

1

Настройка окружения Terraform

Прежде чем приступать к созданию конфигураций Terraform, стоит ознакомиться с лучшими практиками написания конфигураций по методологии *IaC* (*Infrastructure as Code* — *инфраструктура как код*). После этого вы уже сможете установить и настроить локальное окружение разработки для редактирования файлов конфигурации Terraform и применения изменений с помощью Terraform.

В рецептах этой главы вы сначала изучите наиболее важные передовые практики методологии IaC и Terraform, после чего узнаете, как вручную скачать и установить Terraform на компьютере с Windows и как установить этот инструмент в Windows и Linux с помощью скрипта. Мы также рассмотрим, как использовать Terraform в контейнере Docker и как выполнять обновление провайдеров Terraform.

Технические требования

В этой главе вам не потребуются какие-либо особые технические знания. Все действия будут в основном сводиться к использованию *графического пользовательского интерфейса* и выполнению простых скриптов в командной оболочке Linux или Windows. Однако для выполнения примера «Запуск Terraform в контейнере Docker» все же желательно иметь некоторое представление о Docker.

Наконец, в качестве *интегрированной среды разработки* (*Integrated Development Environment, IDE*) для написания конфигураций Terraform мы будем использовать редактор Visual Studio Code, который можно скачать бесплатно по адресу <https://code.visualstudio.com/>.

Исходный код примеров этой главы доступен по адресу <https://github.com/PacktPublishing/Terraform-Cookbook-Second-Edition/tree/main/CHAP01>.

Обзор лучших практик Terraform

Прежде чем переходить к рецептам, иллюстрирующим процесс установки Terraform, необходимо ознакомиться с общепринятыми лучшими практиками использования методологии IaC в сочетании с Terraform.

Первая группа лучших практик относится к методологии IaC и процессу разработки в целом.

- Сохраняйте все файлы конфигураций Terraform и файлы исходного кода с использованием такой системы контроля версий, как GitHub, Azure DevOps или Bitbucket.
- При размещении кода в Git-репозитории придерживайтесь всех рекомендаций по использованию Git, включая советы в отношении ветвления, тегов, фиксации изменений и запросов на извлечение данных.
- Отделяйте друг от друга различные компоненты в своем коде; использование большой единой кодовой структуры усложняет сопровождение и развертывание.
- Выделяйте одинаковый код в отдельные модули и используйте его многократно, чтобы повысить степень повторного использования кода. Следуя принципу *DRY (Don't repeat yourself — «Не повторяйся»)*, старайтесь делать так, чтобы различные компоненты совместно использовали общую бизнес-логику. О реализации модулей в Terraform мы поговорим в главе 7.
- Автоматизируйте внесение инфраструктурных изменений, используя конвейер непрерывной интеграции и развертывания (CI/CD), о котором мы поговорим в главе 13.

Вторая группа лучших практик относится непосредственно к Terraform.

- Явно указывайте в конфигурации требуемую версию двоичного файла Terraform и указывайте версии всех необходимых провайдеров.
- Используйте удаленные хранилища для сохранения и совместного использования файла состояния Terraform. Как это делается, будет показано в нескольких рецептах в главах 8 и 9.
- Вместо жестко заданных значений используйте в конфигурации переменные. Подробнее об этом будет рассказано в главе 2.
- Снабжайте переменные описательным свойством для обозначения их роли.
- Придерживайтесь общепринятого способа организации структуры и именования файлов проекта Terraform с использованием файлов `main.tf`, `version.tf`, `output.tf` и `variables.tf`.
- Чтобы защитить секретные данные, храните их в зашифрованном виде. Подробнее об этом будет рассказано в главе 2.
- Обеспечивайте защиту аутентификационных данных провайдера. В главе 8 будет показано, как это сделать в случае Azure.

- Используйте общепринятый способ именования всех объектов Terraform (ресурсов, переменных, выходных данных и т. д.). Например, используйте символ подчеркивания для разделения слов в имени ресурса.
- Как следует форматируйте код конфигурации Terraform с помощью отступов, чтобы сделать его более читабельным.
- Проверяйте синтаксис кода, используя консольные команды и линтеры.
- Обновляйте провайдеры Terraform, чтобы получать все последние исправления и новые возможности.
- Тестируйте код конфигураций, используя статический анализ кода и интеграционные тесты. Подробнее об этом мы поговорим в главе 11.
- Всегда предварительно запускайте процесс Terraform с помощью команды `dry run`, чтобы посмотреть, какие изменения он внесет.

Это лишь основные лучшие практики IaC и Terraform, помимо которых, конечно, есть и другие. Мы подробно изучим все эти практики по мере ознакомления с главами и рецептами книги.

Для более подробного изучения лучших практик Terraform вы можете обратиться к следующим ресурсам.

- Рекомендуемые практики Terraform от компании HashiCorp: <https://www.terraform.io/cloud-docs/recommended-practices>.
- Лучшие практики Terraform от Google Cloud: <https://cloud.google.com/docs/terraform/best-practices-for-terraform>.
- Сайт, посвященный лучшим практикам Terraform: <https://www.terraform-best-practices.com/>.
- Лучшие практики Terraform от компании Brainboard: <https://docs.brainboard.co/start/cloud-best-practices>.

Теперь вы уже готовы к тому, чтобы установить Terraform.

Как вручную скачать и установить Terraform в Windows

В этом разделе вы узнаете, как скачать и установить Terraform на локальной машине с операционной системой Windows.

Подготовка

Для выполнения этого примера потребуется лишь операционная система Windows.

Что нужно сделать

Выполните следующие шаги.

1. Откройте Проводник Windows. Выбрав подходящее место, создайте папку Terraform, в которой будет храниться двоичный файл Terraform, например C:\Terraform.
2. В браузере перейдите по адресу <https://developer.hashicorp.com/terraform/downloads>.
3. Перейдите на вкладку Windows и щелкните на ссылке AMD64 для скачивания ZIP-архива с файлом Terraform, предназначенным для 64-разрядной версии Windows. Запустится процесс скачивания ZIP-архива на локальную машину¹ (рис. 1.1).

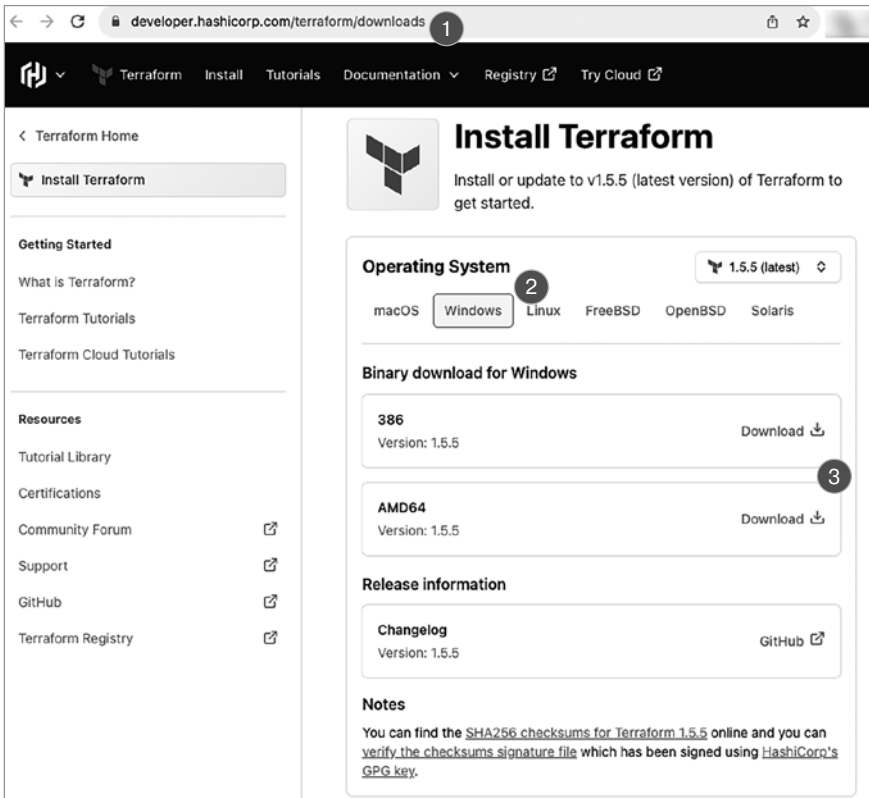


Рис. 1.1. Скачайте двоичный файл Terraform для Windows

¹ На текущий момент скачивание с сайта hashicorp.com в РФ невозможно из-за ограничений. Скачать можно с зеркала: <https://hashicorp-releases.yandexcloud.net/terraform/>. Это справедливо не только для Windows. — Примеч. науч. ред.

4. Распакуйте содержимое скачанного ZIP-архива в папку Terraform, которую мы создали на шаге 1 (рис. 1.2).

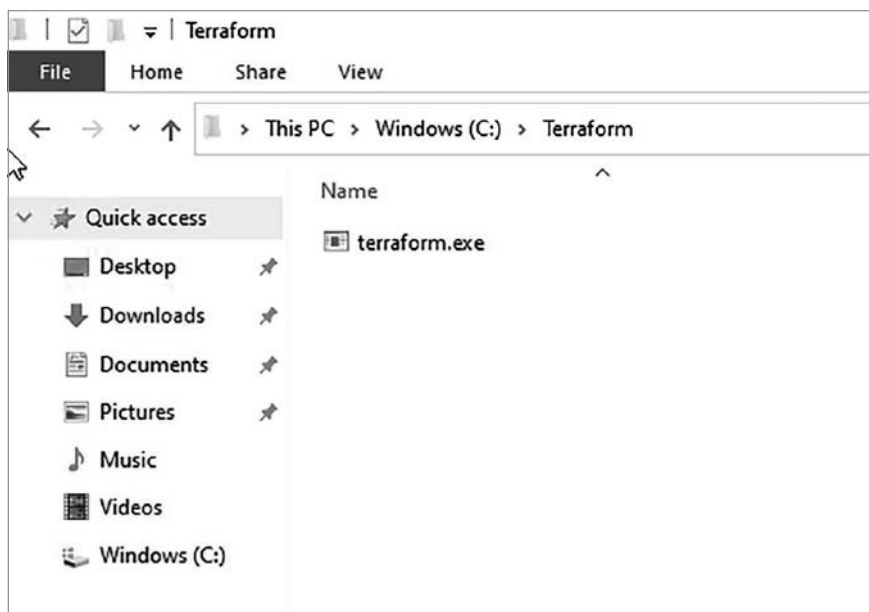


Рис. 1.2. Скопируйте файл Terraform для Windows в соответствующую папку

Теперь для завершения установки нам осталось лишь настроить переменную среды Path, добавив путь к папке с двоичным файлом Terraform.

Переменная среды Path настраивается следующим образом.

1. Открыв Проводник, щелкните правой кнопкой мыши на ссылке Этот компьютер (This PC) и выберите в контекстном меню пункт Свойства (Properties) (рис. 1.3).
2. Щелкните на ссылке Дополнительные свойства системы (Advanced system settings). В открывшемся окне перейдите на вкладку Дополнительно (Advanced) и нажмите кнопку Переменные среды (Environment variables) (рис. 1.4).
3. Выберите переменную Path в списке Переменные среды пользователя (User variables for User) или в списке Системные переменные (System variables) (во втором случае переменная среды будет настроена для всех пользователей компьютера). После этого нажмите кнопку Изменить (Edit). В открывшемся окне нажмите кнопку Создать (New) (рис. 1.5).
4. Добавьте в список путей путь к созданной папке, то есть C:\Terraform\ (рис. 1.6).

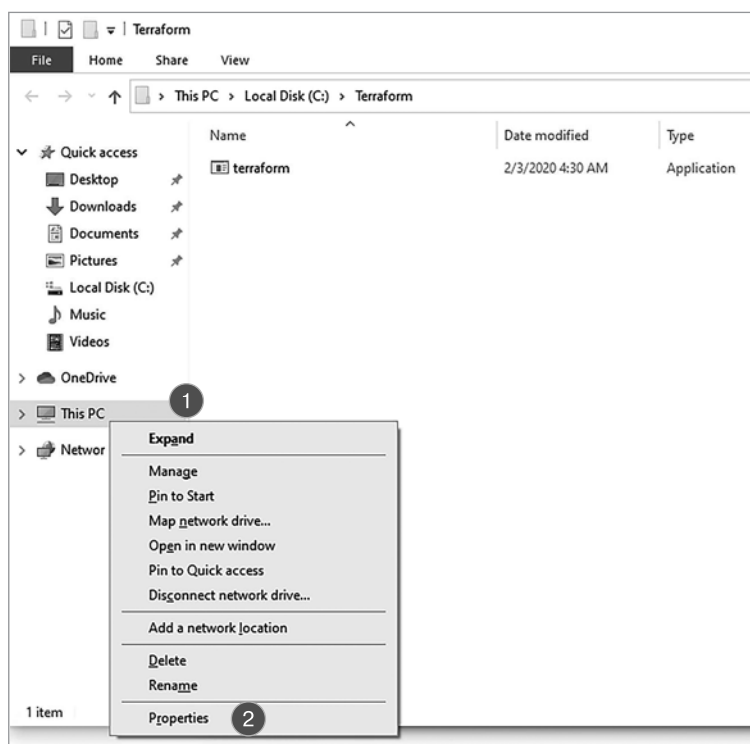


Рис. 1.3. Откройте окно со свойствами системы

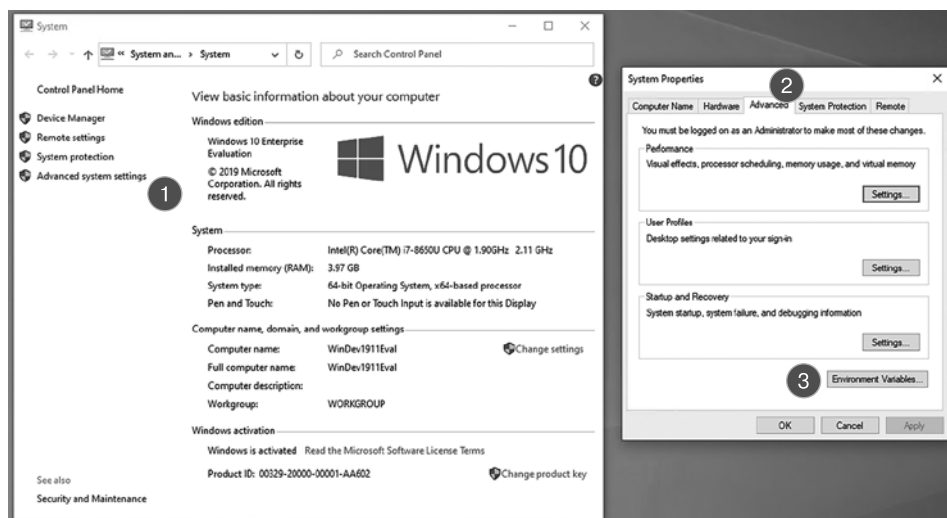


Рис. 1.4. Откройте окно настройки переменных среды

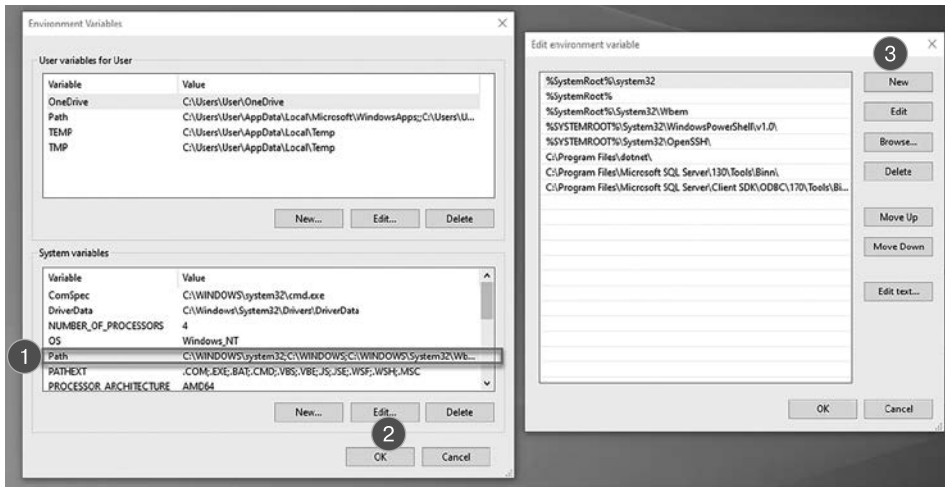


Рис. 1.5. Выберите переменную среды Path

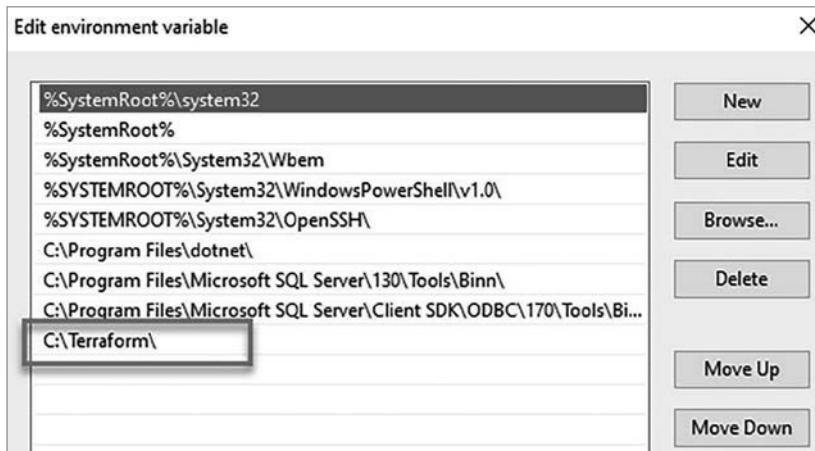


Рис. 1.6. Добавьте в переменную среды Path путь к консольной утилите Terraform

- Чтобы закрыть окно с подтверждением внесенных изменений, нажмите расположенную внизу кнопку OK.

Как это работает

Мы просто скачали и установили Terraform на своем компьютере. Добавив путь к двоичному файлу Terraform в переменную среды PATH, мы сделали возможным выполнение команд Terraform в терминале из любого каталога.

После выполнения этих шагов убедитесь в работоспособности Terraform, выполнив следующую команду в терминале или оболочке PowerShell:

```
terraform --help
```

Результат работы этой команды показан на рис. 1.7.



```
→ terraform --help
Usage: terraform [global options] <subcommand> [args]

The available commands for execution are listed below.
The primary workflow commands are given first, followed by
less common or more advanced commands.

Main commands:
  init          Prepare your working directory for other commands
  validate      Check whether the configuration is valid
  plan          Show changes required by the current configuration
  apply         Create or update infrastructure
  destroy       Destroy previously-created infrastructure

All other commands:
  console       Try Terraform expressions at an interactive command prompt
  fmt           Reformat your configuration in the standard style
  force-unlock  Release a stuck lock on the current workspace
  get           Install or upgrade remote Terraform modules
  graph         Generate a Graphviz graph of the steps in an operation
  import        Associate existing infrastructure with a Terraform resource
  login         Obtain and save credentials for a remote host
  logout        Remove locally-stored credentials for a remote host
  output        Show output values from your root module
  providers     Show the providers required for this configuration
  refresh       Update the state to match remote systems
  show          Show the current state or a saved plan
  state         Advanced state management
  taint         Mark a resource instance as not fully functional
  test          Experimental support for module integration testing
  untaint       Remove the 'tainted' state from a resource instance
  version       Show the current Terraform version
  workspace     Workspace management

Global options (use these before the subcommand, if any):
  -chdir=DIR    Switch to a different working directory before executing the
                given subcommand.
  -help         Show this help output, or the help for a specified subcommand.
  -version      An alias for the "version" subcommand.
```

Рис. 1.7. Список команд Terraform

Появление на экране этого списка команд говорит о том, что установка Terraform прошла успешно и теперь вы можете выполнять его команды в терминале из любого каталога.

Установка Terraform в Windows с помощью Chocolatey

В этом разделе вы узнаете, как установить Terraform на компьютере с Windows с помощью менеджера пакетов *Chocolatey*.

Подготовка

Для выполнения этого примера потребуется операционная система Windows с установленной программой Chocolatey (<https://chocolatey.org/>), которая представляет собой менеджер пакетов для Windows.

Если в вашей системе не установлена программа Chocolatey, вы можете легко ее установить, выполнив следующие шаги.

1. Запустите оболочку PowerShell от имени администратора (рис. 1.8).

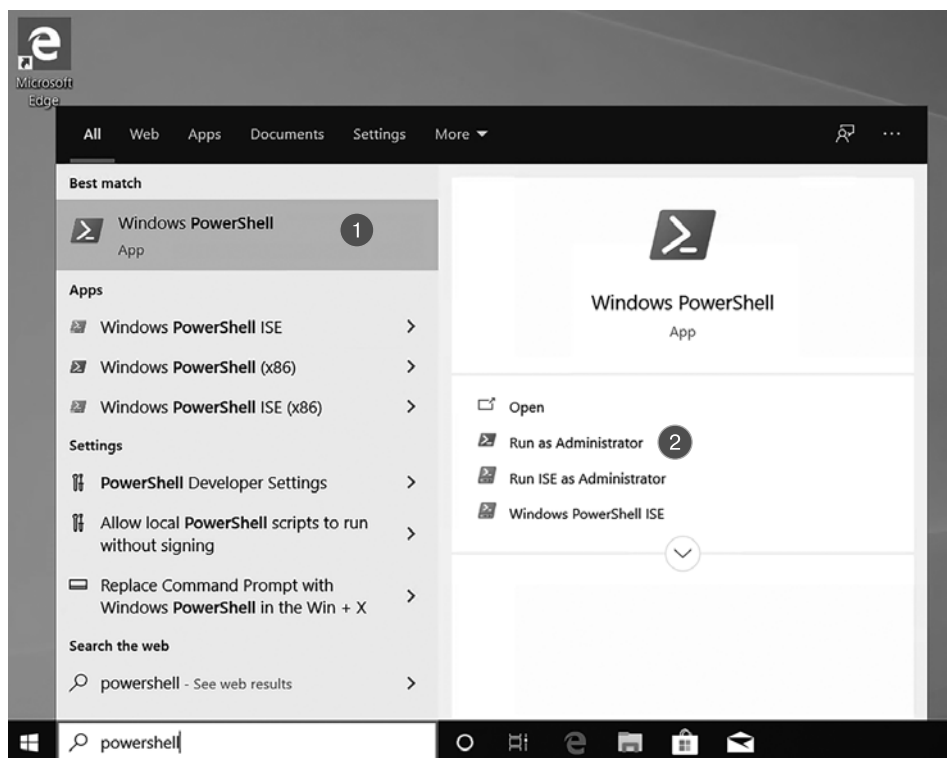


Рис. 1.8. Запуск оболочки PowerShell от имени администратора

2. Теперь выполните в терминале следующий скрипт:

```
Set-ExecutionPolicy Bypass -Scope Process -Force; [System.Net.ServicePointManager]::SecurityProtocol = [System.Net.ServicePointManager]::SecurityProtocol -bor 3072; iex ((New-Object System.Net.WebClient).DownloadString('https://chocolatey.org/install.ps1'))
```



Полная документация по установке Chocolatey доступна по адресу <https://chocolatey.org/install>.

Что нужно сделать

Выполните следующие шаги.

1. Запустите оболочку PowerShell от имени администратора.
2. Выполните следующую команду:

```
choco install -y terraform
```

Результат показан на рис. 1.9.

```
PS C:\WINDOWS\system32> choco install -y terraform
Chocolatey v0.10.15
Installing the following packages:
terraform
By installing you accept licenses for the packages.
Progress: Downloading terraform 1.2.5... 100%

terraform v1.2.5 [Approved]
terraform package files install completed. Performing other installation steps.
Removing old terraform plugins
Downloading terraform 64 bit
  from 'https://releases.hashicorp.com/terraform/1.2.5/terraform_1.2.5_windows_amd64.zip'
Progress: 100% - Completed download of C:\Users\mikael\AppData\Local\Temp\chocolatey\terraform\1.2.5\
terraform_1.2.5_windows_amd64.zip (19.1 MB).
Download of terraform_1.2.5_windows_amd64.zip (19.1 MB) completed.
Hashes match.
Extracting C:\Users\mikael\AppData\Local\Temp\chocolatey\terraform\1.2.5\terraform_1.2.5_windows_amd64.zip
to C:\ProgramData\chocolatey\lib\terraform\tools.
..
C:\ProgramData\chocolatey\lib\terraform\tools
ShimGen has successfully created a shim for terraform.exe
The install of terraform was successful.
  Software installed to 'C:\ProgramData\chocolatey\lib\terraform\tools'

Chocolatey installed 1/1 packages.
See the log for details (C:\ProgramData\chocolatey\logs\chocolatey.log).
```

Рис. 1.9. Установка Terraform с помощью Chocolatey

Параметр `-y` обеспечивает автоматическое подтверждение лицензионного соглашения без вмешательства пользователя.

Как это работает

При установке пакета Terraform менеджер пакетов Chocolatey выполняет скрипты из пакета исходных кодов, расположенного по адресу <https://github.com/jamestoyer/chocolatey-packages/tree/master/terraform>.

Путем выполнения скрипта, расположенного по адресу <https://github.com/jamestoyer/chocolatey-packages/blob/master/terraform/tools/chocolateyInstall.ps1>, Chocolatey скачивает ZIP-архив с файлом Terraform в каталог для двоичных файлов Chocolatey, путь к которому уже добавлен в переменную среды PATH.

Это еще не все

Обновлять Terraform также можно с помощью Chocolatey, используя команду `choco upgrade -y terraform`.

По умолчанию команда `choco install` устанавливает последнюю версию указанного пакета. При желании вы можете прописать версию пакета, добавив в команду параметр `--version`. В нашем случае это будет выглядеть следующим образом:

```
choco install -y terraform --version "1.2.5"
```

Здесь мы указали, что нам нужно установить не последнюю версию пакета Terraform, а версию 1.2.5.

Дополнительные ресурсы

Чтобы ознакомиться с полным списком команд менеджера Chocolatey, обратитесь к документации по адресу <https://chocolatey.org/docs/commands-reference#commands>.

Установка Terraform в Linux с помощью менеджера пакетов APT

В предыдущем разделе вы узнали, как установить Terraform в Windows с помощью скрипта. В этом разделе мы установим Terraform в Linux, используя стандартный менеджер пакетов этой операционной системы *APT (Advanced Packaging Tool)*.

Обратите внимание, что для демонстрации установки в Linux будет использоваться дистрибутив Ubuntu/Debian. Чтобы узнать, как устанавливается Terraform для других дистрибутивов Linux, обратитесь к документации (<https://www.hashicorp.com/official-packaging-guide>).

Подготовка

Для этого примера нам потребуется рабочая станция с Linux и окно терминала.

Что нужно сделать

Чтобы установить Terraform в Linux, используя менеджер пакетов, выполните следующий скрипт в окне терминала:

```
sudo apt update && sudo apt install gpg
wget -O- https://apt.releases.hashicorp.com/gpg | gpg --dearmor | \
sudo tee /usr/share/keyrings/hashicorp-archive-keyring.gpg

gpg --no-default-keyring \
--keyring /usr/share/keyrings/hashicorp-archive-keyring.gpg \
--fingerprint

echo "deb [signed-by=/usr/share/keyrings/hashicorp-archive-keyring.gpg] \
https://apt.releases.hashicorp.com $(lsb_release -cs) main" | \
sudo tee /etc/apt/sources.list.d/hashicorp.list

sudo apt update
sudo apt-get install terraform
```



Исходный код этого скрипта также доступен в GitHub-репозитории книги по адресу https://github.com/PacktPublishing/Terraform-Cookbook-Second-Edition/blob/main/CHAPTER01/install_terraform_package.sh.

Как это работает

Первая команда этого скрипта обновляет индекс пакетов и устанавливает инструменты `gpg`. Также скачивается ключ GPG и проверяется «отпечаток» ключа. Затем регистрируется репозиторий HashiCorp и обновляется индекс пакетов. После этого, наконец, скачивается и устанавливается пакет Terraform.

Дополнительные ресурсы

- Официальное руководство по пакетам Terraform доступно по адресу <https://www.hashicorp.com/official-packaging-guide>.
- Обучающая документация по установке Terraform с помощью пакетного менеджера доступна по адресу <https://learn.hashicorp.com/tutorials/terraform/install-cli>.

Установка Terraform в Linux с помощью скрипта

В этом разделе вы узнаете, как установить Terraform на компьютере с Linux с помощью скрипта.