Отчёт по лабораторной работе

Лабораторная работа № 6

Живцова Анна

Содержание

Список иллюстраций

Список таблиц

1 Цель работы

Развить навыки администрирования ОС Linux. Получить первое практическое знакомство с технологией SELinux. Проверить работу SELinx на практике совместно с веб-сервером Apache.

2 Теоретическое введение

Security-Enhanced Linux (SELinux) - это метод контроля доступа в Linux на основе модуля ядра Linux Security (LSM). SELinux включен по умолчанию во многих дистрибутивах на основе Red Hat, использующих пакетную базу rpm, например, Fedora, CentOS и т д.

SELinux представляет собой систему маркировки, каждый процесс имеет метку. Каждый файл, каталог или даже пользователь в системе имеет метку. Даже портам и устройствам и именам хостов в системе присвоены метки. SELinux определяет правила доступа процесса к объектам с определенными метками. Это и называется политикой. За соблюдением правил следит ядро. Иногда это еще называется обязательный контроль доступа (Mandatory Access Control, MAC)

Владелец файла не имеет полной свободы действий над атрибутами безопасности. Стандартные атрибуты контроля доступа, такие как группа и владелец ничего не значат для SELinux. Полностью все управляется метками. Значения атрибутов могут быть установлены и без прав root, но на это нужно иметь специальные полномочия SELinux [1] .

Apache - это популярнейший свободный веб-сервер. Состоянием на 2020 год он используется на 33% всех сайтов интернета, а это приблизительно 304 миллиарда сайтов. Этот веб-сервер был разработан в далеком 1995, как замена для популярного того сервера NCSA и исправил множество его проблем. Ходят слухи что его имя походит от a patchy, заплатка, так как он исправлял ошибки NCSA. Сейчас же, это кроссплатформенная программа, поддерживающая Windows, Linux и MacOS и обеспечивающая достаточную гибкость, настраиваемость и функцио-

нальность. Программа имеет модульную структуру, что позволяет расширять ее функциональность почти до бесконечности с помощью модулей [2] .

3 Выполнение лабораторной работы

1. Убедились, что SELinux работает в режиме enforcing при политике targeted. Установили и запустили web-сервер apache (см. рис. 3.1).

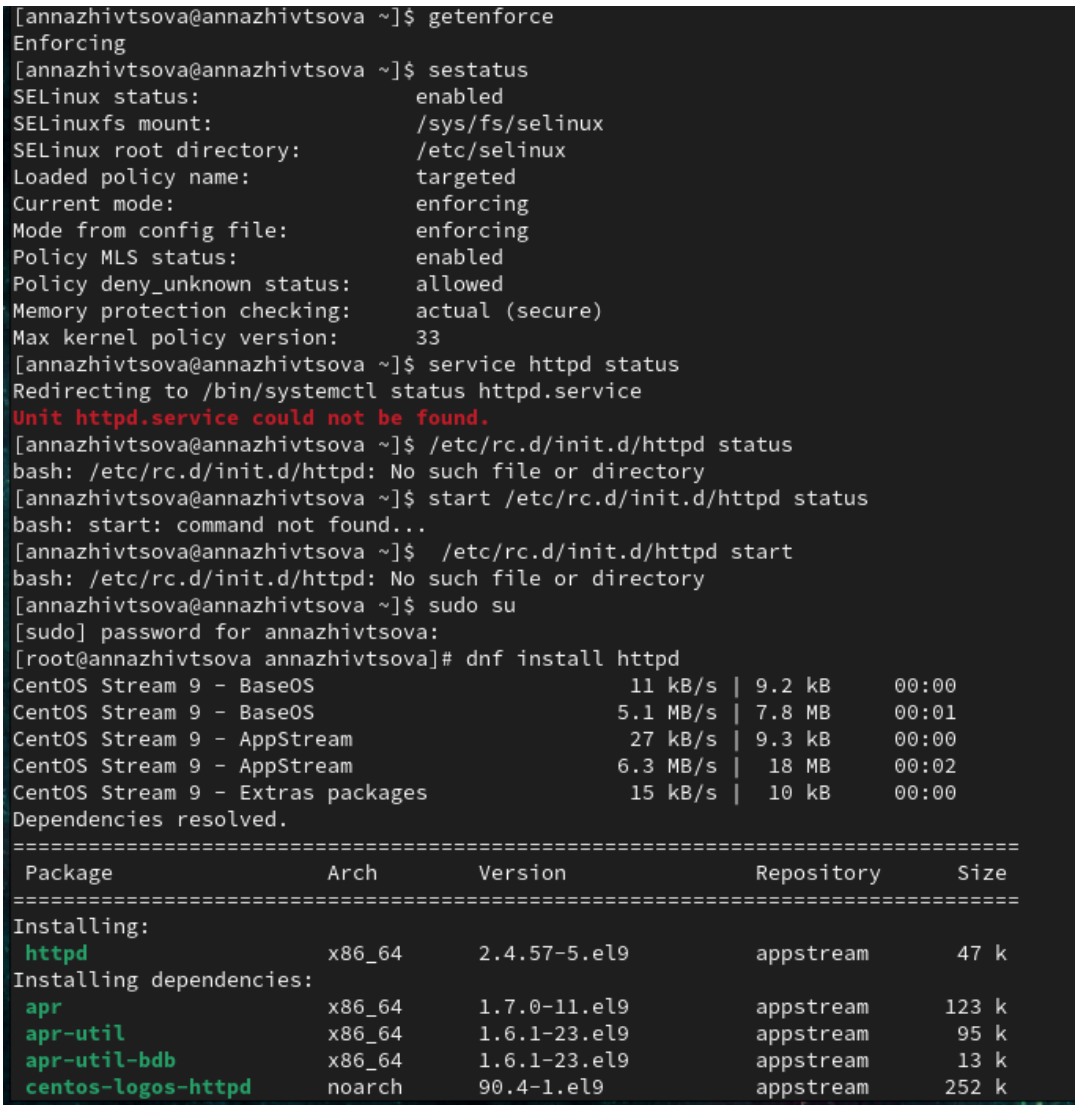

Рис. 3.1: Проверка готовности к выполнению лабораторной работы

2. Получили контекст процесса Apache. Посмотрели текущее состояние переключателей SELinux для Apache (см. рис. 3.2).

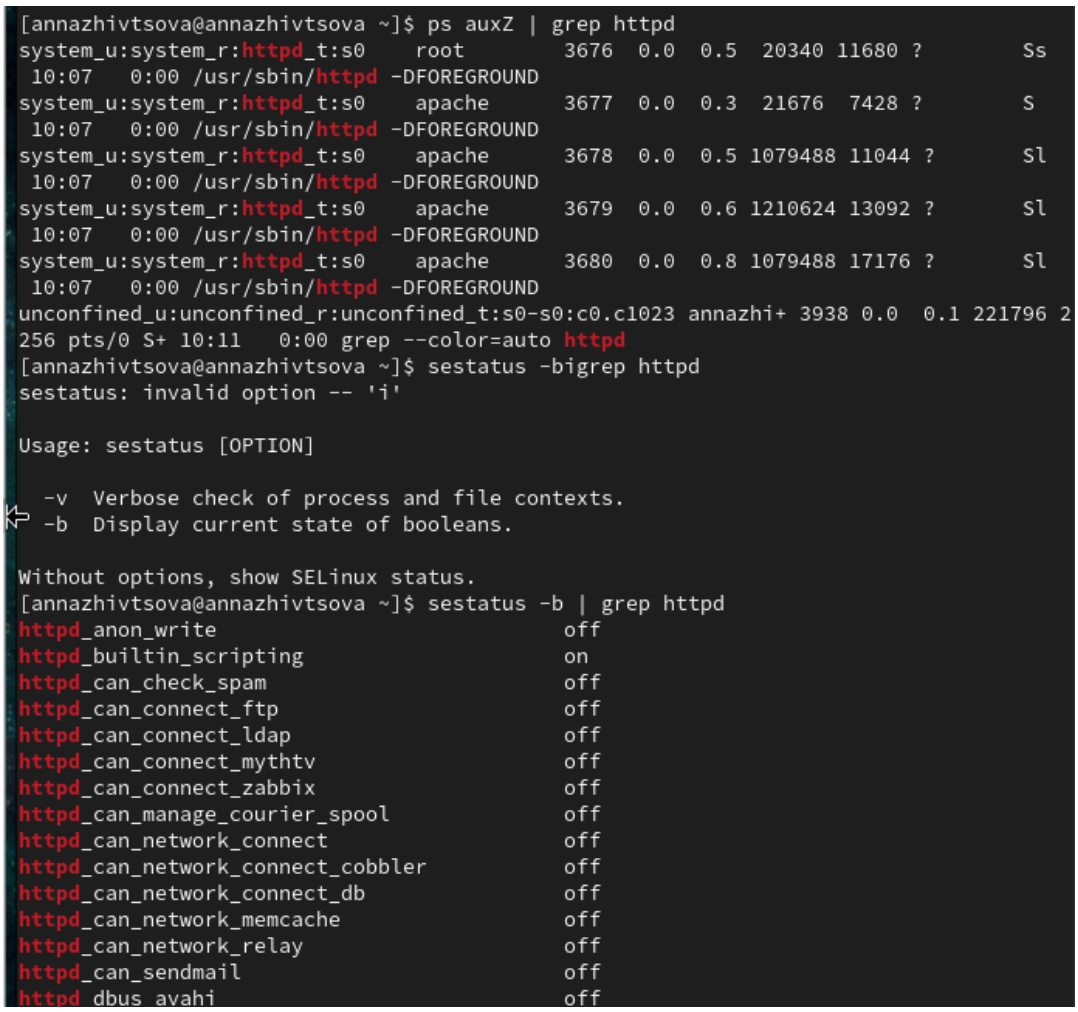

Рис. 3.2: Контекст процесса Apache. Состояние переключателей SELinux для Apache

3. Посмотрели статистику по политике с помощью команды seinfo (см. рис. 3.3), определили множество пользователей, ролей, типов (см. рис. 3.4).

```
seinfo: error: unrecognized arguments: --types
[annazhivtsova@annazhivtsova ~]$ seinfo
Statistics for policy file: /sys/fs/selinux/policy
Policy Version:             33 (MLS enabled)
Target Policy:              selinux
Handle unknown classes:     allow
  Classes:             135    Permissions:         457
  Sensitivities:         1    Categories:         1024
  Types:              5135    Attributes:          259
  Users:                 8    Roles:                15
  Booleans:            357    Cond. Expr.:         390
  Allow:             65380    Neverallow:            0
  Auditallow:          172    Dontaudit:          8647
  Type_trans:       267809    Type_change:          94
  Type_member:          37    Range_trans:        6164
  Role allow:           39    Role_trans:          419
  Constraints:          70    Validatetrans:         0
  MLS Constrain:        72    MLS Val. Tran:         0
  Permissives:           2    Polcap:                6
  Defaults:              7    Typebounds:            0
  Allowxperm:            0    Neverallowxperm:       0
  Auditallowxperm:       0    Dontauditxperm:        0
  Ibendportcon:          0    Ibpkeycon:             0
  Initial SIDs:         27    Fs_use:               35
  Genfscon:            109    Portcon:             665
  Netifcon:              0    Nodecon:               0
[annazhivtsova@annazhivtsova ~]$ seinfo -r

Roles: 15
   auditadm_r
   container_user_r
   dbadm_r
   guest_r
   logadm_r
   nx_server_r
   object_r
   secadm_r
   staff_r
   sysadm_r
   system_r
   unconfined_r
   user_r
```

Рис. 3.3: Статистика по политике

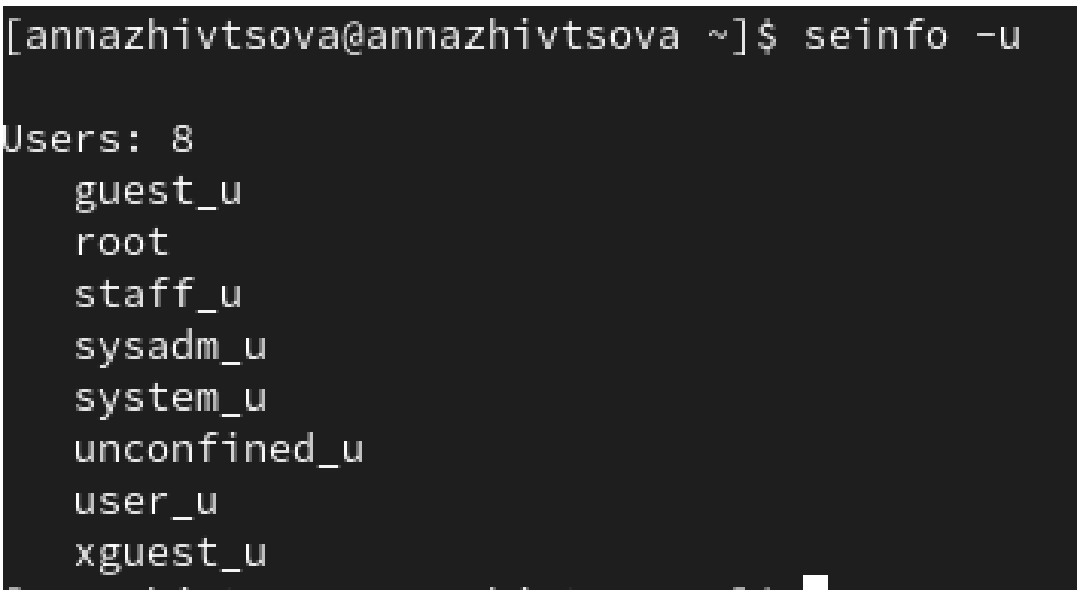

Рис. 3.4: Множество пользователей

4. Определили тип файлов и поддиректорий, находящихся в директориях `/var/www` и `/var/www/html`. Для последней директории установлены права rwxr_xr_x. Создали в ней простой html файл. По умолчанию ему был присвоен контекст unconfined_u:object_r:httpd_sys_content_t:s0. Обратились к фйлу через браузер (см. рис. 3.5).

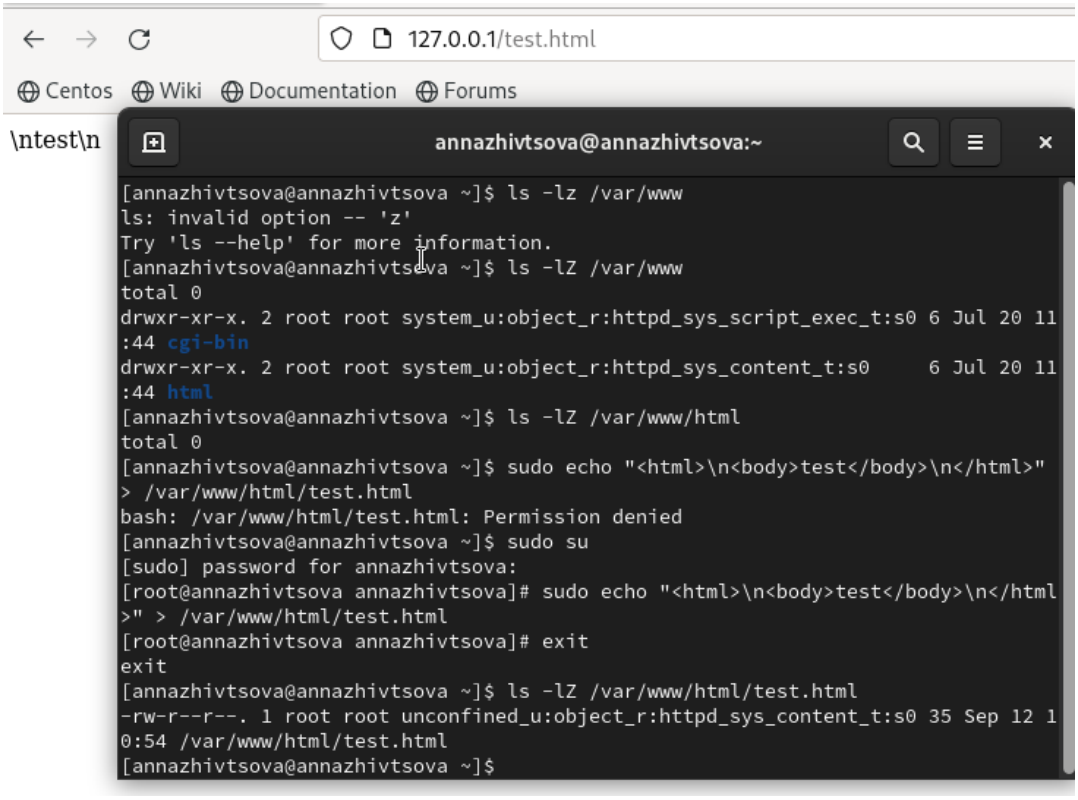

Рис. 3.5: Изучение контекстов и отображение созданного html файла

5. Изменили контекст созданного html файла на тот, к которому процесс httpd не должен иметь доступа и получили ошибку при обращении к файлу через браузер (см. рис. 3.6).

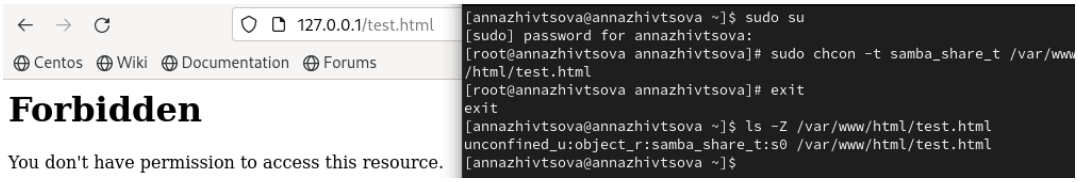

Рис. 3.6: Ошибка приобращении к html файлу с измененным контекстом

6. Посмотрели лог файлы (см. рис. 3.7).

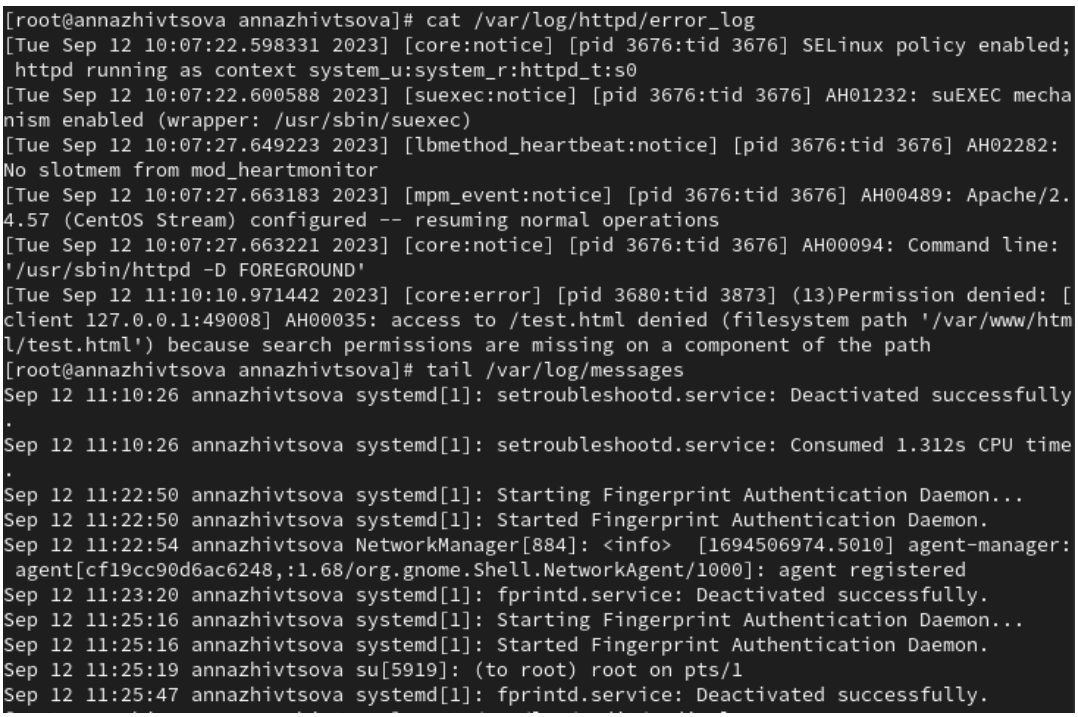

Рис. 3.7: Лог файлы

7. Запустили веб-сервер Apache на прослушивание TCP-порта (см. рис. 3.8).

Рис. 3.8: Смена порта

8. Посмотрели лог файлы, недавно был обновлен файл `/var/log/http/error_log` (см. рис. 3.9).

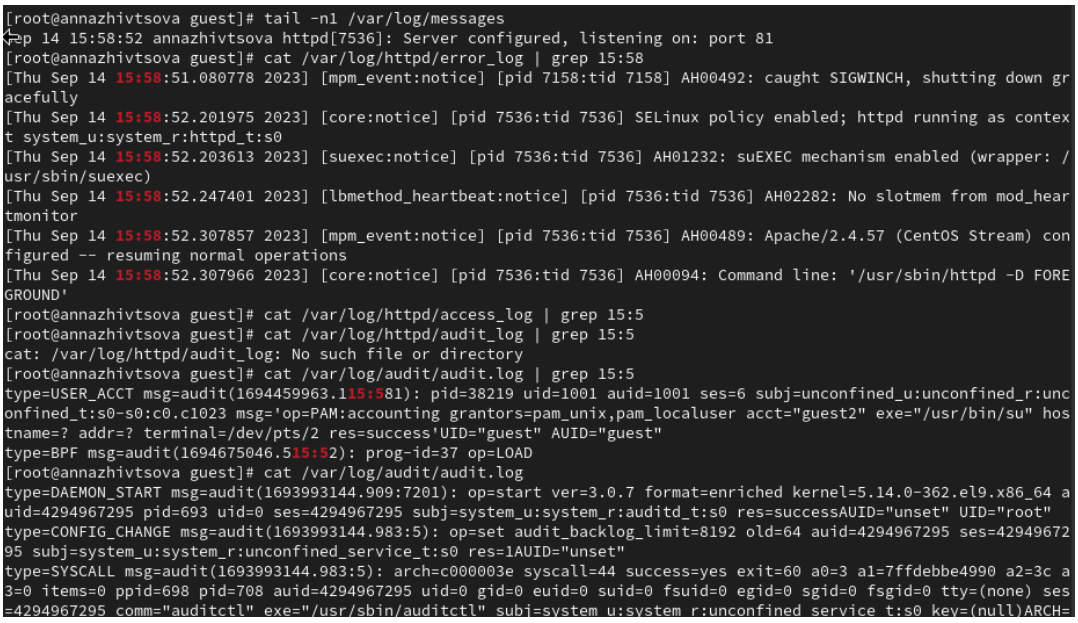

Рис. 3.9: Смена порта

9. Выполнили поднастройку политики командой `semanage port -a -t http_port_t -p tcp 81`, вернули html файлу прежний контекст, обратились к нему из браузера (см. рис. 3.10).

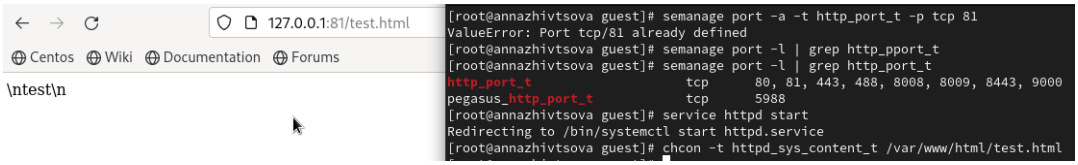

Рис. 3.10: Возвращение контекста html файлу и обращение через браузер

10. Восстановление всех изменений (см. рис. 3.11).

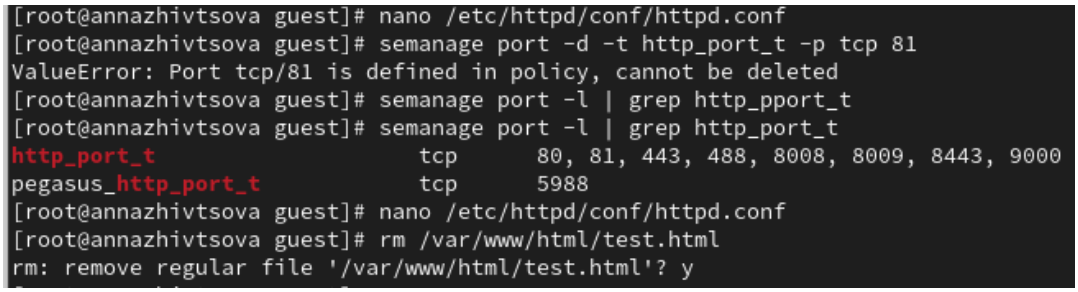

Рис. 3.11: Восстановление всех изменений

4 Выводы

Развили навыки администрирования ОС Linux. Получили первое практическое знакомство с технологией SELinux. Проверили работу SELinx на практике совместно с веб-сервером Apache.

Список литературы

1. Vermeulen S. SELinux System Administration. Packt Publishing Ltd., 2001. 336 с.
2. Хокинс, Скотт. Администрирование web-сервера apache и руководство по электронной коммерции. Издательский дом Вильям, 2020. 459 с.