Содержание

Об авторе 13

О рецензентах 15

Предисловие 16

Глава 1. Фундаментальные концепции SELINUX 21

1.1. Предоставление большей безопасности в Linux 21
1.1.1. Использование модулей безопасности Linux 24
1.1.2. Расширение возможностей стандартного дискреционного разграничения доступа 25
1.1.3. Ограничение привилегий пользователя root 27
1.1.4. Сокращение воздействия уязвимостей 28
1.1.5. Включение возможностей SELinux в операционной системе 29
1.2. Маркировка всех ресурсов и объектов 31
1.2.1. Описание параметров безопасности 32
1.2.2. Принудительный доступ посредством типов функциональных ограничений 35
1.2.3. Распределение по ролям наборов функциональных ограничений 36
1.2.4. Разделение пользователей по ролям 38
1.2.5. Контроль информационных потоков посредством мандатного механизма 39
1.3. Формирование и распределение политик 40
1.3.1. Создание политик SELinux 41
1.3.2. Распределение политик в виде модулей 43
1.3.3. Комплектация модулей в хранилище политик 45
1.4. Различия между политиками 45
1.4.1. Поддержка многоуровневой защиты (MLS) 46
1.4.2. Манера поведения с неизвестными разрешениями 46
1.4.3. Поддержка неограниченных доменов 47
1.4.4. Ограничение межпользовательского обмена 48
1.4.5. Последовательные изменения версий политик 49
1.4.6. Качественное изменение версий политик 50
1.5. Заключение 51

Глава 2. Режимы работы и регистрация событий 53

2.1. Включение и выключение защиты SELinux 53
2.1.1. Установка глобального состояния защиты 54

2.1.2. Переключение в рекомендательный и принудительный режимы 55
2.1.3. Использование параметров загрузки ядра 57
2.1.4. Отключение защиты SELinux для отдельно взятого сервиса 58
2.1.5. Определение приложений, активно взаимодействующих с SELinux 61
2.2. Регистрация событий и аудит в SELinux 61
2.2.1. Последовательность контроля событий о нарушениях безопасности 62
2.2.2. Исключение конкретных отказов в доступе из числа регистрируемых 64
2.2.3. Конфигурирование подсистемы контроля событий безопасности Linux 65
2.2.4. Настройка локального системного регистратора событий 66
2.2.5. Разбор информации об отказах SELinux 67
2.2.6. Другие типы событий, связанные с SELinux 72
2.2.7. Использование команды ausearch 76
2.3. Получение помощи при отказах 77
2.3.1. Диагностика неисправности с помощью службы setroubleshoot 77
2.3.2. Отправка электронной почты, когда случился отказ SELinux 80
2.3.3. Использование утилиты audit2why 81
2.3.4. Взаимодействие с журналом system 82
2.3.5. Использование здравого смысла 83
2.4. Заключение 85

Глава 3. Управление учетными записями пользователей 86

3.1. Параметры безопасности пользователей 86
3.1.1. Сложность допустимого набора функций 87
3.1.2. Определение неограниченных доменов 89
3.2. Пользователи SELinux и их роли 90
3.2.1. Перечень сопоставлений пользователей с пользовательскими типами SELinux 90
3.2.2. Сопоставление учетных записей с пользовательскими типами 92
3.2.3. Настройка учетных записей относительно служб 93
3.2.4. Создание типов пользователей SELinux 94
3.2.5. Перечень типов допустимого набора функций у ролей 95
3.2.6. Управление категориями 96
3.3. Управление ролями SELinux 98
3.3.1. Настройки присвоения допустимых ролей пользователю 98
3.3.2. Проверка параметров безопасности при помощи утилиты getseuser 100
3.3.3. Подключение ролей с помощью команды newrole 100
3.3.4. Управление доступом к роли с помощью команды sudo 101
3.3.5. Переключение параметров безопасности посредством runcon 102

3.3.6. Переключение на системную роль 102
3.4. SELinux и PAM (подключаемые модули аутентификации) 104
3.4.1. Назначение параметров безопасности с помощью подключаемых модулей аутентификации 104
3.4.2. Запрещение доступа в рекомендательном режиме работы защиты 105
3.4.3. Многоэкземплярность каталогов 106
3.5. Заключение 107

Глава 4. Домены как допустимые наборы функций для процессов и контроль доступа на уровне файлов 109
4.1. О параметрах безопасности файлов 109
4.1.1. Получение информации о параметрах безопасности 110
4.1.2. Интерпретация наименований типов SELinux 111
4.2. Закрепление параметров безопасности за объектом и их игнорирование 112
4.2.1. Наследование параметров безопасности по умолчанию 113
4.2.2. Правила преобразования типов и их вывод 113
4.2.3. Копирование и перемещение файлов 115
4.2.4. Временное изменение параметров безопасности файла 117
4.2.5. Установка категорий для файлов и каталогов 118
4.2.6. Использование многоуровневой защиты для файлов 118
4.2.7. Резервное копирование и восстановление расширенных атрибутов 119
4.2.8. Использование опций монтирования для установки параметров SELinux 119
4.3. Формулировка параметров безопасности для файлов 121
4.3.1. Использование выражений, описывающих параметры безопасности 121
4.3.2. Регистрация изменений параметров безопасности файлов 123
4.3.3. Использование заказных типов 125
4.3.4. Различные виды файлов file_contexts и их компиляция 126
4.3.5. Обмен локальными изменениями 127
4.4. Изменение параметров безопасности у файлов 127
4.4.1. Использование команд setfiles, rlpkg и fixfiles 128
4.4.2. Изменение параметров безопасности на всей файловой системе 128
4.4.3. Автоматическое приведение к заданным значениям изменившихся параметров безопасности 129
4.5. Параметры безопасности процесса 130
4.5.1. Получение параметров безопасности процесса 130
4.5.2. Преобразование типа процесса 131
4.5.3. Проверка соответствия параметров безопасности 133
4.5.4. Другие способы преобразования типов 134

4.5.5. Изначально заданные параметры в структуре идентификатора безопасности ... 134
4.6. Определение границ возможных преобразований ... 135
4.6.1. Очистка переменных окружения во время преобразования к другому типу ... 135
4.6.2. Невыполнение преобразований, когда нет ограничивающего родительского типа ... 137
4.6.3. Использование флага, исключающего новые привилегии у процесса ... 138
4.7. Типы, разрешения и ограничения ... 139
4.7.1. Объяснение атрибутов типа ... 139
4.7.2. Запрос разрешений, предоставленных типу процесса ... 140
4.7.3. Рассмотрение наложенных ограничений ... 142
4.8. Заключение ... 143

Глава 5. Контроль сетевого взаимодействия ... 144

5.1. От контроля межпроцессного взаимодействия (IPC) до сокетов базовых протоколов (TCP/UDP) транспортного уровня ... 144
5.1.1. Использование разделяемой памяти ... 145
5.1.2. Локальное взаимодействие, осуществляемое по каналам ... 147
5.1.3. Обращение через сокеты домена UNIX ... 148
5.1.4. Рассмотрение сокетов netlink ... 149
5.1.5. Действия с сокетами протоколов TCP и UDP ... 150
5.1.6. Вывод списка сетевых соединений с параметрами безопасности ... 152
5.2. Межсетевой экран и маркировка сетевых пакетов ... 152
5.2.1. Вводные сведения о межсетевом экране netfilter ... 153
5.2.2. Реализация маркировки сетевых пакетов и соединений ... 154
5.2.3. Назначение меток пакетам ... 155
5.3. Промаркированные сети ... 157
5.3.1. Резервная маркировка в NetLabel ... 158
5.3.2. Ограничение потоков данных на уровне сетевого интерфейса ... 160
5.3.3. Ограничение потоков данных на уровне элементов сети ... 160
5.3.4. Проверка однорангового потока ... 161
5.3.5. Применение управления в старом стиле ... 162
5.4. Метки безопасности для IPsec ... 163
5.4.1. Установка стандартного IPsec ... 165
5.4.2. Подключение маркировки IPsec ... 166
5.4.3. Использование Libreswan ... 167
5.5. Технология маркировки сетей NetLabel с параметром CIPSO ... 168
5.5.1. Настройка сопоставлений потоков данных с доменами ... 169
5.5.2. Добавление сопоставлений для типов допустимого набора функций ... 170
5.5.3. Локальное использование параметра CIPSO ... 171

5.5.4. Поддержка опции безопасности для IPv6 172
5.6. Заключение 172

Глава 6. Поддержка sVirt и Docker 173

6.1. Виртуализация, защищенная SELinux 173
6.1.1. Представление о виртуализации 173
6.1.2. Обзор рисков виртуализации 175
6.1.3. Использование типов для объектов виртуальной инфраструктуры 176
6.1.4. Перенастраиваемое применение существующих типов виртуализации 177
6.1.5. Рассмотрение защиты различных категорий 179
6.2. Поддержка библиотеки libvirt 180
6.2.1. Различные случаи маркировки ресурсов 181
6.2.2. Оценка архитектуры libvirt 181
6.2.3. Настройка libvirt для работы с sVirt 182
6.2.4. Использование статических параметров безопасности 184
6.2.5. Гибкая настройка параметров безопасности 184
6.2.6. Использование разных мест хранения 185
6.2.7. Интерпретация информации в поле вывода данных о метке 185
6.2.8. Управление доступными категориями 186
6.2.9. Поддержка интерпретирующих доменов 186
6.2.10. Изменение параметров безопасности, установленных по умолчанию 187
6.3. Защищенные контейнеры Docker 188
6.3.1. Представление о защите контейнера 188
6.3.2. Интеграция системы защиты с контейнерами без sVirt 189
6.3.3. Перестраховка безопасности Docker средствами защиты sVirt 190
6.3.4. Ограничение привилегий контейнера 191
6.3.5. Применение различных параметров безопасности для контейнеров 193
6.3.6. Перемаркировка подключенного тома данных 194
6.3.7. Понижение контроля со стороны SELinux для специальных контейнеров 195
6.3.8. Изменение параметров безопасности, установленных по умолчанию 195
6.4. Заключение 196

Глава 7. D-Bus и systemd 197

7.1. Фоновый процесс системы (systemd) 197
7.2. Способ поддержки в systemd служб 198
7.2.1. Введение понятия модульных файлов 198

7.2.2. Установка параметров безопасности SELinux для какой-либо службы ... 199
7.2.3. Использование переходных служб ... 200
7.2.4. Требование включения или отключения SELinux для конкретной службы ... 201
7.2.5. Перемаркировка файлов во время запуска службы ... 202
7.2.6. Использование активизации, основанной на сокетах ... 204
7.2.7. Управление доступом к операциям с модулями ... 205
7.3. Регистрация событий с помощью systemd ... 206
7.3.1. Получение информации, относящейся к SELinux ... 206
7.3.2. Запрос событий, содержащих параметры безопасности SELinux ... 207
7.3.3. Интеграция диагностики неисправностей с журналом ... 207
7.4. Использование контейнеров systemd ... 209
7.4.1. Инициализация контейнеров systemd ... 209
7.4.2. Использование специальных параметров безопасности SELinux ... 209
7.5. Управление файлами устройств ... 210
7.5.1. Использование правил udev ... 210
7.5.2. Назначение метки SELinux на узле устройства ... 212
7.6. Взаимодействие с шиной сообщений D-Bus ... 212
7.6.1. Представление о взаимодействии между процессами D-Bus ... 212
7.6.2. Контроль получения доступа к службам с помощью SELinux ... 215
7.6.3. Управление потоками сообщений ... 216
7.7. Заключение ... 217

Работа с политиками SELinux ... 218

8.1. Логические параметры SELinux ... 218
8.1.1. Вывод списка логических параметров ... 219
8.1.2. Изменение значений логических параметров ... 220
8.1.3. Проверка влияния логического параметра ... 221
8.2. Усиление политик SELinux ... 222
8.2.1. Список модулей политики ... 222
8.2.2. Загрузка и удаление модулей политики ... 223
8.2.3. Создание политик с использованием программы audit2allow ... 224
8.2.4. Использование говорящих за себя наименований для модулей политики ... 226
8.2.5. Использование макрокоманд посреднической политики с программой audit2allow ... 227
8.2.6. Использование скрипта selocal ... 228
8.3. Создание модулей политик по специальным требованиям ... 229
8.3.1. Создание модулей SELinux с помощью исходного языка описания политик ... 230
8.3.2. Создание модулей SELinux с помощью посреднического стиля описания политик ... 231

8.3.3. Создание модулей SELinux с помощью обобщенно-промежуточного языка ... 232
8.3.4. Добавление описаний для параметров безопасности файла ... 232
8.4. Создание ролей и пользовательских типов допустимого набора функций ... 233
8.4.1. Создание файла pgsql_admin.te ... 233
8.4.2. Создание прав пользователя ... 234
8.4.3. Предоставление доступа для взаимодействия с командным интерфейсом ... 235
8.4.4. Формирование структуры файлов пользовательской политики ... 236
8.5. Создание новых типов для приложений ... 237
8.5.1. Создание файлов mojomojo.* ... 238
8.5.2. Создание интерфейсов политик ... 239
8.5.3. Создание структуры файлов политики для приложений ... 240
8.6. Замена существующих политик ... 241
8.6.1. Замена политик Red Hat Enterprise Linux ... 241
8.6.2. Замена политик в Gentoo ... 243
8.7. Другие варианты усиления политики безопасности ... 244
8.7.1. Создание типов SECMARK по специальным требованиям ... 244
8.7.2. Регистрация попыток доступа в журнале событий ... 245
8.7.3. Создание типов, соответствующих специальным требованиям ... 245
8.8. Заключение ... 246

Глава 9. Анализ поведения политики ... 248

9.1. Одноступенчатый анализ ... 248
9.1.1. Использование различных файлов политик SELinux ... 249
9.1.2. Отображение информации об объектах политики ... 249
9.1.3. Применение утилиты sesearch ... 251
9.1.4. Запрос разрешающих правил ... 251
9.1.5. Запрос сведений о правилах преобразования типов ... 251
9.1.6. Запрос правил для других типов ... 252
9.1.7. Запрос правил, связанных с ролями ... 253
9.1.8. Отображение данных с помощью графической программы apol ... 253
9.2. Анализ преобразований типов процессов ... 257
9.2.1. Использование программы apol ... 258
9.2.2. Использование программы sedta ... 259
9.3. Анализ потоков информации ... 261
9.3.1. Использование программы apol для анализа потоков информации ... 262
9.3.2. Использование программы seinfoflow для анализа потоков информации ... 265
9.4. Другие виды анализа политик ... 266
9.4.1. Сравнение политик при помощи sediff ... 266

9.4.2. Анализ политик при помощи sepolicy 267
9.5. Заключение 268

Глава 10. Частные случаи настройки защиты 269
10.1. Усиление защиты веб-серверов 269
10.1.1. Описание условий работы 270
10.1.2. Настройка для установки нескольких экземпляров программ 271
10.1.3. Создание категорий SELinux 272
10.1.4. Выбор необходимых параметров безопасности 273
10.1.5. Включение администраторов в систему защиты 275
10.1.6. Управление работой веб-сервера 275
10.1.7. Работа с обновлением содержания 277
10.1.8. Настройка сети и правил межсетевого экрана 279
10.2. Защита командно-строчного интерфейса 279
10.2.1. Разделение SSH на несколько экземпляров 280
10.2.2. Обновление правил работы сети 281
10.2.3. Изменение корневого каталога для отдельной программы 282
10.2.4. Предоставление параметров безопасности пользователю в зависимости от способа доступа 283
10.2.5. Настройка правил для SSH 285
10.2.6. Включение многопользовательского режима использования 286
10.3. Общий доступ к файлам через сетевую файловую систему NFS 287
10.3.1. Базовая настройка службы NFS 287
10.3.2. Включение поддержки NFS на стороне защищенного клиента 288
10.3.3. Настройка правил безопасности для NFS на сервере 288
10.3.4. Подключение общих сетевых ресурсов с различными параметрами безопасности 289
10.3.5. Работа с промаркированной сетевой файловой системой 290
10.3.6. Сравнение файлового сервера Samba с сетевой файловой системой NFS 291
10.4. Заключение 292

Предметный указатель 293

Об авторе

Свен Вермейлен (Sven Vermeulen) – постоянный участник различных проектов свободного программного обеспечения и автор многочисленных руководств и ресурсов в интернете. Свой первый опыт в разработке свободного программного обеспечения он получил в 1997 году и с тех пор только развивал и совершенствовал свои навыки в этом направлении. В 2003 году он присоединился к проекту Gentoo Linux как разработчик документации и затем выступал в разных ролях, включая такие, как доверенное лицо фонда Gentoo, член совета, руководитель проекта по различным инициативам в области документирования, а также руководитель проектов по усилению системой защиты SELinux операционной системы Gentoo и системному интегрированию.

В течение этого времени Свен получил экспертные знания как на уровне операционной системы, так и на уровне серверного прикладного программного обеспечения. Он использовал свой интерес к безопасности, чтобы направлять свои проекты, связанные с формированием руководств, в область защиты информации. Для этой цели им стали применяться:

- языки описания политики безопасности, механизмов контроля и результаты оценки SCAP (*Security Content Automation Protocol* – Протокол автоматизации информационного обеспечения безопасности);
- средства контроля разграничения доступа, реализованные в SELinux;
- аутентифиакция при помощи средства обеспечения защиты PAM (*Pluggable Authentication Modules* – подключаемые модули аутентификации);
- программные межсетевые экраны
- и многое другое.

Для SELinux Свен внес несколько вариантов политик в проект Посреднической политики (Reference Policy project), и он является активным участником проектов по разработке политик и пользовательского пространства.

В своей ежедневной работе Свен – архитектор информационных технологий в одном из европейских финансовых институтов, а также самостоятельно действующий инженер и консультант. Создание безопасных инфраструктур (и сопутствующая архитектурная интеграция) является, конечно, важной частью его работы. Свое образование – степень магистра компьютерной инженерии – Свен Вермейлен получил в Бельгии, в университете города Гент. Вторая степень была получена в магистратуре организации INNOCOM (https://www.inno.com) по специальности информационно-коммуникационной архитектуры предприятия. Работал инженером инфраструктуры веб-приложений.

Свен является основным автором книги *Gentoo Handbook* (Справочник по Gentoo), которая охватывает вопросы установки и настройки операционной системы Gentoo на нескольких архитектурах. Он также автор публикации в интернете *Linux Sea* (Море Linux) – http://swift.siphos.be/linux_sea, которая является

базовым введением в операционную систему Linux для начинающих системных администраторов. Кроме того, он является автором таких книг издательства Packt Publishing, как *SELinux System Administration* (Администрирование системы защиты SELinux, 1-е изд.) и *SELinux Cookbook* (Книга готовых рецептов для системы защиты SELinux).

Я хотел бы поблагодарить сообщество разработчиков ПО с открытым исходным кодом и свободного программного обеспечения за его бесконечное стремление создавать отличные программное обеспечение, документацию, настоящие произведения искусства и сервисы. Именно благодаря этому стремлению компании и организации во всем мире пользуются предоставляемыми средствами высокого качества со всей свободой, которую дает это программное обеспечение. В частности, я хотел бы поблагодарить сообщество Gentoo, поскольку оно предоставляет отличные метадистрибутив и операционную систему. Люди, которых я там встречаю, – все они очень высоко мотивированные, опытные и/или эксперты в определенных областях. Присутствие в сообществе заставляет меня стремиться узнать больше.

О рецензентах

Дэвид Куигли (David Quigley) начал свою карьеру исследователем компьютерных систем в Национальной исследовательской лаборатории по обеспечению информационной безопасности при Агентстве национальной безопасности США, где он работал в качестве члена команды SELinux. Дэвид возглавил работы по проектированию и реализации маркировки сетевой файловой системы NFS в SELinux. До этого участвовал в сообществе открытого программного обеспечения, поддерживая кодовую базу проекта вспомогательной файловой системы Unionfs 1.0, и вносил свой вклад в различные другие проекты. Дэвид выступал с докладами на таких конференциях, как Оттавский симпозиум по Linux, семинар по StorageSS, LinuxCon, а также на нескольких локальных собраниях групп пользователей Linux, где темы презентаций включали хранение, файловые системы и безопасность. В настоящее время Дэвид работает инженером по ядру файловой системы ZFS в отделе высокопроизводительных данных в Intel. И ранее рецензировал книгу *SELinux Cookbook*, опубликованную издательством Packt.

Я хотел бы поблагодарить мою замечательную жену Кэти за все, что она делает, чтобы у меня было время заняться такими вещами, как обзор этой книги, и поездками, связанными с презентациями о SELinux. Она – радость моей жизни и помогла мне стать тем, кем я являюсь сегодня. Я также хотел бы поблагодарить моих детей Зои Джейн и Кэролайн, которые напоминают нам о том, что нужно любить и ценить время, которое мы проводим вместе с семьей.

Сэм Уилсон (Sam Wilson) – старший инженер по системам и безопасности, недавно увлекся конструированием радиотехнического оборудования и специализируется на Red Hat Enterprise Linux. Благодаря обширным знаниям в области безопасности, охватывающим микросервисы, инфраструктуру, и в организации работы команды при обеспечении коллективных целей по обеспечению безопасности (*SecOps*) к Сэму регулярно обращаются за наставничеством и советами по SELinux организации, с которыми он сотрудничает и с которыми работает. Сэм активно участвует в сообществах GNU/Linux с начала 2007 года и добровольно посвятил себя работе над проектами NTFreenet, Darwin Community Arts, Ansible и Fedora.

Сэм является автором сайта https://www.cycloptivity.net, а также работает с командой интеллектуальной безопасности Atlassian Security Intelligence над визуализацией, эксплуатационной безопасностью и средствами управления для поддержки и защиты клиентов Atlassian в облаке.

Предисловие

Безопасное состояние операционной системы или какой-либо службы является результатом многоуровневого подхода к обеспечению безопасности. Системы могут быть защищены от внешнего мира при помощи межсетевых экранов, операционные системы должны регулярно получать обновления безопасности, работающие службы должны быть правильно настроены, необходимо разделять обязанности для конечных пользователей и т. д.

Контроль доступа – это еще один уровень обеспечения защиты, который администраторы должны учитывать. Благодаря системе защиты SELinux (Security Enhanced Linux – повышенная безопасность Linux) в экосистеме Linux появилась надежная и, что удобно, встраиваемая система принудительного контроля доступа. Некоторые дистрибутивы включают SELinux по умолчанию, другие позволяют администраторам включать ее самим. Android, одна из самых популярных операционных систем для мобильных устройств, также использует технологию SELinux под названием SEAndroid.

Но, в отличие от Android, где пользователи и приложения находятся под жестким контролем и где недопустимы отклонения в настройке и организации файлов и ресурсов, настольные компьютеры, рабочие станции и серверы, которые реализуют Linux, имеют большее разнообразие в возможностях управления защитой. В результате настройка SELinux в этих системах требует больше знаний о том, что такое SELinux, как он работает и как его можно использовать.

В этой книге мы обсуждаем, что такое SELinux и как он встроен в операционную систему Linux. Мы рассмотрим различные аспекты конфигурации SELinux и разберем несколько вариантов применения, которые используют сильные стороны SELinux для дальнейшего усиления безопасности системы и служб, размещенных на ней.

О чем эта книга

Глава 1 «*Фундаментальные концепции SELinux*» дает администраторам представление о том, что такое система защиты SELinux и как она взаимодействует на уровне ядра операционной системы Linux. Здесь объясняются различия в реализациях SELinux между дистрибутивами и описывается характерная для SELinux терминология, которая дальше будет часто использоваться по мере углубления в технологию SELinux.

Глава 2 «*Режимы работы и регистрация событий*» описывает различные состояния работы SELinux и показывает, где SELinux регистрирует свои события. Эта глава поможет администраторам разобраться с тем, как следует интерпретировать и анализировать эти события.

Глава 3 «*Управление учетными записями пользователей*» рассказывает администраторам, как управлять пользователями Linux и их правами, а также выполнять сопоставление этих пользователей с различными ролями, которые SELinux поддерживает с помощью собственной организации пользовательского пространства и подключаемых модулей аутентификации Linux. Кроме того, глава охватывает такую сущность, как категории защищаемой информации, реализованные в SELinux.

Глава 4 «*Домены как допустимые наборы функций для процессов и контроль доступа на уровне файлов*» знакомит администраторов с метками параметров безопасности SELinux. С тем, как эти метки хранятся в файловой системе или предоставляются другим ресурсам. В этой главе администраторы и конечные пользователи узнают, как устанавливать и обновлять метки параметров безопасности.

Глава 5 «*Контроль сетевого взаимодействия*» рассматривает стандартные службы сетевой безопасности, утилиту iptables и протокол IPSec с точки зрения их совместной работы с функциями защиты SELinux. Администраторы смогут научиться включать поддержку SELinux в этих службах безопасности и даже включать маркировку, выполняемую между распределенными по сети системами, при помощи таких методов, как Labeled IPSec и NetLabel/CIPSO.

Глава 6 «*Поддержка sVirt и Docker*» рассказывает, как компания Red Hat разработала технологию защищенной виртуализации (sVirt) и реализовала ее в двух системах виртуализации: библиотеке libvirt и контейнерах Docker. В этой главе объясняется, как настроить эти службы с помощью поддержки SELinux и контролировать перемещение ресурсов, используемых гостевыми системами виртуальной инфраструктуры или контейнерами.

Глава 7 «*D-Bus и systemd*» рассказывает о сферах влияния упомянутых системных служб уровня ядра и о том, как они используют правила SELinux для дальнейшего усиления безопасности своих собственных функциональных возможностей. Получив эти знания, администраторы смогут настроить защищенную работу службы межпроцессного взаимодействия D-Bus, а также управлять средствами доступа SELinux, применяемыми через подсистему инициализации systemd.

Глава 8 «*Работа с политиками SELinux*» посвящена настройке и управлению политиками SELinux. Она показывает, как можно создавать политики безопасности по заданным требованиям или даже заменять политику, официально предоставляемую в рамках дистрибутива.

Глава 9 «*Анализ поведения политики*» углубляется в инструменты анализа, которые позволяют инженерам и администраторам более детально рассматривать политику безопасности SELinux. С помощью этих инструментов можно получить наиболее полное представление о том, что политика в себя включает и как поведет себя в различных ситуациях.

Глава 10 «*Частные случаи настройки защиты*» описывает ряд распространенных случаев использования серверов, таких как веб-серверы и файловые серверы, и способы использования SELinux для их защиты. В этой главе рас-

сказывается, как можно изолировать пользовательское окружение с помощью SELinux и как администраторы могут построить защищенную многопользовательскую систему.

Что необходимо для понимания книги

Поскольку SELinux является компонентом для операционной системы Linux, то читателям желательно иметь ее в своем распоряжении вместе с установленной системой защиты SELinux. Процесс установки SELinux не входит в материал данной книги, по этому вопросу стоит обратиться к документации используемого дистрибутива. Кроме того, настройка системы защиты требует наличия привилегий администратора в системе.

Для кого предназначена эта книга

Эта книга предназначена для системных администраторов Linux, которые имеют достаточный опыт обслуживания систем Linux, хотят разбираться в технологии защиты SELinux и работать с ней. Кроме того, данная книга может быть полезна для архитекторов информационных систем, чтобы понять, как можно применять SELinux для повышения безопасности систем Linux и служб, работающих под управлением этой операционной системы для обеспечения нужд компании.

Соглашения

В этой книге вы найдете несколько стилей оформления текста, которые позволяют отличать одни виды информации от других. Вот примеры этих стилей и объяснение их значения.

Ключевые слова в тексте, имена таблиц базы данных, каталогов, файлов, расширения файлов, имена путей, адреса сайтов и данные, вводимые пользователем, выделены моноширинным шрифтом: «Мы выполняем это с помощью команды `semanage login`».

Блок текста, связанного с настройками, выводом команд и программами, выглядит в тексте книги так:

```
dbadm_r
  Dominated roles:
    dbadm_r
  Types:
    qmail_inject_t
    dbadm_t
    ...
    user_mail_t
```

Любая фраза командно-строчного интерфейса дополнительно выделяется жирным шрифтом:

```
# seinfo -amcs_constrained_type -x | grep virt_
```

Новые термины и **важные слова** выделены жирным шрифтом в тексте описания. Слова, которые пользователь может увидеть на экране (например, в меню или диалоговых окнах), отображаются в тексте книги примерно так: «После загрузки выберите пункт меню **Новый анализ** (New Analysis), для того чтобы начать работу с функциями анализа политики».

Предупреждения и важные замечания отмечаются таким значком.

Советы и подсказки выглядят так.

Тематические пояснения для русскоязычного издания.

Отзывы и пожелания

Мы всегда рады отзывам наших читателей. Расскажите нам, что вы думаете об этой книге – что понравилось или, может быть, не понравилось. Отзывы важны для нас, чтобы выпускать книги, которые будут для вас максимально полезны.

Вы можете написать отзыв прямо на нашем сайте www.dmkpress.com, зайдя на страницу книги, и оставить комментарий в разделе «Отзывы и рецензии». Также можно послать письмо главному редактору по адресу dmkpress@gmail.com, при этом напишите название книги в теме письма.

Если есть тема, в которой вы квалифицированы, и вы заинтересованы в написании новой книги, заполните форму на нашем сайте по адресу http://dmkpress.com/authors/publish_book/ или напишите в издательство по адресу dmkpress@gmail.com.

Скачивание исходного кода примеров

Скачать файлы с дополнительной информацией для книг издательства «ДМК Пресс» можно на сайте www.dmkpress.com на странице с описанием соответствующей книги.

Список опечаток

Хотя мы приняли все возможные меры для того, чтобы удостовериться в качестве наших текстов, ошибки все равно случаются. Если вы найдете ошибку в одной из наших книг – возможно, ошибку в тексте или в коде, – мы будем очень благодарны, если вы сообщите нам о ней. Сделав это, вы избавите других читателей от расстройств и поможете нам улучшить последующие версии данной книги.

Если вы найдете какие-либо ошибки в коде, пожалуйста, сообщите о них главному редактору по адресу dmkpress@gmail.com, и мы исправим это в следующих тиражах.

Нарушение авторских прав

Пиратство в интернете по-прежнему остается насущной проблемой. Издательства «ДМК Пресс» и Apress очень серьезно относятся к вопросам защиты авторских прав и лицензирования. Если вы столкнетесь в интернете с незаконно выполненной копией любой нашей книги, пожалуйста, сообщите нам адрес копии или веб-сайта, чтобы мы могли применить санкции.

Пожалуйста, свяжитесь с нами по адресу dmkpress@gmail.com со ссылкой на подозрительные материалы.

Мы высоко ценим любую помощь по защите наших авторов, помогающую нам предоставлять вам качественные материалы.

Глава 1

Фундаментальные концепции SELINUX

SELinux – система защиты, обеспечивающая повышенную безопасность в Linux. Она включает дополнительные меры защиты в операционную систему для большей целостности, доступности и конфиденциальности ее ресурсов.

В данном разделе будут рассмотрены следующие вопросы:

1) почему SELinux использует метки для идентификации ресурсов;
2) как качественно SELinux отличается от штатных систем контроля доступа Linux за счет наличия правил обеспечения безопасности;
3) каким образом правила контроля доступа повышают уровень защиты, распространяясь через файлы политик безопасности SELinux.

В конце главы будут рассмотрены различия между версиями SELinux, реализованными в дистрибутивах Linux.

1.1. Предоставление большей безопасности в Linux

Опытные администраторы систем, построенных на базе Linux, и проектировщики систем защиты знают, что им необходимо доверять пользователям и процессам в обслуживаемой системе. То есть администраторам требуется заручиться некоторой лояльностью со стороны пользователей и гарантиями для выполняемых процессов в системе, чтобы обеспечивался нужный уровень защищенности. Это частично связано с тем, что пользователи могут пытаться использовать уязвимости, найденные у запущенных программ, с которыми они должны работать «по долгу службы», но еще больший их вклад в снижение уровня угроз связан как раз с тем, что безопасное состояние системы зависит от непосредственного поведения пользователей. Дело в том, что пользователь с доступом к конфиденциальной информации (к такой, которая не должна быть предоставлена в общий доступ), сам же и управляет приложениями, в т. ч. запускает и совершает другие действия, которые оказывают влияние на защиту системы. Одним из механизмов стандартной защиты является дискреционный контроль доступа, который настраивается в соответствии с потребностями пользователя.

Механизм ***дискреционного контроля доступа*** DAC (*Discretionary Access Control*) в операционной системе Linux основывается на сопоставлении данных процесса, выполняемого пользователем (и/или группой пользователей), с информацией о разрешениях для пользователя (и/или группы), характерных для какого-либо файла, директории или другого управляемого ресурса. Рассмотрим файл `/etc/shadow`, который содержит информацию о паролях и именах пользователей некой локальной системы:

```
$ ls -l /etc/shadow
rw------- 1 root root 1010 Apr 25 22:05 /etc/shadow
```

Без дополнительных механизмов контроля доступа, имеющихся в наличии, этот файл является разрешенным для чтения и записи любым процессам, которыми владеет пользователь `root`, вне зависимости от целей процесса, преследуемых в системе. Файл `shadow` представляет типичный пример такого конфиденциального файла, который никто из пользователей не хотел бы видеть опубликованным или использованным недобросовестным образом. По сути, кто-то один имеет доступ к данному файлу и может скопировать его в какое-либо другое место, например в домашний каталог, по почте отправить на другой компьютер или попытаться выполнить атаку на защищенное криптографией значение пароля (*hash*) пароля, размещенное в этом файле.

Другой пример того, что стандартный механизм дискреционного контроля доступа требует доверия к пользователям, – это случай, когда в системе расположена база данных. Файлы базы данных сами по себе являются (очень хочется верить) доступными только для сеансовых пользователей, зарегистрированных в *системе управления базой данных* (Database management system – DBMS), и для пользователя операционной системы `root`.

Должным образом защищенные системы будут предоставлять доступ к этим файлам только хорошо проверенным пользователям (например, через команду `sudo`), позволяя им заменить свои эффективные пользовательские идентификаторы на идентификаторы пользователей базы данных или даже на пользователя с правами `root`, и это для строго заданного набора команд. Эти пользователи также могут анализировать файлы базы данных и получать доступ к потенциальной конфиденциальной информации в базе данных без входа через СУБД.

Однако пользователи операционной системы не являются единственной угрозой, из-за которой необходимо защитить систему. Многие фоновые программы (*software daemons*) запускаются с правами пользователя `root` или имеют необходимые для работы привилегии в системе. Ошибки внутри этих программ могут легко привести к утечке информации или даже к удаленному использованию уязвимостей. Программы резервирования, контроля выполнения, администрирования, планирования и им подобные программы: они все часто запускаются с высшими пользовательскими привилегиями, доступными в операционной системе Linux. Даже когда администратор не предоставляет привилегии пользователям, их взаимодействие со службами подразумевает потенциальный риск безопасности. А раз так, то пользователи продолжают

пользоваться своими расширенными правами для корректного взаимодействия с приложениями и обеспечения корректного функционирования системы в целом. Поэтому администратор вынужден строить безопасность системы на порядочности ее пользователей.

Рассмотрим теперь SELinux, который содержит дополнительный уровень контроля доступа, стоящий выше дискреционного механизма. SELinux предоставляет *мандатный контроль доступа* (Mandatory access control – MAC) системы, который нивелирует рассмотренные недостатки дискреционного контроля, предоставляя администратору полный контроль над тем, что является позволенным в системе, а что нет. Он достигает этого путем применения метода управления политиками, определяющими, являются или не являются процессы разрешенными к выполнению, а также путем приведения этих политик в жизнь через ядро операционной системы Linux.

Мандатные средства, которые контролируют доступ, приводятся в исполнение операционной системой и определяются исключительно правилами политики, задействованными системным администратором (или администратором безопасности). Пользователи и процессы не имеют прав на изменение правил безопасности, таким образом они не могут что-либо делать в части контроля доступа; защита больше не находится во власти свободы их действий.

Здесь слово «мандатный» так же, как и ранее слово «дискреционный», было выбрано для описания возможностей системы контроля доступа не случайно: оба термина являются известными в области информационной безопасности и имеют описания в других публикациях, включая стандарт «Критерии оценки доверенных компьютерных систем»[1] (известный также как «Оранжевая книга»), выпущенный Министерством обороны Соединенных Штатов Америки в 1985 году. Этот документ предшествовал стандарту «Общие критерии»[2], предназначенному для сертификации компьютерных систем.

Р В России данные термины по защите информации определяются и раскрываются в нижеприведенных и других документах:

1) Национальный стандарт Российской Федерации. Информационная технология. Методы и средства обеспечения безопасности. Критерии оценки безопасности информационных технологий;
2) руководящий документ ФСТЭК[3] «Защита от несанкционированного доступа к информации. Термины и определения». Утверждено решением председателя Гостехкомиссии России от 30 марта 1992 г.;
3) руководящий документ ФСТЭК «Средства вычислительной техники. Защита от несанкционированного доступа к информации. Показатели защищенности от несанкционированного доступа к информации».

---

[1] Trusted Computer System Evaluation Criteria (TCSEC) 1985 [Электронный ресурс] // http://csrc.nist.gov/publications/history/dod85.pdf.

[2] Common Criteria (ISO/IEC 15408) [Электронный ресурс] // http://www.commoncriteriaportal.org/cc/.

[3] ФСТЭК – Федеральная служба технического и экспортного контроля.