# Содержание

## Глава 7. Укрепление SSH ..... 257

# Об авторе

Дональд А. Треволт – можно просто Донни – пришел в мир Linux еще в 2006 году да так в нем и остался. Он обладатель сертификата Института профессионалов Linux третьего уровня и сертификата GIAC (Global Information Assurance Certification) по обработке инцидентов. Донни – профессиональный преподаватель Linux, а благодаря волшебству интернета он ведет занятия по всему миру, не покидая своей гостиной. Он также работал исследователем по безопасности в компании, специализирующейся на безопасности интернета вещей (IoT).

*Я благодарю всех добрых людей в компании Packt Publishing, сделавших процесс публикации книги таким гладким. Я также благодарю своих кошек, любезно позволивших использовать свои клички в демонстрациях, и Майка, своего отважного технического рецензента, за предложения, позволившие сделать книгу лучше.*

# О рецензенте

**Майкл Эрнстофф** – специалист по инфраструктуре и безопасности Unix и Linux с 25-летним стажем. Является независимым консультантом уже больше 20 лет. Майкл работал по заказам многих ведущих компаний, преимущественно в банковской и финансовой сферах.

Располагая обширными знаниями в области хостовой безопасности, укрепления безопасности, а также управления идентификацией и доступом, Майкл разрабатывал и внедрял решения для обеспечения безопасности и выполнения нормативных требований.

На досуге любит музицировать, отец четырех детей.

# Предисловие

## *Предполагаемая аудитория*

Книга адресована всем администраторам Linux, независимо от того, специализируются они в области безопасности или нет. Описываемые методы можно использовать как на серверах, так и на рабочих станциях под управлением Linux.

Предполагается, что читатель имеет практический опыт работы с командной строкой и знаком с основами Linux.

## *Структура книги*

В главе 1 «Запуск Linux в виртуальной среде» дается обзор ландшафта ИТ-безопасности. Мы поделимся с читателем своим мнением о том, почему изучение безопасности Linux может положительно сказаться на карьере. А также покажем, как настроить виртуальную среду для практических экспериментов.

В главе 2 «Защита административных учетных записей» рассказано, чем опасна постоянная работа от имени учетной записи root и какие преимущества сулит использование `sudo` вместо этого.

Глава 3 «Защита обычных учетных записей» посвящена безопасности учетных записей обычных пользователей и важности стойких паролей.

В главе 4 «Защита сервера с помощью брандмауэра, часть 1» речь пойдет о работе с разными типами брандмауэров.

В главе 5 «Защита сервера с помощью брандмауэра, часть 2» продолжено обсуждение работы с разными типами брандмауэров.

Глава 6 «Технологии шифрования» посвящена вопросу защиты важной информации – как на диске, так и в процессе передачи – с помощью подходящего шифрования.

В главе 7 «Укрепление SSH» рассматривается, как защитить данные в процессе передачи. Конфигурацию Secure Shell, подразумеваемую по умолчанию, никак не назовешь безопасной, и если ничего не предпринять, она может открыть брешь в системе защиты. В этой главе показано, как это исправить.

В главе 8 «Избирательное управление доступом» познакомимся с тем, как задавать владельцев и права доступа для файлов и каталогов. Мы рассмотрим, чем могут быть полезны биты SUID и SGID, а также последствия их использования для безопасности системы. И завершим главу обсуждением расширенных атрибутов файлов.

В главе 9 «Списки управления доступом и управление разделяемым каталогом» объясняется, что обычные права доступа к файлам и каталогам в Linux недостаточно детализированы. Списки управления доступом позволяют предоставить доступ к файлу только определенному лицу или несколь-

ким лицам, но с разными правами. Мы также применим полученные знания к управлению каталогом, разделяемым членами группы.

Глава 10 «Реализация мандатного управления доступом с помощью SELinux и AppArmor» посвящена технологии мандатного управления доступом SELinux, включенной в дистрибутивы на основе Red Hat Linux. Мы вкратце опишем, как использовать SELinux, чтобы не позволить противнику скомпрометировать систему. А также дадим краткое введение еще в одну технологию мандатного доступа, AppArmor, которая включена в дистрибутивы на основе Ubuntu и SUSE.

В главе 11 «Укрепление ядра и изоляция процессов» рассказывается, как сделать ядро Linux еще более защищенным от атак некоторых типов. Рассматриваются некоторые способы изоляции процессов, предотвращающие эксплойты в Linux.

В главе 12 «Сканирование, аудит и укрепление» речь пойдет о том, что вирусы, представляющие большую проблему в Windows, пока еще не стали таковой в Linux. Если в вашей организации имеются Windows-клиенты, обращающиеся к файловым серверам Linux, то эта глава для вас. Вы можете использовать `auditd` для аудита доступа к файлам, каталогам и системным вызовам в Linux. Это не закроет бреши в системе, но зато вы будете знать, что кто-то пытается получить несанкционированный доступ к конфиденциальной информации. SCAP, протокол автоматизации управления данными безопасности (Security Content Automation Protocol), – это инфраструктура обеспечения соответствия, пропагандируемая Национальным институтом стандартов и технологий США (NIST). Реализацию с открытым исходным кодом, OpenSCAP, можно использовать для применения политики укрепления к компьютеру под управлением Linux.

В главе 13 «Протоколирование и защита журналов» излагаются основы работы с `ryslog` и `journald`, двумя самыми распространенными системами протоколирования в Linux. Мы покажем, как упростить просмотр журналов и как настроить безопасный центральный сервер протоколирования. И для этого нам не потребуется ничего, кроме пакетов, уже входящих в состав большинства дистрибутивов Linux.

В главе 14 «Поиск уязвимостей и обнаружение вторжений» объясняется, как организовать проверку систем на предмет упущений в конфигурациях защиты. Мы также кратко рассмотрим систему обнаружения вторжений.

В главе 15 «Предотвращение запуска нежелательных программ» описано, как с помощью программы `fapolicyd` и параметров монтирования раздела воспрепятствовать исполнению недоверенных программ в системе.

В главе 16 «Полезные советы по безопасности для неутомимых тружеников» констатируется, что всякий, кто занимается безопасностью, трудится как пчелка. И даются полезные советы, как облегчить эту работу.

## *Как извлечь максимум пользы из этой книги*

- Необходимы практические навыки работы с основными командами Linux и ее файловой системой.
- Требуются базовые знания таких средств, как `less` и `grep`.

## Скачайте примеры кода

Весь код, на который есть ссылки в этой книге, размещен на GitHub по адресу https://github.com/PacktPublishing/Mastering-Linux-Security-and-Hardening-3E. На сайте https://github.com/PacktPublishing/ имеется также код для других книг и видео из нашего обширного каталога. Поинтересуйтесь!

## Скачайте цветные изображения

Мы также предлагаем PDF-файл, содержащий цветные изображения всех снимков экрана и рисунков. Его можно скачать по адресу https://packt.link/wcaG3.

## Графические выделения

В этой книге применяется ряд соглашений о наборе текста.

`CodeInText`: код в тексте, имена таблиц базы данных, папок и файлов, расширения имен файлов, пути к файлам, данные и адреса в Твиттере. Например: «откройте Firefox и перейдите по адресу `https://localhost:9392`».

Блок кода выглядит следующим образом:

```
Метод HTTP TRACK активен, а значит, хост уязвим к XST-куку wordpress_test_cookie,
созданному без флага httponly.
```

Входные данные и результаты команд выглядят так:

```
sudo apt update
sudo apt install podman
```

**Полужирный**: новые термины и важные слова, а также части пользовательского интерфейса. Так выделяются команды меню и текст в диалоговых окнах, например: «Задайте для одного режим **Bridged**, а другой оставьте в режиме **NAT**».

Предупреждение и важные замечания выглядят так.

Полезные советы выглядят так.

# —Часть I—

# Подготовка безопасной Linux-системы

В этой части мы настроим «лабораторный стенд» с виртуальными машинами под управлением Ubuntu, CentOS и AlmaLinux.

Пользователи Windows узнают, как удаленно обратиться к Linux-машине из Windows.

# 1

# Запуск Linux в виртуальной среде

Вы, наверное, задаетесь вопросом: «Зачем мне изучать защиту Linux? Разве Linux не безопасна изначально? Ведь это же не Windows». Но на самом деле причин много.

Да, действительно, у Linux есть кое-какие преимущества перед Windows в плане безопасности, а именно:

- в отличие от Windows, Linux с самого начала проектировалась как многопользовательская операционная система. Поэтому с безопасностью в ней дело обстоит немного лучше;
- в Linux лучше организовано разделение между администраторами и непривилегированными пользователями. Это создает некоторые препятствия для злоумышленников, а обычному пользователю случайно заразить Linux чем-то неподобающим чуть сложнее;
- Linux значительно более стойка к вирусам и вредоносным программам, чем Windows. В некоторые дистрибутивы Linux уже встроены механизмы, такие как SELinux в Red Hat и его бесплатных клонах и AppArmor в Ubuntu и SUSE, которые не позволяют вторгшемуся злоумышленнику получить контроль над системой;
- Linux – свободное программное обеспечение с открытым исходным кодом. Это позволяет любому человеку, обладающему достаточными знаниями, провести аудит кода Linux на предмет наличия ошибок или закладок.

Но даже со всеми этими преимуществами Linux не отличается от любого другого творения человека. То есть она несовершенна.

И вот какие вопросы мы рассмотрим в этой главе:

- обзор угроз;
- почему любой администратор Linux должен изучать защиту системы;
- немного о конкретных угрозах с примерами того, как злоумышленникам иногда удавалось взломать систему Linux;

- ресурсы, на которых публикуются актуальные новости о безопасности ИТ;
- различия между физической, виртуальной и облачной системами;
- подготовка Ubuntu Server и виртуальных машин типа Red Hat с помощью VirtualBox, а также установка репозитория **Extra Packages for Enterprise Linux** (**EPEL**) для виртуальных машин типа Red Hat;
- создание моментальных снимков виртуальной машины;
- установка Cygwin на хост-компьютер под управлением Windows, чтобы пользователи Windows могли подключаться к виртуальной машине;
- использование оболочки Bash в Windows 10/11 для доступа к системам Linux;
- поддержание систем Linux в актуальном состоянии.

# Обзор угроз

Если вы следили за ИТ-технологиями на протяжении последних нескольких лет, то, вероятно, встречали хотя бы несколько статей о том, как злоумышленникам удавалось скомпрометировать Linux-серверы. Например, хотя Linux и в самом деле невосприимчива к заражениям вирусами, имело место несколько случаев, когда злоумышленникам удалось внедрить на сервер другие типы вредоносного ПО. Приведем несколько примеров:

- ботнет: сервер заставляют присоединиться к ботнету, контролируемому удаленным злоумышленником. В одном из самых известных случаев такого рода Linux-серверы, присоединившиеся к ботнету, запускали атаку типа «отказ в обслуживании» (DoS-атаку) против других сетей;
- программы-вымогатели: шифруют все пользовательские данные и требуют выкуп за расшифровку. Но даже после уплаты выкупа нет никакой гарантии, что данные можно будет восстановить;
- программы майнинга криптовалют: заставляют процессор сервера-жертвы работать на полную мощность и потреблять больше энергии. Добытая криптовалюта переводится на счета злоумышленников, внедривших программу.

И разумеется, существует немало брешей, не связанных с установкой вредоносного ПО, например когда злоумышленник находит способ украсть учетные данные пользователя, данные кредитных карт и другую конфиденциальную информацию.

Причина некоторых брешей – тривиальная беспечность. В статье по адресу https://arstechnica.com/information-technology/2017/09/in-spectacular-fail-adobe-security-team-posts-private-pgp-key-on-blog/ описывается, как беспечный администратор Adobe разместил закрытый ключ компании в публичном блоге по безопасности.

А теперь поговорим подробнее о брешах в системе защиты.

## Откуда берутся бреши?

Будь то Linux, Windows или что-то еще, причины возникновения брешей в системе защиты в основном одни и те же. Это могут быть ошибки в операционной системе или в приложении, работающем под управлением ОС. Зачастую брешь в системе, вызванную ошибкой, можно было бы предотвратить, если бы администраторы вовремя устанавливали обновления.

Еще одна большая проблема – неправильно сконфигурированные серверы. Стандартная конфигурация Linux-сервера «из коробки» совершенно небезопасна и может породить множество проблем. Одна из причин плохо сконфигурированных серверов – недостаток персонала, должным образом подготовленного к безопасному администрированию Linux-серверов. (И это хорошая новость для читателей данной книги, потому что – поверьте мне – недостатка в хорошо оплачиваемых рабочих местах для профессионалов в области информационной безопасности не наблюдается.)

А теперь Linux устанавливается не только на серверах и настольных компьютерах, но еще и на устройствах, являющихся частью интернета вещей (IoT). А уж с их-то защитой связана куча проблем, главным образом потому, что люди просто не знают, как конфигурировать их безопасно.

По ходу чтения книги мы увидим, как сделать так, чтобы наши серверы были максимально безопасны. И немаловажный шаг в этом направлении – следить за новостями.

## Быть в курсе новостей по безопасности

Все работающие в сфере ИТ, не только системные администраторы, должны следить за новостями в области безопасности. В век интернета это нетрудно.

Во-первых, есть ряд сайтов, специализирующихся на новостях по безопасности, например *Packet Storm Security* и *The Hacker News*. На сайтах технических новостей вообще и новостей Linux в частности, таких как *Ars Technica*, *Fudzilla*, *The Register*, *ZDNet* и *LXer*, также публикуются отчеты о найденных сетевых уязвимостях. А если вы предпочитаете не читать, а смотреть видео, то к вашим услугам немало отличных каналов на YouTube, например *Begin-Linux Guru*.

Наконец, какой бы дистрибутив Linux вы ни использовали, обязательно следите за новостями и текущей документацией по этому дистрибутиву. У лиц, отвечающих за его сопровождение, должен быть способ уведомить вас об обнаружении проблем с безопасностью их продуктов.

Ниже приведены ссылки на несколько хороших сайтов по безопасности:

- Packet Storm Security: https://packetstormsecurity.com/;
- The Hacker News: https://thehackernews.com/.

А вот ссылки на технические сайты более общего характера:

- Ars Technica: https://arstechnica.com/;
- Fudzilla: https://www.fudzilla.com/;
- The Register: https://www.theregister.co.uk/;
- ZDNet: https://www.zdnet.com/.

Можете еще заглянуть на сайты по изучению Linux вообще, а также на сайты новостей из мира Linux:

- LXer: http://lxer.com/;
- BeginLinux Guru on YouTube: https://www.youtube.com/channel/ UC88eard_2sz89an6unmlbeA. (Полное раскрытие информации: я и есть тот самый знаменитый на весь мир BeginLinux Guru.)

И вот о чем следует помнить, читая эту книгу: единственная операционная система, безопасная на все сто процентов, – та, что установлена на компьютере, который никогда не включается.

# Различия между физической, виртуальной и облачной системами

Чтобы вы могли выполнять практикумы, я расскажу о концепции виртуальных машин. Это способ выполнить одну операционную систему внутри другой. Не важно, установлена ли на вашей физической машине (хост-компьютере) Windows, macOS или Linux. В любом случае вы сможете запустить виртуальную машину Linux и практиковаться на ней, не думая о том, что будет, если она «слетит».

Мы будем использовать для этой цели Oracle VirtualBox. Для корпоративных систем существуют другие программы виртуализации, более подходящие для использования в центрах обработки данных. В прошлом серверное оборудование могло заниматься только чем-нибудь одним, т. е. нужно было заводить один сервер для DNS, другой для DHCP и т. д. Но современные серверы располагают гигантскими объемами памяти, кучей места на дисках и процессорами по 96 ядер в каждом. Поэтому дешевле и удобнее установить несколько виртуальных машин на каждый сервер, и пусть каждая решает свою конкретную задачу. Это также означает, что беспокоиться нужно не только о безопасности физического сервера, на котором размещены эти виртуальные машины, но и о безопасности каждой виртуальной машины. И еще добавьте сюда заботу о том, чтобы надлежащим образом изолировать виртуальные машины друг от друга, особенно те, что содержат конфиденциальные данные.

А еще есть облако. Имеется много компаний, предоставляющих облачные службы, где любой человек или компания может запустить экземпляр Windows либо своего любимого дистрибутива Linux. В процессе подготовки дистрибутива Linux в облаке нужно сразу же позаботиться о нескольких вещах ради укрепления защиты (об этом речь пойдет в главе 7). И имейте в виду,

что, обустраивая сервер в облаке, вы всегда добавляете себе хлопот в плане безопасности, потому что один из его интерфейсов смотрит прямо в грубый, неотесанный интернет. (Серверы, установленные на территории предприятия, если не считать тех, что призваны обслуживать внешнюю аудиторию, обычно изолированы от интернета.)

Оставив введение позади, перейдем к реальным материям и начнем со знакомства с программами виртуализации.

# Знакомство с VirtualBox и Cygwin

Когда я что-то пишу или преподаю, я делаю все возможное, чтобы не подать своим студентам лекарство от бессонницы. В этой книге мы встретим теорию там, где без нее не обойтись, но в основном я буду сообщать практически полезную информацию. Не будет недостатка в пошаговых практикумах, местами сдобренных толикой юмора.

Лучший способ выполнять практикумы – воспользоваться виртуальными Linux-машинами. Почти все, что мы будем делать, применимо к любому дистрибутиву Linux, но кое-что возможно только в **Red Hat Enterprise Linux** (**RHEL**) или Ubuntu Linux. (RHEL – самая популярная система для использования на территории предприятия, а Ubuntu – для облачного развертывания.) SUSE – третий из крупных корпоративных дистрибутивов Linux. Мы почти не будем иметь дел с SUSE, но изредка я буду отмечать некоторые из его причуд.

Компания Red Hat стоит миллиард долларов, поэтому сомнений по поводу ее места на рынке Linux не возникает. Но Ubuntu Server распространяется бесплатно, так что судить о его популярности только на основе стоимости материнской компании не получится. Реальность же состоит в том, что Ubuntu Server – самый распространенный дистрибутив Linux для развертывания облачных приложений.

Более подробные сведения см. по адресу http://www.zdnet.com/article/ubuntu-linux-continues-todominate-openstack-and-other-clouds/.

Поскольку Red Hat – платный продукт, мы будем вместо него рассматривать CentOS 7, AlmaLinux8 и AlmaLinux9, которые собраны из исходного кода Red Hat и распространяются бесплатно. (Мы будем использовать все три дистрибутива, потому что между ними есть различия и все они будут поддерживаться на протяжении некоторого времени.) Для CentOS и AlmaLinux предлагаются различные загрузочные образы. Вам лучше скачать образы DVD, т. к. они содержат необходимые компоненты, отсутствующие в минимальных образах. Точнее, скачайте следующие файлы образов:

- CentOS 7: CentOS-7-x86_64-DVD-2009.iso;
- AlmaLinux 8: AlmaLinux-8-latest-x86_64-dvd.iso;
- AlmaLinux 9: AlmaLinux-9-latest-x86_64-dvd.iso.

Для Ubuntu мы ограничимся версией 22.04, т. к. это самая последняя версия с долговременной поддержкой (Long Term Support – LTS). (Иногда мы будем обращаться к версии Ubuntu 20.04, потому что она все еще поддерживается, но имеет несколько отличий от 22.04.) Новая LTS-версия Ubuntu выходит каждый четный год в апреле, а версии без долговременной поддержки – каждый нечетный год в апреле и каждый год в октябре. Для производственной эксплуатации рекомендуется использовать LTS-версии, потому что с другими могут возникать проблемы.

Существует несколько доступных платформ виртуализации, но лично мне больше нравится VirtualBox, бесплатные версии которой существуют для Windows, Linux и Mac. (Есть также версия для Solaris, но я сомневаюсь, что читатели этой книги ее используют.) VirtualBox включает ряд средств, за которые на других платформах надо платить, например возможность создавать моментальные снимки виртуальных машин.

В некоторых практикумах от вас потребуется эмулировать соединение между вашим хост-компьютером и удаленным Linux-сервером. Если ваш компьютер работает под управлением Linux или Mac, то нужно будет просто открыть терминал и воспользоваться встроенной безопасной оболочкой Secure Shell (SSH). Если же хост-компьютер работает под управлением Windows, то нужно будет установить какую-то оболочку Bash, например Cygwin, или просто использовать Bash, встроенную в Windows 10/11 Pro.

## Установка виртуальной машины в VirtualBox

Для тех, кто никогда раньше не работал с VirtualBox, ниже приведен краткий перечень шагов.

1. Скачайте и установите VirtualBox и VirtualBox Extension Pack. Их можно скачать с сайта https://www.virtualbox.org/.
2. Скачайте установочные ISO-файлы для Ubuntu Server 22.04, CentOS 7, AlmaLinux8 и AlmaLinux9. Их можно найти на сайтах https://ubuntu.com/, https://almalinux.org/ и https://www.centos.org/.
3. Запустите VirtualBox и щелкните по значку **New** в верхней части экрана. Заполните требуемые поля. Увеличьте размер виртуального диска до 20 ГБ, но для всех остальных параметров оставьте значения по умолчанию, как показано на рисунке ниже.

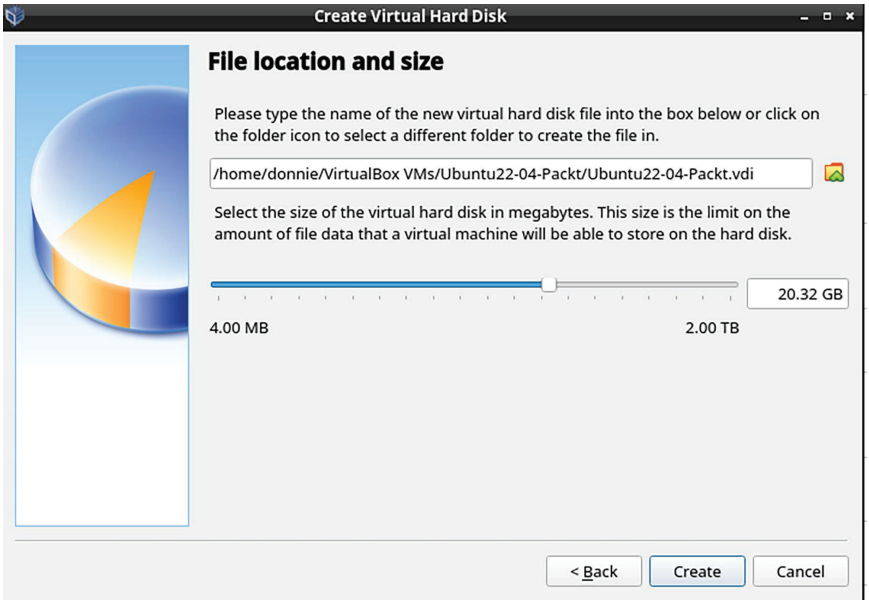


*Рис. 1.1. Создание виртуального диска*

4. Запустите новую виртуальную машину. Щелкните по значку папки справа от поля **Location** и перейдите в каталог, где сохранены скачанные ISO-файлы. Выберите ISO-файл, соответствующий Ubuntu, CentOS или одному из файлов ISO, как показано на рисунке ниже. (Если ISO-файла нет в списке, нажмите кнопку **Add** в левом верхнем углу и добавьте его.)

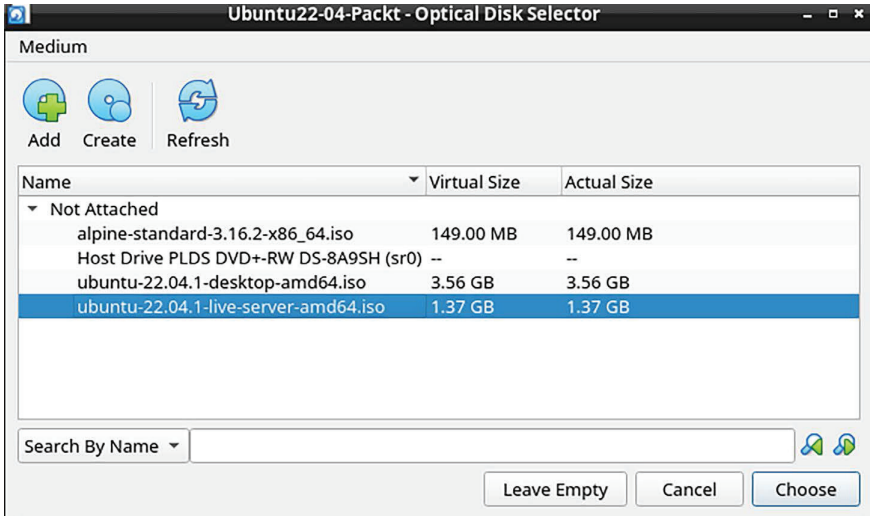


*Рис. 1.2. Выбор iso-файла*

5. Нажмите кнопку `Start` в диалоговом окне, чтобы начать установку операционной системы. Отметим, что для Ubuntu Server интерфейс настольного компьютера устанавливать не надо. Для CentOS 7 и AlmaLinux выберите установку сервера без графического интерфейса. (Впоследствии нам встретится по крайней мере одно упражнение, требующее графического интерфейса для машины AlmaLinux. Тогда и можно будет создать виртуальную машину с графическим интерфейсом.)
6. При установке Ubuntu, дойдя до следующего экрана, выберите **Try or Install Ubuntu Server**.

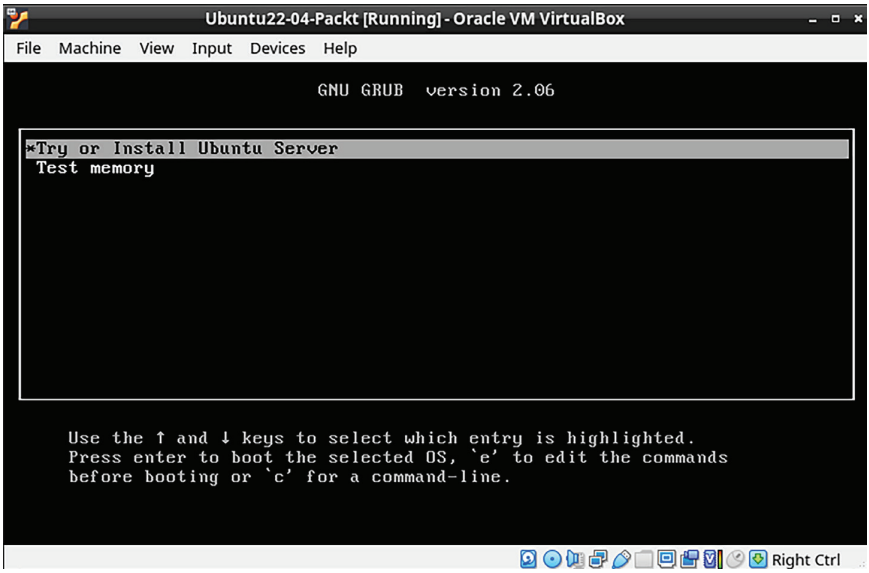


*Рис. 1.3. Установка Ubuntu*

7. Повторите процедуру для других дистрибутивов Linux.
8. Обновите виртуальную машину Ubuntu, выполнив следующие две команды:

```
sudo apt update
sudo apt dist-upgrade
```

9. Не торопитесь обновлять виртуальные машины CentOS и AlmaLinux, мы сделаем это в следующем упражнении.
10. Для Ubuntu выберите установку сервера OpenSSH на экране настройки SSH.

В процессе установки Ubuntu вам будет предложено создать учетную запись обычного пользователя и выбрать для себя пароль. Установщик не будет просить вас создать учетную запись пользователя root, но автоматически добавит вас в группу sudo, так что вы будете иметь привилегии администратора.
Дойдя до экрана создания учетной записи пользователя в установщике CentOS или AlmaLinux, обязательно отметьте флажок **Make this user administrator** (Сделать этого пользователя администратором) для своей учетной записи, т. к. по умолчанию он не отмечен. Вам будет предложено создать пароль для пользователя root, но это необязательно (я никогда этого не делаю).

The user account creation screen of the AlmaLinux 9 installer—which looks the same as the one on CentOS 7 and AlmaLinux 8—is shown here:

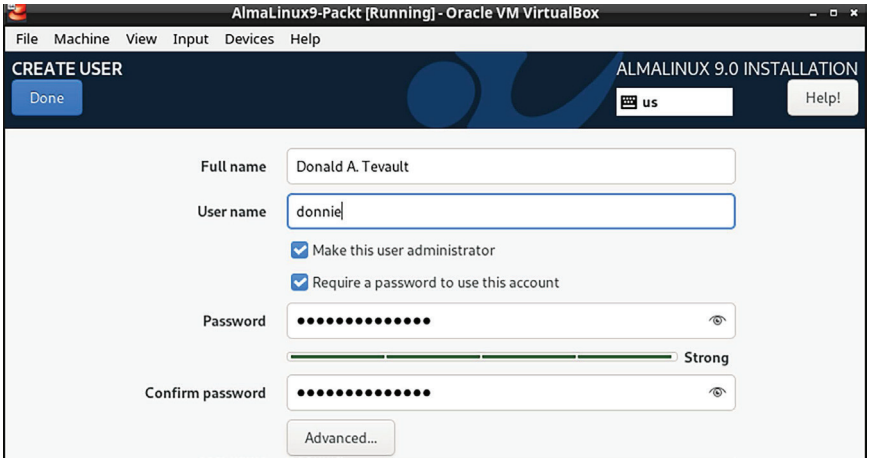


*Рис. 1.4. Создание пользователя в AlmaLinux*

Ниже показан экран создания учетной записи пользователя в установщике AlmaLinux 9, который выглядит так же, как в CentOS 7 и AlmaLinux 8.

**Важно**
Версия RHEL 9.1 и все ее клоны были выпущены спустя несколько месяцев после того, как я написал первую редакцию этой главы. Возможно, вы уже заметили, что в установщике 9.1 есть ошибка, которой не было в установщике 9.0, а именно поля для создания учетной записи обычного пользователя не видны на экране установщика. То есть они присутствуют, только их не видно и добраться до них прокруткой нельзя. Чтобы они появились, продолжайте нажимать клавишу **Tab**, пока не дойдете до поля ввода пароля пользователя root. Затем нажмите **Tab** еще раз, а потом нажмите **Enter**. (Разумеется, вполне возможно, что ошибка будет исправлена к моменту, когда вы будете читать этот текст.)

Для Ubuntu 22.04 вы увидите не нуждающийся в пояснениях экран, на котором можно задать ваше истинное имя, имя пользователя и пароль. Уста-

новщик Ubuntu автоматически добавит вашу учетную запись в группу `sudo`, что даст вам все привилегии администратора.

Ниже показан экран создания учетной записи для Ubuntu 22.04.

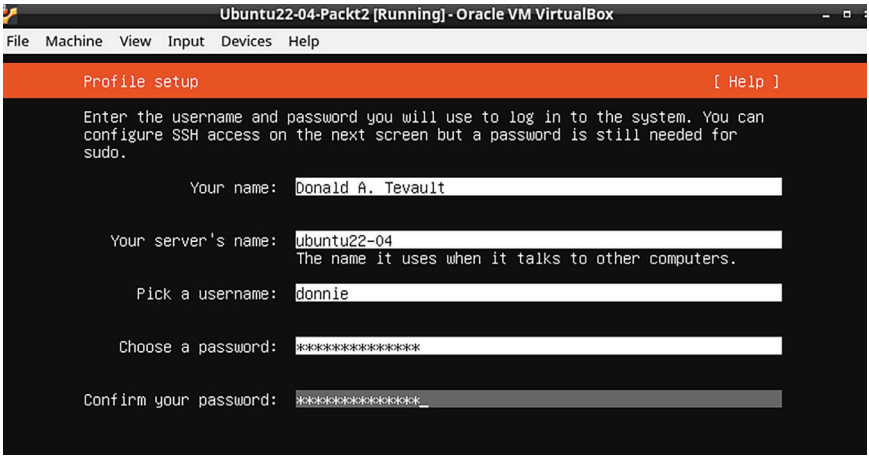


*Рис. 1.5. Создание пользователя в Ubuntu*

Теперь перейдем к CentOS 7.

## Установка репозитория EPEL на виртуальную машину CentOS 7

Если в репозиториях пакетов для Ubuntu есть практически все, что понадобится для чтения этой книги, то репозитории для CentOS и AlmaLinux, скажем так, неполны. Чтобы получить пакеты, необходимые для выполнения практикумов в CentOS и AlmaLinux, придется установить репозиторий EPEL (проект EPEL сопровождается командой Fedora). При установке сторонних пакетов в Red Hat 7 и CentOS 7 также придется установить пакет `priorities` и отредактировать файлы `.repo`, чтобы выставить правильные приоритеты для каждого репозитория. Таким образом, пакеты из сторонних репозиториев не будут иметь преимущества перед одноименными официальными пакетами Red Hat и CentOS. Для установки необходимых пакетов и редактирования файлов `.repo` выполните следующие действия.

1. Два пакета, необходимых для установки EPEL, находятся в стандартных репозиториях CentOS 7. Чтобы установить их, выполните команду

```
sudo yum install yum-plugin-priorities epel-release
```

2. По завершении установки перейдите в каталог `/etc/yum.repos.d` и откройте файл `CentOS-Base.repo` в своем любимом редакторе. После по-

следней строки в секциях `base`, `updates` и `extras` добавьте строку `priority=1`. После последней строки в секции `centosplus` добавьте строку `priority=2`. Сохраните файл и выйдите из редактора. Каждая из отредактированных секций должна иметь примерно такой вид (меняются только название и величина приоритета):

```
[base]
name=CentOS-$releasever - Base
mirrorlist=http://mirrorlist.centos.org/?
release=$releasever&arch=$basearch&repo=os&infra=$infra
#baseurl=http://mirror.centos.org/centos/
$releasever/os/$basearch/
gpgcheck=1
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-CentOS-7
priority=1
```

3. Откройте файл `epel.repo` в редакторе. После последней строки в секции `epel` добавьте строку `priority=10`. А после последней строки в каждой из прочих секций добавьте строку `priority=11`.
4. Обновите систему, а затем создайте список установленных и доступных пакетов, выполнив следующие команды:

```
sudo yum upgrade
sudo yum list > yum_list.txt
```

Теперь перейдем к AlmaLinux.

## Установка репозитория EPEL на виртуальные машины AlmaLinux 8/9

Чтобы установить репозиторий EPEL в AlmaLinux, нужно лишь выполнить следующую команду:

```
sudo dnf install epel-release
```

Пакета `priorities`, как в CentOS 7 и более ранних версиях, здесь нет, поэтому конфигурировать приоритеты репозиториев не нужно.

По завершении установки пакета обновите систему, а затем создайте список установленных и доступных пакетов, выполнив следующие команды:

```
sudo dnf upgrade
sudo dnf list > dnf_list.txt
```

Переходим к конфигурированию сети.

# Конфигурирование сети для виртуальных машин в VirtualBox

В некоторых учебных примерах вам будет предложено эмулировать создание подключения к удаленному серверу. Для этого вы должны будете подключить свой хост-компьютер к виртуальной машине. При первом создании виртуальной машины в VirtualBox устанавливается режим сети **NAT**. Для подключения хоста к виртуальной машине нужно перевести сетевой адаптер виртуальной машины в режим **Bridged Adapter**. Вот как это делается.

1. Остановите все созданные виртуальные машины.
2. На экране VirtualBox Manager откройте диалоговое окно **Settings** (Настройки) для виртуальной машины.
3. Выберите из меню пункт **Network**. Измените значение параметра **Attached to** с **NAT** на **Bridged Adapter**, а параметр **Promiscuous Mode** (Неизбирательный режим) сделайте равным **Allow All** (Разрешать все), как показано на рисунке ниже.

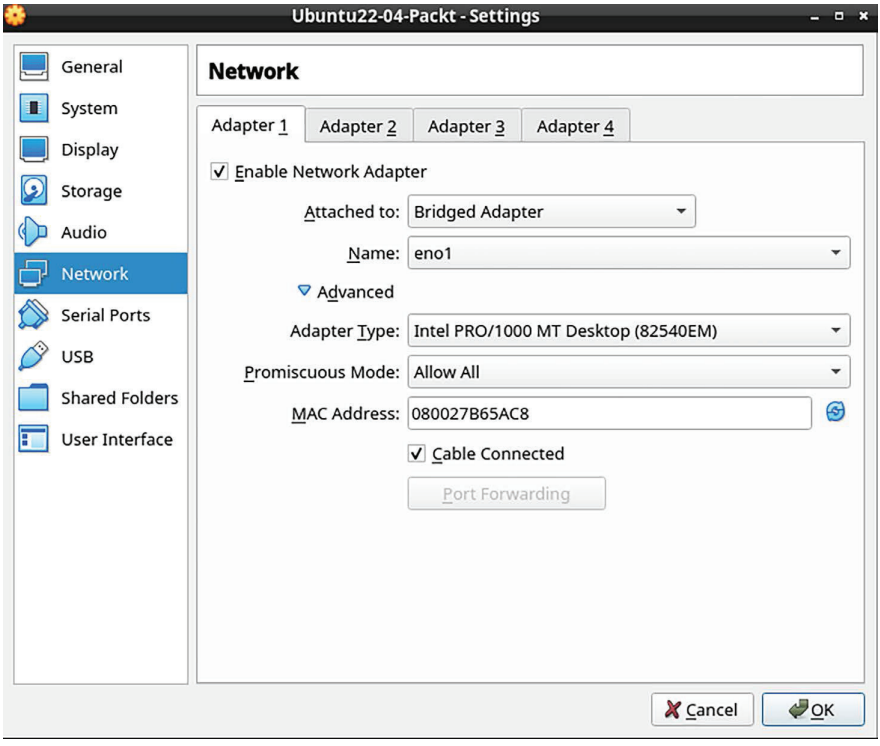


*Рис. 1.6. Конфигурирование сети*

4. Перезапустите виртуальную машину и настройте ее на использование статического IP-адреса.

**Совет**
Если вы будете назначать статические IP-адреса из верхнего конца диапазона подсетей, то будет проще избежать конфликтов с IP-адресами из нижнего конца, которые выдает шлюз к интернету.

# Создание моментального снимка виртуальной машины в VirtualBox

Одна из «приятностей» работы с виртуальными машинами – возможность создать моментальный снимок и откатиться к нему, если что-то пойдет не так. В VirtualBox это легко сделать, выполнив следующие шаги.

1. Из меню **Machine** на экране VirtualBox Manager выберите команду **Tools/Snapshots**
2. Щелкните по значку **Take** в правой части экрана – откроется диалоговое окно **Snapshot**. Либо введите имя в поле **Snapshot Name**, либо оставьте имя по умолчанию. При желании можете ввести описание в поле **Snapshot Description**.

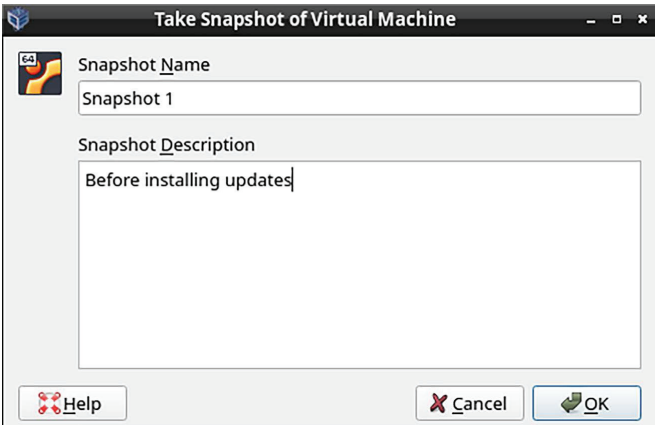


*Рис. 1.7. Создание моментального снимка*

После любых изменений виртуальной машины мы можем откатиться к моментальному снимку; для этого нужно остановить виртуальную машину, выделить имя снимка и нажать кнопку **Restore** (Восстановить).

# Использование Cygwin для подключения к виртуальным машинам

Если ваш компьютер работает под управлением Linux или Mac, просто откройте терминал и воспользуйтесь имеющимися в нем инструментами для подключения к виртуальной машине. В состав Windows 10 и Windows 11, даже в редакции Home Edition, теперь входит клиент **Secure Shell**, встроенный как в обычное окно команд, так и в PowerShell, – если хотите, можете пользоваться им. Но если вы предпочитаете что-то, более близкое к настоящей работе в Linux, то обратите внимание на Cygwin.

Cygwin, проект компании Red Hat, – свободное ПО с открытым исходным кодом, реализующее оболочку Bash для Windows. Оно бесплатно и легко устанавливается.

## *Установка Cygwin на компьютер под управлением Windows*

1. Скачайте файл `setup*.exe`, соответствующий вашей версии Windows, с сайта http://www.cygwin.com/.
2. Дважды щелкните по значку файла, чтобы начать установку. Соглашайтесь с предлагаемыми по умолчанию значениями, пока не дойдете до экрана выбора пакетов. (Есть одно исключение – экран, на котором выбирается зеркало сайта для скачивания.)
3. На экране выбора пакетов выберите из меню **View** пункт **Category**.
4. Раскройте категорию **Net**, как показано на рисунке ниже.

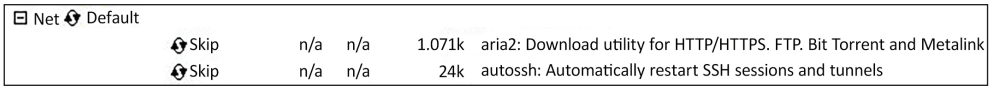


*Рис. 1.8. Установка пакетов Cygwin*

5. Прокрутите экран вниз до пакета `openssh`. В столбце **New** щелкните по слову **Skip** (при этом вместо **Skip** появится номер версии), как показано на рисунке ниже.

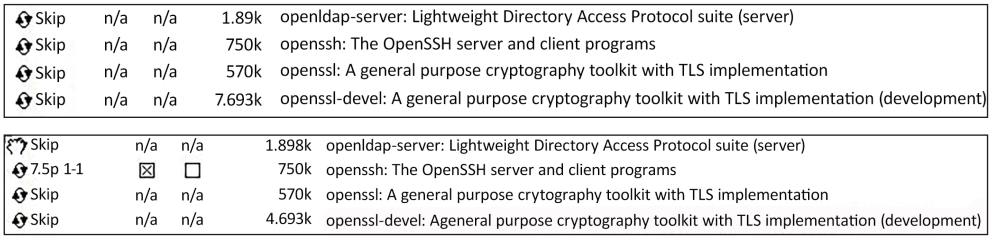


*Рис. 1.9. Выбор пакета OpenSSH*

6. После выбора нужного пакета экран будет выглядеть так:

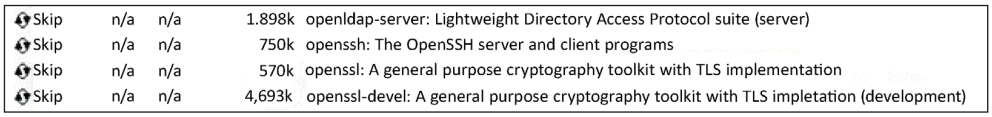


*Рис. 1.10. После выбора пакета OpenSSH*

7. Нажмите кнопку **Next** в правом нижнем углу. Если появится экран **Resolving Dependencies** (Разрешение зависимостей), тоже нажмите **Next**.
8. Сохраните скачанный файл `setup`, потому что позже он понадобится для установки дополнительных пакетов или обновления Cygwin. (При открытии Cygwin все обновленные пакеты будут отображаться в представлении **Pending** в меню **View**.)
9. Если Cygwin открывается из меню **Пуск**, то размер окна можно изменить, а размер шрифта увеличить или уменьшить нажатием комбинаций клавиш **Ctrl**++ или **Ctrl**+– соответственно.

Теперь обратимся к оболочке Bash в Windows 10/11.

# Использование клиента SSH в Windows 10 для взаимодействия с виртуальными машинами Linux

Если вы работаете в Windows 10, то в операционную систему уже встроен SSH-клиент. Посмотрим, как его использовать.

1. Чтобы найти его, откройте традиционное окно командной строки из меню **Windows System**.

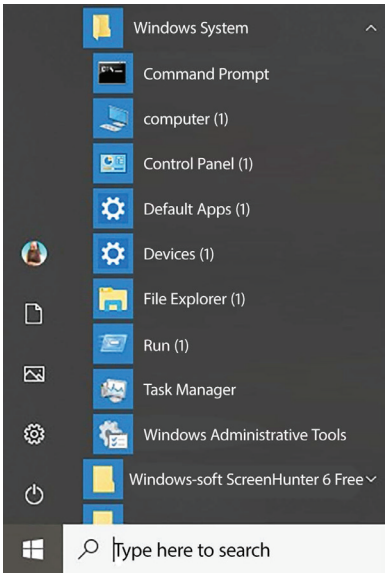


*Рис. 1.11. Командная строка в Windows 10*

2. Затем вводите команды SSH точно так же, как при работе в Mac или Linux.

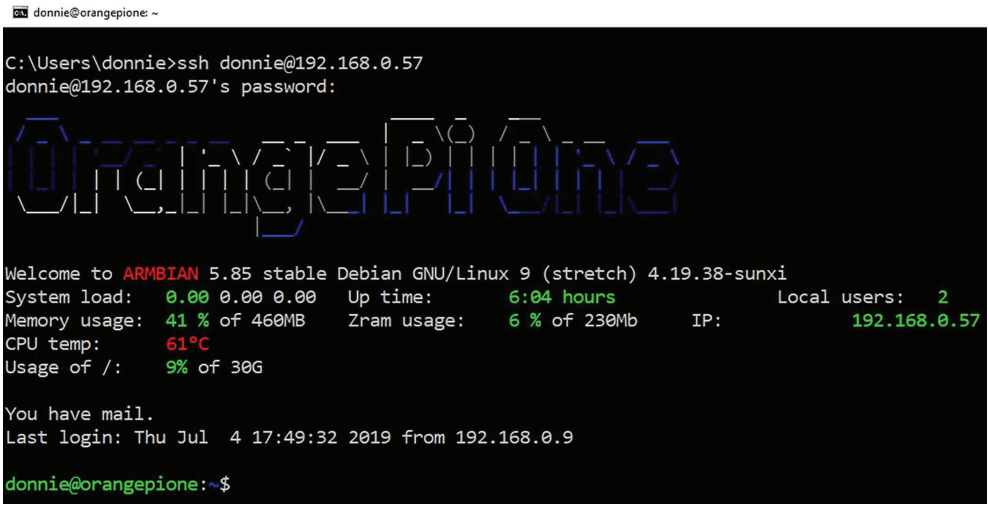


*Рис. 1.12. Подключение к удаленному компьютеру по SSH из командной строки Windows*

3. Еще лучше использовать **Windows PowerShell** вместо обычной командной строки. Как открыть PowerShell, показано ниже.

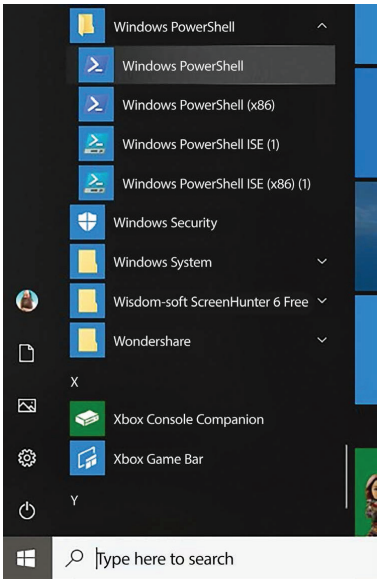


*Рис. 1.13. Командная строка PowerShell*

4. Как и выше, воспользуемся ей, чтобы зайти на мое устройство Orange Pi.

```
donnie@orangepione: ~
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

PS C:\Users\donnie> ssh donnie@192.168.0.57
donnie@192.168.0.57's password:

Welcome to ARMBIAN 5.85 stable Debian GNU/Linux 9 (stretch) 4.19.38-sunxi
System load:   0.00 0.00 0.00   Up time:       5:29 hours              Local users:   2
Memory usage:  41 % of 460MB    Zram usage:    6 % of 230Mb     IP:            192.168.0.57
CPU temp:      62°C
Usage of /:    9% of 30G

[ General system configuration (beta): armbian-config ]

You have mail.
Last login: Thu Jul  4 17:26:18 2019 from 192.168.0.9

donnie@orangepione: $
```

*Рис. 1.14. Удаленный вход из PowerShell*

Если есть выбор, то лучше использовать **PowerShell** вместо командной строки **PowerShell** немного ближе к работе с оболочкой Bash в Linux, и вам это больше понравится.

## Использование клиента SSH в Windows 11 для взаимодействия с виртуальными машинами Linux

Работа в Windows 11 устроена так же, только пункты меню **Command Prompt** и **PowerShell** находятся в других местах. Командной строке теперь отведен отдельный пункт **Terminal** в главном меню, тогда как **PowerShell** переместилась в подменю **Windows Tools**. В Windows 11 появилась третья возможность – встроенная виртуальная машина Ubuntu. Ее значок находится в нижней панели задач.

### *Сравнение Cygwin с оболочкой Windows*

У Cygwin и SSH-клиента, встроенного в Windows 10/11, есть плюсы и минусы. В пользу Cygwin говорит то, что мы можем установить различные пакеты и тем самым настроить ее по собственному вкусу. Кроме того, Cygwin хранит необходимые SSH файлы ключей и `known_hosts` в подкаталоге `.ssh` домашнего каталога пользователя, где мы и ожидаем их найти по опыту работы с Linux. При работе с SSH-клиентом, встроенным в Windows, искать эти файлы придется в других местах.

Аргумент в пользу встроенного в Windows 10/11 SSH-клиента – то, что он уже есть. И пользоваться им гораздо проще, если нужен доступ к привычным папкам Windows, потому что Cygwin не дает выйти за пределы структуры каталогов в своей песочнице.

# Поддержание систем Linux в актуальном состоянии

Потратьте немного времени на изучение базы данных **Common Vulnerabilities and Exposures** (Распространенные уязвимости и риски), и вы очень быстро поймете, почему так важно вовремя обновлять системы. Да-да, вы даже откроете для себя, что в вашей любимой системе Linux обнаруживались изъяны в системе защиты. Вот, пожалуйста:

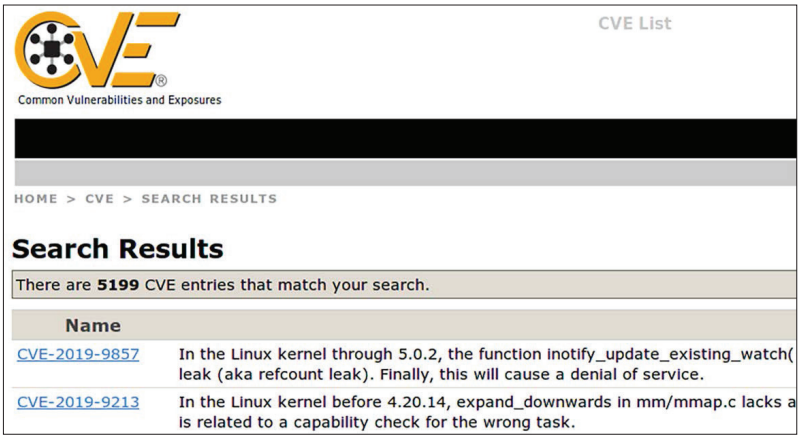


*Рис. 1.15. Распространенные уязвимости и риски*

Для обновления системы Linux нужна всего одна или две простые команды, и обычно все происходит быстрее и не так болезненно, как при обновлении Windows.

Найти базу данных о распространенных уязвимостях и рисках можно по адресу https://cve.mitre.org/.

Любой ответственный, преданный своему делу администратор Linux просто обязан познакомиться с этим сайтом.

Теперь перейдем к обновлению систем на основе Debian, к которым принадлежит и Ubuntu.

## Обновление систем на основе Debian

1. В дистрибутиве Debian и его многочисленных клонах, включая Ubuntu, выполните следующие две команды:

```
sudo apt update
sudo apt dist-upgrade
```

2. Иногда требуется удалить старые, уже ненужные пакеты. Откуда мы об этом знаем? Очень просто. При входе в систему в командной строке появится сообщение. Чтобы удалить старые пакеты, выполните команду

```
sudo apt auto-remove
```

Теперь займемся конфигурированием автоматического обновления в Ubuntu.

# Конфигурирование автоматического обновления в Ubuntu

Сразу после установки Ubuntu 22.04 автоматическое обновление по умолчанию включено. Чтобы убедиться в этом, проверьте состояние службы `unattended-upgrades`:

```
donnie@ubuntu2204-packt:~$ systemctl status unattended-upgrades
• unattended-upgrades.service - Unattended Upgrades Shutdown
     Loaded: loaded (/lib/systemd/system/unattended-upgrades.service; enabled;
vendor preset: enabled)
     Active: active (running) since Sat 2022-10-08 19:25:54 UTC; 52min ago
. . .
. . .
donnie@ubuntu2204-packt:~$
```

Затем загляните в файл `/etc/apt/apt.conf.d/20auto-upgrades`. Если автоматическое обновление включено, то вы увидите такие строки:

```
APT::Periodic::Update-Package-Lists "1";
APT::Periodic::Unattended-Upgrade "1";
```

Должен признаться, что я испытываю смешанные чувства по этому поводу. Конечно, хорошо, что обновления безопасности устанавливаются без моего вмешательства, но многие из этих обновлений не вступают в силу до момента перезагрузки. По умолчанию Ubuntu не перезагружается после установки обновления. Если все так и оставить, то вы увидите сообщение при входе в систему. Но если хотите, может заставить Ubuntu перезагружаться автоматически. Для этого:

1. Перейдите в каталог `/etc/apt/apt.conf.d` и откройте файл `50unattended-upgrades` в любом редакторе. Где-то в районе строки *67* вы увидите такую строку:

   ```
   //Unattended-Upgrade::Automatic-Reboot "false";
   ```

2. Раскомментируйте эту строку, удалив начальные символы косой черты, и измените `false` на `true`:

   ```
   Unattended-Upgrade::Automatic-Reboot "true";
   ```

3. В этой конфигурации Ubuntu будет перезагружаться сразу после завершения процесса автоматического обновления. Если вместо этого вы хотите, чтобы машина перезагружалась в конкретное время, то прокрутите файл до строки *103*:

   ```
   //Unattended-Upgrade::Automatic-Reboot-Time "02:00";
   ```

4. Поскольку эта строка закомментирована, она не дает никакого эффекта. Чтобы машина перегружалась в 2 часа утра, просто раскомментируйте ее. А чтобы она перегружалась, скажем, в 10 вечера, раскомментируйте и измените ее следующим образом:

```
Unattended-Upgrade::Automatic-Reboot-Time "22:00";
```

Конечно, есть старое правило – не устанавливать на производственную систему обновления, не протестировав их сначала на тестовой. Любой поставщик операционных систем время от времени выкладывает проблематичные обновления, и Ubuntu – не исключение. (Я знаю, что вы там бормочете: *не капай мне на мозги, Донни*.) Механизм автоматических обновлений в Ubuntu прямо противоречит этому правилу. Если автоматическое обновление включено, то выключить его очень легко, если вы, конечно, пожелаете.

5. Чтобы выключить автоматическое обновление, перейдите в каталог `/etc/apt/apt.conf.d` и откройте файл `20auto-upgrades` в редакторе. Вот что вы там увидите:

```
APT::Periodic::Update-Package-Lists "1";
APT::Periodic::Unattended-Upgrade "1";
```

6. Измените параметр во второй строке на `0`:

```
APT::Periodic::Update-Package-Lists "1";
APT::Periodic::Unattended-Upgrade "0";
```

Теперь система будет проверять наличие обновлений и показывать сообщение на экране входа, если таковые имеются, но автоматически устанавливать их не станет. И нет нужды повторять, что вы должны регулярно проверять систему на предмет появления обновлений. Если вы все-таки хотите оставить режим автоматического обновления, то либо включите автоматическую перезагрузку, либо возьмите за правило входить в систему по крайней мере пару раз в неделю и смотреть, не надо ли ее перезагрузить.

7. Если вы хотите посмотреть, нет ли каких-нибудь обновлений, связанных с безопасностью, но не хотите видеть прочих обновлений, то воспользуйтесь командой `unattended-upgrade`:

```
sudo unattended-upgrade --dry-run -d
```

8. Чтобы вручную установить обновления, связанные с безопасностью, но не устанавливать прочие обновления, выполните команду

```
sudo unattended-upgrade -d
```

Если вы эксплуатируете какой-нибудь вариант настольной Ubuntu на рабочей станции, которая выключается после каждого использования, то можете, если хотите, включить автоматическое обновление, но включать автоматическую перезагрузку нет нужды.

Также если вы эксплуатируете какой-нибудь клон Debian, отличный от Ubuntu, например Raspbian для Raspberry Pi, то можете наделить его той же функциональностью, что Ubuntu, установив пакет `unattended-upgrades`. Достаточно выполнить команду

```
sudo apt install unattended-upgrades
```

Можно еще использовать команду `apt` для установки только обновлений безопасности, но для этого придется организовать конвейер, состоящий из `apt` и запутанного набора текстовых фильтров, исключающих обновления, не относящиеся к безопасности. Команда `unattended-upgrade` гораздо проще.

Выше я сказал, что обновления всегда нужно проверять на тестовой системе, прежде чем устанавливать их на производственную, и это, безусловно, верно для корпоративных серверов. Но что делать, если имеется целая стая IoT-устройств, которые желательно поддерживать в актуальном состоянии, но эти устройства находятся неизвестно где, да еще и встроены в бытовые приборы?

В дивном мире интернета вещей наиболее популярны дистрибутивы Linux в виде версий Ubuntu, Raspbian и Debian для процессоров ARM, которые работают в разнообразных устройствах на платформе Pi, включая вездесущую Raspberry Pi. Если у заказчиков и в бытовых приборах установлено много ваших IoT-устройств, то прямого контроля над ними после продажи и развертывания у вас может и не быть. Но обновлять их все равно нужно, поэтому установить режим необслуживаемого обновления с автоматическим перезапуском в этой ситуации было бы разумно. Но помните, что в мире IoT важна не только информационная, но и физическая безопасность. Так, например, если устройство работает как особо ответственный промышленный контроллер, эксплуатация которого напрямую связана с безопасностью объекта и людей, то вряд ли вы захотите, чтобы оно автоматически перезагружалось после автоматического обновления. Но если вы производите смарт-телевизоры и устанавливаете на них Linux, то определенно стоит настроить автоматическое обновление с последующей автоматической перезагрузкой.

Следующая наша тема – обновление систем RHEL 7.

# Обновление систем на основе Red Hat 7

В системах на основе Red Hat, включающих CentOS и Oracle Linux, не предусмотрен механизм автоматического обновления, который можно было бы настроить в процессе установки. Так что в конфигурации по умолчанию обновлять систему нужно самостоятельно.

1. Чтобы обновить систему на основе Red Hat 7, выполните всего одну команду:

```
sudo yum upgrade
```

2. Иногда требуется просто посмотреть, есть ли какие-нибудь обновления безопасности, готовые к установке. Для этого выполните команду

```
sudo yum updateinfo list updates security
```

3. Если обновления безопасности имеются, то вы увидите их в конце выдачи. В системе, которую я проверял, было всего одно такое обновление, и выглядело это следующим образом:

```
FEDORA-EPEL-2019-d661b588d2 Low/Sec. nagios-common-4.4.3-1.el7.x86_64
updateinfo list done
```

4. Если вы хотите установить обновления безопасности и только, выполните следующую команду:

```
sudo yum upgrade --security
```

5. Теперь предположим, что вы хотите, чтобы система CentOS обновлялась автоматически. Вам повезло – специально для этой цели есть пакет. Установите его, активируйте и запустите, выполнив следующие две команды:

```
sudo yum install yum-cron
sudo systemctl enable --now yum-cron
```

6. Для конфигурирования пакета перейдите в каталог `/etc/yum` и откройте файл `yum-cron.conf` в редакторе. В начале файла есть такие строки:

```
[commands]
# What kind of update to use:
# default = yum upgrade
# security = yum --security upgrade
# security-severity:Critical = yum --sec-severity=Critical upgrade
# minimal = yum --bugfix update-minimal
# minimal-security = yum --security update-minimal
# minimal-security-severity:Critical = --sec-severity=Critical updateminimal
update_cmd = default
```

Здесь перечислены различные типы обновлений. Последняя строка показывает, что пакет настроен на обновление всего.

7. Предположим, что мы хотим, чтобы только обновления безопасности применялись автоматически. Тогда достаточно изменить последнюю строку:

```
update_cmd = security
```

8. Строки 15 и 20 выглядят так:

```
download_updates = yes
apply_updates = no
```

9. Это означает, что по умолчанию `yum-cron` автоматически лишь скачивает обновления, но не устанавливает их.
10. Если вы хотите, чтобы обновления устанавливались автоматически, то измените параметр `apply_updates` на `yes`.

Отметим, что в отличие от Ubuntu не существует параметра, который заставляет систему автоматически перезагружаться после обновления.

11. Наконец, посмотрим на почтовые настройки `yum-cron`, которые находятся в строках 48–57 файла `yum-cron.conf`:

```
[email]
# The address to send email messages from.
# NOTE: 'localhost' will be replaced with the value of system_name.
email_from = root@localhost
# List of addresses to send messages to.
email_to = root
# Name of the host to connect to to send email messages.
email_host = localhost
```

Как видим, строка `email_to =` настроена так, чтобы сообщения отправлялись учетной записи пользователя root. Если вы хотите получать сообщения сами, то подставьте сюда имя своей учетной записи.

12. Для просмотра сообщений необходимо установить почтового клиента, если он еще не установлен. (Так будет, если при установке операционной системы был выбран режим **Minimal installation**.) Лучше всего установить программу `mutt`:

```
sudo yum install mutt
```

13. В mutt сообщение будет выглядеть следующим образом:

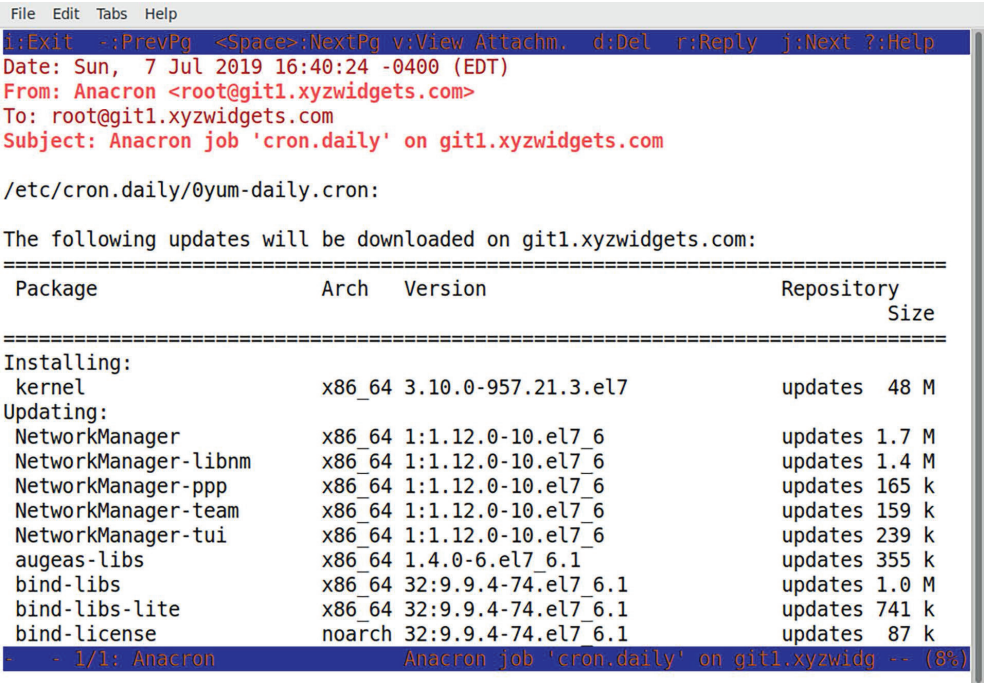


*Рис. 1.16. Почтовый клиент mutt*

14. Как всегда, после некоторых обновлений систему необходимо перезагрузить. А как узнать, что нужна перезагрузка? С помощью команды `needs-restarting`, конечно. Но сначала проверьте, что она установлена. Для этого выполните команду

```
sudo yum install yum-utils
```

После того как пакет установлен, есть три способа воспользоваться командой `needs-restarting`. Выполнив команду без флагов, вы увидите, какие службы необходимо перезапустить и какие пакеты требуют перезагрузки машины. Можно также использовать флаги `-s` и `-r`, как показано в таблице ниже.

| Команда | Пояснение |
|---|---|
| `sudo needs-restarting` | Показывает, какие службы нужно перезагрузить и причины перезагрузки системы |
| `sudo needs-restarting -s` | Показывает только службы, которые нужно перезагрузить |
| `sudo needs-restarting -r` | Показывает только причины перезагрузки системы |

Перейдем к вопросу обновления систем на основе Red Hat 8/9.

# Обновление систем на основе Red Hat 8/9

Старая добрая утилита `yum` существует едва ли не вечно и доказала свою полезность. Но и у нее есть свои странности, а временами она работает мучительно долго. Однако не стоит печалиться. У наших друзей из Red Hat наконец-то дошли до этого руки, и они заменили `yum` на `dnf`. Так что при работе с виртуальными машинами AlmaLinux 8/9 вы будете использовать `dnf` вместо `yum`. Посмотрим, как это делается.

1. В большинстве случаев `dnf` используется точно так же, как `yum`, с теми же флагами и аргументами. Например, для перехода на следующую версию системы достаточно написать:

```
sudo dnf upgrade
```

2. Основное функциональное различие между `yum` и `dnf` заключается в том, что у `dnf` другой механизм автоматического обновления. Вместо установки пакета `yum-cron` нужно установить пакет `dnf-automatic`:

```
sudo dnf install dnf-automatic
```

3. В каталоге `/etc/dnf` имеется файл `automatic.conf`, который конфигурируется так же, как файл `yum-cron.conf` в CentOS 7. Но `dnf-automatic` не является заданием `cron`, как `yum-cron`, а работает с таймером `systemd`. Сразу после установки `dnf-automatic` таймер выключен. Включите его и запустите, выполнив следующую команду:

```
sudo systemctl enable --now dnf-automatic.timer
```

4. Проверьте, что он работает, с помощью команды

```
sudo systemctl status dnf-automatic.timer
```

5. Если таймер запустился успешно, то вы увидите что-то вроде:

```
[donnie@redhat-8 ~]$ sudo systemctl status dnf-automatic.timer
 dnf-automatic.timer - dnf-automatic timer
   Loaded: loaded (/usr/lib/systemd/system/dnf-automatic.timer; enabled;
vendor preset: disabled)
   Active: active (waiting) since Sun 2019-07-07 19:17:14 EDT; 13s ago
  Trigger: Sun 2019-07-07 19:54:49 EDT; 37min left

Jul 07 19:17:14 redhat-8 systemd[1]: Started dnf-automatic timer.
[donnie@redhat-8 ~]$
```

Чтобы узнать, нужно ли перезагрузить систему, установите пакет `yum-utils` и выполните команду `needs-restarting`, как мы делали в CentOS 7. (По каким-то причинам разработчики Red Hat не озаботились изменением имени пакета на `dnf-utils`.)

Для получения дополнительных сведений о `dnf-automatic` введите команду

```
man dnf-automatic
```

Вот и всё.

Автоматическое обновление – отличная штука, не правда ли? Ну, да, иногда. На своих личных рабочих станциях Linux я всегда этот режим выключаю. Потому что меня бесит, что всякий раз, как я хочу установить какой-то пакет, машина говорит мне подождать, пока завершится процесс обновления. В корпоративной системе иногда тоже желательно выключить автоматическое обновление, чтобы у администраторов было больше контроля над процессом обновления.

Теперь рассмотрим некоторые специальные соображения по поводу обновления в корпоративной среде.

## Управление обновлениями на предприятии

Сразу после установки любого дистрибутива Linux он конфигурируется для доступа к собственным репозиториям пакетов, что позволяет пользователям устанавливать пакеты и обновления из этих репозиториев. Это здорово для дома и малых предприятий, но для крупного бизнеса уже не так хорошо.

В корпоративной среде есть два дополнительных соображения:

- мы хотим ограничить набор пакетов, доступных для установки конечными пользователями;
- мы хотим в обязательном порядке проверять обновления в отдельной тестовой сети, прежде чем разрешать установку в рабочей сети.

В силу этих причин предприятия часто заводят собственные серверы репозиториев, на которые выкладываются только одобренные пакеты и обновления. Все остальные машины в сети конфигурируются так, чтобы брать пакеты и обновления с этих серверов, а не из стандартного репозитория, прописанного в дистрибутиве. (Мы не будет здесь описывать, как настраиваются локальные серверы репозиториев, потому что эта тема больше подходит для книги по администрированию Linux.)

Ubuntu всегда был одним из самых инновационных дистрибутивов Linux, но и проблем с контролем качества в нем было изрядно. На заре становления было по меньшей мере одно обновление Ubuntu, которое полностью выводило из строя операционную систему, так что пользователю приходилось переустанавливать все с нуля. Так что, уж пожалуйста, в ответственных системах тестируйте обновления, прежде чем применять их к рабочей системе.

Я думаю, этого достаточно для вводной главы. Подведем итоги.

# Резюме

Мы неплохо начали путешествие в мир защиты и укрепления Linux. В этой главе мы рассмотрели, почему о защите и укреплении Linux знать так же важно, как о защите и укреплении систем Windows. Мы привели несколько примеров того, как можно скомпрометировать плохо сконфигурированную систему Linux, и отметили, что изучение вопросов защиты Linux может положительно сказаться на вашей карьере. Затем обсудили настройку серверов Linux в качестве виртуальных машин и в облаке.

После этого мы поговорили о том, как настроить виртуальную лабораторную среду с применением VirtualBox, Cygwin и оболочки, встроенной в Windows 10/11. И завершили главу кратким обсуждением того, как поддерживать систему Linux в актуальном состоянии.

В следующей главе мы рассмотрим, как защитить учетную запись пользователя и гарантировать, что неподходящие лица никогда не получат привилегий администратора.

# Вопросы

1. Поскольку Linux изначально проектировалась более безопасной, чем Windows, нет нужды тревожиться по поводу безопасности Linux.
   a. Верно.
   b. Неверно.
2. Какие из нижеследующих утверждений об IoT-устройствах под управлением Linux верны?
   a. Их слишком много.
   b. Они захватывают мир.
   c. Слишком многие из них сконфигурированы небезопасно.
   d. Они сконфигурированы настолько безопасно, что оставляют без работы специалистов по безопасности.
3. Какие из нижеследующих утверждений об автоматическом обновлении операционных систем на предприятии верны?
   a. Этот режим всегда следует оставлять включенным.

b. Это нарушает важнейшее правило – проверять обновления в тестовой сети, прежде чем устанавливать их в рабочую.
c. В отличие от ручных обновлений, вам никогда не придется перезагружать систему после автоматического обновления.
d. Для IoT-устройств включать автоматическое обновление вредно.

# Для дополнительного чтения

Ниже перечислено несколько полезных ресурсов для приятного времяпрепровождения.

- Linux Security: https://linuxsecurity.com/.
- Официальный сайт VirtualBox: https://www.virtualbox.org/.
- Официальная страница CentOS: https://www.centos.org/.
- Документация по RHEL (годится также для CentOS и AlmaLinux): https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/9.
- Разрешение автоматических обновлений в RHEL 7 и CentOS 7: https://linuxaria.com/howto/enablingautomatic-updates-in-centos-7-and-rhel-7.
- Управление и мониторинг обновлений безопасности в RHEL 8: https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/8/html/managing_and_monitoring_security_updates/index.

# Ответы

1. b
2. c
3. b

# Присоединяйтесь к сообществу

Присоединяйтесь к нашему сообществу на Discord, где можно обсудить книгу с автором и другими читателями:

https://packt.link/CyberSec