Лаораторная работа №6

Мандатное разграничение прав в Linux

Латыпова Диана. НФИбд-02-21

Содержание

# 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 Задание

1. Войдите в систему с полученными учётными данными и убедитесь, что SELinux работает в режиме enforcing политики targeted с помощью команд getenforce и sestatus.
2. Обратитесь с помощью браузера к веб-серверу, запущенному на вашем компьютере, и убедитесь, что последний работает: service httpd status или /etc/rc.d/init.d/httpd status Если не работает, запустите его так же, но с параметром start.
3. Найдите веб-сервер Apache в списке процессов, определите его контекст безопасности и занесите эту информацию в отчёт. Например, можно использовать команду ps auxZ | grep httpd или ps -eZ | grep httpd
4. Посмотрите текущее состояние переключателей SELinux для Apache с помощью команды

sestatus -bigrep httpd

Обратите внимание, что многие из них находятся в положении «off». 5. Посмотрите статистику по политике с помощью команды seinfo, также определите множество пользователей, ролей, типов. 6. Определите тип файлов и поддиректорий, находящихся в директории /var/www, с помощью команды

ls -lZ /var/www

1. Определите тип файлов, находящихся в директории /var/www/html:

ls -lZ /var/www/html

1. Определите круг пользователей, которым разрешено создание файлов в директории /var/www/html.
2. Создайте от имени суперпользователя (так как в дистрибутиве после установки только ему разрешена запись в директорию) html-файл /var/www/html/test.html следующего содержания:

<html>  
<body>test</body>  
</html>

1. Проверьте контекст созданного вами файла. Занесите в отчёт контекст, присваиваемый по умолчанию вновь созданным файлам в директории /var/www/html.
2. Обратитесь к файлу через веб-сервер, введя в браузере адрес http://127.0.0.1/test.html. Убедитесь, что файл был успешно отображён.
3. Изучите справку man httpd\_selinux и выясните, какие контексты файлов определены для httpd. Сопоставьте их с типом файла test.html. Проверить контекст файла можно командой ls -Z.

ls -Z /var/www/html/test.html

Рассмотрим полученный контекст детально. Обратите внимание, что так как по умолчанию пользователи CentOS являются свободными от типа (unconfined в переводе с англ. означает свободный), созданному нами файлу test.html был сопоставлен SELinux, пользователь unconfined\_u. Это первая часть контекста. Далее политика ролевого разделения доступа RBAC используется процессами, но не файлами, поэтому роли не имеют никакого значения для файлов. Роль object\_r используется по умолчанию для файлов на «постоянных» носителях и на сетевых файловых системах. (В директории /ргос файлы, относящиеся к процессам, могут иметь роль system\_r. Если активна политика MLS, то могут использоваться и другие роли, например, secadm\_r. Данный случай мы рассматривать не будем, как и предназначение :s0).

Тип httpd\_sys\_content\_t позволяет процессу httpd получить доступ к файлу. Благодаря наличию последнего типа мы получили доступ к файлу при обращении к нему через браузер. 13. Измените контекст файла /var/www/html/test.html с httpd\_sys\_content\_t на любой другой, к которому процесс httpd не должен иметь доступа, например, на samba\_share\_t:

chcon -t samba\_share\_t /var/www/html/test.html  
ls -Z /var/www/html/test.html

После этого проверьте, что контекст поменялся. 14. Попробуйте ещё раз получить доступ к файлу через веб-сервер, введя в браузере адрес http://127.0.0.1/test.html. Вы должны получить сообщение об ошибке:

Forbidden  
You don't have permission to access /test.html on this server.

1. Проанализируйте ситуацию. Почему файл не был отображён, если права доступа позволяют читать этот файл любому пользователю?

ls -l /var/www/html/test.html

Просмотрите log-файлы веб-сервера Apache. Также просмотрите системный лог-файл:

tail /var/log/messages

Если в системе окажутся запущенными процессы setroubleshootd и audtd, то вы также сможете увидеть ошибки, аналогичные указанным выше, в файле /var/log/audit/audit.log. Проверьте это утверждение самостоятельно. 16. Попробуйте запустить веб-сервер Apache на прослушивание ТСР-порта 81 (а не 80, как рекомендует IANA и прописано в /etc/services). Для этого в файле /etc/httpd/httpd.conf найдите строчку Listen 80 и замените её на Listen 81. 17. Выполните перезапуск веб-сервера Apache. Произошёл сбой? Поясните почему? 18. Проанализируйте лог-файлы:

tail -n1 /var/log/messages

Просмотрите файлы /var/log/http/error\_log, /var/log/http/access\_log и /var/log/audit/audit.log и выясните, в каких файлах появились записи. 19. Выполните команду

semanage port -a -t http\_port\_t -р tcp 81

После этого проверьте список портов командой

semanage port -l | grep http\_port\_t

Убедитесь, что порт 81 появился в списке. 20. Попробуйте запустить веб-сервер Apache ещё раз. Поняли ли вы, почему он сейчас запустился, а в предыдущем случае не смог? 21. Верните контекст httpd\_sys\_cоntent\_\_t к файлу /var/www/html/ test.html:

chcon -t httpd\_sys\_content\_t /var/www/html/test.html

После этого попробуйте получить доступ к файлу через веб-сервер, введя в браузере адрес http://127.0.0.1:81/test.html. Вы должны увидеть содержимое файла — слово «test». 22. Исправьте обратно конфигурационный файл apache, вернув Listen 80. 23. Удалите привязку http\_port\_t к 81 порту:

semanage port -d -t http\_port\_t -p tcp 81

и проверьте, что порт 81 удалён. 24. Удалите файл /var/www/html/test.html:

rm /var/www/html/test.html

# 3 Теоретическое введение

**SELinux (Security-Enhanced Linux)** — это система управления доступом на уровне ядра, которая реализует обязательное управление доступом (MAC) в операционных системах Linux. Она позволяет администраторам задавать политику доступа, которая определяет, каким процессам разрешено взаимодействовать с объектами системы (файлы, сокеты и т.д.) [1].

SELinux использует три режима работы [2]:

* Enforcing — политика SELinux активно применяется, и все действия, не соответствующие политике, блокируются.
* Permissive — политика не блокирует действия, но все нарушения записываются в журнал.
* Disabled — SELinux отключен.

Политики SELinux подразделяются на несколько видов, среди которых наиболее распространённой является Targeted Policy, которая защищает только определённые процессы, такие как веб-серверы и службы безопасности, оставляя остальные процессы менее защищёнными.

Команды для управления и проверки статуса SELinux [2]:

* getenforce — показывает текущий режим работы SELinux (Enforcing, Permissive, Disabled)
* sestatus — выводит подробную информацию о текущем состоянии SELinux

**Apache** — это один из самых популярных веб-серверов, который используется для обслуживания веб-сайтов и приложений. При установке и настройке веб-сервера на системе с SELinux важно учитывать, что политика SELinux контролирует доступ Apache к файлам и ресурсам системы [3].

Основные команды [4]:

* service httpd status или /etc/rc.d/init.d/httpd status — проверяет, работает ли веб-сервер
* service httpd start или /etc/rc.d/init.d/httpd start — запускает веб-сервер, если он выключен
* ps auxZ | grep httpd — показывает контекст безопасности процессов веб-сервера Apache
* getsebool -a | grep httpd — выводит список всех SELinux переключателей для Apache и их текущее состояние
* ls -Z /var/www/html/test.html — показывает контекст безопасности файла
* chcon -t samba\_share\_t /var/www/html/test.html — изменит тип контекста на samba\_share\_t, который запрещает доступ Apache к этому файлу
* semanage port -a -t http\_port\_t -p tcp 81 — добавляет порт 81 к списку допустимых для Apache
* semanage port -l | grep http\_port\_t - просмотр списка портов, разрешённых SELinux

При работе с веб-сервером и SELinux важно отслеживать логи, чтобы своевременно обнаруживать и устранять ошибки. Основные файлы логов для Apache:

* /var/log/messages — системный лог.
* /var/log/httpd/error\_log — лог ошибок веб-сервера.
* /var/log/audit/audit.log — лог SELinux, в котором фиксируются события, связанные с нарушениями политики безопасности.

# 4 Выполнение лабораторной работы

Я вошла в систему с полученными учётными данными и убедилась, что SELinux работает в режиме enforcing политики targeted с помощью команд getenforce и sestatus (рис. 1):

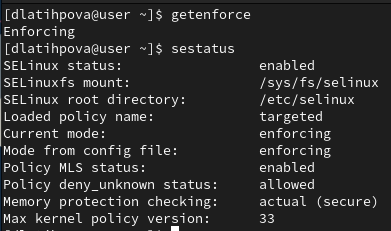


Рис. 1: Пункт 1

Обратилась с помощью браузера к веб-серверу, запущенному на моем компьютере, и убедилась, что последний работает(рис. 2):

service httpd status

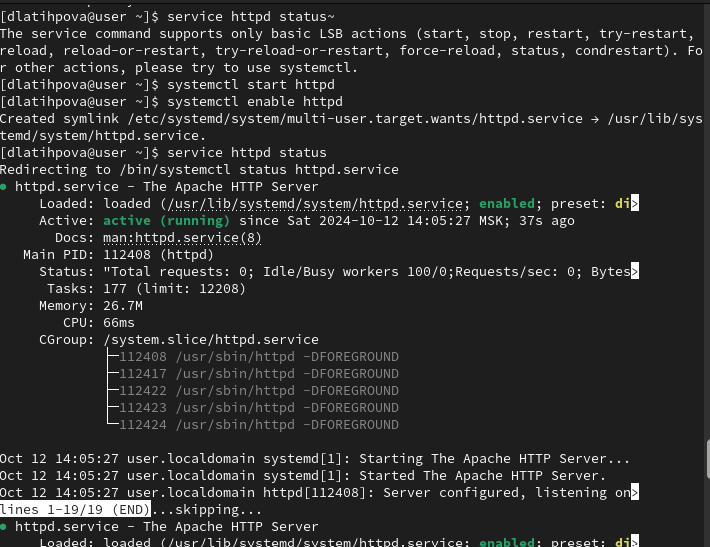


Рис. 2: Пункт 2

Нашла веб-сервер Apache в списке процессов, определила его контекст безопасности - httpd\_t (рис. 3):

ps auxZ | grep httpd

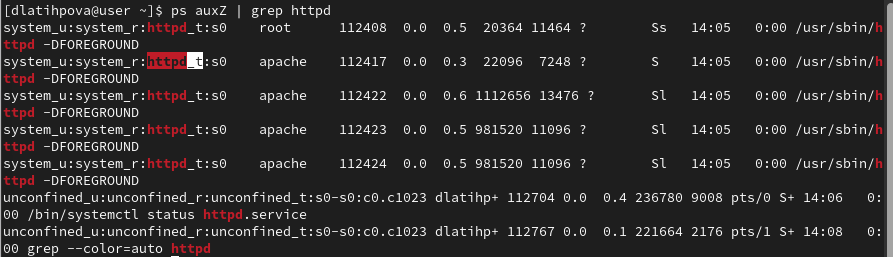


Рис. 3: Пункт 3

Посмотрела текущее состояние переключателей SELinux для Apache с помощью команды:

sestatus -bigrep httpd

Многие из них находятся в положении «off» (рис. 4):

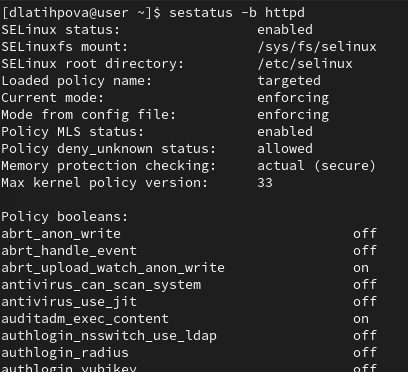


Рис. 4: Пункт 4

Посмотрела статистику по политике с помощью команды seinfo, также выделила множество пользователей, ролей, типов (рис. 5):

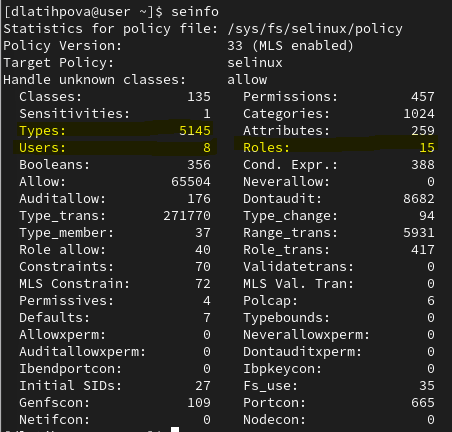


Рис. 5: Пункт 5

Определила тип файлов и поддиректорий, находящихся в директории /var/www, с помощью команды:

ls -lZ /var/www

Определила тип файлов, находящихся в директории /var/www/html:

ls -lZ /var/www/html

Определила круг пользователей, которым разрешено создание файлов в директории /var/www/html - 0 (рис. 6):

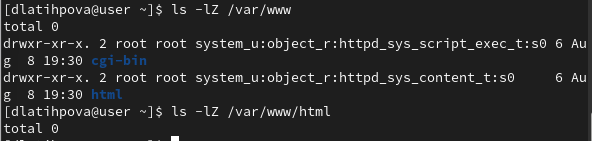


Рис. 6: Пункты 6-8

Создала от имени суперпользователя (так как в дистрибутиве после установки только ему разрешена запись в директорию) html-файл /var/www/html/test.html следующего содержания (рис. 8):

<html>  
<body>test</body>  
</html>

Проверила контекст созданного мной файла (рис. 7):

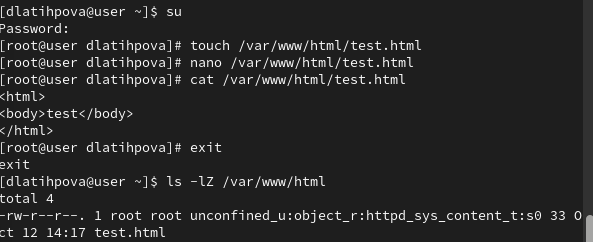


Рис. 7: Пункты 9-10

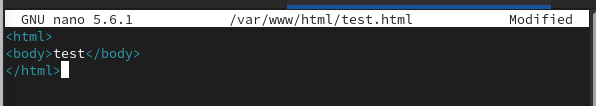


Рис. 8: Содержимое файла test.html

Обратилась к файлу через веб-сервер, введя в браузере адрес http://127.0.0.1/test.html. Файл успешно отображён (рис. 9):

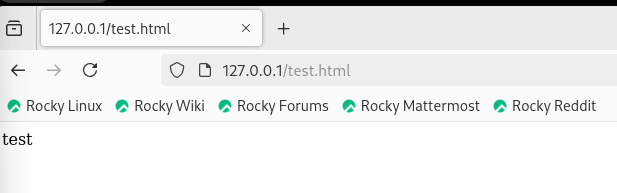


Рис. 9: Пункт 11

Изучила справку man httpd\_selinux и выяснила, какие контексты файлов определены для httpd. Проверила контекст файла командой (рис. 10):

ls -Z /var/www/html/test.html

Пункт 12

Рис. 10: Пункт 12

Изменила контекст файла /var/www/html/test.html с httpd\_sys\_content\_t на samba\_share\_t (рис. 11):

chcon -t samba\_share\_t /var/www/html/test.html  
ls -Z /var/www/html/test.html

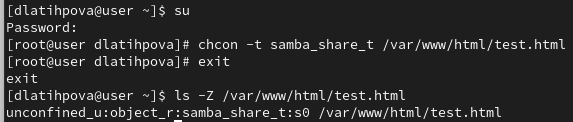


Рис. 11: Пункт 13

1Попробовала ещё раз получить доступ к файлу через веб-сервер, введя в браузере адрес http://127.0.0.1/test.html. Вышла ошибка(рис. 12):

Forbidden  
You don't have permission to access /test.html on this server.

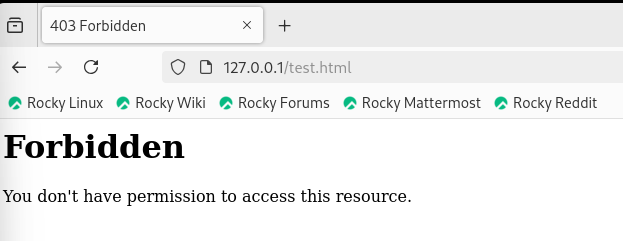


Рис. 12: Пункт 14

Просмотрела log-файлы веб-сервера Apache. Также просмотрела системный лог-файл(рис. 13):

tail /var/log/messages

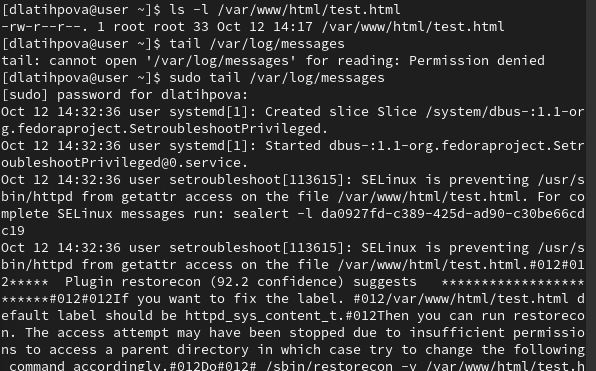


Рис. 13: Пункт 15

Попробовала запустить веб-сервер Apache на прослушивание ТСР-порта 81 (а не 80, как рекомендует IANA и прописано в /etc/services). Для этого в файле /etc/httpd/httpd.conf Listen 80 заменила её на Listen 81 (рис. 14):

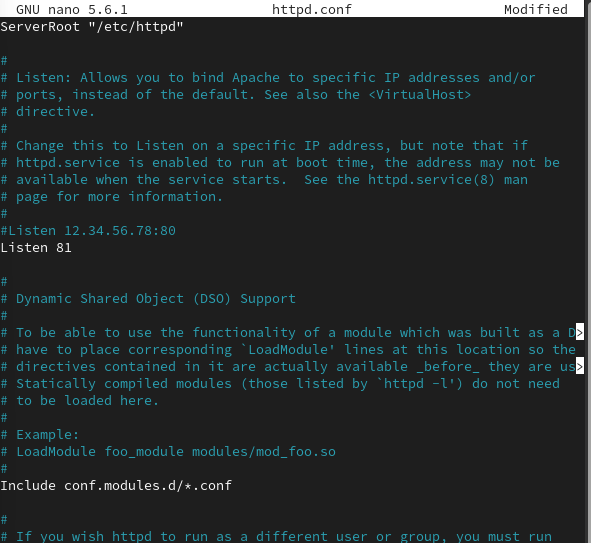


Рис. 14: Замена Listen 80 на Listen 81

Выполнила перезапуск веб-сервера Apache (рис. 15):

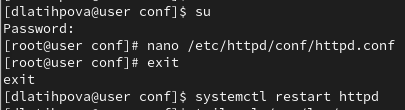


Рис. 15: Пункты 16-17

Проанализировла лог-файлы:

tail -n1 /var/log/messages

Просмотрела файлы /var/log/http/error\_log, /var/log/http/access\_log и /var/log/audit/audit.log. В последнем появились записи (рис. 16):

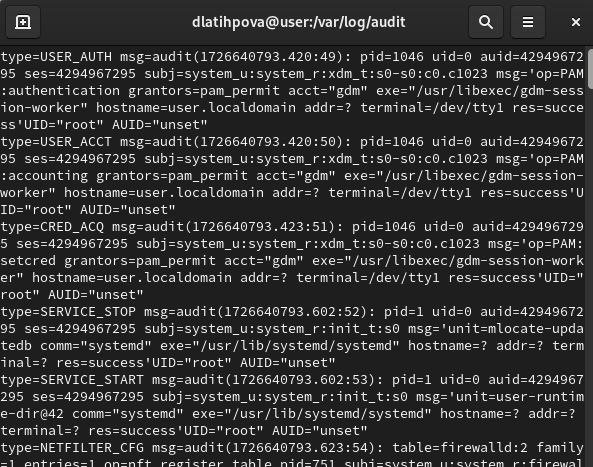


Рис. 16: Пункт 18

Выполнила команду:

semanage port -a -t http\_port\_t -р tcp 81

После этого проверила список портов командой:

semanage port -l | grep http\_port\_t

Порт 81 появился в списке. Попробовала запустить веб-сервер Apache ещё раз (рис. 17):

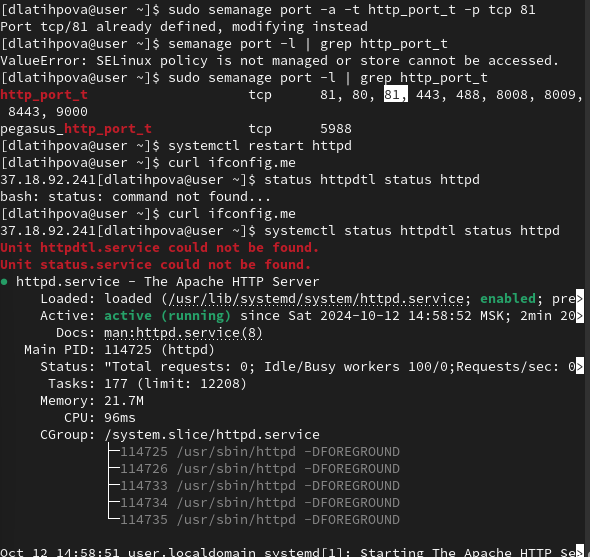


Рис. 17: Пункты 19-20

Вернула контекст httpd\_sys\_cоntent\_\_t к файлу /var/www/html/ test.html:

chcon -t httpd\_sys\_content\_t /var/www/html/test.html

После этого попробовала получить доступ к файлу через веб-сервер, введя в браузере адрес http://127.0.0.1:81/test.html (рис. 18):

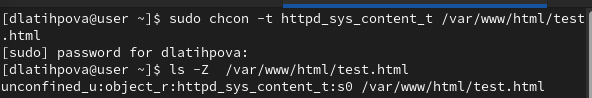


Рис. 18: Пункт 21

Увидела содержимое файла — слово «test» (рис. 19):

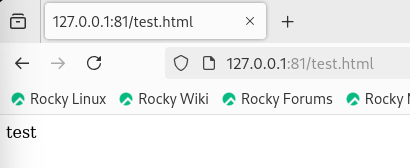


Рис. 19: Проверка сайта

Исправила обратно конфигурационный файл apache, вернув Listen 80 (рис. 20):



Рис. 20: Пункт 22

Удалила привязку http\_port\_t к 81 порту:

semanage port -d -t http\_port\_t -p tcp 81

и проверила, что порт 81 удалён (рис. 21):

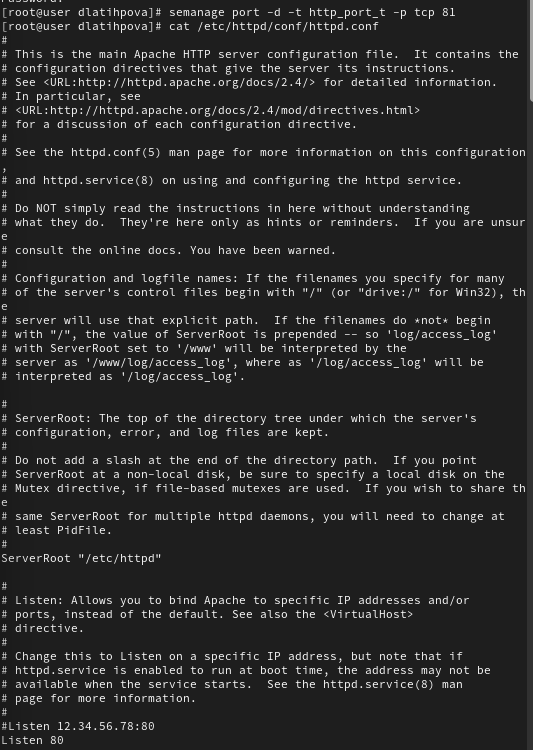


Рис. 21: Пункт 23

Удалила файл /var/www/html/test.html (рис. 22):

rm /var/www/html/test.html

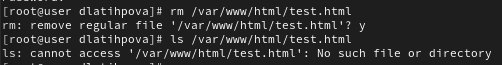


Рис. 22: Пункт 24

# 5 Выводы

Я развила навыки администрирования ОС Linux и получила первое практическое знакомство с технологией SELinux. А также проверила работу SELinx на практике совместно с веб-сервером Apache.

# Список литературы

1. SELinux [Электронный ресурс]. Wikipedia®, 2024. URL: <https://ru.wikipedia.org/wiki/SELinux>.

2. SELinux – описание и особенности работы с системой. Часть 1 [Электронный ресурс]. Habr, 2024. URL: <https://habr.com/ru/companies/kingservers/articles/209644/>.

3. Apache HTTP Server [Электронный ресурс]. Wikipedia®, 2024. URL: <https://ru.wikipedia.org/wiki/Apache_HTTP_Server>.

4. Серверы Linux. Часть I. Серверы Apache и Squid [Электронный ресурс]. Автор: Paul Cobbaut, Перевод: А.Панин, 2015. URL: <https://rus-linux.net/MyLDP/BOOKS/Linux-Servers/ch01.html>.