

LoadRunner Professional

软件版本：2023

安装指南

转到联机帮助中心
<https://admhelp.microfocus.com/lr/>



文档发布日期: 2023 四月

向我们发送反馈



请告知我们如何改善安装指南的用户体验。

将电子邮件发送至: docteam@microfocus.com

法律声明

© 版权所有 1994-2023 Micro Focus 或其附属公司之一。

担保

Micro Focus 及其附属公司和许可方 (“Micro Focus”) 的产品和服务的唯一担保已在此类产品和服务随附的明示担保声明中提出。此处的任何内容均不构成额外担保。Micro Focus 不会为此处出现的技术或编辑错误或遗漏承担任何责任。此处所含信息如有更改，恕不另行通知。

受限权限声明

包含机密信息。除非另有特别说明，否则必须具备有效许可证方可拥有、使用或复制。按照 FAR 12.211 和 12.212，并根据供应商的标准商业许可的规定，商业计算机软件、计算机软件文档与商品技术数据授权给美国政府使用。

免责声明

此处可访问的某些版本的软件和/或文档(以下称为“资料”)可能包含 Hewlett-Packard Company(现为 HP Inc.)和 Hewlett Packard Enterprise Company 的品牌。从 2017 年 9 月 1 日起，该材料现在由 Micro Focus (一家具有独立所有权和经营权的公司) 提供。对 HP 和 Hewlett Packard Enterprise/HPE 标志的任何引用都是历史性的，HP 和 Hewlett Packard Enterprise/HPE 标志是其各自所有者的财产。

目录

欢迎使用 LoadRunner Professional	5
LoadRunner 帮助中心	5
LoadRunner Professional 组件	5
系统要求和先决条件	7
系统要求	7
在 Windows 上安装必备软件	7
由安装程序自动检测和安装	7
手动安装 Windows 更新	8
安装注意事项和限制	8
在 Windows 上安装 LoadRunner Professional	11
安装工作流	11
升级 LoadRunner Professional	14
在 Windows 上安装 LoadRunner Professional	14
安装向导信息	14
运行安装	15
静默安装 LoadRunner Professional	18
安装命令行选项	21
通过包装文件安装完整的 LoadRunner Professional 程序包	21
通过 PFTW 文件安装独立应用程序或其他组件	23
在 Windows 上部署 Docker 化 Load Generator	24
先决条件	25
使用预定义映像运行通过 Docker 设置的 Load Generator	25
使用自定义映像运行 Docker 化 Load Generator	26
Dockerfile 的自定义内容示例	27
运行 Load Generator 容器后	28
提示和准则	29
配置用户登录设置	29
安装 LoadRunner 语言包	30
在 Linux 上安装 Load Generator	31
安装工作流	31
通过安装向导在 Linux 上安装 Load Generator	33
开始之前	34
运行安装向导	34
Linux 上的 Load Generator 静默安装	35
在 Linux 上部署 Docker 化 Load Generator	36
先决条件	36

使用预定义映像运行通过 Docker 设置的 Load Generator	37
使用自定义映像运行 Docker 化 Load Generator	38
运行 Load Generator 容器后	40
提示和准则	40
配置 Linux 环境	41
设置环境变量	41
验证 Linux 安装	42
运行 verify_generator	43
检查 Controller 连接	44
改善 Load Generator 性能	45
增加文件描述符	46
增加进程条目数	47
增加交换空间	47
从 Linux 计算机上卸载 Load Generator	48
卸载 Load Generator 12.55 或更低版本	48
卸载 Load Generator 12.56 或更高版本	49
Load Generator Linux 安装疑难解答	49
安装 LoadRunner Data Hub	56
安装工作流	56
使用安装向导安装 Data Hub	57
开始之前	57
运行安装向导	57
静默 Data Hub 安装	58
运行静默安装	58
静默安装选项	60
验证 Data Hub 安装	60
安装 Data Hub Web Connector	61
安装工作流	61
使用安装向导安装 Web Connector	62
开始之前	62
运行安装向导	62
静默 Web Connector 安装	63
运行静默安装	63
静默安装选项	65
验证 Web Connector 安装	65

欢迎使用 LoadRunner Professional

欢迎阅读《LoadRunner Professional 安装指南》。

LoadRunner Professional 是一个性能测试工具，通过将应用程序的负载增加到极限来找出并确定客户端、网络和服务器的潜在瓶颈。

本指南介绍如何安装和设置 LoadRunner Professional。

LoadRunner 帮助中心

要获得 LoadRunner Professional 产品内的上下文相关帮助，请在对话框内按 F1 或使用“帮助”菜单。

您可以访问 Web 上的产品帮助中心：

- <https://admhelp.microfocus.com/lr/>
- <https://admhelp.microfocus.com/vugen/>

此外，您还可以从“下载帮助中心”页面下载并安装本地版本的帮助。

要在联机模式和本地模式之间切换，请选择 **帮助 > 帮助中心选项 > 联机打开或本地打开**。

注：

- 要在 Internet Explorer 或 VuGen 中查看帮助中心，需要在浏览器设置中启用 JavaScript(活动脚本)，请依次选择 **工具 > 选项 > 安全 > Internet > 自定义级别** 进行设置。
- 如果联机帮助中心在启动 LoadRunner 应用程序会话开始时不可用，则在与该应用程序的整个会话期间，**联机打开选项** 会被禁用。

LoadRunner Professional 组件

LoadRunner Professional 安装包括以下组件。有关组件的详细信息，请参阅 [LoadRunner Professional 帮助中心](#) (选择相关版本)：

- **Virtual User Generator (VuGen)**。用于主要通过录制来创建虚拟用户 (Vuser) 脚本的 Micro Focus 工具。Vuser 脚本通过直接调用函数来模拟用户，无需图形用户界面。
- **Controller**。控制场景和 Vuser 的执行。包括监控并显示测试执行信息的联机监控器。必须将 Controller 安装在用于控制 Vuser 的计算机上。
- **Analysis**。用于分析负载测试的图和报告。
- **Load Generator**。该组件用于运行 Vuser(包括基于 Windows 的 GUI Vuser)以生成负载。可以使用安装程序或通过 Docker 在 Windows 或 Linux 平台上安装 Load Generator。
- **Data Hub**。支持 LoadRunner 组件之间更可靠通信的通信通道。
- **Data Hub Web Connector**。使 Data Hub 能够通过 Web 和防火墙进行通信。
- **MI Listener 组件**。MI Listener 计算机上的组件，用于运行 Vuser 并通过防火墙进行监控。
- **Monitors over Firewall**。代理计算机上的组件，用于通过防火墙进行监控。
- **TruClient**。该组件用于录制和开发基于 Web 的应用程序的测试脚本。

LoadRunner Professional“完全安装”安装包包括上述大部分组件。所有独立安装(如 VuGen 或 Load Generator 安装)都可以在安装包的 **Standalone Applications** 文件夹中找到。

其他组件(如 Citrix 代理)可以在安装程序包的“其他组件”文件夹中找到。

您可以从[免费试用](#)页面或通过在[软件许可和下载](#)页面上选择您的帐户来访问安装包。

此外，您还可以从[AppDelivery Marketplace](#)免费下载 VuGen、LoadRunner 开发人员、VTS、Data Hub 等。

系统要求和先决条件

本章包括：

- [系统要求](#) 7
- [在 Windows 上安装必备软件](#) 7
- [安装注意事项和限制](#) 8

系统要求

[支持列表](#)中列出了在 Windows 系统上运行 LoadRunner Professional 或在 Windows 或 Linux 系统上运行 Load Generator 所需的系统要求。

有关 Network Virtualization 的系统要求和安装说明，请参阅 [适用于 LoadRunner 的 Network Virtualization 帮助](#) (选择相关版本)。您可以在安装 LoadRunner Professional 期间或稍后安装 Network Virtualization。

在 Windows 上安装必备软件

在安装 LoadRunner Professional 前，请确保在 LoadRunner Professional 计算机上安装了必备软件。某些必备软件由安装程序自动检测并安装，还有一些必备软件必须手动安装。

- [由安装程序自动检测和安装 \(第 7 页\)](#)
- [手动安装 Windows 更新 \(第 8 页\)](#)

由安装程序自动检测和安装

运行安装向导时，向导会检测已在计算机上安装的必备软件以及缺少的软件。如果继续安装，将安装所有缺少的必备软件。

必须安装以下必备软件：

- .NET Framework 4.8
- Microsoft Visual C++ Redistributable for Visual Studio 2017-2022 (x86 和 x64)

注: 安装 .NET Framework 4.8 会提示自动重新启动。然后，您需要重新运行 LoadRunner Professional 安装。

手动安装 Windows 更新

在安装任何 LoadRunner Professional 组件之前，请确保已安装整套 Windows 更新。如果该过程未自动完成，则必须手动安装更新。

注: 在安装任何 Windows 更新之前，请禁用 UAC(用户帐户控制)并重新启动计算机。有关如何禁用 UAC 的详细信息，请参阅 Microsoft Windows 文档。

可以在 [支持列表](#) 中找到 Windows 更新列表。

安装注意事项和限制

注: 有关其他注意事项和限制，请参阅 VuGen 或 LoadRunner Professional 帮助中心中的“已知问题”。

安全检查	我们强烈建议您在不包含敏感信息或不提供敏感信息访问的专用计算机上安装 LoadRunner Professional 组件，并且强烈建议您对测试环境中的网络拓扑和访问级别进行彻底的安全检查。
权限要求	<p>对于任何操作系统，所有 LoadRunner Professional 组件的安装过程都需要使用具有完全权限的管理员帐户 (对于 Linux 系统，则为 root 帐户)。</p> <p>您可以在启用 UAC 和 DEP 的情况下安装 LoadRunner Professional 组件，但建议在安装过程中将其禁用 (或仅为必需的 Windows 程序和服务启用 DEP)。</p> <p>安装后，在启用 UAC 和 DEP 的情况下，所有 LoadRunner Professional 应用程序和组件都能使用标准用户帐户运行 (不需要管理员帐户)。</p>

在 Windows 上运行 Web Services Vuser	<p>Axis 工具包和 WSE 不再随 LoadRunner Professional 自动安装。在独立 Load Generator 上运行 Web Services Vuser 时，某些传统配置需要激活 .NET 3.5 (请参阅 MSDN)，需要包括 Axis java 文件，并需要安装 WSE (2.0 SP3 和 WSE 3.0)。</p> <p>为了避免使用这些组件，可以使用默认的 WCF 工具包重新创建脚本并导入 WSDL。</p> <p>要使用这些组件，请执行以下操作：</p> <ul style="list-style-type: none"> 对于 WSE，从 OneLG 安装程序包的文件夹 DVD\lrunner\Common\wse20sp3 和 DVD\lrunner\Common\wse30 中安装 WSE 组件，或者从 Microsoft 网站下载这些组件。 对于 Axis，将 Axis java 文件复制到 %LG_PATH%\bin\java\Axis。对于 OneLG，这些文件位于解压缩安装程序包的 prerequisites\Axis 子文件夹中，默认情况下为 C:\Temp\Micro Focus LoadRunner OneLG <版本>\prerequisites\Axis。这样您就可以在 Axis 工具包上运行传统脚本。 <p>有关更多详细信息，请参阅 Web 服务的已知问题。</p>
在 Windows 上运行 Java Vuser	<p>32 位 OpenJDK 不再随 LoadRunner 自动安装。要使用 Java 协议，您需要手动安装受支持的 JDK 版本。VuGen 使用内部逻辑自动检测您的 JRE 版本。有关详细信息，请参阅 Java 协议设置环境。</p>
LoadRunner Professional 与 UFT 共存	<p>如果将 LoadRunner Professional 和 UFT 安装在同一服务器上，然后卸载其中某个软件程序，则另一个软件程序可能无法正常运行。</p> <p>解决方案： 在计算机上运行要使用的软件程序的修复。</p>
许可证	<p>如果从早于 2020 的版本升级，您需要替换当前许可证才能使用升级后的 LoadRunner Professional 安装。如果您使用的是社区许可证，则在安装 LoadRunner Professional 时会自动替换它。对于任何其他许可证，请联系许可支持以获得有效的兼容许可证。</p> <p>有关详细信息，请参阅 LoadRunner Professional 帮助中心中有关 LoadRunner Professional 许可证的部分。</p>
自定义证书	<p>在从版本 12.55 或更低版本进行升级时，如果在之前安装 LoadRunner 时使用了自定义证书，则在安装期间提示需要证书时再次安装这些证书。否则，安装程序将使用默认证书覆盖这些证书。</p>

Network Virtualization	<p>如果选择自动安装 NV (作为 LoadRunner Professional 安装的一部分), 则必须先禁用 Windows SmartScreen, 然后再继续 NV 安装:</p> <p>在注册表编辑器中打开 HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer, 然后将 SmartScreenEnabled 的“数值数据”更改为 Off。</p> <p>注:手动安装 NV 时, 无需禁用 SmartScreen。</p>
TruClient 版本 冲突	<p>升级 LoadRunner Professional 安装后, 如果遇到任何版本冲突, 请检查原有的 TruClient 注册表项 TcWebIELauncher.exe 是否已被移除。</p> <ul style="list-style-type: none">• 在注册表编辑器中, 打开 HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Internet Explorer\MAIN\FeatureControl\FEATURE_BROWSER_EMULATION 并删除该项。• 或者, 运行命令:
<pre>"REG DELETE "HKEY_LOCAL_ MACHINE\SOFTWARE\Microsoft\Internet Explorer\MAIN\FeatureControl\FEATURE_BROWSER_ EMULATION" /v "TcWebIELauncher.exe" /f"</pre>	
虚拟环境	<p>虚拟化供应商提供的体系结构正在迅速发展。只要第三方供应商保证虚拟化环境完全符合 LoadRunner Professional 核准的硬件要求, LoadRunner Professional 就可以在这些不断变化的环境中按预期设计运行。如果按照系统要求和支持列表创建虚拟机, LoadRunner Professional 将正常工作。</p> <p>在虚拟机上工作可能需要访问虚拟化服务器硬件/监控环境, 以确保虚拟化服务器未饱和; 否则, 这可能会影响虚拟机的度量并导致错误的结果。</p>

在 Windows 上安装 LoadRunner Professional

本章介绍如何在 Windows 平台上安装完整版本的 LoadRunner Professional 或 LoadRunner Professional 组件。

本章包括：

- [安装工作流](#) 11
- [升级 LoadRunner Professional](#) 14
- [在 Windows 上安装 LoadRunner Professional](#) 14
- [静默安装 LoadRunner Professional](#) 18
- [安装命令行选项](#) 21
- [在 Windows 上部署 Docker 化 Load Generator](#) 24
- [配置用户登录设置](#) 29
- [安装 LoadRunner 语言包](#) 30

安装工作流

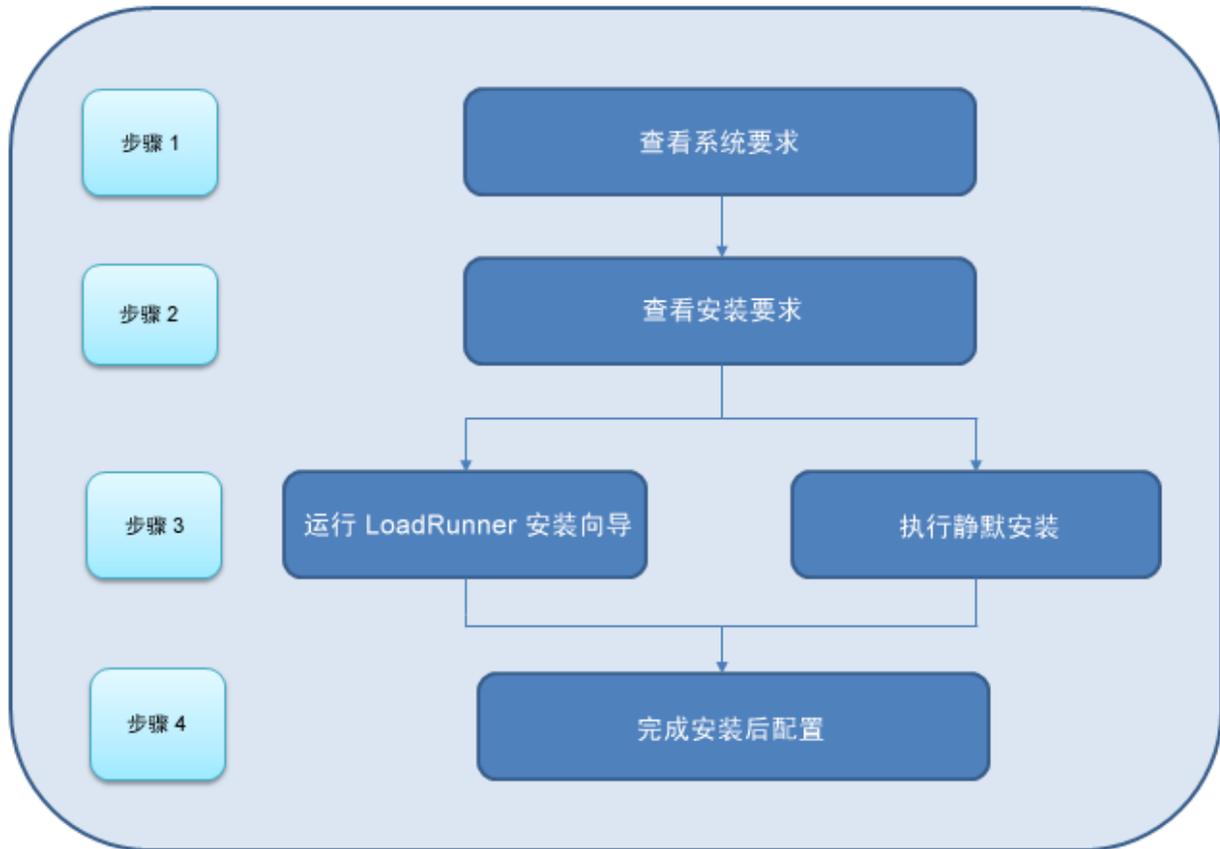
LoadRunner Professional 安装程序包中含有安装向导，用于指导您完成 LoadRunner Professional 组件的安装过程。

安装组件中包含 **OneLG**，它是适用于所有 LoadRunner 系列产品的组合 Load Generator 安装程序。您可以通过安装程序 (基于 UI 或静默安装) 或 Docker 容器安装 Load Generator。有关详细信息，请参阅[在 Windows 上部署 Docker 化 Load Generator \(第 24 页\)](#)。

注：您还可以在云帐户中配置 Load Generator。有关详细信息，请参阅[LoadRunner Professional 帮助中心](#) (选择相关版本) 中有关云 Load Generator 的部分。

安装的最后阶段包括选择是否安装 Network Virtualization。通过安装 Network Virtualization，您可以从 LoadRunner Professional 中生成 NV Insights 报告 (替换 NV Analytics 报告)。NV Insights 报告可深入分析应用程序在各种类型的网络 (虚拟位置) 上对不同负载的执行情况，并包含优化应用程序性能的建议。

在 Windows 系统上安装 LoadRunner Professional 涉及的步骤如下所示：



1. 查看系统要求和注意事项

- 请确保您的系统符合硬件和软件要求。有关详细信息，请参阅[系统要求 \(第 7 页\)](#)。

必备软件：运行 LoadRunner Professional 安装向导时，向导会检测已在计算机上安装的必备软件以及缺少的软件。如果继续安装，将安装所有缺少的必备软件。有关详细信息，请参阅在 [Windows 上安装必备软件 \(第 7 页\)](#)。

- 请参阅[安装注意事项和限制 \(第 8 页\)](#)。

2. 查看安装要求

开始安装前，请确保符合以下安装要求：

- 您必须在指定计算机上具有完全本地管理权限。
- 必须在目标计算机上进行安装。LoadRunner Professional 不支持通过终端服务进行安装。
- 有关可同时在同一物理计算机上安装的组件的完整列表，请参阅[支持列表](#)。

注：建议在安装 LoadRunner Professional 之前关闭所有防病毒应用程序。

3. 执行安装

- a. 如果您使用的是 LoadRunner Professional 的先前版本，请查看升级说明。有关详细信息，请参阅[升级 LoadRunner Professional \(第 14 页\)](#)。
- b. 运行安装向导以在 Windows 系统上安装 LoadRunner Professional 的完整版本、LoadRunner Professional 独立组件或其他组件。有关详细信息，请参阅[在 Windows 上安装 LoadRunner Professional \(第 14 页\)](#)。
然后，根据需要运行任何修补程序安装。
- c. 完成 LoadRunner Professional 安装后，您可以安装本地化版本，以本地语言查看 LoadRunner Professional、VuGen 独立版和 Analysis 独立版用户界面。有关详细信息，请参阅[安装 LoadRunner 语言包 \(第 30 页\)](#)。

注：

- 要执行静默安装，请参阅[静默安装 LoadRunner Professional \(第 18 页\)](#)。
- Windows 上的 VuGen 和 Load Generator 安装包括 32 位 OpenJDK 11。

4. 完成安装后配置

- 将 LoadRunner Professional 配置为在 Load Generator 计算机上运行 Vuser 而无需用户手动登录。有关详细信息，请参阅[配置用户登录设置 \(第 29 页\)](#)。
- 要使用 Controller 运行 Vuser，必须在 Controller 计算机上安装有效的许可证。在 LoadRunner Professional 安装期间，LoadRunner Professional 自动安装社区许可证包 (免费)，提供 50 个 Vuser。

有关管理 LoadRunner Professional 许可证和扩展 Vuser 容量的详细信息，请参阅 [LoadRunner Professional 帮助中心](#) (选择相关版本) 中的许可信息。

升级 LoadRunner Professional

LoadRunner Professional 是完整安装，可以通过安装的任何 LoadRunner 或 LoadRunner Professional (版本 12.5x 及更高版本) 进行安装。

要升级 Windows 安装中的所有组件，请按照在 [Windows 上安装 LoadRunner Professional \(第 14 页\)](#) 中所述执行安装过程。安装过程将检测旧版本，并让您选择升级。

注：有关静默升级，请参阅 [静默安装 LoadRunner Professional \(第 18 页\)](#)。

在 Windows 上安装 LoadRunner Professional

本节介绍如何使用安装向导在 Windows 上安装 LoadRunner Professional。

有关如何执行静默安装的详细信息，请参阅 [静默安装 LoadRunner Professional \(第 18 页\)](#)。

安装向导信息

查看以下信息以帮助您在安装向导成功安装 LoadRunner Professional 组件。有关每个组件的详细信息，请参考 [LoadRunner Professional 帮助中心](#) (选择相关版本)。

LoadRunner 代理 选择“安装后启动 LoadRunner 代理”选项，以在安装后立即在 Load Generator 上启动 LoadRunner 代理。
代理支持 Load Generator 和 Controller 之间的通信。

代理模式	<p>在安装 OneLG Load Generator、MI Listener 或 Monitor over Firewall 组件期间，系统会提示您选择已安装的 LoadRunner 代理的模式：</p> <ul style="list-style-type: none">• LoadRunner Professional 模式。代理作为常规进程运行，需要先登录才能启动该进程。 注：如果要安装用于 Business Process Monitoring (BPM) 的 Load Generator，请选择此模式。• LoadRunner Enterprise 模式。代理作为服务运行，并在操作系统启动时自动启动。• LoadRunner Cloud 模式。这仅与 OneLG 安装有关。选择是否要将 Load Generator 与 LoadRunner Cloud 一起使用。 注：如果需要，可以在安装后更改代理配置。
Silk Performer 代理	<p>选择“安装后安装 Silk Performer 代理”选项，以将 Silk Performer 代理安装在将用于运行 Silk Performer 脚本的 Load Generator 上。</p>
证书	<p>在安装 LoadRunner Professional 完全安装、OneLG、Monitor over Firewall 和 MI Listener 的过程中，可以通过选择“指定 LoadRunner 代理将要使用的证书”选项以选择性地安装 CA 和 TLS (SSL) 证书。</p> <p>这些证书用于进行身份验证和安全通信。这两个证书必须为 *.cer (X.509) 格式。</p> <ul style="list-style-type: none">• CA 证书：输入现有证书的路径，或留空跳过此步骤。CA 证书应提前生成。• TLS 证书：从现有证书文件中选择。或者，如果您提供了包含私钥的 CA 证书，则可以自动生成此证书。

运行安装

以下过程将指导您完成 Windows 安装过程。

要安装 LoadRunner Professional 或其他组件，请执行以下操作：

1. 安装前：

- 查看[系统要求和先决条件 \(第 7 页\)](#)中描述的 安装前信息，包括系统要求和必备软件。
- 如果打算安装 Network Virtualization，请确保您的系统满足要求，如[适用于 LoadRunner 的 Network Virtualization 帮助](#) (选择相关版本)中所述。(您

可以在安装 LoadRunner Professional 期间或稍后安装 Network Virtualization。)

注: 如果将 Network Virtualization 作为 LoadRunner Professional 安装的一部分进行安装，则安装后可能需要重新启动计算机。

- 确保您在指定计算机上具有完全本地管理权限。
 - 确保 Window Update 进程未在运行。
2. **UAC (用户帐户控制):** 您可以在启用 UAC 的情况下在安装计算机上进行安装，但建议您将其禁用，然后重新启动计算机。有关详细信息，请参考 Microsoft Windows 文档。
 3. **DEP (数据执行保护):** 您可以在启用 DEP 的情况下进行安装，但建议您将其禁用，或者仅为必需的 Windows 程序和服务启用它。有关如何更改 DEP 设置的详细信息，请参考 Microsoft Windows 文档。
 4. 运行安装程序包的根文件夹中的 **setup.exe** 文件。

LoadRunner Professional 安装程序启动并显示安装选项。

5. 选择所需的安装选项。

从安装菜单页面中，选择以下某个安装选项：

- **LoadRunner Professional 完整安装。** 安装主要的 LoadRunner Professional 组件，包括 Controller、Virtual User Generator (VuGen)、Analysis、OneLG Load Generator 和 TruClient。对运行负载测试场景的计算机使用此选项。

注: LoadRunner Professional 安装向导可以安装 LoadRunner Professional 或 LoadRunner Enterprise。确保选中 **LoadRunner Professional**。

有关完全安装中包括的组件的详细信息，请参阅 [LoadRunner Professional 组件 \(第 5 页\)](#)。

- **VuGen。** 安装独立版本的 VuGen(包括 TruClient)。
- **Analysis。** 安装独立版本的 LoadRunner Analysis。
- **OneLG。** 安装运行 Vuser 以生成负载所需的独立 Load Generator 组件。请为仅用于生成负载而不控制 Vuser 的计算机使用此选项。

- **Monitors Over Firewall。** 在代理计算机上安装用于通过防火墙进行监控的组件。有关详细信息，请参考 [LoadRunner Professional 帮助中心](#) 中的 **使用防火墙** 部分。
 - **MI Listener。** 安装 MI Listener 计算机所需的组件，用于在防火墙下运行 Vuser 并通过防火墙进行监控。有关详细信息，请参考 [LoadRunner Professional 帮助中心](#) 中的 **使用防火墙** 部分。
 - **语言包。** 仅在本机非英语操作系统上安装时才可使用此选项。可用于在英语版本基础上安装本地化 LoadRunner Professional 用户界面的修补程序。有关详细信息，请参阅 [安装 LoadRunner 语言包 \(第 30 页\)](#)。
 - **其他组件。** 打开位于安装程序包的根文件夹中的 Additional Components 文件夹。有关可以安装的其他组件的详细信息，请参阅 [LoadRunner Professional 帮助中心](#) 中的“高级 > 其他组件”。
6. 需要先安装特定软件 (例如 Microsoft Visual C++)，然后才能安装 LoadRunner Professional (请参阅 [在 Windows 上安装必备软件 \(第 7 页\)](#))。如果计算机上尚未安装必备软件，将会打开一个显示必备程序列表的对话框。

单击**确定**安装列出的软件，然后继续安装 LoadRunner Professional。如果单击“取消”，LoadRunner Professional 安装向导将退出，因为没有必备软件就无法安装 LoadRunner Professional。

注： 安装必备软件后，LoadRunner Professional 可能会重新启动计算机。如果发生这种情况，请重新运行 **setup.exe**。

7. 执行安装。

安装向导将打开，显示“欢迎使用”页面。

按照向导中的说明完成安装。请参考 [安装向导信息 \(第 14 页\)](#) 中的信息以确保成功安装。

注：

- LoadRunner Professional 或 LoadRunner Professional 组件的安装路径不得包含非英语字符。
- 默认安装路径为: **C:\Program Files (x86)\Micro Focus\LoadRunner**

8. 安装完成后，您可以选择安装 Network Virtualization (NV)。

在安装向导中，选择：

- **典型模式**，以执行自动安装。
- **自定义模式**，以更改安装文件夹、数据文件夹和端口。

有关更多详细信息，请参阅 [适用于 LoadRunner 的 Network Virtualization 帮助](#) (选择相关版本) 的安装部分。

9. 如果有适用于已安装版本的修补程序：
 - a. 运行随安装程序包提供的修补程序安装文件。
 - b. 按照屏幕上的安装和部署说明进行操作。

注：

- 要修复安装，请运行位于 LoadRunner Professional 安装包根目录中的 **setup.exe** 文件，选择“LoadRunner Professional 完整安装”，然后选择安装向导中的“修复”选项。
- 您可以将 LoadRunner Professional 配置为在 Load Generator 计算机上运行 Vuser 而无需用户手动登录到此计算机。有关详细信息，请参阅 [配置用户登录设置 \(第 29 页\)](#)。
- 有关安装期间注册的组件的列表，如 **DLL** 和 **OCX** 文件，请参阅安装的 **build_info** 文件夹中的 **RegisteredComponents_Micro Focus LoadRunner.txt** 文件。

静默安装 LoadRunner Professional

静默安装是无需用户交互即可执行的安装。可使用命令行运行安装文件。有关详细信息，请参阅 [安装命令行选项 \(第 21 页\)](#)。

注：

- 安装 LoadRunner Professional 之前，请查看 [系统要求和先决条件 \(第 7 页\)](#) 中描述的安装前信息，包括系统要求和必备软件。
- 在任何计算机上安装 LoadRunner Professional 都需要管理员权限。
- 使用标准 MSI 命令行选项定义安装属性。例如，使用 **INSTALLDIR** 指定备用安装文件夹。
- 要阻止 Load Generator 上的 LoadRunner 代理在安装后立即启动，请在命令行命令中添加以下内容：**START_LGA=0**。代理支持 Load Generator 和

Controller 之间的通信。有关 LoadRunner 代理的更多详细信息，请参阅 [LoadRunner Professional 帮助中心](#) (选择相关版本)。

对 LoadRunner Professional 执行静默安装

1. 从命令行中运行以下其中一个命令：

- 要通过单个命令安装所有 LoadRunner Professional 组件(包括必备软件)，请运行以下命令：

```
<安装包>\lrunner\<您的语言文件夹>\setup.exe /s
```

- 要首先安装所有必备软件，然后安装 LoadRunner Professional，请运行以下命令：

```
<安装包>\lrunner\<语言文件夹>\setup.exe  
/InstallOnlyPrerequisite /s
```

```
msiexec.exe /qn /i "<安装包>\lrunner\MSI\LoadRunner_x64.msi"
```

- 默认情况下，将在安装 LoadRunner Professional 期间安装 Network Virtualization。为便于安装，请将以下内容添加到安装命令：

```
REBOOT_IF_NEED=1
```

- 要阻止安装程序安装 Network Virtualization，请将以下内容添加到安装命令：

```
NVINSTALL=N
```

- 要使用 LoadRunner 产品、OneLG、MiListener 或 MoFW 的 TLS 证书和/或双向 TLS 来保护您的环境，请将以下内容添加到相关安装命令中：

```
CACERT = <*.cer 格式的根证书路径>  
CERT = <*.cer 格式的服务器证书路径>  
CACERT_PK = <CA 证书私钥>  
CACERT_PK_PWD = <使用 CA 证书加密的私钥的密码>  
CERT_PK = <TLS 证书私钥>
```

```
CERT_PK_PWD = <使用 TLS 证书加密的私钥的密码>
```

```
TWO_WAY_SSL = High | Medium
```

以下命令执行 MI Listener 的静默安装。



```
示例: SetupMListener.exe /s /a /s INSTALLDIR="C:\Program Files  
(x86)\Micro Focus\MI Listener" IS_RUNAS_SERVICE=0 START_LGA=1  
NVINSTALL=N CACERT=c:\temp\root.cer CERT=server.cer TWO_  
WAY_SSL=Medium
```

有关详细信息，请参阅 [LoadRunner Professional 帮助中心](#) (选择相关版本) 中有关使用 TLS 进行安全通信的信息。

- 要安装 LoadRunner Professional 独立应用程序 (不是 VuGen 或 OneLG)，请执行以下操作：

```
<安装包>\Standalone Applications\Setup<组件名称>.exe /s /a /s
```

- 要安装 VuGen，请执行以下操作：

```
<安装包>\Standalone Applications\SetupVuGen.exe -s -sp"/s"
```

- 要安装其他组件，请运行以下命令：

```
<安装包>\Additional Components\<安装文件路径> /s /a /s
```



提示: 在此处查看其他示例: [示例 \(第 24 页\)](#)

2. 如果要安装升级程序，请运行以下命令：

```
msiexec.exe /update <msp 文件的完整路径> [/qn] [/l*vx <日志文件的完整路径>]
```

msp 文件位于安装程序包中。

有关详细信息，请参阅 [升级 LoadRunner Professional \(第 14 页\)](#)。

安装命令行选项

您可以使用 Windows 命令行安装完整 LoadRunner Professional 包、独立应用程序、和其他组件。CLI 使用包装文件 (setup.exe) 或 Package for the web 文件 (<PFTW>.exe)。

- 可以使用 setup.exe 文件启动 LoadRunner Professional 完整安装。有关详细信息，请参阅[通过包装文件安装完整的 LoadRunner Professional 程序包 \(第 21 页\)](#)。
- 可以通过 PFTW 文件启动独立应用程序和其他组件的安装。有关详细信息，请参阅[通过 PFTW 文件安装独立应用程序或其他组件 \(第 23 页\)](#)。

通过包装文件安装完整的 LoadRunner Professional 程序包

您可以通过位于 LoadRunner Professional 安装包的 \lrunner\<>您的语言> 文件夹中的安装包装文件 **setup.exe** 安装完整的 LoadRunner Professional。以下命令行选项可用：

选项	描述
/s	在后台(静默)运行安装，无需用户交互。
/qb	在无人照管模式下运行安装，用户交互有限。
/InstallOnlyPrerequisite	仅安装必备软件 – 不安装任何 LoadRunner Professional 组件。 默认情况下，安装程序会检查您的计算机是否具有所需必备软件，并在安装 LoadRunner Professional 组件之前根据需要进行安装。

您可以使用以下语法为命令行安装设置公共属性：

```
setup.exe PROPERTY_NAME="value"
```

以下属性可用：

属性名称	描述
INSTALLDIR="您的路径"	指定应用程序的安装位置。
REBOOT_IF_NEED	1: 根据需要在安装后重新启动计算机。如果包含 NV (Network Virtualization) 安装，则建议使用此选项。 0: 安装后不重新启动计算机。 默认值: 0
NVINSTALL	空字符串: 不包括 NV 组件安装。 Y: 包括 NV 组件安装。 默认值: Y
INSTALL_NV_MODE	1: 将 NV 安装模式设置为 典型 。 2: 将 NV 安装模式设置为 自定义 。 默认值: 典型
INSTALL_DIR="您的路径"	指定安装路径。
START_LGA	空字符串: 指示计算机在安装后不启动 Load Generator。 1: 指示计算机在安装后启动 Load Generator。 默认值: 1
IS_RUNAS_SERVICE	0: LoadRunner Professional 模式，将 Load Generator 的代理作为进程运行。 1: LoadRunner Enterprise 模式，将 Load Generator 的代理作为服务运行。 2: (仅与 OneLG 相关) LoadRunner Cloud 模式，将 Load Generator 的代理作为服务运行，并在安装结束时触发启动 LRC 代理配置。 默认值: 1
INSTALL_SILK_PERFORMER_AGENT	1: 在 Load Generator 上安装 Silk Performer 代理 (用于运行 Silk Performer 脚本)。
SKIP_CHECK_PRIVILEGE	0: 安装过程将检查安装路径中的访问权限。 1: 安装过程将跳过对安装路径的访问检查。 默认值: 0

属性名称	描述
IMPROVEMENTPROGRAM	0: 禁用 LoadRunner 改进计划。 1: 启用 LoadRunner 改进计划。 默认值: 1

示例:

以下命令以静默方式执行安装，然后在必要时重新启动计算机:

```
DVD\lrunner\en\setup.exe /s REBOOT_IF_NEED="1"
```

通过 PFTW 文件安装独立应用程序或其他组件

您可以通过 LoadRunner Professional 安装介质中包含的 Package for the web (PFTW) 文件安装 LoadRunner Professional 独立应用程序或某个其他组件。这些安装文件位于安装介质上的 **Standalone Applications** 或 **Additional Components** 文件夹中。

双击打包文件或者使用以下选项从命令行中运行它:

选项	描述
/s	在后台(静默)运行安装，无需用户交互。
/e	仅提取安装文件；不运行这些文件。
/f	指定用于文件提取的临时文件夹的路径。 例如: /f "c:\my_temp_folder" 如果省略此选项，则使用默认临时文件夹。
/a	可用于将参数或属性传递给自动运行文件，例如 setup.exe。 将上面(在包装文件部分中)定义的属性用于 setup.exe 文件。

注: VuGen 和 OneLG 的静默安装:

- **SetupVuGen.exe** 和 **SetupOneLG.exe** 使用不同的安装命令 (请参阅下面的示例)。
- 如果需要在命令行参数中插入引号字符, 请使用两个连续的引号。

示例

以下命令以静默方式运行安装程序, 并将应用程序安装在指定的文件夹中。在相关情况下, 它们还将在安装后启动代理:

- **SetupVuGen.exe** 和 **SetupOneLG.exe** 的示例:

```
SetupOneLG.exe -s -sp"/s INSTALLDIR=""C:\Program Files (x86)\Micro Focus\LoadRunner OneLG"" IS_RUNAS_SERVICE=0 START_LGA=1 NVINSTALL=N"
```

```
SetupVuGen.exe -s -sp"/s INSTALLDIR=C:\VuGen"
```

- 其他独立应用程序的示例:

```
SetupAnalysis.exe /s /a /s INSTALLDIR="C:\Analysis"
```

```
SetupMILlistener.exe /s /a /s INSTALLDIR="C:\MiListener"
```

```
SetupMoFW.exe /s /a /s INSTALLDIR="C:\MoFW"
```

在 Windows 上部署 Docker 化 Load Generator

本节介绍如何在 Windows 平台上运行 Docker 化 Load Generator。

Docker 是可用于通过容器开发、封装和运行应用程序的平台。

注:

- 有关 Docker 化 Load Generator 上受支持的协议的信息，请参阅[受支持的协议指南](#)。
- 适用于 Windows 上 Docker 的 Load Generator 不完全支持 FTP。支持被动模式 FTP，但不支持主动模式 FTP。

先决条件

注: OneLG Load Generator 的 Docker 映像替换了以前的 Windows 独立 Load Generator Docker 映像。

- 在目标计算机上安装 Docker 及其相关组件，并根据需要设置目标计算机环境。当前仅支持 64 位版本。有关安装详细信息，请参阅 Docker 联机文档。
- 从相关页面中拉取 Windows Load Generator Docker 映像，该映像可从 Docker Hub 的性能测试页面 (<https://hub.docker.com/u/performancetesting>) 进行访问。使用以下命令和适当的 **<标记版本号>**，例如 20.03:

```
docker pull performancetesting/microfocus_onelg_windows:<标记版本号>
```

使用预定义映像运行通过 Docker 设置的 Load Generator

使用即用型映像在于 Windows 的 Docker 上运行 Load Generator (OneLG)。

注: 如果需要对容器进行自定义设置(例如，用于 Java 或以特定用户的身份运行)，请参阅[使用自定义映像运行 Docker 化 Load Generator \(第 26 页\)](#)。

要运行 **Docker 化 Load Generator**，请执行以下操作：

使用以下命令运行 Load Generator 容器：

```
docker run -id -p <主机端口>:54345 performancetesting/microfocus_onelg_windows:<标记版本号>
```

注: 检查计算机上的 **<主机端口>** 是否可用并允许传入请求。连接到此 Load Generator 时，请在 Controller 端指定此端口。

使用自定义映像运行 Docker 化 Load Generator

如果您的环境要求使用自定义设置来运行容器，则可以创建 Dockerfile，为 Windows 上的 Docker 构建一个自定义映像。

自定义映像的示例：

- 使用特定用户帐户运行 Vuser 进程，以便支持访问脚本参数文件等网络资源。运行后，容器应该能够验证用户。
- 在 Windows Load Generator 容器上运行 Java 协议。
- 为代理服务器主机和端口定义环境变量。

要运行 Docker 化自定义 Load Generator，请执行以下操作：

1. 创建一个新文件夹，在其中创建一个名为 **dockerfile** 的文件。将以下 **FROM** 行粘贴到该文件中(注意将其中的 **<标记版本号>** 替换为合适的 LoadRunner Professional 版本)，并添加相关自定义行：

```
FROM performancetesting/microfocus_onelg_windows:<标记版本号>  
<自定义行>
```

有关自定义示例，请参阅 [Dockerfile 的自定义内容示例 \(第 27 页\)](#)



提示：有关可在 Docker 文件中使用的命令的信息，请参阅 Docker 联机文档。

2. 保存 Dockerfile。
3. 在 **dockerfile** 文件夹路径处打开命令行，并使用所需自定义映像名称运行以下命令：

```
docker build -t <自定义映像名称>。
```

4. 通过运行以下命令为要使用的每个 Load Generator 创建容器(或者使用任意 Docker 协调程序工具运行容器)：

```
docker run -id -p <host_port>:54345 <自定义映像名称>
```

如果步骤 3 中的自定义映像是使用标记构建的，则将该标记包含在命令中：

```
docker run -id -p <主机端口>:54345 <自定义映像名称>:<标记版本号>
```

注：检查计算机上的 <主机端口> 是否可用并允许传入请求。连接到此 Load Generator 时，请在 Controller 端指定此端口。

Dockerfile 的自定义内容示例

以指定用户帐户的身份运行 **Vuser** 示例

下面显示的 **dockerfile** 内容示例用来实现以具有共享位置网络访问权限的指定用户帐户的身份运行 **Vuser**。将 <> 之间的值替换为您环境中具有共享资源网络访问权限的有效用户帐户的凭据。



示例：

```
#escape=`  
FROM performancetesting/microfocus_onelg_windows:<标记版本号>  
RUN c:\LG\launch_service\bin\magentservice.exe -remove  
RUN c:\LG\launch_service\bin\magentservice -install <域>\<用户名>  
<密码>
```

运行 Java/JMeter/Gatling 协议的示例

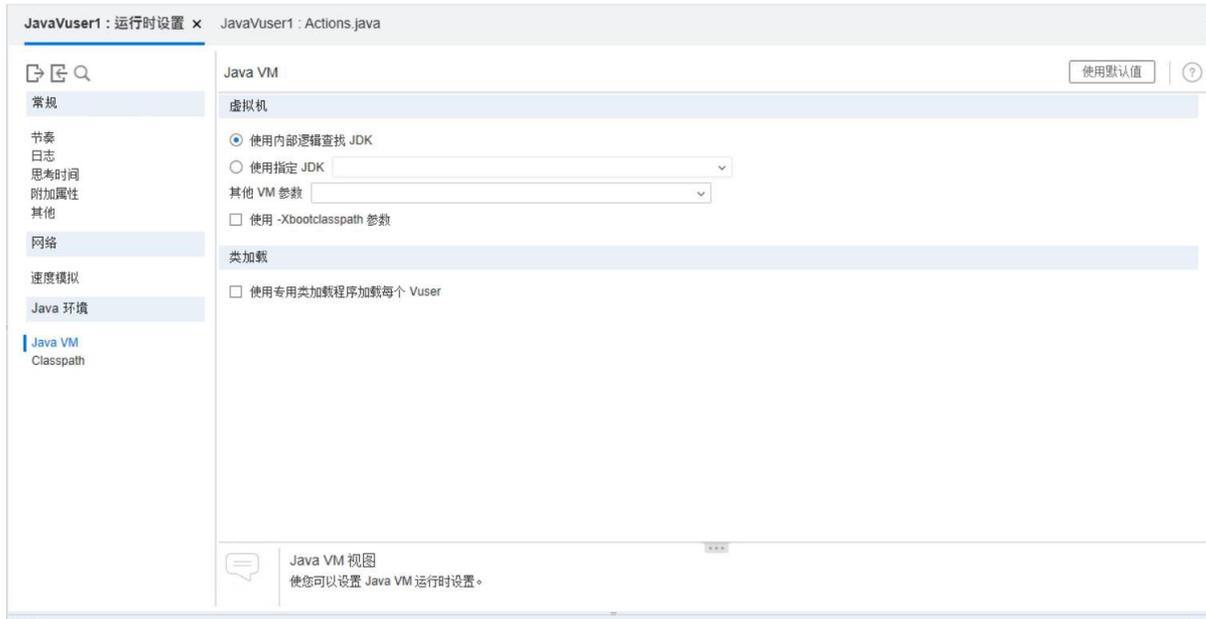
下面显示的是用于运行 Java、JMeter 或 Gatling 协议的 **dockerfile** 内容示例：



示例：

```
#escape=`  
FROM performancetesting/microfocus_onelg_windows:<标记版本号>  
COPY .\<包含 JDK 的文件夹> <容器中的目标路径>
```

- 还必须将为 **COPY** 行中的 <容器中的目标路径> 定义的目标 JDK 目录路径添加到 **Java VM** 运行时设置页面：



- 对于 Java 64 位协议测试，在 dockerfile 中包含以下命令行，以将 JDK 64 位的 **bin** 文件夹路径添加到计算机 PATH 环境变量：

```
RUN powershell [Environment]::SetEnvironmentVariable(\"Path\",  
$env:Path + \"<容器中的目标 JDK 路径>\bin\",  
[EnvironmentVariableTarget]::Machine)
```

- 对于 JMeter/Gatling 协议测试，在 dockerfile 中包含以下命令行，以将 JDK/JRE 的路径添加到计算机 JAVA_HOME 环境变量：

```
RUN powershell [Environment]::SetEnvironmentVariable(\"JAVA_  
HOME\", \"<容器中的目标 JDK/JRE 路径>\",  
[EnvironmentVariableTarget]::Machine)
```

运行 Load Generator 容器后

将 Load Generator 容器添加到场景中。有关详细信息，请参阅 [LoadRunner Professional 帮助中心](#) (选择相关版本) 中有关配置 Docker 化 Load Generator 的信息。

提示和准则

- 在防火墙上运行时，不支持通过预定义映像运行的 Docker 化 Load Generator。
- 使用 `docker ps` 列出正在运行的容器。
- 要停止 Load Generator 服务，请执行以下操作：
 - 如果要重用相同的 Load Generator，请使用 `docker stop <Load Generator 容器名称或 ID>`。
 - 使用 `docker rm -f <Load Generator 容器名称或 ID>` 删除 Load Generator 容器。
- 要直接访问主机网络，请使用 `--net=host` 替代 `-p <主机端口>:54345`。建议在 AUT 生成很多网络活动时使用此标志。

配置用户登录设置

默认情况下，您需要先手动登录计算机，LoadRunner Professional 才能在该计算机上运行 Vuser。但是，您可以将 LoadRunner Professional 配置为在 Load Generator 计算机上运行 Vuser，这样就无需手动登录到该计算机。

要配置用户登录设置，请执行以下操作：

1. 请执行以下其中一个操作：
 - 选择 Windows“开始”菜单 >“Micro Focus”>“LoadRunner 代理运行时设置配置”。
 - 在基于图标的桌面上 (如 Windows 8)，搜索“LoadRunner 代理”，然后选择“LoadRunner 代理运行时设置配置”项。“LoadRunner 代理运行时设置”对话框将打开。
2. 选择下列选项之一：
 - 允许虚拟用户在该计算机上运行，而无需用户登录。LoadRunner Professional 将自动从 Load Generator 计算机登录网络，这样 Vuser 无需任何手动干预即可运行。输入用户计算机所在的网络域以及用户名和密码。

注: 创建后，使用 **LocalSystem** 帐户 (不作为指定用户) 启动 LoadRunner 代理服务。运行脚本时，代理服务使用指定凭据启动

! mdrv.exe 进程。

- **手动登录到该计算机。** 对于每个运行 Vuser 的会话，用户必须从 Load Generator 计算机手动登录到网络。

3. 单击**确定**。

! **注:** 在 LoadRunner Professional 安装后可以自动登录之前，您必须重新启动并登录系统至少一次。

安装 LoadRunner 语言包

语言包支持以本地语言显示 LoadRunner Professional 用户界面。您可以为相关 LoadRunner Professional 组件 (LoadRunner Professional 完整版、VuGen 独立版或 Analysis 独立版) 安装语言包 (通过 LoadRunner Professional 安装包安装)。

有关受支持语言的列表，请参阅[支持列表](#)。

语言包仅适用于本机非英语操作系统。

! **重要:** 操作系统的本机语言必须与要安装的语言包相同。例如，西班牙语语言包必须安装在具有本机西班牙语操作系统的计算机上。

要安装语言包，请执行以下操作：

1. 确保已经安装英语版 LoadRunner Professional。
2. 在 LoadRunner Professional 安装程序包的根文件夹中，运行 **setup.exe**。LoadRunner Professional 安装程序启动并显示安装选项。
3. 单击 **Language Packs**。将打开安装程序包中的 Language Packs 文件夹。
4. 导航到要安装的语言和组件对应的文件夹并运行安装文件。例如，要在 VuGen 独立版计算机上安装法语语言包，则路径为 **..\Language Packs\French\VugenSA\Vugen_FRA.msp**。

按照联机说明操作。

在 Linux 上安装 Load Generator

LoadRunner Professional 使用 Load Generator 运行 Vuser。存在两种版本的 LoadRunner Professional Load Generator。一种版本在 Windows 平台上运行 Vuser，另一种版本在 Linux 平台上运行 Vuser。您可以使用基于 Windows 的 LoadRunner Controller 来控制基于 Windows 和基于 Linux 的 Vuser。

本章描述如何在 Linux 平台上安装 Load Generator。有关如何在 Windows 平台上安装 Load Generator 的详细信息，请参阅[在 Windows 上安装 LoadRunner Professional \(第 11 页\)](#)。

注：

- 您可以按照本节所述使用安装程序(基于 UI 或静默安装)安装 Linux Load Generator，或者通过 Docker 容器进行安装 — 有关详细信息，请参阅[在 Linux 上部署 Docker 化 Load Generator \(第 36 页\)](#)。
- 有关疑难解答信息，请参阅[Load Generator Linux 安装疑难解答 \(第 49 页\)](#)。

本章包括：

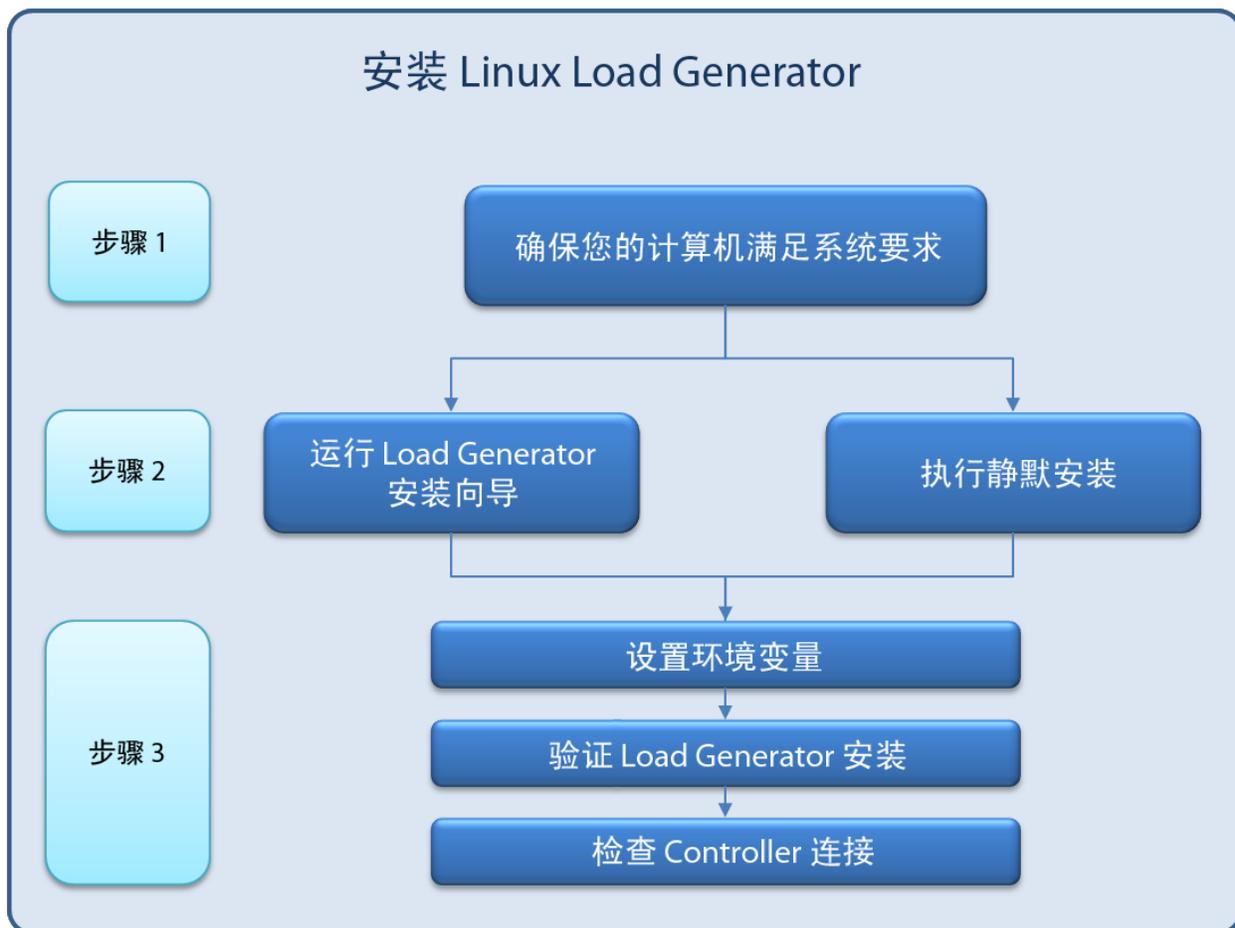
• 安装工作流	31
• 通过安装向导在 Linux 上安装 Load Generator	33
• Linux 上的 Load Generator 静默安装	35
• 在 Linux 上部署 Docker 化 Load Generator	36
• 配置 Linux 环境	41
• 改善 Load Generator 性能	45
• 从 Linux 计算机上卸载 Load Generator	48
• Load Generator Linux 安装疑难解答	49

安装工作流

可以在 Linux 平台上安装 Load Generator 组件以运行 Vuser。基于 Linux 的 Vuser 可以与安装在 Windows 平台上的 LoadRunner Controller 进行交互。

Load Generator 可以安装在本地 Linux 计算机上，也可以在云帐户中进行配置。有关管理云上的 Load Generator 的信息，请参阅 LoadRunner Professional 帮助中心中的[云上的 Load Generator](#)。

下图显示了使用标准安装程序在 Linux 平台上安装 Load Generator 的主要步骤：



1. 查看系统要求

- 在 Linux 平台上安装 Load Generator 之前，请确保系统满足如[系统要求和先决条件 \(第 7 页\)](#)中所述的要求。
- 如果您打算安装 **Network Virtualization**: 确保您的系统满足要求，如[适用于 LoadRunner 的 Network Virtualization 帮助 \(选择相关版本\)](#)中所述。
- 如果您使用的是 **RHEL 8** 操作系统: 默认情况下，RHEL 8.0 中未安装 `libnsl.so.1`，因此您需要在安装 Load Generator 之前手动安装它，否则安装将失败。使用包管理器进行安装:`yum install libnsl.i686`
- 如果您打算在 Linux 计算机上回放 ODBC 协议脚本: 安装 `unixODBC v.2.3.1` 或更高版本。

2. 运行安装过程

注：

- 可以在任何 Load Generator 安装版本 (12.56 版及更高版本) 上安装此版本的 Load Generator。
- 如果安装了 Load Generator 12.55 或更低版本，则必须先将其卸载。有关详细信息，请参阅[卸载 Load Generator 12.55 或更低版本 \(第 48 页\)](#)。
- 建议在安装 LoadRunner Professional 之前关闭所有防病毒应用程序。

- **向导安装：**使用 Load Generator 安装向导开始安装。有关详细信息，请参阅[通过安装向导在 Linux 上安装 Load Generator \(第 33 页\)](#)。
- **静默安装：**要执行静默安装，请参阅[Linux 上的 Load Generator 静默安装 \(第 35 页\)](#)。

注：Linux 上的 Load Generator 安装包括 32 位 OpenJDK 8。

3. 配置环境

在安装完成后开始使用 Load Generator 之前，需要先配置环境。这包括设置相应的环境变量，检查对 Load Generator 的访问权以及验证安装情况。请参阅[配置 Linux 环境 \(第 41 页\)](#)。

通过安装向导在 Linux 上安装 Load Generator

本节介绍如何使用安装向导在 Linux 平台上安装 Load Generator。

注：要通过 Docker 容器执行安装，请参阅[在 Linux 上部署 Docker 化 Load Generator \(第 36 页\)](#)。

要执行静默安装，请参阅[Linux 上的 Load Generator 静默安装 \(第 35 页\)](#)。

本节包括：

- [开始之前 \(第 34 页\)](#)
- [运行安装向导 \(第 34 页\)](#)

开始之前

安装期间，可以选择性地安装 Load Generator 的 CA 和 SSL 证书。这些证书分别用于进行身份验证和安全通信。这两个证书应为 *.cer (X.509) 格式。

- 对于 CA 证书：可以输入现有证书的路径，或留空跳过此步骤。如果要安装 CA 证书，应预先生成此证书。
- 对于 SSL 证书：可以从现有证书文件中选择此证书。或者，如果您提供了包含私钥的 CA 证书，则可以自动生成此证书。

有关生成 CA 证书的详细信息，请参阅[LoadRunner Professional 帮助中心](#) (选择相关版本)中有关创建和安装数字证书的章节。

有关在 LoadRunner Professional 中使用证书的详细信息，请参阅[LoadRunner Professional 帮助中心](#)中有关配置客户端-服务器身份验证的章节。

运行安装向导

要安装 Load Generator，请执行以下操作：

1. 切换为超级用户。
2. 如果计算机上安装了 Load Generator 12.55 或更低版本，则必须手动将其卸载。有关详细信息，请参阅[卸载 Load Generator 12.55 或更低版本 \(第 48 页\)](#)。

如果不卸载 Load Generator，则在安装期间将出现错误。

提示：如果安装了 Load Generator 12.56 或更高版本，安装向导将在安装过程中卸载该版本，然后重新安装。

3. 解压缩 Load Generator 安装包，打开 **.bin** 文件所在的位置。
4. 启动安装向导：

对于：	输入：
[sh 和 bash shells]	source ./installer.sh

对于:	输入:
[csh 和 tcsh shells]	source ./installer.csh
如果当前 shell 不支持 source 命令, 请使用“dot”命令。	例如: ../installer.csh

注: 建议使用 **source** 命令运行安装向导, 如上所示。如果不使用此命令运行安装向导, 则必须手动为当前 Shell 会话设置环境变量。有关详细信息, 请参阅 [设置环境变量 \(第 41 页\)](#)。

Load Generator 安装向导会检查计算机上是否安装了所需必备软件。如果缺少任何必备软件, 则会显示消息, 并且安装向导将中止。安装所需程序包。然后重新运行安装向导。

- 按照联机说明安装 Load Generator。有关安装期间的更多帮助, 请参阅 [Load Generator Linux 安装疑难解答 \(第 49 页\)](#)。

注: 默认安装路径为: `/opt/MF/MF_LoadGenerator/_MF_LoadGenerator_Installation`

- 退出超级用户或切换到其他用户。
- 按照 [配置 Linux 环境 \(第 41 页\)](#) 中所述配置您的环境。

Linux 上的 Load Generator 静默安装

本节介绍如何在 Linux 计算机上执行 Load Generator 静默安装。

注: 您必须拥有超级用户权限。

要执行 Load Generator 静默安装, 请执行以下操作:

- 如果计算机上安装了 Load Generator, 则必须先将其卸载, 否则静默安装将失败。要卸载 Load Generator, 请参阅 [从 Linux 计算机上卸载 Load Generator \(第 48 页\)](#)。
- 将目录更改为 `/<path_to_installer_dvd>/load_generator_linux_x64/`。
- 运行以下命令对 Load Generator 执行静默安装:

```
source ./installer.sh -i silent
```

注: 建议您使用 **source** 命令执行静默安装，如上所示。如果不使用 **source** 命令执行安装，将需要在安装 Load Generator 后设置环境变量。有关详细信息，请参阅 [设置环境变量 \(第 41 页\)](#)。

如果当前 shell 不支持 **source** 命令，请使用“dot”命令。例如
`./installer.sh - i silent`。

如果在安装期间遇到错误，请参阅 [Load Generator Linux 安装疑难解答 \(第 49 页\)](#)。

默认情况下，Load Generator 会在安装结束时启动。如果不希望自动启动 Load Generator，请添加以下命令行选项：

```
source ./installer.sh -i silent -DSTART_PRODUCT_AFTER_INSTALL=No
```

在 Linux 上部署 Docker 化 Load Generator

本节介绍如何在 Linux 分发上运行通过 Docker 设置的 Load Generator。

Docker 是可用于通过容器开发、封装和运行应用程序的平台。

注: 有关 Docker 化 Load Generator 上受支持的协议的信息，请参阅 [受支持的协议](#) 指南。

先决条件

注: OneLG Load Generator 的 Ubuntu 映像替换了先前的 Ubuntu Load Generator Docker 映像。

有关 OneLG 的信息，请参阅 [LoadRunner Professional 帮助中心](#) (选择相关版本) 中的 [Load Generator](#)。

- 在目标计算机上安装 Docker 及其相关组件，并根据需要设置目标计算机环境。当前仅支持 64 位版本。有关安装详细信息，请参阅 Docker 联机文档。

- 获取预定义的 Load Generator Docker 映像。对于 Linux-Ubuntu 和 RHEL，有两个映像可用。

从相关页面中拉取相应映像，这些映像可从 Docker Hub 中的性能测试页面 (<https://hub.docker.com/u/performancecetesting>) 进行访问。

使用以下命令之一和相应的“<标记版本号>”，例如 20.03:

- 对于 Linux-Ubuntu:

```
docker pull performancecetesting/microfocus_onelg_linux_ubuntu:<标记版本号>
```

- 对于 RHEL:

```
podman pull performancecetesting/load_generator_redhat:<标记版本号>
```

使用预定义映像运行通过 Docker 设置的 Load Generator

使用即用型映像在用于 Linux 的 Docker 上运行 Load Generator。

注:

- 要为 LoadRunner Professional 运行 Linux-Ubuntu OneLG 映像，必须在 run 命令中包含适当的环境变量: **ONELG_FLAVOR=1**
- 如果需要对容器(例如，代理服务器)进行自定义设置，请参阅[使用自定义映像运行 Docker 化 Load Generator \(第 38 页\)](#)。

要运行 Docker 化 Load Generator，请执行以下操作:

使用适当的命令运行 Load Generator 容器。

Linux-Ubuntu:

```
docker run -id -p <主机端口>:54345 -e "ONELG_FLAVOR=1" --net=host performancecetesting/microfocus_onelg_linux_ubuntu:<标记版本号>
```

RHEL:

```
podman run -id -p <主机端口>:54345 performancetesting/load_
generator_redhat:<标记版本号>
```

注: 检查 Linux 计算机上的 <主机端口> 是否可用并允许传入请求。连接到此 Load Generator 时, 请在 Controller 端指定此端口。

使用 SSH 的示例

下面给出了使用 SSH 运行多个 Load Generator 容器的简单 C# 代码示例。容器编排工具也可以执行此操作, 例如 Kubernetes、OpenShift、Docker Swarm。

```
using (var client = new SshClient(dockerHost, dockerHostUserName,
dockerHostPasswd))
{
    client.Connect();
    for (int i = 0; i > numOfContainers; i++)
    {
        string command = "docker run -id -p " + lgInitialPort + i) +
":54345 -e "ONELG_FLAVOR=1" --net=host
performancetesting/microfocus_onelg_linux_ubuntu:<tag version
number>";
        var terminal = client.RunCommand(command);
        if (terminal.ExistStatus != 0)
        {
            throw new Exception("Failed to create new Docker container");
        }
        Console.WriteLine("Docker LG with external port" + lgInitialPort +
i + "created.");
    }
    client.Disconnect();
}
```

使用自定义映像运行 Docker 化 Load Generator

如果您的环境需要用于运行容器的自定义设置(例如用于代理服务器), 则可以创建用于构建自定义映像的 Dockerfile。

注: 另一种自定义设置方法: 启动容器; 容器运行时, 设置 Load Generator 环境变量, 然后在容器内手动启动 Load Generator。

要运行 **Docker** 化自定义 **Load Generator**，请执行以下操作：

1. 创建一个新文件夹，在其中创建一个名为 **dockerfile** 的文件。对 **FROM** 行中的 <标记版本号> 使用相应的 LoadRunner Professional 版本，然后将该行以及所需自定义行粘贴到文件中。

例如，对于 Linux-Ubuntu:

```
FROM performancetesting/microfocus_onelg_linux_ubuntu:<标记版本号>  
ENV http_proxy http://my_proxy_name:port
```

注：上面的自定义示例适用于代理。它为目标映像中的代理服务器主机和端口定义环境变量。

2. 保存 **dockerfile**。
3. 在 **dockerfile** 文件夹路径处打开命令行，并使用所需自定义映像名称运行以下命令：

Linux-Ubuntu:

```
docker build -t <自定义映像名称>。
```

RHEL:

```
podman build -t <自定义映像名称>。
```

4. 通过运行以下命令为要使用的每个 Load Generator 创建容器：

Linux-Ubuntu:

```
docker run -id -p <host_port>:54345 -e "ONELG_FLAVOR=1" --  
net=host <自定义映像名称>
```

RHEL:

```
podman run -id -p <主机端口>:54345 <自定义映像名称>
```

如果步骤 3 中的自定义映像是使用标记构建的，则将该标记包含在命令中，例如：

```
docker run -id -p <host_port>:54345 -e "ONELG_FLAVOR=1" --net=host <自定义映像名称>:<标记版本号>
```

注：检查 Linux 计算机上的 <主机端口> 是否可用并允许传入请求。连接到此 Load Generator 时，请在 Controller 端指定此端口。

运行 Load Generator 容器后

将 Load Generator 容器添加到场景中。有关详细信息，请参阅 [LoadRunner Professional 帮助中心](#) (选择相关版本) 中关于配置 Docker 化 Load Generator 的信息。

提示和准则

- 在防火墙上运行时，不支持通过预定义映像运行的 Docker 化 Load Generator。(适用于高级用户的解决方法：您可以利用 MI Listener 来开发自己的 Docker 映像。)
- 使用 `docker ps` 列出正在运行的容器。
- 要停止 Load Generator 服务，请执行以下操作：
 - 如果要重用相同的 Load Generator，请使用 `docker stop <Load Generator 容器名称或 ID>`。
 - 使用 `docker rm -f <Load Generator 容器名称或 ID>` 删除 Load Generator 容器。
- Dockerfile 容器有一个 ENTRYPOINT 部分。该容器先在 ENTRYPOINT 中运行命令。该操作将设置环境，然后启动 Load Generator。该命令使用 While 循环等待输入，以防止容器退出。此行为将阻止您在容器运行时访问容器。确保在启动容器时添加 `-i`，否则 While 循环将占用大量 CPU。
- 如果需要进入容器，请在启动容器时添加参数，如 `--entrypoint=/bin/bash`。进入容器后，设置 Load Generator 环境并启动 Load Generator。然后可以使用 `CTRL+p` 和 `CTRL+q` 切换到主机，同时保持容器在后台运行。要再次访问容器，请使用 `docker attach container_id` 命令。
- 要直接访问主机网络，请使用 `--net=host` 替代 `-p <主机端口>:54345`。建议在 AUT 生成很多网络活动时使用此标志。

配置 Linux 环境

本节介绍在安装 Load Generator 之后，在可以开始使用 Load Generator 之前需要完成的配置步骤。

要在安装 **Load Generator** 之后完成设置过程，请执行以下操作：

1. 设置合适的环境变量。

请参阅[设置环境变量 \(第 41 页\)](#)。

注：如果您使用了 **source** 命令安装 Load Generator，则安装向导会自动设置合适的环境变量，无需执行此步骤。

2. 验证 Load Generator 安装。

请参阅[验证 Linux 安装 \(第 42 页\)](#)。

3. 启动 Load Generator。

```
.cd /opt/MF/MF_LoadGenerator/;source env.csh;cd bin;./m_daemon_
setup -install
```

4. 检查 Controller 能否访问 Load Generator。

有关详细信息，请参阅[检查 Controller 连接 \(第 44 页\)](#)。

设置环境变量

注：此主题仅适用于未使用 **source** (或 "dot") 命令运行 Load Generator 安装向导的情况。如果使用了这些命令，则无需执行下面描述的任何步骤。

要运行 Load Generator，必须定义以下环境变量：

- **M_LROOT**。Linux Load Generator 根文件夹的位置。
- **PATH**。Linux Load Generator **bin** 目录的位置。
- **PRODUCT_DIR**。Linux Load Generator 根文件夹的位置。

Load Generator 安装向导将执行以下与环境变量相关的任务：

- 将环境变量定义添加到整个系统的启动脚本中。

如果在安装时未正确设置变量定义，请参阅 [Load Generator Linux 安装疑难解答 \(第 49 页\)](#)。

- 如果使用 **source** 命令运行安装向导，请为当前 Shell 会话设置环境变量。

此主题描述在未使用 **source** 命令运行安装向导时如何为当前 Shell 会话设置环境变量。

要确定是否设置了环境变量，请运行 **verify_generator**(请参阅 [运行 verify_generator \(第 43 页\)](#))或使用以下命令：

```
echo $M_LROOT
```

如果返回了 Load Generator 根文件夹的名称，则已为当前 Shell 正确设置环境变量。如果未返回 Load Generator 根文件夹的名称，请按照如下所述手动设置变量。

要手动为当前 Shell 会话设置环境变量 (如果未使用 **source** 命令运行安装向导)，请执行以下命令之一：

- Bash 用户：

```
source <Load Generator 根>/env.sh
```

- C Shell 用户：

```
source <Load Generator 根>/env.csh
```

验证 Linux 安装

Load Generator 安装包括安装验证实用程序 **verify_generator**，用于检查 Load Generator 在 Linux 计算机上的安装情况。验证实用程序可检查环境变量和启动脚本 (**/etc/csh.cshrc**、**`\${HOME}/.cshrc** 或 **/etc/profile**、**`\${HOME}/.profile**)，以验证它们是否已正确设置。

强烈建议您在安装 Load Generator 之后、尝试调用 Load Generator 之前，运行 **verify_generator**。有关如何运行 **verify_generator** 实用程序的详细信息，请参阅 [运行 verify_generator \(第 43 页\)](#)。

verify_generator 实用程序将检查以下内容：

- 所有必备软件已安装。(此检查仅针对 64 位安装执行。)
- 至少有 128 个文件描述符
- 已正确定义了 `.rhosts` 权限：`-rw-r--r--`
- 使用 `rsh`，让主机之间能够互相通信。如果主机之间不能互相通信，`rsh` 将检查 `.rhosts` 中的主机名
- 已定义了 **M_LROOT**
- `.cshrc` 或 `.profile` 定义了正确的 **M_LROOT**
- `/etc/csh.cshrc`、`${HOME}/.cshrc` 或 `/etc/profile`、`${HOME}/.profile` 将定义正确的 **M_LROOT**
- 主目录中存在 `.cshrc` 或 `.profile`
- 当前用户是 `.cshrc` 或 `.profile` 的所有者
- **\$M_LROOT** 中已存在 Linux Load Generator 安装
- 可执行文件拥有可执行权限
- `PATH` 包含 **\$M_LROOT/bin** 和 `/usr/bin`

运行 verify_generator

建议在安装 Load Generator 之后和尝试调用 Load Generator 之前，运行 **verify_generator** 实用程序。有关 **verify_generator** 实用程序检查内容的详细信息，请参阅 [验证 Linux 安装 \(第 42 页\)](#)。

注：

- 要运行此命令，您必须是“正常”用户，而不是根用户。
- 在运行 **verify_generator** 实用程序之前，请先确保已在计算机上设置了 `DISPLAY` 环境变量。

要运行 **verify_generator**，请执行以下操作：

1. 从 **<Load Generator 根>/bin** 文件夹中运行以下命令：

```
./verify_generator
```

例如：

```
/opt/MF/MF_LoadGenerator/bin/verify_generator
```

如果您想收到有关检查的详细信息，可以使用 **-v** 选项，如下所示：

```
./verify_generator -v
```

2. 查看结果。

- 如果设置正确，**verify_generator** 将返回 **OK**。
- 如果有任何设置不正确，**verify_generator** 将返回 **Failed**，并就如何纠正设置给出建议。

检查 Controller 连接

如果 Controller 要使用 **rsh** (远程 Shell) 远程连接到 Load Generator，则需要确保 Controller 可以远程访问 Load Generator。

1. 在 Load Generator 计算机上，找到位于用户主目录中的 **.rhosts** 文件。
2. 在 **.rhosts** 文件中，验证 Controller 是否包含在计算机列表中。如果不在此列表中，请将其添加进来。

如果 Controller 仍然无法连接到 Load Generator，请与系统管理员联系。

不使用 rsh 的情况下连接到 Linux Load Generator

您可以将 Controller 配置为在不使用 **rsh** 的情况下连接到 Load Generator。在这种情况下，需要激活 Load Generator 上的代理守护程序，如下所述。

本节介绍如何在不使用 **rsh** 的情况下连接到 Linux Load Generator。

1. 在 Linux Load Generator 上，通过从 **<Load Generator 根>/bin** 输入以下命令来运行代理守护程序：

```
./m_daemon_setup -install
```

这样将运行名为 **m_agent_daemon** 的守护程序，如果成功，您将收到消息：**m_agent_daemon <进程 ID>**。

代理现在开始会一直运行，即使用户注销后也会运行。您可以通过使用下面步骤 9 中描述的命令或重新启动计算机来停止该代理。

注：如果您查看临时目录中的 **m_agent_daemon[xxx].log** 日志文件，则即使安装成功，您也将看到通信错误。

2. 在 Controller 中，选择“场景”>“Load Generator”。Load Generator 对话框打开。
3. 单击**添加**。将打开“添加新的 Load Generator”对话框。
4. 在**名称**框中，输入正运行 Load Generator 的计算机的名称。
5. 从**平台**列表中，选择 **Linux**。
6. 单击**更多**。
7. 单击 **Linux 环境**选项卡，并确保选中**不使用 RSH**复选框。
8. 正常连接。
9. 要停止代理守护程序，请从 <LoadRunner Professional 根/>bin 目录中运行以下命令：

```
./m_daemon_setup -remove
```

这样将停止 **m_agent_daemon** 守护程序，如果成功，您将收到消息：**m_agent_daemon is down**。

改善 Load Generator 性能

本节介绍用于提高 Load Generator 性能的方法。可以通过配置内核增加文件描述符数、进程条目数和交换空间大小。

注：大多数使用 Linux Load Generator 的操作系统都有足够的默认文件描述符、进程条目和交换空间，并且很少需要重新配置。

本节包括：

- [增加文件描述符](#) 46
- [增加进程条目数](#) 47
- [增加交换空间](#) 47

增加文件描述符

Load Generator 使用以下文件描述符资源：

- 用于启动服务的 14 个文件描述符
- 代理的 20 个文件描述符
- 用于每个 Vuser 驱动程序的 30 个文件描述符。默认情况下，每 50 个 Vuser 有一个驱动程序。
- 运行 Vuser 的文件描述符。每个 Vuser 需要两个描述符。

例如，要计算用于运行 100 个线程化 Vuser 的文件描述符数，Load Generator 需要：

描述符	描述符用途
14	用于启动程序
20	用于代理
60	用于 2 个驱动程序(30 x 2, 每个驱动程序对应 50 个 Vuser)
200	用于 100 个 Vuser(每个 Vuser 需要 2 个)

总计：294 个文件描述符

如果 Vuser 作为进程而不是作为线程运行，将为每个 Vuser 运行一个驱动程序。因此，每个 Vuser 就需要 30 个文件描述符。

增加文件描述符数的过程会因 Shell 而异。

在以下示例中，描述符数增加到最大数 1024。

- 对于 sh 和 ksh 用户，输入：

```
ulimit -n 1024
```

- 对于 `csch` 用户，键入：

```
limit descriptors 1024
```

下面是增加文件描述符的备用过程。在此示例中，描述符数增加到最大数 8192。

1. 将以下代码行添加到 `/etc/security/limits.conf` 文件：

```
hard nfile 8192
```

2. 将以下代码行添加到 `/etc/sysctl.conf` 文件：

```
fs.file-max = 8192
```

3. 重新启动计算机。

增加进程条目数

每个 `Vuser` 需要多个空闲的进程条目。要增加系统中的进程条目数，必须重新配置内核。

本节介绍如何为 Linux 平台重新配置内核。

1. 查找 `/etc/security/limits.conf` 文件。
2. 在 `limits` 文件中设置最大进程数。输入：

```
hard nproc 8192
```

3. 重新启动计算机。

增加交换空间

每个 `Vuser` 需要范围从 200 KB 到 4 MB 的交换空间。增加系统配置空间之前，应确定分页需求。如果环境内要运行需要极大内存空间的程序，建议使用 4 倍于物理内存的分页空间。如果没有足够的分页空间，则某些进程可能被终止，而其他进程将无法启动。

从 Linux 计算机上卸载 Load Generator

本节介绍如何从 Linux 计算机中卸载 Load Generator。

本节包括：

- [卸载 Load Generator 12.55 或更低版本 \(第 48 页\)](#)
- [卸载 Load Generator 12.56 或更高版本 \(第 49 页\)](#)

卸载 Load Generator 12.55 或更低版本

您可以使用手动命令或以静默方式卸载 Load Generator 12.55 或更低版本。

要卸载 Load Generator 版本 12.55 或更低版本，请执行以下操作：

1. 确保使用安装 Load Generator 的同一用户登录。
2. 确保 **m_agent_daemon** 进程未在计算机上运行。如果该进程正在运行，请终止它：

```
cd /opt/HP/HP_LoadGenerator/bin;./m_daemon_setup -kill;su -;
```

3. 将当前目录更改为安装目录：

```
cd <安装文件夹的路径>/_HP_LoadGenerator_Installation
```

4. 切换为超级用户。
5. 卸载 Load Generator：

- **手动卸载：** 运行以下命令：

```
sh ./Change_HP_LoadGenerator_Installation
```

- **静默卸载：** 运行下列命令：

```
sh ./Change_HP_LoadGenerator_Installation -i silent
```

卸载 Load Generator 12.56 或更高版本

您可以使用安装向导或手动命令卸载 Load Generator 12.56 或更高版本，也可以静默卸载。

要卸载 Load Generator 版本 12.56 或更高版本，请执行以下操作：

1. 确保使用安装 Load Generator 的同一用户登录。
2. 确保 **m_agent_daemon** 进程未在计算机上运行。如果该进程正在运行，请终止它：

```
cd /opt/MF/MF_LoadGenerator/bin;./m_daemon_setup -kill;su -;
```

3. 将当前目录更改为安装目录：

```
cd <安装文件夹的路径>/_MF_LoadGenerator_Installation
```

4. 切换为超级用户。
5. 卸载 Load Generator：

- **手动卸载：** 运行下列命令：

```
sh ./Change_MF_LoadGenerator_Installation
```

- **静默卸载：** 运行下列命令：

```
sh ./Change_MF_LoadGenerator_Installation -i silent
```

Load Generator Linux 安装疑难解答

本节介绍与 Linux Load Generator 安装有关的任务疑难解答。

本节包括：

- [未正确设置整个系统的启动脚本中的环境变量 \(第 50 页\)](#)
- [在 Linux 平台上安装 Load Generator 时出错 \(第 53 页\)](#)
- [在启用了 SELinux 的 RedHat Enterprise Linux 5.x 上运行 Load Generator 时出错 \(第 54 页\)](#)

- [卸载 Load Generator 后，未取消设置环境变量 \(第 54 页\)](#)
- [无法在 Load Generator 上运行 Vuser \(第 55 页\)](#)

未正确设置整个系统的启动脚本中的环境变量

要运行 Load Generator，必须修改整个系统的启动脚本以设置特定的环境变量。Load Generator 安装向导会对启动脚本进行所需的修改。如果在 Load Generator 安装期间未正确修改启动脚本，您可以手动对启动脚本进行所需的更改，如下所述。C Shell 用户和 Bourne 与 Korn Shell 用户之间的所需更改稍微有所不同。

- **手动修改 C Shell 用户的启动脚本**

在 Load Generator 安装过程期间，向导将创建 **env.csh** 脚本。此脚本包括用于设置 C Shell 用户所需的环境变量的命令。示例 **env.csh** 脚本如下所示。

```
setenv PRODUCT_DIR <Load Generator 安装目录>

setenv M_LROOT ${PRODUCT_DIR}

  if ( ! $?PATH ) then
    setenv PATH ""
  endif

setenv PATH ${M_LROOT}/bin:${PATH}"
```

将以下行添加到 **/etc/csh.cshrc** 或 **~/.cshrc** 启动脚本中，以在 Shell 启动期间执行 **env.csh** 脚本：

```
source <Load Generator 安装目录>/env.csh
```

例如：

```
source /opt/MF/MF_LoadGenerator/env.csh
```

对启动脚本进行如上修改的效果与由安装向导进行的修改类似。安装向导对 **/etc/csh.cshrc** 启动脚本进行修改的示例如下所示：

```
# New environment setting added by MF_LoadGenerator on Wed Jan 30
16:20:10 IST 2020 2.

# The unmodified version of this file is saved in
/etc/.login1557000131.

# Do NOT modify these lines; they are used to uninstall.
setenv PRODUCT_DIR "/opt/MF/MF_LoadGenerator"

# End comments by InstallAnywhere on Wed Jan 30 16:20:10 IST 2020
2.

# New environment setting added by MF_LoadGenerator on Wed Jan 30
16:20:10 IST 2020 5.

# The unmodified version of this file is saved in
/etc/.login1557000131.

# Do NOT modify these lines; they are used to uninstall.
setenv M_LROOT "/opt/MF/MF_LoadGenerator"

# End comments by InstallAnywhere on Wed Jan 30 16:20:10 IST 2020
5.

# New environment setting added by MF_LoadGenerator on Wed Jan 30
16:20:10 IST 2020 8.

# The unmodified version of this file is saved in
/etc/.login1557000131.

# Do NOT modify these lines; they are used to uninstall.

if ( ! $?PATH ) then
setenv PATH ""
endif

setenv PATH "/opt/MF/MF_LoadGenerator/bin:${PATH}"

# End comments by InstallAnywhere on Wed Jan 30 16:20:10 IST 2020
8.
```

- 手动修改 **Bourne** 和 **Korn Shell** 用户的启动脚本

在 Load Generator 安装期间，安装向导将创建 **env.sh** 脚本。此脚本包括用于设置 Bourne Shell 和 Korn Shell 用户所需的环境变量的命令。

将以下行添加到 **/etc/profile** 或 **~/.profile** 启动脚本中，以在 Shell 启动期间执行 **env.sh** 脚本：

```
source <Load Generator 安装目录>/env.sh
```

例如：

```
source /opt/MF/MF_LoadGenerator/env.sh
```

对启动脚本进行如上修改的效果与由安装向导进行的修改类似。安装向导对 **/etc/profile** 启动脚本进行的修改示例如下所示：

```
# New environment setting added by MF_LoadGenerator on Fri Jan 16
11:14:24 IST 2020 1.

# The unmodified version of this file is saved in
/etc/profile1806316421.

# Do NOT modify these lines; they are used to uninstall.
PRODUCT_DIR=/opt/MF/MF_LoadGenerator
export PRODUCT_DIR

# End comments by InstallAnywhere on Fri Jan 16 11:14:24 IST 2020
1.

# New environment setting added by MF_LoadGenerator on Fri Jan 16
11:14:24 IST 2020 4.

# The unmodified version of this file is saved in
/etc/profile1806316421.

# Do NOT modify these lines; they are used to uninstall.
M_LROOT=/opt/MF/MF_LoadGenerator
export M_LROOT

# End comments by InstallAnywhere on Fri Jan 16 11:14:24 IST 2020
4.
```

```
# New environment setting added by MF_LoadGenerator on Fri Jan 16
11:14:24 IST 2020 7.

# The unmodified version of this file is saved in
/etc/profile1806316421.

# Do NOT modify these lines; they are used to uninstall.

PATH="/opt/MF/MF_LoadGenerator/bin:${PATH}"

export PATH

# End comments by InstallAnywhere on Fri Jan 16 11:14:24 IST 2020
7.LoadRunner settings #PATH=${M_LROOT}/bin:$PATH; export PATH
```

在 Linux 平台上安装 Load Generator 时出错

使用 **source installer.sh** 命令在之前已安装 Load Generator 的 Linux 计算机上安装 Load Generator 最新版本时，您可能会收到以下错误消息：

“尝试管理所选实例时发生错误。”

解决方案：

1. 打开注册表文件 **/var/.com.zerog.registry.xml**，并查找属性“**name**”=“**MF_LoadGenerator**”的元素“**product**”。

例如：

```
<product name="MF_LoadGenerator" id="77f695c1-1f0c-11b2-883d-
c486a85f6555" version="20.00.0.0" copyright="2019" info_
url="http://www.microfocus.com" support_url="http://www.microfocus.com"
location="/opt/MF/MF_LoadGenerator" last_modified="2019/11/21 13:12:14">
```

2. 记录“**location**”的属性值。
3. 删除“**location**”属性所引用的整个目录。
4. 删除注册表文件 **/var/.com.zerog.registry.xml**。
5. 重新运行 **source installer.sh** 命令。

在启用了 SELinux 的 RedHat Enterprise Linux 5.x 上运行 Load Generator 时出错

在 RHEL 5.x 上使用 Load Generator 期间，您可能会收到以下错误：

```
"m_agent_daemon: 加载共享库时出错: /opt/MF/MF_LoadGenerator/bin/liblwc_cryptolib.so: 在重定位后无法恢复网段端口: 权限被拒绝。"
```

发生此问题的原因是在计算机上安装并启用了 SELinux。SELinux 正在阻止加载指定的共享库。

解决方案：

有两种可能的解决方法：

1. 在使用 Load Generator 前，使用命令“setenforce 0”禁用 SELinux。
2. 如果要想 SELinux 保持启用状态，您可以更改所有问题库的安全性上下文(例如，将“<LoadGenerator 的路径>/bin/*.so”更改为“textrel_shlib_t”)。为此，请执行以下命令：“chcon -t textrel_shlib_t <LoadGenerator 的路径>/bin/*.so”

卸载 Load Generator 后，未取消设置环境变量

卸载 Linux Load Generator 时，安装向导可能没有取消设置当前 Shell 的 Load Generator 环境变量 (M_LROOT、PRODUCT_DIR 和 PATH)。要取消设置环境变量，请关闭当前 Shell 会话并调用新的 Shell 会话，或手动取消设置变量，如下所述：

- 要取消设置 M_LROOT 和 PRODUCT_DIR 变量，请执行以下操作：
 - [bash shells] 使用 **unset** 命令。
 - [csh shells] 使用 **unsetenv** 命令。
- 要更新 PATH 变量以排除 Load Generator 二进制目录，请键入：
 - [bash shells] PATH=<所需的路径列表>; export PATH
 - [csh shells] setenv PATH <所需的路径列表>

无法在 Load Generator 上运行 Vuser

如果无法在 Load Generator 上运行 Vuser，没有报告明确的错误，且 Vuser 协议要求 Load Generator 端显示第三方应用程序或客户端，请检查该应用程序所使用的动态库。这可用于确定是否找不到任何共享对象。找不到共享对象可能表示缺少必备程序包或存在环境变量问题。

要检查应用程序所使用的动态库，请键入：

```
ldd 应用程序名称
```

例如，键入 `ldd mdrv` 可确定是否可以找到 `mdrv` 可执行文件的所有依赖关系。如果未找到任何依赖关系，请运行 `verify_generator`，如[运行 verify_generator \(第 43 页\)](#)中所述。

注：如果针对要求安装客户端(如 Oracle)的协议运行 Vuser，请确保客户端库的路径包含在动态库路径环境变量(LD_LIBRARY_PATH 或 SHLIB_PATH)中。

安装 LoadRunner Data Hub

LoadRunner Data Hub 是一种基于 Kafka 技术的通信通道，可在 LoadRunner 组件之间实现更可靠的通信。要使用此通道，请安装 Data Hub。此外，如果您的 LoadRunner 组件通过 Web 进行通信，则需要安装 Data Hub Web Connector。有关详细信息，请参阅 [安装 Data Hub Web Connector \(第 61 页\)](#)。

Data Hub 必须安装在 Linux 上。它可以与在 Linux 和 Windows 计算机上运行的 LoadRunner 组件进行通信。

安装 Data Hub 后，需要配置 LoadRunner 组件才能使用它。有关详细信息，请参阅 LoadRunner Professional 帮助中心的 [Data Hub](#) 和 [Web Connector](#) 设置。

本章包括：

- [安装工作流](#) 56
- [使用安装向导安装 Data Hub](#) 57
- [静默 Data Hub 安装](#) 58
- [验证 Data Hub 安装](#) 60

安装工作流

按照以下主要步骤使用标准安装程序安装 Data Hub:

1. 查看系统要求

- 在安装 Data Hub 之前，请确保您的系统满足要求，详见 [支持列表](#)。
- Data Hub 的先决条件是需要安装最新版本的 Java JRE。
- 建议在运行安装之前关闭所有防病毒应用程序。

2. 从 [AppDelivery Marketplace](#) 下载 Data Hub 安装程序包。

3. 运行安装过程

- **向导安装：**使用 Data Hub 安装向导开始安装。有关详细信息，请参阅 [使用安装向导安装 Data Hub \(第 57 页\)](#)。
- **静默安装：**要执行静默安装，请参阅 [静默 Data Hub 安装 \(第 58 页\)](#)。

4. 验证安装

请参阅 [验证 Data Hub 安装 \(第 60 页\)](#)。

使用安装向导安装 Data Hub

本节介绍如何使用安装向导安装 LoadRunner Data Hub。

本节包括：

- [开始之前 \(第 57 页\)](#)
- [运行安装向导 \(第 57 页\)](#)

开始之前

安装期间，可以选择性地安装 CA 和 SSL 证书。这些证书分别用于进行身份验证和安全通信。这两个证书应为 *.cer (X.509) 格式。

- 对于 CA 证书：可以输入现有证书的路径，或留空跳过此步骤。如果要安装 CA 证书，应预先生成此证书。
- 对于 SSL 证书：可以从现有证书文件中选择此证书。或者，如果您提供了包含私钥的 CA 证书，则可以自动生成此证书。

有关生成 CA 证书的详细信息，请参阅[LoadRunner Professional 帮助中心](#) (选择相关版本)中有关创建和安装数字证书的章节。

有关在 LoadRunner Professional 中使用证书的详细信息，请参阅[LoadRunner Professional 帮助中心](#)中有关配置客户端-服务器身份验证的章节。

运行安装向导

要安装 **Data Hub**，请执行以下操作：

1. 切换为超级用户。
2. 解压缩 Data Hub 安装包，打开 **.bin** 文件所在的位置。
3. 启动安装向导：

对于：	输入：
[sh 和 bash shells]	./inst64.bin

4. 按照联机说明安装 Data Hub。

注: 默认安装路径为: `/opt/MF/MF_LoadRunner_Data_Hub/_MF_LoadRunner_Data_Hub_Installation`

5. 退出超级用户或切换到其他用户。

静默 Data Hub 安装

本节介绍如何执行 Data Hub 静默安装。

本节包括:

- [运行静默安装 \(第 58 页\)](#)
- [静默安装选项 \(第 60 页\)](#)

注: 您必须拥有超级用户权限。

运行静默安装

您可以按照以下步骤静默安装 Data Hub。

要执行 **Data Hub** 静默安装, 请执行以下操作:

1. 解压缩 Data Hub 安装包。
2. 使用以下模板创建一个名为 **installer.properties** 的文件, 并根据需要更改参数。

提示: **installer.properties** 文件会在您运行安装向导时生成, 并且可以再次用于以后的静默安装。

```
#Choose Install Folder
#-----
USER_INSTALL_DIR=/opt/MF/MF_LoadRunner_Data_Hub

#Choose Link Location
#-----
USER_SHORTCUTS=/usr/local/bin/MF/MF_LoadRunner_Data_Hub
```

```
#Install
#-----
-fileOverwrite_/opt/MF/MF_LoadRunner_Data_Hub/_MF_LoadRunner_
Data_Hub_Installation/Change_MF_LoadRunner_Data_Hub_
Installation.lax=Yes
-fileOverwrite_/opt/MF/MF_LoadRunner_Data_
Hub/zookeeper.service=Yes
-fileOverwrite_/opt/MF/MF_LoadRunner_Data_Hub/kafka.service=Yes
-fileOverwrite_/opt/MF/MF_LoadRunner_Data_Hub/run_after_
install.sh=Yes

#Install certificate
#-----
INSTALL_CERTIFICATE="\\", \"No\""
```

3. 将 **installer.properties** 文件保存在安装程序文件所在的位置。
4. 使用以下语法运行安装命令：

```
./inst64.bin [-f<安装程序属性文件的路径> | -options]
```

您可以通过向此命令添加选项来修改安装。有关详细信息，请参阅[静默安装选项 \(第 60 页\)](#)。

注：

- 当 **installer.properties** 文件与安装程序保存在同一目录中时，它会覆盖所有其他命令行选项，除非 **-f** 选项用于指定另一个有效的属性文件。
- 安装程序属性文件的路径可以是绝对路径，也可以是安装程序所在目录的相对路径。

例如，以下命令使用与安装程序保存在同一位置的 **installer.properties** 文件静默安装 Data Hub：



示例：./inst64.bin -i silent

静默安装选项

您可以将以下选项添加到静默安装命令：

选项	描述
-i [swing console silent]	指定安装程序的用户界面模式。
-D<名称>=<值>	指定安装程序属性。
-r <生成响应文件的路径>	生成响应文件。

修复和卸载选项

以下选项可用于修复或卸载 Data Hub：

选项	描述
-repair	修复安装。
-uninstall	卸载 Data Hub。

验证 Data Hub 安装

要验证 Data Hub 是否安装成功，请运行以下命令检查 zookeeper 和 kafka 服务的状态。如果这些服务正在运行，则安装成功。

```
service zookeeper status
```

```
service kafka status
```

安装 Data Hub Web Connector

Data Hub Web Connector 使 LoadRunner Data Hub 能够通过 Web 和防火墙进行通信。例如，如果 Load Generator 位于防火墙后面并通过代理服务器进行通信，则 Web Connector 使 Data Hub 能够连接到代理服务器并与 Load Generator 进行通信。

Web Connector 必须安装在 Linux 上。

安装 Web Connector 后，需要配置 LoadRunner 组件才能使用它。有关详细信息，请参阅 LoadRunner Professional 帮助中心的 [Data Hub](#) 和 [Web Connector 设置](#)。

本章包括：

- [安装工作流](#) 61
- [使用安装向导安装 Web Connector](#) 62
- [静默 Web Connector 安装](#) 63
- [验证 Web Connector 安装](#) 65

安装工作流

按照以下主要步骤使用标准安装程序安装 Web Connector:

1. 查看系统要求

- 在安装 Web Connector 之前，请确保您的系统满足要求，详见[支持列表](#)。
- 建议在运行安装之前关闭所有防病毒应用程序。

2. Web Connector 安装程序包包含在 Data Hub 安装程序包中。如果您尚未下载 Data Hub 安装程序包，请从 [AppDelivery Marketplace](#) 进行下载。

3. 运行安装过程

- **向导安装：**使用 Web Connector 安装向导开始安装。有关详细信息，请参阅[使用安装向导安装 Web Connector \(第 62 页\)](#)。
- **静默安装：**要执行静默安装，请参阅[静默 Web Connector 安装 \(第 63 页\)](#)。

4. 验证安装

请参阅[验证 Web Connector 安装 \(第 65 页\)](#)。

使用安装向导安装 Web Connector

本节介绍如何使用安装向导安装 Web Connector。

本节包括：

- [开始之前 \(第 62 页\)](#)
- [运行安装向导 \(第 62 页\)](#)

开始之前

安装期间，可以选择性地安装 CA 和 SSL 证书。这些证书分别用于进行身份验证和安全通信。这两个证书应为 *.cer (X.509) 格式。

- 对于 CA 证书：可以输入现有证书的路径，或留空跳过此步骤。如果要安装 CA 证书，应预先生成此证书。
- 对于 SSL 证书：可以从现有证书文件中选择此证书。或者，如果您提供了包含私钥的 CA 证书，则可以自动生成此证书。

有关生成 CA 证书的详细信息，请参阅[LoadRunner Professional 帮助中心](#) (选择相关版本)中有关创建和安装数字证书的章节。

有关在 LoadRunner Professional 中使用证书的详细信息，请参阅[LoadRunner Professional 帮助中心](#)中有关配置客户端-服务器身份验证的章节。

运行安装向导

要安装 Web Connector，请执行以下操作：

1. 切换为超级用户。
2. 解压缩 Web Connector 安装包，打开 .bin 文件所在的位置。

3. 启动安装向导:

对于:	输入:
[sh 和 bash shells]	./inst64.bin

4. 按照联机说明安装 Web Connector。

注: 默认安装路径为: `/opt/MF/MF_LoadRunner_Data_Hub_Web_Connector/_MF_LoadRunner_Data_Hub_Web_Connector_Installation`

5. 退出超级用户或切换到其他用户。

静默 Web Connector 安装

本节介绍如何执行 Web Connector 静默安装。

本节包括:

- [运行静默安装 \(第 63 页\)](#)
- [静默安装选项 \(第 65 页\)](#)

注: 您必须拥有超级用户权限。

运行静默安装

您可以按照以下步骤静默安装 Web Connector。

要执行 **Web Connector** 静默安装, 请执行以下操作:

1. 解压缩 Web Connector 安装包。
2. 使用以下模板创建一个名为 **installer.properties** 的文件, 并根据需要更改参数。此处的模板使用默认 LoadRunner 证书安装 Web Connector。

提示: **installer.properties** 文件会在您运行安装向导时生成, 并且可以再次用于以后的静默安装。

```
#Choose Install Folder
```

```
#-----
USER_INSTALL_DIR=/opt/MF/MF_LoadRunner_Data_Hub_Web_Connector
#Choose Link Location
#-----
USER_SHORTCUTS=/usr/local/bin/MF/MF_LoadRunner_Data_Hub_Web_
Connector
#Install
#-----
-fileOverwrite_/opt/MF/MF_LoadRunner_Data_Hub_Web_Connector/_MF_
LoadRunner_Data_Hub_Web_Connector_Installation/Change_MF_
LoadRunner_Data_Hub_Web_Connector_Installation.lax=Yes
-fileOverwrite_/opt/MF/MF_LoadRunner_Data_Hub_Web_
Connector/webconnector.service=Yes
-fileOverwrite_/opt/MF/MF_LoadRunner_Data_Hub_Web_Connector/run_
after_install.sh=Yes
-fileOverwrite_/opt/MF/MF_LoadRunner_Data_Hub_Web_
Connector/webconnector.sh=Yes
#Install certificate
#-----
INSTALL_CERTIFICATE=\"\", \"No\"
```

3. 将 **installer.properties** 文件保存在安装程序文件所在的位置。
4. 使用以下语法运行安装命令：

```
./inst64.bin [-f<安装程序属性文件的路径> | -options]
```

您可以通过向此命令添加选项来修改安装。有关详细信息，请参阅[静默安装选项 \(第 65 页\)](#)。

注：

- 当 **installer.properties** 文件与安装程序保存在同一目录中时，它会覆盖所有其他命令行选项，除非 **-f** 选项用于指定另一个有效的属性文件。
- 安装程序属性文件的路径可以是绝对路径，也可以是安装程序所在目录的相对路径。

例如，以下命令使用与安装程序保存在同一位置的 **installer.properties** 文件静默安装 Web Connector：



```
示例: ./inst64.bin -i silent
```

静默安装选项

您可以将以下选项添加到静默安装命令：

选项	描述
-i [swing console silent]	指定安装程序的用户界面模式。
-D<名称>=<值>	指定安装程序属性。
-r <生成响应文件的路径>	生成响应文件。

以下选项可用于修复或卸载 Web Connector：

选项	描述
-repair	修复安装。
-uninstall	卸载 Web Connector。

验证 Web Connector 安装

要验证 Web Connector 是否安装成功，请运行以下命令来检查 webconnector 服务的状态。如果此服务正在运行，则安装成功。

```
服务 webconnector 状态
```