

DVD ДАРОМ!



4,7 ГБ ДЛЯ НОВИЧКА

# LINUX FORMAT

Главное в мире Linux

Март 2018 № 3 (234)

## 60-МИНУТНЫЙ ЭКСПРЕСС-КУРС ПО LINUX!

- » Простая пошаговая установка
- » Индивидуальная настройка
- » Игры, видео и не только!

ЕСТЬ  
**ЭЛЕКТРОННАЯ  
ВЕРСИЯ**  
ДЛЯ ВАШЕГО  
ПЛАНШЕТА!



### Звук & виртуализация

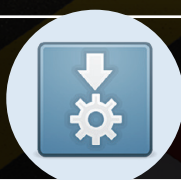
« До ветвления WebKit мы были одним из крупнейших участников, после Apple и Google »

Альберто Гарсиа — про QEmu и испанское СПО

Открытые приложения

### Snap & Flatpak

» СПО становится быстрее и проще устанавливать



Безопасность

### Защитимся с Prey

» Открытое приложение для слежения и блокировки устройств



## РАСКУРОЧИМ РОУТЕР

» Обезопасим свой Wi-Fi и раскроем его мощь

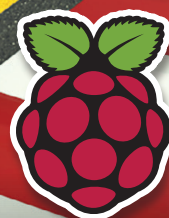


Академия кодига: Лихо расправляемся с CMake



**ПЛЮС!**

Meltdown & Spectre: Как защитить свой ПК



### Плюс: Для Pi

В копилку Raspberry Pi

- » Реанимируем игры на Amiga
- » Возделываем свой сад
- » Строим умного Twitter-бота



ПОДПИСНЫЕ ИНДЕКСЫ В КАТАЛОГАХ  
Агентство «Роспечать» — 36343,  
«Почта России» — 11932, «Пресса России» — 90959

Linux center  
www.linuxcenter.ru



# iTeleRadio

ИНТЕРАКТИВНОЕ РАДИО&ТЕЛЕВИДЕНИЕ

iteleradio.ru





## Что мы делаем

- » Мы поддерживаем открытое сообщество, предоставляя источник информации и площадку для обмена мнениями.
- » Мы помогаем всем читателям получить от Linux максимум пользы, публикуя статьи в разделе «Учебники»: здесь каждый найдет что-то по своему вкусу.
- » Мы выпускаем весь код, появляющийся на страницах раздела «Учебники», по лицензии GNU GPLv3.
- » Мы стремимся предоставлять точные, актуальные и непредвзятые сведения обо всем, что касается Linux и свободного ПО.

## Кто мы

На сей раз мы спросили наших экспертов: в этом месяце мы описываем новичкам в Linux радости и забавы открытого кода. А какие область/проект/работа в этой сфере вам нравятся больше всего?



### Джонни Бидвелл

Если что-то ломается, это вовсе не здорово; но сообразить, что пошло не так и как это починить — огромное удовольствие. Как минимум, когда вы сдвигаетесь с мертвой точки. Раньше я использовал Gentoo и, по-моему, больше чинил Linux, чем в нем работал. Хм-м...



### Нейт Дрейк

Однажды, слоняясь по офису, я наткнулся на этического хакера, и тот, в обмен на печенюшку, рассказал мне об InfoSec. По зрелом размышлении, это была честная сделка, поскольку я монетизировал свою воинствующую паранойю и впридачу обрел смутное чувство защитника людей.



### Джон Найт

Для меня это рабочий стол. Это *мой* рабочий стол. Сам-то я парень из KDE, но практически в любом обычно доступно множество настроек. Всякие штуки вроде виртуальных рабочих столов и управления сессиями позволяют создать окружение, подогнанное под ваши прихоти, с чем проприетарные предложения никогда не сравнятся.



### Лес Паундер

Самый приятный аспект открытого продукта — то, что он всеобъемлющ. Я больше не думаю об Open Source как чисто о программном обеспечении. Теперь у нас есть оборудование с открытым исходным кодом, например, Arduino и 3D-принтеры. Эти устройства затем используются для создания новых открытых проектов, которые растут как грибы!



### Маянк Шарма

Когда я переводеваю техническим журналистом, один из лучших аспектов этой деятельности — взаимодействие с разработчиками проекта и пользователями. В отличие от корпоративно-коммерческих типов, осязаемая страсть всей экосистемы к проектам с открытым исходным кодом делает общение более значимым и живым.



### Валентин Синецын

Уверен, сейчас все коллеги соловьем разливаются, как круто иметь возможность починить в системе все то, что сломалось. А почему никто не говорит, как круто, когда система не ломается? Известно же, больше глаз — меньше ошибок, и в последнее время мне куда реже приходится открывать исходники, чтобы что-то чинить. Если только из любопытства...



## Безвестный герой

» В череде новостей «из окопов кибервойны» и сообщений об очередных ограничениях в Интернете, 25-летний юбилей одного из важнейших событий в его истории остается как-то незамеченным и неотмеченным. А ведь именно четверть века назад руководство CERN приняло историческое решение — разрешить использовать свои разработки WWW (код сервера и браузера) как Public Domain, свободно и бесплатно для всего человечества. По моему мнению, это решение изменило мир едва ли не в большей степени, чем все прочие достижения CERN вместе взятые.

С формально-бюрократической точки зрения правильнее было бы поступить по-другому. Либо закрыть проект и забыть, что и попытаться сделать непосредственный руководитель Тима Бернерса-Ли [Tim Berners-Lee] и Роберта Кайо [Robert Cailliau], либо передать разработку стандарта в Международный союз электросвязи (ITU), благо он находится там же в Женеве. В последнем случае, скорее всего, дело кончилось бы появлением в забытом ныне стеке протоколов ISO еще одного мертворожденного стандарта. Идея свободных лицензий тогда не была столь очевидна, но именно такое решение позволило избежать многолетних патентных войн, в которые впоследствии втянулись авторы первых графических браузеров.

В дни юбилея нельзя не назвать имени человека, принявшего решение. Исторический меморандум вышел за подписью директора CERN по науке Вальтера Хоогланда [Walter Hoogland], безвестного героя цифровой революции...

### Кирилл Степанов

Главный редактор

» [info@linuxformat.ru](mailto:info@linuxformat.ru)

## Как с нами связаться

Письма для публикации: [letters@linuxformat.ru](mailto:letters@linuxformat.ru)

Подписка и предыдущие номера: [subscribe@linuxformat.ru](mailto:subscribe@linuxformat.ru)

Техническая поддержка: [answers@linuxformat.ru](mailto:answers@linuxformat.ru)

Общие вопросы: [info@linuxformat.ru](mailto:info@linuxformat.ru)

Проблемы с дисками: [disks@linuxformat.ru](mailto:disks@linuxformat.ru)

Вопросы распространения: [sales@linuxformat.ru](mailto:sales@linuxformat.ru)

Сайт: [www.linuxformat.ru](http://www.linuxformat.ru), группа «ВКонтакте»: [vk.com/linuxformat](https://vk.com/linuxformat)

» Адрес редакции: Россия, Санкт-Петербург, пр. Медиков, 5, корп. 7

» Телефон редакции: (812) 309-0686. Дополнительная информация на с. 112

# Знаете всё о Linux?

ДА

НЕТ

Станьте автором  
в журнале Linux Format!  
[linuxformat.ru/avtoram.phtml](http://linuxformat.ru/avtoram.phtml)

Зарегистрируйтесь  
на сайте  
[shop.linuxformat.ru](http://shop.linuxformat.ru)



СОМНЕНИЯ

Скачайте бесплатно  
архивные PDF-номера  
журнала с сайта  
[linuxformat.ru/archive](http://linuxformat.ru/archive)  
и загляните на  
[wiki.linuxformat.ru](http://wiki.linuxformat.ru)



Версия для iPad  
и iPhone доступна  
в App Store

Выберите вид подписки

PDF-версия  
на 6 месяцев

1485 ₺

PDF-версия  
на 12 месяцев

2760 ₺

Печатная версия  
на 6 месяцев

2430 ₺

Печатная версия  
на 12 месяцев

4500 ₺

Выберите вид доставки

Оплатите

# Читайте Linux Format!

Станьте Linux-гуру



## PDF-версия журнала Linux Format подойдет для тех, кто:

- Заботится о соблюдении прав деревьев
- Любит читать с экрана
- Мечтает получать каждый номер в день выхода журнала
- Хочет бесплатно скачивать содержимое DVD-приложения к каждому номеру

## Печатная версия Linux Format понравится читателям, которые:

- Любят читать бумажные журналы
- Хотят получить в подарок подписку на PDF-версию Linux Format...
- ...а также диск с архивом журнала с 2005 года
- Порадуются новинкам открытого ПО на DVD-приложении к Linux Format в каждом номере

## Способы доставки

- Курьером по Москве и Санкт-Петербургу
- Курьером и в пункты выдачи **iml.ru** по всей России
- Почтой по всей России заказной или простой бандеролью
- Самовывоз из офиса «ГНУ/Линуксцентра» в Санкт-Петербурге

## Способы оплаты

- По квитанции в любом отделении Сбербанка
- Яндекс.Деньги, Webmoney
- Пластиковой картой Visa/MasterCard
- Наличными в офисе «ГНУ/Линуксцентра»
- Безналичный (для юридических лиц)



«ГНУ/Линуксцентр»  
Санкт-Петербург,  
пр. Медиков, 5, корп. 7  
(метро «Петроградская»)  
(812) 309-0686  
[www.linuxformat.ru](http://www.linuxformat.ru)



# Содержание

«Если наши принципы справедливы, зачем нам быть трусами?» Лукреция Мотт

## Обзоры

### Freespire 3.0 ..... 12

Насколько успешно прошло возрождение ОС Lindows, скончавшейся в 2008 г., и стоит ли с ней связываться в этой новой форме?

### Siduction 2018.01 ..... 13

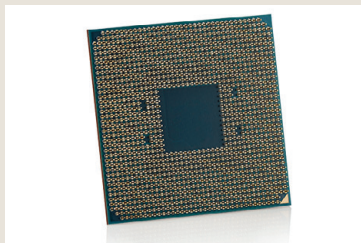
Ну как любителям риска не поинтересоваться дистрибутивом, построенным на явно нестабильном программном обеспечении! Хотя, по правде говоря, не всё так рискованно...

### Daphile 17.09 ..... 14

У вас богатая фонотека, и вы ищете музыкальный сервер? Быть может, решение в виде специализированного дистрибутива вам подойдет.

### AMD Ryzen 3 1200 ..... 15

Реально малобюджетный процессор сдержал обещание предоставить надежную четырехъядерную производительность. Вроде даже без раздражающих недостатков.



➤ Процессор простенький, но не без вкуса.

### Intel Core i3 8350K ..... 16

Угадajte, у чего цена почти как у Core i5, но скорость меньше? У этого самого чипа, якобы дешевого; а на поверку-то — не очень.

### Thimbleweed Park ..... 17

Приключенческая игра типа «наведи-ищелкни» строит сюжет вокруг детективной истории, пользуясь нарезкой из глаголов.



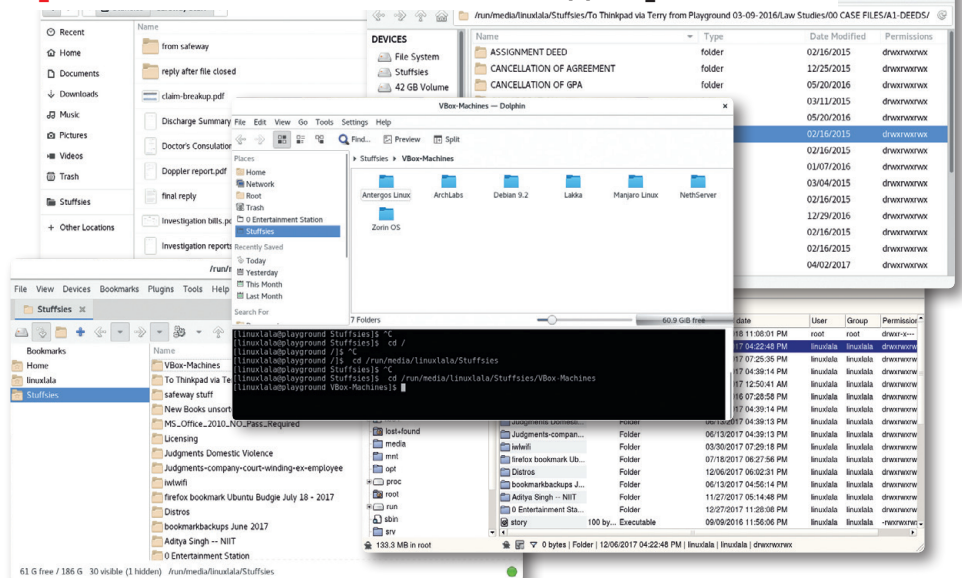
➤ Отзовите свой иск: Малдер и Скалли не любят лишней мороки.

## 60-МИНУТНЫЙ ЭКСПРЕСС-КУРС!

Откройте для себя самую мощную ОС в мире — мы выведем вас из нулей в герои. Начинаем на с. 26



## Сравнение: Файловые менеджеры с. 20



## Интервью



« У меня есть версия Debian, которую я установил еще в 1997 году! »

Альберто Гарсиа — о QEmu и испанском FLOSS с. 36



# На вашем бесплатном DVD



**Elementary OS 0.41** 64-битный  
**Bodhi Linux 4.4** 32-битный  
**ROSA Linux R10** 64-битный

» Каждый месяц — только лучшие дистрибутивы

ПЛЮС: HotPicks, код и библиотека с. 106

Побалуйте себя и любимых подпиской на LXF!

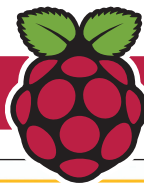


Доступно в AppStore!

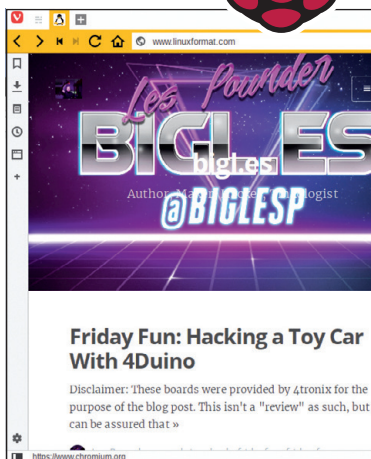


www.linuxformat.ru/subscribe

## Пользователям Raspberry Pi



- Новости Pi** ..... 88  
Что будет, если смонтировать 750 Raspberry Pi в одном шкафу, чего достигли новые контактики в Pi Zero, плюс новости Spectre/Meltdown.
- Web-браузер Vivaldi** ..... 89  
Любителям статистики предоставляется способ сосчитать, сколько раз они искали в Сети пирожки.
- Универсальный эмулятор Amiga** ..... 90  
Amiga близка многим сердцам, и мы попытаемся возродить к жизни золотой век вычислений.
- Садовый Pi** ..... 92  
Pi заботится об огородных посадках и поливает цветочки! А иначе на что и датчики?
- Twitter-боты** ..... 96  
Настройте своего собственного Twitter-бота — и заспамьте Сеть, а потом свалите всё на нас.



## Ищите в номере

- Snaps vs Flatpaks** ..... 40  
Люди требуют более быстрой и простой системы распространения пакетов... поинтересуемся новым поколением средств пакетирования и спросим: оно нам надо?



## Учебники

- Основы терминала** ..... 52
- Мультимедиа** ..... 52  
Потрясающие способности MPlayer ждут не дожидаясь вашего появления в терминале!
- Настольные приложения** ..... 54
- Видео 360 ВР** ..... 54  
Эта новорожденная форма искусства начинает вставать на ноги — спешно обзаводиться программой...
- Основы терминала** ..... 58
- MEncoder** ..... 58  
Делаем риппинг видео прямо из терминала, призвав на помощь старшего брата MPlayer.

- Защита** ..... 60
- Prey** ..... 60  
Путь только кто-нибудь попробует спереть ваш ноутбук или мобильник — за ним следит недреманное око.

- Шифрование** ..... 62
- Энтропия** ..... 62  
Как ни странно, случайность бывает как бы и не совсем случайной. Если бы не гелевый светильник...

- Виртуализация** ..... 64
- Сети и VirtualBox** ..... 64  
Презрев настройки по умолчанию, сколотим сеть, надежно прикрытую брандмауэром.

- Роутеры** ..... 66
- Linux Embedded DE** ..... 66  
Проприетарные прошивки — дело поправимое, и вы увидите, что ваш роутер на самом деле не так уж и плох.

- Разработка** ..... 72
- Toolchain** ..... 72  
Изучаем компиляторы не только в теории, но и на практике: для начала попробуем собрать один такой.

## Академия кодинга

- CMake как профи** ..... 76  
Поговорим о CMake, файлах настройки и визуализации зависимостей проекта с помощью Graphviz.
- Python и массивы** ..... 80  
Самое развеселое, что вы можете натворить в Python, будучи еще в трезвом виде.
- Шифрование с Purple** ..... 84  
Вспомним дни Тихоокеанского конфликта Второй мировой войны, воссоздав японскую шифровальную машину.

```

1 cmake_minimum_required(VERSION
2
3 PROJECT( simple )
4
5 set(CMAKE_BINARY_DIR ${CMAKE_S
6 set(EXECUTABLE_OUTPUT_PATH ${C
7
8 # For including the header fil
9 include_directories(myLibrary/
10
11 set(SOURCES myApplication/mair
    
```

## Постоянные рубрики

- Новости** ..... 6  
AMD начала эру, указаны доли ОС на рынке, вышли российский SSD-накопители и сверхбыстрый от Samsung, Astra вошла на Эльбрус, Huawei патентует флэш-процессоры, Google ведется к квантовому превосходству, а Calculate Linux выдал новый релиз.
- Вести мобильных ОС** ... 18  
Apple использует облака Google, Raspberry Pi стал головным устройством, а ARM занялась машинным обучением и графическими процессорами.
- Сравнение** ..... 20  
Менеджеры файлов способны не только на навигацию по папкам: рассматриваем предложения от Dolphin, Gnome Files, SpaceFM, Thunar, XFE.
- Интервью LXF** ..... 36  
**Альберто Гарсия** верен своему старому доброму Debian и занимается всевозможной виртуализацией.
- Рубрика сисадмина** .... 46  
**Д-р Валентин Синицын** любит облаками, оценивает переход Amazon на KVM и описывает OSTree как Git для файловых систем ОС.
- Ответы** ..... 98  
**Джонни Бидвелл** взялся отвечать на вопросы по Qubes и решить проблемы с EFI, Arduino и оптическим приводом.

- HotPicks** ..... 100  
Лучшие в мире новинки свободного ПО: Android File Transfer, Bemuse, Catimg, Darktable, Ddgr, Desktopfolder, Flameshot, HexGL, Liquidshell, Midnight Commander, Posterizer.
- Диск Linux Format** ..... 106  
Содержимое двустороннего DVD этого месяца.
- Пропустили номер?** ... 108  
А ведь там приводилось реальное средство сделать ваш дистрибутив шустрым и поджарым...
- Через месяц** ..... 112  
Защитим свою конфиденциальность вопреки АНБ!



**В ЭТОМ НОМЕРЕ:** Новая зра AMD » Процентка по ОС » Сверхбыстрый SSD » Накопители России » Наш и только наш » Храним в облаке » Кубиты » Калькулируем

## НОВОСТИ AMD

# Для сетей и индустрии

Линейки процессоров Epyc и Ryzen дополнены встраиваемыми моделями.

**Р**асширение семейства своих продуктов Epyc и Ryzen чипмейкер AMD представляет как «начало новой эры высокопроизводительных встраиваемых процессоров». Модели Epyc Embedded 3000 и Ryzen Embedded V1000 рекламируются как обеспечивающие «взрывную производительность» и простую интеграцию, и предоставляющие расширенную встроенную защиту. «Epyc Embedded 3000 повышает планку производительности для виртуализации сетевых функций следующего поколения, программно определяемых сетей и приложений сетевых систем хранения, [тогда как] Ryzen Embedded V1000 предоставляет блестящую графику в одном

чипе, обеспечивая экономию места и энергии в решениях для медицинской визуализации, а также в игровых и промышленных системах», пояснил вице-президент AMD и генеральный директор Datacenter and Embedded Solutions Business Group Скотт Эйлор [Scott Aylor]. Epyc Embedded 3000 позиционируется как «высокомасштабируемый» чип с числом ядер от 4 до 16, доступный в однопоточной и многопоточной конфигурации. Гибридный Ryzen Embedded V1000 объединяет на одном кристалле CPU 'Zen' и GPU 'Vega' и предлагает до 4 ядер CPU и до 8 потоков и 11 процессорных ядер GPU на чипе, достигая пропускной способности до 3,6 терафлоп для операций FP16.

» AMD представила обладающие «взрывной» производительностью встраиваемые процессоры Epyc и Ryzen.



Некоторые клиенты AMD уже показали решения на новых чипах, в том числе ультразвуковую систему MyLab9 eXP от Esaote для общей медицинской визуализации, игровую платформу для казино Quixant QX-70 4K Ultra HD, и 4 продукта Advantech, включая интегрированную игровую платформу для казино.



» Рубрику готовил  
**АНДРЕЙ  
ГОНДАРЕНКОВ**

## СТАТИСТИКА

# Распределение нагрузки по ОСям

Анализируем отчет Netmarketshare о популярности среди пользователей различных настольных ОС.

**О**чередной отчет ресурса Netmarketshare показывает практически статичную картину. Windows 7 отлично зарекомендовала себя при корпоративном применении, и бизнес-пользователи не спешат отказываться от нее; «мягкие» угрозы, типа поддержки Office 2019 только Windows 10, или удаление старого клиента Skype, для значительной группы пользователей оказались недостаточным аргументом в необходимости инвестирования смены версии ОС. По итогам февраля доля Windows 7 на настольных ПК незначительно снизилась — до 41,61% (-0,78). У Windows 10 — 34,06%; это лучше, чем в декабре

2017 г. (32,93%), но хуже января (34,29%). Несколько удивил небольшой прирост Windows 8.x — 6,92% (+0,49); до 4,7% (+0,65) выросла доля Windows XP. Вполне

**Очередной отчет показывает практически статичную картину.**

предсказуемо ведет себя «яблочная» настольная ОС: доля новейшей macOS 10.13 растет — 4,96% (+0,5), тогда как у предыдущих версий она снижается: у 10.12 — 2,18%

(-0,47), у 10.11 — 1,39% (-0,08), у 10.10 — 0,83% (-0,04). Доля Linux — 1,46% (+0,5). По критерию «десктоп» из 5 лучших ОС по всем форм-факторам и здесь на первом месте Windows 7, с 18,73% (-0,22); далее идет Windows 10 — 15,33% (+/-0,00); на третью позицию поднялась iOS 11.2 — 11,99% (+1,56), потеснив Android 7.0 Nougat — 10,69% (-1,45); замыкает пятерку Android 6.0 Marshmallow — 8,89% (-0,46). Глобально по ОС на всех устройствах и во всех версиях картина выглядит так: Windows — 39,46% (+0,22), Android — 36,1% (-1,22), iOS — 18,46% (+0,91), macOS — 4,45% (+/-0,00), Linux — 0,96% (0,08).



ИМПОРТОЗАМЕЩЕНИЕ

# На базе собственных технологий

GS Group запустил массовое производство первых российских SSD-накопителей.

**Х**олдинг GS Group объявил о начале массового производства первых SSD-накопителей собственной разработки. Полный производственный цикл реализован в инновационном кластере «Технополис GS» (инвестиционный проект GS Group в г. Гусеве Калининградской обл.), мощности которого позволяют выпускать более 1 млн устройств в год. До конца 2018 г. на рынок выйдет линейка SSD емкостью до 1 ТБ для различных применений. GS Group с 2016 г. реализует первый в РФ проект по разработке и массовому производству SSD для клиентского и корпоративного сегментов. Первый серийный образец SSD-накопителя емкостью 256 ГБ в форм-факторе 2,5" был выпущен в 2017 г. В устройствах использован интерфейс SATA 3.0 с максимальной скоростью передачи данных до 6 Гбит/с, накопитель обеспечивает максимальную скорость последовательного чтения до 550 Мб/с и последовательной записи до 450 Мб/с. По своим характеристикам устройства

не уступают зарубежным аналогам. Предусмотрена возможность индивидуальной настройки оборудования под конкретные требования заказчика — как на аппаратном, так и на программном уровнях. Весь производственный цикл — проектирование устройств, включая разработку, монтаж компонентов на плате, корпусирование модулей памяти, финальную сборку и упаковку изделий — реализован на мощностях инновационного кластера «Технополис GS». Локализация полного цикла разработки и производства в РФ обеспечивает максимальную безопасность данных.

В основе устройств новой линейки будут использованы произведенные в «Технополисе GS» модули памяти, в составе которых — последнее поколение кристаллов памяти 3D NAND от ведущих мировых производителей. Специалисты GS Nanotech запускают в производство технологию сверхплотной упаковки кристаллов, благодаря которой будет обеспечена недостижимая ранее удельная емкость памяти микросхем.



» «Технополис GS» — площадка, где создано массовое производство российской электроники.

По прогнозам IDC, мировой рынок твердотельных накопителей ежегодно будет расти на 14,8% в денежном выражении (на 15,1% — в штучном) и достигнет к 2021 г. объема в \$33,6 млрд. Ключевым драйвером роста эксперты называют переход на технологию 3D NAND.

НАКОПИТЕЛИ

## Что имеем — сохраним!

Samsung представила сверхбыстрый SSD огромной емкости в форм-факторе 2,5".

**S**amsung начала массовое производство самого емкого в мире твердотельного накопителя с интерфейсом Serial Attached SCSI (SAS) — PM1643, предназначенного для систем хранения данных нового поколения. Благодаря технологии V-NAND, использующей 64-слойную, 3-битовую NAND в чипах емкостью 512 ГБ, носитель с емкостью 30,72 ТБ вдвое превосходит вместимость и производительность предыдущей модели линейки емкостью 15,36 ТБ. Прорыв стал возможным благодаря объединению 32 новых микросхем флэш-памяти NAND емкостью 1 ТБ, каждая из которых состоит из 16 чипов V-NAND емкостью 512 ГБ. В дополнение к удвоенному объему памяти, по сравнению с твердотельными накопителями с интерфейсом SAS предыдущего поколения, уровень производительности вырос почти в 2 раза. При подключении по интерфейсу SAS со скоростью 12 Гб/с, PM1643

способен достичь скоростей случайного чтения и записи до 400 тыс. IOPS и 50 тыс. IOPS соответственно, а также скоростей последовательного считывания и записи до 2100 Мб/с и 1700 Мбайт/с, соответственно, что в 4 раза превышает стандартную скорость случайного чтения и в 3 раза — скорость последовательного чтения 2,5-дюймового твердотельного накопителя с интерфейсом SATA. В PM1643 применена технология Through Silicon Via (TSV) соединяющая чипы DDR4 емкостью 8 ГБ и создающая 10 блоков TSV DRAM емкостью по 4 ГБ, что в сумме обеспечивает 40 ГБ памяти DRAM. Впервые DRAM с технологией TSV используется на носителях SSD. Расширенное ПО, поддерживающее защиту метаданных, даст возможность восстановления данных после внезапных сбоев питания и алгоритм коррекции ошибок (ECC) для высокой надежности при минимальных затратах на техобслуживание. Носители



» Отличительные особенности Samsung PM1643: емкость, скорость, долговечность.

обеспечат высокий уровень надежности, позволяющий совершать одну перезапись объема накопителя в день (DWPD); за 5-летний гарантийный период каждый день может быть бесперебойно перезаписано до 30,72 ТБ данных. Средняя наработка на отказ (MTBF) составит 2 млн часов.

Источник: gs-group.com; news.samsung.com

СДЕЛАНО В РОССИИ

# Звезды опустили на вершину

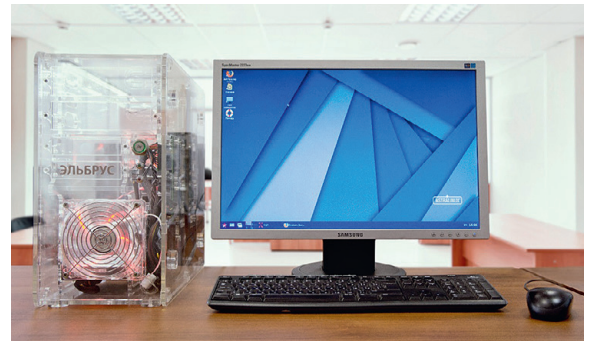
«Astra Linux» заработала на процессорах «Эльбрус».

**В** целях создания доверенных средств вычислительной техники на основе отечественной электронной компонентной базы и отечественной операционной системы, компании «НПО РубиТех» и АО «МЦСТ» проводят совместные работы по обеспечению функционирования операционной системы «Astra Linux Special Edition» на вычислительных комплексах «Эльбрус», использующих микропроцессоры с одноименной архитектурой. В ходе первого этапа работ достигнуты все поставленные цели: успешно осуществляется процесс сборки ОС с использованием отечественного компилятора для платформы Эльбрус, проведена серия успешных запусков ОС «Astra Linux Special Edition», в т. ч. графического пользовательского интерфейса. Запуски производились без использования системы двоичной трансляции кодов x86 в коды Эльбрус, что обеспечивает максимум быстродействия ОС и прикладного ПО. Для ускорения разработки были

использованы компетенции АО «МЦСТ», полученные при создании собственной ОС «Эльбрус».

В соответствии с устоявшимся принципом наименований релизов для идентификации варианта ОС «Astra Linux Special Edition» в честь 75-летия прорыва советскими войсками кольца блокады Ленинграда принято решение о присвоении ему имени города-героя — «Ленинград».

Работы проводятся в соответствии с согласованным сторонами планом-графиком, а полученные предварительные результаты позволяют говорить о том, что в 2018 г. защищенная ОС «Astra Linux Special Edition» сможет в полной мере применяться на компьютерах с отечественной архитектурой Эльбрус для создания информационных систем любой сложности, обрабатывающих информацию ограниченного доступа. В настоящее время ОС уже работает на компьютерах с российскими процессорами Байкал-Т1 и 1890VM8Я (КОМДИВ).



► **Графический интерфейс релиза «Ленинград» операционной системы «Astra Linux Special Edition» на вычислительных комплексах «Эльбрус».**

Кроме того, ОС функционирует на любом типе устройств (рабочие станции, тонкие клиенты, планшеты, серверы, системы хранения данных и др.) на процессорной архитектуре x86 (релиз «Смоленск»), мобильных устройствах с процессорами ARM (релиз «Новороссийск»), мейнфреймах IBM System z (релиз «Мурманск») и IBM System p, а также серверах Yadro с микропроцессорами Power8 (релиз «Керчь»).

## УПРАВЛЕНИЕ ДАННЫМИ

# Противостоять новым вызовам

Huawei представила флэш-ориентированную облачную высокоинтеллектуальную СХД.

**В**ысокопроизводительная интеллектуальная система хранения данных (СХД) типа all-flash (AFA) для критически важных корпоративных сервисов OceanStor Dorado18000 V3 разработана для высоконагруженных критически важных сервисных систем предприятий; благодаря флэш-ориентированной ОС решение способно в полной мере раскрыть потенциал флэш-памяти. Запатентованные флэш-процессоры Huawei и алгоритм FlashLinkTM обеспечивают максимальную производительности на уровне 7 млн IOPS с постоянным значением задержки 0,5 мс. Производительность новой модели в два раза выше, чем у ее предшественниц. Система использует технологию HyperMetro в режиме «active-active»; ресурсы хранения в этом режиме обслуживают одни и те же приложения, что гарантирует круглосуточную работу ключевых сервисов. Благодаря значению задержки

ниже 1 мс она способна гарантировать высочайший уровень надежности платформы SAP HANA, что делает ее оптимальным решением для критически важных сервисов предприятия. В системе используется технология Flash-Ice на основе графенового рассеяния (GDT) и динамического алгоритма теплового баланса на SSD; это позволяет повысить рассеивание тепла в 500 раз и продлить срок службы SSD на 20%. OceanStor Dorado18000 V3 идеально подходит для ключевых сервисов операторского уровня, таких как онлайн-биллинг в режиме реального времени, CRM и ERP, тем самым позволяя расширить абонентскую базу на несколько порядков. По сравнению с традиционными решениями, система поддерживает встроенное уплотнение данных без ущерба для производительности и гарантирует коэффициент уплотнения данных 3:1, тем самым повышая производительность в 10 раз и помогая клиентам



► **В ходе Всемирного мобильного конгресса в Барселоне компания Huawei представила высокопроизводительную интеллектуальную систему хранения данных типа OceanStor Dorado18000 V3.**

сократить объем инвестиций и повысить их доходность (ROI). По данным консалтинговой компании ESG, система OceanStor Dorado способна обеспечить пользователям 75-процентную экономию на общей стоимости владения.

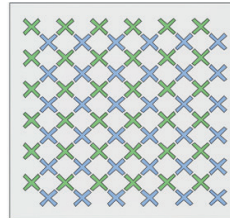


КВАНТЫ ПОБЕЖДАЮТ

# 72... кубита

Процессор Google Bristlecone обеспечит «квантовое превосходство» над традиционными компьютерами.

Подразделение Google Quantum AI Lab продемонстрировало эмулятор 72-кубитного компьютера на базе собственного квантового процессора Bristlecone. Еще один шаг на пути к «квантовому превосходству»: в данный момент Bristlecone является самым производительным решением в сфере квантовых вычислений. Однако задача достижения «квантового супрематизма» пока не решена; для этого необходимо обеспечить функционирование квантового процессора с достаточно низким процентом ошибок. В гонку «квантовых проектов» включились самые авторитетные технологические компании: так, в январе собственный 49-кубитный чип анонсировала Intel. Поскольку алгоритмы для квантовых процессоров выходят за рамки классического моделирования, требуется большое количество кубитов, а также низкое число ошибок при считывании и в логических операциях (1- и 2-кубитные вентили). Специалисты Google считают, что «квантовое превос-



ходство» может быть продемонстрировано даже с 49 кубитами, глубине схемы более 40 и уровне ошибок 2-кубитных вентилей менее 0,5%. 72-кубитный Bristlecone использует ту же схему связи, управления и считывания, что и предыдущий 9-кубитный линейный массив от Google. При аналогичной с предшественником производительности, новинка должна обеспечить снижение ошибок (1% при считывании, 0,1% для 1-кубитного и 0,6% для 2-кубитного вентилей у 9-кубитного устройства); тем самым будет доказана возможность создания крупномасштабных квантовых компьютеров.

➤ Новейший квантовый процессор Google Bristlecone (слева); каждый «X» — кубит в схеме, и его связь с ближайшими соседями (справа).

РОССИЙСКОЕ ПО

# Будем вычислять

Вышел Calculate Linux 17.12.2.

Новый релиз основанного на Gentoo российского дистрибутива Calculate Linux содержит следующие изменения:

- » выполнен переход на профиль Gentoo 17.0. Все двоичные пакеты в репозитории пересобраны и обновлены в загрузочных образах;
- » включены доступные на 22 февраля патчи для Meltdown и Spectre;
- » исправлен сброс прав на файлы **grub.d**;
- » добавлено восстановление file sabilities для lxc-контейнеров;
- » исправлена настройка и сохранение имен сетевых устройств и настройка сети внутри контейнера;
- » исправлена настройка пароля на редактирование записи в *Grub*;
- » удален параметр IO Scheduler;
- » исправлено изменение атрибутов файла при выполнении шаблонов;

- » исправлен запуск ядра системы (предупреждения и ошибки);
- » исправлена поддержка использования файла в качестве swap.

Для загрузки пользователям предлагаются редакции Calculate Linux Desktop с рабочим столом KDE (CLD), Cinnamon (CLDC), Mate (CLDM) и Xfce (CLDX), Calculate Directory Server (CDS), Calculate Linux Scratch (CLS), Calculate Scratch Server (CSS) и Calculate Container Scratch (CCS) для установки в контейнере.

В состав поставляемых с дистрибутивом пакетов входят KDE Frameworks 5.43, KDE Plasma 5.11.5 и KDE Applications 17.08.3, Cinnamon 3.4, Mate 1.18, Xfce 4.12, LibreOffice 5.4.5.1, Firefox 57.0.1, Evolution 3.24.6, GIMP 2.8.22, Rhythmbox 3.4.1, OpenLDAP 2.4.44, Samba 4.5.15, Postfix 3.2.4, ProFTPD 1.3.5e, Bind 9.11.2\_p1, ядро 4.14.19, Calculate Utilities 3.5.8.10. **LXF**

Новости короткой строкой

➤ На MWC 2018 компания LG представила смартфоны среднего ценового сегмента K8 и K10, с расширенной функциональностью по оптимальной цене. Источник: [www.zdnet.com](http://www.zdnet.com)

➤ В новой серии Qualcomm Snapdragon 700 Mobile Platform реализованы улучшения в работе с ИИ и камерой, производительности и батареи, возможности подключения и взаимодействия. Источник: [www.linuxjournal.com](http://www.linuxjournal.com)

➤ Компании Vodafone и Huawei совершили первый в истории звонок на базе стандарта неавтономной архитектуры «нового радио» (Non Stand-Alone (NSA) 3GPP 5G NR) и суб-6-Гц спектра. Источник: [www.huawei.com](http://www.huawei.com)

➤ Некоммерческая организация Eclipse Foundation подвела итоги голосования по смене имени проекта Java Platform Enterprise Edition; в результате Java EE переименован в Jakarta EE. Источник: [mmilinkov.wordpress.com](http://mmilinkov.wordpress.com)

➤ Еврокомиссия планирует введение спецналога на доходы Google, Apple, Facebook и Amazon, рассчитываемого по местонахождению пользователей, а не штаб-квартиры компании. Источник: [www.theregister.co.uk](http://www.theregister.co.uk)

➤ Обновленный Microsoft Quantum Development Kit приносит разработку квантовых приложений в среды macOS и Linux, позволяя использовать Visual Studio Code для создания ПО с поддержкой квантового моделирования. Источник: [www.theinquirer.net](http://www.theinquirer.net)

➤ В апреле состоится дебют GPU следующего поколения Nvidia GeForce GTX 20 на новой архитектуре Ampere. Источник: [www.theinquirer.net](http://www.theinquirer.net)

➤ Разработчики Fedora утвердили создание новой, полноценной и официальной редакции этого дистрибутива для устройств Интернета вещей. Источник: [nullrDute.com](http://nullrDute.com)

➤ Microsoft добавила в каталог-магазин Windows Store сборки Debian GNU/Linux и Kali Linux, обеспечив их быструю установку и запуск в Windows 10 при помощи прослойки WSL. Источник: [blogs.msdn.microsoft.com](http://blogs.msdn.microsoft.com)

➤ В кодовую базу ChromeOS принята изменение, разрешающее запуск виртуальных машин с Linux без перевода устройства в режим разработчика. Источник: [www.reddit.com](http://www.reddit.com)

СКАЖИ УЧЕБЕ

**ДА!**



**СЕРВЕРНЫЕ РЕШЕНИЯ  
LINUX  
SOLARIS  
ХРАНЕНИЕ ДАННЫХ  
JAVA  
ANDROID  
БЕЗОПАСНОСТЬ  
ЗАЩИТА ДАННЫХ**

Санкт-Петербург,  
ул. Яблочкова, 12, литер Ц  
(812) 611 1575  
[unixedu.ru](http://unixedu.ru)

**UnixEducationCenter**



Новинки программного и аппаратного обеспечения в описании наших экспертов



**АЛЕКСЕЙ ФЕДОРЧУК**  
Тэг «сарказм»  
по умолчанию,  
смайлики по вкусу.

## Linux: Какой выбрать?

Вопрос из заголовка лет 15 назад был любимой темой для обсуждения на всех ресурсах соответствующей тематики. Потом показалось, что острота его как-то сгладилась. А нынче он возник в обсуждениях сразу с несколькими моими онлайн-друзьями, отнюдь не начинающими. Правда, еще и в аспекте: какой дистрибутив лучше? Раз так, попробую ответить на оба вопроса. Подчеркивая, что это не более чем мое личное мнение.

На вопрос из заголовка ответить легко. Возьмите Top-100 Distrowatch<sup>а</sup>, закройте глаза и ткните пальцем. Ибо почти все активно развиваемые дистрибутивы равноценны по своим применительским качествам. Хотя могут различаться по потребительским. Так что тут есть риск напороться на узкоспециализированное решение типа Kali. Но скорее всего «слепой» выбор даст результат, вполне приемлемый для первого знакомства.

Со второй гранью «вечного» вопроса — немного сложнее. По вышеозвученной причине оценивать дистрибутивы в терминах «хорошо или плохо» — затруднительно. И мой личный рейтинг с позиций «интересно или нет» таков (кроме Cuntu, который вне конкуренции). Если хочется «гладкой» системы с KDE — Maui. При желании выбора рабочих сред — Antegros. Любителям нестандартных решений — Void Linux или NuTiX. Наконец, тем, кому нужно просто работать, не забывая себе голову всякими Linux'ами — Linux Mint или LMDE. Но (почти) в любом дистрибутиве, исключая откровенно халтурные подделки, найдется что-то интересное. [alv@posix.ru](mailto:alv@posix.ru)

## Сегодня мы рассматриваем:

**Freespire 3.0** ..... 12  
Оригинал этого дистрибутива был разработан в подражание внешнему виду Windows, чтобы завлечь новых пользователей, но в итоге зачух. Группа Black Lab Linux пробует его возродить.

**Siduction 2018.01** ..... 13  
Линуксоиды любят исследовать новое и неизведанное. И какой же дистрибутив предоставит для этого больше шансов, чем тот, что основан на нестабильном ПО?

**Daphile 17.09** ..... 14  
Позиционируемый как «сердце» цифровой музыкальной системы, Daphile поможет вам воспроизводить коллекцию музыкальных файлов, копировать CD и даже подключаться к радиостанциям.

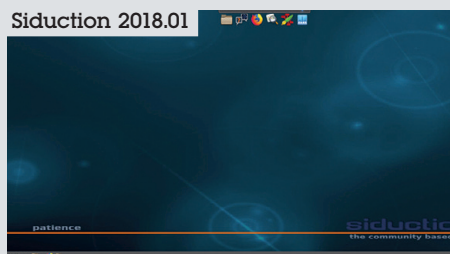
**AMD Ryzen 3 1200** ..... 15  
Этот в принципе малобюджетный чип способен приятно удивить вас отличным соотношением стоимости и качества, поскольку предлагает настоящий четырехъядерный процессор.

**Intel Core i3 8350K** ..... 16  
Приятно то, что этот чип разблокирован; неприятно — что за него заломили чрезмерную цену. Хотя его и можно разогнать, но нет ни гиперпоточности, ни режима Turbo.

**Thimbleweed Park** ..... 17  
В небольшом городке все жители знают друг друга, но найденный под мостом покойник — абсолютно неведомый чужак. Агенты должны его опознать и обнаружить орудие убийства...



➤ На взгляд-то он хорош, но такие деньги можно употребить с большей выгодой.



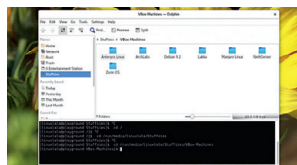
➤ Эта галактика предназначена не для паникеров, а для сильных духом.



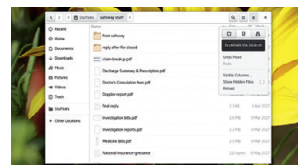
➤ Много глаголов, хороших и разных; выбирай себе на вкус и действуй соответственно.

## Сравнение: Менеджеры файлов с. 20

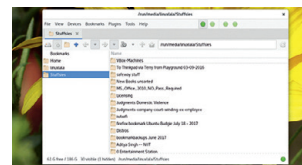
### Dolphin



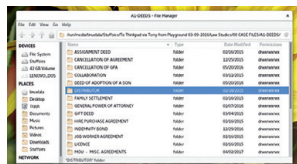
### Gnome Files



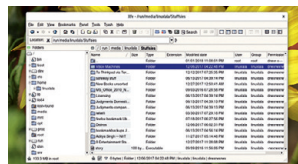
### SpaceFM



### Thunar



### XFE



Файловый менеджер сделался ключевым компонентом большинства рабочих столов. Но есть и независимые инструменты подобного рода. Их-то мы и рассмотрим.

# Freespire 3.0

Слово «возрождение» редко приложимо к мертвым дистрибутивам, но Freespire его заслужил благодаря смене руководства, обнаружил **Шашанк Шарма**.

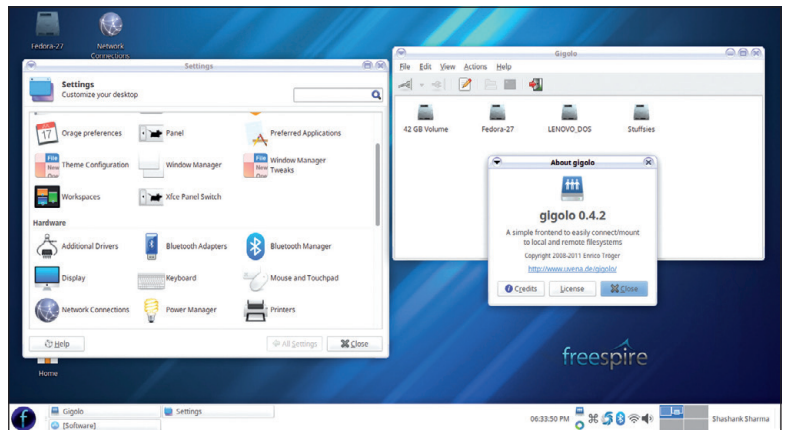
## Вкратце

» Freespire изначально назывался **Lindows** и быстро стал популярным среди пользователей, желающих отойти от **Windows**, но не от его внешнего вида. Основанный на **Ubuntu** и работающий на **Xfce** последний выпуск нацелен на новых пользователей. **Ubuntu**, **Linux Mint**, **Mageia** и наиболее популярные дистрибутивы являются сопоставимыми альтернативами.

Даже первый выпуск этого дистрибутива быстро сумел поляризовать сообщество Linux своим замыслом, подавая простоту в работе как основную особенность. Нет, мы говорим не об **Ubuntu**, а об оригинальной **Lindows OS**, разработанной в подражание внешнему виду **Windows**. С последним выпуском в 2007 г. дистрибутивы **Linspire/Freespire** умерли быстрой смертью после захвата их **Xandros** в 2008 г. Десять лет спустя появились новости о выпуске **Linspire 7.0** и **Freepire 3.0**, которые теперь принадлежат и разрабатываются **PC/OpenSystems LLC**, группы **Black Lab Linux**.

Первоначальный **Lindows** претерпел серьезные потрясения за годы, прошедшие с момента его запуска, включая изменение названия и управления. Проект также одним из первых предложил бесплатный настольный дистрибутив под названием **Freepire**, а также коммерчески лицензированный вариант, предлагающий поддержку под торговой маркой **Linspire**. Предметом нашего обзора является **Freepire**, который соответствует коммерчески предлагаемому **Linspire**, но лишен его проприетарного ПО и некоторых других популярных программ. Дистрибутив доступен как живой устанавливаемый ISO для 64-битных машин и имеет инструменты и ПО, выпущенные под **GPL**, **GPLv3**, **BSD** и кучей других лицензий для открытого исходного кода.

Дистрибутив всего на 1,5 ГБ скромен. В его предложениях программ по умолчанию нет ни одного крупного приложения, которые составляют основу большинства современных дистрибутивов. Например,



» Наряду со средой рабочего стола **Xfce**, дистрибутив включает почтовый клиент **Geary** и другие облегченные инструменты для повышения производительности.

Freepire не поставляет ни **GIMP**, ни даже полноценный офисный пакет. Из коробки предлагаются инструмент резервного копирования **DejaDup** и блокировщик доменов **MintNanny**, наряду со множеством повседневных приложений для Интернета и мультимедиа, и это можно приветствовать; но выбор ряда приложений по умолчанию вызывает недоумение. Дистрибутив поставляется с двумя ортодоксальными файловыми менеджерами вместе с **Thunar**. Также есть инструмент, помогающий настроить соединение **PPP**!

## Дефективная жизнь

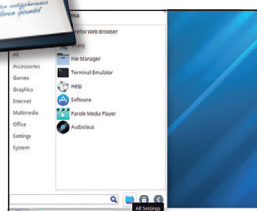
Хотя разработка ведется с июня 2016 г., Freepire поставляется с мелкими ошибками. О каких-либо ошибках не сообщается, но блокировщик домена **MintNanny** не работает, как рекламируется. Добавлять и удалять домены достаточно легко, но к сайтам всё же можно запросто получить доступ. Кроме того, вы не найдете **Settings Manager** в подменю **Settings**. Вместо этого надо нажать кнопку справа от строки поиска в нижней части панели запуска. Мы также сочли панель поиска быстрой и эффективной, что делает излишним включение еще и приложения **File Searcher**, предназначенного в помощь поиску пользователями любимых приложений среди установленных.

Последняя версия основана на **Ubuntu 16.04 LTS** и будет получать поддержку до 2021 г., но проект не предоставляет никакой информации о предстоящих выпусках или функциях, над которыми работает, помимо списка дат будущих выпусков **Linspire/Freespire** на сайте **Linspire**. Во всех

смыслах и целях **Freepire** — совершенно новый дистрибутив, связанный с предыдущими выпусками разве что именем. При таком сценарии крайне важно убедить пользователей в том, что проект какое-то время просуществует, а не просто исчезнет снова. Хуже того, **Freepire** не размещает на сайте никакой документации, кроме двухстраничного PDF-руководства по установке. Этого, конечно, следовало ожидать, раз уж проект продает коммерческие решения по поддержке; но также нет и средств для связи с сообществом пользователей.

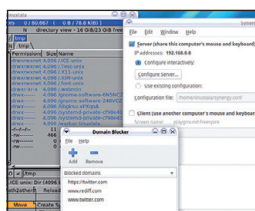
Хотя и годный к работе, нынешний выпуск **Freepire** не дает пользователям оснований уходить с их текущего дистрибутива Linux. А отсутствие информации о будущих выпусках является еще одной причиной, по которой мы советуем принять политику «поживем — увидим». **LXF**

## Свойства навскидку



### Эзотерические инструменты

Менеджер файлов **Worker** и **Superlu** задуманы для использования мыши и клавиатуры между машинами.



### Простота использования

Выбор **Ubuntu** упрощает использование **Freepire**, и вам также не понадобится получить общую поддержку.

## LINUX FORMAT Вердикт

### Freepire 3.0

Разработчик: **PC/OpenSystems LLC**  
Сайт: <http://bit.ly/freespire-3>  
Лицензия: Разные открытые

Функциональность	7/10
Быстродействие	7/10
Удобство в работе	7/10
Документация	1/10

» Легкий и быстрый, но относительно обычный. Отсутствие ясности насчет будущих выпусков повредит его принятию.

## Рейтинг 6/10



# Siduction Cinnamon

Допущение, что весь дистрибутив основан на нестабильном ПО, соблазняет **Шашанка Шарму**. Не является ли проект просто хитрым трюком?

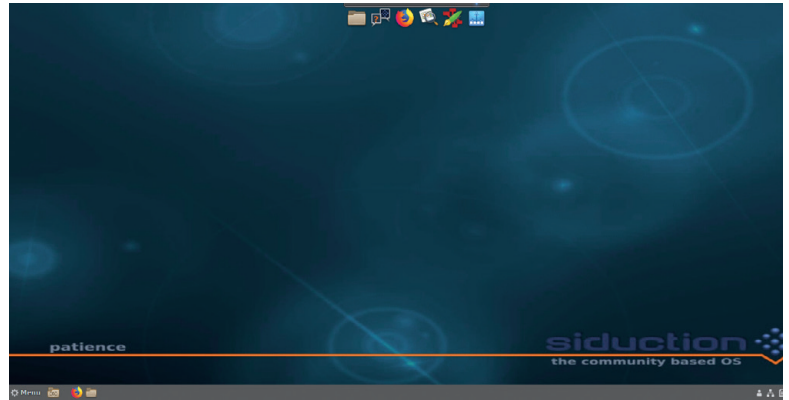
## Вкратце

» Основанный на Sid, нестабильной ветке Debian, дистрибутив полностью пригоден для использования, если его набор приложений по умолчанию о чем-нибудь говорит. Предупреждение о работе нестабильного ПО повторяется в заметках о выпуске, на сайте проекта, а также во время установки. Как и в дистрибутивах непрерывного обновления, типа Arch, вы можете использовать `dist-upgrade`, чтобы обновить вашу систему.

Вы знаете, что находитесь в центре экосистемы Linux, когда даже нестабильная ветвь порождает дистрибутив Linux; Siduction — это дистрибутив Linux на базе Debian Unstable, предназначенный для пользователей, которые хотят запускать новейшее ПО, но не хотят каждые полгода проходить через всю установку. Проект поставляется в виде почти 2 Гб live-образа с возможностью установки для 64-битных машин и имеет собственный установщик *Calamares*. В отличие от многих дистрибутивов-производных от Debian, которые управляются одним разработчиком или небольшой командой, Siduction имеет сравнительно большую команду разработчиков и сопровождающих.

Debian, который хорошо известен своим Debian Free Software Guidelines (DFSG) — набором правил, описывающим, какое ПО и инструменты можно включать в выпуск Debian на основе базовой лицензии на ПО, принял исключение для решения Siduction о поставке с проприетарными кодеками и драйверами из коробки. Более того, пользователям нельзя отказаться от этих проприетарных предложений. Найти список таких инструментов можно в примечаниях к выпуску или, альтернативно, запустить команду `vrms`. В дистрибутиве сперва предлагали специальный скрипт с именем `remove-nonfree`, чтобы избавить вас от установки всего DFSG, но потом прекратили. Вместо этого надо скомандовать `apt purge $(vrms -s)`.

Как и в предыдущих выпусках, дистрибутив поставляется в нескольких вариантах с различными рабочими средами, такими



» Несмотря на риск, присущий запуску нестабильного программного обеспечения, удобство и простоту использования Siduction трудно переоценить.

как *Xfce*, *Cinnamon*, *Mate*, *KDE* и *Gnome*. Для пользователей, которые не увлекаются графическим интерфейсом, есть версия *poX*, поставляемая без X. Однако *Gnome*, *Mate* и *LXDE*, вероятно, удалят из будущих выпусков Siduction, если только проект не найдет кураторов для этих выпусков.

## Спокоен и сдержан

Siduction проделал хорошую работу, чтобы создать своим пользователям приятное впечатление. Это видно по установщику *Calamares*, который теперь использует *Krpmcore* как инструмент разбиения на разделы. В дистрибутиве есть еще несколько собственных инструментов, таких как прозрачно названные скрипты *Activate SSH* и *Deactivate SSH*. Затем, есть скрипт *simple-paste*, который разработчики назвали швейцарским армейским ножом для вставки, плюс полезные и необычные инструменты вроде *G Alternatives*, с которыми администраторы могут определять инструменты, предоставляющие пользователям отдельные услуги.

При том, что проект полностью состоит из ПО, добытого из нестабильной ветки Debian, мы не сочли последний выпуск Siduction глючным или непригодным к употреблению. Единственным исключением была невозможность прокрутки мышью при чтении заметок о выпуске во время процесса установки. На всех остальных этапах установки мышью работает безупречно.

Экран загрузки при работе с живым диском немного запутан. Надо выбирать опцию для загрузки *From CD/DVD/ISO* или для загрузки *From Stick/HDD*. По умолчанию для дистрибутива установлен язык

`en_US` и клавиатура, но это также можно изменить из первого экрана.

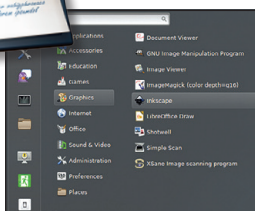
На нижней панели рабочего стола *Cinnamon* слева расположена панель запуска. В верхней части рабочего стола — док-панель, которая по умолчанию имеет значки для файлового менеджера *Nemo*, *Firefox* и клиента *HexChat IRC*. После установки Siduction док-панель также обеспечивает быстрый запуск *Image Viewer*.

Мы не смогли получить доступ к официальным форумам в качестве гостя, как и не смогли зарегистрировать новую учетную запись, но полное руководство — отличный ресурс для всех пользователей.

Пробуя Siduction, мы были готовы столкнуться с частыми сбоями, неотзывчивыми инструментами и другими ошибками поведения. Вместо этого мы обнаружили быстрый дистрибутив со впечатляющим набором программ по умолчанию. **LXF**

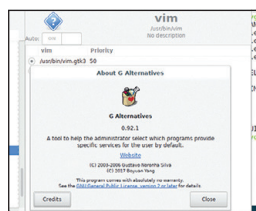


## Свойства навскидку



### Программное обеспечение

Siduction включает полезные приложения разных категорий, таких как Интернет и мультимедиа.



### Свои инструменты

Наряду с собственным установщиком *Calamares*, в дистрибутиве имеются специальные инструменты, например, *simple-paste*.

## LINUX FORMAT Вердикт

### Siduction 2018.1.0 Cinnamon

Разработчик: Siduction Team  
Сайт: [www.siduction.org](http://www.siduction.org)  
Лицензия: Разные

Функциональность	9/10
Быстродействие	8/10
Удобство в работе	9/10
Документация	7/10

» Не предназначен для работы на производстве, но на виртуальной машине идеален для тестирования нового программного обеспечения.

## Рейтинг 8/10

# Daphile 17.09

У вас много музыки, но вы изо всех сил пытаетесь настроить безголовый музыкальный сервер? Тогда **Шашанк Шарма** предложит решение...

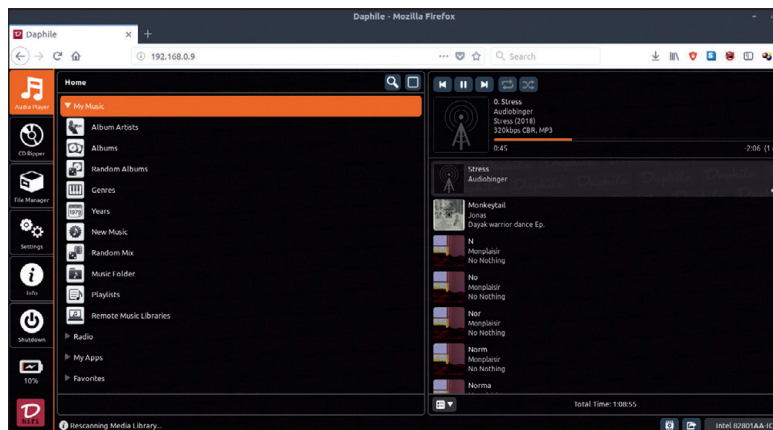
## Вкратце

» Дистрибутив для хранения и воспроизведения музыкальных файлов с сервера, которым вы можете управлять с любого компьютера в сети через web-браузер. Музыкальные файлы должны размещаться на самом сервере, что отличает Daphile от музыкальных стримеров вроде Airsonic.

Daphile идентифицирует себя как «сердце» цифровой музыкальной системы. Этот дистрибутив разработан, чтобы помочь вам воспроизводить свою коллекцию музыкальных файлов, копировать компакт-диски и даже подключаться к радиостанциям, и всё это с комфортом и удобством web-браузера. Однако Daphile — не обычный музыкальный накопитель. Другая особенность, делающая его удобным в использовании, заключается в том, что он не требует установки. Вы можете записать компакт-диск или использовать dd для создания живого образа 200 МБ на USB — и всё. Дистрибутив поставляется со всеми инструментами и кодеками для воспроизведения музыкальных файлов, которые уже хранятся на локальном жестком диске.

Но самым большим недостатком дистрибутива является его лицензионная модель. Хотя об этом не упоминается в FAQ, вики или где-либо на официальном сайте, краткое сообщение единственного разработчика на форумах DiyAudio гласит, что дистрибутив «в основном базируется на открытом исходном коде (около 200 различных пакетов). Всё, что я сделал сам, является частным и закрытым исходным кодом».

Еще более удивительно другое сообщение: «исходный код не распространяется отдельно через web-страницу Daphile. Daphile строится в основном с использованием стандартных сборок Gentoo. Желая получить список пакетов, свяжитесь со мной через контактную форму на сайте [www.daphile.com](http://www.daphile.com), и я отправлю его вам. Некоторые части Daphile, которые



» К сожалению, проект не поддерживает устройства ARM, и нет ни слова о том, будут ли они поддерживаться в будущих выпусках.

я разработал сам, не являются открытым исходным кодом». Мы с готовностью признаем, что происхождение от Gentoo в сочетании с небольшим размером и нулевой настройкой вначале привлекло наше воображение, но его бессистемный подход к лицензированию некоторым пользователям может не понравиться.

## Сетевые подключения

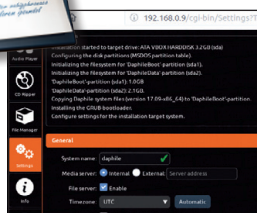
Если Daphile удастся обнаружить и настроить вашу сетевую карту, дистрибутив загрузится на экран с логотипом Daphile и сообщением «Ready. Use the Web Interface. IP address: 192.168.0.8». К любому элементу дистрибутива нет никаких средств доступа, кроме web-интерфейса. По умолчанию используется интерфейс Audio Player, который является первой вкладкой на боковой панели. Есть вкладки для File Mangers, Settings, CD Ripper и других. Двухпанельный Audio Player отображает всю доступную музыку, будь то медиа-файлы, радиостанции и т. д., в панели слева. Прослушиваемый файл или станция отображается в правой панели. Для поиска файлов служит строка вверху левой панели. Она поддерживает поиск по файлу, имени исполнителя, тегам и различным другим элементам, но во время наших тестов файлы найти не смогла.

При работе в живой среде вы не можете изменить какие-либо настройки по умолчанию, поскольку Daphile сообщает, что работает в режиме только для чтения. А значит, вы не можете настроить систему для доступа к музыкальным файлам на других машинах в вашей сети с помощью протоколов CIFS, SSH, NFS или FTP.

Желая настроить Daphile, вы должны установить его на диск. Но для этого у вас должен быть запасной диск без других разделов. В web-интерфейсе нажмите кнопку Settings на боковой панели, прокрутите вниз до System Firmware и выберите диск для установки. В отличие от других дистрибутивов Linux, Daphile не позволяет добавлять или удалять ПО даже после установки. После установки можно настроить музыкальные каталоги на разных машинах в меню Settings > Networking. Если всё будет хорошо, встроенные устройства и сетевые ресурсы будут доступны из файлового менеджера. Daphile не требует ни установки, ни какой-либо настройки перед использованием и может работать даже на машинах с 256 МБ ОЗУ. Если вы хотите превратить устаревшую машину в музыкальный сервер, то Daphile — для вас. **LXF**

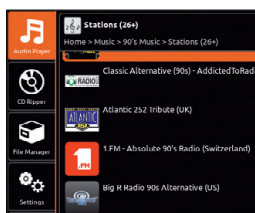


## Свойства новскидку



### Интерфейс браузера

Несмотря на многообразие элементов, составляющих полноценную музыкальную систему, интерфейс у Daphile четкий.



### Всесторонне одаренный

Воспроизведение музыки на разных устройствах, копирование CD и прослушивание радио по жанрам из разных стран.

## LINUX FORMAT Вердикт

### Daphile 17.09

Разработчик: Киммо Таскинен  
[Kimmo Taskinen]  
Сайт: [www.daphile.com](http://www.daphile.com)  
Лицензия: Разные

Функциональность	8/10
Быстродействие	8/10
Удобство в работе	7/10
Документация	4/10

» Без особых проблем или мороки обеспечивает простой интерфейс для музыкального сервера.

## Рейтинг 6/10



# AMD Ryzen 3 1200

Алан Декстер узнал, что самый малобюджетный вариант Ryzen подчеркивает стандарт многоядерности и разницу в производительности былых процессоров.

## Спецификация

- » Разъем AM4
- » Разрядность 64 бит
- » Техпроцесс 14 нм
- » Ядра 4
- » Потоки 4
- » Тактовая частота 3,1 ГГц
- » Турбо 3,4 ГГц
- » Кэш L1 384 КБ, L2 2 МБ, L3 8 МБ
- » Память DDR4, два канала, 64 ГБ макс.
- » Тепловыделение 65 Вт
- » PCIe 20 линий
- » Виртуализация AMD-V, AMD-Vi, 2x AES

**К**ак поется в песне, не всегда вы получаете то, что хочется, и вместо того, чтобы позволить себе монстра высокого полета, приходится обойтись малобюджетным чипом вроде этого. Но, отвлекаясь от цитат из *Rolling Stones*, это не вся история, когда речь идет о самом доступном чипе Ryzen AMD. К этому чипу обращаются не просто потому, что мало денег: он показывает, насколько продвинулись процессоры за год, и дает значительно больше, чем предполагает его цена в £100.

Такая сумма дает вам настоящий четырехъядерный процессор. Да, эта территория ранее была заповедником господствовавшего Core i5. Базовая частота Ryzen номинально равна 3,1 ГГц, но вы редко увидите, что его ядра работают с такой скоростью; обычно он работает ближе к 3,4 ГГц. В AMD Ryzen 3 нет параллельных потоков — возможно, чтобы отделить их от чипов Ryzen 5;

но это всё же быстрый чип, способный обрабатывать сразу четыре потока.

Мы сравнили его с Core i5-8400 восьмого поколения, просто потому, что на данный момент этот чип будет конкурентом в настольных компьютерах. На первый взгляд это несправедливое сравнение. Новый Core i5 — шестиядерный шестипоточный чип, и стоит вдвое больше, чем за это просит AMD, так что помните об этом, просматривая описания результатов тестов.

## Так держать

Говоря о контрольных показателях, мы с удовольствием сообщаем, что у Ryzen 3 1200 нет неприятных сюрпризов. Учитывая количество ядер и рабочую частоту, он работает как ожидалось. Это означает, что он находится внизу таблиц показателей, но делает это по отличной цене. Например, результат *Cinebench R15* в 478 может

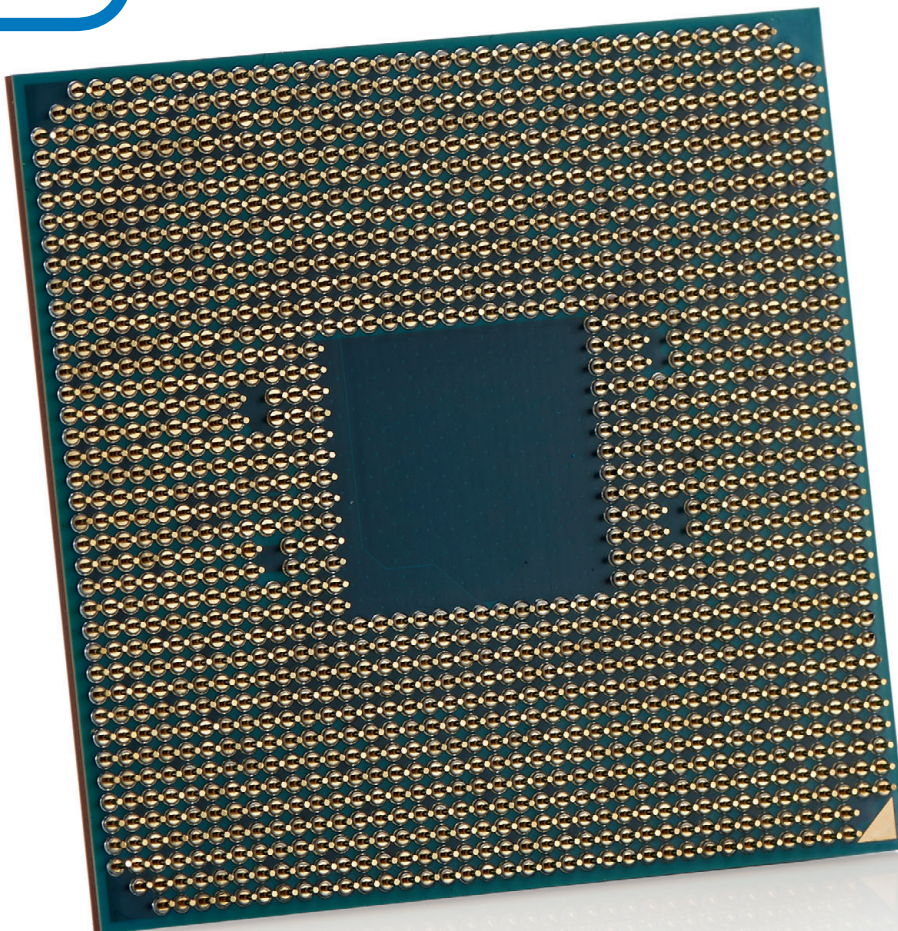
показаться низким по сравнению с 949 Core i5, но в контексте цены рассмотрите оценку 4,78 за фунт, в отличие от 4,75 для Intel. От чипа AMD вы на каждый дензнак получаете больше.

Стоит отметить, что благодаря последним обновлениям BIOS проблемы с памятью, которые мы видели при запуске Ryzen, испарились, и запуск системы до работающего состояния пролетает со свистом.

В корпусе с процессором размещен кулер Wraith Stealth, и у него прекрасные характеристики. Он достоин своего названия "Stealth [англ. украдкой, потихоньку]". Когда дело дошло до разгона, мы всё-таки потянулись за нашим любимым кулером NZXT Kraken X62, но со Stealth вы тоже сможете получить приличные показатели. Разгон Ryzen 3 1200 прост: установите требуемую частоту в BIOS (при необходимости увеличьте напряжение), и готово. В этом случае мы достигли стабильно 3,7 ГГц, что увеличивает показатель *Cinebench R15* до 567 баллов.

В целом это отличный чип за скромные деньги. Если вам нужна серьезная мощь, вы безусловно должны заплатить больше, но если цена — ваша главная забота, то его, конечно, есть за что полюбить; до тех пор, пока не появятся блоки APU Ryzen, оборудованные GPU. **LXF**

» Бюджетный чип AMD предлагает много за не такие уж большие расходы.



## LINUX FORMAT Вердикт

### AMD Ryzen 3 1200

Разработчик: AMD  
Сайт: [www.amd.com](http://www.amd.com)  
Цена: £100

Функциональность	8/10
Производительность	7/10
Удобство в работе	9/10
Оправданность цены	9/10

» Пока нет настольных версий APU Ryzen с поддержкой графики, этот процессор начального уровня дает хорошее соотношение цены и качества.

Рейтинг **8/10**

# Intel Core i3-8350K

Выводя концепцию бюджета на новое направление, Алан Декстер изо всех сил пытается идти в ногу, делая разворот в три приема в трафике Бата.

## Спецификация

- » Разъем 1151
- » Разрядность 64 бит
- » Техпроцесс 14 нм
- » Ядра 4
- » Потоки 4
- » Тактовая частота 4 ГГц
- » Кэш 8 МБ
- » ОЗУ DDR4, два канала, 64 ГБ макс.
- » Тепловыделение 65 Вт
- » PCIe 16 линий
- » Графика Intel UHD Graphics 630
- » Частота GPU 350 МГц (1,15 ГГц макс.)
- » Виртуализация VT-x, VT-d, EPT

У Core i3-8350K есть два важных момента. Во-первых, он разблокирован; а во-вторых, стоит £160. Хорошо, что его можно разогнать, но это, безусловно, слишком дорого для чипа, который большинство из нас воспринимают как малобюджетное предложение. Для пояснения — это не единственный доступный сейчас Core i3 восьмого поколения: скажем, Core i3-8100 укладывается в гораздо более приемлемые £100, и судя по этому ценнику, Intel явно в курсе, какие деньги можно ожидать выручить за бюджетное оборудование, даже если в данном случае они решили игнорировать эту мудрость.

В соответствии с новой маркировкой Intel, Core i3 является четырехъядерным процессором без гиперпоточности (как и Core i5), но заодно и без режима Turbo. Справедливости ради, базовая тактовая частота 4 ГГц достаточна, даже если она не будет прыгать вверх и вниз при задействовании большего или меньшего количества ядер. Как мы уже упоминали, данный CPU также допускает разблокирование, и если вы захотите ускорить чип, то вы можете. И вы действительно можете.

Вы получаете 8 МБ кэша, чтобы всё тархтело веселее; поддержку до 64 ГБ оперативной памяти DDR4; и встроенную графику в виде Intel UHD Graphics 630, с номинальной базовой частотой 350 МГц, способной достигать максимального значения в 1,15 ГГц. Это явно не составит конкуренции выделенным графическим чипам, но в случае управления новейшими играми означает, что можно собрать машину без отдельного графического процессора, тем самым снизив общую цену и габариты устройства по сравнению с системой, скажем, на процессоре Ryzen.

Однако налицо одна небольшая проблема, если уж обеспокоиться понятием бюджета: ваши варианты на материнской плате ограничены одним классом чипсетов, и это Z370, «для энтузиастов». Выбор опций здесь поразительно широк, но даже с самым дешевым, начиная с £90, мы немного отстаем от стартовой точки £55, которая легла в основу многих бюджетных Core i3 с использованием «мамок» B250 из предыдущего поколения. Должно быть больше бюджетных чипсетов, но пока комбинация этого чипа с самой дешевой платформой Z370 начинается с чуть менее £250. Слотны.



» Это мощный чип, но цена разочаровывает.

## О производительности

У нас могут быть свои соображения насчет позиционирования этого чипа, но когда дело доходит до производительности, скажем прямо: это здорово. Высокая тактовая частота в сочетании с четырьмя реальными ядрами дает отличные результаты. Действительно, в тестах этот чип шел по пятам производительности, предлагаемой процессором Core i5-7600K последнего поколения.

По сравнению с ближайшим по цене чипом AMD, Ryzen 5 1500X — который также четырехъядерный, хотя и с SMT (одновременная многopotочность), так что может обрабатывать восемь потоков — это борьба равных. Добавочные потоки Ryzen в некоторых областях помогают выиграть, но в других труды одноядерной производительности Intel перекрывают Команду Красных (т.е. AMD).

Когда дело дошло до разгона, нам удалось заставить наш чип работать на частоте 4,9 ГГц с очень небольшим добавочным напряжением (1,4 В), в результате получив оценку Cinebench 784 (при однопоточковой оценке 205). Печатающие цифры, даже если придется приплатить за кулер, чтобы таковых добиться.

Проблема для этого разблокированного Core i3 — его же собрат Core i5-8400 от Intel, чип, который стоит всего на £40 больше, но имеет на 50% больше ядер. У Core i5 также больше кэш и более низкое рассеяние мощности (TDP) — 65 Вт вместо 91. Конечно, Core i5 вы разогнать не можете, и это плюс для Core i3-8350K, но мы всё же предпочли бы иметь дополнительные ядра от Intel или дополнительные потоки от Ryzen. LXF

## LINUX FORMAT Вердикт

### Intel Core i3-8350K

Разработчик: Intel  
Сайт: [www.intel.com](http://www.intel.com)  
Цена: £160

Функциональность	7/10
Производительность	8/10
Удобство в работе	9/10
Оправданность цены	6/10

» По скорости — фантастический бюджетный процессор, но по цене устройства среднего диапазона, каковым он по сути не является.

Рейтинг **7/10**



# Thimbleweed Park

Мертвые тела, жуткий дух призраков, вербальный отборный поток сознания... По мнению **Энди Келли**, это обычный день в Башнях *Linux Format*.

## Спецификации

- » ОС Steam OS/Ubuntu 64-битный
- » ЦПУ 2 ГГц
- » ОЗУ 4 ГБ
- » Графика Intel HD 3000 или лучше
- » На диске 1 ГБ

**Ф**едеральные агенты Рэй [Ray] и Рейес [Reyes] расследуют убийство в захолустном городке Тимблвид-Парк [Thimbleweed Park]. Тело было найдено под мостом на окраине города, но, похоже, ни один из местных жителей ничего не знает ни об этом преступлении, ни о его жертве. У агентов есть контрольный список задач, которые они должны выполнить, чтобы раскрыть дело, включая опознание тела и обнаружение орудия убийства. Но вычислить убийцу будет нелегко, потому что это приключенческая игра «наведи-и-щелкни» в стиле *Monkey Island* с классическим глагольным интерфейсом «шведский стол»: используй, дай, собери, нажми и т. д.

Одной из первых головоломок является фотография тела, которая представляет собой введение в концепцию смены личностей и обмена предметов между ними. Но этот плавный старт скоро уступает место сложным, изощренным цепочкам головоломок, которыми знаменит (или, может быть, бесславен) этот жанр.

Если вы когда-нибудь задумывались, можно ли разрешить убийство бензопилой, какой-нибудь липкой лентой и монетой, самое время это выяснить. Распутывание одной проблемы в *Thimbleweed Park* может занять несколько часов, и все эти объекты — маленькие кусочки разных концов огромной, сложной головоломки.

Имеются и другие игровые персонажи: Рэнсом [Ransome] — клоун-сквернослов, живущий одиночкой, Делорес [Delores] — дизайнер игр, а кроткий продавец подушек Франклин [Franklin] — призрак, попавший в ловушку в отеле, где был убит.

Каждый персонаж имеет в запасе длинный список дел, что ненавязчиво задает вам направление поисков. Нет никакой



» Нет, это не Скалли и Малдер, хотя сходство заметно.

системы подсказок, но разговоров с людьми и проверок предметов в своем запасе часто достаточно, чтобы подвести вас к решению. Игра поражает тонким балансом между подачей тонких намеков и неуклонным отказом как-то помочь, благодаря чему разгадка особо сложной головоломки доставляет огромную радость. Вы чувствуете, что каждая победа честно заработана.

## На лоне природы

Вы застрянете, но открытая структура игры означает, что вы не упретесь в стену. Можно свободно исследовать город и окружающий район (включая отель и цирк), в конечном счете разблокируя карту, которая позволяет перемещаться между местоположениями почти мгновенно.

С пятью игровыми персонажами — и у четверых из них есть запасы предметов, включая немало таких, которые по сути ничего не делают — игра может иногда озадачивать. Порой мы чувствовали, что блуждаем в потемках, безнадежно комбинируя случайные предметы и пробуя каждый возможный глагол на каждом найденном объекте. Но в конечном итоге нам всегда удавалось вырваться из этих головоломок, и удовлетворение от этого в итоге означало, что все ломания головы и ругань стоили свеч.

*Thimbleweed Park* избегает ловушки «логики приключенческой игры», возможно, потому что манера LucasArts 1990-х гг. — зашибать деньги — давно закрыта. Нет такого ощущения, что туман напускается специально, просто чтобы жизнь не казалась вам малиной.

Юмор в приключенческих играх часто является клеем, который скрепляет всё вместе, и хотя *Thimbleweed Park* забавен, ему недостает теплоты и очарования *Monkey Island*. Игра слишком часто ссылается сама на себя и слишком сильно полагается на сарказм по поводу реальных шуток. Но ваш интерес поддерживают странные и красочные персонажи, развлекательный диалог и притягательная главная тайна.

*Thimbleweed Park* выражает самую суть приключенческой классики, избегая всяких вещей, из-за которых подобные игры вызывают раздражение. Более реальное взаимодействие с персонажами было бы лучше, и иногда эта история становится слишком мета-мета, но это одна из лучших современных приключенческих игр в стиле «наведи-и-щелкни». **LXF**



## LINUX FORMAT Вердикт

### Thimbleweed Park

Разработчик: Terrible Toybox  
Сайт: [www.thimbleweedpark.com](http://www.thimbleweedpark.com)  
Цена: £ 15

Сюжет	9/10
Графика	9/10
Увлекательность	8/10
Оправданность цены	9/10

» Качественная приключенческая игра со сложными головоломками, персонажами-странниками и интригующим сюжетом. Купите непременно.

**Рейтинг 9/10**



## ИРОНИЯ СУДЬБЫ

# Откровение для фанатов «яблок»

Данные iCloud хранятся в «облаках» Google.

Apple подтвердила, что использует Google Cloud Platform для хранения данных iCloud. На данную информацию, появившуюся в последней версии iOS Security Guide, обратил внимание телеканал CNBC. «Зашифрованные фрагменты файлов без какой-либо идентифицирующей пользователя информации хранятся с использованием сторонних сервисов хранения данных, таких как Amazon S3 и Google Cloud Platform», говорится в Security Guide; при этом не уточняется, какого именно типа данные iCloud хранятся в облаках Google. Слухи о том, что Apple использует облачные хранилища Google, начали появляться еще в 2016 г.,

а из Купертино не опровергали, но и не подтверждали свою «печальную тайну». Рост обеспокоенности пользователей по поводу сбора данных и конфиденциальности

**Apple надеется избежать подобных ситуаций в будущем.**

вынудили наконец Apple внести некоторую ясность в отношении того, где хранятся данные iCloud, как они защищены и кто может получить к ним доступ. Учитывая размер аудитории пользователей iPhone

и iCloud, неудивительно, что Apple вынуждена обращаться к помощи предоставляющих услуги облачного хранения сторонних фирм; ранее это были Amazon Web Services и Microsoft Azure. Но Google?.. Отношения между Купертино и Маунтин-Вью остаются прохладными на протяжении многих лет, но, видимо, интересы бизнеса перевесили корпоративную гордость. Напомним, что с iCloud было связано несколько неприятных моментов, в т. ч. возможность извлечь из него историю вызовов iPhone и сохранение «удаленной» истории поиска Safari. Видимо, опираясь на возможности Google, Apple надеется избежать подобных ситуаций в будущем.

## УМЕЛЫЕ РУКИ

# Зачем платить больше?

Linux превратил Raspberry Pi в недорогое головное устройство Android Auto.

Студент, разработчик СПО и блоггер Хуан Труонг [Huan Truong] предложил основанную на Linux Crankshaft OS, благодаря которой мини-компьютер Raspberry Pi 3 (\$35) с 7-дюймовым сенсорным экраном Raspberry Pi (\$60) превращается в функциональное (хотя и неофициальное) головное устройство Android Auto. Crankshaft OS основана на недавно выпущенном проекте OpenAuto, aasdk- и Qt-based эмуляторе для запуска Android Auto на Raspberry Pi. Но если OpenAuto требует дополнительной настройки, то Crankshaft OS, по словам автора, «дистрибутив GNU/Linux под ключ», который требуется только скачать и записать на SD-карту для загрузки Raspberry Pi 3. Представленная демо-модель (<https://youtu.be/oq1TR78kpl4>) использует присоску для крепления на лобовое стекло

и отображает приложения подключенного по USB телефона на Android в более удобном для водителя формате; к автомобильной стереосистеме Raspberry Pi подключается через 3,5-мм разъем для наушников. Обращаясь к пользователям Reddit с просьбой протестировать Crankshaft OS и сообщить об ошибках, Труонг отмечает, что ПО всё еще «находится на очень ранней стадии разработки» и поэтому может не работать на некоторых Android-телефонах. Тем не менее, первые тестировщики сообщают, что Crankshaft OS работает с Raspberry Pi 3, подключенным к телефонам Google Pixel 1 и 2, One Plus 3 и Nexus 6P; не заработали Moto G4 и Galaxy Note 8. Труонг пояснил, что мотивацией для создания Crankshaft послужила цена официальных головных устройств Android Auto, а также отсутствие таких



Создатель Crankshaft OS демонстрирует работу системы Android Auto на Raspberry Pi.

устройств, соответствующих его автомобилю 1998 модельного года, имеющему только один DIN-слот.



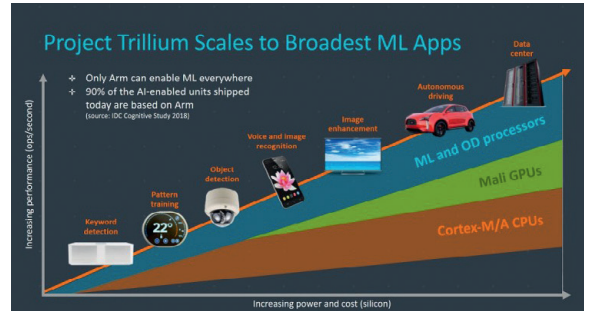
ИСКУССТВЕННЫЙ ИНТЕЛЛЕКТ

# Машинное обучение — на мобилы!

ARM представила варианты интеллектуальных процессоров.

Э то не развитие предыдущих проектов ARM, а две совершенно новые архитектуры; одна нацелена на поддержку рабочих нагрузок машинного обучения (ML), другая предназначена для задач обнаружения объектов (OD). Процессор ARM ML обеспечит машинное обучение на мобильном процессоре с теоретической пропускной способностью более 4,6 TOPs (трлн операций в секунду), позволив эффективно использовать системы ИИ и ПО на локальном устройстве. Процессор ARM OD разработан для идентификации людей и объектов в режиме реального времени, в разрешении Full HD 60 кадр/с. Как заявляет ARM, его производительность эквивалентна 80 традиционным DSP, и значительно превосходит другие чипы с точки зрения качества обнаружения объектов. При совместной работе OD и ML первый выделит необходимые

области изображения, а второй подвергнет их глубокой обработке. Благодаря новым архитектурам ARM, функции распознавания изображений, использующие высокоэффективные датчики и интеллектуальные технологии (типа Face ID в iPhone X), появятся и в устройствах более доступного ценового диапазона. Оба чипа созданы ARM в рамках проекта Trillium, который включает также ПО ARM NN для поддержки нейронных сетей на чипах ARM, что позволит ИИ вырваться из ЦОДов и серверных стоек и проникнуть во всевозможные устройства. Вероятно, ARM ML получат распространение как SoC (производства Qualcomm, Apple, Samsung и т.п.), которые уже используют процессоры ARM Cortex и за счет машинного обучения смогут улучшить смартфоны, планшеты и другие интеллектуальные устройства. ARM OD пойдет по тому же пути, но будет востребован



VP ARM Джем Дэвис [Jem Davies]: «Проект Trillium — абсолютно новый класс продуктов, состоящих из железа и ПО».

и в смарт-камерах, дронах и беспилотных автомобилях. Свой ИИ-чип разрабатывает и Amazon, улучшая интеллект Alexa. Но решения от ARM более популярны, и у британской компании есть реальный потенциал сделать ИИ более доступным и распространенным, чем когда-либо прежде.

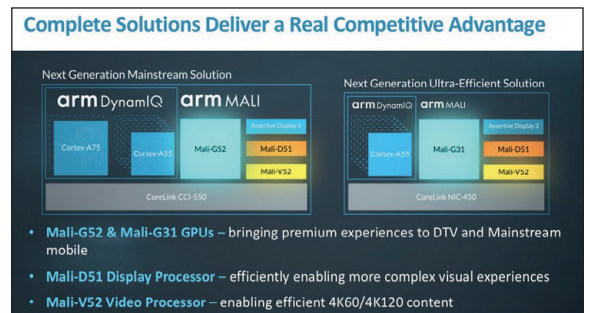
ПРЕДСТАВЛЯЕТ ARM

# Для среднего и малого бюджетов

Компания ARM показала новый комплект графических процессоров Mali.

Б ританский чипмейкер ARM продемонстрировал новые продукты для обработки мультимедиа и изображений в рамках Mali Multimedia suite. Графический процессор Mali-G52 расширяет возможности устройств при работе с модными мобильными приложениями, приложениями дополненной реальности (AR) для смартфонов и алгоритмами машинного обучения, хорошо сочетаемыми с параллельной обработкой GPU; заявлены 30% прироста производительности по сравнению с предыдущим поколением (G51) и 15-процентное повышение энергоэффективности. Процессоры на базе G52 придут в смартфоны и ТВ, обеспечивая высококачественные визуальные эффекты и управление такими функциями, как распознавание голоса. Второй графический процессор, Mali-G31, ориентирован на устройства нижнего уровня. Благодаря эффективной технологии

на архитектуре ARM Bitfrost, G-31 является самым маленьким процессором компании, поддерживающим графические API OpenGL ES 3.2 и Vulkan; это позволит улучшить графические возможности бюджетных устройств, не увеличивая их энергопотребление; новый процессор на 20% меньше и на 20% производительнее, чем Mali-G51. Еще одна новинка — Mali-V52, «мейнстримный» видеопроцессор следующего поколения», предназначенный для кодирования и декодирования видео 4K с 60 кадр/с; его площадь на 38% меньше, чем у более крупного V61. Mali-D51 — новый дисплейный процессор ARM, наиболее эффективный на сегодняшний день; выполненный на архитектуре Komeda процессор предлагает, по сравнению с предшественником, 30% экономии энергии и улучшение латентности памяти на 50%. Благодаря архитектуре ARM, ее дисплейные процессоры берут рендеринг интерфейса пользователя



Новый Mali Multimedia suite включает два графических процессора для основного рынка мобильной связи Mali-G52 и Mali-G31, дисплейный процессор Mali-D51 и видеопроцессор Mali-V52.

на себя, освобождая от этих задач основную GPU, что повышает энергоэффективность устройства.

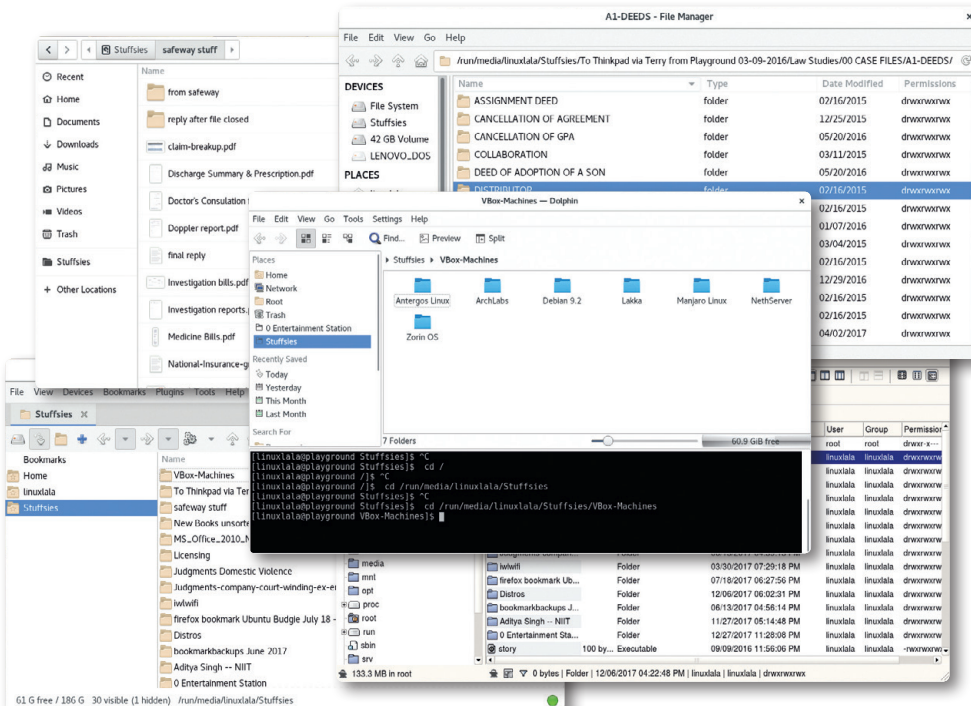
Новый комплект графических, видео- и дисплейных процессоров появится в решениях, предлагаемых Qualcomm и линейкой Samsung Exynos. LXF

# Сравнение

» Каждый месяц мы сравниваем тонны программ — а вы можете отдыхать!

## Менеджеры файлов

Менеджер файлов способен на большее, чем просто навигация по папкам. **Шашанк Шарма** рассматривает средства разборок с вашими данными.



### Про наш тест...

Инструменты в нашем списке были установлены на Fedora 27 Workstation. Начав с редакции Gnome, мы установили поверх среду KDE для доступа ко всем функциям менеджера файлов Dolphin. Все прочие инструменты доступны в программных репозиториях Fedora и других дистрибутивов.

Хотя сами по себе менеджеры файлов легковесны, иногда они требуют огромного количества ресурсов. Помимо производительности, мы рассмотрим возможности поиска в этих инструментах. Мы также рассмотрим, хорошо ли они интегрируются в рабочий стол и позволяют выполнять операции шифрования и сжатия.

Мы ищем инструмент, простой в работе. Бонусные баллы — если он имеет возможности индивидуальной настройки и позволяет добавлять функции через плагины и расширения.

**Д**ля огромного количества пользователей менеджер файлов — это всего лишь симпатичный инструментик, который помогает осуществлять навигацию по директориям на вашем диске. За несколько десятилетий этот вездесущий инструмент стал ключевым компонентом большинства рабочих столов.

KDE, Gnome, Mate, Cinnamon, Xfce и даже более новые рабочие столы, такие как Deepin, обладают собственным менеджером файлов. Эти инструменты хорошо интегрированы в рабочий стол и предлагают богатый ассортимент функций, таких как поиск, возможность создавать сжатые архивы и т. д.

Все инструменты в нашем списке разрабатываются далеко не первый год, и за исключением *SpaceFM* и *XFE* являются менеджерами по умолчанию популярных настольных дистрибутивов. И хотя файловый менеджер по умолчанию должен удовлетворять ваши основные потребности в компьютерном мире, мы искренне надеемся, что наше *Сравнение* откроет для вас нечто новое и интересное в тех проектах, о которых мы расскажем, и вы захотите

попробовать их благодаря их уникальным предложениям.

Правда, имеется огромный выбор менеджеров файлов, о которых мы не в состоянии рассказать из-за ограниченности места и собственного условия описывать исключительно инструменты GUI. Однако вы найдете список весьма полезных альтернатив, включая мощные варианты для командной строки, в разделе «Рассмотрите также...» на стр. 25.

**Мы искренне надеемся, что наше Сравнение откроет для вас нечто новое в тех проектах.**

### Наша подборка

- » Dolphin
- » Gnome Files
- » SpaceFM
- » Thunar
- » XFE



# Полезные функции

Что делает их особенными?

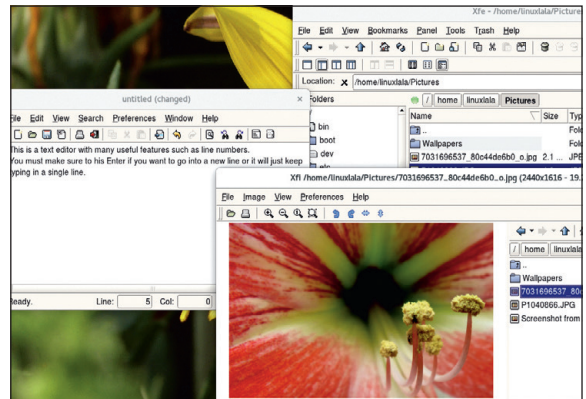
**В**се инструменты из нашего списка умеют выполнять основную функцию навигации по директориям. И все поддерживают функцию breadcrumb, которая позволяет отслеживать пути, когда вы всё глубже погружаетесь во вложенные директории. *Dolphin* и *Gnome Files*, самые старые проекты в нашем списке, также поддерживают функции Undo и Redo. Хотя в *Dolphin* эта функция существует уже долго, а *Gnome Files* только недавно ввели ее.

*Gnome Files* помнит последнюю выполненную операцию и позволяет отменить действие и вернуться к последнему действию, если это возможно. Например, если вы создаете новую папку и потом еще одну, последнее действие можно отменить. Т.е. удалится папка, созданная последней. Но если вы создаете новую папку или копируете несколько файлов в новую папку, тогда опция undo означает, что вы можете восстановить копирование.

*Dolphin* обладает куда большими возможностями касательно воспоминаний

и запоминания всех действий для открытия в данный момент окна. Менеджер файлов по умолчанию для KDE также предлагает интегрированный терминал. Он использует Konsole в качестве эмулятора терминала, так что для доступа к этой функции вы должны установить KDE. А еще *Dolphin* позволяет добавлять теги в ваши файлы и папки. Хотя в *Gnome File* разработка уже некоторое время ведется, всё же на момент написания *Gnome Files* функции тегов не предлагал, а у других менеджеров файлов в *Сравнении* этого месяца ее даже нет в списке планируемых.

Path Bar (адресная панель) в *SpaceFM* расположена в каждой панели над списком файлов для текущей директории. Проще всего сказать, что Path Bar позволяет увидеть путь к текущей папке, но она также обеспечивает функцию breadcrumb «хлебные крошки» — отсыл к истории Мальчика-с-пальчик, — прим. ред.]. Это означает, что вы можете, удерживая Ctrl, нажать на часть пути, чтобы перейти в эту директорию.



» Легковесный менеджер файлов XFE поставляется с такими дополнениями, как текстовый редактор (xftw), менеджер пакетов (xfr) и просмотрщик изображений (xfi).

Если вы находитесь в директории `/home/linuxlala/Documents/articles/2017/PDF/`, и, удерживая Ctrl, нажмете на часть пути `Documents`, то немедленно очутитесь в директории `~/Documents`. Можно также запускать команды из адресной панели, не выходя в *terminal*, но вы должны использовать один из следующих префиксов: `$` — чтобы запустить задачу, `&` для запуска команды в фоне, `+` для запуска команды в терминале и, наконец, `!` для запуска команды от имени root.

**Вердикт**

**Dolphin** ★★★★★

**SpaceFM** ★★★★★

**Gnome Files** ★★★★★

**Thunar** ★★★★★

**XFE** ★★★★★

» Хотя у них много функций, Thunar и XFE по сравнению с другими явно зачеточны.

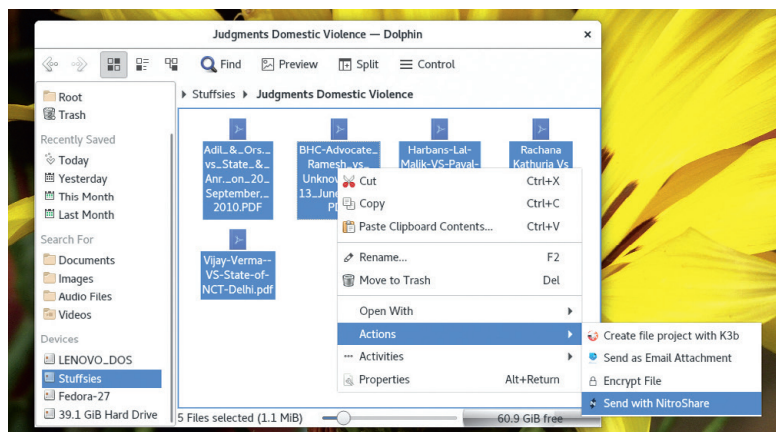
# Интеграция с рабочим столом

Ладят ли эти менеджеры файлов с другими программами?

**М**енеджеры файлов бывают самых разных форм и размеров, но, к счастью, они уже не столь тесно привязаны к своей родной среде рабочего стола, и вы можете установить любой из нашего списка поверх своего текущего рабочего стола. Однако несправедливо было бы слишком строго судить эти проекты, если вне своей родной среды некая функция не работает, как заявлено.

И *Gnome Files*, и *Thunar* позволяют настроить выбранное изображение в качестве обоев из контекстного меню. В нашей установке Fedora 27 Gnome, *Gnome Files* справились с этой задачей безупречно, чего не сумел *Thunar*. В *Gnome Files* можно также создать сжатые архивы выбранных файлов или отправить их в виде вложения в сообщении электронной почты, но это и всё.

По умолчанию *Dolphin* тесно интегрирован с рабочим столом, даже поверх Gnome, но только если вы установили KDE. Причина в том, что некоторые его функции, например, создание файла проекта [записи на CD/DVD — прим. ред.] с помощью *K3b*,



Send as mail [Отправить как почтовое сообщение] и шифрование, обращаются к таким программам KDE, как *Kmail* и *Kgpg* соответственно.

Все проекты распознают типы файлов и обеспечивают соответствующие действия, например, извлечение файлов из архивов, установка/удаление RPM-пакетов, монтирование ISO-образов, и т.д.

Кроме XFE, все проекты этого месяца позволяют выбирать файлы и отправлять их как вложения электронной почты, что по умолчанию требует настройки почтового клиента. Хотя *SpaceFM* поддерживает опцию отправки по электронной почте файлов, реализация этой функции здесь довольно странная: опция предлагается не для всех типов файлов.

» *Thunar* — единственный инструмент, неспособный создавать сжатые архивы файлов.

**Вердикт**

**Dolphin** ★★★★★

**Gnome Files** ★★★★★

**Thunar** ★★★★★

**SpaceFM** ★★★★★

**XFE** ★★★★★

» В отличие от других, XFE по умолчанию не знает, что делать с файлами ISO или PDF.

# Удобство в работе

Дружелюбие к пользователю — это достоинство.

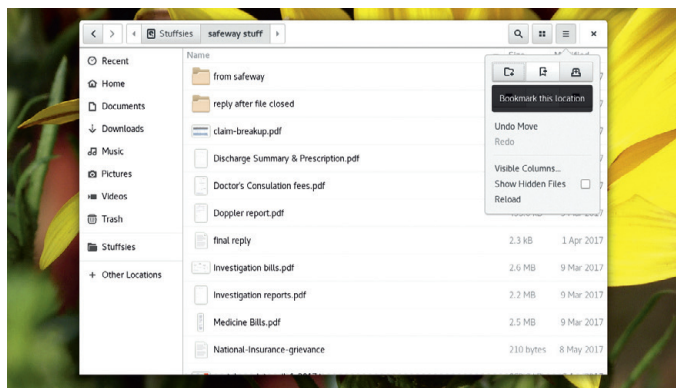
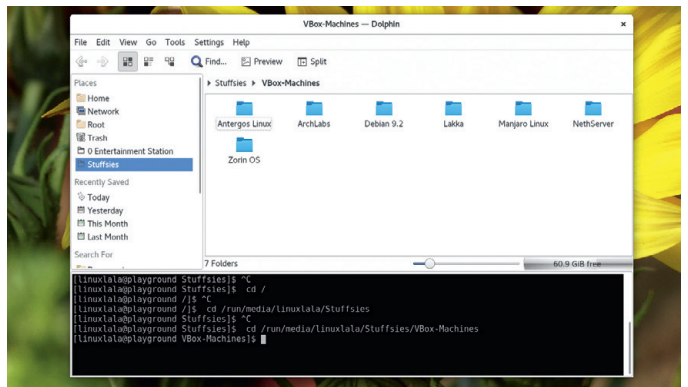
Смысл существования менеджера файлов заключается в том, чтобы вы могли разобраться со своими файлами и директориями. Вот почему так важны организационные способности. Все инструменты в нашем списке позволяют просматривать файлы в директории как список или в виде больших или маленьких значков.

Вы также можете увеличивать и уменьшать масштаб и перетаскивать файлы. Ну и нам нужен инструмент, который упрощает переключение между директориями и доступ к удаленным носителям и другим разделам на диске. Было бы также неплохо, если бы инструменты могли создавать закладки для часто используемых директорий.

По мере увеличения количества файлов, вам не всегда удастся вспомнить содержание файла по его имени. То же можно сказать об изображениях: описательного имени файла может быть недостаточно, чтобы вы вспомнили, какое там изображение, поэтому важна также и функция предпросмотра.

## Dolphin ★★★★★

Содержимое боковой панели делится на Places [Места], Recently Saved [Недавно сохраненные], Search for [Поиск] и Devices [Устройства]. Нажмите на любой пункт, и *Dolphin* перечислит все файлы, которые вы изменяли с момента первой загрузки за день. *Dolphin* также настраивается на показ добавочной информации по файлам и папкам в текущей директории. Из окна Preferences [Избранное] можно выбрать сервисы, обязанные быть доступными в контекстном меню, например, Run in Konsole [Запустить в консоли] или Send via Bluetooth [Отправить по Bluetooth], и *Dolphin* предлагает массу опций по умолчанию. Если в текущей директории много файлов изображений, кнопка Preview [Предпросмотр] на верхней панели покажет вам эскизы. Перетащите любую директорию на левую боковую панель, создав закладку; или, щелкнув правой кнопкой по заголовку Places на боковой панели, жмите на Add entry [Добавить пункт].



## Gnome Files ★★★★★

В отличие от *Dolphin*, у *Gnome Files* минималистский интерфейс, без панелей инструментов или множества кнопок, мешающих нормальной работе. Можно создавать новые папки и изменять масштаб, а также создавать закладки для текущей директории или просто перетаскивать для этого директорию на левую боковую панель. По умолчанию *Gnome Files* показывает имя, размер и дату изменения для каждого файла или папки в текущей директории. Жмите на кнопку Visible Columns [Видимые колонки], если вам нужна дополнительная информация, например, Type [Тип], Owner [Владелец] или Permission [Разрешения]. Намекаясь настроить инструмент, нажмите на значок Files на панели на рабочем столе и затем Preferences [Предпочтения]. Открытие файла/директории одним нажатием настраивается во вкладке Behavior [Поведение] окна Preferences. Как и *Dolphin*, *Gnome Files* умеет открывать несколько вкладок в окне.

# Документация и поддержка

Помощь лишней не бывает.

Помимо обзора функций, User-Base Wiki в KDE также предлагает подсказки, советы и руководства по наилучшему использованию *Dolphin*. Они охватывают навигацию по файловой системе, использование индивидуальных значков для папок, использование функции группового переименования и т.д. Страница File Management [Управление файлами] wiki пространно обсуждает интерфейс, панели, закладки и прочие функции. Хотя здесь нет специального форума, при необходимости вы можете воспользоваться

официальными форумами KDE. Еще один полезный ресурс — Handbook, который можно открыть, нажав на F1 в *Dolphin*.

Справка *Gnome Files* обсуждает все его функции и делится на разные категории, например, общие задачи, задачи, связанные с файлами, внешние диски, и даже предлагает несколько советов.

В дополнение к разработанной пользователями Wiki на GitHub, руководство пользователя *SpaceFM* на сайте предлагает подробное введение в работу с инструментом, а также подробно обсуждает разные опции

настройки и функции, предлагаемые Панелями, Path Bar и т.д. Разработанные пользователями плагины перечислены в wiki проекта. Форумы, размещенные на SourceForge, являются дополнительным ресурсом, и вы также можете обратиться к пользователям и разработчикам в IRC-канале #spacefm.

Сопровождаемая скриншотами документация для *Thunar* охватывает разные аспекты этого менеджера файлов, такие как интерфейс, работа с файлами/папками и полезные плагины.

**Вердикт**

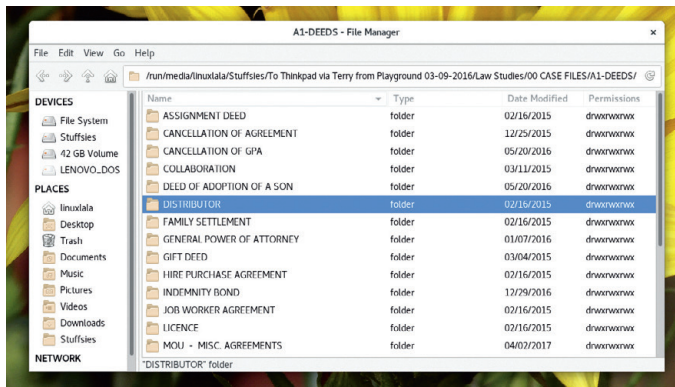
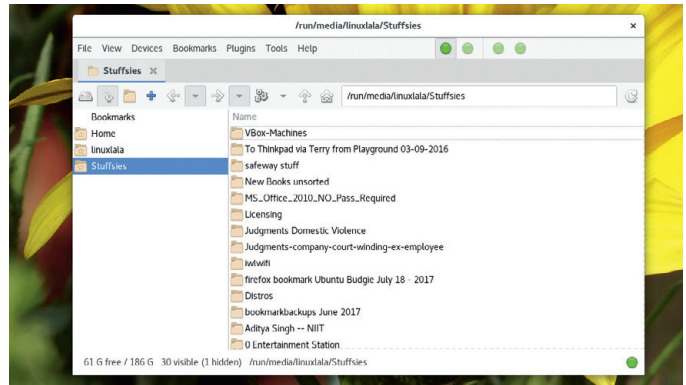
- Dolphin ★★★★★
- Gnome Files ★★★★★
- Thunar ★★★★★
- SpaceFM ★★★★★
- XFE ★★★★★

» Сайт XFE предлагает введение в функции и ничего более.



## SpaceFM ★★★★★

В отличие от *Gnome Files*, который поддерживает вкладки, но не разделенные панели, *SpaceFM* позволяет открыть четыре панели, и в каждой можно открыть несколько вкладок. Это удобно при работе с несколькими директориями одновременно, так как каждую панель можно использовать для независимой навигации по файловой системе. Для каждой панели вы можете выбрать дерево файловой системы, закладки и список внешних устройств, отображенные на боковой панели. Инструмент запоминает последнюю директорию, куда был доступ, и открывает новые экземпляры с той же директорией, в отличие от других инструментов, которые по умолчанию придерживаются директории ~/. Кроме того, интерфейс *SpaceFM* отлично настраивается индивидуально. Хотя прочие инструменты в *Сравнении* поддерживают масштабирование, позволяя увеличить значки в текущей директории, *SpaceFM* и *XFE* этого не умеют.

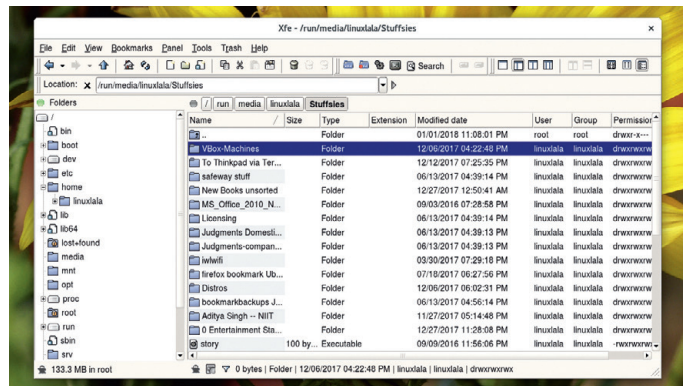


## Thunar ★★★★★

Интерфейс представляет одну строку состояния, строку меню и боковую панель, способную показывать закладки (именуемые ярлычками) или дерево файловой системы. Из окна Preferences можно настроить инструмент, чтобы он открывал файлы и директории. Желая, чтобы на главной панели отображалось не только имя и размер файла, жмите на View > Configure Columns [Вид > Настроить колонки]. Помимо *Dolphin* и *Gnome Files*, *Thunar* — еще один инструмент с поддержкой группового переименования, и делит еще одну функцию с *Gnome Files*: оба допускают Templates [Шаблоны]. Это документ, отформатированный по вашим спецификациям и используемый для создания новых документов. Чтобы создать новый файл-шаблон с требуемым форматированием и базовой структурой, щелкните правой кнопкой и выберите Create Document > <document-template> [Создать Документ > <документ-шаблон>].

## XFE ★★★★★

В отличие от своих соперников, интерфейс *XFE* по умолчанию предлагает ряд панелей инструментов со множеством кнопок. Инструмент поддерживает запуск терминала (Xterm) по умолчанию. Если вы используете альтернативный терминал, можете изменить настроенные приложения в окне Preferences. В теории *XFE* позволяет запускать *XFE* от имени пользователя root, но у нас это не сработало. Главное окно называется панелью и *XFE* предлагает несколько интерфейсов. В дополнение к виду по умолчанию, который представляет собой древовидный список на боковой панели и панель, можно также выбрать двухпанельный интерфейс, напоминающий разделенный интерфейс в *Dolphin*. Среди других опций — «одна панель» и «две панели с деревом». В отличие от других инструментов, *XFE* не показывает эскизов и не предоставляет предпросмотра файлов изображений.



# Поддерживаемые протоколы

## Могут ли эти менеджеры файлов подключаться к удаленным ПК?

Чтобы помочь вам подключиться к разным компьютерам в вашей сети с разными операционными системами или к удаленным компьютерам, менеджеры файлов также поддерживают различные протоколы. А значит, для доступа к удаленным компьютерам не надо устанавливать дополнительных графических программ. Большинство менеджеров файлов в нашем списке поддерживают разные сетевые протоколы. Однако эта функция не предлагается самими инструментами, и вам придется установить

лежащие в основе компоненты самостоятельно. Например, у вас уже должны быть установлены необходимые пакеты для Samba, если в своей сети вы хотите получить доступ к компьютерам Windows. Также *SpaceFM* и *Thunar* для обеспечения поддержки разных протоколов используют *udev* и *gvfs*, поэтому прежде чем тестировать эти менеджеры файлов, обеспечьте установку названных утилит. Помимо использования Samba для подключения к удаленным компьютерам, вы также можете применять наши менеджеры файлов как FTP-клиенты

для доступа к удаленным серверам. Установив требуемые компоненты NFS, можно также монтировать и затем получать доступ к директориям на другом компьютере Linux в сети из удобной среды вашего любимого менеджера файлов. Пользователям, которые всё равно предпочитают протоколы SSH и FISH для безопасной передачи файлов на удаленные компьютеры, это тоже могут. *XFE* — единственный неудачник в этом тесте. Он не поддерживает ни одного протокола и пригоден только для просмотра локального диска и разделов.

## Вердикт

- Dolphin ★★★★★
- Gnome Files ★★★★★
- SpaceFM ★★★★★
- Thunar ★★★★★
- XFE ★★★★★

» Инструменты были равными, показав хорошие результаты при передаче файлов.

# Плагины и расширения

Кто же откажется от дополнительных функций?

**D**olphin не предлагает централизованного списка поддерживаемых плагинов или расширений. Однако магазин KDE предлагает простой доступ к *Dolphin Service Menus* из заголовка KDE App-Addons на боковой панели. Каждое из них добавляет дальнейшие опции в контекстное меню, например, дополнение конвертирования jpg в png, суть которого вполне понятна, и использует *Imagemagick* для конвертирования изо-

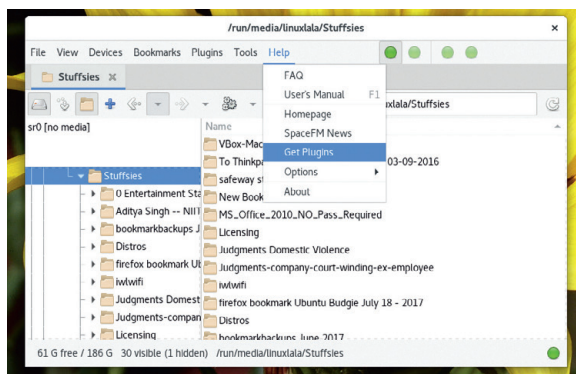
бражений. Есть похожие расширения для конвертирования медиа файлов в разные форматы, экспорта документов *Libreoffice* в PDF, и т. д.

Как и *Dolphin*, *Gnome Files* тоже не предлагает списка плагинов и расширений на сайте проекта или в wiki. Поскольку проект внутри по-прежнему называется своим первоначальным именем, запустите в Fedora команду `dnf search nautilus`, чтобы вывести список плагинов, доступный

в программных репозиториях. Затем можно установить такие полезные расширения, как `gnome-terminal-nautilus` и `nautilus-image-converter`. Похожие расширения помогут вам привязать *Gnome Files* к Dropbox, ownCloud и другим сервисам.

Wiki *SpaceFM* перечисляет все доступные плагины, такие как плагин ClamAV, который дает возможность просканировать выбранные файлы/папки с помощью ClamAV. Другой плагин помогает шифровать/дешифровать выбранные файлы/папки с помощью GPG. Создать плагин для *SpaceFM* весьма несложно, и весь процесс обсуждается весьма подробно в официальной документации.

Ряд плагинов для *Thunar* вы найдете в официальных репозиториях Xfce на Git. Плагины архива добавляют опцию создания архива из контекстного меню. Аналогично, плагин `shares` создает быстрый общий доступ к папкам с помощью Samba. Плагин `media tags` добавляет поддержку ID3/OGG в диалоговое окно группового переименования.



Вы можете получить доступ ко всем плагинам для *SpaceFM*, нажав на **Help > Get plugins** [Справка > Найти плагин].

## Вердикт

- Dolphin ★★★★★
- Gnome Files ★★★★★
- SpaceFM ★★★★★
- Thunar ★★★★★
- XFE ★★★★★

» XFE в этом тесте отстал, не предложив плагинов расширения функциональности.

# Гибкость поиска

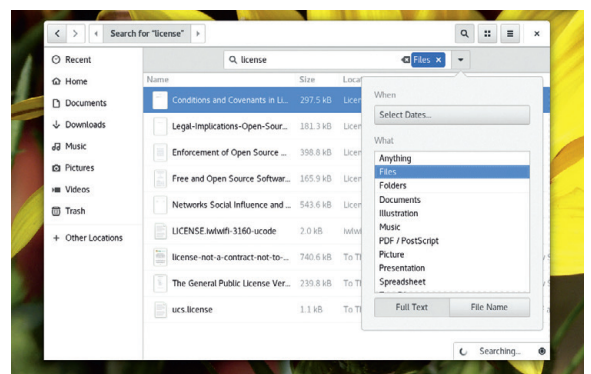
Вот так менеджер файлов создает или разрушает будущее.

**М**етафора двух сторон монеты идеально описывает постоянный рост места на диске. С одной стороны, диски высокой емкости дают возможность сохранять файлы в вашей системе, не перекладывая их на резервную DVD копию. Обратная сторона в том, что, разыскивая файл, вы часто вынуждены продирается через огромное количество директорий. Тем важнее поисковые возможности вашего менеджера файлов.

*Dolphin* для обработки поисковых запросов использует Baloo, среду индексирования и поиска файлов в KDE Plasma. Это означает, что вы должны разрешить Baloo вести активный индекс всех файлов в вашей системе. Если вы этого не сделаете, поиск вернет нулевые результаты, даже если вы точно знаете, что искомый файл существует на диске. Помимо употребления масок имен, можно искать файлы по типу, например, аудио или документ, и сузить результаты на основании рейтинга.

В *Gnome Files* можно ограничить поиск типом файлов. Помимо PDF, изображений, таблиц, презентаций, видео и прочих типов инструмент также позволяет выбрать их впечатляющего списка форматов в алфавитном порядке. Вы также можете задать дату последнего использования или последнего изменения. Для поиска внутри текущей директории инструмент также поддерживает поиск по шаблону. Это означает, что вы можете искать файл с помощью шаблонов, описывающих имя файла, например, `*trip*.png`, что выберет все файлы PNG в текущей директории, у которых в имени файла есть `trip`.

Поиск в *SpaceFM* передается в отдельное окно; доступ к нему можно получить через `File > File Search` [Файл > Поиск Файла]. Хотя поддерживается множество переменных для поиска по шаблону, описанных в руководстве, поиск капризный и непредсказуемый — срабатывает он отнюдь не всегда.



В отличие от других инструментов, *Gnome Files* автоматически выполняет рекурсивный поиск и куда успешнее своих коллег в поиске совпадений.

*XFE* умеет искать в скрытых файлах. Вы также можете ограничить поиск по размеру, типу или дате последнего изменения. Можно даже искать файлы по разрешениям: например, ограничить поиск файлами, созданными пользователем `linuxlala` и с последним изменением за 60 дней ранее с разрешением `0644`. Вы также можете указать, ищите ли вы файл, папку, сокет, ссылку или канал. Поиск допускается также рекурсивный, и можно настроить этот инструмент на употребление символических ссылок.

## Вердикт

- Gnome Files ★★★★★
- XFE ★★★★★
- Dolphin ★★★★★
- SpaceFM ★★★★★
- Thunar ★★★★★

» Подобно *Gnome Files*, *Thunar* поддерживает поиск по шаблону, но на этом и всё.



Менеджеры файлов

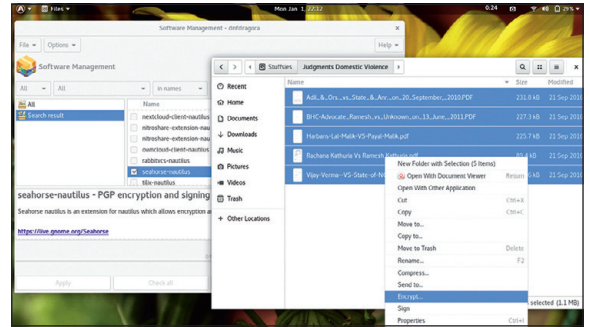
# Вердикт

**Н**а бумаге все инструменты в данном Сравнении хорошо подобраны. XFE, несмотря на то, что это детище разработчика-одиночки, способен потягаться с мощными проектами, поддерживаемыми целой командой разработчиков и обширными сообществами пользователей. XFE не только не хватает плагинов или расширений: он также не поддерживает ни одного сетевого протокола. Это отсутствие поддержки SMB, NFS и прочих протоколов весьма неудачно, и XFE финиширует последним, несмотря на впечатляющие возможности поиска. К сожалению, в Thunar возможностей поиска почти нет, поэтому он не может претендовать на место на пьедестале. Но, хотя Thunar и XFE занимают два последних места, у них есть свои преимущества. Если вы не будете использовать менеджер файлов для подключения к удаленным компьютерам, вас порадуят простота и быстрота XFE.

Хотя SpaceFM пока не поддерживает операций undo/redo, в тесте полезных

функций он получает больше очков, чем Gnome Files, благодаря своей превосходной адресной строке Path Bar, допускающей запуск команд без вызова отдельного терминала. Но ненадежность его поиска и неспособность обеспечить соответствующие опции контекстного меню для всех типов файлов оттягивают его назад.

Dolphin и Gnome Files во всех тестах были примерно одинаковы, за исключением интеграции рабочего стола, где Dolphin превзошел остальных по умолчанию. Но стоит вам установить несколько расширений Gnome Files, как вы получите те же функции, что предлагаются Dolphin. Кроме того, функция undo/redo Gnome Files ограничена в использовании, поскольку она помнит только последнюю операцию,



в отличие от Dolphin, который помнит все операции для открытого в данный момент окна. Будем надеяться, разработчики поработают над этой функцией, применив подход к операциям undo/redo более в стиле Dolphin. Хотя Gnome Files не позволяет добавлять теги, реализация Dolphin кажется не слишком полезной, поскольку нельзя использовать теги как критерий поиска.

Самым важным фактором в определении победителя в этом Сравнении оказалась функция поиска. Gnome Files занимает первое место благодаря скорости, с которой он обрабатывает совпадения в поиске.

» Многие дистрибутивы, такие как Fedora и Ubuntu, обеспечивают ряд полезных расширений для Gnome Files.

**XFE не только не хватает плагинов или расширений: он не поддерживает ни одного сетевого протокола.**

**I Gnome Files** ★★★★★  
 Версия: 3.26.0 Сайт: <http://bit.ly/gnome-files> Лицензия: GPLv3+  
 » Простой интерфейс и впечатляющие функции — первое место.

**IV Thunar** ★★★☆☆  
 Версия: 1.6.13 Сайт: <http://bit.ly/thunar-fm> Лицензия: GPL  
 » Рекомендован тем, кто поклялся никогда не искать файлы.

**II Dolphin** ★★★★★  
 Версия: 17.08.1 Сайт: <https://dolphin.kde.org> Лицензия: GPL  
 » Более быстрый поиск обеспечил бы Dolphin победу.

**V XFE** ★☆☆☆☆  
 Версия: 1.42 Сайт: <http://roland65.free.fr/xfe/> Лицензия: GPL  
 » Идеален, если вам просто нужен менеджер файлов с поиском.

**III SpaceFM** ★★★☆☆  
 Версия: 1.0.5 Сайт: <http://bit.ly/space-fm> Лицензия: GPLv3+  
 » Отлично продуман и настраивается; но нужны тестировщики.

**Обратная связь**  
 Расстроены нашим рейтингом или обижены, что мы проигнорировали ваш любимый менеджер файлов? Шлите свое мнение на [lx.f.letters@futurenet.com](mailto:lx.f.letters@futurenet.com).

## Рассмотрите также...

**В** менеджерах файлов для дистрибутивов Linux недостатка нет. У нас уже были Сравнения, где обсуждались альтернативы в LXF86, LXF119 и не так давно [Ред.: — 7 лет назад!] в LXF143. Инструменты, обсуждавшиеся в данном Сравнении, относятся к категории навигационных менеджеров файлов. Более традиционные

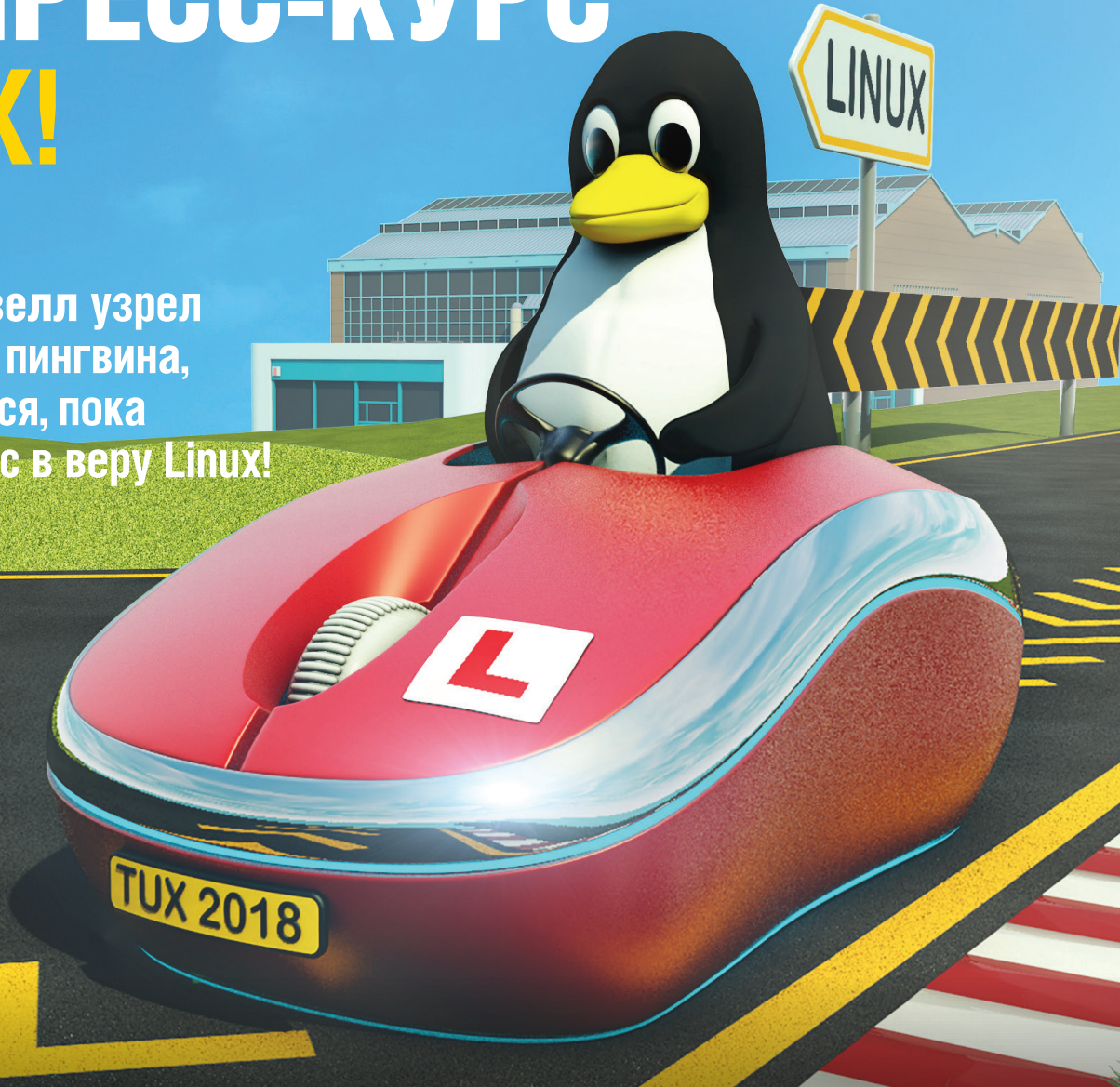
менеджеры файлов, как описанные в LXF119, считаются ортодоксальными. Если вы не против работы в командной строке, попробуйте Midnight Commander. Это превосходный и стабильный инструмент с популярными функциями вроде группового переименования. Если вы любите рабочий стол Mate или Cinnamon, у них есть свои менеджеры

файлов — Caja и Nemo соответственно, сравнимые с Gnome Files. Rox Filer — еще одна альтернатива для рабочего стола Rox. Менеджер файлов Gentoo, предтеча одноименного дистрибутива Linux — еще один стабильный и независимый от рабочего стола выбор, но только если вы не против траты кучи времени на его настройку под свои требования. LXF



# 60-МИНУТНЫЙ ЭКСПРЕСС-КУРС LINUX!

Джонни Бидвелл узрел свет в форме пингвина, и не успокоится, пока не обратит вас в веру Linux!



**Э**й, вы слышали о Linux? Это бесплатная операционная система, которая ничуть не уступит в функциональности любой другой на компьютерном рынке. И она не только бесплатная, но еще и свободная, в том смысле, что ее можно превратить во всё, на что у вас хватит воображения (ну или способностей к кодированию), а также в том смысле, что никто не будет за вами следить, пытаться завалить вас рекламой или всучить ненужное ПО.

Эта свобода касается не только ОС, но и всей программной экосистемы, ее окружающей. Бесплатное, свободное ПО и ПО с открытым кодом [Free, Libre and Open Source Software, FLOSS]

противоположно подходу «запертого сада», практикуемому коммерческими настольными ОС, которые позволяют только устанавливать что-то из магазинов приложений, где как грибы растут всё новые губительные для конфиденциальности функции,

**Весь исходный код Linux может изменить любой, кто пожелает потратить на это свое время.**

и потихоньку отнимают у пользователя контроль, прикрываясь «простотой в использовании».

Существуют буквально тысячи бесплатных и свободных приложений, которые ничуть не хуже, а нередко и лучше своих коммерческих аналогов.

И вам не придется отказываться от новейшего оборудования или даже утрачивать доступ к своей коллекции игр. Однако легко и растеряться, хотя Linux никогда еще не был проще в использовании.

Изучать новый способ работы непросто, и иногда опыт в одной ОС может оказаться минусом при переходе на другую. В основном первые шаги в Linux омрачаются путаницей и замешательством. Часто за ними следуют крики, плач и скрежет зубовой.

И если не проявить осторожности, можно в итоге увязнуть где-то под названием *Vim*. Ужас.

Но это удовольствие мы оставим на потом, и поможем вам освоить мощную elementary OS — один из самых чудесных дистрибутивов Linux.



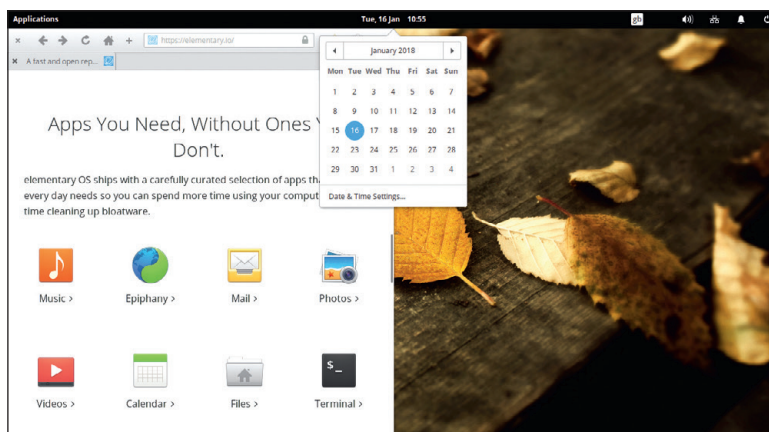
# Прекрасный GNUvый мир...

Мы поможем вам сориентироваться в неизведанной стране GNU/Linux.

**Р**аз уж вы прочитали введение, разумно будет предположить, что вас интересует установка и ознакомление с Linux. Bravo! Возможно, вы слышали о свободе ПО, о борьбе с программными монополиями или о потрясающей мощи и настраиваемости Linux. С тем же успехом вы могли слышать нарекания о разнотыке документации, бубнеж о невозможности сделать какие-то простые вещи без загадочных заклинаний терминала, плюс леденящие душу рассказы о том, как некто застревал в *Vim* (текстовом редакторе) на целые дни. Правда во всем этом есть. Linux может предложить многое, однако и операционные системы, и пользователи бывают ненадежны, и при их столкновении иной раз летят искры (даже и в прямом смысле).

Во врезке внизу вы найдете краткую справку о том, что такое Linux, или читайте статью в Wikipedia. Неплохо знать, откуда мы родом и в чем суть свободного ПО. Обычно люди, говоря «Linux», имеют в виду «настольный дистрибутив Linux», что приводит к путанице. Linux в общем смысле используется повсюду, от крошечных встраиваемых систем до самых мощных суперкомпьютеров мира. Настольные ПК — это, возможно, единственная область, где Linux не удалось добиться господства.

Сегодняшние настольные дистрибутивы Linux предлагают быстрые программы установки, впечатляющие GUI, репозитории с самыми свежими приложениями и менеджерами пакетов для их простой установки. Всё это — отделяемое, и в головах тех, кто пришел с MacOS и Windows, данная идея ущемляется с трудом. Например, необычной выглядит мысль о том, что можно установить новую среду рабочего стола, не затронув настройки основной системы, и затем продолжить штатно использовать все приложения. На самом деле, вся система управления пакетами противится принципу скачивания и запуска некоего случайного исполняемого файла с сайта производителя (или сомнительного зеркала), но об этом мы поговорим позднее. Вы также глазам не поверите, глядя на чуждеса командной строки.



Но мы не настолько глупы, чтобы изображать установку Linux радужной и безоблачной. Всё может пойти плохо, иногда катастрофически плохо. Например, релиз 17.10 Ubuntu пришлось отозвать, потому что он повреждал настройки UEFI определенных ноутбуков Lenovo так, что они переставали загружаться. А заглянув в раздел *Installation & Upgrades* [Установка и обновление] форумов Ubuntu, вы увидите, что как минимум один пост в день рассказывает о попытке установки, по таинственным причинам не удавшейся. Обычно эти истории цепляют взгляд, хотя на каждую неудачную установку приходится сотня прошедших безукоризненно. Люди не очень склонны писать о том, что проходит хорошо — о чем можно лишь пожалеть.

Вы можете попробовать Linux прямо с DVD *Linux Format*, не затрагивая уже установленных у вас ОС. Это будет медленнее, чем при нормальной установке, и вы не сможете сохранить внесенных изменений, но это отличный и, главное, безопасный способ начать знакомство.

» Этот чудесный рабочий стол может стать вашим за время, необходимое на распитие чашечки чая.

»

## Что такое GNU/Linux?

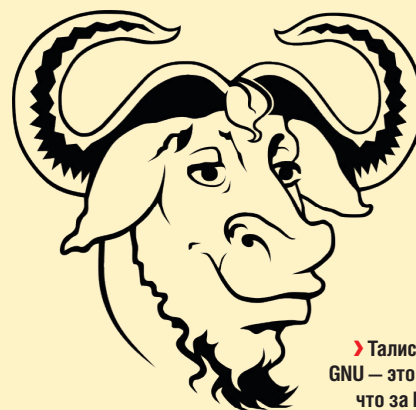
На эту тему можно написать фолианты — и всё равно не дать удовлетворительного ответа. Но если вы хотите поиграть с Linux, было бы полезно узнать, что он собой представляет, и о философии, которая стоит за ним. Поехали (не читайте, если вас раздражает журналистское стремление к упрощительству).

Движение свободного ПО по-настоящему началось с проекта Ричарда Столлмена [Richard Stallman] GNU Project, целью которого было гарантировать пользователям свободу использовать, делиться, изучать и изменять ПО, работающее на его компьютере. С этой целью создали UNIX-подобную ОС под названием GNU (рекурсивный акроним GNU's Not Unix). Многие популярные инструменты UNIX были портированы в GNU. Проект GNU также закрепил свои идеи в лицензии, GNU GPL (General Public License), содержавшей важнейшее условие «копилефта»: все производные работы должны

быть выпущены под той же лицензией. Были также написаны новые программы, такие как GNU C Compiler и GNU Emacs, которые стали обязательными для всех современных дистрибутивов Linux и споров о текстовых редакторах соответственно.

Несмотря на весь этот прогресс, одна вещь осталась незаконченной (и до сего дня остается незаконченной, хотя не заброшенной), и это — ядро. Ядро — это самая крутая часть ОС, которая обращается к оборудованию, управляет памятью и вообще занимается всеми сложностями, о которых вам, обычным пользователям, никогда в принципе не приходится задумываться. Писать ядро сложно, но в 1991 г. молодой студент по имени Линус Торвалдс вышел в Usenet, чтобы сообщить, что именно этим он и занялся. В результате соединения ядра «Linux» с портированными в него GNU Tools родилась новая ОС. Она была выпущена под GPL в 1992 г., и вскоре последовали

первые «дистрибутивы» (тогда это было просто ядро с некоторыми программами, документацией и сырым механизмом установки).



» Талисман GNU — это Гну, что за Гну?

# Установка elementary OS

Пусть ваши друзья завидуют одному из прекраснейших дистрибутивов.

**Н**ас часто попрекают за чрезмерное Ubuntu-любие. Упрек справедлив, ведь есть сотни других дистрибутивов. Но Ubuntu использует большинство (как минимум по части настольного Linux). Месяц назад мы немного сошли с нахоженного пути, рассказав о Linux Mint в нашей статье «Строим Linux-ПК».

➤ Желая отвести на elementary OS целый диск, выберите опцию Erase disk [Очистить диск]. На этом экране вы можете выбрать, какой диск очистить.

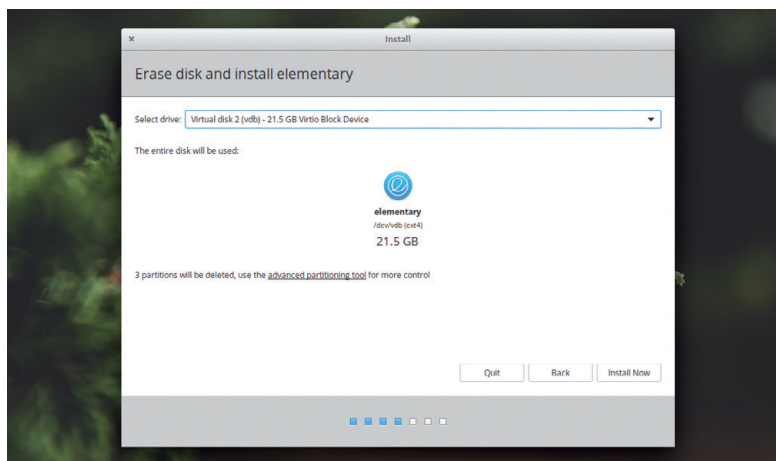
Mint — производное Ubuntu, отличающееся от своего родителя применением более традиционного рабочего стола Cinnamon. Внутри всё практически то же самое. На сей раз мы будем использовать elementary OS, еще одно производное Ubuntu, с другим рабочим столом. И elementary, и Mint основаны на Ubuntu 16.04 LTS, поэтому они унаследовали его стабильность и так же быстро получают его обновления безопасности.

Технически elementary пока на бета-стадии, но пусть это вас не пугает. Мы очень пристально следили за этой разработкой последние пару лет, и были весьма впечатлены. Конечно, главное в Linux — это выбор, поэтому можете спокойно выбирать между Ubuntu и Mint, или любым другим дистрибутивом, который вам по душе. Кое-что в нашем руководстве будет относиться исключительно к elementary, но большая часть применима к любому дистрибутиву на основе Ubuntu, а дополнительные инструменты, о которых мы расскажем позднее, доступны для любой разновидности Linux.

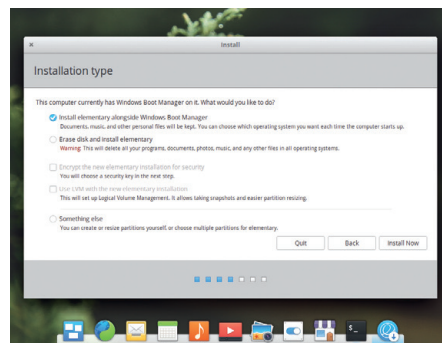
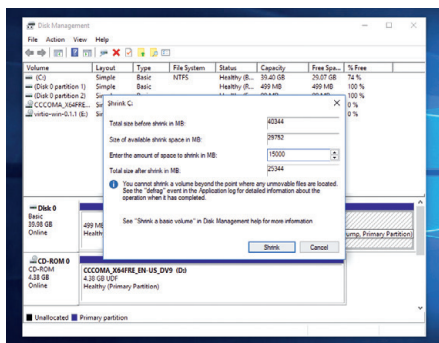
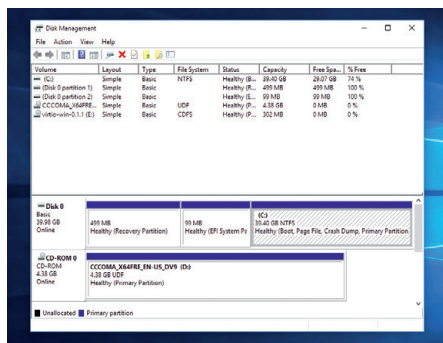
Рабочий стол elementary, *Pantheon*, часто называют похожим на MacOS, что в некоторых кругах служит комплиментом, но мы считаем, что он достоин уважения и как таковой. Безусловно, панель запуска внизу и многопанельный менеджер файлов сильно напоминают яблочную ОС, однако *Pantheon* обладает собственными достоинствами. Помимо привлекательного вида, elementary предлагает несколько уникальных функций, благодаря чему и достоин вашего внимания. Мы расскажем о них подробнее, когда сделаем установку.

## Дуэль ОС

Сейчас самое время сделать резервные копии всех важных файлов на вашем компьютере. Если вы настраиваете двойную загрузку, скажем, с Windows, то вряд ли установка Linux как-то повредит ваши разделы Windows. Однако нельзя сказать, что это дело несложное, так что не стоит рисковать, если ваш компьютер используется для важной работы. Представьте, что будет, если рухнет ваша Windows? У вас есть средства для переустановки? Есть лицензионные ключи для ОС и других программ? Если у вас есть запасной SSD



## Освободите место для Linux



### 1 Управление диском

Загрузитесь в Windows [Ред.: — Странное начало для статьи о Linux]. После множества обновлений, уведомлений и вынужденных перезапусков [Ред.: — Ой, горит] откройте меню Пуск, начните вводить disk и выберите Create and format hard disk partitions [Создать и отформатировать разделы жесткого диска]. Откроется консоль Disk Management [Управление диском], где можно изучить структуру разделов любого установленного диска.

### 2 Сжатие имеющегося раздела

Очень важно не затрагивать EFI или разделы восстановления — это способно закончиться катастрофой. Выберите раздел, где установлена Windows (вероятно, C:), щелкните по нему правой кнопкой и выберите Shrink Volume [Сжать раздел]... Для установки elementary нужны минимум 10 ГБ, но мы рекомендуем 30 ГБ, если вы можете выделить столько, или еще больше, если вам нужно. Определите объем и нажмите Shrink [Сжать].

### 3 Новый раздел

Следуйте тексту выше для установки Linux вплоть до экрана Installation type [Тип установки]. Выберите опцию Install alongside Windows Boot Manager [Установить вместе с менеджером загрузки Windows]. Если вы используете современную систему UEFI, то вам, возможно, понадобится изменить настройки, чтобы загрузить Ubuntu вместо Windows. В меню Grub (загрузчик elementary) будет опция загрузки Windows, если вы вдруг решите туда вернуться.



или жесткий диск, подумайте о том, чтобы для первой установки использовать его, обходя возможность войны за сферы влияния. Еще лучше, если у вас есть запасной компьютер, пусть даже старый; тогда стоит сделать вашу первую установку именно на него.

Позабывшись о предупреждениях и полагая, что вы всё еще хотите сделать двойную загрузку, предлагаем последовать нашему пошаговому руководству (внизу стр. 28). Учтите, что эти шаги нужны только если Linux ставится на тот же диск, что и Windows. Если нет, можете устанавливать прямо с *Linux Format DVD*.

## Поддержка сети

Загляните на страницы DVD и в FAQ (<https://linuxformat.com/dvd-support>), если у вас будут затруднения с тем, чтобы диск заработал. Вам следует отключить Secure Boot, чтобы он загрузился, и, возможно, придется дать команду своему BIOS/UEFI загрузиться с DVD, а не с жесткого диска. Это не входит в тематику данной статьи, но в FAQ есть ряд советов. Если на вашем целевом компьютере нет оптического привода, можете скачать ISO с <https://elementary.io> (пожертвование опционально) и следуйте инструкциям на <https://elementary.io/docs/installation> для записи его на USB-брелок.

Для первой успешной загрузки выберите Try elementary OS without installing [Попробовать elementary OS без установки]. Аннотация ниже поможет вам сориентироваться после загрузки. Исследуйте установленные приложения, и пусть вас не смущает медленная скорость; когда вы будете готовы, выберите Install elementary OS [Установить elementary OS] в меню Applications [Приложения].

Во время установки вам зададут несколько вопросов. Если вы подключены к Интернету, то надо будет отметить окно Download updates while installing... [Скачать обновления при установке...]. Основная причина отметить окно Install third-party software [Установить сторонние программы] — заставить работать вашу беспроводную карту, и если вы сомневаетесь, лучше отметить это окно тоже. Выбирайте Erase disk and install elementary [Очистить

диск и установить elementary] только если вы уверены, что на целевом диске ничего нет. До того, как произойдет нечто деструктивное, вы увидите экран подтверждения, чем вы и воспользуетесь, чтобы убедиться в правильности выбора диска.

Затем у вас добудут информацию по локализации и попросят настроить учетную запись пользователя и пароль. Неплохо бы отметить окно Encrypt my home folder [Шифровать мою домашнюю папку], если вы планируете хранить на этом компьютере важную информацию. Это не позволит прочитать данные в случае утраты жесткого диска, но не защитит от доступа к нему вредоносного ПО, когда вы будете входить в систему. Это также означает, что если вы забудете пароль, ваши данные будут всё равно что утрачены. Поэтому запомните пароль хорошенько. Заварите себе чашечку чая и терпеливо ожидайте окончания установки. Нажмите Restart Now [Перезапустить сейчас], удалите средство установки, когда

## Если у вас есть запасной компьютер, стоит сделать первую установку именно на него.

вам предложат это сделать, и нажмите Enter. Если что-то пойдет не так, вам, возможно, придется храбро взяться за настройки UEFI. Многие реализации грешат ошибками и просчетами, но у вас должно получиться загрузиться оттуда вручную.

Если всё хорошо, то через несколько минут созерцания логотипа elementary OS вы увидите экран приглашения. Введите здесь аутентификационную информацию, и вы окажетесь в среде, удивительно похожей на среду live, в которую мы загрузились изначально, и на аннотацию ниже. Поосмотритесь, а потом продолжайте чтение, чтобы найти другие полезные советы и руководства, как извлечь максимум из elementary OS.



## Сориентируйтесь в elementary

### Приложения

Все графические приложения находятся здесь. Вы можете использовать сочетание клавиш Windows и пробела (или Alt+F2), чтобы сэкономить на перемещениях курсора мыши. Некоторые пользователи замечают эту комбинацию только на клавишу Windows, что нормально для Windows, Unity, Gnome и KDE.

### Индикаторы

Здесь можно получить доступ к управлению звуком и сетью, опциям уведомлений и выключения. Область уведомлений сообщит вам об Обновлениях системы [System Updates], приближающихся событиях в календаре и т. д.

### Многозадачный вид

Откройте несколько приложений. Нажмите на эту кнопку, затем добавьте новый рабочий стол и откройте несколько приложений там. Это может улучшить вашу работу, привязывая определенные действия к определенным рабочим столам.

### Настройки

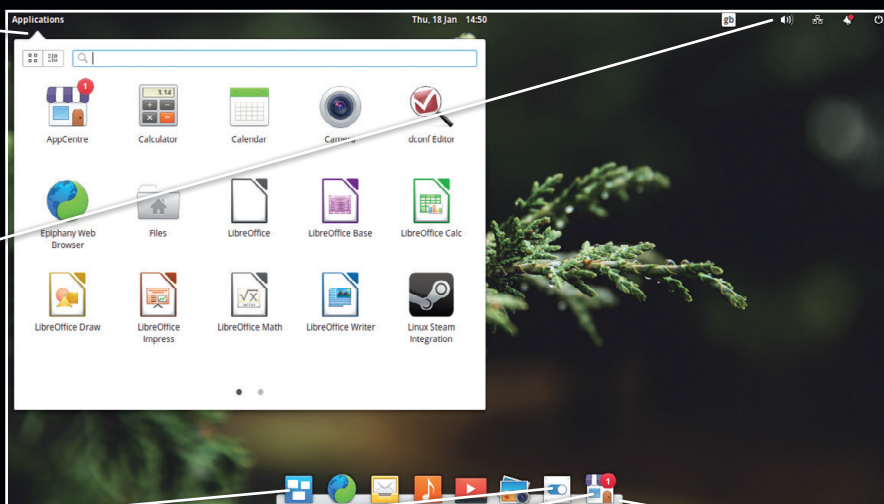
Управляйте всеми возможными настройками из одного удобного пункта. Приложения по умолчанию, языки, родительский контроль и многое другое — обо всем этом можно позаботиться отсюда.

### AppCentre

Это простейший способ добавить или удалить приложения в elementary OS. Нажмите на значок, чтобы запустить его, и вы увидите подборку приложений, включая специально подстроенные под elementary OS.

### Панель запуска Plank

Plank размещает ярлыки для часто используемых приложений, и также работает, как док для запущенных в данный момент приложений (обозначаются голубой точкой под их значком).



# Исследуем elementary OS

Внимательно изучим массу программ для работы, игр и общего развития.

**Ч**его вы точно не сумеете найти в Linux, это раздутого программного обеспечения. В Windows такое обычно добавляют производители и OEM, или оно поставляется в виде ненужных дополнений с драйверами устройств. Те немногие производители оборудования, которые ставят Linux на своих компьютерах, понимают, что учинять подобное безобразие незачем, и поскольку большинство драйверов включены в ядро Linux, им никак не нужны излишества.

Даже по сравнению с другими дистрибутивами Linux, приложений в elementary OS довольно мало. Большинство тех, что включены в установку по умолчанию, предназначены только для elementary, поэтому они хорошо интегрируются с минимализмом рабочего стола *Pantheon*. Пусть вас не страшит отсутствие там меню и опций. Эти приложения, в частности, менеджер файлов, разработаны для обеспечения мощных функций без необходимости в сложной настройке. *Scratch*, текстовый редактор (который

в следующем релизе будет переименован в *Code*), запоминает ваши вкладки и автоматически сохраняет документы. Даже те, которые пока безымянны.

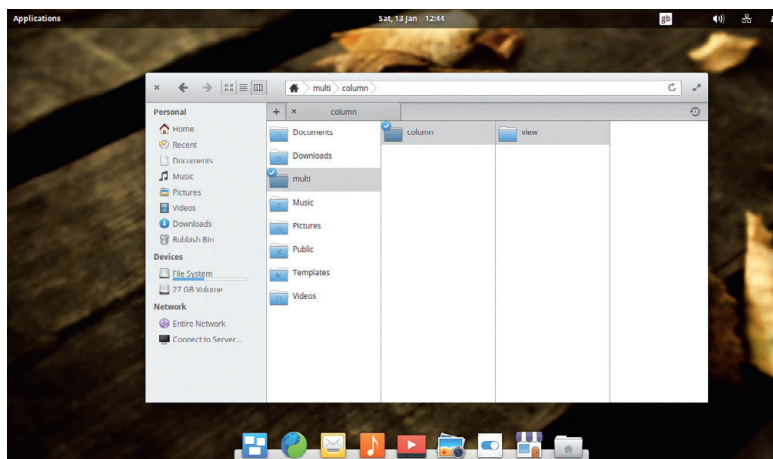
Под завязку наполненные приложениями дистрибутивы могут запутать новичка. Лучше начать с самых основ (некоторые люди вполне обходятся браузером и музыкальным плеером) и добавлять всё остальное по мере необходимости. Если ваша жизнь крутится вокруг документов, электронных таблиц и презентаций (или как минимум предполагает постоянную работу с ними), то вам понадобится офисный пакет. В Linux есть несколько опций, но самая популярная — превосходный *LibreOffice*. Мы могли бы установить его из AppCentre (хотя здесь, похоже, налицо давняя проблема, которая это усложняет, см. <http://bit.ly/libreoffice-wont-open>); но сейчас самое время познакомиться с командной строкой.

В меню приложений нажмите Terminal. Вас поприветствует ожидаемое и смутно пугающее мерцание курсора. Именно здесь любят работать матерые линуксоиды. Когда вы попривыкнете, терминал станет невероятно эффективным способом работы. Вы можете воспроизводить музыку, проверять почту или писать скрипты для автоматизации скучных задач. На самом деле практически всё, что можно сделать из GUI, можно сделать и из командной строки. Мы будем использовать ее для установки *LibreOffice*, что делается посредством запуска

```
$ sudo apt install libreoffice
```

## SuperUsers всю дорогу

Прежде чем мы поговорим о жутком выводе на экран, давайте разберем по частям команду выше. Часть `sudo` указывает, что всё следующее запустится от имени суперпользователя-root (аналог учетной записи администратора в других ОС), это сокращение от (SuperUser-do). Когда бы вы ни увидели команду, начинающуюся с `sudo`, хорошенько осознайте, что именно она делает, поскольку она потенциально способна повредить вашу систему. Запуск команд от имени обычного пользователя может причинить вред только файлам этого пользователя.



➤ Для более эффективного взаимодействия с содержимым папок активируйте отображение в несколько столбцов, по кнопке слева от строки местоположения.

## Есть жизнь и в 32-битной технологии

Одно из часто упоминаемых преимуществ Linux в том, что он работает со старым оборудованием. Это правда, но есть и ограничения. Например, ядро Linux перестало поддерживать процессор Intel 386 (ту самую архитектуру, на которой работала первая сборка Linux Торвальдса) в конце 2012 г.

Вдохнуть жизнь в старое оборудование — хорошая идея, но требования, предъявляемые к такому компьютеру «повседневной работой», легко недооценить. Большинство дистрибутивов отказались от поддержки x86 CPU, которым не хватало управления памятью PAE (представленным в Pentium Pro в 1995 г.), несмотря на то, что ядро всё еще их поддерживает. Поэтому хотя пока что использовать 486 с Linux можно, но ваша работа будет урезана.

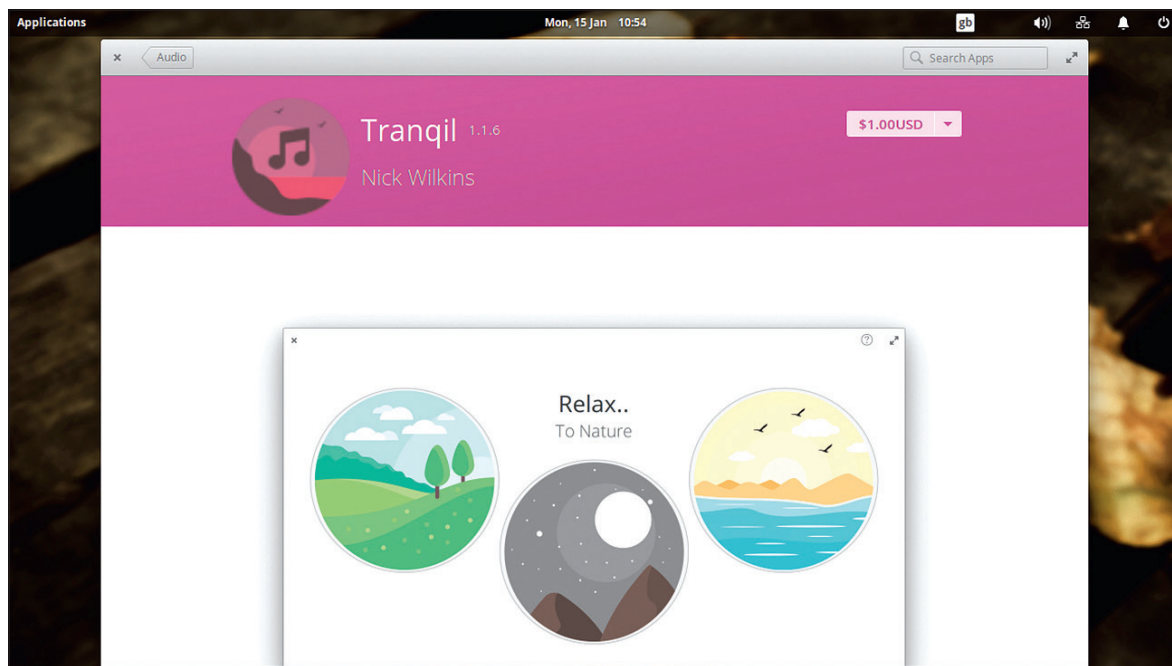
Не так давно дистрибутивы начали обсуждать сокращение поддержки и для 32-битных x86. Это уже произошло в Arch Linux (хотя поддерживаемое сообществом ответвление живет и здравствует), Fedora сейчас предлагает только минимальное 32-битное средство установки, а настольный Ubuntu сделает то же самое в грядущем релизе 18.04.

И хотя спрос затухает, некоторые спецдистрибутивы продолжают поддерживать 32-битные x86 установки. Один из них — превосходная Легаси-редакция Bodhi Linux, которая тоже есть на DVD Linux Format. Как и elementary, она основана на Ubuntu 16.04, и несмотря на очевидные косметические отличия, процедура установки и структура системы одинаковые.



➤ Легковесный Bodhi внутри практически идентичен elementary OS.





Вы найдете индивидуальные приложения elementary OS, такие как *Tranquil* [sic] в AppCentre. Вы не обязаны платить, но спокойствие [англ. Tranquility] бесценно.

*Apt* — это менеджер пакетов, применяемый дистрибутивами на базе Debian. Программы в Linux распространяются в виде пакетов, получаемых из репозитория, которыми занимаются команды дистрибутива. Производители пакетов берут программы от самих разработчиков приложений [upstream] и организуют их так, чтобы они ладили с их дистрибутивом. Новые технологии, типа Snaps и Flatpaks (см. стр. 40), работают по-другому и позволяют разработчикам приложений распространять независимые от дистрибутивов пакеты, но сейчас об этом беспокоиться незачем (мы поговорим о Snaps позже). В целях безопасности и целостности пакеты подписываются и снабжаются контрольной суммой, и их легко удалить без всякой низкопробной утилиты удаления. С остальной частью команды всё ясно: мы распоряжаемся, чтобы *Apt* установил пакет *libreoffice*.

А теперь — к тому самому пресловутому жуткому выводу на экран. *LibreOffice* — большая программа, и зависит от многих штук (среда выполнения Java, шрифты, всякие библиотеки), которые тоже поставляются как пакеты и называются зависимостями. Отдельные компоненты *LibreOffice* также имеют собственные пакеты с именами *libreoffice-writer*, *libreoffice-calc* и т. д. *Apt* попросит нас подтвердить, что мы и вправду хотим скачать и установить всё это, и сообщает нам, сколько на всё это уйдет места (450 МБ, когда мы попробовали).

Поскольку мы живем во имя делопроизводства, давайте продолжим установку, нажав Enter (заглавная Y обозначает, что это по умолчанию; можете ввести у и нажать Enter, если хотите). После нескольких разноцветных строк состояния с рендерингом в ASCII процесс завершится, и заглянув в меню Applications [Приложения], вы найдете там ярлыки для разных граней пакета *LibreOffice*. Запуск *Writer* может вызвать воспоминания о *Microsoft Word* примерно 2000 г., но если опустить эти мелочи, вы обнаружите текстовый редактор, более чем функциональный для работы с вашими символами и абзацами.

## Вернемся в AppCentre

Хотя установка программ из *AppCentre* с виду резко отличается от установки пакетов из командной строки, за сценой именно это и прodelывает *AppCentre*. За очень небольшим исключением, почти всё, что можно сделать с помощью мыши и приложения GUI, можно сделать из командной строки. Например, elementary

OS периодически будет проверять наличие обновлений и (очень вежливо) уведомлять вас об их наличии. На самом же деле происходит то, что команда (или как минимум одна идентичная ей функция) `$ sudo apt update`

запускается в фоновом режиме. Она обновляет кэш *Apt* для доступных пакетов.

По завершении все обновления будут применены с помощью команды

```
$ sudo apt upgrade
```

По мере своего исследования Linux вы найдете множество примеров, начинающихся с этой магической формулы. Если кэш пакета устаревает, вы будете видеть ошибки, когда *Apt* начнет попытки найти устаревшие пакеты, которых больше нет в репозиториях. Наша долгоиграющая и продолжающаяся серия про Терминал (доступная для подписчиков в Archive *Linux Format*) расскажет о командной строке более подробно, но сейчас мы упомянем только

## Терминал позволяет воспроизводить музыку, проверять почту или писать скрипты.

одну команду: `man`. Это сокращение от manual [руководство] — его-то часто цитируемый акроним RTFM и советует вам прочитать. Например, если вы хотите узнать подробности о команде `ls` для вывода списка содержимого директории (в просторечии именуемой «папкой»), наберите `man ls`.

Elementary поставляется с браузером *Epiphany* рабочего стола Gnome (ныне известным просто как Web). Он более чем функционален и работает на WebKitGTK, поэтому соответствует всем самым последним web-стандартам. Как бы то ни было, вы, возможно, предпочтете *Firefox* (сейчас он по умолчанию поддерживает Netflix и другие потоковые сервисы), который вы найдете в *AppCentre*, либо Google *Chrome*, которого там нет. Зато там вы найдете *Chromium* — версию *Chrome* с открытым кодом, где нет всех этих страшных штук и ограничивающей политики распространения. А если вам всё-таки нужен необезжиренный *Chrome*, вы найдете инструкции по установке на [www.google.com/chrome/browser/desktop/index.html](http://www.google.com/chrome/browser/desktop/index.html).

# Получение помощи

Несмысленным побывал любой профи — не стесняйтесь, если зашли в тупик.

**Н**еприятности случаются, и в целом сообщество Linux очень дружелюбное и готово помочь. Но вы многое можете сделать, чтобы помочь им помочь вам. Прежде всего помните, что это — бесплатное ПО. Вы не платили за круглосуточную поддержку, так что будьте благодарны, если кто-то пытается вам помочь. Разглагольствования на тему «Linux не готов к повсеместному использованию» обычно не приветствуются и отнюдь не помогают.

Возможно, таково ваше мнение или ваш опыт, но сотни тысяч довольных пользователей настольного Linux, вероятно, с этим не согласятся. Высказав это мнение, потом удалить его — неплохой способ успокоиться (и не чувствовать себя глупо на следующий день). Одна из радостей Linux — решение проблем, хотя в самом начале трудно понять, с чего начинать. Для elementary OS хорошим началом станет сайт поддержки (<https://elementary.io/support>).

При удачном отметении проблем установки, первым испытанием может стать работа с беспроводным соединением (возможно, вы окажетесь одним из везучиков с чипсетом, который заработает сразу же; тогда поделитесь своими идеями с менее удачливыми). По большей части все горести беспроводного соединения вызваны тем фактом, что код, работающий на беспроводных радиочипах, проприетарный и не может распространяться. Но это не означает незаконности приобретения его для себя.

## Решите свои беспроводные горести

Первый шаг — выяснить, с каким чипсетом мы имеем дело. Номер модели и сборка здесь не сильно помогут, потому что у производителей есть привычка менять оборудование между пересмотрами. Команда терминала `lspci` сообщит нам обо всех устройствах

► Ubuntu Forums на [ubuntuforums.org](http://ubuntuforums.org) — отличное место, чтобы найти ответы. И Bodhi, и elementary внутри используют Ubuntu.



## Ошибки vs «Вы не умеете этим пользоваться»

Иногда Linux ведет себя странно — настолько странно, на самом деле, что вы вполне можете задаться вопросом, не является ли подобное поведение намеренным. Например, вы можете заинтересоваться, почему нельзя использовать мышку для перемещения курсора в текстовом редакторе для терминала *Nano*, или почему обычно используемые для копирования и вставки клавиши (Ctrl+C, и т.д.) там не работают.

Эти примеры, естественно, являются намеренными. Для консольных приложений нет ввода мыши (по крайней мере, не с рабочего стола), и Ctrl+C служит очень важной функции отправки сигнала

SIGINT (user interrupt — прерывание пользователя) в программы консоли (что вынуждает их элегантно закрыться). Возможно, вам также не понравится, что на рабочем столе elementary нельзя размещать значки или даже щелкать по рабочему столу правой кнопкой (т.е. щелкать-то, конечно, можно, да только это ничего не даст). Это тоже особенность, заложенная в дизайн.

Умение отличать эти странности от непреднамеренного поведения — дело опыта. Есть множество случаев, когда вам надо настроить перспективу или некую запутанную настройку, если вы не можете достичь своей цели.

Но и ошибки тоже прокрадываются, и одна из радостей программ с открытым кодом в том, что процессы борьбы с ними прозрачны. Мы бы посоветовали воздержаться от сообщений об ошибках, пока вы не разберетесь, какие процессы задействованы (сбор соответствующих частей системных журналов, работа с заплатками, компиляция кода, отслеживание прошлой деятельности). Но обязательно загляните в bugtracker Ubuntu, чтобы увидеть, как это всё происходит. Например, выясните подробности о прошивке Ubuntu 17.10, упомянутой ранее, на <https://bugs.launchpad.net/ubuntu/+source/linux/+bug/1734147>.



PCI, подключенных к нашей системе (lspci для устройств USB). Эта команда выдает длинный результат, поэтому мы отфильтруем его с помощью grep, чтобы нам сообщали только о сетевых устройствах. Откройте терминал и запустите:

```
lspci -nnk | grep -iA2 net
```

Часть -nn велит lspci выдать коды производителя и устройства, что выглядит как 1814:3090 и уникальным образом идентифицирует оборудование PCI. Часть k велит вывести информацию обо всех драйверах ядра, способных работать с устройством. Мы «передаем» (с помощью символа |) вывод grep, чтобы ограничить его только записями, содержащими текст net. Используйте man grep, чтобы больше узнать о множестве параметров grep. Возможно, вы увидите информацию о своих проводных и беспроводных интерфейсах, но должно быть совершенно понятно, где какой. В нашем случае соответствующий результат был

```
02:00.0 Network controller [0280]: Ralink corp. RT3090 Wireless 802.11n 1T/1R PCIe [1814:3090]
```

```
Subsystem: Lite-On Communications Inc RT3090 Wireless 802.11n 1T/1R PCIe [11ad:6622]
```

```
Kernel driver in use: rt2800pci
```

Эта карта работает сразу, но будь это иначе, мы бы старательно искали (на форумах Ubuntu и elementary) по ключевым словам rt2800pci (это драйвер) и 1814:3090 (коды устройств). Отсутствие доступной прошивки не остановит загрузку драйвера (хотя и не даст ему сделать ничего полезного). Если результат для вашего беспроводного устройства не сообщает об используемом драйвере ядра, есть вероятность, что у вас другая проблема — будем надеяться, такую, которую поможет решить вышеописанное расследование. Неплохо начать с раздела Wireless hardware [Беспроводное оборудование] Ubuntu Wiki на <https://help.ubuntu.com/community/HardwareSupportComponentsWirelessNetworkCardsRealTek>.

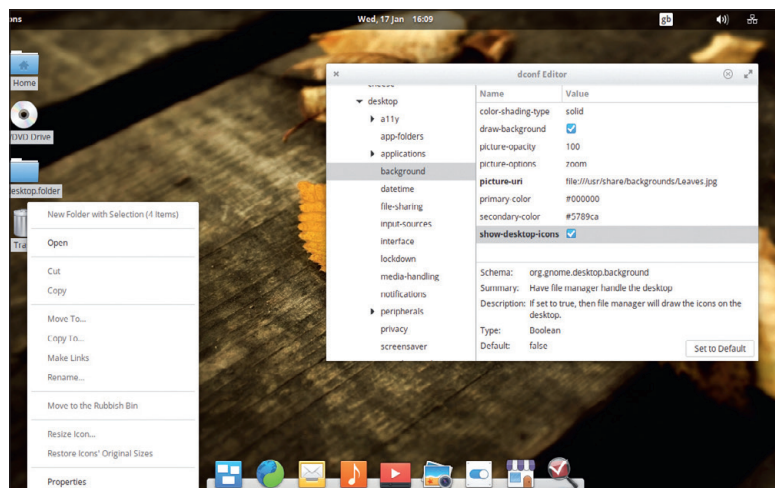
## Найти помощь... быстро

Непослушные беспроводные драйверы — далеко не единственное, что затрудняет людям первые шаги в Linux, но какой бы ни была ваша проблема, есть несколько стандартных шагов, которые стоит проделать, прежде чем обращаться за помощью к сообществу. Возможно, Linux менее популярен, чем другие настольные операционные системы, но достаточно популярен для предположения, что кто-то уже сталкивался с такой же проблемой.

Во-первых, уделите некоторое время поиску по форумам и Сети определенного сообщения об ошибке или оборудовании, которое может быть причиной. Во-вторых, не надо слепо копировать и вставлять команды терминала из разных уголков Интернета. Сообщения об ошибках бывают не так просты: одно и то же решение может оказаться катастрофой в одном случае и безобидным в другом. Попытка решить несуществующую проблему вместо реальной обычно создает еще две, а то и более проблем. Поэтому не следуйте чьему-то решению, если вы не уверены, что вы находитесь в такой же ситуации, и проверьте версии релизов.

Читайте ветви по поддержке полностью. К несчастью, появился целый класс участников форумов, которые выкладывают разнообразные и витиеватые «решения» проблем. Иногда у них благие намерения, и предложенные решения могут даже сработать, но часто это просто люди, жаждущие показать свои невероятно «крутые» знания Linux, и они недооценивают или запутывают размещившего вопрос. Упомянутые знания, как правило, оказываются малопригодными, когда, например, предложенное решение рухнет после обновления системы.

Если только вы не пытаетесь сделать что-то сверхсложное или если всё не пошло катастрофически плохо, требуемое вам решение должно быть довольно простым. Проверьте даты постов, на которые вы наткнулись. Вряд ли решение 2003 г. будет применимо сегодня.



» Отсутствие в elementary опции включения значков на рабочем столе — это не ошибка, но ищите решение — и обряцете.

К сожалению, некоторые из наиболее общих проблем у новичков очень страшные: система скидывает вас в оболочку восстановления Grub или начинает выдавать текст со скоростью, превышающей человеческие возможности восприятия, а потом выдает черный экран.

Super Grub2 Disc ([www.supergrubdisk.org](http://www.supergrubdisk.org)) — диск аварийного восстановления, который поможет решить ваши проблемы с загрузчиком. Зловещий черный экран — обычно признак того, что плохо работают графические драйверы (почти всегда это проприетарные драйверы Nvidia, но иногда виновато старое оборудование или оборудование с причудами). Если вам повезет, вы должны войти в систему, переключившись в консоль (противоположность графическому менеджеру отображения), нажав на Ctrl+Alt+F1. Отсюда вы должны изучить файл X.org с помощью less /var/log/Xorg.0.log на предмет хоть каких-то подсказок (проматывайте его с помощью клавиш со стрелками, а для выхода жмите Q).

## Не надо слепо копировать и вставлять команды из разных уголков Интернета.

Более общий вариант — заглянуть в системный журнал и посмотреть там текущую загрузку, скомандовав journalctl -b. Если что-то пошло не так и вам пришлось перезагрузиться, то вы сможете увидеть сообщения предыдущей загрузки с помощью journalctl -b -1. Системный журнал собирает самую разную информацию: часть просто интересна для чтения, но в основном это компьютерный эквивалент местной болтовни. Записи журнала делятся на восемь уровней, пронумерованных от нуля до семи (программисты любят начинать нумеровать всё с нуля, к большой досаде нормальных людей), где ноль — самый сильный («опасность»), а семь — самый слабый («дебаг»; программисты еще и любят нумерацию в обратном порядке). Большая часть записей журнала выше уровня три («ошибка») не относятся к тому, что пошло не так.

Чтобы увидеть только самые важные сообщения, используйте параметр -p (priority — приоритет), например, journalctl -b -p 3. Информация, которую вы там найдете, может как минимум помочь вам — или добрым волшебникам с форума — локализовать проблему. Не бойтесь просить о помощи, но отдавайте себе отчет в том, что вы делаете. Вот пост на Ubuntu Forums, охватывающий основную информацию: <https://ubuntuforums.org/showthread.php?t=1422475>.

# Сделайте больше с elementary

Копать глубже в надежде решить проблемы — достойное предприятие...

**Б**удем надеяться, мы дали вам неплохое представление об основах elementary OS и немного прояснили, как работает Linux в целом. Elementary, вероятно, идет дальше других дистрибутивов в плане полноценного удобства пользователя без обращения к командной строке. И это делает его идеальным для новичков... но и скрывает множество функций.

Например, есть отличные штуки в *AppCentre*, но они представляют собой лишь крошечную часть того, что имеется в репозиториях Ubuntu. Если бы всё это было видно в одной и той же структуре, никто в здравом уме не стал бы пытаться прокрутить этот список вниз, поэтому мы видим только то, что было тщательно отобрано. Если вы и правда хотите увидеть всю подборку пакетов в графическом приложении, установите *Synaptic*: он покажет вам всю правду о пакетах во всей их неизреченной славе. Однако есть нечто более жизненное, чем изучать массивную подборку пакетов Ubuntu. Возможно, вы там и сям искали возможность переместить кнопку Close вправо, или, возможно, включить кнопку Minimise. Эти и прочие кажущиеся незбылемыми elementary'змы вполне можно

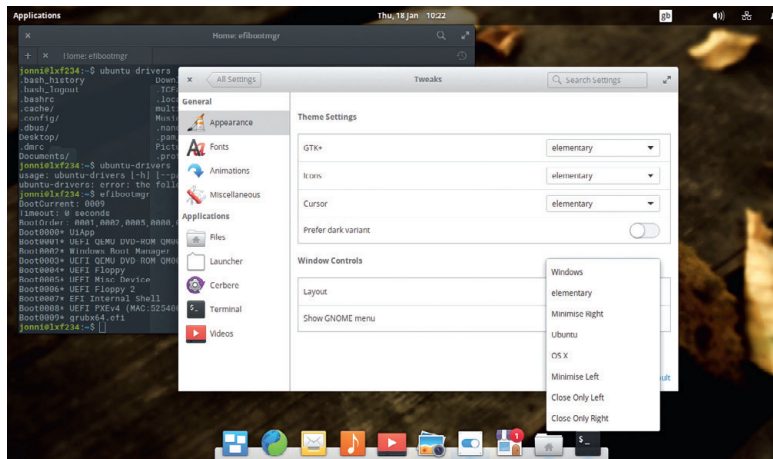
сделать выполнимыми с помощью инструмента *elementary Tweaks*, который вы не найдете в *AppCentre*.

## Elementary, дорогой Ватсон

Одной из прелестей программ с открытым кодом является то, что вы можете взять чей-то код, немного его подправить, самостоятельно скомпилировать и включить в пакеты для собственного дистрибутива. Одна из проблем в том, что этот процесс бывает довольно напряженным и затратным по времени, и часто его приходится повторять каждый раз при обновлении инструмента или любой из его зависимостей.

Ubuntu ввел идею Personal Package Archives (PPAs), которые позволяют разработчикам размещать свои пакеты, не проходя через строгие проверки, необходимые для включения в официальные репозитории. Пользователи могут решить добавить определенный PPA и тем самым отступить от вырывания волос, традиционно ассоциирующегося с развертыванием пакетов. Единственная оговорка, что PPA должен быть доверенным, потому что иногда их создают злоумышленники и размещают там всякие гадости. Так что будьте осторожны и не добавляйте их, пока не узнаете о них подробнее.

➤ **Elementary Tweaks** дает возможность изменить набор инструментов и темы значков, а также разместить кнопки управления окнами по своему желанию.



Снова запустите терминал и передайте ему эту серию команд:

```
sudo apt-get install software-properties-common
sudo add-apt-repository ppa:philip.scott/elementary-tweaks
sudo apt-get update
sudo apt-get install elementary-tweaks
```

Теперь откройте System Settings, и — о чудо! — вы увидите новый апплет Tweaks, где кнопки управления окон и прочее можно перемещать по вашему желанию.

В награду за то, что вы дочитали эту статью до конца, мы оставим вам несколько подсказок, как улучшить ваш игровой опыт с помощью *Steam*. Надеемся, вам понравилось, и надеемся, вам понравится elementary OS. Следите за блогом команды elementary на <https://medium.com/elementaryos> и отправляйте Джонни ваши elementary-вопросы на [lxformat@futurenet.com](mailto:lxformat@futurenet.com). LXF

## Установка Steam

Игровая сцена Linux значительно увеличилась с тех пор, как *Steam* пришел в Linux в 2013 г. Большинство высокобюджетных игр не идут в Linux, а те, которые идут, работают хуже, чем в Windows. Тем не менее, вы все равно найдете тысячи игр для Linux (включая превосходную *Thimbleweed Park*), на которые вы можете радостно угрожать все свое свободное время.

Вы не найдете *Steam* в *AppCentre*, но его можно установить просто по `sudo apt install steam` (чтобы пройти по диалоговым окнам лицензионных соглашений, используйте клавиши со стрелками и Tab). Если вы хотите попробовать что-то более авантюрное, стоит сделать это с помощью *Steam Integration Tool*

от Solus OS. Он не даст *Steam* использовать старые библиотеки времени выполнения, которые он загружает, перехватит системные вызовы и перенаправит их на более новые библиотеки (предложенные в первом `snapt` ниже). Кроме того, он содержит ряд отладок популярных игр, что избавит вас от крайности рвать на себе волосы:

```
$ sudo apt install snapt
$ sudo snap install --edge solusruntime-gaming
$ sudo snap install --devmode --edge linux-steam-integration
```

Если столкнетесь со сложностями, его легко удалить:

```
$ sudo snap remove linux-steamintegration
```



➤ Фанаты *Maniac Mansion*, *Monkey Island* могут заново пережить памятные моменты с помощью *Thimbleweed Park* Рона Гилберта (Ron Gilbert) (см. стр. 17).



# 7 советов быстрого поиска работы от hh.ru

- 1. Определите цель.** Решите, кем вы хотите работать, как бы смешно это ни звучало. Точно сформулируйте вашу должность. Работодатель не найдет вас, если название резюме будет общим: «Менеджер» или «Начальник». Лучше уточните: «Менеджер по закупкам» или «Начальник строительной бригады».
- 2. Узнайте о своих способностях.** Вы все еще в поиске своего призвания? Пройдите онлайн-тест «Профориентация»\* [hh.ru/article/proforientation\\_promo](https://hh.ru/article/proforientation_promo) и узнайте, какая работа вам больше всего подойдет.
- 3. Составьте резюме.** Сделать это на hh.ru легко. Главное — заполните все предлагаемые поля. Уделите особое внимание опыту работы и вашим достижениям — так вы покажете работодателю вашу компетентность.
- 4. Настройте процесс.** Подпишитесь на подходящие вакансии и получайте самые свежие на почту. А также скачайте мобильное приложение HeadHunter, чтобы искать работу в любое время, в любом месте.
- 5. Действуйте.** Откликайтесь на все интересующие вас вакансии. Пишите сопроводительные письма работодателю, поясняя, почему вас интересует эта вакансия.
- 6. Сделайте резюме заметным.** [hh.ru/applicant/services](https://hh.ru/applicant/services) Подключите «Яркое резюме»\*, чтобы выделить резюме цветом, и «Автообновление»\*, чтобы поднимать его в результатах поиска. Работодатели обратят на вас внимание.
- 7. Подготовьтесь к собеседованию.** Поздравляем, вас пригласили! Самое время подготовить ответ на вопрос: «Почему мы должны взять именно вас?». Узнайте максимум информации о компании и подготовьте небольшую речь о том, какой вы классный специалист.

**И помните, что работа найдется для каждого!**



## О РАННЕМ ЗНАКОМСТВЕ С LINUX

Первый дистрибутив, который я пробовал, назывался MiniLinux. Его можно было установить из MS-DOS.







Альберто Гарсиа (Alberto Garcia) является разработчиком в Igalia, довольно успешной компании консалтинга по открытому исходному коду. После скромного старта в своей родной Галиции, на северо-западе Испании, Igalia стала действительно международной организацией, которая занимается всеми видами технологий с открытым исходным кодом, от браузеров и web-движков до собственной разработки Альберто — виртуализации с *KVM* и *QEmu*. Мы поймали его в октябре 2017 г. на саммите Open Source в Пражском отеле «Хилтон».

**Linux Format:** Как вы связались с компьютерами, Linux, программированием и связанной с ними черной магией?

**Альберто Гарсиа:** Я рано заинтересовался компьютерами. Кажется, я получил свой первый компьютер в возрасте 11 лет в виде рождественского подарка от родителей. Тогда игровые консоли были не так популярны в Испании, поэтому я получил Spectrum и научился программировать на нем. Примерно в то же время в журналах Spectrum было много технических материалов: они были не просто об играх, там было много деталей о внутреннем устройстве компьютеров. И я многому научился, вводя код из журнала.

**LXF:** Немного выдавая свой возраст, скажу, что тоже помню дни РЕЕК и РОКЕ.

**АГ:** В точку. Вот это были дни! Итак, ко времени, когда я стал писать свои первые игры и небольшие программы, я решил, что мне нравятся компьютеры и я хочу работать на них. Потом в 1996 г. я изучал информатику, и именно там узнал про Linux. Тогда это было не особенно популярно.

**LXF:** Я тогда пытался заставить работать Slackware. Но безуспешно.

**АГ:** Кажется, первый дистрибутив, который я пробовал, назывался MiniLinux. Его можно было установить из MS-DOS. Там использовалась файловая система под названием imsdos, которая реализовала POSIX поверх файловой системы FAT. И можно было распаковать ZIP-файл и запустить ядро Linux, ничего не переделывая.

Вскоре после этого я перешел на Debian. В нем был самый большой выбор ПО, и можно было установить его с компакт-дисков. Debian мне очень понравился. Нравилось видеть подробности того, как всё работает под капотом. В то время я чувствовал, что в значительной степени освоил MS-DOS и там больше нет секретов, так что Linux стал новым вызовом. В исследованиях нам приходилось многократно использовать UNIX, я обнаружил, что мне он тоже нравится, и подумал, что было бы неплохо, если бы я мог на самом деле над этим работать.

Большинство моих сверстников в своей профессиональной жизни работали на Java, Windows NT или любой другом скучном материале, который был тогда популярен. Но меня больше интересовал Linux, поэтому, когда я получил диплом, то вместе с некоторыми одноклассниками основал

»

# Звук & виртуализация

В Пражском отеле «Хилтон» Джонни Бидвелл выуживает у Альберто Гарсиа все, что вы хотели знать о виртуализации, но боялись спросить.



компанию. Наша цель заключалась в том, чтобы работать над Linux и делать открытый исходный код в целом; этой компанией была Igalia. Мы начинали работать на местном уровне. Тогда рынок сильно отличался. Большинство вещей, которые вы видите сейчас, по сути, не существовало, так что это была совсем другая компания. Но мы очень быстро включились в проект Gnome. Постепенно мы стали меньше работать с местными заказчиками и разработками, а больше — с сообществами, стоящими у истоков, внося свой вклад прямо в эти проекты.

**LXF:** Является ли Debian всё еще вашим выбором дистрибутива?

**АГ:** Так оно и есть. Я даже несколько лет был куратором Debian. Правда, думаю, что у меня всё еще есть версия Debian, которую я установил еще в 1997 г., или когда это было. Я скопировал данные на новый жесткий диск и вставил в новую машину, и эта копия не перестает обновляться. Но, исходная установка всё еще там: есть и файлы двадцатилетней давности. Мне нравится централизованная модель разработки Debian. Там очень мало иерархии — мне нравится, как она работает. Для меня было естественно начать вносить свой вклад в это. Я начал с поддержки нескольких пакетов, но теперь делаю намного больше. Наша компания поддерживает порт WebKitGTK+, и я упаковываю его для Debian. Если есть проблема, Debian позволяет нам видеть, как пакет работает в разных архитектурах, и помогает в разработке следующего потока.

**LXF:** Расскажите нам немного больше об Igalia.

**АГ:** Мы являемся консалтинговой компанией по открытому исходному коду, базирующейся

в Ла-Корунье, регион Галиция в Испании. Компания основана в 2001 г. с двумя целями: открытые инновации и разработка FLOSS. Вначале нас было 10, но теперь у нас 60 инженеров, рассредоточенных по нескольким странам. Более половины компании работает с web-браузерами и связанными с ними технологиями: немного работаем с Mozilla, довольно много с WebKit, а в последнее время также много с Chromium. Еще мы много работаем над графикой. Мы работаем с Intel по разработке Mesa, а также с другими продуктами, связанными с инструментами и оптимизацией в целом. Кроме этого, мы работаем с компиляторами и виртуальными машинами. В основном мы ориентируемся на JavaScript, поэтому у нас есть команда, работающая с Google на V8, команда, работающая с Mozilla на SpiderMonkey, и команда, работающая с Apple.

Мы много работаем и над мультимедиа, в этом месяце мы были на конференции и хак-фесте GStreamer, который тоже был в Праге. Мы поддер-

## УСТАНОВКА DEBIAN У АЛЬБЕРТО

### Исходная установка все еще там: есть и файлы двадцатилетней давности.

живаем его движки WebKit и вносим свой вклад в развитие ядра. Эти области, вероятно, составляют около 90% того, что мы делаем; еще работаем над Accessibility, вносим свой вклад в ядро, и у нас есть небольшая команда, работающая над сетевым инструментарием следующего поколения под названием Snabb (см. [www.igalia.com/networking](http://www.igalia.com/networking)). Я занимаюсь виртуализацией, другие люди в этой команде работают с сопутствующими облачными технологиями, типа системы хранения Ceph.

Исторически так уж сложилось, что мы работали над разными проектами. Сегодня у нас много

разных областей, но я бы сказал, что основное — движки web-браузера. Десять лет назад мы начали поддерживать порт WebKitGTK+, и большую часть разработок выполнили мы. Мы обнаружили, что существует немалый интерес к созданию web-движка с открытым исходным кодом, который производители могут применять в своих продуктах.

До ветвления WebKit мы были одним из крупнейших участников, после Apple и Google. Поэтому мы выросли в этой области и начали новые проекты в близких областях. Web-движки подключены к областям всего стека, поэтому нам нужны люди, разбирающиеся в мультимедиа, компиляторах (для интерпретатора JavaScript) и т.д. Сейчас у нас есть люди, работающие над web-ядром, внедряющие новые стандарты и функции; люди, работающие над компилятором JavaScript, делают его быстрее; мультимедиа, люди, работающие с ускорением GPU. Помимо этого, у нас немалый опыт работы с сообществами открытого исходного кода, и есть другие команды, работающие в разных областях.

**LXF:** Трудно не отставать от WebKit, WebKit GTK+, Blink и всех других web-движков. Не расскажете ли вы немного о том, как появился WebKitGTK+?

**АГ:** Изначально я не был частью этой команды, но мы очень плотно занимались рабочим столом Gnome, а у Gnome был и остается браузер под названием Ephy. Там использовался web-движок Gecko, но, насколько я помню, Gecko не был предназначен для встраивания. Он был разработан как web-движок для Firefox, и если вы хотите употребить его в другом месте, он не совсем подходит. С другой стороны, WebKit был разработан чисто как web-движок и потому собран с учетом встраивания. Вот почему было бы неплохо добавить к нему GTK-слой, чтобы мы могли использовать его в Gnome. Теперь его использует не только Ephy, но и все другие приложения Gnome, обязанные отображать HTML или web-контент в целом.

**LXF:** Каков основной упор вашей деятельности?

**АГ:** Я работаю с QEmu, особенно с дисками ввода-вывода и хранения, поэтому на самом деле не совсем имею дело с web-браузерами. Раньше я работал над виртуализацией аппаратных средств, драйверов устройств ядра и т.п.

**LXF:** Это довольно насыщенная конференция: помимо включения LinuxCon и ContainerCon, есть также MesosCon, Kernel Summit и, что наиболее важно для вас, я думаю, Форум KVM.

**АГ:** Да, здорово, что эти вещи расположены рядом. Я могу укомплектовать стенд, продвигая компанию. Но позже я также расскажу о KVM на Форуме. Это отличная возможность встретиться с людьми из разных сообществ.

**LXF:** Можно ли нам заранее узнать, о чем будет ваша речь?

**АГ:** Конечно. Сейчас я работаю с гипервизором QEmu. При создании виртуальной машины вам необходимо где-то хранить свои данные, а у QEmu имеется этот родной файловый формат QCOW2.







У QCOW2 множество функций. Можно создавать резервные копии файлов, моментальные снимки, по требованию осуществлять их шифрование, сжатие и увеличение. И когда вы начинаете с пустого устройства, оно занимает всего несколько сотен килобайт, а когда вы добавляете к нему, то он увеличивается. Это хорошо, но он не работает так же, как голый [raw] файл, скажем, 20-гигабайтный файл, к которому вы обращаетесь как к блочному устройству. Здесь голый файл предлагает самую быструю возможную производительность.

С QCOW2 есть некоторые накладные расходы. Вам нужны некие структуры данных для обработки моментальных снимков, переводов между адресами VM и хост с файлом, и т.д. Моя работа заключается в уменьшении сопутствующих расходов, чтобы сделать QCOW2 быстрым, как raw-файлы. Во многих случаях это происходит, но иногда *QEmu* всё еще нуждается в настройке: то надо расширить формат QCOW2, а то — улучшить драйвер.

**LXF:** Пользователям *VirtualBox* может быть трудно понять, как ядро, *KVM* и *QEmu* стыкуются друг с другом. Можете ли вы здесь дать некоторое пояснение?

**АГ:** *KVM* (Kernel Virtual Machine) — компонент ядра Linux, который позволяет ему работать как гипервизор. Т.е. это модуль ядра, а значит, программы пространства пользователя могут создавать виртуальные машины. *KVM* не зависит от *QEmu*, и вы можете создать VM, используя только API *KVM*, указав количество ядер, объем памяти и т.д. *KVM* делает всё это и заставляет ядро выполнять всё низкоуровневое планирование CPU. *QEmu* обеспечивает эмуляцию всех драйверов устройств ввода-вывода и всех аппаратных средств. Таким образом, по сути, *KVM* просто создает виртуальную машину и изолирует вызов, потом позволяет всё запускать. *QEmu* обеспечивает эмуляцию дисковых устройств, отображает... все эти вещи.

**LXF:** Я некоторое время использую *Virt-manager* для всех своих виртуальных машин, и это задействует API *libvirt* для управления виртуальными машинами. Можете ли вы объяснить, где это вписывается в общий ход событий?

**АГ:** Ну, *QEmu* в своей исходной форме — это пользовательская программа, настольная программа,

вы можете запускать ее на своем ПК. *Libvirt* — библиотека, позволяющая управлять виртуальными машинами. Вы можете их создавать, запускать, останавливать и уничтожать. Это API, который работает поверх *QEmu*, но также и поверх всех других вариантов виртуализации. Таким образом, вы можете иметь над ним нечто обрабатываемое виртуальной машиной, а *libvirt* обеспечивает уровень абстракции между самим VM-поставщиком и приложением, которое управляет всем.

**LXF:** Наши читатели знакомы с идеей создания моментальных снимков VM, а расскажите о других преимуществах QCOW2.

**АГ:** Сжатие довольно удобно, поэтому изображение QCOW2 делятся на кластеры (аналогичные блокам в файловой системе), и можно эти кластеры сжимать. Полученное изображение намного меньше, чем то, которое вы получили бы с raw-файлом. Есть накладные расходы, потому что при каждом чтении кластера вы должны его распаковывать, но таков компромисс. Кроме того, поддерживается шифрование; в этом году получено шифрование, совместимое с LUKS. Также поддерживаются файлы резервных копий, которые тесно связаны с моментальными снимками.

Теперь у вас есть базовый файл с базовым изображением и активным файлом, в котором записаны данные. Когда надо прочитать данные, которых нет в базовом файле, переходите к новому файлу. Таким образом, можно иметь несколько разных виртуальных машин, используя одну и ту же гостевую ОС, и обмениваться общими данными между ними.

**LXF:** А что у нас с выживаемостью? Если я вытаскиваю вилку, пока работает VM с поддержкой QCOW2, останутся ли там все мои данные, когда я запущу ее снова?

**АГ:** Ну, формат QCOW2 проверяет, что при каждой записи на диск всё выполняется таким образом, что если на виртуальной машине произойдет сбой, файл не будет поврежден. Но если он поврежден, то, по крайней мере, мы его засечем. Так что, разрабатывая формат, мы должны учитывать порядок написания и пытаться удостовериться, что данные не потеряны. Если данные теряются, мы должны убедиться, что можем это обнаружить.

**LXF:** *QEmu* прошел долгий путь. Я помню, как использовал его пять лет назад, струсил и вернулся к комфорту *VirtualBox*; но теперь пользуюсь им всё время. Какие новые функции можно ожидать?

**АГ:** Одной из наиболее важных структур данных в формате QCOW2 является та, которая переводит гостевые адреса в адреса хоста. Другими словами, чтобы получить доступ к диску, вам сначала надо прочитать таблицы, содержащие эти переводы, а для высокой при этом производительности требуется кэш. В противном случае вам придется дважды читать эти таблицы при каждом обращении к диску.

В настоящее время такой кэш уже работает, но я тружусь над некоторым кодом, чтобы сделать его чуть более эффективным. Это позволит вам иметь уточненную структуру записей в кэше и с тем же объемом кэш-памяти добиться более высокой производительности.

**LXF:** В прошлом году я кое-что слышал об advent-календаре *QEmu*. Что это была за история?

**АГ:** Несколько человек из сообщества *QEmu* создали сайт, который каждый день показывал разные образы дисков в течение декабря месяца 2016 г. Там была кое-какая классика. Моим вкладом стало изображение Spectrum с семью играми, разработанными в этом десятилетии. Также была небольшая шахматная игра *BootChess*, ее так называют, поскольку она достаточно мала, чтобы вписаться в 512-байтовый загрузочный сектор. Были еще некоторые полезные вещи, такие как WireGuard VPN и Rustbased Redox OS. Вы можете и должны его посмотреть на сайте [www.qemu-advent-calendar.org/2016](http://www.qemu-advent-calendar.org/2016). Там еще есть издание 2014 г.!

**LXF:** Узнать больше о компании Igalia и их работе с WebKit и Wayland можно из интервью следующего месяца с коллегой Альберто из Igalia, Хуаном Хосе Санчесом Пенасом [Juan José Sánchez Penas]. **LXF**





# Будущее управления пакетами

Маянк Шарма не осознавал, что управление пакетами Linux надо исправить... пока не встретился с проектами, работающими над переосмыслением распространения пакетов в Linux.

**У**правление пакетами часто превозносится как единственное крупнейшее достижение, которое Linux принес в мир операционных систем. Но по сути это мрачный и сложный мир официальных и неофициальных репозиториях, разных форматов пакетов, множества метаданных и нескончаемой очереди библиотек.

Менеджер пакетов зависит от формата пакетов, которыми он управляет. Так, например, *DNF* (а перед ним *Yum*) использует RPM в качестве движка и увеличивает его функциональность посредством добавления новых функций. Системы управления пакетами экономят время, автоматизируя времязатратные задачи, например, разрешение зависимостей. Фактически, *DNF* появился в Fedora

как ответ на долгие проблемы в *Yum*, включая разрешение зависимостей и способ работы с онлайн-репозиториями.

Репозитории вносят еще один уровень путаницы. Кроме официальных, большинство дистрибутивов также разрешают использо-

## Смешение репозиториях может иметь непредсказуемые последствия для стабильности системы.

вать индивидуальные и сторонние репозитории, чтобы дать создателям пакетов возможность продвигать более новые релизы программ, чем те, которые включены в официальные репозитории. Смешение репозиториях способно вызывать непредсказуемые последствия для стабильности системы.

Конечно, вы можете употреблять такие инструменты, как *Alien*, для конвертирования между Linux Standard Base (LSB), совместимыми RPM- и Deb-пакетами. Но из-за конструкции современных систем создания пакетов нельзя использовать Fedora без RPM или Ubuntu без Deb. Можно смело сказать, что хотя современная схема пакетирования и работает, но оставляет немалый простор для улучшения.

За годы предпринималась масса попыток перестроить модель распространения программ Linux, чтобы до мелочей отладить существующую модель. На данный момент, всех привлекают три модели: это *AppImage*, *Flatpak* и *Snaps*. Направим же на них прожектор *Linux Format* и посмотрим, не нарушили ли они кодекс доставки приложений в Linux.



**М**ного лет пользователи Linux прожили с двойственностью Deb-пакетов для дистрибутивов Linux на базе Debian или Ubuntu и RPM для дистрибутивов Linux на базе Fedora или SUSE. Хотя эти пакеты предлагают удобный способ установки программ в соответствующих дистрибутивах, вся система в целом не самая удобная для разработчика приложений. Вследствие аспекта фрагментации разработчикам приходится создавать пакеты для нескольких дистрибутивов по отдельности.

Именно это и попробовал решить AppImage в 2004 г. (когда он был известен как klik). AppImage распространяет все приложение как один исполняемый файл. Приложение работает из этого пакета и не помещает файлов в базовую систему. Кроме того, файлы AppImage содержат все библиотеки и файлы, необходимые им для работы, что позволяет им работать на большом числе дистрибутивов.

Типичная программа Linux создает файлы в разных местах, требуя разрешений root, чтобы применить эти изменения ко всей системе. AppImage этого не делает. Технически AppImage — это ELF-файл, который также содержит встроенную файловую систему squashfs. Все файлы, требуемые

для работы приложения, хранятся там. Когда вы запускаете файл, программа, встроенная в этот файл, монтирует файловую систему в директории под /tmp, а затем запускает приложение внутри этой директории.

В целях безопасности монтирование файловой системы все равно требует наличия прав root. К счастью, большинство дистрибутивов рабочего стола имеют поддержку FUSE, что позволяет монтировать файловую систему без привилегий root. Это используется файлами AppImage для ровной работы, и это — единственная поддержка, которую они ожидают от базовой системы. Кроме того, AppImages хорошо переносимы и могут работать где угодно, в том числе в средах Live. Обычно AppImage устанавливает файл рабочего стола, чтобы интегрировать его с вашей установкой Linux, как приложение, установленное обычным образом, а значит, вашу программу можно будет искать в среде рабочего стола.

## Дорогое удовольствие

Нил Гомпа [Neal Gompa], человек, продуктивно работающий над дистрибутивами, имеет опыт работы с этими тремя форматами, поскольку занимается

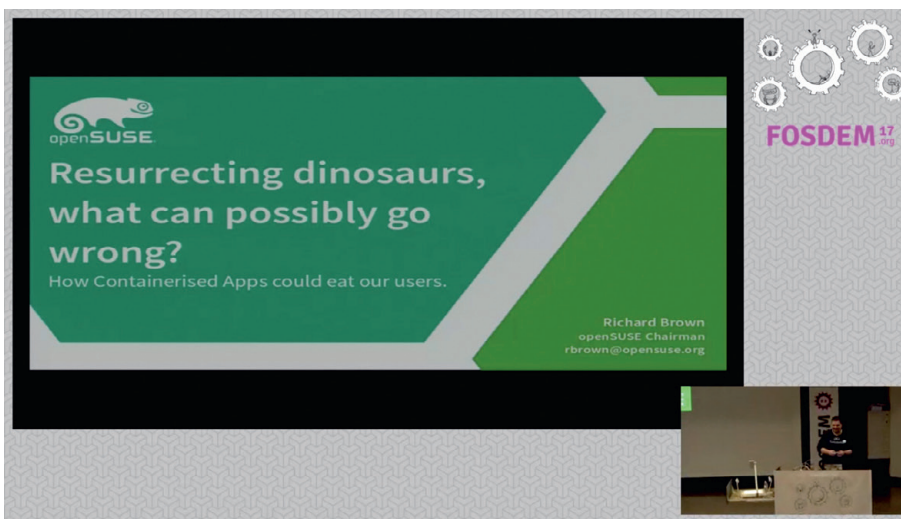
поддержкой Flatpak в Mageia и Snaps в Fedora, а к тому же собирал AppImages. Нил считает, что у AppImage нет корпоративной поддержки, потому что они «по сути ущербны» и не добьются успеха, пока все дистрибутивы Linux не согласятся обеспечивать базовый ABI (application binary interface — двоичный интерфейс приложений).

«AppImage для функционирования используют библиотеки хоста, — говорит он. — Ирония в том, что по сути это идеальный случай, поскольку он максимизирует ценность того, что уже есть в нашей системе, и, в теории, облегчает интеграцию с системой».

Однако на практике связанные библиотеки ведут к несовместимостям, которые трудно отладить: «Не говоря уж о том, что основатель AppImage уделяет очень мало внимания механизмам безопасности и не имеет встроенного механизма для ограничения AppImage. Набор инструментов AppImage (AppImaged, AppImageKit и т.д.) предлагает разные расширенные функции во время выполнения и время сборки соответственно, но ни один из них не делает его простым или очевидным для проверки достоверности AppImage перед их использованием», объясняет Нил.

Нил также не купился на идею отсутствия у AppImage центральных репозитория: «AppStream функционально бесполезен, потому что AppImage предназначены для скачивания с сайтов Джо Рэндо [Joe Rando] Apps o'Fun, которым не обязательно беспокоиться о целостности. Основателю AppImage нужны приложения в стиле MacOS — „скачать, чтобы запустить“. Однако даже Apple сейчас отказывается от этой модели, потому что очень трудно обеспечить целостность приложений и безопасность операционной системы».

Flatpak, как считает Нил, является заметным улучшением по сравнению с AppImage, поскольку целостность и безопасность просто вписаны в дизайн Flatpak: «Он использует некоторые функции ядра Linux для настройки базового ограждения данных (например, seccomp)». Большое различие между AppImage и Flatpak в том, что последний не использует для приложений библиотеки распределения хостов, вместо этого полагаясь на runtime'y [runtime — англ. время выполнения].»



» Пересмотр лицензий связанных зависимостей для обеспечения совместимости — главная проблема универсальных форматов, которой пренебрегают, говорит Ричард Браун из openSUSE.

## Решения в контейнерах

В дополнение к AppImage есть другие независимые от дистрибутивов универсальные форматы пакетов — Snaps и Flatpak — которые надеются облегчить жизнь как пользователям, так и разработчикам.

Главное различие между Flatpak и Snap с одной стороны, и AppImage с другой в том, что первый работает в контейнерах-песочницах. Snaps используют для изоляции приложений модифицированный AppArmor, тогда как Flatpak использует SELinux. Подобно AppImage, Snaps и Flatpak включают все библиотеки и зависимости в сам пакет. Однако если Snaps может использовать только библиотеки, включенные в пакет, Flatpak может использовать

библиотеки, включенные в пакет, а также библиотеки общего доступа из другого Flatpak.

Представители обоих лагерей горячо спорят между собой, стараясь доказать техническое превосходство одного формата перед другим. Момент, на который часто ссылаются представители лагеря Flatpak — что Snap был создан внутри Canonical и что эта технология жестко кодирована, чтобы использовать магазина пакетов Snap. Утверждают также, что Flatpaks уделяют больше внимания доставке программ на рабочий стол, Snaps же — в основном серверная технология, которую в Canonical адаптировали для настольной операционной системы.

Сравнение их технологических достоинств практически бессмысленно, поскольку при неистовом темпе разработки этих форматов их недостатки исчезают в ходе конкуренции. Так, теперь вы можете запускать приложения AppImage внутри контейнеров, созданных с помощью Firejail. Этот процесс не требует практически никаких дополнительных усилий со стороны пользователя, если только у него уже установлен Firejail. Последние версии Firejail даже защищают от атак на сервер отображения X, таких как шпионская клавиатура. Странники AppImage утверждают, что при использовании вместе с Firejail приложения AppImage даже безопаснее, чем пакеты Snap и Flatpak.

Нил объясняет, что runtime — это мини-дистрибутив, разработанный, чтобы Flatpak работал поверх него. «Runtime'ы по умолчанию основаны на Yocto, — говорит он, — но можно создать собственный runtime поверх дистрибутива, чтобы в нем работали приложения, созданные для этого дистрибутива (в самом деле, Fedora намеревается это сделать)».

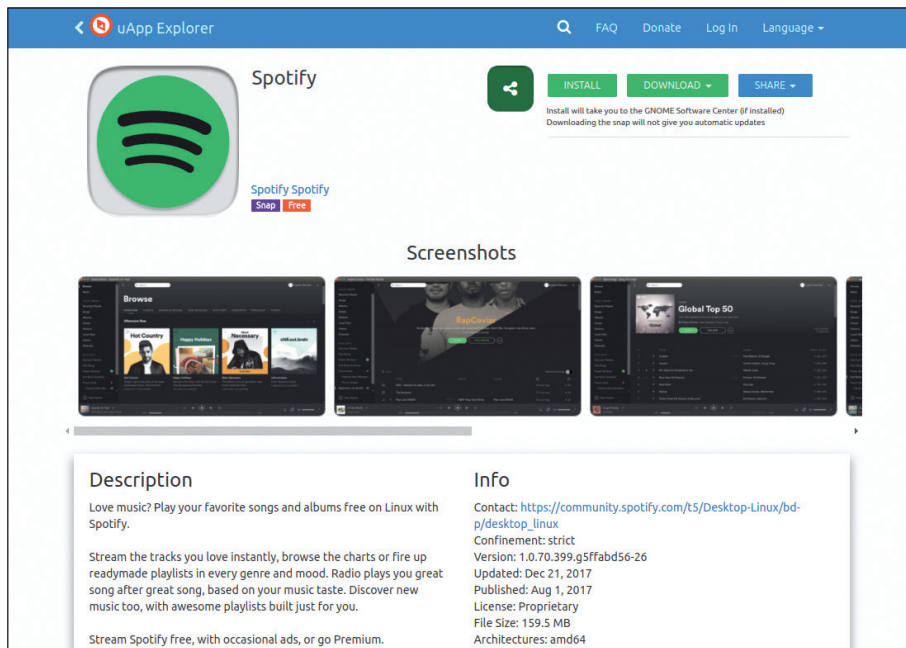
Кстати о модели безопасности. Нил говорит, что, подобно AppImage, Flatpak также работает для непривилегированных пользователей: «Они могут скачать и запустить приложения с удаленного Flatpak, например, FlatHub, бесплатно. Удаленные требуют подписи GPG и контрольных сумм SHA512, так что ненадежные и неverified Flatpaks трудно получить удаленно. Иными словами, пользователи могут просто скачать пакеты Flatpak и установить их для запуска, как AppImage, но это не предполагалось пользовательской моделью по умолчанию. И даже тогда ожидается, что пакеты Flatpak будут подписаны, чтобы Flatpak мог их удостоверить».

## Домены рабочего стола

Несмотря на все свои преимущества, Flatpak предназначен только для настольных GUI-приложений. «Создатели Flatpak полагают, что для серверных приложений должны использоваться другие технологии (Docker, Kubernetes и т.д.). И есть зияющий пробел для приложений CLI», указывает Нил.

Чтобы быть в состоянии работать с серверными приложениями, говорит Нил, Snap предпринимает то, что Flatpak делает на порядок лучше, и для ограждения данных использует AppArmor. Однако обратная сторона в том, что Snap требует солидной работы над дистрибутивом, интегрирующим его, чтобы он заработал. «Нужны изменения в ядре, — говорит он, — и есть целый класс Snaps под названием «классически огражденные Snaps», которые требуют, чтобы дистрибутив установил путь /snap — директорию или символическую ссылку на реальную директорию, настроенную на путь монтирования snap».

Нил указывает, что на сегодняшний день только дистрибутивы Ubuntu и Solus, по его сведениям,



### ➤ Пример использования Flatpak в реальном мире — музыкальный сервис Spotify.

полностью приняли этот нестандартный путь. Другие дистрибутивы, такие как Fedora и Arch, таким путем не пошли.

При всем при том, сборка Flatpak — дело простое, и как пакет инструментов Snappy, очевидно, самый простой в использовании: «Однако сейчас вы можете только создать основанные на Ubuntu snap с помощью официальных инструментов, хотя ведется работа над другими способами распространения», сообщает Нил.

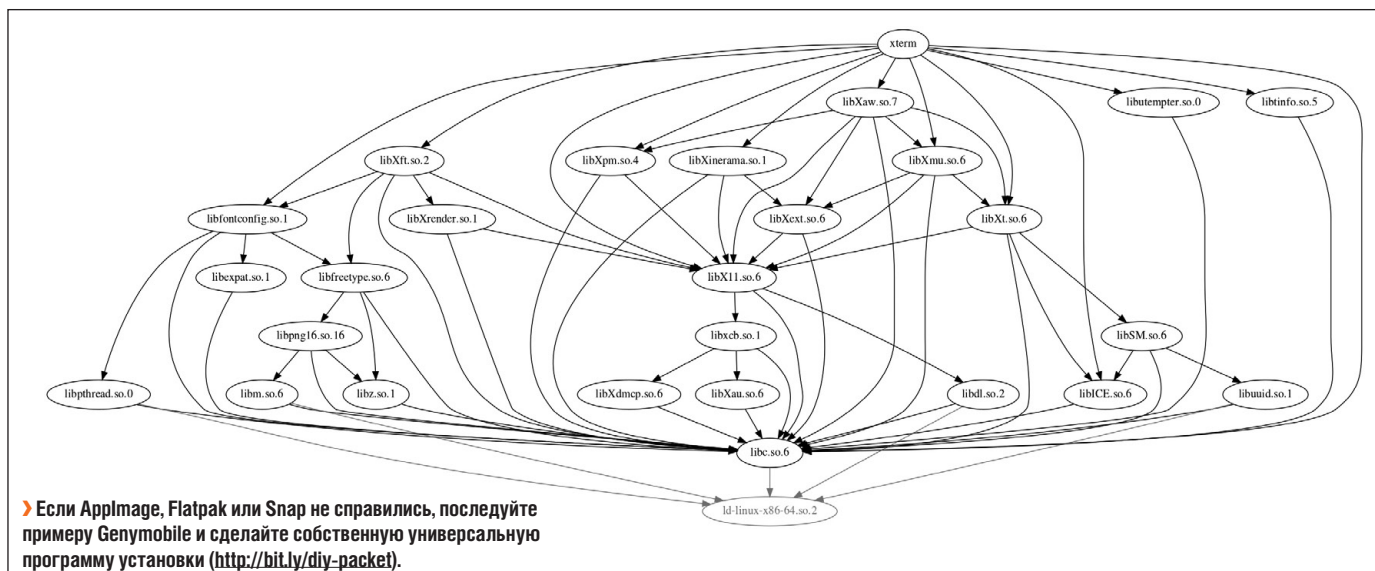
Более того, в отличие от Flatpak, Snappy не поддерживает множества репозиториях (или «магазинов [stores]» на жаргоне snappy). Есть только один магазин Snap, и им занимается Canonical. Код не поддерживает легкого переключения между магазинами, и настройка нескольких магазинов невозможна. Это означает, что любой пользователь Snap прикован к магазину Canonical. «Это в той или иной степени разрушает децентрализованную модель,

за которую выступает Свободное ПО и ПО с открытым кодом, в пользу управляемой из одного центра платформы поставки ПО в стиле Apple», говорит Нил.

## Приветствуя изменения

Если оставить в стороне технические сравнения, внедрение является истинной мерой технологического успеха, и это напрямую связано с предлагаемыми пользователям и разработчикам преимуществами. Джефф Хугланд [Jeff Hoogland], ведущий разработчик Bodhi Linux, считает, что универсальный формат пакетов является разумной идеей, по крайней мере, для приложений для конечного пользователя.

«Мы уже видим, как подобного рода идея используется в игровом мире через приложения вроде Steam, — говорит Джефф, — и таким образом, облегчает задачу для неигровых приложений



➤ Если AppImage, Flatpak или Snap не справились, последуйте примеру Genymobile и сделайте собственную универсальную программу установки (<http://bit.ly/diy-package>).

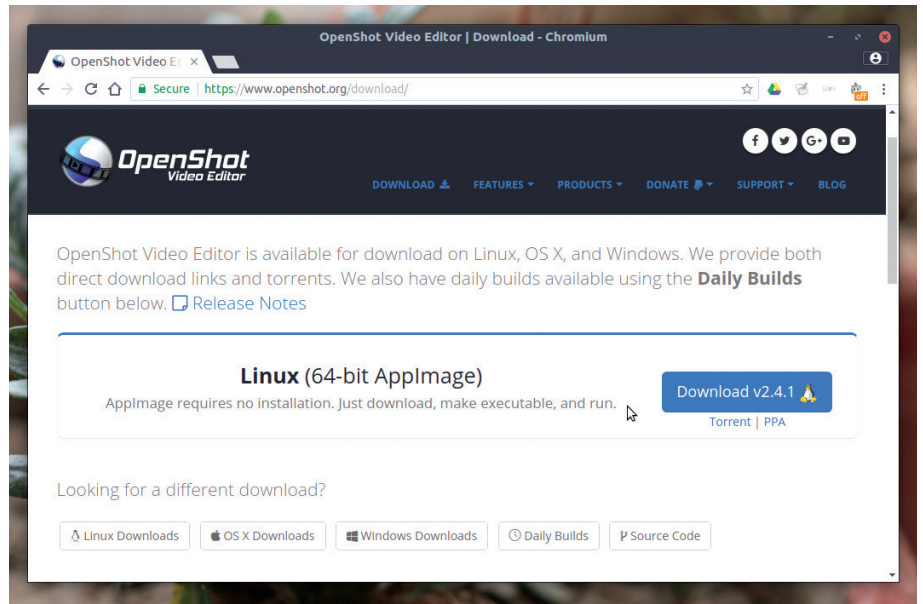


устанавливаться и обновляться независимо от требований к пакетам определенных дистрибутивов, что является крупным шагом к внедрению большего числа популярных программ в системы Linux».

Джонатан Томас [Jonathan Thomas], разработчик OpenShot, признается: «Я очень остро ощущаю потребность в универсальной программе установки в Linux. Большинство разработчиков приложений очень бы хотели, чтобы был согласованный брэндинг, обмен сообщениями и единый поток для установки и обновления их приложений во всех версиях Linux». Простота обеспечения единообразия в установке и обновлении является одной из основных причин внедрения этих универсальных установщиков в *digikam*: «Каждый день мы обеспечиваем сборку всех пакетов для конечного пользователя, чтобы проверить исправление ошибок между стабильными релизами. Это быстро и очень эффективно», говорит ведущий разработчик *digikam*, Жиль Колье [Gilles Caulier].

Чтобы показать различие между распространением RPM-/Deb-пакетов и универсальными установщиками, Жиль отмечает проект, ранее применяемый для получения множества отчетов по нарушенной двоичной совместимости. С универсальными программами установки это изменилось: «Я знаю, как работает приложение и какие нужны зависимости. Нет никакого риска нарушения двоичной совместимости с пакетом: всё уже внутри».

Фактически, когда пользователь *digikam* сообщает об ошибке с дистрибутивом RPM, проект просит его проверить, повторяется ли та же проблема с универсальным установщиком: «И вот удивительно: 50% сообщений относятся к ошиб-



» Если кто-то жаждет потратить на это деньги, Джонатан Томас из OpenShot хотел бы создать сетевое решение, занимающееся всем, от привлечения исходника до добавления зависимостей, создания универсального установщика и его публикации.

дистрибутивами и простоте интеграции, — говорит он. — Я еще был знаком с *Krita* (где тоже используется этот формат), и со мной даже связывался создатель Appliance, предлагая помощь». У Жили был похожий опыт с Appliance. Год назад он занимался Flatpak, но его разочаровала документация.

С другой стороны, дистрибутив Solus использовал Flatpak для решения интересной проблемы с поставкой приложений. В своем посте ведущий

приложения сначала берутся у производителя и затем конвертируются в родной формат пакетов Solus, *.eopkg*. Система была ненадежна и не обеспечивала механизма для автоматического обновления. Тщательно все взвесив с технической точки зрения, Айки выбрал Flatpak, который он счел самым простым для интеграции в Solus.

## Без посредников

Те же причины упоминал один из ведущих разработчиков KDE, Себастьян Кюглер [Sebastian Kügler], в своем блоге в начале прошлого года. Обосновывая необходимость в трех универсальных форматах пакетов, Себастьян написал, что в данный момент проект зависит от поставок приложений и обновлений дистрибутивами Linux. Он заявил, что это «проблематично» и для дистрибутивов, и для разработчиков приложений из-за задержки с обновлениями. Себастьян полагает, что использование универсальных форматов пакетов и форматов пакетов для приложений решит эту проблему: «Это означает, что мы сможем сами поставлять приложения и устраним из этой схемы посредник-дистрибутив».

Дональд Стюарт [Donald Stuart] из Mageia не согласен с позицией KDE: «Надо как следует обдумать ситуацию, когда KDE будет поставлять обновления, чтобы пользователи могли быстро получать от них обновления. Однако команды QA и пакетов в Mageia отлично справляются во-первых, с очень своевременной поставкой обновлений; и во-вторых, обеспечивают то, что поставляемые обновления стабильны для релиза и работают, как планировалось, не конфликтуя с остальными пакетами в дистрибутиве».

Нил считает странным, что KDE рассматривают дистрибутивы как посредников, учитывая, что многие из разработчиков KDE трудятся в разных дистрибутивах. Он считает, что аргумент KDE основывается на модели разработки дистрибутива, где

## Я выбрал Appliance главным образом благодаря его совместимости с разными дистрибутивами.

кам вышележащего [upstream] ПО, — говорит Жиль. — Можете представить себе, сколько времени мы теряем на определение этих проблем без такого независимого пакета, как Appliance, для быстрого определения проблемы с пакетами».

Джонатан также использует Appliance для OpenShot. «Я выбрал Appliance главным образом благодаря его широкой совместимости с разными

разработчик проекта Айки Догерти [Ikey Doherty] написал, что проект, как и большинство дистрибутивов Linux распространяет основную часть своих программ через двоичные репозитории. Однако они не могут поставить некоторые приложения, как *Google Chrome*, в основном из-за лицензионных ограничений. Чтобы распространять эти приложения с помощью «родного» управления пакетами Solus,

Функция	Appliance	Flatpak	Snap
GUI приложения	Да	Да	Да
Процессы сервера	Да	Нет	Да
Работа без установки	Да	Нет	Нет
Работа без прав root	Да	Да (после установки)	Да (после установки)
Автоматические обновления	Нет	Да	Да
Песочница	Через Firejail	Встроенная	Встроенная
Онлайн-репозитории	ApplianceHub	FlatHub	Snap store
Отдельные репозитории	Нет	Да	Нет
Лицензия	MIT	LGPL	GPL

главным считается стабильность. Нил говорит, что в отличие от Mageia, который следует «только LTS и обновлениям внутри серий LTS», дистрибутивы типа Fedora и openSUSE тщательно отслеживают вышедшие релизы и таким образом, являются более активной частью разработки KDE.

«По моему мнению, большая часть страданий KDE вызывается дистрибутивами, которые не обновляют свой стек регулярно, — говорит Нил. —

## Ни один размер не подходит всем

Как известно опытным линуксоидам, собрать дистрибутив — это не просто поместить приложения в репозитории. При подгонке пакетов нужны значительные усилия по обеспечению стабильности, совместимости и безопасности. Поскольку этот процесс требует времени, многие люди во всех звеньях пищевой цепочки приложений Linux воспри-

имчивы к универсальным форматам пакетов, хотя инструменты для их создания включены для всех желающих этим заняться. «Они являются решением, нарывающим на проблемы, и влекут дополнительные проблемы с тестированием и распространением обновлений», говорит он.

«Будь у меня Snaps с гибкостью Flatpaks, я бы принял такое решение, — говорит Нил. — Но сегодня, хотя мне больше нравятся Snaps как система, я не думаю, что они интересны для многих людей. Вместо этого, я считаю, победит Flatpak, потому что он прост для интеграции в дистрибутивы и решает проблему поставки для графических приложений, что и делает большинство производителей проприетарных приложений. И, в отличие от Snaps, поскольку все могут разместить свои удаленные Flatpak, можно интегрировать собственный метод подписки для пользователей или плату за предоставление доступа к приложениям».

С другой стороны, Джонатан рад бы видеть улучшение инструментов для этих универсальных программ установки: «Я по-прежнему мечтаю, что однажды разработчики приложений смогут включать приложения в пакеты для всех платформ (Linux, Mac и Windows) с одним универсальным образом и единым магазином репозитория/приложений, и с минимальными усилиями, и обеспечивать простую, ориентированную на пользователя, установку и обновление. Да, это является моей мечтой».

Но пока светлое будущее не настало, разные форматы продолжают сосуществовать, подобно тому, как сосуществуют форматы двоичных пакетов. Их темп разработки и внедрения будет продолжаться, и скоро ряд популярных программ будет доступен в одном из этих форматов. С наших теперешних позиций кажется, что Snaps продолжит свое развитие с облаком и приложениями IoT, тогда как AppImage и Flatpak поделат между собой настольные приложения.

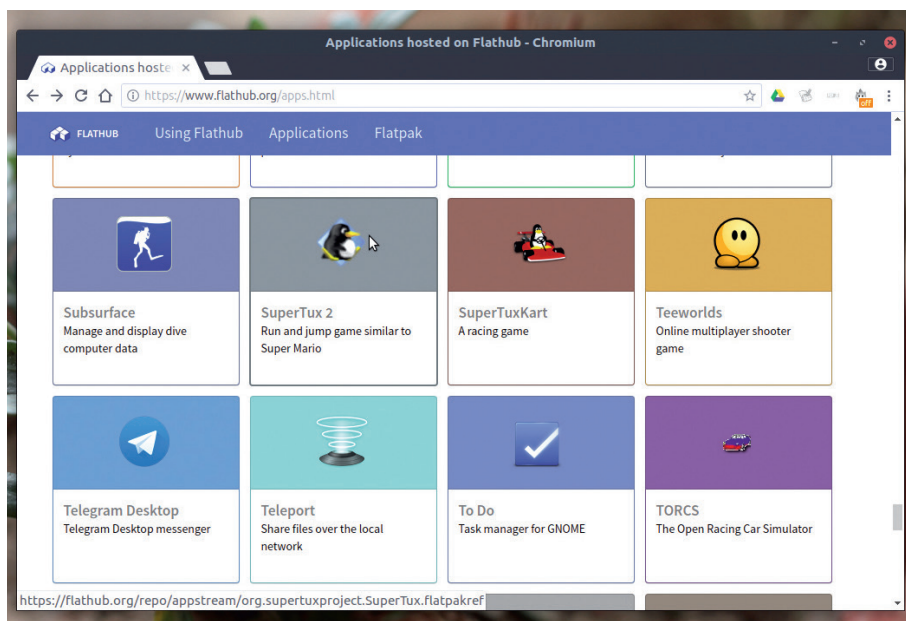
«В конечном итоге, главное в FOSS — это наличие выбора пользователя, — заключает Джефф, — так что эти типы программ установки — просто еще одна опция для конечных пользователей с целью дать им возможность использовать свой компьютер по своему усмотрению». **LXF**

## Многие люди во всех звеньях пищевой цепочки приложений Linux воспринимают это как проблему.

Семейство Debian, включая множество его производных (в том числе KDE neon), предпочитают «стабильность», а отсюда следует, что они не выпускают самые свежие релизы, вводящие функции и отладки, и огорчают пользователей программ KDE в этих дистрибутивах».

нимают это как проблему и всячески приветствуют в качестве решения универсальный формат пакетов. Надеемся, мы показали, что хотя он полезен, но не является панацеей, чем многие его считают.

Дэвид Ходжинс [David Hodgins] из Mageia сказал, что дистрибутив не планирует обеспечивать



➤ Flathub.org — официальный сервис сборки и магазин приложений для настольных программ Flatpak.

## Возобновляемые решения

Частично причина новых систем пакетирования происходит из связи между существующими основанными на репозиториях менеджерами пакетов и фиксированными релизами дистрибутивов. В комментарии к своему посту ведущий разработчик KDE Себастьян Кюглер заявляет, что разработчики дистрибутивов должны тратить свое время на работу «над тем, что принесет пользу».

В ответ на его комментарий другой опытный работник KDE и openSUSE, Люка Бельтрам [Luca Beltrame], подчеркнул, что часть вещей, на которые дистрибутивы уже потратили немало времени и которые приносят пользу, включают интеграцию, с которой, по его утверждению, «ни Snaps, ни Flatpaks,

ни AppImages никогда не справятся как следует».

Он также выделяет несколько других аспектов, на которые дистрибутивы тратят время при пакетировании приложений, например, анализ безопасности, правовой анализ и контроль качества [QA].

Далее Люка сказал, что скорость обновлений приложений внутри обычного дистрибутива непостоянна. По его мнению, решение проблемы своевременной поставки обновлений лежит в возобновляемых релизах. Эти же мысли звучали на выступлении на FOSDEM '17 председателя openSUSE, Ричарда Брауна [Richard Brown], который заявил, что возобновляемые релизы, вероятно, являются лучшей моделью для исправления

поставки обновлений. Использование возобновляемых релизов, говорит Ричард, позволяет им «заново использовать уже имеющиеся знания, уже имеющиеся инструменты, уже имеющуюся инфраструктуру».

Далее он добавляет, что универсальные форматы пакетов проистекают из убеждения, что традиционные дистрибутивы слишком медлительны — чего не скажешь о возобновляемых релизах, таких как openSUSE Tumbleweed. Дистрибутив тщательно тестируется и всё равно умудряется поставлять обновления вышедшего ПО в реальное время: KDE Plasma — в день релиза, Gnome — менее чем за сутки, и т.д., подчеркивает Ричард.



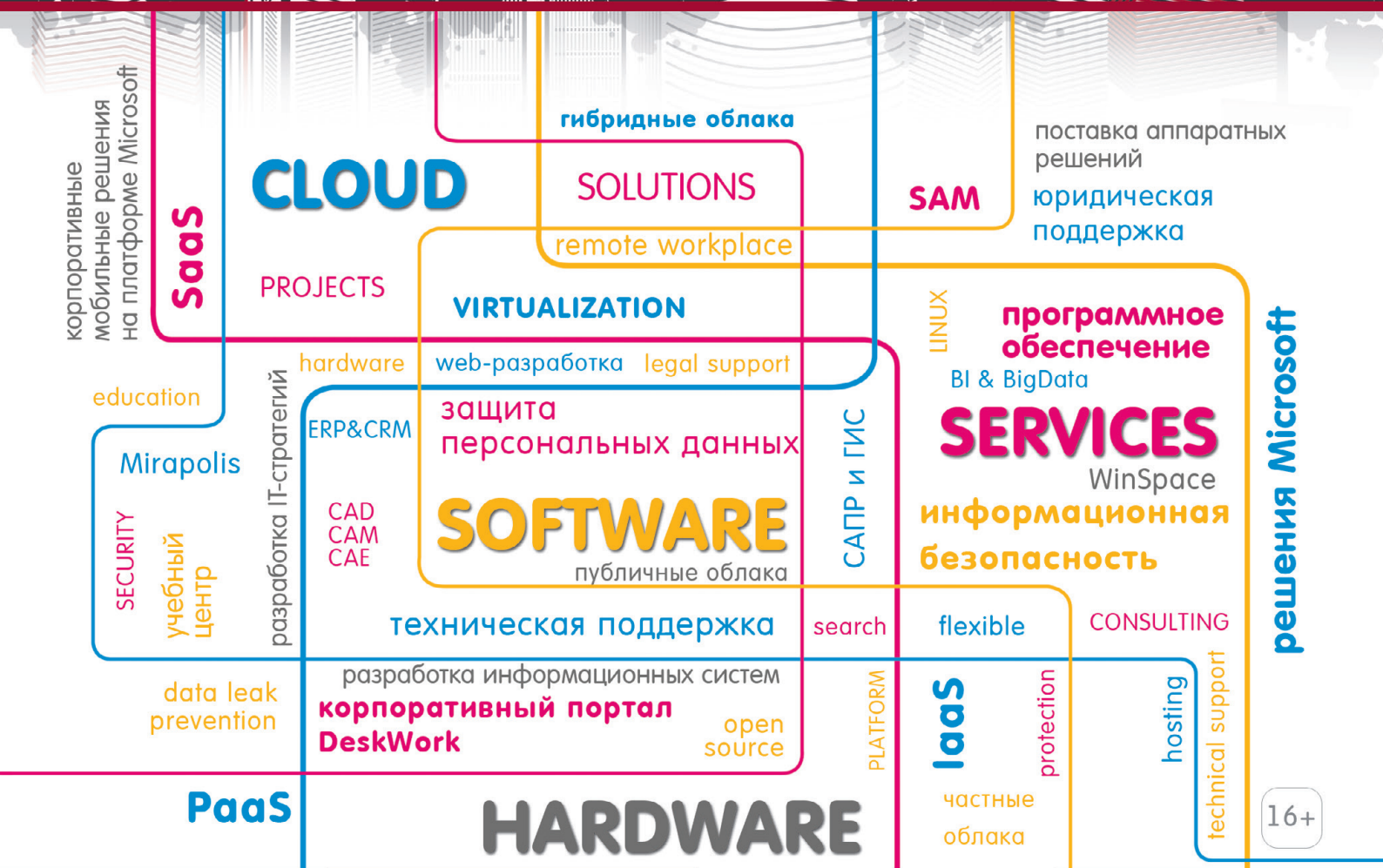
# softline®

Cloud Software Hardware Services

# 20+

Years in IT

## IT-архитектура вашего бизнеса



решения Microsoft

16+



Наш эксперт

Доктор **Валентин Синицын**  
Бывший разработчик KDE с 0.16 практиканта Google в год, так что у него хватает времени на то, чтобы создавать облака Linux и писать слова.

Создаем лучшие облака

Десять или 15 лет назад было обычным именовать каждый последующий год Годом настольного Linux. Больше такого нет. Знаете, почему? Я полагаю, что настольные ПК уже не так важны, как раньше. И одиночные серверы тоже теряют значимость. Похоже, всё перемещается в «облако».

Да, это старая история. Если вы ждете, что я коснусь проблем конфиденциальности, так и есть. Но на сей раз я рассмотрю эту проблему под другим углом. Никто не будет спорить, что без контроля над своим сервером, от центра обработки данных и до программ, вы не контролируете свои данные. Хозяин может захватить ваше оборудование, если здание не принадлежит вам. Провайдер облачных услуг может делать всё что угодно с вашими битами и байтами, просто потому что может. Если ЦОД ваш, но находится в другой стране, поближе к клиентам, то вы его хозяин лишь на бумаге.

Дилемма «Всё сам» или «Общий доступ» не нова. Могут поспорить, где-нибудь у вас в доме есть бойлер. У меня — нет: мы полагаемся на центральное отопление. А значит, у меня нет прямого способа регулировать температуру в комнате, но зато я не отвечаю за трубы и их ремонт при поломке. Я плачу за коммунальные услуги, и для меня всё это «просто работает». Короче, всё дело в цене самообеспечения и в рисках не получить услугу, когда она нужна.

Нечто подобное творится и в облаке. Нередко использовать что-то в виде сервиса дешевле, и риск оправдан. Иногда это не так, и надо за всё отвечать самому. Суть концепции облака в том, чтобы остаться надолго. Сосредоточимся же на его безопасности. Сквозное шифрование (когда ключи у вас, а не у провайдера) и репликация — шаги верные.

# По рецептам д-ра Синицына

Эзотерическое системное администрирование из причудливых заворотов кишок серверной.

## Amazon внедряет KVM: Что дальше?

Смелые утверждения на новостных сайтах и Assembler на слайдах re:Invent 2017. Что же происходит-то?

Постоянные читатели этой рубрики, вероятно, слышали о виртуализации. И что Amazon ушел от Xen. Время от времени компании меняют свое отношение к программам, даже крупные — помните новость «Microsoft любит Linux» несколько лет назад?

Так из-за чего разгорелся сыр-бор?

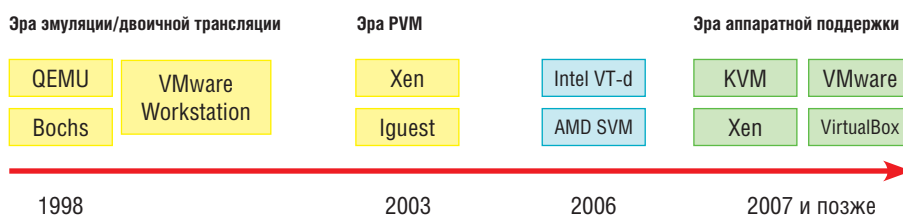
Чтобы лучше в этом разобраться, давайте вспомним историю компьютерной виртуализации в x86. В самой виртуализации нет ничего нового — она существует примерно с 1960-х. Но чтобы виртуализировать архитектуру, необходимо также охватить все ее зависимые от управления инструкции (например, прерывания отключения). x86 долгое время не выполнял эти требования, и такие пионеры, как QEmu или VMware, использовали эмуляцию (которая была медленной) или прибегали к разным уловкам, например, двоичной трансляции (которая была сложной и тоже медленной).

Xen дебютировал в 2003 г. с простой идеей. Если мы не можем заставить оборудование поймать зависимые от управления инструкции, давайте сделаем гостевое ядро и вообще не будем их использовать. Вместо этого ядро будет выдавать гипервызовы практически так же, как непривилегированные процессы выполняют системные вызовы в Linux. Так

уже быстрее (издержки всего 10%), но это не будет работать, если вы не сможете заставить сотрудничать свою ОС (вспомните Windows ранних 2000-х). Последнее не слишком заботило ранние Amazon Web Services: Xen на момент запуска уже был зрелой технологией, поэтому они ее внедрили.

В конечном итоге x86 ввели поддержку виртуализации оборудования (Intel VT-x и AMD SVM) примерно в 2006 г. Xen быстро добавил поддержку этих технологий, но появились другие проекты; особо стоит отметить KVM. Там, где Xen всё делал сам по себе в качестве отдельного проекта, KVM сливалась с ядром Linux и использовала преимущества его обновлений.

Вскоре после этого Red Hat и прочие перешли на KVM. Amazon (и Citrix) были, вероятно, двумя основными игроками, которые предлагали Xen в 2017 г. Вот почему «легковесный» основанный на KVM Nitro Hypervisor стал такой важной новостью. Однако обратите внимание, что Nitro не использует повторно другие части экосистемы KVM, как QEmu или libvirt. Кроме того, Xen дает возможность работать более старым типам экземпляра AWS в течение их жизни. Тем не менее, Amazon ясно дал понять, что будущее — за Nitro, а Xen теперь скорее устаревшая технология, чем мейнстрим.



Краткая история виртуализации x86 в Linux.



# OSTree через Git

От web-разработки до ядра Linux, Git практически вездесущ. А если мы его перенесем в наши корневые файловые системы?

**П**омните условие Теста Джоэля [The Joel Test] ([www.joelonsoftware.com/2000/08/09/thejoel-test-12-steps-to-better-code/](http://www.joelonsoftware.com/2000/08/09/thejoel-test-12-steps-to-better-code/))? Тогда, в 2000 г., это, помимо всего прочего, был хороший критерий культуры разработчика в вашей компании. Первый вопрос был: «Есть ли у вас система управления версиями исходного кода?» Восемнадцать лет спустя, когда у вас есть Github и друзья повсюду, наверное, никто в здравом уме не запустит софтверный проект без VCS типа *Git*. Так почему бы не распространить эту практику до дерева файловой системы ОС?

Бесспорно, самая важная функция, предлагаемая VCS, это возможность «перемотать время назад». Сломав код, вы возвращаетесь к предыдущей версии и потом идете вперед. Говоря в терминах ОС, если последнее обновление мешает работе системы, вы его откатываете и работаете дальше. Идея не нова: такое уже делают моментальные снимки системы, но они еще сыроваты (LVM) или им нужна поддержка файловой системы (btrfs). С другой стороны, инструменты вроде *etckeeper* (<https://etckeeper.branchable.com>) независимы от файловой системы, но имеют ограниченную сферу применения (они могут работать только с файлами config).

*OSTree* (<http://ostree.readthedocs.io>) пытается усидеть на всех стульях. Позиционирующая себя как «*Git* для бинарников операционных систем», на самом деле это контентно-адресуемая файловая система, работающая поверх ext4, btrfs или чего-то еще, и атомарно переключающая деревья файловой системы root. Она интегрируется с загрузчиками, так что вы можете вы брать, какое дерево загружать, и менеджерами пакетов, которые

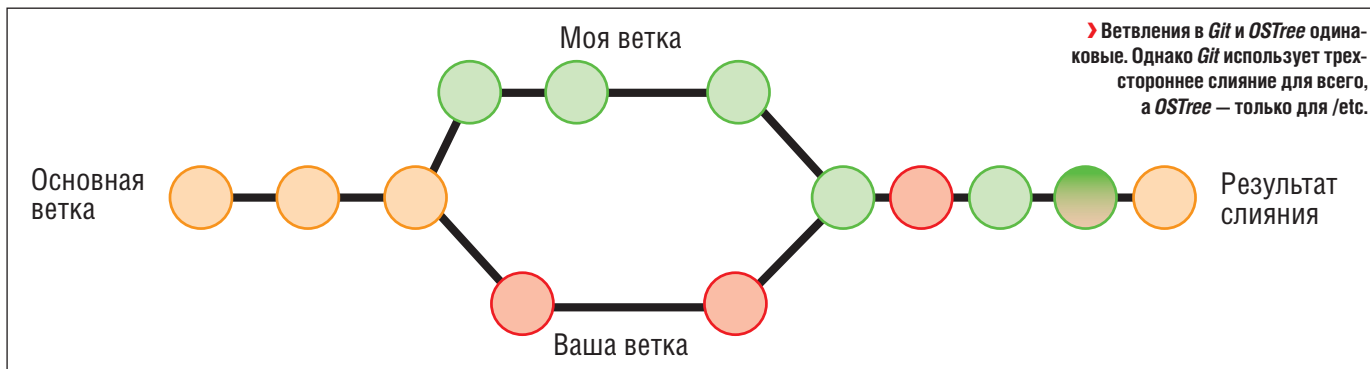
вы используете для создания этих деревьев. Она также обращается к File Hierarchy Standard (FHS) и предлагает вам хранить все бинарники ОС в */usr* (но не принуждает этого делать). */var* — директория общего пользования, так что вы можете сохранить состояние, например, базу данных или индивидуальных приложений в файловых системах. Для */etc* выполняется традиционное трехстороннее слияние при переключении деревьев. Это означает, что конфигурация всегда текущая.

## Git пересмотренный

Что такое трехстороннее слияние, спросите вы. Чтобы ответить на этот вопрос (и лучше понять *OSTree*), давайте сделаем краткий обзор *Git*. Репозиторий *Git* на самом деле — это ряд файлов в директории *.git* в вашей рабочей копии. Файлы в *.git* получают имя после того, как SHA-1 хэширует содержимое — вот почему это еще называется «контентно-адресуемым». Контент объекта дает вам ключ к базе данных *Git*, чтобы каждый объект сохранялся только однажды.

*Git* работает с несколькими типами объектов. Это двоичные объекты-блобы [blobs], которые являются содержимым файла. Это деревья, являющиеся почти директориями, контент и указатель на блобы. И есть коммиты [commits], которые связывают вместе деревья и метаданные, такие как имя автора или сообщение коммита.

Коммиты являются связанными родителями дочерних записей. Есть также ссылки, или сокращенно «refs», которые в той или иной



» Ветвления в *Git* и *OSTree* одинаковые. Однако *Git* использует трехстороннее слияние для всего, а *OSTree* — только для */etc*.

## Применение OSTree: Flatpack

Если вы пользуетесь Linux уже некоторое время, вы помните универсальные менеджеры пакетов, такие как *Autopackage*. Их было несколько, но не все они были успешны. Трудно объяснить это какой-то одной причиной, но, возможно, миру просто не нужно было возиться с лишним форматом пакетов.

Особенность Flatpack в том, что это не менеджер пакетов. Он ближе к Docker, поскольку предлагает вам почти независимые, готовые к работе приложения. Это явный отход от предыдущего положения дел, когда перед запуском приложения надо было

устанавливать все его зависимости (например, набор инструментов GUI). Идея объединения всего необходимого с самим приложением не нова, но Flatpack использует новые технологии (такие как *sgroups*, *namespaces* и *OSTree*), чтобы сделать это умно и безопасно.

Когда вы загружаете приложения как пакеты, вы приносите в жертву простоте управления объемом и безопасностью загрузки. Если библиотека пакета окажется уязвимой, у вас не будет одиночной точки обновления: вы всё предоставляете поставщикам приложения. Flatpack пытается сбалансировать

эти требования через среды исполнения, которые являются большими блоками с общим доступом, такие как те, на которых можно создать приложения платформ *Gnome* или *KDE*. *OSTree* является контентно-ориентированной файловой системой, поэтому каждый бляб хранится только однажды. Далее, общение между приложением Flatpack и операционной системой хоста ограничено, что сводит поверхность атаки к минимуму.

*Ubuntu Snap* походит на Flatpack, но предназначается скорее для серверов, чем для настольных систем. И с *OSTree* не имеет ничего общего.

степени являются символическими именами для хэшей коммитов. При создании ветви формируются новая цепь коммитов и ссылка, чтобы отслеживать ее конец. При слиянии двух ветвей вы объединяете обе цепи и делаете две эталонных точки на один коммит. Играть с *Git* на этом уровне очень интересно, и если вам это нравится, то вы найдете всё, что вам нужно знать, в Части 10 бесплатной книги *Pro Git* (<https://git-scm.com/book/en/v2>).

Вернемся к *OSTree*. Она очень похожа на *Git*, но не идентична. Так же, как и *Git*, *OSTree* имеет понятие репозитория; это просто директория для хранения объектов. Имеется один репозиторий системного уровня, размещенный в `/ostree/repo`. Для любого другого местоположения вам нужно предоставить утилите параметр `--repo` или настроить переменную среды `$OSTREE_REPO`. Далее, *OSTree* использует хэши SHA256 — они безопаснее (поскольку в SHA1 в 2014 г. были обнаружены конфликты) и длиннее. Между тем, *OSTree* не поддерживает сокращений (таких как `5fe1c78` для `5fe1c78faf2b430ab937db1cfaf9f3e16592aca3`), и вам всегда придется вводить хэш полностью. Здесь также есть ветви, но нет слияний. Там, где *Git* использует хитрый мини-язык пересмотров (см. `git-rev-parse(1)`), *OSTree* понимает только символ «крышки» (^), ссылающийся на предыдущий коммит.

Типы объектов в *OSTree* тоже напоминают *Git*. Оба имеют коммиты и объекты контента (блобы). То, что в *Git* именуется объектом «дерево», в *OSTree* разделено между `dirtree` и `dirmeta`. `Dirtree` сохраняет имя файла для хэш-преобразования, тогда как `dirmeta` содержит соответствующие метаданные, такие как `UID` и `GID`. Причина в том, что, будучи *Git* для операционных систем, *OSTree* должна хранить больше метаданных. Она делает это отдельно ради эффективности: если много файлов имеют одинаковый

расширенный список атрибутов (что случается часто), он не будет дублирован.

## Хранение блобов

Формат репозитория тоже немного другой. В *Git* блобы хранятся сжатыми. В *OSTree* — смотря по обстоятельствам. Для так называемых «голых» репозиториях файлы хранятся несжатыми, и *OSTree* «проверяет» их через жесткие ссылки. Репозитории-«архивы» хранят сжатые файлы и статические данные, что удобно для обслуживания образов ОС через HTTP. Есть несколько других форматов хранения, но в основном они являются разновидностями двух вышеприведенных.

И, конечно, *OSTree* предлагает собственный инструмент, именуемый так же — *ostree*, который — вы уже догадались — управляет этими репозиториями. Вот здесь-то и начинается самое интересное, что мы далее и обсудим.

Прежде чем начать, проверьте, установлена ли у вас *OSTree*. *OSTree* — это проект, поддерживаемый `Gnome/Red Hat`, так что в выигрыше те, кто использует `Fedora` или `CentOS`. Если *OSTree* еще не установлена, не волнуйтесь: она должна быть в репозиториях вашего дистрибутива. Для `Ubuntu` вам нужна 16.10 и выше. Чтобы получить самую последнюю и самую свежую версию, можете скомпилировать ее из исходника. Правда, мы на этом пути столкнулись с некоторыми проблемами, которые, разумеется, решаемы, но всё же неприятны, так что лучше оставьте это в качестве запасного варианта.

Если инструмент уже установлен, вот как вы создадите свой первый репозиторий *OSTree*:

```
$ ostree --repo=/path/to/repo --mode=bare init
$ ls /path/to/repo
config extensions objects refs state tmp
```

Здесь мы инициализируем голый репозиторий, что является настройкой по умолчанию. Учтите, что в реальном мире *OSTree* обычно работает от имени `root`, но поскольку мы не работаем на уровне дерева файловой системы ОС, нам вполне подойдет обычный пользователь.

Изначально репозиторий пуст. Давайте создадим несколько файлов и затем сделаем коммит:

```
$ mkdir folder
$ echo 'Hello, World' > file
$ ostree commit --branch=playground --repo=/path/to/repo
```

Здесь надо обратить внимание на некоторые вещи по сравнению с *Git*. Во-первых, нужна ветвь, потому что в *OSTree* нет эквивалента «`master` [основная ветвь]». Обычно у ветвей такие же имена, как у пути, скажем, `gnome-continuous/buildmaster/x86_64-runtime`, но давайте не будем усложнять. В противоположность, тема и тело коммита (краткое сообщение и длинное сообщение, на языке *Git*)

```
valesini@valesini-ubuntu:~/playground$ ostree --repo=/tmp/repo --mode=bare init
valesini@valesini-ubuntu:~/playground$ ls /tmp/repo
config extensions objects refs state tmp
valesini@valesini-ubuntu:~/playground$ mkdir folder
valesini@valesini-ubuntu:~/playground$ echo 'Hello, World' > file
valesini@valesini-ubuntu:~/playground$ ostree commit --branch=playground --repo=/tmp/repo --subject='Some des
9b5ab2d593f3ae1df236537151f10d87fb49a76e4486b33448646924d95f1d
valesini@valesini-ubuntu:~/playground$ echo 'Bye, World' > file
valesini@valesini-ubuntu:~/playground$ ostree commit --branch=playground --repo=/tmp/repo --subject='Some des
841c1a40f2372abcea5ed6c2a5f2240f722b8d54d8a78530e9d0e0b4aed503c2
valesini@valesini-ubuntu:~/playground$ ostree log --repo=/tmp/repo playground
commit 841c1a40f2372abcea5ed6c2a5f2240f722b8d54d8a78530e9d0e0b4aed503c2
Date: 2017-12-25 10:44:16 +0000
    Some descriptive subject
commit 9b5ab2d593f3ae1df236537151f10d87fb49a76e4486b33448646924d95f1d
Date: 2017-12-25 10:44:01 +0000
(no subject)
valesini@valesini-ubuntu:~/playground$ ostree checkout --repo=/tmp/repo 841c1a40f2372abcea5ed6c2a5f2240f722b8
valesini@valesini-ubuntu:~/playground$ ls -l
total 12
drwxrwxr-x 3 valesini valesini 4096 янв 1 1970 841c1a40f2372abcea5ed6c2a5f2240f722b8d54d8a78530e9d0e0b4aed503c2
-rw-rw-r-- 1 valesini valesini 11 дек 25 15:44 file
drwxrwxr-x 2 valesini valesini 4096 дек 25 15:43 folder
valesini@valesini-ubuntu:~/playground$ ls -l 841c1a40f2372abcea5ed6c2a5f2240f722b8d54d8a78530e9d0e0b4aed503c2
total 8
-rw-rw-r-- 2 valesini valesini 11 янв 1 1970 file
drwxrwxr-x 2 valesini valesini 4096 янв 1 1970 folder
valesini@valesini-ubuntu:~/playground$
```

➤ Посмотрите, как *OSTree* переустанавливает временные отметки на файлах, которые проверяет из репозитория.

## Nix: Совсем другое дело

*Nix* (<https://nixos.org/nix>) никак не использует *OSTree*. Это менеджер пакетов для Unix. Но при этом у него есть ряд общих целей с *OSTree* и нечто близкое по духу. Так что имеет смысл взглянуть на *Nix*, чтобы сравнить оба подхода.

*Nix* позиционируется как «чисто функциональный менеджер пакетов». Для тех, кто не следует тенденциям в языках программирования: это означает, что при сборке пакетов нет побочных эффектов, и сами пакеты неизменные и определенные. *Nix* также хэширует график зависимостей компиляции и таким образом обеспечивает изоляцию пакетов. Поэтому у вас может быть установлено рядышком

две версии KDE. Однако *Nix* не является контентно-адресуемой файловой системой.

*Nix* поддерживает как компиляцию из исходника, так и двоичную установку. Иными словами, он может скомпилировать для вас всё, включая компилятор, или использовать кэшированные двоичные пакеты. Каждый пакет устанавливается в `/nix/store`, и использование криптографического хэша в качестве части имени означает, что вы можете атомарно повысить или понизить версию, как в *OSTree*. Однако это работает на уровне пакетов, тогда как *OSTree* работает с целыми деревьями файловой системы.

Официально *Nix* поддерживает Linux и Mac OS X. Есть также — как могут сказать некоторые — нишевый дистрибутив Linux, `NixOS`, который, как вы уже догадались, использует *Nix* в качестве официального менеджера пакетов. Инструмент заодно предлагает удобное средство управления средами компиляции. Если вы когда-нибудь использовали для этой цели *Docker*, предлагаем вам взглянуть поближе.

И, наконец — но это тоже очень важно — если вам это интересно, черкните нам пару строк, чтобы мы могли рассказать о *Nix* в будущих выпусках Рецептов.



по умолчанию опциональны, и можно прекрасно сохранять пустую директорию. Кроме того, помните, что здесь не требуется промежуточной стадии `ostree add`.

Теперь изменим кое-что в директории:

```
$ echo 'Bye, World' > file
```

и снова сделаем коммит к той же ветви. На сей раз добавим описательную тему. Теперь вы можете видеть лог своих изменений с помощью:

```
$ ostree log --repo=/path/to/repo playground
```

Вывод выглядит почти как в *Git*. Помните, что опять требуется ссылка [ref], поскольку здесь нет мастера. Вы можете проверить любую версию этого крошечного дерева:

```
$ ostree checkout --repo=/path/to/repo <hash from ostree log>
```

Помните, что вам нужны все байты хэша. Заметьте, что дерево не включено, как в *Git*, а проверяется в отдельной директории. В реальном мире используется интеграция с загрузчиком, и для проверки нужного вам дерева во время загрузки применяются трюки с *systemd*.

Поддерживается также выгрузка в текущую директорию: просто добавьте `!` в качестве последнего аргумента командной строки. Однако это работает не так, как в *Git*. На самом деле, если вы попробуете это с нашим примером, то *OSTree* будет жаловаться:

```
$ ostree checkout --repo=/path/to/repo <hash from ostree log> .
error: File exists
```

Причина в том, что у нас уже есть и "file" и "folder" в текущей директории, а в *OSTree* важны неизменные деревья только для чтения. Есть функция `union (--union)`, которая просит *OSTree* помещать файловые системы одну поверх другой. Сохраняются все файлы и директории, которые *OSTree* считает неизменными, а всё остальное переписывается контентом коммитов. В основном это применяется для многоуровневого расположения деревьев, как *Docker* поступает с контейнерами. Это устраняет вышеописанное сообщение об ошибке.

Однако помните, что *OSTree* не предназначалась для изменяемых деревьев файловой системы. Если вы возьмете что-то из голого репозитория, вы на самом деле получите жесткую ссылку — т.е. второе имя для блоков на диске. Если вы изменяете что-то под этим именем, вы соответственно изменяете репозиторий напрямую и переписываете свою историю. Такого вам не надо, и чтобы это предотвратить, скрипт интеграции *OSTree* создает ограничение «только для чтения», монтируемое для управляемых *OSTree* директорий при использовании в реальном мире. Также и поэтому `/etc` и `/var` — два типичных местоположения для изменяемых файлов в Linux — обычно находятся вне контроля *OSTree*.

Можно также сделать коммит на дерево из tar-архива. Вы просто добавляете `--tree=tar=something.tar` в команду `ostree commit`. Это очень удобно при интеграции *OSTree* с системами компиляции. Давайте быстренько посмотрим, как это сделать.

## Нечто реальное

*OSTree* проложила себе дорогу во многие проекты. Flatpack (<https://flatpak.org>) использует ее для распространения и управления приложений и сред исполнения и считает дедубликацию контента и откатку к предыдущей версии ведущими функциями. *OSTree* также используется как механизм обновления в Endless OS (<https://endlessos.com>) и формирует основу для Project Atomic ([www.projectatomic.io](http://www.projectatomic.io)). Фактическим стандартом компиляции/поточковой системной интеграции для *OSTree* является, вероятно, *Gnome Continuous* (<https://build.gnome.org>).

При этом *Gnome Continuous* является экспериментальным, и, похоже, на момент написания он был в нерабочем состоянии. По крайней мере, все последние сборки были помечены как «неудавшиеся». У нас также возникли проблемы с образом, которому был год и который был выдан на скачивание. Однако скрипты

интеграции по-прежнему с нами, и мы можем видеть, что они делают и как организована типичная управляемая *OSTree* система.

Компиляция системы проста, по типу *OSTree*. Она берет исходники с *Git*, компилирует их и затем фиксирует результат в репозитории *OSTree*. Здесь важно отметить, что именно *Git*, а не *OSTree*, должен быть изначальным исходником в этой схеме. В *OSTree* предполагается, что файлы, хранящиеся в репозитории, могут быть при необходимости воссозданы. Есть также механизм метаданных, пригодный для сохранения дополнительной информации, чтобы помочь этому воссозданию: например, хэш или tag коммита *Git*.

Обычно вы держите несколько последних коммитов *OSTree* и удаляете всё остальное. *Gnome Continuous*, похоже, не использует эту схему, но ее стоит рассмотреть для вашего собственного употребления. Ман-страница `ostree-prune(1)` объяснит всё в подробностях. С помощью `ostree prune` можно удалить (убрать мусор) недостижимые объекты, всё, что старше установленного порога, или просто определенные коммиты.

С необходимыми бинарниками в репозитории *OSTree* скрипт компиляции начнет создавать т.н. развертывание. Последнее — это просто отладка *OSTree*, и во многом работает как *chroot*. Для этого применяется семейство команд, привязанное к `ostree admin`. Сперва вызывается `ostree admin os-init`, для подготовки нового развертывания. Затем `ostree pull-local` качает данные из репозитория сборки системы. И, наконец, `ostree admin deploy` налаживает целевую ссылку-ref как новое развертывание по умолчанию, чтобы оно вступило в силу после перезагрузки.

*Gnome Continuous* поставляется в виде образа виртуальной машины (qcow2), и структура файловой системы внутри тоже типична для *OSTree*. `/bin`, `/lib` и т.п. являются символическими ссылками на их аналоги в `/usr`, и последний монтируется только для чтения. Точно так же `/home` и `/root` являются символическими ссылками на `/var`, чтобы к ним имели доступ все развертывания *OSTree*. Есть также директория `/sysroot`, которая указывает на реальный корень файловой системы. В основном это для обеспечения доступа к системному репозиторию *OSTree*, чтобы вы могли с ним работать.

Надеемся, это поможет вам понять, что такое *OSTree* и для чего это. Вам не часто придется взаимодействовать с ней, но она может выступать в роли незаметной рабочей лошади в больших системах. На самом деле, сейчас имя проекта — `libostree`, в ознаменование того факта, что его надо интегрировать в Python или нечто подобное. Но если вам нужен старый добрый CLI, не паникуйте — он по-прежнему есть и работает. **LXF**

► *Gnome Continuous* предназначался для того, чтобы попробовать самый свежий *Gnome*, но на данный момент вроде бы вышел из строя.



# ГНУ/Линуксцентр

*Ваш поставщик свободного программного  
и аппаратного обеспечения*

**Комплекты  
легализации СПО**

**Дистрибутивы  
GNU/Linux и СПО**  
на DVD и загрузочных  
флэшках

**Дистрибутивы  
GNU/Linux  
и СПО**  
с сертификатами  
ФСТЭК, ФСБ  
и Минобороны

**Межсетевые  
экраны**  
с сертификатами  
ФСТЭК, ФСБ  
и Минобороны



**Свободное  
аппаратное  
обеспечение**  
Arduino, oLinuxino,  
Cubieboard, Raspberry Pi,  
Intel Edison, Digilent,  
3D-принтеры  
и робототехнические  
конструкторы

**Аппаратное  
обеспечение  
с прошивками  
на базе СПО**

**Обучающая  
литература**

**Атрибутика**

## Фирменный магазин и сервис-центр

*Санкт-Петербург, пр. Медиков, 5, корп. 7*

*+7 812 309 06 86 | [www.linuxcenter.ru](http://www.linuxcenter.ru)*





Наши эксперты помогут вам с любым приложением Linux!



**ЕВГЕНИЙ БАЛДИН**  
Подтвердивший  
свою квалификацию  
физик.

## Средство, а не цель

Я думала, астронавты выше политики, выше войн и гравитации.  
*Танабэ. Серия 1. Planetes*

**К**огда в конце 1990-х Ричард Столлмен создавал лицензию GPL, цель у него была одна: гарантировать свободному ПО выживание в мире патентного права. Подразумевалось, что GPL — всего лишь средство обеспечения свобод пользователей, построенное под наш несовершенный мир. Далее жизнь показала, что мир несовершеннее, чем он думал, и сама GPL поменялась дважды. Одно из изменений, появившееся в GPL3 — пощажка для нарушителей на 30 дней, за которые им следует исправить нарушение. Ведь часто люди нарушают лицензию не со зла, а из-за лени или некомпетентности, и тут же побивать их камнями излишне. В GPL2 на исправление нарушений формально времени не давалось — сразу можно было подавать в суд, были ли ответчики злостными и последовательными нарушителями или просто неграмотными. Обе эти причины нарушений предосудительны, но с точки зрения лучшего для сообщества разница есть, ибо неграмотных можно и нужно обучать, даже если они неосознанно сопротивляются. К чему эти слова? А к прецеденту судебного лицензионного троллинга от бывшего лидера проекта Netfilter (межсетевой экран, встроенный в ядро GNU/Linux и, значит, подпадающий под GPL2) Патрика Мак-Харди (Patrick McHardy). Патрику это принесло добрых €2 млн, но сообществу от такого подхода нет ничего, кроме вреда!  
[E.m.Baldin@inp.nsk.su](mailto:E.m.Baldin@inp.nsk.su)



## В этом месяце вы научитесь...

- Смотреть кино в терминале ... 52**  
**Джон Найт** почти уверен, что вы этим пользоваться не будете, и тем не менее берет на себя труд продемонстрировать вам такие возможности *MPlayer*, о которых вы и не подозревали.
- Снимать круговые панорамы ... 54**  
По мнению **Адама Оксфорда**, видео с обзором 360° достигло той стадии, когда уже пора заняться им вплотную. Вот вам базовая форма виртуальной реальности.
- Обдирать DVD ... 58**  
Подчиняясь **Джону Найту**, *MEncoder* преобразует DVD в удобный для вас формат видеофайла, не выходя из терминала. Графической интерфейс — это же для слабаков!
- Ловить воров ... 60**  
Печальная статистика гласит, что ноутбуки и мобильники частенько становятся добычей похитителей. Но **Маянк Шарма** всегда в курсе местонахождения своих устройств.
- Ругать случайностью ... 62**  
Генерация действительно непредсказуемых битов данных лежит в основе важнейших операций, таких как надежное шифрование web-трафика. **Нейт Дрейк** включает гелевый светильник...
- Управлять сетью ... 64**  
**Маянк Шарма** умудрился запустить сервер внутри виртуальной машины! А все благодаря возможностям *VirtualBox*, позволяющим эмулировать всякие сетевые настройки. Отыскалось место даже для брандмауэра с двумя виртуальными сетевыми адаптерами.
- Настраивать роутер ... 66**  
Обзаведясь специализированным дистрибутивом Linux, **Джон Лэйн** устанавливает на свой роутер прошивку с открытым кодом и обнаруживает, что этот роутер лучше, чем казался.
- Собирать тулчейны ... 72**  
**Дмитрий Кузнецов** убедился в правоте изречения «суха теория, а древо жизни пышно зеленеет», когда попытался самостоятельно собрать компилятор. Кавалерийским наскоком тут не отделаться.

## АКАДЕМИЯ КОДИНГА



И новичкам, и гуру!  
Всегда полезно будет познать  
нечто доселе неведомое

**Собери сам ... 76**  
**Михалис Цукалос** готовит прочный фундамент для вашей разработки приложений и визуализирует зависимости. Отсюда и до вышеупомянутых тулчейнов недалеко.

**Списки в Python ... 80**  
Чтобы не позабыть свои любимые блюда, **Нейт Дрейк** составляет их перечень, после чего переходит к массивам. Не едим, так поглядим...

**Пурпур Японии ... 84**  
Увлечение **Нейта Дрейка** антикварными шифрованными машинами вылилось в статью про Python-имитацию очередного проекта времен II Мировой.

# MPlayer: Видео в терминале

Джон Найт решил уговорить трусоватых пользователей мультимедиа попробовать терминал с удивительными возможностями *MPlayer*.



**Наш эксперт**

**Джон Найт**  
Когда Джон не играет в видеоигры на французском языке, его обычно можно заставить за избиванием большого барабана до тестового состояния.

**M**Player имеет графический интерфейс, и зачем возиться с ним в командной строке? А дело простое: объединение *MPlayer* с терминалом дает возможности, неосуществимые в графическом интерфейсе. Как мама объясняет, почему вам обязательно есть овощи (пока вы в конце концов не сдадитесь!), так и мы введем вас в более крутые способы.

Использовать *MPlayer* без особо причудливых настроек или опций легко. Команда со всеми значениями по умолчанию выглядит так:

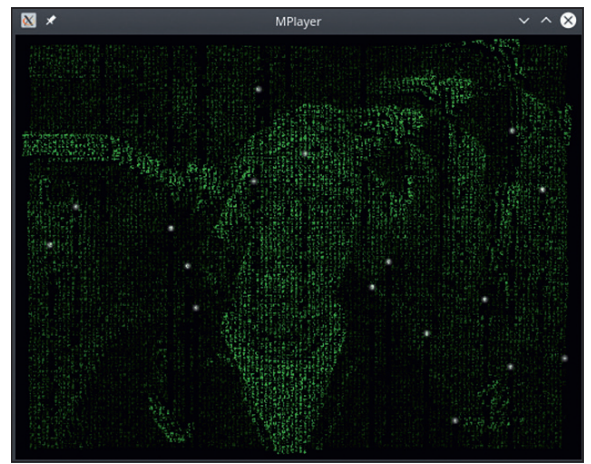
```
$ mplayer thisisanexample.avi
```

Если вы раньше не использовали *MPlayer*, вам следует осознать, что *MPlayer* полностью управляется с клавиатуры. Сначала это кажется странным, но в таком безумии есть свой резон: прокрутка видео намного быстрее, чем с обычным графическим интерфейсом, управляемым мышью, и если вы ищете какой-то эпизод в фильме, можно несколькими нажатиями кнопок за несколько секунд перейти в желаемое место.

Для начала: F — полноэкранный режим, Пробел — пауза, Q — выход, а клавиши со стрелками — переход вперед и назад на одну минуту (вверх и вниз) или на десять секунд (влево и вправо). Page Up и Page Down — с десятиминутным шагом, а громкость управляется клавишами \* и / на клавиатуре. И это всё, что нужно для начала.

Клавиатурный интерфейс *MPlayer* поначалу кажется странным, но вся его организация и эстетика создают ощущение некой специальной тактильной машины, типа станции видеопроизводства или монитора безопасности, а не просто очередного обычного приложения для видео на скучном ПК. Навигация по видео таким образом выполняется чрезвычайно быстро.

Теперь стоит познакомиться с кнопкой O, которая включает индикацию на экране (сокращенно OSD) тем идеальным машинным шрифтом, что вызывает воспоминания о студийном проигрывателе Betamax 1990-х гг. — и это еще больше усиливает ощущение работы на настоящей спецмашине. Однократное нажатие O включает некоторые элементы экранного меню, например, индикатор выполнения, активирующийся при поиске. Второе нажатие вызовет



➤ **Богатая история *MPlayer* означает, что для него были созданы всевозможные интересные модификации и дополнения. Вот видео, проходящее через фильтр "matrixview".**

простой таймер, третье — покажет таймер по всей длине трека, а четвертое нажатие снова отключит экранное меню.

Давайте перед дальнейшим изучением элементов управления рассмотрим некоторые виды мультимедиа. Воспроизведение DVD весьма просто, хотя, возможно, и необычно. Без меню вам надо вручную пробираться по номерам заголовков DVD в поисках желаемого. Для воспроизведения с заголовка [title] 1 команда выглядит следующим образом:

```
$ mplayer dvd://1
```

Вариант «перепробовать все заголовки» не так страшен, как кажется: фильм обычно помещается под заголовком 1 или 2, и если вы получили только раздражающий экран предупреждения, попробуйте заголовок 2. Дополнительные функции обычно хранятся под более высокими номерами.

Это может выглядеть лишней тратой усилий, но вспомните обо всяческих меню и предупреждающих экранах, через которые

## Краткий справочник по командам MPlayer

<b>Пробел</b> .....	Воспроизведение/Пауза	<b>7 и 8</b> .....	Насыщенность
<b>Стрелка вверх/вниз</b> .....	Переход на 1 минуту вперед/назад	<b>9 и 0</b> .....	Вторичный регулятор громкости
<b>Стрелка вправо/влево</b> .....	Переход на 10 секунд вперед/назад	<b>F</b> .....	Полноэкранный режим
<b>Page Up / Page Down</b> .....	Переход на 10 минут вперед/назад	<b>Enter</b> .....	Закрывает текущий файл
<b>Воспроизвести DVD</b> .....	\$ mplayer dvd://1	<b>Q</b> .....	Закрывает MPlayer
<b>Воспроизвести CD</b> .....	\$ mplayer cdda://	<b>O</b> .....	Индикация на экране
<b>1 и 2</b> .....	Контрастность	<b>V</b> .....	Показать субтитры
<b>3 и 4</b> .....	Яркость	<b>*</b> .....	Увеличить громкость
<b>5 и 6</b> .....	Цветность	<b>/</b> .....	Уменьшить громкость



## Крутые штуки, не проходите мимо!

Иногда *MPlayer* бывает «сливом кода», но у него есть целый ряд блестящих и необычных способностей.

С более консервативной стороны, его опции `-zoom` и `-loop` делают *MPlayer* идеальным для рекламных дисплеев и художественных инсталляций (возможно, особенно эффективным в сочетании с Raspberry Pi?).

Вот видео, которое масштабируется и воспроизводится в замкнутом цикле:

```
$ mplayer test.avi -zoom -xy 1600 -loop 0
```

Смеха ради вы можете выводить свои видео в ASCII — теперь это клише. Если вы хотите применить небольшую уловку, чтобы всё выглядело так, как «Матрица», подайте вывод видео на 'matrixview' с помощью параметра `-vo`:

```
$ mplayer -vo matrixview test.avi
```

На самом-то деле это не ASCII, а видеофильтр! Для истинного видео ASCII (что особенно удобно, если вы работаете без X), попробуйте видеовыход «caca»:

```
$ mplayer -vo caca test.avi
```

Говоря о терминале: если вы везунчик, можете использовать поддержку фрейм-буфера для воспроизведения видео в своем терминале без X. Сначала попробуйте

```
$ mplayer -vo fbdev test.avi
```

Не отчаивайтесь, если это не сработает сразу, поскольку вас могут выручить режимы вывода `fbdev2` и `directfb`. В противном случае ознакомьтесь с превосходным руководством Дэна [Dan] на TechieStuffs по запуску видео в консоли Linux: <https://tinyurl.com/yc4yudkn>.

обычно нужно пробираться на DVD, а с *MPlayer* вы обычно попадаете на фильм за нескольких секунд (и велик ли труд попробовать 2, если 1 не работает?). Отличный способ мигом отыскать пасхальные яйца!

Начав воспроизведение, вы можете настроить фильм на лету с помощью нескольких впечатляющих элементов управления видео. Клавиши 1 и 2 регулируют контрастность, 3 и 4 — яркость, 5 и 6 — цветность, а 7 и 8 — насыщенность цвета. А когда вы смотрите иностранный фильм или нуждаетесь в субтитрах, V включает показ субтитров, а J в цикле предлагает доступные языки.

Приверженцы консоли без X будут рады узнать, что *MPlayer* работает и как проигрыватель компакт-дисков; он прост в использовании, а поиск с помощью клавиш со стрелками — отличный способ навигации по компакт-диску. Чтобы воспроизвести компакт-диск с самого начала, просто введите

```
$ mplayer cdda://
```

Для указания трека добавьте номер трека в конец команды:

```
$ mplayer cdda://6
```

## Подробнее

Всё это хорошо и замечательно, но чем это полезней графического интерфейса? Ну, *MPlayer* показывает себя при взаимодействии с оболочкой Linux в целом: каналы, скрипты, дампы информации, вот это вот всё.

Начнем с чего-нибудь попроще. Допустим, вы хотите воспроизвести все видеофайлы в папке. Это легко сделать, набрав

```
$ mplayer *
```

При запуске этой команды клавиша Enter закрывает текущее видео, если вы хотите перейти к следующему.

Возможно, у вас огромная библиотека фильмов или музыкальных клипов, и вы ищете некое видео, но помните только первую букву названия — допустим, оно начинается с W. Помня, что Linux чувствителен к регистру, вы можете воспроизвести все видео, начинающиеся с W в любом регистре, введя

```
$ mplayer W* w*
```

Если вы собираете и храните музыкальные клипы локально, у вас, вероятно, есть огромная папка с произвольными названиями клипов. И если надо перейти прямо к некоторым роликам определенной группы, без необходимости сортировать сотни файлов, терминал очень полезен.

Предположим, вы хотите проиграть все локально сохраненные видео группы Audioslave. Вы не знаете, начинаются ли файлы с буквы верхнего или нижнего регистра, и есть приличный шанс, что имена файлов даже не начинаются с названия группы (например, начинаются с цифр). Это можно обойти поиском середины слова, поместив звездочку с обеих сторон. Пропуская A в начале, эта команда теперь будет воспроизводить любые клипы Audioslave, независимо от того, как начинается или заканчивается имя файла:

```
$ mplayer *udioslave*
```

## Усложним

Конечно, колонка терминала не будет колонкой терминала без старого доброго канала, и *MPlayer* — отнюдь не исключение! Для непосвященных: каналом можно подключить одну программу — или даже много — к другой по цепочке, подав выход одной программы на вход следующей программы.

Конечно *MPlayer* может получать файл непосредственно с его URL-адреса, но мы хотим продемонстрировать его функциональность с каналом. Итак, мы выбрали простой пример, подключая вывод программы загрузки *wget* ко входу *MPlayer*. Синтаксис немного озадачивает: параметр `-O` явно означает «вывод», но эти фантомные дефисы странноваты! Так или иначе, здесь мы транслируем публичный тестовый клип с сайта Cinelegra, демонстрируя некоторые видеоматериалы итальянского телевидения:

```
$ wget https://cinelerra-cv.org/footage/rasseгна2.avi -O - | mplayer -
```

*MPlayer* также удобен для загрузки потокового файла в локальное хранилище. На минутку забудем про каналы и просто просмотрим файл напрямую, но здесь мы «сбрасываем», сохраняем тестовое видео Cinelegra в локальный файл:

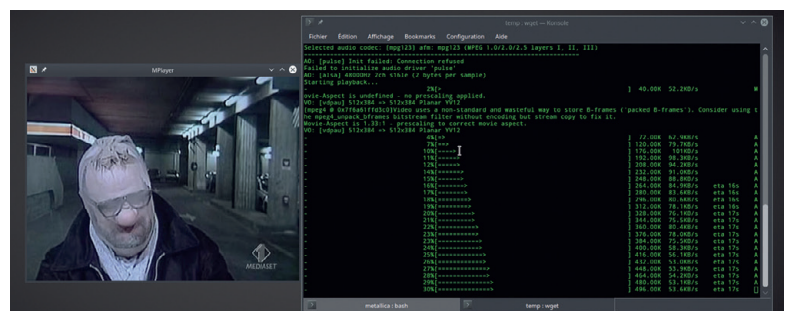
```
$ mplayer https://cinelerra-cv.org/footage/rasseгна2.avi -dumpstream -dumpfile dumped.avi
```

Обратите внимание, что ссылка идет сразу после команды; параметры `-dumpstream` и `-dumpfile` указывают *MPlayer*, что это потоковый интернет-файл, который будет сохранен локально; и, наконец, имя файла указывает место, куда вы хотите его «сбросить». Эта функциональность особенно полезна для старого потокового контента с конца 1990-х и 2000-х гг. (в частности, архивных радиопрограмм), и стоит отметить, что в документации в качестве примера используется старый формат файла `.asf`!

Итак, да, мы знаем, что у *MPlayer* несколько странный способ делать что-либо — и да, вы всё равно предпочтете программу с графическим интерфейсом. Но, надеюсь, мы показали некоторые вещи, о которых вы не знали, на что способен терминал, и ведь это было не так уж сложно? **LXF**



*MPlayer* имеет несколько фильтров деинтерлейсинга для исправления видео с этим жутким «морщинистым» видом. Нам особенно нравится Yadif, который активируется добавлением `add -vf yadif` в свою команду.



► *MPlayer* можно дополнить инструментами командной строки, такими как каналы. Здесь *MPlayer* напрямую воспроизводит видеопоток, который тут же захватывает *wget*.





## Запись видео

Правила для 360-градусного видео пока что не написаны, поэтому стилистически вы будете изобретать их на ходу. Однако есть несколько практических аспектов, которые стоит учесть при съемке, ибо они помогут вам отснять отличный материал.

Помните, что с какой бы камерой вы ни работали, 360-градусные видео монтируются из нескольких картинок практически так же, как панорамный снимок на камере или телефоне. Вероятнее всего, в местах соединения вы увидите довольно странные эффекты.

Самые очевидные области — выше и ниже камеры. В зависимости от модели у вас могут получиться странные артефакты, черные точки или невидимые области. При съемке обычного видео вам нужно подумать о кадрировании съемки для удобства зрителя. В 360-градусном видео эквивалентом будет подумать о том, что вам делать с этой мертвой зоной. Некоторые закрывают ее логотипом или используют, чтобы скрыть саму камеру.

Это не единственные ошибки на стыках, которые можно увидеть в 360-градусных видео, но хитрость

минимизации остальных в том, чтобы не подходить ни к чему слишком близко. Чем ближе объект, тем больше времени понадобится алгоритму сшивания на совмещение двух половин общего изображения. И даже если они совпадут, скорее всего там всё равно будет солидное искажение от объектов — особенно если на вашем устройстве съемки две камеры, а не три или четыре.

Кроме того, старайтесь не снимать в замкнутом пространстве или помещать камеру на низко расположенные стол или плоскую поверхность.

По большей части цель таких видео — дать зрителю ощущение, что он в чем-то участвует; они необработанные, неотредактированные и даже не предпринимают попыток изобразить нечто в подробностях, как рассказ.

Есть редкие исключения. Создатели *Мистер Робот* отредактировали ряд сцен из первой серии в виде 360-градусного видео. В отличие от большинства 360-видео, камера не всегда дает зрителю вид от первого лица: скорее она позволяет им исследовать сцену во время ее воспроизведения. Это очень интригующе — хотя и необычно.

У 360-градусного видео определенно есть потенциал — это логическое развитие для создателей фильмов, которые хотят контролировать то, что видит их зритель — но это территория пока еще более или менее неисследованная и вполне готовая показать любителю, на что она способна.

## Съемка 360-видео

Есть три разных способа сделать 360-градусные сцены. Первопроходцы строили установки, на которые монтировали несколько камер, а потом объединяли отснятый ими материал в пакетах редактирования рабочего стола; но с тех пор всё изменилось.

Самый простой способ записать 360-градусное видео — с помощью специальной камеры, например, Ricoh Theta S или Samsung Gear 360. Спереди и сзади у них имеются широкоугольные объективы, которые охватывают почти-но-не-совсем 180 градусов с каждой стороны камеры, и две видеосъемки монтируются для создания сферического отображения на самой камере. Профессиональные 360-градусные камеры часто имеют третий объектив, направленный вверх, чтобы избежать слепых зон.

Специальные устройства по-прежнему недорогие, поэтому камеры, которые работают с телефоном, являются неплохой заменой. Мы использовали Insta360 Air, которая подключается к USB-порту на телефоне. Она менее удобна в использовании, чем специальная камера, но зато почти на две трети дешевле. Конечные результаты превосходные, приложения телефона соединяют видео, что позволяет вам смотреть другие 360-градусные видео и экспортировать финальный клип в формат, который можно использовать на настольном ПК.



» Превосходная Insta360 Air идет с резиновым корпусом для защиты объективов, когда вы держите ее в сумке.

Наконец, вы можете снимать 360-градусные фотографии с помощью телефона или камеры, сделав много фотографий вокруг и соединив их с помощью программы работы с изображениями. Это отличный (и бесплатный — см. врезку вверху) способ начать работу, но это не видео, и потому не подходит для наших целей.

В Linux редки приложения, способные достойно работать с 360-видео. Например, *Blender* умеет создавать 3D-миры и экспортировать пролеты в качестве 360-видео, но менее удобен для добавления элементов в уже отснятый фильм.

К счастью, 3D-видео можно редактировать и монтировать почти в любом обычном нелинейном пакете редактирования видео. Однако вам придется установить отдельное ПО для предварительного просмотра отредактированного материала, и тут есть выбор из нескольких опций.

Проект с открытым кодом *Valiant 360* (<https://github.com/flimshaw/Valiant360>) превратит ваш браузер в программу просмотра 3D-видео, если вы готовы настроить его как локальный плеер. Если хотите нечто более простое и сразу готовое к использованию, ребята из GoPro предлагают Deb-пакет *GoProVRPlayer* ([www.kolor.com/gopro-vr-player](http://www.kolor.com/gopro-vr-player)), который безукоризненно работает на большинстве производных Debian.

Среди плееров с открытым кодом — версия 3 плеера *VLC* поддерживает 360-градусные видео, хотя в нем нет таких же опций просмотра, как в *GoProVRPlayer*. Однако пока что она не включена в репозитории дистрибутивов. Чтобы у вас была последняя сборка (*VLC 3.0.0-git Vetinari*) в Ubuntu, например, вам нужно добавить «ночной» PPA: `sudo add-apt-repository ppa:videolan/master-daily && apt install vlc`

В целом *GoProVRPlayer* выдал наилучшее качество воспроизведения и внутреннего управления видео. Смонтировав видео, если вы решите поделиться им со всем миром, вы захотите загрузить его на Google, Facebook или Vimeo и т. п. Все поддерживают общий доступ и просмотр 360-градусных видео в их собственных встроенных в браузеры плеерах.

## Астральные проекции

Чтобы начать редактировать свое видео, надо переместить его из вашего телефона на настольный ПК. Это делается по-разному на разных камерах, но принцип обычно одинаковый. Например, в приложении *Insta360* требуется экспортировать видео в вашу галерею Videos с помощью встроенного

### Скорая помощь

Работа с видео занимает много места на жестком диске. Работа с 360-градусным видео занимает и того больше. Вероятно, вам надо будет сделать две копии конечного файла, чтобы добавить теги метаданных для YouTube и т. д.

### Скорая помощь

Вероятно, ваша камера или телефон будут снимать 360-градусные видео в разрешении 4K. Не бойтесь делать рендеринг финального видео в меньшем разрешении, чтобы уменьшить размер файла. Ваши зрители вам простят, если видео хорошее.

## Скорая помощь

При загрузке 360-градусных видео на YouTube они будут воспроизводиться как обычное HD-видео. Отображение эквидистантного видео на сферу может занять до часа.

в приложение инструмента конвертирования в “Share”. Возможно, вы обнаружите, что есть несколько опций для экспортированного файла — в частности, относящиеся к точке обзора, которая может быть “Tiny Earth”, “Spherical”, “Fish-eye” или “Equirectangular”.

Если это ваши опции, выбирайте последнюю. Это наиболее общий и самый простой в работе формат, и он экспортирует 3D-видео в виде обычного файла MP4 2D. Это будет в супер-широкоформатном соотношении 2:1, и вам нужно выбрать наивысшее возможное разрешение (обычно это 4K).

## На компьютере

Эквидистантные проекции сплющивают сферы в прямоугольник, сохраняя точными относительные размеры. Поэтому при просмотре мир, который они изображают, выглядит как два изображения «рыбьего глаза», склеенные вместе. Скругленное в шар, всё обретает нужные пропорции. Лучшие примеры эквидистантных проекций относительно других являют собой карты Земли.

Подумайте о том, как глобус делится на линии широты и долготы. По мере приближения к полюсам физический размер каждого участка, ограниченного линией долготы и широты, становится меньше, потому что кривизна Земли заставляет меридианы сходиться, и это означает, что у параллелей остается всё меньше пространства для охвата.

Это создает проблему для картографов. Как представить кривую линию на плоской карте в виде прямоугольника? Некоторые карты — например, знаменитая проекция Меркатора — делают это, срисовывая каждую «клеточку» глобуса в виде клеточки на карте такого же размера. Если так делать, преувеличивается размер массы суши, так что крошечная Гренландия кажется географически такой же, как Африка, то есть достаточно большой, чтобы поглотить Китай, США и всю Европу.

Эквидистантные карты пытаются компенсировать это, сжимая соответствующим образом клеточки широты и долготы. У них более сжатый вид сверху и внизу, и они заставляют массу суши около полюсов выглядеть меньше, чем на глобусе. Если вы правильно экспортируете свое видео, оно будет выглядеть примерно так же.

## Пакет редактирования

Ваши 360-градусные видео с эквидистантным соотношением сторон и обычным цифровым форматом вы можете начать склеивать в любом пакете для видеоредактирования. Это, на самом деле, весьма просто: всё, в чем вам нужно убедиться перед началом — одинакова ли начальная ориентация камеры в каждом из редактируемых вами клипов. Зрители могут смотреть вверх и вниз, как только видео начинается, но переключение между потолком и полом на полпути будет очень дезориентирующим.

Но как только вы склеили требуемый вам ролик, всё немного усложняется. Лучше всего отрендерить окончательную версию как файл MKV или MP4, поскольку они имеют широкую поддержку в программном обеспечении и облачных сервисах. На самом деле, некоторые онлайн-сервисы, например, YouTube, принимают 360-градусные видео только с расширением файла MP4.

Поскольку 360-градусный формат — нечто новое и по сути не поддерживаемое какими-либо редакторами рабочего стола в Linux, вашему окончательному файлу будет не хватать некоторых метаданных, которые определяют его как 360-градусный фильм.

Но *GoProVRPlayer* это не волнует — он перевернет любой данный ему файл в 3D, в отличие от YouTube и других онлайн-сервисов, которые, при отсутствии должных метаданных (похожих на EXIF-информацию в статическом изображении), воспроизведут файл только в его плоском эквидистантном формате, если вы не определите его правильно.

К счастью, есть простой способ сделать это: инженеры Google создали инструмент под названием *Spatial Media Metadata Injector*. Возьмите файл из его архива Github (<https://github.com/google/spatial-media/tree/master/spatialmedia>) и извлеките его в подходящее место в вашей папке **Home**. Скажем, `~/Injector`.

Вам также понадобится установить Python из ваших репозитивов дистрибутива, если вы уже не сделали это, командой `sudo apt install python`.

После извлечения *Spatial Media Metadata Injector* использовать его просто. Откройте терминал и введите команду `python ~/Injector/spatialmedia -i ~/path/to/input.mp4 ~/path/to/output.mp4`, где **input.mp4** — это ваше отредактированное видео, а **output.mp4** — исправленная версия с добавленными новыми метаданными.

И это, в принципе, всё. Создание 360-градусных видео во многом идентично созданию обычных 2D-видео всего лишь с парой моментов, за которыми нужно следить до и после редактирования. Следующий шаг — подумать, как записать и отобразить в 360-градусных видео звук с нескольких микрофонов. Однако это будет уже совсем другой учебник... **LXF**



Как показывает наш опыт, *GoProVRPlayer* предлагает лучшее качество 360-градусных видео, хоть он и не с открытым кодом.

## Поработайте с неподвижностью

Если вы еще не решили, подходят ли вам 360-градусные видео, попробуйте сделать несколько 360-градусных фото с вашего телефона, чтобы посмотреть, зацепило ли это вас.

Существует великое множество приложений — включая приложение для камеры Android по умолчанию, если ваш производитель не заменил его, или же приложение Google Street View — которые позволят вам поместить перед собой телефон и нанести на идеальную сферу изображения

всего, что расположено вокруг того места, где вы находитесь.

Приложение Street View, как видно из названия, дает возможность загрузить 360-градусные снимки прямо в базу данных карт Google.

Хитрость в создании хороших 360-градусных фото заключается в том, чтобы стоять как можно более неподвижно и пытаться перемещать телефон, а не свою руку. Хотя совершенства здесь добиться почти невозможно, попрактикуйтесь,

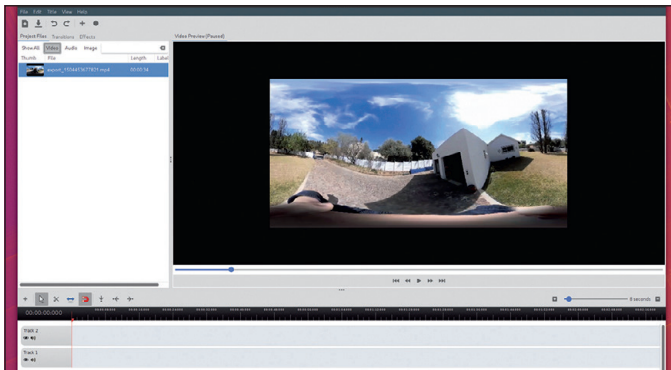
и вы получите довольно неплохие результаты для 360-градусного снимка, не потратив ни копейки на приобретение специального набора.

Помните, что фотографии тоже можно добавять на временную шкалу видео и отображать их так долго, как вы хотите.

Создание последовательности 360-градусных изображений для совместного воспроизведения можно использовать для создания весьма сильных эффектов.

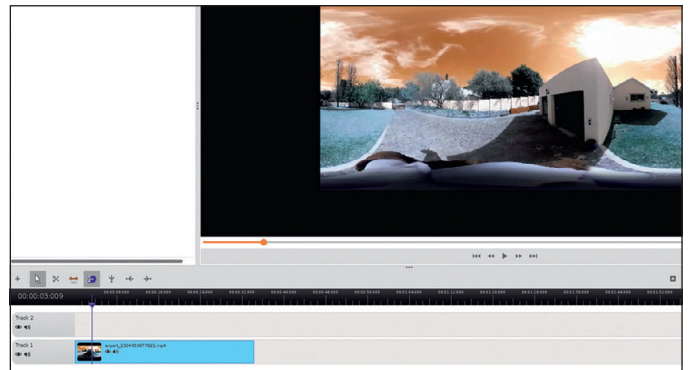


## Склеиваем ваши 360-градусные видео



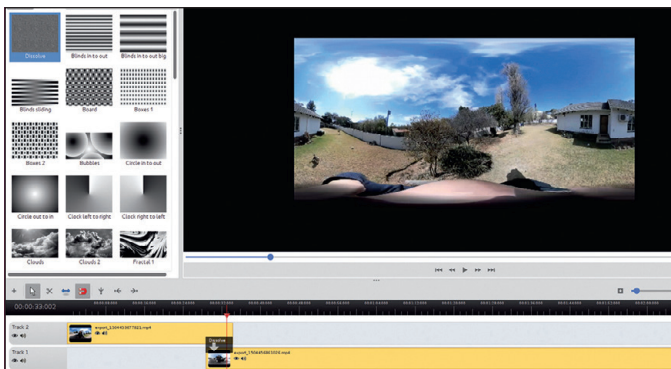
### 1 Начните с Openshot

Для целей этого пошагового руководства мы будем использовать *Openshot*, поскольку он прост. Он не может выдавать 4K-видео: для этого нам понадобится более сложный редактор *Kdenlive*. Для начала загрузите ваши 360-градусные клипы, перетащив их в поле Project Files слева.



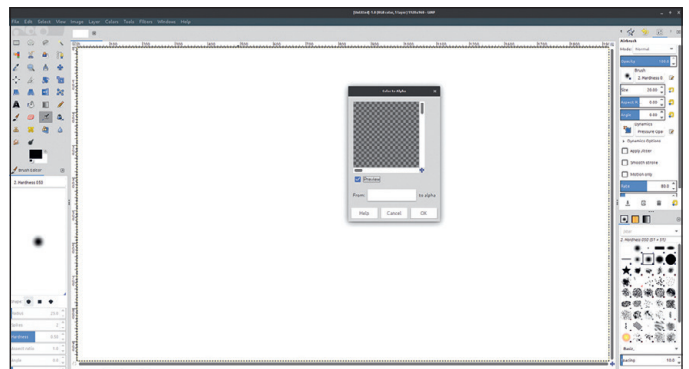
### 2 Простое редактирование

Теперь перетащите клипы на временную шкалу. Вы можете отредактировать их до предпочитаемой вами длины, просто перемещая края шкалы времени. Вы заметите, что на временной шкале имеется две видеодорожки. Следующий клип в очереди надо поместить не на ту дорожку, которая воспроизводится в данный момент.



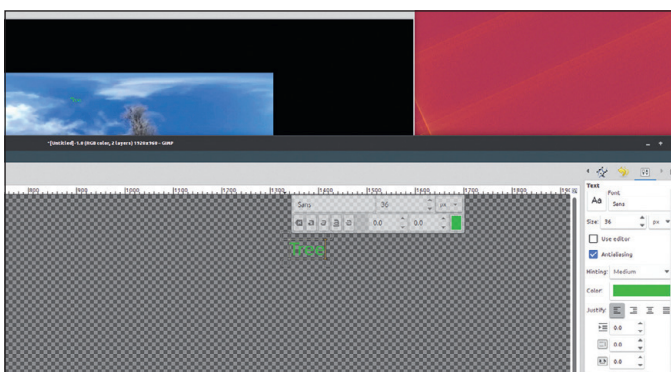
### 3 Выбор переходов

Лучше всего слегка перекрыть две видеодорожки и добавить эффект перехода, чтобы смешать их друг с другом, но использовать что-то простое, например, растворение. Смена изображений в стиле *Звездных войн* в 360 градусах будет выглядеть странно. Нажмите Transitions [Переходы] вверху слева, выберите эффект и перетащите его на область наложения двух дорожек видео. Остальное делает *Openshot*.



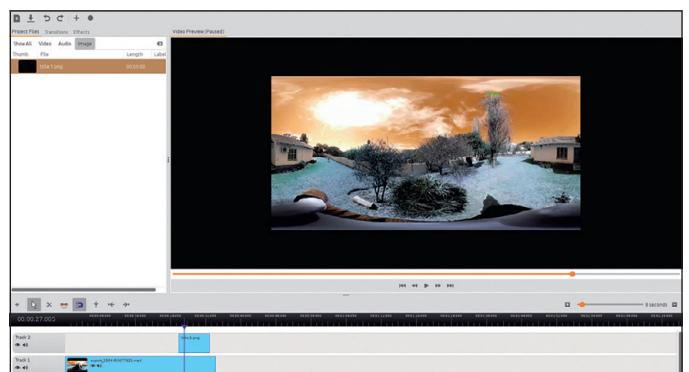
### 4 Создайте ярлыки для объектов

Вы можете добавить к своему видео плавающие ярлыки. Это можно сделать, создав PNG с прозрачным фоном в вашей любимой программе для редактирования фото, например, *GIMP*. Используйте команду Color to Alpha из меню Colors [Цвета] для создания прозрачности. Изображение должно иметь такие же размеры, как и ваше эквидистантное видео.



### 5 Подпишите

Сделайте подпись на ярлыке с помощью текстового инструмента — вам надо разместить его там, где он должен появиться в вашем 360-градусном видео, и выбрать для него правильный размер (используя захват видео, поскольку слой изображения вам поможет), чтобы видеть, где он находится относительно всего прочего на снимке. Затем экспортируйте изображение как заголовок в формате PNG, сохранив характеристики прозрачности.



### 6 Пробы и ошибки

Вернувшись в *Openshot*, импортируйте изображение и поместите его поверх видеодорожки (ваши 360-градусные видео должны быть внизу). В видео появится ярлык, всплывая на несколько секунд. Он может выглядеть растянутым при просмотре в качестве 360-градусных видео — вы можете использовать *Hugin Panorama Editor* для создания криволинейных ярлыков, более точно вписав их в перспективу (<http://bit.ly/2wR18U>).

# MEncoder: Как захватить видео

Джон Найт пытается присоединиться к крутым пацанам, копируя видео в терминале из *MEncoder* — старшего брата *MPlayer* в строгом костюме.



**Наш эксперт**

**Джон Найт**

Когда Джон не играет в видеоигры на французском языке, его обычно можно заставить за избиением большого барабана до тестостеронного состояния.

В развитие темы об *MPlayer* (см. стр. 52), мы представим вам его брата по оружию, *MEncoder*. Как и в случае с *MPlayer*, вы можете спросить: «У *MEncoder* имеется GUI, и зачем мне лезть в терминал?» И мы бы снова ответили, что командная строка (CLI) дает вам такую мощь, с которой GUI просто не сравнится.

Каждая команда *MEncoder* имеет пять компонентов: сама программа, исходное видео для преобразования, имя видеофайла на выходе, выбор аудиокодека и выбор видеокодека. Основной синтаксис команды выглядит так:

```
$ mencoder original-video -o output-video -oac audio-codec -ovc video-codec
```

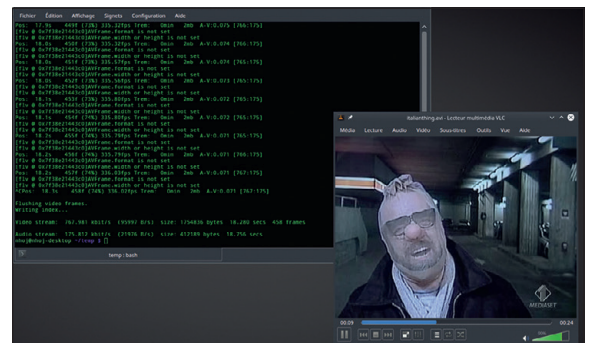
Итак, начав со случайного видео, вот пример минимально достаточного синтаксиса:

```
$ mencoder alien-bovriil.flv -o test.avi -oac x264 -ovc mp3lame
```

Вероятно, с этим стоит разобраться получше. К сожалению, параметры аудио- и видеокодеков являются обязательными — без их указания возвращается ошибка. В приведенном выше примере используется библиотека x264 для кодирования видеопотоков H.264 и библиотека LAME для кодирования аудиопотоков MP3, которые хранятся в контейнере .avi. Чтобы узнать, какие кодеки, пакеты кодеков или контейнеры доступны для использования в вашей системе для аудио и видео, введите соответственно `$ mencoder -oac help` или `$ mencoder -ovc help`.

Чтобы избавить вас от длинного нудного списка, мы включили скриншот, содержащий вывод обеих команд (внизу стр. 59). Это в системе Linux Mint, где не установлено ничего особо необычного или экзотичного. Вы увидите «копию» в разделах аудио и видео, которые захватывают то, что уже есть в источнике. Особенно полезна опция mp3lame, и lavc (сокращение от libavcodec) также отлично подходит для покрытия самых распространенных аудиоформатов.

По части видео, gaw делает несжатое видео и может быть отличным инструментом для ряда сложных видеопроектов, коль скоро вам хватает места на диске! Большинство пользователей будет



➤ Нажатие **Ctrl+C** убьет процесс `mid-rip`, но всё равно создаст видео, которое стоит проверить, прежде чем начать большую работу.

интересовать в первую очередь обширный пакет кодеков lavc, плюс xvid (совместимый с Divx) и x264 (совместимый с H.264).

Чтобы изучить логику и синтаксис *MEncoder*, возьмем базовую команду и будем ее постепенно обогащать, причем таким способом, который, как мы надеемся, прост для понимания. Синтаксис *MEncoder* запутан, из-за различных структур между кодеками, и, к сожалению, нам здесь хватит места только для описания по одному кодеку на каждую из систем.

Мы остановились на видеокодеке x264 для и mp3lame для аудио, но см. врезку внизу с дополнительной информацией о других кодеках. Следующая команда перекодирует наше тестовое видео в нечто с видео H.264 и аудио MP3:

```
$ mencoder alien-bovriil.flv -o test.avi -oac mp3lame -ovc x264
```

Однако эта простейшая команда использует только значения по умолчанию, а вы, вероятно, захотите настроить такие области, как размер файла, качество и соотношение сторон.

Начнем с битрейта аудио. Чтобы изменить настройки mp3lame, надо применить параметр `-lameopts` с необходимыми аргументами. В нашем случае мы примем постоянную скорость передачи

## Libavcodec (он же lavc)

Описание других кодеков будет длительным упражнением, и хотя мы с удовольствием охватили бы Divx/Xvid, вам придется найти документацию, если вас это специально интересует.

А вот Libavcodec мы не можем оставить без внимания, поскольку *MEncoder* в первую очередь ориентирован на lavc и формат AVI. В основной статье мы использовали x264, потому что было легко получить хорошие результаты без сложного синтаксиса, но если вы сумеете освоить структуры команд lavc, то действительно сможете кодировать

практически любой формат, который вам нравится, включая старые материалы с 1990-х и начала 2000-х гг., такие как MPEG1, RealVideo, Quicktime, WMV и такие классные вещи, как Sony Digital Video. Lavc может даже создавать видео, совместимые с Xvid или H.264, и если вы изучите синтаксис lavc, вам может не понадобиться изучать другие кодеки вообще! По умолчанию (т.е. только с lavc для -oac и -ovc), lavc будет использовать Divx-совместимый MPEG4 для видео и MPEG2 для аудио. Однако это немного странная комбинация, которая будет

работать не всегда, и для более распространенного сочетания Divx и MP3 попробуйте следующее:

```
$ mencoder alien-bovriil.flv -o test.avi -oac mp3lame -ovc lavc
```

Если вы хотите поэкспериментировать с другими форматами, вот команда, вроде бы подходящая для FLV-видео:

```
$ mencoder random-video.mkv -o test.flv -oac mp3lame -ovc lavc -lavcopts vcodec=flv -of lavf
```

Узнайте больше на [www.MPlayerhq.hu/DOCS/HTML/en/menc-feat-enc-libavcodec.html](http://www.MPlayerhq.hu/DOCS/HTML/en/menc-feat-enc-libavcodec.html).



## Работа с DVD

DVD-риппинг — сложная тема, но мы хотя бы поделимся с вами парой беглых заметок.

Чтобы выбрать заголовок DVD, используйте формат `dvd://x` и введите номер любого заголовка, который хотите скопировать. Полнометражные фильмы обычно бывают `dvd://1` или `dvd://2`.

Если в итоге у вас окажется аудиопоток с неправильным языком, попробуйте добавить `-aid 128`,

который будет использовать основной поток. Второй язык будет на 129, третий на 130, и т.д.

Нежелательные субтитры выключаются с помощью параметра `-nosub`.

Один из способов копирования из DVD — просто копировать существующие потоки и выгружать их в `.avi`. Результаты будут отличного качества, но у нас возникли проблемы с синхронизацией

почти в каждом плеере! Но если вы хотите попробовать, скомаундите

```
$ mencoder dvd://1 -o test.avi -oac copy -ovc copy
```

Вы, правда, можете привести в недоумение некоторые видеоплееры, такие как *VLC*, но преобразование звука в простой сжатый поток типа `mp3lame` может, по крайней мере, решить проблему синхронизации.

192 кбит/с. Сначала вставьте аргумент `cbr` (Constant Bit Rate), за которым следуют разделитель `:` и, наконец, аргумент самого битрейта (`br = xxx`) — всё это выглядит так:

```
$ mencoder alien-bovril.flv -o test.avi -oac mp3lame -lameopts cbr:br=192 -ovc x264
```

Возможно, вы недовольны значением качества видео по умолчанию и хотите либо поднять его для лучшего изображения, либо опустить, чтобы сэкономить на размере файла. Средний битрейт для аудио и видео отобразится по окончании `mencoder`, и вы примете это число за основу.

Чтобы задать битрейт вручную, сначала используйте параметр `-x264encopts`, с аргументом `bitrate=xxx`. В нашем случае мы преобразуем битрейт до 2000, и *MEncoder* будет кодировать примерно на этом битрейте, например, так:

```
$ mencoder alien-bovril.flv -o test.avi -oac mp3lame -ovc x264 -x264encopts bitrate=2000
```

Если картинка искривилась из-за неверного аспекта, или вы хотите исправить аспект существующего видео, можно сделать и это. Для изображения в пропорции 16×9 просто добавьте аргумент `-aspect 16:9`, как показано ниже:

```
$ mencoder alien-bovril.flv -o test.avi -oac mp3lame -ovc x264 -aspect 16:9
```

А чтобы вернуть растянутое изображение 4×3 к его первоначальным пропорциям, просто измените аспект на 4:3.

```
$ mencoder alien-bovril.flv -o test.avi -oac mp3lame -ovc x264 -aspect 4:3
```

У вас есть клип, который был снят на видео и имеет такой ужасный «линейчатый» вид? Здесь нужен деинтерлейсинг. Для этого есть несколько фильтров, но у нас наилучшие результаты получились с `yadif` (Yet Another De-Interlacing Filter). Чтобы запустить его, используйте параметр видеофильтра `-vf` с аргументом `yadif`:

```
$ mencoder alien-bovril.flv -o test.avi -oac mp3lame -ovc x264 -vf yadif
```

Если объединить последние три примера в одну команду, выйдет так:

```
$ mencoder alien-bovril.flv -o test.avi -oac mp3lame -ovc x264 -x264encopts bitrate=2000 -aspect 16:9 -vf yadif
```

## DVD-риппинг

Теперь, чтобы избежать серьезных правовых проблем для журнала, мы будем действовать в предположении, что копируя [rip] DVD, вы либо имеете законные права на этот DVD, либо живете в стране, законы которой разрешают личные резервные копии. Мы не берем на себя юридическую ответственность за ваше личное использование и не поощряем незаконный обмен файлами [Ред.: — Мы видели торренты **LXF**].

См. врезку вверху для получения дополнительной информации, но для простого копирования из первого наименования на DVD команда выглядит так:

```
$ mencoder dvd://1 -o test.avi -oac mp3lame -ovc x264
```

Вот пока и всё, что можно сделать через графический интерфейс. А зачем вам терминал? А затем, что функциональность

оболочки Linux позволяет делать удивительные вещи, а программы, запускаемые в ней, оболочка расширяет — и дополняет их функции.

Для начала, можно писать скрипты оболочки, которые заранее организуют работы по кодированию. Например, создавая резервную копию серии видео, для легкого доступа, мы смогли настроить скрипт, который сразу копирует все заголовки DVD по отдельности.

Мы дадим вам только две строки нашего скрипта, чтобы дать вам общую идею, но как только вы выработаете настройки для одного заголовка, обычно их можно применить ко всем остальным на диске. Затем понадобится менять только пару символов для каждой строки (возможно, название DVD и номера эпизодов), и вы переберете весь диск. Идеально подходит для резервного копирования комплектов DVD!

```
$ mencoder dvd://1 -o video-1-15.avi -oac mp3lame -lameopts cbr:br=192 -ovc x264 -aspect 16:9 -nosub -aid 128
```

```
$ mencoder dvd://2 -o video-1-16.avi -oac mp3lame -lameopts cbr:br=192 -ovc x264 -aspect 16:9 -nosub -aid 128
```

Ну и под конец — слияние видео. Введя несколько видео одного формата, можно просто использовать `copy` для выбора кодека и объединить все видео в один фильм, например:

```
$ mencoder video1.avi video2.avi video3.avi -o all-threevideos.avi -oac copy -ovc copy
```

Или, если у вас есть видео в разных форматах и вы не против некоторой потери в качестве (что компенсируется выбором более высокого битрейта), можете сшить их вместе в требуемом вам порядке таким образом:

```
$ mencoder video2.mov video1.avi video3.flv video4.mkv -o everything-squished-together.avi -oac mp3lame -ovc x264
```

Очевидно, что если вы просто выпускаете одно-два видео, то удобнее это сделать через GUI. Но если вы хотите изготавливать видео большими партиями или, может быть, вам просто нравится с ними возиться, тогда вам нужен терминал. Немного ноу-хау терминала, в сочетании с такой старомодной изобретательностью, делает столь мощный инструмент, как *MEncoder*, еще мощнее. **LXF**

## Скорая помощь

Храните файл со всеми командами успешных захватов *MEncoder* плюс ваши заметки по синтаксису и форматированию. Такие справки часто являются более быстрым способом работы, чем использование GUI.

```
Fichier  Édition  Affichage  Bookmarks  Configuration  Aide
nhoj@nhoj-desktop ~/temp $ mencoder -oac help
MEncoder 1.2.1 (Debian), built with gcc-5.3.1 (C) 2000-2016 MPlayer Team

Available codecs:
copy      - frame copy, without re-encoding (useful for AC3)
pcm       - uncompressed PCM audio
mp3lame   - cbr/abr/vbr MP3 using libmp3lame
lavc      - FFmpeg audio encoder (MP2, AC3, ...)
twolame   - Twolame MP2 audio encoder

nhoj@nhoj-desktop ~/temp $ mencoder -ovc help
MEncoder 1.2.1 (Debian), built with gcc-5.3.1 (C) 2000-2016 MPlayer Team

Available codecs:
copy      - frame copy, without re-encoding. Doesn't work with filters.
frameno   - special audio-only file for 3-pass encoding, see DOCS.
raw       - uncompressed video. Use fourcc option to set format explicitly.
nuv       - nuppel video
lavc      - libavcodec codecs - best quality!
libdv     - DV encoding with libdv v0.9.5
xvid      - XviD encoding
x264      - H.264 encoding

nhoj@nhoj-desktop ~/temp $
```

Приведен вывод результатов двух команд *MEncoder -oac* и *-ovc* (аудио и видео соответственно).

# Prey: Следим за устройствами

Маянк Шарма завел к своим устройствам приложение в стиле Голлума, и с яростью Назгула обрушивается на любого, кто на них посягнет.



## Наш эксперт

Как бы невероятно это ни звучало, но однажды **Маянк Шарма** умудрился потерять пульт от своего телевизора, уронив его за чужой диван. Такие дела!

Терять по небрежности ноутбук, конечно, не следует, но это не означает просто принять тот факт, что ваш компьютер пропал навсегда. Программа *Prey* [англ. добыча] помогает вернуть ваши украденные устройства, позволяя отслеживать и управлять ими, а также делать их непригодными для того, кто ими завладел. *Prey* помогает следить за злоумышленниками и тем, как они используют ваш компьютер, и собирает информацию, которая вам пригодится, когда вы сообщите о краже властям.

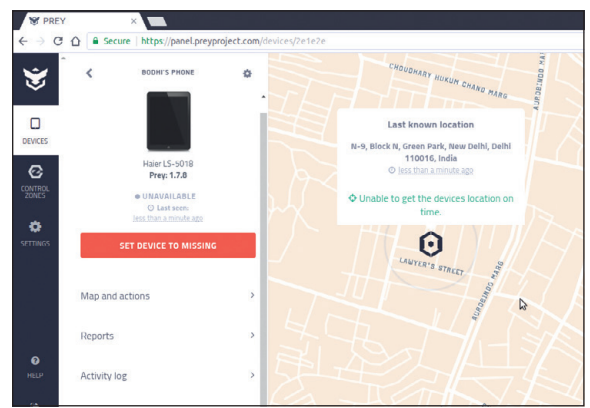
*Prey* устанавливает на ваше устройство агента, работающего в фоне и периодически отправляющего в его штаб-квартиру онлайн HTTP-запрос о том, собирать ли ему информацию, или выполнять какие-то действия, или оставаться в спящем режиме. Потеряв устройство, вы делаете об этом пометку на «приборной панели» *Prey*, и устройство начинает собирать данные, чтобы помочь вам отследить его.

Помимо Linux, *Prey* работает на нескольких операционных системах, включая Windows, Mac OS X и даже Android и iOS, так что вы можете использовать его для отслеживания ноутбуков, а также мобильных устройств. Использование бесплатное для слежения за менее чем за тремя устройствами, или выберите один из нескольких тарифных планов, начинающихся примерно с £3/месяц.

Проект *Prey* имеет прекомпилированные бинарники для Debian-дистрибутивов, таких как Debian и Ubuntu. Чтобы настроить *Prey* в этих дистрибутивах, перейдите в раздел Download [Скачать] на сайте проекта ([www.preyproject.com](http://www.preyproject.com)) и выберите 32- или 64-битный бинарник. Затем дважды нажмите на скачанный Deb-файл для установки агента *Prey*.

Есть пара других способов установить *Prey*. Некоторые дистрибутивы включают *Prey* в свои сторонние репозитории, например, Arch AUR. Более того, поскольку он написан в Node.js, вы можете установить *Prey*, используя менеджер пакетов *npm*. Вначале установите *Node.js* через менеджер пакетов вашего дистрибутива (<https://nodejs.org/en/download/package-manager>), а затем введите `npm install -g prey`, чтобы извлечь клиент *Prey* из репозитория *npm*. Помните, однако, что, в отличие от официальных двоичных пакетов, вам придется вручную обновлять эту установку, запуская `npm update -g prey`, как только появится новая версия

После установки клиента *Prey* запустит свой графический инструмент настройки. В первый раз вам понадобится создать новую учетную запись, введя свое имя, адрес электронной почты и пароль. Вот и всё. В фоне агент *Prey* добавит этот компьютер к списку отслеживаемых устройств. Он автоматически выберет имя устройство и его тип, что вы сможете изменить позднее из панели управления *Prey*. Для последующих установок на других устройствах выберите опцию связать устройства с вашей существующей учетной записью, и устройство будет отслеживаться автоматически.



➤ Панель управления с доступом через Интернет означает, что вы можете настраивать *Prey* на украденном компьютере даже после того, как его украдут!

## Установите ловушку

Настроив свое устройство, вы можете настроить его поведение через панель управления *Prey* с доступом через Интернет. Перейдите на сайт проекта, нажмите кнопку Log In [Войти] в верхнем правом углу и аутентифицируйтесь с помощью адреса электронной почты и пароля, указанных вами при первом запуске.

Панель управления разбита на разделы, управляющие разными аспектами устройства. По умолчанию панель управления переносит вас в раздел Devices, который отображает все добавленные устройства в виде списка. Нажмите на устройство, чтобы просмотреть автоматически собранную и объединенную агентом *Prey* информацию. Начальная страница для каждого устройства показывает краткую информацию об устройстве — такую как его операционная система и версия клиента *Prey*, с которой он работает. Значок шестеренки вверху поможет вам изменить имя устройства или удалить его из списка отслеживаемых устройств.

Перейдите в раздел Hardware information [Информация об оборудовании] для получения более подробной информации об устройстве, например, его серийном номере, который пригодится при предоставлении подробного отчета властям. Раздел Maps and Action [Карты и действия] отображает последнее известное местоположение устройства вместе с кнопкой для отображения его текущего местоположения посредством отправки запроса устройству.

## Храните свое личное

У *Prey* также есть очень полезная функция геозонирования: с ее помощью вы можете отметить на карте личные зоны и попросить *Prey* отправлять вам письмо на электронную почту, когда одно из устройств, присвоенное зоне, покидает или входит

## Скорая помощь



Вы также можете защитить *Prey* двухфакторной аутентификацией с помощью приложения Google Authenticator на вашем устройстве Android или iOS.



в нее. Перейдите на вкладку Control Zones [Управление зонами], чтобы определить зону относительно текущего расположения устройства. После создания зоны вам понадобится отметить устройства, которые вы хотите добавить в эту зону. По умолчанию Prey будет отправлять вам письмо на электронную почту, когда устройство перемещается в эту зону или покидает ее, но вы можете отключить любое из этих двух действий.

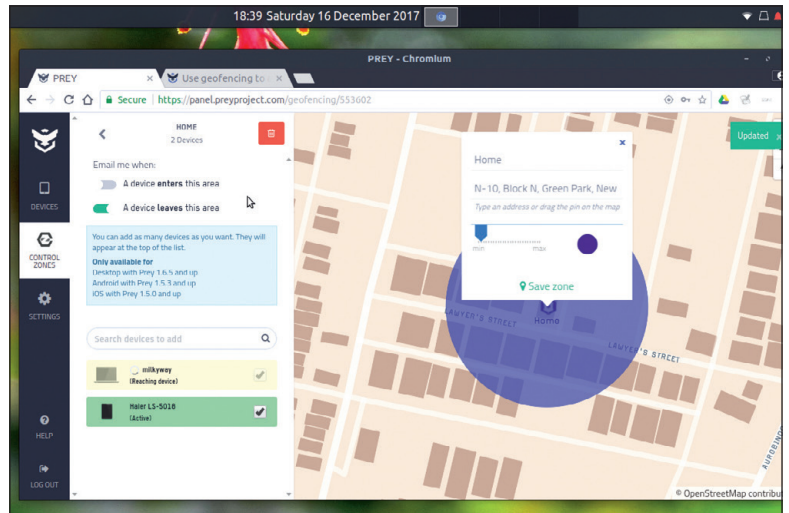
Рядом с картой находится выпадающее меню Actions, отображающее в виде списка те действия, которые вы можете удаленно совершить на устройстве из панели управления. Некоторые из этих опций разработаны, чтобы помочь вам найти ваше устройство, или, возможно, даже переубедить вора, если он в непосредственной близости. Опция Alarm [Сигнал] подает из вашего устройства громкий сигнал, чтобы помочь вам обнаружить его, если оно рядом. Она отображает четыре различных звука, которые будут производиться динамиками устройства в течение 30 секунд. Затем есть опция Alert [Предупреждение], отображающая предупреждение на экране пропавшего устройства. Вы можете использовать ее для отправки сообщения на 250 символов, которое отобразится на экране устройства.

Если переубедить вора не получится, можно применить опцию Lock [Заблокировать], чтобы пресечь использование компьютера без ввода пароля. Вы также можете включить звуковой сигнал уже после блокировки устройства, чтобы не дать злоумышленнику выключить динамики. Блокировка может быть отключена удаленно, а также является хорошим средством, чтобы досадить вору, при совместном применении со звуковыми сигналами и сообщениями.

## Следите за добычей

При потере ноутбука войдите в веб-панель Prey, перейдите на страницу устройства и нажмите на большую красную кнопку, чтобы отметить его как пропавшее. Prey может незаметно собирать информацию о пропавшем устройстве и его текущем операторе.

Как только устройство возобновит работу, Prey употребит ближайшие точки Wi-Fi для определения местоположения вашего устройства. Prey также собирает остальную относящуюся к сети информацию, например, публичный IP-адрес сети, к которой подключено устройство. Вдобавок к местоположению, Prey также делает снимок экрана устройства. Рано или поздно вы получите скриншот, когда он войдет в свою учетную запись электронной почты или другого сайта. Хотя вы не добудете его пароль, вы четко рассмотрите его уникальное имя пользователя, с помощью которого можно связаться с ним. Если ваше устройство снабжено встроенной веб-камерой, Prey будет тайно делать снимки того, кто сидит за устройством. Довольно скоро вы поймаете вора перед вашим украденным устройством.



▶ Пользователи бесплатной версии могут задать одну контрольную зону, а пользователи платной — три или больше, в зависимости от типа своей подписки.

Вы можете задать интервал, после которого Prey будет просыпаться и собирать требуемую информацию. Бесплатная версия собирает до 10 сообщений каждые 10 или 20 минут, а платная версия может отправлять сообщения каждые две минуты. Prey отправит вам сообщения на ту электронную почту, которую вы указывали для регистрации.

Теперь все готово. Как только злоумышленник выйдет в Интернет с вашего похищенного устройства, клиент Prey предупредит веб-службу Prey. Мы надеемся, что вы никогда не потеряете свой ноутбук, но если это всё же случится, вы вполне готовы поймать того, кто его украдет. Prey оснащает вас средством заставить вора вернуть ваше устройство или собрать достаточно информации, предоставив властям сильный аргумент в пользу принятия соответствующих мер.

Платные версии предлагают некоторые дополнительные полезные функции, особенно удаленное стирание, которое поможет вам удалить все виды файлов. Например, если вы выберете Documents [Документы], клиент Prey удалит директории Documents, Downloads [Загрузки], Desktop [Рабочий стол], Pictures [Изображения] и Videos [Видео] для всех пользователей. Похожим образом выбор Password [Пароль] сотрет директории .gnome2/keyrings и .ssh, а Emails [Электронная почта] удалит все данные, относящиеся к локальной почте и Thunderbird. Пользователи версии Business получают ряд дополнительных функций, которые помогут им отслеживать несколько устройств, например, настраиваемые ярлыки и расширенный поиск. LXF

## Prey на Android

Помимо компьютеров, Prey также умеет защищать мобильные устройства на операционных системах Android и iOS. Чтобы приступить к работе, скачайте приложение из Google Play Store или Apple App Store. После установки подключите его к своей учетной записи или создайте новую, таким же образом, как и на настольной версии данного приложения.

После привязки к вашей учетной записи Prey предложит вам активировать администратора

устройства путем блокировки программного обеспечения с паролем для дополнительной безопасности. Когда эти меры будут приняты, любому, кто заполучит ваше устройство, вначале придется удалить привилегии, предоставляемые администратором, прежде чем он сможет удалить программное обеспечение с вашего телефона.

Приложение Prey for Android имеет опцию Disable Power Menu [Отключение меню питания], которая, будучи активирована, предотвращает

выключение вашего устройства, отключая его меню питания. Вы также можете задать PIN безопасности и использовать его, чтобы отправлять инструкции вашему мобильному устройству через текстовые сообщения.

Более того, приборная панель Prey для устройства Android имеет дополнительную функцию. Используйте Toggle camouflage action [Включить камуфляж], что скроет иконку приложения Prey с домашнего [Home] экрана.

» Подпишитесь на печатную или электронную версии на [www.linuxformat.ru/subscribe!](http://www.linuxformat.ru/subscribe!)

# Энтропия: Это ваш случай!

Нейт Дрейк проведет вас через хаотичную тему повышения случайности, ради защиты вашей системы. Не забудьте включить гелевую лампу, пока вы там.



**Наш эксперт**

**Нейт Дрейк** — независимый технологический журналист, специализирующийся на кибербезопасности. Он чувствует, что энтропия уже не та, что раньше.

**Скорая помощь**



Linux сохраняет оценку количества битов шума в пуле энтропии. Чтобы увидеть это, откройте терминал и запустите `cat /proc/sys/kernel/random/entropy_avail`.

➤ Вид камеры, используемой в системе LavaRand от Cloudflare. Видеопоток этой стены гелевых ламп используется для генерации энтропии, потому что движение «геля» крайне непредсказуемо.

Что общего у гелевого светильника, корзины для мусора и детектора задымления? Помимо того, что они легко доступны в вашем неотдаленном холостяцком жилище, все они сыграли определенную роль в увеличении энтропии — или, другими словами, случайности. Генерация по-настоящему непредсказуемых битов данных лежит в основе важнейших операций, таких как надежное шифрование web-трафика. В нашем учебнике мы рассмотрим, как Linux управляет энтропией для создания качественных случайных данных, а также способы повышения и проверки доступной энтропии в вашей системе.

## Битва за энтропию

В настоящее время нет способа окончательно доказать, что некий поток данных является полностью случайным. Однако при определенных обстоятельствах можно доказать, что данные неслучайны. В качестве примера, при написании этой статьи автор бросил две беспристрастные шестигранные кости и получил результат 2–5 (в сумме — семь). С виду сумма этих чисел может показаться случайной, потому что заранее было неизвестно, как выпадет каждая кость. Однако нельзя сказать, что она действительно непредсказуема, так как сумма двух костей более вероятно будет семь (17%), чем любое другое значение.

По этой причине в 1890 г. викторианский статистик Фрэнсис Гальтон [Francis Galton] смог создать систему для генерации случайных чисел, не глядя вытаскивая три самодельные шестигранные кости из корзины для бумаг (см. полное изложение его методов на <http://galton.org/essays/1890-1899/galton-1890-dice.pdf>).

В XXI веке DNS-компания CloudFlare, безусловно, заслуживает признания за творческие способы, которые она пытается использовать в генераторах случайных чисел в своих системах, один из которых включает применение камер для отслеживания каплей воска, движущихся внутри гелевых ламп. Хотя тепловые потоки,

перемещающие гель, не вполне непредсказуемы, камера, записывающая лампы, также регистрирует окружающий шум и свет, позволяя шифровать ключи примерно 10% трафика, используемого в сети Cloudflare.

Если генератор случайных чисел не работает должным образом, последствия могут быть катастрофическими. В августе 2017 г. программисты Дэн Шумов [Dan Shumow] и Нильс Фергюсон [Niels Ferguson] изложили свои выводы о том, что Dual EC PRNG, разработанный NSA в начале 2000-х гг. и одобренный NIST, генерировал случайные числа потенциально небезопасным способом.

Человеческая ошибка также играет свою роль в ослаблении случайности. В 2008 г. проект Debian объявил, что разработчик из лучших побуждений удалил две строки кода из распространяемого ими пакета OpenSSL. Побочным эффектом стало повреждение PRNG (генератор псевдослучайных чисел), используемого OpenSSL, что уменьшило потенциально неограниченное количество ключей SSL/SSH, которые он может генерировать, до всего лишь 32767.

## Устройство генератора случайных чисел

Криптографические ключи создаются с использованием CSPRNG (криптографически безопасного генератора псевдослучайных чисел), имеющего хороший источник случайных данных, чрезвычайно трудоемкий для перебора. Linux поддерживает «пул» энтропии, который собирается из нескольких источников (см. ниже). Данные проходят процесс исключения систематической ошибки для удаления предсказуемых битов, а затем подаются в CSPRNG. CSPRNG, в свою очередь, питает энтропийный пул, который может быть использован через файл устройства `/dev/random`. Например, чтобы сгенерировать 1-МБ файл со случайными данными, полученными исключительно из пула энтропии, откройте терминал и выполните следующее:

```
dd if=/dev/random of=test bs=1M count=1 iflag=fullblock
```

Если пул не содержит достаточное количество случайных битов для создания выбранного файла, он будет «блокироваться» до тех пор, пока не накопится больше энтропии. Возможно, вы сталкивались с этим раньше, когда генерировали ключи GPG/SSH на своей машине. Обычно эта утилита побуждает вас продолжать использовать компьютер, чтобы он мог накапливать случайность из запущенных процессов, нажатий клавиш и т. д.

В случае настольных компьютеров это нормально, но если вы имеете дело с сервером, то «блокировка» способна вызывать проблемы, поскольку серверы обычно работают без клавиатуры, мыши и монитора.

Для обхода этой проблемы приложения могут взамен черпать из `/dev/urandom`, что обеспечивает неблокирующий источник псевдослучайности. Ранее `t` поддерживал отдельный пул энтропии; после Linux 4.8, он извлекает случайные значения непосредственно





## Случайность на Raspberry Pi

Боги Случая улыбнулись всем владельцам Raspberry Pi, потому что каждое устройство содержит генератор случайных чисел (*hwrng*), пригодный для обеспечения качественной случайности в `/dev/random`. Для начала вам надо будет установить набор *rng-tools*, содержащий демон *rngd*, который будет взаимодействовать с *hwrng*.

Откройте терминал на своем Raspberry Pi и выполните `sudo apt-get install rng-tools`. Затем выполните `sudo nano etc/default/rng-tools` и удалите `#` в начале строки с текстом `HRNGDEVICE=/dev/hwrng`. Сохраните и выйдите, затем выполните `sudo service rng-tools restart`.

Если вы считаете маловероятным, что Raspberry Pi Foundation разместил правительственные лазейки в своих устройствах, то можете получить доступ к *hwrng* напрямую через `/dev/hwrng`. По умолчанию только пользователю `root` разрешено использовать устройство. Например, чтобы запустить тесты FIPS (см. внизу справа) на *hwrng*, чтобы определить его случайность, выполните `sudo cat /dev/hwrng | rngtest -c 1000`. Показатель сбоев должен составлять менее одного процента.

Чтобы сгенерировать 1-МБ файл фиктивных данных с использованием аппаратного генератора случайных чисел, откройте терминал и выполните `sudo head -c 1M /dev/hwrng > /tmp/out`. Теперь вы можете запустить `ent /tmp/out` для проведения дальнейших тестов качества случайности, создаваемой Pi.

```
pi@raspberrypi:~
File Edit Tabs Help
rngtest: starting FIPS tests...
rngtest: bits received from input: 20000032
rngtest: FIPS 140-2 successes: 1000
rngtest: FIPS 140-2 failures: 0
rngtest: FIPS 140-2(2001-10-10) Monobit: 0
rngtest: FIPS 140-2(2001-10-10) Poker: 0
rngtest: FIPS 140-2(2001-10-10) Runs: 0
rngtest: FIPS 140-2(2001-10-10) Long run: 0
rngtest: FIPS 140-2(2001-10-10) Continuous run: 0
rngtest: input channel speed: (min=14.352; avg=809.777; max=2170138.889)Kibits/s
rngtest: FIPS tests speed: (min=880.143; avg=6164.235; max=9221.553)Kibits/s
rngtest: Program run time: 28832572 microseconds
pi@raspberrypi:~ $ ent /tmp/out
Entropy = 7.999811 bits per byte.

Optimum compression would reduce the size
of this 1048576 byte file by 0 percent.

Chi square distribution for 1048576 samples is 274.37, and randomly
would exceed this value 19.31 percent of the times.

Arithmetic mean value of data bytes is 127.5603 (127.5 = random).
Monte Carlo value for Pi is 3.149334524 (error 0.25 percent).
Serial correlation coefficient is 0.000557 (totally uncorrelated = 0.0).
```

» Во время наших тестов на Raspberry Pi Zero W *hwrng* производил качественную случайность. Тем не менее, рекомендуем вам провести исследование самим.

из CSPRNG, который в настоящее время использует вариант шифрования потока ChaCha20 для непрерывной генерации псевдослучайных битов на основе случайного начального значения.

### Святое Семя

Онлайн-дебаты об относительных достоинствах `/dev/random` против `/dev/urandom` столь же неистовы, сколь и интересны. Общий консенсус в том, что `/dev/urandom`, вероятно, безопасен для большинства криптографических целей — при условии, что исходные биты «затравки [seed, англ. семя]», используемые для него, достаточно случайны.

В случае `/dev/urandom` каждый экземпляр ChaCha20 обновляется каждые пять минут, хотя случайность данных, полученных от него, будет отличаться от виртуальной и/или системы без клавиатуры. Случайность часто набирается из запросов аппаратных устройств, таких как жесткий диск, отправляющий запросы ОС (известные как прерывания), хотя с появлением SSD их также стало меньше и они стали реже.

Один из возможных способов обойти эту проблему заключается в установке *Haveged*, который доступен в репозиториях Debian и Ubuntu. *Haveged* специально разработан для ситуаций с низкой энтропией, как описано выше. Применение *haveged* «осеменяет» `/dev/random`, используя разность на счетчике метки времени процессора (TSC) после повторных выполнений цикла. Поскольку команды выполняются разное время и в разное время, это может сгенерировать достаточное количество энтропии.

Чтобы начать работу, просто установите *Haveged*, открыв терминал и выполнив `sudo apt-get install haveged`. Также обязательно добавьте *Haveged* в процессы запуска вашей системы, запустив `update-rc.d haveged defaults`.

Если ваш ПК относительно новый, то *Haveged* может оказаться лишним, поскольку все процессоры Intel Ivy Bridge и AMD64 с 2015 г. имеют встроенный генератор случайных чисел. В сентябре 2013 г. Теодор Цо [Theodore Ts'o], создатель `/dev/random`, отметил, что опора исключительно на такие процессоры для генерирования энтропии оставляет системы открытыми для правительственных лазеек [backdoor]. Но начиная с ядра Linux 3.16, система смешивает

энтропию, полученную с аппаратных устройств в `/dev/random`, так что случайность не приходит только из одного источника.

Суперпараноидальные читатели могут также найти инструкции онлайн для создания собственного аппаратного генератора случайных чисел, который измеряет квантовые события, такие как счетчики Гейгера или детекторы дыма.

### Проверка случайности

Как вы узнали ранее, доказать, что произвольная последовательность битов данных вполне случайна, невозможно. Однако существуют утилиты, способные определять, что данные не являются действительно случайными. Самый простой способ интерпретации — *rngtest*, входящий в пакет *rng-tools* (см. врезку вверху, для установки). После установки откройте терминал и выполните `cat /dev/random | rngtest -c 1000` с использованием тестов FIPS (Federal Information Processing Standard) 140-2. Если число сбоев относительно успехов превышает один процент, то данные, вероятно, не случайны. Рассмотрите возможность установки *ent*, которая может запускать дополнительные тесты ваших данных, например, распределение хи-квадрат и значение Монте-Карло для  $\pi$ . И прежде всего, помните, что качественная случайность обусловлена множественностью ее источников. LXF

```
pi@raspberrypi:~
File Edit Tabs Help
rngtest: starting FIPS tests...
rngtest: bits received from input: 20000032
rngtest: FIPS 140-2 successes: 1000
rngtest: FIPS 140-2 failures: 0
rngtest: FIPS 140-2(2001-10-10) Monobit: 0
rngtest: FIPS 140-2(2001-10-10) Poker: 0
rngtest: FIPS 140-2(2001-10-10) Runs: 0
rngtest: FIPS 140-2(2001-10-10) Long run: 0
rngtest: FIPS 140-2(2001-10-10) Continuous run: 0
rngtest: input channel speed: (min=14.352; avg=809.777; max=2170138.889)Kibit
rngtest: FIPS tests speed: (min=880.143; avg=6164.235; max=9221.553)Kibits/s
rngtest: Program run time: 28832572 microseconds
pi@raspberrypi:~ $ ent /tmp/out
Entropy = 7.999811 bits per byte.

Optimum compression would reduce the size
of this 1048576 byte file by 0 percent.

Chi square distribution for 1048576 samples is 274.37, and randomly
would exceed this value 19.31 percent of the times.

Arithmetic mean value of data bytes is 127.5603 (127.5 = random).
Monte Carlo value for Pi is 3.149334524 (error 0.25 percent).
Serial correlation coefficient is 0.000557 (totally uncorrelated = 0.0).
```

### Скорая помощь

Если вы не знаете, какой у вас процессор, загрузите и запустите `spuid` из репозитория вашего дистрибутива. См. [www.fourmilab.ch/random](http://www.fourmilab.ch/random) для более глубокого объяснения различных способов, которыми *ent* может проверить случайность.

» Для проверки неслучайности данных пригодятся инструменты вроде *ent* и *rngtest*. Здесь они выполняются на файле, созданном со встроенным генератором случайных чисел Raspberry Pi.

» Подпишитесь на печатную или электронную версии на [www.linuxformat.ru/subscribe!](http://www.linuxformat.ru/subscribe!)

# VirtualBox: VM В ВАШЕЙ СЕТИ

Маянк Шарма считает, что ПО виртуализации неплохо работает с сетью.



## Наш эксперт

**Маянк Шарма** является техническим писателем и бывшим редактором [Linux.com](http://Linux.com), но теперь проводит свое время, целыми днями играя в игры Linux в одних трусах.

## Скорая помощь

VirtualBox позволяет создавать до восьми виртуальных сетевых адаптеров, но только с помощью команды `vboxmanage modifyvm`.

## Скорая помощь

Интерфейс командной строки `vboxmanage` предлагает гораздо больше функциональности, чем то, что отображается в графическом интерфейсе, включая возможность ограничения полосы пропускания для сетевого ввода-вывода.

**V**irtualBox — чудесное приложение: оно демократизировало виртуализацию, важнейшую корпоративную технологию, и делает ее доступной обычному пользователю настольного ПК. Используя графический интерфейс приложения, можно сразу приступить к созданию виртуальных машин.

При всей простоте использования, *VirtualBox* обладает и сложными функциями. Следует отметить сетевые возможности, позволяющие программе эмулировать различные сетевые настройки. Большинство сетевых функций *VirtualBox* находятся в диалоговом окне Сеть. Чтобы их вызвать, щелкните правой кнопкой мыши по любой виртуальной машине (VM) и выберите Настройки > Сеть. Как вы заметите, к VM можно подключить четыре виртуальных контроллера сетевого интерфейса (NIC).

Есть два важных параметра, определяющих поведение этих виртуальных сетевых адаптеров. Во-первых, надо выбрать, какой тип адаптера требуется эмулировать: вам предлагаются такие опции, как Intel PRO/1000 MT Desktop (82540EM), PCnet-FAST III (Am79C973), сеть паравиртуализации (*virtio-net*) и другое. Во-вторых, и что более важно, вы должны принять решение, как они должны работать с физической сетью вашего хоста (см. врезку внизу стр. 65). Выбор типа виртуального сетевого адаптера сводится к тому, имеются ли для него гостевые драйверы. *VirtualBox* автоматически предлагает правильный тип адаптера на базе гостевой ОС, к которой он подключен; вам на самом деле и незачем изменять этот параметр.

Однако выбор сетевого режима зависит от различных факторов. Хотите ли вы, чтобы VM была частью вашей основной сети? Будет ли на VM работать сервер? Хотите ли вы, чтобы другие компьютеры в вашей сети могли подключаться к VM? По умолчанию виртуальные сетевые адаптеры функционируют как адаптеры NAT

и могут получать доступ к Интернету через хост. Однако большинство пользователей предпочитают переключиться на тип сетевого моста, что превращает VM в независимого члена основной сети.

Недостатком мостового адаптера является то, что он открывает VM в реальную сеть. Кроме того, если вы работаете с несколькими VM, у вас могут закончиться IP-адреса или сетевому администратору не хватит статически назначенных IP-адресов для серверов, работающих внутри VM. Вот почему большинство сетевых администраторов не разрешат вам подключать VM к сети больше чем через один мостовой адаптер. Итак, что если вы захотите запустить сервер внутри вашей виртуальной машины, не создавая несколько мостов в реальную сеть?

## Безопасность через изоляцию

Одно из самых популярных применений *VirtualBox* — это пробная среда для запуска всего: от отдельных приложений до даже полных операционных систем, прежде чем развертывать их на реальной машине. Точно так же, благодаря сетевым возможностям приложения, его можно использовать для тестирования сетевого программного обеспечения, создавая виртуальную лабораторию, изолированную от реальной сети.

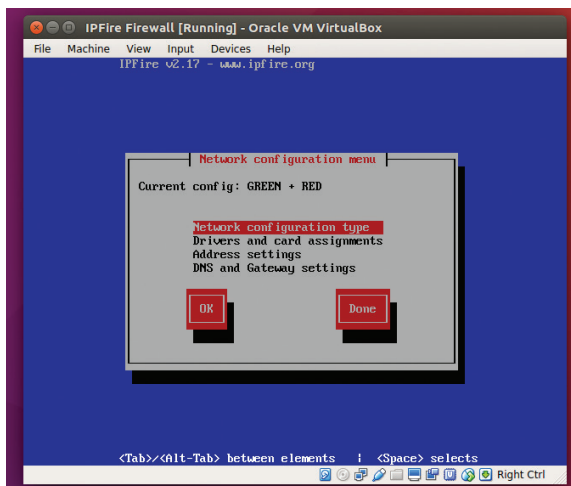
Чтобы реализовать это на практике, давайте создадим внутри VM брандмауэр IPFire ([www.ipfire.org](http://www.ipfire.org)), который будет иметь два виртуальных сетевых адаптера. Один будет подключен к Интернету, а второй — к другим VM на этом компьютере. Брандмауэр выдаст IP-адреса этим другим VM, которые будут подключены к Интернету через брандмауэр виртуальной машины.

Сначала настройте стандартную VM с любым дистрибутивом Linux. Для нашего урока мы создаем VM для сервера IPFire с 512 МБ ОЗУ и одним процессором. Затем откройте окно настроек этой VM и перейдите на вкладку Сеть, чтобы подключить два виртуальных сетевых адаптера. На вкладке Адаптер 1 убедитесь, что он подключен к сетевому мосту. Этот сетевой адаптер подключит сервер брандмауэра и VM, которые он подключает к Интернету. Затем перейдите на вкладку Адаптер 2 и включите его. Используйте выпадающее меню Тип подключения: и выберите опцию Внутренняя сеть. Другие виртуальные машины в нашей сети будут подключаться к серверу брандмауэра через этот сетевой адаптер.

Вот и всё. Остальное находится в конфигурации сервера IPFire. Как только мы настроим сервер, любая виртуальная машина на этом компьютере, использующая ту же внутреннюю сеть, что и на сервере, сможет беспрепятственно обмениваться данными через сервер брандмауэра.

## Настройка виртуальной сети

Сервер брандмауэра будет действовать как шлюз для виртуальной сети. Направьте виртуальный оптический привод сервера на установочный ISO-образ IPFire и загрузите его. Выполните процесс установки брандмауэра с параметрами по умолчанию, который установит IPFire как единственный дистрибутив внутри VM.



► Вы можете легко протестировать сетевое программное обеспечение, такое как брандмауэры и шлюзы, развертывая их в виртуальной сети.



При перезагрузке VM после установки вам будет предложено задать набор паролей для пользователя root и администратора.

Теперь наступает решающая часть, когда вам требуется настроить роли для сетевых адаптеров, подключенных к этому серверу брандмауэра. IPFire поддерживает несколько различных режимов. Режим по умолчанию, известный как Green + Red, предназначен для машин с двумя сетевыми адаптерами, такими как наша VM. После выбора этого режима в Network Configuration Type выберите параметр Drivers and cards assignment, чтобы назначить сетевые адаптеры одному из режимов. Первым адаптером в списке является Bridged NIC, который вам следует пометить как интерфейс Red, и адаптер внутренней сети — интерфейс Green. Вы можете определить сетевые адаптеры, сравнив их MAC-адреса с перечисленными в окне настроек сети в VirtualBox.

Затем прокрутите вниз до параметра Address Settings и настройте интерфейс Green. Назначьте ему 10.0.0.1 в качестве IP-адреса с маской сети 255.255.255.0. Для интерфейса Red выберите вариант DHCP. Далее перейдите к настройке параметров DNS/Gateway и введите 8.8.8.8 в качестве основного и 8.8.4.4 в качестве вторичного DNS.

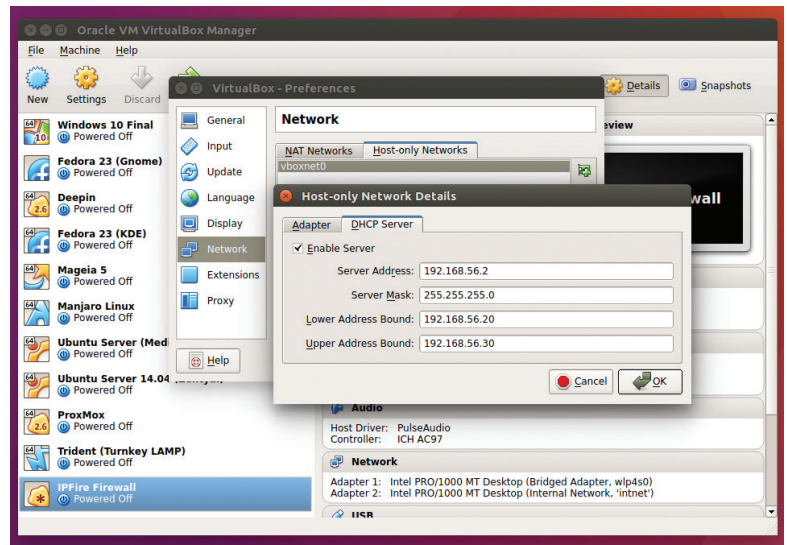
## Раздайте адреса

Когда вы закончите с сетевыми настройками, мастер настройки IPFire откроет параметры для конфигурации сервера DHCP, который будет выдавать адреса всем VM, которые будут подключены к VM брандмауэра. Активируйте сервер DHCP и введите 10.0.0.10 в поле Start Address и 10.0.0.30 в поле End Address. Таким образом вы велите серверу брандмауэра выдавать любым подключаемым VM в нашей виртуальной сети адреса в диапазоне между этими двумя значениями.

Вот и всё. Сохраните настройки и включите загрузку IPFire до запроса на вход. Теперь измените сетевые настройки любой другой VM и переключите ее виртуальный сетевой адаптер на режим внутренней сети. Когда вы загружаете данную VM, это позволит получить доступ к Интернету как раньше. Однако теперь

данные проходят через сервер брандмауэра IPFire. Чтобы проверить это, введите команду ipconfig в терминале этой VM, она будет иметь адрес в указанном диапазоне (от 10.0.0.10 до 10.0.0.30). Кроме того, перейдите на <https://10.0.0.1:444> из любой VM во внутренней сети, и вы попадете в web-панель администрирования IPFire. Используйте admin в качестве пользователя и тот пароль, который вы назначили ему ранее при настройке IPFire.

Теперь у вас есть виртуальная сеть в VirtualBox, которая выдает адреса другим VM. Все они могут получить доступ друг к другу, а также к Интернету через VM брандмауэра. Изучите интерфейс IPFire для настройки брандмауэра и проверьте его поведение на VM в виртуальной локальной сети. **LXF**



➤ Перейдите в File > Preferences > Network, чтобы создавать интерфейсы для сетей NAT и Host-only. Вы также можете включить встроенный в VirtualBox сервер DHCP для выдачи IP-адресов VM в этих сетях.

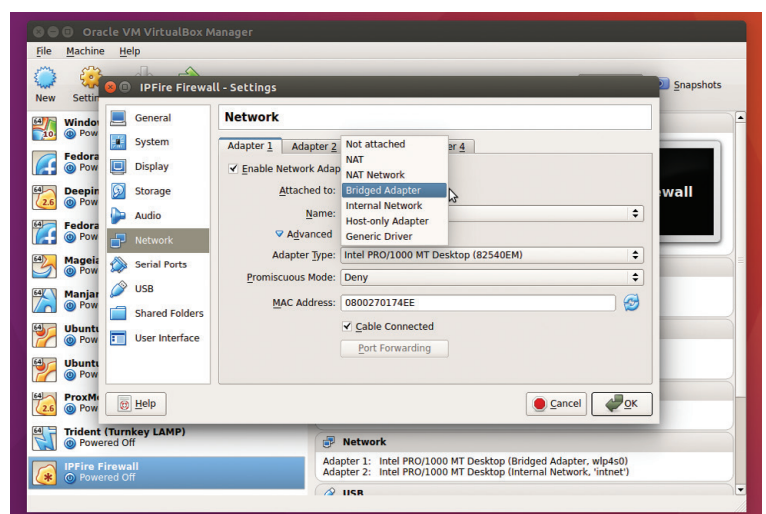
## Различные типы виртуальных сетей

VirtualBox поддерживает несколько типов виртуальных сетей. NAT — это самый простой вариант с точки зрения гостевой системы для доступа к внешним сетям. Ваш хост будет выступать в роли маршрутизатора, а ваши хосты будут находиться в частной подсети. Используйте это, если вы не используете серверы в гостевых системах. Однако VM не могут обращаться друг к другу, и вы также не можете получить к ним доступ с хоста.

Если вы хотите, чтобы гости имели доступ друг к другу, используйте режим NAT Network. Он группирует VM, которые используют его в сети, к которой невозможно получить доступ извне сети. Однако любые VM, использующие эту сеть NAT, смогут взаимодействовать друг с другом, а также получать доступ к Интернету.

Самый простой вариант доступа к VM с хоста и других компьютеров в сети — это режим сетевого моста. Ваши гости получают IP-адрес в той же подсети, что и ваш хост. Этот тип сетевого режима полезен, если вы используете серверы в гостевой системе и хотите подключиться к ним с других компьютеров в локальной сети.

Есть еще внутренняя сеть, которую мы использовали на уроке. В этом режиме только гости, подключенные к одной внутренней сети, смогут взаимодействовать друг с другом в конфигурации внутренней сети. Связь с хост-системой или другой сетью за пределами VirtualBox невозможна. Если вы хотите включить хост во внутреннюю сеть, используйте режим Host Network, который позволяет гостям также взаимодействовать с хостом.



➤ Параметры Host-only, и NAT Network зависят от настраиваемых сетевых интерфейсов, которые необходимо создавать отдельно.

» Подпишитесь на печатную или электронную версии на [www.linuxformat.ru/subscribe!](http://www.linuxformat.ru/subscribe!)

# LEDE: Свободу ВАШИМ роутерам!

Устав от проприетарных прошивок, **Джон Лэйн** разбирается с LEDE и обнаруживает, что его роутер в общем-то неплох...



## Наш эксперт

Джон Лэйн создает сети для удовольствия и выгоды. Без LEDE удовольствия было бы куда меньше...



Dropbear SSH будет использовать агент OpenSSH, так что вы можете выполнить `git push`, не нуждаясь в частном ключе роутера. Возможно, при входе вам придется использовать `ssh -A`.

**L**inux Embedded Development Environment, или LEDE — это отвлечение давнего проекта OpenWRT. Это дистрибутив Linux для встроенных устройств в определенных роутерах и полная замена предлагаемых производителем прошивок для широкого диапазона беспроводных роутеров. Если ваше устройство поддерживается, то установка на него LEDE откроет вам все его возможности и исправит уязвимости в тех случаях, когда производители этого не сделали.

На нашем уроке мы разберемся с новым роутером LEDE с простейшим интернет-соединением и выполним его настройку — в качестве начальных шагов к задействованию всех возможностей этой прошивки с открытым кодом. Таким образом нам удастся заглянуть внутрь этого специализированного Linux и изучить его конфигурацию.

## Поддержите меня!

Посетите сайт LEDE на <https://lede-project.org/toh>, чтобы выяснить, поддерживается ли ваше устройство, и если это так, следуйте инструкциям по установке LEDE на него. Процесс сильно зависит от устройства, как и уровень поддержки, зависящий от спецификации устройства — например, от чипсета и объема памяти. Возможно, вам даже понадобится открыть корпус, чтобы получить доступ к последовательному порту консоли, а то и употребить паяльник, чтобы подключиться к нему.

Помните, что установка на роутер своей прошивки, скорее всего, сделает недействительной любую гарантию, а особенно если применялся паяльник!

Мы будем действовать в предположении, что у вас уже есть совместимое с LEDE устройство и вы уже последовали инструкциям по установке для него. Мы использовали BT Home Hub 5 (см. врезку *Представляем наш роутер* вверху стр. 67) и затем установили LEDE

17.01.04, который не страдает от недавно обнаруженной уязвимости Krack WiFi.

## Первый контакт

Перед началом может потребоваться небольшая настройка сети. Вы можете преднастроит свой роутер с LEDE, не затрагивая любой другой роутер, который у вас может быть; можно оставить рабочим свое интернет-соединение и переключиться по мере готовности.

Настройка LEDE по умолчанию задает IP-адрес роутера как `192.168.1.1` и запускает сервер DHCP, раздающий адреса в подсети `192.168.1.0/24`. Эти же настройки по умолчанию применяют многие роутеры, и если они конфликтуют с вашей сетью, возможно, вы захотите вначале отключить свой компьютер от нее. Если ваша сеть другая (скажем, вы используете диапазон частных адресов `10/8` или `172.16/12`), вы можете вручную присвоить временный IP-адрес по `ip`, примерно так:

```
$ ip add address 192.168.1.10 dev eth0
```

Подключитесь к роутеру с LEDE через Ethernet-кабель. Если ваш существующий роутер использует тот же самый адрес, вам понадобится временно подключить роутер с LEDE напрямую к вашему компьютеру, чтобы изменить его адрес, как мы опишем ниже. В ином случае, подключите его к свободному порту вашего существующего роутера. Включите его, дайте ему некоторое время на загрузку и затем проверьте его доступность: направьте web-браузер на <http://192.168.1.1> для доступа к консоли LuCI (Lua Configuration Interface).

LuCI изначально не имеет пароля (мы исправим это позднее). Сейчас перейдите в `Network > Interfaces` и нажмите кнопку `Edit` для интерфейса LAN. Экран представляет опции, которые вам следует изменить в соответствии с вашей собственной сетью: выберите адрес и маску сети, совместимые с вашей локальной сетью. Введите адрес существующего роутера в качестве шлюза и DNS-сервера, затем нажмите кнопку `Save and Apply` [Сохранить и применить] в нижней части экрана. Если вы напрямую подключились к роутеру с LEDE, теперь вам нужно переподключиться к нему через ваш существующий роутер.

## Dropbear здесь

Изначальная настройка сети должна позволять вам входить в консоль LEDE. LEDE использует легковесную SSH-реализацию Dropbear, предназначенную для встраиваемых систем, в то же время широко совместимую с OpenSSH, используемый дистрибутивами рабочего стола Linux. Используйте `ssh` для подключения к учетной записи `root`. Изначально пароля нет, так что настройте его сейчас:

```
computer$ ssh root@LEDE
```

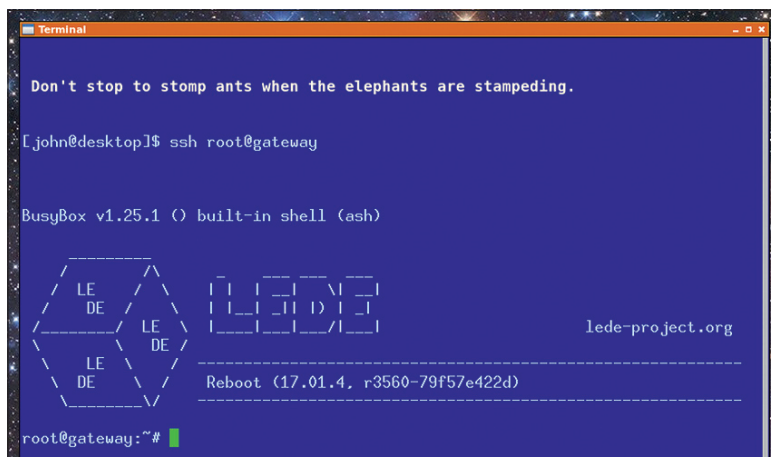
```
root@LEDE's password:
```

```
root@LEDE:~# passwd
```

```
Changing password for root
```

```
New password:
```

```
root@LEDE:~# exit
```



➤ Вот как выглядит командная строка через SSH — большая сила влечет большую ответственность!



## Представляем наш роутер...

Роутер, используемый в этом проекте — BT Home Hub 5 Type A, стандартный для BT (интернет-провайдер в Великобритании) Broadband. Есть также Type B, но его чипсет не поддерживается. Произведенный Sagetcom, этот роутер также выпускается для некоторых клиентов PlusNet как Hub One — единственным отличием будет корпус PlusNet белого цвета, тогда как BT использует черный.

Спецификация весьма неплоха, с двухканальным Wi-Fi, поддерживающая стандарты b/g/n/a/ac, с коммутатором на пять портов, модемом ADSL2+/VDSL и USB-портом для устройства внешнего хранения.

Если у вас нет поддерживаемого роутера, этот доступен примерно за £10 на eBay. Помните, что вам надо будет открыть корпус и припаять провода к двум очень маленьким контактам для доступа к серийному порту. Помощь в вашей работе готов оказать форум сообщества на <http://bit.ly/homehub5>, куда определенно стоит зайти.



► На нашем уроке используется роутер от BT.

Пароль, который мы только что задали, может использоваться для доступа к LuCI, но неплохо было бы использовать аутентификацию для SSH на базе ключей. Сначала загрузите свой открытый ключ:

```
computer$ cat ~/.ssh/id_rsa.pub | ssh root@LEDE '(umask 77 && cat >> /etc/dropbear/authorized_keys)'
```

```
root@LEDE's password:
```

```
root@LEDE:~# passwd
```

```
Changing password for root
```

```
New password:
```

```
root@LEDE:~# exit
```

Ваш ключ по умолчанию загрузится и будет помещен в файл Dropbear — `authorized_keys`, с правильными разрешениями. Теперь войдите снова; на сей раз пароль не потребуется.

Вы можете далее обезопасить доступ, отключив вход в систему на основе SSH пароля. Для этого надо изменить конфигурацию LEDE Dropbear.

LEDE использует для своих сервисов унифицированный формат настройки под названием UCI — Unified Configuration Interface. Он использует текстовые файлы, хранящиеся в одном месте в файловой системе, в `/etc/config`. Поскольку это текст, вы можете использовать *Git* для отслеживания своих изменений в настройке, и если вам это нужно, см. объяснения во врезке на стр. 68).

Настройка Dropbear находится в файле UCI `/etc/config/dropbear`. Отредактируйте опции внутри нее, чтобы они выглядели так (имеется *vi*, но при желании можно установить *nano*):

```
config dropbear
option PasswordAuth off
option RootPasswordAuth off
```

Это иллюстрирует формат UCI. Это простой текст, разбитый на разделы, содержимое которых следует за отступом (пробелом или табуляцией). Параметры опции могут содержать пустую область, если они заключены в кавычки. Альтернатива редактированию вручную — использование инструмента *uci*. Для достижения вышеописанного результата вы должны ввести такие команды:

```
$ uci set dropbear.@dropbear[0].PasswordAuth=off
$ uci set dropbear.@dropbear[0].RootPasswordAuth=off
$ uci commit
```

Читайте [https://lede-project.org/docs/user-guide/introduction\\_to\\_lede\\_configuration](https://lede-project.org/docs/user-guide/introduction_to_lede_configuration), чтобы побольше выяснить о настройке UCI и команде *uci*.

После изменения конфигурации сервис надо перезагрузить или перезапустить. Чтобы перезапустить Dropbear:

```
root@LEDE:~# /etc/init.d/dropbear restart
```

На web-сервере LuCI вы также должны включить SSL, поскольку для входа на него используется пароль `root`, который в противном случае будет гулять по сети ничем не прикрытым. LEDE предлагает пакет, который делает это. Вы используете команду `opkg` для установки пакетов из пакетного репозитория LEDE, но перед использованием его временный индекс пакетов необходимо обновить. Сделайте это, а затем установите *luci-ssl*:

```
root@LEDE:~# opkg update
root@LEDE:~# opkg install luci-ssl
```

Помните, что он использует самоподписанные сертификаты, о чем предупредит браузер. Если хотите, можете отредактировать параметры самоподписанных сертификатов в `/etc/config/uhttpd`:

```
config cert 'defaults'
option days '730'
option bits '2048'
option country 'ZZ'
option state 'Somewhere'
option location 'Unknown'
option commonname 'LEDE'
```

Самоподписанные сертификаты будут создаваться при перезапуске web-сервера, после чего можно будет получить доступ к LuCI через HTTPS, потому что HTTP больше не будет работать.

```
root@LEDE:~# /etc/init.d/uhttpd restart
```

## В эфире

Чтобы заменить существующий роутер, LEDE должен стать главным интернет-шлюзом и точкой доступа Wi-Fi для вашей LAN.

Wi-Fi изначально отключен. Параметры в `/etc/config/wireless` определяют аппаратное устройство (`wifi-device`) для каждого радио (у двухканального роутера их будет два: одно на 2,4 ГГц, а другое на 5 ГГц) и логического интерфейса `wifi-iface` для каждого. Вы можете отредактировать эти параметры, чтобы выбрать канал и настроить SSID идентификатор и параметры шифрования:

```
config wifi-device 'radio0'
option type 'mac80211'
option channel '36'
option hwmode '11a'
option htmode 'VHT80'
```

### Скорая помощь



Чтобы воссоздать ключи SSL, используйте `rm /etc/uhttpd.*` и перезапустите `uhttpd`.

### Скорая помощь



Обратите внимание, что изменение настроек с помощью LuCI или *uci* переписывает файлы `config`. Все добавленные вручную комментарии или форматирование будут утрачены.

» Подпишитесь на печатную или электронную версии на [www.linuxformat.ru/subscribe!](http://www.linuxformat.ru/subscribe!)

## Скорая помощь



Прошивка может обеспечить файл `/etc/board.json`, который описывает карту номеров портов по умолчанию.

```
config wifi-iface 'default_radio0'
option device 'radio0'
option network 'lan'
option mode 'ap'
option ssid 'LEDE'
option encryption 'psk2'
option key 'my secret key'
```

В изначальной конфигурации имеется отключенная настройка. Обязательно удалите или раскомментируйте ее, и затем запустите Wi-Fi:

```
root@LEDE:~# wifi up
```

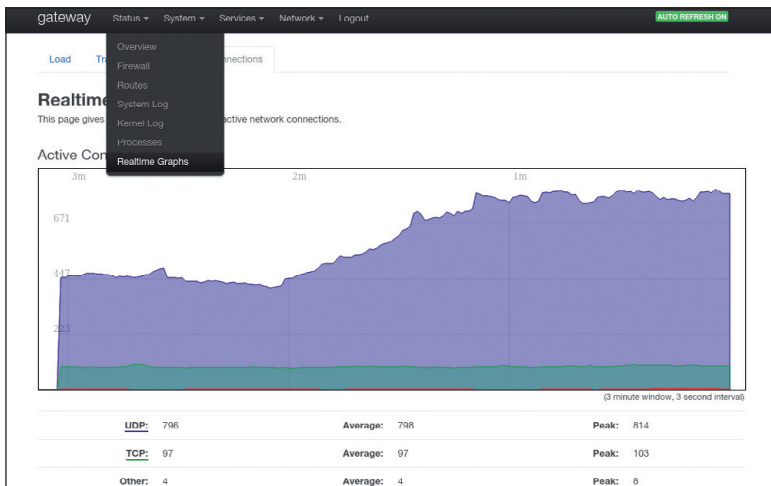
Теперь вы должны быть в состоянии подключиться к беспроводному устройству.

## На проводе

Настройка Интернета, или WAN, находится в `/etc/config/network`. Необходимые параметры будут зависеть от вашего интернет-провайдера и того, как поставляется сервис; базовый сервис ADSL2+ в Великобритании будет таким:

```
config dsl 'dsl'
option annex 'a'
option xfer_mode 'atm'
option line_mode 'adsl'
config interface 'wan'
option proto 'pppoa'
option username 'username@isp.example.uk'
option password 'my password'
option vpi '0'
option vci '38'
option encaps 'vc'
option ipv6 '0'
config device 'wan_dev'
option name 'pppoe-wan'
option macaddr '90:72:82:88:99:00'
```

Графики LuCI в реальном времени помогают следить за использованием сети.



Они будут другими для оптоволоконна (VDSL), которое обычно использует `pppoe`. Вам надо поэкспериментировать для выяснения, что работает с вашим интернет-провайдером.

Затем надо удалить исходный шлюз вашего роутера и DNS из раздела LAN. И сейчас также пора изменить имя хоста с LEDE на какое-то другое, если вы хотите это сделать. Загляните в `/etc/config/system`, где также можно настроить часовой пояс:

```
config system
option hostname 'gateway'
option timezone 'Europe/London'
```

На этом заканчивается базовая настройка. Выключите роутер с LEDE и переместите его в вашу сеть, подключив к нему свою линию DSL, а затем включите его. Вы должны увидеть работающий WAN и подключение к Интернету через роутер с LEDE.

## Самый лучший хост

В этом пункте у вас будет базовый интернет-шлюз, практически такой же, как предоставляемый вашим ИП, но с одним основным отличием: достойной прошивкой. У вас имеется управляемый коммутатор, мощный брандмауэр с `iptables`, а также DHCP и DNS-сервер. Всё остальное может быть доступно через менеджер пакетов `opkg`.

Вы можете настроить не только DHCP, но и статические хосты; каждый из них определен в своем собственном «доменном» разделе в файле `/etc/config/dhcp`. Блоки `host` резервируют DHCP-адреса для определенных хостов, чтобы у них всегда был один и тот же адрес. Оба присваивают DNS-имена:

```
config domain
option name myhost
option ip 192.168.10.1
config host
option name otherhost
option ip 192.168.10.2
option mac DE:AD:BE:EF:CA:FE
```

Вы также можете настроить альтернативные DNS-имена с помощью блоков `spname`:

```
config sname
option sname othername
option target myhost
```

И, наконец, простое правило брандмауэра делает хост доступным из Интернета. Правила брандмауэра определены в `/etc/config/firewall`, и соответствующее правило переадресации портов выглядит так:

```
config redirect
option name 'Web Server'
option src wan
option dest lan
option dest_ip 192.168.1.2
option dest_port 22
option proto tcp`
```

Правила брандмауэра работают по зонам, поэтому вышеприведенное перенаправляет из зоны WAN в зону LAN. Повсюду в том же файле настройки брандмауэра вы найдете определения,

## Контроль версий

Вы можете использовать `Git` для отслеживания изменений в ваших файлах настройки UCI. Тогда сначала используйте менеджер пакетов LEDE, `opkg`, для установки `Git`:

```
root@LEDE:~# opkg update
root@LEDE:~# opkg install git
```

Затем инициализируйте новый репозиторий в `/etc/config`, это директория настройки

UCI, и сделайте начальную отправку `[commit]` следующим образом:

```
root@LEDE:~# cd /etc/config
root@LEDE:~# git init
root@LEDE:~# git add .
root@LEDE:~# git commit -m 'Initial commit'
```

Отсюда и далее вы можете по своему желанию вносить изменения в настройку. В этот момент

вы можете настроить удаленный репозиторий и сделать `git push` для `config`; вы можете настроить `Git` по своему усмотрению и, возможно, добавить `.gitignore`.

Полезной может оказаться опция, которая отключает цветной вывод `Git` (терминал `Busybox`, используемый LEDE, не поддерживает цвета):

```
root@LEDE:~# git config --global color.ui false
```



указывающие, какие сетевые интерфейсы имеются в каждой зоне. Интерфейсы определены в файле **network**. Отредактировав его, не забудьте перезагрузить соответствующие config'i:

```
root@LEDE:~# /etc/init.d/dnsmasq restart
root@LEDE:~# /etc/init.d/firewall restart
```

## Руки проф от моей LAN...

Во многие сети проникают другие устройства, помимо компьютеров. Телефоны и планшеты, детские игрушки, т.н. персональные помощники типа Alexa и многое другое, предлагаемое набирающим популярность Интернетом Вещей. Вы можете не пускать подобные устройства в свою драгоценную LAN, используя демилитаризованную зону (DMZ) и создав виртуальный сетевой интерфейс. Мы настроим DMZ, чтобы помочь вам больше узнать о сетевой конфигурации LEDE.

Взглянув на сетевую конфигурацию, вы увидите, что существующий интерфейс LAN привязан к eth0.1, где суффикс говорит нам о том, что это — виртуальный интерфейс. Мы можем создать новый виртуальный интерфейс — скопируйте конфигурацию LAN, переименуйте ее в dmz и затем используйте другой суффикс:

```
config interface 'dmz'
option type 'bridge'
ifname 'eth0.2'
proto 'static'
ipaddr '172.20.0.1'
netmask '255.255.0.0'
config device 'lan_dev'
option name 'eth0.2'
option macaddr '54:64:d9:01:02:03'
```

Мы также присвоим новому интерфейсу MAC-адрес и собственную IP-подсеть (ipaddr и netmask). Чтобы DHCP-сервер выпустил адреса, понадобится добавление в файл **/etc/config/dhcp**:

```
config dhcp 'dmz'
option interface 'dmz'
option start '256'
option limit '255'
option leasetime '12h'
```

Начальное значение определяет первый адрес в блоке. В данном примере он будет **172.20.1.0**, а пределом будет количество адресов в блоке.

Брандмауэр должен разрешить DMZ-доступ в Интернет. Мы распределяем зону DMZ брандмауэра и определяем правило переадресации:

```
config zone
option name dmz
list network dmz
option input ACCEPT
option output ACCEPT
option forward ACCEPT
config forwarding
option src dmz
option dest wan
```

Вы можете распределить один или более Ethernet-портов роутера DMZ. Порты присоединяются к управляемому Ethernet-коммутатору в роутере, который настроен с помощью блоков коммутатора в файле **network**. Наша настройка по умолчанию выглядит так:

```
config switch
option name 'switch0'
option reset '1'
option enable_vlan '1'
config switch_vlan
option device 'switch0'
option vlan '1'
option ports '0 1 2 4 5 6t'
```

## Вторая точка доступа

Вы можете настроить LEDE как простую беспроводную точку доступа. Это хороший способ расширить диапазон вашей сети. Сделать это можно по-разному — см. <https://wiki.openwrt.org/doc/recipes/bridgedap> — лично мы подключили пару роутеров, использующих Ethernet-соединение.

Поскольку второй роутер является только беспроводной точкой доступа и сетевым мостом, ему не требуются функции DSL. Используются только беспроводные и Ethernet-компоненты.

Мы подключили Ethernet-кабель между портами, настроенными на каждом роутере в качестве магистральных, чтобы можно было проводить трафик LAN и DMZ. Мы также отключили и удалили DHCP и брандмауэр из точки доступа, поскольку эти функции обеспечиваются основным шлюзом:

```
$ opkg remove --force-depends luciapp-firewall firewall dnsmasq
```

Сеть и беспроводное соединение в точке доступа настраиваются так же, как шлюз.

Здесь вы видите настройку по умолчанию VLAN для зоны LAN (параметр `vlan` отражает суффикс сетевого интерфейса, описанного ранее). Она определяет включенные порты. Следует отметить, что номера этих портов могут не иметь никакого отношения к порядку физических портов на роутере. Некоторые номера могут не использоваться (например, порт 3 на нашем устройстве), и есть дополнительный логический порт, к которому внутренне подключен Ethernet-интерфейс роутера (например, eth0). Это порт 6 на нашем роутере, на котором 5 физических портов.

## Забивайте канал

Следует понять, что порт может нести трафик для нескольких виртуальных интерфейсов. Это называется Trunk ports [Магистральный порт] и определяется суффиксом "t", как, например, в порте 6 в примере. Сетевой трафик между магистральными портами «тегируется» идентификатором VLAN, который, поскольку такие порты могут быть в нескольких VLAN, позволяет коммутатору направлять трафик в нужный.

Чтобы распределить порты коммутатора DMZ, надо только удалить их из блока LAN и затем добавить в новый блок DMZ:

```
config switch_vlan
option device 'switch0'
option vlan '1'
option ports '0 1 5 6t'
config switch_vlan
option device 'switch0'
option vlan '2'
option ports '2 4 6t'
```

Вы также можете создать и отдельный Wi-Fi доступ к DMZ, он пригодится для создания гостевой учетной записи. Дополните config-файл «беспроводным» разделом:

»

Enabled	SSL req.	Description
<input checked="" type="checkbox"/>	Yes	focus on mobile ads, infrequent updates, approx. 400 entries
<input type="checkbox"/>	Yes	combined adguard dns filter list, approx. 12.000 entries
<input type="checkbox"/>	No	static local domain blacklist (always deny these domains)
<input checked="" type="checkbox"/>	Yes	mozilla driven blacklist, numerous updates on the same day, approx. 6.5k
<input type="checkbox"/>	Yes	generic blacklist, daily updates, approx. 4.500 entries
<input type="checkbox"/>	Yes	focus on feodo botnet, daily updates, approx. 0-10 entries
<input type="checkbox"/>	Yes	broad blacklist, monthly updates, approx. 50.000 entries

» Вы можете задать списки блокировок с помощью расширения **LuCI — adblock**.

```
config wifi-iface 'dmz_radio0'
option device 'radio0'
option network 'dmz'
option mode 'ap'
option ssid 'LEDE Guest'
option encryption 'psk2'
option key 'guest secret key'
option isolate 1
```

Это похоже на наш основной SSID, но подключается к сети DMZ вместо LAN. Опциональная изоляция не позволяет гостям Wi-Fi видеть друг друга.

Не забудьте перезапустить соответствующие сервисы: сеть, беспроводное соединение, брандмауэр и *dnsmasq* (для DHCP). Или просто перезагрузитесь!

## Защититесь от рекламы, прикройте детей и т.д!

Среди прочего, что вы можете сделать — родительский контроль (<https://lede-project.org/docs/user-guide/parentalcontrols>), настроив OpenVPN-сервер (<https://ledeproject.org/docs/user-guide/openvpn-server>), и одно из более простых и эффективных дополнений, которое запускает ad-blocker.

Установить LEDE adblock пакет относительно просто, и он заработает автоматически. Опциональный пакет LUCI добавляет меню Services > Adblock, где вы можете настроить сервис. Можно вы-

брать, какие blocklists использовать — перечислено несколько, из которых настройку по умолчанию используют три:

```
$ opkg install adblock luci-app-adblock
```

Еще одно интересное приложение — мониторинг трафика; оно полезно, если ваше широкополосное соединение имеет ограниченную полосу или просто потому, что вы хотите внимательно следить за сетью. Самое простое в настройке приложение для этого — Netlink Bandwidth Monitor, *nwblmon*, которое добавляет несколько симпатичных страниц к LuCI. Установите его с помощью

```
opkg install luci-app-nlwbmon
```

и выберите новую страницу LuCI “Bandwith Monitor”.

## Выбирайте NetFlow

Альтернативным или дополнительным методом, который вы можете использовать, является NetFlow. Этот протокол ведет свое начало от роутеров Cisco, являясь способом сбора данных о сетевом трафике. Устройства отправляют данные NetFlow в коллектор, который работает на другом хосте. Одним из коллекторов с открытым кодом является nfdump ([www.nfdump.sourceforge.net](http://www.nfdump.sourceforge.net)), и его можно использовать совместно с web-приложением под названием *Netflow Sensor*, или *NfSen* ([www.nfsen.sourceforge.net](http://www.nfsen.sourceforge.net)), чтобы данные о сетевом трафике были доступны в браузере.

Для отправки данных NetFlow LEDE поддерживает *softflowd*, который легко устанавливается (`opkg install softflowd`) и поддерживается в `/etc/config/softflowd`:

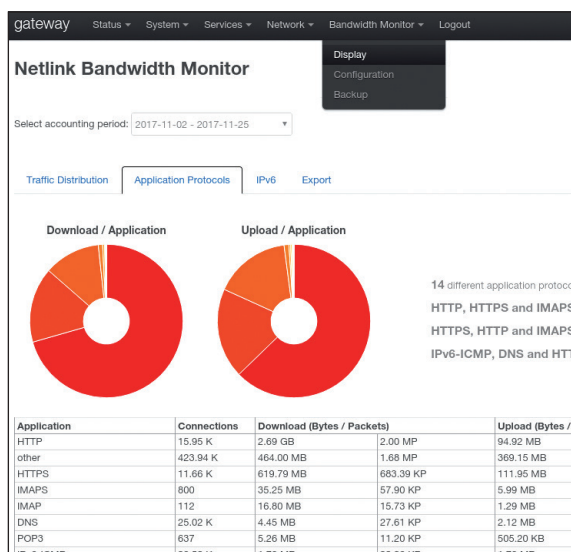
```
config softflowd
option enabled '1'
option interface 'pppoe-wan'
option host_port '192.168.1.10:2055'
option export_version '9'
option track_ipv6 '1'
```

В этом примере `host_port` — это IP-адрес, куда надо отправлять пакеты NetFlow. Это еще один компьютер Linux в вашей сети, и вам надо установить на него *nfsen* и его зависимости.

LEDE предназначен не только для роутеров. Его можно использовать на не-сетевых устройствах. Полный список поддерживаемых устройств вы найдете в официальной Таблице оборудования [Table of Hardware] на <https://lede-project.org/toh>. Обратите также внимание, что он не всегда будет LEDE, поскольку принято решение о повторном слиянии с проектом OpenWRT, что подразумевает возврат к исходному наименованию. Пока этого не случилось, ответвление LEDE остается самым активным и лучше всего поддерживаемым — просто не забывайте, что нас ждет исчезновение данного бренда. **LXF**

**Скорая помощь**

Используйте logread для просмотра лог-файла, например: `logread -e network`.



➤ Проводить мониторинг полосы с LEDE очень легко.

## Борьба с раздутым буфером

Хотя это совершенно нормально — жаловаться на своего интернет-провайдера (даже в соцсетях и на званных обедах), всё же ваше интернет-соединение, возможно, не такое уж скверное.

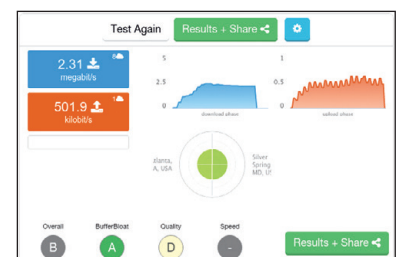
Если вы страдаете от буферизации в вашей сети, то она, возможно, страдает от раздутого буфера. Причиной являются сетевые устройства, отправляющие слишком много данных в буфер, когда сеть под нагрузкой. Проще говоря, пользователь, скачивающий большое количество данных, может повлиять на видеозвонок другого пользователя. Эта проблема объясняется чуть более подробно на [www.bufferbloat.net](http://www.bufferbloat.net).

LEDE предлагает пакет *Smart Queue Management*, чтобы помочь разобраться с проблемой переполнения буфера. Вы можете установить его и также удалить избыточные пакеты QoS, если они установлены:

```
$ opkg remove qos-scripts luci-app-qos
$ opkg install luci-app-sqm
```

Настройте в `/etc/config/sqm` или перейдите в Network > SQM QoS в LuCI.

Затем вы можете протестировать свою сеть на предмет переполнения буфера, например, с помощью теста на скорость DSLReports на [www.dslreports.com/speedtest](http://www.dslreports.com/speedtest).



➤ SQM поможет извлечь максимум из широкополосного соединения среднего качества.

➤ Пропустили номер? Узнайте на с. 108, как получить его прямо сейчас.



# Новое поколение средств защиты

## Межсетевые экраны ССПТ, не имеющие IP-адреса

ССПТ-2 — это сертифицированное ФСТЭК, ФСБ и ГАЗПРОМСЕРТ средство защиты информации нового поколения, реализующее функции меж сетевого экрана, но при этом остающееся «невидимым» для любых протоколов и тестовых воздействий, что достигается за счет отсутствия физических и логических адресов на его фильтрующих интерфейсах. ССПТ-2 невозможно обнаружить никакими известными средствами удаленного мониторинга сети.

Скрытность функционирования меж сетевого экрана повышает надежность системы защиты в целом и существенно упрощает процедуру установки ССПТ-2 в компьютерные сети и функционирующие на их основе информационные и телематические системы.

Защита для высокоскоростных корпоративных сетей Ethernet 100/1000 Мбит/с

Сертифицированы ФСТЭК и ФСБ (3-й класс защиты)

На базе процессоров с 64-разрядной многоядерной архитектурой



## Назначение устройства

Основное средство защиты для реализации различных политик информационной безопасности с помощью:

- фильтрации пакетов на канальном, сетевом, транспортном и прикладном уровнях;
- управления транспортными соединениями между отдельными узлами ЛВС или виртуальной ЛВС (VLAN);
- контроля контента данных на прикладном уровне с учетом направления, времени и типа протоколов передачи трафика.

Дополнительное устройство защиты для:

- обеспечения безопасности функционирования ранее установленных в компьютерной сети средств защиты и устройств маршрутизации;
- мониторинга трафика с возможностью анализа данных регистрации пакетов по различным критериям и интеграции с IDS;
- обеспечения функционирования сетевых распределенных телематических приложений и GRID-ресурсов.

Москва  
+7 (499)

271-49-54

Санкт-Петербург  
+7 (812)

309-06-86

Linux-эксперт для вашего бизнеса. [www.linuxcenter.ru](http://www.linuxcenter.ru)

Linux  center

# АНАТОМИЯ ТУЛЧЕЙНОВ

Вооружившись виртуальным скальпелем, **Дмитрий Кузнецов** препарирует важнейшие инструменты разработчика.



**Наш  
эксперт**

**Дмитрий Кузнецов** более 10 лет ведет неравный бой с монстрами сложности, хотя некоторые почему-то считают их ветряными мельницами.

**К**омпиляторы — одна из самых сложных и интересных областей программирования. К счастью, у пользователей Linux есть отличная возможность изучать ее не только в теории, но и на практике. В любом дистрибутиве Linux можно найти *GCC* (GNU Compiler Collection) — набор компиляторов промышленного качества для нескольких популярных языков программирования, разработанный в рамках проекта GNU. Разумеется, исходный код *GCC* открыт и доступен для изучения всем желающим. Однако первые проблемы подстерегают новичков задолго до погружения в премудрости синтаксического анализа и генерации кода. Просто собрать *GCC* — задача не из легких.

## Что такое тулчейн

Казалось бы, сборка проектов — стандартная и давно отработанная процедура. А тем, кто заинтересовался устройством *GCC*, рассказывать про знаменитую триаду `./configure && make && make install` просто смешно. Даже богатый набор конфигурационных ключей и зависимости от вспомогательных библиотек обычно не сильно усложняют сборку. В сухой теории так оно и есть, но древо жизни *GCC* не просто зеленеет, а пышно цветет и колосится.

Дело в том, что *GCC* может нормально работать только в составе так называемого тулчейна [*англ.* toolchain — цепочка инструментов]. Этот англицизм, уже укоренившийся в русскоязычном профессиональном сообществе, подразумевает каскадно соединенный набор инструментов, решающий общую задачу — преобразование программы, написанной на языке программирования высокого уровня, в исполняемый файл. *GCC* — лишь одно из звеньев этой цепочки, которому, при всей его важности, необходимо кооперироваться с другими. В этом источник большинства проблем сборки тулчейнов. Как их решить, будет показано ниже, а сейчас кратко перечислим основные компоненты тулчейна, для простоты пока опуская некоторые вспомогательные библиотеки.

- » *GCC* — набор компиляторов с языков высокого уровня (C, C++, Ada, Fortran) в ассемблер.
- » *binutils* — инструменты для манипуляций с объектными файлами и компиляции ассемблерного кода в машинный.
- » *libc* — стандартная библиотека языка C.
- » Заголовочные файлы ядра Linux.
- » Набор отладочных инструментов (отладчик *GDB*; библиотеки для поиска утечек памяти *duma*, *dmalloc*; утилиты-трассировщики *strace* и *ltrace*). Строго говоря, это необязательные компоненты, однако в реальных проектах они могут оказаться очень полезны.

## Типы тулчейнов

*GCC* поддерживает несколько десятков процессорных архитектур. ARM, x86, MIPS, SPARK, PowerPC — только самые известные из них. Процесс сборки тулчейна усложняется еще и тем, что в общем случае в него вовлечены три машины:

- » *build* [*англ.* строить] — машина, на которой выполняется сборка тулчейна.

- » *host* [*англ.* хозяин] — машина, на которой собранный тулчейн будет работать.

- » *target* [*англ.* цель] — машина, для которой этот тулчейн будет генерировать код.

Каждая из них может иметь любую из поддерживаемых архитектур. Исходя из этого, различают несколько типов тулчейнов:

- » *Native-тулчейн* [*англ.* native — родной], когда `build = host = target`. Все три архитектуры одинаковы. В качестве примера можно привести тулчейн, имеющийся в составе любого дистрибутива Linux. Он собран на машине с архитектурой x86, на ней же работает и для нее же компилирует программы.

- » *Cross-тулчейн* [*англ.* cross — крест, перекрестный], когда `build = host != target`. Тулчейн собирается и работает на машине с одной архитектурой, а генерирует код для машин с другой. Классический пример — кросс-тулчейны x86/ARM, активно применяемые при программировании для встроенных систем. Такой подход позволяет программисту комфортно работать и быстро компилировать код на мощном настольном компьютере, а слабую ARM-машину использовать только для тестирования уже готовой программы.

- » *Cross-native-тулчейн*, когда `build != host = target`.

- » *Canadian-тулчейн* [*англ.* canadian — канадский; намек на канадскую трехпартийную политическую систему], когда `build != host != target`.

Тулчейны первых двух типов используются наиболее часто; остальные имеют более узкие сферы применения.

## Crosstool-NG

Столкнувшись с проблемой, всегда полезно поинтересоваться, что в подобной ситуации делают другие. Может быть, уже есть готовое решение. Сборка тулчейнов — как раз такой случай. Для ее автоматизации есть специальный инструмент — *Crosstool-NG*, проект с открытым исходным кодом, который уже много лет существует и активно развивается. Не вдаваясь в детали, можно считать, что он состоит из двух частей:

- » Заимствованная из ядра Linux конфигурационная система; она позволяет задать для собираемого тулчейна необходимые параметры.
- » Набор скриптов командной оболочки для выполнения сборки в соответствии с выбранной конфигурацией.

Пользовательский интерфейс *Crosstool-NG* прост. Тем, кто хоть раз собирал ядро Linux, он вряд ли покажется незнакомым. Главная проблема — разобраться в назначении параметров. К счастью, *Crosstool-NG* содержит множество готовых конфигураций, так что даже новичку есть с чего начать свои эксперименты. Кроме того, в архиве с исходным кодом (подкаталог **docs**) можно найти довольно содержательную документацию, а все этапы сборки тулчейна аккуратно заносятся в журнал. Это дает отличную возможность детально разобраться в процессе сборки, не изучая исходный код *Crosstool-NG*.



## Пробуем Crosstool-NG

Попробуем собрать тулчейн. Пока, конечно, с помощью магии *Crosstool-NG*. Затем принципиальные противники чародейства и волшебства смогут проанализировать полученный журнал и научиться делать то же самое вручную.

Скачаем и распакуем последнюю версию *Crosstool-NG*:

```
#wget http://crosstool-ng.org/download/crosstool-ng/crosstool-
ng-1.23.0.tar.xz
#tar -xvf ./crosstool-ng-1.23.0.tar.xz
```

Как и большинство других проектов, *Crosstool-NG* имеет зависимость, поэтому сначала необходимо установить несколько пакетов ПО. В Ubuntu 16.04 LTS это можно сделать так:

```
#apt-get install build-essential flex bison gperf expat texinfo
help2man gawk libncurses5-dev python-dev
```

Сборка не отличается оригинальностью:

```
#cd ./crosstool-ng-1.23.0/ && ./configure --enable-local && make
```

Ключ `enable-local` означает, что пользователь не собирается устанавливать *Crosstool-NG* в систему. Это никак не ограничивает функциональность, просто все команды придется запускать из каталога `./crosstool-ng-1.23.0`. Зато для полного удаления *Crosstool-NG* достаточно будет лишь удалить этот каталог.

Всё, *Crosstool-NG* готов к работе. Но прежде чем начать его использовать, стоит познакомиться с соглашением именования тулчейнов. Оно довольно простое. Название тулчейна должно состоять из четырех компонентов, разделенных дефисами: `<arch>-<vendor>-<OS>-<libc/abi>`, где

» `<arch>` — процессорная архитектура (`arm`, `armv7`, `x86_64`, `mips`).  
 » `<vendor>` — название производителя в произвольной форме (часть пишут `unknown` или вообще опускают).

» `<OS>` — операционная система. Здесь обычно встречаются два варианта: `linux` и `none`. С первым всё ясно, а второй используется для обозначения тулчейнов специального вида. Их еще называют `bare metal` [англ. голое железо]. С их помощью собирают загрузчики или ПО для устройств без ОС. Для компиляции прикладных программ, работающих под управлением Linux, такие тулчейны непригодны.

» `<libc/abi>` — описание используемой стандартной библиотеки (`glibc`, `uClibc`, `newlib`, `musl`) и ABI (Application Binary Interface, от англ. двоичный интерфейс приложений). ABI — это набор соглашений для доступа приложения к ОС и другим низкоуровневым сервисам. Например, в названиях ARM тулчейнов довольно часто встречается `gnueabi`. Это означает, что они собраны с библиотекой `glibc` (GNU C Library) и поддержкой EABI (Embedded ABI, от англ. двоичный интерфейс встраиваемых приложений).

Итак, возвращаемся к *Crosstool-NG*. Попробуем собрать `cross-tулчейн x86/ARM`. Посмотрим список доступных примеров конфигураций и возьмем один из них за основу:

```
#!/ct-ng list-samples
```

Кажется, `arm-unknown-linux-gnueabi` — неплохой вариант.

А так можно посмотреть краткую справку о его параметрах:

```
#!/ct-ng show-arm-unknown-linux-gnueabi
```

Если всё устраивает, выбираем его:

```
#!/ct-ng arm-unknown-linux-gnueabi
```

При необходимости можно запустить систему конфигурации, чтобы подробнее изучить параметры или откорректировать их:

```
#!/ct-ng menuconfig
```

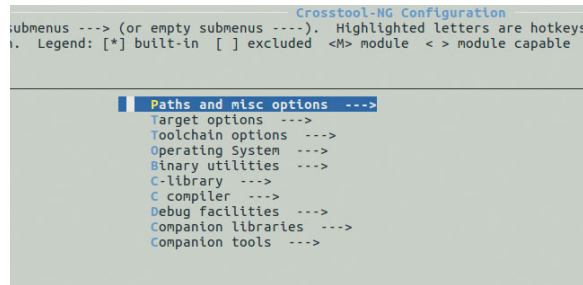
Теперь остается только запустить сборку и дождаться ее окончания:

```
#!/ct-ng build
```

Это может занять несколько десятков минут, в зависимости от мощности рабочего ПК.

## Что получилось

Если последняя команда завершилась без ошибок, *Crosstool-NG* успешно справился с задачей. А это значит, что в `$(HOME)/x-tools/`



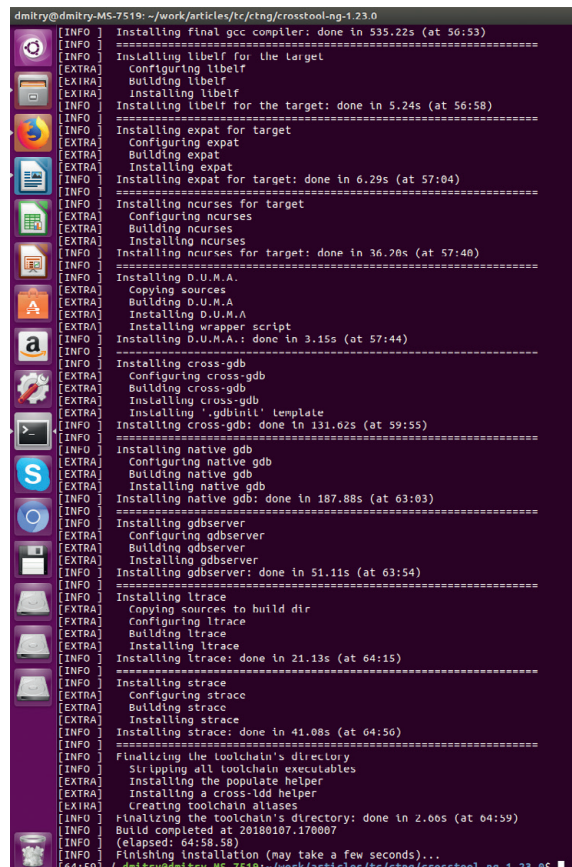
» Рис. 1. *Crosstool-NG*. Конфигурационная система.

`arm-unknown-linux-gnueabi` (путь можно изменить в системе конфигурации перед сборкой) лежит готовый тулчейн. К сожалению, содержимое этого каталога простотой не отличается. Команда `find . | wc -l` насчитает более 7500 файлов и каталогов, а потому имеет смысл немного в них разобраться.

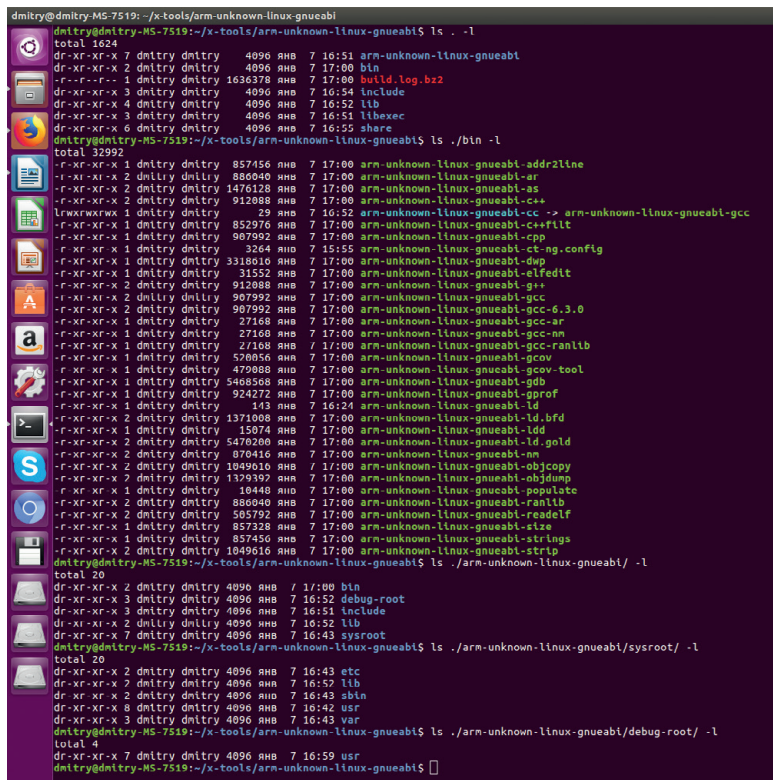
Начнем с самого интересного. Файл `build.log.bz2` — это подробный журнал, по которому легко восстановить каждую команду, необходимую для сборки тулчейна.

Каталог `bin` можно назвать пользовательским интерфейсом тулчейна. В нем лежат компиляторы и полный набор утилит из пакета `binutils`. Вся полезная работа выполняется через них. У имени каждой программы есть префикс `<имя тулчейна>`. Например, компилятор `gcc` в данном случае будет называться `arm-unknown-linux-gnueabi-gcc`. Часто для удобства путь к этому каталогу добавляют в переменную окружения `$PATH`. Префикс позволяет избежать конфликта имен при одновременном использовании нескольких тулчейнов.

Каталог `arm-unknown-linux-gnueabi/sysroot` содержит полный набор системных заголовочных файлов, а также библиотек и программ, собранных под целевую архитектуру (в данном случае, ARM). Они реализованы в разных компонентах тулчейна и необходимы для сборки любого прикладного ПО. Кроме того, они обязательно должны присутствовать в корневой файловой системе



» Рис. 2. *Crosstool-NG*. Успешное окончание сборки тулчейна.



➤ Рис. 3. Собранный тулчейн.

(КФС) целевой машины. Фактически, **sysroot** — это почти готовая КФС. Достаточно добавить в нее *busybox* (набор UNIX-утилит командной строки, реализованный в виде одного исполняемого файла; часто используется во встраиваемых системах) и ядро Linux, чтобы получить минимальный дистрибутив, пригодный для запуска на машине с подходящей процессорной архитектурой.

Каталог `arm-unknown-linux-gnueabi/debug-root` можно считать дополнением к **sysroot**. Он содержит программы и библиотеки, предназначенные для отладки. Например, библиотеку *libduma.so* для поиска утечек памяти, трассировщики *strace* и *ltrace* или отладчик *gdb*.

Остальные каталоги имеют служебное значение. Они, конечно, тоже важны, но служат для обеспечения работы перечисленных выше компонентов.

## Как использовать

Попробуем новый тулчейн в деле. Начнем с простейшего примера — “hello, world” на C++.

Пусть его исходный код хранится в файле `main.cpp`. Поскольку он в нашем проекте единственный, никакие сборочные системы не понадобятся. Даже *Makefile* не нужен. Достаточно просто вызвать компилятор C++:

```
# g++ ./main.cpp -o hw
```

Однако в данном случае используется g++ из системного native-тулчейна, а следовательно, получается исполняемый файл *hw* для архитектуры x86\_64 (если на рабочем ПК установлена 64-разрядная ОС). Это легко проверить командой `file ./hw`, которая наряду с прочей информацией о файле печатает и его целевую процессорную архитектуру. Ну и, конечно же, программа *hw* должна запускаться и печатать свое приветствие миру на рабочем ПК.

Чтобы собрать тот же код для архитектуры ARM, достаточно лишь заменить `g++` на собранный ранее кросс-компилятор `arm-unknown-linux-gnueabi-g++`:

```
~/x-tools/arm-unknown-linux-gnueabi/bin/arm-unknown-linux-gnueabi-g++ ./main.cpp -o hw
```

Путь к новому тулчейну не был добавлен в переменную окружения `PATH`, а потому его приходится явно указывать при вызове

кросс-компилятора. Это, возможно, выглядит несколько громоздко, но на первых порах так понятней. Теперь полученный исполняемый файл *hw* не запускается на рабочем компьютере, ведь у него другая целевая архитектура. В этом легко удостовериться, вызвав команду `file ./hw`.

Возьмем пример посложнее: небольшой, но полноценный проект “GNU sed”. Скачаем его исходный код:

```
# wget http://ftp.gnu.org/pub/gnu/sed/sed-4.5.tar.xz
```

Распакуем полученный архив:

```
# tar -xJvf ./sed-4.5.tar.xz
```

Это классический проект GNU, для сборки которого используется система *autotools* (еще ее называют GNU Build System). Такие проекты, как известно, собираются тремя командами `./configure && make && make install`. Однако это самый простой вариант, когда всем конфигурационным ключам присваиваются значения по умолчанию, сборка выполняется в каталоге с исходным кодом (а значит, там же создаются все промежуточные файлы, необходимые в процессе сборки), а готовое ПО устанавливается в системные каталоги. Разумеется, перспектива затереть системную утилиту *sed* не радует. Поэтому придется использовать чуть более сложный вариант сборки.

Допустим, исходный код *sed* был распакован в каталог `/home/dmitry/tc/sed/`. Создадим здесь подкаталоги `build` и `install`:

```
# mkdir ./build ./install
```

Из первого будем запускать скрипт *configure*, а потому все промежуточные файлы будут создаваться именно там. Второй укажем в ключе `--prefix`, который задает путь установки собранного ПО.

```
# cd ./build
# ../sed-4.5/configure --prefix="/home/dmitry/tc/sed/install"
# make && make install
```

Результат сборки — утилиту *sed* — можно найти в `./install/bin`. Она, конечно, собрана с помощью системного native-тулчейна, а потому ее целевая архитектура совпадает с архитектурой рабочего ПК. Это можно проверить уже упоминавшейся выше командой `file`. Разумеется, `./install/bin/sed` должна без труда запускаться. Например, вот простейший тест — замена во входном тексте ‘123’ символа 2 на A:

```
# echo '123' | ./install/bin/sed 's/2/A/'
```

Как же собрать *sed* для машин с архитектурой ARM? Ответ на этот вопрос нетрудно найти в справочной информации скрипта *configure*. Получить ее можно так:

```
# ../sed-4.5/configure -h
```

Оказывается, решение довольно простое: переменная окружения `CC` задает компилятор, а ключ `--host` — целевую архитектуру.

Очистим каталоги `build` и `install`, а затем попробуем собрать *sed* еще раз.

```
# cd ./build
# CC="/home/dmitry/x-tools/arm-unknown-linux-gnueabi/bin/arm-unknown-linux-gnueabi-gcc" ../sed-4.5/configure --prefix="/home/dmitry/tc/sed/install" --host=arm
# make && make install
```

Проверяем целевую архитектуру собранной программы *sed*:

```
# file ./install/bin/sed
```

Действительно, ARM.

## Алгоритм сборки тулчейна

Как уже было сказано, *Crosstool-NG* ведет подробный журнал, анализ которого позволяет восстановить точную последовательность выполненных команд и детально разобраться в процедуре сборки тулчейна. Однако не стоит с ходу погружаться в его изучение. Обилие низкоуровневых деталей может помешать «за деревьями разглядеть лес». Сначала гораздо полезнее получить общее представление об алгоритме сборки. Он отлично описан в документации *Crosstool-NG* (`docs/9_Toolchain_Construction.md`). Этот раздел статьи — лишь краткий пересказ изложенных там идей. Конечно,



```

dmitry@dmitry-MS-7519: ~/tc/hw
dmitry@dmitry-MS-7519:~/tc/hw$ ls
main.cpp
dmitry@dmitry-MS-7519:~/tc/hw$ cat ./main.cpp
#include <iostream>

int main()
{
    std::cout << "Hello, world!" << std::endl;
    return 0;
}
dmitry@dmitry-MS-7519:~/tc/hw$ g++ ./main.cpp -o hw
dmitry@dmitry-MS-7519:~/tc/hw$ ls
hw main.cpp
dmitry@dmitry-MS-7519:~/tc/hw$ file ./hw
./hw: ELF 64-bit LSB executable, x86-64, version 1 (SYSV), dynamically linked, interpreter /lib64/ld-linux-x86-64.so.2, for GNU/Linux 2.6.32, BuildID[sha1]=9922459b06118
ripped
dmitry@dmitry-MS-7519:~/tc/hw$ ./hw
Hello, world!
dmitry@dmitry-MS-7519:~/tc/hw$ ~/x-tools/arm-unknown-linux-gnueabi/bin/arm-unknown-linux-gnueabi-g++ ./main.cpp -o hw
dmitry@dmitry-MS-7519:~/tc/hw$ file ./hw
./hw: ELF 32-bit LSB executable, ARM, EABI5 version 1 (SYSV), dynamically linked, interpreter /lib/ld-linux.so.3, for GNU/Linux 4.10.8, not stripped
dmitry@dmitry-MS-7519:~/tc/hw$ ./hw
bash: ./hw: cannot execute binary file: Exec format error
dmitry@dmitry-MS-7519:~/tc/hw$

```

► Рис. 4. Сборка при-  
мера “hello, world”.

при наличии свободного времени и хотя бы минимальных знаний английского языка лучше обратиться к оригиналу.

Компоненты тулчейна зависят друг от друга. Следовательно, важен порядок их сборки. Но это еще полбеды, главная проблема — в циклической зависимости между *GCC* и *libc*. *GCC* должен уметь использовать *libc*, а для сборки *libc* необходим *GCC*. «Проблема курицы и яйца», как говорят любители метафор. С первой зависимостью ничего сделать нельзя, а вот от второй легко избавиться. Это возможно благодаря тому, что для сборки *libc* вполне подходит ограниченная версия *GCC*, которую можно собрать специально для этой цели. В терминологии *Crosstool-NG* она называется “pass-2 core C compiler”. К сожалению, этому вспомогательному компилятору тоже нужна *libc*, но не вся, а лишь ее небольшая часть: заголовочные и стартовые файлы. Стартовые файлы по-другому называют CRT или “C runtime” (от *англ.* run time — период выполнения). Для их сборки можно использовать еще более урезанную версию *GCC*, которая от *libc* совершенно не зависит. Ее в *Crosstool-NG* называют “pass-1 core C compiler”.

Кроме того, надо не забыть учесть еще несколько зависимостей. Во-первых, для *libc* необходимы заголовочные файлы ядра Linux. Во-вторых, *GCC* генерирует код ассемблера, для компиляции которого в машинный код требуется *binutils*. В-третьих, *GCC* зависит от вспомогательных библиотек *GMP*, *MPFR* и *MPC*, а при определенной конфигурации — еще и от *PPL*, *ISL*, *CLooG* и *libelf*. Некоторые из них также зависят друг от друга, а потому обязательно должны собираться в указанном порядке. Это сложные математические библиотеки. Их детальное рассмотрение выходит за рамки данной статьи. Подробное обсуждение этой темы без погружения в архитектуру *GCC* и теорию построения компиляторов совершенно невозможно. Тем не менее, ниже назначение каждой библиотеки будет кратко описано. Однако эту информацию скорее стоит рассматривать лишь как материал для формирования поисковых запросов.

В итоге получается такая последовательность сборки:

- » *GMP* (GNU Multi-Precision Library) — библиотека для вычислений с числами произвольной точности.
- » *MPFR* (Multiple-Precision Floating-point computations with correct Rounding) — библиотека для вычислений с плавающей запятой произвольной точности и корректным округлением. Зависит от *GMP*.
- » *MPC* (Multiple-Precision Complex) — библиотека для вычислений с комплексными числами произвольной точности. Зависит от *GMP* и *MPFR*.
- » *PPL* (Parma Polyhedra Library) — библиотека для работы с полидрами (родилась в университете итальянского города Парма). Зависит от *GMP*.
- » *ISL* (Integer Set Library) — библиотека для манипулирования целочисленными множествами и отношениями между ними.
- » *CLooG* (Chunky Loop Generator) — генератор циклов в полидральной модели. Библиотека, используемая *GCC* для

оптимизации генерации кода. Она является частью проекта Chunky [*англ.* плотный, коренастый], исследовательского инструмента, применяемого для улучшения локальности данных.

- » *libelf* — библиотека для работы с форматом ELF (основной формат исполняемых файлов в ОС Linux).
- » *binutils*.
- » pass-1 core C compiler.
- » заголовочные файлы ядра Linux.
- » заголовочные и стартовые файлы *libc*.
- » pass-2 core C compiler.
- » полная версия *libc*.
- » полная версия *GCC*.

Это основные компоненты тулчейна. Их сборка дает полноценный компилятор. При необходимости с его помощью нетрудно собрать дополнительные компоненты. Например, отладочные инструменты (*gdb*, *duma*, *dmalloc*, *strace*, *ltrace*).

## Журнал

Теперь можно заняться журналом. Он хранится в каталоге с собранным тулчейном: `$(HOME)/x-tools/arm-unknown-linux-gnueabi/build.log.bz2`. Это обычный сжатый текстовый файл, который без труда можно распаковать утилитой *bunzip2* и открыть в любом текстовом редакторе:

```
# bunzip2 ./build.log.bz2
# vim ./build.log
```

В этом файле почти 300 тысяч строк. Поэтому на первый взгляд идея заняться его изучением не кажется перспективной. Однако даже беглый просмотр показывает, что значительная часть его содержимого может быть опущена — например, вывод команд *wget*, *tar*, *./configure* и *make*, использующихся при загрузке, распаковке и компиляции компонентов тулчейна. С другой стороны, самые важные строки — исполняемые команды — снабжены специальным префиксом `==> Execution`, что очень облегчает их поиск.

К сожалению, рамки данного урока не позволяют подробно проанализировать журнал и заняться сборкой тулчейна вручную. Да и, вероятно, далеко не всем интересно читать занудные списки конфигурационных параметров тех или иных компонентов тулчейна. Всё равно ничего не запомнишь, пока своими руками с ними не повозимся. Думаю, желающие без труда смогут найти в журнале интересующие их детали самостоятельно.

## Заключение

*GCC*-тулчейны — чрезвычайно обширная, сложная и запутанная тема. На нескольких страницах достойно осветить ее во всех подробностях совершенно нереально. Поэтому можно вполне справедливо отметить, что статья получилась поверхностной и практически каждый абзац оставляет больше вопросов, чем дает ответов. Тем не менее, надеюсь, она послужит неплохим обзором для новичков, а также стимулом для дальнейшего углубления своих познаний. **LXF**

# CMake: Лучшие средства сборки

Михалис Цукалос описывает проект *CMake* и учит вас пользоваться его файлами конфигурации и визуализировать зависимости проекта в *Graphviz*.



Наш эксперт

**Михалис Цукалос** — администратор UNIX, программист, администратор баз данных и математик, который любит писать технические статьи. Он — автор книги «Системное программирование на Go».



```
3. mtsouk@mail: ~/docs/article/working/CMake.LXF234/code/hw/build (ssh)
hw$ ll
total 8
-rw-r--r-- 1 mtsouk mtsouk 78 Dec 30 21:45 CMakeLists.txt
-rw-r--r-- 1 mtsouk mtsouk 94 Dec 30 19:47 hw.c
hw$ mkdir build
hw$ cd build/
build$ cmake ..
-- The C compiler identification is GNU 4.9.2
-- The CXX compiler identification is GNU 4.9.2
-- Check for working C compiler: /usr/bin/cc -- works
-- Check for working C compiler: /usr/bin/cc -- works
-- Detecting C compiler ABI info
-- Detecting C compiler ABI info - done
-- Check for working CXX compiler: /usr/bin/c++
-- Check for working CXX compiler: /usr/bin/c++ -- works
-- Detecting CXX compiler ABI info
-- Detecting CXX compiler ABI info - done
-- Configuring done
-- Generating done
-- Build files have been written to: /home/mtsouk/docs/article/working/CMake.LXF234/code/hw/build
build$ make
Scanning dependencies of target LXF
[100%] Building C object CMakeFiles/LXF.dir/hw.c.o
Linking C executable LXF
[100%] Built target LXF
build$ ll
total 36
```

➤ Вывод команды `cmake` при работе с простым файлом конфигурации `CMakeCache.txt` для проекта, который содержит всего один исходный файл на C.

cmake version 3.10.1

CMake suite maintained and supported by Kitware (kitware.com/cmake).

Желая посмотреть на многочисленные параметры *cmake*, выполните команду `cmake --help`. После этого краткого введения мы готовы приступить к созданию и сборке проектов с помощью *CMake*.

## Основы Make

Простейший пригодный для нас проект включает всего один файл исходного кода. Пусть у вас есть файл с именем `hw.c`, который содержит код C программы *Hello World*, и вы хотите управлять этим файлом с помощью *CMake*.

Конфигурация *CMake* этого простого проекта будет содержаться в файле `CMakeLists.txt`, который по умолчанию является файлом конфигурации для всех проектов *CMake*. Содержимое этого простого файла `CMakeLists.txt` выглядит следующим образом:

```
cmake_minimum_required(VERSION 3.0)
PROJECT( LXF )
add_executable(LXF hw.c)
```

В первой строке определяется минимальная версия *CMake*, необходимая для данного проекта; во второй — название проекта; в третьей же перечислены файлы проекта (`hw.c`), а также имя исполняемого файла (`LXF`).

Обратите внимание, что файлы сборки рекомендуется расположить в отдельном каталоге `build` в корневом каталоге проекта, чтобы не засорять корневой каталог. Кроме того, поскольку *cmake* не предоставляет команду очистки, для очистки проекта вам придется просто удалить каталог `build` вручную.

**К** раеугольный камень процесса сборки — *CMake*, открытый проект, широко поддерживаемый Kitware. Если вы хотите собирать и компилировать код как профессионал, придется познакомиться с ним! На нашем уроке мы покажем, как использовать *CMake* для компиляции, компоновки и установки программного обеспечения на компьютерах Linux и UNIX. Затем мы визуализируем зависимости проекта *CMake* с помощью вывода утилиты *Graphviz*. Помните, что если у вас есть правильно заполненный конфигурационный файл *CMake*, утилита *cmake* сделает большую часть скучной работы за вас. Однако следует отметить, что для обработки проекта *CMake* необходима утилита GNU *make*.

Скорее всего вы установите *CMake* через обычный менеджер пакетов, который упростит задачу. Можно также установить некоторые дополнительные пакеты, которые упростят жизнь в долгосрочной перспективе — в Ubuntu Linux для этого выполните следующую команду:

```
$ sudo apt-get install cmake cmake-data cmake-doc cmake-extras
```

Затем вы можете узнать, какая версия *CMake* у вас установлена, выполнив команду

```
$ cmake --version
```

### Скорая помощь

Дополнительные сведения о *CMake* можно получить на <https://cmake.org> и <https://cmake.org/documentation>. Про *CTest* можно прочесть на странице [https://cmake.org/Wiki/CMake/Testing\\_With\\_CTest](https://cmake.org/Wiki/CMake/Testing_With_CTest), а про *CPack* — на странице [https://cmake.org/Wiki/CMake/Packaging\\_With\\_CPack](https://cmake.org/Wiki/CMake/Packaging_With_CPack).



## Чем CMake может вам послужить?

Прежде всего, вы должны знать, что *CMake* — это сокращение от “Cross Platform Make”, а значит, вы можете создавать файлы для всех вариантов UNIX, а также для компьютеров Windows с помощью *Microsoft Visual Studio*.

*CMake* идеально подходит для работы с большими и сложными проектами и может самостоятельно

распознавать, какие компиляторы нужны для исходных файлов. Эта возможность автоматического распознавания — основная причина применения *CMake*. Также есть версия *smake* для curses — *smake*, она позволяет работать с *smake* более интерактивно. Использование *smake* во время изучения *smake* — хорошая практика. Тем не менее, когда

вы получите большой опыт в *CMake*, вы перейдете к версии для командной строки, которая позволяет быстрее работать с проектами *CMake*.

Помните, что хотя *CMake* позволяет добиться очень многого, вы не обязаны задействовать все возможности *CMake* сразу или внедрять их во все ваши проекты!

После создания синтаксически правильного файла *CMake* надо выполнить команду `smake` с одним параметром — путем к корневому каталогу вашего проекта. При этом автоматически создадутся некоторые дополнительные файлы, в том числе **Makefile**, который нужен утилите *make*. Итак, надо выполнить следующие действия:

```
$ cd hw
$ ls -l
total 12
drwxr-xr-x 2 mtsouk mtsouk 4096 Dec 30 21:45 build
-rw-r--r-- 1 mtsouk mtsouk 78 Dec 30 21:45 CMakeLists.txt
-rw-r--r-- 1 mtsouk mtsouk 94 Dec 30 19:47 hw.c
$ mkdir build
$ cd build
$ smake ..
```

Последняя команда выводит большой объем данных, показанных на экранном снимке стр. 76. При последующем выполнении команды `smake` в корневом каталоге проекта *CMake* создастся исполняемый файл **LXF**. Если вы не хотите вручную создавать каталог **build**, можете выполнить команду `smake -H. -Bbuild` из корневого каталога проекта.

Наконец, если *smake* не сможет найти файл **CMakeLists.txt** в указанном каталоге, вы увидите следующее сообщение об ошибке:

```
$ smake /tmp
CMake Error: The source directory "/tmp" appear to contain CMakeLists.txt.
Specify --help for usage, or press the help button on the CMake GUI.
```

Поскольку приведенный здесь файл *CMake* относительно прост, далее мы не будем им заниматься. Но если вы знакомы с файлами **Makefile**, то можете бегло просмотреть созданный **Makefile**: это не лишено интереса.

В следующем разделе будет представлена конфигурация *CMake* для более реалистичного проекта, с несколькими исходными файлами на C++.

## Начинаем работу с C++

В этом проекте C++ у нас будет три исходных файла, необходимые для создания конечного исполняемого файла. Чтобы не создавать осложнений, код C++ этого проекта будет довольно прост. В проекте *CMake* также можно создать общую или статическую библиотеку — общая идея для таких проектов ничем не отличается от проектов, в которых создаются исполняемые файлы.

Для создания структуры нового проекта *CMake* под названием ‘simple’ необходимо выполнить следующие команды:

```
$ mkdir simple
$ cd simple
$ touch CMakeLists.txt
$ mkdir build
$ mkdir myLibrary
```

```
$ mkdir myLibrary/include
$ mkdir myApplication
$ touch myApplication/main.cpp
$ touch myLibrary/include/aClass.h
$ touch myLibrary/aClass.cpp
```

Первое, что вы здесь видите — хотя каталогов несколько, файл **CMakeLists.txt** только один. Кроме того, наряду с основным файлом C++ **main.cpp**, который содержит функцию `main()` проекта, есть два дополнительных файла: один исходный файл C++ (**aClass.cpp**) и один заголовочный файл (**aClass.h**). В этих двух файлах приведена реализация класса C++, который используется в **main.cpp**.

Содержимое основного файла **CMakeLists.txt**, который находится в корневом каталоге проекта, можно увидеть на экранном снимке внизу. Обратите внимание, что можно было употребить и несколько файлов **CMakeLists.txt**, но для нашего относительно простого проекта такой подход не требуется.

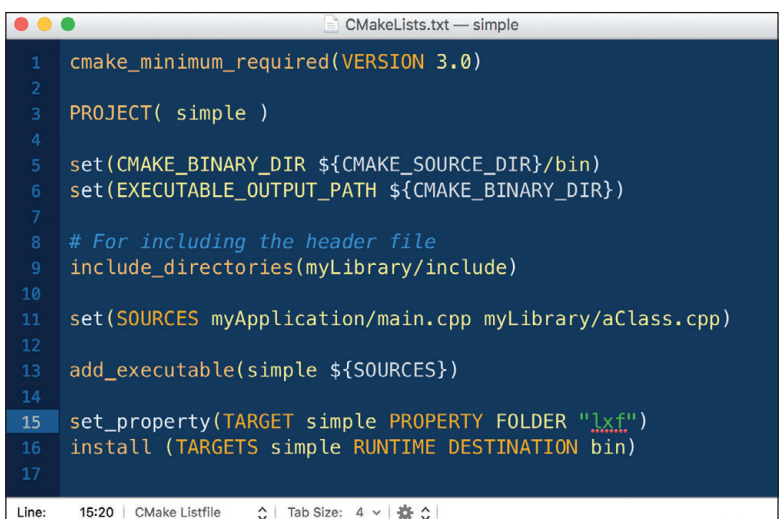
В этом файле **CMakeLists.txt** можно увидеть много новых и любопытных вещей. Во-первых, заметьте, что переменная **CMAKE\_BINARY\_DIR** в сочетании с переменной **EXECUTABLE\_OUTPUT\_PATH** указывают на каталог, где будет размещен исполняемый файл проекта.

Переменная **SOURCES** задает два исходных файла C++ проекта. Кроме того, `include_directories()` указывает расположение файла **aClass.h**. После этого `add_executable()` указывает, что результат должен получить имя **simple** и что он будет основан на значении переменной **SOURCES**.

Последняя строка в файле **CMakeLists.txt** сообщает *CMake*, что созданный двоичный файл будет размещен в подкаталоге **bin** каталога установки.

### Скорая помощь

Если вы хотите больше узнать о *Graphviz*, можете прочесть руководство по использованию *GraphViz* и руководство по языку DOT в **LXF224** или зайти на официальный сайт *Graphviz*: <http://graphviz.org>.



```
CMakeLists.txt — simple
1  cmake_minimum_required(VERSION 3.0)
2
3  PROJECT( simple )
4
5  set(CMAKE_BINARY_DIR ${CMAKE_SOURCE_DIR}/bin)
6  set(EXECUTABLE_OUTPUT_PATH ${CMAKE_BINARY_DIR})
7
8  # For including the header file
9  include_directories(myLibrary/include)
10
11 set(SOURCES myApplication/main.cpp myLibrary/aClass.cpp)
12
13 add_executable(simple ${SOURCES})
14
15 set_property(TARGET simple PROPERTY FOLDER "lxf")
16 install(TARGETS simple RUNTIME DESTINATION bin)
17
```

» Код *CMake* файла конфигурации **CMakeCache.txt** для простого проекта с исходными файлами на C++.

» Подпишитесь на печатную или электронную версии на [www.linuxformat.ru/subscribe!](http://www.linuxformat.ru/subscribe!)

Теперь пора воспользоваться **CMakeLists.txt**. Как вы уже знаете, сперва нужно выполнить команду `cmake`, а затем команду `make`:

```
$ ls -l
$ cmake -H. -Bbuild
$ cd build
$ make
```

Если в файле конфигурации *CMake* есть ошибка, вы увидите сообщения об ошибках, которые будут выглядеть примерно так:

```
CMake Error at CMakeLists.txt:7 (INSTALL):
INSTALL called with unknown mode DESTINATION
[INSTALL вызвано в неизвестном режиме DESTINATION]
```

Причиной первой ошибки является неизвестная переменная в файле **CMakeLists.txt**, которая была исправлена в вашей версии **CMakeLists.txt**:

```
CMake Warning (dev) in CMakeLists.txt:
No cmake_minimum_required command is present. A line of code
such as
```

```
[Не присутствует команда cmake_minimum_required. Строку
вида]
```

```
cmake_minimum_required(VERSION 3.10)
should be added at the top of the file. The version specified may be
lower
```

```
[надо добавить в начале файла. Данная версия м.б. старше]
if you wish to support older CMake versions for this project. For
more
```

```
[Если вы хотите поддерживать для этого проекта более старые
версии CMake]
```

```
information run "cmake --help-policy CMP0000".
```

```
[для подробной информации запустите "cmake --help-policy
CMP0000"]
```

```
This warning is for project developers. Use -Wno-dev to suppress it.
```

```
[Это сообщение для разработчиков проекта. Чтобы его убрать,
используйте -Wno-dev]
```

Второе сообщение — предупреждение о том, что надо определить минимальную версию *cmake*, необходимую для этого проекта. Как вы видите, оба сообщения понятны.

На экранном снимке внизу показан результат, который вы получите при обработке файла **CMakeLists.txt** с помощью *cmake*.

## Переменные CMake

Исполняемый файл *CMake* поддерживает несколько переменных, с помощью которых можно менять различные характеристики проекта *CMake*, не изменяя файла **CMakeLists.txt**. Самая полезная переменная — `CMAKE_INSTALL_PREFIX`, которая позволяет определить путь установки проекта. Значение по умолчанию

```
build$ make
Scanning dependencies of target simple
[ 50%] Building CXX object CMakeFiles/simple.dir/myApplication/main.cpp.o
[100%] Building CXX object CMakeFiles/simple.dir/myLibrary/aClass.cpp.o
Linking CXX executable ../bin/simple
[100%] Built target simple
build$ cd ..
simple$ ll bin/
total 12
-rwxr-xr-x 1 mtsouk mtsouk 11120 Jan 1 20:45 simple
simple$ tree -d
.
├── bin
├── build
├── CMakeFiles
│   ├── 3.0.2
│   ├── CompilerIdC
│   ├── CompilerIdCXX
│   ├── CMakeTmp
│   ├── simple.dir
│   ├── myApplication
│   └── myLibrary
├── myApplication
├── myLibrary
└── include
13 directories
```

```
3. mtsouk@malli: ~/docs/article/working/CMake.LXF234/code/simple (ssh)
build$ cmake .. --graphviz=simple.dot
-- Configuring done
-- Generating done
Generate graphviz: /home/mtsouk/docs/article/working/CMake.LXF234/code/simple/build/simple.dot
Writing /home/mtsouk/docs/article/working/CMake.LXF234/code/simple/build/simple.dot
Writing /home/mtsouk/docs/article/working/CMake.LXF234/code/simple/build/simple.dependers...
Writing /home/mtsouk/docs/article/working/CMake.LXF234/code/simple/build/simple.dependers...
-- Build files have been written to: /home/mtsouk/docs/article/working/CMake.LXF234/code/simple/build
build$ ll simple.dot*
-rw-r--r-- 1 mtsouk mtsouk 88 Jan 1 21:01 simple.dot
-rw-r--r-- 1 mtsouk mtsouk 88 Jan 1 21:01 simple.dot.simple
-rw-r--r-- 1 mtsouk mtsouk 88 Jan 1 21:01 simple.dot.simple.dependers
build$ cat simple.dot
digraph GG {
node [
fontsize = "12"
];
"node6" [ label="simple" shape="house"];
}
build$ cat simple.dot.simple
```

➤ Здесь показано, как создать файлы *Graphviz*, отображающие зависимости между объектами проекта *CMake*.

для переменной `CMAKE_INSTALL_PREFIX` на всех компьютерах UNIX — `/usr/local`. При тестировании проекта в качестве каталога установки можно выбрать `/tmp`. Переменная `CMAKE_BUILD_TYPE` указывает создаваемый тип сборки. Допустимые значения переменной `CMAKE_BUILD_TYPE` — `Debug`, `Release`, `RelWithDebInfo` и `MinSizeRel`. При указании значения `Debug` в создаваемые файлы будут включаться флаги отладки. А при создании окончательной версии проекта для `CMAKE_BUILD_TYPE` надо указать значение `Release`.

Наконец, есть переменная `CMAKE_LANG_FLAGS`. Замените `<LANG>` на строку, связанную с выбранным языком программирования — эта строка используется для определения флагов компилятора. Для C значение этой переменной должно быть равно `C`, а для C++ — `CXX`. Так что для проекта C++ эта переменная должна иметь значение `CMAKE_CXX_FLAGS`.

Полный список доступных переменных можно найти на <https://cmake.org/cmake/help/v3.0/manual/cmakevariables.7.html>.

## Компиляция и установка

В этом разделе вы узнаете, как скомпилировать и установить простой проект *CMake*, используя команду `make`.

Чтобы скомпилировать проект после успешного выполнения команды `cmake`, надо выполнить команду `make` из каталога сборки проекта C++. Если вы хотите скомпилировать проект с включенными флагами отладки, используйте в командной строке переменную `CMAKE_BUILD_TYPE`, как показано ниже:

```
$ rm -rf build
$ make build
$ cd build
$ cmake .. -DCMAKE_BUILD_TYPE=Debug
$ make
```

Команда `objdump(1)` проверяет, что созданный исполняемый файл содержит информацию об отладке:

```
$ objdump --syms bin/simple | grep debug
0000000000000000 | d .debug_aranges 0000000000000000
.debug_aranges
0000000000000000 | d .debug_info 0000000000000000
.debug_info
0000000000000000 | d .debug_abbrev 0000000000000000
.debug_abbrev
0000000000000000 | d .debug_line 0000000000000000
.debug_line
0000000000000000 | d .debug_str
0000000000000000 .debug_str
```

### Скорая помощь

Главное преимущество *CMake* по сравнению с *GNU make* заключается в том, что *CMake* — кроссплатформенная утилита. Кроме того, файлы конфигурации *CMake* проще и короче, чем файлы конфигурации аналогичного проекта *Make*. Для простых проектов годятся обе утилиты, а для более сложных лучше подойдет *CMake*.

➤ Результат выполнения команды `cmake` в корневом каталоге проекта `simple` и `make` в каталоге `build` проекта.

➤ Пропустили номер? Узнайте на с. 108, как получить его прямо сейчас.



Чтобы установить двоичный файл проекта **simple** в подкаталог **bin** каталога по умолчанию (**/usr/local**), скомандуйте `make install`. Но поскольку для размещения файлов в каталоге **/usr/local** необходимы права **root**, скорее всего при выполнении команды `make install` появится сообщение об ошибке. Чтобы оно не появлялось, выполните команду как `sudo make install`.

Если вы хотите установить проект в другом каталоге, на который у вас есть права (**/tmp**), используйте переменную **CMAKE\_INSTALL\_PREFIX** и выполните команду `cmake` следующим образом:

```
$ cd build
$ cmake .. -DCMAKE_INSTALL_PREFIX=/tmp
```

Затем вы можете использовать команду `make install` для установки исполняемого файла в каталог **/tmp**. Чтобы убедиться в наличии файла, просмотрите содержимое каталога **/tmp**:

```
$ make install
[100%] Built target simple
Install the project...
-- Install configuration: ""
-- Up-to-date: /tmp/bin/simple
$ ls -l /tmp/bin/
total 12
-rwxr-xr-x 1 mtsouk mtsouk 11120 Jan 1 20:57 simple
```

Желая проверить результаты команды `make`, можете выполнить `make` с параметром **-n**. При указании этого параметра команды просто выводятся на экран, но не выполняются!

## CTest и CPack

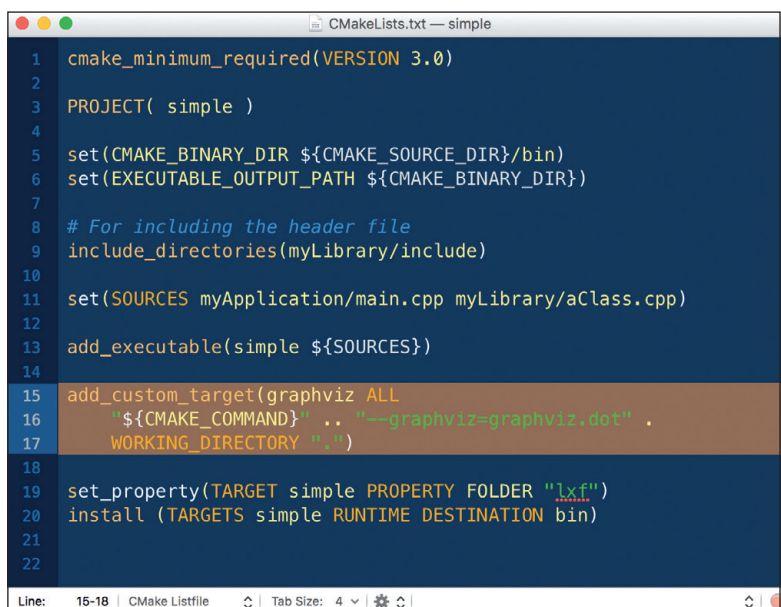
**CTest** — это инструмент для тестирования, который входит в состав **CMake**. Его можно вызвать командой `ctest`. Утилита **CTest** умеет работать в двух режимах. В первом режиме вы используете **CMake**, чтобы создать и запустить тесты для **CTest**. Во втором режиме **CTest** берет управление на себя и запускается как скрипт для запуска процесса тестирования.

**CPack** — кроссплатформенный инструмент для пакетной упаковки, который входит в состав **CMake**. **CPack** использует файл **CPackConfig.cmake**, и для его вызова используется команда `cpack`. Так как **CMake** поставляется вместе с модулем **CPack**, вы можете использовать **CPack** из файла конфигурации **CMake**, включив строку `INCLUDE(CPack)` в файл **CMakeCache.txt**. Наконец, можно отлаживать **CPack** с помощью команды `cpack`, следующим образом: `cpack --debug --verbose`.

## CMake и Graphviz

**Graphviz** — это набор инструментов для создания впечатляющих и удобных ориентированных или неориентированных графов. **Graphviz** использует собственный язык под названием **Dot**. **CMake** может генерировать код **Dot**, который визуализирует зависимости целей проекта **CMake**, с помощью параметра `--graphviz`.

Для ранее созданного проекта **simple** визуализация зависимостей создается следующим образом:



```
1 cmake_minimum_required(VERSION 3.0)
2
3 PROJECT( simple )
4
5 set(CMAKE_BINARY_DIR ${CMAKE_SOURCE_DIR}/bin)
6 set(EXECUTABLE_OUTPUT_PATH ${CMAKE_BINARY_DIR})
7
8 # For including the header file
9 include_directories(myLibrary/include)
10
11 set(SOURCES myApplication/main.cpp myLibrary/aClass.cpp)
12
13 add_executable(simple ${SOURCES})
14
15 add_custom_target(graphviz ALL
16   "${CMAKE_COMMAND}" .. "--graphviz-graphviz.dot" .
17   WORKING_DIRECTORY ".")
18
19 set_property(TARGET simple PROPERTY FOLDER "LXF")
20 install (TARGETS simple RUNTIME DESTINATION bin)
21
22
```

```
$ cd build
$ cmake .. --graphviz=simple.dot
```

На экранном снимке вверху стр. 78 показан результат работы предыдущей команды, а также три сгенерированных файла **dot**, которые довольно просты, поскольку проект **CMake** тоже прост. В файле **simple.dot** показаны все зависимости проекта, а в файле **simple.dot.simple** — зависимости для конкретной цели. И если у вас всего одна цель, оба файла будут одинаковыми! Наконец, в файле **simple.dot.simple.dependers** показаны цели, зависящие от указанной цели. Поскольку наш проект **CMake** имеет всего одну цель, **simple.dot.simple.dependers**, этот файл совпадает с файлом **simple.dot.simple**. В более интересных проектах будут создаваться более сложные файлы **Graphviz** и, следовательно, получится более интересный графический результат. Вы можете создать PNG-файл **simple.png** из файла **dot** с именем **simple.dot**, выполнив команду `dot -Tpng -osimple.png simple.dot`.

Полезно тут в том, что добавив параметры в файл **CMakeCache.txt**, вы можете заставить **CMake** автоматически создать зависимости проекта с помощью **Graphviz**. На экранном снимке вверху показана окончательная версия файла **CMakeCache.txt**.

Файл **CMakeGraphVizOptions.cmake** связан с выводом утилиты **Graphviz** **CMake** и позволяет настраивать вывод утилиты **Graphviz**. Полный список (он довольно велик) поддерживаемых параметров **CMakeGraphVizOptions.cmake** можно получить по команде `cmake --help-module CMakeGraphVizOptions`.

Помните, что оценить преимущества **CMake** можно только в том случае, если вы регулярно используете его в своих проектах. Так чего же вы ждете? **LXF**

➤ Окончательная версия файла конфигурации **CMakeCache.txt**, с которым **CMake** автоматически создает зависимости проекта **CMake** с помощью **Graphviz**. Автоматизация задач — это всегда хорошо!

## Про GNU Make

**Make** — это мощный инструмент автоматизации сборки, который был создан Стюартом Фельдманом [Stuart Feldman] из Bell Labs еще в апреле 1976 г. **GNU Make** — стандартная реализация **make** для Linux и Mac OS X со многими улучшениями, необходимыми для компиляции ядра Linux.

При использовании **GNU Make** надо указать, какие инструменты будут использоваться для компиляции исходных файлов. К ним относится компилятор используемого языка программирования.

**Makefiles** — это файлы конфигурации **make**. Они помогают организовывать и группировать различные команды и выполнять их одновременно. Если файл **Makefile** составлен правильно, вам надо запомнить и выполнить всего несколько команд **make**. Файл конфигурации **make** по умолчанию можно назвать **Makefile** или **makefile**; лучше использовать **Makefile**, так как он находится ближе к началу списка каталогов! После добавления целей, правил, зависимостей и переменных файл

**Makefile** может изрядно усложниться, как и обычный файл, содержащий исходный код. Обратите внимание, что **GNU make** обрабатывает разные типы пропусков места по-разному. Так, символ табуляции отличается от четырех или восьми последовательных пробелов. Это важно, потому что каждая строка файла **makefile** с командой начинается с символа табуляции.

Дополнительную информацию о **GNU make** можно получить на [www.gnu.org/software/make](http://www.gnu.org/software/make).





## Леность с IDLE

Для создания скрипта Python бывает достаточно создать файл с расширением `.py` в вашем любимом текстовом редакторе и добавить в него код. Однако когда дело доходит до подсчета отступов и когда вы хотите внести изменения и быстро запустить скрипт, это не слишком удобно.

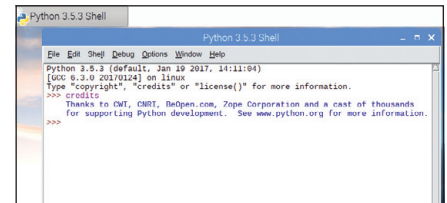
**IDLE** [`idle` — *англ.* праздный] — интегрированная среда для разработки и обучения Python. **IDLE** включает многооконный текстовый редактор с функциями, которые превосходят функции обычных программ обработки текста: это интеллектуальные отступы, подсказки по вызовом и автодополнение распространенных терминов. Сохранив скрипт, вы можете запустить его, нажав `F5` в окне оболочки **IDLE**. Это идеально подходит для интерактивных

скриптов, так как в окне автоматически выделяются цветом входные данные, выводимые данные и сообщения об ошибках.

**IDLE** — кроссплатформенное приложение. Если у вас есть Raspberry Pi, она будет предустановлена в разделе Programming [Программирование] меню Applications [Приложения] — для нашего урока выберите Python 3 (**IDLE**). **IDLE** также можно установить из репозитория Ubuntu. Просто откройте терминал и скомаундите `sudo apt-get update` и `sudo apt-get install idle3`.

На сайте Python имеется обширное руководство пользователя по **IDLE**. Дополнительную информацию см. на <https://docs.python.org/3/library/idle.html>. Для целей нашего урока помните, что после

запуска скрипта вы можете продолжать взаимодействовать с ним, вводя команды в окно оболочки, например, чтобы добавить элемент в массив.



➤ **IDLE** предустановлена для Python 2 и 3 в Raspbian. Среда доступна в репозиториях большинства дистрибутивов Linux.

```
import secrets
from array import *
myarray=array('i',[1,2,3,4,5,6])
roll = secrets.choice(range(0,len(myarray)))
print (myarray[roll])
```

Значение `roll` — это минимальное значение `myarray index (0)`, а максимальная длина — `len(arrayname)`. Вы можете использовать эти методы, чтобы получить количество элементов в стандартном массиве Python.

## Манипулируем массивами

Пока что мы научились создавать элементы массива и выводить их на экран. Но вы также можете манипулировать ими во время работы программы. Вы можете задаться вопросом, зачем это нужно — ведь можно просто остановить скрипт и изменить сам код. Однако это удобно для отслеживания обработанных данных и для добавления новой информации. Например, если у вас есть виртуальное казино (см. далее), вы можете захотеть добавить новых игроков и выигрыши, а также вести запись об уже разыгранных картах.

Чтобы вставить значение в определенную позицию в массиве, используйте метод `insert`. Например, метод `myarray.insert(0,1)` установит значение первого элемента массива в 1. При этом существующие данные не удаляются, потому что другие элементы массива просто будут перемещены соответствующим образом.

Желая просто добавить новые элементы в конец массива, вы можете использовать метод `append`, например:

```
myarray.append(7)

newvalues=array('i',[8,9,10])
myarray.extend(newvalues)
```

## Вызов NumPy

Хотя стандартный модуль `array` Python — отличный способ познакомиться с массивами, в более сложных сценариях он не всегда подходит. Например, нельзя быстро и легко сохранять значения или создавать многомерный массив (подробнее об этом позже).

К счастью, NumPy (числовой Python) был специально разработан для научных расчетов и содержит инструменты, позволяющие создавать очень эффективные и мощные массивы. Чтобы приступить к работе, убедитесь, что у вас есть инсталлятор *Pip*. Чтобы его установить, откройте терминал и выполните команду

```
$ sudo apt-get install python3-pip
Затем скомаундите
$ sudo pip3 install numpy
```

Соглашение, применяемое для вызова модуля NumPy при написании сценария — `import numpy as np`. После этого можно объявить массив NumPy следующим образом: `np.array`. Одним из самых явных преимуществ использования NumPy для создания массива является то, что он умеет автоматически определять тип данных, добавляемых в массив (например, строки или целые числа), что позволяет хранить оба типа данных в одном массиве. В соответствии с созданным ранее электронным кубиком, создайте пустой файл `magic8.py` в своем любимом редакторе и вставьте в него следующий код:

```
import secrets
import numpy as np
```

```
[GCC 7.2.0] on linux
Type "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: /home/lubuntu/magic8.py =====
You may rely on it.
>>> np.insert(myarray,3,'Absolutely!')
array(['It is certain.', 'It is decidedly so', 'Without a doubt.', 'Absolutely!', 'Yes definitely.', 'You may rely on it.', 'As I see it, yes', 'Most likely', 'Outlook good', 'Yes', 'Signs point to yes', 'Reply hazy, try again.', 'Ask again later.', 'Better not tell you now', 'Cannot predict now.', 'Concentrate and ask again.', 'Do not count on it.', 'My reply is no.', 'My sources say no.', 'Outlook not so good.', 'Very doubtful.'],
      dtype='<U26')
>>> np.append(myarray,'No')
array(['It is certain.', 'It is decidedly so', 'Without a doubt.', 'Yes definitely.', 'You may rely on it.', 'As I see it, yes', 'Most likely', 'Outlook good', 'Yes', 'Signs point to yes', 'Reply hazy, try again.', 'Ask again later.', 'Better not tell you now', 'Cannot predict now.', 'Concentrate and ask again.', 'Do not count on it.', 'My reply is no.', 'My sources say no.', 'Outlook not so good.', 'Very doubtful.', 'No'],
      dtype='<U26')
>>>
```

➤ `numpy.insert` поместит элемент в определенную позицию в вашем массиве. Если же вы просто хотите добавить элемент в конец массива, используйте метод `numpy.append`.

**Скорая помощь**

Выполнение описанных здесь команд сделает ваш кубик с шестью гранями десятигранным. Полный пример этого скрипта (`1d10.py`) доступен по ссылке <http://bit.ly/10-sided-dice>.

➤ Подпишитесь на печатную или электронную версии на [www.linuxformat.ru/subscribe!](http://www.linuxformat.ru/subscribe!)



```
myarray=np.array(['Это точно.', 'Это явно так', 'Без сомнения.', 'Да, безусловно.', 'Можете на это положиться.', 'Я так вижу, что да.', 'Скорее всего.', 'Хороший прогноз.', 'Да.', 'Знаки указывают на Да.', 'Ответ туманный, повторите попытку.', 'Снова спросите again позже.', 'Пока лучше вам не рассказывать.', 'Сейчас предсказать нельзя.', 'Сконцентрируйтесь и спросите еще раз.', 'Не рассчитывайте на это.', 'Мой ответ - нет.', 'Мои источники говорят Нет.', 'Прогноз не очень хорош.', 'Очень сомнительно.'])
answer = secrets.choice(range(0,myarray.size))
print (myarray[answer])
```

Этот скрипт также доступен на <https://github.com/nate-drake/arrayexamples/blob/master/magic8.py>.

Наша маленькая пятистрочная «жемчужина» создает массив NumPy с текстовыми значениями, а именно 20 возможных ответов, который давал пользователям классический шар предсказаний. Учтите, что для определения общей длины массива можно использовать `myarray.size`, а не более громоздкую функцию `len`.

## «Ответ туманный, повторите попытку»

Если не считать использования текстовых значений и более эффективного типа массива, основная часть скрипта точно такая же, как у предыдущего: вывод значений со случайным индексом. Стандартный шар для предсказаний содержит 10 положительных

ответов, пять отрицательных ответов и пять уклончивых ответов. Желая сделать шар чуть справедливее, добавьте еще один отрицательный ответ, через функцию `numpy.append`, например:

```
np.append(myarray, 'Нет')
```

Если делать это внутри *IDLE*, программа повторно создаст новый список всех элементов массива автоматически, включая тот, который вы только что добавили.

Для этого скрипта элементы выбираются случайным образом, так что их порядок не имеет значения. Если, однако, вы хотите добавить значение в конкретную позицию массива NumPy, используйте функцию `numpy.insert`, например:

```
np.insert(myarray,3,'Абсолютно!')
```

Эта функция поместит слово 'Абсолютно!' в четвертую позицию списка (не забывайте, что элементы отсчитываются от нуля).

## 2D или не 2D

Хотя NumPy удобен для создания одномерных массивов, например, списка значений, в нем также можно создать многомерную таблицу значений. Эта идея проста, однако в большей части онлайн-документации, связанной с NumPy, ее пытаются объяснить намного сложнее. Поэтому посвятим некоторое время тому, чтобы объяснить ее здесь.

У простейшего типа многомерного массива есть два измерения, называемых осями. Хороший пример применения — создание

➤ Здесь у массива `deck` (англ. колода карт) есть две оси. Оба элемента [2,11] и [3,11] содержат строку Queen (Дама), а масть карты определяется по ее положению в массиве.

```
===== RESTART: /home/lubuntu/drawcard.py =====
Your card is the 2 of Hearts.
>>> print(deck)
[['Ace' '2' '3' '4' '5' '6' '7' '8' '9' '10' 'Jack' 'Queen' 'King']
 ['Ace' '2' '3' '4' '5' '6' '7' '8' '9' '10' 'Jack' 'Queen' 'King']
 ['Ace' '2' '3' '4' '5' '6' '7' '8' '9' '10' 'Jack' 'Queen' 'King']
 ['Ace' '2' '3' '4' '5' '6' '7' '8' '9' '10' 'Jack' 'Queen' 'King']]
>>> print(deck[2,11])
Queen
>>> deck[2,11] = 10
>>> print (deck)
[['Ace' '2' '3' '4' '5' '6' '7' '8' '9' '10' 'Jack' 'Queen' 'King']
 ['Ace' '2' '3' '4' '5' '6' '7' '8' '9' '10' 'Jack' 'Queen' 'King']
 ['Ace' '2' '3' '4' '5' '6' '7' '8' '9' '10' 'Jack' '10' 'King']
 ['Ace' '2' '3' '4' '5' '6' '7' '8' '9' '10' 'Jack' 'Queen' 'King']]
>>>
```

## Создание многомерных массивов

Если двумерный массив можно рассматривать как электронную таблицу, то трехмерный массив похож на куб. NumPy идеально подходит для создания таких N-мерных массивов с помощью `ndarray`. Как и свидетельствует название, никаких теоретических ограничений на число измерений массива нет. Но пока давайте представим, что мы хотим создать виртуальную игру в блэкджек, в которой используются пять колод карт. Если вы уже выполнили указанные действия, то создали массив вида 4,13. Другими словами, каждую карту можно найти по масти и номеру. Чтобы начать работу с созданием трехмерного массива, который будет хранить информацию о том, сыграла ли та или иная карта,

можно использовать некоторые из процедур создания массивов NumPy. Например, запустите

```
import numpy as np
decks = np.zeros((5,4,13), dtype=np.int)
```

Создастся новый массив, и в нем все элементы будут иметь значение 0. Параметр `dtype` определяет тип данных, которые должны храниться в массиве. По умолчанию это `float64`. Если вы захотите добавить единицу вместо нуля, замените нули на единицы. Вы также можете создать массив, заполненный заданными вами значениями. Полный список процедур создания массива см. на сайте <https://docs.scipy.org/doc/numpy/user/basics.creation.html>. Создав многомерный массив, вы сможете работать

с ним точно так же, как и с другими массивами NumPy, например, `decks[1,3,10] = 99`.

```
>>> print(decks)
[[[0 0 0 0 0 0 0 0 0 0 0 0 0]
 [0 0 0 0 0 0 0 0 0 0 0 0 0]
 [0 0 0 0 0 0 0 0 0 0 0 0 0]
 [0 0 0 0 0 0 0 0 0 0 0 0 0]]
 [[0 0 0 0 0 0 0 0 0 0 0 0 0]
 [0 0 0 0 0 0 0 0 0 0 0 0 0]
 [0 0 0 0 0 0 0 0 0 0 0 0 0]
 [0 0 0 0 0 0 0 0 0 0 0 0 0]]
 [[0 0 0 0 0 0 0 0 0 0 0 0 0]
 [0 0 0 0 0 0 0 0 0 0 0 0 0]
 [0 0 0 0 0 0 0 0 0 0 0 0 0]
 [0 0 0 0 0 0 0 0 0 0 0 0 0]]
 [[0 0 0 0 0 0 0 0 0 0 0 0 0]
 [0 0 0 0 0 0 0 0 0 0 0 0 0]
 [0 0 0 0 0 0 0 0 0 0 0 0 0]
 [0 0 0 0 0 0 0 0 0 0 0 0 0]]]
>>> decks[1,3,10] = 99
>>>
```

➤ Используйте `numpy.zeros`, и вы сможете создать свой многомерный массив.

➤ Пропустили номер? Узнайте на с. 108, как получить его прямо сейчас.



двумерного массива для хранения координат X и Y на графике, и, действительно, NumPy предназначен специально для хранения именно таких сложных научных данных.

Более простой способ понять двумерные массивы — представить их как строки и столбцы в электронной таблице. Например, если у вас есть колода карт, вы можете разместить их в виде четырех рядов (представляющих трефы, бубны, червы и пики) и 13 столбцов, представляющих карты каждой масти. Любая карта в этом двумерном массиве имеет уникальный индекс (или, в терминологии многомерных массивов, кортеж). Общий кортеж многомерного массива называется его формой. В приведенном выше примере это будет 4,13.

Для начала надо создать новый файл с именем **drawcard.py** и вставить в него следующий текст. В результате мы получим простой скрипт, использующий двумерный массив для случайного выбора из колоды карт:

```
import secrets
import numpy as np
# Применяется порядок мастей как в бридже: Трефы (0), Бубны (1), Червы (2), Пики (3).
deck=np.array(['Туз',2,3,4,5,6,7,8,9,10,'Валет','Дама','Король',
'Туз',2,3,4,5,6,7,8,9,10,'Валет','Дама','Король','Туз',2,3,4,5,6,7,8,9,
10,'Валет','Дама','Король','Туз',2,3,4,5,6,7,8,9,10,'Валет','Дама',
'Король'])
randomsuit=secrets.choice(range(deck.shape[0]))
suit = ['Трефы', 'Бубны', 'Червы', 'Пики']
randomcard=secrets.choice(range(deck.shape[1]))
yourcard=deck[randomsuit,randomcard]
print('Ваша карта - ' + yourcard + ', ' + suit[randomsuit] + '')
```

## Визуализируем массив

Если вы используете *IDE*, то после запуска скрипта введите команду `print(deck)`, чтобы вывести двумерный массив на экран в виде текста. Вы увидите по 13 карт для каждой масти в виде строк и сможете визуализировать массив. Переменные `deck.shape[0]` и `deck.shape[1]` представляют количество элементов для каждого измерения (соответственно 4 и 13). Они используются для случайного выбора масти карты (например, трефы) и самой карты (например, валет). Обратите внимание, что в массиве NumPy легко хранятся как числовые элементы, например, 9, так и строковые элементы, например, 'Король'.

Индекс (или кортеж) для масти является целочисленным значением, поэтому для сопоставления этого значения с названием масти (трефы, бубны и т. д.) используется список `suit` [англ. масть]. Это не обязательно для работы массива, но позволяет вывести название масти в виде текста.

Любую карту в этой колоде можно найти, пользуясь конструкцией `deck[X,Y]`. Ее удобно использовать с переменной `yourcard`, которая выделяет выбранную карту в колоде на основе случайных значений. Ее также можно использовать для отображения конкретной карты, например, `print(deck[2,11])`. Если вы хотите изменить значения элементов массива, чтобы они представляли количество очков в блэджеке, можете просто объявить переменную, например, таким образом: `deck[2,11]=10`.

## Подробнее о NumPy

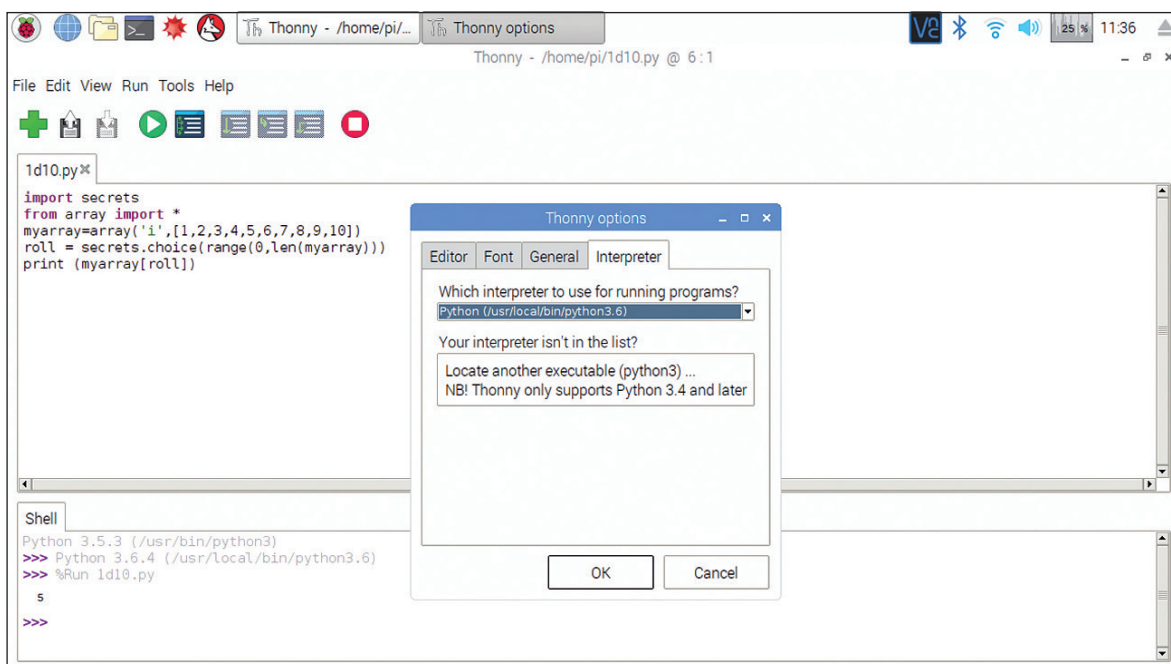
Поскольку наш урок задуман как базовое введение в массивы, в нем практически нет ничего, что могло бы пойти не так. Пользователи Raspberry Pi, которые захотят запустить скрипты, могут натолкнуться на некоторые трудности, поскольку модуль 'secrets', используемый для генерации случайных чисел, работает только с Python 3.6.

Если вы можете скомпилировать программное обеспечение сами, загрузите последнюю стабильную версию Python с сайта [www.python.org/ftp/python](http://www.python.org/ftp/python) (сейчас 3.6.4) и установите ее через терминал Pi. Используйте команду `$ make altinstall`, чтобы сохранить существующую версию Python. Также можете переработать скрипты, чтобы в них использовался собственный генератор (менее случайных чисел NumPy вместо генератора из модуля secrets (<https://docs.scipy.org/doc/numpy/reference/routines.random.html>)).

Излишне говорить, что NumPy способен на большее, чем игровые кубики и колоды карт. Если вы хотите хранить и обрабатывать другие типы данных в массивах, уделите некоторое время на чтение руководства по NumPy (<https://docs.scipy.org/doc/numpy/>). Оно содержит сведения о более сложных способах управления данными, таких как порядок байтов и подклассификация, которые не были рассмотрены в этой статье. По умолчанию во время выполнения массивы доступны только для чтения, поэтому выполняемые вами изменения не сохраняются между сеансами. Если вы хотите сохранить массив в файле, используйте `numpy.save` (подробности см. на <https://docs.scipy.org/doc/numpy/reference/generated/numpy.save.html>). **LXF**

**Скорая помощь**

Более удобные команды NumPy доступны в руководстве по DataCamp, которое можно загрузить с <http://bit.ly/numpy-pdf>.



**» В Raspbian можно использовать IDE Thonny с Python 3.6. Установив новую версию Python, откройте Tools > Options > Interpreter [Инструменты > Параметры > Интерпретатор] и выберите Thonny IDE в раскрывающемся меню.**





Мощь шифрования Enigma заключалась в использовании нескольких роторов. Каждое нажатие ключа приводило как минимум к повороту одного из роторов, что обеспечивало совершенно новый алфавит шифрования. Т.е. устройство могло пройти через тысячи нажатий на ключ перед шифрованием слова одинаковым образом. Тем не менее, с учетом количества сообщений, отправленных во время войны, наличие 17576 возможных настроек роторов не обеспечивало достаточную надежность, вследствие чего нацисты добавили коммутационную панель. Сюда входило сопоставление шести пар букв — например, E могла быть заменена G. Это увеличивало число возможных начальных параметров Enigma до более 100 миллиардов. Прибавим сюда тот факт, что роторы можно было размещать в любом порядке, и количество возможных параметров устройства превысит количество атомов во Вселенной.

Чтобы зашифровать или расшифровать сообщения, операторам надо было только проверить, чтобы их изначальные роторы *Ringstellung* и *Stecker* и настройки коммутационной панели были такими же, как в соответствующих шифровальных книгах.

## Волшебство Purple

Какой бы грандиозной ни казалась Enigma на бумаге, в реальной жизни у нее был ряд фундаментальных недостатков, которые, вместе взятые, помогли исследователям узнать ее секреты. Например, подключение для роторов хотя и отличалось для каждого, всё же было в основном одинаковым в разных устройствах. Кроме того, немцы не ставили ротор в одно и то же место два дня подряд, что значительно облегчало взлом кода, если вам было известны параметры для предыдущего дня. Кроме того, буква никогда не шифровалась как она сама, и если, например, вы предполагали, что где-то в зашифрованном Enigma сообщении было слово ENGLAND (Англия), вы могли исключить любой текст, содержащий эти буквы.

Purple Machine отказалась от роторов в пользу шаговых коммутаторов — именуемых также Uniselectors. Для тех, кто не в курсе, шаговый коммутатор — это электромеханическое устройство для переключения входящего сигнала на один из возможных выходов. Изначально они разрабатывались для автоматических телефонных станций.

Как именно были созданы устройства-оригиналы, уверенно утверждать невозможно, потому что Уильям Фридман [William Friedman] и его коллеги из SIS были вынуждены создавать собственные реплики Purple с нуля, на основе перехваченных сообщений, которые они взломали.

В аналоге Purple, разработанном SIS, формировалась электрическая цепь между электрической печатной машинкой, электромагнитом и переключателем при каждом нажатии ключа. Переключатель перемещался вперед в определенной последовательности (см. ниже). Когда положение переключателя достигало 25, он возвращался в положение 1. При правильной настройке это имело результатом 25 совершенно несвязанных алфавитов шифра, что значительно усовершенствовало роторы, использованные в Enigma.

Шесть букв алфавита, весьма изобретательно названные «шестеркой» — например, AEIOUY, шифровались с помощью одного шагового коммутатора. Они продвигались на один шаг вперед каждый раз, когда нажималась одна из них, и это означает, что каждый раз выбирался новый случайный алфавит.

Оставшиеся буквы алфавита, известные как «двадцатка», например, BCDFGHJKLMNPQRSTVWXZ, шифровались с использованием трех каскадных шаговых коммутаторов. В этом и заключалась суть изобретательности Purple. Один из этих шаговых коммутаторов, определенный оператором как «быстрый [fast]»,

сдвигался на одну позицию вперед при каждом нажатии на ключ, в результате чего получался новый алфавит шифра, как и с переключателем «шестерки».

Более глубокое объяснение работы шаговых коммутаторов вы найдете во врезке «Шаговые коммутаторы» (на стр. 86).

## Подготовим Purple

Сообщения Purple обычно отправляются на Ромадзи — способе записи японских слов латинским алфавитом. Можете спокойно записать короткое сообщение, которое хотите зашифровать, например, "MEET ON THE BEACH AT MIDNIGHT [ВСТРЕЧА НА ПЛЯЖЕ В ПОЛНОЧЬ]".

Далее надо выбрать параметры вашей коммутационной панели. Самый простой способ это сделать — записать все 26 букв алфавита в любой последовательности. В качестве альтернативы можно перетасовать буквы алфавита по команде терминала:

```
echo 'ABCDEFGHIJKLMNOPQRSTUVWXYZ' | sed 's/./&\n/g' | shuf | tr -d "\n"
```

Первые шесть букв будут вашей шестеркой, а оставшиеся — двадцаткой.

Затем вам надо определиться с начальным положением переключателя шестерки и тремя двадцатки. Это может быть любое значение между 1 и 25. Например, вы можете решить, что начальное положение для переключателя шестерки будет 9, а три переключателя двадцатки будут в положениях 1, 24 и 6 соответственно.

Наконец, надо решить, какой из переключателей двадцатки будет быстрым, какой — средним и какой — медленным. В SIS был блистательно эффективный способ записи посредством присвоения чисел от 1 до 3 переключателям от быстрого до медленного. Например, если вы решили, что первый переключатель «двадцатки» будет средним, а второй быстрым, это будет записано как "213".

Вы можете еще раз использовать в Терминале удобную команду `shuf`, чтобы создать случайные начальные положения всех четырех переключателей, а также порядок быстрого, среднего и медленного переключателей Двадцатки с помощью:

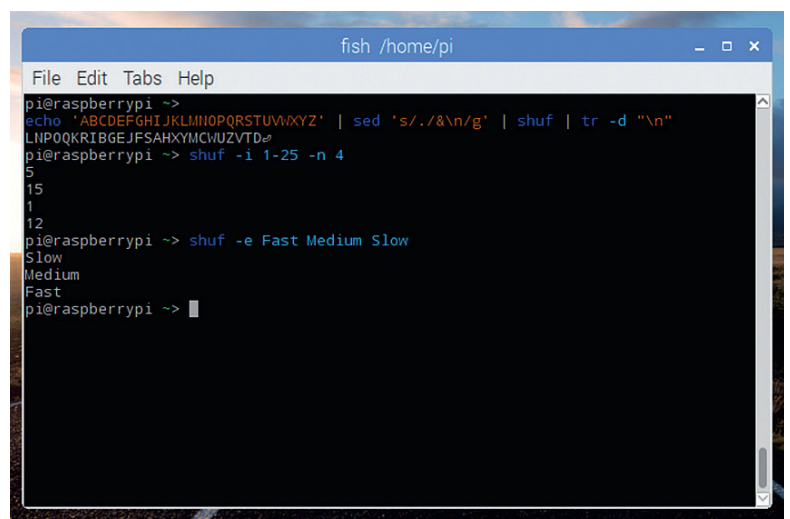
```
shuf -i 1-25 -n 4 && shuf -e Fast Medium Slow
```

## Симулятор Purple

Для продолжения работы в этом проекте сначала убедитесь, что операционная система на вашем ПК не устарела, запустив `sudo apt-get update`, а затем `sudo apt-get upgrade`. Сам Python 3 уже

### Скорая помощь

Если вам нужна помощь в создании действительно случайной последовательности букв, загляните на [www.random.org/strings](http://www.random.org/strings). Проверьте, чтобы буквы не повторялись.



```
fish /home/pi
File Edit Tabs Help
pi@raspberrypi ~->
echo 'ABCDEFGHIJKLMNOPQRSTUVWXYZ' | sed 's/./&\n/g' | shuf | tr -d "\n"
LNPQOKRIBGFEJFSAHXYMCWUZVTD
pi@raspberrypi ~-> shuf -i 1-25 -n 4
5
15
1
12
pi@raspberrypi ~-> shuf -e Fast Medium Slow
Slow
Medium
Fast
pi@raspberrypi ~-> |
```

Используйте команду `shuf` в Терминале, чтобы быстро сгенерировать параметры коммутационной панели, а также начальные положения и порядок шаговых коммутаторов.

» Подпишитесь на печатную или электронную версии на [www.linuxformat.ru/subscribe!](http://www.linuxformat.ru/subscribe!)

предустановлен в большинстве дистрибутивов Linux. Если нет, загляните на <https://docs.python.org/3/using/unix.html>, где вы найдете помощь в скачивании и компиляции. Далее запустите `sudo apt-get install python3-pip` для установки Python Package Index. Это самый простой способ скачать и запустить Purple Simulator.

Чтобы этот проект не мешал другим проектам Python на вашем компьютере, мы также создадим виртуальную среду Python для симулятора Purple. Запустите `apt-get install python3-venv`, чтобы скачать модуль `venv`.

Когда загрузка будет завершена, создайте виртуальную среду или симулятор Purple командой `sudo python3 -m purple_venv`. Переключитесь на эту новую виртуальную среду по команде `source purple_venv/bin/activate`. Запомните эту команду, потому что вам придется запускать ее каждый раз, когда вы захотите загрузить Purple Simulator.

Обеспечьте наличие у вас самой свежей версии `pip`, запустив `pip install --upgrade pip`. По завершении обновления вы сможете установить сам симулятор, запустив `pip install purple`.

## Шифруем свое первое сообщение

Введите `purple`, чтобы просмотреть обязательные и опциональные аргументы. Это довольно простая процедура; единственное препятствие, с которым вы можете столкнуться, связано с параметрами шаговых коммутаторов, которые соответствуют нормам, установленным SIS. В примере выше параметры коммутатора записывались в следующем формате:

```
9-1,24,6-23
```

Первая цифра (9) обозначает положение коммутатора шестерки, а три последующих цифры (1, 24 и 6) представляют начальные положения коммутаторов двадцатки. Две последних цифры (23) представляют положения среднего и быстро шаговых коммутаторов соответственно.

Настройки алфавита коммутационной панели записываются в виде строки символов, причем первые шесть символов в этой строке связаны с шестерками, а оставшиеся — с двадцатками. Например, `OMHFDCEZBPUNQLIAGVRJWSKYX`. Для получения наилучшего результата убедитесь, что имеется ровно 26 букв без повторений, как в этом пример.

Чтобы симулятор шифровального устройства был точен с исторической точки зрения, по умолчанию он не принимает знаки пунктуации и пробелов (для их передачи в сообщениях японцы

используют трехбуквенные коды). На современных клавиатурах такой ввод может быть довольно непросто, поэтому используйте флаг `-f` (filter), чтобы немного ослабить эти ограничения, например: `purple -e -t "MEET ON THE BEACH AT MIDNIGHT" -s 9-1,24,6-23 -a OMHFDCEZBPUNQLIAGVRJWSKYX -f`

Для более длинных сообщений вы, возможно, решите сначала поместить свой текст в файл. Для этого надо использовать флаг `-i` (input), например:

```
purple -e -i message1.txt -s 9-1,24,6-23 -a OMHFDCEZBPUNQLIAGVRJWSKYX -f
```

В соответствии с проверенными временем нормами криптографии, результирующий текст отображается блоками из пяти букв. Не поддавайтесь соблазну попытаться как-то упорядочить текстовое сообщение, потому что для дешифрования сообщений не имеется флага `-filter`.

Как только ваш корреспондент получит ваше зашифрованное сообщение, предложите ему использовать заранее согласованные параметры для дешифровки сообщения с помощью флага `-d` (decrypt), например:

```
purple -d -t "HBILM JYHVA UBOMY WMYNI AJDL" -s 9-1,24,6-23 -a OMHFDCEZBPUNQLIAGVRJWSKYX
```

## Взлом Purple

Кодовое название перехваченного трафика Purple было Magic, и это весьма подходящее название, учитывая те блестящие усилия PPTP США, которые приходилось приложить, чтобы взломать код. Когда команда Алана Тьюринга [Alan Turing] в Блетчли-парк начала взлом Enigma, у них были копии коммерческой версии устройства, а также чертежи механического устройства, созданного поляками для взлома кода.

Команда SIS под руководством Уильяма Фридмана и Фрэнка Роуллетта [Frank Rowlett] не обладала подобными преимуществами, так что они были вынуждены перехватывать и пытаться взломать сообщения вручную там, где это было возможно. Хотя для военных шифровальщиков это относительно легко благодаря тысячам зашифрованных сообщений, отправляемым в военное время, Purple был дипломатическим шифром. Поэтому всё сводилось к ожиданию, пока накопится достаточно материала.

И наоборот, поскольку Purple был относительно новым и довольно громоздким устройством, им помогал тот факт, что далеко не каждое консульство имело подобное устройство. Это вынудило японцев иногда использовать более старые устройства шифрования 'Red [англ. красный]', уже взломанные ранее. Эта известная текстовая атака дала команде несколько потрясающих ключей к принципам работы Purple.

## Слабость на виду

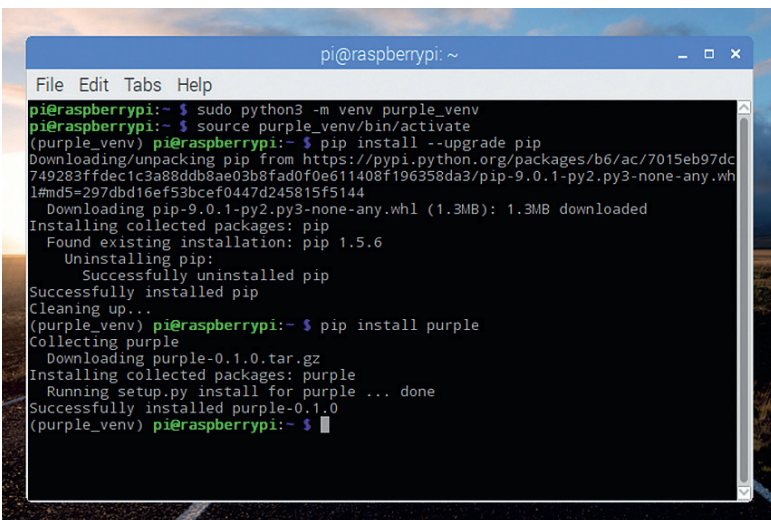
Хитроумные криптографы уже, без сомнения, заметили еще одну фундаментальную слабость Purple, унаследованную им от своего предшественника Red. Алфавиты шифрования, используемые шестерками, повторяются через каждые 25 букв. Это означало, что их можно было легко угадать по фрагментам определенных слов и их сложения, как в криптографическом кроссворде.

Хотя японцы были совершенно уверены в надежности своей системы, они всё же пытались еще больше запутать сообщения, делая их на части и используя пятисимвольные дополнения, однако это не помешало криптоаналитику SIS Женевьеве Гротьян [Genevieve Grotjan] обнаружить в сообщениях общие структуры, чтобы набросать внутренние принципы работы Purple на бумаге. Говорят, что восхищенный Фрэнк Роуллетт вознаграждал эти монументальные усилия по взлому кода во вполне гражданской манере, купив каждому члену команды по бутылке кока-колы.



**Скорая помощь**

Загляните на <https://bitbucket.org/bgneal/purple>, где вы найдете полный список опций командной строки Purple.



➤ Используйте виртуальную среду, обеспечив, чтобы ваш симулятор Purple не мешал другим проектам Python на вашем компьютере.

➤ Пропустили номер? Узнайте на с. 108, как получить его прямо сейчас.



## Шаговые коммутаторы

Таблица демонстрирует положения коммутаторов при нажатии ключей на устройстве Purple, предполагая, что переключатели двадцатки 1, 2 и 3 являются быстрым, средним и медленным коммутаторами соответственно. Коммутатор шестерки продолжает передвигаться на одно положение вперед с каждым нажатием ключа. Быстрый коммутатор делает то же самое, пока коммутатор шестерки не достигнет положения 25, и в этой точке остается неподвижным (в данном случае это положение 4); а средний коммутатор переместится на одно положение вперед (в данном случае, в положение 1).

Если коммутатор шестерки достигает положения 24, а средний коммутатор находится в положении 25, медленный коммутатор перемещается на одно положение вперед (в данном случае, с 5 на 6). Средний и быстрый коммутаторы остаются неподвижными. Конечный результат заключался в том, что алфавиты двадцатки не повторяются, пока не будет написано 15625 букв с использованием тех же самых настроек устройства.

Использование шаговых коммутаторов также означало, что, в отличие от Enigma, буквы можно шифровать самими собой, что сильно осложняет взлом зашифрованных сообщений.

Чтобы зашифровать сообщение, оператор должен сначала подключить каждый ключ к клавиатуре ввода на коммутационной панели, которая, в свою очередь, подключается к шаговым коммутаторам. Это определяло, какие буквы будут шестеркой, а какие — двадцаткой. Настройки менялись ежедневно.

Оператору также нужно было знать начальное положение всех четырех коммутаторов — например, коммутатор шестерки мог быть в положении 9. И, наконец, он задавал, какой из коммутаторов будет быстрым, какой — средним и какой — медленным.

➤ **Перемещением коммутаторов двадцатки управляет переключатель шестерки. Когда один из коммутаторов двадцатки движется, остальные остаются неподвижными.**

Sixes	20 # 1 (Fast)	20 # 2 (Medium)	20 #3 (Slow)
21	1	25	5
22	2	25	5
23	3	25	5
24	4	25	5
25	4	25	6
1	4	1	6
2	5	1	6

Дальнейшие трещины сумел нащупать лейтенант Фрэнсис Рейвен [Francis Raven], который обнаружил, что настройки ключа для каждого месяца делятся на десятидневные группы. Иными словами, начальные настройки для устройства определялись в первый день и лишь незначительно изменялись в последующие десять, что солидно облегчало взлом кода.

К несчастью, взлом Purple оказался слишком большим испытанием для Уильяма Фридмана, у которого случился нервный срыв, закончившийся госпитализацией на полтора года. В его отсутствие Рейвен и прочие сотрудники SIS создали рабочую реплику устройства Purple, ни разу не видев его.

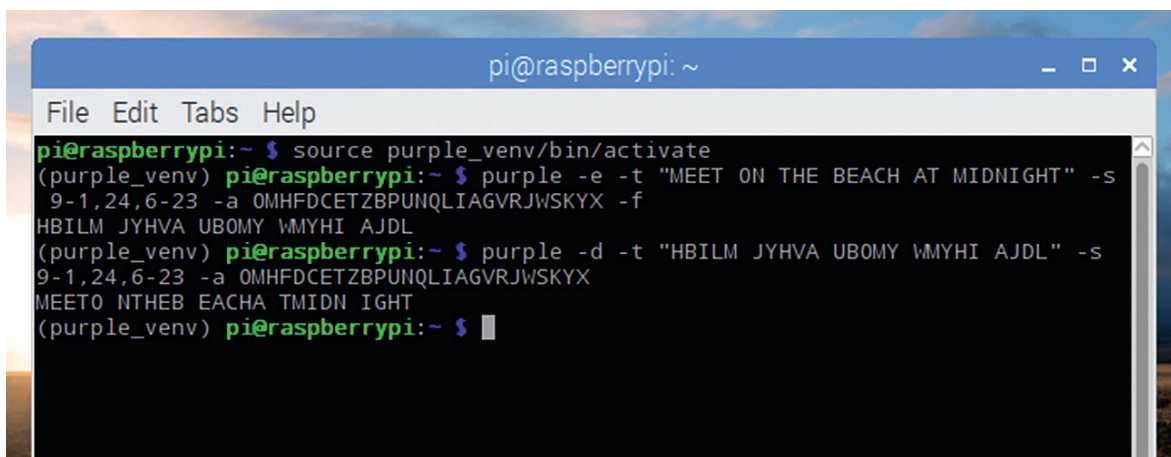
Purple Simulator, как и реплика этого устройства, созданная SIS, позволяет вам присваивать коммутационной панели только один набор настроек.

Хотя после войны японцы уничтожили все устройства Purple, на самом деле внешняя коммутационная панель у них была подключена ко второй печатной машинке, что можно было использовать для невероятного повышения надежности шифрования сообщений. В реальности, однако, настройки второй коммутационной панели не изменялись.

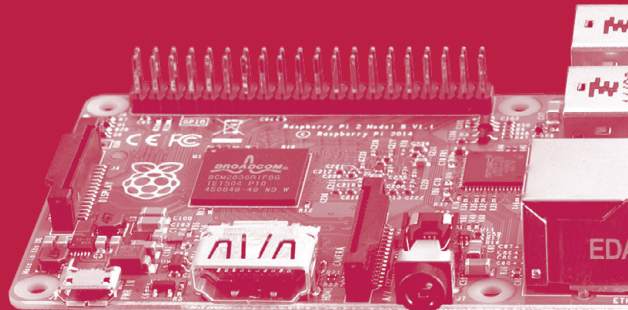
Часть неоправданного доверия японцев к их устройству, возможно, объясняется тем, что они использовали форму «безопасность через безвестность», а именно, полагали, что кодь скоро устройства надежно укрыты в посольствах, ни одна иностранная держава не сможет подвергнуть их обратному инжинирингу. И даже после капитуляции Японии в 1945 г. многие правительственные чиновники продолжали верить в то, что устройства Purple были надежными.

И хотя мы, разумеется, надеемся, что вы получите большое удовольствие от этого проекта, но надеемся также и на то, что вы извлечете из него главный урок, выраженный в форме закона Шнайера [Schneier]: «Любой человек может изобрести систему безопасности настолько умную, что он или она не додумаются, как можно ее взломать».

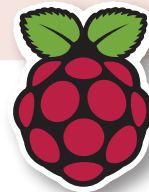
Если ваш аппетит к классической криптографии разыгрался, Брайан также разработал симуляторы для немецкой Enigma и американского устройства шифрования M-209, которые доступны на его странице BitBucket (<https://bitbucket.org/bgneal/>). Оба симулятора, как и Purple, можно установить через Python Package Index и запустить в собственных виртуальных средах. **LXF**



➤ Используйте опцию -f, чтобы фильтровать простой текст. Это облегчает шифрование сообщений.



Ваша порция смачных новостей, обзоров и учебников от Raspberry Pi



**ЛЕС ПАУНДЕР**  
живет, ест и дышит  
захватывающими  
проектами  
создателей.



Привет...

**В**чера вечером я с удовольствием посетил лекцию, которую читал Пит Ломас [Pete Lomas], один из создателей и попечителей Фонда Raspberry Pi. Мы узнали, как создавался Raspberry Pi, и про разговор Пита с еще одним основателем Аланом Майкрофтом [Alan Muscroft] в обеденный перерыв в Гайд-парке, о слабом образовании по информатике у абитуриентов Кембриджского университета. Школьникам преподавали теорию, но им не хватало практического опыта написания кода и отладки программных проблем.

Пит также рассказал, что инвестиции основателей были использованы для первой партии Raspberry Pi, и как Pi протестировали перед самым Рождеством 2011 г. только для того, чтобы найти аппаратную проблему, которую исправил 1 мм припоя для контакта центрального процессора с питанием. Raspberry Pi был почти обречен на неудачу, на грани списания в незавершенные проекты некоего Эбена Аптона. Но как только Raspberry Pi стал готов под разработку, то в тогдешнюю команду программистов прилетела задача поставить Linux на Pi — вот такой им достался пред рождественский подарок для приложения рук.

Итак, вы видите, что Raspberry Pi много раз мог просто исчезнуть или превратиться в очередной провал технического стартапа; но те, кто находился за кулисами, не сдавались и пробовали новые способы.

Каков же вывод из этой истории? Случайные встречи, обсуждение проблем и обмен идеями могут привести к большим свершениям. Будьте открыты и своими мыслями, и кодами.

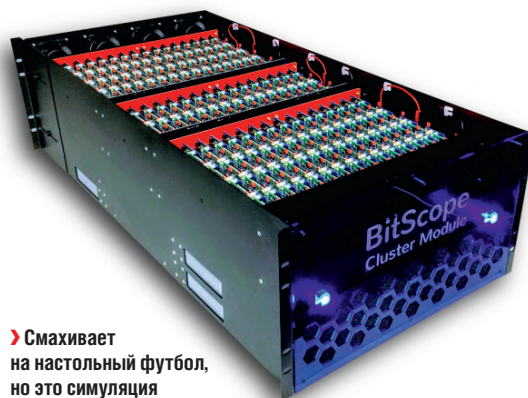
## BitScope 3000

Испытательный стенд для будущих кластеров HPC!

**В**идели мы Pi-кластеры и раньше, но не такие. Работая с Лос-Аламосской национальной лабораторией (дом Тринити, одного из крупнейших суперкомпьютеров мира), BitScope вместил 750 Pi 3 в корпус для стойки 35U. Хотя кластер Pi, потребляющий 4 кВт, не побивает рекордов производительности, он будет информировать и вдохновлять суперкомпьютеры следующего поколения.

Моделирование кластеров может пройти долгий путь, пока мы увидим, как поведут себя машины, которые еще только предстоит построить, но иногда этого мало, и требуется фактическое — хотя и недостаточное — оборудование. Не рискуя миллионами долларов на прототипирование таких штук, команда может многому научиться на скромном кластере Pi и использовать эту информацию для настройки своих экзафлопсных проектов. BitScope (также изготовитель микроосциллографа для Pi, который мы рассматривали в **LXF194**) также предлагает более скромный кластер с 6U,

со 144 узлами для любых начинающих энтузиастов. Узнайте больше на [www.raspberrypi.org/magpi/bitscope-3000-core-raspberry-pi-cluster-computer](http://www.raspberrypi.org/magpi/bitscope-3000-core-raspberry-pi-cluster-computer).

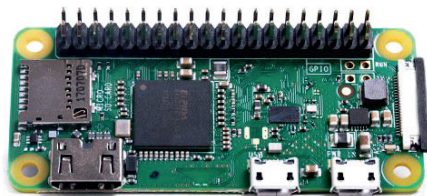


➤ Смаживает на настольный футбол, но это симуляция таких мощных машин, каких пока и в природе-то нет.

## Pi Zero WH

Штырьки, почти как — кхе — у дико-Pi-образа!

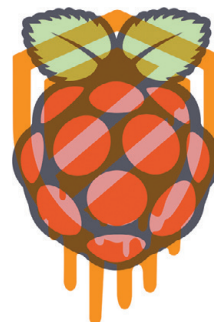
**Н**е удовлетворившись добавлением беспроводного датчика к миниатюрному Pi Zero, команда Raspberry дополнила его насадкой контактов GPIO. До WH пользователи могли бы, конечно, приделать такую насадку вручную или приделать ее прямо к плате, но теперь и хакеры, робевшие паять, могут освоить светодиоды, сенсоры или другие маленькие проекты. Они также будут обращаться к GPIO Expander на своих рабочих столах и ноутбуках. Для получения дополнительной информации перейдите на [www.raspberrypi.org/blog/zero-wh](http://www.raspberrypi.org/blog/zero-wh).



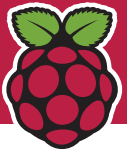
## Pi-ммунитет

Spectre и Meltdown нервно курят в сторонке.

**В**то время как мировая паника с распространением эксплоитов Spectre и Meltdown становится явной, мы-то можем расслабиться, зная, что все Raspberry Pi защищены от угрозы. В блоге Эбена Аптона [Eben Upton] всё публикуется, а также дается отличное, высокого уровня пояснение о том, как возникают уязвимости. Его пост можно прочитать на <https://www.raspberrypi.org/blog/why-raspberry-pi-isnt-vulnerable-to-spectre-or-meltdown/>.







# Vivaldi

Лес Паундер любит статистику, и обнаружил идеальный браузер для того, чтобы выяснить, сколько раз он искал пирожки, а сколько — Pi...

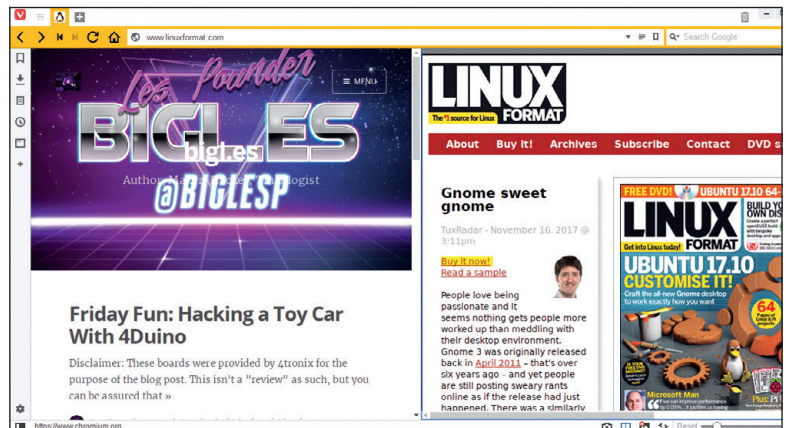
## Вкратце

» *Vivaldi* — браузер, основанный на *Chromium*, но он предлагает более быстрый и лучший интерфейс пользователя, и в нем множество опций настройки. В *Vivaldi* богатый выбор инструментов анализа, и пользователи могут установить расширения *Chrome* или *Chromium*, получив доступ ко множеству популярных дополнений.

Что это такое — браузер для Raspberry Pi? Нужен ли ему еще один? Ну, честно говоря, да. Первые несколько лет существования рабочего стола Raspbian мы пользовались браузером *Epiphany*. Хотя это умелый браузер, и отлично соответствует скудным ресурсам Pi, он просто не очень интересный.

Несколько лет назад новым браузером по умолчанию стал *Chromium*. Он отлично справлялся с задачей балансирования между потребностями пользователя и имеющимися ресурсами, но сделал сомнительный шаг, включив Flash на Pi. А когда *Vivaldi* недавно объявил себя браузером получше, мы должны были на него взглянуть.

Прежде всего, это экспериментальный браузер, основанный на *Chromium* и оптимизированный так, чтобы пользователь мог адаптировать его под свои потребности. Ярчайший пример тому — мозаичные страницы, где множество вкладок можно оставлять открытыми в одном окне. Это удобно для работы с несколькими ресурсами на одном экране. Вкладки также можно привязывать и группировать, чтобы оставлять пользовательский интерфейс ясным и простым в использовании. На самом деле, нас порадовало то, что может предложить интерфейс пользователя. Его настройка по умолчанию имеет стандартную адресную строку и значки навигации вверху экрана с окном поиска (опция по умолчанию — Bing, но это легко изменить) готово осуществлять поиск по запросам. В левой части окна размещены быстрые ссылки на Закладки и Загрузки и приятная опция



» Браузер *Vivaldi* приятен в работе и предлагает логичный интерфейс пользователя, который можно адаптировать к индивидуальным нуждам.

заметок на web-страницах: их также можно делать, выделяя любой текст на странице и нажимая на правую кнопку мыши. Записываются страница и время заметки — очень удобно для образовательных целей.

## Как набрать статистику

Значок History открывает журнал и заодно показывает, сколько раз просматривалась страница. Нажав на значок *Vivaldi* вверху слева и выбрав Tools > History [Инструменты > Журнал] мы можем увидеть более подробный анализ журнала, включающий наиболее часто посещаемые домены, то, как находятся домены (по URL, по ссылке и т. д.) и график времени просмотра, что дает нам полную картину наших привычек по просмотрам. Фанаты статистики [Ред.: — Привет, ребята!] это оценят!

Внизу окна расположена опция создания скриншотов. Когда мы ее тестировали, на рендеринг изображения ушло немало времени. К нашему удивлению, обнаружилось, что *Vivaldi* сделал рендеринг всей web-страницы в виде одного предлинного изображения. Эта функция весьма интересна, поскольку скриншоты можно просмотреть оффлайн позднее.

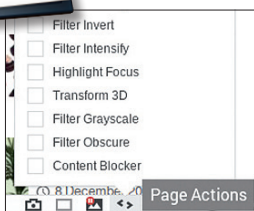
Еще одна имеющаяся внизу экрана функция — Page Options. Она позволяет изменять web-страницу, блокировать контент и делать рендеринг страницы в черно-белом цвете или же в оттенке сери. Опции также расширяются до преобразования в 3D и перекося изображений, но мы не уверены в эффективности применения этой функции.

А как там насчет функций, нацеленных на Raspberry Pi? Что ж, даже могучий Pi 3 с трудом справляется с web-контентом, в частности, анимированными GIF, поэтому их можно отключить в настройках, а режим “reader” позволяет сосредоточиться на текстовом контенте. Мы испытали браузер на YouTube, и во время воспроизведения контента его работа была довольно далека от идеальной.

*Vivaldi* — отличная альтернатива *Chromium*. Он позволяет адаптировать браузер к своим потребностям, и у него имеются отличные инструменты для расширения функциональности браузера. Он способен быть опцией на Pi 3, но владельцам Pi 2 или более старых моделей лучше будет остаться при том, чем вы пользуетесь в данный момент. LXF

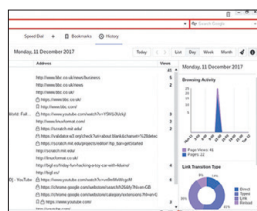


## Свойства навскидку



### Настройка страниц

Нажав на значок <>, вы можете изменить внешний вид и работу web-страницы без кодирования на JavaScript.



### История просмотров

*Vivaldi* предлагает всевозможную статистику просмотров. Найдите ту web-страницу, на которую заходили полгода назад!

## LINUX FORMAT Вердикт

### Vivaldi

Разработчик: Vivaldi Technologies

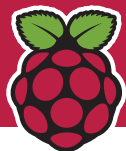
Сайт: www.vivaldi.com

Лицензия: CC Attribution 2.5

Функциональность	9/10
Производительность	8/10
Удобство в работе	8/10
Оправданность цены	8/10

» Предлагает отличный интерфейс пользователя, с богатым наследием *Chromium*. Однозначно достоин исследования.

## Рейтинг 8/10



# Raspberry Pi: Amiga с нами!

Лес Паундер обожал свой Amiga 500, и на своем уроке показывает, как вернуть золотую компьютерную эру благодаря мощи эмуляции.



## Наш эксперт

Лес Паундер работает с такими организациями, как Raspberry Pi Foundation, чтобы продвигать свои навыки умельца. И пишет блог на [www.bigl.es](http://www.bigl.es).

В 1980-х гг. существовал целый ряд домашних компьютеров от различных компаний, включая Amstrad, BBC и Sinclair. Но одна компания, Commodore, выпустила компьютер, который вышел за пределы 8-битной эры и привел к компьютерной революции.

Ее линейка домашних компьютеров Amiga предлагала нечто отличное от других. Внутренний 3,5-дюймовый дисковод для гибких дисков, 512-КБ ОЗУ, которые можно было надставить дополнительной платой, и совместимость со внешними устройствами для управления и создания контента для телевидения (ранние серии Вавилон 5 [Babylon 5] использовали контент, созданный на Amiga 4000 с помощью расширения Video Toaster).

У Amiga также имелись свои специальные журналы, в том числе Amiga Format, чей бывший редактор Ник Вейч [Nick Veitch] пошел дальше и основал периодическое издание под названием Linux Format [Ред.: — Amiga Computing навсегда!]. В этом руководстве мы немного порезвимся и создадим наш собственный Amiga 500, используя Raspberry Pi 3, а затем поиграем в классическую игру!

## Fat Agnus!

Для этого проекта мы будем использовать Amibian, операционную систему на базе Raspbian для эмуляции Amiga. Она включает интерфейс пользователя для UAE (Useless/Universal/UNIX Amiga Emulator [Беспользительный/Универсальный/UNIX]... у него много имен!), который позволит нам настроить эмулятор Amiga. Чтобы скачать Amibian, перейдите на <https://gunkrist79.wixsite.com/amibian>, нажмите Download [Скачать] и извлеките файл образа из архива.

Помимо программ для эмуляции, нам понадобится скачать *Etcher*, это простое в использовании средство создания образа SD-карты и графический инструмент для тех, кого пугает *dd*. Скачайте *Etcher* с <https://etcher.io> и извлеките его из архива.

Чтобы использовать *Etcher*, перейдите в папку, куда вы его извлекли, и дважды щелкните по его значку. *Etcher* — исполняемый файл, и его можно запускать напрямую. Когда *Etcher* откроется, вам надо пройти три этапа. Первый — выбрать образ для

записи на карту microSD. В нашем случае это образ Amibian. Затем выберите диск, на который образ должен быть записан. Обычно *Etcher* правильно определяет диск для использования, поскольку он ищет большие жесткие диски и игнорирует их в пользу меньших SD-карт. И наконец, щелчок по Flash запишет образ на карту microSD. Обратите внимание, что для записи образа вам понадобится ввести свой пароль!

Процесс записи занимает не более пяти минут. По завершении закройте *Etcher* и удалите карту microSD, затем снова вставьте ее, чтобы она могла быть смонтирована для использования.

Наш следующий шаг — получить ПЗУ [ROM] Kickstart (BIOS для Amiga), чтобы мы могли загрузить эмулятор. Если у вас уже есть Amiga, то есть и способы получения ваших собственных образов ПЗУ, но простейший и законный способ получить ПЗУ — купить приложение *Amiga Forever Essentials* для Android ([www.amigaforever.com/android](http://www.amigaforever.com/android)) и скопировать ПЗУ на ваш компьютер. Приложение стоит £1,39 и предлагает законный доступ ко всем файлам ПЗУ Kickstart для Amiga от 500 до 4000 (OC от 1.3 до 3.1).

С ПЗУ Kickstart в руках нам теперь надо скопировать их в директорию **Kickstart** SD-карты Amibian. Эта директория — самый большой раздел карты microSD, и она находится в **/root/amiga/kickstarts**. Поскольку эта папка принадлежит root, надо скопировать файл, используя терминал или менеджер файлов от имени root. Выбор второго варианта означает, что мы можем перетаскивать файлы в правильную папку:

```
$ sudo -i nautilus
```

## Только не снова, Paula!

Первоначально Amiga использовала дискеты (3,5 дюйма, 880 КБ) и эти дискеты содержали игры, приложения и операционную систему под названием Workbench. Если у вас уже есть свои диски, можете создавать их образы, используя программы в Amiga. Есть множество сайтов, предлагающих файлы ADF (Amiga Disk Files [Файлы диска Amiga]), являющиеся образами дисков, но их законность, в лучшем случае, сомнительна, поскольку они не в прямом смысле «заброшенные программы». И если вы пойдете по этому пути, мы бы посоветовали соблюдать осторожность..

Для нашего урока у нас уже есть ADF Workbench 1.3, классическая стратегия *Cannon Fodder* и оригинал диска выпуска 10 Amiga Format. Для создания образов наших дисков мы употребили *EasyADF* (<http://bit.ly/easy-adf>). Затем мы скопировали файлы ADF в директорию **floppys**, расположенную в **/root/amiga/floppys**.

Получив эти копии, размонтируйте карту microSD, вставьте ее в Raspberry Pi 3, подключите ваши внешние устройства и загрузите Amibian.

## Подтолкните меня

При первой загрузке мы видим логотип Amibian, а затем, спустя буквально несколько секунд, появляется интерфейс пользователя.



» *Cannon Fodder*, упрощенная стратегия в реальном времени, была популярна для Amiga.





## Создание крутейшей Amiga

В 1987 г. Amiga 500 стоила около £500, и для 1980-х это были большие деньги! Флагманский Amiga 4000, более мощная машина с лучшим CPU, Advanced Graphics Array [Расширенный графический массив] (AGA), чипсетом для лучшей графики и внутренним жестким диском IDE /CD-ROM, была выпущена в 1992 г. и стоила от \$3699. Так что покупка лучшего Amiga 1992 г. больно ударила по кошельку!

Но в 2018 г. мы можем воссоздать похожую конфигурацию A4000 всего лишь за £35. Всё, что

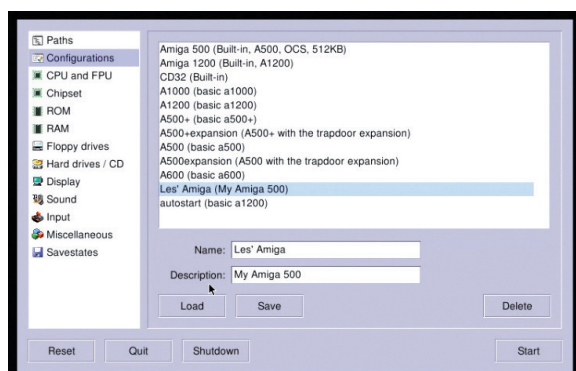
нам требуется — повесить ОЗУ до чего-нибудь между 2 МБ и 18 МБ, установить CPU на 68040 при 25 МГц (или даже быстрее!) и изменить Chipset [чипсет] на AGA.

Мы можем также добавить в нашу конфигурацию жесткий диск и установить операционную систему Workbench на готовый к использованию жесткий диск.

В пользовательском интерфейсе Amibian есть опции под названием Hard drives/CD [Жест-

кие диски/CD], и отсюда мы можем настроить устройство (USB-носитель) или создать жесткий файл (том жесткого диска в виде одного файла), на который мы затем установим нашу операционную систему. Очевидно, вам понадобится ПЗУ Kickstart 3.1 и диски Workbench 3.1, чтобы раскрепостить мощь A4000.

См. отличное видео, объясняющее процесс создания жесткого диска для Workbench 3.1: <https://youtu.be/JybjUir6ius>.



➤ Настройка вашего Amiga проста благодаря интерфейсу пользователя. Здесь мы можем увеличить ОЗУ, скорость CPU, создавать жесткие диски и менять дискеты.

Слева находятся все доступные опции конфигурации (RAM [ОЗУ], ROM [ПЗУ], CPU [ЦПУ] и т.д.). Но для нашей первой загрузки давайте нажмем Quit [Выход], чтобы перенестись в командную строку. Нам понадобится использовать командную строку Amibian, так что из текстового меню выберите опцию 6 [option 6] (Settings [Настройки]) и нажмем Enter. Затем в следующем экране введите следующее для использования *raspi-config*:

```
$ rasp
```

В *raspi-config* выберите опцию меню для расширения файловой системы и нажмем Enter, чтобы начать. Чтобы изменения вступили в силу, нам понадобится перезагрузиться и вернуться к пользовательскому интерфейсу Amibian.

### Придержи и поменяй

Amiga 500 поставлялся с 512 КБ ОЗУ (расширяемого до 9 МБ), а CPU был Motorola 68000, работающий на 7,09 МГц. Несмотря на эту скромно звучащую конфигурацию, Amiga был локомотивом своего времени, способным к многозадачным операциям, воспроизведению оцифрованного видео и отображению до 4096 цветов на экране (в режиме NAM [Hold-And-Modify, особый режим «удержания и поправки» цвета выводимого пикселя]).

Мы намерены настроить базовый Amiga 500 с 1 МБ ОЗУ. Вначале перейдите в Configurations [Настройки], создайте новую конфигурацию как <ВАШЕ НАЗВАНИЕ> Amiga и нажмем Save [Сохранить]. Затем перейдите в CPU и FPU и задайте CPU 68000, FPU — None и CPU Speed — 7 МГц. Далее перейдите в Chipset. Для этого выберите OCS, Blitter to Immediate и не отмечайте окошко Copper. Затем перейдите в ROM и с помощью кнопки ... откройте диалог и перейдите в /root/amiga/kickstarts. Вы увидите ПЗУ Kickstart, которые мы скопировали ранее. Выберите *amiga-os-130.rom*

(или как там называется ваш 1.3 Kickstart ROM) и нажмем OK. Теперь установите RAM для Amiga. Задайте Chip RAM на 1МБ (расширенный A500.)

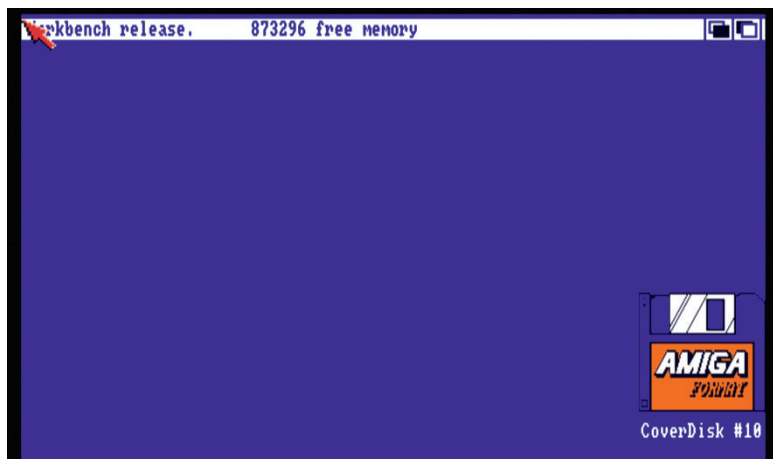
Чтобы поиграть в игру, следует вставить дискету(ы) в Amiga. У Amiga может быть до четырех гибких дисков (один внутренний, три внешних), и для нашей демо-версии мы играли в *Cannon Fodder* с тремя дискетами. Итак, мы включили DFO (внутренний) и использовали меню ..., чтобы разместить ADF-образ игры, которую мы хотели запустить.

Последний этап настройки перед самой игрой — настроить джойстик. Если у вас есть USB-джойстик, то Amibian должен обнаружить его и настроить, но вы можете подстроить его сами, используя опцию конфигурации Input. Не забудьте сохранить свою настройку для вашей следующей игры с помощью опции Configurations.

### Время Rock Lobster

После всей этой возни мы наконец-то можем поиграть. Amibian будет эмулировать скорость гибкого дисководов Amiga, так что не переживайте — ваша игра загрузится через минуту или около того! Если вам не терпится, нажмем F12, чтобы открыть интерфейс пользователя, перейдите во Floppy Drives и измените скорость эмуляции на ту, которая будет под стать вашему терпению. F12 также можно использовать для изменения настроек, загрузки новой дискеты и перезагрузки Amiga в случае сбоя игры.

Итак, мы его получили — эмулированный кусочек видеоигры и компьютерной истории, а также что-то, что вернет нам в *Linux Format* множество чудесных воспоминаний. **LXF**

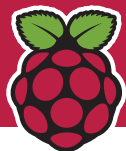


➤ Вот диск 10-го номера *Amiga Format*. На тот момент журнальные диски были самым популярным способом поделиться информацией о приложениях и играх!

### Скорая помощь

Серия компьютеров Amiga предлагалась во множестве конфигураций. Наиболее подходящим для игр был Amiga 500 Kickstart 1.3, но для игр, требующих улучшенной графики и мощности, нужны будут чипсет AGA, 2 МБ ОЗУ и CPU 68020!

» Подпишитесь на печатную или электронную версии на [www.linuxformat.ru/subscribe!](http://www.linuxformat.ru/subscribe!)



# Гидропоника: Поливка огорода

Если у вас нет талантов садовника, не горюйте: **Кент Ельчук** даст вам способ следить за садом по снимкам, видео и датчикам влажности.



## Наш эксперт

**Кент Ельчук** — штатный web-разработчик и энтузиаст Linux, который проводит свое свободное время за программированием и выращиванием гидропонных пищевых культур.

Урок этого месяца будет руководством по выращиванию здоровой гидропонной (она же — органическая гидропонная) пищи с помощью Raspberry Pi. У гидропонии множество преимуществ по сравнению с более привычными методами выращивания продуктов, в том числе быстрый рост и улучшенное водопотребление.

Поскольку растения потребляют свои 17 основных питательных элементов в жидкой форме, выбор гидропонии позволит садоводам-любителям применять для своих растений точную диету, используя при этом для корневой зоны стерильную среду, свободную от патогенных организмов, таких, как кишечная палочка (E.coli) и сальмонелла.

Сейчас кое-кто из вас задается вопросом: а каким боком Raspberry Pi относится к гидропонике или органической гидропонике? Так ведь это настолько полезное устройство, что оно выполняет сразу две задачи: следит за садом удаленно через web-камеру и определяет уровень влажности.

Хотя основной целью нашего урока являются технология Linux и выращивание овощей, мы также вкратце расскажем о процедуре организации сада и процедуре подкормки. Это позволит вам понять, как всё связано.

Есть немало web-камер, которые начинают работать сразу, и две очень популярных и доступных камеры, которые часто встречаются в продаже — это Logitech, модели C170 и C270. Их можно приобрести на сайте Logitech ([www.logitech.com](http://www.logitech.com)) по цене £17 и £26 соответственно.

Для определения влажности мы воспользуемся датчиком Soil Moisture, доступным на сайте Mod My Pi (<http://bit.ly/soil-moisture>). Устройство стоит весьма приемлемые £4, и его можно установить за несколько минут. Оно подключается к Raspberry Pi GPIO, а другой конец втыкается в емкость с растением.

Мы расскажем о двух методах доступа к обновлению статуса полива. Один использует Raspberry Pi в качестве web-сервера, второй получает доступ к удаленному адресу сайта.

## Вырастите web-сервер

Предположим, у нас нет учетной записи web-хостинга для передачи файлов. Не проблема: мы можем использовать свой Raspberry Pi и получить доступ к нему через браузер и IP-адрес. Каждый домашний интернет-план имеет IP-адрес. Некоторые являются фиксированными, а некоторые — статическими.

Мы можем увидеть свой IP-адрес, открыв ПО роутера. Однако более простым решением будет посетить <http://myipaddress.com>, который даст нам всю необходимую информацию.

Узнав свой IP-адрес, мы можем ввести его в браузер, чтобы увидеть поток. Обратите внимание, что для удаленного просмотра web-контента с помощью IP-адреса нужна переадресация портов. Ее можно настроить с помощью регистрации роутера.

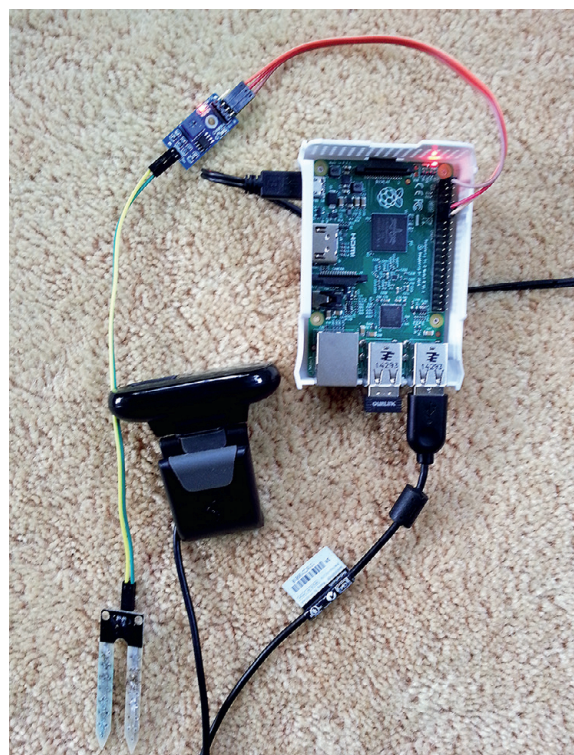
Как мы сказали в начале статьи, для отслеживания влажности почвы будет использоваться датчик Soil Moisture.

Давайте начнем с монтажа. Чтобы собрать датчик влажности, мы подключим его к трем выводам GPIO на Pi: GPIO 17, питание и заземление. Затем подключим плату датчика к металлическим пластинам щупа-влажмера. Весь набор идет со всеми частями. На фото внизу вы видите, как всё должно быть подключено.

После подключения устройство сразу готово измерять влажность. Итак, всё, что нам остается сделать — воткнуть щуп в землю и полить растение. Если влажность определяется, загораются два индикатора. Если нет, только один. По умолчанию вывод цифровой, и для определения влажности применяет простую процедуру Вкл/Выкл. Потратив немного времени и усилий, мы сумели соорудить для устройства аналоговый вывод. Но на нашем уроке всё будет просто.

Это устройство способно определять, когда растение полито свежей водой. Но в итоге, это не так уж много. Примерно через час после того, как вода впитается, индикатор влажности погаснет.

Итак, если эта статья рассказывает о домашних растениях, и индикатор гаснет через час после полива, что нам делать, если поливать их надо только раз в неделю? Немного, честно говоря. Но поскольку это статья о гидропонике, настройка нормальная,

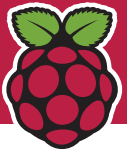


➤ USB-камера присоединена к порту USB, датчик влажности подключен к GPIO 17, питанию и заземлению, а плата датчика — к металлическим пластинам влажмера.

## Скорая помощь

Домашние серверы — это не точная наука. Некоторые интернет-провайдеры включают переадресацию портов для размещенных домашних сайтов, тогда как другие порт блокируют. Чтобы решить эту проблему, мы всегда можем использовать дешевый web-хост и передавать данные на удаленный сервер... или просто сменить интернет-провайдера.





## Сад в движении

Первым делом разберемся с настройкой и подробностями мониторинга растений. Для этого мы просто устанавливаем *Motion*, командой

```
sudo apt install motion
```

После установки *Motion* надо отредактировать файл `/etc/motion/motion.conf`: когда он откроется в нашем обычном любимом редакторе, надо просто заменить словечко, например, 'on' на 'off'.

Хотя порядок строк распространяется на весь файл, список ниже показывает, какие это должны быть строки. Если мы используем редактор *Vim*, для поиска соответствующей строки с требуемым словом можно употребить прямой слэш и слово, за чем следует нажатие на Enter [Ред.: — А в любом другом разумном редакторе, Ctrl+F].

```
daemon on
width 640
height 480
framerate 5
ffmpeg_output_movies on
snapshot_interval 60
stream_localhost off
webcontrol_localhost off
```

В файле `/etc/default/motion` добавьте следующую строку:

```
start_motion_daemon=yes
```

Помимо этого, не помешает обеспечить, чтобы *Motion* запускался автоматически при перезагрузке нашего Pi:

```
update-rc.d motion enable
```

Следующий шаг — добавить web-сервер *Apache* и PHP, если они отсутствуют на нашем Pi. PHP будет использоваться при отправке файлов на удаленный сервер и для незатейливого скрипта мониторинга полива.

Во время установки обязательно введите Y и нажмите на Enter, чтобы процесс установки пошел.

```
sudo apt-get update
```

```
sudo apt-get install apache2
```

```
sudo apt-get install php7.0 php-pear libapache2-mod-php7.0 php7.0-mysql
```

```
sudo apt-get install php7.0-curl php7.0-json php7.0-cgi
```

Теперь, перезагрузив Pi и открыв сетевой адрес Pi, например, `192.168.1.777:8000`, мы должны увидеть live-поток.

поскольку растения подкармливаются один-два раза в день. Т.е. легко работать с обнаружением и на таком уровне, и оно будет не хуже любого другого устройства. Следить за влажностью можно удаленно, так как у нас будет ежедневный таймер для подкормки растений с помощью насоса.

Ключевым в данном проекте является применение процедуры *cron* для проверки вывода — включен он или выключен. Если он оказывается включен, то мы получаем предупреждение. Таким образом мы знаем, совершена ли ирригация или нет.

И если, будучи в дороге, мы получим предупреждение в плановое время, то можем быть уверены, что наши растения не умрут от засухи. Это предупреждение особенно полезно для последней дневной подкормки, чтобы мы могли мирно уснуть или просто знать, что не надо никому звонить и просить полить растения.

## Внимание, влажность!

Итак, давайте опустимся на землю и замараем руки кодом для определения влажности и создания предупреждений. Такой небольшой скрипт можно сделать в Python или PHP.

В нашем случае мы обратимся к PHP, поскольку там имеется функция `shell_exec()`, которая позволяет напрямую использовать команды оболочки Linux, не впуская библиотеку GPIO и ее функции в Python. Но помните, что PHP тоже может обращаться к библиотеке GPIO, а это подразумевает дополнительную установку.

Итак, давайте взглянем на код ниже и скопируем его в файл под названием `moist.php`, расположенный в папке `/var/www/html`. Подробное объяснение будет позднее. Доступ к этому файлу можно получить через URL <http://myipaddress/moist.php>. Таким образом, файл служит двойной цели: посмотреть выведенные данные и снять показания датчика.

```

```

```

```

```
<?php
```

```
$on_off = shell_exec('gpio read 0');
```

```
echo $on_off;
```

```
if($on_off == 1){
```

```
echo "<br/>It is off";
```

```
$fp = fopen('sensor.txt', 'w');
```

```
fwrite($fp, $on_off);
```

```
fclose($fp);
```

```
}else{
```

```
echo "<br/>It is on!<br/>";
$fp = fopen('sensor.txt', 'w');
fwrite($fp, $on_off);
fclose($fp);
}
```

Вот как это работает. Первые две строки отображают live-поток и самое последнее изображение с web-камеры.

Затем библиотека Linux GPIO, поставляемая вместе с Raspberry Pi, считывает показания датчика и возвращает '1' при сухости и '0' при влажности.

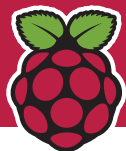
Если вдруг мы решим записать данные, строки между `fopen()` и `fclose()` перепишут файл с именем `sensor.txt`, который обычно хранит 0 или 1 с датчика.

Ну вот, код у нас есть; давайте настроим процедуру *Cron*, которая будет запускаться автоматически. В нашем случае мы хотим, чтобы она запускалась дважды в день, в 10 и в 15 часов: именно такую схему полива автор использовал в прошлом. Тогда пять частей головоломки должны выглядеть так:



» Помимо насоса, для системы подачи служат 1/2-дюймовые трубы, 1/2-дюймовые линии подачи, 1/4-дюймовые штуцеры и всяческие 1/2-дюймовые фитинги из ПВХ.

» Подпишитесь на печатную или электронную версии на [www.linuxformat.ru/subscribe!](http://www.linuxformat.ru/subscribe!)



```
0 10,15 * * * *
```

Пример ниже ежеминутно показывает процедуру *Cron*; ту, которая запускается в 10 и в 15 часов. Для тех, кто пока не знаком с процедурами *Cron*, давайте объясним, как они работают. Есть пять разделов для запуска, слева направо: минуты, часы, дата, месяц, день недели. После расписания команда записывается так:

```
0 10,15 * * * php /var/www/html/moist.php >> /var/www/html/moist.txt 2>&1
```

Теперь небольшое примечание об изображениях. Поскольку путь для вывода изображений задан для папки `/var/www/html`, нам надо скопировать туда этот файл из `/var/lib/motion` и дать ему соответствующие разрешения. Эта процедура будет выполнена *Cron*, как показано ниже:

```
* * * * * cp /var/lib/motion/lastsnap.jpg /var/www/html/test.jpg && chown pi:pi /var/www/html/lastsnap.jpg
```

Вот несколько подсказок по файлу `moist.php`. Он будет проверять текущую влажность и записывать в файл каждый раз, когда вы запустите его в браузере или из процедуры *Cron*. В конце этой статьи будет еще один блок кода, который считывает файл `sensor.txt` и отображает результат. Следовательно, вы всегда можете запустить задание *Cron moist.php* из пути вне web-директории (например, `/home/pi/moist.php`) и просто отобразить результат (показания влажности и изображение) из директории `/var/www/html`. Это наша рекомендуемая опция.

## Удаленные серверы

План заключается в передаче данных на удаленный сервер, но нам всё еще нужна предыдущая настройка Raspberry Pi, поскольку она делает фото и считывает данные о поливе. Итак, для садовода-

любителя, желающего передавать и считывать результаты полива и самые свежие изображения на web-сервер, этот раздел объяснит процедуру.

## Передача файлов

При запуске *Motion* сохраненные файлы получают временную метку, и самый свежий файл всегда именуется `lastsnap.jpg`. Это упрощает отслеживание ситуации. Если у нас работает *Motion*, мы можем заглянуть внутрь папки `/var/lib/motion` и посмотреть все изображения, которые делаются каждые 100 секунд.

Если изображения не появляются в папке `/var/lib/motion`, мы должны обеспечить ей разрешения для пользователя *Motion*. Команда `chown -R motion:motion /var/lib/motion` мигом решит эту проблему.

Есть разные методы передачи файлов с Raspberry Pi на удаленный сервер: среди прочих, SCP, FTP, SFTP или FTPS. Если мы не хотим ничего усложнять, и оба сервера управляются одним администратором, то можно использовать любой из этих методов.

С другой стороны, если у нас есть камеры с разных сторонних сетей и нам надо разместить все изображения на одном удаленном сервере, это потребует больше работы. С этой целью мы можем создать учетные записи FTP для каждого Raspberry Pi, ведущего мониторинг. Затем каждый Pi будет аутентифицировать и загружать файл на собственную учетную запись.

При наличии нескольких устройств, каждому понадобится собственное имя пользователя и пароль, и файл будет передаваться с помощью команды `curl`. Вот и всё, потому что как только сервер опознает пользователя и пароль, он уже будет знать, какая папка получит файл.

## Зеленый FTP

Прежде чем двигаться дальше, давайте рассмотрим первую команду, являющуюся базовой передачей FTP. Мы даем файлу под названием `send.sh` права на исполнение, и остальное будет происходить само по себе.

Кстати, всегда помните, что для запуска файл должен быть исполняемым. Посмотрите на команду ниже и на блок настоящего кода после. Когда мы рассмотрим эти два фрагмента кода, я объясню процесс `ftp`, который делает ту же операцию, что и инструмент типа *Filezilla*, но автоматически.

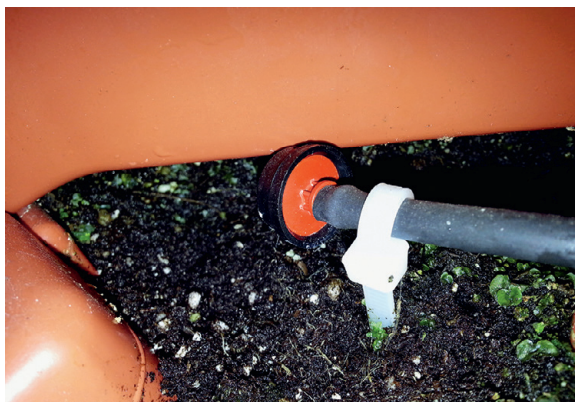
```
chmod +x /var/lib/motion/send.sh
#!/usr/bin/env bash
PASSWORD=Member111
HOST='ftp.example.com'
```

### Скорая помощь



У процедур *Cron* есть пять временных интервалов, которые настраиваются в начале каждой процедуры. Слева направо: минуты (0–59), час (0–3), число (1–31), месяц (1–12) и день недели (0–7).

➤ Хотя опоры для растений отлично работают, эта капельница на 2 литра в час поддерживается обрезком стяжки, который стоит копейки.



## Основы гидропоники

Счетверенные горшки можно ставить друг на друга. Горшки с помидорами размещаются по два или три в высоту, а перцы, салат и огурцы — по пять или шесть. Насос поставляет жидкость через 0,5-дюймовые трубки с колпачками. Вдоль коллектора размещаются штуцеры, которые вставляются в коллектор, присоединяемый к меньшей по размеру, 0,25-дюймовой трубке. Она, в свою очередь, ведет к горшкам.

В конце 0,25-дюймовой трубки находятся капельницы на 2 литра в час. Капельницы крепятся на опоры для растений. Две капельницы в верхнем горшке и отдельные трубки к нижним горшкам

отлично работают. В самом низу системы находится один горшок на 3 галлона [примерно 11 литров, — прим. пер.], где тоже можно разместить растения.

Что касается среды, мы можем выбрать любую гидропонную среду по своему усмотрению, например, гидропонную смесь или кокосовые волокна. Подойдет также любая гидропонная питательная смесь; но дешевле всего купить порошковую. Обычный пищевой контейнер на 55 галлонов вполне сгодится в качестве резервуара.

Два инструмента, которые обязательно должны быть у всех, кто занимается гидропоникой, это pH-измеритель [кислотности] и измеритель

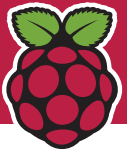
электропроводности EC/TDS. Однако можно сократить расходы и просто использовать полоски pH-теста или жидкостный тест.

Диапазон pH 6.0–6.5 соответствует общим потребностям овощных культур, а 1500–1800 ppm (частей на миллион) годится большинству овощей, хотя для салата-латука хорошо 400–500 ppm.

Поскольку гидропонные культуры дают немалый урожай, таким растениям, как болгарский перец, помидоры и перец-халапеньо, необходимы колья для поддержки. Салат и огурцы требуют меньшей заботы, а многие разновидности помидоров могут просто свисать вниз.

➤ Пропустили номер? Узнайте на с. 108, как получить его прямо сейчас.





➤ К каждой «колонне» прикреплены горшки. В каждом горшке четыре растения, и на колонне может размещаться от двух до четырех горшков в зависимости от выбранных растений.

```
USER='member@members.example.com'
FILE='lastsnap.jpg'
cd /var/lib/motion
ftp -n $HOST <<END_SCRIPT
quote USER $USER
quote PASS $PASSWORD
binary
put $FILE
quit
END_SCRIPT
exit 0
```

В этом скрипте вторая строка со служебным сочетанием символов (#! — «шалаш [shebang]») отсылает файл **send.sh** интерпретатору *Bash*. Затем следует обычная информация в виде имени пользователя и пароля, которая позволяет файлу переместиться на другой сервер. Конечно, файл очевиден.

После этого команда **cd** переходит в директорию, где размещен файл — в данном случае, папку **/var/lib/motion**. Следующая команда большая. Это та самая команда **ftp**, которая вызывает удаленный хост.

Команды в кавычках позволяют FTP делать обход, добавлять имя пользователя и пароль вручную и использовать наши, приведенные в файле, чтобы установить соединение.

Ближе к концу двоичная команда делает очевидное: осуществляет передачу в бинарнике. Команда **put** нужна для перемещения файла. Как только скрипт запускается, она завершается.

Строкой ниже — процедура *Cron*, которая каждую минуту запускает скрипт **send.sh** и отправляет самое последнее изображение на удаленный сервер на **example.com**. На удаленном компьютере создается пользователь FTP, и программа типа *Cpanel* позволит нам управлять пользователями и паролями FTP.

```
*/* * * * * /var/lib/motion/send.sh > /dev/null 2>&1
```

Та же процедура применима для отправки файла с именем **sensor.txt**, который записывает показания датчика. Разница в том, что файл **sensor.txt** размещен в папке **/var/www/html**. Таким образом, имя файла и путь — единственное, что нам надо изменить в приведенном в качестве примера файле **send.sh**, который обсуждался выше, и в методах передачи, которые мы обсудим далее. Итак, всё, что нам надо сделать — использовать команду оболочки Linux **cp** для создания копий файла **send.sh** и изменения пути и имени файла. Далее мы копируем процедуру *Cron* для передачи файла и изменения имени файла и пути.

## Curl-линг с канадцами

Двигаемся дальше: давайте рассмотрим другие методы передачи файлов с помощью *curl* и *scp*.

```
*/1 * * * * /usr/bin/curl --ftp-ssl -T "/var/lib/motion/lastsnap.jpg" -k
-u "member@members.example.com@example.com:Member111"
ftp://example.com
```

## Безопасное копирование

```
*/1 * * * * scp /var/lib/motion/lastsnap.jpg pi@ipaddress:/tmp
*/1 * * * * scp -r /var/lib/motion pi@ipaddress:/tmp
```

Последняя команда отправляет целую папку, которая вскоре заполнится — из-за того, что новое фото снимается каждые 100 секунд. Отправка всей директории может быть хорошим методом «увидеть сразу всё», но это бывает непрактично, поскольку в некоторый момент нам, вероятно, понадобится удалять лишние файлы.

На удаленном сервере будет использоваться простая web-страница для отображения самых последних изображений и показаний датчика. Этот код показан далее. В качестве обновления, файл **sensor.txt** воссоздается только дважды в день после того, как таймер насоса запустит гидропонную систему, т. е. когда мы получаем наши показания датчика влажности.

В порядке защиты от дурака, если снимать показания через полчаса или час после запуска насоса, вокруг влагомеров будет достаточная влажность. Что же касается изображения, оно обновляется каждую минуту, так что мы видим его практически в реальном времени.

```

<?php
foreach(file('sensor.txt') as $on_off) {
//echo $on_off. "\n";
}
echo $on_off;
if($on_off == 1){
echo "<br/>It is off";
}else{
echo "<br/>It is on!<br/>";
}
?>
```

## Займитесь садоводством!

Ну что ж, в основном это всё. У нас есть инструменты и технологии для создания монитора сада и просмотра его состояния из любой точки мира. В фантастическом режиме мы можем расширить наш сад или ферму, охватив разные места и следить за ними всеми — хоть на полмира! Удачи вам и агрономических успехов. **LXF**

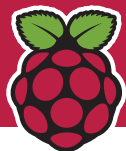


➤ Автор и его растения жарким июльским днем. В это время растениям нужно как минимум три полива в день, запланированные таймером.

### Скорая помощь



Когда это возможно, используется порошковое удобрение, которое разводится за ночь или как минимум за 20 минут перед использованием, чтобы порошок полностью растворился.



# Twitter: Ботаем

## Нейт Дрейк настроит на своем Raspberry Pi Twitter-бот и заспамит Сеть!



### Наш эксперт

**Нейт Дрейк** — технический журналист, специализирующийся на кибербезопасности, ретро-технологиях и общении с российскими ботами, для развлечения и выгоды!

### Скорая помощь

Создав жетон доступа, вы можете удалить номер телефона со своей учетной записи. Для более подробной информации посетите <https://support.twitter.com/articles/81940>.

**П**ростейшее описание Twitter-бота — это программа, разработанная для создания автоматических сообщений в Twitter. Учитывая то, как просто твитить самостоятельно, стоит объяснить, почему люди берут на себя труд сделать это.

Самое распространенное использование бота — для [Ред.: — Русского?] спама. Бот проверяет на наличие ключевых слов — например, “cleaning products [моющие средства]” — и затем отвечает рекламной ссылкой, чтобы люди нажимали на нее и попадали на сайт спамера. Однако боты применяются и не для спама. Бота можно запрограммировать на поиск любого слова или фразы и соответствующий ответ. Одной из развлекательных реализаций этого был @BDZNappa, который искал людей, твитивших фразу “over 9,000 [более 9000]”, и всегда отвечал твитнувшему “WHAT!? NINE THOUSAND? [ЧТО?! ДЕВЯТЬ ТЫСЯЧ?]”.

Боты, кроме того, могут твитить из текстового источника, например, @SunTzuBot, который твитит ежедневные цитаты из *The Art of War* [«Искусство войны» — известный древнекитайский трактат о стратегии и тактике, написанный Сунь-Цзы, — прим. пер.]. Более сложные боты, например, @JustDiedBot, активно ищут в Интернете источники. @JustDiedBot ищет в Википедии информацию о недавних смертях и твитит сообщения о смертях, когда они происходят.

У ботов также есть практическое применение. Их можно запрограммировать твитить с определенными интервалами и использовать как «аварийную кнопку», отправляя твит каждый день, пока вы не перезагрузите его. Также возможно составить расписание твитов на будущее и использовать бота для отправки напоминаний.

### Pi-бот готов!

Если вас заинтересовало какое-либо из этих применений, настройте учетную запись Twitter для своего бота. Не поддавайтесь искушению использовать для этого свою существующую учетку Twitter, если она у вас есть, так как повторяющиеся твиты могут быть ошибочно приняты за спам Twitter, и ваша учетка будет приостановлена.

Чтобы продолжить, вам понадобится новая учетка Twitter с подтвержденным номером телефона. Если у вас уже есть учетка Twitter, посетите <https://support.twitter.com/articles/81940> и следуйте

описанным там шагам, чтобы до поры до времени убрать номер телефона со своей учетки.

Затем посетите [www.twitter.com](http://www.twitter.com) и выберите Sign Up [Регистрация]. Вы можете использовать любой адрес электронной почты по вашему желанию, если он не зарегистрирован в Twitter. Дайте боту подходящее имя. Для целей нашего урока мы создадим бота с именем Sherlock Bot с ID пользователя @holmesbot1.

Вас попросят предоставить ваш номер телефона и подтвердить его по СМС. Twitter требует это для снижения вероятности появления спам-ботов.

Как только ваш учетная запись Twitter будет создана, вам может понадобиться несколько раз нажать Continue вверху справа, чтобы пропустить предложения импортировать ваши контакты или читать других пользователей.

Затем надо создать приложение Twitter, которое позволит вашему Pi получить доступ к вашей учетке Twitter. Перейдите на <http://apps.twitter.com>, а затем нажмите Create New App [Создать новое приложение].

Вас попросят заполнить Application Details [Сведения о приложении]. Для целей данного руководства мы создадим бота, который будет регулярно твитить истории про Шерлока Холмса, но вы спокойно можете изменить Name [Имя] и Description [Описание] как посчитаете нужным.

Под Website теперь просто поместите [www.twitter.com](http://www.twitter.com). Оставьте поле с названием Callback URL [URL обратной связи] пустым. Отметьте галочкой, что вы согласны с Twitter Developer Agreement [Соглашение разработчика Twitter], затем нажмите серую кнопку с названием Create Your Twitter Application [Создать ваше приложение Twitter].

Убедитесь, что Access Level [Уровень доступа] показывает Read and Write [Чтение и запись]. Если это не так, нажмите Modify App Permissions [Изменение разрешений приложения], чтобы изменить это.

Затем нажмите Manage Keys [Управление ключами] и Access Tokens [Жетоны доступа]. Прокрутите вниз страницы и нажмите серую кнопку с названием Create My Access Token [Создать мой жетон доступа].

Вы увидите сообщение, извещающее, что жетон доступа создан. Сохраните копию этой страницы в безопасном месте или оставьте ее открытой в вашем браузере, поскольку она вам в ближайшее время понадобится.

Затем откройте приложение терминала в своем Pi или подключитесь к нему через SSH. Вначале надо установить дополнительные программы, следующей командой:

```
sudo easy_install pip
```

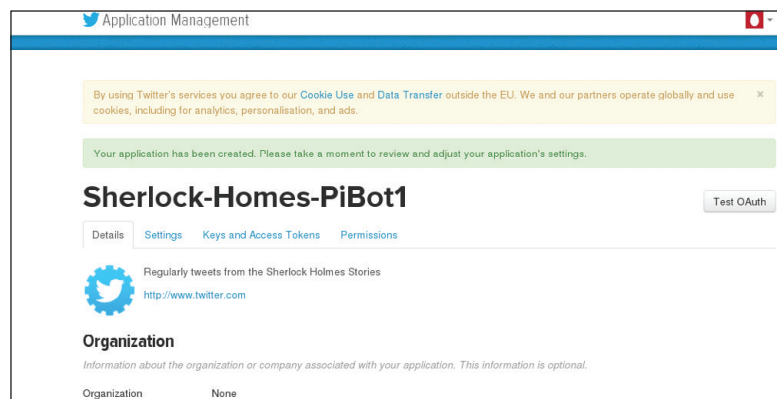
Затем создайте директорию для бота и откройте ее:

```
mkdir holmesbot1 && cd holmesbot1
```

Для этого примера мы перейдем к специализированной библиотеке Python, созданной Эдвином Далмайером [Edwin Dalmaier], под названием Markovbot. Говоря в основном, программа берет текст из источника (в нашем случае, из «Приключений Шерлока Холмса») и случайным образом составляет из него предложения правдоподобного вида.

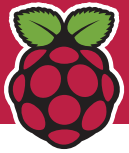
Вам надо скачать и распаковать программу командой

```
wget https://github.com/esdalmajier/markovbot/archive/master.zip && unzip master.zip
```



➤ Создав приложение, прокрутите вниз, чтобы включить жетон доступа. Это позволяет боту входить в вашу учетку Twitter.





## Любопытные боты

Боты существуют не первый год, и порой даже имеют тысячи подписчиков. Некоторые из них просто реагируют на слова в твитах. Например, любой, использующий фразу “illegal immigrant [нелегальный иммигрант]” может ожидать ответа от @DroptelBot с сообщением “People aren’t illegal. Try saying ‘undocumented immigrant’ or ‘unauthorised immigrant’ instead [Люди не нелегалы. Попробуйте вместо этого сказать „Иммигрант без вида на жительство“ или „Несанкционированный иммигрант“]”.

Бот Twitter @everyword в 2007 г. затеял ответить все слова английского языка. Новое слово твитилось каждые 30 минут, пока в 2014 г. бот не выполнил свою задачу после 109157 слов. Автор даже опубликовал книгу об этом событии.

Один из ботов задуман как политическая пародия. Упоминание слов “communism [коммунизм]” или “socialism [социализм]” вызывает гнев @RedScareBot, Робота Дж. Маккарти [Robot J Mc-

Carthy], который сообщает вам “Red Storm is rising [Красный шторм поднимается]” [роман американского писателя Тома Клэнси о Третьей мировой войне между странами Варшавского договора и НАТО, — прим. пер.] и “circle the wagons [Защитить общие интересы]”.

Есть еще @DeepDrumpf — это бот Twitter, созданный MIT, который использует нейротехнологии для анализа данных и создания твитов в предполагаем стиле разговора Дональда Трампа [Donald Trump]. Разработчики утверждают, что бот обучался, используя стенограммы выступлений Дональда Трампа.

Бот Twitter @factbot1 нашел неплохое применение изображениям. Его создатель, Эрик Драсс [Eric Drass], запрограммировал бота в ответ на тенденцию некоторых людей верить недоказанным фактам, если таковые подкреплены картинкой. Бот регулярно твитит почти правдивые, однако



➤ Робот Дж. Маккарти здесь, чтобы напоминать нам о предполагаемой опасности Серпа и Молота.

и бессмысленные факты. Один из примеров таких «фактов» выдает сообщение, что канадское правительство якобы получает 38% своих доходов от продажи пончиков. Звучит правдоподобно, учитывая вездесущность кафе Тима Хортонса [Tim Hortons] в Стране кленового листа, но на самом деле это не так.

Перейдите в новую директорию по `cd markovbot-master` и установите другие требуемые программы с помощью команд

```
wget https://bootstrap.pypa.io/ez_setup.py && sudo python ez_setup.py
sudo easy_install twitter
```

Затем надо скачать текстовый файл, который станет источником наших случайных твитов. Наш файл идет с сайта «Проекта Гутенберг», но вам ничто не мешает использовать любой приглянувшийся файл TXT:

```
wget http://www.gutenberg.org/cache/epub/1661/pg1661.txt
```

Затем мы создаем пустой файл, чтобы разместить там наш код. Вы можете выбрать любое имя на свое усмотрение, при условии, что вы используете расширение `.py` в конце:

```
nano sherlock1.py
```

Введите в новом файле следующий код:

```
import os
import time
from markovbot import MarkovBot
#####
# ИНИЦИАЛИЗАЦИЯ
# Инициализируем экземпляр MarkovBot
tweetbot = MarkovBot()
# Получаем путь к текущей директории
dirname = os.path.dirname(os.path.abspath(__file__))
# Конструируем путь к книге
book = os.path.join(dirname, u'pg1661.txt')
# Заставим бота читать книгу!
tweetbot.read(book)
#####
# TWITTER
# MarkovBot использует @sixohsix Python Twitter Tools,
это обертка Python
# для API Twitter. Ищите ее на GitHub: https://github.com/sixohsix/twitter
# ВСЕ ВАШИ СЕКРЕТЫ!
# Замените '' ниже на свои значения, или поищите
# более безопасный способ работы со своими ключами
# и жетонами доступа. Вы предупреждены, что
# помещать ключи и жетоны в простой текстовый скрипт
НЕБЕЗОПАСНО!
# Consumer Key (API Key)
cons_key = 'вашключ'
```

```
# Consumer Secret (API Secret)
```

```
cons_secret = 'вашсекрет'
```

```
# Жетон доступа
```

```
access_token = 'вашжетондоступа'
```

```
# Секрет жетона доступа
```

```
access_token_secret = 'вашсекретжетонадоступа'
```

```
# Войдите в Twitter
```

```
tweetbot.twitter_login(cons_key, cons_secret, access_token,
access_token_secret)
```

```
# Приступайте к периодическим твитам. Здесь у нас
```

```
# по твиту в минуту. (Выбирайте любой интервал,
```

```
# но пожалуйста, не спамьте других людей.
```

```
# Спаммеров и троллей никто не любит.)
```

```
# Эта функция работает как поток в фоне,
```

```
# она не заблокирует выполнение вашего кода.
```

```
tweetbot.twitter_tweeting_start(days=0, hours=0, minutes=1,
keywords=None, prefix=None, suffix='#IamSherlocked')
```

Когда вы введете код, вам понадобится использовать клавиши со стрелками, чтобы перейти к ‘вашключ’, ‘вашсекрет’, ‘вашжетондоступа’ и ‘вашсекретжетонадоступа’ и заменить их на значения с сайта Twitter, которые вы записали раньше. Учтите, что при замене значений кавычки следует оставить на месте.

Вы также заметите под словами Конструируем путь к книге имя файла `pg661.txt`, так что вам также понадобится изменить имя вашего собственного текстового файла.

Прокрутите код вниз и обратите, что по умолчанию этот код будет твитить каждую минуту. Измените это, если хотите — например, раз в день: `days=1, hours=0, minutes=0`. И наконец, возможно, вы захотите изменить суффикс около каждого твита на что-то другое или на None, т.е. «ничего».

Нажмите `Ctrl+X`, когда закончите, затем `Y`, и `Enter`, чтобы сохранить. Вы можете запустить ваш скрипт в любое время, командой

```
sudo python sherlock1.py
```

Замените `sherlock1.py` на актуальное имя вашего файла. Терминал будет показывать твиты, но вы также сможете посмотреть их на сайте.

Есть куда больше возможных проектов Python, которые можно сделать с Twitter, включая ответ на ключевые слова, автоматические ретвиты и даже захват значений с web-страниц, например, цен на фондовом рынке. Перейдите на [www.raspberrypi.org/blog/tag/python](http://www.raspberrypi.org/blog/tag/python), чтобы увидеть несколько выдающихся проектов, сделанных на Python. LXF

### Скорая помощь

Перейдите на [www.gutenberg.org](http://www.gutenberg.org) для получения бесплатных электронных книг. Здесь есть различные книжные форматы. Убедитесь, что вы выбрали ссылку на Plain Text UTF-8 (Обычный текст UTF-8), для уверенности, что сможете прочитать его.

### Скорая помощь

Если вам просто невероятно лень, можете взять код прямо из Интернета, перейдя на <https://pastebin.com/SnJi18rd>, вот вам и скорая помощь!



Спросите  
Д-ра Бидвелла

# ОТВЕТЫ



Есть вопрос по открытому ПО? На любом уровне, обращайтесь! Мы найдем решение.

## 1 Вопросы по Qubes

В Только что установил Qubes 3.2 и обновил его, но не могу понять, как воспроизводить видео (из новостных станций, фильмов, DVD). Я понимаю, что для этого требуется специальное оборудование, но у меня есть всё: мой процессор — это Intel i7-2760QM, работающий на ноутбуке HP 8460p с SSD и 16 Гб оперативной памяти. У меня 14-летний опыт работы с Linux.

Я пользователь Linux Mint 18.3, и у меня есть всё необходимое для Qubes, но я не могу заставить видео работать. Я не могу загрузить плагин Flash, и не слишком знаком с HTML5. Я не знаю, как установить или извлечь файлы Tar. Если вы можете помочь, спасибо. Если не можете, всё равно спасибо. Я планирую дальше продолжать использовать Linux Mint — он делает всё, чего я прошу. Но у Qubes масса преимуществ.

Стивен Вуд, по электронной почте

Qubes (как Вы, несомненно, знаете), сосредоточен на конфиденциальности и безопасности. Все приложения запускаются в отдельных доменах Хел, которые не могут обращаться друг к другу и их доступ к оборудованию хоста ограничен. Т.е. Ваши мультимедиа-приложения видят не Вашу видеокарту, а только виртуальное видеоустройство, которое, надрываясь, визуализирует всё программно и управляет результатами хосту, который затем передает всё на реальную видеокарту.

Тем не менее, я бы ожидал воспроизведения локальных файлов (по крайней мере, с низким разрешением) чем-то вроде VLC для работы с программным рендерингом, особенно учитывая мощь Вашего оборудования. Вообще-то я бы ожидал, что потоковое видео из Интернета тоже будет работать, хотя и с небольшим дрожанием и заиканием. Я некоторое время не баловался с Qubes, и у меня он не установлен (он не слишком хорошо работает в виртуальной машине, по понятным причинам), так что это всего лишь предположение. Какой шаблон ОС Вы используете с Qubes? Это может быть проблемой специфики дистрибутива. Нет ли соответствующих сообщений об ошибках? Здесь Ваш друг — Google.

В настоящее время нет веских причин для установки плагина Flash (особенно если Вы выбрали Qubes ради безопасности). Почти все сайты теперь предлагают какое-либо видео HTML5, и даже контент, защищенный DRM, поддерживается Firefox из коробки. Возможно, для работы Вам не хватает какой-то библиотеки (*libav* или *ffmpeg*), но тогда всё равно надо как-то объяснить отказ DVD и локальных файлов воспроизводиться на вашем ноутбуке.



"If you're serious about security, Qubes OS is the best OS available today. It's what I use, and free."

— Edward Snowden, whistleblower and privacy advocate



"When I use Qubes I feel like a god. Software thinks that it's in control, that it can do what it wants? It can't. I'm in control."

Если это хорошо для Эда и Мика [Mikah], то, видимо, хорошо и для вас. Пока вы не захотите весь день смотреть видео про котят в высоком разрешении.\*

Я располагаю достоверной информацией о том, что в следующем выпуске у нас будет статья о конфиденциальности [Ред.: — Вот сейчас ты должен ее писать], и, надеюсь, у меня будет возможность поиграть с Qubes. Я обязательно включу любые советы по воспроизведению видео, если на что-то наткнусь.

Содержимое файла tar извлекается командой `tar xvf file.tar`. Большинство tar сжаты *gzip* (или *xz-ed*, или *bz2-ed*), чтобы уменьшить их, а tar (ленточный архив) не делает сжатия, поэтому сначала передайте архив *gunzip*, выполнив `tar xvzf file.tar.gz`. Более подробная информация доступна по `man tar`. Между прочим, я несколько озадачен: как Вы могли за 14 лет не столкнуться с этим? Налицо предупреждение об установке непроверенных двоичных файлов вместо использования менеджера пакетов.

## 2 Как уйти от EFI

В У меня с 2006 г. есть Mac Mini, который больше не получает обновления от Apple. Я надеюсь сделать эту машину снова отличной, установив Linux, но больших успехов не добился.

Я пытался загрузить на ней несколько ваших DVD, но машина просто зависает с сообщением "Select CD-ROM Boot Type". Я понимаю, что есть некоторые проблемы с загрузкой Linux из-за 32-разрядной EFI этих машин, но не могу придумать простое решение. Есть идеи?

Педро Перес, штат Пуэбла, Мексика

Небось это ты, Эффи? В 2006 г. появилось два Mac Mini, и Вы правы, оба имеют 32-битную EFI. Однако новейшая прошивка позволяет им загружаться в режиме

совместимости с BIOS, что также необходимо для улучшения видео в Linux, и Ваша первая задача — обновление прошивки.

Увы, даже это не позволит загружать мультикаталоговые образы, которые используют большинство 64-битных дистрибутивов Linux (и наши DVD). В начале 2006 г. у Mini были 32-битные процессоры Intel Core Duo, а позже — 64-битные Core 2 Duo. Если Ваша модель — начала 2006 г., то единственным решением является загрузка 32-битного дистрибутива. Для более поздней версии можно пересобрать стандартный 64-битный дистрибутив, удалив из ISO поддержку загрузки UEFI. По существу, Вы просто монтируете ISO и перестраиваете его с помощью *xorriso*, используя только загрузочный код из пакета *syslinux* (см., например, раздел об удалении поддержки загрузки UEFI на [https://wiki.archlinux.org/index.php/Unified\\_Extensible\\_Firmware\\_Interface](https://wiki.archlinux.org/index.php/Unified_Extensible_Firmware_Interface)). Вероятно, Вы захотите установить менеджер загрузки *rEFInd*, если у Вас двойная загрузка; и, возможно, понадобится преодолеть и другие препятствия. Удачи!

## 3 Argh-duino

В Привет команде Linux Format, очень надеюсь, что здесь есть какие-нибудь пользователи Arduino. Я использую Mint 18.1 Cinnamon на AMD FX-4100. Недавно я купил Arduino Micro для использования с Pi-Zero. Сначала выполнил `apt-get install arduino`, но некоторые файлы

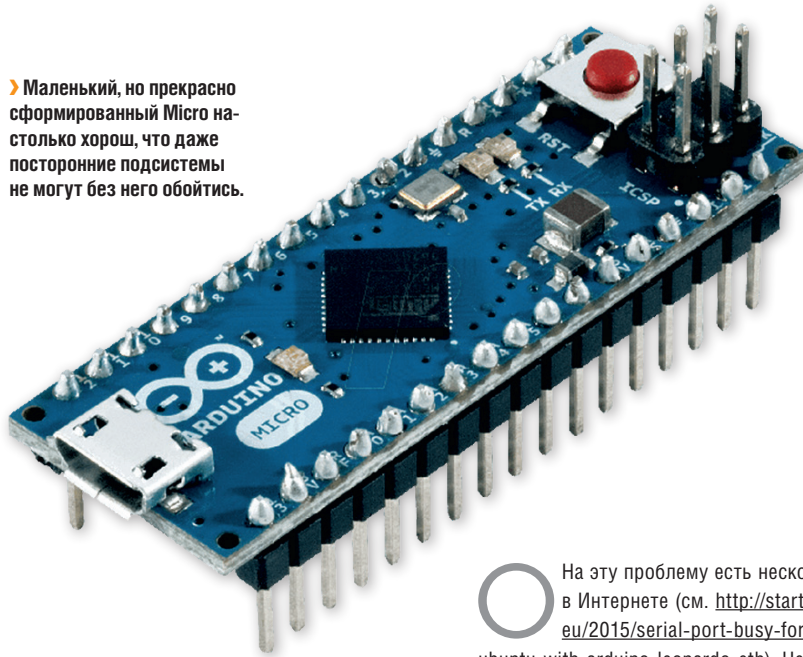
\*Перевод текста на картинке:

«Если безопасность для вас — это серьезно, то Qubes — лучшая из ныне доступных ОС. Я сам ее использую, и это бесплатно.»  
— Эдвард Сноуден [Edward Snowden], разоблачитель и защитник конфиденциальности

«Пользуясь Qubes, я чувствую себя богом. По мнению ПО, оно может всё, что захочет. Не тут-то было. Всё контролирую я.»



› Маленький, но прекрасно сформированный **MiGo** настолько хорош, что даже посторонние подсистемы не могут без него обойтись.



оказались поврежденными. Потом я дважды загрузил их с сайта Arduino, и, несмотря на следование рекомендациям по установке и устранению неполадок, я получаю следующее сообщение об ошибке при запуске простого скетча мигания:

```
avrdude: ser_open(): can't open device "/dev/
ttyACM3": Device or resource busy
An error occurred while uploading the sketch
```

Тот же результат я получил на ноутбук с Mint и старом ноутбуке под управлением XP. Marlin's допустил, что **MiGo** может быть неисправным и заменил его другим, который постыдно выдает ту же ошибку. Нативная установка не устанавливает разрешение на выполнение, но его изменение под root ничего не меняет. Я добавил своего пользователя в группу `diout`, что показывает `ls -l /dev/ttyACM3`, поэтому не думайте, что это простая проблема с разрешениями. У меня также есть Arduino Uno, который отлично работает. Был бы признателен за любые советы.

buffer, с форумов

На эту проблему есть несколько ссылок в Интернете (см. <http://starter-kit.nettigo.eu/2015/serial-port-busy-for-avrdude-on-ubuntu-with-arduino-leonardo-eth>). Некоторые люди сообщили об успехе, неоднократно перезагружая Arduino и повторно загружая скетч, но это явно неудовлетворительное решение.

То, что происходит, является конфликтом с пакетом менеджера модема для управления мобильными широкополосными (2G/3G/4G) устройствами. Демон *modem-manager* (далее *mm*) объявляет ноды устройства, поскольку предполагает, что всё подключаемое через последовательный интерфейс (ноды `/dev/tty*`) является модемом. Это мешает подключению к ним приложения *avrdude*, вызывая описанную Вами ошибку.

Пример решения упоминает добавление правила `udev`, устанавливающего переменную `ID_MM_DEVICE_IGNORE`, чтобы остановить *mm*, запрашивающего эти ноды. Вы удивитесь, но Ubuntu 16.04 и его производные (такие, как Mint 18.1) уже включают такое правило (в файле `/lib/udev/rules.d/77-mm-usb-device-blacklist.rules`). Оно использует идентификатор поставщика USB Arduino (2341) для идентификации устройства и установки

переменной среды, если он обнаружен, так что теоретически этой проблемы больше не должно существовать. И всё-таки она остается, поэтому должна быть какая-то ошибка или другой процесс, позволяющий *mm* вонзаться свои когти (Вы можете проверить, действительно ли это *mm* является виновником, подключив **MiGo** и запустив `lsusb /dev/ttyACM*`). Если Вы не пользуетесь мобильными широкополосными устройствами, то решение простое: просто удалите мешающий пакет с помощью `sudo apt remove modem-manager`. Потом его всегда можно переустановить, если возникнет такая необходимость. Если Вам действительно понадобится *mm*, то, пожалуйста, дайте нам и многим другим растерянным пользователям Arduino знать, если найдете менее жесткое решение. LXF

## Скорая помощь

Мы рады ответить на любые вопросы по Linux, которые вы отправляете на [lxf.answers@futurenet.com](mailto:lxf.answers@futurenet.com), независимо от уровня. У всех нас раньше бывали трудности, так что не стесняйтесь. Тем не менее, мы всего лишь люди (хотя есть подозрение, что Джонни — робот), поэтому важно, чтобы вы включили как можно больше информации. Если что-то работает на одном дистрибутиве, а на другом нет, расскажите нам. Если вы получаете сообщение об ошибке, пришлите нам точное сообщение и что конкретно вы сделали для его вызова.

Если у вас есть проблемы с оборудованием, или вы их подозреваете, опишите нам это оборудование. Рассмотрите возможность установки *Hardinfo* или *Ishw*. Эти программы перечисляют оборудование на вашей машине, поэтому отправьте нам их вывод. Если вы не хотите или не можете их установить, запустите следующие команды в терминале root и отправьте нам файл `system.txt`.

```
uname -a > system.txt
lspci >> system.txt
lspci -vv >> system.txt
```



## Праздничные сказки о воскресении

В У меня было четыре настольных ПК с Ubuntu, но Санта не принес мне Linux-ноутбук на Рождество. Затем я вспомнил, что с год назад старый друг дал мне битый (ну, на самом деле — малость поцарапанный) ноутбук Dell Studio под управлением Windows 7. Потребовалось несколько минут, чтобы он загрузился и был переполнен множеством раздражающих всплывающих окон. Я использовал DVD, поставляемый с LXF231, для тестирования 32-разрядного Lubuntu 17.10 на этом ноутбуке Dell и был поражен скоростью работы.

У меня было только две проблемы, которые теперь успешно преодолены. Во-первых, я обнаружил, что после форматирования жесткого диска и удаления старой ОС удалить DVD из слота ноутбука довольно сложно. Так или иначе, мне удалось установить Lubuntu

с прилагаемого DVD, и я впечатлен результатами. Вторая проблема была в том, что Wi-Fi на ноутбуке перестает работать. Я позаимствовал адаптер Wi-Fi от умного телевизора и подключил его к порту USB ноутбука. Это позволило мне загрузить драйвер для внутренней карты Broadcom Wi-Fi, а затем (радость радостей!) я увидел, что светодиод Wi-Fi на клавиатуре зажегся. У меня теперь есть ноутбук Linux, которого я жаждал, но его принес не Санта. Мне нравится минималистский рабочий стол Lubuntu и узкая панель инструментов по нижнему краю, и что все приложения удобно расположены по категориям. Мне больше не надо прокручивать меню Ubuntu!

Итак, позвольте мне поблагодарить хороших людей в журнале *Linux Format* (и не забыть моего друга, который подарил свой старый

поврежденный ноутбук). Вы все осчастливили 70-летнего мужчину.

Пит, по электронной почте

Спасибо, Пит. Рады, что сумели внести праздничное настроение. Этот Санта должен за многое ответить. В старые времена повсеместно используемые оптические приводы имели механизм аварийного извлечения, который можно было задействовать тычком скрепки. Приводы со слотовой загрузкой обычно этим не заморачиваются, а значит, надо загрузить ОС, чтобы извлечь диск, и это, как Вы обнаружили, привело к некоторым неприятным ситуациям «Уловки 22». Поскольку основные дистрибутивы отходят от поддержки 32 бит, будет интересно посмотреть, как развиваются варианты вроде Lubuntu и Bodhi.



# LXF Hot Picks



**Александр Толстой**

**Darktable » Ddgr » Midnight Commander » Flameshot » Liquidshell » Posterizer » Android File Transfer » Catimg » Desktopfolder » Bemuse » HexGL**

Ему надоел снег в Санкт-Петербурге, но он утешается отличными программами с открытым кодом!

## Фоторедактор

# Darktable

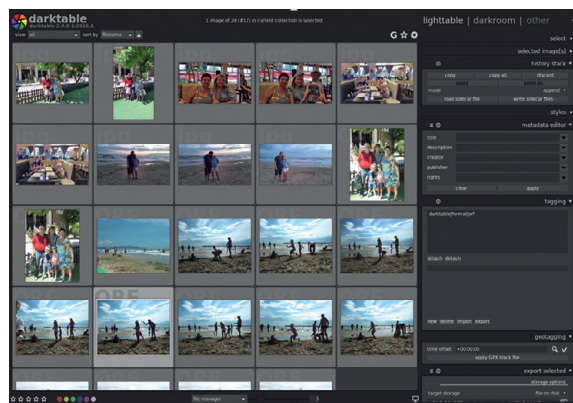
Версия: 2.4 Сайт: [www.darktable.org](http://www.darktable.org)

Прошло больше года с выхода последнего стабильного обновления *Darktable* (более подробный рассказ вы обнаружите в **LXF235**), так что мы были рады сделать новый обзор этого выдающегося представителя мира открытого кода.

*Darktable* — это профессиональный инструмент для обработки RAW-снимков, сделанных зеркалкой или камерой другого типа, способной делать снимки в формате RAW. Вне экосистемы Linux *Darktable* конкурирует с такими коммерческими тяжеловесами, как *Lightroom* от Adobe и *Aperture* от Apple, и одно это уже подтверждает, что Linux в целом достаточно зрелый, чтобы

отвечать требованиям профессиональных фотографов.

Рендеринг изображений RAW и выполнение таких задач, как предпросмотр эффектов live, требует большой мощности CPU. К счастью, *Darktable* использует сложный оптимизированный код для повышения производительности. В качестве примера мы решили посмотреть очень большое растровое изображение — ночной



» *Darktable* скрывает под личиной скромного браузера изображений колоссальную мощь.

## Darktable — инструмент обработки снимков RAW зеркалкой.

вид Земли от NASA. Это массивный JPEG-файл, весом 220 МБ. Большая часть просмотрщиков изображений в Linux не смогла его открыть, даже на системах с 8 Гб ОЗУ, но *Darktable* справился с этой задачей быстро, даже в конфигурации низкого уровня.

Так что же нового в версии 2.4 *Darktable*, помимо почти 3000 фиксаций изменений и более 340 решенных проблем? Самый последний релиз предлагает новый инструмент Haze Removal, полезный для улучшения качества снимков и исправления эффекта грязных линз. Модуль Local Contrast теперь предлагает режим Local Laplacian, а Undo history теперь поддерживает маски. Полностью поддерживаются также сжатые файлы RAF Fujifilm и HDR DNG с плавающей точкой.

Есть еще множество смачных новых дополнений, и если вы регулярно занимаетесь обработкой RAW-фотографий, вы уж точно заметите внутренние улучшения. Такие функции, как группирование в Undo history, инструмент Tone Curve, фильтр de-mosaicing, инструмент определения на карте и прочие — все они были улучшены для обеспечения лучшего и более эффективно взаимодействия с пользователем.

Скачайте самую свежую версию *Darktable* с сайта проекта или проверьте репозитории своего дистрибутива на предмет обновлений.

## Проливая свет на интерфейс Darktable

### Свет и тьма

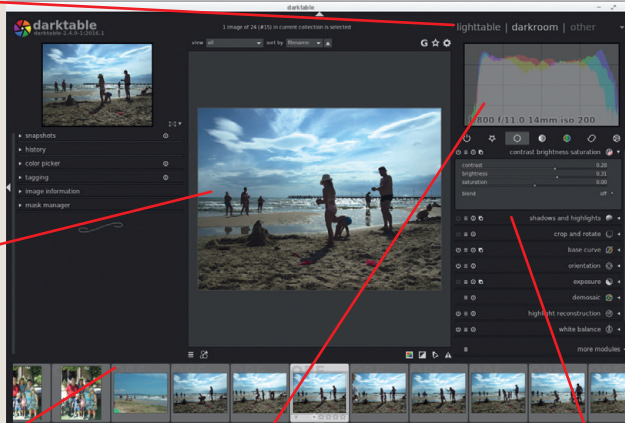
*Lightroom* [Светлая комната] — это место, где вы просматриваете и сортируете изображения. *Darkroom* [Темная комната] — для редактирования выбранных изображений.

### Предпросмотр Live

*Darktable* имеет быстрый движок рендеринга, который буквально на лету применяет все ваши изменения.

### Еще больше всего под рукой

Приглядывайте за другими изображениями, не покидая *Darktable*.



### Исправление цвета

Посмотрите на цветную гистограмму и настройки камеры, использованные при создании снимка.

### Исправление и улучшение

Применяйте разные цветовые инструменты, чтобы исправить яркость, тени или контраст ваших изображений.





Инструмент создания скриншотов

# Flameshot

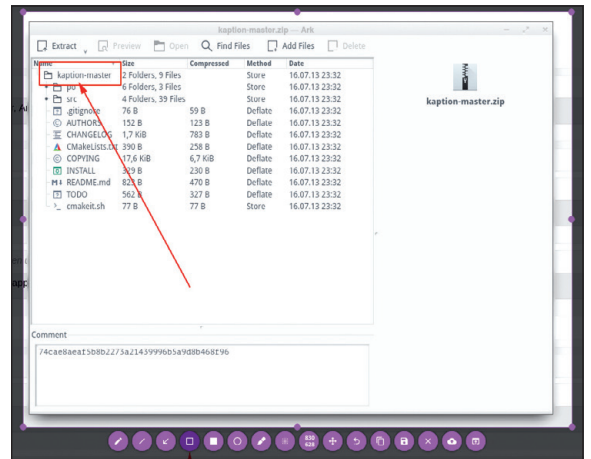
Версия: 0.5.0 Сайт: <http://bit.ly/flameshot>

Да ни, когда рабочий стол Linux предлагал выбор примитивных инструментов для создания скриншотов, давно миновали. Если нас сегодня спросят, на какой платформе самый богатый и полный набор программ создания скриншотов, мы просто с улыбкой укажем на симпатичного пухленького Тукса [Tux]. В подкрепление наших выводов, вот вам еще один соперник *Ksnip* (LXF223) и *Screencloud* (LXF226), бывших фигурантов *HotPicks*.

*Flameshot* — отличный инструмент для снимков отдельных областей рабочего стола, и по этой части намного функциональнее, чем просто замена клавиши Print Screen. В приложении имеется несколько достойных внимания интересных функций. Во-первых, *Flameshot* создает аж три пункта в меню: один для настройки приложения, еще один для запуска его в немом режиме в системном лотке, и еще одну функцию запуска для запуска съемки напрямую.

*Flameshot* предполагает, что вам, возможно, захочется изменить выбор инструментов аннотирования по умолчанию, настроить шаблон наименования файлов с помощью обычных переменных (дата, время и т.д.) и настроить цвета самого инструмента.

Выбор инструментов аннотирования очень хорош, хотя и не идеален. *Flameshot* не умеет размещать меток с автоматической нумерацией, но дает возможность рисовать линии, стрелки, овалы и прямоугольники. Инструмент размывки означает, что вы можете скрыть на своих снимках неуместную информацию, и есть также плагин отправки в облако — *Imgur*. Нам понравилась скорость и простота



➤ Рамки и эти круглые кнопки остаются даже после того, как вы что-то изображали.

в использовании этой функции для общего доступа к скриншотам. Хотя *Imgur* является единственной облачной опцией в *Flameshot*, он весьма неплохо справляется с задачей публикации изображений. В порядке альтернативы, вы всегда можете сохранить свой снимок как локальный файл в любом файловом формате (предлагаемом плагинами изображений *Qt5*) или открыть его в другом приложении.

И в заключение — *Flameshot* предлагает впечатляющий набор переводов и экспериментальную поддержку *Wayland* для рабочих столов *Plasma 5* и *Gnome*.

**Намного функциональнее, чем просто замена клавиши Print Screen.**

Оболочка рабочего стола

# Liquidshell

Версия: Git Сайт: <https://github.com/KDE/liquidshell>

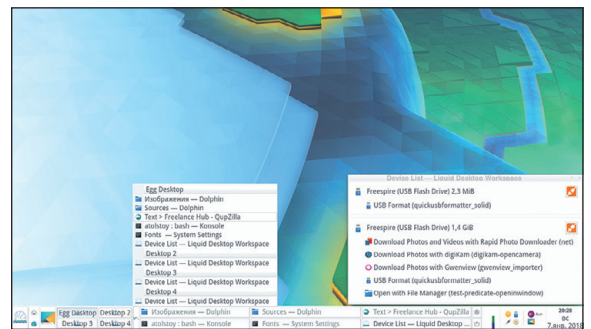
Мы полагаем, что на какой бы графической библиотеке приложение ни основывалось, оно должно хорошо работать в любой среде рабочего стола по вашему выбору. Такие современные рабочие столы, как *Gnome*, *KDE Plasma*, *Mate* и *Cinnamon*, могут сделать встроенными приложения и на базе *GTK*, и на *Qt*.

Однако именно это приложение более привязано к рабочему столу, поскольку пытается заменить процесс 'plasmashell' более легкой альтернативой. *Liquidshell* — альтернативная оболочка рабочего стола, предназначенная для работы с типовым рабочим столом *KDE* (она официально размещается в проекте *KDE*, как вы, возможно, смекнули по адресу его страницы на *GitHub*).

*Plasmashell* не слишком сильно нагружает ресурсы, но *Liquidshell* работает еще быстрее, в основном благодаря более простому дизайну и отказу от использования декларативного языка *Qt*, *QtQuick*.

*Liquidshell* — простая оболочка, построенная из классических управляющих элементов *QtWidgets*, включая нижнюю панель, где размещены системный лоток, часы, меню запуска, набор виртуальных столов и прочие удобные штуки. Она запускается практически мгновенно и очень проста в использовании.

Хотя для компиляции *Liquidshell* из исходника понадобится подборка разных частей разработки *KDE Frameworks 5*, основное препятствие заключается в ее запуске. На сей раз у вас отсутствует отдельный пункт «сессия» в окне приветствия. Вместо этого надо добавить *Liquidshell* в список автозапуска и затем загнать в тень обычный процесс *Plasmashell*. Для этого поместите файл со следующими двумя строками



➤ Если рабочий стол *Plasma* по-прежнему сбивает, выбросьте его и используйте вместо него *Liquidshell*!

в качестве еще одного пункта автоматического запуска:

```
[Desktop Entry]
Hidden=true
```

При следующем входе *Liquidshell* будет работать поверх *Plasmashell*, и вы даже сможете полностью ее остановить (*killall plasmashell*). В новой оболочке имеются ссылки на ваши приложения по умолчанию и системный лоток, где перечислены устройства и активные программы.

Мы отлично провели время с *Liquidshell*. Вся работа похожа на солидную сессию *KDE* или даже на полнофункциональный релиз *LxQt* из будущего!

**Простая оболочка, построенная из классических QtWidgets.**



## Программа оптимизации изображений

## Posterizer

Версия: Git Сайт: <http://bit.ly/lxf-posterizer>

Ранее мы довольно много писали о разных кодировщиках, оптимизаторах и хитростях для уменьшения объема вашей библиотеки изображений без ущерба качеству. Судя по всему, новых инструментов для работы без потерь пока нет, но мы всё же можем взорвать ваш мозг весьма неплохими приложениями для сжатия с потерей качества.

Недавно мы наткнулись на *ImageAlpha*, графическую программу исключительно для MacOS, которая может уменьшить ваши PNG-файлы, особенно если у них есть альфа-канал (прозрачность). Это приложение использует инструмент командной строки под названием *Mediancut-Posterizer* — или, для краткости, *Posterizer*. Как пишет автор, у него два режима: усредняющий фильтр с потерями (*blurizer*), который убирает из изображения шум; и оптимальная постеризация, использующая квантование по методу медианного сечения [Median Cut] для уменьшения количества уникальных

цветов в изображении с минимальным визуальным искажением. Мы провели наши тесты, и обнаружили, как этот инструмент повлиял на истинные цвета наших PNG с прозрачностью. Самое обычное использование, вероятно, выглядело так:

```
$ posterizer -Q 75 in.png out.png
```

Программа сократила размер нашего тестового изображения на целых 30% без каких-либо видимых артефактов. Поиграв с коэффициентом сжатия (-Q), мы выяснили, что даже при очень низких цифрах *Posterizer* дает весьма достойные результаты, и даже хотя изображение уже не выглядит ровным, оно занимало менее половины исходного размера файла. *Posterizer* можно очень быстро превратить в 'blurizer',

**Программа сократила размер изображения на целых 30%.**



➤ Одно из этих изображений было облегчено на 50% с помощью *Posterizer*. Вроде бы то, которое справа.

присоединив параметр `-b` и опять же поиграв с коэффициентом сжатия. Усреднение с потерей качества не сокращает размер файла немедленно (файл на выходе был больше, чем на входе), но суть в том, что усреднение само по себе просто готовит материал для будущего сжатия. Вы можете заново экспортировать размытое изображение PNG в *GIMP*, и посмотреть, насколько еще можно уменьшить размер файла.

Есть масса вариантов извлечь пользу из *Posterizer*, кроме оптимизации сайтов. Так, Raspberry Pi и многие маломощные устройства IoT лучше ладят с более мелкими значками PNG и иным PNG-контентом.

## Инструмент передачи файлов

## Android File Transfer

Версия: Git Сайт: <http://bit.ly/android-file-transfer>

Большинство современных смартфонов снабжено достойной камерой (что и вызвало сужение рынка собственно фотокамер). И даже несмотря на то, что многие из нас загружают свои снимки со смартфона на всевозможные облачные сервисы, всё равно есть потребность периодически скачивать изображения на компьютер.

Телефоны Android больше не используют классический тип подключения USB для массового хранения, отказавшись от него в пользу MTP (Media Transfer Protocol), и хотя последний поддерживается всеми основными настольными дистрибутивами Linux, со стабильностью MTP в Linux всё же налицо некоторые проблемы (по крайней мере, в *Dolphin*). Мы рассматривали способы передачи директории DCIM с устройства Android на наш компьютер Linux, пока не наткнулись на *Android File Transfer*. Это отличный отдельный инструмент GUI, который решает все проблемы, связанные с поддержкой MTP в Linux.

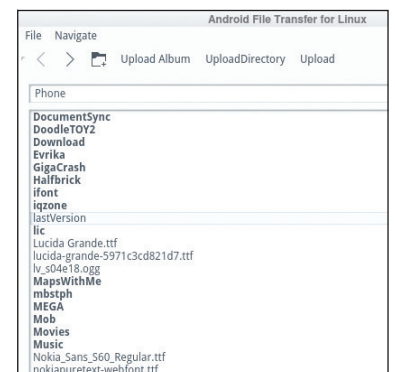
*Android File Transfer* запустится только если вы уже подключили свой телефон к компьютеру Linux через кабель USB. А если нет, вы увидите только предупреждение, что программа не сумела найти устройство MTP.

Главное окно инструмента довольно простое, и, возможно, некоторым покажется чересчур простым для менеджера файлов. Но помните, что *Android File Transfer* — всего лишь инструмент копирования файлов, а не полнофункциональный менеджер файлов. Он позволяет просматривать структуру файлов вашего устройства и выбирать, какой контент скачать.

Вы также можете загрузить некоторые файлы на свой телефон, или удалить там существующие файлы и директории.

**Решает все проблемы, связанные с поддержкой MTP в Linux.**

➤ Легко передавайте файлы со своего смартфона Android на полной скорости.



Итак, во многих отношениях *Android File Transfer* имитирует набор функций *Rapid Photo Downloader* с некоторыми важными отличиями. Здесь не существует эскизов для предпросмотра изображений и видео, но вы можете работать с файлами на своем телефоне.

С нашей точки зрения, основное преимущество *Android File Transfer* — его чрезвычайно надежная поддержка MTP. Пускай это может показаться очевидным, но в реальной жизни такие штуки, как, например, GMTP, GVFS и MTPS, отнюдь не застрахованы от ошибок, и периодически вы сталкиваетесь с зависшими или поврежденными файлами. Использование *Android File Transfer* весьма элегантно решает эту неприятную проблему!

## HotGames Развлекательные приложения

## Музыкальная игра

## Bemuse

Версия: v38 Сайт: <http://bit.ly/lxf-bemuse>

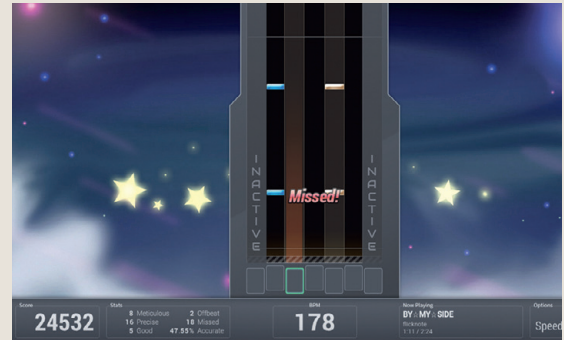
**В**emuse — смесь космической стрельбы с вертикальной прокруткой и ритмической музыкальной игры: надо нажимать на ноты при их появлении на экране. Вы можете использовать разные внешние устройства, например, MIDI-клавиатуру и вертушки, но клавиатуры компьютера тоже вполне хватит. Умение играть на пианино будет полезным, но даже если вы не умеете, Bemuse поможет вам обрести силу в пальцах и запястьях.

Чтобы хорошо выступить (и набрать побольше очков), надо ускорить время реакции и привыкнуть к нетипичной раскладке клавиш. Bemuse запускается с руководства для новичков, где можно научиться, как и когда использовать семь клавиш (S, D, F, пробел, J, K и L), и насладиться предлагаемыми треками в стиле электроникор и транс. В зависимости от ваших результатов Bemuse ставит

вам оценки, от F (так себе) до A (ниндзя!) и приглашает вас поучаствовать в онлайн-рейтинге Bemuse. Эта функция доступна только если вы играете в Bemuse онлайн, потому что локальная установка не позволит вам загрузить ваши результаты.

Bemuse предлагает достойную подборку песен по умолчанию, и вы можете подключить песни на свое усмотрение. Игра принимает BMS-файлы, их можно изучить на <http://bmssearch.net> (правда, сайт на японском). Мы потратили несколько часов, пытаясь перейти на следующий уровень с парочкой танцевальных треков Bemuse, и нам отнюдь не было скучно!

**Bemuse включает подборку современных web-технологий.**



► Интересная игра, которая сочетает в себе музыку и время реакции.

Bemuse включает подборку современных web-технологий, таких как React и HTML5, и очень просто разворачивается на вашем локальном хосте. Надо установить Node.js и Yarn всего лишь двумя командами: \$ yarn и \$ npm start.

Не забудьте дать отдых пальцам после долгой игры!

## Игра-гонки

## HexGL

Версия: Git Сайт: <http://bit.ly/hex-gl>

**В**наши дни можно наслаждаться высококачественными 3D-играми прямо в окне браузера. И Sourceforge, и Github предлагают отличные игры на HTML5, и некоторые из них выводят игру в браузер на новую высоту.

HexGL — одна из таких игр. Это футуристичный симулятор гонок с потрясающей графикой — и в то же время отличная тестовая площадка для проверки WebGL-возможностей вашего браузера. Хотя сама игра не очень велика, ей нужна хорошая производительность OpenGL вашего видеодрайвера, иначе вам придется запустить ее на низких настройках детализации.

В HexGL вы управляете реактивным самолетом, движущимся по определенному маршруту (хотя самолет летит над этим маршрутом, он не может с него сойти). В игре предусмотрен очень динамичный однопользовательский режим:

просто пройдите все три кольца как можно быстрее. Маршрут проложен в урбанистическом пейзаже с небоскребами и живописными видами сверху. Изначально у вашего самолета показатель надежности — 100, но при каждом столкновении с треком он получает повреждение. И если вы не будете лететь плавно, вы можете просто уничтожить свой самолет, не дойдя до финиша! На треке есть точки ускорения, которые разгоняют ваш полет, но могут привести к повреждению, если вы ускоритесь на крутом повороте.

Окно меню HexGL предлагает базовые настройки, которые вы, возможно, захотите изменить перед началом игры.

**Футуристичный симулятор с потрясающей графикой.**



► Не забудьте нажать на тормоз перед крутыми поворотами и используйте ускорение разумно!

Самая важная — это качество графики: по умолчанию оно настроено на Очень высокое [Very High], и если ваш компьютер не соответствует, то игра просто не запустится. Нажмите F5, чтобы перезагрузить страницу, и установите качество на Высокое [High], чтобы это исправить.

Запустить игру несложно, если вам не надо возиться со множеством зависимостей. HexGL использует пакет Python SimpleHTTPServer, и все, что вам надо сделать — выполнить команду \$ python -m SimpleHTTPServer и открыть index.html в своем браузере.



## Просмотрщик изображений

## Cating

Версия: 2.4.0 Сайт: <https://github.com/posva/cating>

У нас есть еще одно сокровище, которое сделает вашу ежедневную сессию в консоли Bash ярче и красочней. И дело не в укрощении вашего `~/.bashrc` и не в игре с эффектами псевдографики ASCII, но во влиянии на способ показа изображений в терминале.

*Cating* — простой инструмент, который сэкономит вам немного времени, когда вам надо просмотреть контент изображения, но не хочется запускать для этого специальную программу. В названии обыгрывается слово 'a cat' (кот) и знакомая команда UNIX `$ cat`.

Короче, без лишних проволочек давайте посмотрим, как она работает:

```
$ cating /path/to/file
```

*Cating* использует для конвертирования изображений *ImageMagick* и может работать со множеством форматов изображений. Рендеринг происходит прямо внутри терминала, и изображения вовсе не выглядят псевдографикой, потому что *Cating* обчисляет каждый пиксель как цветной

квадрат. Качество зависит от физического размера изображения, и вы должны это осознавать. Большие изображения выглядят невероятно детализованными, хотя и кажутся чрезмерно увеличенными в масштабе.

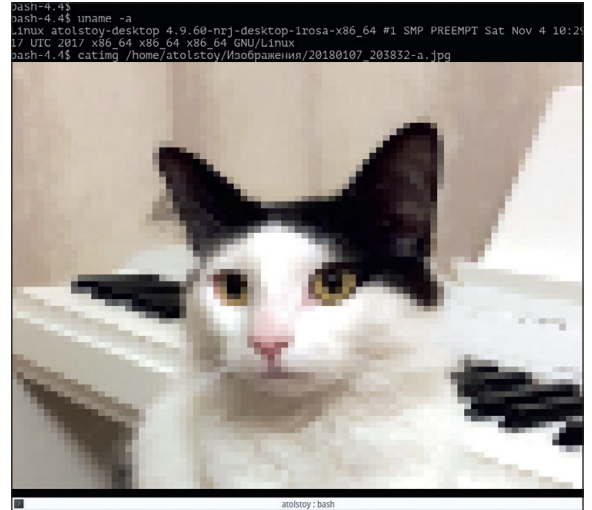
Вы можете открывать изображение в меньшем размере, используя параметр `-w`:

```
$ cating -w 200 /path/to/file
```

Эта команда предполагает, что ширина вашего терминала — 200 пикселей, и если ваше изображение шире, его масштаб будет уменьшен. Оно также будет обрабатываться на меньшем разрешении, потому что размер псевдопикселей терминала постоянный.

Лучше всего *Cating* работает в графическом режиме. Если вы попробуете его без *X11*

**Cating обчисляет каждый пиксель как цветной квадрат.**



Использование *Cating* дает вам еще один повод посидеть в терминале чуть подольше.

или *Wayland*, он напечатает очень грубую версию вашего изображения, с узнаваемым силуэтом, но искаженными цветами. Рекомендуется графический режим с полным разрешением, так как он обеспечивает *Cating* адекватной палитрой. С другой стороны, просмотр небольших графических файлов, таких как значки, аватарки, эскизы, идеально подходит для *Cating*. Вы можете увидеть все подробности и продолжать использовать сессию терминала.

## Аксессуар рабочего стола

## Desktopfolder

Версия: 1.0.5 Сайт: <http://bit.ly/desktopfolder>

Поскольку *Liquidshell* ориентирован на KDE, мы решили рассказать еще об одной похожей программе.

*Desktopfolder* предназначен для *elementaryOS*, или, если быть точнее, для ее рабочего стола *Pantheon*. Пуристы могут сказать, что *Pantheon* также доступен в Arch и Fedora, но всё же большинство пользователей предпочитают запускать его в его «родной» eOS.

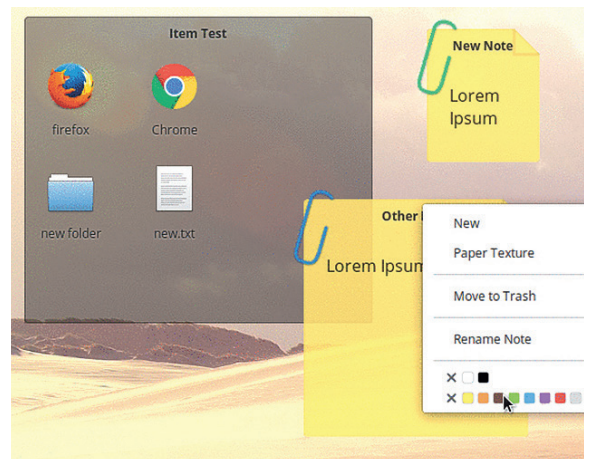
Хотя на одном из наших тестовых компьютеров работает *elementary OS* и мы используем ее довольно часто, бывает непросто сделать обзор стильных интересных приложений из нового *AppCenter elementary*, просто потому что они слишком 'elementary' и не предлагают достаточно функций, достойных рассмотрения.

К счастью, это не относится к *Desktopfolder*, крутому дополнению к *Pantheon*, которое возвращает свой рабочий стол к жизни. Как вы, возможно, знаете, рабочий стол *Pantheon* преднамеренно оставлен пустым,

без каких-либо средств разметить на нем файлы или значки. Если вас не устраивает подобное положение дел, решением будет *Desktopfolder*. Это приложение создает плавающие панели, которые служат контейнерами. Вы можете поместить в них файлы, папки, программы запуска, изображения, и многое еще.

*Desktopfolder* обеспечивает достаточно параметров для настройки панелей под разные нужды, и в конечном итоге у вас есть несколько панелей разного типа: контейнер файлов, заметка-наклейка или рамка изображения. Он работает подобно базовому набору плазмоидов, только в данном случае привязан к набору инструментов *elementary OS* и написан на Vala.

**Дополнение к Pantheon возвращает рабочий стол к жизни!**



Виджеты в материальном стиле поверх рабочего стола elementary OS? Звучит многообещающе...

Нам очень понравилось возиться с панелями *Desktopfolder* — особенно изменять размер, добавлять цветные теги и перетаскивать в панель контент.

Сама концепция противоречит идеалам *elementaryOS*, но это же Linux, так что наша свобода выбора никуда не делась. *Desktopfolder* — проект с открытым кодом, который предлагает прекомпилированные Deb-пакеты и устанавливается парой щелчков через *AppCenter*. Если вы используете *Pantheon*, не пропустите его! **LXF**

# На диске

Дистрибутивы, приложения, игры, книги и всякое-разное...

Лучшее из Интернета, упакованное в 8 ГБ качественного DVD.



## Дистрибутивы

Главная статья этого месяца была посвящена началу работы в Linux. Между Linux и другими ОС есть много различий, к которым надо привыкнуть, но одно из основных — богатство выбора. От сотен разных дистрибутивов через десятки разных рабочих столов к бесчисленным браузерам, почтовикам, редакторам и т.д.

Так что же выбирать? Дистрибутивы обычно включают в установку по умолчанию только то, что считают хорошими программами, поэтому всё, что предлагает ваш дистрибутив, уже неплохой выбор; но всегда есть альтернативы. Не поддавайтесь соблазну отказаться от своего дистрибутива только из-за того, что вам не нравится его программа электронной почты или цвет рабочего стола. Природа открытого кода означает, что установка дистрибутива может быть лишь отправной точкой, и это многим кажется разумным. Ваш дальнейший путь целиком зависит от вас.

Если вы — новичок в Linux, предлагаю вам для начала выбрать дистрибутив и придерживаться его и его программ по умолчанию. Когда у вас появится представление о том, что вам нравится, а что — нет, начинайте исследовать программные репозитории дистрибутива. Вы даже не представляете, какие сокровища там скрываются!

*Neil*

## » Важно ВНИМАНИЕ!

### Порченные диски

В маловероятном случае какого-то дефекта вашего LXF DVD обращайтесь, пожалуйста, по адресу [disks@linuxformat.ru](mailto:disks@linuxformat.ru) или телефону +7 (812) 309-0686.

## Идеальный старт в KDE

# ROSA Desktop R10

На следующей странице у нас два легковесных дистрибутива, так что уравновесим их чем-то потяжелее на этой. Когда речь заходит о полнофункциональных интегрированных рабочих столах, выбор обычно сводится к Gnome или KDE. Вероятно, на наших DVD было больше дистрибутивов с Gnome (особенно сейчас, когда Ubuntu вернулся к Gnome), поэтому давайте сравняем счет с помощью рабочего стола KDE [Ред.: — Это действительно очень сбалансированный DVD, попробуйте его!].

ROSA Fresh — российский дистрибутив с индивидуально настроенным рабочим столом KDE. Поскольку мы здесь немножко покритикуем KDE, спешим заявить, что мы — вполне довольные пользователи KDE, и ярые фанаты KDE могут не обрушивать на нас свою ярость. Мы полагаем, что KDE — хороший рабочий стол в поисках достойной конфигурации: по умолчанию она ужасна.

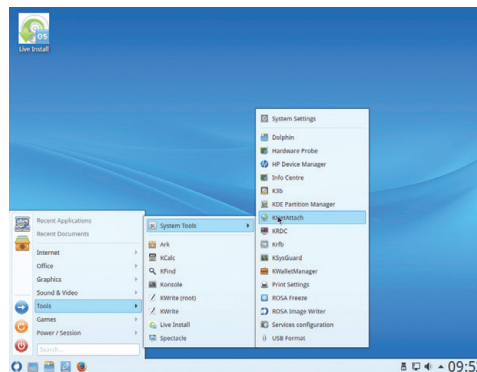
KDE может настраиваться до бесконечности, и ROSA по максимуму это использует, создавая плавное и приятное взаимодействие с KDE, которое образует куда лучшую стартовую точку, чем оригинал. Но тем не менее, это стартовая точка. Воспользовавшись настройками KDE по умолчанию, вы промахнетесь. Также предлагается версия с рабочим столом LXQt, но это — его флагманский релиз KDE. Желая поглядеть на LXQt, вы всегда можете установить его из репозитория ROSA.

ROSA предназначен для более опытных линуксоидов. Но вы не обязаны быть мастером командной строки или этаким джедаем Linux, и ROSA определенно не закапывается в такие глубины, как Gentoo; однако

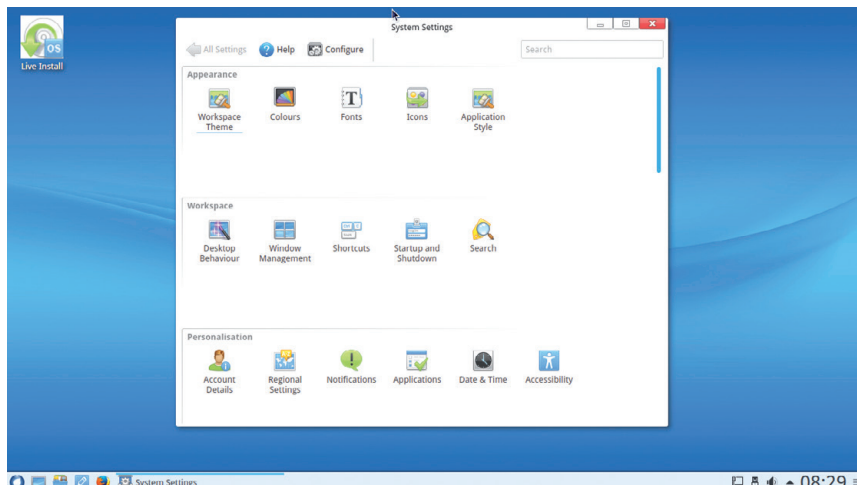
не ждите, что вас будут водить за ручку. Это дистрибутив для пользователей Linux, которые хотят попробовать что-то новое, с очень свежими версиями самых новых программ.

У ROSA хорошая родословная, поскольку это ответвление Mandriva. Mandriva и Mandrake были самыми популярными домашними дистрибутивами для энтузиастов всех уровней, пока их место не занял Ubuntu. Итак, у вас солидная база, современные программы и продуманная конфигурация... чего еще желать? И на wiki имеется полезная информация, которая поможет извлечь из него максимум.

Аутентификационная информация: имя пользователя — **live**; пароля нет.



» Поднастроенный рабочий стол KDE с более традиционным внешним видом — именно то, что нужно многим пользователям, если судить по успеху Linux Mint.







# Пропустили номер?

**LINUX**  
FORMAT  
Главное в мире Linux

Закажите его на сайте [www.linuxformat.ru](http://www.linuxformat.ru) в «ГНУ/Линуксцентре»!  
Журналы доставляются и в печатной, и в электронной форме,  
так что получение нужного вам выпуска LXF может занять всего  
пару минут с момента открытия браузера!

## LXF233 Февраль 2018



- » **ПК своими руками** От железа до софта
- » **Android на Linux** Выбираем эмуляторы, пишем приложения
- » **Едет, едет почта** Почтовые клиенты живы
- » **QR-код с пингвином** Или с другой картинкой

**LXFDVD:** Fedora, Linux Mint, Netrunner, OpenMediaVault, Tails, 11 книг о Linux (на английском языке), горячие новинки и прочее...

**Печатная версия:** [shop.linuxformat.ru/lxf\\_233/](http://shop.linuxformat.ru/lxf_233/)  
**PDF-версия:** [shop.linuxformat.ru/elxf\\_233/](http://shop.linuxformat.ru/elxf_233/)

## LXF232 Январь 2018



- » **32-битные дегковесы** Лебединая песня?
- » **Gnome и его друзья** Дистрибутивы-гномофилы
- » **Linux сладкозвучный** Аудио и еще раз аудио
- » **Скажи пароль** Ни за что не говорите!

**LXFDVD:** AntiX, Debian, Legacy OS, Linux Lite, Manjaro, Peppermint, Tiny Core, Zorin OS, 11 книг о Linux (на английском языке), горячие новинки и прочее...

**Печатная версия:** [shop.linuxformat.ru/lxf\\_232/](http://shop.linuxformat.ru/lxf_232/)  
**PDF-версия:** [shop.linuxformat.ru/elxf\\_232/](http://shop.linuxformat.ru/elxf_232/)

## LXF231 Декабрь 2017



- » **Заметки анималиста** Творческий Трубказуб!
- » **Шеренга столов** Работай — не хочу
- » **Доктор вернулся** В формате импортозамещения
- » **Точки доступа** Обезопасим с OpenVPN

**LXFDVD:** Lubuntu, Q4OS, RevengeOS, Robolinux, Ubuntu, 12 книг о Linux (на английском языке), горячие новинки и прочее...

**Печатная версия:** [shop.linuxformat.ru/lxf\\_231/](http://shop.linuxformat.ru/lxf_231/)  
**PDF-версия:** [shop.linuxformat.ru/elxf\\_231/](http://shop.linuxformat.ru/elxf_231/)

## LXF229/230 Ноябрь 2017



- » **Raspberry Pi как телохранитель** Защищает нашу сеть
- » **Шифруемся** Иначе теперь не проживешь
- » **Немного о сборках** Доведем только воспроизводимым
- » **Пакеты будущего** Независимо от дистрибутивов

**LXFDVD:** ArchLabs, Backbox, Bodhi Legacy, Sparky, LinHES, OpenELEC PC и RPi, Rescatux, 12 книг о Linux (на английском языке), горячие новинки и прочее...

**Печатная версия:** [shop.linuxformat.ru/lxf\\_229-230/](http://shop.linuxformat.ru/lxf_229-230/)  
**PDF-версия:** [shop.linuxformat.ru/elxf\\_229-230/](http://shop.linuxformat.ru/elxf_229-230/)

## LXF228 Октябрь 2017



- » **Linux в школе** Учиться будет весело!
- » **Синхронизация файлов** Без третьих сторон
- » **Спасите наши данные** Инструментом хирурга
- » **Брандмауэр для «чайников»** Естественно, на Raspberry Pi

**LXFDVD:** CAINE, Fedora 26 Gnome, Fedora 26 LXQt, Mageia Xfce, 12 книг о Linux (на английском языке), горячие новинки и прочее...

**Печатная версия:** [shop.linuxformat.ru/lxf\\_228/](http://shop.linuxformat.ru/lxf_228/)  
**PDF-версия:** [shop.linuxformat.ru/elxf\\_228/](http://shop.linuxformat.ru/elxf_228/)

## LXF227 Сентябрь 2017



- » **Польза виртуальности** Машины в машине
- » **Бизнес-серверы** Даже на дому
- » **Ноутбук с Linux** Как не купить кирпич
- » **Биткойны и блокчейны** Станьте эмитентом

**LXFDVD:** ClearOS, Debian, Koozali, NethServer, Voyager, Zentyal, 12 книг о Linux (на английском языке), горячие новинки и прочее...

**Печатная версия:** [shop.linuxformat.ru/lxf\\_227/](http://shop.linuxformat.ru/lxf_227/)  
**PDF-версия:** [shop.linuxformat.ru/elxf\\_227/](http://shop.linuxformat.ru/elxf_227/)

Подпишитесь на печатную версию журнала на [www.linuxformat.ru/subscribe](http://www.linuxformat.ru/subscribe)  
или [www.linuxcenter.ru](http://www.linuxcenter.ru), и получите электронную версию в подарок!

Телефоны отдела подписки

- » Санкт-Петербург: (812) 309-0686
- » Москва: (499) 271-4954



Linux Format ВКонтakte:  
[vk.com/linuxform](https://vk.com/linuxform)



# » Содержание

**Linux**  
формат

## Страница 17

### ДИСТРИБУТИВЫ

**Void Linux** (32-битный)  
Перовский дистрибутив на базе Ubuntu, с рабочим столом Moksha на базе Enlightenment-17. Предлагает строго минималистский подход, модульность, высокую настраиваемость и выбор тем.

**Elementary OS** (64-битный)

Настольный дистрибутив на базе Ubuntu, сделанный своим приоритетом простоту. Предлагает множество готовых приложений, состав которых можно оптимизировать из репозитория Ubuntu и Debian.

**Rosa Desktop R10** (64-битный)

Дистрибутив с глубокой настройваемым рабочим столом KDE и несколькими модификациями для повышения удобства работы. Предназначен для продвинутых пользователей.

### СРАВНЕНИЕ: МЕНЕДЖЕРЫ ФАЙЛОВ

**Dolphin 17.08.1**  
files/dolphin\_kde.lxd  
**GNOME Files 3.26.0**  
file/duopen-files  
**SpaceFM 1.0.5**  
file/duopen-lm

**Thunar 1.6.13**  
file/duopen-lm  
**XFE 1.42**  
xfiles/lee/xfce/  
solid65/lee/xfce/  
**НОТПИСКИ**  
**Android File Transfer** (GSI) Инструмент передачи файлов  
file/duopen-files/transfer  
**Beatsize v38** Музыкальная игра  
file/duopen-files/beatsize  
**Salpinx 2.4.0** Просмотрщик изображений  
duopen.com/dvz/salpinx  
**Dankable 2.4** Фото редактор  
www.dankable.lxd  
**DigIt 1.2** Инструмент помощи  
duopen.com/duopen/digit  
**DesktopFolder 1.0.5** Аксессар рабочего стола  
file/duopen/duopen/desktopfolder  
**Flameshot 0.5.0** Инструмент создания скриншотов  
file/duopen/duopen/flameshot  
**Hexd** (GSI) Игра-гонки  
file/duopen/hexd  
**LinuxShell (GSI)** Оболочка рабочего стола  
duopen.com/KDE/LinuxShell

Описание на обороте »

## Информация о диске

### Что-то потеряли?

Часто случается, что новые программы зависят от других программных продуктов, которые могут не входить в текущую версию вашего дистрибутива Linux.

Мы стараемся предоставить вам как можно больше важных вспомогательных файлов. В большинстве случаев, последние версии библиотек и другие пакеты мы включаем в каталог «Essentials [Главное]» на прилагаемом диске. Поэтому, если в вашей системе возникли проблемы с зависимостями, первым делом следует заглянуть именно туда.

### Форматы пакетов

Мы стараемся включать как можно больше различных типов установочных пакетов: RPM, Deb или любых других. Просим вас принять во внимание, что мы ограничены свободным пространством и доступными двоичными выпусками программ. По возможности, мы будем включать исходные тексты для любого пакета, чтобы вы могли собрать его самостоятельно.

### Документация

На диске вы сможете найти всю необходимую информацию о том, как устанавливать и использовать некоторые программы. Пожалуйста, не забывайте, что большинство программ поставляются вместе со своей документацией, поэтому дополнительные материалы и файлы находятся в соответствующих директориях.

### Что это за файлы?

Если вы новичок в Linux, вас может смутить изобилие различных файлов и расширений. Так как мы стараемся собрать как можно больше вариантов пакетов для обеспечения совместимости, в одном каталоге часто находятся два или три файла для различных версий Linux и различных архитектур, исходные тексты и откомпилированные пакеты. Чтобы определить, какой именно файл вам нужен, необходимо обратить внимание на его имя или расширение:

- » **имя\_программы-1.0.1.i386.rpm** — вероятно, это двоичный пакет RPM, предназначенный для работы на системах x86;
- » **имя\_программы-1.0.1.i386.deb** — такой же пакет, но уже для Debian;
- » **имя\_программы-1.0.1.tar.gz** — обычно это исходный код;
- » **имя\_программы-1.0.1.tgz** — тот же файл, что и выше этажом по списку: “tgz” — это сокращение от “tar.gz”;
- » **имя\_программы-1.0.1.tar.bz2** — тот же файл, но сжатый bzip2 вместо обычного gzip;
- » **имя\_программы-1.0.1.src.rpm** — также исходный код, но поставляемый как RPM-пакет для упрощения процесса установки;
- » **имя\_программы-1.0.1.i386.FC4.RPM** — двоичный пакет RPM для x86, предназначенный специально для операционной системы Fedora Core 4;
- » **имя\_программы-1.0.1.ppc.Suse9.rpm** — двоичный пакет RPM, предназначенный специально для операционной системы SUSE 9.x PPC;
- » **имя\_программы-devel-1.0.1.i386.rpm** — версия для разработчиков.

### Если диск не читается...

Это маловероятно, но если все же прилагаемый к журналу диск поврежден, пожалуйста, свяжитесь с нашей службой поддержки по электронной почте: [disks@linuxformat.ru](mailto:disks@linuxformat.ru)

Внимательно прочтите это перед тем, как использовать LXF DVD!

ВСЕ ДЛЯ СТАРТА В  
**Linux**  
Вставьте диск...  
Загрузи ПК...  
и Linux твой!



# Elementary OS

Красивый и простой Linux

» Быстрая установка  
» Защищено и безопасно  
» Все необходимые приложения

64-БИТНЫЙ



ПЛЮС!  
**ВОДНИ LINUX**  
Идеален для старых ПК



32-БИТНЫЙ

# Содержание



**Midnight Commander 4.8.20** Менеджер файлов терминала  
[www.midnight-commander.org](http://www.midnight-commander.org)  
**Posterizer (Gh)** Программа оптимизации изображений  
[bit.ly/1xI-posterizer](http://bit.ly/1xI-posterizer)

Kernel  
 Memtest86+  
 RAWRITE  
 SBM  
 WvDial

**УЧЕБНИКИ**

Stalke  
 PiGarden  
 Python

**ПОМОЩЬ**

Руководство новичка  
 Руководства  
 Ответы  
 Чаво (FAQ)

**ГЛАВНОЕ**

Checkinstall  
 Coreutils  
 Hardinfo

**ДОКУМЕНТАЦИЯ: 11 КНИГ О LINUX (НА АНГЛИЙСКОМ ЯЗЫКЕ)**

**Advanced Bash Scripting Guide** Подробное руководство по программированию на Bash  
**Bash Guide for Beginners** Руководство по Bash для начинающих  
**Bourne Shell Scripting** Начальное руководство по программированию на Bash  
**The Cathedral and the Bazaar** Классический текст Эрика Рэймонда [Eric S Raymond] «Собор и базар»  
**The Debian Administrator's Handbook** Руководство администратора, написанное разработчиками Debian  
**Dive Into Python** Учебник по программированию на Python  
**Intro to Linux** Начальное руководство по Linux

**Linux Dictionary** Словарь Linux, объясняющий специальную терминологию  
**Linux Kernel in a Nutshell** Описание ядра Linux, созданное одним из его выдающихся разработчиков — Греггом Краа-Хартманом [Greg Kroah-Hartman]  
**Rule** Знаменитый справочник  
**System Administrators Guide** Руководство по базовому администрированию Linux  
**GNU Tools Summary** Руководство по работе в командной строке и обзор основных утилит GNU

**Страница 12**  
**ДИСТРИБУТИВЫ**

**AV Linux (64-битный)**  
 Многоплатформый дистрибутив на базе Debian, с большой коллекцией ПО для производства аудио и видео. Включает настраиваемое ядро с потоковой обработкой IPQ, позволяющее работать с малой задержкой звука.  
**Kali Linux Light (64-битный)**  
 Дистрибутив на базе Debian с набором средств цифровой безопасности и криминалистики. Предлагает своевременные обновления безопасности, поддержку архитектуры ARM, выбор из четырех популярных настольных сред и плавное обновление до более новых версий.

Все дистрибутивы представлены ISO-образами, который можно записать на отдельный носитель, и загрузить в live-режиме прямо с LXF DVD. У всех присутствует возможность установки на жесткий диск.

**Пожалуйста, перед использованием Асдного Диска ознакомьтесь с инструкцией, опубликованной в журнале на стр. 109!**

**КОММЕНТАРИЙ** Присылайте ваши пожелания и предложения по электронной почте: [info@linuxformat.ru](mailto:info@linuxformat.ru)  
**ДЕФЕКТНЫЕ ДИСКИ** В случае обнаружения дефектов на данной странице, пожалуйста, по адресу [feedback@linuxformat.ru](mailto:feedback@linuxformat.ru)

Настоящий диск тщательно тестировался и проверялся на всех этапах производства, однако, как и в случае с любым другим ПО, мы рекомендуем вам использовать авторизованный сканер. Мы также рекомендуем всегда иметь под рукой актуальную резервную копию данных вашего жесткого диска. К сожалению, редакция Linux Format не в состоянии принимать на себя ответственность за любые повреждения, разрушения или иные убытки, которые могут повлечь за собой использование этого DVD. Предоставляем данную программу и данные, разрушения или иные убытки, которые могут повлечь за собой использование этого DVD, предоставляем на свой риск. Прежде чем устанавливать какое-либо ПО на компьютер, пожалуйста, с сайта разработчика узнайте все о вашей системе и оборудовании.  
 Тираж издательства ООО «Марком», 198682, Россия, Ленинградская область, Всеволожский р-н, дер. Юрки, Школьный ул., 7-а. Лицензия ИДТР ВАО N 77-03.

## Создание установочных дисков при помощи cdrecord

Самый быстрый способ записать ISO-образ на чистую матрицу — это обратиться к программе *cdrecord*. Для всех перечисленных ниже действий потребуются права суперпользователя-root. Сначала определите путь к вашему устройству для записи дисков. Наберите следующую команду:

```
cdrecord -scanbus
```

После этого на экране терминала должен отобразиться список устройств, подключенных к вашей системе. SCSI-адрес каждого устройства представляет собой три числа в левой колонке — например, 0,3,0. Теперь вы можете с легкостью записать образ на диск:

```
cdrecord dev=0,3,0 -v /путь к образу/image.iso
```

Чтобы упростить дальнейшее использование *cdrecord*, сохраните некоторые настройки в файле `/etc/default/cdrecord`. Добавьте по одной строке для каждого устройства записи (скорее всего, в вашей системе присутствует только одно такое устройство):

```
Plextor= 0,3,0 12 16M
```

Первое слово в этой строке — метка; затем после адреса SCSI-устройства вы должны указать скорость и размер буфера. Теперь можете заменить SCSI-адрес в командной строке на выбранную вами метку. Все будет еще проще, если вы добавите следующее:

```
CDR_DEVICE=Plextor
```

Для записи ISO-образа вам осталось набрать команду

```
cdrecord -v /path/to/image.iso
```

Если вы не принадлежите к любителям командной строки, в таком случае вам придет на помощь утилита *gcombust*. Запустите ее от имени root и выберите вкладку Burn и ISO 9660 Image в верхней части окна. Введите путь к образу, который вы хотите записать на диск, и смело нажимайте на Combust! Пока ваш образ пишется на диск, можете выпить чашечку кофе.

### Другая ОС?

Использовать Linux для записи компакт-диска не обязательно. Программы вроде *cdrecord* просто переносят двоичные данные на чистую матрицу. Все необходимые файлы уже включены в ISO-образ, который распознается любой операционной системой, будь то Linux, Windows, Mac OS X или AmigaOS.

### Нет устройства для записи дисков?

Если у вас нет устройства, с помощью которого можно было бы записать образ на диск, можно найти какого-нибудь друга или организацию, у кого есть компьютер с дисководом, и прожечь диск у них. Опять-таки, вам подойдет любая операционная система, способная распознать пишущий привод (см. выше).

Некоторые дистрибутивы умеют монтировать образы дисков и выполнять сетевую установку или даже установку с раздела жесткого диска. Конкретные методы, конечно, зависят от дистрибутива. За дополнительной информацией обращайтесь на web-сайт разработчика дистрибутива.



# LINUX FORMAT

# Подписывайтесь и читайте Linux Format на iPad или iPhone!

Доступно  
в AppStore



А если у вас Android, подпишитесь  
на Linux Format через Zinio!

 zinio™  
Доступно в Google Play



Журнал зарегистрирован Федеральной службой по надзору за соблюдением законодательства в сфере массовых коммуникаций и охране культурного наследия ПИ № ФФ77-21973 от 14 сентября 2005 года. Выходит ежемесячно. Тираж печатной версии 1000 экз., распространение электронной версии 30000 экз.

#### РЕДАКЦИЯ РУССКОЯЗЫЧНОЙ ВЕРСИИ

##### Главный редактор

Кирилл Степанов [info@linuxformat.ru](mailto:info@linuxformat.ru)

##### Литературный и выпускающий редактор

Елена Толстякова

##### Переводчики

Даниил Кривошеин, Светлана Кривошеина, Валентин Развозжаев, Валерий Смирнов, Елена Толстякова, Ирина Шулакова

##### Редактор диска

Александр Баракин

##### Верстка, допечатная подготовка

Сергей Рогожников

##### Технический директор

Андрей Смирнов

##### Директор по рекламе

Владимир Савельев [advert@linuxformat.ru](mailto:advert@linuxformat.ru)

##### Генеральный директор

Павел Фролов

##### Учредители

Частные лица

##### Издатель

ООО «Линукс Формат»

Отпечатано в типографии ООО «Типографский комплекс „Девиз“» 195027, Санкт-Петербург, Якорная ул., 10, корп. 2, литер А, пом. 44  
Заказ ТД-0000

#### РЕДАКЦИЯ АНГЛОЯЗЫЧНОЙ ВЕРСИИ

Редактор Нейл Мор [Neil Mohr] [neil.mohr@futurenet.com](mailto:neil.mohr@futurenet.com)

Научный редактор Джонни Бидвелл [Jonni Bidwell]

[jonni.bidwell@futurenet.com](mailto:jonni.bidwell@futurenet.com)

Художественный редактор Эфраин Эрнандес-Мендоса

[Efrain Hernandez-Mendoza] [efrain.hernandez-mendoza@futurenet.com](mailto:efrain.hernandez-mendoza@futurenet.com)

Выпускающий редактор Клиффорд Хоуп [Cliff Hope]

[clifford.hope@futurenet.com](mailto:clifford.hope@futurenet.com)

Директор редакции Грэм Барлоу [Graham Barlow]

Старший художественный редактор Джо Гулливер [Jo Gulliver]

#### ПОДГОТОВКА МАТЕРИАЛОВ

Джонни Бидвелл [Jonni Bidwell], Нейл Ботвик [Neil Bothwick], Алан Декстер [Alan Dexter], Нейт Дрейк [Nate Drake], Кент Ельчук [Kent Elchuk], Энди Келли [Andy Kelly], Джон Найт [John Knight], Джон Лэйн [John Lane], Адам Оксфорд [Adam Oxford], Лес Паундер [Les Pounder], Маянк Шарма [Mayank Sharma], Шашианк Шарма [Shashank Sharma], Валентин Синицын [Valentine Siniysyn], Александр Толстой [Alexander Tolstoy], Михалис Цукалос [Mihailis Tsoukalos], Евгений Балдин, Андрей Гондаренков, Дмитрий Кузнецов, Алексей Федорчук

Иллюстрации Шейн Коллиндж [Shane Collinge]

#### КОНТАКТНАЯ ИНФОРМАЦИЯ

UK: Linux Format, 30 Monmouth Street, Bath BA1 2BW

Тел. +44 (0) 344 848 2852, email: [linuxformat@myfavouritemagazines.co.uk](mailto:linuxformat@myfavouritemagazines.co.uk)

РОССИЯ: Санкт-Петербург, пр. Медиков, 5, корп. 7

Тел. +7 (812) 309-0686, e-mail: [info@linuxformat.ru](mailto:info@linuxformat.ru)

По вопросам сотрудничества, партнерства, оптовых закупок: [partner@linuxcenter.ru](mailto:partner@linuxcenter.ru)

**Авторские права:** статьи, переведенные из английского издания Linux Format, являются собственностью или лицензированы Future Publishing Ltd (Future plc group company). Все права зарегистрированы. Никакая часть данного журнала не может быть повторно опубликована без письменного разрешения издателя.

Все письма, независимо от способа отправки, считаются предназначенными для публикации, если иное не указано явно. Редакция оставляет за собой право корректировать присланные письма и другие материалы. Редакция Linux Format получает неэксклюзивное право на публикацию и лицензирование всех присланных материалов, если не было оговорено иное. Linux Format стремится оставлять уведомление об авторских правах всюду, где это возможно. Свяжитесь с нами, если мы не упомянули вас как автора предложенных вами материалов, и мы постараемся исправить эту ошибку. Редакция Linux Format не несет ответственности за опечатки.

Ответственность за содержание статьи несет ее автор. Мнение авторов может не совпадать с мнением редакции.

Все присланные материалы могут быть помещены на диски — CD или DVD, поставляемые вместе с журналом, если не было оговорено иное.

**Ограничение ответственности:** используйте все советы на свой страх и риск. Ни при каких условиях редакция Linux Format не несет ответственность за повреждения или ущерб, нанесенные вашему компьютеру и периферии вследствие использования тех или иных советов.

Linux — зарегистрированный товарный знак Линуса Торвальдса [Linus Torvalds]. "GNU/Linux" заменяется на "Linux" в целях сокращения. Все остальные товарные знаки являются собственностью их законных владельцев. Весь код, опубликованный в журнале, лицензирован на условиях GPL v3. См. [www.gnu.org/copyleft/gpl.html](http://www.gnu.org/copyleft/gpl.html)

За информацией о журналах, издаваемых Future plc group company, обращайтесь на сайт [www.futureplc.com](http://www.futureplc.com).

В апрельском номере

## Защитим свое личное... Побить GCHQ!

Раз уж Великобритания движется к китайскому уровню госнадзора, прикройте себя онлайн.

### Борьба с эксплойтами

Вслед Meltdown и Spectre, разберемся, как разработчики ядра Linux справляются с эксплойтами и латают дыры.

### Обработка снимков

Изучаем инструменты и Darktable в помощь обработке RAW-фото и оптимизации быстрорастущей коллекции.

### Исследуем Arduinio

Беремся за новый физический проект, чтобы разобраться, как создать и собрать устройство Arduino.

### Станьте конференцменом!

Собирайте встречи и не отчуждайтесь от любимых — тестируем лучшие открытые средства для телеконференций.

Содержание будущих выпусков может меняться — вдруг нас отправят на беседу с Большим Братом...



# РОББО КЛУБ™

## Образовательные кружки для детей

Робототехника ● Программирование ● 3D-моделирование ● 3D-печать



### Обучение: от игры к профессии

Для дошкольников

- ★ **Игровая робототехника**  
Возраст — от 5 лет.  
Требуется уверенный навык чтения

Для школьников

- ★★ **Стартовый**  
Основы знаний
- ★★★ **Базовый**  
Специальность
- ★★★★ **Углубленный**  
Профессия, соревнования,  
собственные проекты

А ТАКЖЕ

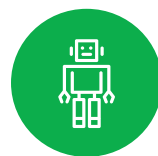
- Праздничные мастер-классы
- Городской лагерь на время каникул
- **Курсы повышения квалификации для педагогов**



Инновационная программа обучения

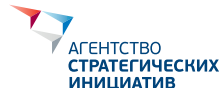


Собственная методическая база



Российские технологии и оборудование

## ВООБРАЖАЙ! ПРОГРАММИРУЙ! СОЗДАВАЙ!



[www.robboclub.ru](http://www.robboclub.ru)

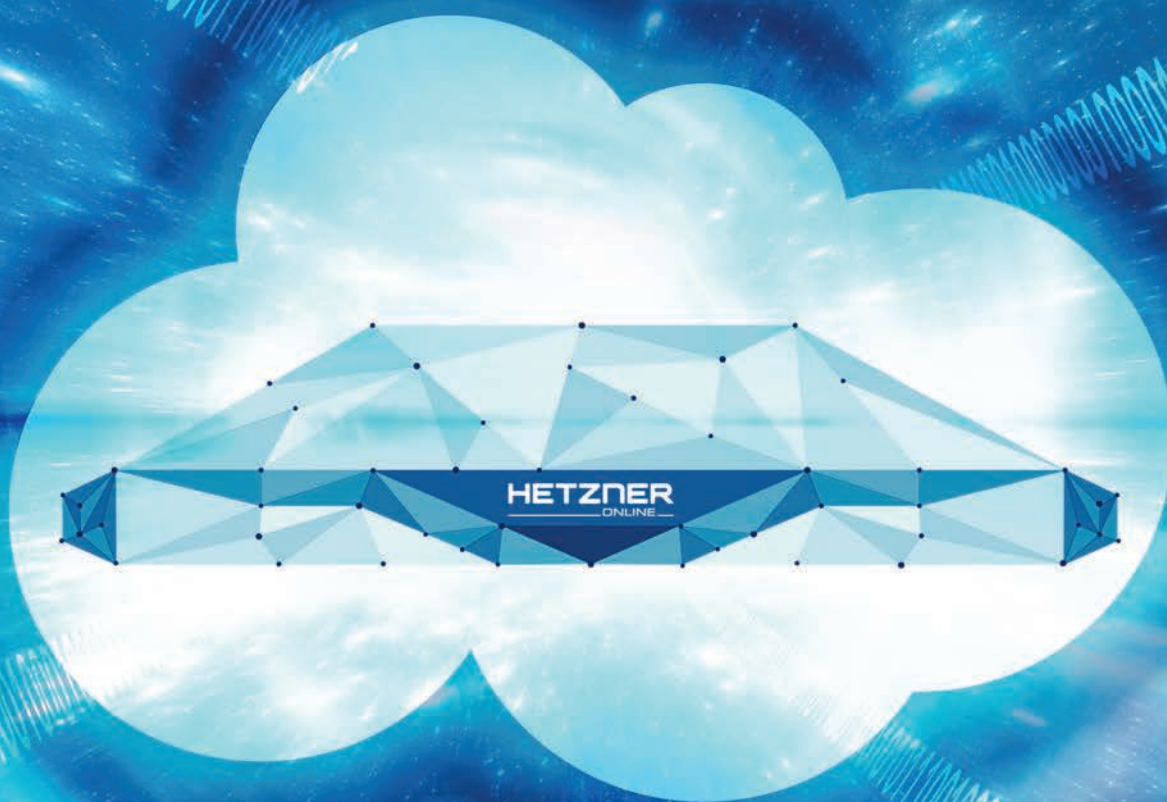
8 800 777 2985

[vk.com/robboclub](https://vk.com/robboclub) ● [info@robboclub.ru](mailto:info@robboclub.ru)

**HETZNER**  
ONLINE

**БУДУЩЕЕ  
НАЧИНАЕТСЯ СЕЙЧАС!**

С НОВЫМ  
**HETZNER CLOUD**



**ru.hetzner.com**