

Для того, чтобы понимать, откуда появился Линукс нам необходимо копнуть чуть глубже в историю и вспомнить прообраз этой ОС.

А прообразом Линукса была ГНУ.

Слайд 1

Тут может показаться, что ГНУ - это аббревиатура какого-то очень сложного названия. Однако ГНУ - это банально антилопа. А теперь откатимся еще чуть назад. К системам ЮНИКС.

Системы ЮНИКС появились в 1970-х годах, благодаря компании AT&T

СЛАЙД 2

Это был проект, который отдали на растерзание исследовательскому центру Белл Лабс. И заведовал этой всей вакханалией Кен Томпсон и Дэнис Ричи

СЛАЙД 3

Благодаря их Работе системы юникс приобрели довольно законченный вид и характеризуются модульным дизайном, благодаря которому каждая задача выполняется отдельной утилитой (которая прописана в отдельном модуле). Взаимодействия осуществляются при помощи единой файловой системы, а для работы с утилитами используется командная оболочка.

И раз уж мы заговорили про сленг, давайте я поясню:

Утилита^[1] ([англ. utility](#)) — вспомогательная [компьютерная программа](#) в составе общего [программного обеспечения](#) для выполнения специализированных типовых задач

А командная оболочка - это оболочка, в которой пользователь может либо давать команды операционной системе по отдельности, либо запускать скрипты, состоящие из списка команд. Основа основ множества современных систем. Так что можно с уверенностью сказать, что это был очередной прорыв в сфере.

На этом слайде мы можем наблюдать несколько ОС, которые сейчас получили огромное развитие в разных средах.

Слайд 4 командной оболочки

С тех пор было создано большое количество различных Unix-систем. И среди них как и наш линукс, так и те системы, которые мы никогда в глаза не увидим. [AIX](#), [HP-UX](#), [IRIX](#), [macOS](#), [LynxOS](#), [QNX](#), [SCO OpenServer](#), [Solaris](#), [Tru64 UNIX](#), [UnixWare](#), [Xenix](#) и [VxWorks](#).

На первый взгляд может показаться, что даже Мак Ос является Юникс системой, так как отвечает всем критериям, которые я назвал чуть выше. Но тут скрывается подвох. Мак ОС - ЮНИКС подобная система. И тут проблема в том, что ЮНИКС подобная система - жертва патента. Точнее авторских прав. Потому что называться Юникс системами могут лишь не много) Шутка за 300 от Ивана программиста.

Ладно, продолжим.

Все дело в том, что [The Open Group](#) обладает [торговой маркой](#) UNIX и управляет разработкой стандарта [Single UNIX Specification](#), где слово «UNIX» используется как [знак соответствия](#). Они не приветствуют употребление термина «UNIX-подобный» и считают, что это злоупотребление их товарным знаком. Руководства, изданные группой, требуют использования заглавных букв в названии UNIX либо выделение другим способом от остального текста и не одобряют написание через дефис (относится к английским текстам). Наиболее близкий термин, который они сочли бы корректным, был бы *Unix system-like*^[2].

Говоря проще МАК ОС - ЮНИКС СИСТЕМ лайк. Хоть и основа у нее от ЮНИКС,

Теперь вернемся к нашим антилопам.

СЛАЙД 5 ГНУ

Проект ГНУ - это проект, который был создан Ричардом Столлманом 27 сентября [1983 года](#) и подразумевал «разработать достаточно свободного программного обеспечения <...>, чтобы можно было обойтись без программного обеспечения, которое не является свободным».^[2] Чтобы этого достичь, проект в 1984 году приступил к разработке операционной системы [GNU](#) (GNU еще расшифровывается, как «*GNU is Not Unix*»). Эта цель была достигнута в [1992 году](#), когда последний пробел в ОС GNU — [ядро системы](#) — был заполнен сторонней разработкой, ядром [Linux](#), которое было выпущено как свободное программное обеспечение в соответствии с лицензией GNU [GPL v2](#).

Теперь мы столкнулись с новым термином. Ядро системы. Что же это такое?

Ядро ([англ. kernel](#)) — центральная часть [операционной системы](#) (ОС), обеспечивающая [приложениям](#) координированный доступ к ресурсам [компьютера](#), таким как [процессорное время](#), [память](#), внешнее [аппаратное обеспечение](#), внешнее устройство ввода и вывода информации. Также обычно ядро предоставляет сервисы [файловой системы](#) и [сетевых протоколов](#).

К 1990 году в рамках проекта [GNU](#), основанного [Ричардом Столлманом](#), были разработаны и постоянно развивались [свободные программы](#), составляющие основной [инструментарий для разработки](#) программ на [языке Си](#): текстовый редактор [Emacs](#), [компилятор](#) языка Си [gcc](#), [отладчик](#) программ [gdb](#), [командная оболочка](#) [bash](#), [библиотека](#) важнейших [функций](#) для программ на С. Все эти программы были написаны для [операционных систем](#), похожих на [UNIX](#). Поэтому в них использовались стандартные для UNIX [системные вызовы](#) — [POSIX](#). При помощи системных вызовов программы получают доступ к [оперативной памяти](#), [файловой системе](#), [устройствам ввода и вывода](#). Благодаря тому, что системные вызовы выглядели более-менее стандартно во всех реализациях UNIX, программы GNU могли работать (с минимальными изменениями или вообще без изменений) в любой UNIX-подобной операционной системе.

С помощью имевшихся инструментов GNU можно было бы писать программы на Си, пользуясь только свободными программными продуктами, однако свободного UNIX-совместимого [ядра](#), на основе которого могли бы работать все эти инструменты, не существовало. В такой ситуации разработчики GNU вынуждены были использовать одну из патентованных реализаций UNIX, Мечта Столлмана о научной разработке ПО, свободной от решений, движимых коммерческими целями, была неосуществима, пока в основе свободной разработки лежало патентованное UNIX-совместимое ядро, [исходные тексты](#) которого оставались тайной для разработчиков.

Теперь давайте поговорим непосредственно о линуксе.

Слайд 6 Торвальдса

В 1991 году во время обучения в [Хельсинкском университете](#) [Линус Торвальдс](#) заинтересовался операционными системами и был разочарован лицензией [MINIX](#), которая ограничивала её использование только образовательными целями (что исключало любое коммерческое использование), вследствие чего начал работать над своей собственной операционной системой, которая в итоге стала [Linux](#).

Торвальдс начал разработку ядра [Linux](#) на [MINIX](#), и перенёс на него ряд приложений. Позже, когда [Linux](#) достиг определённой зрелости, появилась возможность продолжать разработку уже на базе самого [Linux](#)^[19]. Приложения GNU вскоре заменили приложения [MINIX](#), так как [код](#) GNU, находящийся в свободном доступе, был более удобен для применения в молодой операционной системе (исходный код под лицензией [GNU GPL](#) может быть использован в других проектах, если они также выпускаются под той же или совместимой лицензией, для того чтобы сделать [Linux](#) доступным для коммерческого использования, Торвальдс начал переходить от своей первоначальной лицензии на [GNU GPL](#))^[20]. Разработчики работали над полной интеграцией компонентов GNU с [Linux](#) с целью создания полнофункциональной и свободной операционной системы ([Linux](#)).

Слайд Торвальдса

Так появилась Система Линукс

Слайд 7

И важно запомнить, что

Linux-системы реализуются на модульных принципах, стандартах и соглашениях, заложенных в Unix в течение 1970-х и 1980-х годов. Такая система использует монолитное ядро, которое управляет процессами, сетевыми функциями, периферией и доступом к файловой системе. Драйверы устройств либо интегрированы непосредственно в ядро, либо добавлены в виде модулей, загружаемых во время работы системы.

Слайд 8

В отличие от коммерческих систем, таких как Windows или macOS, Linux не имеет географического центра разработки. Нет и организации, которая владела бы этой системой. Linux — результат работы тысяч проектов. Некоторые из этих проектов централизованы^{[27][28]}, некоторые сосредоточены в фирмах.

Помните, я вчера говорил, что в Линуксе нужно постоянно обновлять дистрибутив через терминал? Кстати, вот он, на экранах. Терминал - это подобие Повер шелл в Виндовс. Командная строка, где выполняются все команды. В данном случае он запущен в графической оболочке и имеет красивый вид. В противном случае (особенно на серверах, мы видим черный пугающий на первых парах экран и мигающий курсор, которых ждет наших команд).

Почему нужно обновляться так часто? Потому что обновления выходят постоянно, ведь в разработке принимает участие огромное количество людей по всему миру.

Многие проекты объединяют хакеров^[29] со всего света, которые знакомы только по переписке. Создать свой проект или присоединиться к уже существующему может любой и, в случае успеха, результаты работы станут известны миллионам пользователей. Пользователи принимают участие в тестировании свободных программ, общаются с разработчиками напрямую, что позволяет быстро находить и исправлять ошибки и реализовывать новые возможности.

Дистрибутив:

Дистрибутивы Линукс

Дистрибутив GNU/Linux — общее определение операционных систем, использующих ядро Linux, готовых для конечной установки на пользовательское оборудование. Поставляемая в рамках дистрибутива операционная система состоит из ядра Linux и, как правило, включает в себя набор библиотек и утилит, выпускаемых в рамках проекта GNU

Слайд 8

Дистрибутивов существует целое множество:

Показать количество дистрибутивов расшарив экран

Ubuntu

Ubuntu почти всегда №1 во всех чартах, потому что, помимо того, что его легко установить с ним интуитивно просто работать.

В моей работе я сталкивался, по большей части именно с этим дистрибутивом. Большинству разработчиков он по душе. И выглядит хорошо и пользоваться приятно. Обновляется довольно часто и имеет последнюю версию 21,10

Linux Mint (от [англ.](#) *mint* — «мята») — развиваемый сообществом бесплатный [дистрибутив Linux](#), основанный на [Ubuntu](#) и [Debian](#). Целью проекта является предоставление «современной, элегантной и удобной операционной системы, которая одновременно является мощной и простой в использовании»^[4]. Linux Mint предоставляет полную поддержку разнообразных форматов мультимедиа, включает в себя некоторые [проприетарные программы](#) и поставляется в комплекте с обширным набором приложений с открытым исходным кодом. Основатель проекта — Клемент Лефевр, в развитии также активно участвуют команда разработчиков (Mint Linux Team^[5]) и сообщество пользователей.

OpenSUSE

По некоторым данным это единственный дистрибутив, который управляется сообществом и только сообществом. И не зависит от каких-то важных и главных разработчиков. так это или нет, выяснить мне не удалось. Однако он занимает почетное третье место.

Arch Linux

не простой в использовании дистрибутив по сравнению с **Ubuntu** или **Linux Mint**, потому что он нацелен на компетентных пользователей **Linux**. Для успешного использования вам нужно, немного больше практики чем у обычного пользователя.

благодаря своей системе сборки пользователи могут настраивать внутреннюю работу пакетов по своему вкусу и даже делиться своим кодом с другими пользователями с помощью репозиторий-сервера **Arch Linux**.

Шутка про то, как узнать что у твоего коллеги стоит arch linux

Deepin OS

Это китайский дистрибутив. Но на алиэкспресс его искать бесполезно. Потому что он так же бесплатен.

Подходит только для более новых 64-х битных процессоров. Основан на дистрибутиве Ubuntu

Elementary OS

Этот дистрибутив появился относительно недавно, в 2013 году. И она зарекомендовала себя, как операционная система для новичков. Она интуитивно понятна и очень красиво выглядит. Разобраться в ней не составит труда, ну а красивый вид не оставит равнодушным даже привередливого пользователя.

Manjaro OS

Manjaro делает упор на удобство использования и доступность, а сама система предназначена для работы полностью «прямо из коробки» с разнообразным предустановленным [программным обеспечением](#). **Manjaro** обладает полной поддержкой [мультимедиа](#), распознаёт оборудование, поддерживает несколько ядер процессора.

Если вы уже выросли из предыдущей Elementary - то Манджаро для вас!

Fedora

Этот дистрибутив находится впереди планеты всей по используемым технологиям. И всегда был в топе у большинства разработчиков как мировых, так и русскоговорящих.

Kali Linux

Считается, что **Kali Linux** является самым передовым дистрибутивом для тестирования на уязвимости, который поставляется с множеством инструментов для этого. Поэтому, иногда его называют хакерским. Довольно сложен для новичка, поэтому его используют, как правило, более опытные пользователи. И если вы являетесь таковым, то будьте уверены, когда вы начнете работу с ним, вы будете приятно удивлены.

Слайд 9

В отличие от Windows, программа не находится в одной папке, а, как правило, распределена по корневой файловой системе. Это распределение поддается определенным правилам.

Так же стоит учитывать, что слеш \ в системе виндовс и системе линукс имеет другое направление. Важно не запутываться с первых шагов.

По поводу распределение файлов, если совсем коротко и понятно - [Структура каталогов Linux](#) не просто использует разные имена для папок и файлов. В ней применяется в целом иной принцип их расположения. Например, приложение в Windows может хранить все свои файлы в папке C:\Program Files\Имя_приложения, в то время как в Linux данные файлы будут разделены между несколькими расположениями: двоичные файлы будут помещены в /usr/bin, библиотеки — в /usr/lib, а конфигурационные файлы — в /etc/.

Кратко описать структуру:

/ - КОРЕНЬ

Это главный каталог в системе Linux. По сути, это и есть файловая система Linux. Здесь нет дисков или чего-то подобного, как в Windows. Вместо этого, адреса всех файлов начинаются с корня, а дополнительные разделы, флешки или оптические диски подключаются в папки корневого каталога.

Только пользователь root имеет право читать и изменять файлы в этом каталоге. Обратите внимание, что у пользователя root домашний каталог /root, но не сам /.

/BIN - (BINARIES) БИНАРНЫЕ ФАЙЛЫ ПОЛЬЗОВАТЕЛЯ

Этот каталог содержит исполняемые файлы. Здесь расположены программы, которые можно использовать в однопользовательском режиме или режиме восстановления. Одним словом, те

утилиты, которые могут использоваться пока еще не подключен каталог `/usr/`. Это такие общие команды, как `cat`, `ls`, `tail`, `ps` и т.д.

/sbin - (SYSTEM BINARIES) СИСТЕМНЫЕ ИСПОЛНЯЕМЫЕ ФАЙЛЫ

Так же как и `/bin`, содержит двоичные исполняемые файлы, которые доступны на ранних этапах загрузки, когда не примонтирован каталог `/usr`. Но здесь находятся программы, которые можно выполнять только с правами суперпользователя. Это разные утилиты для обслуживания системы. Например, `iptables`, `reboot`, `fdisk`, `ifconfig`, `swapon` и т.д.

/etc - (ETCETERA) КОНФИГУРАЦИОННЫЕ ФАЙЛЫ

В этой папке содержатся конфигурационные файлы всех программ, установленных в системе. Кроме конфигурационных файлов, в системе инициализации `Init Scripts`, здесь находятся скрипты запуска и завершения системных демонов, монтирования файловых систем и автозагрузки программ. Структура каталогов `linux` в этой папке может быть немного запутанной, но предназначение всех их - настройка и конфигурация.

/dev - (DEVICES) ФАЙЛЫ УСТРОЙСТВ

В `Linux` все, в том числе внешние устройства являются файлами. Таким образом, все подключенные флешки, клавиатуры, микрофоны, камеры - это просто файлы в каталоге `/dev/`. Этот каталог содержит не совсем обычную файловую систему. Структура файловой системы `Linux` и содержащиеся в папке `/dev` файлы инициализируются при загрузке системы, сервисом `udev`. Выполняется сканирование всех подключенных устройств и создание для них специальных файлов. Это такие устройства, как: `/dev/sda`, `/dev/sr0`, `/dev/tty1`, `/dev/usbmon0` и т.д.

/proc - (PROCESS) ИНФОРМАЦИЯ О ПРОЦЕССАХ

Здесь содержится вся информация о запущенных процессах в реальном времени. По сути, это псевдофайловая система, содержащая подробную информацию о каждом процессе, его `Pid`, имя исполняемого файла, параметры запуска, доступ к оперативной памяти и так далее. Также здесь можно найти информацию об использовании системных ресурсов, например, `/proc/cpuinfo`, `/proc/meminfo` или `/proc/uptime`. Кроме файлов в этом каталоге есть большая структура папок `linux`, из которых можно узнать достаточно много информации о системе.

/var (VARIABLE) - ПЕРЕМЕННЫЕ ФАЙЛЫ

Название каталога `var` говорит само за себя, он должен содержать файлы, которые часто изменяются. Размер этих файлов постоянно увеличивается. Здесь содержатся файлы системных журналов, различные кеши, базы данных и так далее. Дальше рассмотрим назначение каталогов `Linux` в папке `var/`.

/var/log - ФАЙЛЫ ЛОГОВ

Здесь содержатся большинство файлов логов всех программ, установленных в операционной системе. У многих программ есть свои подкаталоги в этой папке, например, `/var/log/apache` - логи веб-сервера, `/var/log/squid` - файлы журналов кеширующего сервера `squid`. Если в системе что-либо сломалось, скорее всего, ответы вы найдете здесь.

Почему логи переменные? Лог ротейшен (описать)

/var/lib - БАЗЫ ДАННЫХ

Еще один тип изменяемых файлов - это файлы баз данных, пакеты, сохраненные пакетным менеджером и т.д.

/VAR/MAIL - ПОЧТА

В эту папку почтовый сервер складывает все полученные или отправленные электронные письма, здесь же могут находиться его логи и файлы конфигурации.

/VAR/SPOOL - ОЧЕРЕДИ

Изначально, эта папка отвечала за очереди печати на принтере и работу набора программ cups.

/VAR/LOCK - ФАЙЛЫ БЛОКИРОВОК

Здесь находятся файлы блокировок. Эти файлы означают, что определенный ресурс, файл или устройство занят и не может быть использован другим процессом. Art-get, например, блокирует свою базу данных, чтобы другие программы не могли ее использовать, пока программа с ней работает.

/VAR/RUN - PID ПРОЦЕССОВ

Содержит файлы с PID процессов, которые могут быть использованы, для взаимодействия между программами. В отличие от каталога /run данные сохраняются после перезагрузки.

/TMP (TEMP) - ВРЕМЕННЫЕ ФАЙЛЫ

В этом каталоге содержатся временные файлы, созданные системой, любыми программами или пользователями. Все пользователи имеют право записи в эту директорию.

Файлы удаляются при каждой перезагрузке. Аналогом Windows является папка Windows\Temp, здесь тоже хранятся все временные файлы.

/USR - (USER APPLICATIONS) ПРОГРАММЫ ПОЛЬЗОВАТЕЛЯ

Здесь находятся исполняемые файлы, исходники программ, различные ресурсы приложений, картинки, музыка и документация.

/USR/BIN/ - ИСПОЛНЯЕМЫЕ ФАЙЛЫ

Содержит исполняемые файлы различных программ, которые не нужны на первых этапах загрузки системы, например, музыкальные плееры, графические редакторы, браузеры и так далее.

/HOME - ДОМАШНЯЯ ПАПКА

В этой папке хранятся домашние каталоги всех пользователей. В них они могут хранить свои личные файлы, настройки программ и т.д. Например, /home/sergiy и т.д. Если сравнить с Windows, то это ваша папка пользователя на диске C, но в отличие от Windows, home как правило размещается на отдельном разделе, поэтому при переустановке системы все ваши данные и настройки программ сохраняются.

/BOOT - ФАЙЛЫ ЗАГРУЗЧИКА

Содержит все файлы, связанные с загрузчиком системы.

/LIB (LIBRARY) - СИСТЕМНЫЕ БИБЛИОТЕКИ

Содержит файлы системных библиотек, которые используются исполняемыми файлами в каталогах /bin и /sbin.

/OPT (OPTIONAL APPLICATIONS) - ДОПОЛНИТЕЛЬНЫЕ ПРОГРАММЫ

В эту папку устанавливаются проприетарные программы, игры или драйвера. Это программы созданные в виде отдельных исполняемых файлов самими производителями. Такие программы устанавливаются в под-каталоги /opt/, они очень похожи на программы Windows, все исполняемые файлы, библиотеки и файлы конфигурации находятся в одной папке.

/MNT (MOUNT) - МОНТИРОВАНИЕ

В этот каталог системные администраторы могут монтировать внешние или дополнительные файловые системы.

/MEDIA - СЪЕМНЫЕ НОСИТЕЛИ

В этот каталог система монтирует все подключаемые внешние накопители - USB флешки, оптические диски и другие носители информации.

/SRV (SERVER) - СЕРВЕР

В этом каталоге содержатся файлы серверов и сервисов. Например, могут содержаться файлы веб-сервера apache.

/RUN - ПРОЦЕССЫ

Еще один каталог, содержащий PID файлы процессов, похожий на /var/run, но в отличие от него, он размещен в TMPFS, а поэтому после перезагрузки все файлы теряются.

/SYS (SYSTEM) - ИНФОРМАЦИЯ О СИСТЕМЕ

Назначение каталогов Linux из этой папки - получение информации о системе непосредственно от ядра. Это еще одна файловая система организуемая ядром и позволяющая просматривать и изменять многие параметры работы системы, например, работу swap, контролировать вентиляторы и многое другое.

слайд 10

Так же, очень важно помнить, что системы на линуксе чувствительны к регистру. И в этом нам поможет убедиться вот этот рисунок.