Второй раздел курса Введение в Linux

Дополнительное задание

Тарутина Кристина Олеговна

Содержание

Цель работы	5
Выполнение лабораторной работы	6
Выволы	21

Список иллюстраций

1	Знакомство с сервером: Задание 1	6
2	Знакомство с сервером: Задание 2	7
3	Знакомство с сервером: Задание 3	7
4	Обмен файлами: Задание 4	8
5	Обмен файлами: Задание 5	8
6	Обмен файлами: Задание 6	9
7	Обмен файлами: Задание 7	9
8	Запуск приложений: Задание 8	10
9	Запуск приложений: Задание 9	10
10	Запуск приложений: Задание 10	11
11	Запуск приложений: Задание 10	11
12	Запуск приложений: Задание 11	12
13	Запуск приложений: Задание 11	12
14	Контроль запускаемых программ: Задание 12	13
15	Контроль запускаемых программ: Задание 13	13
16	Контроль запускаемых программ: Задание 14	14
17	Контроль запускаемых программ: Задание 15	14
18	Многопоточные приложения: Задание 16	15
19	Многопоточные приложения: Задание 17	15
20	Многопоточные приложения: Задание 18	16
21	Многопоточные приложения: Задание 19	16
22	Многопоточные приложения: Задание 20	17
23	Менеджер терминалов tmux: Задание 21	17
24	Менеджер терминалов tmux: Задание 22	18
25	Менеджер терминалов tmux: Задание 23	18
26	Менеджер терминалов tmux: Задание 24	19
27	Менеджер терминалов tmux: Задание 25	19
28	Менеджер терминалов tmux: Задание 26	20
29	Менеджер терминалов tmux: Задание 26	20

Список таблиц

Цель работы

Изучить материалы курса, пройти тестовые и интерактивные задания

Выполнение лабораторной работы

Приступаю к выполнению задания 1. Выбираю все четыре варианта, так как на сервере можно хранить конфеденциальные и общедоступные данные, большие объёмы данных, а также выполнять сложные вычисления(рис. @fig:001).

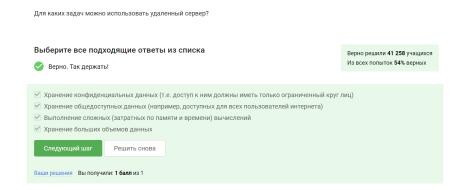


Рис. 1: Знакомство с сервером: Задание 1

Приступаю к выполнению задания 2. В видеоматериалах говорится, что спокойно можно пересылать публичный ключ, так как помимо него есть ещё второй ключ, который как раз таки пересылать не стоит. Публичный ключ хранится в id_rsa.pub(puc. @fig:002)

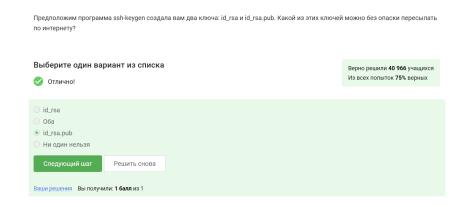


Рис. 2: Знакомство с сервером: Задание 2

Приступаю к выполнению задания 3. Запускаю терминал. Захожу на сервер ssh box@server1.stepik-local(рис. @fig:003) Ввожу указанный в задании пароль. С помощью ниже указанных команд перехожу в нужную директорию pwd cd ../ cd srv cd files_in_server Дальше вывожу в консоль содержимое файла и выхоу из сервера cat secret exit и наконец указанный в файле текст записываю в другой файл уже на компьютере echo "I hacked ssh!" > /home/box/secret

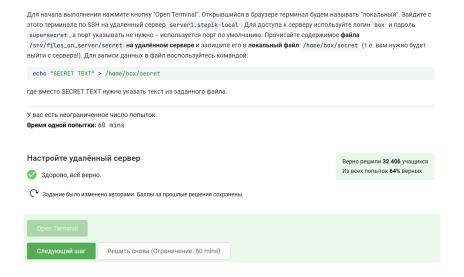


Рис. 3: Знакомство с сервером: Задание 3

Приступаю к выполнению задания 4. Сразу отметаю варианты, которые начинаются с ssh, потому что нам нужна команда scp. Дальше замечаю что вариант

stepik/* скопирует только файлы в папке, но не саму её, а значит правильным ответом является последний вариант(рис. @fig:004)

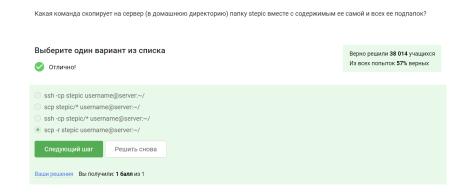


Рис. 4: Обмен файлами: Задание 4

Приступаю к выполнению задания 5. Сразу отметаю варианты с командами sudo ... upgrade и sudo ... -only-upgrade, так как эти команды обновляют не ссылки, а уже установленные пакеты, что в данном случае бесполезно. Так же нам не подходит ответ с местом на диске, следовательно выбирает интернет соединение. Логично при проблемах с загрузкой первым делом проверить именно его.(рис. @fig:005)

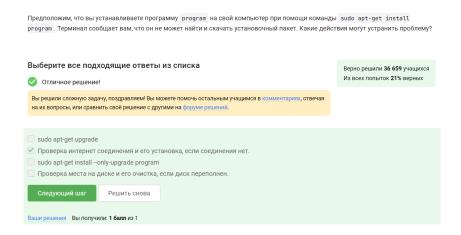


Рис. 5: Обмен файлами: Задание 5

Приступаю к выполнению задания 6. В видеоматериалах к заданию мы рассматривали функционал данного приложения, так что основываясь на них по-

нимаем, что из предложенного только установка приложений на сервер не является его функционалом(рис. @fig:006)

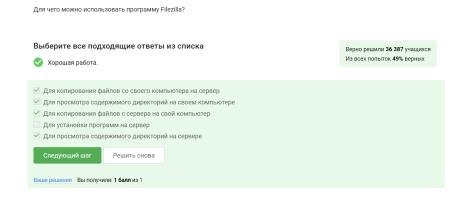


Рис. 6: Обмен файлами: Задание 6

Приступаю к выполнению задания 7. К сожалению его не удалось записать на скринкаст ввиду технический проблем(записался только чёрный экран). Сперва я создаю нужную мне директорию на компьютере mkdir /home/box/files_on_client Далее с помощью команды scp я копирую файлы с сервера на клиент scp -r box@server1.stepik-local:/srv/files_on_server/* ~/files_on_client Во время исполнения данной команды терминал запрашивает у меня пароль. Ввожу его и происходит процесс копирования(рис. @fig:007)

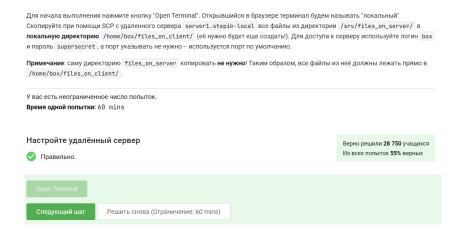


Рис. 7: Обмен файлами: Задание 7

Приступаю к выполнению задания 8. Сразу убираю из возможный вариант

запустить программу на компьютере, так как в задании сказано, что требуется запустить именно на сервере. Вариант где ничего нельзя сделать также нам не подходит, так как мы можем проверить если версия программы для терминала и настроить сервер на вывод информации на экран компьютера(рис. @fig:008).

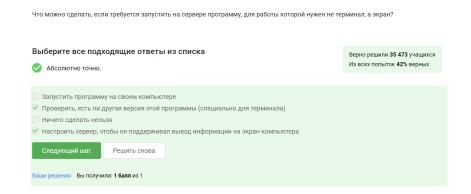


Рис. 8: Запуск приложений: Задание 8

Приступаю к выполнению задания 9. Сразу отметаю вариант с !?, так как нет подобной команды для вывода информации. Все остальные же варианты подходят, способ выведения информации о программе зависит от самой программы(рис. @fig:009)

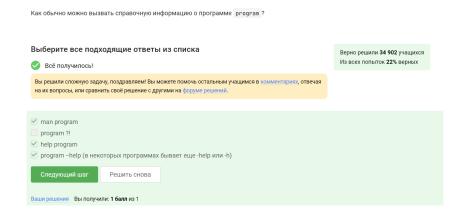


Рис. 9: Запуск приложений: Задание 9

Приступаю к выполнению задания 10. Открываю справку по данной программе, где в строке -f format находится как раз нужный мне ответ(рис. @fig:010 -

@fig:011).



Рис. 10: Запуск приложений: Задание 10

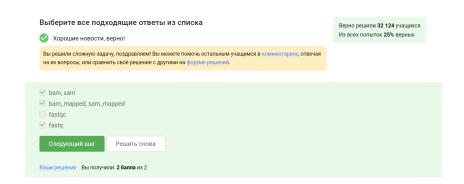


Рис. 11: Запуск приложений: Задание 10

Приступаю к выполнению задания 11. Открываю справку по заданной программе. Там сразу же ищу опцию для сножественного выравнивания(-align). Записываю строку в порядке команда файл опции(рис. @fig:012 - @fig:013).

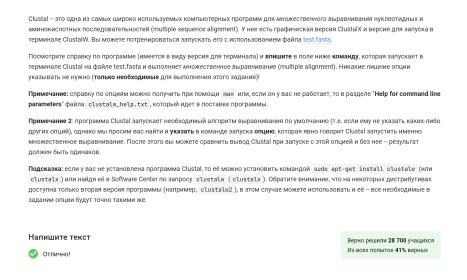


Рис. 12: Запуск приложений: Задание 11

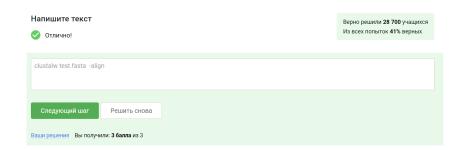


Рис. 13: Запуск приложений: Задание 11

Приступаю к выполнению задания 12. Так как мы прервали выполнение первой программы, то информация о ней не будет выведена, приостановка программы два и работа программы три в фоновом режиме никак не влияет на команду, так что информация о них будет выведена(рис. @fig:014)



Рис. 14: Контроль запускаемых программ: Задание 12

Приступаю к выполнению задания 13. Одинаковые идентификаторы только у рs и top, так как они используют уникальные идентификаторы программ, jobs же присваивает свои, пронумеровав по порядку запуска программ(рис. @fig:015)

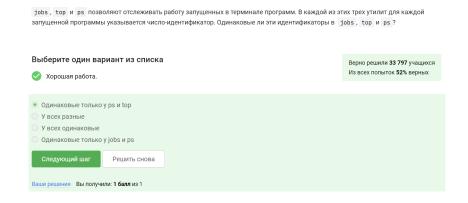


Рис. 15: Контроль запускаемых программ: Задание 13

Приступаю к выполнению задания 14. Мгновенно завершить процесс можно с помощью программы kill -9, однако этого не рекумендуется делать, так как программа kill делает это более плавно, посылая самой программе запрос на выключение(рис. @fig:016)

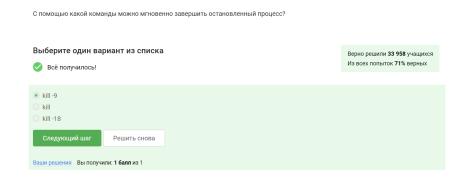


Рис. 16: Контроль запускаемых программ: Задание 14

Приступаю к выполнению задания 15. Процесс приступет к завершению, как только будет продолжен, потому что без опций данная команда посылает лишь запрос на выключение и сама программа делает это во время своей работы, а значит она не закончится, пока приостановлена(рис. @fig:017)

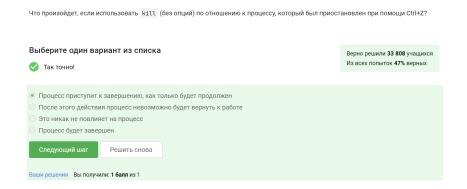


Рис. 17: Контроль запускаемых программ: Задание 15

Приступаю к выполнению задания 16. Программа не использцет вычислительных ресурсов центрального процессора, так как она остановлена и ничего не делает в данный момент(рис. @fig:018)



Рис. 18: Многопоточные приложения: Задание 16

Приступаю к выполнению задания 17. Программа занимает столько же памяти, сколько и во время остановки, так как всё ещё хранит данные и не совершает никаких действий по их удалению(рис. @fig:019)

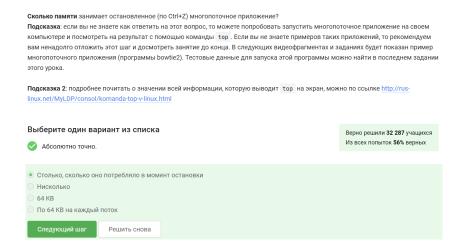


Рис. 19: Многопоточные приложения: Задание 17

Приступаю к выполнению задания 18. Воспользовавшись подсказкой к заданию проверяю данный вопрос на практике, опытным путём устанавливая, что сделать этого нельзя никак(рис. @fig:020)

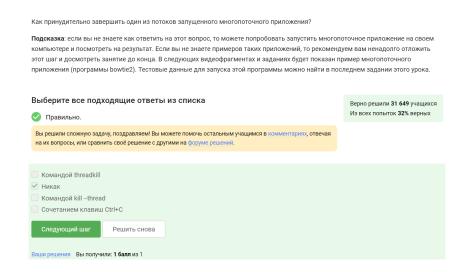


Рис. 20: Многопоточные приложения: Задание 18

Приступаю к выполнению задания 19. По справкам проверяю что в несколько потоков можно запустить как bowtie2, так и bowtie2-build, однако система не принимает этот ответ. В комментариях пишут, что возможно это из-за устаревших заданий, также склоняюсь к этому варианту(рис. @fig:021)

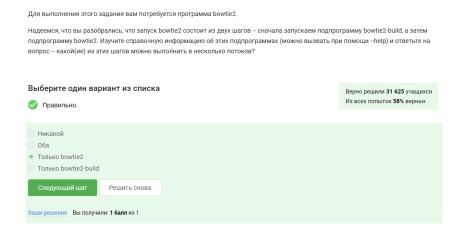


Рис. 21: Многопоточные приложения: Задание 19

Приступаю к выполнению задания 20. С помощью команды wget скачиваю необходимые для работы файлы, запускаю программу и перенаправлю вывод ошибки в нужный мне файл с помощью 2>. Текст файла копирую и вставляю в графу ответа(рис. @fig:022)

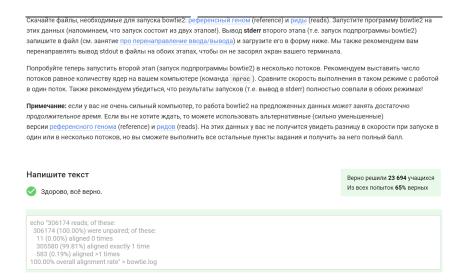


Рис. 22: Многопоточные приложения: Задание 20

Приступаю к выполнению задания 21. Когда мы используем две разные вкладки терминала, то процессы проходящие в них не связаны, а следовательно правильным ответом будет последний(рис. @fig:023)

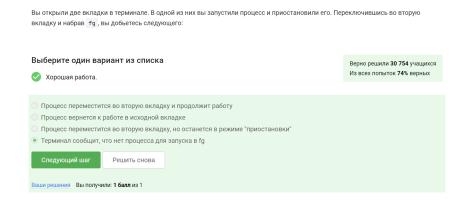


Рис. 23: Менеджер терминалов tmux: Задание 21

Приступаю к выполнению задания 22. В случае если мы ввёдём команду exit, то tmux завершит работу(рис. @fig:024)

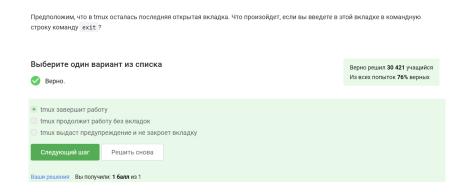


Рис. 24: Менеджер терминалов tmux: Задание 22

Приступаю к выполнению задания 23. Одним из существенных преимуществ tmux, из-за которого им пользуются, является именно то, что даже если соединение с сервером прервётся, он всё равно продолжит работу(рис. @fig:025)

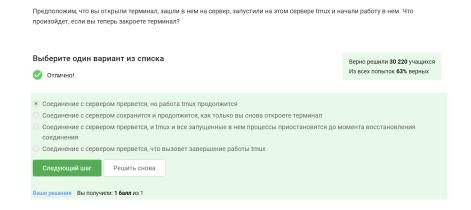


Рис. 25: Менеджер терминалов tmux: Задание 23

Приступаю к выполнению задания 24. Если принудительно закрыть tmux с запущенным в фоновом режиме процессом, то процесс прервётся(хотя в современных версиях он сначала выдаёт предупреждение, новкладку всё равно можно закрыть) (рис. @fig:026)

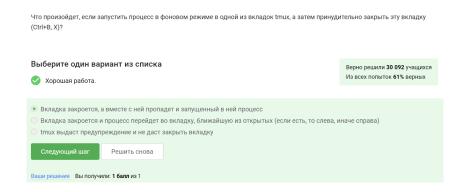


Рис. 26: Менеджер терминалов tmux: Задание 24

Приступаю к выполнению задания 25. Изучаю справку по tmux и выбираю необходимое мне сочетание клавиш(рис. @fig:027)



Рис. 27: Менеджер терминалов tmux: Задание 25

Приступаю к выполнению задания 26. Первое утверждение не верно, разелять можно сколько угодно раз. Второе, третье и четвёртое верны. Пятое не верно, команды разделения действуют в текущей вкладке. Последнее утверждение также верно. (рис. @fig:028 - @fig:029)

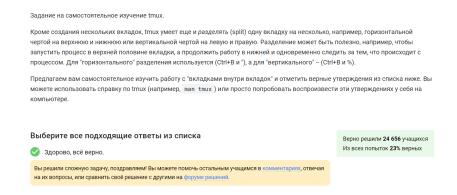


Рис. 28: Менеджер терминалов tmux: Задание 26

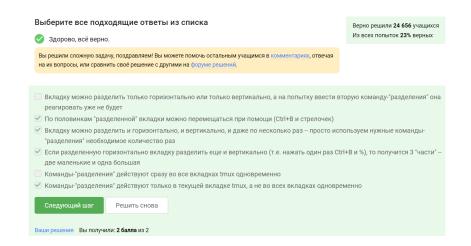


Рис. 29: Менеджер терминалов tmux: Задание 26

Выводы

Курс я прошла успешно, выполнила все задания и получила полный бал, а также составила конспекты по разделу # Список литературы{.unnumbered}