
* This level can be a number between 1 and 9. Lower numbers use less
  CPU time to compress objects, but the resulting compressed objects
  will be larger.
* Defaults to 9. Usually not much CPU time is spent compressing these
  objects so there is not much benefit to decreasing this.

```

```

enable_autocomplete_login = [True | False]
* Indicates if the main login page allows browsers to autocomplete the username
* If True, browsers may display an autocomplete drop down in the username field
* If False, browsers are instructed not to show autocomplete drop down in the username field
* Defaults to False

verify_cookies_work_during_login = [True | False]
* Normally, the login page will make an attempt to see if cookies work
  properly in the user's browser before allowing them to log in. If
  this is set to False, this check is skipped.
* Defaults to True. It is recommended that this be left on.
* NOTE: this setting only takes effect when appServerPorts is set to a
      non-zero value

minify_js = [True | False]
* Indicates whether the static JS files for modules are consolidated and minified
* Enabling improves client-side performance by reducing the number of HTTP
  requests and the size of HTTP responses

minify_css = [True | False]
* Indicates whether the static CSS files for modules are consolidated and
  minified
* Enabling improves client-side performance by reducing the number of HTTP
  requests and the size of HTTP responses
* Due to browser limitations, disabling this when using IE9 and earlier may
  result in display problems.

trap_module_exceptions = [True | False]
* Toggle whether the JS for individual modules is wrapped in a try/catch
* If True, syntax errors in individual modules will not cause the UI to
  hang, other than when using the module in question
* Set this to False when developing apps.

enable_pivot_adhoc_acceleration = [True | False]
* Toggle whether the pivot interface will use its own ad-hoc acceleration
  when a data model is not accelerated.
* If True, this ad-hoc acceleration will be used to make reporting in pivot
  faster and more responsive.
* In situations where data is not stored in time order or where the majority
  of events are far in the past, disabling this behavior can improve the
  pivot experience.
* DEPRECATED in version 6.1 and later, use pivot_adhoc_acceleration_mode
  instead

pivot_adhoc_acceleration_mode = [Elastic | AllTime | None]
* Specify the type of ad-hoc acceleration used by the pivot interface when a
  data model is not accelerated.
* If Elastic, the pivot interface will only accelerate the time range
  specified for reporting, and will dynamically adjust when this time range
  is changed.
* If AllTime, the pivot interface will accelerate the relevant data over all
  time. This will make the interface more responsive to time-range changes
  but places a larger load on system resources.
* If None, the pivot interface will not use any acceleration. This means
  any change to the report will require restarting the search.
* Defaults to Elastic

jschart_test_mode = [True | False]
* Toggle whether JSChart module runs in Test Mode
* If True, JSChart module attaches HTML classes to chart elements for
  introspection
* This will negatively impact performance, so should be disabled unless
  actively in use.

#
# JSChart data truncation configuration
# To avoid negatively impacting browser performance, the JSChart library
# places a limit on the number of points that will be plotted by an
# individual chart. This limit can be configured here either across all
# browsers or specifically per-browser. An empty or zero value will disable
# the limit entirely.

```

```

#

jschart_truncation_limit = <int>
* Cross-browser truncation limit, if defined takes precedence over the
  browser-specific limits below

jschart_truncation_limit.chrome = <int>
* Chart truncation limit for Chrome only
* Defaults to 50000

jschart_truncation_limit.firefox = <int>
* Chart truncation limit for Firefox only
* Defaults to 50000

jschart_truncation_limit.safari = <int>
* Chart truncation limit for Safari only
* Defaults to 50000

jschart_truncation_limit.ie11 = <int>
* Chart truncation limit for Internet Explorer 11 only
* Defaults to 50000

dashboard_html_allow_inline_styles = <bool>
* If this setting is set to false styles attributes from inline HTML elements
  in dashboards will be removed to prevent potential attacks.
* Default is true

max_view_cache_size = <integer>
* Specifies the maximum number of views to cache in the appserver.
* Defaults to 300.

pdfgen_is_available = [0 | 1]
* Specifies whether Integrated PDF Generation is available on this search
  head
* This is used to bypass an extra call to splunkd
* Defaults to 1 on platforms where node is supported, defaults to 0
  otherwise

version_label_format = <printf_string>
* Internal config
* Used to override the version reported by the UI to *.splunk.com resources
* Defaults to: %s

auto_refresh_views = [0 | 1]
* Specifies whether the following actions cause the appserver to ask splunkd
  to reload views from disk.
  * Logging in via the UI
  * Switching apps
  * Clicking the Splunk logo
* Defaults to 0.

#
# Splunk bar options
#
# Internal config. May change without notice.
# Only takes effect if instanceType is 'cloud'.
#

showProductMenu = [True | False]
  * Used to indicate visibility of product menu.
  * Defaults to False.

productMenuUriPrefix = <string>
  * The domain product menu links to.
  * Required if showProductMenu is set to True.

productMenuLabel = <string>
  * Used to change the text label for product menu.
  * Defaults to 'My Splunk'.

showUserMenuProfile = [True | False]
  * Used to indicate visibility of 'Profile' link within user menu.

```

```

    * Defaults to False.

#
# Header options
#
x_frame_options_sameorigin = [True | False]
* adds a X-Frame-Options header set to "SAMEORIGIN" to every response served
* by cherrypy
* Defaults to True

#

```

## SSO

```

# SSO
#

remoteUser = <http_header_string>
* Remote user HTTP header sent by the authenticating proxy server.
* This header should be set to the authenticated user.
* Defaults to 'REMOTE_USER'.
* Caution: There is a potential security concern regarding Splunk's
  treatment of HTTP headers.
* Your proxy provides the selected username as an HTTP header as specified
  above.
* If the browser or other http agent were to specify the value of this
  header, probably any proxy would overwrite it, or in the case that the
  username cannot be determined, refuse to pass along the request or set
  it blank.
* However, Splunk (cherrypy) will normalize headers containing the dash,
  and the underscore to the same value. For example USER-NAME and
  USER_NAME will be treated as the same in SplunkWeb.
* This means that if the browser provides REMOTE-USER and splunk accepts
  REMOTE_USER, theoretically the browser could dictate the username.
* In practice, however, in all our testing, the proxy adds its headers
  last, which causes them to take precedence, making the problem moot.
* See also the 'remoteUserMatchExact' setting which can enforce more exact
  header matching when running with appServerPorts enabled.

remoteGroups = <http_header_string>
* Remote groups HTTP header name sent by the authenticating proxy server.
* This value is used by splunk the match against header name.
* The header value format should be set to comma separated groups that user belongs to.
* Example of header value: Products,Engineering,Quality Assurance
* There is no default value set for this parameter.

remoteGroupsQuoted = [true | false]
* This attribute is considered only when 'remoteGroups' is set.
* When value is set to true, the group header value can be comma separated
  quoted entries. Note: These entries can contain comma.
* Example of header value with quoted entries:
  * "Products","North America, Engineering","Quality Assurance"
* By default value is set to false, in which case group entries should be without quotes.

remoteUserMatchExact = [0 | 1]
* IMPORTANT: this setting only takes effect when appServerPorts is set to a
  non-zero value
* When matching the remoteUser header with value "1", consider dashes and
  underscores distinct (so "Remote-User" and "Remote_User" will be
  considered different headers)
* Value "0" is for compatibility with older versions of Splunk, but
  setting to "1" is a good idea when setting up SSO with appServerPorts
  enabled
* Defaults to "0"

remoteGroupsMatchExact = [0 | 1]
* IMPORTANT: this setting only takes effect when appServerPorts is set to a
  non-zero value
* When matching the remoteGroups header with value "1", consider dashes and
  underscores distinct (so "Remote-Groups" and "Remote_Groups" will be

```

```

    considered different headers)
* Value "0" is for compatibility with older versions of Splunk, but
  setting to "1" is a good idea when setting up SSO with appServerPorts
  enabled
* Defaults to "0"

SSOMode = [permissive | strict]
* Allows SSO to behave in either permissive or strict mode.
* Permissive: Requests to Splunk Web that originate from an untrusted IP
  address are redirected to a login page where they can log into Splunk
  without using SSO.
* Strict: All requests to splunkweb will be restricted to those originating
  from a trusted IP except those to endpoints not requiring authentication.
* Defaults to "strict"

trustedIP = <ip_address>
* Trusted IP. This is the IP address of the authenticating proxy.
* Splunkweb verifies it is receiving data from the proxy host for all
  SSO requests.
* Uncomment and set to a valid IP address to enable SSO.
* Disabled by default. Normal value is '127.0.0.1'
* If appServerPorts is set to a non-zero value, this setting can accept a
  richer set of configurations, using the same format as the "acceptFrom"
  setting.

allowSsoWithoutChangingServerConf = [0 | 1]
* IMPORTANT: this setting only takes effect when appServerPorts is set to a
  non-zero value
* Usually when configuring SSO, a trustedIP needs to be set both here
  in web.conf and also in server.conf. If this is set to "1" then we will
  enable web-based SSO without a trustedIP in server.conf
* Defaults to 0

testing_endpoint = <relative_uri_path>
* Specifies the root URI path on which to serve splunkweb unit and
  integration testing resources.
* Development only setting
* Defaults to '/testing'

testing_dir = <relative_file_path>
* Specifies the path relative to $SPLUNK_HOME that contains the testing
  files to be served at endpoint defined by 'testing_endpoint'.
* Development only setting
* Defaults to 'share/splunk/testing'

ssoAuthFailureRedirect = <scheme>://<URL>
  * Used to set the redirect URL if SSO authentication fails.
    Examples:
    http://www.example.com
    https://www.example.com
  * Defaults to an empty value and will show the default unauthorized error
    page if SSO authentication fails.

# Results export server config

export_timeout = <integer>
* When exporting results, the number of seconds the server waits before
* closing the connection with splunkd. If you do not set a value for
* export_timeout, the value in splunkdConnectionTimeout is used.
* We recommend that you set export_timeout to a value greater than 30

#
# cherryypy HTTP server config
#

server.thread_pool = <integer>
* Specifies the minimum number of threads the appserver is allowed to
  maintain
* Defaults to 20

server.thread_pool_max = <integer>
* Specifies the maximum number of threads the appserver is allowed to

```

```

    maintain
* Defaults to -1 (unlimited)

server.thread_pool_min_spare = <integer>
* Specifies the minimum number of spare threads the appserver keeps idle
* Defaults to 5

server.thread_pool_max_spare = <integer>
* Specifies the maximum number of spare threads the appserver keeps idle
* Defaults to 10

server.socket_host = <ip_address>
* Host values may be any IPv4 or IPv6 address, or any valid hostname.
* The string 'localhost' is a synonym for '127.0.0.1' (or '::1', if your
  hosts file prefers IPv6). The string '0.0.0.0' is a special IPv4 entry
  meaning "any active interface" (INADDR_ANY), and ':::' is the similar
  IN6ADDR_ANY for IPv6.
* Defaults to 0.0.0.0 if listenOnIPv6 is set to no, else ::

server.socket_timeout = <integer>
* The timeout in seconds for accepted connections between the browser and
  splunkweb
* Defaults to 10

listenOnIPv6 = <no | yes | only>
* By default, splunkweb will listen for incoming connections using
  IPv4 only
* To enable IPv6 support in splunkweb, set this to "yes". Splunkweb
  will simultaneously listen for connections on both IPv4 and IPv6
* To disable IPv4 entirely, set this to "only", which will cause splunkweb
  to exclusively accept connections over IPv6.
* You will also want to set server.socket_host (use ":::"
  instead of "0.0.0.0") if you wish to listen on an IPv6 address

max_upload_size = <integer>
* Specifies the hard maximum size of uploaded files in MB
* Defaults to 500

log.access_file = <filename>
* Specifies the HTTP access log filename
* Stored in default Splunk /var/log directory
* Defaults to web_access.log

log.access_maxsize = <integer>
* Specifies the maximum size the web_access.log file should be allowed to
  grow to (in bytes)
* Comment out or set to 0 for unlimited file size
* File will be rotated to web_access.log.0 after max file size is reached
* See log.access_maxfiles to limit the number of backup files created
* Defaults to unlimited file size

log.access_maxfiles = <integer>
* Specifies the maximum number of backup files to keep after the
  web_access.log file has reached its maximum size
* Warning: setting this to very high numbers (eg. 10000) may impact
  performance during log rotations
* Defaults to 5 if access_maxsize is set

log.error_maxsize = <integer>
* Specifies the maximum size the web_service.log file should be allowed to
  grow to (in bytes)
* Comment out or set to 0 for unlimited file size
* File will be rotated to web_service.log.0 after max file size is reached
* See log.error_maxfiles to limit the number of backup files created
* Defaults to unlimited file size

log.error_maxfiles = <integer>
* Specifies the maximum number of backup files to keep after the
  web_service.log file has reached its maximum size
* Warning: setting this to very high numbers (eg. 10000) may impact
  performance during log rotations
* Defaults to 5 if access_maxsize is set

```

```

log.screen = [True | False]
* Indicates if runtime output is displayed inside an interactive tty
* Defaults to True

request.show_tracebacks = [True | False]
* Indicates if a an exception traceback is displayed to the user on fatal
  exceptions
* Defaults to True

engine.autoreload_on = [True | False]
* Indicates if the appserver will auto-restart if it detects a python file
  has changed
* Defaults to False

tools.sessions.on = True
* Indicates if user session support is enabled
* Should always be True

tools.sessions.timeout = <integer>
* Specifies the number of minutes of inactivity before a user session is
  expired
* The countdown is effectively reset by browser activity minute until
  ui_inactivity_timeout inactivity timeout is reached.
* Use a value of 2 or higher, as a value of 1 will race with the browser
  refresh, producing unpredictable behavior.
  (Low values aren't very useful though except for testing.)
* Defaults to 60

tools.sessions.restart_persist = [True | False]
* If set to False then the session cookie will be deleted from the browser
  when the browser quits
* Defaults to True - Sessions persist across browser restarts
  (assuming the tools.sessions.timeout limit hasn't been reached)

tools.sessions.httponly = [True | False]
* If set to True then the session cookie will be made unavailable
  to running javascript scripts, increasing session security
* Defaults to True

tools.sessions.secure = [True | False]
* If set to True and Splunkweb is configured to server requests using HTTPS
  (see the enableSplunkWebSSL setting) then the browser will only transmit
  the session cookie over HTTPS connections, increasing session security
* Defaults to True

response.timeout = <integer>
* Specifies the number of seconds to wait for the server to complete a
  response
* Some requests such as uploading large files can take a long time
* Defaults to 7200

tools.sessions.storage_type = [file]
tools.sessions.storage_path = <filepath>
* Specifies the session information storage mechanisms
* Comment out the next two lines to use RAM based sessions instead
* Use an absolute path to store sessions outside of the splunk tree
* Defaults to storage_type=file, storage_path=var/run/splunk

tools.decode.on = [True | False]
* Indicates if all strings that come into Cherrpy controller methods are
  decoded as unicode (assumes UTF-8 encoding).
* WARNING: Disabling this will likely break the application, as all incoming
  strings are assumed to be unicode.
* Defaults to True

tools.encode.on = [True | False]
* Encodes all controller method response strings into UTF-8 str objects in
  Python.
* WARNING: Disabling this will likely cause high byte character encoding to
  fail.
* Defaults to True

```

```

tools.encode.encoding = <codec>
* Force all outgoing characters to be encoded into UTF-8.
* This only works with tools.encode.on set to True.
* By setting this to utf-8, CherryPy's default behavior of observing the
* Accept-Charset header is overwritten and forces utf-8 output. Only change
  this if you know a particular browser installation must receive some other
  character encoding (Latin-1 iso-8859-1, etc)
* WARNING: Change this at your own risk.
* Defaults to utf08

tools.proxy.on = [True | False]
* Used for running Apache as a proxy for Splunk UI, typically for SSO
  configuration. See http://tools.cherrypy.org/wiki/BehindApache for more
  information.
* For Apache 1.x proxies only. Set this attribute to "true". This
  configuration instructs CherryPy (the Splunk Web HTTP server) to look for
  an incoming X-Forwarded-Host header and to use the value of that header to
  construct canonical redirect URLs that include the proper host name. For
  more information, refer to the CherryPy documentation on running behind an
  Apache proxy. This setting is only necessary for Apache 1.1 proxies. For
  all other proxies, the setting must be "false", which is the default.
* Defaults to False

tools.proxy.base = <scheme>://<URL>
* Used for setting the proxy base url in Splunk
* Defaults to an empty value

pid_path = <filepath>
* Specifies the path to the PID file
* Equals precisely and only var/run/splunk/splunkweb.pid
* NOTE: Do not change this parameter.

enabled_decomposers = <intention> [, <intention>]...
* Added in Splunk 4.2 as a short term workaround measure for apps which
  happen to still require search decomposition, which is deprecated
  with 4.2.
* Search decomposition will be entirely removed in a future release.
* Comma separated list of allowed intentions.
* Modifies search decomposition, which is a splunk-web internal behavior.
* Can be controlled on a per-app basis.
* If set to the empty string, no search decomposition occurs, which causes
  some usability problems with report builder.
* The current possible values are: addcommand, stats, addterm, addtermgt,
  addtermлт, setfields, excludefields, audit, sort, plot
* Default is 'plot', leaving only the plot intention enabled.
* When you need a good mulch, we recommend antibeethoven.
* However, for a traditional compost, antmozart is preferred.

simple_xml_perf_debug = [True | False]
* If True, simple xml dashboards will log some performance metrics to the
  browser console
* Defaults to False

job_min_polling_interval = <integer>
* Minimum polling interval for job in milliseconds (ms)
* The default value is 100
* This is the intial time wait for fetching results
* The poll period increases gradually from min interval to max interval when
  search is in queued state or parsing state (and not running state) for a
  some time.
* Set this value between 100 to job_max_polling_interval

job_max_polling_interval = <integer>
* Maximum polling interval for job in milliseconds (ms)
* The default value is 1000
* This is the maximum time wait for fetching results
* The recommended maximum value is 3000

acceptFrom = <network_acl> ...
* IMPORTANT: this setting only takes effect when appServerPorts is set to a
  non-zero value

```



- \* Lists a set of networks or addresses to accept connections from. These rules are separated by commas or spaces
- \* Each rule can be in the following forms:
  1. A single IPv4 or IPv6 address (examples: "10.1.2.3", "fe80::4a3")
  2. A CIDR block of addresses (examples: "10/8", "fe80:1234/32")
  3. A DNS name, possibly with a '\*' used as a wildcard (examples: "myhost.example.com", "\*.splunk.com")
  4. A single '\*' which matches anything
- \* Entries can also be prefixed with '!' to cause the rule to reject the connection. Rules are applied in order, and the first one to match is used. For example, "!10.1/16, \*" will allow connections from everywhere except the 10.1.\*.\* network.
- \* Defaults to "\*" (accept from anywhere)

maxThreads = <int>

- \* NOTE: this setting only takes effect when appServerPorts is set to a non-zero value
- \* Number of threads that can be used by active HTTP transactions. This can be limited to constrain resource usage.
- \* If set to 0 (the default) a limit will be automatically picked based on estimated server capacity.
- \* If set to a negative number, no limit will be enforced.

maxSockets = <int>

- \* NOTE: this setting only takes effect when appServerPorts is set to a non-zero value
- \* Number of simultaneous HTTP connections that we accept simultaneously. This can be limited to constrain resource usage.
- \* If set to 0 (the default) a limit will be automatically picked based on estimated server capacity.
- \* If set to a negative number, no limit will be enforced.

forceHttp10 = auto|never|always

- \* NOTE: this setting only takes effect when appServerPorts is set to a non-zero value
- \* When set to "always", the REST HTTP server will not use some HTTP 1.1 features such as persistent connections or chunked transfer encoding.
- \* When set to "auto" it will do this only if the client sent no User-Agent header, or if the user agent is known to have bugs in its HTTP/1.1 support.
- \* When set to "never" it always will allow HTTP 1.1, even to clients it suspects may be buggy.
- \* Defaults to "auto"

crossOriginSharingPolicy = <origin\_acl> ...

- \* IMPORTANT: this setting only takes effect when appServerPorts is set to a non-zero value
- \* List of HTTP Origins for which to return Access-Control-Allow-\* (CORS) headers
- \* These headers tell browsers that we trust web applications at those sites to make requests to the REST interface
- \* The origin is passed as a URL without a path component (for example "https://app.example.com:8000")
- \* This setting can take a list of acceptable origins, separated by spaces and/or commas
- \* Each origin can also contain wildcards for any part. Examples:
  - \*://app.example.com:\* (either HTTP or HTTPS on any port)
  - https://\*.example.com (any host under example.com, including example.com itself)
- \* An address can be prefixed with a '!' to negate the match, with the first matching origin taking precedence. For example, "!\*://evil.example.com:\* \*://\*.example.com:\*" to not avoid matching one host in a domain
- \* A single "\*" can also be used to match all origins
- \* By default the list is empty

allowSslCompression = true|false

- \* IMPORTANT: this setting only takes effect when appServerPorts is set to a non-zero value. When appServerPorts is zero or missing, this setting will always act as if it is set to "true"
- \* If set to true, the server will allow clients to negotiate SSL-layer data compression.
- \* Defaults to false. The HTTP layer has its own compression layer

which is usually sufficient.

```
allowSslRenegotiation = true|false
```

- \* IMPORTANT: this setting only takes effect when appServerPorts is set to a non-zero value
- \* In the SSL protocol, a client may request renegotiation of the connection settings from time to time.
- \* Setting this to false causes the server to reject all renegotiation attempts, breaking the connection. This limits the amount of CPU a single TCP connection can use, but it can cause connectivity problems especially for long-lived connections.
- \* Defaults to true

```
sendStrictTransportSecurityHeader = true|false
```

- \* IMPORTANT: this setting only takes effect when appServerPorts is set to a non-zero value
- \* If set to true, the REST interface will send a "Strict-Transport-Security" header with all responses to requests made over SSL.
- \* This can help avoid a client being tricked later by a Man-In-The-Middle attack to accept a non-SSL request. However, this requires a commitment that no non-SSL web hosts will ever be run on this hostname on any port. For example, if splunkweb is in default non-SSL mode this can break the ability of browser to connect to it. Enable with caution.
- \* Defaults to false

```
dedicatedIoThreads = <int>
```

- \* If set to zero, HTTP I/O will be performed in the same thread that accepted the TCP connection.
- \* If set to a non-zero value, separate threads will be run to handle the HTTP I/O, including SSL encryption.
- \* Defaults to "0"
- \* Typically this does not need to be changed. For most usage scenarios using the same thread offers the best performance.
- \* NOTE: this setting only takes effect when appServerPorts is set to a non-zero value

```
termsOfServiceDirectory = <directory>
```

- \* If set, we will look in this directory for a "Terms Of Service" document which each user must accept before logging into the UI
- \* Inside the directory the TOS should have a filename in the format "<number>.html" Where <number> is in the range 1 to 18446744073709551615. The active TOS is the filename with the larger number. For instance if there are two files in the directory named "123.html" and "456.html", then 456 will be the active TOS version.
- \* If a user hasn't accepted the current version of the TOS, they'll be required to the next time they try to log in. The acceptance times will be recorded inside a "tos.conf" file inside an app called "tos"
- \* The TOS file can either be a full HTML document or plain text, but it must have the ".html" suffix
- \* It is not necessary to restart Splunk when adding files to the TOS directory
- \* Defaults to empty (no TOS)
- \* NOTE: this setting only takes effect when appServerPorts is set to a non-zero value

```
appServerProcessShutdownTimeout = <nonnegative integer>[smhd]
```

- \* IMPORTANT: this setting only takes effect when appServerPorts is set to a non-zero value.
- \* The amount of time splunkd will wait "politely" for a Python-based application server process to handle outstanding/existing requests.
- \* If a Python-based application server process "outlives" this timeout, the process is forcibly killed.
- \* Defaults to '10m'

```
enableWebDebug = true|false
```

- \* Controls the visibility of the debug endpoints (i.e., /debug/\*\*splat).
- \* Defaults to false

```
allowableTemplatePaths = <directory> [, <directory>]...
```

- \* A comma separated list of template paths that may be added to template lookup white list.
- \* Paths are relative to \$SPLUNK\_HOME.
- \* Defaults to empty

```
enable_risky_command_check = <bool>
```

- \* Enable checks for data-exfiltrating search commands.
- \* default true

```

customFavicon = <pathToMyFile, myApp:pathToMyFile, or blank for default>
* Customize the favicon image across the entire application. If no favicon image file, the favicon defaults to the
Splunk favicon.
* Supported favicon image files are .ico files, and should be square images.
* Place favicon image file in default or manual location:
  * Default destination folder: $SPLUNK_HOME/etc/apps/search/appserver/static/customfavicon.
  * Example: If your favicon image is located at
$SPLUNK_HOME/etc/apps/search/appserver/static/customfavicon/favicon.ico, type customFavicon =
customfavicon/favicon.ico.
  * Manual location: $SPLUNK_HOME/etc/apps/<myApp>/appserver/static/<pathToMyFile>, and type customFavicon =
<myApp:pathToMyFile>.

loginCustomLogo = <fullUrl, pathToMyFile, myApp:pathToMyFile, or blank for default>
* Customize the logo image on the login page. If no image file, the logo defaults to the Splunk logo.
* Supported images are:
  * Full URL image file (secured or not secured), such as https://www.splunk.com/logo.png or
http://www.splunk.com/logo.png.
  * Image file, such as .jpg or .png. All image formats are supported.
  * Place logo image file in default or manual location:
    * Default destination folder: $SPLUNK_HOME/etc/apps/search/appserver/static/logincustomlogo.
    * Example: If your logo image is located at
$SPLUNK_HOME/etc/apps/search/appserver/static/logincustomlogo/logo.png, type loginCustomLogo =
logincustomlogo/logo.png.
    * Manual location: $SPLUNK_HOME/etc/apps/<myApp>/appserver/static/<pathToMyFile>, and type loginCustomLogo =
<myApp:pathToMyFile>.
* Logo height limit is 100px. Logo displays original dimensions unless the image height is greater than 100px.
* If image height exceeds 100px, the image is resized and width automatically adjusts.

loginBackgroundImageOption = [default| custom | none]
* Controls display of the background image of the login page.
* Defaults to "default".
  * "default" displays the Splunk default background image.
  * "custom" uses the background image defined by the backgroundImageCustomName setting.
  * "none" removes any background image on the login page. A dark background color is applied.

loginCustomBackgroundImage = <pathToMyFile or myApp:pathToMyFile>
* Customize the login page background image.
  * Supported image files include .jpg, .jpeg or .png with a maximum file size of 20Mb.
  * Landscape image is recommended, with a minimum resolution of 1024x640 pixels.
  * Using Splunk Web:
    * Upload a custom image to a manager page under General Settings.
    * The login page background image updates automatically.
  * Using the CLI or a text editor:
    * Set loginBackgroundImageOption = "custom".
    * Place custom image file in default or manual location:
      * Default destination folder: $SPLUNK_HOME/etc/apps/search/appserver/static/logincustombg.
      * Example: If your image is located at $SPLUNK_HOME/etc/apps/search/appserver/static/logincustombg/img.png,
type loginCustomBackgroundImage = logincustombg/img.png.
    * Manual location: $SPLUNK_HOME/etc/apps/<myApp>/appserver/static/<pathToMyFile>, and type
loginCustomBackgroundImage = <myApp:pathToMyFile>.
    * The login page background image updates automatically.
    * If no custom image is used, the default Splunk background image displays.

[framework]
# Put App Framework settings here
django_enable = [True | False]
* Specifies whether Django should be enabled or not
* Defaults to True
* Django will not start unless an app requires it

django_path = <filepath>
* Specifies the root path to the new App Framework files,
relative to $SPLUNK_HOME
* Defaults to etc/apps/framework

django_force_enable = [True | False]
* Specifies whether to force Django to start, even if no app requires it
* Defaults to False

#
# custom cherryypy endpoints

```

```
#

[endpoint:<python_module_name>]
* registers a custom python CherryPy endpoint
* The expected file must be located at:
  $SPLUNK_HOME/etc/apps/<APP_NAME>/appserver/controllers/<PYTHON_MODULE_NAME>.py
* This module's methods will be exposed at /custom/<APP_NAME>/<PYTHON_MODULE_NAME>/<METHOD_NAME>

#
# exposed splunkd REST endpoints
#
[expose:<unique_name>]
* Registers a splunkd-based endpoint that should be made available to the UI
  under the "/splunkd" and "/splunkd/__raw" hierarchies
* The name of the stanza doesn't matter as long as it starts with "expose:"
  Each stanza name must be unique, however

pattern = <url_pattern>
* Pattern to match under the splunkd /services hierarchy. For instance,
  "a/b/c" would match URIs "/services/a/b/c" and "/servicesNS/*/*a/b/c"
* The pattern should not include leading or trailing slashes
* Inside the pattern an element of "*" will match a single path element.
  For example, "a/*/c" would match "a/b/c" but not "a/1/2/c"
* A path element of "***" will match any number of elements. For example,
  "a/**/c" would match both "a/1/c" and "a/1/2/3/c"
* A path element can end with a "*" to match a prefix. For example,
  "a/elem-*/b" would match "a/elem-123/c"

methods = <method_lists>
* Comma separated list of methods to allow from the web browser
  (example: "GET,POST,DELETE")
* If not included, defaults to "GET"

oidEnabled = [0 | 1]
* If set to 1 indicates that the endpoint is capable of taking an embed-id
  as a query parameter
* Defaults to 0
* This is only needed for some internal splunk endpoints, you probably
  should not specify this for app-supplied endpoints

skipCSRFProtection = [0 | 1]
* If set to 1, tells splunkweb that it is safe to post to this endpoint
  without applying CSRF protection
* Defaults to 0
* This should only be set on the login endpoint (which already contains
  sufficient auth credentials to avoid CSRF problems)
```

## web.conf.example

```
# Version 6.5.0
#
# This is an example web.conf. Use this file to configure data web
# settings.
#
# To use one or more of these configurations, copy the configuration block
# into web.conf in $SPLUNK_HOME/etc/system/local/. You must restart Splunk
# to enable configurations.
#
# To learn more about configuration files (including precedence) please see
# the documentation located at
# http://docs.splunk.com/Documentation/Splunk/latest/Admin/Aboutconfigurationfiles

# This stanza heading must precede any changes.
[settings]

# Change the default port number:
httpport = 12800

# Also run the python application server on a non-default port:
```

```

appServerPorts = 12801

# Turn on SSL:
enableSplunkWebSSL = true
# absolute paths may be used here.
privKeyPath = /home/user/certs/myprivatekey.pem
caCertPath = /home/user/certs/mycacert.pem
# NOTE: non-absolute paths are relative to $SPLUNK_HOME

```

## wmi.conf

以下为 wmi.conf 的规范和示例文件。

### wmi.conf.spec

```

# Version 6.5.0
#
# This file contains possible attribute/value pairs for configuring Windows
# Management Instrumentation (WMI) access from Splunk.
#
# There is a wmi.conf in $SPLUNK_HOME/etc/system/default/. To set custom
# configurations, place a wmi.conf in $SPLUNK_HOME/etc/system/local/. For
# examples, see wmi.conf.example.
#
# You must restart Splunk to enable configurations.
#
# To learn more about configuration files (including precedence) please see
# the documentation located at
# http://docs.splunk.com/Documentation/Splunk/latest/Admin/Aboutconfigurationfiles

```

### 全局设置-----

```

#####
#----GLOBAL SETTINGS-----
#####GLOBAL SETTINGS-----

[settings]
* The settings stanza specifies various runtime parameters.
* The entire stanza and every parameter within it is optional.
* If the stanza is missing, Splunk assumes system defaults.

initial_backoff = <integer>
* How long, in seconds, to wait before retrying the connection to
  the WMI provider after the first connection error.
* If connection errors continue, the wait time doubles until it reaches
  the integer specified in max_backoff.
* Defaults to 5.

max_backoff = <integer>
* The maximum time, in seconds, to attempt to reconnect to the
  WMI provider.
* Defaults to 20.

max_retries_at_max_backoff = <integer>
* Once max_backoff is reached, tells Splunk how many times to attempt
  to reconnect to the WMI provider.
* Splunk will try to reconnect every max_backoff seconds.
* If reconnection fails after max_retries, give up forever (until restart).
* Defaults to 2.

checkpoint_sync_interval = <integer>
* The minimum wait time, in seconds, for state data (event log checkpoint)
  to be written to disk.
* Defaults to 2.

```

### 特定于输入的设置-----

```
#####
#---INPUT-SPECIFIC SETTINGS-----
#####INPUT-SPECIFIC SETTINGS-----

[WMI:$NAME]
* There are two types of WMI stanzas:
  * Event log: for pulling event logs. You must set the
    event_log_file attribute.
  * WQL: for issuing raw Windows Query Language (WQL) requests. You
    must set the wql attribute.
* Do not use both the event_log_file or the wql attributes. Use
  one or the other.

server = <comma-separated strings>
* A comma-separated list of servers from which to get data.
* If not present, defaults to the local machine.

interval = <integer>
* How often, in seconds, to poll for new data.
* This attribute is required, and the input will not run if the attribute is
  not present.
* There is no default.

disabled = [0|1]
* Specifies whether the input is enabled or not.
* 1 to disable the input, 0 to enable it.
* Defaults to 0 (enabled).

hostname = <host>
* All results generated by this stanza will appear to have arrived from
  the string specified here.
* This attribute is optional.
* If it is not present, the input will detect the host automatically.

current_only = [0|1]
* Changes the characteristics and interaction of WMI-based event
  collections.
* When current_only is set to 1:
  * For event log stanzas, this will only capture events that occur
    while Splunk is running.
  * For WQL stanzas, event notification queries are expected. The
    queried class must support sending events. Failure to supply
    the correct event notification query structure will cause
    WMI to return a syntax error.
  * An example event notification query that watches for process creation:
    * SELECT * FROM __InstanceCreationEvent WITHIN 1 WHERE
      TargetInstance ISA 'Win32_Process'.
* When current_only is set to 0:
  * For event log stanzas, all the events from the checkpoint are
    gathered. If there is no checkpoint, all events starting from
    the oldest events are retrieved.
  * For WQL stanzas, the query is executed and results are retrieved.
    The query is a non-notification query.
  * For example
    * Select * Win32_Process where caption = "explorer.exe"
* Defaults to 0.

batch_size = <integer>
* Number of events to fetch on each query.
* Defaults to 10.

index = <string>
* Specifies the index that this input should send the data to.
* This attribute is optional.
* When defined, "index=" is automatically prepended to <string>.
* Defaults to "index=main" (or whatever you have set as your default index).
```

**特定于事件日志的属性：**

```
#####
# Event log-specific attributes:
#####Event log-specific attributes:

event_log_file = <Application, System, etc>
* Tells Splunk to expect event log data for this stanza, and specifies
  the event log channels you want Splunk to monitor.
* Use this instead of WQL to specify sources.
* Specify one or more event log channels to poll. Multiple event log
  channels must be separated by commas.
* There is no default.

disable_hostname_normalization = [0|1]
* If set to true, hostname normalization is disabled
* If absent or set to false, the hostname for 'localhost' will be converted
  to %COMPUTERNAME%.
* 'localhost' refers to the following list of strings: localhost, 127.0.0.1,
  ::1, the name of the DNS domain for the local computer, the fully
  qualified DNS name, the NetBIOS name, the DNS host name of the local
  computer
```

### 特定于 WQL 的属性：

```
#####
# WQL-specific attributes:
#####WQL-specific attributes:

wql = <string>
* Tells Splunk to expect data from a WMI provider for this stanza, and
  specifies the WQL query you want Splunk to make to gather that data.
* Use this if you are not using the event_log_file attribute.
* Ensure that your WQL queries are syntactically and structurally correct
  when using this option.
* For example,
  SELECT * FROM Win32_PerfFormattedData_PerfProc_Process WHERE Name = "splunkd".
* If you wish to use event notification queries, you must also set the
  "current_only" attribute to 1 within the stanza, and your query must be
  appropriately structured for event notification (meaning it must contain
  one or more of the GROUP, WITHIN or HAVING clauses.)
* For example,
  SELECT * FROM __InstanceCreationEvent WITHIN 1 WHERE TargetInstance ISA
  'Win32_Process'
* There is no default.

namespace = <string>
* The namespace where the WMI provider resides.
* The namespace spec can either be relative (root\cimv2) or absolute
  (\\server\root\cimv2).
* If the server attribute is present, you cannot specify an absolute
  namespace.
* Defaults to root\cimv2.
```

### wmi.conf.example

```
# Version 6.5.0
#
# This is an example wmi.conf. These settings are used to control inputs
# from WMI providers. Refer to wmi.conf.spec and the documentation at
# splunk.com for more information about this file.
#
# To use one or more of these configurations, copy the configuration block
# into wmi.conf in $SPLUNK_HOME\etc\system\local\. You must restart Splunk
# to enable configurations.
#
# To learn more about configuration files (including precedence) please see
# the documentation located at
```

```

# http://docs.splunk.com/Documentation/Splunk/latest/Admin/Aboutconfigurationfiles

# This stanza specifies runtime parameters.

[settings]
initial_backoff = 5
max_backoff = 20
max_retries_at_max_backoff = 2
checkpoint_sync_interval = 2

# Pull events from the Application, System and Security event logs from the
# local system every 10 seconds. Store the events in the "wmi_eventlog"
# Splunk index.

[WMI:LocalApplication]
interval = 10
event_log_file = Application
disabled = 0
index = wmi_eventlog

[WMI:LocalSystem]
interval = 10
event_log_file = System
disabled = 0
index = wmi_eventlog

[WMI:LocalSecurity]
interval = 10
event_log_file = Security
disabled = 0
index = wmi_eventlog

# Gather disk and memory performance metrics from the local system every
# second. Store event in the "wmi_perfmon" Splunk index.

[WMI:LocalPhysicalDisk]
interval = 1
wql = select Name, DiskBytesPerSec, PercentDiskReadTime, PercentDiskWriteTime, PercentDiskTime from
Win32_PerfFormattedData_PerfDisk_PhysicalDisk
disabled = 0
index = wmi_perfmon

[WMI:LocalMainMemory]
interval = 10
wql = select CommittedBytes, AvailableBytes, PercentCommittedBytesInUse, Caption from
Win32_PerfFormattedData_PerfOS_Memory
disabled = 0
index = wmi_perfmon

# Collect all process-related performance metrics for the splunkd process,
# every second. Store those events in the "wmi_perfmon" index.
[WMI:LocalSplunkdProcess]
interval = 1
wql = select * from Win32_PerfFormattedData_PerfProc_Process where Name = "splunkd"
disabled = 0
index = wmi_perfmon

# Listen from three event log channels, capturing log events that occur only
# while Splunk is running, every 10 seconds. Gather data from three remote
# servers srv1, srv2 and srv3.

[WMI:TailApplicationLogs]
interval = 10
event_log_file = Application, Security, System
server = srv1, srv2, srv3
disabled = 0
current_only = 1
batch_size = 10

# Listen for process-creation events on a remote machine, once a second.

[WMI:ProcessCreation]

```



```

interval = 1
server = remote-machine
wql = select * from __InstanceCreationEvent within 1 where TargetInstance isa 'Win32_Process'
disabled = 0
current_only = 1
batch_size = 10

# Receive events whenever someone connects or removes a USB device on
# the computer, once a second.

[WMI:USBChanges]
interval = 1
wql = select * from __InstanceOperationEvent within 1 where TargetInstance ISA 'Win32_PnPEntity' and
TargetInstance.Description='USB Mass Storage Device'
disabled = 0
current_only = 1
batch_size = 10

```

## workflow\_actions.conf

以下为 workflow\_actions.conf 的规范和示例文件。

### workflow\_actions.conf.spec

```

# Version 6.5.0
#
# This file contains possible attribute/value pairs for configuring workflow
# actions in Splunk.
#
# There is a workflow_actions.conf in $SPLUNK_HOME/etc/apps/search/default/.
# To set custom configurations, place a workflow_actions.conf in either
# $SPLUNK_HOME/etc/system/local/ or add a workflow_actions.conf file to your
# app's local/ directory. For examples, see workflow_actions.conf.example.
# You must restart Splunk to enable configurations, unless editing them
# through the Splunk manager.
#
# To learn more about configuration files (including precedence) please see
# the documentation located at
# http://docs.splunk.com/Documentation/Splunk/latest/Admin/Aboutconfigurationfiles

```

### 全局设置

```

# GLOBAL SETTINGS
# Use the [default] stanza to define any global settings.
# * You can also define global settings outside of any stanza, at the top
#   of the file.
# * Each conf file should have at most one default stanza. If there are
#   multiple default stanzas, attributes are combined. In the case of
#   multiple definitions of the same attribute, the last definition in the
#   file wins.
# * If an attribute is defined at both the global level and in a specific
#   stanza, the value in the specific stanza takes precedence.

#####
# General required settings:
# These apply to all workflow action types.
#####

type = <string>
* The type of the workflow action.
* If not set, Splunk skips this workflow action.

label = <string>
* The label to display in the workflow action menu.
* If not set, Splunk skips this workflow action.

```

```
#####
# General optional settings:
# These settings are not required but are available for all workflow
# actions.
#####

fields = <comma or space separated list>
* The fields required to be present on the event in order for the workflow
  action to be applied.
* When "display_location" is set to "both" or "field_menu", the workflow
  action will be applied to the menu's corresponding to the specified
  fields.
* If fields is undefined or set to *, the workflow action is applied to all
  field menus.
* If the * character is used in a field name, it is assumed to act as a
  "globber". For example host* would match the fields hostname, hostip, etc.
* Acceptable values are any valid field name, any field name including the *
  character, or * (e.g. *_ip).
* Defaults to *

eventtypes = <comma or space separated list>
* The eventtypes required to be present on the event in order for the
  workflow action to be applied.
* Acceptable values are any valid eventtype name, or any eventtype name plus
  the * character (e.g. host*).

display_location = <string>
* Dictates whether to display the workflow action in the event menu, the
  field menus or in both locations.
* Accepts field_menu, event_menu, or both.
* Defaults to both.

disabled = [True | False]
* Dictates whether the workflow action is currently disabled
* Defaults to False
```

## 使用字段名称将值嵌入到 workflow 操作设置

```
#####
# Using field names to insert values into workflow action settings
#####Using field names to insert values into
workflow action settings

# Several settings detailed below allow for the substitution of field values
# using a special variable syntax, where the field's name is enclosed in
# dollar signs. For example, $_raw$, $hostip$, etc.
#
# The settings, label, link.uri, link.postargs, and search.search_string all
# accept the value of any valid field to be substituted into the final
# string.
#
# For example, you might construct a Google search using an error message
# field called error_msg like so:
# link.uri = http://www.google.com/search?q=$_error_msg$.
#
# Some special variables exist to make constructing the settings simpler.

$@field_name$
* Allows for the name of the current field being clicked on to be used in a
  field action.
* Useful when constructing searches or links that apply to all fields.
* NOT AVAILABLE FOR EVENT MENUS

$@field_value$
* Allows for the value of the current field being clicked on to be used in a
  field action.
* Useful when constructing searches or links that apply to all fields.
* NOT AVAILABLE FOR EVENT MENUS

$@sid$
```

\* The sid of the current search job.

`$@offset$`

\* The offset of the event being clicked on in the list of search events.

`$@namespace$`

\* The name of the application from which the search was run.

`$@latest_time$`

\* The latest time the event occurred. This is used to disambiguate similar events from one another. It is not often available for all fields.

## 字段操作类型

```
#####
# Field action types
#####Field action types
```

```
#####
# Link type:
# Allows for the construction of GET and POST requests via links to external
# resources.
#####
```

`link.uri = <string>`

\* The URI for the resource to link to.  
\* Accepts field values in the form `$<field name>$`, (e.g. `$_raw$`).  
\* All inserted values are URI encoded.  
\* Required

`link.target = <string>`

\* Determines if clicking the link opens a new window, or redirects the current window to the resource defined in `link.uri`.  
\* Accepts: "blank" (opens a new window), "self" (opens in the same window)  
\* Defaults to "blank"

`link.method = <string>`

\* Determines if clicking the link should generate a GET request or a POST request to the resource defined in `link.uri`.  
\* Accepts: "get" or "post".  
\* Defaults to "get".

`link.postargs.<int>.<key/value> = <value>`

\* Only available when `link.method = post`.  
\* Defined as a list of key / value pairs like such that `foo=bar` becomes:  
    `link.postargs.1.key = "foo"`  
    `link.postargs.1.value = "bar"`  
\* Allows for a configurable method of defining multiple identical keys (e.g.):  
    `link.postargs.1.key = "foo"`  
    `link.postargs.1.value = "bar"`  
    `link.postargs.2.key = "foo"`  
    `link.postargs.2.value = "boo"`  
    ...  
\* All values are html form encoded appropriately.

```
#####
# Search type:
# Allows for the construction of a new search to run in a specified view.
#####
```

`search.search_string = <string>`

\* The search string to construct.  
\* Accepts field values in the form `$<field name>$`, (e.g. `$_raw$`).  
\* Does NOT attempt to determine if the inserted field values may break quoting or other search language escaping.  
\* Required

`search.app = <string>`

\* The name of the Splunk application in which to perform the constructed

```

    search.
* By default this is set to the current app.

search.view = <string>
* The name of the view in which to preform the constructed search.
* By default this is set to the current view.

search.target = <string>
* Accepts: blank, self.
* Works in the same way as link.target. See link.target for more info.

search.earliest = <time>
* Accepts absolute and Splunk relative times (e.g. -10h).
* Determines the earliest time to search from.

search.latest = <time>
* Accepts absolute and Splunk relative times (e.g. -10h).
* Determines the latest time to search to.

search.preserve_timerange = <boolean>
* Ignored if either the search.earliest or search.latest values are set.
* When true, the time range from the original search which produced the
  events list will be used.
* Defaults to false.

```

## workflow\_actions.conf.example

```

# Version 6.5.0
#
# This is an example workflow_actions.conf. These settings are used to
# create workflow actions accessible in an event viewer. Refer to
# workflow_actions.conf.spec and the documentation at splunk.com for more
# information about this file.
#
# To use one or more of these configurations, copy the configuration block
# into workflow_actions.conf in $SPLUNK_HOME/etc/system/local/, or into your
# application's local/ folder. You must restart Splunk to enable
# configurations.
#
# To learn more about configuration files (including precedence) please see
# the documentation located at
# http://docs.splunk.com/Documentation/Splunk/latest/Admin/Aboutconfigurationfiles
#
# These are the default workflow actions and make extensive use of the
# special parameters: $@namespace$, $@sid$, etc.

[show_source]
type=link
fields = _cd, source, host, index
display_location = event_menu
label = Show Source
link.uri = /app/$@namespace$/show_source?sid=$@sid$&offset=$@offset$&latest_time=$@latest_time$

[ifx]
type = link
display_location = event_menu
label = Extract Fields
link.uri = /ifx?sid=$@sid$&offset=$@offset$&namespace=$@namespace$

[etb]
type = link
display_location = event_menu
label = Build Eventtype
link.uri = /etb?sid=$@sid$&offset=$@offset$&namespace=$@namespace$

# This is an example workflow action which will be displayed in a specific
# field menu (clientip).

[whois]
display_location = field_menu

```

```

fields = clientip
label = Whois: $clientip$
link.method = get
link.target = blank
link.uri = http://ws.arin.net/whois/?queryinput=$clientip$
type = link

# This is an example field action which will allow a user to search every
# field value in Google.

[Google]
display_location = field_menu
fields = *
label = Google @$field_name$
link.method = get
link.uri = http://www.google.com/search?q=@$field_value$
type = link

# This is an example post link that will send its field name and field value
# to a fictional bug tracking system.

[Create JIRA issue]
display_location = field_menu
fields = error_msg
label = Create JIRA issue for $error_class$
link.method = post
link.postargs.1.key = error
link.postargs.1.value = $error_msg$
link.target = blank
link.uri = http://127.0.0.1:8000/jira/issue/create
type = link

# This is an example search workflow action that will be displayed in an
# event's menu, but requires the field "controller" to exist in the event in
# order for the workflow action to be available for that event.

[Controller req over time]
display_location = event_menu
fields = controller
label = Requests over last day for $controller$
search.earliest = -3d
search.search_string = sourcetype=rails_app controller=$controller$ | timechart span=1h count
search.target = blank
search.view = charting
type = search

```