

Содержание

От издательства	17
Об авторе	18
О рецензенте	19
Предисловие	20
Часть I. ПОДГОТОВКА БЕЗОПАСНОЙ LINUX-СИСТЕМЫ	23
Глава 1. Запуск Linux в виртуальной среде	24
Обзор угроз	25
Откуда берутся бреши?	26
Быть в курсе новостей по безопасности.....	26
Различия между физической, виртуальной и облачной системами.....	27
Знакомство с VirtualBox и Cygwin.....	28
Установка виртуальной машины в VirtualBox.....	29
Установка репозитория EPEL на виртуальную машину CentOS 7.....	33
Установка репозитория EPEL на виртуальные машины AlmaLinux 8/9.....	34
Конфигурирование сети для виртуальных машин в VirtualBox.....	35
Создание моментального снимка виртуальной машины в VirtualBox.....	36
Использование Cygwin для подключения к виртуальным машинам.....	37
Установка Cygwin на компьютер под управлением Windows.....	37
Использование клиента SSH в Windows 10 для взаимодействия с виртуальными машинами Linux.....	38
Использование клиента SSH в Windows 11 для взаимодействия с виртуальными машинами Linux.....	41
Сравнение Cygwin с оболочкой Windows.....	41
Поддержание систем Linux в актуальном состоянии.....	41
Обновление систем на основе Debian.....	42
Конфигурирование автоматического обновления в Ubuntu.....	43
Обновление систем на основе Red Hat 7.....	45

Обновление систем на основе Red Hat 8/9.....	49
Управление обновлениями на предприятии	50
Резюме	51
Вопросы	51
Для дополнительного чтения.....	52
Ответы	52
Присоединяйтесь к сообществу	52

Глава 2. Защита административных учетных записей.....

Риски входа в систему от имени root.....	53
Преимущества использования sudo.....	54
Задание привилегий sudo для пользователей с полными правами администратора	56
Добавление пользователей в предопределенную группу администраторов.....	56
Создание записи в файле политики sudo	58
Задание привилегий sudo для пользователей, которым делегирована только часть прав	59
Практикум: назначение ограниченных привилегий sudo	63
Дополнительные приемы работы с sudo.....	64
Таймер sudo	65
Просмотр своих привилегий sudo.....	65
Практикум: отключение таймера sudo	66
Предотвращение доступа к оболочке root со стороны пользователей.....	67
Предотвращение выхода пользователей в оболочку	67
Предотвращение запуска других опасных программ пользователями	70
Ограничение действий пользователя при вызове команд	71
Разрешение пользователям работать от имени других пользователей.....	72
Предотвращение злоупотреблений с помощью пользовательских скриптов оболочки	73
Обнаружение и удаление учетных записей по умолчанию	74
Новые возможности sudo.....	76
Особенности sudo в SUSE и OpenSUSE.....	76
Резюме	79
Вопросы	79
Для дополнительного чтения.....	80
Ответы	80

Глава 3. Защита обычных учетных записей

Защита домашних каталогов пользователей в Red Hat	81
Защита домашних каталогов пользователей в Debian/Ubuntu.....	83
useradd в Debian/Ubuntu	83
adduser в Debian/Ubuntu	84
Практикум: создание зашифрованного домашнего каталога с помощью adduser.....	86

Задание критериев стойкости паролей	86
Установка и конфигурирование pwquality	87
Практикум: задание критериев сложности пароля	90
Задание срока действия пароля и учетной записи	91
Задание данных об истечении срока действия для useradd в системах типа Red Hat	92
Задание данных о сроке действия для каждой учетной записи в отдельности с помощью useradd и usermod	94
Задание данных о сроке действия для каждой учетной записи в отдельности с помощью chage	96
Практикум: задание данных о сроке действия для учетной записи и пароля	97
Предотвращение атак полным перебором	98
Конфигурирование PAM-модуля pam_tally2 в CentOS 7	99
Практикум: конфигурирование pam_tally2 в CentOS 7	100
Конфигурирование pam_faillock в AlmaLinux 8 и 9	101
Практикум: конфигурирование pam_faillock в AlmaLinux 8 и 9	102
Конфигурирование pam_faillock в Ubuntu 20.04 и Ubuntu 22.04	103
Практикум: конфигурирование pam_faillock в Ubuntu 20.04 и Ubuntu 22.04	104
Блокировка учетных записей	104
Блокировка учетной записи с помощью usermod	105
Блокировка учетной записи с помощью passwd	105
Блокировка учетной записи root	106
Настройка баннеров безопасности	107
Использование файла motd	107
Использование файла issue	108
Использование файла issue.net	109
Обнаружение скомпрометированных паролей	110
Практикум: обнаружение скомпрометированных паролей	113
Системы централизованного управления пользователями	114
Microsoft Active Directory	114
Samba в Linux	115
FreeIPA (управление идентификацией) в дистрибутивах типа RHEL	115
Резюме	116
Вопросы	117
Для дополнительного чтения	118
Ответы	118

Глава 4. Защита сервера с помощью брандмауэра, часть 1..... 119

Технические требования	120
Обзор брандмауэров в Linux	120
Обзор iptables	121
Основы iptables	122
Блокирование ICMP с помощью iptables	126
Блокирование всего, что не разрешено, с помощью iptables	128

Практикум: основы работы с iptables.....	131
Блокирование недопустимых пакетов с помощью iptables	132
Восстановление удаленных правил	139
Практикум: блокирование недопустимых IPv4-пакетов	139
Защита IPv6.....	141
Практикум: работа с ip6tables	144
nftables – более универсальная система для построения брандмауэров	145
Таблицы и цепочки nftables	146
Конфигурирование nftables в Ubuntu	147
Использование команд nft	150
Практикум: работа с nftables в Ubuntu.....	155
Резюме	157
Вопросы	158
Для дополнительного чтения.....	159
Ответы	159

Глава 5. Защита сервера с помощью брандмауэра, часть 2..... 160

Технические требования.....	161
Uncomplicated Firewall для систем Ubuntu	161
Конфигурирование ufw.....	161
Работа с конфигурационными файлами ufw	163
Практикум: основы работы с ufw.....	167
firewalld для систем Red Hat	169
Проверка состояния firewalld	169
Работа с зонами firewalld.....	170
Добавление служб в зону по умолчанию	174
Добавление портов в зону firewalld	178
Блокирование ICMP.....	179
Использование режима паники	182
Протоколирование отброшенных пакетов	182
Использование развитых языковых правил firewalld	184
Правила iptables в firewalld для RHEL/CentOS 7	186
Создание прямых правил firewalld в RHEL/CentOS 7	188
Правила nftables для firewalld в RHEL/AlmaLinux 8 и 9	190
Создание прямых правил firewalld в RHEL/AlmaLinux	191
Практикум: команды firewalld.....	191
Резюме	195
Вопросы	195
Для дополнительного чтения.....	196
Ответы	196

Глава 6. Технологии шифрования 197

GNU Privacy Guard (GPG)	198
Практикум: создание собственных ключей GPG	199

Практикум: симметричное шифрование собственных файлов	201
Практикум: шифрование файлов открытыми ключами	204
Практикум: подписание без шифрования	208
Шифрование разделов с помощью Linux Unified Key Setup (LUKS)	209
Шифрование диска в процессе установки операционной системы	209
Практикум: добавление зашифрованного раздела с помощью LUKS	212
Конфигурирование автоматического монтирования раздела LUKS	216
Практикум: конфигурирование автоматического монтирования раздела LUKS	217
Шифрование каталогов с помощью eCryptfs	218
Практикум: шифрование домашнего каталога для учетной записи нового пользователя	218
Создание частного каталога внутри существующего домашнего каталога	219
Практикум: шифрование других каталогов с помощью eCryptfs	221
Шифрование раздела swap с помощью eCryptfs	223
Использование VeraCrypt для кросс-платформенного разделения зашифрованных контейнеров	224
Практикум: получение и установка VeraCrypt	224
Практикум: создание и монтирование тома VeraCrypt в консольном режиме	225
Работа с VeraCrypt в графическом режиме	228
OpenSSL и инфраструктура открытых ключей	229
Коммерческие удостоверяющие центры	230
Создание ключей, запросов на подписание сертификата и сертификатов	233
Создание самоподписанного сертификата с ключом RSA	233
Создание самоподписанного сертификата с эллиптическим ключом ..	235
Создание ключа RSA и запроса на подписание сертификата	235
Создание EC-ключа и CSR	237
Создание локального УЦ	238
Практикум: настройка УЦ Dogtag	239
Добавление УЦ в операционную систему	243
Практикум: экспорт и импорт сертификата УЦ Dogtag	243
Импорт УЦ в Windows	245
OpenSSL и веб-сервер Apache	245
Укрепление Apache SSL/TLS в Ubuntu	246
Укрепление Apache SSL/TLS в RHEL 9/AlmaLinux 9	247
Задание режима FIPS в RHEL 9/AlmaLinux 9	249
Укрепление Apache SSL/TLS в RHEL 7/CentOS 7	251
Настройка взаимной аутентификации	252
Введение в квантово-стойкие алгоритмы шифрования	252
Резюме	253
Вопросы	254
Для дополнительного чтения	255
Ответы	256

Глава 7. Укрепление SSH	257
Запрет протокола SSH 1.....	258
Создание и управление ключами для входа без пароля.....	258
Создание пользовательского набора ключей SSH	259
Перенос открытого ключа на удаленный сервер.....	262
Практикум: создание и перенос ключей SSH.....	264
Запрет входа от имени root	266
Запрет входа по имени пользователя и паролю	267
Практикум: запрет входа по имени пользователя и паролю	267
Включение двухфакторной аутентификации	268
Практикум: настройка двухфакторной аутентификации в Ubuntu 22.04	269
Практикум: использование Google Authenticator в сочетании с обменом ключами в Ubuntu	271
Практикум: настройка двухфакторной аутентификации в AlmaLinux 8.....	272
Практикум: использование Google Authenticator в сочетании с обменом ключами в AlmaLinux 8	274
Конфигурирование Secure Shell со стойкими алгоритмами шифрования	274
Что такое алгоритмы шифрования в SSH	275
Сканирование с целью узнать, какие алгоритмы SSH разрешены	278
Практикум: сканирование с помощью Nmap	278
Запрещение слабых алгоритмов шифрования SSH.....	280
Практикум: запрещение слабых алгоритмов шифрования SSH в Ubuntu 22.04	280
Практикум: запрет алгоритмов шифрования SSH в CentOS 7.....	281
Задание системных политик шифрования в RHEL 8/9 и AlmaLinux 8/9.....	283
Практикум: задание политик шифрования в AlmaLinux 9	284
Конфигурирование более подробного протоколирования	285
Практикум: конфигурирование более подробного протоколирования SSH.....	286
Конфигурирование управления доступом с помощью белых списков и TCP Wrappers.....	287
Конфигурирование белых списков в sshd_config	288
Практикум: конфигурирование белых списков в sshd_config	288
Конфигурирование белых списков с помощью TCP Wrappers.....	290
Конфигурирование автоматического выхода из системы и баннеров безопасности	291
Настройка автоматического выхода для локальных и удаленных пользователей	291
Настройка автоматического выхода в sshd_config	292
Создание предупредительного баннера безопасности	292
Конфигурирование прочих параметров безопасности.....	293
Запрет проброса X11	293
Запрет SSH-туннелей	294

Изменения порта SSH по умолчанию.....	295
Управление ключами SSH.....	296
Задание разных конфигураций для различных пользователей и групп.....	299
Задание разных конфигураций для различных узлов	300
Задание окружения chroot для пользователей SFTP	301
Создание группы и конфигурирование файла sshd_config	301
Практикум: задание каталога chroot для группы sftputers.....	303
Разделение каталога с помощью SSHFS	304
Практикум: разделение каталога с помощью SSHFS	305
Удаленное подключение с рабочего стола Windows.....	306
Резюме	311
Вопросы	312
Для дополнительного чтения.....	313
Ответы	314

Часть II. УПРАВЛЕНИЕ ДОСТУПОМ К ФАЙЛАМ И КАТАЛОГАМ..... 315

Глава 8. Избирательное управление доступом..... 316

Использование chown для изменения владельца файлов или каталогов.....	316
Использование chmod для задания прав доступа к файлам или каталогам.....	318
Символический способ задания прав доступа	319
Числовой способ задания прав доступа	320
Использование SUID и SGID для регулярных файлов	322
Последствия установки битов SUID и SGID с точки зрения безопасности	323
Нахождение посторонних SUID- и SGID-файлов.....	323
Практикум: поиск SUID- и SGID-файлов	325
Предотвращение использования SUID и SGID в разделе	326
Использование расширенных атрибутов для защиты важных файлов	326
Задание атрибута a.....	327
Задание атрибута i.....	328
Защита системных конфигурационных файлов.....	330
Резюме	333
Вопросы	333
Для дополнительного чтения.....	336
Ответы	336

Глава 9. Списки управления доступом и управление разделяемым каталогом..... 337

Создание ACL для пользователя или группы.....	337
Создание наследуемого ACL для каталога	340
Удаление конкретного права доступа с помощью маски ACL	342

Использование команды <code>tar --acls</code> для предотвращения потери ACL при создании резервной копии.....	343
Создание группы пользователей и добавление в нее членов.....	345
Добавление членов при создании их учетных записей.....	346
Использование <code>usermod</code> для добавления существующего пользователя в группу.....	346
Добавление пользователя в группу путем редактирования файла <code>/etc/group</code>	347
Создание разделяемого каталога.....	348
Установка бита SGID и бита закрепления для разделяемого каталога.....	349
Использование ACL для доступа к файлам в разделяемом каталоге.....	351
Задание прав доступа и создание ACL.....	352
Практикум: создание разделяемого каталога для группы.....	353
Резюме.....	355
Вопросы.....	355
Для дополнительного чтения.....	357
Ответы.....	357

Часть III. Дополнительные методы укрепления системы..... 358

Глава 10. Реализация мандатного управления доступом с помощью SELinux и AppArmor..... 359

Чем SELinux может быть полезна системному администратору.....	360
Настройка контекстов безопасности для файлов и каталогов.....	361
Установка инструментов SELinux.....	363
Создание файлов контента при включенной SELinux.....	364
Исправление неверного контекста SELinux.....	367
Использование <code>chcon</code>	367
Использование <code>restorecon</code>	368
Использование <code>semanage</code>	369
Практикум: установка типа SELinux.....	371
Использование <code>setroubleshoot</code> для отладки проблем в SELinux.....	372
Просмотр сообщений <code>setroubleshoot</code>	372
Использование графической утилиты <code>setroubleshoot</code>	373
Отладка в разрешительном режиме.....	375
Работа с политиками SELinux.....	378
Просмотр булевых признаков.....	378
Конфигурирование булевых признаков.....	380
Защита веб-сервера.....	381
Защита сетевых портов.....	382
Создание специальных модулей политики.....	385
Практикум: булевы признаки SELinux и порты.....	387
Чем AppArmor может быть полезна системному администратору.....	388
Знакомство с профилями AppArmor.....	389
Работа с командными утилитами AppArmor.....	392

Отладка проблем в AppArmor	395
Отладка профиля AppArmor – Ubuntu 16.04	395
Отладка профиля AppArmor – Ubuntu 18.04	398
Практикум: отладка профиля AppArmor	399
Отладка проблем Samba в Ubuntu 22.04	400
Эксплуатация системы с помощью вредоносного контейнера Docker	401
Практикум: создание вредоносного контейнера Docker	402
Резюме	404
Вопросы	405
Для дополнительного чтения	406
Ответы	407
Глава 11. Укрепление ядра и изоляция процессов	408
Файловая система /proc	409
Просмотр процессов, работающих в режиме пользователя	409
Просмотр информации о ядре	411
Задание параметров ядра с помощью sysctl	413
Конфигурирование файла sysctl.conf	414
Конфигурирование sysctl.conf – Ubuntu	415
Конфигурирование sysctl.conf – CentOS и AlmaLinux	419
Задание дополнительных параметров для укрепления ядра	420
Практикум: сканирование параметров ядра с помощью Lynis	420
Запрет пользователям просматривать чужие процессы	423
Что такое изоляция процессов	424
Что такое контрольные группы	425
Что такое изоляция пространств имен	428
Что такое возможности ядра	430
Практикум: задание возможности ядра	433
SECCOMP и системные вызовы	434
Использование изоляции процессов при работе с контейнерами	
Docker	435
Организация песочницы с помощью Firejail	436
Практикум: работа с Firejail	439
Организация песочницы с помощью Snappy	440
Организация песочницы с помощью Flatpak	444
Резюме	447
Вопросы	447
Для дополнительного чтения	449
Ответы	450
Глава 12. Сканирование, аудит и укрепление	451
Установка и обновление ClamAV и maldet	452
Практикум: установка ClamAV и maldet	453
Практикум: конфигурирование maldet	455

Обновление ClamAV и maldet	456
Сканирование с помощью ClamAV и maldet	459
Проблемы SELinux	460
Поиск руткитов с помощью Rootkit Hunter	460
Практикум: установка и обновление Rootkit Hunter	461
Поиск руткитов	462
Быстрый анализ на предмет вредоносности с помощью strings и VirusTotal	463
Анализ файла с помощью strings	464
Сканирование вредоносного файла с помощью VirusTotal	465
О демоне auditd	466
Создание правил аудита	467
Аудит изменений файла	467
Аудит каталога	469
Аудит системных вызовов	470
Использование ausearch и aureport	471
Поиск уведомлений об изменении файла	471
Поиск нарушений правил доступа к каталогам	474
Поиск нарушений правил системных вызовов	478
Генерирование отчетов об аутентификации	480
Использование предопределенных наборов правил	482
Практикум: использование auditd	483
Практикум: использование предопределенных правил для auditd	485
Аудит файлов и каталогов с помощью inotifywait	485
Применение политик OpenSCAP с помощью oscar	487
Установка OpenSCAP	487
Просмотр файлов профилей	488
Получение недостающих профилей для Ubuntu	489
Сканирование системы	489
Лечение системы	491
Использование SCAP Workbench	493
Выбор профиля OpenSCAP	496
Применение профиля OpenSCAP на этапе установки системы	497
Резюме	499
Вопросы	499
Для дополнительного чтения	501
Ответы	501

Глава 13. Протоколирование и защита журналов

Знакомство с системными журналами Linux	503
Системный журнал и журнал аутентификации	503
Файлы utmp, wtmp, btmp и lastlog	506
Знакомство с rsyslog	509
Правила протоколирования в rsyslog	509
Знакомство с journald	511
Упрощение работы с помощью Logwatch	514

Практикум: установка Logwatch.....	514
Настройка сервера удаленного протоколирования.....	516
Практикум: настройка простого сервера протоколирования	516
Создание зашифрованного подключения к серверу протоколирования.....	518
Создание подключения через stunnel в AlmaLinux 9 – сторона сервера	518
Создание подключения через stunnel в AlmaLinux 9 – сторона клиента.....	519
Создание подключения через stunnel в Ubuntu – сторона сервера.....	520
Создание подключения через stunnel в Ubuntu – сторона клиента	522
Разнесение сообщений клиентов по отдельным файлам	523
Обслуживание журналов на крупных предприятиях.....	524
Резюме	525
Вопросы	525
Для дополнительного чтения.....	527
Ответы	527

Глава 14. Поиск уязвимостей и обнаружение вторжений.....

Введение в Snort и Security Onion.....	529
Получение и установка Snort	529
Практикум: установка Snort с помощью контейнера Docker	530
Использование Security Onion	532
IPFire и встроенная в нее система предотвращения вторжений.....	534
Практикум: создание виртуальной машины IPFire	535
Сканирование и укрепление с помощью Lynis	539
Установка Lynis в Red Hat/CentOS.....	540
Установка Lynis в Ubuntu	540
Сканирование с помощью Lynis	540
Поиск уязвимостей с помощью Greenbone Security Assistant	544
Сканирование веб-сервера с помощью Nikto.....	552
Nikto в Kali Linux	552
Практикум: установка Nikto с Github	553
Сканирование веб-сервера с помощью Nikto	554
Резюме	556
Вопросы	556
Для дополнительного чтения.....	557
Ответы	557

Глава 15. Предотвращение запуска нежелательных программ

Монтирование разделов с параметрами по	559
Демон fapolicyd.....	565
Правила fapolicyd	568
Установка fapolicyd	570
Резюме	571

Для дополнительного чтения.....	571
Вопросы.....	571
Ответы.....	572

Глава 16. Полезные советы по безопасности

для неумолимых тружеников	573
Технические требования.....	573
Аудит системных служб.....	574
Аудит системных служб с помощью systemctl	574
Аудит сетевых служб с помощью netstat	575
Практикум: просмотр сетевых служб с помощью netstat	580
Аудит сетевых служб с помощью Nmap.....	581
Состояния портов	582
Типы сканирования	582
Практикум: сканирование с помощью Nmap	587
Парольная защита начального загрузчика GRUB2	588
Практикум: сброс пароля	
для Red Hat/CentOS/AlmaLinux	589
Практикум: сброс пароля для Ubuntu.....	591
Предотвращение редактирования параметров ядра	
в Red Hat/CentOS/AlmaLinux	594
Предотвращение редактирования параметров ядра в Ubuntu.....	595
Отключение подменю для Ubuntu	598
Безопасное конфигурирование BIOS/UEFI	600
Контрольный список мер защиты конфигурации системы.....	602
Резюме	605
Вопросы	605
Для дополнительного чтения.....	607
Ответы	607
Предметный указатель.....	608

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 `syslog` 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 `fail0vcyd` 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.
- Требуется 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-gpg-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://lxaer.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, обустроив 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-todominat-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.

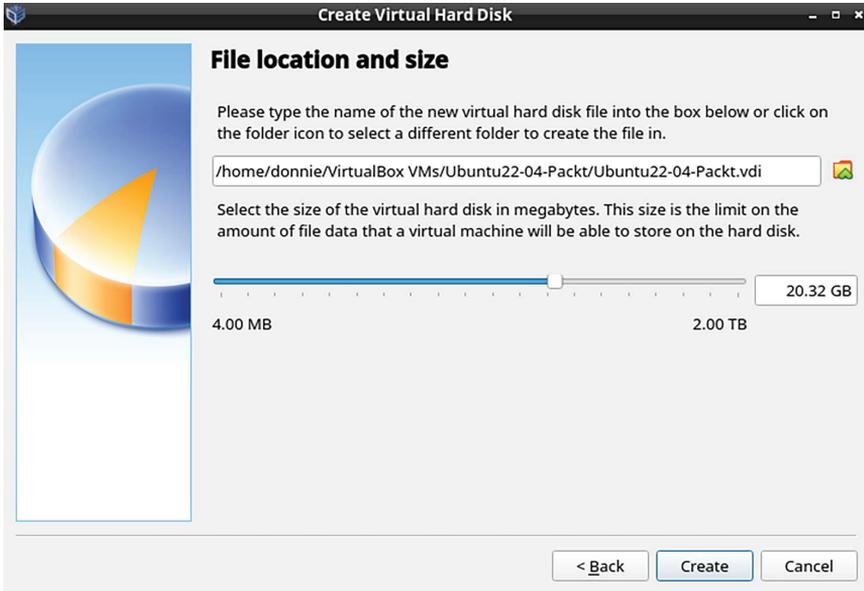


Рис. 1.1. Создание виртуального диска

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.)

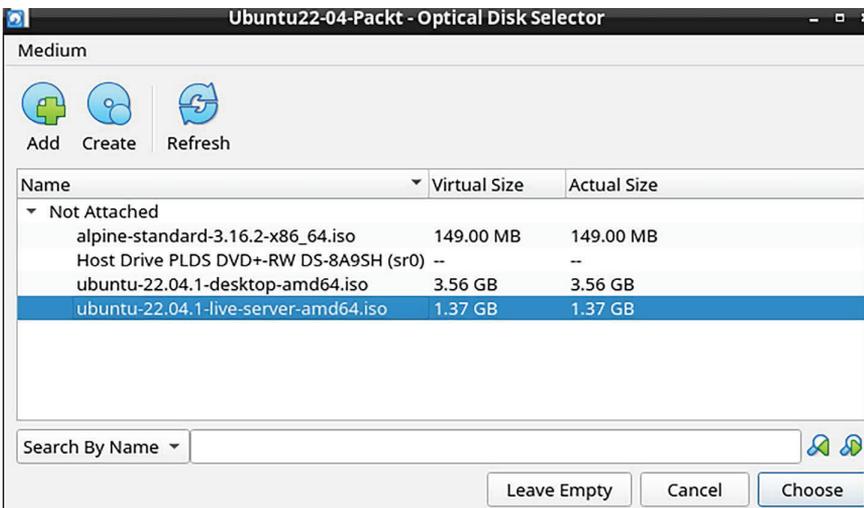


Рис. 1.2. Выбор iso-файла

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**.

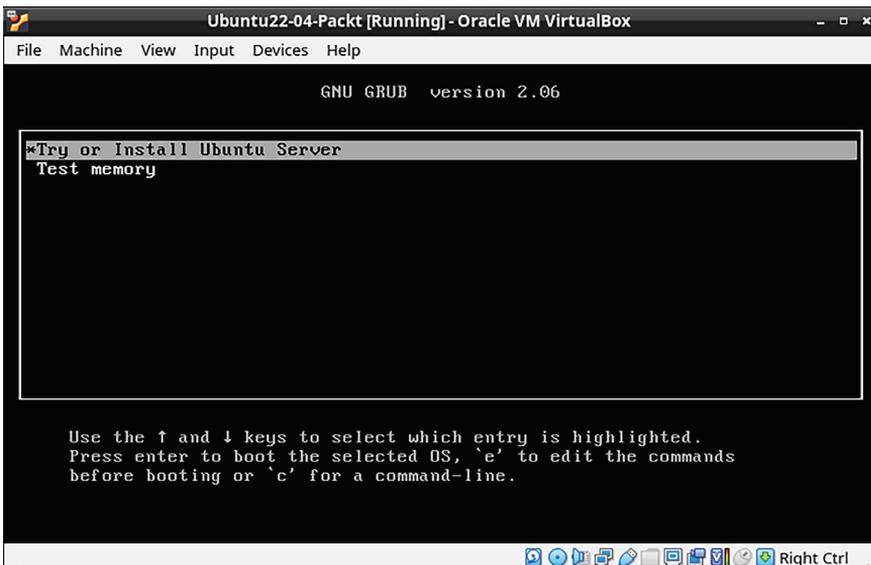


Рис. 1.3. Установка Ubuntu

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:

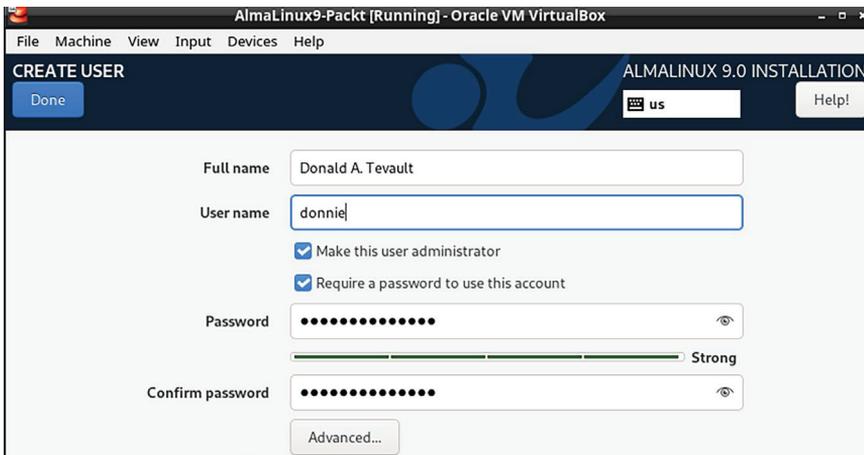


Рис. 1.4. Создание пользователя в AlmaLinux

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-

новичок 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 `.геро`, чтобы выставить правильные приоритеты для каждого репозитория. Таким образом, пакеты из сторонних репозитив не будут иметь преимущества перед одноименными официальными пакетами Red Hat и CentOS. Для установки необходимых пакетов и редактирования файлов `.геро` 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.геро` 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 репозитория 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.

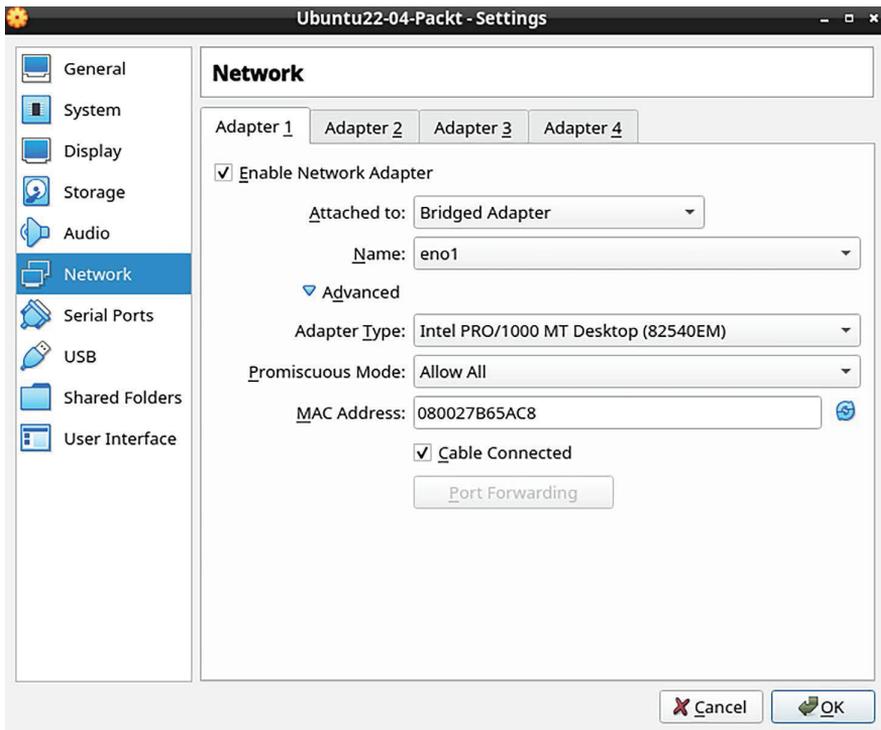


Рис. 1.6. Конфигурирование сети

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**.

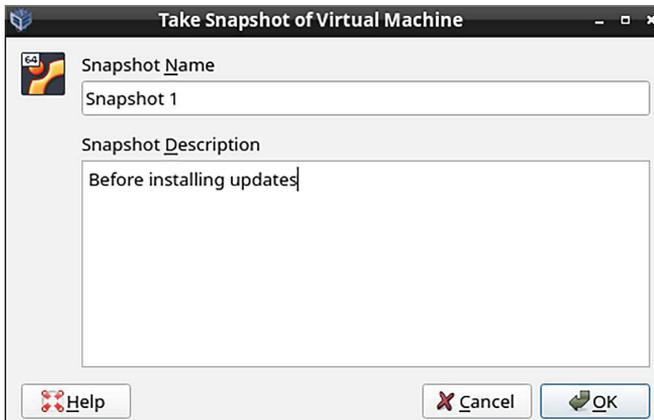


Рис. 1.7. Создание моментального снимка

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.

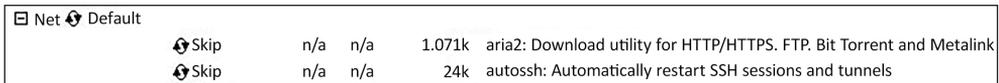


Рис. 1.8. Установка пакетов Cygwin

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.

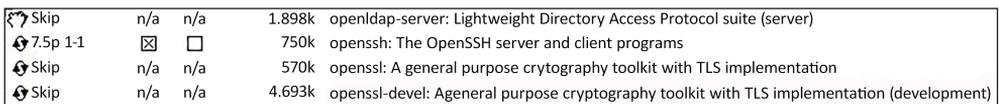
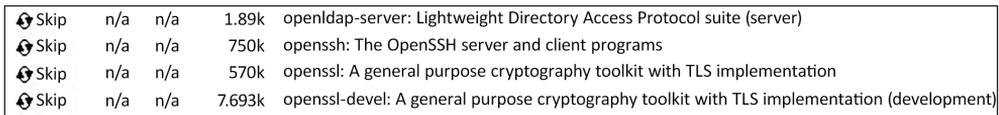


Рис. 1.9. Выбор пакета OpenSSH

6. После выбора нужного пакета экран будет выглядеть так:

⚙ Skip	n/a	n/a	1.898k	openldap-server: Lightweight Directory Access Protocol suite (server)
⚙ Skip	n/a	n/a	750k	openssh: The OpenSSH server and client programs
⚙ Skip	n/a	n/a	570k	openssl: A general purpose cryptography toolkit with TLS implementation
⚙ Skip	n/a	n/a	4,693k	openssl-devel: A general purpose cryptography toolkit with TLS impletation (development)

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**.

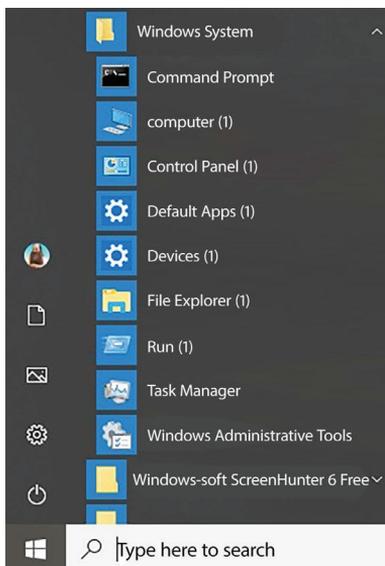


Рис. 1.11. Командная строка в Windows 10

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

```
donnie@orangeplane: ~  
C:\Users\donnie>ssh donnie@192.168.0.57  
donnie@192.168.0.57's password:  
OrangePiOne  
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: 6:04 hours Local users: 2  
Memory usage: 41 % of 460MB Zram usage: 6 % of 230Mb IP: 192.168.0.57  
CPU temp: 61°C  
Usage of /: 9% of 30G  
You have mail.  
Last login: Thu Jul 4 17:49:32 2019 from 192.168.0.9  
donnie@orangeplane:~$
```

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:

  OrangePiOne

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.

Поддержка 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:

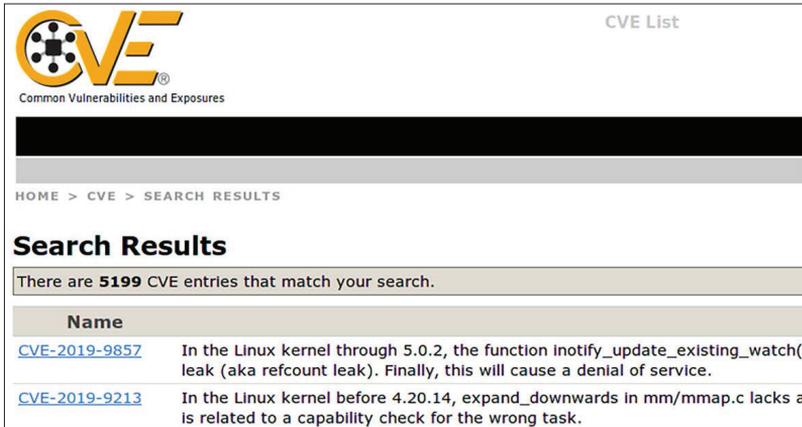


Рис. 1.15. Распространенные уязвимости и риски

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 кривой 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:

```
File Edit Tabs Help
!:Exit -:PrevPg <Space>:NextPg v:View Attachm. d:Del r:Reply j:Next ?:Help
Date: Sun, 7 Jul 2019 16:40:24 -0400 (EDT)
From: Anacron <root@git1.xyzwidgets.com>
To: root@git1.xyzwidgets.com
Subject: Anacron job 'cron.daily' on git1.xyzwidgets.com

/etc/cron.daily/0yum-daily.cron:

The following updates will be downloaded on git1.xyzwidgets.com:
=====
Package                Arch  Version                                Repository
                               Size
=====
Installing:
kernel                  x86_64 3.10.0-957.21.3.el7                    updates  48 M
Updating:
NetworkManager         x86_64 1:1.12.0-10.el7_6                      updates  1.7 M
NetworkManager-libnm  x86_64 1:1.12.0-10.el7_6                      updates  1.4 M
NetworkManager-ppp    x86_64 1:1.12.0-10.el7_6                      updates  165 k
NetworkManager-team   x86_64 1:1.12.0-10.el7_6                      updates  159 k
NetworkManager-tui    x86_64 1:1.12.0-10.el7_6                      updates  239 k
augeas-libs            x86_64 1.4.0-6.el7_6.1                        updates  355 k
bind-libs              x86_64 32:9.9.4-74.el7_6.1                    updates  1.0 M
bind-libs-lite        x86_64 32:9.9.4-74.el7_6.1                    updates  741 k
bind-license          noarch 32:9.9.4-74.el7_6.1                    updates   87 k
- - 1/1: Anacron Anacron job 'cron.daily' on git1.xyzwidg -- (8%)
```

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
<code>sudo needs-restarting</code>	Показывает, какие службы нужно перезагрузить и причины перезагрузки системы
<code>sudo needs-restarting -s</code>	Показывает только службы, которые нужно перезагрузить
<code>sudo needs-restarting -r</code>	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 репозиториях. 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 репозиториях, 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 будем здесь описывать, как настраиваются локальные серверы репозиториях, 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 получают 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>

