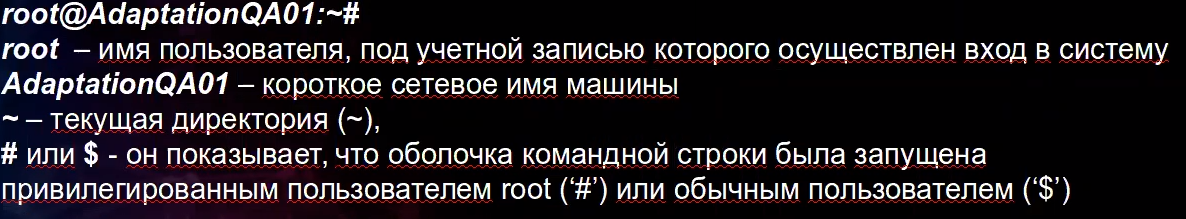
**CLI (command line interface) –** интерфейс командной строки. Данный интерфейс встроен в ядро системы, он будет доступен, даже если графический интерфейс не запустится

****

**Команды Терминала.**

./исполяемаяпрограмма – запускает программу которая находится в настоящем файле (директории). Расширение .sh

. отвечает за настоящую директорию

.. отвечает за директорию выше (Родительскую)

~ - заменяет /home/user (Домашняя директория)

\* - значит все файлы/папки. Ex “\*.txt” – все файлы расширения txt

\ - знак экранирования (использовать перед опцией команды, если мы хотим запустить ее в фоне или сделать grep по опциям команд)

* **man chmod** | **grep** ‘\-r’ -A3 – найти в инфе по команде “-r” и показать 3 строки после нее. Если не поставить экранирование, то grep будет думать, что **-r** это его опция, а не то что нужно найти в файле
* **ping** google.com > pinging.txt & \ ; **tail** **-f** pinging.txt – перед конвейером ; нужно поставить знак экранирования, что бы интерпретатор понял, что конвейер не является продолжением команды, которую мы запустили в фоне

**Просмотр файлов**

**cat** /…/… - показать содержимое файла (что бы не лезть и не искать его через графический интерфейс)

* **–n** имя\_Файла – нумеровать все строки
* **–s** имя\_Файла – удалить повторяющиеся строки

**tac** /../ - показать содержимое файла в обратном порядке

**less** – показ по строчкам в отдельно открытом терминале. Less более новый и стоит использовать его

**more**- показ по строчкам в настоящем терминале

**head** имя\_файла – выводит первые 10 строк

* **-n** колличество\_строк имя\_файла - выводит первые строки в указанном количестве

**tail** имя\_файла – выводит последние 10 строк

* **-n** колличество\_строк имя\_файла – выводит последние строки в указанном количестве
* **–f** имя\_файла – все изменения в файле будут немедленно выводиться на экран
  + Ping google.com > test.txt &
  + Tail –f test.txt

**wc** имя\_файла – выводит количество строк, слов, символов файла

**Поиск файлов**

**grep** ‘слово’ имя\_файла – вывести из файла «words.txt» все строки, содержащие слово «first», в качестве поискового запроса может быть регулярное выражение

**grep** слово ./\* - показать в каких файлах данной директории встречается данное слово

* **–A3** слово ./\* - показать тоже самое, но еще 3 строки после искомого слова
* **–B3** слово ./\* - показать тоже самое, но еще 3 строки перед искомым словом
* **–i** ‘слово’ директория – не принимать во внимание регистр

**whereis** [имя\_команды]/[имя\_файла] – где находится, какая директория

**locate** [имя\_команды]/[имя\_файла] – где находится, какая директория. сканирует более обширно

**find** /директория параметр имя\_файла – поиск файлов

* -**name** названиеФайла - поиск определённого файла (название, формат) в определённой директории
* **-not -name** “\*.txt” – выводит все файлы, которые не txt формата
* **-name** “\*release\*” – находит все файлы, где в имени есть слово release
* **-name** “\*.jpg” **-not -name** “\*photo\*” – находит все картинки, у которых нет в названии photo
* **-atime** колличество\_дней\_назад – показывает файлы, октрытые более заданного количества дней назадэ
* **–maxdepth** число имя\_файла – ищет по заданным уровням вложенности
* **-type** [f][d][l]–поиск по файлам, директориям или символическим ссылкам

**Навигация**

**cd** /название – переход в эту папку через корневую ветвь

**cd** / - переход в корневую папку

**cd** название/название – переход в следующую папку из настоящей папки

**cd** .. – выход из папки в предыдущую (в прошлую директорию)

**cd** ../название – (вначале выход из этой директории, а потом заход в другую папку которая в этой же директории)

**cd** ../.. – поднятие на 2 уровня вверх

**cd** – перемещение в /home/user

**cd** - - вернуться в прошлую папку

**sudo cd** – перемещение в /root

**pwd** – указать в какой папке находишься сейчас

**ls** – список файлов (директорий) в данной директории

* **-a** – посмотреть файлы даже скрытые (у них в начале точка)
* **-l** – посмотреть права доступа к файлам директории
* **-l** имяФайла – посмотреть права на конкретный файл
* **-t** – сортировать по времени изменения (новые в начале)
* **-u** – вместе с **-lt** отсортировать по времени изменения и показать время

**tree** /директория – показывает деревом связь всех папок

**tree** -L 1 / - посмотреть деревом директории, не уходя во вложенные директории.

**Создание файлов (создание, изменение, копирование, удаление, перемещение)**

**mkdir(make dir)** название - создание директории (папки)

**mkdir -p(parent)** название/название – создание вложенной директории в созданной директории

**touch** имя­\_файла.расширение– создаёт файл. Если кликнуть на существующий, то обновляет время его создания. Может быть полезно если мы решим сделать Бэкап только новых файлов, тем самым обновим какой то из старых и он не удалится.

**mv** ../имя\_файла куда\_перемещаем - перемещение файла в другую папку. (для начала нужно перейти в папку с исходным файлом)

**mv** имя\_файла новое\_имя\_файла – переименовать файл

**mv** имя\_файла\*.формат куда\_перемещаем - перемещение файлов где окончание любое (может быть цифры) но название это же

* **-n** – не перезаписывать существующие файлы
* **-f** название\_папки куда\_перемещаем – скопировать папку без спрашивания нужно ли перезаписывать файлы

**cp** имя\_файла /…/… - копирование файла в другую папку.

**cp** имя\_файла новое\_имя\_файла – скопировать файл в текущую директорию

**cp** имя\_файла\*.\* имя\_файла – копирование всех файлов любого формата где есть эта часть названия

* **-R**(рекурсивное) /папка /папка – копирование целой папки с файлами в другую папку
* **–n** – не перезаписывать существующие файлы

**rm** имя\_файла – удалить файл. Удаляет не в корзину, а сразу стирает.

**rm** директория/\* - удаление всех файлов из этой директории

* **-r** имя\_папки - удалить каталог
* **-rf** имя\_папки - удалить каталог если в нем файлы

**ln** путь\_к\_файлу куда – создать дубликат (при смене содержимого в 1, меняется содержимое 2)

* **-s**(shortcut) путь\_к\_файлу/папке путь\_куда – сделать ярлык(ссылку) папки/файла в новом месте. Путь указывать полностью

**echo** “текст” – линия текста

**echo** 'команда; команда;' > файл.txt – в терминале вводим sh файл.txt и терминал выполняет

* **-e** ‘текст\nтекст\nтекст’ - линия текста, только разделенная на строчки

**file** имя\_файла – посмотреть расширение файла

**pdfunite** пдффайл(несколько подряд) пдффайл(что на выходе) – объеденить несколько пдф в один. Идёт с ПРОГРАММОЙ poppler-utils

**wget** ссылкасайта – скачать файл с этой страницы, или же всю страницу HTML

**wkhtmltopdf** ссылкасайта файл – скачать страницы сайта и перевести их в pdf с выводом в файл. ПРОГРАММА

**Манипуляции с файлами (сравнение, сортировка)**

**sort** имя\_Файла – сортирует данные внутри файла

* **-u** имя\_файла – не выводит дублирующиеся строки
* **-n** имя\_файла – используется всегда, если нужно сортировать числа
* **-k** имя\_файла – позволяет задавать объект сортировки (столбцы, колонки, и тому подобные элементы форматирования файла)

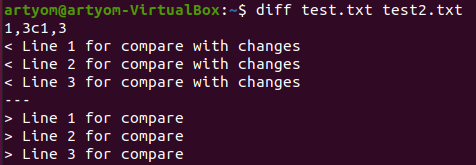
**uniq** имя\_файла– выводит только уникальные (не повторяющиеся строки). Строки должны идти друг за другом. По этому может потребоваться заранее сортировка **sort**

* **–u** имя\_файла – выводит только те строки, которые **не** повторяются на входе
* **–d** имя\_файла – выводит только те строки, которые повторяются на входе
* **–c** имя\_файла – перед каждой строкой выводит число повторений этой строки на входе и один пробел
* **–i** имя\_файла – сравнивает строки без учета регистра

**diff** имя\_файла имя\_второго\_файла – программа, которая сравнивает 2 файла.

Если линии отличатся, то это **<** если совпадают, то это **>**

**1**, **3с1**, **3**. **начало строки**, **с какой по какую строку не совпадают**, **конечная строка в файле**



* **–d** имя\_файла имя\_второго\_файла – выводит только отличия
* **–s** имя\_файла имя\_второго\_файла – выводит только совпадающие части
* **–y** имя\_файла имя\_второго\_файла – выводит в форме двух колонок
* **–r** имя\_файла имя\_второго\_файла – рекурсивное сравнение каталогов
* **–i** имя\_файла имя\_второго\_файла – игнорировать регистр

**Установка программ**

**sudo nano /etc/apt/source.list** – открываем файл с репозиториями

* **deb** http://url [release] [component] - добавляем туда APT строку. Чем «выше» (т.е. ближе к началу файла) стоит строка, тем больший приоритет получит добавленный репозиторий

**sudo add-apt-repository ppa:**репозиторий/ppa - добавление репозитория

**sudo add-apt-repository -r ppa:**репозиторий/ppa – удаление репозитория

**sudo apt-cache search** имя\_пакета -поиск как называется прилага в репозитории

**sudo apt-get update** – обновляет индексы пакетов. Обновление базы обновлений. Ничего не устанавливает

**sudo apt-get upgrade** – установить обновления

**sudo apt dist-upgrade** – установить обновление с удалением старых версий

**sudo apt-get download** имя\_пакета – загрузка пакета без установки

**sudo apt-get install** имя\_пакета – скачать приложение

* **-y -q** – не что бы в не задавался вопрос “вы уверены”
* **--no-install-recommends** – что бы установились только нужные зависимости без “мусора”

**sudo apt-get install** ./файл\_установки – установить приложение из текущей папки

**sudo apt-get remove** имя\_пакета – удаляет приложение, но не его зависимости или конфигурационные файлы

**sudo apt-get purge** имя\_пакета – удаляет приложение, конфигурационные файлы, но не зависимости

**sudo apt-get autoremove** – удаляет неиспользуемые зависимости (делать после удаления приложения)

**dpkg -l** – получение списка пакетов в системе

**dpkg -L** имя\_пакета – список файлов пакета в системе

**dpkg -S** путь\_к\_файлу – какому пакету принадлежит файл

**dpkg -c** пакет – просмотр содержимого пакета

**dpkg –s** имя\_пакета – проверить установлена ли запрашиваемая программа. Dpkg (система управления файлами deb расширения, только в debian)

**dpkg -i** имя\_пакета – установить пакет формата .deb

**dpkg -r** имя\_пакета – удаление пакета

**dpkg --extract** имя\_файла путь\_папки – распаковать .deb файл

**Git clone** (github…) – скачать (клонировать) утилиту с сайта github. Git – система управления версиями, ветвь разработки приложения. Github – всемирная сеть, ветвь, разработчиков, куда они выкладывают свои приложения, утилиты.

**snap list** – список всех установленных snap пакетов (Установленные snap пакеты не отображаются в выводе dpkg -l)

**snap find** имя\_пакета – найти пакет в snapstore

**snap install** имя\_пакета – установка пакета

**snap refresh** имя\_пакета – обновить пакет/пакеты

**snap remove** имя\_пакета – удаление пакета

**tar -xf** (extract file) название\_файла – распаковать файл

**tar -cf** (create File) название\_дирректории – архивировать папку

Многие пакеты при установке запрашивают данные для конфигурации. Примерами таких пакетов могут служить:

* mysql — при установке спрашивает пароль пользователя root;
* libnss-ldap — спрашивает данные на доступ к LDAP-серверу.

Для обычного пользователя такое поведение apt-get может быть полезным, однако если вы пытаетесь установить эти пакеты в автоматическом режиме, оно станет серьезной помехой.

Для того, чтобы инсталлировать пакет в Debian/Ubuntu Linux «без лишних вопросов», надо установить переменную окружения DEBIAN\_FRONTEND в значение noninteractive. Это можно сделать следующим образом.

sudo **DEBIAN\_FRONTEND=noninteractive** apt-get install имя\_пакета

Если же вам надо устанавливать несколько пакетов, воспользуйтесь export:

**export DEBIAN\_FRONTEND=noninteractive**

**Права доступа**

**su** – переключиться на root пользователя (switch user), дать права на сессию (пока не истечет сессия). Запрашивает пароль рута

**su** имя\_пользователя – переключиться на другого пользователя. Запрашивает пароль пользователя

**sudo** команда – запрашивает пароль пользователя, если он есть в **etc/sudoers**. Выдает права на 15 минут и позволяет выполнить команду

**visudo /etc/sudoers** – для редактирования файла sudoers используем редактор visudo

**sudo useradd -m** -g [основная группа] -G [дополнительные группы] -s [командный интерпретатор] имя\_пользователя

* **-m** — создаёт домашний каталог пользователя, вида /home/[имя пользователя]; в пределах которого, пользователь, не имеющий прав доступа root, может создавать и удалять файлы, устанавливать программы, и т. д.
* **-g** — определяет имя или номер основной группы пользователя; группа должна существовать; номер группы должен относится к уже существующей группе; если параметр не указан, пользователю будет присвоена группа в соответствии с переменной USERGROUPS\_ENAB, находящейся в /etc/login.defs.
* **-G** — определяет список дополнительных групп, в которые входит пользователь; каждая группа отделяется от другой запятой без пробелов; по умолчанию пользователь принадлежит только основной группе.
* **-s** — определяет командную оболочку пользователя (например /bin/bash); после завершения запуска системы и логина в нее пользователя, командная оболочка будет той, что указана в данном параметре.

**sudo passwd** имя\_пользователя – изменить пароль

* **–d** – удалить пароль пользователю
* **– l** – заблокировать учетную запись пользователя
* **–u** – Разблокировать учетную запись пользователя

**sudo usermod** имя\_пользователя – редактирование пользователя

* **-a -G** имя\_группы имя\_пользователя – добавить пользователя в группу

**sudo gpasswd --delete** имя\_пользователя имя\_группы – удаление пользователя из группы

**sudo userdel** имя\_пользователя – удаление пользователя

* **-f** имя\_пользователя – удаляет пользователя, если он сейчас работает в системе
* **-r** имя\_пользователя – удаляет домашний каталог пользователя

Группы хранятся в /etc/group

**sudo groupadd** имя\_группы – добавление группы

* **-g** номерID имя\_группы – установить собственный GID
* **-p** пароль имя\_группы – установить пароль на группу
* **-r** имя\_группы – создать системную группу

**sudo groupmod** имя\_группы – изменение группы

* **-g** имя\_группы – установить другой GID
* **-n** новое\_имя\_группы имя\_группы – новое имя группы
* **-p** имя\_группы – изменить пароль группы

**sudo groupdel** имя\_группы – удаление группы

**sudo chmod 3цифры** имя\_файла – определить доступ другим пользователям к этому файлу

* **-R 3цифры** имя\_папки - определить доступ другим пользователям ко всем файлам вложенных в данный

4 — разрешено только чтение

2 — только изменение/запись

1 — только запуск/исполнение

0 — ничего не разрешен

**sudo chmod** u=rwx,g=rx mydir1 = 750

**sudo chmod** u-r,g+w = добавление/удаление прав

* u – пользователь
* g – группа
* o – остальные юзеры
* a – все юзеры

**sudo chown** имя\_пользователя имя\_файла– поменять владельца для файла

**sudo chgrp** имя\_группы имя\_файла – изменить группу файлов/каталогов

**Управление процессами**

**sudo ls /etc/init.d** – получить список сервисов

**service** **--status-all** – получить список сервисов, а так же увидеть какие из них запущены

**sudo** **service** имя\_сервиса restart– перезапуск сети

**sudo** **service** имя\_сервиса stop – остановить сервис

**sudo systemctl status** имя\_сервиса– посмотреть статус сервиса

**sudo systemctl [start, stop disable, restart,..]** имя\_сервиса.service – управление сервисами

**sudo systemctl** **list-unit-files** – посмотреть все файлы конфигурации служб на диске

**sudo systemctl** **list-units** **--type** service **--state** running – показывает только запущенные в данный момент службы

**sudo systemctl** **list-units** **--type** service -**all** – получить список служб linux

**sudo systemctl** **enable** **имяСерсвиса** – добавить службу в автозагрузку

**sudo systemctl** **disable** **имяСервиса** – убрать службу из автозагрузки

**sudo systemctl** **is-enabled** **имяСервиса** – посмотреть разрешена ли сейчас автозагрузка для службы

**ls** \*.py > output.txt & - запускает команду в фоновом режиме

**jobs** – находит задания запущенные в фоновом режиме

**jobs –l** – отображает PID процессов

**%**идентификатор**\_**процесса – переводит задачу из фонового режима в приоритетный

Что бы обратно перевести задачу в фоновый режим, нужно нажать CTRL + Z, при этом выполнение будет приостановлено

%идентификатор\_процесса & - опять запустить задачу в фоновом режиме, после остановки

**kill %**номер\_job – убить фоновый процесс

**fg** – вывести job из фона

**bg** – отправить job в фон

**ps aux** – посмотреть все процессы

**ps -e** – посмотреть все процессы

**kill** номер\_процесса – завершить процесс

**kill all** название\_процесса – убивает все процессы (например у Chrome может быть запущено 10+ процессов)

**kill** -сигнал PIDпроцесса – завершить процесс

* 2 (INT) - Interrupt. В случае выполнения простых команд, вызывает прекращение выполнения, в интерактивных программах — прекращение активного процесса
* 9 (KILL) - Всегда прекращает выполнение процесса
* 15 (TERM) - Software Termination. Требование закончить процесс (программное завершение)
* 18 (CONT) - Продолжение выполнения приостановленного процесса
* 19 (STOP) - Приостановка выполнения процесса

**cat /proc/meminfo** – информация о потреблении RAM

**free** – информация о потреблении RAM

**cat /proc/cpuinfo** – информация о потреблении CPU

**cat /proc/loadavg** – хранит среднюю загрузку процессора за последние 1,5 и 15 минут

**top** – посмотреть динамическое изменение загруженности системы

**PID** – айдишник процесса

**USER** – пользователь от которого запущен процесс

**PR** – приоритет процесса

**NI** – nice value процесса

**VIRT** – виртуальная память

**RES** – физическа память, которая не в swap

**SHR** – объединенная память

**S** – статус процесса

• 'D' = uninterruptible sleep

• 'R' = running

• 'S' = sleeping

• 'T' = traced or stopped

• 'Z' = zombie (Такое может произойти, если **процесс-потомок** (дочерний процесс) завершился раньше, чем этого ожидал **процесс-родитель**. **Процессы-зомби** не занимают процессорного времени (т. е. их выполнение прекращается), но соответствующие им структуры ядра не освобождаются. В некотором смысле это «мертвые» процессы.

**%CPU** – загруженность процессора

**%MEM** – загруженномть памяти

**TIME+** - время которое процесс выполняется

**COMMAND** – название процесса

**Сеть**

**Канальный уровень Статус сети**

**Ifconfig –** просмотр конфигурации сети

**ip link show** – показать все сетевые интерфейсы

**ip link show up** – показывает статус текущих сетевых интерфейсов

**sudo lspci** – список pci устройств, среди которых есть и сетевые карты

**cat /proc/net/dev** – показывает все сетевые интерфейсы, подключенные к системе

**sudo lshw -class network** – просмотр конфигурации сети. Показывает Ethernet интерфейс с логическим именем ethX вместе с информацией по шине, сведениями о драйвере и всеми поддерживаемыми возможностями.

**Канальный уровень - Интерфейсы**

**sudo ip** **link set** **dev** существующийEthИнтерфейс **up –** Поднять интерфейс

**sudo ip link set** **dev** существующийEthИнтерфейс **down –** отключить интерфейс

**ifconfig** интерфейс **up** – поднять интерфейс

**ifconfig** интерфейс **down** – отключить интерфейс

**sudo ifconfig** существующийEthИнтерфейс:новыйНомерВиртуальногоИнтерейса ip\_address **netmask** маска **broadcast** ip\_address **up**– создание временного виртуального сетевого интерфейса + назначение ему ip адреса, маски сети, широковещательного адреса и поднятие интерфейса

**sudo ifconfig** существующийEthИнтерфейс ip\_address **netmask** маска **broadcast** ip\_address– назначение ip адреса, маски сети, широковещательного адреса уже существующему интерфейсу

**Сетевой уровень Статус сети**

**ip address** **show** - показывает статус всех сетевых интерфейсов + IP адреса

**Сетевой уровень - Адреса**

**sudo ip** **address add** ip\_address/mask **broadcast** ip\_address **dev** существующийEthИнтерфейс **–** добавление нового IP адреса, маски сети, широковещательного адреса

**sudo ip address add** ip\_address/mask **broadcast** ip\_address **dev** существующийEthИнтерфейс **label** существующийEthИнтерфейс: новыйНомерВиртуальногоИнтерейса – создание нового интерфейса

**sudo ip** **address del** ip\_address/mask **dev** существующийEthИнтерфейс **–** удаление IP адреса

**Маршрутизация – просмотр таблицы**

**route -n** – открыть таблицу маршрутизации

**netstat -rn** – открыть таблицу маршрутизации

**ip route show** – открыть таблицу маршрутизации

**Маршрутизация – маршруты**

**sudo route add default gw** ip\_шлюза интерфейс – добавить шлюз по умолчанию (где destination будет 0.0.0.0)

**sudo route add -net** ip\_address/mask **gw** ip\_шлюза – добавление маршрута через шлюз (нужно указывать ip адрес сети)

**sudo** **route add -net** ip\_address/mask **dev** интерфейс – добавление маршрута через интерфейс (нужно указывать ip адрес сети)

**sudo** **route add -host** ip\_address **gw** ip\_шлюза – маршрут до отдельного хоста (нужно указывать ip адрес хоста и не указывать маску сети)

**sudo** **route del default –** удаление маршрута по умолчанию

**sudo** **route del -net** ip\_address/mask – удаление маршрута к сети

**sudo** **route del -host** ip\_address/mask – удаление маршрута к отдельному хосту

**sudo** **ip route add default** **via** ip\_шлюза – маршрут по умолчанию для всех адресов

**sudo** **ip route add default** **dev** интерфейс – маршрут по умолчанию для всех адресов

**sudo** **ip route add** ip\_address/mask **via** ip\_шлюза – добавление маршрута через шлюз (нужно указывать ip адрес сети)

**sudo ip route add** ip\_address/mask **dev** интерфейс - добавление маршрута через интерфейс (нужно указывать ip адрес сети)

**sudo** **ip route add** ip\_address/mask **dev** интерфейс **metric** метрика – маршрут с метрикой

**sudo ip route change** ip\_address/mask **via** ip\_шлюза – поменять gateway у маршрута

**sudo ip route del** ip\_address/mask – удаление маршрута

**sudo ip route del default** – удаление шлюза по умолчанию (маршрута во внешнюю сеть)

**sudo ip route flush table main** – очистить всю routing таблицу (после этого перезапускаем интерфейс, что бы восстановился дефолтный маршрут)

**Диагностика сети**

**ping** ip\_address – посылает ICMP пакеты для проверки доступности узла

**traceroute/tracepath** ip\_address – показывает путь пакета до адреса назначения

**mtr** ip\_address – утилита, которая объединяет ping + traceroute

**Firewall**

**$ ufw опции действие параметры**

**Включение Fire Wall**

**sudo ufw status** – проверяем статус firewall

sudo ufw status verbose

**sudo ufw enable** – включаем firewall (если мы работаем по SSH, нужно заранее добавить правило, разрешающее работу по SSH. Иначе соединение прервется)

**Политики по умолчанию**

По умолчанию UFW настройки запрещают все входящие соединения и разрешают все исходящие. Это значит, что если кто-то попытается  достичь ваш сервер, он не сможет подключиться, в то время как любое приложение на сервере имеет доступ к внешним соединениям.  
  
Cоответствующие правила фаервола прописываются так:

**sudo ufw default deny incoming**  
  
**sudo ufw default allow outgoing**

**Добавление правил**

Чтобы создать разрешающее правило, используется команда allow. Вместо allow могут использоваться и запрещающие правила ufw - deny и reject. Они отличаются тем, что для deny компьютер отсылает отправителю пакет с уведомлением об ошибке, а для reject просто отбрасывает пакет и ничего не отсылает. Для добавления правил можно использовать простой синтаксис:

**$ ufw allow имя\_службы**  
**$ ufw allow порт**  
**$ ufw allow порт/протокол**

**sudo ufw allow** [protocol\_name][port] – разрешить соединение, можно использовать либо название протокола, либо его порт

**sudo ufw allow** 22/tcp – разрешить входящие и исходящие подключения только по протоколу tcp

**sudo ufw allow out** 80/tcp – разрешить исходящий трафик на 80 порт по протоколу tcp

**sudo ufw deny in** 80/tcp – запретить входящий трафик на порт 80 по протоколу tcp

**sudo ufw limit** 22/tcp – если с одного ip адреса будет инициировано более 6 подключений в течении 30 секунд, соединение будет запрещено

**Добавление диапазона портов**

**sudo ufw allow** 3000:3100/tcp – обязательно указываем TCP или UDP протокол при добавлении диапазона портов   
**sudo ufw allow** 3000:3100/udp–обязательно указываем TCP или UDP протокол при добавлении диапазона портов

**Добавление IP адресов**

**sudo ufw allow from** ip\_address – указать IP адрес которому будет разрешен доступ ко всем портам

**sudo ufw allow from** ip\_address **to any port** [port] – задать доступ только к определенному порту

**sudo ufw allow from** ip\_address/mask – указать диапазон IP адресов которым будет разрешен доступ ко всем портам

**sudo ufw allow from** ip\_address/mask **to any port** [port] – задать доступ только к определенному порту диапазону IP адресов

**Удаление правил**

**sudo ufw status numbered** – узнаем какие правила сейчас активны и их номер

**sudo ufw delete** номерПравила – удаляет правило

**sudo ufw delete allow** [protocol\_name][port] – удаляет правило

**Отключение Firewall, сброс правил**

**sudo ufw disable –** отключение firewall

**sudo ufw reset** – сброс правил

**Логи**

**ls /var/log/ufw\*** - просмотр файлов логов

**sudo ufw logging on** – включить логирование

**sudo ufw logging off** – выключить логирование

**sudo ufw logging [off][low][medium][high][full]** – задать уровень логирования

* **off** - отключен.
* **low**- регистрирует все заблокированные пакеты, не соответствующие заданной политике (с ограничением скорости), а также пакеты, соответствующие зарегистрированным правилам.
* **medium**- все то, что при значении **low.**Плюс все разрешенные пакеты, не соответствующие заданной политике, все недопустимые пакеты, и все новые соединения. Все записи ведутся с ограничением скорости.
* **high**-работает также как и **medium.**Плюс все пакеты с ограничением скорости.
* **full** - также как и **high,**но без ограниения скорости.

**Жесткий диск**

**df** – выводит информацию о свободном, занятом месте на дисках.

* **–h** – выводит информацию в удобном для восприятия виде

**sudo fdisk -l** – посмотреть какие диски есть в системе

**sudo fdisk -l /dev/**sdx – просмотр информации о конкретном диске

**sudo fdisk /dev/**sdx – перейти в оснастку редактирования диска

* **m** -краткая справка
* **p** - текущий список дисковых разделов с указанием их типа и размер
* **o** – создать MBR таблицу разделов
* **n** – создать новый раздел
  + При назначении last sector можем использовать **+**[количество]**G** что указать сколько гигабайт будет раздел
* **d** – удалить раздел
* **w** – записать изменения
* **q** – выход

**sudo parted /dev/**sdx – запуск parted в интерактивном режиме

* **help** – посмотреть список команд
* **select /dev/**sdx – выбрать другой диск
* **print** – получить информацию о разделах диска
* **mklabel msdos|gtp** – создать таблицу разделов (msdos - MBR)
* **mkpart [part-type fs-type name start end]** – создать раздел
  + **part-type** - primary|extended|logical указывается для MBR
  + **fa-type** - файломая система, см [Форматирование разделов](https://wiki.a1qa.com/pages/viewpage.action?pageId=125994603)
  + **name** - имя раздела, указывается для GPT
  + **start** - начало раздела, может указываться в [unit](http://www.gnu.org/software/parted/manual/parted.html" \l "unit) или в %
  + **end** - конец раздела, может указываться в [unit](http://www.gnu.org/software/parted/manual/parted.html" \l "unit) или в %

Если не задавать параметры part-type, fs-type, name, start, end, то они будут запрошены

* **quit** - выход

**Форматирование разделов**

**mkfs -t *<fstype> <device>***

Где <fstype>:

* reiserfs
* ext3
* ext4
* jfs
* xfs
* vfat
* ntfs
* и т.д.

**Volume Manager**

**lvm** – запуск Linux Volume Manager

* **help** – вывести список всех команд
* **pvcreate /dev/**sdx ; **pvcreate /dev**/sdx – указать физические устройства, на которых создаем группу томов
* **pvremove /dev/sd**x – удалить LVM labels
* **pvdisplay /dev/sd**x – показать атрибуты физических дисков
* **vgcreate** vg\_name **/dev/sd**x **/dev/sd**x2– создать группу томов
* **vgremove** vg\_name– удалить группу томов
* **vgdisplay** vg\_name– показать информацию по группе томов
* **vgextend** vg\_name– добавить физический диск в группу томов
* **lvcreate** -**n** name-**L** disk\_sizevg\_name – создать логический диск
  + -**L [K, M, G]** – Kilobytes, Megobytes, Gigobytes
  + **-l <N>%FREE** – позволяет указать объем в процентах
* **lvremove** name– удалить логический том
* **lvdisplay** name– показать информацию о логическом диске
* **lvextend** [params] – расширить логический диск

**Монтирование разделов**

**mount *<device****>* ***<directory>*** - Если написать просто **mount**, то увидим все смонтированные диски

* **-a** - cмонтировать все файловые системы, перечисленные в файле /etc/fstab. Исключение составляют помеченные как noauto, перечисленные после опции -t и уже смонтированные
* **-f** - монтировать поврежденный раздел (опасно!), или форсировать отмену всех запросов на запись при изменении режима монтирования с "чтение-запись" на "только чтение"
* **-r** - монтировать файловую систему в режиме "только для чтения". То же самое, что и указание аргумента ro для опции -o
* **-v** - выдавать более подробную информацию
* **-w** - монтировать файловую систему в режиме "чтение-запись". То же самое, что и указание аргумента rw для опции -o

**umount *<device> <derectory>*** *-* демонтировать

**Процессор**

**lspci** – показать hardware

**lscpu** – показать процессор

**uname** – выводит информацию о системе

* **–a** – вывести всю информацию

**Помощь по командам**

**echo** $PATH – показать директории в которых Linux будет искать команды/программы

**echo** $SHELL – показать директорию откуда берётся интерпретатор shell

**man** (manual) название\_команды – узнать все про эту команду

* **-k** (keywords) ключевое\_слово – узнать все команды где есть это слово (время, Копировать, настроить и тд)

любая\_команда --help – ключ help выводит список всех доступных ключей для этой команды

**info** название\_команды – узнать все про эту команду

**Выключение ПК**

**uptime** – выводит время, прошедшее с последней перезагрузки, число подключенных пользователей и среднюю загрузку

**shutdown** – выключить ПК

**shutdown now** – выключить сейчас

**shutdown** -c – отменить выключение

**shutdown** +5 – выключить через 5 минут

**shutdown** 12:12 – выключить в заданное время

**shutdown** -r – перезагрузить

**Терминал**

**clear** – очистка окна консоли

**reset** – сброс консоли

**history** – вывод истории вводимых команд пользователем

**ГОРЯЧИЕ КЛАВИШИ**

Alt Space - активировать меню окна

Alt F2 - командная строка

Alt F4 - закрыть окно

Alt F7 - переместить окно

Alt F8 - изменить размер окна

Ctrl Alt F1-F6 - вызов tty

Alt Tab - переключение между окнами

Ctrl Alt T - открыть терминал

Ctrl C - отменить программу

Ctrl D - выйти из программы

Ctrl Z - приостановить программу

*В ТЕРМИНАЛЕ*

Ctrl B(backwards) - переместиться на 1 символ назад

Ctrl F(forward) - на 1 символ вперёд

Alt B - переместиться на 1 слово назад

Alt F - переместиться на 1 слово вперёд

Ctrl A - переместиться в начало строки

Ctrl E - переместиться в конец строки

Ctrl W - удаление предыдущего слова

Alt D - удаление слова впереди

Ctrl H - удаление предыдущего символа

Ctrl D - удаление следующего символа

Ctrl U - удаление всей строки

Ctrl K - удаление всей строки когда вначале строки

Ctrl T - поменять буквы местами

Alt T - поменять слова местами

Ctrl Shift T - открыть новую вкладку

Ctrl D - закрыть вкладку

Ctrl Shift C – копировать

Ctrl Shift V - вставить

*ТЕКСТОВЫЙ РЕДАКТОР*

Alt U – отменить помледнее действие

Alt E – повторить последнее действие