MICRO FOCUS
is now **opentext**

LoadRunner Professional

Software Version: 2023

Installation Guide

Document release date: May 2023

Send Us Feedback

Let us know how we can improve your experience with the Installation Guide.
Send your email to: docteam@microfocus.com

Legal Notices

© Copyright 1994-2023 Micro Focus or one of its affiliates.

Contents

Welcome to LoadRunner Professional

Welcome to the LoadRunner Professional Installation Guide.

LoadRunner Professional, a tool for performance testing, stresses your application to isolate and identify potential client, network, and server bottlenecks.

This guide describes how to install and set up LoadRunner Professional.

LoadRunner Help Center

For context sensitive help within the LoadRunner Professional products, press F1 within a dialog box or use the Help menu.

You can access the product Help Centers on the web:

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

You can also download and install local versions of the help from the **Download Help Centers** page.

To switch between online and local modes, select **Help > Help Center Options > Open Online** or **Open Locally**.

> **Note:**
>
> - To view the Help Center in Internet Explorer or within VuGen, you need to enable JavaScript (Active scripting) in your browser settings (**Tools > Options > Security > Internet > Custom Level**).
> - If the online Help Center is unavailable at the start of a LoadRunner application session, the **Open Online** option may be disabled during the whole session with that application.

LoadRunner Professional components

LoadRunner Professional installation includes the following components. For more information about the components, refer to the LoadRunner Professional Help Center (select the relevant version):

- **Virtual User Generator (VuGen).** Micro Focus's tool for creating virtual user (Vuser) scripts, primarily through recording. Vuser scripts emulate users without a graphical user interface by using direct function calls.

- **Controller.** Controls the execution of scenarios and Vusers. Includes the online monitors which monitor and display information about the test execution. Controller must be installed on the computer used to control the Vusers.

- **Analysis.** Graphs and reports for analyzing the load test.

- **Load Generator.** Component for running Vusers (including Windows-based GUI Vusers) to generate load. The load generator can be installed on Windows or Linux platforms, using the installer or via Docker.

- **Data Hub.** Communication channel that supports more robust communication between LoadRunner components.

- **Data Hub Web Connector.** Enables the Data Hub to communicate via the web and over firewalls.

- **MI Listener Component.** Component for the MI Listener machine used in running Vusers and monitoring over the firewall.

- **Monitors over Firewall.** Component on the agent machine for monitoring over the firewall.

- **TruClient.** Component for recording and developing test scripts for web-based applications.

The LoadRunner Professional **Full Setup** installation package includes most of the above components. All of the standalone installations (for example, for VuGen or the load generator) can be found in the installation package's **Standalone Applications** folder.

Other components (such as the Citrix Agent) can be found in the installation package's **Additional Components** folder.

You can access the installation packages from the free trial page or by selecting your account on the Software Licenses and Downloads page.

You can also download VuGen, LoadRunner Developer, VTS, Data Hub, and more, for free from AppDelivery Marketplace.

System requirements and prerequisites

This chapter includes:

System requirements

The system requirements necessary for running LoadRunner Professional on a Windows system, or for running a load generator on a Windows or Linux system, are listed in the Support Matrix.

For system requirements and installation instructions for Network Virtualization, see the Network Virtualization for LoadRunner Help (select the relevant version) . You can install Network Virtualization during the LoadRunner Professional installation, or at a later time.

Prerequisite software for installation on Windows

Before installing LoadRunner Professional, make sure the prerequisite software is installed on the LoadRunner Professional machine. Some of the prerequisite software is automatically detected and installed by the installer; other prerequisite software must be installed manually.

- "Automatically detected and installed by the installer" on the next page

- "Manual installation for Windows updates" on the next page

Automatically detected and installed by the installer

When you run the installation wizard, the wizard detects which prerequisite software is already installed on the machine and which software is missing. When you continue with the installation, all missing prerequisite software is installed.

The following prerequisite software must be installed:

- .NET Framework 4.8

- Microsoft Visual C++ Redistributable for Visual Studio 2017-2022 (x86 and x64)

> **Note:** Installation of .NET Framework 4.8 prompts an automatic restart. You then need to rerun the LoadRunner Professional installation.

Manual installation for Windows updates

Before installing any of the LoadRunner Professional components, make sure that the full set of Windows updates has been installed. If this was not done automatically, you must install them manually.

> **Note:** Before installing any of the Windows updates, disable UAC (User Account Control) and restart the computer. For details on how to disable UAC, see the Microsoft Windows documentation.

You can find the list of Windows updates in the Support Matrix.

Installation notes and limitations

> **Note:** See **Known issues** in the VuGen or LoadRunner Professional Help Center for additional notes and limitations.

Security review	We strongly recommend that you install LoadRunner Professional components on dedicated machines that do not contain, or provide access to, sensitive information; and that you do a thorough security review of the network topology and access levels in your testing environment.
Permission requirements	The installation process for all LoadRunner Professional components requires a full administrator account (root account for Linux systems) for all operating systems. You can install LoadRunner Professional components with UAC and DEP enabled, but we recommend to keep them disabled during installation (or enable DEP for essential Windows programs and services only). After installation, all of the LoadRunner Professional applications and components can run under a standard user account (an administrator account is not required), with UAC and DEP enabled.
Running Web Services Vusers on Windows	The Axis toolkit and WSE are no longer automatically installed with LoadRunner Professional. When running Web Services Vusers on a standalone load generator, some legacy configurations may require activation of .NET 3.5 (see MSDN), the inclusion of the Axis java files, or an installation of WSE (2.0 SP3 and WSE 3.0). To avoid using these components, recreate the scripts and import the WSDL using the default WCF toolkit. To use these components: • For WSE, install the WSE components from the OneLG installation package's folders, **DVD\lrunner\Common\wse20sp3** and **DVD\lrunner\Common\wse30**, or download them from the Microsoft website. • For Axis, copy the Axis java files to **%LG_PATH%\bin\java\Axis**. For OneLG, the files are located in the **prerequisites\Axis** subfolder of the extracted installation package, by default **C:\Temp\Micro Focus LoadRunner OneLG <version>\prerequisites\Axis**.This allows you to run legacy scripts on an Axis toolkit. For more details, see Known issues for Web Services.

Running Java Vusers on Windows	OpenJDK 32-bit is no longer automatically installed with LoadRunner. To work with Java protocols, you need to manually install a supported version of the JDK. VuGen uses internal logic to automatically detect your JRE version. For details, see Set up the environment for Java protocols.
LoadRunner Professional and UFT coexistence	If you install LoadRunner Professional and UFT on the same machine, and then uninstall one of them, the remaining software program may not function correctly. **Resolution:** Run a Repair of the software program you want to use on the machine.
License	If upgrading from a version earlier than 2020, you need to replace your current license to work with the upgraded LoadRunner Professional installation. If you are using the Community license, it is automatically replaced during LoadRunner Professional installation. For any other license, contact licensing support to obtain a valid, compatible license. For more information, see the section on LoadRunner Professional Licenses in the LoadRunner Professional Help Center.
Custom certificates	When upgrading from version 12.55 or earlier, if you used custom certificates in your previous installation of LoadRunner, install them again when prompted for certificates during setup. Otherwise, the setup program will overwrite them using the defaults.
Network Virtualization	If you choose to install NV automatically (as part of LoadRunner Professional installation), you must disable Windows SmartScreen before proceeding with the NV installation: Open HKEY_LOCAL_ MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Expl orer in the Registry Editor, and change the Value data for **SmartScreenEnabled** to **Off**. **Note:** You do not need to disable SmartScreen when installing NV manually.

TruClient version conflict	After upgrading your LoadRunner Professional installation, if you encounter any version conflict, check that the legacy TruClient registry key **TcWebIELauncher.exe**, has been removed. • In the Registry Editor, open **HKEY_LOCAL_ MACHINE\SOFTWARE\Microsoft\Internet Explorer\MAIN\FeatureControl\FEATURE_BROWSER_ EMULATION** and delete the key. • Or, run the command: ```\n"REG DELETE "HKEY_LOCAL_\nMACHINE\SOFTWARE\Microsoft\Internet\nExplorer\MAIN\FeatureControl\FEATURE_BROWSER_\nEMULATION" /v "TcWebIELauncher.exe" /f"\n```
Virtual environments	The architectures provided by virtualization vendors are rapidly evolving. LoadRunner Professional is expected to function as designed in these changing environments, as long as the third-party vendor guarantees full compatibility of the virtualized environment with the LoadRunner Professional-approved hardware requirements. If you follow the system requirements and support matrix to create the virtual machine, LoadRunner Professional will work correctly. Working on top of a virtual machine may require access to the virtualization server hardware/monitoring environment, to ensure the virtualization server is not saturated; otherwise, this might obscure the virtual machine's measurements and lead to false results.

Installing LoadRunner Professional on Windows

This chapter describes how to install either the full version of LoadRunner Professional or a LoadRunner Professional component on a Windows platform.

This chapter includes:

Installation workflow

Your LoadRunner Professional installation package includes the Setup wizard that guides you through the process of installing the LoadRunner Professional components.

The installation components include **OneLG**, which is a combined load generator installer for all LoadRunner family products. You can install the load generator using the installer (UI based or silent), or through a Docker container. For more information, see "Deploy Dockerized load generators on Windows" on page 29.

> **Note:** You can also provision a load generator in a cloud account. For details, see the section on load generators on the cloud in the LoadRunner Professional Help Center (select the relevant version).

The final stage of the installation includes the optional installation of Network Virtualization. Installing Network Virtualization enables you to generate the NV Insights Report (replaces the NV Analytics report) from LoadRunner

Professional. The NV Insights Report provides an in-depth analysis of how your application performs with different loads over various types of networks (virtual locations) and includes suggestions for optimizing your application's performance.

Installing LoadRunner Professional on a Windows system involves the steps shown below:
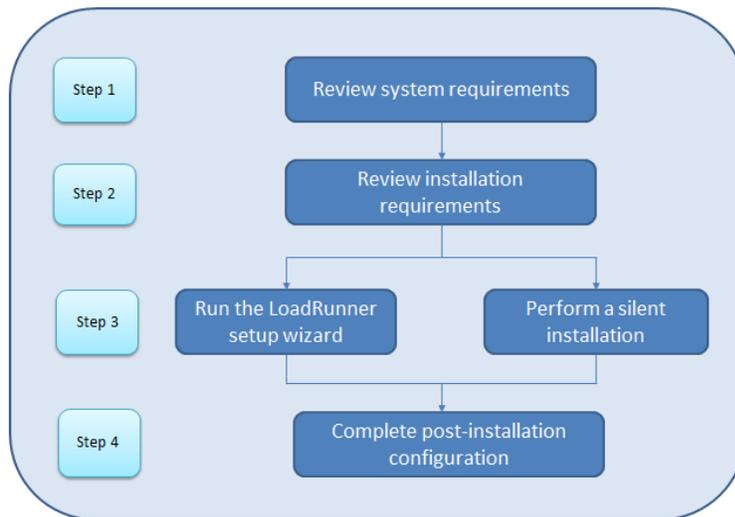
1. **Review system requirements and notes**

   - Make sure that your system meets the hardware and software requirements. For details, see "System requirements" on page 8.

     > **Prerequisite software:** When you run the LoadRunner Professional Setup wizard, the wizard detects which prerequisite software is already installed on the machine and which software is missing. When you continue with the installation, all missing prerequisite software is installed. For details, see "Prerequisite software for installation on Windows" on page 8.

   - Review the "Installation notes and limitations" on page 9.

2. **Review installation requirements**

   Before you begin the installation, make sure you meet the following installation requirements:

- You must have full local administrative rights on the designated machine.

- Installation must be performed at the destination machine. LoadRunner Professional does not support installation via terminal service.

- For the full list of components that can be installed on the same physical machine at the same time, see the Support Matrix.

> **Note:** It is recommended that you close all antivirus applications before installing LoadRunner Professional.

3. **Perform the installation**

   a. If you were working with an earlier LoadRunner Professional version, review the upgrade instructions. For details, see "Upgrade LoadRunner Professional" on the next page.

   b. Run the Setup wizard to install the full version of LoadRunner Professional, LoadRunner Professional standalone components, or additional components on a Windows system. For details, see "Install LoadRunner Professional on Windows" on the next page.

      Then run any patch installations if necessary.

   c. After you complete the LoadRunner Professional installation, you can install a localized version to view the LoadRunner Professional, VuGen Standalone, and Analysis Standalone user interface in your local language. For details, see "Install a LoadRunner language pack" on page 35.

   > **Note:**
   > - To perform a silent installation, see "Install LoadRunner Professional silently" on page 22.
   > - VuGen and load generator installation on Windows includes OpenJDK 11, 32-bit.

4. **Complete post-installation configuration**

   - Configure LoadRunner Professional to run Vusers on a load generator machine without the need to log on manually. For details, see "Configure user login settings" on page 34.

- To run Vusers with Controller there must be a valid license installed on the Controller machine. LoadRunner Professional automatically installs the Community license bundle (free of charge) during LoadRunner Professional setup, providing 50 Vusers.

  For details on managing LoadRunner Professional licenses and extending your Vuser capacity, see the licensing information in the LoadRunner Professional Help Center (select the relevant version).

Upgrade LoadRunner Professional

LoadRunner Professional is a full installation that can be installed over any LoadRunner or LoadRunner Professional installation (version 12.5x and later).

To upgrade all components in your Windows installation, follow the installation process as described in "Install LoadRunner Professional on Windows" below. The installation process detects the older version, and gives you the option to upgrade.

> **Note:** For silent upgrade, see "Install LoadRunner Professional silently" on page 22.

Install LoadRunner Professional on Windows

This section describes how to install LoadRunner Professional on Windows using the Setup wizard.

For details on how to perform a silent installation, see "Install LoadRunner Professional silently" on page 22.

Setup wizard information

Review the following information to help you successfully install your LoadRunner Professional component using the Setup wizard. For more information on each

component, refer to the LoadRunner Professional Help Center (select the relevant version).

LoadRunner Agent	Select the **Start LoadRunner Agent after installation** option to start the LoadRunner Agent on a load generator immediately after installation. The agent enables communication between the load generator and Controller.

Agent mode	During the installation of the OneLG load generator, MI Listener, or Monitors over Firewall components, you are prompted to select the mode for the installed LoadRunner Agent:
	
	• **LoadRunner Professional** mode. The agent runs as a regular process, requiring log in before the process can start.
	  Note: Select this mode if you are installing a load generator for use with Business Process Monitoring (BPM).
	• **LoadRunner Enterprise** mode. The agent runs as a service, and is launched automatically when the operating system starts.
	• **LoadRunner Cloud** mode. This is relevant for OneLG installation only. Select if you want to use the load generator with LoadRunner Cloud.
	  Note: Agent configuration can be changed after installation, if required.

Silk Performer Agent	Select the **Install Silk Performer Agent after installation** option to install the Silk Performer Agent on a load generator that will be used to run Silk Performer scripts.

Certificates	During installation of LoadRunner Professional Full Setup, OneLG, Monitor over Firewall, and MI Listener, you have the option to install CA and TLS (SSL) certificates by selecting the **Specify a certificate that will be used by the LoadRunner Agent** option.
	These certificates are used for authentication and secure communication. Both certificates must be in *.cer (X.509) format.
	• **CA certificate**: Enter the path to an existing certificate, or leave it blank to skip the step. The CA certificate should be generated in advance.
	• **TLS certificate**: Select from an existing certificate file. Alternatively, the certificate can be generated automatically if you provide the CA certificate containing the private key.

Running the installation

The following procedure guides you through the Windows installation process.

To install LoadRunner Professional or an additional component:

1. Before installing:

   • Review the pre-installation information, including the system requirements and prerequisite software, described in "System requirements and prerequisites" on page 8.

   • If you plan to install Network Virtualization, make sure that your system meets the requirements, as described in the Network Virtualization for LoadRunner Help (select the relevant version). (You can install Network Virtualization during LoadRunner Professional installation, or at a later time.)

   > **Note:** If you install Network Virtualization as part of LoadRunner Professional installation, you may need to restart the machine after setup.

   • Make sure you have full, local administrative rights on the designated machine.

   • Make sure the Windows Update process is not running.

2. UAC (User Account Control): You can install with UAC enabled on the installation machine, but we recommend that you disable it, then reboot the machine. For details, refer to your Microsoft Windows documentation.

3. DEP (Data Execution Prevention): You can install with DEP enabled, but we recommend that you disable it, or enable it for essential Windows programs and services only. For details on how to change DEP settings, refer to your Microsoft Windows documentation.

4. Run the **setup.exe** file in the root folder of the installation package.

   The LoadRunner Professional installation program begins and displays the installation options.

5. Select the required installation option.

   From the installation menu page, select one of the following installation options:

   - **LoadRunner Professional Full Setup.** Installs the main LoadRunner Professional components, including Controller, Virtual User Generator (VuGen), Analysis, OneLG Load Generator, and TruClient. Use this option for the machine that runs the load testing scenarios.

     > **Note:** The LoadRunner Professional Setup wizard can install either LoadRunner Professional or LoadRunner Enterprise. Make sure to select **LoadRunner Professional**.

     For details on the components that are included in the full installation, see"LoadRunner Professional components" on page 6.

   - **VuGen.** Installs a standalone version of VuGen (includes TruClient).

   - **Analysis.** Installs a standalone version of LoadRunner Analysis.

   - **OneLG.** Installs the standalone load generator component, needed for running Vusers to generate load. Use this option for machines that are used to generate load only, and not to control Vusers.

   - **Monitors Over Firewall.** Installs the components on the agent machine for monitoring over the firewall. For more information, refer to the **Working with Firewalls** section in the LoadRunner Professional Help Center.

- **MI Listener.** Installs the components needed on the MI Listener machine used in running Vusers over a firewall and monitoring over a firewall. For more information, refer to the **Working with Firewalls** section in the LoadRunner Professional Help Center.

- **Language Packs.** This option is available only when installing on native non-English operating systems. Enables you to install a patch for localized LoadRunner Professional user interface, on top of the English version. For details, see "Install a LoadRunner language pack" on page 35.

- **Additional Components.** Opens the Additional Components folder located in the root folder of the installation package. For details of the additional components that you can install, see the **Advanced > Additional Components** section in the LoadRunner Professional Help Center .

6. Specific software, for example, Microsoft Visual C++, needs to be installed before you can install LoadRunner Professional (see "Prerequisite software for installation on Windows" on page 8). If the prerequisite software is not already installed on your computer, a dialog box opens displaying the list of prerequisite programs that are required.

   Click **OK** to install the listed software before continuing with the LoadRunner Professional installation. If you click **Cancel**, the LoadRunner Professional Setup wizard exits because LoadRunner Professional cannot be installed without the prerequisite software.

   > **Note:** After installing prerequisite software, LoadRunner Professional may restart the machine. If that occurs, rerun **setup.exe**.

7. Perform the installation.

   The Setup wizard opens, displaying the Welcome page.

   Follow the instructions in the wizard to complete the installation. Refer to the information in "Setup wizard information" on page 16 to ensure a successful installation.

   > **Note:**

> - The installation path for LoadRunner Professional or LoadRunner Professional components cannot contain non-English characters.
> - The default installation path is: **C:\Program Files (x86)\Micro Focus\LoadRunner**

8. After the installation is complete, you can choose to install Network Virtualization (NV).

   In the Setup wizard, select:

   - **Typical mode** for an automatic installation.

   - **Custom mode** to change the installation folder, data folder, and the port.

   For more details, see the installation section of the Network Virtualization for LoadRunner Help (select the relevant version).

9. If there is a patch available for the installed version:

   a. Run the patch installation file provided with the installation package.

   b. Follow the onscreen installation and deployment instructions.

> **Note:**
>
> - To repair the installation, run the **setup.exe** file located in the root directory of the LoadRunner Professional installation package, select **LoadRunner Professional Full Setup**, and then select the **Repair** option in the Setup wizard.
>
> - You can configure LoadRunner Professional to run Vusers on a load generator machine without the need for the user to manually log in to the machine. For more information, see "Configure user login settings" on page 34.
>
> - For a list of the components that were registered during setup, such as **DLL** and **OCX** files, see the **RegisteredComponents_Micro Focus LoadRunner.txt** file in the **build_info** folder of the installation.

Install LoadRunner Professional silently

A silent installation is an installation that is performed without the need for user interaction. You use the command line to run the setup files. For details, see "Installation command line options" on page 24.

> **Note:**
> - Before you install LoadRunner Professional, review the pre-installation information, including the system requirements and prerequisite software, described in "System requirements and prerequisites" on page 8.
> - All machines on which you install LoadRunner Professional require administrator privileges.
> - Use standard MSI command line options to define installation properties. For example, use INSTALLDIR to specify an alternate installation folder.
> - To prevent the LoadRunner Agent on the load generator from starting immediately after installation, add the following to the command line command: `START_LGA=0`. The agent enables communication between the load generator and Controller. For more details on the LoadRunner Agent, see the LoadRunner Professional Help Center (select the relevant version).

To perform a silent installation of LoadRunner Professional

1. Run one of the following commands from the command line:
   - To install all of the LoadRunner Professional components including the prerequisite software in a single command:

   ```
   <Installation_package>\lrunner\<your_language_
   folder>\setup.exe /s
   ```

   - To first install all of the prerequisites, and then LoadRunner Professional, run the following commands:

   ```
   <Installation_package>\lrunner\<language_folder>\setup.exe
   /InstallOnlyPrerequisite /s
   ```

```
msiexec.exe /qn /i "<Installation_
package>\lrunner\MSI\LoadRunner_x64.msi"
```

- By default, Network Virtualization is installed during the LoadRunner Professional installation. To facilitate this installation, add the following to the installation command:

```
REBOOT_IF_NEED=1
```

- To prevent the installation program from installing Network Virtualization, add the following to the installation command:

```
NVINSTALL=N
```

- To secure your environment with TLS certificates and/or two-way TLS for LoadRunner products, OneLG, MiListener, or MoFW, add the following to the installation command as relevant:

```
CACERT = <path to the root certificate in *.cer format>
CERT = <path to the server certificate in *.cer format>
CACERT_PK = <the CA certificate private key>
CACERT_PK_PWD = <password for the CA certificate encrypted
private key>
CERT_PK = <the TLS certificate private key>
CERT_PK_PWD = <password for the TLS certificate encrypted
private key>
TWO_WAY_SSL = High | Medium
```

The following command performs a silent installation of MI Listener.

> **Example:** SetupMIListener.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

For details, see the information on secure communication with TLS in the [LoadRunner Professional Help Center](#) (select the relevant version).

- To install a LoadRunner Professional standalone application (not VuGen or OneLG):

```
<Installation_package>\Standalone
Applications\Setup<component_name>.exe /s /a /s
```

- To install VuGen:

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

- To install additional components:

```
<Installation_package>\Additional Components\<setup_file_path>
/s /a /s
```

> **Tip:** See additional examples here: ["Examples " on page 28](#)

2. If you are installing an upgrade, run the following command:

```
msiexec.exe /update <full path to msp file> [/qn] [/l*vx <full
path to log file>]
```

The msp files are located in the installation package.

For details, see ["Upgrade LoadRunner Professional" on page 16](#).

Installation command line options

You can use the Windows command line to install the full LoadRunner Professional package, standalone applications, and additional components. The CLI uses Wrapper file (setup.exe) or Package for the web files (<PFTW>.exe).

- The full LoadRunner Professional installation can be launched using the setup.exe file. For details, see ["Installing the full LoadRunner Professional](#)

package from the wrapper file" on the next page.

- The installations of standalone applications and additional components can be launched via PFTW files. For details, see .

Installing the full LoadRunner Professional package from the wrapper file

You can install full LoadRunner Professional from the installation wrapper file, **setup.exe**, located in **\lrunner\<your_language>** folder of your LoadRunner Professional installation package. The following command line options are available:

Option	Description
/s	Runs the installation in the background (silently), with no user interaction.
/qb	Runs the installation in unattended mode, with limited user interaction.
/InstallOnlyPrerequisite	Installs only the prerequisites – does not install any LoadRunner Professional components. By default, the setup program checks that your machine has the required prerequisites, and installs them if necessary, before installing the LoadRunner Professional components.

You can set public properties for the command line installation, using the following syntax:

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

The following properties are available:

Property name	Description
INSTALLDIR="*your_path*"	Specifies the location where the application will be installed.

Property name	Description
REBOOT_IF_NEED	1: Reboots the machine after installation, if required. This is recommended if you include the installation of NV (Network Virtualization). 0: Does not reboot the machine after installation. Default: 0
NVINSTALL	Empty string: Excludes the installation of the NV component. Y: Includes the installation of the NV component. Default: Y
INSTALL_NV_MODE	1: Sets the NV installation mode to **Typical**. 2: Sets the NV installation mode to **Custom**. Default: Typical
INSTALL_DIR="*your_path*"	Specifies a path for the installation.
START_LGA	Empty string: Instructs the machine not to start the load generator after installation. 1: Instructs the machine to start the load generator after installation. Default: 1
IS_RUNAS_SERVICE	0: LoadRunner Professional mode, runs the load generator's agent as a process. 1: LoadRunner Enterprise mode, runs the load generator's agent as a service. 2: (Relevant for OneLG only) LoadRunner Cloud mode, runs the load generator's agent as a service, and triggers launch of LRC Agent configuration at the end of the installation. Default: 1
INSTALL_SILK_ PERFORMER_AGENT	1: Installs the Silk Performer Agent on load generators (used to run Silk Performer scripts).

Property name	Description
SKIP_CHECK_PRIVILEGE	0: The installation process checks the install path for access privileges.
	1: The installation process skips the access check for the install path.
	Default: 0
IMPROVEMENTPROGRAM	0: Disables the LoadRunner Improvement Program.
	1: Enables the LoadRunner Improvement Program.
	Default: 1

Example:

The following command performs the installation silently and then reboots the machine if necessary:

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

Installing a standalone application or an additional component from a PFTW file

You can install a LoadRunner Professional standalone application or one of the additional components via the Package for the web (PFTW) files that are included with the LoadRunner Professional installation media. The installation files are located under the **Standalone Applications** or **Additional Components** folders on the installation media.

Double-click on the packaged file, or run it from the command line using the following options:

Option	Description
/s	Runs the installation in the background (silently), with no user interaction.

Option	Description
/e	Only extracts the installation files; does not run them.
/f	Specifies the path of a temporary folder for file extraction. For example: `/f "c:\my_temp_folder"` If you leave out this option, the default temporary folder is used.
/a	Allows you to pass parameters or properties to the autorun file, such as setup.exe. Use the properties defined above (in the wrapper file section) for the setup.exe file.

> **Note:** Silent installation for VuGen and OneLG:
>
> - **SetupVuGen.exe** and **SetupOneLG.exe** uses different install commands (see the examples below).
> - If you need to insert a quotation mark character into the command line parameter, use two consecutive quotation marks.

Examples

The following commands silently run setup, and install the application in the specified folder. Where relevant, they also start the agent after installation:

- Examples for **SetupVuGen.exe** and **SetupOneLG.exe**:

```
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"
```

- Examples for **other standalone applications**:

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

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

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

Deploy Dockerized load generators on Windows

This section describes how to run a Dockerized load generator on a Windows platform.

Docker is a platform that allows you to develop, ship, and run applications via a container.

> **Note:**
>
> - For supported protocols on Dockerized load generators, see the Supported Protocols guide.
> - FTP is not fully supported with load generator for Docker on Windows. Passive mode FTP is supported, but Active mode FTP is not.

Prerequisites

> **Note:** The Docker image for the OneLG load generator replaces the previous Windows standalone load generator docker image.

- Install Docker on the target machine, along with its dependencies, and set up the target machine environment as required. Currently, only the 64-bit version is supported. For installation details, see the Docker online documentation.

- Pull the Windows load generator Docker image from the from the relevant page, accessible from the performance testing page (https://hub.docker.com/u/performancetesting) in the Docker hub. Use the following command and appropriate **<tag version number>**, for example, 20.03:

```
docker pull performancetesting/microfocus_onelg_windows:<tag
version number>
```

Run a Dockerized load generator using the predefined image

Use the ready-to-use image to run a load generator (OneLG) on Docker for Windows.

> **Note:** If you need customization for your container, for example, for Java or to run under a specific user, see "Run a Dockerized load generator using a custom image" below.

To run a Dockerized load generator:

Run the load generator container using the following command:

```
docker run -id -p <host_port>:54345 performancetesting/microfocus_
onelg_windows:<tag version number>
```

> **Note:** Check that the <host_port> on the machine is available and allows incoming requests. You specify this port on the Controller side when connecting to this load generator.

Run a Dockerized load generator using a custom image

If your environment requires customized settings for running the container, you can create a Dockerfile to build a custom image for Docker on Windows.

Examples for custom images:

- To use a specific user account for the processes under which the Vusers are running, to provide support for accessing network resources like script parameter files. After running, the container should be able to verify the user.

- To run Java protocols on Windows load generator containers.

- To define environment variables for proxy server host and port.

To run a custom Dockerized load generator:

1. Create a new folder, and within it create a file named **dockerfile**. Paste the following **FROM** line into the file, using the appropriate LoadRunner Professional version for the **<tag version number>**, and add the relevant customization lines:

   ```
   FROM performancetesting/microfocus_onelg_windows:<tag version
   number>
   <Customization lines>
   ```

   For customization examples, see "Examples of customized content for Dockerfiles" on the next page

   > **Tip:** For information on commands that can be used in Docker files, see the Docker online documentation.

2. Save the Dockerfile.

3. Open a command line at the **dockerfile** folder path and run the following command, using the name you want for your custom image:

   ```
   docker build -t <custom image name> .
   ```

4. Create a container for each load generator you want to use, by running the following command (or use any Docker orchestrator tool for running containers):

   ```
   docker run -id -p <host_port>:54345 <custom image name>
   ```

   If the custom image in step 3 was built with a tag then include it in the command:

   ```
   docker run -id -p <host_port>:54345 <custom image name>:<tag
   version number>
   ```

> **Note:** Check that the <host_port> on the machine is available and allows incoming requests. You specify this port on the Controller side when connecting to this load generator.

Examples of customized content for Dockerfiles

Example for Vusers under a specified user account

The following gives an example of dockerfile content for running the Vusers under a specified user account with network access to shared locations. Replace the values between **<>** with credentials for a valid user account in your environment, with network access to the shared resources.

```
Example:
#escape=`
FROM performancetesting/microfocus_onelg_windows:<tag version
number>
RUN c:\LG\launch_service\bin\magentservice.exe -remove
RUN c:\LG\launch_service\bin\magentservice -install
<domain>\<user name> <password>
```

Example for running Java/JMeter/Gatling protocols

The following gives an example of dockerfile content to run Java, JMeter, or Gatling protocols:

```
Example:
#escape=`
FROM performancetesting/microfocus_onelg_windows:<tag version
number>
COPY .\<folder contains JDK> <target path in the container>
```

- The path to the target JDK directory defined in the **COPY** line for the **<target path in the container>** must also be added to the **Java VM** runtime settings

page:
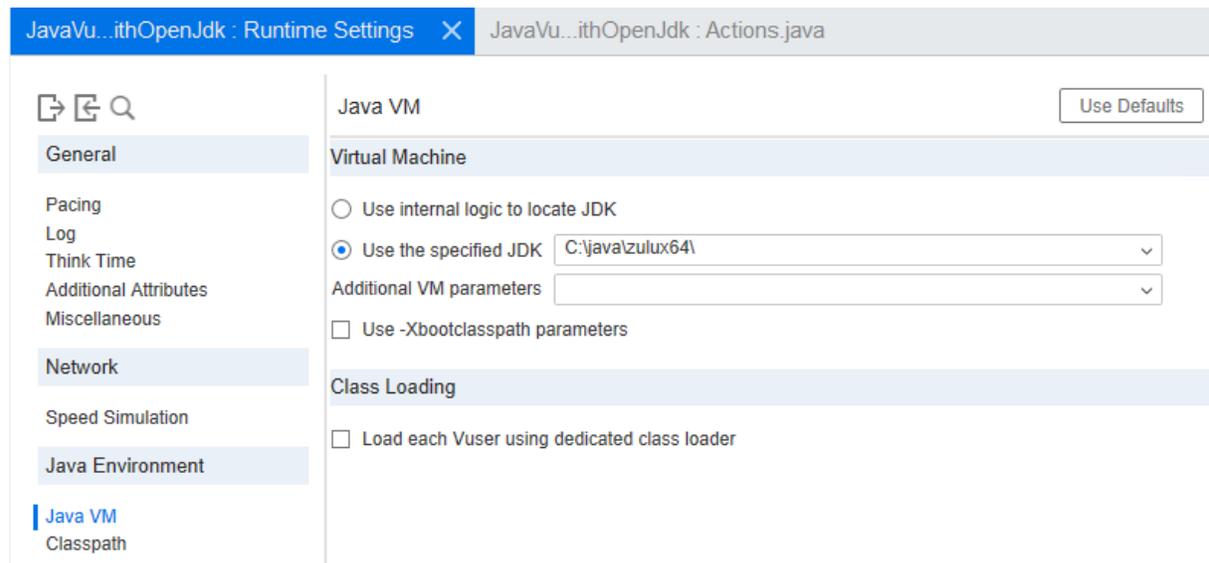
- For **Java 64-bit protocol** testing, include the following command line in the dockerfile, in order to add the path to the JDK 64-bit **bin** folder to the machine PATH environment variable:

```
RUN powershell [Environment]::SetEnvironmentVariable(\"Path\",
$env:Path + \";<target JDK path in the container>\bin\",
[EnvironmentVariableTarget]::Machine)
```

- For **JMeter/Gatling protocol** testing, include the following command line in the dockerfile, in order to add the path for the JDK/JRE to the machine JAVA_HOME environment variable:

```
RUN powershell [Environment]::SetEnvironmentVariable(\"JAVA_
HOME\", \"<target JDK/JRE path in the container>\",
[EnvironmentVariableTarget]::Machine)
```

After running the load generator containers

Add the load generators containers to scenarios. For details, see the information on configuring Dockerized load generators in the LoadRunner Professional Help Center (select the relevant version).

Tips and guidelines

- Dockerized load generators, run from the predefined image, are not supported when running over a firewall.

- Use `docker ps` to list the containers that are running.

- To stop the load generator service:

  - Use `docker stop <load generator container name or ID>` if you want to reuse the same load generator.

  - Use `docker rm -f <load generator container name or ID>` in order to remove the load generator container.

- To access the host network directly, use `--net=host` in place of `-p <host_port>:54345`. We recommend you use this flag if the AUT generates a lot of network activity.

Configure user login settings

By default, you need to manually log on to a computer before LoadRunner Professional can run Vusers on that computer. However, you can configure LoadRunner Professional to run Vusers on a load generator machine, without the need to manually log on to the machine.

To configure user login settings:

1. In the Windows Start menu, search for **LoadRunner Agent** and select the **LoadRunner Agent Runtime Settings Configuration** item.

   The LoadRunner Agent Runtime Settings dialog box opens.

2. Select one of the following options:

   - **Allow virtual users to run on this machine without user login.** LoadRunner Professional automatically logs on to the network from the load generator machine, so the Vusers can run without any manual intervention. Enter the network domain where the user machine resides, a user name, and password.

> **Note:** When created, the LoadRunner Agent service starts with the **LocalSystem** account (not as a specified user). The specified credentials are used by the agent service to start the *mdrv.exe* process when you run the script.

- **Manual log in to this machine.** The user must manually log on to the network from the load generator machine for each session of running Vusers.

3. Click **OK**.

> **Note:** You must reboot and log on to the system at least once after the LoadRunner Professional installation before the automatic login can work.

Install a LoadRunner language pack

The language packs enable you to view the LoadRunner Professional user interface in your local language. You install the language pack for the relevant LoadRunner Professional component: LoadRunner Professional full, VuGen standalone, or Analysis standalone from the LoadRunner Professional installation package.

For a list of supported languages, see the Support Matrix.

The language packs are supported for native non-English operating systems only.

> **Important:** The native language of the operating system must be the same as the language pack you are installing. For example, the Spanish language pack must be installed on a machine with a native Spanish operating system.

To install a language pack:

1. Make sure that LoadRunner Professional in English is already installed.

2. In the root folder of the LoadRunner Professional installation package, run **setup.exe**. The LoadRunner Professional installation program begins and displays the installation options.

3. Click **Language Packs**. The Language Packs folder in the installation package opens.

4. Navigate to the folder for the language and component you want to install and run the installation file. For example, to install the French language pack on a VuGen standalone machine, the path is **..\Language Packs\French\VugenSA\Vugen_FRA.msp**.

   Follow the online instructions.

Installing the load generator on Linux

LoadRunner Professional uses load generators to run Vusers. There are two versions of the LoadRunner Professional load generator. One version runs Vusers on Windows platforms, and the other version runs Vusers on Linux platforms. You use Windows-based LoadRunner Controller to control both the Windows-based and the Linux-based Vusers.

This chapter describes how to install the load generator on a Linux platform. For details on how to install the load generator on a Windows platform, see "Installing LoadRunner Professional on Windows" on page 13.

> **Note:**
>
> - You can install the Linux load generator using the installer (UI based or silent), as described in this section, or through a Docker container—for more information, see "Deploy Dockerized load generators on Linux" on page 43.
> - For troubleshooting information, see "Troubleshoot the load generator Linux installation" on page 58.

This chapter includes:

Installation workflow

You can install the load generator component on a Linux platform to run Vusers. Linux-based Vusers interact with LoadRunner Controller, installed on a Windows platform.

The load generator can be installed on a local Linux machine, or provisioned in a cloud account. See Load generators on the cloud in the LoadRunner Professional Help Center for information on managing load generators on the cloud.

The following diagram shows the primary steps in installing the load generator on a Linux platform using the standard installation program:
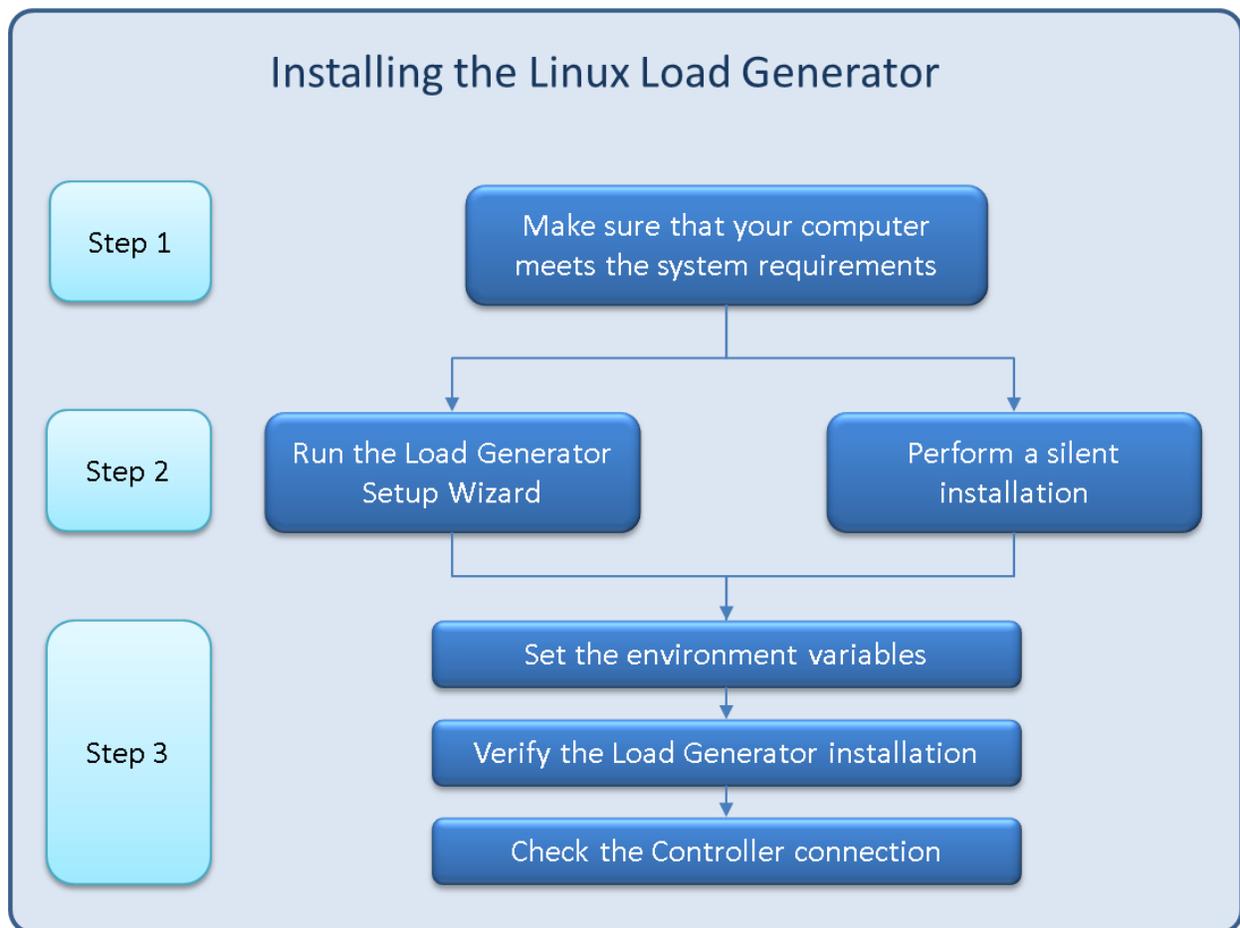
1. **Review the system requirements**

   - Before you install the load generator on a Linux platform, make sure that your system meets the requirements, as described in "System requirements and prerequisites" on page 8.

   - If you plan to install **Network Virtualization**: Make sure that your system meets the requirements, as described in the Network Virtualization for LoadRunner Help (select the relevant version).

- If you are using a **RHEL 8** operating system: By default, libnsl.so.1 is not installed with RHEL 8.0, so you need to manually install it before installing the load generator, or installation will fail. Install using the package manager: `yum install libnsl.i686`

- If you plan to replay **ODBC Protocol** scripts on the Linux machines: Install unixODBC v.2.3.1 or higher.

2. **Run the setup**

> **Note:**
>
> - The load generator for this version can be installed over any load generator installation from version 12.56 and later.
> - If load generator version 12.55 or earlier is installed, you must first uninstall it. For details, see "Uninstall load generator 12.55 or earlier" on page 57.
> - It is recommended that you close all antivirus applications before installing LoadRunner Professional.

- **Wizard installation:** Start the setup using the Load Generator Setup wizard. For more information, see "Install a load generator on Linux with the Setup wizard" on the next page.

- **Silent installation:** To perform a silent installation, see "Silent load generator installation on Linux" on page 42.

> **Note:** The load generator installation on Linux includes OpenJDK 8, 32-bit .

3. **Configure your environment**

After the installation, before you can begin working with the load generator you need to configure your environment. This involves setting the appropriate environment variables, checking access to the load generator, and verifying the installation. See "Configure the Linux environment" on page 49.

Install a load generator on Linux with the Setup wizard

This section describes how to install the load generator on a Linux platform using the Setup wizard.

> **Note:** To perform the installation through a Docker container, see "Deploy Dockerized load generators on Linux" on page 43.
>
> To perform a silent installation, see "Silent load generator installation on Linux" on page 42.

This section includes:

- "Before you start" below
- "Run the Setup wizard" on the next page

Before you start

During installation, you can optionally install CA and SSL certificates for the load generator. These certificates are used for authentication and secure communication respectively. Both certificates should be in *.cer (X.509) format.

- For a CA certificate: You can enter the path to an existing certificate, or leave the path blank to skip the step. If you want to install the CA certificate, it should be generated in advance.
- For an SSL certificate: You can select it from an existing certificate file. Alternatively, it can be generated automatically if you provide the CA certificate containing the private key.

For details on generating a CA certificate, see the section about creating and installing digital certificates in the LoadRunner Professional Help Center (select the relevant version).

For more information on working with certificates in LoadRunner Professional, see the section about configuring client-server authentication in the LoadRunner Professional Help Center.

Run the Setup wizard

To install the load generator:

1. Switch to super user.

2. If Load Generator version 12.55 or earlier is installed on the machine, you must uninstall it manually. For details, see "Uninstall load generator 12.55 or earlier" on page 57.

   If you do not uninstall the load generator, you will get an error during the installation.

   > **Tip:** If Load Generator 12.56 or later is installed, the Setup wizard uninstalls it during the installation process, then reinstalls it.

3. Unpack the load generator installation package and open the location of the **.bin** file.

4. Launch the Setup wizard:

For:	Type:
[sh and bash shells]	`source ./installer.sh`
[csh and tcsh shells]	`source ./installer.csh`
If the source command is not supported by the current shell, use the "dot" command.	For example: `. ./installer.csh`

   > **Note:** It is recommended that you use the **source** command to run the Setup wizard, as shown above. If you run the Setup wizard without using this command, you must manually set the environment variables for the

> current shell session. For details, see "Set the environment variables" on page 49.

The Load Generator Setup wizard checks that the required prerequisite software is installed on the computer. If any prerequisite software is missing, a message is displayed, and the Setup wizard aborts. Install the required package. Then rerun the Setup wizard.

5. Follow the online instructions to install the load generator. For more help during installation, see "Troubleshoot the load generator Linux installation" on page 58.

> **Note:** The default installation path is: **/opt/MF/MF_LoadGenerator/_MF_LoadGenerator_Installation**

6. Exit super user or switch to another user.

7. Configure your environment as described in "Configure the Linux environment" on page 49.

Silent load generator installation on Linux

This section describes how to perform a silent installation of the load generator on a Linux machine.

> **Note:** You must have super user privileges.

To perform a silent installation of the load generator:

1. If a load generator is installed on the machine, you must first uninstall it, otherwise the silent installation will fail. To uninstall the load generator, see "Uninstall the load generator from a Linux machine" on page 56.

2. Change directory to /`<path_to_installer_dvd>/load_generator_linux_ x64/`.

3. Run the following command to silently install the load generator:

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

> **Note:** It is recommended that you use a **source** command to perform a silent installation, as shown above. If you perform the installation without using the **source** command, you will need to set up the environmental variables after installing the load generator. For details, see "Set the environment variables" on page 49.
>
> If the **source** command is not supported by the current shell, use the "dot" command. For example, `. ./installer.sh - i silent`.
>
> If you encounter an error during the installation, see "Troubleshoot the load generator Linux installation" on page 58.

By default, the load generator is launched at the end of the installation. If you do not want to automatically launch the load generator, add the following command-line option:

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

Deploy Dockerized load generators on Linux

This section describes how to run a Dockerized load generator on a Linux distribution.

Docker is a platform that allows you to develop, ship, and run applications via a container.

> **Note:** For supported protocols on Dockerized load generators, see the Supported Protocols guide.

Prerequisites

> **Note:** The Ubuntu image for the OneLG load generator replaces the previous Ubuntu load generator docker image.

> For information on OneLG, see Load generators in the LoadRunner Professional Help Center (select the relevant version).

- Install Docker on the target machine, along with its dependencies, and set up the target machine environment as required. Currently, only the 64-bit version is supported. For installation details, see the Docker online documentation.

- Obtain the predefined load generator Docker image. Two images are available, for Linux-Ubuntu and RHEL.

  Pull the image from the from the relevant page, accessible from the performance testing page (https://hub.docker.com/u/performancetesting) in the Docker hub.

  Use one of the following commands and appropriate **\<tag version number\>**, for example, 20.03:

  - For Linux-Ubuntu:

    ```
    docker pull performancetesting/microfocus_onelg_linux_
    ubuntu:<tag version number>
    ```

  - For RHEL:

    ```
    podman pull performancetesting/load_generator_redhat:<tag
    version number>
    ```

Run a Dockerized load generator using the predefined image

Use the ready-to-use image to run a load generator on Docker for Linux.

> **Note:**
>
> - To run the Linux-Ubuntu OneLG image for LoadRunner Professional, the appropriate environment variable must be included in the run command: **ONELG_FLAVOR=1**
>
> - If you need customization for your container, for example, for proxy servers, see "Run a Dockerized load generator using a custom image" on page 46.

To run a Dockerized load generator:

Run the load generator container using the appropriate command.

Linux-Ubuntu:

```
docker run -id -p <host_port>:54345 -e "ONELG_FLAVOR=1" --net=host
performancetesting/microfocus_onelg_linux_ubuntu:<tag version
number>
```

RHEL:

```
podman run -id -p <host_port>:54345 performancetesting/load_
generator_redhat:<tag version number>
```

> **Note:** Check that the <host_port> on the Linux machine is available and allows incoming requests. You specify this port on the Controller side when connecting to this load generator.

Example using SSH

The following gives a simple C# code example for running multiple load generator containers using SSH. There are container orchestrator tools which do the same, for example, Kubernetes, OpenShift, Docker Swarm.

```csharp
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();
}
```

Run a Dockerized load generator using a custom image

If your environment requires customized settings for running the container, for example for proxy servers, you can create a Dockerfile to build a custom image.

> **Note:** Another alternative for customized settings: Start the container; once it is running, set up the load generator environment variables, then start the load generator manually inside the container.

To run a custom Dockerized load generator:

1. Create a new folder, and within it create a file named **dockerfile**. Paste the **FROM** line, plus the required customization lines, into the file, using the appropriate LoadRunner Professional version for the <tag version number>.

   For example, for Linux-Ubuntu:

   ```
   FROM performancetesting/microfocus_onelg_linux_ubuntu:<tag
   version number>
   ENV http_proxy http://my_proxy_name:port
   ```

   > **Note:** The above customization example is for a proxy. It defines an environment variable for the proxy server host and port in the target image.

2. Save dockerfile.

3. Open a command line at the dockerfile folder path and run the following command, using the name you want for your custom image:

   Linux-Ubuntu:

```
docker build -t <custom image name> .
```

RHEL:

```
podman build -t <custom image name> .
```

4. Create a container for each load generator you want to use, by running the following command:

   Linux-Ubuntu:

   ```
   docker run -id -p <host_port>:54345 -e "ONELG_FLAVOR=1" --
   net=host <custom image name>
   ```

   RHEL:

   ```
   podman run -id -p <host_port>:54345 <custom image name>
   ```

   If the custom image in step 3 was built with a tag then include it in the command, for example:

   ```
   docker run -id -p <host_port>:54345 -e "ONELG_FLAVOR=1" --
   net=host <custom image name>:<tag version number>
   ```

   > **Note:** Check that the <host_port> on the Linux machine is available and allows incoming requests. You specify this port on the Controller side when connecting to this load generator.

After running the load generator containers

Add the load generators containers to scenarios. For details, see the information on configuring Dockerized load generators in the LoadRunner Professional Help Center (select the relevant version).

Tips and guidelines

- Dockerized load generators that are run from the predefined image are not supported when running over a firewall. (Workaround for advanced users: You

can develop your own Docker image with MI Listener support.)

- Use `docker ps` to list the containers that are running.

- To stop the load generator service:

  - Use `docker stop <load generator container name or ID>` if you want to reuse the same load generator.

  - Use `docker rm -f <load generator container name or ID>` in order to remove the load generator container.

- The Dockerfile container has an ENTRYPOINT section. The container first runs the commands in ENTRYPOINT. It sets up the environment and then starts the load generator. The command uses a While loop to wait for input, in order to keep the container from exiting. This behavior prevents you from accessing the container while it is running. Make sure to add **-i** while starting the container; otherwise the While loop will consume an excessive amount of CPU.

- If you need entry into the container, add an argument such as `--entrypoint=/bin/bash` when starting the container. After entering the container, set the load generator environments and start the load generator . You can then switch to the host using CTRL+p and CTRL+q while keeping the container running in the background. To access the container again, use the `docker attach container_id` command.

- To access the host network directly, use `--net=host` in place of `-p <host_port>:54345`. We recommend you use this flag if the AUT generates a lot of network activity.

Configure the Linux environment

This section describes the configuration steps you need to complete after installing the load generator, before you can begin working with the load generator.

To complete the setup process after installing the load generator:

1. Set the appropriate environment variables.

    See "Set the environment variables" below.

    > **Note:** If you used a **source** command to install the load generator, the Setup wizard automatically sets the appropriate environment variables, and there is no need to perform this step.

2. Verify the load generator installation.

    See "Verify the Linux installation" on page 51.

3. Start the load generator.

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

4. Check that Controller is able to access the load generator.

    For details, see "Check Controller connection" on page 52.

Set the environment variables

> **Note:** This topic is applicable only if you ran the Load Generator Setup wizard without using a **source** (or "dot") command. If you used these commands, there is no need to perform any of the procedures described below.

To enable the load generator to run, the following environment variables must be defined:

- **M_LROOT.** The location of the Linux load generator root folder.
- **PATH.** The location of the Linux load generator **bin** directory.
- **PRODUCT_DIR.** The location of the Linux load generator root folder.

The Load Generator Setup wizard performs the following tasks relating to the environment variables:

- Adds the environment variable definitions to the system-wide startup scripts.

  If the variable definitions were not correctly set during the setup, see for possible solutions.

- Sets environment variables for the current shell session if the **source** command was used to run the Setup wizard.

  This topic describes how to set the environment variables for the current shell session if a **source** command was not used to run the Setup wizard.

To determine if environment variables are set, run **verify_generator** (see) or use the following command:

```
echo $M_LROOT
```

If the name of the load generator root folder is returned, then the environment variables are correctly set for current shell. If the name of the load generator root folder is not returned, then manually set the variables as described below.

To manually set the environment variables for the current shell session (if the **source** command was not used to run the Setup wizard), execute one of the following commands:

- Bash users:

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

- C Shell users:

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

Verify the Linux installation

The load generator installation includes a setup verification utility, **verify_ generator**, that checks the load generator setup on your Linux machine. The verification utility checks environment variables and your startup scripts (**/etc/csh.cshrc**, **${HOME}/.cshrc** or **/etc/profile**, **${HOME}/.profile**) to verify that they are set up correctly.

It is strongly recommended that you run **verify_generator** after installing the load generator, before attempting to invoke the load generator. For details on how to run the **verify_generator** utility, see "Run verify_generator" below.

The **verify_generator** utility checks the following:

- All the prerequisite software is installed. (This check is performed for 64-bit installations only.)
- There are at least 128 file descriptors
- The `.rhosts` permissions have been defined properly: `-rw-r--r--`
- The host can be contacted by using rsh to the host. If not, it checks for the host name in .rhosts
- **M_LROOT** is defined
- **.cshrc** or **.profile** defines the correct **M_LROOT**
- `/etc/csh.cshrc`, `${HOME}/.cshrc` or `/etc/profile`, `${HOME}/.profile` defines the correct **M_LROOT**
- **.cshrc** or **.profile** exists in the home directory
- The current user is the owner of the **.cshrc** or **.profile**
- A Linux load generator installation exists in **$M_LROOT**
- The executables have executable permissions
- PATH contains **$M_LROOT/bin** and **/usr/bin**

Run verify_generator

It is recommended that you run the **verify_generator** utility after installing the load generator, before attempting to invoke the load generator. For details on what is

checked by the **verify_generator** utility, see "Verify the Linux installation" on the previous page.

> **Note:**
> - To run this command, you must be a "normal" user and not root user.
> - Before you run the **verify_generator** utility, make sure that you have set the DISPLAY environment variable on your machine.

To run verify_generator:

1. From the **<Load Generator root>/bin** folder, run the following command:

```
./verify_generator
```

For example:

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

If you want to receive detailed information about the checks, you can use the -v option, as follows:

```
./verify_generator -v
```

2. View the results.
   - If the settings are correct, **verify_generator** returns **OK**.
   - If any of the settings are incorrect, **verify_generator** returns **Failed**, and suggestions on how to correct the setup.

Check Controller connection

If Controller is to connect remotely to the load generator using **rsh** (remote shell), you need to make sure that the load generator can be remotely accessed by Controller.

1. On the load generator machine, locate the **.rhosts** file which is located in the user home directory.

2. In the **.rhosts** file, verify that Controller is included in the list of machines. If it is not listed, add it to the list.

If Controller still cannot connect to the load generator, contact your system administrator.

Connecting to a Linux load generator without using rsh

You can configure Controller to connect to the load generator without using **rsh**. In this case, you need to activate the agent daemon on the load generator, as described below.

This section describes how to connect to a Linux load generator without using **rsh**.

1. On the Linux load generator, run the agent daemon by entering the following command from **<Load Generator root>/bin**:

```
./m_daemon_setup -install
```

This runs a daemon called **m_agent_daemon**, and if successful, you receive a message: **m_agent_daemon <process ID>**.

The agent now keeps running, even if the user is logged off. You can stop the agent by using the command explained in step 9 below, or by rebooting the machine.

> **Note:** If you look at the **m_agent_daemon[xxx].log** log file in the temp directory, you may see communication errors, even if the installation succeeded.

2. In Controller, select **Scenario > Load Generators**. The Load Generators dialog box opens.

3. Click **Add**. The Add New Load Generator dialog box opens.

4. In the **Name** box, enter the name of the computer on which the load generator is running.

5. From the **Platform** list, select **Linux**.

6. Click **More**.

7. Click the **Linux Environment** tab, and make sure that the **Don't use RSH** check box is selected.

8. Connect as usual.

9. To stop the agent daemon, run the following command from the **<LoadRunner Professional root>/bin** directory:

```
./m_daemon_setup -remove
```

This stops the **m_agent_daemon** daemon, and if successful, you receive the message: **m_agent_daemon is down**.

Improve load generator performance

This section includes recommendations for improving load generator performance. You can increase the number of file descriptors, process entries, and amount of swap space by configuring the kernel.

> **Note:** Most operating systems using the Linux load generator have sufficient default file descriptors, process entries, and swap space, and rarely require reconfiguration.

This section includes:

Increase file descriptors

A load generator uses the following file descriptor resources:

- 14 file descriptors for the launch service

- 20 file descriptors for the agent

- 30 file descriptors for each Vuser driver. By default, there is a driver for every 50

Vusers.

- File descriptors for the running Vusers. Each Vuser requires two descriptors.

For example, to compute the number of file descriptors used in running 100 threaded Vusers, the load generator requires:

Descriptors	Purpose of the descriptors
14	For the launcher
20	For the agent
60	For 2 drivers (30 x 2, each one drives 50 Vusers)
200	For 100 Vusers (each Vuser requires 2)

Total: 294 file descriptors

If Vusers are run as processes instead of threads, one driver is run per Vuser. Therefore, each Vuser requires 30 file descriptors.

The procedure to increase the number of file descriptors differs between shells.

In the examples below, the number of descriptors is increased to the maximum of 1024.

- For sh and ksh users, type:

```
ulimit -n 1024
```

- For csh users, type:

```
limit descriptors 1024
```

Below is an alternate procedure to increase file descriptors. In this example, the number of descriptors is increased to the maximum of 8192.

1. Add the following line to the **/etc/security/limits.conf** file:

```
hard nfile 8192
```

2. Add the following line to the **/etc/sysctl.conf** file:

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

3. Reboot the machine.

Increase process entries

Each Vuser requires several free process entries. To increase the number of process entries on your system, you must reconfigure the kernel.

This section describes how to reconfigure the kernel for Linux platforms.

1. Locate the **/etc/security/limits.conf** file.
2. Set the maximum number of processes in the limits file. Type:

```
hard nproc 8192
```

3. Reboot the machine.

Increase swap space

Each Vuser requires swap space ranging in size from 200 KB to 4 MB. Before adding space to your system configuration, you should determine your paging requirements. For environments running programs with very large memory requirements, it is recommended to have paging space of four times the physical memory. If you do not have enough paging space, certain processes may be killed, and others will be unable to start.

Uninstall the load generator from a Linux machine

This section describes how to uninstall the load generator from a Linux machine.

This section includes:

-
-

Uninstall load generator 12.55 or earlier

You can uninstall load generator 12.55 or earlier versions using a manual command or silently.

To uninstall load generator version 12.55 or earlier:

1. Make sure that you are logged in as the same user who installed the load generator.

2. Make sure that the **m_agent_daemon** process is not running on the machine. If it is running, kill its process:

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

3. Change the current directory to the installation directory:

   ```
   cd <path_to_installation_folder>/_HP_LoadGenerator_Installation
   ```

4. Switch to super user.

5. Uninstall the load generator:

   - **Manual uninstall:** run the following command:

     ```
     sh ./Change_HP_LoadGenerator_Installation
     ```

   - **Silent uninstall:** Run the following command:

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

Uninstall load generator 12.56 or later

You can uninstall load generator 12.56 or later using the Setup wizard or a manual command, or silently.

To uninstall load generator version 12.56 or later:

1. Make sure that you are logged in as the same user who installed the load generator.

2. Make sure that the **m_agent_daemon** process is not running on the machine. If it is running, kill its process:

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

3. Change the current directory to the installation directory:

   ```
   cd <path_to_installation_folder>/_MF_LoadGenerator_Installation
   ```

4. Switch to super user.

5. Uninstall the load generator:

   - **Manual uninstall:** Run the following command:

     ```
     sh ./Change_MF_LoadGenerator_Installation
     ```

   - **Silent uninstall:** Run the following command:

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

Troubleshoot the load generator Linux installation

This section describes troubleshooting tasks relating to the setup of the Linux load generator.

This section includes:

Environment variables were not set correctly in the system-wide startup scripts

To enable the load generator to run, the system-wide startup scripts must be modified to set specific environment variables. The required modifications to the startup scripts are made by the Load Generator Setup wizard. If the startup scripts were not correctly modified during the setup of the load generator, you can manually make the required changes to the startup scripts as described below. The required changes differ slightly between C shell users, and Bourne and Korn shell users.

- **Manually modifying the startup scripts for C shell users**

  During the load generator installation process, the wizard creates the **env.csh** script. This script includes the commands to set the required environment variables for C shell users. A sample **env.csh** script is shown below.

```
setenv PRODUCT_DIR <Load Generator installation directory>

setenv M_LROOT ${PRODUCT_DIR}

   if ( ! $?PATH ) then

      setenv PATH ""

   endif

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

Add the following line to the **/etc/csh.cshrc** or **~/.cshrc** startup script to execute the **env.csh** script during the shell startup:

```
source <Load Generator installation directory>/env.csh
```

For example:

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

The effect of making the above modification to the startup script is similar to the modifications that are made by the Setup wizard. A sample of the modifications that the Setup wizard makes to the **/etc/csh.cshrc** startup script is shown below:

```
# 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.
```

- **Manually modifying the startup scripts for Bourne and Korn shell users**

  During the load generator installation, the Setup wizard creates the **env.sh**
  script. This script includes commands to set the required environment variables
  for Bourne shell and Korn shell users.

  Add the following line to the **/etc/profile** or **~/.profile** startup script to execute
  the **env.sh** script during the shell startup:

  ```
  source <Load Generator installation directory>/env.sh
  ```

  For example:

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

  The effect of making the above modification to the startup script is similar to the
  modifications that are made by the Setup wizard. A sample of the modifications
  that the Setup wizard makes to the **/etc/profile** startup script is shown below:

  ```
  # 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
```

Error when installing the load generator on a Linux platform

When you use the **source installer.sh** command to install the latest load generator version on a Linux machine on which a load generator was previously installed, you may receive the following error message:

"An error occurred while trying to manage the selected instance."

Solution:

1. Open the registry file **/var/.com.zerog.registry.xml** and locate the element "**product**" with attribute "**name**"="**MF_LoadGenerator**".

   For example:

   <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. Record the value of the "**location**" attribute.

3. Remove the entire directory that is referred to by the "**location**" attribute.

4. Delete the registry file **/var/.com.zerog.registry.xml**.

5. Rerun the **source installer.sh** command.

Error when running load generator on RedHat Enterprise Linux 5.x with SELinux enabled

During use of the load generator on RHEL 5.x, you might receive the following error:

```
"m_agent_daemon: error while loading shared libraries: /opt/MF/MF_
LoadGenerator/bin/liblwc_cryptolib.so: cannot restore segment prot
after reloc: Permission denied."
```

This problem occurs because SELinux is installed and enabled on the machine. SELinux is preventing the specified shared library from loading.

Solution:

There are two possible workarounds:

1. Before using the load generator, disable SELinux using the command "setenforce 0".

2. If you want to keep SELinux enabled, you can change the security context of all problematic libraries (for example, <Path_to_LoadGenerator>/bin/*.so" to "textrel_shlib_t").

To do this, execute the command: "chcon -t textrel_shlib_t <Path_to_ LoadGenerator>/bin/*.so"

Environment variables are not unset after uninstalling the load generator

When you uninstalled the Linux load generator, the Setup wizard might not have unset the load generator environment variables (M_LROOT, PRODUCT_DIR, and PATH) for the current shell. To unset the environment variables, close the current shell session and invoke a new one, or manually unset the variables as described below:

- To unset the M_LROOT and PRODUCT_DIR variables:
  - [bash shells] Use the **unset** command.
  - [csh shells] Use the **unsetenv** command.
- To update the PATH variable to exclude the load generator binary directory, type:
  - [bash shells] `PATH=<required list of paths>; export PATH`
  - [csh shells] `setenv PATH <required list of paths>`

Unable to run Vusers on the load generator

If you are unable to run Vusers on the load generator, no specific error is reported, and the Vuser protocol requires a third-party application or client to be present on load generator side, check the dynamic libraries used by the application. This enables you to establish if any shared objects cannot be found. A shared object that cannot be found may indicate either a missing prerequisite package or an environment variable issue.

To check the dynamic libraries used by an application, type:

```
ldd application_name
```

For example, type `ldd mdrv` to determine if all the dependencies of the **mdrv** executable can be found. If any dependencies are not found, run **verify_generator** as described in .

> **Note:** If you are running Vusers for a protocol that requires a client installation (for example, Oracle), make sure that the path for the client libraries is included in the dynamic library path environment variable (LD_LIBRARY_PATH or SHLIB_PATH).

Installing the LoadRunner Data Hub

The LoadRunner Data Hub is a communication channel based on Kafka technology that enables more robust communication between LoadRunner components. To use this channel, install the Data Hub. In addition, if your LoadRunner components communicate via the web, you need to install the Data Hub Web Connector. For details, see "Installing the Data Hub Web Connector" on page 72.

The Data Hub must be installed on Linux. It can communicate with LoadRunner components running on Linux and Windows machines.

After you install the Data Hub, you need to configure the LoadRunner components to use it. For details, see Data Hub and Web Connector setup in the LoadRunner Professional Help Center.

This chapter includes:

Installation workflow

Follow these primary steps to install the Data Hub using the standard installation program:

1. **Review the system requirements**

   - Before you install the Data Hub, make sure that your system meets the requirements, as detailed in the Support Matrix.

   - The Data Hub requires the latest version of Java JRE to be installed as a prerequisite.

   - It is recommended that you close all antivirus applications before running the installation.

2. Download the Data Hub installation package from AppDelivery Marketplace.

3. **Run the setup**

- **Wizard installation:** Start the setup using the Data Hub Setup wizard. For more information, see "Install the Data Hub with the Setup wizard" below.
- **Silent installation:** To perform a silent installation, see "Silent Data Hub installation" on the next page.

4. **Verify the installation**

   See "Verify the Data Hub installation" on page 71.

Install the Data Hub with the Setup wizard

This section describes how to install the LoadRunner Data Hub using the Setup wizard.

This section includes:

- "Before you start" below
- "Run the Setup wizard" on the next page

Before you start

During installation, you can optionally install CA and SSL certificates. These certificates are used for authentication and secure communication respectively. Both certificates should be in *.cer (X.509) format.

- For a CA certificate: You can enter the path to an existing certificate, or leave the path blank to skip the step. If you want to install the CA certificate, it should be generated in advance.
- For an SSL certificate: You can select it from an existing certificate file. Alternatively, it can be generated automatically if you provide the CA certificate containing the private key.

For details on generating a CA certificate, see the section about creating and installing digital certificates in the LoadRunner Professional Help Center (select the relevant version).

For more information on working with certificates in LoadRunner Professional, see the section about configuring client-server authentication in the LoadRunner Professional Help Center.

Run the Setup wizard

To install the Data Hub:

1. Switch to super user.

2. Unpack the Data Hub installation package and open the location of the **.bin** file.

3. Launch the Setup wizard:

For:	Type:
[sh and bash shells]	`./inst64.bin`

4. Follow the online instructions to install the Data Hub.

> **Note:** The default installation path is: **/opt/MF/MF_LoadRunner_Data_Hub/_MF_LoadRunner_Data_Hub_Installation**

5. Exit super user or switch to another user.

Silent Data Hub installation

This section describes how to perform a silent installation of the Data Hub.

This section includes:

- "Run a silent installation" on the next page
- "Silent installation options" on page 70

> **Note:** You must have super user privileges.

Run a silent installation

You can install the Data Hub silently by following the steps below.

To perform a silent installation of the Data Hub:

1. Unpack the Data Hub installation package.

2. Create a file named **installer.properties** using the following template, changing the parameters as needed.

   > **Tip:** An **installer.properties** file is generated when you run the Setup wizard, and can be used again for later silent installations.

```
#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. Save the **installer.properties** file in the same location as the installer file.

4. Run the installation command using the following syntax:

```
./inst64.bin [-f<path to installer properties file> | -options]
```

You can modify the installation by adding options to this command. For details, see "Silent installation options" below.

> **Note:**
> - When the **installer.properties** file is saved in the same directory as the installer, it overrides all other command line options unless the `-f` option is used to specify another valid properties file.
> - The path to the installer properties file may be either absolute, or relative to the directory in which the installer resides.

For example, the following command installs the Data Hub silently using an **installer.properties** file that is saved in the same location as the installer:

> **Example:** `./inst64.bin -i silent`

Silent installation options

You can add the following options to the silent installation command:

Option	Description
-i [swing │ console │ silent]	Specifies the user interface mode for the installer.
-D<name>=<value>	Specifies the installer properties.
-r <path to generate response file>	Generates a response file.

Repair and uninstall options

The following options can be used to repair or uninstall the Data Hub:

Option	Description
-repair	Repairs the installation.
-uninstall	Uninstalls the Data Hub.

Verify the Data Hub installation

To verify that the Data Hub was installed successfully, run the following commands to check the status of `zookeeper` and `kafka` services. If these services are running, the installation was successful.

```
service zookeeper status
```

```
service kafka status
```

Installing the Data Hub Web Connector

The Data Hub Web Connector enables the LoadRunner Data Hub to communicate via the web and over firewalls. For example, if a load generator is behind a firewall and communicates via a proxy, the Web Connector enables the Data Hub to connect to the proxy and communicate with the load generator.

The Web Connector must be installed on Linux.

After you install the Web Connector, you need to configure the LoadRunner components to use it. For details, see Data Hub and Web Connector setup in the LoadRunner Professional Help Center.

This chapter includes:

Installation workflow

Follow these primary steps to install the Web Connector using the standard installation program:

1. **Review the system requirements**

   - Before you install the Web Connector, make sure that your system meets the requirements, as detailed in the Support Matrix.

   - It is recommended that you close all antivirus applications before running the installation.

2. The Web Connector installation package is included in the Data Hub installation package. If you have not already downloaded the Data Hub installation package, download it from AppDelivery Marketplace.

3. **Run the setup**

   - **Wizard installation:** Start the setup using the Web Connector Setup wizard. For more information, see "Install the Web Connector with the Setup wizard"

on the next page.

- **Silent installation:** To perform a silent installation, see "Silent Web Connector installation" on the next page.

4. **Verify the installation**

See "Verify the Web Connector installation" on page 77.

Install the Web Connector with the Setup wizard

This section describes how to install the Web Connector using the Setup wizard.

This section includes:

- "Before you start" below
- "Run the Setup wizard" on the next page

Before you start

During installation, you can optionally install CA and SSL certificates. These certificates are used for authentication and secure communication respectively. Both certificates should be in *.cer (X.509) format.

- For a CA certificate: You can enter the path to an existing certificate, or leave the path blank to skip the step. If you want to install the CA certificate, it should be generated in advance.

- For an SSL certificate: You can select it from an existing certificate file. Alternatively, it can be generated automatically if you provide the CA certificate containing the private key.

For details on generating a CA certificate, see the section about creating and installing digital certificates in the LoadRunner Professional Help Center (select the relevant version).

For more information on working with certificates in LoadRunner Professional, see the section about configuring client-server authentication in the LoadRunner Professional Help Center.

Run the Setup wizard

To install the Web Connector:

1. Switch to super user.

2. Unpack the Web Connector installation package and open the location of the .bin file.

3. Launch the Setup wizard:

For:	Type:
[sh and bash shells]	`./inst64.bin`

4. Follow the online instructions to install the Web Connector.

> **Note:** The default installation path is: **/opt/MF/MF_LoadRunner_Data_Hub_Web_Connector/_MF_LoadRunner_Data_Hub_Web_Connector_Installation**

5. Exit super user or switch to another user.

Silent Web Connector installation

This section describes how to perform a silent installation of the Web Connector.

This section includes:

- "Run a silent installation" on the next page
- "Silent installation options" on page 76

> **Note:** You must have super user privileges.

Run a silent installation

You can install the Web Connector silently by following the steps below.

To perform a silent installation of the Web Connector:

1. Unpack the Web Connector installation package.

2. Create a file named **installer.properties** using the following template, changing the parameters as needed. The template here installs the Web Connector with the default LoadRunner certificates.

> **Tip:** An **installer.properties** file is generated when you run the Setup wizard, and can be used again for later silent installations.

```
#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. Save the **installer.properties** file in the same location as the installer file.

4. Run the installation command using the following syntax:

```
./inst64.bin [-f<path to installer properties file> | -options]
```

You can modify the installation by adding options to this command. For details, see "Silent installation options" below.

> **Note:**
> - When the **installer.properties** file is saved in the same directory as the installer, it overrides all other command line options unless the `-f` option is used to specify another valid properties file.
> - The path to the installer properties file may be either absolute, or relative to the directory in which the installer resides.

For example, the following command installs the Web Connector silently using an **installer.properties** file that is saved in the same location as the installer:

> **Example:** `./inst64.bin -i silent`

Silent installation options

You can add the following options to the silent installation command:

Option	Description
-i [swing | console | silent]	Specifies the user interface mode for the installer.
-D<name>=<value>	Specifies the installer properties.
-r <path to generate response file>	Generates a response file.

The following options can be used to repair or uninstall the Web Connector:

Option	Description
-repair	Repairs the installation.
-uninstall	Uninstalls the Web Connector.

Verify the Web Connector installation

To verify that the Web Connector was installed successfully, run the following command to check the status of the `webconnector` service. If this service is running, the installation was successful.

```
service webconnector status
```