LX1 Основы командной строки

Лекция: Основы командной строки Linux

Ссылка и интернет ресурс, в котором есть видео и презентация

<https://koroteev.site/os/1/1-cli/>

В данной лекции рассматриваются основные приемы работы с командной строкой Linux. Вы поймете, зачем нужно пользоваться командной строкой, какова структура команды, как находить информацию по командам.

Практика: 10 основных команд

В этой работе рассматриваются 10 самых частых команд оболочки bash. Вы изучите самые полезные команды работы с файлами и папками. Эти команды обычно составляют до 90% любой работы сисадмина.

Видео занятия

#### Цель работы Познакомиться с основными приемами работы в командной строке Linux, основные приемы адресации файлов и освоить основные и самые распространенные команды - \_pwd, cd, ls, touch, rm, mkdir, cp, mv, cat, man\_ #### Основное задание 1. Запустите виртуальную машину; познакомьтесь со справкой к изучаемым в данной работе командам. Обратите внимание на используемые ключи данных команд. 2. Запустите текстовый терминал или программу-эмулятор терминала; 3. Выведите на экран содержимое корневой директории файловой системы; 4. Перейдите в свою домашнюю директорию, задав относительный путь; 5. Выведите содержимое файла /etc/passwd; 6. Создайте текстовый файл с именем user.txt в домашнем каталоге. 7. Создайте директорию test в текущей. 8. Переместите только что созданный файл в нее. 9. Скопируйте директорию под именем test2. 10. Удалите обе директории. #### Методические указания Для успешного выполнения работы необходимо ознакомиться со справочными материалами по командам \_pwd, cd, ls, touch, rm, mkdir, cp, mv, cat\_. Сделать это можно используя универсальную справочную команду man. Таким образом, набрав в командной строке: bash$manpwdbash$manpwd мы получим полную и актуальную справку (мануал) по команде pwd, включая все возможные ее ключи и опции. Команда man может показать мануал по любой команде bash. Некоторые команды завершаются моментально (\_pwd\_), а другие (\_ping\_) - выполняются продолжительное время. Кроме того, некоторые команды запускают интерактивные программы (\_top\_). Для успешной и быстрой работы в командной строке существует большое количество сочетаний клавиш. Ознакомьтесь с ними на практике и попробуйте запомнить наиболее полезные: \* <Ctrl>+<C> - завершение текущей команды; \* <Ctrl>+<D> - выход из текущего сеанса (разлогин); \* <Tab> - подсказки и множественное дополнение; \* \_clear, <Ctrl> + <L>\_ - очистка экрана (перемотка вперед); \* <↑> - переход к предыдущей команде; \* <Ctrl> + <A> - переход к началу команды; \* <Ctrl> + <E> - переход к концу команды; \* <Alt> + <F>, <Alt> + <B> - переход к следующему и предыдущему слову в команде; \* history - вывод истории команд в текущем сеансе; \* <Ctrl> + <R> - поиск команды в истории; В Linux, как и во многих других Unix-подобных системах принята единая система адресации файлов. Все файлы в Linux имеют имя и организуются в каталоги, образуя иерархическое дерево папок. В данном контексте, мы будем использовать понятия “каталог”, “папка” и “директория” как синонимы. Для того, чтобы идентифицировать файл, вы должны указать путь к нему, то есть порядок вложенности папок, позволяющий найти этот файл. Все пути в Linux отсчитываются от корня файловой системы, обозначаемого прямым слешем /. Корень всегда один, не существует никаких букв дисков, как в Windows. Корень еще называют корневой папкой или директорией, так как именно в нем содержаться все другие папки. В пути к файлу последовательно указываются папки, которые нужно пройти, чтобы найти файл, а затем - имя самого файла. Имена папок и файла разделяются прямым слешем. Так, в пути /etc/passwd первый символ / обозначает корневой каталог, etc - имя папки в корневом каталоге, passwd - имя нужного нам файла. Обратите внимание, что в Linux не принято использовать расширения в именах файлов (но, впрочем, и не запрещено, так что ими многие пользуются). Находясь в командной строке, вы всегда находитесь в какой-то директории. По умолчанию, это домашний каталог пользователя (обычно находящийся по адресу /home/<login>). Текущий каталог можно изменить командой cd. Путь к файлу можно задавать двумя способами - абсолютным или относительным. Абсолютный путь начинается с символа / и отсчитывается от корневого каталога. Относительный путь начинается с имени папки и отсчитывается от текущего каталога. Это сделано для удобства указания пути, чтобы не повторять одно и то же, если вы работаете в данный момент преимущественно с файлами в одной папке. При указании путей к файлам можно пользоваться некоторыми сокращениями. Вы можете использовать два специальных имени: точку (.), означающую текущую директорию, и пару точек (..), означающую родительскую директорию текущей директории. Также вы можете использовать символ тильды (~), который означает вашу домашнюю директорию, и сочетание ~username, означающее домашнюю директорию пользователя с именем username. Помните, что нужно четко представлять себе, где лежит тот или иной файл для того, чтобы с ним работать. Часто /etc/passwd и etc/passwd - это два совершенно разных файла в разных местах. #### Контрольные вопросы 1. Как происходит авторизация пользователя в текстовом режиме? 2. Как посмотреть рабочую директорию? 3. Как посмотреть содержимое текущей директории? 4. Чем отличается абсолютный и относительные пути к директориям? 5. Как перейти в другую директорию в командной строке? 6. Какие основные стандартные каталоги существуют в Linux? #### Задания для самостоятельного выполнения 1. Выяснить, какой каталог является текущим (см. man pwd). 2. Выяснить, в каком терминале выполняется текущий сеанс. 3. Перейдите в директорию /var; 4. С помощью команды узнайте, в какой директории находимся; 5. Прочитайте мануал по команде ls (только основное положение и про флаги -l, -t, -r, -A); 6. Отобразите список директорий и файлов; 7. Перейдите в директорию /var/log (без упоминания “var” в команде); 8. Перейдите в директорию /var (снова без упоминания “var”); 9. Создайте директорию ~/tmp/lesson/; 10. Перейдите туда, задав абсолютный путь; 11. Создайте текстовый файл txt\_file с произвольным содержимым; 12. Удалите созданную папку lesson; 13. Сохраните подробное описание полного содержимого домашней директории (по команде ls -la) в файл home.dir. 14. (\*) Определить границы файлового пространства, где система позволяет создавать собственные файлы и каталоги (возможно использование автоматического скрипта). 15. (\*) Проверить, возможно ли вмешательство в личное файловое пространство другого пользователя? Пользователя root?

Практика: Служебные команды

Видео занятия

#### Цель работы Научиться основным навыкам работы в многопользовательском режиме, переключению между терминалами и самым базовым командам - \_tty, who, whoami, sudo, su, less, who, exit, shutdown, grep\_. #### Основное задание 1. Откройте первый виртуальный терминал и залогиньтесь. 2. Выполните команду tty. 3. Выполните команду pwd. 4. Повторите предыдущие пункты для второго и третьего виртуального терминала. 5. Откройте эмулятор терминала в графическом режиме и введите команду tty. 6. Выполните команду who. Посмотрите в справке, что она показывает. 7. Выполните команду whoami. Прочитайте справку про нее. 8. Перейдите в первый виртуальный терминал и разлогиньтесь. 9. Вернитесь в графическое окружение и еще раз просмотрите вывод команды who. Объясните ее результат. 10. Откройте файл /etc/passwd при помощи программы-пейджера. 11. Найдите в этом файле запись о вашем текущем пользователе при помощи команды grep. 12. Изучите команду man и прочитайте справку по всем командам, использованным в данной лабораторной. #### Методические указания Операционная система Linux, как и ее предшественник - UNIX изначально создавалась для многопользовательской работы. Для организации многопользовательской работы при загрузки системы создается семь виртуальных устройств (это число может меняться в зависимости от дистрибутива, но семь - это стандартное количество). Раньше предполагалось, что за ними будут работать из разных мест при помощи оконечных устройств (терминалов или тонких клиентов), но сейчас это уже не очень актуально. Обычно на последнем, седьмом терминале запускается графическая оболочка, если она установлена в данном дистрибутиве. Для того, чтобы узнать, номер текущего терминала, можно использовать команду tty (см. man tty). Команда очень проста и не требует параметров. Пример работы tty: `bash tty/dev/tty1‘‘‘Чтобыузнать,какиепользователивкакихтерминалахзалогиненывсистеме,можноиспользоватькомандуwho:‘‘‘bashtty/dev/tty1‘‘‘Чтобыузнать,какиепользователивкакихтерминалахзалогиненывсистеме,можноиспользоватькомандуwho:‘‘‘bash who root tty1 Feb 17 17:11 user tty2 Feb 17 17:12 user tty5 Feb 17 17:40 Иногдавозникаетнеобходимостьвыполнитьнекоторыедействияотименидругогопользователя(например,отимениroot).Этоможносделать,открывновыйтерминальныйсеанс,номожноииначе,втекущемсеансе.Дляэтогоиспользуетсякомандаsu.Этакомандапоумолчанию(безпараметров)открываетсеанссуперпользователя,выполняющийсявнутрисеансанепривилегированногопользователя.Примервыполнениякомандыsu:Иногдавозникаетнеобходимостьвыполнитьнекоторыедействияотименидругогопользователя(например,отимениroot).Этоможносделать,открывновыйтерминальныйсеанс,номожноииначе,втекущемсеансе.Дляэтогоиспользуетсякомандаsu.Этакомандапоумолчанию(безпараметров)открываетсеанссуперпользователя,выполняющийсявнутрисеансанепривилегированногопользователя.Примервыполнениякомандыsu:bash su   
Password:   
[root@localhost user]#  
```  
  
Еще один пример, с явным указанием имени пользователя:  
  
  
```bashsu Password: [root@localhost user]#```Еще один пример, с явным указанием имени пользователя:```bash su stud Password: [stud@localhost ~]‘‘‘Дляокончанияпользовательскогосеанса,запущенногокомандойsu,иливыходаизтерминала,используетсякомандаexitиликомбинацияклавишCtrl+D.Влюбомслучаесистемазакроетсеансиотобразитстрокуприглашения.Оченьчастоприработескоманднойстрокойприходитсяпросматриватьдовольнодлинныефайлы.Длятого,чтобыможнобылоперемещатьсяпофайлу,содержимоекоторогонепомещаетсянаодномэкранебылисозданыспециальныепрограммы−пейджеры.Этонетекстовыйредактор,атолькосредствопросмотра.Самыйстандартныйнасегодня−программаless.Ейможнопередатьимяфайлатакимобразом:‘‘‘bash‘‘‘Дляокончанияпользовательскогосеанса,запущенногокомандойsu,иливыходаизтерминала,используетсякомандаexitиликомбинацияклавишCtrl+D.Влюбомслучаесистемазакроетсеансиотобразитстрокуприглашения.Оченьчастоприработескоманднойстрокойприходитсяпросматриватьдовольнодлинныефайлы.Длятого,чтобыможнобылоперемещатьсяпофайлу,содержимоекоторогонепомещаетсянаодномэкранебылисозданыспециальныепрограммы−пейджеры.Этонетекстовыйредактор,атолькосредствопросмотра.Самыйстандартныйнасегодня−программаless.Ейможнопередатьимяфайлатакимобразом:‘‘‘bash less filename Познакомьтесьсней,прочитавинтерактивнуюсправку.Длятого,чтобызакрытьеенеобходимонажатькнопкуq.Такжечастоприходитсяискатьопределеннуюстрокувфайлеиливвыводедругихкоманд.Дляэтогоиспользуютутилитуgrep.Ейпередаётсядвапараметра−строкапоискаиимяфайлавоттак:Познакомьтесьсней,прочитавинтерактивнуюсправку.Длятого,чтобызакрытьеенеобходимонажатькнопкуq.Такжечастоприходитсяискатьопределеннуюстрокувфайлеиливвыводедругихкоманд.Дляэтогоиспользуютутилитуgrep.Ейпередаётсядвапараметра-строкапоискаиимяфайлавоттак:bash grep koroteev /etc/passwd ` Дополнительная информация об указанных командах доступна в справочном руководстве формата man или info. Еще одно важное замечание. При работе с эмулятором терминала в графическом режиме, копирование и вставка работают с комбинациями клавиш Ctrl + Shift+C и Ctrl + Shift+V. Это потому, что комбинацией клавиш Ctrl + C аварийно завершается (прерывается) выполнение текущей команды. #### Контрольные вопросы 1. Какой каталог будет установлен текущим сразу же после входа пользователя в систему? 2. Какой каталог будет установлен текущим после выполнения команды su? 3. Какой терминал (tty) будет открыт по нажатию Ctrl+F7? 4. Как и где в Linux можно вводить команды командной строки? #### Задания для самостоятельного выполнения 1. Подключитесь к данной машине через ssh и введите команду who. Объясните, что значит ее результат. 2. Загрузить систему в текстовом режиме и войти с собственной учетной записью 3. Ознакомиться со справкой по команде shutdown; 4. С её помощью перезагрузить виртуальную машину; 5. С её помощью выключить виртуальную машину; 6. Запустить отложенное выключение виртуальной машины на 2 минуты.

Лекция: Работа с файлами[Permalink](https://koroteev.site/os/1/1-cli/#%D0%9B%D0%B5%D0%BA%D1%86%D0%B8%D1%8F:%20%D0%A0%D0%B0%D0%B1%D0%BE%D1%82%D0%B0%20%D1%81%20%D1%84%D0%B0%D0%B9%D0%BB%D0%B0%D0%BC%D0%B8)

Презентация[Permalink](https://koroteev.site/os/1/1-cli/#4-slides)

Смотреть [слайды](https://koroteev.site/pres/os11-3/) к этому занятию.

Практика: Работа с файлами[Permalink](https://koroteev.site/os/1/1-cli/#%D0%9F%D1%80%D0%B0%D0%BA%D1%82%D0%B8%D0%BA%D0%B0:%20%D0%A0%D0%B0%D0%B1%D0%BE%D1%82%D0%B0%20%D1%81%20%D1%84%D0%B0%D0%B9%D0%BB%D0%B0%D0%BC%D0%B8)

В этой работе вы познакомитесь с полезными приемами работы с файлами - созданием жестких и символических ссылок, командами архивации файлов.

#### Цель работы Ознакомиться со структурой файловой системы Линукс, типами файлов и командами управления файловой системой. #### Основное задание 1. В домашнем каталоге создать подкаталоги src, dst и temp; 2. В каталоге temp создать файлы user.txt, root. txt и stud.txt произвольного содержания; 3. В каталог src скопировать файлы user.txt, root. txt и stud.txt, различного содержания; 4. В каталоге dst создать «жесткие» ссылки на все файлы из каталога src; 5. В домашнем каталоге создать «мягкие» ссылки на файлы из каталога src; 6. Вывести рекурсивно расширенную информацию о содержимом домашнего каталога. Обратить внимание на поле размера для физических файлов и ссылок; 7. Из домашнего каталога выполнить команды: bashcatsrcuser.txtcatdstuser.txtcatuser.txtbashcatsrcuser.txtcatdstuser.txtcatuser.txt 8. Заархивируйте содержимое папки src/ в архив .tar.gz. 9. Распакуйте этот архив в директорию \~/backup 10. Выведите названия всех файлов домашней директории, имеющих в названии .txt (подсказка: используйте команду find); 11. Удалите файл user.txt. Что произошло со ссылками на него? 12. Удалите каталог dst; 13. Переместите каталог temp в src; 14. Рекурсивно удалите каталог src #### Методические указания ##### Организация файловой системы Файловая система ОС Линукс (как и прочих unix-подобных систем) устроена так, что все ресурсы представлены единообразно, в виде файлов. Такой подход позволяет обеспечить универсальный интерфейс доступа к любым ресурсам: от физических устройств, до процессов, выполняющихся в системе. С точки зрения пользователя файловая система представляет логическую структуру каталогов и файлов. С другой стороны, невидимой пользователю, внутреннее устройство файловой системы реализует физические операции чтения/записи файлов на различные носители, алгоритмы доступа и многое другое. ##### Типы файлов Для обеспечения единообразного доступа к файлам их прежде всего необходимо классифицировать. В Линукс это сделано следующим образом: \* обычные (regular) файлы - текстовые, исполняемые, графические и пр. файлы, создаваемые пользователями и прикладными программами; \* каталоги (directories) - именованные группы файлов и вложенных каталогов (т.е. содержимое каталога - файлы и другие каталоги); \* файлы устройств (devices) - соответствуют присутствующим в системе реальным (жесткие диски, принтеры, мыши, ЦП и т.д.) устройствам и т. н. псевдоустройствам (например, /dev/null). Файлы устройств представляют символьные (последовательного доступа) и блочные (произвольного доступа) устройства. К первыми относятся, например, параллельные и последовательные порты, ко вторым - жесткие диски; \* специальные файлы - сокеты (sockets) и именованные каналы (named pipes), которые предназначены для обмена информацией между процессами; \* символьные ссылки (symlinks) - именованные указатели на физические файлы (аналог ярлыков ОС Windows), содержащие только путь к некоторому файлу. Символьные ссылки могут указывать на файлы, хранящиеся как локальных, так и в сетевых каталогах. \* Символьные ссылки (или "мягкие") не нужно путать с "жесткими", которые указывают на inode файла. Inode (идентификатор узла) - это уникальный числовой идентификатор узла (файла или каталога) файловой системы, по которому и осуществляется доступ к нему. Символьное имя файла (включая полный путь) ориентировано на пользовательское восприятие. Для человека-оператора проще запомнить осмысленные имена файлов (например: report.txt, myfoto.jpg и т.п.), чем абстрактные числовые значения. Прочие отличия "жестких" и "мягких" ссылок вам предстоит выяснить в ходе выполнения этой лабораторной работы. ##### Каталоги Линукс Все файлы упорядочены по каталогам. Структура и назначение каждого из каталогов, созданных на этапе установке предопределены, хотя и могут быть (что не стоит делать без крайней необходимости) изменены суперпользователем. Файловая система имеет иерархическую структуру и начинается от корневого каталога (/). Его подкаталогами являются: \* /bin - исполняемые файлы общего назначения; \* /boot - содержит образ загружаемого ядра; \* /dev - файлы устройств; \* /etc - конфигурационные файлы общего пользования; \* /home - домашние каталоги пользователей, включая программы и файлы личных предпочтений; \* /lib - общесистемные библиотеки; \* /mnt - каталог монтирования внешних файловых систем; \* /proc - виртуальная файловая система для чтения информации о процессах; \* /root - домашний каталог суперпользователя; \* /sbin - программы системного администрирования; \* /tmp - каталог для хранения временной информации; \* /usr - каталог пользовательских прикладных программ со всеми их исполнимыми и конфигурационными файлами. Например, в подкаталог /usr/local инсталлируются программы, не входящие в дистрибутив Линукс, или собираемые из исходных текстов. \* /var - каталог для хранения часто изменяющихся файлов. Например, спулера печати, различных лог-файлов, почтовых сообщений и т.п. \* /lost+found - каталог для нарушенных фрагментов файлов, обнаруженных в результате проверки файловой системы после сбоя. Такая структура типична для большинства дистрибутивов Линукс, но могут иметься и дополнительные каталоги. Например, /opt - для дополнительных компонентов, /selinux - расширение системы безопасности и т.п. ##### Именование файлов и каталогов Файловая система Линукс поддерживает "длинные" имена, содержащие символы латиницы, национальных алфавитов, знаки пунктуации и спецсимволы. Абсолютно запрещенными к использованию в имени являются прямой и обратный слэши (/ и \). Максимальное количество символов в имени - 255. Понятие "расширения файла" в unix-системах отсутствует как таковое, поэтому в имени может быть несколько частей, разделенных точками. Все имена - регистрозависимые. Приведенные выше правила справедливы и для каталогов. Файлы и каталоги, названия которых начинаются с точки (т.н. dot-файлы), являются аналогами "скрытых" файлов Windows. То есть по умолчанию они не отображаются при просмотре содержимого файловой системы. Для быстрого доступа к файлам в оболочке имеются несколько переменных окружения, хранящих соответствующие пути. Это, например, переменная HOME,вкоторойсодержитсяпутикдомашнемукаталогутекущегопользователя.Т.е.действиякоманд‘‘‘bashHOME,вкоторойсодержитсяпутикдомашнемукаталогутекущегопользователя.Т.е.действиякоманд‘‘‘bash cd /home/usr1  и `bash cdcdHOME ` приведут к одному результату - переходу в домашний каталог пользователя usr1. Более того, в оболочке определен псевдоним для домашнего каталога - символ ~ (тильда) можно использовать аналогично HOME.Например:‘‘‘bashHOME.Например:‘‘‘bash cd ~ pwd   
/home/usr1   
```  
  
Некоторые другие возможности оболочки будут рассмотрены в следующих лабораторных работах. Дополнительная информация доступна в справочном руководстве bash (man bash).  
  
  
##### Команды управления файловой системой  
  
Для управления файловой системой имеются различные команды, реализующие операции по созданию, чтению, копированию, переименованию/перемещению, изменению и удалению файлов и каталогов. Как правило, это специализированные команды, хорошо выполняющие свою задачу, однако некоторые функции могут частично дублироваться другими командами, что только добавляет гибкости управлению файлами.  
  
Основными командами для выполнения файловых операций являются: pwd, ls, cp, mv, dir, rm, cd, rmdir, mkdir, ln. Информацию о их назначении и параметрах доступна в формате man и info.  
  
Несколько слов о способах создания файлов в Линукс:  
  
  
  
\* во-первых, файлы могут быть созданы как результаты работы прикладных программ и иметь определенный формат (например, графические файлы, созданные редактором GIMP);  
\* во-вторых, файлы могут быть созданы пользователем путем ввода информации с клавиатуры, например так:  
  
  
```bashpwd /home/usr1 ```Некоторые другие возможности оболочки будут рассмотрены в следующих лабораторных работах. Дополнительная информация доступна в справочном руководстве bash (man bash).##### Команды управления файловой системойДля управления файловой системой имеются различные команды, реализующие операции по созданию, чтению, копированию, переименованию/перемещению, изменению и удалению файлов и каталогов. Как правило, это специализированные команды, хорошо выполняющие свою задачу, однако некоторые функции могут частично дублироваться другими командами, что только добавляет гибкости управлению файлами.Основными командами для выполнения файловых операций являются: pwd, ls, cp, mv, dir, rm, cd, rmdir, mkdir, ln. Информацию о их назначении и параметрах доступна в формате man и info.Несколько слов о способах создания файлов в Линукс:\* во-первых, файлы могут быть созданы как результаты работы прикладных программ и иметь определенный формат (например, графические файлы, созданные редактором GIMP);\* во-вторых, файлы могут быть созданы пользователем путем ввода информации с клавиатуры, например так:```bash cat > f1 Hello, world! // нажатие Ctrl+D завершает ввод команд aag@stilo:~> ⋅в−третьих,файлымогутбытьсозданыпутемперенаправлениявыводакомандысостандартногопотока,напримертак:⋅в-третьих,файлымогутбытьсозданыпутемперенаправлениявыводакомандысостандартногопотока,напримертак:bash # echo "Hello, World!" > f1 ` В первом и втором случаях символ ">" - это команда перенаправления стандартных потоков ввода/вывода, встроенная в оболочку. В первом случае она получает информацию со стандартного потока ввода (клавиатура) и, по окончании ввода (Ctrl+D), отправляет ее в файл. Во втором - принимает строку, переданную командой echo и также отправляет ее в файл. Если файл отсутствует, то он будет создан, если имеется, то будет перезаписан. Для добавления информации в файл следует использовать команду ">>". Больше информации о командах перенаправления можно получить в справке о возможностях оболочки. #### Контрольные вопросы 1. Чем отличаются жесткие и символические ссылки? 2. Что произойдет с жесткой ссылкой, если удалить исходный файл? А с символической? 3. Совпадает ли размер символьных и жестких ссылок? 4. Пусть в системе имеется каталог "\~/test". Что произойдет при попытке выполнить следующую команду: "cp \~/test \~/test/newdir"? 5. Какие основные форматы архивов принято использовать в Linux? Какими командами можно пользоваться для создания / открытия этих архивов? #### Задания для самостоятельного выполнения 1. Настройте автоматическую инкрементальную архивацию домашнего каталога. Для этого создайте команду архивирования всех файлов, измененных за последние сутки и добавьте ее на автоматическое выполнение при помощи службы cron. 2. Измените программу из предыдущего пункта так, чтобы пользователь мог опциями изменять настройки программы: 1. Задавать папку, откуда происходит копирование; 2. Задавать расположение архивов; 3. Задавать частоту архивации; 4. Исключать определенные типы файлов; 5. (\*) Передавать список папок и типов файлов, которые включаются в общий архив.