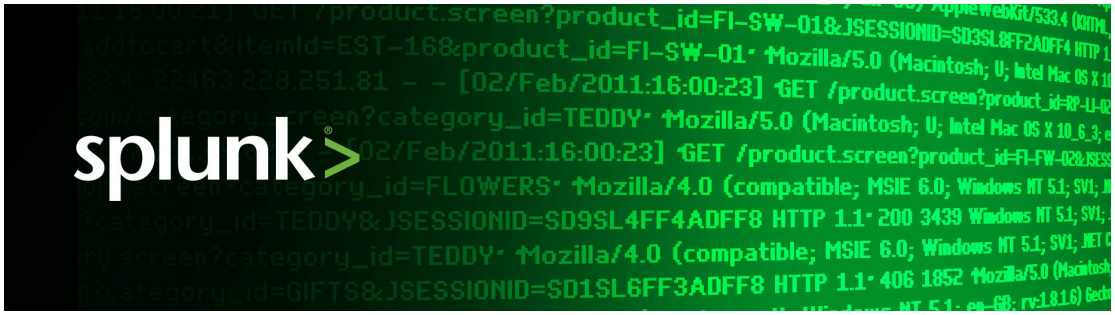


Splunk® Enterprise 7.2.0

更新 Splunk Enterprise 实例

生成时间：2018 年 10 月 17 日，上午 11:15

Table of Contents

部署服务器和转发器管理

关于部署服务器和转发器管理

重要提示：在阅读本手册之前，您应熟悉 Splunk Enterprise 分布式环境，如同《分布式部署手册》所示。

Splunk Enterprise 提供部署服务器，并可通过其转发器管理界面来管理 Splunk Enterprise 分布式实例中的更新过程。

什么是部署服务器？

部署服务器是一款用来将配置、应用和内部更新分发到各组 Splunk Enterprise 实例的工具。您可以使用部署服务器将更新分发到大多数 Splunk Enterprise 组件：转发器、非群集索引器和搜索头。

部署服务器只是一个已配置为用来管理其他各组 Splunk Enterprise 实例中的更新过程的 Splunk Enterprise 实例。部署服务器正在将更新部署到实例，根据实例的数量，部署服务器实例可能需要专门用于管理更新。有关更多信息，请参阅"计划部署"。

> **部署服务器处理的是现有 Splunk Enterprise 安装的配置和内容更新。**不能将部署服务器用于 Splunk Enterprise 的初始安装或升级安装，也不能将其用于通用转发器。要了解如何安装和部署 Splunk Enterprise，请参阅完整 Splunk Enterprise 的"分步安装程序"和 Splunk Enterprise 通用转发器的"安装通用转发器软件"。要了解如何将部署升级为新的 Splunk Enterprise 版本，请参阅*《安装手册》*中的"升级分布式 Splunk Enterprise 部署"。

部署服务器是强制的吗？

不需要部署服务器来管理转发器和其他 Splunk Enterprise 实例。如果喜欢，您可使用第三方工具，如 Chef、Puppet、Salt 或 Windows 配置工具中的一种。

什么是转发器管理？

转发器管理是一个建立在部署服务器之上的图形界面，可用于轻松配置部署服务器和监视部署更新状态。虽然其主要目的是用来管理大量转发器，但您也可以使用转发器管理配置部署服务器来进行更新，包括管理和部署非群集索引器和搜索头的更新。在大多数场合下，转发器管理的功能与部署服务器的功能相同。有关更多信息，请参阅"转发器管理概述"。

重要提示：如果升级 6.0 版本之前的部署服务器，则现有的 `serverclass.conf` 文件可能与转发器管理界面不兼容。这是因为转发器管理只能处理部分通过 `serverclass.conf` 实现配置的子集。某些情况下，您可能还需要直接处理 `serverclass.conf`，而不是改为使用转发器管理作为配置工具。有关哪些配置与转发器管理兼容以及如何处理部署服务器升级的详细信息，请参阅主题"兼容性和转发器管理"。

部署服务器功能

部署服务器可以按照共同特性对 Splunk Enterprise 组件进行分组，然后根据分组结果分发内容。

例如，如果您的组织内具有用于满足各种不同需求的 Splunk Enterprise 实例，则这些实例的配置很可能随着使用者和用途的不同而变化。某些实例可能供服务团队使用，因此配置有特定的应用便于加速 Windows 桌面问题的处理。另一些实例则可能供运营人员使用，因此设置了一些不同的应用便于跟踪网络问题、安全事故和电子邮件流量管理。还有一些实例可能供运营小组的网络托管团队使用。

您不必分别管理和维护这些分散的 Splunk Enterprise 实例，而是可以根据这些实例的用途对其进行分组，确定每组所需的配置和应用，然后使用部署服务器根据需要更新其应用和配置。

除了按用途分组 Splunk Enterprise 实例外，还可以指定其他一些有用的分组类型。例如，可以按照操作系统或硬件类型、版本、地理位置或时区分组实例。

管理转发器组的配置便是一个关键用例。例如，如果转发器驻留在不同类型的计算机上，则可以使用部署服务器将不同的内容部署到每种计算机上。Windows 转发器可以更新一组配置；而 Linux 转发器则可以更新另一组配置等等。

部署服务器和群集

不能使用部署服务器更新索引器群集对等节点或搜索头群集成员。

索引器群集

切勿使用部署服务器或转发器管理来管理索引器群集中对等节点（索引器）之间的配置文件。应改为使用**配置软件包**方法。但是，您可使用部署服务器将更新分发到主节点，该主节点随后会使用配置软件包方法将更新分发到对等节

点。请参阅*管理索引器和索引器群集*手册中的“更新通用对等节点配置”。

搜索头群集

切勿使用部署服务器更新搜索头群集成员。

不支持通过部署服务器将配置或应用分发到群集成员。要在成员集中分发配置，必须使用搜索头群集 deployer。请参阅*《分布式搜索》*手册中的“使用 deployer 分布应用和配置更新”。

部署服务器架构

部署服务器用于将内容和配置（统称为**部署应用**）分发到分组为不同**服务器类**的**部署客户端**上。部署应用既可以是完整的**应用**（例如 Splunkbase 中提供的应用），也可以只是一些简单的配置组。

关键架构元素

部署服务器是用作任意多个称为“部署客户端”的其他实例的中央配置管理器的 Splunk Enterprise 实例。任何完整的 Splunk Enterprise 实例都可用作部署服务器，甚至包括在本地索引数据的这类实例。部署服务器本身不能是客户端。

部署客户端是由部署服务器远程配置的 Splunk 实例。部署客户端可以是通用转发器、重型转发器、索引器或搜索头。每个部署客户端属于一个或多个服务器类。

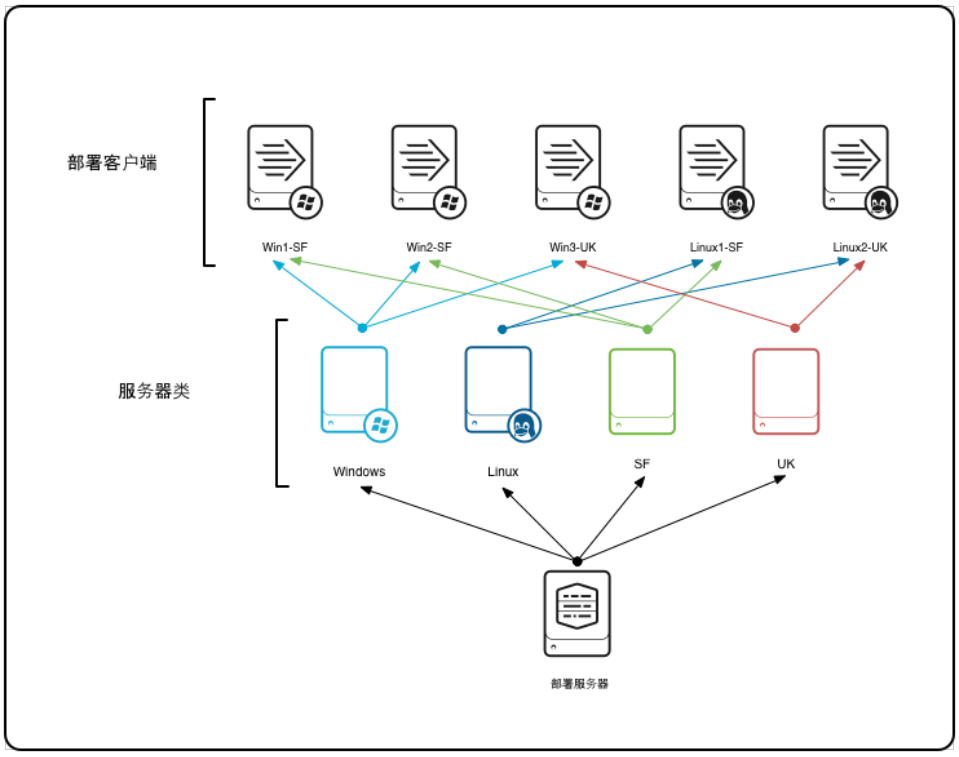
部署应用是一组在部署服务器上进行维护且作为单元部署到服务器类的客户端的内容（包括配置文件）。一个部署应用可能只包含一个配置文件，也可能包含多个文件。应用可以随着时间的推移更新为新内容，然后重新部署到其指定客户端。部署应用既可以是现有的 Splunk Enterprise 应用，也可以是出于部署目的而专门开发的用于分组某些内容的应用。

注意：术语“应用”的含义在部署服务器上下文中与在常规 Splunk Enterprise 上下文中存在一定的区别。有关 Splunk Enterprise 应用的更多常规信息，请参阅《管理员手册》中的“应用和加载项是什么？”。

服务器类是一组共享一个或多个所定义特性的部署客户端。例如，可以将所有 Windows 客户端分组为一个服务器类，所有 Linux 客户端分组为另一个服务器类。服务器类用于将一组部署客户端映射到一个或多个部署应用。通过新建服务器类，即可指示部署服务器某一组特定的客户端应接收一组特定应用形式的配置更新。

关联方式

该图对部署服务器及其一组部署客户端和服务器类进行了概念性介绍：



在本例中，每个部署客户端都是一个属于两个服务器类的 Splunk Enterprise 转发器，其中，一个服务器类代表客户端操作系统，另一个服务器类代表客户端地理位置。部署服务器对服务器类列表进行维护，并使用这些服务器类来

确定要向每个客户端分发的内容。有关如何通过实施此类排列来管理客户端内容流的示例，请参阅“将配置部署到多个转发器”。

有关部署应用的更多信息，请参阅“新建部署应用”。有关服务器类的更多信息，请参阅“关于服务器类”。有关部署客户端的更多信息，请参阅“配置部署客户端”。

关键术语摘要

以下概括了关键定义：

术语	含义
部署服务器	用作中央配置管理器的 Splunk Enterprise 实例。可将配置更新部署到其他实例。也指包含部署服务器、客户端和应用在内的全部配置更新工具。
部署客户端	远程配置的 Splunk Enterprise 实例。接收部署服务器发出的更新。
服务器类	一组部署客户端共享的部署配置类别。一个部署客户端可以属于多个服务器类。
部署应用	部署到一个或多个服务器类的成员的内容单位。

部署更新方式

部署更新过程的工作方式如下：

1. 每个部署客户端定期轮询部署服务器并标明自身身份。

2. 部署服务器根据客户端所属的服务器类确定客户端的一组部署应用。

3. 部署服务器向客户端发送属于该客户端的应用列表以及这些应用的当前校验和。

4. 客户端将来自部署服务器的应用信息与其自身的应用信息相比较，确定是否有任何新应用或更改应用需要下载。

5. 如果有新应用或更新应用，部署客户端将进行下载。

6. 根据给定应用的配置情况，应用更改可能需要在重启客户端之后才生效。

转发器管理概述

转发器管理界面是一个用于新建**服务器类**的交互式可视工具，服务器类用于将**部署客户端**映射到**部署应用**。也可以使用转发器管理来管理和监视您的部署。

转发器管理界面将服务器类配置保存在一个 serverclass.conf 文件中，该文件位于部署服务器上的 `$SPLUNK_HOME/etc/system/local` 下。

界面功能

转发器管理界面的主要用途是新建和编辑服务器类。也可以使用该界面实现其他一些功能：

- 跟踪系统状态
- 监视部署活动
- 查看应用、客户端和服务器类之间的关联
- 配置应用行为
- 从客户端卸载应用

访问转发器管理界面

您可通过部署服务器上的 Splunk Web 访问转发器管理界面。要打开该界面，请执行以下操作：

1. 单击 Splunk Web 顶部的**设置**链接。随即弹出一个包含指向该组系统界面的链接的窗口。

2. 在**分布式环境**部分中选择“转发器管理”。此时将转到界面主页。

以下示例显示了您已经拥有一些应用、客户端和服务器类时转发器管理可能显示的主页：

该页面包含以下功能（由上而下）：

- **存储库位置**。存储库位置为部署应用在部署服务器上的驻留位置。
- 一个状态部分，其中包含部署客户端和最近下载的相关信息。
- 三个选项卡：
  - **应用**。此选项卡列出了位于存储库位置中的一组部署应用及其状态。可在此编辑某些应用属性。
  - **服务器类**。此选项卡列出了一组服务器类及其状态。可在此新建新服务器类和编辑现有服务器类。您也可以钻取现有服务器类来了解与其关联的应用和客户端的信息。对于未加配置的新部署服务器，此列表为空。
  - **客户端**。此选项卡列出了部署服务器的所有客户端及其状态信息。可通过多种方式过滤此列表以限制当前视图。

有关此页面的更多信息，请参阅“使用转发器管理定义服务器类”。

界面限制

转发器管理界面支持绝大多数部署服务器用例。不过，对于一些复杂的配置要求，您可能需要直接编辑 `serverclass.conf`。

重要提示：如果由使用转发器管理改为直接编辑 `serverclass.conf`，您可能无法使用转发器管理执行任何后续配置操作。这是因为转发器管理界面只能处理部分通过 `serverclass.conf` 实现的配置。

以下是该界面相对于直接编辑配置文件而言存在的一些限制：

- 任何应用在所有服务器类中的部署行为必须相同。例如，不能指定应用在通过某个服务器类下载时引发客户端重新启动，而通过另一服务器类下载时却不引发重新启动。
- 不能修改部署服务器上的默认 `repositoryLocation`。
- 用来控制白名单和黑名单组合行为的 `filterType` 必须使用 `whitelist` 的默认值。
- 只有服务器类级别支持客户端过滤器。

有关转发器管理限制以及转发器管理与 `serverclass.conf` 之间兼容性的详细信息，请参阅主题“兼容性和转发器管理”。

配置部署系统

计划部署

要设置部署服务器，您需要配置部署服务器和部署客户端，不过大多数配置都发生在部署服务器端。需要执行的主要操作如下：

- 配置部署客户端以连接到部署服务器。
- 在部署服务器上新建用来保存部署应用的目录并向其中填充内容。
- 新建部署客户端与应用目录之间的映射（**服务器类**）。

虽然上述操作的执行顺序在一定程度上由您决定，但以下“基本步骤”中还是提供了建议的程序。

设置完客户端、应用目录和映射后，即可向应用目录中填充内容。您可以随时指示部署服务器将应用目录中的新内容或更新内容分发到这些目录所映射到的客户端。

重要提示：切勿使用部署服务器或转发器管理将更新分发到索引器群集中对等节点（索引器）。同样，切勿使用部署服务器将应用或配置文件分发到搜索头群集成员。请参阅“部署服务器和群集”。

部署服务器系统要求

部署服务器计算机要求

由于应用下载期间的 CPU 使用率和内存使用率非常高，因此建议将部署服务器实例驻留在专门的计算机上。

操作系统兼容性

Unix 部署服务器可更新 Windows 和 Unix 客户端。但是，Windows 部署服务器只能与 Windows 客户端一起使用。

使用 Unix 部署服务器更新 Unix 部署客户端。采用脚本式输入、告警，搜索命令等的应用，在从 Windows 部署到 Unix 时可能会遇到权限问题。尤其是，脚本和其他程序一交付到 Unix 客户端就会被设置为可执行。

客户端版本兼容性

7.x 部署服务器与运行 6.0 及以上版本的部署客户端兼容。

部署服务器和其他角色

对于大多数部署，部署服务器必须运行在不用作索引器或搜索头的专用 Splunk Enterprise 实例上。例外情况是当部署服务器只有少量（50 个或更少）客户端的时候。在这些受限情况下，一个索引器或搜索头有可能兼任部署服务器。

或者您可以在一个部署服务器上托管这些管理组件中的任意一个，但只能在部署服务器客户端数量为 50 或少于 50 时进行：

- **许可证主服务器**
- **监视控制台**
- **搜索头群集 Deployer**

任何情况下，均不得将部署服务器和索引器群集主节点放在同一位置。

群集主节点和部署服务器在执行各自任务时均使用大量的系统资源。主节点需要持续可靠地访问资源，以持续管理群集，部署服务器可能会在向其部署客户端部署更新时轻松覆盖这些资源。

有关部署服务器规模的更多信息，请参阅“评估部署服务器性能”。

有关管理组件共存的常规讨论，请参阅*《分布式部署手册》*中的“帮助管理部署的组件”。

配置内容

您需要配置部署服务器和部署客户端：

- **在每个部署客户端上，**通过调用 CLI 命令、直接编辑配置文件或者在安装期间（仅适用于 Windows 通用转发器）指定部署客户端的部署服务器。
- **在部署服务器上，**新建用来存放部署应用的目录。然后可以使用转发器管理来定义用于封装客户端/应用映射的服务器类。

基本步骤

要设置部署服务器，您需要在部署客户端和部署服务器上执行几个步骤。虽然步骤顺序在一定程度上是可选的，但此处还是提供了建议的顺序：

1.确定远程配置需求。需要考虑的问题包括：

- 希望远程配置哪些类型的 Splunk Enterprise 组件？例如：转发器、索引器。
- 在每种组件中，哪些特性指示配置需求？例如：计算机类型、地理位置、应用程序。

2.按配置需求分组部署客户端。可以按照应用程序、计算机类型或其他任何对部署拓扑有意义的条件分组客户端。一个客户端可以是多个组的成员。例如，转发器 x 可以是计算机类型 linux-x86_64、北美地理位置和安全应用程序组的成员，转发器 y 可以是计算机类型 windows-intel、亚洲地理位置和安全应用程序组的成员。

这些组构成了服务器类的基础。服务器类将一组部署客户端映射到为其部署的各组内容（以部署应用形式）。一个客户端可以属于多个服务器类。有关如何将部署客户端分组到服务器类的指导，请参阅“关于服务器类”。

3.选择其中一个要用作部署服务器的 Splunk Enterprise 实例。Splunk Enterprise 会自动启用部署服务器功能，因此在此步骤中，除了选择实例外，无需执行任何其他操作。您将在该实例中放置可下载内容并定义服务器类。部署服务器将内部更新分发给它的部署客户端集合。

在多数情况下，部署服务器需要专用的 Splunk Enterprise 实例。请参阅“部署服务器系统要求”。

重要提示：部署服务器本身不能是部署客户端。否则，以下错误将显示在 `splunkd.log` 中：“该 DC 与其 DS 共享一个 Splunk 实例：不支持的配置”。

4.在每个部署客户端上，指定第 3 步中选择的部署服务器。请参阅“配置部署客户端”了解详细信息。后续可以添加更多客户端。

5.在部署服务器的文件系统中，为部署应用新建用来存放计划分发到客户端上的内容的目录。立即或稍后将应用内容放入这些目录中。请参阅“新建部署应用”了解详细信息。后续可以添加更多部署应用。

6.在部署服务器上，新建将部署客户端映射到部署应用的服务器类。有关配置服务器类的详细信息，请参阅“关于服务器类”。

注意：大多数情况下，转发器管理界面可以处理服务器类配置。在一些不常见的情况下，则可能需要直接编辑基本的配置文件。无论是使用转发器管理还是直接编辑配置文件，基本步骤都相同。

完成此配置过程后，即可开始向客户端分发内容。有关如何将新内容部署到客户端的详细信息，请参阅“将应用部署到客户端”。

SSL 加密

使用默认证书的 SSL 加密是预先启用的。如果更改部署服务器上的 SSL 配置，则也必须更改其部署客户端上的这一配置。部署服务器与其客户端的 `splunkd` 管理端口 SSL 设置必须一致。*必须都启用 SSL 或都禁用 SSL。*

要在 Splunk Enterprise 实例上禁用 SSL 配置，可将 server.conf 中的属性 `enableSplunkdSSL` 设置为 "false"：

```
[sslConfig]
enableSplunkdSSL = false
```

有关对部署服务器使用 SSL 的详细信息，请参阅《*确保 Splunk 安全*》手册中的“确保部署服务器和客户端安全”。

配置部署客户端

本主题介绍如何设置**部署客户端**以从**部署服务器**接收内容。大多数情况下，您只需指定要将客户端连接到的部署服务器。

虽然此步骤在部署客户端而不是部署服务器上执行，但这仍是整个部署服务器系统配置的基本组成部分。

重要提示：部署服务器本身不能是部署客户端。否则，以下错误将显示在 `splunkd.log` 中：“该 DC 与其 DS 共享一个 Splunk 实例：不支持的配置”。可能会出现部署客户端无法联系部署服务器的情况。

指定部署服务器

在每个客户端上，您必须指定将连接的部署服务器。通过配置客户端的 deploymentclient.conf 文件完成。每个部署客户端必须有一个唯一网络主机名。

此文件有三种配置方法：

- 使用 CLI。请参阅本主题后面介绍的“使用 CLI”。
- 直接编辑文件。请参阅本主题后面介绍的“编辑 deploymentclient.conf”。

- **仅适用于 Windows 通用转发器：**可以在安装过程中将 Windows 转发器配置为部署客户端。请参阅如下*《通用转发器》*手册主题：
  - 通过安装程序安装 Windows 通用转发器
  - 通过命令行安装 Windows 通用转发器

重要提示：请勿使用部署服务器推动 `deploymentclient.conf` 更新到部署客户端。可能会出现部署客户端无法联系部署服务器的情况。

使用 CLI

在部署客户端上运行以下 CLI 命令：

```
splunk set deploy-poll <IP_address/hostname>:<management_port>
splunk restart
```

使用客户端要连接到的部署服务器的 `IP_address/hostname` 和 `management_port`。

例如：

```
splunk set deploy-poll deploymentserver.splunk.mycompany.com:8089
splunk restart
```

编辑 deploymentclient.conf

也可以在 `$SPLUNK_HOME/etc/system/local` 中直接新建和编辑 `deploymentclient.conf` 文件。

语法

`deploymentclient.conf` 文件需要两个**段落**：

段落	用途为何
`[deployment-client]`	配置多个属性，包括查找新内容或更新内容的位置。通常不需要更改此段落的默认值。
`[target-broker:deploymentServer]`	指定该客户端部署服务器的位置。`deploymentServer` 是部署服务器的默认名称。必须在此段落下指定部署服务器。

此文件有大量可选属性，但在大多数部署当中，都只需设置 `[target-broker:deploymentServer]` 段落下的 `targetUri` 属性。此属性指定了客户端的部署服务器。以下是该属性的语法：

属性	用途为何	默认
`targetUri`	指定部署服务器连接信息。 设置为 `<deployment_server_URI>:<management_port>`。管理端口通常为 8089。	n/a

有关 `deploymentclient.conf` 属性的完整列表，请参阅《管理员手册》中的 deploymentclient.conf 规范文件。

重要提示：必须重新启动部署客户端，更改才能生效。

示例

以下为典型的客户端配置：

```
[deployment-client]

[target-broker:deploymentServer]
targetUri = deploymentserver.splunk.mycompany.com:8089
```

通常情况下，此示例中的所有属性几乎都采用默认值。不过必须将部署服务器位置这一属性值设置为 `deploymentserver.splunk.mycompany.com:8089`。

设置客户端名称

可以为每个部署客户端分配一个客户端名称。部署服务器可以对客户端名称进行过滤，如“设置客户端过滤器”中所述。

默认情况下，客户端名称设置为部署客户端的 GUID。如果要在过滤中使用客户端名称，则建议您将其明确设置为一些合理的可读名称。

重要提示：客户端名称应该是唯一的。

要配置客户端名称，请将 `deploymentclient.conf` 中的属性 `clientName` 设置为所选名称。例如：

```
[deployment-client]
...
clientName = Fflanda-LINUX1
```

重新启动部署客户端以使配置更改生效。

获取部署客户端信息

可从以下两个位置查找部署客户端的相关信息：

- 部署客户端上
- 部署服务器上

从部署客户端查看状态

可以从 Splunk Web 查看部署客户端的状态：

1.单击 Splunk Web 顶部的**设置**链接。随即弹出一个包含指向该组系统界面的链接的窗口。

2.在**系统**部分中选择**服务器设置**。

3.选择**部署客户端设置**。此时将转到一个只读屏幕，其中提供了一些客户端信息：

- 客户端的部署服务器。
- 客户端的服务器类和应用。
- 客户端状态。

从部署服务器查看客户端

配置并重新启动客户端后，将与指定部署服务器启动一个握手过程。部署服务器将此客户端添加到转发器管理界面**客户端**选项卡下的客户端列表中。例如：

禁用部署客户端

要禁用部署客户端，在部署客户端上运行此 CLI 命令：

```
splunk disable deploy-client
```

升级部署客户端

您根据客户端是否是通用转发器或完整 Splunk Enterprise 实例，以正常方式升级客户端。事实上一个实例就是一个部署客户端，在如何执行升级方面没有任何差异。

但是，您升级客户端之后，客户端将在部署服务器维护并通过转发器管理界面显示的客户端清单中显示两次。为消除重复列表，您必须在客户端升级后重新启动部署服务器。

新建部署应用

部署应用包含您要下载到某一组**部署客户端**的任意内容。这些内容可以包括：

- 一个 Splunk Enterprise 应用（例如 Splunkbase 上的应用）
- 一组 Splunk Enterprise 配置
- 其他内容，例如脚本、图像和支持文件

通过在部署服务器上为部署应用新建目录可添加部署应用。新建目录后，即可使用转发器管理界面将应用映射到部署客户端。

可以随时添加或更改应用内容，包括最初新建目录时或后续准备将应用部署或重新部署到部署客户端时。

新建应用目录

您可以在部署服务器上的某个特殊位置为每个部署应用新建单独的目录。默认位置为 `$SPLUNK_HOME/etc/deployment-apps`，但这不能通过 serverclass.conf 中的属性 `repositoryLocation` 进行配置。在此位置下，每个应用都必须拥有各自的子目录。子目录的名称用作转发器管理界面中的应用名称。

注意：下载应用后，应用将驻留在部署客户端上的 `$SPLUNK_HOME/etc/apps` 下。

可以随时添加应用。新建任何新应用目录后，必须运行 CLI `reload deploy-server` 命令以使部署服务器获知这些目录：

```
splunk reload deploy-server
```

即使目录中不包含任何内容，只要新建应用目录，就可以有效地新建应用本身。应用此时显示在转发器管理界面中，您可以使用它来定义服务器类，如“关于服务器类”中所述。

重要提示：指定应用名称时（即新建应用目录时），应注意配置文件优先顺序规则，如《管理员手册》中的主题“配置文件优先顺序”中所述。尤其要注意应用目录优先顺序取决于 ASCII 排序顺序。例如，如果在名称为 "A" 的应用目录的配置文件 x.conf 中设置属性/值对 `whatever=1`，则应用 A 中的设置会覆盖应用 "B" 中 `x.conf` 中的设置 `whatever=0`，依此类推。

填充应用

将应用部署到其客户端之前，可以随时向应用目录中放入内容。稍后可以更新和重新部署应用。要更新应用，只需在目录中添加文件或覆盖其中的文件。有关更新应用和将应用部署到客户端的信息，请参阅“将应用部署到客户端”。

通过转发器管理查看应用

新建应用目录后，部署服务器会将此应用添加到转发器管理界面**应用**选项卡下的应用列表中。例如：

应用管理问题

决定是否使用部署服务器管理应用之前，需要考虑一些事项。

决定使用部署服务器管理应用后不可撤消

重要提示：开始使用部署服务器管理应用后，后续无法停止使用部署服务器管理应用。了解此影响非常重要。

如果从部署服务器的 `repositoryLocation` 中删除应用（如 `serverclass.conf` 所定义），则部署客户端将删除其应用副本。无法通过任何方式指示部署客户端改为自行管理该应用。

例如，假设您要使用部署服务器来管理 "appA" 的更新。为此，在部署服务器上新建了一个名称为 "appA" 的目录，并将应用内容放入其中。之后，部署客户端每次轮询服务器确认是否发生更新时，都会将其 appA 校验和与服务器的 appA 校验和进行比较。如果校验和不同，客户端将从服务器下载最新版本的应用。然后，如果 appA 已从服务器的应用存储库中删除，则客户端也将删除其上的应用实例。

因此，从部署服务器删除应用后，并**不会**指示客户端停止使用部署服务器管理应用并开始自行管理应用。实际上却是指示客户端删除应用。*部署服务器管理某个应用后，它将始终管理该应用。*

警告：由于这一行为的存在，在决定使用部署服务器管理 Splunk Enterprise 搜索应用之前应*格外*谨慎。通过部署服务器管理搜索应用可防止用户在其搜索头上保存任何唯一的搜索。另外，由于无法通过任何方式指示部署服务器停止管理应用，因此可保持这一决定。

带查找表的应用

某些情况下，索引器或搜索头可能运行将信息保存在查找表中的应用。使用部署服务器管理此类应用时应格外注意。部署服务器分发更新的应用配置时，将覆盖现有应用。此时将丢失相关查找表。

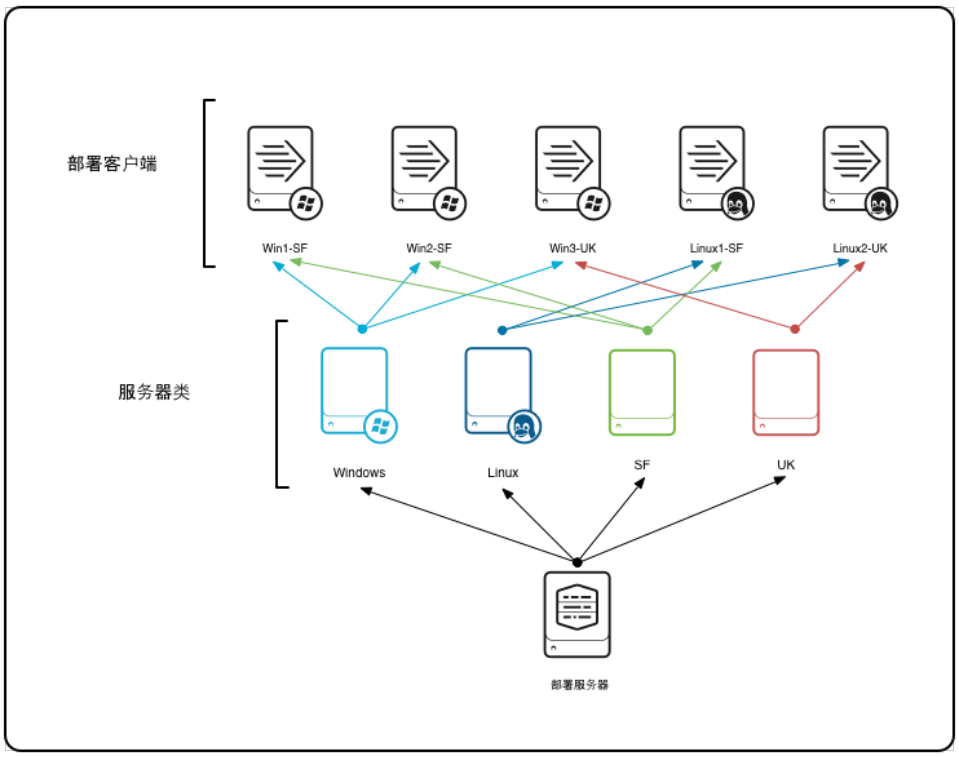
新建服务器类

服务器类可将一组**部署客户端**映射到一个或多个**部署应用**。通过新建服务器类，即可指示**部署服务器**某一组客户端应接收一组应用形式的更新。

客户端分组方法

客户端分组可基于多种条件，例如计算机类型、操作系统、地理区域或应用程序类型。

一个客户端可以属于多个服务器类。例如，某台位于基隆拿、为网站托管团队提供信息的 Windows 转发器可能属于三个服务器类："canada"、"windows" 和 "web_host"。此图显示了客户端如何跨越多个服务器类：



在本例中，每个部署客户端都是一个属于两个服务器类的 Splunk Enterprise 转发器，其中，一个服务器类代表客户端操作系统，另一个服务器类代表客户端地理位置。部署服务器对服务器类列表进行维护，并使用这些服务器类来确定要向每个客户端分发的内容。有关如何通过实施此类排列来管理客户端内容流的示例，请参阅"将配置部署到多个转发器"。

另一种分组客户端的常用方法是定义一个默认应用到所有部署客户端的服务器类。然后为子客户端组定义更多特定的服务器类，根据需要覆盖此服务器类各个方面。例如，如果结合使用 Windows 和 Linux 通用转发器向同一索引器发送数据，则可指定所有转发器采用通用的 `outputs.conf` 文件，但 Windows 转发器采用一个 `inputs.conf` 文件，而 Linux 转发器则采用另一个不同的文件。为此，可定义一个"所有转发器"服务器类以将包含通用 `outputs.conf` 文件的部署应用分发到所有转发器，同时定义 Windows 和 Linux 服务器类以将各个包含不同 `inputs.conf` 文件的应用分发到相应的子转发器组。

服务器类配置概述

服务器类是部署客户端与应用之间的映射。它指示部署服务器将哪些应用发送到哪些客户端。因此，定义服务器类时，可将一个或多个应用与一组部署客户端关联。

在部署服务器上定义服务器类采用的步骤如下：

1.新建服务器类。

2.为该服务器类指定一个或多个部署应用。

3.指定属于该服务器类的客户端。

本部分提供这些步骤的概述。配置过程的具体步骤取决于新建服务器类所采用的方法：转发器管理界面或直接编辑 serverclass.conf。请参阅"服务器类定义方法"。

1.新建服务器类

通过转发器管理界面或直接在 `serverclass.conf` 中新建服务器类并为其命名。

重要提示：服务器类名称必须是唯一的，

2.为服务器类指定应用

新建服务器类后，需要将部署应用与其关联。这些应用为之前在部署服务器的文件系统中所新建的应用，如"新建部署应用"中所述。服务器类与应用之间没有必然的一对一关系。更确实地说是一个服务器类可以包括多个应用。同样，一个应用可以属于多个服务器类。

3.为服务器类指定客户端

接下来需要将客户端与服务器类关联。通常不会单独指定客户端。而是基于一些可识别的共同特性设置过滤器，使其动态决定哪些客户端属于服务器类。

服务器类定义方法

在部署服务器上定义服务器类。服务器类定义保存在部署服务器的 `serverclass.conf` 文件中。有两种方法可用来定义服务器类：

- 使用转发器管理界面。有关详细信息，请参阅"使用转发器管理定义服务器类"。这种定义服务器类的方法非常简单。可以快速交互式过滤客户端并将其映射到应用。
- 直接编辑部署服务器的 `serverclass.conf` 文件。有关详细信息，请参阅"使用 serverclass.conf 定义服务器类"。一些高级配置要求直接编辑 `serverclass.conf`。

重要提示：直接编辑 `serverclass.conf` 后，可能无法使用转发器管理界面执行后续配置操作。这是因为转发器管理可能只能处理配置的子集。有关哪些 `serverclass.conf` 变更与转发器管理相兼容的详细信息，请参阅主题"兼容性和转发器管理"。

serverclass.conf 文件

`serverclass.conf` 文件是关键的部署服务器配置文件。所有服务器类定义都被写入该文件中。此外，其中还保存了大量与部署服务器功能相关的设置。有关包含所有配置属性列表在内的该文件的详细说明，请参阅 serverclass.conf 规范文件。

有关 Splunk Enterprise 配置文件的背景信息，请参阅《管理员手册》中的"关于配置文件"。

使用转发器管理界面配置服务器类时，它会将定义写入 `serverclass.conf` 的副本中。也可以直接编辑 `serverclass.conf`，一些复杂配置中可能需要使用此方法。

部署服务器可以具有多个版本的 `serverclass.conf`。配置文件存在多个版本时，Splunk Enterprise 将按照定义的流程合并所有版本中的属性。有关 Splunk Enterprise 所采用的优先顺序，请参阅《管理员手册》中的"配置文件优先顺序"。

使用转发器管理新建服务器类时，它会将服务器类定义保存在 `$SPLUNK_HOME/etc/system/local` 下的 `serverclass.conf` 副本中。如果不使用转发器管理，而是直接编辑 serverclass.conf，则建议在同一目录 `$SPLUNK_HOME/etc/system/local` 下新建 `serverclass.conf` 文件。

如果服务器类定义位于一些其他目录（最常见的是位于 `$SPLUNK_HOME/etc/apps/<app_name>/local` 的应用目录）下的 `serverclass.conf` 文件中，则转发器管理界面仍将显示该服务器类，以及位于系统中其他位置的文件中的其他任何服务器类定义。如果使用转发器管理编辑现有服务器类，则该界面会将编辑结果保存到同一版本的 `serverclass.conf`。即，如果编辑的是 `$SPLUNK_HOME/etc/apps/SomeApp/local/serverclass.conf` 中定义的服务器类，则转发器管理会将更新后的定义重新保存到该相同目录。不过，如果之后使用转发器管理新建一个新的服务器类，则该界面会将新的服务器类保存到 `$SPLUNK_HOME/etc/system/local/serverclass.conf`。

重要提示：在所有 `serverclass.conf` 文件中，服务器类名称必须是唯一的。

使用转发器管理定义服务器类

转发器管理界面是一个用于新建和编辑**服务器类**的交互式可视工具，服务器类用于将**部署客户端**映射到**部署应用**。服务器类运行于**部署服务器**上。有关该界面的简介（包括详细的访问方式），请参阅"转发器管理概述"。

该界面将服务器类配置保存在 serverclass.conf 文件中。首次保存服务器类时，转发器管理会在部署服务器上的 `$SPLUNK_HOME/etc/system/local` 下新建 `serverclass.conf` 文件。有关 `serverclass.conf` 的信息，请参阅“serverclass.conf 文件”。

注意：对于一些高级的服务器类配置，可能需要直接编辑 `serverclass.conf`。有关转发器管理限制的信息，请参阅“兼容性和转发器管理”。有关直接编辑 `serverclass.conf` 的详细信息，请参阅主题“使用 serverclass.conf 定义服务器类”。

定义服务器类

服务器类将部署客户端映射到部署应用。因此，定义服务器类之前，需要配置客户端并新建要映射的应用。相关信息，请参阅主题“配置部署客户端”和“新建部署应用”。

设置客户端和应用后，它们将自动显示在转发器管理界面中。

服务器类的定义包含三个步骤：

1.新建服务器类。

2.向服务器类添加部署应用。

3.为服务器类指定客户端。

第 2 步和第 3 步可以互换。也可以随时更改客户端和应用组。

重要提示：每次使用转发器管理界面编辑和保存服务器类配置时，部署服务器都将重新加载最新的应用内容并将其部署到尚未收到该内容的客户端。例如，如果向服务器类添加应用，则服务器类中的所有客户端都将收到应用。如果向服务器类添加一些新客户端，但不更改应用，则只有新客户端会收到分发内容。有关哪些操作会提示部署服务器部署或重新部署应用的详细介绍，请参阅主题“将应用部署到客户端”。

1.新建服务器类

要新建服务器类，请转到转发器管理界面（如“访问转发器管理界面”所述）并执行以下操作：

1.选择**服务器类**选项卡。

2.选择**新服务器类**按钮。

3.在弹出窗口的**标题**字段中输入服务器类名称，单击**保存**。

此时将转到一个提示您添加应用和客户端的屏幕。

重要提示：服务器类名称必须是唯一的，而且不能包含空格。

2.添加应用

保存服务器类时，界面将转到一个提示您添加应用和客户端的屏幕。要将应用添加到服务器类，请执行以下操作：

1.单击**添加应用**按钮。此时将转到**编辑应用**页面。（也可以稍后编辑该组应用，具体操作为转到**服务器类**选项卡并单击特定服务器类的**编辑**按钮。）

编辑应用页面上显示有两列：**取消选定的应用**和**选定的应用**，每列下面都有一系列应用。对于新的服务器类，所有应用最初都将位于**取消选定**列中。其中显示的应用为在部署服务器上的存储库位置中具有子目录的应用（默认情况下为 `$SPLUNK_HOME/etc/deployment-apps`），如“新建部署应用”所述。

2.要将应用添加到服务器类，可单击该应用。这样便将应用从**取消选定的应用**列移动到**选定的应用**列。

3.为服务器类选择所有应用后，单击**保存**。如果尚未添加任何客户端，此时将弹出提示要求进行添加。

假定您已经为此服务器类指定客户端，则保存应用设置时，部署服务器将重新加载应用并将其部署到该组客户端。如果尚未指定客户端，则应用会在指定客户端后得到部署。

3.指定客户端

保存服务器类时，界面将转到一个提示您添加应用和客户端的屏幕。要将客户端添加到服务器类，请执行以下操作：

1.单击**添加客户端**按钮。（也可以稍后编辑该组客户端，具体操作为转到**服务器类**选项卡并单击具体服务器类的**编辑**按钮。）随即显示**编辑客户端**页面：

在**编辑客户端**页面上，有**所有**、**匹配**和**不匹配**这三个按钮，按钮下面是客户端列表。选择**所有**按钮时显示的客户端为之前为此部署服务器配置的客户端，如“配置部署客户端”所述。**匹配**和**不匹配**按钮根据客户端是否与您在页面顶部新建的过滤器相匹配来列出相应的客户端。对于新服务器类，最初没有任何客户端显示在**匹配**选项卡下。

2.要将客户端添加到服务器类，不用单独指定各个客户端。而是新建一个动态包括或排除客户端的过滤器。使用页面顶部附近的字段来输入过滤器：**包括**、**排除**和**按计算机类型过滤**。新建客户端过滤器需要了解很多事项，因此主题“设置客户端过滤器”专门介绍了相关详细信息。

输入一个或多个过滤器后，单击**预览**按钮可查看匹配的一组客户端。

如果客户端列表比较长，而您只想查看其中一部分客户端，则可使用**所有/匹配/不匹配**按钮右侧的**过滤器**字段。此特定过滤器只会限制表中显示的客户端；而*不会*过滤服务器类中包括的那组客户端。您可以对客户端表中的各列进行过滤；例如主机名或 IP 地址。

3.指定要包括到服务器类中的客户端后，单击**保存**。如果尚未添加任何应用，此时将弹出提示要求进行添加。

保存客户端设置时，部署服务器将重新加载此服务器类的一组应用，并将其部署到尚未收到这些应用的任何客户端。只有已经指定服务器类的应用时才会执行此操作。

查看服务器类

要查看服务器类的内容，请执行以下操作：

1.转到转发器管理主页上的**服务器类**选项卡。其中显示了部署服务器上的所有服务器类的列表。

2.单击要查看的服务器类。此时将转到该服务器类的页面。该页面包含三个部分：

- **状态。**页面顶部列出了服务器类中的应用和客户端数量。其中也提供了部署成功率信息。
- **应用。**中间部分列出了服务器类的应用及其部署状态。您可以从此处：
  - 通过单击列表顶部的**编辑**链接来编辑服务器类的一组应用。
  - 单击任何应用名称获取与之相关的更多信息。
  - 通过单击应用的**编辑**操作来编辑应用的属性。
- **客户端。**底部部分列出了服务器类的客户端以及某些与之相关的信息，包括其状态。您可以从此处：
  - 通过单击列表顶部的**编辑**链接来编辑服务器类的一组客户端。
  - 按照客户端多久回拨、客户端部署状态或客户端表中的其他字段来过滤列表。
  - 通过单击每行最左侧一列中的箭头获取有关客户端的更多详细信息。

有关管理应用的信息，请参阅“使用转发器管理管理应用”。有关管理客户端的信息，请参阅“使用转发器管理管理客户端”。

编辑服务器类

您可以随时编辑服务器类。转到转发器管理入口页面的**服务器类**选项卡。找出所需的服务器类并单击该行的**编辑**按钮。可选择以下选项：

- **编辑客户端。**编辑该服务器类的一组客户端。请参阅“使用转发器管理管理客户端”。
- **编辑应用。**编辑该服务器类的一组应用。请参阅“使用转发器管理管理应用”。
- **重命名。**重命名该服务器类。
- **删除。**删除该服务器类。

如果在编辑该组客户端或应用后单击**保存**，则部署服务器将重新加载最新的应用内容，并为尚未收到此内容的任何客户端提供此内容。

注意：各个服务器类页面右上角附近的**编辑**按钮也可以执行相同的编辑操作。

设置客户端过滤器

定义服务器类过程中的一项关键活动是指定一组属于该服务器类的客户端。此操作通过客户端过滤器来完成。

过滤器类型

客户端过滤器有三种类型：

- **白名单。**根据 IP 地址、主机名、DNS 名称或客户端名称指定要包括的客户端。
- **黑名单。**根据 IP 地址、主机名、DNS 名称或客户端名称指定要排除的客户端。
- **计算机类型。**根据计算机类型（例如 `linux-i686`, `linux-x86_64` 等）指定要包括的客户端。

注意：“客户端名称”是一个通过 `deploymentclient.conf` 中的属性 `clientName` 分配给部署客户端的逻辑名称。有关更多信息，请参阅“设置客户端名称”。

您可以使用上述过滤器之一或全部来定义服务器类的一组客户端。例如，可以将北美地区所有客户端的白名单与包括所有 Windows 客户端的计算机类型过滤器合并起来。这样，北美 Windows 客户端将位于其自身的服务器类中，并独立于北美 Linux 客户端或其他区域的客户端得到处理。

通过转发器管理定义过滤器

*在转发器管理中，始终针对整个服务器类来定义过滤器。*如果需要在全局级别或应用级别定义过滤器，则必须直接编辑 `serverclass.conf`。

过滤器在指定服务器类的客户端过程中进行定义。客户端的指定过程如“使用转发器管理定义服务器类”所述。该主题中的说明可引导您进入**编辑客户端**仪表板。在该仪表板顶部，有一些独立的字段，即“包括”（白名单）、“排除”（黑名单）和“计算机类型”：

注意：

- 可根据需要为多个过滤器字段输入值。也可以为每个字段输入多个过滤器。
- 要输入多个白名单/黑名单过滤器，可使用逗号分隔每个过滤器。
- 指定 IP 地址或主机名时，可以使用通配符，例如 `10.1.1.*` 或 `fwdr-*`。
- 对于计算机类型过滤器，请使用下拉列表。该列表中填充了部署服务器的客户端组的所有计算机类型。例如，下拉列表可包括这些值：`linux-x86_64`、`linux-i686` 和 `windows-x64`。根据客户端所采用分组方式的不同，每种计算机类型可能需要一个单独的服务器类，或者您也可能能够将多种计算机类型（例如，所有 Linux 计算机类型）合并到单个服务器类中。
- 转发器管理中的计算机类型过滤器相当于 `serverclass.conf` 中的属性 `machineTypesFilter`，而*不*是属性 `machineTypes`。

过滤器合并方式

各种过滤器按如下方式合并：

- 默认情况下，客户端不包括在服务器类中。
- 与任何白名单条目匹配但不与任何黑名单条目匹配的客户端包括在服务器类中。
- 与任何黑名单条目匹配的客户端都包括在服务器类中，无论客户端是否与白名单匹配。
- 计算机类型过滤器作用于白名单/黑名单过滤器的输出上。*如果要使用计算机类型作为唯一的限定符，则还必须在白名单过滤器中加入一个星号。*

以下示例介绍了不同的结果以及为获得这些结果需要输入到过滤器字段中的内容。它们只作示例使用，与客户端主机

名和 DNS 名称的特定约定有关。

示例 1

本示例包括所有 Windows 64 位客户端：

- 白名单：`*`
- 黑名单：
- 计算机类型：`windows-x64`

重要提示：白名单过滤器中必须使用星号。否则计算机类型过滤器没有适用的客户端。

示例 2

本示例包括所有 Windows 64 位转发器：

- 白名单：`fwdr-*`
- 黑名单：
- 计算机类型：`windows-x64`

示例 3

本示例包括北美以外属于公司运营领域的 Linux 客户端：

- 白名单：`*.ops.yourcompany.com`
- 黑名单：`northamerican-*`
- 计算机类型：`linux-i686, linux-x86_64`

涵盖 `serverclass.conf` 的本主题部分包括过滤器的其他信息，这些信息也与转发器管理过滤器配置相关：

- 过滤器行为的详细信息
- 更多示例

通过 serverclass.conf 定义过滤器

当您的过滤需求超出转发器管理界面的功能范围之时，可以直接编辑 `serverclass.conf`，这些场合包括需要在多个级别之间分层过滤器时，或者需要为某个服务器类中的不同应用定义不同的过滤器时。

在 `serverclass.conf` 中，您可以在以下三个级别当中的任一级别定义过滤器：

- **全局。**此类过滤器应用于所有服务器类。
- **服务器类。**此类过滤器应用于某一个完整的服务器类。
- **应用。**此类过滤器只应用于服务器类中的某一个应用。

您可以在不同的级别定义多个过滤器。过滤器越具体，优先级越高。例如，您可以在全局级别设置一个白名单过滤器，然后在各服务器类级别使用不同的黑名单过滤器从各服务器类中排除某些客户端。

编辑 `serverclass.conf` 后，必须重新加载部署服务器，更改才能生效。要重新加载部署服务器，可调用 CLI `reload deploy-server` 命令：

```
splunk reload deploy-server
```

有关详细信息，请参阅“重新加载部署服务器”。

重要提示：直接编辑 `serverclass.conf` 后，您很有可能无法再使用转发器管理界面进行配置操作。有关详细信息，请参阅主题“兼容性和转发器管理”。

确定部署客户端的主机名

过滤器可以作用于部署客户端的主机名。要确定要过滤的主机名的正确集合，您可以在部署服务器上运行 CLI 命令：

```
splunk list deploy-clients
```

过滤属性列表

`serverclass.conf` 中的以下属性决定了过滤器行为：

过滤器属性	用途为何	默认
filterType	设置为**“白名单”**或**“黑名单”**。	whitelist

	这决定过滤器的执行顺序。如果 `filterType` 为 `whitelist`，则首先应用所有白名单过滤器，然后应用黑名单过滤器。如果 `filterType` 为 `blacklist`，则首先应用所有黑名单过滤器，然后应用白名单过滤器。 `whitelist` 设置表示包括某一子集的过滤策略： • 默认情况下项目视为与段落不匹配。 • 与任何白名单条目匹配但不与任何黑名单条目匹配的项目视为与段落匹配。 • 与任何黑名单条目匹配的项目视为与段落不匹配，无论项目是否与白名单匹配。 `blacklist` 设置表示排除某一子集的过滤策略： • 默认情况下项目视为与段落匹配。 • 与任何黑名单条目匹配但不与任何白名单条目匹配的项目视为与段落不匹配。 • 与任何白名单条目匹配的项目视为与段落匹配。 总结： • `whitelist`：默认不匹配 -> 启用白名单 -> 禁用黑名单 • `blacklist`：默认匹配 -> 禁用黑名单 -> 启用白名单 可以在 `serverClass` 和 `serverClass:app` 级覆盖此值。如果在全局级别指定 `whitelist`，然后为一个单独的服务器类指定 blacklist，则该服务器类的设置将变为 `blacklist`，您需要在该服务器类定义中提供其他过滤器来替换所覆盖的过滤器。 **重要提示：**转发器管理界面要求 `filterType` 保留其默认值 `whitelist`。	
whitelist.`<n>` blacklist.`<n>`	`<n>` <n> 是一个未签名的整数。此序列可以以任何值开始，并可以为非连续性。 将该属性设置为 `ipAddress, hostname, DNSname,` 或 `clientName`： • `ipAddress` 为部署客户端的 IP 地址。可使用通配符，例如 `10.1.1.*` • `hostname` 为部署客户端的主机名。可使用通配符，例如 `*.splunk.com` • `DNSname` 为部署客户端的 DNS 名称。可使用通配符，例如 `*.ops.yourcompany.com` • `clientName` 为 `deploymentclient.conf` 中可分配给部署客户端的逻辑名称或标记名称。将客户端与过滤器匹配时，其优先级高于 `ipAddress`、`hostname` 或 `DNSname`。 以下是一些示例：当 `filterType` 为白名单时： ``` whitelist.0=*.fflanda.com blacklist.0=printer.fflanda.com blacklist.1=scanner.fflanda.com ``` 将使 fflanda.com 中除 "printer" 和 "scanner" 以外的所有部署客户端与此服务器类相匹配。 当 `filterType` 为黑名单时： ``` blacklist.0=* whitelist.0=*.web.fflanda.com whitelist.1=*.linux.fflanda.com ``` 只有 "web" 和 "linux" 客户端与服务器类相匹配。其他客户端不与之匹配。 部署服务器在 `<n>` 值序列中第一次遇到中断时将停止评估过滤器。 可以在 `serverClass` 和 `serverClass:app` 级覆盖过滤器。 **重要提示：**覆盖其中一种过滤器（白名单/黑名单）将导致另一种过滤器也被覆盖。例如，如果覆盖白名单，则不会从父项继承黑名单；您必须在段落中提供一个黑名单。	n/a
whitelist.from_pathname	将这些属性设置为指定纯文本文件或包含过滤值的逗号分隔值 (CSV) 文件的 `<pathname>`。	n/a

blacklist.from_pathname	这些属性会指示部署服务器从指定文件中导入 <clientName>、<IP address> 或 <hostname> 列表。 如要从 CSV 文件中导入值，您必须将这些属性与 whitelist|blacklist.select_field、whitelist|blacklist.where_field 或 whitelist|blacklist.where_equals 属性（指定包含所需值的 CSV 文件中的字段）结合使用。 关于所有这些属性的使用以及如何从外部文件中导入筛选值的详细信息，请参阅 serverclass.conf。	
machineTypesFilter	匹配以逗号分隔的列表中的任何计算机类型。 此设置允许您使用部署客户端的硬件类型作为过滤器。此过滤器仅在客户端与白名单/黑名单过滤器匹配时使用。 machineTypesFilter 的值是一个包含计算机类型的以逗号分隔的列表，例如 linux-i686, linux-x86_64 等。每种计算机类型都是一个由硬件平台本身指定的特定字符串。 **注意：**machineTypesFilter 值可以包含通配符，例如：linux-*、windows-* 或 aix-* 此字符串在客户端上的查找方式随平台而变化，但是，如果部署客户端已经连接到部署服务器，则可以在部署服务器上使用此 CLI 命令来确定字符串的值： splunk list deploy-clients 这将为 utsname 返回一个值，可用来指定 machineTypesFilter。 此设置将匹配以逗号分隔的列表中的任何计算机类型。常用计算机类型：linux-x86_64, windows-x64, linux-i686, freebsd-i386, darwin-i386, sunos-sun4u, linux-x86_64, sunos-i86pc, freebsd-amd64.	

有关这些属性的更多详细信息，请参阅《管理员手册》中的 serverclass.conf 规范文件。

示例

以下是几个 serverclass.conf 客户端过滤示例。请注意，由于各种原因，无法在转发器管理中完全复制这些示例。例如，第一个示例中在全局级别定义了一个白名单。而在转发器管理中，所有过滤器都在服务器类级别进行定义。在第二个示例中，其中一个服务器类将 filterType 属性设置为 blacklist。而在转发器管理中，filterType 始终设置为 whitelist。

```
# Example 1
# Match 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 hostnames.

[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]
# whitelist.0=* at the global level ensures that the machineTypesFilter attribute
# invoked later will apply.
whitelist.0=*

[serverClass:WindowsMachineTypes]
machineTypesFilter=windows-*

[serverClass:WindowsMachineTypes:app:WindowsApp]

[serverClass:LinuxMachineTypes]
machineTypesFilter=linux-i686, linux-x86_64

[serverClass:LinuxMachineTypes:app:LinuxApp]

# Example 4
# Blacklist a range of IP addresses, using a regular expression

[global]

[serverClass:ExcludeSomeIPAddresses]
filterType=whitelist
whitelist.0=*
blacklist.0=192.168.1.1[34][0-9]

# Example 5a
# Use (whitelist|blacklist) text file import.

[serverClass:MyApps]
whitelist.from_pathname = etc/system/local/clients.txt

# Example 5b
# 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 5c
# 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 5d
# 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*
```

部署应用

将应用部署到客户端

部署服务器将**部署应用**分发到**客户端**。

以下场合下会对应用进行部署：

- 新建服务器类并将一组客户端映射到一个或多个应用。
- 更改服务器类（例如，更改服务器类的一组应用或客户端）。
- 更改应用内容。
- 当服务器类中加入新客户端。

某些情况下，部署服务器会自动部署应用。另外一些情况下，则需要手动启动部署操作。这在一定程度上取决于您是使用转发器管理还是直接编辑 `serverclass.conf`。

想要获取有关评估部署应用所需时间的帮助，请参阅“评估部署服务器性能”。

将应用部署到新服务器类或更新服务器类中的客户端

新建或更改服务器类后，下一步是将其应用部署到其过滤器限定的客户端。如果通过转发器管理配置服务器类，则会自动执行上述过程。如果通过直接编辑 `serverclass.conf` 来配置服务器类，则必须手动启动部署操作。

使用转发器管理

首次新建服务器类时，即可将一组客户端映射到一组应用。指定客户端过滤器和应用后，部署服务器即自动将应用部署到限定的客户端。此过程如“使用转发器管理定义服务器类”所述。

稍后编辑服务器类后，即更改服务器类的一组应用或客户端过滤器后，部署服务器将重新部署*所有*服务器类。也就是说，如果任何服务器类（不仅仅是刚才编辑的服务器类）中的任何应用内容自上次部署后发生更改，则部署服务器会在此时将最新版本部署到限定客户端。

无论何时使用转发器管理更改配置，都会自动重新加载部署服务器。进而使部署服务器将任何发生更改的应用重新部署到所有服务器类当中。

直接编辑 serverclass.conf

可通过直接编辑 `serverclass.conf` 来新建服务器类。与通过转发器管理进行编辑时不同，直接编辑 `serverclass.conf` 后，部署服务器不会自动部署应用。而是必须通过手动重新加载部署服务器来启动部署操作。

要重新加载，可调用 CLI `reload deploy-server` 命令：

```
splunk reload deploy-server
```

运行 `reload deploy-server` 后，部署服务器将部署*所有*服务器类。也就是说，如果任何服务器类中的任何应用为新应用或自上次部署后发生更改，则部署服务器会将最新版本部署到该服务器类的限定客户端。同样，如果自上次重新加载部署服务器后编辑过客户端过滤器，则部署服务器会确保对于每个服务器类而言，所有当前限定的客户端都具有一组最新的应用。

有关通过直接编辑 `serverclass.conf` 来新建服务器类的信息，请参阅“使用 serverclass.conf 定义服务器类”。

更改应用内容后重新部署应用

更新应用内容后，必须重新加载部署服务器以便部署服务器重新部署应用。

如果使用转发器管理，则还须手动重新加载部署服务器才能立即重新部署应用。不过，如果不手动重新加载部署服务器，应用仍然会在转发器管理中发生“任何”后续配置更改之后重新得到部署。

要重新部署包含更新内容的应用，请执行以下操作：

1.更新部署服务器上相关部署应用目录中的内容。

2.重新加载部署服务器，使部署服务器获知发生更改的内容。

部署服务器随即将应用重新部署到其映射到的所有客户端。

1.更新内容

主题“新建部署应用”介绍了如何在部署服务器上新建应用目录。您可以随时添加或覆盖这些目录中的内容。

2.重新加载部署服务器

编辑应用内容后，必须重新加载部署服务器以便部署服务器获知发生更改的应用。随后部署服务器会将应用重新部署到所映射的一组客户端。

要重新加载部署服务器，可使用 CLI `reload deploy-server` 命令：

```
splunk reload deploy-server
```

该命令将对所有应用进行检查确认是否发生更改，并通知相关客户端。

将应用部署到新客户端

部署客户端首次连接到部署服务器时，部署服务器会自动部署其所限定的任何服务器类的应用。这种情况下不需要重新加载部署服务器。

配置新部署客户端且该客户端具有现有服务器类过滤的计算机类型时即属于这种情况。

从应用更新中排除内容

您可以从应用更新中排除指定文件或目录。如果启用此功能，部署客户端会在第一次下载应用时复制内容，但在以后更新此应用时会忽略该内容。这对于阻止 `/local` 目录中的内容在更新时被去掉尤为有用。

此功能需要部署服务器及其部署客户端运行 6.2 及以上版本。

要使用此功能，在部署服务器上的 `serverclass.conf` 中设置 `excludeFromUpdate` 属性。

例如，假设您想要阻止应用更新覆盖 `my-app` 的 `/local` 目录。假设该应用具有典型的目录结构：

```
my-app/
   default/
   local/
       some-conf.conf
       ...
```

要将 `/local` 目录的内容排除在更新之外，将属性 `excludeFromUpdate` 放置在 `serverclass.conf` 的 `my-app` 段落：

```
[serverClass:my-class:app:my-app]
excludeFromUpdate = $app_root$/local
```

当部署客户端第一次下载该应用时，会复制 `/local` 目录及其内容。在后续的下载中，它会完全忽略该目录。任何已经存在于客户端上 `/local` 目录中的内容仍然存在。同样，任何存在于下载的 `/local` 目录中的新内容会被忽视。

请注意以下事项：

- 您必须使用 `$app_root$` 指定应用根目录。
- 您可排除单个文件或整个目录。
- 您可在以下三个段落级别中的任何一个指定 `excludeFromUpdate`：全局、服务器类或应用。例如，如果您在全局级别指定它，则它会从所有应用中将指定内容排除在外。

查看应用部署状态

可以使用转发器管理界面查看应用分发状态。

转到**应用**选项卡。对于每个应用，都会显示该应用所部署到的客户端数量的相关信息。单击应用转到该应用的详细页面：

仪表板顶部附近的**应用数据大小**字段指定了应用**软件包**的大小。该软件包为包含应用在内的压缩文件。部署服务器通过这种方式打包应用并将其分发给客户端。客户端收到软件包后，将解压缩软件包并在适当的位置安装应用。

也可以使用 CLI 获取有关部署状态的信息。要确认所有客户端都已收到部署，可从部署服务器运行此命令：

```
splunk list deploy-clients
```

这将列出自上次重新启动部署服务器以来联系过部署服务器的所有部署客户端。

管理部署服务器

评估部署服务器性能

本主题可帮助您评估将应用下载到某一组客户端所需的时间。以下关键因素起决定作用：

- 部署服务器规范
- 回拨间隔（即，每个客户端与部署服务器核对更新的频率）
- 客户端数量
- 应用总大小

部署服务器配置

配置部署服务器时，请注意以下事项：

- 如果要部署到 50 个以上的客户端，则必须在专用的 Splunk Enterprise 实例上运行部署服务器。实例不能兼任索引器或搜索头。
- 在有 50 个以上客户端的部署服务器上，不要布置主要是搜索头的分布式管理控制台。
- 不要在有 50 个以上客户端的部署服务器上布置搜索头群集 deployer。
- 如果部署服务器的客户端少于 50 个，则部署服务器可以与索引器或搜索头（包括分布式管理控制台）共存。
- 由于应用下载过程中的 CPU 使用率和内存使用率非常高，因此建议将实例驻留在专门的主机上。

在 Linux 上部署需要的时间

本指南基于采用以下设置完成的测试：

- 部署服务器运行在 RAM 为 12GB 的 12 核专用裸金属 64 位 Linux 系统上。
- 部署在高速 LAN 网络中实施。在延迟较长的网络中，部署时间更长。
- 客户端回拨间隔为 60 秒。
- 一组总大小为 50MB 的应用。我们认为此部署大小相对较大，因为大多数情况下只需一次性部署一组应用，然后进行增量更新。

要评估将 50MB 部署到所有客户端需要的总时长（假设采用上述部署服务器硬件），则可以使用此公式：

```
T = 0.0075 * C + 2
```

其中，T 为以分钟为单位的最长部署时间，C 为部署客户端数量。

例如，将 50MB 的应用部署到 1000 个客户端最多需要 9.5 分钟。

在 Windows 上部署需要的时间

本指南基于采用以下设置完成的测试：

- 部署服务器运行在 RAM 为 12GB 的 8 核专用 64 位 Windows Server 2008 虚拟机上。
- 部署在高速 LAN 网络中实施。在延迟较长的网络中，部署时间更长。
- 客户端回拨间隔为 60 秒。
- 一组总大小为 50MB 的应用。我们认为此部署大小相对较大，因为大多数情况下只需一次性部署一组应用，然后进行增量更新。

要评估将 50MB 部署到所有客户端需要的总时长（假设采用上述部署服务器硬件），则可以使用此公式：

```
T = 0.04 * C + 8
```

其中，T 为以分钟为单位的最长部署时间，C 为部署客户端数量。此公式适用于对多达 3000 个客户端的部署操作。

如果要部署到 2000 个以上的客户端，则将回拨间隔延长为五分钟（300 秒）可以显著提升性能。

使用转发器管理管理应用

转发器管理的主要功能是新建**服务器类**以将**部署应用**映射到**客户端**。此过程如主题“使用转发器管理定义服务器类”所述。

该界面还提供了一些用来管理部署应用的工具。您可以：

- 编辑应用
- 卸载应用
- 查看应用部署状态

编辑应用

要编辑应用，请执行以下操作：

1.转到**应用**选项卡，其中显示了一个包含所有应用在内的列表。

2.找到要编辑的应用并单击其**编辑**操作。

3.选择**编辑**选项。此时将转到该应用的**编辑应用**屏幕。

您可以在此执行一些操作：

- 将应用添加到新的服务器类或将应用从现有服务器类删除。屏幕上部有一个称为**服务器类**的部分。单击 **+** 按钮可将应用添加到新的服务器类。单击服务器类右侧的 **x** 可将应用从该服务器类删除。
- 指定部署客户端下载应用后立即执行的操作。共有两个选项：
  - **启用应用**。这将在客户端上启用应用。
  - **重新启动 Splunkd**。这将在客户端上重新启动 `splunkd`。

完成更改后，单击**保存**按钮。这会使部署服务器保存更改并重新部署该应用（以及过渡期间发生更新的其他任何应用）。有关如何发生应用部署的详细信息，请参阅“将应用部署到客户端”。

重要提示：无法从转发器管理界面编辑应用的实际内容。要更改应用内容，必须通过部署服务器的文件系统更新应用，如“新建部署应用”所述。

卸载应用

您可以从所有服务器类中或仅从一个服务器类中卸载应用。

从所有服务器类中卸载

要从所有服务器类中卸载应用，请执行以下操作：

1.转到**应用**选项卡，其中显示了一个包含所有应用在内的列表。

2.找到要编辑的应用并单击其**编辑**操作。

3.选择**卸载**选项。

此操作会将该应用从所有服务器类中删除并从所有客户端卸载。实际上并*不会*将该应用从部署服务器的文件系统中删除。

从单个服务器类中卸载

要从单个服务器类中卸载应用，请执行以下操作：

1.转到**服务器类**选项卡，其中显示了一个包含所有服务器类在内的列表。

2.找到要从中删除应用的服务器类并单击其**编辑**操作。

3.选择**编辑应用**选项。此时将转到**编辑应用**页面。**编辑应用**页面上显示有两列：**取消选定的应用**和**选定的应用**，每列下面都有一系列应用。

4.在**选定的应用**列中找出要卸载的应用。

5.单击应用名称将该应用从**选定的应用**列移动到**取消选定的应用**列。

6.单击**保存**。

部署服务器随即将应用从该服务器类的所有客户端删除。

查看应用部署状态

要查看应用部署状态，请执行以下操作：

1.转到**应用**选项卡，其中显示了一个包含所有应用在内的列表。

2.找到**客户端**列。该列指示有多少个客户端已经收到该应用。

有关更多信息，请单击应用名称。此时将转到一个屏幕，其中显示了有关应用、应用的客户端及其部署状态的信息汇总。您可以单击每个客户端列左侧的箭头查看与该客户端关联的应用和服务器类。

使用转发器管理管理客户端

您可以使用转发器管理编辑客户端映射和查看客户端状态。

编辑客户端映射

要更改服务器类的客户端映射，请执行以下操作：

1.转到**服务器类**选项卡，其中显示了一个包含所有服务器类在内的列表。

2.找到要更改客户端映射的服务器类并单击其**编辑**操作。

3.选择**编辑客户端**选项。此时将转到**编辑客户端**页面，其中顶部显示这些过滤器字段：**包括**、**排除**和**按计算机类型过滤**。

4.编辑该组过滤器。有关过滤器信息，请参阅“设置客户端过滤器”。

5.单击**保存**。

如果更改过滤器导致客户端从服务器类取消映射，则之前下载的部署应用仍会保留在该客户端上，但无法再通过部署服务器更新。

查看客户端状态

要查看特定客户端的状态，请转到**客户端**选项卡，其中显示了一个包含所有客户端在内的列表。列表顶部是各种用于过滤列表的选项。

列表中提供了每个客户端的相关信息，包括其主机名、客户端名称、IP 地址、计算机类型、为其部署了多少个应用以及上一次与部署服务器联系的时间。单击行中最左侧的箭头可以获取客户端的更多相关信息。

列表中还包含**删除记录**操作。此操作只会从部署服务器中临时删除客户端记录。客户端下次与部署服务器联系时，将重新生成记录。该操作不会以任何方式影响客户端本身。

高级配置

使用 serverclass.conf 定义服务器类

您可以选择通过直接编辑 `serverclass.conf` 配置文件而不是使用转发器管理界面来定义服务器类。由于转发器管理界面只能处理部分可能的配置子集，因此较为高级的配置可能需要您对 `serverclass.conf` 进行编辑。您也可以通过转发器管理开始配置过程，然后改为直接编辑配置文件以处理高级配置问题。

重要提示：如果直接编辑 `serverclass.conf`，后续可能无法重新通过转发器管理界面进行配置。这是因为转发器管理界面只能处理部分通过 `serverclass.conf` 实现的配置。有关哪些变更与转发器管理相兼容的详细信息，请参阅主题“兼容性和转发器管理”。

serverclass.conf 位置

在部署服务器的 `$SPLUNK_HOME/etc/system/local` 中新建 serverclass.conf 文件。如果之前已经通过转发器管理界面定义一个或多个服务器类，则此文件已存在，您只需编辑或附加到此文件。有关 `serverclass.conf` 的信息，请参阅“serverclass.conf 文件”。

可配置的服务器类内容

最重要的设置定义了每个服务器类的一组部署客户端和一组应用。您可以在三个段落级别的任何一个级别设置大多数属性。

段落级别

您可以在全局级别指定设置，也可以为各个服务器类或服务器类中的应用指定设置。有三个级别的**段落**可实现此操作：

段落	含义	范围
`[global]`	全局级别。	在此级别定义的属性与所有服务器类相关。
`[serverClass:<serverClassName>]`	一个单独的服务器类。 可存在多个 `serverClass` 段落，每个段落对应一个服务器类。	在此级别定义的属性仅与服务器类 `<serverClassName>` 相关。 **注意：** `<serverClassName>` 不能包含空格。此外，`<serverClassName>` 在所有 serverclass.conf 文件之中必须是唯一的。
`[serverClass:<serverClassName>:app:<appName>]`	已命名的服务器类中的应用。此级别用于指定服务器类适用的应用。 可存在多个此类段落，每个段落对应服务器类中的一个应用。	`<appname>` 既可以是某一个应用的名称（通常是 `repositoryLocation` 中它的目录名称），也可以是用于指定 `repositoryLocation` 中所有应用的通配符 "*"。 在此级别定义的属性仅与 `<serverClassName>` 中指定的部署应用 `<appName>`（指定 "*" 时则与所有应用）相关。

如果 serverclass.conf 规范文件中没有特别规定，则可以在每个段落级别定义属性。段落中较为具体的属性将覆盖相对不具体的属性。因此，在 `[serverClass:<serverClassName>]` 段落中定义的属性将覆盖在 `[global]` 中定义的同一属性。

客户端过滤属性

最常见的属性是用来配置客户端过滤的属性。有关这些属性的详细信息，请参阅主题“设置客户端过滤器”。

非过滤属性

大多数非过滤属性都极少从其默认值更改为其他值。这些属性有专门的作用：

属性	用途为何	默认
repositoryLocation	将要为此服务器类部署的内容在部署服务器上的存储位置。	`$SPLUNK_HOME/etc/deployment-apps`
stateOnClient	设置为 "enabled"、"disabled" 或 "noop"。此设置指定接收应用的部署客户端应在安装应用后启用还是禁用应用。"noop" 值适用于不需	启用

	要启用的应用；例如，仅包含事件和来源类型的应用。	
restartSplunkWeb	设置为 "true" 或 "false"。决定客户端的 Splunk Web 是否在收到更新后重新启动。	false
restartSplunkd	设置为 "true" 或 "false"。决定客户端的 splunkd 是否在收到更新后重新启动。	false
issueReload	设置为 "true" 或 "false"。决定客户端的 splunkd 是否在收到更新后重新加载。	false

注意：给定配置文件最准确且最完整的设置列表在该配置文件的 .spec 文件中。您可以在《管理员手册》**配置文件参考**中的 serverclass.conf 中或 $SPLUNK_HOME/etc/system/README 中找到 serverclass.conf 的最新版本的 .spec 文件和 .example 文件。

restartSplunkd 和 issueReload 的交互

客户端行为因 restartSplunkd 和 issueReload 的设置而异。选项如下：

issueReload	restartSplunkd	行为
true	false	仅重新加载。不重新启动。可能有必要进行手动重新启动来完全激活下载的应用。
true	true	重新加载客户端。如果某些应用组件需要重新启动来激活，则重新启动客户端。
false	false	下载的应用未激活。
false	true	在应用更新后始终要重新启动客户端。

这些设置在服务器类的基础上可自定义。

示例

有关 serverclass.conf 配置的一些简单示例，请参阅“设置客户端过滤器”。此外，本手册中还将介绍几个内容更多、更为复杂的示例。

重新加载部署服务器

为使更改生效，添加服务器类或更改服务器类配置后，必须重新加载部署服务器。例如，如果向服务器类添加一个应用，则只有在重新加载部署服务器后，部署服务器才会将新应用部署到服务器类客户端。同样，如果更改服务器类的客户端过滤器，则也只有在重新加载后该组客户端（以及任何后续应用部署）的更改才会生效。

要重新加载部署服务器，可调用 CLI reload deploy-server 命令：

```
splunk reload deploy-server
```

有关重新加载部署服务器的更多信息，请参阅主题“将应用部署到客户端”。

兼容性和转发器管理

转发器管理可以处理大多数部署服务器的配置需求。不过，如果您的要求复杂，则可能仍然需要编辑 serverclass.conf。

如果使用一个工具或采用其他方式（转发器管理或直接编辑 serverclass.conf）完成所有配置，则可跳过本主题的其余部分。不过，如果交替使用转发器管理和配置文件，通过这两个界面来执行配置，则可能出现兼容性问题。

有关直接配置 serverclass.conf 的信息，请参阅“使用 serverclass.conf 定义服务器类”。

单向兼容性

转发器管理界面可提供部分通过 serverclass.conf 实现的关键配置功能子集。只需在转发器管理中处理，即可满足大多数配置需求。

如果具有复杂程度一般的要求，则可以先在转发器管理中进行配置，然后切换到 serverclass.conf 执行高级配置。*不过，开始直接编辑配置文件后，后续您极可能无法重新通过转发器管理界面执行配置。*这是因为转发器管理界面仅支持部分通过配置文件实现功能的子集。

如果在编辑 serverclass.conf 后返回到转发器管理，则转发器管理将检测到任何不兼容性并在界面中的适当位置生成错误消息。只要不兼容性存在，您便无法通过转发器管理界面进行配置。

虽然无法继续使用该界面编辑配置，但您仍然可以使用它监视部署。它将正确显示应用、客户端和服务器类之间的映

射。也将正确报告部署指标。

从 6.0 版本之前的部署服务器升级

如果在 6.0 版本之前的 Splunk Enterprise 中（即推出转发器管理界面之前）配置了 `serverclass.conf`，则在升级为 6.0 之后，可能遇到使转发器管理界面无法正常工作的不兼容问题。第一次查看转发器管理界面时，它将检测任何不兼容性并根据需要生成错误消息。只要不兼容性存在，您便无法通过转发器管理进行配置。

如果检测到不兼容性，但您仍希望使用转发器管理界面继续操作，则需要执行以下两项操作之一：

- 删除现有 `serverclass.conf` 文件并重新启动转发器管理界面，然后使用该界面重新新建服务器类和其他设置。
- 直接编辑现有 `serverclass.conf` 文件，以使其严格遵守转发器管理界面的功能要求。这种方法在大多数情况下都可行。如果要执行此操作，请参阅以下部分获取需要考虑的关键不兼容问题的列表。

重要提示：转发器管理能否处理您的配置取决于您的具体需求。

不兼容列表

一些 `serverclass.conf` 属性与转发器管理界面不兼容。此外，一些属性可在配置文件中的多个级别（全局、服务器类和应用）进行设置，而转发器管理中则只能在一个级别设置。

如果您的 `serverclass.conf` 文件包含不兼容的属性，转发器管理界面将进入锁定模式。解决不兼容问题之前，您无法使用该界面进行配置。

此表对 `serverclass.conf` 属性与其在转发器管理中的支持进行了关联。

属性	默认	全局	服务器类	应用
repositoryLocation	$SPLUNK_HOME/etc/deployment-apps	支持	不支持	n/a
targetRepositoryLocation	$SPLUNK_HOME/etc/apps	不支持	n/a	n/a
tmpFolder	$SPLUNK_HOME/var/run/tmp	不支持	n/a	n/a
continueMatching	True	不支持	不支持	n/a
endpoint	$deploymentServerUri$/services/streams/deployment?name=$serverClassName$:$appName$	不支持	不支持	n/a
filterType	whitelist（转发器管理隐式使用此默认值）	不支持	不支持	不支持
whitelist	无	不支持	支持	不支持
blacklist	无	不支持	支持	不支持
whitelist.from_pathname	无	不支持	支持	不支持
blacklist.from_pathname	无	不支持	支持	不支持
machineTypesFilter	无	不支持	支持	不支持

stateOnClient	启用	不支持	所有服务器类之间的单个按应用设置	
restartSplunkd	False	不支持	所有服务器类之间的单个按应用设置	
issueReload	False	不支持	不支持	不支持
restartSplunkWeb	False	不支持	不支持	不支持
appFile	无	n/a	n/a	不支持

关于表中条目的说明：

- **n/a**：表示无法在 `serverclass.conf` 中的该级别设置相关属性。
- **所有服务器类之间的单个按应用设置**：`stateOnClient` 和 `restartSplunkd` 属性在应用级别段落下配置。应用级别段落包括 `appName` 和 `serverClassName` 组件，如下：`[serverClass::app:]`。请参阅“使用 serverclass.conf 定义服务器类”。

  如果您不在使用转发器管理，则可以基于服务器类对于这些属性进行不同的配置，即使对于相同的应用也是如此。例如，在 `[serverClass:X:app:A]` 中您可以指定 `stateOnClient=enabled`，而在 `[serverClass:Y:app:A]`（相同的应用，不同的服务器类）中，您可以指定 `stateOnClient=disabled`。因此，当同样的应用下载到客户端时，根据服务器类其将被启用或禁用。

  但是，转发器管理在其所使用的所有服务器类之中对于每个应用只允许有一个单一的定义。因此，`[serverClass:X:app:A]` 和 `[serverClass:Y:app:A]` 都必须同样地配置 `stateOnClient`。相同的条件也适用于 `restartSplunkd`。

示例

延伸示例：将配置部署到多个转发器

目标

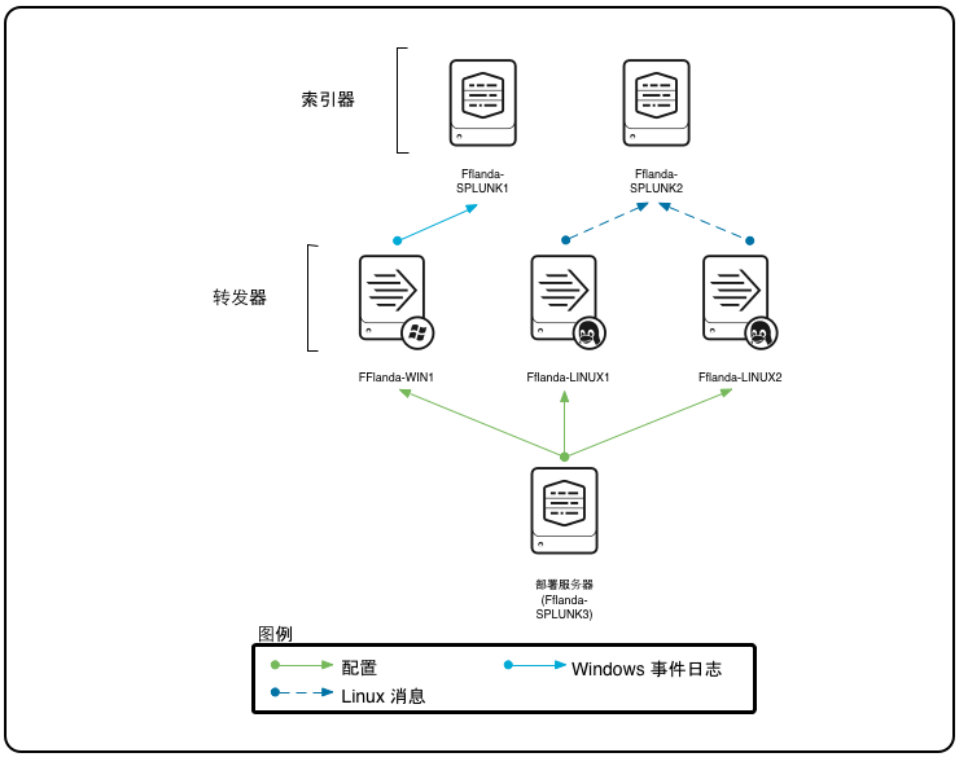
部署服务器的一种常见用途是管理**转发器**的配置文件。在一些分布式环境中，转发器的数量可能达到数千个，这时部署服务器便可以大大简化配置和更新这些转发器的工作。本示例介绍如何使用部署服务器初始配置一组不同的通用转发器。在“示例”中接下来的示例：向转发器添加输入”示例中介绍了如何使用部署服务器将转发器的配置更新为新输入。

该示例设置了以下分布式环境，部署服务器在其中为三个通用转发器部署配置，从而将数据发送到两个**索引器**：

- 部署服务器 Fflanda-SPLUNK3 (10.1.2.4) 用于管理以下通用转发器的部署：
  - Fflanda-WIN1
  - Fflanda-LINUX1
  - Fflanda-LINUX2
- Fflanda-WIN1 将 Windows 事件日志转发到接收索引器 Fflanda-SPLUNK1 (10.1.2.2)。
- Fflanda-LINUX1 和 Fflanda-LINUX2 将 Linux 消息转发到接收索引器 Fflanda-SPLUNK2 (10.1.2.3)。

这些转发器都是**部署客户端**，从部署服务器接收其配置文件。

以下为基本设置：



有关转发器的信息，另请参阅*《转发数据》*中的“关于转发和接收”。有关通用转发器的更多信息，请参阅*《通用转发器》*产品文档中的“关于通用转发器”。

有关监视 Windows 事件日志数据的信息，请参阅*《数据导入》*中的“监视 Windows 事件日志数据”。

有关监视文件的信息（例如消息日志），请参阅*《数据导入》*中的“监视文件和目录”。

配置概述

以下为设置过程概述（详细步骤见下一部分）：

在将要从转发器接收数据的每个索引器上：

1.通过 CLI 启用接收。

2.重新启动索引器。

在每个部署客户端（转发器）上：

1.新建一个指向部署服务器的 `deploymentclient.conf` 文件。

2.重新启动转发器。

在部署服务器上：

1.新建部署应用目录。

2.使用转发器管理为部署客户端（转发器）新建一组**服务器类**。您将新建两个服务器类来表示两种操作系统类型（Windows 和 Linux）。每个服务器类将一组客户端映射到两个不同的应用，共四个应用。这些应用在后续步骤中进行了定义，用于封装：

- 输入类型 - 通用转发器将要监视的数据（Windows 事件日志或 Linux 消息）。
- 输出类型 - 转发器要将数据发送到的索引器（SPLUNK1 或 SPLUNK2）。

此配置将使每个通用转发器属于一个服务器类并接收两个应用：一个对应输入，另一个对应输出。

注意：您也可以直接在 `serverclass.conf` 中新建服务器类，如下文"在 serverclass.conf 中配置服务器类"所述。

3.在应用目录中填入应用内容。每个应用都包含一个配置文件，`outputs.conf` 或 `inputs.conf`。

4.通过重新加载部署服务器来部署应用。

在经过短暂的延时后（转发器在此期间接收并处理其部署内容），Windows 事件日志开始从 Fflanda-WIN1 传送到 Fflanda-SPLUNK1，/var/log/messages 开始从 Fflanda-LINUX1 和 Fflanda-LINUX2 传送到 Fflanda-SPLUNK2。

详细配置步骤

对于每个接收索引器（Fflanda-SPLUNK1 和 Fflanda-SPLUNK2）：

1.在要存放索引器的计算机上安装 Splunk Enterprise。

2.运行以下 CLI 命令：

```
splunk enable listen 9997 -auth <username>:<password>
```

该命令指定索引器将用作接收器，侦听端口 9997 上的数据。授予适当的权限后，此时任何转发器都可以通过指定接收器 IP 地址和端口号来将数据发送到接收器。必须启用索引器作为接收器，*之后才*能将转发器配置为向这些索引器发送数据。

有关启用接收器的更多信息，请参阅《*转发数据*》手册中的"启用接收器"。

3.重新启动索引器。

对于每一个通用转发器（Fflanda-WIN1、Fflanda-LINUX1 和 Fflanda-LINUX2）：

1.如果还没有安装转发器，则在要存放转发器的计算机上安装。

2.指定转发器将要轮询的部署服务器。您既可以在转发器安装过程执行此操作，也可以稍后通过编辑 `$SPLUNK_HOME/etc/system/local/deploymentclient.conf` 来实现，例如：

```
[deployment-client]

[target-broker:deploymentServer]
# Specify the deployment server; for example, "10.1.2.4:8089".
targetUri= <URI:port>
```

有关如何配置此文件的详细信息，请参阅"配置部署客户端"。

3.重新启动转发器。

在部署服务器上 (Fflanda-SPLUNK3)：

A. 准备部署服务器

1.如果还没有安装，则安装 Splunk Enterprise。

2.新建以下部署应用目录：

- `$SPLUNK_HOME/etc/deployment-apps/fwd_to_splunk1`

- $SPLUNK_HOME/etc/deployment-apps/fwd_to_splunk2
- $SPLUNK_HOME/etc/deployment-apps/winevt
- $SPLUNK_HOME/etc/deployment-apps/linmess

目录名称决定了应用名称："fwd_to_splunk1"、"fwd_to_splunk2"，依此类推。

B. 配置 Windows 转发器的服务器类

注意：此程序以及接下来的程序都使用转发器管理来配置服务器类。有关如何配置这些服务器类而不是直接编辑 serverclass.conf 的示例，请参阅本主题后面介绍的"在 serverclass.conf 中配置服务器类"。

要配置 Windows 转发器的 Fflanda-WIN 服务器类，请转到转发器管理界面并执行以下操作：

1.选择**服务器类**选项卡。

2.选择**新服务器类**按钮。

3.在弹出窗口的**标题**字段中输入 Fflanda-WIN 并单击**保存**。此时将转到一个提示您添加应用和客户端的屏幕。

4.单击**添加应用**按钮。此时将转到**编辑应用**页面。此页面上包含两列：**取消选定的应用**和**选定的应用**，每列下面都有一系列应用。对于新的服务器类，所有应用最初都将位于**取消选定**列中。

5.在**取消选定的应用**列中，找到应用 winevt 和 fwd_to_splunk1。单击各个应用将其移动到**选定的应用**列。

6.单击**保存**将应用保存在服务器类中。此时系统将提示您向服务器类添加客户端。单击**添加客户端**按钮。此时将转到**编辑客户端**页面。

7.要将过滤器配置为包括所有 Windows 客户端，可在**包括**字段中输入 Fflanda-WIN*，然后单击**保存**。

8.设置应用的部署后行为。对于每个应用，转到**应用**选项卡，找到应用所在行并单击**编辑**操作编辑应用。在页面顶部附近的**安装后**部分中，选择**启用应用**和**重新启动 Splunkd**。

C. 配置 Linux 转发器的服务器类

要配置 Linux 转发器的 Fflanda-LINUX 服务器类，请转到转发器管理界面并执行以下操作：

1.选择**服务器类**选项卡。

2.选择**新服务器类**按钮。

3.在弹出窗口的**标题**字段中输入 Fflanda-LINUX 并单击**保存**。此时将转到一个提示您添加应用和客户端的屏幕。

4.单击**添加应用**按钮。

5.在**取消选定的应用**列中，找到应用 linmess 和 fwd_to_splunk2。单击各个应用将其移动到**选定的应用**列。

6.单击**保存**将应用保存在服务器类中。此时系统将提示您向服务器类添加客户端。单击**添加客户端**按钮。此时将转到**编辑客户端**页面。

7.要将过滤器配置为包括所有 Linux 客户端，可在**包括**字段中输入 Fflanda-LINUX*，然后单击**保存**。

8.设置应用的部署后行为。对于每个应用，转到**应用**选项卡，找到应用所在行并单击**编辑**操作编辑应用。在页面顶部附近的**安装后**部分中，选择**启用应用**和**重新启动 Splunkd**。

有关转发器管理界面的详细信息，请参阅"使用转发器管理定义服务器类"。

D. 填充应用

1.新建采用以下设置的 $SPLUNK_HOME/etc/deployment-apps/fwd_to_splunk1/default/outputs.conf：

```
[tcpout]
defaultGroup=splunk1

[tcpout:splunk1]
# Specifies the server that receives data from the forwarder.
server=10.1.2.2:9997
```

有关 outputs.conf 的信息，请参阅《*转发数据*》手册中的"使用 outputs.conf 配置转发器"。

2.新建采用以下设置的 $SPLUNK_HOME/etc/deployment-apps/fwd_to_splunk2/default/outputs.conf：

```
[tcpout]
defaultGroup=splunk2
```

```
[tcpout:splunk2]
server=10.1.2.3:9997
```

3.新建采用以下设置的 `$SPLUNK_HOME/etc/deployment-apps/winevt/default/inputs.conf`：

```
[WinEventLog:Application]
disabled=0

[WinEventLog:Security]
disabled=0

[WinEventLog:System]
disabled=0
```

有关监视 Windows 事件日志数据的信息，请参阅《数据导入》手册中的"监视 Windows 事件日志数据"。

4.新建采用以下设置的 `$SPLUNK_HOME/etc/deployment-apps/linmess/default/inputs.conf`：

```
[monitor:///var/log/messages]
disabled=false
sourcetype=syslog
```

有关监视文件的信息（例如消息日志），请参阅《数据导入》手册中的"监视文件和目录"。

E. 部署应用

要部署应用，只需重新加载部署服务器：

```
splunk reload deploy-server
```

此时每个转发器都会轮询部署服务器，下载其配置文件，重新启动并向其接收索引器转发数据。

注意：更改应用内容后，请务必调用 `reload` 命令，这样客户端才了解需要下载新内容。

有关接下来介绍如何使用部署服务器更新转发器配置的示例，请参阅"示例：向转发器添加输入"。

使用 serverclass.conf 配置服务器类

您可以通过直接编辑 `serverclass.conf` 而不是使用转发器管理来配置服务器类。不过，在大多数情况下，使用转发器管理可以更加轻松快速地完成配置任务。

在所述示例中，您可以完全使用转发器管理定义所需的服务器类，因此不必编辑 `serverclass.conf`。不过，对于倾向于直接使用配置文件的用户而言，本子主题介绍了如何通过编辑 `serverclass.conf` 来为示例新建一组服务器类。此版本利用了只能通过直接编辑 `serverclass.conf` 来提供的部分辅助优化功能。

新建采用以下设置的 `$SPLUNK_HOME/etc/system/local/serverclass.conf`：

```
# Global server class
[global]
# Filter (whitelist) all clients
whitelist.0=*

# Server class for Windows
[serverClass:Fflanda-WIN]
# Filter (whitelist) all Windows clients
whitelist.0=Fflanda-WIN*

# App for inputting Windows event logs
# This app is only for clients in the server class Fflanda-WIN
[serverClass:Fflanda-WIN:app:winevt]
#Enable the app and restart Splunk, after the client receives the app
stateOnClient=enabled
restartSplunkd=true

# App for forwarding to SPLUNK1
# This app is only for clients in the server class Fflanda-WIN
[serverClass:Fflanda-WIN:app:fwd_to_splunk1]
stateOnClient=enabled
restartSplunkd=true
```

```
# Server class for Linux
[serverClass:Fflanda-LINUX]
# Filter (whitelist) all Linux clients
whitelist.0=Fflanda-LINUX*

# App for inputting Linux messages
# This app is only for clients in the server class Fflanda-LINUX
[serverClass:Fflanda-LINUX:app:linmess]
stateOnClient=enabled
restartSplunkd=true

# App for forwarding to SPLUNK2
# This app is only for clients in the server class Fflanda-LINUX
[serverClass:Fflanda-LINUX:app:fwd_to_splunk2]
stateOnClient=enabled
restartSplunkd=true
```

有关如何配置此文件的详细信息，请参阅“使用 serverclass.conf 定义服务器类”。

部署服务器与其客户端之间如何通信

以上述示例为例，端口 8089 上从 Fflanda-WIN1 到 Fflanda-SPLUNK3 的通信如下：

Fflanda-WIN1：你好，我是 Fflanda-WIN1。

Fflanda-SPLUNK3：你好，Fflanda-WIN1。我一直在等你的消息。我邀请你加入 Fflanda-WIN 服务器类，你应该有 fwd_to_splunk1（校验和 = 12345）和 winevt（校验和 = 12378）应用吧。

Fflanda-WIN1：呃，我没有这些配置。我刚使用这个连接向你开放了访问权限，我能从你这获取这些配置吗？

Fflanda-SPLUNK3：当然可以！我都为你准备好了。

Fflanda-WIN1：谢谢！暂时离开，请稍候，时间不超过 60 秒钟（当有大量客户端在轮询时）.....好了，现在发给我文件吧。

Fflanda-SPLUNK3：发送完毕！现在你应该收到 fwd_to_splunk1-timestamp.bundle 和 winevt-timestamp.bundle 了。

Fflanda-WIN1：好极了！我马上将它们存储在我的 $SPLUNK_HOME/etc/apps 目录下。现在我要重新启动了，重新开启后我会直接从 .bundle 文件读取你发送给我的配置，我知道这些只是扩展名不同的 tar 文件。

几分钟过后....

Fflanda-WIN1：你好，我是 Fflanda-WIN1。

Fflanda-SPLUNK3：你好，Fflanda-WIN1。我一直在等你的消息。我邀请你加入 Fflanda-WIN 服务器类，你应该有 fwd_to_splunk1（校验和 = 12345）和 winevt（校验和 = 12378）应用吧。

Fflanda-WIN1：呃，我已经有这两个应用了，不过还是表示感谢！

之后，管理员修改了 Fflanda-SPLUNK3 上的 `winevt/inputs.conf` 文件以禁用系统事件日志的集合，然后运行 CLI 命令 `splunk reload deploy-server` 强制部署服务器重新扫描 serverclass.conf 和应用目录。Fflanda-WIN1 与 Fflanda-SPLUNK3 之间的下一段对话如下：

Fflanda-WIN1：你好，我是 Fflanda-WIN1。

Fflanda-SPLUNK3：你好，Fflanda-WIN1。我一直在等你的消息。我邀请你加入 Fflanda-WIN 服务器类，你应该有 fwd_to_splunk1（校验和 = 12345）和 winevt（校验和 = 13299）应用吧。

Fflanda-WIN1：呃，我有这些配置，不过我手头的 winevt 配置检验和同你刚才告诉我的不一样。我刚使用这个连接向你开放了访问权限，我能从你这获取更新后的 winevt 配置吗？

Fflanda-SPLUNK3：当然可以！我都为你准备好了。

Fflanda-WIN1：谢谢！暂时离开，请稍候，时间不超过 60 秒钟（当有大量客户端在轮询时）.....好了，现在发给我更新配置吧。

Fflanda-SPLUNK3：发送完毕！你现在应该收到 winevt-newer_timestamp.bundle 了。

Fflanda-WIN1：好极了！我马上将它存储在我的 $SPLUNK_HOME/etc/apps 目录中并移除原来的 winevt.bundle。现在我要重新启动了，重新开启后我会拥有最新的配置。

示例：向转发器添加输入

上一主题“延伸示例：将配置部署到多个转发器”介绍了设置一个分布式环境来管理一组通用转发器。其中介绍了如何通过配置新部署服务器来将内容部署到一组新的部署客户端。本例以此作为切入点，继续使用该主题中新建的配置。其中介绍如何更新转发器配置文件，以及如何将更新后的文件部署到服务器类所定义的转发器子集。

更新过程概述

本例以“延伸示例：将配置部署到多个转发器”。Linux 通用转发器现在需要开始监视从另一个来源发出的数据。要完成此任务，请在部署服务器上执行以下步骤：

1.编辑 Linux 服务器类的 `inputs.conf` 文件以添加新来源，覆盖其应用目录中原有的版本。

2.重新加载部署服务器，以便服务器获知更改并将更改部署到一组适当的客户端（转发器）。

只能在部署服务器上执行更改。当 Linux 服务器类中的部署客户端下次轮询服务器时，将获知更改后的 `inputs.conf` 文件。客户端将下载此文件并将其启用，重新启动 `splunkd`，然后立即开始监视另一个数据源。

详细配置步骤

在部署服务器上：

1.编辑 `$SPLUNK_HOME/etc/deployment-apps/linmess/default/inputs.conf` 以添加新输入：

```
[monitor:///var/log/messages]
disabled=false
sourcetype=syslog

[monitor:///var/log/httpd]
disabled=false
sourcetype = access_common
```

2.重新加载部署服务器：

```
splunk reload deploy-server
```

运行此命令后，部署服务器将通知属于 Fflanda-LINUX 服务器类成员的客户端有关更改文件的事项。客户端将下载此文件并将其启用，重新启动 `splunkd`，然后立即开始监视另一个数据源。