

Внешний курс №2

Введение в Linux

Луангсуваннавонг Сайпхачан

15 мая 2025

Российский университет дружбы народов, Москва, Россия

Информация

- Луангсуваннавонг Сайпхачан
- Студент из группы НКАбд-01-24
- Российский университет дружбы народов
- <https://sayprachanh-lsvnv.github.io>



Этап 2 внешнего курса (Работа на сервере)

2.1 Знакомство с сервером 6 out of 6 steps passed 2 out of 2 points received

Вся вторая неделя курса посвящена **работе на удаленных серверах**. Кроме того, мы хотим обратить ваше внимание на то, что на этой неделе будет показан ряд примеров с использованием биоинформатических программ. Однако для выполнения этих заданий не нужно обладать никакими специальными знаниями или навыками в области биоинформатики или биологии, так что не стоит волноваться!

На первом занятии этой недели мы обсудим, что такое сервер и зачем на нем работать. Кроме того, мы рассмотрим два способа войти на удаленный сервер с помощью терминала.

Показанные на занятии слайды доступны по [ссылке](#).

 1254  140

Step 1

Next step >



Рис. 1: 2.1

2.2 Обмен файлами 8 out of 8 steps passed 3 out of 3 points received

Для полноценной работы на удаленном сервере нужно уметь не только заходить на него с компьютера-клиента, но и обмениваться файлами как в одну, так и в другую сторону. На этом занятии мы рассмотрим два способа обмена файлами: с использованием терминала и при помощи графического приложения [Filezilla](#).

Кроме того, на этом занятии будет рассмотрен новый способ установки программ на Linux. Этот подход использует терминал и не требует графического приложения, поэтому он может быть успешно применен как на своем локальном компьютере, так и на удаленном сервере.

Показанные на занятии слайды доступны по [ссылке](#).

 1142  210


Step 1

Next step >

Рис. 2: 2.2

2.3 Запуск приложений 8 out of 8 steps passed 7 out of 7 points received

Для полноценной работы на сервере нужно уметь не только зайти на него и обменяться с ним файлами, но и запускать приложения на нем. На этом занятии мы рассмотрим несколько примеров таких запусков. С приложениями, работе которых необходим только терминал, особых сложностей у нас не возникнет. Однако мы также рассмотрим и запуск программ, которые обычно требуют графический вывод (на примере программы [FastQC](#)).

 951  719

Step 1



Next step >

Рис. 3: 2.3

2.4 Контроль запускаемых программ 11 out of 11 steps passed 4 out of 4 points received

Обычно на сервере могут одновременно работать очень много пользователей. Кроме того, каждый из них может загружать сервер множеством задач. Чтобы не мешать друг другу, каждый пользователь должен уметь контролировать все запущенные им программы. На этом занятии мы научимся следить за своими (и чужими) приложениями, запускать их в фоновом и основном режимах, и, наконец, отменять некоторые из запусков в случае необходимости.

Показанные на занятии слайды доступны по [ссылке](#).

 903  64

Step 1

Next step >



Рис. 4: 2.4

2.5 Многопоточные приложения 14 out of 14 steps passed 6 out of 6 points received

На этом занятии мы обсудим понятие потоков и процессов, без которых практически невозможно представить обработку больших объемов данных. Запуск и контроль многопоточного приложения будет рассмотрен на примере программы [bowtie2](#).

Если программы bowtie2 еще нет на вашем компьютере, то её можно установить следующим образом. Скачивайте архив ([версия для 32-разрядного Linux](#) и [версия для 64-разрядного](#)) с исполняемыми файлами программы и распакуйте его соответствующей программой-архиватором. Теперь все нужные файлы (bowtie2-build и bowtie2) лежат в той директории, куда произошла распаковка архива и их можно запустить как и любую другую программу (например, через ./bowtie2-build из директории, где они лежат или из любой другой директории задав абсолютный путь до bowtie2-build, см. [соответствующее занятие](#)).

Показанные на занятии слайды доступны по [ссылке](#).

 882  483

Step 1

Next step >

Рис. 5: 2.5

Менеджер терминалов tmux

2.6 Менеджер терминалов tmux 19 out of 19 steps passed 7 out of 7 points received

На этом занятии мы рассмотрим менеджер терминалов [tmux](#). Это специальная программа, которая может объединять в себе несколько терминалов, каждый из которых содержит свои процессы. Важная особенность tmux заключается в том, что если запустить его на сервере, то он будет помнить историю запусков процессов и они продолжат выполнение даже если соединение с сервером будет потеряно. Кроме того, при новом соединении с сервером можно будет подключиться к уже запущенному tmux и продолжить работу с момента потери связи.

Показанные на занятии слайды доступны по [ссылке](#).

Примечание: для прохождения большинства заданий вам может понадобиться программа tmux на своем компьютере. Если её у вас нет, то её можно установить командой `sudo apt-get install tmux` (на Ubuntu) или найдя её в Software Center по запросу `tmux`.

Внимание! В этом задании вам понадобится вводить tmux-команды. Все они построены на нажатии клавиш Ctrl+B и вводе еще одного символа с клавиатуры. Для правильного ввода команд следует учитывать два важных момента:

1. **Сперва нужно зажать** (одновременно) клавиши **Ctrl и B**, **затем отпустить** их и только **потом** ввести "командный" символ. Часть "командных" символов представляют собой буквы, поэтому для их ввода достаточно нажать соответствующую клавишу на клавиатуре. Другая часть символов – знаки, поэтому для их ввода может потребоваться два нажатия: на Shift и на нужную клавишу. Например, для ввода символа "%" нужно нажать Shift и клавишу 5 (конечно, если на вашей клавиатуре % находится на клавише 5). Важно отметить, что знаки одинарных кавычек (') и двойных (" , т.е. Shift+') имеют разное значение!
2. Обращайте внимание на включенную у вас **раскладку** клавиатуры. Все буквенные командные символы нужно вводить в английской раскладке, а знаковые можно вводить в любой. Кроме того, в любой раскладке можно нажимать и ключевые клавиши Ctrl+B, например, в русской раскладке это будет Ctrl+И. Однако во избежание путаницы мы рекомендуем вводить все команды в **английской** раскладке.

Как установить Linux: расширенное руководство

2.7 Как установить Linux: расширенное руководство 6 out of 6 steps passed

Это занятие предназначено для тех, кто устанавливал Linux на свой компьютер при помощи VirtualBox (см. [соответствующее занятие](#) на первой неделе курса). Кроме того, вы могли попасть сюда прочитав [важное сообщение об устаревании курса](#) и исполнившись желанием установить самую последнюю версию Linux и VirtualBox с самого начала обучения.

На этом занятии вы узнаете:

- как полноценно установить Linux на ваш компьютер (т.е. без виртуальных машин),
- как использовать виртуальную машину, но поставить туда более новый Linux, чем предлагается на первой неделе курса
- как настроить VirtualBox для более удобной работы (растягивать окошко виртуальной машины на весь экран, обмениваться файлами между основной системой и Linux, копировать текст между системами).

Если озвученные темы вам неинтересны или вы их уже итак знаете, то можете смело пропускать занятие – оно необязательное и баллов за него не полагается!

Формат этого занятия следующий. Сначала идет видео-шаг по соответствующей теме, а затем текстовый-шаг со всеми использованными в видео ссылками и командами терминала. Шагов с вопросами в этом занятии не предусмотрено.

Примечание: видео снималось в 2014 году и его основной целью было научить установке Linux на компьютер без использования виртуальной машины. Однако так как в видео, по понятным причинам, виртуальная машина всё-таки используется, то его же можно рассматривать и как наглядную иллюстрацию установки произвольной версии Ubuntu (в частности самой актуальной) в виртуальную машину (которую мы тоже рекомендуем обновить до самой [последней версии](#)).



871



35

Step 1

Next step >

Научился работать с сервером: настройка общего доступа к файлам, запуск приложений, многопоточные приложения и управление ими.

Спасибо за внимание
