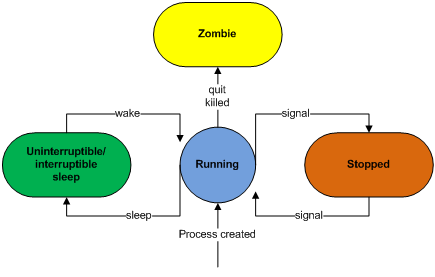
Part1

1. How many states could has a process in Linux?

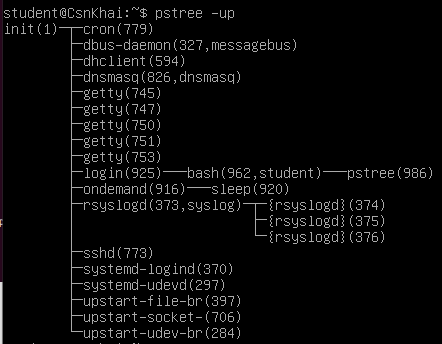
Процессы могут пребывать в следующих состояниях : состояние выполнения (running), состояние сна (uninterruptible/interruptible sleep), состояние остановки выполнения (stopped), zombie-состояние.



2. Examine the pstree command. Make output (highlight) the chain (ancestors) of the current

process.

Команда *pstree* выводит процессы в форме дерева. Одно из преимуществ: вы можете сразу видеть, какой процесс является родителем какого (когда вам нужно будет убить целый ряд процессов, в случае когда интересующие вас процессы являютя дочерними для одного родительского, то вы можете просто убить родителя). Вы можете использовать опцию *-p*, которая отображает PID каждого процесса и опцию*-u*, которая отображает имя пользователя, который запустил этот процесс.



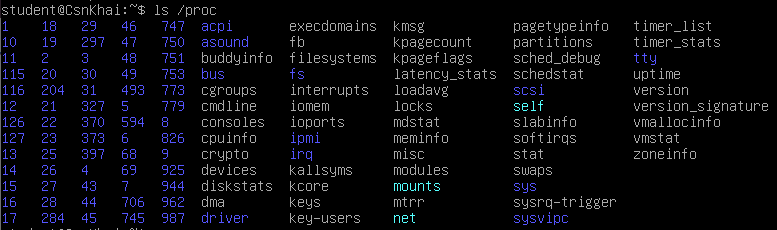
3. What is a proc file system?

В системе Linux присутствует виртуальный файловый объект -*/proc (process)*. Этот объект существует только во время работы системы в оперативной памяти компьютера. Это виртуальная файловая система, которая обеспечивает связь с ядром и монтируется в каталогу /proc.

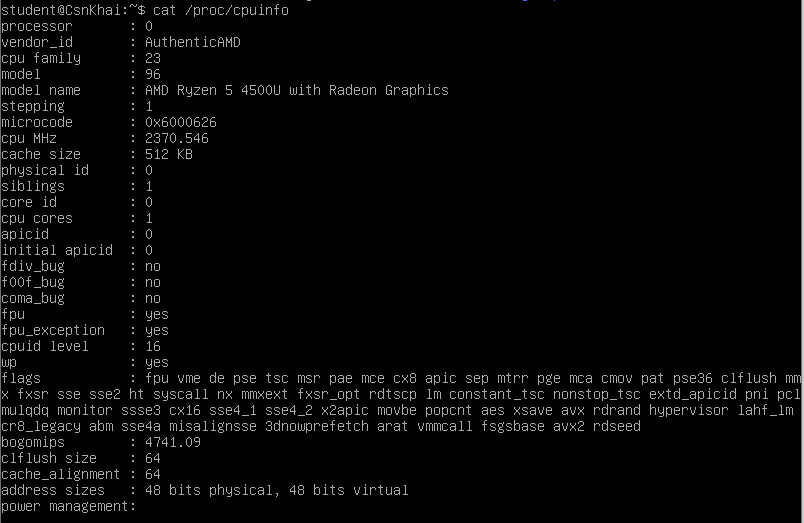
Виртуальный каталог */proc* предназначен для предоставления текущих параметров о компьютерной системе. Многие из утилит, выводящие информацию о системе (например, команда ps), берут исходные данные именно из этого каталога.

В виртуальной файловой системе /proc размер почти всех файлов равен нулю, и любой пользователь, включая администратора, лишен права на запись в эти файлы. Право на чтение почти всех файлов имеет каждый зарегистрированный пользователь системы. В то же время некоторая часть виртуальных файлов доступна и на запись, например, для установки параметров ядра.

Объект */proc* включает в себя файлы и каталоги с числовыми и символьными именами. Каталоги с числовыми именами содержат информацию о каждом выполняющемся процессе, а само имя каталога образовано идентификатором выполняемого процесса PID. При создании процесса соответствующий каталог появляется, а при уничтожении процесса - исчезает. В каждом из таких «числовых» каталогов содержатся одни и те же файловые объекты.



1. Print information about the processor (its type, supported technologies, etc.).

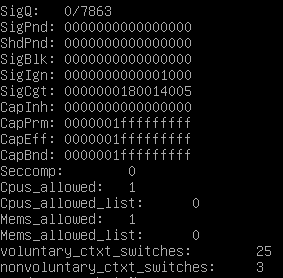
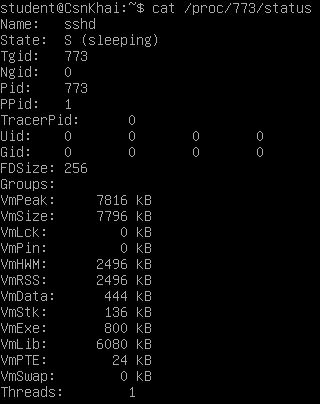


5. Use the ps command to get information about the process. The information should be as

follows: the owner of the process, the arguments with which the process was launched for

execution, the group owner of this process, etc.

//Подробный вывод статуса



//Адрес в ячейках оперативной памяти, которые занял процесс  


//Комманда, которой был запущен процесс  
  


//Символьная ссылка на рабочий каталог



6. How to define kernel processes and user processes?

*Процессы ядра* - части самого ядра, функции общего кода ядра, работающие в том же адресном пространстве и с теми же привилегиями, что и остальной код ядра. Единственное их отличие от других частей ядра: для них создаются отдельные записи в таблице процессов. Процессами они сделаны для того, что бы их выполнение происходило независимо от остальных частей ядра, с более низким приоритетом. Их выполнение происходит под контролем планировщика, как и все остальные процессы в системе.

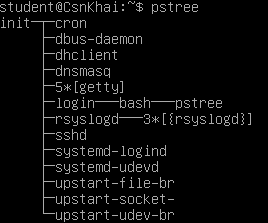
Процессы ядра Linux запускаются самим ядром, при этом родительским процессом якобы их породившим, назначается процесс *kthread*, с *PID=2*.

По умолчанию *pstree* без параметров показывает только дерево процессов порожденных init, т.е. *пользовательских процессов*. Процессы ядра покажет *sudo pstree 2*.

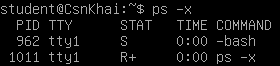
//Процессы ядра



//Пользовательские процессы

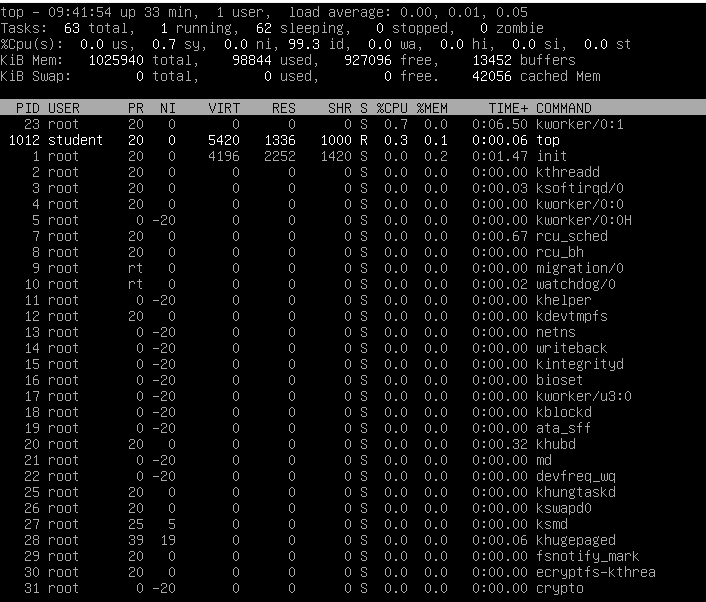


//Процессы от имени текущей учетной записи



7. Print the list of processes to the terminal. Briefly describe the statuses of the processes.

What condition are they in, or can they be arriving in?



Колонка STAT может содержать следующие значения:

*R* - процесс выполняется или готов к выполнению (состояние готовности)

*D* - процесс ожидает дискового ввода/вывода

*T* - процесс остановлен (stopped) или трассируется отладчиком

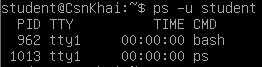
*S* - процесс в состоянии ожидания (sleeping)

*Z* - процесс-зобми

1. Display only the processes of a specific user.

Показано в вопросе 6 (последний скриншот)

Или в с помощью команды ps и опции *-u*



9. What utilities can be used to analyze existing running tasks (by analyzing the help for the ps

command)?

Опции утилиты делятся на два типа - те, которые идут с дефисом Unix и те, которые используются без дефиса - BSD.

*-A, (a)*- выбрать все процессы;

*-a*- выбрать все процессы, кроме фоновых;

*-d, (g)* - выбрать все процессы, даже фоновые, кроме процессов сессий;

*-N*- выбрать все процессы кроме указанных;

*-С* - выбирать процессы по имени команды;

*-G* - выбрать процессы по ID группы;

*-p, (p)*- выбрать процессы PID;

*--ppid*- выбрать процессы по PID родительского процесса;

*-s* - выбрать процессы по ID сессии;

*-t, (t)* - выбрать процессы по tty;

*-u, (U)* - выбрать процессы пользователя.

Опции форматирования:

*-с* - отображать информацию планировщика;

*-f -* вывести максимум доступных данных,;

*-j, (j)*- вывести процессы в стиле Jobs, минимум информации;

*-M, (Z)* - добавить информацию о безопасности;

*-o, (o)* - позволяет определить свой формат вывода;

*--sort, (k)* - выполнять сортировку по указанной колонке;

*-L, (H)* - отображать потоки процессов в колонках LWP и NLWP;

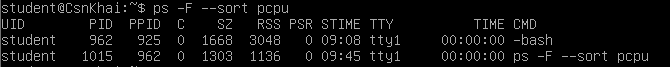
*-m, (m)* - вывести потоки после процесса;

*-V, (V)*- вывести информацию о версии;

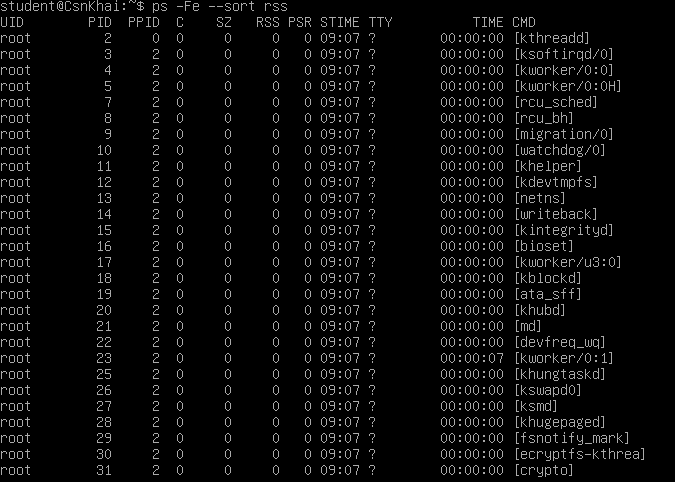
*-H* - отображать дерево процессов linux;

*Примеры*

//Сортировка по проценту загрузки cpu



//Сортировка по проценту загрузки памяти



10. What information does top command display?

Команда *top* в Linux системах позволяет вывести в виде таблицы перечень запущенных процессов и оценить, какой объем ресурсов они потребляют, т.е., какую нагрузку создают на сервер и дисковую подсистему.

Расшифровка данных перед таблицей

*Первая строчка* (top) дает нам общее представление о загрузке системы. Здесь указано:

\*\* текущее время  
\*\* up — продолжительность работы системы с момента запуска  
\*\* user — текущее количество пользователей системы  
\*\* load average — средняя загруженность системы одну минуту назад, пять и 15 соответственно.

*Вторая строка* данных дает информацию об общем количестве процессов в системе, а также об их статусе: выполняемые (running), в ожидании (sleeping), приостановленные (stopped), zombie.

Далее идет информация о состоянии процессора (%Cpu), ОЗУ (Mem), SWAP.

Расшифровка столбцов в таблице

*Первые два столбика* — номер процесса (PID) и имя пользователя, его запустившего (USER).

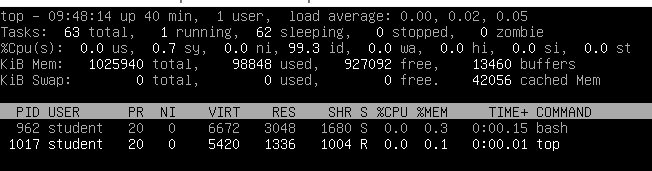
*Следующие 2 столбца* показывают, какой приоритет имеет процесс в данный момент (PR) и приоритет, присвоенный ему командой NICE (NI).

Информация, содержащаяся в других колонках, характеризует непосредственно уровень потребления ресурсов. Расшифровываются они следующим образом:

*\*\*VIRT* — виртуальная память, которую использует процесс  
*\*\*RES* — физическая память, занятая данным процессом  
*\*\*SHR* — общий объем памяти, которую данный процесс делит с другими  
*\*\*S* — текущий статус процесса: R — running; S — sleeping, Z — zombie  
*\*\*%CPU* — процент используемого времени центрального процессора  
*\*\*%MEM* — процент ОЗУ, используемой процессом  
*\*\*TIME+* — продолжительность работы процесса с момента запуска  
*\*\*COMMAND* — название команды (программы), которая инициировала процесс.

1. Display the processes of the specific user using the top command.

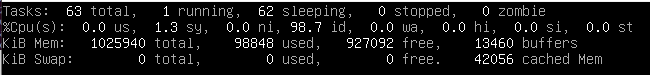




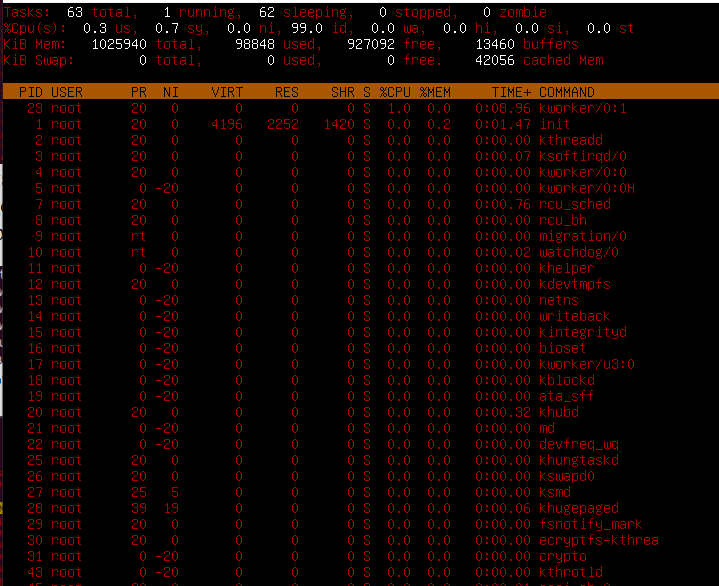
12. What interactive commands can be used to control the top command? Give a couple of

examples.

Убрать либо вернуть верхнюю строку, показывающую нагрузку и аптайм, для этого нажимайте клавишу *l*.



С помощью кнопки *z* можно переключаться между монохромным и цветным отображением.



С помощью кнопки*1* вы можете переключаться и выводить обобщённую информацию о загрузке процессора в одной строке, либо в нескольких строках по каждому ядру.

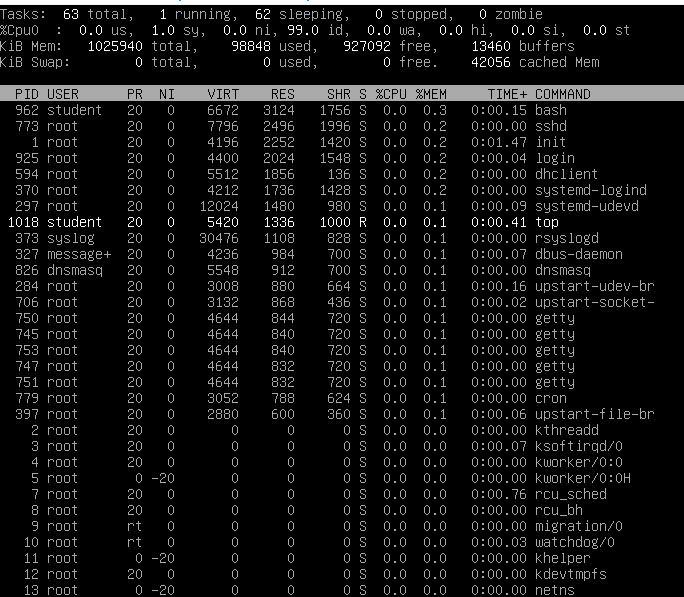


13. Sort the contents of the processes window using various parameters (for example, the

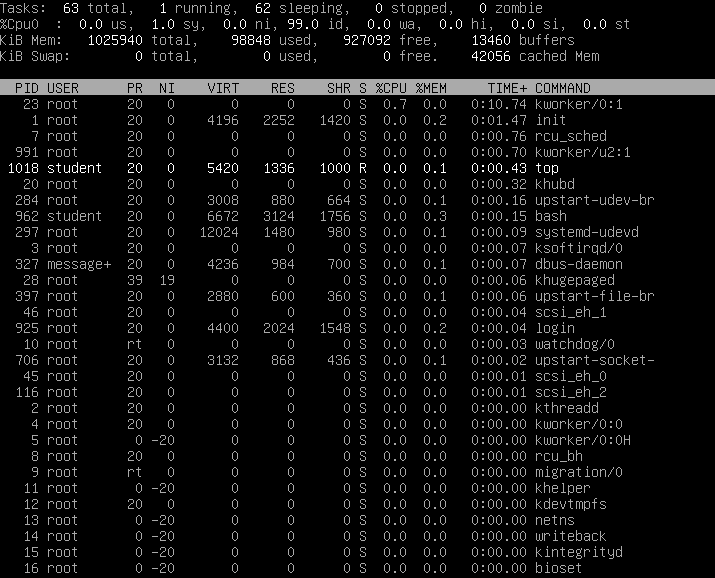
amount of processor time taken up, etc.)

Изначально перечень составлен в порядке убывания нагрузки, приходящейся на процессор.

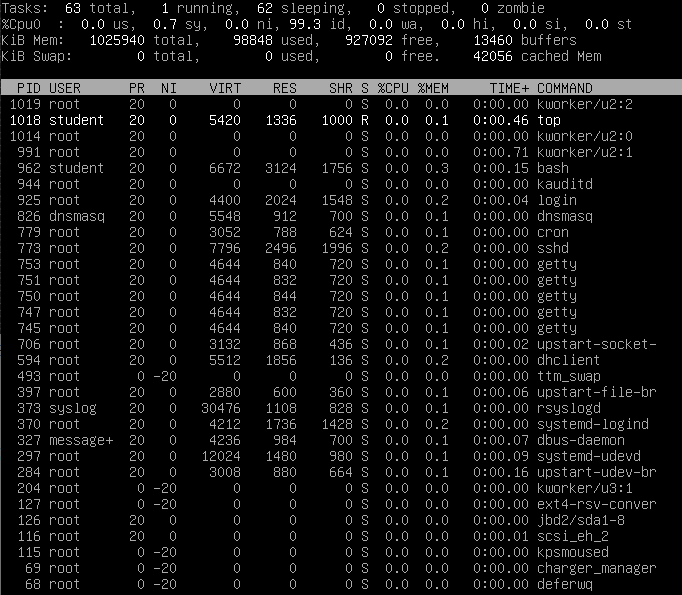
//Сортировка по памяти (shift+M)



//Сортировка по времени (Shift + T)



Можно отсортировать процессы по их номеру (PID), набрав на клавиатуре *Shift+N.*



Чтобы вернуться к режиму сортировки по уровню потребления ресурсов процессора, воспользуйтесь комбинацией *Shift+P*.

1. Concept of priority, what commands are used to set priority?

Все процессы в системе работают с определёнными приоритетами, также называемыми «значениями nice», которые могут изменяться от -20 (наивысший приоритет) до 19 (наименьший приоритет).Для процессов с более высоким приоритетом (меньшим значением nice, вплоть до -20) будет выделено больше системных ресурсов по сравнению с другими процессами с меньшим приоритетом (до 19), предоставляя им большее число циклов процессора. Все пользователи, кроме root, могут только понижать приоритет своих собственных процессов в диапазоне от 0 до 19. Суперпользователь (root) для любого процесса может установить любое значение приоритета.

Для установки приоритета используются команды *nice* и *renice.*

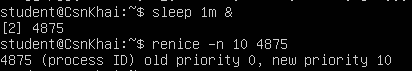
Чтобы запустить программу с определенным приоритетом - *nice*.

Изменить приоритет запущенных задач - *renice*.

//Значение базового приоритета



// Запуск sleep в фоновом режиме и изменение приоритета

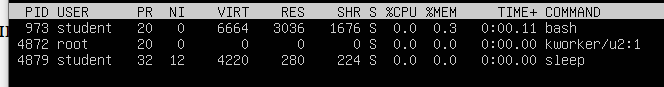


1. Can I change the priority of a process using the top command? If so, how?

Нажмите клавишу *r*, если вы хотите изменить приоритет процесса (не уничтожая сам процесс). Будет запрошен PID процесса для переопределения, введите PID и приоритет.





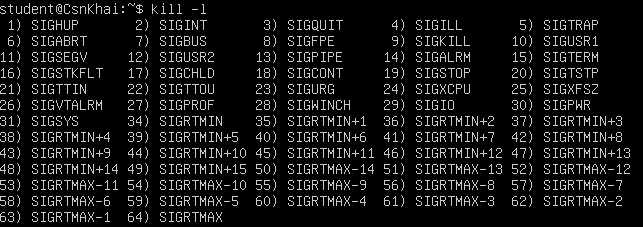


16. Examine the kill command. How to send with the kill command

process control signal? Give an example of commonly used signals.

При выполнении команды "kill" вы посылаете системе сигнал, чтобы заставить ее завершить некорректно ведущее себя приложение.

//Сигналы kill



*SIGTERM* - этот сигнал запрашивает остановку работы процесса. Он может быть проигнорирован. Процессу дается время на корректное завершение. Если программа завершается корректно, значит она использовала данное время на то, чтобы сохранить свое состояние или результаты работы и освободить ресурсы. Другими словами, ее не заставляли остановиться.  
*SIGKILL* - этот сигнал заставляет процесс прекратить работу немедленно. Программа не может проигнорировать этот сигнал. Несохраненные результаты будут потеряны.

//Принудительно завершила sleep (флаг "-9" ссылается на сигнал SIGKILL)



17. Commands jobs, fg, bg, nohup. What are they for? Use the sleep, yes command to

demonstrate the process control mechanism with fg, bg.

Родительский процесс может либо ждать завершения дочернего, либо продолжать свое выполнение. Если в роли родителя выступает командная оболочка, то это значит, что процессы, запущенные с одной консоли (или виртуального терминала), распадаются на две группы: те, которых оболочка ждет, — они взаимодействуют с пользователем, занимая консоль, — и те, после запуска которых с пользователем взаимодействует сама оболочка (консоль свободна). Эти группы называются передним и задним планами.

Чтобы запустить процесс на заднем плане (в фоновом или асинхронном режиме), нужно завершить командную строку управляющим оператором *&*.

Подкоманды оболочки*jobs, bg* (background) и *fg* (foreground) позволяют манипулировать заданиями, выполняющимися на переднем и заднем планах:

 Команда *jobs* выводит список процессов, которые выполняются в фоновом режиме,

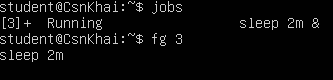
*fg* <номер\_задания> переводит процесс на передний план,

*bg* <номер\_задания> — переводит процесс на задний план.

Номер задания — это не PID, а число, которое команда*jobs* выводит в квадратных скобках. С ключом -I она будет выводить, кроме того, и PID процесса.

//Запустила свой скрипт, в котором команда sleep, вывела *jobs*





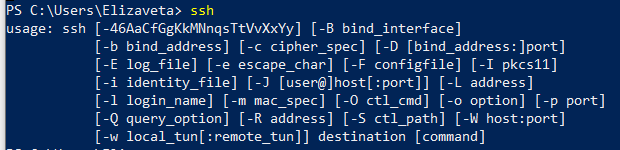
Part2

1. Check the implementability of the most frequently used OPENSSH commands in the MS

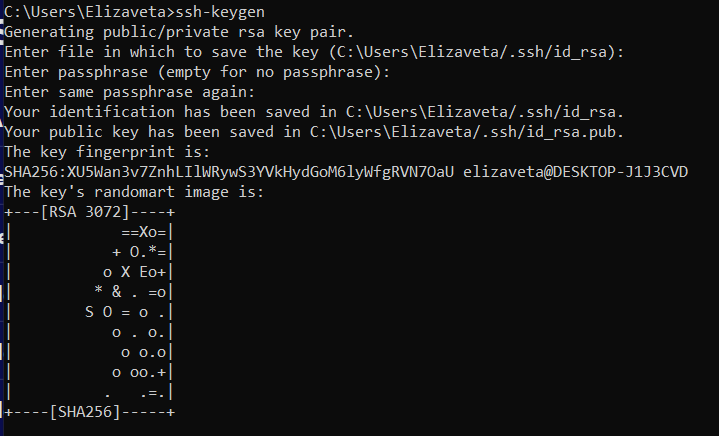
Windows operating system. (Description of the expected result of the commands +

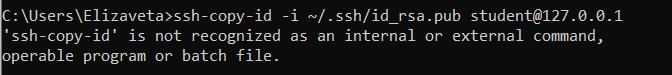
screenshots: command – result should be presented)

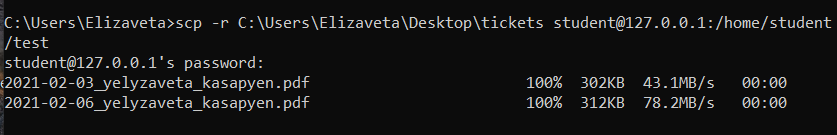
Информация о работе с данной командой:

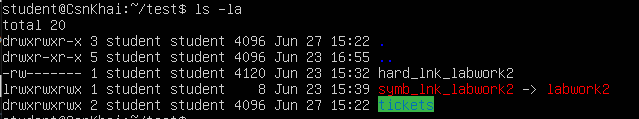


Генерация ключа:









2. Implement basic SSH settings to increase the security of the client-server connection (at least

3. List the options for choosing keys for encryption in SSH. Implement 3 of them.

4. Implement port forwarding for the SSH client from the host machine to the guest Linux

virtual machine behind NAT.

5\*. Intercept (capture) traffic (tcpdump, wireshark) while authorizing the remote client on the

server using ssh, telnet, rlogin. Analyze the result.