



# Red Hat Developer Tools 1

## 使用 Go 1.19.6 Toolset

安装和使用 Go 1.19.6 Toolset



# Red Hat Developer Tools 1 使用 Go 1.19.6 Toolset

---

安装和使用 Go 1.19.6 Toolset

Jacob Valdez

[jvaldez@redhat.com](mailto:jvaldez@redhat.com)

## 法律通告

Copyright © 2023 Red Hat, Inc.

The text of and illustrations in this document are licensed by Red Hat under a Creative Commons Attribution–Share Alike 3.0 Unported license ("CC-BY-SA"). An explanation of CC-BY-SA is available at

<http://creativecommons.org/licenses/by-sa/3.0/>

. In accordance with CC-BY-SA, if you distribute this document or an adaptation of it, you must provide the URL for the original version.

Red Hat, as the licensor of this document, waives the right to enforce, and agrees not to assert, Section 4d of CC-BY-SA to the fullest extent permitted by applicable law.

Red Hat, Red Hat Enterprise Linux, the Shadowman logo, the Red Hat logo, JBoss, OpenShift, Fedora, the Infinity logo, and RHCE are trademarks of Red Hat, Inc., registered in the United States and other countries.

Linux<sup>®</sup> is the registered trademark of Linus Torvalds in the United States and other countries.

Java<sup>®</sup> is a registered trademark of Oracle and/or its affiliates.

XFS<sup>®</sup> is a trademark of Silicon Graphics International Corp. or its subsidiaries in the United States and/or other countries.

MySQL<sup>®</sup> is a registered trademark of MySQL AB in the United States, the European Union and other countries.

Node.js<sup>®</sup> is an official trademark of Joyent. Red Hat is not formally related to or endorsed by the official Joyent Node.js open source or commercial project.

The OpenStack<sup>®</sup> Word Mark and OpenStack logo are either registered trademarks/service marks or trademarks/service marks of the OpenStack Foundation, in the United States and other countries and are used with the OpenStack Foundation's permission. We are not affiliated with, endorsed or sponsored by the OpenStack Foundation, or the OpenStack community.

All other trademarks are the property of their respective owners.

## 摘要

Go Toolset 是 Red Hat Enterprise Linux (RHEL) 操作系统的开发人员的红帽产品。使用本指南获取 Go Toolset 的概述，了解如何调用和使用不同版本的 Go 工具，并通过更深入的信息查找资源。

---

# 目录

<b>使开源包含更多</b> .....	<b>3</b>
<b>第 1 章 GO TOOLSET</b> .....	<b>4</b>
1.1. GO TOOLSET 组件	4
1.2. GO TOOLSET 兼容性	4
1.3. 在 RED HAT ENTERPRISE LINUX 7 上访问 GO TOOLSET	4
1.4. 安装 GO TOOLSET	5
1.5. 安装 GO 文档	6
1.6. 其他资源	6
<b>第 2 章 GO 编译器</b> .....	<b>7</b>
2.1. 先决条件	7
2.2. 设置 GO 工作区	7
2.3. 编译 GO 程序	7
2.4. 运行 GO 程序	8
2.5. 安装编译的 GO 项目	8
2.6. 下载并安装 GO 项目	9
2.7. 其他资源	10
<b>第 3 章 GOFMT 格式工具</b> .....	<b>11</b>
3.1. 先决条件	11
3.2. 格式化代码	11
3.3. 预览对代码的更改	11
3.4. 简化代码	12
3.5. 重构代码	13
3.6. 其他资源	14
<b>第 4 章 GO 竞争检测器</b> .....	<b>15</b>
4.1. 先决条件	15
4.2. 使用 GO 竞争检测器	15
4.3. 其他资源	15
<b>第 5 章 带有 GO TOOLSET 的容器镜像</b> .....	<b>16</b>
5.1. RED HAT ENTERPRISE LINUX GO TOOLSET 容器镜像内容	16
5.2. 访问 RED HAT ENTERPRISE LINUX 容器镜像	16
5.3. 访问 RHEL 8 上的 UBI GO TOOLSET 容器镜像	16
5.4. 从 RHEL 8 上的基本 UBI 容器镜像访问 GO TOOLSET	16
5.5. 在 RED HAT ENTERPRISE LINUX 7 中使用容器镜像作为 SOURCE-TO-IMAGE 构建器镜像	17
5.6. 其他资源	17
<b>第 6 章 GO TOOLSET 中的更改</b> .....	<b>19</b>



## 使开源包含更多

红帽致力于替换我们的代码、文档和 Web 属性中存在问题的语言。我们从这四个术语开始：master、slave、黑名单和白名单。由于此项工作十分艰巨，这些更改将在即将推出的几个发行版本中逐步实施。有关更多详情，请参阅[我们的首席技术官 Chris Wright 提供的消息](#)。

# 第 1 章 GO TOOLSET

Go Toolset 是 Red Hat Enterprise Linux (RHEL) 上的开发人员的产品。它提供 Go 编程语言工具和库。请注意，Go 也称为 golang。

Go Toolset 作为 Red Hat Developer Tools for Red Hat Enterprise Linux 7 的一部分发布。Go Toolset 作为 Red Hat Enterprise Linux 8 的模块提供。Go Toolset 作为 Red Hat Enterprise Linux 9 的软件包提供。

## 1.1. GO TOOLSET 组件

以下组件作为 Go Toolset 的一部分提供：

Name	版本	Description
golang	RHEL 7 - 1.19.6, RHEL 8 - 1.19.4, RHEL 9 - 1.19.6	Go 编译器。
delve	RHEL 7 - 1.9.0, RHEL 8 - 1.9.0, RHEL 9 - 1.9.0	Go 调试器。

## 1.2. GO TOOLSET 兼容性

在以下构架中，Red Hat Enterprise Linux 7 和 Red Hat Enterprise Linux 8 和 Red Hat Enterprise Linux 9 提供了 Go Toolset：

- AMD 和 Intel 64 位
- 64 位 ARM (RHEL 8 和 RHEL 9)
- IBM Power Systems, Little Endian
- 64-bit IBM Z

## 1.3. 在 RED HAT ENTERPRISE LINUX 7 上访问 GO TOOLSET

为了能够在 Red Hat Enterprise Linux 7 上安装 Go Toolset，您必须访问并启用 Red Hat Developer Tools 和 Red Hat Software Collections 软件仓库。

如果这些软件仓库已附加到您的系统，[请参阅安装 Go Toolset](#)。

### 流程

1. 运行以下命令安装 **Wget**：

```
# yum install wget
```

2. 运行以下命令下载最新的订阅数据：

```
# subscription-manager refresh
```

3. 运行以下命令注册您的系统：



```
# subscription-manager register
```

要使用图形用户界面(GUI) [注册您的系统](#)，请按照[注册和取消注册系统](#) 指南进行操作。

- 显示所有可用订阅列表，并运行以下命令来识别池 ID：

```
# subscription-manager list --available
```

- 在以池 ID 开头的行中找到 **池 ID**。
- 运行以下命令，将提供对 **Red Hat Developer Tools** 存储库的访问权限的订阅附加：

```
# subscription-manager attach --pool=<pool ID from the subscription>
```

- 将 **subscription** 中的 **&lt;** pool ID 替换为您在上一步中标识的池 ID。

- 运行以下命令验证哪些订阅是否已附加到您的系统：

```
# sudo subscription-manager list --consumed
```

- 运行以下命令启用 **rhel-7-变体-devtools-rpms** 存储库：

```
# subscription-manager repos --enable rhel-7-<variant>-devtools-rpms
```

- 将 **<variant>** 替换为您的 Red Hat Enterprise Linux 系统变体：**server** 或 **workstation**。使用 **server** 访问广泛的开发工具。

- 运行以下命令启用 **rhel-variant-RHSCL-7-rpms** 存储库：

```
# subscription-manager repos --enable rhel-<variant>-rhscl-7-rpms
```

- 将 **<variant>** 替换为您的 Red Hat Enterprise Linux 系统变体：**server** 或 **workstation**。

- 运行以下命令，在您的系统中添加 Red Hat Developer Tools GPG 密钥：

```
# cd /etc/pki/rpm-gpg
# wget -O RPM-GPG-KEY-redhat-devel https://www.redhat.com/security/data/a5787476.txt
# rpm --import RPM-GPG-KEY-redhat-devel
```

## 其他资源

- 有关注册您的系统并将其与订阅关联的更多信息，请参阅 [红帽订阅管理](#) 指南集合。

## 1.4. 安装 GO TOOLSET

完成以下步骤以安装 Go Toolset，包括所有依赖软件包。

### 先决条件

- 在 Red Hat Enterprise Linux 7 中，提供对 Red Hat Developer Tools 内容集合的访问权限的订阅已附加到您的系统中。要附加订阅，请参阅 [获取 Red Hat Enterprise Linux 7 上的 Go Toolset 的访问权限](#)。

- 已安装所有可用的 Red Hat Enterprise Linux 更新。

## 流程

在 Red Hat Enterprise Linux 7 中，运行以下命令安装 **go-toolset-1.19** 集合：

```
# yum install go-toolset-1.19
```

在 Red Hat Enterprise Linux 8 中，运行以下命令安装 **go-toolset** 模块：

```
# yum module install go-toolset
```

在 Red Hat Enterprise Linux 9 中，运行以下命令安装 **go-toolset** 软件包：

```
# dnf install go-toolset
```

## 1.5. 安装 GO 文档

您可以在本地系统中安装 Go 编程语言的文档。

### 先决条件

- 已安装 Go Toolset。  
如需更多信息，[请参阅安装 Go Toolset](#)。

## 流程

要安装 **golang-docs** 软件包，请运行以下命令：

- 在 Red Hat Enterprise Linux 7 上：

```
# yum install go-toolset-1.19-golang-docs
```

您可以在以下路径中找到文档：**/opt/rh/go-toolset-1.19/root/usr/lib/go-toolset-1.19-golang/doc/go\_spec.html**。

- 在 Red Hat Enterprise Linux 8 中：

```
# yum install golang-docs
```

您可以在以下路径中找到文档：**/usr/lib/golang/doc/go\_spec.html**。

- 在 Red Hat Enterprise Linux 9 中：

```
# dnf install golang-docs
```

您可以在以下路径中找到文档：**/usr/lib/golang/doc/go\_spec.html**。

## 1.6. 其他资源

- 有关 Go 编程语言、工具和库的更多信息，[请参阅官方 Go 文档](#)。

## 第 2 章 GO 编译器

Go 编译器是 Go 编程语言的构建工具和依赖项管理器。它提供代码的错误检查和优化。

### 2.1. 先决条件

- 已安装 Go Toolset。  
如需更多信息，[请参阅安装 Go Toolset](#)。

### 2.2. 设置 GO 工作区

要编译 Go 程序，您需要设置 Go 工作区。

#### 流程

1. 创建一个工作区目录，作为 **\$GOPATH/src** 的子目录。  
一个常见的选择是 **\$HOME/go**。
2. 将源文件放到您的工作区目录中。
3. 运行以下命令，将工作区目录的位置设置为 **\$HOME/.bashrc** 文件的环境变量：

```
$ echo 'export GOPATH=<workspace_dir>' >> $HOME/.bashrc  
$ source $HOME/.bashrc
```

将 `<workspace_dir>` 替换为工作区目录的名称。

#### 其他资源

- [官方 Go 工作区文档](#)。

### 2.3. 编译 GO 程序

您可以使用 Go 编译器编译 Go 程序。Go 编译器会创建一个可执行文件，从而进行编译。

#### 先决条件

- 设置 Go 工作区。  
有关如何设置工作区的详情，请参考 [设置 Go 工作区](#)。

#### 流程

在项目目录中，运行：

- 在 Red Hat Enterprise Linux 7 上：

```
$ scl enable go-toolset-1.19 'go build <output_file>'
```

- 将 `<output_file>` 替换为输出文件所需的名称，将 `<go_main_package>` 替换为主软件包的名称。
- 在 Red Hat Enterprise Linux 8 中：

```
$ go build <output_file>
```

- 将 **<output\_file>** 替换为输出文件所需的名称，将 **<go\_main\_package>** 替换为主软件包的名称。
- 在 Red Hat Enterprise Linux 9 中：

```
$ go build <output_file>
```

- 将 **<output\_file>** 替换为输出文件所需的名称，将 **<go\_main\_package>** 替换为主软件包的名称。

## 2.4. 运行 GO 程序

Go 编译器会创建一个可执行文件，从而进行编译。完成以下步骤以执行此文件并运行安装程序。

### 先决条件

- 您的程序已编译。  
有关如何编译程序的更多信息，请参阅 [编译 Go 程序](#)。

### 流程

要运行程序，请在包含可执行文件的目录中运行：

```
$.<file_name>
```

- 将 **<file\_name>** 替换为可执行文件的名称。

## 2.5. 安装编译的 GO 项目

您可以安装已编译的 Go 项目，以便在其他 Go 项目中使用其可执行文件和库。安装后，项目的可执行文件和库会根据 Go 工作区中的目录被复制到。另外，也会安装其依赖项。

### 先决条件

- Go 工作区。  
如需更多信息，请参阅 [设置 Go 工作区](#)。

### 流程

要安装 Go 项目，请运行：

- 在 Red Hat Enterprise Linux 7 上：

```
$ scl enable go-toolset-1.19 'go install <go_project>'
```

- 将 **<go\_project>** 替换为您要安装的 Go 项目的名称。
- 在 Red Hat Enterprise Linux 8 中：

```
$ go install <go_project>
```

- 将 **<go\_project>** 替换为您要安装的 Go 项目的名称。

- 在 Red Hat Enterprise Linux 9 中：

```
$ go install <go_project>
```

- 将 **<go\_project>** 替换为您要安装的 Go 项目的名称。

## 2.6. 下载并安装 GO 项目

您可以从在线资源下载并安装第三方 Go 项目，以便在其他 Go 项目中使用其可执行文件和库。安装后，项目的可执行文件和库会根据 Go 工作区中的目录被复制到。另外，也会安装其依赖项。

### 先决条件

- Go 工作区。  
如需更多信息，[请参阅设置 Go 工作区](#)。

### 流程

- 要下载并安装 Go 项目，请运行：

- 在 Red Hat Enterprise Linux 7 上：

```
$ scl enable go-toolset-1.19 'go get <third_party_go_project>'
```

- 将 **<third\_party\_go\_project>** 替换为您要下载的项目的名称。

- 在 Red Hat Enterprise Linux 8 中：

```
$ go get <third_party_go_project>
```

- 将 **<third\_party\_go\_project>** 替换为您要下载的项目的名称。

- 在 Red Hat Enterprise Linux 9 中：

```
$ go get <third_party_go_project>
```

- 将 **<third\_party\_go\_project>** 替换为您要下载的项目的名称。

- 有关第三方项目可能值的信息，请运行：

- 在 Red Hat Enterprise Linux 7 上：

```
$ scl enable go-toolset-1.19 'go help importpath'
```

- 在 Red Hat Enterprise Linux 8 中：

```
$ go help importpath
```

- 在 Red Hat Enterprise Linux 9 中：

```
$ go help importpath
```

## 2.7. 其他资源

- 有关 Go 编译器的更多信息，请参阅 [官方 Go 文档](#)。
- 要显示 Go Toolset 中包含的 **帮助** 索引，请运行：

- 在 Red Hat Enterprise Linux 7 上：

```
$ scl enable go-toolset-1.19 'go help'
```

- 在 Red Hat Enterprise Linux 8 中：

```
$ go help
```

- 在 Red Hat Enterprise Linux 9 中：

```
$ go help
```

- 要显示特定 Go 软件包的文档，请运行：

- 在 Red Hat Enterprise Linux 7 上：

```
$ scl enable go-toolset-1.19 'go doc <package_name>'
```

- 在 Red Hat Enterprise Linux 8 中：

```
$ go doc <package_name>
```

- 在 Red Hat Enterprise Linux 9 中：

```
$ go doc <package_name>
```

有关 [Go 软件包](#) 的概述信息，请参阅 [Go 软件包](#)。

## 第 3 章 GOFMT 格式工具

Go 编程语言使用 **gofmt** 代码格式工具，而不是风格的指南。**gofmt** 根据 Go 布局规则自动格式化您的代码。

### 3.1. 先决条件

- 已安装 Go Toolset。  
如需更多信息，[请参阅安装 Go Toolset](#)。

### 3.2. 格式化代码

您可以使用 **gofmt** 格式工具来格式化给定路径中的代码。当路径导致单个文件时，更改仅适用于该文件。当路径生成某个目录时，将处理目录中的所有 **.go** 文件。

#### 流程

要将代码格式化为给定路径，请运行：

- 在 Red Hat Enterprise Linux 7 上：

```
$ scl enable go-toolset-1.19 'gofmt -w <code_path>'
```

- 将 **<code\_path>** 替换为您要格式的代码的路径。

- 在 Red Hat Enterprise Linux 8 中：

```
$ gofmt -w <code_path>
```

- 将 **<code\_path>** 替换为您要格式的代码的路径。

- 在 Red Hat Enterprise Linux 9 中：

```
$ gofmt -w <code_path>
```

- 将 **<code\_path>** 替换为您要格式的代码的路径。



#### 注意

要将格式化的代码输出到标准输出，而不是将其写入原始文件中，请省略 **-w** 选项。

### 3.3. 预览对代码的更改

您可以使用 **gofmt** 格式工具，通过格式化给定路径中的代码来预览更改。统一 diff 格式的输出输出到标准输出。

#### 流程

要显示给定路径中代码的不同，请运行：

- 在 Red Hat Enterprise Linux 7 上：

```
$ scl enable go-toolset-1.19 'gofmt -d <code_path>'
```

- 将 `<code_path>` 替换为您要比较的代码的路径。
- 在 Red Hat Enterprise Linux 8 中：

```
$ gofmt -d <code_path>
```
- 将 `<code_path>` 替换为您要比较的代码的路径。
- 在 Red Hat Enterprise Linux 9 中：

```
$ gofmt -d <code_path>
```
- 将 `<code_path>` 替换为您要比较的代码的路径。

### 3.4. 简化代码

您可以使用 `gofmt` 格式工具来简化代码。

#### 流程

- 要简化给定路径中的代码，请运行：
  - 在 Red Hat Enterprise Linux 7 上：

```
$ scl enable go-toolset-1.19 'gofmt -s <code_path>'
```
  - 将 `<code_path>` 替换为您要简化的代码的路径。
  - 在 Red Hat Enterprise Linux 8 中：

```
$ gofmt -s <code_path>
```
  - 将 `<code_path>` 替换为您要简化的代码的路径。
  - 在 Red Hat Enterprise Linux 9 中：

```
$ gofmt -s <code_path>
```
  - 将 `<code_path>` 替换为您要简化的代码的路径。
- 要应用这些更改，请运行：
  - 在 Red Hat Enterprise Linux 7 上：

```
$ scl enable go-toolset-1.19 'gofmt -w <code_path>'
```
  - 将 `<code_path>` 替换为您要格式的代码的路径。
  - 在 Red Hat Enterprise Linux 8 中：

```
$ gofmt -w <code_path>
```
  - 将 `<code_path>` 替换为您要格式的代码的路径。



- 在 Red Hat Enterprise Linux 9 中：

```
$ gofmt -w <code_path>
```

- 将 **<code\_path>** 替换为您要格式的代码的路径。

### 3.5. 重构代码

您可以使用 **gofmt** 格式工具通过应用任意替换来重构代码。

#### 流程

- 要在给定路径中重构您的代码，请运行：

- 在 Red Hat Enterprise Linux 7 上：

```
$ scl enable go-toolset-1.19 'gofmt -r <rewrite_rule> <code_path>'
```

- 将 **<code\_path>** 替换为您要重构的代码的路径，将 **<rewrite\_rule>** 替换为您要重写它的规则。

- 在 Red Hat Enterprise Linux 8 中：

```
$ gofmt -r <rewrite_rule> <code_path>
```

- 将 **<code\_path>** 替换为您要重构的代码的路径，将 **<rewrite\_rule>** 替换为您要重写它的规则。

- 在 Red Hat Enterprise Linux 9 中：

```
$ gofmt -r <rewrite_rule> <code_path>
```

- 将 **<code\_path>** 替换为您要重构的代码的路径，将 **<rewrite\_rule>** 替换为您要重写它的规则。

- 要应用这些更改，请运行：

- 在 Red Hat Enterprise Linux 7 上：

```
$ scl enable go-toolset-1.19 'gofmt -w <code_path>'
```

- 将 **<code\_path>** 替换为您要格式的代码的路径。

- 在 Red Hat Enterprise Linux 8 中：

```
$ gofmt -w <code_path>
```

- 将 **<code\_path>** 替换为您要格式的代码的路径。

- 在 Red Hat Enterprise Linux 9 中：

```
$ gofmt -w <code_path>
```

- 将 **<code\_path>** 替换为您要格式的代码的路径。

## 3.6. 其他资源

- [官方 gofmt 文档](#)。

## 第 4 章 GO 竞争检测器

Go Toolset 包括 Go 竞争条件，这是用于查找竞争条件的 Go 标准库的工具。请注意，竞争检测器具有显著的运行时资源开销。

### 4.1. 先决条件

- 已安装 Go Toolset。  
如需更多信息，[请参阅安装 Go Toolset](#)。

### 4.2. 使用 GO 竞争检测器

使用 Go 竞争条件检查您的代码是否有竞争条件。

#### 流程

要使用竞争检测器，请运行：

- 在 Red Hat Enterprise Linux 7 上：

```
$ scl enable go-toolset-1.19 'go build -race -o <output_file> <go_main_package>'
```

- 将 **<output\_file>** 替换为您的可执行文件的名称，**<go\_main\_package>** 替换为您要测试的软件包的名称。

- 在 Red Hat Enterprise Linux 8 中：

```
$ go build -race -o <output_file> <go_main_package>
```

- 将 **<output\_file>** 替换为您的可执行文件的名称，**<go\_main\_package>** 替换为您要测试的软件包的名称。

- 在 Red Hat Enterprise Linux 9 中：

```
$ go build -race -o <output_file> <go_main_package>
```

- 将 **<output\_file>** 替换为您的可执行文件的名称，**<go\_main\_package>** 替换为您要测试的软件包的名称。

### 4.3. 其他资源

- [官方 Go 竞争检测器文档](#)。

## 第 5 章 带有 GO TOOLSET 的容器镜像

您可以从 Red Hat Enterprise Linux 容器镜像或 Red Hat Universal Base Images (UBI) 构建自己的 Go Toolset 容器。

### 5.1. RED HAT ENTERPRISE LINUX GO TOOLSET 容器镜像内容

Go Toolset 的 Red Hat Enterprise Linux 7、Red Hat Enterprise Linux 8 和 Red Hat Enterprise Linux 9 容器镜像包括以下软件包：

组件	Version	软件包
Go	1.19	RHEL 7 – go-toolset-1.19.6 RHEL 8 – go-toolset-1.19.4 RHEL 9 – go-toolset-1.19.6

### 5.2. 访问 RED HAT ENTERPRISE LINUX 容器镜像

在运行容器并执行操作前，从红帽 registry 中拉取容器镜像。

#### 流程

要拉取所需的镜像，请运行：

- 对于基于 Red Hat Enterprise Linux 7 的镜像：

```
# podman pull registry.redhat.io/devtools/go-toolset-rhel7
```

- 对于基于 Red Hat Enterprise Linux 8 的镜像：

```
# podman pull registry.redhat.io/rhel8/go-toolset
```

- 对于基于 Red Hat Enterprise Linux 9 的镜像：

```
# podman pull registry.redhat.io/rhel9/go-toolset
```

### 5.3. 访问 RHEL 8 上的 UBI GO TOOLSET 容器镜像

在 RHEL 8 中，安装 UBI Go Toolset 容器镜像来访问 Go Toolset。

另外，您可以将 Go Toolset 安装到 RHEL 8 基本 UBI 容器镜像中。如需更多信息，请参阅从 [RHEL 8 上的基本 UBI 容器镜像访问 Go Toolset](#)。

#### 流程

要从 Red Hat registry 中拉取 UBI Go Toolset 容器镜像，请运行：

```
# podman pull registry.access.redhat.com/ubi8/go-toolset
```

### 5.4. 从 RHEL 8 上的基本 UBI 容器镜像访问 GO TOOLSET

在 RHEL 8 中，Go Toolset 软件包是 Red Hat Universal Base Images (UBI) 存储库的一部分，这意味着您

可以安装 Go Toolset 作为基本 UBI 容器镜像的补充。要保持容器镜像大小小，请只安装单独的软件包，而不是安装整个 Go Toolset。  
或者，您可以安装 UBI Go Toolset 容器镜像来访问 Go Toolset。如需更多信息，请参阅在 [RHEL 8 上访问 UBI Go Toolset 容器镜像](#)。

### 先决条件

- 现有的 Containerfile。  
有关创建 Containerfiles 的详情，请参考 [Dockerfile 参考](#) 页面。

### 流程

- 要创建包含 Go Toolset 的容器镜像，请在 Containerfile 中添加以下行：

```
FROM registry.access.redhat.com/ubi8/ubi:latest
```

```
RUN yum module install -y go-toolset
```

- 要创建仅包含单个软件包的容器镜像，请在 Containerfile 中添加以下行：

```
RUN yum install -y <package-name>
```

- 将 **<package-name>** 替换为您要安装的软件包的名称。

## 5.5. 在 RED HAT ENTERPRISE LINUX 7 中使用容器镜像作为 SOURCE-TO-IMAGE 构建器镜像

您可以在 Red Hat Enterprise Linux 7 上将 Go Toolset 容器镜像用作 Source-to-Image (S2I) 构建器镜像。

### 流程

1. 将 **IMPORT\_URL** 变量设置为指定代码位置的 URL。
2. 要构建 S2I 构建器镜像，请运行 **s2i build** 命令。



### 注意

如果主软件包位置与 **IMPORT\_URL** 变量指定的位置不同，请将 **INSTALL\_URL** 变量设置为 URL，用于指定构建时提供应用主可执行文件的软件包位置。

### 其他资源

- [OpenShift 容器平台镜像创建指南](#)。
- 有关使用 Source-to-Image (S2I) 的更多信息，请参阅 [使用 Red Hat Software Collections 容器镜像](#)。

## 5.6. 其他资源

- [Go Red Hat Container Registry 中的 Toolset 容器镜像](#)。
- [使用 Red Hat Software Collections 容器镜像构建应用程序镜像](#)。

- 有关 Red Hat UBI 镜像的更多信息，请[参阅使用容器镜像](#)。
- 有关 Red Hat UBI 存储库的更多信息，请[参阅通用基础镜像\(UBI\)：镜像、存储库、软件包和源代码](#)。

## 第 6 章 GO TOOLSET 中的更改

Go Toolset 已从 RHEL 7、RHEL 8 和 RHEL 9 版本 1.18 更新至 1.19.6。

主要变更包括：

- 对以下软件包的安全修复：
  - **crypto/tls**
  - **mime/multipart**
  - **net/http**
  - **path/filepath**
- 程序错误修复：
  - **go** 命令
  - linker
  - 运行时
  - **crypto/x509** 软件包
  - **net/http** 软件包
  - **time** 软件包

有关更新的详情，请查看上游 [Go 1.19 发行注记](#)。